



Osaamista  
ja oivallusta  
tulevaisuuden  
tekemiseen

Sara Kaipainen

## 2D-grafiikka tasohyppelyyn

Metropolia Ammattikorkeakoulu

Insinööri (AMK)

Tieto- ja viestintäteknikka

Insinööriyö

22.5.2019

Tekijä Otsikko	Sara Kaipainen 2D-grafiikka tasohyppelypeliin
Sivumäärä Aika	54 22.5.2019
Tutkinto	Insinööri (AMK)
Tutkinto-ohjelma	Tieto- ja viestintäteknikka
Ammatillinen pääaine	Pelisovellukset
Ohjaaja	Lehtori Antti Laiho
<p>Opinnäytetyön tarkoituksena oli 2D-grafiikan tuottaminen videopeleihin ja grafiikan erilaisten käyttötarkoitusten tutkiminen. Työssä selvitettiin, kuinka grafiikkaa on tehty jo julkaistuissa videopeleissä ja millaisia tekniikoita niissä on käytetty lisätehosteina. Työssä tutkittiin myös erilaisia polkuja grafiikan teon toteuttamiseksi ja vertailtiin tuotetun grafiikan ominaisuuksia ja eroavaisuuksia. Huomioon otettiin valikoima kuvanmuokkausohjelmia, joiden avulla grafiikkaa pystyy luomaan. Prosessin myötä löydettiin työkaluja, jotka autoivat grafiikan tuottamisessa.</p> <p>Grafiikkapaketti tuotettiin kuvanmuokkausohjelmaa apuna käyttäen sen laaja-alaisen työkalutarjonnan pitkälle hiottujen ominaisuuksien vuoksi. Koska grafiikalta vaadittiin moitteetonta toimimista ja monipuolisia käyttömahdollisuuksia pelimoottorin puolella, tutustuttiin työssä pelimoottorin työkalu- ja editointiominaisuuksiin. Niiden vaatimuksien pohjalta toteutettiin grafiikkaa projektia varten. Työhön luotiin seamless-tekniikalla erilaisia tekstuureja, joita pystyttiin käyttämään pelimoottorissa sekä saumattoman skaalauksen että tilemap-tyyppisten pelitoteutusten kanssa. Lisäksi tehtiin sprite-formaattiin sopeutuvia taustaelementtejä sekä hahmo, jolle luotiin erilaisilla tekniikoilla toteutettua grafiikkaa animaatioprosessia silmällä pitäen.</p> <p>Lopputuloksessa vertailtiin sprite- ja tilemap-metodeilla toteutetun grafiikan hyviä ja huonoja puolia ottaen huomioon ajankäyttö, tarvittava tekninen osaaminen ja visuaalinen lopputulos. Valmiiseen grafiikkapakettiin sisällytettiin mahdollisimman monipuolisia grafiikkaelementtejä, jotta siitä olisi hyötyä monissa erilaisissa 2D-peliprojekteissa. Näin saatiin aikaan toimiva 2D-grafiikkavalikoima opetusmateriaalikäyttöön.</p>	
Avainsanat	2D-grafiikka, peligrafiikka, Photoshop, Sprite, Tilemap, 9-Slice

Author Title	Sara Kaipainen 2D-graphic production to platform game
Number of Pages Date	54 17 May 2019
Degree	Bachelor of Engineering
Degree Programme	Information and Communications Technology
Professional Major	Game Applications
Instructor	Antti Laiho, Senior Lecturer
<p>The purpose for the final year project was to create a versatile package of 2D-graphics for teaching material in Metropolia's game projects. The goal was to make multi-purpose sprites and textures which can be used when creating games and studying how game engine's different tools for graphics work.</p> <p>This study explored how the graphics for published 2D-games were made and what kind of techniques and effects have been used to make the games the way they look. The process of how to make graphics for games are analyzed from the draft to a finished product ready to use in game engine. Various graphic editors were taken into comparison, focusing on their flaws and gains when creating graphical artifacts.</p> <p>The final product for this project was made using graphics software and different techniques for extensive implementation when importing graphics to game engine. The seamless texture-techniques were used to make diverse accessibility for the graphics.</p> <p>The thesis also presents tools in game engine that are practical and time-saving for artists to utilize when working inside game engine. Showcasing different solutions for creating 2D-graphics are meant to be easily available for those who are thinking of building their own graphical content for games.</p>	
Keywords	2D-Graphics, Photoshop, Unity, Sprite, Tilemap, 9-Slice

## Sisällys

### Lyhenteet ja käsitteet

1	Johdanto	1
2	Grafiikan tuottaminen peleihin	2
2.1	Peligrafiikan historia	2
2.2	Kaksiulotteisten pelien esimerkkejä	3
2.3	Olellaiset elementit 2D-peleissä	10
3	Peligrafiikan tuottamiseen käytettävät sovellukset ja työkalut	12
3.1	2D-grafiikan tuottamiseen käytetyt sovellukset	12
3.2	Pelimootorin grafiikkatyökaluja	16
4	Peligrafiikan toteutuksen vaiheet	20
4.1	Hahmottelu ja design	20
4.2	Luonnoksista sprite-grafiikaksi	25
4.3	Pelin kokoaminen pelimootorissa	31
5	Toteutettu grafiikka	34
5.1	Työntekoprosessi ja tulokset	34
5.2	Projektin käännekohtat ja työn arvioiminen	41
5.3	Grafiikan käyttömahdollisuudet tulevaisuudessa	43
6	Yhteenveto	45
	Lähteet	47

## Lyhenteet ja käsitteet

UI	User Inteface, käyttöliittymä
GUI	Graphic User Interface, käyttöliittymäelementti
PNG	Portable Network Graphics, tiedostoformaatti
SVG	Scalable Vector Graphics, tiedostoformaatti
EGA	Enhanced Graphics Adapter, näyttöstandardi
JPEG	Joint Photographic Experts Group, tiedostoformaatti
sprite	Kaksiulotteinen kuva, joka ei sisällä taustaa, vaan on täysin läpinäkyvä. Erityisesti pelin hahmot on toteutettu tällä tekniikalla.
seamless	Saumaton 2D-grafiikkatekstuuri.
grid	Ruudukko, jota käytetään tilemap-tekniikan pohjalla.
tilemap	Kaksiulotteinen, samankokoisista soluista koostuva ruudukko, joka sisältää grafiikkaelementtejä.
tileset	Kaksiulotteinen, samankokoisista grafiikkaelementeistä koostuva arkki.
rigidbody	Komponentti, joka antaa objektille fysiikan pelimoottorissa.

## 1 Johdanto

Insinööriyön tarkoituksena oli 2D-grafiikkapaketin luominen Metropolia Ammattikorkeakoululle ja sen pelisovellukset-pääaineen opetuskäyttöön, koska grafiikantuottamisopetusta ei juuri ole tarjolla. Työssä haluttiin tehdä opiskelijoille toimiva kokonaisuus graafisia komponentteja, joita voidaan hyödyntää tulevilla projekteilla Unity-pelimootorissa. Paketin tuli sisältää tarvittavat elementit pelin grafiikan kokoamiseen, aina animoitavista hahmoista käyttöliittymäelementteihin ja spriteihin, joita hyödyntäen eri tekniikoiden testaaminen ja havainnollistaminen onnistuu helpommin. Spritellä tarkoitetaan kaksiulotteista kuvaa, jonka tausta on läpinäkyvä. Esimerkiksi pelin hahmoa tallennettaessa tausta halutaan jättää pois, jotta se sulautuu pelin muuhun ympäristöön.

Pelaajat kiinnittävät aina huomion peleissä ensin visuaaliseen näköön ennen toimivuuden analysointia. Siksi monet graafisesti viehättävät pelit saavatkin paljon huomiota, vaikka pelin sisältöön ei ole panostettu. Toisessa ääripäässä ovat mielenkiintoiset ja mekaanisesti toimivat pelit, joissa grafiikka ei vastaa muuten toimivan pelin tasoa, ja ne jäävät yleensä vaille huomiota.

Grafiikkakokonaisuutta valmistamaan lähdettäessä suurimmat kysymykset olivat, mitä erilaisia peruselementtejä projektiin tarvitaan ja mitä voidaan lisätä haluttuun tunnelman kasvattamiseksi. Unityn Asset Storella on tarjolla nykyisin paljon ilmaiseksi ladattavia paketteja, jotka sisältävät erilaisia grafiikkaelementtejä, joilla peliin saa nopeasti eloa. Niitä ei kuitenkaan hyödynnetä tarpeeksi, joten valmistamalla matalakynnyksinen, helposti saatavilla oleva paketti rohkaistaan pelinrakentajia käyttämään edukseen myös valmiita spritejä itsetehtyjen rinnalla.

## 2 Grafiikan tuottaminen peleihin

### 2.1 Peligrafiikan historia

Videopelit ovat kautta aikain olleet suosittuja. Kun tekniikka edistyi ja pelihalleihin saatiin ensimmäiset värikkäät pelikoneet, voittokulku lähti etenemään. Ensimmäinen kokonaan viihdekäyttöön tarkoitettu peli, Tennis for Two, oli rakennettu oskilloskoopilla eli sähkötekniikassa käytetyllä mittalaitteella pelattavaksi. Peli koostui visuaalisesti lähinnä yhdestä pysty- ja yhdestä vaakasuorasta viivasta, jotka esittivät maata ja verkkoa, sekä tennispalloa esittävästä näytöllä poukkoilevasta pisteestä. [1.]

Seuraava askel otettiin Computer Space (1971) - nimen saaneella arcade-pelillä, joka oli rakennettu kokonaan vektoreista. Niillä onnistuttiin tekemään siistejä, yksinkertaisia muotoja, mutta yksityiskohdat vielä puuttuivat. Pian vektoreiden tilalle nousivat rasterit, eli pikselipohjainen grafiikka, joilla saatiin tehtyä myös epäsymmetrisiä ja tarkempia kuvia. Se syrjäytti nopeasti vektorigrafiikan, jonka visuaalinen muokkaaminen oli puutteellista. [1.]

1970-luvun aikana huomattiin, ettei yksinkertainen ja mustavalkoinen grafiikka vetänyt pelaajia puoleensa, vaan sai heidät kyllästymään helposti, joten ryhdyttiin kehittämään erilaisia tekniikoita pelaajien houkuttelemiseksi takaisin pelien ääreen. Yksi tapa oli laittaa pelikoneen taustalle värillinen piirretty kalvo, joka toi peliin hieman kolmiulotteisuuden tuntua. Myös erillisiä väritasoja kokeiltiin, jotta pelaajan hahmo muuttaisi väriä. Nämä eivät olleet kovinkaan mullistavia ratkaisuja, eivätkä ne tuoneet suurta visuaalista muutosta peleihin. [1.]

Uuden aikakauden peligrafiikkaan teki arcade-peli Galaxian vuonna 1979. Julkaisussa oli käytetty moniväristä sprite-grafiikkaa niin pelaajan hahmon, vihollisten kuin räjähdysien toteutuksessa. Tämä sai arcade-pelien suosion nousuun, ja 1980-luku olikin niiden kulta-aikaa. Pelien suosion koko ajan kasvaessa myös pelien kehittäjiin kohdistui paineita saada aikaiseksi kilpailukykyistä grafiikkaa. Vuonna 1981 ilmestynyt Turbo-rallipeli toi autopelit suosioon horisonttiin skaalautuvalla grafiikalla, joka sai aikaan mielenkiintoisen 3D-efektin. [2] Tämä ei kuitenkaan estänyt 2D-pelien voittokulkua.

Samana vuonna ilmestynyt Donkey Kong, joka myöhemmin siirtyi seikkailemaan myös konsoleille, on edelleen yksi pelatumista arcade-peleistä Pac-Manin ja Street Fighter II:n ohella.

Kotikonsoleiden leistyminen 1990-luvulta tähän päivään on nostattanut 3D-pelien suosiota ja jättänyt 2D-pelit taka-alalle. Koska nykyiset pelikonsolit ovat tehokkaita eikä suorituskykyä puutu, visuaalisesta räiskinnästä ei tarvitse tinkiä. Kuitenkin 1980- ja 1990-luvun lapset ovat tuoneet aikuisiällään takaisin arcade-pelien käyttämän grafiikan indiepelien muodossa ja nämä ovat saaneet suosiota nostalgisen vetovoimansa vuoksi.

## 2.2 Kaksiulotteisten pelien esimerkkejä

Ensimmäiset mielikuvat 2D-peleistä saattavat olla pikselikuvista koostuvat tasohyppelypelit kimeine ääniefekteineen. Pikselitaide on edelleen käytössä, eikä sen suosio ole laskenut, vaikka sen vierelle on tullut paljon muita toteutustapoja. Arcade-peleistä tuttu värien ja efektien käyttö on edelleen runsasta, mutta myös hillitymmät, askeettiset tunnelmat ovat saavuttaneet suosiota varsinkin indiepeleissä. Tämän myötä myös "Art Game" -genre on kasvattanut pelaajien mielenkiintoa. Näissä peleissä ei yleensä ole juurikaan toiminnallisuuksia, vaan pelaaja kulkee visuaalisesti näyttävissä ympäristössä ja on vuorovaikutuksessa pelin tarinan kanssa. Hyvänä esimerkkinä voidaan mainita vuonna 2012 ilmestyneen kolmiulotteisen, vuoden paras pelikilpailuja voittaneen tarinapeli Journeyn, jossa pelaajan hahmo vaeltaa kauniissa pelimaailmassa kohti taustalla siintävää vuotta käytössään vain muiden hahmojen kanssa vuorovaikuttaminen sekä hyppääminen. [3.]

Insinöörityössä haluttiin ottaa analysoitavaksi kolme 2D-peliä, jotta voitaisiin tarkastella tarkemmin niiden tekniikoita ja toteutuksia grafiikan tekemisessä. Peleiksi valittiin mahdollisimman paljon toisistaan eroavia projekteja, jotta voitiin tutkia, kuinka suuri variaatio 2D-pelien maailmassa on. Valitut pelit olivat mustavalkoinen indiepeli Limbo, käsin piirretty ja animoitu Cuphead ja 1990-luvulla julkaistu roolipeligenren edustaja Diablo I. Yhteinen tekijä näillä peleillä on kaksiulotteisuus ja erilaisia tekniikoita käyttäen luodut syvyysvaikutelmat.



## Limbo

Yksi konseptitaiteen perusteella laajaa mielenkiintoa herättänyt tasohyppelypeli viime vuosikymmenen puolelta on film noir -estetiikkaan nojaava Limbo [4]. Tämä uniikilta näyttävä peli on saanut yksinkertaisilla keinoilla kenttiin syvyyttä ja mielenkiintoa. Itse hahmo on silmiä lukuun ottamatta pelkkä tumma siluetti, kuten myös etualan taso, jossa pelaaja kävelee. Koska Limbo on tehty mustavalkoisesti, tehosteina on käytetty värien sijaan kontrasteja. Tätä on hyödynnetty muun muassa pelaajan silmissä, jotka erottuvat kirkkaudellaan tummanpuhuvasta taustasta. Synkissä kohtauksissa pienet valkeat efektit tuovat tunnetta pelikokemukseen. Tästä esimerkkinä ovat pimeässä hehkuvat madot, jotka kiinnittyvät pelaajan hahmoon, ja rikkinäisten valotaulujen lentelevät sähkökiteet. Myös rakeisuus ja tiettyjen tasojen epätarkkuus tuovat peliin vanhanaikasta mykkäfilmin tuntua. Kuten kuvasta 1 voi havaita, taustalla olevien tasojen pienet sävyvaihdokset saavat aikaan aavemaisen tunnelman ja reunojen vinjetointi on yksi lisätehoste vanhanaikaisen atmosfääriin aikaansaamiseksi. Sama teema jatkuu koko pelin ajan. Viholliset, kuten pelissä kohdattava suuri hämähäkki, ovat vain siluettimaisia varjoja, jotka vaanivat pelaajaa mustavalkoisessa kauhu-maailmassa. [4].



Kuva 1. Limbon pelikuva [5].

Syvyyden tunnetta on saatu käyttämällä mahdollisimman monta harmaasävyä ja kerrostamalla niitä sävyasteittain, tummasta vaaleaan horisonttia kohti mentäessä [4]. Kerrosten käyttäminen on ollut osana 2D-pelien grafiikkaa jo pitkään, mutta monokromaattinen värimaailma saa aikaan synkän ja kiinnostavan tausta-asetelman.

Pelkästään käyttämällä harmaan eri sävyjä on Limboon saatu luotua uskottavaa syvyytsvaikutelmaa. Etualan tummat värit tuovat pelaajalle ahdistavan jännitysmomentin pimeässä odottavista vaaroista ja poispääsemättömyyden tunteesta. Taustalla hohtavat sävyt antavat kuitenkin toivoa paremmasta huomisesta ja pois pääsystä. Tämä efekti on tehty käyttäen apuna moninkertaisia kuvakerroksia ja parallaksiefektiä.

