



2D-materiaalipaketin toteuttaminen Unity Asset Store -verkkopalveluun

Sonja-Maria Juslin

Opinnäytetyö
Maaliskuu 2015
Tietojenkäsittelyn koulutus-
ohjelma
Pelituotanto

TIIVISTELMÄ

Tampereen ammattikorkeakoulu
Tietojenkäsittely
Pelituotanto

JUSLIN, SONJA-MARIA:

2D-materiaalipaketin toteuttaminen Unity Asset Store -verkkopalveluun

Opinnäytetyö 71 sivua, joista liitteitä 6 sivua
Maaliskuu 2015

Unity 3D -pelimoottori on kolmiulotteisten pelien tekemiseen tarkoitettu pelimoottori, johon julkaistiin vuonna 2013 työkalut myös kaksiulotteisille peleille. Unitylla työskentelevillä pelien kehittäjillä on käytössään Unity Asset Store -palvelu, jossa käyttäjät myyvät tekemäänsä pelimateriaalia valmispakettien muodossa. Uuden pelimoottoriversion myötä kaksiulotteisten pelien grafiikkapaketeille on kysyntää. Tämän opinnäytetyön tarkoituksena oli esitellä kaksiulotteisten pelien grafiikan tuottamiseen liittyvää teoriaa, jota voitaisiin hyödyntää peligrafiikkaa tehtäessä. Opinnäytetyön aikana toteutettiin projektiluontoisesti kaksiulotteisen pelin tekstuuripaketin, jonka tavoitteena oli havainnollistaa teoria- ja työskentelyosuudessa esiintyneitä asioita.

Materiaalipaketti suunniteltiin peligrafiikan toteuttamiseen liittyvän teorian pohjalta, joka koottiin erilaisista digitaaliseen grafiikkaan ja värioppiin liittyvistä kirjallisista materiaaleista. Työskentelyn aikana tehdyt tekniset ratkaisut pohjautuivat digitaalisen grafiikan optimointiin ja verkkopalvelun vaatimuksiin.

Graafinen materiaalipaketti valmistui tutkitun teorian pohjalta ja Unity Asset Store -palvelun vaatimusten mukaan. Teoriaosio painottaa digitaalisen grafiikan teknisiin ratkaisuihin kiinnitettäviä asioita ja värimaailman tärkeyttä pelin ymmärrettävyydessä ja teemassa. Tekstuuripaketin suunnittelu toi esille uusia näkökulmia pelin graafisten ratkaisujen toteuttamisessa. Teoriaosuudessa esiteltyjä toimintatapoja voidaan käyttää pelimoottorista riippumatta myös muissa peligrafiikkaan liittyvissä projekteissa. Jatkotutkimuksena voitaisiin tutkia peligrafiikan toteuttamista eri pelimoottorien kesken. Grafiikan optimointi ja käytettävyys peleissä ovat pelituotannon kannalta tärkeitä nouseita aiheita, joita voisi myös tutkia laajemmin.

ABSTRACT

Tampereen ammattikorkeakoulu
Tampere University of Applied Sciences
Bachelor's Degree in Business Information Systems
Game Production

JUSLIN, SONJA-MARIA:
The Making of a 2D Game Graphical Asset for Unity Asset Store

Bachelor's thesis 71 pages, appendices 6 pages
March 2015

Unity 3D is a popular game engine designed for 3D game development. A new version that included tools for 2D games was released in 2013. After the release there has been a need for ready-to-use 2D game art for Unity game developers in the Unity Asset Store – an online service for Unity – where game developers buy and sell game assets. The objective for this thesis was to create a 2D game graphics asset to be published in the Unity Asset Store. The purpose of this thesis was to research game art theory to be used in 2D game projects.

The asset was done as a project. It was designed on the basis of the thesis' theoretical section. A selection of material that covered basics of digital images, the colour theory in games and the usability related to game graphics was researched. The making of the asset pack was detailed in the thesis and it was used in the examples throughout the thesis.

The final result for the project was a graphical asset for a 2D platformer game. It follows guidelines set by the Unity Asset Store and it met the initial requirements. While thesis' theoretical section can be considered as a collection on the basics of the game art theory, the subject could be researched further from the aspect of optimization and usability of graphics in the game environment.

Key words: game graphics, digital graphics, 2D, Unity

SISÄLLYS

1	JOHDANTO.....	6
2	DIGITAALINEN GRAFIIKKA PELEISSÄ	8
2.1	Kaksiulotteisen peligrafiikan historia	8
2.2	Digitaalisen kuvan anatomia.....	10
2.3	Additiivinen ja subtraktiivinen värimalli	11
2.4	Rasteri- ja vektorigrafiikka	15
2.5	Sprite-kuvat ja tekstuurit.....	16
2.6	Digitaalisten kuvien pakkausmuodot.....	18
2.7	Peligrfiikan rajoitukset ja optimointi	21
2.8	Digitaaliset työvälineet	22
3	PELIEN VÄRIMAAILMA JA ESTETIIKKA	23
3.1	Värioppi peligrfiikassa	23
3.1.1	Vastavärit	24
3.1.2	Jaettu vastaväri ja vastaväripari	26
3.1.3	Lähivärit	27
3.1.4	Kolmisointu ja valööriharmonia	29
3.2	Värien hahmottaminen.....	31
4	UNITY-PELIMOOTTORI.....	33
4.1	Unity 3D ja Unity 2D.....	33
4.2	Unity Asset Store -palvelu.....	35
4.3	Tekstuuripaketin vaatimukset	35
5	TEKSTUURIPAKETIN SISÄLTÖ	36
5.1	Konsepti	36
5.2	Käytetyt työkalut.....	40
6	TEKSTUURIPAKETIN TOTEUTUS	41
6.1	Suunnittelu- ja luonnosvaihe	41
6.2	Taustatekstuurien toteutus	45
6.3	Peliobjektien toteutus.....	52
6.4	Kuvien viimeistely ja lähettäminen Unity Asset Storeen	59
7	POHDINTA.....	63
	LÄHTEET	64
	LIITTEET	66
	Liite 1. Lista tekstuuripakettiin valmistuneista kuvista.....	66
	Liite 2. Valmistuneet tekstuurit ja tekstuuriatlakset.....	67

LYHENTEET JA TERMIT

2D, kaksiulotteinen	pelialue rajoittuu kahteen ulottuvuuteen akseleilla X ja Y
3D, kolmiulotteinen	pelialue ulottuu kolmeen ulottuvuuteen akseleilla X, Y ja Z
kuvakaappaus	tallennettu kuva monitorin näkymästä
peligenre	pelin lajityyppi joka määrittelee pelin säännöille ja mekaniikalle tietyt rajat
pelikaappi	rakennelma jonka sisälle kolikkopeli asennetaan
pelimoottori	ohjelmistokehys, joka sisältää valmiita työkaluja pelien kehittämiseen
resoluutio	pikselien määrä vaakasuunnassa ja pystysuunnassa, esitetään esimerkiksi muodossa 1600 x 1200 pikseliä
saturaatio	värin kirkkaus
tekstiseikkailu	pelejä, jossa ongelmanratkaisu ja eteneminen riippuvat pelaajan syöttämistä käskyistä
skripti	valmiiksi ohjelmoitu koodikonaisuus
videopeli	elektroninen peli, jota pelataan erillisellä pelaamiseen tarkoitettulla koneella

1 JOHDANTO

Valmiiksi rakennetun pelimoottorin käyttäminen on kasvava trendi modernissa pelinkehityksessä. Eräs suosituimmista pelimoottoreista on yhdysvaltalaisen Unity Technologies -yrityksen kehittämä Unity 3D, jonka käyttäjiksi oli vuoden 2013 heinäkuuhun mennessä rekisteröitynyt 2 miljoonaa pelinkehittäjää. Unity 3D on kehitetty kolmiulotteisten pelien tekemiseen, mutta käyttäjät ovat voineet muokata ohjelmistokehystä tullaan myös kaksiulotteisten pelien toteuttamista. Marraskuussa 2013 Unity 3D -pelimoottori päivitettiin sisältämään valmiit työkalut kaksiulotteisten pelien tekemisessä. Tätä osaa pelimoottorista kutsutaan nimellä Unity 2D.

Ohjelmistokehityksen lisäksi Unity Technologies ylläpitää Unity Asset Store -nimistä verkkopalvelua. Unity-pelimoottoria käyttävät pelien kehittäjät voivat myydä tai ostaa palvelun kautta pelinkehityksessä käytettäviä materiaalipaketteja (engl. *asset*). Näitä paketteja ovat muun muassa valmiiksi ohjelmoidut koodikokonaisuudet eli *skriptit* (engl. *script*), kolmiulotteiset mallinnukset, valmiit käyttöliittymät ja peligrafiikkakokonaisuudet. Tarve kaksiulotteisia pelejä varten suunnitelluille grafiikkapaketeille on kasvanut Unity 2D -pelimoottorin ilmestymisen myötä.

Tämän opinnäytetyön aikana toteutetaan projektiluontoisesti kaksiulotteisen pelin grafiikkapaketti eli tekstuuripaketti. Peligrafiikan visuaalinen teema ja kohdepelin lajityyppi eli *genre* valitaan tekstuuripaketin suunnittelun aikana vertailemalla Unity Asset Store -palvelussa myytäviä materiaalipaketteja. Paketti laitetaan projektin päätyttyä myyntiin Unity Asset Storeen.

Tekstuuripaketin kuvakokonaisuuksien työstämistä tarkastellaan digitaalisen grafiikan tekemiseen liittyvän teorian kautta. Tavoitteena on ottaa selvää niistä graafisen alan osa-alueista, jotka auttavat ymmärtämään peligrafiikan tekemiseen liittyviä rajoituksia ja käytäntöjä. Tällaisiksi osa-alueiksi valittiin peligrafiikan historia, värioppi, käytettävyys ja digitaalisten kuvien optimointi. Opinnäytetyön tavoitteena on tunnistaa mahdollisia digitaalisen grafiikan tekemiseen liittyviä ongelmakohtia ja esitellä niitä toimintatapoja, jotka tekevät peligrafiikan toteuttamisesta sujuvampaa. Peligrafiikkaa tarkastellaan myös Unitylla toteutettavien pelien teknisten vaatimusten kautta. Tekstuuripaketin sisältö suunnitellaan ja toteutetaan siten, että sitä voidaan käyttää sekä pöytäkoneille että

mobiililaitteille tarkoitetuissa peleissä. Paketin sisällön laatuksiteerit noudattavat Unity Asset Storessa asetettuja vaatimusmäärittelyjä. Opinnäytetyössä esiteltyä teoriaa ja käytäntöä voidaan hyödyntää peligrafiikan projekteissa myös Unity-kehysten ulkopuolella. Valmistu opinnäytetyötä tullaan käyttämään opetusmateriaalina Tampereen ammattikorkeakoulun tietojenkäsittelyn koulutusohjelman pelituotannon kursseilla.

Opinnäytetyön luvut 2–4 sisältävät digitaaliseen grafiikkaan liittyvää teoriaa, jota hyödynnetään materiaalipakettia tehtäessä. Luku 2 selostaa digitaalisen grafiikan anatomiaa ja peligrafiikkaan liittyvää sprite-kuvia. Värioppia ja sen hyödyntämistä peleissä käsitellään luvussa 3. Unitya tarkastellaan peligraafikon näkökulmasta luvussa 4 ja samassa luvussa esitellään Unity Asset Storen hyötyjä pelinkehityksessä.

Luvut 5–6 keskittyvät tekstuuripaketin suunnittelun ja toteutuksen kuvailuun. Luvussa 5 esitellään suunnitelma paketin sisällöstä ja perustellaan sisältöön liittyviä ratkaisuja. Tekstuuripaketissa käytetyt työkalut käydään läpi tässä luvussa. Luku 6 kuvailee tekstuurien suunnittelua, toteuttamista ja kokoamista valmiiksi kokonaisuudeksi. Paketin valmistelemista myyntiin kuvaillaan alaluvussa 6.4.

2 DIGITAALINEN GRAFIIKKA PELEISSÄ

Pelikoneiden näytöillä esiintyvä pelitilanne koostuu monista erilaisista visuaalisista elementeistä. Kaksiulotteisen pelin hahmot, objektit ja käyttöliittymä muodostuvat yksittäisistä kuvista, jotka piirretään ohjelmallisesti näytölle. Tässä luvussa käydään läpi kaksiulotteisen peligrafiikan historiaa ja sitä, mitä peligrafiikka on kaksiulotteisten pelien yhteydessä.

2.1 Kaksiulotteisen peligrafiikan historia

Elektronisia pelejä on tunnettu jo 1940-luvulta lähtien, mutta varsinaiset ohjelmoidut pelit alkoivat ilmestyä laajemmalle yleisölle pelattavaksi vasta 1970-luvulla. Vuonna 1971 julkaistua Computer Space -kolikkopeliä pidetään pelihistorian ensimmäisenä kaupallisena videopelinä. Tätä ennen tietokoneille ohjelmoidut pelit – muun muassa Spacewar johon Computer Space perustui – ohjelmoitiin yhdysvaltalaisissa teknillisissä yliopistoissa tutkimuskäyttöön tarkoitetuille tietokoneille, eivätkä ne olleet suuren yleisön ostettavissa tai pelattavissa (Eddy 2012, 5–6; Levy 2001, 65). Nykyisen peliteollisuuden voidaan katsoa alkaneen vuoden 1972 Atarin Pong-videopelin ilmestymisen myötä. Pongia pidetään ensimmäisenä kaupallisesti menestyneenä kolikkopelinä Atarin myytyä kaikkiaan 35 000 Pong-pelikaappia. (Eddy 2012, 7.)

Pong oli rakennettu mustavalkotelevisioon, jonka ruudulla näkyvät peliobjektit – pelaajien mailat, pallo ja pisteet – piirrettiin näkyviin television pikselijanojen eli viivojen avulla (Goldberg & Vendel 2012, 70, 72). Pelit tehtiin yksiväriä näytöille, jotka tukivat vain yhtä väriä mustaa taustaa vasten. Näyttöjen pieni resoluutio ja huomattavan rajattu värivalikoima oli vaatimaton alku peligrafiikan historialle eikä tällä teknologialla pystytty visualisoimaan monimutkaisia pelitapahtumia tai -ympäristöjä. Suurin osa pelien visuaalisesta annista syntyi pelaajien omassa mielikuvituksessa. Esimerkiksi 1970-luvulla ohjelmoitu Colossal Cave Adventure -peli, joka tunnetaan myös nimellä ”Adventure”, oli seikkailupeli joka ei sisältänyt varsinaista grafiikkaa ja pelin tapahtumien visualisointi oli pelaajan mielikuvituksen varassa. Pelin ympäristö, tapahtumat ja esineet kuvailtiin tekstinä. Pelaaja antoi tekstistä ilmenevien tietojen perusteella yhden tai kahden sanan komentoja, joilla hän pyrki pääsemään eteenpäin pelissä. Colossal Cave Ad-

venture aloitti uuden peligenren: tekstiseikkailun, joka edelsi graafisia seikkailupelejä. (Eddy 2012, 22–23, 26; Levy 2001, 141.)

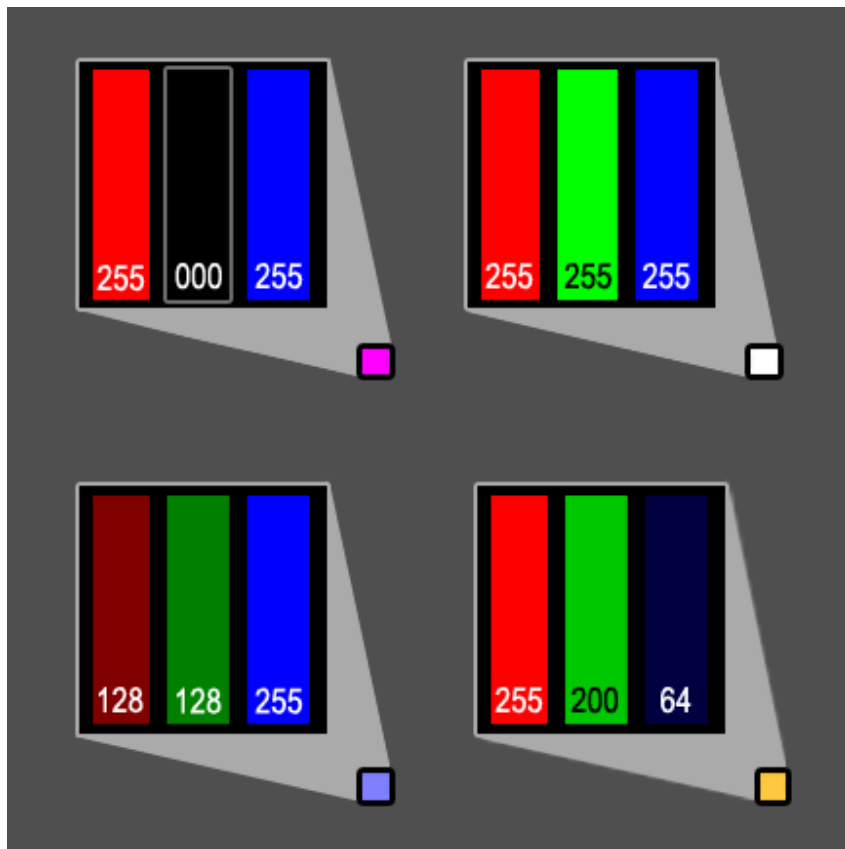
Vuonna 1980 Ken ja Roberta Williams julkaisivat ensimmäisen tekstiseikkailun, jossa peliympäristöä visualisoitiin graafisesti. Ken Williams ohjelmoi pelin rungon Apple II -tietokoneelle ja siirsi Roberta Williamsin 70 käsin piirrettyä kuvaa peliin. Kuvat saatiin mahtumaan $5\frac{1}{4}$ -tuuman levykkeelle käyttämällä konekielisiä komentoja, jotka osoittivat kuville ne koordinaatit, joiden väliin kuvissa esiintyvät geometriset janat piirrettäisiin. (Levy 2001, 298.) Tällainen peligrafiikka, joka perustuu koordinaatistossa sijaitseviin geometrisiin kappaleisiin, tunnetaan *vektorigrafiikkana* (Kuorikoski 2014, 251.)

Videopelien suosio kasvoi edelleen tulevina vuosikymmeninä, erityisesti kotikäyttöisten tietokoneiden yleistyessä ja teknologian kehittyessä. Peleissä pystyttiin esittämään useita värejä, monitorien resoluutio kasvoi ja peleihin pystyttiin piirtämään digitaalisia kuvia matemaattisten koordinaattien laskemisen sijaan. Peligraafikko Miha Rinne kuvailee peligrafiikan piirtämistä 1990-luvulla ”raskaaksi käsityöksi”, joka tarkoitti jokaisen pelissä esiintyvän kuvan ääriviivojen piirtämistä, kuvan värittämistä ja reunaviivojen korjausta pikseli kerrallaan (Kuorikoski 2014, 39). Rinteen kuvailema pikseleihin perustuva peligrafiikka tunnetaan *rasterigrafiikkana*. Opinnäytetyön luvussa 2.4 tarkastellaan lähemmin vektori- ja rasterigrafiikkaa peligrafiikkana.

Nykyisin peligrafiikan tuottaminen on helpottunut piirtopöytien, piirtomonitorien ja valmiiden piirto-ohjelmien ansiosta. Käsin piirrettyjä kuvia ja tekstuureja voidaan piirtää ruudulle samaan tapaan kuin kynällä paperille. Digitaalinen piirtojälki on nähtävissä välittömästi ruudulla. Peligrafiikkaa tehtäessä tietyt rajoitukset ovat edelleen voimassa. Grafiikkaa suunniteltaessa tuleekin ymmärtää, paljonko muistia tuotettu grafiikka vie ja kuinka paljon levytilaa kuvat vievät pelilaitteelta. Näillä tiedoilla voidaan ennakoida niitä tapoja, joilla voidaan estää peligrafiikkaa viemästä turhaa tilaa tai muistia, aivan kuten pelihistorian alkuaikoinakin.

