

VILNIAUS UNIVERSITETAS  
MATEMATIKOS IR INFORMATIKOS FAKULTETAS  
PROGRAMŲ SISTEMŲ KATEDRA

**BetEasier**  
**statymų bei lažybų programėlė**

Programų sistemų inžinerijos I-asis laboratorinis darbas

Atliko:	2 kurso studentai	
	Greta Beresinevičiūtė	(parašas)
	Julija Detkova	(parašas)
	Paulius Nemeikšis	(parašas)
	Viktorija Šimėnaitė	(parašas)
Darbo vadovas:	Vytautas Valaitis, Asist., Dr.	(parašas)

# Anotacija

Studentų G.Beresinevičiūtės, J.Detkovos, P.Nemeikšio, V.Šimėnaitės „BetEasier“ mobilioji lažybų platforma. Programų sistemų inžinerijos laboratorinis darbas, VU PS katedra, Vilnius, 2019. Šiame dokumente apibūdinama programų sistemos architektūra, aprašant sistemą keliais pjūviais. Kartu su parengtu sistemos projektu pristatomas ir veikiantis sistemos prototipas. Grupės narių informacija:

- Greta Beresinevičiūtė  
El.paštas: greta.beresineviciute@mif.stud.vu.lt  
Indėlis į darbą:
- Julija Detkova  
El.paštas: julija.detkova@mif.stud.vu.lt  
Indėlis į darbą:
- Paulius Nemeikšis  
El.paštas: paulius.nemeiksis@mif.stud.vu.lt  
Indėlis į darbą:
- Viktorija Šimėnaitė  
El.paštas: viktorija.simenaite@mif.stud.vu.lt  
Indėlis į darbą:

## TURINYS

ANOTACIJA .....	1
IVADAS .....	3
1. UŽDUOTYS IR JŲ VYKDYMO SCENARIJAI .....	4
1.1. Sistemos vykdomos užduotys .....	4
1.1.1. Prisijungusio naudotojo užduočių dekompozicija .....	5
1.1.1.1. Užduotis: „Nusistatyti pinigų limitą“ .....	5
1.1.1.2. Užduotis: „Peržiūrėti esamas lažybas“ .....	5
1.1.1.3. Užduotis: „Filtruoti įrašus pagal kategorijas“ .....	5
1.1.1.4. Užduotis: „Statyti pinigus“ .....	6
1.1.1.5. Užduotis: „Peržiūrėti lažybų istoriją“ .....	7
1.1.1.6. Užduotis: „Peržiūrėti balansą“ .....	7
1.1.1.7. Užduotis: „Peržiūrėti įplaukas“ .....	7
1.1.1.8. Užduotis: „Peržiūrėti išplaukas“ .....	8
1.1.1.9. Užduotis: „Patvirtinti užsakymo apmokėjimą“ .....	8
1.1.2. Administratoriaus užduočių dekompozicija.....	8
1.1.2.1. Užduotis: „Matyti lažybų sąrašą“ .....	8
1.1.2.2. Užduotis: „Paruošti informaciją apie naujas lažybas“ .....	8
1.1.2.3. Užduotis: „Atnaujinti lažybų sąrašą“ .....	10
1.1.3. PayPal užduočių kompozicija.....	10
2. STRUKTŪRINIS PROGRAMŲ SISTEMOS MODELIS .....	11
2.1. UML klasių diagrama.....	11
2.2. Sistemos logikos apibūdinimas .....	11
3. DINAMINIS PROGRAMŲ SISTEMOS MODELIS .....	13
3.1. Prisijungusio naudotojo veiklos diagrama .....	13
3.2. Naudotojo prisijungimo bei registracijos veiklos diagrama .....	14
3.3. Statymo užsakymo vykdymo veiklos diagrama .....	15
4. PROGRAMŲ SISTEMOS KOMPONENTAI .....	16
4.1. Nulinio lygio UML komponentų diagrama .....	16
4.2. Pirmojo lygio UML komponentų diagrama.....	17
4.3. Užduočių ir komponentų ryšių matrica .....	17
5. KOMPONENTŲ IŠSKIRSTYMAS TINKLE .....	19
REZULTATAI .....	20
IŠVADOS .....	21
TERMINŲ ŽODYNAS .....	22

# Ivadas

## **Programų sistemų pavadinimas**

Kuriamos sistemos pavadinimas – mobilioji lažybų aplikacija „BetEasier“.

## **Dalykinė sritis**

Lažybos, asmeninių finansų balanso stebėjimas.

## **Probleminė sritis**

- Esamų panašaus tipo sistemų sudėtingumas
- Nepatogi įplaukų, išlaidų ataskaita
- Neefektyvi bei lėta prieiga prie statymų

## **Naudotojai**

Vyresni nei 18 metų asmenys, turintys išmanųjį telefoną bei bazinius įgūdžius kaip naudotis programėmis.

## **Darbo pagrindas**

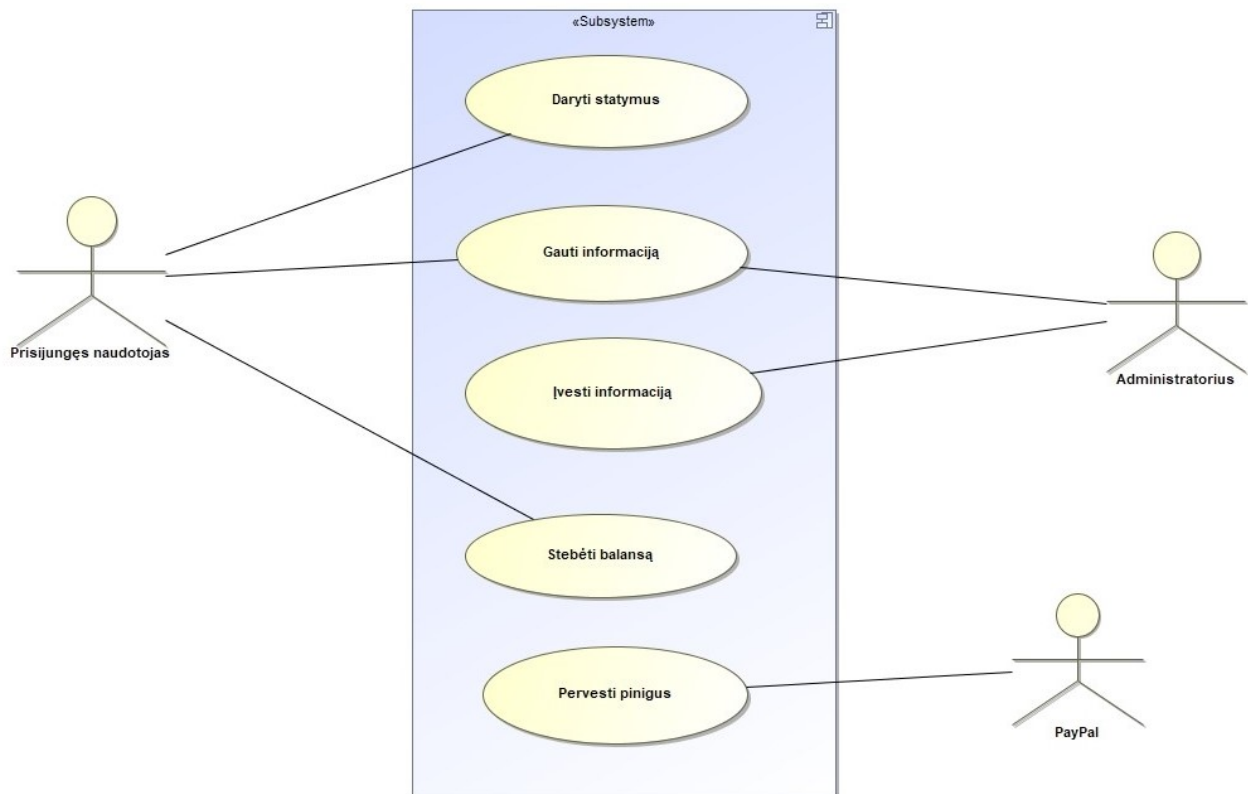
Dokumentas yra parengtas kaip programų sistemų inžinerijos laboratorinis darbas.

