

Opinnäytetyö (AMK)
Tietotekniikan koulutusohjelma
Mediatekniikka
2014

Susanna Savander

3D-HAHMON VALMISTAMINEN PELIMOOTTORIA VARTEN



TURUN AMMATTIKORKEAKOULU
TURKU UNIVERSITY OF APPLIED SCIENCES

OPINNÄYTETYÖ (AMK) | TIIVISTELMÄ

TURUN AMMATTIKORKEAKOULU

Tietotekniikan koulutusohjelma | Mediatekniikka

2014 | 46

Yliopettaja, FT Mika Luimula

Susanna Savander

3D-HAHMON VALMISTAMINEN PELIMOOTTORIA VARTEN

Tämän opinnäytetyön tarkoituksena oli valmistaa 3D-hahmo pelimoottoria varten käyttäen Blenderiä. Työn alussa tutustuttiin 3D-mallinnuksen eri osa-alueiden teoriaan sekä hyvään työtapaan luoda peliin sopiva hahmo. Tämän jälkeen hahmo mallinnettiin, rigattiin, skinnattiin ja animoitiin. Viimeisenä hahmo vietiin Unity3D-pelimoottoriin, jossa sille tehtiin lopputestaus.

Jokaisessa hahmon luomisvaiheessa hahmoa kuvattiin ja testattiin mallinnusohjelman sisällä, jotta työssä ei olisi tarvinnut palata aiempaan vaiheeseen useaan otteeseen. Tämä opetti hyvää työtapaa, jossa 3D-mallissa olevat virheet korjattiin heti niiden syntymisen jälkeen eikä työtä tarvinnut aloittaa uudestaan.

Opinnäytetyön valmistuttua tuloksena oli hahmo, jonka pystyi suoraan ohjelmoimaan toimivaksi peliympäristössä. Prosessi oli yksinkertainen, mutta aloittavalle mallintajalle haasteellinen työrytmin ja taitojen puuttumisen vuoksi. Tämä opinnäytetyö dokumentoi koko prosessin, jonka tarkoituksena on saada pelissä toimiva hahmo mallinnettua ja valmistettua. Se tarjoaa tietoa siitä, millainen hyvä pelihahmo on ja miten sen voi luoda.

ASIASANAT:

3D-mallinnus, animaatio, peliteknologia, pelimoottori, Blender, Unity3D, riggaaminen

BACHELOR'S THESIS | ABSTRACT

TURKU UNIVERSITY OF APPLIED SCIENCES

Degree Programme in Information Technology | Digital Media

2014 | 46

Principal Lecturer, Ph.D. Mika Luimula

Susanna Savander

PREPARING A 3D CHARACTER FOR A GAME ENGINE

The purpose of this thesis was to prepare a 3D character for a game engine using Blender. The thesis starts with an introduction of the different steps of modelling and animating a game character. The thesis then explains how the character was modelled, rigged, skinned and animated. The last part of the thesis presents how the character was imported to Unity3D and prepared it for the final testing.

In each phase of the preparation process, the character was observed and tested inside the modelling program. That helped to find an efficient working method so the process would not have to take back steps. It also prevented starting over the whole project.

The outcome of this thesis was a character that could be programmed to work in a game environment. The process was simple, but starting a modeler was challenging due to the lack of principal rules and working method. This thesis documents the whole process of making a proper game character that could be used in a game. It is useful guide for beginners and provides information of what a good game character is and how it is made.

KEYWORDS:

3D modelling, animation, game technology, game engine, Blender, Unity3D, rigging

SISÄLTÖ

SANASTO	6
1 JOHDANTO	7
2 3D-MALLINNUS JA TEKNOLOGIAKATSAUS	9
2.1 3D-mallintaminen	9
2.2 Mallinnustekniikat	10
2.3 Käytetyimmät 3D-mallinnusohjelmat peleissä	12
3 HAHMON LUONTI	13
3.1 Hahmomallinnuksen haasteet	13
3.2 Hahmon mallintaminen referenssikuvia käyttäen	14
3.2.1 Hahmon mallinnus	14
3.2.2 Hahmon yksityiskohdat	21
3.3 Hahmon riggaus	23
3.3.1 Riggaus Armature-työkalulla	25
3.3.2 Hahmon skinnaus	28
3.3.3 Rigin ja skinnauksen testaus	30
3.4 Hahmon animaatio	32
3.4.1 Idle-animaatio	34
3.4.2 Kävely-animaatio	35
3.4.3 Juoksu-animaatio	36
3.4.4 Hyppy-animaatio	37
4 TULOSTEN TESTAUS	39
4.1 Hahmon vieminen Unityyn	39
4.2 Hahmon animaatioiden testaus Unityssa	41
5 POHDINTA	44
LÄHTEET	46

LIITTEET

Liite 1. Referenssikuva.

KUVAT

Kuva 1. Tavallisen 3D-mallin ja NURBS-mallin ero (T-Splines 2013).	11
Kuva 2. Silmän muodon rajaaminen.	15
Kuva 3. Ulommaisia vertexejä juuri vedetty Extrudella jalkateriä varten.	17
Kuva 4. Valmiit jalkaterät.	17
Kuva 5. Valmis lantio, josta erottaa lantion ja vatsan seudun muodot.	18
Kuva 6. Kainalot rajattuina.	19
Kuva 7. Virhe mallinnuksessa: hahmolla on vain yksi nivel etusormessa.	20
Kuva 8. Valmis käsi liitettynä muuhun vartaloon.	21
Kuva 9. Iiriksen muoto ennen paikoilleen asettamista.	22
Kuva 10. Hahmonnettu hahmo.	23
Kuva 11. Kasvojen ilmeet mallinnettuna PyDriven avulla (Mullen & Andaur 2010, 150).	24
Kuva 12. Ihmis-rig juuri lisättynä hahmolle.	26
Kuva 13. Pole Targetin asetusten muuttaminen.	27
Kuva 14. Root Bone asetettuna kahvojen vanhemmaksi.	28
Kuva 15. Skinnaus Weight Paintilla.	29
Kuva 16. Vasemmalla Blenderin automaattinen painotus ja oikealla rajattu luun vaikutusalue.	30
Kuva 17. Epäonnistunut skinnaus ensimmäisessä testissä.	31
Kuva 18. Kolmannessa testauksessa osoittautuneet skinnauksen virheet.	32
Kuva 19. Vasemmalla pelattava hahmo ja oikealla sama hahmo cut scenessa (Final Fantasy X HD Remake 2014).	33
Kuva 20. Keyframen asettaminen.	35
Kuva 21. Kävelyn animaation toiminta loopissa.	36
Kuva 22. Hahmon juoksusykli.	37
Kuva 23. Hyppy-animaation testaus.	38
Kuva 24. Autodesk FBX –tiedostomuodon viemisasetukset.	39
Kuva 25. Unityn perusnäkyvä, johon on viety Standart Assets –paketin osia.	40
Kuva 26. Hahmo vietynä näkymään.	41
Kuva 27. CharacterController lisättynä animaation kontrolloijaksi.	41
Kuva 28. Animator-välilehden vakionäkyvä.	42
Kuva 29. Animaatioiden testaus Unityssa.	43
Kuva 30. Testauksessa Unityn puolella oli hyppy-animaatiossa virheitä.	43

SANASTO

Edge	Reunaviiva, mikä erottaa eri facet
Face	Tasopinta, joista 3D-malli koostuu
Idle	Animaatiossa kohteen toimeton tila
Keyframe	Piste aikajanalla, johon tietyt komennot asetetaan animoitaessa
Layer	Tasot ohjelmistossa, joille voi tallentaa dataa
Loop	Itseään loputtomiin toistava käsky tai komento
Mesh	3D-objektin verkko, joka koostuu edgeistä, faceista ja vertexeistä
NURBS	Non-uniform rational basis spline, algoritmi pintojen mallintamiseen
Polygoni	Monikulmio, joka koostuu edgeistä, facesta ja vertexeistä
Rig	Riggaus, luiden asettaminen 3D-hahmolle
Skinning	Skinnaus, luiden vaikutusalueiden asettaminen tai muokkaaminen
Skripti	Komentosarja hahmon toiminnallisuutta varten
Vertex	Verteksi eli piste, missä kaksi tai useampi edge liittyy yhteen

1 JOHDANTO

Tässä opinnäytetyössä tutkitaan videopeliin tarkoitettua 3D-hahmon mallintamista, riggausta ja animaatiota prosessina sekä kartoitetaan nykyisiä ammattilaisten käyttämiä tekniikoita. Tästä prosessista ei ole nykyisin montaa yhtenäistä tieteellistä tutkimusta, vaan ohjeistukset perustuvat käyttäjäkokemuksiin. Työn tarkoituksena on suunnitella mahdollisimman kattava ja tehokas työtapa pelihahmon tärkeimpien elementtien luomisesta ja niiden käytöstä. Tämän takia prosessi käydään vaihe vaiheelta läpi ja hahmon toimivuus testataan.

Työn alussa luodaan Edge Modelling -tekniikalla ihmishahmo käyttäen referenssikuvaa. Referenssikuva koostuu hahmosta, mikä on kuvattu sekä edestä että sivulta. Nämä tekniikat valittiin mallintamisen helpottamiseksi, koska on helpompaa seurata referenssikuvaa kuin mallintaa niin sanotusti vapaalla kädellä. Vaihtoehtona olisi ollut myös Box-modelling-tekniikka, mutta se ei tarjonnut yhtä tarkkaa lähestymistapaa kuin valitut tekniikat. Box-modelling on yleisin mallinnuskeino, eli hahmo rakennetaan kuutiosta, jolloin mallintaminen on vapaampaa. Esimerkiksi Blender-mallinnusohjelmassa aloitusnäkyssä on valmiina kuutio, jonka muokkaamisesta voi aloittaa mallinnusprosessin.

Työn tavoitteena on selvittää yksinkertainen keino rigata hahmo, joten hahmon valmistuttua tutustutaan riggauksen työkaluihin ja tuodaan hahmolle Blenderin valmis ihmis-rig. Hahmon riggausprosessi on nopea ja työn eteneminen tehostuu, kun ei tarvitse luoda jokaista tarvittavaa luuta erikseen, vaan voi muotoilla valmiin rigin luut vastaamaan oman hahmon raajoja. Riggaamisen jälkeen hahmo skinnataan eli määritellään alueet, joihin luut vaikuttavat. Tähän on olemassa kaksi keinoa, joihin molempiin tutustutaan. Riggaamisen ja skinnaamisen onnistumista seurataan liikuttelemalla hahmoa eri asentoihin, jolloin mahdolliset virheet rigissä tai skinnauksessa voidaan välttää tai korjata.

Hahmolle animoidaan tärkeimmät peleihin liittyvät liike-elementit, eli tässä tapauksessa idlaus, käveleminen, juokseminen ja hyppääminen. Samalla tarkistetaan, pitääkö prosessissa palata taaksepäin muokkaamaan hahmon rigiä. Tätä

pyritään kuitenkin välttämään prosessissa mallintamalla hahmo asentoon, josta animoiminen on helppoa.

Työ toteutetaan kokonaisuudessaan Blenderillä, koska se on käyttäjäystävällinen, helposti saatavilla sekä ilmaisohjelma eikä sen käyttämiseen vaadita maksullista lisenssiä, kuten esimerkiksi Autodeskin 3ds Maxin käyttämiseen. Työn lopuksi testaus toteutetaan Unity3D:llä ja tarkastellaan, miten hahmo viedään pelimoottoriin niin, että sen animaatiot säilyvät ja miten ne toimivat.

2 3D-MALLINNUS JA TEKNOLOGIAKATSAUS

2.1 3D-mallintaminen

3D-mallinnuksella tarkoitetaan 3D-mallinnusohjelmalla kolmeakselisessa karteesisessa koordinaatistossa olevan objektin muokkaamista (Slick 2014c). Karteesisessa koordinaatistossa pisteen paikan määrittää koordinaatit, jotka määräytyvät kolmen normaalin, eli suoran mukaan (The Free Dictionary 2014). Oikeassa elämässä objektit esiintyvät luonnostaan kolmeulotteisina, mutta tietokoneella muodostettavassa grafiikassa kolmiulotteisuus esitetään matemaattisesti. 3D-mallinnusohjelmat laskevat automaattisesti matemaattisen puolen, kun käyttäjä voi asettaa 3D-objektin graafisessa käyttöliittymässä. 3D-mallinnusohjelmiin on yleensä myös sisäänrakennettu ”real-time render engine”, eli reaaliajassa toimiva hahmontajamoottori, joka näyttää realistisesti 3D-mallin. (Slick 2014c.)

Perinteisesti 3D-mallinnus tehdään vertexien, edgejen ja facejen avulla, mutta on olemassa muitakin tekniikoita, joihin tutustutaan myöhemmin. Vertexillä tarkoitetaan pistettä, joka on asetettu 3D-ohjelmassa koordinaatistoon ja siihen voi liittää edgejä. Edget muodostavat tulevalle 3D-mallille muodon ja niiden välille voi luoda facen. Facen avulla 3D-malli saa muotonsa. Näistä kolmesta elementistä koostuu polygoni, missä voi olla useampi vertex ja edge, mutta vain yksi face. (Siltakorpi 2014, 5.)

Nykyään melkein jokainen julkaistu peli sisältää 3D-objekteja. Vaikka moni julkaisee 2D-pelejä, niiden tekemiseen käytettävät ohjelmistot sisältävät usein 3D-mallinnusta tai mahdollisuuden siihen. (Silverman 2013.) Tällaisia pelejä, jotka ovat 2D-pelejä, mutta toimivat 3D-ympäristössä, kutsutaan 2.5D-peleiksi. 2.5D-mallin pelejä ovat esimerkiksi Kirby 64: The Crystal Shards ja Shadow Complex.

