

KÄYTTÄJÄKOKEMUS VIIHDEPELIN SUUNNITTELUSSA

Android-mobiilipeli Kuumat Kinkut

Mariella Heikkilä (& Sirpa Lehtimäki ja Aino Niskasaari)
Opinnäytetyö (AMK)
Syksy 2024
Tietojenkäsittely
Oulun ammattikorkeakoulu

TIIVISTELMÄ

Oulun ammattikorkeakoulu
Informaatioteknologia
Tietojenkäsittely

Tekijä(t): Mariella Heikkilä, Sirpa Lehtimäki & Aino Niskasaari
Opinnäytetyön otsikko: Käyttäjäkokemus viihdepelin suunnittelussa
Työn ohjaaja(t): Matti Viitala
Työn valmistumislukukausi ja -vuosi: syksy 2024
Sivumäärä: 63

Tämän opinnäytetyön tavoitteena oli tutkia käyttäjäkokemuksen merkitystä mobiilipelien suunnittelussa ja sitä, kuinka se vaikuttaa pelin menestymiseen, sen viihdyttävyyteen ja siihen kuinka käyttäjää saadaan sitoutettua pelin pelaamiseen. Pelinkehitysprosessia kuvataan pelikehityksen suhteen kokemattomien henkilöiden näkökulmasta. Pelin idea lähti ryhmän yhdessä keksimästä aiheesta, jota haluttiin lähteä kehittämään opinnäytetyön kautta.

Opinnäytetyö aloitettiin tutkimustyöllä, jonka tavoitteena oli kerätä tietoa mobiilipelien kehityksestä, pelisuunnittelusta, käyttöliittymä- ja käyttäjäkokemussuunnittelusta ja saavutettavuudesta. Aineistoa kerättiin kirjastoista, e-kirjoista, artikkeleista ja videoista. Tutkimustyössä käytettiin lisäksi hyväksi kyselyä, jolla kartoitettiin jo peleistä kiinnostuneiden henkilöiden motiiveja pelata mobiilipelejä.

Toiminnallisessa osuudessa laadittiin mobiilipelin suunnitelma teoretiedon pohjalta. Pelin ideoinnissa, visualisoinnissa ja suunnittelussa käytettiin Figma-työkalua. Figma-työkalulla toteutettiin myös suunnitelman mukainen prototyyppi pelistä. Tämän jälkeen aloitimme pelin kehitystyön Unity-pelimoottorilla.

Pelin kehityksessä ei tavoiteltu valmista peliä vaan haluttiin testata ja kehittää käyttäjäkeskeisen suunnittelun periaatteita, joita haluttiin tuoda esiin pelinkehityksessä. Lopputuloksena oli pelin käyttöliittymä, johon ryhmällä on jatkokehitysideoita, liittyen pelin pisteystylogiikkaan. Pelin jatkokehitykseen tulisi sisällyttää lisäksi pelidemon käyttäjättestaus ja valmiin julkaistavan pelin testaus. Lisäksi peliin halutaan tuoda lisää visuaalisia elementtejä, uusia etenemistasoja ja myöhemmin mahdollisesti moninpeliominaisuus.

ABSTRACT

Oulu University of Applied Sciences
Degree Program in Information Technology
Option of Degree Programme in Business Information Systems

Author(s): Mariella Heikkilä, Sirpa Lehtimäki & Aino Niskasaari
Title of thesis: User Experience in Entertainment Game Design
Supervisor(s): Matti Viitala
Term and year when the thesis was submitted: Autumn 2024
Number of pages: 63

The goal of this thesis was to study the importance of user experience in mobile game designing and how it affects to the success of the game, its entertainment value and how to keep the user engaged to the game. The process of game development was described from the perspective of unexperienced developers.

Thesis started with research phase, where the goal was to gather information about mobile game development, game design, user interface-/user experience design and accessibility. Information was gathered from library, e-books, articles and videos. Survey was also made to learn about the motivation to play mobile games from people interested in gaming.

Next, during functional phase, we made a design using the information gathered during research. Ideation, visualization and design was made using Figma, which was also used when creating a prototype of the game. After design phase, a development phase began with Unity game engine.

The goal was to test and develop user-centered design principles in game development and include those into our design. Result was a user interface, with further development ideas, including scoring system, design elements, progression levels and later perhaps multiplayer system. The future development should also include a usability test for the game demo and a test for the final product before release.

SISÄLLYS

TIIVISTELMÄ	2
ABSTRACT	3
SISÄLLYS	4
1 JOHDANTO	6
2 KÄYTTÖLIITTYMÄ	8
2.1 Käyttöliittymän suunnittelu yleisesti	8
2.2 Mobiilipelin käyttöliittymän suunnittelu.....	10
3 KÄYTTÄJÄKOKEMUS.....	12
3.1 Käyttäjäkokemuksen merkitys mobiilipelissä.....	13
3.2 Käytettävyys.....	15
3.3 Saavutettavuus.....	17
3.3.1 Pelin saavutettavuusohjeet.....	18
3.3.2 Avustava teknologia	19
3.3.3 Käyttäjätestaus.....	21
4 PELIN KEHITYSPROSESSI.....	23
4.1 Kehitysvälineet	23
4.1.1 Unity	23
4.1.2 Figma	28
4.1.3 Selainpohjaiset testityökalut	29
4.2 Pelisuunnittelun perusteet ja pelisuunnitelman luominen	31
4.3 Viihdyttävän pelikokemuksen tekijät.....	32
4.4 Prototyyppi mobiilipelin käyttöliittymästä	33
5 IDEASTA PELIKSI.....	34
5.1 Ideointi ja suunnittelu.....	34
5.2 Kysely ja analyysi	35
5.3 Pelin tekninen eteneminen	36
5.4 Prototyyppi	38
5.4.1 Ensimmäinen versio	39
5.4.2 Toinen versio	41
5.5 Demo.....	42
5.5.1 Ensimmäinen versio	43

5.5.2	Toinen versio	49
6	YHTEENVETO	54
6.1	Tekniset haasteet	54
6.2	Jatkokehitysideat.....	55
7	LOPPUPOHDINTA.....	57
	LÄHTEET	58

1 JOHDANTO

Tämän opinnäytetyön keskiössä on käyttäjäkokemuksen tärkeys mobiilipelin suunnittelussa ja kehityksessä. Lisäksi pohdittiin ja kokeiltiin erilaisilla testiversioilla kuinka käyttäjäkeskeinen ulkoasu ja toiminnallisuus käyttöliittymän kehittämisessä vaikuttavat viihdyttävyyteen, tuotteen käytettävyyteen ja pelin menestymiseen markkinoilla. Projektin aikana lähestyttiin käyttäjäkokemusta erilaisten tutkimusmetodien ja olemassa olevan asiantiedon pohjalta.

Videopelien historia ulottuu aina 50-luvulle saakka, jolloin William Higinbotham kehitti Tennis for Two –videopelin, vuonna 1958. Oskilloskoopin ruudulta pelattu peli keräsi suosiota näyttelyiden kautta, mutta luonnollisesti oli liian kallis markkinoille. 1962 Nolan Bushnell ja Ted Dabney kehittävät PDP-1-tietokoneelle Spacewar!-nimeä kantavan videopelin, joka kalliiden PDP-1-koneiden vuoksi oli myös liian kallis markkinoille. Spacewar!-peliä seuraa kolikkopeli Computer Space vuonna 1971, jota myytiin yli 1000 kappaletta. Computer Space -pelistä saaduilla tuotoilla Bushnell ja Dabney perustivat Atari-yhtiön, joka käytännössä aloittaa kolikoilla pelattavien videopelien historian ja jatkuu aina tähän päivään saakka. (Heinonen, s.a.)

Ensimmäiset videopelit ovat olleet yksinkertaisia ja suoraviivaisia, kuitenkin suosion kasvaessa ominaisuudet ovat lisääntyneet. Videopelien alkuaikoina käyttäjäkokemukset ovat olleet epäintuitiivisia ja oppiminen on ollut haasteellista. Pelin sisäinen opastus on ollut epämääräistä, mikäli sellaista on ollut lainkaan. Tämä on johtanut usein siihen, että pelaaja ”eksyy” peliin ja tuntee pelin läpäisemättömäksi (Greenan, 2023.)

Kun lankapuhelinten aikakausi alkoi olla loppuillaan, yhä useampi omisti kännykän. Tämän kyseisen aikakauden nuoret ja aikuiset muistavat todennäköisesti yksinkertaisemmat kännykkäpelit kuten matopelit, joiden parissa saattoi viettää tunteja. Näiden yksinkertaisten pelien tavoite oli selkeä, pitää mato liikkeessä mahdollisimman pitkään ja kasvattaa se mahdollisimman pitkäksi. Myöskään käyttöliittymä ei ollut tähän aikaan vielä sen monimutkaisempi. Matopelit sisälsivät usein myös mahdollisuuden haastavuusasetuksen muokkaamiseen, joka

tyypillisesti oli madon liikkeen nopeuttaminen. Kun puhutaan vanhoista peleistä, monelle tulee myös mieleen PC-alustainen miinaharava, jota moni on varmasti yrittänyt kerran, jos toisen läpäistä siinä kuitenkaan onnistumatta. Vanhoja pelejä pelatessa saattoi huomata, että ne harvemmin sisälsivät tutoriaaleja yhtä usein kuin nykyisin ja tästä syystä pelien, kuten miinaharava, säännöt saattoivat jäädä monelta ymmärtämättä. Käyttäjäkokemuksen kannalta pelin ymmärtäminen aikaisessa vaiheessa on ratkaisevassa asemassa, kun tavoitellaan aktiivista pelaajakuntaa. Ilmaisia pelejä ei ollut tällöin kuitenkaan saatavilla samalla tavalla ja usein päädyttiin pelaamaan juuri näitä laitteen mukana tulleita pelejä.

Kahden viimeisen vuosikymmenen aikana, kun älypuhelin suosio on kasvanut siihen pisteeseen, että lähes jokaisella on päivittäisessä käytössä älypuhelin, on samalla mobiilipelien suosio ollut räjähdysmäisessä kasvussa. Maailmanlaajuisen mobiilipelien markkinan on odotettu kasvavan 128.5 miljardiin dollariin vuoteen 2025 mennessä, kun vuonna 2020 sen markkina-arvo on ollut 77.2 miljardia dollaria (30DC 2024). Uusia pelejä ilmestyy sovelluskauppoihin päivittäin ja kilpailu näiden välillä on kovaa. Harva näistä peleistä saavuttaa suuren suosion. Osa taas saatetaan ladata ja poistaa lähes välittömästi. Statistan tekemän mittauksen mukaan, vuonna 2023, lokakuun ja marraskuun välisenä aikana, mobiilipelien poistamisten määrä Android-puhelimista oli 52,02 % maailmanlaajuisesti (Ceci 2024a).

Käyttökokemuksen merkitys mobiilipeleissä, kuin myös muiden mobiilisovellusten kohdalla, on tehdä sovelluksesta saumaton ja miellyttävä käyttää. Käyttökokemuksen suunnittelun tavoitteena on jättää käyttäjälleen positiivinen ja mukansatempaava mielikuva, joka pitää heidät sitoutuneena kyseisen sovelluksen maailmaan. (Tyson 2023.)

Näihin tavoitteisiin pyrimme opinnäytetyön pohdinnassa ja toteutuksessa.

2 KÄYTTÖLIITTYMÄ

Käyttöliittymä tietokone- tai mobiilipelissä, on se osa, jonka kautta käyttäjä tuotetta käyttää ja jonka käyttäjä näkee, sekä myös tapa, jolla ohjelmaa käytetään, esimerkiksi hiiri, näppäimistö tai kosketus (Wikipedia 2024). Käyttöliittymä (engl. user interface) eli UI mahdollistaa käyttäjän ja sovelluksen välisen vuorovaikutuksen käyttäen visuaalisia ja auditiivisia elementtejä, kuten fontteja, ikoneita, painikkeita, animaatioita ja ääniä. Hyvä käyttöliittymä mahdollistaa tarkoituksenmukaisen käytön ja helpon navigoinnin käyttöliittymän läpi. (Hashemi-Pour & Churchville 2024a.)

2.1 Käyttöliittymän suunnittelu yleisesti

Käyttöliittymän suunnittelu voi olla yksi sovelluksen aikaa vievimmistä osuuksista. Hyvät, käyttökelpoiset ja käyttäjäystävälliset käyttöliittymät eivät synny sattumalta. Käyttöliittymän täytyy tarjota käyttäjälleen kaikki tarpeellinen tieto sopien samalla pelin visuaaliseen ilmeeseen ja sen tulisi olla käytettävä ilman erillisiä ohjeistuksia. Huono käyttöliittymä voi pilata hyvän pelin. Hyvää käytettävyyttä silmällä pitäen suunniteltu käyttöliittymä sulautuu usein huomaamattomasti taustalle. (Menard 2012, 309).

Konventiot ovat tärkeitä käyttöliittymän suunnittelussa. Konventio tarkoittaa, että jokin asia on tuttu, se on opittu jo aiemmin vastaavista tuotteista. Käyttäjää tarkkailemalla, haastatteleamalla ja käytettävyytestauksilla voidaan selvittää käyttäjien konventiot, jotta niitä voidaan hyödyntää suunnittelutyössä oikealla tavalla. (Sinkkonen ym. 2006, 136.)

Käyttöliittymän elementit tulisi suunnitella käyttäjän näkökulma huomioiden. On tärkeää ymmärtää ihmisen havaintojärjestelmän tiettyjä piirteitä. Käyttäjät eivät välttämättä huomaa kaikkia käyttöliittymän elementtejä. Tuotteen suunnittelija taas voi nähdä sen eri tavalla kuin uudet käyttäjät. Suunnittelijalle jokaisella elementillä on merkitys, mutta käyttäjälle näin ei välttämättä ole. Käyttäjän aiemmat miellelyhtymät ja kokemukset vaikuttavat siihen, miten hän tulkitsee näkemänsä.

Ihmiset ovat hyviä tunnistamaan ja oppimaan tuttuja elementtejä, jos niillä on heille merkitystä. (Sinkkonen ym. 2006, 69.)

Käyttöliittymän elementtien asettelun lähtökohtana on intuitiivisuus, suunnittelussa tulee keskittyä käyttäjän luonnollisiin tavoitteisiin ja tehtäviin, jotta käyttöliittymä on tehokas ja vaivaton käyttää. Tavoitteena on luoda selkeä ja käyttäjäystävällinen kokemus, jossa navigointi ja toimintojen ymmärtäminen on helppoa. Elementtien ryhmittely ja järjestys sekä oikeanlainen hierarkia ovat käyttöliittymän suunnittelun keinoja. Tasapainoinen asettelu saavutetaan käyttämällä kontrasteja, tyhjää tilaa, elementtien muotoja ja sijoittelua, liikesuuntia sekä värejä. (Sinkkonen ym. 2006, 155.)

Käyttöliittymän värejä suunniteltaessa on tärkeää huomioida, että noin kahdeksan prosenttia miehistä ja neljä prosenttia naisista kärsii jonkinlaisesta värisokeudesta, yleisimpänä puna-vihersokeus. (Sinkkonen ym. 2006, 133.)

Ärsykeillä ja visuaalisilla vihjeillä voidaan suunnittelussa vaikuttaa siihen, miten tärkeä tieto erottuu käyttöliittymässä. Jotta ärsyke havaittaisiin, sen on ylitettävä tietty kynnyks. Tämä tarkoittaa, että esimerkiksi tekstin on erotettava taustastaan selkeästi, jotta se voidaan aistia. Ärsykekynnyksen ylittämiseen voidaan vaikuttaa esimerkiksi järjestämällä tiedot siten, että tärkeiden asioiden ympärillä on tarpeeksi tyhjää tilaa, käyttämällä visuaalisia vihjeitä, välttämällä huomion kiinnittämistä epäolennaiseen tietoon tai kuviin. On myös hyvä varmistaa, ettei taustakuva ole liian häiritsevää tai levoton. (Sinkkonen ym. 2006, 77–78.)

Hyvä käyttöliittymäsuunnittelu huomioi myös, miten käyttäjät ovat vuorovaikutuksessa laitteiden kanssa. On tärkeää tarjota selkeä ja merkityksellinen palaute jokaisesta suoritetusta toiminnosta, jotta käyttäjä tietää toiminnon onnistuneen. Liian hidas palaute on lähes yhtä huono kuin ei palautetta ollenkaan. Suunnittelua kannattaa lähestyä uuden käyttäjän näkökulmasta ja johdonmukaisuus on avainasemassa; samanlaisten painikkeiden tulisi tuottaa samanlaisia toimintoja. Lisäksi käyttöliittymän olisi hyvä hyödyntää jo käyttäjälle tuttuja standardeja ja paradigmoja, kuten suljepainiketta oikeassa yläkulmassa. Käyttöliittymän tulee olla responsiivinen, selkeä, ja näyttää järjestelmän tila käyttäjälle ymmärrettävällä tavalla. Aikainen ja tiheä testaaminen on tärkeimpiä vaiheita. On suositeltavaa

testata peliä uusilla käyttäjillä. Mikäli käyttäjä tarvitsee paljon ohjeistusta, kertoo tämä yleensä jatkokehityksen tarpeesta. (Menard 2012, 309–312.)

Käyttöliittymän suunnittelussa on eroja myös kulttuurisesta näkökulmasta. Koska kulttuuriero vaikuttavat tapamme havaita ja käyttää käyttöliittymiä, sen huomiointi suunnittelussa voi vahvistaa käyttöliittymän intuitiivisuutta ja toimivuutta. (Sinkkonen ym. 2006, 37.)

Eri kulttuureissa on eroja värisymbolismilla, ikonografialla ja sisällön esitysmuodolla. Visuaalisten elementtien, kuten kuvien ja ikonien valinnassa tulee huomioida eri kulttuurista tulevat käyttäjät, sillä joillain symboleilla voi olla yleinen merkitys, kun taas joillain merkitys voi vaihdella paljonkin kulttuurin normien ja uskomusten mukaan. (Microsoft 10.5.2024.)

2.2 Mobiilipelin käyttöliittymän suunnittelu

Erilaisille laitteille ja ohjelmistoille sekä sovelluksille on erilaisia käyttöliittymiä. Mobiilisovelluksien suosion myötä on muotoutunut mobiilikäyttöliittymä, jonka keskiössä on parantaa käytettävyyttä ja saavutettavuutta älypuhelimien ja tabletien pienet näyttöruudut huomioiden. Älypuhelimien kosketusnäytöt hyödyntävät kosketustoimintoja, joka on yksi olennainen osa mobiilikäyttöliittymää. (Hashemi-Pour & Churchville 2024b.)