Programų sistemos architektūra pateikiama naudojant 4+1 pjūvių modelį. Pagal šį modelį dokumentą sudaro šios dalys: užduotys ir jų vykdymo scenarijai, struktūrinis programų sistemos modelis, dinaminis programų sistemos modelis, programų sistemos komponentai ir komponentų išskirstymas tinkle.

# 1. Užduotys ir jų vykdymo scenarijai

Šiame skyriuje aprašomų sistemos naudojimo atvejų vykdymo scenarijų paskirtis yra apibrėžti architektūrą bei patikrinti jos korektiškumą. Bus naudojamos UML naudojimo atvejų ir sekų diagramos.

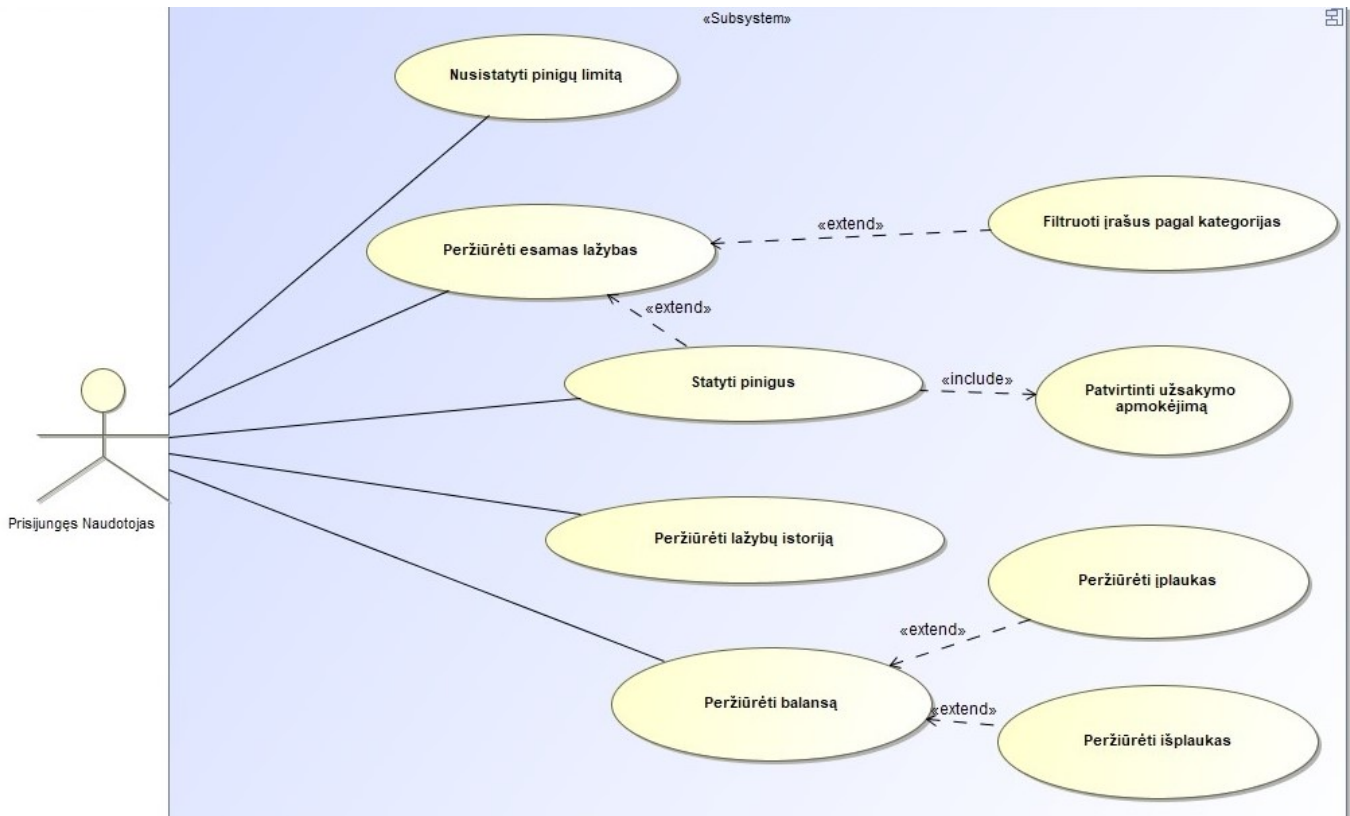
## 1.1. Sistemos vykdomos užduotys



1 pav. Sistemos vykdomų užduočių diagrama

1 pav. – sistemos naudojimo atvejų vykdymo diagrama, kurioje parodomi tikslai, kaip naudojami sistema agentai (prisijungęs naudotojas, lažybų bendrovės administratorius, banko darbuotojas).

### 1.1.1. Prisijungusio naudotojo užduočių dekompozicija



2 pav. Prisijungusio naudotojo užduočių dekompozicija

2 pav. dekomponuojamos prisijungusio naudotojo užduotys.

#### 1.1.1.1. Užduotis: „Nusistatyti pinigų limitą“

PAGRINDINIS SCENARIJUS: Naudotojas nuėjęs į „Personal Betting Bank Account“ skiltį, paspaudžia mygtuką „Limits“. Naudotojas pasirenka pinigų sumą, kurios nenori viršyti per vienas lažybas ir pinigų sumą, kurios nenori viršyti per mėnesį.

ALTERNATYVUS SCENARIJUS: Skiltyje naudotojas gali pasirinkti pinigų sumą, kurios nenori viršyti per vienas lažybas arba tik uždėti limitą mėnesiui.

#### 1.1.1.2. Užduotis: „Peržiūrėti esamas lažybas“

PAGRINDINIS SCENARIJUS: Naudotojas pagrindiniame mobiliosios programėlės lange „Betting Odds“ gali matyti visas numatomas rungtynes.

ALTERNATYVUS SCENARIJUS: Jei šiuo metu vykstančių lažybų nėra: sistema išmeta pranešimą „šiuo metu vykstančių lažybų nėra, prašome bandyti vėliau“.

#### 1.1.1.3. Užduotis: „Filtruoti įrašus pagal kategorijas“

PAGRINDINIS SCENARIJUS: Naudotojas pagrindiniame mobiliosios programėlės lange „Betting Odds“ viršutiniame dešiniajame kampe paspaudžia „Filter“ mygtuką ir pasirenka norimą filtravimą pagal kategorijas: čempionatus, sporto rūšį, laiką. Sistema atidaro filtruotą rungtynių sąrašą pagal naudotojo nurodytus kriterijus.

ALTERNATYVUS SCENARIJUS: Jei pagal pasirinktus filtravimus kriterijus varžybų nerasta, sistema išmeta langą su pranešimu „Varžybų, pagal pasirinktus kriterijus nerasta“.



