



OULUN YLIOPISTO
UNIVERSITY of OULU

TEKNILLINEN TIEDEKUNTA

**STAGE-GATE -MALLIN HYÖDYNTÄMINEN
PELINKEHITYKSESSÄ**

Juuso Råman

TUOTANTOTALOUS

Kandidaatintyö

Huhtikuu 2017



OULUN YLIOPISTO
UNIVERSITY of OULU

TEKNILLINEN TIEDEKUNTA

**STAGE-GATE -MALLIN HYÖDYNTÄMINEN
PELINKEHITYKSESSÄ**

Juuso Råman

Ohjaaja: Henri Jounila

TUOTANTOTALOUS

Kandidaatintyö

Huhtikuu 2017

TIIVISTELMÄ

OPINNÄYTETYÖSTÄ Oulun yliopisto Teknillinen tiedekunta

Koulutusohjelma (kandidaatintyö, diplomityö) Tuotantotalous		Pääaineopintojen ala (lisensiaatintyö)	
Tekijä Råman, Juuso		Työn ohjaaja yliopistolla Jounila H, tohtorikoulutettava	
Työn nimi Stage-Gate -mallin hyödyntäminen pelinkehityksessä			
Opintosuunta	Työn laji Kandidaatintyö	Aika Huhtikuu 2017	Sivumäärä 27 s.
<p>Tiivistelmä</p> <p>Viime vuosina pelinkehityksessä on ilmennyt ongelmia, jotka ovat aiheuttaneet arvioitujen kustannusten ja aikataulujen ylittymistä. Tämän kandidaatintyön tavoitteena on selvittää, kuinka Stage-Gate -mallia hyödynnetään pelinkehityksessä ratkaisemaan nämä ongelmat. Työssä selvitetään myös, millaisia muutoksia Stage-Gate -malliin tulee tehdä, jotta sitä voidaan soveltaa pelinkehitykseen.</p> <p>Tutkimus suoritettiin kirjallisuuskatsauksena, joka sisältää keskeisimmän teorian Stage-Gate -malliin, ketteriin menetelmiin, Agile-Stage-Gate -hybridimalliin sekä pelinkehitykseen ja sen haasteisiin liittyen. Stage-Gate on projektinhallinnan malli, joka jakaa tuotekehitysprojektin hyvin määriteltyihin vaiheisiin, joita erottavat tiukat läpäisyvaatimukset omaavat portit. Stage-Gate -mallilla keskitytään kehityksen suunnitteluun ja laadukkaaseen toteutukseen mahdollisimman vähillä muutoksilla. Ohjelmistokehityksessä hyödynnetään ketteriä menetelmiä, jotka pitävät muutoksia välttämättömyyksinä. Ketterässä ohjelmistokehityksessä tuote kehitetään iteratiivisesti viimeisimpiin vaatimuksiin nopeasti reagoiden. Stage-Gate -malli ja ketterät menetelmät eivät kuitenkaan ole toisiaan poissulkevia, sillä viime vuosikymmenen aikana on kehitetty molempien hyvät puolet yhdistäviä hybridimalleja.</p> <p>Pelinkehityksen suurimmat haasteet ovat projektin laajuuden arvioiminen, viime hetken muutoksiin reagoiminen sekä monialaisten tiimien johtaminen. Pelinkehitysprojekti tapahtuu hyvin määriteltyjen vaiheiden läpi, joten sen johtaminen Stage-Gate -mallia hyödyntämällä auttaa pitämään projektin suunnitellun budjetin ja aikataulun sisällä. Monialaisten tiimien aiheuttamiin kommunikaatio- ja teknisiin ongelmiin auttaa yhteisen vision ja aikataulun mukaan työskenteleminen. Kehityksen aikana tapahtuviin muutoksiin sekä hyvän pelattavuuden etsimiseen käytetään ketteriä menetelmiä.</p> <p>Kirjallisuuskatsauksen perusteella on selvää, että pelinkehitykseen ei riitä mikään aikaisempi menetelmä sellaisenaan. Pelinkehityksessä yhdistetään Stage-Gate -projektin tarkka rakenne ja huolellinen suunnittelu ketterien menetelmien iteratiiviseen kehitykseen. Uusi Agile-Stage-Gate on hyvä esimerkki tällaisesta hybridimallista, johon pelinkehittäjien kannattaa perehtyä suunnitellessaan tulevia projekteja.</p>			
Muita tietoja			

ABSTRACT FOR THESIS

University of Oulu Faculty of Technology

Degree Programme (Bachelor's Thesis, Master's Thesis) Industrial Engineering and Management		Major Subject (Licentiate Thesis)	
Author Râman, Juuso		Thesis Supervisor Jounila H, doctoral student	
Title of Thesis Using the Stage-Gate model in Game Development			
Major Subject	Type of Thesis Bachelor's Thesis	Submission Date April 2017	Number of Pages 27 p.
Abstract <p>In the recent years, game development has faced problems causing budget and schedule overruns. The goal of this bachelor's thesis is to discover how the Stage-Gate model is used in game development to solve these problems. This thesis also explores the modifications required for the Stage-Gate model to be suitable for game development.</p> <p>The research was conducted as a literature review containing the essential theories behind the Stage-Gate model, Agile methods, the Agile-Stage-Gate hybrid model as well as game development and its challenges. Stage-Gate is a project management model that divides a product development project into well-defined stages separated by gates with strict throughput requirements. The Stage-Gate model focuses on planning the development and quality of execution with minimal changes on the way. Software development uses Agile methods which considers changes as a necessity. In Agile software development, the development happens iteratively with fast reactions to the latest requirements. However, the Stage-Gate model and Agile methods are not mutually exclusive since hybrid models combining the merits of both have been developed in the last decade.</p> <p>The most significant challenges in game development are estimating the scale of the project, reacting to last-minute changes and managing multidisciplinary teams. A game development project goes through well-defined stages so managing it using the Stage-Gate model supports keeping the project within budget and schedule. Working for mutual vision and schedule helps solving communication and technical problems caused by multidisciplinary teams. Agile methods are used to manage changes during development and to find the right gameplay.</p> <p>Based on the literature review, any of the previous methods is not suitable for game development as such. Game development combines the strict project structure and thorough planning of the Stage-Gate model with the iterative development of Agile. The new Agile-Stage-Gate is a good example of such hybrid model in which game developers should keep an eye on while planning future projects.</p>			
Additional Information			

SISÄLLYSLUETTELO

TIIVISTELMÄ

ABSTRACT

SISÄLLYSLUETTELO

1 JOHDANTO	5
2 STAGE-GATE -MALLI JA KETTERÄT MENETELMÄT	7
2.1 Mikä on Stage-Gate -malli	7
2.2 Stage-Gate -mallin vaiheet	8
2.3 Stage-Gate -mallin portit	9
2.4 Ketterä ohjelmistokehitys	10
2.5 Agile-Stage-Gate -hybridimalli	11
3 PELINKEHITYS	13
3.1 Pelinkehityksen haasteet ja ongelmat	13
3.2 Pelinkehitysprojekti	14
3.3 Go/Kill -päätökset pelinkehityksessä	17
3.4 Ketterien menetelmien hyödyntäminen	18
3.5 Monialaiset kehitystiimit	18
4 POHDINTA	20
5 YHTEENVETO	22
LÄHDELUETTELO	24

1 JOHDANTO

1970-luvun alun ensimmäisistä kaupallisista videopeleistä lähtien videopeliteollisuus on nopeasti kasvanut pienistä kohdistetuista markkinoista valtavirtaan (Wolf 2008, s. 29). Peliteollisuuden trendejä tutkivan SuperDatan vuosikatsauksen mukaan 2016 oli 91 miljardin dollarin kokonaistuloilla tähän mennessä suurin vuosi kansainvälisillä videopelimarkkinoilla. Tämä on suurelta osin mobiilipelien ansiota, sillä mobiilipelien kokonaistulot ovat kasvaneet 2,9 miljardista dollarista 41 miljardiin dollariin pelkästään viimeisen vuosikymmenen aikana. (Gartner 2007; SuperData 2016)

Mobiilipelien ohessa suureen suosioon ovat viime vuosina nousseet niin sanotut Indie-pelit, joita pienet tiimit tai jopa yksittäiset henkilöt kehittävät ilman julkaisijan taloudellista tukea. 2000-luvun ensimmäisellä vuosikymmenellä markkinoille tulleet digitaaliset jakelukanavat, kuten Applen App Store sekä Valven Steam, ovat pienentäneet kynnystä saada peli julkaistuksi kaikkien saataville. (Behrmann ym. 2012; McGuire & Jenkins 2009, s. 27-28) Näin ovat syntyneet muun muassa sellaiset menestystarinat kuten Minecraft ja Angry Birds (Bisset 2013; Downin 2011).

