

VILNIAUS UNIVERSITETAS  
MATEMATIKOS IR INFORMATIKOS FAKULTETAS  
PROGRAMŲ SISTEMŲ KATEDRA

**Socialinis Vilniaus universiteto tinklalapis**

**SocialVU**

Programų sistemų inžinerijos I-III laboratoriniai darbai

Atliko:	2 kurso 4 grupės studentai	
	Andrejus Voitovas	(parašas)
	Eglė Puodžiūnaitė	(parašas)
	Kasparas Kralikas	(parašas)
	Ieva Vizgirdaitė	(parašas)
Darbo vadovas:	asist. dr. Vytautas Valaitis	(parašas)

# ANOTACIJA

Šis dokumentas sudarytas iš trijų dalių. Pirmoje dalyje pateikiama dalykinės srities analizė. Išorinėje verslo proceso analizėje identifikuojamos pagrindinės grėsmės ir problemos, bei teikiama nauda. Vidinėje verslo proceso analizėje nustatomos verslo stiprybės ir silpnybės, kylančios iš paties verslo proceso. Pateikiamos problemos su kuriomis galima susidurti, bei jų sprendimo būdai, sistemos naudojimo scenarijus. Antroje dalyje detalai išdėstomi pagrindiniai socialinio tinklalapio reikalavimai. Iškiriami funkciniai, nefunkciniai bei vartotojo sąsajos reikalavimai. Trečioje dalyje aprašoma kuriamos sistemos architektūra, naudojant UML4+1 požiūrių rinkinį. Žvelgiama į sistemą 5 skirtingais požiūriais. Loginiame pjūvyje pateikiami architektūros projektavimo etapai. Užduočių pjūvyje aprašomi socialinio tinklalapio panaudojimo atvejai, pasitelkiant kiekvienos užduoties scenarijus. Kūrimo pjūvyje pateikiamos komponentų diagramos skirtingais lygmenimis. Fiziniame pjūvyje parodoma, kaip sistema išdėstoma tinkle, kaip ji diegiama, kokia iranga naudojama. Procesų pjūvis pateikia dinaminį sistemos modelį: paaiškina sistemoje vykstančius procesus, parodo, kaip procesai komunikuoja, galimus sistemos darbo atvejus. Rašant šį dokumentą buvo naudojamosi:

1. dr. Vytauto Valaičio internetinis puslapis (<https://klevas.mif.vu.lt/valaitis/>)
2. doc., dr. Karolio Petrausko internetiniu puslapiu (<http://klevas.mif.vu.lt/karolis/>)
3. Latex programa
4. lab. asst. Jono Brusoko parengtais kursinio darbo šablonais (<https://gitlab.com/JohnLogic/SE-course-work-template>)

## TURINYS

ANOTACIJA .....	2
ĮVADAS .....	5
1. KURIAMO VERSLO PROCESO ANALIZĖ .....	6
1.1. VERSLO PROCESO APRAŠAS .....	6
1.2. IŠORINĖ VERSLO PROCESO ANALIZĖ .....	7
1.2.1. Įeigos .....	7
1.2.2. Išeigos .....	7
1.2.3. Įvaizdis.....	7
1.2.4. Reguliavimas .....	8
1.2.5. Grėsmės.....	8
1.2.6. Neišnaudotos galimybės .....	8
1.3. VIDINĖ VERSLO PROCESO ANALIZĖ .....	10
1.3.1. Dalykinės srities statinė struktūra .....	10
1.3.2. Užduotys .....	11
1.3.3. Užduočių vykdymo scenarijai .....	12
1.3.4. Dalykinės srities dinaminė struktūra .....	13
1.4. ĮGYVENDINAMUMO IR NAUDOS ANALIZĖ .....	16
1.4.1. Operacinis įgyvendinamumas .....	16
1.4.2. Techninis įgyvendinamumas.....	16
1.4.3. Ekonominis įgyvendinamumas .....	16
1.4.4. Juridinis įgyvendinamumas.....	17
1.5. SISTEMOS NAUDOJIMO SCENARIJUS .....	18
1.5.1. Scenarijus.....	18
1.5.2. Sistemos teikiama nauda.....	22
1.5.3. Esama būklė .....	22
1.5.4. Priemonės scenarijui įgyvendinti .....	23
2. REIKALAVIMŲ SPECIFIKAVIMAS .....	24
2.1. FUNKCINIAI REIKALAVIMAI .....	24
2.1.1. Internetinės svetainės langai .....	24
2.1.2. Prisijungimas .....	24
2.1.3. Atsijungimas.....	25
2.1.4. Paskyros valdymas .....	25
2.1.5. Naujienos .....	26
2.1.6. Renginiai.....	27
2.1.7. D.U.K. ....	28
2.1.8. Dėstytojų sąrašas .....	29
2.1.9. Dėstytojo puslapis.....	29
2.1.10. Žinutės .....	30
2.1.11. Konspektai.....	32
2.2. NEFUNKCINIAI REIKALAVIMAI.....	33
2.2.1. Vidinių interfeisų reikalavimai .....	33
2.2.2. Veikimo reikalavimai .....	34
2.2.3. Diegimo reikalavimai .....	35
2.2.4. Aptarnavimo ir priežiūros reikalavimai .....	35
2.2.5. Tiražuojamumo reikalavimai .....	35
2.2.6. Apsaugos reikalavimai .....	36
2.2.7. Juridiniai reikalavimai .....	36

2.3. VARTOTOJO SAŠAJOS REIKALAVIMAI .....	37
2.3.1. Dalykinės srities metaforos reikalavimai.....	37
2.3.2. Užduočių formulavimo kalbos reikalavimai .....	39
2.3.3. Užduočių formulavimo protokolo reikalavimai.....	39
2.3.4. Pranešimo formulavimo reikalavimai .....	42
2.3.5. Interfeiso darnos ir standartizavimo reikalavimai .....	42
2.3.6. Interfeiso individualizavimo reikalavimai .....	42
3. KURIAMOS SISTEMOS ARCHITEKTŪRA .....	43
3.1. LOGINIS PJŪVIS .....	43
3.1.1. Esybių klasių diagrama (nulinis lygis) .....	43
3.1.2. Klasių diagrama (pirmas lygis) .....	44
3.2. UŽDUOČIŲ PJŪVIS .....	45
3.2.1. Sistemoje vykdomos užduotys .....	45
3.2.2. Užduočių vykdymo scenarijai .....	45
3.2.3. Užduoties "Pridėti naujieną" scenarijus.....	46
3.2.4. Užduoties "Ištrinti naujieną" scenarijus.....	46
3.2.5. Užduoties "Redaguoti naujieną" scenarijus .....	47
3.2.6. Užduoties "Peržiūrėti naujienas" scenarijus .....	48
3.2.7. Užduoties "Pridėti renginį" scenarijus .....	48
3.2.8. Užduoties "Ištrinti renginį" scenarijus .....	49
3.2.9. Užduoties "Redaguoti renginį" scenarijus .....	50
3.2.10. Užduoties "Peržiūrėti renginius" scenarijus .....	50
3.2.11. Užduoties "Pridėti D.U.K." scenarijus .....	51
3.2.12. Užduoties "Peržiūrėti D.U.K." scenarijus .....	52
3.2.13. Užduoties "Redaguoti pasirinktą klausimą" scenarijus .....	52
3.2.14. Užduoties "Peržiūrėti pasirinkto dėstytojo teikiamą informaciją" scenarijus .....	53
3.2.15. Užduoties "Peržiūrėti pasirinktą žinutę" scenarijus .....	54
3.2.16. Užduoties "Siųsti žinutę" scenarijus .....	55
3.2.17. Užduoties "Redaguoti asmeninį dėstytojo puslapį" scenarijus.....	56
3.2.18. Užduoties "Įkelti konspektą" scenarijus.....	57
3.2.19. Užduoties "Peržiūrėti konspektus" scenarijus .....	58
3.3. KŪRIMO PJŪVIS .....	59
3.3.1. Komponentų diagramos nulinis lygmuo .....	59
3.3.2. Komponentų diagramos pirmasis lygmuo.....	60
3.3.3. Komponentų diagramos antrasis lygmuo.....	61
3.4. FIZINIS PJŪVIS .....	62
3.4.1. Dislokavimo diagrama nr. 1 (komponentų ir artefaktų ryšių diagrama).....	62
3.4.2. Dislokavimo diagrama nr. 2 (mazgų ir artefaktų ryšių diagrama) .....	63
3.4.3. Dislokavimo diagrama nr. 3 (mazgų ir artefaktų egzempliorių diagrama) .....	64
3.5. PROCESO PJŪVIS .....	65
3.5.1. Proceso sekų diagramos .....	65
3.5.2. Veiklos diagramos.....	66
SAVOKOS .....	69
REZULTATAI .....	70

# ĮVADAS

Socialinis VU tinklapis kuriamas siekiant supaprastinti informacijos prieinamumą visai fakulteto bendruomenei. Tinklapis suteikia galimybę patogiai dalintis informacija, susisiekti su dėstytojais/studentais. Fakultetui suteikiama galimybė platinti su fakultetu susijusias naujienas. Dokumente pateikiamas pagrindinis programos funkcionalumas ir apribojimai jos kūrimui. Pasielkiant ICONIX metodą sudaromi užduočių scenarijai, apibrėžiamos pagrindinės kuriamos sistemos esybės, sudaromas struktūrinis dalykinės srities modelis. Šis dokumentas padeda projektuoti ir nuspręsti, kaip sistema turėtų būti įgyvendinta.

## **Temos aktualumas**

Šiuo metu, dauguma informacijos skirtos studentams yra išdėstyta skirtinguose tinklapiuose, todėl nuspręsta sudėti visą informaciją į vieną vietą. Yra atskiras universiteto naujienų puslapis, kiekvienas dėstytojas turi savo asmeninį puslapį bei el. paštą, tačiau šią informaciją studentams surasti yra sunku, o dėstytojams pateikti informaciją savo puslapiuose yra nepatogu.

## **Dalykinė sritis**

Socialinis Vilniaus Universiteto tinklapis.

## **Probleminė sritis**

Socialinis VU tinklapis suteiktų galimybę greitai ir paprastai pasiekti dėstytojų pateiktą informaciją. Taip pat susisiekti su pačiais dėstytojais, sekti aktualias fakulteto naujienas, renginius. Pagrindinis šio produkto išskirtinumas - greitai ir patogiai pasiekiamą informaciją.

## **Naudoti dokumentai**

Dokumentas parengtas pagal kursinio darbo reikalavimus naudotojant Latex programą ir jau sukurtus šablonus.

## **Darbo pagrindas**

Dokumentas parengtas kaip Programų sistemų inžinerijos I-III laboratoriniai darbai.

# 1. KURIAMO VERSLO PROCESO ANALIZĖ

## 1.1. VERSLO PROCESO APRAŠAS

Ši sistema turi trijų tipų vartotojus - studentus, dėstytojus bei administratorius. Visų pirma, prisijungus prie sistemos vartotojas yra nukreipiamas į pagrindinį sistemos puslapį.

Studentai pagrindiniame sistemos puslapyje gali matyti universiteto naujienas, artėjančius renginius. Taip pat čia yra prieinamos nuorodos į kitas esmines sistemos dalis - dėstytojų puslapius, konspektų dalinimosi platformą, D.U.K. skiltį bei žinučių sistemą. Kiekvienas studentas turi prieigą prie savo dėstytojų asmeninių puslapių. Čia galima rasti informaciją apie dėstomus dalykus, jų medžiagą, bei artėjančius atsiskaitymus. Žinučių sistemoje studentas įgalinamas komunikuoti tiek su kitais studentais, tiek su dėstytojais, taip pat žinučių pavidalu gaunami priminimai apie artėjančius renginius ar atsiskaitymus. Konspektų dalinimosi platformoje studentas gali peržiūrėti kitų studentų įkeltus konspektus, juos įsigyti. Taip pat studentas gali peržiūrėti jau įsigytus konspektus, juos peržiūrėti naršyklėje arba atsisiųsti į savo kompiuterį ar mobilųjį įrenginį. Galiausiai, studentas gali į sistemą patalpinti ir savuosius konspektus, nurodydamas konspektų paskirtį bei jų kainą. D.U.K. skiltyje studentai gali rasti naudingą informaciją įvairia tematika, taip pat atsakymus į didžiąją dalį iškylančių klausimų.

Dėstytojai pagrindiniame sistemos puslapyje taip pat gali matyti universiteto naujienas, artėjančius renginius. Čia jiems yra prieinamos ir nuorodos į esmines, dėstytojams skirtas, sistemos dalis - naujienų, renginių bei D.U.K. skilties redagavimą, žinučių sistemą bei asmeninio puslapio administravimo erdvę. Naujienų redagavimo skiltyje dėstytojai gali patalpinti informaciją apie įvairias aktualijas, taip pat redaguoti senesnius įrašus, juos ištrinti. Renginių redagavimo skiltyje dėstytojai gali kurti naujus renginius, siųsti kvietimus žinučių pavidalu, peržiūrėti senesnius renginius, juos ištrinti. D.U.K. skilties redagavimo aplinkoje dėstytojai gali peržiūrėti esamus klausimus ir atsakymus, redaguoti jų turinį, pridėti naujus klausimus. Žinučių sistemoje dėstytojai gali matyti gautas bei išsiųstas žinutes, gali siųsti žinutes studentams, kitiems dėstytojams, čia gauna priminimus apie artėjančius svarbius įvykius (renginius, atsiskaitymus). Asmeninio puslapio administravimo erdvėje dėstytojai gali talpinti informaciją apie savo dėstomus dalykus, šią informaciją paslėpti nuo studentų ar padaryti ją prieinamą tik šiuo metu tam tikrą dėstomą dalyką turintiems studentams. Taip pat dėstytojas gali kurti papildomus puslapius, prieinamus iš pagrindinio dėstytojo asmeninio puslapio.

