

Jussi Seppänen

# 2D-pelin tasosuunnittelu

Metropolia Ammattikorkeakoulu

Insinööri (AMK)

Mediatekniikan koulutusohjelma

Insinöörityö

4.5.2016

Tekijä Otsikko	Jussi Seppänen 2D-pelin tasosuunnittelu
Sivumäärä Aika	47 sivua + 1 liite 4.5.2016
Tutkinto	Insinööri (AMK)
Koulutusohjelma	Mediatekniikka
Suuntautumisvaihtoehto	Digitaalinen media
Ohjaaja	Yliopettaja Harri Airaksinen
<p>Insinööriyön tavoitteena oli tehdä pelaajaa kiinnostava ja viihdyttävä tasosuunnittelu 2D-peliin. Työssä selvitettiin, mitä tasosuunnittelu on, millä tavalla se toteutetaan ja voidaanko teoriaan pohjautuvilla tasopohjamalleilla toteuttaa tasosuunnitelmista toimiva ja kiinnostava käytännön sovellus pelimoottorilla.</p> <p>Työssä tehtiin tasosuunnitelmat viiden tason mittaista pelidemoa varten. Tasosuunnitelmissa tavoiteltiin vaikeustason tasapainon säilymistä tasojen aikana ja tasoilta toisille, pelaajaa kiinnostavien ratkaisujen luomista ja sujuvaa haasteiden sijoittelua tasoille. Tasosuunnitelmien perusteella toteutettiin viiden tason mittainen pelidemo GameMaker-pelimoottorilla. Tasojen luominen havaittiin kohtuullisen helpoksi, ja ainut suurempi havaittu ongelma oli pelidemon toteutuksen pohjana käytetyn pelidemopohjan pelihahmon pääseminen läpi seinäobjekteista. Tämä ratkaistiin kuitenkin nopeasti tekemällä seinäobjekteista suurempia kokonaisuuksia ja asettelemalla ne kulmista päällekkäin.</p> <p>Pelidemoa testasivat 36-vuotias mies ja nainen sekä 10-vuotias poika. Testeissä havaittiin, että tasosuunnitelmien perusteella toteutetut tasot pysyivät tasojen 1, 2 ja 5 kohdalla aika hyvin balanssissa vaikeustason suhteen. Tasojen 3 ja 4 aikana vaikeustaso nousi ja laski huomattavasti, mikä aiheutti ahdistusta ja myös ikävystymistä, eikä flow-tilassa aina pysytty. Yleisesti kuitenkin tasot 1, 2 ja 5 olivat liian helppoja, kun taas tasot 3 ja 4 olivat vaikeustason puolesta lähempänä tavoiteltua flow-tilaa. Testeistä saatujen arvojen mukaan tason 3 tasosuunnitelma oli onnistunein, kun taas tasolla 5 oli suurin hajonta arvojen suhteen. Se ei välttämättä johtunut tasosuunnittelusta, vaan viimeisen tason luonteesta.</p> <p>Tulosten perusteella voitiin päätellä, että luomalla hyvät peli- ja tasosuunnittelun teoriaan perustuvat tasosuunnitelmat voidaan päästä lähelle haluttua lopputulosta, ilman että tasoja on käytännössä edes toteutettu. GameMaker-pelimoottori todettiin hyväksi työvälineeksi ainakin peli- ja tasosuunnittelun alkuvaiheessa oleville, koska sillä voi toteuttaa kohtuullisen helposti yksinkertaisia pelejä ja niille tasoja.</p>	
Avainsanat	2D, tasot, pelimoottorit

Author Title	Jussi Seppänen Designing levels for 2D game
Number of Pages Date	47 pages + 1 appendix 4 May 2016
Degree	Bachelor of Engineering
Degree Programme	Media Technology
Specialisation option	Digital Media
Instructor	Harri Airaksinen, Principal Lecturer
<p>The goal of this study was to make a level design for a 2D game that would be interesting and entertaining for the player. In the study it was analysed what level design is and how to execute it and whether anyone can make a working and interesting app with a game-engine based on theory.</p> <p>In the study level design was made for a five-level long game demo. The goal in the level designs was to balance the difficulty during levels and when moving from one level to another, making solutions that would interest the player and integrating challenges fluently into the level. The demo was made with a game engine called GameMaker, and it was based on the level designs. It was concluded that making level layouts was fairly easy and the only bigger problem was that in the model game demo that was used to make the game demo, the player character could go through walls. This was quickly resolved by making the wall objects larger and placing the edges of the objects on top of each other.</p> <p>The game demo was tested by a 36-year-old man and woman and a 10 year-old-boy. The tests showed that in the levels that were based on the level designs, the difficulty stayed in quite a good balance for levels 1, 2, and 5. In levels 3 and 4 the difficulty dropped and raised fast which caused anxiety and boredom and the player could not stay in a state of flow. Levels 1, 2 and 5 were found too easy and levels 3 and 4 were more enjoyable. Based on the tests, the level design for level 3 was the most successful whereas level 5 was met with most criticism. This was caused by the nature of the last level rather than the level design.</p> <p>The results indicated that making good level designs based on theory makes it easier to achieve the results one wants before actually creating levels in reality. The study also showed that the GameMaker game engine is a good tool for people in the beginning of their game and level design because it is fairly easy to do simple games and levels with it.</p>	
Keywords	2D, levels, game, engine

## Sisällys

### Lyhenteet ja käsitteet

1	Johdanto	1
2	Pelien tasosuunnittelu	1
2.1	Pelaamisen syyt	1
2.2	Pelien peruselementit	3
2.3	Tasosuunnittelu	6
3	Tasosuunnitelmat	18
3.1	Kokonaissuunnitelma	18
3.2	Tasokohtaiset suunnitelmat	20
4	Pelidemon tasojen toteutus	26
4.1	Pelimoottorit yleisesti	26
4.2	Tasojen luominen GameMaker-pelimoottorissa	28
5	Tulokset ja johtopäätökset	34
6	Yhteenveto	40
	Lähteet	42
	Liitteet	
	Liite 1. Pelidemon testaamisen yhteydessä esitetyt kysymykset	

## Lyhenteet ja käsitteet

FPS-peli	FPS-peli on ensimmäisen persoonan ammuntapeli (First Person Shooter), jossa keskitytään vastustajien ampumiseen. Kuvakulman voi kuvitella kameraksi, joka on pelaajan silmien tilalla.
Renderöinti	Renderöinnillä tarkoitetaan sitä, että tietokoneohjelma luo kuvan valmiista tietokonemallista.
Sprite	Spritet ovat kuvapisteistä koostuvaa, osittain läpinäkyvää grafiikkaa. Sprite-grafiikkaan pohjautuvien pelien objektit ovat usein noin 100 – 300 kuvapisteen kokoisia spriteja.
Skripti	Skripti on pieni ohjelmakoodi.
Animoida	Animointi tarkoittaa sitä, että kun yksittäisiä graafisia kuvia näytetään nopeasti peräkkäin, syntyy illuusio liikkuvasta kuvasta.

## 1 Johdanto

Insinööriyön tavoitteena on tehdä pelaajaa kiinnostava ja viihdyttävä tasosuunnittelu 2D-peliin. Pelialustalle ei aseteta ennakkorajoituksia. Suunnitelmissa pyritään löytämään hyväksi todettuja käytäntöjä ja malleja, joiden avulla lopputulos olisi pelaajien mielestä sujuva, kiinnostava ja haastava. Mahdolliset pelaajaa turhauttavat ongelmat pyritään löytämään ja ratkaisemaan etukäteen.

Työssä esitetään teoriaan ja itse keksittyyn lähtötarinaan perustuvat tasosuunnitelmat. Ne sisältävät tasoista luodun haasteiden hierarkiakaavion, tasojen pohjamallit, peliobjektien sijoittelun ja tasojen tasapainottamisen vaikeustason perusteella suhteessa toisiinsa. Tasosuunnitelmien perusteella toteutetaan viisi tasoa sisältävä pelidemo, jonka luomisessa käytetään apuna pelimoottoria. Pelimoottorissa tapahtuvan tasojen rakentamisen tärkeimmät vaiheet käydään läpi yleisellä tasolla. Niissä selvitetään, miten pelimoottorissa voidaan säätää tasojen ominaisuuksia, rakentaa tasopohjia ja lisätä niihin peliobjekteja. Suurinta osaa pelidemon toiminnoista, grafiikoista ja äänistä ei luoda itse, vaan ne on rajattu insinööriyön ulkopuolelle.

Tasosuunnitelmien perusteella luodun pelidemon tekemisen haasteista kerrotaan ja sille suoritetaan testejä, joissa testihenkilöt arvoivat tasojen suunnittelun eri osa-alueita taso kerrallaan. Testitulosten perusteella analysoidaan, mikä tasosuunnitelmissa onnistui ja mihin pitää tehdä parannuksia. Lopuksi arvioidaan, vastasivatko työn tulokset asetettuja tavoitteita ja mikä vaatii vielä kehitystyötä.

## 2 Pelien tasosuunnittelu

### 2.1 Pelaamisen syyt

Ihmisen kyky leikkiä, kuvitella olevansa jotakin muuta kuin hän todellisuudessa on ja taito kokea maailmansa toisenlaiseksi on sosiaalisesti tärkeää. Leikkimiseen liittyy henkinen kyky erottaa kuviteltu todellisuudesta. Leikkiessä voi keksiä uusia asioita tai tapahtumia ja myös hylätä ne milloin haluaa. Leikkimisen ja pelaamisen erottaa toisistaan se, että peleissä on tarkemmat säännöt ja selvemmat tavoitteet. Nämä säännöt, tavoitteet ja kyky irtautua leikkimisen tavoin todellisuudesta, ovatkin tärkeä osa pelien

pelaamista. Tätä on tutkittu hyvin laajasti psykologisena ja kulttuurisena ilmiönä. [1, s. 1.]

Peli viihdyttää joka kerta eri tavalla, kun taas perinteinen media, kuten kirjat ja elokuvat eivät suuresti muutu elämyksinä. Erityisesti yksinpelissä on tärkeää olla haasteita ja ongelmia, joita pelaaja ratkaisee. Pelaaja oppii ratkaistessaan peliin liittyviä ongelmia samalla tavalla kuin ratkaistessaan matemaattisia ongelmia. Yleensä peleissä opittua voi kuitenkin käyttää vain tietynlaisissa peleissä, mutta parhaimmillaan peleissä opittua voi hyödyntää pelien ulkopuolellakin. Usein oppimista ei edes itse havaitse. Ongelmanratkaisun lisäksi peleissä viehättää niihin liittyvä sosiaalisuus. Väite, että pelaaminen on epäsosiaalista, ei useimmiten pidä paikkaansa. Digitaalisten pelien lisäksi suosittuja pelejä ovat esimerkiksi Monopoli, Alias ja Afrikan tähti. Pelien pelaaminen on houkutteleva tapa viettää aikaa perheen ja ystävien kanssa, ja se on hyvä vaihtoehto passiiviselle elokuvissa käynnille tai television katselulle. [2, s. 2–3.]

Verkossa tai yhdessä ystävien kanssa pelaaminen on sosiaalista toimintaa. Pelaamiseen liittyy kuitenkin jonkin verran epäsosiaalisuutta silloin, kun pelaaja haluaa pelata yksin omissa oloissaan. Yksin pelatessaan pelaaja on kuitenkin vuorovaikutuksessa pelien kanssa, jotka simuloivat ihmismäistä toimintaa tekoälyn avulla, joten siinä mielessä pelit eroavat perinteisestä mediasta. Yksinpeleissä on usein tulostaulukko, jonka avulla pelaaja voi seurata omaa kehittymistään. Peleissä kehittyminen voi nostaa myös itsetuntoa, kun pelaaja huomaa, että saa enemmän pisteitä kuin kaverit. Kaverit saattavat arvostaa pelaajan taitavuutta pelissä, vaikka pelaaja ei olisikaan esimerkiksi koulun matematiikkanero. Pelaaja voi tuntea suurta iloa itse ratkaisemalla ongelmia tai arvoituksia ja läpäistessään seuraavan pelitason. [2, s. 5.]

Usein pelaajat hakevat peleistä voimakkaita tunne-elämyksiä, kuten jännitystä ja adrenaliinipurkauksia. Pelaajalle perinteinen media tuottaa usein laimeampia kokemuksia, eivätkä kokemukset tunnu niin henkilökohtaisilta kuin pelatessa. Vaikka esimerkiksi elokuvat vaikuttavat tunteisiin: elokuvissa koetaan jännitystä, iloa, surua ja pelkoa, niin pelit haastavat enemmän. Peli turhauttaa vaikean haasteen edessä, mutta tuottaa suurta tyydytystä, kun haaste selätetään. Pelisuunnittelijoiden olisikin hyvä laajentaa pelien tunneskaalaa perinteisestä jännityssaavutustasosta vähemmän käytettyihin ja tutkittuihin tunnekokemuksiin. [2, s. 6–7.]

Pelit, joissa edetään taso kerrallaan, ovat palkitsevia ja pelaajat odottavat aina innolla, mitä hienoa uusi taso tarjoaa, mutta pelaajat haluavat myös usein itse tutkia pelin maailmaa, ilman että tilanteet tapahtuvat suoraan tietynlaisessa järjestyksessä. Omien strategioiden luominen, resurssien kasvattaminen ja erilaisten pelihahmojen erilaisiin persooniin tutustuminen, voivat olla täysin uusi kokemus verrattuna muiden medioiden tuottamiin kokemuksiin. Peleissä halutaan matkustaa toiseen maailmaan, pois omasta arjesta ja tosimaailman realiteeteista, glamourin keskelle. Siksi fantasia on niin suosittu tarinankerronnan muoto sekä peleissä että myös muissa medioissa. Pelien etuna on se, että pelaaja voi itse ohjata tarinan kulkua ja olla yksi fantasiahahmoista. Pelihahmona voi myös tehdä turvallisessa ympäristössä vaarallisia tai sosiaalisesti ei niin hyväksyttäviä asioita, kuten iskeä jotakuta miekalla ja varastaa sen jälkeen hänen omaisuutensa. [2, s. 6–7.]

## 2.2 Pelien peruselementit

Pelialalla on yleistynyt termi taikaympyrä, joka tarkoittaa kuviteltua todellisuutta, jolla pelaaja korvaa hetkellisesti todellisen maailman. Taikaympyrän idea on se, että siellä voi tapahtua mitä vain, mikä ei olisi todellisessa maailmassa mahdollista. Pelaaja voi esimerkiksi kulkea valonnopeudella paikasta toiseen. Jotkut ovat väittäneet, että taikaympyrä-termin käyttämisestä ei ole todellista hyötyä, mutta koska ihmiset sitoutuvat tunteissaan täysin kuvitteellisiin asioihin, on hyvä olla jokin yleinen termi kuvaamaan tätä ilmiötä. [1, s. 4.]

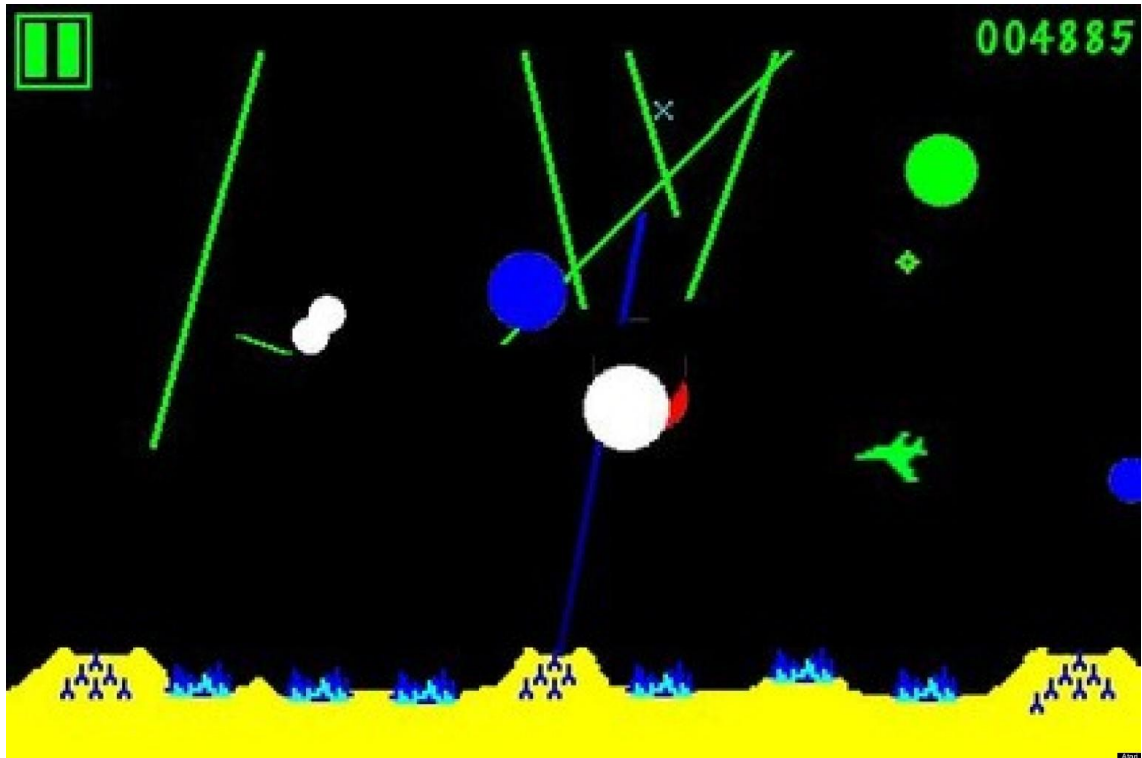
Pelaamisessa pitää olla jokin tavoite tai se on vain yleistä leikkimistä ja sitä ei voi kutsua varsinaisesti peliksi. Usein tämä merkitsee, että peli voidaan joko voittaa tai hävitä. Pelin tavoite määritellään säännöillä, jotka pelinsuunnittelija voi määrittellä oman halunsa mukaan. Tavoite ei saa olla liian helppo, koska jos tavoitteen saavuttaminen ei aseta pelaajalle mitään haastetta, pelaaminen ei tunnu siltä, miltä sen pitäisi tuntua. Peleissä on usein päätavoite, joka voidaan jakaa moneen pienempään osatavoitteeseen. Tavoite voi olla tarkasti määritelty. Koripalloa pelatessa on tavoitteena määräaikaan mennessä saada enemmän pisteitä kuin vastustaja saa. Osatavoitteena voidaan pitää erien voittamista, yksittäisten pisteiden tai jopa seuraavan syötön perille saamista. Usein pelisuunnittelijat haluavat, että pelillä on aina voittaja, joten sitä varten luodaan peliin mekanismi, jolla tasatilanteessa ratkaistaan voittaja. [1, s. 5.]