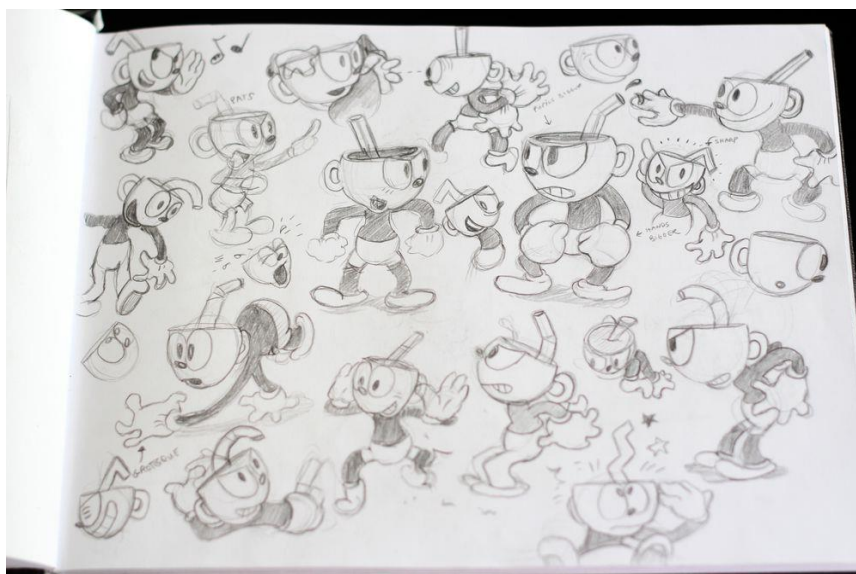
Parallaksiefektillä tarkoitetaan kulkusyvyyttä simuloivia efektejä, joissa kuvakerroksia käytetään myös apuna. Efekti syntyy, kun pelikenttään liitetään kameroita, jotka seuraavat tasoja eri tahtiin. Taaimmaisat tasot liikkuvat hitaammin ja eteenpäin tultaessa tahti voimistuu. Näin saadaan aikaan parallax scrollingina tunnettu ilmiö, joka tuo peliin kolmiulotteisuuden vaikutelmaa pelaajan liikkuessa kentässä [6]. Tekniikan voi myös toteuttaa eri tavoin, esimerkiksi liittämällä kameraan vektorin, joka mittaa tasojen etäisyyttä kamerasta ja laskee niille uudet nopeudet. Efekti havainnollistuu, kun istutaan esimerkiksi liikkuvassa ajoneuvossa ja tarkkaillaan ympäristöä: kaukana horisontissa kiintävät objektit näyttävät pysyvän paikallaan, kun taas lähellä olevat esineet ja elementit vilisevät nopeasti ohitse.

## Cuphead

2D-peli Cuphead [7] jää varmasti pelihistoriaan yhtenä kunnianhimoisimmista projekteista, joka palkitsi pelaajien odotukset venyneestä aikataulusta huolimatta. Tämän lukuisia palkintoja voittaneen run-and-shoot-kategorian pelin alkuperäiset luojat Chad ja Jared Moldenhauer hakivat inspiraatiota vanhoista animaatioista, joita he katsoivat lapsina. Kippari-Kallesta ja Betty Boop -hahmoista tunnettu Max Fleischer sekä Disney toivat 1930-luvulla suurelle yleisölle ensimmäiset piirretyt, jotka ovat edelleen suosittuja niiden visuaalisuuden näyttävyyden vuoksi. Amerikkalaisen animaation Rubber hose -animaatiotekniikka on yksi 1930-luvun animaatioiden peruspiirteistä.

Nimen mukaisesti animaation tukipilarina hahmojen liikkeiden jatkuvuus on kaarevaa, eikä hahmojen raajoissa käytetä kulmia tai niveliä laisinkaan. [8].

Nivelet eivät olleet ainoita venyviä osia pelin tuottamisessa. Cuphead sai osakseen paljon huomiota sosiaalisessa mediassa trailereiden ja kuvankaappauksien myötä, ja siitä tuli yksi odotetuimmista peleistä, jonka valmistuminen venyi pitkään. Ensimmäisten suunnitelmien pohjalta julkaisuvuodeksi oli ilmoitettu loppuvuosi 2014, mutta tähän mennessä pelistä oli saatavilla vain demoversio, jota tekijät esittelivät pelimessuilla. Alun perin kolmen henkilön muodostama pelistudio huomasi pian, että sekä peliä että firmaa oli laajennettava suurta kysyntää vastaavaksi. Vuoden päästä mukana oli neljä uutta artistia, jotka hoitivat kaikki peliin tarvittavat osat konseptitaiteesta käsin piirrettyihin animaatioihin. [7.] Kuvassa 2 on konseptiprosessin tuloksia pelin päähahmon luonnoksista.



Kuva 2. Käsin piirrettyjä hahmotteluja Cupheadista [7].

Jake Clarke, yksi Cupheadin animaattoreista kertoo prosessistaan yhden päivihollisen, ilkeän merenneidon animoinnissa. Pelkästään idle-animaatio eli ns. joutokäyntianimaatio, jota hahmo suorittaa odottaessaan muita tapahtumia, vei 32 paperin verran luonnostelua. Kun luonnokset olivat valmiita, ne piirrettiin uudelleen musteella, jotta saavutetaan sarjakuvamaiset paksut ja erottuvat ääriviivat. Niiden valmistuttua paperit

skannattiin vietäväksi Photoshopiin muokkausta varten. Siellä hahmon liikkeet väritettiin ja niistä poistettiin taustat sprite-animaatiota varten. [7.]

Jake Clarke on sanonut haastattelussaan, että käsin piirrettyihin ruutujen mukanaan tuoma epätäydellisyys oli yksi syistä, jotka tekevät pelistä ainutlaatuisen. Loopilla kiertävien animaatioiden ääriviivat pomppivat peliruudulla ja tuovat elementtiin eloa, mutta myös hauskan kauneusvirheen, joka tekee niistä mielenkiintoisia. [9.] Koska nykypäivänä on siirrytty peligrafiikassa digitaalisuuteen, näitä käsin tehtyjä virheitä ei enää näy, vaan kaikki on tarkasti siistitty ja kiillotettu. Digitaalisen grafiikan virheiden korjaaminen on helppoa kuvanmuokkausohjelmilla, kun taas käsin piirtävän artistin tulee pahimmassa tapauksessa aloittaa alusta. Tekevätkö siis pienet virheet ja harha-askleet pelistä lähestyttävämmän, kuin täydelliseksi hiottut yksityiskohdat?

## Diablo

Diablo on Blizzard Entertainmentin vuonna 1996 julkaisema hack-and-slash-roolipeli, joka sijoittuu fiktiiviseen Khandurasin valtakuntaan [10]. Pelaaja kokee matkallaan kohti Helvettiä ja siellä odottavaa "Diabloa" 16 satunnaisessa järjestyksessä tuotettua tyrmätasoa, joissa sankari taistelee vihollisia vastaan urhoollisesti.

Diablo otettiin mukaan tarkasteluun sen muista peleistä poikkeavan isometrisen grafiikan vuoksi. Sen jatko-osissakin on käytetty samaa tekniikkaa, vaikka Diablo III:ssa kuvakulma ja syvyysvaikutelma on saatu aikaan 3D-mallinnuksilla. Kuvista 3 ja 4 voidaan tarkastella tällä tekniikalla tuotettujen pelien ulkomuotoa ja tutkia niissä esiintyvien pelien Diablo I:n ja Diablo III:n eroavaisuuksia.



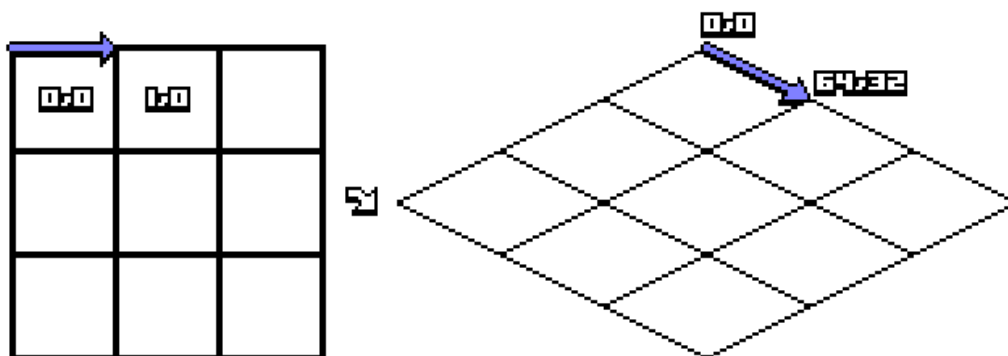
Kuva 3. Diablo I:n pelikuva [10].



Kuva 4. Diablo III:n pelikuva [11].

Isometrisellä näkymällä tarkoitetaan 2D-kuvaa, jonka kuvakulma saa elementin näyttämään kolmiulotteiselta. Oikeaoppinen isometrinen projektio näyttää geometrisen kappaleen osat niin, että xyz-akseleiden väliset kulmat vastaavat 120:tä astetta. Näin syntyy vaikutelma kuvakulmasta, jossa kappaleesta voi nähdä kolme sivua. [12.]

Tekniikkaa kutsutaan myös nimellä "pseudo 3D". Tätä grafiikkaa käyttävät pelit ovat yleensä grid-pohjaisia, eli niissä pelin maailma koostuu kooltaan standardisoidusta ruudukosta. Isometrinen grafiikka tehdään valmiiksi oikeaan kuvakulmaan käännetyille ruudukolle, joka antaa suuntaviivat elementin rakenteelle (kuva 5). Näin grafiikka saa saman katselukulman ja näyttää yhtenäiseltä, eikä outoja perspektiivivaihdoksia synny.



Kuva 5. Isometrinen projektio [12].

Implementoidessa varjoille tehdään oma tasonsa, joka sijoittuu maa-elementtitason ja pelaajan tason väliin. Näin varjot eivät peitä pelaajan tai vihollisten hahmoa. Kaikista tekstuuripalasista tulee tehdä samankokoisia turhan skaalauksen tai poikkeaman korjauksen välttämiseksi. [13.]

Sprite-pohjainen isometrinen tyyli oli 1990-luvulla suosittua sen yksityiskohtaisia maailmoja aikaansaavan tekniikan takia. Tekniikka saa myös kaksiulotteisen grafiikan nousemaan pelialustasta ja saa näin aikaan kolmiulotteisen, eloisan kenttäareenan. Kentän kokonaisuutta pelaaja harvoin näkee kerralla, vaan osa viehätystä on vaeltava kenttää halki ja löytää uusia paikkoja harhailun tuloksena. Paljon käytetty vinjetointi eli reunojen häivytytys tuo lisää mystiikkaa pelikokemukseen. Pelaaja pääsee liikkumaan pelikentässä kahdeksaan eri ilmansuuntaan ruudukoiden mukaisesti. Tämä tarkoittaa, että pelaajalle tehdään kahdeksan erilaista spriteä vastaamaan kaikkia valittavissa olevia ilmansuuntia.

### 2.3 Olennaiset elementit 2D-peleissä

Kun lähdetään toteuttamaan pelin rakennetta, yksi keskeisistä kysymyksistä lienee tarvittavien elementtien teko ja käyttöönotto. Jokaisessa pelissä, oli se tasohyppely-, tornipuolustus- tai topdown-genren edustaja, ensimmäisenä tarvitaan pelaajan hahmo. Oli hahmo putkimies, keltainen, pisteitä syövä olio tai pikseleistä koostuva avaruusalus, se tuo kantavan teeman peliin. Hahmot ovat yleensä sympaattisia, ja ne antavat pelaajalle päämäärän pelikokemuksessa. Yleisesti pelaajan hahmo nähdään ruudulla, ja ensimmäisen persoonan perspektiivi ei olekaan kovin käytetty 2D-peleissä sen kolmiulotteisen vaikutelman luonnin hankalan toteutuksen vuoksi. Niitäkin kuitenkin on: tunnetuin on ehkä vuonna 1993 julkaistu First-Person-Shooter eli "FPS" -kategorian pioneeri Doom (kuva 6), jossa grafiikan tekniikkana on käytetty niin sanottua 2.5D-perspektiiviä [14].



Kuva 6. Id Softwaren vuonna 1993 julkaisema Doom oli uranuurtajia uusien grafiikkatekniikoiden käyttöönotossa [15].

Hahmon ohella ympäristö saa tuotua pelikokemuksen uudelle tasolle. Tarkkaan suunniteltu visuaalinen teema ja sen toteutus voivat olla yksi asioista, jotka joko nostavat pelin suosiota tai latistavat sen muuten hyvin tehtyä mekaniikkaa. Jo pienillä muokkauksilla ja säädöillä saadaan paljon aikaan. Erilaisten värikarttojen käyttö

tunnelman luomisessa ja kontrasteilla leikkiminen ovat tärkeitä halutun tunnelman luomisessa. Esimerkiksi vastavärien käyttö on efektinä voimakas, mutta pienellä vaivalla saavutettu.

Vihollinen mielletään yleensä pelaajalle haitallisina elävinä olentoina, joiden päämääränä on saada hahmo epäonnistumaan tehtävässään erilaisilla pelaajan aseman heikentämiseen vaikuttavilla tekijöillä. Viholliset eli pelaajan voittamisen esteenä toimivat elementit voivat myös olla elottomia esineitä, kuten putoavia kiviä, eteenpäin liikkumista estäviä ansoja tai kenttää pikkuhiljaa täyttävä vesi. Esimerkkinä voidaan mainita tornipuolustus-logiikkaa käyttävä Space Invaders, jossa pelaaja yrittää tuhota omaa tukikohtaansa saartavat lentävät vihollisjoukot. Indietasohyppelypelin nimeä kantavassa Super Meat Boyssa (2010) vihollisina toimivat kentässä erilaiset väijytykset, kuten pyörivät sahat ja tappavat tasoelementit [16]. Tämän pelin platform-grafiikassa on käytetty myös seamless-tekniikkaa kenttien luonnissa (kuva 7). Seamless-tekniikalla toteutetussa grafiikassa kuvalla ei ole saumoja, ja sitä pystyy monistamaan vierekkäin ilman häiritseviä reunoja.



Kuva 7. Super Meat Boyn kentässä vihollisina toimivat muun muassa sahanterät [16].

Käyttöliittymän luomiseen ei välttämättä tarvita itse tehtyä grafiikkaa. Unity, kuin muutkin pelimoottorit, tarjoaa valmiita, yksinkertaisia muotoja, joita voidaan käyttää UI-elementtien luomisessa. Tämä kuitenkin suhteellisen helposti itse tehtävä grafiikka tuo peliin viimeistelyn tuntua, kun se on luotu sopimaan pelin muuhun teemaan. Samaan



kategoriaan lasketaan myös kentässä mahdollisesti kerättävät esineet ja rahat, joiden kuvakkeiden tekemiseen kannattaa jättää aikaa.

### 3 Peligrafiikan tuottamiseen käytettävät sovellukset

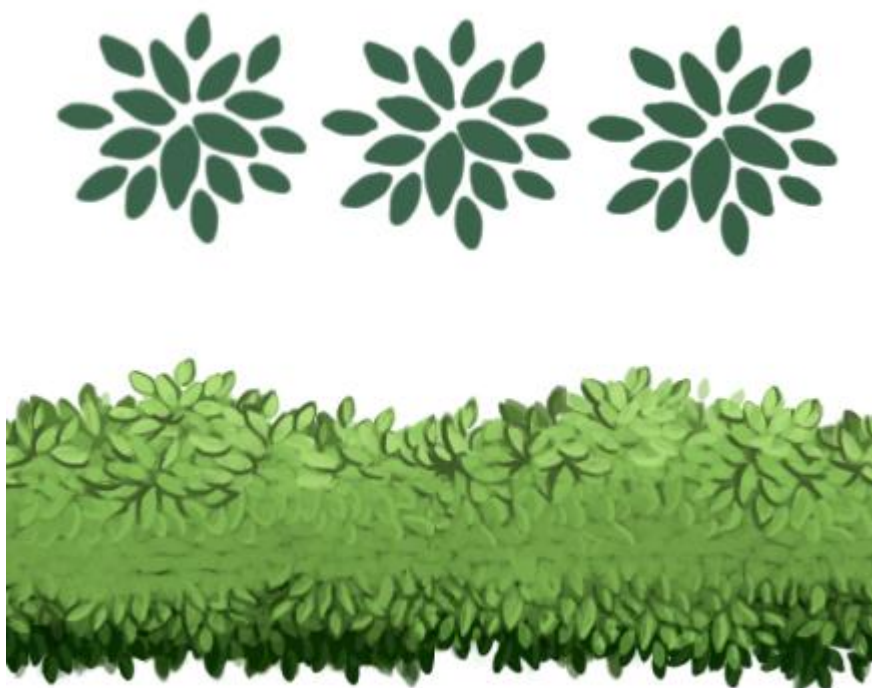
#### 3.1 2D-grafiikan tuottamiseen käytettävät sovellukset

Photoshop on varmasti yksi käytetyimmistä grafiikkaohjelmista, kun lähestytään 2D-kuvien tekoa. Sen pitkään hiotut ominaisuudet helpottavat artistin työskentelyä erilaisilla työkaluilla ja editointivalikoimilla. Myös Unityyn on tehty erilaisia paketteja, jotka toimivat yhteistyössä Photoshopin kanssa. Niistä ehkä eniten käytetty on PSB-importer, jossa kaikki Photoshopissa luodut kuvan tasot näkyvät erillisinä spriteinä Unityssa. Unity tukee myös PSD-kuvaformaattia, joka vähentää sprite-tiedostojen määrää. Yhdessä kuvatiedostossa voidaan tuoda pelimoottoriin monta eri elementtiä ja tehdä niistä erilliset spritet slice-toiminnon avulla. Photoshop tarjoaa tuen monelle kuvaformaatile, minkä vuoksi sen käyttö on suosittua aina kuvanmuokkaajasta mainosten ja mediasisällön tuottajiin. Vektorigrafiikkaformaatti SVG:tä se ei kuitenkaan tue, ja sen tekijät turvautuvat Adoben ohjelmissa Illustratoriin.