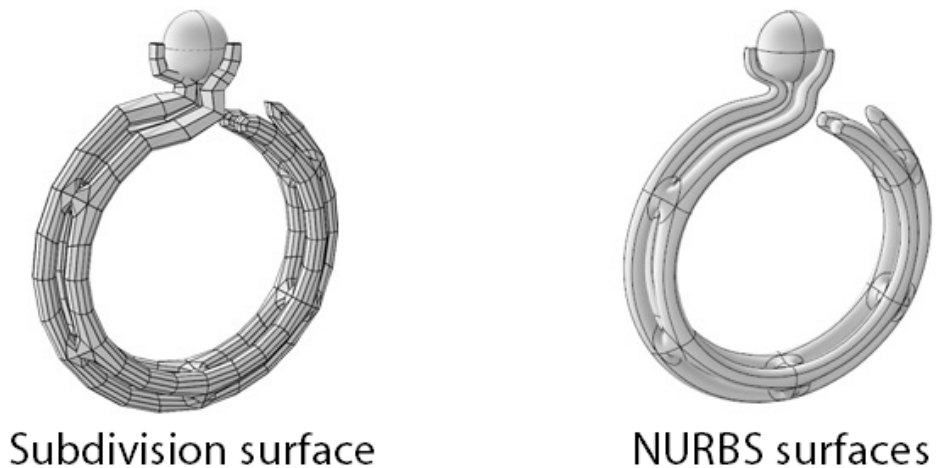
2.2 Mallinnustekniikat

Yksi yleisimmistä mallinnustekniikoista on box-modelling-tekniikka. Box-modellingissa muokataan polygonia, joka voi olla mikä tahansa geometrinen objekti. Tätä tekniikkaa käyttäessä työskennellään vaihe vaiheelta. Ensimmäisessä vaiheessa työ aloitetaan pieniresoluutioisella meshillä, jonka muotoja muokataan. Työtä pehmennetään eri työkaluilla ja polygoneja jaetaan pienempiin osiin, jolloin niitä on enemmän ja malliin pystyy tekemään yksityiskohtia. Tällöin objektin mesh sisältää tarpeeksi polygoneita, jotta se muodostaa halutun objektin. (Slick 2014b.)

Edge modelling –tekniikassa objektin polygonit muodostetaan pala palalta. Muotoja tehdään vertexeillä ja edgeillä, joiden tyhjät välit täytetään, jolloin muodostuu face. Edge modelling –tekniikkaa käytetään tyypillisesti vaikeissa mallinnuskohteissa. Kasvojen mallinnuksessa käytetään usein tätä tekniikkaa, koska on tehokkaampaa määrittää vertexien ja edgejen avulla esimerkiksi silmän muoto kuin muokata se 3D-pallosta. (Slick 2014b.)

Procedural modelling –tekniikalla tarkoitetaan erilaisten tekniikoiden kirjoja, joiden avulla luodaan automaattisesti 3D-malli. Useimmin sitä käytetään ympäristön luomiseen, esimerkiksi puiden ja maisemien luontiin. Tekniikalla on myös mahdollista luoda rakennuksia tai kaupunkeja. Procedural-malleissa on yleensä hyvin vähän dataa, joka määrittää mallin käyttökohteet. Näiden avulla tekniikka luo sattumanvaraisesti yhden tai useamman 3D-objektin. (Rotenberg 2005.)

NURBS-mallinnustekniikassa luodaan kontrollipisteitä ja –polygoneita, jotka muodostavat erilaisia aaltoja asetettujen pisteiden välille (RealWorld Graphics 2006). Tämä tekniikka eroaa muista mallinnustekniikoista siinä, että malleja ei luoda vertexien, edgejen ja facejen avulla. NURBS:in avulla luodaan peleihin esimerkiksi köynnöksiä. (Slick 2014b.) Tekniikan avulla tehdyt mallit ovat kevyitä käyttää pelimoottoreissa, koska ne eivät sisällä yhtä paljon dataa kuin yleisimmät 3D-objektit (kuva 1).



Kuva 1. Tavallisen 3D-mallin ja NURBS-mallin ero (T-Splines 2013).

3D-skannauksella tarkoitetaan laitetta, joka tunnistaa objektin ja analysoi sen tietokoneohjelman avulla 3D-objektiksi. Se kerää tietoa objektin muodosta ja muista ominaisuuksista. 3D-skannausta käytetään perinteisesti teollisuudessa, mutta sen käyttö on lisääntynyt peliteknologiassa yksinkertaisten esineiden muuttamiseksi 3D-objektiksi. Yksi 3D-skannauksen huonoista puolista on se, että skannerin keräämä tieto voi vääristyä, jos objekti on kiiltävä, peilaava tai läpinäkyvä. (Wikipedia 2014a.)

Digital Sculpting –tekniikka on digitaalista kuvanveistoa, jossa 3D-objektia muokataan Sculpting-työkalulla. Työkalussa on erilaisia toimintoja, esimerkiksi miten laajalle alueelle se vaikuttaa ja miten voimakkaasti. Digital Sculpting –tekniikalla toteutetaan hahmon ensimmäiset taiteelliset 3D-versiot. (Freeman 2013.)

Image Based Modelling –tekniikka on kuvassa esiintyvän objektin tai maiseman muuttamista 3D-ympäristöön. Tekniikan toteuttaa ohjelmaa varten suunniteltu algoritmi. Kuvasta muuttamista käytetään yleensä silloin, kun manuaalinen 3D-mallinnus ei ole mahdollista. (Slick 2014a.)

Lathe Modelling –tekniikan avulla mallinnetaan symmetrisiä 3D-objekteja. Tällaisia objekteja ovat tavallisimmin erilaiset juomapullot, kupit ja vaasit, jotka pystyy luomaan yksinkertaisella muodon antavalla janalla, joka pyörytetään akse-

linsa ympäri. Tämä antaa objektille symmetrisen muodon. (Wang & Chang 2008, 118.)

2.3 Käytetyimmät 3D-mallinnusohjelmat peleissä

Autodeskin 3ds Max on yksi käytetyimmistä mallinnusohjelmistoista videopeli-kehityksessä sekä muissa teollisuuden aloilla. Se on laajempi käyttötarkoituksiltaan kuin samaan tuoteperheeseen kuuluva Maya 3D. 3ds Max on aiemmin ollut vain mallinnustarkoituksessa, mutta nykyään sen ominaisuuksien parannuttua sen avulla myös animoidaan ja luodaan esimerkiksi varjostuksia. Maya 3D:n avulla puolestaan luodaan hahmomalleja ja animaatioita. (PixelSmith Studios 2012.)

Cinema 4D:tä pidetään helppokäyttöisenä mallinnusohjelmana: mallintajan ei tarvitse tietää ohjelmasta kaikkea saadakseen aikaa tuloksia. Se tarjoaa mahdollisuuden mallintaa eri mallinnustekniikoilla sellaisen objektin kuin mallintaja haluaa, eikä rajoita esimerkiksi vain teknisiin malleihin. Cinema 4D:llä pystyy myös mallintamaan hahmon sekä riggaamaan ja animoimaan sen. (Pichot 2013.)

Blender on ainoa ilmainen ammattilaikäytössä oleva 3D-mallinnus- ja animaatio-ohjelma. Se tarjoaa mahdollisuuden siihen, että ohjelma tai tietokone ei tarvitse lisenssiä sen käyttämiseen, joten projektia voi jatkaa millä tahansa tietokoneella ilman lisenssisopimuksien rikkomista. Aikaisemmin Blender oli vaikeakäyttöinen, koska se vaati ohjelman omien pikakomentojen osaamista, jotta mallintaja pystyi toteuttamaan halutut toiminnot. Palautteen perusteella Blender on muokannut ulkoasuun käyttäjäystävällisemmäksi. (Mullen & Andaur 2010, 1–2.)

3 HAHMON LUONTI

3.1 Hahmomallinnuksen haasteet

Perinteisesti pelihahmoa suunniteltaessa yhdistyy sekä graafinen sisältö että luova kirjallinen osuus. Pelattavalle hahmolle luodaan persoonallisuus eli mahdollisesti hahmon luonne, perhetausta, mieltymykset sekä muut taustakartoitukset, jotka tekevät hahmosta erilaisen ja ihmismäisen (Totten 2012, 4). Persoonallinen hahmo herättää pelaajan kiinnostuksen tutustua sekä hahmoon että peliin, jolloin taustatyö hahmon luomisessa korostuu.

Kun hahmolle on luotu persoonallisuus, alkaa prosessin visuaalinen osuus. Hahmosta piirretään eri vedoksia eli konseptikuvitusta, ja niistä yhdistellään sopivat teemat yhteen. Tällöin myös 3D-mallintajat saavat esitietoa hahmosta ja pelin miljööstä, mikä helpottaa 3D-mallinnusta. Päähahmon konseptikuvitus on tämän takia yksi tärkeimmistä osa-alueista 2D-artistien osalta, mutta yleensä konseptitaidetta käytetään myös pelin markkinoinnissa. (Totten 2012, 4.)

Aloitettaessa 3D-mallinnusta tulisi päättää, mallinnetaanko hahmo käyttäen neljää vai kolmea verteksiä facessa. Tavallisesti hahmomallinnuksessa käytetään neljää, jotta hahmosta saataisiin mahdollisimman tarkka ja elävän näköinen. Parhaimmat 3D-mallit sisältävätkin paljon yksityiskohtia, jolloin niiden polygonien määrä on suuri. Peleihin mallinnettaviin hahmoihin käytetään yleisesti kolmea verteksiä yhtä facea kohti, koska tällöin hahmo on kevyt käyttää pelissä ja vertexien yhteislukumäärä ei nouse korkeaksi. Lopputuloksena neljän vertexin tekniikalla luodut hahmot ovat pehmeämpiä hahmontaessa, kun taas kolmen tekniikalla luodut hahmot ovat kulmikkaampia. (Slick 2014a.) Tässä työssä hahmo mallinnetaan käyttämällä Edge Modelling- tekniikan avulla neljää verteksiä jokaisessa facessa.

Hahmon 3D-mallinnuksessa suositaan niin sanottua T-asentoa, jossa hahmon kädet on mallinnettu hartioden tasalle ja hahmon jalat ovat vähän erillään toisistaan. Tällöin voidaan mallintaa helpommin 90° kulman avulla, mutta mallinnuk-

sen jälkeen hahmoa animoitaessa voi esiintyä jäykkyyttä ja robottimaista liikettä. Tätä voi välttää mallintamalla hahmon kädet noin 45° kulmaan ja taivuttamalla hiukan hahmon polvia. (Toten 2012, 7.)

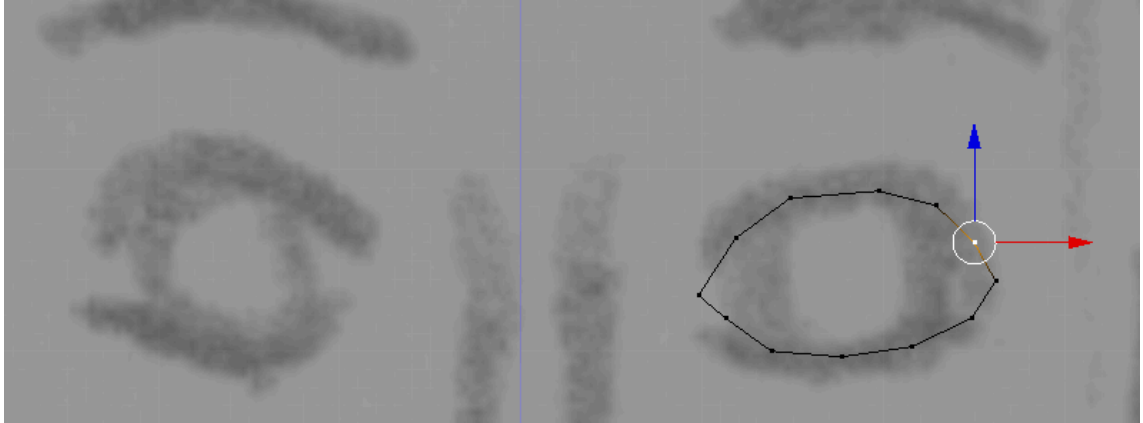
3.2 Hahmon mallintaminen referenssikuvia käyttäen

Mallintamisprosessia varten hahmosta piirrettiin referenssikuva (liite 1), joka toimi hahmon mittasuhteiden määrittäjänä. Referenssikuva ei sisältänyt vaatteita tai muita yksityiskohtia, koska tavoitteena oli mallintaa ihmisen anatomiaa mukaileva hahmo. Tämän vuoksi referenssinä ollut hahmo oli pelkistetty versio lopullisesta 3D-mallista. Työn rajauksessa päätettiin myös, että vain hahmon liikkeet rigataan ja animoidaan, joten kasvojen yksityiskohtiin huomion kiinnittäminen ei tässä tapauksessa ollut tarpeellista. Referenssikuvaa käyttäessä tulisi myös muistaa, ettei mittasuhteet välttämättä vastaa toisiaan sekä edestä piirretyssä että sivulta piirretyssä, joten mallintaminen onnistuu seuraamalla joko tarkasti toista referenssikuvaa tai mallintamalla kahden kuvan välille kompromissi.

Referenssikuvaa laitettaessa Blenderiin tuli tarkistaa, että hahmo asettuu oikealle kohdalle koordinaatistossa Y-akselin suhteen. Jos hahmo on asetettu tasaisesti akselin molemmille puolille, onnistuu vain toisen puolen mallintaminen kun toinen puoli kopioituu samanlaisena Mirror-työkalun avulla. Tämä vähentää työmäärää ja hahmon molemmista puolista tulee identtiset.

3.2.1 Hahmon mallinnus

Hahmon mallintaminen aloitettiin silmien rajaamisesta ja vertexien paikoilleen asettamisesta (kuva 2). Tämän jälkeen rajoja laajennettiin, jotta saatiin lisää polygoneja silmän muodon mallintamiseen. Silmien ympärille lisättiin edgejä, jotta silmiin saatiin esimerkiksi luomet luonnollisen näköisiksi. Lopuksi silmät liitettiin yhteen lisäämällä faceja ja tuomalla nenän muotoa esille.



Kuva 2. Silmän muodon rajaaminen.

Nenän rakennetta jäljiteltiin sivulta piirretyn referenssikuvan avulla asettamalla vertexejä paikoilleen ja yhdistelemällä niitä edgejen ja facejen avulla. Samaa tapaa käyttäen jatkettiin huulien mallintamiseen. Koska hahmolle ei ollut tarkoitus tehdä kasvorigiä, suun eri osia – hampaita ja kieltä – ei tarvinnut mallintaa. Huulet aseteltiin yhteen ja mahdollinen sisäosa suusta vedettiin hahmon pään sisään ja liitettiin toisiinsa.

