

3D-HAHMON TUOTANTO PELIIN – MALLINTAMISESTA PELIMOOTTORIIN

Saloheimo Essi

Opinnäytetyö

Kuvataiteilijakoulutus
Kuvataiteilija (AMK)

2025

Kuvataiteilijakoulutus
Kuvataiteilija (AMK)

Tekijä	Essi Saloheimo	Vuosi	2025
Ohjaaja(t)	Eija Rajalin		
Toimeksiantaja			
Työn nimi	3D-hahmon tuotanto peliin mallintamisesta pelimoottoriin		
Sivumäärä	46+ 1		

Opinnäytetyön tavoitteena oli tutkia työnkulkua peliin räätälöidyn 3D-hahmon luomiseksi. Loin 3D-hahmon taiteilija Alexandre Diboisen konseptin pohjalta. Pää-tutkimuskysymykseni olivat, millainen on 3D-hahmon tuotantoprosessi. Apututkimuskysymyksiä oli mitä työvaiheita ja työkaluja tarvitsen prosessissa ja mitä ominaisuuksia ja teknisiä vaatimuksia tulee löytyä videopeleihin optimoiduista hahmoista.

Pelihakmon luominen jakautui eri työvaiheisiin kuten hahmon sculptaamiseen eli mallintamiseen, retopologyn tekemiseen, uv-mappaukseen, teksturointiin, riggaukseen ja hahmon käyttöönottoon pelimoottorissa. Keskeiset käsitteet liittyvät 3D-mallinnus sanastoon. Työn tietoperusta koostui verkkolähteistä kuten youtube-tutoriaaleista ja asiantuntija artikkeleista. Toteutin hahmon ammattilaisten käyttämällä ohjelmilla Zbrush, Maya ja Substance Painter ja pelimoottorina on Unreal Engine.

Opinnäytetyön tuloksena syntyi 3D-hahmo, joka on optimoitu peleihin. Pelihakmoa voi hyödyntää pelimoottorissa ja sitä voisi teoriassa käyttää oikeassa pelissä. Voin hyödyntää sen tekemiseen vaadittavia taitoja ja opettelemiani 3D-ohjelmia tulevaisuudessa oikeassa produktiossa ja työskentelemään pelifirmassa hahmomallintajan roolissa.

Avainsanat 3D-mallintaminen, retopology, uv-mappaus, teksturointi, riggaus, pelihakmo, pelimoottori

Degree program in Visual Arts
Bachelor of Culture and Arts

Author	Essi Saloheimo	Year	2025
Supervisor(s)	Eija Rajalin		
Commissioned by			
Title	3D-character production for games – from modeling to the game engine		
Number of pages	46 + 1		

The aim of my thesis was to study the workflow steps to create a game ready 3D-character. I created the 3D-character based on the character concept of the artist, Alexandre Diboine. My main research question is what's the 3D-character production process like. Auxiliary research questions are what workflow steps and tools will be needed in the process and what features and technical requirements are necessary in a character made for games.

The creation of a game character was made up of different workflow steps; 3D-modeling, sculpting, retopology, uv-mapping, texturing, rigging and making the character functional in the game engine. The key concepts were related to 3D-modeling glossary. The information base consisted of online sources like YouTube tutorials and expert articles. I created the character with programs used by professionals; Zbrush, Maya, Substance Painter and Unreal Engine game engine.

The result of the thesis is a 3D-character, that's been optimized for games. The game character can be used in the game engine, and it could be used in a real game. I can use the skills required to make it and the 3D-programs I learned in the future in a real game production and work in a game company as a character modeler.

Keywords 3D-modeling, retopology, uv-mapping, texturing, rigging, game character, game engine

SISÄLLYS

1 JOHDANTO	6
2 MALLINNETTAVAN HAHMOKONSEPTIN VALINTA.....	8
3 3D-MALLINTAMINEN ZBRUSHILLA: MITTASUHTEET JA BLOCKOUT.....	9
3.1 Hahmon kasvojen ja hiusten 3D-mallintaminen	12
3.2 Vaatteiden ja asusteiden mallintaminen.....	14
4 RETOPOLOGY MAYALLA	16
5 UV-MAPPAUS MAYALLA	20
6 TEKSTUROIINTI ADOBE SUBSTANCE 3D PAINTER-OHJELMALLA.....	23
6.1 Baking mesh maps	23
6.2 Teksturointi	24
7 HAHMON RIGGAUS BLENDERISSÄ	27
8 HAHMO PELIMOOTTORI UNREAL ENGINESSÄ.....	30
9 POHDINTA.....	39
LÄHTEET.....	41
LIITTEET	46

ALKUSANAT

Haluan kiittää taiteilijaa Alexander Zedig Diboinea, joka antoi minulle luvan mallintaa hänen hahmokonseptinsa 3D-muotoon.

1 JOHDANTO

3D-mallintaminen on prosessi, jossa tietokoneella 3D-mallinnusohjelmalla luodaan virtuaalinen kolmiulotteinen objekti tai pinta. 3D-objektit muodostuvat pisteistä, viivoista ja polygonipinnoista. Mallinnusprosessissa objektista saa määritettyä sen koon, muodon ja tekstuurin. (FutureLearn 2022.) 3D-mallinnusta hyödynnetään monilla eri aloilla, kuten elokuvissa, tv:ssä VFX- ja animaatiostudioissa ja pelinkehityksessä. Myös arkkitehdit ja insinöörit hyödyntävät 3D-mallinnusta, ja sitä käytetään myös tuotantoteollisuudessa. (Autodesk 2023.)

Seuratessani eri ammattitaiteilijoita olen huomannut, että heidän kohdallaan käytetyimpiä 3D-mallinnusohjelmia ovat TV-, elokuva-, VFX- ja pelialalla Maya ja Zbrush, kompositiossa Hudini ja Nuke sekä teksturoinnissa Mari ja Substance 3D Painter, mutta tämä vaihtelee alasta ja työpaikasta riippuen. Havaintoni mukaan 3D-mallintamisen työpaikkailmoituksissa peli- ja animaatiosektorilla vaaditaan useimmiten Mayan, Zbrushin tai 3ds Maxin osaamista. Opinnäytetyön projektissa mallintamiseen käytän Zbrushia ja Mayaa sekä teksturointiin Substance 3D painteriä. Pelimoottorina käytän Unreal Engineä.

3D-mallintaminen peleihin jakautuu karkeasti kolmeen osa-alueeseen objekteihin ja proppeihin, hahmoihin ja ympäristön elementteihin (Rocketbrush studio 2024). Molemmissa täytyy ottaa huomioon eri asioita, ja 3D-malleilla on eri teknisiä vaatimuksia. Oma opinnäytetyöni käsittelee pelitarkoitukseen tulevan hahmon 3D-tuotantoprosessia. Päättökysymys on millainen on 3D-pelihahmon tuotantoprosessi. Apututkimuskysymyksiä ovat seuraavat: Mitä työvaiheita ja työkaluja tulen tarvitsemaan prosessiin? Mitä ominaisuuksia ja teknisiä vaatimuksia on 3D-hahmolta, joka on tehty optimoidusti videopelikäyttöön?

Mallinnan 3D-hahmon valmiin konseptin pohjalta, koska 3D-mallintajana työni on mallintaa hahmo 3D-muotoon, eikä työni ole hahmojen suunnittelu. Keskeisimmät työvaiheet 3D-hahmon luomiseen ovat high-poly-version mallintaminen, re-topology eli yksinkertaisemman peliin optimoidun version tekeminen, UV-mappaus, eri mappien beikkaus ja teksturointi. Riggaan hahmon Unreal Enginen luurankoon ja testaan sitä pelimoottorin animaatiolla.

Opinnäytetyöni on toiminnallinen opinnäytetyö. Se soveltuu omaan tutkimuskysymykseeni, koska sen tavoitteena on tutkia, miten pelihahmo mallinnetaan, joten minun täytyy mallintaa se ja kirjoittaa tämä prosessi auki. Toiminnallisen opinnäytetyön muoto jakautuu toiminnalliseen osuuteen ja raporttiin. Toiminnallisessa osuudessa tehdään ammatillisen taidon, tiedon ja tutkivan tekemisen näyte eli syntyy jokin tuotos, jossa opiskelija näyttää ammatillisen taitonsa liittyen valitsemaansa aiheeseen. Raportissa tutkiva tekeminen sanallistetaan. (Vilkkä & Airaksinen 2003, 9, 12.)

Katson tutoriaaleja, etsin lähteitä peleihin optimoiduista 3D-hahmoista ja teen oman hahmoni niiden perusteella ja niiden vaatimuksiin. Kirjoitan toiminnallista prosessia auki samalla kun teen sitä, sillä näin minulla on tuoreessa muistissa, mitä tein, ja lähteet, joita käytin. Otan välivaihekuvia ja kirjoitan tietoperustan vaatimuksista, joita pelien 3D-hahmoilla on. Voi olla hankalaa löytää tarkkoja studiokohtaisia lähteitä, miten pelifirmoissa tehdään hahmot, mutta yritän löytää luotettavia lähteitä.

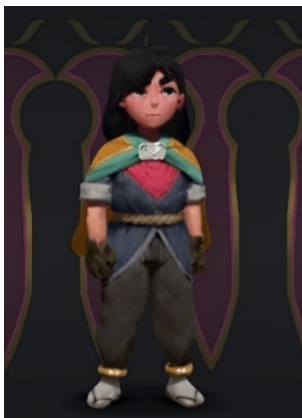
2 MALLINNETTAVAN HAHMOKONSEPTIN VALINTA

Valitsin Alexandre Diboinen hahmon tehtäväksi 3D-muotoon. Valitsin sen, koska hahmo on tarpeeksi yksinkertainen, eikä siinä ole paljon monimutkaista anatomiaa, joten se on nopeampi mallintaa. Siinä ei ole liikaa irrallisia asusteita, jotka voisivat hankaloittaa hahmon riggausta. Hahmo on tyylitelty ja erilainen kehonmuodoltaan aikaisemmin mallintamistani hahmoista, joten se tuo vaihtelevuutta portfoliooni (kuvio 1).



Kuvio 1. Opinnäytetyön 3D-mallintamisessa hyödynnettävä Alexandre Diboinen nimetön hahmo (Diboine 2014a)

Pidän hahmon värimaailmasta ja renderoitu maalaus siitä antaa osviittaa teksturointiin asusteiden materiaaleista (kuvio 2). Yksinkertaisempi hahmo on parempi peleihin, koska sen pyörittämiseen menee vähemmän resursseja ja hahmo toimii pelissä paremmin. Minulle jää myös enemmän aikaa teksturointiin ja hahmon jatkotyöstöön, jos mallintaminen etenee nopeammin.

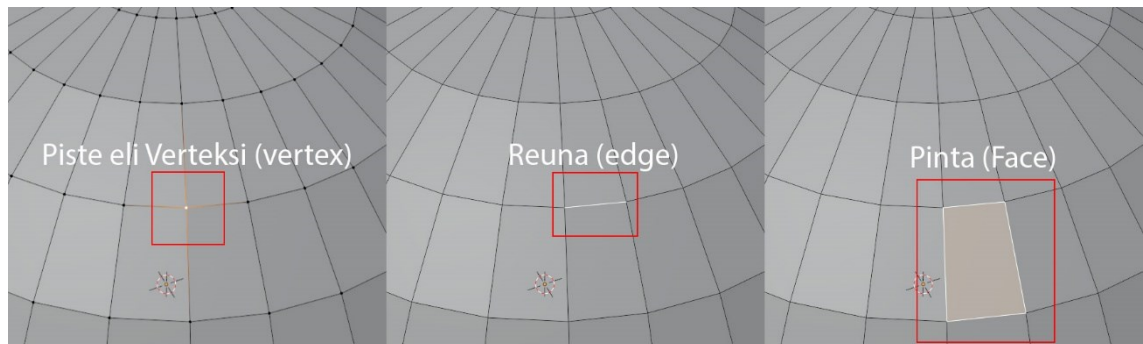


Kuvio 2. Alexandre Diboinen nimettömän hahmon värimaailmaa ja asusteiden materiaaleja (Diboine 2024b)

3 3D-MALLINTAMINEN ZBRUSHILLA: MITTASUHTEET JA BLOCKOUT

3D-mallinnus on tietokonegrafiikkaprosessi, jossa 3D-ohjelmistolla luodaan kolmiulotteinen objekti. Luotuja 3D-objekteja kutsutaan 3D-malleiksi, ja niitä hyödynnetään monella eri alalla kuten tuotekehityksessä, elokuvissa, arkkitehtuurissa ja pelien kehittämisessä. (Autodesk 2023.)

3D-mallit rakentuvat kolmesta perusrakenteesta: pisteistä, reunoista ja pinnoista (kuvio 3). Pinnat, eli polygonit, ovat monikulmioita, jotka koostuvat reunoista. Reuna yhdistää kaksi pistettä suoralla viivalla. Piste on yksittäinen piste tai sijainti 3D-avaruudessa. (Blender 2024.) 3D-malli koostuu rakenteellisesti polygonien muodostamasta verkosta, jota kutsutaan pisteverkoksi tai meshiksi. 3D-polygonimeshit voivat koostua kolmioista (triangle mesh), neliöistä (quad mesh) tai n-määrän monikulmioista (n-gon mesh). Nämä polygonimallit voivat olla high poly, eli koostuvat korkeasta määrästä polygoneja, tai low poly, eli ne on luotu vähemmällä määrällä polygoneja. Peleissä käytetään low poly -malleja, koska siellä on tärkeää, että 3D-objektien renderöinti on nopeaa. (Awati 2024.)



Kuvio 3. 3D-objektin rakenne, englanniksi suluissa

3D-mallintaminen tapahtuu valitun ohjelman sisällä. Polygonimallintaminen edellyttää pisteiden, reunojen ja pintojen manipulointia 3D-meshin rakentamiseksi. Sculptaus on toinen tapa mallintaa, ja siinä digitaalista savea muotoillaan tarkasti erilaisilla siveltimillä ja kuvanveistomenetelmillä. (Adobe 2024.) Itse käytin sculptausta mallintamiseen.

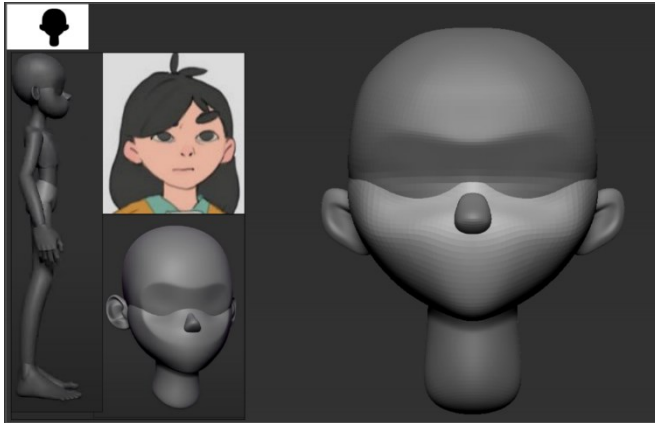
Ennen 3D-mallintamisen aloittamista on hyvä tietää mallinnettavan hahmokonseptin oikeat mittasuhteet. Vein molemmat kuvat Photoshop-ohjelmaan ja

hahmottelin mittasuhteet kuvien päälle. Pään mitta menee vähän vajaa neljä kertaa vartalon mittaan, ja hahmojen pituudella on konseptitaiteiden välillä vähän pituuseroa (kuvio 4).