Photoshopin työkaluista tärkeimpiä ovat siis kuviin tehtävät tasot, joiden avulla voidaan muokata, poistaa ja lisätä yksityiskohtia, varjoja tai vaikka hahmon sormien määrää ilman koko kuvatiedoston muokkaamista. Koordinaatistotyökalu "grid" eli ruudukko auttaa suunnittelemaan ja toteuttamaan mahdollisimman helppokäyttöisiä ja yhtenäisiä tilesetejä, jotta tekstuurien ja elementtien vienti ja käyttö pelimoottorissa on vaivatonta. Tileset on valikoima samankokoisia kuvia, joita pystytään upottamaan pelimoottorissa tehtyyn ruudukkoon. Grid-työkalu tekee myös pikseligrafiikan toteutuksesta helppoa.

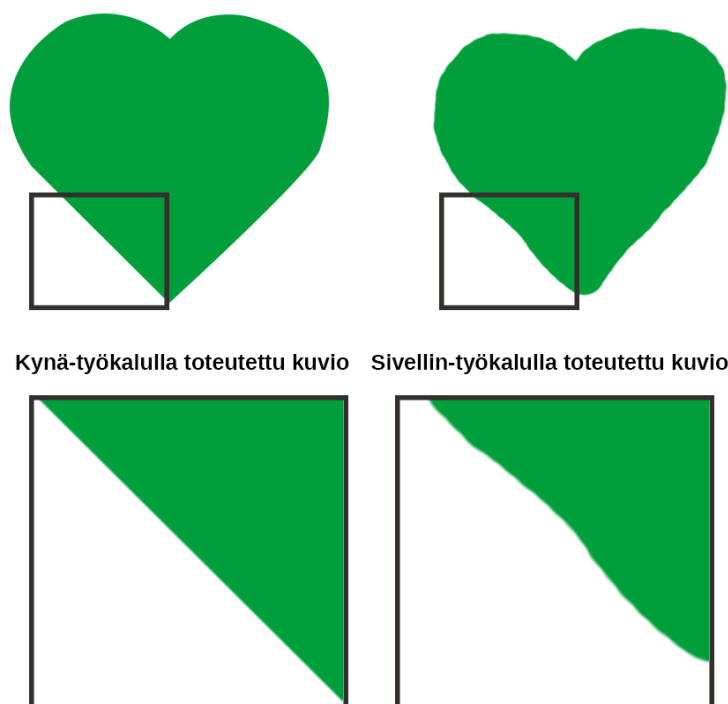
Artistin tärkein työkalu on pensseli myös Photoshopin puolella. Sen sisäänrakennetuissa, valmiissa siveltimissä on paljon erilaisiin tarkoituksiin käytettäviä asetuksia, ja niitä pystytään myös muokkaamaan tarpeen mukaisesti halutun työnjäljen perusteella. Verkosta löytyy myös paljon ladattavia lisäsiveltimiä näiden rinnalle. Monet niistä ovat artistien itsensä tekemiä ja töissään käyttämiä. Työkaluja on niin ikään erilaisten

tekstuurien, kuten öljyvärinäisten maalausten, spraymaalien ystäville tai realistisen grafiikan tekemistä tekstuurien kannalta hakeville. Siveltimistä voidaan myös tehdä leimasinmaisia työkaluja, jotka nopeuttavat paljon kuviota toistavan grafiikan tekemistä. Puiden, ruohotupsujen tai aaltojen monistaminen käy helposti ja säästää aikaa, kun piirustuksesta halutaan yhdenmukainen ilman suuria eroavaisuuksia muotojen kanssa. Tätä ominaisuutta hyödynnettiin insinööriyössä pensaikko-spriten tekemisessä luomalla lehdistä koostuva sivellin, jolla saatiin aikaan eloisa tekstuuri piirtämättä jokaista erillistä lehteä yksi kerrallaan (kuva 8). Photoshopissa pystytään luomaan myös omia sivellinvalikoimia helposti ja nopeasti. Siveltimien teko onnistuu jokaiselta vähänkään ohjelmaa käyttäneeltä. Tarvitaan vain valkoinen tausta, jolle haluttu siveltimen kuvio piirretään yksivärisenä. Kun ollaan tyytyväisiä kuvioon, navigoidaan Edit-valikosta Define Brush Preset -kohtaan. Tämän jälkeen nimetään uusi sivellinasetus, ja näin se on valmis käytettäväksi.



Kuva 8. Pensasmateriaalin tekstuurin luominen Adobe Photoshopissa tehdyn siveltimen avulla.

Kynä-työkalua (pen tool) käytetään paljon vektorigrafiikan valmistuksessa. Sillä pystytään helposti tekemään niin suoria viivoja kuin kaarevia, mutta teräviä ääriviivoja ja muotoja. Photoshop käyttää kynä-työkalulla tehdyissä kuvioissa vektorigrafiikassa hyödynnettävää tekniikkaa. Muodot valmistetaan tekemällä ääriviivojen väliin pisteitä, jotka lopulta muodostavat pisteiden välisen polun. Pisteitä kutsutaan ankkureiksi, ja niitä voidaan kuvion luonnin yhteydessä tai sen jälkeen käsitellä haluttuun muotoon ja paikkaan xy-akselissa. [17.] Työkalun avulla saadaan aikaan suoria, tarkkareunaisia kuvioita, kun grafiikasta halutaan mahdollisimman siistiä orgaanisen sivellingrafiikan sijaan. Kuvasta 9 voidaan havaita kahden eri tekniikalla tehdyn symbolin eroavaisuuksia reunojen siisteyden suhteen. Valmiita muotoja voidaan myös vektorigrafiikan mukaisesti skaalata sopivan kokoiseksi tarkkuuden kärsimättä. Tallennuksen jälkeen Photoshop rasteroi kuvion pikselikuvaksi, ja näin suurentaminen ei enää onnistu tallennuksen jälkeen ilman, että kuvanlaatu ei kärsisi.



Kuva 9. Kynä- ja sivellintyökalujen erot.

Täydellinen ohjelma Adobe Photoshop ei kuitenkaan ole. Sen käyttöliittymä on sekava ja saattaa saada monen aloittelijan hakemaan muita kuvanmuokkausohjelmavaihtoehtoja. Ohjelma sisältää erilaisia työkaluja, joita aloittelija tai keskitason käyttäjä ei tarvitse työssään. Pitempäänkin ohjelmaa käyttäneet saattavat välillä turhautua valikoiden

runsauteen. Creative Cloudin mukana tulleet lisätyökalut ja niiden hidas oppiminen ovat tuoneet lisähaasteita varsinkin tavalliselle käyttäjälle. [18. Ohjelma vaatii käyttäjältä päättäväisyyttä käytön opettelussa, mutta palkitsee lopulta monimuotoisilla käyttömahdollisuuksilla ja työvälineillä.

Adoben tuotteet eivät myöskään ole ilmaisia ja harrastelijoille, ja Photoshop voi olla liian kallis vaihtoehto. Monet turvautuvat halvempiin tai ilmaiisiin vaihtoehtoihin, joita on nykypäivänä monia. Yksi suosituimmista on avoimen lähdekoodin kuvanmuokkausohjelma Gimp. Se on vuosien kuluessa saanut lisää käyttäjäkuntaa maksuttomuuden ja monipuolisten työkalujen ansiosta, jotka haastavat Adoben tuottamia palveluja. Gimpin käyttöliittymä on saanut paljon vaikutteita Photoshopista, jonka käyttäjä voi helposti siirtyä tähän ilmaisohjelmaan ilman paljoa harjoittelua. [19.]

Gimpin käyttöliittymässä on kuitenkin myös omat vaiheensa, joita uuden käyttäjän on harjoiteltava. Kuvaformaattien tuki on miltei yhtä laaja kilpailijan kanssa, ja rasterikuvien ohella Gimp tukee myös vektorigrafiikan tekoa ja formaattia. Avoimen lähdekoodin ja suuren käyttäjäkommuninsa avulla Gimp toimii Windows ja macOS -käyttöjärjestelmien lisäksi Linuxin koneilla. Photoshop puolestaan tarjoaa ohjelmia käyttöönsä vain kahdelle ensimmäiselle. Ohjelmaan voidaan myös ohjelmoida erilaisia algoritmeja ja sopivia liitännäisiä käytön helpottamiseksi ja automatisoimiseksi. Tuettavia ohjelmakieliä ovat muun muassa C, C++, Perl ja Python. [19.]

Pikseligrafiikkaan mentäessä tarjolla on myös paljon ilmaisohjelmia, joista suurin osa toimii HTML5-pohjaisina, web-selaimissa toimivina ohjelmina. Nämä ainoastaan pikseligrafiikan tekoon käytettävät ohjelmat ovat suuri apu grafiikan teossa, ja niiden avulla saadaan nopeasti aikaan monenlaisia elementtejä peliin. Yksi näistä on selaimessa toimiva ja offline-tilassa käytettäväksi ladattava kuvanmuokkausohjelma PiskelApp, joka on saanut arvioissa paljon kehuja ominaisuuksistaan. Sen ladattava versio toimii Windowsilla, Macilla kuin Linuxilla, ja käytetyimmät selaimet tukevat sen verkkosivupohjaista versiota. Myönteistä huomiota se on saanut hyvällä animaatiotarjonnallaan, miellyttävällä käyttöliittymän ulkomuodolla ja tarvittavalla valikoimalla erilaisia työkaluja pikseligrafiikan tuottamiseen. Kritiikkiä on aiheuttanut rajallinen rotaatiotuki ja yksittäisten osien skaalauksen puuttuminen. [20.]

Rasterigrafiikasta mainittakoon myös käsin tehty grafiikka. Vaikka sen muokkaamista ei juuri tarvitse tehdä ohjelmistojen puolella, skannauksen jälkeen siitä on tehtävä sprite-formaatissa toimivia tiedostoja, kuten PNG-kuvia. Näissä tiedostoissa on läpinäkyvä tausta eikä hahmolla ole ympärillään paperia tai muuta taustaväriä, ja näin ne sulautuvat hyvin käytettäväksi pelimoottorissa.

Vektorigrafiikan tekoon käytettäviä ohjelmia ovat niin Adobe Illustrator kuin avoimen lähdekoodin Inkscape. Yleisesti ottaen vektorigrafiikan käyttö peleissä on nykyään suhteellisen harvinaista muun muassa sen ulkonäköä rajaavien ominaisuuksiensa vuoksi. Tällä hetkellä Unity ei automaattisesti tue SVG-tiedostoja, mutta sille on tehty erillinen svg importer -paketti, joka voidaan ostaa Unityn verkkokaupasta.

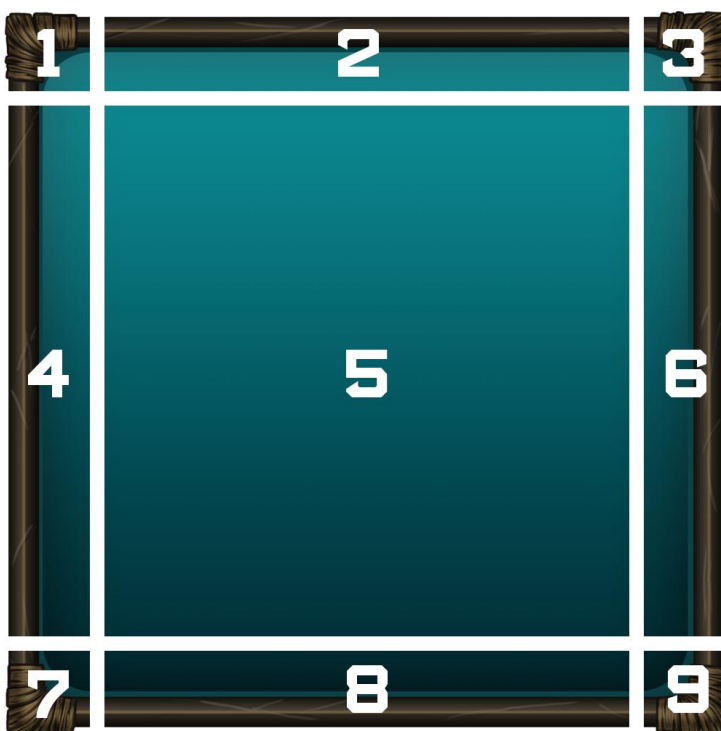
Vektorigrafiikan ominaisuutena on sen rikkoutumaton kuvanlaatu: sitä voidaan skaalata loputtomiin, eikä se menetä tarkkuuttaan missään vaiheessa. Tämä ominaisuus pääsee kuitenkin hyötykäyttöön paremmin media- ja painotekniikan puolella kuin pelien teossa. Vektorigrafiikan visuaalinen ilme on hyvin suoraviivainen ja erottuva tarkkoine viivoineen ja tasaisine, yksiulotteisine väriyhdistelmineen. Vaikka vektorigrafiikkaa voidaan käyttää peleissä, sen on laskettu myös vievän ajonaikaista suorituskykyä. Microsoftin tutkijoiden Charles Loopin ja Jim Blinnin laskelmien pohjalta tehdyssä artikkelissa todetaan, että videopelien parasta mahdollista suorituskykyä etsiessä huomattiin vektoreiden hidastavan suoritusta niiden ajonaikaisen uudelleen laskennan takia, kun taas bittikarttagrafiikan koko ja väri-informaatio on määritelty etukäteen eikä niiden muuntamista tapahdu ajon aikana [21]. On siis hyvä punnita hyötyjen ja haittojen määrää, kun suunnittelee vektorigrafiikkapohjaista pelintekoa.

## 3.2 Pelimoottorin grafiikkatyökaluja

### 9-Slice

Peleissä tarvitaan erikokoisia elementtejä esimerkiksi pelaajan interaktiivisissa kohtaamisissa pelimaailmassa, ja niissä tekstille tarvitaan sopiva taustakuva sen erottamiseen muusta ympäristöstä. Tekstin pituuden vaihtelevuuden takia kuvitellaan, että artisti joutuisi tekemään suuren määrän erikokoisia spritejä, joista sitten Unityssa

valitaan sopivat asiakohtaisesti. Samankaltaisia tilanteita ovat myös taustaan rakennettavat erikokoiset elementit, kuten muurit ja seinämät. 9-Slice-työkalu poistaa tämän työvaiheen kokonaan ja on yksi tehoa ja aikaa säästävimmistä Unityn ja myös muiden pelimoottoreiden tukemista työkaluista. Tekniikan pohjana hajotetaan valittu sprite yhdeksään osaan, joista eri osat käyttäytyvät eri tavoin (kuva 10).



Kuva 10. Sprite on jaoteltu 9-Slice-tekniikan mukaisesti yhdeksään osaan.

- Kuvassa 10 numeroin 1, 3, 7 ja 9 merkityt kuvan kulma-alueet eivät skaalautu tai muuta kokoaan kuvan koon uusimisen yhteydessä.
- Ylä- ja alareuna (kuvassa 10 numerot 2 ja 8) venyvät vaakasuunnassa.
- Sivureunat (4, 6) puolestaan venyvät pystysuunnassa.
- Viimeiseksi jäänyt suurin keskiosa (5) skaalautuu sekä horisontaalisesti että vertikaalisesti.

