



**KAUNO TECHNOLOGIJOS UNIVERSITETAS**  
**INFORMATIKOS FAKULTETAS**

**Tomas Kasparavičius**

**ANDROID ĮRENGINIŲ VIETOS SLĖPIMO GALIMYBIŲ**  
**TYRIMAS**

Baigiamasis magistro projektas

**Vadovas**

doc. dr. Jonas Čeponis

**KAUNAS, 2018**

**KAUNO TECHNOLOGIJOS UNIVERSITETAS**  
**INFORMATIKOS FAKULTETAS**

**ANDROID ĮRENGINIŲ VIETOS SLĖPIMO GALIMYBIŲ**  
**TYRIMAS**

Baigiamasis magistro projektas

**Informacijos ir informacinių technologijų sauga (kodas 621E10003)**

**Vadovas**

doc. dr. Jonas Čeponis

2018-05-22

**Recenzentas**

doc. Tomas Adomkus

2018-05-22

**Projektą atliko**

Tomas Kasparavičius

2018-05-22

**KAUNAS, 2018**



KAUNO TECHNOLOGIJOS UNIVERSITETAS  
INFORMATIKOS FAKULTETAS

---

(Fakultetas)

Tomas Kasparavičius

---

(Studento vardas, pavardė)

Informacijos ir informacinių technologijų sauga, 621E10003

---

(Studijų programos pavadinimas, kodas)

Baigiamojo projekto „Android įrenginių vietos slėpimo galimybių tyrimas“

**AKADEMINIO SAŽININGUMO DEKLARACIJA**

20 \_\_\_\_\_ m. \_\_\_\_\_ d.  
\_\_\_\_\_ Kaunas \_\_\_\_\_

Patvirtinu, kad mano, **Tomo Kasparavičiaus**, baigiamasis projektas tema „Android įrenginių vietos slėpimo galimybių tyrimas“ yra parašytas visiškai savarankiškai ir visi pateikti duomenys ar tyrimų rezultatai yra teisingi ir gauti sąžiningai. Šiame darbe nei viena dalis nėra plagijuota nuo jokių spausdintinių ar internetinių šaltinių, visos kitų šaltinių tiesioginės ir netiesioginės citatos nurodytos literatūros nuorodose. Įstatymų nenumatytų piniginių sumų už šį darbą niekam nesu mokėjęs.

Aš suprantu, kad išaiškėjus nesąžiningumo faktui, man bus taikomos nuobaudos, remiantis Kauno technologijos universitete galiojančia tvarka.

---

(vardą ir pavardę įrašyti ranka)

---

(parašas)

Kasparavičius, Tomas. „Android įrenginių vietos slėpimo galimybių tyrimas“. Magistro baigiamasis projektas / vadovas doc. dr. Jonas Čeponis; Kauno technologijos universitetas, Informatikos fakultetas.

Mokslo kryptis ir sritis: Informacijos ir informacinių technologijų sauga

Reikšminiai žodžiai: *GPS, Android, programa, vieta, slėpimas.*

Kaunas, 2018. 54 p.

## **SANTRAUKA**

Šiame tiriamajame darbe analizuojamos išmaniųjų įrenginių vietos slėpimo galimybės. Pristatomi išmaniųjų įrenginių koordinacių slėpimo būdai.

Tyrimas remiasi atlikta esamų programų, GPS koordinacių nustatymo ir klastojimo būdų analize. Analizės metu išskirti pagrindiniai vietos slėpimo, išmaniuosiuose įrenginiuose, būdai, atliktas jų palyginimas, išanalizuotos esamos programos. Remiantis analizės išvadomis suprojektuota išmaniųjų „Android“ įrenginių vietos slėpimo programa. Išmaniosios programos pagalba vartotojas gali realistiškai keisti įrenginio buvimo vietą taip paslepiant tikrąją, atlikti paiešką pagal konkretų vietos adresą, importuoti egzistuojančius maršrutus, gauti jų vidutinį greitį ir laiką. Eksperimento metu ištirtos sukurtos programos galimybės ir patikrintas jos veikimas.

Tiriamąjį darbo metu gauti rezultatai parodė, kad programa gali paslėpti arba pakeisti išmaniojo įrenginio buvimo vietą, automatiškai generuoti maršrutą su vartotojo nurodytais parametrais, tokiais kaip pradžios taškas, pabaigos taškas, greitis, tikslumas.

Kasparavičius, Tomas. *Research of location concealing for android devices*: Master's thesis in Information and Information Technologies Security / supervisor assoc. prof. Jonas Čeponis. The Faculty of Informatics, Kaunas University of Technology.

Research area and field: Information and Information Technologies Security.

Key words: *GPS, Android, Application, Device, Location, Mocking, Fake.*

Kaunas, 2018. 54 p.

## **SUMMARY**

This research explores the possibilities for hiding Android devices location. Introduces ways to hide the real coordinates of smart devices.

The study is based on an analysis of current programs GPS coordinates and counterfeiting techniques. The analysis highlights the main methods of hiding location of smart devices, comparing them, analyzing existing applications. Based on the findings of the analysis, new android application for location mocking has been designed. With the help of the application, the user can realistically change the location of the device by hiding the actual one, perform a search according to a specific local address, importing existing routes, obtaining their average speed and time. During the experiment, the possibilities of the created program were examined and its operation checked.

The results obtained during the research show that the program can hide or change the location of a smart device, automatically generate a route with user-specified parameters such as starting point, end point, speed, accuracy.

## TURINYS

LENTELIŲ SĄRAŠAS .....	8
PAVEIKSLŲ SĄRAŠAS .....	9
TERMINŲ IR SANTRUPŲ ŽODYNAS .....	10
ĮVADAS.....	11
1. ĮRENGINIŲ VIETOS NUSTATYMO GALIMYBIŲ ANALIZĖ .....	13
1.1. Analizės tikslas .....	13
1.2. Tyrimo objektas, sritis ir problema .....	13
1.3. Mobilųjų operacinių sistemų pasiskirstymas pasaulinėje išmaniųjų įrenginių rinkoje .....	13
1.4. „Android“ operacinė sistema .....	14
1.5. GPS koordinacių nustatymo būdai.....	14
1.6. GPS koordinacių klastojimo būdai .....	15
1.6.1. GPS API .....	15
1.6.2. GPS modulis .....	16
1.6.3. Išorinio serverio API .....	17
1.6.4. Imituojant išmanųjį įrenginį .....	18
1.7. GPS koordinacių klastojimų būdų palyginimas.....	18
1.8. Esamų problemos sprendimo metodų analizė.....	19
1.9. Siekiamo sprendimo apibrėžimas .....	23
1.10. Analizės išvados.....	24
2. IŠMANIŲJŲ ĮRENGINIŲ VIETOS SLĖPIMO PROGRAMOS SPRENDIMO REIKALAVIMŲ SPECIFIKACIJA IR PROJEKTAVIMAS .....	25
2.1. Reikalavimų specifikacija .....	25
2.1.1. „Hide Yourself“ programos panaudojimo atvejų diagrama .....	25
2.1.2. Funkciniai reikalavimai .....	28
2.1.3. Nefunkciniai reikalavimai .....	30
2.1.4. Techninė specifikacija .....	31
2.2. Dalykinės srities modelis .....	31

2.3. Naudotojų sąsajos modelis.....	32
2.4. Reikalavimų apibendrinimas .....	33
3. IŠMANIŪJŲ ĮRENGINIŲ VIETOS SLĖPIMO REALIZACIJA.....	34
3.1. Sprendimo pagrindimas ir esminiai principai .....	34
3.2. Išmaniųjų įrenginių vietos slėpimo įrankio architektūra .....	34
3.3. Duomenų modelis .....	37
3.4. Sistemos elgsenos modelis.....	39
3.5. Programinių komponentų architektūra .....	42
3.6. Realizavimo dalies išvados .....	42
4. EKSPERIMENTINIS IŠMANIŪJŲ ĮRENGINIŲ VIETOS SLĖPIMO SPRENDIMO TYRIMAS .....	43
4.1. Eksperimento planas .....	43
4.2. Eksperimento rezultatai .....	45
4.3. Sprendimo veikimo ir savybių analizė, kokybės kriterijų įvertinimas .....	48
4.4. Eksperimento apibendrinimas.....	49
5. REZULTATŲ APIBENDRINIMAS IR IŠVADOS.....	50
6. LITERATŪRA.....	52
7. PRIEDAI.....	55
7.1. Priedas. Testavimo maršruto ekrano kopijos „Run Tracker“ programoje.....	55
7.2. Priedas. Testavimo maršruto ekrano kopijos „LocationTracker“ programoje .....	55
7.3. Priedas. Testavimo maršruto ekrano kopijos „Glympse“ programoje.....	56
7.4. Priedas. Naujo testavimo maršruto ekrano kopijos „Hide Yourself“ programoje.....	56
7.5. Priedas. Naujo testavimo maršruto ekrano kopijos „Google Maps“ programoje.....	57
7.6. Priedas. Naujo testavimo maršruto ekrano kopijos „Run Tracker“ programoje .....	57

## LENTELIŲ SĄRAŠAS

1.1 lentelė. Esamų metodų palyginimas.....	19
1.2 lentelė. Esamų sprendimų palyginimas.....	23
2.1 lentelė. Naujos vietos pasirinkimo panaudojimo atvejo lentelė.....	29
2.2 lentelė. Naujo maršruto sukūrimo panaudojimo atvejo lentelė.....	29
2.3 lentelė. Išsaugotos vietos pasirinkimo panaudojimo atvejo lentelė .....	29
2.4 lentelė. Vietos išsaugojimo į mėgstamiausių sąrašą panaudojimo atvejo lentelė .....	29
2.5 lentelė. Maršruto įkėlimo panaudojimo atvejo lentelė.....	29
2.6 lentelė. Maršruto pasirinkimo panaudos atvejo lentelė.....	30
2.7 lentelė. Maršruto analizės panaudos atvejo lentelė .....	30
2.8 lentelė. Pozicijos keitimo įjungimo panaudos atvejo lentelė .....	30
2.9 lentelė. Nefunkcinių reikalavimų <i>Volere</i> šablono lentelė .....	30
4.1 lentelė. Programos atitikimo palyginimo lentelė .....	48



## PAVEIKSLŲ SĄRAŠAS

1.1 pav. Išmaniųjų įrenginių operacinių sistemų rinkos dalis pasauliniu mastu.....	13
1.2 pav. 3-čios šalies programos vietos keitimo veikimo principas.....	16
1.3 pav. Modifikuotos operacinės sistemos vietos keitimo veikimo principas.....	17
1.4 pav. Įgaliotojo serverio veikimo principas.....	18
1.5 pav. FakeGPS Free programos langas .....	20
1.6 pav. FakeGPS programos langas .....	21
1.7 pav. Mock GPS PRO programos langas .....	22
2.1 pav. Panaudojimo atvejų diagrama .....	25
2.2 pav. Panaudojimo atvejo „Slėpti įrenginio vietą“ veiklos diagrama .....	26
2.3 pav. Panaudojimo atvejo „Įkelti naują maršrutą“ veiklos diagrama.....	26
2.4 pav. Panaudojimo atvejo „Atlikti maršruto analizę“ veiklos diagrama .....	27
2.5 pav. Panaudojimo atvejo „Sukurti naują maršrutą“ veiklos diagrama.....	28
2.6 pav. Klasių diagrama.....	31
2.7 pav. Veikimo sekų diagrama.....	32
2.8 pav. „Hide Yourself“ pradinio vaizdo ir meniu vaizdo maketas .....	33
3.1 pav. Robastiškumo diagrama .....	34
3.2 pav. Loginė sistemos architektūra.....	35
3.3 pav. Maršruto valdymo posistemė .....	36
3.4 pav. Maršruto analizės posistemė .....	36
3.5 pav. Kelio taškai, keliai, takai užregistruoti GPS pagalba .....	37
3.6 pav. Panaudojimo atvejo „Pasirinkti maršrutą“ sekų diagrama .....	39
3.7 pav. Panaudojimo atvejo „Atlikti maršruto analizę“ sekų diagrama .....	40
3.8 pav. Panaudojimo atvejo „Peržiūrėti analizės rezultatus“ sekų diagrama .....	41
3.9 pav. „Slėpti įrenginio poziciją“ sekų diagrama.....	41
3.10 pav. „Hide Yourself“ programos diegimo modelis.....	42
4.1 pav. Programos pradinis langas .....	45
4.2 pav. Maršruto vietos pasirinkimas .....	46
4.3 pav. Maršruto analizė.....	46
4.4 pav. Vartotojo vietos imitavimas maršrutu funkcijos veikimas „Hide Yourself“ programoje.....	47
4.5 pav. Vartotojo vietos imitavimas maršrutu funkcijos veikimas „Maps“ programėlėje .....	48

## TERMINŲ IR SANTRUPŲ ŽODYNAS

API (angl. <i>Application Programming Interface</i> )	Programų sąsaja, kurią suteikia kompiuterinė sistema, biblioteka ar programa tam, kad programuotojas per kitą programą galėtų pasiekti jos funkcionalumą ar apsiektų su ja duomenimis.
Duomenys	Faktai ar teiginiai, kurie yra ar gali būti surinkti, išsaugoti, apdoroti, tačiau nėra organizuoti ar pateikti kontekste.
Emuliatorius (angl. <i>Emulator</i> )	Įrenginys ar programa, kuri priima tuos pačius duomenis bei komandas, kaip ir emuliuojamoji (pamėgdžijama) sistema, bei duoda analogiškus emuliuojamai sistemai rezultatus.
GPS (angl. <i>Global Positioning System</i> )	Leidžia nustatyti objekto koordinates bet kurioje pasaulio vietoje visomis oro sąlygomis, kuomet yra galimybė gauti signalus iš keturių ar daugiau palydovų. Sistemos pagrindas – IT technologijų sąveika su planetą gaubiančiu GPS palydovų tinklu.
Įgaliotasis serveris (angl. <i>Proxy</i> )	Serveris, esantis tarp kompiuterio arba vietinio tinklo ir išorinio tinklo. Per jį gali eiti visi kliento (vietinio tinklo) ryšiai su išoriniu pasauliu. Kai klientas nori užmegzti ryšį su kuria nors svetaine, jis kreipiasi į įgaliotąjį serverį ir pateikia jam tos svetainės adresą ir užklausa jai.
GPX (angl. <i>GPS Exchange Format</i> )	Atviras, XML dokumento struktūra apibūdintas formatas keistis GPS duomenimis.

## ĮVADAS

2011 m., kai paaiškėjo, kad „Apple“ savo įrenginių savininkų buvimo vietos duomenis seka net tuomet, kai jie išjungia tokių duomenų perdavimą, buvo kilęs visuomenės sujudimas dėl asmeninių duomenų saugumo užtikrinimo [1]. Vėliau panašiai buvo kaltinama ir „Android“ sistema. Taip pat yra daug populiarių programėlių, tokių kaip „Facebook“, „FourSquare“, ir pan., kurios nuolatos seka vartotojo įrenginio vietą. Pačios kompanijos teigia, kad toks sekimas joms padeda kuo geriau tarnauti savo vartotojams – pateikti naudingas nuorodas, išmanius pasiūlymus, išsaugoti dažniausiai lankomų vietų adresus. Gal ir taip, tačiau ne visi to nori, ir galima imtis atsargumo priemonių. Viena iš išeičių yra naudoti vietos slėpimo programą savo išmaniajame įrenginyje.

Privatumas šiandieniniame pasaulyje tampa vis aktualesne tema. Ši problema auga, nes daugėja išmaniųjų įrenginių, kurie lengvai gali pasiekti ir parodyti vartotojo buvimo vietą, veiklą ar įpročius. Nuo 2015 metų rugsėjo priskaičiuojama daugiau nei 1,5 milijardai aktyvių „Android“ įrenginių [2]. Šie įrenginiai būna arčiausiai vartotojo didžiąją dalį dienos. Išmanusis įrenginys leidžia parsisiųsti ir įsidiegti trečiosios šalies programėles, daugeliui net nesusimąstant apie tai, kokias operacijas parsisiųsta programa atlieka vartotojo išmaniajame įrenginyje. Vienas iš tokių neleistinių veiksmų yra vartotojo vietos sekimas. Siekiant to išvengti šiame darbe bus siekiama sukurti „Android“ vietos slėpimo programą.

### **Darbo problematika ir aktualumas**

Išmaniuosiuose įrenginiuose esančios techninės priemonės leidžia stebėti vartotojų buvimo vietą dažnai jiems apie tai net nežinant. Siekiant apsisaugoti nuo sekimo, reikalingos priemonės, padedančios vartotojams savo nuožiūra pakeisti tikrąją įrenginio buvimo vietą.

### **Darbo tikslas ir uždaviniai**

Tikslas: Leisti vartotojui paslėpti arba savo nuožiūra pakeisti išmaniojo įrenginio vietos (koordinatų) duomenis.

Uždaviniai:

- išanalizuoti įrenginiuose vietos nustatymo sistemų duomenų keitimo arba slėpimo galimybes;
- suprojektuoti programinę įrangą įrenginiams, kurie leistų realistiškai klastoti jų vietą kitoms programoms;
- realizuoti programinę įrangą vietos nustatymo sistemų duomenų slėpimui arba keitimui kitoms programoms;
- eksperimento metu patikrinti sukurtą programinę įrangą.

## **Darbo rezultatai ir jų svarba**

Išmanojo įrenginio buvimo vietos slėpimo programa kuriama taip, kad potencialus vartotojas galėtų ją įsidiegti į savo „Android“ išmanųjį įrenginį. Pagrindinis programos tikslas – pritaikant pagrindinius buvimo vietos slėpimo algoritmus garantuoti vartotojo vietos nuslėpimą, tai darant kuo realistiškiau, t. y. imituojant žmogaus judėjimą pagal tam tikrą maršrutą. Tokiu būdu būtų dar sunkiau susekti tikrąją išmanaus įrenginio savininko poziciją.

Buvimo vietos slėpimo programos tinkamumui įvertinti buvo atliktas eksperimentas, kurio metu buvo išbandyti skirtingi maršrutai ir sekama esama buvimo pozicija su tam tikromis programėlėmis, pvz., „Facebook“, „Messenger“, „Google Maps“.

## **Darbo struktūra**

– Pirmajame tiriamojo darbo skyriuje apibrėžta tyrimo sritis, objektas ir problema, atlikta panašių buvimo vietos slėpimo programų analizė, aptarti pagrindiniai komponentai ir jų savybės, pateikti pavyzdžiai. Susipažinta su vietos slėpimo principais. Remiantis analize išskirti svarbiausi vietos slėpimo elementai, remiantis jais suformuotos siekiamos programos veikimo galimybės. Atlikta esamų įrankių analizė ir palyginimas. Suformuota būsimos sistemos vizija ir veikimo principai.

– Antrajame skyriuje pateikti vietos slėpimo įrankio funkciniai ir nefunkciniai reikalavimai, panaudojimo atvejų diagrama, klasių modelis, apibrėžtos ir aprašytos pagrindinės vietos slėpimo taisyklės.

– Trečiajame skyriuje atlikta reikalavimų analizė, sudaryta loginė sistemos architektūra. Sekų diagramomis atvaizduotas sistemos elgsenos modelis, veiklos diagrama parodytas vietos slėpimo algoritmo veikimas. Panaudojimo atvejai realizuoti klasių diagramomis, aprašyti metodai, kurie naudojami pagrindiniuose įrankio valdikliuose. Sudaryta programinių komponentų architektūra ir diegimo modelis.

– Ketvirtajame skyriuje apibendrinti gauti rezultatai, pateiktas sistemos veikimo aprašymas, aprašyti galimi maršrutų pasirinkimai. Pateikti eksperimento metu gauti rezultatai.

# 1. ĮRENGINIŲ VIETOS NUSTATYMO GALIMYBIŲ ANALIZĖ

## 1.1. Analizės tikslas

Šios analizės tikslas yra išsiaiškinti išmaniųjų įrenginių vietos slėpimo ir keitimo metodus, perprasti egzistuojančias GPS ar kitų vietos nustatymo sistemų duomenų keitimo ir slėpimo programas, suformuoti kuriamos programos viziją ir aprašyti analizės išvadas.

Siekiant atlikti išsamų tyrimą analizuojami pasirinktos dalykinės srities literatūros šaltiniai. Problemos sprendimui įgyvendinti nagrinėjami ne tik pozicijos slėpimo metodai ir būdai, bet ir esamos išmaniosios programėlės, kurios padeda paslėpti įrenginio vietą.

## 1.2. Tyrimo objektas, sritis ir problema

Pagrindinis tyrimo objektas yra pozicijos slėpimo metodo išgryninimas ir vietos slėpimo „Android“ įrenginiui programėlės sukūrimas.