Kuvio 4. Mittasuhteet mitattu: pään suhde vartaloon

Mallinnusprosessi alkaa blockout-vaiheella. Tässä hahmon perusmuodot hahmotellaan yksinkertaisilla 3D-muodoilla, esimerkiksi kuutioilla tai palloilla. Tämän jälkeen sculptia tehdään asteittain yksityiskohtaisemmaksi ja osia yhdistellään, kunnes muodot on hiottu valmiiksi. Blockoutia pidetään usein tärkeimpänä työvaiheista mallintaessa, koska se on perusta, jonka päälle seuraavat vaiheet rakentuvat. Tässä vaiheessa voi helpommin radikaalisti muuttaa hahmon mittasuhteita, joka on vaikeampaa ja tuhoisampaa mallinnuksen myöhemmissä vaiheissa. (Follygon 2020.) Katsoin Follygonin (2023a) tutorialia sculptaamisesta pohjustukseksi omalle blockout-vaiheelle. Blokkasin pään perusmuodot palloilla, eli osat ovat vielä toistaiseksi erillisiä (kuvio 5).



Kuvio 5. Oikealla pään osat blokattuna palloista vielä erillisinä objekteina

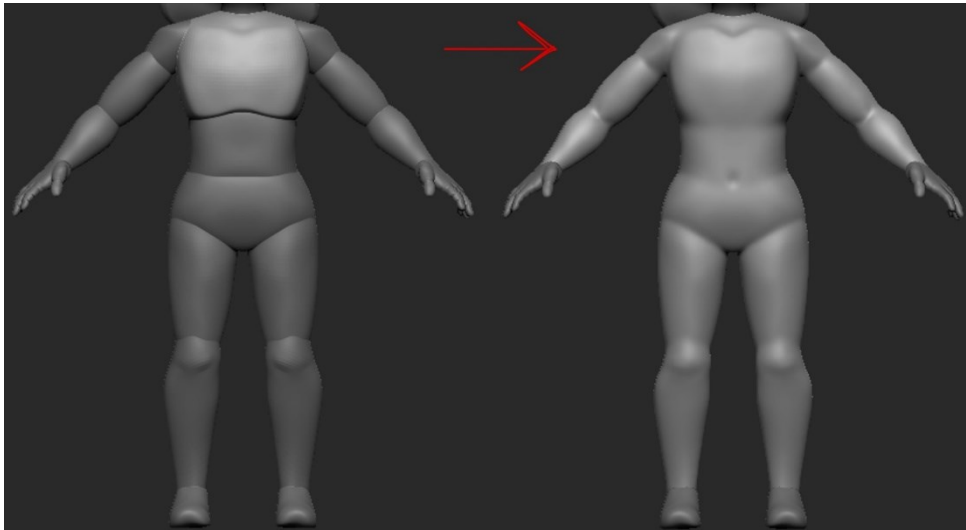
Pelihakmojen vartalot mallinnetaan yleisemmin A- tai T-asentoon, eikä sitä mallinneta suoraan konseptitaiteessa olevaan asentoon. A-asennon tarkoituksena on saada hahmo taipumaan ja deformatumaan eli muotoutumaan luonnollisesti animaatioissa. Se vähentää UV:den venytymistä, koska käsivarret ovat lähempänä lepoasentoa. (AskGameDev 2022.) Itse mallinnan hahmon A-asentoon, jossa kädet taipuvat vähän kyynärtaipeesta eteenpäin ja polvet ovat hieman koukussa (kuvio 6).



Kuvio 6. Esimerkki A-asennosta (AskGameDev 2022)

Blokkasin vartalon muodot palloilla, kuten tein pään kanssa seuraten Follygonin toista tutorialia (Follygon 2023b). Yhdistin vartalon erilliset objektit yhteen käyttämällä merge down- ja dynamesh-toimintoja Zbrushissa (kuvio 7). Työtä nopeutakseni määritin pikinäppäimet useimmiten käyttämilleni siveltimille ja tein

kustomoidun pop-up paletin menujen sisällöille (Michael Pavlovich 2021; Maxon Zbrush 2016).

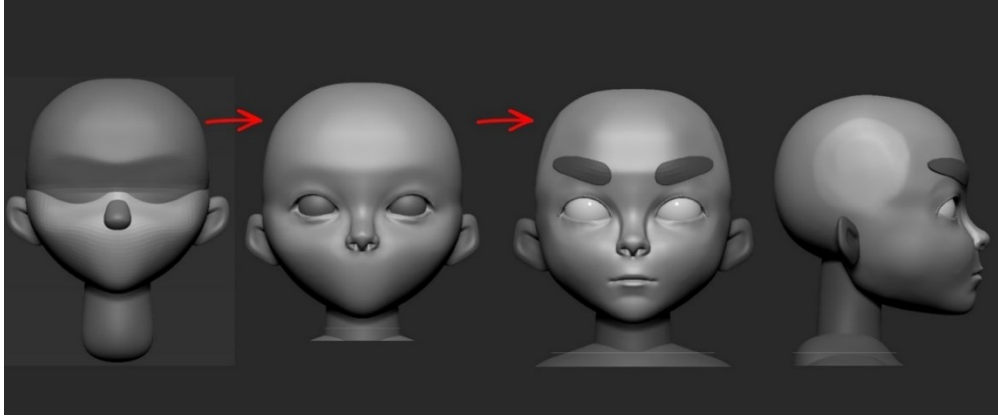


Kuvio 7. Vasemmalla vartalon blockout ja oikealla valmis yhtenäinen vartalo

3.1 Hahmon kasvojen ja hiusten 3D-mallintaminen

Mallintamisen aikana keräsin visuaalista kirjastoa mallikuvista PureRef sovellukseen. Käytin tyylieltyjä 3D-hahmoja ja realistisia valokuvia ihmisistä mallikuvina. Valitsemani hahmokonseptin kasvopiirteet oli aika pelkistetyt mutta tulkitsin, että hahmo oli aasialainen, koska hänellä näyttää olevan monolids eli aasialaistyyppiset silmät. Käytin taiteilija Zach Shartsin artikkelia lähteenä omalle mallinnukselle Zbrushissa (Sergeev & Sharts 2020).

Katsoin Follygonin (2022) tutorialia määrittääkseni kasvoihin "plane changes" eli kohdat, jotka erottavat kasvojen eri puolet toisistaan esimerkiksi etupuolen sivupuolesta. Käytin tähän pinch eli nipistys sivellintä. Mallinsin kulmakarvat ja ripset nähdäkseni, miten hyvin sculpti näyttää hahmokonseptilta. Kasvojen mallintamisen päävaiheet näkyvät kuviosta 8.



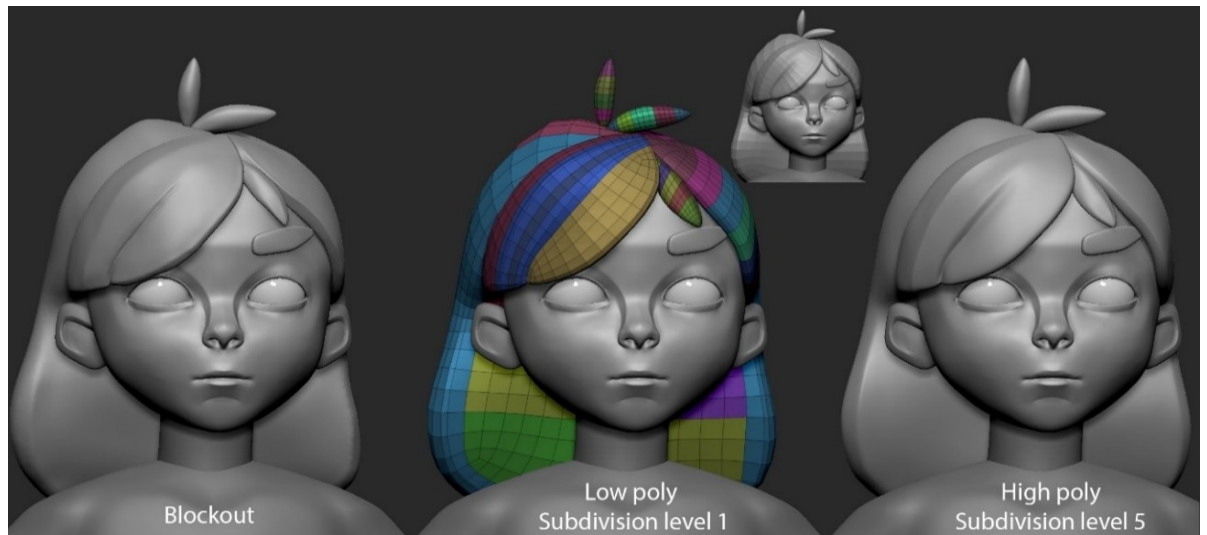
Kuvio 8. Hahmon kasvojen mallinnusprosessin tärkeimmät vaiheet

Hiukset blokkasin seuraten Dan Ederin (2020) artikkelia tyylyttelyjen hiusten mallintamisesta Zbrushissa. Ensin erottelin konseptitaide kuvasta eri väreillä erilliset hiusmassat, jotka mallinnan erillisiksi objekteiksi (kuvio 9). Tämä auttaa minua hahmottamaan miten hahmon hiukset rakentuvat, etenkin kun konseptitaiteen tyyli on suhteellisen pelkistetty hiusten muotojen kanssa ja minun täytyy tietää, miten toteutan ne oikein 3D-muotoon.



Kuvio 9. Eri hiusten osat eriteltyinä

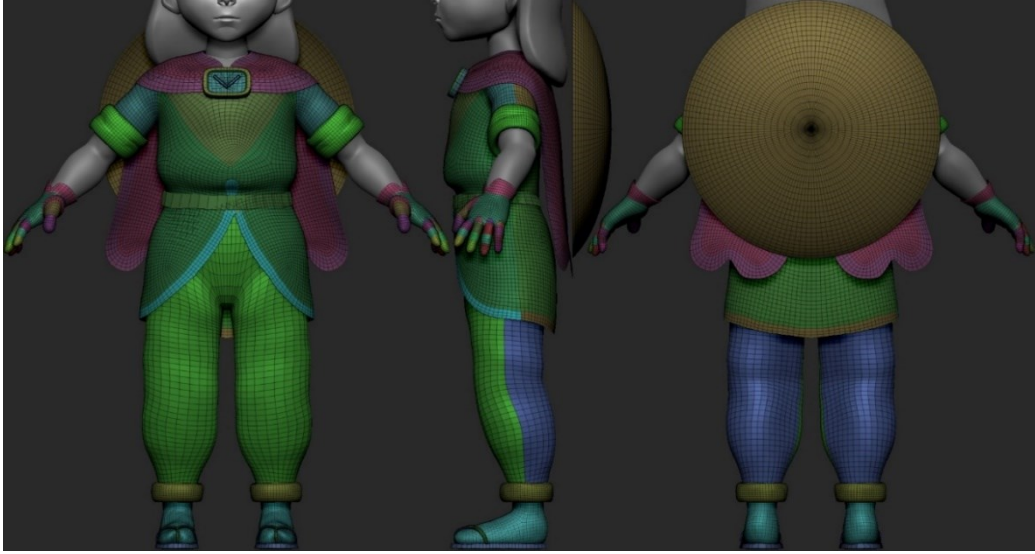
Blockout-vaiheessa mallinsin hiusten eri osat palloista, käytin dynamesh työkalua ja eri siveltimiä manipuloimaan ne oikeisiin muotoihin. Tämän jälkeen mallinsin low poly hiukset blokatuista hiuksista käyttäen eri työkaluja Zbrushissa, kuten maskausta, määrittämällä polygroupit eri osille ja Zremeshaamalla hiukset. High poly hiusten tekoon lisäsin subdivision leveleitä, joka asteittain nelinkertaistavat hiusten osien polygoni määrän, joilla sain terävät yksityiskohdat hiuksiin. Tärkeimpien työvaiheiden lopputulokset on kuvattu visuaalisesti kuviossa 10.



Kuvio 10. Hiusten kehitys blockoutista niiden lopulliseen muotoon

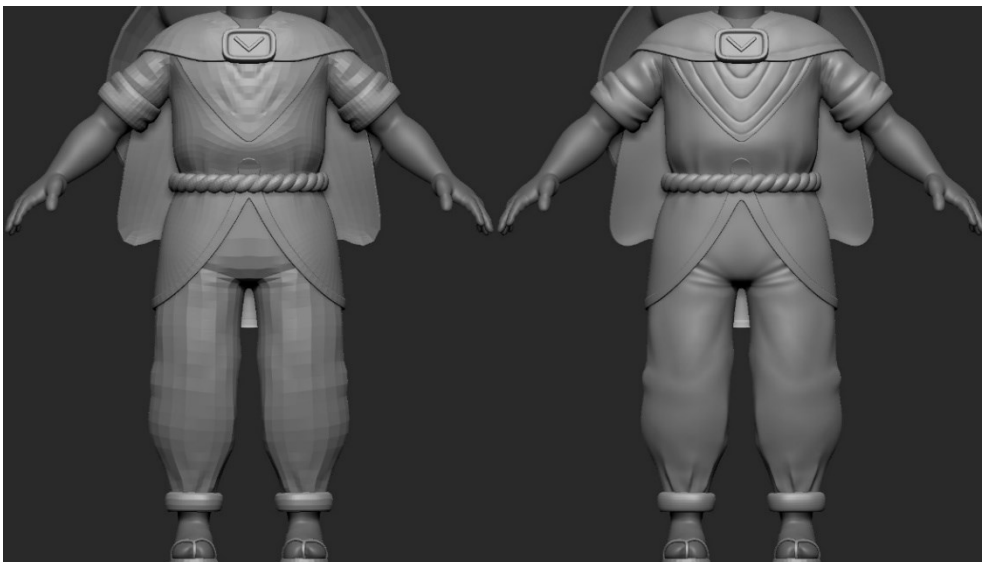
3.2 Vaatteiden ja asusteiden mallintaminen

Marvelous Designer on videopelialan standardiohjelma vaatteiden mallintamiseen ja simuloimiseen. Yleensä vaatteet blokataan Marvelous designerissä ja ne sculptataan loppuun Zbrushissa. Vaatteisiin laitetaan pelimoottorissa “cloth dynamics” tai kangassimulointuminen, jotta se liikkuu hahmon animaation mukana (Tokarev & Karimfazil 2019). Päätin oman hahmoni kanssa, että en halua sen vaatteisiin kangassimulointia muuten kuin viitan. Mallinsin vaatteet pelkästään Zbrushissa. Katsoin Outgang youtube-kanavan videota vaatteiden sketchauksesta Zbrushissa. Videossa painotettiin miten ennen high poly sculptin tekemistä kannattaa tehdä sketch eli hahmotelma vaatteista. Näin saa korjattua mahdolliset ongelmat mittasuhteisiin, rakenteisiin ja volyymeihin liittyen ja tietää miten vaatteet tulisi rakentua (Outgang 2020). Vaatteiden blokkaamiseen katsoin Reallusionin tutoriaalia aiheesta (Bretz 2021). Vaatteet blokkasin Zbrushissa maskaamalla vaatteiden muodot hahmon vartalosta ja sitten jatkotyöstämällä ne loppuun omina objekteinaan. Jotkin osat, kuten kilven ja nilkkarenkaat mallinsin muokkaamalla ne perusmuodoista (puolikas pallo ja donitsi), jotka lisäsin valikosta. Blokattut vaatteet ovat vielä tässä vaiheessa ilman paksuutta ja yksityiskohtia (kuvio 11).