Luovissa peleissä haasteena voi olla itse luominen, joka vaatii taitoa, jonka kehittäminen voi olla tavoitteena. Pelien tavoitteiden ei tarvitse aina olla saavutettavissa. Tärkeintä on, että pelaaja haluaa yrittää saavuttaa ne. Esimerkiksi kuvissa 1 ja 2 näkyvässä alkuaikojen tietokonepelissä, Missile Commandissa, kuvastettiin kylmän sodan pelkoja sillä, että ennemmin tai myöhemmin ydinsota tuhoaa kaiken. Pelaajan tavoitteena oli jatkaa peliä mahdollisimman pitkään tai saada enemmän pisteitä kuin muut pelaajat ennen välttämätöntä loppua. [1, s. 5.]



Kuva 1. Missile Command -pelikone [3].



Kuva 2. Pelitilannekuva pelistä Missile Command [4].

Verrattuna todellisen elämän vapauteen peleissä joudutaan käytännön syistä usein rajoittamaan sitä, miten pelaaja voi toimia ja liikkua. Esimerkiksi värikuulasodassa voi todellisuudessa juosta, hyppiä, kumartua, kiivetä tai tehdä mitä ikinä keksiikään vihollisalueelle tunkeutumiseksi, kun taas pelissä joudutaan rajaamaan toiminta vain pelimaailman mahdollistamiin tapoihin liikkua. Peleihin kuuluvat säännöt määrittävät nämä pelin rajoitukset ja ohjeet, jotka pelaajan pitää hyväksyä pelin ajaksi. Ne mahdollistavat tavoitteen määrittämisen ja eri tapahtumien merkityksen taikaympyrässä. [1, s. 6–10.]

Säännöt luovat kehyksen, jonka avulla pelaaja voi arvioida, miten olisi hyvä toimia pelin tavoitteen saavuttamiseksi. Säännöt ohjaavat ja pakottavat pelaajan tekemään vaadittavat asiat, mutta joskus on myös hyvä luoda tapahtumia ja toimintaa, jotka eivät ole välttämättömiä tavoitteiden saavuttamiseksi, mutta tekevät pelaamisesta viihdyttävämpää. Sääntöjen reiluus aiheuttaa usein keskustelua pelaajien keskuudessa. Reiluuskäsite toimii taikaympyrän ulkopuolella, ja pelien sääntöjen arvosteleminen on sosiaalisesti hyväksyttyä. Jos säännöt vaikuttavat epäreiluilta, pelaajat päättävät joskus muut-

taa pelien sääntöjä kesken kaiken. Siksi pitäisi olla jokin yleinen yhteisymmärrys siitä, mikä koetaan reiluksi peliksi. [1, s. 6–10.]

Pelimaailman tulee olla looginen. Esimerkiksi tappelupelissä potkujen ja lyöntien pitää toimia aina samalla periaatteella, jotta pelissä voi kehittyä eikä pelaaminen tunnu liian sattumanvaraiselta. Epäonnistunut suoritus ei saa myöskään näyttää samalta kuin onnistunut suoritus. Pelaajan tulisi olla perillä pelimaailman rajoista ja siitä, mitä yleisesti voi ja on tarkoitus tehdä. Vaikka pelaaja ei tiedäkään aluksi kaikkia mahdollisuuksia, hän pystyy silti etenemään pelimaailmassa. Peli ei myöskään muutu epäloogisesti siten, että pelaajan tehtävänä on ollut aluksi ampua kaikki vastaantulijat ja pelaaja yhtäkkiä häviää, koska tehtävänä olikin humoristinen keskustelu vastaantulijoiden kanssa ja samalla satunnaisten arvoitusten ratkaiseminen. Toki vaihtelu voi toimia piristävänä ja yllättävänä elementtinä, mutta yhtäkkiset muutokset saattavat turhauttaa. Usein pelaajat eivät kaipaa jatkuvia muutoksia pelin toimintatapoihin. [2, s. 8–11.]

Kun peliä on pelattu jo jonkin aikaa ja pelaaja on ratkaisut tietynlaisia haasteita useasti, pelaaja olettaa, että myös seuraava haaste ratkeaa tietynlaisella logiikalla. Jos näin ei käy, pelaaja saattaa turhautua, vaikka pelinsuunnittelija olisikin luonut täysin järjestyneen ratkaisun haasteeseen. Tosimaailmassa joutuu välillä etsimään ratkaisua ilman ohjeita ja suuntaviivoja, mutta peleissä pelaaja haluaa ajoittaista ohjausta oikeaan suuntaan. Päämäärätön harhailu ympäriinsä on joskus antoisaa, mutta se ei saa jatkua loputtomiin, vaan pelin pitää riittävän nopeasti ohjata pelaaja taas oikeille urille tai vähintään luoda uusia haasteita. [2, s. 8–11.]

### 2.3 Tasosuunnittelu

#### **Tasosuunnittelijan rooli**

Suurissa tai keskisuurissa tuotannoissa pelillä on erikseen tasosuunnittelija, joka luo pelisuunnittelijan suunnittelemissa komponenteista erilaisia tasoja tai kenttiä, joissa pelaaja seikkailee. Tasosuunnittelija tuo pelimaailman irrallisista ideoista käytännön toteutukseen ja rakentaa ympäristöjä, joissa pelaaja toimii. Pelisuunnittelija siis luo pelimaailman ja sen sisällön, mutta tasosuunnittelija asettelee asiat kohdalleen niin kuin pelaaja tulee ne kohtaamaan. Tasosuunnittelija asettaa tasojen tärkeimmät lähtökohdat ja muuttuvat ominaisuudet, kuten seikkailupeleissä pelaajan kohtaamien vihollisten

määrän, pelaajan resurssit tason alussa ja resurssit, joita tason ympäristöstä voi löytää. Tasosuunnittelija asettaa myös kyseiselle tasolle luodut erilaiset haasteet sopiville paikoille ja järjestykseen. Hän myös sulauttaa tarinankerronnan yhdessä tarinankirjoittajan kanssa oikeille paikoilleen ja luo mahdolliset voitto- tai häviötapaukset. [1, s. 439–440.]

Vaikka pelisuunnittelija yhdessä graafisen suunnittelijan kanssa luo tason yleisen tunnelman ja tietynlaiset esineet ja elementit, yhdistelemällä eri elementtejä tasosuunnittelija toteuttaa sen, miten tunnelma käytännössä luodaan. Esimerkiksi, jos tason tarkoituksena on olla kummitustalo, tasosuunnittelija päättää, minkälainen talo on ja miten siitä luodaan pelottava ja kummitteleva talo. [1, s. 439–440.]

### **Suunnittelukäytännöt**

Tasosuunnittelukäytäntöjä on kahdentyyppisiä: yleiset ja pelityyppikohtaiset. Yleisten käytäntöjen mukaan pitäisi tarjota jo tasojen alkupuolella mahdolliset ohjeistetut osiot, vaihdella tasojen etenemistä ja rytmitystä sekä kohdatun haasteen jälkeen tarjota mahdollisesti menetettyjä resursseja. Jos tasoa ei ole tarkoituksella suunniteltu epätoolliseksi tai hauskaksi, tulisi välttää outoja ratkaisuja, joita tervejärkinen pelaaja ei heti ajattelisi. Pelaajan tulisi myös nopeasti tajuta, mitkä ovat lyhyen aikavälin tavoitteet ja tehtävät kyseisellä tasolla. Vaikka tarkoitus ei ole kertoa pelaajalle tarkasti, mitä pitää tehdä, ei pelaajan pitäisi jäädä loputtomasti arvuuttelemaan seuraavaksi vaadittuja toimia. [1, s. 440–442.]

Pelisuunnittelun historian alkuvaiheissa käytettiin usein kokeile ja opi -menetelmää, jolloin pelaajan tehdessä virheen taso alkoi alusta, mutta seuraavalla kerralla pelaaja tiesi, mihin ei ainakaan kannata mennä tai mitä ei ainakaan pidä tehdä. Nykyään suunnittelijat ajattelevat, että pelaajan olisi hyvä tietää hyödyt ja mahdolliset seuraukset, ennen kuin ottaa riskin pelitilanteessa. Siksi suunnittelijat pyrkivät tarjoamaan ratkaisuja, jotka eivät heti katkaise peliä, mutta silti saattavat aiheuttaa pelaajalle lisähaasteita. Esimerkiksi pelaajan aukaistessa oven saattaa seurauksena olla se, että huoneeseen hyökkää tappajarobotti. Peli ei silti lopu heti oven aukaisemiseen, vaan pelaajalle annetaan mahdollisuus selviytyä tilanteesta hengissä. [1, s. 440–442.]

Koska pelaajat pitävät siitä, että heidän kerrotaan tehneen hyvää työtä, tason pitäisi palkita pelaajaa taidosta, mielikuvituksesta, älykkyydestä ja omistautumisesta. Palkintoina voivat olla vaikka erilaiset resurssit, oikopolut, salaiset tasot, minipelit, poistetut kohtaukset ja muut kerronnalliset materiaalit. Joskus riittää pelkkä yksinkertainen kiittävä lause. Käytännössä pitää siis palkita enemmän kuin rangaista, jotta pelaaja ei lanistuisi ja hylkäisi peliä. Pelaajakeskeisessä suunnittelussa tärkein tavoite on tuottaa pelaajalle nautittava kokemus. Tekoälyvastustajien on tarkoitus antaa hyvä vastus ja sitten hävitä. Pelaajan tulisi kehittyä haasteita kohdatessaan niin, että lopulta hän pysyy pääsemään tasot läpi. Tasot, joita ei voi läpäistä, ovat huonoa pelisuunnittelua. Yksi vaihtoehto on luoda peliin eri vaikeustasoja, joista pelaaja voi valita itsellensä sopivan. [1, s. 440–442.]

Yleisen suunnittelun lisäksi on joitain käytäntöjä, jotka on suunniteltu tietynlaisille pelityypeille. Ammuntapeleissä tulisi palkita tarkkuudesta ja ajoituksesta, eli tarkasti ampuvan tulisi pärjätä paremmin kuin paljon ohi ampuvan. Esimerkiksi väijytystilanteessa myös ajoitus on tärkeää. Paljon nopeatempoista toimintaa sisältävissä peleissä on suositeltavaa vaihdella pelin rytmiä, jotta pelaajalla on aikaa toipua nopeatempoisten tilanteiden välissä niin fyysisesti kuin henkisesti. Strategiapeleissä on muistettava palkita suunnittelu. Esimerkiksi pelaajalla on mahdollisuus luoda puolustus- ja hyökkäyssuunnitelma, joka ottaa huomioon vastustajan liikkeen. Tasosuunnittelijan tulisi luoda hyviä sijainteja, joihin rakentamalla pelaaja saa etua puolustukseen ja hyökkäykseen. Pelaajan tulisi kuitenkin löytää nämä sijainnit itse. [1, s. 443–445.]

Roolipeleissä pitää antaa tilaa pelaajan kasvulle ja itseilmaisulle. Pelihahmon kehittäminen on oleellinen osa tätä pelityyppiä, ja jotkut pitävät sitä jopa tärkeämpänä kuin voittamista. Joka tason pitää tarjota erilaisia mahdollisuuksia kehittää pelihahmon ominaisuuksia ja tilaisuuksia tehdä erilaisia valintoja, jotka pelaaja persoonallisesti valitsee. Urheilupeleissä pelitasoina voidaan pitää eri tapahtumapaikkoja ja pelikenttiä. Usein urheilupeleissä pyritään simuloimaan oikeaa urheilua ja todellisia tapahtumapaikkoja, joten on tärkeää suunnitella tasot mahdollisimman tarkasti vastamaan todellisuutta. Ajosimulaattoreissa ohjauksen suunnittelu ja hyvästä ohjaamisesta palkitseminen ovat olennaisen tärkeitä. Erilaisten räjäytysten ja ympäristön tutkimisten pitäisi tulla vasta sen jälkeen. [1, s. 443–445.]

Rakennus- ja simulaattoripeleissä ei välttämättä tarvita varsinaista tasosuunnittelua. Niihin voidaan kuitenkin luoda erilaisia lähtötilanteita, jotka eivät välttämättä liity toisiin-

sa. Esimerkiksi kuvan 3 SimCity 3000 Unlimited sisältää 13 lähtötilannetta aina Itä- ja Länsi-Berliinin yhdistämisestä jalkapallon maailmanmestaruuskisaottelun järjestämiseen Soulessa. Seikkailupeleissä suunnittelijat jakavat usein pelin jaksoihin, joilla on oma sijaintinsa ja ympäristönsä. Näissä peleissä pelin houkuttavuus tulee paikkojen tutkimisesta ja ongelmanratkaisusta, ja siksi pelin haasteet liittyvät usein ympäristön tarjoamiin näkymiin ja esineisiin. Esimerkiksi maatilalle sijoittuvassa tasossa voisi haaste liittyä sadon korjaamiseen tai maatilalan eläinten hoitamiseen. Ongelmanratkaisupeleissä on tärkeää antaa vihjeitä oikeasta ratkaisusta, koska kokeile- ja erehdy tyyppiset pelit eivät ole yleensä kiinnostavia. [1, s. 443–445.]



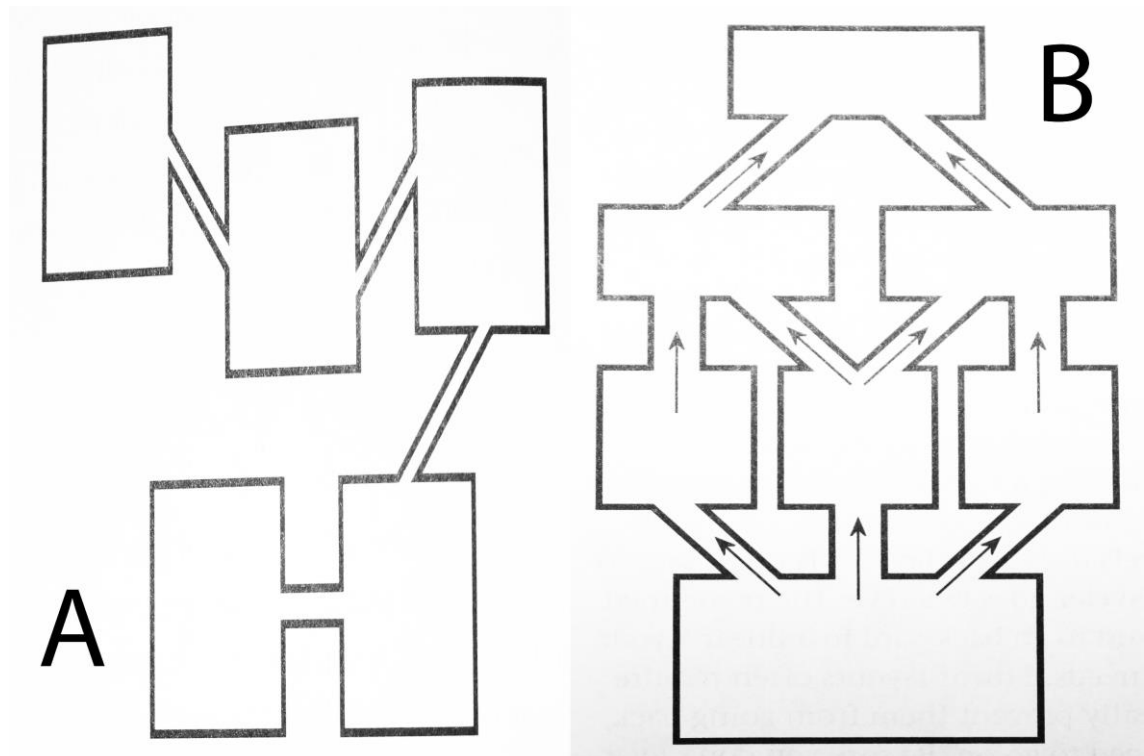
Kuva 3. Pelitilanne pelissä SimCity 3000 Unlimited [5].

## Tasopohjamallit

Vuosien myötä pelien tasojen luomiseksi on kehittynyt yleisiä malleja ja tasopohjaratkaisuja. Ulkoilmassa käytetään usein avoimia tasopohjaratkaisuja, joissa pelaaja voi

lähes rajattomasti liikkua mihin tahansa kyseisessä tasossa tai maailmassa. Niitä käytetään paljon sota- ja roolipeleissä. Myös niissä on usein rajatuilla alueilla käytössä muitakin tasopohjamalleja, esimerkiksi rakennusten sisätiloissa. [1, s. 445–449.]

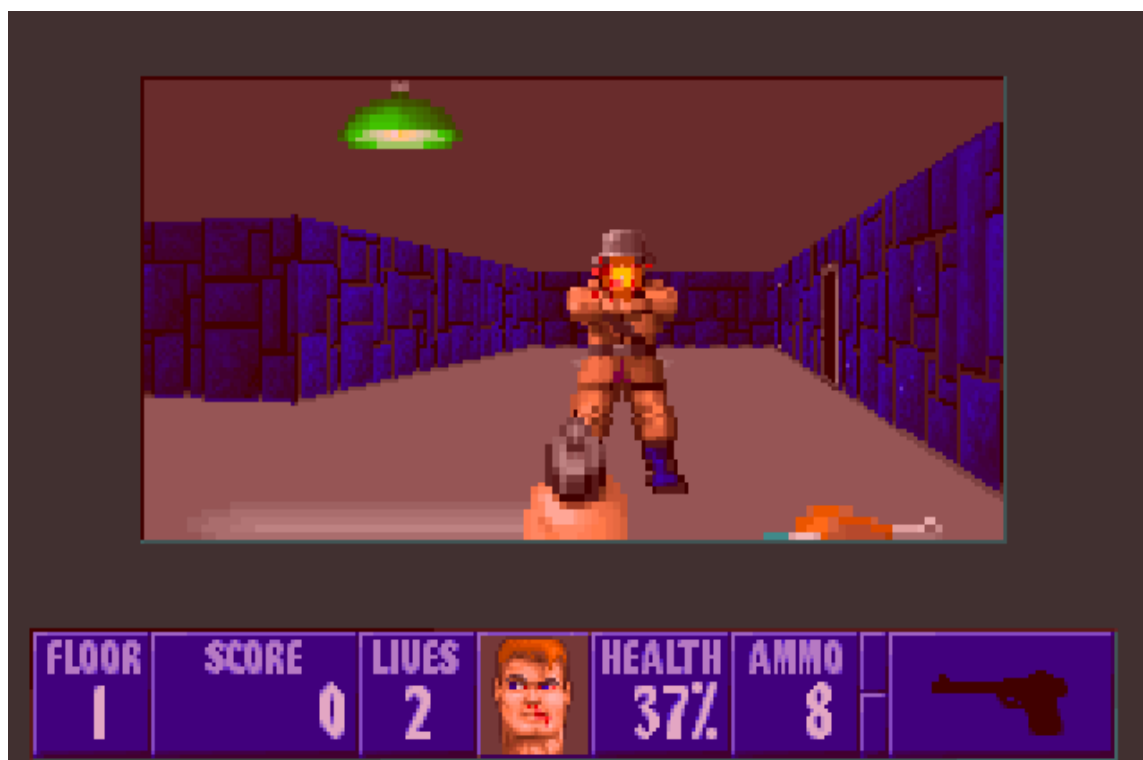
Kuvan 4 A-kohdan tapaisissa suoraviivaisissa tasopohjamalleissa ei yleensä ole vaihtoehtoisia reittejä, mutta se ei tarkoita, että ne ovat välttämättä suoria tasoja. Tällaisten tasojen sanotaan menevän usein kuin raiteilla, ja niissä ei voi siirtyä kuin edelliseen tai seuraavaan tasoon, useimmiten vain seuraavaan. Pelaaja ei siis voi vaikuttaa pelin kulkuun merkittävästi. On tärkeää, että pelaaja pystyy kuitenkin hakemaan mahdollisesti ohittamansa tärkeän esineen, kuten avaimen lukittuun oveen. Suoraviivaisten tasopohjamallien sijaan on nykyään suositumpaa käyttää kuvan 4 B-kohdan tapaista rinnakkaisreittistä tasopohjamallia. Näiden kahden eroa voi kuvata vaikka junaradoilla. Suoraviivaisessa on vain yksi raide, kun taas rinnakkaisessa on monta raidetta ja pelaaja pystyy vaihtamaan rataa määrättyissä paikoissa tarvittaessa. [1, s. 445–449.]