Visą sistemos darbą prižiūri sistemos administratorius, jis atsakingas už tinklapio tvarkos palaikymą.

## 1.2. IŠORINĖ VERSLO PROCESO ANALIZĖ

Išorinės analizės metu išsiaiškinami ištekliai reikalingi pradėti verslą, išryškinamos kuriamo verslo grėsmės ir problemos, kuriami tinklapio teikiami rezultatai ir išanalizuojama rinkoje esanti situacija. Nustačius rinkoje esančius ir potencialius konkurentus išanalizuojamos neišnaudotos galimybės bei įvertinama situacija.

1, 2, 3 ir 4 lentelėse pateikiami įeigos, išeigos, reguliavimo ir įvaizdžio svarbiausi vertinimo kriterijai, jų matavimo vienetai. Nustatoma kritinė vertė, esama vertė ir siekiama vertė. Tai padeda nustatyti, ar tam tikras vertinimo kriterijus atitinka normas.

### 1.2.1. Įeigos

1. Studentai
2. Dėstytojai
3. Administratorius
4. Duomenų bazė
5. Serveris

1 lentelė. Įeigos

Vertinimo kriterijus	Vertinimo matas	Siekiama vertė	Kritinė vertė
Studentai	Kiekis studentų, naudojančių „SocialVU“ tinklapį	21281	10000
Dėstytojai	Kiekis dėstytojų, naudojančių „SocialVU“ tinklapį	2890	1500
Administratorius	Pašalintų naudotojų skaičius per mėnesį	0	30
Duomenų bazė	Talpa GB	50	10
Serveris	Vidutinis užklausos apdorojimo laikas sekundėmis	1	5

### 1.2.2. Išeigos

1. Studentai, radę visą reikiamą informaciją
2. Dėstytojai, pateikę visą norimą informaciją

2 lentelė. Išeigos

Vertinimo kriterijus	Vertinimo matas	Siekiama vertė	Kritinė vertė
Studentai, radę visą reikiamą informaciją	Kiekis per mėnesį %	100	50
Dėstytojai, pateikę visą norimą informaciją	Kiekis per mėnesį %	100	50

### 1.2.3. Įvaizdis

1. Dėstytojų bei studentų atsiliepimai
2. Sistemos žinomumas

3 lentelė. Įvaizdis

Vertinimo kriterijus	Vertinimo matas	Siekiamą vertė	Kritinė vertė
Dėstytojų bei studentų atsiliepimai	Teigiamų atsiliepimų % nuo visų atsiliepimų	90	50
Sistemos žinomumas	Vilniaus universiteto studentų % žinančių šį tinklalapį	100	50

#### 1.2.4. Reguliavimas

1. Asmens duomenų teisinės apsaugos įstatymas
2. Darbo kodeksas

4 lentelė. Įvaizdis

Vertinimo kriterijus	Vertinimo matas	Siekiamą vertė	Kritinė vertė
Asmens duomenų teisinės apsaugos įstatymas	Pažeidimų skaičius	0	0
Darbo kodeksas	Viršytas darbo laikas (kartais per metus)	0	0

#### 1.2.5. Grėsmės

Kuriamas verslas yra priklausomas nuo dviejų pagrindinių įvesčių - dėstytojų ir studentų. Jų skaičius ir aktyvumas lemia kuriamos programos sėkmę. Rinkoje jau egzistuoja nemažai socialinių tinklų, kuriuos naudoja dauguma studentų (Facebook, Instagram, LinkedIn, Twitter ir t.t.) be to dauguma dėstytojų yra įpratę skelbti visą informaciją savo sukurtuose puslapiuose arba bando integruoti informaciją į Vilniaus universiteto virtualią mokymosi aplinką, todėl konkurencija yra gana didelė. Kuo daugiau dėstytojų pradės naudoti mūsų sukurtą platformą, tuo daugiau studentų taip pat privalės naudoti šį tinklalapį, nes tik ten galės rasti jiems reikiamą informaciją. Sėkmingam kuriamo verslo proceso egzistavimui turi įtakos reitingų ir atsiliepimų skaičius bei kokybė, tačiau visų pirma reikia užtikrinti, jog dauguma studentų bei dėstytojų išbandytų mūsų sukurtą sistemą bei ją įvertintų.

#### 1.2.6. Neišnaudotos galimybės

Rinkoje jau egzistuojančios Vilniaus universiteto platformos turi trūkumų (dauguma dėstytojų skelbia informaciją skirtinguose puslapiuose, todėl studentams sunku juos surasti, ne visada yra galimybė susisiekti su dėstytojais, dauguma konspektų nėra lengvai pasiekiami visiems studentams), todėl kuriama sistema gali juos išnaudoti ir pasiūlyti studentams visą reikiamą informaciją rasti vienoje vietoje. Sistema suteiks galimybę išsiųsti laišką dėstytojui bei rasti visą paskelbtą



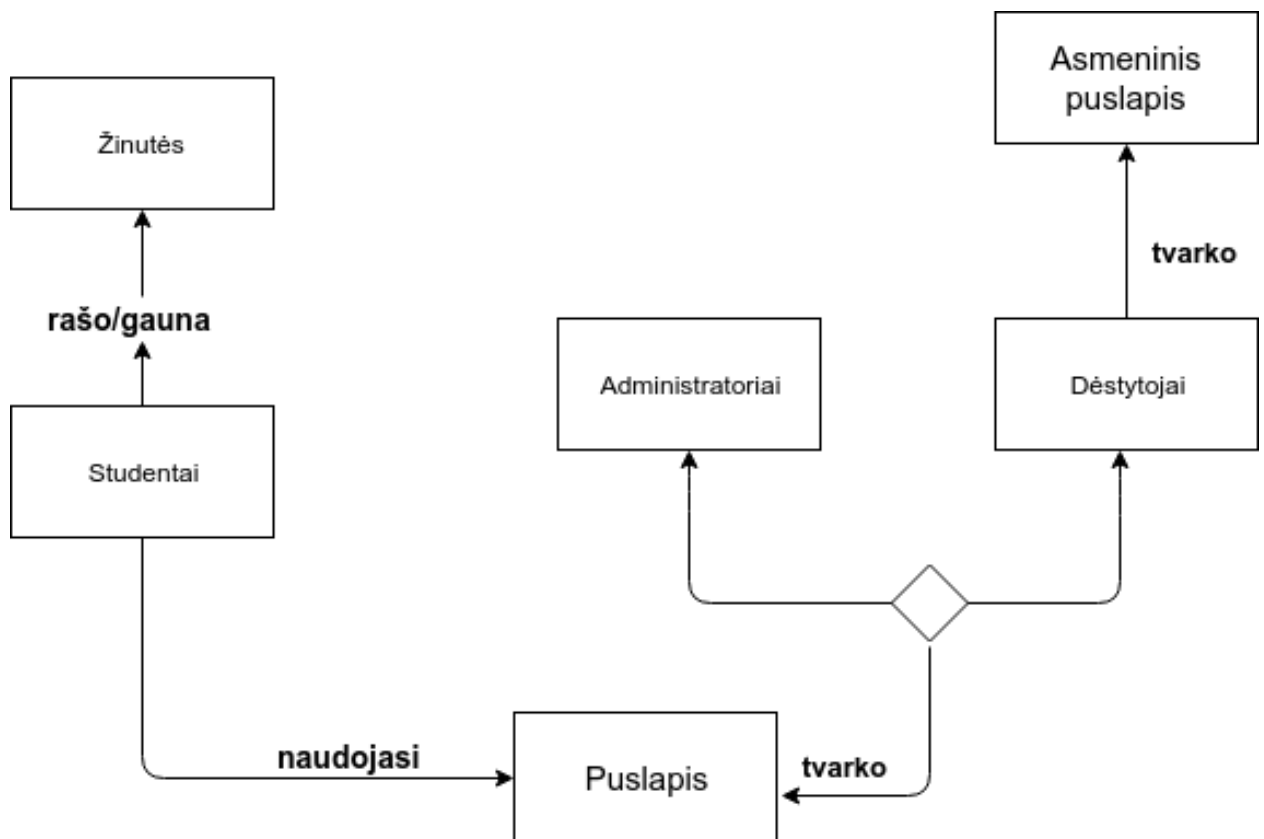
dėstytojo informaciją vienoje vietoje, studentai galės matyti naujausią informaciją bei terminus iki kada turi atlikti tam tikras užduotis. Be to vienoje vietoje galės rasti visus reikiamus konpektus bei pasidalinti turimais konspektais su kitais studentais ir taip padėti mokytis kitiems.

### 1.3. VIDINĖ VERSLO PROCESO ANALIZĖ

Toliau pateikiama keliais aspektais atlikta vidinė verslo proceso analizė, kuria siekiama nustatyti nagrinėjamo verslo stiprybes ir silpnybes, kylančias iš verslo proceso.

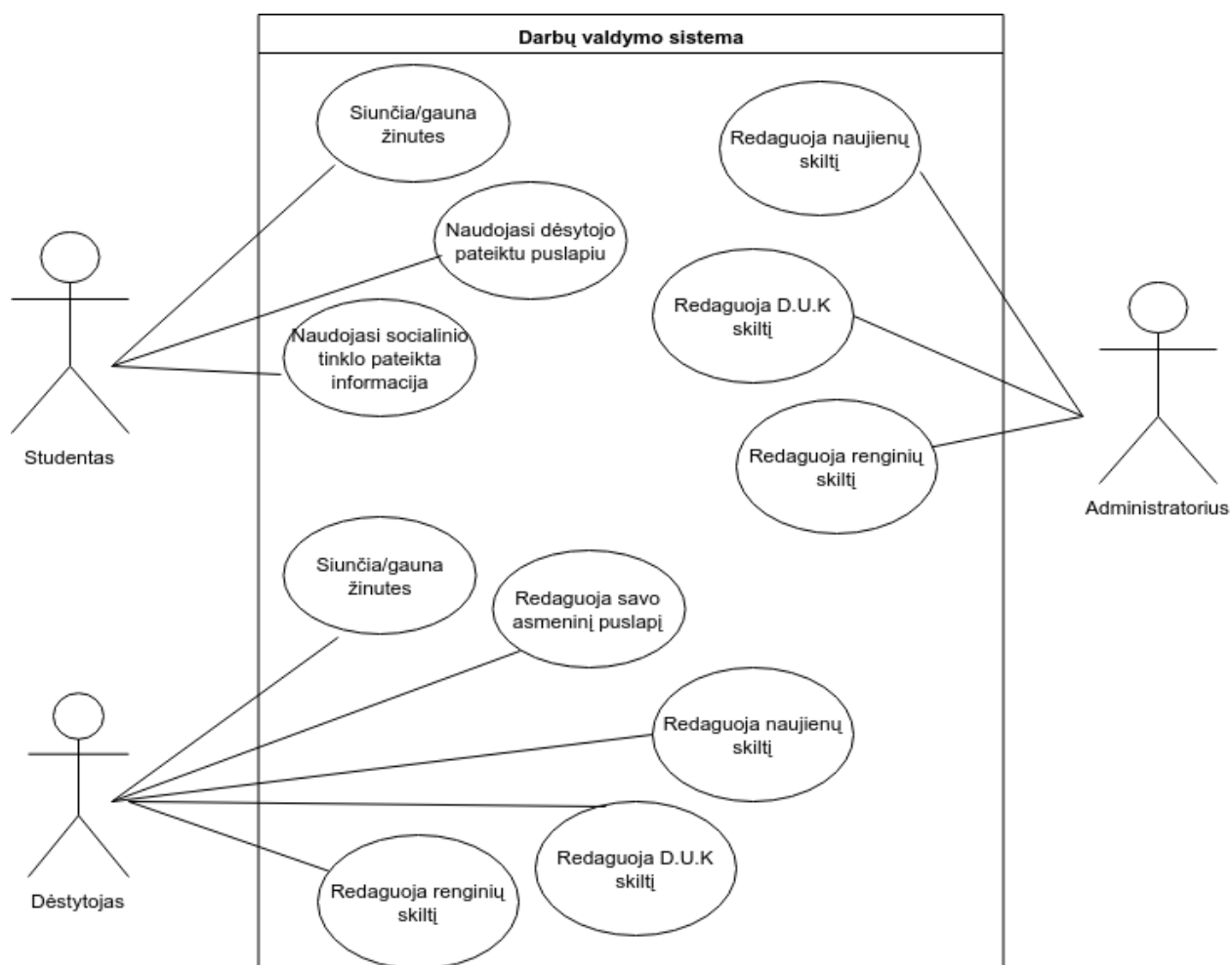
#### 1.3.1. Dalykinės srities statinė struktūra

Toliau pateikiama dalykinės srities statinės struktūros UML diagrama. Joje matomos pagrindinės esybės bei jų tarpusavio sąveika. Klientai yra trijų tipų: studentai, dėstytojai ir administratoriai (dėstytojai bei administratoriai skiriasi turinio redagavimo teisėmis). Studentams, administratoriams ir dėstytojams registruotis nereikia, nes visa vartotojų duomenų bazė yra imama iš bendros VU vartotojų duomenų bazės. Vartotojui tereikia autentifikuoti save - tokiu būdu, priklausomai nuo užimamų pareigų, jam bus suteiktos teisės prie numatyto turinio. Studentai gali tik skaityti tokius puslapius kaip "Dėstytojai", "Naujienos", "Renginiai", "D.U.K", "Konspektai". Tuo tarpu dėstytojai skiltyje "Naujienos" gali parašyti naujieną. Be to, jiems suteikta prieiga prie jų profilio, kur turi teisę redaguoti savo asmeninį puslapį, siųsti bei gauti žinutes, pridėti organizuojamą renginį. Administratoriumi gali būti fakulteto paskirtas žmogus, atsakingas už tikslingo turinio formavimą vidiniame socialiniame tinkle arba VU Studentų atstovybės narys, kuris turi teisę redaguoti visą turinį. Tačiau joks administratorius negali redaguoti dėstytojo sukurtą turinio.



