



Toni Kokkonen

Mainosjärjestelmien käyttöönotto peleissä

Metropolia Ammattikorkeakoulu

Insinööri (AMK)

Tieto- ja viestintäteknikan tutkinto-ohjelma

Insinöörityö

9.5.2024

Tiivistelmä

Tekijä: Toni Kokkonen
Otsikko: Mainosjärjestelmien käyttöönotto peleissä
Sivumäärä: 27 sivua + 2 liitettä
Aika: 9.5.2024

Tutkinto: Insinööri (AMK)
Tutkinto-ohjelma: Tieto- ja viestintätekniikka
Ammatillinen pääaine: Pelisovellukset
Ohjaaja: Heini Puuska, Lehtori

Opinnäytetyön tarkoituksena oli selvittää, kuinka mainosjärjestelmä käyttöönotetaan koodipohjaisesti. Järjestelmäksi otettiin Unity Ads -mainosalusta, jonka käyttöönotosta tehtiin käytännöntyö. Työtä dokumentoitiin päivittäin työpäiväkirjaan, jonka avulla ohjeistuksen kehitys tuli helpommaksi.

Työssä esitellään mainostyyppejä mainostukseen ja ohjeistusta hyviin käyttötilanteisiin. Työssä on tutkittu monetisaation tärkeyttä, tapoja kehittää sitä ja kuinka sitä voi käyttää tehokkaasti peleissä. Unity Ads -mainosalustan ja muiden mainosalustojen tuotto-odotuksista kerättiin ja kirjattiin dataa. Opinnäytetyö sisältää tietoa mainosten markkinan toiminnasta, niiden muuttujista ja tietoa siitä, kuinka mainostajat ostavat mainoksia.

Opinnäytetyössä on myös tarkat ohjeet mainosten lisäämiseen Unity Ads -alustalle. Ohjeistus käy läpi vaiheet käyttäjän luomisesta mainosten ohjelmoimiseen ja sisältää suomennetut liitteet Unity Ads SDK API-rajapinnan ohjelmakoodin viiteluetteloon.

Opinnäytetyön tietoja, tuloksia ja ohjeistusta on mahdollista käyttää Unity Ads -alustan käyttöönottoon peleihin sekä saada myös tietämystä mainostamisesta kokonaisuutena.

Avainsanat: mobiili, mobiilipeli, mainos, mainokset, käyttöönotto, monetisaatio, Unity, mainostyyppit, markkina

Tämän opinnäytetyön alkuperä on tarkastettu Turnitin Originality Check -ohjelmalla.

Abstract

Author: Toni Kokkonen
Title: The implementation of advertisement systems in games
Number of Pages: 27 pages + 2 appendices
Date: 9 May 2024

Degree: Bachelor of Engineering
Degree Programme: Information and Communications Technology
Professional Major: Game Applications
Supervisors: Heini Puuska, Senior Lecturer

The point of this thesis was to figure out, how code-based advertisement systems are implemented and deployed. The selected advertisement platform was Unity Ads and its deployment was made into a practical work. Progress was documented daily in a work diary, which made creating instructions for deployment easier.

This thesis introduces many ad formats for advertisement and instructions on practical use cases. In this thesis monetization importance was studied, ways it can be developed and how it can be used efficiently in games. The estimated revenue data in the Unity Ads -platform and other platforms was collected and documented. The thesis also goes in to detail on how the market for ads work, what variables affect it and how advertisers buy ad space.

The thesis also contains precise instructions on how to add ads with the Unity Ads -platform. The instructions go through every step, from creating the user to programming the ads and contains translated attachments to Finnish for the Unity Ads SDK API scripting reference.

This thesis' information, results and instructions can be used to deploy Unity Ads to a game and can be used to gain knowledge on advertisement as a whole.

Keywords: mobile, mobile game, ad, ads, initialization, monetization, Unity, ad formats, market

Sisällys

Lyhenteet

1	Johdanto	1
2	Monetisaatio	1
2.1	Pelinsisäinen valuutta ja taloudellisuus	2
2.2	Palkkiomainosten käyttö	4
3	Mainostyypit	4
3.1	Bannerimainokset	4
3.2	Väli-ikkunamainokset	5
3.3	Palkkiomainokset	6
3.4	Muut mainostyypit	7
4	Tuotto-odotukset	8
4.1	CPM ja eCPM	8
4.2	Muut mainosalustat	9
4.3	Alustojen tuotto	9
5	Mainosten markkinat	10
5.1	Mainosten hinnan määrittäminen	10
5.2	Mainospaikan ostot	12
6	Mainosten lisääminen Unityssä	14
6.1	Projektin liitos palveluun	15
6.2	Mainosten alustus	16
6.3	Mainosten käyttö	17
7	Yhteenveto	23
	Lähteet	25

Liite 1: Suomennettu Unity SDK API-rajapinnan ohjelmakoodin viiteluettelo

Liite 2: Mainostamisen maiden tasoluokitukset (suomennettu)

Lyhenteet

SDK: *Software development kit*. Ohjelmistokehitykseen luotu työkalusarja.

API: *Application programming interface*. Ohjelmointirajapinta on komponenttien ja moduulien välinen raja ohjelmoitavassa järjestelmässä.

CPM: *Cost per mille*. Mittayksikkö, jolla lasketaan mainostamisessa paljonko mainostajat maksavat julkaisijalle tuhatta näyttökertaa kohden.

eCPM: *Effective cost per mille*. Mittayksikkö, jolla lasketaan mainostamisessa paljonko julkaisija sai tuottoa tuhatta näyttökertaa kohden.

CPC: *Cost-per-click*. Mittayksikkö, jolla määritellään keskimääräinen summa, joka velotetaan mainostajalta mainoksen napsautuksesta.

1 Johdanto

Mainoksia näkee melkein joka kerta puhelinpelejä pelatessa ja sovelluksia sela-
tessa. Mainokset toimivat hyvin dynaamisesti ja melkein saumattomasti käytet-
tynä oikein. Tämän vuoksi tuli selvittää, miten mainokset toimivat, mitä vaihto-
ehtoja niihin on, kuinka paljon niistä saa rahaa, kuinka saatuja tuloja voi paran-
taa ja miten mainokset toteutetaan koodipohjaisesti.

Tässä opinnäytetyössä tutkittiin mainosten käyttöönottoa, jota dokumentoitiin
päivittäin pelin kehityksen aikana luomalla peli Unity-pelimoottorissa ja laitta-
malla siihen lopuksi mainokset käyttöön. Mainosalustana käytettiin Unityn omaa
Unity Ads -alustaa. Työssä tutkittiin pelinsisäistä valuuttaa ja taloutta sekä palk-
kiomainoksia monetisaatiotapoina, mainostukseen liittyviä tuotto-odotuksia ja
tuottoa muihin alustoihin ja mainosjärjestelmiin verrattuna.

Opinnäytetyössä luotiin tarkat ohjeet, kuinka mainokset saadaan käyttöönotet-
tua luodun työpäiväkirjan avulla. Käytännön työssä ja siitä luodusta ohjeistuk-
sesta käydään läpi vain mainosten käyttöönottoa, eikä muita monetisaatiota-
poja. Kaikki tarpeelliset vaiheet mainosten käyttöönottoon käydään läpi vaihe
vaiheelta koodipohjaisesti Unity Ads -alustaa käyttäen. Opinnäytetyössä tarkas-
tellaan monetisaatiota ja tuotto-odotuksia painotetusti Unity-pelimoottorissa ja
Unity Ads -alustassa.

2 Monetisaatio

Moni kehittäjä pelkää, että mainokset saavat pelaajat lopettamaan pelaamisen,
mutta ilman monetisaatiota kehittäjä ei tule saamaan työstään yhtään rahaa.
Tämän vuoksi hyvin suunniteltu ja toteutettu monetisaatiostrategia auttaa pelaa-
jaa ja kehittäjää, jolloin monetisaatio toimii pelaajien kanssa eikä heitä vastaan.

Esimerkki hyvästä monetisaatiostrategian tuloksesta löytyy Unityn tekemästä
raportista vuodelta 2022, jossa huomattiin hyvin toteutettu mainostus vaikutti
pelaajien jäämiseen positiivisesti. Datasta huomattiin, että peleissä, joissa oli

mainokset, etenkin pelit, jotka sisälsivät palkkiomainoksia, saivat pelaajat palaamaan peliin huomattavasti useammin. Pelejä jäi pelaamaan 26 % pelaajista päivän aikana, 8 % viikon aikana ja jopa 3 % kuukauden päästä. Pelaajat pelasivat peliä enemmän verrattuna peliin, jossa ei ollut mainoksia (Unity Gaming Report 2022, s. 14). Mainokset eivät siis suoraan saa pelaajia häipymään, vaan se, kuinka niitä käytetään. Pelikehityksessä kannattaa miettiä, mitä pelaajana haluaa ja kuinka välttää mainosten liikakäyttöä.