Kuva 4. A-kohdassa suoraviivainen tasopohjamalli ja B-kohdassa rinnakkaisreittinen tasopohjamalli [1, s. 446].



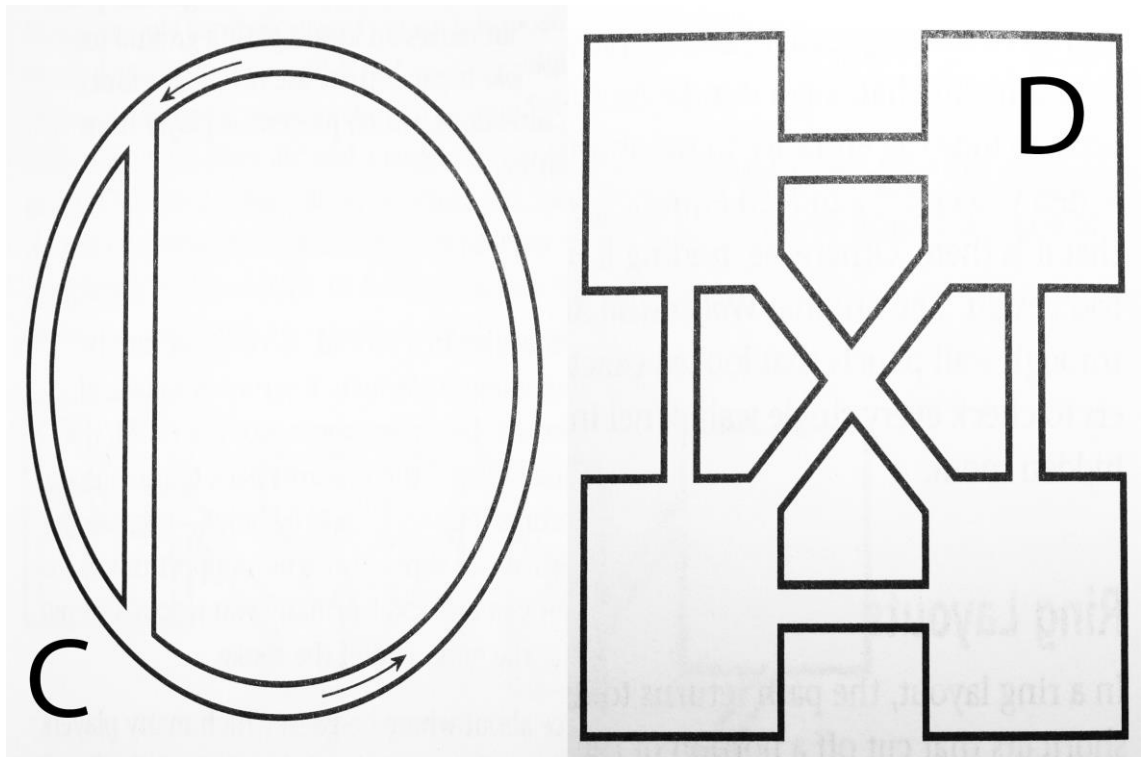
Rinnakkaisreittisissä tasoissa pelaajat usein tutkivat kaikki reitit, vaikka yhden reitin kautta kulkeminen riittäisi. Näissä tasoissa suunnittelija voi myös luoda pelaajalle valinnanmahdollisuuksia riippuen haluaako pelaaja haasteita vai syventyä pelin tarinaan. Lisäksi suunnittelija voi luoda salaisia oikopolkuja, joista olisi kuitenkin hyvä vihjaista pelaajalle, jotta vältettäisiin kokeile- ja erehdy -tyyppinen huono suunnittelu. Kuvan 5 kuuluisassa Wolfenstein 3D -nimisessä pelissä salahuoneisiin pääsee vain tutkimalla seiniä, jotka näyttävät kaikki samanlaisilta. Salahuoneet löytääkseen pitää siis tutkia joka huoneen jokainen seinä. [1, s. 445–449.]



Kuva 5. Pelitilanne pelissä Wolfenstein 3D [6].



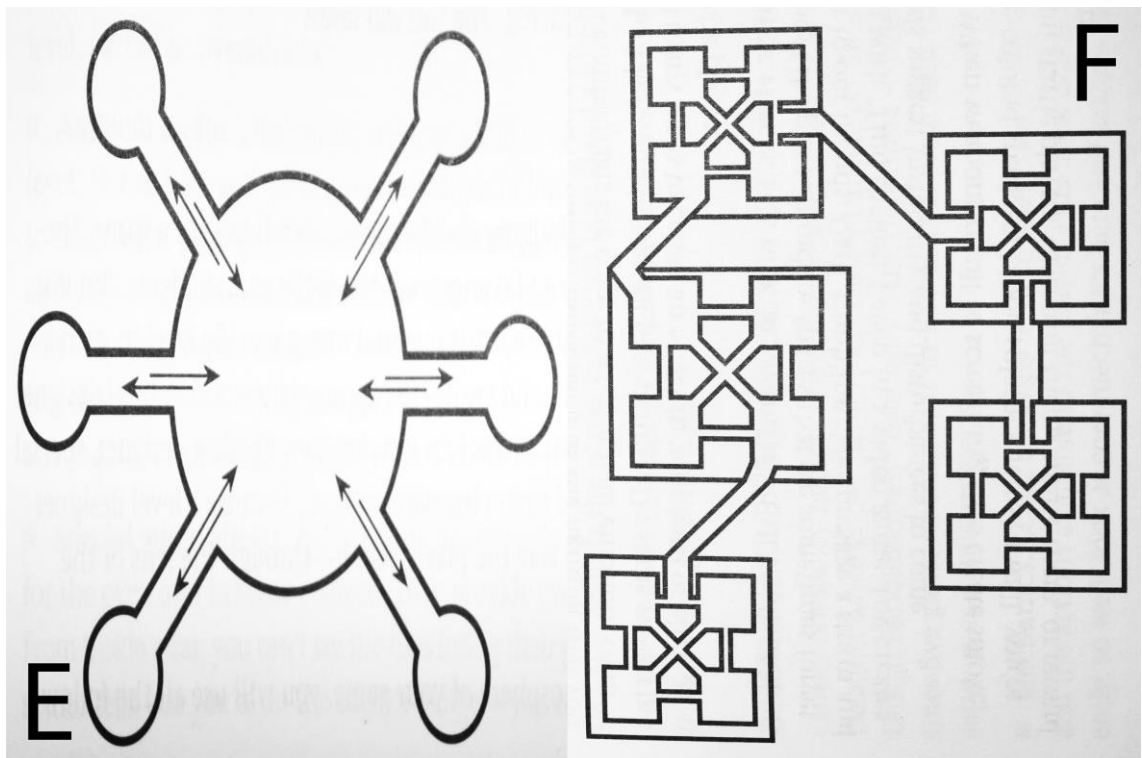
Kuvan 6 C-kohdan tapaisia ympyrämäisiä tasopohjamalleja käytetään usein autopeleissä. Niissä tärkeintä on tasapainottelu varsinaisen reitin ja oikopolkujen haastavuuksien kanssa, jotta kumpikaan ei olisi aina paras reitti pelaajalle. Jos ympyrämäisissä tasoissa on usein selkeät reitit, kuvan 6 D-kohdan verkkomaisissa tasopohjamalleissa ei niitä ole välttämättä ollenkaan. Niissä ei yleensä voi rakentaa tarinaa määrätyn tapahtumaketjun mukaan, vaan tapahtumat pitää voida kokea siinä järjestyksessä kuin pelaaja itse haluaa. Laajoissa verkkomaisissa tasoissa suuri osa peliä onkin löytää itse omat reittinsä peliä pelatessaan. Pienempiä verkkomaisia tasopohjamalleja sen sijaan pidetään ideaaleina pelaajien väliseen taisteluun perustuvissa nopeatempoisissa peleissä. [1, s. 445–449.]



Kuva 6. C-kohdassa ympyrämäinen tasopohjamalli ja D-kohdassa verkkomainen tasopohjamalli [1, s. 447–448].

Kuvan 7 E-kohdassa näkyvän tapaisessa, turvakeskus ja valittavat tasot -tyyppisessä tasopohjamallissa, pelaaja voi itse turvallisessa keskiötilassa valita, mitä tasoja suorittaa ja missä järjestyksessä. Suunnittelussa on kuitenkin hyvä rajoittaa valintaa, jos ta-

sojen välillä on suuria tasoeroja. Hyvässä suunnittelussa tasoja ei kuitenkaan vapauteta yksi kerrallaan järjestyksessä, vaan useampia tasoja kerrallaan, koska muuten peli ei eroa suoraviivaisesta tasopohjamallista mitenkään. Seikkailu- ja roolipeleissä käytetään usein kuvan 7 F-kohdan tapaisia yhdistettyjä tasopohjamalleja, jolloin saadaan vaihtelua peliin ja voidaan luoda vapaasti tutkittavia alueita, joista kuitenkin liikutaan suoraviivaisesti seuraaviin. Suoraviivaisissa peleissäkin, esimerkiksi erilaisissa bonutsoissa, käytetään välillä eri tasopohjamalleja. [1, s. 445–449.]

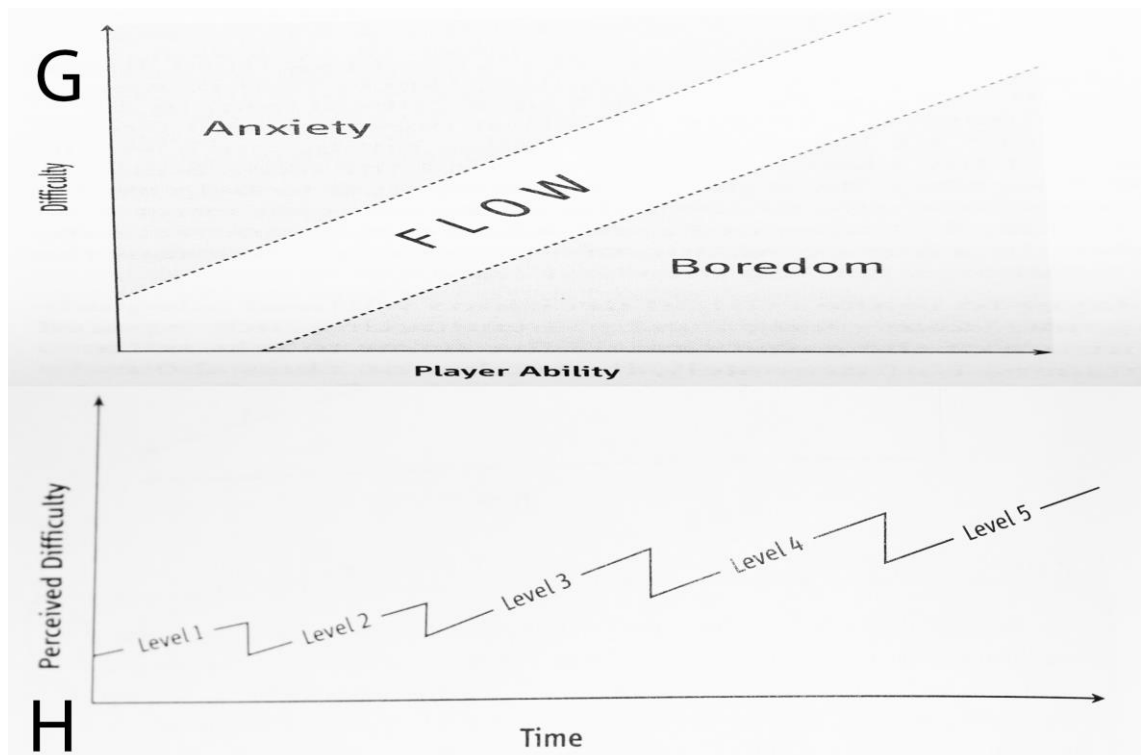


Kuva 7. E-kohdassa turvakeskus ja valittavat tasot -tyyppinen tasopohjamalli ja F-kohdassa yhdistetty tasopohjamalli [1, s. 449].

### Tasojen tasapainottaminen

Psykologi Mihaly Csikszentmihalyi on huomannut, että tehtäviä suorittaessaan ihmisillä on tietty tuottavuuden huipputila, jota hän kutsui flow'ksi. Jos haaste on liian vaikea, he kokevat ahdistusta, ja liian helpon haasteen edessä ihmiset ikävystyvät. Kuvan 8 G-kohdassa vaaka-akselilla ovat pelaajan kyvyt ja pystyakselilla pelin vaikeustaso. Flow

on kuvassa näkyvällä tavalla tila näiden kahden välissä. Tähän tilaan pelisuunnittelijan tavoitteena on tasapainottaa tasoja haastavuus. Siksi pelin edetessä pelaajan tuntemassa vaikeustasossa ei saisi olla suuria vaihteluja. Suunnittelija ei kuitenkaan pysty ottamaan kaikkia asioita huomioon vaikeustasoa määriteltäessä. Pelaaja on ehkä pelannut samankaltaisia pelejä ja ratkaissut samankaltaisia arvoituksia useasti aikaisemminkin, jolloin vaikeustaso saattaa tuntua pelaajasta liian helpolta. Suunnittelija ei voi myöskään tietää pelaajan luontaista lahjakkuutta pelien pelaamisessa. Moninpeli- peleissä vaikeustaso määräytyy yleensä mukana olevien pelaajien taitojen mukaan. [1, s. 418–419.]

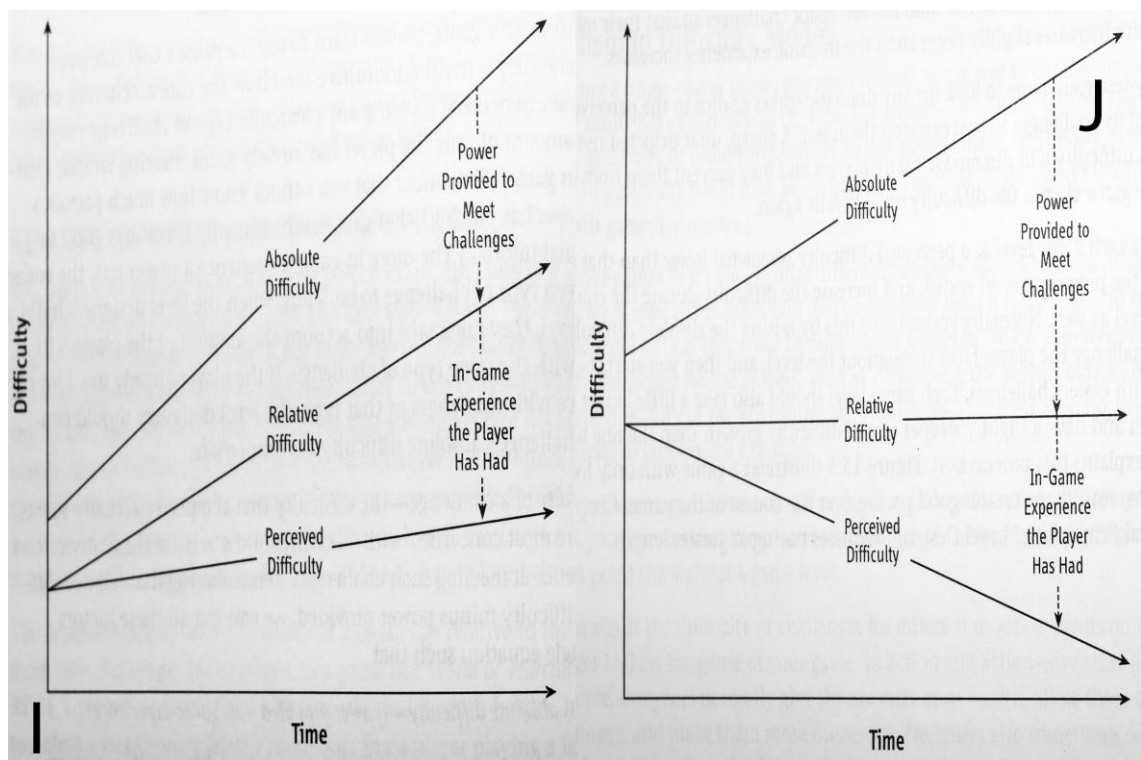


Kuva 8. Flow ja tasojen vaikeusasteen eteneminen tasolta toiselle [1, s.418, 424].

Kuvan 8 H-kohdassa on esitetty pelin vaikeusasteen kasvu tasokohtaisesti. Kuvan vaaka-akselilla kulkee aika, pystyakselilla havaittu vaikeustaso ja sahabiivalle on laitettu tasot järjestyksessä yhdestä viiteen. Yleisenä ohjenuorana voidaan pitää sitä, että tason pelaamisen aikana vaikeustaso nousee joka tasolla tasaisesti saman verran.

Tasojen vaihtuessa vaikeustasoa lasketaan hieman, jotta pelaaja saa pienen hengähdystauon edellisen tason läpäisemisen jälkeen. Uuden tason läpäisemisen pitäisi myös kestää kauemmin kuin edellisen tason. [1, s. 424.]

