



Michael Lock

# Sosiaalisen median elementtejä ja teknologioita hyödyntävä lautapelisovellus

Metropolia Ammattikorkeakoulu

Insinööri (AMK)

Tieto- ja viestintätekniikan tutkinto-ohjelma

Insinöörityö

27.10.2022

# Tiivistelmä

Tekijä:	Michael Lock
Otsikko:	Sosiaalisen median elementtejä ja teknologioita hyödyntävä lautapeli-sovellus
Sivumäärä:	54 sivua
Aika:	27.10.2022
Tutkinto:	Insinööri (AMK)
Tutkinto-ohjelma:	Tieto- ja viestintätekniikka
Ammatillinen pääaine:	Mobiilisovellukset
Ohjaajat:	Lehtori Ulla Sederlöf

---

Insinööriyden tarkoituksena oli tehdä Flutter-ohjelmistokehityksen avulla sosiaalisen median sovellus, joka on tarkoitettu käytettäväksi Android-puhelimella. Insinööriydessä perehdyttiin siihen, mikä ohjelmistokehitys sopii tietyn sovelluksen kehittämiseen.

Työssä tutkittiin, mikä on käytetyin mobiiliohjelmistokehitys sovelluksien tekemiseen ja käytetyin kehittäjien keskuudessa. Suosituin mobiiliohjelmistokehitys on Flutter, jota käytettiin vuonna 2021 eniten mobiilisovelluksien tekemiseen. Työssä tutkittiin myös Apache Cordovaa sen vuoksi, että työssä ei haluttu verrata kahta suosituinta, vaan erilaisia mobiiliohjelmistokehityksiä. Muiden ohjelmistokehitysten osalta käytetyimmät olivat Ionic, Xamarin, Unity, NativeScript, PhoneGap ja Kotlin Multiplatform. Sen lisäksi tutkittiin eri sosiaalisen median alustoja ja ohjelmistorajapintojen suosiota ja käyttöastetta BuiltWith-työkalulla ja Stackoverflow-tageilla.

Lopputuloksena tuotettiin lautapeli-aiheinen Tinder-kloonin, jota voidaan jatkokehittää ja julkaista sovelluskauppaan, jonka kautta lautapeliharrastajat voivat sitä hyödyntää. Sovellusta testattiin käyttäjillä. Käyttäjätestauksen aikana havaittiin ongelmia käyttäjäprofiilin muokkauksessa, jossa käyttäjät eivät osanneet löytää haluamaansa lautapeliä, ja pelikategorian valitseminen koettiin turhaksi. Käyttäjät onnistuivat löytämään ja lähettämään pareille viestejä. Tulosten perusteella selvitettiin, että jatkokehityksen kannalta täytyy parantaa sovelluksen käyttökokemusta, saavutettavuutta, toiminnallisuuksia ja ulkoasua.

Avainsanat: ohjelmistokehitys, sosiaalinen media, mobiilikehitys

## Abstract

Author: Michael Lock  
Title: Board game application using social media elements and technologies  
Number of Pages: 54 pages  
Date: 27 October 2022

Degree: Bachelor of Engineering  
Degree Programme: Information and Communications Technology  
Professional Major: Mobile Solutions  
Supervisors: Ulla Sederlöf, Senior Lecturer

---

The purpose of the thesis was to use the Flutter software framework to create a social media application that is intended to be used on Android phones. Thesis involved familiarizing with which mobile application framework is suitable for developing a specific application.

This thesis studied the most used mobile software framework for making applications and the most used among developers. The most popular mobile software framework is Flutter, which was most used in 2021 to make mobile applications. Apache Cordova was also studied in the work because the work did not want to compare two popular ones, but different mobile software frameworks. Regarding other software frameworks, the most used were Ionic, Xamarin, Unity, NativeScript, PhoneGap and Kotlin Multiplatform. In addition, different social media platforms and the popularity and usage of the APIs were studied using the BuiltWith tool and Stackoverflow tags.

The result was a Tinder clone with a board game theme, which can be further developed and published on the Google Play app store that board game hobbyists can use. The application was tested with users. During user testing, issues were found in editing the user profile, where users could not find the board game they wanted and choosing a game category was felt to be pointless. The users managed to find matches and send them messages. Based on the results, it was determined that in terms of further development, the application's user experience, accessibility, functionality and appearance must be improved.

Keywords: framework, social media, mobile app development

# Sisällys

1	Johdanto	1
2	Sosiaalinen media	2
2.1	Sosiaalinen media terminä ja käytännössä	2
2.2	Sosiaalisen median historia	3
2.3	Sosiaalisen median käyttötarkoitukset ja edut	5
2.4	Sosiaalisen median ongelmat	8
3	Sosiaalisen median teknologiat	10
3.1	TikTok	10
3.2	Instagram	11
3.3	Twitter	12
3.4	Ohjelmistorajapintojen käyttöaste ja suosio	14
4	Mobiilikehitys	17
5	Lautapelisovellus	20
5.1	Suunnittelu	22
5.2	Automaatio ja työkalut	25
5.3	Palaute	44
6	Yhteenveto	45
	Lähteet	47

## 1 Johdanto

Insinööriyössä perehdytään sosiaalisen mediaan ja siihen, mitä eri sosiaalisen median sovellukset hyödyntävät. Nykyään sovelluksia on valtavasti ja useimmat niistä keskittyvät yhteen tai useampaan palveluun, kuten kuvien jakeluun ja mikrobloggaukseen. Uudet teknologiat ovat mahdollistaneet edullisemmän ja nopeamman tavan kehittää sovelluksia. Työn tarkoituksena on tutkia, mitä sosiaalista mediaa sovellukset hyödyntävät, ja vertailla eri mobiilikehityksen ohjelmistokehyksiä. Opinnäytetyön tekijä on kiinnostunut sosiaalisen median elementeistä ja teknologioista, ja tarkoitus on hyödyntää näitä ajatuksia teknologisissa ratkaisuissa ja tutkia, miten ne toimivat.

Insinööriyöraportti on jaettu kolmeen osaan. Ensimmäinen osa, luku 2, käsittelee sosiaalista mediaa, sen historiaa sekä hyötyjä ja haittoja. Osassa kootaan tietoa siitä, miten sosiaalisen median elementtejä voidaan hyödyntää ja miten oppimista voidaan vahvistaa niitä käyttämällä.

Toisessa osassa, luvussa 3, tarkastellaan, mitä teknologioita eri sosiaalisen median alustat hyödyntävät, niiden ohjelmistorajapintojen suosiota kehittäjien keskuudessa ja sitä, mitä niistä käytetään eniten eri sivustoilla.

Kolmannessa osassa, luvussa 4, vertaillaan ohjelmistokehityksen suosiota ja eroja mobiilikehityksessä. Osan tarkoituksena on muodostaa käsitys siitä, mikä ohjelmistokehitys sopii sovelluksen käyttötarkoitukseen, kun lähdetään suunnittelemaan sovellusta. Insinööriyötä varten kehitetään mobiilisovellus, jossa haetaan lautapeliseuraa sen yhteisöstä. Sovelluksessa käytetään ohjelmistorajapintoja ja sosiaalisen median ominaisuuksia, kuten käyttäjäprofiilin muokkausta ja pikaviestin lähettämistä. Projekti toimii esimerkkinä siitä, miten sosiaalisen median elementtejä hyödynnetään mobiilisovelluksen suunnittelussa ja kehittämisessä.

Sovellus tehtiin Metropolian opinnäytetyötä varten, mutta se toimii tämän jälkeen oman osaamisen esittelynä ja tarkoitus on jatkokehittää ja ylläpitää sitä aktiivisesti. Työn aiheeseen päädyttiin, koska tekijä on kiinnostunut lautapeleistä ja kiinnostunut luomaan uudenlaisen tavan etsiä lautapeliseuraa.

## **2 Sosiaalinen media**

### **2.1 Sosiaalinen media terminä ja käytännössä**

Sosiaalisesta mediasta ei ole tarkkaa määritelmää. Nykyään sosiaalisella medialla voidaan tarkoittaa verkkoviestintäympäristöjä, jotka mahdollistavat sisällön demokratisoinnin ja antavat sisällönkuluttajalle mahdollisuuden nousta sisällöntuottajaksi. Sosiaalisella medialla on kyky saavuttaa valtavaa skaalautuvuutta reaaliajassa. Nämä verkkoteknologiat antavat ihmisille mahdollisuuden olla yhteydessä toisiinsa luodakseen arvoa verkkokeskustelun ja yhteistyön avulla. [1, s. 5.]

Jonathan A. Obarin ja Steve Wildmanin vuonna 2015 julkaistussa tutkimusartikkelissa tutkittiin sosiaalisen median määritelmiä kirjallisuudessa ja tunnistettiin neljä yhteistä ominaisuutta tämänhetkisille sosiaalisen median palveluille:

1. Sosiaalinen media on verkkopohjaisia web 2.0 -sovelluksia.
2. Käyttäjän luoma sisältö on sosiaalisen median elinehto.
3. Yksilöt ja ryhmät luovat käyttäjäkohtaisia profiileja sivustolle tai sovellukseen, jota sosiaalisen median palvelu suunnittelee ja ylläpitää.
4. Sosiaalinen media helpottaa verkkoyhteisöjen kehittymistä yhdistämällä käyttäjän profiiliin muiden käyttäjien tai ryhmien profiiliin.

[2, s. 2.]

Yhdysvaltalaisen Merriam-Websterin sanakirjassa määritellään sosiaalinen media sähköisen viestinnän muodoiksi (kuten sosiaalisen verkostoitumisen ja mikroblogin verkkosivustot), joiden kautta käyttäjät luovat verkkoyhteisöjä jakaakseen tietoja, ideoita, henkilökohtaisia viestejä ja muuta sisältöä (kuten videoita ja valokuvia). Monimuotoiset, kehittyvät ja sisäänrakennetut sosiaalisen median palvelut tekevät niiden määrittämisestä haastavaa. [3; 2, s. 6.]

Markkinoinnin ja sosiaalisen median asiantuntijat ovat yleisesti sitä mieltä, että sosiaalinen media sisältää seuraavat 13 tyyppiä:

- blogit (esim. Huffington Post, Boingboing)
- ammatilliset verkkoyhteisöpalvelut (esim. LinkedIn, XING)
- yhteistyöhankkeet (esim. Wikipedia, Mozilla)
- yrityksen verkkoyhteisöpalvelut (esim. Yammer, Socialcast)
- keskustelupalstat (esim. Gaia Online, IGN Boards)
- mikroblogit (esim. Twitter, Tumblr)
- internetkuvagalleriat (esim. Flickr, Photobucket)
- arvostelusivustot (esim. Amazon, Elance)
- verkkokirjanmerkit (esim. Delicious, Pinterest)
- verkkovideopelit (esim. World of Warcraft, Mafia Wars)
- verkkoyhteisöpalvelut (esim. Facebook, Google+)
- videopalvelut (esim. YouTube, Vimeo)
- virtuaalimaailmat (esim. Second Life, Twinity).

[4, s. 4.]

## 2.2 Sosiaalisen median historia

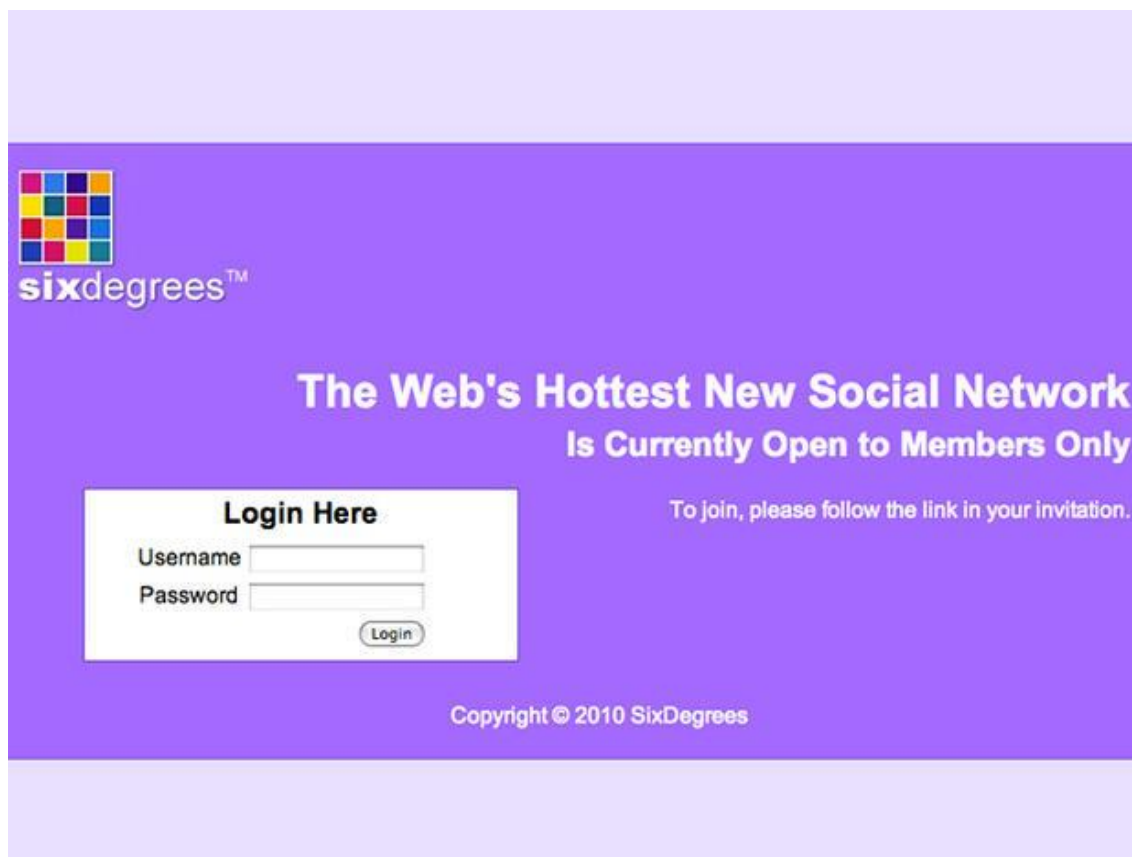
Sosiaalinen media alkoi lähes kaksi vuosisataa sitten, 24. toukokuuta vuonna 1844, kun Samuel F. B. Morse lähetti ensimmäisen kerran viestin lennätinlinjalla Washingtonista Baltimoreen. Tuhansien vuosien ajan viestejä olivat rajoittaneet nopeus ja etäisyys, joilla pystyttiin matkustamaan ja näkemään signaaleja, kuten lippuja tai savua. Morsen keksintö kaupallistettiin nopeasti ja hyödynnettiin yhdysvaltalaisessa maataloudessa ulkomaisten hintojen ja luottojen uutisointiin. [5, s. 22; 6.]

Internetin juuret juontavat ARPANET-nimisestä (Advanced Research Projects Agency Network) yhdysvaltalaisesta tietokoneverkosta, jota alettiin kehittää Sputnikin laukaisun yhteydessä vuonna 1957. Yhdysvaltain hallitus perusti ARPANETin tavoitellakseen johtoasemaa teknologiassa ja tieteessä painottaen sotilaallisia sovelluksia. Nimi vaihdettiin muotoon DARPA (Defense Advanced Research Project Agency) ja myöhemmin vaihdettiin takaisin alkuperäiseen nimeen. Kun ARPA oli vakiinnuttamassa asemaansa, oli useita keskeisiä tutkimuksia pakettivälitteisessä tiedonsiirrossa ja tietokoneviestinnässä. Tutkimukset ja yritykset antoivat perustan, joka mahdollisti hajautettujen pakettiverkkojen käyttöönoton. Internetin pioneeri J. C. R. Lickider Massachusetts Institute of Technologyn teknillisestä korkeakoulusta kirjoitti vuonna 1962 sarjan esseitä, joissa hän visioi maailmanlaajuisesta tietokoneverkostosta, joka antaisi helpon pääsyn tietokoneen tietoihin ja ohjelmiin mistä tahansa sivustosta. Yleisesti ottaen tämä ajatus ei eroa paljon nykyajan internetistä. [7, s. 28.]

Nykyistä verkkoviestintäympäristöä muistuttavat palvelut syntyivät 1960-luvulla Illinoisin yliopistossa, jossa kehitettiin PLATO-niminen tietokonejärjestelmä, joka tarjosi sosiaalisen median varhaisia ominaisuuksia vuoden 1973 innovaatioilla. Nämä innovaatiot olivat keskustelupalsta (Pad), Notesfiles, joka oli edeltäjä uutisryhmille, chat-huone (Talkomatic), pikaviestintä (TERM-talk), joukkorahoitettu verkkolehti ja blogi (News Report) sekä käyttöoikeuksien rajoittaminen omistajan muistiinpanotiedostoon tai muuhun sovellukseen (Access Lists). [8.]

Six Degrees -palvelua pidetään laajalti ensimmäisenä sosiaalisen verkostoitumisen sivustona (kuva 1). Andrew Weinreich perusti yrityksen toukokuussa vuonna 1996, ja verkkopalvelu julkaistiin seuraavana vuonna. Se sisälsi tyypillisiä sosiaalisen median elementtejä, kuten käyttäjäprofileja ja kaverilistoja. Palvelu mahdollisti yhteyksien luomisen käyttäjien kesken. Huiman suosion vuoksi verkkopalvelu houkutteli 3,5 miljoonaa käyttäjää. Kysyntä ei vastannut senhetkisen teknologian tasoa kasvavan käyttäjämäärän ylläpitämisessä, minkä takia verkkopalvelun toiminta lakkautettiin. [9; 10.]





Kuva 1. Six Degrees -palvelun etusivu [9].

### 2.3 Sosiaalisen median käyttötarkoitukset ja edut

UGT-teoria, joka on englanniksi "Uses and gratifications theory", on yleisökeskeinen lähestymistapa joukkoviestinnän ymmärtämiseen. Eron muihin mediavaikutusteorioihin, jotka kysyvät "Mitä media tekee ihmisille?", UGT keskittyy siihen "Mitä ihmiset tekevät medialla?" Viestintätutkija Thomas E. Ruggieron on tutkimuksissaan todennut, että uuden median nousu tekee UGT-teoriasta tärkeämmän kuin koskaan. Tämä näkökulma on erityisen hyödyllinen selitettäessä, miksi ihmiset omaksuvat uusia medioita. Nykyään UGT on edelleen yksi yleisimmistä mediavaikutusten tutkimuksessa käytetyistä teorioista. [11.]