1 pav. Dalykinė srities UML diagrama

### 1.3.2. Užduotys



2 pav. Užduočių veiklos UML diagrama

Užduočių sąrašas pagal 2 pav.:

**Užduotis:** sukurti naujieną.

**Tikslas:** publikuoti naujieną, kuri būtų įdomi.

**Trigeris:** administratoriaus arba dėstytojo kuriama naujiena tituliniui.

**Prioritetas:** aukštas.

**„Prieš“ sąlygos:** naujiena bus skaitoma.

**Sėkmingos baigties „po“ sąlyga:** naujiena perskaitoma ir įsisavinama nauja informacija.

**Nesėkmingos baigties sąlyga:** naujiena visiškai nėra skaitoma.

**Pirminis agentas:** administratorius.

**Antriniai agentai:** studentas.

**Užduotis:** redaguoti puslapį.

**Tikslas:** redaguoti dėstytojo puslapį.

**Trigeris:** prisijungęs dėstytojas gali redaguoti puslapį.

**Prioritetas:** aukštas.

**„Prieš“ sąlygos:** dėstytojas patogiai talpina informaciją.

**Sėkmingos baigties „po“ sąlyga:** dėstytojo informacija sudėliota pagal šabloną ir patogiai naudojimi.

**Nesėkmingos baigties sąlyga:** dėstytojas nesudaro savo puslapio.

**Pirminis agentas:** dėstytojas.

**Antriniai agentai:** studentas.

**Užduotis:** sukurti renginį.

**Tikslas:** publikuoti renginį, kuris būtų naudingas.

**Trigeris:** administratoriaus arba dėstytojo sukurtas renginys atsiranda puslapyje.

**Prioritetas:** aukštas.

**„Prieš“ sąlygos:** renginį aplankys daug žmonių.

**Sėkmingos baigties „po“ sąlyga:** renginys pateisino savo lūkesčius.

**Nesėkmingos baigties sąlyga:** apie renginį studentai sužinojo ne iš socialinio tinklo.

**Pirminis agentas:** administratorius.

**Antriniai agentai:** studentas.

**Užduotis:** parašyti žinutę.

**Tikslas:** parašyti žinutę tiesiai vartotojui.

**Trigeris:** studentas, paspaudęs ant dėstytojo, gali parašyti jam žinutę.

**Prioritetas:** aukštas.

**„Prieš“ sąlygos:** vartotojas gaus žinutę.

**Sėkmingos baigties „po“ sąlyga:** vartotojai patogiai komunikuos tarpusavy.

**Nesėkmingos baigties sąlyga:** žinutė liks neperskaityta.

**Pirminis agentas:** studentas.

**Antriniai agentai:** administratorius.

**Užduotis:** apsilankyti asmeniniame puslapyje.

**Tikslas:** pasiekti dėstytojo asmeninį puslapį.

**Trigeris:** studentų apsilankymas dėstytojų profilyje.

**Prioritetas:** aukštas.

**„Prieš“ sąlygos:** vartotojai lankysis dėstytojų puslapiuose per socialinį tinklą.

**Sėkmingos baigties „po“ sąlyga:** vartotojams bus patogiu pasiekti puslapį ir sužinoti informaciją.

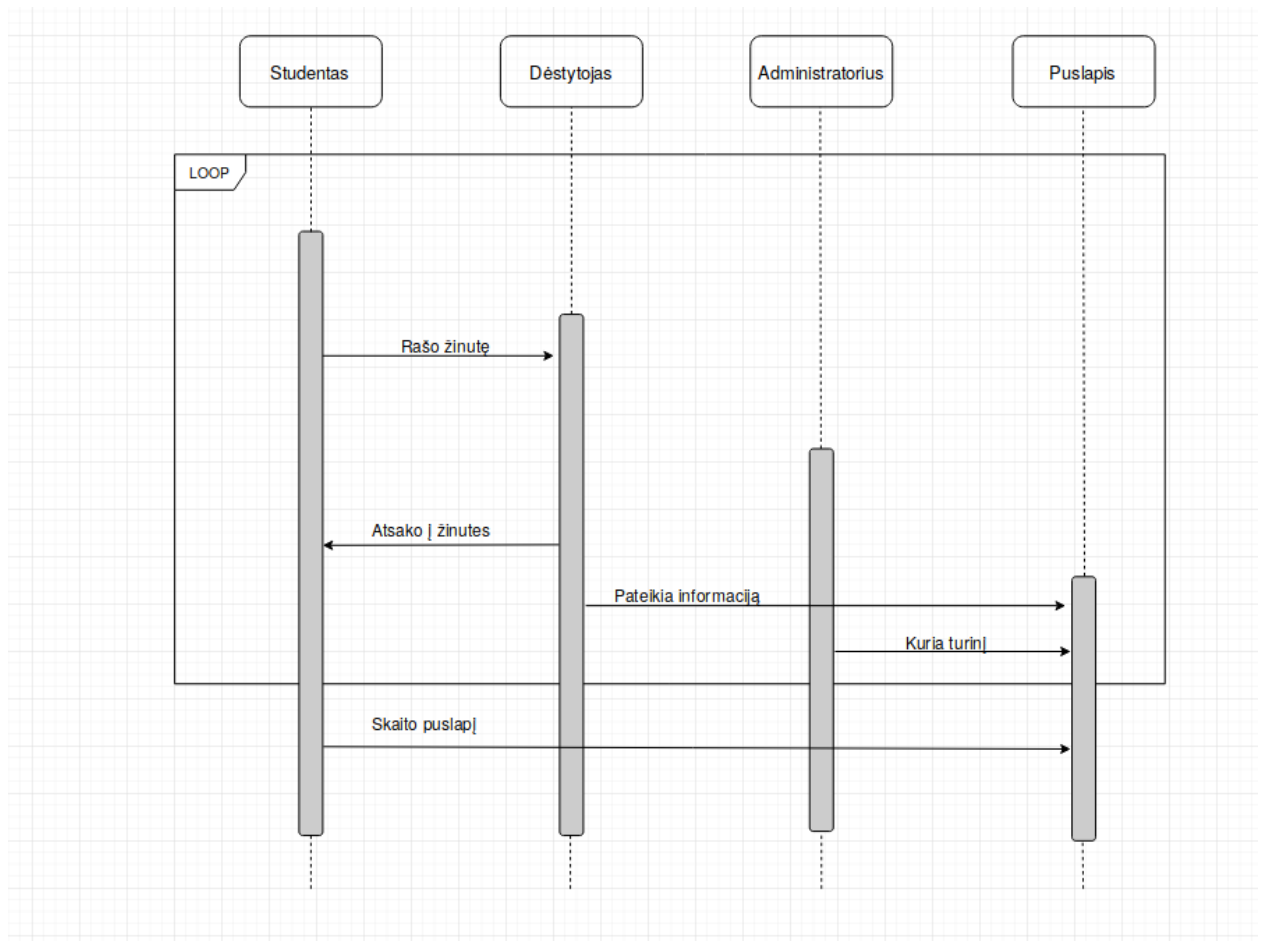
**Nesėkmingos baigties sąlyga:** vartotojai toliau ieškos puslapių naudojantis "Google".

**Pirminis agentas:** studentas.

**Antriniai agentai:** administratorius.

### 1.3.3. Užduočių vykdymo scenarijai

3 pav. paveikslėlyje pavaizduota esybių diagrama. Pateikiamos pagrindinės dalykinės srities esybės, pavaizduoti ryšiai tarp jų.

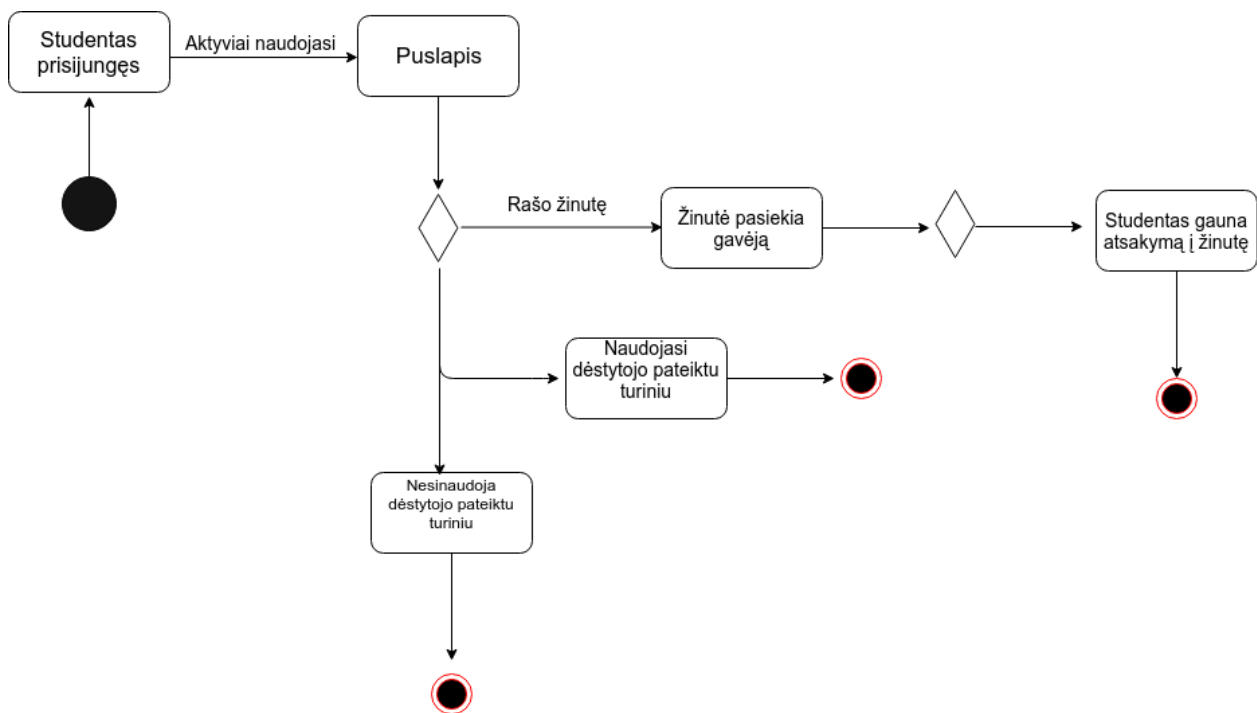


3 pav. Klasių diagrama

3 pav. vaizduojamas socialinio tinklo naudojimosi modelis. Studentas gali rašyti dėstytojui žinutę, o dėstytojas jam gali atsakyti. Taip pat studentas skaito puslapį, kurį pats dėstytojas ir sukūrė. Administratorius kuria turinį, į kurį įeina: renginiai, naujienos, D.U.K.

#### 1.3.4. Dalykinės srities dinaminė struktūra

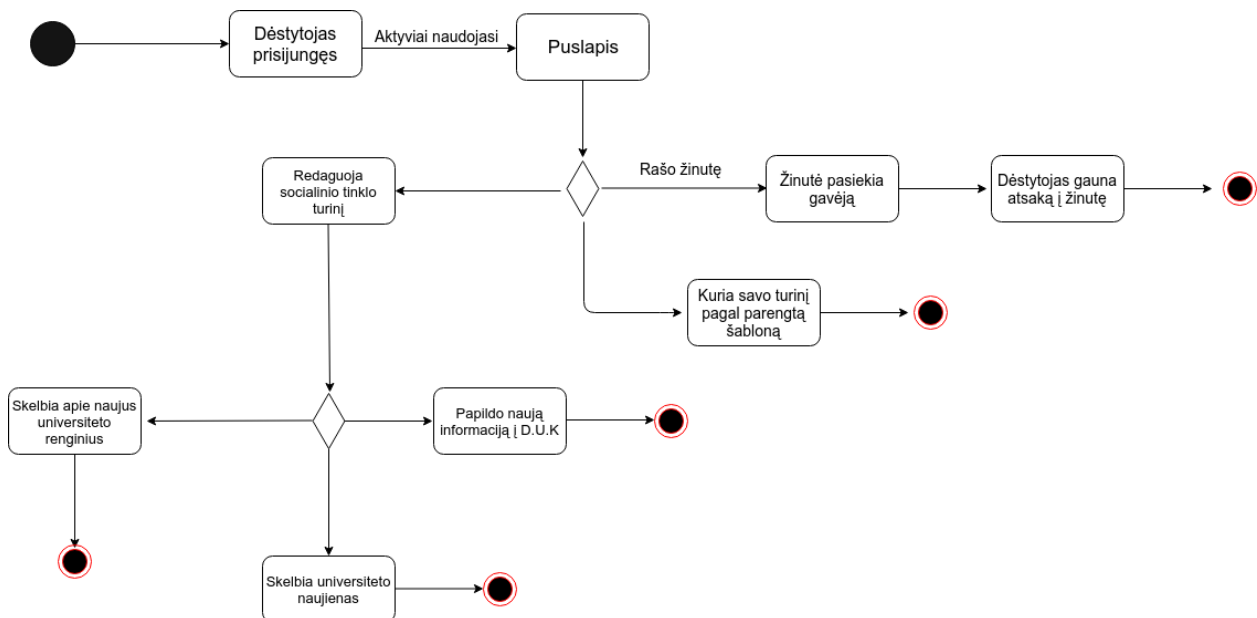
Studentas:



4 pav. Studento UML klasių diagrama

4 pav. Prisijungęs studentas turi galimybę matyti bendrą turinį. Vėliau gali parašyti žinutę dėstytojui arba naudoti dėstytojo pateiktą turinį. Jie tarpusavyje gali komunikuoti socialinio tinklo pagalba, nenaudojant trečiųjų šalių servisų.