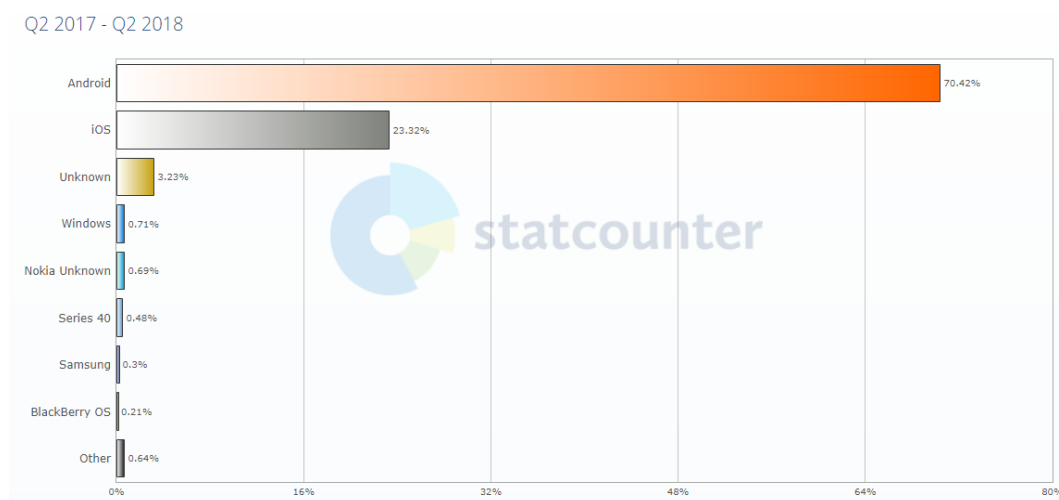
Magistro darbo tyrimo sritis yra išmaniųjų įrenginių keitimo ir slėpimo būdai.

## 1.3. Mobilųjų operacinių sistemų pasiskirstymas pasaulinėje išmaniųjų įrenginių rinkoje

Remiantis *statcounter.com* [3] ir *netmarketshare.com* [4] svetainių statistiniais duomenimis nuo 2017 iki 2018 metų „Android“ yra populiariausia operacinė sistema išmaniesiems įrenginiams. Pasak *netmarketshare.com* didžiausią rinkos dalį mobiliųjų įrenginių operacinių sistemų rinkoje užima „Android“ (69,46 % [4]), po jos rikiuojasi „iOS“ (29,04 % [4]) ir kitos ne tokios žinomos operacinės sistemos (1,5 % [4]). Dėl didelio vartotojų skaičiaus ir mobiliųjų įrenginių rinkos dalies, programai kurti pasirinkta „Android“ operacinė sistema.

Operacinių sistemų palyginimas remiantis *statcounter.com* turimais duomenimis atvaizduojamas

### 1.1 paveiksle [3].



1.1 pav. Išmaniųjų įrenginių operacinių sistemų rinkos dalis pasauliniu mastu

#### 1.4. „Android“ operacinė sistema

„Android“ yra kompanijos Google sukurta operacinė sistema. Kai pirmą kartą Google išigijo „Android“ jų misija buvo sukurti atvirojo kodo platformą, kuri leidžia kūrėjams skelbti savo sukurta programas (angl. *Applications*). Kitaip nei kiti gamintojai, Google leido kiekvienam įkelti programą, kurią galėjo atsisiųsti kiti platformos vartotojai. Tačiau prieš vartotojui pradėdant naudotis atsisiūsta programa jis turi leisti programai prieiti prie tokių telefono funkcijų kaip GPS. Remiantis atliktais tyrimais [5], didelė žmonių dalis neskaito arba nesupranta teisių, kurių paprašė programa. Be to, „Android“ neleidžia rankiniu būdu pakeisti programų leidimų, todėl jei vartotojas nori naudotis parsisiūsta programa, turi sutikti su jos sąlygomis. Kiekviena paleista programa gali veikti fone su neapribotais leidimais ir naudotis telefone esančiais duomenimis [6], [7].

Programos „Android“ įrenginyje yra atskiriamos viena nuo kitos naudojant „Dalvik“ virtualiąją mašiną (angl. *Dalvik Virtual Machine*). DVM apsaugo nuo kenkėjiškų programų, kurios bando įsilaužti ar pakenti „Android“ operacinei sistemai. Siekiant užtikrinti patogų programų kūrimą Google sukūrė API, kuris leidžia lengvai kontroliuoti sistemos resursus. Duomenys „Android“ įrenginyje gali būti dviejų rūšių: statiniai (pvz.: el. pašto adresai, vardai, kontaktai, skambučiai, trumposios žinutės ir pan.) ir dinaminiai (pvz.: IP adresas, įrenginio vieta, barometras, užduotys ir pan.) [8].

#### 1.5. GPS koordinačių nustatymo būdai

Pagrindiniai GPS koordinačių gavimo būdai [9]:

1. GPS – įrenginio vieta, nustatoma palydovais. Operacija užtrunka apie 7 sekundes ir jai atlikti reikalingas GPS modulis, kurį naudojant nustatoma vieta apytiksliai 5 metrų spinduliu.
2. Tinklo – įrenginio vieta nustatoma panaudojant telefono ryšio bokštelių ir interneto prieigos taškus (angl. *Wi-Fi*), tai nėra labai tikslu, vieta nustatoma apytiksliai 60 metrų spinduliu. Šis būdas taip pat reikalauja GPS modulio, tačiau vietos nustatymui nenaudojami palydovai.
3. Pasyvus - vieta nustatoma tik panaudojant telefono ryšio bokštelių ir interneto prieigos taškus. Nereikalingas GPS modulis, bet dėl to vietos nustatymo tikslumas yra nuo 1 iki 1.5 km.

Apžvelgus koordinačių nustatymo būdus išmaniosios programos kūrimui nuspręsta panaudoti vietos keitimą keičiant GPS koordinates, kadangi jos yra tiksliausios ir jų paslėpimas kitoms programoms yra aktualiausias.

## 1.6. GPS koordinatų klastojimo būdai

Pagrindiniai GPS koordinatėms klastoti naudojami būdai [10]:

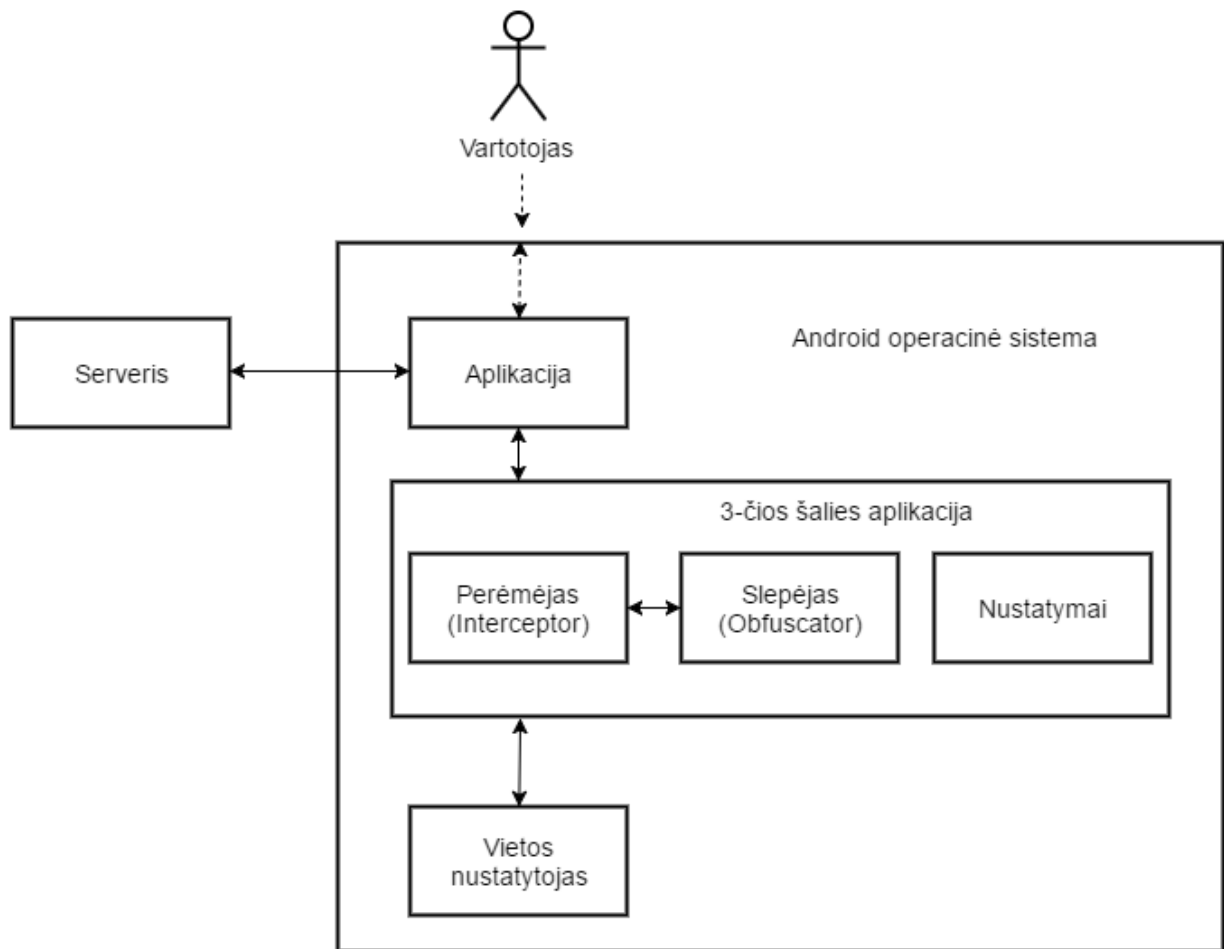
1. GPS API;
2. GPS modulis;
3. išorinio serverio API;
4. imituojant išmanųjį įrenginį.

Kiekvieno GPS koordinatų būdo aprašymas pateikiamas tolimesniuose skyriuose.

### 1.6.1. GPS API

Vietos klastojimas naudojant GPS API - metodas koreguoja API, kurios išmaniajame įrenginyje panaudoja esamas GPS koordinates ir grąžina suklastotą įrenginio vietą. Tai vienas iš paprasčiausių būdų, nes atvirojo kodo programos, kurias naudoja operacinės sistemos, pvz.: „Android“, yra paplitusios tarp vartotojų [10], [11]. Programos gali būti modifikuotos siekiant gauti GPS vietą iš kitų šaltinių nei telefone naudojamas GPS modulis, pavyzdžiui iš serverio, kuris grąžina netikrą GPS koordinates arba vietinio failo. Šis metodas yra ypač paplitęs atviro kodo sistemose, nes yra universalus ir lengvai pritaikomas. Vienas iš GPS API pavyzdžių gali būti, kai į jau egzistuojančią sistemą įrašoma 3-čios šalies programa (TPP). Ši programa įsiterpia tarp GPS nustatymo tarnybos (angl. *Service*) ir programos, norinčios pozicijos. TPP komponentas gali pakeisti arba paslėpti vietos koordinates naudojant tam tikrus algoritmus. Kadangi visos vietos nustatymo užklausos žinutės yra identiškos, jos yra atpažįstamos ir nusiunčiamos į TPP. Šis procesas yra nematomas serveriui [12].

1.2 paveikslėlyje matome, kad vartotojo naudojama programa nori gauti įrenginio vietą iš vietos nustatytojo (angl. *Location provider*), bet operacinėje sistemoje yra įrašyta 3-čios šalies programa, kuri perima vietos nustatymo užklausą. Ši programa siunčia naują esamos pozicijos užklausą vietos nustatytojui ir gauna tikrą informaciją. Pozicijos informacija programoje yra keičiama arba nekeičiama pagal nustatymus. Šie pozicijos duomenys yra grąžinami pirmajai programai [13].



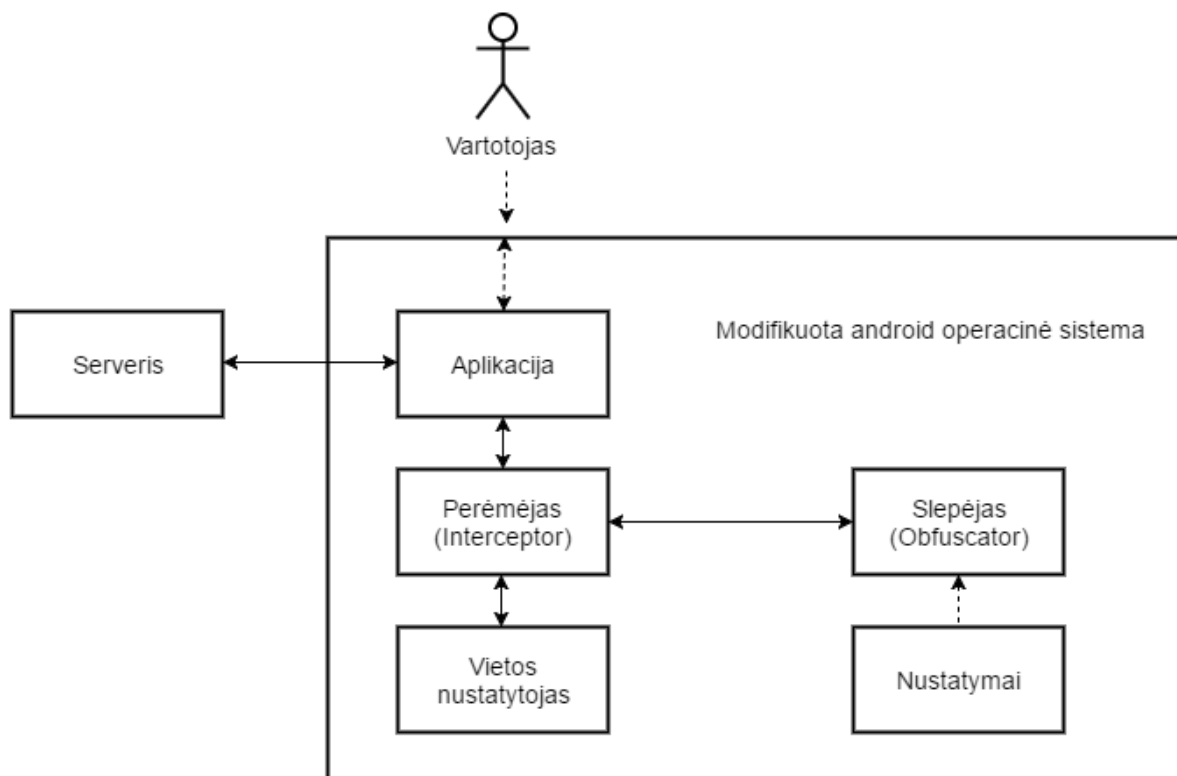
1.2 pav. 3-čios šalies programos vietos keitimo veikimo principas

### 1.6.2. GPS modulis

Vienas iš kelių būdų atlikti vietos klastojimą naudojant GPS modulį yra tiesioginis įsilaužimas į išmaniojo įrenginio GPS modulį. GPS vietovės pakeitimui naudojant techninę arba programinę įrangą. Įrenginio vieta koreguojama sugadinant arba pakeičiant GPS siųstuvo nustatymus. Pavyzdžiui, užpuolikas gali parašyti programą, kuri imituoja Bluetooth GPS imtuvą ir leidžia išmaniajam įrenginiui prisijungti prie minėto prietaiso, taip įgalinant GPS įrenginį grąžinti netikras koordinatas. Rinkoje yra panašaus pobūdžio įrankių (pvz.: Skylab GPS Simulator, Zyl Soft, GPS Generator Pro), kurie buvo sukurti tam, kad būtų lengviau ištaisyti su GPS susijusias problemas, tačiau gali būti panaudoti ir vietos klastojimui [10].

Taip pat pakeitus „Android“ operacinės sistemos karkasą, galima keisti vietos nustatymo tikslumą, privatumo nustatymus. Toks GPS modulio keitimas leidžia nustatyti skirtingas privilegijas skirtingoms programoms. Operacinė sistema leidžia tam tikroms programoms duoti netikras privilegijas pvz.: matyti vartotojo įrenginio vietą, nors iš tikrųjų pastaroji programai yra nerodoma. [14]



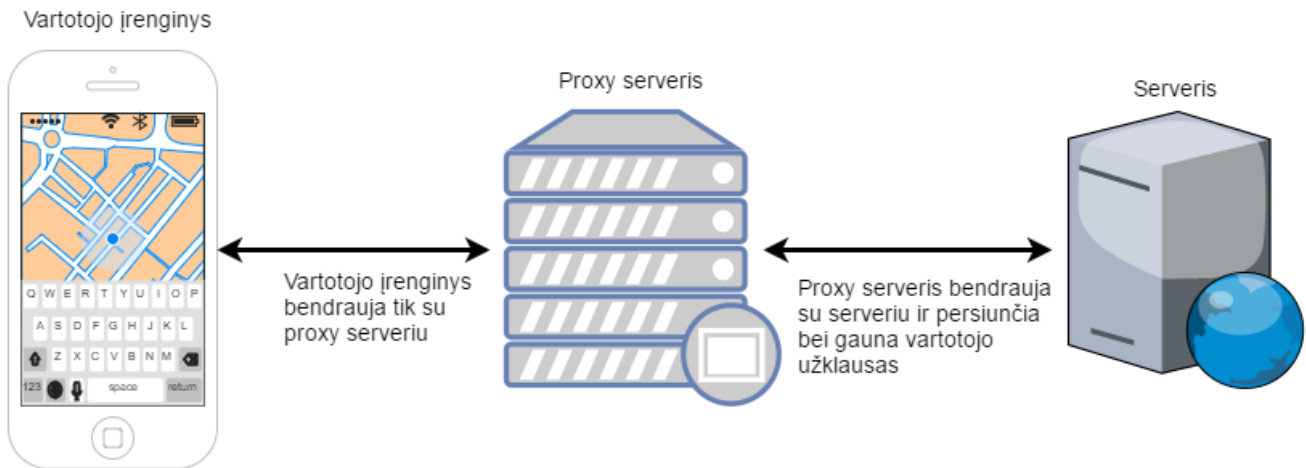


**1.3 pav. Modifikuotos operacinės sistemos vietos keitimo veikimo principas**

1.3 paveikslėlyje matome, kad vartotojo naudojama programa nori gauti įrenginio vietą iš vietos nustatytojo (angl. *Location provider*), bet modifikuotoje operacinėje sistemoje yra papildomas komponentas, kuris perima vietos nustatymo užklausa ir atmintyje išsaugoja unikalų programos ID. Perėmėjas siunčia naują esamos pozicijos užklausa vietos nustatytojui ir gauna tikrą informaciją. Šią informaciją kartu su programos ID siunčia slėpėjui (angl. *Obfuscator*), kuris pagal nustatymus šiai programai keičia arba nekeičia vietos informacijos. Viskas grąžinama perėmėjui, kuris atitinkamai duomenis perduoda programai.

### 1.6.3. Išorinio serverio API

Programinė įranga „Foursquare“ programuotojui leidžia sukurti naują programą pvz.: nuotraukų, kurioje jos buvo nufotografuotos, žymėjimo API. Tokio pobūdžio programos gali būti naudojamos vartotojų, kurie nori klastoti savo geografinę padėtį nufotografuodami nuotrauką ir panaudodami minėtąją programą fotografijos koordinatėms pakeisti. Vienas iš tokių pavyzdžių yra įgaliojasis serveris [10].



**1.4 pav. Įgaliotojo serverio veikimo principas**

Įgaliotojo serverio naudojimas pavaizduotas 1.4 paveikslėlyje [15]. Matome, kad vartotojas užklaudas siunčia į įgaliotąjį serverį, o jis jas persiunčia programos/puslapio serveriui. Atsakymai iš programos/puslapio serverio taip pat eina per įgaliotąjį serverį ir taip pasiekia vartotojo įrenginį. Kadangi taip yra daroma su visomis užklausomis, vartotojas lieka nematomas galiniam serveriui.

#### **1.6.4. Imituojant išmanųjį įrenginį**

Išmaniųjų įrenginių gamintojai leidžia programuotojams testuoti įrenginių emuliatorius, tam, kad pastarieji galėtų ištestuoti kuriamą produktą. Įrenginio emuliatorius yra pilnai veikianti šio įrenginio virtualioji mašina. Viena iš pagrindinių šio prietaiso savybių yra ta, kad jie yra konfigūruojami. Gali būti konfigūruojamas ir GPS modulis. Pavyzdžiui, paimkime „Android“ prietaiso emuliatorių. Naudodami jį galime nusiųsti specifinę komandą vietos nustatymui į imituojamą GPS modulį. Emuliatoriaus GPS modulis grąžins koordinates, kurias bus galima nusiųsti į bet kurį išmanųjį įrenginį, kuris naudoja GPS modulį [10].