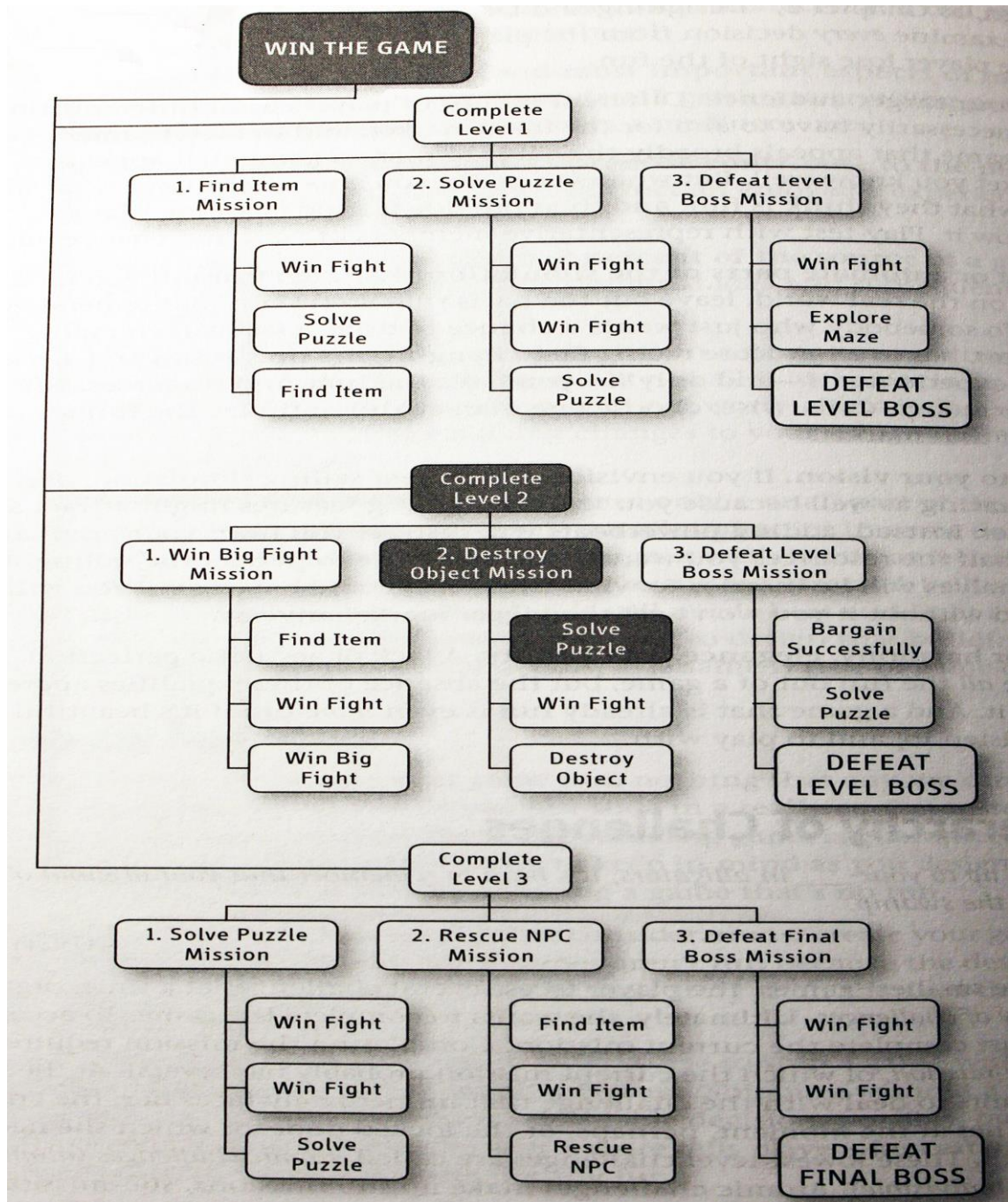
Kuvan 9 I-kohdassa on kuvattu vaikeustason nousu kolmella nuolella suhteessa aikaan. Ylin nuoli kuvaa absoluuttista vaikeustasoa, jota ei pitäisi käyttää pelisuunnittelun pohjana, vaan sen sijaan pitäisi tarjota apuvoimia pelaajalle. Apuvoimien avulla pelaajan havaitsema vaikeustaso laskee kuvan keskimmäisen nuolen tasolle, jota kutsutaan relatiiviseksi vaikeustasoksi. Lopulta pelissä kehitettyjen taitojen avulla vaikeusasteen kasvu hidastuu havaitun vaikeustason tasolle, jota kuvaa alin nuoli. Havaittu vaikeustaso ei saisi koskaan laskea kokoajan tasaisesti alaspäin pelin edetessä loppua kohti (kuvan 9 J-kohdassa). Siksi myöskään relatiivinen vaikeustaso ei saisi pysyä muuttumattomana, vaan sen pitäisi kasvaa edes hieman. [1, s. 422–423.]



Kuva 9. Vaikeustason eteneminen [1, s. 422–423].

## **Tasojen haasteiden hierarkia**

Pelaajan tavoitteena on läpäistä pelaamansa peli. Sitä ennen hänen pitää suorittaa meneillään oleva tehtävä, joka taas voi sisältää alitehtävän, ja silläkin usein on useita alitehtäviä. Kaikkein alin tehtävä on pelaajan juuri sillä hetkellä kohtaama tehtävä, joka sisältyy isompaan kokonaisuuteen ja se taas isompaan, kunnes peli on läpäisty. Kuvassa 10 näkyy pienen seikkailutoimintapelin haasteiden hierarkia. Suunnittelijan on hyvä luoda vastaavanlainen hierarkia pelilleen, koska vaikka pelaaja on juuri taistelemassa vihollisen kanssa, hänellä on mielessä taustalla ajatus laajemmasta tehtävästä. Tämä luo pelille odotusarvoa, joka on pelaajan viihtyvyyden kannalta erittäin tärkeää. Yleisesti pelit neuvovat pelaajaa alkuvaiheen ja loppuvaiheen tasoilla, kun taas keskivaiheiden tasoilla pelaaja saa selvittää tehtäväkuvaa enemmän itse. [1, s. 315–317.]



Kuva 10. Tasojen haasteiden hierarkia [1, s. 316].

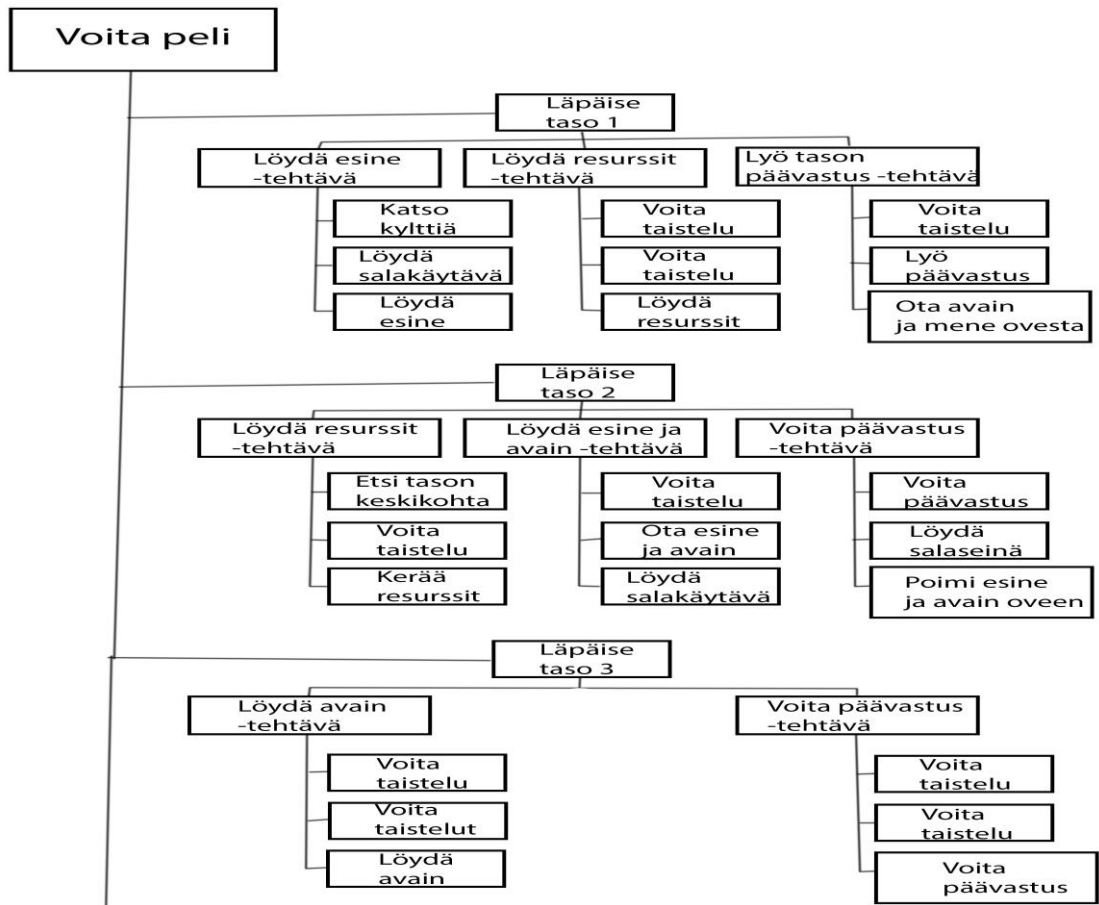
### 3 Tasosuunnitelmat

#### 3.1 Kokonaissuunnitelma

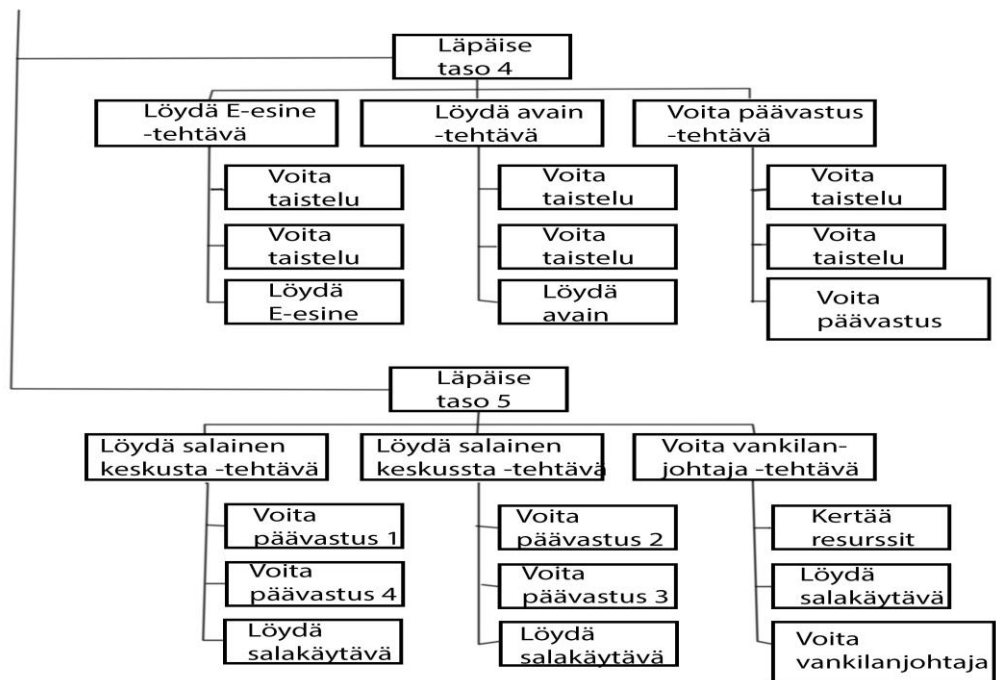
Omien tasosuunnitelmien tekoa varten luotiin pohjatarina, jossa avaruusolennot ovat laskeutuneet maahan ja vanginneet sankarin vankileirille. Sankarin tehtävänä on paeta ja varoittaa maailmaa avaruusolennoista. Koska kyseessä on vankileiri, siellä on myös ovia, joissa on lukkoja, joihin tarvitaan avaimia. Sankarin pitää siis löytää joka tasolta avain, jolla pääsee seuraavalle tasolle ja lopulta pois vankileiriltä. Matkalla sankarin pitää päihittää tai ohittaa kohtaamansa avaruusolennot eli vartijat. Tasoilta voi löytää resursseja, joita saa myös päihittämällä vartijoita. Avaimet ovat usein tasojen päävartijoilla tai niiden läheisyydessä. Päävartijat ovat myös tasojen päävastukset. Viimeisellä tasolla pelaaja kohtaa kuitenkin suurimman haasteensa, vankileirin johtajan.

Pelaajan mielenkiinnon lisäämiseksi tasojen suunnittelussa käytettiin hyväksi havaittuja tasopohjamalleja, joihin sijoitettiin eri sijainteihin mysteerinen E-esine, joka tuo pelaajalle kunniaa, ja avain, jolla pääsee seuraavalle tasolle. Pohjamallien avulla pelikokemus ja tason tutkiminen pyrittiin luomaan toimivaksi kokonaisuudeksi, joka myös luo jännitystä siitä, mitä nurkan takana odottaa. [1, s. 445–449.]

Kuvissa 11 ja 12 on tasosuunnitelman pohjaksi luotu kaavio tasojen haasteiden hierarkiasta. Sitä käytettiin apuna tasosuunnitelmien teossa. Kaavion tekemiseen käytettiin pohjana kuvaa 10. Kaaviossa kuvattujen tehtävien kannalta oleelliset peliobjektit sijoitettiin tasolle tehtävien vaatimiin järjestyksiin. [1, s. 315–317.] Koska kohdatun haasteen jälkeen seuraava haaste ei saisi tuntua pelaajasta kohtuuttomalta, tasoihin sijoitettiin haasteiden sijaintien lähelle resursseja joiden avulla pelihahmo toipuisi mahdollisista menetyksistä [1, s. 440–442.]. Jotta pelaaja ei myöskään turhautuisi etsimällä sala-käytäviä joka puolelta, niiden läheisyyteen sijoitettiin niistä vihjaavat kyltit [1, s. 445–449.]



Kuva 11. Tasojen 1-3 haasteiden hierarkiatasot



Kuva 12. Tasojen 4 ja 5 haasteiden hierarkiatasot



Tasosuunnitelman tärkeimpänä tavoitteena on pitää tasojen aikana pelaajan havaitsema tasojen ja vihollisten vaikeustaso arvojen 3 ja 4 välillä, asteikolla 1–5. Yksi tarkoittaa helppoa ja viisi vaikeaa. Vihollisten ja resurssien määrän tulisi pysyä lähellä arvoa 3, asteikolla 1–5. Yksi tarkoittaa liian vähän ja viisi liikaa. Tasojen laajuuden tulisi pysyä lähellä arvoa 3 asteikolla 1–5. Yksi tarkoittaa liian pieni ja viisi liian suuri. Tasolla asioiden etsimisen tulisi pysyä lähellä arvoa 3 asteikolla 1–5. Yksi tarkoittaa turhauttavaa ja viisi tosi kivaa. [1, s. 418–419, 422–423.]

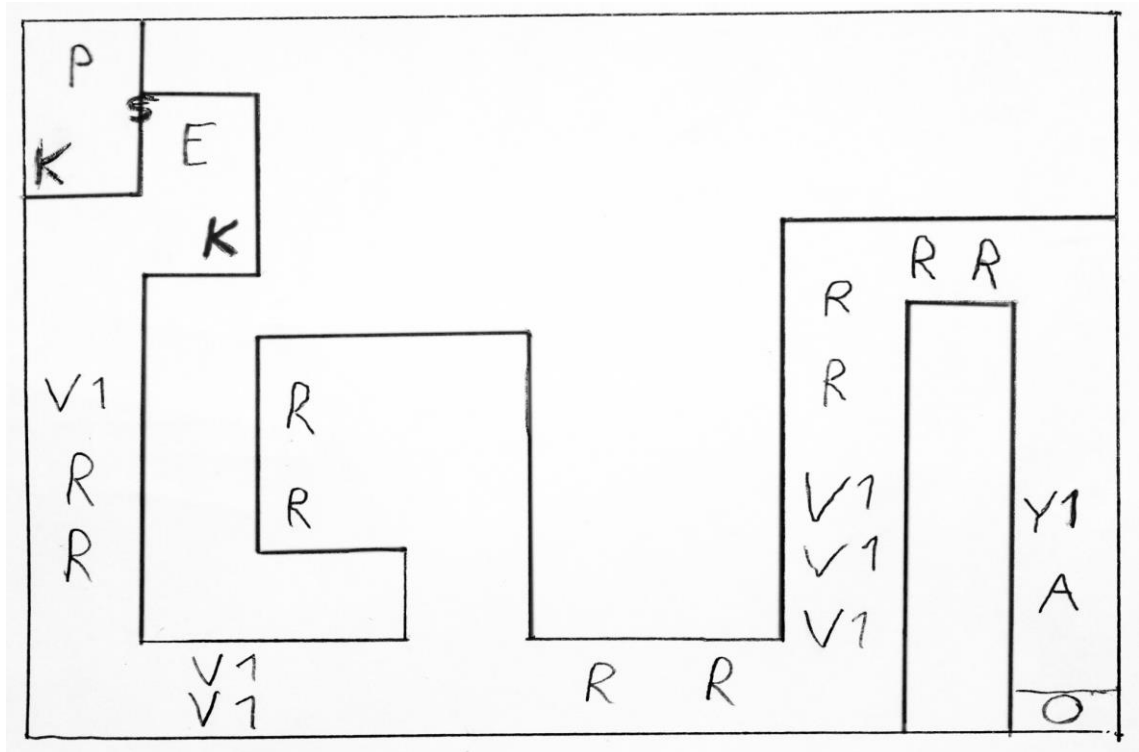
Pelidemoon luodut viisi tasoa on tarkoitus päästä läpi yhdellä pelikerralla, niin että pelaaja on koko ajan flow-tilassa. Jotta pelaajan pelaamisen aikana kehittämät kyvyt eivät laskisi havaittua vaikeustasoa ja pelaaja ei ikävystyisi, suunnitelmissa pelin vaikeustasoa nostettiin asteittain jokaisen tason aikana joko kasvattamalla vihollisten määriä tai niiden vaikeustasoja. Toisaalta vaikeustasoa ei nostettu jyrkästi, jotta pelaaja ei ahdistuisi sen edessä. [1, s. 418–419.] Tasojen lopun ja seuraavan tason alun välillä pitää tapahtua pieni lasku vaikeustasossa, jotta pelaajalla olisi hetki aikaa rauhoittua ennen kuin vaikeustaso nousee taas pelaajan koko keskittymiskyvyn vaatimalle tasolle. Siksi tasojen alkuun valittiin edellisen tason loppua huomattavasti heikommat viholliset, minkä jälkeen alettiin taas kasvattaa vihollisten määriä tai vaikeustasoa kohti tason loppua, jolloin tason lopun vaikeustaso oli korkeampi kuin edellisen tason lopun taso. Näin pyrittiin välttämään tasojen välillä havaittavan kokonaisvaikeustason lasku ja pitämään vaikeustaso vähintään samana tai nostamaan sitä hieman joka tasolla koko ajan. [1, s. 422–423.]