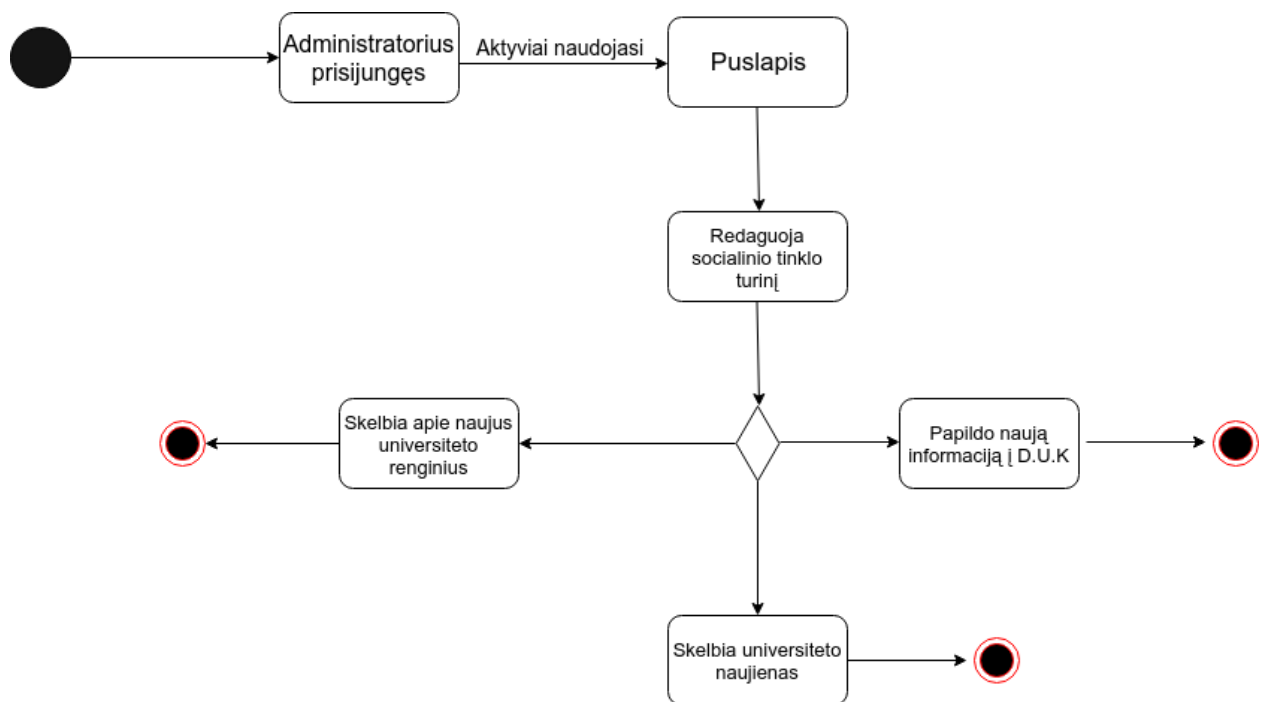
Dėstytojas:



5 pav. Dėstytojo UML klasių diagrama

5 pav. Prisijungęs dėstytojas turi galimybę redaguoti savo puslapio turinį, pagal iš anksto paruoštą šablono. Dar turi galimybę rašyti bei atsakyti į studentų gautas žinutes. Naujienų rašymas, renginių kūrimas, D.U.K skilties papildymas irgi įeina į aukščiau aprašomo vartotojo galimybes.

Administratorius:



6 pav. Administratoriaus UML klasių diagrama

6 pav. Prisijungęs administratorius, tai žmogus, kuris atsakingas už bendrą socialinio tinklo turinį, į kurį neįeina dėstytojų puslapių redagavimas. Administratorius gali kurti bei redaguoti tik naujienų, D.U.K ir renginių skiltis.

## 1.4. ĮGYVENDINAMUMO IR NAUDOS ANALIZĖ

Šiame skyriuje pateikiama informacija, kuri padeda nustatyti, ar darbų valdymo sistemą įmanoma sukurti, kokią konkrečią naudą ji duos, kokios problemos gali kilti ir kaip jos bus sprendžiamos.

### 1.4.1. Operacinis įgyvendinamumas

Šiame skyriuje pateikiami galimi trukdžiai, kurie gali kilti naudantis sistema, bei pateikiami galimi jų sprendimo būdai.

**Problema:** Pasikartojančių konspektų bei kitos dalykinės informacijos perteklius.

**Sprendimo būdas:** Studentų atstovybės paskirtas žmogus nuolat tikrina konspektų naudą arba prieš įkeliant konspektą, studentas turi gauti patvirtinimą.

**Problema:** Universiteto bendruomenė nesinaudoja mūsų sistema t.y. dėstytojai toliau pateikia savo asmeninius puslapius per senąsias nuorodas, studentai ieško informacijos kituose šaltiniuose.

**Sprendimo būdas:** Prašyti studentų atstovybės pagalbos, kad informuotų bendruomenę apie naudojimosi galimybes bei teikiamą naudą.

**Problema:** internetinis puslapis dažnai patiria sistemos perkrovą.

**Sprendimo būdas:** Naudoti galingesnius serverius, kurie sugeba apdoroti didesnius vartotojų srautus.

### 1.4.2. Techninis įgyvendinamumas

Šiuo metu jau egzistuoja atskiri Vilniaus universiteto informacinės sistemos komponentai (el. paštas, VU informacinė sistema, fakultetų puslapiai, dėstytojų asmeniniai puslapiai). Mūsų kuriama sistema visus šiuos komponentus apjungtų į vieną bendrą sistemą. Vienintelis reikalavimas vartotojams - interneto prieiga.

### 1.4.3. Ekonominis įgyvendinamumas

#### Išlaidos

Prognozuojama sistemos sukūrimo kaina - 8960€. Daroma prielaida, jog sistemos sukūrimas truks 16 savaičių, jei prie sistemos kūrimo puse etato dirbs keturių žmonių komanda (kiekvienam asmeniui mokamas 7€/h atlyginimas).

Sistemos palaikymo kaina - 2184€/mėn. Daroma prielaida, jog sistemą prižiūri du pilnu etatu už 6€/h darbo užmokestį dirbantys sistemos administratoriai, serverio nuoma kainuoja 139€/mėn, o reklamos išlaidos sudaro 125€ per mėnesį.

Kitos išlaidos pateikiamos sekančioje lentelėje.



5 lentelė. Išlaidos reikalingos sistemos sukūrimui ir palaikymui

Išlaidos	Kaina
Sistemos sukūrimas	8960€
Domeno registracija	10€
SSL sertifikatas	155.55€
Serverio nuoma	139€ mėnesiui
Administratoriai	1920€ mėnesiui
Reklama	125€ mėnesiui

### **Pajamos**

Pagrindinis kuriamos sistemos pajamų šaltinis - konspektų dalinimosi platforma. Darant prielaidą, jog iš šiuo metu esančių 20000 Vilniaus universiteto studentų 3 procentai jų nuspręs patalpinti konspektus į konspektų dalinimosi platformą pardavimui už vidutinę 5€ kainą (20 procentų pelno lieka platformos savininkams) ir kiekvieno studento patalpintas egzempliorius bus nuperkamas vidutiniškai kartą į savaitę, metinės sistemos pajamos turėtų siekti apie 31284€.

### **Pelnas/nuostolis**

Metinės sistemos palaikymo išlaidos turėtų sudaryti 26208€. Atsižvelgiant į sukūrimo kaštus, sistema turėtų pradėti generuoti pelną po 1.76 metų. Tikėtinas metinis pelnas - 5076€.

#### **1.4.4. Juridinis įgyvendinamumas**

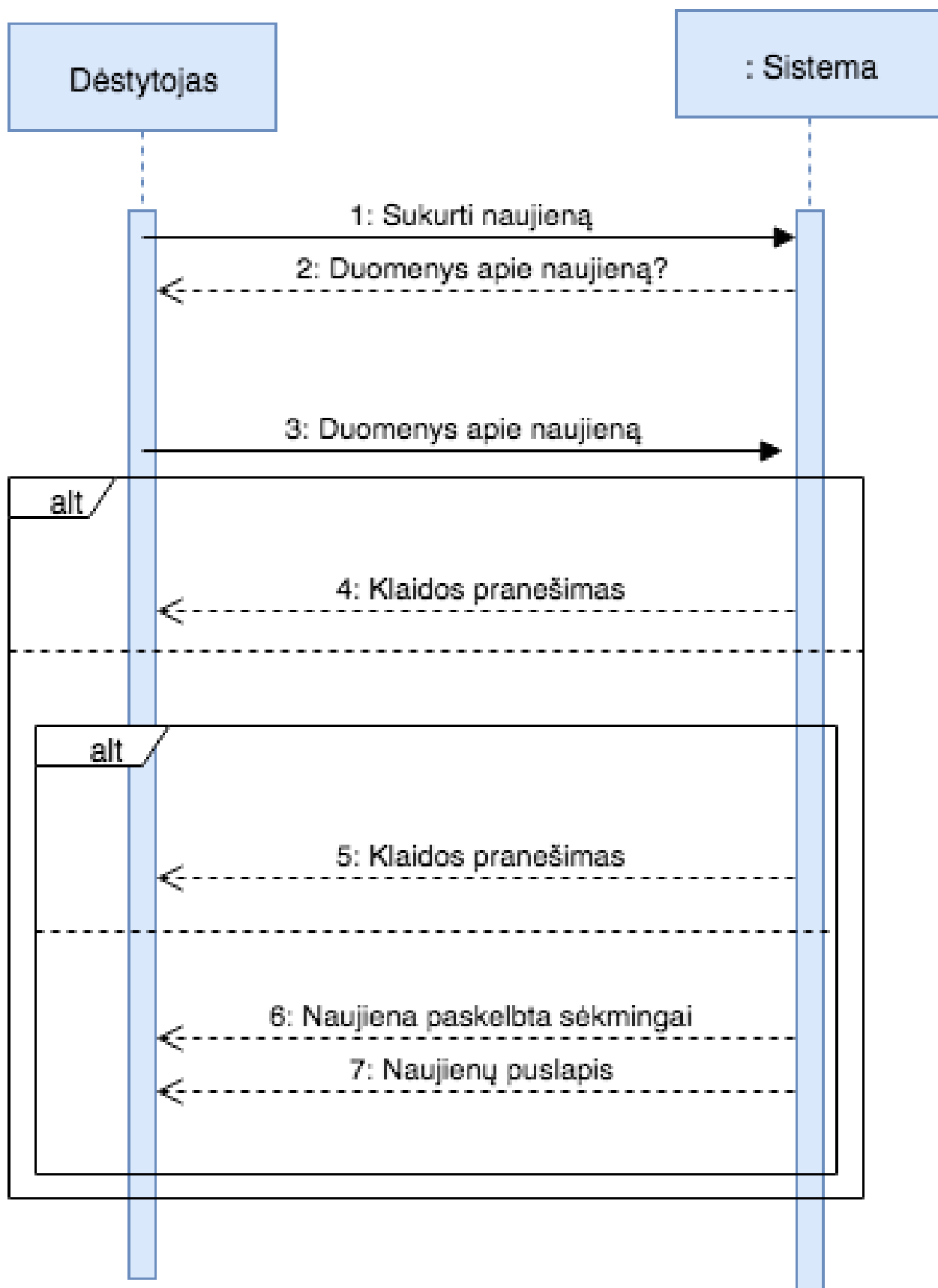
Sistema nepažeis Lietuvos respublikos įstatymų, Europos Sąjungos direktyvų bei Lietuvos respublikos konstitucijos. Vartotojų duomenys bus saugomi, tvarkomi bei perduodami naudojantis užšifruota sąsaja laikantis asmenų duomenų apsaugos įstatymo. Vartotojų talpinamų konspektų autorinės teisės bus saugomos remiantis Lietuvos Respublikos autorių teisių ir gretutinių teisių įstatymu.