Kosketuksen kautta toimivat interaktiot ja kontrollointi vaativat kosketusnäytöltä riittävän suuret elementit, joihin on helpompi osua ja tarpeeksi tilaa elementtien ympärille, jotta vahinko-osumia ei tulisi herkästi. Käyttöliittymän elementtien asetelu tehdään harkiten tarkkaan ergonomia ja visuaalisuus. Jokaisen pyyhkäisyn, pinch-, touch- ja tap- tapahtuman tulee tuntua luonnolliselta ja integroitua pelin mekaniikan kanssa hyvin mahdollistaen immersion. Mobiilipelin käyttöliittymältä vaaditaan myös skaalautuvuutta ja responsivisuutta enemmän verrattuna esimerkiksi PC-peleihin, koska mobiilisovelluksia pelataan erikokoisilla älypuhelimilla ja tableteilla. Myös suorituskyvyn optimointiin tulee kiinnittää huomiota, ja mobiilipelin käyttöliittymässä tämä voi näkyä esimerkiksi kevyempien resurssien käyttönä, näytöllä olevien elementtien määrässä ja animaatioiden optimoinnissa. Mobiilipelin käyttöliittymässä ei ole kyse pelkästään visuaalisesta

miellyttävyydestä, vaan tavoitteena on varmistaa, että käyttäjäkokemus on intuitiivinen ja koukuttava. Kun sekä käyttöliittymä että käyttäjäkokemus integroituvat hyvin yhteen, saadaan aikaan hyvä pelikokemus. (Kruchkow, 2024.)

Aloittelijoiden tarpeet ja taitotasot käyttöliittymän suunnittelussa

Mobiilipelien suunnittelijan haasteena on pieni näyttö, jolloin on tärkeää ottaa huomioon käyttöliittymän yksinkertaisuus. Navigoinnin ja pelimekaniikan tulisi olla mahdollisimman selkeitä, jotta aloittelevakin pelaaja voi omaksua pelin ominaisuudet ja kykenee niiden intuitiivisen käytön.

Kokeneilla pelaajilla on pelien käyttöliittymän suhteen tyypillisesti vahva intuitio siitä, miten pelin käyttöelementit toimivat, mutta pelien suunnittelijoiden tulisi kuitenkin ottaa aina huomioon miltä käyttöliittymä aloittavan pelaajan näkökulmasta näyttää. Nykyisin useat pelit suunnitellaan alkamaan tutoriaalinäköymästä ja sisältävät helpon aloituksen, jonka tavoitteena on viedä pelaajaa pelin poikki, jolloin sen päämäärä tulee pelaajalle selväksi taitotasosta riippumatta.

3 KÄYTTÄJÄKOKEMUS

Käyttäjäkokeusta, eli käyttäjän tunnesidettä tuotteeseen, pidetään usein epämääräisenä ja abstraktina käsitteenä, minkä vuoksi se on jaettu hallittavampiin osiin. Käyttäjäkokeus muodostuu tunteista ja riippuu sekä tuotteesta että käyttäjästä ja käyttöolosuhteista. Ensivaikutelma syntyy, kun tuote havaitaan tai kuullaan ensimmäisen kerran. Tilannetta kutsutaan houkuttelevuudeksi, joka voi myöhemmin johtaa emotionaaliseen kiintymykseen tuotetta kohtaan. Käyttäjäkokeukseen liittyviä tuoteominaisuuksia ovat tuotteen substanssi, tuotteen vetovoima, tuotteen käytettävyys ja tuotteen uskottavuus. (Sinkkonen ym. 2006, 260–262.)

Immersio

Immersio on pelimaailmassa esiintyvä tunne, joka saavutetaan silloin, kun pelaaja ei enää ajattele pelaavansa tietokonepeliä, vaan on itse siinä pelissä. Immersio tarkoittaa siis tilanteeseen tai tuotteeseen uppoutumista. Uppoutuminen voi tapahtua rationaalisesti toiminnan kautta tai tunnepohjaisesti samaistumalla pelihahmoon ja tarinaan. Eteneminen immersioon voi edetä esimerkiksi seuraavissa vaiheissa: sitoutuminen, mukaan tempautuminen, täysi uppoutuminen eli immersio. Sitoutuminen vaatii pelaajalta keskittymisen peliin, joka toteutuu parhaiten pelin ollessa riittävän helppokäyttöinen. Tällöin pelaaja voi keskittyä itse sisältöön käyttöliittymän sijaan. Käytettävyys- tai käyttöongelmat estävät immersion synnyn. (Sinkkonen ym. 2006, 267.)

Tunnekokemukset

Tunnekokemuksiin käyttötilanteissa sisältyy myös niin kutsuttu virtauskokemus (engl. flow), joka jakaa immersion kanssa monia piirteitä, kuten täydellinen keskittyminen, ajantaju ja tietoisuus itsestä katoavat, taidot ja haasteet ovat tasapainossa, ja toiminta rytmittyy palautteen kanssa. Toisin kuin immersiossa, virtauskokemus syntyy itse suorittamisen kautta ja kestää pidempään. Keskeistä on tietoinen energian kohdistaminen tavoitteellisesti tehtävään. Virtauskokemus syntyy, kun henkilön tavoitteet ovat selkeät ja keinot niiden toteuttamiseen ovat

selvillä sekä palaute on välitöntä ja ymmärrettävää. Pelin tarkasti rajatut toimintasäännöt ja vuorovaikutustavat sekä rajoitteet ja palautteet antavat pelin suunnittelijalle mahdollisuuden vaikuttaa virtauskokemuksen syntymiseen. (Sinkkonen ym. 2006, 267–268.)

3.1 Käyttäjäkokemuksen merkitys mobiilipelissä

Koska käyttäjäkokemukseen vaikuttavat ensisijaisesti käyttäjän vuorovaikutus pelin kanssa ja sen luomat tunne-elämykset, onkin tärkeää luoda mobiilipeli, jonka suunnittelu on toteutettu niin että käyttäjä viihtyy sen parissa. Miellyttävän mobiilipelin tärkeitä lähtökohtia ovat erityisesti sen käyttäjäkokemuksen keskeisissä komponenteissa, kuten pelimekaniikassa, visuaalisessa ilmeessä ja äänitehosteissa. Yhtä tärkeää on myös käyttäjälähtöinen suunnittelu, jossa kartoitetaan, millaiselle kohderyhmälle peli halutaan tehdä. Kaikki pelaajat eivät nauti moninpeleistä, mutta se ei tarkoita, etteivätkö nämä henkilöt olisi kilpailuhenkisiä ja toivo päihittävänsä muita pelaajia pistetaulukoissa. Sosiaalinen aspekti on kuitenkin hyvä kohde pohdinnalle, sillä pelin, joka onnistuu luomaan vahvaa yhteisöllisyyttä pelaajiensa keskuudessa, voidaan ajatella onnistuneen käyttäjäkokemuksen toteutuksessa positiivisesti. Onnistuneesta sosiaalisen mobiilipelin suunnittelusta hyvänä esimerkkinä toimii vuonna 2016 julkaistu Pokémon GO (kuva 1), joka paitsi sai pelaajat liikkeelle, myös sai aikaan suuria pelaajayhteisöjä (Toth 2016).



KUVA 1. Näkymä Pokémon Go -pelistä (Pokémon Go 2024)

Mobiilipelejä voi ladata sovelluskaupoista hetkessä, ja niiden parissa voidaan viettää aikaa tunneista päiviin ja jopa kuukausiin – tai ne voidaan poistaa lähes yhtä nopeasti kuin ne on ladattukin. Nopeasta sovelluksen poistamisesta tuntuvat kärsivän erityisesti treffisovellukset, mutta lähes yhtä suuren poistomäärän kokee mobiilipelit. Enemmän kuin joka toinen mobiilipeli poistetaan 30 päivän kuluessa sen latauksesta (Ceci 2024b). Osaa mobiilipeleistä voisi kuvailla jopa “pelimaailman pikamuotina”. Mobiilipelejä ladataan matalalla kynnyksellä nimenomaan siksi, että ne ovat tyypillisesti ilmaisia. Kuitenkin näitä pelejä kehittävien yritysten liikevaihdot voivat olla todella massiiviset, mikäli pelistä onnistutaan kehittämään käyttäjää viihdyttävä, ja täten onnistutaan kehittämään peli, joka kerää suuren aktiivisen pelaajakunnan.

Käyttäen esimerkkinä monen tuntemaa Candy Crush Saga -mobiilipeliä, joka peruseriaatteeltaan on yksinkertainen älypeli, mutta tunnetusti aiheuttaa riippuvuutta käyttäjiensä keskuudessa. Vuonna 2022 Candy Crush Saga -mobiilipelillä oli 210 miljoonaa aktiivista pelaajaa (Curry 2024) ja sen mainoksista saadut tuotot olivat yli miljardi dollaria (Long 2023). Candy Crush on pelinä loistava esimerkki pelien tarjoamasta audiovisuaalisesta palautteesta, jonka avulla pelaaja kokee pelissä saavuttavansa asioita ja etenevänsä pelissä. Aivan kuten kasinoiden kolikkopeleissä vilkkuvat valot ja animaatiot on suunniteltu niin, että pelaajan aivojen tuottama dopamiini ja endorfiini saavat hänet pelaamaan vielä muutaman kierroksen lisää. Pelin tavoite on suhteellisen yksinkertainen, mutta peli sisältää lisäksi loputtoman määrän tasoja ja antaa pelaajalle mahdollisuuden osallistua kilpailuihin muiden pelaajien kanssa. Kun ajatellaan lisäksi muita koukuttavia elementtejä kyseisessä pelissä, voidaan ajatella sen värimaailmaa ja animaatioita, jotka tekevät siitä miellyttävän ajankulutuspelein.

Andrey Spencer avaa Candy Crush-pelin käyttäjäkokemusta omassa analyysissään ”Does Candy Crush have good UX?” (2020), viittaamalla sen johdonmukaisuuteen tyyliässä, joka näkyy sen kirkkaissa väreissä, leikkisässä musiikissa ja animaatioissa. Tämän leikkisän tyylin vuoksi kuitenkin, esimerkiksi fonttivalinta voi olla osalle käyttäjistä haasteellista lukea (kuva 2).

Omalatuinen tyyli on tyypillisesti pelin brändin kannalta hyvä asia, mutta sillä voi olla negatiivinen vaikutus pelin saavutettavuuteen, mikäli käyttäjäkokemusta ei mietitä riittävän tarkasti tai kehitetä koko pelin elinkaaren ajan riittävästi.



KUVA 2. Candy Crush Saga -pelin värimaailmaa (Candy Crush Saga 2024)

3.2 Käytettävyys

Käytettävyys määritellään ISO/IEC 9241-11 -standardin mukaisesti tuotteen kyvyksi tietyillä käyttäjillä saavuttaa ennalta määritellyt tavoitteet tehokkaasti, tuloksellisesti ja tyytyväisyyttä tuottaen tietyssä käyttötilanteessa (NIST s.a.). Määritelmä korostaa käyttäjäkeskeisyyttä, jossa sujuvuus, tehokkuus ja käyttäjäkokemuksen laatu on tarkastelun alla. Käytettävyys on yksinkertaisella kuvauksella sitä, kuinka luontaista, intuitiivista ja helppoa tuotteen käyttö käyttäjälle on.

Käytettävyys on yksi perusasioihin kuuluvista ominaisuuksista, jonka tulee olla kunnossa, jotta käyttäjäelämyksiä voi syntyä (Sinkkonen ym. 2006, 10).

Suunnittelutyön apuna ovat erilaiset standardit, heuristiikka ja kultaiset säännöt, mutta niiden lisäksi tärkeää on myös ymmärrys ihmisen ja tuotteen vuorovaikutuksesta (Sinkkonen ym. 2006, 11). Korkean käytettävyyden omaava tuote tarjoaa intuitiivisen ja vaivattoman kokemuksen jättäen käyttäjät tyytyväisiksi, minimoiden turhautumisen ja maksimoiden tuottavuuden (Suleiman 2024a).

Käytettävyys auttaa suunnittelijoita ymmärtämään käyttäjien ajattelutapaa ja toimintamalleja paremmin. Se voi auttaa luomaan positiivisemmän tunnereaktion uuteen tuotteeseen. Käytettävyyttä voidaan kuvailla ja ajatella monin eri tavoin. Suunnittelutyössä käytettävyys ja käytettävyyden huomioiminen auttaa luomaan toimivamman ja hyödyllisemmän tuotteen. (Sinkkonen ym. 2006, 272.)

Tunteiden, motiivien ja tarpeiden mukaanotto laajentaa perinteisen käytettävyyden käsitteen käyttäjäkokemus-käsitteeksi (Sinkkonen ym. 2006, s. 248). Käytettävyyden määritelmät pyrkivät yleensä välttämään negatiivisia tuntemuksia käyttötilanteessa, mutta käyttäjäkokemus kattaa laajemmin kaikki tekijät, jotka vaikuttavat käyttäjän ja organisaation välisiin suhteisiin tuotteen kautta. Tämä sisältää sekä tuotteen käyttökokemuksen että mielikuvat valmistajan brändistä ja tuoteperheestä. (Sinkkonen ym. 2006, 248.)

Käytettävyyttä voidaan pohtia myös kulttuuristen erojen kautta, miten kulttuuri vaikuttaa käyttäjän vuorovaikutukseen ja kokemukseen käyttöliittymän kanssa. Kulttuuri määrittelee osaltaan käyttäjän toimintaa. Vaikka biologiset ominaisuudet, kuten muistirakenteet ja värien aistiminen, ovat samankaltaisia useimmilla ihmisillä, muistisisällöt ja värien havaitseminen vaihtelevat suuresti kulttuurien ja yksilöiden välillä. Kulttuurierot vaikuttavat merkittävästi havainnointiin ja toimintaan, muodostaen osan käyttäjän toimintaympäristöstä. Esimerkiksi suomalainen käyttäjä etsii verkkosivun valikkoa yleensä ruudun vasemmalta puolelta, kun taas kiinalainen saattaa katsoa ensin oikealle. (Sinkkonen ym. 2006, 37.)

Mobiilisovelluksen käytettävyys tarkoittaa käytännössä sitä, miten helpoksi ja miellyttäväksi käyttö koetaan. Käytettävyys vaikuttaa voimakkaasti käyttäjäkokemukseen, joka taas vaikuttaa siihen, kuinka käyttäjät omaksuvat sovelluksen käytön. Käytettävyys kattaa käyttöliittymäsuunnittelun ja käyttäjäkokemuksen lisäksi navigoinnin, reagointikyvyn ja virheidenkäsittelyn. Hyvä käytettävyys voidaan

saavuttaa perusteellisella käyttäjätutkimuksella, iteratiivisella työskentelyllä ja testaamisella. Käytettävyyden huomioiminen ja siihen panostaminen vaikuttaa positiivisesti käyttäjätuottavuuteen ja käyttäjien säilyttämiseen. Käytettävyyden periaatteet ovat myös linjassa esteettömyysohjeiden kanssa. (Suleiman 2024b.)

3.3 Saavutettavuus

Yksi olennaisista tekijöistä käyttäjäkokemuksessa käytettävyyden lisäksi, on tuotteen tai palvelun saavutettavuus. Saavutettavuus tarkoittaa esteettömyyttä digitaalisissa palveluissa kuten verkkosivuissa ja mobiilisovelluksissa, ja niiden sisällöissä. Se parantaa yhdenvertaisuutta digitaalisessa yhteiskunnassa, huomioimalla ihmisten erilaisuuden ja moninaisuuden tuotteen tai palvelun kehityksessä, läpi koko prosessin, suunnittelusta toteutukseen. (Aluehallintovirasto 2024a.)

Saavutettavuus on keskeisesti asiakaslähtöisyyttä, ja osa suunnittele kaikille – periaatetta. Suunnittelussa huomioidaan tuotteen tai palvelun tekninen saavutettavuus, joka voi sisältää esimerkiksi avustavia teknologioita, kuten puheohjausta. Avustavien teknologioiden toteutuksien tulee olla virheetöntä. Toinen osa-alue on helppokäyttöisyys, joka tarkoittaa, että tuote on helppo hahmottaa, navigaatio eli liikkuminen näkymästä tai toiminnosta toiseen, on selkeä, ja haluttu toiminto on vaivaton suorittaa. Mobiilipelissä tämä tarkoittaa selkeää ja hahmotettavaa käyttöliittymää. Lisäksi sisällön tulee olla kieleltään ymmärrettävää ja helppolukuista. Vaikka mobiilisovellusten saavutettavuus on välttämätöntä joillekin ihmisille, se on samalla hyödyllistä meille kaikille. (Aluehallintovirasto 2024b.)

Saavutettavuuden periaatteet

Saavutettavuutta ohjaa WCAG (Web Content Accessibility Guidelines), suomeksi Verkkosisällön saavutettavuusohjeet. Ohjeiden tavoitteena on varmistaa erilaisten ihmisten kyky käyttää verkkopalveluita itsenäisesti, riippumatta mahdollisesta vammasta tai toimintarajoitteesta. WCAG - ohjeistuksessa saavutettavuuden neljä periaatetta ovat havaittavuus, hallittavuus, ymmärrettävyys ja toimintavarmuus. Saavutettavuutta määritellään tarkemmin näiden periaatteiden alla 13: lla käskymuotoisella ohjeella, jotka tarkentavat puitteita ja yleisiä tavoitteita. (Saavutettavasti.fi 2023a.)

Edellä mainitut saavutettavuusohjeet sisältyvät paitsi WCAG 2.1 - ohjeistukseen, myös lakiin digitaalisten palvelujen tarjoamisesta. Vaikka laki koskee pääasiassa viranomaisen asemassa toimivia tai julkisoikeudellisia laitoksia, järjestöjä ja organisaatioita, voi sitä ja WCAG ohjeistusta käyttää ohjeena myös yksityisen sektorin tuotteissa, kuten vaikka pelikehityksessä. Näiden ohjeistuksien ja lain lisäksi jo perustuslaki takaa kaikille yhdenvertaisuuden.

“Ketään ei saa ilman hyväksyttävää perustetta asettaa eri asemaan sukupuolen, iän, alkuperän, kielen, uskonnon, vakaumuksen, mielipiteen, terveydentilan, vammaisuuden tai muun henkilöön liittyvän syyn perusteella.” (Perustuslaki 731/1999, 6§.)

3.3.1 Pelin saavutettavuusohjeet

Pelin saavutettavuusohjeet (Game accessibility guidelines) on tuotettu yhteistyössä eri pelistudioiden, asiantuntijoiden ja tutkijoiden kanssa. Ohjeita ei ole kirjattu lakiin eikä asetukseen, vaan niillä on oma web-sivusto. Tavoitteena on ollut tarjota kaikille saatavilla olevat, kehittäjäystävälliset kehykset tapoihin, joilla pelejä kehitetään, vältetään pelaajien tarpeeton poissulkeminen ja pelit ovat pelattavia sekä hauskoja mahdollisimman monille. (Game accessibility guidelines 2016a.)

Pelin saavutettavuuden ohjeet antavat hyvän kehyksen pelisuunnitteluun niin, että pelistä tulee mahdollisimman monelle saavutettava, mahdollisista esteistä huolimatta. Nämä ohjeet on jaettu kolmeen kategoriaan; perustaso, keskitaso ja edistynyt taso, jotka kaikki perustuvat kolmen kohdan tasapainoon. Nämä kolme, kunkin kategorian tasapainon kohtaa ovat tavoitettavat ihmiset, jotka hyötyvät näistä ohjeista, vaikutus, joka saavutetaan ohjeita seuraamalla, ja arvo tai kustannus, joka tulee ohjeiden mukaisesta toteutuksesta. Nämä kohdat ovat edelleen ryhmitettynä alakategorioihin mahdollisen esteen tai vamman mukaan sekä muutamaan yleisesti pätevään ohjeeseen. (Game accessibility guidelines 2016b.)

Pelin saavutettavuusohjeissa annetaan esimerkki prosessin etenemisestä, jota seuraamalla saavutetaan suuri hyöty. Prosessi on jaettu kuuteen vaiheeseen.

