

Grafiikan muodostuminen indie-mobiilipeliprojektissa

Antti Väänänen

Opinnäytetyö

Marraskuu 2017

Luonnontieteiden ala

Tradenomi (AMK), Tietojenkäsittelyn tutkinto-ohjelma

Tekijä(t) Väänänen, Antti	Julkaisun laji Opinnäytetyö, AMK	Päivämäärä 11/2017
	Sivumäärä 39	Julkaisun kieli Suomi
		Verkojulkaisulupa myönnetty: x
Työn nimi Grafiikan muodostuminen indie-mobiilipeliprojektissa		
Tutkinto-ohjelma Tietojenkäsittelyn tutkinto-ohjelma		
Työn ohjaaja(t) Karhulahti, Mika		
Toimeksiantaja(t) Clusterloop Oy		
Tiivistelmä <p>Tutkimuksen toimeksiantaja, Clusterloop Oy, on mobiilipelimarkkinoille suuntautunut Jyväskyläläinen indie-pelifirma, joka työstää tällä hetkellä free-to-play -mobiilipeliä nimeltä "Medical Force: Overdrive!".</p> <p>Tutkimuksen tavoitteena oli selvittää, kuinka "Medical Force: Overdrive!" pelin grafiikka on muodostunut, ja miten grafiikan tuottaminen mobiilialustalle eroaa muihin alustoihin verrattuna. Tutkimus toimii Clusterloop Oy:lle projektin graafisen puolen retrospektiivina. Tutkimus toteutettiin kvalitatiivisena tutkimuksena aloittelevan peligraafikon näkökulmasta.</p> <p>Työtä varten etsittiin tietoa kirjallisista ja sähköisistä lähteistä, kuten blogeista, artikkeleista ja ohjekirjoista tukemaan laadullista puolta peligrafiikan tuottamisesta. Myös käytännön havaintoja, joita ilmeni grafiikan työstämisessä, hyödynnettiin. Kyseinen aineisto auttaa lukijaa ymmärtämään, millaista peligrafiikka voi olla ja kuinka sitä voidaan toteuttaa.</p> <p>Tuloksista käy ilmi, kuinka grafiikka muodostui projektiin sitä työstävän tiimin ja päägraafikon päätöksistä. Päätökset alkoivat vuoden alussa pelin suunnittelusta ja graafisen tyylin määrittämisestä. Itse grafiikan muokkautumiseen vaikutti pelisuunnitteluun tehdyt muutokset, jotka perustuivat suoraan eri testivaiheissa saatuihin palautteisiin. Kun peliä testattiin eri alustoilla, kävi ilmi, että grafiikkaa joutuu optimoimaan mobiilialustaa varten monella eri tavalla. Optimoimalla saadaan aikaiseksi käyttäjälle hyvä pelikokemus.</p> <p>Näiden tulosten valossa pohditaan, olisiko grafiikkaa voinut toteuttaa projektiin muulla tavalla, ja kuinka toimittaisiin uudessa vastaavanlaisessa peliprojektissa. Tutkimus on joka tapauksessa hyödyttänyt tiimiä parempien menetelmien löytämisessä.</p>		
Avainsanat (asiasanat) 3D, 2D, mobiili, indie, peli, grafiikka, Blender		
Muut tiedot (salassa pidettävät liitteet)		

Author(s) Väänänen, Antti	Type of publication Bachelor's thesis	Date 11/2017 Language of publication: Finnish
	Number of pages 39	Permission for web publication: x
Title of publication How graphics are formed in indie-mobile game project		
Degree programme Business information systems		
Supervisor(s) Karhulahti, Mika		
Assigned by Clusterloop Ltd		
Abstract <p>Clusterloop Ltd is a Jyväskylä -based indie-game developer, which is currently working on a free-to-play mobile game called "Medical Force: Overdrive!".</p> <p>The objective was to research how the game's graphics are formed and how producing graphics on a mobile platform differs from producing graphics on other gaming platforms. The research also works as a retrospect from the graphical standpoint for Clusterloop Ltd. The research was conducted as a qualitative research from a beginner artist's point of view.</p> <p>Information was gathered from written and electrical sources, consisting of blogs, articles and manuals for the research to support the qualitative side of producing game graphics. Practical observations that came up during the time of developing the graphics were also used. The material helps the reader to understand, what game graphics are and how they can be made.</p> <p>The results show how the graphics were formed by the decisions of the team and the leading artist. The early decisions were done in the beginning of the year by planning the game project and designing the art style. The alterations to the graphics came from the changes made to the game design based on the received feedback from the testing. When the game was tested on different platforms, it became clear that optimization for the graphics was necessary in many ways for the mobile platform. Optimization needs to be done to achieve a good gaming experience for the user. The results gave more perspective about using different methods in developing game graphics and if a different approach were to be taken in another similar project. The research has helped the team with finding better methods regardless.</p>		
Keywords/tags (subjects) 3D, 2D, mobile, indie, game, graphics, Blender		
Miscellaneous (Confidential information)		

Sisältö

1	Johdanto	5
2	Tutkimusasetelma	6
2.1	Toimeksiantaja	6
2.2	Tutkimusmenetelmät	7
2.3	Tutkimuksen rajaukset	7
2.4	Tutkimuskysymykset	8
3	Indie-pelit ja mobiilipelit.....	8
3.1	Indie-peli.....	8
3.2	Mobiilipelit	9
3.3	F2P.....	9
4	Peligrfiikka ja sen luomiseen liittyviä työkaluja	9
4.1	2D-grafiikka	9
4.2	3D-grafiikka ja mallintaminen	11
4.2.1	Mesh	12
4.2.2	Teksturointi.....	13
4.3	Pelimoottori.....	15
4.4	Työkalut ja ohjelmistot.....	16
4.4.1	DrawPile.....	16
4.4.2	GIMP	17
4.4.3	InkScape.....	18
4.4.4	Blender.....	19
5	Grafiikan muodostuminen peliprojektissa.....	20
5.1	Grafiikan suunnittelu.....	20
5.2	3D-mallien toteuttaminen.....	22

	2
5.2.1 Pelaajahahmon luonti.....	22
5.2.2 Potilashahmon luonti	28
5.2.3 Pelialue ja ympäristö	30
5.3 2D-objektien toteutus	32
5.4 Tulokset	36
6 Pohdinta.....	36
Lähteet	38

Kuviot

Kuvio 1. Medical force: Overdrive! -pelin viimeisin versio (Clusterloop 2017)	6
Kuvio 2. Bittikarttakuva 100 pikselin koosta suurennettuna seitsemänkertaiseksi	10
Kuvio 3. Vektorikuva 100 pikselin koosta suurennettuna seitsemänkertaiseksi	11
Kuvio 4. Kuution Mesh ja sen osat kuvattuna	12
Kuvio 5. Kuvakaappaus pelaajahahmon meshistä ja sen UV-mapista UV-editorissa ..	14
Kuvio 6. Kuvakaappaus Drawpilen perusnäkömystä	16
Kuvio 7. Kuvakaappaus GIMP:in perusnäkömystä	17
Kuvio 8. Kuvakaappaus InkScapen perusnäkömystä.....	18
Kuvio 9. Kuvakaappaus Blender 3D-editorista	19
Kuvio 10. Pelin päämekaniikan kuvaus sarjakuvana	20
Kuvio 11. Konseptikuva pelin mahdollisesta first person -kuvakulmasta.....	21
Kuvio 12. Ensimmäinen nopea konseptikuva pelaajamallista	23
Kuvio 13. Referenssikuva pelaajahahmosta.....	23
Kuvio 14. Meshin luominen referenssikuvaa vasten	24
Kuvio 15. Kuvakaappaus Medical Force: Overdrive! -pelin aikaisesta prototyypistä ..	24
Kuvio 16. Pelaajahahmon mesh	25
Kuvio 17. Pelaajahahmon tekstuuri	26
Kuvio 18. Pelaajahahmon 3D-malli tekstuurilla	26
Kuvio 19. Pelaajahahmon Rig.....	27
Kuvio 20. Dope sheet ja action editor ikkuna	28
Kuvio 21. Potilashahmon mesh referenssipiirrosta vasten.....	29
Kuvio 22. Sairaalan väriteeman suunnittelua	30
Kuvio 23. TV -objektin aikaisempi ja nykyinen versio	31
Kuvio 24. Ensimmäinen versio pelin logosta.....	32
Kuvio 25. Nykyinen versio pelin logosta (Clusterloop 2017)	33
Kuvio 26. HUD:in keskeiset ikonit	33
Kuvio 27. HUD:in keskeiset ikonit pelin pre-alpha versiossa	34
Kuvio 28. Uuden HUD:in konseptitaidetta pelin kuvakaappauksen päällä	35
Kuvio 29. Kolme kuvankaappausta pelin nykyisen version eri osista	36

Yleiset käsitteet

2D	Kaksiulotteinen
3D	Kolmiulotteinen
Shader	Valoa ja varjoja tuottava tietokoneohjelma
F2P	Free-to-play
Vertex/vertice	Koordinaatti 3D-avaruudessa
Face	Verteksien muodostama pinta
Edge	Verteksien muodostama reuna
Plugin	Ohjelmiston liitännäinen
Polygon	Verteksien muodostama alue
Texture	Kaksiulotteinen kuva, jota käytetään polygonien pintojen kuvioimiseen
UV-map	Mallin polygoneista tehty kartta tekstuureita varten
AAA-peli	Suuren tuotannon peli
Rig	Digitaalisten luiden asettaminen 3D-mallille
Pipeline	Liukuhihna grafiikan muodostumisesta alusta loppuun
Mesh	3D-objektin ulkomuoto, joka koostuu vertekseistä, reunoista ja pinnoista
Bug	Bugi, ohjelmointivirhe
Ragdoll	Fysiikkamoottorin animaatiotyyppi
UI/User interface	Käyttöliittymä

1 Johdanto

Peliala on kasvanut maailmanlaajuisesti hurjaa vauhtia viime vuosikymmenien ajat. Erityisesti mobiilipelien suuri suosio on houkuttanut paljon tekijöitä. Lähes jokaiselta ihmiseltä löytyy nykyään joko iOS- tai Android -käyttöjärjestelmällä varustettu älypuhelin, jolle voi ladata pelejä sovelluskaupasta vaivattomasti. Myös Suomessa ollaan herätty pelialan nousuun sellaisten menestyspelien kuin ”Angry Birds”, ”Clash of Clans” ja ”Hill Climb Racing” myötä. Tämä näkyy pelialaan suuntaavien opintojen lisääntymisestä ja uusien peliyritysten muodostumisesta.

Pelialalla työskentely on usein projektiluonteista ja vaihtelevaa erilaisten tiimien ja tekijöiden suhteen. Pelialalla työskentelee mm. graafikoita, ohjelmoijia, testaajia, pelisuunnittelijoita, sekä erilaisia projektien johtamiseen ja markkinointiin erikoistuneita ammattilaisia. Tämä tutkimus toteutetaan aloittelevan graafikon näkökulmasta ja sen tavoitteena on tuoda esille havaintoja ja yleisiä ongelmia tämän kaltaisen peliprojektin grafiikan työstämisessä.

Opinnäytetyön toteuttaa tutkimuskohteena olevan peliprojektin päägraafikko. Opinnäytetyössä esitetään myös tiimin harjoittelijagraafikon ja kenttäsuunnittelijan tuottamia grafiikkaobjekteja. Tämä opinnäytetyö on suunnattu nimenomaan aloitteleville graafikoille sekä tutkimuksen toimeksiantajalle Clusterloop Oy:lle, jolle tutkimus toimii retrospektiivinä peligrafiikan muodostumisesta Clusterloopin peliprojektissa.