2.1 Pelinsisäinen valuutta ja taloudellisuus

Pelissä hyvin yleinen tapa luoda monetisaatiota on lisätä valuutan ja taloudellisuuden aspekti peliin, joka antaa esimerkiksi pelin esineille tai tavaramalle arvon. Oikein tehtynä pelaajat voivat maksaa näistä asioista. Peliin lisätään joku pelinsisäinen valuutta, jolla on arvoa pelimaailmassa. Valuuttaa ja taloudellisuutta voi luoda peliin etenkin, jos niillä keksii tavan parantaa käyttäjäkokemusta. Valuutan lisääminen mahdollistaa pelin laajentamisen myöhemmissäkin vaiheissa pelin jatkokehityksessä. (Monetization Strategy.)

Pelaajille voidaan palkita valuuttaa ajastetuilla palkinnoilla, palkkiomainoksilla tai ostamalla valuuttaa pelinsisäisillä ostoilla. Ajastetuilla palkinnoilla pelaajat palaavat peliin säännöllisesti. Pelin valuutan arvoa kannattaa kuitenkin seurata, ettei se tule liian kalliiksi tai liian arvottomaksi. Valuuttaa ei kiinnosta hankkia, jos valuutta on liian kallista pelaajille tai pelaajat eivät hyödy mitenkään ostamalla asioita korkeiden hintojen vuoksi. Liian alhainen valuutan hinta voi kiihdyttää pelin kulkua liian nopeaksi, jolloin pelaajakunta voi kyllästyä peliin, eikä valuutan keräämisellä tule olemaan väliä. (Monetization Strategy.)

Hyvin kehitetyn valuutan esimerkkinä Clash of Clans on käyttänyt strategiana rajoittaa sitä, paljonko pelaaja voi kerätä tiettyjä resursseja. Alemmalla tasolla resursseja ei voi ostaa yhtä paljon tai kerätä yhtä nopeasti, sillä pelaajalle tulee vastaan pelaajan tason mukainen raja. Pelaajan tulee käyttää resurssit päivityksiin tai joukkoihin, joiden kehittäminen vie aikaa. Valuutan maksimiraja nousee, kun resurssit ovat käytetty päivityksiin. Rajan noustessa pelaajan ostokset

maksavat enemmän ja päivitykset seuraavalla tasolle vievät kauemmin. Kuvassa 1 näkyy pelin valuutan enimmäisraja ja tuotanto tunnissa.



Kuva 1. Kuvankaappaus pelin oikeasta yläreunasta, jossa näytetään kullan tuoton enimmäismäärä ja sen tuotanto tunnissa (Clash of Clans, 2024).

Himotuin valuutta pelissä ovat jalokivet, joita voi kerätä hitaasti työtehtävillä tai ostaa suoraan oikealla rahalla. Jalokivillä päivityksiä saa nopeutettua, suoritettua heti tai saa ostettua poissuljettuja pelialueen ja hahmojen ulkonäköä muuttavia esineitä (kuva 2).



Kuva 2. Kuvankaappaus Clash of Clans -pelin sisäisestä kaupasta, jossa oikealla rahalla voi ostaa pelissä käytettävää valuutta (Clash of Clans, 2024).

2.2 Palkkiomainosten käyttö

Palkkiomainoksilla voidaan palkita pelaajaa vapaaehtoisesta mainoksen katso-
misesta. Pelaajaa voidaan palkita millä tahansa, kunhan se on pelaajan mie-
lestä reilua. Palkkion antaminen palkkiomainoksella on todella monipuolista, ja
palkinnoille löytyy melkein kaikista peleistä ainakin yksi käyttötarkoitus. Palkkio
voi olla esimerkiksi lisäelämä pelissä, tietty summa pelinsisäistä valuuttaa tai te-
hostus pelin etenemiseen. Palkkiomainoksia kannattaa yrittää rajoittaa kuten
pelivaluuttaa, sillä palkkioiden arvo voi laskea, jos niitä annetaan pelaajalle lii-
kaa. (Monetization Strategy.)

3 Mainostyypit

Unity Ads -mainosalustan mukana tulee kolme eri mainostyyppiä. Mainostyyp-
pejä Unity Ads SDK:ssa ovat väli-ikkuna-, palkkio- ja bannerimainokset. Mai-
noksia helposti sijoitetaan peleihin väärin, jos niiden käyttötarkouksia ei ymmär-
retä.

Mainostyyppin mukaan pelaajalle voidaan mainostaa suoraan videolla, mainos-
taa pelattavan alueen ulkopuolella häiritsemättä pelaajaa tai kannustaa heitä
katsomaan mainos palkinnosta. Peliä kehittäessä mainostamisen kanssa voi
olla luova, mutta mainostyypppejä kannattaa käyttää yleisesti niihin kehitettyihin
tarkoituksiin, jotta käyttäjäkokemus ei pilaannu.

3.1 Bannerimainokset

Bannerimainokset ovat suorakulmaisen muotoisia mainoksia, jotka esiintyvät
yleisesti näytön reunalla poissa pelattavalta alueelta. Mainosta voi hyödyntää
parhaiten tilanteessa, jossa mainos on näkyvissä ilman, että se häiritsee pelaa-
jaa (kuva 3).

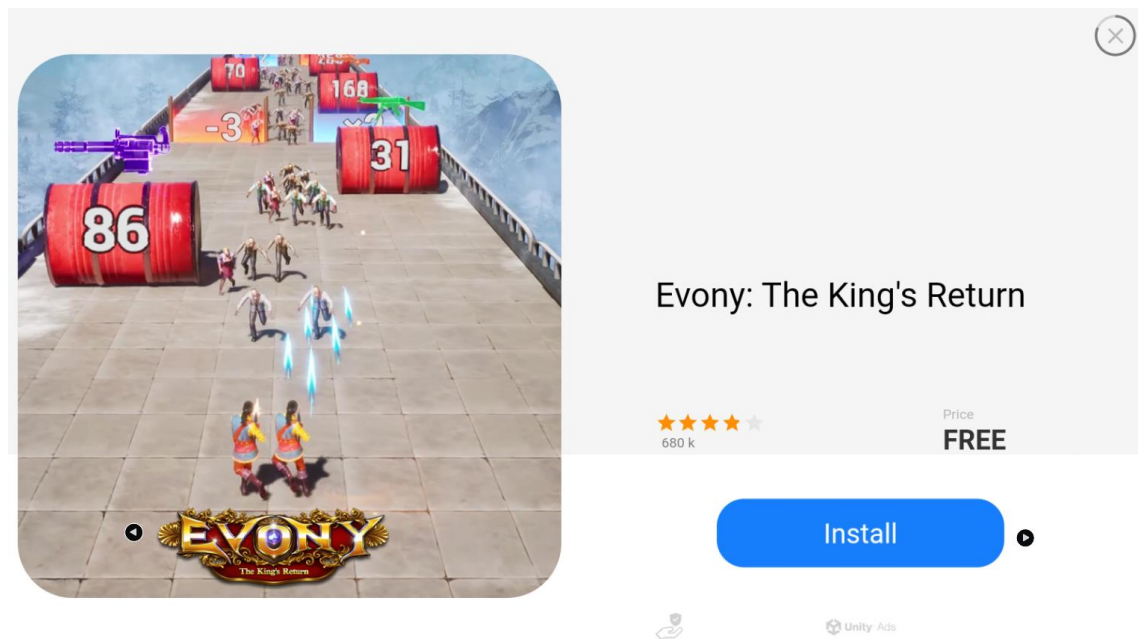


Kuva 3. Kuvankaappaus Hill Climb Racing -mobiilipelistä, jossa bannerimainos toisesta pelistä esitetään pelattavan alueen ulkopuolella näytön alaosassa, kun peli on pysäytetty. (Hill Climb Racing 2012.)

Bannerimainokset tuottavat vähiten Unity Ads:n mainostyypeistä, mutta niille kannattaa silti etsiä mahdollisimman paljon näyttöaikaa, sillä ne tuottavat kuitenkin rahaa (Admob Ads vs Unity Ads Comparison. 2022.). Bannerimainosta voi pitää aina esillä pelaajalle esimerkiksi valikoissa tai pelin aikana ulkona pelattavalta alueelta. Bannerimainokset yleisesti piilotetaan pelaamisen ajaksi, sillä mainokset saattavat häiritä keskittymistä tai mainosta saatetaan vahingossa painaa pelin aikana.

3.2 Väli-ikkunamainokset

Väli-ikkunamainokset toistavat videon koko laitteen näytöllä isona mainoksena. Mainokset sisältävät painikkeet sulkemiseen ja mainoksen ohittamiseen. Väli-ikkunamainoksia toistetaan eniten tasojen välillä tai jos peliä on pelattu tietyn verran, sillä silloin ne eivät riko pelin kulkua yhtä helposti. Yleisesti väli-ikkunamainokset esiintyvät videoina, joita ei voi heti ohittaa (kuva 4).

