



**Kauno technologijos universitetas**

Informatikos fakultetas

**Žmogaus sportinės veiklos atpažinimas ir registravimas iš  
nešiojamojo įrenginio duomenų**

Baigiamasis magistro projektas

---

**Bernardas Zokas**  
Projekto autorius

**Mantas Lukoševičius**  
Vadovas

---

**Kaunas, 2018**



**Kauno technologijos universitetas**

Informatikos fakultetas

# **Žmogaus sportinės veiklos atpažinimas ir registravimas iš nešiojamojo įrenginio duomenų**

Baigiamasis magistro projektas

Programų sistemų inžinerija (621E16001)

---

**Bernardas Zokas**

Projekto autorius

**Mantas Lukoševičius**

Vadovas

**Doc. dr. Šarūnas Packevičius**

Recenzentas

---

**Kaunas, 2018**



**Kauno technologijos universitetas**

Informatikos fakultetas

Bernardas Zokas

## **Žmogaus sportinės veiklos atpažinimas ir registravimas iš nešiojamojo įrenginio duomenų**

Akademinio sąžiningumo deklaracija

Patvirtinu, kad mano, Bernardo Zoko, baigiamasis projektas tema „Žmogaus sportinės veiklos atpažinimas ir registravimas iš nešiojamojo įrenginio duomenų“ 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)

# Turinys

<b>Turinys</b> .....	<b>4</b>
<b>Įvadas</b> .....	<b>11</b>
<b>1. Analitinė dalis</b> .....	<b>12</b>
1.1. Tikslas .....	12
1.2. Egzistuojantys panašūs moksliniai tyrimai .....	13
1.3. Egzistuojančios programėlės sportinei veiklai fiksuoti .....	14
1.3.1. Runtastic Sit-Ups Abs Workout .....	15
1.3.2. Squats Workout .....	15
1.3.3. Push Ups Workout .....	15
1.3.4. Pull Ups Workout .....	16
1.3.5. S Health .....	16
1.4. Dažniausiai naudojami sprendimai ir technologijos .....	17
1.4.1. Jutikliai .....	17
1.4.2. Jutiklių sintezė .....	18
1.4.3. Mašininio mokymosi algoritmai.....	21
1.5. Programinės priemonės.....	24
1.5.1. Android.....	24
1.5.2. Android „SensorManager“ biblioteka .....	24
1.5.3. Bibliotekos realizuojančios mašininio mokymosi algoritmą.....	26
<b>2. Projektinė dalis</b> .....	<b>27</b>
2.1. Funkciniai reikalavimai.....	27
2.1.1. Duomenų modelis.....	27
2.1.2. Sistemos sudėtis.....	28
2.1.3. Funkciniai reikalavimai .....	31
2.2. Nefunkciniai reikalavimai.....	35
2.2.1. Reikalavimai sistemos išvaizdai .....	35
2.2.2. Reikalavimai panaudojimui .....	36
2.2.3. Reikalavimai vykdymo savybėms .....	37
2.2.4. Reikalavimai veikimo sąlygoms.....	39
2.2.5. Reikalavimai sistemos priežiūrai .....	40
2.2.6. Reikalavimai saugumui .....	40
2.2.7. Kultūriniai – politiniai reikalavimai .....	41
2.2.8. Teisiniai reikalavimai .....	42

2.3. Sistemos statinis vaizdas .....	42
2.3.1. Apžvalga .....	42
2.3.2. Paketų detalizavimas .....	43
2.4. Sistemos dinaminis vaizdas.....	45
2.4.1. Būsenų diagramos.....	45
2.4.2. Veiklos diagramos .....	47
2.4.3. Sekų diagramos.....	48
2.5. Sistemos išdėstymo vaizdas .....	52
2.6. Duomenų vaizdas .....	52
<b>3. Tyrimo dalis .....</b>	<b>54</b>
3.1. Tyrimo tikslas .....	54
3.2. Tyrimo aprašymas.....	54
3.3. Tyrimo eiga.....	54
3.3.1. Duomenų rinkimas.....	54
3.3.2. Duomenų savybių išgryninimas .....	55
3.3.3. Mašininio mokymosi algoritmų palyginimas .....	57
3.3.4. Sportinės veiklos pakartojimų skaičiavimas.....	58
3.4. Tyrimo išvados.....	60
<b>4. Eksperimento dalis .....</b>	<b>61</b>
4.1. Eksperimento tikslas .....	61
4.2. Eksperimento aprašymas.....	61
4.3. Eksperimentas ir jo rezultatai.....	61
4.4. Eksperimento išvados.....	64
<b>Išvados .....</b>	<b>65</b>
<b>Literatūros sąrašas .....</b>	<b>66</b>
<b>Terminai .....</b>	<b>70</b>
<b>Priedai.....</b>	<b>71</b>

## Paveikslų sąrašas

1 pav. Sportinės veiklos atpažinimo principinė schema.....	12
2 pav. Vienaąšio akselerometro architektūros schema [43] .....	17
3 pav. Giroskopo architektūros schema [18] .....	18
4 pav. Jutiklių sintezės diagrama .....	19
5 pav. Orientacijos išskaičiavimas pagal Kalman filtrą .....	20
6 pav. Bajeso klasių atskyrimas.....	21
7 pav. Neuroninio tinklo medelis .....	22
8 pav. Pasikartojantis neuroninis tinklas [42].....	23
9 pav. Sistemos duomenų modelis .....	27
10 pav. Panaudos atvejų diagrama .....	28
11 pav. Sistemos suskirstymas į paketus .....	43
12 pav. Paketas „Grafinė vartotojo sąsaja“ .....	43
13 pav. Paketas „Grafinės vartotojo sąsajos adapteris“ .....	44
14 pav. Paketas „Skaičiavimai“ .....	44
15 pav. Paketas „Duomenų bazės adapteris“ .....	45
16 pav. Būsenos diagrama „PA2 Treniruotės tikslo kūrimas“ .....	46
17 pav. Būsenos diagrama „PA3 Treniruotės tikslo redagavimas“ .....	46
18 pav. Būsenos diagrama „PA4 Automatinis sportinės veiklos atpažinimas“ .....	46
19 pav. Būsenos diagrama „PA6 Redaguoti rezultatus“ .....	47
20 pav. Veiklos diagrama „PA4 Automatinis sportinės veiklos atpažinimas“ .....	47
21 pav. Sekų diagrama „PA1 Peržiūrėti treniruočių tikslus“ .....	48
22 pav. Sekų diagrama „PA2 Treniruotės tikslo kūrimas“ .....	48
23 pav. Sekų diagrama „PA3 Treniruotės tikslo redagavimas“ .....	49
24 pav. Sekų diagrama „PA4 Automatinis sportinės veiklos atpažinimas“ .....	50
25 pav. Sekų diagrama „PA5 Duomenų registravimas“ .....	50
26 pav. Sekų diagrama „PA6 Redaguoti treniruotės rezultatus“ .....	51
27 pav. Sekų diagrama „PA7 Peržiūrėti treniruočių rezultatus“ .....	51
28 pav. Sistemos išdėstymo diagrama .....	52
29 pav. Duomenų bazės modelis .....	53
30 pav. Programinės įrangos veikimas algoritmas .....	54
31 pav. Atsispaudimai .....	56
32 pav. Pritūpimai .....	56
33 pav. Atsilenkimai.....	56
34 pav. Išfiltruoto signalo pavyzdys.....	57

35 pav. Mašininio mokymosi algoritmų atpažinimo palyginimas .....	58
36 pav. „Random Forest“ ir „Multilayer Perceptron“ sportinės veiklos atpažinimo palyginimas....	59
37 pav. Sportinių veiklų atpažinimo tikslumas .....	62
38 pav. Atsispaudimų atpažinimo rezultatų diagrama.....	63
39 pav. Pritupimų atpažinimo rezultatų diagrama.....	63
40 pav. Atsilenkimų atpažinimo rezultatų diagrama .....	63
41 pav. Mobiliosios programėlės pagrindinis meniu .....	71
42 pav. Mobiliosios programėlės tikslo pasirinkimas prieš sportinės veiklos registravimą .....	71
43 pav. Mobiliosios programėlės sportinės veiklos registravimo langas .....	72
44 pav. Mobiliosios programėlės treniruočių rezultatų peržiūros langas.....	72
45 pav. Mobiliosios programėlės treniruočių rezultatų redagavimo langas.....	73
46 pav. Mobiliosios programėlės treniruočių tikslų peržiūros langas.....	73
47 pav. Mobiliosios programėlės treniruočių tikslų kūrimo ir redagavimo langas .....	74

## Lentelių sąrašas

1 lentelė. Egzistuojančių programėlių palyginimo lentelė .....	14
2 lentelė. Android „SensorManager“ bibliotekos jutiklių nuskaitymo metodų aprašymas [44] .....	25
3 lentelė. Duomenys naudojami sistemos duomenų modelyje.....	27
4 lentelė. Mašininio mokymosi algoritmų atpažinimo palyginimas .....	58
5 lentelė. Eksperimento rezultatai .....	61



Zokas, Bernardas. Žmogaus sportinės veiklos atpažinimas ir registravimas iš nešiojamojo įrenginio duomenų. Magistro baigiamasis projektas / vadovas Mantas Lukoševičius; Kauno technologijos universitetas, Informatikos fakultetas.

Studijų kryptis ir sritis (studijų krypčių grupė): Informatikos inžinerija (E100), Programų sistemų inžinerija (E160).

Reikšminiai žodžiai: sportinės veiklos atpažinimas, jutikliai, išmanusis telefonas, mašininis mokymasis, atsispaudimai, pritūpimai, atsilenkimai.

Kaunas, 2018. 70 p.

### **Santrauka**

Šiame darbe kuriamas sportinių veiklų (atsispaudimų, pritūpimų ir atsilenkimų) atpažinimo ir registravimo mobiliosios programėlės prototipas, naudojantis išmaniojo telefono jutiklių surinktus duomenis ir mašininio mokymosi algoritmus.

- Pirmame skyriuje analizuojama mokslinė medžiaga žmogaus veiklų atpažinimui naudojant mobilųjį telefoną, egzistuojantys sprendimai, technologijos ir būdai projektui įgyvendinti.
- Antrame skyriuje sudaryta vartotojo reikalavimų specifikacija, suprojektuota ir realizuota programinė įranga sportinėms veiklos atpažinti.
- Trečiame skyriuje atliekamas tyrimas, kuriame analizuoti surinkti sportinių veiklų duomenys, lyginami mašininio mokymosi algoritmai.
- Ketvirtame skyriuje atliekamas eksperimentas sportinių veiklų atpažinimo tikslumui nustatyti.

Zokas, Bernardas. Human Sport Activities Identification and Registration from the Data of Portable Device. Master's Final Degree Project / supervisor assoc. Mantas Lukoševičius; Faculty of Informatics, Kaunas University of Technology.

Study field and area (study field group): Informatics Engineering (E100), Software Engineering (E160).

Keywords: sport activity recognition, sensors, smartphone, machine learning, push-ups, squats, sit-ups.

Kaunas, 2018. 70 pages.

### **Summary**

In this study, we are creating sports activities (push-ups, sit-ups and squats) recognition and tracking mobile application prototype using only smartphone sensors and a machine learning algorithms.

- The first chapter analyzes scientific material for the human activities recognition, existing solutions, technologies and methods for implementation of the project.
- In the second chapter, a specification of user's requirements is made; software for the sports activities recognition is designed and implemented.
- In the third chapter, the research of sports activities data analysis is made and machine learning algorithms are compared.
- In the fourth chapter an experiment is performed to evaluate the accuracy of the sports activities recognition rate.

## Ivadas

Šiuo metu naudojimas išmaniaisiais telefonais užima didelę mūsų gyvenimo dalį. Šiuose įrenginiuose gausu įvairiausių jutiklių, kaip mobiliojo ryšio antena, Wi-Fi ir Bluetooth radijas, mikrofonas, kamera, GPS, akcelerometras, giroskopas, magnetometras, šviesos ir atstumo jutiklis be kurių pagalbos šiandien neišsiverstume. Šių išmaniųjų telefonų galingi procesoriai gali ne tik įvykdyti pagrindines įrenginio funkcijas, bet ir lygiagrečiai atlikti skaičiavimus su dideliais duomenų kiekiais. Turėdami galingus nešiojamuosius telefonus galime jų jutiklių fiksuojamus duomenis panaudoti žmogaus judesių atpažinimui.

Žmogaus judesių atpažinimas iš išmaniojo telefono jutiklių duomenų nėra nauja mokslo tema. Jau daugiau kaip 30 metų mokslininkai bando, atlieka tyrimus ir sėkmingai kuria sistemas, kurių veikimas pagrįstas judesių atpažinimu. Šiais laikais tokios sistemos puikiai žinomos kaip mobiliosios programėlės, kurių pagrindinis tikslas yra sekti mūsų judesius ir mums priminti apie tai kiek žingsnių nuėjome, kiek sudeginome kalorijų, kokį atstumą įveikėme, seka širdies pulsą, fiksuoja miego režimą bei kita. Iš to mums kilo idėja apie mobiliąją programėlę, kuri panašiu veikimo principu galėtų atpažinti ir skaičiuoti šiek tiek sudėtingesnes sportines veiklas, kaip atsispaudimai, pritūpimai ir atsilenkimai.

Kuriama mobilioji programėlė būtų valdoma per grafinę vartotojo sąsają, kur vartotojas galės pasirinkti atitinkamas funkcijas kaip sportinės veiklos atpažinimas, išsikelti tikslus savo sportinei veiklai, stebėti ir redaguoti sportinės veiklos rezultatus. Programėlė turėtų turėti galimybę atvaizduoti sportinės veiklos rezultatus ekrane realiu laiku. Kuriama mobilioji programėlė turėtų būti interaktyvi ir įspėti vartotoją, kai jis pasieks savo nustatytą sportinės veiklos tikslą.

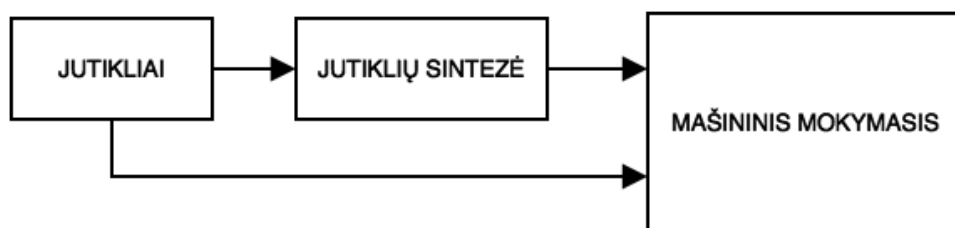
Šiuo metu jau egzistuoja mobiliųjų programėlių, kurios turi panašų funkcionalumą. Esminis aspektas kuris paverstų mūsų programėlę unikalia tai, kad išmanusis telefonas visos sportinės veiklos metu būtų pritvirtintas vienoje specifinėje vietoje ant žmogaus kūno, o išvardintoms sportinėms veikloms atpažinti nereikėtų skirtingų mobiliųjų programėlių. Šiuo metu konkurentai siūlo atskiras mobiliąsias programėles, kurios gali atpažinti tik vieną sportinę veiklą vienu metu. Taip pat, naudojant šias programėles išmaniojo telefono vieta ant žmogaus kūno keičiasi atitinkamai pagal sportinę veiklą.

Įgyvendinus projektą sukursime produktą, kuris bus skirtas plačiai vartotojų grupei, visame pasaulyje. Programinė įranga naudojama kartu su išmaniuoju telefonu atpažins sportines veiklas, skaičiuos jų pakartojimus ir atvaizduos sportinės veiklos progresą realiu laiku.

## 1. Analitinė dalis

### 1.1. Tikslas

Sportinės veiklos atpažinimo programėlės principas pavaizduotas paveiksle (1 pav.). Kur „Jutikliai“ – tai nešiojamojo įrenginio turimi jutikliai (akselerometras [17], giroskopas [18] ir magnetometras [29]). „Jutiklių sintezė“ – tai papildomi išskaičiuojami duomenys iš jutiklių duomenų (taikant tam tikrus algoritmus), skirti sumažinti jau turimų duomenų triukšmui. „Mašininis mokymasis“ – tai jutiklių ir jutiklių sintezės duomenų klasifikavimo algoritmai, su kuriais bus bandoma atpažinti ir registruoti sportines veiklas. Magistrinio darbo metu bus atliekamas tyrimas kuriame bus tiriami pagrindiniai ir populiariausi mašininio mokymosi algoritmai iš kurių bus atrinktas vienas geriausias ir panaudotas kuriamame programėlės prototipe. Taip pat, tyrimo metu taip pat bus sprendžiamos nenumatytos (darbo eigoje atsiradusios) problemos, renkami ir analizuojami sportinių veiklų duomenys. Projekto projektinė dalis vyks lygiagrečiai su tiriamąją dalimi.



1 pav. Sportinės veiklos atpažinimo principinė schema

Darbo tikslas yra sukurti mobiliąją programėlę veikiančią Android 5.0 operacinėje sistemoje [11], kuri leis registruoti vartotojo sportines veiklas kaip atsispaudimai, pritūpimai ir atsilenkimai. Programėlė leis nusistatyti skirtingų sportinių veiklų (pratimų, fizinio aktyvumo) tikslus bei automatiškai sekti progresą jų siekiant. Veiklos bus atpažįstamos ir registruojamos pasitelkiant išmaniojo telefono jutiklius, bei mašininio mokymosi algoritmus. Išmanusis telefonas bus įtvirtintas ant žmogaus kūno, vietoje, kurioje judėtų kartu su vartotojo kūnu ir fiksuotų specifinius judesius.

Apibendrinant galima išskirti kelis pagrindinius šio darbo tikslus:

1. Reikalingų testavimo duomenų surinkimas ir jų paruošimas naudojimui.
2. Surasti tinkamiausią mašininio mokymosi algoritmą ir jį pritaikyti sportinių veiklų atpažinimui bei pakartojimų skaičiavimui.
3. Sukurti programinės įrangos grafinę sąsają ir realizuoti prototipą pagal kuriamą metodiką.
4. Atlikti eksperimentą ir nustatyti sportinių veiklų atpažinimo bei skaičiavimo tikslumą.

## 1.2. Egzistuojantys panašūs moksliniai tyrimai

Žmogaus judesių atpažinimas tiriamas jau daugybe metų ir tai yra plati sfera mokslui, tyrimams ir duomenų gavybai. Mokslininkai ir analitikai pasitelkdami išmaniąsias technologijas bando kiek įmanoma tiksliau išgauti, atpažinti, išskirti tam tikrus žmogaus judesius iš daugybės surinktų duomenų. Žmogaus judesių atpažinimui yra naudojami įvairūs judesių jutikliai, pagrindiniai iš jų: akcelerometras, giroskopas, magnetometras, kamera, atstumo jutiklis, mikrofonas ir kiti. Vėliau jutiklių duomenys yra apdorojami, klasifikuojami, jiems pritaikomi įvairūs filtrai. Surinkus pakankamai žmogaus judesių duomenų yra vykdoma jų analizė naudojant įvairius analizavimo, spėjimo, grupavimo, filtravimo algoritmus.

Remiantis anksčiau atlikta apžvalga [1], žmogaus judesių atpažinimas gali būti išskirtas į dvi pagrindines grupes kaip žmogaus veiklų atpažinimas iš vaizdo medžiagos ir žmogaus veiklų atpažinimas naudojant fizinius jutiklius. Pagal minėtą apžvalgą žmogaus veiklų atpažinimas iš fizinių jutiklių gali būti skirstomas į dar dvi grupes: nešiojamus jutiklius ir žmogaus veiklų atpažinimas naudojant buityje esančius objektus. Kitoje apžvalgoje „Žmogaus veiklų atpažinimas naudojant išmaniusius telefonus“ [2] buvo išdėstyti pagrindiniai jutikliai esantys išmaniajame telefone, taip pat iškeltas pagrindinis žmogaus veiklų atpažinimo tikslas, aprašyti pagrindiniai metodai, duomenų išgavimo būdai ir pagrindiniai principai kaip taikyti duomenų gavyba (*angl. Data Mining*). „Judesių atpažinimo principai buvo taikomi ir detalai aprašyti šiuose moksliniuose darbuose: Judesių atpažinimas naudojant išmaniojo telefono jutiklius“ [19], „Žmogaus judesių atpažinimas naudojant išmaniuosius telefonus“ [20]. Mašininio mokymosi [21] duomenų analizavimo algoritmai kaip: sprendimų medis, Bajeso klasifikatorius, atraminių vektorių metodas, artimiausio kaimyno yra minimi literatūros šaltiniuose [19], [22]. Šiuose šaltiniuose detalai aprašyta išvardintų algoritmų veikimas, kada juos tinkamiausia naudoti, kokius parametrus taikyti, dalinai aprašytos apskaičiavimo formulės.

Yra daugybė tyrimų ir mokslinių darbų, kurie tyrinėją šią sritį, ir mūsų projektuojama sistema nėra išimtis. Vienas savo struktūra panašus darbas [5] buvo atliktas dviejų studentų iš Suomijos. Jų darbe buvo aprašytas atsispaudimų atpažinimas nuo kitų veiklų tyrimas. Jų darbe buvo aprašytas tyrimas atsispaudimų atpažinimui iš kitų sportinių veiklų naudojant atraminių vektorių algoritmą [35] (*angl. Support Vector Machine*) ir neuroninį tinklą [34]. Surinktų duomenų apdorojimui naudojo slenkančio lango metodą [16] su 1s, 2s, 4s, 8s langų ilgiais.

Kiti tyrinėjimai [3-4, 6, 8], sprendė žmogaus veiklų, kaip ėjimo, lėto ėjimo, bėgimo, lipimo laiptais, nulipimo laiptais ir stovėjimo atpažinimo problemas. Visuose tyrimuose buvo naudojama Weka [9]

duomenų analizavimo įrankis, taip pat analizuoti bendriniai mašininio mokymosi algoritmai. Duomenų surinkimui buvo naudojamas akselerometras ir giroskopas iš išmaniojo telefono.

Dar vienas panašus tyrimas [7] buvo atliktas, tačiau naudojant kitą mašininio mokymosi algoritmą. Tyrėjai naudojo dinaminės laiko deformacijos (*angl. Dynamic Time Warping*)[10] algoritmą, paprasčiausiai lygindami akselerometro signalo su tam tikrų veiklų šablonais. Papildomai buvo matuojamas širdies ritmas, kas leido pagerinti žmogaus veiklos atpažinimą.

### 1.3. Egzistuojančios programėlės sportinei veiklai fiksuoti

Lentelėje (1 lentelė) pavaizduota keletas egzistuojančių išmaniojo telefono programėlių pavyzdžių sportinei veiklai fiksuoti. Yra ir daug kitų Android programėlių su panašiomis funkcijomis, tačiau analizei buvo pasirinktos šios: „Runtastic Sit-Ups Abs Workout“ [23] – programėlė stiprinti pilvo presą; „Squats Workout“ [24] – programėlė pritūpimams; „Push Ups Workout“ [25] – programėlė atsispaudimams; „Pull Ups Workout“ [26] – programėlė prisitraukimams ir „S Health“ [27] – programėlė kasdieniai veiklai fiksuoti ir gerai fizinei būklei palaikyti. Šios keturios programėlės išsiskiria iš kitų, nes jose yra naudojami papildomi išmaniojo telefono jutikliai: akselerometras ir giroskopas.

1 lentelė. Egzistuojančių programėlių palyginimo lentelė

Pavadinimas	Veiklos tipas	Naudojami jutikliai	Veiklos atpažinimas	Kaina	Papildomos galimybės
<b>Runtastic Sit-Ups Abs Workout</b>	Atsilenkimai	Akselerometras	Ne	1,99 €	<ul style="list-style-type: none"> <li>• Skaičiuoja atsilenkimus naudojant akselerometrą;</li> <li>• Fiksuoja duomenis pagrindinėje svetainėje;</li> <li>• Automatiškai skaičiuoja laiką tarp pratimų.</li> </ul>
<b>Squats Workout</b>	Pritūpimai	Akselerometras, giroskopas ir atstumo jutikliai	Ne	Nemokama	<ul style="list-style-type: none"> <li>• Į gravitacijos pasikeitimą orientuotas pritūpimų skaičiavimas;</li> <li>• Rankinis duomenų pildymas;</li> <li>• Grafikai ir statistika;</li> <li>• Pranešimai.</li> </ul>
<b>Push Ups Workout</b>	Atsispaudimai	Atstumo jutiklis	Ne	0,99 €	<ul style="list-style-type: none"> <li>• Atsispaudimų skaičiavimas pagal atstumą;</li> <li>• Rankinis duomenų pildymas;</li> <li>• Grafikai ir statistika.</li> </ul>
<b>Pull Ups Workout</b>	Prisitraukimai	Akselerometras, giroskopas ir atstumo jutikliai	Ne	Nemokamas	<ul style="list-style-type: none"> <li>• Į gravitacijos pasikeitimą orientuotas prisitraukimų skaičiavimas;</li> <li>• Rankinis duomenų pildymas;</li> <li>• Grafikai ir statistika;</li> <li>• Pranešimai.</li> </ul>
<b>S Health</b>	Žingsniamatis	Akselerometras, giroskopas	Taip	Nemokamas	<ul style="list-style-type: none"> <li>• Skaičiuoja žingsnius;</li> <li>• Skaičiuoja kalorijas;</li> </ul>

Pavadinimas	Veiklos tipas	Naudojami jutikliai	Veiklos atpažinimas	Kaina	Papildomos galimybės
					<ul style="list-style-type: none"> <li>• Atvaizduoja duomenis grafiškai;</li> <li>• Mitybos programa.</li> </ul>
<b>Mūsų programėlė</b>	Beveik visos sportinės veiklos	Akselerometras, giroskopas, magnetometras	Taip	Nemokamas	<ul style="list-style-type: none"> <li>• Sportinės veiklos atpažinimas ir registravimas;</li> <li>• Automatinis;</li> <li>• Rankinis duomenų pildymas;</li> <li>• Duomenų atvaizdavimas grafiškai;</li> <li>• Galimybė programėlę apmokinti nauja sportine veikla.</li> </ul>

### 1.3.1. Runtastic Sit-Ups Abs Workout

Programėlės pagrindinės funkcijos yra atvaizduoti atsilenkimų rezultatus ir fiksuoti jų kiekį. Taip pat yra sudaroma kasdienė individualaus sporto programą atsilenkimams. Programėlė ragina (balsu) vartotoją sportuoti atliekant nustatytą atsilenkimų skaičių per dieną.

Programėlė veikia Android operacinėje sistemoje. Programėlė naudoja išmaniojo telefono jutiklius: akselerometrą. Šio jutiklio pagalbą yra fiksuojami įrenginio judėjimo pagreičiai erdvėje, pagal tai išskaičiuojamas atsilenkimų skaičius.

### 1.3.2. Squats Workout

Programėlė yra skirta pritūpimams fiksuoti ir raginti (balsu) vartotoją juos daryti. Programėlė veikia išmaniojo telefono Android operacinėje sistemoje. Pritūpimų fiksavimui yra naudojami atstumo, akselerometro ir giroskopo jutikliai. Programėlėje yra realizuotos papildomos funkcijos:

- Asmeninės sporto programos sudarymas pritūpimams;
- Duomenų redagavimas ir įvedimas rankiniu būdu;
- Rezultatų atvaizdavimas grafiškai ir statistiškai.

### 1.3.3. Push Ups Workout

Programėlės pagrindinės funkcijos yra atsispaudimų fiksavimas ir rezultatų atvaizdavimas grafiškai. Atsispaudimų fiksavimui naudojamas atstumo jutiklis (*angl. proximity sensor*), užfiksuoti rezultatai yra išvedami į ekraną, o kad vartotojui būtų smagiau, programėlė balsu skatina daryti atsispaudimus. Programėlė veikia išmaniajame telefone su Android operacinėje sistemoje. Papildomos funkcijos:

- Asmeninės sporto programos sudarymas atsispaudimams;
- Duomenų redagavimas ir įvedimas rankiniu būdu;

- Atsispaudimų rezultatų statistikos atvaizdavimas grafiškai.

#### **1.3.4. Pull Ups Workout**

Android programėlė skirta fiksuoti prisitraukimams ir raginti (balsu) vartotoją juos daryti. Programėlė veikia išmaniajame telefone su Android operacine sistema. Programėlė prisitraukimus fiksuoja naudodama atstumo, akselerometro ir giroskopo jutiklius, gauti rezultatai yra išvedami į įrenginio ekraną. Programėlėje yra realizuotos papildomos funkcijos:

- Duomenų redagavimas ir įvedimas rankiniu būdu;
- Duomenų statistikos pateikimas grafiškai;
- Asmeninės sporto programos sudarymas prisitraukimams;
- Duomenų atsarginių kopijų darymas ir atkūrimas;
- Treniruočių pranešimai.

#### **1.3.5. S Health**

Samsung S Health programėlė yra unikali savo lankstumu ir įvairių funkcijų gausa. Ši programėlė veikia išmaniuosiuose telefonuose su Android operacine sistema. S Health naudoja jutiklius: akselerometrą, giroskopą, kamerą ir atstumo jutiklius. Naudojant išvardintus jutiklius programėlėje yra realizuotos funkcijos:

- Fizinės būsenos nustatymas;
- Planuoti sportinės veiklos užsiėmimus;
- Nustatyti tikslus sportinėms veiklos;
- Miego stebėjimas (naudojamas giroskopas);
- Sekti dietą pagal suvartotas kalorijas;
- Integruotis su kitomis sistemomis.

Programėlė yra nemokama, dažniausiai būna įrašyta į išmanųjį telefoną su Android operacine sistema.