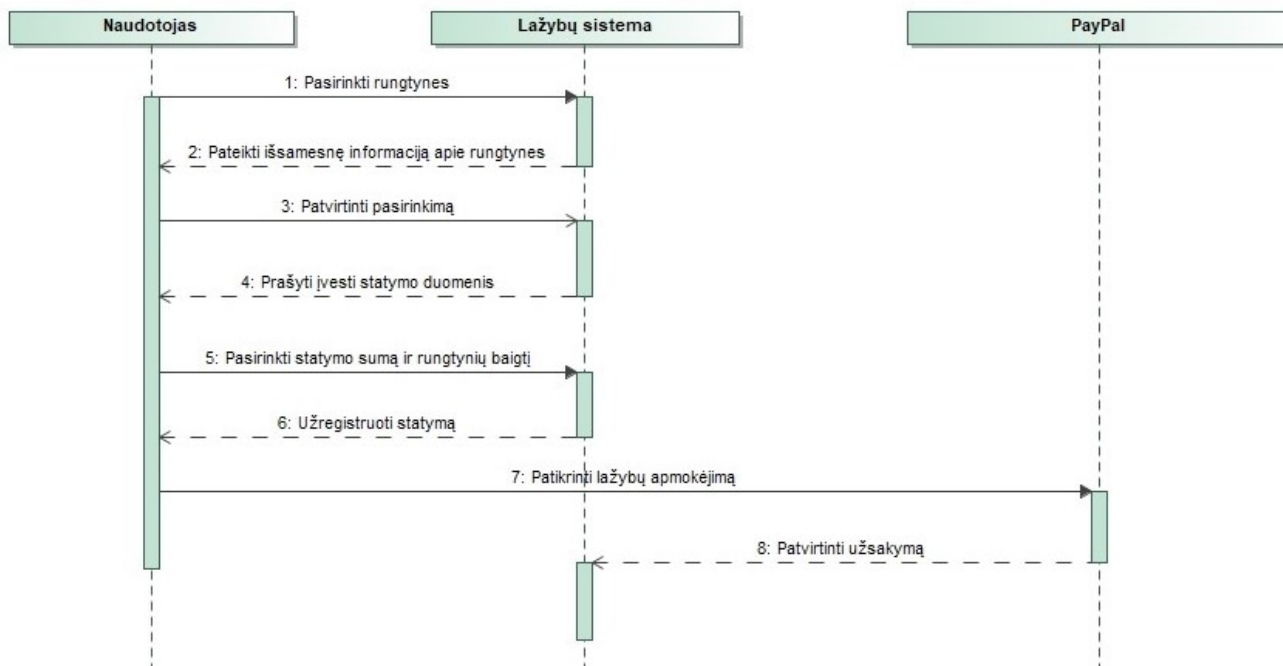
3 pav. pavaizduotas įrašų filtravimo užduoties vykdymas

Užduoties įgyvendinimo išsamesnė analizė pateikta 3 pav.

#### 1.1.1.4. Užduotis: „Statyti pinigus“

PAGRINDINIS SCENARIJUS: Sistemos lange „Betting odds“ rodo visas ateinančias rungtynes, naudotojas pasirenka rungtynes, kuriose nori statyti pinigus. Sistema išmeta išsamesnę informaciją apie pasirinktas lažybas: lygą, komandas, dabartinius koeficientus. Naudotojas spaudžia „BET“ mygtuką, jei nori tęsti toliau. Naudotojas pasirenka pinigų sumą, kurią nori statyti, lažybų baigtį ir spaudžia mygtuką „PAY“. Esate sujungiami su PayPal sistema.

ALTERNATYVUS SCENARIJUS: Naudotojas yra nusistatęs pinigų limitą: jei pinigų suma, kurią ruošiamąsi statyti yra didesnė už nustatyta limitą, sistema išmeta langą su pranešimu „Negalite viršyti nusi- statyto pinigų limito“.



4 pav. Užduoties „Statyti pinigus“ įgyvendinimas

4 pav. pavaizduotas pinigų statymo užduoties vykdymas. Naudotojas pasirenka lažybas iš sąrašo, kurias nori peržiūrėti. Sistema pateikia išsamesnę informaciją apie rungtynes. Jei naudotojas nori lažybose sudalyvauti spaudžia „BET“, sistema paprašo naudotojo pasirinkti statymo sumą. Suvedus duomenis statymas yra užregistruojamas. Naudotojas yra sujungiamas su PayPal sistema, kur turi patvirtinti lažybų apmokėjimą.

#### 1.1.1.5. Užduotis: „Peržiūrėti lažybų istoriją“

PAGRINDINIS SCENARIJUS: Naudotojas „Personal Betting Bank Account“ skilty pasirenka kategoriją „History“. Sistema išmeta langą su informacija apie lažybas, kuriose naudotojas dalyvavo, rungtynių baigtį, laimėtą/pralaimėtą sumą.

ALTERNATYVUS SCENARIJUS: Naudotojas nėra dalyvavęs lažybose: sistema išmeta pranešimą „nerasta lažybų“.

#### 1.1.1.6. Užduotis: „Peržiūrėti balansą“

PAGRINDINIS SCENARIJUS: Naudotojas „Personal Betting Bank Account“ skilty pasirenka kategoriją „Balance“, sistemoje pavaizduojamas laimėtų ir pralaimėtų pinigų santykis.

ALTERNATYVUS SCENARIJUS: Naudotojas nėra dalyvavęs lažybose: sistemoje nepateikiama jokia statistika.

#### 1.1.1.7. Užduotis: „Peržiūrėti įplaukas“

PAGRINDINIS SCENARIJUS: Naudotojas „Personal Betting Bank Account“ skilty pasirenka kategoriją „Income“, sistemoje naudotojas mato lažybose laimėtus pinigus.

ALTERNATYVUS SCENARIJUS: Naudotojas nėra dalyvavęs lažybose: sistemoje nepateikiama jokia statistika.



#### 1.1.1.8. Užduotis: „Peržiūrėti išplaukas“

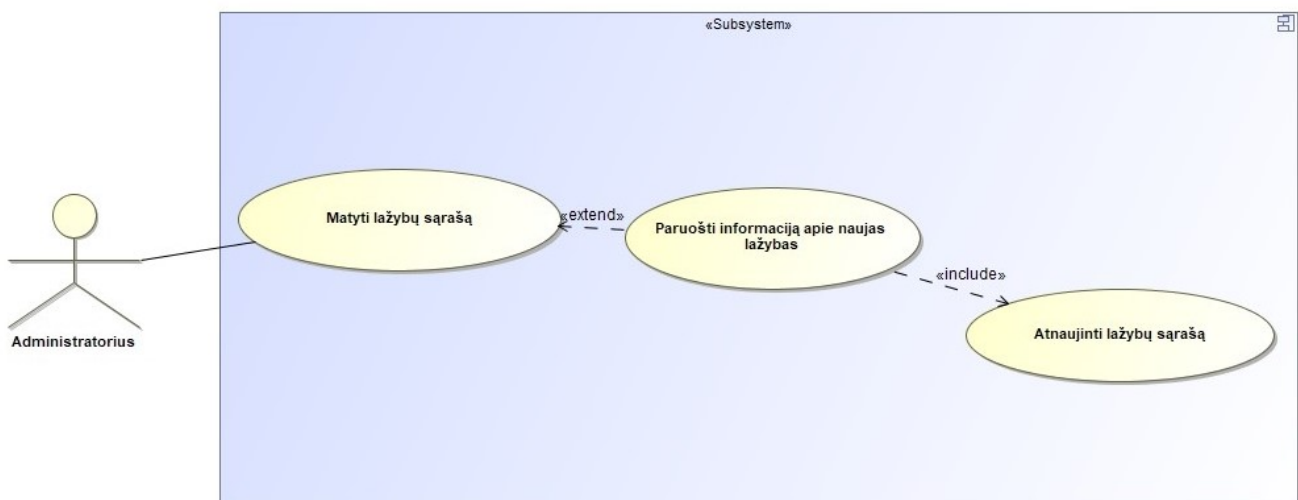
PAGRINDINIS SCENARIJUS: Naudotojas „Personal Betting Bank Account“ skilty pasirenka kategoriją „Outcome“, sistemoje naudotojas mato lažybose pralaimėtus pinigus. ALTERNATYVUS SCENARIJUS: Naudotojas nėra dalyvavęs lažybose: sistemoje nepateikiama jokia statistika.