Vastaavasti kuluttajien odotukset täysihintaisten pelien teknisille vaatimuksille kasvavat jatkuvasti tietokoneiden laskentatehon kehittymisen myötä (Fuller & Millett 2011, s. 25). Tämän vuoksi viimeisintä tekniikkaa edustavien AAA-pelien keskimääräiset kehityskustannukset ovat nousseet kymmeneen miljooniin dollareihin (Nunneley 2010). Julkaisija, joka on vastuussa pelin rahoittamisesta ja markkinoinnista, saa rahansa takaisin mahdollisten tuottojen kera vasta julkaisun jälkeen, joten pelin kehityksen seuraaminen ja hallinta ovat tärkeässä roolissa (McGuire & Jenkins 2009, s. 25-26).

Pelinkehityksessä ilmenneet ongelmat ovat selvästi vaivanneet peliteollisuutta viime vuosien aikana. Tästä ovat osoituksena muun muassa jatkuvat viivästymiset ilmoitetusta julkaisupäivästä, pelien julkaiseminen keskeneräisenä ja jopa satunnaiset peruutukset. (Ashcraft 2010; Goldfarb 2017) Esimerkiksi vuonna 2011 julkaistun Duke Nukem Foreverin kehitykseen osallistui neljä eri tiimiä 15 vuoden kehityksensä aikana, ja sen arvioidaan aiheuttaneen pelin suunnittelijalle kymmenien miljoonien dollarien tappiot (Ashcraft 2010). Aiheutuneet tappiot eivät kuitenkaan aina ole vain taloudelliset, sillä huonosti kehitetyn pelin julkaiseminen on usein kolaus myös pelistudion maineelle (Crawley 2015).

Vaikka pelinkehitysprojekti tapahtuu usein hyvin määriteltyjen vaiheiden mukaan, ei ohjelmistokehityksessä perinteisesti käytettyä vesiputousmallia ole pidetty siinä toimivana sellaisenaan (Kanode & Haddad 2009). Pelinkehitysprojekti on usein pitkä ja siihen osallistuu paljon ihmisiä, joten projektin lopputulosta on vaikea määrittää etukäteen (Bates 2004). Mobiilipelien puolella on kuitenkin saatu menestyksekkäitä tuloksia viime aikoina muun muassa Stage-Gate -mallilla ja sen muunnelmilla. Esimerkiksi mobiili- ja Facebook-pelejä kehittävän Woogan ”hittifiltteriksi” kutsutulla lähestymistavalla kehitetään paljon konsepteja mahdollisista peleistä, mutta vain muutama peli päätyy julkaisuasteelle. Hittifiltterillä seurataan kehityksen edetessä vaiheittain, onko pelillä potentiaalia olla mahdollinen hitti, ja tehdään päätös kannattaako projektia jatkaa. (Wooga 2014)

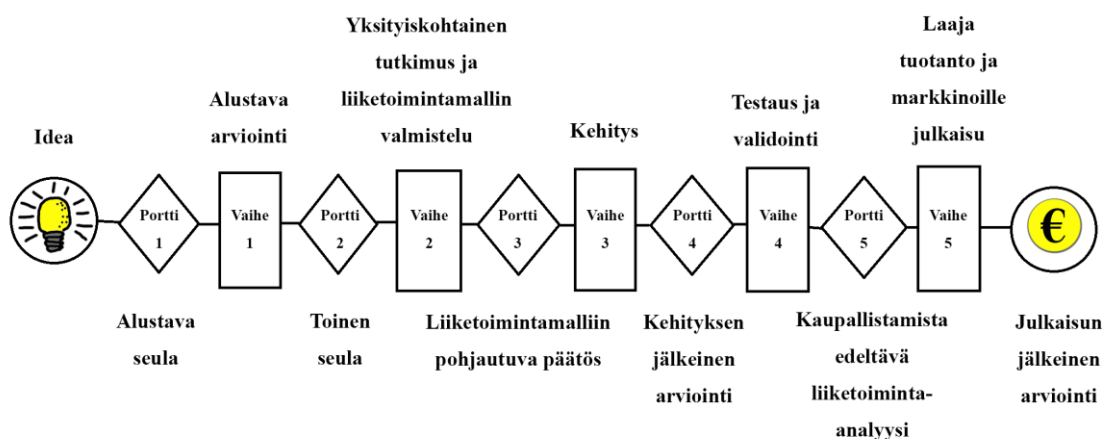
Työni tavoitteena on selvittää, miten Robert G. Cooperin Stage-Gate -mallia voidaan hyödyntää yleisesti pelinkehityksessä ratkaisemaan siinä viime vuosina ilmenneet ongelmat. Selitän saatavilla olevan kirjallisuuden pohjalta, mikä Stage-Gate -malli on ja kuinka sitä on vastikään kehitetty soveltumaan paremmin ohjelmistokehitykseen. Käyn myös pelinkehitysprojektin läpi vaihe vaiheelta, ja samalla selvitän, kuinka Stage-Gate -mallia voidaan käytännössä hyödyntää. Tutkimuskysymykseni on siis, miten Stage-Gate -mallia voidaan hyödyntää pelinkehityksessä, ja millaisia muutoksia se vaatii soveltuakseen pelinkehitykseen.

2 STAGE-GATE -MALLI JA KETTERÄT MENETELMÄT

2.1 Mikä on Stage-Gate -malli

Stage-Gate on Robert G. Cooperin kehittämä malli, jonka avulla uusi tuote kuljetetaan ideasta markkinoille. Sen tavoitteena on tuotekehitysprosessin tehokkuuden kasvattaminen ja uusien tuotteiden onnistumisasteen parantaminen. Stage-Gate -mallilla tuotekehitysprosessi toteutetaan järjestelmällisesti kiinnittäen huomiota toteutuksen laatuun. Näin prosessi nopeutuu, kriittisiä vaihteita ei unohdeta ja resurssit hyödynnetään tehokkaammin. (Cooper 1990)

Yksittäiset yritykset saattavat viitata tuotekehitysmalleihinsa eri nimillä, kuten Woogan hittifiltteri, mutta käytännössä Stage-Gate -malliset lähestymistavat ovat hyvin samanlaisia. Stage-Gate -mallissa tuoteinnovaatiota pidetään prosessina, jota parannetaan jatkuvasti samoin kuin esimerkiksi tuotteen valmistusprosessia: vähentämällä siitä muuttujia. Samoin kuin tuotteen valmistusprosessi on jaettu erillisille työpisteille, on Stage-Gate -mallisessa tuotekehitysprosessissa hyvin määritellyt vaiheensa, joiden kautta tuote etenee ideasta markkinoille. Jokaista vaihetta edeltää portti, jossa tarkastellaan täyttääkö tuote sille annetut kriteerit, ja voidaanko vaiheeseen siirtyä. Tyypillinen Stage-Gate -malli on esitettyä kuvassa 1. (Cooper 1990)



Kuva 1. Tyypillinen Stage-Gate -malli (mukaillen Cooper 1990).

2.2 Stage-Gate -mallin vaiheet

Riippuen yrityksestä ja tuotteesta, Stage-Gate -malli sisältää yleensä neljästä seitsemään vaihetta ja porttia. Tyypillinen malli sisältää viisi vaihetta: alustava arviointi, yksityiskohtainen tutkimus ja liiketoimintamallin valmistelu, kehitys, testaus ja validointi sekä laaja tuotanto ja markkinoille julkaisu. Varsinaisia vaiheita edeltää idean saaminen, jonka myöskin tulee läpäistä oma porttinsa. (Cooper 1990)

Vaiheet ovat yleensä toinen toistaan kalliimpia, mutta samanaikaisesti varmuus onnistumisesta kasvaa prosessin edetessä, joten siihen liittyvä riski voidaan hallita. Jokainen vaihe sisältää ennalta määrättyjä ja toisiinsa liittyviä toimenpiteitä, jotka usein toteutetaan yhtä aikaa. Tämä nopeuttaa prosessin etenemistä ja varmistaa, että yrityksen eri osat työskentelevät yhteistä päämäärää eli seuraavaa porttia varten. (Cooper 1990)

Stage-Gate -mallille on ominaista myös vahva markkinasuuntautuneisuus. Prosessin eri vaiheet sisältävät paljon markkinoihin liittyviä aktiviteetteja, kuten asiakastarpeiden selvitystä, kilpailija-analyysyjä ja yksityiskohtaisen markkinointisuunnitelman laatimisen. Stage-Gate -mallissa nämä aktiviteetit ovat suunniteltu sisällytettäväksi jo etukäteen sen sijaan, että tuotekehityksen suuntaa muutettaisiin viime hetkellä, mikä taas aiheuttaisi ylimääräisiä kustannuksia. (Cooper 1990) Monet tuotteet epäonnistuvat pyrkiessään markkinoille vain tekemällä samoin kuin sen kilpailijat, mutta Stage-Gate -mallilla saavutetaan etumatka suunnittelemalla tuote asiakkaiden tarpeiden pohjalta (Cooper 2000).