#### **1.7. GPS koordinačių klastojimų būdų palyginimas**

Nors visi analizuoti metodai gali slėpti esamą vartotojo įrenginio vietą ir leidžia vartotojui saugoti privatumą, tačiau metodų funkcijos ir patogumas skiriasi, tai pavaizduota 1.1 lentelėje. Palyginimo metu buvo įvertintas pagrindinių metodų skirtumas pagal vietos nustatymą, išmanaus įrenginio prieinamumą paprastam vartotojui, o ne administratoriui, vietos slėpimą kai vietos nustatymo tarnyba yra įjungta/išjungta.

### 1.1 lentelė. Esamų metodų palyginimas

Palyginimo kriterijus	Išorinio serverio API	GPS modulis	GPS API	Imituojant išmanųjį įrenginį
Gali slėpti įrenginio vietą, kai vietos nustatymo tarnyba įjungta.	Ne	Taip	Taip	Taip
Reikalingos telefono administratoriaus privilegijos (root).	Nereikia	Reikia	Nereikia	Nereikia
Gali slėpti įrenginio vietą, kai vietos nustatymo tarnyba išjungta.	Taip	Taip	Taip	Taip

Naudojantis 1.1 lentele, įvertinus visus kriterijus, pasirinkta toliau analizuoti patikimos 3-čios šalies (TPP) programos variantą, nes tai suteikia didžiausią apsaugą, kuriai nereikia telefono administratoriaus privilegijų. Daugumai vartotojų šių privilegijų gavimas yra nepriimtinas arba iš vis neįmanomas, kadangi tai pažeidžia mobilaus įrenginio garantinio remonto sąlygas.

### 1.8. Esamų problemos sprendimo metodų analizė

Mokslinių straipsnių analizė parodė, kad yra darbų glaudžiai susijusių su GPS vietos klastojimu arba vartotojų elgesio analize naudojant GPS vietos sekimą [16]. Tačiau daugelio jų tyrimų sritis buvo pagrįsta tik vartotojo veiksmų imitavimu pasirinktoje vietoje. Kiti autoriai veiklos atpažinimui naudoja akcelerometrą, kuris skirtas matuoti pagreitį [17], [18]. Straipsniai leido apibūdinti kaip naudojant žmogaus judėjimo trajektoriją galima atpažinti vartotojo veiklą (pvz.: jogą, bėgimą, sėdėjimą, stovėjimą).

Didžioji dalis nagrinėtų straipsnių rėmėsi pagreičio duomenimis, kuriuos išmatuoja akcelerometru. Šiuo atveju pastarieji nesirėmė GPS duomenimis, tam, kad būtų sukurta reali individo judėjimo trajektorija. GPS vietos sekimas dažniausiai naudojamas šnipinėjimui arba vartotojo sekimui marketingo tikslais siekiant nuspėti jo veiklą ar pomėgius [19]. Tačiau išmaniųjų įrenginių vartotojai siekdami apsaugoti asmeninis duomenis pradėjo klastoti GPS vietos koordinates [20]. Šiam tikslui yra kuriamos vietos klastojimo programos, kurių tikslas atvaizduoti suklastotą išmaniojo įrenginio vietą, taip klaidinant kenkėjiškas programas.

Šio darbo tikslas yra realistiškai GPS vietą klastojančios programos sukūrimas, šiam tikslui buvo atlikta analizė, kurios metu pasirinktos penkios populiariausios (2018-01-10) Google programų

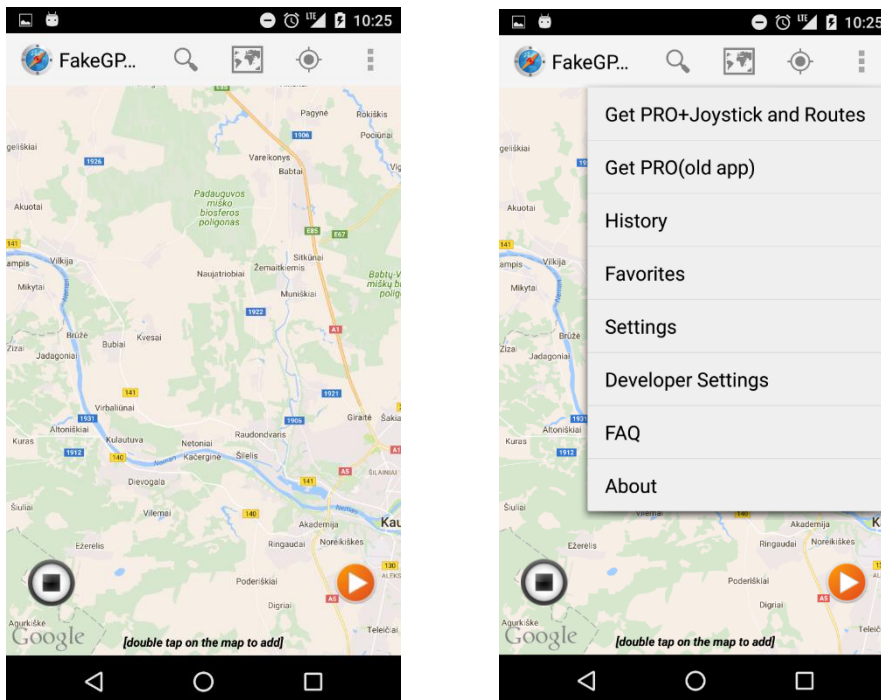
parduotuvės vietovės klastojimo programos t. y. FakeGps Free, FakeGPS PRO, Mock GPS PRO, Fake GPS ir Location Soofer. Analizės metu įvertinama programos kaina, funkcijų kiekis, papildomų maršrutų įkėlimo paslauga, teisių politika.

### **FakeGPS Free**

Vietos klastojimo programa FakeGPS geba perrašyti vartotojo esamą vietą, pagal klastojamą vietą suteikia galimybę rasti žmones aplink pasirinktą vietą, taip pat programa gali užpildyti Geotag duomenis nuotraukoje pamiršus įjungti savo GPS vietos atvaizdavimą fotografijos darymo metu [21].

Programa yra nemokama, todėl ja gali naudotis kiekvienas turintis išmanųjį įrenginį su Android operacine sistema. Prieš naudojant FakeGPS Free programa paprašo suteikti prieigą prie esamos įrenginio vietos, pastaroji nustatoma dviem būdais: apytikslė vieta (paremta tinklu) ir tiksli įrenginio vieta (paremta GPS ir tinklu). Taip pat programai turi būti užtikrinta prieiga prie internetinio tinklo ryšio, sistemos nustatymų, klastotos vietos šaltinio.

Paskutinį kartą programa atnaujinta 2016 metais spalio mėnesį, Fake GPS Free reikalinga „Android“ operacinė sistema, naujesnė nei 2.3 versijos. Programos pagrindinis langas pavaizduotas 1.5 paveiksle.



**1.5 pav. FakeGPS Free programos langas**

## FakeGPS PRO

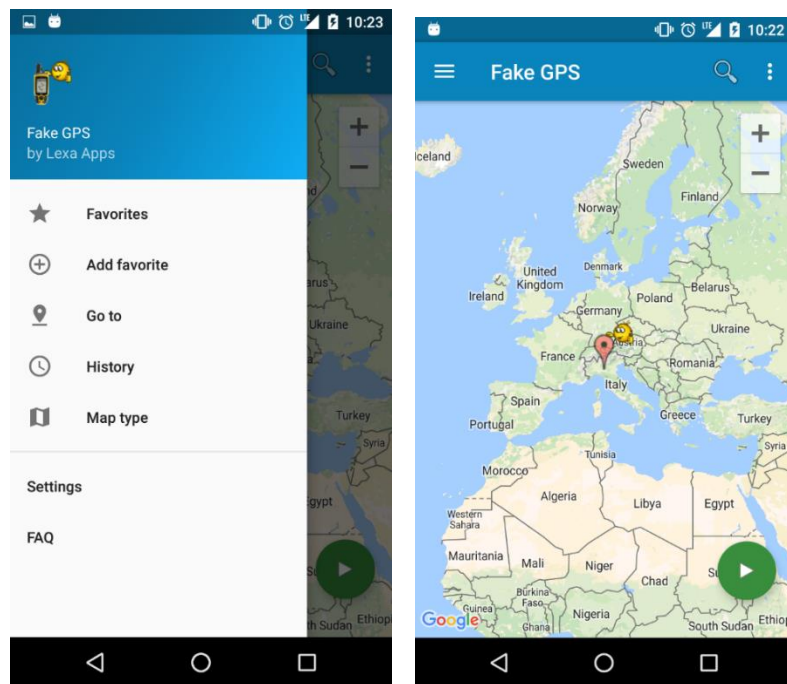
FakeGPS PRO programa yra patobulinta FakeGPS Free programos versija [22]. Atnaujinta GPS vietos klastojimo funkcija leidžia geriau atvaizduoti klastojamą vietą. Programa taip pat turi vietos nustatymus vieną paremta tinklu bei kitą – paremta GPS ir tinklu, pastarasis yra tikslesnis. FakeGPS PRO galima naudotis tik tuo atveju jei programai suteikiami duomenys iš klastotų vietovių, leidžiama prieiti prie tinklo ryšių, keisti sistemos nustatymus, darbą pradėti vos tik operacinė sistema įjungiamas.

Paskutinį kartą programa atnaujinta 2016 metais lapkričio mėnesį. Programa reikalauja, kad telefone „Android“ operacinė sistema būtų naujesnė nei 2.2 versijos.

## Fake GPS Location

Fake GPS Location programa skirta vietos klastojimui, kuri turi papildomų funkcijų tokių kaip mėgstamų vietų žymėjimas, vietos klastojimas vos tik įjungiamas išmanusis įrenginys, kuriame yra instaliuota Fake GPS Location programa. Programa taip pat išsiskiria tuo, kad sukuria judėjimo trajektoriją klastojamoje vietoje. Šios programos vietos nustatymas veikia dviem būdais: apytiksle pozicija, kuri yra sudaroma remiantis tinklu ir įrenginio vieta, kuri nustatoma remiantis tinklu ir GPS koordinatėmis [23]. Tačiau ši programa reikalauja daugiau prieigų prie vartotojo duomenų, nuo kurių nepriklauso jos veikimas. Pirmą kartą paleidžiant programą vartotojo yra paprašoma suteikti prieigą ne tik prie sistemos saugumo nustatymų, bet ir telefono statuso, Wi-Fi ryšių, klastotų vietovių istorijos, visapusiško priėjimo prie jame ryšių ir mediatekos duomenų. Programos pagrindinis langas pavaizduotas 1.6 paveiksle.

Paskutini kartą programa atnaujinta buvo 2016 metais spalio mėnesį.

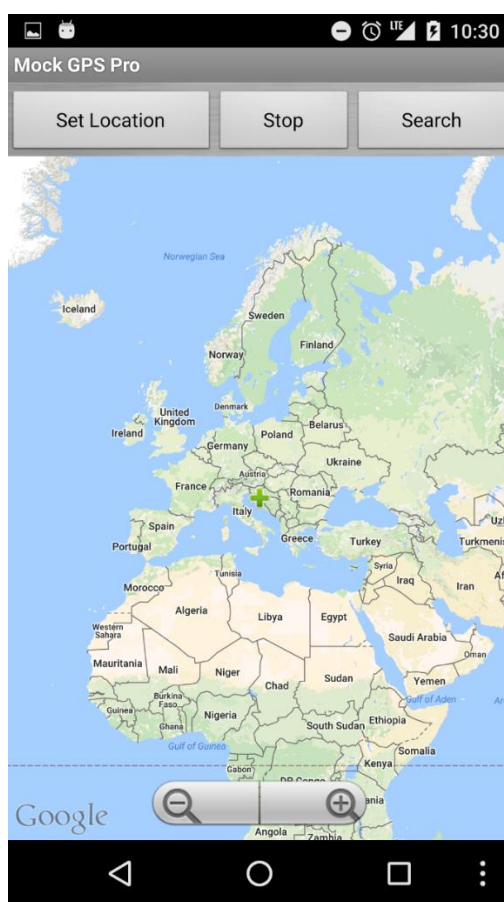


1.6 pav. FakeGPS programos langas

## Mock GPS PRO

Programa Mock GPS PRO yra vietos klastojimo programa, pasižyminti patogia vartotojo sąsaja bei vietos, iš kurios buvo klastotojas GPS, pažymėjimu. Programa vietovę gali nustatyti keliais būdais: apytikslis vietos nustatymas yra paremtas tinklu, tiksli vieta nustatoma GPS ir tinklo pagalba, taip pat programa leidžia naudotis papildomomis funkcijomis vietos nustatymui atlikti. Mock GPS PRO reikalauja papildomų prieigų prie telefone esančių failų ir nustatymų [24].

Paskutinį kartą programa atnaujinta buvo 2012 metais liepos mėnesį, nepaisant to, ji vis dar veikia. Mock GPS PRO programai reikalinga „Android“ operacinė sistema naujesnė, nei 1.6 versijos. Programos vartotojo sąsaja pavaizduota 1.7 paveiksle.



1.7 pav. Mock GPS PRO programos langas

Įrankių įvertinimui pasirinktos tos savybės, kurios padeda sukurti realistišką GPS vietos klastojimo įvaizdį. Šiuo atveju išskirtas programų prieinamumas, reklamų atvaizdavimas naudojimosi metu, administratoriaus privilegijos siekiant naudotis programa, vietų istorija, mėgstamiausių vietų pažymėjimas, vietovės paieška, egzistuojančių maršrutų pasirinkimas, žemėlapių vaizdavimo tipo pasirinkimas ir prieinamumo prie įrenginio duomenų teisių suteikimas.

## 1.2 lentelė. Esamų sprendimų palyginimas

Palyginimo kriterijus	FakeGPS Free	FakeGPS Pro	Mock GPS Pro	Fake GPS	Siekiamas sprendimas
Nemokama	Taip	Ne	Taip	Taip	Taip
Be reklamos	Ne	Taip	Taip	Ne	Taip
Nereikalingos telefono administratoriaus privilegijos (root)	Taip	Ne	Taip	Taip	Taip
Vietų istorija	Ne	Taip	Ne	Taip	Taip
Mėgstamiausios vietos	Ne	Taip	Ne	Taip	Taip
Vietos paieška	Ne	Taip	Taip	Taip	Taip
Kitų žmonių maršrutų naudojimas	Ne	Ne	Ne	Ne	Taip
Maršruto sukūrimas	Ne	Ne	Ne	Ne	Taip
Pasirinkimas tarp kelių žemėlapių vaizdavimo tipų	Taip, tarp 2	Taip, tarp 2	Ne	Taip, tarp 4	Taip
Reikalinga nustatymuose įjungti „Mock location“	Taip	Ne	Taip	Taip	Taip
Reikalinga daug perteklinių teisių	Ne	Ne	Taip, pvz.: Naudoti blykste, arba valdyti vibraciją	Taip, matyti įrenginio duomenis, bei išsaugotus duomenis	Ne

## 1.9. Siekiamo sprendimo apibrėžimas

Atlikus esamų sprendimų analizę nuspręsta sukurti patikimą 3-čios šalies programą, kuri galėtų patikimai apsaugoti vartotojo vietos privatumą, kuriai nereikėtų telefono administratoriaus privilegijų bei perteklinių teisių.

Toliau sprendžiami uždaviniai:

1. Specifikuoti funkcinius ir nefuncinius reikalavimus.
2. Pasirinkti naudojamą technologijas.
3. Suprojektuoti sistemos architektūrą, veiklos modelį.

#### 4. Suprojektuoti vartotojo sąsajos modelį.

### 1.10. Analizės išvados

1. Atlikus išmaniųjų įrenginių operacinių sistemų rinkos palyginimą, paaiškėjo, kad didžiausią rinkos dalį mobiliųjų įrenginių operacinių sistemų rinkoje užima „Android“ operacinė sistema, po jos rikiuojasi Apple iOS ir kitos ne tokios žinomos operacinės sistemos. Dėl didelio vartotojų skaičiaus ir mobiliųjų įrenginių rinkos dalies programos kūrimui pasirinkta „Android“ operacinė sistema.
2. Atlikus vietos keitimo būdų analizę, įvertinus visus kriterijus, pasirinkta kurti patikimos 3-čios šalies (TPP) programą, taip sukuriant didžiausią vartotojo duomenų apsaugą ir kuriai nereikia telefono administratoriaus privilegijų, kadangi, daugumai vartotojų šių privilegijų gavimas yra nepriimtinas, nes tai pažeidžia mobilaus įrenginio garantinio remonto sąlygas.
3. Atlikus panašių sprendimų analizę, pastebėta, kad jeigu programa yra nemokama, tai joje yra reklamų. Taip pat, programos skiriasi tuo, kad turi arba neturi naudotų vietų istorijos, mėgstamiausios vietos išsaugojimo bei paieškos. Daugumoje programų skiriasi žemėlapių vaizdavimo tipai. Norint, kad nereiktų įjungti Mock location nustatymuose reikalingos telefono administratoriaus privilegijos. Taip pat, kai kurios programos nori teisių, kurios su vietos slėpimu neturi nieko bendra. Tokios programos atrodo nepatikimos.
4. Remiantis atlikta analize, pasirinkta kurti programą, kuri turėtų maršrutų sukūrimą bei kitų žmonių įrašytų maršrutų panaudojimo galimybes, būtų nemokama ir kuriai nereiktų administratoriaus privilegijų. Programai pasirinktas pavadinimas „Hide Yourself“.



## 2. IŠMANIŲJŲ ĮRENGINIŲ VIETOS SLĖPIMO PROGRAMOS SPRENDIMO REIKALAVIMŲ SPECIFIKACIJA IR PROJEKTAVIMAS

Šiame skyriuje, remiantis analizės išvadomis, bus aprašyti projekto apribojimai, funkciniai reikalavimai, nefunkciniai reikalavimai, techninė specifikacija ir naudojamos technologijos.

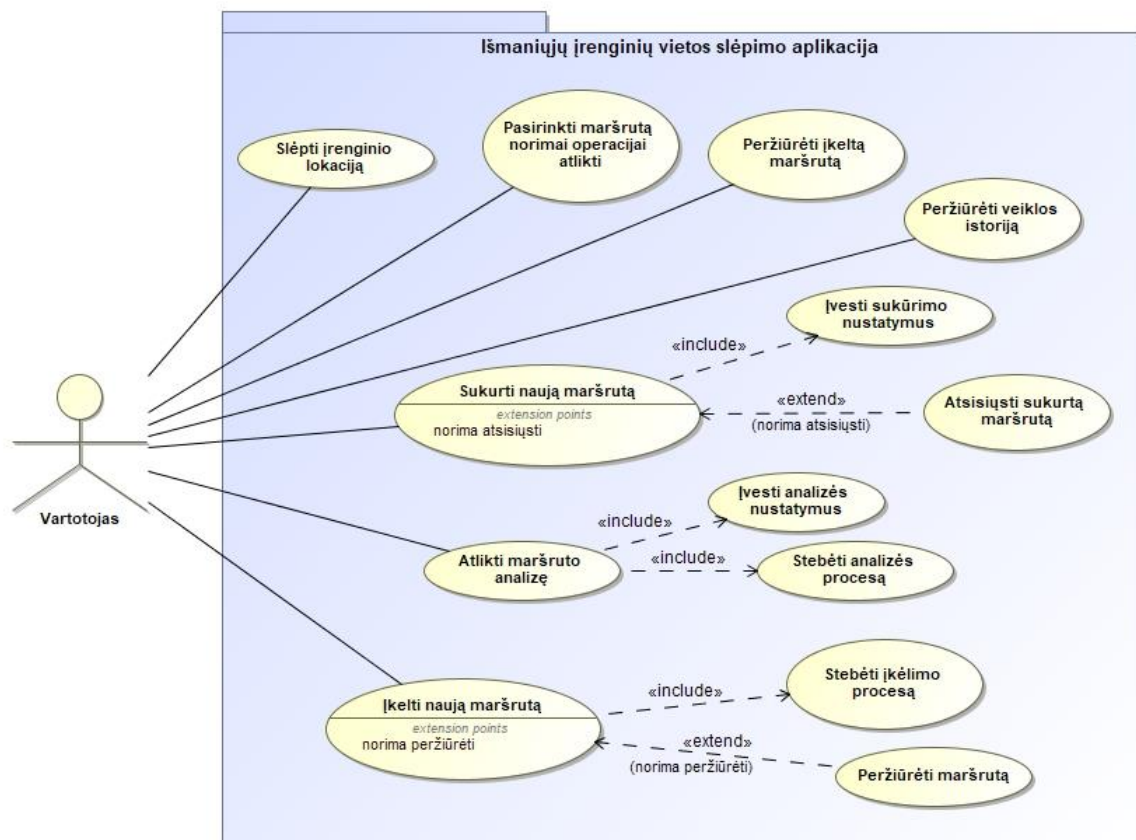
### 2.1. Reikalavimų specifikacija

Siekiant užtikrinti sklandų „Hide Yourself“ veikimą apibrėžtos funkcijos, kurias turi atlikti minėta programa:

1. Programa turi suteikti galimybę slėpti arba pakeisti įrenginio vietą.
2. Programa turi suteikti vartotojui prieigą prie naudotų vietų istorijos.
3. Programa turi turėti mėgstamiausių vietų pridėjimo, pasirinkimo ir ištrynimo funkcijas.
4. Kuriamas sprendimas vartotojui turi suteikti galimybę vietos ieškoti pagal adresą.
5. Programa turi leisti pasirinkti tarp kelių vietų vaizdavimo tipų.
6. Būsimas sprendimas taip pat turi leisti importuoti naujus maršrutus, juos analizuoti.