### 3.2 Tasokohtaiset suunnitelmat

Kuvissa 13–17 käytettyjen kirjainten merkitykset ovat seuraavat: avain (A), E-esine (E), kyltti (K), tasojen välinen ovi (O), pelaajahahmo (P), resurssi (R), salakäytävä seinien välissä (S), vartija (V) ja tason päävastus (Y). Kirjaimen perässä oleva numero tarkoittaa hahmon päihittämisen vaikeustasoa.

## Taso 1

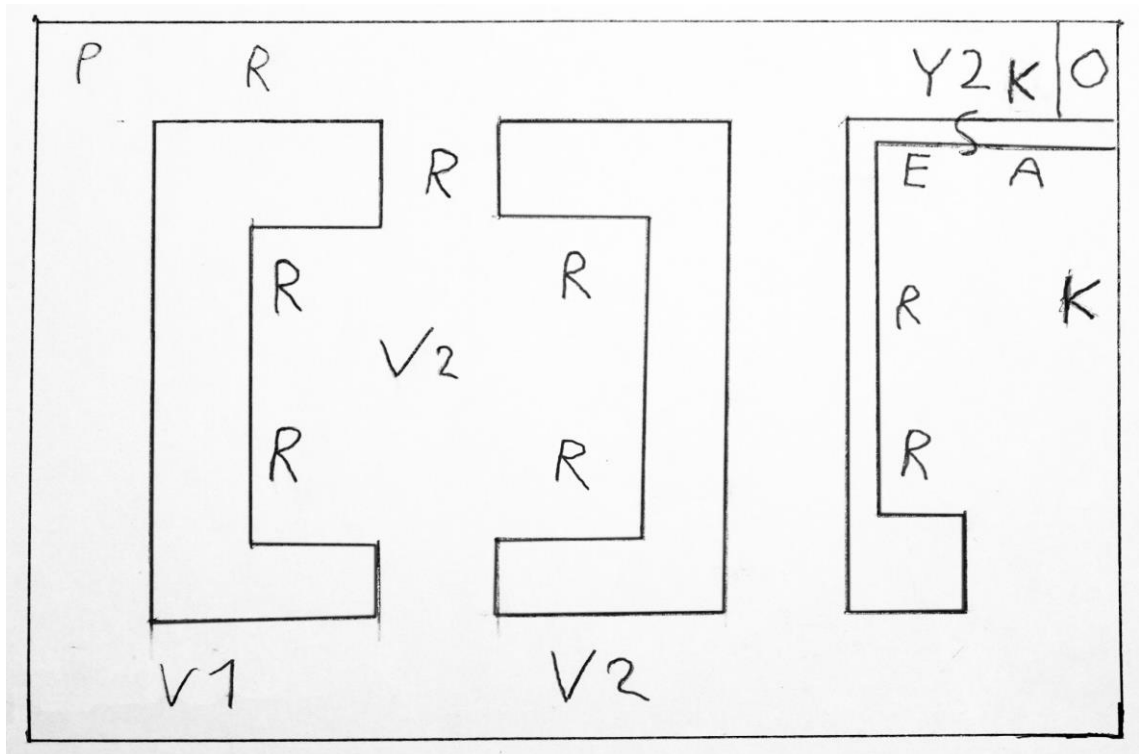
Koska varsinkin pelin alussa pelaaja tarvitsee aikaa ja tilaa pelitoimintoihin tutustumiseen ja yleisiä ohjeita pelin säännöistä ja toiminnoista, tason 1 alkuun luotiin tila, jossa pelaaja voi turvallisesti harjoitella liikkumista ja lyömistä. Tilassa on myös kyltti, joka kertoo tasojen käytännöt, lähellä sijaitsevasta salakäytävästä, E-esineen merkityksen ja mistä avaimia voi saada. [1, s. 440–442.] Tasolle valittiin suoraviivainen tasopohjamalli, jotta pelaaja pystyy keskittymään pelin harjoitusta vaativiin toimintoihin ja käytäntöihin joutumatta liikaa miettimään, mitä tekisi seuraavaksi. [1, s. 445–449.]



Kuva 13. Tason 1 pohjamalli ja peliohjainten sijainnit

## Taso 2

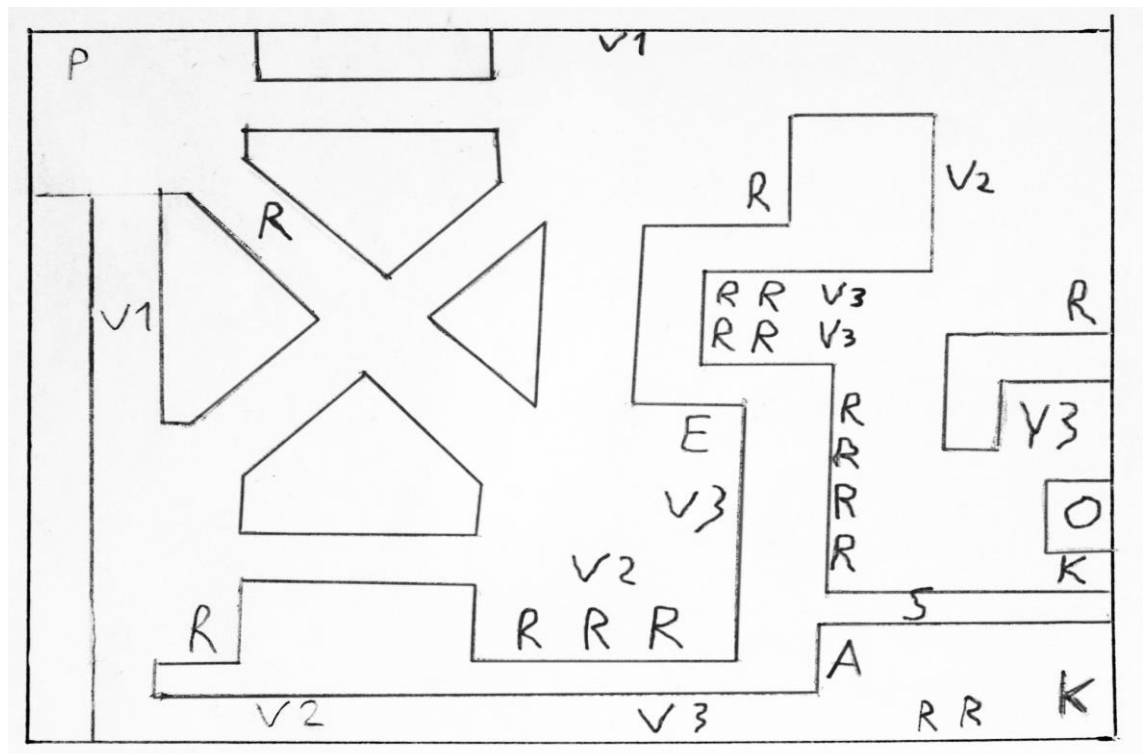
Tason 2 pohjassa on sekoitus verkkomaista, ympyrämäistä ja rinnakkaisreittistä taso-pohjamallia. Tason 1 suorittamisen jälkeen pelaajalla on hetki aikaa rentoutua ja tarvittaessa poimia lähistöllä oleva resurssi. Sen jälkeen pelaajalla on mahdollisuus lukuisiin valintoihin. Jos pelaaja ei pelkää suuria haasteita, hän voi hyökätä suoraan päävastuksen kimppuun. Vaihtoehtoisesti hän voi löytää reitin ilman vastuksia avaimen luo ja salakäytävän kautta suoraan ovesta läpi tai kohdata viholliset silmästä silmään. [1, s. 445–449.]



Kuva 14. Tason 2 pohjamalli ja peliobjektien sijainnit

### Taso 3

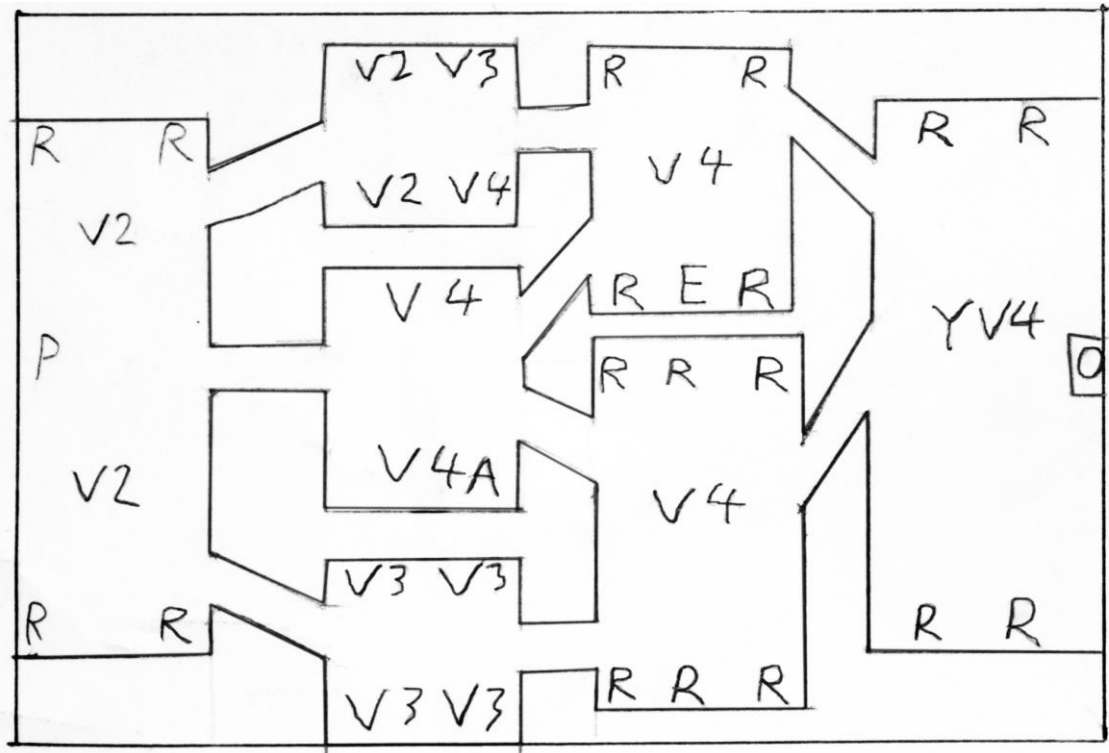
Tasolla 3 on valittavana paljon reittejä tason läpi pääsemiseksi. Reitit on luotu tarkoituksella lähes yhtä vaikeiksi, jotta pelaaja kohtaisi haasteita, varsinkin jos taso 2 on läpäisty hieman onnekkaasti ilman suurempaa haastetta. Tason alku koostuu verkko-pohjaisesta tasomallista. Jonka jälkeen on valittavana kaksi reittiä, joista toinen vaatii kuitenkin salakäytävän löytämisen. [1, s. 445–449.]



Kuva 15. Tason 3 pohjamalli ja peliobjektien sijainnit

#### Taso 4

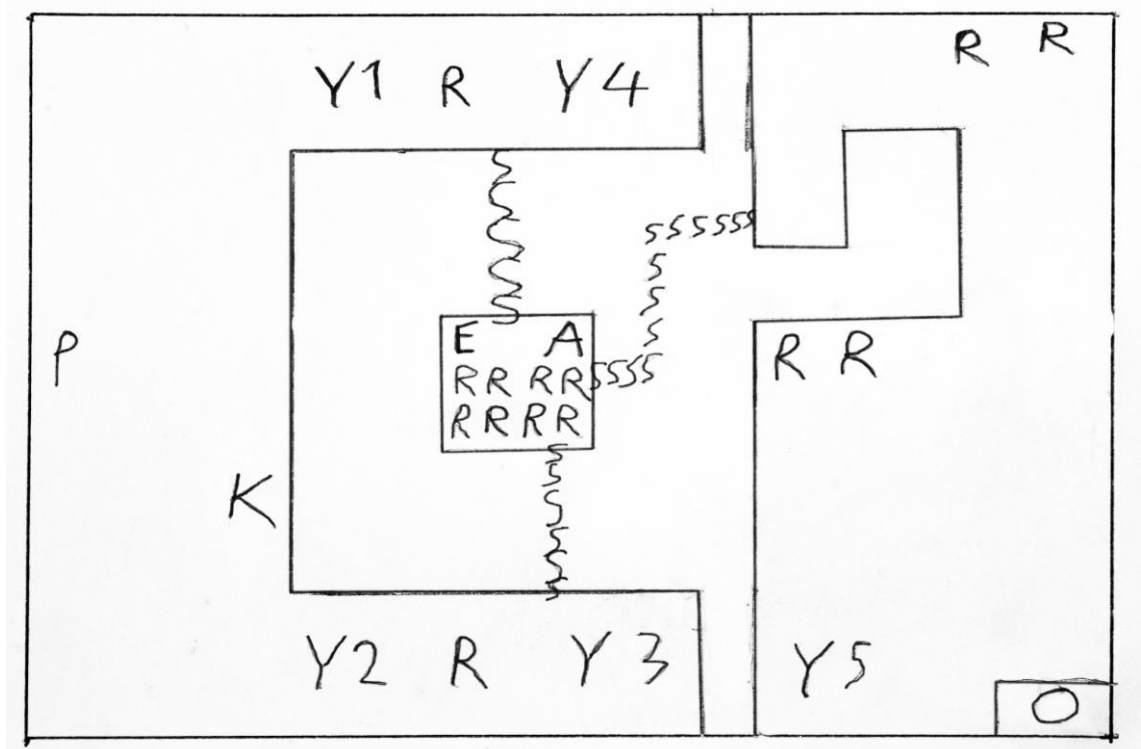
Tasolla 4 on käytetty suoraan pohjana rinnakkaisreittistä tasopohjamallia, koska halutaan testata mallin toimivuutta käytännössä ja mahdollisesti taata pelaajalle onnistunut pelikokemus. Koska lähestytään jo pelidemon loppua ja vihollisia on joka puolella, haasteen on kovennuttava, jotta pelaajan pelaamisen aikana kehittyneet taidot eivät huononna pelikokemusta. [1, s. 445–449.]



Kuva 16. Tason 4 pohjamalli ja peliohjainten sijainnit

## Taso 5

Tasolle 5 haluttiin lopullinen haaste. Se toteutettiin sijoittamalla kaikkien tasojen päävastukset ympärämäistä tasomallipohjaa muistuttavalle alueelle, kuitenkin jättämällä välihaaste, jossa täytyy löytää salakäytävät resurssien, esineiden ja lopulta koko pelidemon päävastustajan luokse. [1, s. 445–449.]



Kuva 17. Tason 5 pohjamalli ja peliobjektien sijainnit

## 4 Pelidemon tasojen toteutus

### 4.1 Pelimoottorit yleisesti

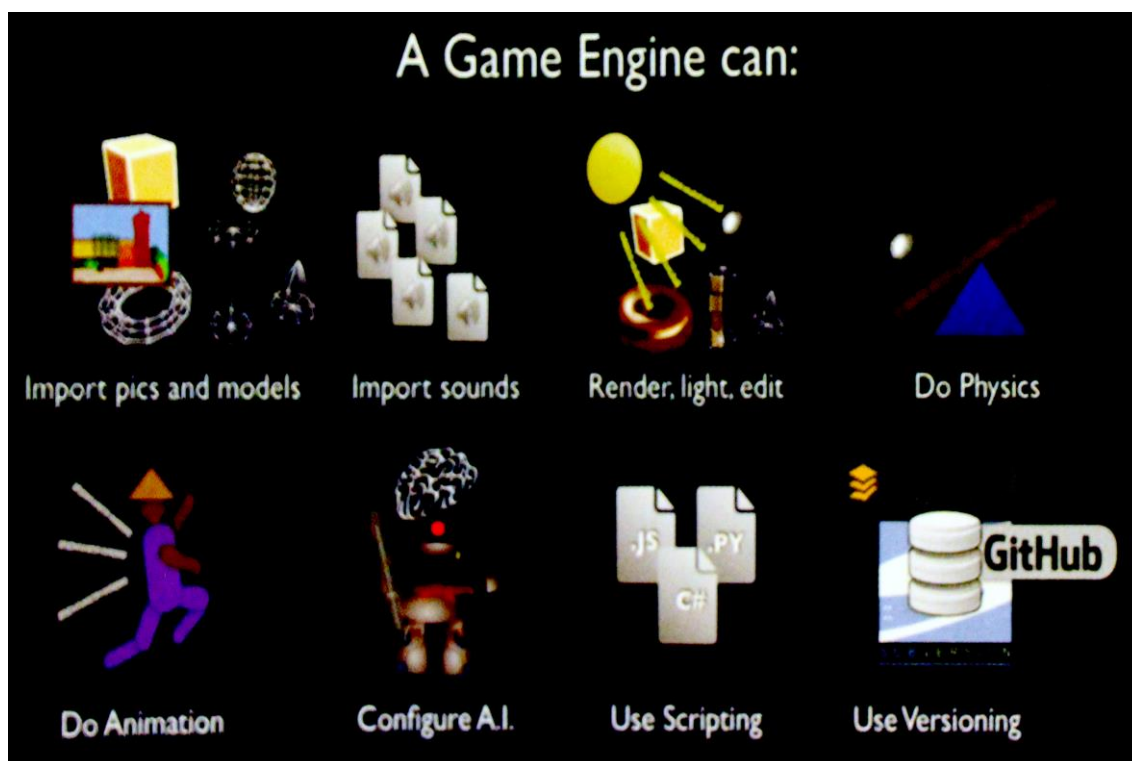
Ennen pelimoottoreiden kehittämistä pelien tekijät joutuivat laitteiden eroavuuksien vuoksi ohjelmoimaan pelit lähes alusta asti uudelleen, jos halusivat julkaista sen toisella pelialustalla. Se oli turhauttavaa ja aikaa vievää, mutta osin pakon sanelemaa, koska laitteistoresurssit olivat hyvin rajoitettuja ja niistä piti saada kaikki irti. Kuitenkin kilpailun kasvaessa laitteistot kehittyivät niin nopeasti, että niistä ei tarvinnut aina ottaa kaikkea irti näyttävien pelien aikaansaamiseksi. Siten pelejä voitiin kehittää samaan aikaan monille alustoille, niin että myös heikommat laitteistot riittivät pelien pyörittämiseen. Ratkaisu löytyi, kun PC-pelimarkkinat kehittyivät ja kasvoivat. PC:n grafiikalle, äänille ja ohjaamiselle piti luoda standardit, jotta ohjelmat toimivat samalla tavalla eri laitteistoissa. [7, s. 149–150.]

