

Topi-Matti Sillantaus

PELIALAN & PELINTEKOVÄLINEIDEN SELVITYS

Opinnäytetyö
Tietojenkäsittelyn ko.

Marraskuu 2013




MIKKELIN AMMATTIKORKEAKOULU

Mikkeli University of Applied Sciences

KUVAILULEHTI

 <p>MIKKELIN AMMATTIKORKEAKOULU Mikkeli University of Applied Sciences</p>		Opinnäytetyön päivämäärä 28.11.2013	
Tekijä(t) Topi-Matti Sillantaus		Koulutusohjelma ja suuntautuminen Tietojenkäsittely	
Nimeke Pelialan & pelintekovälineiden selvitys			
Tiivistelmä <p>Tämän opinnäytetyön aiheena oli pelialan katsaus ja erityisesti Unity3D-ohjelma. Kävin ensiksi läpi pelialaa yleisesti kertoen sen historiasta ja mielestäni sen tärkeimmistä asioista. Seuraavaksi otin esille muutamia pelienteko-ohjelmistoja, minkä jälkeen siirryin käymään tarkemmin läpi Unityä. Unity3D-osuudessa kävin läpi Unityn historiaa ja sen perustoimintoja. Näiden aiheiden jälkeen oli vuorossa pieni tutkimukseni Unityn käytöstä Suomessa. Olin tehnyt tutkimusta siitä, kuinka laajaa Unityn käyttö oli suomalaisissa pelifirmoissa. Tutkimusteni pohjalta esitin tietoja siitä, kuinka keskeinen Unity Suomalaisessa pelimaailmassa tämän tutkimuksen aikana oli.</p> <p>Näiden asioiden jälkeen vuorossa oli oma projektini, jossa olin tehnyt Unityllä pienimuotoisen kokeellisen demonstraation. Tein aluksi Mikkelin ammattikorkeakoululla sijaitsevan CampusIT-yrityksen tiloista 3D-mallinnuksen Google SketchUp -ohjelmalla. Minulla oli myös eräältä opettajaltani saatu 3D-malli koko Mikpolista, jota käytin myös demonstraatiossani. Demonstraation tarkoituksena oli tuoda Mallinnukset Unity3D-ohjelmaan, jossa sitten pystyisi liikkumaan mallien välillä vapaasti. Kerroin projektin eri vaiheista ja prosessin kulusta mahdollisimman tarkasti.</p> <p>Päätännössä tein yhteenvedon opinnäytetyöstäni ja sen eri asioiden merkityksistä. Kerroin opinnäytetyöni kulusta, ongelmatilanteista, missä onnistuin ja mitä olisin voinut tehdä toisin. Päätäntöön sisältyi myös tekemäni demonstraation analysointia esimerkiksi siitä, voisiko projektia jatkaa tulevaisuudessa eteenpäin.</p>			
Asiasanat (avainsanat) peliala, 3D, Unity3D, Google SketchUp			
Sivumäärä 42	Kieli Suomi	URN	
Huomautus (huomautukset liitteistä)			
Ohjaavan opettajan nimi Jukka Selin		Opinnäytetyön toimeksiantaja Mikkelin Ammattikorkeakoulu	

DESCRIPTION

 <p>MIKKELIN AMMATTIKORKEAKOULU Mikkeli University of Applied Sciences</p>		Date of the master's thesis 28 of November 2013	
Author(s) Topi-Matti Sillantaus		Degree programme and option Business information technology	
Name of the master's thesis Report of the game industry and game development programs			
Abstract <p>The subject of this bachelor's thesis was an overview of the game industry and game development tools, especially the aspects of Unity3D program. At first, I went through the game industry in general by telling about its history and the most important aspects in my opinion. Next, I introduced a couple of different game development programs, and after that moved on to more detailed information concerning Unity3D. The Unity3D part went through its history and some of Unity's basic functions. I studied how widespread the use of Unity3D was in Finnish game development companies. Based on this research I presented information on the importance of Unity3D in the Finnish game development field at the time of this study.</p> <p>The practical part of this study was my own small scale experimental demonstration which I made with Unity3D. At first, I did a 3D-model of CampusIT located at Mikkeli University of applied sciences. I did the model with the Google sketchup program. Also, one of my teachers gave me a 3D-model of the entire Mikkeli building of Mikkeli University of applied sciences which I also used in this demonstration. The purpose of this demonstration was to bring the 3D-models to Unity3D where the models could be moved freely. The different stages of the project and the flow of the process were discussed as closely as possible.</p> <p>The study ended with a summary and discussion of the meanings of its relevant aspects. It described my progress, problematic situations, what I was able to do and what I could have done differently. There was also an analysis of my demonstration and how the project could be continued in the future.</p>			
Subject headings, (keywords) Game Industry, 3D, Unity3D, Google SketchUp			
Pages 42	Language Finnish	URN	
Remarks, notes on appendices			
Tutor Jukka Selin		Master's thesis assigned by Mikkeli University of applied sciences	

SISÄLTÖ

1	JOHDANTO	1
2	PELIALASTA YLEISESTI	2
2.1	Ensiaskleet	2
2.2	Nintendon aikakausi	4
2.3	3D-pelien historiasta	7
3	MUUTAMISTA PELINTEKOVÄLINEISTÄ.....	10
3.1	XNA	10
3.2	Game Maker & Game Maker: Studio	11
3.3	Unreal Engine & Unreal Development Kit	13
4	UNITY 3D	15
4.1	Taustatietoa Unity3D:stä	15
4.2	Käyttöliittymästä ja sen ominaisuuksista	16
4.2.1	Scene ja Hierarchy -näkymät	18
4.2.2	Inspector-näkymä	19
4.2.3	Project-näkymä ja Assets	20
4.2.4	Game Window	22
5	TUTKIMUS UNITYN KÄYTTÄMISESTÄ SUOMESSA	23
5.1	Kyselyn taustoja.....	23
5.2	Analysointia	25
5.3	Yhteenvedo tuloksista	28
6	OMA CASE: UNITY DEMONSTRAATIO.....	29
6.1	Demon idea ja tarkoitus.....	29
6.2	Mallien tuominen Unityyn ja projektin kulku	32
7	PÄÄTÄNTÖ.....	40
	LÄHTEET	43

LIITE/LIITTEET

- 1 Yksisivuinen liite
- 2 Monisivuinen liite

1 JOHDANTO

Opinnäytetyöni aiheena on pelialan katsaus yleisesti ja oman demonstraation tekeminen Unity3D:llä. Tarkoituksena on tuoda Unityyn toisella ohjelmalla tehty 3D-malli ja rakentaa sitä Unityyn pienimuotoinen itse tehty demonstraatio. Opinnäytetyön tilaajana on Mikkelin ammattikorkeakoulu ja ohjaavana opettamani toimii Jukka Selin.

Tämän opinnäytetyön tekemiseen sain ensimmäisen kipinän ollessani työharjoittelussa Mikkelin ammattikorkeakoululla sijaitsevassa CampusIT:ssä. Olin jo pitkään puntaroinut ja miettinyt sopivaa aihetta opinnäytetyölleni. CampusIT:llä pääsin ohjelmoimaan melko paljon ja tutustuin myös 3D-mallinnukseen entistä syvällisemmin. Harjoittelun loppuvaiheessa tein 3D-mallia CampusIT:n toimitiloista Google Sketchup -ohjelmalla. Tätä kautta kiinnostuin myös muista 3D-ohjelmista kuten Unity3D:stä. Useiden opettajieni kanssa käytyjen keskustelujen ja palaverien jälkeen saimmekin räätälöityä itselleni sopivan ja mielenkiintoisen aiheen.

Toisessa luvussa kerron aluksi pelialasta yleisesti ja sen eri osa-alueista. Kerron pelituotannon historiasta aloittaen aivan juuritasolta, minkä jälkeen kertaan hieman 80-luvun pelimaailmaa Arcaden ja Nintendon muodossa, ja kolmantena otan esiin hieman 3D-pelien ensitahteja.

Kolmannessa luvussa käyn läpi muutamia erityyppisiä ohjelmistoja pelituotannon saralla. Tarkoitukseni on selventää ja pohjustaa hieman minkätyyppisillä ohjelmistoilla aivan tavallinen peleistä ja ohjelmoinnista kiinnostunut henkilö kykenee tekemään omia projektejaan suurten pelien tuottajien ulkopuolelta.

Neljännessä luvussa otan esille historiaa ja perustoimintoja Unity3D-ohjelmasta. Pyrin kertomaan Unitystä tiiviissä paketissa oleellista tietoa lähteiden pohjalta. Aloitan kertomisen Unityn perustamisajoista lähtien siirtyen sen jälkeen Unityn käyttöliittymään. Kerron Unitystä hieman syvällisemmin, kuin aiemmassa osuudessa läpikäymistäni ohjelmista. Tarkoitus ei ole kuitenkaan tässä osuudessa mennä Unityn perustoimintoja syvemmälle.

Viidennessä osuudessa tulen selventämään omien kesällä tekemiäni pienimuotoisten tutkimusten ja kyselyiden pohjalta, kuinka laajaa Unityn käyttö tällä hetkellä

Suomessa on. Etsin ja kyselin tietoja enimmäkseen johtavilta suomalaisilta pelifirmoilta, järjestöiltä sekä itse Unityltä itseltään. Tutkimus ei ole kovinkaan massiivinen, sillä arvatenkin pelifirmoilta saamani tieto oli usein vähäistä. Kuitenkin pyrin parhaani mukaan antamaan pienimuotoisen selvityksen saamieni tietojen pohjalta.

Kuudennessa luvussa on vuorossa oma projektini eli CASE. Tarkoituksena on luoda 3D-malli CampusIT:n toimitiloista eri ohjelmalla kuin Unity, ja sen jälkeen tuoda ohjelma Unityyn pyöriteltäväksi. Tein alkuperäisen mallini Google Sketchupilla ja tämän lisäksi sain myös opettajaltani Esa Hannukselta koko Mikpoli-rakennuksesta kattavan Sketchup-mallinnuksen. Demonstraation tarkoituksena siis on saada nämä mallit Unityyn ja tehdä sellainen demo, jossa voit kulkea Mikpolin ulko-ovelta CampusIT:n huoneeseen seikkailemaan. Kyseessä ei siis ole peli, vaan havainnollistava demonstraatio Mikpolista ja CampusIT:stä.

Viimeisessä eli seitsemännessä luvussa niputan koko opinnäytetyöni yhteen. Kerron omien johtopäätöksieni ja kokemusteni pohjalta projektin kulusta, haasteista sekä mitä olisin voinut kenties tehdä paremmin. Annan omia johtopäätöksiäni aiheesta, tutkimuksestani ja demonstraatiostani.

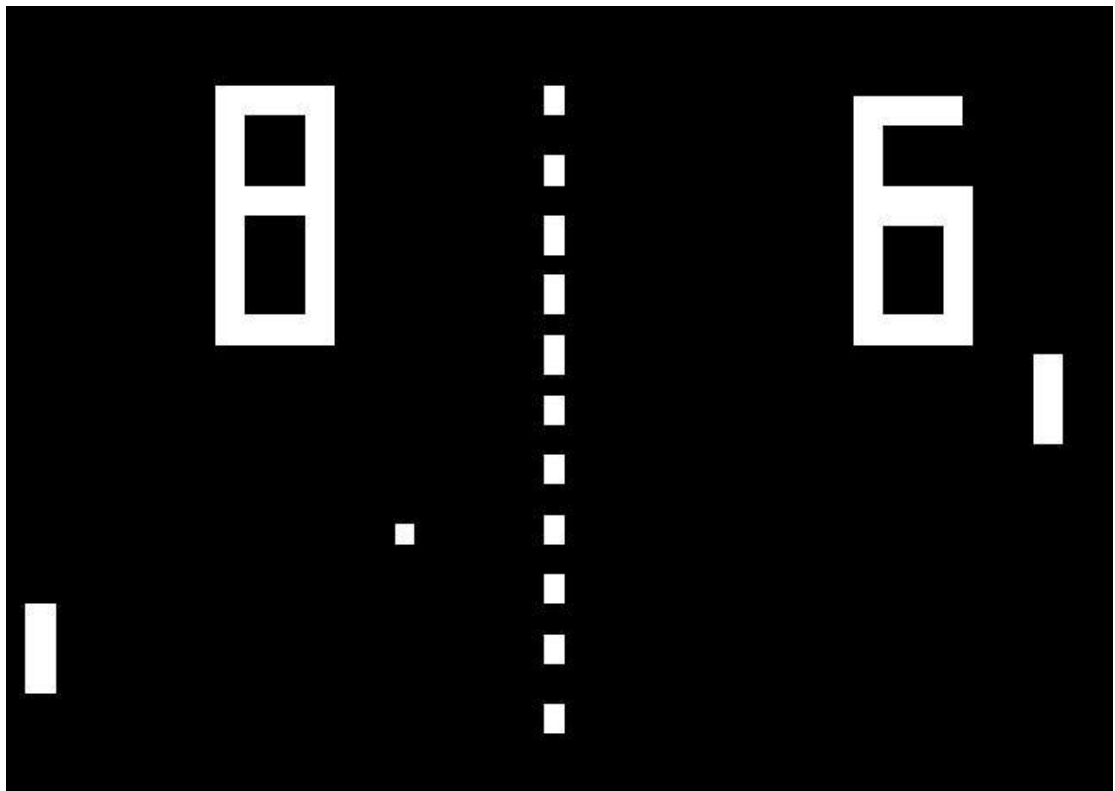
2 PELIALASTA YLEISESTI

Tässä luvussa kerron pelialan historiasta yleisesti. Käyn läpi pelialan menneisyyttä muutamien valitsemani esimerkein. Ensin käyn läpi aivan alkuaikoija, minkä jälkeen siirryn 1980-luvulle ja Nintendon valtakaudelle. Viimeiseksi historiikiksi valitsin 3D-pelien alkuajat ja mielestäni merkittävimmät läpimurtopelit 1990-luvun ajalta.

2.1 Ensiaskeleet

Peliohjelmointi käsitteenä on käynyt läpi hurjia muutoksia sen alkuajoista tähän päivään. Varmasti jokaiselle videopelit ovatkin ainakin jollakin tapaa tuttu käsite ja enemmistö väestöstä videopelejä on joskus pelannutkin. Mutta mitä pelituotanto onkaan ja mitä sen historiaan kuuluu?

Ensimmäiset tunnetut videopelit syntyivät lähinnä erilaisista kokeiluista ja tekijöidensä omaksi iloksi. Vuonna 1958 fyysikko nimeltä Willy Higinbotham keksi yhden tunnetuimmista ja kloonatuimmista peleistä/pelityypeistä. Hän oli töissä Brookhavenin kansainvälisen laboratorion voimalaitoksessa. Vuodesta toiseen henkilökunta ja tarkastajat kävivät tarkkailemassa hänen työnjälkeään ja sen turvallisuutta. Higinbotham huomasi, kuinka tylsistyneitä ihmiset olivat ja päätti tehdä asialle jotakin. Hän käytti oskilloskooppia (elektroniiikan ja sähkötekniikan mittalaite, joka piirtää mitattavan signaalin näytölle) ja yksinkertaista analogista tietokonetta luodakseen pelin, jossa valopallo liikkui ruudulla kahden palkin välissä. Tarkoituksena oli saada pallo pysymään pelialueella osumalla siihen palkeilla. Tätä ideaa on kopioitu lukemattimia kertoja peliyhtiöiden toimesta mm. yksinkertaisissa peleissä kuten Atarin vuoden 1972 pelissä *Pong*. (The history of video game programming 2001.)



KUVA 1. PONG (Atari 1972)

Pelissä *Pong* (kuva 1) on nähtävissä Higinbothamin kehittämä idea, ja se miten Atari vuonna 1972 sitä hyödynsi. *Pong*-pelistä ja Higinbothamin kehittämästä pelityypistä tuli yksi kaikkien aikojen tunnetuimmista ja kopioinnuimmista.

2.2 Nintendon aikakausi

Ensiksi esille ottamani Atarin tekemän Pongin ja Nintendon tulemisen välissä ehti tapahtua joitakin merkittäviä ja vähemmän merkittäviä asioita pelituotannon saralla. Tämän osion ollessa kuitenkin eräänlaista pohjustusta, siirryn seuraavaksi suoraan Nintendon maailmaan.