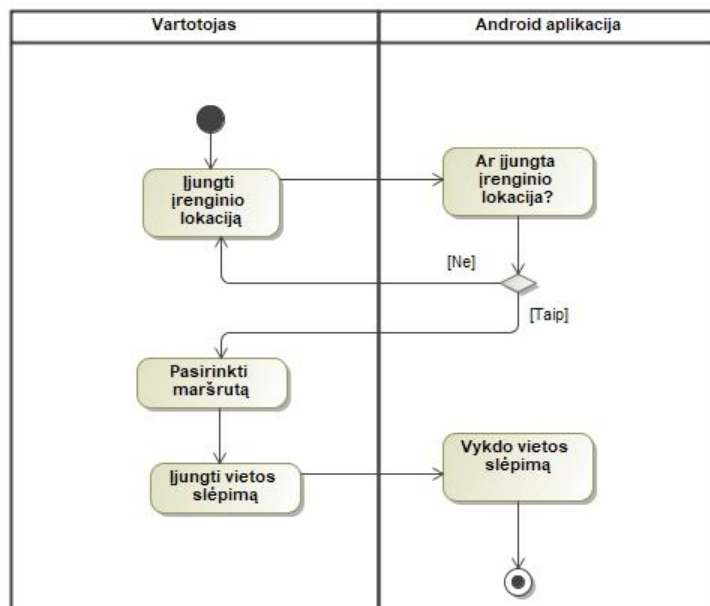
#### 2.1.1. „Hide Yourself“ programos panaudojimo atvejų diagrama

Paveiksle 2.1 pavaizduoti panaudojimo atvejai, kuriuos atliks įrenginių vietos slėpimo programa.



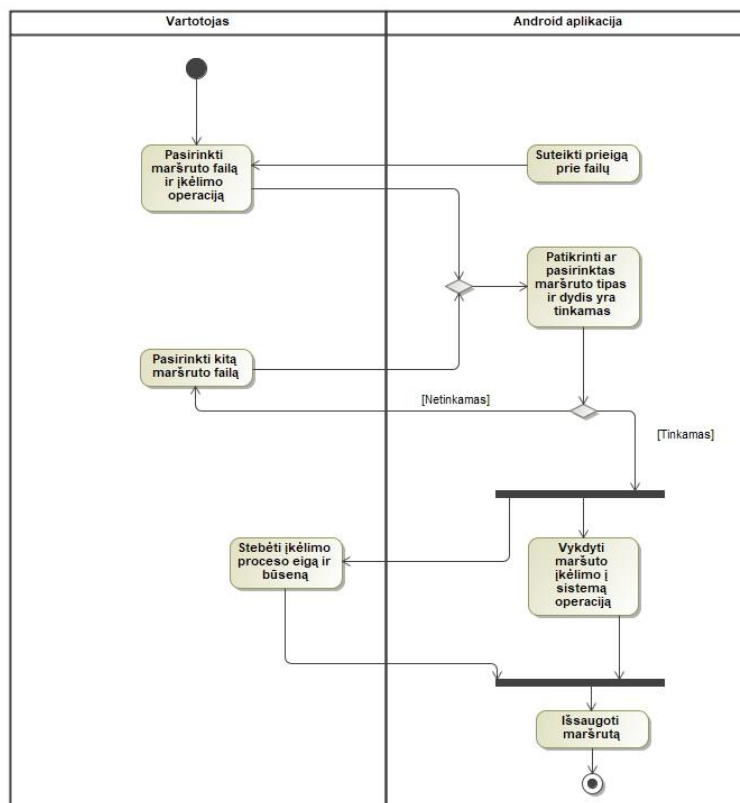
2.1 pav. Panaudojimo atvejų diagrama

Pagrindiniai panaudojimo atvejai yra išskirstyti į veiklos diagramas, kurios parodo konkretų programą veikimą nurodytu atveju. Panaudojimo atvejo „Slėpti įrenginio vietą“ veiklos diagrama pavaizduota 2.2 paveikslėlyje.



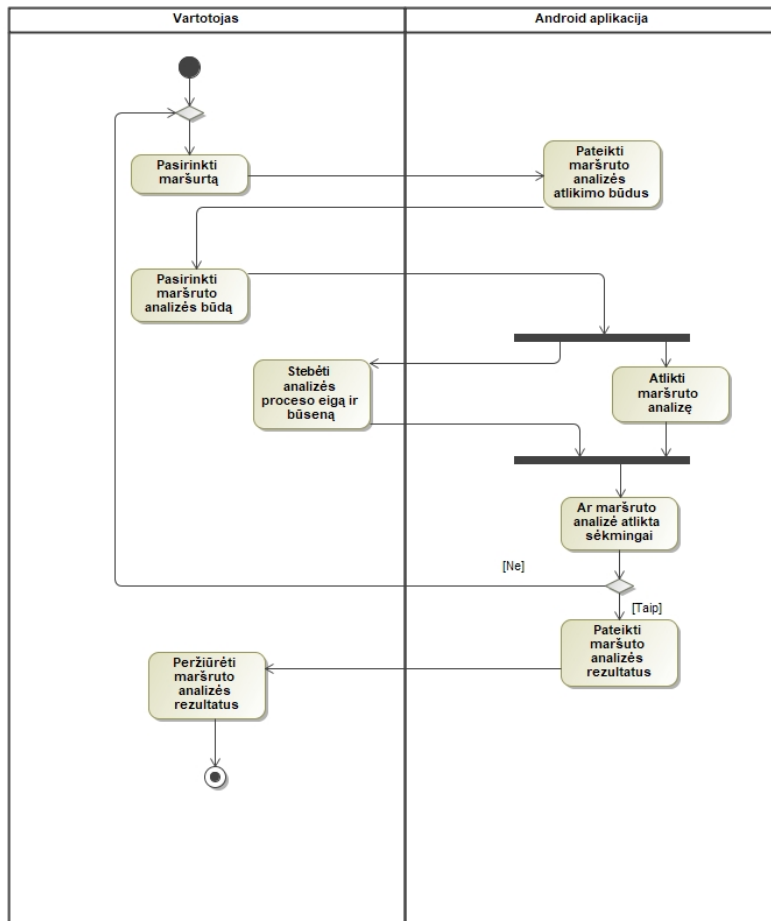
2.2 pav. Panaudojimo atvejo „Slėpti įrenginio vietą“ veiklos diagrama

Vartotojas iš pradžių turi išmaniajame įrenginyje įjungti GPS vietą, tada pasirinkti norimą įrenginio vietą ar maršrutą, paspausti paleidimo mygtuką, kuris pradeda vietos keitimo tarnybą. Visos gaunamos užklauskos į vietos tarnybą grąžina pakeistą vietą.



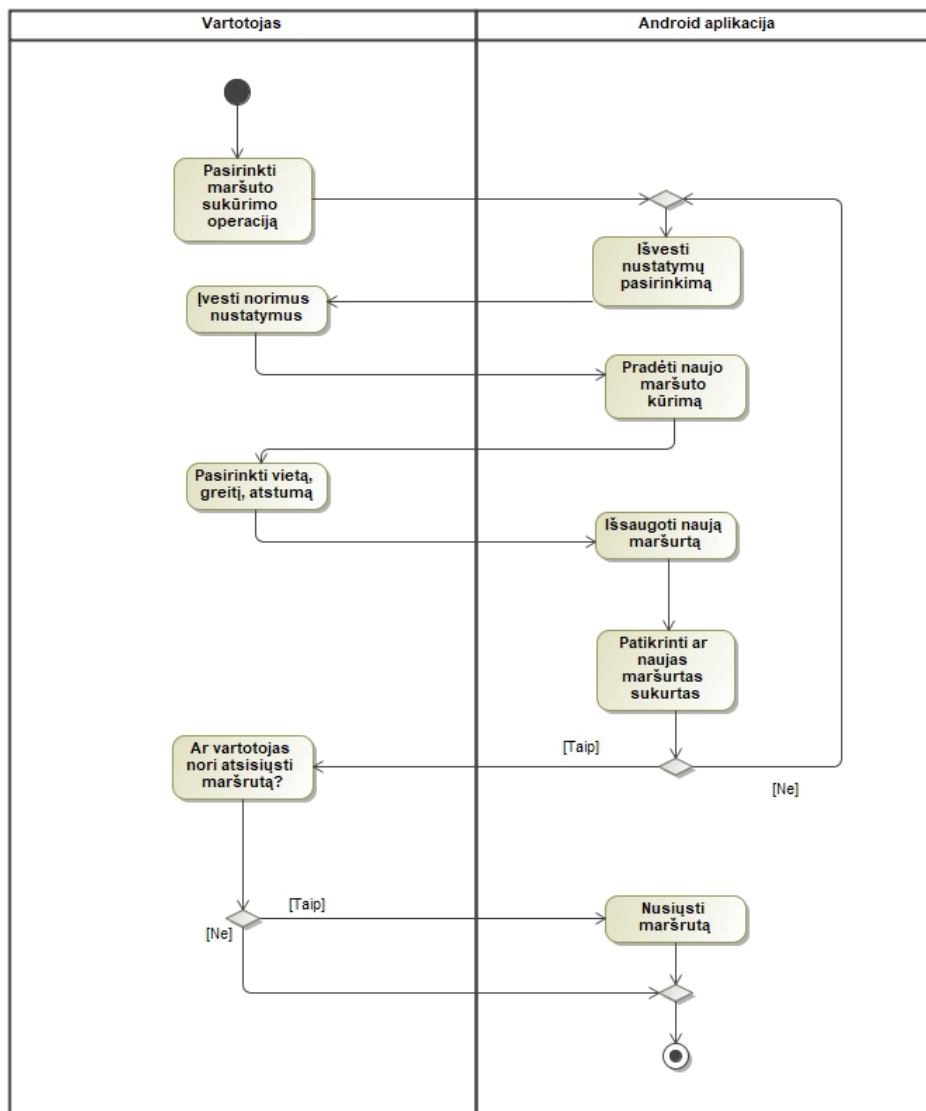
2.3 pav. Panaudojimo atvejo „Įkelti naują maršrutą“ veiklos diagrama

2.3 paveiksle vaizduojama „Įkelti naują maršrutą“ veiklos diagrama. Šiuo panaudojimo atveju metu vartotojas gali pasirinkti jau išsaugotą maršruto failą, jį įkelti į programą. Tada šis failas yra nuskaitomas ir išanalizuojamas, taip nustatoma ar jis tinkamas. Jeigu maršrutas tinkamas, jis yra išsaugomas ir jį galima vėliau panaudoti vietos slėpimui.



**2.4 pav. Panaudojimo atvejo „Atlikti maršruto analizę“ veiklos diagrama**

2.4 paveiksle pavaizduota maršruto analizės veiklos diagrama. Jos metu vartotojas pasirenka maršrutą, kurį nori papildomai išanalizuoti, programai užbaigus minėta procesą, programos naudotojas gali pamatyti rezultatus, kuriuose yra nurodomi tokie dalykai kaip: maršruto greitis, atstumas, laikas.



2.5 pav. Panaudojimo atvejo „Sukurti naują maršrutą“ veiklos diagrama

2.5 paveikslėlyje yra pavaizduota „Sukurti naują maršrutą“ veiklos diagrama. Joje parodoma, kaip galima sukurti naują maršrutą. Visų pirma vartotojas turi įvesti norimus nustatymus, tada pasirinkti taškus, kur bus maršruto pradžia ir pabaiga, įvesti greitį. Viską atlikus maršrutas yra išsaugojamas programoje, bet taip pat dar vartotojui leidžiama išsaugoti maršrutą ir atskirai, kad galėtų jį persiųsti į kitus įrenginius ar kitiems žmonėms.

### 2.1.2. Funkciniai reikalavimai

2.1 – 2.8 lentelėse yra aprašomi įrenginio vietos slėpimo projekto funkciniai reikalavimai naudojant *Volere* šabloną.

**2.1 lentelė. Naujos vietos pasirinkimo panaudojimo atvejo lentelė**

<b>Vartotojas/Aktorius</b>	Vartotojas
<b>Aprašas</b>	Apima procesą, kurio metu vartotojas nori pakeisti savo įrenginio vietą
<b>Prieš sąlyga</b>	-
<b>Sužadinimo sąlyga</b>	Reikia pakeisti įrenginio vietą
<b>Po-sąlyga</b>	Programa, kuri nori gauti įrenginio vietą gauna pakeistą vietą

**2.2 lentelė. Naujo maršruto sukūrimo panaudojimo atvejo lentelė**

<b>Vartotojas/Aktorius</b>	Vartotojas
<b>Aprašas</b>	Apima procesą, kurio metu vartotojas nori sukurti naują maršrutą
<b>Prieš sąlyga</b>	-
<b>Sužadinimo sąlyga</b>	Reikia sukurti maršrutą, kuriuo įrenginys galėtų judėti
<b>Po-sąlyga</b>	Naujas maršrutas yra sukuriamas ir jį galima panaudoti pozicijos keitimui

**2.3 lentelė. Išsaugotos vietos pasirinkimo panaudojimo atvejo lentelė**

<b>Vartotojas/Aktorius</b>	Vartotojas
<b>Aprašas</b>	Apima procesą, kurio metu vartotojas išsirenka prieš tai naudotą vietą ir ją pasirenka
<b>Prieš sąlyga</b>	-
<b>Sužadinimo sąlyga</b>	Reikia pakeisti įrenginio vietą
<b>Po-sąlyga</b>	Programa, kuri nori gauti įrenginio vietą gauna pakeistą vietą

**2.4 lentelė. Vietos išsaugojimo į mėgstamiausių sąrašą panaudojimo atvejo lentelė**

<b>Vartotojas/Aktorius</b>	Vartotojas
<b>Aprašas</b>	Apima procesą, kurio metu vartotojas išsaugo vietą į mėgstamų vietų sąrašą
<b>Prieš sąlyga</b>	-
<b>Sužadinimo sąlyga</b>	Norima išsaugoti dažnai naudojamą įrenginio vietą į mėgstamų vietų sąrašą, kad lengviau ją pasiekti
<b>Po-sąlyga</b>	Vieta yra įrašoma į sąrašą

**2.5 lentelė. Maršruto įkėlimo panaudojimo atvejo lentelė**

<b>Vartotojas/Aktorius</b>	Vartotojas
<b>Aprašas</b>	Apima procesą, kurio metu vartotojas įkelia išsaugotą maršrutą į programą
<b>Prieš sąlyga</b>	Turimas .GPX failas
<b>Sužadinimo sąlyga</b>	Norima panaudoti išsaugotą maršrutą įrenginio vietai pakeisti
<b>Po-sąlyga</b>	Maršrutas yra įkeliamas į programą

## 2.6 lentelė. Maršruto pasirinkimo panaudos atvejo lentelė

<b>Vartotojas/Aktorius</b>	Vartotojas
<b>Aprašas</b>	Apima procesą, kurio metu vartotojas pasirenka norimą maršrutą, kurį programa imituos
<b>Prieš sąlyga</b>	-
<b>Sužadinimo sąlyga</b>	Norima pasirinkti maršrutą kurį programa imituos
<b>Po-sąlyga</b>	Maršrutas yra pasirenkamas, parodomas ir išanalizuojamas naudojimui

## 2.7 lentelė. Maršruto analizės panaudos atvejo lentelė

<b>Vartotojas/Aktorius</b>	Vartotojas
<b>Aprašas</b>	Apima procesą, kurio metu vartotojas nori išanalizuoti maršrutą
<b>Prieš sąlyga</b>	-
<b>Sužadinimo sąlyga</b>	Norima išanalizuoti maršrutą, kad pamatyti jo parametrus
<b>Po-sąlyga</b>	Maršrutas yra išanalizuojamas

## 2.8 lentelė. Pozicijos keitimo įjungimo panaudos atvejo lentelė

<b>Vartotojas/Aktorius</b>	Vartotojas
<b>Aprašas</b>	Apima procesą, kurio metu vartotojas paleidžia vietos klastojimo tarnybą
<b>Prieš sąlyga</b>	Turi būti pasirinktas ir išanalizuotas maršrutas
<b>Sužadinimo sąlyga</b>	Norima paslėpti įrenginio buvimo vietą
<b>Po-sąlyga</b>	Vietos keitimo tarnyba yra įjungiamą ir taip klastojama įrenginio buvimo vieta

### 2.1.3. Nefunkciniai reikalavimai

Šiame skyriuje yra aprašomi įrenginio vietos slėpimo projekto nefunkciniai reikalavimai pasitelkiant *Volere* šabloną, kuris matomas 2.9 lentelėje.

## 2.9 lentelė. Nefunkcinių reikalavimų *Volere* šablono lentelė

<b>Išvaizdai</b> (angl. <i>Look and feel</i> )	Programoje matoma esama buvimo vieta žemėlapyje. Ekranu viršuje yra vieta, kurioje galima rašyti adresą, kurį norima surasti. Viršuje dešinėje pusėje yra ieškojimo mygtukas ir mygtukas „daugiau“, kurį paspaudus atidaromas meniu, kuriam galima pasirinkti kur norima eiti toliau. Dešinėje apačioje yra „start“ arba „stop“ mygtukas.
<b>Panaudojamumui</b> (angl. <i>Usability</i> )	Programa naudojama „Android“ platformoje.
<b>Vykdyimo charakteristikoms</b> (angl. <i>Performance</i> )	Tik atidarius programą yra užkraunamas žemėlapis, o vėliau ieškoma ir nustatoma įrenginio buvimo vieta, kuri parodoma žemėlapyje.

<b>Veikimo sąlygoms</b> (angl. <i>Operational</i> )	-
<b>Sistemos/paslaugos priežiūrai</b> (angl. <i>Maintainability and portability</i> )	Atliekama su JAVA kalba.
<b>Saugumui</b> (angl. <i>Security</i> )	-

#### 2.1.4. Techninė specifikacija

##### Reikalavimai programinei įrangai:

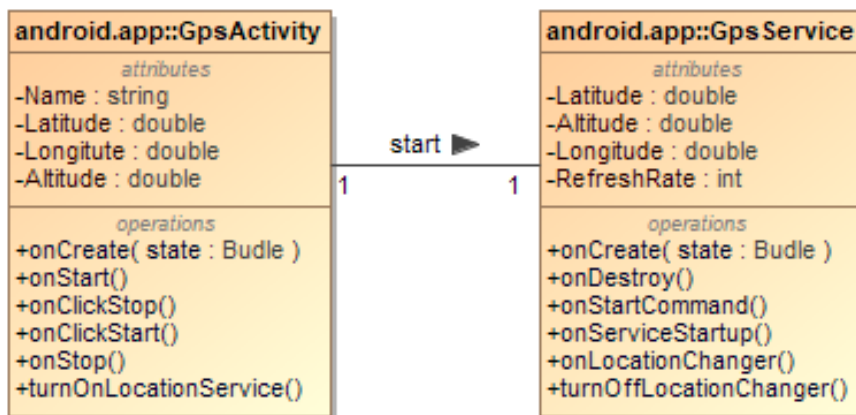
Programuotojui: „Android Studio“, „Android“ 23 SDK, Java JDK

Vartotojui: Bent 6.0 versijos „Android“ sistema, parsųsta ir įrašyta programa.

##### Reikalavimai tinklinei įrangai ir protokolams:

Vartotojui: Interneto prieiga, vietos nustatymo galimybė.