UGT-teorian kulmakivi on, että yleisö on keskeisessä roolissa valittaessaan mediaa, jota se käyttää. Lisäksi yleisö on tietoinen, miksi se haluaa kuluttaa mediaa ja käyttää niitä tietoisesti tehdäkseen valintoja, jotka täyttävät sen tarpeita ja toiveita. UGT määrittää viisi oletusta median kulutuksesta:

1. Median käyttö on motivoitunutta ja tavoitteellista. Ihmisillä on aina syy kuluttaa mediaa, vaikka se olisi vain tapa tai viihdyttävää.
2. Ihmiset valitsevat median sen perusteella, että se täyttää tietyt toiveet ja tarpeet.
3. Median käyttöä ohjaavat yksilölliset sosiaaliset ja psykologiset tekijät.
4. Media kilpailee muiden viestintämuotojen, kuten sanallisen viestinnän ja yleisöviestinnän kanssa. Nykyään suuri osa mediakulutuksesta tapahtuu mobiililaitteella ja viestintämuodot, kuten tekstiviestit, sovellukset ja sosiaaliset verkostot, kilpailevat jatkuvasti yleisön huomiosta.
5. Koska ihmiset ovat aktiivisia median käyttäjiä, mediaviestit eivät vaikuta ihmisiin erityisen voimakkaasti.

[11.]

Suuri osa UGT-tutkimuksesta keskittyy tyytyväisyyteen, jota media tuottaa tai sen pitäisi synnyttää. Tämä on johtanut erilaisiin typologioihin, jotka luokittelevat tyydytyksen tiiviisiin luokkiin. Vuonna 1973 Elihu Katz, Michael Gurevitch ja Hadassah Haas loivat tunnetun järjestelmän viidestä sosiaalisesta ja psykologisesta tarpeesta, joita median käyttö tyydyttää, mukaan lukien

1. tiedon hankkiminen ja ymmärryksen parantaminen
2. esteettisten tai emotionaalisten kokemusten saaminen
3. aseman, uskottavuuden ja luottamuksen vahvistaminen

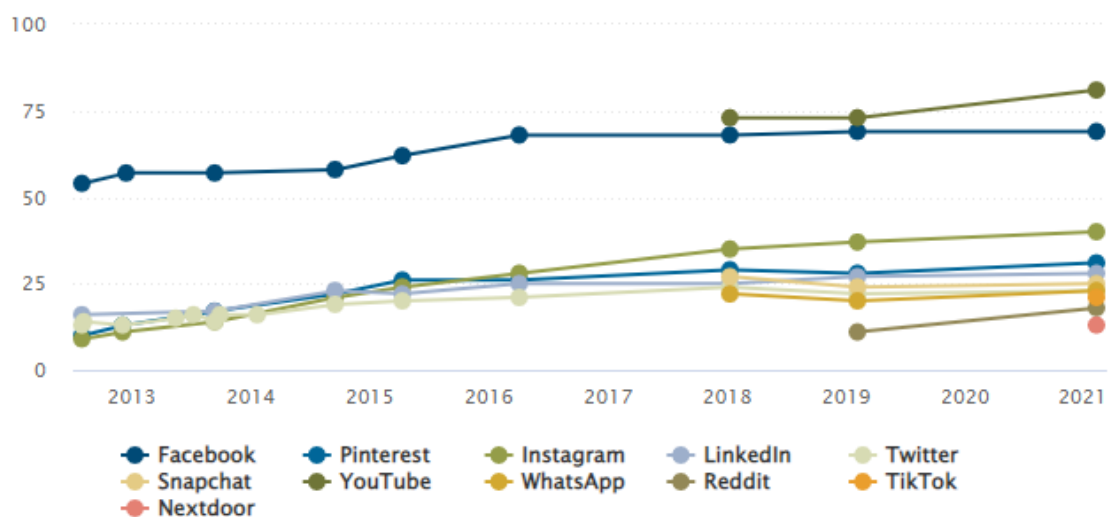
4. suhteiden vahvistaminen ystäviin ja perheeseen

5. rentoutuminen.

[11.]

Vuonna 2021 noin seitsemän kymmenestä yhdysvaltalaisesta käytti sosiaalista mediaa pitääkseen yhteyttä ystäviin ja perheeseen ja pysyäkseen ajan tasalla uutisista ja tapahtumista. YouTube ja Facebook olivat eniten käytettyjä verkkoalustoja, ja niiden käyttäjäkunta edusti laajimmin koko väestöä. Pienempi osa väestöä käytti Twitteriä, Pinterestiä, Instagramia ja LinkedIniä (kuva 2). [12.]

*% of U.S. adults who say they ever use ...*



Note: Respondents who did not give an answer are not shown.

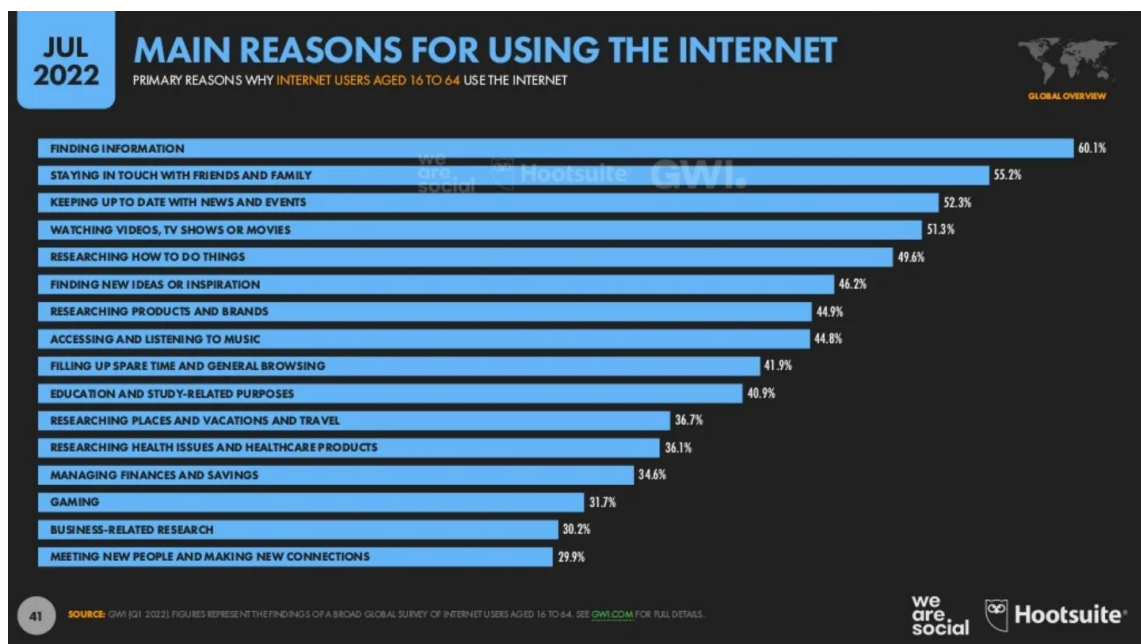
Source: Surveys of U.S. adults conducted 2012-2021.

PEW RESEARCH CENTER

Kuva 2. Pew-tutkimuskeskuksen vuonna 2021 tekemän tutkimuksen tulokset yleisimmistä sosiaalisen median alustoista [12].

DataReportalin vuonna 2022 julkaistu maailmanlaajuinen Global Statshot -tutkimus yhteistyössä We Are Socialin ja Hootsuiten kanssa osoittaa, että

edelleen on vahvaa kasvua digitaalisessa käyttöotossa ja aktiivisuudessa ympäri maailmaa. Sadat miljoonat ihmiset käyttivät sosiaalista mediaa ensimmäistä kertaa vuoden aikana. Internetin ja sosiaalisen median käyttäjistä 60 % etsii uutta tietoa, 55 % piti yhteyttä perheeseen ja ystäviin ja 52 % pysyi ajan tasalla uutisista ja tapahtumista. Muita pääasiallisia syitä olivat suoratoistopalveluiden käyttäminen, sen tutkiminen, miten joitakin asioita tehdään, ja uusien ideoiden tai inspiraation löytäminen. (Kuva 3.) [13.]



Kuva 3. GlobalWebIndexin neljännesvuosiraportti Q1/2022:n luvut 16—64-vuotiaiden internetin käytön tutkimusten tuloksista [13].

## 2.4 Sosiaalisen median ongelmat

Anonymiteettia on, kun jonkun puhe, kirjoitukset tai toimet eivät sisällä henkilökohtaisia tunnistetietoja ja siksi niitä ei voi yhdistää henkilökohtaiseen identiteettiin. Anonymiteetti ei vastaa pseudonyymiä tai aliaista, jossa henkilön toimet voidaan yhdistää tiettyyn henkilöllisyyteen, joka eroaa hänen todellisesta henkilöllisyydestään. Esimerkiksi jotkut julkaisijat Redditissä kehittävät hyvin rikkaita identiteettejä yhteisössä, mutta näitä identiteettejä ei voida yhdistää heidän todelliseen henkilöllisyyteensä. Self-disclosure eli omat paljastukset eroavat myös anonymiteetistä. Joku saattaa esimerkiksi kirjoittaa omasta

treffielämästään blogia, joka sisältää paljon paljastuksia, koska hän saattaa kertoa treffien intiimeistä yksityiskohdista. Blogia voi kuitenkin kirjoittaa nimettömänä ilman sellaista tietoa, joka tekisi siitä henkilökohtaisesti tunnistettavan. Tutkijat ovat havainneet, että mitä enemmän erittelevää anonymiteettiä blogilla on, sitä todennäköisemmin se sisältää itsensä paljastamista. [14, s. 90; 75.]

Anonymiteetti verkossa voi johtaa antisosiaaliseen käyttäytymiseen, kuten 4chan-kuvafoorumilla. Kuvafoorumien viestit ovat oletukseltaan anonyymejä, joten graafista, loukkaavaa ja säädyttöä sisältöä on runsaasti. Verrattuna Facebookin kaltaisiin sivustoihin sisältö on suhteellisen tavanomaista ja yleensä yhdistettävissä henkilön todelliseen henkilöllisyyteen. Philip Zimbardo oli yksi ensimmäisistä, jotka tutkivat anonymiteetin vaikutuksia ihmisen käyttäytymiseen. Osallistujia pyydettiin tekemään vihamielisiä tekoja muita kohtaan silloin, kun he olivat naamioituneita tai kun he olivat tunnistettavissa. Havaittiin, että anonyymit käyttäytyivät aggressiivisemmin kuin tavallisissa vaatteissa olevat. Havainnon vahvistamiseksi sosiaalisen median tutkimus on osoittanut, että anonyymit ovat vähemmän halukkaita yhteistyöhön toistensa kanssa. [14, s. 90.]

Yksityisyys internetissä tarkoittaa sitä, että henkilön selaushistoria, salasana ja käyttäjänimi, yksityiset viestit ja muu toiminta pysyvät suojassa uteliailta katseilta. Nämä tiedot ovat arvokkaita jäljittäjille, mainostajille ja rikollisille. Yksityisyys suojaa vallanpitäjien väärinkäytöksiltä, vaikka käyttäjä ei olisi tehnyt mitään väärää valvonnan tapahtuessa. Mainostajat seuraavat yhä enemmän ihmisten verkkokäyttämistä, ja käyttämällä kerättyjä tietoja näytetään käyttäjille yksilöllisesti kohdennettuja mainoksia. Kohdennettu mainostus on saanut paljon huomiota mainostajilta, kuluttajilta, päättäjiltä ja tutkijoilta. [15; 16; 17.]

Nykyään käyttäjät arvostavat omaa anonymiteettiä. He eivät halua, että henkilökohtaisia tietoja vuodatetaan verkkoon, mikä loukkaa omaa yksityisyyttä. Sensor Towerin tutkimus osoittaa, että anonymiteettiä hyödyntävien

sovelluksien kysyntä on kasvanut. Vuonna 2014 julkaistiin 527 mobiilisovellusta, jotka käyttivät kuvauksessa anonymiteettiä avainsanana. Sovellusten määrä oli yli kaksinkertaistunut vuoteen 2021 mennessä 1257:ään, joista 446 oli yksityisyyteen keskittyviä työkaluja ja 172 sosiaalisen median alustoja. Vuonna 2021 WhatsApp-viestintäpalveluun julkaistiin tietosuojapäivitys, jossa tiedotettiin, että jaettaisiin enemmän tietoja Facebookin kanssa parantamaan tuotteita ja mainontaa. Miljoonat käyttäjät siirtyivät Signaliin, joka tarjoaa parhaan salatun viestintäpalvelun. WhatsAppin tavoin sillä voidaan lähettää viestien lisäksi myös tiedostoja, äänimuistiinpanoja, kuvia ja videoita. Vuonna 2019 sovelluksen käyttäjiä oli vain 500 000. Käyttäjämäärä oli saavuttanut vuoteen 2021 mennessä 40 miljoonan rajan. Tämä osoittaa, että käyttäjät arvostavat yksityisyyttään ja ovat valmiita käyttämään anonyymejä sovelluksia sen varmistamiseksi. Sosiaalisen median maailma on jatkuvassa muutoksessa. Käyttäjät, jotka haluavat olla yhteydessä muihin käyttäjiin säilyttäen samalla anonymiteetin ja yksityisyyden, kääntyvät sitä tarjoavien sovellusten puoleen. [18; 19.]

### **3 Sosiaalisen median teknologiat**

#### **3.1 TikTok**

ByteDance julkaisi Kiinassa sovelluksen nimeltä Douyin syyskuussa vuonna 2016. Sovellus houkutteli vuoden aikana yli 100 miljoonaa käyttäjää Kiinassa, ja myöhemmin sovellus julkaistiin kansainvälisillä markkinoilla nimellä TikTok. [20; 21; 22.]

TikTokin avulla käyttäjät voivat luoda lyhyitä videoita, joissa on usein musiikkia taustalla ja joita voidaan nopeuttaa, hidastaa tai muokata. Käyttäjät voivat myös lisätä oman äänensä taustamusiikin päälle. Alun perin videot saivat kestää enintään vain 15 sekuntia, mutta heinäkuusta 2021 alkaen videot voivat kestää jopa kolme minuuttia. [23; 24.]

TikTokista on esitetty kritiikkiä ja kohua sen sisällöstä, joka on tulkittu eri kulttuureissa hävyttömäksi ja mauttomaksi. Useat maat, kuten Indonesia, Bangladesh, Intia ja Pakistan ovat asettaneet väliaikaisia estoja palveluun ja varoituksia palvelun sisällöstä. Vuodesta 2020 TikTok poistettiin Intian markkinoilta Kiinan ja Intian välisten jännitteiden seurauksena. [25; 26; 27; 28.]

Viranomaisten asiakirjat ovat paljastaneet, että TikTokin valvojen käsketään sensuroida poliittista sisältöä ja estää käyttäjien videoita, jotka vaikuttavat "liian rumilta, vammautuneilta tai huonoilta" [29; 30].

TikTokin sosiaalisen median puoli on sen käyttäjissä, käyttäjien seuraajissa ja käyttäjien tuottamassa sisällössä. Käyttäjät voivat lähestyä sisältötuottajaa kommentoimalla tai tykkäämällä videosisältöä sekä tutustua ja verkostoitua sisältötuottajien ja seuraajien kanssa. TikTokia hyödynnetään markkinointiin, henkilöbrändin rakentamiseen sekä internetilmiöiden luomiseen ja levittämiseen. Palvelussa keskitytään luovaan sisältöön ja pysytään trendien kärjessä. Muita hyötyjä ovat yhteyden ottaminen yleisöön, bränditietoisuuden kasvattaminen sekä tuotteiden ja palveluiden mainonta. [31.]

### 3.2 Instagram

Instagram on sosiaalisen median palvelu, jonka perustivat Kevin Systrom ja Mike Krieger vuonna 2010. Vuonna 2012 Facebook osti yhtiön miljardilla dollarilla. Palvelussa jaetaan kuvia ja videoita, joita voidaan muokata erilaisilla suodattimilla ja järjestää paikkamerkintöjen ja hashtagien eli aihetunnisteiden avulla. Käyttäjäs sisältöä voidaan kommentoida ja tykätä. Julkaisuja jaetaan julkisesti tai ennalta hyväksytyjen seuraajien kanssa. Käyttäjät voivat selata trendikästä sisältöä ja seurata muita käyttäjiä. Tykätyt kohteet ja seurattujen käyttäjien sisältö näytetään omassa syötteessä. [32.]

Kuvien ja videoiden lisäksi palvelu vahvistaa ystävyyssuhteita. Käyttäjät voivat yhdistää Instagramin Facebook-tiliinsä ja ladata samoja julkaisuja molempiin palveluihin. Instagramilla on enemmän taiteellista arvoa kuin muilla sosiaalisen

median alustoilla. Huhtikuussa vuonna 2022 enemmän kuin kaksi kolmasosaa maailmanlaajuisista Instagram-käyttäjistä oli alle 35-vuotiaita. Tämä tekee alustasta erityisen houkuttelevan markkinoijille. [33.]

Vuonna 2015 Mashable-sivusto nimesi Instagramin parhaaksi sovellukseksi kaikkien aikojen 100 parhaan iPhone-sovelluksen luettelossa. Toukokuussa 2017 Royal Society for Public Health teki kyselyn, jossa oli mukana 1479 14–24-vuotiasta, ja pyysi heitä arvioimaan sosiaalisen median alustoja ahdistuksesta, masennuksesta, yksinäisyydestä, kiusaamisesta ja kehonkuvasta riippuen. Vastauksena häiritseviin ja negatiivisiin kommentteihin käyttäjien valokuvista Instagram on antanut käyttäjille paremman hallinnan julkaisuihinsa ja niiden kommenttiosioihin. Vuodesta 2016 käyttäjät ovat pystyneet ottamaan kommentit pois käytöstä omista julkaisuistaan sekä hallitsemaan kommenttiosion kielenkäyttöä syöttämällä sanoja, joita he pitävät loukkaavina. [34; 35; 36; 37.]

Instagram on hyvä oman brändin rakentamisessa ja markkinoimisessa, sillä palvelu tarjoaa hyvän tavan näyttää käyttäjälle sisältöä. Instagramissa on monia ominaisuuksia, joita käytetään liiketoiminnan edistämiseen ja henkilöbrändin rakentamiseen. Käyttäjät voivat lisätä videoita ja kuvia Omaan tarinaan, joka katoaa vuorokauden kuluessa. Lisäksi Instagram Reels -sisältömuoto antaa käyttäjän luoda 3–30 sekunnin mittaisen videon, jonka taustalle voi lisätä musiikkia Instagramin musiikkikirjastosta. Näin voidaan osallistua trendeihin, ilmaista itseään luovasti ja valistaa muita käyttäjiä. Käyttäjät voivat selata läheisten toimintaa, joiden julkaisut tuovat käyttäjille yhteisöllisyyttä. Julkaisut antavat käyttäjille tavan ilmaista itseään muokkaamalla kuvia julkaisuvaiheen työkaluilla. Palvelussa voidaan myös viestittää WhatsAppin tavoin ja lähettää live-lähetyksiä. [38.]