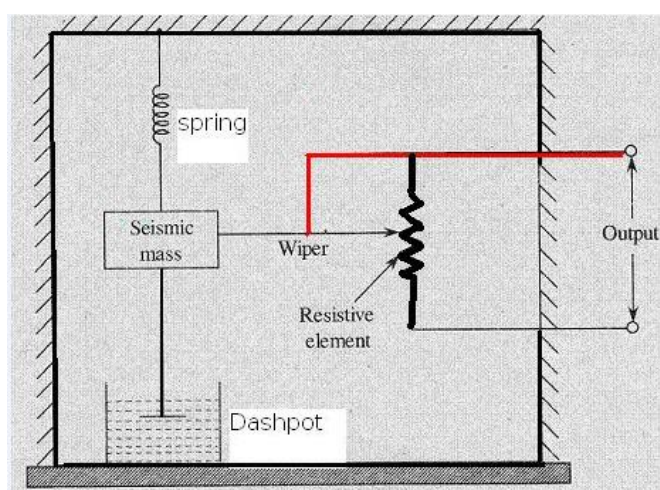


## 1.4. Dažniausiai naudojami sprendimai ir technologijos

### 1.4.1. Jutikliai

#### 1.4.1.1. Akselerometras

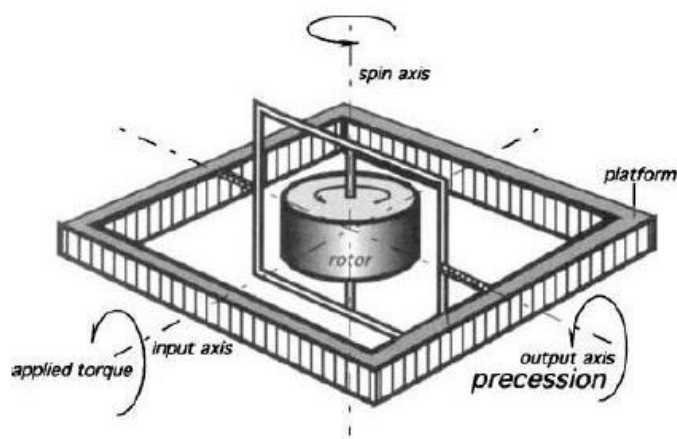
Akselerometras (*angl. Accelerometer*) – prietaisas, kuris matuoja pagreitį. Dauguma akselerometrų nerodo pagreičio duomenų, o juos perduoda kitiems prietaisams [17]. Vienaašiai arba daugiaašiai modeliai suranda vektoriaus normą, jo kryptį ir pagreitį kaip vektoriaus dydį. Gali aptikti orientaciją, greitėjimą, vibraciją, smūgį, ir kritimą. Mikroelektromechaniniai akselerometrai vis dažniau įmontuojami nešiojamuose elektroniniuose aparatuose, mobiliuosiuose telefonuose ir vaizdo žaidimų valdymų pulteliuose. Paveiksle (2 pav.) pavaizduota akselerometro architektūros schema.



2 pav. Vienaašio akselerometro architektūros schema [43]

#### 1.4.1.2. Groskopas

Giroskopas – prietaisas, skirtas sukimuisi registruoti. Bendru atveju tai yra bet koks greitai besisukantis kietas kūnas, kurio ašis gali erdvėje keisti savo kryptį [18]. Visi reiškiniai susiję su greitu giroskopo sukimusi vadinami girokopiniais. Giroskopas yra skridinys, kurio ašies kryptis laisvai kinta. Žaislinis giroskopas yra paprasto giroskopo modelis. Įsuktas giroskopas priešinasi tokioms jėgoms kaip sunkis, kurie veikia jo ašį, ir stengiasi išlaikyti pusiausvyrą. Paveiksle (3 pav.) pavaizduota giroskopo architektūros schema.



3 pav. Giroskopo architektūros schema [18]

### 1.4.1.3. Magnetometras

Magnetometras – prietaisas, skirtas matuoti magnetinei indukcijai [28]. Vietoje kontūro, kuriuo teka elektros srovė, įtaisytas nedidelis magnetas. Apie magnetinę indukciją sprendžiama iš magnetą veikiančių magnetinio lauko jėgų momento.

## 1.4.2. Jutiklių sintezė

### 1.4.2.1. Pagrindai

Dauguma šiandienos sudėtingų sistemų reikalauja duomenų atvaizdavimo iš triukšmingo mūsų pasaulio, kur užfiksuoti duomenys yra netikslūs ir dažnai nepilni. Tam tikslui yra smarkiai vystoma jutiklių sintezė, kuri iš aplinkos gautus netikslus ir triukšmingus duomenis paverčia prasmingais ir tiksliais [29]. Pasitelkiant dirbtinį intelektą, modelių atpažinimą, signalų apdorojimą, valdymo teoriją ir statistinį vertinimą iš kelių mikroelektromechaninių sistemų užfiksuoti duomenys yra susintezuojami siekiant padidinti jų tikslumą ir išgauti naujus, jutikliais neužfiksuojamus duomenis [30-31].

Jutiklių technologijų tobulėjimas ir skaičiavimų technologijos sujungtos patobulinta techninė įranga leidžia atlikti jutiklių sintezę realiu laiku. Šie pranašumai naudojant techninę ir programinę įrangą leidžia pamėgdžioti natūralius jutiklių sintezės pritaikymus egzistuojančius sujungiant jutiklių (akselerometro, giroskopo, slėgio matuoklio, magnetometro ir mikrofono) duomenis į daugybę visokių kombinacijų, juos apdorojant ir ištransliuojant bevieliu ryšiu. Šios jutiklių sintezės savybės sukėlė didelį programinės įrangos augimą. Šiuo metu rinkoje vyrauja 9 ašių judesio jutikliai.

Programinės įrangos augimo priežastis yra ta, kad jutikliai kainos nebėra tokios aukštos, o kokybė ir naudojamos technologijos patobulėjo. Jutiklių kalibravimas yra atliekas neišvykstant iš gamyklos. O taip pat didelį pranašumą teikia jutiklių dydžio sumažinimas. Didžiausias privalumas jutiklių sintezės

taikyme yra neišmatuojamos galimybės pačiuose jutikliuose. Vis tobulėjantys jutiklių sintezės algoritmai leidžia pademonstruoti dar geresnį jutiklių tikslumą, o taip pat platesnį jų panaudojimą.

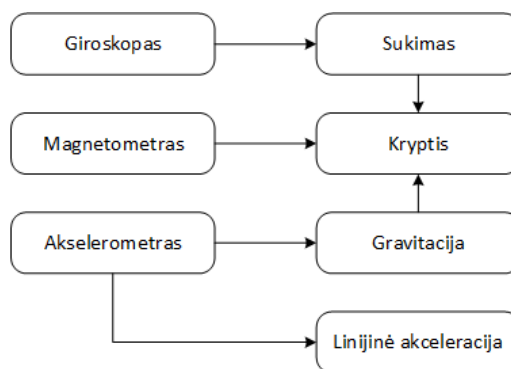
#### 1.4.2.2. Nuo kariuomenės iki naudotojų

Vienu laiku, visas jutiklių panaudojimas buvo sutelktas karinės pramonės sferoje, kur jutiklių sintezė buvo paremta keliais jutikliais siekiant pagerinti naujų ginklų taiklumą automatiškai sekant taikinius, juos atpažįstant, ginklo valdymui ir be pilotės transporto priemonės valdymui. Ypač didelė nauda karinei pramonei buvo tai: pagerėjęs ginklų taiklumas, išvystyta galimybė atpažinti, identifikuoti ir nustatyti buvimo vietą tam tikrų objektų. Jutiklių sintezės didelis pasisekimas karinėje pramonėje buvo perimtas civilių ir naudojamas automatizavimo, sudėtingų įrenginių valdymo, robotų, elektronikos pramonėje ir daugumoje kitų sričių.

Priklausomai nuo taikymo srities, jutiklių sintezės duomenys gali būti naudojami daugybe skirtingų būdų, dažniausiai pagerinant esamus jutiklių matavimus:

- Naujų galimybių sukūrimas, kaip apjungiant kelių jutiklių matavimus ir gaunant visiškai naudą duomenų masyvą;
- Sprendimų priėmimas kaip nuskaičius individualių jutiklių matavimus, su jais atliekant jutiklių sintezę ir gaunant reikšmę pagal kurią galim priimti konkretų sprendimą.

Tam tikra eilės tvarka sudėjus akcelerometro, giroskopo ir magnetometro jutiklių (x, y, z ašis) duomenis galime gauti sukimo, orientacijos, gravitacijos ir linijinio pagreičio reikšmes. Detalesnė jutiklių sintezės diagrama pavaizduota (4 pav.).



4 pav. Jutiklių sintezės diagrama

Kaip pavaizduota (4 pav.) naudojami jutiklių duomenys yra:

- Akselerometro jutiklio duomenys: x, y, z ašys;
- Giroskopo jutiklio duomenys: x, y, z ašys;
- Magnetometro jutiklio duomenys: x, y, z ašys.

Po jutiklių sintezės gauname naujus duomenis:

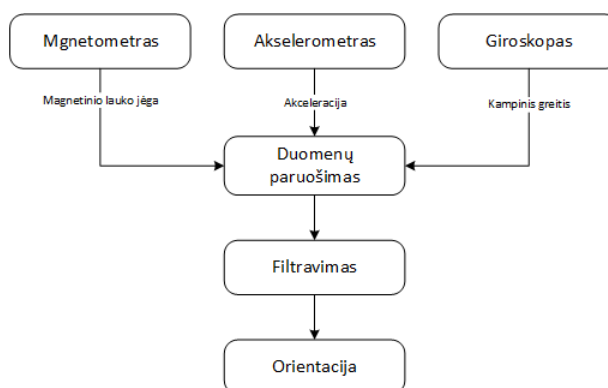
- Ervinė orientacija sudaryta iš 4 skaičių;
- Sukimasi: (*angl. pitch roll yaw*) ir kryptį;
- Linijinį pagreitį;
- Gravitaciją

### 1.4.2.3. Kalman filtras

Standartinis Kalman filtras yra tiesinė prognozė, kuri turi galimybę sumažinti klaidų nuokrypius [32]. Tiesinės prognozės lygčiai būdingi grįžtamojo tipo skaičiavimai, kurie gali būti gana paprasti ir atliekami mikroprocesoriuje. Kalman filtro apskaičiuoti išėjimo duomenys priklauso nuo esamų įėjimo duomenų ir jų būsenos. Filtras remiasi pagrindine formule tiesinei prognozei apskaičiuoti:

$$X_{k+1} = Ax_k + Bu_k + w_k$$
$$y_k = Hx_k + z_k$$

Formulės tiesinei prognozei apskaičiuoti, kur: A, B ir H yra būsena, įvesties duomenys ir matavimų matricos atitikimai, x sistemos būsena, u sistemos įvesties duomenys, y išmatuoti išėjimo duomenys, w vyksmo triukšmas ir z matavimų triukšmas. Kaip be būtų negalima pilnai remtis apskaičiuotais y išvesties duomenimis, nes y duomenys yra su triukšmu. Atsižvelgiant į sistemą, visi kintamieji gali būti vektorinės išraiškos.



5 pav. Orientacijos išskaičiavimas pagal Kalman filtrą

Kaip matome (5 pav.) paveiksle yra pateiktas Kalman filtras apskaičiuojant orientacijos vektorių. Pavaizduoto proceso metu buvo gautos horizontalaus ir vertikalaus judėjimo reikšmės, taip pat sukimasis apie ašį (*angl. pitch, roll, yaw*).

Kalman filtras daugiausiai naudojamas išskaičiuoti orientacijai, vietai erdvėje, sukimasi, papildomai naudojant GPS galima apskaičiuoti vietą pagal pasaulines koordinatas. Šis filtras naudingas tuo, kad

padeda sumažinti triukšmą gauta kartu su jutiklių duomenimis, o taip patikslinti jau turimus jutiklių duomenis. Taip pat mašininis mokymasis gerai žinomas internete, kaip nepageidaujamų žinučių

### 1.4.3. Mašininio mokymosi algoritmai

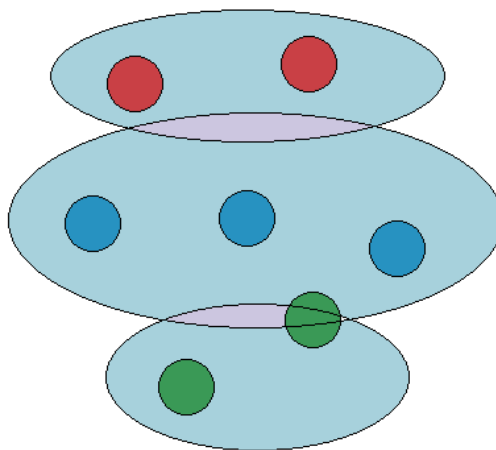
Mašininio mokymosi algoritmų pagrindiniai privalumai yra pagerinti kompiuterinių programų veikimą, kurios naudoja statistinį analizavimą. Taip pat, yra daug kitų sričių, užduočių, kurias gali išspęsti mašininis mokymasis. Tai kalbos atpažinimas, žaidimų scenarijai, medicininių problemų sprendimas ir duomenų gavyba.

Mašininiam mokymuisi yra išrasta nemažai algoritmų. Šie algoritmai plačiai naudojami dirbtiniame intelekto, tikimybėse ir statistikoje, neurobiologijoje ir kitur. Pagrindiniai klasifikavimo metodai (*angl. Base-level clasifiers*):

- Naive Bayes;
- Neural Networks;
- Support Vector Machine;
- Decision tree.

#### 1.4.3.1. Bajeso klasifikatoriai (*angl. Naive Bayes*)

Bajeso klasifikatorius veikia pritaikant Bajeso teoremą, darant prielaidą, jog visi požymiai yra nepriklausomi [33]. Elementų priklausymas klasei nusakomas įvertinant tikimybes. Vizualiai šis procesas - plokštumos dalijimas elipsėmis, kaip pavaizduota paveiksle (6 pav.). Esant aibių susikirtimui, pasirenkama aibe su didesne tikimybe.



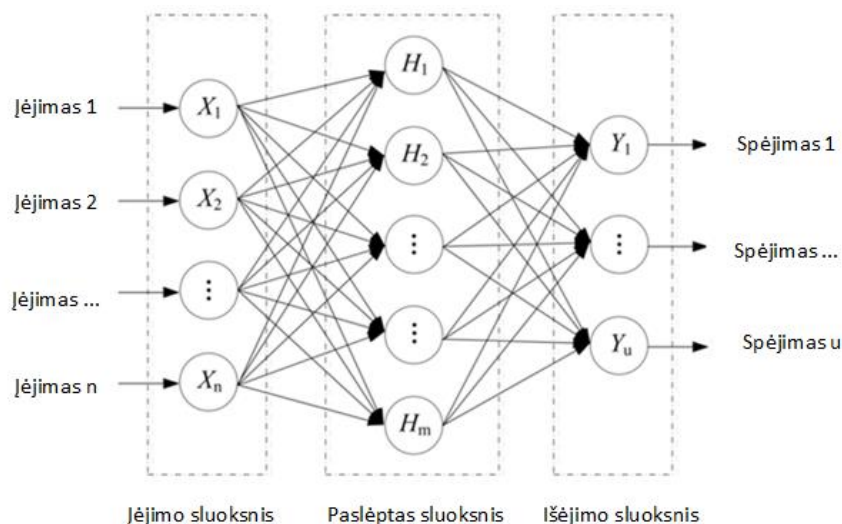
6 pav. Bajeso klasių atskyrimas

Tikimybė apskaičiuojama pagal formulę pateiktą žemiau. Kur  $T$  atstoja hipotezės teoriją, kurią mes tikriname, o  $E$  atvaizduoja nauja įvykį, kurį turime patvirtinti arba atmesti. Kiekvienam spėjimui naudosime  $S$ , kuri atstos pasikliautinumą (subjekto tikimybę), kad  $S$  teisinga.

$$P(T|E) = \frac{P(E|T) \times P(T)}{P(E|T) \times P(T) + P(E|\neg T) \times P(\neg T)}$$

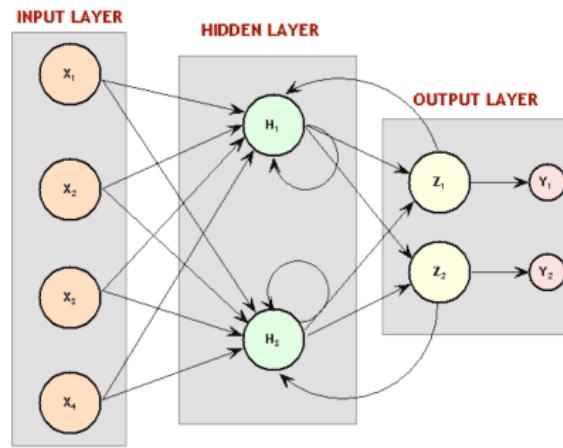
#### 1.4.3.2. Neuroninių tinklų (angl. *Neural Networks*)

Neuroninio tinklo algoritmas yra skaičiavimo metodas, kuris grindžiamas dideliu skaičiumi nervinių mazgų panašiai atitinkančių biologinių smegenų struktūrą ir problemų sprendimo metodiką [34]. Kiekvienas neuroninio tinklo mazgas yra sujungtas su kitais mazgais tam tikromis jungtimis su svoriais (panašiai kaip svorinis grafas). Didinant arba mažinant jungčių svorius yra keičiami neuroninio tinklo gražinami rezultatai. Neuroninis tinklas gali būti sudarytas iš vieno ir daugiau vidinių sluoksnių, pagal kuriuos sprendžiamas neuroninio tinklo sudėtingumas. Neuroninio tinklo vizualus pavyzdys pateiktas paveiksle (7 pav.). Kur  $X$  yra įėjimo signalas,  $H$  nematomas sluoksnis (gali būti nuo 1 iki  $n$ ) ir  $Y$  yra išėjimo sluoksnis.  $X$  ir  $Y$  skaičius sluoksniuose gali skirtis, kadangi į  $Y$  sluoksnį grįžta neuroniniame tinkle apdorotas rezultatas, kuris nepriklauso nuo  $X$  skaičiaus.



7 pav. Neuroninio tinklo medelis

Egzistuoja ir kitas neuroninio tinklo medelis, kuris turi grįžtamąjį ryšį. Tai pasikartojantis neuroninis tinklas RNN (angl. *Recurrent Neural Network*). Šis modelis pasižymi savo galimybe įsiminti prieš tai buvusį neuroninio tinklo rezultatą. Gebėjimas atsižvelgti į prieš tai buvusią tinklo būseną šiam modeliui duoda didelį pranašumą atpažįstant tam tikrų signalų duomenis. Pasikartojančio neuroninio tinklo medelis pavaizduotas paveiksle (8 pav.).



8 pav. Pasikartojantis neuroninis tinklas [42]

#### 1.4.3.3. Atraminių vektorių metodas (*angl. Support Vector Machine*)

Atraminių vektorių klasifikatoriai panaudodami mokymo duomenis (*angl. train*), sukuria klasifikavimo taisykles. Klasifikavimo taisyklė pasirašo kaip dviejų aibių atskyrimas viena plokštuma. Atskyrimas vykdomas paviršiu dalijant i dvi dalis naudojant hiperplokštumą. Hiperplokštuma iš abiejų pusių turi paraštes nutolusias tam tikru atstumu (vektoriaus riba). SVM (*angl. Support Vector Machine*) [35] metodas taikomas tada, kai nepavyksta atskirti dviejų aibių viena nuo kitos. SVM metodas suranda tinkamiausia plokštumą aibėms atskirti. SVM metodas skaidomas į minkštos ir kietos paraštės metodus.

#### 1.4.3.4. Sprendimų medis (*angl. Decision Tree*)

Sprendimų medis yra vienas iš labiausiai naudojamų klasifikavimo algoritmų. Labai veiksmingas dirbant su triukšmingais duomenimis, kuriuos geba atvaizduoti pagal domenų grupes [36]. Taip pat, šis algoritmas visiems yra gerai žinomas ir aiškiai suprantamas jo veikimas.

Sprendimų medis remiasi medžio principu ir gali turėti daugiau nei vieną duomenų išėjimą. Pagrindinė sprendimų medžio užduotis yra suklasifikuoti duomenis pagal duomenų grupes. Pagrindinis algoritmas:

- Medis konstruojamas nuo viršaus į apačią pasikartojančiai dalinant duomenis į dvi grupes;
- Pačioje pradžioje visi apmokyti pavyzdžiai yra šaknys;
- Atributai yra suskirstyti į kategorijas;
- Duomenų pavyzdžiai yra sugrupuojami į grupes remiantis pasirinktais atributais;
- Testo atributai yra atvaizduoti medžio apačioje;
- Medis formuojamas, kol grupėje nebelieka atributų.

## **1.5. Programinės priemonės**

### **1.5.1. Android**

Android tai atviro kodo operacinė sistema, labiausiai naudojama išmaniuosiuose telefonuose, taip pat ją galima įdiegti ir į kitus išmaniuosius įrenginius, tokius kaip planšetiniai kompiuteriai, išmanieji laikrodžiai ir kiti. Šiuo metu Android operacinė sistema yra nusipirkęs Google, kurie ją ir toliau tobulina.

#### **1.5.1.1. Sistemos aprašymas**

Android tai ant Linux branduolio paremta operacinė sistema, sukurta daugiausia Google, tačiau turi savyje ir daug kitų įvairiausių plėtinių. Pagrindinė šios sistemos programavimo kalba yra Java. Kuriamos programos sąsaja dalinai aprašoma XML programavimo kalba. Android turi papildomas priemones pasiekti jutiklius (akselerometrą, giroskopą, magnetometrą), GPS ir kitas sistemas kurios dažnos išmaniuosiuose telefonuose. Android gerai pritaikyta veikti esant ribotam ekrano dydžiui, neturint pelės ir klaviatūros, nors šiuos įtaisus (jei jie yra) sistema palaiko.

#### **1.5.1.2. Programavimas**

Android programas gali kurti kiekvienas programuotojas, pagrindiniai ir gana efektyvūs įrankiai (kaip Android Studio) yra nemokami. Kūrimui net nebūtina investuoti į Android palaikantį mobilųjį telefoną ar kitą prietaisą – tam yra virtualus Android operacinės sistemos imitatorius. Jei išmanusis telefonas ar kitas prietaisas yra prijungiamas prie programuotojo kompiuterio per mikro-USB jungtį – jis pavirsta programavimo įrankiu. Taip nesunku patikrinti programos darbą su daugumos išmaniojo telefono davikliais bei matyti ar užduočiai pakanka išmaniojo telefono atminties, procesoriaus resursų. Vienas iš didesnių virtualaus Android imitatoriaus privalumų, tai kad galima patikrinti kaip programa veikia esant įvairiai ekrano raiškai ir įvairioms Android versijoms.

### **1.5.2. Android „SensorManager“ biblioteka**

Lentelėje (2 lentelė) pateikti Android operacinės sistemos metodai jutikių užfiksuotiems duomenims gauti. Biblioteka jau yra realizuota Android operacinėje sistemoje, jokių papildomų plėtinių diegti nereikia, visi lentelėje pateikti metodai yra laisvai pasiekiami per programinį kodą [37].

Jutiklių duomenis galima nuskaityti pagal keturis nuskaitymo dažnius:

- `SENSOR_DELAY_FASTEST` – 0 ms, maksimalus nuskaitymo dažnis;
- `SENSOR_DELAY_GAME` – 20 ms, nuskaitymo dažnis 50Hz;
- `SENSOR_DELAY_UI` – 67 ms, nuskaitymo dažnis apytiksliai 15Hz;



- SENSOR\_DELAY\_NORMAL – 200 ms, nuskaitymo dažnis 5Hz.

Taip pat „SensorManager“ turi jau realizuotus „SensorFusion“ metodus:

- TYPE\_ORIENTATION – Orientacijos gavimo metodas;
- STANDART\_GRAVITY – Žemės gravitacijos (g) gavimo metodas;
- TYPE\_LINEAR\_ACCELERATION – Linijinio pagreičio gavimo metodas.

Visi išvardinti metodai ir funkcijos yra pasiekiami nuo Android 4.0 versijos operacinės sistemos.

2 lentelė. Android „SensorManager“ bibliotekos jutiklių nuskaitymo metodų aprašymas [44]

Jutiklio pavadinimas	Metodas	Aprašymas	Matavimo vnt.
TYPE_ACCELEROMETER	SensorEvent.values[0]	Akselerometro jėga X ašimi	m/s <sup>2</sup>
	SensorEvent.values[1]	Akselerometro jėga Y ašimi	
	SensorEvent.values[2]	Akselerometro jėga Z ašimi	
TYPE_GRAVITY	SensorEvent.values[0]	Gravitacijos jėga X ašimi	m/s <sup>2</sup>
	SensorEvent.values[1]	Gravitacijos jėga Y ašimi	
	SensorEvent.values[2]	Gravitacijos jėga Z ašimi	
TYPE_GYROSCOPE	SensorEvent.values[0]	Sukimosi kiekis X ašimi	rad/s
	SensorEvent.values[1]	Sukimosi kiekis Y ašimi	
	SensorEvent.values[2]	Sukimosi kiekis Z ašimi	
TYPE_GYROSCOPE_UNCALIBRATED	SensorEvent.values[0]	Sukimasis apie X (su nuokrypiu) ašį	rad/s
	SensorEvent.values[1]	Sukimasis apie Y (su nuokrypiu) ašį	
	SensorEvent.values[2]	Sukimasis apie Z (su nuokrypiu) ašį	
	SensorEvent.values[3]	Numatomas nuokrypis X ašimi	
	SensorEvent.values[4]	Numatomas nuokrypis Y ašimi	
	SensorEvent.values[5]	Numatomas nuokrypis Z ašimi	
TYPE_LINEAR_ACCELERATION	SensorEvent.values[0]	Pagretis X ašimi (bet gravitacijos)	m/s <sup>2</sup>
	SensorEvent.values[1]	Pagretis Y ašimi (bet gravitacijos)	
	SensorEvent.values[2]	Pagretis Z ašimi (bet gravitacijos)	
TYPE_ROTATION_VECTOR	SensorEvent.values[0]	Sukimosi vektoriaus componentė X ašimi ( $X * \sin(\theta/2)$ )	Nenustatytas
	SensorEvent.values[1]	Sukimosi vektoriaus componentė Y ašimi ( $Y * \sin(\theta/2)$ )	
	SensorEvent.values[2]	Sukimosi vektoriaus componentė Z ašimi ( $Z * \sin(\theta/2)$ )	
	SensorEvent.values[3]		
TYPE_SIGNIFICANT_MOTION	N/A	N/A	N/A
TYPE_STEP_COUNTER	SensorEvent.values[0]	Žingsnių skaičius nuo paskutinio perkrovimo kai jutiklis buvo aktyvus	Žingsniai
TYPE_STEP_DETECTOR	N/A	N/A	N/A

### **1.5.3. Bibliotekos realizuojančios mašininio mokymosi algoritmą**

#### **1.5.3.1. Weka**

Atviro kodo biblioteka, parašyta Java programavimo kalba. Biblioteką lengva įkomponuoti į SDK failą [38].

Pagrindiniai komponentai:

- Instances – Atskirieji atvejai duomenims;
- Filter – Filtravimas duomenų apdorojimui;
- Classifier/Clusterer – Klasifikatoriai, klasterizavimas duomenų suskirstymui;
- Evaluating – Duomenų įvertinimas;
- Attribute selection – Nereikšmingų atributų pašalinimas iš duomenų.

#### **1.5.3.2. TensorFlow**

2015 metais lapkričio mėnesį Google paskelbė apie atviro kodo mašininio mokymosi biblioteką TensorFlow [39]. Ši mašininio mokymosi biblioteka yra pritaikyta veikti Android operacinėje sistemoje. Pagrindinis TensorFlow uždavinys yra atpažinti vaizdus ir tam tikrus signalus, juos analizuoti ir atlikti jų atpažinimą.

#### **1.5.3.3. Scikit-learn**

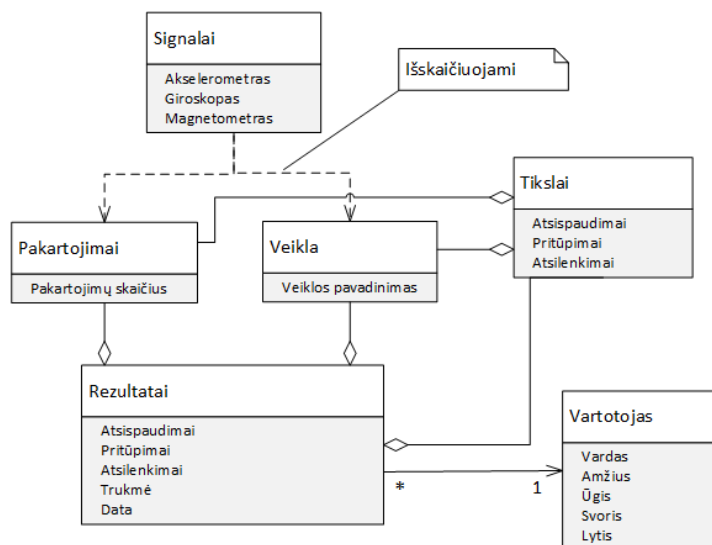
Atviro kodo mašininio mokymosi biblioteka, skirta Python programavimo kalbai [40]. Pagrindiniai uždaviniai:

- Clasiffication – Duomenų klasifikavimas;
- Regression – Nenutrūkstamas duomenų prognozavimas;
- Clustering – Automatinis duomenų grupavimas į klasterius;
- Dimensionality reduction – Sumažintas atsitiktinių duomenų parinkimas;
- Model selection – Lygina, patvirtina ir pasirenka parametrus bei modelius;
- Preprocessing – Požymių išskyrimas ir normalizavimas.