Kuvio 11. Blockout-vaiheen vaatteet

Viimeistelin vaatteet sculptaamalla niihin yksityiskohdat käyttäen eri siveltimiä. Laitoin niihin paksuuden koska niin oli tehty Maxon Zbrush-youtube kanavan videossa Horizon Zeron Dawn pelin hahmotaitteesta (Maxon Zbrush 2017). Yksityiskohtien mallintamiseen seurasin Gaarajapanimen (2022) ja Mangosen (2023) youtube kanavien videoita. Heillä oli useita mallintamistekniikoita, joita hyödynsin omassa työssäni. Lisäämällä low poly vaatteisiin subdivision leveitä, sain sculpattua high poly version vaatteisiin tarkat yksityiskohdat (kuvio 12).

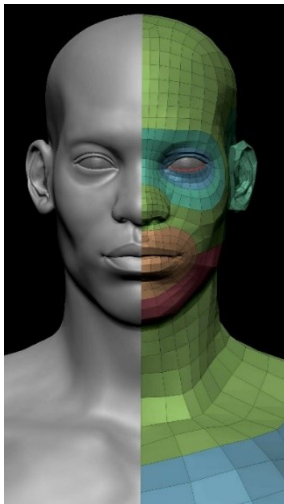


Kuvio 12. Vasemmalla low poly vaatteet, oikealla high poly vaatteet, jossa on korkein subdivision level

4 RETOPOLOGY MAYALLA

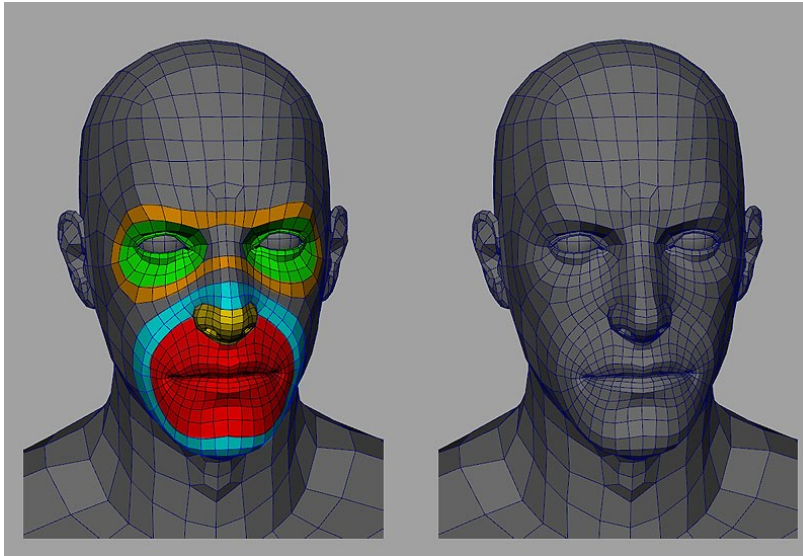
Topologia viittaa pintojen eli polygonien järjestäytymiseen 3D-mallissa. Topologi-alla on väliä, koska se takaa, että hahmo ei ole vain kaunis vaan se on funk-
tionaalinen ja se toimii animaatiossa. Se saa hahmon deformatumaan oikein ani-
maatiossa, vähentää renderöintiäikää ja optimoi resurssien käyttöä. Deformatu-
minen viittaa siihen, miten malli muuttuu muotoaan animaatiossa vasteena koh-
distetuille voimille, kuten käden taipuessa. Topologia, joka seuraa objektin muo-
don luonnollista virtausta, muotoutuu ennakoivammin ja realistisemmin. (Gara-
gefarm.net 2023.) Sculptaaminen saa aikaan yksityiskohtaisen mallin lopputulok-
sena, mutta se myös tuottaa paljon polygoneja, että tarvittavat muodot saa tuo-
tettua. Liika yksityiskohtaisuus hidastaa tietokoneita. (Petty 2024.)

Retopology on tärkeä työvaihe hahmon optimointia peleihin. Retopologyssä uusi
pintarakenne tehdään korkearesoluutioisen eli high polygon mallin päälle niin,
että hahmon pinta koostuu neliöistä (Merheb 2023) (kuvio 13). Tämä luo 3D-mal-
lin, jolla voi animoida ja jossa on optimaalinen määrä polygoneja. Retopologyn
tekeminen on siis mallin polygonien vähentämistä. (Room8studio 2020.) Ilman
retopology työvaihetta, hahmoa ei voisi hyödyntää produktiossa, eikä sitä voisi
rigata animoitavaksi.



Kuvio 13. Esimerkki retopologysta. Vasemmalla high poly sculpti, oikealla päälle tehty retopology (Knol 2017)

Pelihahmoilla on tiettyjä rajoitteita, joita retopologyn täytyy noudattaa. Polycount eli polygonien määrä on rajattu, mallista täytyy löytyä oikeanlainen edgeflow eli pintojen kulkusuunta. Esimerkiksi kasvoista täytyy löytyä tiettyjä looppeja pintoja, jotka auttavat hahmon animoitumisessa (edge loops) (kuvio 14).



Kuvio 14. Värillä merkitty tärkeät edge loopit kasvoissa (Parker 2011)

Decimeittasin high poly mallin Decimation Masterilla Zbrushissa ennen Mayaan tuontia. Tämä pienensi polygoni määrän ja tiedostokoon säilyttäen yksityiskohdat (Totenn 2024). Käytin quad tools työkalua Mayassa retopologyn tekemiseen. Katsoin FlippedNormals (2018) youtube-kanavan videon oppiakseni sen käytöstä ja miten käytännössä retopologyn tekeminen toimii Mayassa. Pelihahmojen retopologyssä tulee ottaa huomioon, että hahmossa tulee olla rajallinen määrä polygoneja, joten riippuen projektista pelihahmon vaatteet voi olla yhdistettynä hahmon vartaloon. Tämä säästää polygonien määrässä, kun piilossa vaatteiden alla olevia pintoja ei ole ja tämä myös estää hahmon clippausta, jossa osa hahmon geometriaa, jota pelaajan ei pitäisi nähdä tulee hahmon läpi. Tämä myös helpottaa animoimista, koska vaatteet liikkuvat hahmon liikkeen kanssa, eikä niitä tarvitse rigata erikseen. Tällaista työkulkua hyödynnetään mielestäni Overwatch-pelin hahmoissa. Yritän ottaa tämän pelin hahmojen topologiasta mallia oman retopologyn tekemiseen (kuviot 15).



Kuvio 15. Vasemmalla valmis Overwatch hahmo tekstuureilla. Oikealla saman hahmon topologia (Harisova 2020)

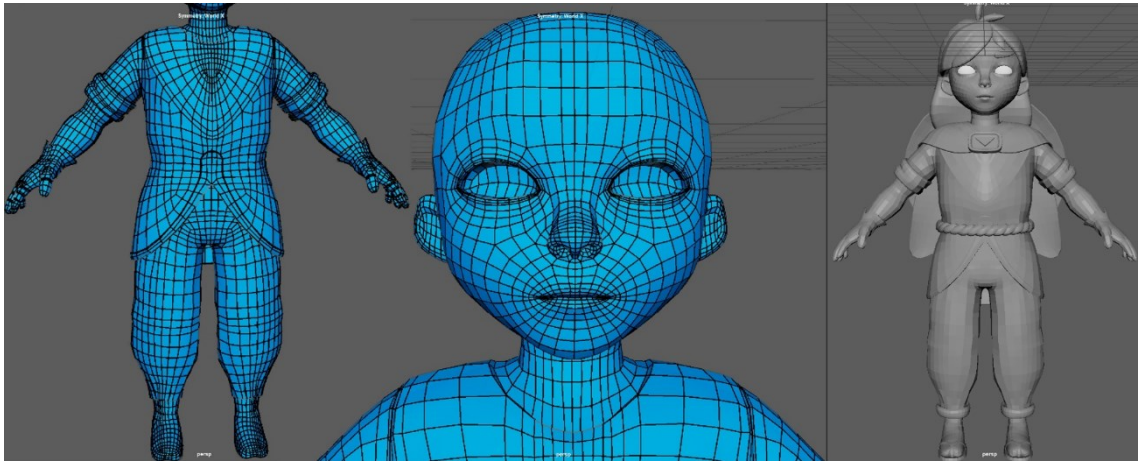
Suosittelun polygonimäärä videopelihahmolle vaihtelee pelimoottorin ja hahmon yksityiskohtaisuuden mukaan. Polygonin määrä riippuu myös, millaisesta pelistä on kyse, ja onko hahmo esimerkiksi taustalla oleva hahmo, jota ei nähdä läheltä vai pelattava päähenkilö hahmo, jolla tulee pystyä animoimaan ilmeitä ja taistelu animaatioita. Tällöin päähenkilön polygonimäärä olisi paljon korkeampi kuin tausta hahmon, jolla ei välttämättä ole yhtä korkeat vaatimukset. Pyrin pitämään low poly hahmon kokonais- polygoni määrän alle 20 000:ssa (kuvio 16). Jos hahmossa on liikaa polygoneja se kuormittaa pelimoottoria mikä hidastaa renderöinti aikaa ja voi vaikuttaa pelin yleiseen suorituskykyyn. (3D-ace 2024.)

Recommended Polygon and Triangle Counts for PCs/Consoles

Game asset	Polygon count (UE)	Triangle count (UE)	Polygon count (Unity)	Triangle count (Unity)
Low-detail character	10,000-20,000	20,000-40,000	5,000-10,000	10,000-20,000
High-detail character	20,000-60,000	40,000-120,000	10,000-30,000	20,000-60,000
Simple prop	500-2,000	1,000-4,000	250-1,000	500-2,000
Complex prop	2,000-10,000	4,000-20,000	1,000-5,000	2,000-10,000
Basic environment	10,000-50,000	20,000-100,000	5,000-25,000	10,000-50,000
Detailed environment	50,000-200,000	100,000-400,000	25,000-100,000	50,000-200,000

Kuvio 16. Suositeltu polygoni määrä PC ja konsolipelien aseteille riippuen pelimoottorista (3D-ace 2024)

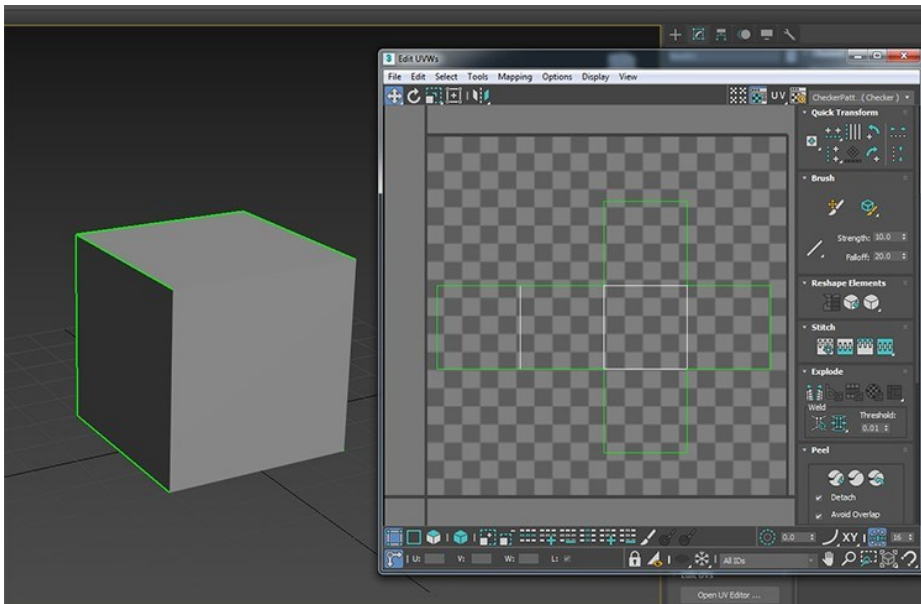
Omassa retopologyssä yhdistin kaulan paitaan kiinni, hihat käsivarsiin, hanskat ranteisiin ja niin edelleen, yhdistin eri vaatteiden osia kroppaan kiinni. Lopullisen low poly version polygoni määrä oli 16 484. Lähikuva vartalon ja kasvojen retopologysta ja lopullinen low poly malli on nähtävissä kuviosta 17. Lopullisen hahmon vartalo on kahdessa osassa: yläkroppa päästä paidan alareunaan ja housut ja jalat toisessa osassa. Luulen, että optimoidumpi ja parempi tapa olisi ollut pitää paita kokonaan erillisenä objektina ja yhdistää hahmon yläkroppa torsosta housuihin kiinni.



Kuvio 17. Vasemmalla: lähikuva vartalon ja kasvojen retopologysta. Oikealla: valmis low poly malli asusteineen

5 UV-MAPPAUS MAYALLA

UV-kartoitus eli UV-mappaus on 3D-mallinnuksen työvaihe, jossa kolmiulotteinen hahmo avataan auki saumoista ja sen eri osat asetetaan litteänä 2D-tilaan (kuvio 18). Tämä vaihe nimeltään UV unwrapping, on tärkeää 3D-mallinnusprosessissa, koska teksturointi perustuu 2D-kuviin, ei ole olemassa 3D-tekstuureja. (Calvello 2022.) Katsoin On Mars 3D (2020a) youtube kanavan videon oppiakseni miten käytännössä UV-unwrapping tehdään Maya ohjelmalla. Hän myös tiivistää UV-unwrapping prosessin hyvin, että se on prosessi, jolla 3D-objektin pinta konvertoidaan 2D-koordinaatti systeemiksi.

