

GLADIAATTORI-PELIN PILOTTI

Koukkari Joonas

Opinnäytetyö

Tietojenkäsittelyn koulutus
Tradenomi (AMK)

2022

Tietojenkäsittelyn koulutus
Tradenomi (AMK)

Tekijä	Joona Koukkari	Vuosi	2022
Ohjaaja(t)	Ani Ruusila		
Työn nimi	Gladiaattori-pelin pilotti		
Sivu- ja liitesivumäärä	17		

Opinnäytetyö on UnityEnginellä tehty roguelike-tietokonepelin pilotti nimeltään Gladiaattori. Opinnäytetyön tavoitteena oli opetella UnityEnginen käyttöä pelin teossa ja käyttää opittuja taitoja pelin kehittämiseen.

Työssä käytiin läpi mitä roguelike tarkoittaa, sen historiaa ja mitä tunnettuja roguelike-pelejä on olemassa. Tämän kaiken lisäksi kerrottiin, mitä ominaisuuksia pelin tekemiseen tarvitaan, kuten grafiikkaa, teemoja, karttoja, mekaniikoita, sekä mitä vaihteita pelin tekoon kuuluu, kuten ideointivaihe, suunnitteluvaihe, toteutusvaihe eli ohjelmointivaihe ja testausvaihe. Opinnäytetyössä kerrottiin, mitä apua Unity Engine tarjoaa ohjelmoinnissa tuleviin ongelmiin ja haastaviin tilanteisiin. Tähän kuuluvat Unity Asset Store ja UnityEnginen visuaalisuus ja sen ohjeet.

Suurimpina haasteina oli pelin toteutus yksilötyönä ja lyhyessä ajassa. Tästä syystä peli oli toteutettu pilottina, eikä kokonaisuutena pelinä ja se ei täyttänyt kaikkia kokonaisen pelin vaatimuksia. Kokonaisen peli yleensä tehdään tiimeissä ja aikaa käytetään pelin kehittämiseen useampi vuosi.

Gladiaattori-pelin pilotti on tehty ja suunniteltu yksinkertaiseksi ja helposti jatko-kehittäväksi. Tästä syystä ajan, kiinnostuksen ja resurssien salliessa sitä tul- laan jatkokehittämään eteenpäin tulevaisuudessa. Opinnäytetyön Gladiaattori- pelin pilotti toimii myös työelämässä ohjelmoinnin osaamisen näytteenä ja tuo lisäarvoa työtä hakiessa. Tästä syytä sen kehittäminen on hyödyllistä.

Avainsanat
Muita tietoja

Ohjelmointi, Roguelike, Tietokonepeli, UnityEngine
Työhön liittyy ohjelmoitu tietokonepeli

Degree Programme in Business In-
formation Technology
Bachelor of Business Administration

Author	Joona Koukkari	Year	2022
Supervisor	Ani Ruusila		
Subject of thesis	Gladiator game demo		
Number of pages	17		

This thesis is a roguelike computer game demo called Gladiator. It was made with UnityEngine. The goal of writing this thesis was to learn how to use the UnityEngine in making a game and use the skills learned to develop the game.

The thesis goes through what is roguelike and the history of roguelike and what are the prevailing popular roguelike games. This thesis describes features about making games, including graphics, theme, maps and mechanics, that has four stages which are ideation, planning, implementing and testing. The thesis further reviewed, how UnityEngine helps game creators to make the games and where to find ways to solve problems encountered during game development. This includes Unity Asset Store and UnityEngine's visual user interface and instructions.

Biggest problems faced were making the game alone in a short amount of time, because this Gladiator game was demo version and does not have full game requirements. Usually the game is made in teams and takes several years to develop.

Gladiator game demo is made and designed in simple and easy to further develop. With enough time, resources and the Interest, the development of this game is possible in the future. This thesis will help also when applying for job, because it can be used as a sample of competence. Therefore, its developments would be valuable in future.

Key words Programming, Roguelike, Game, UnityEngine
Special remarks The thesis includes a programmed computer game.