#### 2.2. Dalykinės srities modelis

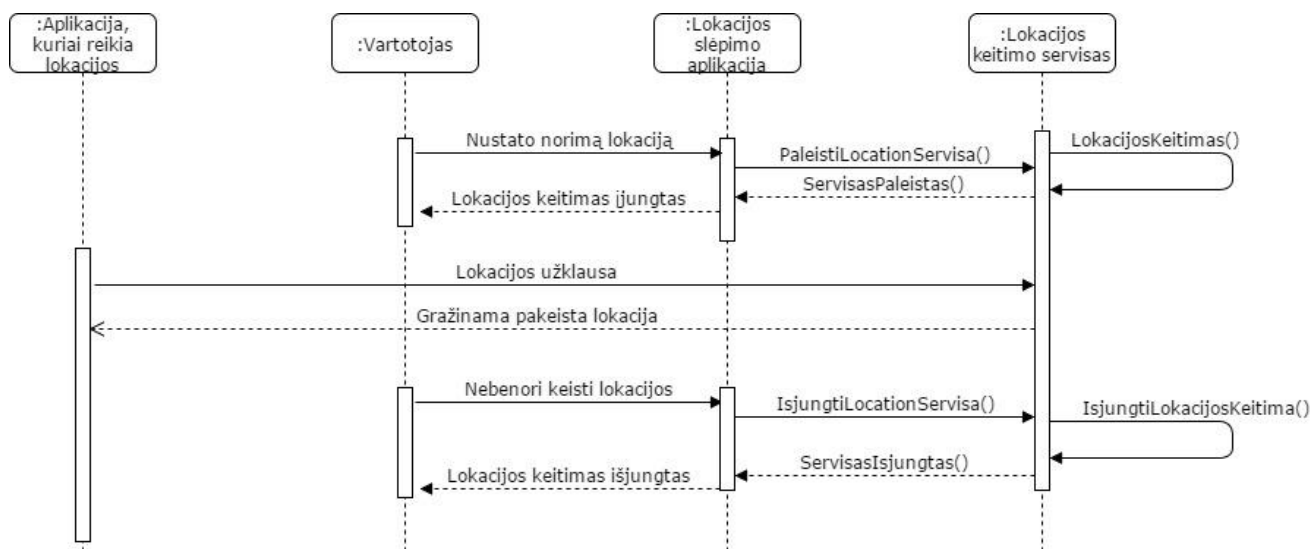


2.6 pav. Klasių diagrama

2.6 paveiksle pavaizduotas dalykinės srities modelis, kuriame *name* reikalingas nurodyti programos/tarnybos vardą, *latitude*, *longitude* ir *altitude* reikalingi nurodyti vietai. Funkcijos *onCreate*, *onStart*, *onClockStop*, *onClickStart*, *onStop* yra privalomos programos veikimui. Funkcija *turnOnLocationService* yra reikalinga įjungti tarnybą, o *turnOffLocationChanger* yra reikalinga nusiųsti komandai, kad tarnyba būtų išjungta.

Tie patys kintamieji naudojami ir tarnybos, bet prisideda skaičius *RefreshRate*, kuris reikalingas, nustatyti kas kiek laiko įrenginio vieta yra atnaujinama. Funkcijos *onCreate*, *onDestroy*,

*onStartCommand* yra privalomos tarnybos paleidimui bei išjungimui, *onLocationChanger* reguliariai atkartojimas funkcijos kvietimas atsižvelgiant į *RefreshRate* nurodytą skaičių.



2.7 pav. Veikimo sekų diagrama

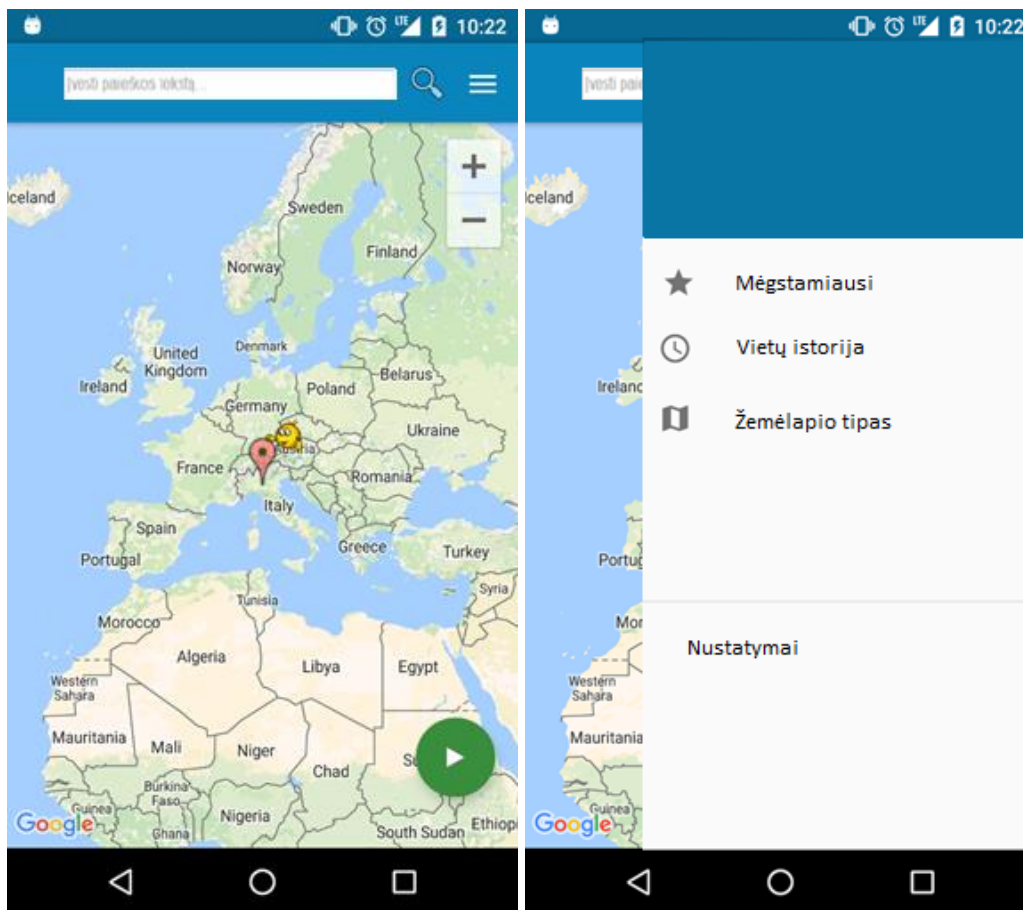
2.7 paveiksle pavaizduota, veikimo sekų diagrama, kurioje matosi, kad vartotojas iš pradžių nusistato norimą įrenginio vietą, tada paspaudžia mygtuką, kuris paleidžia vietos keitimo tarnybą. Ši tarnyba kas tam tikrą laiko tarpą keičia įrenginio vietą atitinkamai gautam maršrutui. Kai tarnyba yra įjungta, vartotojas eina į savo norimas kitas programas, kurioms reikia vietos ir jos siunčia užklausą į vietos tarnybą, kuri joms grąžina pakeistą įrenginio vietą. Pabaigęs darbą su minėta programa vartotojas nebenori keisti vietos ir paspaudžia išjungimo mygtuką, kuris išjungia vietos keitimo tarnybą.

### 2.3. Naudotojų sąsajos modelis

2.8 paveiksle, kairėje pusėje matome „Hide Yourself“ programos maketą. Programoje matoma esama buvimo vieta žemėlapyje. Ekranu viršuje yra vieta, kurioje galima rašyti adresą, kurį norima surasti. Viršuje dešinėje pusėje yra ieškojimo mygtukas ir mygtukas „daugiau“, kurį paspaudus atidaromas meniu, kuriam galima pasirinkti kur norima eiti toliau. Dešinėje apačioje yra „start“ arba „stop“ mygtukas.

Dešinėje pusėje matome programos meniu maketą, kuris parodomas paspaudus „daugiau“ mygtuką. Atidaromas meniu, kuriame yra vietų istorija, mėgstamiausios vietos, pasirinkimas tarp kelių žemėlapio vaizdavimo tipų, programos nustatymų meniu.





2.8 pav. „Hide Yourself“ pradinio vaizdo ir meniu vaizdo maketas

## 2.4. Reikalavimų apibendrinimas

Reikia sukurti „Hide Yourself“ programą, kuri turėtų galimybes slėpti arba pakeisti įrenginio vietą, naudotų vietų istoriją, turėtų mėgstamiausių vietų pridėjimą, pasirinkimą bei ištrynimą, suteiktų galimybę ieškoti vietos pagal adresą, leistų pasirinkti tarp kelių vietų vaizdavimo tipų. Siekiamos sukurti „Hide Yourself“ programos funkcijos aprašomos funkcinuose reikalavimuose. Programos kūrimo palengvinimui pasitelkiama klasių ir veiklos sekų diagramos. Toliau bus projektuojamos papildomos padedančios diagramos, tokios kaip robastiškumo, loginė sistemos architektūra, programos posistemių diagramos ir pagrindinių funkcijų sekų diagramos.

### 3. IŠMANIŲJŲ ĮRENGINIŲ VIETOS SLĖPIMO REALIZACIJA

Siekiant sukurti funkcionalų išmaniųjų įrenginių vietos slėpimo taikomąją programą buvo atliktas detalus įrankio projektavimas. Šiame skyriuje atlikta reikalavimų analizė, suprojektuota robastiškumo diagrama, sukurta loginė įrankio architektūra, specifikuotos sekų diagramos, pateiktas išmaniųjų įrenginių vietos slėpimo algoritmas ir diegimo diagrama.

#### 3.1. Sprendimo pagrindimas ir esminiai principai

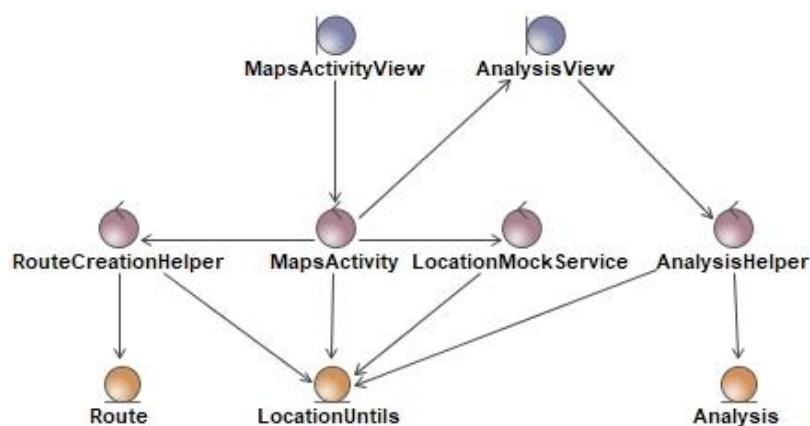
Išmaniųjų įrenginių vietos slėpimo taikomoji programa yra mobiliųjų programų kūrimo dalis. Šis įrankis keičia įrenginio vietą remdamasis projektavimo metu išskirtomis GPS vietos nustatymo metodologijomis.

#### 3.2. Išmaniųjų įrenginių vietos slėpimo įrankio architektūra

##### Reikalavimų analizė

Pagrindinis projekto tikslas yra suprojektuoti išmaniųjų įrenginių vietos slėpimo įrankį panaudojant UML diagramos kūrimo įrankius. Apsibrėžti vartotojo sąsajos pagrindinį langą, nustatyti programinių langų skaičių, realizuoti sistemos logiką ir diegimo modelį.

Robastiškumo diagrama padeda atskleisti kokios pagrindinės vartotojo sąsajos turi būti įgyvendintos, kiek ir kokie turi būti valdikliai užtikrinantys programos veikimo logiką. Analizės diagramos parodo, kokie turi būti įėjimo ir išėjimo failai. Robastiškumo diagrama pateikiama 3.1 paveiksle.

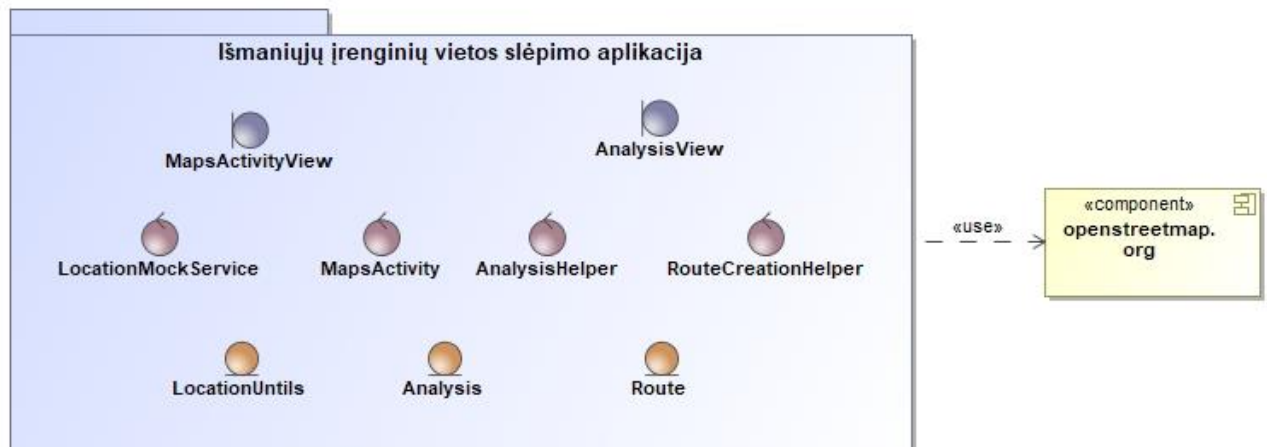


3.1 pav. Robastiškumo diagrama

3.1 diagramoje matome, kad pagrindinis kontroleris yra *MapsActivity*, kuris naudoja kitas tarnybas, kontrolerius, duomenų klases ir vaizdus. Yra du vaizdai, pagrindinis yra *MapsActivityView*, o kitas yra *AnalysisView*, kuris parodomas tik tada, kai atliekama analizė. *RouteCreationHelper* kontroleris yra pasitelkiama maršrutui sukurti, tam panaudodamas *Route* duomenų klasę. *LocationUtils* duomenų klasė naudojamiems duomenims saugoti, kurie yra panaudojami visuose kontrolieriuose.

### Loginė sistemos architektūra

Loginė sistemos architektūra susideda iš pagrindinių paslaugų t. y. išmaniųjų įrenginių vietos slėpimo programos lango ir *openstreetmap.org* komponento, kuris papildomai iškviečia tinklo paslaugą. Išmaniųjų įrenginių vietos slėpimo programos loginė architektūra pateikiama 3.2 paveiksle.

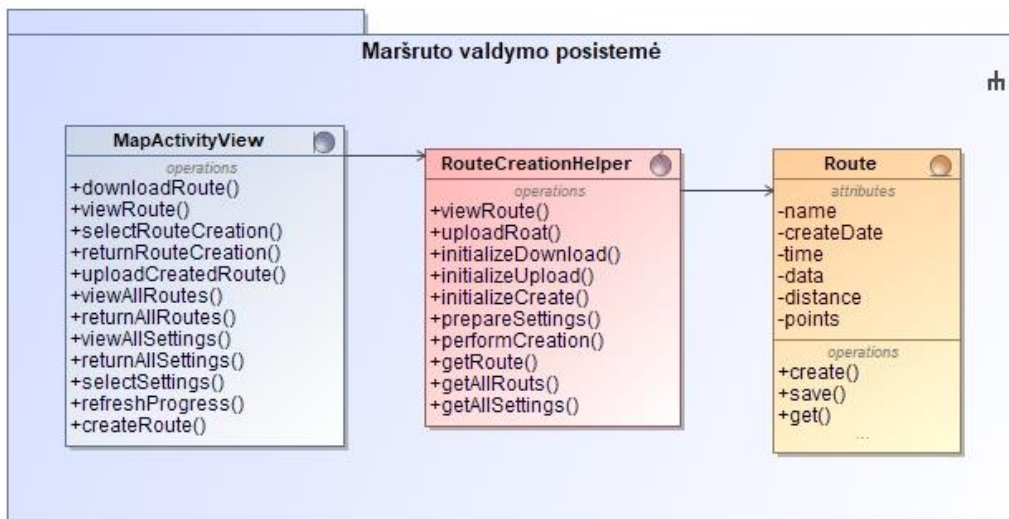


3.2 pav. Loginė sistemos architektūra

3.2 paveiksle matoma loginė sistemos architektūra, kurio kontroleriai, vaizdai ir duomenų klasės yra vienoje programoje. Programa panaudoja išorėje esančio *openstreetmap.org* API (angl. *Application Programming Interface*).

### Maršruto valdymo posistemė

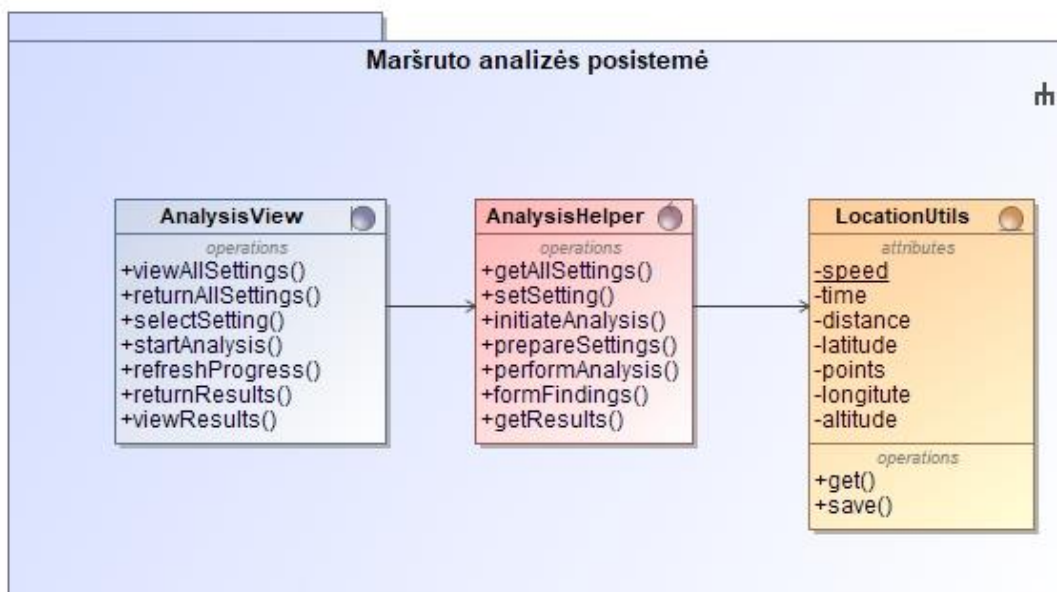
Atsidarius išmaniąją programėlę vartotojas patenka į maršruto valdymo posistemę, iš kurios mato įkėlimo langą ir gali įkelti naują maršrutą. Valdiklis kontroliuoja įkėlimo procesą ir galiausiai vartotojas pagrindiniame programos lange gali matyti įkeltą maršrutą. Maršruto valdymo posistemės schema pateikiama 3.3 paveiksle.



3.3 pav. Maršruto valdymo posistemė

### Maršruto analizės posistemė

Detalus maršruto analizės posistemės modelis vaizduoja naudojamą klases ir metodus. Maršruto analizės posistemės veikimo modelis yra pateiktas 3.4 paveiksle.



3.4 pav. Maršruto analizės posistemė

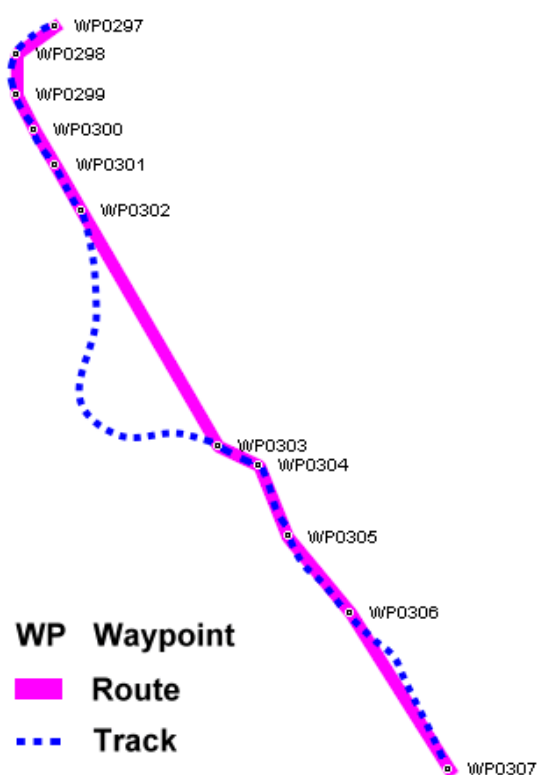
3.4 paveiksle yra parodyta maršruto analizės posistemė, kurioje nurodyti sujungimai tarp *AnalysisView* vaizdo, *AnalysisHelper* kontrolerio ir *LocationUtils* duomenų klasės. Nurodoma, kad duomenų klasėje saugojami pagrindiniai duomenys, tokie kaip greitis, platumas, ilgumas, maršruto taško

eilės numeris ir kt. Analizės kontroleryje yra nustatymų keitimo, analizės paleidimo ir atsakymo formavimo metodai. Vaizdo faile atitinkami metodai tiems duomenims parodyti.

### 3.3. Duomenų modelis

Duomenys saugojami .GPX formatu, XML schemeje. Šis formatas gali būti naudojamas užrašyti kelio taškus, maršrutus, takus. Jis yra atviras, lengvai prieinamas ir gali būti naudojamas nemokamai. Vietos informacija ir pagal pasirinkimą laikas, aukštuma bei kita informacija gali būti saugojama žymose. Tai leidžiant lengvai dalintis GPS informacija tarp visų įrenginių ir programų.