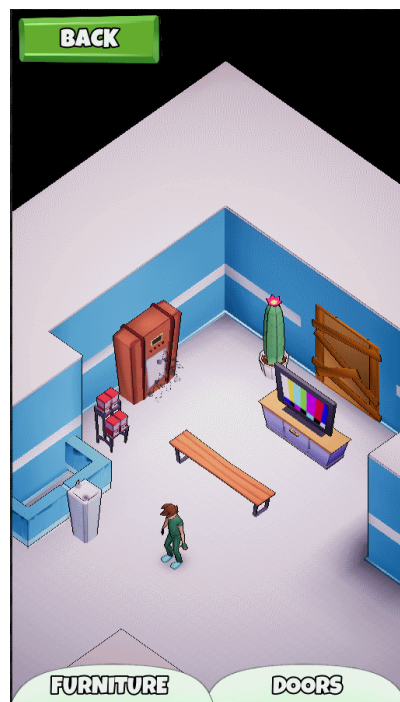
Tutkimusasetelmassa käydään läpi toimeksiantaja, tutkimuksen kohteena oleva projekti, sekä tutkimuskysymykset. Työn teoriaosuudessa pohjustetaan mitä ovat indie- ja mobiilipelit, joihinka peliprojekti lukeutuu ja minkälaista peligrafiikka projektiin työstettiin. Käytännön osuudessa käydään läpi grafiikan suunnittelua, toteuttamista, muutoksista ja syistä sen muuttumiseen. Tässä osuudessa pyritään tuomaan ilmi haasteita, joita voi esiintyä tämänkaltaisissa peliprojekteissa. Käytännön osuuden jälkeen pohditaan tuloksia ja mahdollisia vaihtoehtoisia ratkaisuja samankaltaisen projektin uudelleen työstämisessä.

2 Tutkimusasetelma

2.1 Toimeksiantaja

Tutkimuksen toimeksiantajana toimii Clusterloop Oy, jossa tutkimuksen tekijä toimii päägraafikkona tutkimuksen kohteena olevassa peliprojektissa. Tutkimus tulee hyödyttämään toimeksiantajaa kehittämällä tiimin grafiikan tuottamista. Tavoitteena on tuottaa firmalle tutkimus, joka toimii samalla retrospektiivina peliprojektin grafiikasta.

Clusterloop Oy perustettiin keväällä 2016, jolloin firmassa oli pääsuuntana lähinnä pelillistäminen suoranaisen pelin kehittämisen sijaan. Clusterloop työstää tällä hetkellä f2p, eli Free-to-play -mobiilipeliä nimeltä Medical force: Overdrive!, joka julkaistaan vuoden 2018 aikana Apple iOS ja Google Android -käyttöjärjestelmille. Medical force: Overdrive! Tuli toiseksi Assembly Summer 2017 gamedev combossa. (Clusterloop 2017.)



Kuvio 1. Medical force: Overdrive! -pelin viimeisin versio (Clusterloop 2017)

Medical force: Overdrive! on sairaala-ympäristöön sijoittuva mobiilipeli. Pelissä on humoristinen ja sarjakuvamainen ote, jossa realismi on uhrattu hauskuuden nimissä.

Pelissä pelaaja parantaa sairaita potilaita ottamalla ne kantaan ja heittämällä ne potilaiden oireita parantaviin huoneisiin, jolloin pelaaja saa pisteitä ja täten paranneltua sairaalaa, jossa hahmo työskentelee.

2.2 Tutkimusmenetelmät

Opinnäytetyössä käytetään laadullista tutkimusotetta eli kvalitatiivista tutkimusta. Laadullinen tutkimus käyttää sanoja ja lauseita, kun taas määrällinen tutkimus perustuu lukuihin. Kvalitatiivisessa tutkimuksessa ei pyritä määrällisen tutkimuksen mukaisiin yleistyksiin. Tutkimuksen kohteena on ilmiön kuvaaminen, ymmärtäminen ja mielekkään tulkinnan antaminen. Laadullisella tutkimuksella pyritään ilmiön syvälliseen ymmärtämiseen. Tarkkaa viitekehystä ei ole Kvalitatiivisen tutkimuksen toteuttamiseksi, kuten Kvantitatiivisessa tutkimuksessa. Laadullisen aineiston analyysi on syklinen prosessi, josta puuttuvat kvantitatiivisen tutkimuksen tiukat säännöt. Analyysi ei ole laadullisen tutkimuksen viimeinen vaihe, vaan koko tutkimusprosessin eri vaiheissa mukana oleva toiminta, joka ohjaa itsessään tutkimusprosessia ja tiedonkeruuta. On epäilyttävää kerätä ensin koko aineisto ja vasta tämän jälkeen aloittaa aineiston analysointi. Laadullinen tutkimus mahdollistaa tutkittavan ilmiön syvällisen ja rikkaan kuvaamisen ja selittämisen. (Kananen 2008, 24)

Laadullinen tutkimusote soveltuu hyvin kyseiseen opinnäytetyöhön, sillä opinnäytetyön tarkoituksena ei ole löytää absoluuttista totuutta tai täysin oikeaa näkökulmaa, joita peligrafiikan suhteen on hyvin monia. Sen sijaan tarkoituksena on tuoda esille havaintoja sen toteuttamisen ja iteroimisen syistä.

Opinnäytetyötä pyrittiin ohjaamaan peliprojektin graafiseen kehitykseen. Peligrafii-kasta ja sen luomisesta on olemassa hyviä opinnäytetöitä. Tässä opinnäytetyössä keskitytään enemmän niihin syihin, miksi peli juuri näyttää sen kaltaiselta kuin se kirjoitushetkellä on.

2.3 Tutkimuksen rajaukset

Peligrafiiikka, ja sen työstäminen on aiheena erittäin laaja aihealue ja sen toteuttaminen vaihtelee peliprojektien, tekijöiden sekä yleisen osaamisen mukaan. Tutkimuksia on aikaisemmin tehty peligrafiiikan luomisesta eri alustoille. Tässä opinnäytetyössä

rajataan tutkimuskohde Medical Force: Overdrive! peliprojektiin, jota työstetään pienessä indie-tiimissä mobiililaitteille.

2.4 Tutkimuskysymykset

Miten pelin grafiikka on muodostunut ja muuttunut Medical Force: Overdrive! -peliprojektissa?

Mitä mahdollisia rajoituksia mobiililaitte aiheuttaa peligrafiikan tuottamiselle verrattuna muihin pelialustoihin?

3 Indie-pelit ja mobiilipelit

3.1 Indie-peli

Indie sanalla tarkoitetaan "itsenäistä". Indie löytyy eri medioista, mm. elokuvista, musiikista sekä peleistä. Indie-peleistä voidaan siis puhua itsenäisinä peleinä. Nicholas Mamo (2015), joka on yksi Nyphoon Gamesin perustajista ja pelikehittäjästä, kertoo, kuinka Indie-pelit perinteisesti toteutetaan pienellä budjetilla. Tämä johtuu siitä, ettei kehittäjillä ole julkaisijaa tai kehitystiimi on hyvin pieni. Tästä yleensä seuraa kehittäjän kannalta kaksi asiaa: paljon riskejä, sillä kilpailu pelialalla on suurta, mutta myös enemmän vapauksia luovuuden suhteen projekteissa.

Indie-pelin vastakohtana pelimaailmassa voidaan pitää AAA-pelejä. AAA-peleille tyypillisiä piirteitä ovat suuret budjetit ja suuri tekijöiden määrä, jotka voi yleensä koostua sadoista pelinkehittäjistä ja graafikoista. (Mamo 2015.)

Leviathyn the gamer's chronicle -artikkelissa kerrotaan kuinka indie-pelit sallivat usein kokeellisempaa pelisuunnittelua, kun AAA-pelit yleensä keskittyvät jo toimiviksi todettuihin käytänteisiin. Niiden avulla pyritään toteuttamaan viihdyttävä peli. Usein ongelmana indie-pelien kehityksessä ei ole rahan puute, vaan ajan puute. Tällä saralla AAA-pelit menestyvät, kun pelin eri osa-alueita työstää erittäin suuri tiimi. Tästä syystä on hyvin todennäköistä, että peli saadaan tehtyä valmiiksi nopeammin, kuin jos vastaavanlaista peliä työstäisi vain muutama henkilö. (Joshua 2013.)

3.2 Mobiilipelit

Mobiililaitteiksi mielletään mm. tabletit, puhelimet ja jopa nykyiset kannettavat tietokoneet voidaan luokitella mobiililaitteiksi niiden pienen koon ja kevyen painon vuoksi. Mobiililaitteet eroavat mm. käsikonsoleista niiden suunnitteluperiaatteilta. Siinä missä käsikonsolit suunnitellaan pääsääntöisesti pelikonsoleiksi, niin muotoilunsa kuin käyttötarkoituksen mukaan, ovat mobiililaitteet muotoilultaan ja periaatteiltaan enemmän monikäyttötarkoituksiin suunniteltuja.

Suosittu varhainen mobiilipeli oli mobiiliversio Arcade pelistä nimeltä Snake, joka oli esiasennettuna Nokian puhelimille ja täten se löytyi yli 400 miljoonalta laitteelta. (Wright 2008.)

3.3 F2P

Medical force: Overdrive! kuuluu mobiilipeleihin, jotka luokitellaan Free-to-play (F2P) -peleiksi. Free-to-play -pelit ovat nimensä mukaisesti ilmaispelejä, joihin pelaaja voi ostaa lisäsisältöä, ne sisältävät mainoksia tai molempia. Free-to-play ei ole pelkästään sidonnainen mobiilipeleihin, vaan free-to-play -pelejä on tehty myös esimerkiksi PC:lle. Tunnettuja free-to-play -pelejä ovat mm. Team Fortress 2, Candy Crush Saga, Pokemon Go ja Warframe.

4 Peligrafiikka ja sen luomiseen liittyviä työkaluja

Tässä luvussa käsitellään lyhyesti, minkä tyyppistä grafiikkaa ja mitä ohjelmistoja sekä työkaluja käytettiin grafiikan luomiseen Medical force: Overdrive! -peliprojektissa.

4.1 2D-grafiikka

2D-grafiikaksi luokitellaan kaksiulotteisia kuvia, jotka usein koostuvat joko pikseleistä tai vektoreista. Lähes jokaisessa pelissä on mukana 2D-grafiikkaa jonkin verran, kuten esimerkiksi tekstilaatikoina, pelisessiosta kertovina mittareina, pelin valikoissa olevina nappeina tai esimerkiksi 3D-mallien tekstuureina, joista kerrotaan lisää 3D-grafiikan osiossa.

Sue Chastain kertoo ThoughtCo:n sivuilla, kuinka 2D-grafiikka jaetaan yleensä kahteen eri muotoon, vektoreihin (vectors) ja bittikarttoihin (bitmaps).

Bittikartat

Bittikarttakuvat (tunnetaan myös rasterikuvina) muodostuvat pikseleistä koordinaatistossa. Pikselit ovat kuvaelementtejä, pieniä neliöitä tai yksittäisiä värejä, jotka muodostavat kuvan joka näytetään ruudulla. Bittikarttakuvat ovat resoluutiiriippuisia. Tästä syystä bittikartta kuvia on mahdotonta kasvattaa tai pienentää ilman, että samalla uhrataan kuvan laatua. (ks. kuvio 2.) Resoluutio viittaa pikseleiden määrään kuvassa ja se yleensä kuvaillaan dpi:nä (dots per inch) tai ppi:nä (pixels per inch). Bittikarttakuvat näytetään tietokoneen näytöllä noin 100 ppi:n tarkkuudella. Yleisiä bittikarttakuva formaatteja ovat mm. PNG, GIF, JPG ja JPEG. (Chastain 2017.)