SISÄLLYS

1 JOHDANTO	5
2 UNITYENGINE JA UNITY ASSET STORE	6
3 ROGUELIKE.....	7
3.1 Mikä on roguelike ja sen historia.....	7
3.2 Roguelike-pelejä	8
4 PELIN TOTEUTUS	9
4.1 Pelin ideointi ja suunnittelu	9
4.2 Pelin mekaniikka ja grafiikka	9
4.3 Pelin teema, tarina ja kartta	10
4.4 Pelin ohjelmoinnin vaiheet ja testaus	11
5 GLADIAATTORI PILOTTIPELINÄ	12
6 POHDINTA.....	14
LÄHTEET	16

1 JOHDANTO

Opinnäytetyöhön on toiminnallinen opinnäytetyö, jossa tarkoituksena on ohjelmoida UnityEnginellä Gladiaattori pelin pilotti, joka on yksinkertainen, viihdyttävä, toimiva, ja roguelike-genreen sijoittuva tietokone peli, hyödyntäen jo opittuja ohjelmoinnin taitoja ja niitä samalla kehittäen. Selvitän, mitä taitoja pelin tekemiseen tarvitaan. Opinnäytetyössä kerrotaan mikä on roguelike ja mistä se on saanut alkunsa, mitä muuta merkittäviä käänteitä sen historiassa on tapahtunut sekä mitä tunnettuja roguelike-pelejä on tehty. Tämän lisäksi kerrotaan mikä on UnityEngine, mitä apuja se antaa pelin kehittämiseen ja mitä asioita pelin toteutukseen kuuluu.

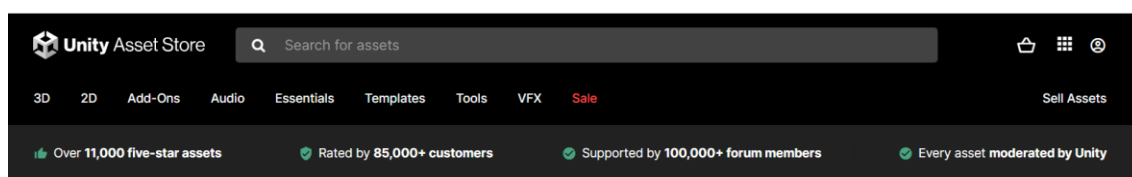
Työnantajat katsovat aiempia töitä positiivisesti, koska se näyttää hakijan osaamisen. Tästä syystä hakijan on hyvä panostaa näyttille tuotavien töidensä selkeyteen, koska työnantaja näkee siitä hakijan ammattitaidon. (Laakkonen 2020; Duunitori 2013.)

Opinnäytetyössä kerrotaan, miten Gladiaattori-pelin pilotti luotiin ja mitä pelinosia ja vaiheita siihen kuului. Lisäksi pohditaan sen teossa huomattuja suurimpia ongelmia ja haasteita ja miten ne vaikuttivat työnkulkuun sekä sitä, miten ne olisi voinut välttää. Tämän lisäksi pohditaan, miltä Gladiaattori-pilotin tulevaisuus näyttää sekä kuinka helposti sitä pystytään jatkamaan ja onko muita jatkokehityssuunnitelmia. Lopuksi pohditaan, kuinka paljon pilotin rakentaminen opetti ja kuinka hyvin se pysyi tavoitteissa.

2 UNITYENGINE JA UNITY ASSET STORE

UnityEngine on suosittu pelimoottori, koska se on ilmainen ja siinä on tarjolla paljon ohjeita ja apua pelien tekemiseen. Sen avulla on mahdollista tehdä peli aivan alusta aina julkaisuun asti. Käytön helpottamiseksi UnityEnginestä löytyy paljon tutoriaaleja, ohjeita ja visuaalinen käyttöliittymä. Tämän lisäksi UnityEnginen yhteisöstä löytyy apuja eri ongelmiin ja tilanteisiin. UnityEnginen suosiosta kertoo se, että yli 50 % kaikista maailman peleistä on tehty sillä. UnityEnginen suosiota myös kasvattaa sen tarjoama mahdollisuus ohjelmoida pelejä yli 25 eri käyttöjärjestelmälle. (Unity 2022a.)

Grafiikka, musiikki ja äänet peliin on hyvä hankkia Unity Technologiessin omasta Unity Asset Storesta (Kuva 1). Pelin kaikkien osien ja ominaisuuksien tekeminen vie paljon aikaa. Tästä syystä Unity Asset Storen tarjoamien osien avulla pääsee nopeasti testaamaan ja toteuttamaan pelin toiminnallisuuksia. Sen avulla myös näkee nopeammin pelin idean kannattavuutta jatkoa ajatellen. Pelin alkuvaiheen testaaminen näkyy etenkin pelin kehityksessä ja haasteiden ratkaisuisa säästetyllä ajalla. Tämän avulla pääsee nopeammin pelin sääntöjen, ulkonäön, kontrollien ja pelin muiden tärkeiden osien suunnitteluun. Ainakin alkuvaiheessa pitää miettiä mitkä osat kannattaa tehdä itse ja mitkä osat ladataan kaupasta valmiina paketteina. Tähän vaikuttaa suuresti aikataulu ja ohjelmoijan omat taidot. (Unity3d 2022; Unity 2022b.)