2.2 Digitaalisen kuvan anatomia

Kaikki tietokoneen näytöllä havaittava väri saa alkunsa valosta. Esimerkiksi nestekide eli LCD-näytöissä esitettävään värikuvaan vaaditaan neljä tekijää: polarisoitava taustavalo, nestemäinen kristalli joka kuljettaa ja taittaa polarisoitua valoa, sähkövirta joka muuttaa kristallin rakenteen toiseen muotoon ja läpinäkyvä sähkövirtaa johtava aines. Taustavalosta alkunsa saava valonsäde polarisoituu kahden lasin kautta kulkiensa edelleen ruudun etuosaan, johon värit heijastuvat. (Tyson 2011, 2.) Jos näytössä havaittavia kuvia tarkastellaan tarpeeksi läheltä, voidaan nähdä kuinka ruudulle heijastetut kuvat koostuvat pikseleistä. Jokainen pikseli koostuu edelleen punaisesta, vihreästä ja sinisestä alipikselistä (kuva 1). Alipikseleiden värifiltterit heijastavat väriänsä taustavalosta lähtevästä kirkkaudesta riippuen ja näiden alipikseleiden värien kirkkaudet vaikuttavat yhdessä siihen, minkä värinen näytöllä näkyvä pikseli on. Kauempaa katsottuna nämä kolme eri pääväriä sulautuvat katsojan silmissä yhdeksi väriksi. (Tyson 2011, 6; Kane 2014, 123.)



KUVA 1. Näytöllä näkyvät pikselit alipikseleineen.

Pääväreillä tarkoitetaan niitä värimallin värejä, joita yhdistelemällä saadaan aikaan muita värejä (Stone, Adams & Morioka 2006, 10). Päävärejä ei voida sekoittaa värimallin muista väreistä. Värimallit ovat värien kuvaamiseen kehitettyjä väritiloja, joiden avulla tietty väri voidaan kuvata mahdollisimman tarkasti samanlaisena käyttökohteesta riippuen.

Usein käytettyjä värimalleja ovat muun muassa RGB, RYB ja CMYK. RGB-värimallista puhutaan valoon perustuvia värejä kuvatessa, esimerkiksi aiemmin mainittu LCD-näyttö käyttää RGB-värimallia. CMYK-värimallia käytetään painotyössä ja RYB viittaa perinteiseen taiteessa käytettävään värioppiin. Värimallien lyhenteet tulevat niiden päävärien englanninkielisten nimien alkukirjaimista: RGB-värimallin päävärit ovat punainen, vihreä ja sininen (Red, Green, Blue) ja RYB-värimallin taas punainen, keltainen ja sininen (Red, Yellow, Blue). CMYK-värimallissa puolestaan päävärit ovat syaani, magenta, keltainen ja jokin avainväri, yleensä musta (Cyan, Magenta, Yellow, Key color). Värimallit jaetaan joko additiiviseen tai subtraktiiviseen värimalliin riippuen siitä, miten värimallissa muodostetaan värejä. (Stone ym. 2006, 10–11.)

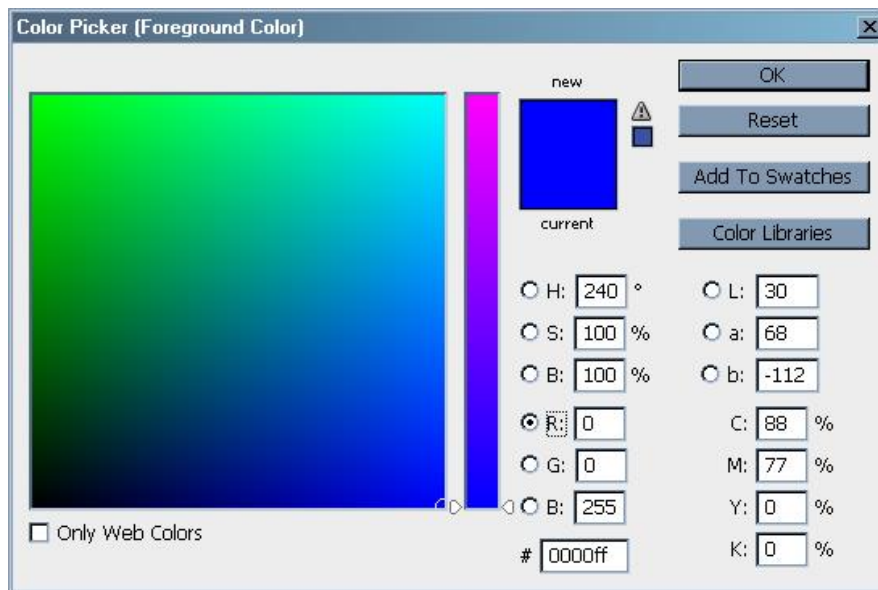
2.3 Additiivinen ja subtraktiivinen värimalli

Tietokonegrafiikassa, jota ei tulla tulostamaan esimerkiksi sanomalehtiin tai julisteisiin, käytetään additiivista RGB-värimallia. Additiivinen värimalli perustuu valoisuuden lisäämiseen (Kane 2014, 339). Ihmissilmin havaittava valkoinen valo on kaikkien spektrin värien yhdistelmä, eli mitä enemmän ja vahvemmin värejä yhdistetään keskenään additiivisessa värimallissa, sitä lähempänä sekoitettu väri on puhdasta valkoista. Esimerkiksi silmin havaittava valkoinen valo koostuu kaikkien spektrin värien yhdistelmästä (Stone ym. 2006, 10).

Digitaalisia värejä voidaan ilmaista värimallista riippuen erilaisina päävärejä kuvaavina lukusarjoina. Esimerkiksi kuvassa 2 nähdään, miten RGB-värimallin päävärejä vastaaville tekstikentille on annettu arvo lukujen 0 ja 255 väliltä. Nämä kolme lukua merkitsevät kunkin pääväriin kirkkautta ja muodostavat yhdessä numerosarjan, jonka avulla värit voidaan esittää mahdollisimman samanlaisina eri tarkkailualustojen välillä. RGB-mallissa värit ilmaistaan esittämällä kolme lukusarjaa pilkuilla erotettuina järjestyksessä

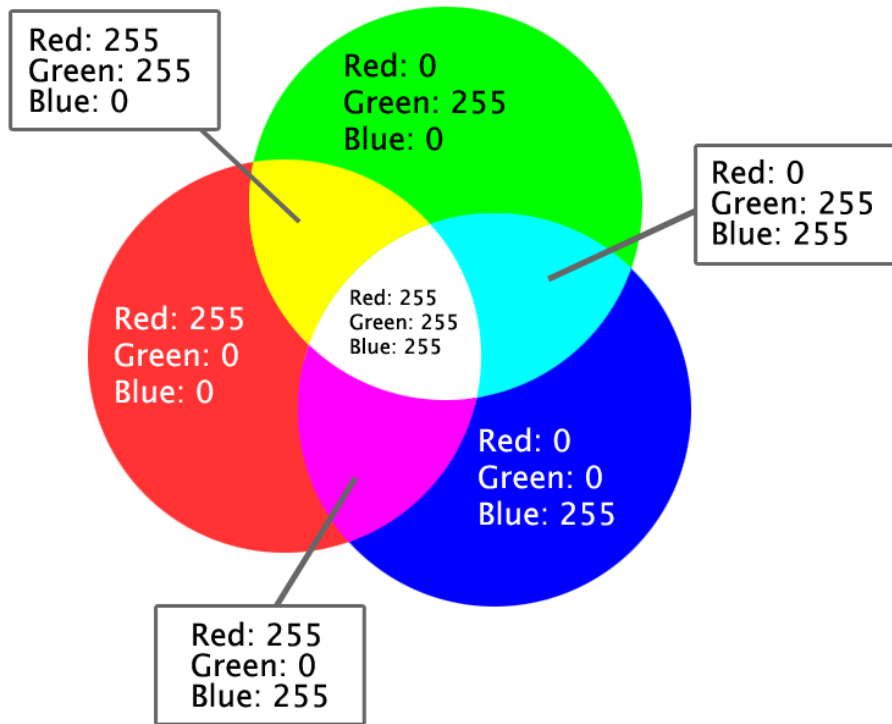
punainen, vihreä ja sininen. Esimerkiksi kuvan 2 väri esitetään RGB-mallissa muodossa 0,0,255.

Adobe Photoshop -ohjelmiston Color Picker -näkyssä voidaan tarkastella RGB-väriin määrääytymistä sen mukaan, miten suuri komponenttiarvo on annettu joko punaiselle, vihreälle tai siniselle päävärille (kuva 2). Mitä lähempänä arvo on maksimiarvoa 255, sitä kirkkaampi eli valoisampi väri on. Vastaavasti arvon ollessa 0 pääväriä ei esiinny lainkaan värissä. Kuvassa 2 punaiselle ja vihreälle päävärille on annettu arvoiksi 0. Ainoastaan siniselle on annettu maksimiarvo 255, jolloin tuloksena on puhdas sininen väri.



KUVA 2. Color Picker -näky Adobe Photoshop CS3 -ohjelmasta.

Päävärien avulla voidaan muodostaa muita värejä eli niin kutsuttuja välivärejä (kuva 3). Esimerkiksi kirkkain mahdollinen syaani luodaan lisäämällä vihreälle ja siniselle maksimiarvot 255. Kuten aiemmin on todettu, additiivinen värimalli perustuu valoon ja valkoinen valo on kaikkien värien yhdistelmä. Jos kaikkien värien komponenttiarvoksi annetaan 255, saadaan puhtain mahdollinen valkoinen väri. Vastaavasti täydellinen musta väri saadaan antamalla värien komponenttiarvoiksi 0.



KUVA 3. Additiivinen RGB-värimalli pää- ja väliväreineen.

Jokainen pääväri vastaa yhtä kahdeksan bitin joukkoa eli tavua¹, joka voidaan kuvata arvoilla 0 ja 255 välillä tai heksadesimaalilukuna. Heksadesimaalissa muodostuu kolmesta lukuparista, joita edeltää #-merkki. RGB-värimallin kolme arvoa muutetaan heksadesimaalimuotoon, jolloin esimerkiksi RGB-värimallissa syaani 0,255,255 muuttuisi muotoon 00,ff,ff ja tästä edelleen heksadesimaaliluvuksi #00ffff². Heksadesimaali on helpompi tapa ilmaista väriä, sillä kopioitavia lukuja on yksi kolmen tai neljän sijaan.

RGBA-mallissa värejä vastaavien tavujen jatkoksi tulee vielä kahdeksan bittiä, jotka kuvaavat väripikselin läpinäkyvyyttä. Tätä tavua kutsutaan alfa-kanavaksi (engl. *alpha channel*). (Kane 2014, 341.) Luvussa 2.5 kuvataan läpinäkyvyyden käyttöä peligrafii-kassa.

Värien kirkkautta eli saturaatiota säädellään muokkaamalla värien arvoa pienempään tai isompaan lukuun. Taulukossa 1 esitetään kuinka erilaiset väriarvojen yhdistelmät tuottavat uusia välivärejä, esimerkiksi väriarvojen joukkio 126, 126, 126 merkitsee harmaata väriä. Eri värisävyjä tehdään lisäämällä toista sävyä tai vähentämällä värien arvoa.

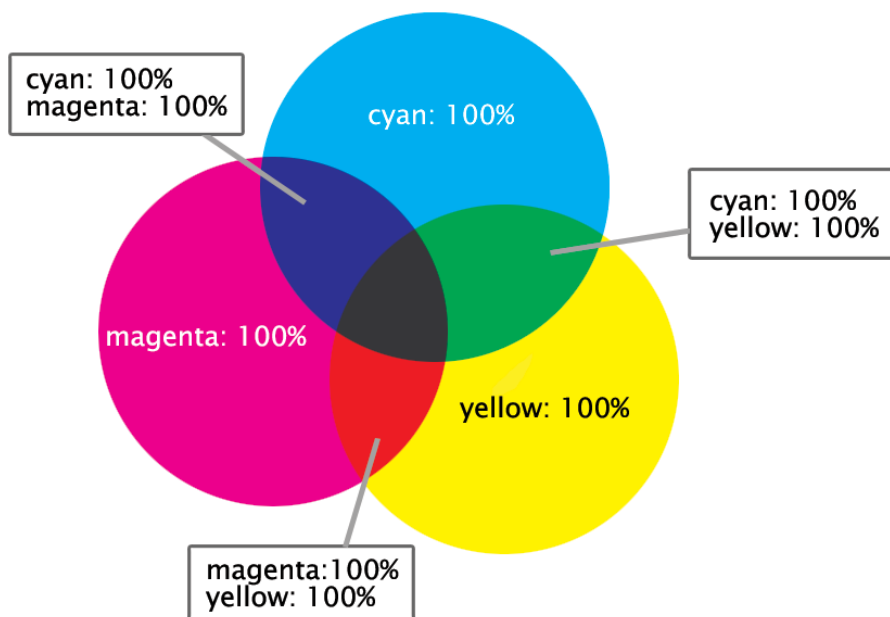
¹ Yhdellä bitillä voi olla kaksi tilaa, jolloin bittien tilojen määrä kasvaa eksponentiaalisesti sen mukaan, mitä enemmän bittejä käytetään. Päävärin eri tiloja voidaan laskea olevan $2^8 = 256$.

² Heksadesimaalissa luvut ilmaistaan välillä 0–9 ja a–f, josta a–f vastaavat desimaalijärjestelmän lukuja 10–15. Esimerkin 'ff' on desimaalijärjestelmässä 255.

TAULUKKO 1. Eri väriyhdistelmiä RGB-värimallissa.

Punainen	Vihreä	Sininen		Sekoitettu väri
126	126	126	=	126,126,126
20	200	255	=	20,200,255
255	190	0	=	255,190,0

Additiivisen värimallin vastakohta on subtraktiivinen värimalli, joka perustuu valoisuuden vähentämiseen. Tällaisessa värimallissa pääväreille annetut korkeat arvot muuttavat väriä lähemmäs mustaa, kun taas alhaiset arvot muuttavat väriä lähemmäs valkoista (Stone ym. 2006, 11). Digitaalisissa töissä käytetyin subtraktiivinen värimalli on CMYK, jota käytetään pääasiassa tulostustöiden painamiseen. Tätä värimallia ei tulisi käyttää digitaalisessa grafiikassa tai peligrafiikassa, sillä RGB-malliin verrattuna CMYK-värit näyttävät harmaammilta tai haaleammilta eivätkä esitä kaikkein kirkkaimpia ja runsaimpia värisävyjä näytöllä oikein. Tämä johtuu painotekniikasta, joka perustuu neljään pääväriin. Näytöllä näkyvät värit ovat mahdollisimman lähellä niitä värejä, jotka esiintyvät painotöissä. CMYK-mallin väreissä värikylläisyys esitetään prosentteina (kuva 4). Värien prosenttiarvot annetaan piirto-ohjelmissa neljälle komponenttiarvolle C, M, Y ja K (kuva 1).



KUVA 4. Subtraktiivinen CMYK-värimalli pää- ja väliväreineen.

Toinen subtraktiivinen värimalli on RYB-värimalli, jonka päävärejä ovat punainen, sininen ja keltainen. Toisin kuin edellä mainittuja värimalleja, tätä mallia ei käytetä digitaalisessa grafiikassa. Sen sijaan se toimii perinteisessä taiteessa väriopin perustana (Stone 2010, 18). Tätä värimallia käytetään esimerkkinä luvussa 3.1 väriopin yhteydessä.

2.4 Rasteri- ja vektorigrafiikka

Digitaalisia kuvia voidaan toteuttaa joko rasteri- tai vektorigrafiikkana. Rasterigrafiikka koostuu väripikseleistä, kun taas vektorigrafiikka koostuu tietokoneen avulla lasketuista geometrisista muodoista, esimerkiksi janoista, pisteistä ja kaarista. Näin ollen vektorigrafiikkakuva näyttää aina samalta suurennettaessa tai pienennettäessä, koska kuvassa esiintyvät geometriset kuviot lasketaan uudestaan annettuja sääntöjä noudattaen. Lisäksi pelaamiseen käytettävän laitteen muistia kuormittuu vähemmän, jos grafiikka toteutetaan vektoreilla, kun erillisiä kuvatiedostoja ei tallenneta laitteen muistiin pelaamisen aikana. Rasterikuvassa taas tapahtuu häviötä kuvaa suurennettaessa tai pienennettäessä, sillä kuvaa käsittelevä ohjelma valikoi mitkä väripikselit säästetään, mitkä muutetaan ja minne pikselit lopulta sijoitetaan. Tämän takia kuva näyttää erityisesti suurennettaessa suttuisemmalta. Pienennettäessä kuva voi menettää tarkkuuttaan pakkaustavasta riippuen. Rasterikuvassa on kuitenkin mahdollista esittää yksityiskohtaisempaa kuvitusta kuin vektorigrafiikalla, eikä kuville tarvitse ohjelmoida samanlaisia geometrisia matemaattisia sääntöjä kuin vektorigrafiikalle.

Peligrfiikkaa tehtäessä tulisikin miettiä, tullaanko peligrfiikka esittämään vektori- vai rasterimuodossa. Kuten luvussa 2.1 todettiin, peligrfiikassa käytettiin pelihistorian alkuvuosina enemmän vektorigrfiikkaa teknologisten rajoitusten vuoksi. Nykyisin peleissä käytetään enemmän rasterigrfiikkaa muun muassa käyttöliittymän kuvakkeissa, taustoissa ja hahmoissa. Rasterigrfiikan rajoituksena on kuvissa tapahtuva kuvanlaadun heikkeneminen, jos kuvaa esimerkiksi yritetään suurentaa jälkikäteen. Rasterigrfiikkaa tulisi mieluummin tehdä mahdollisimman suuressa koossa, joka pienennetään siihen kuvakokoon, jota pelissä tullaan käyttämään, sillä kuvanlaatu heikkenee vähemmän kuvaa pienennettäessä (Unity Art Asset Best Practice Guide 2014). Yksinkertaisista elementeistä koostuvaa grafiikkaa voidaan toteuttaa aluksi vektorigrfiikkaan tarkoite-

tulla ohjelmistolla ja tallentaa lopuksi eri pelissä käytettäviin kokoihin rasterikuvamuodossa, jolloin grafiikassa on hyödynnetty molempien kuvamuotojen parhaita puolia.

2.5 Sprite-kuvat ja tekstuurit

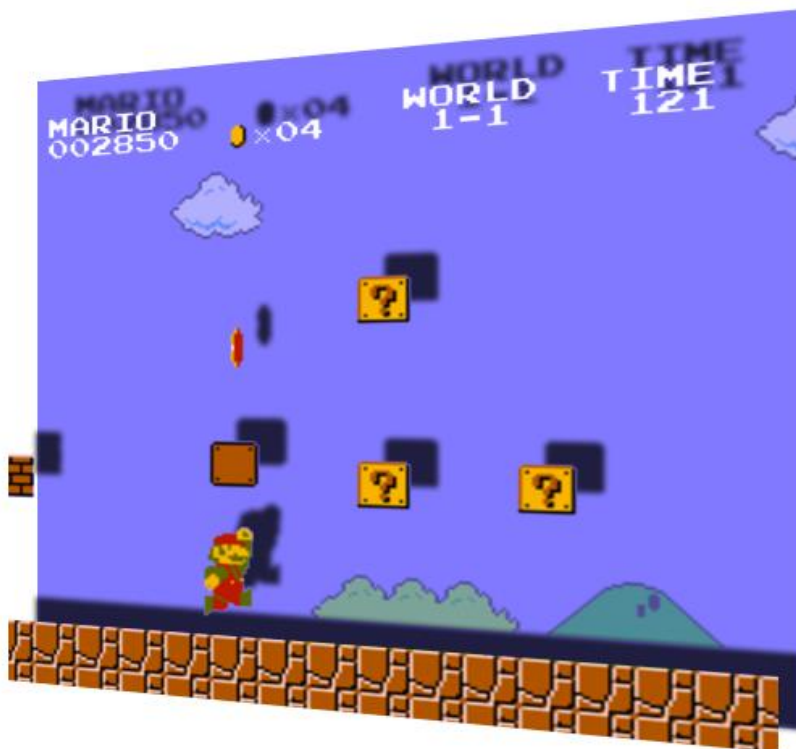
Digitaaliset kuvat voivat sisältää täydellistä tai osittaista läpinäkyvyyttä RGBA-värimallilla. Läpinäkyvyyden avulla kuva-tasoja voidaan sijoittaa päällekkäin tai lomitain siten, että ne näyttävät olevan toistensa edessä tai takana. Pelit hyödyntävät paljon tällaista grafiikkaa erityisesti kaksiulotteisissa peleissä. Tällainen peliobjekti tai -hahmo tunnetaan sprite-kuvana (Kuorikoski 2014, 251).