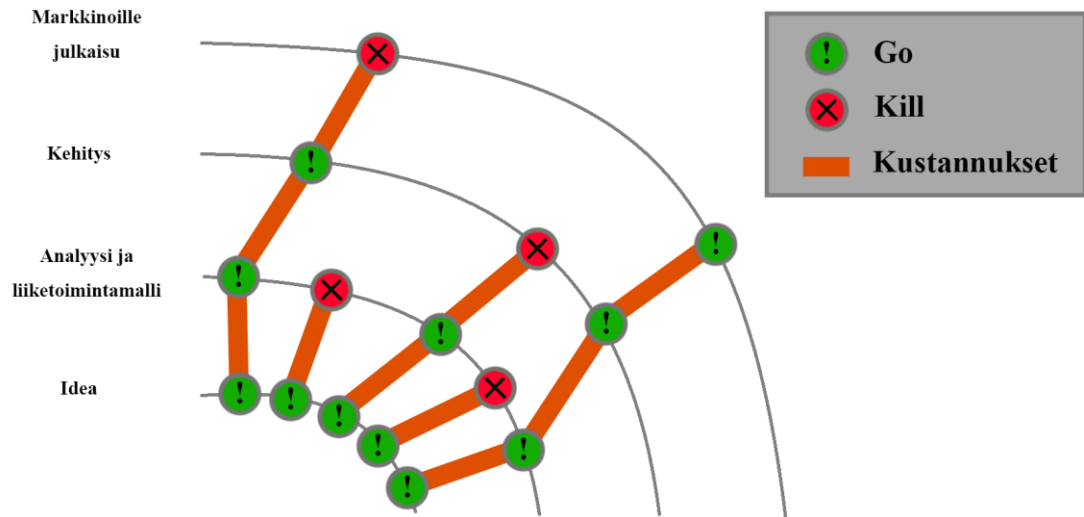
Kehitystä edeltäviin vaiheisiin käytetäänkin enemmän rahaa kuin tavallisissa tuotekehitysprojekteissa myös siksi, jotta itse kehitysvaihe tapahtuu nopeammin ja ilman odottamattomia ongelmia. Markkinoille julkaistava tuote on harvoin samanlainen kuin alkuperäinen idea, mutta panostamalla kehitystä edeltäviin vaiheisiin voidaan välttyä suurilta ja kalliilta muutoksilta kehityksen aikana ja sen jälkeen. (Cooper 1990) Huonosti määritellyt tuotteen ominaisuudet johtavat usein epäonnistumiseen, joten tuotteen lopullinen muoto, hinta ja kohdemarkkinat tulisivat olla määriteltynä viimeistään kehitysvaiheeseen siirtyessä (Cooper 2000).

2.3 Stage-Gate -mallin portit

Tyypillisen Stage-Gate -mallin portit ovat alustava seula, toinen seula, liiketoimintamalliin pohjautuva päätös, kehityksen jälkeinen arviointi ja kaupallistamista edeltävä liiketoiminta-analyysi. Näitä seuraa myös julkaisun jälkeinen arviointi. Porttien tehtävänä on varmistaa, että laatu pysyy tavoitetulla tasolla. Jokaiselle portille on asetettu omat vaadittavat suoritteensa, jotka toimivat syötteinä portille, sekä laatukriteerit, jotka tuotteen on läpäistävä ennen siirtymistä seuraavaan vaiheeseen. Tulosteena portista saadaan ”Go/Kill/Hold/Recycle” -päätös, eli päätös siitä siirrytäänkö seuraavaan vaiheeseen (Go), lopetetaanko projekti kokonaan (Kill), pysäytetäänkö projekti nykyiseen vaiheeseen (Hold) vai hyödynnetäänkö aikaansaannos toisessa projektissa (Recycle). (Cooper 1990)

Päätöksen teko on portinvartijoina (gatekeeper) toimivien ylemmän johdon henkilöiden vastuulla. Heidän tehtäviinsä kuuluu arvioida puolueettomasti täyttävätkö vaiheen suoritteet niille ennalta asetetut kriteerit, arvioida onko projektilla kokonaisuudessaan vielä potentiaalia taloudelliselta ja liiketoiminnalliselta kannalta sekä mahdollisesti tehdä toimintasuunnitelma ja jakaa resurssit seuraavaa vaihetta varten. (Cooper 1990)

Stage-Gate -mallilla tiedostetaan se, että vain harvalla ideatasolla olevalla tuotteella on mahdollisuuksia menestyä kaupallisesti. Kun tuotekehitysprojektissa havaitaan ongelmia, ne koetaan usein vain esteinä, jotka tulee ylittää. Stage-Gate -mallilla kannattamattomat projektit pyritään tunnistamaan ja lopettamaan jo aikaisessa vaiheessa, jotta resurssit eivät kuluisi liian pitkälle eteneviin ja lopulta huonoihin tuotteisiin. (Cooper 1990) Tehokkaassa tuoteportfolion hallinnassa jokainen tuote nähdään sijoituksena, joten vain parhaiten kannattaviin kannattaa panostaa (Cooper 2000). Kuvassa 2 on esitetty, miltä Stage-Gate -mallia hyödyntävän yrityksen tuoteportfolio voisi näyttää. Kuvasta ilmenee myös, kuinka tuotekehityksen suunta voi muuttua projektin aikana, ja kuinka se lisää kustannuksia. (Cook 2007)



Kuva 2. Stage-Gate -mallia hyödyntävän yrityksen tuoteportfolio (mukaillen Cook 2007).

2.4 Ketterä ohjelmistokehitys

Ohjelmistokehityksen puolella asiakkaiden vaatimukset ja käytössä oleva teknologia muuttuvat nopeasti. Yrityksellä ei usein olekaan tavoitteena toteuttaa alkuperäistä suunnitelmaa vaan vastata asiakkaan tarpeisiin silloin, kun lopullinen tuote toimitetaan. Koska kustannukset tuotteen muuttamiselle nousevat tuotteen kehityskaaren edetessä, on ohjelmistokehityksessä lähdetty miettimään kokonaan uutta lähestymistapaa, jossa kehityksen aikaisia muutoksia pidetään välttämättömyytenä. Tavoitteena on näiden muutosten aiheuttamien kustannusten pienentäminen samalla pitäen tuotteen laatu sen vaatimalla tasolla. (Highsmith & Cockburn 2001)

Näihin tavoitteisiin on lähdetty vastaamaan niin kutsutuilla ketterillä menetelmillä, joihin kuuluvat muun muassa Extreme Programming, Scrum ja DSDM. Jokainen menetelmä vastaa tavoitteisiin omalla tavallaan, mutta peruseriaatteisiin kuuluvat jatkuva ohjelman toimivuus ja tiimin tiivis yhteistyö. Tiimi yrittää saada toimivan ohjelman aikaiseksi jo projektin alussa, ja lähtee kehittämään sitä eteenpäin pienin ja yksinkertaisin askelin. Tiimin tiiviillä yhteistyöllä varmistetaan, että tiedon ja ideoiden siirto tapahtuu nopeasti ja tehokkaasti. Myös asiakkaiden ja rahoittajien kanssa pysytään yhteisymmärryksessä projektin etenemisestä sekä mahdollisista kohdatuista ongelmista ja niiden ratkaisemisesta. (Highsmith & Cockburn 2001)

Käytännössä ketterä kehitys tapahtuu lyhyissä kahdesta kuuteen viikkoon kestävässä iteraatioissa. Jokaiselle iteraatiolle asetetaan omat tavoitteensa siitä, millaiset uudet ominaisuudet jo toimivaan ohjelmaan pyritään lisäämään. Näin myös asiakas pystyy helposti ymmärtämään, miten kehitys etenee. Jokaisen iteraation lopussa asiakkaalle annetaan mahdollisuus esittää toiveensa mahdollisista uusista ominaisuuksista, jota tulevissa iteraatioissa kehitetään, tai vanhojen ei-toivottujen ominaisuuksien poistamisesta. Näin tuote on aina asiakkaan viimeisimpien tarpeiden mukainen ja tarvittaessa valmiina toimitettavaksi. (Highsmith & Cockburn 2001)