Kuva 1. Unity Asset Store (Unity Asset Store 2022)

Unity Asset Store on Unityn Technologiesin omistama kauppa, joka on linkitetty UnityEngineen. Tämän avulla asiakas saa ostoksensa suoraan UnityEngineen. Kaupasta löytyy ilmaisia ja maksullisia grafiikoita, ohjelman pätkiä, musiikkia, animaatioita ja muita hyödyllisiä ohjelmointiin tarvittavia osia. (Unity3d 2022.)

3 ROGUELIKE

3.1 Mikä on roguelike ja sen historia

Roguelike-genreen kuuluvat pelit ovat saaneet vaikutteita alkuperäisestä Rogue pelistä vuodelta 1980. Tämä peli on tunnettu satunnaisesti luodusta kartoistaan ja vihollisistaan, ja tavarat ovat satunnaisesti luotuja, esimerkiksi taikajuomat ovat värikoodattuja, mutta niiden vaikutus on eri jokaisella peli kerralla. Roguessa taistelu tapahtuu välittömästi vuoropohjaisesti. Siinä on permadeath-ominaisuus, mikä tarkoittaa sitä, että jos pelaajan hahmo kuolee, silloin kaikki pelitallennuksetkin katoavat ja pelaajan on aloitettava alusta. Pelissä pelaaja saa vapaasti liikkua kartalla ja taistella vihollisia vastaan sekä kehittää hahmoaan jatkuvasti vaikenavia vihollisia vastaan. Roguessa ei ole grafiikkaa, vaan kaikki grafiikat ovat yksinkertaisilla symboleilla tehty. Sitä pelataan pelkällä näppäimistöllä, eikä näppäinyhdistelmiä tarvittu. (Roguebasin 2020; Spencer 2013.)

Vuonna 1987 julkaistiin Nethack, joka perustuu Dungeons&Dragons-roolipeliin. Nethack oli edeltäjäänsä Rogueta paljon monimutkaisempi. Siinä oli grafiikat sekä enemmän toimintoja mihin näppäimistö antaa mahdollisuuden. Nethack toivutena peliin esimerkiksi hahmoluokat, kaupan, lemmikit, yli 100 erilaista vihollista sekä paljon muuta. Tämän takia myös sanasto laajeni aikaisempaan Rogueen verrattuna. Nethackissä oli aikoinaan korkea oppimiskynnys. (Spencer 2013.)

Vuonna 2002 julkaistiin DoomRL, koska roguelike-genreen oli vaikea päästä mukaan sen monimutkaisuuden takia. DoomRL laski kynnyksestä huomattavasti, sillä se lisäsi roguelike-genreen tyhmänhauskoja ominaisuuksia. DoomRL tehtiin vain 9 päivässä, mutta sitä päivitettiin 11 vuotta. Siitä poistettiin paljon roguelike-ominaisuuksia, kuten satunnaisesti luodut tavarat, ja sitä pystyi pelaamaan vain hiiressä. DoomRL uskalsi ensimmäisenä tuoda peliin roguelike-perusominaisuuksia yhdistäen siihen muiden genrejen ominaisuuksia. Tämä on ollut suosittua nykyään roguelike-genressä. DoomRL ei ollut suuri peli, mutta siinä tuotiin alkuperäisen Doomien ominaisuuksia tehden siitä lähes yhtä hauskan. (Spencer 2013.)

3.2 Roguelike-pelejä

Vampire Survivors on peli, jossa pelaaja yrittää päihittää vihollisia, selvitä mahdollisimman pitkään ja kerätä kultaa, kun samalla viholliset koittavat päihittää pelaajan. Tämä on pieni toimiva roguelike-peli, eikä pelaaja voi piiloutua vihollisilta. (Poncle 2022.)

Toinen nykyään tunnettu, roguelike-peli on The Binding of Isaac, jossa hahmo seikkailee satunnaisesti luoduissa huoneissa. Tämän lisäksi siinä on satunnaisesti luotuja vihollisia, ja seuraavaan huoneeseen pääsee, kun edellinen huone on tyhjennetty vihollisista. Hahmo kehittyy keräämällä tavaroita, joita löytyy satunnaisesti uusista huoneista. Pelaaja löytää pomohuoneista varmuudella satunnaisesti luotuja tavaroita, joiden avulla pärjää seuraavan kerroksen huoneissa. Pelaajan kuollessa hän joutuu aloittamaan pelin alusta. Jokainen pelikerta on erilainen, koska huoneet, tavarat, viholliset ja pomot ovat aina satunnaisesti valittuja. Tästä kertoo esimerkkinä se, että pelissä on yli 600 erilaista tavaraa. Tämä tuo mahdollisuuden pelaajalle käyttää erilaisia esineyhdistelmiä jokaisella pelikerralla. (Saltzman 2021.)