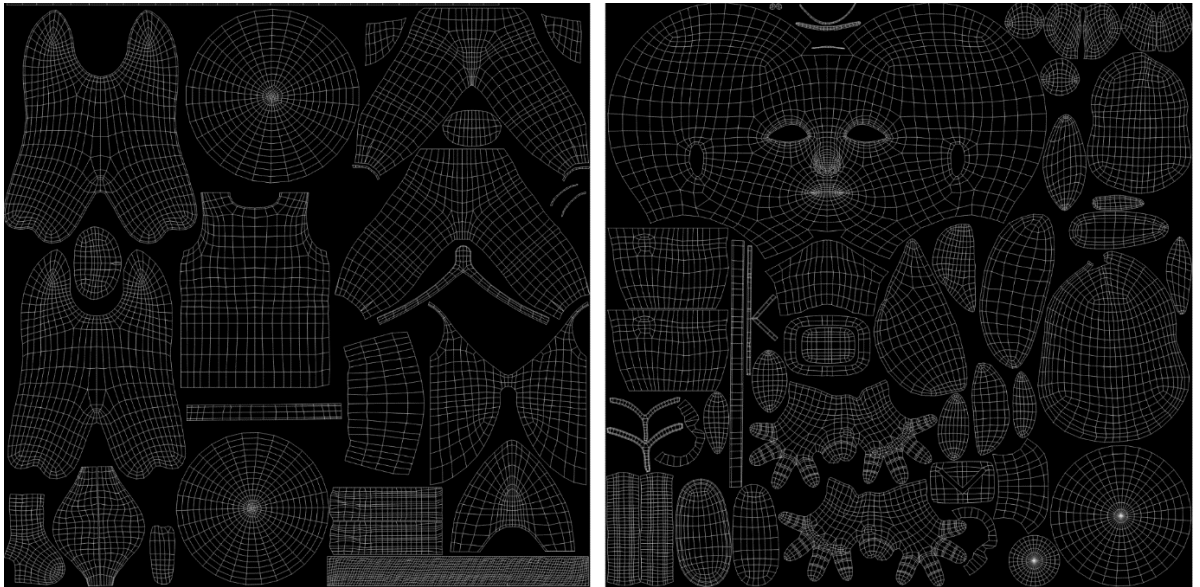


Kuvio 18. Kuution UV-mappaus. Merkityt saumat vihreinä ja oikealla UV:t auki levitetynä 2D-koordinaatille (Calvello 2022)

Omassa hahmossani yritin sijoitella saumat paikoille, jossa ne olisivat mahdollisimman piilossa, koska niiden kohdalla tekstuurin rajat tulevat pakostakin näky-mään. Myös kiinnitin huomiota, mihin eri osiin hahmossa piti tulla eri materiaali. Esimerkiksi yläkropassa hanskoihin halusin nahka materiaalin, käsivarsiin ihon ja hihaan kangasmateriaalin, joten jaoin alueet saumoilla omiksi UV-saariksi, koska itse 3D-mallissa olin yhdistänyt nämä osat. Saumojen määrittelyn jälkeen, malli “unwrapataan” eli sen UV:t levitetään esille 2D-koordinaatistoon ja niitä voi siirrellä ja muokata tarvittaessa. Osien suunta koordinaatistossa (pysty/vaakasuora)

määrittää mihin suuntaan tekstuurit levittyvät mallissa, joten yritin siihen kiinnittää huomiota.

Järjestin UV-saaret UDIM työkulun avulla kahteen laattaan. Kahdella UDIM-laattalla saan UV-saariin lisää resoluutiota verrattuna yhteen, joten saan tarkemmat yksityiskohtaisemmat tekstuurit. Jaoin karkeasti hahmon vaatteet ja kilven yhteen laattaan ja kaiken muun toiseen. Rajasin UV:t kahteen laattaan, että tekstuurit eivät kuormittaisi pelimoottoria liikaa (kuvio 19). Jälkikäteen ajateltuna minun olisi kannattanut laittaa kilven ja viitan UV:t kokonaan omille UV laatoille ja laittaa niille omat materiaalit.



Kuvio 19. Hahmon valmiit UV laatat

UV-mappauksen jälkeen määritin high poly mallin eri osiin omat vertex värinsä (kuvio 20). Tämä on tärkeää, koska teksturointi nopeutuu huomattavasti, kun seuraavassa työvaiheessa tämän perusteella saa luotua ID-mapin, joka rajaa eri värien alueet omiksi maskeiksi.

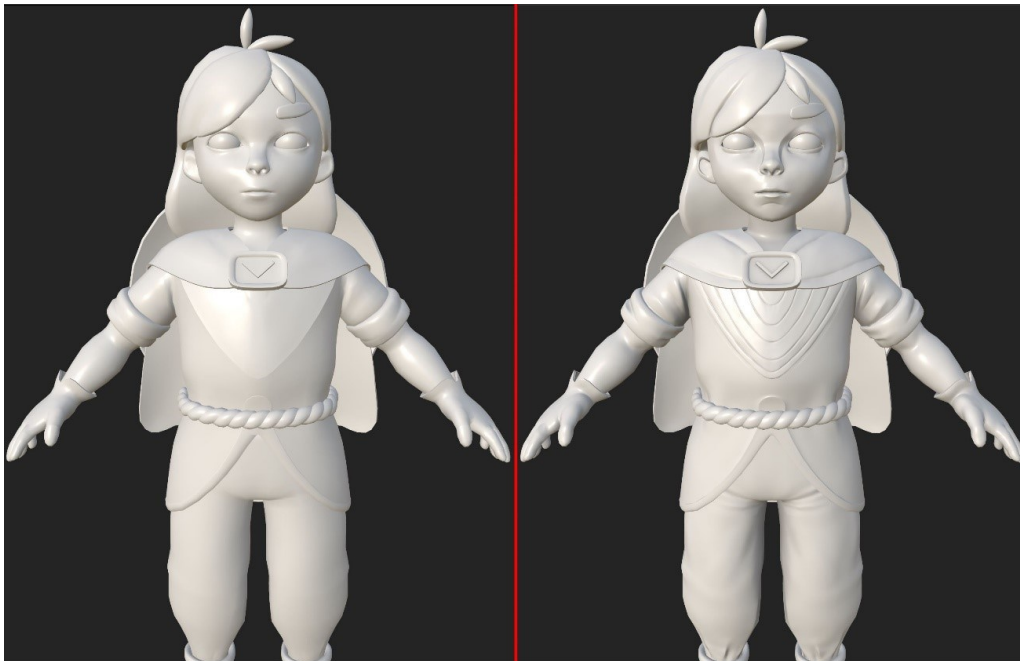


Kuvio 20. Vertex värit high poly mallissa

6 TEKSTUROIINTI ADOBE SUBSTANCE 3D PAINTER-OHJELMALLA

6.1 Baking mesh maps

Baking on prosessi, jossa 3D-mallista tallennetaan informaatiota tekstuuri tiedosto muotoon (bitmap). Useimmiten tähän tarvitaan 2 toisiaan vastaavaa 3D-mallia high poly- ja low poly malli. High poly mallissa on usein miljoonia polygoneja, joten se pystyy näyttämään 3D-yksityiskohdat korkearesoluutioisina. Low poly mallissa on usein vain tuhansia polygoneja, joten sitä on helpompi varastoida ja renderöidä. Baking prosessissa high poly mallin informaatio siirretään low poly mallille ja tallennetaan tekstuuriksi. Tekstuuri baking ansiosta saa mallin, jossa on high poly mallin korkeatasoiset yksityiskohdat, mutta alhaiset suorituskyökustannukset low poly mallista (kuvio 22).



Kuvio 22. Low poly malli ennen (vasen) ja jälkeen baking (oikea)

Substance Painter tuottaa erilaisia mesh mapeja baking prosessin aikana, joita voi hyödyntää teksturoinnissa. Monet filterit ja materiaalit mukautuvat mallin geometriaan käyttäen näitä beikattuja tekstuureja, joka tuottaa lopputuloksena realistisemmat ja yksityiskohtaisemmat tekstuurit. (Adobe 2023.) Minulla tässä työvaiheessa syntyi normal, world space normal, ID, ambient occlusion, curvature, position ja thickness mapit. Korjasin normal ja ambient occlusion mapeista

ongelmakohdat pois seuraten Stu Lloyd (CG Stu):n (2024) youtube tutorialia aiheesta.

6.2 Teksturointi

Tekstuuri on kuin käärepaperi, joka on kiedottu 3D-mallin ympärille. Se antaa 3D-mallille sen värin, kuvioinnin ja pintarakennetta visuaalisesti (Edesberg 2024). Peleissä tekstuurien tehtävänä on lisätä visuaalista realismia ja immersiota pelin maailmaan. Materiaaleja on kahden tyyppisiä perusmateriaalit ja PBR (Physically-Based Rendering) materiaalit. Perusmateriaalit ovat yksinkertaisia ja objekteihin, jotka eivät ole vuorovaikutuksessa pelin valaisujärjestelmän kanssa. PBR materiaalit jäljittelevät realistisesti, miten valo vaikuttaa eri pintoihin albedo, roughness, metallic ja normal mapin kautta. (Rocketbrush studio 2023.) Omalle hahmolleni tein PBR-materiaalit.

Määritin pohjavärit eri osille hyödyntäen ID-mappia ja käytin filttäreitä, kuten ambient occlusionia korostamaan varjoja, ja light filteriä korostamaan kohokohtia objekteissa. Lisäsin vaatteisiin kangas materiaaleja ja metallisiin osiin smart materiaaleja, jolla sain niihin realistisia naarmuja ja metallin kiiltoa. Katsoin eri Youtube-videoita saavuttaakseni haluamani ilmeen tekstuureille. Esimerkiksi hiuksien kohdalla seurasin On Mars 3D:n (2020b) tutorialia, jolla sain hiuksiin syvyydellistä hiussuortuva tekstuuria hyödyntämällä procedural materiaalia height kanavassa (kuvio 23).



Kuvio 23. Valmiit tekstuurit renderoituna Substance Painterissa

Valmiit tekstuurit ovat nähtävissä kokonaisuudessaan kuviossa 24. Halusin tekstuureista tyyllitellyt, mutta silti suhteellisen realistiset. Eri materiaalit kuten metallit, vaatteiden kankaat ja hiukset piti olla tunnistettavissa ja mielestäni onnistuin tässä tavoitteessa. Yritin pitää hahmon tekstuurien värit mahdollisimman lähellä konseptitaiteen värejä.



Kuvio 24. Tekstuurit Substance Paintin viewportissa

Kun tekstuurit olivat valmiit, huomasin että olin valinnut väärän shader profiilin niille. Shader kuvaa objektin pinnan ominaisuuksia ja kontroloi miten valo vaikuttaa niihin. Shader profiileja on erilaisia ja niillä on eri ominaisuuksia. (Elixier 2024.) Tekstuurit olivat specular/glossiness profiilissa, jotka käyttävät enemmän muistia, koska ne perustuvat RGB mappeihin. Metallic roughness profiilin tekstuurit ovat optimoidut Unreal Engine pelimoottorille, koska ne perustuvat harmaa-sävy mappeihin ja käyttävät vähemmän muistia. (Akram 2022.) Muutin syntyneet tekstuurikartat eli tekstuuri mapit Photoshopissa metallic roughness profiiliin seuraten Luminous Labsin (2021) youtube-tutoriaalia. Tekstuurikartat ovat kuva

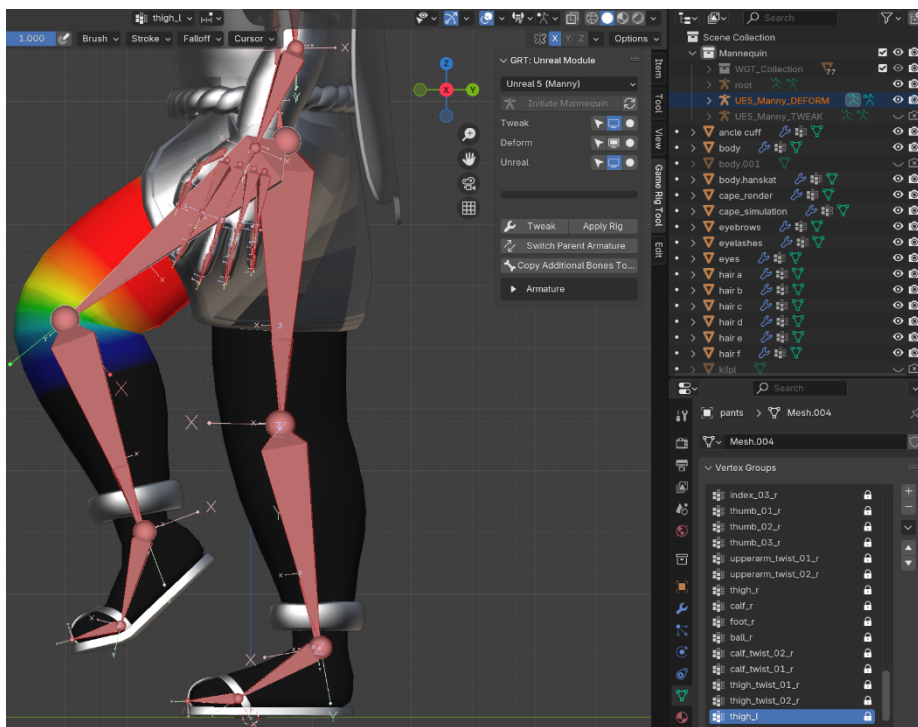
tiedostoja (esimerkiksi jpeg tai png), jotka luodaan virtuaaliobjektille antamaan tietoa kohteen pinnan luonteesta (Render That 2024). Exporttasin korjatut tekstuurit uudestaan ulos Substance painterista Unreal Engineille optimoidussa muodossa (kuvio 25).



Kuvio 25. Valmiit tekstuurikartat. Ylin rivi: base color, keski rivi: normal mapit, alin rivi: ambient occlusion, roughness ja metallic mapit yhdistettynä

7 HAHMON RIGGAUS BLENDERISSÄ

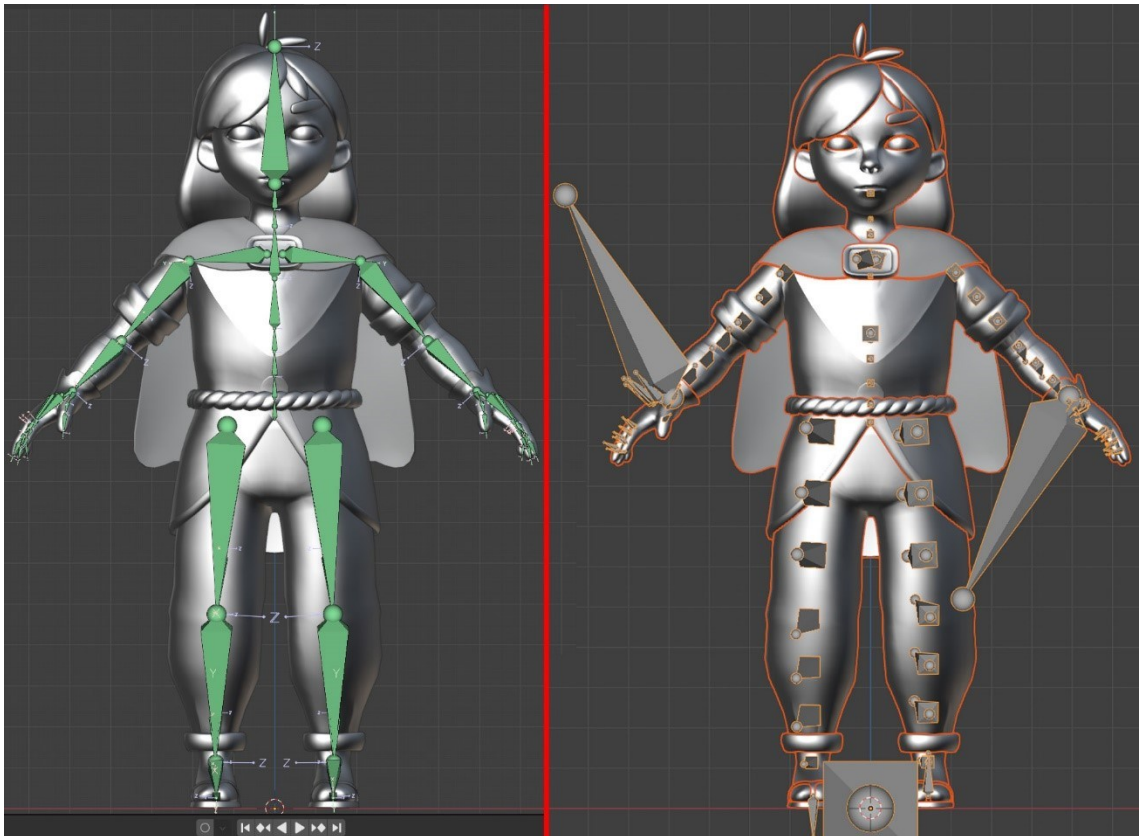
Riggaaminen eli “rigging” on prosessi, jossa hahmo valmistellaan animoitavaksi. Rigin avulla animaattorit voivat manipuloida 3D-mallia ja liikuttaa sitä haluamallaan tavalla. Käytännössä riggaaminen tarkoittaa luurankon tekemistä hahmolle, joka koostuu toisiinsa liittyvistä nivelistä ja luista. (Hassenfratz 2024.) Luut asettellaan oikeille paikoille ja niille määritetään vaikutusalueet “weight paint” tilassa Blenderissä eli luille maalataan mihin alueeseen meshissä ne vaikuttavat (kuvio 26).