## 2. Projektinė dalis

### 2.1. Funkciniai reikalavimai

#### 2.1.1. Duomenų modelis



9 pav. Sistemos duomenų modelis

3 lentelė. Duomenys naudojami sistemos duomenų modelyje

Pavadinimas	Turinys	Tipas
Signalai	Išmaniojo telefono jutiklių duomenų fiksavimas	Klasė
Pakartojimai	Iš jutiklių duomenų išskaičiuojamas sportinės veiklos pakartojimų skaičius	Klasė
Veikla	Iš jutiklių duomenų išskaičiuojamas sportinė veikla	Klasė
Tiksiai	Sportinės veiklos tikslų išsikėlimas, saugojimas	Klasė
Rezultatai	Sportinės veiklos rezultatų skaičiavimas, saugojimas	Klasė
Vartotojas	Vartotojas, vartotojo duomenys	Klasė
Akselerometras	* Akselerometro signalo duomenys*	Atributas/Elementas
Giroskopas	* Giroskopo signalo duomenys*	Atributas/Elementas
Magnetometras	* Magnetometro signalo duomenys*	Atributas/Elementas
Pakartojimų skaičius	* Išskaičiuotas sportinės veiklos pakartojimų skaičius*	Atributas/Elementas
Veiklos pavadinimas	* Išskaičiuotas sportinės veiklos pavadinimas*	Atributas/Elementas
Atsispaudimai	* Sportinės veiklos – atsispaudimų pakartojimų skaičius*	Atributas/Elementas
Pritūpimai	* Sportinės veiklos – pritūpimų pakartojimų skaičius*	Atributas/Elementas
Atsilenkimai	* Sportinės veiklos – atsilenkimų pakartojimų skaičius*	Atributas/Elementas
Trukmė	* Sportinės veiklos atlikimo trukmė*	Atributas/Elementas
Data	* Sportinės veiklos atlikimo data, yyyy-MM-dd HH:mm*	Atributas/Elementas
Vardas	* Vartotojo vardas*	Atributas/Elementas
Amžius	* Vartotojo amžius*	Atributas/Elementas
Ūgis	* Vartotojo ūgis*	Atributas/Elementas
Svoris	* Vartotojo svoris*	Atributas/Elementas
Lytis	* Vartotojo lytis*	Atributas/Elementas

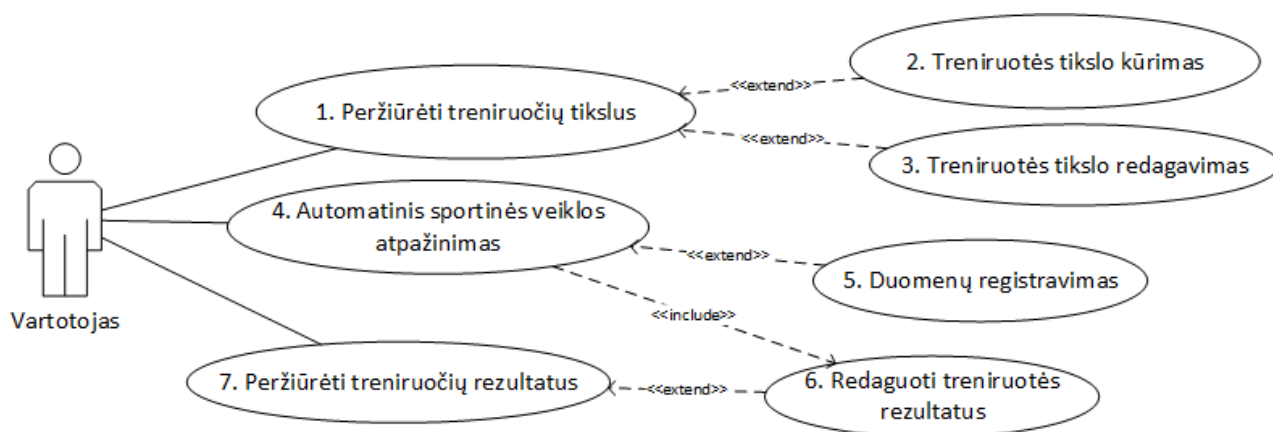
Sistemos vartotojas kaip klasė nėra būtinas šios programinės įrangos prototipui. Sistemos prototipas yra projektuojamas vienam vartotojui, o šiuo metu numatomas programinės įrangos funkcionalumas nereikalauja vartotojų duomenų modelio.

Išskaičiuojami „Pakartojimai“ ir „Veikla“ yra tarpinėje duomenų apdorojimo formoje (duomenys išskaičiuoti atliekant jutiklių duomenų analizę su mašininio mokymosi algoritmais), todėl šie duomenų modeliai yra orientaciniai ir skirti palengvinti supratimą apie sistemos duomenų srautus.

## 2.1.2. Sistemos sudėtis

### 2.1.2.1. Sistemos ribos

Panaudojimo atvejų diagrama sudaroma įvertinant kiekvieną išskirtą veiklos įvykį naujoje sistemoje ir priskirtus objektus sistemoje:



10 pav. Panaudos atvejų diagrama

### 2.1.2.2. Panaudojimo atvejai

PA 1 Peržiūrėti treniruočių tikslus	
<b>Tikslas/uždavinys.</b> Peržiūrėti treniruočių tikslus.	
<b>Aprašymas.</b> Vartotojas gali peržiūrėti treniruočių tikslus iš esamų tikslų sąrašo.	
<b>Prieš-sąlyga</b>	Vartotojas ketina peržiūrėti treniruočių tikslus.
<b>Aktorius</b>	Vartotojas.
<b>Sužadinimo sąlyga</b>	Atsirado poreikis peržiūrėti treniruočių tikslus.
<b>Veiklos taisyklės</b>	Privaloma pasirinkti meniu punktą „Tikslai“, norint peržiūrėti treniruočių tikslus.
Pagrindinis scenarijus	
Pagrindinis įvykių srautas	Sistemos reakcija
1. Vartotojas paspaudžia meniu punktą „Tikslai“.	1.1. Sistema užfiksuoja vartotojo pasirinkimą. 1.2. Sistema atidaro naują langą.
<b>Po-sąlyga</b>	Atveriamas treniruočių tikslų sąrašas.
<b>Alternatyvūs scenarijai</b>	Nėra

PA 2 Treniruotės tikslo kūrimas	
<b>Tikslas/uždavinys.</b> Surinkti sportinės veiklos treniruotės tikslą.	
<b>Aprašymas.</b> Sportinės veiklos tikslo sukūrimas.	
<b>Prieš-sąlyga</b>	Vartotojas ketina sukurti treniruotės tikslą.

<b>Aktorius</b>	Vartotojas.
<b>Sužadavimo sąlyga</b>	Noras sukurti treniruotės tikslą.
<b>Veiklos taisyklės</b>	Sportinės veiklos sukūrimas galimas tik iš sportinės veiklos tikslų sąrašo.
<b>Pagrindinis scenarijus</b>	
<b>Pagrindinis įvykių srautas</b>	<b>Sistemos reakcija</b>
1. Paspaudžiamas naujo treniruotės tikslo kūrimo mygtukas.	1.1. Sistema atidaro treniruotės tikslo kūrimo langą.
2. Įvedami treniruotės tikslo duomenys.	2.1. Sistema atlieka grafines vizualizacijas.
3. Išsaugomas treniruotės tikslas.	3.1. Sistema išsaugo treniruotės tikslo duomenis į duomenų bazę.
<b>Po-sąlyga</b>	Treniruotės tikslų sąrašė atsiranda naujai sukurtas įrašas.
<b>Alternatyvūs scenarijai</b>	Nėra

<b>PA 3 Treniruotės tikslo redagavimas</b>	
<b>Tikslas/uždavinys.</b> Redaguoti sportinės veiklos treniruotės tikslą.	
<b>Aprašymas.</b> Sportinės veiklos tikslo redagavimas/ištrynimasis.	
<b>Prieš-sąlyga</b>	Vartotojas ketina redaguoti treniruotės tikslą.
<b>Aktorius</b>	Vartotojas.
<b>Sužadavimo sąlyga</b>	Noras redaguoti treniruotės tikslą.
<b>Veiklos taisyklės</b>	Sportinės veiklos redagavimas galimas tik iš sportinės veiklos tikslų sąrašo.
<b>Pagrindinis scenarijus</b>	
<b>Pagrindinis įvykių srautas</b>	<b>Sistemos reakcija</b>
1. Pasirenkamas treniruotės tikslo įrašas iš treniruotės tikslų sąrašo.	1.1. Sistema atidaro treniruotės tikslo redagavimo langą.
2. Redaguojami treniruotės tikslo duomenys.	2.1. Sistema atlieka grafines vizualizacijas.
3. Išsaugomas treniruotės tikslas.	3.1. Sistema išsaugo treniruotės tikslo duomenis į duomenų bazę.
<b>Po-sąlyga</b>	Treniruotės tikslų sąrašė matomas atnaujintas įrašas.
<b>Alternatyvūs scenarijai</b>	
1. Pasirenkamas treniruotės tikslo įrašas iš treniruotės tikslų sąrašo.	1.1. Sistema atidaro treniruotės tikslo redagavimo langą.
2. Paspaudžiamas treniruotės tikslo trynimo mygtukas.	2.1. Sistema išsaugo treniruotės tikslo duomenis į duomenų bazę.
<b>Po-sąlyga</b>	Treniruotės tikslų sąrašė įrašo nebelieka.

<b>PA 4 Automatinis sportinės veiklos registravimas</b>	
<b>Tikslas/uždavinys.</b> Automatiškai registruoti ir atpažinti sportinę veiklą.	
<b>Aprašymas.</b> Programėlė automatiškai registruoja/fiksuoja sportines veiklas, bei jų rezultatus.	
<b>Prieš-sąlyga</b>	Vartotojas ketina pradėti registruoti sportinę veiklą.
<b>Aktorius</b>	Vartotojas.
<b>Sužadavimo sąlyga</b>	Atsirado noras pradėti registruoti sportinę veiklą.
<b>Veiklos taisyklės</b>	Vartotojas turi pasirinkti automatinio sportinės veiklos registravimo funkciją iš sistemos meniu punkto „Treniruotė“.
<b>Pagrindinis scenarijus</b>	
<b>Pagrindinis įvykių srautas</b>	<b>Sistemos reakcija</b>

1. Vartotojas pasirenka automatinį sportinės veiklos registravimą paspausdamas meniu punktą „Treniruotė“.	1.1. Sistema atidaro treniruotės pasirinkimo langą.
3. Vartotojas pasirenka treniruotės tikslą iš sąrašo.	2.1 Sistema atidaro naują langą automatiniam sportinės veiklos registravimui.
3. Registruojama sportinė veiklą ir jos duomenys.	3.1. Sistema automatiškai atpažįsta ir registruoja sportinės veiklą ir jos rezultatus.
3. Sportinės veiklos duomenų atvaizdavimas išmaniojo telefono ekrane.	3.1. Sistema atvaizduoja duomenis į ekraną.
<b>Po-sąlyga</b>	1. Atvaizduojama į išmaniojo telefono ekraną sportinės veiklos rezultatai. 2. Sukuriamas naujas sportinės veiklos įrašas rezultatų sąrašė. 3. Vartotojas redaguoja sportinės veiklos rezultatus.
<b>Alternatyvūs scenarijai</b>	
1. Sportinės veiklos duomenų redagavimas ranka.	1.1. Sistema vartotojui leidžia redaguoti sportinės veiklos duomenis.

<b>PA 5 Duomenų registravimas</b>	
<b>Tikslas/uždavinys.</b> Išsaugoti jutiklių surinktus duomenis į failą.	
<b>Aprašymas.</b> Sportinės veiklos metu jutiklių užfiksuotų duomenų išsaugojimas į failą po automatinio sportinės veiklos atpažinimo nutraukimo.	
<b>Prieš-sąlyga</b>	Vartotojas ketina išsaugoti jutiklių užfiksuotus treniruotės duomenis.
<b>Aktorius</b>	Vartotojas.
<b>Sužadavimo sąlyga</b>	Noras išsaugoti treniruotės duomenis.
<b>Veiklos taisyklės</b>	Sportinės veiklos jutiklių užfiksuotų duomenų išsaugojimas galimas tik užbaigus automatinį sportinės veiklos atpažinimą ir pažymėjus varnelę „Išsaugoti į failą“.
<b>Pagrindinis scenarijus</b>	
<b>Pagrindinis įvykių srautas</b>	<b>Sistemos reakcija</b>
1. Pažymima varnelė „Išsaugoti į failą“.	1.1. Sistema atlieka grafines vizualizacijas.
2. Užbaigiamas automatinis sportinės veiklos atpažinimas.	2.1. Sistema išsaugo sukauptus sportinės veiklos duomenis į failą.
<b>Po-sąlyga</b>	Išmaniojo telefono fizinėje atmintyje atsiranda failas su sportinės veiklos duomenimis.
<b>Alternatyvūs scenarijai</b>	
Nėra	

<b>PA 6 Redaguoti treniruotės rezultatus</b>	
<b>Tikslas/uždavinys.</b> Redaguoti sportinės veiklos rezultatus.	
<b>Aprašymas.</b> Sportinės veiklos rezultatų redagavimas/ištrynimasis.	
<b>Prieš-sąlyga</b>	Vartotojas ketina redaguoti treniruotės rezultatus.
<b>Aktorius</b>	Vartotojas.
<b>Sužadavimo sąlyga</b>	Noras redaguoti treniruotės rezultatus.
<b>Veiklos taisyklės</b>	Sportinės veiklos rezultatų redagavimas galimas po automatinio sportinės veiklos atpažinimo nutraukimo arba iš sportinės veiklos rezultatų sąrašo.
<b>Pagrindinis scenarijus</b>	
<b>Pagrindinis įvykių srautas</b>	<b>Sistemos reakcija</b>
1. Pasirenkamas treniruotės rezultato įrašas iš treniruotės rezultatų sąrašo arba baigiamas automatinis sportinės veiklos atpažinimas.	1.1. Sistema atidaro treniruotės rezultatų redagavimo langą.
2. Redaguojami treniruotės rezultatų duomenys.	2.1. Sistema atlieka grafines vizualizacijas.
3. Išsaugomas treniruotės rezultatas.	3.1. Sistema išsaugo treniruotės rezultatų duomenis į duomenų bazę.
<b>Po-sąlyga</b>	Treniruotės rezultatų sąrašė matomas atnaujintas įrašas.
<b>Alternatyvūs scenarijai</b>	
1. Pasirenkamas treniruotės rezultatų įrašas iš treniruotės rezultatų sąrašo.	1.1. Sistema atidaro treniruotės rezultatų redagavimo langą.

2. Paspaudžiamas treniruotės rezultatų trynimo mygtukas.	2.1. Sistema išsaugo treniruotės rezultatų duomenis į duomenų bazę.
<b>Po-sąlyga</b>	Treniruotės rezultatų sąrašė įrašo nebelieka.

<b>PA 7 Peržiūrėti treniruočių rezultatus</b>	
<b>Tikslas/uždavinys.</b> Peržiūrėti treniruočių rezultatus.	
<b>Aprašymas.</b> Vartotojas gali peržiūrėti treniruočių rezultatus iš esamų rezultatų sąrašo.	
<b>Prieš-sąlyga</b>	Vartotojas ketina peržiūrėti treniruočių rezultatus.
<b>Aktorius</b>	Vartotojas.
<b>Sužadavimo sąlyga</b>	Atsirado poreikis peržiūrėti treniruočių rezultatus.
<b>Veiklos taisyklės</b>	Privaloma pasirinkti meniu punktą „Rezultatai“, norint peržiūrėti treniruočių rezultatus.
<b>Pagrindinis scenarijus</b>	
<b>Pagrindinis įvykių srautas</b>	<b>Sistemos reakcija</b>
1. Vartotojas paspaudžia meniu punktą „Rezultatai“	1.1. Sistema užfiksuoja vartotojo pasirinkimą. 1.2. Sistema atidaro naują langą.
<b>Po-sąlyga</b>	Atveriamas treniruočių rezultatų sąrašas.
<b>Alternatyvūs scenarijai</b>	Nėra

### 2.1.3. Funkciniai reikalavimai

<b>Reikalavimas #:</b>	<b>FR-01</b>	<b>Reikalavimo tipas:</b>	<b>V9a</b>	<b>PA/FR #:</b>	<b>PA4, PA6, PA7</b>
<b>Aprašymas:</b>	Sistema turi atvaizduoti sportinių veiklų rezultatų duomenis ekrane.				
<b>Pagrindimas:</b>	Duomenų vaizdavimas ekrane yra privalomas, jeigu norime atlikti pagrindines sistemos funkcijas.				
<b>Šaltinis:</b>	Projekto kūrėjas.				
<b>Atitikimo kriterijus:</b>	Duomenys atvaizduojami ekrane grafiškai.				
<b>Užsakovo pasitenkinimas:</b>	5	<b>Užsakovo nepasitenkinimas:</b>	1		
<b>Prioritetas:</b>	1	<b>Konfliktai:</b>	Nėra		
<b>Istorija:</b>	Sukurtas: 2016-11-28				

<b>Reikalavimas #:</b>	<b>FR-02</b>	<b>Reikalavimo tipas:</b>	<b>V9a</b>	<b>PA/FR #:</b>	<b>PA6</b>
<b>Aprašymas:</b>	Sistema turi leisti redaguoti sportinės veiklos rezultatų duomenis.				
<b>Pagrindimas:</b>	Užfiksavus neteisingus ar netikslus sportinės veiklos duomenis vartotojas turi galėti juos koreguoti.				
<b>Šaltinis:</b>	Projekto kūrėjas.				
<b>Atitikimo kriterijus:</b>	Leisti redaguoti rezultatų duomenis.				
<b>Užsakovo pasitenkinimas:</b>	5	<b>Užsakovo nepasitenkinimas:</b>	1		
<b>Prioritetas:</b>	1	<b>Konfliktai:</b>	Nėra		
<b>Istorija:</b>	Sukurtas: 2016-11-28				

<b>Reikalavimas #:</b>	<b>FR-03</b>	<b>Reikalavimo tipas:</b>	<b>V9a</b>	<b>PA/FR #:</b>	<b>PA6</b>
------------------------	--------------	---------------------------	------------	-----------------	------------

<b>Aprašymas:</b>	Sistema turi leisti ištrinti sportinės veiklos rezultatų duomenis.		
<b>Pagrindimas:</b>	Užfiksavus neteisingus ar netikslus sportinės veiklos duomenis vartotojas turi galėti juos ištrinti.		
<b>Šaltinis:</b>	Projekto kūrėjas.		
<b>Atitikimo kriterijus:</b>	Ištrinti nekokybiškus sportinės veiklos rezultatų duomenis.		
<b>Užsakovo pasitenkinimas:</b>	4	<b>Užsakovo nepasitenkinimas:</b>	1
<b>Prioritetas:</b>	1	<b>Konfliktai:</b>	Nėra
<b>Istorija:</b>	Sukurtas: 2016-11-28		

<b>Reikalavimas #:</b>	<b>FR-04</b>	<b>Reikalavimo tipas:</b>	<b>V9a</b>	<b>PA/FR #:</b>	<b>PA1, PA2, PA3</b>
<b>Aprašymas:</b>	Sistema turi atvaizduoti treniruočių tikslų duomenis ekrane.				
<b>Pagrindimas:</b>	Duomenų vaizdavimas ekrane yra privalomas, jeigu norime atlikti pagrindines sistemos funkcijas.				
<b>Šaltinis:</b>	Projekto kūrėjas.				
<b>Atitikimo kriterijus:</b>	Duomenys atvaizduojami ekrane grafiškai.				
<b>Užsakovo pasitenkinimas:</b>	5	<b>Užsakovo nepasitenkinimas:</b>	1		
<b>Prioritetas:</b>	1	<b>Konfliktai:</b>	Nėra		
<b>Istorija:</b>	Sukurtas: 2016-11-28				

<b>Reikalavimas #:</b>	<b>FR-05</b>	<b>Reikalavimo tipas:</b>	<b>V9a</b>	<b>PA/FR #:</b>	<b>PA2</b>
<b>Aprašymas:</b>	Sistema turi leisti sukurti treniruotės tikslo planą.				
<b>Pagrindimas:</b>	Vartotojas atlikdamas sportinę veiklą turi sekti jos planą, tam turi būti galimybė sukurti treniruotės planą.				
<b>Šaltinis:</b>	Projekto kūrėjas.				
<b>Atitikimo kriterijus:</b>	Treniruotės tikslo plano sukūrimas.				
<b>Užsakovo pasitenkinimas:</b>	5	<b>Užsakovo nepasitenkinimas:</b>	1		
<b>Prioritetas:</b>	1	<b>Konfliktai:</b>	Nėra		
<b>Istorija:</b>	Sukurtas: 2016-11-28				

<b>Reikalavimas #:</b>	<b>FR-06</b>	<b>Reikalavimo tipas:</b>	<b>V9a</b>	<b>PA/FR #:</b>	<b>PA3</b>
<b>Aprašymas:</b>	Sistema turi leisti redaguoti treniruotės tikslo planą.				
<b>Pagrindimas:</b>	Vartotojas turi galėti redaguoti treniruotės tikslo planą, jei duomenys buvo suvesti neteisingai.				
<b>Šaltinis:</b>	Projekto kūrėjas.				
<b>Atitikimo kriterijus:</b>	Treniruotės tikslo plano redagavimas.				
<b>Užsakovo pasitenkinimas:</b>	5	<b>Užsakovo nepasitenkinimas:</b>	5		
<b>Prioritetas:</b>	1	<b>Konfliktai:</b>	Nėra		
<b>Istorija:</b>	Sukurtas: 2016-11-28				

<b>Reikalavimas #:</b>	<b>FR-07</b>	<b>Reikalavimo tipas:</b>	<b>V9a</b>	<b>PA/FR #:</b>	<b>PA3</b>
<b>Aprašymas:</b>	Sistema turi leisti ištrinti treniruotės tikslo planą.				



<b>Pagrindimas:</b>	Vartotojas turi galėti ištrinti treniruotės tikslo planą, jei treniruotės planas nebereikalingas.		
<b>Šaltinis:</b>	Projekto kūrėjas.		
<b>Atitikimo kriterijus:</b>	Treniruotės tikslo plano ištrinimas.		
<b>Užsakovo pasitenkinimas:</b>	5	<b>Užsakovo nepasitenkinimas:</b>	1
<b>Prioritetas:</b>	1	<b>Konfliktai:</b>	Nėra
<b>Istorija:</b>	Sukurtas: 2016-11-28		

<b>Reikalavimas #:</b>	<b>FR-08</b>	<b>Reikalavimo tipas:</b>	<b>V9a</b>	<b>PA/FR #:</b>	<b>PA4</b>
<b>Aprašymas:</b>	Treniruotės tikslo planas turi būti pasirenkamas prieš automatinės sportinės veiklos atpažinimą.				
<b>Pagrindimas:</b>	Vartotojas sportinės veiklos metu turi sekti treniruotės planą, tam prieš automatinį sportinės veiklos atpažinimą yra reikalingas treniruotės tikslo plano pasirinkimas.				
<b>Šaltinis:</b>	Projekto kūrėjas.				
<b>Atitikimo kriterijus:</b>	Treniruotės tikslo plano pasirinkimas.				
<b>Užsakovo pasitenkinimas:</b>	5	<b>Užsakovo nepasitenkinimas:</b>	2		
<b>Prioritetas:</b>	1	<b>Konfliktai:</b>	Nėra		
<b>Istorija:</b>	Sukurtas: 2016-11-28				

<b>Reikalavimas #:</b>	<b>FR-09</b>	<b>Reikalavimo tipas:</b>	<b>V9a</b>	<b>PA/FR #:</b>	<b>PA4</b>
<b>Aprašymas:</b>	Sportinės veiklos progresas turi būti fiksuojamas automatiškai.				
<b>Pagrindimas:</b>	Sportinės veiklos pratimų pakartojimai turi būti fiksuojami automatiškai.				
<b>Šaltinis:</b>	Projekto kūrėjas.				
<b>Atitikimo kriterijus:</b>	Automatinis sportinės veiklos pratimų pakartojimai užfiksavimas.				
<b>Užsakovo pasitenkinimas:</b>	5	<b>Užsakovo nepasitenkinimas:</b>	1		
<b>Prioritetas:</b>	2	<b>Konfliktai:</b>	Nėra		
<b>Istorija:</b>	Sukurtas: 2016-11-28				

<b>Reikalavimas #:</b>	<b>FR-10</b>	<b>Reikalavimo tipas:</b>	<b>V9a</b>	<b>PA/FR #:</b>	<b>PA4</b>
<b>Aprašymas:</b>	Sistema turi automatiškai atpažinti besikeičiančias sportines veiklas.				
<b>Pagrindimas:</b>	Sistema automatinio sportinių veiklų atpažinimo režimo metu turi atpažinti daugiau nei vieną sportinę veiklą.				
<b>Šaltinis:</b>	Projekto kūrėjas.				
<b>Atitikimo kriterijus:</b>	Automatinis kelių sportinių veiklų atpažinimas.				
<b>Užsakovo pasitenkinimas:</b>	5	<b>Užsakovo nepasitenkinimas:</b>	1		
<b>Prioritetas:</b>	1	<b>Konfliktai:</b>	Nėra		
<b>Istorija:</b>	Sukurtas: 2016-11-28				

<b>Reikalavimas #:</b>	<b>FR-11</b>	<b>Reikalavimo tipas:</b>	<b>V9a</b>	<b>PA/FR #:</b>	<b>PA4</b>
<b>Aprašymas:</b>	Sportinių veiklų atpažinimas turi būti vykdomas realiu laiku.				
<b>Pagrindimas:</b>	Sistema turi vykdyti sportinių veiklų atpažinimą ir jų rezultatų fiksavimą realiu laiku, kad vartotojas galėtų sekti treniruotės planą.				

<b>Šaltinis:</b>	Projekto kūrėjas.		
<b>Atitikimo kriterijus:</b>	Sportinės veiklos rezultatų atvaizdavimas ekrane turi tilpti į 0 – 5 sekundžių duomenų atvaizdavimo vėlavimo intervalą.		
<b>Užsakovo pasitenkinimas:</b>	5	<b>Užsakovo nepasitenkinimas:</b>	1
<b>Prioritetas:</b>	1	<b>Konfliktai:</b>	Nėra
<b>Istorija:</b>	Sukurtas: 2016-11-28		

<b>Reikalavimas #:</b>	<b>FR-12</b>	<b>Reikalavimo tipas:</b>	<b>V9a</b>	<b>PA/FR #:</b>	<b>PA4</b>
<b>Aprašymas:</b>	Sportinės veiklos atpažinimui turi būti naudojamas mašininio mokymosi algoritmas.				
<b>Pagrindimas:</b>	Sportinės veiklos atpažinimui ir registravimui turi būti naudojamas mašininio mokymosi algoritmas, kuris leistu pasiekti aukštą duomenų klasifikavimo lygį, sumažintų sistemos apkrovą, ir sumažintu programinio kodo sudėtingumą.				
<b>Šaltinis:</b>	Projekto kūrėjas.				
<b>Atitikimo kriterijus:</b>	Sportinės veiklos atpažinimui ir registravimui panaudotas mašininio mokymosi algoritmas.				
<b>Užsakovo pasitenkinimas:</b>	3	<b>Užsakovo nepasitenkinimas:</b>	1		
<b>Prioritetas:</b>	2	<b>Konfliktai:</b>	Nėra		
<b>Istorija:</b>	Sukurtas: 2016-11-28				

<b>Reikalavimas #:</b>	<b>FR-13</b>	<b>Reikalavimo tipas:</b>	<b>V9a</b>	<b>PA/FR #:</b>	<b>PA5</b>
<b>Aprašymas:</b>	Jutiklių užfiksuoti duomenys pasirinktinai turi būti saugomi duomenų faile.				
<b>Pagrindimas:</b>	Vartotojas turi turėti galimybę išsaugoti savo sportinės veiklos duomenis į failą, kad esant poreikiui juos galėtų panaudoti detalesnei analizei pasitelkiant kitus analizavimo įrankius.				
<b>Šaltinis:</b>	Projekto kūrėjas.				
<b>Atitikimo kriterijus:</b>	Duomenys yra išsaugomi tekstiniame faile.				
<b>Užsakovo pasitenkinimas:</b>	2	<b>Užsakovo nepasitenkinimas:</b>	1		
<b>Prioritetas:</b>	2	<b>Konfliktai:</b>	Nėra		
<b>Istorija:</b>	Sukurtas: 2016-11-28				

<b>Reikalavimas #:</b>	<b>FR-14</b>	<b>Reikalavimo tipas:</b>	<b>V9a</b>	<b>PA/FR #:</b>	<b>PA4, PA6</b>
<b>Aprašymas:</b>	Sportinės veiklos rezultatai turi būti saugojami duomenų bazėje.				
<b>Pagrindimas:</b>	Vartotojas turi turėti galimybę peržiūrėti, redaguoti ir trinti savo sportinės veiklos duomenis, todėl šie duomenys turi būti saugomi duomenų bazėje.				
<b>Šaltinis:</b>	Projekto kūrėjas.				
<b>Atitikimo kriterijus:</b>	Sportinių veiklų duomenys saugomi duomenų bazėje.				
<b>Užsakovo pasitenkinimas:</b>	5	<b>Užsakovo nepasitenkinimas:</b>	1		
<b>Prioritetas:</b>	1	<b>Konfliktai:</b>	Nėra		
<b>Istorija:</b>	Sukurtas: 2016-11-28				

<b>Reikalavimas #:</b>	<b>FR-15</b>	<b>Reikalavimo tipas:</b>	<b>V9a</b>	<b>PA/FR #:</b>	<b>PA2, PA3</b>
<b>Aprašymas:</b>	Treniruočių tikslo planas turi būti saugojamas duomenų bazėje.				