Esimerkkejä pelissä esiintyvistä sprite-kuvista voi nähdä kuvista 5 ja 6. Kuva 5 esittää kuvakaappausta Nintendo Entertainment System -pelikoneelle ilmestyneestä Super Mario Bros. -pelistä vuodelta 1985. Pelinäkömman yläosa on varattu pelissä olennaiselle tiedolle eli pisteille, kerätyille kolikoille, pelikentän numerolle sekä jäljellä olevalle peliajalle. Taustalla on yksivärinen sininen taustaväri kuvaamassa taivasta ja taustan edessä on koristeiksi tarkoitettuja elementtejä kuten pilviä sekä pensaita, joiden kuva-tasojää pelihahmon kuva-tason taakse. Pelihahmo, maankamara ja kysymysmerkkilaatikot sijaitsevat taustan edessä. Yksi kysymysmerkkilaatikko on muuttunut paljaan ruskeaksi kuvaamaan sitä, että laatikko on tyhjä eikä se reagoi enää pelihahmon toimintoihin. Laatikon päälle on noussut animoitu kolikko, joka kertoo pelaajalle, että hän on kerännyt kolikon mukaansa.



KUVA 5. Super Mario Bros. -pelin kuvankaappaus (Nintendo 1985).

Kuvassa näkyvät pelihahmo, laatikot ja kolikko ovat sprite-grafiikkaa. Pelinäkömman eri elementit voidaan erottaa yksittäisiksi sprite-kuviksi, jotka voidaan esittää kerroksittain toistensa päällä (kuva 6). Pelihahmo näyttää tällä tavalla kulkevan taustalla olevien pensaiden ja vuorien edestä luoden yksinkertaisen syvyysvaikutelman.



KUVA 6. Pelin eri tasot ja sprite-kuvat havainnollistettuna.

Sprite-kuvia voidaan koota niin kutsutuksi sprite-kartaksi tai tekstuuriatlakseksi (kuva 7). Kuvassa on esimerkki sprite-kartasta, johon on koottu yksittäisiä sprite-kuvia. Tällaiset kuvakokonaisuudet ovat suositeltuja muistin säästämiseksi (de Byl 2014, 213), sillä ne ladataan muistiin vain kerran. Laitteen muistiin tallennetusta atlaksesta rajataan yksittäisiä tekstuureja peliin. Tekstuuriatlaksia tulisikin käyttää Unity 3D -pelimoottorissa erityisesti mobiilipelien grafiikoissa, tai peleissä jotka vaativat paljon erilaisia kuvia, koska mobiililaitteiden muisti on rajatumpi pöytäkoneisiin verrattuna. (Unity Manual Optimizations 2014.)



KUVA 7. Esimerkki sprite-kartasta.

2.6 Digitaalisten kuvien pakkausmuodot

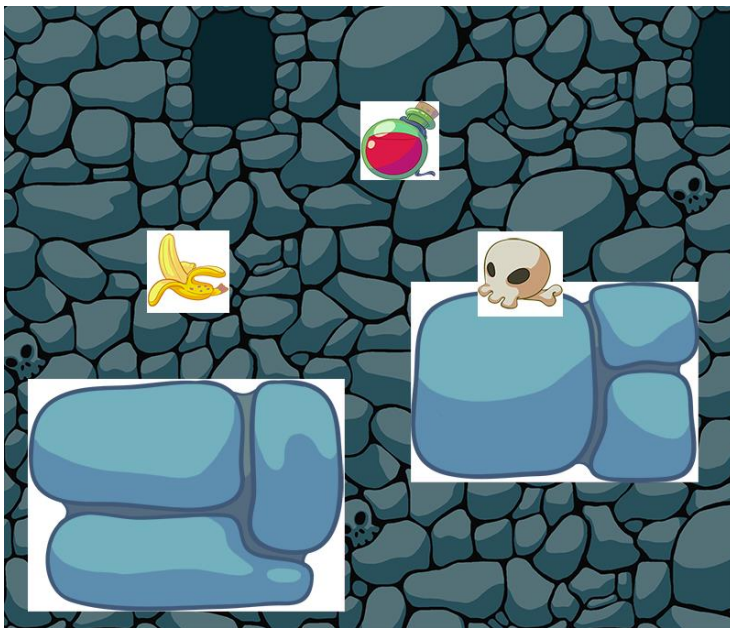
Kuva 7 on tallennettu 32-bittiseksi PNG-tiedostoksi (Portable Network Graphics), joka on toinen kuville tarkoitetuista läpinäkyvyyttä tukevista tiedostotyypeistä. Kuvassa esiintyvä harmaa ruudukko esittää niitä kohtia kuvasta, jotka ovat läpinäkyviä. Toinen läpinäkyvyyttä tukeva kuvatiedostomuoto on GIF (Graphics Interchange Format).

Unity 3D -pelimoottori tukee sekä PNG- että GIF-tiedostoja, mutta PNG-tiedostojen käyttäminen on suositeltavampaa. GIF-tiedostot ovat tiedostokooltaan pieniä, mutta ne tukevat enintään 256 väriä. Tämä rajoittaa peligrafiikan värimaailmaa huomattavasti enemmän kuin PNG-kuva, joka tukee yli 16 miljoonaa väriä 24-bittisenä versiona. PNG- ja GIF-pakkausmuodot ovat niin sanotusti häviöttömiä pakkausmuotoja, eli tiedostoa pakattaessa data pysyy muuttumattomana ja kuvataan sellaisella tavalla, joka pienentää tiedoston kokoa. Pakkausta purettaessa data esitetään sellaisena kuin se oli alun perin (Gullen 2013).

Kuvat 8 ja 9 esittävät kuvitteellista pelitilannetta, jossa käytetään tämän opinnäytetyön aikana toteutettuja sprite-kuvia. Kuten aiemmassa Super Mario Bros. -esimerkissä, tässä kuvassa on linnan taustalla linnan seinämä, kaksi tasoa sen edessä ja kolme erilaista objektia tasojen yllä tekstuuripaketista irrotettuna. Kuvassa 9 sprite-kuvat on tallennettu JPEG-tiedostomuotoon, joka ei tue kuvien läpinäkyvyyttä. Tällöin läpinäkyvä tausta on tallentunut automaattisesti valkoiseksi. JPEG on erityisesti valokuville soveltuva tiedostomuoto, joka on niin sanottua häviöllistä pakkausformaattia, eli alkuperäistä dataa voidaan muuttaa tiedoston koon pienentämiseksi. JPEG-tiedostoon voidaan tallentaa yli 16 miljoonaa väriä, mutta se ei tue minkäänlaista läpinäkyvyyttä. Tiedostomuotoa voidaan haluta käyttää peligrafiikassa, koska JPEG-tiedosto on kooltaan pienempi kuin vastaava PNG-tiedosto. Jos pelissä halutaan käyttää JPEG-grafiikkaa, sen tulisi olla kuvitusta, joka ei vaadi läpinäkyvyyttä. On myös syytä ottaa huomioon, että Unity Asset Store -palveluun ei hyväksytä kuvatiedostoja, jotka ovat häviöllisessä pakkausmuodossa.



KUVA 8. Taustoiltaan läpinäkyvät objektit taustan edessä.



KUVA 9. Peliobjektit ei-läpinäkyvillä taustoilla.

Digitaalisia kuvia tallennettaessa värien määrään vaikuttaa kuvan värisyvyys, joka kuvataan biteissä. Värisyvyydessä ilmoitettu bittimäärä ilmaisee, kuinka montaa erilaista väriä voidaan käyttää yhtä kuvassa esiintyvää pikseliä kohden. PNG-tiedostoissa käytetään 8- ja 24-bittistä värisyvyyttä käyttötarkoituksesta riippuen. Nämä luvut kuvaavat sitä, kuinka montaa eri väriä tai värisävyä tallennetussa tiedostossa on mukana. 8-bittinen PNG-kuvatiedosto voi sisältää enimmillään 256 värisävyä ja 24-bittisessä kuvassa voi olla 16 777 216 väriä. Kuvalle sopiva tallennusmuoto tulee valita kuvan värimäärän mukaan.

2.7 Peligrafiikan rajoitukset ja optimointi

Peligrafiikkaa suunniteltaessa tulee ottaa huomioon millaiselle pelialustalle peli toteutetaan, sillä jokaisella pelialustalla on omat rajoituksensa grafiikan suhteen. Esimerkiksi mobiilialustoille toteutettavat pelit ladataan pääsääntöisesti netin kautta, jolloin ladattavan paketin ei ole suotavaa olla liian isokokoinen. Lisäksi mobiilialustoilla on vähemmän muistia käytössä kuin pöytäkoneissa, jolloin liian suureksi paisuneet tai huonosti optimoidut grafiikat aiheuttavat pelin hidastelua ja kuvataajuuden pienenemistä. Muilla pelialustoilla on enemmän muistia, mutta tällöinkin liian isojen tekstuurien käyttämistä tulisi välttää, jotta pelit eivät hidastuisi turhaan. (Unity Optimization 2014.)

Unity-pelimoottori on rakennettu siten, että tekstuurikuvien leveys ja korkeus tulisi olla pikseleissä laskettuna kahden potenssissa, esimerkiksi 64×64 (2^6) tai 256×512 (2^8 ja 2^9) (Unity Art Asset Best Practice Guide 2014). Tällöin Unityn ei tarvitse lisätä tekstuurihin ylimääräistä tyhjää dataa saavuttaakseen halutun koon. Esimerkiksi kuva, jonka koko on 600×550 pikseliä, ei noudata kahden potenssin laskutapaa leveys- tai korkeussuunnassa. Tällaiseen kuvaan lisätään sen verran ylimääräisiä läpinäkyviä pikseleitä, että se täyttää kahden potenssin ehdot. Esimerkkikuva on leveämpi ja korkeampi kuin 512 pikseliä, joka on yhtä kuin 2^9 . Seuraava askel, 2^{10} , on yhtä kuin 1024 eli Unity laskee näin ollen kuvan lähes kaksinkertaiseksi.

Vertailun vuoksi voidaan laskea kuinka paljon edellä mainitut kuvat vievät muistia biteissä ennen pakkaamista. Aliluvussa 2.3 kerrottiin kuinka jokainen kuvassa esiintyvä pikseli saa värinsä neljästä eri komponenttiarvosta (punainen, vihreä, sininen, alfa eli läpinäkyvyys). Jokainen näistä arvoista sisältää yhden tavun verran tietoa, joten yksi pikseli on neljän tavun kokoinen. 1024×1024 pikselin kokoisessa kuvassa on yhteensä $4\,194\,304$ tavua, joka vastaa neljää megatavua, kun taas 600×550 kuva olisi vastaavasti $1,3$ megatavua eli noin 69% pienempi kuin suurennettu kuva.³ (Gullen 2013.) On myös huomattava, että häviöttömästi pakatun kuvan koko kiintolevyille tallennettuna ei vastaa sitä, kuinka paljon muistia kuva vie pelin aikana. Unity purkaa kuvan pakkauksen, jolloin kuvan koko on todellisuudessa edellisten laskelmien mukainen.

³ 8 bittiä = 1 tavu, 1024 tavua = 1 kilotavu, 1024 kilotavua = 1 megatavu.

Kuten luvussa 2.6 todettiin, sprite-kartat ja tekstuuriatlakset säästävät muistia. Pelejä pelattaessa kuvat ladataan niin kutsuttuun video ram -muistiin eli VRAM-muistiin joka on erillään tietokoneen käyttämästä RAM-muistista (Gullen 2013). Tähän muistiin tallennetut kuvat voidaan kutsua nopeammin näkyviin pelin sisällä. Muistia säästyy, jos samankaltaisista kuvista on tehty yksi isompi kuva useiden erillisten kuvien sijaan, jotka ohjelma kutsuu useissa eri instansseissa.

Kuvia ei tulisi myöskään tallentaa osittain läpinäkyvinä, jos siihen ei ole tarvetta, sillä tämä vie turhaan muistia. Jos kuvassa ei ole läpinäkyvyyttä, eroa ei huomaa paljain silmin, vaikka kuvaan jäisi läpinäkyvyydelle varattuja bittejä. Esimerkiksi sprite-kuvissa läpinäkyvyyttä tarvitaan, mutta taustakuvissa läpinäkyvyys voidaan jättää pois.

2.8 Digitaaliset työvälineet

Peligrfiikan tekemiseen voidaan käyttää monia erilaisia kuvankäsittelyohjelmia. Eräitä suosituimpia kaksiulotteisen grafiikan tekoon käytettyjä ohjelmia ovat yhdysvaltalaisen Adobe-yrityksen kuvankäsittelyohjelmat Illustrator ja Photoshop. Illustrator on vektorigrfiikkaan tarkoitettu ohjelmisto, kun taas Photoshopilla tehdään pääasiassa rasterigrfiikkaa. Adoben ohjelmat ovat maksullisia ja niitä käytetään pääasiassa ammattikäytössä.

Tunnetuimpia grafiikkaan tarkoitettuja ilmaisohjelmia ovat Inkscape ja GIMP. Inkscape on kehitetty vektorigrfiikan tekoon ja GIMP vastaavasti on tarkoitettu rasterigrfiikalle. Laajan käyttäjäkunnan ja vapaaehtoistoiminnan kautta edellä mainituilla ohjelmitoilla pystyy nykyisin tekemään samantasoisia jälkeä kuin ammattiohjelmistoilla. Tässä opinnäytetyössä tehty tekstuuripaketti on toteutettu Adobe Illustrator ja Photoshop-ohjelmien versioilla CS3.

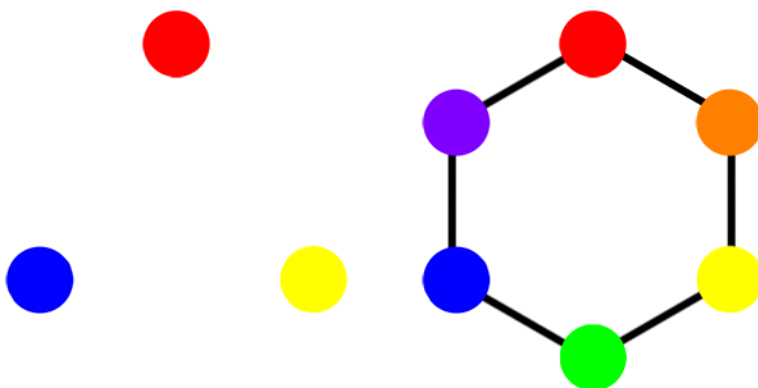
3 PELIEN VÄRIMAILMA JA ESTETIIKKA

Peligrafiikan värit ja tyyli vaikuttavat vahvasti pelin tunnelmaan ja pelitilanteen ymmärrettävyyteen. Tässä luvussa käsitellään erityisesti värioppia, joka on tärkeä osa peligrafiikan estetiikkaa ja josta ei löydy paljoa materiaalia tai ohjeistusta. Luvussa tarkastellaan peligrafiikan vaatimuksia käytettävyyden ja estetiikan kannalta.

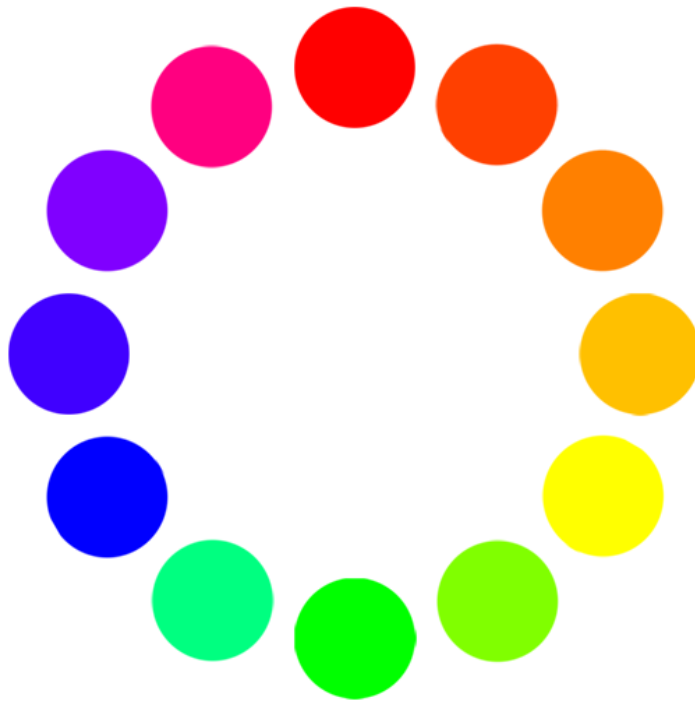
3.1 Värioppi peligrafiikassa

Peleissä esiintyvä värimaailma on harvoin syntynyt sattumalta. Silmämääräisesti voidaan nähdä, miten eri väriyhdistelmät sopivat paremmin yhteen kuin toiset. Ihmiset pysyvät yleensä kertomaan, että tarkasteltavan kuvan värivalinnoissa on jotain pielessä, vaikka he eivät osaisi sanoa tarkalleen, mitä kuvassa pitäisi muuttaa.

Väriopin sääntöjä noudattamalla voidaan toteuttaa esteettisesti miellyttäviä värikokonaisuuksia. Keskenään miellyttäviltä näyttäviä väriyhdistelmiä kutsutaan väriharmonioiksi. Väriharmonian suunnitteluun käytetään yleensä väriympyrää, joka noudattaa RYB-värimallin päävärejä eli punaista, keltaista ja sinistä. (Stone ym. 2006, 20.) Päävärejä sekoittamalla saadaan aikaan sekundaari- eli välivärejä: oranssia, vihreää ja violettiä (kuva 10). Päävärejä ja välivärejä voi edelleen sekoittaa keskenään, jolloin saadaan lisää tiettyyn päävärisuuntaan taittavia sävyjä (kuva 11).



KUVA 10. Pää- ja välivärit subtraktiivisessa RYB-värimallissa.



KUVA 11. Kahdentoista värin väriympyrä RYB-värimallissa.

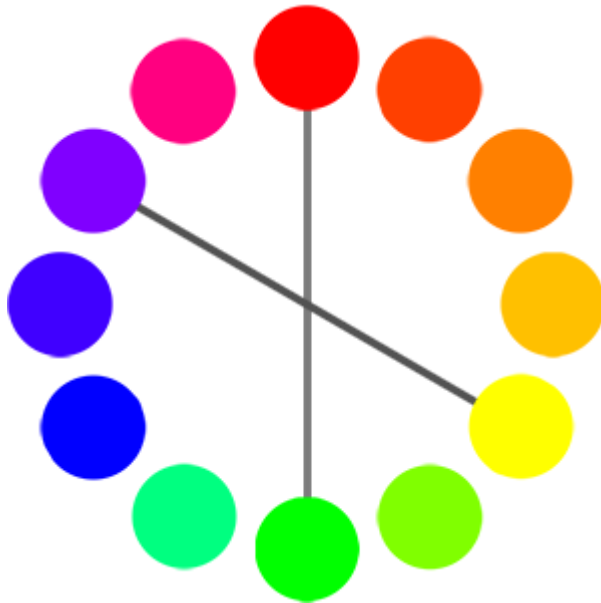
Väriharmonian valintaan liittyy ainakin kuusi sääntöä, joita voidaan pitää niin sanotusti aina toimivina väriyhdistelminä (Stone ym. 2006, 21). Väriharmoniassa valitaan keskenään hyvin sointuvia väriyhdistelmiä väriympyrän avulla. Väriharmonia tulisi valita tarpeen mukaan pelin tunnelman ja teeman mukaan. Kuusi väriharmoniasääntöä ovat vastavärit, jaettu vastaväri, vastaväripari, lähivärit, kolmisointu ja valööriharmonia.

3.1.1 Vastavärit

Väriympyrässä värit sijaitsevat vierekkäin ympyrän keskipisteen kautta ympyrän kehässä (kuva 11). Värin vastaväri löytyy piirtämällä väristä suora jana väriympyrän vastakkaiselle puolelle. Janan vastakkaiset päät ovat keskenään vastavärejä (kuva 12). Janan voidaan ajatella jakavan väriympyrän kahteen yhtä suureen osaan. Kuvaa 12 tarkasteltaessa voidaan todeta, että punainen ja vihreä ovat vastavärejä, samoin keltainen ja violetti.