Kuvio 26. Reisiluun weight paint väreillä kuvattuna. Punaisen alueella luun vaikutus on korkea eli 1.0, sinisen värin alueella vaikutus on olematon eli 0.0 (Wiki.Blender 2024)

Jotta saan 3D-hahmoni toimimaan Unreal Engineessä olevilla valmiilla animaatioilla, minun piti rigata hahmo pelimoottorin omaan luurankoon. Käytin riggaamiseen Game Rig Tools lisäosaa, joka muuttaa Unreal Enginen rigin muotoon, jota voi käsitellä blenderissä. Luiden asetteluun ja game rig toolssin käyttöön katsoin CGDiven tutorialin aiheesta (CGDive (Blender Rigging Tuts) 2022). Hahmon ja sen asusteiden weight paintin tekemiseen katsoin Yami 3D:n (2024) videon. Viitallani oli paksuus, joten jotta saan Unrealissa viitan kangassimulaation

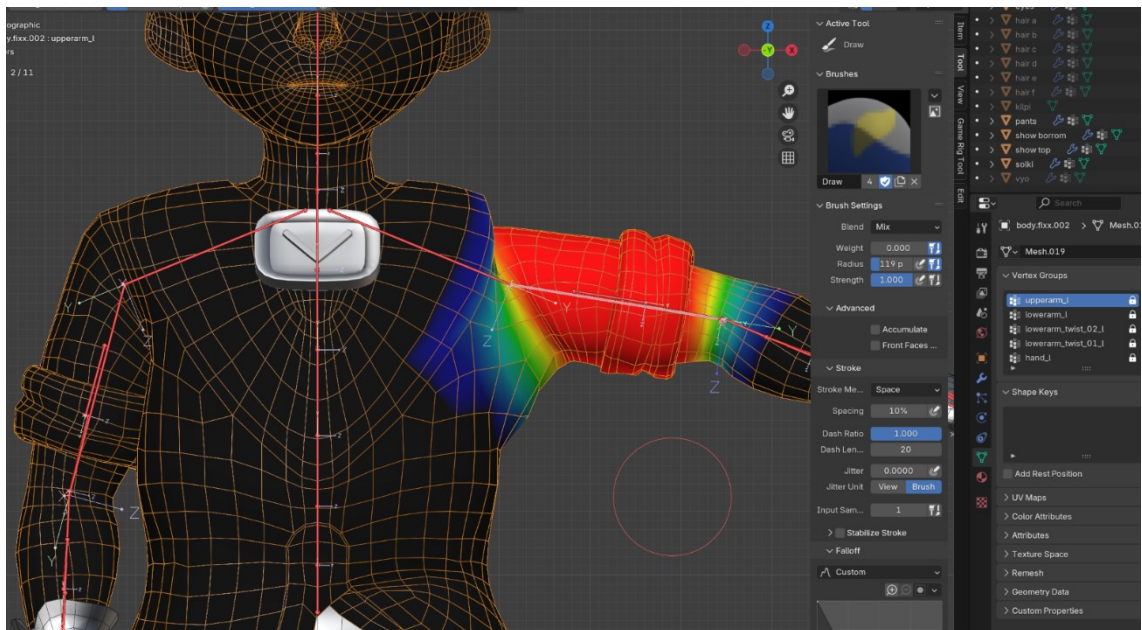
toimimaan kunnolla, minun piti tehdä siitä simuloitava versio, jossa paksuutta ei ollut. Paksuudella oleva viitta on render versio, joka toistaa kangassimulaation liikkeitä ja näyttää tekstuurit. (MattLakeTA 2022.) Riggasin molemmat versiot viitasta siirtämällä niihin jo maalaamani weightit hahmon vartalosta ja muokkamalla niitä niin, että viitta seuraa vain hahmon yläkropan liikkeitä (Royal Skies 2020). Laitoin molemmille viitoille omat materialit; `cape_render` ja `cape_simulation`. Exporttasin hahmon valmiin rigin kanssa Unreal Engineen (kuvio 27).



Kuvio 27. Vasemmalla valmis riggi Blender muodossa, oikealla Unreal Engine muodossa exportatessa sen muiden meshejen kanssa Unreal Engineen

Hahmoa rigatessa ja riggiä testatessa huomasin, että se ei toiminut niin hyvin kuin olisin halunnut. Minun olisi pitänyt rigata hahmo heti kun sain retopologyn tehtyä, ja siten olisin huomannut hahmon topologiassa olevat virheet ja olisin voinut korjata ne tässä vaiheessa prosessia. Olin tehnyt hahmon topologian kanssa kriittisiä virheitä ja hahmon olkapää alueelta ei löydy nyt oikeanlaista edge flowta, joka mahdollistaisi oikeanlaisen deformatumisen kättä nostaessa ylemmäs A-asennosta. Olin ajatellut paidan topologian, miten se paita objektissa pitäisi olla,

enkä miten ihmisen käsivarren topologia pitäisi olla. Kun yhdistin paidan käsivarteen kiinni, olin unohtanut korjata käsivarren topologian olkapäästä optimoiduksi. Riggissä testatessa törmäsin tähän ongelmaan, nyt hahmo ei pysty nostamaan käsivartta paljon A-asennosta korkeammalle ilman, että hahmon muoto vääristyy (kuvio 28). Hahmon käsivarsia ei myöskään voi nostaa korkeammalle, kuten ihan ylös asti, koska sen topologia ei ole oikein sallimaan tällaista liikerataa. Koska huomasin nämä ongelmat vasta sen jälkeen, kun olin tehnyt hahmon UV-mappauksen ja tekstuurit, en voinut enää korjata topologia ongelmia, ilman että minun olisi pitänyt tehdä nämä työvaiheet uudestaan. Minun täytyy myöhemmin tehdä hahmon topologia uudestaan optimoidummaksi animoimista varten ja sitten tehdä UV-mappaus ja tekstuuritkin uudestaan. Samalla voin korjata muut virheet mitä tein UV:den kanssa ja teksturoinnissa. Jatkossa tiedän rigata hahmon testiksi retopologyn jälkeen niin löydän ongelmakohtat ja voin korjata ne helpommin.

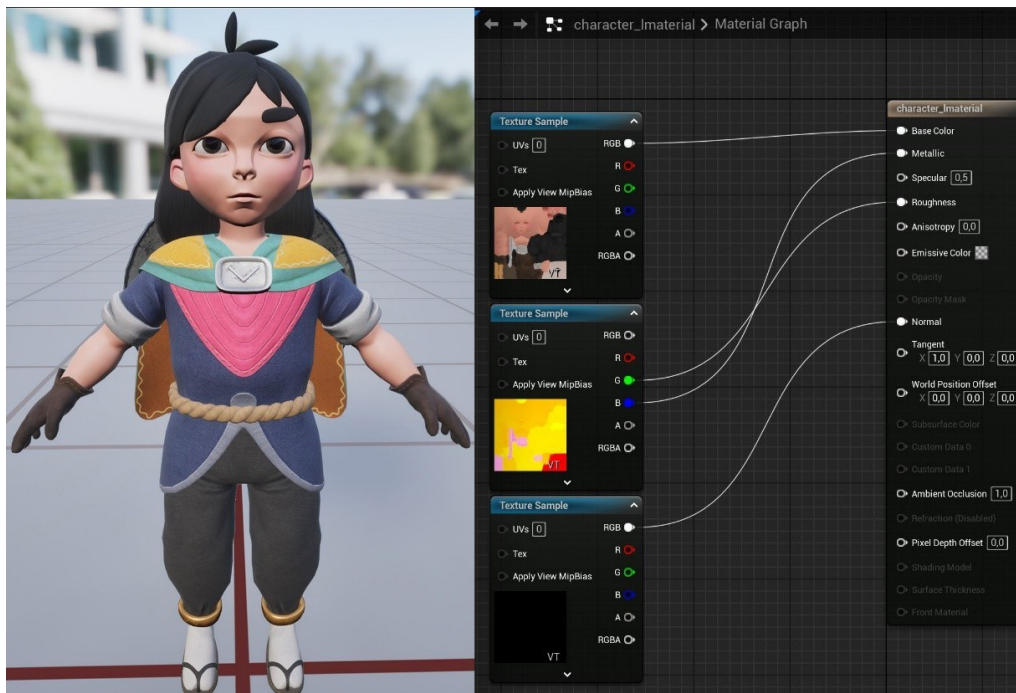


Kuvio 28. Rigin ongelmakohta käsissä

8 HAHMO PELIMOOTTORI UNREAL ENGINESSÄ

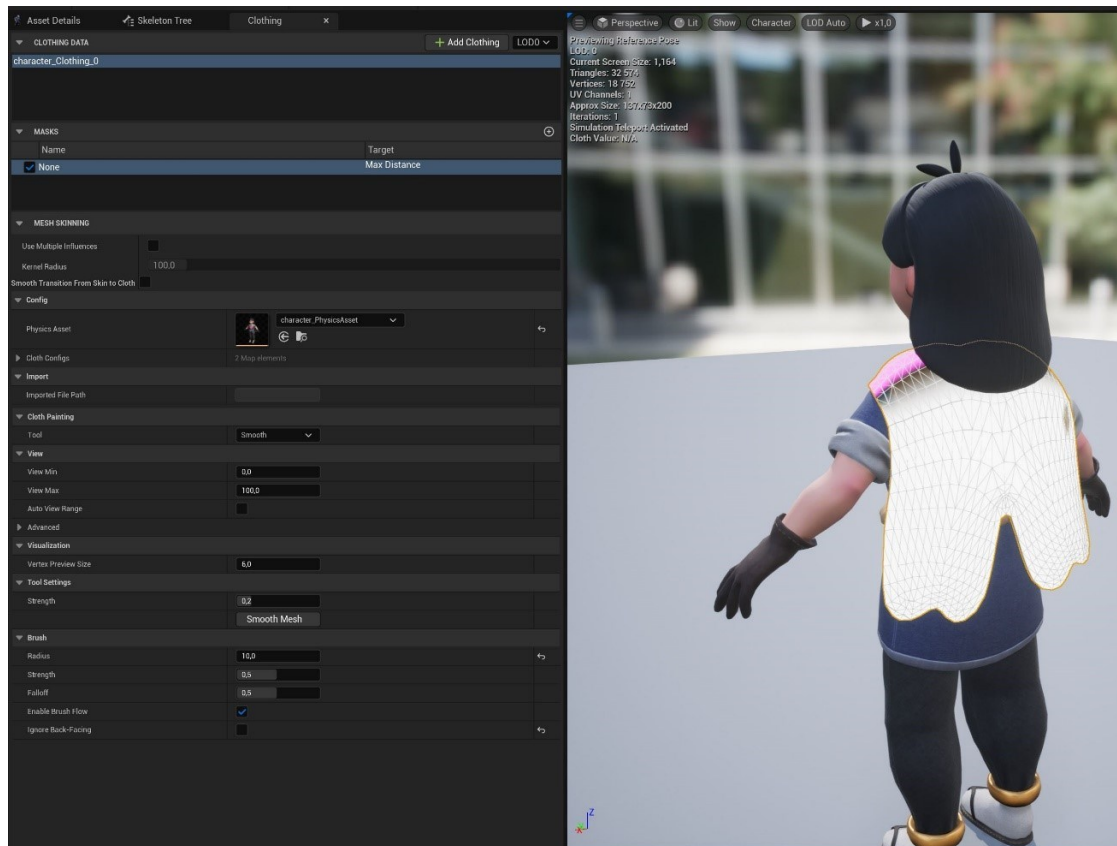
Kun 3D-malli, tekstuurit ja hahmon riggaus olivat valmiit, niitä täytyy testata pelimotorissa (Unreal Engine 5). Tätä varten nämä resurssit täytyy tuoda sinne. Tuontiprosessin aikana eri resursseille pitää määrittää oikeat asetukset riippuen siitä onko kyseessä 3D-malli vai tekstuurikartat, joita importataan. Resurssien tuontiin katsoin James, How Do I (2022) youtube-kanavan tutorialin. Koska olin käyttänyt UDIM-työnkulkua UV-mappien kanssa, minulla oli yhden UV-laatan sijasta niitä 2, joten myös tekstuureita oli kolmen tekstuurikartan (Base color, Normal ja OcclusionRoughnessMetallic) sijasta 6. Katsoin the Craters (2023) kanavan videon, että sain UDIM materialit importattua oikein.

Teksturointia varten olin määrittänyt kaikille hahmon osille yhden materiaalin; character_material, joka toistaa hahmon tekstuurit eri tekstuurikarttojen kautta. Tämä materiaali tuli pelimoottoriin hahmon meshin importtauksen yhteydessä. Jotta tekstuurit saa toimimaan Unreal Engineessä, tekstuurikartat pitää yhdistää materiaaliin oikealla tavalla Material Graphissa, jossa materiaalia saa editoitua (James, How Do I 2022). Tekstuurikartat yhdistetään oikeisiin kohtiin nodejen avulla (kuvio 29).