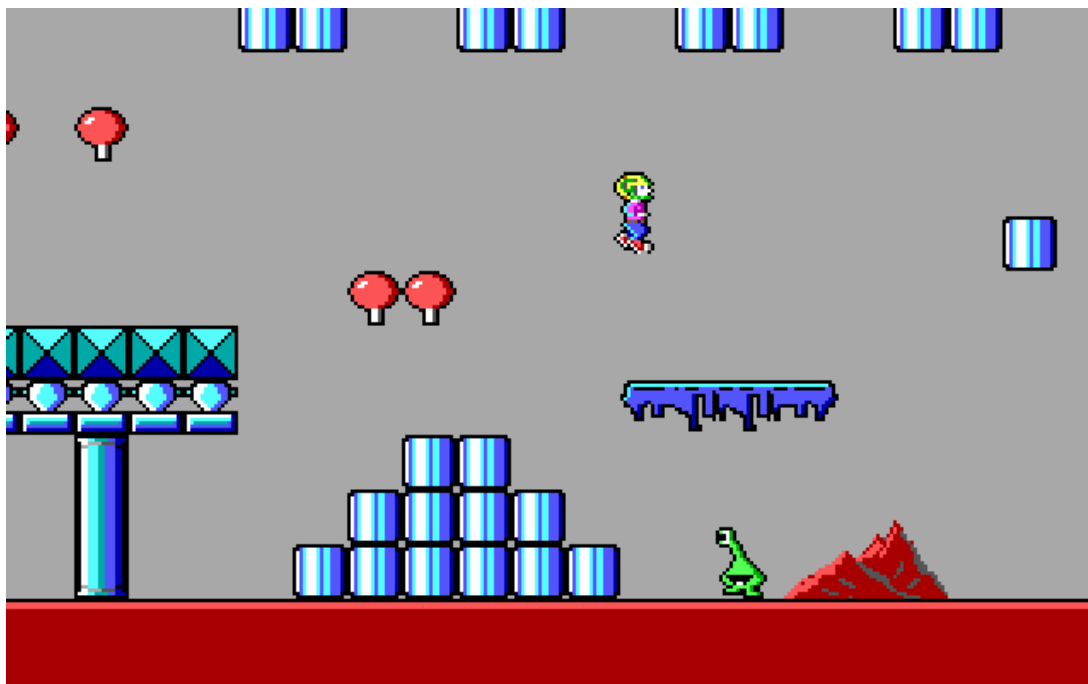
Keskiosan skaalautumista voidaan säätää Unityssä niin, että se joko venyttää kuvaa isommaksi (*“scaled”*) tai toistaa kuviota ruudukkomaisesti kuvan laajentuessa (*“tiled”*). Näistä kahdesta valitaan parempi vaihtoehto kuvakohtaisesti. Esimerkkikuvassa 10 on parempi käyttää skaalausta, koska keskiosassa tapahtuu väriaihtelua, joka ei

pinottaessa toistu saumattomasti. Skaalaaminen onnistuu vain horisontaalisti, kun lopputuloksesta halutaan hyvä. [22.] Jos näin ollen monistetaan keskiosaa tiled-metodilla, lopputuloksesta tulee shakkilautamainen ja katkonainen.

Spriten kulmat, jotka eivät siis muuta muotoaan skaalauksen aikana, voivat olla hyvinkin yksityiskohtaiset, eikä niiden tarvitse olla suorakulmaiset. Paloittelun aikana on otettava huomioon kuitenkin nurkan muodot ja jätetään kulmien sisäpuolelle kaikki kaareutuvat ja suorasta viivasta poikkeavat viivat. Näin reunapalat pysyvät suorakulmioina, eikä skaalaus saa spriteä muodostamaan epätasaisia kehyksiä. Spriten reunat skaalautuvat yhtä lailla, eikä niihin tule tehdä juurikaan yksityiskohtia samojen, aikaisemmin todettujen ongelmien välttämiseksi. 9-Slice-tekniikka estää myös reunojen paksuuden pienenemisen, kun spriteä kutistetaan. Tavallista skaalausta tehtäessä myös ääriviivat pienenevät, ei pelkästään muoto.

## Tilemap

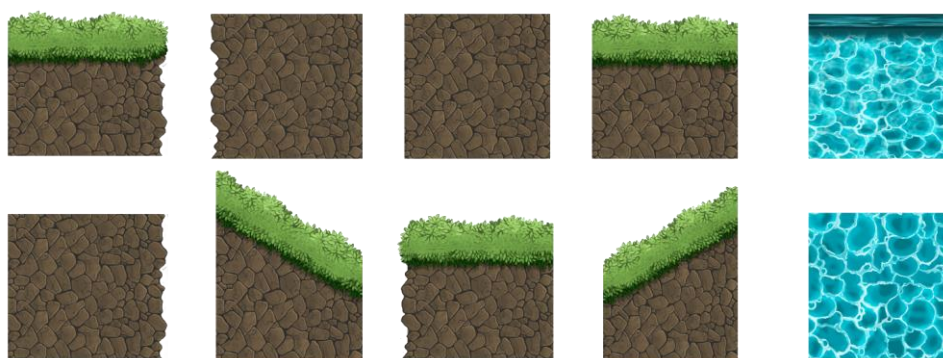
Tilemap-pohjaiset pelit ovat ruudukossa olevista palasista koostuvia kenttiä. Niitä ovat niin sivulta kuvatut perinteiset tasohyppelypelit, kuten EGA-grafiikkaa ensimmäisinä peleinä käyttänyt Commander Keen (kuva 11), kuin isometristä projektiota hyödyntävät pelit, joihin Simcity 2000, aikaisemmin mainittu Diablo (1996) sekä pelisarja Civilization kuuluvat, toisin sanoen kaikki pelit, joiden kenttänä toimii samankokoisista soluista koottu alue. Solujen muotoina käytetään eniten suorakulmaisia spritejä, mutta esimerkiksi kuusikulmioitakin on otettu käyttöön vuosien varrella.



Kuva 11. Commander Keen käytti ensimmäisinä peleinä EGA-näyttöstandardia, joka mahdollistaa 16 värisävyn käytön [23].

Unity on sisäänrakentanut tilemap-työkalun, joka helpottaa kentän suunnittelua ja kokoamista. Pelikenttään luodaan tilemap-objekti, joka tuottaa valinnaisella muodolla toistuvan ruudukon pelikentän alustan luontiin. Valmiita kuvioita on tarjolla tavallisen suorakulmion ja kuusikulmion lisäksi isometrisiä vaihtoehtoja. Sprite-arkeista (kuva 13) tehdään tämän jälkeen Palette-työkalu erottamalla spritet toisistaan slice-editorin avulla. Jotta Unity tekee jokaisesta palasta oman spriten, tulee Inspector-valikosta valita Multiple-vaihtoehto. Tämä eroteltu arkki viedään paletti-ikkunaan, joka muodostaa osista kentän tilemap-ruudukkoon maalattavia elementtejä. Nyt aikaisemmin luotu ruudukko toimii canvasina paletille, ja niitä voidaan raahata, maalata ja toistaa soluihin halutunmukaisen kentän saavuttamiseksi. [24.]





Kuva 13. Valmiista tekstuureista koostuva arkki eli tileset.

Tätä tekniikkaa ei ole tietenkään pakko käyttää siihen helposti sovellettavissa olevissa peleissä, mutta se nopeuttaa kentän rakentamista ja suunnittelua. Kentän suunnittelusta ja kokoamisesta vastaava tekijä voi helposti valita paletista haluamansa kentän palasen ja maalata sitä ruudukkoon. Erilaisten poikkeamien, kuten mäkien ja kumpareiden, toteutus käy tällä työkalulla vaivattomasti. Jos kenttä tehdään manuaalisesti raahaamalla spritejä kenttään, on pidettävä huoli, että sprite on oikeassa kohdassa xy-akselilla, niin että elementit eivät limity tai niiden väliin ei jää tyhjää tilaa. Ruudukossa tätä ongelmaa ei ole, kunhan spritet ja ruudukon solut on tehty samankokoisiksi.

## 4 Peligrafiikan toteutuksen vaiheet

### 4.1 Hahmottelu ja design

Yleisesti ottaen design-vaiheessa artistille on annettu jo tiettyjä suuntaviivoja, joita vasten pelin graafista ilmettä aletaan muokata. Pelin designer on suunnitellut, millaista peliä aletaan luoda tarinan ja visuaalisen teeman pohjalta. Pelejä on olemassa niin paljon kuin pelaajiaakin, ja pelkästään rajaamalla pelit 2D:hen, on mahdollisuuksia ja genrejä silti runsaasti.

Tornipuolustus-genressä yritetään suojella omaa tukikohtaa vihollisilta. Tätä tekniikkaan on käytetty muun muassa Space Invaders -pelissä sekä sosiaalisessa mediassa vaihtovuoroin pelattavissa peleissä, yksi näistä on Plants vs. Zombies Adventures (2013). Top down-genren peleissä on ylhäältä kuvattu kamerakulma, joka saa pelin näyttämään lautapeliltä. Isometriset roolipelit, pulmapelit, genrelistaa voitaisiin jatkaa parin sivun verran. Asian ydin kaikissa pelivaihtoehdoissa on se, mistä suunnasta pelattavaa kenttää tarkastellaan ja näin, kuinka katselusuunta toteutetaan grafiikan avulla. Insinööriyö pohjautuu suosittuun platformer-formaattiin eli sivulta kuvattuun tasohyppelypeligenreen, jonka suosio lähti liikkeelle Super Mariosta ja PacManeista 1980-luvun vaihteessa.

Teema ja pelin tarinapohja on siis päätetty, kun hahmotteluun tarvittavia ohjeita annetaan artistille. Tässä työssä suunta oli vapaa, koska projekti tehtiin yksin, eikä rajoituksia ollut kuin ajankäytön rajallisuus. Koska sekä 2D-pelin genre että teema oli valittu, pystyttiin projekti aloittamaan ensimmäisten spriteistä ilman, että tässä vaiheessa olisi tarvinnut miettiä kentän pohjakarttaa. Koska työhön päätettiin tehdä muun muassa tilemapseista koostuvia saumattomia tekstuureita, niitä olisi helppo muokata pelimootorissa haluttaviin kokonaisuuksiin pelin rakennusvaiheessa.

Ensimmäiseksi prioriteetiksi otettiin kuitenkin pelihahmon teko. Grafiikasta haluttiin tehdä omannäköistä, mutta samalla moneen eri tarkoitukseen sopivaa. Parin luonnoksen jälkeen löydettiin sopiva suunta hahmolle, joka pääsikin nopeasti skannauksen kautta digitaaliseen muokkausaihioon Photoshopin puolelle. Luonnosteluvaihe voidaan tietysti tehdä jo suoraan piirtopöytää ja kuvanmuokkausohjelmaa käyttäen. Tässä säästetään hieman aikaa, kun digitalisointi jää tuotantoketjusta pois. Monella kuitenkin manuaalinen, kynällä ja paperilla tehty hahmottelu sujuu luontevammin.

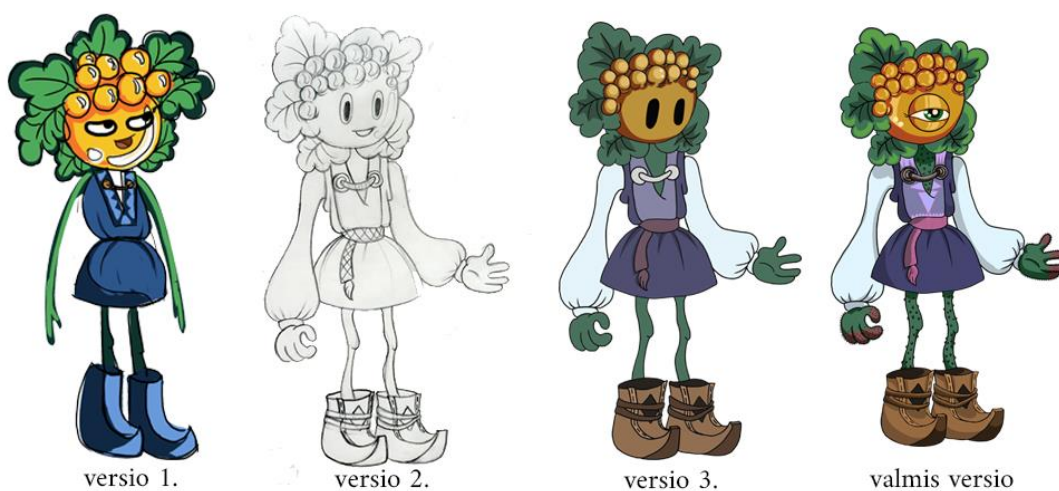
Kehitysvaiheessa kiinnitettiin huomiota vain pelaajan hahmon ulkonäköön ajattelematta toiminnallisuuksien hallintaa animaatiovaiheessa. Tämän vuoksi koko hahmon ryhtiä ja seisoma-asentoa tuli myöhemmin muokata raajojen pituuksien hienosäädön ohella. Tätä prosessia helpotti jälleen Photoshopin eri tasoihin rakennettu grafiikka.

Hahmoa suunnitellessa lähdetään liikkeelle perusmuodoista. Tämä nykyään pitkälti vain 3D-artistien käyttämä tapa juontaa juurensa pitkälle taiteen historiaan. Tutkimuksissa on huomattu, kuinka ihminen reagoi tiettyihin muotoihin ja mitä ne tuovat katsojalle mieleen. Kun tarkastellaan geometrisia perusmuotoja, niistä voidaan saada mielikuvia jo ilman, että niille on lisätty mitään erityispiirteitä tai värejä. Mitä miellelyhtymiä tuovat mieleen ympyrä, neliö tai kolmio? Ympyrä on ystävällinen, lapsekas ja tuttavallinen. Esimerkkinä on esimerkiksi Volkswagen Kupla, jossa muotoilu on hyvin pyöreää ja laineikasta. Tove Janssonin Muumi-hahmot viestivät sympaattisuutta pulskeilla ääriviivoillaan, ja ne ovat helposti lähestyttäviä. Neliö on puolestaan voimakas, järkkymätön hahmo, jonka puoleen käännytään neuvottomissa tilanteissa. Tähän kategoriaan mahtuvat niin supersankari Hulk, Umbrella Academy -sarjakuvasta tuttu Luther Hargreeves kuin Jeepin maastoautojen muotoilu: ne ovat kovatahtoisia ja luotettavia. Viimeinen, kolmio, on hyökkäävä, maskuliininen ja epävakaa. Piikikkaita, teräviä kulmia sisältäviä hahmoja nähdään paljon vihollisten designissa, kuten ilkeä Doctor Neo Cortex Crash Bandicoot-pelisarjassa [25]. Tarkoituksena oli tehdä peligrafiikasta sekä helposti lähestyttävä että hieman humoristisesti synkähkö, joten hahmoa päätettiin lähteä toteuttamaan pyöreiden muotojen kautta.

Seuraava askel oli lähteä hankkimaan inspiraatiota hahmon ilmeelle. Moodboardit eli inspiraationa toimivat leikekokoelmat ovat tärkeitä oikeanlaisen tunnelman luomisessa. Kaikki nykyisin kuuluisat hahmot aina Mikki Hiirestä Fortnite-pelin pelihahmoihin ovat saaneet inspiraatioita artistien löytämisestä referenssikuvista, eikä vaikutteiden saamista tarvitse pelätä, kunhan ei suoraan kopioida jo tehtyjä hahmoja ja omista niitä itselle. Hahmojen ulkomuoto voi vaihtua myös lennosta, kuten kävi myös tässä projektissa. Ensimmäisenä versiona tehty fauni-fantasiaolento (kuva 14) vaihtui lakkapäiseen sarjakuvahahmoon (kuva 15). Myös vaihtuneen hahmon ulkomuoto kävi läpi erilaisia versioita ennen valmista lopputulosta.



Kuva 14. Pelaaja-hahmon ensimmäisen version luonnoksia ja liikkuvuushahmotelmia.



Kuva 15. Hahmon versiot ja lopullinen tulos.

Jos olisi lähdetty käyttämään ihmishahmomaista spriteä, ensimmäisenä ongelmana olisi tullut vastaan sen melkein edestä kuvattu asento. Koska peli on sivusta päin kuvattu, hahmon animointi ja liikkuminen eivät olisi sopineet projektin perspektiiviin.

Samankaltainen prosessi käytiin läpi myös pelin taustan suunnittelussa. Cupheadin tekijöistä inspiroituneena maalattiin konseptikuvia sekä käsin että piirtopöytää käyttäen (kuva 16). Tulevan pelin visuaalinen teema ja alkuperäinen suunnitelma oli käyttää näitä

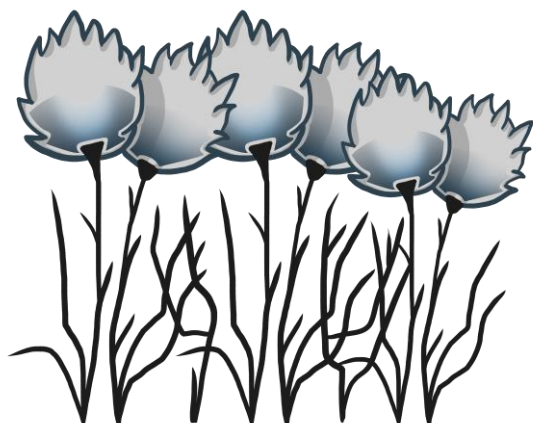
käsinmaalattuja töitä pelin taustana. Ne eivät kuitenkaan sopineet sivulta kuvatun pelin taustalle perspektiivin vaihtelujen takia. Työn edetessä nämä maalaukset jäivät pois ja ovat nyt vain osa hahmotelman prosessista jäänyttä todistusaineistoa.



Kuva 16. Suunnitteluprosessin aikana maalattu konseptikuva pelikentästä.

Konseptikuvien valmistuttua lähdettiin piirtämään prototyyppiä pelin mapista eli sitä, mihin erilaiset maa-elementit sijoitetaan kentän rakenteessa, missä pelaaja pääsee kävelemään, mitä vaaroja kentässä voisi piillä ja mitä taustalle ja ympäristöön halutaan tuoda rakentamaan pelin miljööttä. Ideoiden muuttuminen ja muokkaaminen on arkipäiväistä pelinteossa.

