

Janne Tyni

**DATANKERÄYS KÄYTTÄEN UNITYN ANALYTICS-PALVELUA**

# DATANKERÄYS KÄYTTÄEN UNITYN ANALYTICS-PALVELUA

Janne Tyni  
Opinnäytetyö  
Kevät 2022  
Tietotekniikan tutkinto-ohjelma  
Oulun ammattikorkeakoulu

## TIIVISTELMÄ

Oulun ammattikorkeakoulu  
Tietotekniikan tutkinto-ohjelma, ohjelmistokehityksen suuntautumisvaihtoehto

---

Tekijä: Janne Tyni

Opinnäytetyön nimi: Datankeräys käyttäen Unityn Analytics-palvelua

Työn ohjaaja: Pekka Alaluukas

Työn valmistumislukukausi ja -vuosi: kevät, 2022

Sivumäärä: 21

---

Opinnäytetyön tavoitteena on tehdä sovellus, joka hakee Luppakorva-pelistä kerättyä dataa käyttäen Unity Analyticsiä ja rajapintoja. Luppakorva on Outloud-puheterapiayrityksen tekemä mobiilipeli. Sovelluksen tarkoituksena on helpottaa ja nopeuttaa datankeräysprosessia.

Opinnäytetyössä tutustutaan Unity Analyticsin toimintaan, eventeihin sekä datan hakemiseen käyttäen rajapintoja. Näiden tietojen pohjalta tehtiin datankeräyssovellus, jonka ohjelmointiprosessi käydään läpi tässä työssä.

Datankeräyssovelluksen lopputulos oli onnistunut hyvin, mutta työ kesti ajallisesti toivottua pidempään, koska datankeräyksen testaus oli hidasta.

---

Asiasanat: datankeräys, peliala, pelianalytiikka, Unity Analytics

## ABSTRACT

Oulu University of Applied Sciences  
Degree Programme in Information Technology, Option of Software Development

---

Author: Janne Tyni  
Title of thesis: Data collection using Unity's Analytics service  
Supervisor: Pekka Alaluukas  
Term and year when the thesis was submitted: Spring, 2022  
Number of pages: 21

---

The objective of the thesis is to create an application, which retrieves data collected from Luppakorva-game using Unity Analytics and API. Luppakorva is a mobile game created by the speech therapy company Outloud. The purpose of the application is to make the data collection process easier and faster.

The thesis explores the basics of the Unity Analytics, events and the API used for retrieving data. The data collection application was created based on the former information, and its programming process is written in the thesis.

The result of the data collection application was a success, but the work took longer time than anticipated, since the testing for the data collection was slow.

---

Keywords: data collection, game industry, game analytics, Unity Analytics

# SISÄLLYS

TIIVISTELMÄ.....	3
ABSTRACT.....	4
1 JOHDANTO .....	6
2 UNITY ANALYTICS -ANALYTIKKAJÄRJESTELMÄ.....	7
2.1 Data Explorer .....	8
2.2 Event Manager .....	8
2.3 Segment Builder.....	9
2.4 Remote Settings.....	9
3 EVENTIEN KÄYTTÖ .....	10
3.1 Core-eventit.....	10
3.2 Standard-eventit.....	10
3.3 Custom-eventit .....	11
4 DATAN HAKEMINEN RAW DATA EXPORTISTA.....	13
5 DATANKERÄYSSOVELLUS .....	15
5.1 Luppakorva-pelin käyttötarkoitus ja haettava tutkimusdata .....	15
5.2 Datankeräysovelluksen prosessi .....	16
5.2.1 Raakadatapaketin muodostaminen Unityn palvelimelle .....	16
5.2.2 Raakadatapaketin hakeminen Unityn palvelimelta.....	17
5.2.3 Raakadatapaketin muuntaminen luettavaan formaattiin .....	17
5.3 Pohdinnat ja ongelmat.....	18
6 YHTEENVETO .....	19
LÄHTEET.....	20