Kuvio 2. Bittikarttakuva 100 pikselin koosta suurennettuna seitsemänkertaiseksi

Vektorit

Vektorikuvat koostuvat monista yksittäisistä ja skaalattavista objekteista. Nämä objektit määritellään matemaattisilla yhtälöillä ja ne renderöidään aina parhaimmalla

laadulla. Vektoripohjaiset kuvat ovat riippumattomia resoluutiosta ja niitä voi suurentaa tai pienentää niin paljon kuin käyttäjä haluaa ja kuvan laatu pysyy samana. Vektorikuvat koostuvat usein yhden värin alueista tai gradienteista mutta ne eivät pysty kuvastamaan herkkiä värien vaihtelevuuksia, kuten valokuvissa. Tästä syystä vektorikuvat näyttävät usein pelkistetyiltä tai sarjakuvamaisilta. Siitä huolimatta, vektorigrafiikka kehittyy ja vektori piirtämisellä voidaan tehdä paljon enemmän kuin kymmenen vuotta sitten. Vektorikuvat yleensä muunnetaan niiden vektoritiedostosta bittikarttakuviksi (tätä metodia kutsutaan usein rasteroinniksi), jos kuvaa halutaan käyttää esimerkiksi webissä, tai peliprojektissa. Fontit ovat yleisimpiä vektoriohjeiteja. Yleisiä vektorikuvaformaatteja ovat mm. AI, SVG, DXF ja WMF. (Chastain 2017.)



Kuvio 3. Vektorikuva 100 pikselin koosta suurennettuna seitsemänkertaiseksi

4.2 3D-grafiikka ja mallintaminen

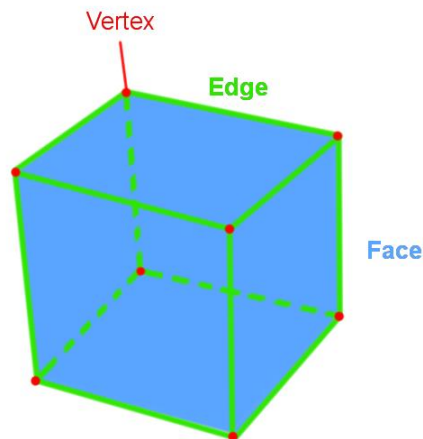
PC Magazine Encyclopedian mukaan 3D-grafiikka on objektien luomista, näyttämistä ja manipulointia tietokoneen sisällä kolmessa ulottuvuudessa. 3D-grafiikan luomiseen tarkoitettut ohjelmat sallivat objektien luonnin X-Y-Z -asteikolla (leveys, pituus,

syvyys). Objekteja voidaan käänellä ja katsoa kaikista kulmista, sekä suurentaa ja pienentää. 3D-objektit myös sallivat tuoda niihin valaistuksen renderöintivaiheessa. (PC Magazine 2017.)

Koordinaatisto (x , y ja z) on ensimmäinen asia, jota grafiikkaohjelmistossa tulisi olla ennen kuin se voi käsitellä käyttäjän antamaa dataa. Graafikoiden ei tarvitse tietää tätä tehdäkseen 3D-objekteja, vaan ohjelmissa on tämä funktio sisäänrakennettuna. Jos käyttäjä ei ymmärrä, miten tilan dataan viitataan ohjelman sisällä, se voi aiheuttaa paljon turhautumista, sillä se vaikuttaa ohjelman työkalujen käyttäytymiseen. Koordinaatisto on, miten ohjelma näkee maailman ja se täytyy ymmärtää, jotta 3D-ohjelman käyttö sujuu vaivatta. (Paquette 2013, 11.)

4.2.1 Mesh

3D-mallia kutsutaan peleissä mesheiksi, jotka ovat kokoelma verteksejä, reunoja, ja pintoja. (Blender Foundation 2017.) Kaikki pelin 3D-mallit sisältävät näitä elementtejä.



Kuvio 4. Kuution Mesh ja sen osat kuvattuna

Verteksi

Verteksi (engl. Vertex) on pienin mahdollinen osa meshistä. Verteksi on piste 3D-tilassa. (Blender Foundation 2017.)

Edge

Edge on nimensä mukaisesti reuna, joka on käytännössä viiva, joka muodostuu kahdesta verteksistä. Ne ovat yleensä näkymättömiä renderöinti vaiheessa, ja niitä käytetään pintojen rakentamiseen. (Blender Foundation 2017.)

Face

Faceja käytetään 3D-objektin pintojen luomiseen. Ne myös nähdään, kun mesh renderöidään (Blender Foundation 2017). Pinnat myös voidaan mieltää polygoneiksi.

Monet pelimoottorit ja yleiset renderit eivät renderöi polygoneja joissa on enemmän kuin neljä kulmaa. Näitä kutsutaan n-goneiksi, joka on lyhenne sanoista "n-sided-polygon". Jotkut renderit pystyvät esittämään n-goneja, mutta ne eivät välttämättä renderöidy oikein. Tästä syystä vain nelikulmaiset ja kolmiot ovat laillisia polygoneja. (Paquette 2013, 131-132.)

Tunnettuja 3D-ohjelmistoja ovat mm. Maya ja 3Ds max, jotka molemmat ovat Autodeskin omistamia sekä Blender.

4.2.2 Teksturointi

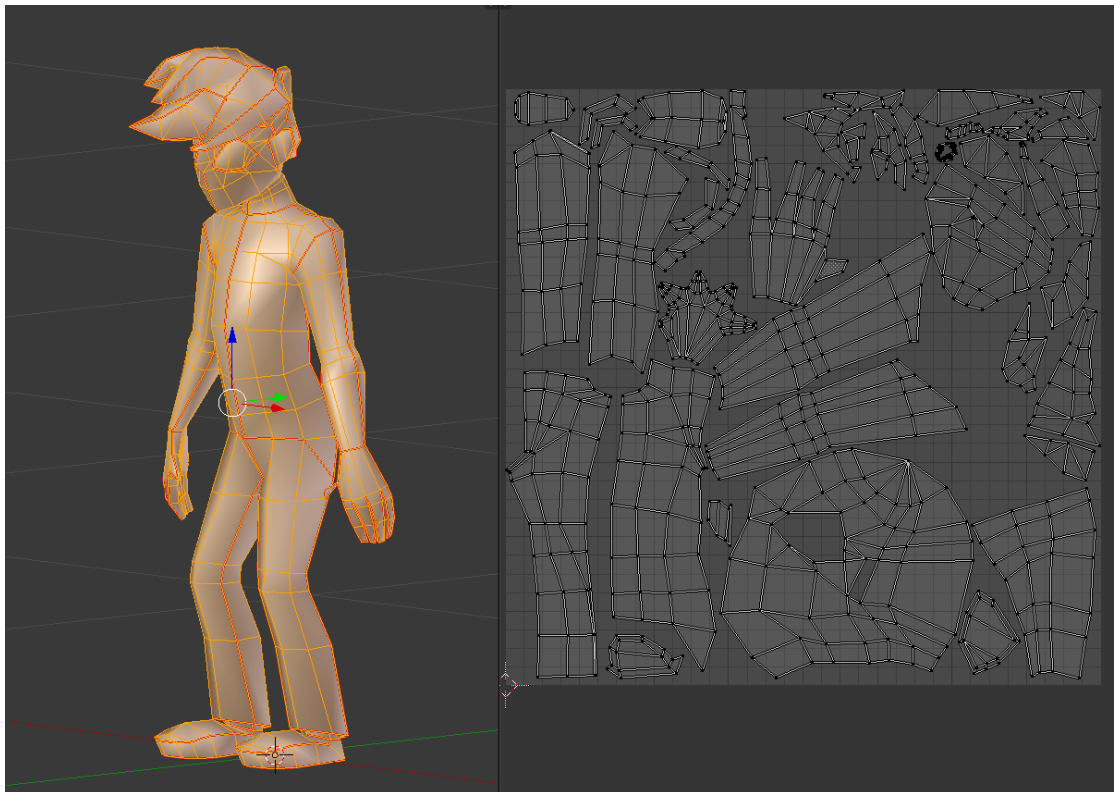
Tietokone grafiikassa tekstuurimappaus on tapa lisätä yksityiskohtia objektiin, heijastamalla sen pintaan kuvioita tai värejä. (Blender Foundation 2017.)

Tekstuurien asettaminen polygonien päälle sellaisenaan ei kuitenkaan näytä tekstuuria renderöintivaiheessa, vaan tekstuurille tulee asettaa UV koordinaatit mallista. UV:t ovat kaksiulotteisia tekstuurikoordinaatteja, jotka pohjautuvat suoraan mallin polygoneihin. (ks. kuvio 5.) UV:t siis toimivat merkintäkohtina, jotka ovat suoraan yhteyksissä mallin vertekseihin meshissä. UV mappaus tehdään, kun mallin mesh on saatu valmiiksi ja ennen kuin meshille asetetaan tekstuuri. Jos meshiä muutetaan UV mappauksen jälkeen, sen UV mappi ei enää vastaa kyseistä mallia ja täten voi näyttää siihen asetetut tekstuurit eri tavalla. (Autodesk Inc. 2017.)

Tekstuurimappien koko määritellään pikseli-per-yksikkö mitalla (engl. pixel-per-unit). Tekstuurimappien resoluutioiden pienin mahdollinen koko voi käytännössä olla 1x1,

mutta tämän kokoinen tekstuuri olisi hyödytön tekstuurimapiksi, sillä käytössä olisi vain yksi väri. 2x1 mappia voidaan pitää jo hyödyllisempänä, koska sillä voidaan määrittellä jo kaksi väriä objektille. Tämänkaltaista mappia voidaan käyttää värikanavana, tai raidallisessa vaatekappaleessa. (Paquette 2013, 94.)

Yleisimmät tekstuurikoot, joita käytetään peleissä, ovat 64 x 64, 128 x 128, 256 x 256, 512 x 512, 1024 x 1024, 2048 x 2048 ja 4096 x 4096. Lähes kaikki pelimoottorit tukevat näitä kokoja, jonka takia niitä käytetään. Nykypelit suosivatkin 1:1 kokosuhteita tekstuurien käytössä ja joissakin tapauksissa myös 2:1.



Kuvio 5. Kuvakaappaus pelaajahahmon meshistä ja sen UV-mapista UV-editorissa

Hyvin toteutetut tekstuurit käyttävät tehokkaasti UV tilaa hyödyksi. Tämä tarkoittaa sitä, että hyvin vähän UV mapin tilaa on haaskattu tyhjiin kohtiin. Tekstuurikoordinaattien järjevä pakkaus ja käyttö UV mapissa voidaan verrata esimerkiksi vaatteiden valmistukseen, jossa vaatteiden valmistajalle on tärkeää saada mahdollisimman paljon irti käyttämistä kankaan palasta. (Paquette 2013, 150.)

Kun UV-mappi on saatu valmiiksi, voidaan sen koordinaattien mukaan piirtää bittikartta pohjainen tekstuurimappi erilaisilla kuvanmuokkausohjelmilla.

4.3 Pelimoottori

Yleisesti pelimoottorin konsepti on hyvin yksinkertainen, se käsittää kaikki (joskus alustasta riippuvaisia) yleisimmät pelikohtaisiin toimintoihin liittyvät tehtävät, kuten renderöinnin, fysiikat ja syötteet, jotta kehittäjät (artistit, suunnittelijat ja ohjelmoijat) voivat keskittyä muihin yksityiskohtiin, jotka tekevät heidän peleistään uniikkeja.

Pelimoottorit tarjoavat uudelleenkäytettäviä komponentteja, joita manipuloimalla voidaan tuoda peli eloon. Graafiset käyttöliittymät, fysiikat, törmäyksen mallinnus objektien kanssa, mallien animointi, latautumiset ja näyttämiset sekä osat pelin tekoälystä voivat olla komponentteja, jotka muodostavat pelimoottorin. Näihin verrattuna pelin sisältö, kuten tietyt mallit, tekstuurit, pelin objektien törmäysten takana oleva tarkoitus ja tapa, jolla objektit ovat vuorovaikutuksessa keskenään pelimaailmassa kuuluvat komponentteihin, jotka muodostavat itse pelin. (Ward 2008.)