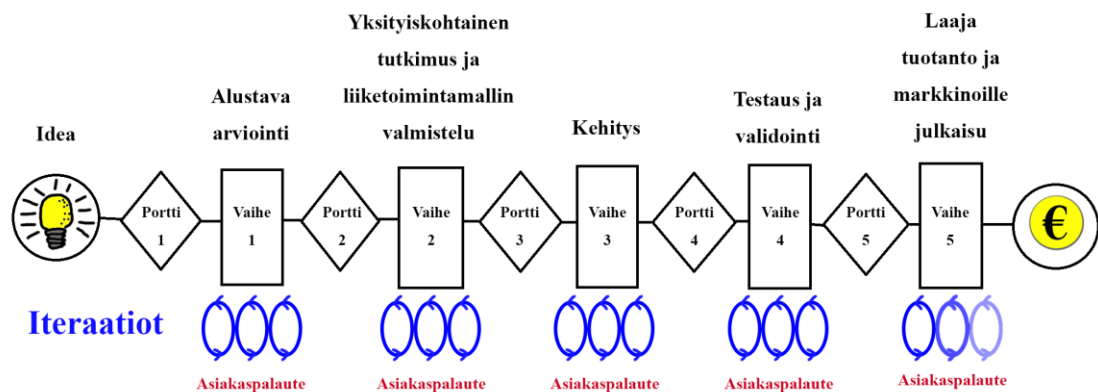
2.5 Agile-Stage-Gate -hybridimalli

2000-luvun ensikymmenen puolella välissä ohjelmistokehittäjät huomasivat, että ketteriä menetelmiä voitiin käyttää myös Stage-Gate -mallin ohessa täydentämään toisiaan. Näin voitiin saavuttaa hyödyt molemmista menetelmistä: parempi projektinhallinta ja vastaaminen asiakkaiden tarpeisiin sekä tiimin välinen kommunikaatio. (Karlström & Runeson 2005) Viime aikoina myös fyysisten tuotteiden valmistajat ovat kiinnostuneet ketteristä menetelmistä selvittääkseen epävakaita markkinoita ja jatkuvasti muuttuvista asiakkaiden vaatimuksista. Näin ovat syntyneet Agile-Stage-Gate -hybridimallit. (Cooper 2016)

Monilla yrityksillä, jotka ovat alkaneet hyödyntää ketteriä menetelmiä tuotekehityksessään, oli jo valmiina käytössään Stage-Gate -malliin perustuva kehitysprosessi. Ketteriä menetelmiä hyödynnetään enimmäkseen tuotteen kehitys- ja testausvaiheissa, joten tuotteen kehityskaari seuraa edelleen Stage-Gate -mallia. Ketterät menetelmät parantavat kehitystiimin sisäistä kommunikaatiota ja tehostavat suunnittelua asiakastarpeiden mukaisesti, mutta samalla keskittyminen kehitystiimiin ja meneillään olevaan iteraatioon vie huomiota muilta tuotekehityksen osallisilta ja pitkän tähtäimen suunnitelmalta. Ketterät menetelmät helpottavat siis päivittäistä työskentelyä kehitysvaiheessa, mutta Stage-Gate -malli tukee laajemmin tuotteen kehityskaarta, helpottaa yhteistyötä muiden tiimien, kuten markkinoinnin kanssa ja antaa paremman kuvan projektin etenemisestä. (Cooper 2016; Karlström & Runeson 2005)

Uudessa Agile-Stage-Gate -hybridimallissa ketteriä menetelmiä voidaan hyödyntää kaikissa Stage-Gate -mallin vaiheissa, ei pelkästään kehitys- ja testausvaiheessa. Vaiheet eivät ole tarkasti etukäteen suunniteltuja kuten perinteisessä mallissa, vaan ne sisältävät

ketterien menetelmien mukaisia iteraatioita, joilla jokaisella on omat tavoitteensa perustuen asiakkaan palautteeseen ja siihen, mikä on ylipäättään mahdollista saavuttaa kyseisen iteraation aikana. (Cooper 2016) Kuvassa 3 on esitetty, kuinka ketterien menetelmien iteraatiot sopivat tyypilliseen viisivaiheiseen Stage-Gate -malliin (Vedsmann ym. 2016). Ohjelmistokehityksessä pyritään siihen, että ohjelma on julkaistavassa kunnossa jokaisen iteraation lopussa, mutta fyysisillä tuotteilla tämä on harvoin mahdollista (Cooper 2016).



Kuva 3. Asiakaspalautteen mukaiset iteraatiot tyypillisessä Stage-Gate -mallissa (mukaillen Vedsmann ym. 2016).

Hybridimalli tasapainottaa Stage-Gate -mallin ja ketterien menetelmien hyvät ja huonot puolet luoden samalla monia tärkeitä etuja. Kehitys iteraatioissa ja asiakaspalautteen kerääminen ovat hyödyllisiä etenkin hyvin innovatiivisista tuotteista puhuessa, koska asiakas ei usein tiedä, mitä tuotteelta haluaa ennen kuin näkee sen. Innovatiivisia tuotteita kehitettäessä myös käytettävä teknologia kehittyy projektin aikana, joten hybridimalli ottaa hyvin huomioon uuden teknologian tuomat uhat ja mahdollisuudet, joita voi olla vaikea ennustaa etukäteen. Hybridimallin käyttäminen voi myös nopeuttaa kehitystä, koska tiimi yrittää jatkuvan paineen alla ennättää seuraavaan tavoitteeseensa iteraation lopussa. Tiimin keskittyminen yhteiseen tavoitteeseen ja parempi kommunikaatio lisäävät myös tuotteen onnistumisvarmuutta. (Cooper 2016)

3 PELINKEHITYS

3.1 Pelinkehityksen haasteet ja ongelmat

Peliteollisuudesta on viime vuosien aikana tullut merkittävä osa viihdeteollisuutta tuoden samalla esiin uusia kannattavia markkinoida (Callele ym. 2005; Cohendet & Simon 2007). Kilpailun ja teknologian kehityksen myötä kuluttajan odotukset peleistä ovat kasvaneet, ja nykyään pelinkehitysprojekti usein kestää vuosia, siihen osallistuu satoja ihmisiä ja se kustantaa kymmeniä miljoonia dollareita (Fuller & Millett 2011, s. 25; Kanode & Haddad 2009; Nunneley 2010). Lisäksi pelinkehitysprojektit ovat perinteisiä ohjelmistokehitysprojekteja paljon monimutkaisempia, sisältäen ohjelman lisäksi myös taiteellisia elementtejä. Tämän vuoksi pelinkehityksessä esiintyy sille ominaiset haasteet ja ongelmat. (Callele ym. 2005; Kanode & Haddad 2009)

Suurin syy pelinkehitysprojektien epäonnistumisille on kykenemättömyys arvioida projektin laajuus oikein. Kun projektille ei aseteta selkeää tavoitetta, projektin monimutkaisuus voi yllättää kehitystiimin. Myös teknologian kehittyminen ja asiakkaan jatkuvasti muuttuvat vaatimukset painostavat kehitystiimiä viime hetken muutoksiin aiheuttaen aikataulusta viivästymistä ja lisäkustannuksia. (Petrillo ym. 2008) Kehitysprojektiin kuuluvat monialaiset tiimit aiheuttavat kommunikaatio-ongelmia ja viivästymistä, kun joudutaan odottamaan tietyn työvaiheen loppuun saamista (Callele ym. 2005; Petrillo ym. 2008). Uuden teknologian hyödyntäminen sekä erilaisten taiteellisten elementtien yhteen saattaminen aiheuttaa myös teknisiä ongelmia. Kaiken lisäksi menestyäkseen kaupallisesti pelin tulee sisältää asiakkaalle tarjottava viihdearvo, mikä on vaikea suunnitella etukäteen. (Kanode & Haddad 2009)

Projektinhallinta peliteollisuudessa on pääpiirteittäin samanlaista kuin perinteisimmillä aloilla, mutta siihen kuuluu myös omat ominaispiirteensä, joiden avulla tuetaan luovuuden käyttöä ja johdetaan monialaisia tiimejä. Pelin eri osa-alueisiin keskittyvät erikoistuneet tiimit, jotka vastaavat muun muassa kuvituksesta, mallinnuksesta ja animaatiosta, työskentelevät pääosin itsenäisesti omissa moduuleissaan. (Cohendet & Simon 2007; Kanode & Haddad 2009) Kokonaisuutena katsottuna pelinkehitysprojekti tapahtuu Stage-Gate -mallin mukaan, jossa kaikki tiimit työskentelevät yhteisen vision ja aikataulun mukaan. Tämä mahdollistaa projektin etenemisen seuraamisen ja mahdollisiin ongelmiin puuttumisen ylemmän johdon näkökulmasta. (Cohendet & Simon 2007)