Hades on saanut vuodenpelejä 2020 palkinnon. Tässä roguelike-pelissä pelaaja yrittää paeta helvetistä. Viholliset ja palkinnot ovat satunnaisesti luotuja. Pomot ja muut aktiviteetit ovat aina samanlaiset. Pakoyritysten välillä pelaaja saa tehtäviä ja mahdollisuuden vahvistaa itseään seuraavaa pakoyritystä varten. Jokainen pakokerta on erilainen, sillä pelaaja on joka kerta hieman vahvempi ja valmiimpi pakenemaan. Tästä syystä kuolematkaan eivät pelissä haittaa, koska jokaisella pakoyrityksellä saa kokemusta ja tietoa pelin mekaanikoista. Hades on tarinapohjainen roguelike-peli. Tämän takia jokaisella pakokerralla pomot käyttäytyvät eri tavalla. Tähän käyttäytymiseen vaikuttavat yritysten määrä, pelattavan hahmon ikä ja se, montako kertaa on aiemmin voittanut kyseisen pomon. (Saltzman 2021.)

4 PELIN TOTEUTUS

4.1 Pelin ideointi ja suunnittelu

Pelin ensimmäinen idea on muun muassa teema tai mekaniikka, jota haluaa käyttää. Teemaa tai mekaniikkaa saatetaan kehittää jopa muutama päivä miettimällä sitä päässä, ja tähän ideaan voi lisätä ja poistaa ominaisuuksia. Idean voi kirjoittaa paperille, kun se kuulostaa mieleiseltä. Idea pysyy ideana niin kauan, kun siinä ei ole selkeää perusmekaniikkaa, perussääntöä, eikä selkeää kehystä. Myöhemmin, kun idealla on selkeä perusmekaniikka, perussäännöt ja selkeäkehys, ja siinä pitäisi olla nähtävissä selkeä ymmärrettävä alku, keskiosa ja loppu, idea muuttuu peliksi. (Malmioja 2020a.)

Pelin suunnittelu voidaan jakaa kolmeen osaan, perusmekaniikkaan, kerrontaan ja vuorovaikutteisuuteen. Perusmekaniikkaan kuuluu pelin perussäännöt ja toiminnallisuus. Kerrontaan kuuluu pelin säännöt ja pelimaailma, joihin pelaaja jouuu pelatessaan peliä. Ne ohjaavat pelaajaa eteenpäin ja näyttävät pelin maailmaa ja tarinaa. Vuorovaikutus on kaikki toiminnallisuudet, seuraukset ja tapahtumat pelissä. Näihin pelaaja voi reagoida omalla valitsemallaan tavalla ja pelissä asioita muuttuu pelaajan tekemiin valintoihin nähden sopiviksi. (Manninen 2007, 61–63.)

4.2 Pelin mekaniikka ja grafiikka

Pelimekaniikat ovat toimintoja ja sääntöjä, joita noudattamalla tai käyttämällä pelaaja voi olla vuorovaikutuksessa pelimaailmaan kanssa. Pelissä on yleensä useita mekaniikoita ja niiden yhdistelmiä. Mekaniikoita on mahdollista yhdistää ja jakaa osiksi mielensä mukaan. Näitä mekaniikoita voi myös käyttää toisissa peleissä, koska tällä hetkellä on todella paljon pelejä olemassa ja täysin uuden mekaniikan keksiminen on tämän takia vaikeaa. (Sicart 2008; Malmioja 2020.)

Mekaniikoiden vanha nimitys on Quick time events, ja ne eivät roguelike-genren alkuvaiheessa antaneet paljokaan armoa pelaajan epäonnistumisille. Mekaniikat aloittivat pelaajan pelin alusta tai vaikeuttivat pelaajan etenemistä todella paljon.

Nykyään mekaniikat ovat armollisempia pelaajaa kohtaan ja epäonnistuminen lisää ainoastaan muutaman esteen, jotka hidastavat pelaajan etenemistä. (Suominen 2020.)