#### 1.1.1.9. Užduotis: „Patvirtinti užsakymo apmokėjimą“

PAGRINDINIS SCENARIJUS: Atlikus norimą statymą, esate automatiškai sujungiami su PayPal sistema. Ten turite prisijungti, kad patvirtinti savo tapatybę.

ALTERNATYVUS SCENARIJUS: Klaidingai suvesti duomenys ar suma viršija limitą : nesate sujungiami su PayPal sistema.

### 1.1.2. Administratoriaus užduočių dekompozicija



5 pav. Administratoriaus užduočių dekompozicija

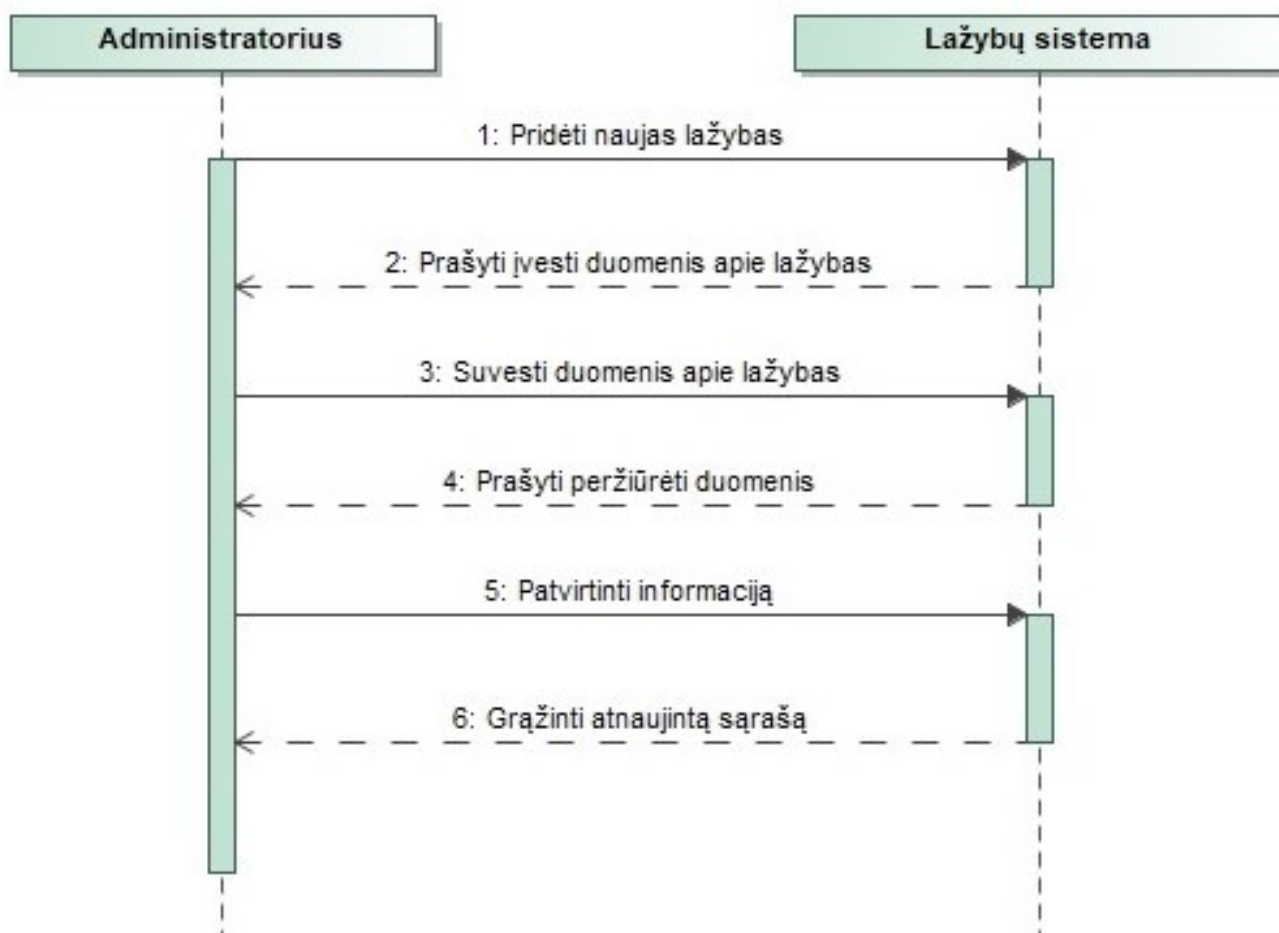
5 pav. pateikiama administratoriaus užduočių dekompozicija.

#### 1.1.2.1. Užduotis: „Matyti lažybų sąrašą“

Lažybų bendrovės administratorius sistemoje mato visas numatomas lažybas, informaciją apie jas.

#### 1.1.2.2. Užduotis: „Paruošti informaciją apie naujas lažybas“

Administratorius duomenų administravimo skiltyje paspaudęs mygtuką „+“ gali pridėti naujas sporto lažybas bei redaguoti informaciją apie rungtynes: rungtynių pradžios ir pabaigos laikai, komandos, čempionato tipas, sporto šaka ir kt. 6 pav. pavaizduotas išsamus užduoties vykdymas.

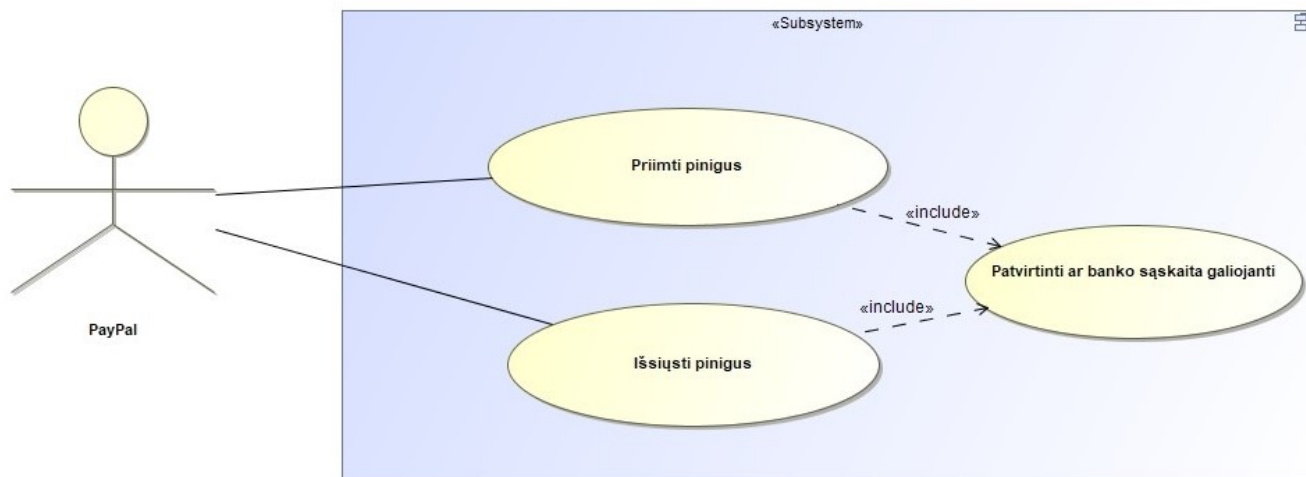


6 pav. Užduoties: „Paruošti informaciją apie naujas lažybas“ vykdymas

### 1.1.2.3. Užduotis: „Atnaujinti lažybų sąrašą“

Administratorius prideda naujas patvirtintas lažybas prie sąrašo. Sąrašas atnaujinamas.