# 1 JOHDANTO

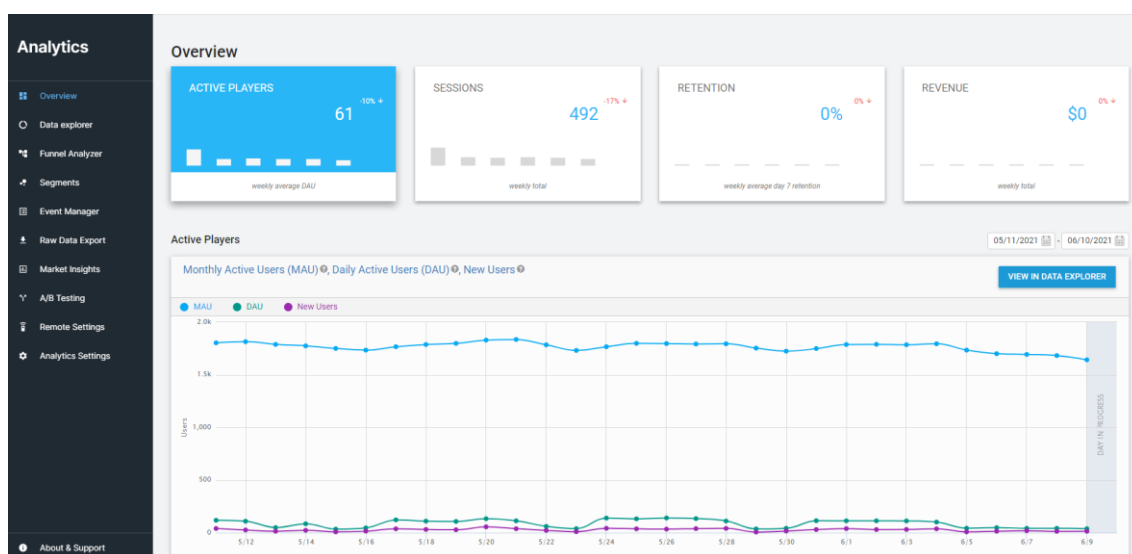
Keväällä 2021 aloitettiin opinnäytetyö Outloudin Luppakorva-projektista. Luppakorva on mobiilipeli, joka on tarkoitettu lapsille lukemisen kehittämistä varten. Pelissä käyttäjä lukee ääneen koiralle tekstiä mistä vain lähteestä, esimerkiksi kirjasta. Mitä enemmän käyttäjä lukee koiralle, sitä enemmän käyttäjän taso nousee ja eri ominaisuuksia tulee saataville. Jatkuvan ja aktiivisen lukemisen toivotaan parantavan käyttäjän lukutaitoa. Peliä voi pelata suomeksi, ruotsiksi tai englanniksi. (Outloud, linkit Sovellukset -> Lue Luppakorvalle -> Käyttöohjeet.pdf.)

Outloud on yhteistyössä kansainvälisesti yliopistojen kanssa. Yhteistyöhön kuuluu muun muassa yliopistot Iso-Britanniasta sekä Chilestä. Iso-Britannian yliopisto on pyytänyt tehdä tutkimustyötä Luppakorvasta ja sen mahdollisista vaikutuksista. Yliopiston tarkoituksena on saada sovellus koulujen luokkatiloihin, jossa oppilaille asennettaisiin Luppakorva käyttäen henkilökohtaista tunnusta, jolloin heidän sovelluksen käyttöönsä voitaisiin seurata. Tämän opinnäytetyön tarkoitus ei ole kerätä henkilötietoja käyttäjältä, vaan rakentaa yliopistolle työkalut tutkimustietoa varten.

Tarkoitus on yhdistää sovellus Unityn Analytics-palveluun, jota kautta voidaan lähettää ja kerätä dataa. Unityllä voi lähettää eventejä käyttäjien sovelluksen käytöstä suoraan Analytics-palveluun, josta ne voidaan hakea käyttämällä rajapintoja, esimerkiksi REST APIa. Saatua raakadataa tulee JSON-tiedostona, jonka voi mahdollisesti myös purkaa vaikkapa Excel-tiedostoon helpommin käsiteltäväksi.

## 2 UNITY ANALYTICS -ANALYTIKKAJÄRJESTELMÄ

Unity Analytics on Unityn luoma analytiikkajärjestelmä, jolla voidaan kerätä dataa pelien käyttäjiltä. Kerättävää dataa on muun muassa päivittäisten käyttäjien määrä, heidän käyttämänsä aika sovelluksessa sekä käyttäjien sijainti. Unity Analytics ei ole automaattisesti mukana Unity-projekteissa, vaan projektinhallitsijan on aktivoitava se päälle Unityn Editorissa. Tämän jälkeen voidaan siirtyä selaimella verkkosivulle, joka sisältää projektin hallintapaneelin. Ennen hallintapaneeliin pääsyä (kuva 1) on kirjauduttava sisään Unity-tilillä. (Unity Technologies 2020a.)