Nintendon ensiaskeleet sijoittuvat niinkin kauas kuin vuoteen 1889. Silloin Nintendo playing card Co.Ltd. niminen yhtiö valmisti ns. Hanafuda pelikortteja. Myöhemmin vuonna 1953 yhtiöstä tuli kyseisten muovisten pelikorttien ensimmäinen massatuottaja. Vuonna 1963 yhtiö vaihtoi nimekseen Nintendo Co. ja rupesi tuottamaan videopelejä pelikorttien lisäksi. Uusien teknologioiden ja mahdollisuuksien saattelemana Nintendo keskittyi entistä enemmän pääsääntöisesti videopelien tuottamiseen ja kehittelyyn. Nintendon ensimmäinen varsinainen hittituote *Donkey Kong* (kuva 2) julkaistiin vuonna 1981 kolikkopelinä. Pelin ideana oli pelastaa Mario nimisen henkilön avulla hänen Pauline niminen tyttöystävänsä Donkey Kong nimisen gorillan kynsistä. Marion piti saada Pauline turvaan Donkey Kongilta ennen kuin peliaika loppuu. Donkey Kongista tuli sen ajan suosituin kolikkopeli ympäri maailmaa, ja se on edelleenkin ikonisessa asemassa useiden uusien versioiden ja jatko-osien saattelemana. (History of Nintendo Worldwide 2013.)



KUVA 2. Donkey Kong (Nintendo 1981)

Nintendo Co. oli vihdoinkin pääsemässä kunnolla vauhtiin ja vuonna 1983 julkaistiin Japanissa Famicom (Nintendo Family Computer), jonka suunnittelusta vastasi henkilö nimeltä Masayuki Uemura. Famicomille (kuva 3) tehtiin aluksi käännökset mm. edellämainitusta *Donkey Kong*-pelistä sekä *Popeye*-pelistä. Konsolin suosio ei lähtenyt kuitenkaan räjähdysmäiseen nousuun, mutta parin vuoden sisään se nousi Japanin suosituimmaksi konsoliksi.



KUVA 3. Vuoden 1983 Famicom ja vuoden 1986 Nintendo Entertainment System (NES)

Vuonna 1984 Nintendo teki mm. pelejä kuten *Metroid* to *The Legend of Zelda*, *Punch out* ja *Super Mario Brosin*. Kaikki pelit olivat suosittuja ja nykyään legendaarisessa asemassa, mutta *Super Mario Bros* oli se peli joka viimeistään räjäytti potin ja avasi Nintendolle ovet maailmanmarkkinoille (Nintendo (kuva 3) julkaistiin uudella ulkoasulla kansainvälisesti vuonna 1986). Pelissä seikkaili Mario niminen putkimies, jonka tehtävänä oli pelastaa prinsessa ilkeältä Koopa nimiseltä hahmolta ja hänen kätyreiltään erilaisissa tasoissa ja maailmoissa. Pelissä pystyi myös seikkailemaan Marion veljellä nimeltä Luigi. Tutkimukset ovat myöhemmin näyttäneet, että lapset ovat samaistuneet enemmän Marioon kuin esimerkiksi Mikki hiireen. Mario tavattiin jo ensimmäisen kerran aiemmin mainituissa *Donkey Kong* -peleissä. (History of Nintendo Worldwide 2013.)

Alkuperäinen *Super Mario Bros* sai kaksi jatko-osaa, joista viimeisin on varmasti tunnetuin. Useissa pelien ranking-listoissa Super Mariot ovat usein hyvin korkealla, ja eritoten *Super Mario Bros 3:sta* (kuva 4) pidetään yleisesti yhtenä kaikkien aikojen parhaimmista ja koukuttavimmista peleistä. Super Mariosta on tehty enemmän uusia pelejä ja versiointeja, kuin mistään muusta pelistä ja lähes kaikille tunnetuille pelikonsoleille.



KUVA 4. Super Mario Bros 3 (Nintendo 1988)

Ensimmäinen Super Mario -peli julkaistiin 1985 ja siitä tuli ajaton klassikko. Kakkososa julkaistiin Japanissa 1986 ja muualla maailmassa 1988. Se erosi pelattavuudeltaan ensimmäisestä ja kolmannesta osasta, ja onkin usein jäänyt niiden varjoon suosiossa. Pelisarjan viimeinen osa julkaistiin vuonna 1988, ja se menestyi todella hyvin ympäri maailmaa. Pelattavaa oli kolmannessa osassa entistä enemmän ja se oli aiempia monipuolisempi. Omasta mielestäni pelisarjan paras peli.

Alkuperäiselle Nintendolle julkaistiin vuosien 1985–1994 aikana reilusti yli sata peliä. Donkey Kongin, Super Marioiden, Zeldan yms. pelien ansioista Nintendolle aukesivat mahdollisuudet tuotannon jatkumoon ja uudistumiseen uusien ohjelmistojen kuten Gameboy, Super Nintendon, Nintendo64:n ja Nintendo Wii:n osalta. Nintendo on

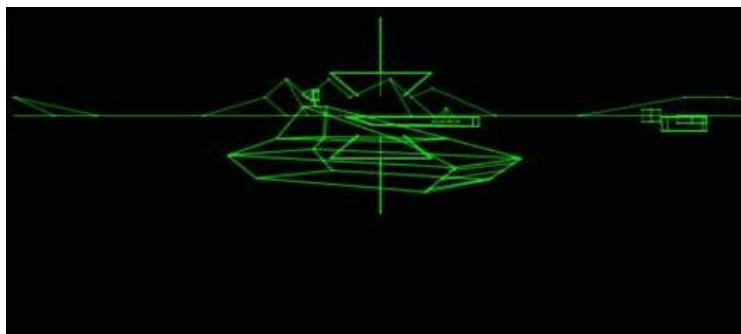
yhtiönä ja brändinä kulkenut pitkän matkan alkuajoista nykypäivään ja se on yksi tunnetuimmista videopelimaailman nimikkeistä.

2.3 3D-pelien historiasta

Opinnäytetyöni myöhemmissä vaiheissa keskitytään suurimmaksi osaksi 3D - peliohjelmointiin menetelmien, tutkimusten ja omien kokeilujen kautta. Arcade - kolikkopelien, Nintendon yms. jälkeen 3D-pelit tekivät vahvaa tulemistaan. Onkin siis hyvä palauttaa mieleen milloin 3D-pelit ja niiden ohjelmointi varsinaisesti alkoikaan.

3D-pelit tulivat varsinaisesti suuren maailman tietoon 1990-luvun alkupuolella, mutta 3D:n lähtölaukaukset oli jo ammuttu paljon aikaisemmin. 1970-luvun lopulla ja 1980-luvun alkupuolella oli jo havaittavissa yksinkertaisia kokeiluja 3D-pelien maailmaan. Tuolloin pidettiin yleisesti hienona asiana, jos yleensä sai tuotettua kolmiulotteista kuvaa. Nykyajan peleihin nämä alkuaikojen pelit ja kokeilut näyttävät usein lähes huvittavilta. Yksi ensimmäisistä 3D-grafiikkaa hyväksikäyttävistä ns. kaupallisista menestyksistä oli Atarin vuonna 1980 julkaisema Arcade-peli nimeltä *Battlezone* (kuva 5). Peli käytti hyväkseen vektorigrafiikkaa rasterigrafiikan sijaan. Peli oli hyvin yksinkertaisen näköinen pelkistettyine vektorigrafiikoin piirretyin maisemin ja komponentteineen. Pelin ideana oli tuhota pelaajan ohjaamasta panssarivaunusta vihollisen panssarivaunuja jne. (The evolution of 3D games 2010.)

Nyt yli 30-vuotta pelin ilmestymisen jälkeen voidaan rehellisesti Battlezonea pitää hyvänä saavutuksena sen aikaisien mahdollisuuksien puitteissa. Battlezonea pidettiin sen julkaisemisen aikoihin niin realistisena, että Yhdysvaltain puolustusjoukot käyttivät Battlezonea harjoittaakseen omia panssarivaunujoukkojaan. (The evolution of 3D games 2010.)



KUVA 5. Battlezone (Atari 1980)

1980-luku piti sisällään myös mm. erilaisia 3D-autoilupelejä, jossa oli ns. jahtaava kamerakuvakulma. 3D-pelit kuitenkin löivät varsinaisesti läpi 1990-luvun alkupuolella. Vuonna 1992 amerikkalainen tietokonepeliyritys ID-software julkaisi pelin nimeltä *Wolfenstein3D* (kuva 6). Peli oli yksi ensimmäisistä ns. First person shooter peleistä, joissa pelaajan näkökulmasta liikuttiin erilaisissa tasoissa tuhoamalla vastustajia ja etsimällä reittejä tasojen läpi pääsemiseksi. (History of the first 3D video games 2013.) Peli herätti paljon huomiota ollessaan aikakauteensa nähden varsin onnistunut tapaus. Se myikin kansainvälisesti vähintäänkin kohtalaisesti ja se on yksi First person shooter pelien pioneereista. Myöhemmin *Wolfenstein3D* valittiin Computer Gaming Worldin Hall of Fameen, koska sen katsottiin muokanneen ratkaisevasti peliteollisuuden maailmaa ja sen se myös toden totta teki. (id history 2013.)



KUVA 6. Wolfenstein3D (ID-software 1992)

Tämä oli ID-softwarelle ja First person shooter genrelle kuitenkin vasta alkua. Wolfenstein laukaisi First person shooter pelien aallon, josta seurasi toinen toistaan parempia tekeleitä. ID-software etunenässä oli tämän genren pelien yksi onnistujista. Vuonna 1994 yhtiö julkaisi sen ensimmäisen todellisen maailmanlaajuisen massahitin nimeltä *Doom* (kuva 7). *Doom* oli ehdottomasti sen ajan visuaalisesti kenties hienoin ja sykähdyttävin taidonnäyte pelien tekemisestä ja niiden ohjelmoinnista. *Doom* oli brutaali, tunnelmaltaan häijy, musiikeiltaan loistava ja muutenkin sykähdyttävä peli. Doomilla on todella vakiintunut fanattinen kannattajakuntansa, joka edelleen vannoo alkuperäisten *Doom*-pelien nimeen. *Doom* myi maailmanlaajuisesti miljoonia

kappaleita, ja se on yksi kaikkien aikojen menestyneimmistä tietokonepeleistä. (id history 2013.)

Peli myös muutti käsitystä nettipelaamisesta. ID-software antoi faneilleen mahdollisuuden kamppailla internetissä toisiaan ja pelin hirviöitä vastaan neljän pelaajan LAN-peleissä tai yksi vastaan yksi kamppailuissa. Yhtiön ja pelin innovaattorit julkaisivat netissä myös Doomien lähdekoodin, joka antoi pelaajilleen ja faneilleen mahdollisuuden rakentaa ja tehdä omia tasojaan tai muunnoksia Doomiin. Tämä lisäsi entisestään pelien mielenkiintoa ja suosiota. *Doom* on julkaistu suurimmalle osalle tunnetuista pelikonsoleista. Alkuperäisen Doomien pelimoottorille julkaistiin vuonna 1994 jatkumo nimeltä *Doom 2: Hell on Earth*. Peli sai myös lukuisia lisäosia ja *Doom 3* julkaistiin pitkän tauon jälkeen vuonna 2004. Kolmannessa osassa oli sykehdyttävät grafiikat ja se vaatikin ilmestyessään tietokoneelta paljon tehoja sen pyörittämiseen. Wolfensteinin ja Doomien jälkeen muita tunnettuja 1990-luvulla julkaistuja legendaarisia first person shooter pelejä olivat mm. *Quake* jatko-osineen ja *Duke Nukem*. (Id history 2013.)



KUVA 7. Doom (ID-software 1994)

Peliohjelmointi ja pelituotanto on kehittynyt edellä mainituista ajoista edelleen räjähdysmäisesti. Tässä osuudessa annetut esimerkit olivat ns. pintaraapaisua koko pelituotannon historiaan. Mielestäni nämä merkkipaalat olivat tärkeitä ja siksi ne valitsin. Nykypäivänä pelit ovat parhaimmillaan uskomattoman kauniita ja realistisia. Nykyään lähes kaikilla suurimmilla pelifirmoilla/taloilla on omat pelimoottorit ja välineet peliensä toteuttamiseen. Tämä mahdollistaa upeiden, näyttävien ja yksilöllisten pelien tuotannon. Tällä hetkellä on myös olemassa sellaisia pelintekovälineitä, joilla tavalliset peliohjelmoinnista sekä peleistä kiinnostuneet

ihmiset voivat itse tehdä omia pelejään. Kyseisillä ohjelmilla tehdyt pelit kuitenkin harvoin ovat verrattavissa pelitalojen näyttävimpiin tekeleisiin. Seuraavassa osuudessa siirrytäänkin tutkailemaan näitä ohjelmistoja, jotka mahdollistavat pelien tuotannon kaikille aiheesta kiinnostuneille.

3 MUUTAMISTA PELINTEKOVÄLINEISTÄ

Olemme käyneet läpi pääpiirteisen katsauksen muutamista merkittävimmistä asioista pelimaailman historiasta, joten on aika käydä läpi millaisilla välineillä pelifirmojen ulkopuolle pelejä usein tehdään. Osuudessa otan esille sellaisia pelintekovälineitä, joilla kaikki halukkaat pelifirmojen lisäksi voivat osatessaan tehdä mieleisiään pelejä. Osiossa keskitytään muutamaa pelintekovälineeseen ns. pienpelien tuotannossa.

3.1 XNA

Ensimmäisenä otamme esille XNA:n, joka on Microsoftin vuonna 2006 julkaisema pelinkehitysympäristö. Se pohjautuu pääsääntöisesti Microsoftin Visual Studioon ja .NET-runkoon. XNA oli yksi aikaisimmista asioista, jotka liitettiin Microsoftin Xbox 360 -pelikonsoliin. Sen pääsääntöinen tavoite oli pienentää koko ajan kasvavia tuotantokuluja suoraviivaistamalla Windows -ja Xbox pelien tekemistä. XNA antoi myös mahdollisuuden ns. Indie-pelien tekemiseen harrastajien keskuudessa. (What is the xna framework 2006.)

Tiedämme jo että pelien tekeminen on varsin haastavaa puuhaa, ja tähän XNA tarjoaa helpotusta pelintekijöille. XNA:n ero perinteiseen peliohjelmointiin on sen suhteellisen helppokäyttöinen työympäristö, ja että sen kehitysympäristö on räätälöity eri käyttötarkoitusten mukaan. XNA:ssa voi suoraan keskittyä itse peliohjelmakoodin kirjoittamiseen, kun taas perinteisemmässä peliohjelmoinnissa pitää keskittyä useisiin muihinkin seikkoihin. Tällainen seikka on esimerkiksi pelin toimiminen ja näkyminen oikein eri laitteilla ja alustoilla. XNA:n käyttäminen on siis perinteistä peliohjelmointia käyttäjäystävällisempää. (What is the xna framework 2006.)

Tammikuussa 2013 Microsoft ilmoitti, että se ei aio tulevaisuudessa tehdä enää uusia versioita XNA:sta. Joka tapauksessa XNA game Studio tulee olemaan edelleen jatkossa tuettu työväline pelien tekemiseen mm. Xboxille sekä Windows Phonelle.

Useat pelinkehittäjät ovat onnistuneet rahallisesti tekemällä Indie-pelejä Xbox-konsolille XNA:n avulla. Tästä huolimatta uusia versioita ei kuitenkaan enää tehdä ja Microsoft on jo kääntänyt katseensa uusiin pelinkehitysympäristöihin. (Microsoft's XNA framework no longer in active development 2013.)



KUVA 8. The Dishwasher: Dead Samurai (Ska-studios 2009)

Dishwasher: Dead Samurai (kuva 8) on XNA-pohjainen ns. “beat’em up” 2D-taistelu/räiskintäpeli. Pelin päähenkilö on Dishwasher, jonka tehtävänä on edetä taso kerrallaan vastustajansa tuhoten erilaisia aseita käyttäen. Pelin kehitti itsenäinen pelikehittelijä nimeltä James Silva. Pelin kehittäminen oli alkanut jo vuonna 2004, ja vuonna 2007 Silva voitti Microsoftin järjestämän pelintekemiskilpailun. Tämän myötä hän sai palkinnoksi 10 000 Amerikan dollaria ja sopimuksen pelin julkaisusta Xbox Live Arcade alustalle.

3.2 Game Maker & Game Maker: Studio

Toisena mahdollisena omien pelien luomisvälineenä kerron hieman Game Maker-tuotesarjasta. Game Makerin (alunperin nimellä Animo) kehitti Hollantilainen Marc Overmars, ja Game Makerin ensimmäinen julkinen versio ilmestyi Marraskuussa vuonna 1999. Ensimmäisessä versiossakin oli sisään rakennettu ohjelmointikieli, joka ei ollut yhtä monipuolinen kuin myöhemmissä versioissa. Aluksi Game Makerilla ei ollut käytössä DirectX-tukea, joten jälki ei vielä silloin ollut aivan uomissaan. Vuosien saatossa Game Maker on saanut uusia julkaisuja ja päivityksiä melko tasaisin

väliajoin. Vuonna 2007 Game Makerin 7.0 version aikoihin Overmars ilmoitti alkaneensa yhteistyöhön Brittiläisen YoYo Gamesin kanssa. Yhteistyön tarkoituksena oli tukea Game Makerin kehitystä ja luoda uusi yhteisö pelinkehittäjille ja tavallisille peliharrastajille.