### 3.3 Twitter

Twitter on yhdysvaltalainen vuonna 2006 julkaistu yhteisö- ja mikroblogipalvelu, jossa käyttäjät lähettävät ja lukevat toistensa päivityksiä. Tekstipohjaiset



viestipäivitykset eli twiitit sisältävät korkeintaan 280 merkkiä. Palvelun tarjoaa Twitter, Inc., joka perustettiin San Franciscossa, Kaliforniassa. Huhtikuussa vuonna 2022 Elon Musk teki Twitter-palvelusta ostotarjouksen, jonka johtokunta hyväksyi. [39; 40.]

Käyttäjät voivat seurata toisten twiittejä, ja yksittäiset twiitit voidaan lähettää uudelleen omille seuraajille. Uudelleenlähetettyyn twiittiin on mahdollista kirjoittaa viesti viitaten kyseisen käyttäjän twiittiin. Twiitin tykkäämisellä käyttäjä voi osoittaa pitävänsä viestin sisällöstä, ja hashtagien eli aihetunnisteiden avulla voidaan etsiä Twitteristä tietoa sekä nähdä twiittivirta kyseisellä asiasanalla. [42; 43; 41, s. 23.]

Twitteriä käytetään paljon katastrofiviestinnässä, sillä tieto leviää nopeammin sosiaalisten verkostojen kautta. Useat tunnetut henkilöt käyttävät alustaa riippumatta siitä, millä alalla he työskentelevät. Alustaa voidaan käyttää markkinointitarkoituksiin, kuten uusimman trendin, tuotteiden tai tarjouksien levittämiseen ja asiakkaiden reaktion vastaanottamiseen. Palvelussa käyttäjät ja seuraajat ovat vuorovaikutuksessa toistensa kanssa, ja varmennetut tilit luovat turvaa ja luottamusta käyttäjien kesken. [44.]

Helmikuusta 2022 lähtien Kiinan, Iranin, Myanmarin, Pohjois-Korean, Venäjän, Turkmenistanin ja Uzbekistanin hallitukset ovat kieltäneet pääsyn Twitteriin. Viime vuosien aikana Twitter on poistanut valtavan määrän propagandatilejä ja -twiittejä. Twitter poisti twiittejä liittyen Pietarin trollitehtaaseen, joka oli yrittänyt vaikuttaa julkiseen mielipiteeseen vuoden 2016 Yhdysvaltain vaalien aikana ja sen jälkeen. Kesäkuussa 2020 poistettiin myös 175 000 propagandatiliä, jotka levittivät puolueellisia twiittejä Kiinan kommunistisesta puolueesta, Yhtenäisen Venäjän puolueesta tai Turkin presidentti Erdoğanista. [45; 46.]

Twitterin tarkoitus ei ole varsinaisesti tutustua uusiin ihmisiin, vaan saada tykkäyksiä, vastauksia ja mielipiteitä twiitteihin, jotka voivat liittyä uutisiin, poliittisiin kirjoituksiin ja markkinointitarkoituksiin. Käyttäjä, jolla on tarpeeksi seuraajia, voi nostattaa kohun uudelleentwiittaamalla ja mainita käyttäjiä, joita

twiitti koskee. Jos uudelleentwiitattu viesti sisältää aihetunnisteen, uudelleentwiittaus nostaa aihetunnisteen suosiota.

### 3.4 Ohjelmistorajapintojen käyttöaste ja suosio

Insinööriyössä selvitettiin, mitä ohjelmistokehityspaketteja ja -rajapintoja Twitterin, TikTokin ja Instagramin mobiilisovelluksessa käytetään. Näin haluttiin selvittää, miten nämä sovellukset on kehitetty ja mitä rajapintoja niissä on käytetty. Tarkoituksena oli hyödyntää sosiaalisen median rajapintoja ja ohjelmistokehityspaketteja insinööriyössä. Ensin tutkittiin mobiilisovelluksen tilastoja tarjoavia verkkopalveluja, kuten 42 Matters ja Appfigures. Ne tarjoavat tietoa sovelluksen integroiduista ohjelmistokehityspaketeista, latauksista, arvosteluista ja saatavuudesta eri maissa Google Play- tai App Store -kaupassa. Päädyttiin olemaan käyttämättä näitä sivustoja, sillä ne antavat harhaanjohtavaa tietoa siitä, mitä teknologioita on käytetty sovelluksen palvelinteknologian toteuttamisessa. 42 Mattersin mukaan Instagram käyttää Google Firebase -palvelua taustajärjestelmänä, vaikka virallisen Instagram-insinöörien blogin mukaan he käyttävät Django-ohjelmistokehitystä ja PostgreSQL-tietokantaa. [47.] Työssä löydettiin toinen vaihtoehto, joka tutkisi ainoastaan sovelluksen rajapintoja. Selvitettäisiin Charles web debuggaus -ohjelmalla reaaliaikaisesti Twitterin, TikTokin ja Instagramin http-verkkoliikennettä. Ohjelmanäkymään päivitettiin reaaliajassa viimeisimpiä http-kutsuja, jotka liittyivät markkinointidatan keräämiseen, sisäänkirjautumispyynnön lähettämiseen ja käyttäjän luoman sisällön hakemiseen. Rajapintojen joukossa olivat muun muassa Twitter API ja Facebook Graph API. Lopulta päädyttiin olemaan käyttämättä sosiaalisen median rajapintoja, sillä niistä ei olisi välttämätöntä tarvetta insinööriyöprojektin versiossa. Teknisesti oli mielenkiintoista tutkia ja tietää näistä rajapinnoista ja siitä, kuinka aktiivisesti yritykset keräävät markkinointidataa käyttäjistä.

BuiltWith-työkalun avulla selvitettiin, mitkä ovat tämän hetken käytetyimmät sosiaalisen median ohjelmistorajapinnat. BuiltWithilla on

tietojenanalysointijärjestelmä, joka on yli 11 vuoden ajan kerännyt lähes 100 %:n kattavuudella aktiivisten verkkotunnusten ja aliverkkotunnusten tietoja. BuiltWith-työkalulla on suuri määrä tietoa, joka on tilastollisesti merkittävä otos siitä, mitä teknologioita internetsivut käyttävät. Tämä työkalu käyttää uusimpia liikenteen tietolähteitä varmistaakseen, että se kattaa kaikki uusimmat sivustot, jotka ovat ajan tasalla 24 tunnin sisällä. [48, s. 24.]

Jokainen verkkosivu antaa signaaleja siitä, millä teknologialla sivusto on rakennettu. Järjestelmä seuraa näitä signaaleja ja määrittää, käyttääkö sivusto tiettyä verkkotekniikkaa siitä teknologiasta. [48.]

Taulukkoon 1 on koottu yhteen, kuinka paljon eri sosiaalisen median rajapintoja käytetään internetsivuilla maailmanlaajuisesti.

Taulukko 1. Sosiaalisen median ohjelmistorajapintoja käyttävien verkkosivustojen määrä maailmanlaajuisesti vuonna 2022 [49].

<b>Ohjelmistorajapinta</b>	<b>Rajapintaa käyttävien sivustojen määrä, kpl</b>
Twitter	816 194
Facebook Graph	604 224
TikTok	347 737
Tumblr	102 227
Instagram Graph	83 003
Snapchat	1 225

Suosituin ohjelmistorajapinta on Twitter API: sitä on käytetty yli 800 000 sivustossa. Toiseksi suosituin on Facebook Graph -rajapinta, jota on käytetty yli puolessa miljoonassa sivustossa. Snapchatin rajapintaa on käytetty vähiten maailmanlaajuisesti. [49.]

Stackoverflow tags -arviot

Stackoverflow on suosittu kehittäjien keskustelusivusto, jossa voi kysyä ja saada vastauksia koodausaiheisiin kysymyksiin. Tällä sivustolla on myös

ominaisuutena tagit, joilla voi merkitä ja luokitella samankaltaisia kysymyksiä. Stackoverflow-tagit ovat mittari, kuinka suosittu jokin teknologia on kehittäjien keskuudessa. [50.]

Taulukossa 2 näkyy sosiaalisen median rajapintojen suosio kehittäjien keskuudessa. Facebook Graph -rajapinta saa eniten koodausaiheisia kysymyksiä, sillä se mahdollistaa enemmän ominaisuuksia kuin muut rajapinnat ja Facebookilla on maailmanlaajuisesti eniten aktiivisia käyttäjiä. Lisäksi Instagram Graph ja muut rajapinnat, kuten Facebook LIVE ja Facebook Marketing, on rakennettu Facebook Graphin päälle. Jokainen sovellus, joka on vuorovaikutuksessa Facebookin kanssa, käyttää Graph-sovellusliittymää tietojen hallintaan. [51.]

Opinnäytetyön kannalta haluttiin tutkia, mitä sosiaalisen median alustoja kannattaa ottaa huomioon omassa sovelluksessa, jotta saa enemmän käyttäjiä. Integroimalla rajapintoja ja luomalla jakamispainikkeet sovellukseen kannustetaan käyttäjiä sosiaaliseen jakamiseen. Sosiaalinen jakaminen on erityisen tärkeää, sillä se houkuttelee uusia käyttäjiä.

Taulukko 2. Stackoverflow-tagien määrä rajapinnalla [52; 53; 54; 55; 56; 57].

<b>Ohjelmistorajapinta</b>	<b>Stackoverflow tags -määrät</b>
Facebook Graph	34 801
Instagram Graph	2 822
Twitter	2 056
Tumblr	201
TikTok	17
Snapchat	13

## 4 Mobiilikehitys

Valtaosa yrityksistä siirtyy työpöydän kokoisista sivustoista mobiilisovelluksiin, joista on tullut olennainen osa ihmisten jokapäiväistä elämää. Älypuhelimista on tullut välttämätön osa yrityksen selviytymisessä, ja mobiili ensin - suunnitteluperiaatteeseen kiinnitetään yhä enemmän huomiota.

Mobiilisovellusmarkkinoiden koko oli 187,58 miljardia Yhdysvaltain dollaria vuonna 2021, ja sen ennustetaan kasvavan 13,4 %:n vuotuisella kasvuvauhdilla (CAGR) vuosina 2022—2030. [58; 59.]

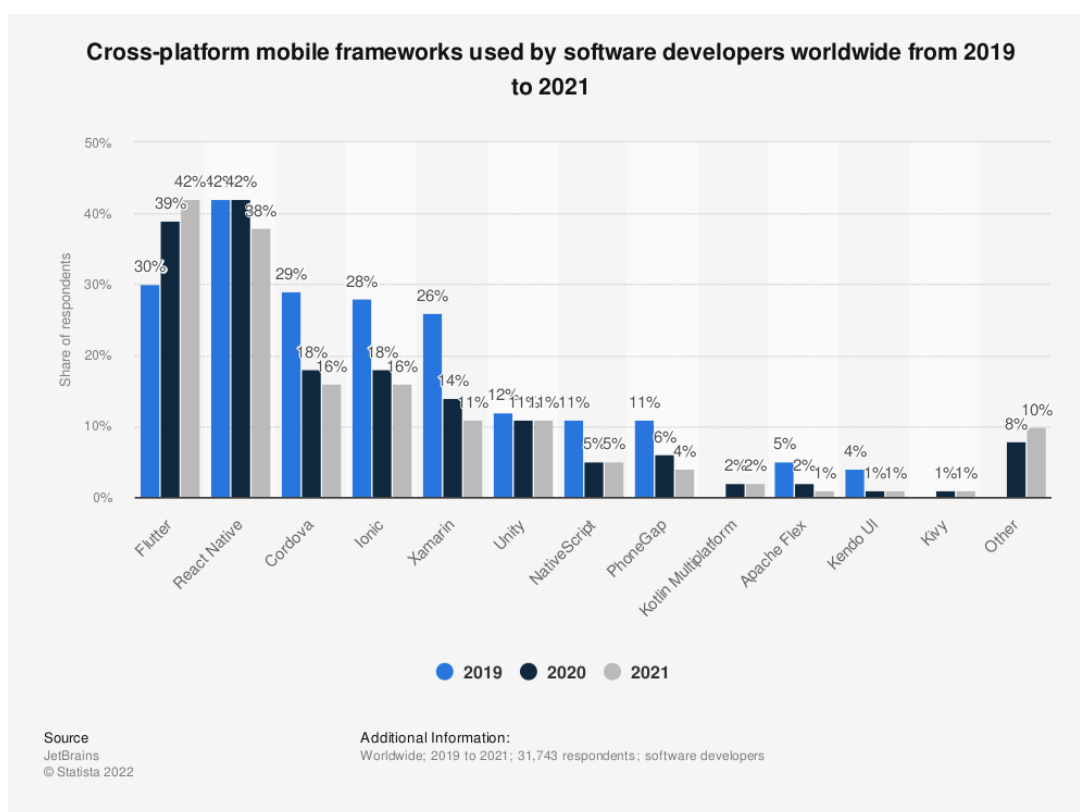
Ennen kuin lähdetään kehittämään mobiilisovellusta, valitaan sovellustyyppi. Sovellustyypit ovat natiivi-, web- ja hybridisovellukset. Jos sovelluksessa tavoitellaan Android-käyttäjiä ja tarvitaan laajempi pääsy laitteisto- ja ohjelmisto-ominaisuuksiin, natiivisovellus sopii tähän tarkoitukseen. Kehityksen suurimpiin hyötyihin kuuluvat parempi suorituskyky muihin sovellustyyppeihin ja parempi käyttäjän tietosuoja. Natiivisovellus voidaan kehittää vastaavanlaisesti myös iOS-alustalle, mutta kehitys molemmille alustoille on kuitenkin aikaa vievä ja kallis prosessi. Tyypillisessä natiivisovelluksessa voidaan hyödyntää esimerkiksi Google AR -teknologiaa ja matkalentoa tutkivia Android-laitteen antureita. [60.]

Web-sovelluksen kehitykseen käytetään tavallisia web-tekniikoita, kuten HTML5:sta, CSS3:sta ja JavaScriptiä. Sovellusta voidaan käyttää kaikissa mobiilialustoissa, ja seurauksena on rajoitettu pääsy laitteen laitteisto- ja ohjelmisto-ominaisuuksiin, mutta saavutetaan nopeampi sovelluskehitys. Web-sovelluksessa saavutetaan nopeampi sovelluskehitys, sillä kehittäjän ei tarvitse opetella alustakohtaista ohjelmointikieltä ja -ympäristöä. Sovellus kärsii huonommasta suorituskyvystä ja käyttäjäkokemuksesta alustakohtaisten sovellusliittymien puuttumisen vuoksi. Tyypillinen hybridisovellus, joka on kehitetty Cordovalla, koostuu verkkosivustonäkymästä, josta löytyy esimerkiksi omat jäsenkorttitiedot. [60.]

Hybridisovellukset säästävät kehittäjien resursseja ja aikaa luomalla alustariippumattoman sovelluksen. Sovellus on koodipohjaltaan laajempi, sillä

sitä kehitetään useammalle alustalle, minkä vuoksi sovellusta ei voida erikseen optimoida jokaiselle alustalle. Hybridisovellukset ovat joustavampia, kun kehitetään animaatioita ja käyttöliittymäkomponentteja. Suosituimpia hybridisovelluksia ovat Instagram, Discord ja Twitter. [60; 61.]

The State of Developer Ecosystem 2021 -nimisessä JetBrainsin tutkimuksessa todettiin Flutterin olevan suosituin alustariippumaton ohjelmistokehitys, jota kehittäjät käyttävät maailmanlaajuisesti. Tutkimukseen vastasi 1 743 kehittäjää, ja tutkimustuloksien perusteella 42 % ohjelmistokehittäjistä käytti Flutteria. Kokonaisuutena noin kolmasosa mobiilikehittäjistä käyttää alustariippumattomia teknologioita tai ohjelmistokehityksiä. Loput mobiilikehittäjät käyttävät natiiviyökaluja. Vuonna 2020 maailmanlaajuisesti oli 24,5 miljoonaa ohjelmistokehittäjää, ja kehittäjämäärän odotetaan kasvavan tulevina vuosina digitalisaation lisääntymisen vuoksi. (Kuva 4.) [62.]



Kuva 4. Vuonna 2021 JetBrainsin julkaisemassa tutkimuksessa verrattiin mobiilikehityksen ohjelmistokehityksien käyttösuosiota kehittäjien keskuudessa [62].

Flutter on avoimen lähdekoodin ohjelmistokehityspaketti, jonka on kehittänyt Google. Se julkaistiin toukokuussa 2017 ja on uusin tunnetuimmista ohjelmistokehyksistä. Sillä kehitetään alustariippumattomia sovelluksia verkkoon, Androidille, iOS:lle, Linuxille, Fuchsia OS:lle, Mac OS:lle ja Windowsille. Flutterin on kehitetty olevan suorituskyvyltään tehokkaampi ja käyttöliittymäkomponenteiltaan joustavampi kuin sen kilpailija, React Native. Flutter on suosituimpia ohjelmistokehyksiä ja sen Pub-paketinhallintaohjelma tarjoaa laajan valikoiman erilaisia paketteja, jotka helpottavat ohjelmointia ja tarjoavat joustavia käyttöliittymäelementtejä. Pub.dev-verkkosivuston jokaisesta ohjelmakirjastosta on saatavilla hyödyllistä tietoa, kuten readme-osio, esimerkkikoodi, asennusohjeet, suosio ja tykkäyksen määrä. Flutter-ympäristössä ohjelmoidaan Dart-ohjelmointikielellä, joka noudattaa olio-ohjelmoinnin periaatteita ja käsitteitä, kuten kapselointia, abstraktiota, periytymistä ja polymorfismia. Heinäkuussa 2020 Dartille otettiin käyttöön Null safe -ominaisuus, joka tekee koodista turvallisemman ja poistaa null-tyyppiset ohjelmointivirheet. Dartin lisäksi Flutter pohjautuu widget-komponentteihin, jotka piirtävät koko käyttöliittymän näytölle. Widgetit eli pienoishjelmat määrittävät, miltä ne näyttävät riippuen tilastaan ja asetelmastaan. [63; 64.]

React Native on JavaScript-pohjainen ohjelmistokehitys, jolla voidaan kehittää alustariippumattomia mobiilisovelluksia iOS:lle ja Androidille. Mobiilisovelluksen kehitys tapahtuu käyttäen React-kirjastoa, jonka kanssa voi halutessaan käyttää TypeScriptiä, joka tekee koodista turvallisen. CSS-tyylittely- ja HTML-merkintäkielen sijasta käytetään React Native -kirjaston käyttöliittymäkomponentteja, joita voidaan muokata. Jos kehittäjällä on verkkosivusto, joka on ohjelmoitu Reactilla, hän voi uudelleen käyttää valtaosan koodista myös mobiilisovelluksessa. Tämä on React Nativen suurin etu verrattuna muihin ohjelmistokehyksiin. [64.]