Kuvio 29. Vasen: hahmo tekstuureilla Unreal Engineessä. Oikea: tekstuurikartat yhdistettynä oikeille paikoille

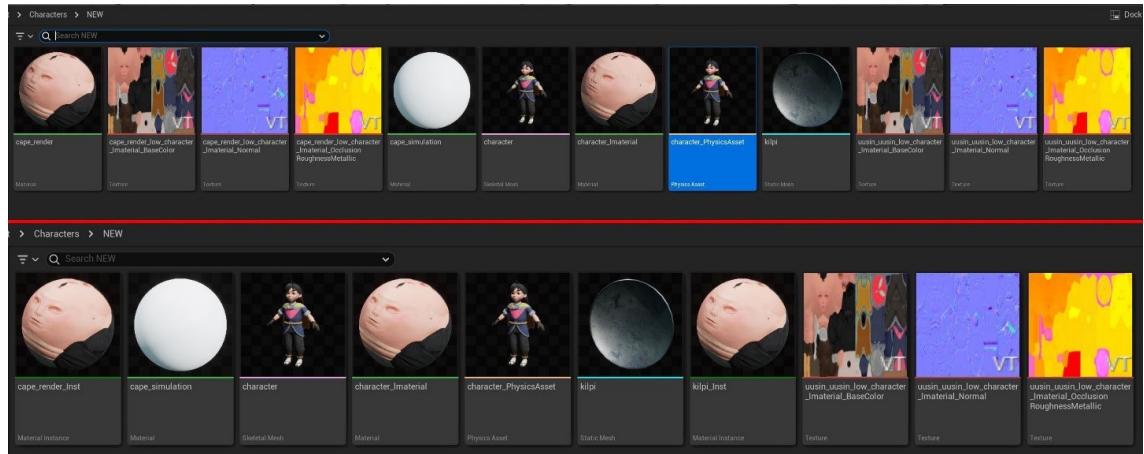
Kangassimulaatio toimii Unreal Engineissä niin, että siellä meshiin maalataan, miten kauas kangas voi liikkua sen weight paint positioista (Kuvio 30). Unreal Engineissä kangassimulaatio koskee kaikkea, mikä on saman materiaalin slotin sisällä. Minun olisi pitänyt määrittää ennen teksturointia viitan meshiin oma materiaali, joka on erillinen muista osista hahmossa. Nyt viitan tekstuuri on character_lmaterial materialissa. Jotta saan kangassimulaation pelkästään koskemaan viittaa, minun piti määrittää sille jälkikäteen omat materiaalit (cape_render ja cape_simulation). Tämä ei ole pelimoottorin resurssien käytön kannalta optimaalista, koska minun piti laittaa kopiot hahmon tekstuurikartoista uudestaan Unreal Enginen sisään cape_render materiaaliin, jotta sain viitan toistamaan tekstuurit simuloituessa. Tämä kuormitti systeemiä turhaan, joka luultavasti oli syy, miksi tekstuurikartoissa oli visuaalista erroria. (MattLakeTA 2022.)



Kuvio 30. Viitan kangassimulaation ”Cloth Paint”. Valkoinen osa simuloituu, vaaleanpunainen seuraa luiden liikettä. Musta on ”smooth” tasoittava välialue

Korjasin tekstuurit paremmiksi luomalla character_lmaterialista materiaali instanssit ”material instance” viitalle ja kilvelle. Tämä luo kopion materiaalista, joka toistuu samojen tekstuurikarttojen kautta kuin alkuperäinen versio. Pystyin siis

muuttamaan viitan ja kilven meshien tekstuurit näiksi instance materiaaleiksi ja poistamaan vanhan cape_render materialin ja sen tekstuurikartat (kuvio 31). Tämä vähentää pelimoottorin kuormitusta, koska jokainen erillinen materiaali tuottaa oman shader koodin (Unreal Engine 2019).



Kuvio 31. Ylhäällä tekstuurit alussa, alhaalla optimoidumpi tilanne käyttämällä material instanceja

Kaikista optimoiduin tilanne pelin kannalta on kuitenkin se, että ennen teksturointia ja UV-mappausta määrittää niille osille hahmoa omat materiaalit, joiden pitää simuloitua tai muuten olla erillisiä muusta hahmosta, esimerkiksi välillä olla näkymättömiä. Tekstuurit vieläkin sekoilevat välillä jostain syystä, jos niihin zoomaa liian lähelle (kuvio 32). Saan korjattua ongelman tekstuurikarttojen asetuksista, mutta se tulee välillä takaisin jostain syystä, joten tilanne ei ole optimaalinen.



Kuvio 32. Tekstuurit sekoilemassa Unreal Engineissä

Jotta hahmon saa liikkumaan ja viitta simuloitumaan oikein, hahmon kollissionit piti korjata oikeanlaisiksi “character_physicsAsset” osiosta. Kollissionit ovat vähän läpinäkyviä liiloja palloja, jotka ovat hahmon piilotettujen luiden ympärillä (kuvio 33). Muutin luille määritellyt kollission boksit oikeisiin kokoihin ja kohtiin, ja yritin korjata nivelien rotaatioita vastaamana ihmisten nivelten liikeratojen rajoja (Manim8 2022). Unreal Enginen hahmon fysiikka asetti (physics asset) on perusedustus hahmosta fysiikka moottorille pelimoottorissa (Helios 2021).

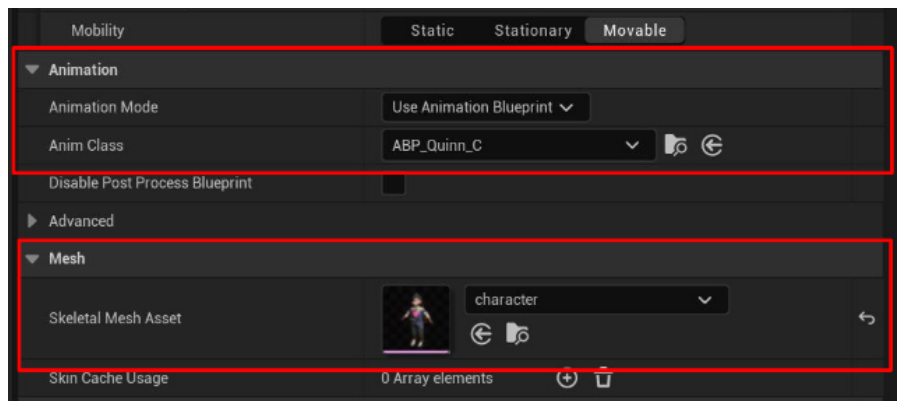


Kuvio 33. Hahmon physics asset: oikealla hahmon kollission boksit (liilat pallot) alussa. Vasemmalla ne korjauksen jälkeen

Ilmeni, että aikaisemmin hahmoni Unreal Engine version rigissä oli tapahtunut jokin bugi, joka oli siirtänyt yhden selkärangan luun väärään paikkaan ja vaikka korjasin sen manuaalisesti, niin tämä sekoitti hahmon fysiikan täysin, joten hahmo ei pysty simuloitumalla kaatumaan oikein. Hahmon viitan kangassimulaatio kuitenkin toimii ja hahmo toimii Unreal Enginen oletushahmon animaatioilla, joten voin yrittää myöhemmin rigata hahmon uudestaan, ja katsoa korjaako se tämän ongelman.

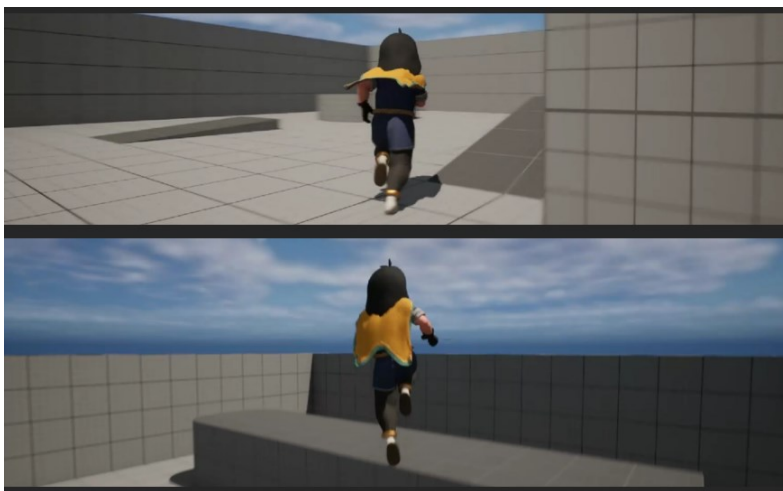
Jotta oma hahmoni toimii pelihahmoa kontrolloidessa, Unreal Enginen oletushahmo pitää korvata omallani. Tämä tehdään Third Person Blueprints hahmon asetuksista muuttamalla skeletal mesh assetin valikosta oletushahmon tilalle

oma hahmo (kuvio 34). Unreal Engine tunnistaa hahmon rigin, koska siinä on samat luut kuin oletushahmon rigissä, joten oma hahmoni toimii pelimoottorissa samoilla animaatioilla ja samalla tavalla kuin oletushahmo toimii ja seuraa animaatio blueprinttejä (CGDive (Blender Rigging Tuts) 2022). Unreal engineen toiminta perustuu “Blueprint visual scripting” eli blueprint visuaaliseen skriptaus systeemiin, joka on visuaalinen ohjelmointikieli, jossa käytetään nodeja luomaan pelin toiminnan elementit Unreal editorissa. Blueprint on yleistermi Unrealin ohjelmointikielessä eri elementeille, joita nodeilla voi kontrolloida. (Epic Games 2024.)



Kuvio 34. Third person blueprint asetukset, että hahmo toimii Unrealissa

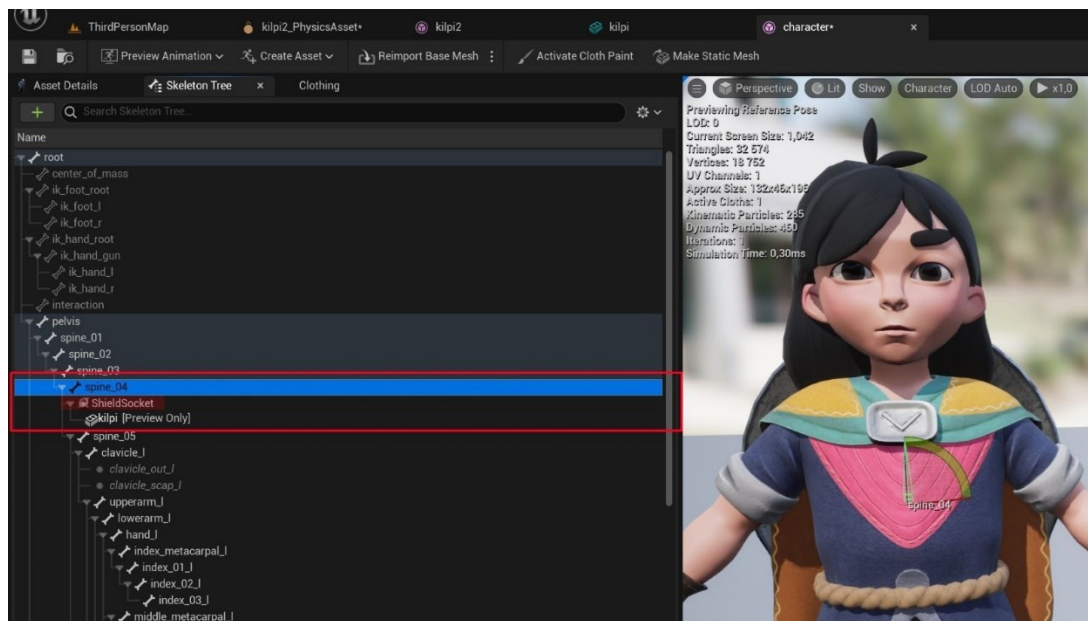
Testasin hahmoa, ja se toimii eri animaatioilla, joita Unreal enginestä löytyy valmiina. Pelikentällä pelatessa, hahmo toistaa pelaajan haluamat liikkeet sulavasti ja viitan kangassimulaatio toimii hyvin. Hahmon housut klippaavat hieman paidan läpi juostessa. Tämä luultavasti johtua siitä, että hahmo on kahdessa osassa. Hahmolla pystyy juoksemaan ja hyppimään haluamaansa suuntaan (kuvio 35).



Kuvio 35. Hahmo juoksemassa ja hyppäämässä pelikentällä pelatessa

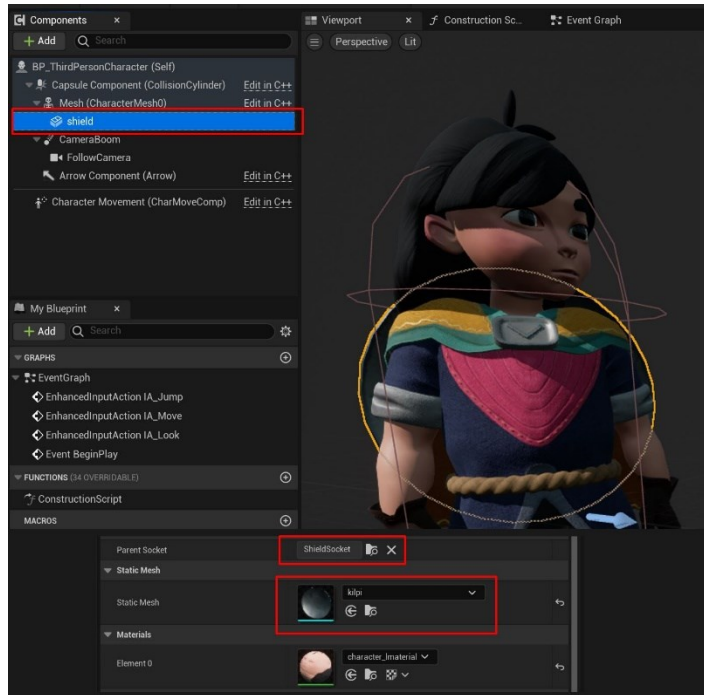
Halusin testata kilven liittämistä hahmon selkään Unreal Engineessä, niin että se liikkuisi mukana hahmon animoitaessa. Halusin saada kilven hahmon selkään näkyviin ja pois näkyvistä nappia painamalla niin että viitta voisi liehua vapaasti, kun kilpi ei ole selässä.

Kiinnitin hahmoni skeletal meshin luurangon neljänteen selkärangan luuhun kilven kiinni socketilla. Socketin saa painamalla spine_04 päällä hiiren oikeaa näppäintä ja valikosta valitaan add socket. Socket kannattaa nimetä myös tunnistettavalla tavalla, itse nimesin omani ShieldSocket. Lisäsin kilpeni shieldSockettiin preview assetin, jotta pystyn asettelemaan kilven paikalle, jossa haluan sen olevan hahmon selässä (Just Another Dang How To Channel 2023.) Luurangon hierarkia, sieltä löytyvä tehty socket ja kilven paikka selässä on visuaalisesti näkyvillä kuviossa 36.



Kuvio 36. Kilven socket laitettuna hahmon spine_04 luuhun skeletal mesh asse-
tissa

Seuraavaksi third person blueprintistä lisätään static mesh ja sen asetuksista parent socketiksi laitetaan tehty ShieldSocket ja static meshin valikosta valitaan kilven 3D-malli (kuvio 37). Kilpi seuraa nyt spine_04 luun liikettä ja liikkuu mukana hahmon liikkeessä pelikentällä. Katsoin tähän youtube tutoriaalia socketin käytöstä Unreal Engineessä. (Just Another Dang How To Channel 2023.)