### 1.1.3. PayPal užduočių kompozicija



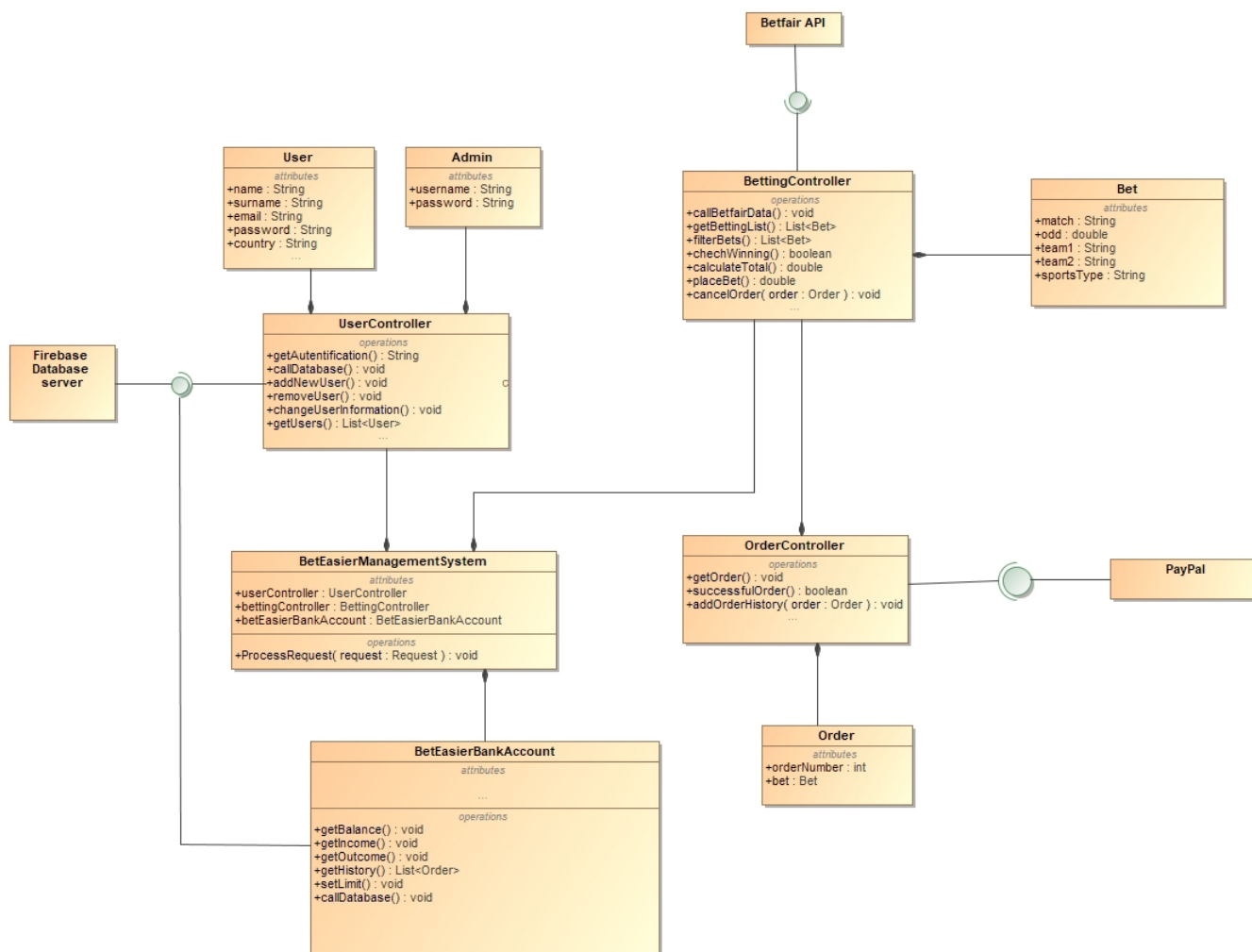
7 pav. PayPal užduočių kompozicijaa

7 pav. pateikta bendrinė PayPal užduočių vykdymo kompozicija.

## 2. Struktūrinis programų sistemos modelis

Šiame skyriuje sistema aprašoma, apibrėžiant ją įgyvendinančias klases ir ryšius tarp jų bei sąsajas, naudojantis UML klasių diagrama.

### 2.1. UML klasių diagrama



8 pav. Loginė sistemos UML komponentų diagrama

### 2.2. Sistemos logikos apibūdinimas

Remiantis 8 pav., pateikiamas detalus projekto sistemos logikos aprašas. Užpildęs prisijungimo duomenis, kurie naudojami UserController klasėje, naudotojas (User) įgauna teisę prisijungti prie mobiliosios aplikacijos. Sistemos administratorius, turintis asmeninį prisijungimo vardą bei slaptažodį, patvirtina arba prideda naują naudotoją prie sistemos. Administratorius turi papildomas funkcijas: keisti, pašalinti duomenis. Naudotojas, gavęs prieigą prie mobiliosios aplikacijos, išvysta galimų statymų sąrašą: vykstančias rungtynes, komandas, koeficientus. Pastaroji informacija yra gaunama iš Betfair API, kuri sistemai pateikia ne tik rungtynių sąrašą, bet ir informuoja apie rungtynių baigtį. Jau prasidėjusios rungtynės nėra rodomos ir pasiekiamos statymų sąrašuose, kadangi informacija yra nuolat atnaujinama, gaunant duomenis iš Betfair duomenų bazės. Naudotojo poreikiams patenkinti, šalia pateikiamas išplėstinis filtras, leidžiantis atrinkti