Grafiikat tarkoittavat pelin kuvia, hahmoja, taustoja ja pelin visuaalista näkyvyyttä. Grafiikka koostuu kuvista ja vastakohtana sille on teksti, joka koostuu taas merkeistä. Tietokoneella tehtyä grafiikkaa on kaksi- tai kolmiulotteista eli 2D- tai 3D-grafiikkaa. Ensimmäiset koneet käyttivät mustavalkoista 2D-grafiikkaa. 3D-grafiikkaa alettiin käyttämään vasta 1990-luvulla ja nykyään niitä käytetään miljoonissa peleissä. Ensin grafiikassa käytettiin 16 tai jopa 256 väriä, mutta nykyään on keksitty 3D-näytönohjain, jonka avulla tietokone pystyy käyttämään miljoonia värejä suuremmissakin peleissä ilman ongelmia. (Techterms 2009.)

4.3 Pelin teema, tarina ja kartta

Pelin teema on se, miltä peli näyttää, eli sen graafiset ominaisuudet ja se, mitä tapahtumallisia sääntöjä pelissä käytetään. Esimerkiksi teema voi olla maanviljely tai kauhu. (Malmioja 2020b.) Teema kertoo pelin aiheen, joka voi olla hyvin tarinaan sijoittuva tai abstrakti aihe. Tämän avulla pelaaja voi etsiä itselleen häntä kiinnostavia ja sen kaltaisia pelejä, mistä hän pitää. (BoardGaming 2022.)

Pelin tarina tarkoittaa pelin kulkua ja karttaa, jossa pelaaja seikkailee. Siihen ei vaikuta pelin mekaniikalliset säännöt. Tarina on enemmän pelin juoni, eli mitä tapahtumia pelissä on. (Malmioja 2020b.) Pelin kerronta tuo pelaajalle tunteen, että hän on osa tarinaa ja siihen sisältyy pelin juoni, äänet, musiikit, tunnelma ja dialogit. Pelaaja voi vaikuttaa tarinan kulkuun omilla valinnoillaan ja pelaamisellaan. (Stargame 2018.)

Kartta on maailma, jossa pelaaja pelaa. Kartta on hyvä, jos pelaajalla on useampi reitti ja useampi mahdollisuus läpäistä kenttiä. Kartta osaa kontrolloida pelaajan näkökenttää, sillä se pitää pelaajan mielenkiinnon pelissä. Tämä näkyy etenkin hiiviskely ja ensimmäisen persoonan ampumapeleissä, joissa pelaaja pitää jatkuvasti mieltä paras vaihtoehto läpäistäkseen pelin. Esimerkiksi hyvinä ensimmäisen persoonan ampumapelin karttoina ovat CSGO pelin usea kartta, joissa on kolme pääreittiä, mutta niiden lisäksi on useita sivureittejä ja mahdollisuuksia yllättää vastustaja (Wagar 2018.)

4.4 Pelin ohjelmoinnin vaiheet ja testaus

Pelin koodausvaiheessa eli työvaiheessa on hyvä käyttää ilmaista pelimoottoria, koska se säästää paljon aikaa. Oman pelimoottorin teko ei ole helppoa ja vie paljon aikaa. Tämä vaatii myös vuosien erikseen opiskelun ennen ohjelmoinnin aloittamista. Pelistä on tärkeää saada nopeasti aikaiseksi prototyyppi, koska sen avulla näkee pelin potentiaalin. Tämä avulla pääsee testaamaan ja näkemään pelin käytännön ongelmat ja virheet, ja peliä testaamalla ne löytyvät helpommin. On myös hyvä ottaa huomioon mitkä kaikki osat kannattaa tehdä itse ja mitkä osat kannattaa ostaa muualta, esimerkiksi pelin äänet, animaatiot, grafiikat ja skriptit. (Holotna Suomi 2018; Holotna Suomi 2019.)

Valmiin ison pelin tekemisessä on yleensä useita henkilöitä. Jokaisella on oma osaamisalueensa, joten kaikkien ei tarvitse opiskella ja osata kaikkea. Pelaajaa pitää muistaa palkita tarpeeksi erilaisilla tavoilla suorituksistaan, jotta pelaajalle tulee onnistumisen tunne. Jotta peli ei kävisi tylsäksi, niin siihen pitää tuoda vaihtelevuutta, kuten peli muuttuessa sitä pelatessa. Tähän pystyy vaikuttamaan uusilla mekaniikoilla, teemoilla, ympäristöllä, vihollisilla ja vaihtoehtoisilla poluilla. Ei kannata toistaa samoja mekaniikoita, kenttiä tai muita jo aiemmin suoritettuja asioita jatkuvasti. Satunnaisesti muuttuvat asiat myös tuovat vaihtelua uusille pelikerroille. Musiikki, teema ja tarina ovat tapoja tuoda tunnelmaa pelille. Hyvin tehtynä peli tuo pelaajalleen hyvän kokemuksen. (Holotna Suomi 2018; Holotna Suomi 2019.)