Monet alkuperäiset ideat tuntuivat luontevilta, kunnes ne vietiin Unityyn, jossa moni valmis sprite pudotettiin pois prosessista, esimerkkinä kuvan 17 sprite, jonka oli tarkoitus sijoittua osaksi kentän etualaa. Näitä poisjääneitä elementtejä kertyi projektin aikana runsaasti, kun ideat kehittyivät ja elementtejä karsittiin lopullisesta tuotteesta.



Kuva 17. Prosessin aikana tehty sprite, joka jätettiin pois lopputuotoksesta.

#### 4.2 Luonnoksista sprite-grafiikaksi

Kun hahmon valmista versioita aletaan koota grafiikkaohjelmassa, on olennaista ajatella jo tässä vaiheessa mahdollista animointia. Jokainen liikkuva osa hahmossa, kuten muissakin elementeissä tehdään erillisille tasoille, jotta spriten palastelu on vaivatonta animaatiolle sopivan kuvaformaatin teossa. Hahmolle piirrettiin ääriviivat, ja jokaista hahmon osaa alettiin työstää erikseen omalle tasolleen. Projektiin luodussa hahmossa erillisiin osiin tehtiin pää, torso, kädet ja jalat. Animoinnista ja artistin näkemyksestä riippuen voidaan hahmo pilkkoa myös pienempiin osiin, jos halutaan jokaisen nivelen kohdalla erillinen liike. Mahdollisuuksia olisi siis ranteiden, kyynärpäiden tai jopa sormien erillinen animointi realistisien liikkeiden luomiseksi. Työ päätettiin kuitenkin aloittaa yksinkertaisemmista liikkeistä, jotka sopivat pelin naiiviin ulkonäköön.

Hahmon teossa on otettava huomioon myös kuvaformaatin koko. Muistisääntönä on tehdä kuvasta ensin liian iso, koska kutistaminen on helppoa, kun taas valmiiksi liian pienestä formaatista on mahdotonta saada laadukasta lopputulosta. On kuitenkin suositeltavaa tallentaa valmis sprite oikeassa koossa, jotta Unityssä skaalausta tarvitsisi tehdä mahdollisimman vähän muistin säästämiseksi. Jotta jokainen sprite saatiin samankokoiseksi ja sopivaksi peliin, käytettiin HD-kokostandardia (2160 x 1080 px) yhtenäisen lopputuloksen saavuttamiseksi.

Kuvaformaatin, johon hahmon grafiikka halutaan tallentaa, tulee olla spritelle ominainen, taustatonta kuvaa tukeva formaatti. Paljon valokuvien tallennuksessa käytetty JPEG-tiedosto ei sovellu tähän tarkoitukseen sen automaattisesti taustan täyttävän ominaisuuden takia. JPEG myös pakkaa kuvaa ja saa helposti kuvan värit suttuisiksi. Aikaisemmin Unityssa käytettiin paljon PNG-spritejä, mutta rinnalle on nyt saatu Photoshopin PSD-formaattia tukeva lisäpaketti. Tämä formaatti toimii hyvin juuri hahmon tuonnin ja rakennuksen kanssa. Unity tunnistaa PSD:ssa olevat erilliset tasot ja slice-työkalun avulla saa ne erillisiksi osiksi, vaikka ne olisivat päällekkäin, osittain tai kokonaan toistensa alla. PNG:ssa tätä ei ole, vaan se litistää kuvan yhteen tasoon tallennusvaiheessa. Jos PNG:tä halutaan silti käyttää, tulee kaikkien osien olla eriteltynä toisistaan Photoshopissa ennen sen tallentamista. Tämä tulee muistaa myös muiden peliin tuotavien spriteiden tekovaiheessa. Myös kuvan rajausta on hyvä suorittaa niin, ettei kuvan ympärille jää ylimääräistä taustaa. Kun rajausta on tiivis, se helpottaa Unityn rigidbodin automaattista luomista eikä tee elementin ympärille liian suuria rajoja, joita on käsin muokattava spriteen sopivaksi.

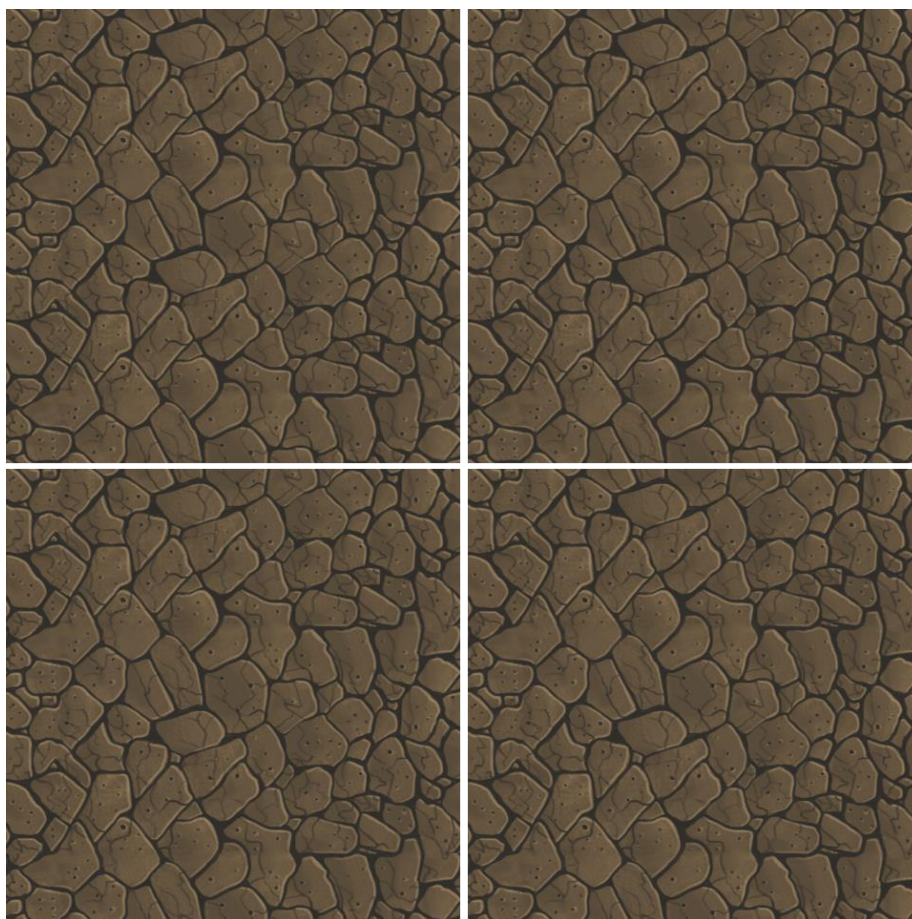
Koska peliin haluttiin tehdä sekä tilemapeja että spritejä, päätettiin aloittaa seamless-tekniikkaan syventymällä. Seamless tarkoittaa saumatonta, ja tämä kuvaakin hyvin grafiikan käyttäytymistä monistamisessa ja skaalauksessa. Tekstuuri rakennetaan niin, että sitä vierekkäin kopioidessa ei muodostu näkyviä rajoja tiilien reunoille. Tekstuuria voidaan monistaa tasaisesti niin suureksi alueeksi, kuin on tarpeen.

Seamless-tekstuuri voidaan tehdä niin käsin kuin valmiista valokuvastakin. Valokuvia käytetään paljon 3D-mallintamisessa, kun halutaan mahdollisimman realistinen lopputulos. Valokuvista toteuttamalla tulee ottaa huomioon kuvanottohetkellä kuvakulma, jonka tulisi olla mahdollisimman kohtisuoraan edestä. Myös linssivääristymiä tulisi välttää, jotta säästytään turhalta työltä Photoshopin puolella.

Paneudutaan kuitenkin piirtämällä toteutettuun teksturiin. Tekstuuria on helpointa käyttää, kun se vastaa kuvasuhdetta 1:1. Näin tuleva monistaminen onnistuu parhaan lopputuloksen saamiseksi. Värivaihteluja tulee käyttää harkiten eikä tehdä tietyistä yksittäisistä kohdista liian erilaista muihin kuvioihin verrattuna. Niin vältetään liiallinen toistuvuus, joka pistää silmään lopullisessa tuloksessa. Kuvion, joka halutaan

muodostaa, tulee olla myös mahdollisimman yhdenmukainen, mutta satunnainen elävän pinnan luomiseksi.

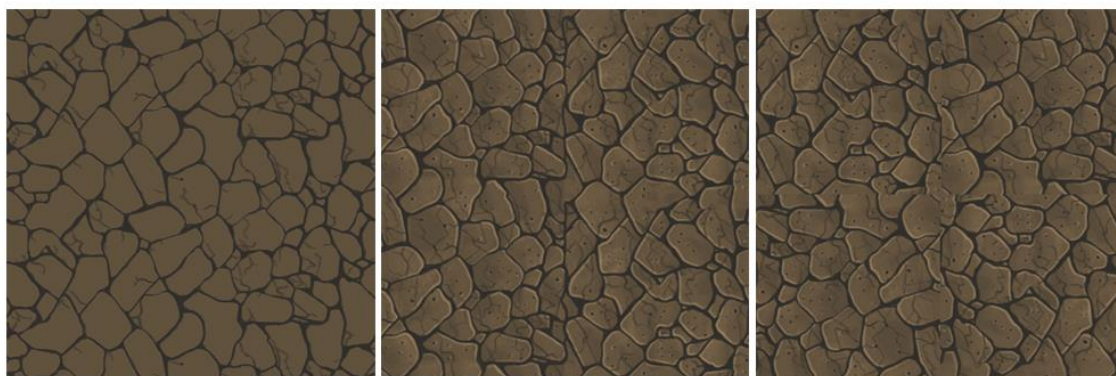
Esimerkkinä toimiva kivitekstuuri (kuva 18) on toteutettu käyttämällä mahdollisimman paljon samankokoisia kuvioita. Jos tekstuuriin esimerkiksi tehdään yksi, muista kuvioista erottuvan muoto, se pistää silmään myöhemmässä vaiheessa. Varjojen ja korostuksien tulee olla yhtenäisiä ja samassa suunnassa olevia. Kuvakulmaksi halutaan tässäkin mahdollisimman kohtisuora, jotta lopputulos toimii. Jos tekstuurista tehdään vaikkapa kupera, pelimoottorissa monistettava tekstuuri toistaa pieniä kumpareita eikä lopputulos näytä tasalaatuiselta. Jos tekstuuriin haluaa lisätä yksityiskohtia, kannattaa tekstuurista tehdä erillisiä versioita, joihin yksityiskohtia lisätään ja käyttää tätä harkiten lisäämällä niitä sekalaisesti sinne tänne kenttään.



Kuva 18. Laittamalla seamless tekstuurin palasia vierekkäin, nähdään kuinka kuvio toistaa itseään saumattomasti monistamisesta huolimatta.



Kun tekstuuri on värien ja muotojen kannalta valmis, otetaan käyttöön Photoshopin valikosta löytyvä Offset-toiminto. Valittavissa ovat sekä horisontaalisesti että vertikaalisesti kuvaa vierittävät toiminnot. Kun halutaan tehdä sekä vaaka että pystysuorassa saumattomasti toimiva tekstuuri, tulee molempia työkaluja käyttää. Offset-toiminto siirtää kuvan reunat keskelle tehden ristimäisen kuvion tekstuuriin keskelle (kuva 19). Hahmotuksen vuoksi voidaan ajatella paperilla olevan neliön muotoisen kuvan, jonka reunat taitetaan taaksepäin 90 asteen kulmassa selustan keskipisteeseen. Nyt reunat ovat saumattomia, mutta kuvan keskikohdat eivät täsmää keskenään. Tätä lähdetään korjaamaan peittämällä keskelle muodostunutta ristikuviota samoilla työvälineillä ja väreillä kuin aikaisemmin aikaansaadussa tekstuurissa.



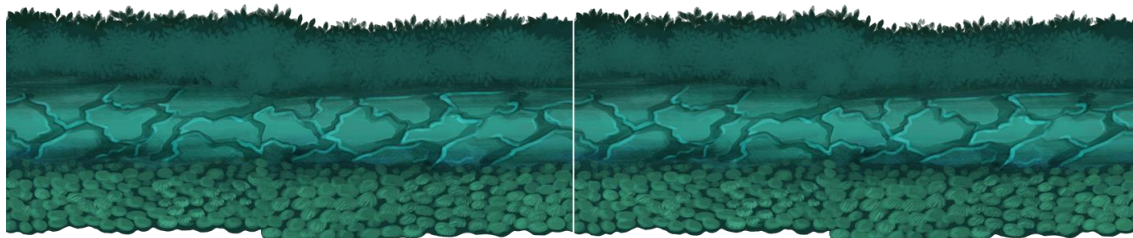
1. Taustaväri ja ääriiviivat    2. Varjot ja korostukset    3. Offset - filter

Kuva 19. Seamless-tekstuurin työvaiheita [26].

Jos kuvio on tehty alun perin käyttäen apuviivoja tai ruudukkoa, häivytyks onnistuu suhteellisen helposti, kun ääriviivat ovat samalla korkeudella. Satunnaisesti toistuvien kuvioiden kohdalla joudutaan tekemään enemmän työtä, koska kuvioiden rajat eivät välttämättä kohtaa taittamisen jälkeen. On huomioitava, ettei reunoille tehdä enää muutoksia, jotta saumaton efekti pysyy tasaisena.

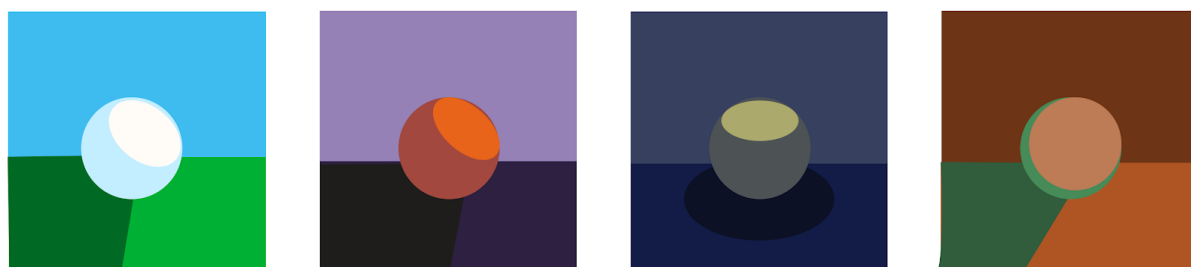
Sprite-pohjaisissa elementeissä kohdalla ei tarvitse ottaa huomioon juurikaan muuta kuin perspektiivin pysyminen mahdollisimman yhdenmukaisena tile-tekstuuriin kanssa. Offset-työkalua voidaan käyttää myös sprite-grafiikan tekemisessä. Peliin tehty polku, kuvassa 20, on käytetty myös offset-taitossa, mutta vain x-akselia muokaten. Kuvasta muokataan saumattomaksi vain pystysuoraan tullut sauma skaalauksen onnistumiseksi.

Kun kuvan molemmat reunat ovat saumattomat, voidaan spriteä venyttää vaakatasossa valitsemalla tiled-työkalu Sprite Render -valikosta. Näin samaa spriteä voidaan käyttää niin pitkän kappaleen verran kuin on tarvetta. Kuten hahmon kuvaformaatti, tulee spriteillekin valita sprite-pohjaa vastaava formaatti.



Kuva 20. Sprite-elementtejä voidaan tehdä saumattomaksi myös pelkästään yhden akselin suuntaisesti. Polun osaa toimittava grafiikka on tehty horisontaaliksi saumattomaksi, jotta sen pituutta voi säädellä vaivattomasti kentän mukaan.

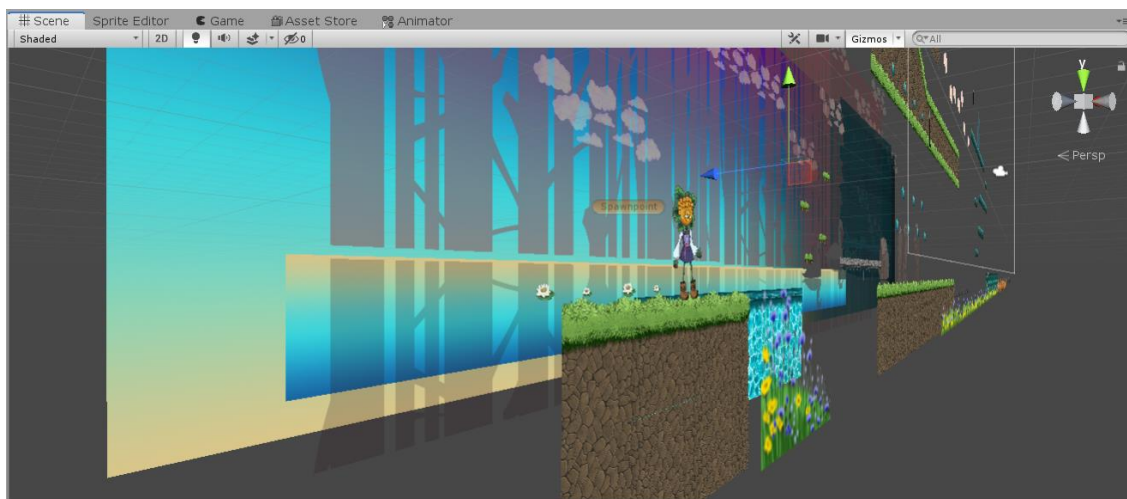
Kaikkia elementtejä tehdessä kannattaa valita valmiiksi värikartta, johon kaikki elementit väritään pohjautuvat. Värikarttaa tulee miettiä jo hahmotusvaiheessa, jotta säästytään liialta jälkeempään muokkaukselta ja lisätyöltä. Verkosta ja kirjoista kannattaa etsiä erilaisia värikarttoja, jotka on rakennettu erilaista tunnelmaa silmällä pitäen. Mallia voi myös ottaa kuvaamalla ulkona erilaisissa säätiloissa ja eri kellonajoin, tutkien, minkäväriset ovat esimerkiksi valot ja varjot ilta-auringossa tai pilvisellä säällä. Kuvassa 21 havainnollistetaan erilaisten värimaailmojen tunnelmanluontimahdollisuuksia. Kokonaisuus saadaan sidottua yhteen, kun jokainen elementti ei ole eri värimaailmasta. Sekä Photoshopissa että Unityssa pystytään sävyttämään spriteja jälkikäteenkin, mutta lopputulos on yleensä kalvomainen ja epäluonnollinen.