Pelisuunnittelussa ensiaskeleen ottivat kolmannen osapuolen kehittäjät, jotka alkoivat luoda grafiikkamoottoreita, jotta riippumatta näytönohjaimesta pelit toimisivat samalla koodilla. Luotiin ohjelmia, joilla pystyi piirtämään useita pelielementtejä tehokkaasti. Standardien luonnin jälkeen John Carmack kehitti kuuluisaan FPS-peli Doomiin pelimoottorin tasosuunnittelua varten ja se johti pelimoottorin lisensointiin muille samankaltaisten pelien kehittäjille. [7, s. 149–150.]

Modernit pelimoottorit ovat kehittyneet eräiksi monimutkaisimmista ohjelmista, joita on suunniteltu. Ne sisältävät pitkälle hiottuja työkaluja keskittyen pienimpiinkin yksityiskohtiin. Valaistuksen, äänien, pelifysiikan ja renderöinnin yhteistyöllä syntyy sujuva pelikokemus. Tämä on johtanut siihen, että usein peliyrityksissä graafisia suunnittelijoita on monikertainen määrä verrattuna varsinaisiin ohjelmoijiin. Pelimoottorit ovat myös helpottaneet ja nopeuttaneet jatko-osien kehittämistä peleille. [7, s. 152–154.]

Kuvassa 22 on ote siitä, mitä tyypillisessä pelimoottorissa voi tehdä. Pelimoottoreissa voi tuoda kuvia ja ääntä ja muokata niistä oman kokonaisuuden. Niissä voi myös tehdä objekteista kaikenkokoisia ja -muotoisia ja valita niihin kohdistuvan valon tulokulman ja voimakkuuden. Peliobjektien fyysiset ominaisuudet voidaan määrittellä tyypillisesti kolmen asian suhteen: painovoiman, kääntymisen ja törmäyksen tunnistamisen. Peliobjektit voidaan animoida, ja niille voidaan luoda tekoäly. Peliobjekteille voidaan myös

kirjoittaa skriptejä, joilla on mahdollista vaikuttaa pelitapahtumiin. Esimerkiksi pelihahmon mennessä ennalta määriteltyyn paikkaan voidaan skriptillä tarkistaa, onko sillä ennalta määriteltyä esinettä, jotta pelihahmo voidaan päästää etenemään. [7, s. 152–154.]



Kuva 18. Pelimoottorin mahdollistamat asiat [7, s. 153].

Laitteiston turhan rasittamisen välttämiseksi pelimoottorissa on tärkeää asioiden uudelleenkäyttö, minkä vuoksi ei esimerkiksi tarvitse ladata samaa animaatiota tai ääntä moneen kertaan. Siksi jokaista pelin elementtiä tulee seurata tarkasti, ja ne on pidettävä hyvässä järjestyksessä omissa kategorioissaan. Vaikka pelimoottorit riittävät pitkälle pelikehityksessä, joskus tarvitaan tietoturvasyistä erillinen ohjelma pelien eri kehitysversioiden hallintaan. Taulukossa 1 on listattu viisi suosituinta pelimoottoria Framebench-blogin mukaan. [7, s. 152–154.]



Taulukko 1. Viisi parasta pelimoottoria Framebench-blogin mukaan [8].

Sija	Nimi	Verkko-osoite
1	Unreal Engine 4	www.unrealengine.com
2	Unity 5	www.unity3d.com
3	Source 2	developer.valvesoftware.com
4	CryEngine 3	www.cryengine.com
5	Frostbite 3	www.frostbite.com

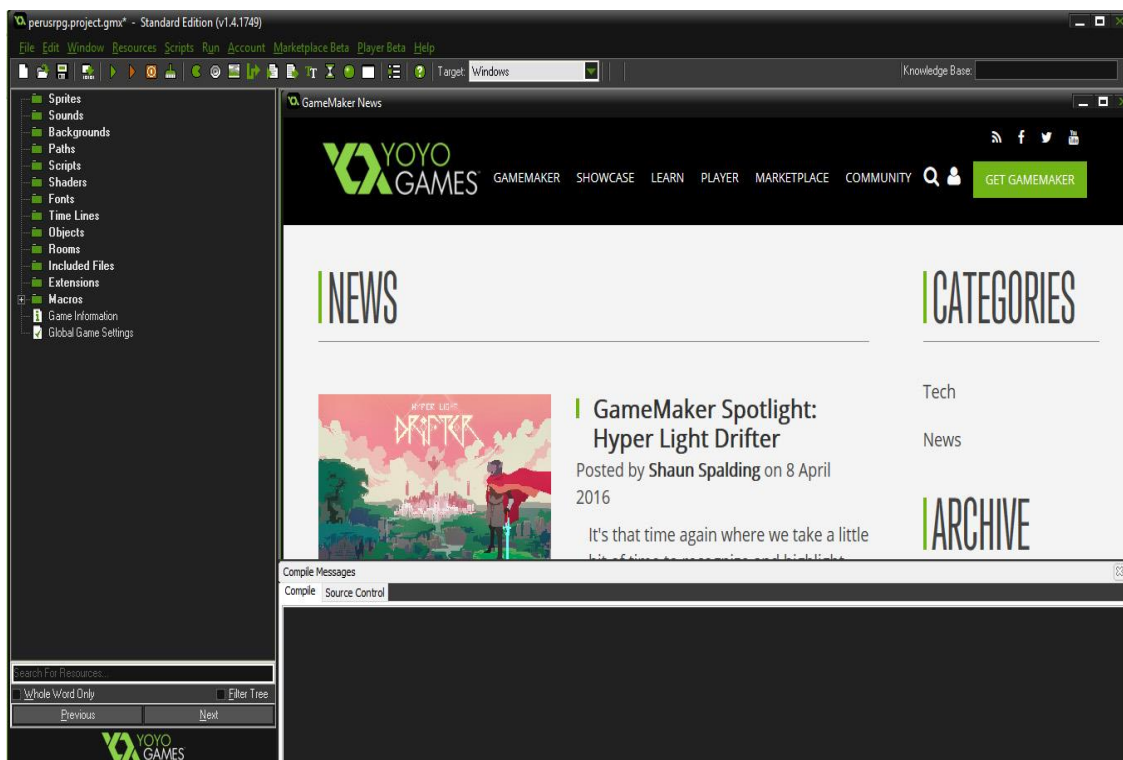
#### 4.2 Tasojen luominen GameMaker-pelimoottorissa

Benjamin Anderson on GameMaker-pelimoottorin asiantuntija, joka on kymmenien opetusvideoiden lisäksi kirjoittanut GameMakeristä kirjan nimeltä GameMaker Language – An In-Depth Guide. Aloittelijoille tarkoitettussa opetusvideosarjassaan, [Beginner Tutorial] Make an RPG in GameMaker, hän kertoo GameMaker: Studioversiota käyttäen, miten luodaan roolipeli GameMaker-pelimoottorissa. Tässä insinööriyössä luotiin tasosuunnitelmien perusteella uudet pelitasot Andersonin opetusvideoissa esittämään roolipelipohjaan. [9.]

Mark Overmarsin luoman GameMakerin omistaa ja sitä ylläpitää YoYo Games. GameMaker on kaksiulotteisten pelien kehitysympäristö, jolla voi myös tehdä jonkin verran kolmiulotteista pelikehitystä, mutta se ei ole yleistä. Suosittuja GameMakerillä kehitettyjä pelejä ovat muun muassa kuvassa 19 näkyvä Hyper light Drifter -peli, joka keräsi yli 650 000 dollaria hankerahoituksen keräystä varten tarkoitettulla Kick Starter-sivustolla, Spelunky, jonka alkuperäinen versio kehitettiin GameMakerillä, ja Death's Gambit, josta Andersson oli eniten innostunut videon tekohetkellä. [10.]

Kuvassa 19 vasemmalla näkyy resurssipuu, joka on kansio rakennepelissä käytettävissä olevista resursseista, joita ovat kuvat, äänet, taustat, polut, skriptit, varjostimet, fontit, aikajanat, objektit, huoneet eli tasot, lisätyt tiedostot, lisäosat ja makrot. Kuvassa 19 ylhäällä on kuvakkeita, joita klikkaamalla suoritetaan seuraavat toiminnot: uusi peli, avaa peli, tallenna peli, rakenna ajettava ohjelma pelistä, aja peli, aja peli virheentarkis-

tustilassa, pysäytä verkkopalvelin, tyhjennä resurssien välimuisti, luo sprite, luo ääni, luo tausta, luo polku, luo skripti, luo varjostin, luo fontti, luo aikajana, luo objekti, luo taso, muuta pelin asetuksia ja näytä ohjeet. [10.]

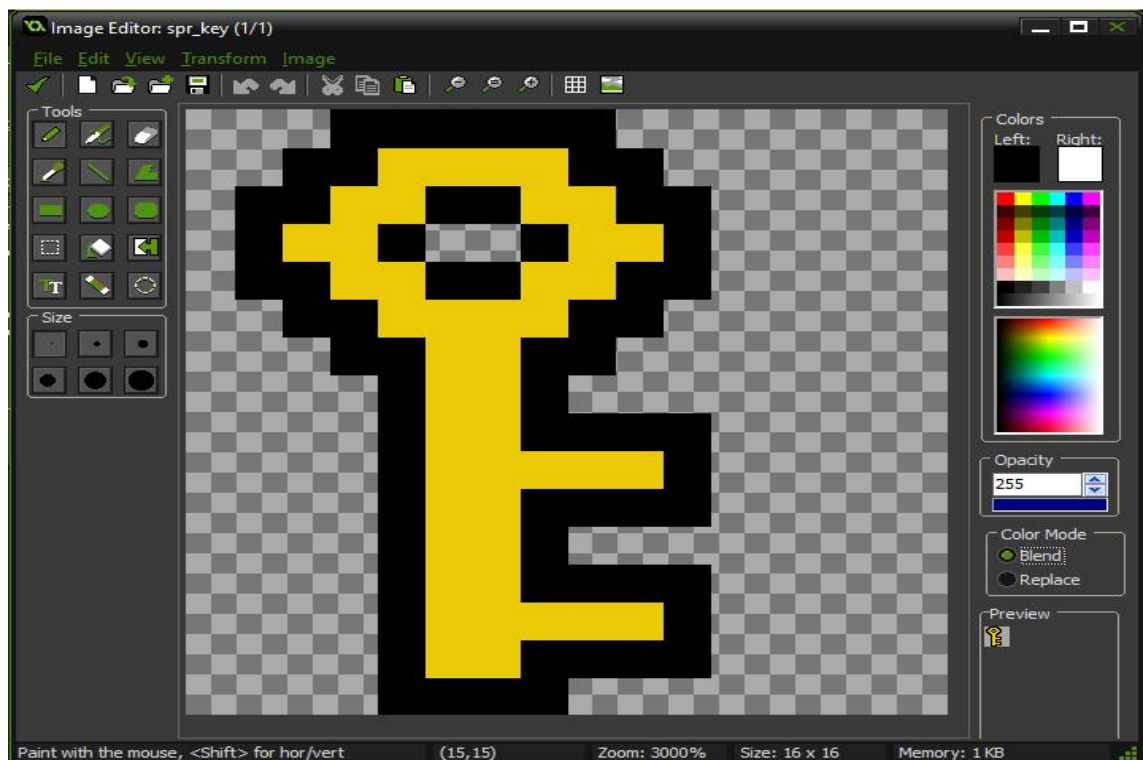


Kuva 19. GameMakerin aloitusnäky

## Objektien luominen

Tason seinien sprite ja minkä tahansa muun elementin sprite voidaan luoda klikkaamalla spriten luonti -kuvaketta, ja sen jälkeen avautuvissa valikoissa klikkaamalla seuraavassa järjestyksessä: edit sprite (muokkaa spritea), create a new sprite (luo uusi sprite) ja image 0 (kuva 0). Tämän jälkeen avautuu kuvassa 20 näkyvä kuvanmuokausikkuna, jossa voi piirtää ja maalata kuten missä tahansa perinteisessä piirto-ohjelmassa. Shakkilaudannäköisen alueen saa suuremmaksi tai pienemmäksi rullamalla hiiren keskinäppäintä, kun hiiren osoitin on alueen päällä. Kun on valmista, klikataan ok -kuvaketta valikoissa, kunnes palataan alkutilanteeseen. Spriten luontivalikois-

sa on myös muita toimintoja, kuten spriten nimeäminen, koon säätäminen ja lataaminen tiedostosta. Luotu sprite voidaan lisätä objektiin seuraavalla tavalla: klikataan kuvaketta luo objekti, avautuvassa valikossa sprite-kohdassa valitaan seinää varten luotu sprite, ja sen jälkeen klikataan ok-ikonia. Tämän jälkeen on luotu objekti, johon on liitetty itse luotu sprite, ja sitä voidaan käyttää tasojen seinien rakentamiseen eli tasopohjien luomiseen. [10.]

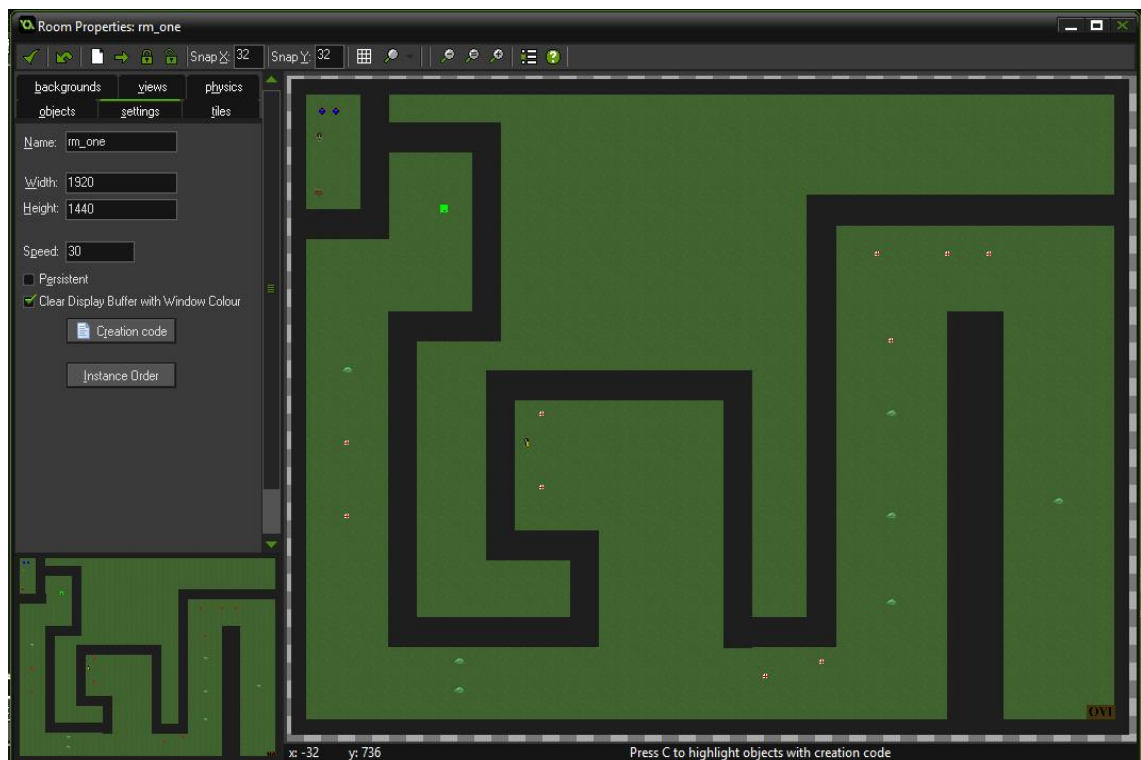


Kuva 20. Itse luodun avaimen spriten kuvanmuokkausikkuna

## Tasojen luominen

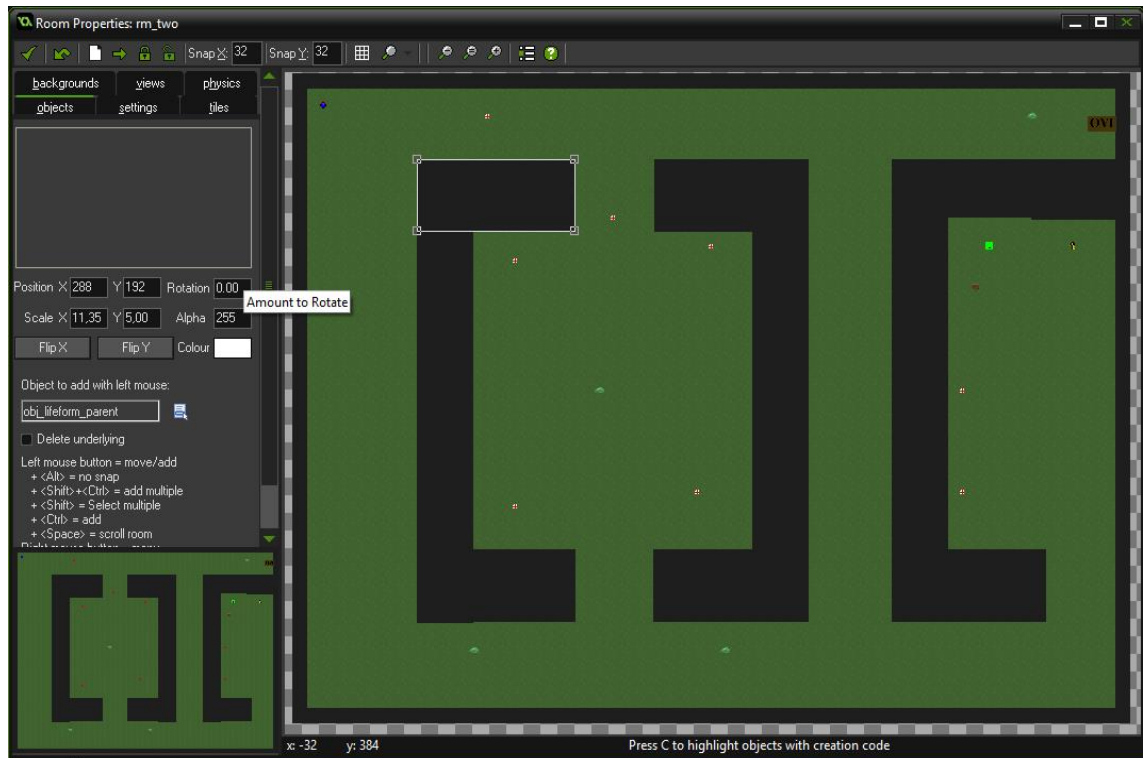
Taso luodaan klikkaamalla luo tasokuvaketta, jolloin avautuu kuvassa 21 näkyvä tason ominaisuuksien määrittämiseen tarkoitettu ikkuna, jossa on näkymä tasosta ja valikko, jossa ovat seuraavat välilehdet: taustat (backgrounds), näkymät (views), fysiikat (physics), objektit (objects), asetukset (settings) ja tiilet (tiles). Näistä uuden tason luodessa avautuu asetukset-välilehti. Siinä voi muun muassa määrittää tason nimen, mittasuht-

teet, ruudun päivitysnopeuden ja ominaisuuksien siirtymisen seuraaviin tasoihin ja avata koodieditorin tason luonnin yhteydessä suoritettavan koodin editoimiseen. Näkyvää alaa tasosta voi säätää rullaamalla hiiren keskinapista ja sen sijaintia pitämällä keskinnappia pohjassa ja raahaamalla. Tasolle voi lisätä objekteja valitsemalla objektivälilehden ja klikkaamalla neliötä, jossa näkyy käytössä oleva objekti, jolloin avautuu valikko, jossa näkyy kaikki käytettävissä olevat objektit. Sen jälkeen valitaan haluttu objekti ja klikataan tasonäkymässä kohtaa johon se halutaan sijoittaa. [10.]