Game Makerilla tehdään enimmäkseen 2D-pelejä, mutta myös 3D-pelejä on mahdollista tehdä kohtalaisella mittapuulla verrattaen. Game Maker sopii hyvin esim. aloitteleville pelien tekijöille, sillä se on rakennettu niin että koodaamista ei välttämättä hirveämmin tarvitse. Game Makerissa on mahdollista käyttää ns. Drag & Drop -menetelmää, jolloin aloittelijakin voi saada valmiista palasista siedettäviä pelejä. Tosin jos haluaa tehdä edes hieman näyttävämpia ja toimivampia pelejä, on koodia hyvä osata. (Top 5 game development softwares 2011.) Game Maker käyttää omaa sisäänrakennettua ohjelmointikieltä nimeltä Game Maker Language. Koodikielen käyttäminen Game Makerissa antaa pelille enemmän monipuolisuutta, kuin pelkkien standarditoimintojen käyttäminen. Game Maker Language on jokseenkin saman tapainen esimerkiksi JavaScriptin kanssa, joten koodin ei pitäisi tuottaa suurempia vaikeuksia koodauksen harrastajille.

Useiden aikaisempien Game Maker versioiden jälkeen syksyllä 2011 julkaistiin Game Maker:HTML5. Tämä mahdollisti pelien ja animaatioiden toimimisen selaimesta riippumatta mitään liitännäisiä tarvitsematta. Tämä helpottaa luonnollisesti paljon pelinkehittäjän elämää. (GameMaker: HTML5 launch 2011.) Toukokuussa 2012 julkaistiin tuorein Game Maker -perheen jäsen nimeltä Game Maker: Studio. Julkaisussa oli kaikki vanhat ja tutut toiminnot aikaisemmista versioista, mutta mukana oli myös muutamia uusia jippoja. Esim. perus Game Maker: Studiossa tekeleitä voi viedä Windowsille ja Mac Os:lle, mutta lisäosien avulla myös iOS:lle, Androidille ja Windows Phonelle. Näistä osista kuitenkin joutuu jo maksamaan. (GameMaker studio 2013.)

Game Makerin aikaisempia versioita ja Game Maker: Studiota voi ladata ilmaisena versiona, mutta kattavammat versiot maksavat n. 50:stä dollarista aina n. 800:aan dollariin. Game Maker on tosiaankin aloittelijalle varsin hyvä kapistus pelien tekemisen aloittamiseen. Lisäosien ja pro-versioiden avulla myös kokeneempikin pelikoodaaja saa ohjelmasta enemmän irti. (GameMaker studio 2013.)



KUVA 9. Quadropus Rampage (Butterscotch Shenanigans 2013)

Hyvä esimerkki Game Maker: Studiolla tehdystä pelistä on vuonna 2013 julkaistu *Quadropus Rampage* (kuva 9). Game Maker: Studio soveltuu mainiosti juuri tämän tyyppisiin melko yksinkertaisiin, mutta koukuttaviin 2D-tasohyppelypeleihin. Peli on saatavilla PC, Android, iOS ja Mac-alustoille ja se on ladattavissa täysin ilmaiseksi Butterscotch Shenanigansin nettisivuilta. (Quadropus Rampage 2013.)

3.3 Unreal Engine & Unreal Development Kit

Unreal Enginen ensimmäinen pelimoottori ilmestyi vuonna 1998 suosittu *Unreal*-pelin viralliseksi pelimoottoriksi. Tämän jälkeen Unreal Enginestä on julkaistu uusia versioita, ja se on kehittynyt yhdeksi parhaista ja suosituimmista pelinmoottoreista. Viimeisin julkaistu versio Unreal Engine -pelimoottorista on vuoden 2006 Unreal Engine 3.0. Enimmäkseen Unreal Enginellä tuotetaan ensimmäisen ja kolmannen persoonan räiskintä/ammuskelupelejä. Pelimoottorin joustavuus antaa myös mahdollisuuden toisenkin tapaisten pelien luomiseen. (What is UDK? 2012.) Unreal Engine -pelimoottoreiden ja Unreal Development Kit pelinkehitysympäristön ansiosta niin kaupalliset kuin harrastelijoidenkin tekemät pelit ovat saaneet hyvää näkyvyyttä.

Unreal development kit on vuonna 2009 julkaistu Epic Gamesin versiointi Unreal Engine 3:sta. Unreal Development Kit on täysin ilmainen käyttäjilleen ei-kaupalliseen, sekä opetuskäyttöiseen tarkoitukseen. Kaupallisesta käytöstä joutuu pulittamaan 99 dollaria + rojalit. Unreal Development Kit mahdollistaa Unreal Engine 3.0:n tuomat mahdollisuudet näyttävien pelien, ja sovellusten tekemiseen mm. harrastajien ja opisekelijoiden parissa. Se antaa käyttäjilleen mahdollisuuden luoda

korkealaatuisia pelejä, sovelluksia, animoitua grafiikkaa yms. melko helppokäyttöisten työkalujen avulla. Unreal Development kit -ohjelmistossa käytetään UnrealScript -ohjelmointikieltä. Ensimmäisen UDK:n ilmestymisen jälkeen uusia päivityksiä on julkaistu tasaisesti yleensä kuukausittain. (UDK frequently asked questions 2013.)

Kehitysympäristönä Unreal Development Kitissä toimii Windows PC. Unreal Engine 3.0:n ansiosta pelejä ja sovelluksia on mahdollista tehdä esimerkiksi Windowsille, Xbox 360:lle, Playstation 3:lle, Applen iOS-käyttöjärjestelmälle, Mac Os:lle ja Androidille. Esimerkiksi iPhone 4, iPhone 3GS, iPad, iPod ovat tuettuja Applen iOS-käyttöjärjestelmää käyttäviä tuotteita. Luodakseen sovelluksia yms. esimerkiksi iOS-käyttöjärjestelmälle, ei käyttäjä tarvitse erikseen Applen työvälineitä kuten tietokoneita. (Platforms 2012.)



KUVA 10. Gears Of War (Epic Games 2006)

Gears Of War (kuva 10) on Epic Gamesin vuonna 2006 julkaisema kolmannen persoonan räiskinta/ammuntapeli. Pelimoottorina pelissä toimi jo aiemmin mainittu Unreal Engine 3. Peli sai aikanaan erittäin paljon kiitosta sen näyttävästä ulkoasusta ja graafiikoista. Peli on saanut kolme jatko osaa, joista viimeisin *Gears Of War: Judgement* julkaistiin Maaliskuussa 2013. Jatko-osat käyttivät myös Unreal Engine 3:sta pelimoottorinaan. Käytetty pelimoottori tosin on paljon enemmän muokatumpi, kuin ensimmäisessä *Gears Of War* -pelissä.

4 UNITY 3D

Edellisessä osuudessa käytiin läpi muutamaa valitsemaani pelintekovälinettä tarkoituksella melko tiivistetyssä paketissa. Tässä osuudessa keskitytään kuitenkin hieman tarkemmin Unity3D-ohjelmaan, joka on yksi tämän opinnäytetyön keskeisimmistä asioista. Ensin käydään läpi hieman taustatietoa ja Historiaa Unitystä ohjelmasta. Sen jälkeen pyrin selvittämään lukijalle, että minkälainen on Unity3D:n peruskäyttöympäristö. Osuudessa ei kuitenkaan mennä aivan hirveän syvälle Unityn syövereihin, sillä Unityn tarkemmasta käyttämisestä ja projektin kulusta kerron lisää osuudessa kuusi, jossa kerron omasta projektistani Unityllä.

4.1 Taustatietoa Unity3D:stä

Unity Technologiesin Unity3D-ohjelma on tällä hetkellä pelituotannon, ja erityisesti pienpelituotannon alalla yksi suosituimmista ellei suosituin ohjelmisto. Unityllä tehdään mm. 2D -ja 3D pohjaisia pelejä, simulaatioita ja visualisointeja esimerkiksi nettiin, työpöydälle, mobiilisovelluksina, erilaisille konsoleille tai ihan vain omaksi iloksi. Unityllä on tällä hetkellä n. 2 miljoonaa rekisteröityä käyttäjää ympäri maailman ja n. 400 000 aktiivista käyttäjää kuukausittain. Voidaan siis varmasti puhua kohtalaisen menestyneestä pelintekovälineestä ja suuresta käyttäjäkunnasta. (Democratizing development 2013.)

Unityn ensiaskeleet näkivät päivänvalonsa 2000-luvun alkupuolella, kolmen nuoren ohjelmoijan toimesta. David Helgason, Joachim Ante ja Nicholas Francis kokoontuivat ilman suurta rahaa koodailemaan ja urakoimaan sen eteen, mikä nykyään tunnetaan Unityna. He halusivat tehdä jotain saman kaltaista kuin Applen Final Cut Pro teki filmintekijöille. Final Cut Pro antoi amatööri filmintekijöille kohtuuhintaiset ja ammattimaiset välineet filmientekemiseen. Helgason, Ante ja Francis halusivat saavuttaa samankaltaisella idealla järkevän ohjelmiston myös harrastajille pelintekoaikana.

Alkuaikoina kolmikko käytti ennenkaikkea Mac-pohjaisia alustoja, ja tämän takia bisnesnäkökulmasta katsottuna mahdollisuudet olivat jokseenkin rajalliset. Kyseisenä aikana Mac ei ollut vielä kohdannut uutta tulemistaan markkinoille, ja sen paikka

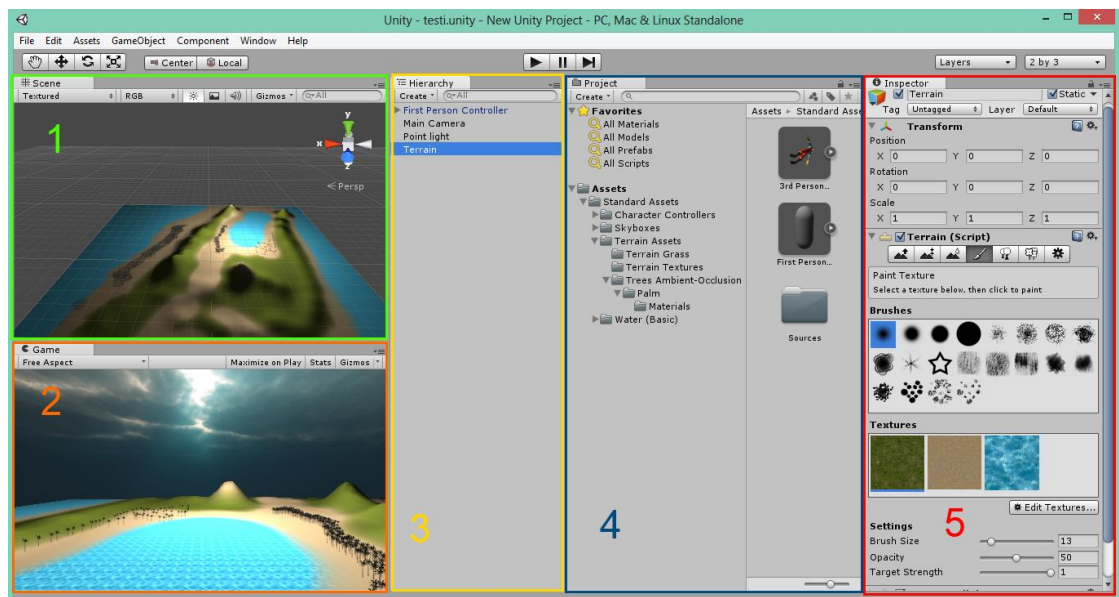
pelialan markkinoilla oli nykyistä paljon vähäpitoisempi. Pitkäjänteinen yritteliäisyys ja sinnikkyys alkoi kuitenkin tuottaa edes pientä hedelmää vuonna 2005. 2005 julkaistiin Unitystä ensimmäinen versio, joka tosin oli huomattavan primitiivinen nykyaikaiseen verrattuna. Näihin aikoihin Unityllä oli olemassa jo tarvittava tuki Windows PC:lle ja nettiselaimille. Vuonna 2008 Unitystä julkaistiin taas hieman jouhevampi versio, ja samana vuonna tapahtui todellinen käännekohta yrityksen ja ohjelmiston kannalta. Silloin Apple julkaisi iPhone AppStoren, ja Unitystä tuli ensimmäinen pelimoottori joka pystyi tukemaan iPhonea. Tämän jälkeen Unity oli kaikkien huulilla.

Läpimurrosta tähän päivään Unitylle on tapahtunut paljon ja se on todellakin noussut kolmen nuoren koodarin projektista kenties tunnetuimmaksi pelintekovälineeksi harrastelijoiden ja pienpelien tekijöiden keskuudessa, mutta myöskin enemmän ja enemmän ammattimaisemmassa sekä suuremmissa mittakaavassa. David Helgason jatkaa edelleen yrityksen johdossa, Joachim Ante on teknologianjohtaja, mutta Nicholas Francis on astunut syrjään Unity Technologiesista. Nyt vuonna 2013 yritys työllistää vajaat 300 muuta työntekijää ympäri maailman, mukaan lukien Suomesta. Unityn uusin versio on taannoin Syyskuussa 2013 julkaistu Unity 4.2.1, joka tukee tällä hetkellä Windowsia, Mac:ia, Linuxia, Unity Web Playeria, iOS:ää, Androidia, Xbox 360:stä ja PS3:sta. Tulevia sopivia alustoja tulevat olemaan mm. Nintendo Wii U, Playstation 4, Windows Phone 8 ja BlackBerry 10. Unityn Pro -versiosta kaikilla herkuilla joutuu pulittamaan repeat 1500 dollaria. Saatavilla on myös ilmaisversio jossa ei ole kaikkia ominaisuuksia, mutta silläkin pelien tekeminen kyllä onnistuu varsin hyvin. (How Unity3D became a game – development beast 2013.)

4.2 Käyttöliittymästä ja sen ominaisuuksista

Unity on siis tällä hetkellä yksi suosituimmista, ellei suosituin pelinteko-ohjelma varsinkin pienpelien tekijöiden keskuudessa. Unityn yksi mukaansatempaavimmista asioista on sen suhteellisen korkea käyttäjäystävällisyys. Unityä käyttämällä pelientekeminen onnistuu tiettyyn pisteeseen asti lähes kokonaan ilman minkäänlaista tietämystä koodaamisesta. Unityssä käyttäjä rakentaa usein projektinsa valmiilla komponenteilla, niitä sitten muokkaamalla omanlaisekseen Unityn tarjoamilla muokkausvälineillä. Koodiscriptejä käyttäjä tarvitsee lähinnä erilaisten toimintojen ja fysiikoiden toteutukseen.

Unityn käyttöliittymä (kuva 11) on ulkoasultaan noviisinkin silmään monia muita kilpailevia alan ohjelmistoja selkeämpi. Käyttöliittymän ulkoasu on Unityssä muokattavavissa muutamien vaihtoehtojen avulla. Näitä ovat mm. default, 4 Split ja 2 by 3-vaihtoehdot. Itse olen Unityä käyttäessäni suosinut 2 by 3-näkymää, mutta projektin aikana näkymää joutuu lähes aina vaihtamaan tarpeen mukaan myös muihin vaihtoehtoihin. Käyttöliittymässä Unityllä on kolme tärkeää näkymää, jotka kannattaa sisäistää jo heti ohjelmaan tutustuessa. Scene-näkymässä luodaan ja pyöritellään peli halutun näköiseksi. Game-näkymässä pystytään katsomaan play-nappulaa painamalla, minkä näköinen tekele oikeasti kyseisellä hetkellä on. Project-näkymässä löytyvästä Assets-valikosta käyttäjä voi hallita, sekä lisäillä erilaisia komponentteja tarpeensa mukaan. Muita näkymiä ovat Hierarchy ja Inspector -näkymät, mutta näistä kaikista pikapuolin hieman enemmän.



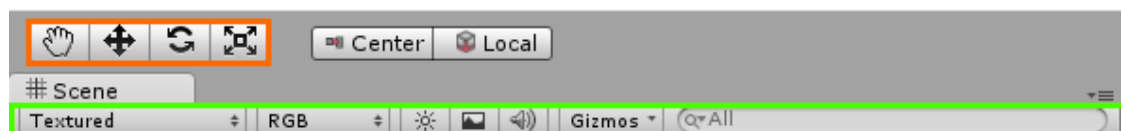
KUVA 11. Unityn käyttöliittymä

Kuvassa 11. näemme esimerkin Unityn käyttöliittymästä, tässä tapauksessa 2 by 3-näkymästä katsottuna. Kuvaan on numeroitu ja havainnoillistettu Scene, Game, Hierarchy, Project ja Inspector -ikkunat kyseisessä järjestyksessä.