Vastaväreillä saadaan aikaan voimakkain väriyhdistelmä. Vastaväriharmoniaa voidaan käyttää sellaisissa töissä, joissa halutaan tuoda esille voimaa, ristiriitaisuutta ja energisyyttä. (Stone ym. 2006, 21.) Väriympyrää tarkasteltaessa voidaan nähdä, että päävärin vastavärinä on aina jokin väliväri. Välivärin vastavärinä voi olla joko pääväri tai toinen

väliväri. Tästä voidaan päätellä, että vastaväripareiksi valituista väreistä enintään puolet ovat päävärejä.



KUVA 12. Vastavärejä.

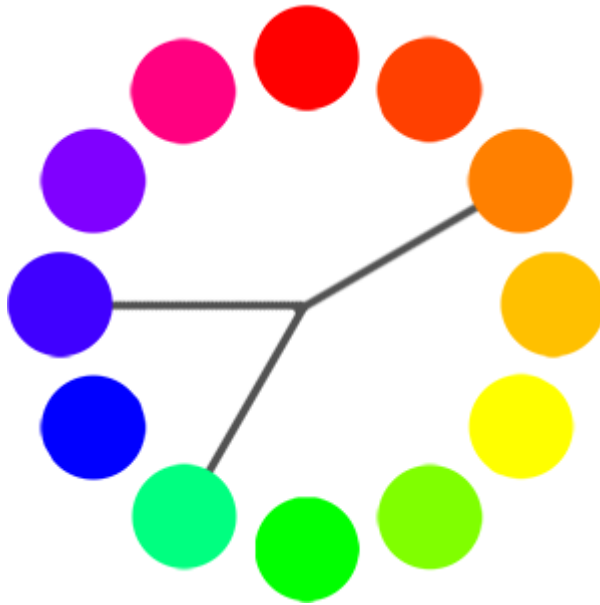
Tekstuuripakettiin tehty punainen taikajuomapullo käyttää vastaväreinä punaista ja vihreää (kuva 13). Taikajuomapullo on pelin kentältä kerättävä esine, joka ei saa jäädä pelaajalta huomaamatta, jolloin siihen tarvitaan voimakkaat värit. Punavihreä esine erottuu taustasta ja näyttää esineeltä, joka ei kuulu taustaan.



KUVA 13. Taikajuomapullo vastaväreillä.

3.1.2 Jaettu vastaväri ja vastaväripari

Vastavärien keskinäistä ristiriitaisuutta voidaan keventää valitsemalla värille tämän vastaväriin kummallakin puolella sijaitsevat värit (Stone ym. 2006, 21). Kuvan 14 esimerkissä näkyy kuinka tällainen jako tapahtuu. Kolmen värin yhdistelmä ei ole niin aggressiivinen kuin vastavärit ja tuloksena on sovittelevampi väriyhdistelmä.



KUVA 14. Jaettu vastaväri.

Kun kuvaan halutaan sisällyttää monia eri sävyjä, voidaan yhden vastaväriparin sijasta valita kaksi vastaväriparia. Neljän keskenään ristiriidassa olevan värin käyttö voi olla ongelmallista. Näin isolla hajonnalla sijaitsevat värit eivät välttämättä sovi yhteen toistensa kanssa, etenkin jos kaikkia värejä käytetään samassa suhteessa (Stone ym. 2006, 21). Eri värien keskinäistä määrää muuttamalla voidaan saada aikaan värikäs ja mielenkiintoinen kokonaisuus, mutta oikeanlaisen lopputuloksen saaminen voi vaatia paljon värisuhteiden testaamista kuvassa. Vastaväriparin ristiriitaisuutta voidaan murtaa lisäämällä kuvaan yhtä tai useampaa väriä.

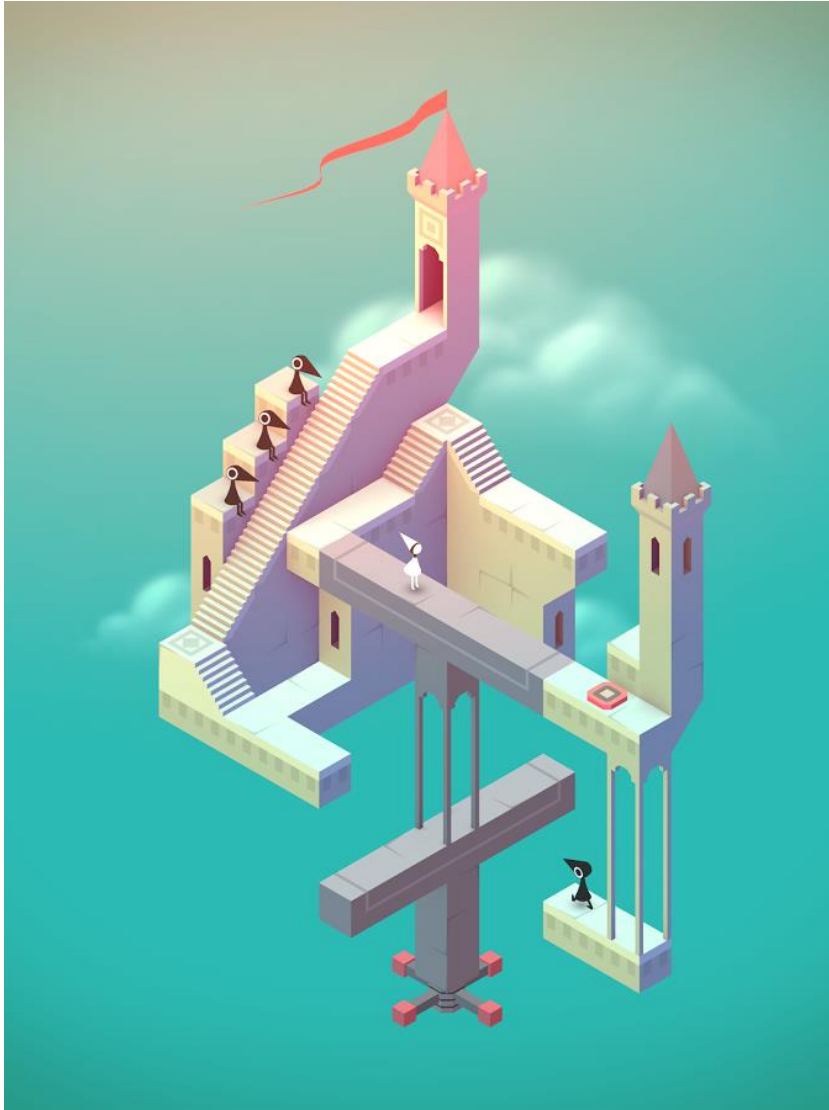
Banner Saga -pelin välianimaatiossa vuodelta 2013 nähdään, kuinka vastaväripareja on käytetty visuaalisena tehokeinona (kuva 15). Kahdella hahmolla on selkeä vihreän ja punaisen vastaväripari viitoissaan. Jousiampujan kirkkaan oranssit hiukset ja puiden syvä sininen ovat keskenään toinen vastaväripari. Näillä väreillä etualan hahmot ja objektit on erotettu selkeästi taustalla näkyvistä asioista.



KUVA 15. Banner Saga (Stoic 2013).

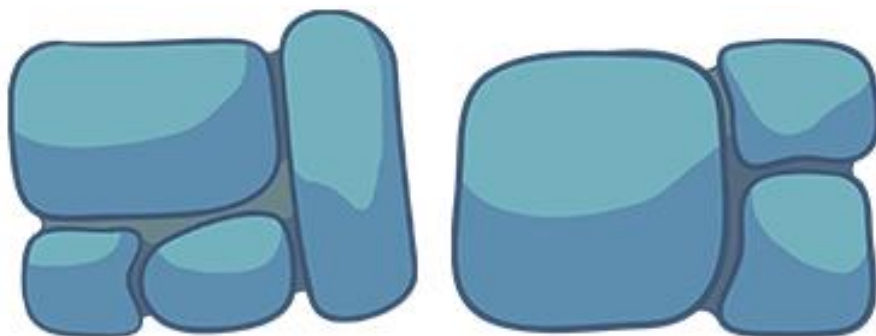
3.1.3 Lähivärit

Edellä mainittuja väriharmonioita voidaan käyttää voimakkaita tunteita nostattaviin kuvakokonaisuuksiin. Nämä väriharmoniat eivät kuitenkaan sovi peleihin, joissa esimerkiksi halutaan herättää levollisia ja rauhallisia tuntemuksia. Lähiväriharmonia voi olla sopiva valinta tällaiseen peliin. Värit valitaan väriympyrästä siten, että ne ovat väriympyrän vierekkäisiä sävyjä (Stone ym. 2006, 21). Esimerkiksi kuvassa 16 lähivärejä ovat violetti, punertava violetti ja punainen.



KUVA 16. Monument Valley (Ustwo 2014).

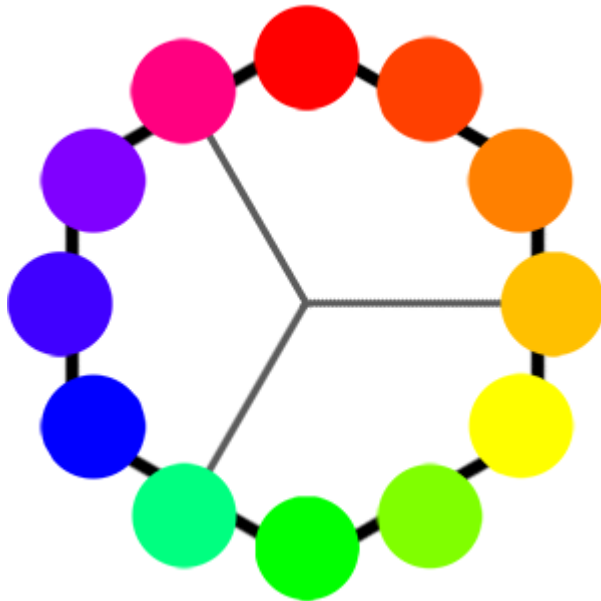
Värien määrässä tulisi pitää huolta siitä, ettei liian monien värien käyttäminen aiheuta kuvan tukkoistumista tai värien välistä tahatonta ristiriitaisuutta. Tekstuuripaketissa olevissa kivitasoissa (kuva 17) ja taustakuvassa käytetään lähivärejä, koska kivisen aineksen halutaan vaikuttavan neutraalilta ja hillityltä.



KUVA 17. Kivitasojen värit ovat lähivärejä.

3.1.4 Kolmisointu ja valööriharmonia

Mielenkiintoisia väriyhdistelmiä voidaan saada aikaan yhdistämällä väriympyrän kolmea toisistaan yhtä kaukana olevaa väriä (Stone ym. 2006, 21). Väriympyrässä näistä kolmesta väristä piirretyt kuvitteelliset janat leikkaisivat väriympyrän kolmeen yhtä suureen osaan (kuva 18). Koska nämä kolme väriä sijaitsevat yhtä kaukana toisistaan, ne voivat olla joko päävärejä tai välivärejä. Pelkkien päävärien valitseminen voi saada kuvan näyttämään naivistiselta tai epätasapainoiselta. Väliväreillä värit ovat keskenään sopuisammassa asemassa. Kuvankaappaus pelistä *The Wolf Among Us* vuodelta 2014 (kuva 19) käyttää kolmisointua kuvan 18 esimerkin mukaan. Vaaleanpunaista ja oransiin taittuvaa keltaista väriä on käytetty hahmoon lankeavana valona. Kirkasta syaania on käytetty vain muutamassa kohdassa kiinnittämään huomiota taustaan. Kuvan varjoissa vaaleanpunaiseen on lisätty sinistä rikkomaan monotonisuutta.



KUVA 18. Kolmisointu.



KUVA 19. The Wolf Among Us - Episode 2 (Telltale Games 2014).

Edellä mainitut väriharmoniat pohjautuivat kahteen tai useampaan väriin. Valööriharmoniassa värejä on vain yksi. Värimaailma toteutetaan valitun värisävyn tummuutta ja kirkkautta vaihtelemalla, esimerkiksi Limbo-pelin painostava värimaailma on toteutettu mustan eri sävyillä (kuva 20). Lopputulokseksi saadaan hyvin rauhallinen ja unenomainen värimaailma.



KUVA 20. Limbo (Playdead 2010).

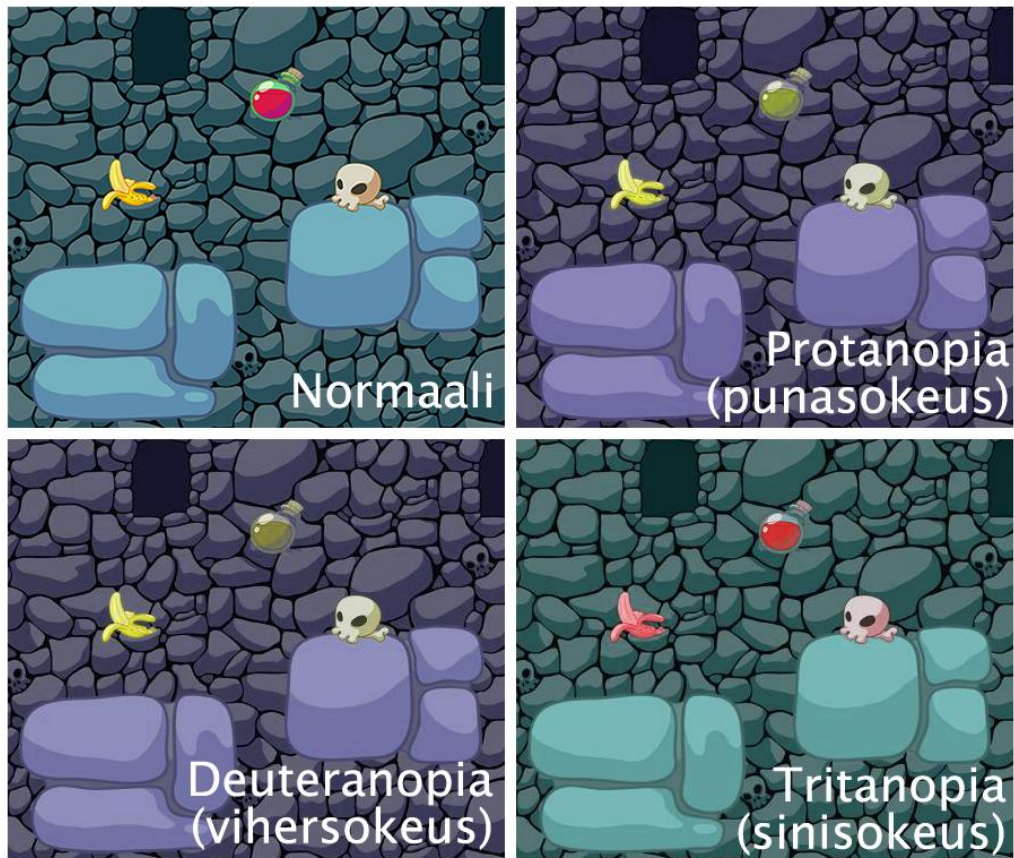
3.2 Värien hahmottaminen

Ihminen havaitsee reaali maailman asioita yksilökohtaisesti. Erilaiset näön ja värin näkemisen heikentymät aiheuttavat sen, että kaksi eri pelaajaa eivät välttämättä havaitse ruudulla näkyviä pelielementtejä samalla tavalla. Pahimmassa tapauksessa pelaaja voi olla huomaamatta pelin pelaamiseen tarvittavia tärkeitä pelillisiä elementtejä, jos hänellä on esimerkiksi värisokeus tai värinäön heikentymä.

Värit voivat näkyä pelissä ei-toivotulla tavalla myös ihmisille, joiden värinäkö on moitteeton. Silmiä on helppo huijata, sillä värit vaikuttavat toistensa rinnalla oleviin väreihin, jolloin kaksi samaa väriä voivat näyttää eri väreiltä taustasta riippuen. Silmät voivat niin sanotusti muistaa värejä ja niiden haamukuvia väärin. (Kane 2014, 299; Stone ym. 2006, 5253.)

Tekstuuripaketin elementtejä on testattu värinäön eri heikentymien kautta (kuva 21). Esimerkkikuva on syötetty erilaisia värinäön heikentymiä simuloivaan Coblis-verkkosovellukseen⁴. Puna- ja vihersokeus ovat yleisimmät värinäön heikentymät ja sinisokeus eli tritanopia on näitä harvinaisempi. Kuvan esimerkeistä voidaan nähdä, että pelikentältä kerättävät banaani ja taikajuomapullo ovat väreiltään lähellä toisiaan. Esineet eroavat toisistaan kirkkaudeltaan ja muodoiltaan.

⁴ <http://www.color-blindness.com/coblis-color-blindness-simulator/>



KUVA 21. Tekstuurikuvia havainnollistettuna eri värinäön heikkenemisen ilmentymille.

Pelin sisältä löytyviä tärkeitä elementtejä tulisikin visualisoida myös muilla keinoilla kuin pelkästään väreillä. Esimerkiksi pelin käyttöliittymässä eri värit voidaan nimetä, jos ne on tarpeen erottaa toisistaan (Broome 2014). Jos pelaajan tulisi esimerkiksi tunnistaa kaksi eri värein luokiteltua joukkuetta toisistaan, joukkueiden värien nimet voidaan esittää sanoina pelin käyttöliittymässä.

Eri funktioita edustavat objektit voivat olla uniikin muotoisia, jolloin esineiden tunnistaminen ei olisi kiinni pelkästään väreistä. Edellisessä esimerkissä mainitut banaani ja juomapullo ovat erimuotoisia, mutta jos peliin haluttaisiin kahdenlaisia juomapulloja, ne tunnistetaan helpoiten erilaisiksi muuttamalla muotoa pelkän värin sijaan. Tässä paketissa on kahden muotoisia juomapulloja tunnistettavuuden vuoksi. Värien kontrasteja voi myös muuttaa, jolloin tummemman ja vaaleamman väriset esineet eroavat selvemmin toisistaan (Stone ym. 2006, 66).

4 UNITY-PELIMOOTTORI

Unity Asset Store -palvelussa tarjotaan valmista materiaalia Unity-pelimoottorille. Tämän vuoksi on hyvä tuntee Unitysta perusteet, jotta työstettävät materiaalit voidaan tehdä mielekkäiksi käyttää ja jotta niitä voisi käyttää tarvittaessa sellaisenaan peliprojektissa. Tässä luvussa käsitellään lyhyesti Unitya pelimoottorina ja esitellään Unity Asset Store -palvelua.

4.1 Unity 3D ja Unity 2D

Unity 3D eli Unity on pelimoottori, joka tarjoaa valmiita työkaluja peliohjelmointiin. Pelimoottori on tarkoitettu pääasiassa kolmiulotteisten pelien ohjelmoimiseen. Unity 3D:stä on olemassa ilmainen, ominaisuuksiltaan karsittu versio ja maksullinen, yritysten käyttöön tarkoitettu täysversio. Vuoden 2015 helmikuussa uusin versio Unitysta on 4.6.2. Tämä versio sisältää vuonna 2014 mukaan liitetyn Unity 2D -ympäristön, joka tukee kaksiulotteisten pelien tekemistä. Unityssa ei ollut aiemmin valmiita työkaluja kaksiulotteisten pelien kehittämiseen.

Unitylla on toteutettu paljon kaupallisia pelejä, muun muassa inXile Entertainmentin Wasteland 2. Pelissä on hyödynnetty kolmiulotteisia hahmoja ja ympäristöjä, ja pelin kuvakulma lukittu yläviistosta katsottavaksi. Kaksiulotteista grafiikkaa käytetään pääasiassa pelin käyttöliittymässä ja kolmiulotteisten mallien tekstuureissa (kuva 22).



KUVA 22. Wasteland 2 (InXile Entertainment 2014).