Ensimmäinen vaihe on ohjeisiin tutustuminen ennen minkään pelikehitystyön aloittamista. Suurin osa ohjeiden hyödyistä saavutetaan yksinkertaisilla muotoilu- ja suunnittelupäätöksillä.

Toinen vaihe on arviointi ja suunnittelu, jossa päätetään mikä tai mitkä ohjeista sopivat parhaiten kehitteillä olevaan peliin.

Kolmannessa vaiheessa priorisoidaan ja ajoitetaan ohjeet oikeisiin kohtiin pelikehitysprosessissa sen perusteella, millä on missäkin vaiheessa suurin ja hyödyllisin vaikutus sekä tuotannossa, että myös pelaajien saavuttamisessa ja hyvän pelikokemuksen tarjoamisessa.

Neljäntenä vaiheena on toteutus. Tämä sisältää prototyyppitestausta mahdollisimman monipuolisella testiryhmällä, jossa on edustettuna kattavasti eri vammaisryhmiin kuuluvia ihmisiä sekä erilaisia kognitiivisia, motorisia tai muita pysyviä tai väliaikaisia esteitä omaavia ihmisiä.

Viidentenä vaiheena ohjataan pelikehittäjiä tiedottamaan pelinsä esteettömyysominaisuuksista itse pelissä ja omilla sivuilla sekä kertomaan niistä myös peliarvostelusivustoille.

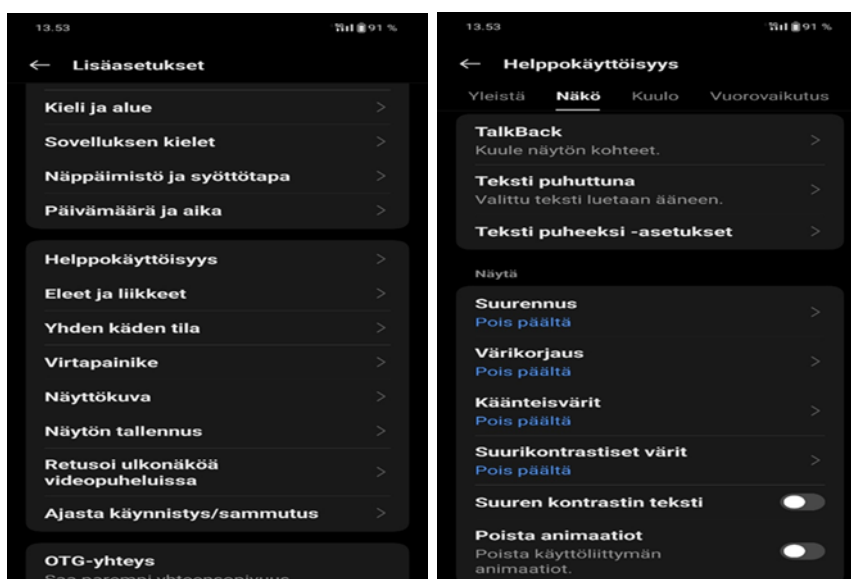
Kuudennessa vaiheessa kehoitetaan arvioimaan ja mittaamaan esteettömyysominaisuuksien vaikutusta kustannuksiin, pelaajien määrään ja ominaisuuksien arvoon pelaajaa kohden. Näillä voidaan mitata kannattavuutta, ja tuloksia voidaan hyödyntää tulevissa peliprojekteissa. (Game accessibility guidelines 2016c.)

Mobiilipelissä, kuten peleissä yleensäkin, yksi tärkeä kriteeri saavutettavuudessa onnistumisessa ovat värit ja niiden kontrasti. WCAG ohjeistuksessa säädetään väreistä havaittavuuden periaatteissa. Tasolla AA vähimmäiskontrasti on 4,5:1 ja tasolla AAA vähimmäiskontrasti on 7:1.(Saavutettavasti.fi 2024b.)

3.3.2 Avustava teknologia

Avustavalla teknologialla tarkoitetaan esimerkiksi näytönlukijaa, näppäimistöä, puhesyntetisaattoria tai suurennuslasia. Nämä teknologiat tarjoavat

loppukäyttäjälähtöisen näkökulman testattavaan tuotteeseen. Käyttö kuitenkin vaatii testaajalta tuntemusta avustavasta teknologiasta ja kykyä asettua rajoitteita omaavan henkilön asemaan. Avustavina teknologioina Android-laitteissa ja Applen mobiililaitteissa on maksuttomana suurennuslasiominaisuus, ruudunlukuohjelma sekä koneäänet. Applen Mac OS-ympäristössä tämä on VoiceOver-ruudunlukuohjelma ja Android-järjestelmässä TalkBack-ohjelma (kuva 3). Näitä tekniikoita voidaan käyttää apuna mobiilipeliä pelattaessa. Windows-ympäristöön on puolestaan saatavilla ladattava ilmainen NVDA-ohjelma, jossa mukana on puhesyntetisaattori. (Näkövammaisten liitto 2023.)



KUVA 3. Android -puhelimesta löytyviä avustavia teknologioita (Android 2024)

Monet pelaajat, jotka eivät pysty käyttämään fyysistä näppäimistöä, käyttävät virtuaalista näytöllä olevaa näppäimistöä, jota voidaan ohjata hiirellä tai aputekniikalla, kuten katseen tai kytkimen avulla. Jotta nämä toimisivat, pelin on oltava ikkunatilassa ja sallittava muiden sovellusten yhtäaikainen päällä oleminen. (Game accessibility guidelines 2016d.)

Avustavat teknologiat mobiilipelissä

Mobiilipelissä voidaan käyttää Switch Access-teknologiaa saavutettavuustyökäluna. Teknologia hyödyntää ulkoisia kytkimiä, jotka voidaan räätälöidä tarpeen mukaan suorittamaan erilaisia toimintoja pelissä. Kytkimiä voi käyttää esimerkiksi kädellä, jalalla tai vaikka pään liikkeellä. Lisäksi on olemassa elesäätimiä, jotka käyttävät puhelimen kameraa, muuttaen liikkeen toiminnoksi pelissä. TalkBack-

toiminnon lisäksi mobiilipeliä voi pelata äänitunnistustyökalun avulla, joka on avuksi silloin, kun peliä pitää pystyä pelaamaan ilman käsiä tai muun motorisen esteen vuoksi. Haptinen palaute puolestaan antaa tuntopalautetta värähtelyllä ja näin parantavat pelikokemusta kuulo- tai näkövamman omaavalle henkilölle. Monia muitakin avustavia tekniikoita mobiilipelin pelaamiseen on olemassa. Näitä ovat silmän liikettä seuraavat teknologiat, mukautuvat ohjaimet ja Tecla-apulaite vakavia fyysisiä vammoja omaavalle henkilölle. (Ablegamers 2024.)

3.3.3 Käyttäjätestaus

Käyttäjätestausta varten kannattaa koota mahdollisimman monipuolinen, eri käyttäjäryhmistä ja erilaisista käyttäjistä koostuva testiryhmä. Mukana on hyvä olla myös erilaisia rajoitteita omaavia henkilöitä. Toinen vaihtoehto on käyttää palautekanavaa, jolla saadaan tietoa käyttäjien havainnoista. Yleisesti ottaen käyttäjätestaus on aikaa ja resursseja vievää, sekä kallista. Siitä huolimatta tämä menetelmä tarjoaa autenttista tietoa testattavasta tuotteesta ja antaa uusia näkökulmia tuotteen kehitykseen. (Nyqvist 2023a.)

Yleisesti saavutettavuuden ja käytettävyyden testaukseen on saatavilla paljon erilaisia ohjelmia, ja testausta tarjoavat monet yritykset ostettavana palveluna. Myös yhdistykset, kuten Näkövammaisten liitto, tarjoavat palveluita ja opastusta tuotteen saavutettavuuden arviointiin ja kehittämiseen. Ostettavien palveluiden etuna on asiantuntijuus ja palvelun räätälöinti sopivaksi valmiilla menetelmillä ja mietityillä prosesseilla. Yrityksen sisäisen testauksen etuja ovat edullisuus verrattuna ostettavaan palveluun, sekä yhteistyön helppous testaajien ja kehittäjien välillä. Saavutettavuuden ja käytettävyyden optimoinnista hyötyvät kaikki käyttäjät, eivät ainoastaan ne henkilöt, joilla on rajoitteita. Saavutettavuuden huomioiminen kehitystyön alusta asti säästää kustannuksia ja aikaa, tehostaa kehitystyötä ja tuo myös mahdollisuuden parempaan taloudelliseen hyötyyn sekä ennen kaikkea luo hyvän käyttäjäkokemuksen.

Saavutettavuustestaus

Saavutettavuuden testauksen tarkoituksena on selvittää puutteet ja esteet digitaalisen palvelun tai tuotteen, kuten mobiilipelin, käytössä. Testaus voidaan

suorittaa useilla eri menetelmillä ja työkaluilla. Haaste on löytää menetelmät, jotka auttavat juuri oman tuotteen testaamisessa parhaiten. Manuaalista testausta ei pidä unohtaa, sillä runsaasta tarjonnasta huolimatta, mikään testausohjelma ei korvaa sitä täysin. Testiohjelmat eivät esimerkiksi kykene empaattiseen arviointiin. Testaamista tulisi tehdä koko tuotteen elinkaaren ajan, suunnittelusta valmiiseen julkaistuun tuotteeseen asti. (Nyqvist 2023b.)

Testaustyökalut voi karkeasti jaotella neljään kategoriaan, joita ovat selainpohjaiset työkalut, kokonaisia sivustoja testaavat työkalut, avustavan teknologian käyttö testauksessa sekä käyttäjättestaus. (Nyqvist 2023c.)

4 PELIN KEHITYSPROSESSI

Osiossa käymme läpi opinnäytetyössä käytetyimmät kommunikoinnin, kollaboraation ja mobiilipelin kehitystyön työvälineet. Lisäksi kappaleessa kuvataan peliprosessin etenemistä suunnittelun perusteista itse suunnitteluun ja valmista tuotetta imitoiviin, testattaviin ja pelattaviin prototyyppeihin asti.

4.1 Kehitysvälineet

Tässä opinnäytetyössä käytettiin erilaisia kommunikoinnin ja kollaboraation mahdollistavia työvälineitä. Ryhmän keskinäiseen viestittelyyn ja ajatusten vaihtoon käytettiin WhatsApp-sovellusta sekä Teams-sovellusta. Molemmilla sovelluksilla voi jakaa liitteitä ja kuvia, Teams-sovelluksella reaaliaikainen yhteistyö on vielä tehokkaampaa. Yhtenä työvälineenä käytetty Unity-pelimoottori keskittyy olennaisesti itse pelin kehitykseen. Figma puolestaan hyvänä visuaalisen työn välineenä, tarjoaa sekin monia keinoja kommunikointiin, esimerkiksi mahdollisuuden kommentoida tuotettua sisältöä. Tämän lisäksi Figma-sovelluksella voi tehdä sovelluksen tai tuotteen kehitystyötä alusta loppuun, suunnittelun alkuvaiheista prototyypitestaukseen asti.

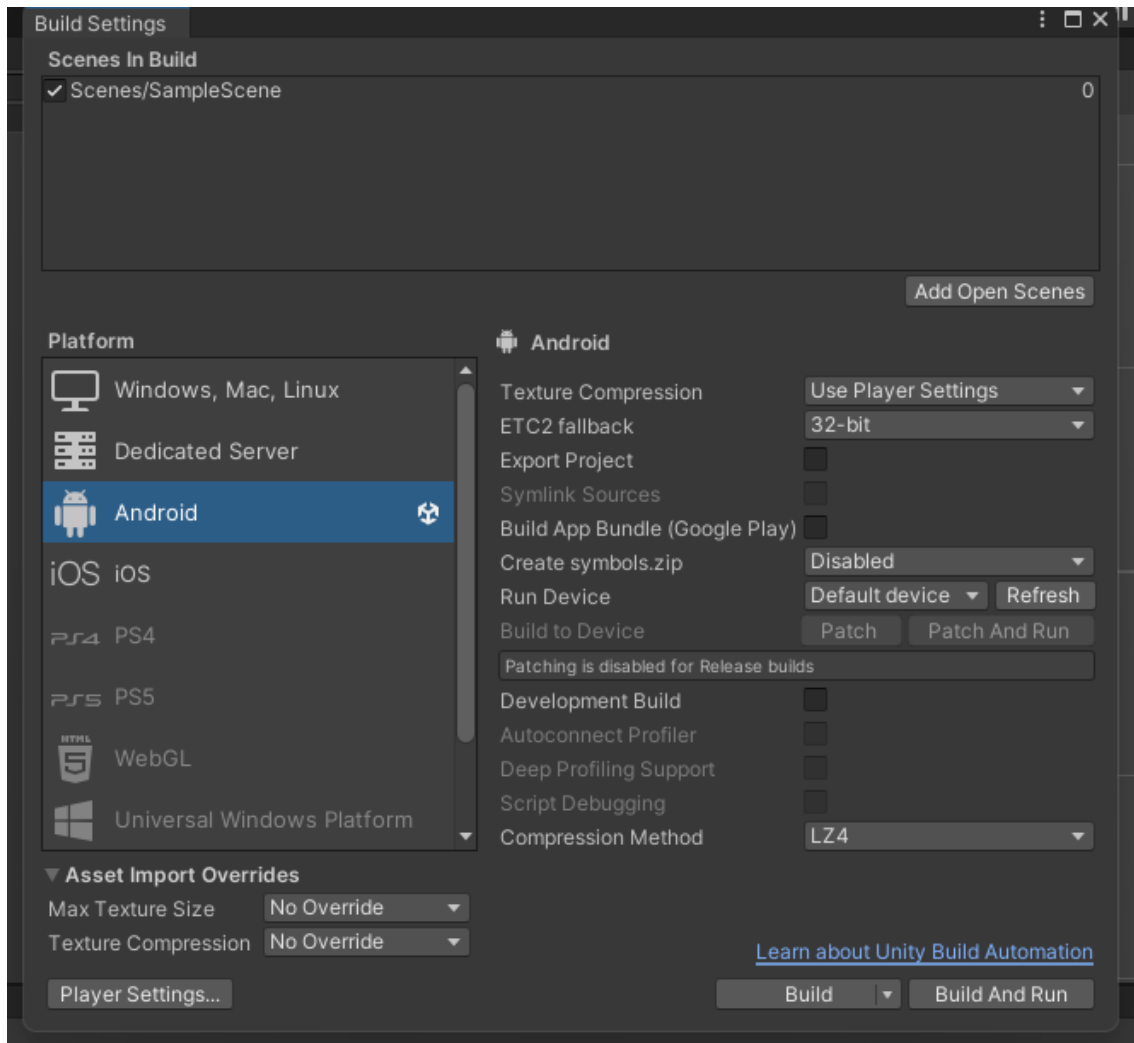
4.1.1 Unity

Pelikehityksessä usein keskeisessä roolissa on itse pelimoottori, vaikka pelien kehittäminen onkin mahdollista myös ilman. Näppärin voisi pelimoottorin tehdä itse, mutta valmiiden avoimien pelimoottorien käyttö on kuitenkin suosittua ja niitä on paljon, joten pyörää ei niin sanotusti tarvitse keksiä uudelleen.

Valmiin pelimoottorin käytössä on kehittäjälle laajasti etuja, kuten helppokäyttöiset työkalut ja nopeampi kehitysaika, koska itse kehitystyö ei vaadi yhtä laajaa koodaustaitoa. Lisäksi valmiisiin pelimoottoreihin on sisällytetty monialustatuki, joka mahdollistaa kehittämisen PC-, mobiili- ja konsolialustoille. Valmiit

pelimoottorit helpottavat lisäksi kehitystyötä sisäänrakennetun testausympäristön ja debuggaus-työkalujen ansiosta.

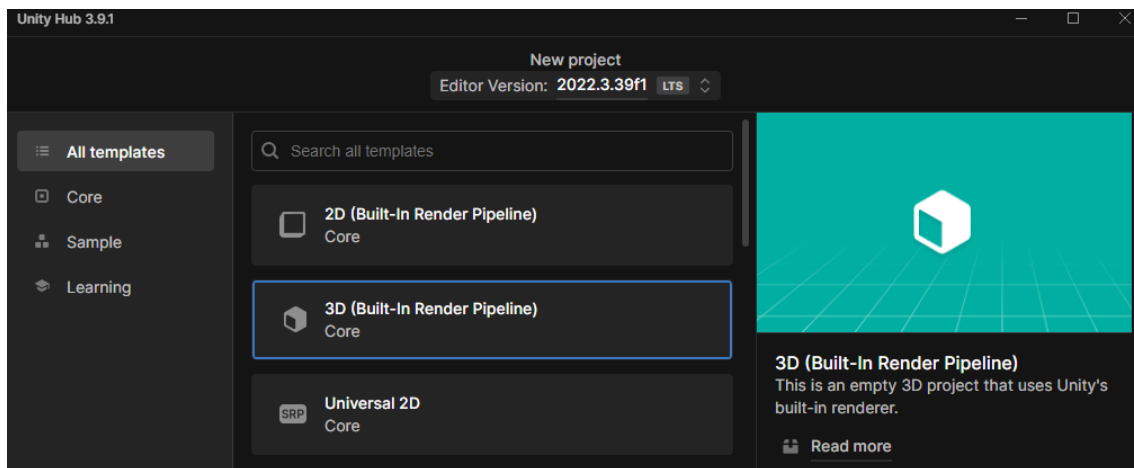
Itse mobiilipelin tekniseen toteutukseen tiimimme käyttöön valikoitui Unity Technologies-yrityksen kehittämä Unity-pelimoottori sen monipuolisuuden ja monialustaisuuden vuoksi. Unity antaa kehittäjälle mahdollisuuden valita alustan Build Settings -asetuksen kautta (kuva 4).



KUVA 4. Build Settings (Unity 2024)

Unity on tällä hetkellä suosituin pelimoottori heti kilpailijansa Unreal Engine-pelimoottorin jälkeen (Toftedahl 2019). Unity tukee sekä 2D- että 3D-grafiikkaa ja sen avulla on lisäksi mahdollista toteuttaa VR- ja AR-projekteja. Unityllä tuotettuja suosittuja pelejä ovat esimerkiksi Hollow Knight, Angry Birds, Pokemon Go ja Beat Saber (Drake 2023).