## **1.5. SISTEMOS NAUDOJIMO SCENARIJUS**

Šiame skyriuje aprašomas socialinio tinklalapio naudojimo scenarijus.

### **1.5.1. Scenarijus**

Šiame skyriuje pateikiami pagrindinių funkcijų modeliai, kurie parodo, kaip pagrindiniai sistemos agentai, šiuo atveju studentai, dėstytojai ir administratoriai, naudosis sistema. Tam naudojamos UML sekų diagramos.



7 pav. Užduoties įrašyti naujieną modelis

**Užduotis:** Įrašyti naujieną

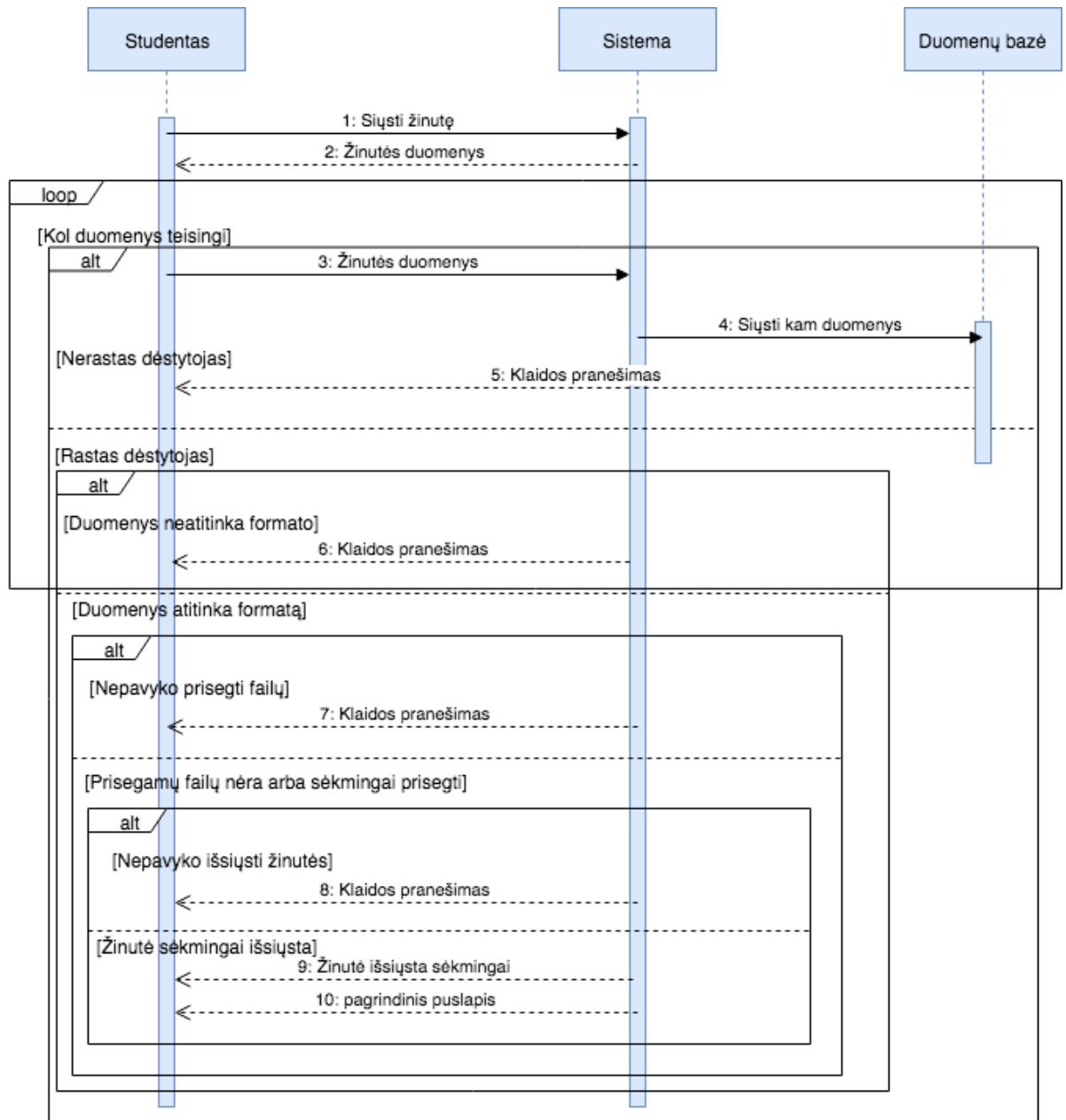
**Tikslas:** paskelbti aktualią naujieną.

**Pirminis agentas:** dėstytojas.

**„Prieš“ sąlyga:** naudotojas yra prisijungęs tinklalapyje, dėstytojo aplinkoje.

**„Po“ sąlyga:** aktuali naujiena yra patalpinta tinklalapyje.

**Scenarijus:** naudotojas, prisijungęs kaip dėstytojas, pasirenka naujienos sukūrimo puslapį ir gražintoje formoje suveda norimą paskelbti informaciją bei pasirenka, kam ši naujiena yra skirta (studentams, dėstytojams, visiems). Jeigu naujiena neatitinka formato ar yra tuščia, sistema įspėja dėstytoją ir leidžia pakeisti duomenis. Jeigu duomenys atitinka formatą, dėstytojas yra informuojamas apie sėkmingą naujienos patalpimą pagrindiniame puslapyje.



8 pav. Užduoties siųsti žinutę modelis

**Užduotis:** studentui išsiųsti žinutę dėstytojui

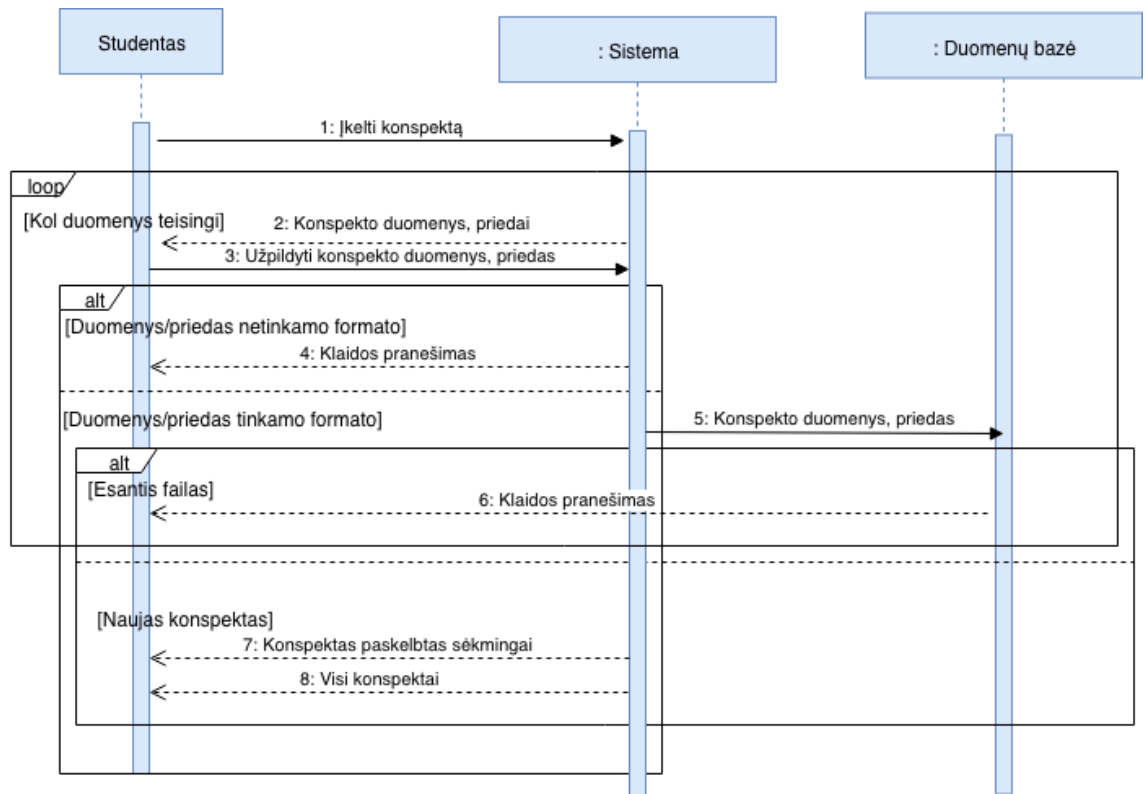
**Tikslas:** studento išsiųsta žinutė pasiekia dėstytoją

**Pirminis agentas:** studentas

**„Prieš“ sąlyga:** naudotojas yra prisijungęs tinklalapyje, studento aplinkoje.

**„Po“ sąlyga:** studento žinutė yra išsiųsta

**Scenarijus:** naudotojas, prisijungęs kaip studentas, pasirenka puslapį siųsti žinutę. Gražintoje formoje pasirenka dėstytoją(us), kuriam(iem) ši žinutė yra skirta. Užpildo formoje prašomus duomenis (tema, žinutė), prisega failus, jeigu reikia. Jeigu žinutė neatitinka formato, sistema išpėja studentą ir leidžia pakeisti duomenis. Jeigu bandomų pridėti failų napavyko prisegti, studentas išpėjamas. Jeigu žinutė atitinka formatą ir failai sėkmingai prisegti, studentas informuojamas, jog žinutė išsiųsta sėkmingai.



9 pav. Užduoties įkelti konspektą modelis

**Užduotis:** studentui įkelti konspektą

**Tikslas:** studento įkeltas konspektas bus prieinamas kitiems studentams

**Pirminis agentas:** studentas

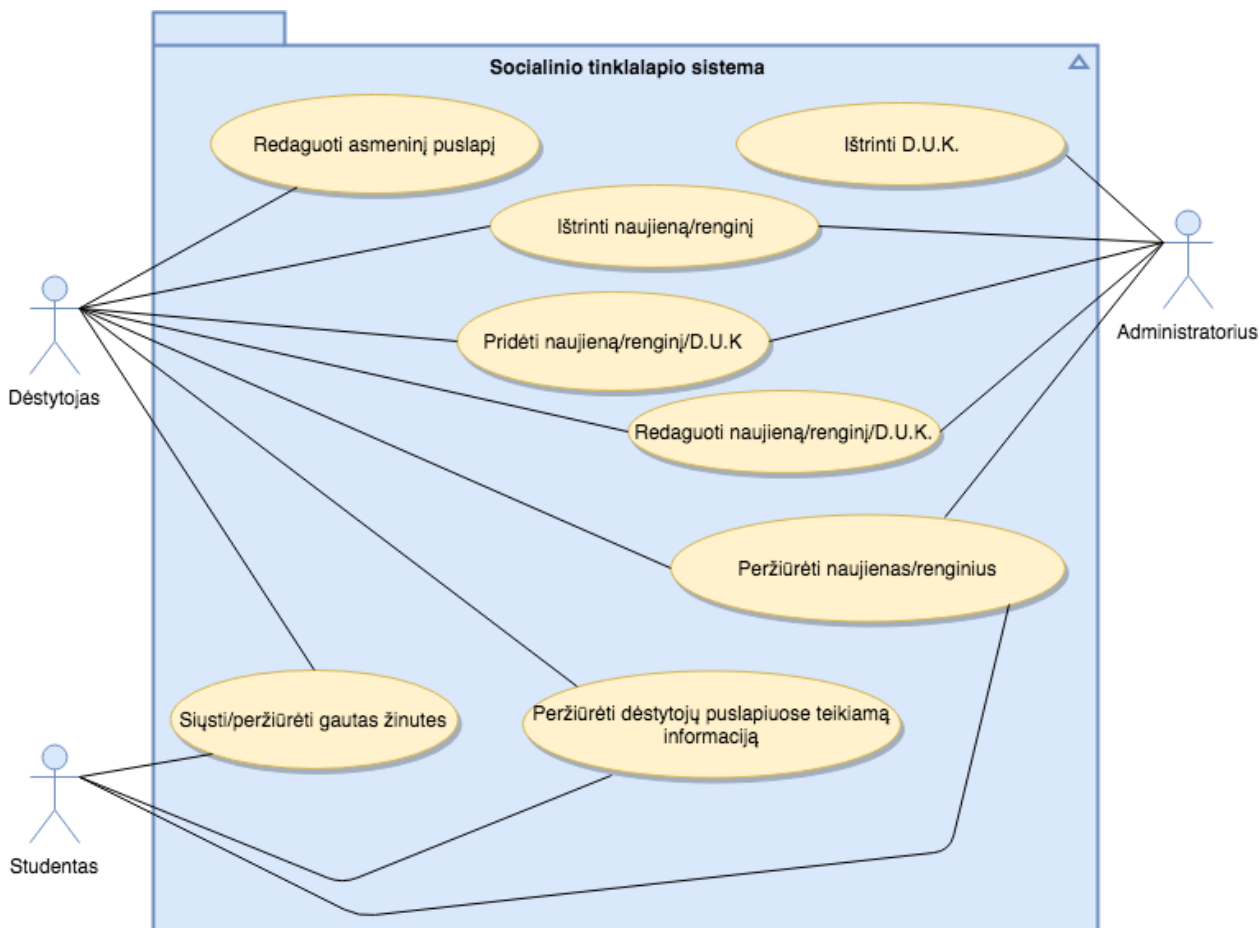
**„Prieš“ sąlyga:** naudotojas yra prisijungęs tinklalapyje, studento aplinkoje.