Kuva 21. Tason 1 ominaisuuksien määrittämiseen tarkoitetun ikkunan asetukset-välilehti

Kuvassa 22 on tasosuunnitelmien perusteella rakennettu taso 2. Objekteja voi siirtää raahaamalla ja poistaa pitämällä ctrl-näppäintä pohjassa ja samalla painamalla hiiren oikeaa näppäintä. Valituilla objekteilla on myös venytysneliöt nurkissaan, joista vetämällä objektia saa venytettyä haluttuun suuntaan, ja tällä tavalla voi piirtää vaikka seinän. Arvot voi myös syöttää x- ja y-akselin suhteen suoraan valikossa. [10.]

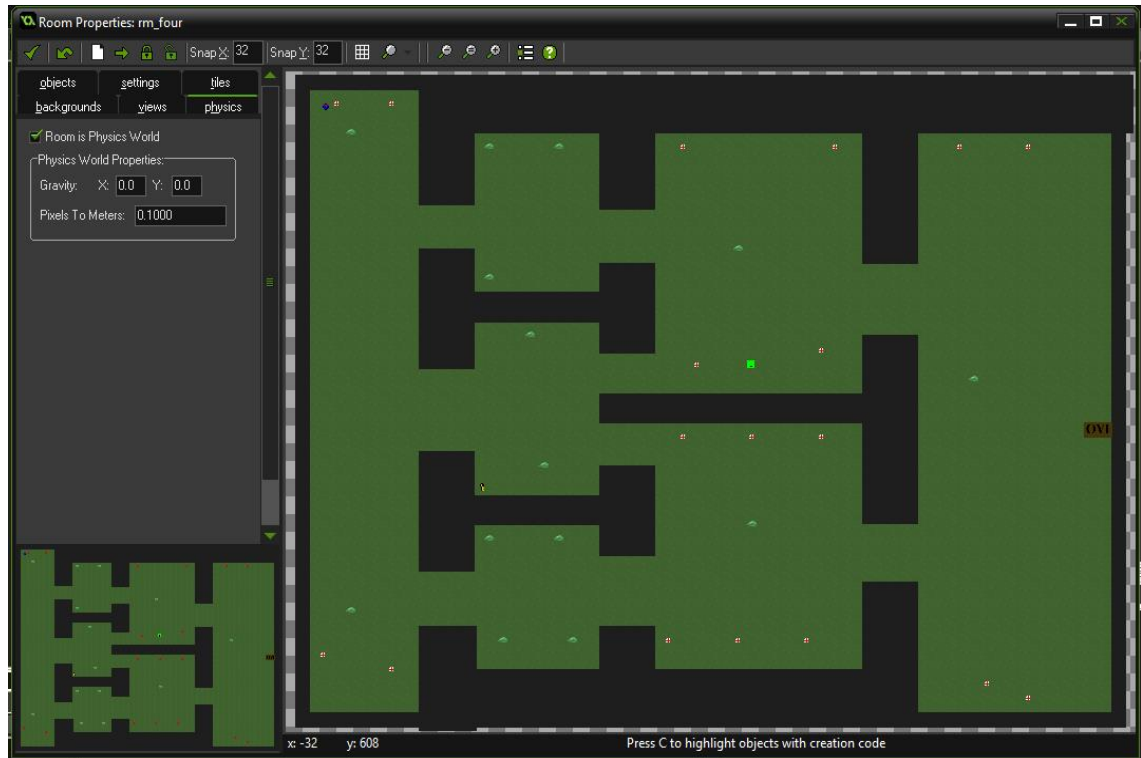


Kuva 22. Tason 2 ominaisuuksien määrittämiseen tarkoitetun ikkunan objektit-välilehti

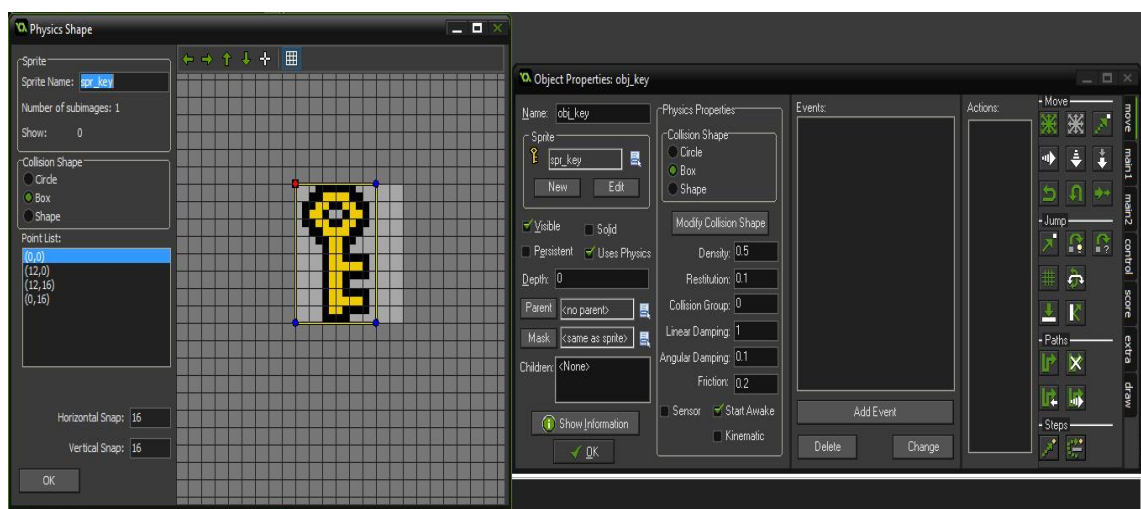
## Tasojen kontrollointi

Jotta tason objekteilla olisi fyysikaalisia ominaisuuksia, esimerkiksi, niin että pelaaja ei voi mennä seinien läpi, ne pitää kytkeä päälle merkitsemällä rasti kuvassa 23 näkyvällä fysiikat-välilehdellä kohtaan: taso on fyysinen maailma (room is physics world). Samalta välilehdeltä voi myös säätää painovoiman kohdassa: painovoima (gravity), jolle hyvä aloitusarvo on 0.0 x- ja y-akselilla. Myös peliobjekteissa pitää asettaa fyysiset ominaisuudet päälle. Se tehdään klikkaamalla resurssipuussa objektia ja merkitsemällä rasti avautuvassa valikossa kohtaan: käyttää fyysisiä ominaisuuksia (uses physics). Sen jälkeen aukeavissa kuvan 24 oikealla puolella näkyvissä lisäsäädöissä voi määrittää objektille sen törmäysalueen muodon ja vielä hienosäätää sen täsmälleen klikkaamalla painiketta: muokkaa törmäysmuoto (modify collision shape), jolloin aukeaa kuvan 24 vasemmalla puolella näkyvä ikkuna. Jos ei haluta, että seinät liikkuvat, kun pelaaja törmää niihin, seinäobjektien tiheys (density) eli massa kannattaa asettaa arvoon 0.0, jolloin objekteista tulee äärettömän painoisia, jolloin muut objektit, kuten pelaajan objekti, ei voi niitä liikuttaa. Tämän lisäksi pitää lisätä törmäystapahtuma objektille klik-

kaamalla nappia: lisää tapahtuma (add event)” ja aukeavasta valikosta valita kohta: törmäys (collision) ja objekti, jonka halutaan törmäävän säädettävänä olevan objektin kanssa. [10.]



Kuva 23. Tason 4 ominaisuuksien määrittämiseen tarkoitetun ikkunan fysiikka-välilehti

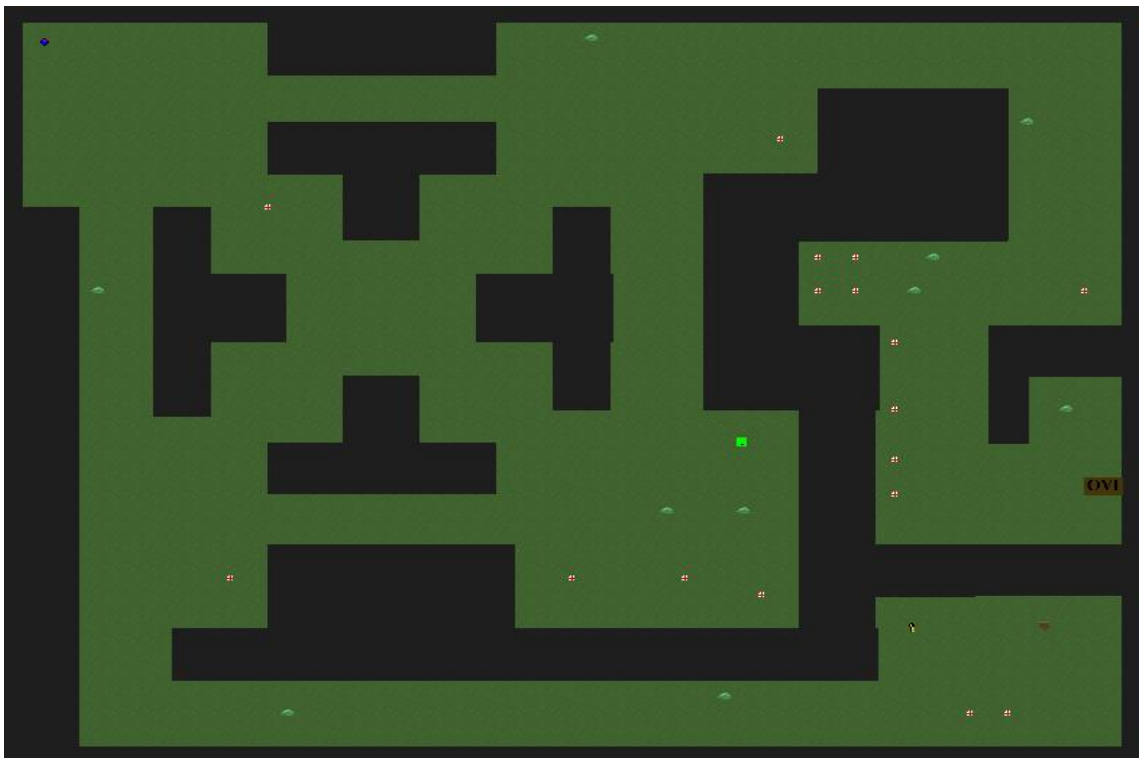


Kuva 24. Vasemmalla avainobjektin fyysisen muodon määrittämisen mahdollistava ikkuna ja oikealla sen ominaisuuksien määrittämiseen tarkoitettu ikkuna.

## 5 Tulokset ja johtopäätökset

GameMaker-pelimoottorissa rakennettiin Benjamin Andersonin opetusvideosarjan pohjalta pelidemo [10]. Pelidemoon rakennettiin tasosuunnitelmien pohjalta tasojen pohjarakenteet seinäobjekteja käyttäen. Tämä sujui pääpiirteittäin hyvin ja oli suhteellisen helppoa. Pelissä satunnaisesti ilmenneen virheen takia pelaajahahmo onnistui kulkemaan seinäobjektien välistä. Siksi seinät kannatti tehdä ja tehtiin mahdollisimman isoista elementeistä. Tämä ratkaistiin venyttämällä tasovalikossa seinäobjekteista suuria kokonaisuuksia, jotka vielä laitettiin reunoistaan päällekkäin. Työssä käytetyssä GameMaker-versiossa tasovalikossa seinäobjektien venyttäminen aiheutti kuitenkin virheilmoituksen, johon väliaikaisena ratkaisuna käytettiin arvojen syöttämistä valikossa suoraan x- ja y-akseleille. Myöhemmin virhe ei enää esiintynyt, mikä nopeutti työskentelyä huomattavasti.

Koska venytetyistä seinäobjekteista seinien rakentaminen täsmälleen tasosuunnitelmien mukaan ei ollut mahdollista, niistä jouduttiin tekemään kulmikkaampia, kuten tasosta 3 otetusta kuvasta 25 näkyy. Tätä pidettiin kuitenkin vain kosmeettisena haittana, eikä sen katsottu kuitenkaan vaikuttavan oleellisesti tasojen toimivuuteen.



Kuva 25. Omien tasosuunnitelmien pohjalta GameMakerissä rakennettu taso 3

Pelidemopohjasta puuttuneet tasopohjasuunnitelmissa esitetyt peliobjektit luotiin ja ohjelmoitiin itse, jotta pelidemossa voitaisiin toteuttaa tasosuunnitelmien vaatimat toiminnot. Tämä oli jo hieman haasteellisempaa, mutta koska opetusvideosarjassa, jonka pohjalta pelidemon pohja luotiin, selitettiin riittävän seikkaperäisesti, miten objekteja luodaan ja miten niitä hallitaan, objektien luonti ja ohjelmointi onnistui muutaman yrityksen ja erehdys -kokemuksen jälkeen.

Itse luotuja peliobjekteja olivat vartijat (ei 1-vaikeustason vartijat), E-esineet ja avaimet. Peliobjekteista vartijat perustuivat 1-vaikeustason vartija peliobjektiin, kun taas E-esine ja avaimet luotiin alusta asti itse. Avaimen käyttöä varten luotiin skripti, joka havaitsee onko pelaajalla avainta, kun hän menee tason oville ja päästää läpi vain pelaajat, joilla on avain. Jos pelaajalla on avain ja tieto kuinka monta esinettä pelaaja on löytänyt siihen mennessä, pelihahmon ruudulla koko ajan näkyviin tietoihin lisätään siitä ilmoittava teksti. E-esineillä ei ole muuta käyttötarkoitusta kuin se kunnia, että löytää kaikki esineet pelin aikana.

Tasosuunnittelun toimivuuden testaamiseksi järjestettiin testitilaisuus, johon osallistui kolme testihenkilöä. Testihenkilöt olivat 10-vuotias poika (A) ja 36-vuotiaat mies (B) ja nainen (C), joista A pelaa digitaalisia pelejä päivittäin, B noin kerran kuukaudessa ja C useamman kerran viikossa. Testin alussa testaajille kerrottiin pelin vaatimat ohjausnäppäimet, mutta ei muuta. Testaajat eivät nähneet tasosuunnitelmia ennakkoon eivätkä saaneet seurata muiden testaajien pelaamista ennen omaa testipeliään. Testihenkilö A hävisi pelidemon tason 3 puolivälissä, B ja C pääsivät läpi kaikki tasot. Testissä oli tarkoitus pelata vain kerran, jottei pidempi kokemus pelistä vaikuttaisi testituloksiin. Siksi henkilö A ei pelannut uudestaan yrittääkseen läpäistä kaikki tasot. Tasot suunniteltiin läpäistäväksi kerralla tavoitteena se, että pelaaja olisi flow-tilassa.

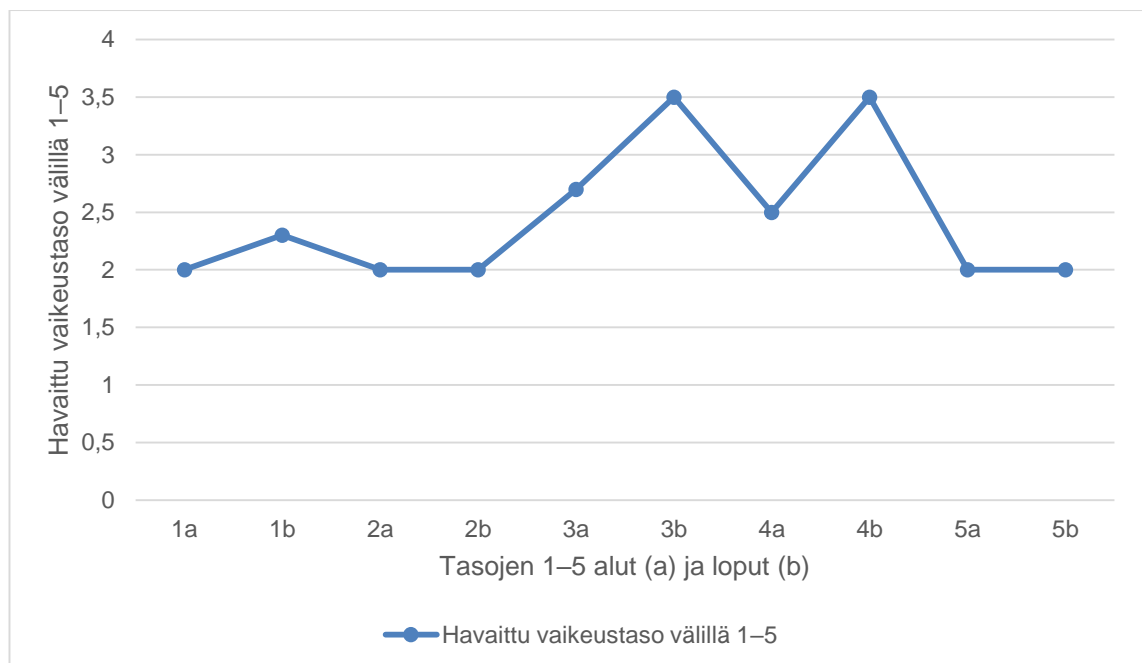
Testipelaamista seurattiin vierestä, ja jokaisen tason jälkeen esitettiin ennakkoon laaditut kysymykset (ks. liite 1). Kysymyksiin 1–7 vastattiin numeroasteikolla 1–5, ja näistä vastauksista laadittiin taulukot 2 ja 3.