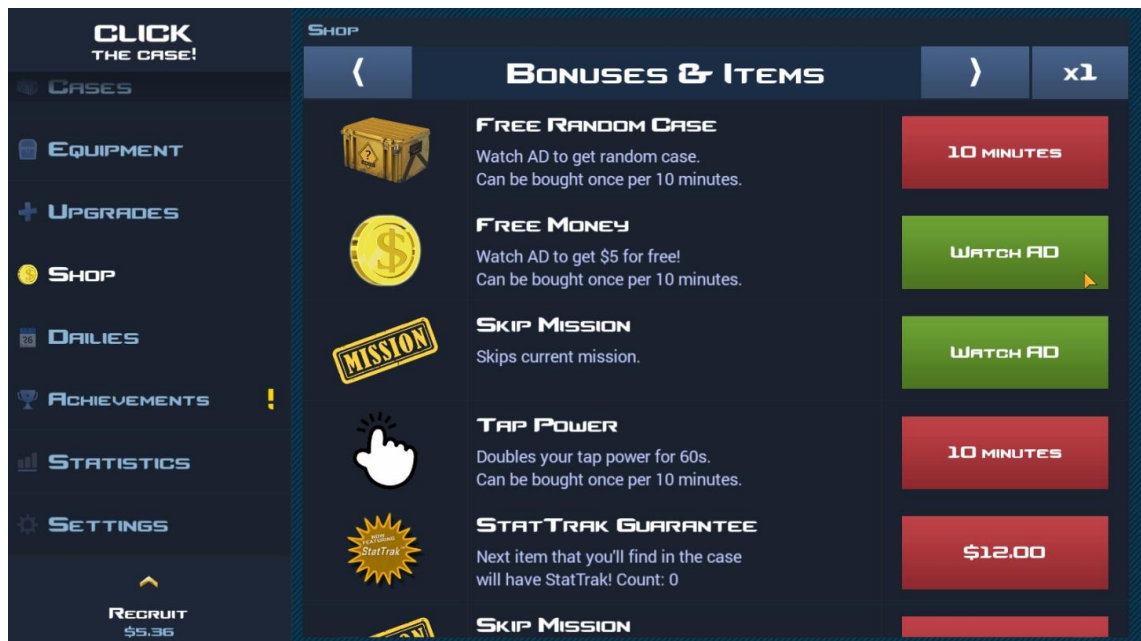


Kuva 4. Väli-ikkunamainos, jossa mainoksen sulkemisessa on ajastin (Hill Climb Racing 2012).

Väli-ikkunamainokset tuottavat eniten rahaa mainostyypeistä palkkiomainosten kanssa Unity Adsisssa (Admob Ads vs Unity Ads Comparison. 2022). Mainokset kannattaa kehittää pelaaja mielessä, sillä niiden väärinkäyttö voi helposti lannistaa pelaajia pois pelistä. Mainokset eivät saa olla häiritseviä pelaajalle ja kannattaa näkyä kohtuullisella aikavälillä.

3.3 Palkkiomainokset

Palkkiomainokset palkitsevat pelaajia, jotka vapaaehtoisesti katsovat mainoksen ansaitakseen jotakin pelinsisäistä etua. Pelaajalle tulee kehote katsoa mainos, jonka jälkeen pelaajaa palkitaan. Mainos toimii väli-ikkunamainoksen tavalla ja sisältää ohitukseen ja sulkemiseen tarkoitetut painikkeet (kuva 5).



Kuva 5. Kuva Case Clicker -mobiilipelistä, jossa pelaajalle tarjotaan mahdollisuutta katsoa mainos palkinnon saamiseksi (Case Clicker 2015).

3.4 Muut mainostyypit

Muut mainostusalustat voivat tarjota enemmän mainostyyppejä Unity Adsin kolmen mainostyyppin lisäksi. Muita mainostyyppejä on hyvä käyttää, jos Unity Adsin mainostyyppien määrä rajoittaa mainostamista pelissä pelin tyylin vuoksi.

Esimerkkinä Google AdMob tarjoaa kuutta mainostyyppiä:

- sovelluksen avaus
- väli-ikkunamainos
- bannerimainos
- kehittynyt natiivimainos
- palkkiomainos
- palkittu väli-ikkunamainos.

Sovelluksen avauksen aikana tapahtuva mainos esittää mainoksen, kun sovellus käynnistetään tai jos siihen palataan myöhemmin, kun se on vielä käynnissä.

Kehittynyt natiivimainos mahdollistaa mainoksen ulkonäön ja käyttöliittymän muuntautua näyttämään peliltä tai sovellukselta ja tuntumaan olevan osa peliä, jossa se esiintyy.

Palkittu väli-ikkunamainos palkitsee pelaajia, jotka katsovat vapaaehtoisesti mainoksen toiseen tasoon siirtyessä. Tämä mainosformaatti on vielä AdMobilla kehityksessä, ja sen voi helposti luoda Unity Adsissä.

4 Tuotto-odotukset

Jotta julkaisijana saa toivotun verran tuottoa, niin kannattaa verrata eri mainosalustoja. Mainosalustaa etsiessä pelille kannattaa tutkia alustalta saatavat mainostyypit ja alustan tuoton keskiarvo, sillä tuotto-odotukset riippuvat mainosalustasta ja mainosten markkinasta.

4.1 CPM ja eCPM

Mainosten tuottoa lasketaan käyttämällä mittayksiköitä CPM (Cost per mille) ja eCPM (Effective cost per mille).

CPM:n avulla lasketaan, kuinka paljon mainostajat keskimäärin maksavat julkaisijalle mainosten näyttämisestä tuhatta mainosnäyttökertaa kohden. CPM:n arvo ei muutu, vaikka mainoksesta napautetaan tai jos sen sisältö ladataan. CPM lasketaan seuraavasti:

$$CPM = \left(\frac{\textit{kampanjan hinta}}{\textit{mainosnäyttökerrat}} \right) * 1000 \quad (1)$$

Esimerkiksi jos mainostajan kampanjan hinta on 7 500 € ja mainosnäyttökertoja saadaan 1 000 000, CPM on 7,5 €. Mainostaja maksaa siis julkaisijalle tuhatta näyttökertaa kohden 7,5 €. (CPM vs eCPM – Confused? We Got You Covered.)

eCPM:n (Effective cost per mille) avulla lasketaan, kuinka paljon julkaisija saa yhteensä tuottoa tuhatta mainosnäyttökertaa kohden. Tätä käytetään silloin, kun

julkaisijalla on enemmän kuin yksi mainosalusta ja CPM:n arvot vaihtelevat. eCPM lasketaan seuraavasti:

$$eCPM = \left(\frac{\text{tuottojen summa}}{\text{mainosnäyttökerrat}} \right) * 1000 \quad (2)$$

Esimerkiksi jos julkaisija saa yhteensä 6 000 € tuottoa 1 750 000 mainosnäyttökerrasta, eCPM on noin 3,42 € tuhatta näyttökertaa kohden. (CPM vs eCPM – Confused? We Got You Covered.)

4.2 Muut mainosalustat

Unity Mediation SDK (Software development kit) mahdollistaa muiden mainosvälittäjien kanssa yhteistyön. Välityksen avulla pelissä on enemmän kilpailua mainosten näyttämiseen ja lisää tuottoa peliin.

Unity Mediation SDK tarjoaa tietoverkot:

- Unity ADS
- Meta (Audience Network)
- AdColony
- Vungle
- ironSource
- Google AdMob
- AppLovin

Suurimmat mainosalustat tietoverkoista ovat AdMob ja Meta, mutta kaikki tietoverkot ovat saatavilla Unity Mediationia käyttäville, jos toivoo niitä käyttää (Kovalenko, 2023).

4.3 Alustojen tuotto

Unity Ads -mainosalustan tuotto on vaihdellut paljon vuosien aikana. Vuonna 2016 Unityn tukitiimin jäsen totesi nyt poistetussa artikkelissa Unityn CPM:n

olevan keskimäärin 6–12 \$ (Lara, 2016). Uudemmassa artikkelissa vuodelta 2023 Unity Adsin CPM:n laskettiin olevan 0,45–11,67 \$ (Kovalenko, 2023).

AdMob on Googlen omistama, joten myös Googlen AdSense -palvelu on siihen yhteydessä. Tämän avulla Googlen mainoksia voi käyttää pelien ulkopuolella kuten sovelluksissa. Googlen mainokset tavoittavat suurimman osan käyttäjistä koko verkossa, sillä Googlen mainostusta käyttää noin 44 miljoonaa käyttäjää verkkosivuissa, sovelluksissa ja peleissä (Websites using Google AdSense). Unity Adsiin verrattuna hinnat ovat hieman alhaisemmat, 0,2–10 \$ (Kovalenko, 2023). AdMob tarjoaa myös kolmea mainostyyppiä enemmän Unity Adsiin verrattuna, jonka vuoksi myös AdMobin tuotto on suurempi (Elliz, 2023).

Metan Audience Network tavoittelee Metan omistamien sovellusten, Instagramin, Metan, Messengerin ja WhatsAppin käyttäjäkunnan mainoksillaan. Tämän lisäksi Facebook vuonna 2016 oli julkaissut kerätystä datasta artikkelin, jossa ilmoitettiin yli miljardin käyttäjän näkevän kuukaudessa Audience Networkin kautta mainoksen (Audience Network Update, 2016). Audience Network liittyi Unityn kumppaniksi mainostamisessa vuonna 2019 (Unity Ads Now Offers Access to Meta Audience Network Through Unity Mediation, 2019).

Unity Ads ei ole AdMob- ja Audience Network -mainosalustoihin verrattuna paras, mutta tämä ei tee siitä huonoa. Mikäli kehittäjä testaa muita alustoja, kannattaa käyttää niitä Unity Mediationin kautta. Mainosalustojen CPM:ää on vaikeaa määrittää, sillä mainosten markkinan vuoksi CPM:t voivat vaihdella hyvin nopeasti. Projekti osoitti, että erilaisia mainosalustoja kannattaa kokeilla.