Wasteland 2 -pelin kehittäjät Brian Fargo ja Chris Keenan kertoivat omilla tahoillaan kahden eri pelisivuston haastatteluissa päätöksestä toteuttaa peli Unitylla Unreal Engine -pelimoottoriin sijaan, jota oli käytetty aiemmissa InXile Entertainmentin peliprojekteissa. Fargo kuvaili Unitya oikeanlaiseksi sekoitukseksi ominaisuuksia ja sellaiseksi alustaksi, joka mahdollistaa pelinkehittämisen maailmanlaajuisella työpanostuksella. Unitylle on paljon ohjelmointiosaamista omaavia pelinkehittäjiä, joilta voi ostaa osaamista maailmanlaajuisesti. Keenan kommentoi, että Unityn valintaan liittyvä suuri tekijä oli Unity Asset Store. Hän suositteli pelintekijöille Unity Asset Storen sisällön hyödyntämistä ja totesi InXile Entertainmentin säästäneen valmiita materiaalipaketteja käyttämällä 200 000 dollaria ensimmäisten työkuukausien aikana. (Batchelor 2014; NowGamer 2012.)

InXile Entertainment järjesti vuonna 2012 yhteistyössä Unity Technologies -yrityksen kanssa peligraafikoille mahdollisuuden osallistua Wasteland 2 -pelin kehittämiseen jakamalla Unity Asset Storen kautta Artist Starter Kit -pakettia, jota hyödyntämällä käyttäjät pystyivät suunnittelemaan peliin tekstuureita, 3D-malleja ja 2D-grafiikkaa rahallista korvausta vastaan.

4.2 Unity Asset Store -palvelu

Unity tarjoaa pelinkehittäjille paljon ohjelmallisia työkaluja nopeuttamaan pelin toteuttamista. Se ei kuitenkaan tarjoa kattavaa graafista kirjastoa ja se sisältää vain rajatun valikoiman valmiita skriptejä. Tätä varten on perustettu Unity Asset Store -palvelu, jonka tarjontaa voi selata sekä selaimesta että pelimoottorin sisäisestä kaupanäkymästä.

Unity Asset Store tarjoaa käyttäjien luomia valmismateriaaleja, kuten esimerkiksi tekstuureja, malleja, animaatioita ja valmiita skriptejä tai liitännäisiä. Materiaaleja voi laittaa palveluun ladattavaksi joko ilmaiseksi tai maksullisina. Palvelu veloittaa myytyjen materiaalipakettien hinnasta 30 %. Loput 70 % maksetaan käyttäjälle kerran kuussa Paypal-palvelun kautta. (Unity Asset Store 2014.)

4.3 Tekstuuripaketin vaatimukset

Kuten luvussa 1 todetaan, tekstuuripakettiin tehtyjen kuvien on tarkoitus olla käytettäviä sekä tietokoneille että mobiililaitteille tehdyissä peleissä. Näin ollen kuvia on hyvä olla useampaa kokoa: korkearesoluutioisia kuvia niitä tukeville laitteille ja pakkauskooltaan pienempiä kuvia laitteille, joissa ei ole yhtä paljon muistia käytössä. Kuvien koko noudattaa kahden potenssin sääntöä, koska muutoin Unity-pelimoottori lisää tekstuureihin ylimääräistä muistia vieviä tyhjiä pikseleitä, kunnes tekstuurin koko täyttää toisen potenssin säännön. Lopulliset tekstuurikartan yksittäiset kuvaosiot ovat 128 x 128 pikseliä, 256 x 256 pikseliä ja 512 x 512 pikseliä.

Kuvat tallennetaan PNG-tiedostoiksi luvussa 2.6 esitettyjen ominaisuuksien vuoksi. PNG-tiedostomuoto tukee osittaista läpinäkyvyyttä, on häviämätöntä pakkausmuotoa ja on oletuksena tuettu Unity 2D:ssä. Lisäksi Unity Asset Storen ohjeistuksessa vaaditaan tekstuurikuvien tallentamista häviämättömään pakkausmuotoon (Unity Submission Guidelines 2014). Yksittäiset kuvat ryhmitellään kokonaisuuksiksi ja tallennetaan tekstuuriatlakseksi tilan ja muistin säästämiseksi. Animaatioille tehdään omat tekstuuriatlaksen kaltaiset kuvakokonaisuutensa.

5 TEKSTUURIPAKETIN SISÄLTÖ

Tekstuuripaketin työstäminen alkoi paketin sisällön ideoinnilla, luonnoksilla ja paketin sisällön määrittelyllä. Ideoinnissa käytiin läpi useampi erilainen vaihtoehto paketin teemalle, ennen kuin lopulliseen ratkaisuun päädyttiin. Tässä luvussa esitellään tekstuuripaketin sisältö perusteluineen.

5.1 Konsepti

Tekstuuripaketin suunnittelu alkoi asettamalla sille selkeät kriteerit. Unity Asset Stores- sa tarjotaan enimmäkseen kolmiulotteisiin peleihin tarkoitettua materiaalia, sillä kirjoittamishetkellä Unity 3D on ollut saatavilla kauemmin kuin Unity 2D. Kaksiulotteiselle pelille tarkoitettulle materiaalille on tarvetta, joten tällaisen tekstuuripaketin toteutus on perusteltu vaihtoehto.

Kaksiulotteisiin peleihin tarkoitettu grafiikka listattiin Unity Asset Storessa “Textures & Materials”-osion alaisuudessa vielä kirjoittamishetkellä lokakuussa 2014, eli sille ei ollut omistettu omaa osiota. Kaksiulotteisille peleille tarkoitettuja paketteja tarkasteltaessa voitiin todeta, että grafiikkaa oli tehty pääasiassa tasoloikkapeleille (engl. *platformer game*), joita oli suosituimpien pakettien listauksesta useita kappaleita (kuva 23 ja 24). Päätös toteuttaa peligrafiikkaa tasoloikkapeleille perustuu jo myytyjen grafiikkapakettien suosioon. Paketin teeman haluttiin kuitenkin erottuvan jo olemassa olevasta tarjonnasta.



KUVA 23. 2D Forest Pack (RegnarForge 2014).



KUVA 24. 2D Games Assets Pack Vol 1 (Creepy Cat 2014).

Tekstuuripaketin esteettiseksi temaksi valikoitui fantasialinnan sisättila. Unity Asset Storen tekstuuripaketeista suurin osa sijoittui ulkoilmaan, yleensä metsään. Lopullinen tekstuuripaketti sijoitettiin fantasialinnan sisätilaan, joka on pidetty mahdollisimman yleisluontoisena, jotta grafiikoita voitaisiin käyttää erilaisissa fantasiateemaisissa peleissä. Tekstuurit suunnitellaan sivulta katsottavaan kaksiulotteiseen tasoloikkaan, jonka tekstuuripaketti sisältäisi taustoja, päältä käveltäviä tasoja, peliobjekteja eli pelin sisällä olevia elementtejä tai tavaroita, jotka ovat pelihahmon käytettävissä, ympäristöön liittyviä vaaratekijöitä ja koristeita.

Tärkeimmäksi visuaaliseksi elementiksi oli valittu selkeys, joten paketin visuaalinen ilme noudattaa sarjakuvamaista tyyliä. Erityisesti mobiilipelien kuvien on hyvä olla yksinkertaisia ja helppoja hahmottaa pienessäkin koossa. Vaikka mobiililaitteille voi

tehdä myös realistisen näköisiä tai yksityiskohtaisia sprite-kuvia, niitä voi olla haastava saada näyttämään yhtä selkeältä kahdella erikokoisella resoluutiolla. Kuvassa 23 oleva tekstuuripaketti on esimerkki yksityiskohtaisesta, realismiin taittuvasta grafiikasta, jossa käytetään murrettuja värisävyjä. Kuva 24 puolestaan sisältää värikylläistä, naivistista ja selkeää peligrafiikkaa.

Koska visuaalinen ilme tavoittelee sarjakuvamaista tyyliä, paketissa esiintyvillä tärkeillä peliobjekteilla on selkeät taustaelementeistä erottuvat ääriviivat. Kuvassa 25 on esimerkkinä Ruukki Miner -peliin tekemäni aarrekstuuri, joka toimii mallina tekstuuripaketin kuvien visuaaliselle ilmeelle. Tässä kuvassa ääriviivat on sävytetty kuvan varsinaisen täytevärin mukaan tummankeltaiseksi. Ääriviivojen paksuus vaihtelee yksityiskohtien ja oletetun valonlähteen mukaan.



KUVA 25. Käyttämätön aarrekuva Ruukki Miner -mobiilipelistä.

Pakettia varten listattiin sellaisia tekstuureita ja sprite-kuvia, jotka ovat tasoloikkapelille keskeisimpiä. Kaksiulotteisessa tasoloikassa voidaan olettaa olevan kaksi tai useampi päällekkäin olevaa kuvatasoa (engl. *layer*), joita kutsutaan myös tasoiksi. Näistä tasoista ensimmäinen on tausta, joka ei sisällä pelin kannalta oleellisia elementtejä, vaan se toimii tapahtumapaikan ja tunnelman visualisoijana. Toinen taso on taustan edessä oleva pelialue, johon sijoitetaan pelihahmo sekä varsinaiset pelielementit. Etualan tasot joiden päällä pelihahmo kulkee (engl. *platform*) ovat pelillisiä elementtejä, joista peligenren

nimi juontaa juurensa. Luvussa 2.5 esitellään etualan ja taustan eroa pelitilanteessa (kuva 5 ja kuva 6).

Tekstuuripakettia suunniteltaessa oletetaan, että pelin taustakuva muodostuu useammasta kuvasta eli tiilestä, joita voi yhdistellä isommiksi kokonaisuuksiksi saumattomasti. Taustan voisi halutessaan toteuttaa myös isona kuvana, joka ladataan pelilaitteen muistiin, mutta tämä kuormittaa muistia huomattavasti enemmän kuin pienempiin kuviin jaettu kuvatiedosto, eikä se näin ollen välttämättä sopisi kaikkiin mobiililaitteisiin. Taustan toimintatapaa ja toteutusta kuvataan luvussa 6.2.

Tyypillisessä tasoloikassa pelihahmo kerää kentältä löytyviä aarteita tai objekteja. Tekstuuripaketin kerättäväksi objekteiksi valittiin kultakolikoita, erilaisia hedelmiä sekä sekalaisia aarteita. Lista kaikista tekstuureista on luettavissa luvusta 5.2.

Fantasiaelementtejä sisältävät pelit voivat olla värimaailmaltaan erilaisia. Luvussa 3 käsiteltyjen esimerkkipelien kautta nähdään kuinka värit vaikuttavat sekä pelin tunnelmaan että yleisilmeeseen. Sarjakuvamaiseen tyyliin sopivat kirkkaat, selkeät värit, joita voi käyttää myös fantasiaateemaan.

5.2 Käytetyt työkalut

Tekstuuripaketin tekoon on käytetty Wacom Bamboo Fun Pen & Touch -piirtopöytää (engl. graphic tablet). Piirtopöydällä piirrosjälki on tarkempaa kuin esimerkiksi hiirellä. Pöytä tunnistaa, miten voimakkaasti kynällä painetaan tunnistuspintaa vasten. Paineentunnistuksella piirto-ohjelmisto määrittää, kuinka paksu piirretty viiva on tai kuinka voimakkaasti väri kuultaa läpi. Tällöin puhutaan paineentunnistuksen tasosta, esimerkiksi Bamboo Fun Pen & Touch -kynän paineentunnistus on 1024 tasoa, mikä on riittävä määrä tällaiselle tietokonegrafiikalle. Ammattikäytössä olevissa piirtopöydissä paineentunnistus voi olla parhaimmillaan 2048 tasoa. Mitä korkeammalla tasolla paineentunnistus on, sitä herkemmin piirtopöytä tunnistaa piirtokynän painetta.

Sprite-kuvat on luonnosteltu joko Adobe Photoshopilla tai perinteisesti kynällä paperille. Paperille tehdyt luonnokset on skannattu korkealla resoluutiolla suoraan Adobe Photoshopiin ja Illustratoriin muokattaviksi. Kuvien reunaviivat on tehty Adobe Illustratiorilla luonnosten mukaan, jotta viivat näyttäisivät mahdollisimman hyvältä myös suurentaessa isoimpiin kokoihin. Kuvat on lopuksi väritetty Adobe Photoshopilla, jossa tehtiin myös tarvittavat retusoinnit. Käyttökokemukset ja näiden ohjelmistojen osaaminen vaikuttivat piirto-ohjelmien valintaan tässä projektissa.

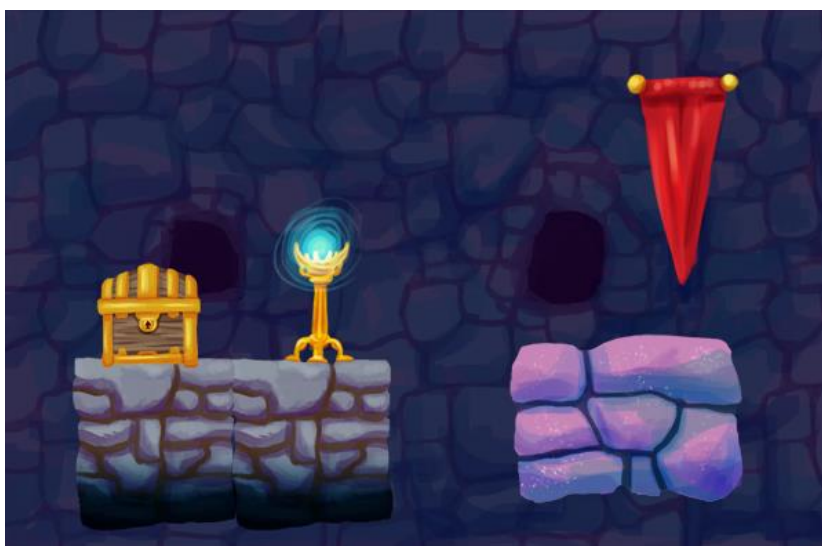
6 TEKSTUURIPAKETIN TOTEUTUS

Tekstuuripaketin sisällön ja teeman suunnittelusta siirryttiin varsinaisten kuvien toteuttamiseen edellisessä luvussa esitellyillä menetelmillä ja vaatimuksilla. Toteuttaminen alkoi luonnoksilla ja eteni askel kerrallaan kohti valmista kuvamateriaalia.

6.1 Suunnittelu- ja luonnosvaihe

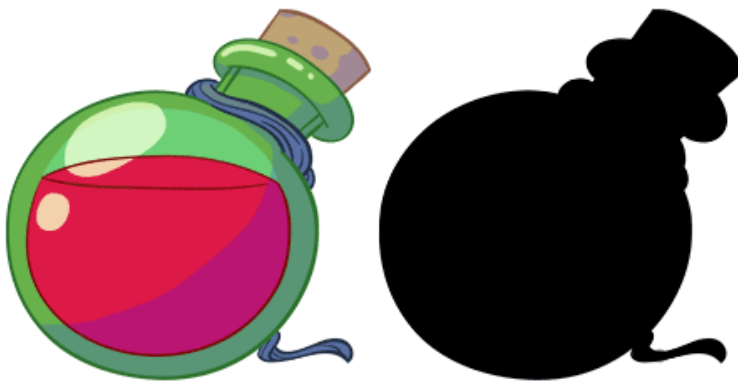
Tekstuuripaketin sisältö suunniteltiin listaamalla ne graafiset elementit, joita tasoloikka-peleissä saatetaan tarvita. Tekstuuripakettiin ei sisällytetty käyttöliittymään liittyvää grafiikkaa, koska projekti olisi muuttunut liian laajaksi. Tekstuuripaketin sisällön suunnitelma on esitelty liitteessä 1.

Luvussa 5.1 esiteltiin tyyli, jolla tekstuurit toteutettaisiin. Pelien värimaailmaa, pelimaailmaa ja tunnelmaa suunniteltaessa on tapana työstää niin sanottuja konseptikuvia, jotka toimivat eräänlaisena visuaalisena ohjenuorana. Niiden ole tarkoitus kuvata pelin lopullista ilmettä (Anhut 2014). Fantasiaalinnaa suunniteltaessa tutkittiin erilaisia väri vaihtoehtoja, joiden pohjalta luonnosteltiin nopea kuva siitä, mihin suuntaan tekstuuripaketin kuvia työstettäisiin (kuva 26).



KUVA 26. Tekstuuripaketin konseptikuva.

Sarjakuvamaiset kuvat haluttiin koostaa yksinkertaisista muodoista, jotta objektien visuaalinen ilme noudattaisi ei-realistista, helposti lähestyttävää ja humoristista ilmettä. Chris Solarski esittelee käsitteen *shape spectrum of emotions*, joka voidaan suomentaa *tunnetiloja kuvaavaksi muotojen spektriksi*. Tällä spektrillä sijaitsee kolme perusmuotoa: ympyrä, neliö ja kolmio. Näistä muodoista ympyrää voidaan pitää viattomuutta, nuoruutta ja energiaa kuvaavana muotona. (Solarski 2013, 1.) Pyöreät muodot ja kovien, terävien reunojen välttäminen toistuvat kaikissa tekstuuripaketin kuvissa (kuva 27 ja kuva 28). Peliobjekteja voidaan tarkastella niiden siluettien perusteella. Kuvan aihe tulisi olla tunnistettavissa yksivärisestä siluettista.



KUVA 27. Taikajuomapullo ja siluetti.



KUVA 28. Kerättävä aarre ja siluetti.

Kuten Stone, Adams ja Morioka (2006, 41) opastavat väripaletin kokoamisessa, värikoisuuden valinta on alkanut valitsemalla yksi keskeinen väri jonka kautta muut värit valitaan. Tämä valinta tapahtui pelin hallitsevimman visuaalisen elementin kautta. Tässä tekstuuripaketissa tausta tulee olemaan hallitseva osa pelinäkymää, joten sen väri tulee määrittelemään muiden värien valinnat.

Taustakuva koostuu kiviseinästä, jonka väriksi on haluttu siniseen taittuva harmaa, kuten kuvassa 29 esitetään. Alun perin värityksen piti seurata konseptikuvan puhtaampaa sinistä, mutta tätä väriä kokeiltaessa se osoittautui liian levottomaksi pelin taustan osana. Lisäksi tausta ei erottunut tarpeeksi hyvin edessä olevista elementeistä, kuten luvussa 6.4 kerrotaan.

Täydellisen harmaita värejä sekä puhdasta mustaa ja valkoista tulisi välttää värejä valittaessa, sillä ne latistavat helposti kuvaa. Esimerkiksi kuvassa 30 taustan kiviseinä koostuu pelkästään harmaista väreistä. Etualan tasot ja esineet erottuvat selkeästi taustasta, mutta saturaation eli värin kirkkauden erot ovat liian suuret etualan ja taka-alan välillä. Eräs keino tasapainottaa tällaista tilannetta on häivyttää etualan objektien saturaatiota. Kuvassa 31 nähdään tällaisen korjauksen tulos: kuva on harmonisempi ja mukavampi katsoa, mutta tunnelma on muuttunut kuvasta 29 sumuisemmaksi, rauhallisemmaksi ja tomuisemmaksi. Tällaista värimaailmaa voisi esimerkiksi hyödyntää realistisempaan tyyliin suuntautuviissa peleissä tai jos pelille suunniteltu tunnelma vaatisi sitä. Tekstuuripaketin lopullinen värimaailma noudattaa kuvan 29 mallia.



KUVA 29. Esimerkki pelinäkömästä tekstuuripaketin kuvilla.



KUVA 30. Esimerkki pelkästään harmaan sävyillä väritetystä taustasta.



KUVA 31. Etualan objektien saturaatio on korjattu vastaamaan taustan väritystä.

Kuvien valoissa ja varjostuksissa ei myöskään tulisi käyttää mustaa, valkoista tai harmaata, ellei sille ole tyyllillisesti perusteltua syytä. Värien tummuutta ja valoisuutta li-

säämällä kuva menettää eloisuuttaan ja voi saada kuvan näyttämään muoviselta. Kaikki pinta, jossa nähdään väriä, heijastaa jollain tavalla ympäröivää valoa ja esineitä. Jos esimerkiksi pelissä olisi vihreitä neonvaloja, ympäristön objektit heijastaisivat tätä valoa. Samalla tavalla reaali maailmassa ympärillä havaittavissa kohteissa varjot eivät ole täysin mustia. Esineiden varjojen väreihin vaikuttavat niihin osuvan valon väri ja muista esineistä heijastuvat värit.