**„Po“ sąlyga:** studento konspektas yra sėkmingai įkeltas

**Scenarijus:** naudotojas, prisijungęs kaip studentas, pasirenka įkelti konspektą. Gražintoje formoje užpildo duomenis apie konspektą, prisega atitinkamą failą. Užpildo formoje prašomus duomenis (tema, dėstytojas, ...), prisega failą(-us). Jeigu duomenys/priedas neatitinka formato, sistema išpėja studentą ir leidžia pakeisti duomenis. Jeigu bandomų pridėti failų napavyko įkelti, studentas išpėjamas. Jeigu duomenys atitinka formatą ir failai sėkmingai prisegti, studentas informuojamas, jog konspektas įkeltas sėkmingai.

### 1.5.2. Sistemos teikiama nauda

Šiame skyriuje nagrinėjamos užduotys, kurias gali atlikti naudotojai. Tam pavaizduoti naudojama UML užduočių diagrama, kurioje agentai yra mūsų sistemos naudotojai - studentai ir dėstytojai.



10 pav. Užduočių diagrama

Sistemoje vykdomos pagrindinės užduotys: studentas gali peržiūrėti pasirinkto dėstytojo puslapius. Dėstytojams sudaroma galimybė pateikti aktualias naujienas, informaciją apie renginius, redaguoti savo asmeninius tinklalapius, kuriuose gali talpinti informaciją apie savo dėstomus dalykus bei kitą naudingą informaciją, kurią matys jų studentai. Tiek dėstytojai, tiek studentai gali gauti informaciją apie juos dominančius renginius, matyti aktualias naujienas, gauti bei siųsti žinutes. Studentams suteikiama galimybė dalintis konspektais (10 pav.).

### 1.5.3. Esama būklė

Šiuo metu visi komandos nariai turi asmeninius kompiuterius kuriais gali kurti ir testuoti sistemą. Nariai taip pat turi prieigą prie Android ir iOS išmaniųjų telefonų, kurių prireiks tinklapio testavimui mobiliuose įrenginiuose. Grupė sudaryta iš keturių asmenų, kurie yra įgiję reikalingas žinias sistemos sukūrimui ir palaikymui.

#### **1.5.4. Priemonės scenarijui įgyvendinti**

Norint sukurti socialinį tinklalapį bus reikalinga:

1. Domeno registracija
2. Serveris
3. SSL sertifikatas

## 2. REIKALAVIMŲ SPECIFIKAVIMAS

### 2.1. FUNKCINIAI REIKALAVIMAI

Šiame skyriuje pateikiami funkciniai reikalavimai – nagrinėjami scenarijai, ką sistema turi daryti, kaip elgtis vienu ar kitu atveju. Apibrėžiant funkcinius reikalavimus naudojamos procesų sekų diagramos, sistemoje vykdomų užduočių diagramos.

#### 2.1.1. Internetinės svetainės langai

6 lentelė. Funkciniai reikalavimai. Internetinės svetainės langai.

Kodas	Reikalavimas	Svarba
Internetinės svetainės langai		
FR 1.1	Svetainės langas „Prisijungimas“ turi būti matomas visiems net ir neprisijungusiems naudotojams.	Būtina
FR 1.2	Svetainės langai „Pagrindinis puslapis“, „Dėstytojai“, „Dėstytojo puslapis“, „D.U.K.“, „Renginiai“, „Žinutės“, „Mano profilis“, „Renginio informacija“ turi būti matomi visiems prisijungusiems naudotojams.	Būtina
FR 1.3	Svetainės langas „Konspektai“ turi būti matomas visiems prisijungusiems studentams bei sistemos administratoriams.	Būtina
FR 1.4	Svetainės langas „Dėstytojo puslapio redagavimas“ turi būti matomas visiems prisijungusiam puslapį sukūrusiam dėstytojui.	Būtina

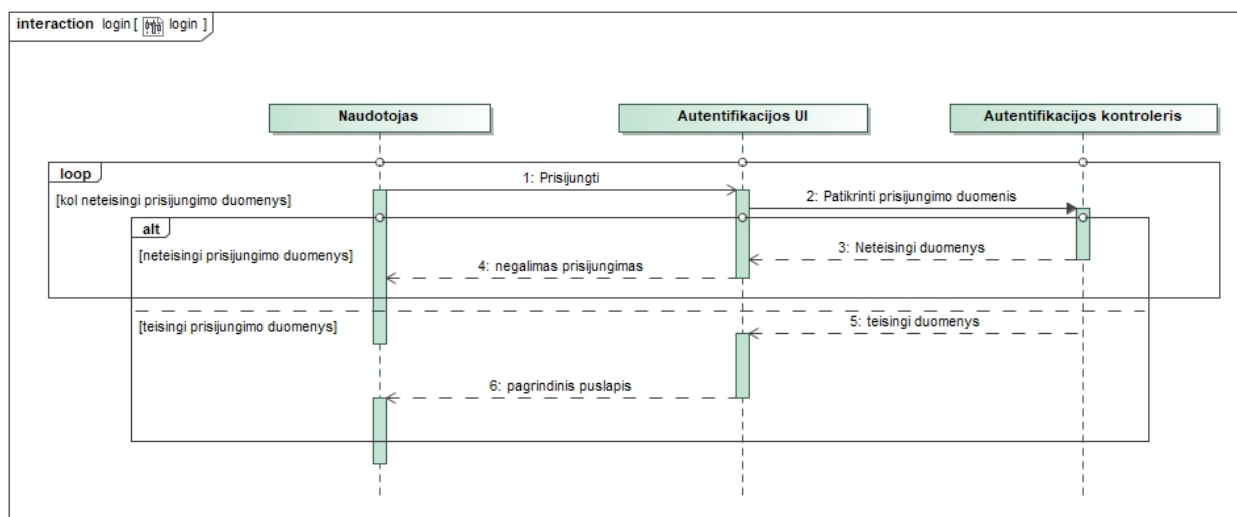
#### 2.1.2. Prisijungimas

7 lentelė. Funkciniai reikalavimai. Prisijungimas.

Kodas	Reikalavimas	Svarba
Prisijungimas		
FR 2.1	Naudotojui suvedus tinkamus prisijungimo duomenis jis turi būti prijungiamas prie sistemos.	Būtina
FR 2.2	Naudotojui netinkamai įvedus prisijungimo duomenis jis neturi būti prijungiamas prie sistemos ir turi būti išmetamas klaidos pranešimas.	Būtina
FR 2.3	Turi būti galimybė užmiršus slaptažodį gauti naują slaptažodį į el. paštą.	Pageidaujama
FR 2.4	Naudotojų bandymų prisijungti prie sistemos skaičius neturi būti ribojamas.	Būtina

44 pav. pateikiama užduoties „Prisijungimas“ sekų diagrama. Joje vaizduojamas pagrindinis prisijungimo prie sistemos scenarijus ir taip pat nagrinėjami alternatyvūs scenarijai.





11 pav. Proceso „Prisijungimas“ sekų diagrama

### 2.1.3. Atsijungimas

8 lentelė. Funkciniai reikalavimai. Atsijungimas.

Kodas	Reikalavimas	Svarba
Atsijungimas		
FR 3.1	Naudotojas paspaudęs mygtuką atsijungti turi būti atjungiamas nuo sistemos.	Būtina
FR 3.2	Atsijungus nuo sistemos ir bandant paspausti grįžimo mygtuką naudotojas turi būti nukreipiamas į prisijungimo langą.	Būtina

### 2.1.4. Paskyros valdymas

9 lentelė. Funkciniai reikalavimai. Paskyros valdymas.

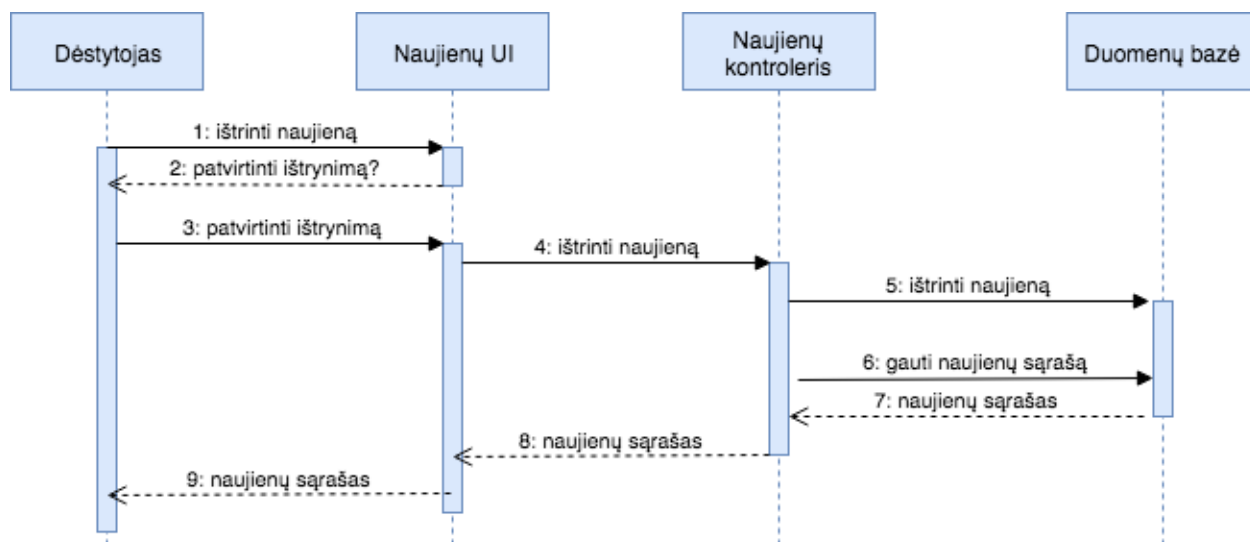
Kodas	Reikalavimas	Svarba
Paskyros valdymas		
FR 4.1	Naudotojui paspaudus mygtuką „Mano profilis“ turi būti matoma visa žinoma informacija apie naudotoją.	Būtina
FR 4.2	Naudotojui paspaudus mygtuką „Pakeisti slaptažodį“, įvedus tinkamą seną ir naują slaptažodžius bei paspaudus mygtuką „Patvirtinti“ slaptažodis turi būti pakeičiamas į naują.	Būtina
FR 4.3	Naudotojui paspaudus mygtuką „Pakeisti slaptažodį“ ir įvedus netinkamą seną slaptažodį turi būti išvedamas klaidos pranešimas.	Būtina
FR 4.4	Naudotojui paspaudus mygtuką „Pakeisti slaptažodį“ ir įvedus netinkamo formato naują slaptažodį turi būti išvedamas klaidos pranešimas ir slaptažodis neturi būti pakeičiamas nauju.	Būtina

## 2.1.5. Naujienos

10 lentelė. Funkciniai reikalavimai. Naujienos.

Kodas	Reikalavimas	Svarba
Naujienos		
FR 5.1	Naujienos turi būti matomos pagrindiniame tinklalapio puslapyje.	Būtina
FR 5.2	Visas naujienų sąrašas turi būti pateikiamas viename puslapyje.	Būtina
FR 5.3	Naujienos turi būti rikiuojamos pagal naujienos paskelbimo datą (naujausios turi būti matomos pirmos).	Būtina
FR 5.4	Jei naujienų sąrašas tuščias turi būti rodomas pranešimas, kad naujienų šiuo metu nėra.	Būtina
FR 5.5	Naujienos turi būti matomos visiems prisijungusiems naudotojams.	Būtina
FR 5.6	Naujienos redagavimo funkcija turi būti matoma bei panaudojama tik naujieną paskelbusiam naudotojui.	Būtina
FR 5.7	Skelbti naujienas turi būti galimybė tik prisijungusiems dėstytojams bei sistemos administratoriams.	Būtina
FR 5.8	Naujienos ištrynimo funkcija turi būti matoma bei panaudojama tik naujieną paskelbusiam naudotojui bei sistemos administratoriui.	Būtina

12 pav. pateikiama užduoties „Naujienos ištrynimas“ sekų diagrama. Joje vaizduojamas pagrindinis naujienos ištrynimo scenarijus.