5 Mainosten markkinat

5.1 Mainosten hinnan määrittäminen

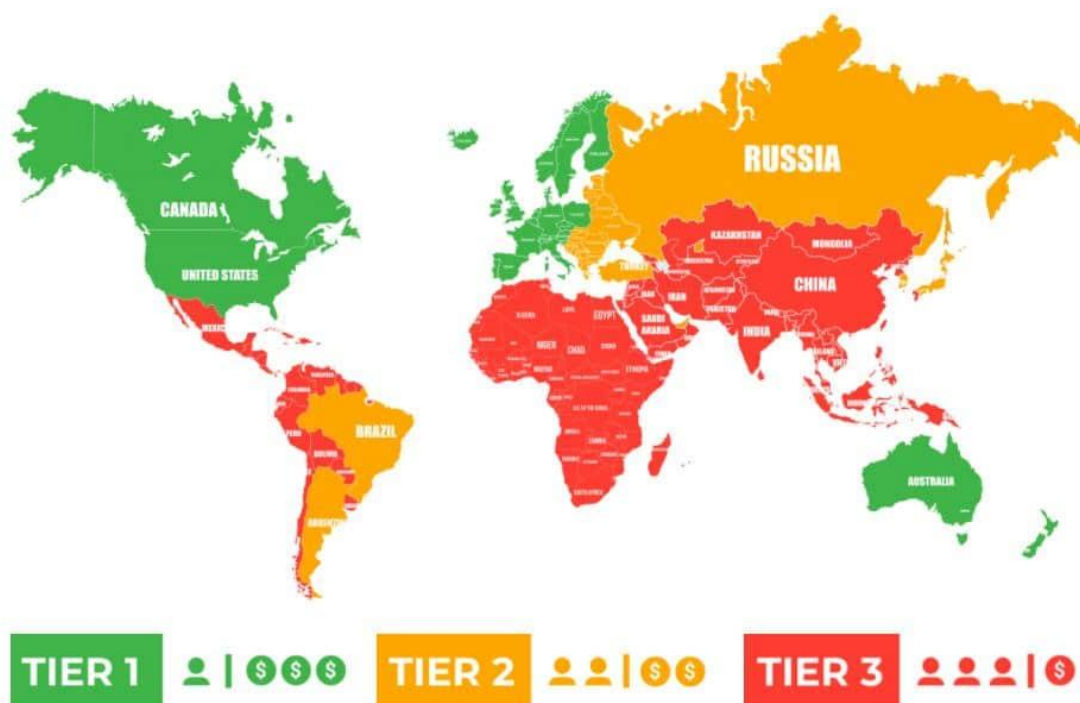
Mainosten hinnan määrittäminen riippuu monista vaikuttavista tekijöistä. Tekijöiden vuoksi CPM vaihtelee laajasti eri mainosalustoilla. Peliä voi ajoissa lähteä

kehittämään määryksien ympärille, jos tuottoa toivoo nostavansa. Suurimpia mainosten vaikuttavia tekijöitä ovat alue ja väestötilastot sekä alusta ja mainostyyppi.

Mainoksiin vaikuttaa myös kellonaika, jos mainos katsotaan loppuun, jos mainosta klikataan ja jos mainoksessa esiintyvä sisältö ladataan tai ostetaan. (Revenue and payment.)

Mainosten alueet ja väestötilastot

Mainostajat ovat jakaneet melkein kaikki maailman valtiot ja alueet kolmelle eri tasolle. Mainostajat yleisesti toivovat ensimmäisen tason alueita, sillä niissä on hyvä tuottoprosentti ja ostokunta. Toisen ja kolmannen tason maissakin on omat hyödyt kuten alhaiset CPC:t (Cost-per-click), joilla mainostajat määrittelevät, kuinka paljon he ovat valmiita maksamaan yhdestä klikkauksesta mainoskampanjan aikana. Kuvassa 3 on visualisoitu nämä alueet tason mukaan. Liite 2 sisältää kaikki arvioidut maat ja niiden sijoitukset.



Kuva 6. Kuva maailmankartasta, jossa alueet on värjätty tason mukaan. (Shakal, 2024.)

Taso 1 koostuu maista, joissa asukkailla on suuremmat palkat, hyvä talous ja enemmän rahaa käytettävissä. Maissa myös puhutaan huomattavasti enemmän sujuvaa englantia kuin muualla ja puhelinten käyttö on hyvin yleistä. Näissä maissa CPC on korkeampi kuin muualla, sillä kilpailu on suurempaa tällä tasolla.

Taso 2 koostuu maista, joissa asukkailla on suhteellisen hyvät palkat, talous ja kohtuullisesti rahaa käytettävissä muihin ostoksiin. Näissä maissa kilpailu ei ole yhtä raskasta, tuotteet ovat halvempia, kustannusten määrä per klikkaus on halvempaa ja lait rajoittavat vähemmän mainostamista.

Taso 3 koostuu maista, joissa talous ja palkka eivät ole kovin hyviä ja asukkailla on rajoitetusti tai ei ollenkaan rahaa käytettävissä muihin ostoksiin. Kustannusten määrä per klikkaus on todella halpaa ja rajoittavia lakeja ei löydy lähes ollenkaan. Asiakkaiden löytämiseksi mainosten täytyy painotetusti mainostaa vain paikallisia ostoksia. (Carr. 2020; Malekos. 2023; Understanding Tier 1, 2 & 3 countries.)

Mainosten tyypit ja alustat

Eri mainostyypit tuottavat eri summia, joista väli-ikkuna ja palkkiomainokset maksavat julkaisijalle eniten, jonka jälkeen tulevat bannerimainokset. Tähän myös vaikuttaa alusta, eli jos käyttäjä käyttää Android- tai iPhone-laitetta. (Admob Ads vs Unity Ads Comparison. 2022.)

5.2 Mainospaikan osto

Mainostajat maksavat kehittäjille mainoksien näyttämisestä ostamalla mainospaikkoja kehittäjältä huutokaupoilla. Kaikki mainosalustat tarjoavat oman hinnan mainospaikalle riippuen niiden monista muuttujista, kuten alueesta ja alustasta.

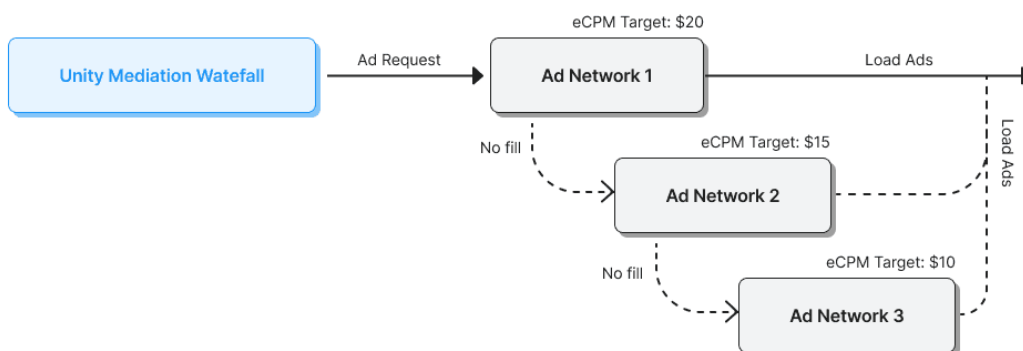
Mainospaikkojen ostamiselle on kehitetty järjestelmä, jonka avulla markkinoilla on kilpailua.

Huutokauppa

Unity Ads hankkii mainoksia esittämällä mainostilaa huutokaupalla. Kun pelissä pyydetään mainosta, mainostajille tulee huutokauppa pelin mainoksen näyttökerrasta. Mainostajat ovat asettaneet valmiit arvot omille mainoskampanjoille, ja huutokauppa tapahtuu automaattisesti. Mainostajat määrittävät maksimihinnan yhdelle mainosnäyttökerralle ja kuinka monta mainosta mainoskampanjan budjetilla saa ostaa. Mainostaja, joka maksaa eniten näyttökerrasta, saa mainoksen näytettyä pelissä. (Rewarded ads best practices; Determine a bid strategy based on your goals).

Vesiputous

Unityssä toinen tapa luoda kilpailua mainonnan hankinnassa on vesiputouksella. Vesiputouksessa määritellään toivottu eCPM ylimmäksi, ja mikäli ostajia ei löydy, hintaa lasketaan alas yksi taso kerrallaan, kunnes ostaja löytyy. Tämä mahdollistaa tasaisemman ja suuremman virran tuotossa verrattuna huutokauppaan (kuva 7). (Waterfalls.)