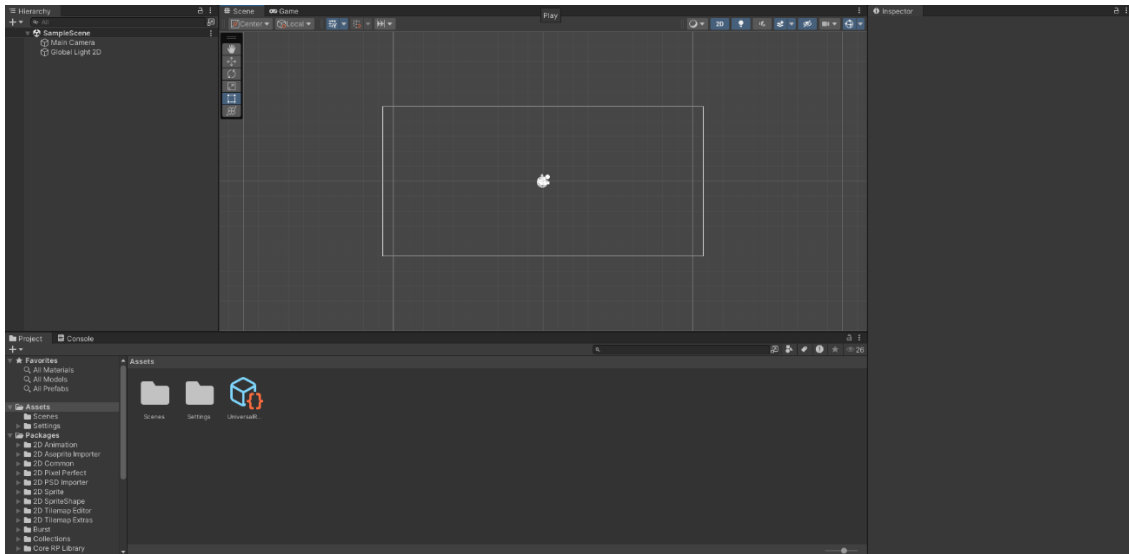
Unity-pelimoottorilla liikkeelle lähdetään tyypillisesti Unity Hub-ohjelman asennuksesta ja editorin valinnasta, johon asennetaan tarvittavat komponentit kuten alustatuet, versionhallinta-työkalut, Visual Studio koodausta varten, Asset store integraatio ja muut tarvittavat työkalut. Kun tarvittavat asennukset on tehty, on aika luoda uusi projekti, jonka pohjaksi voidaan valita valmiita pohjia, jotka sisältävät esiasetuksia, riippuen kehittäjän tarpeista (kuva 5).



KUVA 5. Unity-projektin luominen (Unity 2024)

Aloittelijalle Unityn opettelu voi tuntua hieman haasteelliselta ja sen ymmärtäminen on paljon kiinni kehittäjän omasta ohjelmointitaustasta. Apua Unityn ymmärtämiseen voi olla muun muassa C#-kielen perusteiden hallinnasta, mutta sekään ei ole täysin pakollista, mikäli hallitsee muita ohjelmointikieliä. Adoben työkalujen hallinnasta on myös hyötyä, erityisesti animointeja tehdessä voi huomata samankaltaisia ominaisuuksia kuin esimerkiksi Adobe Animate-työkalussa.

Unityn käyttöliittymä (kuva 6) koostuu lyhyesti neljästä tärkeimmästä paneelista: hierarchy, inspector, scene ja project.

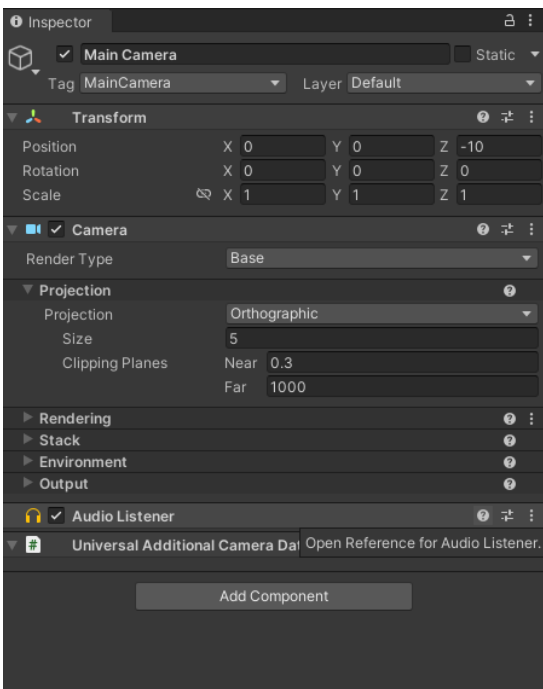


KUVA 6. Unity-editorin käyttöliittymä (Unity 2024)

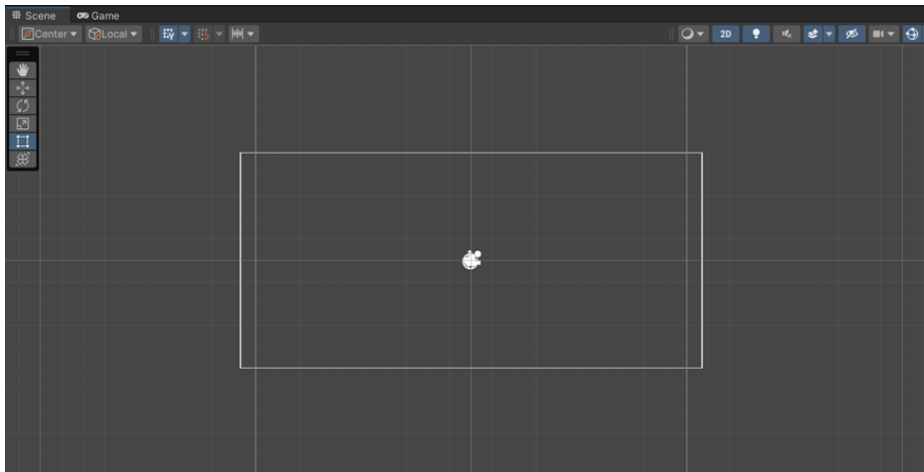
Hierarchy (kuva 7), joka sisältää työstettävän scenen sisällön, pelissä näitä kutsutaan yleisemmin tasoiksi. Hierarchy-paneeliin luodaan muun muassa peliobjektit, joiden ominaisuuksia muokataan Inspector-paneelissa (kuva 8). Tällaisia ominaisuuksia voivat olla esimerkiksi esineen painovoima, materiaali, koko, asema ja animaatiot sekä koodit, joilla peliobjektien liikkeitä ja käyttöä tehdään pelinkulkuun sopivaksi. Scene-paneeli (kuva 9) sisältää tehtyjen muutosten tuloksen ja mahdollistaa myös peliobjektien liikkuttelun manuaalisesti. Scene-paneelin kyljessä oleva Game-paneeli mahdollistaa pelin testauksen ja sisältää pelin UI-näkymän, eli sen miltä pelin käyttöliittymä pelaajalle näyttää. Game-ikkunalle vaihtaessa peliä on mahdollista testata tehtyjen muutosten jälkeen. Project-paneeli (kuva 10) sisältää kaikki tiedostot mitä peli tai sovellus itsessään sisältää, kuten kuvat, materiaalit, ääniefektit, fontit ja koodit.



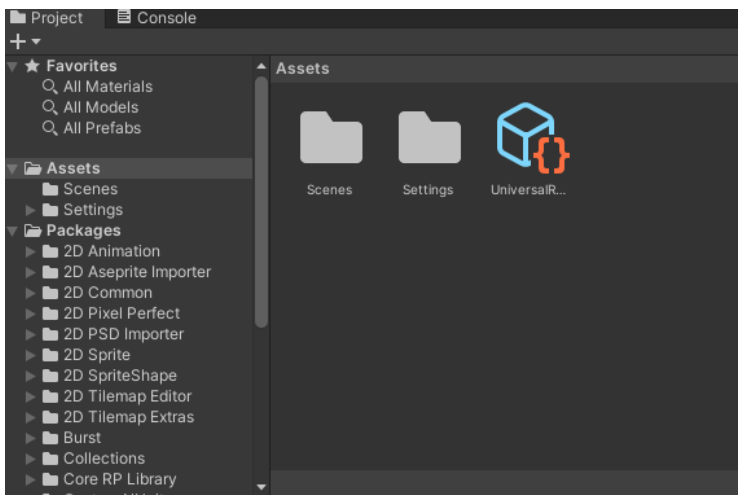
KUVA 7. Hierarchy (Unity 2024)



KUVA 8. Inspector (Unity 2024)



KUVA 9. Scene (Unity 2024)



KUVA 10. Project (Unity 2024)

4.1.2 Figma

Figma on yleisesti käytettävä työkalu, jota käytetään erityisesti apuna käyttöliittymien suunnittelussa web- ja mobiilisovelluksiin. Sen edut ovat saumaton ryhmätyöskentely ja mahdollisuus tehdä sovellusten design vastaamaan lähes täysin lopullista sovellusta. Figma-sovelluksella on myös mahdollisuudet tuottaa laadukkaita prototyyppejä ja kehittää käyttäjäkokemusta projekteissa, ennen varsinaista teknistä toteutusta. Figma-sovellusta voi käyttää selaimella, työpöytäsovelluksena tai mobiilisovelluksena. Työt tallentuvat aina automaattisesti, joten erillistä työn tai tiedoston lataamista tai useampien versioiden luomista ei tarvita.

Vaikka Figma ei ole laajalti käytössä pelisuunnittelun työkaluna, ryhmämme valitsi sen yhdeksi projektin työvälineeksi sen ominaisuuksien vuoksi, jotka ovat hyödyllisiä erityisesti UI/UX-suunnittelun puolella. Koska projekti keskittyy erityisesti mobiilipelien käyttäjäkokemukseen, halusimme hyödyntää työkalua mm. ulkoasun, asettelun ja käyttäjäpolun suunnittelemisessa.

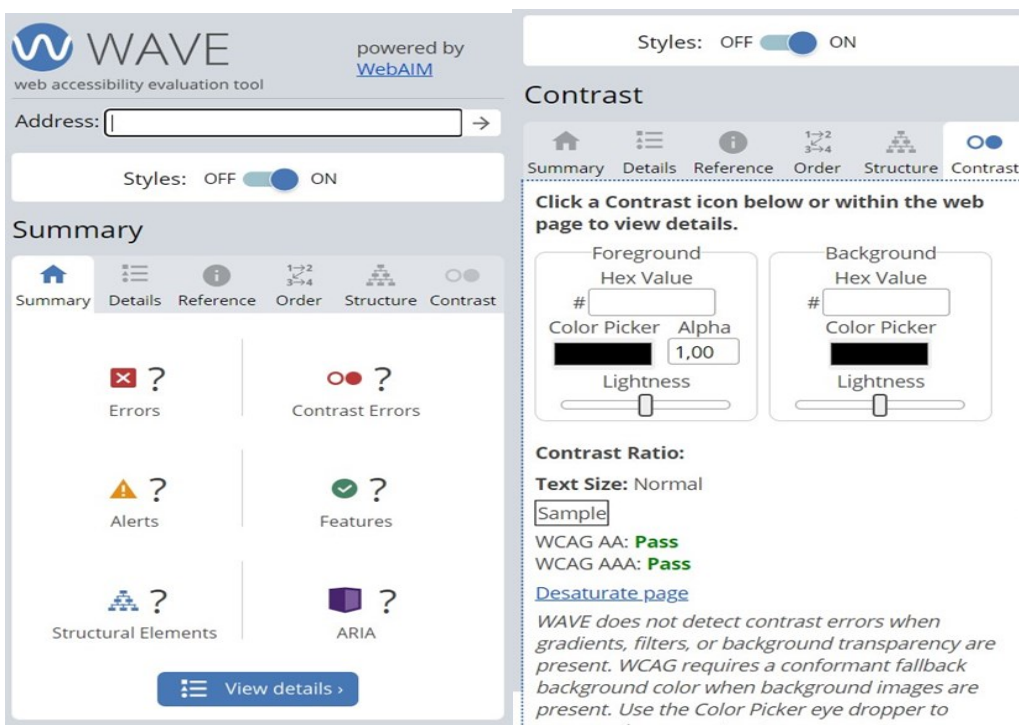
FigJam on yksi Figma-sovelluksen työkalu. Alustalla henkilöt voivat osallistua reaaliaikaisesti tai yksittäin erilaisiin kokouksiin, suunnittelusessioihin ja tutkimukseen. Näkymä on avoin piirtoalusta, johon kukin osallistuja voi lisätä tekstiä, muotoja, piirroksia, tarroja tai muistilappuja. Alustaan voi myös tuoda omia kuvatiedostoja ja kaavioita tai käyttää sovelluksen tarjoamia, valmiiksi muotoiltuja ja mietittyjä pohjia. Elementtejä voi lisätä alustalle visuaalisten ideoiden esittämiseen ja yhdessä työskentelyyn. Elementtejä ja tiedostoja voi myös jakaa sovelluksen ulkopuolelle tarvittaessa ja apuna suunnittelussa voi käyttää myös tekoälyä. Alustalle voi kerätä omia kokoelmia tulevia projekteja varten tai vaikkapa valmiita väriprofiileja helpottamaan luovaa työtä. Lisäksi tiimit voivat ladata äänitiedostoja tai kustomoituja fontteja tai fonttikokoelmia töitään varten. (Figma Learn 2024.)

4.1.3 Selainpohjaiset testityökalut

Selainpohjaisia testaustyökaluja voidaan hyödyntää tässä opinnäytetyössä käytetyn Figma-työkalun kanssa. Selainpohjaisia testaustyökaluja ovat mm. SiteImprove, Google Lighthouse (kuva 11), Wave (kuva 12) ja Spectrum. Nämä kaikki ovat ilmaisia yleistyökaluja, jotka testaavat yhden sivun kerrallaan ja ovat ladattavissa selaimelle tai löytyvät selaimista jo valmiina, kuten Chromen Google Lighthouse. Kyseisen työkalun etuna on erillinen mobiilitesti sekä samanaikaisesti tehtävät nopeus- ja SEO-testit. Wave puolestaan näyttää löydettyjen virheiden hyvän visualisoinnin lisäksi alt-tekstit sekä otsikoiden tasot. SiteImprove näyttää virheet luokittain, eli sisällöntuottajaa, sivuston suunnittelijaa ja kehittäjää koskevat virheet on eritelty. Spectrum-työkalulla voi testata sivuja värisokeuksien kautta ja matalan kontrastin näkymänä. Näiden lisäksi maininnan ansaitsee Firefoxin kontrastityökalu, jolla voi testata myös taustakuvaa, varsinaisten kuvien lisäksi. (Holopainen 2023a.)



KUVA 11. Selaimen Lighthouse-työkalu (Chrome Developer tools 2024)



KUVA 12. Wave-työkalun saavutettavuustyökalu (Wave Chrome extension 2024)

W3C on avoin ja ilmainen validointi- ja testityökalu, joka on kaikkien käytettävissä. Sivun tai tiedoston validointi tapahtuu syöttämällä sivun osoite eli URL, lataamalla palveluun testattava ja validoitava sivu, tai suoraan kirjoittamalla työkalun tyhjään tekstikenttään testattava sisältö. Tämän jälkeen valitaan “Check” ja näin sivu käy läpi valitun prosessin. Palvelussa on valittavana muutamia lisäominaisuuksia, kuten koodin siistiminen. (W3C Markup Validation Service 2024.)

Kokonaisten sivustojen testaamiseen käytettävät työkalut ovat yleensä maksullisia. Niiden käyttö on kuitenkin suositeltavaa, jos halutaan saada kokonaiskuva

sivun saavutettavuudesta tai saavutettavuuden kehitystä halutaan mitata säännöllisesti. Näitä työkaluja ovat Wave, Axe tools, SiteImprove ja Google Lighthouse. Mainitut työkalut omaavat laajemmat testauskyvyt kuin samojen toimijoiden selainpohjaiset työkalut. Näistä viimeisin sisältää sekä ilmaisia, että maksullisia kolmannen osapuolen palveluja. (Holopainen 2023b.)

4.2 Pelisuunnittelun perusteet ja pelisuunnitelman luominen

Hyvä lähtökohta aloittelevalle pelintekijälle on tutustua peliteorian perusteisiin ja sen terminologiaan, tämä auttaa jäsentämään ideat ensin hallittavamiksi kokonaisuuksiksi ennen varsinaista toteutusta. Ensimmäisiä asioita pelin suunnittelussa, on tunnistaa pelin ydin (engl. core). Ydinsilmukka (engl. coreloop) on yksi määrittävä ja yhdistävä lause, joka kuvaa peliä. Vahvan ydinlauseen kehittäminen voi helpottaa ja tehostaa pelin suunnitteluprosessia. Ydinlauseet laaditaan usein ”kuka-mitä-missä” – tyylinä lauseina, jotka kuvaavat, keitä pelaajat ovat, missä he ovat ja mitä he pelissä tekevät. Ydinlauseen jälkeen suunnitellaan lista ominaisuuksista, jotka kuvaavat yksityiskohtaisemmin pelin sisältöä. Ominaisuudet tukevat pelin ydinlauseita ja auttavat peli-idean kehittämisessä, muovaten pelin kokonaisuutta. (Menard 2012, 47–50.)

Pelin peruselementti on mekaniikka (engl. mechanic), joka määrittelee yksittäisen pelin osa-alueen, kuten pisteiden keräämisen tai nappulan painamisen toiminnon suorittamiseksi. Samankaltaiset ja toisiinsa liittyvät mekaniikat yhdistyvät laajemmiksi sääntöjoukoiksi, joita kutsutaan järjestelmiksi (engl. system). Hyvin suunnitellut ja mielikuvitukselliset järjestelmät muodostavat minkä tahansa hauskan pelin ytimen. Vaikka peleissä voi olla samankaltaisia järjestelmiä, juuri yksittäiset mekaniikat tekevät jokaisesta pelistä ainutlaatuisen. (Menard 2012, 48.)

Suunnittelun eri työvaiheissa voidaan hyödyntää monia menetelmiä, kuten brainstormausta, vapaata kirjoittamista sekä satunnaisia hakuja verkosta tai fyysisistä kirjoista sekä muiden pelien tutkimista. Paperiprototyypit ovat hyvä työkalu nopeaan ja edulliseen käyttäjätestaamiseen. Saadun palautteen perusteella voidaan tehdä muutoksia jo varhaisessa suunnitteluvaiheessa, ennen varsinaista

pelin ohjelmointia ja toteutusta. Paperiprototyypointi mahdollistaa nopean iteroinnin. (Menard 2012, 51-54.)

4.3 Viihdyttävän pelikokemuksen tekijät

Tässä opinnäytetyössä viihdearvolla tarkoitetaan niitä ominaisuuksia ja tekijöitä, jotka tekevät mobiilipelistä hauskan ja kiehtovan pelaajalle. Se kuvaa pelin kykyä tarjota kiinnostava kokemus, joka saa palaamaan pelin pariin toistuvasti ja perustuu kykyyn yhdistää erilaisia viihdyttäviä elementtejä tavalla, joka houkuttelee pelaajia ja kannustaa heitä jatkamaan pelaamista.

Ihmiset pitävät pääasiassa enemmän järjestyksestä, kuin kaaoksesta. Liiallinen kaaos koetaan epämiellyttäväksi. Liiallinen järjestyskään ei kuitenkaan ole hyvä, vaan järjestystä ja sääntöjä sekä vaihtelua ja yllätyksiä pitäisi olla sopivassa suhteessa. (Koster 2014, 24.)

Tylsinä koetut pelit ovat sellaisia, jotka eivät stimuloi aivoja riittävästi. Kun pelin kaavat (engl. patterns) tulevat tutuksi, tarvitaan jotain uutta, jotta kiinnostus peliin säilyisi. Yksinkertaiset pelit, kuten ristinolla, on hyvä esimerkki pelistä, jonka viehäytys loppuu nopeasti. Kyseinen peli on hyvin rajoitettu, eikä tarjoa mitään uutta, kun pelin kaavan on oppinut. Peliä suunnitellessa on tärkeää ymmärtää, että tiukemmat säännöt luovat rajoittuneemman pelin. Kiinnostusta pitää yllä erityisesti tietynlainen sisältö, joita ovat muun muassa matemaattiset haasteet, joihin pelaaja ei tiedä ratkaisua tai ennakoimattomat muutokset, kuten fysiikan ilmiöt ja psykologia. (Koster 2014, 38.)