Huulien alareunan keskikohdasta vedettiin edgejä leukaan. Tästä pisteestä noustiin ylöspäin hahmon pään ääriviivoja myöten ylös pääläelle. Kun vertexit oli saatu oikeisiin kohtiin myös sivunäkymässä, täytettiin tyhjät kohdat faceilla. Muotoja pyöristettiin mukaillemaan poskipäitä, posken pientä kuoppaa ja leukaa. Kun hahmon alaosa oli valmis, siirryttiin mallintamaan hahmon otsaa ja pääläkeä. Pääläeltä siirryttiin takaraivon mallintamiseen ja siitä niskaan.

Kun hahmon pää oli saanut muodon, alettiin mallintamaan korvaa. Korvan muotoa mallinnettiin ensin asettelemalla vertexejä sen ulkoreunoihin referenssikuvan avulla. Tämän jälkeen jatkettiin samalla tavalla kuin silmää mallintaessa: lisättiin faceja ja osaa nostettiin kun toisia taas upotettiin. Korvan yläreunoista lisättiin faceja päätä kohti ja aseteltiin ne menemään alaspäin, jotta korvan muoto tulisi paremmin esiin. Myös korvalehteen käytettiin samaa tekniikkaa, mutta facejen määrä oli vähäisempi, koska korvalehti ei vaatinut yksityiskohtia.

Korvien päähän yhdistämisen jälkeen päätä pystyi viimeistelemään. Hahmolle oli laitettu Mirror-työkalun lisäksi myös Subdivision Surface –työkalu, mikä peh-

mentää hahmon muotoja ja antaa paremman kuvan mahdollisesta lopputuloksesta. Hahmolle lisättiin kasvoihin yksityiskohtia, esimerkiksi poskipäitä levitettiin ja nenää suurennettiin. Työn jälki tarkastettiin vaihtamalla näkymää Edit Modesta Object Modeen, mikä näyttää työn siltä miltä se näyttäisi hahmontaessa. Kun pään muotoihin oltiin tyytyväisiä, jatkettiin leukaperistä ja takaraivosta kaulaa hiukan alaspäin. Myös hahmon silmäkuopat täytettiin silmämunan pinnalla.

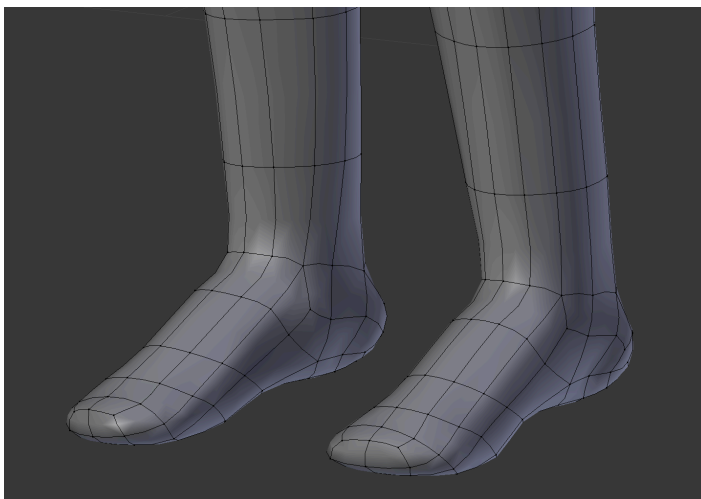
Jalkojen mallinnus aloitettiin lisäämällä hahmon nilkkojen kohdalle vertexeistä ja edgeistä koostuvan ympyrän. Vertexien lukumääräksi asetettiin 12, jolloin yksityiskohtien lisääminen polviin ja kantapäähän on helpompaa. Tämän jälkeen käytettiin Extrude-komento, mikä luo toisen samanmuotoisen kappaleen ja yhdistää polygonit facella. Extruden avulla jalkojen pituutta kasvatettiin haaraväliin saakka. Polvien kohdalle lisättiin edgejä, jotta yksityiskohtien tekeminen helpotuisi.

Kun jalat olivat mallinnettuina, siirryttiin mallintamaan kantapäätä. Kantapään tekeminen tuotti muutaman kerran ongelmia kun vertexejä piti poistaa sekä lisätä, jotta yksityiskohtien tekeminen onnistuisi. Tämä johti siihen, ettei välillä vertexien määrä jokaisessa facessa ollut neljä. Tämä korjattiin lisäämällä edgejä, mikä lisäsi myös vertexien määrää. Jalkateriä varten valittiin jokainen ulommainen vertex jaloista ja niitä kasvatettiin Extrudella (kuva 3).



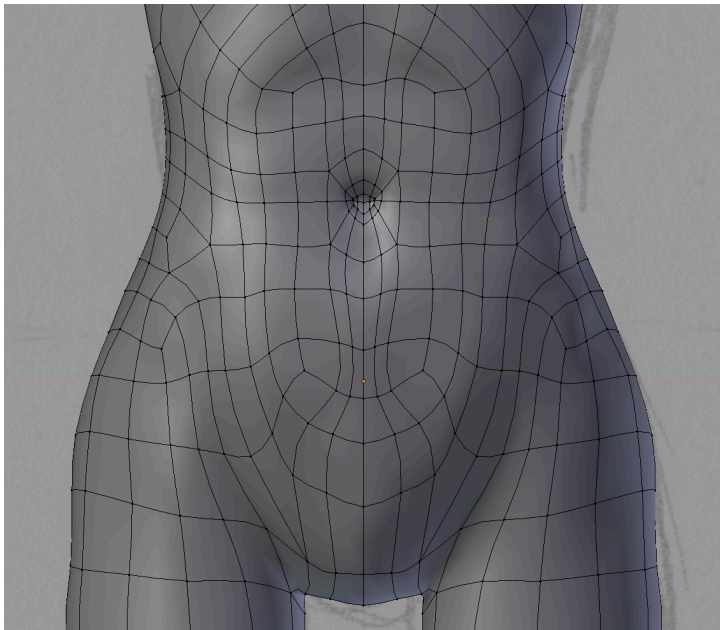
Kuva 3. Ulommaisista vertexeistä juuri vedetty Extrudella jalkateriä varten.

Jalkaterien mallinnus onnistui myös Extrude-komennolla, mutta jokaisen lisäyksen jälkeen lisättyä face-ympyrää pienennettiin. Jalkaterien ollessa tarpeeksi pitkät, niiden ylä- ja alaosat liitettiin yhteen ja muotoja pyöristettiin. Oikeiden jalkojen muotoa jäljiteltiin myös kasvattamalla paikkaa, jossa isovarvas olisi ja muiden varpaiden kohtia pienennettiin. Myös jalan holvikaarta jäljiteltiin tuomaan paremmin oikean jalan muotoa esille. Työssä ei mallinnettu varpaita, koska hahmolla tulisi olemaan vaatteet, eikä tällöin varpaita näkyisi (kuva 4).



Kuva 4. Valmiit jalkaterät.

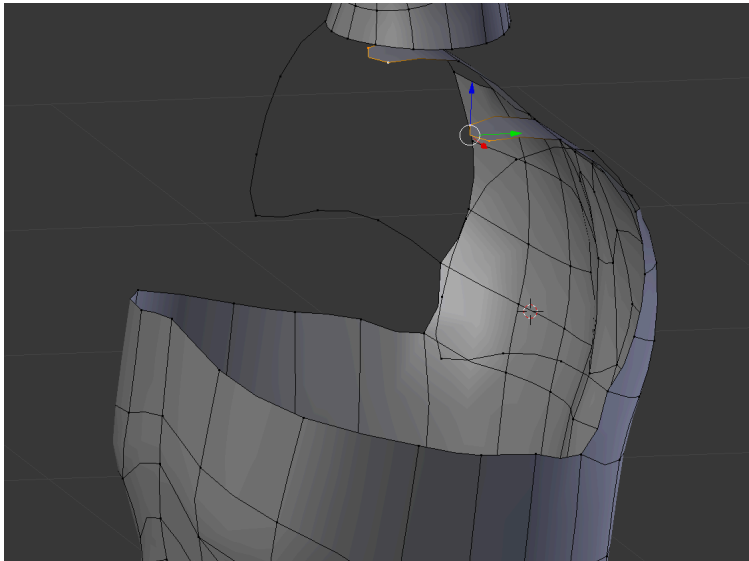
Hahmon lantiota lähdettiin mallintamaan nostamalla Extruden avulla kylkien puolelta kolme facea. Tämän jälkeen hahmon haaraväli liitettiin yhteen ja hahmon lantio-vatsa-alueella mallinnettiin pyöreämmillä muodoilla. Navan kohdalla käytettiin Extrude-komentoa työntämään napaa sisäänpäin kehoon, jotta se näyttäisi realistiselta. Vyötäröä kavennettiin Scale-työkalun avulla, mikä helpotti työtä niin, ettei jokaista verteksiä tarvinnut siirrellä käsin. Myös vatsalihaksia tuotiin esiin anatomian harjoittelun ja valaistuksen tarkistamisen vuoksi. Lantioluun näkyvyyttä jäljiteltiin korostamalla kohtaa, jossa luun tulisi olla koholla (kuva 5). Takapuolta vedettiin ulospäin ja pyöristettiin, sekä keskikohtaa vedettiin takaisin sisäänpäin, jotta pakarat saataisiin erottumaan.



Kuva 5. Valmis lantio, josta erottaa lantion ja vatsan seudun muodot.

Rintakehä tehtiin laajentamalla faceja, jotta hahmon rintakehä näyttäisi enemmän miehen rintakehältä. Tämä sai aikaan halutun kulmikkaan olemuksen hahmon kehon yläosaan. Samaan aikaan mallinnettiin myös selkää, missä piti kiinnittää huomiota siihen, että lapaluut asettuvat oikein. Lapaluiden kohtaa piti kierittää muutama aste vinoon, jotta uloimmat osat laskivat. Tämän jälkeen lapaluita vedettiin ulospäin ja selkärankaa hieman hahmon sisään. Lapaluiden mallinnuksen jälkeen kainalo rajattiin samalla tavalla kuin silmä ja korva aikai-

semmin (kuva 6). Tällöin hartioiden ja kaulan välisen alueen pystyi mallintamaan Extrude-komennon avulla.



Kuva 6. Kainalot rajattuina.

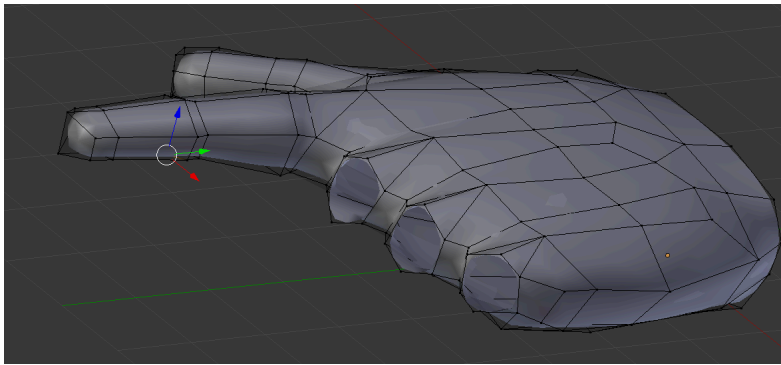
Hahmon rintakehän valmistuessa piti valita tapa, jolla liittää kaula ja alavartalo yhteen. Tätä ennen piti varmistaa, että kaulassa on sekä pään että alavartalon puolella sama määrä vertexejä. Kun vertex-määrät täsmäsivät, valittiin työkaluksi Bridge-toiminto, mikä luo faceja liitettävien kohtien välille. Kaulaan lisättiin poikittain kulkevia edgejä, jotta hahmolle pystyisi mallintamaan kaulan lihaksia ja muita muotoja.

Käsien mallinnus aloitettiin hartioista. Hartioiden korkeampi kohta luotiin lisäämällä malliin yksi ylimääräinen face, mitä pystyi leventämään ja nostamaan, jolloin siitä tuli hartian näköinen. Tästä jatkettiin Extrude-komennolla luomalla vapaasti faceja sopiva määrä kyynärpäähän kohtaan asti. Kyynärpäähän lisättiin kaksi edge ympyröimään aluetta ja kyynärpää muotoiltiin hahmon selkäpuolelle. Hahmon loppu käsivarsi toteutettiin käyttämällä Extrudea ja Scalea. Käsivarren aukko jätettiin tyhjäksi, jotta siihen pystyisi liittämään käden myöhemmin.

Kämmenten ja sormien mallinnus aloitettiin luomalla vertexeistä ja edgeistä koostuva ympyrä, joka tulisi olemaan ranne. Tämän jälkeen luotiin neljä ympyrää, mutta niiden vertexien lukumäärä oli alhaisempi. Nämä ympyrät aseteltiin

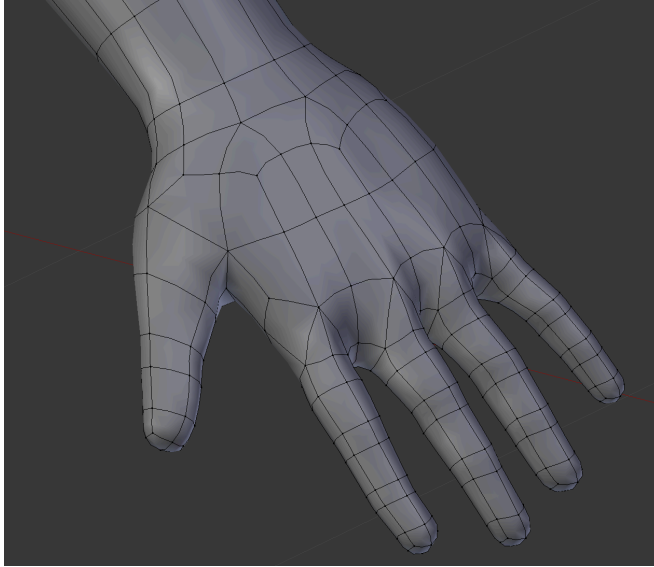
kauemmas isosta ympyrästä, koska ne toimisivat aukkoina, joista sormet lähtevät. Tämän jälkeen sormien ja ranteen välistä tyhjää tilaa täytettiin faceilla.