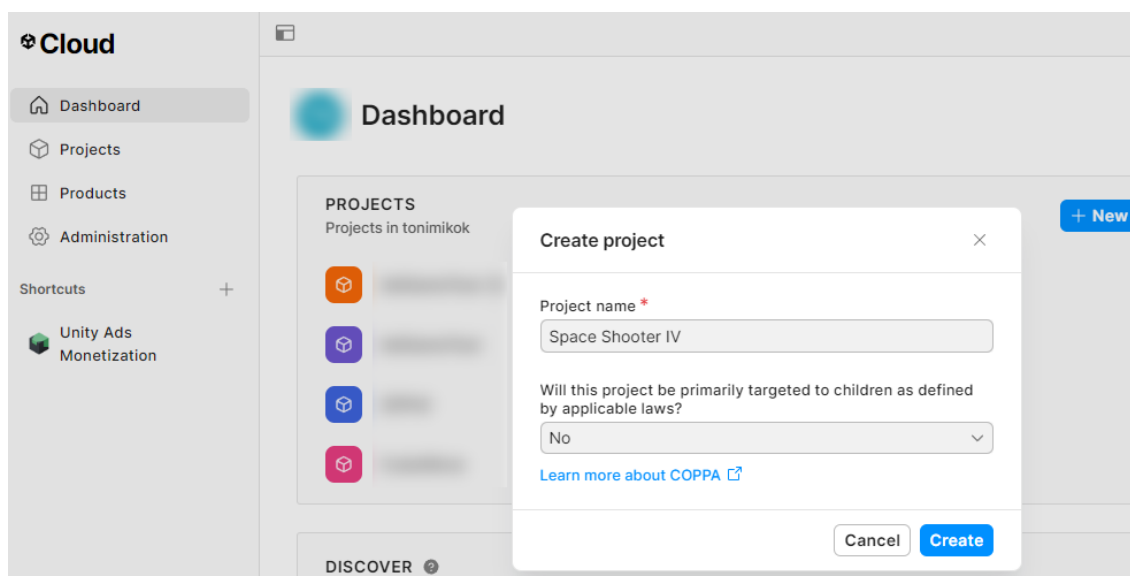
Kuva 7. Diagrammi vesiputouksen toiminnasta mainonnassa (Waterfalls).

Huutokaupat ja vesiputoukset voidaan määrittää Unity Cloud -palvelun mainostamisen työpöydässä, kun projektiin on lisätty mainokset.

6 Mainosten lisääminen Unityssä

Mainosten lisääminen Unityyn on hyvin dokumentoitu ja suhteellisen yksinkertaiseksi tehty prosessi, jotta pelin tekijät saavat vaivatta mainokset peliin. Ohjeistuksessa mainosten lisäämiseen käytetään Unity Ads -palvelua. Unity Ads on yksi monista vaihtoehdoista, jota mainostukseen voi käyttää. Unity Ads palvelun vanhin tuettu versio Unityssä on 2022.3.

Mainosten käyttöönotto alkaa Unityn Cloud -palvelussa osoitteessa cloud.unity.com. Sivustolla pyydetään ensin luomaan käyttäjä tai kirjautumaan sisään olemassa olevaan käyttäjään. Käyttäjäprofiililla kirjaututaan lisäksi sisään Unity Hub -ohjelmistossa tai suoraan Unity -editorin sisällä. Kirjautumisen jälkeen Unity Cloud -palvelun etusivulla esiintyy työpöytä, josta voi luoda uuden tai valita aiemmin luodun projektin (kuva 8). Projektin voi myös luoda Projects -välilehdeltä valitsemalla sen vasemmasta laidasta.



Kuva 8. Unity Cloud -sivusto, jossa New -painikkeen klikkauksen jälkeen avautuvan ikkunan tekstikenttään on syötetty projektin nimeksi "Space Shooter IV".

Projektin luomisen tai valinnan jälkeen siihen tulee vielä erikseen lisätä mainokset. Tämä onnistuu valitsemalla projekti ja suuntaamalla alas palveluiden välilehteen, josta löytyy Unity Ads Monetization -widgetti. Palvelun etusivulta painetaan Launch-painiketta ja avautuvasta ikkunasta painetaan vielä Get started -painiketta, johon täytetään projekti- ja käyttäjätiedot. Täältä voi myös halutesaan asettaa mahdollisen mainostenvälittäjän mainostukseen mukaan.

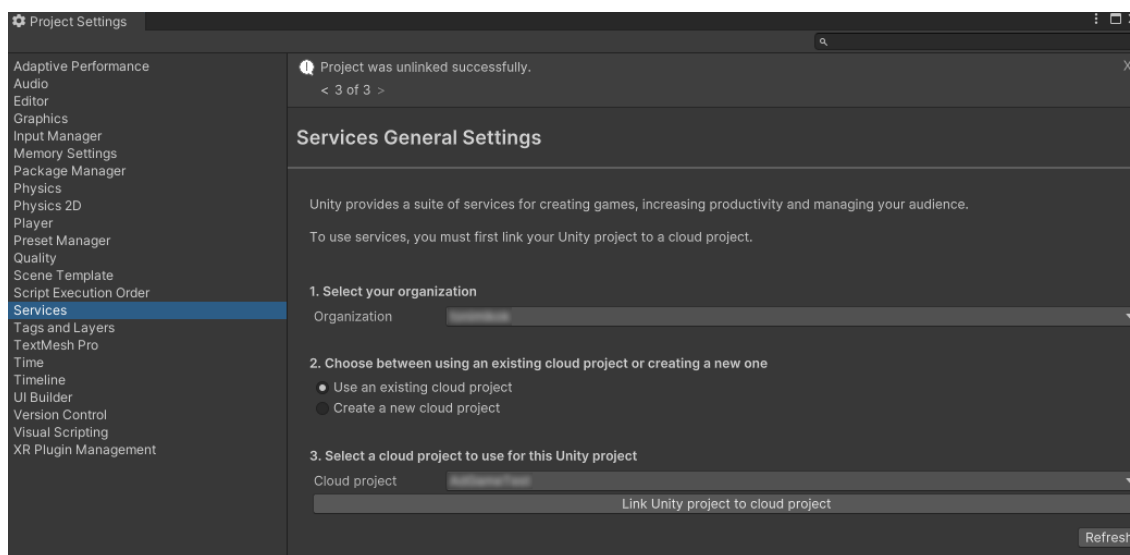
Käyttäjätietojen ja projektin mainostuksen määrittämisen jälkeen verkkosivustolle avautuu uusi ikkuna, josta tulee ottaa talteen projektin omakohtaiset tunnukset Android- ja iOS-alustoille. Tunnuksia käytetään projektin liitokseen Unityn sisällä.

6.1 Projektin liitos palveluun

Projekti liitetään pilveen Unity-projektin sisällä Unity SDK:n avulla. Unityssä tarvitsee asentaa Advertisement Legacy paketti Package Manager -ikkunasta (Window > Packet Manager). Paketin asentamisen jälkeen Unityn ikkunassa siirrytään Services -valikkoon projektin liittämiseksi.

Services -valikosta pääsee General Settings -ikkunaan, jossa organisaation ja projektin tiedot täytetään. Oikeiden tietojen syöttämisen jälkeen liitos palveluun

viimeistellään painamalla Link Unity project to cloud project -painiketta (kuva 9).



Kuva 9. Unityn Project Settings -ikkuna, jossa Cloud -palvelun linkitykseen vaadittavat organisaation ja projektin tiedot on syötetty sisään.

6.2 Mainosten alustus

Projekti, joka on linkitetty Unity Cloud -palvelun kanssa, voi ottaa mainokset käyttöön. Unity Ads SDK API-rajapinta (Application Programming Interface) tarjoaa laajan valikoiman komentoja mainoksien käytölle. Näitä komentoja voi käyttää kutsumalla API-rajapinnan nimiavaruutta projektin skriptien alussa (using UnityEngine.Advertisements). Mainokset tulee alustaa omalla skriptillään ennen, kun niitä voidaan käyttää.

Esimerkkikoodi 1:ssä mainokset alustetaan koodilla, joka on implementoitu viikaturvalliseksi väärää kutsuja varten. Projektille aiemmin luodut Android- ja iOS -tunnukset syötetään skriptin alussa oleviin muuttujiin. Pelin mainosten testaaminen editorissa valitsee automaattisesti Android-alustan testaukseen. Mainokset alustetaan esimerkkikoodi 1:n viimeisellä rivillä, jossa funktioon syötetään ehtolauseessa valittu alustan tunnus ja testaamisen tila. Koodia voi julkaisun aikana optimoida poistamalla muuttujat sekä ehtolauseet ja syöttämällä ne suoraan sisään funktioon, mikäli mobiilipeliä kehitetään vain tietyllä alustalle.

```

using UnityEngine.Advertisements; // Nimiavaruuden määrittely
...
string _androidGameId; // Android sovellusnumero
string _iosGameId; // IOS sovellusnumero
bool _testMode = true; // Testauksen aikana TOSI
private _gameId; // Muuttuja johon ID säilötään

public void InitializeAds()
{
#if UNITY_IOS
    _gameId = _iosGameId;
#elif UNITY_ANDROID
    _gameId = _androidGameId;
#elif UNITY_EDITOR
    _gameId = _androidGameId; // Oletuksena Android
#endif

if (!Advertisement.isInitialized && Advertisement.isSupported)
{
    Advertisement.Initialize(_gameId, _testMode, this);
}
}

```

Esimerkkikoodi 1. Vikaturvalla laadittu mainosten käyttöönottokoodi.

Pelin alustan vaihtaminen Unityssä Android-alustaan voi korjata ongelman, mikäli alustus tai mainokset eivät alkuvaiheessa toimi. Mainoksia voidaan alustuksen jälkeen käyttää ja niitä voidaan luoda kolmesta mainostyypistä, jotka Unity Ads tarjoaa.