Hauskaksi koettu peli tuottaa hyvää oloa ja vapauttaa siis endorfiineja keuhomme. Uuden oppiminen, pulmien ratkaiseminen ja haasteissa onnistuminen aiheuttaa hyvän olon tunteita. (Koster 2014, 40.)

Viihdyttävä peli stimuloi aivoja sopivissa määrin, erilaisten haasteiden ja onnistumisien kautta, aiheuttaen hyvän olon tunteita. Uuden oppiminen ja erityisesti ongelmien ratkaiseminen on yksi tärkeä palanen siinä, mikä tekee pelistä hauskan. Tylsä peli taas on sellainen, joka ei tarjoa meille uutta opittavaa eikä uusia haasteita.

4.4 Prototyyppi mobiilipelin käyttöliittymästä

Prototyyppi on tuotteesta tehtävä vajaa versio, jonka avulla idean tai tuotteen toimivuutta voidaan testata kohderyhmällä varhaisessa vaiheessa (Loijas, Mäkeläinen & Pippola 2023). Prototyyppien päätarkoituksena on osoittaa toimimattomat aspektit ennen kuin pelin, joka ei olekaan hauska, kehittämiseen on käytetty runsaasti aikaa. Huono palaute ei tarkoita kuitenkaan huonoa ideaa, vaan ideoita on syytä tarkastella uusiksi ja hyödyntää aktiivista iteroivaa kehittämisotetta alusta lähtien. Nopea prototypointi ja testaus onnistuu hyödyntämällä jo olemassa olevia ideoita ja työvälineitä. (Promark Business Solutions, 2021.)

Mobiilipelien prototypoinnissa Figma on saavuttanut suosiota myös pelien kehitystyössä. Mobiilipelin kehittämisessä on jopa kriittistä onnistua responsiivisessa suunnittelussa, ja Figman auto-layout sekä responsiivisuuteen liittyvät ominaisuudet koetaan erityisen hyödyllisinä. Tämän lisäksi pilviperustaisuus ja reaaliaikainen kollaboraatiomahdollisuus ovat suunnittelijoiden ja kehittäjien mieleen. (Kruchkow, 2024.)

5 IDEASTA PELIKSI

Kipinä Kuumat Kinkut– peli-ideasta syntyi jo keväällä 2023. Reilun vuoden ajan tämä idea kulki opintojemme mukana ja keväällä 2024 aloimme pohtia olisiko tästä opinnäytetyön aiheeksi ja peliksi saakka.

5.1 Ideointi ja suunnittelu

Suunnitteluvaiheessa lähdimme ensin työstämään ideaa eteenpäin miettien millainen pelimme olisi. Meillä kaikilla oli mielessämme saunova kinkku, jolle heitellään löylyä. Kinkku ei saa kuivahtaa ja sitä voi valella erilaisilla kastikkeilla. Pelin ydinlauseeksi muodostui ”Pelaaja pitää kuumaa kinkkua mehukkaana saunan lauteilla, heittää löylyä ja valitsee kastikkeita välttämällä kinkun kuivumista ja keräten pisteitä.”

Seuraavaksi mietimme tarkemmin pelin ominaisuuksia, pelimaailmaa ja päätavoitetta. Keskeisimmät ominaisuudet ovat lämpö- ja kosteusmittarit sekä kinkun mehukkuusmittari. Löylyä heittämällä vaikutetaan lämpöön ja kosteuteen, kastiketta valelemalla kosteuteen ja mehukkuuteen. Kiuas on kinkun lisäksi keskeinen elementti. Löylykauha ja kastikesuti ovat pelaajan käytössä olevat työkalut. Pelaajan tehtävänä on säätää saunan lämpötilaa heittämällä löylyä ja valita oikeat kastikkeet, jotta kinkku pysyy kosteana eikä kuivu. Kinkun ulkonäkö muuttuu riippuen sen mehukkuudesta ja lämpötilasta. Jos kinkku alkaa kuivua, se muuttuu ruskeaksi ja kutistuneeksi, kun taas mehukas kinkku säilyttää punertavan ja kimmoisan ulkomuotonsa. Visuaalisella palautteella autetaan pelaajaa arvioimaan tilannetta. Pisteitä keräännyttyä pelaajan onnistuessa pitämään kinkun optimaalisessa mehukkuustilassa mahdollisimman pitkään. Mitä mehukkaampi kinkku, sitä paremmat pisteet. Pelin päätavoite on siis pitää kinkku mahdollisimman mehukkaana mahdollisimman pitkään. Kirjasimme nämä suunnitteludokumenttiin, josta löytyy kattavampi listaus ominaisuuksista.

Ideoinnissa hyödynsimme muun muassa brainstormausta ja paperiprototyyppejä. Brainstormauksessa työvälineenä hyödynsimme Figman FigJam -tiedostoa (kuva 13), jonne keräsimme ideat, ajatukset, luonnokset, teorialähteitä ja niin edelleen. Tutustuimme myös itse erilaisiin peleihin alkuvaiheessa, jotta saimme tuntumaa, millaisia ovat suosituimmat mobiilipelit ja toisaalta myös avarsimme näkemystämme kokeilemalla muitakin pelejä ja pelilaitteita.



KUVA 13. Figman FigJam – tiedoston ideoinnit

5.2 Kysely ja analyysi

Tutkimme yhtenä näkökulmana asioita, jotka tekevät peleistä viihdyttäviä ja hauskoja, sekä millaiset asiat saavat palaamaan pelaamisen ja pelin pariin.

Opinnäytetyön osana toteutettiin pienimuotoinen anonyymi kysely, jonka tavoitteena oli selvittää, mitkä tekijät tekevät peleistä viihdyttäviä pelaajille. Kysely suunnattiin eri-ikäisille pelaajille ja sen avulla pyrittiin saamaan kuva pelaajien mieltymyksistä ja motivaatiotekijöistä. Kyselyyn osallistui yhteensä 48 vastaajaa. Kyselyssä esitettiin kolme monivalintakysymystä, jotka kartoittivat pelaajien

näkemyksiä muun muassa pelien tarinan, haasteiden, sosiaalisen vuorovaikutuksen ja visuaalisen ilmeen merkityksestä pelikokemuksen viihdyttävyyteen, sekä mitkä tekijät motivoivat pelaajia palaamaan pelin pariin.

Kyselyn tulosten mukaan pelin viihdyttävyyteen vaikuttavat eniten mielenkiintoinen tarina, sosiaalinen vuorovaikutus sekä pelin tarjoama rentouttava kokemus. Vastaajista 68,8 % piti tärkeimpänä viihdyttävyyden tekijänä pelin tarinaa, ja 64,6 % arvosti mahdollisuutta pelata muiden kanssa. Tämä korostaa tarinallisuuden ja sosiaalisen vuorovaikutuksen merkitystä pelaajien sitouttamisessa. Pelissä tarjottavat haasteet, kuten tehtävät ja pulmat, olivat myös suosittuja (56,3 %), mikä osoittaa, että pelaajat arvostavat älyllisesti stimuloivia elementtejä. Toisaalta graafinen ilme ja äänisuunnittelu (31,3 %) eivät olleet ensisijaisia tekijöitä, vaikka niitä pidettiin tärkeinä tukemaan pelikokemusta. Kyselyyn vastanneet kokivat visuaalisen ilmeen kuitenkin merkityksellisenä, sillä 70,8 % piti sitä jossain määrin tärkeänä viihdyttävälle pelikokemukselle. Tämä viittaa siihen, että visuaalinen ilme tukee pelin houkuttelevuutta, vaikka se ei ole pääasiallinen viihdyttävyyden lähde. Peliin palaamisen tärkeimmiksi motivaatiotekijöiksi nousivat rentouttava ajanviete (89,6 %) ja sosiaalinen vuorovaikutus (58,3 %). Tämä korostaa pelien roolia rentoutumisen välineenä ja sosiaalisten kokemusten tarjoajana. Uusien tasojen tai sisältöjen avaaminen motivoi pelaajia jonkin verran, mutta kilpailulliset tavoitteet, kuten korkeampien pisteiden saavuttaminen, olivat selvästi vähäisempiä motivaatiotekijöitä.

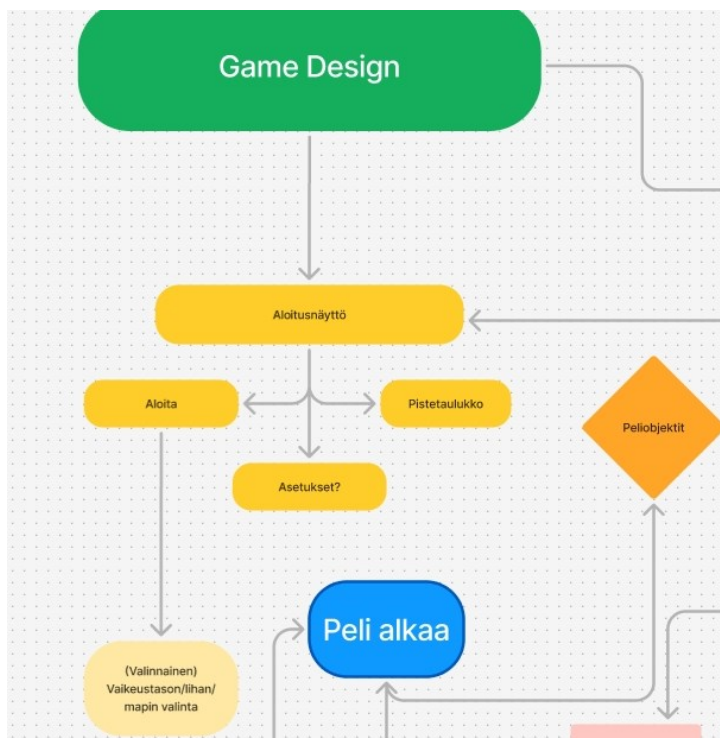
Tulokset osoittivat, että useimmille pelaajille pelin viihdyttävyys perustuu vahvasti emotionaalisiin ja sosiaalisiin tekijöihin, kun taas kilpailullisuus ei ole yhtä merkittävä tekijä.

5.3 Pelin tekninen eteneminen

Kun ideaa oli saatu suunniteltua pitkälle, oli aika ryhtyä pohtimaan pelin teknistä toteutusta. Helppointa oli aloittaa pohtimalla pelin kulkua userflow-mallina, jossa pelin etenemistä pohdittiin käyttäjän näkökulmasta. Userflow-kaavio on UI/UX-suunnittelussa suosittu työkalu, jolla halutaan visualisoida käyttäjän kulkemaa matkaa sovelluksen läpi. Sen avulla voidaan askel askeleelta selvittää

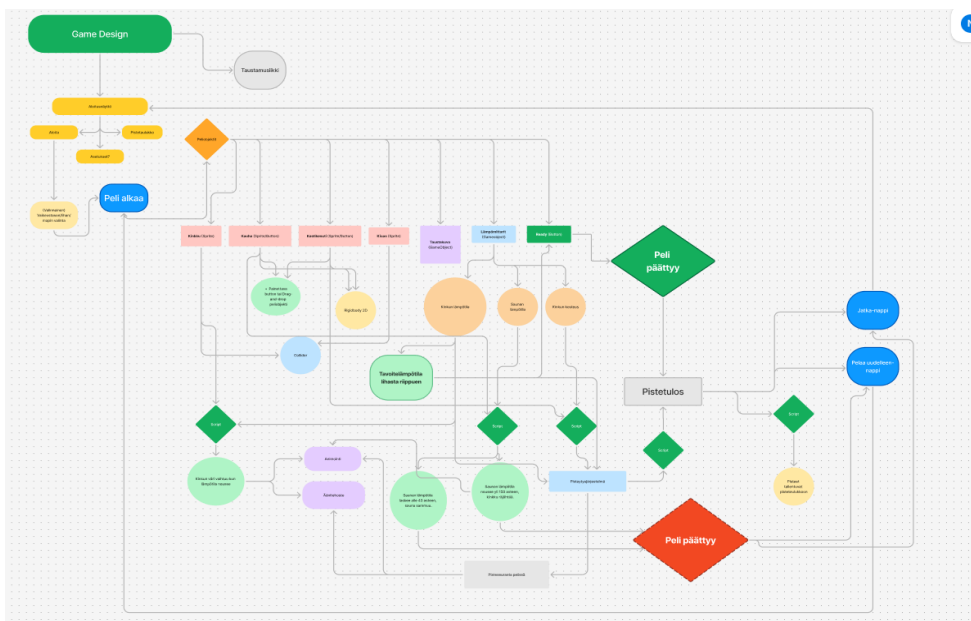
esimerkiksi, kuinka monta vaihetta käyttäjä käy läpi sovelluksessa ennen kuin saavuttaa tavoitteensa. Kaavion tavoitteena oli paitsi kehitystyön helpottaminen pelimekaniikkaa visualisoimalla, mutta myös navigoinnin helppouden arviointi käyttäjän näkökulmasta.

Suunnittelun työkaluna käytettiin apuna Figman FigJam-alustaa, jolle oli helppo luoda pelin etenemiseen liittyviä vaiheita (kuva 14) ja lisäksi tehdä muistiin Unity-ohjelman tarvittavien ominaisuuksien paikat ja tarpeet.



KUVA 14. Game Design -tiedosto

Lopputuloksena oli flow-tyylinen kartta (kuva 15), jossa pelin logiikkaa ja etene- mistä kuvattiin muotokuvakkeilla, joiden tarkoituksena oli visualisoida pelissä ta- pahtuvaa ominaisuutta, elementtiä tai käyttäjän päätöksentekoa. Kaavioon visu- alisoitiin muun muassa Unity-ohjelmaan luotavat scene-näkymät, kuten aloitus- näyttö, josta pelin voi aloittaa, ja joka sisältäisi myös asetukset ja pistetaulukon. Pelin aloittamisen jälkeen näkyviin tulee itse pelin scene-näkymässä olevat as- setit ja osa niiden ominaisuuksista. Käyttäjän pelin aikana tekemistä päätöksistä muodostui kartta ja siihen yhdistettiin kuvakkeet, jotka kuvaavat osaa tarvittavista koodista eli script-ominaisuuksista.



KUVA 15. FigJam-työkalu (Figma 2024)

5.4 Prototyyppi

Peli-ideaa alettiin hahmotella visuaalisesti ensimmäisenä paperiprototyyppien piirtelyllä. Ensin piirtelimme muutamia nopeita wireframe-hahmotelmia paperille yhdestä tai useammasta näkymästä (kuva 16) Meille oli jo muotoutunut pelistä ajan myötä melko samanlaisia mielikuvia, kovin suuria eroavaisuuksia ei tullut esille. Tässä vaiheessa päädyimme valitsemaan tähän hetkeen pelin käyttöliittymän orientaatioksi pystykuvan. Kääntömahdollisuutta emme halunneet tässä vaiheessa suunnitella mukaan ollenkaan, koska tämä olisi laajentanut työmäärää ja lisännyt haastavuutta turhan paljon tämän opinnäytetyön yhteyteen. Mietimme jo melko tarkasti elementtien ja toimintojen määrää, jotta käyttöliittymä ei tuntuisi

liian ahtaalta ja sekavalta. Jokaisen hahmotelmassa oli tuotu esille eri tavoin mitarit ja jokin tapa näyttää pelin edistymisen vaihe. Myös painikkeita löylykauhalle ja kastikesudille oli mietittynä. Kolmannen napin tarkoitus on antaa käyttäjälle mahdollisuus valita, milloin kinkku on käyttäjän mielestä sopivan mehevä eli valmis.



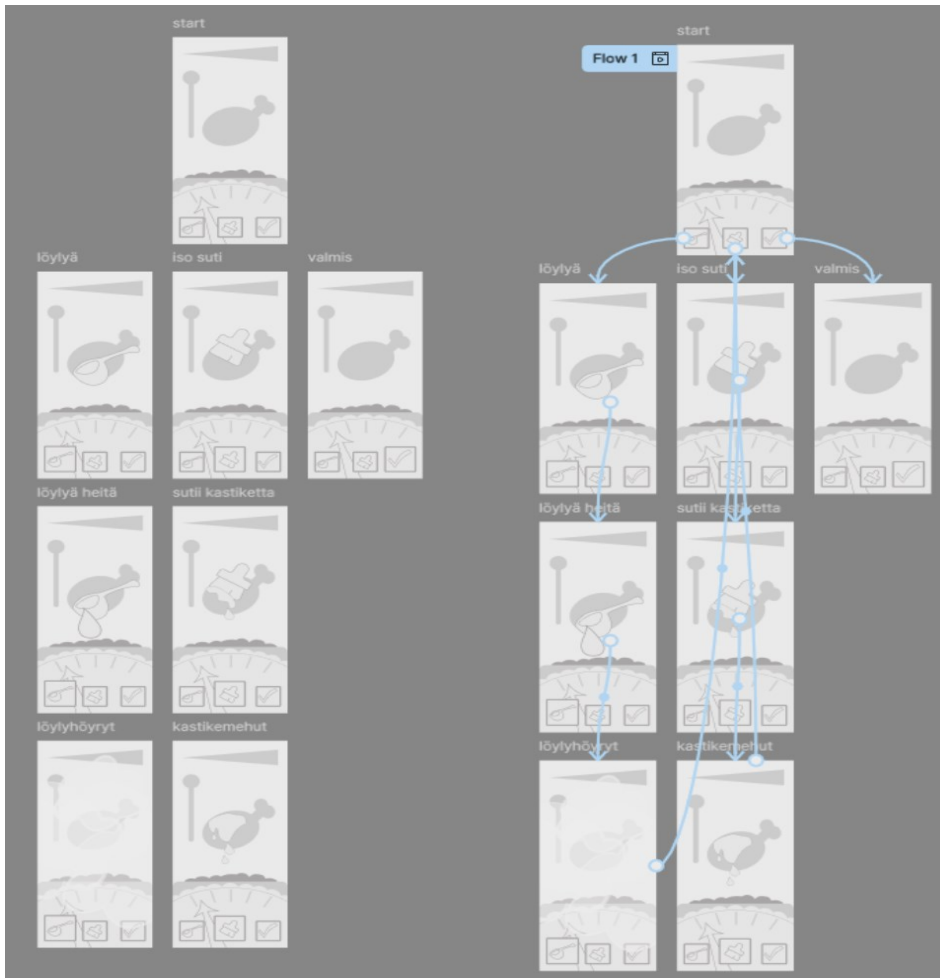
KUVA 16. Ensimmäiset paperi-wireframe-hahmotelmat

Tässä vaiheessa peli-ideasta oli myös keskustelua lähipiirin kanssa ja esitimme ensimmäisiä luonnoksia ulkopuolisille. Lähipiirimme koostui lapsista, aikuisista ja eläkeikäisistä. Saatu palaute oli kohtalaisen positiivista ja kannustavaa, kinkkupeli tuntui hausalla tavalla huvittavalta eri ikäryhmien mielestä. Erityisesti lapset antoivat hyvin avoimesti palautetta, tunnistivat toiminnallisuuksia ja elementtejä sekä lähtivät jo tavoittelemaan pelaamista painellen nappien kuvia ja sutien kastiketta kinkulle ilman erillisiä ohjeita.