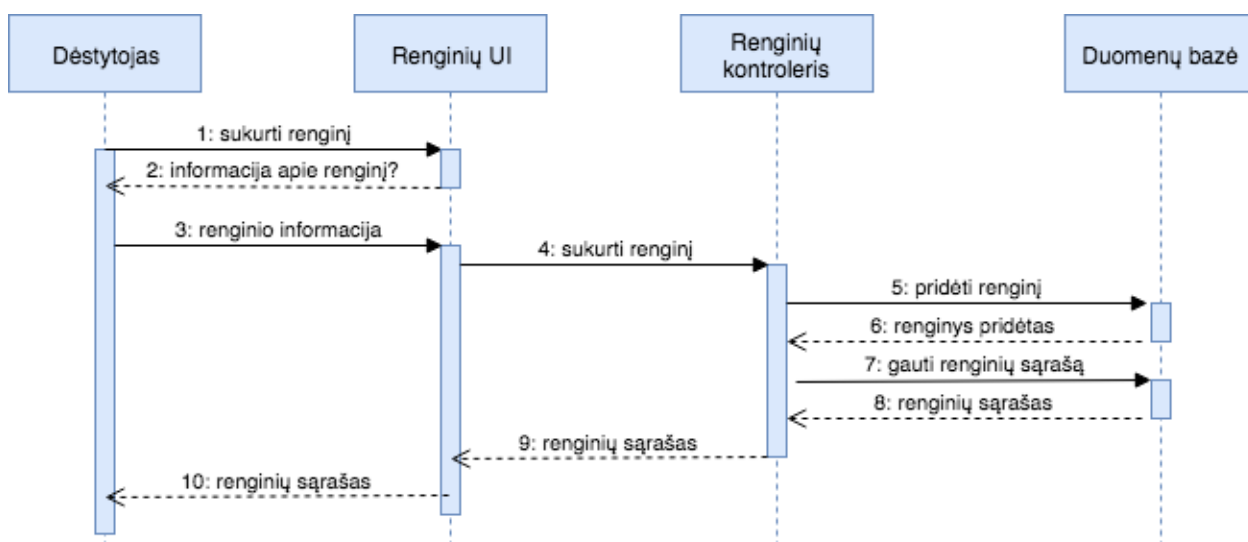
12 pav. Proceso „Naujienos ištrynimas“ sekų diagrama

## 2.1.6. Renginiai

11 lentelė. Funkciniai reikalavimai. Renginiai.

Kodas	Reikalavimas	Svarba
Renginiai		
FR 6.1	Renginiai turi būti matomi puslapyje „Renginiai“.	Būtina
FR 6.2	Visas renginių sąrašas turi būti pateikiamas kalendoriuje esančiame puslapyje „Renginiai“.	Būtina
FR 6.3	Paspaudus ant kalendoriuje pateikto renginio turi atsirasti puslapis „Renginio informacija“, kuriame renginys turi būti aprašytas detaliau.	Būtina
FR 6.4	Jei nėra sukurta jokių renginių, puslapyje „Renginiai“ turi būti pateikiamas tuščias kalendorius.	Būtina
FR 6.5	Renginiai turi būti matomi visiems prisijungusiems naudotojams.	Būtina
FR 6.6	Renginio redagavimo funkcija turi būti matoma bei panaudojama tik renginį sukūrusiam naudotojui.	Būtina
FR 6.7	Kurti renginius turi būti galimybė tik prisijungusiems dėstytojams bei sistemos administratoriams.	Būtina
FR 6.8	Renginio ištrynimo funkcija turi būti matoma bei panaudojama tik naujieną paskelbusiam naudotojui bei sistemos administratoriui.	Būtina

13 pav. pateikiama užduoties „Renginio sukūrimas“ sekų diagrama. Joje vaizduojamas pagrindinis renginio sukūrimo scenarijus.



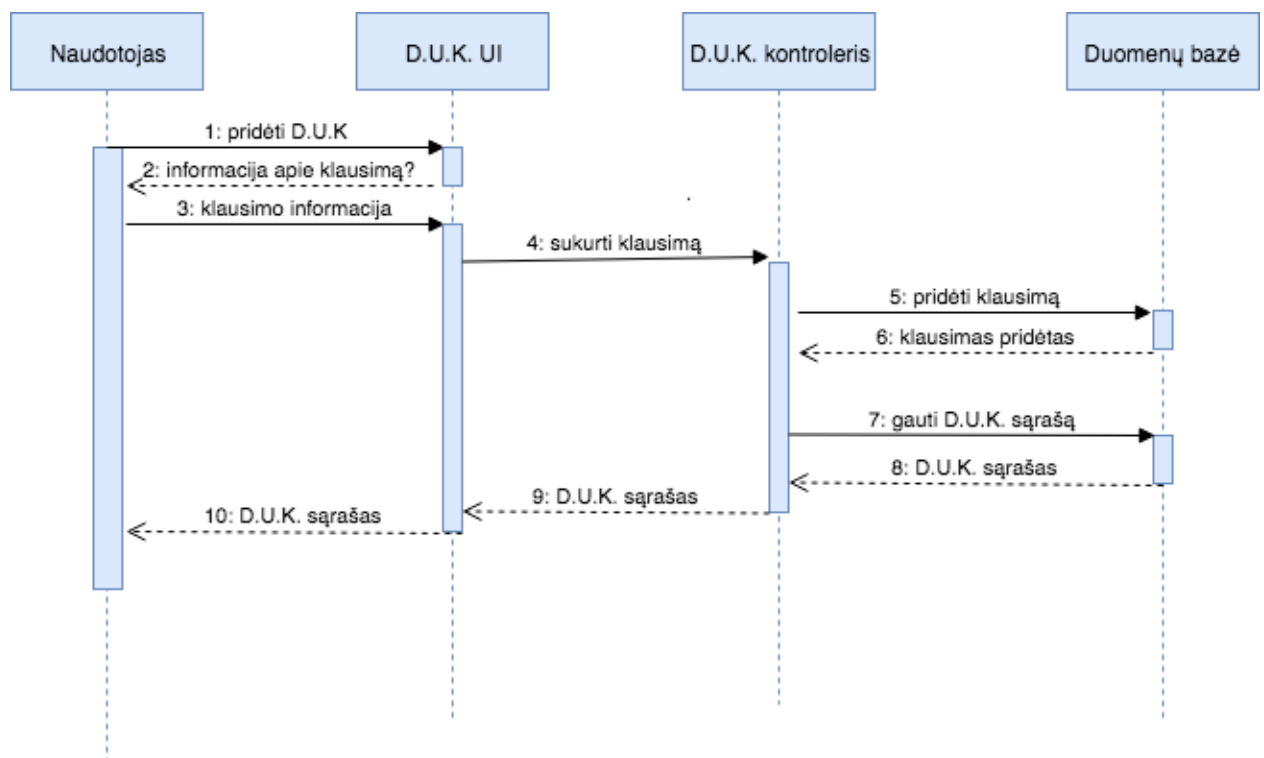
13 pav. Proceso „Renginio sukūrimas“ sekų diagrama

### 2.1.7. D.U.K.

12 lentelė. Funkciniai reikalavimai. D.U.K.

Kodas	Reikalavimas	Svarba
D.U.K.		
FR 7.1	Dažniausiai užduodami klausimai turi būti matomi puslapyje „D.U.K.“.	Būtina
FR 7.2	Visas dažniausiai užduodamų klausimų sąrašas turi būti pateikiamas viename puslapyje.	Būtina
FR 7.3	Paspaudus ant vieno iš pateiktų klausimų turi būti matomas atsakymas į tą klausimą.	Būtina
FR 7.4	D.U.K. turi būti matomi visiems prisijungusiems naudotojams.	Būtina
FR 7.5	D.U.K. redagavimo funkcija turi būti matoma bei panaudojama tik sistemos administratoriams.	Būtina
FR 7.6	Sukurti naują klausimą turi būti galimybė tik sistemos administratoriams.	Būtina
FR 7.7	Klausimo ištrynimo funkcija turi būti matoma bei panaudojama tik sistemos administratoriui.	Būtina

14 pav. pateikiama užduoties „D.U.K.pridėjimas” sekų diagrama. Joje vaizduojamas pagrindinis D.U.K. pridėjimo scenarijus.



14 pav. Proceso „D.U.K.pridėjimas” sekų diagrama

### 2.1.8. Dėstytojų sąrašas

13 lentelė. Funkciniai reikalavimai. Dėstytojų sąrašas

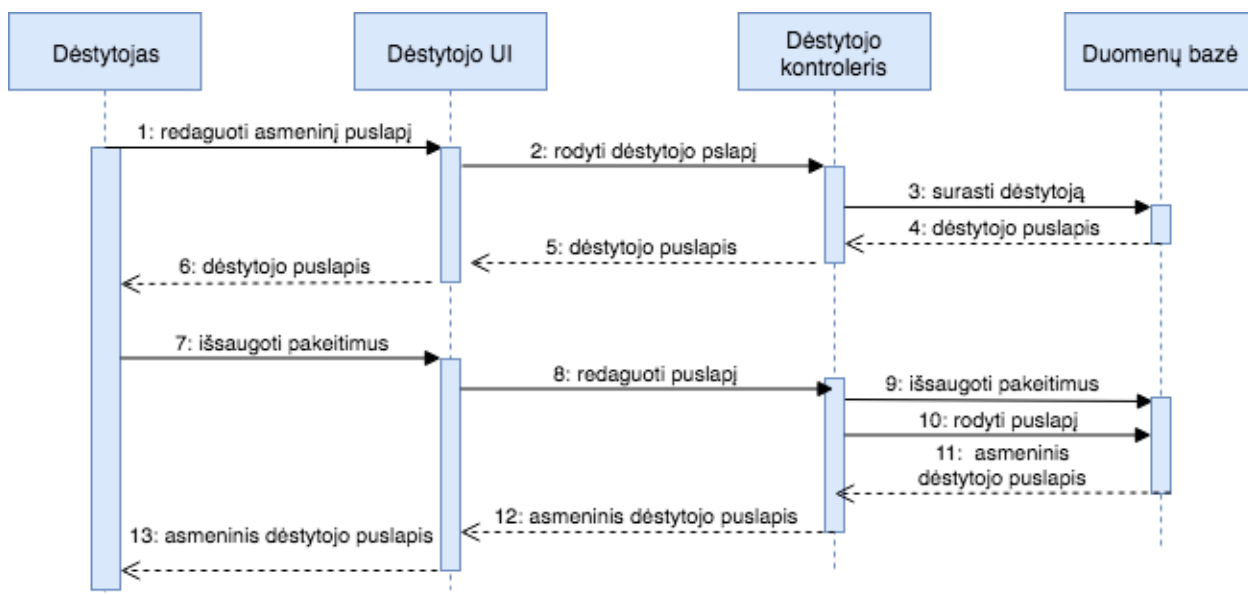
Kodas	Reikalavimas	Svarba
Dėstytojų sąrašas		
FR 8.1	Dėstytojų sąrašas turi būti matomas puslapyje „Dėstytojai“.	Būtina
FR 8.2	Puslapyje „Dėstytojai“ turi būti paieškos laukas, kurio pagalba galima surasti reikiamą dėstytoją.	Būtina
FR 8.3	Visas dėstytojų sąrašas turi būti pateikiamas viename puslapyje.	Būtina
FR 8.4	Dėstytojai turi būti surikiuoti abėcėlės tvarka pagal pavardę ir vardą.	Būtina
FR 8.5	Dėstytojų sąrašas turi būti matomas visiems prisijungusiems naudotojams.	Būtina
FR 8.6	Prie dėstytojo vardo bei pavardės turi būti pateikiamos nuorodos į dėstytojo puslapį bei į žinutės išsiuntimo dėstytojui puslapį.	Būtina

### 2.1.9. Dėstytojo puslapis

14 lentelė. Funkciniai reikalavimai. Dėstytojo puslapis

Kodas	Reikalavimas	Svarba
Dėstytojo puslapis		
FR 9.1	Dėstytojo puslapis turi būti matomas visiems prisijungusiems naudotojams.	Būtina
FR 9.2	Tik prisijungęs dėstytojas turi galimybę sukurti savo puslapį.	Būtina
FR 9.3	Tik prisijungęs dėstytojas sukūręs savo puslapį turi galimybę jį redaguoti.	Būtina
FR 9.4	Dėstytojo puslapyje turi būti nuoroda, kurią paspaudus turi būti galimybė rašyti žinutę pasirinktam dėstytojui.	Būtina

15 pav. pateikiama užduoties „Dėstytojo puslapio redagavimas“ sekų diagrama. Joje vaizduojamas pagrindinis dėstytojo puslapio redagavimo scenarijus.