<b>Pagrindimas:</b>	Vartotojas turi turėti galimybę peržiūrėti, kurti, redaguoti ir trinti savo treniruočių plano duomenis, todėl šie duomenys turi būti saugomi duomenų bazėje.		
<b>Šaltinis:</b>	Projekto kūrėjas.		
<b>Atitikimo kriterijus:</b>	Treniruočių plano duomenys saugomi duomenų bazėje.		
<b>Užsakovo pasitenkinimas:</b>	5	<b>Užsakovo nepasitenkinimas:</b>	1
<b>Prioritetas:</b>	1	<b>Konfliktai:</b>	Nėra
<b>Istorija:</b>	Sukurtas: 2016-11-28		

<b>Reikalavimas #:</b>	<b>FR-16</b>	<b>Reikalavimo tipas:</b>	<b>V9a</b>	<b>PA/FR #:</b>	<b>PA3, PA6</b>
<b>Aprašymas:</b>	Visi duomenų bazėje esantys duomenys negali būti fiziškai ištrinami.				
<b>Pagrindimas:</b>	Vartotojas neturi galėti fiziškai ištrinti duomenų iš duomenų bazės, kad nesugadintų duomenų ryšių tarp lentelių.				
<b>Šaltinis:</b>	Projekto kūrėjas.				
<b>Atitikimo kriterijus:</b>	Duomenų trynimas bus atliekamas pakeičiant jų atvaizdavimo būseną iš matomas į nematomą.				
<b>Užsakovo pasitenkinimas:</b>	4	<b>Užsakovo nepasitenkinimas:</b>	1		
<b>Prioritetas:</b>	1	<b>Konfliktai:</b>	Nėra		
<b>Istorija:</b>	Sukurtas: 2016-11-28				

## 2.2. Nefunkciniai reikalavimai

### 2.2.1. Reikalavimai sistemos išvaizdai

#### 2.2.1.1. Išvaizda

<b>Reikalavimas #:</b>	<b>NF-01</b>	<b>Reikalavimo tipas:</b>	<b>V10a</b>	<b>PA/FR #:</b>	<b>PA1 – PA7</b>
<b>Aprašymas:</b>	Sistema privalo veikti naudojant grafinę sąsają.				
<b>Pagrindimas:</b>	Duomenų ir valdymo funkcijų atvaizdavimui reikalinga grafinė vartotojo sąsaja, kad vartotojas galėtų valdyti ir stebėti sistemos darbą.				
<b>Šaltinis:</b>	Projekto kūrėjas.				
<b>Atitikimo kriterijus:</b>	Visiems PA turi būti realizuota grafinė vartotojo sąsaja.				
<b>Užsakovo pasitenkinimas:</b>	5	<b>Užsakovo nepasitenkinimas:</b>	1		
<b>Prioritetas:</b>	1	<b>Konfliktai:</b>	Nėra		
<b>Istorija:</b>	Sukurtas: 2016-11-28				

#### 2.2.1.2. Stilius

<b>Reikalavimas #:</b>	<b>NF-02</b>	<b>Reikalavimo tipas:</b>	<b>V10b</b>	<b>PA/FR #:</b>	<b>PA1 – PA7</b>
<b>Aprašymas:</b>	Išlaikyti vientisą stilių visoje sistemoje.				
<b>Pagrindimas:</b>	Sistemos estetinio vaizdo atvaizdavimui reikalingas vientisas vieno stiliaus dizainas.				
<b>Šaltinis:</b>	Projekto kūrėjas.				
<b>Atitikimo kriterijus:</b>	Turi būti išlaikytas vientisas sistemos grafinės vartotojo sąsajos stilius naudojant stilių šablonus.				
<b>Užsakovo pasitenkinimas:</b>	4	<b>Užsakovo nepasitenkinimas:</b>	1		
<b>Prioritetas:</b>	1	<b>Konfliktai:</b>	Nėra		

Istorija:	Sukurtas: 2016-11-28
-----------	----------------------

## 2.2.2. Reikalavimai panaudojimui

### 2.2.2.1. Naudojimosi paprastumas

<b>Reikalavimas #:</b>	<b>NF-03</b>	<b>Reikalavimo tipas:</b>	<b>V11a</b>	<b>PA/FR #:</b>	<b>PA1 – PA7</b>
<b>Aprašymas:</b>	Minimalistinis sistemos dizainas.				
<b>Pagrindimas:</b>	Sistemą bus naudojama per grafinę vartotojo sąsają, kuri yra reikalinga sistemos valdymui ir duomenų atvaizdavimui, kadangi sistema veiks išmaniajame telefone su riboto dydžio ekranu, todėl yra būtina užtikrinti duomenų ir funkcijų įskaitomumą.				
<b>Šaltinis:</b>	Projekto kūrėjas.				
<b>Atitikimo kriterijus:</b>	Sistemoje bus atvaizduojami tik svarbiausi duomenys ir funkcionalumas.				
<b>Užsakovo pasitenkinimas:</b>	5	<b>Užsakovo nepasitenkinimas:</b>	1		
<b>Prioritetas:</b>	1	<b>Konfliktai:</b>	Nėra		
<b>Istorija:</b>	Sukurtas: 2016-11-28				

### 2.2.2.2. Vartotojui skirtos savybių ir kalbos konfigūravimo priemonės

<b>Reikalavimas #:</b>	<b>NF-04</b>	<b>Reikalavimo tipas:</b>	<b>V11b</b>	<b>PA/FR #:</b>	<b>PA1 – PA7</b>
<b>Aprašymas:</b>	Pirminėje sistemos versijoje nebus vartotojui skirtų savybių ir kalbos konfigūravimo priemonių.				
<b>Pagrindimas:</b>	Pirminė sistemos versija bus skirta Lietuvos rinkai ir sistemos funkcionalumas bus formuojamas pagal Lietuvos rinkos standartus.				
<b>Šaltinis:</b>	Projekto kūrėjas.				
<b>Atitikimo kriterijus:</b>	Pirminė sistemos versija pritaikyta Lietuvos rinkai.				
<b>Užsakovo pasitenkinimas:</b>	2	<b>Užsakovo nepasitenkinimas:</b>	3		
<b>Prioritetas:</b>	1	<b>Konfliktai:</b>	Nėra		
<b>Istorija:</b>	Sukurtas: 2016-11-28				

### 2.2.2.3. Mokymosi reikalavimai

<b>Reikalavimas #:</b>	<b>NF-05</b>	<b>Reikalavimo tipas:</b>	<b>V11c</b>	<b>PA/FR #:</b>	<b>PA1 – PA7</b>
<b>Aprašymas:</b>	Kuriama sistema yra skirta patyrusiam vartotojui ir turi nesukelti problemų valdyti sistemą per grafinę vartotojo sąsają.				
<b>Pagrindimas:</b>	Esant nepatyrusiam vartotojui jis gali nemokėti naudotis sistema.				
<b>Šaltinis:</b>	Projekto kūrėjas.				
<b>Atitikimo kriterijus:</b>	Esant nepatyrusiam vartotojui yra reikalingi apmokymai.				
<b>Užsakovo pasitenkinimas:</b>	4	<b>Užsakovo nepasitenkinimas:</b>	1		
<b>Prioritetas:</b>	1	<b>Konfliktai:</b>	Nėra		
<b>Istorija:</b>	Sukurtas: 2016-11-28				

### 2.2.2.4. Suprantamumas ir mandagumas

<b>Reikalavimas #:</b>	<b>NF-06</b>	<b>Reikalavimo tipas:</b>	<b>V11d</b>	<b>PA/FR #:</b>	<b>PA1 – PA7</b>
<b>Aprašymas:</b>	Sistema naudos žodžius ir simbolius kurie yra suprantami vartotojui.				
<b>Pagrindimas:</b>	Sistema neturės pagalbos lango, todėl sistemoje naudojami žodžiai ir simboliai turi būti suprantami.				

<b>Šaltinis:</b>	Projekto kūrėjas.		
<b>Atitikimo kriterijus:</b>	Sistemoje bus naudojami vartotojui suprantami žodžiai ir simboliai.		
<b>Užsakovo pasitenkinimas:</b>	4	<b>Užsakovo nepasitenkinimas:</b>	3
<b>Prioritetas:</b>	1	<b>Konfliktai:</b>	Nėra
<b>Istorija:</b>	Sukurtas: 2016-11-28		

### 2.2.2.5. Prieinamumas neįgaliesiems

<b>Reikalavimas #:</b>	<b>NF-07</b>	<b>Reikalavimo tipas:</b>	<b>V11e</b>	<b>PA/FR #:</b>	<b>PA1 – PA7</b>
<b>Aprašymas:</b>	Sistemą naudos tik vartotojai, su tam tikra IT kvalifikacija.				
<b>Pagrindimas:</b>	Vartotojas pilnai sugeba valdyti išmanųjį telefoną bei suprasti ką jame daro.				
<b>Šaltinis:</b>	Projekto kūrėjas.				
<b>Atitikimo kriterijus:</b>	Sistemos valdymas nebus pritaikytas neįgaliesiems.				
<b>Užsakovo pasitenkinimas:</b>	4	<b>Užsakovo nepasitenkinimas:</b>	1		
<b>Prioritetas:</b>	1	<b>Konfliktai:</b>	Nėra		
<b>Istorija:</b>	Sukurtas: 2016-11-28				

### 2.2.3. Reikalavimai vykdymo savybėms

#### 2.2.3.1. Užduočių vykdymo greitis

<b>Reikalavimas #:</b>	<b>NF-08</b>	<b>Reikalavimo tipas:</b>	<b>V12a</b>	<b>PA/FR #:</b>	<b>PA4</b>
<b>Aprašymas:</b>	Sistema turi sugebėti atlikti 10 jutiklių užfiksuotų duomenų nuskaitymo operacijų per sekundę.				
<b>Pagrindimas:</b>	Surinkus mažesniai kiekiui duomenų mašininio mokymosi algoritmas negalės teisingai nustatyti sportinės veiklos.				
<b>Šaltinis:</b>	Projekto kūrėjas.				
<b>Atitikimo kriterijus:</b>	Jutiklių duomenys bus registruojami 10 kartų per sekundę dažniu.				
<b>Užsakovo pasitenkinimas:</b>	5	<b>Užsakovo nepasitenkinimas:</b>	1		
<b>Prioritetas:</b>	1	<b>Konfliktai:</b>	Nėra		
<b>Istorija:</b>	Sukurtas: 2016-11-28				

#### 2.2.3.2. Darbo saugos reikalavimai

Nėra

#### 2.2.3.3. Reikalavimai tikslumui

<b>Reikalavimas #:</b>	<b>NF-09</b>	<b>Reikalavimo tipas:</b>	<b>V12c</b>	<b>PA/FR #:</b>	<b>PA4</b>
<b>Aprašymas:</b>	Sistemos galima sportinės veiklos pakartojimų fiksavimo paklaida yra 5% viso fiksavimo kiekio.				
<b>Pagrindimas:</b>	Tik su 5% paklaida bus galima preliminariai teisingai nustatyti sportinę veiklą ir jos rezultatus.				
<b>Šaltinis:</b>	Projekto kūrėjas.				
<b>Atitikimo kriterijus:</b>	Mašininio mokymosi algoritmo atpažinimo kokybė turi būti nemažesne nei 95%.				
<b>Užsakovo pasitenkinimas:</b>	5	<b>Užsakovo nepasitenkinimas:</b>	1		
<b>Prioritetas:</b>	1	<b>Konfliktai:</b>	Nėra		
<b>Istorija:</b>	Sukurtas: 2016-11-28				

#### 2.2.3.4. Patikimumas ir pasiekiamumas,

<b>Reikalavimas #:</b>	<b>NF-10</b>	<b>Reikalavimo tipas:</b>	<b>V12d</b>	<b>PA/FR #:</b>	<b>PA4</b>
<b>Aprašymas:</b>	Sistema be pertraukimų turi veikti viso sportinės veiklos fiksavimo metu.				
<b>Pagrindimas:</b>	Sportinės veiklos registravimo metu turi būti galimybė sistema perkelti į foninį veikimą.				
<b>Šaltinis:</b>	Projekto kūrėjas.				
<b>Atitikimo kriterijus:</b>	Esant būtinybei turi būti galim sistemos darbą perkelti į foninio veikimo režimą.				
<b>Užsakovo pasitenkinimas:</b>	5	<b>Užsakovo nepasitenkinimas:</b>	1		
<b>Prioritetas:</b>	1	<b>Konfliktai:</b>	Nėra		
<b>Istorija:</b>	Sukurtas: 2016-11-28				

#### 2.2.3.5. Atsparumas trukdžiams

<b>Reikalavimas #:</b>	<b>NF-11</b>	<b>Reikalavimo tipas:</b>	<b>V12e</b>	<b>PA/FR #:</b>	<b>PA1 – PA7</b>
<b>Aprašymas:</b>	Sistema turi sėkmingai pasileisti po elektros dingimo.				
<b>Pagrindimas:</b>	Dingus išmaniojo telefono elektros šaltiniui įrenginys nustos veikti.				
<b>Šaltinis:</b>	Projekto kūrėjas.				
<b>Atitikimo kriterijus:</b>	Po elektros atsiradimo, sistema privalo sėkmingai pasileisti ir atlikti visas funkcijas.				
<b>Užsakovo pasitenkinimas:</b>	4	<b>Užsakovo nepasitenkinimas:</b>	1		
<b>Prioritetas:</b>	1	<b>Konfliktai:</b>	Nėra		
<b>Istorija:</b>	Sukurtas: 2016-11-28				

#### 2.2.3.6. Reikalavimai apdorojamų duomenų apimtims

<b>Reikalavimas #:</b>	<b>NF-12</b>	<b>Reikalavimo tipas:</b>	<b>V12f</b>	<b>PA/FR #:</b>	<b>PA4</b>
<b>Aprašymas:</b>	Pradedant PĮ gyvavimo laiką, sistema turi fiksuoti minimaliai vienos sportinės veiklos duomenis.				
<b>Pagrindimas:</b>	Minimalus fiksuojamų sportinių veiklų kiekis gali būti viena sportinė veikla.				
<b>Šaltinis:</b>	Projekto kūrėjas.				
<b>Atitikimo kriterijus:</b>	Fiksuoti vieną ir daugiau sportinių veiklų vienu metu.				
<b>Užsakovo pasitenkinimas:</b>	5	<b>Užsakovo nepasitenkinimas:</b>	1		
<b>Prioritetas:</b>	1	<b>Konfliktai:</b>	Nėra		
<b>Istorija:</b>	Sukurtas: 2016-11-28				

#### 2.2.3.7. Reikalavimai išplečiamumui

<b>Reikalavimas #:</b>	<b>NF-13</b>	<b>Reikalavimo tipas:</b>	<b>V12g</b>	<b>PA/FR #:</b>	<b>PA1 – PA7</b>
<b>Aprašymas:</b>	Sistema turi atpažinti ir fiksuoti iki 10 skirtingų sportinių veiklų automatinio sportinės veiklos atpažinimo režimo metu.				
<b>Pagrindimas:</b>	Šiuo metu sistema galima fiksuoti tik tris sportines veiklas.				
<b>Šaltinis:</b>	Projekto kūrėjas.				
<b>Atitikimo kriterijus:</b>	Ateityje planuojam išplėsti sistemą kuri atpažintu ir fiksuotu iki 10 skirtingų sportinių veiklų.				
<b>Užsakovo pasitenkinimas:</b>	5	<b>Užsakovo nepasitenkinimas:</b>	1		
<b>Prioritetas:</b>	1	<b>Konfliktai:</b>	Nėra		
<b>Istorija:</b>	Sukurtas: 2016-11-28				

### 2.2.3.8. Reikalavimai produkto ilgaamžiškumui

<b>Reikalavimas #:</b>	<b>NF-14</b>	<b>Reikalavimo tipas:</b>	<b>V12h</b>	<b>PA/FR #:</b>	<b>PA1 – PA7</b>
<b>Aprašymas:</b>	Produktas gyvuos ne mažiau penkis metus.				
<b>Pagrindimas:</b>	Numanoma jog šis produktas, su minimaliu biudžetu, gyvuos ne mažiau penkis metų, dėl šios temos aktualumo.				
<b>Šaltinis:</b>	Projekto kūrėjas.				
<b>Atitikimo kriterijus:</b>	Produktas gyvuos ne mažiau penkių metų.				
<b>Užsakovo pasitenkinimas:</b>	4	<b>Užsakovo nepasitenkinimas:</b>	1		
<b>Prioritetas:</b>	1	<b>Konfliktai:</b>	Nėra		
<b>Istorija:</b>	Sukurtas: 2016-11-28				

### 2.2.4. Reikalavimai veikimo sąlygoms

#### 2.2.4.1. Numatoma fizinė aplinka

<b>Reikalavimas #:</b>	<b>NF-15</b>	<b>Reikalavimo tipas:</b>	<b>V13a</b>	<b>PA/FR #:</b>	<b>PA1 – PA7</b>
<b>Aprašymas:</b>	Sistema turi pilnai funkcionuoti Android 5.0 operacinėje sistemoje.				
<b>Pagrindimas:</b>	Projektuojama sistema skirta Android 5.0 ir didesnės versijos operacinei sistemai naudojantiems išmaniesiems telefonams.				
<b>Šaltinis:</b>	Projekto kūrėjas.				
<b>Atitikimo kriterijus:</b>	Sistema veikia Android 5.0 operacinėje sistemoje.				
<b>Užsakovo pasitenkinimas:</b>	4	<b>Užsakovo nepasitenkinimas:</b>	2		
<b>Prioritetas:</b>	1	<b>Konfliktai:</b>	Nėra		
<b>Istorija:</b>	Sukurtas: 2016-11-28				

#### 2.2.4.2. Reikalavimai darbui su gretimomis sistemomis

Sistemos veikimas su gretimomis sistemomis nėra numatytas sistemos projektavime.

#### 2.2.4.3. Reikalavimai sistemos platinimo/gamybos formatui

Reikalavimai sistemos platinimo/gamybos formatui nėra nurodyti.

#### 2.2.4.4. Reikalavimai leidybos procesui

<b>Reikalavimas #:</b>	<b>NF-16</b>	<b>Reikalavimo tipas:</b>	<b>V13d</b>	<b>PA/FR #:</b>	<b>PA1 – PA7</b>
<b>Aprašymas:</b>	Sistemos atnaujinimo metu, sistemos prieš tai buvęs funkcionalumas neturi pranykti.				
<b>Pagrindimas:</b>	Vartotojo surinkti duomenys ir nustatytos konfigūracijos negali pradingti po sistemos atnaujinimo, kitaip sistema praras funkcionalumą.				
<b>Šaltinis:</b>	Projekto kūrėjas.				
<b>Atitikimo kriterijus:</b>	Duomenų ir konfigūracijų išsaugojimas prieš sisteminį atnaujinimą.				
<b>Užsakovo pasitenkinimas:</b>	5	<b>Užsakovo nepasitenkinimas:</b>	1		
<b>Prioritetas:</b>	1	<b>Konfliktai:</b>	Nėra		
<b>Istorija:</b>	Sukurtas: 2016-11-28				

## 2.2.5. Reikalavimai sistemos priežiūrai

### 2.2.5.1. Sistemos aptarnavimas

<b>Reikalavimas #:</b>	<b>NF-17</b>	<b>Reikalavimo tipas:</b>	<b>V14a</b>	<b>PA/FR #:</b>	<b>PA1 – PA7</b>
<b>Aprašymas:</b>	Išmaniojo telefono gedimo metu sistema turi būti galima perkelti į kitą įrenginį.				
<b>Pagrindimas:</b>	Sistemos veikimo metu užfiksuoti duomenys ir nustatymai bus saugomi failuose ir duomenų bazėje.				
<b>Šaltinis:</b>	Projekto kūrėjas.				
<b>Atitikimo kriterijus:</b>	Perkelti nustatymus ir užfiksuotus duomenis į kitą Android operacinę sistemą turintį įrenginį.				
<b>Užsakovo pasitenkinimas:</b>	5	<b>Užsakovo nepasitenkinimas:</b>	1		
<b>Prioritetas:</b>	1	<b>Konfliktai:</b>	Nėra		
<b>Istorija:</b>	Sukurtas: 2016-11-28				

### 2.2.5.2. Sistemos palaikymas

<b>Reikalavimas #:</b>	<b>NF-18</b>	<b>Reikalavimo tipas:</b>	<b>V14b</b>	<b>PA/FR #:</b>	<b>PA1 – PA7</b>
<b>Aprašymas:</b>	Sistemos palaikymas yra susitarimo su projekto vykdytojais reikalas.				
<b>Pagrindimas:</b>	Užsakovui mokant tam tikrą mokestį suteikiama galimybė palaikyti sistemą.				
<b>Šaltinis:</b>	Projekto kūrėjas.				
<b>Atitikimo kriterijus:</b>	Sistemos palaikymas.				
<b>Užsakovo pasitenkinimas:</b>	3	<b>Užsakovo nepasitenkinimas:</b>	1		
<b>Prioritetas:</b>	1	<b>Konfliktai:</b>	Nėra		
<b>Istorija:</b>	Sukurtas: 2016-11-28				

### 2.2.5.3. Perkėlimo į kitas platformas reikalavimai

<b>Reikalavimas #:</b>	<b>NF-19</b>	<b>Reikalavimo tipas:</b>	<b>V14c</b>	<b>PA/FR #:</b>	<b>PA1 – PA7</b>
<b>Aprašymas:</b>	Produktas privalo veikti Android operacinėje aplinkoje.				
<b>Pagrindimas:</b>	Projektuojama sistemos numatyta veikimo aplinka yra Android operacinė sistema.				
<b>Šaltinis:</b>	Projekto kūrėjas				
<b>Atitikimo kriterijus:</b>	Kiti platformos reikalavimai nėra paminėti.				
<b>Užsakovo pasitenkinimas:</b>	4	<b>Užsakovo nepasitenkinimas:</b>	2		
<b>Prioritetas:</b>	1	<b>Konfliktai:</b>	Nėra		
<b>Istorija:</b>	Sukurtas: 2016-11-28				

## 2.2.6. Reikalavimai saugumui

### 2.2.6.1. Prieigos reikalavimai (teisės)

Prieigos reikalavimų nėra.

### 2.2.6.2. Vientisumo reikalavimai

<b>Reikalavimas #:</b>	<b>NF-20</b>	<b>Reikalavimo tipas:</b>	<b>V15b</b>	<b>PA/FR #:</b>	<b>PA1 – PA7</b>
<b>Aprašymas:</b>	Sistemos apsauga remsis išmaniojo telefono apsaugos tipu.				



<b>Pagrindimas:</b>	Apsaugos vientisumui gali ir nebūti.		
<b>Šaltinis:</b>	Projekto kūrėjas		
<b>Atitikimo kriterijus:</b>	Išmaniojo telefono apsaugos sistema, pagal vartotojo norus.		
<b>Užsakovo pasitenkinimas:</b>	3	<b>Užsakovo nepasitenkinimas:</b>	2
<b>Prioritetas:</b>	1	<b>Konfliktai:</b>	Nėra
<b>Istorija:</b>	Sukurtas: 2016-11-28		

### 2.2.6.3. Reikalavimai privatumui

<b>Reikalavimas #:</b>	<b>NF-21</b>	<b>Reikalavimo tipas:</b>	<b>V15c</b>	<b>PA/FR #:</b>	<b>PA4, PA5</b>
<b>Aprašymas:</b>	Sistema turi prašyti patvirtinimo registruoti jutiklių duomenis.				
<b>Pagrindimas:</b>	Sistema negali nustatinėti išmaniojo telefono jutiklių duomenų, jeigu vartotojas nėra sutikęs.				
<b>Šaltinis:</b>	Projekto kūrėjas				
<b>Atitikimo kriterijus:</b>	Jutiklių duomenų registracija patvirtinama įsirašius sistemą į išmanųjį telefoną.				
<b>Užsakovo pasitenkinimas:</b>	5	<b>Užsakovo nepasitenkinimas:</b>	1		
<b>Prioritetas:</b>	1	<b>Konfliktai:</b>	Nėra		
<b>Istorija:</b>	Sukurtas: 2016-11-28				

### 2.2.6.4. Audito reikalavimai

<b>Reikalavimas #:</b>	<b>NF-22</b>	<b>Reikalavimo tipas:</b>	<b>V15d</b>	<b>PA/FR #:</b>	<b>PA2, PA3, PA4, PA6</b>
<b>Aprašymas:</b>	Duomenų saugojimas auditui.				
<b>Pagrindimas:</b>	Sistema saugo pasirinktus duomenis, kuriuos galės pateikti audito patikrinimui.				
<b>Šaltinis:</b>	Projekto kūrėjas				
<b>Atitikimo kriterijus:</b>	Programėlės gyvavimo metu pasirinkti užfiksuoti duomenys bus saugomi failuose ir duomenų bazėje.				
<b>Užsakovo pasitenkinimas:</b>	5	<b>Užsakovo nepasitenkinimas:</b>	1		
<b>Prioritetas:</b>	1	<b>Konfliktai:</b>	Nėra		
<b>Istorija:</b>	Sukurtas: 2016-11-28				

### 2.2.6.5. Reikalavimai savisaugai nuo išorinių grėsmių

Sistema neturės apsaugos tapatumui. Kitos išorinės grėsmės nėra žinomos.

### 2.2.7. Kultūriniai – politiniai reikalavimai

#### 2.2.7.1. Kultūriniai reikalavimai

Nėra.

#### 2.2.7.2. Politiniai reikalavimai

<b>Reikalavimas #:</b>	<b>NF-23</b>	<b>Reikalavimo tipas:</b>	<b>V16b</b>	<b>PA/FR #:</b>	<b>PA1 – PA7</b>
<b>Aprašymas:</b>	Sistema neturės politinių reikalavimų.				
<b>Pagrindimas:</b>	Užsakovas prieigos teisės į sistemą nebus ribojamos pagal politinius reikalavimus.				

<b>Šaltinis:</b>	Projekto kūrėjas		
<b>Atitikimo kriterijus:</b>	Užsakovas turės visas prieigas prie sistemos.		
<b>Užsakovo pasitenkinimas:</b>	5	<b>Užsakovo nepasitenkinimas:</b>	1
<b>Prioritetas:</b>	1	<b>Konfliktai:</b>	Nėra
<b>Istorija:</b>	Sukurtas: 2016-11-28		

## 2.2.8. Teisiniai reikalavimai

### 2.2.8.1. Įstatyminiai reikalavimai

<b>Reikalavimas #:</b>	<b>NF-24</b>	<b>Reikalavimo tipas:</b>	<b>V17a</b>	<b>PA/FR #:</b>	<b>PA1 – PA7</b>
<b>Aprašymas:</b>	Sistema negali pažeisti asmens privatumo teisių.				
<b>Pagrindimas:</b>	Sistema negali pažeisti asmens privatumo teisių, todėl vartotojas naudodamasis sistema sutinka su jo išmaniojo telefono jutiklių duomenų fiksavimu.				
<b>Šaltinis:</b>	Projekto kūrėjas				
<b>Atitikimo kriterijus:</b>	Vartotojas sutinka su išmaniojo telefono jutiklių duomenų fiksavimu.				
<b>Užsakovo pasitenkinimas:</b>	5	<b>Užsakovo nepasitenkinimas:</b>	1		
<b>Prioritetas:</b>	1	<b>Konfliktai:</b>	NF-21		
<b>Istorija:</b>	Sukurtas: 2016-11-28				

### 2.2.8.2. Reikalavimai standartams

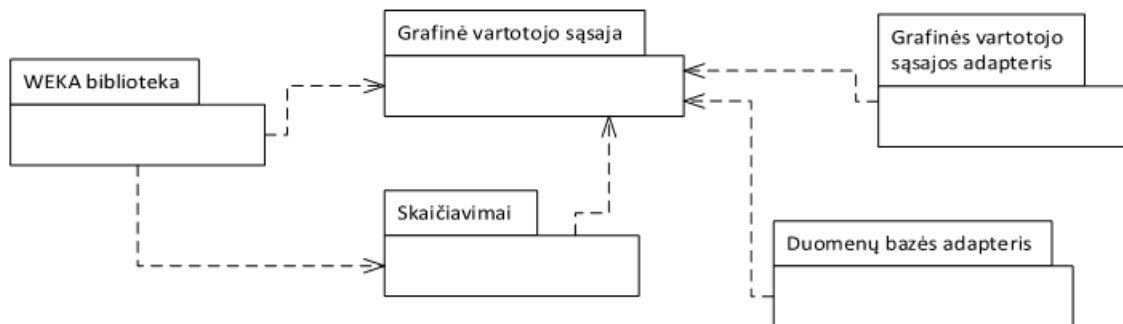
<b>Reikalavimas #:</b>	<b>NF-25</b>	<b>Reikalavimo tipas:</b>	<b>V17b</b>	<b>PA/FR #:</b>	<b>PA1 – PA7</b>
<b>Aprašymas:</b>	Kuriama PĮ turi atitikti SRS standartams.				
<b>Pagrindimas:</b>	Sistemoje esantis funkcionalumas bus aprašytas sistemos reikalavimų specifikacijoje				
<b>Šaltinis:</b>	Projekto kūrėjas				
<b>Atitikimo kriterijus:</b>	Kuriama PĮ atitiks SRS standartams.				
<b>Užsakovo pasitenkinimas:</b>	5	<b>Užsakovo nepasitenkinimas:</b>	1		
<b>Prioritetas:</b>	1	<b>Konfliktai:</b>	Nėra		
<b>Istorija:</b>	Sukurtas: 2016-11-28				

## 2.3. Sistemos statinis vaizdas

Šiame skyriuje pateikiamas statinis sistemos vaizdas (sistemos suskirstymas į paketus) ir pateikta detali sistemos paketų struktūra.

### 2.3.1. Apžvalga

Sistema suskaidyta į keturis paketus, kurie pateikti (11 pav.) paveikslėlyje.