4.2.1 Scene ja Hierarchy -näkymät

Scene-näkymä on Unityn ns. hallitseva näkymä. Siihen käyttäjä rakentaa kokonaisuudessaan pelinsä, animaationsa, visualisaationsa yms. Ikkunassa käyttäjälle tarjotaan perspektiivinen 3D-näkymä, jota pystyy pyörittelemään ja katsomaan jokaisesta haluamastaan kuvakulmasta (edestä, takaa, ylhäältä, alhaalta ja molemmilta sivuilta). Scene-näkymä on käytännössä täysin renderoitu muokkaajan näkymä tekeillä olevasta projektista. Kun ikkunaan raahaa hiirellä jonkin objektin (asset), se luo siitä välittömästi peliobjektin projektiin. Scene on täysin sidoksissa Hierarchy-näkymään, jossa näkyy kaikki projektissa käytettävät peliobjektit aakkosjärjestyksessä. (Goldstone 2011.)

Scene-näkymässä apuvälineinä Unityllä on neljä hyödyllistä kontrollityökalua, jotka helpottavat projektin hahmottamisessa ja tarkkailemisessa (kuva 12). Toimintoja voi käyttää niille kuuluvista painikkeista tai niille annetuilla kirjaimilla. Joihinkin toimintoihin on myös käytössä erilaisia näppäinyhdistelmiä, jotka helpottavat näkymässä toimimista. Käsi-työkalun (Q) avulla voidaan liikkua ja navigoida näkymässä vapaasti vasemman hiiren painikkeen avulla. Käyttämällä Alt-painiketta hiiren kanssa voidaan kierrellä aluetta johon käyttäjä katsoo, ja käyttämällä Alt-painiketta hiiren oikean painikkeen kanssa voidaan kohdetta zoomata hiiren rullan avulla. Lisäksi käyttämällä näiden painikkeiden apuna Shift-painiketta voidaan toimintoja nopeuttaa. Siirtotyökalulla/ristillä (W) sananmukaisesti voidaan siirtää objektia/objekteja Scenessä halutessaan paikasta A paikkaan B. Kiertotyökalu (E) toimii käytännössä samalla lailla kuin siirtotyökalu, mutta tarkalleen tällä työvälineellä muutetaan objektin asentoa pikemminkin kuin sijaintia. Viimeisimpänä apuvälineenä on skaalaus (R) työkalu, jolla sitten voidaan muuttaa objektin kokoa tai skaalausta. (Goldstone 2011.)



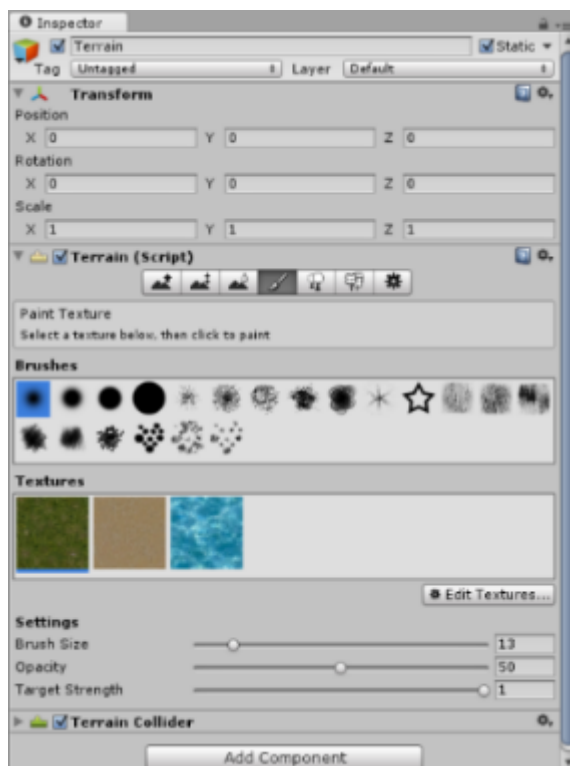
KUVA 12. Scenen kontrollityökalut (oranssi) ja kontrollipalkki (vihreä)

Edellämainittujen apuvälineiden lisäksi scene-näkymän yläpuolella on myös muutamia muita näppäimiä (kuva 12). Näppäimet sijaitsevat ns. kontrollipalkissa ja

niillä voidaan lähinnä muuttaa scenen asetuksia. Vasemman puoleisesta painikkeesta voidaan muuttaa scenen tarkkuustasoa. Vakiona on Textured-vaihtoehto, muita ovat mm. Wireframe ja Lightmap resolution vaihtoehdot. Seuraavasta voidaan muuttaa projektin värijärjestelmää esim. RGB (Red, Green, Blue) tai Overdraw. Kolmannesta saadaan valaistus joko päälle tai pois, neljännessä saadaan grid eli ruudukko näkyviin scenen pohjalle ja viidennestä painikkeesta voidaan säätää äänien voimakkuutta. Palkissa on myös Gizmos-valintapainike, jossa voidaan valita mitä ja miten jotakin Scenessä halutaan näyttää. Esim. mikäli scenessä on valaistus käytössä, voidaan Scenessä näkyvää valaistuspainiketta suurentaa tai pienentää oman valintansa mukaan Gizmos-valinnan alta. Mukana on myös etsi-valikko, josta voidaan kätevästi etsiä jotakin objektia yms. scenestä ja hierarkiasta. Tämä valikko on kätevä varsinkin silloin, kun tekeillä on tavallista suurempia tai monipuolisempia projekteja. (Goldstone 2011.)

4.2.2 Inspector-näkymä

Inspector-näkymä on itseasiassa ns. valvontatyökalu, jossa käyttäjä voi hallita ja valvoa eri elementtien ja objektien ominaisuuksia. Unityssä kaikki pelissä käytettävät objektit ovat yksilöllisiä olioita, jotka sisältävät kaikki yksilöllistä tietoa ja toiminnallisuutta. Inspector-näkymässä voidaan säätää ja muokata jokaista elementtiä ja objektia tekeillä olevassa projektissa. Tämä tarkoittaa käytännössä sitä, että minkä tahansa elementin/objektin valitset, Inspector näyttää juuri sen kyseisen valinnan ominaisuudet ja asetukset. Inspectorissa objektin muokkausvaihtoehtoja ovat mm. positio ja erilaiset tekstuurit. (Selin 2013.) Muokkaaminen näkymässä on tehty selkeäksi ja helppokäyttöiseksi, sen käyttäessä suurimmaksi osaksi valintanappuloita, valikoita, liukusäädin valikoita yms. Unityssä hyödynnetään yleensäkin ns. drag & drop -menetelmiä, joissa haluamiaan asioita voi suoraan valita, raahata, ja pudottaa projektiin. Inspector näkymässäkin tiettyihin valintoihin voi soveltaa drag & drop -menetelmää, mikäli se tuntuu itselleen parhaimmalta vaihtoehdolta. (Goldstone 2011.)



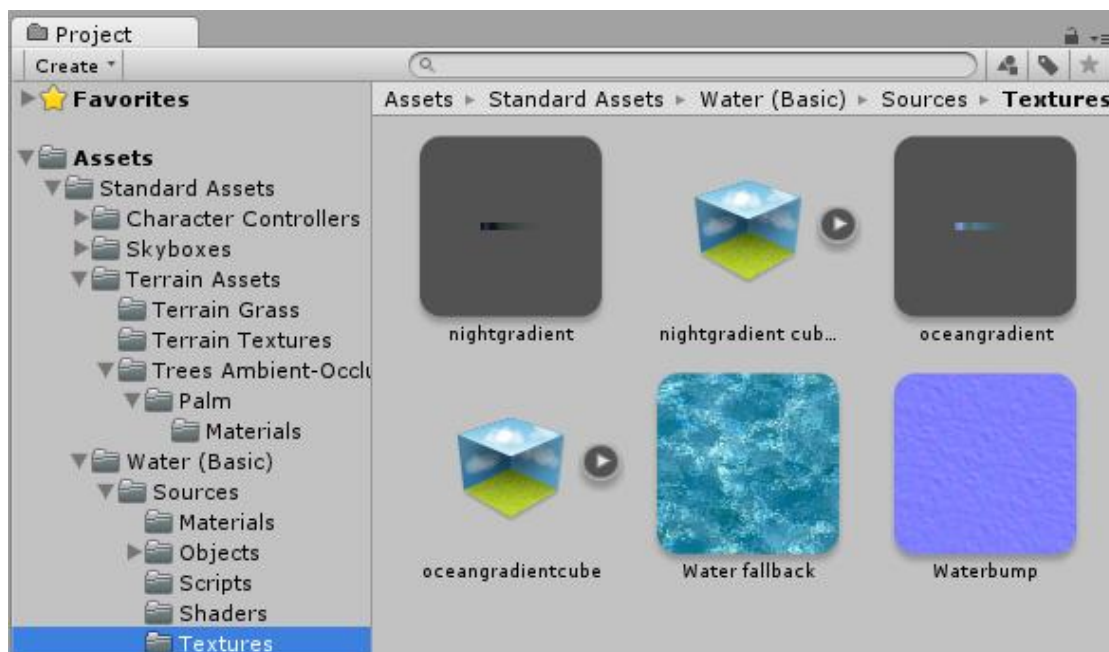
KUVA 13. Inspector-näkymä

Kuvassa 13 nähdään joitakin Inspector-näkymän työkaluista. Tässä valittuna on projektin terrain eli scenessä käytettävä alusta ja sen muokkausvaihtoehtoja.

4.2.3 Project-näkymä ja Assets

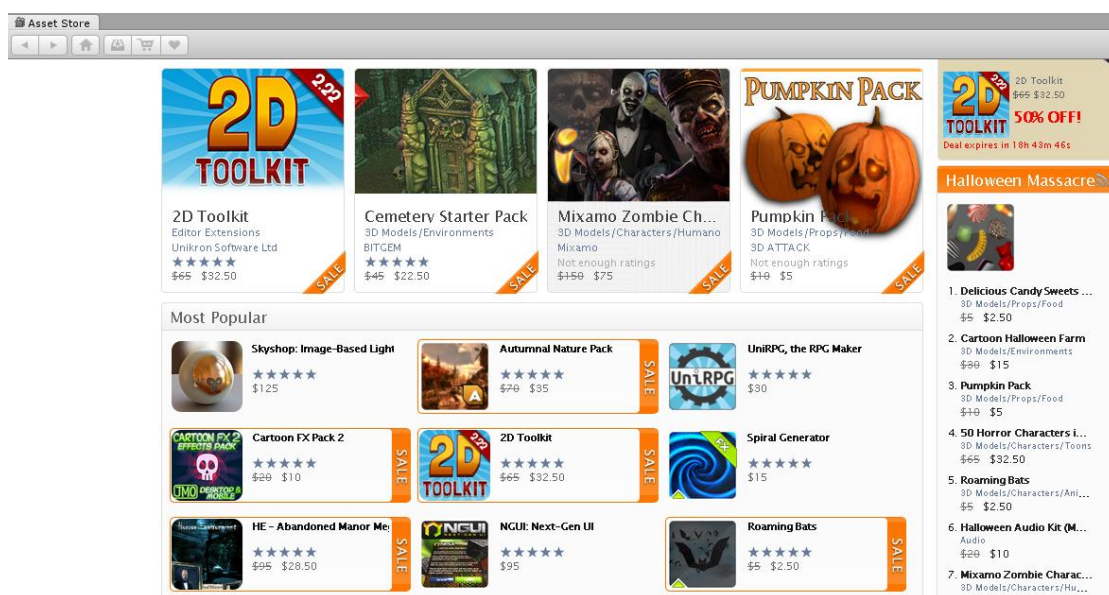
Project-näkymässä (kuva 14) pystytään näkemään kaikki Unity projektissa käytettävät Assetit. Unityssä jokainen tekeillä oleva projekti koostuu pääkansioista, joka sisältää kolme alakansiota. Alakansiot ovat Assetit, Kirjasto ja Unityn ollessa toiminnassa myös Temp-kansio. Kun Assetteja/objekteja lisätään Assets-kansioon, ne näkyvät samantien Project-ikkunassa ja tulevat osaksi projektiasi. (Goldstone 2011.)

Assetit ovat elementtejä joita valmiin pelin yms. valmistumiseen tarvitaan. Assetteja voivat olla mm. erilaiset tekstuurit, mallinnukset, koodinpätkät, äänet ja videot. Pelinteko-ohjelmissa valtaosin käytettävät assetit ja objektit ovat alunperin tehty ja muokattu muissa ohjelmistoissa kuten vaikkapa Adobe Photoshopissa. Unity ei ole tässä asiassa poikkeus, vaan tässäkin tapauksessa projektit usein kootaan valmiista elementeistä ja toimivuus saadaan ohjelmoitua aikaiseksi Unityn omalla pelimoottorilla. (Selin 2013.)



KUVA 14. Project-näkymä ja joitakin Assetteja

Unityssä tulee mukana ns. Standard Assets kokoelma, joka sisältää valmiita elementtejä käyttäjälleen. Unityn ilmaisversiossa valmiita Assetteja on luonnollisesti jonkin verran vähemmän, kuin Pro-versiossa. Valmiita Assetteja ja niiden kokoelmia pystyy myös hankkimaan helposti Unityn omasta Asset Storesta (kuva 15). Se on Unityn oma Assettien verkkokauppa, josta pystyy ostamaan erilaisia elementtejä, tekstuureja yms. Asset Storesta löytyy myöskin täysin ilmaisia tuotteita käyttäjilleen. Asset Store löytyy Unityssä paavalikon Window- kohdasta, ja sen alavalikosta Asset Store. (Selin 2013.)



KUVA 15. Asset Store

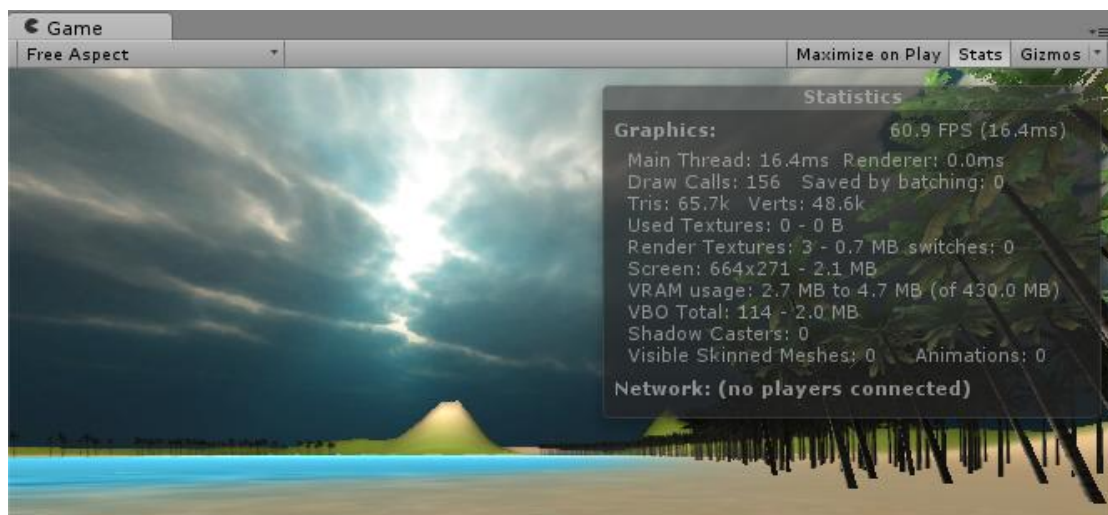
Unityn Asset Storesta (kuva 15) löytyy lukematon määrä erilaisia maksullisia ja ilmaisia Assetteja. Kuva on Asset Storen etusivulta, johon on yleensä listattu suosituimmat, myydyimmät ja parhailla alennuksilla hankittavat tuotteet.

4.2.4 Game Window

Game-näkymä (kuva 16) toimii eräänlaisena testausvälineenä, jossa näet sen hetkisen realistisen version tekeleestäsi. Tämä näkymä on todella helppokäyttöinen, mutta on kuitenkin muutama asia, jotka käyttäjän on hyvä tietää. Näkymä voidaan aktivoida Scene-näkymän yläpuolella olevasta Play-painikkeesta, pistää tauolle Pause-painikkeesta sekä uudestaan Play:tä painamalla näkymä saadaan pois päältä ja aktiivisesta tilasta. Vakiosuhdanteena näkymässä on Free Aspect, muita ovat mm. 4:3, 16:10 sekä laajakuva eli 16:9 kuvasuhdanne. Kuvasuhdennetta voidaan muuttaa kätevästi klikkaamalla vakiona olevaa Free Aspect -vaihtoehtoa, jolloin muutkin vaihtoehdot tulevat esille.

Game-näkymä voidaan tottakai myöskin maksimoida koko ruudulle, jolloin käyttäjä saa paremman näkymän ja tuntuman peliinsä. Tämä tapahtuu yksinkertaisesti näkymän yläpuolella olevasta Maximize on play-painikkeesta Game-näkymän ollessa aktiivisena. Edellämainitun painikkeen oikealta puolelta löytyy vielä Stats niminen painike, josta saadaan näkymiin testin aikaista statistiikkaa ja tilastoa. (Selin 2013.)