6.3 Mainosten käyttö

Mainosten alustuksen jälkeen, mainosta käytetään määrittelemällä ensimmäisenä sen tyyppi ja alusta merkijonolla, jonka jälkeen mainoksen sisältö ladataan ja näytetään käyttäjälle. Mainosten käyttötavat ja niiden funktiot eroavat eri mainostyypeissä.

Mainosten alustan määrittäminen

Mainoksilla on oma merkkijono niiden alustaan ja mainostyyppiin liittyen. Näillä merkkijonoilla määritellään, minkälainen mainos on kyseessä sitä ladatessa ja näyttäessä. Taulukko 1:ssä on kaikki merkkijonot ja alustat, joita Unity Ads tukee.

Mainostyyppi	Android	iOS
Väli-ikkuna	Interstitial_Android	Interstitial_iOS
Palkkio	Banner_Android	Banner_iOS
Banneri	Rewarded_Android	Rewarded_iOS

Taulukko 1. Merkkijonot, joilla määritellään mainostyyppi ja alusta.

Merkkijono määritellään muuttujaan mainostyyppistä riippumatta samalla tavalla jokaisessa skriptissä. Merkkijono syötetään lataamisen funktioon alustan selvityksen jälkeen. Kaikissa esimerkkikoodeissa tämä muuttuja esiintyy nimellä `_adUnitId`.

Mainosten lataaminen

Mainoksen sisältö tulee ladata ennen mainoksen näyttämistä. Mainosten lataamisen kesto on riippuvainen pelaajan verkkoyhteyden nopeudesta, joten mainosten lataus kannattaa suorittaa pelin käynnistyksen aikana. Esimerkkikoodeissa 2 väli-ikkunamainoksen latauksessa samaa pohjaa käyttäen voidaan myös alustan merkkijonoa vaihtamalla ladata palkkiomainos.

```
// Toivomme väli-ikkunamainosta, joten laitamme seuraavat arvot:
string _androidAdUnitId = "Interstitial_Android";
string _iosAdUnitId = "Interstitial_iOS";

// Tarkista ja määritä alusta _adUnitId muuttujaan
_adUnitId = (Application.platform == RuntimePlatform.IPhonePlayer)
    ? _iosAdUnitId
    : _androidAdUnitId;

// Väli-ikkunamainoksen lataus
Advertisement.Load(_adUnitId, this);
```

Esimerkkikoodi 2. Väli-ikkunamainoksen lataaminen alustatarkastuksen kanssa.

Bannerimainos toimii samalla periaatteella, mutta sitä ei voi ladata suoraan samalla funktiolla, jota väli-ikkuna- ja bannerimainos käyttää. Ennen bannerimainoksen lataamista, asetukset tulee olla takaisinkutsufunktioita varten (esimerkkikoodi 3).

```
// Mainosten asetusten luonti
BannerLoadOptions options = new BannerLoadOptions
{
    loadCallback = OnBannerLoaded,
    errorCallback = OnBannerError
};

// Mainoksen lataaminen asetusten kera
Advertisement.Banner.Load(_adUnitId, options);
```

Esimerkkikoodi 3. Bannerimainoksen lataamiseen tarvittavat asetukset ja funktio.

Mainosten näyttäminen ja piilottaminen

Mainoksen sisällön lataamisen jälkeen mainos voidaan esittää pelaajalle. Väli-ikkuna- ja palkkiomainokset toimivat taas samalla näyttämiseen luodulla

funktiolla, ja bannerimainos tarvitsee erikseen taas muutamia takaisinkutsun asetuksia.

Väli-ikkuna- ja palkkiomainokset suljetaan mainoksiin valmiiksi rakennetuilla painikkeilla, mutta bannerimainos tulee piilottaa erikseen funktiolla. Bannerimainoksen funktiota tulee käyttää usein, sillä bannerimainos ilmenee kaiken päällä, jopa muiden mainosten (esimerkkikoodi 4).

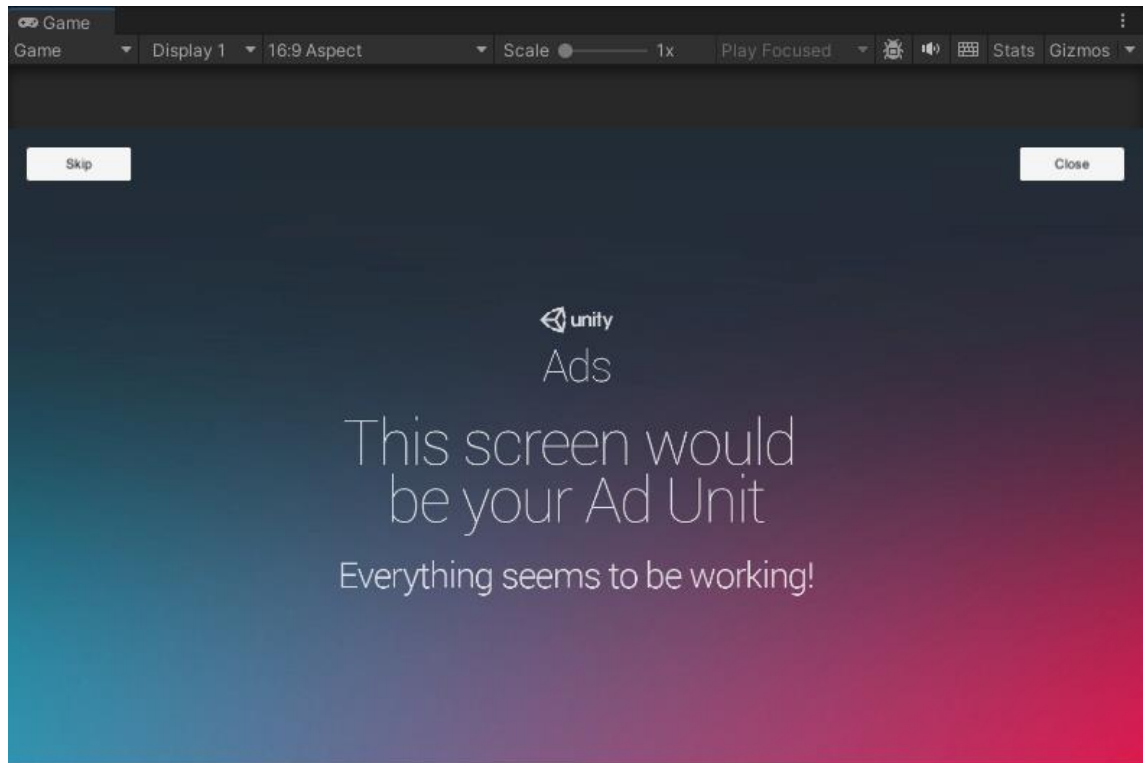
```
// Väli-ikkuna ja palkkio mainoksen näyttö
Advertisement.Show(_adUnitId, this);
...
// Bannerimainoksen näytön takaisinkutsu   $asetukset
BannerOptions options = new BannerOptions
{
    clickCallback = OnBannerClicked,
    hideCallback = OnBannerHidden,
    showCallback = OnBannerShown
};

// Näytä ladattu bannerimainos
Advertisement.Banner.Show(_adUnitId, options);

// Piilota ladattu bannerimainos
Advertisement.Banner.Hide(_adUnitId, options);
```

Esimerkkikoodi 4. Mainosten näyttämiseen ja piilottamiseen luodut funktiot niiden lataamisen jälkeen.

Mainostuksen toimivuutta voi testata helposti luomalla Unityyn painikkeet lataamiseen, näyttämiseen ja bannerimainoksen piilottamiseen. Jos mainos ei toimi tai siinä ilmenee muita ongelmia, sitä voidaan korjata SDK:n mukana tulevilla rajapinnoilla ja funktioilla. Unityssä mainoksen testaaminen tuo esille kuvan pelin sisälle, joka ilmenee vain, jos mainoksen näyttäminen onnistuu (kuva 10).



Kuva 10. Kuvankaappaus Unityn peli-ikkunasta, jossa on vahvistus väli-ikkunamainoksen toiminnasta.

Rajapinnat ja lisäfunktiot

Lisäfunktiot, listojen muuttujat ja rajapinnat mahdollistavat enemmän joustavuutta mainoksiin ja niiden toimintoihin. Lisäfunktiolla voidaan suorittaa koodia esimerkiksi, kun mainos avataan tai jos mainos ei lataudu oikein.

SDK:n rajapintoja löytyy kolme:

- IUnityAdsInitializationListener
- IUnityAdsLoadListener
- IUnityAdsShowListener.

Rajapinnat nimiensä mukaan seuraavat alustamisen, lataamisen ja näyttämisen tapahtumia. Jokaisessa rajapinnassa on paljon ominaisuuksia, joten jokaisesta

rajapinnasta esitellään vain yksi esimerkki. Kaikki luokat, listojen muuttujat ja rajapinnat löytyvät liite 1:stä.