Apache Cordova on avoimen lähdekoodin ohjelmistokehitys, jolla voidaan kehittää web-sovelluksia. Cordovan avulla voidaan käyttää verkkotekniikoita alustojen väliseen kehitykseen, ja se on tarkoitettu kehittäjille, jotka haluavat laajentaa sovellusta useammalle kuin yhdelle alustalle ilman, että heidän

tarvitsee kehittää se kunkin alustan kehityskielellä. Cordovaa voidaan käyttää esimerkiksi terveyssovelluksissa, jotka hyödyntävät laitteen sensoreita, kuten GPS:ää ja kiihtyvyyssanturia. [65.]

## 5 Lautapelisovellus

Insinööriytyö sai alkunsa omasta lautapeliharrastuksesta, josta tuli idea luoda uudenlainen tapa etsiä lautapeliseuraa. Tavoitteena oli kehittää sovellus, jossa kaksi käyttäjää, joilla on samankaltaisia lautapelimieltymyksiä, yhdistetään samaan keskusteluryhmään. Käyttäjä pystyisi pyyhkäisemällä toisia käyttäjiä löytämään sopivan peliparin, jolla olisi samankaltaisia lautapelimieltymyksiä. Muita käyttäjiä voisi suodattaa ainakin etäisyyden, lautapelimieltymysten ja kielten mukaan. Lisäksi sovellukseen suunniteltiin kalenteria lautapelikerhojen ajoista ja tapahtumista ja ominaisuutta lautapeliryhmän hakemiseen. Sovellukseen olisi integroitu Googlen ja Facebookin sisään kirjautuminen ja painikkeita sosiaaliseen jakamiseen, ja käyttäjät pystyisivät vertailemaan toisten lautapelimieltymyksiä ja suosikkilautapelejä.

Tyypillisesti lautapeliseuraa voidaan etsiä oman kaveripiirin lisäksi esimerkiksi Suomen lautapeliseura -yhdistyksen keskustelupalstalta. Keskustelupalstalta löytää eri paikkakuntien ketjuja, joissa tiedotetaan tapaamispaikasta ja seuraavasta pelikerrasta. Useimmiten lautapelitoimintaa harrastetaan kirjastossa, ravintolassa tai kahvilassa. Muita tapoja löytää lautapelitoimintaa on esimerkiksi kirjoittamalla Googlen hakukoneeseen ”kirjasto lautapelikerho” tai etsimällä seuraa digitaalisista lautapeleistä, kuten Tabletop Simulator.

Olen aikaisemmin kehittänyt vapaa-ajalla sosiaalisen median periaatteisiin perustuvaa Urbaani löytöretkeily -tyyppistä sovellusta, joka oli jäänyt kesken. Sovellukseen oli tarkoitus sisällyttää keskeisiä sosiaalisen median elementtejä, kuten käyttäjän luomaa sisältöä ja verkkoyhteisöjen kehittymistä.

Otin yhteyttä Suomen lautapeliseura Ry:hyn, ja hallituksen sihteeri kiinnostui lisätoteutuksena integroida mobiilisovellukseen jäsenkorttitoiminnallisuus.



Seuralla ei ole virallista mobiilisovellusta, joten tämä tarjosi hyvän mahdollisuuden työstää sovellus seuran käyttöön.

Keräsin Google Formsin avulla palautetta lautapeleihin, sosiaalisen median ja lautapelisovelluksen käyttöön liittyvistä hyvistä ominaisuuksista. Kyselyyn vastasi 18 vastaajaa, jotka olivat pääosin Discordin Lautapelaajat-ryhmästä ja tuttavapiiristäni. (Kuva 5.)

### 3 tärkeintä ominaisuutta sosiaalisen median sovelluksessa

	1. tärkein	2. tärkein	3. tärkein
Yksinkertainen ja käyttäjäystävällinen käyttöliittymä	4	2	3
Käyttäjän tietoturvasuus	3	6	2
Visuaalisesti miellyttävä ja esteetön käyttö vammautune	0	2	2
Verkostoituminen	1	1	3
Sisällön luominen ja välittäminen	1	4	1
Viestien lähetys	7	0	2
Avoimet keskustelupalstat	1	1	1
Reaaliaikaiset ilmoitukset	0	0	0
Muokattavat tietosuoja-asetukset	1	1	1
Responsiivisuus	0	1	3

### 3 tärkeintä ominaisuutta lautapeliseuraa hakevassa mobiilisovelluksessa?

	1. tärkein	2. tärkein	3. tärkein
Lautapelikokoelmien vertailu	1	6	4
Lautapeliseurauksen etsiminen	15	2	1
Lautapelitapahtumat	1	8	5
Uusien lautapelien löytäminen	1	2	8

### Muut ominaisuudet

- Lautapelaamiseen soveltuva paikka
- Helppokäyttöisyys, integrointi esim. Boardgamegeekkiin
- Etsiä useamman hengen peliseuraa tietyllä alueella
- Testaus ja mainonta
- Pelaamisen reunaehtojen määrittely: milloin, kuinka pitkään, missä paikassa ja leppoisasti vai kilpailullisesti?
- Turvallisuuskysymykset: uskaltaako sovellusta käyttää, jos on nainen tai vähemmistön edustaja?
- Sovelluksen lokalisointi (sovelluksen kääntäminen eri kielille)
- Helpot säännöt

Kuva 5. Tulevan lautapelisovelluksen ominaisuuksista noin 20 henkilölle tehdyn kyselyn tulokset.

Kyselyn jälkeen lähetin vielä kysymyksen lautapelisovelluksen ideoista Discordin Lautapelaajat-ryhmään, jossa ehdotettiin käyttäjien lautapelimieltymysten vertailua ja ryhmäseuran hakemista. Sieltä sain idean kehittää Tinder- ja Badoon-kaltaisen sovelluksen, sillä useimmalla nuorella on kokemusta sentyyppisistä sovelluksista.

## 5.1 Suunnittelu

Mobiilisovelluksen suunnittelu alkoi palautteen tutkimisesta. Tärkeimpiä ominaisuuksia vastausten pohjalta olivat lautapeliseuran löytäminen, lautapelitapahtumat ja uusien lautapeliöiden löytäminen. Muita olennaisia palvelun osia olivat tietoturva ja hyvä käyttäjäkokemus. Toteutin sovelluksessa kolme tärkeintä ominaisuutta, jotka olivat lautapeliseuran löytäminen, keskusteleminen parien kanssa ja oman käyttäjäprofiilin muokkaus.

Valitsin Flutterin, koska muissa natiiveissa ja alustariippumattomissa ohjelmistokehyksissä alhaalta ylöspäin liukuvan ponnahdusikkunan tekeminen ei ole yhtä helppoa. Tein ponnahdusikkunat käyttäjäkortteja pinoavaan kotinäkymään ja käyttäjäprofiilin muokkaukseen, jotta käyttäjän ei tarvitse siirtyä toiseen näkymään muokatakseen tietoja ja jotta uusi informaatio päivitetäisiin käyttäjän tämänhetkiseen näkymään. Toisin kuin React Nativessa ja natiivikehityksessä, Flutterista löytyi helpot ohjeet käyttäjäkortin animaatioiden tekemiseen, jota tarvitsin sovelluksen kotinäkymässä. Chat-näkymä listaa keskusteluryhmät, jotka luodaan, kun käyttäjä onnistuu löytämään parin.

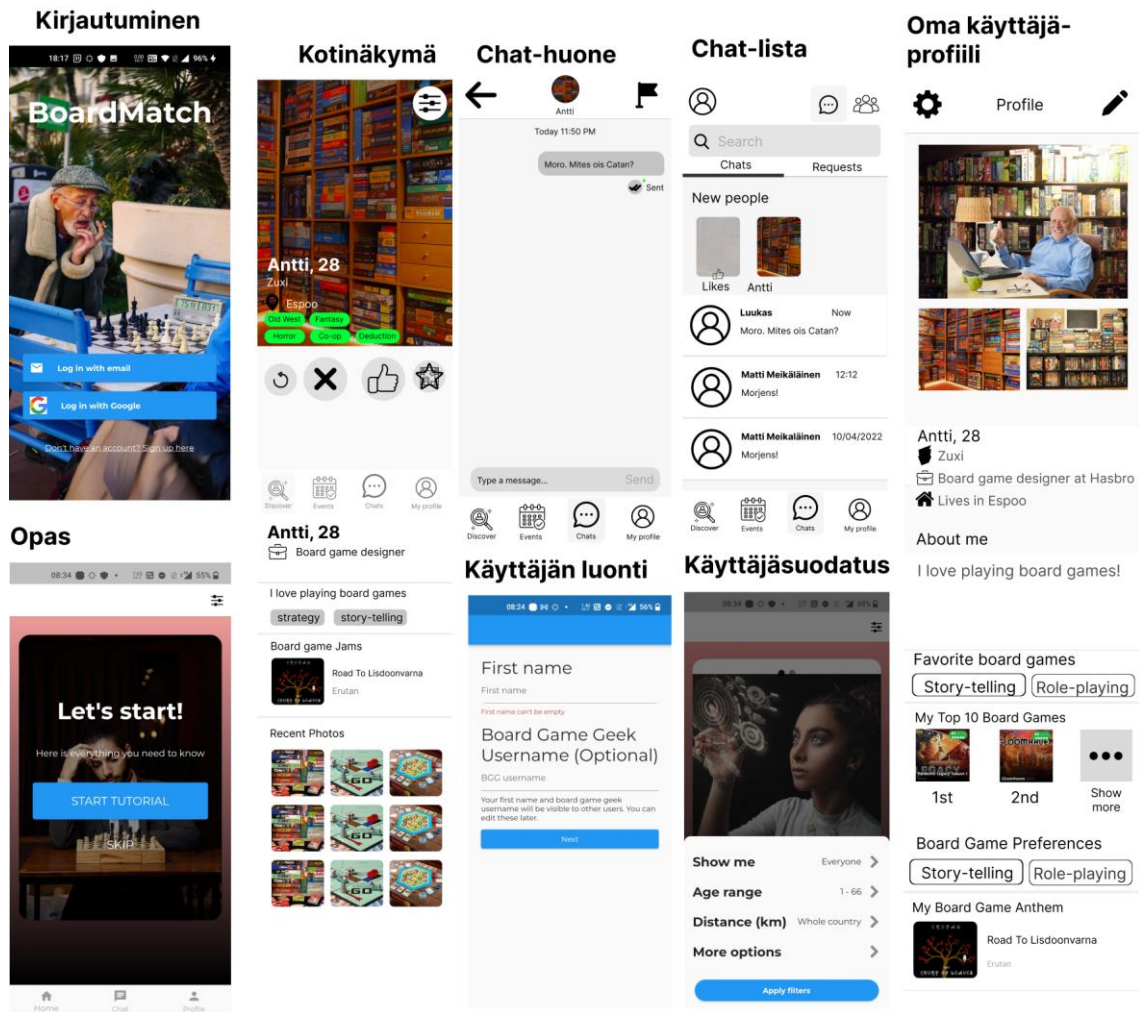
En sisällyttänyt lopulliseen versioon Facebookin kirjautumista ja sosiaalista jakamista, lautapeliryhmän hakemista ja kalenteria lautapelikerhojen ajoista ja tapahtumista, koska niille ei ollut tarvetta sovelluksen ensimmäisessä versiossa ja halusin hioa tärkeimmät ominaisuudet loppuun. Toteutin sovelluksen navigointipalkin, joka sisältää kuvakkeet koti-, chat- ja profiilinäkymälle. Tätä varten Flutter tarjoaa hyödyllisen kirjaston kuvakkeille ja tavan lisätä omaa vektorigrafiikkaa hyödyntäviä kuvakkeita. Vektorigrafiikkaa tarvitaan, koska kuvakkeiden täytyy skaalautua päätekäyttäjän näytön mukaan. Käytin olemassa

olevia seuranhakusovelluksia mallina ja tein rautalankamallit sovelluksesta Figma-palvelussa.

Figma on selainpohjainen suunnittelutyökalu, joka mahdollistaa tiimin reaaliaikaisen yhteistyön. Palvelun pääominaisuuksiin kuuluvat prototyypin, käyttösuunnittelun ja kokemuksen suunnittelu ja piirtäminen. [66.]

Toteutin seuraavanlaiset sovelluksen näkymät ja toiminnot Figmaan (kuva 6):

- kirjautuminen ja rekisteröinti
  - (a) omalla sähköpostiosoitteella tai Google-tilillä
- käyttäjän luonti
  - (a) tietojen täyttäminen: etunimi, Board Game Geek -käyttäjänimi, syntymäaika ja sukupuoli (Board Game Geek on sivusto, jossa käyttäjät pystyvät vertaamaan toistensa pelimakuja [67.]
  - (b) profiilikuvien lataaminen Firebase Firestore -palveluun
- sovelluksen käyttöopas
  - (a) opettaa, miten annetaan ja saadaan tykkäyksiä eli miten löydetään toinen henkilö, kenen kanssa voidaan keskustella
  - (b) lopussa kerrotaan sovelluksen muista toiminnoista
- chat-huone ja -lista
  - (a) lista keskusteluista ja keskusteleminen parin kanssa
- kotinäkö ja käyttäjäsuodatus
  - (a) pyyhkäisemällä käyttäjiä ohitetaan tai tykätään käyttäjästä ja voidaan suodattaa käyttäjiä sukupuolen, iän, etäisyyden, lautapelimieltymysten, kielten ja paikkakunnan mukaan
- käyttäjäprofiilin muokkaus
  - (a) profiilikuvien lataaminen, etunimen ja Board Game Geek -käyttäjänimen muuttaminen sekä syntymäajan, paikkakunnan, henkilökuvauksen ja pelimieltymysten muokkaus.



Kuva 6. Sovelluksen eri näkymät.

Valitsin Flutterin, koska se on alustariippumaton ohjelmistokehys ja sillä voidaan helposti muokata ja kehittää käyttöliittymää. Lisäksi Dart-ohjelmointikielen ja Flutter-ympäristön oppiminen oli helppoa ja verkosta löytyi runsaasti apua. Flutter antaa melkein natiivin suorituskyvyn, mikä tarkoittaa, että sovellus toimii paremmin päätekäyttäjällä, kuin jos kehittäisi sovelluksen jossain toisessa alustariippumattomassa ohjelmistokehityksessä, kuten React Nativessa. Insinööriprojektia varten käytin myös Firebaseia, joka on Googlen pilvipalveluna tarjoama sovellusalusta, jota käytetään muun muassa mobiili- ja web-sovelluksissa. Sovellusalustan tietokanta Firestore on joustava, skaalautuva NoSQL-tietokanta mobiili- ja verkkosovelluksille ja palvelinkehitykseen, ja se tarjoaa integroinnin myös muihin Firebase-tuotteisiin, mukaan lukien Cloud

Functions. Sovelluksen tietokannan suunnitteluun käytin ilmaista app.diagrams-verkkosivustoa.

Sosiaalisen median teknologioiden tutkiminen antoi idean integroida Google-kirjautumisen sovellukseen, jotta käyttäjän ei tarvitse joka kerta näppäillä käyttäjätunnuksia kirjautuakseen sisään. Profiilinäkymän monipuoliset muokkaustyökalut tekevät käyttäjän luomasta profiilista houkuttelevamman ja persoonallisen. Kotinäkyssä ja käyttäjäprofiilin muokkauksessa on alhaalta ylöspäin ponnahtavia ikkunoita, jotka luovat saumattoman käyttökokemuksen.

## 5.2 Automaatio ja työkalut

Mobiilisovellus toteutettiin Flutterin ohjelmistokehityspaketilla, ja kehitysympäristönä käytettiin VS Code -koodieditoria. Kehitin sovelluksen Flutterin versiolla 3.0.4 ja Dart-ohjelmointikielen versiolla 2.17.5. Projektin aikana käytin Ubuntu-versiota 20.04.3, mikä helpotti tiettyjen kirjastojen asentamista. Käytin alusta asti Git-versiohallintaa.

Jotta Firebase-pilvipalvelua voisi käyttää omassa projektissaan, täytyi asentaa Firebase CLI ja kirjautua palvelun sisään käyttämällä Google-tiliä. Lisäksi piti asentaa Flutter-komennoilla tarvittavat Firebase-paketit ja kirjoittaa Flutterin main.dart-tiedostoon alustuskoodi, joka alustaa palvelun mobiilisovelluksessa esimerkkikoodissa 1. [68.]

```
await Firebase.initializeApp(
  options: DefaultFirebaseOptions.currentPlatform
);
```

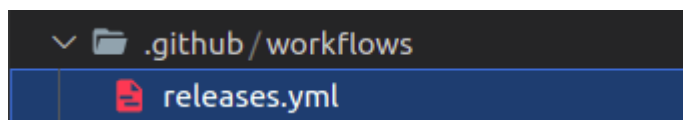
Esimerkkikoodi 1. Firebase-alustuskoodi Flutterissa [68].

Mietin tapoja, joilla olisin pystynyt haastamaan itseäni, ja keksin, että voisin kokeilla ottaa projektin käyttöön jatkuvan integraation ja julkaisun pipeline -menetelmän. Käyttöönnotossa oli monta edellytystä, kuten GitHub-tietolähde, Google service -tili, Google Play -kehittäjätili ja Flutter-projektin määrittäminen

Firebase App Distributionin kanssa. Lisäksi piti määrittää GitHub-tietolähteessä GitHub Actions. [69.]

GitHub Actions on vuonna 2018 julkaistu jatkuvan integraation ja julkaisun alusta, jonka avulla voi automatisoida ohjelmistoprojektin eri vaiheita, kuten testaus ja jakaminen. Sen avulla voi luoda työnkuluja, jotka kokoavat ja testaavat jokaisen GitHub-tietolähteen vetopyynnön, tai ottaa käyttöön yhdistettyjä vetopyyntöjä tuotantoversioon. [70.]

Ensimmäinen vaihe jatkuvan integraation ja julkaisun pipeline -menetelmän käyttöönotossa oli github/workflows-hakemiston luominen projektin juureen (kuva 7).



Kuva 7. GitHub Actions -alustan luominen juurihakemistoon.

Toinen vaihe oli keystore-tiedoston luominen sovelluksen apk-tiedoston allekirjoittamista varten. Esimerkkikoodi 2 määrittää käytettävän algoritmin, avainkoon, voimassaolon pituuden päivinä ja aliaksen nimen. [71.]

```
keytool -genkey -v -keystore ~/upload-keystore.jks -keyalg RSA -  
keysize 2048 -validity 10000 -alias upload
```

Esimerkkikoodi 2. Komento, joka luo keystore-tiedoston.