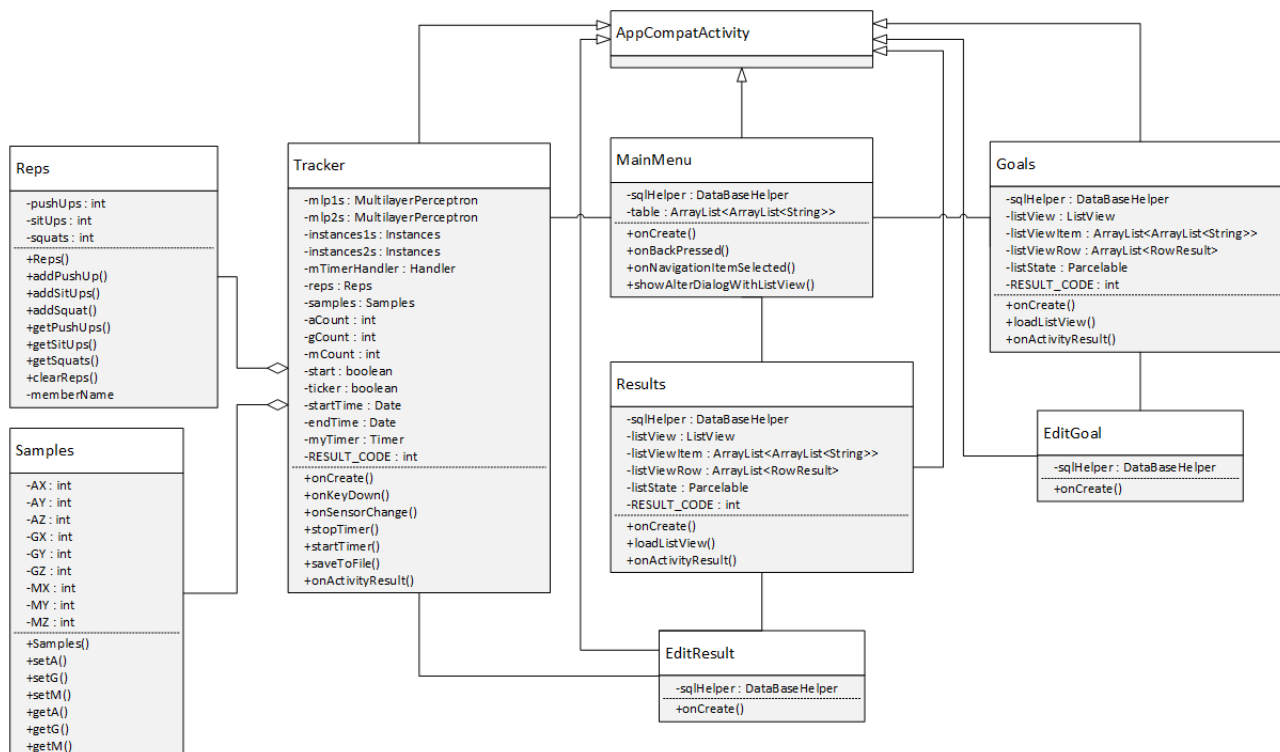
11 pav. Sistemos suskirstymas į paketus

### 2.3.2. Paketų detalizavimas

Šiame skyriuje plačiau detalizuojamas statinis sistemos vaizdas pagal paketus.

#### 2.3.2.1. Paketas „Grafinė vartotojo sąsaja“

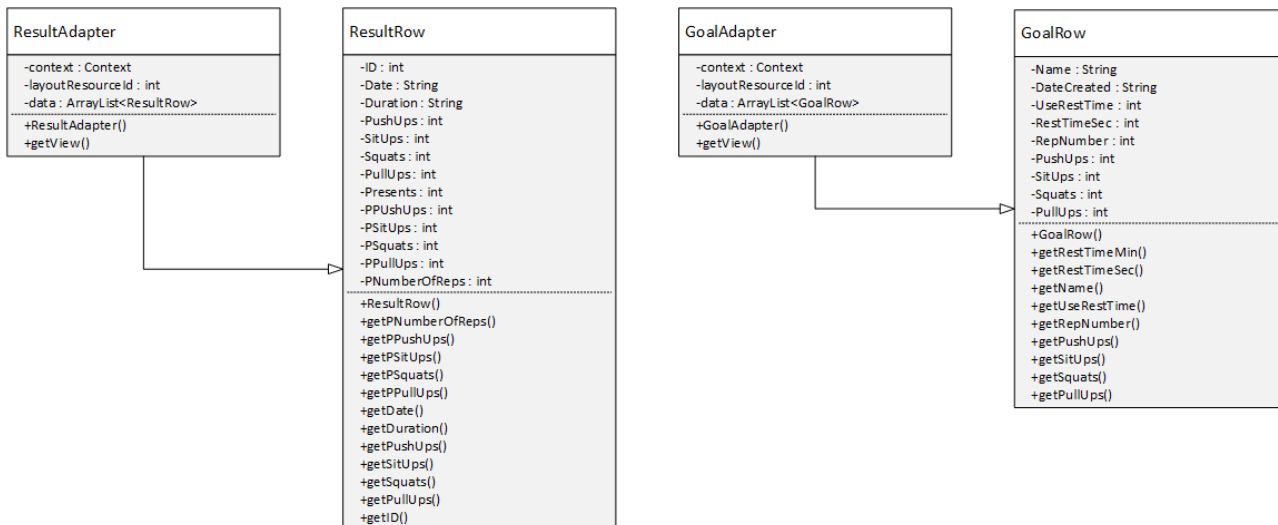
Paketas „*Grafinė vartotojo sąsaja*“ pavaizduotas paveiksle (12 pav.), kuris yra atsakingas už grafinių duomenų atvaizdavimą į ekraną, vartotojo funkcijų vykdymą ir sportinės veiklos atpažinimą bei registravimą.



12 pav. Paketas „Grafinė vartotojo sąsaja“

### 2.3.2.2. Paketas „Grafinės vartotojo sąsajos adapteris“

Paketas „*Grafinės vartotojo sąsajos adapteris*“ pavaizduotas paveiksle (13 pav.), kuris yra skirtas sukurti treniruotės rezultato ir treniruotės tikslo duomenų sąsajoms su grafinės vartotojo sąsajos komponentais treniruočių rezultatams ir treniruočių tikslams atvaizduoti.



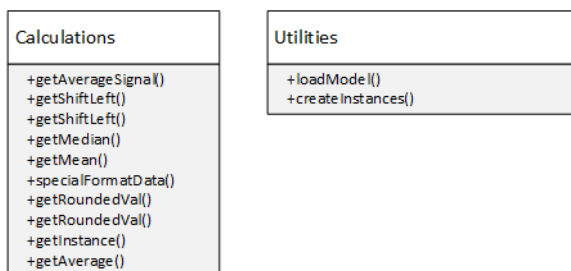
13 pav. Paketas „Grafinės vartotojo sąsajos adapteris“

### 2.3.2.3. Paketas „WEKA biblioteka“

Paketas „*WEKA biblioteka*“ yra WEKA mašininio mokymosi algoritmų biblioteka, kuri bus naudojama apsirrašyti mašininio mokymosi algoritmui ir papildomiems duomenų skaičiavimams, kad būtų galima realizuoti sportinės veiklos atpažinimą.

### 2.3.2.4. Paketas „Skaičiavimai“

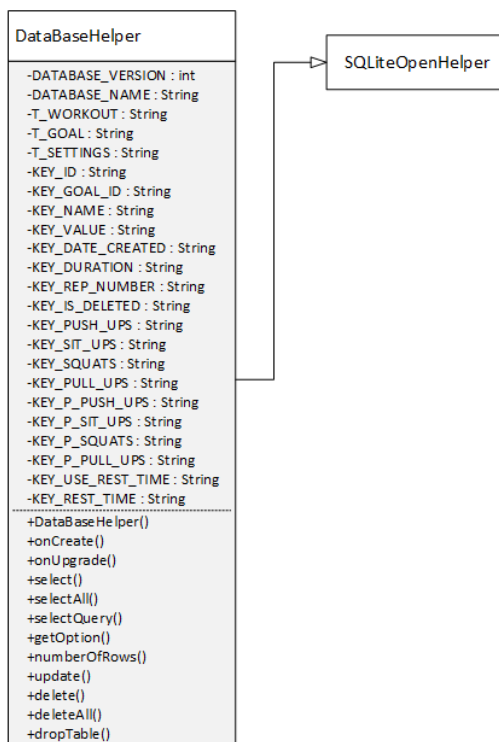
Paketas „*Skaičiavimai*“ pavaizduotas paveiksle (14 pav.), kuris yra skirtas realizuoti papildomiems duomenų skaičiavimams, kurie buvo išskelti į atitinkamas klases, kad būtų prieinami ir galimi naudoti visoje sistemoje.



14 pav. Paketas „Skaičiavimai“

### 2.3.2.5. Paketas „Duomenų bazės adapteris“

Paketas „*Duomenų bazės adapteris*“ pavaizduotas paveiksle (15 pav.), kuris yra skirtas aprašyti sąsajai tarp sistemos ir duomenų SQLite [12] duomenų bazės. Taip pat aprašyta duomenų bazės struktūra ir reikalingi metodai.



15 pav. Paketas „Duomenų bazės adapteris“

## 2.4. Sistemos dinaminis vaizdas

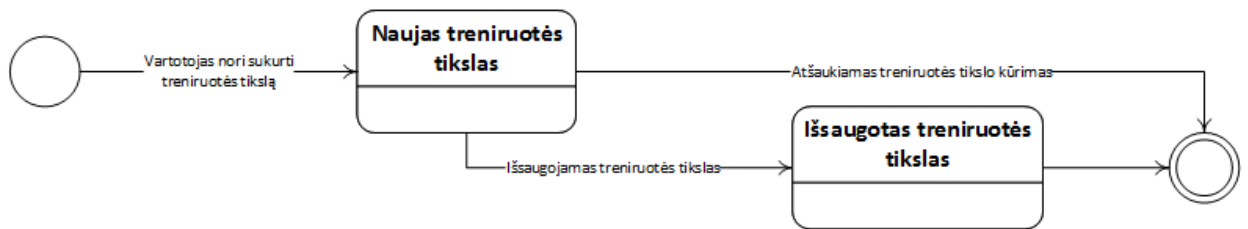
Šiame skyriuje pateikiamos kuriamos sistemos būsenų, veiklų bei sekų diagramos.

### 2.4.1. Būsenų diagramos

Šiame skyriuje pateikiamos svarbiausios sistemos būsenų diagramos pagal panaudos atvejus (PA)

#### 2.4.1.1. PA2 Treniruotės tikslo kūrimas

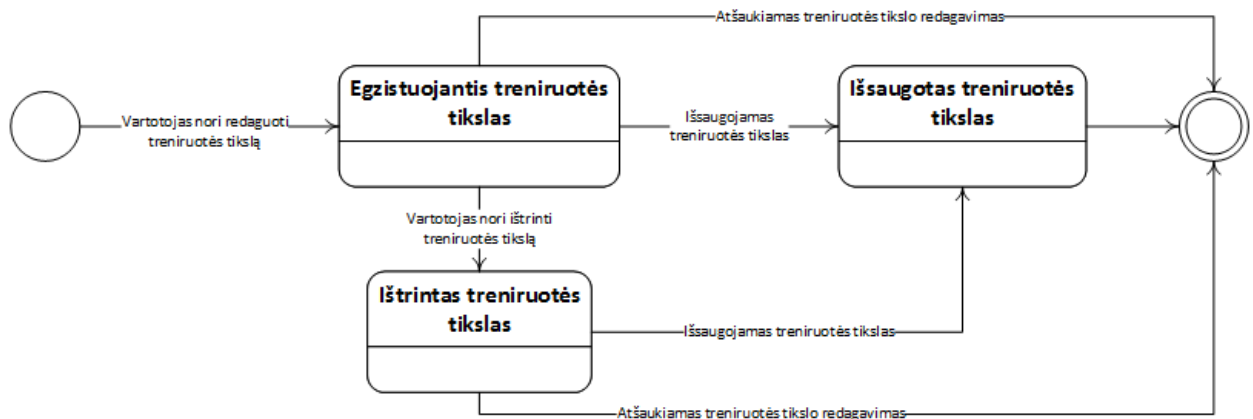
Šioje diagramoje pavaizduotos treniruotės tikslo būsenos, kai vartotojas nori sukurti naują treniruotės tikslą.



16 pav. Būsenos diagrama „PA2 Treniruotės tikslo kūrimas“

#### 2.4.1.2. PA3 Treniruotės tikslo redagavimas

Šioje diagramoje pavaizduotos treniruočių tikslų būsenos, kai vartotojas redaguoja treniruočių tikslus.



17 pav. Būsenos diagrama „PA3 Treniruotės tikslo redagavimas“

#### 2.4.1.3. PA4 Automatinis sportinės veiklos atpažinimas

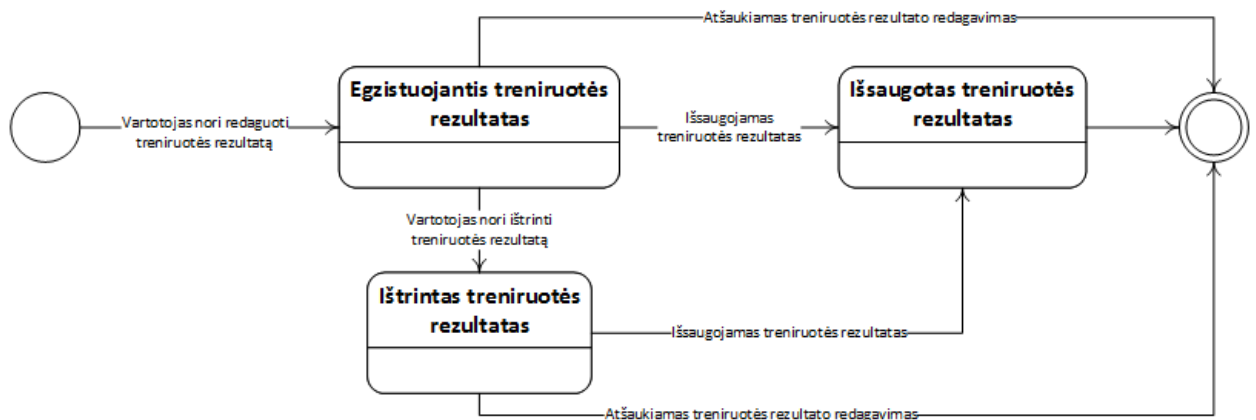
Šioje diagramoje pavaizduotos programinės įrangos būsenos kai vartotojas atlieką automatinį sportinės veiklos atpažinimą.



18 pav. Būsenos diagrama „PA4 Automatinis sportinės veiklos atpažinimas“

#### 2.4.1.4. PA6 Redaguoti rezultatus

Šioje diagramoje pavaizduotos treniruočių rezultatų būsenos, kai vartotojas nori redaguoti treniruočių rezultatus.



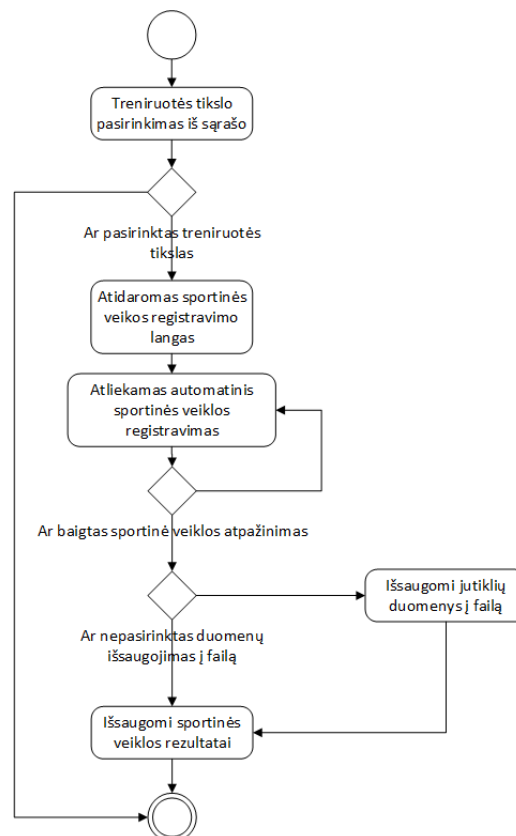
19 pav. Būsenos diagrama „PA6 Redaguoti rezultatus“

## 2.4.2. Veiklos diagramos

Šiame skyriuje pateikiamos svarbiausios sistemos veiklų diagramos pagal panaudos atvejus (PA)

### 2.4.2.1. PA4 Veiklos diagrama

Šioje diagramoje pavaizduotas automatinis sportinės veiklos atpažinimas. Matoma sistemos veikla atliekant automatinį sportinės veiklos atpažinimą.



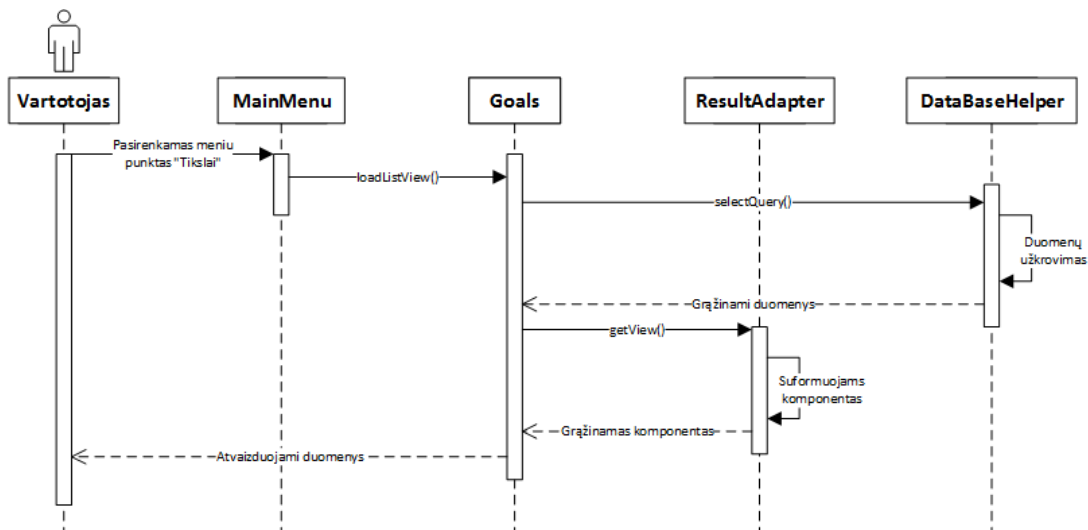
20 pav. Veiklos diagrama „PA4 Automatinis sportinės veiklos atpažinimas“

### 2.4.3. Sekų diagramos

Žemiau pateiktos sekų diagramos pagal panaudos atvejus (PA).

#### 2.4.3.1. PA1 Peržiūrėti treniruočių tikslus

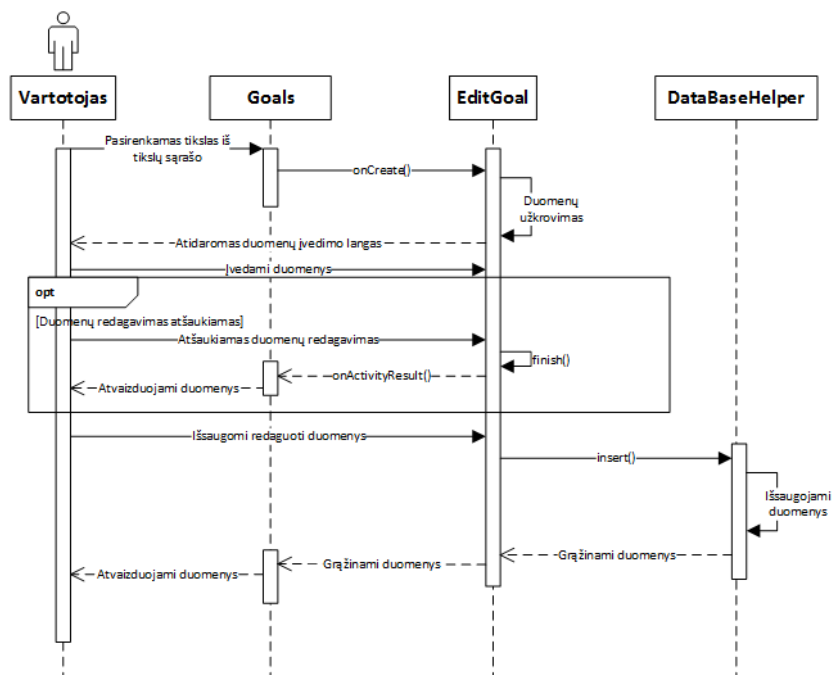
Šioje diagramoje pavaizduota veiksmų seka vartotojui norint peržiūrėti sportinės veiklos tikslus.



21 pav. Sekų diagrama „PA1 Peržiūrėti treniruočių tikslus“

#### 2.4.3.2. PA2 Treniruotės tikslo kūrimas

Šioje diagramoje pavaizduota veiksmų seka vartotojui norint sukurti naują treniruočių tikslą.

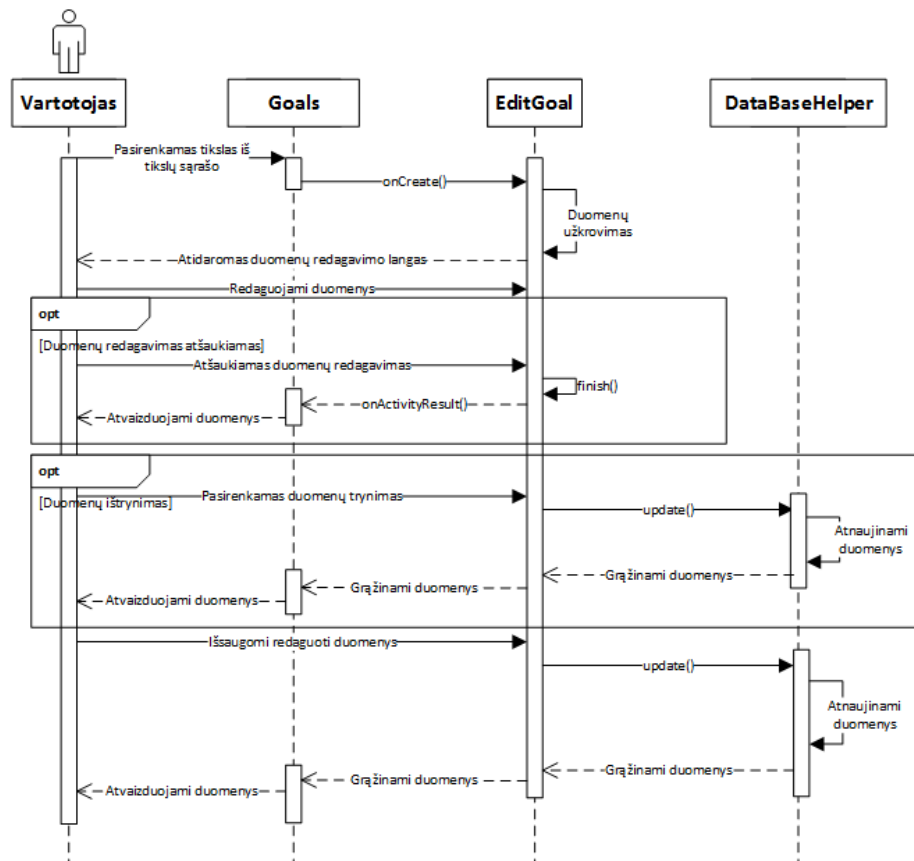


22 pav. Sekų diagrama „PA2 Treniruotės tikslo kūrimas“



### 2.4.3.3. PA3 Treniruotės tikslo redagavimas

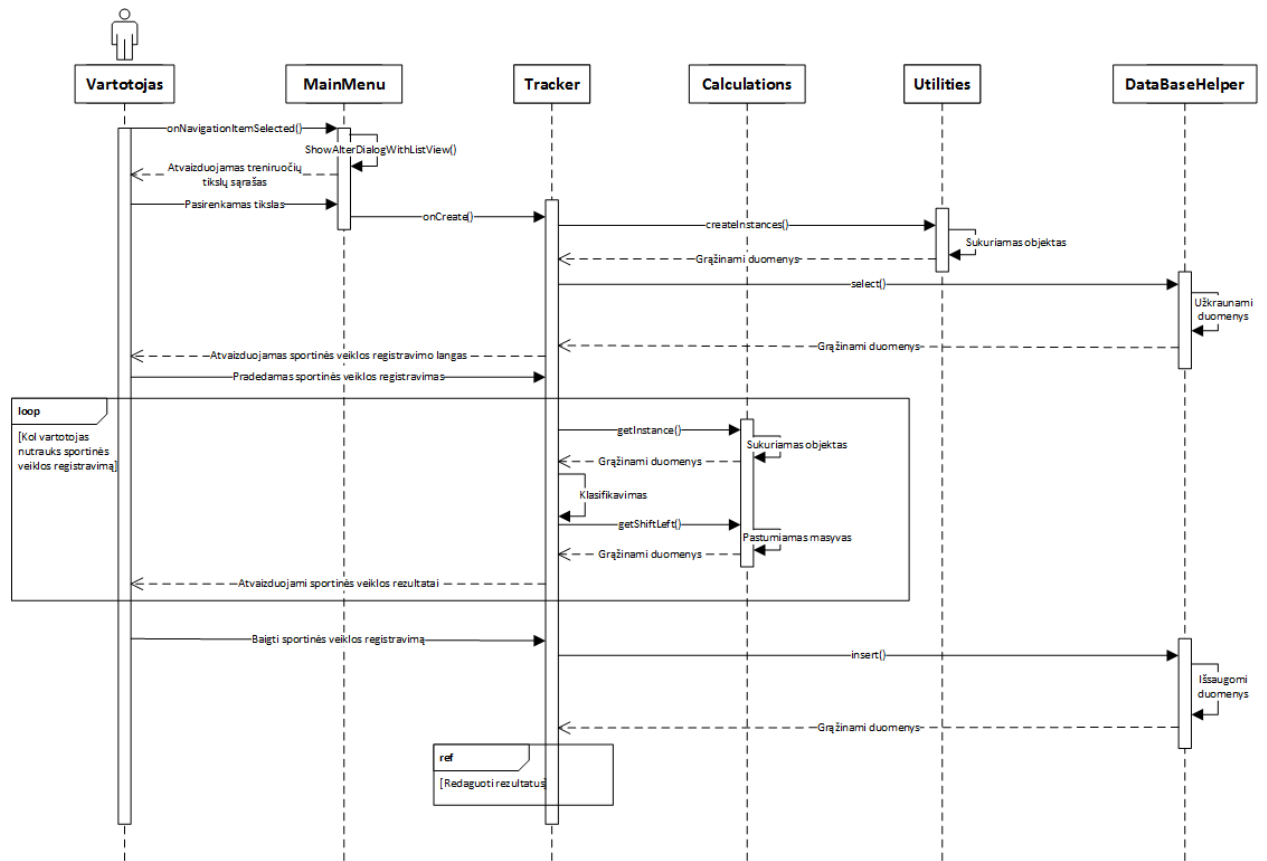
Šioje diagramoje pavaizduota veiksmų seka vartotojui norint redaguoti arba ištrinti sportinės veiklos tikslą.



23 pav. Sekų diagrama „PA3 Treniruotės tikslo redagavimas“

### 2.4.3.4. PA4 Automatinis sportinės veiklos atpažinimas

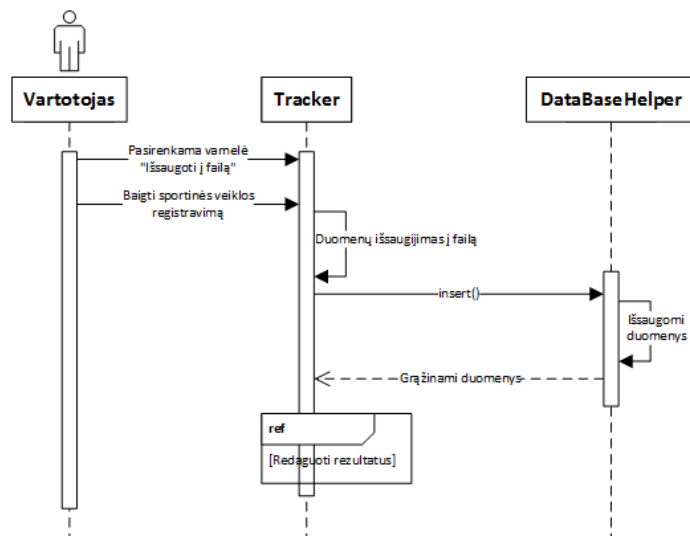
Šioje diagramoje pavaizduota veiksmų seka vartotojui norint atlikti automatinę sportinės veiklos atpažinimą.



24 pav. Sekų diagrama „PA4 Automatinis sportinės veiklos atpažinimas“

### 2.4.3.5. PA5 Duomenų registravimas

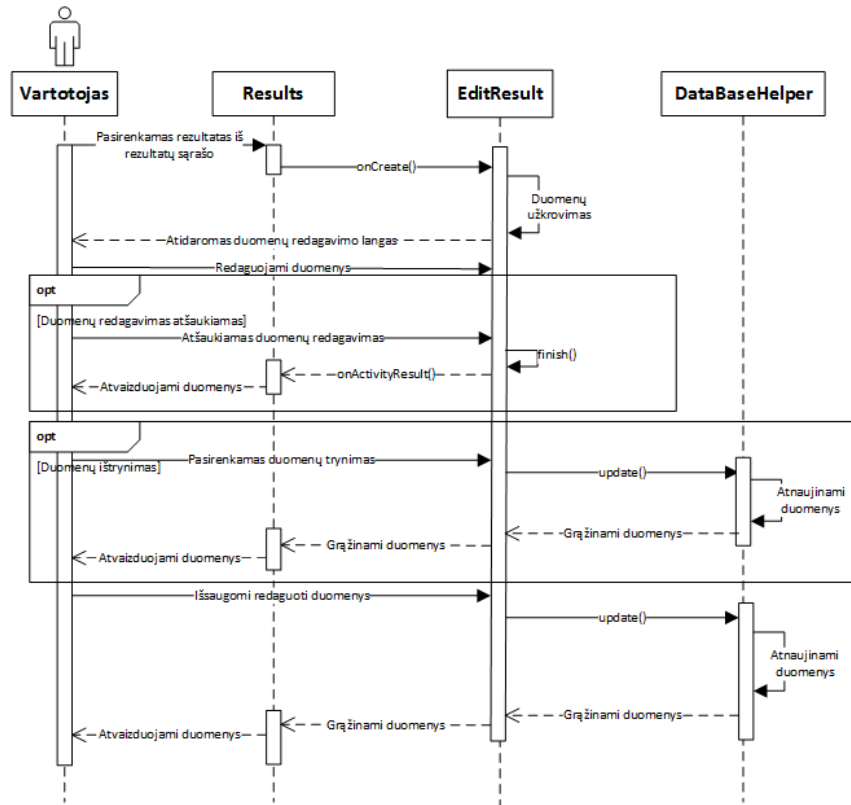
Šioje diagramoje pavaizduota veiksmų seka vartotojui norint išsaugoti sportinės veiklos jutiklių duomenis į išorinį failą.