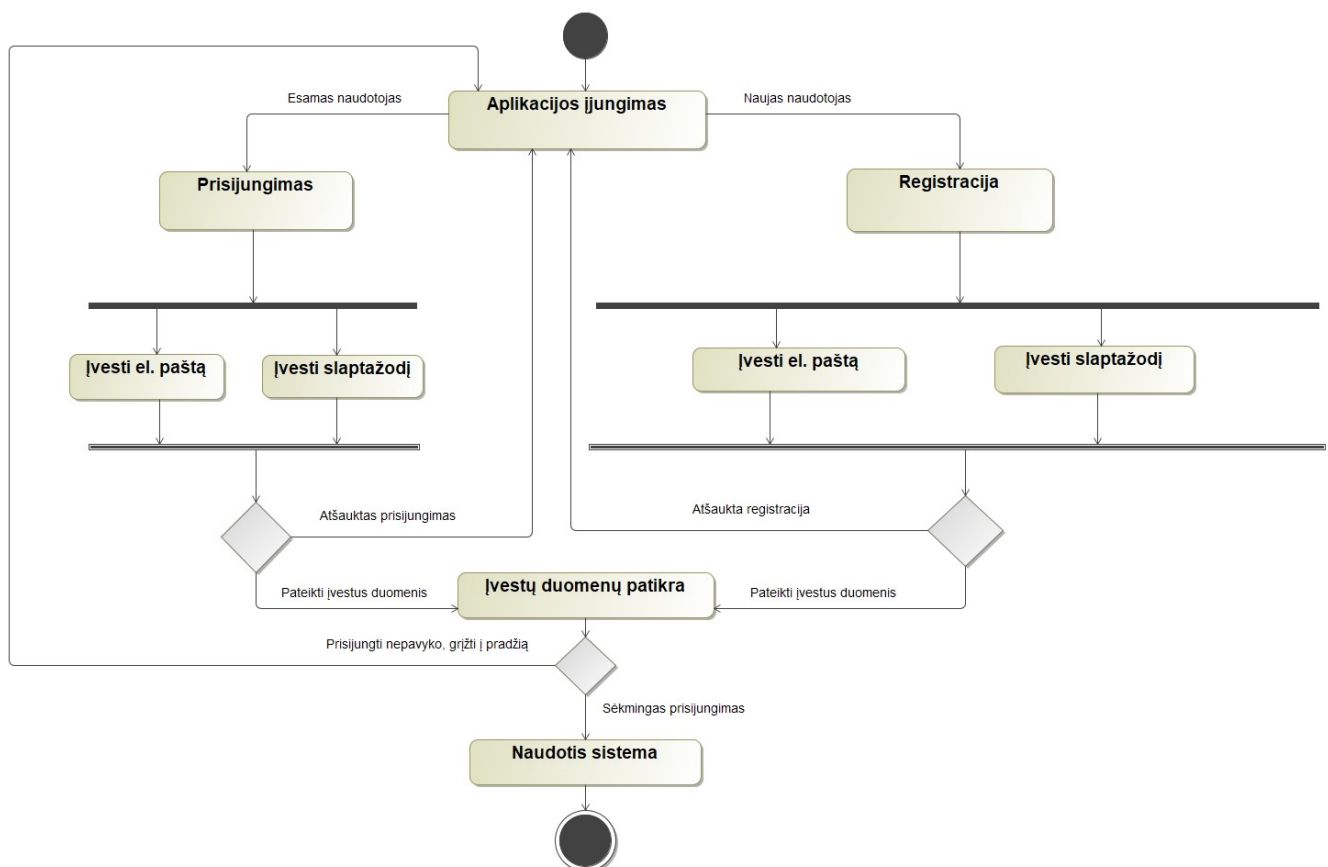
statymus pagal atitinkamus parametrus. Pasirinkus pageidaujamas rungtynes, naudotojas nukreipiamas į užsakymo užklausa, kurioje naudotojas turi įrašyti norimą statymo sumą. Toliau naudotojas turi patvirtinti savo tapatybę prisijungiant prie PayPal serverio sistemos, ši patvirtina arba atšaukia naudotojo pateiktą užsakymą ir grąžina sistemai informaciją apie sėkmingą arba nepavykusią transakciją. Sėkmingo užsakymo įvykdymo atveju naudotojas yra nukreipiamas į BetEasier asmeninio naudotojo banką, kuriame fiksuojamos patirtos išlaidos, gautos įplaukos, bendras balansas bei užsakymų istorija. Taip pat, naudotojas turi teisę limituoti savo statymų sumą. Gavus informaciją apie laimėtus/pralaimėtus pinigus, informacija asmeniniame programėlės banke yra atnaujinama.



prisijungimo duomenis. Esant nesėkmingam prisijungimui, prašoma įvesti naudotojo duomenis dar kartą. Sėkmingai prisijungus, einama prie mokėjimo patvirtinimo. Atšaukus mokėjimą, galima iš naujo rinktis rungtynes, už kurias bus statoma. Patvirtinus statymą, išsaugoma ir parodoma užsakymo informacija. Baigus peržiūrą, galima statyti dar kartą.

- Lažybų istorijos peržiūra – rodomos lažybos, kuriose naudotojas dalyvavo, ir informacija apie jas: rungtynių baigtis, laimėta/pralaimėta suma.
- Lažybų balanso peržiūra – rodomas laimėtų ir pralaimėtų pinigų santykis.
- Įplaukų peržiūra – rodoma laimėtų pinigų suma.
- Išplaukų peržiūra – rodoma pralaimėtų pinigų suma.

### 3.2. Naudotojo prisijungimo bei registracijos veiklos diagrama



10 pav. Naudotojo prisijungimo bei registracijos veiklos diagrama

Procesai, vykstantys naudotojo prisijungimo ir registracijos metu, pateikti 10 pav. Prisijungimas arba registracija vyksta naudotojui įjungus aplikaciją. Jau užregistravęs sistemoje naudotojas, prisijungia su elektroniniu paštu bei slaptažodžiu. Atvejais pasirinkus:

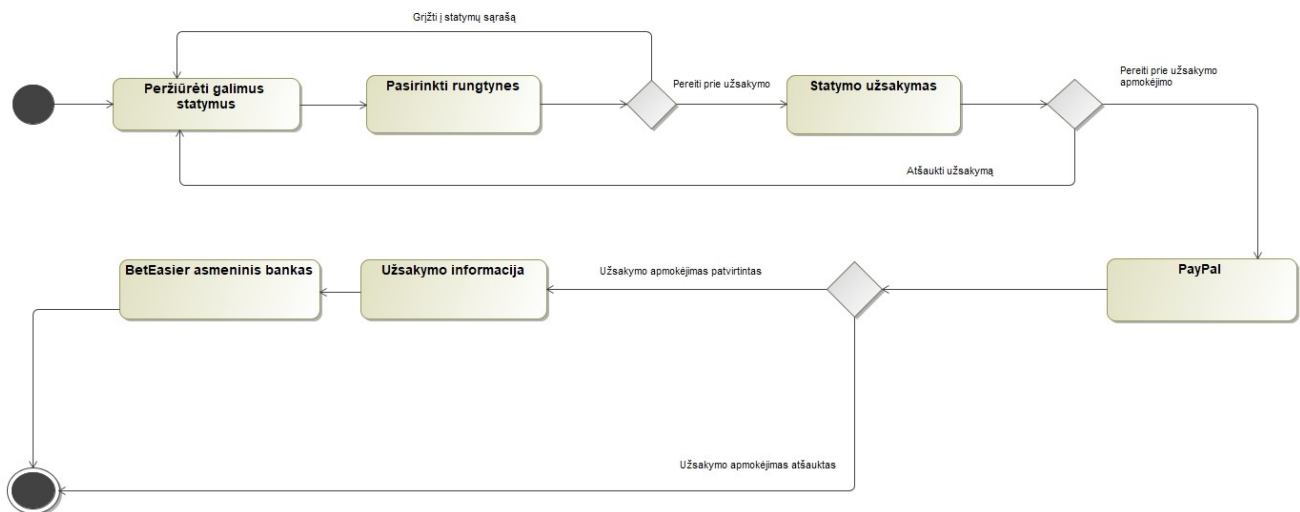
- prisijungimo atšaukimą - naudotojas nukreipiamas atgal į aplikacijos pradžios užklausą.
- tęsti prisijungimą - yra patikrinama įvestų duomenų validacija.

Duomenų patikros atvejais:

- įvesti neteisingi duomenys - prisijungimas nesėkmingas, naudotojas grąžinamas į pradžios užklausa.
- įvesti duomenys patvirtinami – naudotojas sėkmingai prisijungia prie sistemos ir gali ja naudotis.

Naujo naudotojo registracija sistemoje vyksta analogiškai.

### 3.3. Statymo užsakymo vykdymo veiklos diagrama



11 pav. Statymo užsakymo vykdymo veiklos diagrama

11 pav. pateikiamas detalesnis statymo užsakymo procesas. Prisijungusiam naudotojui yra pateikiamas visų šiuo metu galimų statymų sąrašas. Pasirinkus norimas rungtynes, naudotojas gali arba pereiti prie užsakymo užklauso, arba grįžti į pradinį statymų sąrašą. Užsakymo proceso vykdymo metu pasirenkama:

- pereiti prie apmokėjimo – naudotojas nukreipiamas į PayPal.
- atšaukti užsakymą – grįžtama prie statymų sąrašo.