Kuvio 37. Lisätyn static mesh kilven asetukset third person blueprintistä

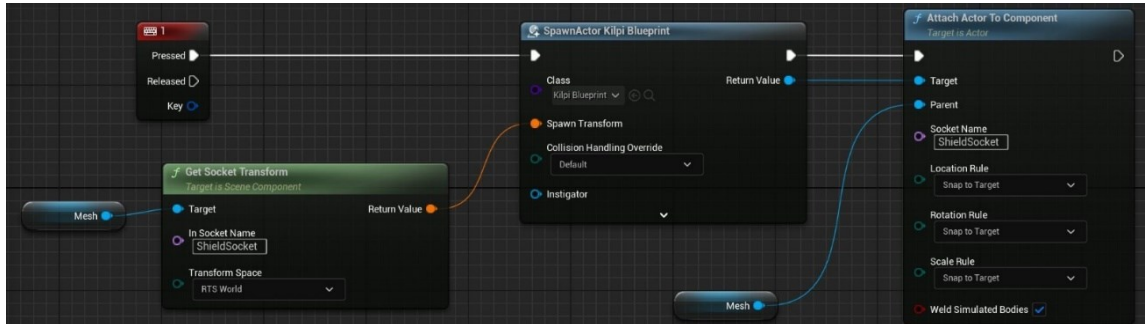
Ongelma on, että kun hahmo juoksee pelikentällä kilven kanssa, niin viitta menee vieläkin kilven läpi. Myös hahmon kyynärpää menee siitä läpi (kuvio 38). Viitta ei huomioi static mesh kilven kollissioneita ollenkaan.



Kuvio 38. Viitan kangassimulaatio tulee läpi kilvestä hahmon juostessa

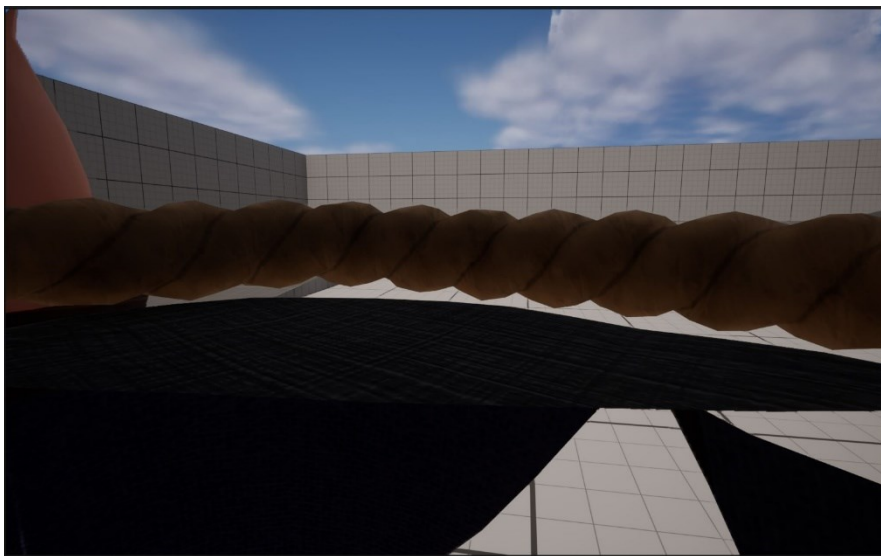
Jotta sain kilven ilmaantumaan hahmon selkään nappia painamalla, minun piti ensin poistaa sen lisätty mesh ThirdPersonCharacter Blueprintistä components osiosta, jotta hahmo juoksee ilman kilpeä alussa, kun mitään nappia ei ole painettu. Katsoin tutorialia siitä, miten mesh actorin saa spawnattua (eli ilmestymään) socketin sijaintiin. Seurasin tutorialia ja rakensin videon mukaan tälle toiminnolle blueprintin (kuvio 39). Haluan että kilpeni ilmestyy sille tehdyn

ShieldSocket paikalle nappia 1 painamalla. (Aspland 2022.) Minun piti muuttaa kilven static mesh blueprint muotoon niin sain laitettua sen oikein SpawnActor Class valikosta (Adam The Chips 2022).



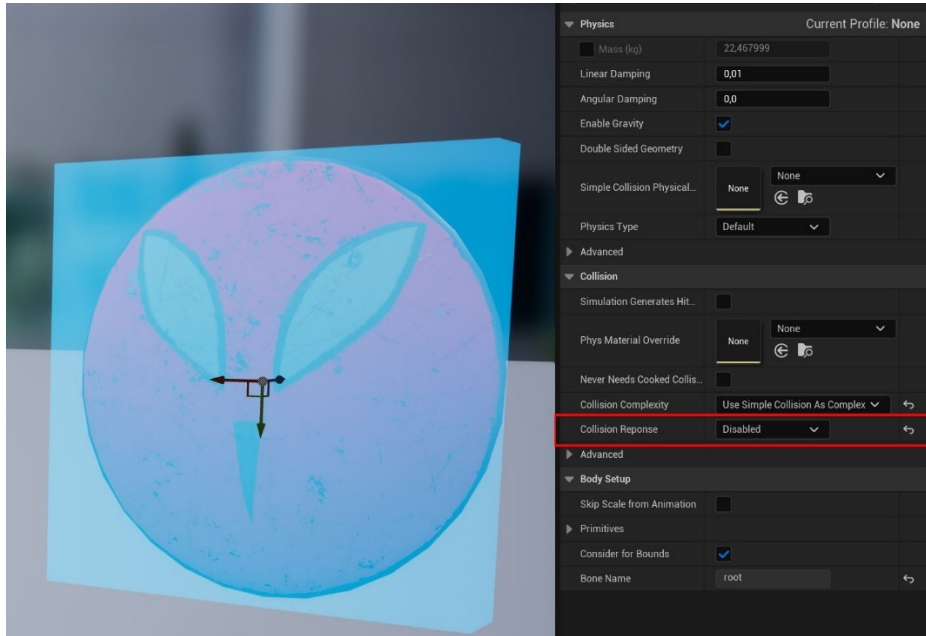
Kuvio 39. Blueprint kilven ilmestymiseen 1 nappia painamalla selän ShieldSocket sockettiin

Tätä toimintoa testatessa huomasin, että kilpi kyllä tulee hahmon selkään oikein kun 1 painaa, mutta kamera siirtyy oudosti hahmon sisälle mitä ei pitäisi tapahtua. Tämä on “bug” eli virhe pelin toiminnassa (kuvio 40).



Kuvio 40. Kamera hahmon sisällä virheellisesti kilven ollessa hahmon selässä
Yritin korjata viitan kangassimulaation menemisen kilven läpi laittamalla kilven static meshin tilalle kilven, johon olin laittanut yhden luun blenderissä, joten objekti on nyt skeletal mesh ja kilvellä on oma kollission boksi. Laitoin tämän skeletal meshin kilven blueprintin ThirdPersonCharacter blueprintin SpawnActor Blueprint Class kohdalle (kuvio 39). Tutkin netistä, miten saan kameran siirtymis bugin

korjattua. Löysin Unreal Engine forumilta vastauksen tähän ongelmaan (Unreal Engine 2022). Kiven PhysicsAssetista Collision Response piti vaihtaa Enabled tilasta Disabled tilaan niin bugi lähtee pelatessa (kuvio 41).



Kuvio 41. Skeletal mesh kilven asetuksen korjaus. Sininen on kilven kollisio boksi

Nyt hahmolla pelatessa hahmo juoksee ilman kilpeä, kunnes 1 näppäintä painetaan ja kilpi ilmaantuu hahmon selkään sen ShieldSocket paikalle. Viitan kangassimulaatio menee edelleen kilvestä läpi. Pitää jatkaa työstöä blueprinttejen kanssa, että saan kilven ilmaantumaan ja katoamaan hahmon selästä 1 nappia painamalla ja pitää tehdä jatkotyötä, että viitan kangassimulaatio ei menisi kilvestä läpi, kun kilpi on hahmon selässä.

Screen recordasin hahmostani videon ThirdPerson kamerasta, josta näkyy, miten hahmo toimii pelissä sillä pelatessa. Olen laittanut linkin tähän videoon opinäytetyöni liitteeseen 1. Hahmo reagoi W, A, S ja D näppäimiin hahmon juoksemisen suunnan suhteen ja se hyppää välilyönti näppäimestä. 1 näppäintä painessa kilpi ilmaantuu hahmon selkään. Hahmo on nähtävissä portfolio sivuiltani, joista löytyy lisää kuvia ja videoita siitä (liite 1).

9 POHDINTA

Opinnäytetyöni tavoitteena oli luoda peliin optimoitu, game ready hahmo, joka toimii Unreal Engine pelimoottorissa. 3D-hahmon teon työkaluja olivat eri 3D-ohjelmat, joita käytin prosessissa eli Zbrush, Maya, Substance Painter ja Blender. Halusin käyttää ammattilaisten käyttämiä 3D-ohjelmia, mutta tiedonhaku vaiheessa huomasin, että minun oli hankalaa löytää täysin puolueetonta lähdettä siitä mitä ohjelmia studioissa käytetään. Sivut, joista näitä ohjelmista puhuttiin, oli 3D-ohjelmien valmistajien omat sivut. Halusin keskittyä käyttämään ohjelmia, jotka voisivat auttaa minua tulevaisuudessa työllistymään, joten käytin Maya ja Zbrushia mallintamiseen.

Hahmon tuotantoprosessin aikana avasin päätutkimuskysymykseni auki eli millainen on 3D-hahmon tuotantoprosessi. Se on minun työni kannalta tiivistettynä high polyn sculptaaminen, low polyn teko eli retopology, uv mappaus, teksturointi ja riggaaminen. Vein hahmon pelimoottoriin, tein sen toimivaksi siellä ja testasin sen käyttöä. Sain viitan kangassimulaation toimimaan ja hahmon toistamaan pelimoottorin animaatiot. Hahmolla pystyy pelaamaan pelikentässä.

Hahmoni voisi olla vieläkin optimoidumpi peleihin. Olin tehnyt retopology vaiheessa virheitä hahmon topologiassa, niin hahmon riggi ei toimi niin hyvin kuin se olisi voinut toimia. Hahmon liikerata rigissä ei ole paras mahdollinen, esimerkiksi hahmo deformatuu väärin, jos sen käsivarret laitetaan paljon ylemmäs kuin mitä ne ovat A-asennossa. Tiedän jatkossa kiinnittää vielä enemmän huomiota topologiaan ja testaan sitä riggaamalla hahmon ennen kuin siirryn UV:den tekoon. Tällöin voin korjata topologia ongelmat mitkä näkyvät riggiä testatessa. Optimoidummalla topologialla hahmon pintamäärä olisi myös pienempi koska minun piti lisätä retopology vaiheessa joitain looppeja, että sain rigin toimimaan paremmin. Tiedän jatkossa laittaa oman materiaalin osille, johon haluan kangassimulaation ja esimerkiksi osille, jotka voin haluta joskus olevan näkymättömiä pelissä.

Minun olisi enemmän pitänyt pyytää muita 3D-taiteilijoita katsomaan työtäni läpi sen eri vaiheissa. Muutamalta taiteilijalta kysyin apua prosessin aikana, mutta lopputulos olisi vielä parempi, jos olisin enemmän kysynyt apua ihmisiltä, jotka ovat aikaisemmin tehneet pelihahmoja. Samoin minulla olisi kannattanut olla

jokin kontakti ammattilaiseen pelifirmassa, joka olisi osannut auttaa minua prosessin aikana korjaamaan virheeni ajoissa. Minulla ei ole ulkopuolista mielipidettä siitä, tulisiko hahmoni oikeasti toimimaan tarpeeksi hyvin oikean peliproduktion kannalta, olen joutunut vetämään rajallisen tietoni pohjalta omat johtopäätökset hahmon toimivuudesta pelissä. Koska minulla ei ollut tällaista kontaktia, se hidasti työskentelyä ja teki prosessista vaikeampaa. Jatkossa pyrin luomaan tällaisia kontakteja ja pitämään niitä paremmin yllä.

Onnistuin kuitenkin tekemään pelissä toimivan hahmon, joka toimii Unrealin valmiilla animaatioilla ja blueprinteillä. Koska sillä pystyi korvaamaan oletushahmon, se toimii miten Unreal Enginen oletushahmo toimii, ja sitä voisi käyttää peliprojektissa kuin oletushahmoa. Opin projektin aikana paljon pelihahmon mallintamisesta, teksturoinnista, riggaamisesta ja Unreal Enginen käytöstä sekä pelihahmon teknisistä vaatimuksista. Pystyn tekemieni virheiden ja oppimieni asioiden kautta tekemään seuraavasta pelihahmostani vielä optimoidumman ja paremman.

Hahmon kehitystä voisi jatkaa vielä eteenpäin riggaamalla hahmon hiukset ja laittamalla niihin omat fysiikat, niin ne reagoisivat hahmon liikkeisiin. Voisin tehdä hahmon topologian uudestaan optimoidusti niin että paita olisi erillinen objekti muusta vartalosta ja vartalon ja paidan topologia olisi myös parempi. Voisin myös rigata hahmon uudestaan paremmin, nämä asiat korjaisivat hahmon liikeradan paremmaksi rigissä. Hahmolle voisi tehdä myös omia animaatioita kuten kilven heittäminen ja puolustus animaation kilvellä, jotka voisi laittaa blueprinteillä toimivaksi elementiksi pelissä eli jotain näppäintä painamalla hahmo tekisi nämä toiminnot. Kilven heittämisellä voisi ratkoa jotain puzzleja pelissä, eli siitä voisi tehdä toimivan elementin pelin pelaamisen kannalta.

LÄHTEET

3D-ace 2024. Does Polygon Count Matter in 3D Modeling for Game Assets? Viitattu 3.5.2024 <https://3d-ace.com/blog/polygon-count-in-3d-modeling-for-game-assets/>.

Adam The Chips 2022. How to convert placed actors/meshes into blueprints UE5. 16.5.2022. Viitattu 26.11.2024 <https://www.youtube.com/watch?v=MSYki36PJh8>.

Adobe 2023. Bake Mesh Maps. Viitattu 10.5.2024 <https://helpx.adobe.com/fi/substance-3d-painter/using/baking.html>.

Adobe 2024. What is 3D modeling? Viitattu 20.9.2024 <https://www.adobe.com/products/substance3d/discover/what-is-3d-modeling.html>.

Akram, M. 2022. PBR Texture Maps Explained. Manson Akram 10.10.2022. Viitattu 20.9.2024 https://www.youtube.com/watch?v=6lioyniEI_E.

AskGameDev 2022. Are there any differences between an A-pose and a T-pose, or is it really just a modeler's personal preference? Viitattu 1.3.2024 <https://askgamedev.tumblr.com/post/694298319820849152/are-there-any-differences-between-an-a-pose-and-a>.

Aspland, M. 2022. How To Spawn Actor At Socket Location | Plus Attach Actor To Component Unreal Engine Tutorial. Matt Aspland 3.1.2022. Viitattu 25.11.2024 <https://www.youtube.com/watch?v=o5U7ejVp-gA>.