5.4.1 Ensimmäinen versio

Paperille piirretyistä rautalankamalleista yhdistelimme sopivia elementtejä ja ideoita yhteen ensimmäistä low-fidelity-prototyyppiä varten. Figmalla toteutettiin yksinkertainen ja nopea prototyyppi löylyn heitosta sekä kastikesudin käytöstä

(kuva 17). Tässä versiossa pystyi valitsemaan joko löylykauhan tai kastikesudin. Valinnan jälkeen työväline ilmestyy näytölle ja työvälinettä napauttamalla toiminto eteni. Löylykauha kaataa vettä kiukaaseen ja kiukaasta nousee hetkeksi höyryä, tämän jälkeen näkymä palaa tilaan, jossa työvälinettä ei ole valittuna. Kastikesuti lisää kinkulle kastiketta ja samalla kosteuttaa kinkkua. Statuksesta indikoivia mittareita oli tässä testiversiossa näkyvillä kolme.

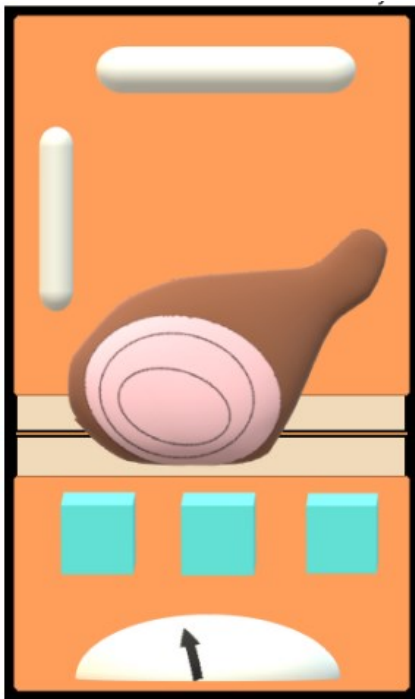


KUVA 17. Ensimmäinen Figma lo-fi –prototyyppi (Figma 2024)

Ensimmäisessä testiversiossa ei vielä varsinaisesti muita toiminnallisuuksia ollut, ainoastaan löylykauhan ja kastikesudin käyttö karkeasti. Tällä saimme kuitenkin hyvin jo mielikuvaa, minne suuntaan kannatti seuraavaksi peliä viedä eteenpäin. Pystyimme keräämään myös arvokasta käyttäjäpalautetta heti alkuvaiheessa. Testikäyttäjien mukaan ensimmäinen nopea prototyyppi vastasi melko hyvin paperisista luonnoksista syntynyttä mielikuvaa. Yhden testikäyttäjän mielestä statuksesta viittaavien mittareiden sijoittelu hyödyntäen koko ruutua oli hyvä idea, jotta peliin tulee hieman lisää haastetta.

5.4.2 Toinen versio

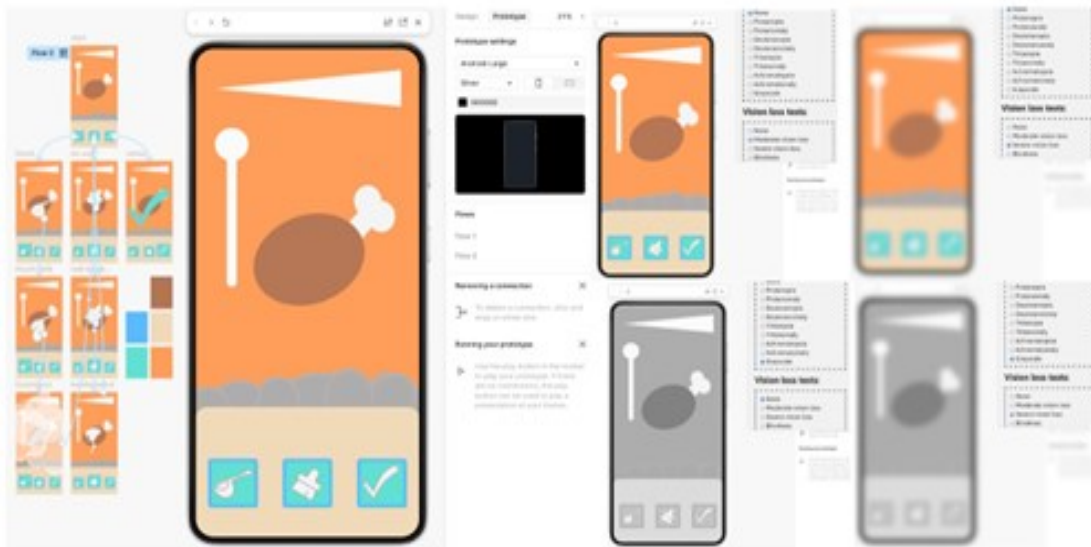
Figman prototyypeissä etenimme vielä pääasiassa hyödyntäen Figman muoto-työvälineen tarjoamia valintoja, jotta meille hahmottuisi tässä vaiheessa lähinnä käyttöliittymän asettelu ja sopiva elementtien määrä. Lisäsimme toiseen versioon värikokeilut (kuva 18) sekä kokeilimme ilmapampaa ratkaisua erityisesti alareunan näkymään. Ilmavuudella haettiin painikkeiden parempaa sijoittelua motorisia haasteita omaavan pelaajan kannalta. Tällä versiolla testasimme pienimuotoisesti myös saavutettavuutta hyödyntäen selaimen asennettavaa lisäosaa. Lisäosalla saimme testattua näkökyvyn heikkenemisen vaikutuksia sekä tehtyä harmaasävytestausta.



KUVA 18. Ensimmäinen värihahmotelma

Saavutettavuustestauksessa ilmeni, että värit erottuvat riittävästi sekä värillisessä versiossa, että myös mustavalkoisessa tai harmaassa testinäkyvässä (kuva 19). Värivalinnassa oli huomioitu erityisesti puna-vihersokeus sekä elementtien erottuminen taustasta. Painikkeiden siirrosta näkyvässä ylemmäs oli keskustelua. Myös valmis-painikkeen sijoittaminen keskelle oli yksi ehdotuksista. Tämä siksi, että reunoilla olevien kauhan ja sudin käyttö ovat pelin päätoiminnallisuuksia ja niiden käyttö tulee olla mahdollisimman helppoa motorisia ja hahmottamisen haasteita omaavalle henkilölle. Painikkeissa on käytetty samaa

pohjaväriä ja ne on myös ryhmitelty samalle alueelle indikoimaan niiden toiminnallisuutta eli CTA-ominaisuutta (engl. call to action- ominaisuus).



KUVA 19. Toinen versio prototyypistä ja version saavutettavuustestaus

5.5 Demo

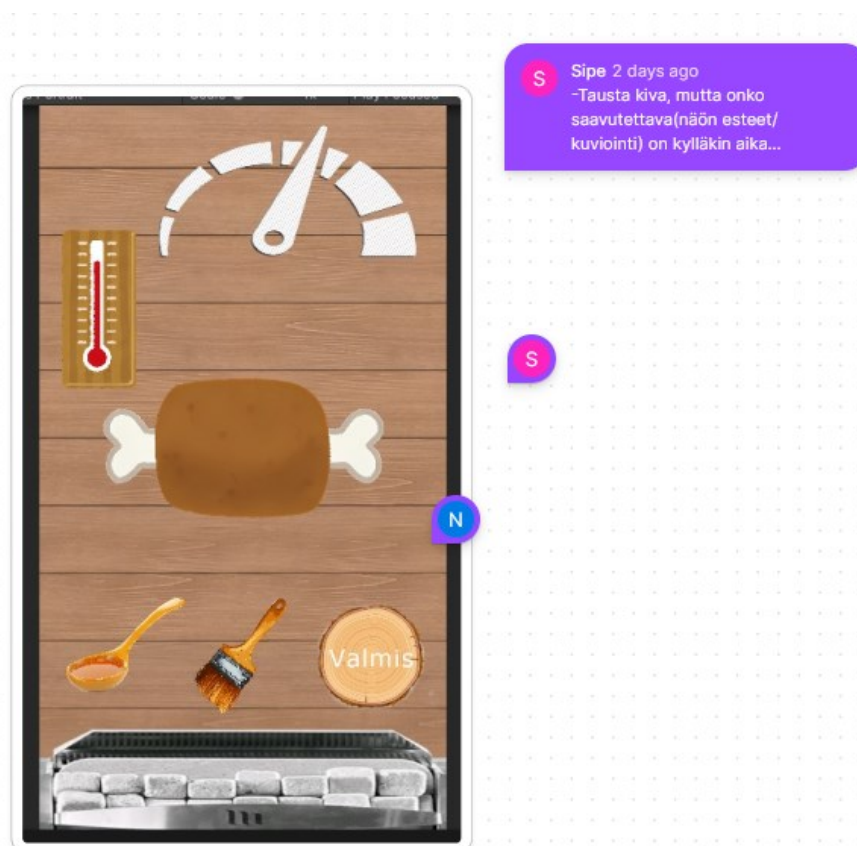
Figman prototyypin jälkeen oli aika siirtyä kehitystyössä pelidemon tekemiseen. Aloittelevina kokemattomina pelinkehittäjinä tutustuimme Unity Editor – ohjelmistoon hyödyntäen erilaisia materiaaleja, kuten Unity-alustan dokumentaatioon ja YouTube-tutoriaalien kautta. YouTube-tutoriaaleja hyödynnettiin C#-skriptauksen ja Unityn perusteiden oppimiseen ja videotutoriaalit tukivat hyvin Unity dokumentaation tulkintaa. Hyvänä lähteenä aloittelijoille toimi Simulation Lab – YouTubetilin tekemä Unity Crash Course for Absolute Beginners! Learn C# and Make Games!! – tutoriaali, jossa esitellään työvälineiden asennuksesta lähtien työvaiheet (Simulation Lab 2023).

Koska tavoitteenamme oli kehittää Android-mobiilipelistä ensimmäinen demo, aloitimme luomalla 2D- projektin ja valitsimme Android-alustan. Peliä kehitettiin yhdessä, joten käytimme versionhallintajärjestelmää. Tähän tarkoitukseen valikoitui GitHub.

5.5.1 Ensimmäinen versio

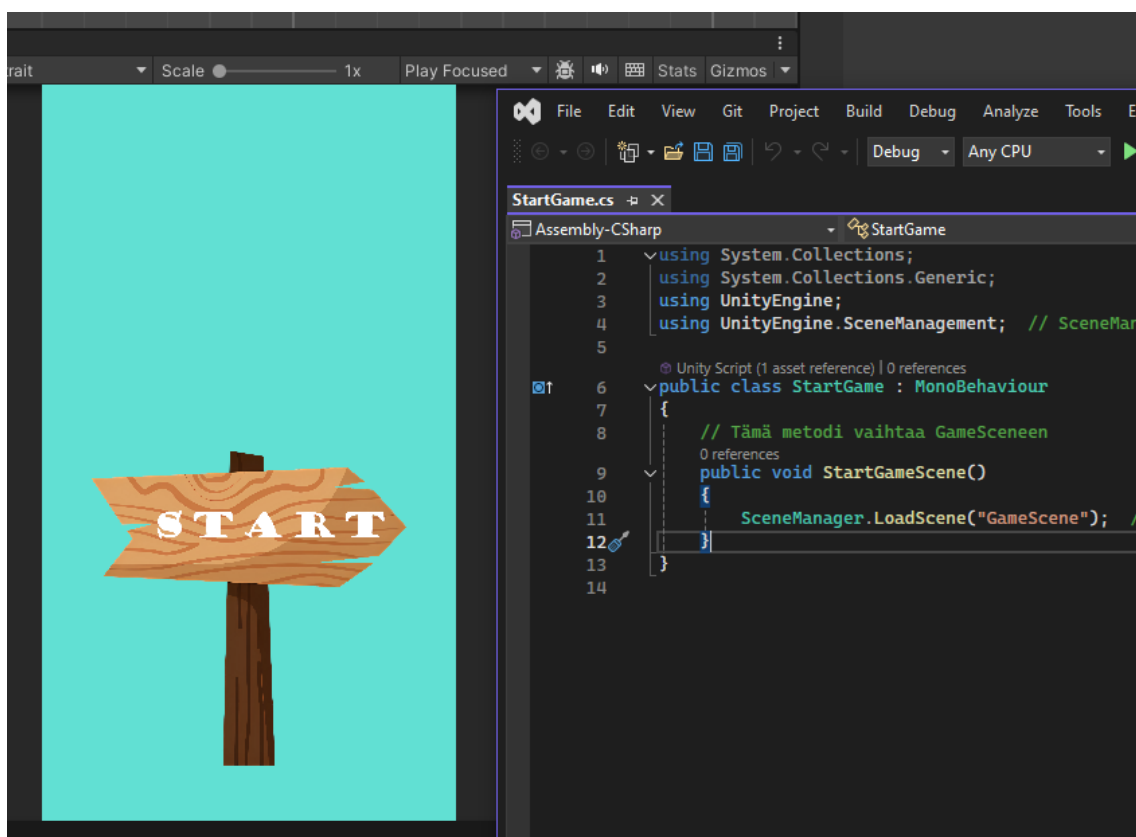
Ensimmäisenä etsimme erilaisia vaihtoehtokuvia demoon. Vaikka halusimme lopullisen pelin visuaalisen ilmeen muodostuvan itse tekemistämme materiaaleista, tämän opinnäytetyön demoversioihin hyödynsimme pääasiassa ilmaisia kuvapalveluita ja löysimme projektiin sopivia png-kuvia. Lähdimme kokeilemaan elementtien asettelua aiempien testituloksien ja analyysien perusteella.

Jätimme alareunassa olevan tilanneindikaattorin seuraavasta käyttöliittymän asettelusta kokonaan pois (kuva 20). Päädyimme tähän, koska tavoitteenamme oli aloittelijaystävällinen ja saavutettava käyttöliittymä. Kolmas tilanneindikaattori eli mittari olisi tästä näkökulmasta tehnyt pelin käyttöliittymästä liian sekavan. Kaikki kokeilut eivät siis päässeet jatkoon, yritimme miettiä valintoja käyttäjäkemuksen ja saavutettavuuden kautta menettämättä kuitenkaan visuaalista ilmettä. Ensimmäisessä demossa ollut tausta jäi myös pois sen huonon responsiivisuuden vuoksi. Myöskään painikkeet eivät erottuneet tarpeeksi taustakuvasta.



KUVA 20. Pelin käyttöliittymän hahmotelma ja Figma:n kommentointinäkymä

Jotta pelidemo olisi tuntunut jo saman tien peliltä, halusimme siihen ehdottomasti aloitusnäkyvän. Aloitusnäkyvässä oli painike, jossa oli kuva kyltistä start-tekstillä. Painike toteutettiin Image-elementtiä käyttäen, ja siihen lisättiin kylttikuva ja Button-komponentti. Image-elementti on objekti, niin kuin muutkin käyttöliittymä-elementit kuten kuvat ja tekstit (Quero 2023). Kaikki käyttöliittymäelementit ovat Canvas-peliobjektin lapsia ja Unity Editor lisää automaattisesti Canvas-peliobjektin näkymään, ellei sellaista ole käyttöliittymäelementtiä luodessa (Unity Technologies 2024a). Button-komponentin OnClick-tapahtumaan liitettiin script, joka toteutti siirtymän pelinäkömään (kuva 21).



KUVA 21. Pelin aloitusnäkyvä ja aloitusskripti

Button-komponentti mahdollistaa interaktiot, siirtymät ja navigoinnin On Click-tapahtuman kautta (Unity Technologies 2024b). Oleellinen osa pelien kehittämisessä Unity-alustalla ovat skriptit, joiden avulla voidaan esimerkiksi vastata pelaajan syötteisiin, hallita objektien fyysistä käyttäytymistä ja luoda graafisia efektejä (Unity Technologies 2024c).

Pelinäkymään lisättiin seuraavaksi ensimmäisiä interaktiivisia ominaisuuksia. Kauha toteutettiin samalla tavalla Image-elementtinä ja Button-komponentilla,

kuin aloitusnäkyvän kyltti. Kauhaan koskiessa tavoitteena oli saada höyryä nousemaan näytön alareunasta yläreunaan ja samalla lämpötila nousee.

Toiminnallisuus aloitettiin höyrystä. Höyry-assetti muutettiin prefabiksi ja sille lisättiin tunniste "hoyry", johon voitiin viitata toiminnallisuutta varten tehdyssä HoyryLiikkuu-scriptissä. Prefab toimii mallina, josta voidaan luoda uusia prefab-instansseja pelinäkyymiin (Unity Technologies 2024d). Höyry toteutettiin niin, että höyryn noustua näkyvän ulkopuolelle se tuhoutuu ja sen jälkeen luodaan uudelleen alareunaan. Luotiin HoyryController-peliobjekti, jolle lisättiin HoyryLiikkuu-script ja hoyry-prefab. Start()-metodissa luodaan höyry-instanssi ja määritellään sijainti näkyvän alareunaan (kuva 22).

```
Unity Message | 0 references
void Start()
{
    // Sijoitetaan höyry alkuun näytön ulkopuolelle
    hoyryInstance = Instantiate(hoyryPrefab, new Vector2(0, -8), Quaternion.identity);
    hoyryInstance.tag = "hoyrypilvi"; // Varmistetaan, että tagi on asetettu

    // Asetetaan kohdepositio näytön ulkopuolelle yläreunaan
    targetPosition = new Vector2(0, 8); // Riippuu näytön koosta, muokkaa tarvittaessa
}
```

KUVA 22. Höyryn start()-metodi

NostaHoyry() -metodia kutsutaan kauhan painalluksella ja metodi laittaa höyryn liikkeelle sekä lisää jokaisella kauhan painalluksella löylylaskuriin +1 (kuva 23). Update()-metodissa katsotaan pitääkö höyryn liikkua, liikutetaan tarvittaessa höyry ylös näkyvän ulkopuolelle, jonka jälkeen höyry tuhoetaan sekä liike lopetetaan (kuva 24). Update()-metodia kutsutaan jokaisen ruudunpäivityksen yhteydessä.

```
0
1 // Tätä metodia kutsutaan napin painalluksella
2 0 references
3 public void NostaHoyry()
4 {
5     if (hoyryInstance == null)
6     {
7         // Luodaan uusi höyry näytön ulkopuolelle alareunaan
8         hoyryInstance = Instantiate(hoyryPrefab, new Vector2(0, -8), Quaternion.identity);
9         hoyryInstance.tag = "hoyrypilvi"; // Asetetaan tagi
10    }
11    shouldMove = true; // Kun nappia painetaan, asetetaan liikkuminen päälle
12    löylyLaskuri++;
13
14 }
15
```

KUVA 23. NostaHoyry()-metodi

```
Unity Message | 0 references
void Update()
{
    // Jos höyryn pitää liikkua, siirretään sitä
    if (shouldMove && hoyryInstance != null)
    {
        // (smooth movement) kohti kohdepositiota
        hoyryInstance.transform.position = Vector2.MoveTowards(hoyryInstance.transform.position, targetPosition, Time.deltaTime * 6f);

        // Lopetetaan liike, kun kohde on saavutettu
        if ((Vector2)hoyryInstance.transform.position == targetPosition)
        {
            shouldMove = false;
            Destroy(hoyryInstance);
            hoyryInstance = null;
        }
    }
}
```

KUVA 24. Update()-metodi

Kinkun ja kastikesudin toiminnallisuutta kehiteltiin seuraavaksi. Kinkku-spriteille lisättiin Box Collider 2D -komponentti, jotta voitiin tunnistaa spriteen tulevat osumat. Box Collider 2D on törmäyskomponentti, jota voi käyttää törmäyspintana ja saadaan kiinni objektien väliset interaktiot (Unity Technologies 2024e). Kinkulle lisättiin myös tunnisteeksi ”kinkku”. Tunnisteella (engl. tag) voidaan viitata peliohjelmeihin skripteissä (Unity Technologies 2024f). Kastikesudin liikettä varten tehtiin SutiLiikkuu -script. Seuraavassa kuvassa (kuva 25) esitellään kastikesudin liikettä kinkun päällä.