Seuraavaksi piti viitata upload-keystore.jks-tiedostoon luomalla key.properties-niminen tiedosto projektin juurihakemistoon "/android/key.properties", jonka koodi on esimerkkikoodissa 3.

```
storePassword=<password from previous step>
keyPassword=<password from previous step>
keyAlias=upload
storeFile=<location of the key store file, such as /Users/<user
name>/upload-keystore.jks>
```

### Esimerkkikoodi 3. Viitataan keystore-tiedostoon.

Määritin projektipolussa ”/android/app” olevan build.gradle-tiedoston sovelluksen allekirjoittamista varten. Esimerkkikoodissa 4 määritetään keystore-tiedoston sijainti ja luodaan FileInputStream-olio tiedoston lukemista varten. Release-lohkoon kuuluvat avain-arvokentät, jotka määritettiin esimerkkikoodissa 3.

```
def keystoreProperties = new Properties()
    def keystorePropertiesFile = rootProject.file('key.properties')
    if (keystorePropertiesFile.exists()) {
        keystoreProperties.load(new FileInputStream(keystoreProp-
ertiesFile))
    }

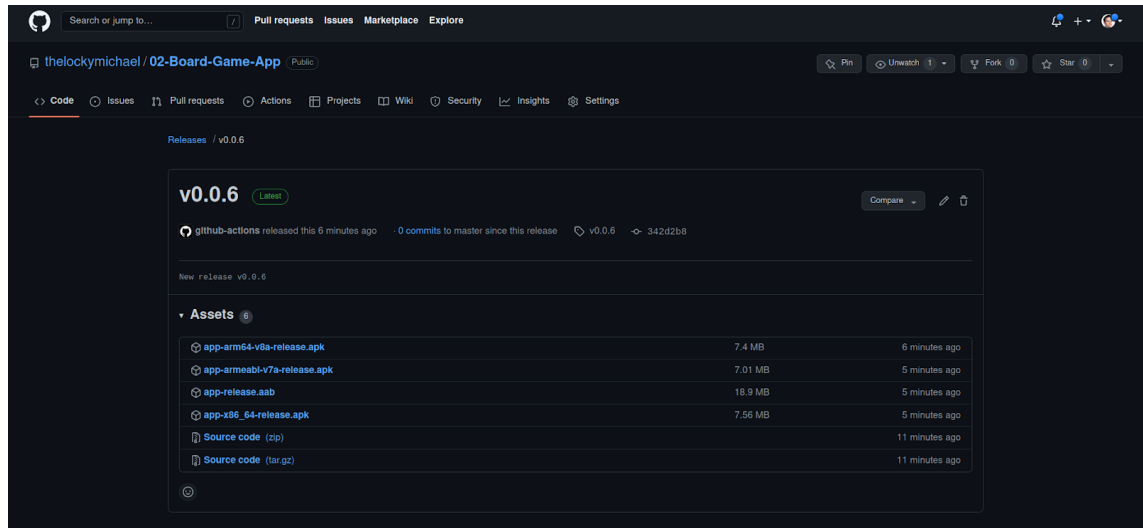
    android {
        ...

        signingConfigs {
            release {
                keyAlias keystoreProperties['keyAlias']
                keyPassword keystoreProperties['keyPassword']
                storeFile keystoreProperties['storeFile'] ? file(keystore-
Properties['storeFile']) : null
                storePassword keystoreProperties['storePassword']
            }
        }
        buildTypes {
            release {
                signingConfig signingConfigs.release
            }
        }
    }
}
```

### Esimerkkikoodi 4. Koodia sovelluksen allekirjoittamista varten.

Vaikein osuus oli yml-tiedoston kirjoittaminen GitHub Actions -työnkulkua varten. Tiedoston alkuun määritin, että kun työntää muutoksen, joka sisältää tagin Git-tietolähteeseen, suoritetaan työnkulku. Ohjeet eivät erikseen kertoneet, miten luodaan tageja Git-versiohallinnassa, joten jouduin etsimään sen tiedon itse esimerkkikoodissa 5. Jotta työnkulku toimisi, määritin GitHub-tietolähteeseen ympäristömuuttujan KEY\_JKS. Ympäristömuuttujaa varten

koodasin upload-keystore.jks-tiedoston binääriset merkkijonot tekstimuotoon base64-komennon avulla, joka näkyy esimerkikoodissa 6. Työnkulun suorita-komennot kokoavat sovelluksesta eri tuotantoversioita Android App Bundle- ja APK-formaateissa ja luovat GitHub Actions -artifakteja. Kuvassa 8 näkyvät pusketut tuotantoversiot sovelluksesta, ja lopullinen työnkulku näkyy esimerkikoodissa 7.



Kuva 8. Sovelluksen asennustiedostot GitHubissa.

```
git tag -a v0.0.6 {commit hash} -m "New release v0.85"
git push --tags
```

Esimerkkikoodi 5. Git-tagin luonti ja puskeminen GitHub-tietolähteeseen.

```
base64 upload-keystore.jks
```

Esimerkkikoodi 6. Upload-keystore.jks-tiedoston muuntaminen koodattuun muotoon.



```

name: Flutter CI/CD # disable this to save storage
on:
  push:
    tags:
      - "v*"
jobs:
  build:
    runs-on: ubuntu-latest
    permissions:
      contents: write
    env:
      KEY_JKS: ${ secrets.KEY_JKS }
      KEY_PATH: "upload-keystore.jks"
      KEY_PASSWORD: ${ secrets.KEY_PASSWORD }
      ALIAS_PASSWORD: ${ secrets.ALIAS_PASSWORD }
    steps:
      - uses: actions/checkout@v2
      - uses: actions/setup-java@v2
        with:
          distribution: 'adopt'
          java-version: '12.x'
      - name: Create key file
        run: |
          echo $KEY_JKS | base64 -di > upload-keystore.jks
      - uses: subosito/flutter-action@v1
        with:
          flutter-version: '2.10.5'
      - run: flutter pub get
      - run: flutter build apk --release --split-per-abi
      - run: flutter build apk --debug
      - run: flutter build appbundle
      - name: Create GitHub artifact release # Disable this to save
storage
uses: ncipollo/release-action@v1
with:
  artifacts: "build/app/outputs/apk/re-
lease/*.apk,build/app/outputs/bundle/release/app-release.aab"
  token: ${ secrets.GITHUB_TOKEN } # This is automatically
provided by GitHub
  - name: Upload app bundle artifact
uses: actions/upload-artifact@v3
with:
  name: appbundle
  path: build/app/outputs/bundle/release/app-release.aab

```

### Esimerkkikoodi 7. Sovelluksen tuotantoversion puskeminen käyttämällä GitHub Actions -työnkulkua.

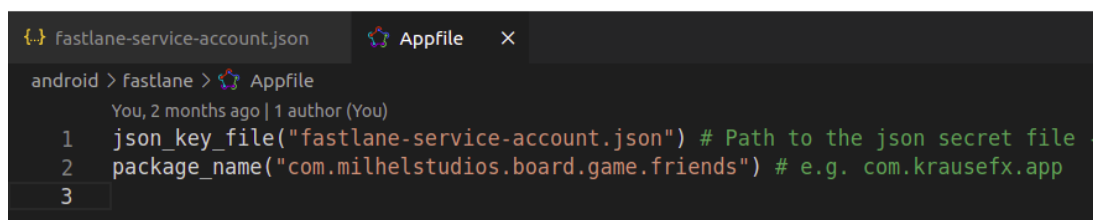
Erikoisuutena halusin kokeilla GitHub Actions -integraation lisäksi Fastlanen käyttöönottoa. Fastlane on avoimen lähdekoodin alusta, jonka tarkoituksena on yksinkertaistaa sovelluksen kehittämistä Androidille ja iOS:lle. Alustalla voidaan luoda automaattisesti sovelluksesta lokalisoituja kuvakaappauksia, jakaa beetaversioita testaajille sekä julkaista ja allekirjoittaa sovelluksia sovelluskauppaan. Otin Fastlanen käyttöön, jotta voisin automatisoida

koontiversion lataamisen Google Play -kauppaan ja Firebase App Distribution -palveluun testattavaksi. Jotta Google Play -kauppa hyväksyisi uusimman version sovelluksesta, versiolukua täytyi kasvattaa yhdellä. Esimerkkikoodissa 8 asensin ensin Rubyn, jotta pystyin asentamaan ja alustamaan Fastlanen.

```
sudo apt install ruby-full
sudo gem install fastlane
fastlane init
```

Esimerkkikoodi 8. Rubyn ja Fastlanen asennuskomennot ja Fastlanen määrittäminen projektikansiossa.

Tarvittiin sovelluksen tunnus ja Google Service -tilin ja JSON-avaimen polku (kuva 9). Tässä vaiheessa olin määrittänyt väärän polun JSON-avaimelle. Seuraamalla tarkemmin ohjeita huomasin, että täytyi erikseen luoda Google Service -tili ja sen JSON-avain Google Cloud -alustassa. Tilin luomisen jälkeen oli annettava käyttäjättilille lupa Google Play -kehittäjäkonsolissa ja otettava Google Play Android Developer -ohjelmistorajapinta käyttöön sovelluksen hallintaan. Google Play -kehittäjäkonsolia varten tarvittiin noin 24 euron kehittäjälisenssi. Maksoin lisenssin itse, sillä tarvitsen sitä myös muissa vapaa-ajan projekteissa.



```
fastlane-service-account.json Appfile
android > fastlane > Appfile
You, 2 months ago | 1 author (You)
1 json_key_file("fastlane-service-account.json") # Path to the json secret file -
2 package_name("com.milhelstudios.board.game.friends") # e.g. com.krausefx.app
3
```

Kuva 9. Sovelluksen tunnus ja JSON-avaimen polun määrittäminen Appfile-tiedostossa.

Kun sain määritettyä Appfile-tiedoston, suoritin komennon, joka loi uuden kansion nimeltä metadata esimerkkikoodissa 9. Komennolla varmistettiin, että Fastlane toimii sovelluksessa.

```
fastlane supply init
```

Esimerkkikoodi 9. Komento metadata-kansion luomiselle.

Nyt Fastlane oli määritetty sovelluksessa ja seuraava vaihe oli kirjoittaa yml-tiedosto Fastlane-työprosessia varten. Tätä varten otin mallia ensimmäisestä yml-tiedostosta.

Esimerkkikoodissa 10 lisäsin kaksi uutta ympäristömuuttujaa Fastfile-tiedostolle ja Google Service -tilin JSON-avaimelle. Lane-nimisten avaimien arvot "closed\_beta" ja "upload\_firebase" hakevat vastaavanlaisesti nimetyt funktiot Fastfile-tiedostosta. Funktiot lähettävät viimeisimmän koontiversion Firebase App Distribution -palveluun ja Google Play -kehittäjäkonsoliin.

```

name: Firebase App Distribution CI/CD
on:
  push:
    tags:
      - release
jobs:
  build:
    env:
      FASTFILE: ${ secrets.FASTFILE }
      FASTLANE_SERVICES: ${ secrets.FASTLANE_SERVICES }
      ...
    steps:
      ...
      - name: Decode base64 secrets
        run: |
          ...
          echo $FASTLANE_SERVICES | base64 -di > android/fastlane-ser-
vice-account.json
          echo $FASTFILE | base64 -di > android/fastlane/Fastfile
      - uses: subosito/flutter-action@v1
        with:
          flutter-version: '3.0.0'
      - run: flutter pub get
      - run: flutter build apk --release --split-per-abi
      - run: flutter build apk --debug
      - run: flutter build appbundle
      - name: Run Fastlane
        uses: maierj/fastlane-action@v1.4.0
        with:
          lane: closed_beta
          subdirectory: android
      - name: Run Firebase App Distribution
        uses: maierj/fastlane-action@v1.4.0
        with:
          lane: upload_firebase
          subdirectory: android

```

Esimerkkikoodi 10. GitHub Actions -työnkulku, joka lähettää koontiversion Google Play -kehittäjäkonsoliin ja Firebase App Distribution -palveluun.

Tärkein koodi määritettiin Fastfile-tiedostoon. Saadakseni Firebase App Distributionin toimimaan Fastlanen kanssa jouduin lisäämään laajennuksen suorittamalla seuraavan komennon: "fastlane add\_plugin firebase\_app\_distribution".

Esimerkkikoodissa 11 on määritetty sovelluksen oletusalustaksi Android ja kaksi funktiota. Ensimmäinen funktio nimeltä "closed\_beta" hakee Google Play -kehittäjäkonsolista sovelluksen viimeisimmät versiot eri julkaisuvaiheista, löytää maksimiarvon ja kasvattaa sovelluksen versiolukua 1:llä. Sovellus lähetetään beeta-julkaisuvaiheeseen korotetulla versioluvulla. Toinen funktio, joka on

nimeltään "upload\_firebase", lähettää sovelluksen testiversion Firebase App Distribution -palveluun jaettavaksi.

```

default_platform(:android)
platform :android do
  desc "Deploy to closed beta track"
  lane :closed_beta do
    begin
      gradle(task: "clean")
      g = google_play_track_version_codes
      gb = google_play_track_version_codes(track: 'beta')
      ga = google_play_track_version_codes(track: 'alpha')
      max_value = [g[0].to_i, gb[0].to_i, ga[0].to_i].max
      puts "this is the version number"
      puts max_value
      version_updated = max_value + 1
      puts "this is the incremented version number"
      puts version_updated
      increment_version_code(
        gradle_file_path: "./app/build.gradle",
        version_code: version_updated.to_i
      )
      gradle(
        task: "bundle",
        build_type: "Release"
      )
      upload_to_play_store(
        track: "beta",
        aab: "../build/app/outputs/bundle/release/app-release.aab",
        skip_upload_metadata: true,
        skip_upload_images: true,
        skip_upload_screenshots: true,
        release_status: "draft"
      )
    end
  end

  desc "Firebase app distribution"
  lane :upload_firebase do
    begin
      gradle(
        task: "assemble",
        build_type: "Debug"
      )
      firebase_app_distribution(
        app: "{app_id}",
        android_artifact_type: "APK",
        firebase_cli_token: "{firebase_cli_token}",
        release_notes: "RELEASE-NOTE",
        android_artifact_path: "../build/app/outputs/apk/debug/app-de-
bug.apk",
        debug: true
      )
    end
  end
  ...

```

**Esimerkkikoodi 11.** Fastfile-tiedosto, joka sisältää funktiot sovelluksen levitykseen.

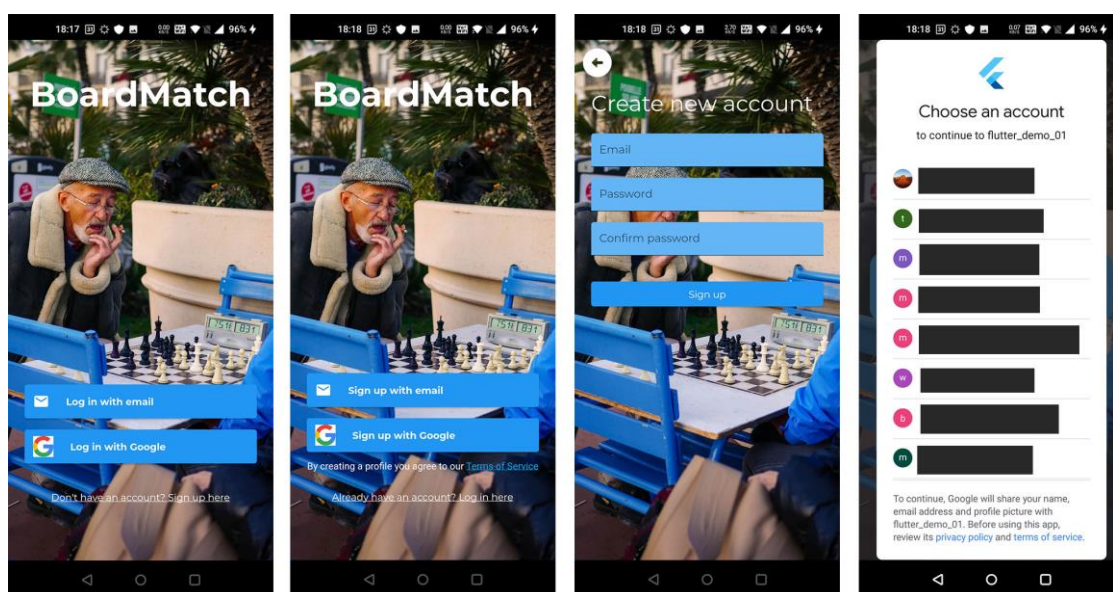
Käytettäessä GitHub Actions -työnkulkuja ja työnnettäessä enemmän sovelluksen koontiversioita sain GitHubilta sähköpostia. Sähköpostissa tiedotettiin, että olin käyttänyt 100 % GitHubin palveluista. Olin käyttänyt yli 512 Mt tilaa, ja jos haluaisin jatkaa koontiversioiden lataamista GitHubiin, minun täytyi maksaa. Tästä syystä en enää ladannut koontiversioita GitHubiin, vaikka käyttöraja nollataan kuitenkin aina kuukauden välein. Tyydyin lataamaan uusimmat versiot sovelluksesta suoraan Firebase App Distribution -palveluun. Firebase App Distribution -palvelussa voidaan kirjoittaa koontiversion muutoksista ja lähettää se napin painalluksella testaajien App Tester -sovellukseen, jonka kautta haetaan sovelluksen viimeisin versio. App Tester asennetaan ensimmäisen kerran, kun käyttäjä liittyy testaajaksi.

Projektissa tehty lautapelisovellus toteutettiin Flutter-ohjelmistokehyksellä Android-puhelimille. Tein sovelluksen Flutterin versiolla 3.0.4, jotta saisin asennettua varmempia ja uusimpia Firebasen ja käyttöliittymäkomponenttien kirjastoja. Koska sovellus tehtiin mobiililaitteelle, piti ladata tarpeellinen Android SDK -lähdekehityspaketti ja integroida se Flutterin kehitysympäristöön. Tein työtä itsenäisesti, ja koska en tehnyt sovellusta asiakastyönä, sitä ei tarvinnut muokata asiakkaan tarpeisiin. Työn aikana lähettelin sovelluksen testiversioita tutuille Firebase App Distribution -teknologian avulla, jotta varmistaisin sovelluksen laadun ja toimivuuden. Käytin projektissa alusta asti Git-versiohallintaa, jotta voisin palata aiempaan versioon ja integroida turvallisesti suurempia ominaisuuksia sovellukseen.