3.5 paveiksle parodyti skirtumai tarp kelio taškų (angl. *Waypoint*), kelio (angl. *Route*), takas arba maršrutas (angl. *Track*).



3.5 pav. Kelio taškai, keliai, takai užregistruoti GPS pagalba

#### Žymių reikšmės:

- „Waypoint“ (wpt arba WP) yra individualus kelio taškas, kuris yra dalis tokio tipo taškų kolekcijos, kuri neturi jokių laiko žymų, tik koordinatės taško ir galbūt kitokią vietą aprašančią informaciją.
- „Route“ (rte) yra kelias, tai - sąrašas, kuris susideda iš individualių kelio taškų einančių vienas paskui kitą. Tai nėra tikslus kelias, taškai padedami tik tada, kai yra padaromas staigus posūkis. Turi pradžią ir pabaigą.

- **„Track“ (trk)** yra takas (maršrutas), kuris yra sudarytas iš bent vieno segmento, kuris susideda iš kelio taškų, kurie parodo tikslų kelią. Tako segmentas turi tako taškų sąrašą, kurie logiškai susieti paeiliui. Tako segmentas baigiasi tik tada, kai yra išjungiamas GPS imtuvas, arba prarandamas GPS signalas. Tada pradedamas naujas tako segmentas.

Šiame darbe bus naudojama „Track“, o ne „Route“, kadangi „Track“ yra tikslesnis, turi daugiau taškų ir taškas yra išsaugojamas kas sekundę. Tai palengvina koordinacių atvaizdavimą. „Waypoint“ bus naudojama kelio taškams aprašyti.

„trk“ gali turėti tokias žymes:

```
<name> pavadinimas </name>
<cmt> komentaras </cmt>
<desc> aprašymas </desc>
<src> šaltinis </src>
<link> nuoroda su daugiau informacijos </link>
<number> segmento eilės numeris </number>
<type> tako tipas </type>
<trkseg> trk segmentas </trkseg>
```

Iš „trk“ seka „trkseg“, kuris gali turėti tokias žymes:

```
<trkpt> wpt segmentas </trkpt>
<time> taško gavimo laikas </time>
```

Iš „trkseg“ seka „trkpt“, kuris gali turėti tokias žymes:

```
lat="platuma"
lon="ilguma">
  <ele> aukštis </ele>
  <time> gavimo laikas </time>
  <magvar> magnetinė paklaida </magvar>
  <geoidheight> aukštis virš jūros lygio </geoidheight>
  <name> pavadinimas </name>
  <cmt> komentaras </cmt>
  <desc> aprašymas </desc>
  <src> šaltinis </src>
  <link> nuoroda su daugiau informacijos </link>
```

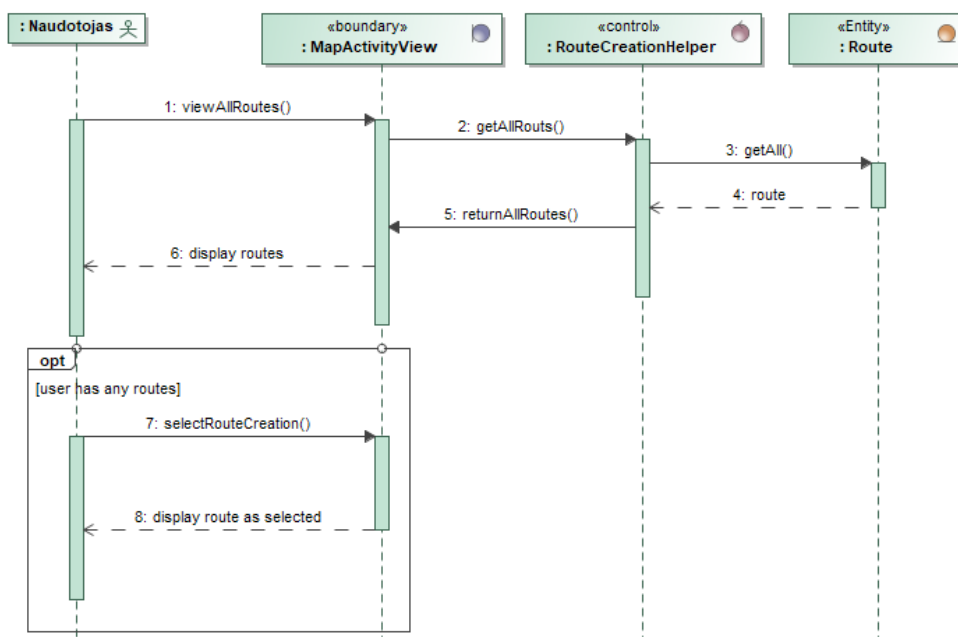
```
<type> taško tipas </type>
```

### Duomenų pavyzdys:

```
<?xml version="1.0" encoding="UTF-8"?>
<gpx version="1.0" creator="KŪRĖJAS" xmlns="http://www.topografix.com/GPX/1/0">
  <trk>
    <name>VARDAS</name>
    <desc>APRAŠYMAS</desc>
    <url>NUORODA IŠ KUR GAUTAS GPX FAILAS</url>
    <trkseg>
      <trkpt lat="51.5441466" lon="0.0000162">
        <time>2018-04-24T08:46:23Z</time>
      </trkpt>
    </trkseg>
  </trk>
</gpx>
```

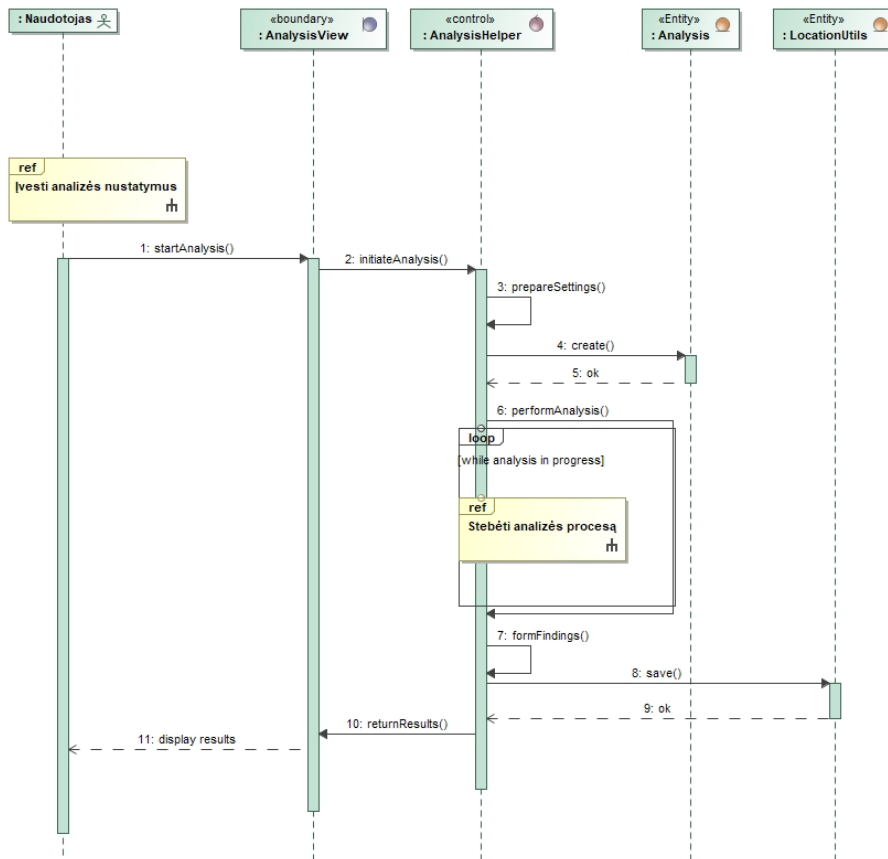
### 3.4. Sistemos elgsenos modelis

Šiame skyriuje pateikiamos sekų diagramos, kurios parodo sistemos veikimą tarp sistemos langų, valdiklių ir klasių. Diagramose parodyti metodai, kurie manipuliuoja gautais duomenimis ir užtikrina galutinį rezultatą, kurio buvo tikimasi.



3.6 pav. Panaudojimo atvejo „Pasirinkti maršrutą“ sekų diagrama

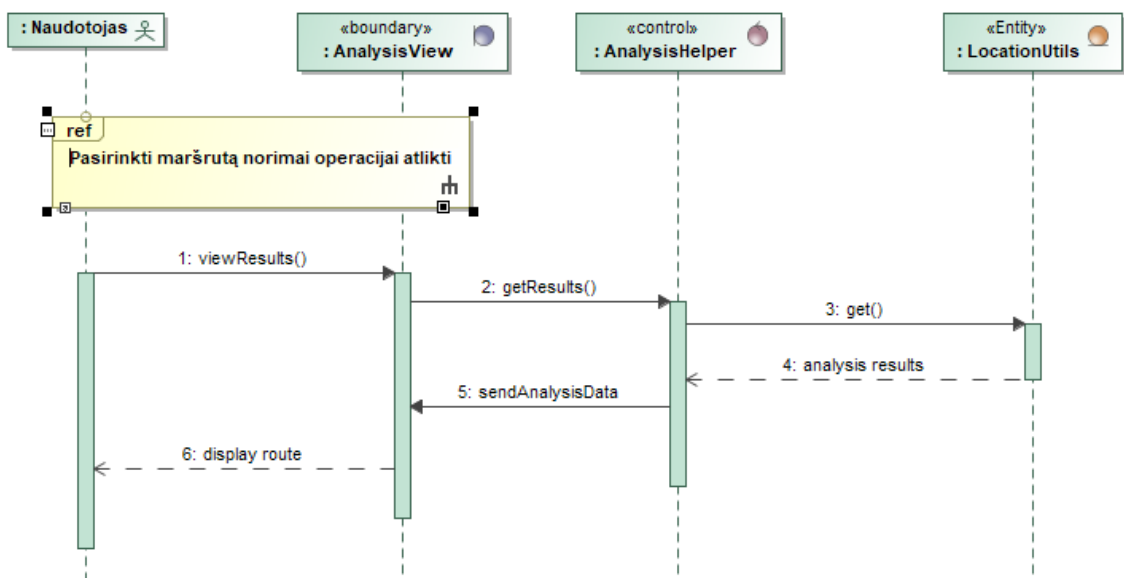
3.6 paveiksle yra „Pasirinkti maršrutą“ sekų diagrama, kurioje parodoma, kad vartotojui norint pamatyti maršrutus, yra kviečiamas *RouteCreationHelper* kontroleris, kuris iš duomenis gauna iš *OpenStreetMap.org* API.



3.7 pav. Panaudojimo atvejo „Atlikti maršruto analizę“ sekų diagrama

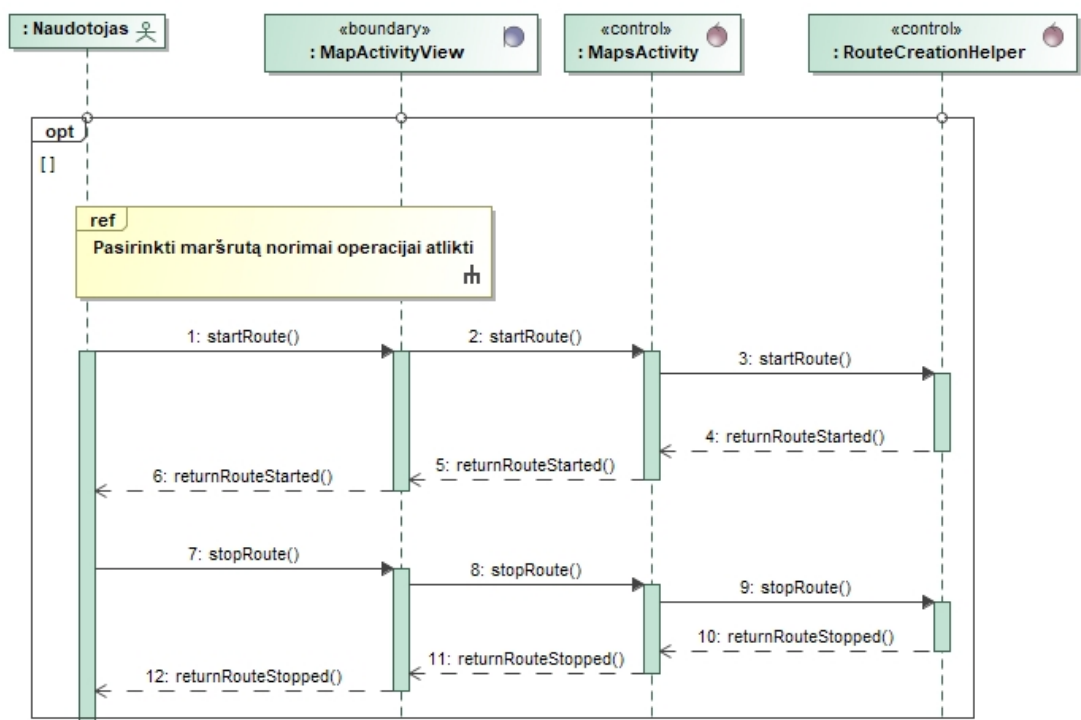
3.7 paveiksle yra „Atlikti maršruto analizę“ sekų diagrama, kurioje parodoma, kaip atliekama analizė. Vartotojui norint atlikti maršruto analizę, yra kviečiamas *AnalysisHelper* kontroleris, kuris pasirinktą maršrutą išanalizuoja naudojant *Analysis* ir *LocationUtils* pagalba.





3.8 pav. Panaudojimo atvejo „Peržiūrėti analizės rezultatus“ sekų diagrama

3.8 paveiksle yra „Peržiūrėti analizės rezultatus“ sekų diagrama, kurioje parodoma, kaip atliekama analizė. Vartotojui norint pamatyti atliktą maršruto analizę, yra kviečiamas *AnalysisHelper* kontroleris, kuris parodo analizės rezultatus pasitelkiant *LocationUtils*.



3.9 pav. „Slėpti įrenginio poziciją“ sekų diagrama

3.9 paveiksle yra „Slėpti įrenginio poziciją“ sekų diagrama, kurioje parodoma, kaip atliekamas įrenginio vietos slėpimas. Vartotojui norint paslėpti įrenginio vietą yra kviečiamas *MapsActivity*

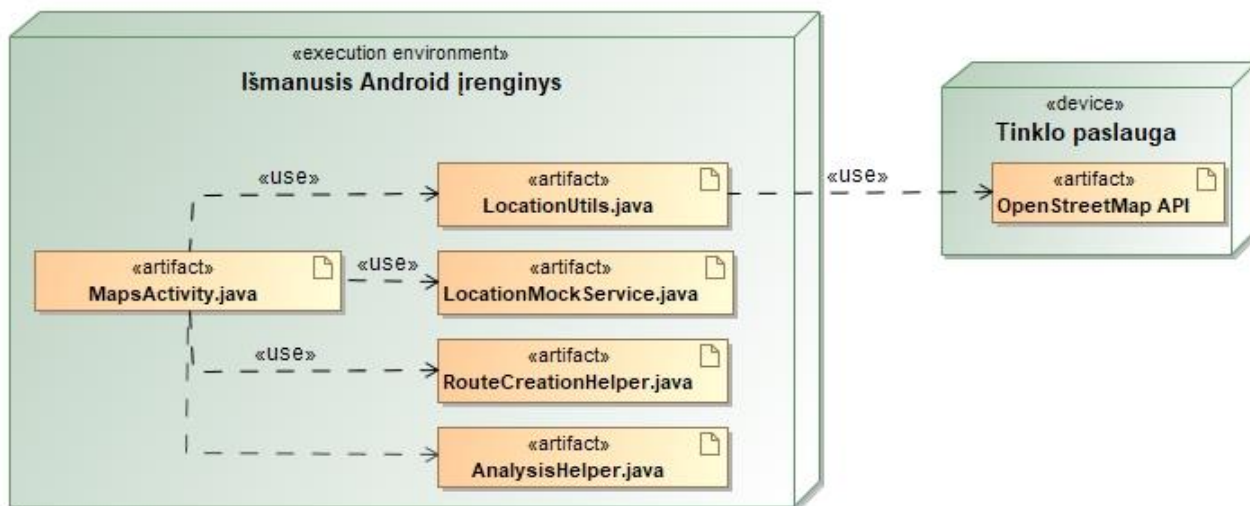
kontroleris, kuris pasiima maršrutą iš *RouteCreationHelper* ir jį panaudoja paleisti tarnybą, kuris pradeda slėpti vietą. Tada vartotojui paspaudus mygtuką, kad baigti slėpimą, tarnyba yra išjungiamas.

### 3.5. Programinių komponentų architektūra

Komponentų, komponentų realizavimo ir diegimo diagramos sudaro realizacijos modelį. Komponentų diagrama parodo iš kokių komponentų susideda „Android“ programa.

#### Diegimo modelis

Programos diegimo modelis pateiktas 3.10 paveiksle.



3.10 pav. „Hide Yourself“ programos diegimo modelis

### 3.6. Realizavimo dalies išvados

Tiriamąjį darbo metu buvo sukurta realistiškai įrenginio vietą slepianti programa pavadinimu „Hide Yourself“, kuri gali ne tik išanalizuoti kitų žmonių įrašytus realius maršrutus, gauti taškus, maršruto vidutinį greitį, tikslumą, laiką bei juos panaudoti vartotojo įrenginio slėpimui, imituojant, kad jis juda šiuo maršrutu, bet ir automatiškai generuoti maršrutą su vartotojo nurodytais parametrais, tokiais kaip pradžios taškas, pabaigos taškas, greitis, tikslumas. Maršrutai gali būti išsaugojami ir panaudojami kitą kartą. Taip pat maršrutai gali būti eksportuojami .GPX formatu. Toliau bus atliekamas eksperimentinis sprendimo tyrimas, programa bus testuojama.

## 4. EKSPERIMENTINIS IŠMANIŪJŲ ĮRENGINIŲ VIETOS SLĖPIMO SPRENDIMO TYRIMAS

Eksperimento tikslas yra paslėpti „Android“ įrenginio vietą, didžiausią dėmesį skiriant koordinatų slėpimui naudojant maršrutus pateiktus GPX failais, arba nurodytus iš *openstreetmap.org*. Išmaniųjų įrenginių vietos slėpimo eksperimento rezultatai parodys ar programa yra sukurta korektiškai ir, ar gali būti panaudota slėpti įrenginio vietą.

### 4.1. Eksperimento planas

Išmanaus įrenginio vietos klastojimo įrankio testavimui svarbu pasirinkti maršrutą kuris būtų viešai prieinamas ir lengvai panaudojamas.

Išmanaus įrenginio vietos klastojimo algoritmo testavimas susideda iš šių punktų:

1. maršruto įkėlimo ar pasirinkimo, ar kūrimo;
2. maršruto analizavimo;
3. jeigu maršrutas kuriamas, tai nustatymų parinkimo;
4. vietos klastojimo tarnybos paleidimo;
5. patikrinimo ar vietos klastojimas veikia su programa „Google Maps“.

#### Testavimo duomenys parsisiųsti iš svetainės:

<http://api.openstreetmap.org/api/0.6/trackpoints?bbox=0,51.5,0.25,51.75&page=1> (2018-05-12)

Duomenys pateikiami ir gaunami .GPX formatu, bet juose saugojamą informaciją galima lengvai nuskaityti naudojant XML schemą.