Tunnettuja pelimoottoreita ovat nykyään mm. Game Maker, Unity 3D, sekä Unreal Engine 4. Valitsimme pelimoottoriksi Unreal Engine 4:n, jonka pohjalta aloimme rakentaa peliämme tiimin ohjelmoijan suosituksesta.

Medical force: Overdrive! -peliprojektin pelimoottorina toimi Unreal Engine 4, joka valittiin projektin pääkoodaajan toimesta.

Unreal Engine 4

Unreal engine on Epic Gamesin kehittämä pelimoottori, jota käytettiin ensimmäisen kerran vuonna 1998 Epic Gamesin pelissä "Unreal". Unreal Enginen neljäs versio ilmestyi vuonna 2014.

Näkyvimpänä uudistuksena Unreal Engine 4:ssa voidaan pitää uutta "blueprint" ominaisuutta, jonka avulla voidaan luoda skriptejä visuaalisesti. Blueprint on nimenomaan suunniteltu graafikoita ja pelisuunnittelijoita varten. Unreal Engine 4 on nykyään suuressa suosiossa suuremmissa peliyrityksissä.

Gamasutran julkaisemassa artikkelissa, jossa Tim Sweeney (2014) Unreal Engine 4:n uusista tekniikoista, kuten unrealin fysiikkapohjaisesta materiaalista- ja varjostusjärjestelmästä. Nämä mahdollistavat työkalut, joilla voidaan luoda ultrarealistisia pin-

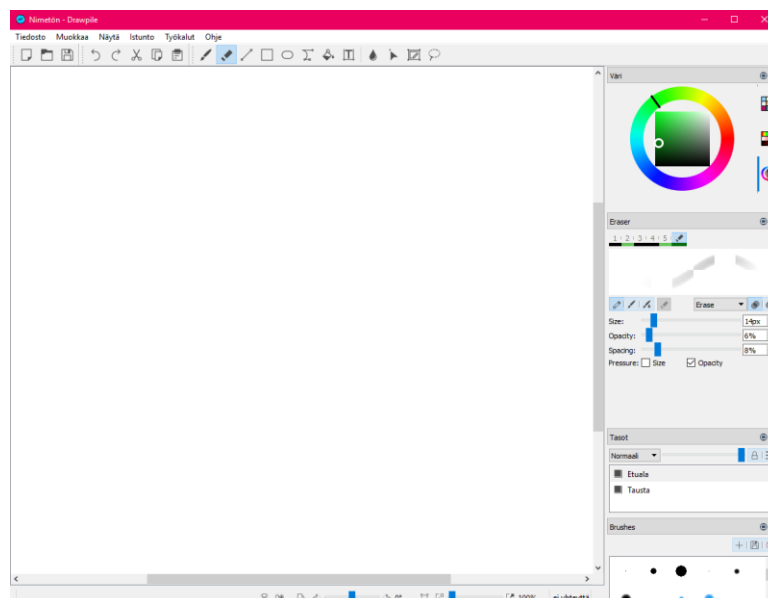
toja, jotka näyttävät hienoilta ja antavat mahdollisuuden lähes rajattomiin pintayksityiskohtiin. Uusi materiaalisysteemi antaa artistien maalata erilaisia kerroksia toisensa päälle, kuten esimerkiksi ruostetta metallimateriaalin päälle. (Sweeney 2014.)

4.4 Työkalut ja ohjelmistot

Graafikolle on tarjolla paljon eri ohjelmia ja työkaluja erilaisen grafiikan työstämiseen. Mikäli projektiin halutaan tuottaa grafiikkaa, joka on tasoltaan kilpailukykyinen suosittujen pelien rinnalla, on suositeltavaa, että graafikolta löytyy laitteisto, joka on riittävän tehokas mahdollisimman monipuolisen grafiikan tuottamiseen. erilaiset kuvanmuokkausohjelmat ja 3D-editorit vaativat laitteistoltaan rutkasti muistia, toimiakseen sulavasti.

4.4.1 DrawPile

DrawPile on suomalaisen Calle Laakkosen luoma, ilmainen piirto-ohjelma, joka mahdollistaa piirtäjien voida piirtää samalle kanvasille verkko- tai internet-yhteyden yli. Ohjelmalla on myös mahdollisuus luoda pieniä animaatioita. Drawpile tukee Open raster (ora) -tiedostoformaattia, joka on yhteensopiva mm. Kritan, GIMP:in ja MyPaintin kanssa. (Laakkonen 2014.)

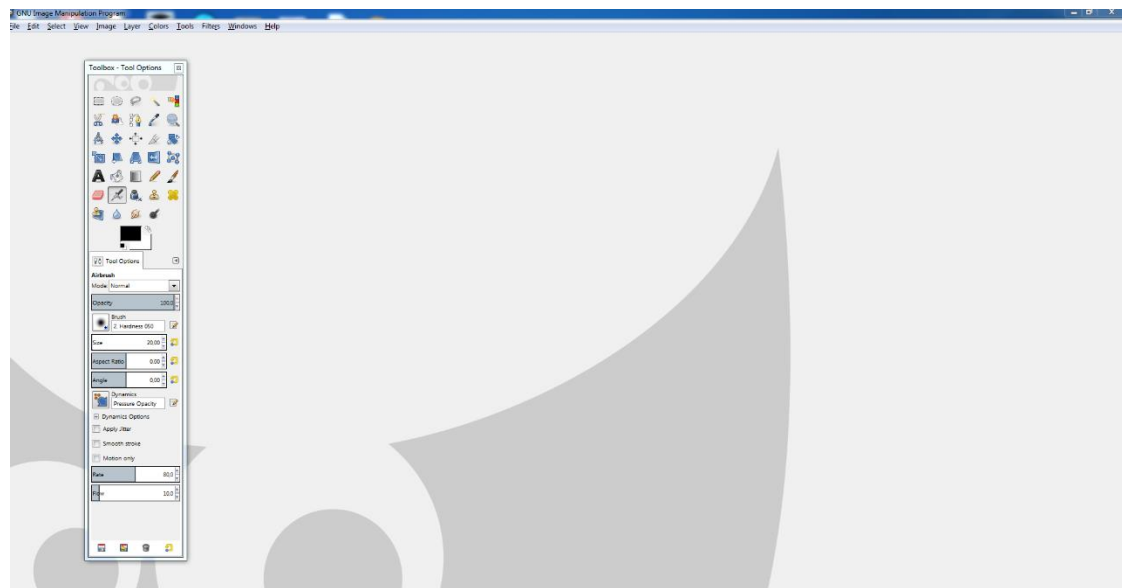


Kuvio 6. Kuvakaappaus Drawpilen perusnäelmästä

Drawpile on kätevä ohjelma luonnosteluun toisen henkilön kanssa samalle kanvasille. Ohjelmasta löytyvät työkalut eivät ole niin monipuolisia kuin useimmissa piirto-ohjelmissa ja ohjelmalla ei pysty suorittamaan erityisemmin kuvanmuokkausta, mutta perinteiset piirto ja valinta sekä muokkaustyökalut löytyvät. Taitava ja kokenut graafikko voi saada paljon irti ohjelman rajoitteista. Graafikko käytti ohjelmaa pikaisien mallien tekstuuriin luomiseen.

4.4.2 GIMP

GIMP on lyhenne sanoista GNU Image Manipulation Program. Kuten nimestä voi päätellä, GIMP on ilmainen kuvanmuokkaus ohjelma. Se on GNU-lisenssin alainen ohjelma, joka antaa käyttäjälle täydet oikeudet ohjelman koodin muokkaamiseen ja uuden koodin julkaisuun. GIMP:iin on saatavilla myös monia liitännäisiä, joiden avulla käyttäjä voi muokata ohjelmaa haluamukseen. GIMP on saatavissa Linux- OS X- ja Windows-käyttäjärjestelmille. (The GIMP Team 2017.)

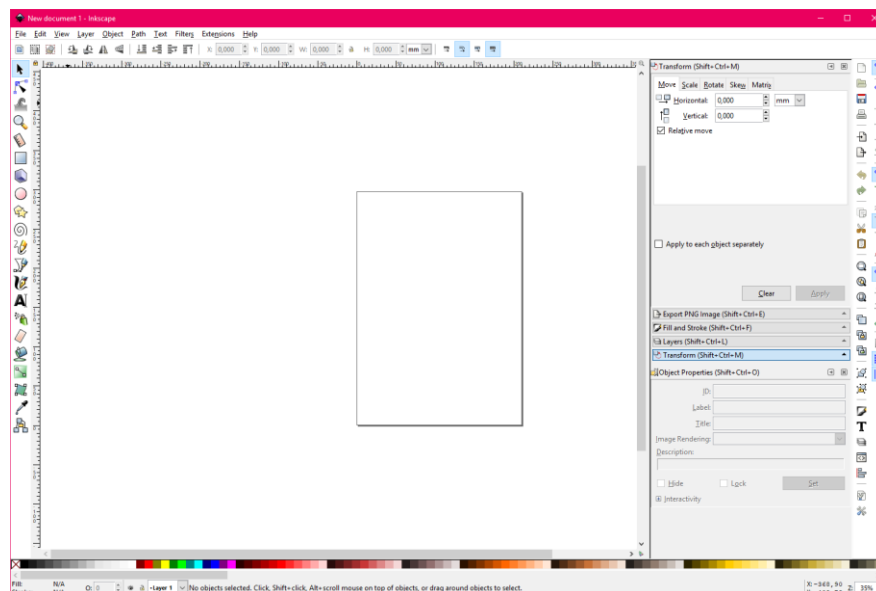


Kuvio 7. Kuvakaappaus GIMP:in perusnäkökymästä

GIMP:ä käytettiin tekstuuriin luomisessa ja muokkaamisessa. GIMP:in käyttöliittymä poikkeaa perinteisistä kuvanmuokkausohjelmista ja mahdollisesti vaatii paljon totuttelua käyttäjältään, joka on tottunut esimerkiksi Adobe Photoshopin käyttöliittymään.

4.4.3 Inkscape

Inkscape on ammattilaistasoinen vektori grafiikan luomiseen tarkoitettu ohjelma, joka on yhteensopiva Windows- Max OS X- ja Linux-käyttöjärjestelmien kanssa. Sitä käyttävät sekä ammattilaiset että harrastelijat maailmanlaajuisesti luodakseen erilaisia graafisia objekteja ja kuvia kuten ikoneita, logoja, diagrammeja, karttoja ja webbi grafiikkaa. Inkscape ilmainen ja avoimen lähdekoodin sovellus. Se on verrattavissa muihin vektori grafiikka ohjelmiin, tunnetuimpana esimerkiksi Adobe Illustrator. (The Inkscape Project 2017.)



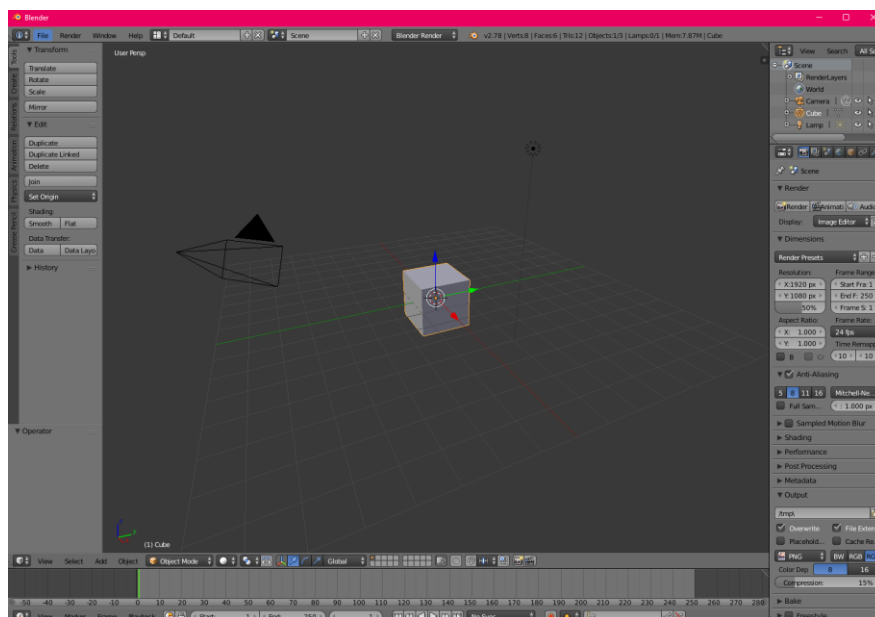
Kuvio 8. Kuvakaappaus Inksapen perusnäkyästä

Inkscapea käytettiin juuri yllämainituista syistä projektissa ja sillä luotiin pelin UI-elementit, kuten napit, menu-ikonit sekä pelin logo. Vektorigrafiikan ehdottomasti paras ja käytännöllisin puoli on sen skaalattavuus. Vektoreilla toteutetun grafiikan laatu ei muutu, jos sen kokoa muuttaa. Tämän ansiosta pelin vektoreilla luotuja objekteja voidaan uudelleen käyttää myös mm. markkinointimateriaaliin.