Sovelluksessa käyttäjä kirjautuu sisään omalla sähköpostilla tai Google-tilillä päästäkseen käyttämään lautapelisovelluksen palvelua. Käyttäjä pysyy sisään kirjautuneena, ellei hän tyhjennä sovelluksen välimuistia tai kirjaudu ulos. Sovelluksessa löydetään pareja, kun käyttäjät ovat antaneet toisilleen tykkäyksen, minkä jälkeen heidät yhdistetään chat-huoneeseen. Käyttäjän voi myös ohittaa, jolloin häntä ei enää näytetä sovelluksessa eikä kerrota, että käyttäjä on ohittanut hänet. Käyttäjät voivat jatkaa chat-keskusteluaan sovelluksen ulkopuolella, kuten Discordissa, tai tavata lähimmässä lautapelikerhossa. Käyttäjiä voi suodattaa iän, etäisyyden, sukupuolen, kielten,

paikkakunnan ja pelimieltymysten mukaan. Omaa käyttäjäprofiilia voi muokata eläväisemmäksi ja persoonallisemmaksi ja esitellä omat suosikkilautapelit. Monipuolinen profiili tekee käyttäjästä houkuttelevamman.

Aloitin sovelluksen tekemisen luomalla sisäänkirjautumis- ja rekisteröintinäkömän (kuva 10). Asensin projektihakemistoon `firebase_core`-, `firebase_auth`- ja `google_sign_in`-kirjastot. Tämän jälkeen alustin Firebasen `main.dart`-tiedoston `main`-funktioon (kuva 11) [72]. Käyttäjä voi valita, haluaako hän rekisteröityä omalla sähköpostitilillä vai Google-tilillä. Kun valitaan kirjautuminen omalla sähköpostitilillä vaaditaan että sähköposti on kirjoitettu oikeassa muodossa. Salasana-kentän lisäksi vaaditaan vielä salasanan vahvistus. Jos käyttäjä kirjautuu tai rekisteröityy Google-tilillä, hänen ei tarvitse erikseen syöttää sähköpostiosoitetta ja salasanaa.



Kuva 10. Kirjautuminen ja rekisteröinti sovelluksessa tehdään omalla Google- tai sähköpostitilillä.

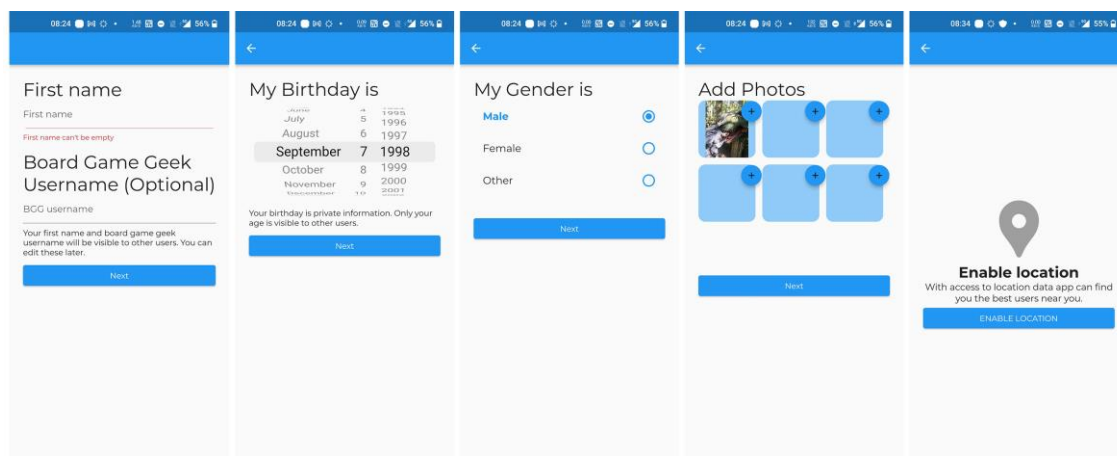
```

35     await Firebase.initializeApp(
36         options: DefaultFirebaseOptions.currentPlatform,
37     );

```

Kuva 11. Firebasen alustuskoodi [72].

Kun sain sisäänkirjautumisen ja rekisteröinnin valmiiksi, siirryin toteuttamaan käyttäjän luontiprosessia. Käyttäjän luontiprosessi koostuu viidestä eri vaiheesta. Ensimmäisessä näkymässä vaaditaan käyttäjän etunimi ja halutessaan voi syöttää Board Game Geek -käyttäjänimensä. Syntymäajan valitsemisen toteutin iOS-tyyppisellä DatePicker-komponentilla. Käyttäjän syntymäaikaa ei näytetä toisille käyttäjille, vaan ainoastaan käyttäjän ikä. Käyttäjän sukupuolen valitsemisen toteutin valintanapeilla. Seuraava vaihe on profiilikuvien lisääminen. Käyttäjän täytyy lisätä vähintään yksi profiilikuva jatkaakseen eteenpäin. Profiilikuvia voi lisätä enintään kuusi, mikä on riittävä määrä lautapelialiheiseen seuranhakusovellukseen. Viimeisessä käyttäjäluontivaiheessa halusin, että käyttäjä ottaisi puhelimensa GPS-paikantimen käyttöön, jotta sovellus näyttäisi käyttäjät, jotka ovat lähellä. Käytin erilaisia käyttöliittymäkomponentteja käyttäjän luontiprosessissa, koska halusin tehdä käyttöliittymästä ja -kokemuksesta monipuolisemman ja vaihtelevan. (Kuva 12.)



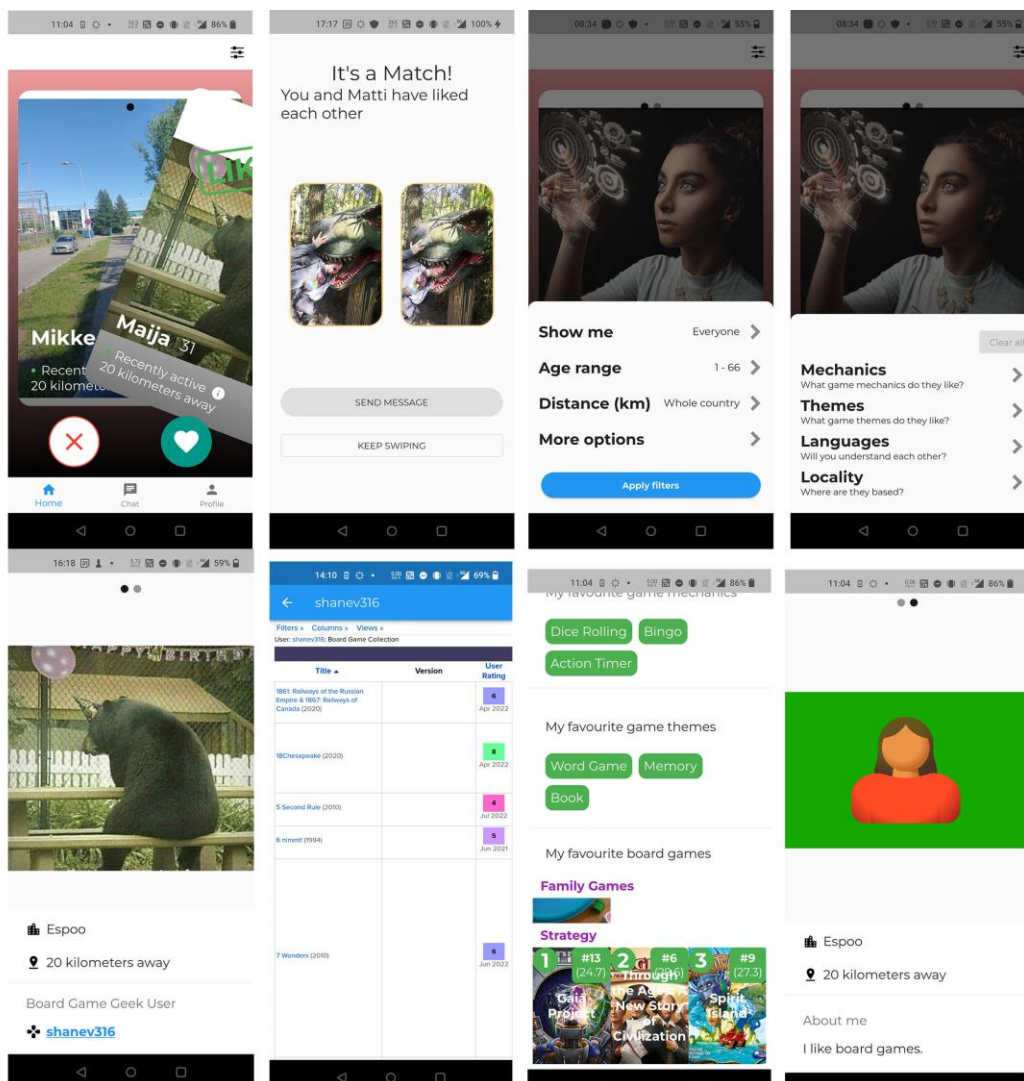
Kuva 12. Käyttäjän luontiprosessi.

Käyttäjäluonnin jälkeen aloin tekemään alaosan navigointipalkkia, johon tein kolme päänäkymää, jotka olivat koti, chat ja profiili. Tässä vaiheessa halusin varmistaa navigointipalkin toimivuuden ja sen, että päänäkymien keskellä näytettäisiin vastaavanlaiset tekstit. Seuraavaksi lähdin toteuttamaan kotinäkymää, jonka sisällölle määritin seuraavat kriteerit:



- Korttipino käyttäjistä, joita pyyhkäisemällä voidaan ohittaa tai antaa käyttäjälle tykkäys. Kun toista käyttäjää on pyyhkäisty, sen tunnisteläisätään sisään kirjautuneen käyttäjän ignoreSwipelds-taulukkoon.
- Käyttäjäsuodatus on alhaalta ylöspäin liukuva ponnahdusikkuna. Käyttäjiä suodatetaan sukupuolen, iän, etäisyyden, kielten, paikkakuntien ja käyttäjän lempipelimekaniikkojen ja -teemojen mukaan
- Korttipinon päällimmäisen käyttäjän tietoja ja kuvia voi tarkastella. Kun tarkastelee käyttäjätietoja, kortti laajennetaan käyttäjän kokonäkymään, jotta fonttien näkyvyys ja käyttökokemus olisi parempi. Halusin noudattaa tavanomaista käyttöliittymäsuunnittelua deittiyhteisön kaltaisissa sovelluksissa.
- Kun käyttäjä löytää parin, hänet navigoidaan It's a Match! -näkyymään, josta hän pääsee jatkamaan chat-huoneeseen tai jatkamaan käyttäjien pyyhkäisyä. Chat-näkymään lisätään kahden käyttäjän välinen chat-huone.

(Kuva 13.)



Kuva 13. Kotinäkö, johon kuuluu käyttäjien suodatus ja käyttäjätietojen tarkistelu ja näkö, joka ilmoittaa löydetyistä parista.

Tinderin kaltaisen sovelluksen kannalta oli tärkeää, että sovelluksen käyttäjäkortteja pystyi pyyhkäisemään vasemmalle, oikealle tai ylöspäin. Pääsin heti kokeilemaan Flutterin animaatioita käyttämällä Johannes Milken kaksiosaisesta Flutter Tutorial - Tinder Swipe Cards -tutoriaalia. Valitettavasti hänen GitHub-tietolähteensä ovat maksullisia, joten selviydyin ainoastaan katsomalla tarkasti hänen opetusvideoitaan. Yhdistin käyttäjän pyyhkäisyn toiminnallisuuden käyttämään Firebase Firestore -palvelua soveltamalla GitHubista löytynyttä tietolähdettä. Otin supertykkäys-painikkeen pois näkyvistä opinnäytetyön versioon, sillä painikkeella ei ollut toiminnallisuutta. [73; 74.]

Halusin sovelluksen näyttävän aktiiviset ja lähimmät käyttäjät. Firebase-alustassa on Cloud Functions -palvelu, jossa loin tämän toiminnon. Sovelluksen sisällä voidaan hakea ja valinnaisesti suodattaa käyttäjiä lähettämällä http-kutsu. Kutsu palauttaa vastauksena käyttäjämäärän ja taulukon kilometrin tarkkuudella (kuva 14).

```
const usersRef: admin.firestore.Query<admin.firestore.DocumentData> = db
  .collection('users')
  .limit(limit)
  .orderBy('updatedAt', 'desc')

const users = await usersRef.get()

const newPromise: Array<ResultAppUser> = await new Promise((resolve, reject) => {
  const tmpUsers: Array<ResultAppUser> = []

  users.docs.forEach((doc) => {
    const user = doc.data() as ResultAppUser

    const ignoreIdExists = ignoreIds.some((ignoreId) => user.id === ignoreId)
    if (ignoreIdExists) {
      return
    }

    // 1. My user
    const myUserLat: number = lat as unknown as number
    const myUserLong: number = long as unknown as number
    // 2. Other user
    const otherLat = user.currentGeoLocation.latitude
    const otherLong = user.currentGeoLocation.longitude

    const distanceFromMyUser = getDistanceFromLatLonInKm(
      myUserLat,
      myUserLong,
      otherLat,
      otherLong,
    )
  })
})
```

Kuva 14. Koodi, joka hakee lähimmät käyttäjät.

Kun käyttäjä siirtyy ensimmäisen kerran kotinäkömään, ladataan muita käyttäjiä, joita ei ole vielä pyyhkäisty. Asynkroninen funktio hakee käyttäjän sisäisestä muistista käyttäjätunnuksen. Tunnuksella haetaan käyttäjädokumentti ja lisätään listaan pyyhkäistyt käyttäjät. Lopuksi Firebase-pilvifunktiolla haetaan käyttäjät. (Kuva 15.)

```

Future<List<ResultAppUser>?> loadUsers() async {
  String? id = await SharedPreferencesUtil.getUserId();

  if (id != null)
    _myUser = AppUser.fromSnapshot(await _databaseSource.getUser(id));
  _ignoreSwipeIds = <String>[];

  var swipes = await _databaseSource.getSwipes(_myUser.id);
  for (var i = 0; i < swipes.size; i++) {
    Swipe swipe = Swipe.fromSnapshot(swipes.docs[i]);
    _ignoreSwipeIds.add(swipe.id);
  }
  _ignoreSwipeIds.add(_myUser.id);

  var res = await GetNearestUsers().getNearestUsers(
    limit: 50,
    myUser: _myUser,
    gender: defaultSelectedGender[0],
    minAge: defaultMinAgeValue[0],
    maxAge: defaultMaxAgeValue[0],
    distance: defaultDistance[0],
    mechanics: defaultMechanics,
    themes: defaultThemes,
    languages: defaultLanguages,
    localities: defaultLocality,
    ignoreSwipeIds: _ignoreSwipeIds,
  );

  if (res.isNotEmpty) {
    You, 4 weeks ago • get nearest users 2.5 ...
    cardProvider.setUsers(res.reversed.toList());

    return res.reversed.toList();
  }

  return null;
}

```

Kuva 45. Funktio käyttäjien lataamista varten.

Kun käyttäjä ohittaa tai antaa käyttäjälle tykkäyksen pyyhkäisemällä käyttäjäkorttia tai painamalla tykkäyspainiketta, kutsutaan personSwiped-niminen funktio (kuva 16). Molemmille käyttäjille lisätään tietokantaan match-niminen dokumentti. Sillä hetkellä, kun sisään kirjautunut käyttäjä antaa tykkäyksen toiselle käyttäjälle, joka on myös antanut tälle tykkäyksen, työnnetään käyttäjän eteen sovellusnäkyvä, jossa ilmoitetaan uudesta löydetyistä parista.

```

void personSwiped(BuildContext context, AppUser myUser, bool isLiked) async {
  _databaseSource.addSwipedUser(myUser.id, Swipe(users.last.id, isLiked));
  _ignoreSwipeIds.add(users.last.id);

  if (isLiked) {
    if (await isMatch(myUser, users.last) == true) {
      _databaseSource.addMatch(myUser.id, Match(users.last.id));
      _databaseSource.addMatch(users.last.id, Match(myUser.id));

      String chatId = compareAndCombineIds(myUser.id, users.last.id);

      _databaseSource.addChat(Chat(chatId, myUser.id, users.last.id, null));

      Navigator.pushNamed(context, MatchedScreen.id, arguments: {
        "my_user_id": myUser.id,
        "my_profile_photo_path": myUser.profilePhotoPaths[0],
        "other_user_profile_photo_path": users.last.profilePhotoPaths[0],
        "other_user_id": users.last.id,
        "other_user_name": users.last.name
      });
    }
  }
}

```

Kuva 16. Toiminto pyyhkäistylle käyttäjälle.

Toiminnolla isMatch tarkastellaan, onko käyttäjän swipes-kokoelmassa toisen käyttäjän tunnusta. Jos dokumentti on olemassa ja likes-avainkentän arvo on totuusarvoltaan True, käyttäjä on löytänyt uuden parin (kuva 17).

```

Future<bool> isMatch(AppUser myUser, ResultAppUser otherUser) async {
  DocumentSnapshot swipeSnapshot =
    await _databaseSource.getSwipe(otherUser.id, myUser.id);

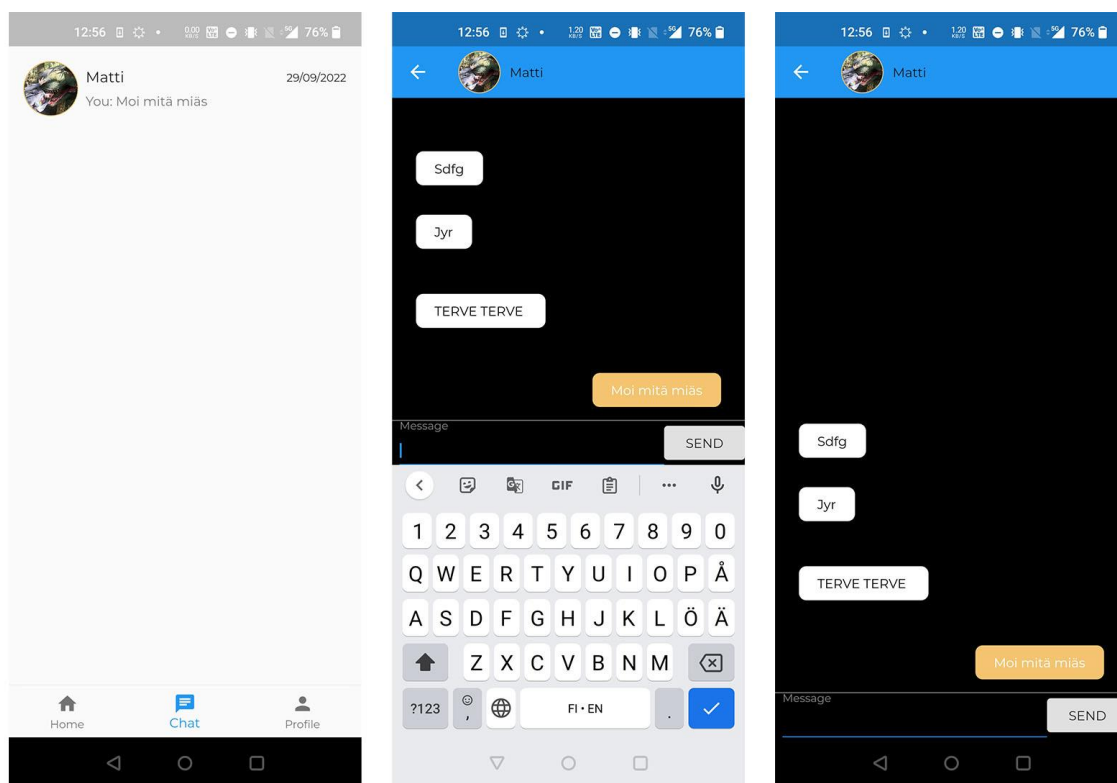
  if (swipeSnapshot.exists) {
    Swipe swipe = Swipe.fromSnapshot(swipeSnapshot);

    if (swipe.liked) {
      return true;
    }
  }
  return false;
}

```

Kuva 17. Funktio tarkistaa, ovatko molemmat käyttäjät antaneet toisilleen tykkäyksen.