3.2 Pelinkehitysprojekti

Riippuen pelistä, pelinkehitysprojekti voi kestää muutamista kuukausista moniin vuosiin. Kehitysprojektin pituudesta huolimatta jokainen etenee läpi tiettyjen hyvin määriteltyjen vaiheiden, jotka ovat muodostuneet standardiksi kautta peliteollisuuden. Nämä vaiheet ovat konseptikehitys, esituotanto, kehitys, testaus, julkaisu sekä julkaisun jälkeiset päivitykset ja korjaukset. (Bates 2004, s. 203)

Pelinkehitys, samoin kuin mikä tahansa uuden tuotteen kehitys alkaa kuitenkin idean saamisella. Idea voi perustua esimerkiksi siihen, miten peliä pelataan, millaisia teemoja se käsittelee tai mitä uutta teknologiaa tai vaikka olemassa olevaa lisenssiä halutaan hyödyntää. Pelinkehityksessä idea on harvoin täysin alkuperäinen, vaan se perustuu usein jo vakiintuneisiin pelimekaniikkoihin ja lajityyppeihin. (Bates 2004, s. 3)

Idean saamisen jälkeen siirrytään konseptikehitykseen, jossa pienen tiimin kanssa määritellään keskeiset pelimekaniikat, kehitetään konseptikuvia siitä, miltä peli tulee näyttämään ja täsmennetään mahdollinen pelin tarina. Itsenäiset pelistudiot joutuvat usein rahoittamaan tämän vaiheen itse, koska julkaisijat haluavat nähdä etukäteen mihin rahansa sijoittavat. Tämän vuoksi tiimin on idean saamisen jälkeen päätettävä, onko projektiin sijoittaminen kannattavaa. (Bates 2004, s. 203) Stage-Gate -mallilla idean saamista seuraa ensimmäinen portti, alustava seula, jossa arvioidaan projektin toteutettavuus ja laajuus, erottuvuus kilpailijoista, kuinka hyvin projekti vastaa tiimin ydinosaamista ja käytössä olevia resursseja (Cooper 1990). Pelinkehitysprojektissa Go/Kill -päätöksistä vastaa toimikunta ihmisiä, joilla on erityinen tietämys uusien pelimekaniikkojen, konseptien ja mahdollisten brändien tunnistamiseksi (Cohendet & Simon 2007).

Jos toimikunta ylemmän johdon kanssa hyväksyy idean tutkittavaksi, muodostetaan pelisuunnittelijasta, ohjelmoijasta ja luovasta johtajasta koostuva tiimi konseptikehitystä varten (Cohendet & Simon 2007). Konseptikehityksessä pelistä laaditaan asiakirja, johon keskeisen kuvauksen lisäksi määritellään sen kohdeyleisö ja alustat, arvioidaan kehityksen aikataulu, budjetti ja tuloslaskelma sekä analysoidaan kilpailuasetelma, tiimin ja pelin vahvuudet sekä kehitykseen liittyvät riskit (Bates 2004, s. 204-207). Tyypillisen Stage-Gate -mallin ensimmäisen vaiheen eli alustavan arvioinnin tavoitteena onkin juuri selvittää tuotteen asema markkinoilla sekä arvioida kehityksen toteutettavuus,

kustannukset ja aikataulu. Toisen portin tarkoituksena on alustavalla arvioinnilla kerätyn tiedon perusteella tehdä jälleen päätös projektin jatkamisesta tai lakkauttamisesta. (Cooper 1990) Konseptiasiakirjan pohjalta arvioidaan, onko pelillä potentiaalia menestyä kaupallisesti, ja tehdään päätös seuraavan vaiheen rahoittamisesta (Bates 2004, s. 206).

Esituotanto tapahtuu ennen varsinaista kehitystä. Tässä vaiheessa tapahtuu pelisuunnittelu eli siinä yksityiskohtaisesti määritellään, mitä pelissä tapahtuu alusta loppuun asti. Tähän kuuluu muun muassa pelin tarina ja hahmot sekä miten peliä pelataan, miltä se näyttää ja miltä se kuulostaa. Tämän pohjalta suunnitellaan myös pelin taide ja tekninen toteutus. (Callele ym. 2005) Kun pelin sisältö on tiedossa, voidaan tehdä projektisuunnitelma, josta käy ilmi työtehtävät sekä niiden pituudet ja riippuvuudet. Projektisuunnitelman pohjalta jaetaan työvoima ja resurssit sekä arvioidaan jälleen budjetti ja aikataulu. Pelille asetetaan usein jopa tavoitejulkaisupäivä, jotta pelin markkinointi voidaan suunnitella etukäteen. Esituotannon lopuksi tiimi kehittää pelistä pienen osan prototyypiksi, jonka tarkoituksena on kuvastaa, miltä valmis peli tulee näyttämään ja tuntumaan. Samalla vakuutetaan julkaisijalle, että pelillä on selkeä visio, johon pyritään, ja se on toteutuskelpoinen. (Bates 2004, s. 207-212) Stage-Gate -mallin mukaisesti toisessa vaiheessa määritetään lopullinen tuote ja varmistetaan, että projekti on vielä kannattava ja toteutettavissa. Projektisuunnitelma, liiketoimintamalli ja yksityiskohtainen taloudellinen analyysi ovat tässä vaiheessa tärkeitä, sillä kolmannessa portissa tehdään niiden pohjalta päätös siirtyä kehitykseen, mikä tarkoittaa kulujen kasvua. (Cooper 1990)

Kaikista suurimman osan projektista peli on kehitysvaiheessa. Indie- ja mobiilipelit, jotka yleensä ovat pienempää kokoluokkaa, voidaan saada valmiiksi jopa kuudessa kuukaudessa. Suurempiin AAA-peleihin voi kulua jopa kaksi vuotta. Yli kaksi vuotta kehityksessä viettävät pelit kärsivät usein henkilöstön vaihtuvuudesta, ominaisuuksien muuttumisesta ja teknologian kehityksestä jälkeen jäämisestä. Tämän vuoksi Stage-Gate -mallilla keskitytään paljon kehitystä edeltäviin vaiheisiin. (Cohendet & Simon 2007) Kehitys jaetaan usein pienemmiksi tehtäviksi, jolloin sekä projektin että yksittäisten kehittäjien etenemistä on helpompi seurata ja johtaa. Näin myös julkaisija tietää, että rahoittamansa projekti pysyy aikataulussa ja millaisiin riskeihin tulevaisuudessa kannattaa varautua. Yhteistyö markkinointitiimin kanssa on oleellisessa osassa, sillä kehitysvaiheessa peliä mainostetaan ja esitellään muun muassa pelimessuilla. (Bates 2004, s. 212-213) Kehitysvaiheessa pelinkehitysprojekti eroaa perinteisestä Stage-Gate -

mallista, koska siinä tapahtuu usein vielä muutoksia lopulliseen tuotteeseen. Tässä ketterillä menetelmillä on tärkeä rooli. (Cohendet & Simon 2007) Neljännessä portissa tarkistetaan, että kehitys on toteutettu suunnitelmien mukaan laadukkaasti, tehdään tarkempi taloudellinen analyysi ja markkinointisuunnitelma sekä suunnitelmat testausta ja validointia varten (Cooper 1990).