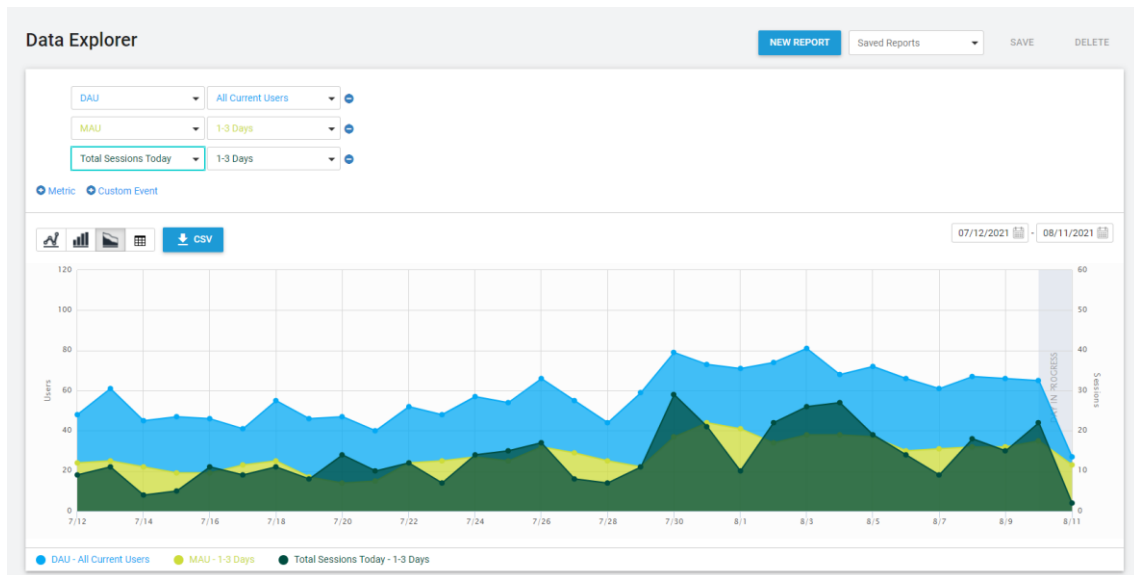


KUVA 1. Hallintapaneelin päänäkymä Unityn verkkosivuilla

Unity Analyticsin hallintapaneelissa on valikoima työkaluja erilaisten datojen seurantaan sekä ase-  
tuksia datan kontrollointiin. Data Explorerissa voi tarkastella kaikkea kerättyä dataa miltä vain ai-  
kaväliltä ja eri visuaalisissa muodoissa, kuten kaavioina ja taulukoina. Segment Builderissa voi  
tarkastella Unityn muodostamia kategorioita kaikkien pelin käyttäjien jaotteluun, muun muassa  
kulunut aika ensimmäisestä käytöstä, maantieteellinen sijainti ja markkinointi. Event Managerissa  
käsitellään ja tarkastellaan Custom Eventtejä, joita ohjelmoidaan Unity-projekteissa. Raw Data Ex-  
portista saadaan haettua kaikki kerätty raakadata JSON- tai TSV-muodossa. Remote Settingsissä  
voi päivittää pelin ominaisuuksia ja muokata muuttujien arvoja, eikä toimenpide vaadi uutta julkai-  
supäivistystä koko sovellukselle. (Unity Technologies 2020b.) Hallintapaneeli sisältää myös pelin  
markkinointiin liittyviä työkaluja, mutta niitä ei ole tarkoitus käydä tässä läpi.

## 2.1 Data Explorer

Data Explorerilla (kuva 2) voi tehdä raportteja, joista voi tarkastella pelin kerättyjen tietojen määrää tietyiltä aikaväleiltä. Raporttiin voi lisätä kolmea erilaista datatyyppiä: Metric, Custom Event sekä Ad Metric. Metric-dataan sisältyy Core Eventeistä kerätyt tiedot, jotka lähetetään palvelimelle automaattisesti aina käyttäjän pelatessa kyseistä peliä. Core Eventien tietoihin kuuluu muun muassa pelaajien määrä päivittäin tai kuukausittain, pelaamisen kesto ja peliin käytetty rahan määrä. Toisella datatyypillä, ”Custom Event”, voi tarkastella mukautettuja tietoja, joita peli voi lähettää uniikkien tapahtumien seurauksena. Kolmantena datatyypinä on ”Ad Metric”, jossa näkyy pelissä olevien mainoksien tiedot ja tuotot, mikäli mainokset on aktivoitu pelissä. (Unity Technologies 2020e.)



KUVA 2. Data Explorerin näkymä

## 2.2 Event Manager

Event Managerissa voi tarkastella peliin määriteltyjä Custom Eventeitä, joita pelin käyttäjä voi mahdollisesti lähettää käytön aikana. Eventin aktivoituessa pelaaja lähettää palvelimeen dataa, jonka sisältö on määritelty pelin skriptissä. Managerissa voi poistaa käytöstä tarpeettomia eventeitä sekä käynnistää niitä uudelleen. (Unity Technologies 2020f.) Lisää tietoa eventeistä on luvussa 3.

## 2.3 Segment Builder

Segment Builderissa voi analysoida käyttäjien jaottelemista kategorioittain. Käyttäjiä voi ryhmitellä muun muassa maantieteellisiin sijainteihin, aikaan ensimmäisestä käyttöpäivästä, pelaajien kaukallisiin ostoksiin sekä pelaajien käyttöliittymiin. Unity luo automaattisesti näitä kategorioita tarkasteltavaksi, mutta projektinhallitsijalla on mahdollisuus luoda uniikkeja kategorioita. Mikäli peliin on lisätty Custom Eventejä, voi niitä lisätä Segmentin vaatimukseksi. Segmenttiin voi lisätä useampia vaatimuksia käyttäen aiemmin mainittuja ryhmityksiä, esimerkkinä suomalaiset käyttäjät, jotka aloittivat pelin ensimmäistä kertaa alle viikko sitten. On mahdollista valita, halutaanko kaikkien vaatimusten täytyvän (AND), minkä vaan niistä (OR) tai ei mikään niistä (NONE). Uniikkeja segmenttejä on myös mahdollista analysoida aiemmin mainitulla Data Explorerilla. (Unity Technologies 2020g.)

## 2.4 Remote Settings

Remote Settings on työkalu, jolla voi vaihtaa pelin muuttujien arvoja ilman, että käyttäjällä tarvitsee päivittää sovellusversiota. Esimerkiksi pelin vaikeustasoa voi tarvittaessa pienentää nopealla muutoksella, mikä säästäisi pelin päivittämisen vaivan kaikilta käyttäjiltä. Muutoksia voi tehdä myös erilaisille käyttäjäryhmille käyttäen aiemmin mainittuja Segmentejä. Mikäli käyttäjä kuuluu useampaan kuin yhteen Segmenttiin, ristiriitaisuuksia voi esiintyä eri arvoilla. Tätä varten Segmentit voidaan luokitella tärkeysjärjestykseen, jolloin käyttäjää saa parhaiten vastaavat arvot. Kun käyttäjä käynnistää pelin, se lataa uudet arvot suoraan palvelimelta, joten käyttäjä kuitenkin tarvitsee nettiyhteyden arvojen päivittämiseen. Koodipuolella muuttujien arvot asetetaan hakemalla Unityn palvelimelta Remote Settingsien tiedot, joka onnistuu käyttämällä Unityn RemoteSettings-luokkaa. (Unity Technologies 2020h.)