Sormien mallintaminen aloitettiin peukalosta. Tällöin tuli huomioida peukalon nivelkohta sekä sormen asema suhteessa kämmeneen. Peukalon ollessa valmis siirryttiin mallintamaan etusormea. Sormea mallintaessa tuli kiinnittää huomiota siihen, että muissa sormissa on kaksi niveltä ja mallintaa ne (kuva 7). Kun etusormi oli valmis, sen pystyi kopioimaan ja liittämään tyhjien aukkojen kohdalle. Välit korjattiin facejen avulla ja sormia venytettiin tai lyhennettiin riippuen siitä mikä sormi oli kyseessä.



Kuva 7. Virhe mallinnuksessa: hahmolla on vain yksi nivel etusormessa.

Käden ollessa valmis, se kopioitiin samalle tasolle, jossa muu vartalo oli. Liitetty käsi oli melkein yhtä suuri kuin koko hahmo itsessään, joten sitä pienennettiin Scale-työkalun avulla. Käsi aseteltiin paikalleen ja liitettiin käsivarteen Bridge-työkalun avulla. Lopuksi käden muotoilua vielä korjattiin (kuva 8).



Kuva 8. Valmis käsi liitettynä muuhun vartaloon.

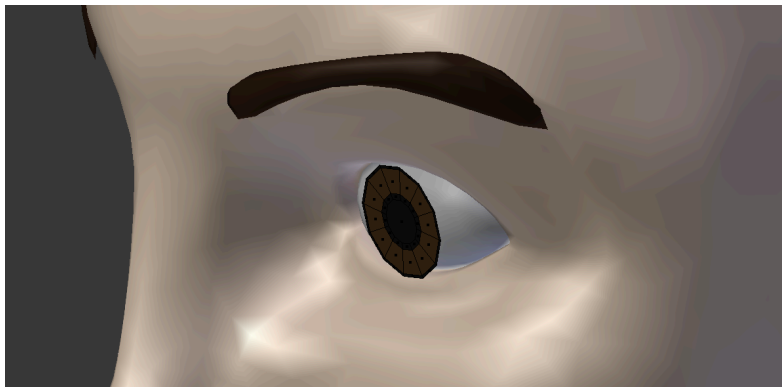
3.2.2 Hahmon yksityiskohdat

Hahmolle mallinnettiin yksityiskohtina kulmakarvat ja pupillit vaatteiden lisäksi kun vartalon mallinnusprosessi oli valmistunut. Mallintaminen aloitettiin housuista, koska ne tulisivat jäämään osittain T-paidan alle. Mallinnus alkoi yksittäisen vertexin lisäämisestä vyötärölle ja jatkettiin vartalon ympäri Extrude-komennolla, jolloin vertexien väliin syntyi edgejä. Extrudella pystyttiin myös peittämään hahmon lantio, jolloin housut muistuttivat enemmän hametta. Tämä korjautui lisäämällä faceja Extruden avulla hahmon sivulle alaspäin nilkkoihin asti. Tämän jälkeen lahkeet yhdistettiin lisäämällä tarvittava määrä vertexejä ja yhdistämällä ne faceilla. Housuihin lisättiin harmaasävyinen materiaali, jotta ne erottuisivat muusta työstä.

T-paita tehtiin samalla tekniikalla kuin housut, mutta paidan asettelussa ilmeni ongelmia. Paidan polygonimäärä oli vähäinen, jolloin facet olivat isoja. Tämä aiheutti sen, että työssä ei ollut tarpeeksi vertexejä, mitä siirrellä, jotta paita asettuisi lihasten päälle niin, ettei hahmon ihoa näy paidan alta. Paitaa jouduttiin suurentamaan ja taas asettelemaan, jotta se istuisi hahmolle. T-paidalle laitettiin sininen materiaali antamaan väriä.

Hahmo jätettiin kaljuksi, mutta kulmakarvat lisättiin tuomaan ihmismäisyyttä. Kulmakarvojen teko aloitettiin lisäämällä yksittäinen vertex, josta jatkettiin Extruden avulla lisäämään edgejä ja vertexejä kulmakarvan muotoon. Kulmakarva täytettiin faceilla ja sitä paksunnettiin kasvattamalla sen kokoa Extrude-komennon avulla. Tämän jälkeen kulmakarvaan lisättiin Mirror-työkalu, joka kopioi kulmakarvan toiselle puolelle hahmoa. Kulmakarva aseteltiin mukailemaan kasvojen muotoa iholla.

Viimeisenä hahmolle tehtiin silmiin iirikset. Tämä aloitettiin luomalla vertexeistä ja edgeistä koostuvan ympyrän, jota pienennettiin ja siirrettiin silmien kohdalle. Tämän jälkeen käytettiin Extrude- ja Scale-komentoja iiriksen pinnan tekemiseen. Keskiosa liitettiin yhteen, jolloin ympyrään ei jäänyt aukkoa. Tämän jälkeen iiriksen eri osille annettiin värejä, jotta silmien väri ja pupillin musta väri korostuisivat (kuva 9). Iiris aseteltiin lopulta silmän pinnalle.



Kuva 9. Iiriksen muoto ennen paikoilleen asettamista.

Hahmon valmistumisen jälkeen siitä tehtiin ensimmäinen hahmontaminen, eli tuotettiin hahmosta kuva. Tiedostoon lisättiin kaksi kohdevalaisinta, jotta hahmo näkyisi kuvassa, koska hahmontaminen mallintaa hahmon sellaisena kuin se olisi oikeassa elämässä. Myös kamera lisättiin, jotta hahmontaminen onnistuisi (kuva 10). Tiedostoon ei lisätty muuta, esimerkiksi tyhjiä taustoja, koska ne olisi jouduttu kuitenkin poistamaan.



Kuva 10. Hahmonnettu hahmo.

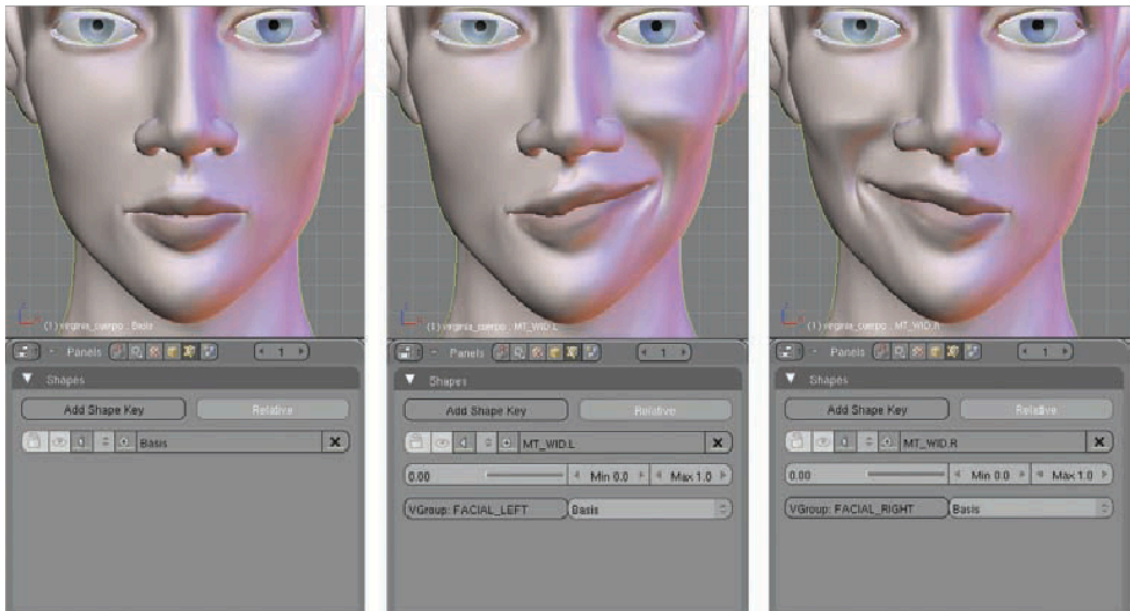
3.3 Hahmon riggaus

Riggaamisella tarkoitetaan prosessia, missä luodaan kontrollisarjoja liikuttamaan 3D-objektin meshiä. Tällöin luodaan luita, jotka liikuttavat vertex-joukkoja samaan aikaan. (Vasconcelos 2011, 5.) Tämä mahdollistaa hahmon liikkumisen, mikä ei muuten onnistuisi liikuttamalla hahmoa käsin, koska hahmo sisältää usein lukumäärältään suuren joukon vertexejä. Riggaamista pidetään vaikeimpana asiana 3D-animoinnissa (Vasconcelos 2011, 6).

3D-hahmon kaksi tärkeintä muistisääntöä on rigin pitäminen mahdollisimman yksinkertaisena ja rigin tekeminen tarpeeksi monimutkaiseksi. Rigin tulee olla tarpeeksi yksinkertainen, jotta animaattori pystyy käyttämään sitä ilman ohjausta tai hankaluuksia. Toisaalta jos rig on liian yksinkertainen, niin hahmon liikkeet eivät ole realistisia. Tyypillisesti jokainen animaatioprojekti on ainutkertainen, joten ei ole olemassa oikeaa tai väärää tapaa rakentaa hahmon rigiä. (Vasconcelos 2011, 6.)

Koska riggaamiseen ei ole oikeaa tai väärää tapaa, ammattilaiset tavallisesti yhdistävät monta eri tekniikkaa rigata hahmo, jotta jokaiselle hahmolle valmistuisi sopiva rig. Blender tarjoaa monta eri tapaa löytää sopiva ratkaisu hahmon riggaamiseen: Armaturesin ja PyDriversin. Armaturesilla voidaan luoda luita, jotka vaikuttavat tiettyyn alueeseen meshissä. Tämän lisäksi määritellään myös painoalueet vertexeille, eli mihin alueisiin luut vaikuttavat eniten ja mihin vähiten. (Mullen & Andaur 2010, 135–136.) Tätä kutsutaan skinnaukseksi. Skinnaamisella varmistetaan, ettei esimerkiksi toisen luun painoalue tule liian lähelle, jolloin se painaa hahmon meshiä kasaan tai venyttää sitä. Tätä ongelmaa voi esiintyä esimerkiksi, jos käsien painoalueet ovat liian suuret ja ne painavat hahmon kylkiä ja mahaa hahmon sisälle.

PyDriverilla tarkoitetaan rigiä, mitä ohjataan Python-koodikielellä. Se mahdollistaa monien muotojen yhdistämisen 3D-mallissa sekä niiden liikuttamisen. Rigin koodaus voi olla hyvin monimutkainen, koska Blenderissä ei ole asetettu rajoituksia koodin pituuteen. PyDriven avulla on mahdollista kontrolloida esimerkiksi yhden luun avulla vain kasvojen ilmeitä suun avulla (kuva 11). (Mullen & Andaur 2010, 149.)



Kuva 11. Kasvojen ilmeet mallinnettuna PyDriven avulla (Mullen & Andaur 2010, 150).

3.3.1 Riggaus Armature-työkalulla

Hahmon riggaus toteutettiin Blenderin omalla työkalulla, koska se oli ensimmäistä kertaa riggaavalle helpoin käyttää. Blenderillä on mahdollista luoda kahdella eri tavalla hahmon rig: lisäämällä joko yhden luun kerrallaan tai lisäämällä kokonaisen ihmisrigin. Molemmilla tavoilla luotu rig toimii Armature-työkalun avulla. Valmiissa rigissä luut on aseteltu vastaamaan ihmisen luita lukuun ottamatta jalkaa, johon valmiissa rigissä tulee vain yksi varvasluu jaloissa. Tätä käytetään todennäköisesti sen takia, että hahmot ovat harvoin paljain jaloin, jolloin tarvitsisi mallintaa myös varpaat. Rig tulee skaalata oikean kokoiseksi mallinnetun hahmon sisälle sekä asettaa luut oikeisiin kohtiin, esimerkiksi sormien luiden sijoittelu vastaamaan sitä miten luut menevät ihmisellä. Rigin asettelu tehdään Edit Modessa, Pose Modessa määritetään hahmon asentoja ja luiden liikealueita. On tärkeää muistaa näiden kahden tilan ero, jotta halutut muutokset jäisivät hahmossa voimaan.

Tälle hahmolle lisättiin valmis ihmisrig, koska sen hallinta on aloittelijalle helpompaa. Luita yksi kerrallaan lisätessä tulisi ottaa huomioon lisäsjärjestys sekä lapsi-vanhempisuhteet luissa. Tämä tarkoittaa sitä, että kun yksi luu liikkuu niin toinenkin luu liikkuu. Valmis rig oli liian pieni hahmolle (kuva 12), joten koko rigiä piti skaalata suuremmaksi. Tämän lisäksi jokaista kolmea selkärangan luita venytettiin, jotta lantio saataisiin oikealle kohdalle. Myös solisluun alueen luita venytettiin, jotta käden luut lähtisivät oikeasta kohdasta. Tämän jälkeen käden luut aseteltiin paikoilleen. Asettelyä helpotti valitsemalla Armaturen asetuksesta X-Ray, mikä pitää luut esillä, vaikka ne on upotettu hahmon sisälle.

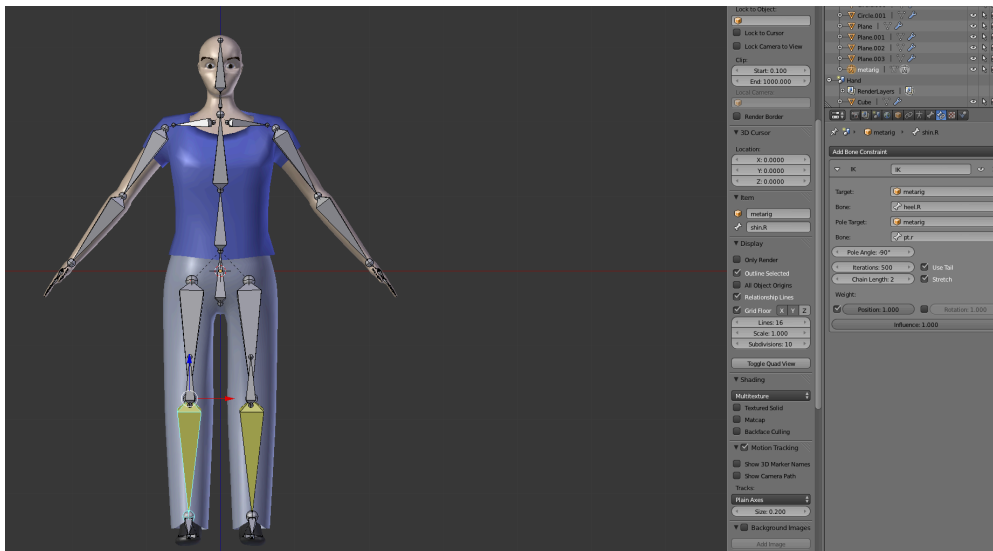


Kuva 12. Ihmis-rig juuri lisättynä hahmolle.