25 pav. Sekų diagrama „PA5 Duomenų registravimas“

### 2.4.3.6. PA6 Redaguoti treniruotės rezultatus

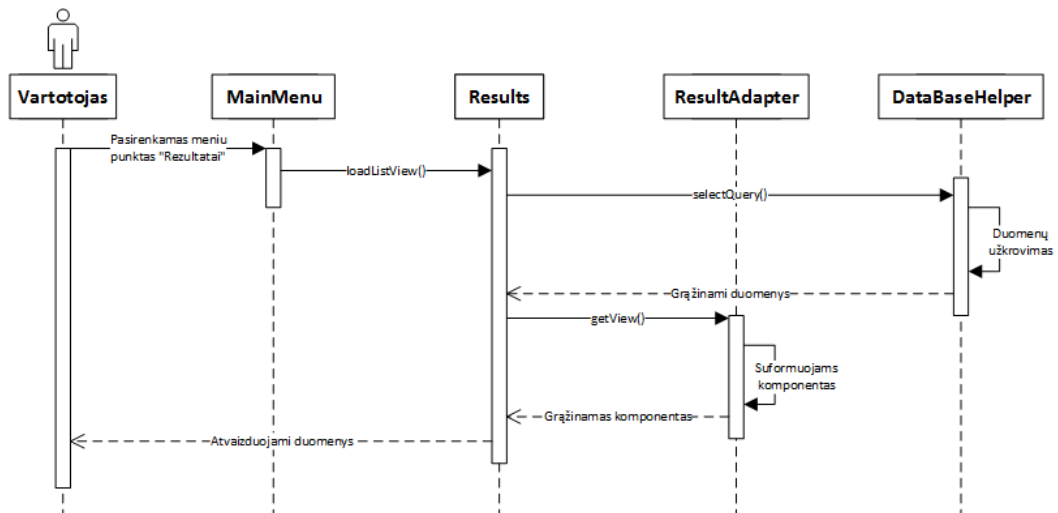
Šioje diagramoje pavaizduota veiklų seka vartotojui norint redaguoti arba ištrinti sportinės veiklos rezultatų duomenis.



26 pav. Sekų diagrama „PA6 Redaguoti treniruotės rezultatus“

### 2.4.3.7. PA7 Peržiūrėti treniruočių rezultatus

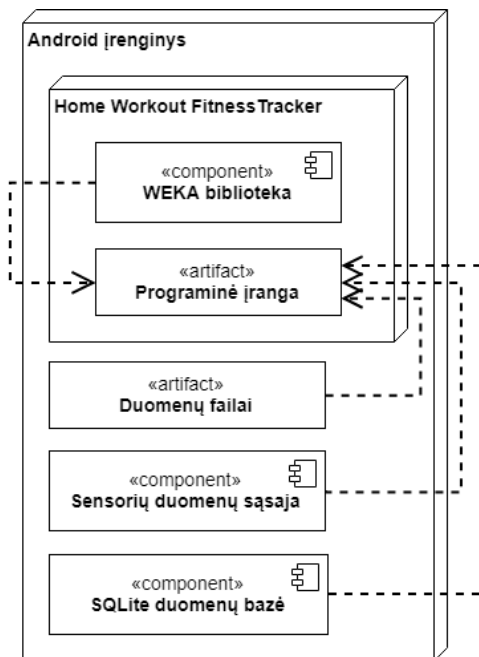
Šioje diagramoje pavaizduota veiksmų seka vartotojui norint peržiūrėti sportinės veiklos rezultatus.



27 pav. Sekų diagrama „PA7 Peržiūrėti treniruočių rezultatus“

## 2.5. Sistemos išdėstymo vaizdas

Paveiksle (28 pav.) pavaizduota projektuojamos sistemos (Home Workout Fitness Tracker) veiklos taisyklių išdėstymo diagrama. Sistemos numatoma eksploatacijos vieta yra išmanusis telefonas su Android operacine sistema (Android įrenginys).



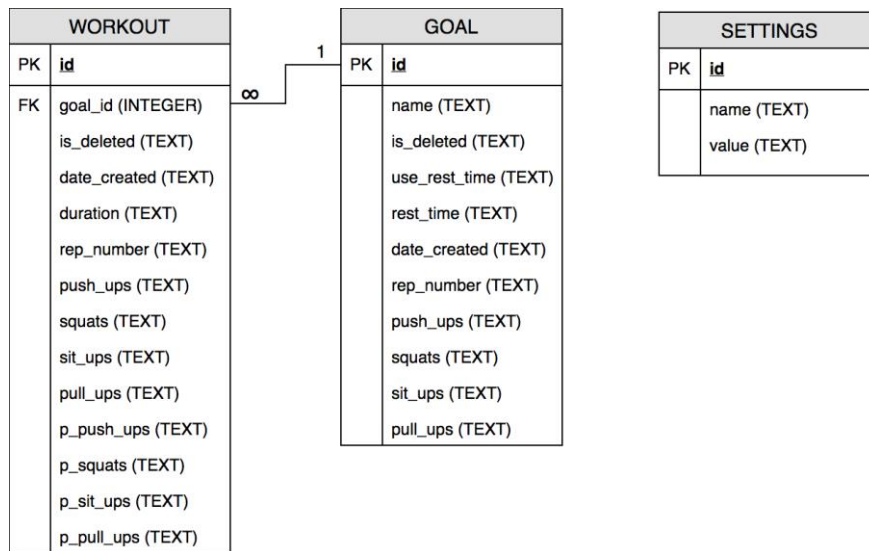
28 pav. Sistemos išdėstymo diagrama

## 2.6. Duomenų vaizdas

Projektuojamai sistemai bus reikalinga duomenų bazė sportinės veiklos, treniruočių tikslų, bei sisteminiams nustatymams saugoti. Taip pat duomenų bazę naudosime magistrinio darbo tyrimui.

Duomenų bazės prototipas pavaizduotas paveiksle (29 pav.), o lentelių paskirtis aprašyta žemiau:

- WORKOUT – Duomenų bazės lentelė skirta treniruočių rezultatų duomenims saugoti.
- GOAL – Duomenų bazės lentelė skirta treniruočių tikslų duomenims saugoti.
- SETTINGS – Duomenų bazės lentelė skirta sistemos konfigūracijų duomenims saugoti.



29 pav. Duomenų bazės modelis

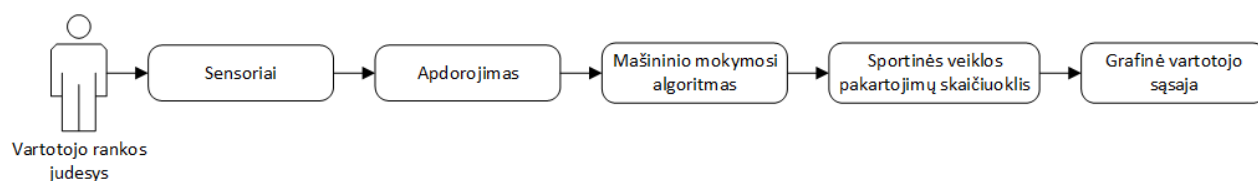
### 3. Tyrimo dalis

#### 3.1. Tyrimo tikslas

Tyrimo tikslas yra nustatyti koks mašininio mokymosi algoritmas yra geriausias sportinių veiklų atpažinimui (atpažįstamos sportinės veiklos: atsispaudimai, pritūpimai ir atsilenkimai), pririnkti duomenų mašininio mokymosi algoritmo apmokymui ir testavimui. Išsiaiškinti kokie duomenų apdorojimo būdai yra tinkamiausi duomenų paruošimui. Išbandyti mašininio mokymosi algoritmą su sukurta programine įranga. Surasti programinės įrangos problemines sritis, jas ištaisyti ir atlikti programinės įrangos patobulinimus.

#### 3.2. Tyrimo aprašymas

Tyrimo tikslo įgyvendinimui turime žinoti kaip veikia sportinės veiklos atpažinimas, jos veikimo algoritmas pavaizduotas paveiksle (pav. 30). Kur vartotojo rankos judesys (rankos judesys atliekant sportinę veiklą) yra registruojamas išmaniojo telefono jutiklių, surinkti duomenys yra saugomi įrenginio vidinėje atmintyje. Vėliau šie duomenys yra apdorojami ir išskaičiuojamos esminės duomenų savybės, kurios gali būti naudojamos mašininio mokymosi algoritmui apmokyti. Mašininio mokymosi algoritmas atlieka sportinės veiklos atpažinimą, surinkus tam tikrą atpažinimų skaičių galime pradėti skaičiuoti sportinės veiklos pakartojimus. Kiekvienas sportinės veiklos skaičiuoklio suskaičiuotas pakartojimas yra priskiriamas prie sportinės veiklos tipui ir atvaizduojamas grafiškai išmaniojo telefono ekrane.



30 pav. Programinės įrangos veikimas algoritmas

Tyrimo metu yra apžvelgiami ankščiau paminėti algoritmo žingsnius, kiekviename iš jų nustatomas ir išanalizuojamas geriausias būdas jam įgyvendinti.

#### 3.3. Tyrimo eiga

##### 3.3.1. Duomenų rinkimas

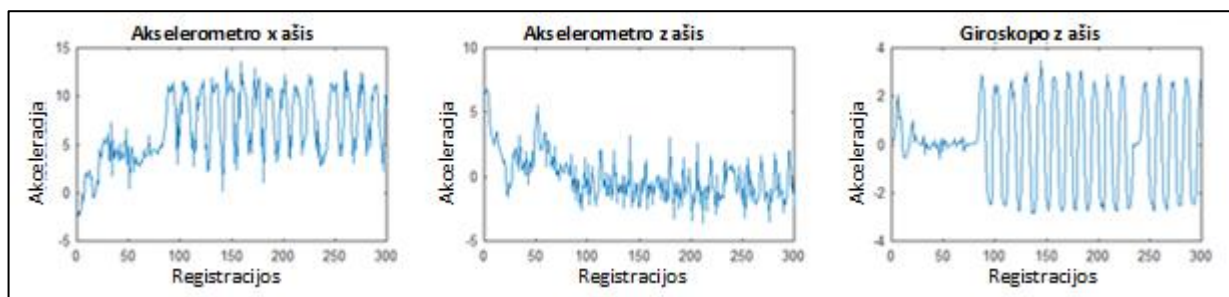
Prieš pradėdant registruoti sportinės veiklos duomenis, yra svarbu nustatyti tinkamiausią išmaniojo telefono vietą ant žmogaus kūno. Išmaniojo telefono vieta turi būti taip parinkta [14], kad būtų galima registruoti sportines veiklas (atsispaudimai, pritūpimai ir atsilenkimai) vienu metu. Po keleto eksperimentų nustatėme, kad jautriausia ir geriausia vieta registruoti sportinėms veikloms yra

kairiosios rankos žastas (viršutinė rankos dalis tarp peties ir alkūnės). Kairė ranka buvo pasirinkta dėl galimybės valdyti išmanųjį telefoną su dešine ranka (nuo 75% iki 90% žmonių populiacijos yra dešiniarankiai). Išmanusis telefonas buvo pritvirtintas prie rankos ekranu į išorę naudojant lipnią juostą, dėl stipresnio tvirtinimo. Testavimo duomenims registruoti buvo parašyta programėlė, kuri registravo jutiklių (akselerometro, giroskopo ir magnetometro) užfiksuotus duomenis. Kiekvienas iš jutiklių buvo trijų dimensių ir galėjo matuoti tris ašis (x, y, z). Susumavus visus duomenis buvo gauti 9 skirtingi jutiklių signalai (akselerometro x, y ir z, giroskopo x, y ir z, magnetometro x, y ir z), kurie buvo registruojami 10Hz (10 kartų per sekundę) dažniu. Kiekvienas išmaniojo telefono jutiklių užfiksuotų duomenų nuskaitymas buvo išsaugojamas masyvo sąrašė, prie kurio dar pridėdavome nuskaitymo laiką (*angl. timestamp*). Sportinės veiklos registravimo pabaigoje, programėlė sustabdoma, o duomenys išsaugomi išmaniojo telefono vidinėje atmintyje .csv failo formatu. Failas yra išsaugomas kompiuteriui pasiekiamoje vietoje, kad vėliau būtų galima atlikti duomenų analizę.

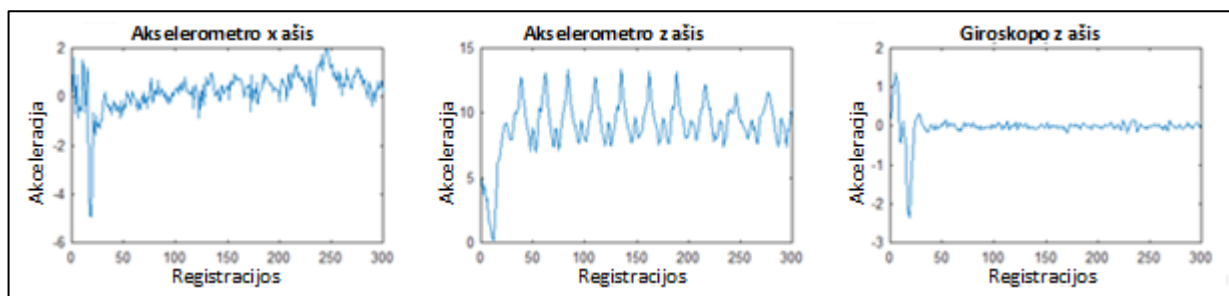
### **3.3.2. Duomenų savybių išgryninimas**

Prieš tai minėtame etape buvo surinkti testavimo duomenys. Šiame etape iš testavimo duomenų išskaičiuosime svarbiausias savybes, sudarysime duomenų rinkinį ir jį panaudosime mašininio mokymosi algoritmo apmokymui.

Atliekant duomenų analizę buvo pastebėta, kad kai kurių jutiklių signalų duomenų negalima naudoti duomenų rinkiniui sudaryti. Problema atsiradavo kai tarp skirtingų sportinių veiklų, kai kurie jutiklių signalai nevienodai registruodavo duomenis, vienoje sportinėje veikloje labiau registruodavo vienos jutiklių ašys, kitoje kitos. Buvo užfiksuoti ir duomenų dublikatai, kai dubliuodavosi keli signalai, pavyzdžiui: akselerometro y ir z ašys dubliuodavosi darant atsispaudimus. Po daugybės eksperimentų, problemos sprendimui buvo pasirinkta naudoti tik pačius reikšmingiausias išmaniojo telefono jutiklių signalus, kurie nesidubliavo, rodė rodmenis visose sportinėse veiklose ir atitiko pasikartojantį sportinės veiklos šabloną (kiekvienas sportinės veiklos pakartojimas turi jam būdingą pasikartojimo kreivę). Pasirinkome išmaniojo telefono jutiklių akselerometro x, z ir giroskopo z ašis. Pasirinktų jutiklių signalų duomenys atliekant sportines veiklas pavaizduoti paveiksluose (pav. 31 - 33).



31 pav. Atsispaudimai



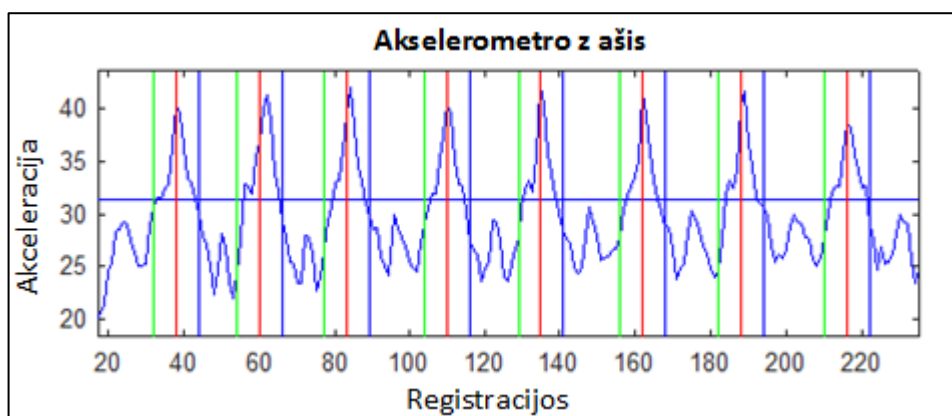
32 pav. Pritūpimai



33 pav. Atsilenkimai

Surinktus sportinių veiklų testavimo duomenis, juos suskirstėm į tris klases kiekvienai sportiniai veiklai. Dauguma surinktų duomenų negalima klasifikuoti kaip sportinės veiklos, iš ankstesnių bandymų žinome, kad klasifikavimui galim naudoti tik signalo dalis kuriuose atliekamas sportinės veiklos pakartojimas, visi kiti signalai yra nulinės klasės (signalų dalys, kurių negalima klasifikuoti kaip sportinės veiklos, tačiau jie puikiai tinka kaip duomenys priešingi sportinėms veikloms). Kad išskirti klases sportinių veiklų duomenų failuose – parašėme klasių išskyrimo programą naudodami MATLAB [15]. Programėlės algoritmas veikia kaip filtras, kuris naudodamas specifinius jutiklio signalo parametrus (aukštis, mediana, signalo ilgis) filtravo ir žymėjo klases tiems signalams, kurie tenkino parametrus. Signalų žymėjimas pavaizduotas paveiksle (pav. 34). Kur vertikali žalia linija žymi sportinės veiklos pakartojimo pradžia, raudona – vidurį, o mėlyna – pabaigą.





34 pav. Išfiltruoto signalo pavyzdys

Atliekant duomenų filtravimo, klasių priskyrimo ir mašininio mokymosi algoritmo apmokymo eksperimentus, pasirinkome naudoti du fiksuoto ilgio sportinės veiklos signalus. Vieną 13 ir kitą 25 duomenų registracijų ilgio. Kiekvienas registravimas atliekamas kas 0,1 sekundę. Mašininio mokymosi algoritmo apmokymo duomenų rinkinys sudaromas naudojant slenkančio lango metodą (*angl. Sliding Window*) [16]. Ankstesniuose bandymuose buvo naudojamas vienas duomenų rinkinys, kurio vieną eilutę sudarė 10 signalo duomenų registracijų (viena registracija sudaryta iš akselerometro x, z ir giroskopo z jutiklių ašių duomenų). Tačiau kai sportinės veiklos pakartojimas būdavo atliekamas ilgiau nei įprastai (ilgiau nei 1,5 sekundės), jo duomenys nebetilpdavo į vieną duomenų rinkinio eilutę (ankščiau naudotą 10 duomenų registracijų kas ekvivalentu 1 sekundės signalui). Iš to nuspręsta naudoti du skirtingo eilutės ilgio duomenų rinkinius – 10 ir 20 sportinės veiklos signalo duomenų registracijų. Remiantis ankstesniais teiginiais 10 registracijų duomenų rinkinys vienoje eilutėje talpins 10 akselerometro x, z ir giroskopo z jutiklių ašių registracijų, o 20 registracijų duomenų rinkinys vienoje eilutėje talpins 20 tokių porų. Slenkančio lango metodas leis sudaryti duomenų rinkinį, kuris talpins 1 sekundės arba 2 sekundžių trukmės sportinės veiklos signalus. Toks būdas leis atpažinti sportinių veiklų pakartojimus kurie trukmė vyraus nuo 0,5 sekundės iki 3 sekundžių.

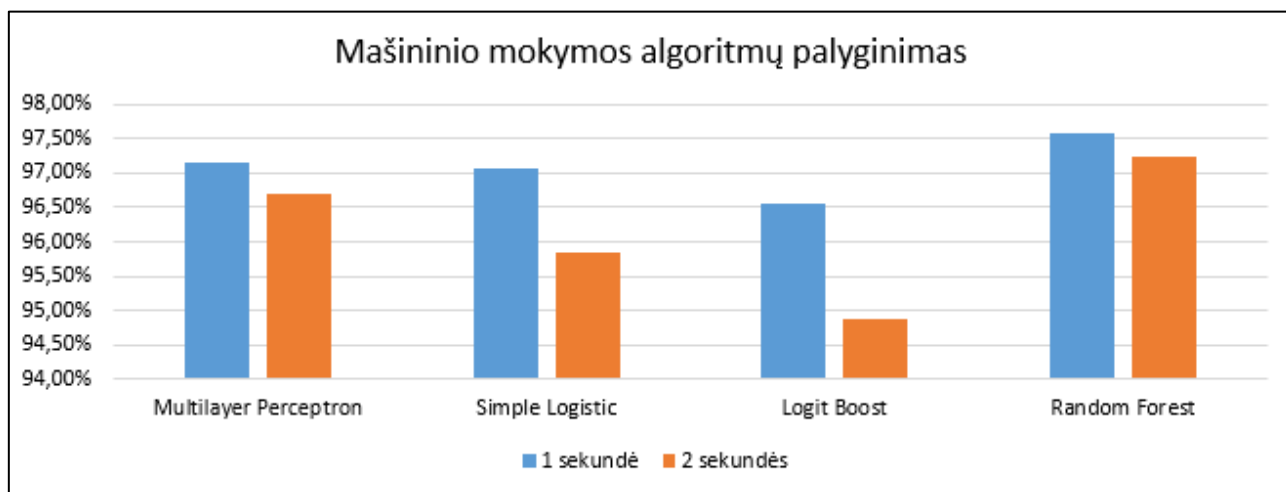
### 3.3.3. Mašininio mokymosi algoritmų palyginimas

Mes pasirinkome, išbandėme ir įvertinome mašininio mokymosi algoritmus realizuotus WEKA duomenų analizavimo įrankyje. Detalesnei analizei ir mašininio mokymosi algoritmų palyginimui pasirinkome: „Multilayer Perceptron“, „Simple Logistic“, „Logit Boost“ ir „Random Forest“ mašininio mokymosi algoritmus. Jie parodė didžiausią atpažinimo tikslumą ankstesniuose bandymuose. Visi algoritmai buvo apmokyti su sportinių veiklų duomenų rinkiniais (1 ir 2 sekundžių), o algoritmo vertinimas atliktas naudojant 10 kartų kryžminį patikrinimą (*angl. 10-fold cross validation*) [45]. Visiems mašininio mokymosi algoritmams buvo parinkti numatytieji (*angl. default*) nustatymai.

Mašininio mokymosi algoritmų palyginimo rezultatai atpažįstant sportines veiklas pateikti lentelėje (4 lentelė) ir pavaizduoti paveiksle (35 pav.). Geriausią sportinės veiklos atpažinimo tikslumą parodė „Random Forest“ (1s – 97,582% ir 2s – 97,252%) ir „Multilayer Perceptron“ (1s – 97,144% ir 2s – 96,7%). Lyginant „Random Forest“ ir „Multilayer Perceptron“, pirmasis pirmauja netik atpažinimo tikslumu, tačiau ir mašininio mokymosi algoritmo apmokymo laiko sąnaudomis (2 minutės lyginant su 15 minučių apmokant su 2s duomenų rinkiniu), kurios daug trumpesnės nei „Multilayer Perceptron“. Likę algoritmai: „Simple Logistic“ ir „Logit Boost“ parodė mažesnę sportinės veiklos atpažinimo tikslumą, tačiau mašininio mokymosi algoritmo apmokymo laikas buvo trumpesnis už „Multilayer Perceptron“.

4 lentelė. Mašininio mokymosi algoritmų atpažinimo palyginimas

Mašininio mokymosi algoritmas	Atpažinimo tikslumas, %	
	1 sekundė	2 sekundės
Multilayer Perceptron	97.144	96.7
Simple Logistic	97.06	95.837
Logit Boost	96.56	94.885
Random Forest	97.582	97.252

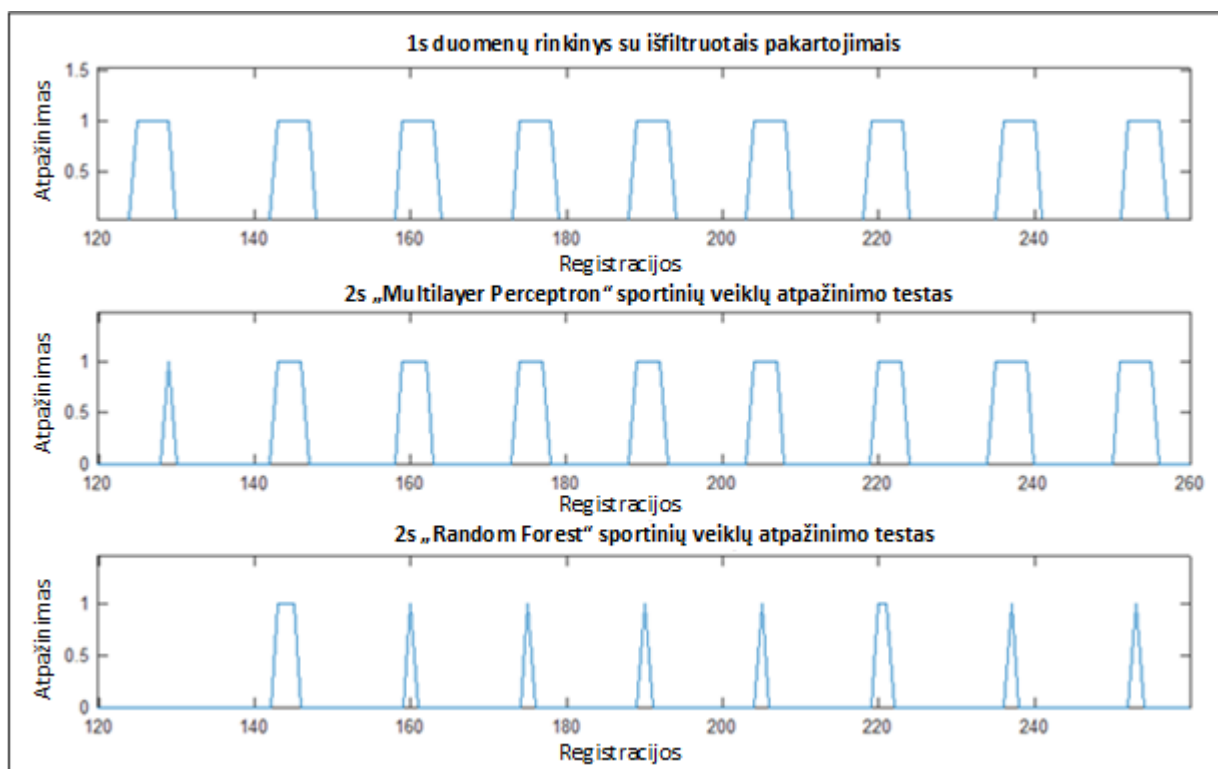


35 pav. Mašininio mokymosi algoritmų atpažinimo palyginimas

### 3.3.4. Sportinės veiklos pakartojimų skaičiavimas

Pagal lentelės (4 lentelė) duomenis, tinkamiausias kandidatas mašininio mokymosi algoritmui yra „Random Forest“, tačiau, buvo atlikus dar vienas bandymas, kurio tikslas išsiaiškinti kaip skiriasi išfiltruotas (filtravimo programėlę parašyta su MATLAB programinę įrangą) sportinės veiklos signalas nuo atpažinto, naudojant mašininio mokymosi algoritmą. Tyrimo duomenys pateikti paveiksle (pav. 36), kur matomas „Multilayer Perceptron“ pranašumas prieš „Random Forest“ realaus pasaulio sąlygomis. „Multilayer Perceptron“ beveik identiškai atkartotojo išfiltruotą sportinės veiklos

signalą. Tyrimas buvo atliekamas naudojant 1s sportinės veiklos duomenų rinkinį, kurį analizavo mašininio mokymosi algoritmai apmokyti su 2s sportinės veiklos duomenų rinkiniu.



36 pav. „Random Forest“ ir „Multilayer Perceptron“ sportinės veiklos atpažinimo palyginimas

Sportinės veiklos pakartojimų skaičiavimas remiasi mašininio mokymosi algoritmo gražinamų atpažinimų rezultatu ir kiek teisingų atpažinimų pavyko atlikti be pertrūkių tarp vienos sportinės veiklos pakartojimo. Kaip matome paveiksle (pav. 36), mums naudingas yra ilgesnis atpažinimo signalas, kuris tęsiasi 3 ir daugiau registracijų be trukdžių. Sportinės veiklos pakartojimų skaičiavimo algoritmas užfiksavęs 3 ir daugiau iš eilės besitęsiančius tos pačios sportinės veiklos atpažinimus fiksuoja sportinės veiklos pakartojimą. Toks algoritmas savo struktūra leidžia išfiltruoti triukšmą ir teisingai fiksuoti sportinės veiklos pakartojimą, pavyzdžiui: fiksuojami 3 iš eilės sportinės veiklos atpažinimai, o tada fiksuojamas triukšmas arba kita sportinė veikla, tačiau algoritmas atlieka dar 2 atpažinimus, kad įsitikintų, jog sportinės veiklos pakartojimas tikrai baigtas. Jei buvo atpažinti 2 triukšmo atpažinimai, o tada toliau fiksuojama ta pati sportinė veikla, algoritmas nefiksuoja dviejų pakartojimų, tačiau gražina tik vieno pakartojimo fiksavimą.