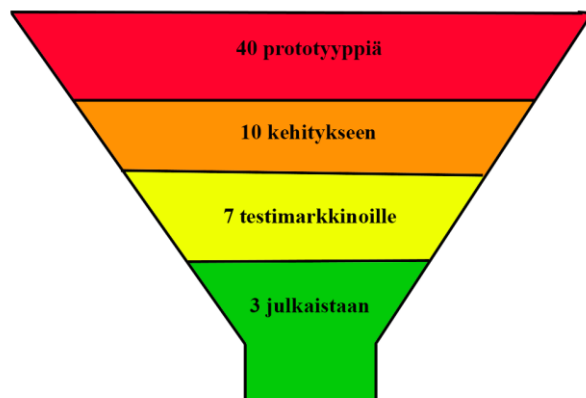
Seuraavaksi pelinkehitys siirtyy testausvaiheeseen, joka on jaettu Alpha- ja Beta-vaiheisiin. Alpha-vaiheessa peli on enimmäkseen pelattavissa alusta loppuun asti, ja pelimoottori sekä käyttöliittymä ovat valmiita. Tässä vaiheessa mukaan otetaan myös kehitystiimin ulkopuolisia testajia, joiden palautteen perusteella tehdään lopulliset päätökset siitä, mitkä ominaisuudet kannattaa sisällyttää tai jättää pois aikataulussa pysymiseksi. Beta-vaiheessa pelin lopullinen muoto on lyöty lukkoon ja jäljellä on enää vikojen etsiminen ja korjaaminen, jälleen ulkopuolisten testajien avulla. Beta-vaiheen lopuksi peli lähetetään konsolivalmistajalle tai pelien testaamiseen erikoistuneelle yritykselle, joka varmistaa, että peli täyttää julkaisualustan laatuvaatimukset. (Bates 2004, s. 214-215) Tyypillisen Stage-Gate -mallin neljäs vaihe sisältää juurikin testausta ja validointia. Tuotteen laatua ja luvattujen ominaisuuksien toimivuutta testataan sekä yrityksen sisällä että asiakaskokeiluilla. Tulosten perusteella tehdään viimeinen Go/Kill -päätös ennen tuotteen julkaisemista markkinoille. Myös toiminta- ja markkinointisuunnitelma arvioidaan ja hyväksytään toteutettavaksi Stage-Gate -mallin viimeisessä vaiheessa, joka on laaja tuotanto ja markkinoille julkaisu. (Cooper 1990)

Perinteisesti Stage-Gate -mallin mukaan uuden tuotteen kehitysprojekti lopetetaan jossain vaiheessa markkinoille julkistamisen jälkeen, ja tuotteesta tulee vakituinen osa yrityksen tuoteperhettä. Projektin lopullisia tuottoja ja kustannuksia verrataan alussa tehtyihin arvioihin, mistä nähdään, kuinka hyvin projektissa onnistuttiin, ja mitkä olivat sen vahvuudet ja heikkoudet. Näitä arvioita voidaan käyttää hyödyksi tulevia projekteja suunnitellessa. (Cooper 1990) Pelinkehityksessä projekti ei ole aivan vielä ohi julkaisun aikaan. Julkaisun jälkeen peliä pelataan useilla erilaisilla alustoilla ja tavoilla, joten testausvaiheessa huomaamatta jääneitä vikoja löydetään lähes poikkeuksetta. Internet on tehnyt mahdolliseksi julkaisun jälkeiset korjaukset ja päivitykset peliin, joilla pelinkehittäjä voi vastata kuluttajien kohtaamiin ongelmiin. (Bates 2004, s. 216)

3.3 Go/Kill -päätökset pelinkehityksessä

Peliteollisuutta on usein kutsuttu hitteihin perustuvaksi (hit-driven) liiketoiminnaksi. Tämä näkyy etenkin mobiilipelin puolella, sillä esimerkiksi vuoden 2014 kokonaistuloista jopa neljännes oli peräisin vain 10 parhaiten menestyneeltä peliltä (Activate 2015). Muun muassa suomalainen Rovio piti Applen App Storen kärkipaikkaa vuosien 2010-2012 aikana Angry Birds -pelillään, mutta ei ole myöhemmin onnistunut toistamaan menestystään samalla mittakaavalla (Guibourg 2015).

Menestyneimmät mobiilipelinkehittäjät, kuten King ja Wooga, ovat tiedostaneet peliteollisuuden hitteihin perustuvan luonteen ja rakentaneet liiketoimintansa sen pohjalta. King kehittää paljon pelejä julkaistavaksi verkkosivuilleen ja sijoittaa rahansa parhaiten menestyneisiin saadakseen ne muille alustoille. (Peterson 2012) Woogan ”hittifiltterin” (Kuvassa 4) ajatuksena on kehittää kymmeniä konsepteja ja prototyyppejä, joista lupaavimmat siirtyvät kehitykseen. Jälleen pienempi osa siirtyy testimarkkinoille, ja lopulta vain muutama julkaistaan kaikkien saataville. Säännöllisesti kehityksen aikana tehtävien arvioiden tarkoituksena on karsia projektit, joilla on vähiten potentiaalia menestyä. Tärkeää on myös oppia, mikä lopetetuissa projekteissa meni pieleen. (Wooga 2014)



Kuva 4. Woogan hittifiltteri (mukaillen Wooga 2014).

Stage-Gate -malli sopii hyvin tähän ajattelutapaan. Jokaisen vaiheen jälkeen arvioidaan, onko pelillä potentiaalia menestyä, ja tehdään Go/Kill -päätös projektin jatkamisesta. Kannattamattomaksi todetut projektit lopetetaan, ja resurssit käytetään tehokkaammin voittojen maksimoimiseksi. Vain parhaiten menestyvät tuotteet yrityksen portfolioissa pääsevät julkaisuasteelle. Cooperin mukaan Stage-Gate -malli onkin ”suppilo, ei tunneli” (vrt. Woogan hittifiltteri, Kuva 4). (Cooper 2000)

3.4 Ketterien menetelmien hyödyntäminen

Stage-Gate -mallin lisäksi myös ohjelmistokehityksen puolelta on löydetty apua pelinkehityksen ongelmiin projektin sekä nousevien kustannusten ja viivästysten hallinnassa. Ketterien menetelmien, erityisesti Scrumin ja Extreme Programming:in hyödyntämisestä on tullut varsin yleistä peliteollisuudessa. Pelinkehityksessä toimivat samat ketterien menetelmien peruseriaatteen kuin ohjelmistokehityksessäkin: yhteistyö, yksinkertaisuus, sopeutuvuus ja askel askeleelta kehittäminen. (Godoy & Barbosa 2010)

Teknologia kehittyy nopeasti, ja etenkin peliteollisuudessa siitä pyritään aina saamaan kaikki irti. Teknologian sekä kuluttajien ostokäyttäytymisen muutosten myötä julkaisijat vaativat pelinkehittäjien reagoivan näihin muutoksiin, joten esituotantovaiheessa laadittuun pelisuunnitteludokumenttiin ei voida sokeasti luottaa. Extreme Game Development, pelinkehityksen vastine Extreme Programming:lle, korostaa tiivistä yhteistyötä julkaisijan ja kehittäjän välillä. Kehittäjä toimittaa pelattavan version pelistä mahdollisimman aikaisin kehitysvaihetta ja parantaa tätä iteratiivisesti pienin säännöllisin askelin, jolloin julkaisijalle jää päätös siitä, milloin peli on julkaisukelpoinen. (Demachy 2003) On myös vaikea suunnitella etukäteen, kuinka hauskaa pelin pelaaminen tulee olemaan. Pelin kehittäjät joutuvat etsimään pelin tarjoamaa viihdearvoa useiden prototyyppien ja iteraatioiden kautta. (Kanode & Haddad 2009)

Ketterien menetelmien hyödyntäminen alkaa pelinkehityksen kehitysvaiheessa, jolloin esituotantovaiheessa pelisuunnitteludokumentti jaetaan pieniin osiin ja tehtäviin. Jokaiselle tehtävälle asetetaan omat aikarajansa sen perusteella, kuinka tärkeä ja hyödyllinen se on lopullisen pelin kannalta. (Godoy & Barbosa 2010) Samoin kuin ohjelmistokehityksessä, nämä tehtävät toteutetaan kuuden viikon sprinteissä, joiden lopuksi peli tulee olla pelattavassa muodossa. Jokaisen sprintin lopuksi arvioidaan, kuinka hyvin tavoitteissa onnistuttiin ja mitkä ovat toimivat työkalut. (Demachy 2003)

3.5 Monialaiset kehitystiimit

Pelikehitykseen kuuluu kuitenkin paljon muutakin kuin pelkästään ohjelmointia. Nykyään kehitystiimit ovat monialaisia, sisältäen muun muassa mallinnusta, animaatiota ja kuvitusta. Peli tulee kehittää yhdessä myös usein yrityksen ulkoisten tiimien kanssa, jotka vastaavat ääninäyttelystä, äänitehosteista, musiikista ja esimerkiksi lokalisoinnista.

(Bates 2004, S. 183) Pelinkehityksessä ohjelmoijien ja artistien välillä on selvä jako, jollaista ei tapahdu perinteisessä ohjelmistokehityksessä (Petrillo ym. 2008). Tämän vuoksi pelinkehityksessä ei voida suoraan soveltaa ketterää ohjelmistokehitystä, vaan monialaiset tiimit vaativat toisenlaisen lähestymistavan (Godoy & Barbosa 2010).