### 3 EVENTIEN KÄYTTÖ

Kun sovellusta tai peliä käytetään, Unity kerää analyttistä dataa suorittamalla eventejä. Peli lähettää eventit Unityn palvelimeen, jossa ne siirretään projektin Analytics-kojelaudalle tarkasteltavaksi. Eventejä käytetään, kun halutaan tallentaa joitain käyttäjän tekemiä tapahtumia. Eventejä voidaan jakaa seuraaviin kategorioihin: Core Eventit, Standard Eventit, Custom Eventit sekä Transaction Eventit. (Unity Technologies 2020c.) Viimeisenä mainittuja transaction eli kaupallisia eventejä ei käydä läpi tässä osiossa.

#### 3.1 Core-eventit

Core-eventit ovat Unity Analyticsin automaattisia ja laiteperäisiä eventejä, joilla saadaan käyttäjän perustiedot. Core-eventien avulla saadaan luotua perusta Analytics-järjestelmän useisiin datapisteisiin, kuten päivittäisten ja kuukausittaisten käyttäjien määrä, pelisessioiden määrät sekä käytetty aika sovelluksessa. AppStart, AppRunning sekä DeviceInfo ovat kolme erilaista core-eventiä, joita voidaan tarkastella. (Unity Technologies 2020i.)

Kun käyttäjä asentaa ja käynnistää sovelluksen ensimmäistä kertaa tai käyttäjän laitteen tiedot muuttuvat, lähetetään DeviceInfo-eventi. DeviceInfoon kuuluu lukuisia laitetietoja, kuten käyttöjärjestelmä, laitteen malli, prosessorityyppi, järjestelmän muistin määrä ja monia muita. Jokaisen uuden pelisession alussa lähetetään AppStart-eventi, josta saadaan selville aikaleima, käyttöjärjestelmä, sovelluksen versio ja useita muita. Lisäksi on AppRunning-eventi, joka lähetetään ajoittain sovelluksen ollessa päällä. AppRunning on lähes identtinen AppStartin kanssa, mutta se sisältää myös session keston.

#### 3.2 Standard-eventit

Standard-eventeillä voidaan seurata käyttäjäkokemusta viidellä eri osa-alueella: sovellus- (Application), etenemis- (Progression), perehdytys- (Onboarding), sitoumus- (Engagement) sekä kaupallistamistapahtumilla (Monetization). Nämä standard eventit, lukuunottamatta sovellus-eventejä, on tarkoitettu pääosin pelin sisäiseen seurantaan. (Unity Technologies 2020j.)

Sovellus-eventeillä voidaan seurata käyttäjien vuorovaikutusta pelin ulkopuolella. Screen\_visit eventti kertoo käyttäjän käyneen jossain käyttöliittymän ikkunoissa, esimerkiksi asetusvalikossa. Cutscene\_start ja cutscene\_skip-eventit liittyvät elokuvamaisiin kohtauksiin, mikäli niitä on sovelluksessa.

Etenemis-eventeillä seurataan pelaajien edistymistä pelissä. Perinteiset game\_start ja game\_over ovat pelin aloittamisen ja lopettamisen eventit. Mikäli pelissä on tasoja, voidaan käyttää level\_start- ja level\_complete-eventejä niiden seurantaan.

Perehdytys-eventit ovat ensimmäisen käyttökokemuksen seuranta varten. First\_interaction eventti voidaan käyttää, kun pelaaja on tehnyt minkä tahansa vuorovaikutuksen ensimmäisen käynnistyksen jälkeen. Jos pelissä on tutoriaaleja eli pelin johdatuskursseja, voidaan käyttää tutorial\_start ja tutorial\_complete eventejä, joista selviää, että pelaaja on aloittanut ja suorittanut loppuun tutorialin.

Sitoumus-eventeillä saadaan selville, kuinka pelaajan toimet vaikuttavat muuhun ympäristöön ja kuinka sitoutunut pelaaja on itse peliin. Näihin eventeihin kuuluu muun muassa achievement\_unlocked sekä achievement\_step, jotka liittyvät pelinsisäisiin vaihtoehtoisin tavoitteisiin. Mikäli pelissä voi keskustella muiden pelaajien kanssa, voi käyttää chat\_msg\_sent eventtiä keskustelun aktiivisuuden seurantaan. Jos pelaaja voi yhdistää peliin sosiaalisella verkostolla, voidaan käyttää user\_signup eventtiä, sekä mikäli sosiaalisen verkoston kautta voi jakaa ja hyväksyä peliin liittyviä asioita, voidaan ne tallentaa social\_share sekä social\_accept eventeillä.

Kaupallistamis-eventeillä voidaan seurata käyttäjien muodostamia tuloja pelille sekä pelinsisäisen talouden kulkua. Esimerkiksi iap\_transaction eventti kertoo pelaajan tehneen sovelluksessa ostoksen oikealla rahalla ja ad\_start sekä ad\_complete ovat mainoksien katseluun liittyviä eventejä.