Programinėje įrangoje yra realizuoti du mašininio mokymosi algoritmai: vienas 10 registracijų, kitas 20 registracijų sportinių veiklų pakartojimas fiksuoti. Jie abu veikia lygiagrečiai, sumoje yra gaunami du atpažinimo signalai kurie sujungiami į vieną. Galutiniame rezultate sportinių veiklų pakartojimų skaičiuoklis naudoja abiejų mašininio mokymosi algoritmų sujungtą atpažinimų signalą.

### 3.4. Tyrimo išvados

Tyrimo metu buvo surinkti ir išanalizuoti sportinių veiklų duomenys, nustatyta, kad sportinių veiklų duomenų fiksavimui labiausiai tinkami yra akselerometro x, z ir magnetometro z ašių jutiklių duomenys.

Sportinių veiklų duomenis suskirstyti į dvi grupes pagal sportinės veiklos pakartojimo atlikimo trukmę: trumpos sportinės veiklos (sportinės veiklos pakartojimo vidutinė trukmė nuo 0,5s iki 1,5s) ir ilgos sportinės veiklos (sportinės veiklos pakartojimo vidutinė trukmė nuo 1,5s iki 2,5 sekundės). Tokiu būdu galima naudoti du, lygiagrečiai veikiančius mašininio mokymosi algoritmus trumpiems ir ilgiems pakartojimams atpažinti, kad padidinti sportinių veiklų atpažinimo tikslumą.

Palyginus ir įvertinus mašininio mokymosi algoritmus (lentelė 4) sportinių veiklų atpažinimui, pasirinktas „Multilayer Perceptron“ mašininio mokymosi algoritmas, kuris parodė geriausią atpažinimo tikslumą ir lankstumą.

Apjungus 1 sekundės ir 2 sekundžių mašininio mokymosi algoritmų išėjimo signalus realizuotas sportinių veiklų ir jų pakartojimų skaičiuoklio algoritmas.

## 4. Eksperimento dalis

### 4.1. Eksperimento tikslas

Eksperimento pagrindinis tikslas yra išbandyti sukurtą programinės įrangos prototipą realių sportinių veiklų atpažinimui ir pakartojimų skaičiavimui. Apskaičiuoti sportinių veiklų atpažinimo tikslumą.

### 4.2. Eksperimento aprašymas

Eksperimento metu bus testuojamas sukurtas sportinių veiklų atpažinimo programinės įrangos prototipas realiomis lauko sąlygomis. Eksperimentui atlikti naudosime išmanųjį telefoną kuriame bus įdiegta sportinės veiklos atpažinimo programinė įranga. Išmanusis telefonas bus pritvirtintas prie eksperimentą atliekančio žmogaus kairiojo žasto. Išmaniojo telefono vieta ant žasto bus sukalinama taip, kad jo ekranas žiūrėtų horizontaliai į viršų, kai žmogus atsistoja į pritūpimų padėtį, ištiesia rankas į priekį, o jo delnai žiūri į apačią. Eksperimento užduotis yra:

- Atlikti sportines veiklas kaip atsispaudimų, pritūpimų ir atsilenkimų darymas;
- Kiekvieną sportinę veiklą pakartojant po 10 kartų lėtu (nuo 0,5s iki 1s pakartojimo atlikimo laiko) vidutiniu (nuo 1s iki 2s pakartojimo atlikimo laiko) ir greitu (nuo 2s iki 3s pakartojimo atlikimo laiko) tempu.

Eksperimento metu visi užfiksuoti sportinių veiklų duomenys bus suvedami į eksperimento duomenų lentelę (5 lentelė). Prieš pradėdant eksperimentą kiekvienas jį atliekantis asmuo bus trumpai instrukuojamas kaip taisyklingai atlikti sportines veiklas.

### 4.3. Eksperimentas ir jo rezultatai

Eksperimente dalyvavo 9 dalyviai, visi iš jų buvo vyrai, dalyvių amžius vyravo nuo 18 iki 24 metų. Dalyviai buvo paprašyti savo noru atlikti sportines veiklas su ant žasto pritvirtintu išmaniuoju telefonu. Programinėje sportinės veiklos atpažinimo įrangoje naudojant grafinę vartotojo sąsają buvo paleistas automatinis sportinės veiklos atpažinimas, o dalyvis atliko sportinę veiklą. Atlikus 10 tam tikros sportinės veiklos pakartojimų, buvo registruojami programinės įrangos pateikti rezultatai. Visų 9 dalyvių rezultatai atliekant sportines veiklas pateikti lentelėje (5 lentelė).

5 lentelė. Eksperimento rezultatai

Dalyviai	Atsispaudimai			Pritūpimai			Atsilenkimai		
	Greiti	Vidutiniai	Lėti	Greiti	Vidutiniai	Lėti	Greiti	Vidutiniai	Lėti
1.	10	10	10	10	10	10	9	10	10
2.	8	10	10	8	10	10	7	9	10
3.	9	10	10	8	10	10	7	10	10
4.	10	10	9	10	10	10	8	10	10
5.	10	10	10	10	10	10	9	10	10
6.	10	9	8	10	9	9	10	10	10

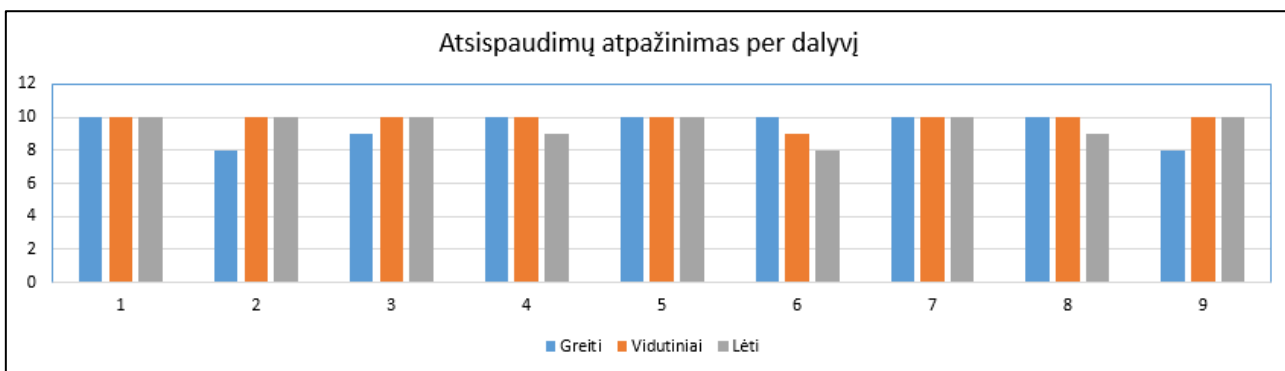
Dalyviai	Atsispaudimai			Pritūpimai			Atsilenkimai		
	Greiti	Vidutiniai	Lėti	Greiti	Vidutiniai	Lėti	Greiti	Vidutiniai	Lėti
7.	10	10	10	10	10	9	10	10	9
8.	10	10	9	10	10	8	10	10	9
9.	8	10	10	9	10	10	8	10	10
<b>Atpažinimo tikslumas, %</b>	94.444	98.889	95.556	94.444	98.889	96.667	86.667	98.889	97.778

Visų trijų sportinių veiklų atpažinimo tikslumas naudojant išmanųjį telefoną pavaizduotas paveiksle (37 pav.). Remiantis pavaizduotos diagramos duomenimis, geriausiai buvo atpažįstamos sportinės veiklos, kurios buvo atliekamos vidutiniu greičiu (nuo 1s iki 2s pakartojimo trukmė), remiantis lentelės (5 lentelė) duomenimis šių veiklų atpažinimo tikslumas siekia 98,889%. Sportinės veiklos kurios buvo atliekamos lėčiau (nuo 2s iki 3s pakartojimo trukmė), buvo atpažįstamos mažesniu tikslumu, kuris vyravo nuo 95,556% iki 97,778%. Greitu temptu (nuo 0,5s iki 1s pakartojimo trukmė) atliekamos sportinės veiklos buvo atpažįstamos mažiausiu tikslumu, kuris vyravo nuo 86,667% iki 94,444%. Prasčiausiai atpažįstama sportinė veikla buvo atsilenkimai atliekami greitu tempu, jų atpažinimo tikslumas siekė viso labo 86,667%. Tokį atpažinimo tikslumą gelėjo lemti per skubėjimą neteisingai atliekamas atsilenkimų pratimas, kai atsilenkimo judesys atliekamas nepilnai arba per greitai. Dirbant su žmonėmis labai sunku sukontroliuoti judesių atlikimo greitį, ypač kai yra reikalaujamas greitas jų atlikimas. 0,5 sekundės laiko tarpas yra labai mažas įvertinti ar teisingai yra atliekamas sportinės veiklos pratimas.

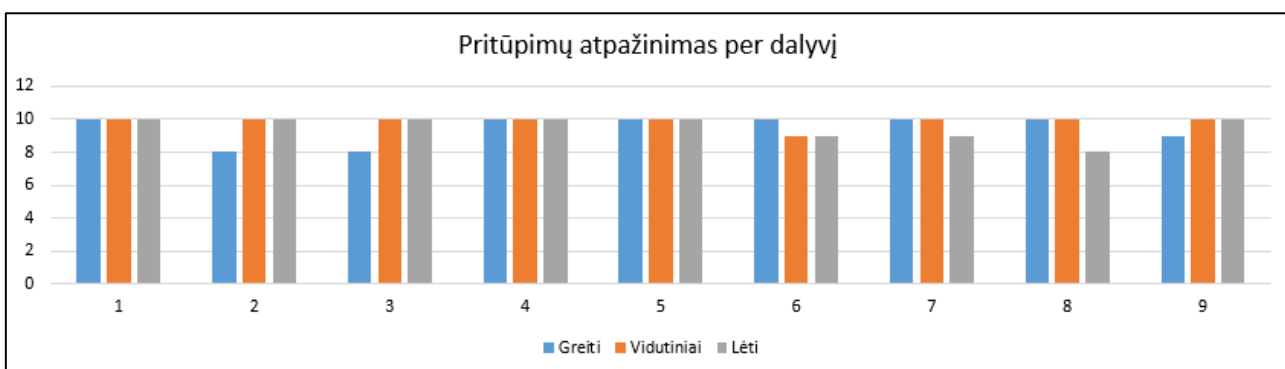


37 pav. Sportinių veiklų atpažinimo tikslumas

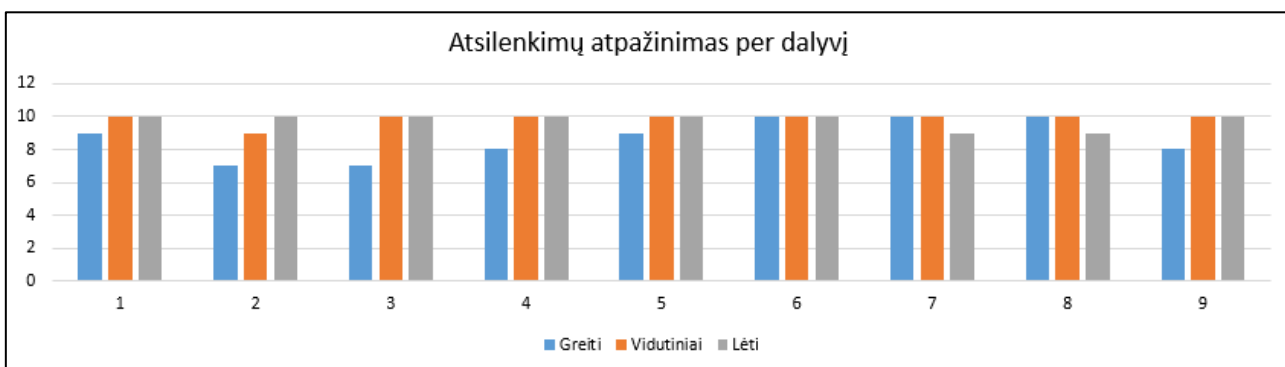
Paveiksluose (38 – 40 pav.) yra pavaizduotos diagramos kiekvienai sportinei veiklai, kur lyginamas sportinių veiklų atlikimo tempas ir atpažinimo rezultatai kiekvienam eksperimento dalyviui. Diagramose X ašis žymi dalyvius, Y ašis žymi pakartojimų skaičių. Kiekvienas dalyvis turėjo atlikti po 10 sportinės veiklos pratimo pakartojimų lėtu, vidutiniu ir greitu tempu. Diagramos stulpelio aukštis nurodo kiek sportinės veiklos pakartojimų buvo atpažinta teisingai.



38 pav. Atsispaudimų atpažinimo rezultatų diagrama



39 pav. Pritūpimų atpažinimo rezultatų diagrama



40 pav. Atsilenkimų atpažinimo rezultatų diagrama

Eksperimento metu buvo pastebėta, kad neužfiksavus sportinės veiklos pakartojimo, dalyviai bandė keisti sportinės veiklos pratimo atlikimo būdą, kol sportinė veikla būdavo atpažįstama teisingai. Dalyviai pradėdavo atlikti tikslesnius sportinės veiklos pakartojimo judesius, pakeisdavo savo laikyseną. Galima teigti, kad sukurta programinė įranga netik atpažįsta ir skaičiuoja sportines veiklas, bet ir gerina vartotojo laikyseną jas atliekant. Vartotojas bando koreguoti savo judesius, mažinti, arba didinti sportinių veiklų atlikimo tempą. Esant gerai apmokytam mašininio mokymosi algoritmui su teisingai atliekamos sportinės veiklos duomenimis, programinė įranga naudodama išmanųjį telefoną gali koreguoti vartotojo sportinės veiklos atlikimo judesius kai jie atliekami neteisingai.

#### **4.4. Eksperimento išvados**

Eksperimento metu buvo atliekamas sportinės veiklos atpažinimo tikslumo vertinimas, sukurtam programinės įrangos prototipui. Eksperimente dalyvavo 9 dalyviai, kurie atliko sportines veiklas pagal eksperimento aprašymą. Eksperimento metu užregistruoti sportinių veiklų atpažinimo rezultatai yra pateikti lentelėje (lentelė 5).

Atlikus eksperimentą buvo įvertinti ir išanalizuoti rezultatai, kur nustatyta, kad programinė įranga geriausiai atpažino vidutiniu tempu (nuo 1s iki 2s pakartojimo trukmė) atliekamas sportines veiklas, o šių veiklų atpažinimo tikslumas siekia 98,889%. Blogiausiai atpažįstami buvo greitai tempu atliekami atsilenkimai, kur buvo pasiektas tik 86,667% sportinės veiklos atpažinimo tikslumas. Didelę įtaką sportinių veiklų atpažinimui darė jų atlikimo greitis ir sportinės veiklos atlikimo būdas, kuris (šiuo atveju lyginamas su greitai atliekamais atsilenkimais) lėmė sportinių veiklų atpažinimo tikslumo mažėjimą.

Sportinės veiklos atpažinimo programėlė daro įtaką vartotojo sportinės veiklos atlikimo būdui, skatinant vartotoją atlikti sportinės veiklos pratimus teisingai.



## Išvados

1. Analitinės dalies metu, sportinių veiklų atpažinimo programėlei realizuoti buvo analizuojamos reikalingos technologijos, priemonės, metodai ir būdai. Apžvelgtos ir palygintos jau esamos mobiliųjų telefonų programėlės su panašiu sistemos funkcionalumu.
2. Atlikus analizę sportinės veiklos atpažinimui pasirinkta naudoti mašininio mokymosi algoritmus, o duomenų surinkimui – jutikliai: akselerometras, giroskopas ir magnetometras.
3. Sistemai suprojektuoti buvo surinkti užsakovo reikalavimai, sudaryti sistemos panaudos atvejai, aprašyti funkciniai ir nefunkciniai reikalavimai, sukurtas ir detalizuotas sistemos paketų planas, panaudos atvejams aprašytos veiklų, būsenų ir sekų diagramos.
4. Tyrimo metu, siekiant sumažinti mobiliojo telefono resursų naudojimą buvo ištirta ir nuspręsta, sportinės veiklos duomenis registruoti 10Hz (10 kartų per sekundę) dažniu ir naudoti pačius reikšmingiausias jutiklių duomenis: akselerometro x, z ašių ir magnetometro z ašies signalus.
5. Naudojant MATLAB programinę įrangą, buvo parašyta programėlė, sportinių veiklų duomenų rinkiniams sudaryti pagal specifinius parametrus. Programėlės tikslas yra išfiltruoti ir priskirti klases sportinių veiklų pakartojimams.
6. Atlikus tyrimą ir palyginus mašininio mokymosi algoritmus, sportinių veiklų atpažinimui pasirinktą naudoti du lygiagrečiai veikiančius „Multilayer Perceptron“ mašininio mokymosi algoritmus (palyginimo rezultatai pateikti lentelėje 4). Vienas apmokytas 0,5 – 1,5 sekundės sportinės veiklos atpažinimui, antras apmokytas 1,5 – 3 sekundžių sportinės veiklos atpažinimui.
7. Suprogramuotas ir realizuotas mobiliosios programėlės prototipas sportinėms veikloms atpažinti ir registruoti. Mobiliosios programėlės prototipo ekrano nuotraukos pavaizduotos 1 priede.
8. Sportinių veiklų atpažinimo tikslumui nustatyti buvo atliktas eksperimentas, kurio metu 9 dalyviai atliko sportines veiklas (rezultatai pateikti lentelėje 5). Eksperimento metu vidutinio tempo (sportinės veiklos pakartojimo atlikimas nuo 1s iki 2s) sportinių veiklų atpažinimo tikslumas siekė 98,9%. Sunkiau buvo atpažįstamos lėto tempo (sportinės veiklos pakartojimo atlikimas nuo 2s iki 3s) sportinės veiklos, jų atpažinimo tikslumas vyravo nuo 95,6% iki 97,8%. Sunkiausiai atpažįstamos buvo greito tempo (sportinės veiklos pakartojimo atlikimas nuo 0,5s iki 1s) sportinės veiklos, kurių atpažinimo tikslumas vyravo nuo 86,7% iki 94,4%. Bendras sportinių veiklų atpažinimo vidurkis visoms veikloms (atliekant greitai, vidutiniškai ir lėtai) siekė 95,8%.
9. Eksperimento metu pastebėta, kad mobilioji programėlė daro įtaką vartotojo sportinės veiklos atlikimo būdai, skatinant vartotoją teisingai atlikti sportinės veiklos pratimus.

## Literatūros sąrašas

1. D. Guan, T. Ma, W. Yuan, Y. Lee and A. M. J. Sarkar, „Review of Sensor-based Activity Recognition Systems“, IETE Technical Review, 28:5, 418-433, 2011.
2. Incel O.D., Kose M., Ersoy C. „A Review and Taxonomy of Activity Recognition on Mobile Phones“, BioNanoScience, 3:145-171, 2013.
3. J. R. Kwapiz, G.M. Weiss and S. A. Moore, „Activity Recognition Using Cell Phone Accelerometers“, ACM SIGKDD Explorations Newsletter, 12:2, 74-82, 2010.
4. A. Bayat, M. Pomplun, D. A. Tran, „A Study on Human Activity Recognition Using Accelerometer Data from Smartphones“, Procedia Computer Science 34, 450-457, 2014.
5. M. Nilsson and H. Wilén, „Push-up Tracking through Smartphone Sensors“, School of Engineering Sciences, 23p, 2016.
6. R. Piyare and S. R. Lee, „Dynamic Activity Recognition using Smartphone Sensor Data“, 1st International Electronic Conference on Sensors and Applications, 2014.
7. S. M. S. Hasan, M. Masnad, M. M. Khan, H. Mahmud and M. K. Hasan, „Human Activity Recognition using Smartphone Sensors with Context Filtering“, The Ninth International Conference on Advances in Computer-Human Interactions, 2016.
8. M. Shoaib, S. Bosch, O. D. Incel, H. Scholten and P. JM Havinga, „Complex Human Activity Recognition Using Smartphone and Wrist-Worn Motion Sensors“, Multidisciplinary Digital Publishing Institute, 16:4, 2016.
9. Weka mašininio mokymos įrankis [žiūrėta 2017-04-18], prieiga per internetą <http://www.cs.waikato.ac.nz/ml/index.html>.
10. Dynamic Time Warping [žiūrėta 2018-02-05], prieiga per internetą : [web.science.mq.edu.au/~cassidy/comp449/html/ch11s02.html](http://web.science.mq.edu.au/~cassidy/comp449/html/ch11s02.html).
11. Android operacinė sistema [žiūrėta 2016-12-10], prieiga per internetą: [https://en.wikipedia.org/wiki/Android\\_\(operating\\_system\)](https://en.wikipedia.org/wiki/Android_(operating_system)).
12. SQLite duomenų bazė [žiūrėta 2018-02-23], prieiga per internetą: <https://www.sqlite.org/about.html>.
13. N. D. Lane, E. Miluzzo, H. Lu, D. Peebles, T. Choudhury, and A. T. Campbell, „A survey of mobile phone sensing“, IEEE Communications Magazine, vol. 48, no. 9, pp. 140–150, 2010.

14. W. Xu, M. Zhang, A. A. Sawchuk, and M. Sarrafzadeh, „Co-recognition of human activity and sensor location via compressed sensing in wearable body sensor networks“, in Proc. 9th IEEE Int. Conf. Wearable Implantable Body Sensor Netw., 2012, pp. 124–129.
15. MATLAB [žiūrėta 2017-10-15], prieiga per internetą: <https://se.mathworks.com/products/matlab.html>.
16. Selenkančio lango metodas (*angl. Sliding Window*) [žiūrėta 2017-10-15], prieiga per internetą: [http://wcipeg.com/wiki/Sliding\\_window](http://wcipeg.com/wiki/Sliding_window).
17. Akselerometras [žiūrėta 2016-12-10], prieiga per internetą: <http://www.sensorwiki.org/doku.php/sensors/accelerometer>.
18. Giroskopas [žiūrėta 2016-12-10], prieiga per internetą: <http://www.sensorwiki.org/doku.php/sensors/gyroscope>.
19. X. Su, P. Ji, H. Tong, „Activity Recognition with SmarPhone Sensors“, vol. 19, no. 3, pp. 235–249, 2014. DOI: 10.1109/TST.2014.6838194.
20. A. Rasekh, C. Chen, Y. Lu, „Human Activity Recognition using Smartphone“, Computers and Society, 2011.
21. Mašininis mokymasis [žiūrėta 2016-12-10], prieiga per internetą: [https://en.wikipedia.org/wiki/Machine\\_learning](https://en.wikipedia.org/wiki/Machine_learning).
22. J. Figura, „Machine Learning for Google Android“, 2012.
23. Runtastic Sit-Ups Workout [žiūrėta 2016-12-10], prieiga per internetą: <https://play.google.com/store/apps/details?id=com.runtastic.android.situp.lite>.
24. Squats Workout programėlės aprašymas [žiūrėta 2016-11-25], prieiga per internetą: <https://play.google.com/store/apps/details?id=com.northpark.squats>.
25. Push-Ups Workout programėlės aprašymas [žiūrėta 2016-11-25], prieiga per internetą: <https://play.google.com/store/apps/details?id=com.northpark.pushups>.
26. Pull-Ups Workout programėlės aprašymas [žiūrėta], prieiga per internetą: <https://play.google.com/store/apps/details?id=com.northpark.pullups&hl=lt>.
27. S Health programėlės aprašymas [žiūrėta 2016-11-25], prieiga per internetą: <https://www.discovery.co.za/portal/individual/vit-partner-samsung-s-health>.
28. Magnetometras [žiūrėta 2016-12-10], Prieiga per internetą: <https://en.wikipedia.org/wiki/Magnetometer>.

29. Jutiklių sintezės paaiškinimas [žiūrėta 2016-22-20], prieiga per internetą: <https://www.youtube.com/watch?v=C7JQ7Rpwn2k>.
30. J. N. Gross; Y. Gu, M. B. Rhudy, S. Guruajan, M. R. Napolitano, „Flight-Test Evaluation of Sensor Fusion Algorithms for Attitude Estimation“, IEEE Transactions on Aerospace and Electronic Systems, vol. 48, no., pp. 2128 – 2139, 2012. DOI: 10.1109/TAES.2012.6237583.
31. H. Wu, M. Siegel, R. Stiefelwagen, J. Yang, „Sensor fusion using Dempster-Shafer theory“, Proceedings of the 19th IEEE Instrumentation and Measurement Technology Conference, vol. 1, no., pp. 7-12, 2002. DOI: 10.1109/IMTC.2002.1006807.
32. Ramsey Faraghe, „Understanding the Basis of the Kalman Filter Via a Simple and Intuitive Derivation“, System Advanced Technology Center, 2012.
33. Bayeso teorema [žiūrėta 2016-12-10], prieiga per internetą: <http://www.trinity.edu/cbrown/bayesweb/>.
34. Dirbtinis neuroninis tinklas [žiūrėta 2016-12-10], prieiga per internetą: [https://en.wikipedia.org/wiki/Artificial\\_neural\\_network](https://en.wikipedia.org/wiki/Artificial_neural_network).
35. Atraminių vektorių metodas (SVM) [žiūrėta 2016-12-02], prieiga per internetą: [https://en.wikipedia.org/wiki/Support\\_vector\\_machine](https://en.wikipedia.org/wiki/Support_vector_machine).
36. Sprendimų medžio metodas [žiūrėta 2016-12-02], prieiga per internetą: [https://en.wikipedia.org/wiki/Decision\\_tree](https://en.wikipedia.org/wiki/Decision_tree).
37. Android operacinės sistemos integruotas jutiklių fiksavimas [žiūrėta 2016-11-20], prieiga per internetą: [https://developer.android.com/guide/topics/sensors/sensors\\_overview.html](https://developer.android.com/guide/topics/sensors/sensors_overview.html).
38. Weka biblioteka mašiniam mokymuisi [žiūrėta 2016-12-05], prieiga per internetą: <http://weka.sourceforge.net/doc.stable/>.
39. TensorFlow biblioteka mašiniam mokymuisi [žiūrėta 2016-12-04], prieiga per internetą: <https://jalanmar.github.io/Supercharging-android-apps-using-tensorflow/>.
40. Scikit-learn biblioteka mašiniam mokymuisi [žiūrėta 2016-12-04], prieiga per internetą: <http://scikit-learn.org/stable/>.
41. Cordova biblioteka jutiklių sintezei [žiūrėta 2016-11-20], prieiga per internetą: <https://libraries.io/npm/cordova-plugin-device-sensor-fusion>.
42. Pasikartojančio neuroninio tinklo paveikslas [žiūrėta 2018-05-16], prieiga per internetą: <http://www.mattmoocar.me/blog/RNNCountryLyrics/>.

43. Vienašio akselerometro schemas paveikslas [žiūrėta 2018-05-16], prieiga per internetą: <http://ei-notes.blogspot.lt/2012/04/accelerometer.html>.
44. Android operacinės sistemos jutiklių lentelė [žiūrėta 2018-05-16], prieiga per internetą: [https://developer.android.com/guide/topics/sensors/sensors\\_motion](https://developer.android.com/guide/topics/sensors/sensors_motion).
45. Kryžminis patikrinimas [žiūrėta 2018-05-17], prieiga per internetą: [https://en.wikipedia.org/wiki/Cross-validation\\_\(statistics\)](https://en.wikipedia.org/wiki/Cross-validation_(statistics)).

## Terminai

Android – Mobili operacinė sistema sukurta Google, paremta Linux branduoliu ir pritaikyta mobiliesiems įrenginiams su liečiamu ekranu kaip išmanieji telefonai ir planšetiniai kompiuteriai.

Google – JAV tarptautinė korporacija, besispecializuojanti teikti interneto paslaugas ir gaminti su jomis susijusius produktus.

Linux – Laisvos (atviro kodo) operacinės sistemos branduolio pavadinimas.

Akselerometras – prietaisas, kuris išmatuoja pagreitį.

Giroskopas – prietaisas, skirtas sukimuisi surasti.

Magnetometras – prietaisas, skirtas matuoti magnetinei indukcijai.

Android Studio – Atviro kodo integruota kūrimo aplinka Java programavimo kalba.

Java – Bendrosios paskirties, objektiškai orientuota programavimo kalba.

USB – Universalioji jungtis, kuri naudojama kompiuteriuose.

Data mining – Tai yra duomenų gavybos procesas apimantis duomenų savybių išgavimą iš didelių duomenų rinkinių, naudojant statistinius mašininio mokymosi įrankius.

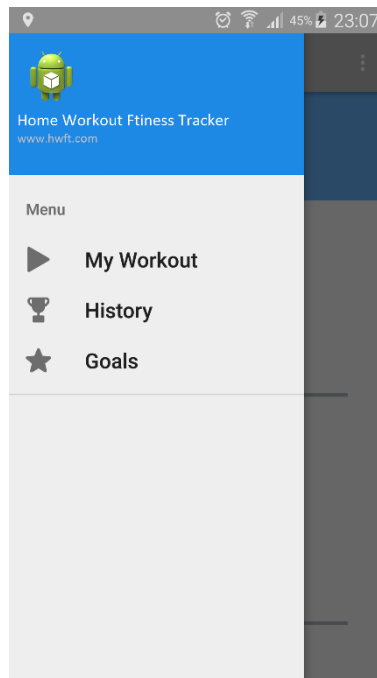
Mašininis mokymasis – Dirbtinis intelektas, kuris kuria metodus kompiuteriui mokinimuisi.

XML – Duomenų struktūrų ir turinio aprašomoji kalba.