Eräs tapa testata erilaisia väriharmonioita on kokeilla Internetistä löytyviä työkaluja. ColorHexa-sivustossa⁵ olevaan värihakuun voi syöttää värin heksanumeron, jolloin sivusto ilmoittaa värin tekniset tiedot, esimerkiksi RGB- ja CMYK-arvot. Sivusto laskee haetulle värille matemaattisesti valmiita väriharmonioita. Tekstuuripaketin värimaailmat on suunniteltu tämän palvelun avulla erilaisia väriyhdistelmiä vertailemalla. Kuvissa pyrittiin pitämään värit melko vapaana harmaan sävyistä taustaa lukuun ottamatta, johon haluttiin luoda kontrastia etualaan nähden.

6.2 Taustatekstuurien toteutus

Tekstuuripaketin valmistus aloitettiin taustalla näkyvien tekstuurien eli taustan toteuttamisesta. Tausta koostuu erilaisista neliön muotoisista tekstuureista, jotka sopivat saumattomasti yhteen. Näin pienemmillä kuvilla saadaan luotua isompia kokonaisuuksia. Halutessaan pelin ohjelmoija voi tällaisten taustatiilien avulla luoda taustaelementtejä, jotka näyttävät erilaisilta tilanteista ja kohteista riippuen. Kuvassa 32 esitellään neljä saumatonta tekstuuria, jotka yhdessä muodostavat isomman kokonaisuuden. Kuvia voi järjestellä eri tavoin toistensa viereen ilman, että pelaaja huomaa taustan muodostuvan pienemmistä kuvista.

⁵ www.colorhexa.com

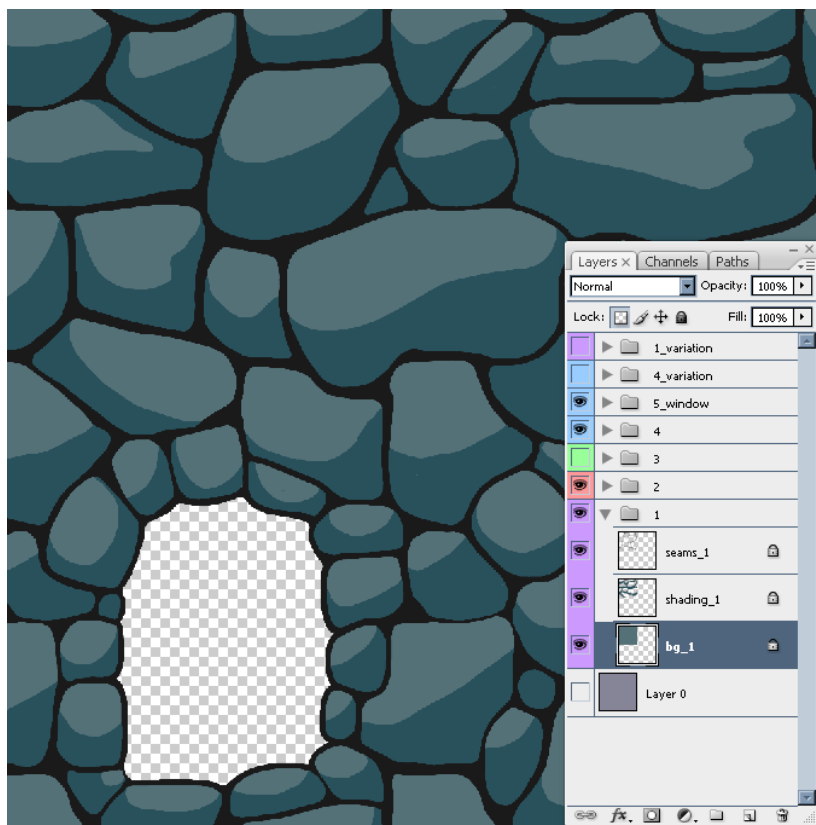


KUVA 32. Esimerkki saumattomista tekstuureista.

Taustatekstuurit sijaitsevat taustalla eikä niiden ole tarkoitus olla hallitsevassa roolissa. Näin ollen niille ei tehdä samanlaisia reunaviivoja kuin tekstuuripaketin merkittävimmille objekteille, esimerkiksi kerättäville esineille. Reunaviivojen sijaan kivien välistä näkyvät saumat on sävytetty hyvin tummaksi siniseksi, joka toisintaa reunaviivamaista tyyliä. Kuvassa 32 esiintyvät saumattomat tekstuurit ovat esimerkkejä paketin lopullisista tekstuureista.

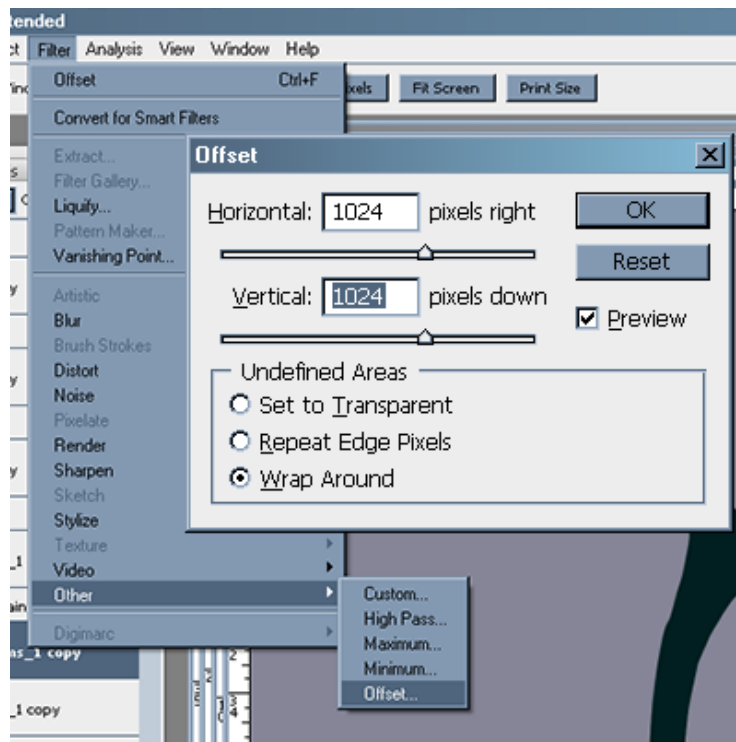
Tekstuurien luonnostelu tehtiin Adobe Photoshop -ohjelmassa. Kiviseinästä kuviota hahmoteltiin melko ylimalkaisesti, jotta saataisiin nopeasti käsitys siitä, miltä kiviseinän halutaan näyttävän. Yleisilmeeksi muodostui epäsymmetrinen, isoista kivilohkareista muurattu seinä, joiden väliin on jäänyt pienempiä murenevia lohkareita. Yhden tekstuurikuvan haluttiin sisältävän osittain tai kokonaan läpinäkyvän ikkunan.

Kuvan taustaväri ja muuraussauma pidettiin erillisinä omissa tasoissaan, jotta kuvien muokkaaminen olisi myöhemmin helpompaa, esimerkiksi jos värejä halutaan myöhemmin muuttaa tai korjata. Jokainen kuvan taso lajiteltiin kuvaavasti nimetyn kansion alle Photoshopin näkymässä, jotta ne olisi helposti löydettävissä myöhemmin. On tärkeää huomata, että tasot tulisi nimetä mahdollisimman kuvaavasti, sillä kuvassa 33 esiintyvän taustatekstuurin tiedostossa oli 24 erilaista tasoa. Muokattavien taustojen löytäminen muiden seasta voi olla haastavaa ilman johdonmukaista lajittelujärjestelmää.

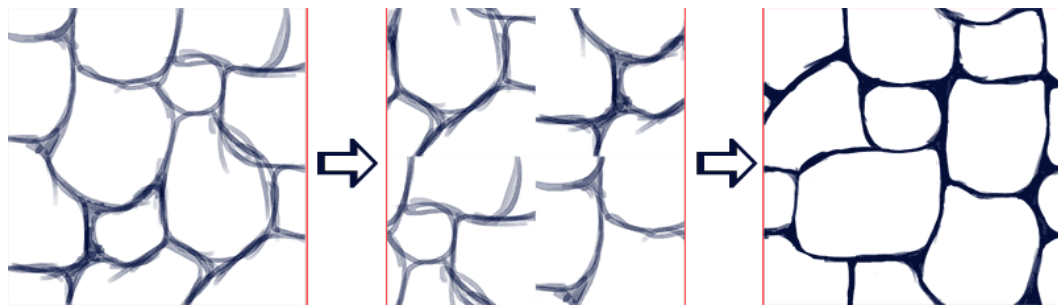


KUVA 33. Tasot järjesteltyinä kansioihin Photoshopissa.

Ensimmäisen luonnoksen jälkeen tasot muokattiin eräänlaiseksi alkukuvaksi, josta voitaisiin toisintaa muita saumattomia kuvia. Adobe Photoshopin offset-filtterillä tekstuuri leikattiin neljään yhtä suureen osaan ja nämä osat asetettiin vierekkäin. Tällä metodilla voidaan tarkistaa, että saumattoman tiilen reunat ovat keskenään yhteneviä. Offsetin saa esille Photoshopin ylävalikon Filter -> Other-kohdasta (kuva 34). Offset-ikkunan arvoiksi on annettu puolet kuvan leveydestä ja korkeudesta. ”Wrap around” tulee olla valittuna, jotta kuvan reunat olisivat toisiaan vasten. Offset-filtterin käyttämisen jälkeen kuvan reunat korjattiin keskenään yhteneviksi (kuva 35). Tarvittaessa offset-filtteri voidaan toistaa, jotta nähdään ovatko reunat edelleen keskenään saumattomia.



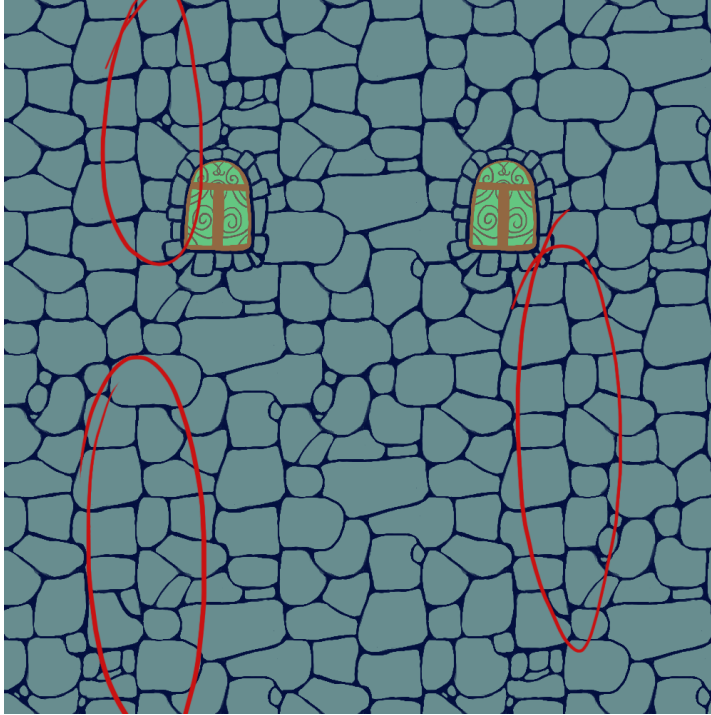
KUVA 34. Offset-filteri.



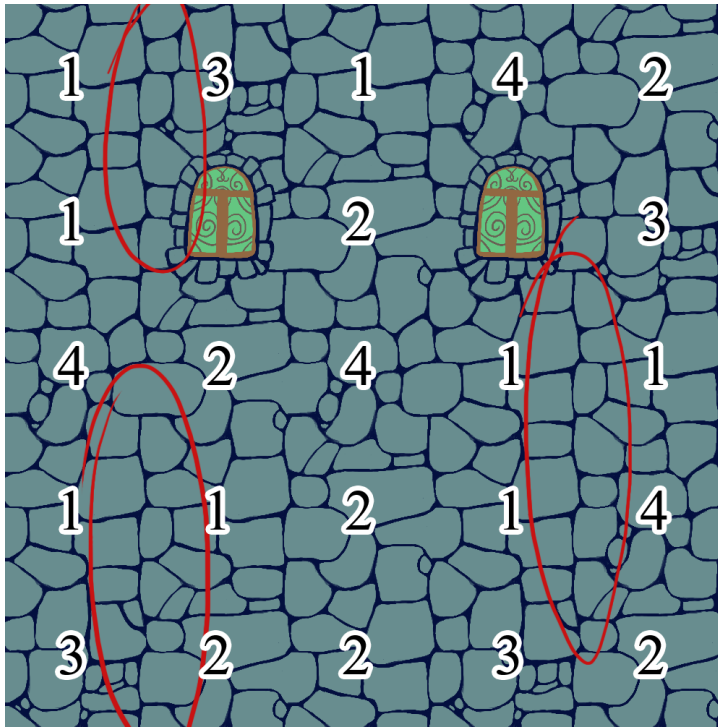
KUVA 35. Taustatiilen luonnostelu, offset-filteri ja reunojen korjaus.

Korjausten jälkeen kuvan luonnos hyväksyttiin ensimmäiseksi tiileksi. Yhden tiilen alkuperäinen koko oli 1024 x 1024 pikseliä, joka lopulta pienennettiin 512 x 512 pikselin kokoiseksi. Uusi tiili aloitettiin kopioimalla ensimmäisen tiilen tasot seuraavaa tiiltä varten. Kopioitujen tasojen keskusosa poistettiin siten, että reunoille jäivät muurauksen reunaviivat niille paikoille, joissa viivojen tuli yhdistyä saumattomasti. Näiden reunojen perusteella luonnosteltiin uusi kiviseinän näkymä. Offset-filterin avulla tarkistettiin, että uusi tiili on edelleen saumaton. Saumattomuutta testattiin myös asettamalla kummatkin tiilet vierekkäin ja katsomalla, että saumakohtat eivät olisi tunnistettavissa. Tässä työskentelyvaiheessa taustan tiilissä ei ollut yksityiskohtia ja varjostuksia.

Tällä työskentelytavalla toteutettiin kaikki taustan tiilet valmiiksi. Tämän jälkeen kuvia testattiin vielä kerran viemällä tasot uuteen dokumenttiin, joka oli kuvakooltaan 5120 x 5120 pikseliä. Tässä koossa tiiliä mahtui vierekkäin ja päällekkäin viisi kappaletta. Tiiliä kopioitiin satunnaisesti vierekkäin, jotta nähtäisiin miltä tausta näyttäisi pelinäky-
mässä. Ensimmäisten luonnosten kohdalla löytyi ongelma, jossa tietyt vierekkäiset tiilet loivat illuusion pitkittäisistä janamaisista saumakohdista. Tällaiset saumakohdat toistui-
vat usein eivätkä näyttäneet hyvältä yhdessä, kuten kuvissa 36 ja 37 on esitetty.

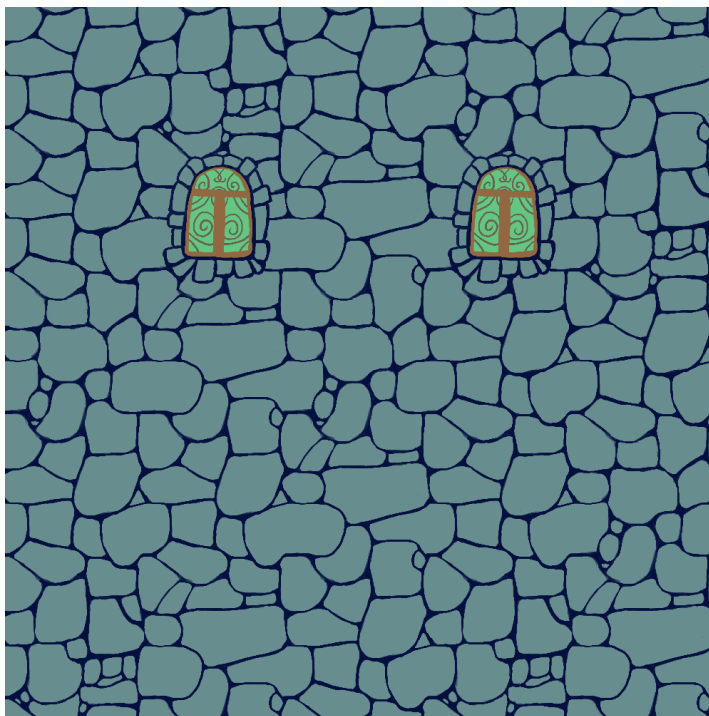


KUVA 36. Esimerkki saumattomien tekstuureiden ongelmakohdista.



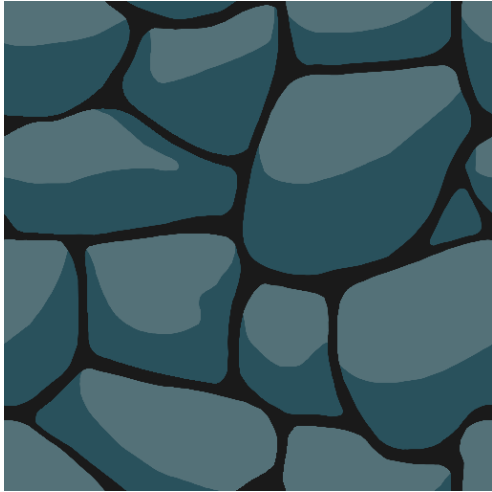
KUVA 37. Eri tekstuurit numeroituina.

Ongelmalliset reunojen yhteneväisyydet pystyttiin korjaamaan tutkimalla niitä tiiliä, jotka aiheuttivat toistuvia janamaisia muotoja, tässä tapauksessa tiilet 1 ja 3. Näissä tiilissä rikottiin yhteneväisyyksiä siten, että kivien muotoa jatkettiin edemmäs ja varmistettiin, etteivät kivien reunat päättyneet samalle linjalle (kuva 38).



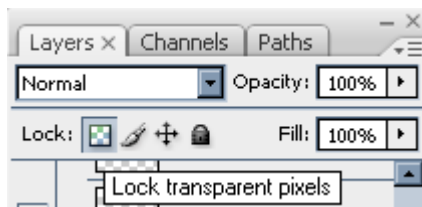
KUVA 38. Korjatut tekstuurit.

Yleisnäkymän ollessa valmis kuvat varjostettiin, jotta niihin saataisiin syvyysvaikutelmaa. Varjojen väri taittui syvempään siniseen ja niitä varten luotiin uudet tasot perusvärin ja muuraussauaman väliin. Varjot piirrettiin kivien muotoja myötäillen (kuva 39). Piirtopöydän paineentunnistuksen avulla voitiin kontrolloida digitaalisen siveltimen paksuutta varjostuksissa.



KUVA 39. Yksittäinen varjostettu tiili.

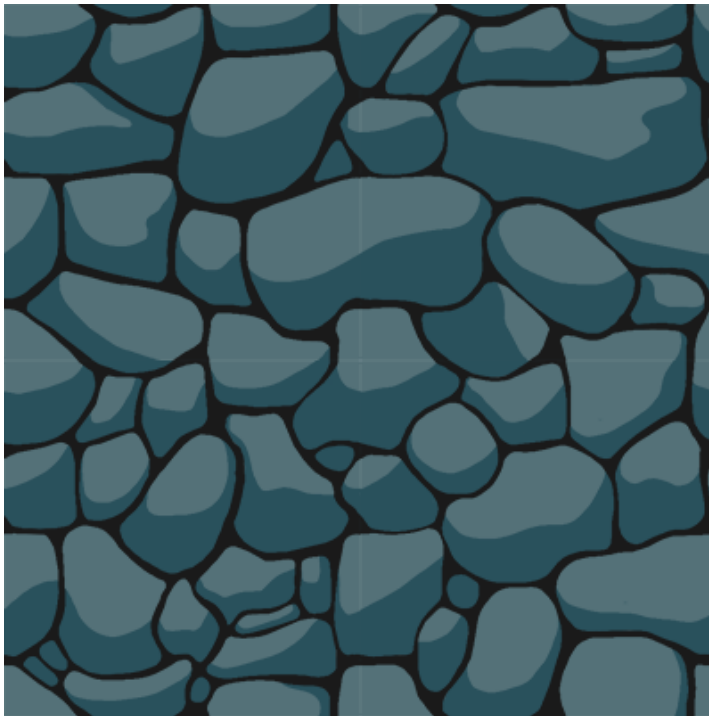
Varjostamisen jälkeen kivien väliset saumat väritettiin tumman sinisestä ruskeaan taittuviksi, koska tausta näytti liian monotoniselta. Ruskeat saumat toivat kiviin toivottua syvyysvaikutelmaa. Lasitetut ikkunat muutettiin pelkiksi aukoiksi, sillä ikkunamaalauksia ei saatu näyttämään suunnitellun kaltaisilta. Valmiiksi piirrettyjen kuvien värejä on helppo muokata valitsemalla Layers-ikkunasta tasolle ”Lock transparent pixels”-lukitus päälle (kuva 40). Tämä suojelee piirtotasolla olevia läpinäkyviä pikseleitä. Lukittuun tasoon piirretylle kuvalle voidaan maalata siveltimellä uusi väri ilman värin karkaamista kuvan reunojen yli.