Taulukko 2. Pelidemon testaamisen yhteydessä esitettyjen kysymysten 1 ja 2 vastaukset

Arvioi tason alku- ja loppupuolen vaikeustaso asteikolla 1–5, kun yksi tarkoittaa helppoa ja viisi vaikeaa											
	Numero tarkoittaa tasoa, a tason alkupuolta ja b tason loppupuolta										
Testihenkilö	1a	1b	2a	2b	3a	3b	4a	4b	5a	5b	Keskiarvo
A	2	3	3	3	4	-	-	-	-	-	3,0
B	2	3	1	2	1	3	2	3	2	2	2,1
C	2	1	2	1	3	4	3	4	2	2	2,4
<b>Keskiarvo</b>	2,0	2,3	2,0	2,0	2,7	3,5	2,5	3,5	2,0	2,0	2,5

Taulukon 2 perusteella luodusta kaaviosta kuvassa 26 havaittiin, että tasosuunnitelmat pysyivät tasojen 1, 2 ja 5 kohdalla aika hyvin tasapainossa vaikeustason suhteen, koska havaittu vaikeustaso pysyi suunnilleen samalla tasolla. Tasojen 3 ja 4 aikana vaikeustaso nousi ja laski huomattavasti, mikä aiheutti ahdistusta ja myös ikävistymistä eikä flow-tilassa pysytty. Yleisesti kuitenkin tavoiteltiin keskimääräistä vaikeustasoa välillä 3–4, joten tasot 1, 2 ja 5 olivat liian helppoja, kun taas loppujen lopuksi tasot 3 ja 4 olivat vaikeustason puolesta lähempänä flow-tilaa.



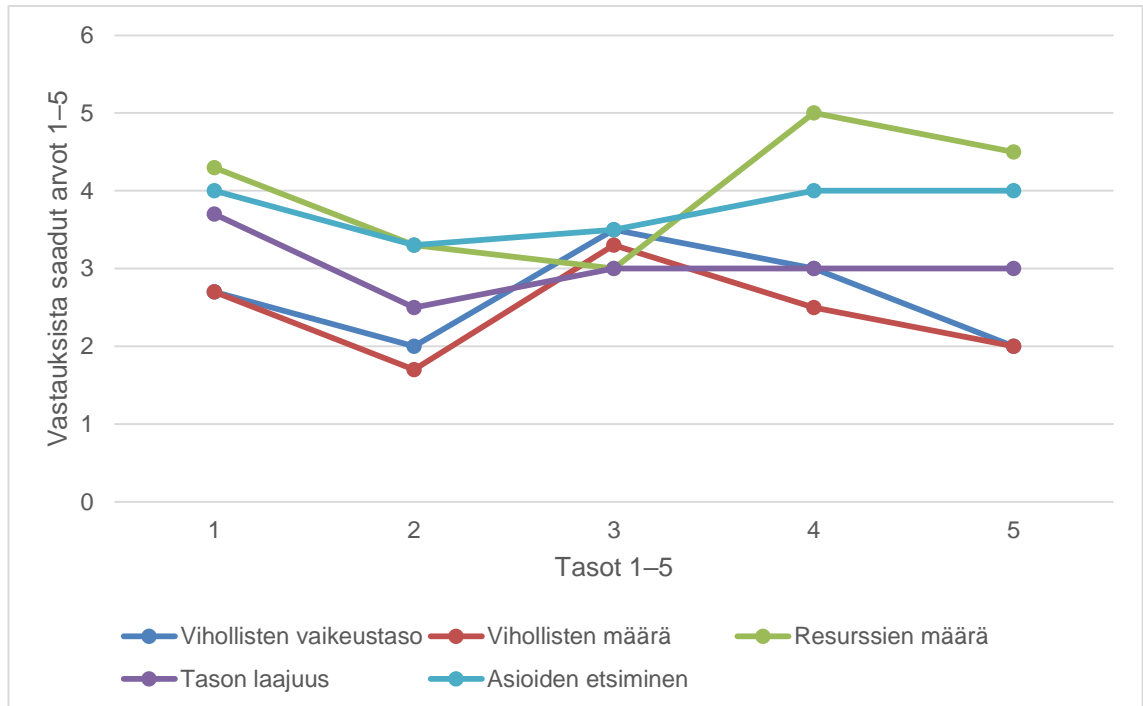
Kuva 26. Tasojen aikana havaitut vaikeustason muutokset

Taulukko 3. Pelidemon testaamisen yhteydessä esitettyjen kysymysten 3–7 vastaukset

<b>Pelidemon testaamisen yhteydessä esitettyjen kysymysten 3–7 vastaukset</b>						
<b>3. Arvioi tason vihollisten vaikeustaso asteikolla 1–5, kun yksi tarkoittaa helppoa ja viisi vaikeaa</b>						
	Tasot					
Testihenkilö	1	2	3	4	5	Keskiarvo
A	3	2	-	-	-	2,5
B	3	2	3	3	2	2,6
C	2	2	4	3	2	2,6
<b>Keskiarvo</b>	2,7	2,0	3,5	3,0	2,0	2,6
<b>4. Arvioi tason vihollisten määrää asteikolla 1–5 kun yksi tarkoittaa liian vähän ja viisi liikaa</b>						
Testihenkilö	1	2	3	4	5	Keskiarvo
A	3	3	4	-	-	3,3
B	3	1	3	2	2	2,2
C	2	1	3	3	2	2,2
<b>Keskiarvo</b>	2,7	1,7	3,3	2,5	2,0	2,4
<b>5. Arvioi tason resurssien määrää asteikolla 1–5, kun yksi tarkoittaa liian vähän ja viisi liikaa.</b>						
Testihenkilö	1	2	3	4	5	Keskiarvo
A	4	3	1	-	-	2,7
B	5	4	4	5	5	4,6
C	4	3	4	5	4	4,0
<b>Keskiarvo</b>	4,3	3,3	3,0	5,0	4,5	4,0
<b>6. Arvioi tason laajuus asteikolla 1–5, kun yksi tarkoittaa liian pieni ja viisi liian suuri.</b>						
Testihenkilö	1	2	3	4	5	Keskiarvo
A	5	-	-	-	-	5,0
B	3	3	3	3	3	3,0
C	3	2	3	3	3	2,8
<b>Keskiarvo</b>	3,7	2,5	3,0	3,0	3,0	3,0
<b>7. Arvioi tasolla asioiden etsiminen asteikolla 1–5, kun yksi tarkoittaa turhauttavaa ja viisi tosi kivaa.</b>						
Testihenkilö	1	2	3	4	5	Keskiarvo
A	5	2	-	-	-	3,5
B	4	4	3	4	4	3,8
C	3	4	4	4	4	3,8
<b>Keskiarvo</b>	4,0	3,3	3,5	4,0	4,0	3,8

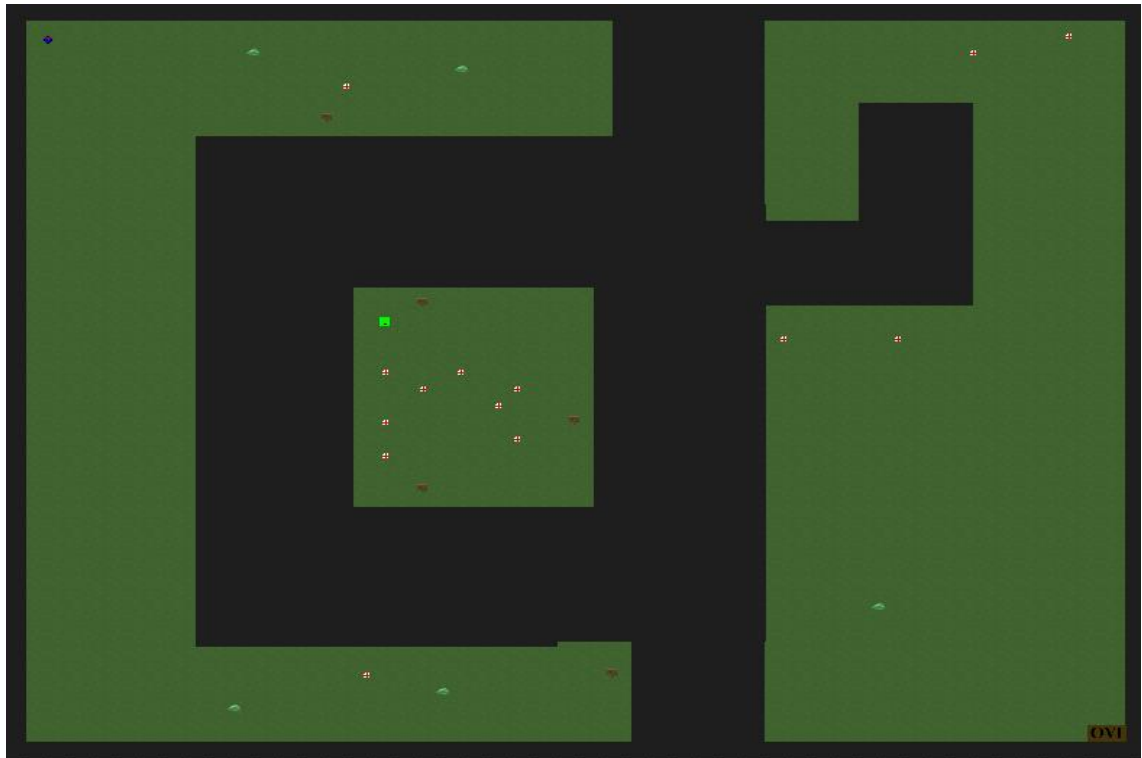
Kuvasta 27 havaitaan, että tason 3 kohdalla testeissä saatujen vastausten arvot osuvat joko vihollisten vaikeustason tasosuunnitelmissa tavoiteltujen arvojen 3–4 välille tai lähelle tasosuunnitelmien tavoitearvoa 3 tason laajuuden, asioiden etsimisen, vihollis-

ten ja resurssien määrän suhteen. Kaikilla tasoilla parhaiten suunnitelmissa onnistui tason laajuuden toteutus, jolla ei tarkoiteta tason fyysistä kokoa, vaan tason koettua laajuutta luotujen reittien kautta.



Kuva 27. Taulukossa 3 listattujen kysymyksistä 3–7 saatujen tulosten mukaan laadittu kaavio

Kuvan 28 tasolla 5 oli suurin hajonta arvojen suhteen, mutta se ei välttämättä johtunut tasosuunnittelusta, vaan viimeisen tason luonteesta. Tosin viimeisellä tasolla oli vähemmän vihollisia kuin muilla tasoilla, ja niiden vaikeustason alhaisuus voidaan hyvin havaita kuvasta 27.



Kuva 28. Taso 5

Testitulosten perusteella voi päätellä, että luomalla hyvät peli- ja tasosuunnittelun teoriaan perustuvat tasosuunnitelmat voidaan päästä jo lähelle haluttua lopputulosta, ilman että tasoja on käytännössä edes toteutettu. Kokeilu ja erehdy -tyyppisen tasojen suunnittelun ja toteutuksen sijaan, hyvät teoriaan perustuvat tasosuunnitelmat voivat säästää peliprojekteissa paljon aikaa ja rahaa. Säästyneitä rahoja voidaan käyttää markkinointiin ja pelien muiden osa-alueiden toteuttamiseen.

Pelidemoa varten luotujen viiden tason jälkeen tulevat tasot voisivat sijoittua vankileirin ympäröimään metsään. Tasojen viihdyttävyyttä voitaisiin parantaa luomalla huomattavasti enemmän peliobjekteja, kuten erilaisia esineitä ja vihollisia. Tasojen nykyisiä haasteita olisi hyvä laajentaa luomalla lisää erilaisia arvoituksia ja tehtäviä. Testeissä havaittu liian alhainen pelaajan havaitsema tasojen ja vihollisten vaikeustaso pitäisi saada nostettua korkeammalle ja tasattua arvojen 3–4 välille kaikilla tasoilla.

GameMaker on mielestäni hyvä työväline ainakin peli- ja tasosuunnittelun alkuvaiheissa oleville. Sillä voi toteuttaa kohtuullisen helposti yksinkertaisia pelejä ja niille tasoja.

Ilmaisversiossa voi luoda pelistä Windowsissa suoritettavan exe-päätteisen tiedoston, jonka avulla peliä voi pelata ilman GameMaker-ohjelmaa. Työssä toteutetun pelidemon exe-päätteisen tiedoston lopullinen koko oli vain hieman yli 2 megatavua, joten sen voi jakaa helposti vaikka kavereiden ihmeteltäväksi.

## 6 Yhteenveto

Insinööriyön tavoitteena oli tehdä kiinnostava tasosuunnittelu 2D-peliin. Siinä pyrittiin löytämään hyviä käytännön tasosuunnittelumalleja, joiden avulla pelin tasoista saataisiin luotua pelaajaa kiinnostavat ja pelaajalle riittävän haastavat. Työssä tehtiin teoriaan ja itse keksittyyn lähtötarinaa perustuvat tasosuunnitelmat, jotka sisälsivät tarinan lisäksi haasteiden hierarkiakaavion, tasojen pohjamallit, peliobjektien sijoittelun ja tasojen tasapainottamisen vaikeustason suhteen.

Tasosuunnitelmien perusteella toteutettiin viisi tasoa sisältävä pelidemo, jonka koostamisessa käytettiin apuna GameMaker-pelimootoria. Tasojen luominen GameMakerissa osoittautui kohtuullisen helpoksi. Ongelmia aiheutti lähinnä pelidemopohjassa havaittu virhetilanne, jossa pelihahmo pääsi seinäobjektien läpi. Tämä kuitenkin saatiin ratkaistua seinäobjekteja suurentamalla ja muokkaamalla tasojen tasopohjia hieman tasosuunnitelmia kulmikkaammiksi. Jotta pelidemo vastaisi tasosuunnitelmia paremmin, siihen luotiin itse muutama lisäpeliobjekti, joita ei ollut pelidemon pohjana käytetyssä pelidemopohjassa.

Testien perusteella voitiin havaita, että tasosuunnitelmien perusteella toteutetut tasot pysyivät tasojen 1, 2 ja 5 kohdalla aika hyvin tasapainossa vaikeustason suhteen. Tasojen 3 ja 4 aikana vaikeustaso nousi ja laski huomattavasti, mikä aiheutti ahdistusta ja myös ikävistymistä eikä flow-tilassa aina pysytty. Yleisesti kuitenkin tasot 1, 2 ja 5 olivat liian helppoja, kun taas tasot 3 ja 4 olivat vaikeustasoltaan lähempänä tavoiteltua flow-tilaa. Testeistä saatujen vastausten arvojen mukaan tason 3 tasosuunnitelma oli onnistunein. Tasolla 5 oli suurin hajonta arvojen suhteen, mutta se ei välttämättä johtunut tasosuunnittelusta, vaan viimeisen tason luonteesta.

Testitulosten perusteella voi päätellä, että luomalla hyvät peli- ja tasosuunnittelun teoriaan perustuvat tasosuunnitelmat voidaan päästä lähelle haluttua lopputulosta, ilman että tasoja on käytännössä edes toteutettu. GameMaker-pelimoottori on mielestäni hyvä työväline ainakin peli- ja tasosuunnittelun alkuvaiheessa oleville, koska sillä voi toteuttaa kohtuullisen helposti yksinkertaisia pelejä ja niille tasoja.

## Lähteet

- 1 Adams, Ernest. 2014. Fundamentals Of Game Design. Third edition. USA: New Riders.
- 2 Rouse, Richard. 2005. Game Design: Theory & Practice. 2nd ed. USA: Wordware Publishing.
- 3 Creating A Missile Command Clone in Scratch. 2012. Verkkodokumentti. Logical moon. <<http://logicalmoon.com/2012/08/creating-a-missile-command-clone-in-scratch/>>. Luettu 12.4.2016.
- 4 Missile Command: Victor Sandberg Breaks 30-Year-Old Arcade Record With 81 Million Points. 2013. Verkkodokumentti. The Huffington Post. <[http://www.huffingtonpost.co.uk/2013/03/18/missile-command-victor-sandberg-record\\_n\\_2901172.html](http://www.huffingtonpost.co.uk/2013/03/18/missile-command-victor-sandberg-record_n_2901172.html)>. Luettu 12.4.2016.
- 5 SimCity 3000 Unlimited. 2016. Verkkodokumentti. GameFAQs. <<http://www.gamefaqs.com/pc/256694-simcity-3000-unlimited/images/23>>. Luettu 12.4.2016.
- 6 Wolfenstein 3D. 2016. Verkkodokumentti. DOSome Games. <<http://dosomegames.com/GamePages/Wolfenstein3D.html>>. Luettu 12.4.2016.
- 7 Horowitz, Steve & Looney, Scott. 2014. The essential guide to game audio: the theory and practice of sound for games. USA: Focal Press.
- 8 Top 5 Game Engines For Developers. 2016. Verkkodokumentti. Framebench. <<http://blog.framebench.com/top-5-game-engines-developers/>>. Luettu 12.4.2016.
- 9 Anderson, Benjamin. 2016. Official HeartBeast GameMaker Website. Verkkodokumentti. <<http://www.heartbeast.co/>>. 10.4.2016. Luettu 10.4.2016
- 10 Anderson, Benjamin. 2015. [Beginner Tutorial] Make an RPG in GameMaker [P1] Basic Movement and Collisions. Verkkodokumentti. <<https://www.youtube.com/watch?v=WB2alpO62GM>>. 25.8.2015. Luettu 10.4.2016.

**Pelidemon testaamisen yhteydessä esitetyt kysymykset**

1. Arvioi tason alkupuolen vaikeustaso asteikolla 1–5, kun yksi tarkoittaa helppoa ja viisi vaikeaa.
2. Arvioi tason loppupuolen vaikeustaso asteikolla 1–5 kun yksi tarkoittaa helppoa ja viisi vaikeaa.
3. Arvioi tason vihollisten vaikeustaso asteikolla 1–5 kun yksi tarkoittaa helppoa ja viisi vaikeaa.
4. Arvioi tason vihollisten määrää asteikolla 1–5, kun yksi tarkoittaa liian vähän ja viisi liikaa.
5. Arvioi tason resurssien määrää asteikolla 1–5, kun yksi tarkoittaa liian vähän ja viisi liikaa.
6. Arvioi tason laajuus asteikolla 1–5, kun yksi tarkoittaa liian pieni ja viiden liian suuri.
7. Arvioi tasolla asioiden etsiminen asteikolla 1–5, kun yksi tarkoittaa turhauttavaa ja viisi tosi kivaa.
8. Olivatko E-esineet kiinnostavia, vaikka niistä ei saanut kuin kunniaa?
9. Onko muita kommentteja tasoon liittyen?