### **3.3 Custom-eventit**

Mikäli pelaajan toimintoihin ei löydy eventtiä, jota löytyisi Core- tai Standard-eventipaketeista, voidaan sille määrittää oma custom-eventi. Custom eventtiä tulisi käyttää vain silloin, kun sille ei löydy sopivaa vastinetta standard-eventeistä, sillä standard-eventit ovat enemmän määriteltyjä, ja avulla Unity voi tuottaa parempaa analyysiä. Custom-eventejä voi lähettää ohjelmoimalla ne sovelluksen

koodiin, käyttämällä Unityn AnalyticsEvent.Custom luokkaa. (Unity Technologies 2020k.) Esi-  
merkki custom-eventistä:

```
public void ReportSecretFound(int secretID){  
    AnalyticsEvent.Custom("secret_found", new Dictionary<string, object>  
    {  
        {"secret_id", secretID },  
        {"time_elapsed", Time.timeSinceLevelLoad }  
    });  
}
```

## 4 DATAN HAKEMINEN RAW DATA EXPORTISTA

Yksi Unity Analyticsin ominaisuuksista on Raw Data Export, jonka kautta voidaan hakea kaikki sovellukseen kerättyjen tietojen raakadatat. Datan voi hakea kahdella eri tavalla. Projektinhallitsija voi käyttää joko Unity Analyticsin kojelautaa verkkoselaimessa datan hakemiseen tai REST APIa, jolla datan voi hakea käyttämällä HTTP-kutsuja. (Unity Technologies 2020d.)

Kojelauta antaa mahdollisuuden hakea raakadatat ilman koodin kirjoittamisen tarvetta. Ensin valitaan datapaketti, joka halutaan tuoda palvelimelta. Mahdollisia datapaketteja ovat muun muassa appRunning, jolla saa pelin käytön kestonmäärät, appStart, jolla saa aloitettujen sessioiden kellonajat, deviceInfo, jolla saa ylimääräisiä tietoja käyttäjän laitteesta, sekä custom, johon voi määritellä custom-eventejä. Datapaketin valinnan jälkeen voidaan määritellä, miltä aikaväliltä kyseiset datat haetaan. Lopuksi valitaan, halutaanko datat JSON- vai TSV-tiedostomuodossa. Kun Unity on suorittanut haun loppuun, voidaan datatiedosto ladata kojelaudan Activity-osiosta.

Raakadata voidaan myös hakea REST APIlla, joka on ohjelmointirajapinta HTTP-protokollan käyttämiseen. API-kutsujen tekeminen vaatii Unity Project ID:n sekä API-avaimen. Raw Data Exportin tekemiseen voidaan käyttää seuraavaa cURL-kutsua komentokehotteessa:

```
curl --user {UNITY_PROJECT_ID}:{API_KEY} --request POST  
--header "Content-Type: application/json" --data {REQUEST_JSON}  
https://analytics.cloud.unity3d.com/api/v2/projects/{UNITY\_PROJECT\_ID}/rawdataexports
```

Yllä olevassa esimerkissä laitetaan UNITY\_PROJECT\_ID:n tilalle Unity-projektin ID-tunnus ja API\_KEY:n tilalle API-avain. Tekstin REQUEST\_JSON tilalle määritellään haettujen tietojen aikaväli käyttämällä startDate- ja endDate-muuttujia, haluttu tiedostomuoto format muuttujalla sekä datapaketti dataset muuttujalla. Kun pyyntö on lähetetty palvelimelle, Unity alkaa muodostamaan datatiedostoa määritetyillä hakuehdoilla sekä palvelin vastaa HTTP-pyyntöön lähettämällä lisätietoa hausta. Esimerkki palvelimen vastauksesta:

```
"id": "247b513b-147c-4a9b-a6f0-a90fdc998d4c",
"upid": "b80219c0-f779-43a0-aafa-617786c387b3",
"status": "running",
"createdAt": "2021-08-30T12:27:22.577+0000",
"duration": 0,
"request": {
  "startDate": "2021-04-01T00:00:00.000+0000",
  "endDate": "2021-04-21T00:00:00.000+0000",
  "format": "json",
  "dataset": "appRunning"
```

Kun palvelin on luonut datatiedoston, sen hakemiseen voidaan käyttää datatiedostolle luotua ID:tä, joka on tullut vastauksena yllä olevassa esimerkissä. Seuraavaksi tehdään cURL-kutsu käyttäen GET-komentoa, johon laitetaan halutun datatiedoston ID-tunnus:

```
curl --user {UNITY_PROJECT_ID}:${API_KEY} https://analytics.cloud.unity3d.com/api/v2/projects/{UNITY_PROJECT_ID}/rawdataexports/${ID}
```

Vastauksen mukana saadaan URL-osoite, jota kautta datatiedosto saadaan tallennettua lokaalisti laitteelle.

## 5 DATANKERÄYSSOVELLUS

Tämän opinnäytetyön tavoitteena oli tehdä Outloudille sovellus, joka mahdollisti Luppakorva-mobiilipelin käyttäjien datanhaun. Pelialusta Unity kerää automaattisesti pelaajien dataa palvelimelle, mutta kyseisen datan hakeminen palvelimesta manuaalisesti sekä haetun raakadatan jaottelu ja selvittely on vaivalloista. Tätä varten tehtiin kätevä ja helppokäyttöinen sovellus, jonka avulla tutkittavaa dataa saadaan haettua helpommin. Tässä opinnäytetyössä ei käsitellä henkilökohtaisia tietoja, vaan tehdään työkalu, jolla toinen osapuoli voi hakea tutkimustietoa pelin käytöstä.