Pelintestaus on tärkeää, koska sen avulla peliä voi hienosäätää ja sen huonot ja hyvät puolet, ongelmat ja virheet tulevat esille. Pelintestausta on tehtävä pelin tekovaiheessa ja loppuvaiheessa. Tekovaiheessa pelintestaus on rakennetestausta, jossa katsotaan, että kaikki toimii niin kuin pitää, ja peliä voi pelata sekä hienosäätötestausta, jossa hienosäädetään pelin yksityiskohtia ja poistetaan ongelmia ja virheitä. Loppuvaiheessa peliin on hyvä tehdä hajotustestaus, jossa testaaja koittaa rikkoa pelin sekä käytettävyydestä, jossa testataan pelin kokonaisuutta ja nähdään, miten kokematon pelaaja etenee peliä pelatessaan ilman ulkopuolista apua. (RollID6Games 2022.)

5 GLADIAATTORI PILOTTIPELINÄ

Tämän opinnäytetyön pelin idea on tehdä roguelike-genreen liittyvää tietokonepelin pilotti (Kuva 2). Gladiaattori-pelissä pelaajan hahmo on gladiaattori, jonka aikeena on taistella tiensä vapauteen. Tämä tie ei ole helppo ja suora, koska pelaajan hahmoa väijyy viholliset jokaisen kulman takana. Matkalla on mahdollisuus löytää vihollisten ja muiden ei-pelaajahahmojen joukosta historiasta tuttuja nimiä. Tämän lisäksi on pelaajalla mahdollisuus löytää esineitä ja aseita helpottaakseen pelin etenemistä. Pelin vaikeus kasvaa taistelun aikana.



Kuva 2. Gladiaattori-pelin pilotti

Gladiaattori-pelin pilotissa on kolme karttaa, joita ovat areena, kaupunki ja portti. Areena on normaalin gladiaattoriareenan näköinen, kaupunki keskiaikainen Rooman kaupunkia muistuttava kaupunki ja portti on kaupungin tarkennettu osa, jossa vihollisten pomo ilmestyy ja jonka jälkeen peli on suoritettu. Peli aloitetaan areenalta, jossa pelaaja taistelee vihollisia vastaan ja voitettuaan viholliset hän siirtyy kaupunkiin. Kaupungissa pelaajaa vastaan tulee vartijat ja pelaajan pitää taistella uudestaan vihollisia vastaan, kunnes pääsee portille kohdatakseen siellä vartijoiden pomon. Pomon päihitettyään pelaaja pääsee portista pihalle (Kuva 3).



Kuva 3. Gladiaattori pilotin kenttä porttien ulkopuolelta.

Pelaaja pääsee kentän läpi, kun kaikki viholliset on kukistettu kyseisestä kentästä ja sen jälkeen seuraava kenttä aukeaa. Jos pelaaja epäonnistuu päihittämään viholliset kentästä, pelin voi aloittaa alusta ja tällöin viholliset, esineet ja tavarat vaihtuvat. Tämä satunnaisuus muuttaa pelin kulkua ja antaa vähän erilaisia kokemuksia.

Pelaaja pystyy vuorovaikuttamaan pelin kanssa monella eri tavalla. Pelaaja pystyy päihittämään vihollisia, jotka koittavat estää pelaajan etenemistä. Vihollisia päihittämällä pelaaja kehittyy ja pelin vaikeusaste kasvaa. Tämä on tärkeää, koska pelaajan pitää taistella yhä vaarallisempia vihollisia vastaan pelin edetessä. Viholliset ilmestyvät satunnaisesti kartan oikeaan ja vasempaan laitaan (Kuva 4) ja lähtevät juoksemaan kohti pelaajaa.

```
private IEnumerator spawnEnemy(float aika, GameObject enemy){
    //wait until spawnrate time is over
    yield return new WaitForSeconds(aika);

    //random spanpoint
    int rP;
    rP = Random.Range(0,6);
    //Debug.Log(rP);

    //Spawn enemy in randomized spanpoint and use enemy prefab
    GameObject newEnemy = Instantiate(enemy, spawnpoint[rP].transform.position, Quaternion.Euler(0,0,0)) as GameObject;
    //GameObject newEnemy = Instantiate(enemy, new Vector3(Random.Range(-5f, 5), Random.Range(-6f, 6f), 0), Quaternion.identity);

    //Start again enemy spawner using same values
    StartCoroutine(spawnEnemy(aika, enemy));
}
```

Kuva 4. Vihollisten luonti satunnaisesti luonti pisteisiin.