Gavus užsakymo apmokėjimo patvirtinimą iš PayPal, užsakymo informacija išsaugoma ir pateikiama BetEasier asmeninio banko statistikai. Neįvykdžius užsakymo apmokėjimo – tolesniuose sistemos procesuose pateikiama atitinkama informacija.

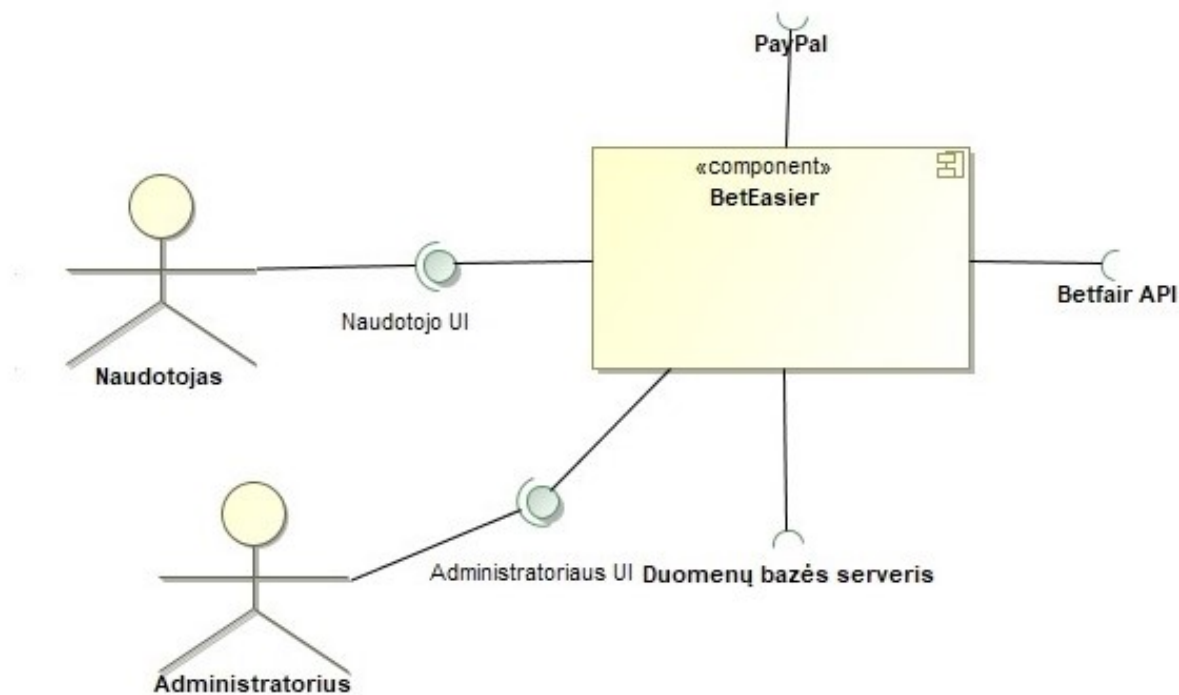


## 4. Programų sistemos komponentai

Šiame skyriuje sistemą įgyvendinančios klasės suskirstomos į komponentus, apibrėžiami ryšiai tarp jų. Komponentai ir jų ryšiai apibrėžia programų sistemos loginę struktūrą - sluoksnius, posistemių hierarchiją bei fizinę struktūrą. Toliau pateikiamas pagrindinis sistemos komponentų suskirstymas, kuris yra pateiktas UML komponentų diagramomis: L0 – nulinio lygmens bei L1 – pirmojo lygmens.

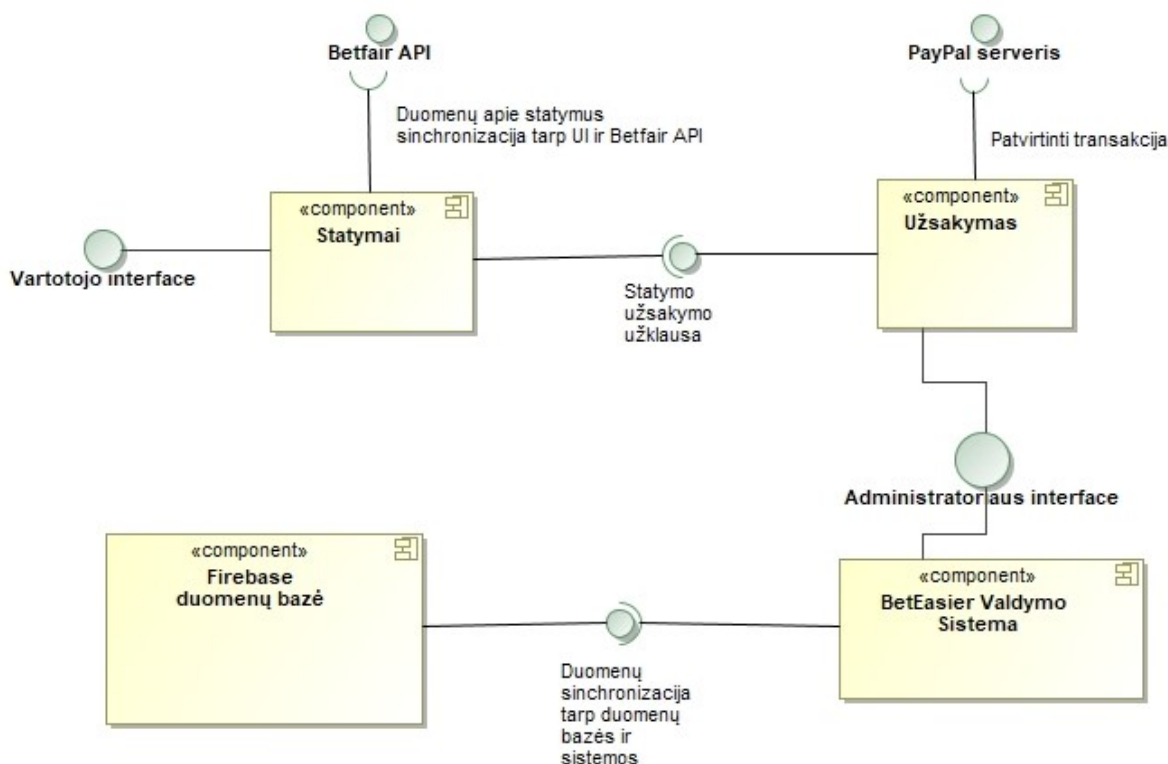
### 4.1. Nulinio lygio UML komponentų diagrama

12 pav. pateikiama bendrausio lygio (L0) komponentų UML diagrama. BetEasier sistemos naudotojai: prisijungę klientai bei administratorius. Sistema naudoja duomenų bazę, Betfair API ir PayPal serverį, sklandžiam statymų užklausų vykdymui, patvirtinimui, duomenų išsaugojimui.



12 pav. L0 lygio UML komponentų diagrama

## 4.2. Pirmojo lygio UML komponentų diagrama



13 pav. L1 lygio UML komponentų diagrama

13 pav. - L1 lygmenyje BetEasier komponentas dekomponuojamas. BetEasier komponento dekompozicija sudaryta iš 4 komponentų:

1. „BetEasier valdymo sistema” – pagrindinė programėlės sistema. Šis komponentas, kuris yra valdomas BetEasier administratoriaus, tvarko išorinių bei vidinių naudotojo užklausų tarpusavio ryšius, pakeitimus, kurie gaunami ir saugojami duomenų bazėje.
2. „Firebase duomenų bazė” – komponentas, skirtas duomenų saugojimui.
3. „Statymai” – šis komponentas atvaizduoja visą naudotojui reikalingą informaciją apie rungtynes bei statymus (rungtynių tipas, žaidžiančios komandos, koeficientai), kurią gauna iš Betfair API. Šis komponentas leidžia naudotojui iškviesti statymo užsakymo užklausa.
4. „Užsakymas” – klientui pateikus statymų užklausa, pastarosios apmokėjimo užklausa patenka į PayPal serverį, kuri po finansinės operacijos patvirtinimo praneša „BetEasier valdymo sistema” apie sėkmingą sandorį bei duomenys apie įvykdytą užsakymą patenka į duomenų bazę.