4.4.4 Blender

Blender on hollantilaisen Blender Foundationin kehittämä, ilmaiseen jakeluun julkaistu 3D-työstämiseen ja editointiin tarkoitettu ohjelma, joka tukee täysin 3D-tuotannon liukuhihnaa, johonka sisältyy mallinnusta, riggausta, animointia, liiketunnistusta sekä video editointia ja pelin tekoa. Kokeneemmat käyttäjät pystyvät Python-ohjelmointikielen avulla ohjelmoimaan ja muokkaamaan omia työkaluja Blenderin ohjelmistorajapinnassa.

Blender on GNU General Public -lisenssin alaisena, joka antaa oikeuden käyttää, kopioida, muuttaa ja jakaa ohjelmia ja näiden lähdekoodia. Blenderiä kehitetään maailmanlaajuisesti niin ammattilaisten kuin harrastelijoiden toimesta. (Blender Foundation 2017.)



Kuvio 9. Kuvakaappaus Blender 3D-editorista

Blender valittiin käytettäväksi projektiin, sillä se oli kaikista 3D-mallinnusohjelmista tutuin tiimin jäsenille ja samalla myös kaikista kattavin ja helposti saatavilla oleva 3D-editointi ohjelmisto jolla pystyttiin toteuttamaan pelin 3D-objektit. Blenderistä on tarjolla paljon oppaita video ja tekstimuodossa ja ne ovat helposti saatavilla internetistä.

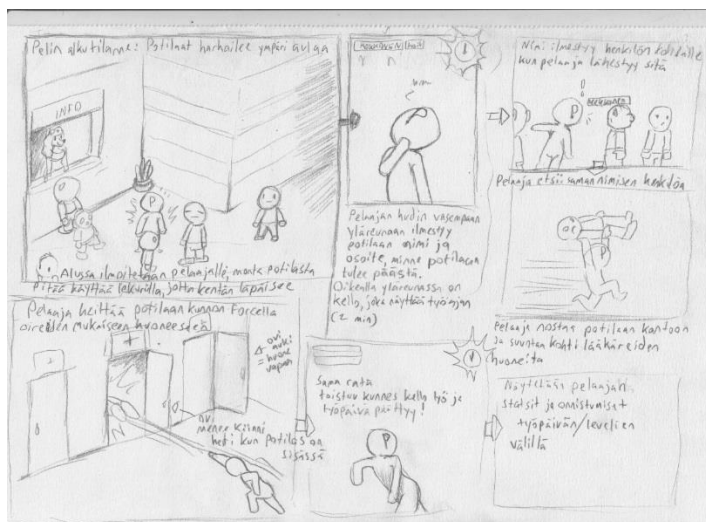
5 Grafiikan muodostuminen peliprojektissa

5.1 Grafiikan suunnittelu

Peliprojektin suunnittelu aloitettiin vuoden 2017 alussa. Koko peliprojekti perustui Clusterloop firman edelliseen prototyyppiin, joka oli eräänlainen sairaussimulaattori nimeltään Sickness simulator. Sairaussimulaattorin prototyyppiä testautettiin ja näytettiin eri pelialan ammattilaisille Pocket gamer ja Slush tapahtumassa vuonna 2016. Kokeneilta pelinkehittäjiltä saatiin paljon hyvää ja rakentavaa palautetta. Tähän palautteeseen perustuen päätettiin siirtää peli sairaalaan ja tehdä siihen yksi hauska pelimekaniikka, jonka ympärillä peli pyörisi. Suunnittelua käytiin läpi niin sanallisesti, kuin visuaalisesti konseptikuvien avulla pelin mahdollisista hahmoista ja mekaniikoista. Konseptikuvitus toteutettiin projektin aikana pääsääntöisesti kynällä ja paperilla, mutta välillä myös digitaalisesti.

Pelimekaniikan kuvaus

Pelimekaniikan kuvaamista varten tiimin graafikko loi siitä pienen sarjakuvan, millainen pelisessio tulisi olemaan pelin aikana. (ks. kuvio 10.) Sarjakuvassa kuvattiin keskeiset asiat, eli miltä pelisessio voisi näyttää heti alussa, miltä pelimekaniikka näyttäisi toiminnassa ja mitenkä pelaajan odotettaisiin etenevän. Tällaisen konseptipiirroksen on tarkoitus lähinnä antaa suuntaa pelin kehittämiselle ja tuoda sama visio koko tiimille.



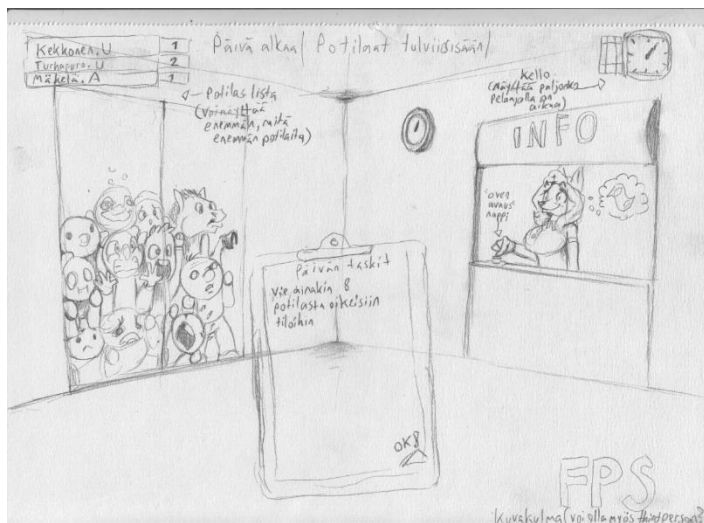
Kuvio 10. Pelin päämekaniikan kuvaus sarjakuvana

Pelin graafista tyyliä haluttiin viedä enemmän sarjakuvamaiseen suuntaan. Koska pelin päämekaniikan kanssa haluttiin ottaa riittävästi vapauksia ja keskittyä hauskuuteen, sen haluttiin näkyvän myös pelin graafisessa tyyliässä. Inspiraatiota haettiin lukuisista ns. lauantai aamupiiretyistä, kuten "Aanimaaniset", "Väiski vemmelsääri" ja "Scooby Doo". Tätä varten pelin grafiikan parissa työskentelevät kasasivat moodboardin. Moodboard on käytännössä kollaasi, johonka on kiinnitetty halutun teeman mukaisia kuvia, jolla artistit pystyvät kommunikoimaan pelin visuaalista ilmettä, tai asettamaan pääinspiraatioksi graafisen tyyliä. Tämä auttaa graafisen tyylin määrittelyssä.

On usein suotavaa, että graafikko osaisi ennestään piirtää tai luoda realistisia kuvia, ennen kuin luodaan ns. tyylikkään näköistä grafiikkaa. Tämä tuo varmempaa tietoa graafikolle siitä, miksi jokin objekti näyttää siltä kuin se näyttää oikeassa elämässä ja täten ymmärtää, objektin muodon rakenteet ja niiden tarkoitukset. Tämä tieto auttaa huomattavasti visuaalien tyyllittelemisessä ja se on nopeasti huomattavissa.

Kuvakulman määrittäminen

Se mistä kuvakulmasta peli näytetään pelaajalle, on suoraan yhteydessä kuinka tarkkaa ja raskasta grafiikkaa joudutaan tuottamaan. Aluksi kaavailtiin kuvakulmaksi "first person", eli ensimmäisen persoonan kuvakulmaa (ks. kuvio 11). Konseptikuvassa ollaan jo hieman käyty läpi peliin tulevia HUD-elementtejä, kuten kello ja potilaslista.



Kuvio 11. Konseptikuva pelin mahdollisesta first person -kuvakulmasta

Tämä johtui osittain siitä, että Sickness simulator -projekti oli myös kyseisestä kuvakulmasta, mutta tämä muutettiin hyvin nopeasti ”third person”, eli kolmannen persoonan kuvakulmaksi. Third person -kuvakulmassa pelikuva on usein pelihahmon takaa. Tämän kuvakulman mukaan lähdettiin suunnittelemaan pelin päähahmoja. Tämä ei kuitenkaan osoittautunut lopulliseksi vaihtoehdoksi, vaikka pelimekaniikka oli jo suunniteltu. Peliä täytyi testauttaa moneen kertaan, että voitiin arvioida pelin kannalta toimivin ratkaisu kuvakulman suhteen. Tähän vaikutti hyvin paljon pelin kontrollit ja ohjattavuus, joka oli vaikea saada toimimaan kolmannen persoonan kuvakulmasta.

5.2 3D-mallien toteutus

Projektin alussa määriteltiin keskeisimmät 3D-mallit, tässä tapauksessa sairaalaympäristö, sen sisältämät objektit ja hahmot. Sairaala työsti projektin kenttäsuunnittelija, kun taas päägraafikko paneutui hahmojen luomiseen, mallintamiseen ja animoimiseen.

Pelin hahmojen suhteen haluttiin hullunkurista ja hyvin laajaa lähestymistä. Hahmo-kaarti koostuisi mm. Ihmisistä, antropomorfisista eläinhahmoista ja roboteista. Pelin vakio pelaajahahmoksi ja vakio potilashahmoksi päätettiin kuitenkin ihmishahmot, jotka ovat keskeisimmät hahmot joita pelissä näkee.

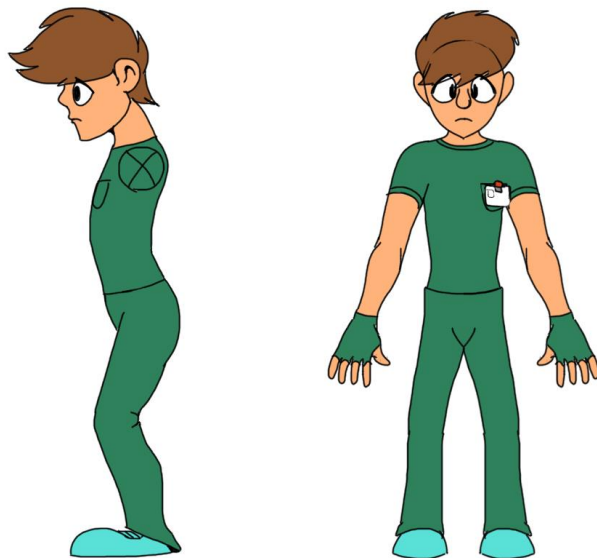
5.2.1 Pelaajahahmon luonti

Pelaajahahmosta kaavailtiin neutraalia ja helposti lähestyttävää hahmoa. Täten pelaajahahmon olemus pyrittiin mallintamaan sellaiseksi, johonka pelaajan oli helppo asettamaan oma identiteettinsä. Hahmon haluttiin olevan nuori aikuinen, mahdollisesti sairaalassa vasta-alkanut harjoittelija, joka sattuu olemaan luonnollinen lahjakkuus työssään.