Asettelyn jälkeen hahmon jalasta kantapäähän kohdalta katkaistiin luiden lapsi-vanhempi –suhde. Tämä tehtiin sen takia, jotta myöhemmin tätä hahmon ulkopuolelle jäänyttä luuta voitaisiin käyttää kahvana, jolla liikutetaan jalkaa. Suhteen katkaisemisen jälkeen siirryttiin Pose Modeen, jotta voitaisiin säätää jalan maksimiliikesuunnat. Pose Modessa valittiin ensin kahvana käytettävä luu ja sen jälkeen sääriluu, mistä tehtiin aktiivinen luu. Valinta muuttaa rigiä niin, että kahvalla pystytään kontrolloimaan sääriluuta. Lopuksi sääriluun Chain Length -arvoa muutettiin nolasta kahteen, jotta luu vaikuttaisi vain kahteen muuhun luuhun. Kahden luun vaikutusalue varmistaa sen, että liike vaikuttaa vain jalan luihin, eikä liikuta esimerkiksi lantiota. Samoin myös jalan sivuttaista liikerataa haluttiin rajoittaa, jolloin liikuttaminen lukittiin Y- ja Z-akseleilla. X-akselilla sitä rajoitettiin alkuperäisestä -180° :sta 0° :n kulmaan. Tällöin jalka liikkui samalla tavalla kuin ihmisen jalka, eikä noussut liian ylös tai taipunut liikaa sivuittain. Toiminnot tehtiin molempiin jalkoihin.

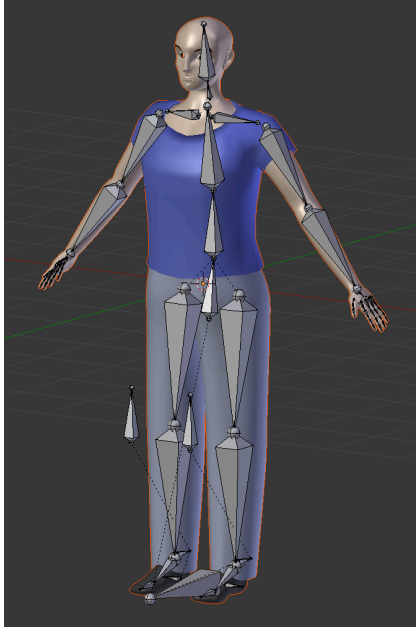
Tulevan animointiprosessin helpottamiseksi rigiin lisättiin kaksi Pole Targetia, eli luuta, jotka tässä rigissä ohjaavat jaloissa polvien liikkeitä. Pole Targetit lisättiin

Edit Modessa polvien eteen ja niiden nimiä muutettiin, jotta ne erotettaisiin muista luista, koska Blender antaa ”Bone”-nimen lisätyille luille. Nimeämisen jälkeen siirryttiin Pose Modeen, valittiin sääriluu aktiiviseksi luuksi ja avattiin luiden asetukset sivupaneelista. Paneelista valittiin Pole Targetin kohdalle ”meta-rig”, jolloin Bone-kohdasta pystyi valitsemaan halutun Pole Target –luun (kuva 13). Valinnan jälkeen polvi siirtyi pois paikoiltaan, minkä pystyi korjaamaan säätämällä Pole Anglea 90°:n kulmaan. Toiselle puolelle tuli tehdä samat muutokset, mutta Pole Angle oli vastaavasti toisella puolella -90°. Lopuksi muutettiin lapsi-vanhempisuhteita Edit Modessa, missä Pole Targetista tehtiin kahvan vanhempi. Tällöin Pole Target liikkui jalkaa liikuttaessa, eikä jäänyt paikoilleen.



Kuva 13. Pole Targetin asetusten muuttaminen.

Pole Targetin lisäksi rigiin lisättiin luu, jota kutsutaan Root Boneksi. Root Bonen avulla koko hahmoa pystyy nostamaan ylöspäin helposti, koska muuten kaikki luut olisi valittava ja tämän jälkeen hahmoa nostettava. Luu lisättiin Edit Modessa ja se aseteltiin jalkojen väliin jalkaterien kohdalle. Tämän jälkeen Root Bone asetettiin lapsi-vanhempi –suhteessa vanhemmaksi ja sen lapsiksi kahvoina toimivat luut (kuva 14). Valmiissa ihmisrigissä oli lantioluu asetettu valmiiksi toimimaan Root Bonena, joten se tuli asettaa lapseksi ja uusi Root Bone asetettiin sen vanhemmaksi. Tämä varmisti sen, että rig pysyy siinä kohdassa, mihin se on asetettu. Kokonaisen rigin liikuttaminen vaati vain Root Bonen liikuttamisen.



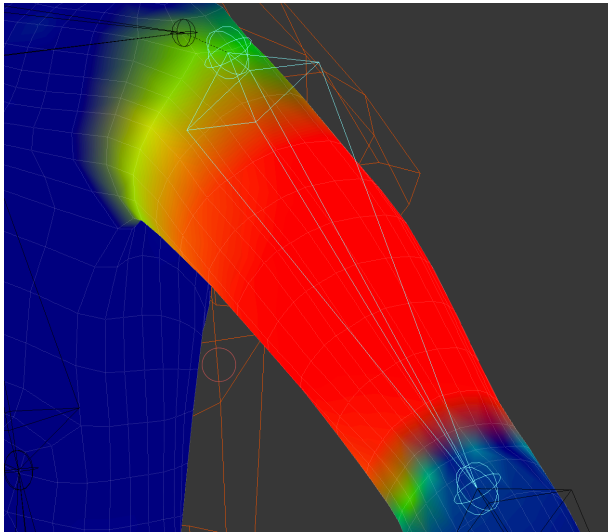
Kuva 14. Root Bone asetettuna kahvojen vanhemmaksi.

3.3.2 Hahmon skinnaus

Hahmon skinnaus aloitettiin valitsemalla työtilaksi Object Mode ja klikkaamalla hahmon meshit ja rig aktiivisiksi. Tämän jälkeen valittiin Armature Deform –valikon alta With Automatic Weights –kohta. Tällöin Blender laskee ja määrittää automaattisesti hahmon luiden painoalueet sekä muokkaa tarvittavia asetuksia, jotta luiden ja meshin suhteet jakautuvat oikein (Herreño 2011, 262). Toiminnan jälkeen skinnausta tuli kuitenkin muuttaa, koska Blender laski liian suuria vaikutusalueita, jolloin hahmon mesh ei välttämättä käyttäydy halutulla tavalla.

Blenderissä on mahdollista muokata painoalueita valitsemalla Weight Paint Mode työtilaksi. Tällöin valitaan haluttu luu ja tarkastetaan sen painoalueet. Alueet, joihin luu vaikuttaa vähiten, ovat tummansinisiä. Eniten vaikuttavat alueet ovat väriltään punaisia. Väri vaihtuu oranssista keltaiseen ja vihreään sitä mitä vähemmän luu vaikuttaa kyseiseen kohtaan (kuva 15). Tämä työtila voi kuitenkin olla aloittelijalle vaikeampi kuin meshin työkalujen avulla vertex-ryhmien muokkaaminen, koska alueet maalataan hiiren kanssa ja maalauksen voimakkuus voi olla liian pieni tai suuri kuin halutun tuloksen saaminen vaatisi. Tämän takia

työssä kokeiltiin ensin Weight Paint Modea, mutta siirryttiin sen jälkeen käyttämään vertex-ryhmien muokkaamista.



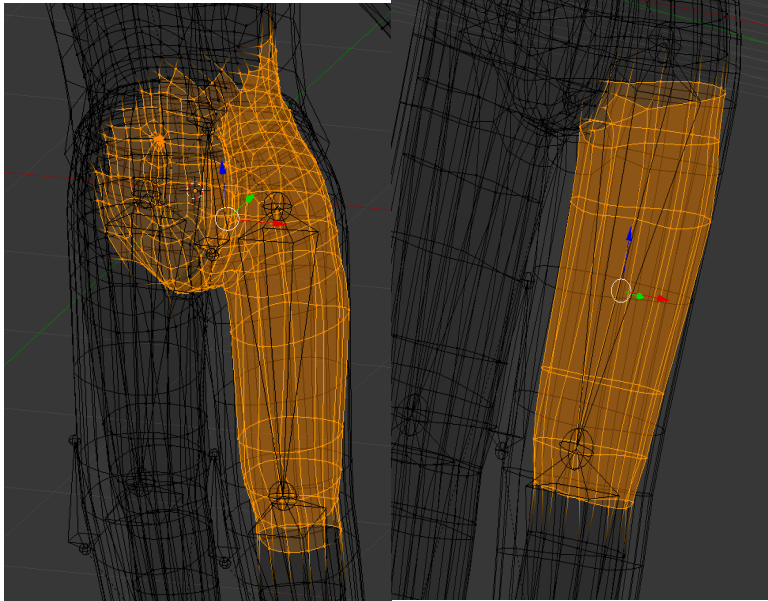
Kuva 15. Skinnaus Weight Paintilla.

Skinnauksen korjaaminen vertex-ryhmien avulla aloitettiin valitsemalla Object Modessa hahmon ihon mesh aktiiviseksi. Valinnan jälkeen siirryttiin Edit Modeen, jossa sivupalkkiin valittiin työtilaksi Object Data. Työtilasta etsittiin Vertex Groups, josta pystyi näkemään hahmon rigissä käytetyt luut listattuna. Valitsemalla halutun luun listasta ja klikkaamalla Select-painiketta pystyi näkemään luun vertex-ryhmän, jota luun liikuttaminen ohjaisi.

Ensimmäisenä korjattiin pään skinnausta poistamalla vertexien valinta sellaisista kohdista, joihin vaikutusalue haluttiin jättää (kuva 16). Tarpeettomien vertex-ryhmien ollessa valittuna painettiin Remove-painiketta, joka poisti tarpeettomat vertexit lopullisesta vaikutusalueesta. Kun pään valitsi uudelleen aktiiviseksi, vain halutut vertex-ryhmät olivat aktiivisena meshissä eli näkyivät oranssina.

Samat valinnat tehtiin jokaisen luun vertex-ryhmille rajaten aluetta, joihin automaattinen painoalueiden laskenta oli ne asettanut. Suurimmat virheet olivat olkapäissä ja lantiolla, joihin Blender oli laskenut kaksinkertaiseksi sen alueen, jota luu liikuttaisi. Käsien luiden nimet ja niiden mukaan ryhmien korjaus tuotti ongelmia sen Blenderin tavasta, jolla luut nimitään. Tässä tapauksessa luut olivat samannimisiä, mikä olisi voinut tuottaa ongelmia, mikäli esimerkiksi koko

vertex-ryhmä olisi vahingossa poistettu. Ongelmia tuottivat myös ylimääräiset luut, mitkä osoittautuivat hahmon ulkopuolella oleviksi ohjaaviksi luiksi.



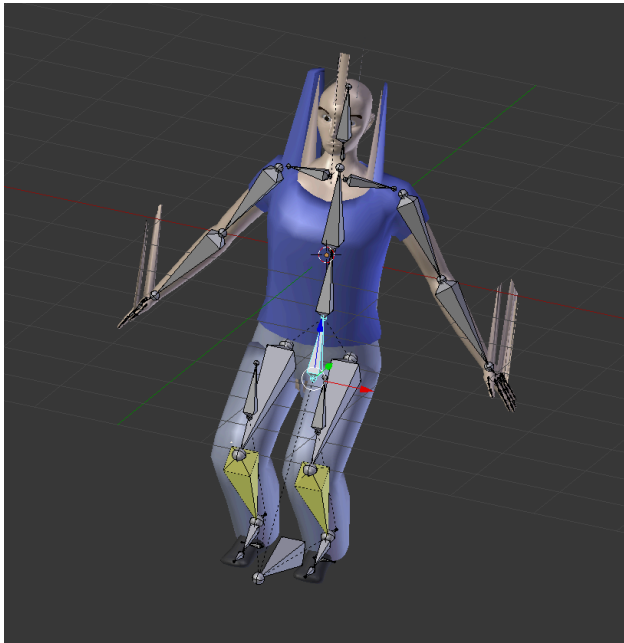
Kuva 16. Vasemmalla Blenderin automaattinen painotus ja oikealla rajattu luun vaikutusalue.

Ihomeshin skinnauksen jälkeen oli sama prosessi tehtävä myös vaatteille, jotta ne mukautuisivat liikkeisiin. Vaatemeshien painotusalueiden määrittäminen oli helpompaa kuin ihon, koska housujen liikkeisiin vaikuttivat vain jalkojen, lantion ja selkärangan luut. Paitaan vaikutti vain lantio, selkäranka, rintakehä sekä pieni osa kaulan vaikutusalueesta.

3.3.3 Rigin ja skinnauksen testaus

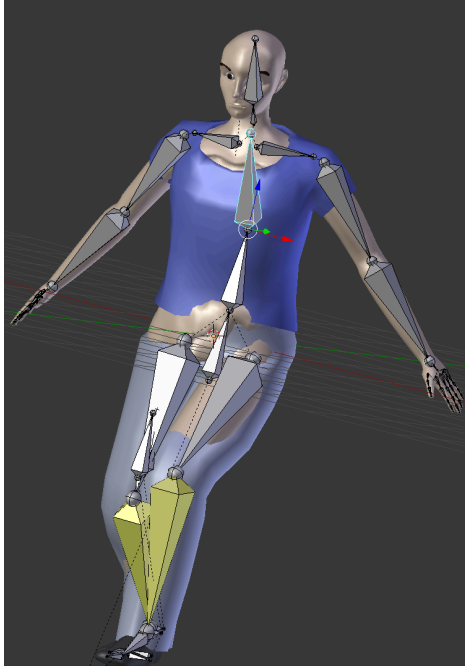
Rigin ja skinnaamisen onnistumista testattiin Pose Modessa liikuttamalla hahmoa. Ensimmäisellä testikerralla huomattiin, että hahmon ihomeshistä oli jäänyt valitsematta silmät ja kulmakarvat, jolloin ne jäivät paikoilleen, kun hahmon päätä liikutti. Tähän ei löytynyt parempaa korjausta kuin tehdä skinnaus ja vertex-ryhmät uudestaan. Tämä onnistui kuitenkin paljon nopeammin kuin ensimmäisellä kerralla, eikä se vaikuttanut vaatteiden mesheihin.