### 5.1 Luppakorva-pelin käyttötarkoitus ja haettava tutkimusdata

Luppakorva (kuva 3) on Outloudin tekemä, lapsille suunnattu mobiilipeli, jossa luetaan ääneen virtuaaliselle koiralle ja saadaan pisteitä, joilla voidaan parantaa koiran elinympäristöä. Pelissä on tarkoitus lukea koiralle ääneen suoraan kirjasta, minkä tavoitteena on kasvattaa lapsen lukutaitoa. Jatkuva ja aktiivinen lukeminen tuo pisteitä ja lisää ominaisuuksia peliin, mikä voi kannustaa lukemaan lisää.



KUVA 3. Luppakorva-pelin päänäkökymä

Tutkimusdatan kohteena on käyttäjien päivittäinen peliaika sekä lukemiseen käytetty aika. Lisäksi pelin aloitus- ja lopetusajat dokumentoidaan. Kyseisestä datasta muodostetaan Excel-taulukko, josta voidaan selvittää käyttäjien peliaktiivisuus. Käyttäjät ovat jaoteltu uniikkien ID-tunnuksien perustella, mutta mitään henkilötietoja ei käsitellä. ID-tunnus on käyttäjän mobiilipuhelimen automaattisesti generoima GUID-luku.

## **5.2 Datankeräyssovelluksen prosessi**

Sovellus tehtiin käyttäen ohjelmankehitysympäristönä Visual Studiota sekä ohjelmointikielenä C#:a. Sovelluksessa käytetään myös RestSharp-kirjastoa, joka on HTTP-protokollaan perustuva ohjelmointirajapinta. RestSharpilla voidaan lähettää HTTP-pyyntöjä ja käsitellä verkkosivun lähettämää dataa.

### **5.2.1 Raakadatapaketin muodostaminen Unityn palvelimelle**

Jotta dataa voidaan hakea Unityn palvelimelta, tarvitaan oikea verkko-osoite, joka sisältää ID-tunnuksen kyseisen projektin Raw Data Export-sivulle. Lisäksi tarvitaan autentikaatio, jolla varmistetaan pääsy tekemään datapyyntöä. Tässä tapauksessa autentikaatio varmistetaan HTTP Basicilla, joka on generoitu 64-merkkinen avainkoodi projektinhallitsijan tunnuksesta ja salasanasta.

Ensimmäisenä on määriteltävä aikaväli, jolta kerätty data halutaan hakea. Unity on rajannut aikavälin maksimissaan 31 päivään. Lisäksi määritellään, halutaanko data JSON- vai TSV-formaatissa. Aikaväli määritellään sovellusta käyttäessä, mutta dataformaatti on jo määritelty koodissa, joka on aina JSON-formaatissa. Lopuksi määritellään datasetti, joka vastaa aiemmin mainittuja eventikategorioita. Tässä sovelluksessa haettava data perustuu custom-eventeihin.

Kun kaikki parametrit ovat määriteltynä HTTP-pyyntöön, voidaan se lähettää POST-komennolla. Mikäli pyyntö onnistuu, Unity aloittaa prosessin raakadatapaketin muodostamiseen ja vastaa pyyntöön datalla, joka sisältää kyseisen raakadatapaketin ID-tunnuksen.

## 5.2.2 Raakadatapaketin hakeminen Unityn palvelimelta

Vastaanotetun raakadatapaketin ID-tunnus lisätään HTTP-linkin loppuun, jotta oikeaa dataa voidaan seurata. Tekemällä HTTP-pyyntöjä tähän url-osoitteeseen saadaan Unityn palvelimelta vastauksena tietoa datapaketin statuksesta. Mikäli Unity ei ole vielä muodostanut datapakettia, se palauttaa running-statusen. Tämän aikana datankeräyssovellus odottaa ja lähettää pyynnön uudelleen viiden sekunnin välein. Kun Unity on valmistanut datapaketin, se palauttaa completed-statusen sekä url-osoitteen, josta datapaketti voidaan ladata lokaalisesti. Datankeräyssovellus käyttää WebClient-kirjastoa, jonka avulla se yhdistää saatuun url-osoitteeseen ja ohjaa ladattavan datapaketin määrättyyn kansioon.

## 5.2.3 Raakadatapaketin muuntaminen luettavaan formaattiin

Raakadatapaketti on pakattu GZ-formaattiin, joten se on purettava tekstitiedostoksi. Datapaketti voi olla jaoteltu useampiin GZ-tiedostoihin, riippuen haetun datan aikavälistä. Datankeräys-sovellus etsii kaikki GZ-tiedostot määrätystä kansioista ja purkaa ne samaan kansioon. Purettuja tiedostoja voi käytännössä jo lukea ja tutkia, mutta ne ovat sujuvampaa siirtää Excel-taulukkaan.