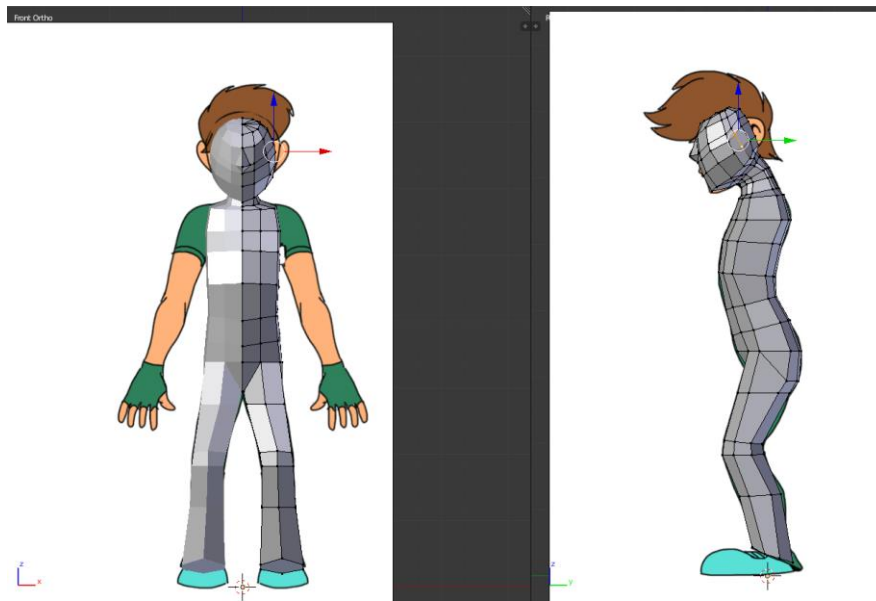


Kuvio 12. Ensimmäinen nopea konseptikuva pelaajamallista

Pelaajahahmoa jalostettiin hieman pidemmälle ja hahmosta piirrettiin referenssikuva, jota voitiin käyttää 3D-mallin rakentamisessa. Päägraafikko koki tämän huomattavasti nopeammaksi, kuin mallin luominen ilman referenssikuvaa. Usein ilman referenssikuvaa mallintaminen voi turhauttaa ja viedä liikaa aikaa merkityksettömien yksityiskohtien hiomiseen, tai hahmon anatomian korjailuun.



Kuvio 13. Referenssikuva pelaajahahmosta



Kuvio 14. Meshin luominen referenssikuvaa vasten

Peli käytti third person -kuvakulmaa aikaisissa prototyyppi versioissa ja täten pelaajahahmo oli jatkuvasti näkyvässä oleva 3D-objekti, sekä lähellä kameraa (ks. kuvio 15). Tästä syystä pelaajahahmolle annettiin suurin polybudjetti (engl. Polybudget). Projektin alussa oltiin otettu jo huomioon, että polygonimäärät malleilta kannattaa pitää hyvin matalina, jos halutaan saada peliin aikaiseksi sulava ruudunpäivitys. Tämä on tärkeää pelin ollessaan kolmiulotteinen ja mobiilialustoille suunnattu.



Kuvio 15. Kuvakaappaus Medical Force: Overdrive! -pelin aikaisesta prototyypistä

Unity -dokumentaatiossa suositellaan polygonimääräksi pelihahmoille yleensä 300-1500 polygonin väliltä, kun on kyse mobiilialustasta. Polygonien määrää voi joutua laskemaan, jos hahmoja tulee näkymään ruudulla paljon. (Unity Technologies 2017.)

Mobiililaitteiden ollessa pienikokoiset, niihin ei mahdu yhtä tehokkaita prosessoreita tai näytönohjaimia, kuten esimerkiksi nykyisiin tietokoneisiin tai pelikonsoleihin. Tämä on selkeästi nähtävissä, kun vertailee suosittujen mobiilipelien ja konsolipelien graafista tasoa keskenään.



Kuvio 16. Pelaajahahmon mesh

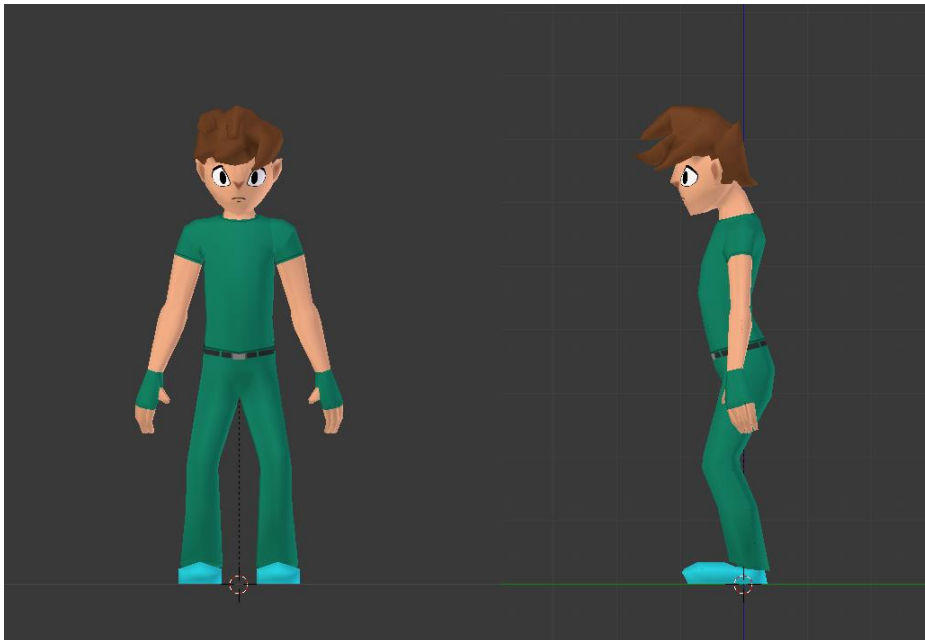
Pelaajahahmolle annettiin tässä tapauksessa polygonien maksimirajaksi noin 1500 polygonia. Pelaajahahmon polygonien määrä on noin 900, joka pysyy Unityn ehdottamien määreiden sisällä.

Kun hahmon mesh oli hahmolla valmis, luotiin hahmon meshistä UV-kartta.

Tekstuurin resoluutioksi määriteltiin 1024 x 1024, mikä katsottiin riittäväksi third person kuvakulmaan. Tekstuuri toteutettiin suurimmaksi osaksi itse Blenderin teksturointi työkalulla, sekä tehden pieniä muokkauksia, käyttäen DrawPile -kuvanmuokaus ohjelmistioa. Blenderin teksturointityökalu osoittautui erittäin käteväksi, sillä sen avulla voi nopeasti maalata haluamansa alueet mallista suoraan 3D näkymässä.



Kuvio 17. Pelaajahahmon tekstuuri



Kuvio 18. Pelaajahahmon 3D-malli tekstuurilla

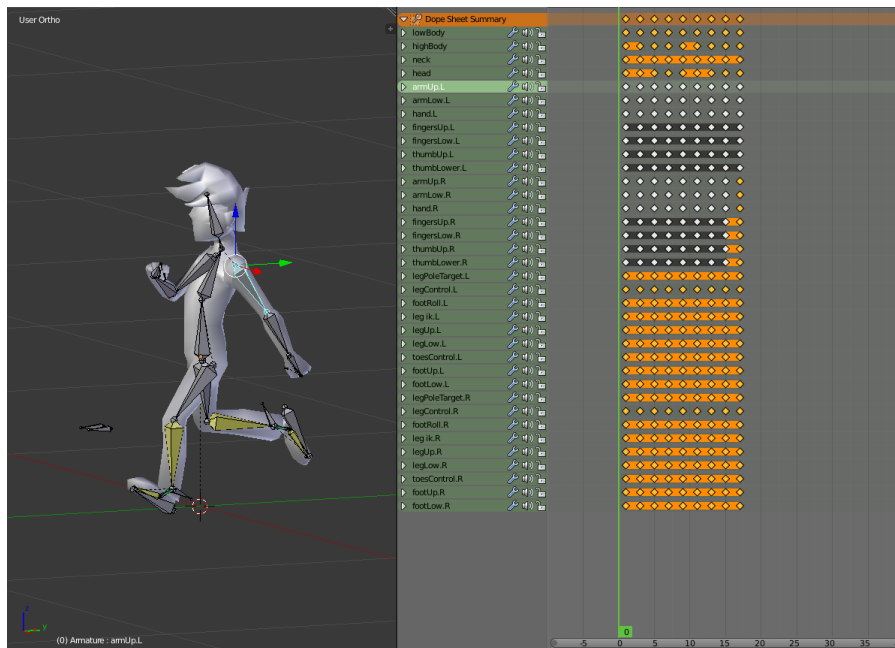
Kun mallin mesh ja tekstuuri oltiin saatu valmiiksi, oli vuorossa hahmon luuston rakentaminen animointia varten. Tätä kutsutaan riggaamiseksi. Riggauksessa luodaan 3D-mallille digitaalinen luusto, joka voi koostua yhdestä, tai monista luista ja nivelistä. Kyseinen mallin luonut graafikko ei ollut aikaisemmin rigannut tai animoinut 3D-malleja, ja katsottiin tarpeelliseksi graafikon opetella kyseiset taidot, jotta mallien luomisen liukuhihna (engl. pipeline) olisi hallussa mahdollisimman ajoissa.



Kuvio 19. Pelaajahahmon Rig

Pelaajahahmon ja luustosta haluttiin tehdä mahdollisimman yksinkertainen, jolla silti voitaisiin suorittaa ja animoida kaikki tarpeelliset liikkeet. Keskeisimmät animaatiot pelaajahahmolle tulisi olemaan Idle (paikoillaan seisoskelu), kävely, juoksu, nosto, heitto, sekä kantamisanimaatio. Animaatiot toteutettiin blenderin sisäisellä Dope sheet ja action editoreilla.

Dope sheet (ks. kuvio 20) on Blenderin käyttöohjeen mukaan digitaalinen kaavio, jossa voidaan tarkastella milloinka kuva, ääni tai kameraliike tapahtuu ja mihinkä aikaan (Blender Foundation 2017). Action editorilla määritetään ja kontrolloidaan haluttua toimintoa dope sheetissa.



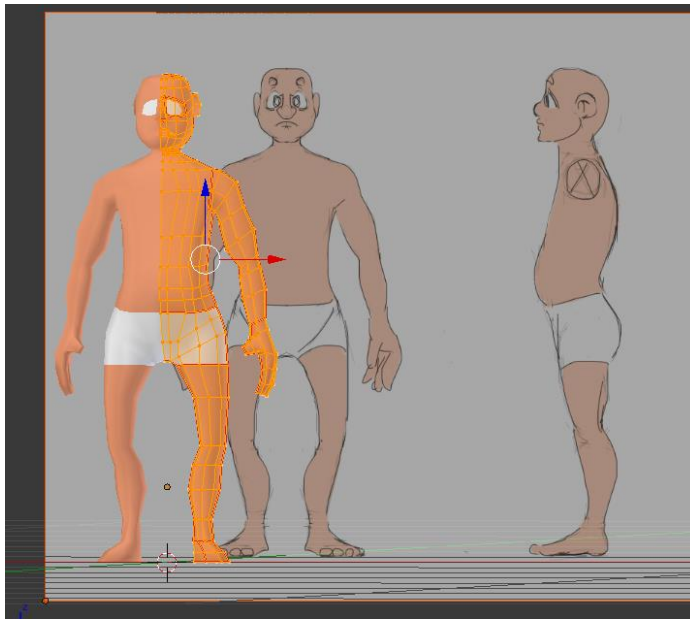
Kuvio 20. Dope sheet ja action editor ikkuna

Pelaajahahmoa tullaan todennäköisesti optimoimaan ja iteroimaan jatkossa monelta osin, kuten turhien polygonien vähentämisessä, tekstuurin ja pelaajahahmon luurangan muokkaamisessa. Pelin kuvakulma on kirjoitushetkellä todettu toimivan parhaiten kaukaa isometrisestä kuvakulmasta, jolloin pelin hahmot ovat hyvin pieniä. Tästä syystä pieniä yksityiskohtia, kuten yksittäisiä sormia ei hahmossa näe. Tästä syystä ei ole järkeä jättää hahmolle isokokoista tekstuuria, tai yksityiskohtaista meshiä, jos yksityiskohtia ei tule näkemään, koska se vain kuormittaa laitteistoa turhaan ladatesaan näitä yksityiskohtia.