Toisessa testauksessa osoittautui, että meshin vertex-ryhmiä oli rajattu liikaa. Tällöin osa vertexeistä jäi paikoilleen, kun hahmoa liikutti (kuva 17). Tällöin palattiin jokaiseen virheelliseen skinnaukseen eri mesheissä ja kasvatettiin vaikutusalueita valitsemalla isompi alue kuin aikaisemmin ja klikkaamalla Assign-painiketta. Assign asettaa valitut vertexit luun vertex-ryhmään poistamatta mitään. Isoimmaksi ongelmakohtaksi osoittautui ihomeshistä suu, koska suuta oli vedetty hahmon pään sisään näkymättömiin. Näihin vertexeihin pääsi käsiksi vaihtamalla katselua Wireframe-tilaan, jolloin näki myös hahmon sisälle.



Kuva 17. Epäonnistunut skinnaus ensimmäisessä testissä.

Kolmannessa testauksessa osoittautui, että hahmon vaatteet painuivat hahmon sisälle, kun hahmoa liikutti (kuva 18). Ongelma korjattiin asettamalla automaattiset painoalueet uudestaan vaatteille ja hahmo testattiin uudestaan. Vaatteet painuivat vain hiukan tietyissä asennoissa hahmon sisälle, mutta asennot olivat ääriasentoja, mihin oikea ihminen ei taipuisi, joten ongelma katsottiin korjaantuneeksi. Samalla huomattiin, että hahmon kulmakarvat painuivat hahmon sisään päätä taivuttaessa eteenpäin. Tällöin kulmakarvat valittiin aktiiviseksi ja testattiin niiden vaikutusalueita. Blender oli automaattisessa painoalueiden laitossa laskenut kulmakarvojen kuuluvan myös kaulan rigiin, joten kaulan vaikutus kulmakarvoihin poistettiin. Lopuksi tarkastettiin, että vain pään luu vaikutti niihin.



Kuva 18. Kolmannessa testauksessa osoittautuneet skinnauksen virheet.

3.4 Hahmon animaatio

Animaatiolla tarkoitetaan prosessia, jossa luodaan animoitavalle objektille jatkuvia liikkeitä sekä illuusiota muotojen muuttumisesta. Tällä illuusiolla tarkoitetaan tekniikkaa, jossa nopeasti esitetystä kuvasarjasta jokaisessa eri kuvassa on vain pieni muutos, mikä etenee liikkeeksi. Animaatioita voidaan myös nauhoittaa digitaaliseen muotoon. 3D-animaatiossa animaattori muokkaa hahmon asentoja siihen sopivalla ohjelmistolla. Tavanomaisin 3D-animaatiokeino on muokata objektin meshiä, mutta myös matemaattisia funktioita voi käyttää, esimerkiksi efektien ja hiusten liikuttamiseen. (Wikipedia 2014.)

Ennen animaatioprosessin aloittamista tulee huomioida, että animaatio antaa mallinnetulle hahmolle elämän (Mullen & Andaur 2010, 167). Animoidessa yksi suurimmista haasteista on saada hahmo ilmaisemaan tunteita uskottavasti. Ilmaistakseen tunteita tulee ensin ymmärtää tunteita. Jotkut ihmiset tunnistavat tunteita toisia paremmin ja hyvän animaattorin tulisi osata sekä tunnistaa että ilmaista tunteet animoitavan hahmon kautta. (Pardew 2007, 1.)

Tavallisesti animaatioelokuva vaatii useamman osa-alueen animoimista, kuten vaatteet sekä ympäristö. Pelihahmolle puolestaan riittää tietyt liikkumistavat ja mahdollinen kontakti pelimaailmaan, esimerkiksi esineen poimimisen maasta. Eri peleissä on kuitenkin niin sanottuja ”cut scene”- tai ”scinematic”-kohtauksia, missä on käytetty animaatioelokuvan tyyliä. Kohtaukset ovat yleensä lyhyitä ja vievät tarinaa eteenpäin näyttävästi, eli ne ovat grafiikan tasoltaan parempia (kuva 19). Näiden elementtien käyttäminen pelissä vaatii peliprojektilta monipuolisempaa osaamista.



Kuva 19. Vasemmalla pelattava hahmo ja oikealla sama hahmo cut scenessa (Final Fantasy X HD Remake 2014).

Lyhyenkin animaation tekeminen voi viedä paljon aikaa. Aloitteleva animoija tyypillisesti aloittaa animaatioprosessin alusta, mikäli lopputulos epäonnistuu. Näin voi tuhllata viikkoja, mikä ei ole sopivaa esimerkiksi ammattilaisprojektissa. Tämän takia sopivan työskentelytavan omaksuminen on tärkeää animoijalle, jotta hän animoi hyvää jälkeä nopeasti ja on kykenevä korjaamaan virheitään. (Vasconcelos 2011, 145.)

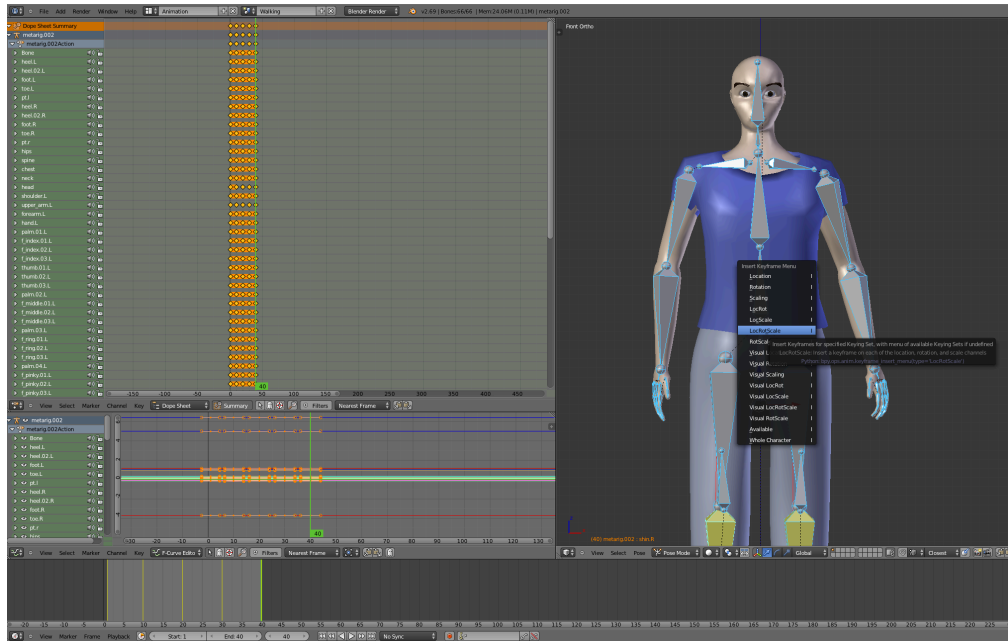
Animaattorin tulisi myös itse nousta työpisteeltään ja suorittaa animoitavia liikkeitä. Vastaavasti animaattori voi pyytää toista ihmistä toimimaan mallina ja tarkastella liikkeitä ja eleitä. Animaattoria voikin kutsua näyttelijäksi, joka esiintyy

animoimansa hahmon kautta. Kaikki hahmon liikkeet ovat viimeistelyjä ja tarkkaan mietittyjä, eikä mitään ole jätetty sattuman varaan. Olisikin toivottavaa, että animaattorit harrastaisivat pienimuotoisissa ryhmissä näyttelemistä tai tutkisivat esimerkiksi kadulla ihmisten liikkeitä ja eleitä. (Pardew 2007, 3.)

Yksi uusimpia animointikeinoja on Motion Capture eli liikkeentunnistus oikean ihmisen liikkeistä. Liikkeentunnistuslaite, esimerkiksi Microsoftin kehittämä Kinect, tunnistaa ihmisen siihen asetetun tiedon perusteella ja erottelee eri vartalonosien muodot. Lopuksi se laskee algoritmin perusteella vastaavat nivelkohdat, joihin rig asetetaan ja 3D-malli alkaa liikkua. (Tong ym. 2012, 347.)

3.4.1 Idle-animaatio

Animointi aloitettiin ottamalla valinnaksi rig ja vaihtamalla tila Pose Modeen. Sen jälkeen valittiin Screen layoutiksi Animation. Blenderin oletusasetukset asettavat animaation framet 1-250 ja animaation alkamaan ensimmäisestä framesta Timelinesssa. Frameihin voi asettaa keyframeja, mihin on tallennettu hahmon asentoja ja toistettaessa Timelinea hahmon asennot vaihtuvat. Keyframe asetettiin valitsemalla koko rig aktiiviseksi ja painamalla näppäimistöä I:tä. Tällöin avautui Insert Keyframe Menu, josta valittiin LocRotScale (kuva 20), mikä tallensi asennon keyframeen.



Kuva 20. Keyframien asettaminen.

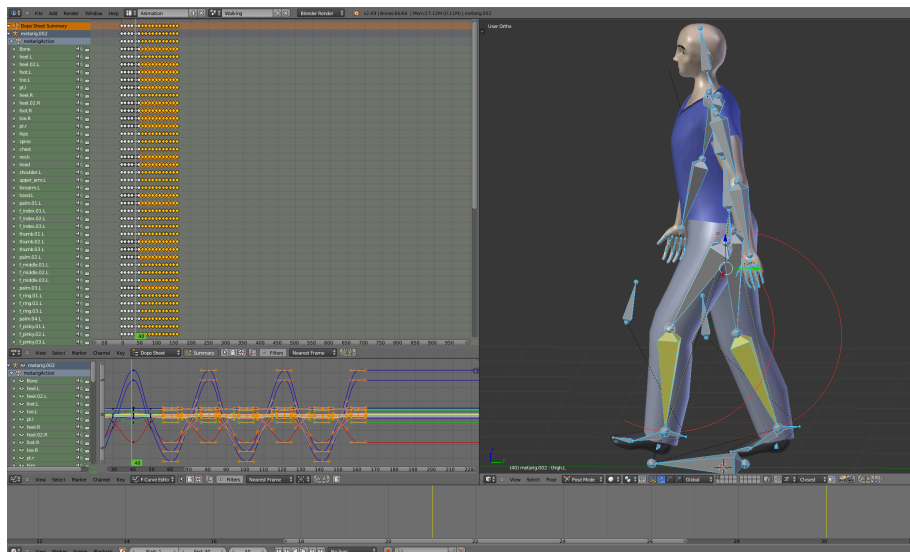
Hahmon perusasento tallennettiin ensimmäiseen keyframeeseen, josta siirryttiin 10 framea eteenpäin. Tässä kohdassa hahmon asentoa muutettiin ja tallennettiin uuteen keyframeeseen. Samoja toimintoja toistettiin niin kauan kunnes hahmon animaatio näytti luonnolliselta.

3.4.2 Kävely-animaatio

Kävelyn animoiminen aloitettiin kävelysyklin ääriasennosta. Kävelyn ääriasennossa hahmon toinen jalka on takana hieman koukussa ja toinen jalka on edessä polvi hieman taitettuna. Myös jalkateriä käännettiin niin, että hahmon varpaat osoittavat ylös. Käsä liikutettiin vastakkaisiin suuntiin eli kun oikea jalka oli edessä, niin myös vasen käsi oli edessä. Sormia liikutettiin lähemmäs toisiaan, jotta ne olisivat luonnollisemman näköisiä sekä hahmoa taitettiin hieman etukumaraan. Hartioita käännettiin liikkumaan käsien mukana. Asento asetettiin keyframeeseen.

Kävelyn toinen ääriasento asetettiin niin, että hahmon asento kopioitiin ja liitettiin 20 keyframea myöhemmäksi pelikuvana. Animaatio tarkistettiin klikkaamalla

”Play animation”-painiketta. Hahmo liikkui muuten hyvin, mutta kävely ei näyttänyt luonnolliselta, koska hahmon yläruumis ei liikkunut kuin sivuttain. Tällöin palattiin 10 keyframea taaksepäin liikkeiden väliin, mihin tallennettiin uusi keyframe, jossa hahmon yläruumis laskeutui hieman. Tällöin kävely näytti normaalilta. Jotta liike olisi jatkuvaa eli toimisi niin sanotusti loopissa, sitä tuli kopioida muutaman kerran ja varmistaa, että animaatio loppui samaan asentoon kuin mistä se alkoi (kuva 21).



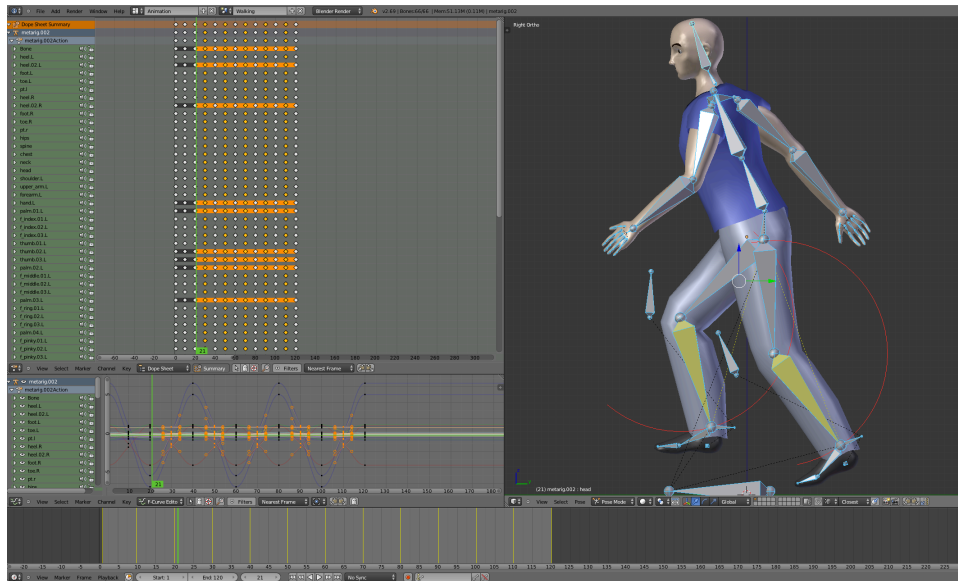
Kuva 21. Kävelyn animaation toiminta loopissa.