Datankeräys-sovellus käyttää .NET API:n DataTable-luokkaa taulukon muodostamiseen datapaketeista. Datapaketit sisältävät tietoja pelaajien jokaisesta pelisessioista. Käsiteltävät dataelementit tähän Luppakorva-tutkimukseen ovat

- UserID: uniikki ID-tunnus, jonka puhelin generoi käyttäjälle ensimmäisellä pelikerralla
- startDay: Pelisession aloituspäivämäärä
- startTime: Pelisession aloituskellonaika
- endTime: Pelisession lopetuskellonaika
- duration: Pelisession ajallinen kesto
- readTime: Koiralle lukemiseen käytetty aika pelisessiossa
- wordsTime: Keskimääräinen lukemisen aika, joka sisältää oikeita sanoja.

Prosessin alussa määriteltiin, että datapaketit muodostetaan JSON-formaatissa. Datankeräys-sovellus käyttää JsonSerializer-luokkaa jaskaakseen JSON-tiedoston osiin. Taulukkoon muodostetaan uusi rivi, jonka jälkeen dataelementit lisätään siihen. Kun kaikki datapaketit ja niiden rivit on jaoteltu ja lisätty, on taulukko valmis Excel-tiedostoksi. Ohjelma käy läpi kaikki taulukon rivit sekä

erottaa dataelementit pilkulla. Lopuksi muodostunut teksti siirretään Luppakorva-kansioon Excelin csv-formaatissa (kuva 4).

	A	B	C	D	E	F	G	H
1	userid	startDay	startTime	endTime	duration	readTime	wordsTime	
2	e11533	22/12/2021	08:18:40	08:19:26	00:00:46	00:00:00		
3	e11533	22/12/2021	08:19:33	08:38:38	00:19:05	00:17:49	00:11:37	
4	e11533	22/12/2021	08:38:42	08:59:06	00:20:24	00:17:06	00:14:18	
5	9718ab	17/12/2021	08:43:17	08:43:36	00:00:19	00:00:00		
6	29f93a	20/12/2021	09:38:17	09:42:19	00:04:02	00:04:54	00:04:22	
7	29f93a	20/12/2021	09:36:31	09:36:37	00:00:06	00:00:00		
8	9718ab	20/12/2021	15:59:04	16:00:01	00:00:57	00:01:04	00:00:54	
9	29f93a	20/12/2021	07:26:37	07:26:40	00:00:03	00:00:00		
10	29f93a	20/12/2021	11:27:04	11:47:53	00:20:49	00:32:19	00:30:36	
11	29f93a	20/12/2021	10:29:07	10:29:34	00:00:27	00:00:00		
12	29f93a	20/12/2021	11:09:42	11:09:57	00:00:15	00:00:00		
13	9718ab	17/12/2021	08:43:40	08:44:11	00:00:31	00:00:19	00:00:13	
14	29f93a	20/12/2021	11:10:04	11:26:59	00:16:55	00:31:48	00:28:06	
15	29f93a	20/12/2021	11:02:30	11:02:37	00:00:07	00:00:00		
16	9718ab	20/12/2021	15:58:04	15:58:12	00:00:08	00:00:00		
17	9718ab	20/12/2021	15:58:50	15:59:01	00:00:11	00:00:00		
18	29f93a	20/12/2021	09:37:09	09:38:10	00:01:01	00:00:00		
19	9718ab	20/12/2021	15:58:37	15:58:44	00:00:07	00:00:00		
20	29f93a	20/12/2021	09:48:30	09:51:09	00:02:39	00:00:00		
21	29f93a	20/12/2021	09:42:24	09:48:28	00:06:04	00:07:39	00:07:16	
22	29f93a	20/12/2021	11:01:42	11:02:27	00:00:45	00:00:00		
23	e11533	21/12/2021	12:31:51	12:35:24	00:03:33	00:02:11	00:01:49	
24	e11533	21/12/2021	12:41:09	12:43:12	00:02:03	00:00:56	00:00:32	
25	e11533	21/12/2021	12:35:24	12:35:25	00:00:01	00:00:00		
26	e11533	21/12/2021	12:31:26	12:31:49	00:00:23	00:00:00		
27	e11533	21/12/2021	12:31:03	12:31:21	00:00:18	00:00:00		
28	e11533	21/12/2021	12:40:57	12:41:03	00:00:06	00:00:00		
29	e11533	21/12/2021	14:10:32	14:10:55	00:00:23	00:00:00		
30								

KUVA 4. Esimerkki muodostuneesta Excel-taulukosta

### 5.3 Pohdinnat ja ongelmat

Projektin alussa oli kaksi vaihtoehtoa sovelluksen päätoiminnalle: joko tehdä Windows-sovellus, joka sisältäisi käyttöliittymää, tai pelkästään komentokehoteelta ajettava. Lopulta päätös oli tehdä komentokehoteelta ajettava, sillä koodissa ilmeni ongelmia RestClientin ja Windows-kirjastojen yhdistämisen kanssa. Muihin ongelmiin kuului muun muassa sovelluksen jakaminen ja toiminta eri laitteilla. Tarkoituksena oli, että sovelluksen jakamiseen riittää pelkästään suoritettava tiedosto (.exe), mutta joidenkin kirjastojen kanssa ilmeni ongelmia, jonka seurauksena kaikki projektin tiedostot jaettiin, jotta toiminta onnistuisi eri laitteella.