Pelin taiteellisen sisällön luominen tapahtuu usein tuotantolinjana, jossa omien alojensa asiantuntijat siirtävät työnsä eteenpäin seuraavalle sen suoritettuaan. Esimerkiksi mallintaja antaa työnsä animaattorille, tämä äänisuunnittelijalle ja niin edelleen. Työ siis etenee jatkuvana virtana, eikä sen odoteta olevan julkaistavassa kunnossa jokaisen sprintin jälkeen. (Godoy & Barbosa 2010) Pelin taiteellinen sisältö voidaan usein suunnitella etukäteen kehitysvaiheen alussa, joten sen aikaansaaminen sopii paremmin Stage-Gate -malliin, ja pelin ohjelmointi, jatkuva testaaminen sekä jalostaminen tapahtuvat samalla iteratiivisesti ketterien menetelmien mukaan (Bates 2004, S. 227). Artistien ja ohjelmoijien välinen yhteistyö kehitysvaiheessa on hyvin tärkeää, jotta pelin taiteelliset ominaisuudet saadaan sulavasti integroitua osaksi ohjelmaa (Kanode & Haddad 2009). Monialaisten tiimien yhteistyö Stage-Gate -mallin tiukkojen sääntöjen mukaan on myös todettu lisäävän luovuutta, jota voidaan hyödyntää tulevissa projekteissa (Cohendet & Simon 2007).

4 POHDINTA

Peliteollisuus on syntynyt tyhjästä ja kasvanut nopeasti viime vuosikymmenien aikana. Teknologian kehittymisen myötä kasvaneet odotukset pelejä kohtaan ovat tehneet pelinkehitysprojekteista yhä suurempia ja monimutkaisempia, minkä vuoksi niiden hallintaan on keksitty jatkuvasti uusia ja erilaisia menetelmiä ja malleja. Suuntaa on haettu erityisesti ohjelmistokehityksen puolelta, mutta myös perinteisimmiltä malleilta, kuten Stage-Gate.

Kirjallisuuskatsauksen perusteella on selvää, että mikään aikaisempi malli ei sovellu pelinkehitykseen sellaisenaan. Pelinkehitys on hyvin omalaatuista projektia, sillä peli sisältää ohjelman lisäksi myös taiteellisia elementtejä ja vuorovaikutusta pelaajan kanssa. Vesiputousmallia on pidetty soveltumattomana, sillä pelin lopullista sisältöä on vaikea suunnitella täysin etukäteen ja oikea pelattavuus löytyykin usein testaamalla ja iteratiivisella kehittämisellä. Toisaalta täysin iteratiivinen kehitys johtaa usein budjetista ja aikataulusta lipsumiseen. Myöskään pelin taiteellisten elementtien kehittäminen ei voi tapahtua iteratiivisesti. On alettu etsiä menetelmiä, jotka yhdistävät kehityksen tarkan rakenteen pitämisen ja kehityksen suunnitteluun panostamisen ohjelmistokehityksen ketterien menetelmien kanssa.

Pelinkehityksessä on selkeästi määritellyt vaiheet, joilla on omat tavoitteensa. Kehitystä edeltäviin vaiheisiin panostamista pidetään hyvänä tapana vähentää kehityksen aikana kohdattavia odottamattomia ongelmia. Tämän vuoksi Stage-Gate -malli on toimiva tapa johtaa pelinkehitystä kokonaisuutena katsottuna. Se antaa pelinkehitykselle selkeän rakenteen, joka helpottaa projektin etenemisen seuraamista, ja korostaa selvästi kehityksen huolellista suunnittelua etukäteen. Yhteisen ennalta laaditun suunnitelman mukaan toimiminen mahdollistaa monialaisten tiimien toimimisen yhdessä yhteisen vision ja aikataulun mukaan. Lisäksi peliteollisuuden sanotaan olevan hyvin hittivetoista, jonka vuoksi kannattamattomaksi todetut projektit kannattaa lopettaa ajoissa. Tässä asiassa Stage-Gate toimii hyvin tiukkojen Go/Kill -päätöstenä vuoksi.

Pelinkehityksen yhteydessä puhutaan Stage-Gate -mallista tai muokatusta vesiputousmallista. Vaikka ketterien menetelmien yhdistämisestä Stage-Gate -malliin on puhuttu ohjelmistokehityksen puolella jo viimeisen vuosikymmenen aikana, ensimmäiset tieteelliset artikkelit yleisesti tuotekehityksessä käytettävästä Agile-Stage-Gate -

hybridimallista on julkaistu vasta vuonna 2016, itse Stage-Gate -mallin kehittäjän Robert G. Cooperin toimesta. Hybridimalli, joka yhdistää Stage-Gaten paremman projektinhallinnan ketterien menetelmien nopean reagoinnin muutoksiin kanssa vaikuttaisi olevan juuri sitä, mitä pelinkehitysprojekti tarvitsee ongelmista ja haasteista selviytyäkseen. Jatkotutkimuksia varten onkin kiinnostava seurata, kuinka Cooperin uusi hybridimalli tulee yleistymään, ja aletaanko sen käytöstä puhua myös pelinkehityksen ohessa.

5 YHTEENVETO

Tämän kandidaatintyön tavoitteena oli selvittää, kuinka Robert G. Cooperin Stage-Gate -mallia voidaan hyödyntää pelinkehityksessä, ja kuinka se vastaa viime vuosina pelinkehityksessä kohdattuihin ongelmiin. Tutkimus toteutettiin kirjallisuuskatsauksena ja siinä käytiin läpi keskeinen teoria Stage-Gate -malliin, ketteriin menetelmiin sekä pelinkehitysohjelmaan ja sen hallintaan liittyen.

Stage-Gate -mallissa tuotekehitysprojekti on jaettu selkeästi määriteltyihin vaiheisiin, jossa jokaisella on omat tavoitteensa ja vaatimuksensa. Stage-Gate -mallilla keskitytään vaiheiden suunnitteluun ja laadukkaaseen toteutukseen, ja kannattamattomat projektit pyritään tunnistamaan ja lopettamaan mahdollisimman ajoissa. Panostamalla kehitystä edeltäviin vaiheisiin vältetään myöhemmissä vaiheissa tapahtuvat muutokset, jotka aiheuttavat suuria kustannuksia. Ohjelmistokehityksessä puolestaan asiakkaiden vaatimuksiin pyritään vastaamaan toimitushetkellä, ja ketteriä menetelmiä hyödyntämällä muutokset nähdään välttämättömyyksinä ja niihin pyritään reagoimaan mahdollisimman nopeasti. Tämä onnistuu tiimin tiiviillä yhteistyöllä, tehokkaalla kommunikaatiolla ja iteratiivisella kehittämisellä. Stage-Gate -malli ja ketterät menetelmät eivät kuitenkaan ole toisiaan poissulkevia, sillä viime vuosikymmenen aikana on kehitetty hybridimalleja, jotka yhdistävät molempien hyvät puolet. Tuotteen kehityskaari seuraa Stage-Gate -mallin rakennetta, mutta vaiheet eivät ole ennalta tarkkaan suunniteltuja, vaan ne toteutetaan nykyisiin vaatimuksiin perustuvina iteraatioina.

Pelinkehitysprojektin suurimmiksi ongelmiksi todettiin väärin arvioitu projektin laajuus, viime hetkellä tapahtuvat muutokset ja monialaisten tiimien luomat kommunikaatio- ja tekniset ongelmat. Nämä johtavat budjettien ja aikataulujen ylittymiseen. Kokonaisuutena projekti seuraa Stage-Gate -mallia yhteisen vision ja aikataulun mukaan, mikä mahdollistaa sen etenemisen ja mahdollisten kohdattujen ongelmien seuraamisen. Pelinkehitysprojekti seuraa hyvin määriteltyjä vaiheita, joilla on omat tavoitteensa ja vaatimuksensa, minkä vuoksi se sopii hyvin Stage-Gate -malliin. Tiukoilla Go/Kill -päätöksillä karsitaan kannattamattomaksi todetut projektit ajoissa, ja keskitytään peleihin, joilla on ainesta menestyä. Eniten perinteisestä Stage-Gate -mallista poiketaan kehitysvaiheessa, joka on jaettu ketterien menetelmien mukaisiin sprintteihin. Näin voidaan varmistaa, että ohjelmasta on aina toimiva versio valmiina, ja voidaan keskittyä

pelattavuuden parantamiseen ja hauskuuden löytämiseen. Pelin taiteellisten elementtien luominen ei tapahdu iteratiivisesti, minkä vuoksi Stage-Gate -mallin mukainen kehitystä edeltävä suunnittelu on niiden kannalta tärkeää. Tiimien välinen kommunikaatio ja yhteisen vision mukaan työskentely mahdollistavat, että osat saadaan lopulta sovitettua yhteen.