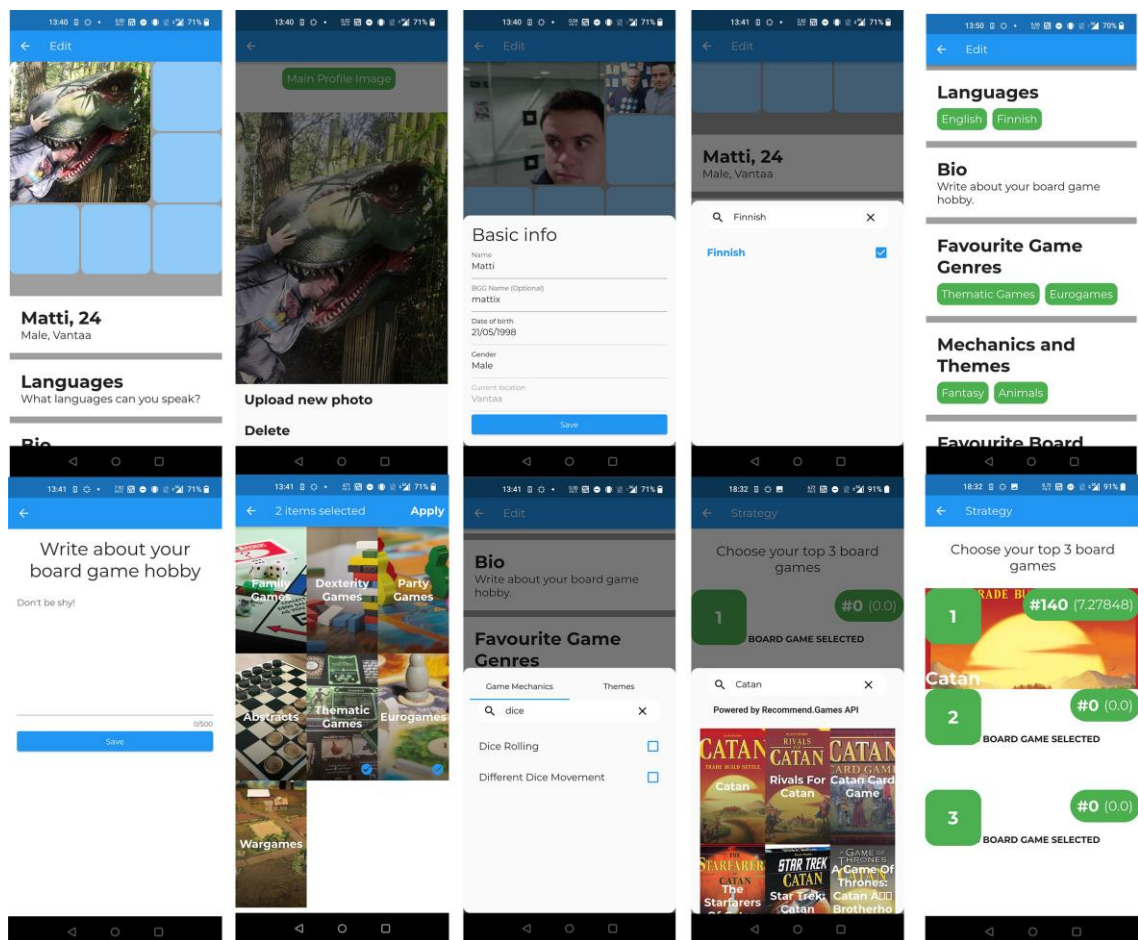
Vielä puuttui Chat-listan ja -huoneen toteutus. Halusin, että chat-lista muistuttaisi WhatsAppia. Listassa jokainen chat-huonepainike sisältää toisen käyttäjän profiilikuvan, viimeksi lähetetyn viestin ja sen päivämäärän. Chat-huonetta painamalla pääsee keskusteluun, jossa voi lähettää viestin toiselle käyttäjälle. Toteutin tämän tavanomaisen viestintäratkaisun, sillä valtaosa sosiaalisen mediankäyttäjistä on tottunut siihen. (Kuva 18.)



Kuva 18. Chat-lista ja -huone.

Sovellus alkoi vahvasti muistuttaa Tinderiä. Jotta sovelluksesta ei tulisi Tinder-kloonina, toteutin lautapeliaiheisen käyttäjäprofiilimuokkauksen. Tavanomaisten tietojen, kuten etunimen, syntymäajan ja sukupuolen muuttamisen lisäksi voi lisätä Board Game Geek -käyttäjänimen. Lisäksi voi valita mieluiset pelimekaniikat ja -teemat ja lautapelityypit, kuten perhe- ja partypelit. Muita muokattavia asioita ovat oman kuvauksen muokkaus ja kielten lisääminen. Käyttäjä voi valita myös kolme parasta suosikkilautapeliään jokaisesta kategoriasta. Yritin etsiä yhteensopivan ohjelmistorajapinnan, jolla voidaan hakea lautapelejä kategorian tai nimen perusteella. Ajattelin ensimmäiseksi

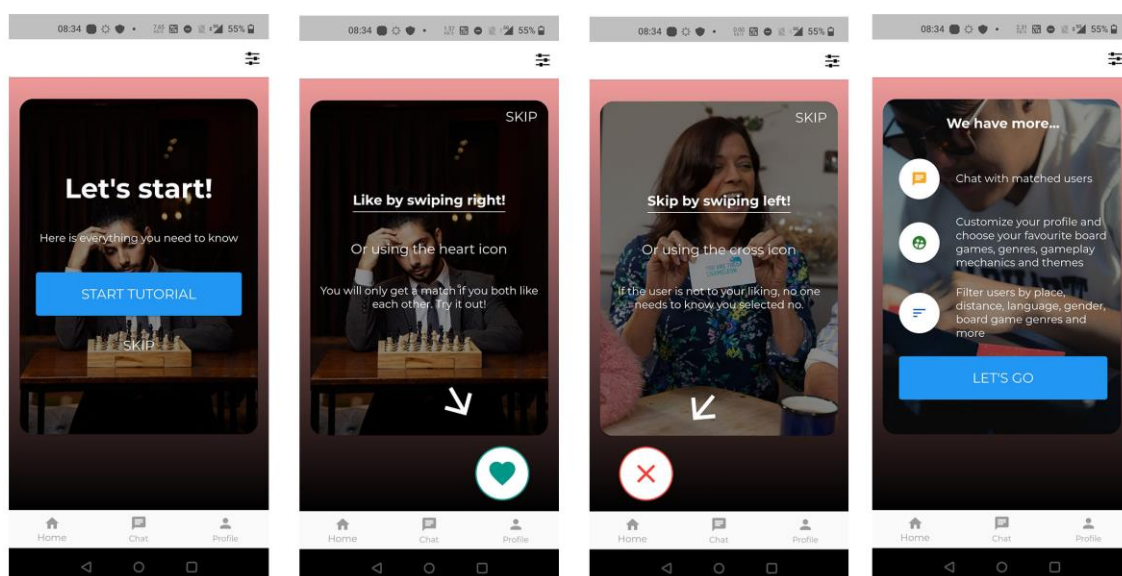
kokeilla Board Game Geekin XMLAPI2 -rajapintaa. Tämä kävi kuitenkin työlääksi, ja rajapinta käytti JSON-formaatin sijaan XML:ää, jota en ole juurikaan ennen sssskäyttänyt. Rajapinta ei tarjonnut sujuvaa tapaa tehdä API-kutsuja pelikategorian ja nimen perusteella, joten päädyin kokeilemaan Lautapelaajien Discord-ryhmäläisen ehdottamaa Recommend Games -rajapintaa, joka toimi tähän käyttötarkoitukseen. Käytin alhaalta ylöspäin liukuvia ponnahtusikkunoita antaakseni käyttäjälle saumattoman käyttökokemuksen. Käyttäjän ei tarvitse poistua toisesta näkymästä palatakseen ensimmäiseen näkymään. (Kuva 19.)



Kuva 19. Eri näkymiä käyttäjäprofiilin muokkauksessa.

Insinööriyön ohjaaja Ulla Sederlöf suositteli, että tekisin sovellukselle käyttöoppaan. Tämä olisi viimeinen lisäys sovellukseen insinööriyöversiossa. Käyttöopas avataan heti, kun käyttäjä on suorittanut käyttäjänluontiprosessin.

Kun käyttöoppaan on suorittanut, sovelluksen preferensseihin tallennetaan totuusarvo True. Sovellus ei näytä sovelluksen käynnistyksen ja sisään kirjautumisen yhteydessä käyttöopasta. Käyttöopas esittelee, miten löydetään pareja painamalla tykkäyspainiketta tai pyyhkäisemällä käyttäjäkorttia oikealle. Jos käyttäjä ei ole mieleinen, hänet voi ohittaa, eikä tätä kerrota hänelle. Käyttöopas toimii tärkeänä esittelynä sovelluksen käytölle ja esittelee oleellisen käyttäjäpyyhkäisyn animaatiot. Lopussa kerrotaan sovelluksen muista ominaisuuksista, kuten chat-keskusteluista, käyttäjäprofiilin muokkauksesta ja käyttäjien suodatuksesta. (Kuva 20.)



Kuva 20. Sovelluksen käyttöoppaan näkymät.

### 5.3 Palaute

Insinööriyöprojektin aikana en löytänyt isompaa ihmisryhmää, jolle esittää mobiilisovellusta, mutta kokeilin sitä tuttavien keskuudessa. Suoritin käyttäjätestauksen viidellä eri käyttäjällä, jotka edustavat eri ikäluokkia. Annoin testikäyttäjille Android-puhelimeni, jossa lautapelisovellus oli valmiiksi asennettuna, ja annoin heille tehtäviä suoritettavaksi. Rajasin käyttäjätestauksen viiteen eri päätehtävään, jotka olivat käyttäjän luonti, tutoriaali, parin löytäminen, käyttäjäprofiilin muokkaaminen ja käyttäjien suodatus. Eniten ongelmia oli käyttäjäprofiilin muokkauksessa. Käyttäjät eivät



osanneet löytää haluamaansa lautapeliä, ja pelikategorian valitseminen koettiin turhaksi. Muut huomioitavat asiat liittyivät käyttökokemukseen ja saavutettavuuteen chat-ruudussa, käyttäjämuokkauksessa ja luontiprosessissa. Nämä ongelmat kirjattiin muistiin, mutta ne eivät liity opinnäytetyön sisältöön, vaan ne muutetaan seuraavassa versiossa.

## 6 Yhteenveto

Insinööriyössä tutkittiin, mikä on hyvä mobiilisovelluksen ohjelmistokehitys sosiaalisen median sovelluksen kehittämiseen. Työssä tehtiin lautapelialeinen seuranhaku-sovellus, joka toteutettiin Flutter-ohjelmistokehityksellä. Kun tutkittiin eri ohjelmistokehityksiä, suosituin kehittäjien ja käytön suhteen oli Flutter. Tätä ohjelmistokehitystä käytettiin eniten maailmanlaajuisesti vuosina 2019–2021. Tutkittiin myös sosiaalisen median suosituimpia ohjelmistorajapintoja käyttöasteen ja suosion mukaan BuiltWith- ja Stackoverflow tags -arvioiden mukaan, ja Facebook ja Twitter olivat suosituimpia. Pääteltiin, että Facebookin rajapinta kannatti integroida omaan sovellukseen.

Automaatio vei melko suuren osan kehityksestä, sillä sen käyttöönotto oli hyvin haastavaa. Ohjeet olivat usein epäselvät, aikaa kului ohjelmointivirheiden selvittämiseen ja hämminkiä aiheutti automaatioon tarvittava toinen Google Service -tilin avain. Insinööriyön aikana käytettiin ainoastaan automaation App Distribution -osuutta, mutta siitä oli kuitenkin hyötyä, koska saavutettiin ainakin pari aktiivista testaaajaa ja insinööriyön aikana onnistuttiin työntämään 21 koontiversiota. Automaatiota täytyy kehittää siihen pisteeseen, että käyttäjät voivat kirjoittaa tekijälle ohjelmointivirheraportteja. Käyttäjäprofiilin lempilautapelien valitsemiseen täytyy etsiä myös toinen ratkaisu, sillä Recommend Games -rajapinta on otettu pois käytöstä.

Projekti toimi hyvänä harjoitteluna projektin hallintaan, sovelluksen suunnitteluun ja sovelluksen kehittämiseen. Jatkokehityksen kannalta voisi vielä parantaa ja kehittää sovelluksen käytettävyyttä, saavutettavuutta ja visuaalista ilmettä. Ennen kuin sovellus voidaan julkaista Google Play -kauppaan, täytyy

suojata käyttäjän tietoja, parantaa käyttäjän yksityisyysuojaa ja muita asetuksia ja kehittää sovelluksen keskeisiä ominaisuuksia. Lisäksi tarvitaan alfa- ja beetatestaajia, jotta ohjelmointivirheitä voidaan korjata sitä mukaa, kuin niitä tulee.

Vaikka opinnäytetyön tekijä oli aikaisemmin kehittänyt keskeneräisen Urban Exploration -tyyppisen sovelluksen React Nativella, Flutterin käyttö antoi uudenlaisen tavan ja kokemuksen kehittää mobiilisovellusta.

Käyttöliittymäkomponenttien ja Firebase-palveluiden tuominen projektiin sujui helposti. Sovelluksesta saatu palaute oli suurimmaksi osaksi positiivista, ja se antaa eväitä jatkokehitykseen.

Sovellusta ei julkaista Suomen lautapeliseuran käyttöön, koska se ei ole valmis julkaistavaksi.

## Lähteet

- 1 Scott, Peter & Jacka, Mike. 2011. Auditing social media. E-kirja. John Wiley & Sons.
- 2 Obar, Jonathan & Wildman, Steven. 2015. Social media definition and the governance challenge: An introduction to the special issue. Telecommunications policy, 39(9), s. 745–750.
- 3 Definition of social media. 2022. Verkkoaineisto. Merriam-Webster. <<https://www.merriam-webster.com/dictionary/social%20media#h1>>. Päivitetty 9.4.2022. Luettu 13.4.2022.
- 4 Aichner, Thomas & Jacob, Frank. 2015. Measuring the Degree of Corporate Social Media Use. International Journal of Market Research. Vol. 57, s. 257–275.
- 5 Howe, Daniel. 2007. What Hath God Wrought. E-kirja. Oxford University Press.
- 6 Before Twitter and Facebook, there was Morse code: Remembering social media’s true inventor. 2017. Verkkoaineisto. The Washington Post. <<https://www.washingtonpost.com/news/retropolis/wp/2017/05/24/before-there-was-twitter-there-was-morse-code-remembering-social-medias-true-inventor/>>. Päivitetty 24.5.2017. Luettu 14.4.2022.
- 7 Kaminow, Ivan & Li, Tingye. 2002. Optical Fiber Telecommunications IV-B: Systems and Impairments. E-kirja. Elsevier.
- 8 Woolley, David. 1994. PLATO: The Emergence of Online Community. Verkkoaineisto. Matrix News. <<https://web.archive.org/web/20040404002038/http://thinkofit.com/plato/dwplato.htm>>. Luettu 18.4.2022.
- 9 Ngak, Chenda. 2011. Then and now: a history of social networking sites. Verkkoaineisto. CBS News. <<https://www.cbsnews.com/pictures/then-and-now-a-history-of-social-networking-sites/>>. Päivitetty 6.7.2011. Luettu 14.4.2022.
- 10 What led to the failure of six degrees? 2019. Verkkoaineisto. Renewable Freelancer Services. <<https://www.renewablefreelancer.com/blog/what-led-to-the-failure-of-six-degrees/>>. Päivitetty 26.11.2019. Luettu 14.4.2022.
- 11 Vinney, Cynthia. 2022. What Is Uses and Gratifications Theory in Media Psychology? Verkkoaineisto. Verywell Mind.

- <<https://www.verywellmind.com/what-is-uses-and-gratifications-theory-in-media-psychology-5217572>>. Päivitetty 7.2.2022. Luettu 21.9.2022.
- 12 Auxier, Brooke & Anderson, Monica. 2021. Social Media Use in 2021. Verkkoaineisto. Pew Research Center. <<https://www.pewresearch.org/internet/2021/04/07/social-media-use-in-2021/>>. Päivitetty 7.4.2021. Luettu 21.9.2022
  - 13 Kemp, Simon. 2022. Digital 2022: July global statshot report. Verkkoaineisto. DataReportal. <<https://datareportal.com/reports/digital-2022-july-global-statshot>>. Päivitetty 21.7.2022. Luettu 21.9.2022.
  - 14 Humphreys, Ashlee. 2016. Social media: Enduring Principles. Oxford University Press.
  - 15 Suojaa yksityisyytesi netissä — 5 yllättävää hyötyä. 2022. Verkkoaineisto. F-Secure. <<https://www.f-secure.com/fi/home/articles/5-ways-privacy-protection-benefits-you#:~:text=Yksityisyys%20netiss%C3%A4%20tarkoittaa%2C%20ett%C3%A4%20selaushistoriasi,halua%20antaa%20n%C3%A4it%C3%A4%20tie-toja%20heille.>>. Luettu 6.6.2022.
  - 16 Schneier, Bruce. 2006. The Value of Privacy. Verkkoaineisto. Schneier. <[https://www.schneier.com/blog/archives/2006/05/the\\_value\\_of\\_pr.html](https://www.schneier.com/blog/archives/2006/05/the_value_of_pr.html)>. Päivitetty 19.5.2006. Luettu 10.6.2022.
  - 17 Boerman, Sophie; Kruikemeier, Sanne & Borgesius, Frederik. 2017. Online Behavioral Advertising: A Literature Review and Research Agenda. Verkkoaineisto. Taylor and Francis Online. <<https://www.tandfonline.com/doi/full/10.1080/00913367.2017.1339368>>. Päivitetty 29.6.2017. Luettu 10.6.2022.
  - 18 Muhammad, Zia. 2022. 515 New Apps Mark the Rise of Anonymity. Verkkoaineisto. Digital Information World. <<https://www.digitalinformation-world.com/2022/09/515-new-apps-mark-rise-of-anonymity.html>>. Päivitetty 20.9.2022. Luettu 27.9.2022.
  - 19 Curry, David. 2022. Signal Revenue & Usage Statistics (2022). Verkkoaineisto. Business of Apps. <<https://www.businessofapps.com/data/signal-statistics/>>. Päivitetty 6.9.2022. Luettu 27.9.2022.
  - 20 Kendrick, Davis. 2018. The App That Launched a Thousand Memes. Sixth Tone. Verkkoaineisto. <<https://www.sixthtone.com/news/1001728/the-app-that-launched-a-thousand-memes>>. Päivitetty 20.2.2018. Luettu 10.6.2022.