5.2.2 Potilashahmon luonti

Unreal Engine 4:sta löytyy jo valmis perushahmo, jolla on muutamia perusanimaatioita, kuten käveleminen, juokseminen, hyppiminen jne. Tätä hahmoa käytettiin potilashahmoihin projektin alussa, kun luotiin ensimmäiset prototyypit pelistä. Näissä hahmoissa oli suuri polygoni määrä ja katsottiin tarpeen luoda huomattavasti pienemmällä polygon määrällä olevat hahmot, jottei hahmojen suuri polygonmäärä veisi ruudunpäivitystä alas, kun peliä testattiin mobiililaitteilla.

Potilashahmon luomisprosessi oli lähes sama kuin pelaajahahmolla. Potilaan ulko-muoto pidettiin myös jokseenkin neutraalina. Potilaasta haluttiin tehdä ns. ”modulaarinen”, eliikä tiettyjä osia, kuten käsiä, jalkoja, torsoja, päitä, vaatekappaleita ja hiuksia pystyttiin vaihtaa ja lisäilemään. Vaatekappaleet jätettiin valkoiseksi eikä teksturoitu. Koodin avulla pystyttiin värjäämään vaatekappaleet satunnaisilla väreillä valkoiseksi jätetyt kohdat.

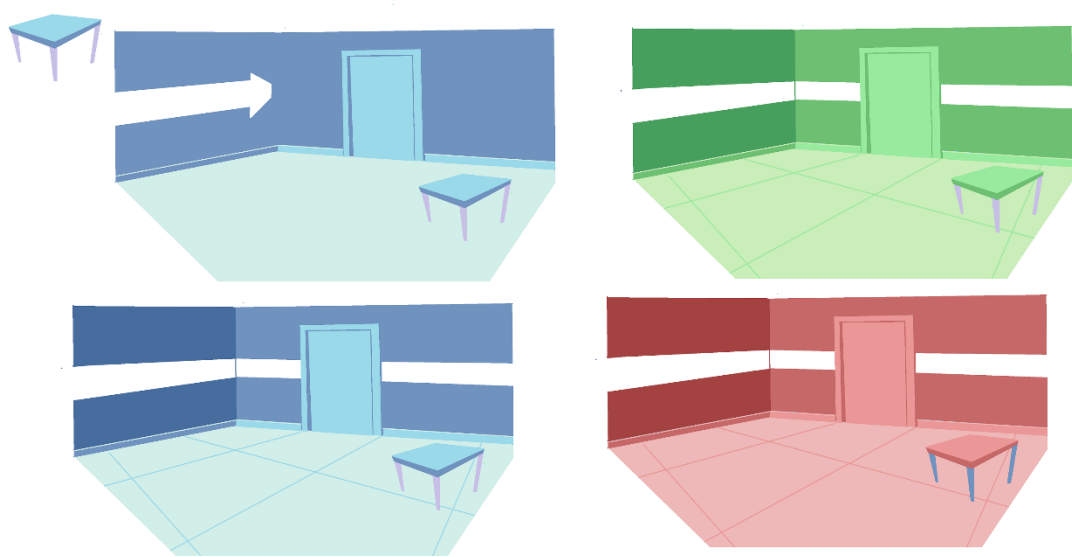


Kuvio 21. Potilashahmon mesh referenssipiirrosta vasten

Potilashahmo oltiin alusta alkaen suunniteltu ragdollaamaan, kun potilaan heittää tai kaataa. Potilashahmo tämän toiminnon tapahtuessa muuttuu ”sätkynukeksi” pelin fysiikkamoottorin rajoitteissa. Tällä haluttiin tuoda pelille viihdearvoa ja korostamaan pelin sarjakuvamaista maailmaa. Potilashahmon rig oli muokattu versio pelaajahahmon rigistä. Potilashahmon rigistä poistettiin väliaikaisesti kontrolliluut, koska niiden aiheuttamista bugeista, joita ilmeni ragdoll -ominaisuuden ohjelmoimisessa. Tämä teki potilashahmon animoimisesta hankalaa, sillä kontrolliluut auttavat realistisen liikkeen luomisessa. Potilashahmolle tehtiin kävely, juoksu, seisoskelu ja istumis animaatiot, sekä erilaisia tautikohtaisia animaatioita elävöittämään pelimaailmaa, kuten aivastelua, yskäisyä, pään pitelemistä ja ontumista jne.

5.2.3 Pelialue ja ympäristö

Koska pelin pääympäristönä toimi sairaala, otettiin peliin inspiraatiota sekä oikeista, että sarjakuvamaisista sairaalaympäristöistä. Väriteemaa määriteltiin erilaisia esimerkkejä sairaalalle. Sairaalaan ulkonäköä ajateltiin aluksi hyvin pelkistetyn värisenä pohjana, jonne myöhemmin asetetaan sairaalaympäristöön liittyviä huonekaluja ja esineitä, jotka tuovat sairaalaan ympäristöön lisää eloa ja uskottavuutta.



Kuvio 22. Sairaalan väriteeman suunnittelua

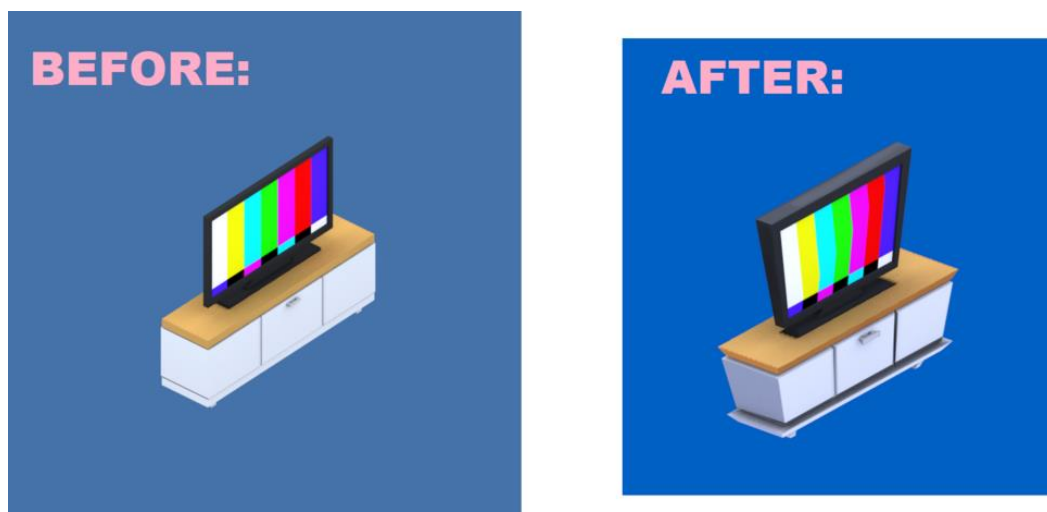
Mikael Paananen toimi projektin kenttäsuunnittelijana ja tuotti sairaalaympäristöön suunniteltuja objekteja. Sairaalaympäristöön kuuluvat mallit pohjautuivat vahvasti oikean elämän huonekaluihin tai esineisiin, joista otettuja kuvia käytettiin referenssimateriaaleina mallien luomista varten.

-kuva-

Sairaalamallien polygonimäärät olivat jo ennestään suhteellisen pieniä ja niitä pienennettiin lisää optimointia varten, kun pelin kuvakulma vaihdettiin kauemmaksi.

Projektin alussa sairaalaympäristö muodostui modulaarisista paloista, jotka mallinnettiin Blenderissä. Näillä kenttäsuunnittelija pystyi työstämään erilaisia sairaalatasoja nopeasti ja vaivattomasti. Modulaarisista seinäpaloista koitui kuitenkin näkyviä varjostusongelmia. Tämän seurauksena päätettiin väliaikaiseksi ratkaisuksi yhdistää modulaarisista paloista rakennetun sairaalan yhdeksi meshiksi.

Vaikka peli oli luonteeltaan sarjakuvamainen ja idealtaan hullunkurinen, haluttiin pelin opetuksellinen puoli olla tasapainossa pelin hupipuolen kanssa, jolla arvioitiin saavan tärkeää syvyyttä peliin. Tämä oli yksi merkittävimmistä syistä siihen, miksi pelimaailman objektit olivat alkuunsa realistisemman näköisiä. Myöhemmin keskityttiin enemmän huumoriin, sillä se vetosi testipelaajiin enemmän. Tätä haluttiin myös pelimaailman kuvastavan, jolloin maailman objekteista muokattiin myös enemmän sarjakuvamaisemman näköisiä. (ks. kuvio 23.)



Kuvio 23. TV -objektin aikaisempi ja nykyinen versio

Tällä hetkellä kentämalleista työstetään enemmän sarjakuvamaisemman näköisiä versioita, jotta pelimaailman kokonaisuus näyttää teemallisesti yhtenäisemmältä. Myös mallien värimaailmaa tullaan uusimaan pelin uuteen väripalettiin sopivammaksi. Tässä tapauksessa kyse on sekä iteroimisesta että optimoimisesta.

5.3 2D-objektien toteutus

Pelissä 2D-grafiikaksi luokitellaan kaikki kaksiulotteiset assetit, kuten valikot, napit, logot ja HUD. 2D-objekteja työsti tiimin harjoittelija graafikko, sekä päägraafikko, 3D-mallintamisen yhteydessä. Kuten 3D-malleissa, hyvin usein piirrettiin nopeaa konseptitaidetta, ennen kuin ruvettiin piirtämään varsinaista 2D-objektia.

2D-objektit koostuivat enimmäkseen pelin graafisista käyttöliittymä (engl. User Interface) elementeistä, kuten HUD:ista (Heads up display), valikoista ja napeista. Graafiset käyttöliittymäelementit yksinkertaisesti auttavat välittämään tarpeellista tietoa ohjelmasta käyttäjälle ja käyttäjää navigoimaan ohjelmassa. Yleensä hyvä graafinen käyttöliittymä peleissä mielletään sellaiseksi, joka on selkeä, intuitiivinen ja joka istuu pelin maailmaan.

Lähes kaikki pelin 2D-objektit toteutettiin vektoreilla, jotka muunneltiin bittikarttaviksi (tässä tapauksessa .PNG tiedostomuotoon), kuten esimerkiksi pelin logo (ks. kuvio 23 & kuvio 24). Tämä oli suuri etu uudelleenkäytettävyyden kannalta, sillä 2D-objektien resoluutiota pystyttiin muuttamaan alkuperäisestä vektoritiedostosta tarpeen tullen, ilman että 2D-objektin kuvanlaatu kärsi.

Ensimmäinen versio logosta oli mahdollisimman nopeasti toteutettu pelin ensimmäistä teaseria varten.



Kuvio 24. Ensimmäinen versio pelin logosta

MEDICAL FORCE OVERDRIVE!

Kuvio 25. Nykyinen versio pelin logosta (Clusterloop 2017)

Molemmissa versioissa käytettiin ilmaisessa jakelussa olevia fontteja. Nykyisen logon (ks. kuvio 25) fontti toimii myös pelin pääfonttina pelin sisäisissä tekstissä sekä numeroissa. Kyseinen iteraatio logosta koettiin olevan lähempänä pelin teemaa.