Pelinkehitys on hyvin omalaatuisensa projekti, minkä vuoksi olemassa olevia malleja ei olla voitu soveltaa sellaisenaan. Pelinkehityksessä toteutuvat Stage-Gate -mallin mukainen projektin tarkka rakenne, huolellinen suunnittelu ja vaiheiden toteutuksen laatuun panostaminen yhdessä ketterien menetelmien mukaisen iteratiivisen kehityksen kanssa, jolla kehitetään toimivaa ohjelmaa pienin askelin ja vastataan muuttuviin vaatimuksiin. Uusi Agile-Stage-Gate -hybridimalli on esimerkki mallista, joka yhdistää molempien hyvät puolet, ja aika näyttää kuinka se otetaan yleiseen käyttöön pelinkehityksessä kuten myös muussakin tuotekehityksessä.

LÄHDELUETTELO

Activate, 2015. Think Again: Tech and Media Outlook 2016 [verkkodokumentti].

Saatavissa: <https://www.slideshare.net/ActivateInc/activate-tech-and-media-outlook-2016> [viitattu 8.3.2017]

Ashcraft, B., 2010. Duke Nukem Forever Designer Apparently Lost Millions

[verkkouutinen]. Saatavissa: <http://kotaku.com/5657935/duke-nukem-forever-designer-apparently-lost-millions> [viitattu 1.2.2017]

Bates, B., 2004. Game Design. 2. painos. Boston: Thomson Course Technology PTR, 350 s., ISBN 1-59200-493-8

Behrmann, M., Noyons, M., Johnstone, B., MacQueen, D., Robertson, E., Palm, T. & Point, J., 2012. State of the Art of the European Mobile Games Industry

[verkkodokumentti]. Saatavissa: http://www.mobilegamearch.eu/wp-content/uploads/2012/12/Mobile-Game-Arch_D3.1_06122012_PU.pdf [viitattu 27.1.2017]

Bisset, L., 2013. The App Store's most downloaded games of all time: Angry Birds, Fruit Ninja and Temple Run [verkkouutinen]. Saatavissa:

<http://www.pocketgamer.biz/news/50604/the-app-stores-most-downloaded-games-of-all-time-angry-birds-fruit-ninja-and-temple-run/> [viitattu 27.1.2017]

Callele, D., Neufeld, E. & Schneider, K., 2005. Requirements Engineering and the Creative Process in the Video Game Industry. Teoksessa: Proceedings of the 2005 13th IEEE International Conference on Requirements Engineering (RE'05). Saskatoon, Saskatchewan, Canada: IEEE. ISBN 0-7695-2425-7

Cohendet, P. & Simon, L., 2007. Playing across the playground: paradoxes of knowledge creation in the videogame firm. Journal of Organizational Behavior, 28 (5), S. 587–605.

Cook, D., 2007. Rockets, Cars and Gardens: Visualizing waterfall, agile and stage gate [verkkodokumentti] Saatavissa: <http://www.lostgarden.com/2007/02/rockets-cars-and-gardens-visualizing.html> [viitattu 8.2.2017]

Cooper, R. G., 1990. Stage-gate systems: A new tool for managing new products. *Business horizons*, 33 (3), S. 44-54.

Cooper, R. G., 2000. Winning with New Products: Doing It Right. *Ivey Business Journal*, 64 (6), S. 54-60.

Cooper, R. G., 2016. Agile-Stage-Gate Hybrids. *Research-Technology Management*, 59 (1), S. 21-29.

Crawley, D., 2015. Sega CEO admits a betrayal of trust; studio now aiming to build high-quality console games again [verkkouutinen] Saatavissa: <http://venturebeat.com/2015/07/07/sega-ceo-admits-a-betrayal-of-trust-aiming-to-build-high-quality-console-games-again/> [viitattu 1.2.2017]

Demachy, T., 2003. Extreme Game Development: Right on Time, Every Time [artikkeli] Saatavissa: <http://www.gamasutra.com/view/feature/131236/> [viitattu 15.3.2017]

Downin, J., 2011. Minecraft beta cracks 4 million [verkkouutinen] Saatavissa: <https://www.gamespot.com/articles/minecraft-beta-cracks-4-million/1100-6344310/> [viitattu 29.3.2017]

Fuller, S. H. & Millett L. I., 2011. *The Future of Computing Performance: Game Over or Next Level?*. Washington, D.C.: The National Academies Press, 185 s. ISBN 978-0-309-15951-7

Gartner, 2007. Gartner Says Worldwide Mobile Gaming Revenue to Grow 50 Percent in 2007 [lehdistötiedote]. Saatavissa: <http://www.gartner.com/newsroom/id/507467> [viitattu 26.1.2017]

Godoy, A. & Barbosa, E. F., 2010. Game-Scrum: An Approach to Agile Game Development. *SBC - Proceedings of SBGames 2010*, S. 292-295

Goldfarb, A., 2017. Microsoft confirms Scalebound is cancelled [verkkouutinen]. Saatavissa: <http://www.ign.com/articles/2017/01/09/microsoft-confirms-scalebound-is-cancelled> [viitattu 1.2.2017]

- Guibourg, C., 2015. Angry Birds maker Rovio axes 260 jobs after failing to replicate smartphone game success [verkkouutinen]. Saatavissa: <http://www.cityam.com/223116/angry-birds-maker-rovio-axes-260-jobs-after-failing-replicate-smartphone-game-success> [viitattu 29.3.2017]
- Highsmith, J. & Cockburn, A., 2001. Agile Software Development: The Business of Innovation. *Computer*, 34 (9), S. 120-122.
- Kanode, C. M. & Haddad, H. M., 2009. Software Engineering Challenges in Game Development. Teoksessa: 2009 Sixth International Conference on Information Technology: New Generations. Kennesaw, Georgia: IEEE. S. 260-265. ISBN 978-0-7695-3596-8
- Karlström, D. & Runeson, P., 2005. Combining Agile methods with stage-gate project management. *IEEE Software*, 22 (2), S. 43-49.
- McGuire, M. & Jenkins, O. C., 2009. *Creating Games: Mechanics, Content, and Technology*. Wellesley, Massachusetts: CRC Press, 526 s. ISBN 978-1-4398-6592-7
- Nunneley, S., 2010. Report: Average game development budget sits between \$18-\$28M. [verkkouutinen] Saatavissa: <https://www.vg247.com/2010/01/12/report-average-game-development-budget-sits-between-18-28m/> [viitattu 31.1.2017]
- Petrillo, F., Pimenta, M., Trindade, F. & Dietrich, C., 2008. Houston, we have a problem...: A Survey of Actual Problems in Computer Games Development. Teoksessa: Proceedings of the 2008 ACM Symposium on Applied Computing (SAC). Fortaleza, Ceara, Brazil: ACM, S. 707-711. ISBN 978-1-59593-753-7
- Peterson, S., 2012. King.com CEO: "Hit-proofing" a hit-driven business [verkkoartikkeli] Saatavissa: <http://www.gamesindustry.biz/articles/2012-11-05-king-com-ceo-hit-proofing-a-hit-driven-business> [viitattu 9.3.2017]
- SuperData Research, 2016. Market Brief — Year in Review 2016 [verkkodokumentti]. Saatavissa: <https://www.superdataresearch.com/market-data/market-brief-year-in-review/> [viitattu 26.1.2017]

Vedsmand, T., Kielgast, S. & Cooper, R. G., 2016. Integrating Agile with Stage-Gate® – How New Agile-Scrum Methods Lead to Faster and Better Innovation

[verkkojulkaisu] Saatavissa:

<http://www.innovationmanagement.se/2016/08/09/integrating-agile-with-stage-gate/>

[viitattu 16.2.2017]

Wolf, M. J. P., 2008. The Video Game Explosion: A History from PONG to Playstation and Beyond. Westport, Connecticut: Greenwood Press, 380 s. ISBN 978-0-313-33868-7

Wooga, 2014. Critical Hits and near Misses: The Wooga Hit Filter [verkkojulkaisu]

Saatavissa: [http://blog.wooga.com/post/133339025518/critical-hits-and-near-misses-](http://blog.wooga.com/post/133339025518/critical-hits-and-near-misses-the-wooga-hit)

[the-wooga-hit](http://blog.wooga.com/post/133339025518/critical-hits-and-near-misses-the-wooga-hit) [viitattu 2.2.2017]