## 4.3. Užduočių ir komponentų ryšių matrica

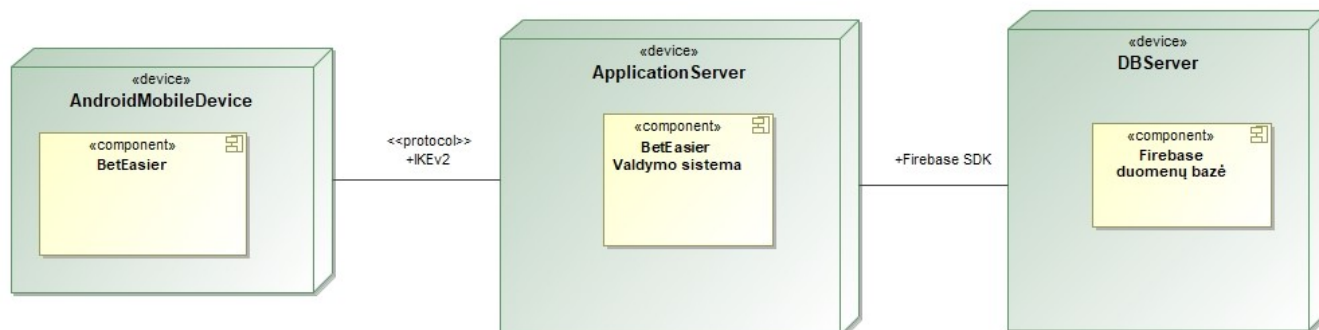
14 pav. pavaizduota užduočių ir komponentų ryšių matrica, kuri geriau perteikia komponentų ir užduočių ryšį, t.y. naudojamos informacijos keitimąsi tarpusavy.

<i>Komponentai</i> <i>Užduotys</i>					
	<i>Statymai</i>	<i>Statymo užsakymas</i>	<i>Valdymo sistema</i>	<i>Duombazė</i>	<i>PayPal serveris</i>
<i>Nusistatyti pinigų limitą</i>			✗	✗	
<i>Peržiūrėti esamas lažybas</i>	✗				
<i>Filtruoti įrašus pagal kategorijas</i>	✗				
<i>Statyti pinigus</i>	✗	✗	✗	✗	
<i>Peržiūrėti lažybų istoriją</i>				✗	
<i>Peržiūrėti balansą</i>				✗	
<i>Peržiūrėti įplaukas</i>				✗	
<i>Peržiūrėti išplaukas</i>				✗	
<i>Patvirtinti užsakymo apmokėjimą</i>		✗	✗	✗	✗
<i>Matyti lažybų sąrašą</i>	✗		✗		
<i>Paruošti informaciją apie naujas lažybas</i>	✗		✗		
<i>Atnaujinti lažybų sąrašą</i>	✗		✗		

14 pav. Užduočių ir komponentų ryšių matrica

## 5. Komponentų išskirstymas tinkle

Skyriuje „Komponentų išskirstymas tinkle“ apibrėžiama PS naudojama aparatinė įranga, komunikacija tarp tinklo mazgų bei PS komponentų išdėstymas juose. Toliau pateikta diegimo diagrama, parodanti PS išsidėstymą fiziniuose įrenginiuose:



15 pav. Komponentų išskirstymo tinkle diagrama

15 pav. išdėstymo diagramoje pavaizduota, jog Android aplinkoje veikia BetEasier programėlė, kuri turi būti įdiegta į vartotojo Android mobilųjį įrenginį. Duomenys tarp AndroidMobileDevice mazgo ir ApplicationServer mazgo yra sinchronizuojami pagal IKEv2 protokolą. Nuotoliniame duombazės serverio (DBServer) mazge išsidėsčiusi Firebase duomenų bazė. Pastarosios integracija su Android mobiliosios aplikacijos serveriu (ApplicationServer) vykdoma naudojantis Firebase SDK (software development kit), kuomet mobilioji aplikacija gauna prieigą prie duomenų bazės per SDK esančias bibliotekas/API. Prie nuotolinio duombazės serverio gali prisijungti daug lokalių prietaisų.

## Rezultatai

Projektavimo dalyje, pasitelkiant skirtingus sistemos pjūvius, aprašyta statymų įvykdymo sistemos architektūra.

I-jame „Užduotys ir jų vykdymo scenarijai” pjūvyje, naudojantis užduočių bei sekų diagramomis, buvo įvardyti pagrindiniai Naudotojo, Administratoriaus ir PayPal naudojimosi sistema tikslai.

II-jame „Strukūrinis programų sistemos modelis” pjūvyje, pasitelkiant UML klasių diagramą, išaiškintas bendras sistemos loginis veikimas: apibrėžtos klasės ir ryšiai tarp jų.

III-jame „Dinaminis programų sistemos modelis” pjūvyje, naudojantis UML veiklos diagramomis, buvo išskirtas bendras sistemos veiklos modelis ir apibrėžtos detalesės prisijungimo/registracijos bei statymo užsakymo veiklos diagramos.

IV-jame „Programų sistemos komponentai” pjūvyje, naudojantis UML komponentų diagramomis, apibrėžti sistemos komponentai ir atlikta funkcinė dekompozicija bei užduočių ir komponentų ryšių matrica.

V-jame „Komponentų išskirstymas tinkle” pjūvyje, išanalizuotas sistemos panaudojimas fiziniuose įrenginiuose ir apibrėžtas sistemos diegimas, lokalūs ir nuotoliniai serveriai.

## **Išvados**

Remiantis projektavimo rezultatais, galima sukurti Android mobiliąją programėlę, skirtą naudotojui: tarptautiniu mastu matyti galimus statymus ir jų koeficientus, patogiai ir nesunkiai atlikti statymus bei sekti savo statymų asmeninius finansus, jų valdymą.

## Terminų žodynas

**Balansas** – laimėtų ir pralaimėtų pinigų santykis.

**Įplaukos** – lažybų metu laimėti pinigai.

**Išplaukos** – lažybų metu pralaimėti pinigai.

**Naudotojas** – asmuo, turintis prieigą prie lažybų aplikacijos ir aktyvią anketą joje.

**Pinigų limitas** - pinigų suma, kurios naudotojas nenori viršyti per vienas lažybas ar mėnesį.

**Prisijungęs naudotojas** – asmuo, įjungęs lažybų aplikaciją bei sėkmingai prisijungęs prie savo anketos su teisingais jam priklausančiais prisijungimo duomenimis.

**Sistemos administratorius** – asmuo, turintis prieigos teises prie vykstančių lažybų valdymo (galintis koreguoti vystančių lažybų detales ar pridėti naujas).

**Statymas** – lošėjo veiksmai, kuriais lošėjas pasirenka konkretų lošimą, statomą sumą ir savo atitinkamais veiksmais išreiškia norą dalyvauti lošime.

**Statoma suma** – pinigų suma, kurią lošėjas įmoka iš jam priklausančios sąskaitos, siekiant dalyvauti lošime.