KUVA 40. Tason läpinäkyvien pikseleiden lukitus.

Valmiiden tekstuurien pienemmäksi skaalaamisessa esiintyi ongelma. Kuvia pienennettäessä Photoshop pehmentää oletuksena kuvien reunoja, jolloin reunoissa esiintyi yhden

pikselin leveydeltä läpinäkyvyyttä. Tämä aiheutti sen, etteivät kuvat enää olleet saumattomia (kuva 41). Tämän ongelman pystyi kiertämään Photoshopin asetuksista (Edit - Preference - General). Kohdasta “Image Interpolation” valitaan asetukseksi “Nearest Neighbor (Preserve Hard Edges)”, joka estää Photoshopia pehmentämästä reunoja. Kuvassa voi kuitenkin esiintyä sahalaitaisuutta, jonka joutuu korjaamaan käsin esimerkiksi pehmentävällä filterillä tai piirtämällä ongelmakohtien reunoja uusiksi. Tämän ongelman olisi voinut välttää valitsemalla vektorikuville tarkoitetun ohjelman kuvien tekemiseen ja suunnitella peligrafiikan vektorigrafiikan kautta. Vektorikuvia olisi voitu skaalata isommaksi ja pienemmäksi ilman samanlaisia ongelmia.



KUVA 41. Skaalauksessa esiintynyt reunojen osittainen läpinäkyvyys.

6.3 Peliobjektien toteutus

Pelimaailmaan sijoitettavat objektit luonnosteltiin sekä Adobe Photoshopilla että käsin piirrettyinä. Erilaisia objektien luonnoksia tehtiin yhteensä noin kolmekymmentä kappaletta, joista valittiin parhaiten peliin sopivat lopulliseen tekstuuripakettiin kuvitettaviksi (kuva 42). Luonnos vietiin Illustratoriin valitsemalla ylävalikosta File - Place -toiminto, jonka kautta haetaan kuvatiedosto joka halutaan taustalle malliksi. Kuva näkyy dokumentissa osittain läpinäkyvänä ja taso on oletuksena lukittu.



KUVA 42. Luonnoksia tekstuuripaketin yksittäisistä esineistä.

Pääkallon luonnosta käytetään tässä esimerkkinä ääri viivojen piirtämisestä. Ääri viivat piirretään uudelle piirtotasolle luonnoksen päälle. Adobe Illustrator hyödyntää piirto-pöydän paineentunnistusta ja sen avulla voi kontrolloida sivellin- eli brush-työkalulla tehtyjen ääri viivojen voimakkuutta (kuva 43). Viivan paksuuden vaihtelulla saadaan kuva näyttämään orgaanisemmalta ja luonnollisemmalta (kuva 44). Vastaavasti paineentunnistus voidaan ottaa pois käytöstä, jos halutaan viivan paksuuden pysyvän aina samana (kuva 45).



KUVA 43. Luonnoksen päälle piirretty ääri viiva.

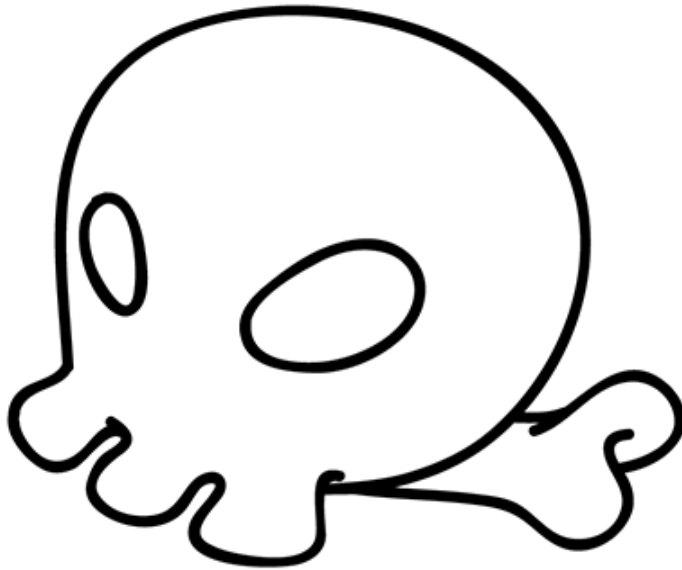
Siveltimellä piirretyt janat ovat vektorijanoja. Vektorijana jakautuu pisteisiin joihin ohjelma luo muodon myötäiset sangat. Sankojen suuntaa muuttamalla ja vetämällä janan pisteitä uusiin kohtiin viivan muotoa voidaan muuttaa piirtämisen jälkeen (kuva 44). Tämä helpottaa piirtämistä, koska viivan ei tarvitse olla heti oikeassa kohdassa luonnokseen verrattuna.



KUVA 44. Jälkeenpäin muokattu jana.

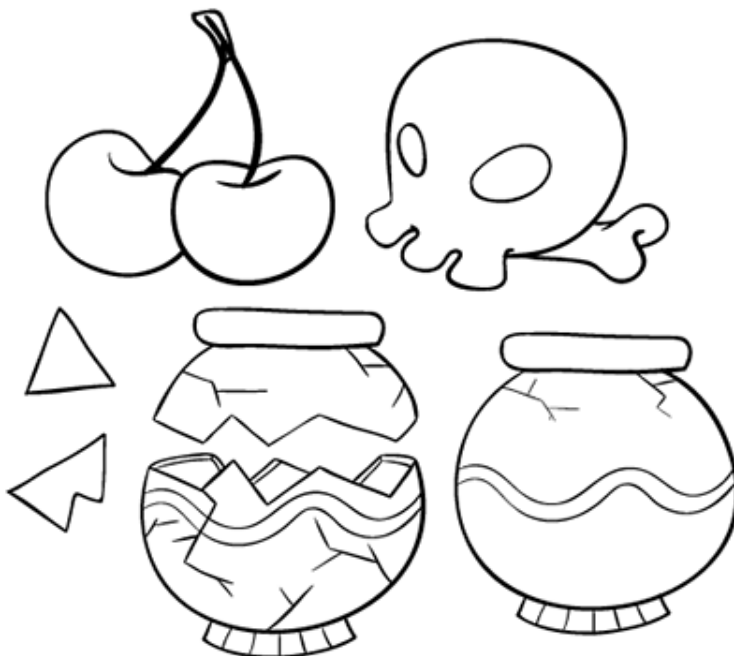


KUVA 45. Valmiit ääri viivat.



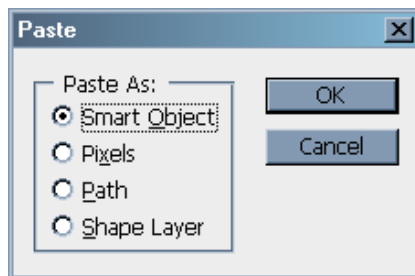
KUVA 46. Ääriviivoitus ilman paineentunnistusta.

Kallo-esimerkissä ääriviivojen piirtämistä on jatkettu yksi jana kerrallaan, kunnes kaikki halutut janat on saatu kohdilleen. Lopuksi janojen pisteet on kohdistettu lopullisiin paikkoihinsa ja sankojen siirtämisen avulla pyöreät muodot on kohdistettu halutulla tavalla. Muiden objektien ääriviivat piirrettiin samalla tavalla (kuva 47).



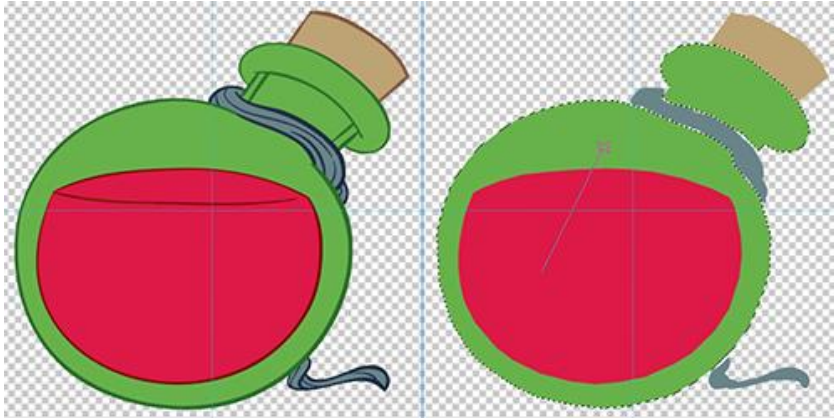
KUVA 47. Valmiita ääriviivallisia kuvia ilman luonnoksia.

Valmiit ääriiviivat voidaan viedä Illustratorista Photoshopiin usealla eri tavalla. Yksi tapa on tallentaa ääriiivakuvat läpinäkyviksi PNG-tiedostoiksi ja avata tämä tiedosto uuteen tasoon Photoshopissa. Kuvista tehdään näin rasterikuvia eikä niitä voi enää pienentää tai suurentaa siten, että reunat pysyvät tarkkoina. Tämän tekstuuripaketin kohdalla tapa ei ollut sopiva, koska ääriiviivat haluttiin pitää vektoreina, joten Illustratorissa tehdyt ääriiviivat kopioitiin ja liitettiin sellaisinaan Photoshopiin. Photoshop kysyy liittäessä, missä muodossa muotoja halutaan käsiteltävän (kuva 48). Smart Object -valinta mahdollistaa kuvan vapaan skaalaamisen niin kauan kunnes taso rasteroidaan. Muodot voi halutessaan liittää myös pikseleinä eli rasteroituina (pixel), polkuina (path) sekä muototasona (shape layer).

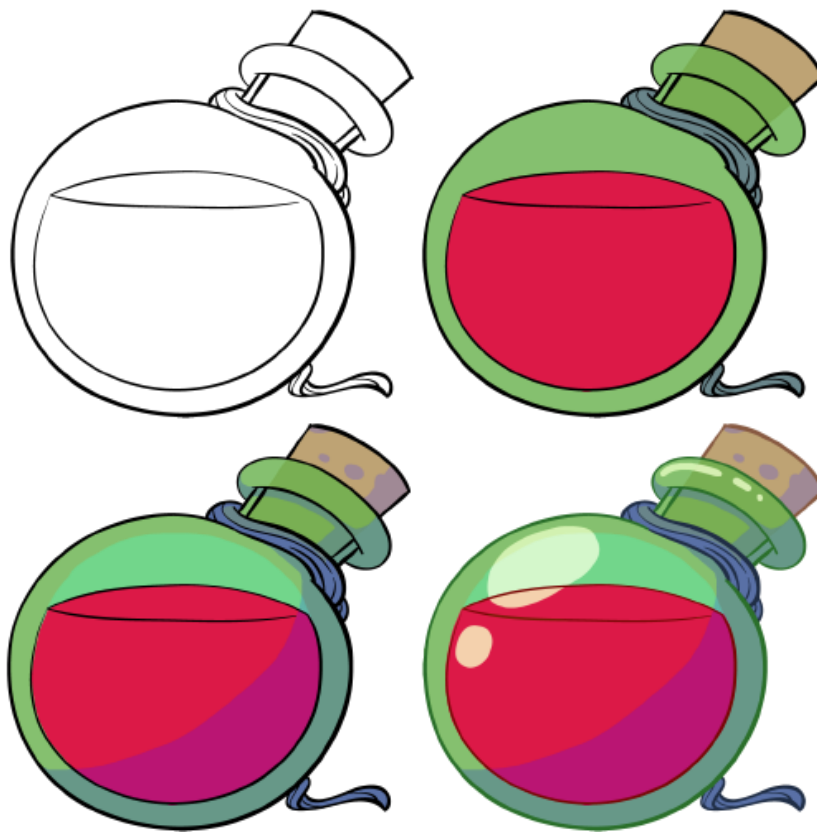


KUVA 48. Illustratorista kopioitujen muotojen liittäminen Photoshopiin.

Photoshopissa objektit väritetään samalla tavalla kuin taustatekstuuritkin, eli jokaisen objektin tasot on ryhmitelty ja nimetty kuvaavasti omiin kansioihinsa. Aluksi tasoja on kahdenlaisia: Illustratorista tuodut ääriiivaryhmä ovat omalla tasollaan ja perusvärit (engl. *flat*) ovat omilla tasoillaan. Perusvärit rajataan reunaviivojen takaa polygoni lasso-työkalulla, joka tekee työkalulla piirretyn polun mukaisen valinnan. Esimerkkinä toimivalle taikajuomapullolle on valittu neljä perusväriä: punainen, vihreä, ruskea ja sininen (kuva 49). Jokainen väri tulisi pitää omalla tasollaan, jotta niitä on helpompi muokata. Tällä tavalla yksittäisen värin vaihtaminen toiseen on nopeampaa. Pullon vihreässä osassa on osittainen läpinäkyvyys, jotta kuvaan tulisi lasimaista tekstuuria (kuva 50). Läpinäkyvyyttä voi säätää tason läpinäkyvyys-asetuksesta (engl. *opacity*).



KUVA 49. Taikajuomapullon perusvärien rajaaminen lasso-työkalulla.

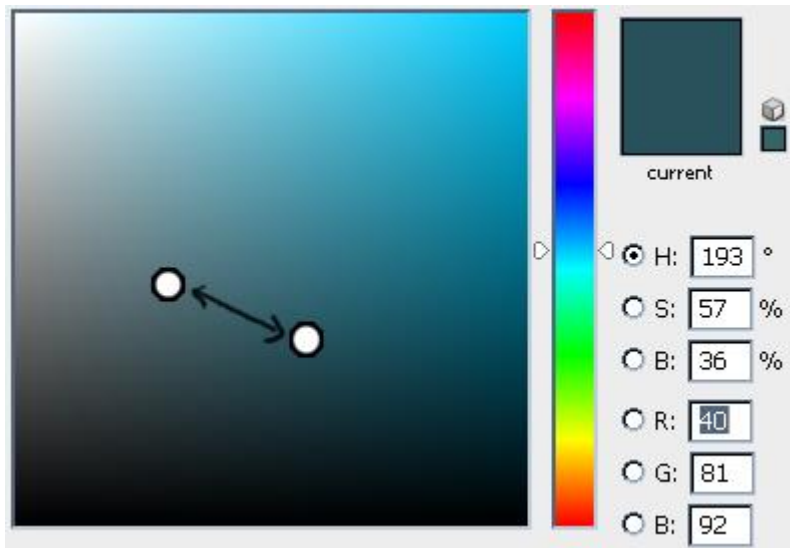


KUVA 50. Objektin värityksen eri vaiheet.

Seuraavaksi perusvärien päälle on luotu uusi taso, johon maalataan kuvassa esiintyvät varjot. Perusväreistä voidaan tehdä valinta (engl. *selection*), joka toimii eräänlaisena maskina, jonka sisällä voi maalata vapaasti. Tämä tarkoittaa, ettei väriä saa maalatuksi kuvan rajojen yli, vaikka sivellin maalaisi valinnan ulkopuolelle. Tasosta saa tehtyä valinnan klikkaamalla tasoa samalla, kun pitää alhaalla control-näppäintä. Kuten aiemmin on todettu, varjojen värien ei tulisi olla tummennettu puhtaalla mustalla, jotta pelissä esiintyvä esine ei näyttäisi lattealta tai muoviselta. Sen sijaan pullon varjot on tehty sinertäviksi pelin taustan mukaan (kuva 50). On tärkeää, että kaikkien esineiden valot ja

varjot ovat yhteneväisiä valon suunnan suhteen. Tietyissä peleissä objektien valot ja varjot toteutetaan ohjelmallisesti, jos pelistä löytyy selkeitä valonlähteitä. Jos oletetaan, ettei tällaisia valonlähteitä välttämättä löydy, kaikkien väritettävien elementtien kohdalla on hyvä päättää, mistä suunnasta oletettu valo osuu kaikkiin esineisiin. Tässä tekstuuripaketissa valonlähteeksi on päätetty haalean keltainen taustavalo, joka sijaitsee pelialueen vasemman yläkulman suunnalla.

Lopuksi väritettävien elementtien ääriviivat on haluttu sävyttää muuksi kuin puhtaan mustaksi. Ääriviivat sisältävä taso on lukittu, jotta reunaviivojen yli ei pysty maalamaan. Reunaviivan viereinen väri on valittu color picker -työkalulla. Väriä on tummennettu ja saturoitu jonkin verran ja tästä on saatu reunaviivalle uusi väri (kuva 51). Kuvan 50 viimeinen esimerkki havainnollistaa, miltä ääriviivat lopulta näyttävät.



KUVA 51. Esimerkki tummemman värisävyn valitsemisesta.

6.4 Kuvien viimeistely ja lähettäminen Unity Asset Storeen

Kuvien värittämistä jatkettiin niin kauan, kunnes halutut tekstuurit oli saatu valmiiksi. Erillisiä tekstuureita lajiteltiin omiksi kuvakartoikseen eli tekstuuriatlaksiksi. Yksittäisten tekstuurien koot ovat 128 x 128 pikseliä, 256 x 256 pikseliä ja 512 x 512 pikseliä. Tekstuuriatlaksiin pyrittiin asettamaan samaan kuvaan mahdollisimman monta samankokoista elementtiä ilman, että kuvaan jäisi tyhjiä osioita. Mahdollisiin tyhjiksi jääneisiin osioihin merkittiin tekstuuriatlasta kuvaava nimi ja yksittäisten ruutujen koko. Esimerkiksi kuvassa 52 yksittäisen animaatoruudun korkeus ja leveys on 256 pikseliä.



COIN ANIMATION
Frame size 256 x 256 pixels

KUVA 52. Esimerkki kolikkoanimaation sisältävästä tekstuuriatlaksesta.

Väritettyjä elementtejä testataan keskenään joko testaamalla niitä Unity 2D -ympäristössä tai tekemällä testikuvia Photoshopissa, jossa elementtejä asetellaan kuten ne esiintyisivät kuvitteellisessa tasoloikassa. Tällä tavalla voidaan huomata värien valinnoissa esiintyvät ongelmat mahdollisimman aikaisessa vaiheessa, jotta ne voidaan korjata valmiiseen grafiikkapakettiin.

Kuvassa 53 näytetään tilanne, jossa on testattu kolikkojen animaatiota, taustan toimivuutta ja koristeiksi tarkoitettujen sammalmättäiden sijoittamista kivisille tasoille. Tausta ei erotu näillä värivalinnoilla tarpeeksi hyvin, joten värejä on korjattu kuvassa 29 esiintyvään näkymään. Tämä saatiin muuttamalla taustan kontrastia tummemmaksi etualaan verrattuna. Tällaiset kuvien testaamiset ovat tärkeitä koko projektin ajan, ja niitä tulisi tehdä mahdollisuuksien mukaan. Näin vältetään turhan työn tekemistä, kun kuvia koskeviin ongelmiin ehditään reagoida ajoissa.



KUVA 53. Elementtien testaamista rinnakkain.

Aiemmassa luvussa tarkasteltiin valoja ja varjoja. Varjojen väriä päätettäessä voidaan tehdä mielenkiintoisen näköisiä ratkaisuja Photoshopin *Sekoitustila*-ominaisuuden avulla. Tasoissa käytettävät sekoitustilat määrittelevät sen, miten taso sekoittuu sen alla oleviin tasoihin. Taso voi esimerkiksi toimia kovana tai pehmeänä valona, lineaarisena lisävalotuksena, tummentaa värejä tai sulauttaa värejä keskenään. Kuvan 48 taikajuomapullon sininen varjostus on saatu muuttamalla varjostusvärin sisältävän tason läpinäkyvyyttä 50 prosenttiin ja valitsemalla sekoitustilaksi ”kova valo”. Pullon valoa heijastavat kohdat on vastaavasti sekoitettu vaalennus-sekoitustilalla. Kuvassa 51 esitetään miten värit sekoittuvat alempana oleviin väritasoihin edellä mainituilla asetuksilla.