- 21 Graziani, Thomas. 2018. How Douyin became China's top short-video App in 500 days. Verkkoaineisto. Walkthechat. <<https://walkthechat.com/douyin-became-chinas-top-short-video-app-500-days/>>. Päivitetty 30.7.2018. Luettu 10.6.2022.
- 22 Tik Tok, a Global Music Video Platform and Social Network, Launches in Indonesia. 2017. Verkkoaineisto. PR Newswire Asia Ltd. <<https://walkthechat.com/douyin-became-chinas-top-short-video-app-500-days/>>. Päivitetty 13.9.2017. Luettu 10.6.2022.
- 23 Matsakis, Louise. 2019. A Beginner's Guide to TikTok. Wired. Verkkoaineisto. <<https://www.wired.com/story/how-to-use-tik-tok/>>. Päivitetty 6.3.2019. Luettu 10.6.2022.
- 24 Perez, Sarah. 2021. TikTok expands max video length to 3 minutes, up from 60 seconds. Verkkoaineisto. TechCrunch. <[https://techcrunch.com/2021/07/01/tiktok-expands-max-video-length-to-3-minutes-up-from-60-seconds/?guccounter=1&guce\\_referrer=aHR0cHM6Ly9maS53aWtpcGVkaWEub3JnLw&guce\\_referrer\\_sig=AQAAAFci5xojOOK5HydDcZ7i7OPc6b oukvG1DjBdWrMcZGfAAIXacP\\_eJ-xdrWOIo2GBGHu8cqAKNS\\_6tnj-VxhYDpuzJRWhahpJzmpdhDeM9WN4f4ZkR5KuFNi3mR3i4DatmvRflv4alRMh-SOve7U7F9FYsk7Qj6mzsd3dxLOjRV2](https://techcrunch.com/2021/07/01/tiktok-expands-max-video-length-to-3-minutes-up-from-60-seconds/?guccounter=1&guce_referrer=aHR0cHM6Ly9maS53aWtpcGVkaWEub3JnLw&guce_referrer_sig=AQAAAFci5xojOOK5HydDcZ7i7OPc6b oukvG1DjBdWrMcZGfAAIXacP_eJ-xdrWOIo2GBGHu8cqAKNS_6tnj-VxhYDpuzJRWhahpJzmpdhDeM9WN4f4ZkR5KuFNi3mR3i4DatmvRflv4alRMh-SOve7U7F9FYsk7Qj6mzsd3dxLOjRV2)>. Päivitetty 1.7.2021. Luettu 10.6.2022.
- 25 Ellis-Petersen, Hannah. 2020. India bans TikTok after Himalayan border clash with Chinese troops. Verkkoaineisto. The Guardian. <<https://www.theguardian.com/world/2020/jun/29/india-bans-tiktok-after-himalayan-border-clash-with-chinese-troops>>. Päivitetty 29.6.2020. Luettu 10.6.2022.
- 26 Purwaningsih, Ayu. 2018. Indonesia blocks 'pornographic' Tik Tok app. Verkkoaineisto. Deutsche Welle. <<https://www.dw.com/en/indonesia-blocks-pornographic-tik-tok-app/a-44537230>>. Päivitetty 5.7.2018. Luettu 10.6.2022.
- 27 Islam, Arafatul. 2019. Bangladesh 'anti-porn war' bans blogs and Google books. Verkkoaineisto. Deutsche Welle. <<https://www.dw.com/en/bangladesh-anti-porn-war-bans-blogs-and-google-books/a-47684058>>. Päivitetty 25.2.2019. Luettu 10.6.2022.
- 28 'It Encourages Pornography': Madras High Court Asks Government to Ban Video App TikTok. 2019. Verkkoaineisto. News 18. <<https://www.news18.com/news/india/it-encourages-pornography-madras-high-court-asks-government-to-ban-video-app-tiktok-2088337.html>>. Päivitetty 4.4.2019. Luettu 10.6.2022.

- 29 Biddle, Sam; Ribeiro, Paulo & Dias, Tatiana. 2020. Invisible Censorship – TikTok told moderators to suppress posts by “Ugly” people and the poor to attract new users. Verkkoaineisto. The Intercept. <<https://theintercept.com/2020/03/16/tiktok-app-moderators-users-discrimination/>>. Päivitetty 16.3.2020. Luettu 10.6.2022.
- 30 Hern, Alex. 2020. TikTok 'tried to filter out videos from ugly, poor or disabled users'. Verkkoaineisto. The Guardian. <<https://www.theguardian.com/technology/2020/mar/17/tiktok-tried-to-filter-out-videos-from-ugly-poor-or-disabled-users>>. Päivitetty 17.3.2020. Luettu 10.6.2022.
- 31 4 Ways TikTok Differs from Other Social Media Platforms. 2021. Verkkoaineisto. Sumner Group. <<https://www.sumnergroup.com/business-tips/4-ways-tiktok-is-different/#:~:text=First%20and%20foremost%2C%20TikTok%20centers,with%20platforms%20such%20as%20YouTube.>>. Päivitetty 28.10.2021. Luettu 11.6.2022.
- 32 Blystone, Dan. 2022. The Story of Instagram: The Rise of the #1 Photo-Sharing Application. Verkkoaineisto. Investopedia. <<https://www.investopedia.com/articles/investing/102615/story-instagram-rise-1-photo0sharing-app.asp#:~:text=Instagram%20is%20a%20photo%20and,app%20was%20launched%20on%20Oct./>>. Päivitetty 29.3.2022. Luettu 12.6.2022.
- 33 Dixon, S. 2022. What Is the Difference Between Facebook and Instagram? Verkkoaineisto. Statista. <<https://www.statista.com/statistics/325587/instagram-global-age-group/#statisticContainer>>. Päivitetty 12.5.2022. Luettu 12.6.2022.
- 34 The 100 best iPhone apps of all time. 2015. Verkkoaineisto. Mashable, Inc. <<https://mashable.com/archive/100-best-iphone-apps>>. Päivitetty 8.12.2015. Luettu 12.6.2022.
- 35 Instagram 'worst for young mental health'. 2017. Verkkoaineisto. BBC. <<https://www.bbc.com/news/health-39955295#comments>>. Päivitetty 19.5.2017. Luettu 12.6.2022.
- 36 Tsukayama, Hayley. 2016. Instagram will soon let you filter comments on your own account. Verkkoaineisto. The Washington Post. <<https://www.washingtonpost.com/news/the-switch/wp/2016/07/29/instagram-will-soon-let-you-filter-comments-on-your-own-account/>>. Päivitetty 29.7.2016. Luettu 12.6.2022.
- 37 Kastrenakes, Jacob. 2016. Instagram is building the anti-harassment tools Twitter won't. Verkkoaineisto. The Verge.

- <<https://www.theverge.com/2016/7/29/12327312/instagram-anti-harassment-commenting-tools>>. Päivitetty 29.7.2016. Luettu 12.6.2022.
- 38 What Is the Difference Between Facebook and Instagram? 2021. Verkkoaineisto. SocialBee. <<https://socialbee.io/blog/difference-between-facebook-and-instagram/#:~:text=Differences%20Summary&text=Facebook%20allows%20you%20to%20post,t%20be%20shared%20on%20Instagram.>>. Päivitetty 20.6.2021. Luettu 12.6.2022.
- 39 Gil, Paul. 2021. What Is Twitter & How Does It Work? Verkkoaineisto. Lifewire. <<https://www.lifewire.com/what-exactly-is-twitter-2483331>>. Päivitetty 29.8.2021. Luettu 12.6.2022.
- 40 Feiner, Lauren. 2022. Twitter accepts Elon Musk's buyout deal. Verkkoaineisto. CNBC LLC. <<https://www.cnbc.com/2022/04/25/twitter-accepts-elon-musks-buyout-deal.html>>. Päivitetty 25.4.2022. Luettu 12.6.2022.
- 41 Hanse, Derek; Shneiderman, Ben & Smith, Marc. 2011. Analyzing Social Media Networks with NodeXL: Insights from a Connected World. E-kirja. Elsevier Science & Technology.
- 42 Brown, Amanda. 2022. The tricky business of business tweeting. Verkkoaineisto. The Irish Times DAC. <<https://www.irishtimes.com/life-and-style/people/the-tricky-business-of-business-tweeting-1.561138>>. Päivitetty 2.3.2022. Luettu 12.6.2022.
- 43 Porter, Jon. 2020. Twitter quote tweets are now easier to find. Verkkoaineisto. The Verge. <<https://www.theverge.com/2020/9/1/21409925/twitter-quote-tweets-counter-retweet-with-comments-interface-ratiod>>. Päivitetty 1.9.2020. Luettu 12.6.2022.
- 44 Garg, Bhanu. 2022. Why Twitter is Better than Other Social Media Website. Verkkoaineisto. Digital Marketing Deal. <<https://digitalmarketing-deal.com/blog/why-twitter-is-better-than-other-social-media/>>. Päivitetty 15.1.2022. Luettu 14.6.2022.
- 45 Ingram, Mathew. 2018. Do journalists pay too much attention to Twitter? Verkkoaineisto. Columbia Journalism Review. <[https://www.cjr.org/the\\_media\\_today/journalists-on-twitter-study.php](https://www.cjr.org/the_media_today/journalists-on-twitter-study.php)>. Päivitetty 10.10.2018. Luettu 14.6.2022.
- 46 Winder, Davey. 2020. Twitter's Powerful Move Silences 175,000 Chinese And Russian Fake News Accounts. Verkkoaineisto.

- <<https://www.forbes.com/sites/daveywinder/2020/06/12/twitters-powerful-move-silences-175000-chinese-and-russian-fake-news-accounts-hong-kong-politics-social-media-disinformation/?sh=3263de7921c9>>. Päivitetty 12.6.2020. Luettu 14.6.2022.
- 47 Sharding & IDs at Instagram. 2012. Verkkoaineisto. Instagram Engineering. <<https://instagram-engineering.com/sharding-ids-at-instagram-1cf5a71e5a5c>>. Päivitetty 30.12.2012. Luettu 4.10.2022.
- 48 Selonen, Juha. 2020. Responsiivinen mobiilisivusto CSS-ohjelmistokehyksellä. Insinööriyö. Metropolia Ammattikorkeakoulu. Theseus-tietokanta.
- 49 BuiltWith APIs Social Media. 2022. Verkkoaineisto. Google Drive. <<https://drive.google.com/drive/u/2/folders/1b650JRlZElARTftU-eFobulYPDI6f6qD>>. Katsottu 15.6.2022.
- 50 What are tags, and how should I use them? 2022. Verkkoaineisto. Stackoverflow. <<https://stackoverflow.com/help/tagging>>. Luettu 15.6.2022.
- 51 Sarawagi, Shivang. 2022. What is Facebook SDK & API? Explained In Depth. Verkkoaineisto. Scale Your App. <<https://scaleyourapp.com/what-is-facebook-sdk-api-explained-in-depth/>>. Päivitetty 9.8.2022. Luettu 27.9.2022.
- 52 Questions tagged [facebook-graph-api]. 2022. Verkkoaineisto. Stackoverflow. <<https://stackoverflow.com/questions/tagged/facebook-graph-api?sort=Newest&edited=true>>. Päivitetty 15.6.2022. Luettu 15.6.2022.
- 53 Questions tagged [instagram-api]. 2022. Verkkoaineisto. Stackoverflow. <<https://stackoverflow.com/questions/tagged/instagram-api?sort=Newest&edited=true>>. Päivitetty 15.6.2022. Luettu 15.6.2022.
- 54 Questions tagged [twitter-api]. 2022. Verkkoaineisto. Stackoverflow. <<https://stackoverflow.com/questions/tagged/twitter%20api?sort=Newest&edited=true>>. Päivitetty 15.6.2022. Luettu 15.6.2022.
- 55 Questions tagged [tumblr-api]. 2022. Verkkoaineisto. Stackoverflow. <<https://stackoverflow.com/questions/tagged/tumblr%20api?sort=Newest&edited=true>>. Päivitetty 15.6.2022. Luettu 15.6.2022.
- 56 Questions tagged [tiktok-api]. 2022. Verkkoaineisto. Stackoverflow. <<https://stackoverflow.com/questions/tagged/tiktok%20api?sort=Newest&edited=true>>. Päivitetty 15.6.2022. Luettu 15.6.2022.



- 57 Questions tagged [snapchat-api]. 2022. Verkkoaineisto. Stackoverflow. <<https://stackoverflow.com/questions/tagged/snapchat%20api?sort=Newest&edited=true>>. Päivitetty 15.6.2022. Luettu 15.6.2022.
- 58 Top 10 Mobile App Development Frameworks in 2022. 2022. Verkkoaineisto. Sparity. <<https://www.sparity.com/top-10-mobile-app-development-frameworks-in-2022/>>. Päivitetty 9.6.2022. Luettu 7.7.2022.
- 59 Mobile Application Market Size, Share, & Trends Analysis Report By Store Type (Google Store, Apple Store, Others), By Application, By Region, And Segment Forecasts, 2022 - 2030. 2022. Verkkoaineisto. Grand View Research, Inc. <<https://www.grandviewresearch.com/industry-analysis/mobile-application-market>>. Luettu 7.7.2022.
- 60 Kapoor, Ajay. 2022. Native vs. Hybrid vs. Cross-Platform: Which is Right for You in 2022? Verkkoaineisto. Enclear Academy. <<https://enlear.academy/native-vs-hybrid-vs-cross-platform-which-is-right-for-you-in-2022-84a4adc0633c>>. Päivitetty 6.7.2022. Luettu 1.8.2022.
- 61 Almeida, Joana. 2022. What is Hybrid App Development? The Ultimate Guide. Verkkoaineisto. DistantJob. <<https://distantjob.com/blog/hybrid-app-development/>>. Päivitetty 17.5.2022. Luettu 1.8.2022.
- 62 Vailshery, Lionel. 2022. Cross-platform mobile frameworks used by software developers worldwide from 2019 to 2021. Verkkoaineisto. Statista. <<https://www.statista.com/statistics/869224/worldwide-software-developer-working-hours/>>. Päivitetty 21.2.2022. Luettu 7.7.2022.
- 63 Flutter vs React Native vs Native: Deep Performance Comparison. Verkkoaineisto. 2020. InVerita <<https://medium.com/swlh/flutter-vs-react-native-vs-native-deep-performance-comparison-990b90c11433>>. Päivitetty 29.6.2020. Luettu 29.7.2022.
- 64 Fanchi, Christopher. 2022. Flutter vs. React Native: Which is Better in 2022. Verkkoaineisto. Backendless. <<https://backendless.com/flutter-vs-react-native-which-is-better-in-2022/#:~:text=What%20is%20React%20Native%3F,iOS%20from%20a%20single%20codebase>>. Päivitetty 25.1.2022. Luettu 31.7.2022.
- 65 Barcia, Roland. 2022. Apache Cordova: Develop mobile apps with HTML, CSS, and JavaScript. Verkkoaineisto. IBM. <[https://www.ibm.com/garage/method/practices/code/tool\\_apache\\_cordova/](https://www.ibm.com/garage/method/practices/code/tool_apache_cordova/)>. Päivitetty 25.1.2022. Luettu 31.7.2022.

- 66 Gonzales, Robbie. 2017. Figma Wants Designers to Collaborate Google-Docs Style. Verkkoaineisto. Wired. <<https://www.wired.com/story/figma-updates/>>. Päivitetty 25.7.2017. Luettu 29.7.2022.
- 67 Welcome to BoardGameGeek? Verkkoaineisto. Board Game Geek. <[https://boardgamegeek.com/wiki/page/Welcome\\_to\\_BoardGameGeek](https://boardgamegeek.com/wiki/page/Welcome_to_BoardGameGeek)>. Luettu 8.8.2022.
- 68 Add Firebase to your Flutter app. 2022. Verkkoaineisto. Google. <<https://firebase.google.com/docs/flutter/setup?platform=ios>>. Päivitetty 25.7.2022. Luettu 29.7.2022.
- 69 Thomas, Alan. 2022. CI/CD For flutter using GitHub actions and Fastlane. Verkkoaineisto. Level Up Coding. <<https://levelup.gitconnected.com/ci-cd-for-flutter-using-github-actions-and-fastlane-6dfc9431ee9a>>. Päivitetty 3.2.2022. Luettu 3.8.2022.
- 70 Understanding GitHub Actions. 2022. Verkkoaineisto. GitHub Docs. <<https://docs.github.com/en/actions/learn-github-actions/understanding-github-actions>>. Luettu 3.8.2022.
- 71 Build and release an Android app. 2022. Verkkoaineisto. Flutter. <<https://docs.flutter.dev/deployment/android#signing-the-app>>. Päivitetty 3.22.2022. Luettu 3.8.2022.
- 72 Add Firebase to your Flutter app. 2022. Verkkoaineisto. Firebase. <<https://firebase.google.com/docs/flutter/setup?platform=android/>>. Päivitetty 29.9.2022. Luettu 5.10.2022.
- 73 Bogg, Fredrik. 2021. Tinder-App-Flutter. Verkkoaineisto. GitHub. <<https://github.com/dgewe/Tinder-App-Flutter>>. Päivitetty 23.6.2021. Luettu 8.8.2022.
- 74 Milke, Johannes. 2021. Flutter Tutorial - Tinder Swipe Cards | The Right Way [2021] 2/2 Flutter Dating App UI. Verkkoaineisto. YouTube. <[https://www.youtube.com/watch?v=BniNWBF8NH8&ab\\_channel=JohannesMilke](https://www.youtube.com/watch?v=BniNWBF8NH8&ab_channel=JohannesMilke)>. Päivitetty 18.10.2021. Luettu 8.8.2022.
- 75 Scott, Craig & Qian, Hua. 2007. Anonymity and Self-Disclosure on Weblogs. Journal of Computer-Mediated Communication. Vol. 12, s. 1428–1451.