Kuva 21. Värikaavioesimerkkejä erilaisten tunnelmien luomiseksi.

Pelin taustana voidaan käyttää mitä vain oikeankokoista grafiikkaa tai vaikkapa valokuvaa. Taustakuvastakin voidaan tehdä saumaton niin haluttaessa. Jos peliin

halutaan tehdä luvussa 2 läpikäyty parallax scrolling -efekti, tehdään taustalle osin läpinäkyviä, erikokoisia sprite-elementtejä kolmiulotteisen vaikutelman aikaansaamiseksi. Erilliset tasot tuovat peliin myös eloa ja syvyyttä, mikä jää pelkän yksivärisen taustan ja tasojen uupumisen kautta puuttumaan. Pelimoottorissa otetusta kuvankaappauksesta (kuva 22) voidaan nähdä, kuinka erillään olevista tasoista voidaan toteuttaa syvyysvaikutelmaa ja parallax scrolling -tekniikkaa.



Kuva 22. Unityn peliympäristöstä kaapatusta kuvasta nähdään eri tasojen sijoittelu 2D-pelimaailmassa.

Peliin haluttiin luoda sekä tilemapeina toimivina saumattomia tekstuureja että perinteisiä spritejä, jotta nähtiin, mitä ominaisuuksia ja eroavaisuuksia näillä kahdella tekniikalla on. Teknisesti näiden kahden tekniikan sekoittaminen Unityssa onnistuu huoletta, mutta niiden haluttiin sopeutuvan yhteen myös visuaalisesti. Jotta tämä onnistui, tuli tiili-tekstuurien rigidbodya rajata kuvan ylärajaa pienemmäksi. Tämä saa aikaan efektin, jossa pelaaja ei kävele suoraan kuvan yläreunan päällä vaan pelivaiheessa näkymättömällä janalla. Näin saadaan aikaan kolmiulotteinen perspektiivi-efekti, jossa pelaajan hahmo näyttää kulkevan tekstuurin sisällä.

Graafisten käyttöliittymäelementtien tekovaiheessa kiinnitetään huomiota, miten ne toimivat Unityssa. 9-Slice-työkalun käyttöönottoa kannattaa miettiä niitä tehdessä. Peliin tarvittiin ainakin reunukset erilaisten vierityspalkkien kehyyksiksi. 9-Slice-tekniikkaa tukevaa spriteä voidaan käyttää erikokoisiin käyttötarkoituksiin, eikä erikokoisille kehysvalikoimille ole tarvetta. On hyvä myös tehdä tyhjiä, ilman tekstejä olevia

painikkeita, jotka toimivat sekä alkuvalikon että pause-menun kuten muidenkin painikkeita tarvitsevien elementtien taustaksi. GUI-elementtejä tehdessä otetaan huomioon, kuinka ne tulevat erottumaan pelistä. Ne eivät saa olla liian hallitsevia, jotta pelaajan huomio ei keskity reunassa sijaitseviin valikoihin, mutta samalla niiden tulee erottua tarpeeksi peliympäristöstä helpon seuraamisen kannalta. Jos 9-Slice-työkalua ei Unityssa olisi, vektorigrafiikkaa toteuttavat elementit olisivat toimivia käyttöliittymän osien teossa. Näin skaalaus ei vähentäisi kuvan terävyyttä ja yhtä elementtiä voitaisiin käyttää moneen eri tarkoitukseen.

Valikoille tarvittavat kehykset tehdään 9-Slice-tekniikan periaatteita käyttäen. Reunuksen kulmiin jäävissä palasissa voidaan käyttää yksityiskohtia, jotka eivät muuta kokoaan skaalauksessa. Reunat skaalautuvat xy-akselissa, joten niiden toistuvuutta kannattaa pitää silmällä ja jättää liialliset yksityiskohdat pois. Jos kehyksiin halutaan liittää myös taustaa, sen tulee olla tarkoitukseen sopiva, joko täysin seamless-tekniikalla toteutettu grafiikka tai yksivärinen kuva. Jos taustalla halutaan käyttää esimerkiksi liukuväriä, taustaa ei voida skaalata kuin sivusuunnassa.

#### 4.3 Pelin kokoaminen pelimoottorissa

Ensimmäinen askel pelimoottorissa oli tuoda peliin spritet ja aloittaa taselementteinä toimivien palasten peliympäristöön tuonti. Design-vaiheessa luotu pelikartta otetaan avuksi, jos pelin rakennetta on suunniteltu muunkin kuin pelkän grafiikan puolesta. Vaikka saumattomat tekstuurit tehtiin sopiviksi myös grid-käyttöön, haluttiin pelikentän rakentamiseen käyttää manuaalisia työkaluja sprite-elementtien takia. Lopputuloksesta ei haluttu visuaalisesti liian graafista ja yksiulotteista, kuten tilemap-pelit yleensä ovat. Koska sprite-pohjaisissa peleissä kentässä tulee olemaan paljon tasoja, kannattaa niille tehdä omat tyhjät gameobject-kansiot, joihin spritet tallennetaan. Tässä vaiheessa on hyvä tehdä hierarkiatasot taustalle, etualalle, pelaajalle sekä GUI-elementeille. Tämä helpottaa pelinrakennusta niin järjestelyn kuin ohjelmoinnin osalta.

Hahmon animoinnissa kokeiltiin vielä ennakkajulkaisuna toimivia 2D Animation- ja IK Animation -paketteja. Näiden tekniikoiden pohjalla toimivat hahmolle slice-editorissa muodostettava luuranko ja nivelet, jotka helpottavat hahmon liikuttelua animoinnin

osalta. Tämä nopeuttaa animointitekniikkaa, joka toteutetaan erillisinä palasina eroteltujen raajojen liikuttamisella. Paketit olivatkin pienistä ohjelmistovirheistä huolimatta suhteellisen helppoja käyttää. Niiden seuraksi tuli ladata myös PSB import -paketti, jotta animaatio onnistui erittelemään tasot toisistaan. Projektiin ei kuitenkaan haluttu käyttää liikaa vielä kehitysvaiheessa olevia paketteja niiden epävarmojen toimivuuksien vuoksi.

Hahmosta päätettiin tehdä animaatio piirtämällä neljästä asennosta muodostuva liikerata, jota Animation Editoriin tuotaessa toistamalla saatiin hahmolle erilaisia liikkeitä. Kuvassa 23 näkyvällä hahmolla on neljästä asennosta koostuva kävelyanimaatio, idle-animaatio ja hyppyanimaatio. Animointia voidaan lisätä niin moneen elementtiin, kuin suinkin halutaan ja aikataulu kestää. Peli herää eloon taustalla loopilla toistuvilla animaatioilla, kerättävien elementtien värähtelyillä ja keinotekoisessa bittiavaruustuulessa heiluvilla puiden oksilla.



Kuva 23. Animaatioarkki hahmon liikesarjasta.

Animoitaessa otetaan huomioon, kuinka monen asennon kierrättämisestä on kyse "frames per second" -kuvataajuutta valittaessa. Vain neljästä kuvasta muodostuva animaation toistuvuuden vauhti kasvaa, kun ruutuja lisätään. 12 ruudun määrä toimii hyvin tässä tarkoituksessa. Jos kävelystä halutaan tehdä nopea, juoksua simuloiva, kuvaruutujen määrää tulee lisätä entisestään. Liikkeen vauhti kannattaa sovittaa hahmon liikenoiteen sopivaksi. Jos hahmon liikutus on hidastempoista, nopeasti liikkuvat raajat saavat aikaan humoristisen vaikutelman. Hyppyanimaation tekemiseen voidaan haluttaessa käyttää vain yhtä spritea. Tämä animaatio toistetaan pelaajan irrotessa maasta, jonka hahmon groundcheck-objekti laskee. Animator-editorissa voi antaa animaatioille haluttuja ehtoja, jotka määrittelevät, milloin animaatio pyöritetään. Tässä tapauksessa kävely toistetaan, kun pelaajan nopeus ylittää tietyn summan, ja vastaavasti idle-animaatioon siirrytään, kun nopeuden määrä lähentelee nollaa. [27.]

Erilaisia liikkuvia efektejä voidaan tuoda peliin myös partikkelianimaation myötä. Nämä yhtä tai useaa muotoaan vaihtavat kuviot pyöritetään uudelleen editorissa annetuin reunaehdoin. Näin voidaan tehdä pelikenttään esimerkiksi lumi- tai vesisadetta, ampuessa ilmaantuvia kipinöitä tai viholliseen osuvien luotien räjähdysisiä. Nämä lähinnä lisätunnelmaa luovat efektit kannattaa jättää viimeisenä lisättäväksi, jotta jää aikaa pelin pelattavuuden tekoon ja hiomiseen.

Aloittelijan pelinrakentamisen apuna kannattaa käyttää kirjoissa ja verkkossa tarjolla olevia tutoriaaleja. Pelin kokoamisen apuna käytettiin "Brakeys"-nimellä toimivan pelintekijän ohjevideoita, jotka ovat suosittuja ja pidettyjä [28]. Nykyään Unityssakin monet toiminnot on saatu automaattisiksi, eikä pelinrakentajan tarvitse välttämättä osata ohjelmointia ollenkaan toimivan pelin rakennuksessa. Unityyn on myös saatavilla blueprints-paketti, jolla luodaan graafisia sinikopiota pelin toiminnallisuuksista. Tämä helpottaa tekijöitä, jotka tarvitsevat visuaalisia esimerkkejä ja karttoja tietyn elementin toimivuudesta. Epic Gamesin pelimoottori Unreal Engineessä tämä työkalu on upotettu alkuperäisiin asetuksiin, eikä sitä tarvitse erikseen asentaa.

## 5 Toteutettu grafiikka

### 5.1 Työntekoprosessi ja tulokset

Insinöörityön hahmotus aloitettiin alkuvuodesta 2019, ja kevään myötä mukaan mahtui onnistumisia ja myös epätoivon tunnetta. Hahmottelu ja työn kaavoitus vaati eniten aikaa koko prosessista, kun haluttiin varmistaa mahdollisimman pitkälle ja hyvin suunniteltu projekti ennen konkreettisten töiden aloittamista. Tässä ei kuitenkaan aivan onnistuttu liian monen idean ja epärealististen toiveiden vuoksi. Kuitenkin suunnittelusta saatiin myös hyviä tuloksia, kuten kuvassa 24 näkyvä projektin alkuvaiheessa käsin piirretty hahmotelma pelin sprite-elementistä ja sen valmis versio käytettäväksi peliprojektissa. Turhaa materiaalia ja käyttämätöntä grafiikkaa syntyi projektin aikana runsaasti. Työkansioon ilmestyi projektin päätteeksi 364 eri kuvaformaatteja edustavaa grafiikkaelementtiä, joista lopulliseen työhön käytettiin alle 50:tä. Toki seassa on myös samasta kuvasta erikokoisiksi ja muodoltaan erilaisiksi tallennettuja versioita. Tässä kuitenkin nähdään, kuinka paljon työtä tehdään ja kuinka paljon lopputuloksesta karsitaan osia.



Kuva 24. Vaikka suunnitteluvaiheessa tehtiinkin turhaa työtä, myös positiivisia tuloksia saatiin aikaan projektin edetessä. Kuvassa vasemmalla hahmottelu ja oikealla valmis versio sprite-elementistä, joka päättyi käytettäväksi työhön.

Grafiikan käyttöönottoa vähensi myös aikataulun tiukkuus: kaikkia suunniteltuja palasia ei ennätetty lisäämään peliin ohjelmoinnin osalta. Esimerkiksi kuvassa 25 nähtävästä käyttöliittymäelementtipaketista osa jäi pelissä kokonaan käyttämättä ajan loppumisen

myötä. Toimivat palaset sisällytetään joka tapauksessa osaksi työtä, jotta niitä voidaan haluttaessa hyödyntää muissa projekteissa.



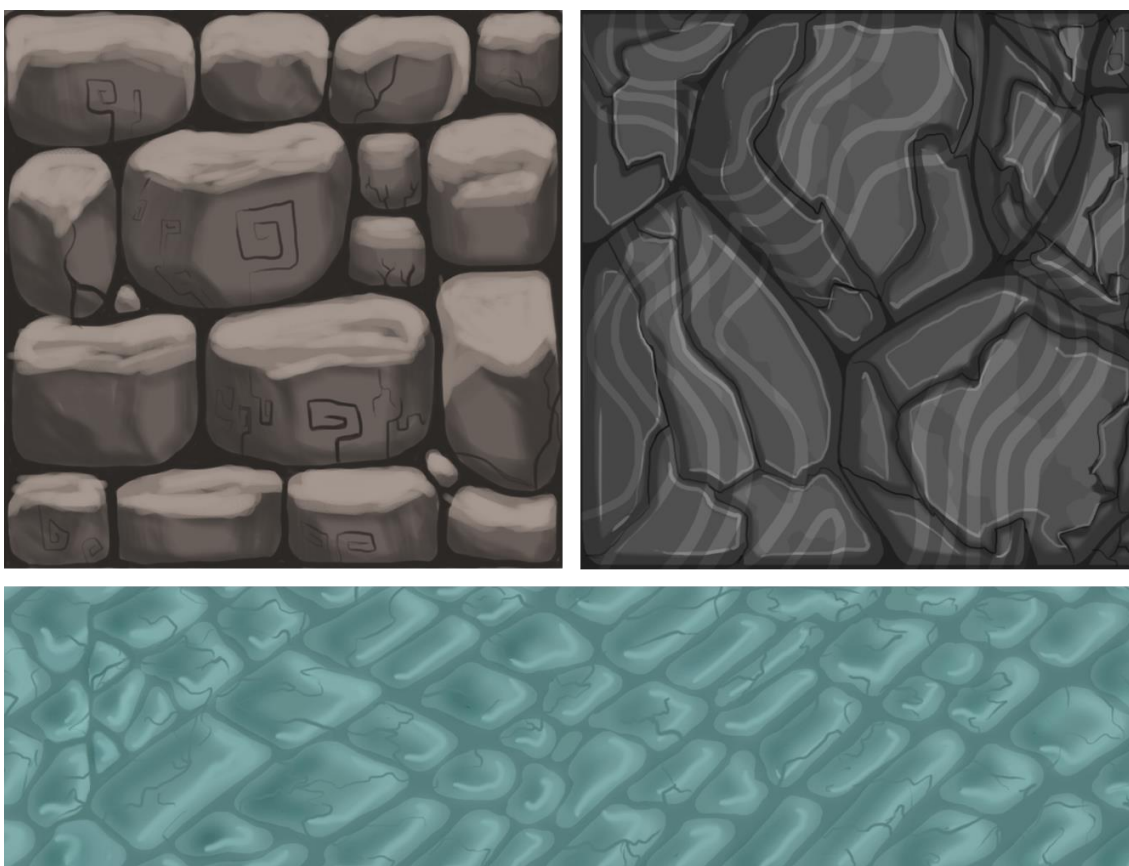
Kuva 25. Peliin tehtyjä GUI-elementtejä, joista osa ei koskaan päätnyt lopulliseen Unity-projektiin.

Myös työn sisältö muuttui projektin edetessä ja painottui enemmän grafiikkaohjelmien puolelle. Alkuperäisenä tarkoituksena oli kiinnittää enemmän huomiota pelimoottorissa käytettäviin työkaluihin ja niiden ominaisuuksiin. Työtä varten oli etsitty erilaisia grafiikan työstämiseen liitettäviä lisäosia, joita Unityyn voi ladata pakettimanagerin kautta. Listalta löytyi aikaisemmin mainittu 2D Animation -paketti, jolla pystytään muodostamaan animaatiolle oma luuranko sen liikuttamisen helpottamiseksi, Inverse Kinematics -paketti, joka laskelmoi animaation nivelten välistä liikkuvuutta haluttavissa liikeradoissa [29], ja Sprite Shape -paketti, jolla tehtyä grafiikkaa voidaan käsitellä eri muotoihin sille dynaamisesti laskettujen viivainten avulla. Suoralle kappaleelle voidaan käyrien ja ankkuripisteiden avulla muodostaa pehmeitä kulmia [30]. Pakettien käytön ongelmana olivat kuitenkin niiden vielä työstövaiheessa olevat versiot, eikä pelkästään jo valmiiden lisäosien käyttäminen työssä olisi tuonut sille minkäänlaista lisäarvoa tutkimustyön osalta.