Yksi tärkeä asia mikä käyttäjän tarvitsee tietää Game-näkymää käyttäessään on, että se tosiaanakin on vain testausta ja havainnointia varten. Eli toisin sanoen näkymän ollessa aktiivisena käyttäjän on aivan turha tehdä muutoksia projektiinsa, sillä kun näkymä otetaan pois käytöstä, sen aikana tehdyt muutokset ja lisäykset eivät ole tallentuneet projektiin. (Goldstone 2011.)



KUVA 16. Havainnoillistava kuva Game-näkymästä

Kuvassa 16. näemme tätä osiota varten tekemästäni yksinkertaisesta Scenestä aktiivisen Game-näkymän. Näkymän yläpuolella ovat osiossa aiemmin mainitut painikkeet. Stats-painike on aktiivisena, joilloin se näyttää erilaisia tilastoja projektista.

5 TUTKIMUS UNITYN KÄYTTÄMISESTÄ SUOMESSA

Tässä osuudessa käsittelen hieman sitä kuinka laajaa Unityn käyttäminen on Suomessa. Osuudessa pyrin selventämään lukijalle omien kyselyideni kautta saamieni tietojeni perusteella parhaani mukaan aihetta. Ensimmäiseksi taustoitan hieman kyselyn perusteita ja muitakin taustoja. Sen jälkeen pyrin analysoimaan tuloksia ja kyselykeinoja mahdollisimman fiksulla tavalla, jonka jälkeen vuorossa on vielä eräänlainen yhteenveto tuloksista ja tiedoista, sekä suomalaisen pelialan ja Unityn tulevaisuuden näkymistä. Pidemmittä puheitta käydään aiheen kimppuun ja katsotaan mitä olen saanut selville.

5.1 Kyselyn taustoja

Heti ensimmäiseksi haluaisin painottaa, että tämä tutkimus ei ole lainkaan mikään kaikista kattavin ja laajin. Sen sijaan tutkimuksen tarkoituksena on kevyesti raottaa ja selventää Unityn käyttämistä Suomessa muutamien merkittävien vastausten ja tietojen valossa. Tämä tutkimus tuli osaksi tätä opinnäytetyötä ehkäpä hieman ns. takaporttien

kautta, mutta tiedostin että on olemassa joitakin tietoja joita on hyvä tuoda esille tämän kaltaisten kyselyiden kautta.

Sen jälkeen kun olin suurinpiirtein hahmotellut ja saanut jonkinlaisen kuvan siitä mihin suuntaan opinnäytetyöni tulee etenemään, tuli myöskin idea tämän tutkimuksen tekemiseen. Yhdessä ohjaavan opettajan Jukka Selinin kanssa käytyjen keskustelujen jälkeen olimme molemmat sitä mieltä, että tutkimus olisi hyvä mauste tähän opinnäytetyöhön. Silloin jo tuli kuitenkin selville, että tietojen saaminen suoraan suomalaisilta peliyrityksiltä Unityn käyttämisestä tulisi jo itsessään olemaan jonkin sorttinen haaste. Kaikki jotka edes hieman seuraavat pelialaa yms. varmasti tietävät, että kilpailu on melkoisen kovaa yritysten kesken. Tämä taas usein johtaa ympäripyöreisiin vastauksiin ja jopa tietynlaiseen asian vierestä kiertelyyn käytettävistä ohjelmistoista yms. kysyttäessä.

Tällä hetkellä suomalainen peliala ja peliohjelmointi ovat melkoisen huimassa nosteessa ennen kaikkea Rovion ja Supercellin kaltaisten peliyritysten ansiosta. Rovio tunnetaan parhaiten heidän huimaan maailmansuosioon ponnahtaneesta Angry Birds -pelisarjasta ja Supercell teki juuri syksyllä 2013 miljardikaupat, joka teki yhtiön omistajista ja osakkaista hetkessä miljonäärejä. Juuri tämänkaltaisilta maailmanlaajuisesti tunnetuilta miljoonien eurojen yrityksiltä, on usein aika vaikeaa saada mitään konkreettista tietoa irti heidän työmenetelmistään. Usein yritykset haluavat pitää omat metodinsa yksityisinä ja salassa kilpailevilta yrityksiltä ja muilta tahoilta. Vaikka vastauksen saisikin, siinä saattaa kestää hyvinkin kauan, koska tämän kaltaiset yritykset saavat niin paljon kyselyitä ja tiedusteluja.

Minulla ei itseasiassa ole itse pelialasta ns. taustojen ja yritysten osalta sen syvemmin kokemusta, muuten kuin että olen aina pitänyt videopeleistä ja pelannut niitä enemmän tai vähemmän koko ikäni. Koska tutkimuksen pääsääntöisenä aiheena oli nimenomaan Unity3D-ohjelman käyttäminen Suomessa ja suomalaisissa peliyrityksissä, ensimmäiseksi otin selvää mitkä ovat sellaisia merkittäviä yrityksiä ja lähteitä joista tietoa kannattaa ruveta haalimaan. Kriteerinä itselleni oli että yritys olisi menestynyt edes jollakin tasolla ja pystynyt julkaisemaan jokseenkin onnistuneita pelituotteita. Tämän kaltaisia yrityksiä olivat itseoikeutetusti Rovio, Supercell ja mm. Frozenbyte, Remedy Games ja Sulake. Yhteensä kyselyideni kohteiksi päättyi n. 10-20 pelialan vaikuttajaa. Lisäksi kyselin tietoa mm. suomalaisen pelialan pääjärjestöltä

NeoGamesilta ja itse Unityltä. Kaikki kyselyni suoritin Mikkelin Ammattikorkeakoulun oman sähköpostipalvelun kautta, jotta kyselyn kohteelle tulisi heti selville kyseessä ei ole mikä tahansa tietojen urkkija.

5.2 Analysointia

Kuten jo aiemmin mainitsin oli tämän pienen tutkimuksen tekeminen yllättävänkin työlästä vaikka kohteita ei ollutkaan kovinkaan useita. Kyselyihin vastattiin vaihtelevasti ja mielestäni jokseenkin aika nihkeästikin. Tämä saattoi toki johtua kokemattomuudestani siihen, miten tämän kaltaisia kohteita tulisi lähestyä. Sain kuitenkin jotakin konkreettista tietoa, ja siitä haluankin kertoa tässä osuudessa parhaani mukaan.

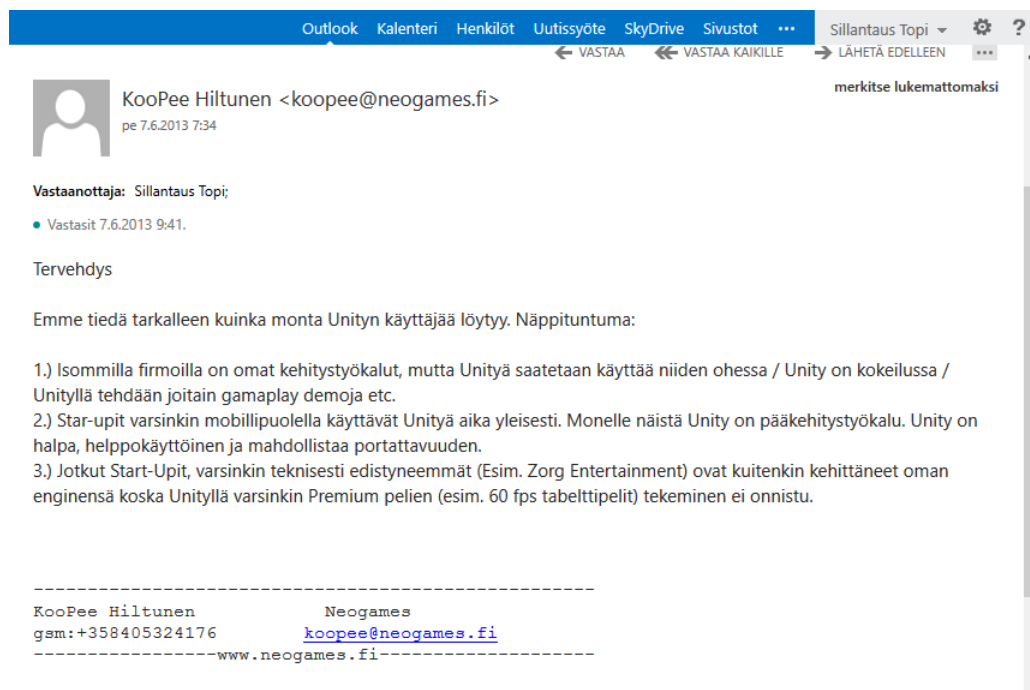
Alunperinkin pelkäämäni nihkeys kävi osaltaan toteen tutkimustuloksia puntaroidessani. Ensimmäiseksi lähettämiini pelialan yrityksille tarkoitettuihin viesteihin sain vastauksen vain murto-osalta yrityksiltä. Toisaalta onnistuin onkimaan tietoa Itse Unityltä englanninkielisen kyselyn kautta, sekä saamaan Suomen pelialan pääjärjestö NeoGamesilta mielenkiintoisen vastauksen. Unityltä saamani vastaus oli ennalta arvatenkin melkoisen ympäröörä, mutta siitä sain kuitenkin irti muutamia tärkeitä tietoja. Unityn ja NeoGamesin vastauksista hieman myöhemmin lisää.

Super Celliltä ei tullut missään vaiheessa vastausta viestiini, enkä toisin sanoen tiedä onko yritys käyttänyt Unityä missään tarkoituksessa. Luulisin kuitenkin, että Supercellillä kuten useimmilla muillakin menestyneimmillä pelialan yrityksillä on omat pelimotorit ja pelintekovälineet tuotteiden tekemiseen. Roviolta vastattiin kyselyihin aina perinteiseen tyyliin ”Kiitos yhteydenotostanne, otamme teihin yhteyttä mahdollisimman pian”, kuitenkin vastausta koskaan saamatta. Rovio on kuitenkin tiedetysti käyttänyt Unityä ainakin osittain pelimoottorinaan mm. vuonna 2012 julkaistussa hyvän vastaanoton saaneessa pelissä *Bad Piggies*. *Bad Piggies* on ulkoasultaan hyvin paljon megasuosisitun *Angry Birds* -pelin kaltainen. Toisin kuin *Angry Birds*issä linnut hajoittavat, niin *Bad Piggies*issä possut rakentavat. *Bad Piggies* on saatavilla mm. iPhonelle, Androidille, PC:lle, Mac:lle ja iPadille. (*Bad Piggies* 2012.)

Muutamasta yrityksestä kuten vaikkapa Frozenbytelä ja Sulakkeelta sain hyvin samankaltaisen vastauksen. Yritykset kertoivat tekevänsä arvatenkin kaikki pelinsä omilla pelimoottoreilla jne. Frozenbyte paljasti, että he eivät käytä Unityä laisinkaan mihinkään tarkoitukseen, mutta Sulake joka on tunnettu lähinnä *Habbo Hotel* -pelistään oli käyttänyt Unityä ainakin tietyissä tarkoituksissa kehitystyökaluna. Sulakkeelta saamassani viestissä he myös kertoivat, että ovat puntaroineet mahdollisuutta Unityn käyttöön tulevaisuudessa. He myös olisivat halunneet minulta tietoa opinnäytetyöni edistyessä siitä, mitä olen saanut selville yrityksistä jotka Unityä mahdollisesti käyttävät. Tämä seikka oli hyvin mielenkiintoinen ja korosti vielä enemmän sitä, että kilpailu on suomalaisella pelialalla kovaa, ja että jokainen tiedonriipe ympäröivästä pelialan tilanteesta on mahdollisesti hyödyllistä. Sulake selkeästi osoitti enemmän mielenkiintoa kyselyihini kuin muut vastaavat yritykset.

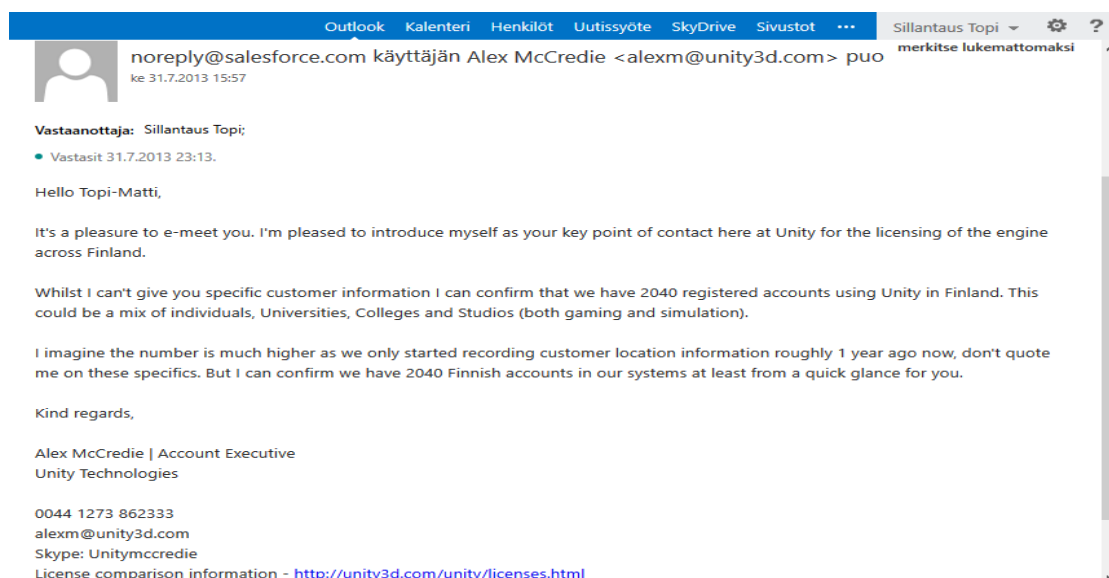
Yhdessä ohjaavan opettajani Jukka Selinin kanssa käymiemme keskustelujen jälkeen päädyimme siihen, että yksi merkittävimmistä lähteistä tähän kyselyyn saattaisi olla pelialan kattojärjestö Neo Gamesilta mahdollisesti saatu tieto. Arvelin aluksi, että Neo Gamesilta ei välttämättä tietoa heru, mutta toisin kuitenkin kävi. Neo Games itsessään on suomalaisen pelialan ns. kattojärjestö, jonka alla suomalaiset peliyritykset toimivat. Neo Games, myös ajaa suomalaisten peliyritysten etua myös ulkomaalaisissa pelialan järjestöissä. (Mikä on Neo Games? 2013.) Neo Gamesin yhteyshenkilönä kyselyssäni toimi järjestön johtaja Koopee Hiltunen.

Olin erittäin positiivisesti yllättynyt, että Koopee vastasi kyselyyni ja vieläpä melko nopealla aikavälillä. Vastaus (kuva 17) sinänsä ei tuottanut hirveämmin yllätyksiä, mutta tässä vaiheessa kaikki tieto liittyen Unityn käyttöön oli pelkkää bonusta. Vastauksesta kävi ilmi jo moneen kertaan todettu, että suurimmat pelialan yritykset käyttävät lähes poikkeuksetta omia pelimoottoreita ja välineitään. Selville kävi myös, että eritoten mobiilipuolella Unityn käyttö oli melko yleistä sen ollessa edullinen ja helppokäyttöinen. Hiltunen ei myöskään suoralta kädeltä osannut/tahtonut kertoa osviittaa antavaa lukumäärää Unityä mahdollisesti käyttävistä suomalaisista yrityksistä.



KUVA 17 Neo Gamesin Koopee Hiltusen vastaus kyselyyni

Näiden selvitysten jälkeen mietin, mistä vielä saisin sellaista tietoa jota voisin tässä kyselyssä käyttää. Tietoa kuitenkin tarvitsin vielä, sillä sitä oli tullut itselleni nihkeänlaisesti. Päätin kokeilla kepillä jäätä ja kysäistä itse Unityltä olisiko heillä mitään näppituntumaa siihen kuinka paljon Suomessa on heidän ohjelmansa käyttäjiä. Tämän kyselyn tein, kuten jo aiemmin mainittua Englanniksi suoraan yrityksen kotisivuilta löytyvistä yhteydenottovaihtoehtoista. Vastauksen (kuva 18) sain melko nopeasti parin päivän kuluessa.