6 POHDINTA

Opinnäytetyö alkoi tarvittavan tiedon etsinnällä, ja jatkui tämän tiedon kirjoittamisella ylös opinnäytetyön dokumenttiin. Tämän jälkeen toiminnallinen osuus alkoi, jossa ohjelmoin Gladiaattori-pelin pilotti, ja samalla korjailin opinnäytetyön dokumenttia. Ohjelmointi osuudessa aluksi mietin Gladiaattori-pelin pilotille ideaa ja suunnittelin mitä siinä tapahtuu. Tähän vaikutti suuresti grafiikka mitä löysin ja josta ostin lisenssin. Tämän jälkeen aloin ohjelmoimaan mekaniikoita, ja tuomaan Gladiaattori-pelin pilottia pelattavaksi. Tarkoituksenani oli saada mahdollisimman nopeasti pelattava ja toimiva tuotos, sillä sen avulla pääsen testaamaan nopeasti pilottia. Lopuksi vielä testailin pilottia eri tilanteissa, ja korjailin vastaan tulevia virheitä.

Gladiaattori-pelin pilotti ei ole valmis peli, mutta siinä on potentiaalia jatkolle. Sitä on helppo jatkokehittää ja saada monipuolisemman. Siihen on todella paljon ideoita, jota ei ole ehtinyt ohjelmoimaan, esimerkiksi, erilaisia tavaroita, lisävoimia, pomoja, kartoja, vihollisia ja puolustusmekaniikka. Tästä syystä tulevaisuudessa Gladiaattori pilotista tehdään peli, jos on tarpeeksi aikaa, innostusta ja muita resursseja riittää. Gladiaattori-pelin pilottia on hyödyllistä jatkokehittää sillä työtä hakiessa sen avulla voi näyttää omia ohjelmoinnin taitoja.

Kaikki Gladiaattori-pelin pilotin tavoitteet eivät toteutuneet siten, kuin alussa oli suunniteltu. Peli kuitenkin pysyi alkuperäisen idean mukaan yksinkertaisena ja helposti kehitettävänä roguelike-pelinä. Pelistä piti karsia ideoita pois, alla mainittujen suurimpien haasteiden takia, jotta sen tekeminen pysyy aikataulussa.

Suurin haaste, jota pelin tekovaiheessa ilmeni, oli aika. Vaikka idea ei ollut iso ja vaikka siinä oli vain muutama ominaisuus, siinä oli silti liikaa tekemistä aikaan nähden. Lopputulos oli paljon pienempi, mitä ideointi ja suunnittelu vaiheessa oli ajateltu. Suurimpina syinä tähän oli ideointivaiheessa tullut liian suuri mielenkiinto ja innostus peliä ja sen jatkokehitystä kohtaan, ja tein pelin yksin, kun normaalisti pelinteossa on mukana useampi eri osa-alueen ammattilainen.

Toinen haaste oli löytää teemaan liittyviä yhteensopivia grafiikoita. Grafiikoiden tekeminen itse olisi vienyt liikaa aikaa, ja opinnäytetyön pääpainona oli ohjel-

mointi. Vaihtoehtoja oli vähän ja suurin ongelma oli, ettei kaikkia grafiikoita löytynyt 2D:nä tai 3D:nä, vaan näitä löytyi sekaisin. Tämä takia pelistä tuli 2D, koska siinä löytyi muutama yhteen sopiva hahmo ja kartta, joista kartta ei ollut ihan sitä mitä haettiin.

Näihin edellä mainittuihin haasteisiin pystyy varautumaan, jos suunnittelee mallilla ja tiedostaa paljonko aikaa menee eri osien tekemiseen. Avuksi on, jos tekee ryhmätyönä pelin ja jokaiselle jaetaan omat vastualueet. Tämä edistää pelin tekoa, mahdollistaa itsetehdyt grafiikat ja antaa pelille enemmän ominaisuuksia. Yhteistyö ryhmällä täytyy olla toimivaa ja kommunikointi ryhmäläisten välillä täytyy olla hyvä.

Gladiaattori-pelin pilotin ohjelmoiminen opetti paljon enemmän, kuin aluksi olisi voinut ajatella. Se opetti roguelike-pelin ohjelmointiin tarvittavaa tietoa, vastuullisuutta ja aikataulutusta. Ohjelmoin pilottia yksin ja sen takia piti suunnitella aikataulu ja vastuut eri tavalla kuin ryhmässä. Se opetti myös ongelmanratkaisua niissä haasteissa, mitä tuli ohjelmoinnissa ja koko opinnäytetyön teossa vastaan. Tämän lisäksi, tietoa etsiessä, siinä oppi gladiaattoritaisteluista ja itse roguelikestä. Unitystä itse en oppinut niin paljoa, koska kyseinen ohjelma oli jo ennestään tuttu. Se toi uusia tapoja sillä ohjelmalla työskentelyyn, koska tein pelin pilottia yksin ja piti huolehtia kaikesta itse, eikä voinut kysyä kaverilta neuvoa tai mielipiteitä.