Projektin päämäärä oli siis tuottaa mahdollisimman moniin käyttötarkoituksiin sopivia grafiikkaelementtejä, joita voidaan hyödyntää teknisten työkalujen testauksessa ja peliprojektien grafiikkana. Tämä saatiin tekemällä grafiikkapakettiin teknisesti käyttäytyviä tekstuureja ja spritejä sekä hyvä valikoima eri formaateissa tallennettuja palasia, jotka sopivat eri tarkoituksiin pelinrakennuksessa.

Seamless-tekniikalla toteutettua grafiikkaa olivat kentän maaelementit, joihin kuului erinäisiä palasia täytetekstuurista maan pintaosiin ja reunapalasiin. Kuten muistakin osista, myös kivitekstuurista tehtiin projektissa erilaisia versioita (kuva 26) ennen lopullisen, eniten silmää miellyttävän tekstuurin valikoitumista työhön.



Kuva 26. Valmiiden grafiikkaelementtien tekovaiheessa syntyy usein erilaisia testiversioita. Tässä nähdään kolme kivitekstuuria, jotka tehtiin ennen lopullista, pelissä käytettyä elementtiä. Näitä tekstuureja ei ole tehty seamless-tekniikkaa vastaaviksi, mutta niitä pystyttäisiin haluttaessa hyödyntämään muissa käyttötarkoituksissa.

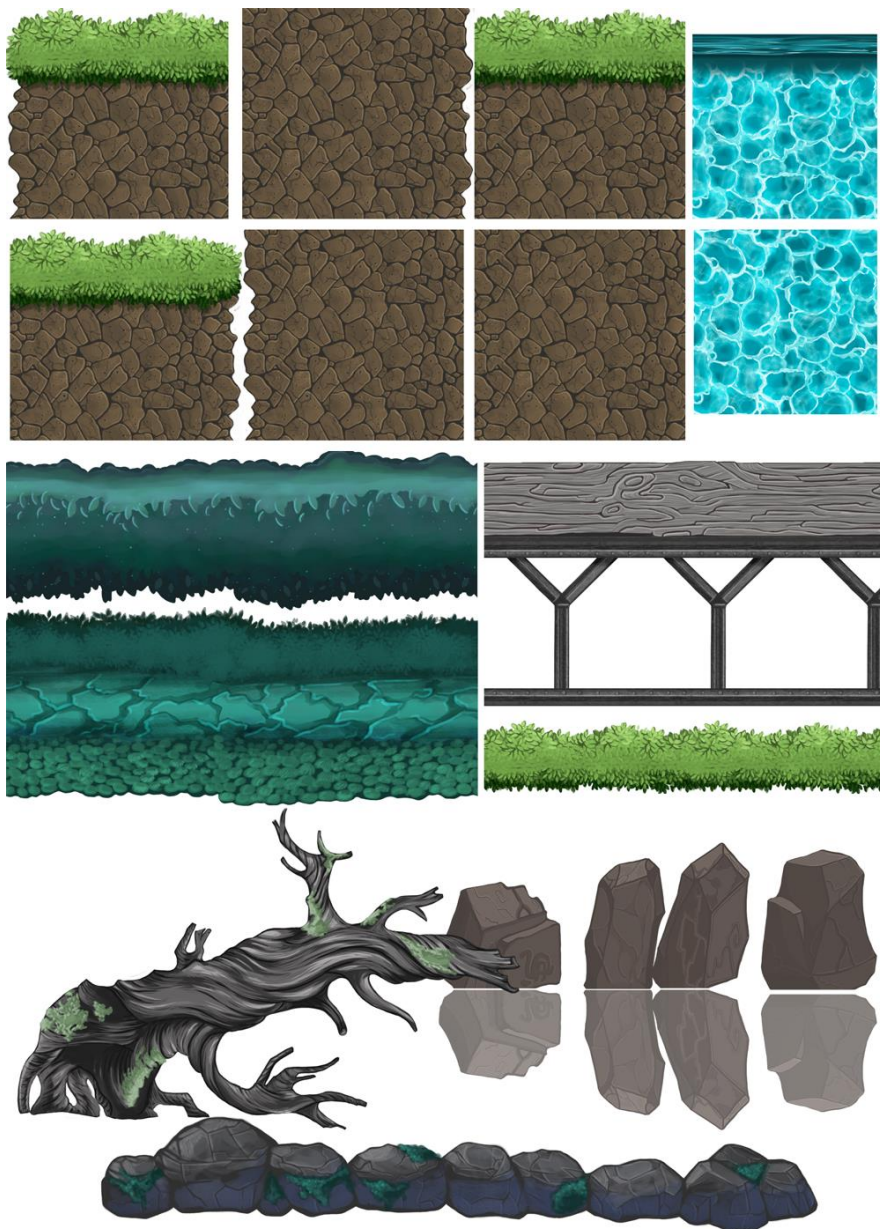
Valmiiseen työhön tehtiin lopulta enemmän sprite-elementtejä kuin saumattomia tekstuureja. Toisin kuin oletettiin, spriten tekoon kului huomattavasti enemmän aikaa kuin seamless-elementteihin, jotka tuli toteuttaa omien reunaehtojensa mukaisesti. Sprite-elementit vaativat enemmän yksityiskohtien ja niiden määrittelemättömien muotojen hiomista toimiakseen pelimoottorissa sivustakuvattujen tekstuurien rinnalla. Myös kunnianhimoisuus ja pikkutarkka viimeistely vei aikaa grafiikan toteutuksessa. Esimerkiksi taustalle sijoittuva metsäelementti (kuva 27) oli projektin teknisen puolen kannalta turha ja tuo peliin ainoastaan visuaalista lisäarvoa.



Kuva 27. Taustalle sijoitettu metsätekstuuri on visuaalisesti näyttävä, mutta ei välttämättä tarpeellinen ainakaan pelin toimivuuden tai toiminnallisuuden kannalta.

Kaiken kaikkiaan valmiiseen peliin sisällytettiin kaksi erilaista saumatonta tekstuuria, maa-aines ja vesitekstuuri, ja niille tehdyt reunapalaset käyttöalueen ympärille. Niiden rinnalle tehtiin niin sanottuja leijuvia maaosia, joita pystyttiin hyödyntämään kentän rakennuksessa. Horisontaalista tiled-skaalausta kestäviä elementtejä ovat pensastekstuuri ja siltatekstuuri, joka on jaettu sekä puiseen osaan että metallisiin rakenteisiin. Näitä voidaan käyttää haluttaessa joko yhdessä tai erikseen. Kentän

työnimeä “metsäalue” kantavassa osiossa käytettävä polkumateriaali ja ilmassa leijuvat tasot pelaajan liikkumista varten toimivat myös tiled-tekniikan mukaisesti. Nämä kaikki elementit ovat esitettyinä kuvassa 28.



Kuva 28. Kollaasi peliin tehdyistä elementeistä, joista osa toimii seamless-tekniikan mukaisesti.

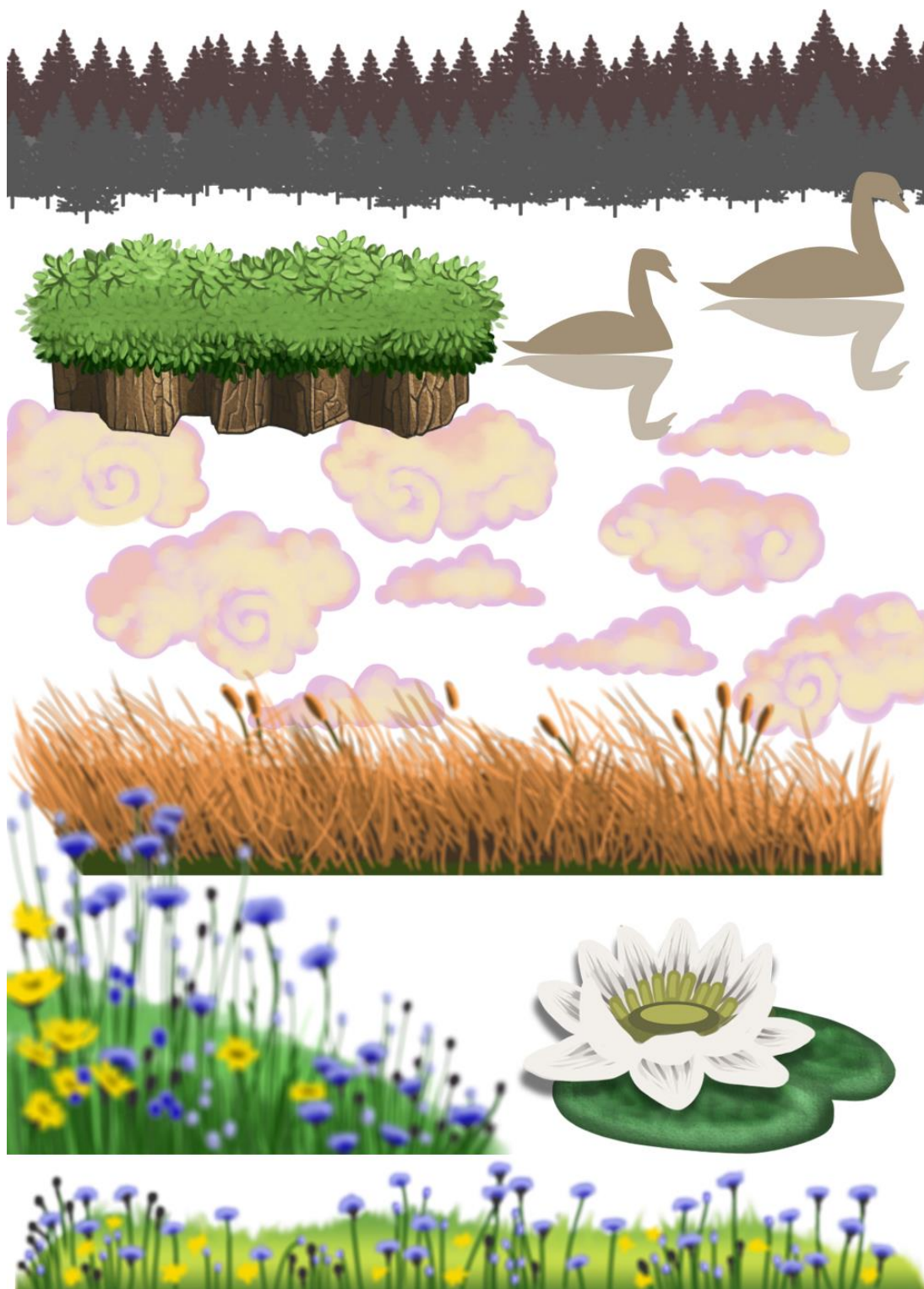
Pelaajan hahmosta tehtiin sekä PNG- että PSD-formaatin tallennukset auttamaan myöhempien käyttötarkoitusten käyttöönottamista. Hahmo jaettiin kymmeneen eri

palaseen, joita liikuttamalla niistä voidaan muodostaa animaatioita Photoshopin, Unityn tai halutun animaatio-ohjelman avulla. Siitä tehtiin myös valmis animaatioarkki, jolle hahmosta piirrettiin neljästä asennosta koostuva kävelyanimaatio, silmää räpäyttävä Idle-animaatio ja yhdestä asennosta koostuva hyppyanimaatio (kuva 29). Muita hahmoja olivat viholliset, joista tummanpuhuvalla, leijuvalle hahmolle tehtiin liikkuvaksi osaksi luurankomainen käsi. Päävihollisen käytyreinä toimiville vihtahousuolioille tehtiin myös raajojen palastelu mahdollista animointia varten.



Kuva 29. Pelin hahmojen animointi onnistuu helposti oikein tehdyillä tallennusformaateilla.

Muut osat pelissä toimivat lähinnä parallax scrolling -efektin toimimista edellyttävinä tasoina, esimerkiksi etualalle laadittiin luontoelementtejä (kuvassa 30), jotka loivat efektille lisää syvyysvaikutelmaa tuomalla grafiikan mahdollisimman lähelle kameraa. Taustalle tehty grafiikka, kuten pystymetsä ja pilvet, on eroteltu muista tasoista, ja ne toimivat myös efektin kannalta tärkeässä osassa.



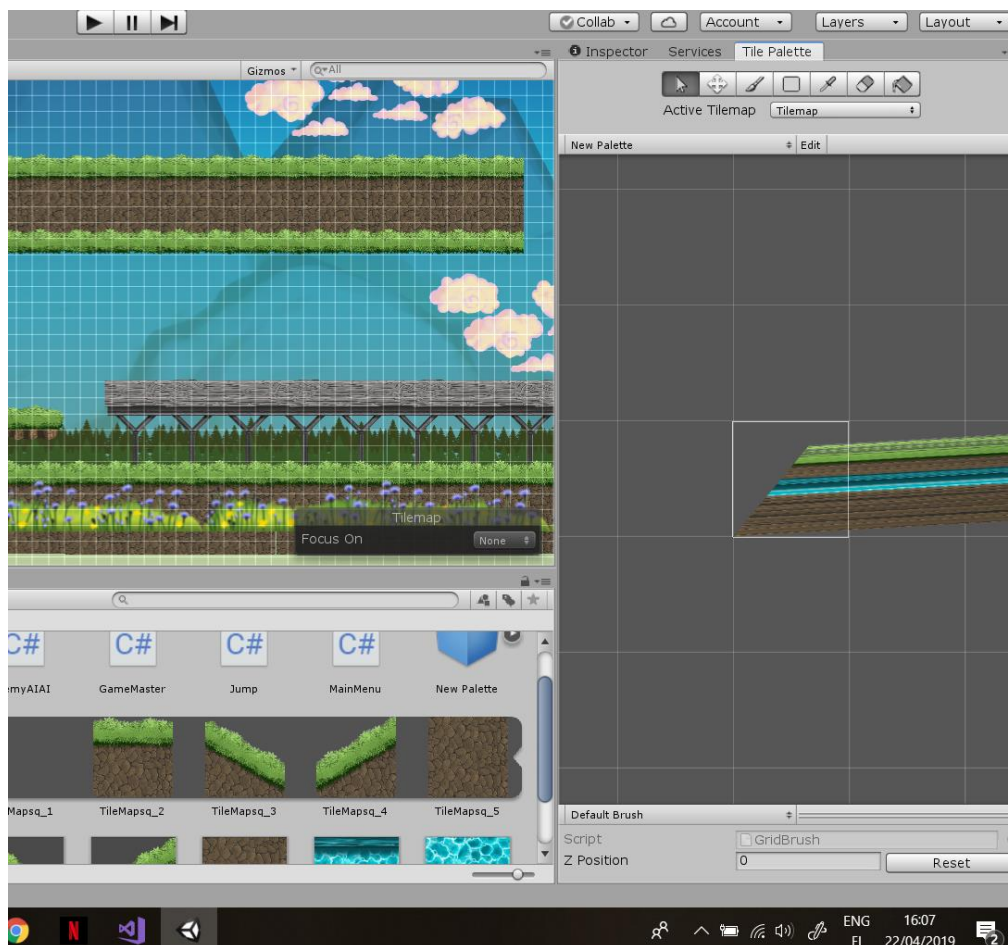
Kuva 30. Peliin tehtiin ympäristön mielenkiintoa lisääviä ja parallax scrolling -tekniikan toimimista edellyttäviä grafiikkaelementtejä.

Vielä senkin jälkeen, kun peliä oli alettu koota jo pelimoottorissa, työskentely jatkui myös Photoshopin puolella. Uusia ideoita syntyi päivittäin, ja haasteellista oli erottaa hyvät ideat turhasta, lähinnä täytteeksi kelpaavasta grafiikasta. Etualaan sijoitetut kukkaketo-elementit tehtiin loppupuolella projektia, ja ne tulivatkin tarpeeseen parallax scrolling -efektin lisäämiseksi.

## 5.2 Projektin käännekohtat ja työn arvioiminen

Vaikka luonnoksia ja esimerkkikarttoja pelistä kertyi runsaasti projektin alkutaipaleella ja ideoita areenan osien paikoista syntyi päivittäin, oli kamerakulmankin päättäminen vaivan takana. Kuitenkin hahmon valmistuttua sen piirtotyyli määritteli pelin visuaalisen lähtökohdan ja tekstuurien ulkomuotoa oli helppo lähteä toteuttamaan pelin suuntaiseksi. Seamless-tekstuureiden valmistuttua ja niitä Unityssa testaamalla nähtiin, millainen lopputulos pelistä saadaan ja millaisia elementtejä spriteistä haluttiin yhtenäisen teeman säilyttämiseksi.