Autodesk 2023. 3D modeling software. Viitattu 19.2.2024 <https://www.autodesk.com/solutions/3d-modeling-software>.

Awati, R. 2024. 3D Mesh. Techtarget 1.1.2024. Viitattu 3.3.2024 <https://www.techtarget.com/whatis/definition/3D-mesh>.

Blender 2024. Structure. Viitattu 3.3.2024 <https://docs.blender.org/manual/en/latest/modeling/meshes/structure.html>.

Bretz, C. 2021. Game Character #2 - Modeling Clothes and Accessories - by Crystal Bretz. Reallusion 10.4.2021. Viitattu 20.4.2024 <https://www.youtube.com/watch?v=zLdzWZo7iv4>.

Calvello, M. 2022. What Is UV Mapping? How It Makes 3D Models Come to Life. G2 27.4.2022. Viitattu 8.5.2024 <https://www.g2.com/articles/uv-mapping#what-is-uv-mapping>.

CGDive (Blender Rigging Tuts) 2022. [Blender addon] Rig any character with the UE5 Manny/Quinn skeleton. 28.9.2022. Viitattu 15.10.2024 https://www.youtube.com/watch?v=U23WyAd8o_8&t=945s.

Diboine, A. 2014a. I'm not dead, I'm working on personal projects. Tumblr 2.4.2014. Viitattu 10.2.2024

<https://www.tumblr.com/alexandrediboine/81421345052/i-m-not-dead-i-m-working-on-personal-projects?source=share>.

– 2014b. Lil bit of rendering tonight. Tumblr 11.5.2014. Viitattu 10.2.2024.

<https://www.tumblr.com/alexandrediboine/85393973246/lil-bit-of-rendering-tonight?source=share>.

Eder, D. 2020. Guide: Sculpting Stylized Hair in ZBrush. 80.lv 11.9.2020.

Viitattu 1.4.2024 <https://80.lv/articles/guide-sculpting-stylized-hair-in-zbrush/>.

Edesberg, A. 2024. What Are 3D Textures? A Beginner's Guide. Sloyd

13.11.2024. Viitattu 15.11.2024 <https://www.sloyd.ai/blog/what-are-3d-textures-a-beginners-guide>.

Elixxier 2024. Shaders and textures in detail. Viitattu 20.10.2024

<https://www.elixier.com/en/docs/3d-importer-add-on/shaders-and-textures-in-detail/>.

Epic Games 2024. Introduction to Blueprints. Viitattu 5.11.2024

<https://dev.epicgames.com/documentation/en-us/unreal-engine/introduction-to-blueprints-visual-scripting-in-unreal-engine>.

FlippedNormals 2018. Retopology for Beginners in Maya. 29.1.2018. Viitattu

5.5.2024 <https://www.youtube.com/watch?v=xpDWta5O3n8>.

Follygon 2020. Why Blockout? 20.2.2020. Viitattu 1.3.2024

https://www.youtube.com/watch?v=F7frk_7P00o

Follygon 2022. 2D to 3D! Sculpting a Killer Pink Haired Girl From Start to Finish!

26.5.2022. Viitattu 15.3.2024

<https://www.youtube.com/watch?v=vK42CIDOu6o>.

Follygon 2023a. How I Sculpt Hyper CLEAN shapes. 28.6.2023. Viitattu

1.3.2024 <https://www.youtube.com/watch?v=Vlu70MkuTgc>

– 2023b. How I blockout Characters: Live Zbrush demo. 16.6.2023. Viitattu

2.3.2024 <https://www.youtube.com/watch?v=Ny5uvEn7fHo>.

FutureLearn 2022. What is 3D modelling and what is it used for? Viitattu

10.2.2024 <https://www.futurelearn.com/info/blog/general/what-is-3d-modelling>.

Gaarajapanime 2022. ZBRUSH Sculpting Healer | Character creation

Timelapse. 25.7.2022. Viitattu 20.8.2024

https://www.youtube.com/watch?v=AeGj_o5ojKQ&t=3495s.

Garagefarm.net 2023. Understanding Topology in 3D Modeling. Viitattu

24.11.2024 <https://garagefarm.net/blog/understanding-topology-in-3d-modeling>.

Harisova, E. 2020. Topology for Game and Game-Based Characters. 80.lv 11.7.2020. Viitattu 10.5.2024 <https://80.lv/articles/topology-for-game-and-game-based-characters>.

Hassenfratz, E. 2024. What is 3D Rigging for Animation? Schoolofmotion. Viitattu 28.10.2024 <https://www.schoolofmotion.com/blog/what-is-3d-rigging-for-animation>.

Helios 2021. Physics Asset | Tutorial. 19.4.2021. Viitattu 28.10.2024 <https://www.youtube.com/watch?v=UqJXKFIdJSM>.

James, How Do I 2022. Substance 3D Painter Textures into Unreal Engine 5 Including Emission. 19.9.2022. Viitattu 5.10.2024 https://www.youtube.com/watch?v=wcpz9_stKgs&list=PL3r_vqAz1Bw-Fea_s7QALImzMH0oPhYPI&index=40).

Just Another Dang How To Channel 2023. How To Use Sockets in Unreal Engine. Viitattu 10.11.2024 https://www.youtube.com/watch?v=r9E-ubMaELE&list=PL3r_vqAz1Bw-Fea_s7QALImzMH0oPhYPI&index=67.

Knol, C. 2017. Realistic Face male. ArtStation 13.8.2017. Viitattu 5.5.2024 <https://www.artstation.com/artwork/1VJ2K>.

Luminous Labs 2021. CONVERTING PBR Material Workflows! Diffuse to Color! Specular to Metalness! 6.6.2021. Viitattu 9.10.2024 https://www.youtube.com/watch?v=lgzAvIQxWmE&list=PL3r_vqAz1Bw-Fea_s7QALImzMH0oPhYPI&index=39.

Mangose 2023. Game-ready Character [SERANA from SKYRIM] - Part 2: Sculpting Clothes in ZBrush. 27.3.2023. Viitattu 20.8.2024 <https://www.youtube.com/watch?v=m1L92d42jIQ>.

Manim8 2022. UE5 | Human Physics Constraints Setup for Ragdoll. 1.11.2022. Viitattu 26.10.2024 <https://www.youtube.com/watch?v=KTCn7fmC6pc>.

MattLakeTA 2022. Cloth Simplified Simulation Mesh Pipeline in Unreal Engine 5 Tutorial! 28.5.2022. Viitattu 20.9.2024 https://www.youtube.com/watch?v=cdwBBs3VO0c&list=PL3r_vqAz1Bw-Fea_s7QALImzMH0oPhYPI&index=62.

Maxon Zbrush 2016. #AskZBrush: "How can I create a Custom Pop-Up Menu inside of ZBrush?" 4.3.2016. Viitattu 7.3.2024 <https://www.youtube.com/watch?v=1fOjFVbRel8>.

Maxon Zbrush 2017. The Character Art of Horizon Zero Dawn with Guerrilla Games. 7.10.2017. Viitattu 21.4.2024 <https://www.youtube.com/watch?v=WabkLpQNIyY>

Merheb, A. 2023. Retopology Best Practices for Faster Mobile Game Development. 80.lv 31.3.2023. Viitattu 10.4.2024
<https://80.lv/articles/retopology-best-practices-for-faster-mobile-game-development/>.

On Mars 3D 2020a. UV Map Anything in 5 Minutes with Maya. 12.10.2020. Viitattu 10.5.2024
https://www.youtube.com/watch?v=t5Co6SuzoQw&list=PL3r_vqAz1Bw-Fea_s7QALImzMH0oPhYPI&index=28.

– 2020b. Intro to Substance Painter - Simple Workflow for Stylized Hair! 18.11.2020. Viitattu 20.8.2024
https://www.youtube.com/watch?v=q3l5hV4ie5U&list=PL3r_vqAz1Bw-Fea_s7QALImzMH0oPhYPI&index=38.

Outgang 2020. Tips on how to sculpt a Zbrush garment sketch. 26.11.2020. Viitattu 20.4.2024 <https://www.youtube.com/watch?v=eL7r1Yav7qs>.

Parker, T. 2011. Started on my ganondorf today :) and some topology update stuff on the base mesh. Tom Parker 5.7.2011. Viitattu 5.5.2024
<https://tomparkersartdump.wordpress.com/2011/12/19/1232/>.

Pavlovich, M. 2021. Intro to Zbrush 101 – Hotkeys! Saving, storing and utilizing hotkeys to speed up your workflow! Michael Pavlovich 19.3.2021. Viitattu 7.3.2024 https://www.youtube.com/watch?v=1sYMoWz_Mk4.

Petty, J. 2024. What is Retopology? (A Complete Intro Guide For Beginners). Concept Art Empire. Viitattu 10.4.2024
<https://conceptartempire.com/retopology/>.

Render That 2024. Texture Map. Viitattu 18.11.2024
<https://www.renderthat.com/en/glossary/texture-map>.

Rocketbrush studio 2023. Texturing 3D Models in Game Art: Core Principles. Rocketbrush 15.12.2023. Viitattu 20.8.2024
<https://rocketbrush.com/blog/texturing-3d-models-in-game-art-core-principles>.

Rocketbrush studio 2024. Importance and Challenges of 3D modeling for games. Rocketbrush 26.11.2024. Viitattu 5.1.2025
<https://rocketbrush.com/blog/importance-and-challenges-of-3d-modeling-for-games>.

Room8studio 2020. The Basics of 3D Character Development for Video Games. 16.12.2020. Viitattu 20.4.2024 <https://room8studio.com/news/the-basics-of-3d-character-development-for-video-games/#modeling>.

Royal Skies 2020. Blender 2.82: Rigged Character Weight Paint Transfer (In 60 Seconds!). 27.5.2020. Viitattu 20.9.2024
https://www.youtube.com/watch?v=bR_Vke__voU.

Sergeev, A. & Sharts, Z. 2020. SKEEN Girl: 3D Stylized Character Workflow in ZBrush. 80.lv 14.4.2020. Viitattu 20.3.2024 <https://80.lv/articles/skeen-girl-3d-stylized-character-workflow-in-zbrush/>.

Stu Lloyd (CG Stu) 2024. Paint On Baked Maps To Fix Issues | Substance Painter. 1.5.2024. Viitattu 5.8.2024 https://www.youtube.com/watch?v=oRkgEuGKPtW&list=PL3r_vqAz1Bw-Fea_s7QALImzMH0oPhYPI&index=37.

The Craters 2023. How to use UDIM Materials in Unreal Engine 5 – Tutorial. 12.1.2023. Viitattu 10.10.2024 https://www.youtube.com/watch?v=SkUW4JSYrEo&list=PL3r_vqAz1Bw-Fea_s7QALImzMH0oPhYPI&index=42.

Tokarev, K. & Karimfazil, P. 2019. Game Character Production: Industry Standards. 80.lv 14.8.2019. Viitattu 20.3.2024 <https://80.lv/articles/game-character-production-industry-standards/>.

Totenn 2024. Quick tip: Decimation in Zbrush by Totenn. Artstation 30.1.2024. Viitattu 20.5.2024 https://www.artstation.com/blogs/artist_reference_3dsk/DVVwy/quick-tip-decimation-in-zbrush-by-totenn.

Unreal Engine 2019. Performance difference between material vs material instances? 31.1.2019. Viitattu 10.11.2024 <https://forums.unrealengine.com/t/performance-difference-between-material-vs-material-instances/121783>.

Unreal Engine 2022. Bug when I attach a spawned actor on socket. Viitattu 26.11.2024 <https://forums.unrealengine.com/t/bug-when-i-attach-a-spawned-actor-on-socket/734258>.

Vilkkä, H. & Airaksinen, T. 2003. Toiminnallinen opinnäytetyö. Helsinki: Tammi.

Wiki.Blender 2024. Weight Paint in a nutshell. Viitattu 20.9.2024 https://wiki.blender.jp/Doc:2.6/Manual/Modeling/Meshes/Vertex_Groups/Weight_Paint.

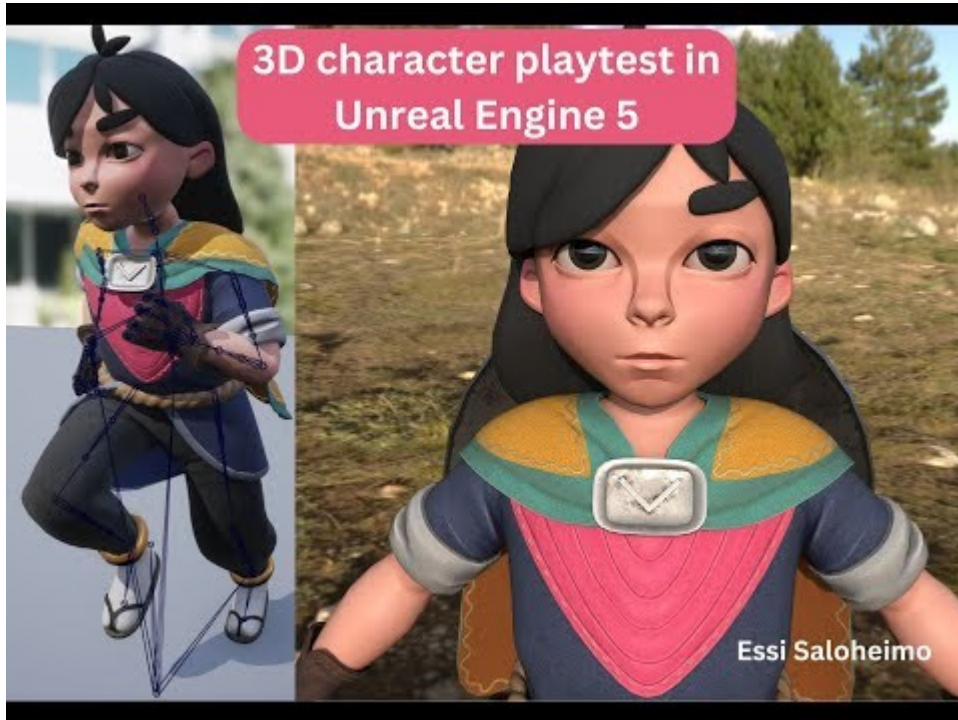
Yami 3D 2024. Weight Painting Complete Guide!! (Blender 3D). 16.2.2024. Viitattu 15.9.2024 <https://www.youtube.com/watch?v=3yrwKXQbRpl>.

LIITTEET

Liite 1. Kooste opinnäytetyön julkaisusivuista

Liite 1. Kooste opinnäytetyön julkaisusivuista

Video hahmosta, sillä pelatessa:



Saloheimo, E 2024. Thesis Character Gameplay Test (Unreal Engine). Youtube: Essi Saloheimo 13.12.2024. Viitattu 17.1.2025. <https://www.youtube.com/watch?v=EPaUxXeFj-M>

Tulen julkaisemaan työni portfolio sivuillani

Artstation: <https://www.artstation.com/essisaloheimo>

The Rookies: <https://www.therookies.co/u/essisaloheimo>

Nettisivu: <https://essisaloheimo.wixsite.com/essi-saloheimo/portfolio>