KUVA 25. Kuva pelinäköymästä ja kastikesudin liikkeestä

Start() -metodissa tallennetaan sudin alkuperäinen sijainti. ResetPosition() -metodi palauttaa sudin alkuperäiseen sijaintiinsa (kuva 26).

```
// Tämä metodi palauttaa sudin alkuperäiseen sijaintiin
1 reference
private void ResetPosition()
{
    transform.position = originalPosition;
}
```

KUVA 26. ResetPosition() -metodi

Update() -metodi tarkistaa, pidelläänkö sudista kiinni ja aloitetaan drag -toiminto. Jos kosketus vapautetaan, kutsutaan ResetPosition() -metodia ja lopetetaan drag-toiminto. OnTriggerEnter2D() -metodi tunnistaa, kun suti on kinkkun päällä käyttäen kinkku-tunnistetta, samalla kastikelaskurin summaan lisätään +1 (kuva 27).

```
// Tämä metodi tunnistaa, kun suti on kinkkun päällä
Unity Message | 0 references
private void OnTriggerEnter2D(Collider2D other)
{
    // Tarkista osuuko suti kinkkuun
    if (other.gameObject.CompareTag("kinkku"))
    {
        // Lisää kastikelaskuriin, kun suti osuu kinkkuun
        kastikeLaskuri++;
        Debug.Log("Kastike lisätty! Nykyinen laskuri: " + kastikeLaskuri);
    }
}
```

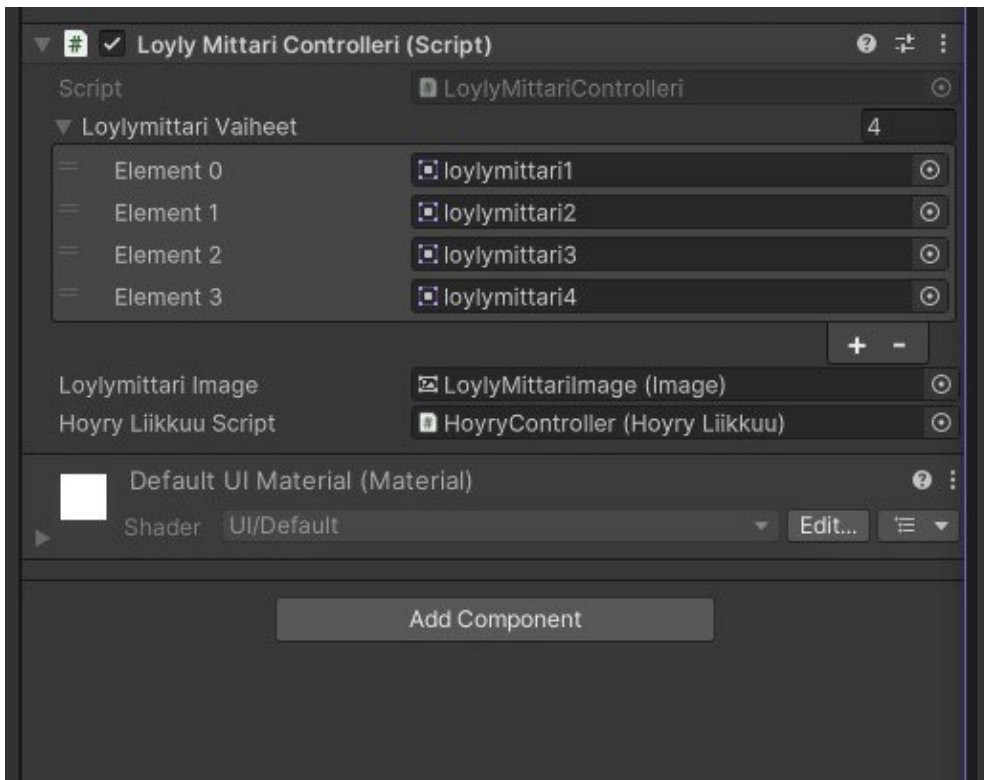
KUVA 27. OnTriggerEnter2D() -metodi

Suti-spriteille lisättiin Box Collider 2D -komponentti, jotta voitiin tunnistaa, milloin suti osuu kinkkuun. Lisäksi Is Trigger -ominaisuus ruksattiin (Unity Technologies 2024e). Tämä mahdollistaa osuman ilman törmäämistä. Sudille lisättiin myös Rigidbody 2D, jotta suti reagoi fyysisiin tapahtumiin kuten liikuttamiseen (Unity Technologies 2024g). Body Type -valikosta valittiin Kinematic-valinta, jotta suti ei liiku itsestään, mutta pelaaja voi liikuttaa sutia ja liikettä seurata koodissa (Unity Technologies 2024h). Script-komponenttiin lisättiin SutiLiikkuu -script.

Näiden toiminnallisuuksien myötä Unity Editor alkoi käymään tutummaksi ja koska kauha sekä suti toteutti jo jonkinlaista toiminnallisuutta ja liikettä, alkoi demo tuntua peliltä. Alustavat laskurit oli lisätty löylyn heitelle ja kastikkeeseen

lisäämiselle, mutta laskukaavaa ei tässä vaiheessa vielä ollut tarkemmin hahmoiteltuna ja visuaalisestikin näkymässä oli vasta yksi lämpömittari. Tähän demoversion jatkoimme tämän lämpömittarin toiminnallisuuden eteenpäin viemistä niin, että lämpömittarin kuva vaihtuisi löylylaskurin lukeman mukaisesti.

Löylymittari toteutettiin käyttäen Image-elementtiä. Löylymittarille tehtiin oma LoylyMittariControlleri -script, jossa käytettiin viitteen avulla HoyryLiikkuu -scriptissä olevaa loylylaskuri-muuttujaa. Scriptissä luotiin tyhjä taulukko mittarin kuville. Löylymittarille lisättiin script-komponenttiin LoylyMittariControlleri -script ja lisättiin taulukkoon lämpömittarin vaihekuvat (kuva 28). LoylyMittariControlleri -script sisältää VaihdaLoylymittarinKuva() -metodin, joka vaihtaa löylylaskurin mukaisesti kuvan pelinäkömään (kuva 29).

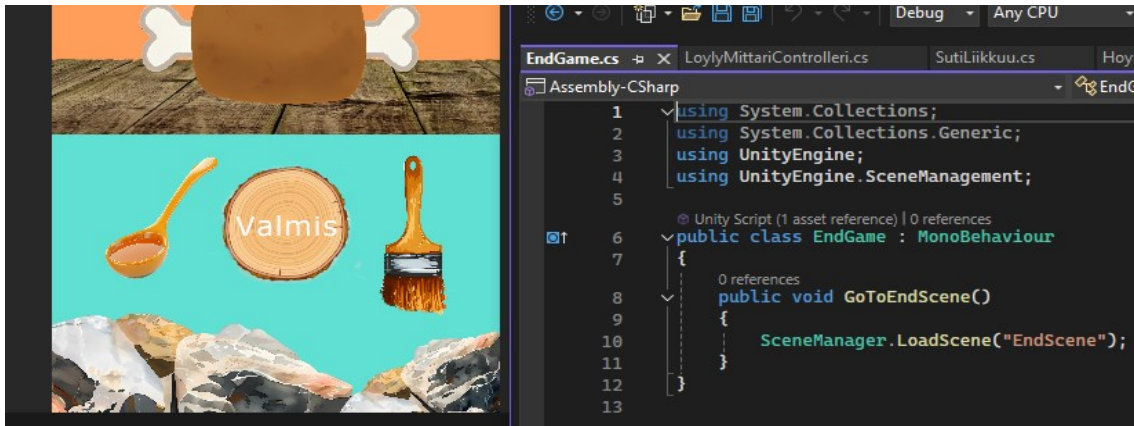


KUVA 28. Löylymittarikuvataulukko inspector-paneelissa (Unity 2024)

```
// Vaihda löylymittarin kuva laskurin mukaan
2 references
void VaihdaLoylymittarinKuva()
{
    int laskuri = hoyryLiikkuuScript.loylyLaskuri;
    loylymittariImage.sprite = loylymittariVaiheet[laskuri]; // Vaihdetään kuva
}
```

KUVA 29. VaihdaLoylymittariKuva()-metodi

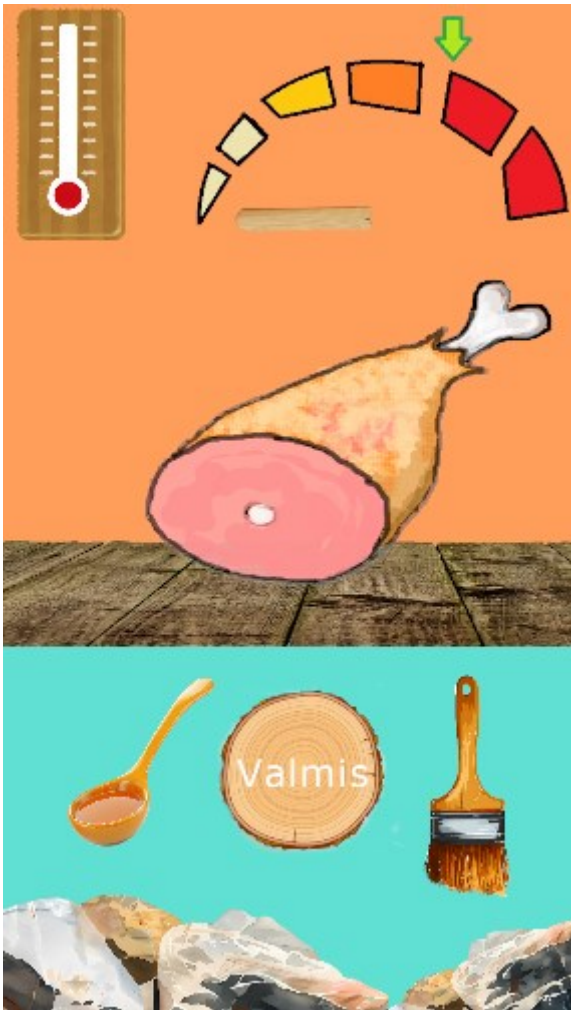
Pelissä on myös painike, jolla pelaaja voi valita milloin kinkku on sopivan mehukas. Tässä versiossa tämän painikkeen virkaa suoritti kuva Valmis-tekstillä (kuva 30). Lisäsimme tälle vielä siirtymän loppunäkymään. Tämän toiminnallisuus toteutettiin samalla tavoin kuin alussa toteutettu siirtymä pelin aloitukseen.



KUVA 30. Pelinäkö ja lopetusskripti

5.5.2 Toinen versio

Pelin ensimmäistä demoa lähdettiin kehittämään pidemmälle, vaikka tässä vaiheessa oli selvää, ettei aikataulu välttämättä sallisi täysin valmista tuotetta. Ensimmäisen demon kinkun lähdekuva vaihdettiin omaan tuotokseen (kuva 31), jotta tiimin oma mielikuvitus näkyisi mahdollisimman paljon myös demossa.



KUVA 31. Unity-projektin UI-näkymä

Suunnitelmassa pelin käyttöliittymässä oli yhteensä kolme mittaria, joiden nousua pelaajan kuuluu seurata. Päämittarina näistä toimii mittari, joka seuraa kinkun lämpötilaa. Kinkun lämpötilaa seuraava mittari toteutettiin teknisesti lähes samalla kaavalla kuin saunan lämpötilamittari, eli koodin avulla luotiin Inspector-paneeliin taulukko kaikille sprite-kuville, jotka vaihtuvat pelaajan heittäessä lisää löylyä. Kuvien vaihtumista kontrolloi UpdateKinkkuMittari()-metodi (kuva 32). Muutokset tehtiin samaan koodipohjaan kuin löylyn mittarin liikkuminen. Koodi vaati lisäksi laskukaavan, jotta kinkun lämpötila ei nouse samassa suhteessa saunan lämpötilan kanssa.

```

// Update the kinkku meter image
void UpdateKinkkuMittari()
{
    kinkkuMittariImage.sprite = kinkkuMittariVaiheet[kinkkuCounter];
}

```

KUVA 32. UpdateKinkkuMittari() -metodi

Toiseen demoon haluttiin audiota visuaalisen höyrypilvianimaation lisäksi. Tätä varten peliobjekteihin on mahdollista lisätä äänilähde (eng. Audio Source) -komponentteja, jotka tuottavat peliin ääntä, ja jotka on mahdollista soittaa välittömästi peli-ikkunan avautuessa, tai kiinnittää toimimaan skriptien ehtojen mukaisesti. Tässä tapauksessa äänilähteiden ei haluttu kuuluvan heti, joten komponentista poistettiin, Play on Awake -ominaisuus (kuva 33). Äänilähde-komponentti lisättiin tyhjäan peliobjektiin, joka kontrolloi höyrypilvianimaatiota ja aiheuttaa samalla äänen, joka kuulostaa veden kuumalle kiukaalle heittämiseltä. Kinkun suti-peliobjektiin lisättiin myös oma äänilähde, joka aktivoituu, kun pelaaja vetää sutia ja se osuu kinkun collider-komponenttiin.



KUVA 33. HoyryAani -audiokomponentti (Unity 2024)

Äänilähde-komponentteja testatessa huomattiin muun muassa, että ääniraidan kesto oli toiminnan toistuvuuteen nähden turhan pitkä ja sitä päädyttiin lyhentämään. Lisäksi löylyjen heittojen tiheyden vuoksi, äänenvoimakkuus alkoi kuulostamaan liian kovalta ja terävältä, joten näitä säädettiin myös käyttömukavuuden parantamiseksi (kuva 34).



KUVA 34. Audio Source -komponentti (Unity 2024)

Kun saunan ja kinkun lämpömittarit oli saatu toimimaan, haluttiin peliin vielä lisätä pistelaskuri, jotta peli olisi mahdollista pelata läpi. Pistelaskuri, joka toteutettiin CheckScoreAndEndGame()-metodilla (kuva 35), seuraa kinkun lämpömittarin kuvataulukkoa ja antaa pelaajalle sitä suuremman pistemäärän mitä lähempänä pelaaja on kuvaa, jossa lämpötila osoittaa vihreää tavoitelämpötila-merkintää, joka näkyy kuvassa 31.

```
// Check if mittari10 is active, calculate partial score based on closeness, and end the game
void CheckScoreAndEndGame()
{
    // Get the current index of the sauna meter image
    int currentIndex = loylyCounter; // This represents the current "heat" level
    int maxIndex = loylymittariVaiheet.Length - 4; // Index of mittari10

    // Calculate score based on closeness to maxIndex
    float closeness = (float)currentIndex / maxIndex; // Percentage of closeness
    score = Mathf.RoundToInt(maxScore * closeness); // Calculate partial score

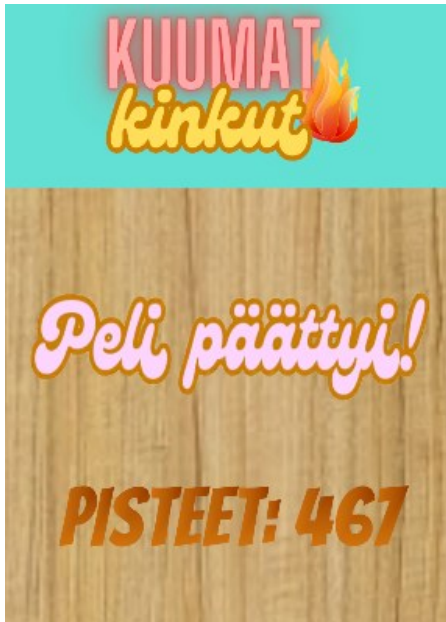
    Debug.Log("Final Score: " + score);

    // Save the score using PlayerPrefs
    PlayerPrefs.SetInt("FinalScore", score);

    // Call GoToEndScene method from the EndGame script attached to Valmisbutton
    valmisButton.GetComponent<EndGame>().GoToEndScene();
}
```

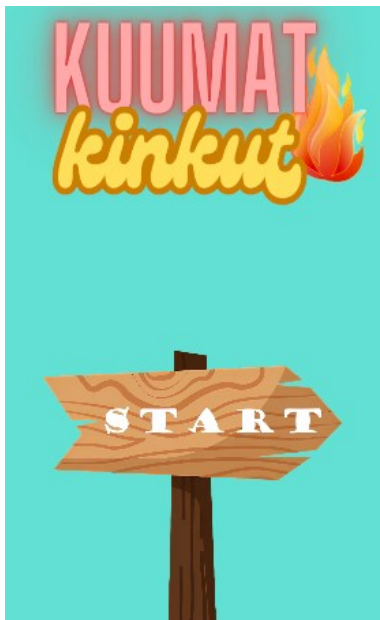
KUVA 35. CheckScoreAndEndGame()-metodi

Kun pelaaja painaa pelin käyttöliittymässä näkyvää Valmis-painiketta, lopetus-taso ilmoittaa pelaajalle kokonaispistemäärän (kuva 36).



KUVA 36. Pelaajan pistemäärä

Lopuksi keskityttiin audiovisuaalisten elementtien lisäämiseen, jotta yksinkertaisesta ja vielä vajaasta pelistä voisi saada mahdollisimman miellyttävän näköisen ja kuulaisen. Aloitus- ja lopetusnäkyään lisättiin rauhallinen taustamusiikki, jossa hyödynnettiin rojaltivapaita ääniraitoja. Pelin ilmettä paranneltiin käyttämällä Canva-verkkosivuston graafisen suunnittelun työvälinettä, jolla saatiin tehtyä pelille otsikko (kuva 37) ja lopetusnäkyään "peli päättyi!"-ilmoitus.



KUVA 37. Aloitusnäky

6 YHTEENVETO

6.1 Tekniset haasteet

Tämän opinnäytetyön toteutuksessa isoimmat haasteet liittyivät erityisesti pelin tekniseen toteutukseen Unity-pelimoottorilla. Ryhmä keräsi projektin aikana laajasti tietoa käyttäjäkokemuksen suunnittelusta, sekä hyödynsi jo opittua tietoa. Pelimoottorien käytöstä ryhmällä ei ollut aikaisempaa kokemusta, joten haasteellisin osuus tiedettiin jo alussa olevan pelin demon toteuttaminen.

Unity-pelimoottorin oppiminen oli pintapuolista ja haasteina oli muun muassa koodilogiikan käsittäminen ja suunnittelemisen peliin ja niiden toteuttaminen vaati paljon soveltamista muiden pelisuunnittelijoiden projektien ja tutoriaalien avulla.