Alustamisen rajapinta sisältää funktiot alustuksen jälkeen tapahtuviin toimiin ja funktion siihen, jos alustus epäonnistuu. Esimerkkikoodi 5:ssä on yksinkertainen funktio, jolla voi vahvistaa alustuksen valmistumisen.

```
public void OnInitializationComplete()
{
    Debug.Log("Alustus on valmis!");
}
```

Esimerkkikoodi 5. Funktio rajapinnasta, jolla voi kuitata alustuksen tilan.

Lataamisen rajapinnasta löytyvät funktiot, joita voi käyttää, kun mainos on ladattu tai jos mainoksen lataus epäonnistuu. Rajapinnasta löytyy vian korjaamiseen sopiva funktio, jolla voi saada palautetta tilanteesta, jossa mainoksen lataus epäonnistuu (esimerkkikoodi 6).

```
// Funktio, joka käynnistyy jos mainos epäonnistuu näytössä
void OnUnityAdsFailedToLoad (string _adUnitId, UnityAdsLoadError
error, string message) {
    Debug.Log($"Virhe: ` {_adUnitId} - {error} - {message}`");
}
```

Esimerkkikoodi 6. Lataamisen rajapintaan luotu funktio, joka kertoo syyn lataamisen epäonnistumiseen lukemalla palautetut arvot.

Näyttämisen rajapinta sisältää funktioita mainosten näyttämisen aikana tapahtuviin tilanteisiin. Mainokselle löytyvät funktiot, joita voi käyttää, kun mainoksen näyttäminen epäonnistuu, mainos alkaa, mainoksesta napautetaan ja kun mainoksen loppuessa.

Mainoksen loppumiseen luotua funktiota on hyvä käyttää palkkiomainosten kanssa, sillä funktiossa palautetaan UnityAdsShowCompletionState, jonka avulla kolmesta arvosta (SKIPPED, COMPLETED, UNKNOWN) voidaan

tarkistaa, ansaitseeko käyttäjä palkintoa mainoksesta. Kuva 11:ssä on funktio, joka suorittaa koodia, kun mainos on katsottu loppuun.

```
1 reference
public void OnUnityAdsShowComplete(string adUnitId, UnityAdsShowCompletionState showCompletionState)
{
    if (adUnitId.Equals(_adUnitId) && showCompletionState.Equals(UnityAdsShowCompletionState.COMPLETED))
    {
        // Tee koodia
    }
}
```

Kuva 11. Kuvankaappaus funktiosta, jossa mainoksen katsomisen jälkeen jos-funktiolla suoritetaan koodia, mikäli mainoksen palautusarvo vastaa COMPLETED-arvoa.

7 Yhteenveto

Opinnäytetyössä oli tavoitteena oppia mainostamisesta, mainoksiin liittyvistä kysymyksistä, tuotto-odotuksista ja myös järjestelmien toimivuudesta koodipohjaisesti. Työn jälkeen voin sanoa itse, että olen saanut perehdyttyä aiheeseen ja kysymyksiini löytänyt vastaukset.

Mainosten markkinoista oli hyvin mielenkiintoista oppia tietoa ja tutkia, kuinka kaikki toimii. Mielestäni mainosalustojen tulisi olla avoimempia keskimääräisistä tuloista eikä laittaa käyttäjiä itse hakemaan eri tuloksia monista lähteistä. Vaikka valitettavasti CPM on käyttäjäkohtaista ja sitä on vaikea mitata, uskon, että mainosalustat voisivat julkaista maksutietoja kehittäjille olemassa olevista maksuista. Tämän lisäksi hintoja ei voi tutkia vanhasta datasta kunnolla, sillä mainosten hinnat teknologian kasvamisen vuoksi ovat muuttuneet.

Unity Ads -alusta oli testauksesta suhteellisen yksinkertainen ja loistava mainosalusta aloittelijoille sekä kehittyneille kehittäjille. Palvelu on suoraviivainen, suhteellisen helppo käyttöönottaa ja tarjoaa tilaa laajennukselle. Mainosten alustaminen peliin oli miellyttävä kokemus. Unity Mediationin avulla mainostamista ja pelin tuottoa voi lisätä, minkä takia edistynyt mainostaminenkin onnistuu sitä hakeville.

Työn tekeminen innosti tutkimaan lisää aihealueita tulevaisuudessa. Voin itsevarmuudella myös sanoa, että työtä pystyy käyttämään ohjeena, jos itse haluaa lähteä toteuttamaan mainostusta peleihin tai jos mainostuksesta tarvitsee laajan ymmärryksen.

Lähteet

Admob Ads vs Unity Ads Comparison. 2022. Verkkoaineisto. Ten Pixel Studio. <<https://www.linkedin.com/pulse/admob-ads-vs-unity-comparison-ten-pixel-studio/>>. Julkaistu 6.7.2022. Luettu 30.3.2024.

Audience Network Update. 2016. Verkkoaineisto. Facebook. <<https://www.facebook.com/business/news/audience-network-update>>. Julkaistu 2016. Luettu 1.4.2024.

Carr, Sam. 2020. Tier 1, 2, 3 Countries List. Understanding Country Tiers In Digital Advertising. Verkkoaineisto. Lunio. <<https://lunio.ai/blog/strategy/country-tiers-digital-advertising/>>. Julkaistu 29.8.2020. Luettu 1.4.2024.

Case Clicker. Versio 2.0.0. 2015. Hawk Games.

Clash of Clans. Versio 16.137.13. 2024. Supercell.

CPM vs eCPM – Confused? We Got Your Covered. 2023. Verkkoaineisto. Adnimation. <<https://www.adnimation.com/cpm-vs-ecpm/>>. Julkaistu 2023. Luettu 12.4.2024.

Determine a bid strategy based on your goals. Verkkoaineisto. Google. <<https://support.google.com/google-ads/answer/2459326>>. Luettu 29.3.2024.

Elliz, Trang. 2023. What are the differences between Unity Ads and AdMob? Verkkoaineisto. Google Adx. <<https://www.linkedin.com/pulse/what-differences-between-unity-ads-admob-trang-elliz/>>. Julkaistu 3.8.2023. Luettu 1.4.2024.

Fuddsworth. 2017. Unity Ads – My First Mobile Game, Experience, Revenue. Verkkoaineisto. <https://www.reddit.com/r/gamedev/comments/5mxrta/unity_ads_my_first_mobile_game_experience_revenue>. Päivitetty 28.1.2023. Luettu 29.3.2024.

Hill Climb Racing. Versio 1.61. 2024. Fingersoft.

Lara. 2016. How do I earn revenue from Unity Ads? Verkkoaineisto. Unity Technologies. <<https://web.archive.org/web/20200925072005/https://support.unity3d.com/hc/en-us/articles/205263819-How-do-I-earn-revenue-from-Unity-Ads->>. Päivitetty 2023. Luettu 30.3.2024.

Malekos, Nick. 2023. Country tier list for Advertising: Tier 1, Tier 2, and Tier 3. Verkkoaineisto. Marketing Experts Hub. <<https://marketingexpertshub.com/marketing/country-tier-list-advertising/>>. Julkaistu 22.7.2023. Luettu 1.4.2024.

Monetization Strategy. Verkkoaineisto. Unity Technologies. <<https://docs.unity.com/ads/en-us/manual/MonetizationStrategy>>. Luettu 30.3.2024.

Kovalenko, Irina. 2023. Mobile Advertising Network: How to Choose and Which One is the Best One? Verkkoaineisto. Smartyads. <<https://smartyads.com/blog/top-mobile-ad-networks>>. Julkaistu 23.6.2023. Luettu 1.4.2024.

Kovalenko, Irina. 2023. Verkkoaineisto. The Power of Interstitial Ad Network: How to Monetize Its Potential and Yield Ad Revenue. <<https://smartyads.com/blog/interstitial-ad-networks>>. Julkaistu 10.11.2023. Luettu 29.3.2024.

Revenue and payment. Verkkoaineisto. Unity Technologies. <<https://docs.unity3d.com/Packages/com.unity.ads@3.3/manual/Monetization-ResourcesRevenueAndPayment.html>>. Luettu 29.3.2024.

Rewarded ads best practices. Verkkoaineisto. Unity Technologies. <<https://docs.unity3d.com/Packages/com.unity.ads@3.3/manual/Monetization-ResourcesBestPracticesAds.html>>. Luettu 29.3.2024.

Salomäki, Timo. 2020. IT-sanasto. Verkkoaineisto. <<https://github.com/TimoSalomaki/IT-sanasto/tree/master>>. Päivitetty 10.1.2020. Luettu 15.3.2024.

Shakal, Aksana. 2024. Tier 1 countries. Verkkoaineisto. RichAds Blog. <<https://richads.com/blog/what-is/tier-1-countries/>>. Julkaistu 2024. Luettu 30.3.2024.

Streit, Bryan. 2022. Ads as your asset: Improving the player experience. Verkkoaineisto. <<https://blog.unity.com/games/improve-player-retention-with-in-game-ads>>. Julkaistu 16.1.2024. Luettu 30.3.2024.

Understanding Tier 1, 2 & 3 countries. Verkkoaineisto. Pushground. <<https://www.pushground.com/blog/understanding-tier-1-2-3-countries>>. Luettu 1.4.2024.

Unity Ads (Publishers). Verkkoaineisto. Unity Technologies. <<https://support.unity.com/hc/en-us/sections/201163835-Unity-Ads-Publishers>>. Luettu 22.3.2024.