Projektin valmistumisessa tuli viivästyksiä, sillä dataan liittyviä ongelmia ilmeni Unityn palvelimella. Datapaketit saapuivat Unityn palvelimelle joskus myöhässä, useammista päivistä jopa kahteen viikkoon. Tämä vaikutti tutkimusdatan testaamiseen ja pitkitti sovelluksen lopullista viimeistelyä.

## 6 YHTEENVETO

Unity Analytics on yksi Unityn työkaluista, joka on tarkoitettu pelien käytön datankeruuseen sekä kerätyn datan analysointiin. Kerättyä dataa voi hakea Raw Data Exportilla, johon määritellään datan haettava aikaväli, datapaketin formaatti sekä eventit-datasetti, joka on joko yksi core-eventeistä tai custom-eventi. Core- ja standard-eventit ovat Unityn omia eventejä, joita voidaan käyttää datan lähettämiseen, mutta custom-eventit määritellään pelin koodissa.

Tässä projektissa tehtiin Outloud-yritykselle sovellus, joka mahdollisti Unityn datan hakemisen vattomasti sekä muodosti raakadatan Excel-taulukon. Outloud kerää Luppakorva-pelistään tutkimusdataa Iso-Britanniassa olevaan yliopistoon, joka on yhteistyössä Outloudin kanssa. Tutkittavaan dataan kuuluu pelaajien päivittäinen aktiivisuus sekä pelissä käytetty luku-aika.

Projekti on konsolisovellus, joka on työkalu tarvittavan datan hakemiseen suoraan lokaalisesti laitteelle. Sovellukseen ei tarvitse syöttää muuta kuin haettavan datan aikaväli, jonka jälkeen sovellus yhdistää Outloudin Unity-tunnuksilla Luppakorva-projektiin ja tekee Unitylle pyynnön aloittaa prosessi datapaketin muodostamiseen. Kun datapaketti on muodostettu, sovellus lataa sen lokaalisesti käyttäjän laitteelle. Datapaketit ovat GZ-formaatissa, jotka sovellus purkaa tekstitiedostoksi. Koska kyseisten tekstitiedostojen datat ovat JSON-formaatissa, sovellus jakaa ne luettaviin osiin. Viimeisenä vaiheena datasta muodostetaan Excel-taulukko ja prosessi on edennyt loppuun.

Yleisesti projekti onnistui hyvin, mutta ajallisesti se kesti pitkään. Itse sovelluksen ohjelmoinnissa ei ollut isoja vastoinkäymisiä, mutta suuri osa ajasta meni tutkimusdatan testaamiseen. Sovelluksen teko opetti lisää tietoa C#-kielen ohjelmoinnista.

## LÄHTEET

Outloud. Hakupäivä 9.3.2022. <https://www.outloud.fi/>

Unity Technologies 2020a. Legacy Analytics. Unity Manual, version: 2020.3. Unity Documentation. Hakupäivä 9.3.2022. <https://docs.unity3d.com/Manual/UnityAnalytics.html>

Unity Technologies 2020b. Legacy Analytics Dashboard. Unity Manual, version: 2020.3. Unity Documentation. Hakupäivä 9.3.2022. <https://docs.unity3d.com/Manual/UnityAnalyticsDashboard.html>

Unity Technologies 2020c. Legacy Analytics Events. Unity Manual, version: 2020.3. Unity Documentation. Hakupäivä 9.3.2022. <https://docs.unity3d.com/Manual/UnityAnalyticsEvents.html>

Unity Technologies 2020d. Unity Analytics Raw Data Export. Unity Manual, version: 2020.3. Unity Documentation. Hakupäivä 9.3.2022. <https://docs.unity3d.com/Manual/UnityAnalyticsRawDataExport.html>

Unity Technologies 2020e. Unity Analytics Data Explorer. Unity Manual, version 2020.3. Unity Documentation. Hakupäivä 9.3.2022. <https://docs.unity3d.com/Manual/UnityAnalyticsDataExplorer.html>

Unity Technologies 2020f. Unity Analytics Event Manager. Unity Manual, version 2020.3. Unity Documentation. Hakupäivä 9.3.2022. <https://docs.unity3d.com/Manual/UnityAnalyticsDashboardEventManager.html>

Unity Technologies 2020g. Unity Analytics Segment Builder. Unity Manual, version 2020.3. Unity Documentation. Hakupäivä 9.3.2022. <https://docs.unity3d.com/Manual/UnityAnalyticsSegmentBuilder.html>

Unity Technologies 2020h. Unity Analytics Remote Settings. Unity manual, version 2020.3. Unity Documentation. Hakupäivä 9.3.2022. <https://docs.unity3d.com/Manual/UnityAnalyticsDashboardRemoteSettings.html>

Unity Technologies 2020i. Unity Analytics Core Events. Unity manual, version 2020.3. Unity Documentation. Hakupäivä 9.3.2022. <https://docs.unity3d.com/Manual/UnityAnalyticsCoreEvents.html>

Unity Technologies 2020j. Unity Analytics Standard Events. Unity manual, version 2020.3. Unity Documentation. Hakupäivä 9.3.2022. <https://docs.unity3d.com/Manual/UnityAnalyticsStandardEvents.html>

Unity Technologies 2020k. Unity Analytics Custom Events. Unity manual, version 2020.3. Unity Documentation. Hakupäivä 9.3.2022. <https://docs.unity3d.com/Manual/UnityAnalyticsCustomEvents.html>