KUVA 54. Esimerkki sekoitustiloista Adobe Photoshop CS3:ssa.

Valmiit tekstuuriatlatset tallennetaan niitä kuvaaviin kansioihin. Unity Asset Store -palvelussa peligrafiikkaa on jaettu sekä png-tiedostoina että Unitylle valmiiksi tehtyinä tekstuuri-tiedostoina. Tällaiset tekstuuri-tiedostot toteutetaan Unity-pelimoottorissa. Tekstuuripaketin mukaan on hyvä liittää myös alkuperäiset tiedostot, jotta paketin ostaneet käyttäjät voivat muokata paketin sisältöä tarvittaessa ilman, että he joutuvat ottamaan yhteyttä paketin tekijään.

Tekstuuripaketti on valmis myytäväksi, kun sen todetaan noudattavan Unity Asset Storen vaatimuksia. Viimeistään tässä vaiheessa tekstuureja tulisi testata Unityn sisällä, jotta ne varmasti toimivat kuten on tarkoitus. Materiaalipaketin pakkaamista zip-tiedostoon ei suositella, mutta zip-pakattu tiedosto ei johda materiaalipaketin hylkäämiseen, jos zip-tiedosto luo purettaessa kansiot oikeisiin polkuihin tiedostoineen. Lisäksi häviöllisten kuvaformaattien (esimerkiksi JPEG) käyttäminen materiaalipaketissa johtaa koko paketin hylkäämiseen⁶.

Tekstuuripaketti lähetetään tarkistettavaksi erillisellä Asset Store Tools -lisäosalla, joka asennetaan osaksi Unitya. Tämän lisäosan avulla tekstuuripaketin voi lähettää tarkistettavaksi suoraan Unityn sisällä. Tätä ennen materiaalipaketin tuottajan tulisi rekisteröityä sisällön julkaisijaksi Unity Asset Storen ”Publisher Administration⁷” -sivulla. Sivusto näyttää kirjautumisikkunan, johon syötetään Unity-käyttäjätilin tunnus ja salasana. Julkaisijaksi rekisteröityminen vaatii lyhyen lomakkeen täyttämisen ja lähettämisen sähköisesti hyväksyttäväksi. Julkaisijan hallintasivuilla voidaan lisätä uusi materiaalipaketti

⁶ Vaatimuslistan voi tarkistaa osoitteesta <http://unity3d.com/asset-store/sell-assets/submission-guidelines>.

⁷ <https://publisher.assetstore.unity3d.com/>

hyväksyttäväksi syöttämällä vaaditut tiedot paketille tarkoitetulle lomakkeelle. Tietoja ovat mm. paketin nimi, versionumero, paketin hinta, kuvaus ja kuvankaappaukset. Varsinainen tekstuuripaketti lähetetään Asset Store Tools -lisäosan avulla Unityn sisältä. Jos pakkausta ei hyväksytä myyntiin, mahdollisista puutteista saa tietoa kommentteina. Näin ollen pakettia voi tarjota uudelleen tarkastettavaksi myöhemmin, kun puutteet on korjattu.

7 POHDINTA

Opinnäytetyön aikana tehdyssä projektissa toteutettiin materiaalipaketti, joka koostui useista tekstuuriatlaksista. Ajanhallinnallisista syistä tekstuuripakettia ei viety Unity Asset Storeen projektin päätyttyä. Sen sijaan opinnäytetyössä kuvataan lyhyesti se toimintatapa, jolla Unity Asset Store -palveluun viedään myytäväksi omaa materiaalia. Tekstuuripaketti laitetaan myyntiin opinnäytetyön valmistuttua.

Peligrafiikan työstämisestä haettiin tietoa grafiikan teknisten vaatimusten ja esteettisten valintojen kautta. Käytettävyys otettiin myös tärkeäksi osaksi tiedonhaussa. Edellä mainituista tarkastelukulmista teknisistä vaatimuksista oli helpointa löytää tietoa. Hyvin harva kirjallinen lähde otti kantaa värioppiin tai käytettävyteen pelien graafisessa toteutuksessa. Jotta peleihin liittyvä värioppi ja käytettävyys eivät pohjautuisi pelkästään omakohtaisiin huomioihin, opinnäytetyössä on otettu lähteiksi kuvitustöihin tarkoitettua lähdemateriaalia, josta on johdettu tekstuuripaketin työstämisen aikana hyväksi todettuja käytäntöjä.

Omalta osaltani peligrafiikkaan liittyvä osaaminen perustuu suurimmalta osin aiempaan koulutukseen ja osaksi Internetistä etsittyjen neuvojen kokeilemiseen ja testaamiseen. Neuvoja ovat suurimmaksi osaksi tarjonneet graafisen alan harrastajat ja ammattilaiset. Opinnäytetyön yhtenä tavoitteena oli löytää luotettavaa tietoa ja neuvoja, joista voisi olla apua muille peligrafiikkaa työstäville. Syvällisesti peligrafiikkaa käsittelevän ohjeistuksen kirjoittaminen jäi opinnäytetyössä pintapuolisemmaksi kuin mitä sen oli alun perin tarkoitus olla. Yksityiskohtaisen ja samalla laajan asian käsittely olisi vaatinut suhteettomasti kirjoittamistyötä opinnäytetyölle varattuun työskentelyaikaan verrattuna. Sen sijaan tämän opinnäytetyön luvut 5 ja 6 keskittyvät peligrafiikan toteuttamiseen liittyvien ratkaisujen perustelemiseen ja työn aikana hyväksi todettujen työskentelytapojen kuvaamiseen.

Unitya on kuvattu palveluna ja pelimoottorina peligraafikon kannalta yksinomaan hyötyjen ja positiivisten puolien osalta. Jatkotutkimuksena voitaisiin käsitellä Unityn ohella muita pelimoottoreita peligrafiikan toteuttamisen suhteen. Lisäksi voitaisiin tutkia löytykö muita Unity Asset Storen kaltaisia palveluita verrattaviksi.

LÄHTEET

Kirjalähteet

de Byl, Penny. 2014. *Holistic Mobile Game Development with Unity*. Burlington: Focal Press.

Eddy, Brian R. 2012. *Classic Video Games. The Golden Age, 1971–1984*. Oxford: Shire.

Goldberg, M. & Vendel C. 2012. *Atari Inc. Business is Fun*. New York: Syzygy.

Kane, Carolyn L. 2014. *Chromatic Algorithms Synthetic Color, Computer Art, and Aesthetics after Code*. Chicago: The University of Chicago Press.

Kuorikoski, Juho. 2014. *Sinivalkoinen pelikirja. Suomen pelialan kronikka 1984–2014*. Saarijärvi: Fobos.

Levy, Steven. 2001. *Hackers. Heroes of the Revolution*. New York: Penguin Books.

Stone, T. L., Adams, S. & Morioka, N. 2006. *Color Design Workbook. A Real-World Guide to Using Color in Graphic Design*. Beverly: Rockport Publishers.

Digitaaliset lähteet

Anhet, Anjin. 2014. Let's Get Real About Concept Art. Viitattu 8.3.2014. <http://howtonotsuckatgamedesign.com/2014/02/lets-get-real-concept-art/>

Batchelor, James. 2014. Unity Focus: Bringing life to the Wasteland. Viitattu 20.2.2015. <http://www.develop-online.net/interview/unity-focus-bringing-life-to-the-wasteland/0198811>

Broome, Jeff. 2014. Inside the Box: Adding a Colorblind Mode to Borderlands 2. Viitattu 20.11.2014. <http://www.gearboxsoftware.com/community/articles/1128/inside-the-box-adding-a-colorblind-mode-to-borderlands-2>

Gullen, Ashley. 2013. Remember not to waste your memory. Viitattu 3.12.2014. <https://www.scirra.com/blog/112/remember-not-to-waste-your-memory>

NowGamer. 2012. Brian Fargo Interview: Wasteland 2, Kickstarter & Unity 3D. Viitattu 18.2.2015. <http://www.nowgamer.com/brian-fargo-interview-wasteland-2-kickstarter-unity-3d/#>

Solarski, C. 2013. The Aesthetics of Game Art and Game Design. Viitattu 15.11.2014. http://www.gamasutra.com/view/feature/185676/the_aesthetics_of_game_art_and_.php

Tyson, Jeff. How LCDs Work. 2011. HowStuffWorks. Viitattu 1.11.2014. <http://electronics.howstuffworks.com/lcd.htm>

Unity. 2014. Unity Manual. Asset Store. Viitattu 20.11.2014. <http://docs.unity3d.com/Manual/AssetStore.html>

Unity. 2014. Unity. Submission Guidelines. Viitattu 18.11.2014. 5.1.
<http://unity3d.com/asset-store/sell-assets/submission-guidelines>

Unity. 2014. Unity Manual. Art Asset Best Practise Guide. Textures. Viitattu 1.12.2014.
<http://docs.unity3d.com/Manual/HOWTO-ArtAssetBestPracticeGuide.html>

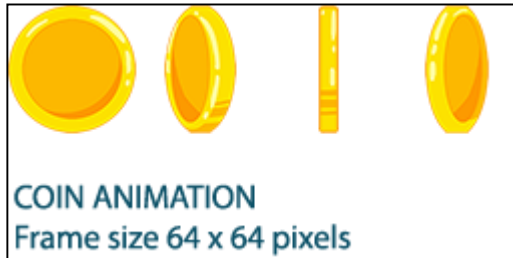
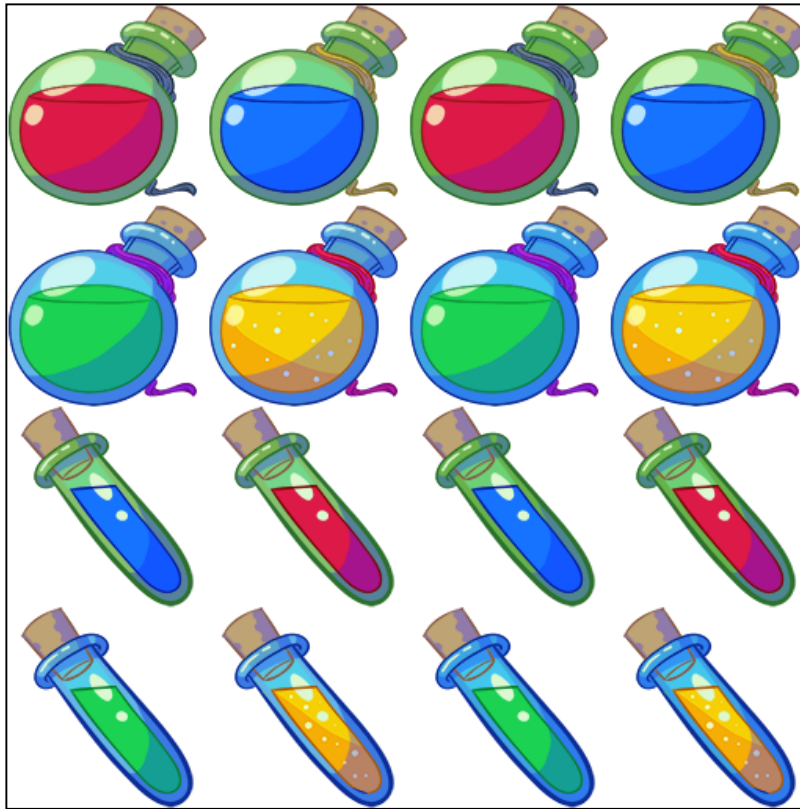
Unity. 2014. Unity Manual. Architecture. Mobile Developer Checklist. Optimization.
Viitattu 1.12.2014. <http://docs.unity3d.com/Manual/MobileOptimisation.html>

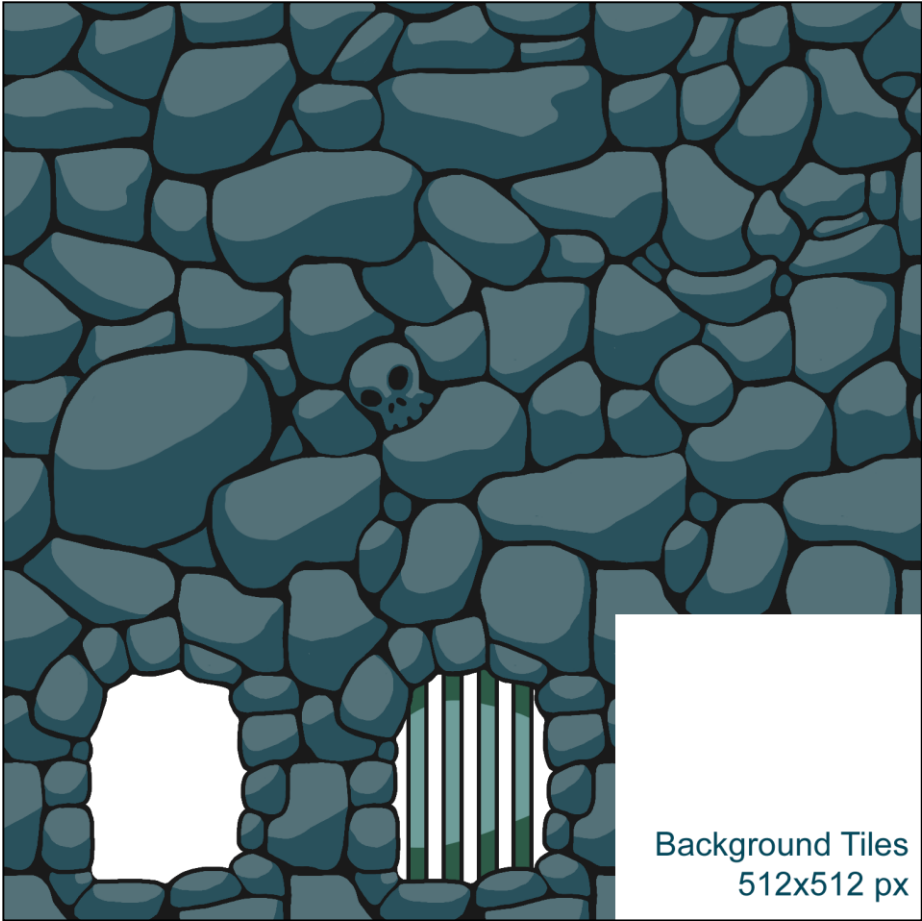
LIITTEET

Liite 1. Lista tekstuuripakettiin valmistuneista kuvista

Elementin nimi	Kuvien määrä	Kuvaus
Taustakuvat - tiilet	8	Keskenään yhteensopivat tiilet, joista muodostetaan pelin tausta.
Lattia - tiilet	3	Kentän lattian muodostavat tiilet.
Tasot ja sammaleet	16	Erilaisia tasoja, joiden päällä pelihahmo voi hyppiä. Tasoilta voi sijoittaa sammalmättäitä koristeiksi.
Vaara - piikit	6	Erilaisia piikkimuodostelmia, joita voi laittaa pelihahmon esteeksi.
Kyltti ja symbolit	7	Kyltti, jonka päälle voi liittää erilaisia ohjesymboleita pelaajalle.
Vipu	6	Animoitava vipu.
Kolikko	4	Animoitava kolikko.
Kerättävät esineet	7	Erilaisia kerättäviä esineitä.
Taikajuomapulloja	16	Valikoima kerättäviä juomapulloja.

Liite 2. Valmistuneet tekstuurit ja tekstuuriatlakset

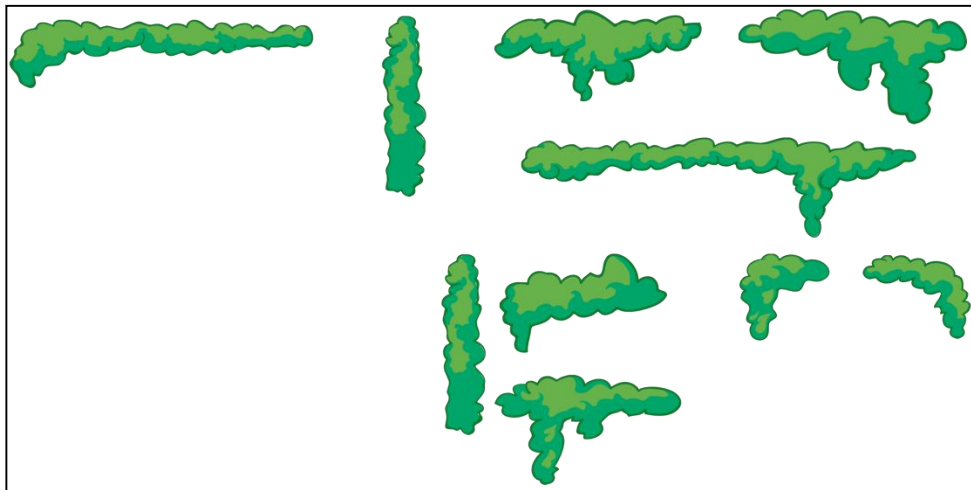


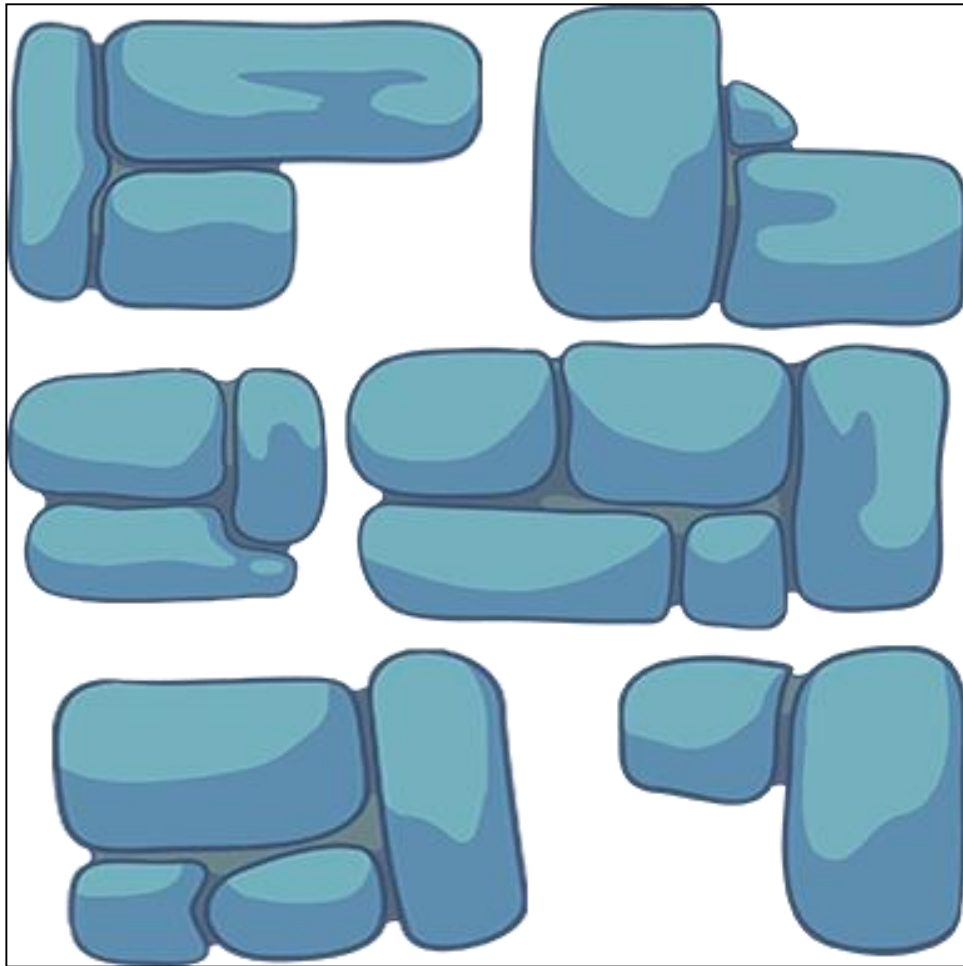


Background Tiles
512x512 px



Collectibles
128x128 px





Levers and bases

Lever: 128x256 pixels

Base: 256x144 pixels