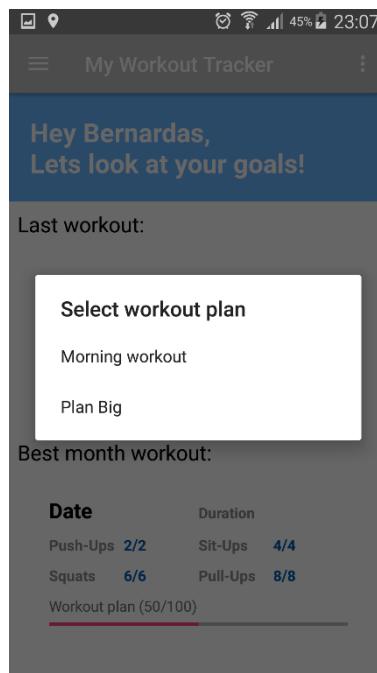
## Priedai

1 priedas

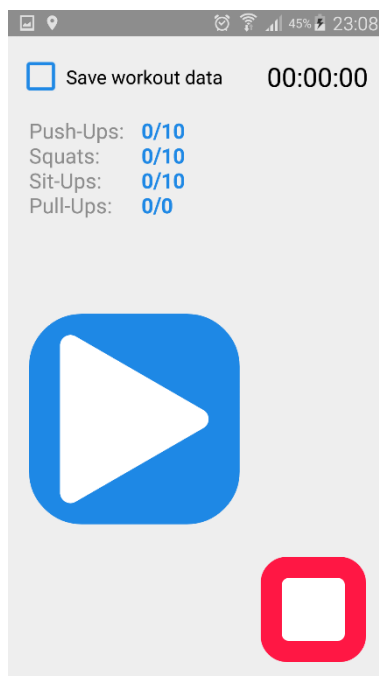
### Mobiliosios programėlės ekrano nuotraukos



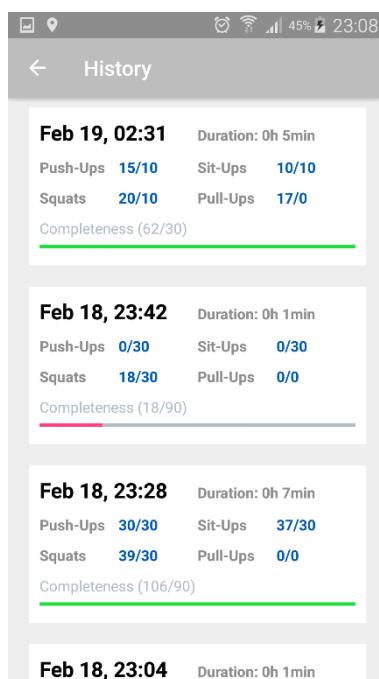
41 pav. Mobiliosios programėlės pagrindinis meniu



42 pav. Mobiliosios programėlės tikslo pasirinkimas prieš sportinės veiklos registravimą

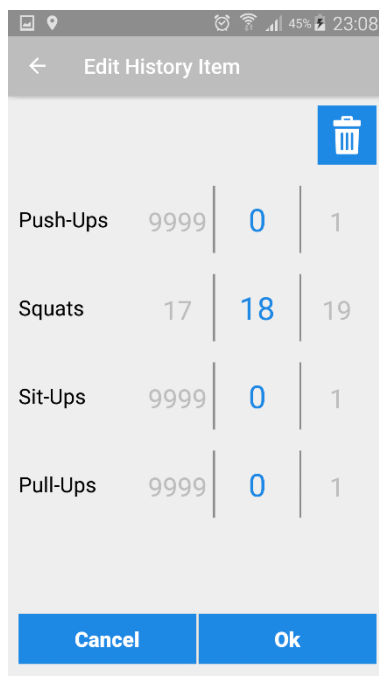


43 pav. Mobiliosios programėlės sportinės veiklos registravimo langas

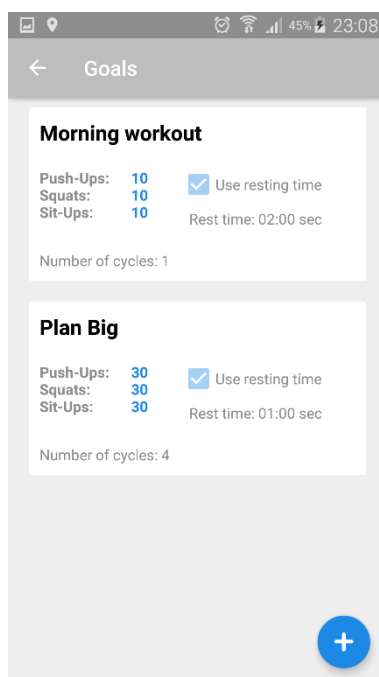


44 pav. Mobiliosios programėlės treniruočių rezultatų peržiūros langas

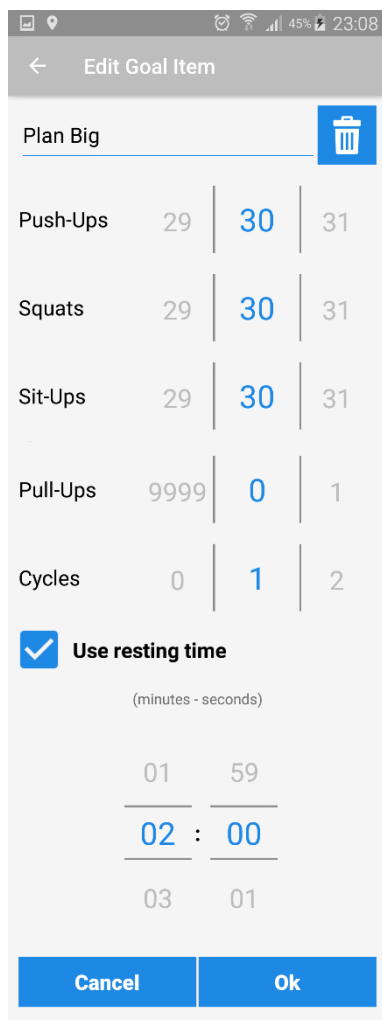




45 pav. Mobiliosios programėlės treniruočių rezultatų redagavimo langas



46 pav. Mobiliosios programėlės treniruočių tikslų peržiūros langas



47 pav. Mobiliosios programėlės treniruočių tikslų kūrimo ir redagavimo langas

Straipsnis

# Human Sport Activities Recognition and Registration from Portable Device

Bernardas Zokas, Mantas Lukoševičius  
 Department of Software Engineering  
 Kaunas University of Technology  
 Kaunas, Lithuania  
 zokas.bernardas@ktu.edu, mantas.lukosevicius@ktu.lt

**Abstract**—The drop in costs of hardware prices have led to significant changes in the size of various processors and sensors in smartphones. These devices come with a big range of new, precise measurement taking tools and multi-location sensors (distance sensor, accelerometer, gyroscope, magnetometer, camera and lighting sensors). This has opened the door for new smart device apps that can use data mining applications relying on sensor data. One of the main uses is the recognition of human movements. In this study, we propose a recognition and tracking method in sports activities such as push-ups, sit-ups and squats using only smartphone sensors and a machine learning algorithm. The key location for the smartphone is the upper part of the user's left arm. To collect the data and produce features for classifying sports activities, the motion data from accelerometer and gyroscope sensors is used. The features are made of two sliding windows and additional data processing which renders our classifier even more versatile. Fast response time, light-weight and accurate sport recognition can be used in mobile applications like our Home Workout Fitness Tracker which can process all the data in real time and create a real time sports activities tracking system.

**Keywords**—human activity recognition, machine learning, sensors, Android application, accelerometer, gyroscope, push-ups, squats, sit-ups.

## I. INTRODUCTION

Currently, more and more people are filling their lives with sports activities. For this purpose, smartphones are used to assist in capturing achievements of sport activities. Lower costs in making hardware have led to significant changes in the size, capabilities and functionality of various processors and sensors in smartphones. These devices come with a wide range of new, precise measurement taking tools and multi-location sensors like cellular radio, Wi-Fi radio, Bluetooth radio, microphone, cameras, GPS, accelerometer [17], gyroscope [18], compass [19], light and proximity sensors [2, 13]. This has opened the door to new smartphone apps that use different sensor data. One of the main uses is the recognition of human movements. The growing presence of sport activity recognition and data capturing gadgets still has a low market share. A common problem for people who are interested in their health and staying in shape is how to log and track sport activities during the workout. Therefore we came up with an idea of a smartphone application – a sport activity tracker, which uses tri-axial accelerometer and gyroscope data for recognition.

Smartphone sensor-based activity recognition topic is not new. Our work is slightly different from most previous works because we use already built commercial mass-marketed device rather than a research-only device. Instead of five or more devices placed on different parts of the human body we are using only one, at specific location. Our work goal is to recognize specific sport activities and count their repetitions, while performing and storing all recognitions and calculations to a single device. The classifier was built using only one person's sport activity data, but after processing and extracting features we created a universal classifier model, which is independent from the user; the key factor is location and position of the smartphone.

Our work has several features. Since we tested our classifier model in a real world environment with real people we can start collecting more data and improve our model for more sports activities, as a solution we can see improvement in pull-ups recognition and offer our app to a larger audience of users. In this way, our classifier would be able to recognize push-ups, squats, sit-ups and pull-ups. For such an application, there are almost no limits in functionality growth like joining heart rate monitor, GPS workouts and voice coaching program, etc. Secondly, we can offer the classifier model and dataset that we developed for further future researches as a solid foundation to start with.

In the following paragraphs, we discuss the related work (paragraph 2), describe our dataset collection, and describe our classifier model and an approach which we use to recognize activity from sensor data (paragraph 3). As a result of the research, we will show our experiment data and the capabilities of our application (paragraph 4-5).

## II. RELATED WORK

Human activity recognition is a wide field for science, researches and data mining. Human activity recognition can be divided in two main groups of video sensor based activity recognition (VSAR) and physical sensor based activity recognition (PSAR) [1]. According to the review (PSAR) can be split in to two smaller groups of wearable sensors and object usage based activity recognition (WSAR and OUAR). Our research will uses wearable sensors embedded on the smartphone itself. In the other article of Activity Recognition on Mobile Phones [2] the author made a review about available

sensors on the mobile phones. This review also provides main scope of activity recognition, methods of solving activity recognition problems, extracting features and basic algorithms for applying data mining. We can specify that our research in the category of using wearable sensors (in our case sensors on the smartphone) for human activity recognition.

Activity recognition using smartphones has a growing potential in the research of data mining since smartphones become equipped with a wide variety of sensors. The most effective sensors for activity recognition and data mining is accelerometer and little less used in the field are gyroscope and magnetometer. Each of those sensors producing three dimensional motion data. There are many studies done using these smartphone sensors for specific uses or applications, our project is not an exception. One of them is very similar [5], where two students were solving almost the same problem. They were making push-ups and non-push-ups recognition classifier using Support Vector Machine and Multilayer Perceptron. For feature extraction they used sliding window method with 1s, 2s, 4s and 8s length windows. They were solving not only push-up recognition problem but also introduced classifier of squats and sit-ups as a non-push-ups class. They achieved 98% classification rate for push-ups.

Other researchers [3-4, 6, 8] were solving walking, slow walking, jogging, going up stairs, going down stairs, sitting and standing recognition problems. In all these papers authors were using Weka (Waikato Environment for Knowledge Analysis) toolkit [9] and analyzed existing machine learning algorithms. Mostly in all researches the best recognition rate of over 90% was produced by the Multilayer Perceptron algorithm. Data were collected using smartphone accelerometer and gyroscope tri-axial sensors. Features for training set were produced using sliding window methods together examining and extracting values of mean, standard deviation, mean absolute deviation, time between peaks and the resultant magnitude.

Similar work but using different machine learning algorithm was done in a study [7] where authors were solving the problem of walking, jogging, running, going upstairs and going downstairs activity recognition. Recognition rate of 94% was achieved using DTW (Dynamic Time Warping) [10] algorithm simply by matching templates of the accelerometer sensor signal data. The strength of this method is, that it does not depend on the length of the signal. In this case specific accelerometer signal can be shorter or longer than its template. In this work as an additional feature was introduced context filtering, which was done by measuring user's heart rate and barometer readings when he was performing the activity.

### III. METHODOLOGY

In this section we describe our main task of sport activity recognition and all the processes from start to finish of performing this task. In the section Overview we explain the algorithm of our sport activity recognition application.

#### A. Overview

To begin with at first we would like to introduce our algorithm structure which is shown in the Fig. 1. In this figure we can see User motion (human arm movement during sport activity) which is captured using smartphone sensors

(accelerometer, gyroscope and magnetometer) and logged into smartphone's memory, then collected sensor data is send to Processing where main features are extracted. The next step leads to Machine Learning Algorithm where extracted features are classified and prediction is made for estimating sport activity. Having the same prediction which continuous for certain amount of time the Activity Repetition Counter counts and triggers about new counted sport activity repetition and all the results are send to GUI where user is able to see his progress. The main steps are explained in details below: Sensor – Section C; Preprocessing – Section D; Machine Learning Algorithm – Section E; Activity Repetition Counter – Section F.

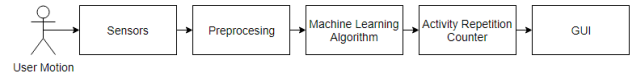


Fig. 1. Structure of the sports activity recognition system

#### B. System Environment

We have chosen Android-based smartphones as the platform for our project. The Android operating system [11] is free, open-source, easy to program, and dominant in the smartphones market. For the project we will use 5.0 and higher versions of Android operating system, which gives us more features to work with and much bigger computing power. Furthermore, smartphones with higher OS versions have more precise sensors. Because of big computing power and built-in sensors, smartphones are an ideal platform for our application; we also require no internet access or additional machine for calculations. The collected data can be stored directly in the smartphone's memory, because the Android OS provides a built-in SQLite [12] database and big storage capacity.

#### C. Data Collecting

To begin with, first we need to decide which part of the human body is the best for locating the smart phone. We are collecting sport activities data such as push-ups, sit-ups, and squats workout, we need to locate smartphone [14] on human body so we can measure all of the listed activities. After some experimenting we found out that the most sensitive place for taking measurements with smartphone on human body is left upper arm next to the shoulder (left arm was chosen because of better smartphone GUI control with the right hand) with the screen facing outwards. We mounted the smartphone using duct tape instead of an armband for a better grip. For data collecting smartphone accelerometer, gyroscope and magnetometer sensor's data is captured with the Android application designed to access to Android sensor event services where we read sensor data. Because we are reading three tri-axial sensors, in total we have nine different arm motion signals (magnetometer x, y, z axis, gyroscope x, y, z axis and magnetometer x, y, z axis). Data reading frequency is 10Hz: every 100ms a new reading is made. Since we have limited resources of processing power in smartphone, we decided to use lower frequency of sensor data readings. Each sensor reading is added to the array list with an included timestamp. At the end of sport activity data logging we stop our application and save all collected data into a .csv format file located in smartphone's external data storage with a

specified file name of sport activity, to easily access it with the computer.

#### D. Features Extraction

In previous step we collected our sensor data and now we need to extract some features to be able to make a training dataset and to train a classifier. During examination of our collected data, we noticed that some of the sensor's signals are unreliable. This is happening because sport activity repetition have a specific pattern, which can be seen in some of the sensors signals, but this pattern is not visible in all of the sensors signals. Some other signals have no pattern at all, for example all of the magnetometer signals were not matching the pattern and were unusable. Some of the signals were duplicated like accelerometer y and z axis. And some of the signals were reliable for one sport activity but not for another. We needed to select only those sensor (magnetometer x, y, z axis, gyroscope x, y, z axis and magnetometer x, y, z axis) signals which were useful for all sport activities, matched pattern and were most promising, also we need not to overload the smartphone processor. After time consuming and frustrating experimentation with the different combinations of signals we found out that the most appropriate signals are accelerometer x, z axis and gyroscope z axis. The problem was that we have different sport activities and at some point not all sensor's axis were used so the signal readings were almost strait line. Some of the measurements taken in the process are shown in the Fig. 2-4, where we can see different sport activities and three chosen signals (accelerometer x and z axis and gyroscope z axis) for later data processing.

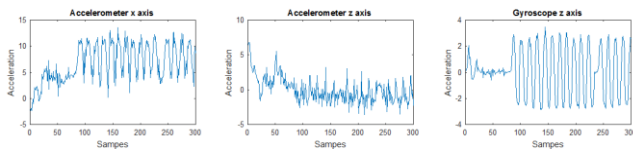


Fig. 2. Example of push-ups signals.

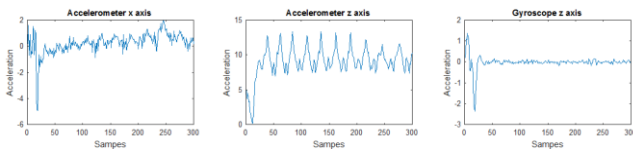


Fig. 3. Example of squats signals

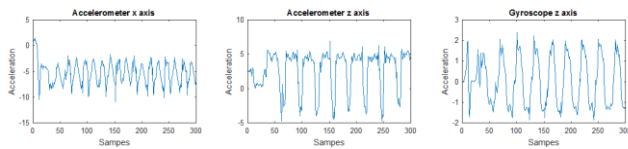


Fig. 4. Example of sit-ups signals

Since we have unlabeled data of accelerometer x, z axis and gyroscope z axis, we need to label class for each activity. We have three activities and lots of false data in between repetitions. From previous experimentations we learned, that in order to make accurate classifier we need to label only those signal parts which contain repetition and other must be labeled as false data.

To do so we created simple program with MATLAB [15] for filtering and labeling signal data. The algorithm is very simple, using specific parameters of signal height, median, and signal length, we label those signal parts where it meets all parameter values. An example of the signal labeling is shown in the Fig. 5. Green vertical line is start of the filtered signal, red vertical line is middle of the filtered signal and blue vertical line is end of the filtered signal. During lots and lots of experimentations by filtering repetitions and making classifiers we came up with the solution of using two fixed signal lengths one of 13 samples and one of 25 samples for repetition filtration. That way we also need two classifiers instead of one, because two classifiers are more versatile for short and long signal sample ranges. We separate all collected data files into two groups of short and long repetitions and applied labeling algorithm for each of them.

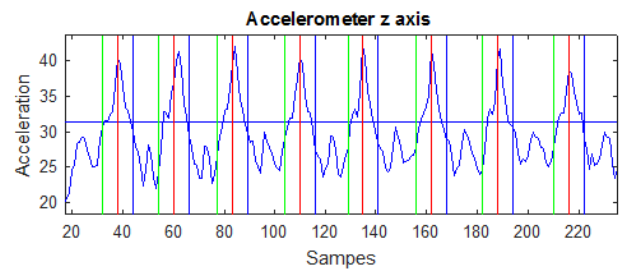


Fig. 5. Labeled signal example after filtration

The next question that we faced is how to train the classifier with the signal data. We cannot use the whole signal and also we cannot use only one signal sample for training the classifier. The idea is to split signal into small parts of 10 samples for 1 second signal classifier and 20 samples for 2 second signal classifier which came from the dataset analysis where we spotted that most of the sport activities' repetitions are in the interval of 0.5 to 3 seconds. The best way to split signal data is by using a Sliding Window (SW) [16] method. Our SW consists of 10 samples in length, for each iteration we will select 10 rows at a time and transform it to 30 columns (30 columns because we have three different signals for each row) and after each iteration we will move one row lower. The same step we will repeat for 20 sample length SW, but instead 10 rows we will select 20.

#### E. Machine Learning Algorithms Comparison

Three sport activities were studied as listed above. We extracted features and labeled them to the classes of separate different sport activities, in total we have two training datasets for our machine learning classification problem. One dataset for short sport activities repetitions in average of 1 second time duration and one for longer sport activities repetitions in average of 2 second time duration.

We performed and evaluated the performance of the following classifiers available in the Weka (Waikato Environment for Knowledge Analysis) toolkit [9], which are: Multilayer Perceptron, Random Forest, Simple Logistic and Logit Boost. Classifiers were trained and tested using a 10-fold cross validation method on the training datasets using default options.

The summary of the trained and tested classifier with both datasets recognition rate for sport activities is shown in the TABLE I. The most promising are Multilayer Perceptron and Random Forest. Random Forest have advantage in training time and recognition rate for both 1 and 2 second classifier. In the other hand Multilayer Perceptron is losing in recognition rate for both classifiers and also in training time which is significant longer. Other two classifier methods Simple Logistic and Logit Boost are slightly lower in recognition rate, but still take less time to train than Multilayer Perceptron.

TABLE I. TABLE OF MACHINE LEARNING CLASSIFIERS TEST RESULTS

Machine Learning Classifier	Recognition rate, %	
	1 second	2 seconds
Multilayer Perceptron	97.144	96.7
Simple Logistic	97.06	95.837
Logit Boost	96.56	94.885
Random Forest	97.582	97.252

#### F. Sport Activity Repetition Counter

Looking at the result the best candidate would be Random Forest, but we decided to use Multilayer Perceptron. It seems to be, that even if perceptron is not so accurate in overall recognition rate, but in real world test it shown exact prediction for the current class comparing with Random Forest. Test was performed using 2 second signal classifier to classify 1 second test dataset. A Multilayer Perceptron and Random Forest recognition accuracy test on labeled dataset is shown in the Fig. 6. We can see that Multilayer Perceptron gives us almost the same prediction like one labeled in the training dataset. However Random Forest gives us only spikes that are far away from the required prediction. We must keep in mind that our goal is not just to recognize sport activities, but it is also to calculate repetitions and the closer we are to the labeled dataset pattern the better it is. Having spikes or one sample predictions per repetition like Random Forest have, we can't programmatically correct it or filter out some of the features, but with Multilayer Perceptron it is possible. For example, to count one repetition we must have at least 3 continuous predictions in a row without noise. Even if classifier is failing at same point, we still can rely on the average of continuous activity's predictions and improve repetition counter.

The additional repetition counter filtration is called as a safety feature. Furthermore, we have two classifiers instead of one universal and it provides us more chances to correctly recognize sport activities.

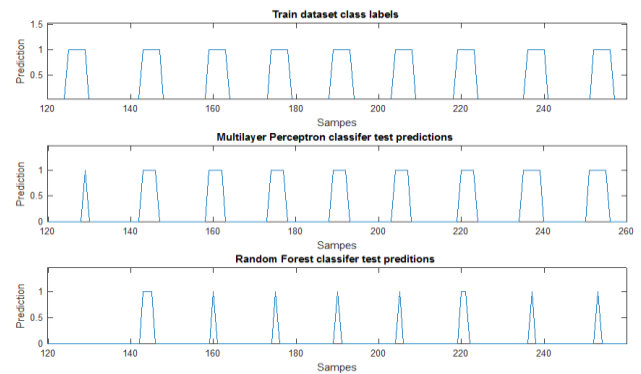


Fig. 6. Random Forest and Multilayer Perceptron prediction comparison

## IV. RESULTS

The summary of results for our activity recognition experiments are presented in the TABLE II. This table specifies the experiment made in real world environment with real humans. Experiment consists of 9 subjects who were asked to perform specific workouts (push-ups, squats and sit-ups) and for each of the workout subject needed to do 10 repetitions of fast (from 0.5s to 1s), normal (from 1s to 2s) and slow (from 2s to 3s) repetition speeds. Before performing sport activities recognition test we provide short tutorial to the participants of how to perform each workout correctly. Then we mounted smartphone with the installed application (the application screen shots of main menu, history window and main sport activity tracker window are represented in the Fig. 7) on the subject's left upper arm. To calibrate smartphone's exact location on the arm we asked to stretch out participant's hands in horizontal position with the palms facing down. All the participants were men of different heights and weights.

TABLE II. TABLE OF SPORT ACTIVITY RECOGNITION TEST WITH REAL SUBJECTS

Participants	Push-ups			Squats			Sit-ups		
	Fast	Normal	Slow	Fast	Normal	Slow	Fast	Normal	Slow
1.	10	10	10	10	10	10	9	10	10
2.	8	10	10	8	10	10	7	9	10
3.	9	10	10	8	10	10	7	10	10
4.	10	10	9	10	10	10	8	10	10
5.	10	10	10	10	10	10	9	10	10
6.	10	9	8	10	9	9	10	10	10
7.	10	10	10	10	10	9	10	10	9
8.	10	10	9	10	10	8	10	10	9
9.	8	10	10	9	10	10	8	10	10
Recognition rate, %	94.444	98.889	95.556	94.444	98.889	96.667	86.667	98.889	97.778

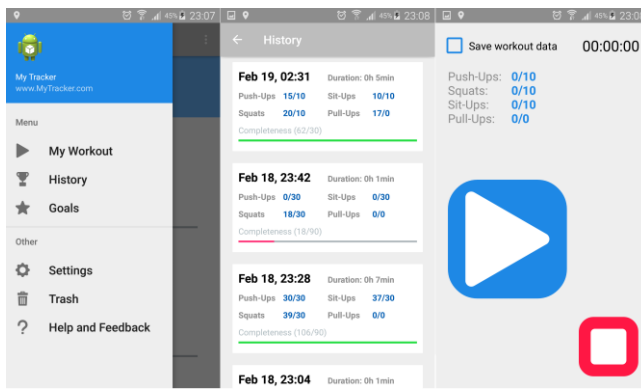


Fig. 7. Sport activity recognition application print screens

## V. CONCLUSION

In this study, human sport activity recognition accuracy of up to 95% on sport activities such as push-ups, squats and sit-ups using a tri-axial accelerometer and gyroscope was obtained. The sensors' data was collected from one subject by performing listed sport activities workouts using a smartphone as a sensor which was located on the subject's upper left arm. Collected data was separated into two groups of short and long activity's repetition performing speeds, labeled, and features were extracted. The training datasets were made and two classifiers were trained for each of collected data group. Combining two classifiers, smartphone's sensors and additional filtering for repetition counting the Android application was created. To measure sport activity recognition accuracy the test of nine male subjects was performed and achieved the average of 95.8% sport activity recognition rate.

The main benefits of the project are that sport activity recognition is done in smartphone by itself, using its own computing power and no additional devices, or servers, or internet access are required. All the collected data is processed in real time and have only 2 seconds time delay at displaying data in GUI, which is limited of 2 second classifier (longest time for getting 20 samples of data). The extracted features are little different from the previous works in other papers. The difference occurs that our method uses raw sensor's signal data instead of calculating additional features. During the recognition testing we have noticed that bad or misshape workout repetitions weren't counted, which leads us to make conclusion that application also promotes subject to perform right movements for the sport activity.

Existing project can be easily improved with additional testing with more subjects especially women and new classifier can be made with the bigger training dataset made of more than one subject's data. Furthermore, according to the plan current application needs to be improved to be capable of recognizing pull-ups and etc., which was mentioned in the application specification. Then it would have more sports activities on the list and make application even more advance. Other feature is to set up additional module for voice coaching to make sport activity recognition application more user friendly by using phrases to describe how much repetitions left, when the goal is achieved and push user to exercise harder. The existing sport

activity recognition algorithm has perfect core for adding more features by its flexibility and light framework. During the testing smartphone showed no visual lagging or overstress in performing the recognitions, it can be also used in a background mode when user can use other applications at the same time.

## REFERENCES

1. D. Guan, T. Ma, W. Yuan, Y. Lee and A. M. J. Sarkar, "Review of Sensor-based Activity Recognition Systems", *IETE Technical Review*, 28:5, 418-433, 2011.
2. Incel O.D., Kose M., Ersoy C. "A Review and Taxonomy of Activity Recognition on Mobile Phones", *BioNanoScience*, 3:145-171, 2013.
3. J. R. Kwapiz, G.M. Weiss and S. A. Moore, "Activity Recognition Using Cell Phone Accelerometers", *ACM SIGKDD Explorations Newsletter*, 12:2, 74-82, 2010.
4. Bayat, M. Pomplun, D. A. Tran, "A Study on Human Activity Recognition Using Accelerometer Data from Smartphones", *Procedia Computer Science* 34, 450-457, 2014.
5. M. Nilsson and H. Wilén, "Push-up Tracking through Smartphone Sensors", *School of Engineering Sciences*, 23p, 2016.
6. R. Piyare and S. R. Lee, "Dynamic Activity Recognition using Smartphone Sensor Data", *1st International Electronic Conference on Sensors and Applications*, 2014.
7. S. M. S. Hasan, M. Masnad, M. M. Khan, H. Mahmud and M. K. Hasan, "Human Activity Recognition using Smartphone Sensors with Context Filtering", *The Ninth International Conference on Advances in Computer-Human Interactions*, 2016.
8. M. Shoaib, S. Bosch, O. D. Incel, H. Scholten and P. JM Havinga, "Complex Human Activity Recognition Using Smartphone and Wrist-Worn Motion Sensors", *Multidisciplinary Digital Publishing Institute*, 16:4, 2016.
9. Weka machine learning toolkit. <http://www.cs.waikato.ac.nz/ml/index.html>. Accessed March 2018.
10. Dynamic Time Warping. <http://web.science.mq.edu.au/~cassidy/comp449/html/ch11s02.html>. Accessed March 2018.
11. Android OS. [https://en.wikipedia.org/wiki/Android\\_\(operating\\_system\)](https://en.wikipedia.org/wiki/Android_(operating_system)). Accessed March 2018.
12. SQLite. <https://www.sqlite.org/about.html>. Accessed March 2018.
13. N. D. Lane, E. Miluzzo, H. Lu, D. Peebles, T. Choudhury, and A. T. Campbell, "A survey of mobile phone sensing," *IEEE Communications Magazine*, vol. 48, no. 9, pp. 140-150, 2010.
14. W. Xu, M. Zhang, A. A. Sawchuk, and M. Sarrafzadeh, "Co-recognition of human activity and sensor location via compressed sensing in wearable body sensor networks," in *Proc. 9th IEEE Int. Conf. Wearable Implantable Body Sensor Netw.*, 2012, pp. 124-129.
15. MATLAB. <https://se.mathworks.com/products/matlab.html>. Accessed March 2018.
16. Sliding Window. [http://wcipeg.com/wiki/Sliding\\_window](http://wcipeg.com/wiki/Sliding_window). Accessed March 2018.
17. Accelerometer. <http://www.sensorwiki.org/doku.php/sensors/accelerometer>. Accessed March 2018.
18. Gyroscope. <http://www.sensorwiki.org/doku.php/sensors/gyroscope>. Accessed March 2018.