3.4.3 Juoksu-animaatio

Juoksu-animaatio aloitettiin liikuttamalla hahmon vasenta jalkaa yläviistoon niin, että polvi taittui. Tämän jälkeen oikeaa jalkaa vietiin taaksepäin ja jalkaterää käännettiin alaspäin. Kädet asetettiin vastakkaisiin suuntiin ja niitä koukistettiin Rotate-komennolla. Hahmoa taivutettiin lopuksi vielä selkärangasta kallistumaan eteenpäin. Asento kopioitiin ja liitettiin vastakkaisena kopiona 20 framea eteenpäin ja tallennettiin. Asentoja kopioitiin ja liitettiin 20:n framen välein niin, että animaation pituus oli 120 framea.

Animaatiota tarkastellessa huomattiin, että hahmon pää pysyi melkein staattisesti paikoillaan ja näytti luonnottomalta. Tämän takia hahmon päätä kallistettiin

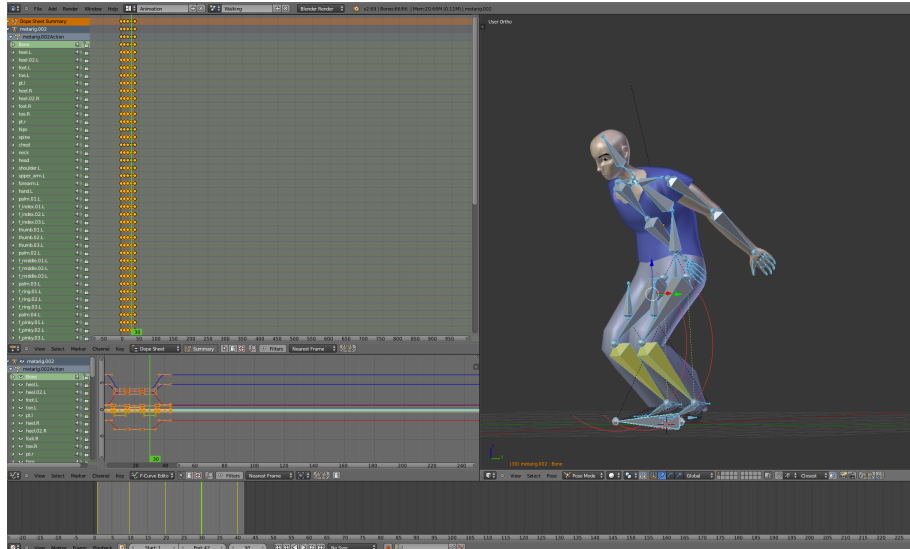
eteenpäin 10 framen välein. Animaatio näytti tällöin luonnollisemmalta. Lopuksi tarkastettiin, että juoksuanimaatio toimi loopissa (kuva 22).



Kuva 22. Hahmon juoksusykli.

3.4.4 Hyppy-animaatio

Hyppäämisen animoiminen erosi muista aiemmin tehdyistä animaatioista niin, että se oli automaattisesti loopissa, koska hyppy alkoi perusasennosta. Hahmon perusasento tallennettiin ensimmäiseen keyframeen ja 10 framea eteenpäin hahmoa laskettiin pitäen kiinni lantioluusta, jolloin jalat taittuivat automaattisesti. Hahmoa taitettiin eteenpäin ylimmästä selkärangan luusta, jolloin hahmo kumartui eteenpäin ja sen kädet osoittivat taaksepäin. Tämä asento toimisi hyppeyn ponnistamisen asentona ja se tallennettiin keyframeen. Nyt hahmo pystyi nostamaan Root Bonesta, jolloin koko hahmo nousi ja sen pystyi asettelemaan suurempaan asentoon. Hypyn ääriasento tallennettiin 20. frameen ja aikaisempi 10:s frame kopioitiin ja liitettiin 30. frameen. Hahmon perusasento kopioitiin viimeiseen, 40. frameen, jolloin hypyn sykli oli valmis. Animaatio tarkastettiin toistamalla sitä (kuva 23).

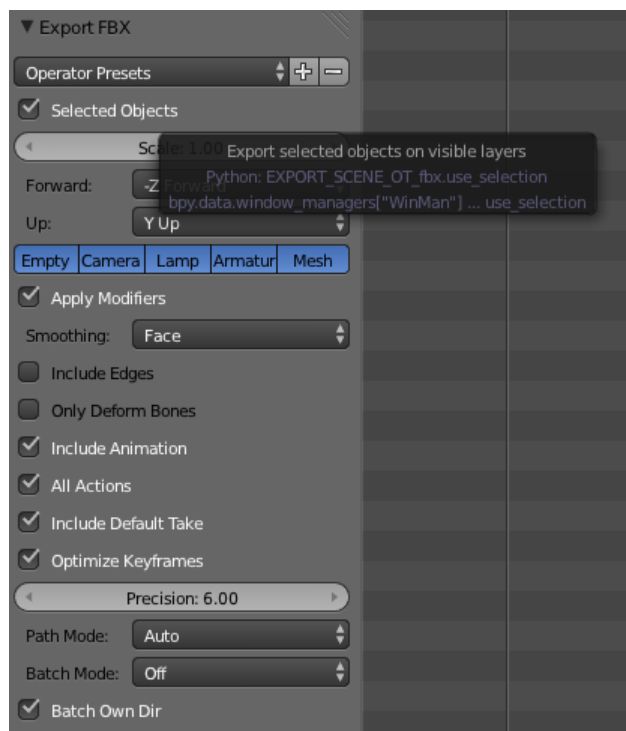


Kuva 23. Hyppy-animaation testaus.

4 TULOSTEN TESTAUS

4.1 Hahmon vieminen Unityyn

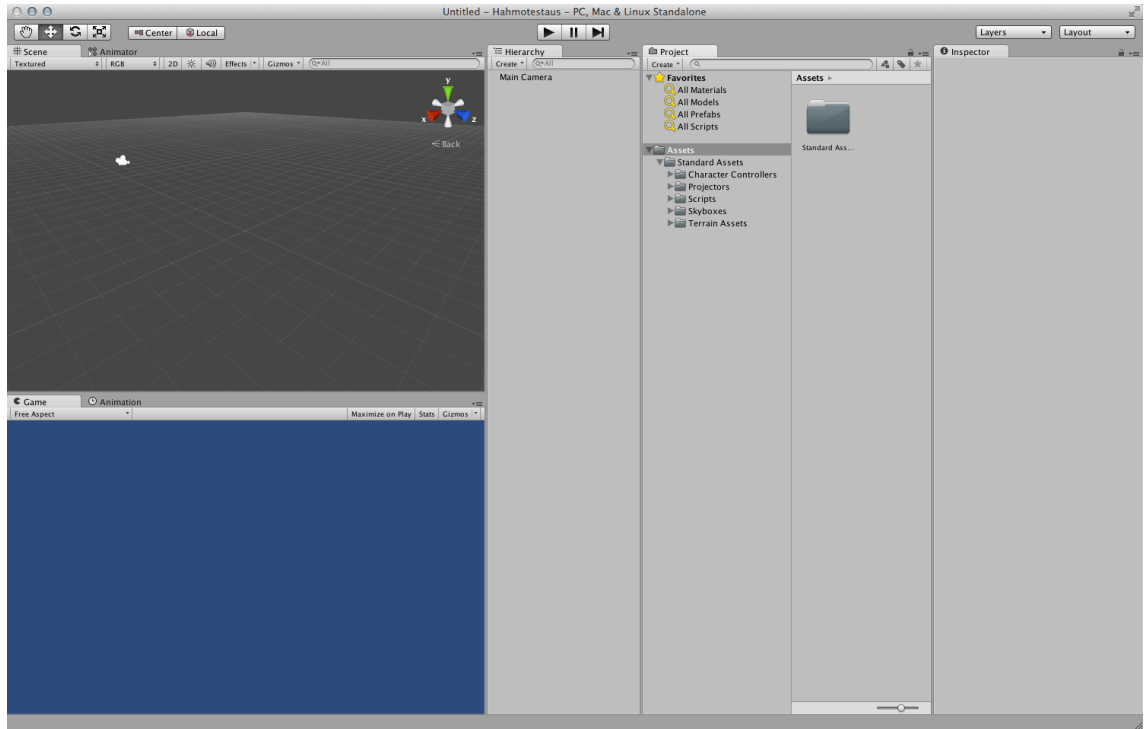
Hahmon pystyi viemään kahdella eri tavalla Unityyn, jotta sen animaatiot ja muut tiedot säilyisivät. Blender-tiedoston pystyi suoraan siirtämään haluamaansa Unity Project –kansioon. Tämä mahdollisti hahmon muokkaamisen samassa tiedostossa Blenderin puolella, jolloin kaikki hahmoon tehdyt muutokset päivitivät automaattisesti Unityyn. Tiedoston pystyi myös muuttamaan Autodesk FBX –muotoon ja viemään tämän tiedoston Unityyn. FBX:n valinnassa tuli kuitenkin tietää tarkasti mitä hahmosta valitaan vietäväksi uuteen tiedostoon (kuva 24). Valintojen avulla Blender liitti tiedostoon kaiken hahmossa olevan datan, eikä jättänyt mitään osa-alueita pois.



Kuva 24. Autodesk FBX –tiedostomuodon viemisasetukset.

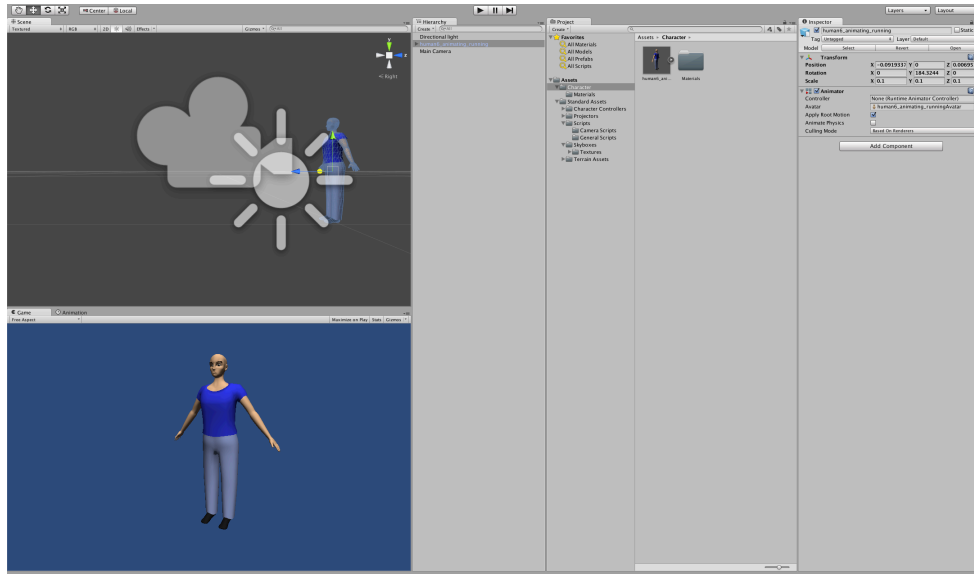
Unityssa tuli luoda uusi projekti, johon pystyi valitsemaan erilaisia Standart Assetseja. Nämä ovat Unityn omia valmiiksi tehtyjä työvälineitä, joilla pystyy no-

peuttamaan työn tekemistä. Vaihtoehtoina oli esimerkiksi valmis taivas, hahmon kontrollit ja kamerat sekä skriptit (kuva 25). Valinnan jälkeen Assets-kansion alle luotiin Character-kansio, johon hahmo tulitaisiin viemään. Omien kansioiden luonti selkeytti projektitiedostoa, eikä hahmo päätynyt väärään kansioon.



Kuva 25. Unityn perusnäkö, johon on viety Standart Assets –paketin osia.

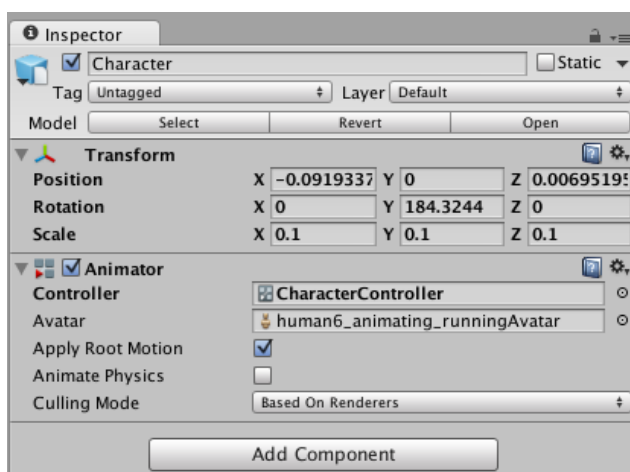
Hahmon, jossa oli juoksuanimaatio, raahattiin ensimmäisenä Unityyn Character-kansioon. Kun Unity oli automaattisesti tuonut hahmon, sen asetuksia pystyi muokkaamaan esimerkiksi siten miten Unity skaalaa hahmon kokoa tai animaation asetuksia. Animation-kohdasta muokattiin animaation alkamis- ja loppumisaikaa, jotta animaatio toimisi varmasti samalla tavalla kuin se oli asetettu Blenderissä toimimaan. Hahmon pystyi raahaamaan Sceneen, eli näkymään ja kamera aseteltiin näyttämään hahmo. Näkymään luotiin myös Directional Light –valo, jotta hahmo näkyisi paremmin. Hahmon perusasetukset olivat kunnossa animaation testaukseen (kuva 26).



Kuva 26. Hahmo vietynä näkömään.