LÄHTEET

BoardGaming 2022. Game Themes. Viitattu 1.5.2022
<https://boardgaming.com/themes>.

Duunitori 2013. Mitä kannattaa ottaa mukaan työhaastatteluun. Viitattu 1.5.2022
<https://duunitori.fi/tyoelama/mita-mukaan-tyohaastatteluun>.

Holotna Suomi 2018. Kuinka tehdä oma peli. Viitattu 6.5.2022
<https://www.youtube.com/watch?v=mot1-CvfR1A>.

Holotna Suomi 2019. Kuinka tehdä hyvä peli. Viitattu 6.5.2022
<https://www.youtube.com/watch?v=Rzw7PN0lqIE>.

Malmioja, K. 2020a. Lautapeli suunnittelun askeleita 2 – Mikä on idea, mikä on peli? Lautapeliopas 7.2.2020. Viitattu 26.10.2022
<https://www.lautapeliopas.fi/artikkelit/pelisuunnittelu/lautapelisuunnittelun-askeleita-2-ideapeli/>.

Malmioja, K. 2020b. Lautapeli suunnittelun askeleita 4 – Peli = tarina + teema + mekaniikka. Lautapeliopas 7.2.2020. Viitattu 13.5.2022
<https://www.lautapeliopas.fi/artikkelit/pelisuunnittelu/lautapelisuunnittelun-askeleita-4-peli-tarina-teema-mekaniikka/>.

Manninen, T. 2007. Pelisuunnittelijan käsikirja ideasta eteenpäin, Tallinna-kustannus OY Rajalla.

Poncle. 2022. Vampire survivors. Viitattu 1.5.2022 <https://poncle.itch.io/vampire-survivors>.

Roguebasin. 2020. What a roguelike is. Viitattu 1.5.2022
http://www.roguebasin.com/index.php?title=What_a_roguelike_is.

RollD6Games. 2022. Pelitestaus. Viitattu 28.4.2022
<https://www.rolld6.com/pelikehittajalle/pelitestaus/>.

Saltzman, M. 2021. The Best Roguelike Games. IGN Entertainment 17.12.2021. Viitattu 1.5.2022 <https://www.ign.com/articles/top-10-roguelikes>.

Sicart, M. 2008. Defining Game Mechanics. Game Studies 2008. Viitattu 13.5.2022 <http://gamestudies.org/0802/articles/sicart>.

Laakkonen, C. 2020. GitHub CV:nä. Sofokus 29.1.2022. Viitattu 1.5.2022
<https://www.sofokus.com/fi/blogi/2020/01/29/github-cv/>.

Spencer, A. 2013. A History of Roguelikes in 6 Free Games. IGN Entertainment 26.12.2013. Viitattu 1.5.2022 <https://www.ign.com/articles/2013/12/26/a-history-of-roguelikes-in-6-free-games>.

Stargame, A. 2018. What game narrative is and what it means in casual games. Medium 13.9.2018. Viitattu 17.5.2022

<https://medium.com/@alexstargame/what-game-narrative-is-and-what-it-means-in-casual-games-67f35c191424>.

Suominen, I. 2020. Tiesitkö? Näin syntyi monien inhoama pelimekaniikka. Pelaaja 12.1.2020. Viitattu 26.10.2022 <https://pelaaja.fi/artikkelit/tiesitko-nain-syntyi-monien-inhoama-pelimekaniikka>.

Techterms. 2009. Graphics. Viitattu 27.10.2022 <https://techterms.com/definition/graphics>.

Unity. 2022a. Viitattu 15.5.2022 <https://unity.com>.

Unity. 2022b. Build smarter, not harder. Viitattu 6.12.2022 <https://unity.com/pages/introduction-to-asset-store#resources-beginners>.

Unity3d. 2022. Quick guide to the Unity Asset Store. Viitattu 4.5.2022 <https://unity3d.com/quick-guide-to-unity-asset-store>.

Unity Asset Store. 2022. Viitattu 7.12.2022 <https://assetstore.unity.com/>.

Wagar, C. 2018. Good FPS Map Desing. Celia Alexis Wagar's Critpoints 18.2.2018. Viitattu 23.11.2022 <https://critpoints.net/2018/02/18/good-fps-map-design/>.