KUVA 18. Unityltä vastaanotettu Alex McCredien vastaus

Kyselyviestiini vastasi Unityltä Alex McCredie Brightonista Isosta-Britanniasta, joka onnistui antamaan minulle tärkeää tietoa liittyen Unityn käyttäjämääriin Suomessa. Alex on Unityllä Account Executiven roolissa, eli hän vastaa Unityn käyttäjien sopimuksista ja lisensöinneistä. Hän kertoi, että Unity3D-ohjelmalla on tällä hetkellä ainakin 2040 rekisteröitynyttä ohjelman aktiivikäyttäjää Suomessa. Hän kertoi että Unityllä oltiin aloitettu paikannettu käyttäjien tarkistus n. reilu vuosi sitten, ja että lukumäärä todennäköisesti on kasvanut huomattavasti viimeisten tilastointien jälkeen. Unityä käytettiin Suomessa enimmäkseen koulutustarkoitukseen esimerkiksi yliopistoissa, eri asteiden korkeakouluissa, pelialan yrityksissä ja studioissa sekä yksittäisten henkilöiden omissa projekteissa. Tämä kaikki tieto, ennen kaikkea edes jonkinlainen varmennettu käyttäjien pyöristetty lukumäärä oli mieluisaa tietoa. Vaikka näihin tietoihin olinkin tyytyväinen olisin myös halunnut saada ehkä jonkinlaista tietoa siitä, kuinka paljon Suomessa on käyttäjiä Unityn Pro-versiolla jne.

5.3 Yhteenveto tuloksista

Tutkimuksen tarkoitus ei siis ollut se, että siinä selvitetäisiin kaikkien Suomen pelifirmojen ja vaikuttajien taustat Unityn käyttämiseen. Halusin siis valita omasta mielestäni varteenotettavimpia vaikuttajia ja kysellä heiltä heidän suhdettaan Unity3D-ohjelmaan. Mielestäni sain näilläkin lähteillä lähestulkoon kaikki haluamani vastaukset kuten miten Unityä käytetään Suomessa, kuinka paljon sitä käytetään, ketkä sitä käyttävät ja miten pelifirmat pääsääntöisesti siihen suhtautuvat ja sitä hyödyntävät.

Kuten jo niin moneen kertaan tässä osuudessa todettua, oli aivan selvää jo alun alkaenkin että vastausten onkiminen tulee olemaan vaikeaa. Vaikkakin haluamistani pelifirmoista vastasi vain kourallinen, selvisi siitä kuitenkin että lähes kaikki käyttivät Unityä tällä hetkellä vain kehitystyökaluna tai sitten eivät ollenkaan. Kuitenkin Koopee Hiltusen mukaan Unityä käytettiin, jopa melko yleisesti pienemmissä yrityksissä ja projekteissa eritoten mobiilipuolella. Myös kiinnostus joidenkin yritysten puolesta kyselyäni kohtaan oli yllättävää, heidän halutessaan mahdollisesti itsekin tietoja Unityä käyttävistä toisista pelialan yrityksistä.

Tätä pientä tutkimusta kuvasi mielestäni koko ajan pieni salamyhkäisyys pelialan yritysten puolesta, vastausten ollessa usein todella ympäröityjä. Kuitenkin selkeää laajempi kiinnostus Unityä kohtaan on Suomessa havaittavissa. Yksinkertainen totuus on tällä hetkellä se, että vaikka jotkin suuremmistakin peliyrityksistä voisivatkin ehkä Unityä tositarkoituksella käyttää, heillä on jo todella hyvät pelimoottorit omasta takaa, joten he eivät ainakaan toistaiseksi koe tarpeelliseksi Unityn käyttöä.

Alex McCredien kertoma tieto siitä, että Suomessa on ainakin 2040 rekisteröitynyttä oli mielestäni Unityn Suomalaiskäytön kannalta lupaava. Osuudessa 4. kerroin, että Unityllä on tällä hetkellä maailmanlaajuisesti 2 miljoonaa rekisteröityä käyttäjää ja 400 000 aktiivikäyttäjää kuukausittain. Tähän verrattuna McCredien kertomat lukumäärät Suomen Unity käyttäjistä on vain yksi kivi louhoksessa. Täytyy kuitenkin muistaa, että Suomi on pieni maa, joten siihen nähden käyttäjälukumäärät ovat todella positiivisia. Suomalaisen pelialan loistavan nykytilan ja taitotason tietäen uskoisin, että Unityn käyttäminen ja mielenkiinto siihen lisääntyy entisestään kovalla kohinalla lähitulevaisuudessa.

6 OMA CASE: UNITY DEMONSTRAATIO

Tässä osuudessa kerron tekemästäni demosta liittyen Mikkelin ammattikorkeakoulun Mikpoli rakennukseen. Demossa käytin hyväkseni Google SketchUp sekä Unity3D-ohjelmia. Aluksi kerron hieman pohjustusta projektiin, sekä sen tämänhetkiseen tilaan pääsemisen prosessista. Sen jälkeen on vuorossa demon Unity3D-osuus, eli työn tuominen Unityyn ja projektin viimeistely. Haluan tarkentaa, että ainakaan toistaiseksi kyseessä ei ole peli vaan enemmänkin pelkistetty havainnoillistava demonstraatio Mikpoli-rakennuksesta ja tietyistä sen alueista/paikoista.

6.1 Demon idea ja tarkoitus

Ensimmäiset ideani ja ajatukseni tämän kaltaisen havainnoillistavan demon tekemiseen sain alkukesällä 2013 ollessani työharjoittelussa Mikkelin ammattikorkeakoululla sijaitsevassa CampusIT:ssä. Työharjoittelun loppuvaiheessa rupesin omaksi ratokseni tekemään muiden töiden lomassa 3D-mallinnusta CampusIT:n toimisto/työtiloista. Pienen pohdinnan jälkeen päädyin tekemään

mallinnusta Google SketchUp -ohjelmalla, koska se oli itselleni sillä hetkellä helppokäyttöisin ja tuttavallisin.

Tässä vaiheessa en vielä tosiaankaan tiennyt, että mallinnukseni tulisi olemaan osa lopputyötäni. Mallini (kuva 19) ei missään vaiheessa ollut todellakaan mikään monimutkainen yms. Tarkoituksena oli tehdä toimistotilat, jossa sitten olisi joitakin tarvikkeita kuten pöytiä, tuoleja, tietokoneita, kaappeja jne. Tarvittavia tavaroita malliini sain helposti käyttämällä mm. SketchUpin omaa 3D Warehouse -toimintoa. Ladattuani tavaroita, muokkailin ja asettelin niitä haluamieni tarpeiden mukaan huoneen sisällä.

Mitä pidemmälle mallinteluni jatkui, aloin päässäni pyöritellä erilaisia ajatuksia jatkon kannalta. Ajattelin että jos saisin mallista suhteellisen realistisen oikeisiin CampusIT:n tiloihin verrattuna, voisin kenties tuoda mallin johonkin toiseen ohjelmaan ja luoda siihen jotain toiminnallisuutta. Tässä vaiheessa tuli ensimmäisenä ja oikeastaan ainoana vaihtoehtona esiin Unity3D. Tämä siksi koska Unity3D oli itselle kaikkein tutuin tarpeeseen sopiva ohjelma, sekä kosketukseni muihin 3D-ohjelmiin olivat jääneet hyvin rajallisiksi.



KUVA 19. Mallinnukseni CampusIT:n tiloista Google SketchUp -ohjelmassa

Kun olin mallintelussani kuvan 19. kuvaamassa vaiheessa keskustelimme useasti CampusIT:n vastuuopettaja Janne Turusen sekä opinnäytetyöni ohjaavan opettajani Jukka Selinin kanssa projektin jatkumahdollisuuksista. Pienen viilaamisen jälkeen päädyimme alustavasti sellaiseen vaihtoehtoon, että toisin mallinnukseni Unity3D-ohjelmaan, jonka jälkeen huoneessa voisi vapaasti liikkua ja esim. tietokoneelle pysähdyttäessä tietokoneella olisi toimintaa. Tämä kuitenkin oli vain alustava suunnitelma, joka sitten muuttuikin vielä melko radikaalisti.

Myöhemmin sain tietooni, että eräällä opettajallani Esa Hannuksella olisi hallussaan koko Mikpolista SketchUpilla tehty raakamallinnus (kuva 20). Mallinnuksen oli suurimmaksi osaksi tehnyt CampusIT:llä aiemmin työskennellyt harjoittelija. Kysyin Esalta, voisiko mahdollisesti koko Mikpolista tehtyä mallinnusta käyttää osana projektiani, mikä sopikin Hannukselle hyvin. Saamani mallinnus Mikpolista oli tehty oikeisiin mittakaavoihin ja kyseessä oli ollut ilmeisen laaja projektio. Kuitenkin mallinnus oli todella raaka siinä mielessä, että siinä ei ollut oikeastaan mitään tekstuureja käytetty hyväksi. Muokkasin mallia (kuva 20) lisäämällä siihen tekstuureja kuten tiiliseinää, ikkunanlasia ja kattotekstuuria. Nämä muokkaukset tein toistaiseksi vain mallinnuksen ulkopinnoilla jättämällä sisällä olevat muokkaukset mahdollisesti myöhemmäksi.



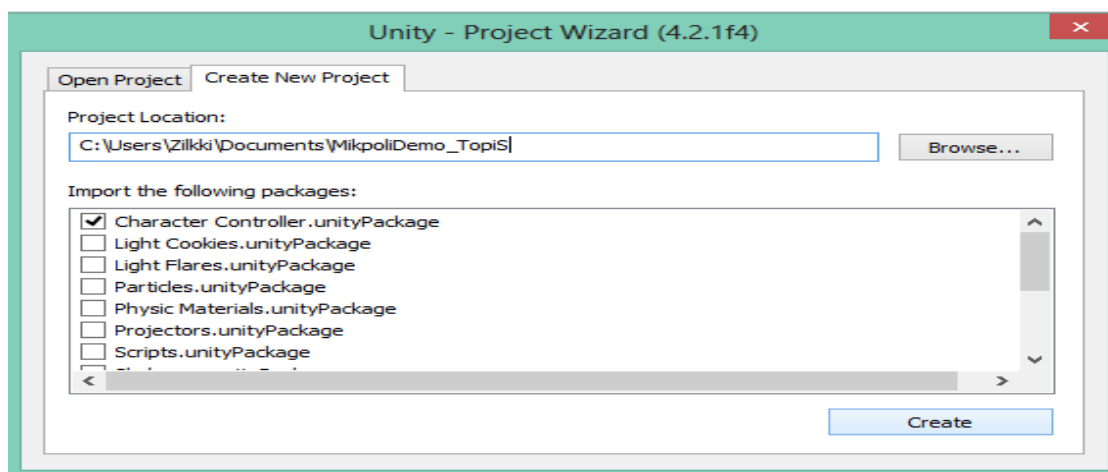
KUVA 20. Yläpuolella Esa Hannukselta saamani malli ja alapuolella samasta mallista muokkaamani versiointi

Muokkailtuani Google SketchUp -malleja päädyin ainakin lähtökohtaisesti yhteen tiettyyn ratkaisuun. Tarkoituksena olisi, että Mikpolin mallin sisällä voisi liikkua vapaasti ja CampusIT:n tilojen läheisyydestä pääsisi toimiston tiloihin sisälle käyskentelemään. Seuraavassa osuudessa tarkastellaan demoilun seuraavaa vaihetta Unity3D-ohjelmassa.

6.2 Mallien tuominen Unityyn ja projektin kulku

Tässä vaiheessa kun Google SketchUp -mallinnukset ovat valmiita on aika siirtyä Unityn puolelle jatkamaan demoa. Tarkoituksena olisi siis saada tuotua mallinnukset Unityyn ja saada ne toimimaan erillisinä scene-näkyminä omissa tarkoituksissaan. Ensimmäiseksi tässä osuudessa on luvassa ns. alkuvalmistelut projektin kulun kannalta. Seuraavaksi pyritään saamaan tarvittava toimivuus demoon, jonka jälkeen pyrin kertaamaan mitä olen saanut aikaan ja miten tätä demoa voisi kenties tulevaisuudessa jatkaa.

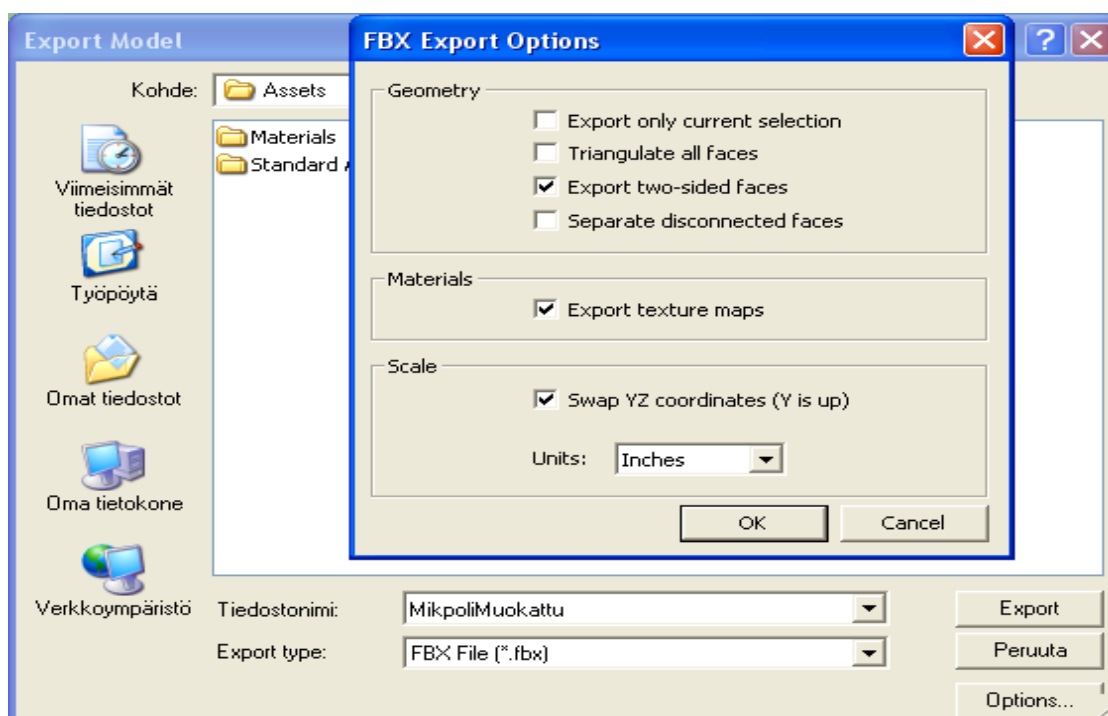
Tätä projektia varten valmistelut Unityn puolella ovat todella yksinkertaiset, sillä demoa varten tehdyt mallit ovat Unityyn tuomista vaille valmiit. Unityn avatessa ohjelma kysyy jatketaanko mahdollisesti vanhaa projektia, vai ruvetaanko tekemään uutta. Tässä tapauksessa tietenkin otamme uuden projektin ja annamme sille nimeksi vaikkapa MikpoliDemo_TopiS (kuva 21). En tarvitse mitään Assetteja, scriptejä yms. tässä vaiheessa. Kunhan pääsen Unityyn puuhastelemaan, voin sitten sieltä lisäillä mahdollisia tarvittavia osasia työhön.



KUVA 21. Projektin aloitus Unityssä

Ensimmäisenä Unityssä aloitin yksinkertaisesti tekemällä alustan eli terrainin Demoani varten. Terrainin luomisen jälkeen, sen päälle ruvetaan luomaan haluttua projektia ja maailmaa. Terrain saadaan luotua yksinkertaisesti valikosta Game Object, sen alavalikosta Create Other ja sieltä valitaan Terrain. Oman demoni pohjaksi loin aivan pelkistetyn harmaan hieman asfalttia muistuttavan alustan.