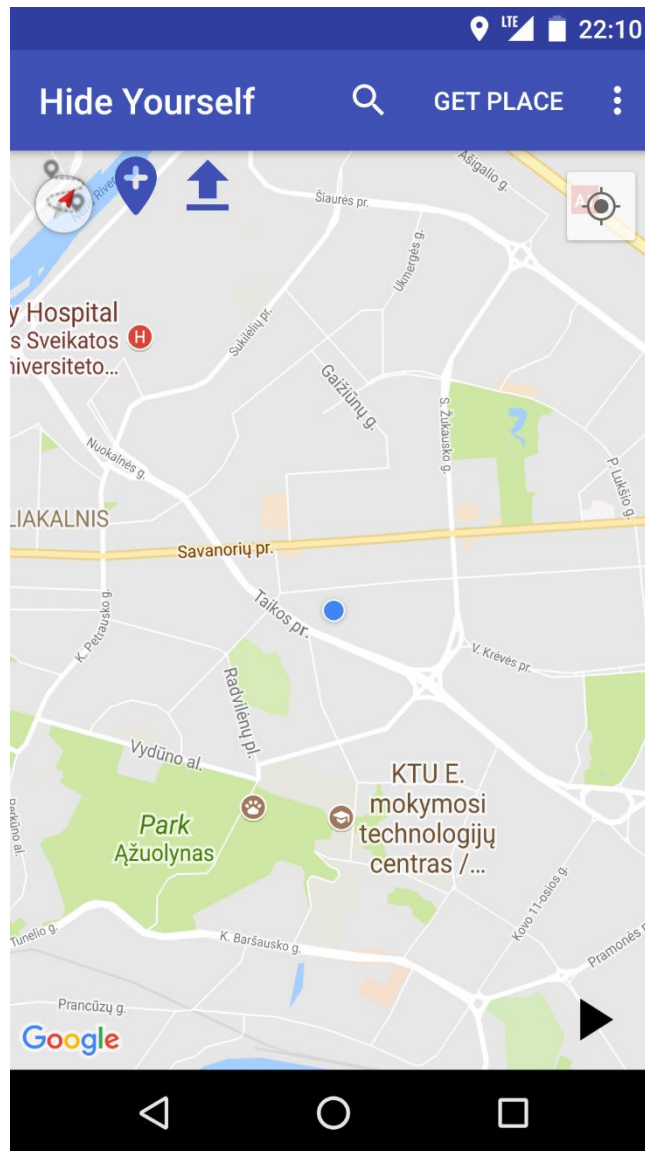
#### Testavimo duomenys:

```
<?xml version="1.0" encoding="UTF-8"?>
<gpx version="1.0" creator="OpenStreetMap.org" xmlns="http://www.topografix.com/GPX/1/0">
  <trk>
    <name>7_Feb_2018_11_27_46_2018_02_07_11_27_46.gpx</name>
    <desc>c2c</desc>
    <url>https://api.openstreetmap.org/user/h2g2bob/traces/2600305</url>
    <trkseg>
      <trkpt lat="51.5423441" lon="0.0750939">
        <time>2018-02-07T11:57:44Z</time>
      </trkpt>
```

```
<trkpt lat="51.5424342" lon="0.0748394">
  <time>2018-02-07T11:57:45Z</time>
</trkpt>
<trkpt lat="51.5425184" lon="0.0745874">
  <time>2018-02-07T11:57:46Z</time>
</trkpt>
<trkpt lat="51.5425902" lon="0.0743377">
  <time>2018-02-07T11:57:47Z</time>
</trkpt>
<trkpt lat="51.5426875" lon="0.0740307">
  <time>2018-02-07T11:57:48Z</time>
</trkpt>
<trkpt lat="51.5427542" lon="0.0737344">
  <time>2018-02-07T11:57:49Z</time>
</trkpt>
...
<trkpt lat="51.5308370" lon="0.0003070">
  <time>2017-12-19T18:18:08Z</time>
</trkpt>
<trkpt lat="51.5308790" lon="0.0001790">
  <time>2017-12-19T18:18:11Z</time>
</trkpt>
<trkpt lat="51.5308950" lon="0.0001360">
  <time>2017-12-19T18:18:12Z</time>
</trkpt>
<trkpt lat="51.5309130" lon="0.0000960">
  <time>2017-12-19T18:18:13Z</time>
</trkpt>
</trkseg>
</trk>
</gpx>
```

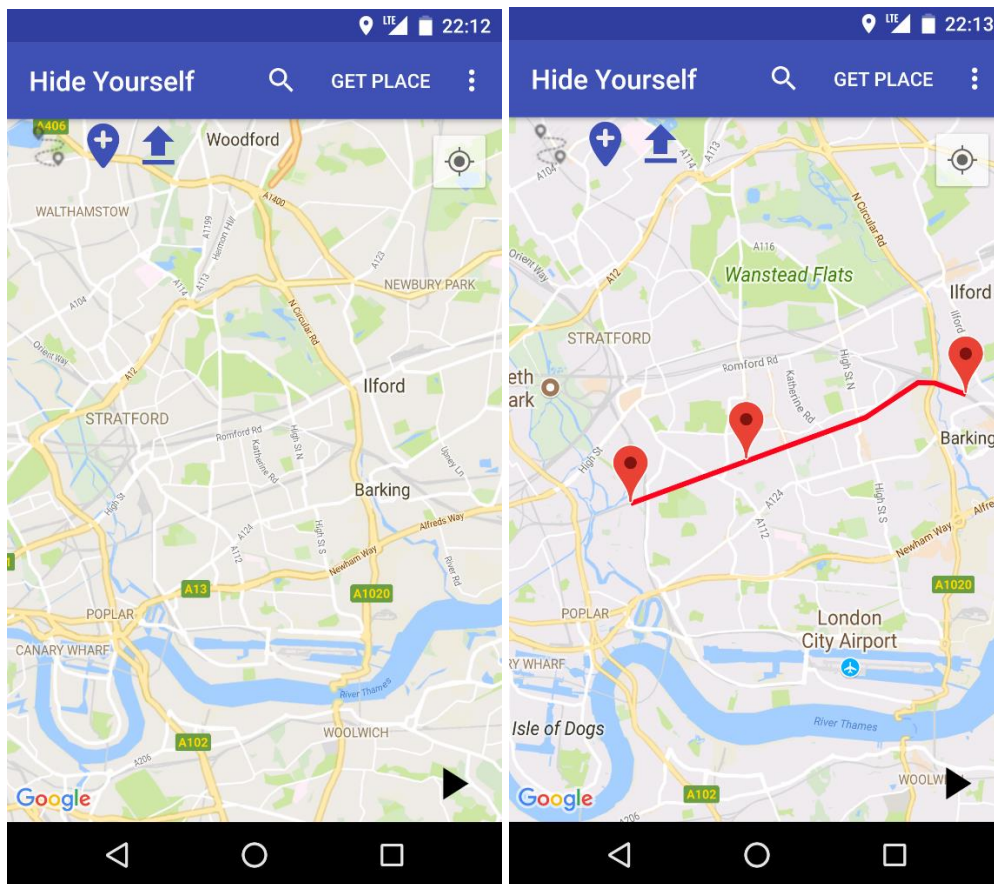
## 4.2. Eksperimento rezultatai

Atlikus eksperimentą, gauti rezultatai pateikiami atskiruose paveikslėliuose. Eksperimentas buvo atliktas vietą keičiant „Hide Yourself“ programa, tolimesniuose paveiksluose atvaizduojamas eksperimento atkartojimas. Taip pat padaryta lentelė, kuri parodo ar sukurta programa atitinka analizėje ir projektavime išskeltus reikalavimus.



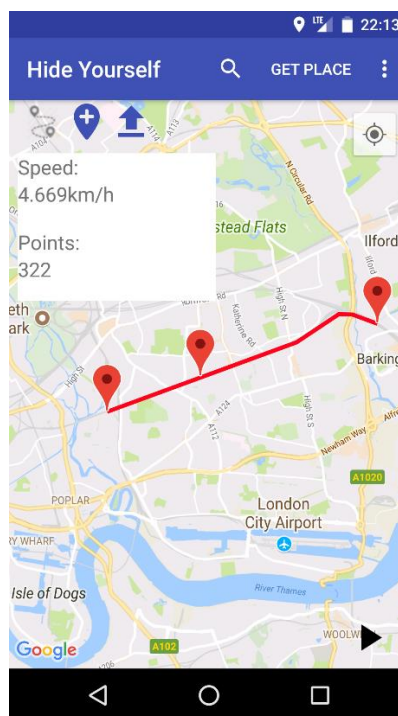
4.1 pav. Programos pradinis langas

4.1 paveiksluke matome vaizdą tik įsijungus pozicijos keitimo programą. Matosi, kad esame Lietuvoje, Kaune, prie Taikos prospekto. Viršuje matosi maršruto suradimo, vietos parinkimo mygtukas bei maršruto failo įkėlimo mygtukas. Apačioje dešinėje pusėje yra paleidimo mygtukas, kurį paspaudus jis tampa sustabdymo mygtuku.



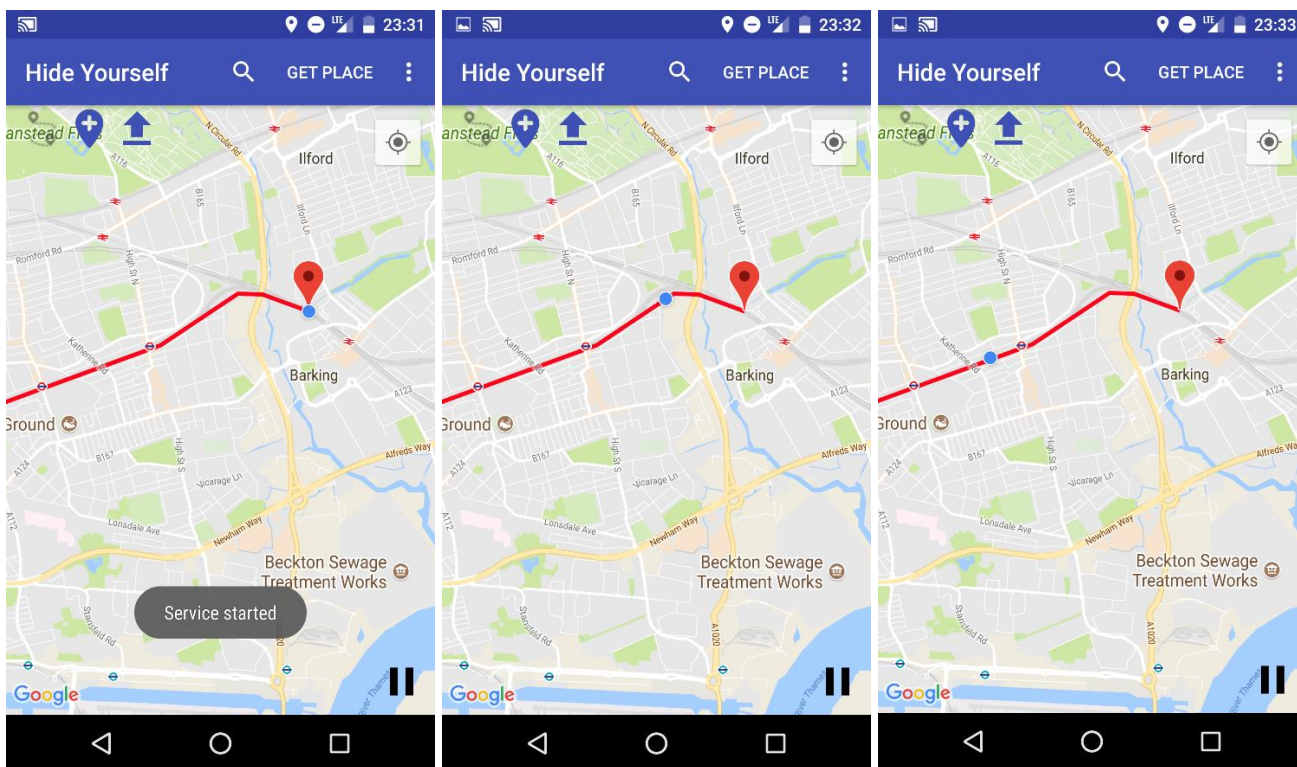
4.2 pav. Maršruto vietos pasirinkimas

4.2 paveiksluke, kairėje pusėje matome pasirinktą vietą maršrutui rasti. Tai yra Jungtinėje karalystėje, Londono centre, Stratfordo rajone. Paspaudžiamas maršruto pasirinkimo mygtukas, ir dešinėje matomas gautas maršrutas.



4.3 pav. Maršruto analizė

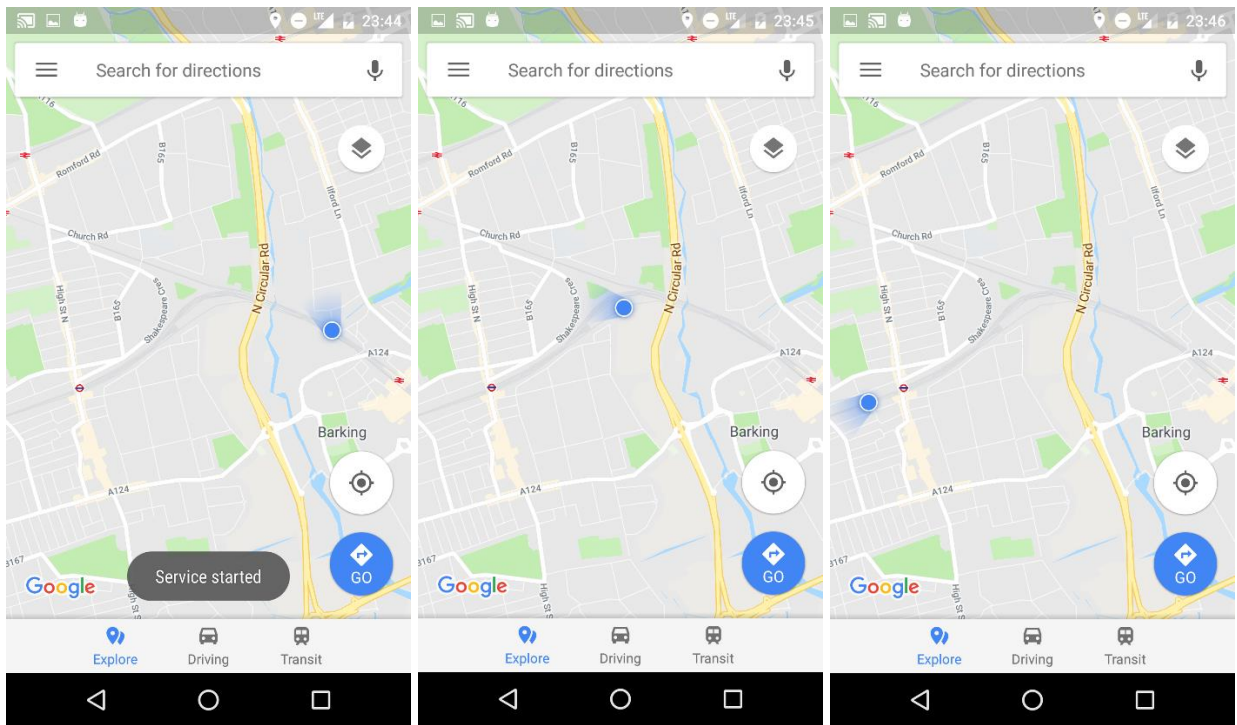
4.3 paveiksle matoma pasirinkto maršruto analizė, ji parodoma paspaudus dešinėje viršuje esantį trijų taškų mygtuką ir paspaudus ant analizės pasirinkimo. Nustatyti maršruto duomenys. Maršrutas susideda iš 322 taškų. Vidutinis greitis 4.669 km/h. Matoma pažymėta maršruto pradžia, vidurys ir pabaiga.



#### 4.4 pav. Vartotojo vietos imitavimas maršrutu funkcijos veikimas „Hide Yourself“ programoje

4.4 paveiksle matomas programos veikimas „Hide Yourself“ programoje. Pasirinkus maršrutą, atlikta jo analizė, gauti 322 maršruto taškai, gautas kelias, kurio vidutinis greitis 4.669km/h. Kairiajame paveiksliuke vaizduojama, kad paleidus klastojimo tarnybą vieta pasikeičia ir įrenginys programėlėje imituoja įrenginio judėjimą prieš tai pasirinktu maršrutu. Viduriniame paveiksle matome programos vaizdą ir kaip įrenginio vieta pasikeitė po 1 minutės. Dešiniajame paveiksliuke matome vaizdą koku atstumu įrenginio vieta pasikeitė po 2 minučių nuo paleidimo pradžios. Šie įrenginio vietos pakeitimai atitinkamai matosi 4.5 paveiksliuke, „Google Maps“ programėlėje.

Paleidus klastojimo tarnybą vieta pasikeičia ir įrenginys programėlėje pradeda judėti prieš tai pasirinktu maršrutu. Kairiajame paveiksliuke matoma vieta tik paleidus klastojimą. Viduriniame paveiksliuke matome programos vaizdą, kaip įrenginio vieta pasikeitė po 1 minutės, o dešiniajame paveiksliuke - po 2 minučių nuo paleidimo pradžios. Kiekvienas taškas yra atvaizduojamas kas sekundę, taigi maršrutas patampa 322 sekundžių keliu.



#### 4.5 pav. Vartotojo vietos imitavimas maršrutu funkcijos veikimas „Maps“ programėlėje

1 priede, paveiksluke parodyta testavimo maršruto „Run Tracker“ programos ekrano kopijos.

2 priede, paveiksluke parodyta testavimo maršruto „LocationTracker“ programos ekrano kopijos.

3 priede, paveiksluke parodyta testavimo maršruto „Glympse“ programos ekrano kopijos.

4-6 prieduose, paveikslukuose parodytas naujas testavimo maršrutas „Hide Yourself“, „Google Maps“ ir „Run Tracker“ programose.

Matoma, kad programa atvaizduoja įrenginio judėjimą testavimo duomenų maršrutu.

### 4.3. Sprendimo veikimo ir savybių analizė, kokybės kriterijų įvertinimas

4.1 lentelė parodo, kokie reikalavimai buvo išskirti planuojamai sukurti programai. Remiantis lentele galime teigti, kad sukurtoji programa buvo įgyvendinta išpildant visus išsikeltus reikalavimus.

4.1 lentelė. Programos atitikimo palyginimo lentelė

Palyginimo kriterijus	Planuota programa	Skurta programa
Nemokama	Taip	Taip
Be reklamos	Taip	Taip
Nereikalingos telefono administratoriaus privilegijos (root)	Taip	Taip
Vietų istorija	Taip	Taip



Mėgstamiausios vietos	Taip	Taip
Vietos paieška	Taip	Taip
Kitų žmonių maršrutų naudojimas	Taip	Taip
Maršruto sukūrimas	Taip	Taip
Pasirinkimas tarp kelių žemėlapių vaizdavimo tipų	Taip	Taip
Reikalinga nustatymuose įjungti „Mock location“	Taip	Taip
Reikalinga daug perteklinių teisių	Ne	Ne

#### 4.4. Eksperimento apibendrinimas

Sukurta programa „Hide Yourself“ turi vietos paiešką, maršrutų pasirinkimą, kelis žemėlapių vaizdavimo tipus, gali atvaizduoti GPX maršrutus ir nereikalauja daug perteklinių teisių. Programa keičia vartotojo įrenginio buvimo vietą, tai buvo patikrinta su 4 programomis, „Glympse“, „Google Maps“, „Run Tracker“, „LocationTracker“. Visose programose buvo matoma pakeista įrenginio vieta, kuri yra netikra. Taip pat buvo daromas papildomas testavimas, su kitokiu, nauju maršrutu.

Eksperimento metu pasirinkus tikslią imituojamą vietą žemėlapyje, gavus norimą maršrutą buvo atlikta jo analizė. Analizės metu gauti 322 maršruto taškai ir nustatytas vidutinis greitis 4.669 km/h. Kiekvienas taškas yra atvaizduojamas kas sekundę, taigi maršrutas patampa 322 sekundžių keliu. Paleidus klastojimo tarnybą vieta pradeda keistis ir įrenginys „Google Maps“ programėlėje pradeda judėti prieš tai pasirinktu maršrutu.

## 5. REZULTATŲ APIBENDRINIMAS IR IŠVADOS

1. Išmaniuosiuose įrenginiuose esančios techninės priemonės leidžia stebėti vartotojų buvimo vietą dažnai jiems apie tai net nežinant. Siekiant apsisaugoti nuo sekimo, reikalingos priemonės padedančios vartotojams savo nuožiūra pakeisti tikrąją įrenginio buvimo vietą. Taigi, šio darbo tikslas yra leisti vartotojui savo nuožiūra pakeisti išmaniojo įrenginio vietos (koordinacinių) duomenis taip paslepiant savo tikrąją buvimo vietą.
2. Atlikus išmaniųjų įrenginių operacinių sistemų pasiskirstymą rinkoje nuspręsta kurti programą „Android“ platformoje, kadangi ji yra populiariausia.
3. Tiriamajame darbe atlikus vietos keitimo būdų analizę, įvertinus visus kriterijus, pasirinkta kurti programą, kuriai nereikia telefono administratoriaus privilegijų. Taip pat, palyginus kelias daugiausiai tokio tipo parsuončiamas programas iš „Play Store“, nei viena iš palygintų programų neturi galimybės realistiškai klastoti vartotojo įrenginio buvimo vietą, todėl kuriamame sprendime buvo siekiama kuo realistiškiau imituoti vartotojo buvimo vietą.
4. Išanalizavus GPS slėpimo galimybes buvo pasirinkti du būdai – atvaizdavimas jau kitų žmonių nueitų/nuvažiuotų maršrutų, arba maršruto sukūrimas automatiškai. Pastaruoju atveju vartotojas pasirenka vietų taškus, greitį, bet klastojimo metu yra pridėdamos paklaidos, kurios prideda realistiškumo.
5. Pagrindinės „Android“ programos funkcijos: naujos vietos pasirinkimas, naujo maršruto sukūrimas, išsaugotos vietos pasirinkimas, vietos išsaugojimas į mėgstamiausių sąrašą, maršruto įkėlimas, maršruto pasirinkimas, maršruto analizė. Nefunkciniuose reikalavimuose ir programos kūrimo detaliam atvaizdavimui pasitelkiamos klasių ir veiklos sekų diagramos.
6. Tiriamojo darbo realizacijos palengvinimui pasitelktos robastiškumo, sekų ir programos loginės architektūros diagramos. Sukurtos sistemos pagrindinės posistemės, išsiaiškinta naudojamų maršrutų duomenų saugojimo ir užrašymo struktūra, kuri užrašoma .GPX formatu, XML schema.
7. Tiriamojo darbo metu buvo sukurta realistiškai įrenginio vietą slepianti programa „Hide Yourself“. Programa gali ne tik išanalizuoti kitų žmonių įrašytus realius maršrutus, gauti taškus, maršruto vidutinį greitį ir laiką, bei juos panaudoti vartotojo įrenginio slėpimui, apsimetant, kad jis juda šiuo maršrutu, bet ir automatiškai generuoti maršrutą su vartotojo nurodytais parametrais, tokiais kaip pradžios taškas, pabaigos taškas, greitis, tikslumas. Maršrutai gali būti išsaugojami ir panaudojami kitą kartą. Taip pat maršrutai gali būti eksportuojami .GPX formatu.
8. Eksperimento metu pasirinkus tikslią imituojamą vietą žemėlapyje, gavus norimą maršrutą buvo atlikta jo analizė. Analizės metu gauti 322 maršruto taškai ir nustatytas vidutinis greitis

4.669 km/h. Kiekvienas taškas yra atvaizduojamas kas sekundę, taigi maršrutas patampa 322 sekundžių keliu. Paleidus klastojimo tarnybą vieta pradeda keistis ir įrenginys „Google Maps“ programėlėje pradeda judėti prieš tai pasirinktu maršrutu.