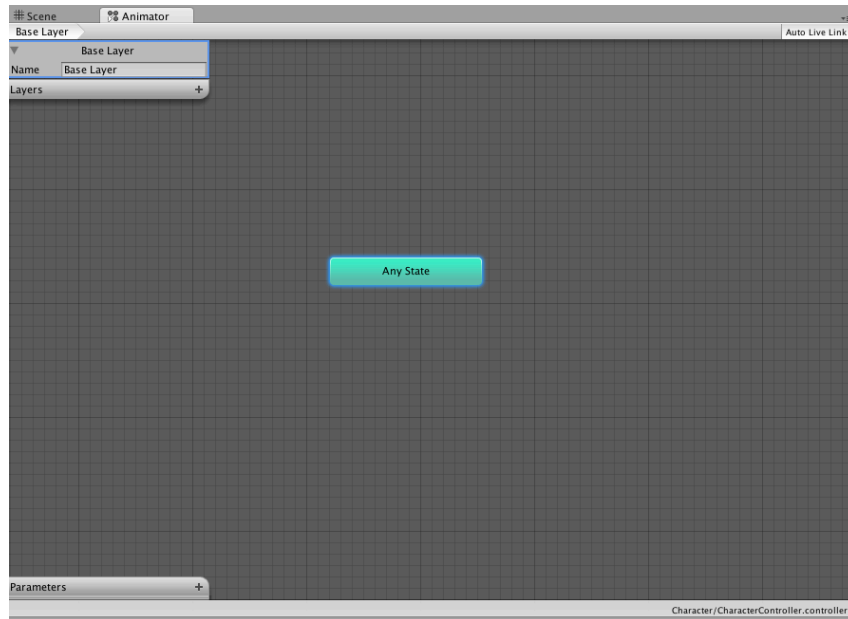
4.2 Hahmon animaatioiden testaus Unityssa

Jotta hahmon animaatiot toimisivat Unityssa, tuli projektiin luoda Assets-valikon alta Animator Controller –objekti. Tälle objektille annettiin nimeksi Character Controller ja se raahattiin Hierarchy-välilehteen ja upotettiin hahmoon. Tällöin hahmon klikatessa aktiiviseksi se näytti Inspector-välilehdessä animaation kontrolloijaksi CharacterControllerin (kuva 27).



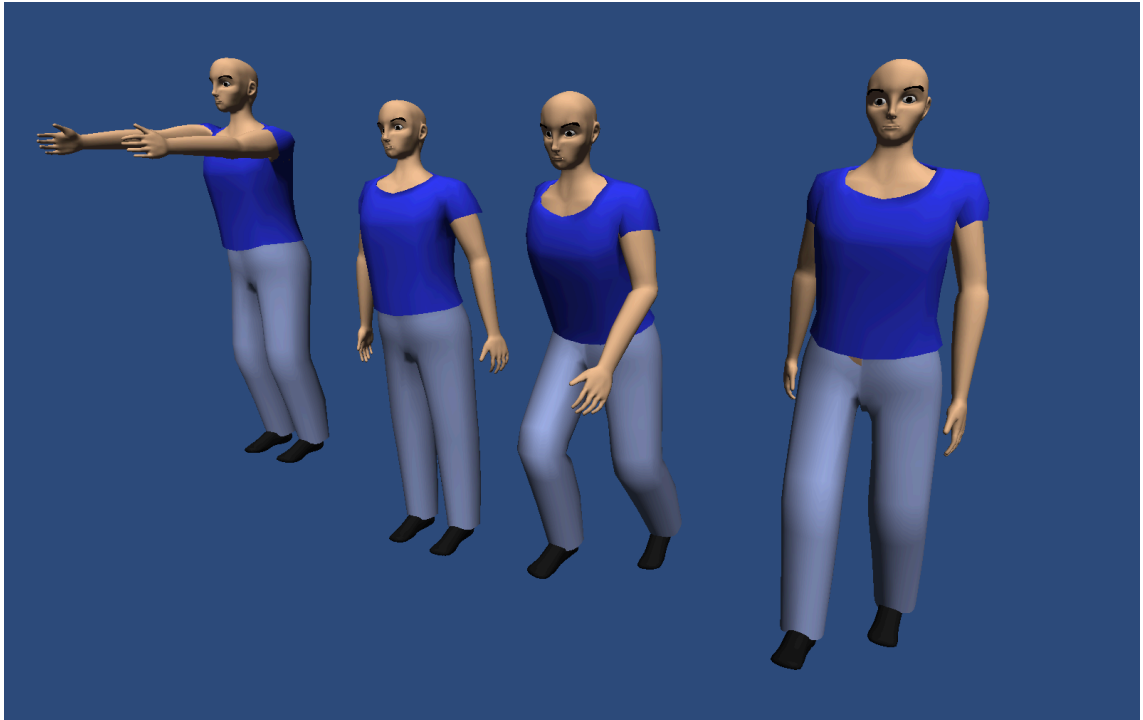
Kuva 27. CharacterController lisätynä animaation kontrolloijaksi.

Animator Controllerin lisäämisen jälkeen sitä kaksoisklikattiin, jotta saatiin Animator-välilehti näkyville Unityn perusnäkykseen. Vaihtaessa Animator-välilehden aktiiviseksi Unity antoi vakionäkymän, joka sisälsi Base Layerin, Layers- ja Parameters-valikot sekä Any State –objektin (kuva 28). Kun Any Stateen liitetään animaatio, se toistaa sen heti kun Unityssa klikataan Play-painiketta.



Kuva 28. Animator-välilehden vakionäkymä.

Animaation pystyi raahaamaan Any Staten alle, jolloin klikatessa Play-painiketta animaatio rupesi toistamaan itseään. Tässä vaiheessa ongelmaksi muodostui se, että animaatiot oli tehty eri tiedostoihin, joten testaaminen vaati jokaiseen eri hahmoon Animator Controllerin lisäämisen. Myös jokaiseen animaatioon tuli laittaa Loop Time –toiminto, jotta hahmo toistaisi liikkeen useamman kerran. Asetusten ollessa valmiita animaatioiden toimivuus tarkastettiin painamalla Play-painiketta. Hahmojen animaatiot toimivat kuten oli tarkoitus (kuva 29) ja joidenkin animaatioiden nopeutta tuli muuttaa, jotta animaatio olisi luonnollisempi. Ainoastaan hyppy-animaatiossa hahmon ihomesh tuli esiin paitameshin alta, vaikka sitä ei Blenderin puolella tapahtunut (kuva 30).



Kuva 29. Animaatioiden testaus Unityssa.



Kuva 30. Testauksessa Unityn puolella oli hyppy-animaatiossa virheitä.

5 POHDINTA

Opinnäytetyön yksi haastavimmista osioista oli projektinhallinta. Vaikka olin aikatauluttanut työn tekemisen niin vaativimmat osat veivät enemmän aikaa kuin odotin ja muutaman kerran prosessissa taaksepäin meneminen takia hidasti työn etenemistä. Alustava kartoitus 3D-mallinnustekniikoista tarjosi pohjan tehokkaaseen mallinnukseen sekä selvitti, että millä tekniikoilla muita elementtejä kannattaa luoda. Omasta mielestäni hahmo 3D-mallina on aloittelijalta onnistunut, mutta sen olisi voinut myös tehdä paremmalla tavalla, esimerkiksi olisi voinut kutistaa vertex-määrää ja mallintaa kasvot uudelleen. Kasvojen tekeminen oli mielestäni vaikeinta mallinnusosiossa ja siihen olisi voinut käyttää hyvin paljon aikaa. Palasinkin muutaman kerran muokkaamaan esimerkiksi hahmon huulia pienemmäksi, vaikka olin siirtynyt jo mallintamaan toisia ruumiinosia.

Riggaamiseen ja skinnaamiseen kului toiseksi eniten aikaa, mutta se oli työssä haastavin osuus. Huomasin vasta skinnaamisen jälkeen hahmon rigissä muutaman virheen, jotka eivät kuitenkaan vaikuttaneet hahmon toimivuuteen. Tämän takia jätin rigin sellaiseksi kuin se oli. Skinnaus piti tehdä muutamaan kertaan uudestaan, koska en ollut omaksunut hyvää työtapaa, jolloin ensin rajasin vertexejä liikaa. Skinnauksen opettelua tulisikin jatkaa ja paneutua tarkemmin siihen, miten laajalle alueelle vertexien on hyvä vaikuttaa, jotta liikkeestä tulisi aidon näköistä. Myös Weight Paintin käyttöä tulisi opetella, koska samanlainen vaikutusalueiden määrittäminen on useassa eri mallinnusohjelmassa.

Animaatioita tehdessä huomasin, että Blenderin käyttöä olisi pitänyt opetella enemmän ja ajan kanssa. Esimerkiksi layerien eli eri tasojen käyttäminen olisi ollut tehokkaampaa työskentelyä kuin tallentaa jokaisesta animaatiosta oma tiedosto. Idle-animaation tekeminen vei aikaa, koska ensin tuli päättää, että mitä hahmo tekisi silloin, kun siihen ei kohdistu mitään toimintoja. Lopulta päädyin siihen, että hahmo tarkastelee ympäristöään ja heiluttaa käsiään kärsimättömästi.

Työn edetessä huomasin sen, että joidenkin prosessien aloittaminen uudelleen tuntui helpommalta kuin että olisin korjannut virheeni. Yritin pitää tämän mahdollisimman vähäisenä, jotta oppisin paremman ja tehokkaamman työtavan, sekä myös välttämään virheitä. Ainoat kohdat, joissa aloitin prosessin alusta, olivat skinnaus ja ensimmäiset animaatiot. Skinnausta en ollut aikaisemmin tehnyt, joten virheiden korjaamiseen olisi luultavasti mennyt enemmän aikaa kuin skinnauksen uudelleen tekemiseen. Animaation tekeminen oli minulle uutta Blenderissä, joten kaikki komennot eivät olleet tuttuja eikä työrutiinia ollut muodostunut. Näin jälkeen päin osaan tehdä parempaa jälkeä nopeasti.

Kokonaisuudessaan opinnäytetyö antoi minulle paljon: se opetti ammattimaisempaa työskentelytapaa sekä tiedon etsimistä. Se myös toi esille heikkouteni tietyillä osa-alueilla, ja sain harjoittaa niiden korjausta turvallisessa ympäristössä, vaikka työllä oli aikaraja ja mahdollisuus epäonnistua. Se mahdollisti paineen sietämisen oppimisen ja liiallisen itsekritiikin poistumisen. Tiesin, että työ ei ole täydellinen, mutta sen ei sitä tarvinnutkaan olla. Tärkeintä oli se, mitä sen edetessä olin oppinut.

Hahmon työstämistä voisi tulevaisuudessa jatkaa viemällä kaikki animaatiot samaan Blender-tiedostoon, jotta Unityyn ei tarvitsisi viedä monta eri tiedostoa. Samoin Unityssa hahmon testausta pystyisi jatkamaan luomalla hahmolle ympäristö ja ohjelmoida sille toiminnallisuus. Tämä edistäisi omaa osaamista ja eri osa-alueiden hallinta toimisi vahvuutena työelämässä.

LÄHTEET

- 2.5D 2014. GiantBomb. Viitattu 15.5.2014 <http://www.giantbomb.com/25d/3015-660/>
- 3D scanner 2014a. Wikipedia. Viitattu 15.5.2014. http://en.wikipedia.org/wiki/3D_scanner
- Animation 2014b. Wikipedia. Viitattu 14.4.2014 <http://en.wikipedia.org/wiki/Animation>
- Cartesian coordinate system 2014. The Free Dictionary. Viitattu 15.5.2014 <http://www.thefreedictionary.com/Cartesian+coordinate+system>
- Final Fantasy X/X-2 HD Remaster. 2014. Square Enix & Virtuos. Square Enix.
- Freeman, M. 2013. What is Digital Sculpting? Viitattu 15.5.2014 <http://vimeo.com/81614746>
- Herreño, J. 2011. Blender 2.5 HOTSHOT. Olton: Packt Publishing Ltd.
- Mullen, T. & Andaur, C. 2010. Blender Studio Projects Digital Movie-Making. Hoboken: Sybex.
- Pardew, L. 2007. Character Emotion in 2D and 3D Animation. Boston: Course Technologies / Cengage Learning.
- Pichot, J. 2013. Cinema 4D R15 review. Viitattu 15.5.2014. <http://www.digitalartsonline.co.uk/reviews/motion-graphics/cinema-4d-r15-review/>
- Rotenberg, S. 2005. Procedural Modeling. Viitattu 15.5.2014 http://graphics.ucsd.edu/courses/sce167_f05/CSE167_18.ppt
- Siltakorpi, N. 2014. Sisällöntuotannon menetelmiä 3D-peliympäristöön. Opinnäytetyö. Metropolia Ammattikorkeakoulu. Viitattu 11.6.2014 <http://www.theseus.fi/handle/10024/74064>
- Silverman, D. 2013. 3D Primer for Game Developers: An Overview of 3D Modeling in Games. Viitattu 15.5.2014 <http://gamedevelopment.tutsplus.com/articles/3d-primer-for-game-developers-an-overview-of-3d-modeling-in-games--gamedev-5704>
- Slick, J. 2014a. Anatomy of a 3D Model. Viitattu 24.3.2014. <http://3d.about.com/od/3d-101-The-Basics/a/Anatomy-Of-A-3d-Model.htm>
- Slick, J. 2014b. 7 Common Modeling Techniques for Film and Games. Viitattu 5.2.2014 <http://3d.about.com/od/3d-101-The-Basics/a/Introduction-To-3d-Modeling-Techniques.htm>
- Slick, J. 2014c. 3D Defined – What is 3D? Viitattu 15.5.2014 <http://3d.about.com/od/3d-101-The-Basics/a/3d-Defined-What-Is-3d.htm>
- Tong, X.; Xu, P. & Yan, X. 2012. Research on skeleton animation motion data based on Kinect. 2012 Fifth International Symposium on Computational Intelligence and Design.
- Totten, C. 2012. Game Character Creation with Blender and Unity. Hoboken: Wiley.
- T-Splines, 2013. Subd to NURBS (Subdivision surface to NURBS conversion). Viitattu 11.6.2014 <http://www.tsplines.com/subdtonurbs>
- Vasconcelos, V. 2011. Blender 2.5 Character Animation Cookbook. Olton: Packt Publishing Ltd.
- Wang, L. & Zhang, J. 2008. A Fast and Efficiency Character Model Method for 3D Game. 2008 International Conference on Internet Computing in Science and Engineering.

What is a NURBS? 2006. RealWorld Graphics. Viitattu 15.5.2014 <http://www.rw-designer.com/NURBS>

Referenssikuva

Työssä käytetty referenssikuva, jonka tarkoitus oli tuoda mittasuhteita ja anatomiaa paremmin esille, jotta mallintaminen helpottuisi.