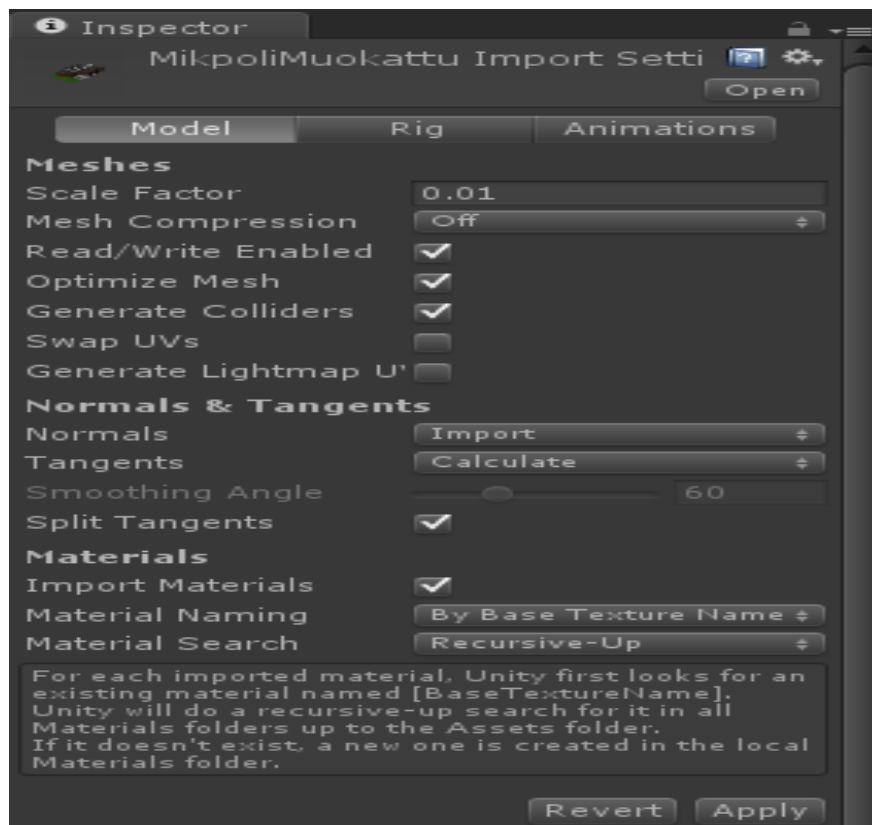
Alustan luomisen jälkeen tarvitsin Google SkethUp -mallinnukseni tuoduksi Unityyn. Mallinnukset pitää tuoda SkethUpista käsin Unityyn valitsemalla File-valikon alta Export → 3D Model. Sen jälkeen valitsin oman projektini kansion ja sieltä Assets, jolloin malli menee suoraan Unityprojektin Assetteihin. Tämän vaiheen kanssa minulla oli pieniä vaikeuksia, koska en oikein aluksi tiennyt että missä tiedostomuodossa mallit kannattaisi tuoda Unityyn. Ensinnäkin kokeilin aluksi tuoda mallini Unityn Free-versioon mm. .fbx ja .dae muodoissa. En onnistunut saamaan .fbx-muodossa tekstuureja mukaan kunnolla ilmaisversioon, koska kyseinen muoto on tarkoitettu toimivan oikein vain Pro-versiossa (Start with Unity3D 2013). Tiedostomuodossa .dae taas onnistuin tuomaan mallit ongelmitta Unityyn, mutta en siltikään ollut täysin tyytyväinen lopputulokseen. Loppujen lopuksi asensin Unityn uudestaan, otin tätä projektia varten käyttöni Pro-version kokeiluversion ja toin mallinnukseni Unityyn .fbx-muodossa.



KUVA 22. Exporttaus Google SketchUp -ohjelmasta

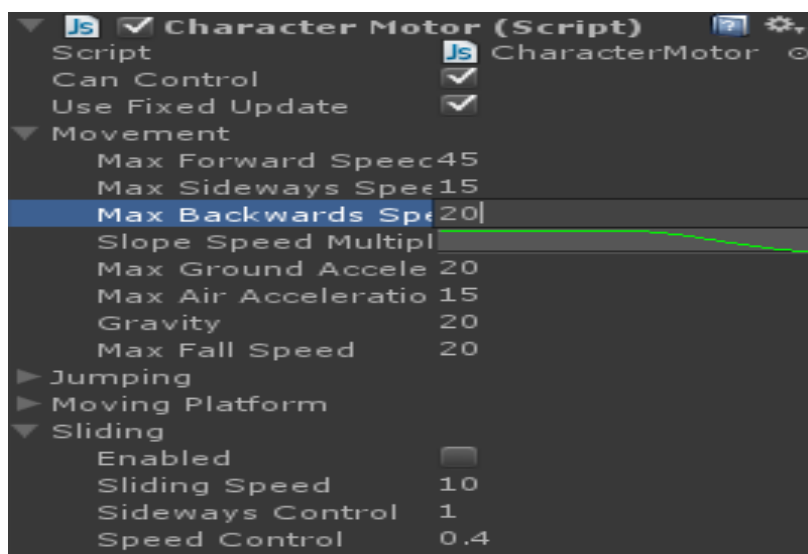
Tärkeää mallin tuomisen kannalta oli valita export vaiheessa Options-painikkeen alta vaihtoehto Export two-sided faces (kuva 22). Tämä kannattaa valita sen takia, että Unityyn ulkopuolelta tuoduista malleista Unity usein tunnistaa vain päälle päin näkyvät tekstuurit. Nyt kun Export two-sided faces on valittu, niin Unity huomioi myös rakennuksen sisällä olevat ulkoapäin näkymättömissä olevat tekstuurit. Tämän jälkeen mallinnukset näyttivät mielestäni tähän tilanteeseen riittävän hyviltä. Nämä samat temput toistuivat jokaisessa uudessa scenessä demossani.

Kun kyseessä oleva malli oli onnistuneesti tuotu Unityyn, asettelin sen Terrainin kanssa samalle korkeudelle, jotta se ei uppoaisi terrainin sisään. Ensin valitsin mallin aseteista ja raahasin sen Hierarchyyn, jonka jälkeen muutin sen sijaintia terrainilla koordinaattien ja työkalujen avulla. Isonsin myös malleja, koska tuonnin jälkeen ne näkyivät lähes mikroskooppisen pieniltä. Myös mallin Inspector-valikosta valitsin päälle muutamia vaihtoehtoja, kuten Generate Colliders (kuva 23). Tämä on Unityssä aivan perusjuttuja, mutta on todella tärkeää että malleissa liikkeessa ei kävellä ns. seinien läpi.



KUVA 23. Esimerkki Mikpolin mallin asetuksista

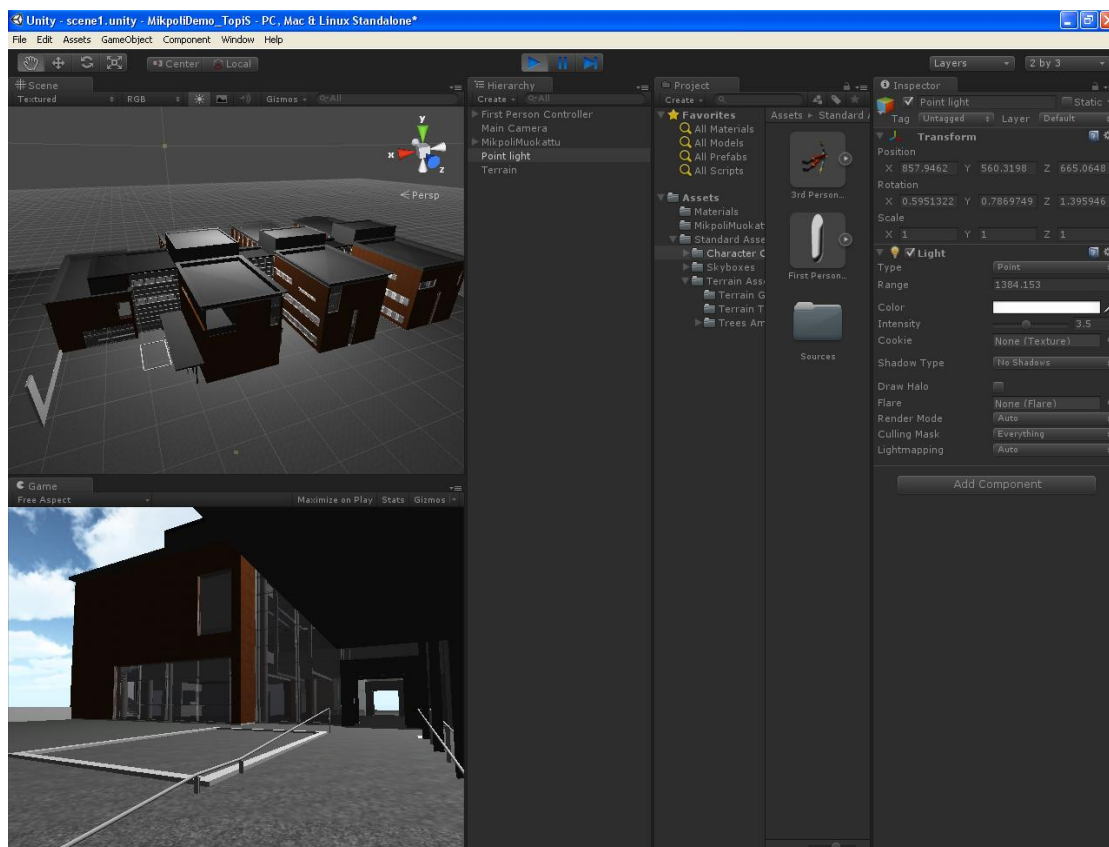
Tässä vaiheessa kun malli on saatu terrainille haluttuun sijaintiin tarvitaan ns. Character Controller. Kyseisen komponentin voi ladata projektiin aivan kuten muitakin asetteja, eli asset ikkunassa hiiren oikea painallus ja sieltä Import package → Character Controller. Kun komponentti on tuotu voi valita mm. ensimmäisen tai kolmannen persoonan näkymän. Itse olin valinnut Character Controller -paketin jo projektin perustamisvaiheessa annetuissa vaihtoehtoissa (kuva 21). Tähän demoan halusin mahdollisimman yksinkertaisen ja pelkistetyt ensimmäisen persoonan kontrollerin. Kontrollerin halusin tässä tapauksessa Mikpolin rakennuksen ulko-ovien eteen, josta myös myöhemmin Scene1:ssä koko demo lähtee liikkeelle. Esimerkiksi Scene2:ssa hahmokontrolleri tulee olemaan SketchUpissa luomani CampusIT:n mallin tilojen sisällä. Joitakin kontrollerin asetuksia (kuva 24) halusin hieman muokata, kuten liikkumisnopeutta eteen, sivulle ja taakse. Myös liukumisen, eli Sliding-vaihtoehdon halusin pois päältä kokonaan, jottei hahmo ns. huojuisi holtittomasti esimerkiksi rakennusten sisällä ahtaissa tiloissa ja käytävissä.



KUVA 24. Hahmokontrollerin liikehdinnän asetuksia

Hahmokontrollerin lisäämisen jälkeen lisäsin sceneen vielä hieman valaistusta. Valaistuksen saa Unityssä lisättyä valikosta GameObject → Create Other → ja sen jälkeen jokin valaistusvaihtoehdoista. Tässä työssäni käytän pelkästään valovaihtoehtoa Point Light, jonka avulla saa hyvin tarkkaa valaistusta tietylle alueelle, kuten vaikkapa rakennusten sisälle. Muita valovaihtoehtoja ovat Area Light, Directional Light ja Spotlight.

Kaikkien edellämainittujen vaiheiden jälkeen on demo saatu rakennettua sellaiseen tilanteeseen, että scene-näkymän sisällä voi liikkua, törmäyksen tunnistukset ovat asetettu ja muitakin asetuksia on muokattu jne. Kaikki vaiheet olen tähän osioon tähän asti kertonut Scene1-näkymän tilanteista. Scene2-näkymä on myöskin jo tässä vaiheessa samanlaisessa tilanteessa kuin Scene1. Onkin siis aivan turha kertoa jokaisen Scenen edistymisestä, mikäli prosessi tiettyyn pisteeseen asti on kaikissa melko pitkälti samanlainen.



KUVA 25. Scene1 ja näkymä demon alkutilanteesta

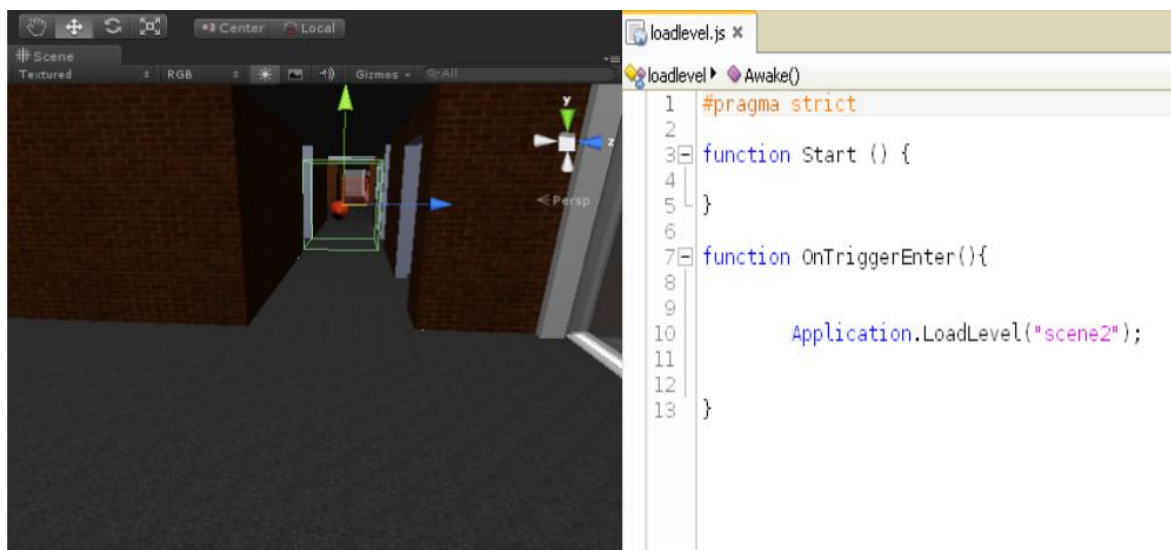
Tässä vaiheessa minulla on siis kaksi sceneä valmiina projektiani varten. Tulevaisuudessa scenejä voi tulla lisää, mutta tässä vaiheessa mennään näillä kahdella eteenpäin. Scene1:ssä on koko Mikpolin malli ja scene2:ssa CampusIT:n toimistotilat. Päämäärä on ainakin paperilla erittäin yksinkertainen. Tarkoitus on, että Scene1-näkymän sisällä pystyy tutkailemaan sekä kulkemaan vapaasti, ja tiettyyn sijaintiin saapuessa näkymä vaihtuisi automaattisesti Scene2-näkymään. Miten siis saankaan demoni toimimaan tällä tavalla?

Tarvitsen siis jonkinlaisen toimivuuden eri scene-näkymien välille. Unity3D-ohjelmassa Scenen vaihto voidaan toteuttaa useammallakin tavalla, mutta itselleni oli alusta asti melko selvää, että tässä projektissa tulisin käyttämään mahdollisimman yksinkertaista menetelmää. Tämä sen takia, että olen tällä hetkellä vasta kunnolla rupeamassa käsittämään Unityn erilaisia hienouksia, ja myöskin tämän opinnäytetyön tekemiseen käytettävä aika alkaa olla lopussa.

Halusin, että Scene1 alkaisi Mikpolin ulkopuolelta, ja kun päädytään riittävän lähelle CampusIT:n toimistotiloja näkymä vaihtuisi automaattisesti Scene2-näkymään. Google SketchUp -ohjelmassa tein joitakin tekstuurilisäyksiä myöskin Mikpoli-rakennuksen sisätiloihin. Ilman edes jonkinlaisia tekstuureja olisi rakennuksen sisäpuoli näyttänyt todella alkeelliselta, ja ennen kaikkea erittäin vaikealta hahmottaa. Tein siis Mikpolin käytävien seinille hieman tiiliseinää ensimmäiseen kerrokseen, jotta käytävät erottuisivat paremmin jne. Tämä toivottavasti auttaa edes hieman Scenen vaihtumiskohdan löytämisessä.

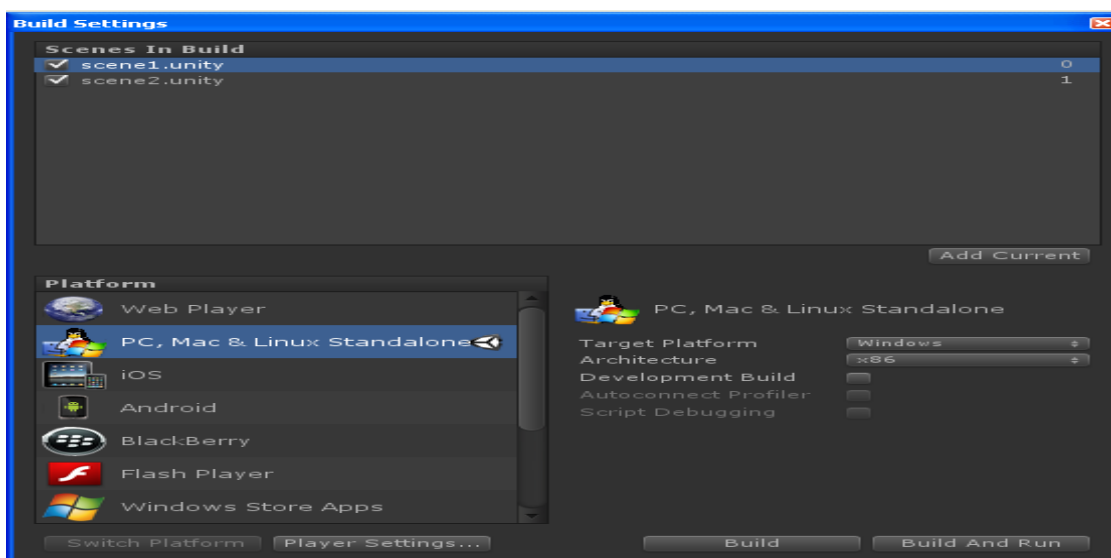
Itse Scenen vaihtoon käytin yhtä yleisimmistä menetelmistä Unityssä. Halusin sijoittaa kuution/cube tai alustalevyn/plane lähelle haluttua kohdetta. Tarkoituksena, että kun pelaajan hahmokontrolleri osuu valitsemaani elementtiin, niin Scene vaihtuu. Tässä demonstraatioissa päädyin Cube-vaihtoehtoon (kuva 26). Tässäkin prosessissa käytetään samaa jo aiemmissa kappaleissa mainittua konstia, eli GameObject → Create Other → sen jälkeen vaihtoehto Cube, eli kuutio.