Unity Ads Now Offers Access to Meta Audience Network Through Unity Mediation. 2019. Verkkoaineisto. Business Wire. <<https://www.business-wire.com/news/home/20211216005238/en/>>. Julkaistu 16.12.2021. Luettu 1.4.2024.

Unity Cloud. Verkkoaineisto. Unity Technologies. <<https://cloud.unity.com>>.

Unity Gaming Report. 2022. Verkkoaineisto. Unity Technologies. <<https://create.unity.com/gaming-report-2022>>. Luettu 30.3.2024.

Unity Mediation. Verkkoaineisto. Unity Technologies. <<https://unity.com/products/mediation>>. Luettu 30.3.2024.

Unity Ads SDK API reference. Verkkoaineisto. Unity Technologies. <<https://docs.unity.com/ads/en-us/manual/UnityAPI/>>. Luettu 10.3.2024.

Uvarova, Katya. Top 15 Mobile App Ad Networks and Platforms. Verkkoaineisto. <<https://messapps.com/allcategories/top-15-mobile-app-ad-networks-and-platforms/>>. Luettu 1.4.2024.

Waterfalls. Verkkoaineisto. Unity Technologies. <<https://docs.unity.com/monetization-dashboard/en-us/manual/Waterfalls>>. Luettu 30.3.2024.

Websites using Google AdSense. 2024. Verkkoaineisto. Built With. <<https://trends.builtwith.com/websitelist/Google-AdSense>>. Luettu 1.4.2024.

Suomennettu Unity SDK API-rajapinnan ohjelmakoodin viiteluettelo (Unity Ads SDK API reference).

Luokka	Funktio	Käyttötarkoitus	Huomioitavaa
Advertisement	<code>public static void Initialize(string gameId, bool testMode, IUnityAdsInitializationListener initializationListener)</code>	Alusta mainokset	
	<code>public static void Load (string adUnitId, IUnityAdsLoadListener loadListener)</code>	Lataa mainokset	
	<code>public static void Show(string adUnitId, ShowOptions showOptions, IUnityAdsShowListener showListener)</code>	Näytä mainokset	
Banner	<code>public static void Load(string adUnitId, BannerLoadOptions options)</code>	Lataa bannerimainokset	
	<code>public static void Show(string adUnitId, BannerOptions options)</code>	Näytä bannerimainokset	
	<code>public static void Hide(bool destroy = false)</code>	Piilota bannerimainokset	
	<code>public void SetPosition (BannerPosition bannerPosition)</code>	Määritä bannerin lokaatio	
ShowOptions	<code>public ShowResult resultCallback { get; set; }</code>	Kertoo mainoksen tuloksen	Vanhentunut. Käytä IUnityAdsShowListener
	<code>public string gamerSid { get; set; }</code>	Määritä tunniste tietylle käyttäjälle pelissä	
BannerLoadOptions	<code>public LoadCallback loadCallback { get; set; }</code>	Ilmoittaa kun bannerimainos lataa onnistuneesti materiaalin	
	<code>public ErrorCallback errorCallback { get; set; }</code>	Ilmoittaa kun mainos kohtaa ongelman latauksessa	
BannerOptions	<code>public BannerCallback bannerCallback { get; set; }</code>	Ilmoittaa kun käyttäjä näkee bannerimainoksen	
	<code>public BannerCallback hideCallback { get; set; }</code>	Ilmoittaa kun käyttäjältä piiloitetaan bannerimainos	
	<code>public BannerCallback clickCallback { get; set; }</code>	Ilmoittaa kun käyttäjä klikkaa bannerimainoksesta	

Lista	Arvot
PlacementState	Failed; Skipped; Finished
ShowResult	Failed; Skipped; Finished
UnityAdsInitializationError	UNKNOWN; INTERNAL_ERROR; INVALID_ARGUMENT; AD_BLOCKER_DETECTED
UnityAdsLoadError	INITIALIZE_FAILED; INTERNAL_ERROR; INVALID_ARGUMENT; NO_FILL; TIMEOUT; UNKNOWN
UnityAdsShowCompletionState	SKIPPED; COMPLETED; UNKNOWN
UnityAdsShowError	NOT_INITIALIZED; NOT_READY; VIDEO_PLAYER_ERROR; INVALID_ARGUMENT; NO_CONNECTION; ALREADY_SHOWING; INTERNAL_ERROR; UNKNOWN
BannerPosition	TOP_LEFT; TOP_CENTER; TOP_RIGHT; BOTTOM_LEFT; BOTTOM_CENTER; BOTTOM_RIGHT; CENTER

Käyttötarkoitus	Huomioitavaa
Kertoo mainoksen tilan	Vanhentunut. Käytä IUnityAdsShowListener
Kertoo miten käyttäjällä meni mainoksen kanssa	
Syyt SDK:n alustuksen epäonnistumiseen	
Syyt mainoksen lataamisen epäonnistumiseen	
Syyt mainosten päättymiseen	
Syyt mainoksen näyttämisen epäonnistumiseen	
Bannerimainoksen sijainti	

Rajapinnat	Funktio
IUnityAdsInitializationListener	<pre>public interface IUnityAdsInitializationListener { void OnInitializationComplete(); void OnInitializationFailed(UnityAdsInitializationError error, string message); }</pre>
IUnityAdsLoadListener	<pre>public interface IUnityAdsLoadListener { void OnUnityAdsAdLoaded(string adUnitId); void OnUnityAdsFailedToLoad(string adUnitId, UnityAdsLoadError error, string message); }</pre>
IUnityAdsShowListener	<pre>public interface IUnityAdsShowListener { void OnUnityAdsShowFailure(string adUnitId, UnityAdsShowError error, string message); void OnUnityAdsShowStart(string adUnitId); void OnUnityAdsShowClick(string adUnitId); void OnUnityAdsShowComplete(string adUnitId, UnityAdsShowCompletionState showCompletionState); }</pre>

Tilan ja kuvan säilyvyyden vuoksi käyttötarkotuksen ja huomiotavan taulukot ovat siirretty rajapinta taulukon alle.

Käyttötarkoitus	Huomioitavaa
OnInitializationComplete käy läpi logiikan kun SDK on alustettu.	Aktivoituu esimerkissä kun mainos lataa
OnInitializationFailed käy läpi logiikan jos SDK alustus epäonnistuu	
OnUnityAdsAdLoaded kertoo kun mainokset ovat ladattu.	
OnUnityAdsFailedToLoad kertoo jos mainokset epäonnistuivat latautumaan.	
OnUnityAdsShowFailure kertoo jos mainos on epäonnistunut.	Palautusarvot näkyvät Lista osuudessa
OnUnityAdsShowStart kertoo kun mainosten näyttö alkaa.	
OnUnityAdsShowClick kertoo jos käyttäjä painaa mainoksesta.	
OnUnityAdsShowComplete kertoo kun mainos on suoritettu loppuun.	

Mainostamisen maiden tasoluokitukset (suomennettu) (Malekos, 2023).

Taso 1		
Alankomaat	Itävalta	Suomi
Australia	Kanada	Sveitsi
Belgia	Luxemburg	Tanska
Espanja	Norja	Uusi-Seelanti
Irlanti	Ranska	Yhdysvallat
Iso-Britannia	Ruotsi	
Italia	Saksa	

Taso 2		
Andorra	Japani	Portugali
Argentiina	Kazakstan	Puerto Rico
Bahama	Kiina	Puola
Bolivia	Kolumbia	Qatar
Bosnia ja Hertsegovina	Korean tasavalta (etelä)	Romania
Brasilia	Kreikka	Saudi-Arabia
Brunei	Kroatia	Serbia
Bulgaria	Kypros	Singapore
Chile	Latvia	Slovakia
Costa Rica	Liettua	Slovenia
Dominikaaninen tasavalta	Macao	Thaimaa
Ecuador	Malesia	Turkki
Egypti	Malta	Tšekin tasavalta
Etelä-Afrikka	Marokko	Ukraina
Fidži	Meksiko	Unkari
Filippiinit	Montenegro	Uruguay
Guyana	Nepal	Valko-Venäjä
Hong Kong	Oman	Vanuatu
Indonesia	Panama	Venäjän federaatio
Islanti	Paraguay	Viro
Israel	Peru	Yhdistyneet Arabiemiirikunnat

Taso 3		
Albania	Irak	Namibia
Algeria	Jamaika	Nicaragua
Angola	Jordania	Niger
Armenia	Kambodža	Nigeria
Azerbaidžan	Kamerun	Pakistan
Bahrain	Kap Verde	Sambia
Bangladesh	Kenia	Senegal
Barbados	Kirgisia	Sri Lanka
Belize	Komorit	Suriname
Benin	Kongo	Swazimaa
Botswana	Kuwait	Tadžikistan
Burkina Faso	Laos	Tansania
Burundi	Lesotho	Togo
El Salvador	Libanon	Trinidad ja Tobago
Etiopia	Madagaskar	Tunisia
Gabon	Makedonia	Turkmenistan
Georgia	Mali	Tšad
Guatemala	Mauritania	Uganda
Guinea	Mauritius	Uzbekistan
Haiti	Moldova	Vietnam
Honduras	Mongolia	
Intia	Mosambik	