Ensimmäisenä 2D-grafiikka, mitä peliin toteutettiin, olivat juurikin pelin HUD. Pelin prototyyppien alussa HUD koostui vain tekstistä ja numeroista, Unreal Engine 4:n vakio fontilla (ks. kuvio 15). Ensimmäinen vektoreilla tehdystä pelin HUD:ista tehtiin klinisen näköinen (ks. kuvio 25). Väreiksi valittiin sellaisia, jotka toisivat mieleen sairaalaa ja terveydenhuoltoon liittyvää teemaa. Ensimmäiset ikonit toteutettiin hyvin suoriksi ja kylttimäisiksi, jossa jokainen ikoni kuvasti pelisession sisäisen HUD:in elementtejä, joilla annettaisiin tietoa pelaajalle pelisessioista. Näihin elementteihin lukeutuivat kello, vihreät ristit, sekä säätövivut. Kello kuvastaa pelisession aikamäärettä. Ristit kuvastavat pelistä saatuja pisteitä, ja säätövivut, jotka kuvastavat ominaisuudet -navigointinappia.



Kello



Ominaisuudet



Pisteet

Kuvio 26. HUD:in keskeiset ikonit



Kuvio 27. HUD:in keskeiset ikonit pelin pre-alpha versiossa

Pelin aikaisessa kehitysvaiheessa oli ideana parantaa vain tietyn nimiset potilaat sairaalasta, jotta tason saa läpäistyä. Nämä potilaat kuvattiin potilasinfokortilla ja potilasnimelistalla (ks. kuvio 27). Tätä ideaa testattiin ensimmäisissä versioissa, mutta se kävi pelaajille hyvin hankalaksi ymmärtää.

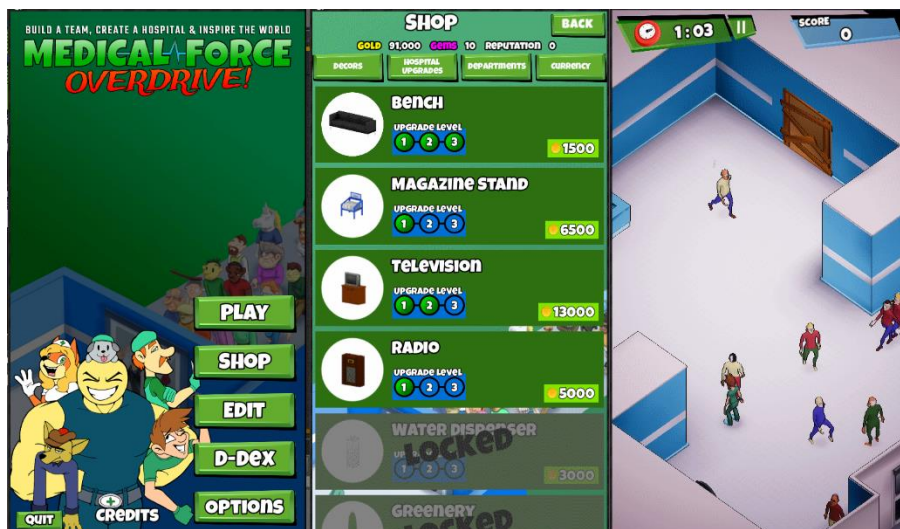
Alun perin pelissä oli ideana hoitaa tietyt listan mukaiset potilaat. Potilaslista myöhemmissä versioissa korvattiin periaatteella, jolla pelaaja hoitaa kaikki sairaalaan tulevat potilaat aikarajan sisällä, koska se koettiin pelitestauksessa selkeämmäksi tavoitteeksi. (Puuronen 2017.)

Tästä syystä pelin UI:ta lähdettiin suunnittelemaan uudelleen. Pelin UI on kirjoitushetkellä vielä työn alla. UI:sta on tavoitteena iteroida mahdollisimman dynaaminen, yksinkertainen, ja pelin teemaan sopiva.



Kuvio 28. Uuden HUD:in konseptitaidetta pelin kuvakaappauksen päällä

Tämän hetkinen iteraatio UI:sta vastaa aikaisempaa konseptointia, mutta UI:ta tul-
laan optimoimaan ja mahdollisesti muokkaamaan uudelleen, kun peliin ollaan pää-
tetty lukkoon uusi väriteema. Optimoimiseen tulee myös vaikuttamaan pelin kuva-
suhteen vaihtuminen vaakasuunnasta pystysuuntaiseksi. Tämä muutos tehtiin siksi,
koska peliä haluttiin voivan pelata yhdellä kädellä puhelimesta. Pelkästään UI-
elementtien uudelleen asettelu ei välttämättä riitä korjaamaan ongelmaa, vaan UI-
elementtejä joutuu mahdollisesti suunnittelemaan uudelleen. Jotta pelikokemuk-
sesta saataisiin vielä laadukkaampi, UI-elementit pyritään asettamaan paremmin uu-
teen kuvasuhteeseen.



Kuvio 29. Kolme kuvankaappausta pelin nykyisen version eri osista

5.4 Tulokset

Ongelmia on muodostunut grafiikan suhteen siitä, kuinka saada grafiikka näyttämään hyvältä ja ammattilaismaiselta ja samalla saada pelin ruudunpäivitys pysymän tasaisen sulavana. Tämä alue vaatii vielä lisää tutkimusta ja testailua koko tiimin osalta.

Grafiikan muodostuminen peliprojektiin tapahtui alkuunsa epävarmalla pohjalla, mikä johtui osittain päägraafikon kokemattomuudesta. Periaatteessa peliin oltiin alkuunsa mietitty ja testattu toimivaa uniikkia pelimekaniikkaa. Sitä tukemaan päägraafikko halusi toteuttaa sarjakuvamaista grafiikkaa. Mitä pidemmälle peliprojekti on edennyt, sitä enemmän päägraafikko on oppinut uusia menetelmiä monipuoliseen grafiikan tuottamiseen ja jakanut oppiansa muille tiimin jäsenille.

Grafiikan muodostumiseen on vaikuttanut käytännössä kolme asiaa: pelin testauksista saatu palaute, grafiikan tuottajien aikaisempi kokemus ja grafiikan optimointi mobiililaitteita varten.

Grafiikan tuottaminen mobiilialustalle eroaa nimenomaan grafiikan optimoimisella, jota mobiilialustoille voi joutua tekemään enemmän, verrattuna esimerkiksi PC- ja konsolialustoihin. Toteutustavat ovat muuten samankaltaiset. Mobiilialustat kehittyvät jatkuvasti ja ohjelmistotuki kasvaa niille, mikä helpottaa grafiikan optimointia tulevaisuudessa.

6 Pohdinta

Opinnäytetyön toteuttamisen aikana olen löytänyt mielenkiintoisia lähteitä, jotka ovat auttaneet tiimiä pelin työstämisessä. Koko peliprojektin aikana on herännyt ajatuksia, kuinka asioita oltaisiin voitu tehdä toisin. Olisiko sittenkin ollut viisaampi käyttää ilmaiseen jakeluun levitettyjä malleja pelin demoamisvaiheessa, ja käyttää aika korkealaatuisten mallien työstämiseen? Näitä korkealaatuisia malleja olisi voitu käyttää mm. markkinointimateriaaliin, kuten postereihin ja mainoskuviin

Peliprojekti on elänyt hyvin paljon koko vuoden aikana, johonka suurin vaikuttaja on ollut testaajilta saatu palaute. Pelin testaajina on ollut niin harrastelijoita, kuin pelialan ammattilaisia. Molempien ryhmien palaute on ollut erittäin arvokasta ja palaute itsessään on määrännyt suuntaan seuraaville toimenpiteille pelin suhteen.

Tutkimukseen kerättiin teoretietoa kirjoista, nettiartikkeleista ja eri ohjelmistojen omista ohjekirjoista. Myös blogeja ja haastatteluja hyödynnettiin. Pysin etsimään blogeista, joiden takana oli selvästi kokenut graafikko. Eräs lähteistä voi tosin olla kyseenalainen, sillä sen lähde ei juurikaan ole pelialaan keskittynyt sivusto. Yleisesti grafiikasta kertovana lähteenä havaitsin kattavimmaksi Andrew Paquetten kirjan ” An introduction to computer graphics for artists”. Tätä kyseistä kirjaa käytetään mm. peligraafikoiden opetuksessa mm. Hollannissa, Bredan ammattikorkeakoulun peligraafikoiden opinnoissa. Tästä syystä lähteet ovat suurimmalta osin kansainvälisiä ja mukana oli hyvin minimaalinen määrä suomenkielisiä lähteitä. Ohjelmien pääkielenä on englanti, mutta niihin on mahdollista hakea suomenkielisiä kielipaketteja. Tätä en itse kuitenkaan suosittelisi, mikäli haluaa pelialalle suuntautua, sillä peliala on hyvin kansainvälinen ala. Englanti on hyvinkin universaali kieli kyseisellä alalla. käytännön osio pohjautui kirjoittajan ja ryhmäläisten omakohtaisiin havaintoihin, joidenka pohjalta muodostui retrospektiivi Clusterloop Oy:lle.

Lähteet

Chastain, S. 2017. ThoughtCo:n sivuilta vector and bitmap images. ThoughtCo. Viitattu 12.11.2017. <https://www.thoughtco.com/vector-and-bitmap-images-1701238>.

Definition of: 3D graphics. 2017. Sanamääritelmä PC Magazinen verkkosivuilla. The computer language company inc. Viitattu 26.10.2017. <https://www.pcmag.com/encyclopedia/term/37072/3d-graphics>.

Editors. N.d. Viitattu 8.11.2017. https://docs.blender.org/manual/en/dev/editors/dope_sheet/introduction.html.

Inkscape Overview. N.d. Viitattu 30.10.2017. <https://inkscape.org/en/about/>.

Josua, S. 2013. What Is The Difference Between Indie And Triple-A? Leviathyn the gamer's chronicle. Viitattu 30.10.2017. <https://www.leviathyn.com/2013/08/22/what-is-the-difference-between-indie-and-triple-a/>.

Kananen, J. 2008 Kvali – Kvalitatiivisen tutkimuksen teoria ja käytänteet. Jyväskylän ammattikorkeakoulu.

Laakkonen, C. 2017. About drawpile. Drawpile collaborative drawing. Viitattu 30.10.2017. <https://drawpile.net/about/>.

Mamo, N. 2015. What makes an indie game an indie game? Quora. Viitattu 24.10.2017. <https://www.quora.com/What-makes-an-indie-game-an-indie-game>.

Mapping UVs. 2014. Introduction to UV mapping. Viitattu 1.11.2017. <https://knowledge.autodesk.com/support/maya/learn-explore/caas/CloudHelp/cloudhelp/2015/ENU/Maya/files/Modeling-htm.html>.

Modeling. N.d. Viitattu 2.11.2017. <https://docs.blender.org/manual/en/dev/index.html#>.

Modeling characters for optimal performance. N.d. Viitattu 1.11.2017. <https://docs.unity3d.com/Manual/index.html>.

Our story. 2017. Viitattu 30.10.2017. <http://www.clusterloop.fi/>.

Paquette, A. 2013. An introduction to computer graphics for artists. Springer Science+Business Media.

Puuronen, J. 2017. Toimitusjohtaja. Clusterloop Oy. Haastattelu 15.11.2017.

Sternberg, R. 2003. What is an "expert student?". Educational researcher.

The GIMP Team. N.d. GIMP GNU image manipulation program. Viitattu 30.10.2017. <https://www.gimp.org/about/>.

The software. N.d. Viitattu 30.10.2017. <https://www.blender.org/about/>.

Ward, J. 2008. What is a game engine? Game career.com. Viitattu 29.10.2017. https://www.gamecareerguide.com/features/529/what_is_a_game_engine.php?page=1.

Wright, C. 2008. A brief history of mobile games. PocketGamer. Viitattu 30.10.2017.
<http://www.pocketgamer.biz/r/PG.Biz/A+Brief+History+of+Mobile+Games/feature.asp?c=10618>.