Yksi suurimmista vastoinkäymisistä projektissa oli kesken projektia tehty Unityn versiopäivitys. Projektissa käytettiin pitkään Unityn vanhempaa versiota, jossa kaikki osat toimivat järjestelmällisesti. Kuitenkin "tilepalette"-työkalua kootessa huomattiin pelimoottorissa outo ohjelmistovirhe (kuva 31), joka rikkoi paletissa olevan grafiikan monista kokeiluyrityksistä huolimatta. Päivitys toteutettiin ajateltiin asian korjaantuvan suuremmista ongelmista. Päivityksen myötä edellisessä versiossa olleet lisäpaketit eivät toimineet enää ja ohjelmakoodi ilmoitti virhetilannetta toisen perään. Korjausyritysten jälkeen päätettiin paketit poistaa kokonaan käytöstä, ja samalla tuli siistittyä myös koodia.



Kuva 31. Unityn tilepalette-editorissa vaivannut ohjelmistovirhe, joka vaati pelimoottorin päivityksen korjaantuakseen.

Hieman kömpelömmiin toimiva peli saatiin herätettyä takaisin eloon ja työkalu, jota haluttiin kokeilla alun perinkin vain testimielessä, alkoi toimia. Tästä voidaan ottaa oppia päivittämällä pelimoottori joko heti projektin aloituksen yhteydessä tai sen jo ollessa valmis. Myös ennakkona julkaistujen lisäpakettien käyttöä kannattaa käyttää harkiten mahdollisten epävakauksien välttämiseksi.

Projektin myötä aikataulutuksen ja tehtävien priorisointi nousi tärkeään osaan työn etenemisen kannalta. Huomio kiinnittyi siihen, kuinka tärkeää on listata kaikki tarvittavat työtehtävät ja asettaa ne tärkeysjärjestykseen. Toki erilaisia projektinhallintatyökaluja on käytetty aikaisemmissa opiskelu- ja työprojekteissa, mutta yksintyöskentelyssä asia korostui huomattavasti. Työmäärään hukkuminen estetään keskittymällä yhteen asiaan kerralla. Myös työtehtävät on hyvä asettaa tärkeysjärjestykseen, niin ettei kulu liikaa

työlle epäolennaisten yksityiskohtien työstämiseen. Hahmotteluvaiheessa pitäisi tehdä tarkemmat suunnitelmat ja runko pelin toteutukselle, jotta vältetään mahdollisimman monen turhan työn tekemiseltä.

Monesti työn edetessä huomataan turhautuminen omaan työnjälkeen ja sitä pidetään kelvottomana, vaikka edellisenä päivänä innostus ja ylpeys omista aikaansaannoksista olivat päällimmäisiä tunteita. Tämä sama tilanne tulee varmasti jokaisella vastaan, niin myös tämän projektin aikana. Valmiiksi saatua grafiikkaa verrattiin muihin töihin, ja töissä nähtiin vain virheitä. Puolivälissä projektia grafiikan uudelleen tekemistä aivan alusta harkittiin ja tyylin ja teeman vaihto eri suuntaan oli päällimmäisenä listassa.

Kriittinen ajattelu omaa työtä kohtaan on suotavaa ja sallittua, mutta jossain kohtaa tulee piste, jossa kannattaa nousta työpisteeltään ja pitää tauko projektistaan. Tämä ei koske pelkästään artisteja vaan kaikkea työtä koodaajasta designeriin. Projektityölle tulee nopeasti sokeaksi ja siihen turtuu liiallisen katselun ja työstämisen vuoksi. Muiden peliprojektien tarkkailu ja Youtubesta esimerkiksi Thomas Brushin pelientekovideoiden seuraaminen [31] sai huomaamaan tämän turhautumisen olevan yksi monen projektin vastoinkäymisistä. Työtä tehdessä edistyy joka päivä ja taidot kehittyvät koko ajan. Puoli vuotta sitten tehtyjä projekteja voisi nykyisillä tietotaidoilla luultavasti toteuttaa paremmin ja tehokkaammin, mutta paikalleen jääminen ylikorjaamisen ja pakonomaisen tarpeen vuoksi tehdä kaikki uudelleen ei projekteja saada koskaan valmiiksi. On hyvä katsoa tehtyä työtä kauempaa ja pyytää ulkopuolisilta kritiikkiä ja apua, mutta muistaa mennä eteenpäin.

### 5.3 Grafiikan käyttömahdollisuudet tulevaisuudessa

Työn tarkoituksena oli luoda grafiikkapaketti, jolle riittäisi käyttöä tulevaisuudessa. Jotta luoduilla tekstuureilla ja spriteilla olisi mahdollisimman paljon käyttöpotentiaalia, niistä tehtiin mahdollisimman monia erilaisia versioita, jotta ne toimisivat pelinrakennuksen kannalta kunkin projektin tarpeen mukaisesti. Tilemaps-pohjaista peliä rakentaville on tarjolla saumattomat tekstuurit. Myös taustan spritejä pystytään hyödyntämään tarvittaessa tämän kuvakulman ja tyylin pelissä. Sprite-kuvien käyttömahdollisuuksia lisää myös Unityyn ladattava SpriteShape-paketti. Sen avulla suorista elementeistä



voidaan taivuttaa halutulla tavalla vaikkapa kuperia tai aaltomaisia muotoja. Tämä lisää yksittäisen spriten käyttömahdollisuuksia runsaasti ja tekee pohjakartasta eloisamman. Kaiken kaikkiaan taustan grafiikkaa voidaan siis käyttää sekä tilemap-pohjaisten että sprite-pohjaisten pelien rakennuksessa.

Hahmosta tehdyn animaatiokollaasin sekä palaset toisistaan erotellun kuvan avulla kukin voi kokeilla haluamiaan animaatiotapoja pelimoottorissa vaivatta. Kollaasista voi luoda valmista liikerataa tuottavaa animaatiota, joka on helppo toteuttaa Unityssä. Hahmosta tehdyn erillisen spriten avulla, jossa raajat ja liikkuvat osat ovat erillään toisistaan, pystytään tekemään animaatioita halutuille liikkeille.

Valmiin grafiikan käyttöä tulee uskaltaa hyödyntää enemmän pelinrakennuksessa. Projektin loppuvaiheessa mietittiin, halutaanko työhön tehtyä grafiikkaa antaa opetuskäyttöön vai pidetäänkö osat julkaistavaa peliä varten itsellä. On kuitenkin parempi, että ne saavat mahdollisuuden tulla käyttöön opetusmateriaalina.

## 6 Yhteenveto

Insinööriyön tarkoituksena oli löytää erilaisia tapoja toteuttaa yleismaailmallisesti toimivaa peligrafiikkaa kaksiulotteisiin peleihin. Työssä selvitettiin syvemmin grafiikka-tekniikoiden eroavaisuuksia sekä tuotantovaiheessa että pelimoottorin puolella työskenneltäessä. Työssä käytiin läpi grafiikan koko tuotantokaari pelkästä ideasta aina valmiiseen tuotteeseen asti. Prosessissa huomattiin, kuinka pelkästään hyvät piirustustaidot eivät yksin riitä tuottamaan hyvää grafiikkaa peleihin, vaan myös teknistä osaamista tulee harjoittaa, jotta vältetään yllätyksiltä kuvien pelimoottoriintuontivaiheessa.

Piirtäminen on monen muun työn ohella prosessi, jonka ajankulkua ei voi etukäteen tarkasti arvioida. On tärkeää luoda mahdollisimman valmis suunnitelma pelistä ennen muiden osa-alueiden aloittamista. Virheitä tehtiin aloittamalla piirtäminen projektin ollessa vielä ajatuspohjalla ja näin hukattiin kallisarvoista aikaa tehtäessä grafiikkaa, jota peliin ei koskaan käytetty. Työstettävien osa-alueiden aikatauluttaminen ja tehtävälisan luonti on erityisen tärkeää; sitä ei turhaan ole opintojen aikana korostettu.

Työssä opittiin hyödyntämään Photoshopissa erilaisia työkaluja grafiikan teossa, jotta lopputuloksen aikaansaanti helpottuu. Aikaisemmin pelkästään taittoon tarkoitettulle grafiikalle luullut ruudukko- ja apuviivatyökalut osoittautuivat suureksi avuksi tarkkaa työskentelyä vaativien elementtien, kuten saumattomien tekstuuriin tuottamisessa. Myös Unitysta löytyi uusia ominaisuuksia ja ladattavia lisäosia, jotka vähentävät työmäärää tulevaisuudessa.

Kun verrataan spriten ja tile-tekstuurin kanssa työskentelyä, huomattiin jälkimmäisen teon olevan paljon teknisempää ja tarkkuutta vaativampaa. Pienet virheet koossa tai esimerkiksi piirtoviivan paksuuden vaihtelussa eivät välttämättä kuvanmuokkausohjelmissa vaikuta suurilta, mutta pelimoottorissa kootessa ne tulevat nopeasti esiin monistuksen ja skaalauksien myötä. Ajankäytön suhteen molemmissa tekniikoissa on omat haasteensa yksittäisiä kuvia vertailtaessa. Kuitenkin sprite-pohjaisten pelien tekoon kuluu enemmän aikaa hyvän sprite-valikoiman piirtämiseen. Ruudukkopohjaisissa peleissä pelkästään muutamalla tekstuurilla voidaan saada jo mielenkiintoisia pelikenttiä aikaan. Molemmissa tekniikoissa on siis sekä hyvät että

huonot puolensa niin ajankäytön kuin teknisen osaamisenkin suhteen. Kannattaa siis valita omaan projektiin parhaiten sopiva grafiikka oman visuaalisen mieltymyksen ja pelin teknisen sopivuuden mukaan.

## Lähteet

- 1 Brown, Stuart. 2014. A Brief History of Graphics. Verkkoaineisto. Youtube. <<https://www.youtube.com/watch?v=dzN2pgL0zeg&list=PLOQZmjD6P2HIOoEVKOPaCFvLnjP865X1f&index=2&t=0s>>. Luettu 21.5.2019.
- 2 Stahl, Ted. 2013. Chronology of the History of Video Games. Verkkoaineisto. <[http://www.thocp.net/software/games/golden\\_age.htm](http://www.thocp.net/software/games/golden_age.htm)>. Luettu 21.5.2019.
- 3 Radulovic, Petrana. 2018. Journey comes to pc for the first time. Verkkoaineisto. Polygon. <<https://www.polygon.com/game-awards-tga/2018/12/6/18130052/journey-pc-trailer-release-tga-2018>>. Luettu 21.5.2019.
- 4 Root, Dan. 2018. How Limbo & Inside Use Tone to Create Space. Verkkoaineisto. Youtube. <<https://www.youtube.com/watch?v=MQkADGiNsMg&t=277s>>. Luettu 21.5.2019.
- 5 Playdead's Limbo. 2019. Verkkoaineisto. Playdead. <<https://playdead.com/games/limbo/#&gid=1&pid=3>>. Luettu 21.5.2017.
- 6 How to make a 2D Platformer - Parallax Scrolling. 2014. Verkkoaineisto. Youtube. <[https://www.youtube.com/watch?v=5E5\\_Fquw7BM](https://www.youtube.com/watch?v=5E5_Fquw7BM)> Luettu 21.5.2019.
- 7 Webster, Andrew. 2017. Cuphead : Creating a game that looks like a 1930s cartoon. Verkkoaineisto. The Verge. <<https://www.theverge.com/2017/9/28/16378364/cuphead-art-design-1930s-animation>>. Luettu 21.5.2019.
- 8 How Babbitt Changed Animation Methodology Part 2 of 4. 2014. Verkkoaineisto. Babbittblog. <<https://babbittblog.com/2014/02/06/how-babbitt-changed-animation-methodology-pt-2-of-4/>>. Luettu 21.5.2019.
- 9 Purdom, Clayton. 2014 Where did Cuphead come from? Verkkoaineisto. Kill Screen Daily. <<https://web.archive.org/web/20140808040730/http://killscreendaily.com/articles/where-did-cuphead-come/>>. Luettu 21.5.2019.
- 10 Diablo Now Available on gog.com. 2019. Verkkoaineisto. Blizzard Entertainments, Inc. <<https://us.diablo3.com/en/blog/22887361/diablo-now-available-on-gogcom-3-7-2019>>. Luettu 21.5.2019

- 11 Murnane, Kevin. 2017. Comparing 'Diablo 3' On The Xbox One X And PS4 Pro. Verkkoaineisto. Forbes. <<https://www.forbes.com/sites/kevinmurnane/2017/11/08/comparing-diablo-3-on-the-xbox-one-x-and-ps4-pro/#9343ca878dfd>>. Luettu 21.5.2019.
- 12 Bellanger, Clint. 2015. Isometric Tiles Math. Verkkoaineisto. Clint Bellanger. <[http://clintbellanger.net/articles/isometric\\_math/](http://clintbellanger.net/articles/isometric_math/)>. Luettu 21.5.2019.
- 13 Bose, Juwal. 2013. Creating Isometric Worlds: A Primer for Game Developers. Verkkoaineisto. Game Development. <<https://gamedevelopment.tutsplus.com/tutorials/creating-isometric-worlds-a-primer-for-game-developers--gamedev-6511>>. Luettu 21.5.2019.
- 14 Doom Wasn't 3D! Digressing and Sidequesting. 2016. Verkkoaineisto. Youtube. <<https://www.youtube.com/watch?v=zb6Eo1D6VW8>>. Luettu 21.5.2019.
- 15 Doom 1 - Mission 1 Gameplay HD. Verkkoaineisto. Youtube. <<https://www.youtube.com/watch?v=MnqLJpgq7jc>>. Luettu 21.5.2019
- 16 How to use Pen Tool in Photoshop? Complete Tutorial. 2014. Verkkoaineisto. Guru99.<<https://www.guru99.com/how-to-use-pen-tool-photoshop-cc-tutorial.html>>. Luettu 21.5.2019.
- 17 Gaille, Brandon. 2016. 8 Pros and Cons of Adobe Photoshop. Verkkoaineisto Brandon Gaille. <<https://brandongaille.com/8-pros-and-cons-of-adobe-photoshop/>>. Luettu 21.5.2019.
- 18 Ruef, Josh. 2014. Gimp Review – A Great Free Alternative to Photoshop. Verkkoaineisto. Kevin Muldoon. <<https://www.kevinmuldoon.com/gimp-review/>>. Luettu 21.5.2019.
- 19 Piskel Review. Verkkoaineisto. Slant. <<https://www.slant.co/options/5480/~piskel-review>>. Luettu 21.5.2019.
- 20 Vector Or Bitmap Images In Your Game, Why Choose? 2013. Verkkoaineisto. Game Builder Studio. <<http://gamebuilderstudio.com/blog/91/Vector-Or-Bitmap-Images-In-Your-Game,-Why-Choose%3F>>. Luettu 21.5.2019.
- 21 Jaworski, Mateusz. 2017. Madness of 2D assets in Unity3D. Verkkoaineisto. Daft Mobile. <<https://blog.daftmobile.com/madness-of-2d-assets-in-unity3d-621792a74967>>. Luettu 21.5.2019.

- 22 Duffy, Sean. 2018. Introduction to the New Unity 2D Tilemap System. Verkkoaineisto. Ray Wenderlich. <<https://www.raywenderlich.com/23-introduction-to-the-new-unity-2d-tilemap-system>>. Luettu 21.5.2019.
- 23 Solarski, Chris. 2013. The Aesthetics of Game Art and Game Design. Verkkoaineisto. Gamasutra. <[http://www.gamasutra.com/view/feature/185676/the\\_aesthetics\\_of\\_game\\_art\\_and\\_.php](http://www.gamasutra.com/view/feature/185676/the_aesthetics_of_game_art_and_.php)>. Luettu 21.5.2019.
- 24 Commander Keen: Marooned on Mars. Verkkoaineisto. Universal Videogame List. <<https://www.uvlist.net/game-166439-Commander+Keen+Marooned+on+Mars>>. Luettu 21.5.2019
- 25 How to make a 2D Platformer – ANIMATION. 2014. Verkkoaineisto. Youtube. <[https://www.youtube.com/watch?v=FRMy5B3dD\\_I](https://www.youtube.com/watch?v=FRMy5B3dD_I)>. Luettu 21.5.2019.
- 26 Thomas, Tony. 2015. How to make a seamless texture in photoshop. Verkkoaineisto. Media Loot. <<https://medialoot.com/blog/how-to-make-a-seamless-texture-in-photoshop-redux/>>. Luettu 21.5.2019
- 27 Brakeys - Homepage for tutorials. Verkkoaineisto. Youtube. <<https://www.youtube.com/user/Brackeys>>. Luettu 21.5.2019.
- 28 Inverse Kinematics: Unity Documentation. 2019. Verkkoaineisto. Unity User Manual. <<https://docs.unity3d.com/Manual/InverseKinematics.html>>. Luettu 21.5.2019.
- 29 Hinton-Jones, Alice. 2018. Intro to 2D world building with sprite shape. Verkkoaineisto. Unity Blogs. <<https://blogs.unity3d.com/2018/09/20/intro-to-2d-world-building-with-sprite-shape/>>. Luettu 21.5.2019.
- 30 Brush, Thomas. 2018. How To Make A Game alone ( 5 secrets ! ). Verkkoaineisto. Youtube. <<https://www.youtube.com/watch?v=ZjDibmEtJmY>>. Luettu 21.5.2019.