Itse Unity-projektin alkuvaiheissa ensimmäiseksi isommaksi haasteeksi muodostui "Adaptive Performance"-työkalun virhekoodi, joka viittasi työkaluun, joka mobiilipelien suunnittelussa on tehty optimoimaan suorituskykyä mobiililaitteilla, jotta pelit toimivat hyvin laitteilla, joissa on rajoitetusti resursseja muun muassa akun keston, lämpötilan ja suorituskyvyn suhteen.

Pelin testaus antoi virheen "Adaptive Performance Provider"-konfiguraatiosta, vaikka kyseiset tarjoajat oli projektiin asennettu. Virhe toistui, vaikka työkalua ja tarjoajia asennettiin uudelleen useampaan otteeseen. Tässä vaiheessa Unity-hub-sovellukseen oli asennettu editor versiolla 2022.3.41f1 ja selvisikin Unity-keskustelupalstan kautta, että editor-versiot, jotka ovat vanhempia kuin 2023.1, eivät ole tuettuja Android Provider-ominaisuuden kautta, niiden käyttämien toimintojen takia. Keskustelupalstalla Unity-tukihenkilö kehoitti asentamaan Adaptive Performance-työkalun versiolla 5.0.1 ja niihin provider-ominaisuudet versioilla 1.1.2 (Android Provider) ja 5.0.0 (Samsung Provider) lisäämällä vanhat versiot projektin kansion tiedostoon (ProjectDir/Packages/manifest.json) dependencies-osioon (kuva 38). (David-Berger, 2023.)

```
"com.unity.modules.xr": "1.0.0"|  
  "com.unity.adaptiveperformance": "5.0.1",  
  "com.unity.adaptiveperformance.google.android": "1.1.2",  
  "com.unity.adaptiveperformance.samsung.android": "5.0.0"  
}
```

KUVA 38. Kuumat Kinkut-projektin manifest.json-tiedosto

Mikäli oikeat versiot eivät asentuneet, vaihtoehtoisiksi jäi ottaa Adaptive Performance -työkalu pois käytöstä kokonaan ja poistaa Package Manager -osiosta Mobile-ominaisuus, jolloin virhettä ei enää tullut. Tämän opinnäytetyön tapauksessa päädyttiin juuri tähän ratkaisuun, koska kyseessä oli pelkästään pelin demo. Jatkokehitystä ja pelin julkaisua ajatellen olisi kuitenkin tärkeää toteuttaa peli valmiiksi uudemmilla editor-versioilla, jotka tukevat kaikkia Android Provider -ominaisuuden toimintoja.

6.2 Jatkokehitysideat

Jatkokehitystä ja käyttäjälähtöisyyttä ajatellen pelin aloitusnäytölle voisi lisätä kysymysmerkki-ikonin, jota klikatessa avautuu ohjesivu. Kysymysmerkki-ikonin voisi sijoittaa vasempaan yläkolmannekseen, valkoisella pohjavärillä, jotta sen näkyvyys olisi hyvä ja sen käytettävyys olisi helppoa. Käyttöliittymä tulisi olla muokattavissa esimerkiksi yhdellä kädellä pelattavaksi tai eleillä ohjattavaksi. Ohjesivulla voisi olla yksinkertaiset ja selkeät ohjeet pelin pelaamiselle. Tämä sisältäisi toiminnallisuuksien kuvauksen, eli miten painikkeikoneita käytetään ja miten mittarit toimivat.

Saavutettavuuden näkökulmasta peliobjektien ja tekstien kontrastissa ja väreissä on vielä paranneltavaa, koska erityisesti pohja, joka sisältää tekstuuria, saattaa tehdä tekstistä hankalasti luettavan.

Viihdyttävyyteen vaikuttavien tekijöiden tutkimiseen tekemämme kyselyn perusteella voisimme viedä eteenpäin pelimme tarinallisuutta ja miettiä sosiaalisen vuorovaikutuksen mahdollisuuksien lisäämistä, esimerkiksi moninpelimahdollisuudella. Ideoita erilaisiin tasoihin ja lisävalintoihin on kehitystyön aikana syntynyt, ja nämä myös osaltaan lisäisivätkin pelin vetovoimaa. Erilaiset mahdollisuudet mukauttaa pelin ulkoasua ja pelihahmoa auttaisi pelaajia luomaan syvemmän

yhteyden peliin ja näin myös sitoutumaan siihen. Esimerkiksi kinkulle tuli useita erilaisia vaihtoehtoja ideointivaiheessa, muun muassa joulukinkku, juhannuskinkku tai mahdollisuus valita vaikkapa harmaasuolattu kinkku.

Pisteiden kertymiseen vaikuttavaa logiikkaa on suunnitelmissa parantaa, niin että siitä tulisi monimutkaisempi. Tällä hetkellä pelin pistesaldoon ei vaikuta suti-peliobjekti, vaikka alkuperäisessä suunnitelmassa tämän ominaisuuden oli tarkoitus kerryttää pelaajalle lisäpisteitä. Hyvin toteutettu palkitseminen lisäisi pelikokemuksen positiivisuutta. Tällaisen voisi toteuttaa esimerkiksi tasoittain etenevillä tavoitteilla, jolloin pelaaja kokisi jatkuvaa edistymistä.

Pientä mielipidekyselyä ja käytettävyydestä tehtiin kehitystyön edetessä. Demon kehitys jäi kuitenkin vielä alkuvaiheeseen, joten täysin pelattavaa versiota ei ole vielä testaukseen mahdollista tarjota. Tällaisen pelattavan version kohdalla voisimme tulevaisuudessa suunnitella ja toteuttaa käyttäjätestauksen, kerätä käyttäjäpalautetta ja soveltaa saatua palautetta käyttäjäkokemuksen parantamiseen.

7 LOPPUPOHDINTA

Projektin aikana onnistuttiin tutkimaan pelin suunnittelua käyttäjäkokemuksen näkökulmasta ja kehittämään pelisuunnitelma, joka noudattaa näitä periaatteita. Toiminnallisessa osiossa saatiin myös aikaan Figman prototyyppien kautta työtettyä lähes tavoitteiden mukainen pelidemo Unity-pelimootorilla. Pelidemon jatkokehitysideat kehittyivät käyttäjäkokemuksen, käytettävyyden ja saavutettavuuden parantamisen kautta. Projektiryhmä sai syventävää tietoa käyttäjälähtöisestä suunnittelusta, sitä ohjaavista säädöksistä ja laeista ja oppi hyödyntämään saamaansa tietoa pelisuunnittelun eri vaiheissa.

Kolmen henkilön raportin kirjoittamisessa oli hyvät ja huonot puolensa. Koska tekijöitä oli useita, lähdettiin käyttäjäkokemusta tutkimaan laajasti käyttöliittymän suunnittelusta alkaen, käytettävyyden ja saavutettavuuden kautta. Pelintekijöinä olimme myös noviiseja, joten otimme teorian kattavasti mukaan pelin kehitysprosessiin, suunnitteluvaiheesta toteutukseen. Hyödynsimme hankittua tietoperustaa pelin suunnitteluprosessin toiminnallisessa osiossa. Raportin jäsentely ja yhdessä kirjoittaminen oli osin haasteellista. Sisällön järjesteleminen lopulliseen muotoonsa vei paljon aikaa ja viimeistely oli työlästä. Vaivannäkö kuitenkin kannatti ja lopputuloksena saimme mielestämme aikaan kattavan ja mielenkiintoisen opinnäytetyön käyttäjäkokemuksen huomioimisesta mobiilipelien suunnittelussa.

LÄHTEET

Ablegamers 2024. Assistive technology for mobile gaming: tools and apps that make a difference. Luettavissa: <https://ablegamers.org/assistive-technology-for-mobile-gaming-tools-and-apps-that-make-a-difference/> . Luettu: 9.10.2024.

Aluehallintovirasto 2024a. Yleistä saavutettavuudesta. Luettavissa: <https://saavutettavuusvaatimukset.fi/yleista-saavutettavuudesta/> . Luettu: 12.10.2024.

Aluehallintovirasto 2024b. Yleistä saavutettavuudesta. Luettavissa: <https://saavutettavuusvaatimukset.fi/yleista-saavutettavuudesta/> . Luettu: 12.10.2024.

Ceci L. 2024a. App uninstall rate in selected markets worldwide 2023. Luettavissa: <https://www.statista.com/statistics/1278067/app-uninstall-rate-global/> . Luettu: 27.9.2024.

Ceci L. 2024b. App uninstall rate in selected markets worldwide 2023. Luettavissa: <https://www.statista.com/statistics/1278067/app-uninstall-rate-global/> . Luettu: 27.9.2024.

Curry, D. 2023. Candy Crush Revenue and Usage Statistics (2024). Luettavissa: <https://www.businessofapps.com/data/candy-crush-statistics/> . Luettu: 27.9.2024.

David-Berger 2023. Error when Installing Android Adaptive Performance Package. Luettavissa: <https://discussions.unity.com/t/error-when-installing-android-adaptive-performance-package/922277/2> . Luettu: 3.10.2024.

Drake, Jeff 2023. 24 Great Games That Use The Unity Game Engine. Luettavissa: <https://www.thegamer.com/unity-game-engine-great-games/> . Luettu: 27.9.2024.

Figma Learn 2024. What is Figma?. Luettavissa: <https://help.figma.com/hc/en-us/articles/14563969806359-What-is-Figma> . Luettu: 12.10.2024.

Greenan, R. 2023. What is Game UX? The Complete Guide for 2024. Luettavissa: <https://careerfoundry.com/en/blog/ux-design/game-ux/> . Luettu: 10.10.2024.

Game accessibility guidelines 2016a. Luettavissa: <https://gameaccessibilityguidelines.com/if-producing-a-pc-game-support-windowed-mode-for-compatibility-with-overlaid-virtual-keyboards/> . Luettu: 12.10.2024.

Game accessibility guidelines 2016b. Luettavissa: <https://gameaccessibilityguidelines.com/why-and-how/> . Luettu: 12.10.2024.

Game accessibility guidelines 2016c. Luettavissa: <https://gameaccessibilityguidelines.com/why-and-how/> . Luettu: 12.10.2024.

Game accessibility guidelines 2016d. Luettavissa: <https://gameaccessibilityguidelines.com/why-and-how/> . Luettu: 12.10.2024.

Hammal, F 2024. Ad Revenue in mobile gaming – Earning tips for marketers. Luettavissa: <https://www.tekrevol.com/blogs/ad-revenue-in-mobile-gaming-earning-tips-for-marketers/> . Luettu: 30.9.2024.

Hashemi-Pour, Cameron & Churchville, Fred 2024a. user interface (UI). Tech-Target. Luettavissa: <https://www.techtarget.com/searchapparchitecture/definition/user-interface-UI> . Luettu: 15.9.2024.

Hashemi-Pour, Cameron & Churchville, Fred 2024b. user interface (UI). Tech-Target. Luettavissa: <https://www.techtarget.com/searchapparchitecture/definition/user-interface-UI> . Luettu: 15.9.2024.

Heinon, M s.a.. Tietokonepelien Historia. Suomen tietokone museo. Luettavissa: <https://suomentietokonemuseo.fi/portfolio-item/tietokonepelien-historia/> . Luettu: 4.9.2024.

Holopainen, P. 2023a. Työkaluja verkkopalveluiden saavutettavuuden testaamiseen. Tietotalo. Luettavissa: <https://www.tietotalo.fi/fi/Ajankohtaista/Tyokaluja-verkkopalveluiden-saavutettavuuden-testaamiseen> . Luettu: 8.10.2024.

Holopainen, P. 2023b. Työkaluja verkkopalveluiden saavutettavuuden testaamiseen. Tietotalo. Luettavissa: <https://www.tietotalo.fi/fi/Ajankohtaista/Tyokaluja-verkkopalveluiden-saavutettavuuden-testaamiseen> . Luettu: 8.10.2024.

Koster, Raph 2014. A Theory of Fun Game Design. Second Edition. O'Reilly Media. E-kirja. Luettu: 28.8.2024.

Kruchkov, Aleksandr 2024. Winning gamers' hearts: 5 must-have mobile game ui features. Kreonit. Luettavissa: <https://kreonit.com/testing-and-qa/mobile-game-ui/> . Luettu: 15.9.2024.

Loijas, J., Mäkeläinen, J. & Pippola, T. 2023. Digitaalisen hyötypelin prototyyppi. eSignals PRO. Luettavissa: <https://esignals.fi/pro/2023/02/06/digitaalisen-hyoty-pelin-prototyyppi/#963c915a> . Luettu: 17.9.2024.

Long, N. 2023. 2022's top grossing mobile games: Honot of Kings, PUBG Mobile, Genshin Impact and more. Luettavissa: <https://mobilegamer.biz/2022s-top-grossing-mobile-games-honor-of-kings-pubg-mobile-genshin-impact-and-more/> . Luettu: 26.9.2024.

Menard, Michelle 2012. Game Development with Unity. Course technology PTR. Boston.

Microsoft. 10.5.2024. Power Platform. Luettavissa: <https://learn.microsoft.com/fi-fi/power-platform/well-architected/experience-optimization/user-interface-content> . Luettu 4.10.2024.

Navigating the World of Mobile Game Development: Challenges and Opportunities 2024. 30 Days Coding. Luettavissa: <https://30dayscoding.com/blog/mobile-game-development-challenges-and-opportunities> . Luettu: 27.9.2024.

NIST s.a. Usability. Luettavissa: <https://csrc.nist.gov/glossary/term/usability> . Luettu: 13.9.2024.

Nyqvist, P. 20.4.2023a. Saavutettavuuden testaamiseen on useita tapoja. Tietoenvy. Luettavissa: <https://www.tietoenvy.com/fi/blogi/2023/04/saavutettavuuden-testaamiseen-on-useita-tapoja/> . Luettu: 8.10.2024.

Nyqvist, P. 20.4.2023b. Saavutettavuuden testaamiseen on useita tapoja. Tietoevry. Luettavissa: <https://www.tietoevry.com/fi/blogi/2023/04/saavutettavuuden-testaamiseen-on-useita-tapoja/> . Luettu: 8.10.2024.

Nyqvist, P. 20.4.2023c. Saavutettavuuden testaamiseen on useita tapoja. Tietoevry. Luettavissa: <https://www.tietoevry.com/fi/blogi/2023/04/saavutettavuuden-testaamiseen-on-useita-tapoja/> . Luettu: 8.10.2024.

Näkövammaisten liitto. 10.10.2023. Miten näkövammaisen käyttää tietokonetta tai mobiililaitetta? . Luettavissa: <https://www.nakovammaistenliitto.fi/fi/miten-na-kovammaisen-kayttaa-tietokonetta-tai-mobiililaitetta> . Luettu: 8.10.2024.

Perustuslaki 731/1999. Luettavissa: <https://www.finlex.fi/fi/laki/ajantasa/1999/19990731#L2P6> . Luettu 12.10.2024.

Promark Business Solutions 2021. Tips for Successful Mobile Game prototyping. Luettavissa: <https://promarksolutions.ca/tips-for-successful-mobile-game-prototyping/> . Luettu: 15.9.2024.

Quero, D 2023. The Ultimate Guide to Unity UI Image. Medium. Luettavissa: <https://medium.com/geekculture/the-ultime-guide-to-unity-ui-image-9d5afc28c2de> . Luettu: 13.10.2024.

Saavutettavasti.fi 2023a. Verkkosisältöjen saavutettavuus. WCAG. Luettavissa: <https://www.saavutettavasti.fi/verkkosisaltojen-saavutettavuus/wcag/> . Luettu: 9.10.2024.

Saavutettavasti.fi 2023b. Verkkosisältöjen saavutettavuus. WCAG. Luettavissa: <https://www.saavutettavasti.fi/verkkosisaltojen-savutettavuus/wcag/> . Luettu: 9.10.2024.

Simulation Lab 2023. Unity Crash Course for Absolute Beginners! Learn C# and Make Games!!. Video. Katsottavissa: <https://www.youtube.com/watch?v=5DF55w6b4b4> . Katsottu: 13.10.2024.

Sinkkonen, I., Kuoppala, H., Parkkinen, J. & Vastamäki, R. 2006. Käytettävyyden psykologia. 3. uudistettu painos. Edita Publishing Oy. Helsinki.

Spence, A 2020. Does Candy Crush have a good UX? Luettavissa: <https://uxdesign.cc/does-candy-crush-have-a-good-ux-3c1a865d24e> . Luettu: 26.7.2024.

Suleiman, Abdul 2024a. What Is Usability And Why Does it Matter In App Development?. UX4Sight. Luettavissa: <https://ux4sight.com/blog/what-is-usability-and-why-does-it-matter-in-app-development> . Luettu: 13.9.2024.

Suleiman, Abdul 2024b. What Is Usability And Why Does it Matter In App Development?. UX4Sight. Luettavissa: <https://ux4sight.com/blog/what-is-usability-and-why-does-it-matter-in-app-development> . Luettu: 13.9.2024.

Toftedahl, Marcus 2019. Which are the most commonly used Game Engines? Luettavissa: <https://www.gamedeveloper.com/production/which-are-the-most-commonly-used-game-engines-> . Luettu: 27.9.2024.

Toth, K. 2016. Building Community Thanks To A Little Game Called Pokémon Go. Luettavissa: <https://www.wshu.org/culture/2016-07-15/building-community-thanks-to-a-little-game-called-pokemon-go> . Luettu: 8.10.2024.

Tyson, J. 2023. The Role of UX/UI Design in Mobile Game App Development. Luettavissa: <https://medium.com/@jimmytyson687/the-role-of-ux-ui-design-in-mobile-game-app-development-fa004e051048> . Luettu: 27.9.2024.

Unity Technologies 2024a. Canvas. Luettavissa: <https://docs.unity3d.com/Packages/com.unity.ugui@2.0/manual/UICanvas.html> . Luettu: 13.10.2024.

Unity Technologies 2024b. Button. Luettavissa: <https://docs.unity3d.com/Packages/com.unity.ugui@2.0/manual/script-Button.html> . Luettu: 13.10.2024.

Unity technologies 2024c. Scripting. Luettavissa: <https://docs.unity3d.com/Manual/ScriptingSection.html> . Luettu: 13.10.2024.

Unity Technologies 2024d. Prefab. Luettavissa: <https://docs.unity3d.com/Manual/Prefabs.html> . Luettu: 13.10.2024.

Unity Technologies 2024e. Box Collider 2D component reference. Luettavissa: <https://docs.unity3d.com/Manual/2d-physics/collider/box-collider-2d-reference.html> . Luettu: 13.10.2024.

Unity Technologies 2024f. Tags. Luettavissa: <https://docs.unity3d.com/Manual/Tags.html> . Luettu: 13.10.2024.

Unity Technologies 2024g. Rigidbody 2D. Luettavissa: <https://docs.unity3d.com/Manual/rigidbody2D.html> . Luettu: 13.10.2024.

Unity Technologies 2024h. Rigidbody 2D body types. Luettavissa: <https://docs.unity3d.com/Manual/rigidbody2D-body-types.html> . Luettu: 13.10.2024.

W3C Markup Validation Service 2024. Luettavissa: https://validator.w3.org/#validate_by_uri+with_options . Luettu: 8.10.2024.

Wikipedia 2024. Käyttöliittymä. Luettavissa: <https://fi.wikipedia.org/wiki/Käyttöliittymä> . Luettu: 12.10.2024.