Kuution sijoitin Mikpolin käytävälle CampusIT:n toimistotiloihin kääntyvän käytävän eteen. Kun kuutio oli luotu, se näkyi Hierarchy-näkymässä itsenäisenä objektina. Tärkeää oli kuitenkin raahata kuutio Mikpoli-malli objektiin Hierarchy-näkymässä, jotta toimivuus olisi Mikpoli-mallin sisällä. Kuutioon lisäsin myös Inspector-näkymässä Box Collider -vaihtoehdon ja Is Trigger -vaihtoehdon aktiivisiksi. Otin myös Mesh Collider -valikosta valinnan pois, että aikanaan Play-näkymässä kuutio olisi läpinäkyvä, eikä iso ruma möhkäle keskellä käytävää.



KUVA 26. Läpinäkyvä Cube-objekti Mikpolin käytävällä ja oikealla Scenen vaihtoon käytettävä yksinkertainen scripti

Tässä vaiheessa törmäysobjekti oli käytävällä ja nyt tarvitaan koodipätkä (kuva 26), jolla Scenen vaihto saataisiin laukaistua. Unity3D:ssä voi käyttää kolmea eri koodikieltä koodaamiseen. Vaihtoehdot ovat JavaScript, C# ja Boo, joka on Unityä varten kehitetty oma koodikieli. Itse valitsin kuitenkin JavaScriptin, koska se on minulle kaikkein tutuin näistä kolmesta vaihtoehdosta. Koodipätkä Scenen vaihtoon on äärimmäisen yksinkertainen ja toimiva. Käytännössä koodi vain käynnistää funktion, jossa hahmokontrollerin osuessa kuutioon triggeri/laukaisin käynnistää näkymänvaihdon seuraavaan Sceneen. Myös JavaScript -koodi pitää raahata aivan normaalisti haluttuun objektiin, tässä tapauksessa kuutioon.



KUVA 27. Build Settings

Tässä vaiheessa demonstraationi on jo varsin hyvällä mallilla. Kuitenkin, että projekti toimisi kunnolla tarvitsee vielä tarkastella yhtä tärkeää asiaa sen kannalta. Tämän hetkisessä projektissani on siis vain kaksi Sceneä, mutta toimivuuden kannalta on aivan sama onko niitä kaksi vai 200. Koodiin on jo kirjoitettu LoadLevel -kohtaan seuraavan Scenen nimi. Tämä pelkästään ei kuitenkaan riitä, vaikka kaikki olisi muuten kunnossa projektissa. File-valikon alta löytyvästä Build Settings -valikosta (kuva 27) on tärkeää käydä katsomassa, että kaikki tarvittavat Scenet ovat siellä ja oikeassa järjestyksessä. Mikäli Scene puuttuu Buildista voi sen raahata vaikkapa Asseteista oikealle kohdalle. Tämäkin on todella yksinkertainen asia, mutta välttämätöntä projektin oikein toimivuuden kannalta. Samaisesta ikkunasta myös rakennetaan ja viimeistellään projekti aikanaan haluttuun kuosiin.



KUVA 28. Scene2-näkymä toimivassa ja rakennetussa projektissa

Tässä vaiheessa olen testannut projektini ja todennut, että se toimii haluamallani tavalla. Scene-näkymä siis vaihtuu tiettyyn Mikpoli-rakennuksen kohtaan saavuttaessa. Projektista tuli melko raskas, ja scenen vaihdon välillä on pientä lataamista. Tämä on kuitenkin välttämätön paha, koska mallinnukset sisältävät melko paljon tekstuureja ja tavaroita.

Viimeinen tarvittava asia projektin loppuunviennin kannalta on vielä rakentaa se haluttuun muotoon. Tämä onnistuu jo aiemmin mainitun Build Settings -valikon kautta (kuva 27). Ensimmäisenä tarkistetaan, että kaikki halutut scenet ovat projektissa halutussa järjestyksessä. Tämän jälkeen työ on valmis viimeisteltäväksi.

7 PÄÄTÄNTÖ

Opinnäytetyön aihetta pohtiessani minulla oli aika ajoin melkoisia vaikeuksia keksiä mielekästä aihetta itselleni. Lopulta aiheekseni muodostui pelialan & pelintekovälineiden selvitys. En ollut kovinkaan paljon kiinnostunut pelkästään koodaamispainotteisesta työstä, tai esim. pelkästä tutkimustyöstä. Sen sijaan peliala oli aina kiinnostanut minua jollakin asteella, ja räätälöimäni aihe olikin minulle loppupeleissä oikein mieluista. Projektissa sain tavallaan yhdistellä kahta pienimuotoista tutkimusongelmaa. Ensimmäisenä Unity3D-ohjelman käyttäminen Suomessa, ja toisena Unityllä tehty pienimuotoinen demonstraationi Mikpoli-rakennuksesta.

Pelialan historiasta, pelinteko-ohjelmista yms. aiheista kirjoitettuani oli luvassa osuus Unityn käyttämisestä Suomessa. Millainen olikaan tutkimusprosessi? Tutkimusprosessi sinällään oli melko hankala ja tiedonsaannin valossa äärimmäisen nihkeä. Jo miettiessäni vartenotettavia kontakteja kyselylleni, tiesin tiedonsaamisesta tulevan vaikeaa. Useat kontaktit eivät vastanneet lainkaan ja toiset vastasivat usein erittäin pyöreästi kysymyksiini Unityn käytöstä heidän tarpeisiinsa. Sain kuitenkin sen verran tietoa, että pystyin esittämään niistä jokseenkin tutkimuspohjaisen raportin.

Tutkimuksen tärkeimpinä asioina pidin henkilökohtaisesti vastausviestejä Suomen pelialan kattojärjestö Neo Gamesin Koopee Hiltuselta, ja Unity3D:ltä saamaani vastausta Alex McCredien toimesta. Erityisesti Koopeelta saamani viesti yllätti minut positiivisesti, koska tieto jota häneltä sain oli varmasti luotettavaa Suomen pelialaa koskien. Eritoten edellämainittujen tahojen vastauksia raporttiini pidin tärkeinä, sillä Neo Games ja Unity olivat aihealueeni kannalta merkittäviä suuria tekijöitä.

Case-osuudessani olin tehnyt demonstraation Unity3D:llä käyttäen avuksi myös mm. Google Sketchup -ohjelmaa. Tarkoituksena oli luoda havainnoillistava demonstraatio Mikpoli-rakennuksesta ja siellä sijaitsevasta CampusIT-yrityksestä. Miten omasta

mielestäni onnistuin tässä projektiossa? Mielestäni onnistuin saamaan demooni ainakin kaikki itselleni asettamat minimivaatimukset. Tällaisia olivat mm. Onnistuneet mallinnukset Google Sketchupissa, niiden tuominen Unityyn sekä pieni toiminnallisuus näiden kahden mallinnuksen välillä. Demonstraatiosta ei tullut kovinkaan monipuolinen, mutta tämän opinnäytetyön puitteissa onnistuin siinä mielestäni ihan hyvin.

Mitä olisin voinut tehdä toisin opinnäytetyössäni? Pääsääntöisesti olen opinnäytetyöhöni tyytyväinen, mutta joitakin asioita olisin voinut tehdä tosiaankin toisin ja paremmin. Opinnäytetyön alusta alkaen työni yhtenä ongelmana oli, että minulla ei oikein ollut aivan sataprosenttista suuntaa mihin työni vaiheittain etenee. Tämä tuotti pieniä ongelmia työn uudelleen jäsentelyssä ja prosessin valmistuksen hitaudessa. Kuitenkin pystyin parsimaan työstäni sellaisen, että siinä on selkeä alku ja loppu. Unityn käyttäminen Suomessa -osuudessa olisin voinut olla monipuolisempi kyselyprosessin toteutuksessa. Esimerkiksi olisin voinut tehdä kyselyäni sähköpostin lisäksi mm. puhelimitse yms. Täten olisin saattanut saada hieman enemmän hyviä vastauksia, joita olisi voinut esittää julkki mm. jonkinlaisessa taulukossa tai muussa sellaisessa. Kuitenkin voin rehellisesti sanoa, vaikka vastausten ja tiedon määrä oli rajallista, niin sain kuitenkin haalittua tutkimukseen omasta mielestäni selkeän punaisen langan. Unity-projektiossa ongelmia tuottivat lähinnä pienimuotoinen kokemattomuuteni kyseiseen ohjelmaan, mutta aiheita tutkiessani ja siihen perehdyttyäni prosessi oli kuitenkin suhteellisen kivuton.

Tulevaisuudessa Unityllä tekemääni projektia voisi hyvinkin jatkaa eteenpäin monipuolisempaan muotoon. Alunperinhän tarkoitus oli kenties saada myöskin jotain toiminnallisuutta esim. CampusIT:n toimistohuoneessa sijaitseville tietokoneille. Tämä kuitenkin oli toistaiseksi itselleni vielä turhan monipuolinen prosessi. Ajattelinkin, että mikäli aihe kiinnostaa minua jatkossakin niin voisin tehdä tämän kaltaisia lisäyksiä projektiin. Myöskin projektiin voisi lisäillä muita huoneita, jossa sitten voisi olla omanlaistaan toiminnallisuutta.

Kaiken kaikkiaan koin tämän opinnäytetyöni tekemisen olevan itselleni erittäin opettavainen prosessi. Opin paljon siitä, kuinka tärkeää oikeasti on suunnitella tämän kaltaiset laajemmat työt alusta loppuun. On tärkeää miettiä tarkasti työn kulku, aihepiirit ja niiden sisältö ainakin jonkinlaisella alustavalla suunnitelmalla.

Kiinnostuin myös paljon enemmän pelialasta ja Unitystä ohjelmana. Tulevaisuus näyttää sen, tulenko jatkamaan tätä tai tämän kaltaisten Unityprojektien tekemistä tulevaisuudessa. Itselläni ainakin heräsi aivan uudenlainen kipinä kyseiseen aihepiiriin.

LÄHTEET

Donkey kong. 2013. Arcade-museum.com. WWW-dokumentti.
http://www.arcade-museum.com/game_detail.php?game_id=7610. Ei päivitystietoja.
 Luettu 3.8.2013.

Bad Piggies. 2013. Rovio.com. WWW-dokumentti.
<http://www.rovio.com/en/our-work/games/view/47/bad-piggies>. Ei päivitystietoja.
 Luettu 15.10.2013.

Blackman, Sue 2011. Beginning 3D Game Development with Unity. New York: Appress.

Brodkin, Jon. 2013. Slashdot.org. WWW-dokumentti.
<http://slashdot.org/topic/cloud/how-unity3d-become-a-game-development-beast/>.
 Päivitetty 3.6.2012. Luettu 10.9.2013.

Dusty consoles. 2013. Wordpress.com. WWW-dokumentti.
<http://dustyconsoles.files.wordpress.com/2013/03/famicom-and-nes.jpg>. Luettu 3.8.2013.

Game maker: HTML5 launch. 2011. Yoyogames.com. WWW-dokumentti.
<http://www.yoyogames.com/news/49>. Päivitetty 22.09.2012. Luettu 5.9.2013.

Game maker: Studio. 2013. Yoyogames.com. WWW-dokumentti.
<http://www.yoyogames.com/studio>. Ei päivitystietoja. Luettu 7.9.2013.

Gears of war images. 2013. Ignimgs.com. WWW-dokumentti.
http://assets2.ignimgs.com/2006/06/12/igears-of-wari-20060612102825835-1543663_640w.jpg. Luettu 29.8.2013.

Goldstone, Will 2011. Unity 3.x Game Development Essentials. Birmingham: Packt Publishing.

Id history. 2013. Idsoftware.com. WWW-dokumentti.

<http://www.idsoftware.com/business/history/>. Ei päivitystietoja. Luettu 13.8.2013.

Johnson, Erik. 2013. Mcvuk.com. WWW-dokumentti.

<http://www.mcvuk.com/news/read/microsoft-s-xna-framework-no-longer-in-active-development/0110259>. Päivitetty 1.2.2013. Luettu 21.8.2013.

Kosur, James. 2013. Socialnewsdaily.com. WWW-dokumentti.

<http://images.socialnewsdaily.com/wp-content/uploads/2013/07/Pong.jpg>. Luettu 2.8.2013.

Laud, Joshua. 2013. Ehow.com. WWW-dokumentti.

http://www.ehow.com/about_6436661_history-first-3d-video-games.html. Ei päivitystietoja. Luettu 11.8.2013.

Neogames esittely. 2013. Neogames.fi. WWW-dokumentti.

<http://www.neogames.fi/neogames/>. Ei päivitystietoja. Luettu 15.10.2013.

Nintendo history. 2013. Nintendo.co.uk. WWW-dokumentti.

<http://www.nintendo.co.uk/Corporate/Nintendo-History/Nintendo-History-625945.html>. Ei päivitystietoja. Luettu 3.8.2013.

PC plus. 2010. Techradar.com. WWW-dokumentti.

<http://www.techradar.com/news/gaming/the-evolution-of-3d-games-700995>. Päivitetty 11.7.2010. Luettu 11.8.2013.

Platforms. 2013. Unrealengine.com. WWW-dokumentti.

<http://www.unrealengine.com/en/platforms/>. Ei päivitystietoja. Luettu 29.8.2013.

Public relations. 2013. unity3d.com. WWW-dokumentti.

<http://unity3d.com/company/public-relations>. Ei päivitystietoja. Luettu 10.9.2013.

Reed, Kristan. 2009. History of first-person shooters. Videogamesdaily.com. WWW-dokumentti. <http://videogamesdaily.com/content/battlezone-history-of-fps-420.jpg>. Luettu 11.8.2013.

Scoopweb.com. WWW-dokumentti.

http://www.scoopweb.com/Game_Maker. Luettu 5.9.2013.

Selin, Jukka 2013. Unity perusteet. Mikkeli.

Seppala, Timothy J. 2009. Review – Dishwasher: Dead samurai punishes thumbs. Mlive.com. WWW-dokumentti.

http://blog.mlive.com/xbutton/2009/05/large_TheDishwasherDeadSamurai_Screenshot3.jpg. Luettu 22.8.2013.

Silverman, David. 2013. Gamedev.tutsplus.com. WWW-dokumentti.

<http://gamedev.tutsplus.com/articles/how-to-learn/how-to-learn-udk-the-unreal-development-kit/>. Päivitetty 28.11.2012. Luettu 29.8.2013.

Start with Unity3D. 2013. Blogspot.fi. WWW-dokumentti.

<http://startunity3d.blogspot.fi/2013/05/importing-models-from-google-sketchup.html>. Päivitetty 13.5.2013. Luettu 15.11.2013.

Super mario bros. 3. 2013. Gamefaqs.net. WWW-dokumentti.

http://img.gamefaqs.net/screens/1/8/3/gfs_28422_2_18.jpg. Luettu 3.8.2013.

The history of video game programming. 2001. Everything2.com. WWW-dokumentti.

<http://everything2.com/title/The+History+of+Video+Game+Programming>. Päivitetty: 23.10.2001. Luettu 2.8.2013.

Top 5 game development software. 2011. Gamenacho.com. WWW-dokumentti.

<http://gamenacho.com/top-5-game-development-software/>. Päivitetty 3.8.2011. Luettu 5.9.2013.

UDK frequently asked questions. 2013. Udn.epicgames.com. WWW-dokumentti.

<http://udn.epicgames.com/Three/DevelopmentKitFAQ.html>. Ei päivitystietoja. Luettu 29.8.2013.

What is the xna framework. 2006. MSDN blogs. WWW-dokumentti.

<http://blogs.msdn.com/b/xna/archive/2006/08/25/what-is-the-xna-framework.aspx>.

Päivitetty 25.8.2006. Luettu 21.8.2013.

Willoughby, Shane. 2012. Doom classic collection is PSN bound. Thegamingliberty.com. WWW-Dokumentti.

<http://www.thegamingliberty.com/wp-content/uploads/doom2-Classic.jpg>. Luettu 13.08.2013.

Wolfenstein 3d. 2013. 3drealms.com. WWW-dokumentti.

<http://www.3drealms.com/wolf3d/images/shot1.jpg>. Luettu 11.08.2013

LIITE 1.

Yksisivuinen liite

LIITE 2(1).

Monisivuinen liite