9. Tiriamojo darbo metu sukurta vietos slėpimo programa gali būti pritaikyta GPS koordinatijų nustatymo ir klastojimo būdų tyrimui, ir tolimesnei analizei.

## 6. LITERATŪRA

- [1] C. Williams, „The Telegraph,“ Telegraph Media Group, 2011. [Tinkle]. Available: <https://www.telegraph.co.uk/technology/apple/8464122/Apple-iPhone-tracks-users-location-in-hidden-file.html>. [Kreiptasi 15 04 2018].
- [2] Brenda Stolyar, Julian Chokkattu, „Digital Trends,“ Designtecnica Corporation, 2015. [Tinkle]. Available: <https://www.digitaltrends.com/mobile/android-distribution-news/>. [Kreiptasi 01 05 2018].
- [3] „StatCounter Global Stats,“ StatCounter , 2018. [Tinkle]. Available: <http://gs.statcounter.com/os-market-share/mobile-tablet/worldwide/#monthly-201704-201804-bar>. [Kreiptasi 14 05 2018].
- [4] „NetApplications,“ 2018. [Tinkle]. Available: <https://netmarketshare.com/operating-system-market-share.aspx?options>. [Kreiptasi 14 05 2018].
- [5] Adrienne Porter Felt, Elizabeth Ha, Serge Egelman, Ariel Haney, Erika Chin, David Wagner, „Android Permissions: User Attention, Comprehension, and Behavior,“ Computer Science Department School of Information. University of California, Berkeley, 2012. [Tinkle]. Available: <http://doi.acm.org/10.1145/2335356.2335360>. [Kreiptasi 23 04 2018].
- [6] D. Wenyun, . Q. Meikang, Q. Longfei ir C. Longbin, „Who Moved My Data? Privacy Protection in Smartphones,“ 19 01 2017. [Tinkle]. Available: <https://ieeexplore.ieee.org/abstract/document/7823333/>. [Kreiptasi 14 05 2018].
- [7] M. Qiu, J. Li ir Z. Zong, „Privacy Protection for Preventing Data Over-Collection in Smart City,“ IEEE, 2016. [Tinkle]. Available: <https://ieeexplore.ieee.org/document/7054465/>. [Kreiptasi 14 05 2018].
- [8] A. Okonski, „Privacy Concerns in Android Devices,“ Department of Information Technology, Sweden, 12 2014. [Tinkle]. [Kreiptasi 17 03 2018].
- [9] NAZMULIDRIS, „Android Location Providers – gps, network, passive – Tutorial,“ 20 10 2010. [Tinkle]. Available: <https://developerlife.com/2010/10/20/gps/>. [Kreiptasi 14 05 2018].
- [10] Wenbo He, Xue Liu, Mai Ren, „Location Cheating: A Security Challenge to Location-Based Social Network Services,“ Distributed Computing Systems (ICDCS), 2011 31st International Conference on, 2011. [Tinkle]. Available: 10.1109/ICDCS.2011.42. [Kreiptasi 03 04 2018].

- [11] N. P. Hoang, Y. Asano ir M. Yoshikawa, „Your neighbors are my spies: Location and other privacy concerns in GLBT-focused location-based dating applications,“ 19 02 2017. [Tinkle]. Available: <https://ieeexplore.ieee.org/abstract/document/7890236/>. [Kreiptasi 14 05 2018].
- [12] Kassem Fawaz, Kang G. Shin, „Location Privacy Protection for Smartphone Users. In Proceedings of the 2014,“ Conference on Computer and Communications Security, New York, NY, USA, 2014. [Tinkle]. Available: <http://dx.doi.org/10.1145/2660267.2660270> . [Kreiptasi 05 01 2017].
- [13] Amit Kushwaha, Vineet Kushwaha, „Location Based Services using Android Mobile Operating System,“ *Department of Electronics & Communication Engineering, India*, nr. 2231-1963, pp. 14-20 , 2011.
- [14] Alastair R. Beresford, Andrew Rice, Nicholas Skehin, Ripduman Sohan, „MockDroid: trading privacy for application functionality on smartphones. In Proceedings of the 12th Workshop on Mobile Computing Systems and Applications (HotMobile '11),“ New York, NY, USA, 2011. [Tinkle]. Available: <http://dx.doi.org/10.1145/2184489.2184500>. [Kreiptasi 05 01 2017].
- [15] Christoph Hochreiner, Markus Huber, Georg Merzdovnik, Edgar Weippl, „Towards Practical Methods to Protect the Privacy of Location Information with Mobile Devices. In Proceedings of the 7th International Conference on Security of Information and Networks,“ New York, NY, USA, 2014. [Tinkle]. Available: <http://dx.doi.org/10.1145/2659651.2659680>. [Kreiptasi 05 01 2017].
- [16] Kushwaha, Amit ir Vineet Kushwaha, „Location Based Services Using Android Mobile Operating System,“ Indian Institute of Information Technology, Allahabad, International Journal of Advances in Engineering & Technology, 2011, pp. 14-20.
- [17] Gary M. Weiss, Samuel A. Moore, Kwapisz, Jennifer R., „Activity Recognition Using Cell Phone Accelerometers,“ Department of Computer and Information Science, ACM SIGKDD Explorations Newsletter, 2011.
- [18] Se-Kyeong Heo, In-Cheol Kim, Kim, Joo-Hee, Sang-Ha Nam, „Design of an Activity Recognition System Using Smartphone Accelerometer,“ Yonsei University , KIPS Transactions on Software and Data Engineering, 2013, pp. 49-54.
- [19] „Hacking Team's Tradecraft and Android Implant,“ įtraukta *The Citizen Lab Police Story*, 2014.
- [20] A. K. Kini ir S. A. Kulkarni, „Real time implementation of fake location generation algorithm to protect location privacy in location based services,“ 13 10 2017. [Tinkle]. Available: <https://ieeexplore.ieee.org/abstract/document/8125831/>. [Kreiptasi 14 05 2018].

[21] „Fake GPS Location Spoofer Free,“ Google, 2016. [Tinkle]. Available:

<https://play.google.com/store/apps/details?id=com.incorporateapps.fakegps.fre&hl=en>.

[Kreiptasi 05 01 2017].

[22] IncorporateApps, „Fake GPS Go Location Spoofer PRO,“ Google, 2016. [Tinkle]. Available:

<https://play.google.com/store/apps/details?id=com.incorporateapps.fakegps>. [Kreiptasi 07 01

2017].

[23] Lexa, „Fake GPS location,“ Google, 2016. [Tinkle]. Available:

<https://play.google.com/store/apps/details?id=com.lexa.fakegps&hl=en>. [Kreiptasi 08 01

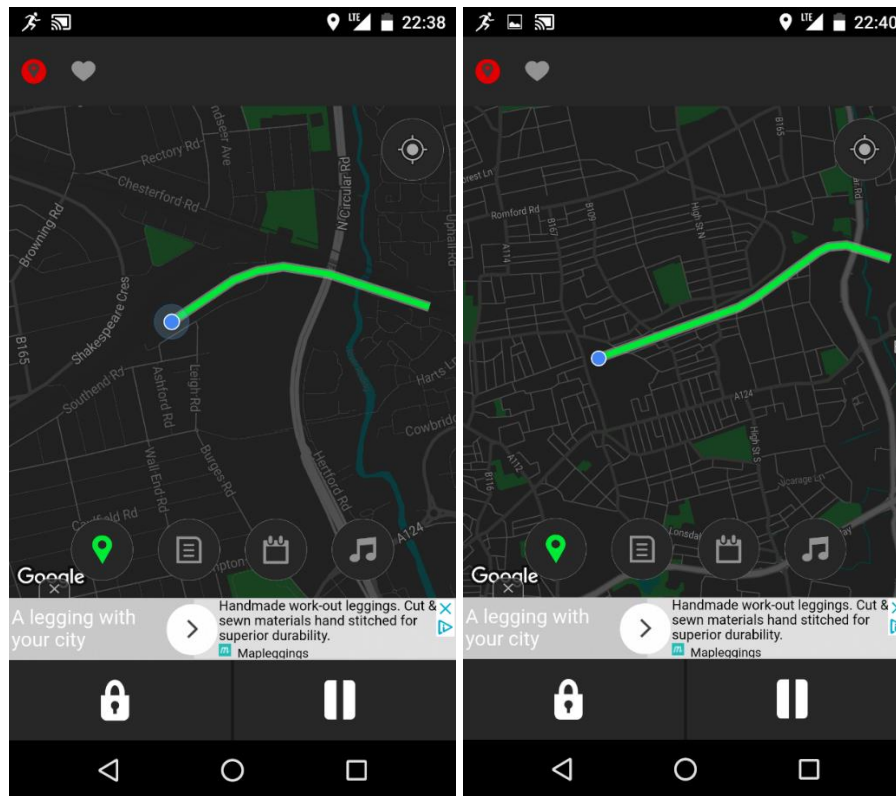
2017].

[24] FansBoy, „Mock GPS Pro,“ Google, 2012. [Tinkle]. Available:

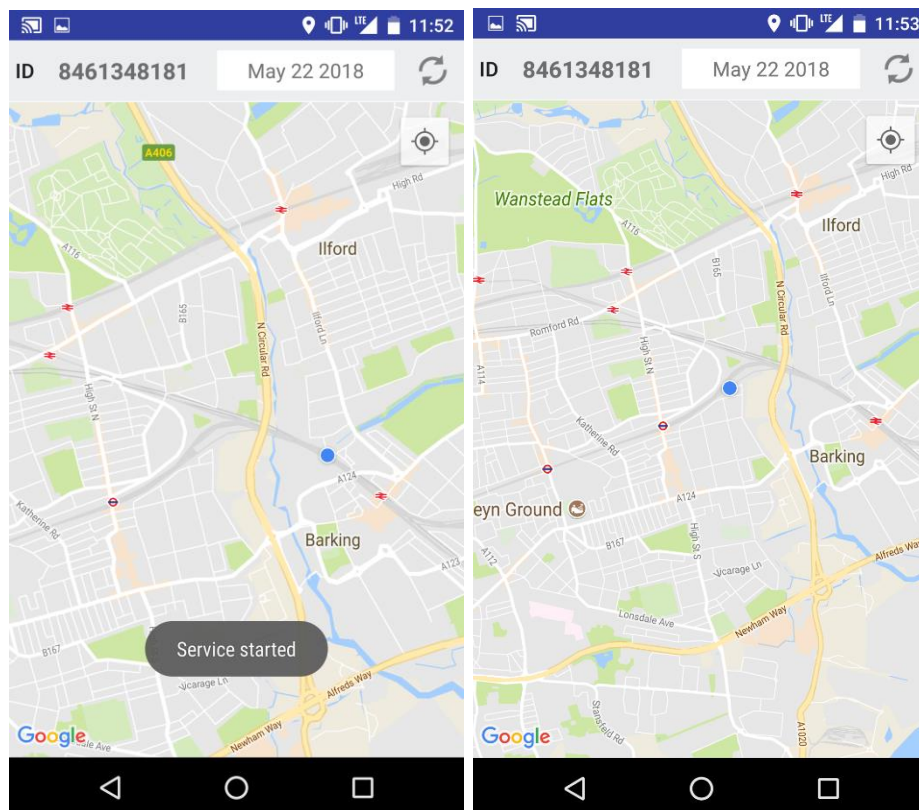
<https://play.google.com/store/apps/details?id=com.tim.apps.mockgps>. [Kreiptasi 09 01 2017].

## 7. PRIEDAI

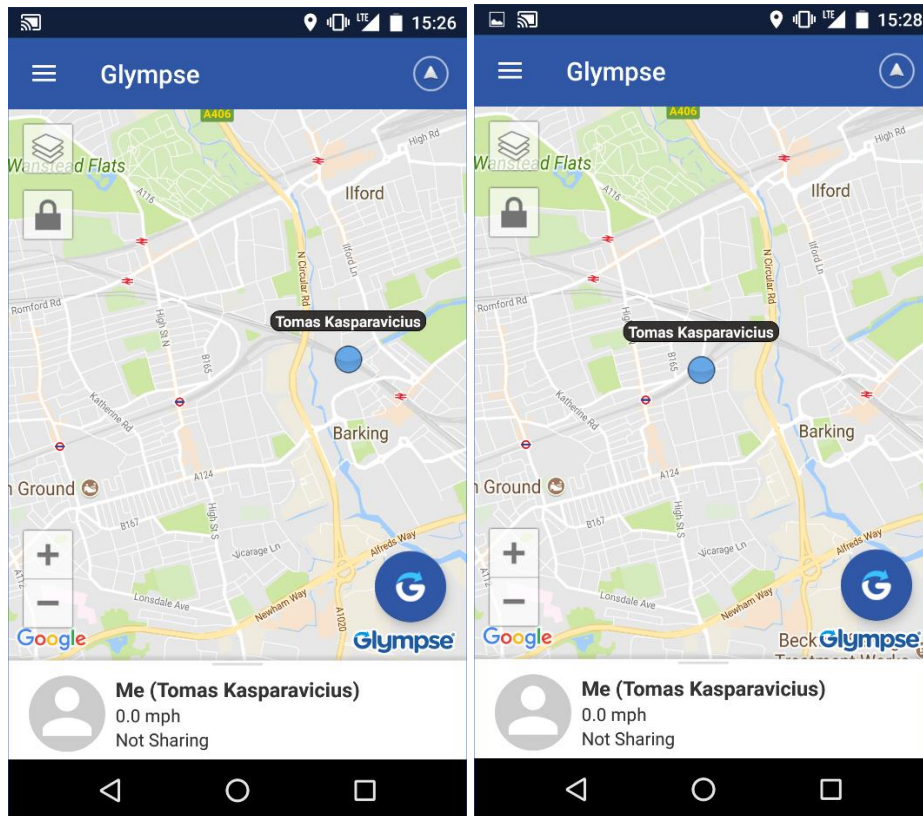
### 7.1. Priedas. Testavimo maršruto ekrano kopijos „Run Tracker“ programoje



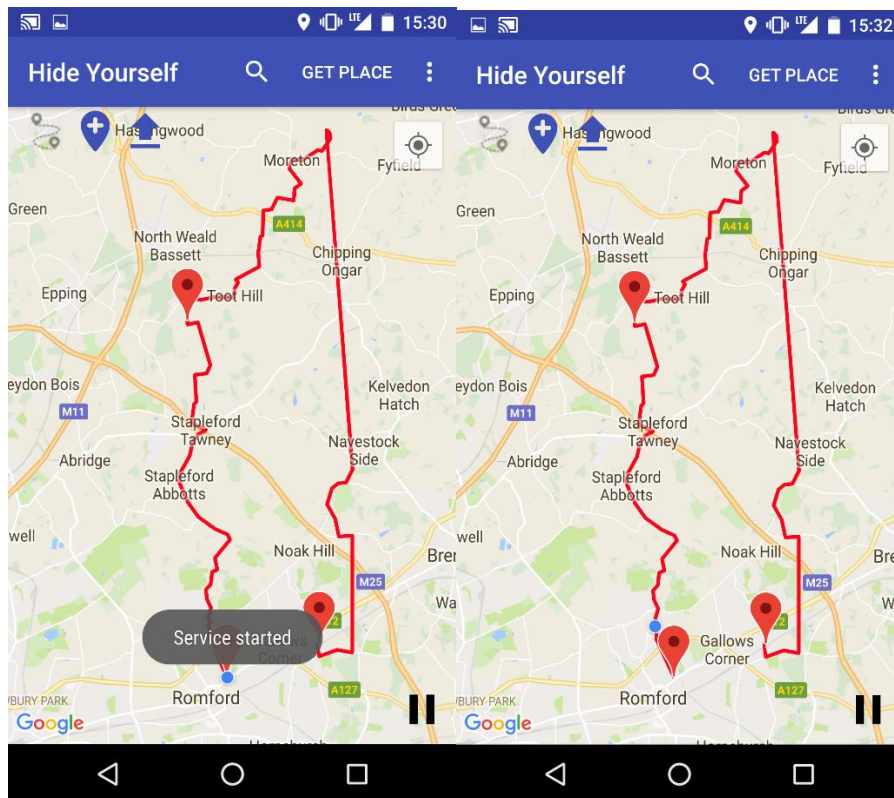
### 7.2. Priedas. Testavimo maršruto ekrano kopijos „LocationTracker“ programoje



### 7.3. Priedas. Testavimo maršruto ekrano kopijos „Glympse“ programoje

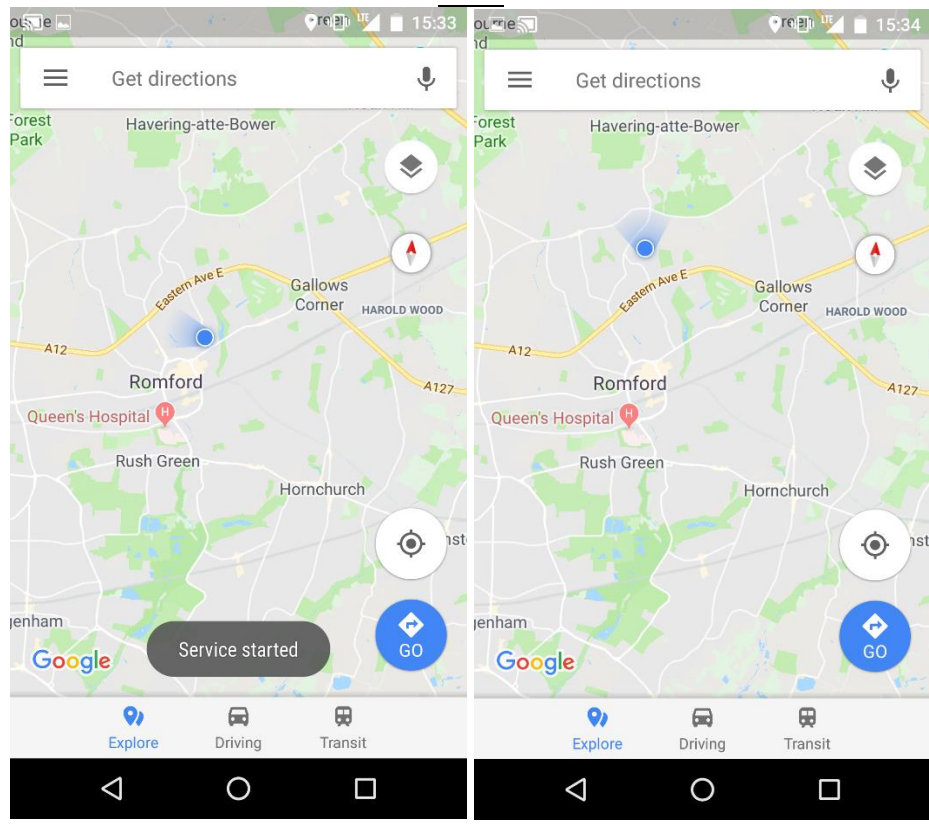


### 7.4. Priedas. Naujo testavimo maršruto ekrano kopijos „Hide Yourself“ programoje





**7.5. Priedas. Naujo testavimo maršruto ekrano kopijos „Google Maps“ programoje**



**7.6. Priedas. Naujo testavimo maršruto ekrano kopijos „Run Tracker“ programoje**