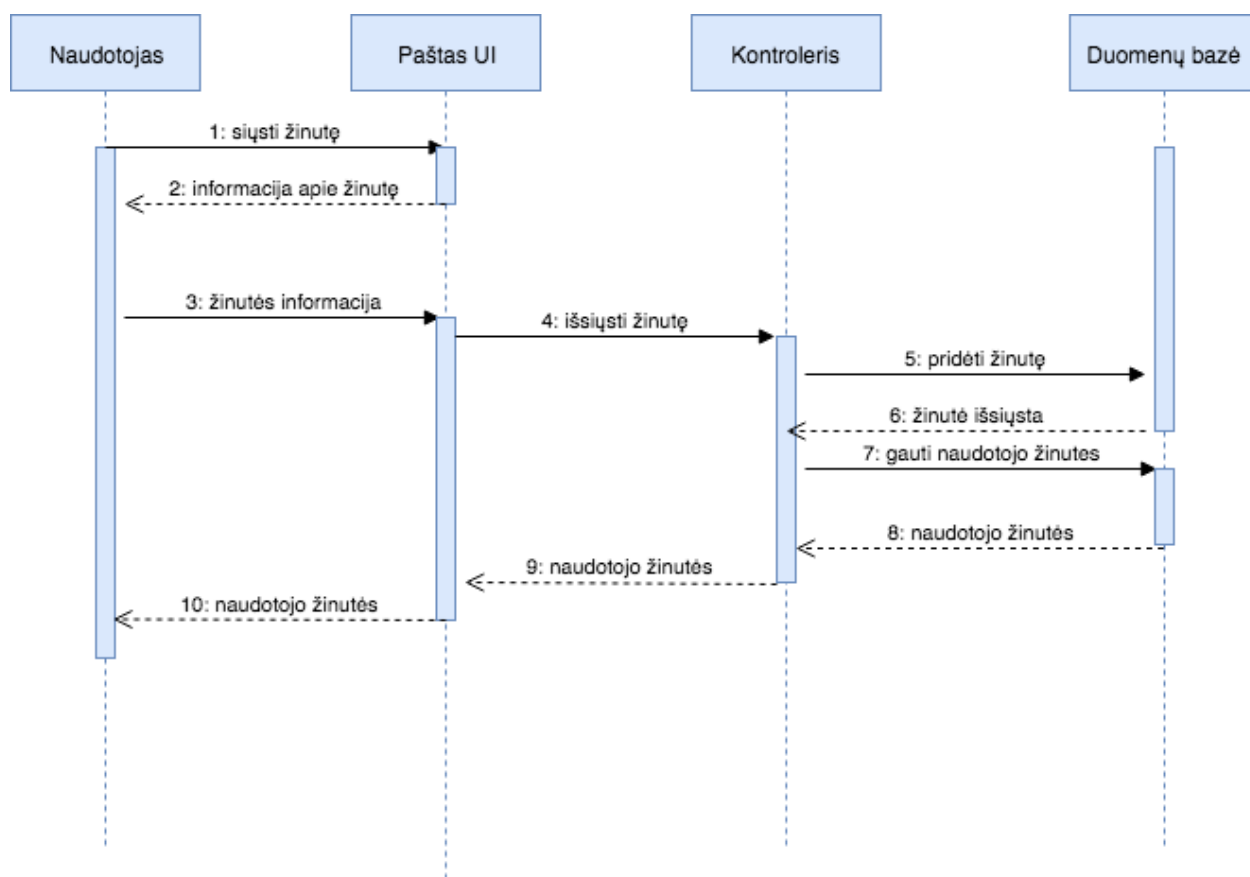
15 pav. Proceso „Dėstytojo puslapio redagavimas“ sekų diagrama

## 2.1.10. Žinutės

15 lentelė. Funkciniai reikalavimai. Žinutės

Kodas	Reikalavimas	Svarba
Žinutės		
FR 10.1	Puslapis „Žinutės“ turi būti matomas visiems prisijungusiems naudotojams.	Būtina
FR 10.2	Puslapyje „Žinutės“ turi būti pateikiamos visos gautos bei išsiųstos žinutės.	Būtina
FR 10.3	Visoms gautoms bei išsiųstoms žinutėms pateikti naudojami puslapiai (viena puslapyje 25 žinutės).	Būtina
FR 10.4	Išsiųsti žinutę turi turėti galimybę visi prisijungę naudotojai.	Būtina
FR 10.5	Dėstytojo puslapyje turi būti nuoroda, kurią paspaudus galima būtų rašyti žinutę pasirinktam dėstytojui.	Būtina
FR 10.6	Paspaudus mygtuką „Išsiųsti žinutę“ turi atsirasti žinutės rašymo langas, kuriame turi būti galimybė pasirinkti naudotoją, kuriam žinutė bus išsiųsta.	Būtina
FR 10.7	Paspaudus žinutės išsiuntimo patvirtinimo mygtuką, žinutė turi būti nusiųsta naudotojui, kuriam ši žinutė turėjo būti išsiųsta.	Būtina
FR 10.8	Turi būti galimybė išsiųsti žinutę daugiau nei vienam naudotojui vienu metu.	Būtina
FR 10.9	Dėstytojas turi turėti galimybę pasirinkti išsiųsti žinutę visai grupei arba kursui studentų.	Būtina

16 pav. pateikiama užduoties „Žinutės išsiuntimas” sekų diagrama. Joje vaizduojamas pagrindinis žinutės išsiuntimo scenarijus.



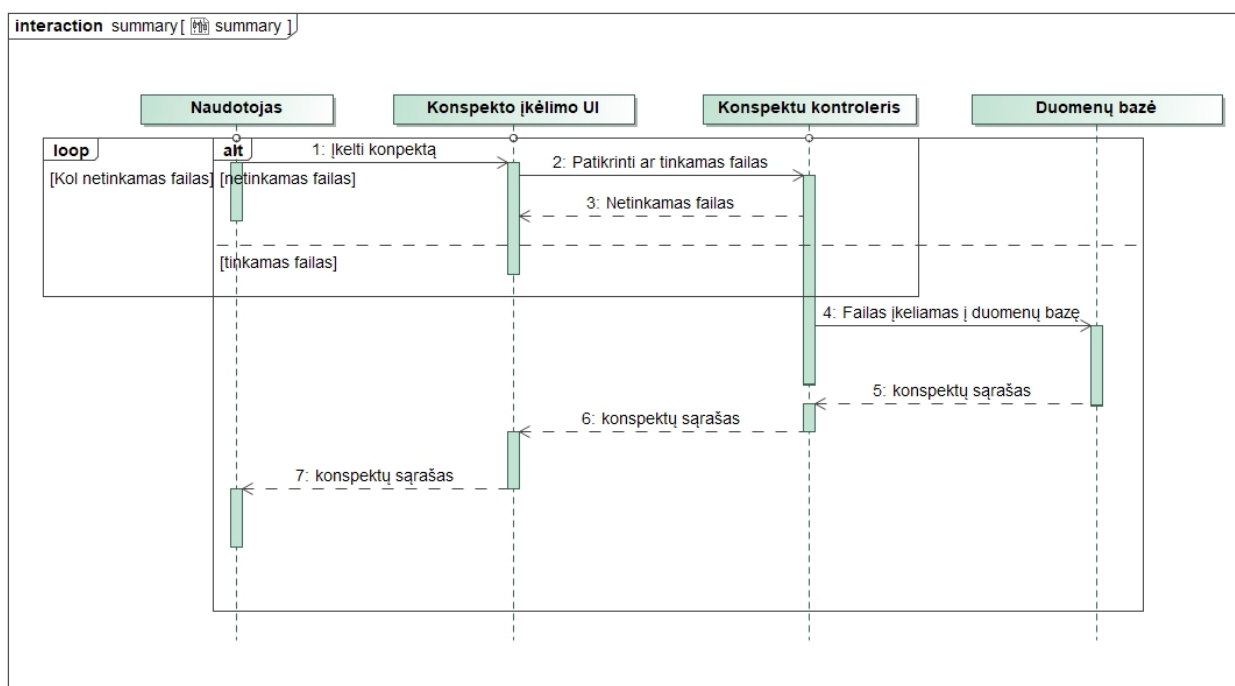
16 pav. Proceso „Žinutės išsiuntimas” sekų diagrama

## 2.1.11. Konspektai

16 lentelė. Funkciniai reikalavimai. Konspektai

Kodas	Reikalavimas	Svarba
Konspektai		
FR 11.1	Puslapis „Konspektai“ turi būti matomas visiems prisijungusiems studentams bei sistemos administratoriui.	Būtina
FR 11.2	Konspektai turi būti pateikiami šalia dėstomo dalyko pavadinimo, o dėstomi dalykai turi būti surikiuoti abėcėlės tvarka.	Būtina
FR 11.3	Puslapyje „Konspektai“ turi būti paieškos laukelis, kuriame turi būti galimybė ieškoti dėstomo dalyko.	Būtina
FR 11.4	Puslapyje „Konspektai“ turi būti pateikiami visi studentų įkelti konspektai.	Būtina
FR 11.5	Įkelti konspektą turi turėti galimybę visi prisijungę studentai.	Būtina
FR 11.6	Paspaudus mygtuką „Įkelti konspektą“ turi atsirasti failų pasirinkimo langas.	Būtina
FR 11.7	Paspaudus konspekto įkėlimo patvirtinimo mygtuką, konspektas turi atsirasti visų konspektų sąraše.	Būtina
FR 11.8	Įkeliant konspektą turi būti nurodomas dėstomas dalykas.	Būtina

45 pav. pateikiama užduoties „Konspekto įkėlimas“ sekų diagrama. Joje vaizduojamas pagrindinis konspekto įkėlimo scenarijus ir taip pat nagrinėjami alternatyvūs scenarijai.



17 pav. Proceso „Konspekto įkėlimas“ sekų diagrama



## 2.2. NEFUNKCINIAI REIKALAVIMAI

2-ame skyriuje pateikiami nefunkciniai reikalavimai bei jų svarba. Aprašoma, kaip sistema turi veikti ir kaip ji turi būti kuriama.

### 2.2.1. Vidinių interfeisų reikalavimai

17 lentelė. Nefunkciniai reikalavimai. Vidinių interfeisų reikalavimai.

Kodas	Reikalavimas	Svarba
Vidinių interfeisų reikalavimai		
<b>OS naudojimo reikalavimai</b>		
NFR 1.1	Tinklapis pritaikytas tiek kompiuteriams, tiek mobiliems įrenginiams.	Būtina
NFR 1.2	Puslapis pasiekiamas per visas populiariausias naršyklės (Google Chrome, Mozilla Firefox, IE (nuo 8 versijos), Edge, Safari).	Būtina
<b>Sąveikos su DB reikalavimai</b>		
NFR 2.1	Tinklapis turi turėti duomenų bazę, kurioje saugomi naudotojų duomenys, renginiai, D.U.K., konspektai bei dėstytojų puslapių informacija.	Būtina
NFR 2.2	Duomenys saugomi reliaciniu būdu, naudojama MySQL duomenų bazių valdymo sistema.	Būtina
NFR 2.3	Naudojama Microsoft Azure SQL Database paslauga.	Pageidautina
<b>Dokumentų mainų reikalavimai</b>		
NFR 3.1	Naudotojų įkeliamos nuotraukos turi būti .jpg, .png, .bmp formato bei neviršyti 5MB dydžio.	Būtina
<b>Darbo kompiuterių tinkluose reikalavimai</b>		
NFR 4.1	Duomenys perduodami naudojant HTTPS protokolą.	Būtina
<b>Sąveikos su kitomis programomis reikalavimai</b>		
NFR 5.1	Vartotojo autentifikacija vykdoma per is.vu.lt sistemą.	Būtina
<b>Programavimo aplinkos reikalavimai</b>		
NFR 6.1	Tinklapis kuriamas PHP programavimo kalba, naudojant Symfony karkasą.	Būtina
NFR 6.2	Kodo saugojimui ir dalinimuisi naudojama privati Github repositoryja.	Pageidautina
NFR 6.3	Naudojama PHPStorm programavimo aplinka.	Pageidautina

### 2.2.2. Veikimo reikalavimai

18 lentelė. Nefunkciniai reikalavimai. Veikimo reikalavimai.

Kodas	Reikalavimas	Svarba
Veikimo reikalavimai		
<b>Vaizdavimo reikalavimai</b>		
NFR 7.1	Tinklapis turi būti palaikomas visose populiariausiose naršyklėse (IE (nuo 8 versijos), Edge, Chrome, Safari, Firefox).	Būtina
NFR 7.2	Keičiant naršyklės dydį, tinklapis vaizdą pritaiko automatiškai.).	Būtina
NFR 7.3	Data turi būti vaizduojama formatu YYYY-MM-DD, kur YYYY – metai, MM – mėnuo, DD – diena.	Būtina
NFR 7.4	Laikas turi būti vaizduojamas formatu hh:mm, kur hh - valandos, mm - minutės.	Būtina
NFR 7.5	Pavadinimai – ne daugiau 50 simbolių.	Būtina
<b>Robastiškumo reikalavimai</b>		
NFR 8.1	Sistemoje turi būti įdiegtos apsaugos priemonės nuo duomenų sugadinimo, praradimo, klaidingų duomenų įvedimo į DB.	Būtina
NFR 8.2	Pranešti naudotojui, jei interneto ryšys nutrūko.	Pageidautina
<b>Našumo reikalavimai</b>		
NFR 9.1	Užklausiai įvykdyti turi užtekti ne daugiau nei 5 sekundžių.	Būtina
<b>Darbo kompiuterių tinkluose reikalavimai</b>		
NFR 10.1	Svetainės talpinimo (hostingo) planas turi būti parinktas atsižvelgiant į prognozuojamą klientų srautą. Rekomenduojamas duomenų srautas – 50GB/mėn., vieta serveryje - iki 3GB.	Būtina
NFR 10.1	Didžiausia leistina tinklapio sistemos apkrova yra 1000 naudotojų, prisijungusių vienu metu.	Būtina

### 2.2.3. Diegimo reikalavimai

19 lentelė. Nefunkciniai reikalavimai. Diegimo reikalavimai.

Kodas	Reikalavimas	Svarba
Diegimo reikalavimai		
<b>Ruošinio reikalavimai</b>		
NFR 11.1	Dokumentacija	Būtina
NFR 11.2	Hostingo prisijungimo duomenys.	Būtina
NFR 11.3	MS Azure prisijungimo duomenys.	Pageidautina
<b>Instaliavimo reikalavimai</b>		
NFR 12.1	Apsilankęs internetiniame puslapyje, vartotojas privalo sutikti su slapukų naudojimo sąlygomis.	Būtina
<b>Pradinio DB kaupimo reikalavimai</b>		
NFR 13.1	Turi būti sukurtos lentelės.	Būtina
NFR 13.2	Naudotojų lentelėje turi būti administratoriaus duomenys.	Būtina
<b>Sistemos įsisavinamumo reikalavimai</b>		
NFR 14.1	Sistema turi funkcionuoti dvejomis kalbomis: lietuvių ir anglų.	Būtina
NFR 14.2	Negali būti klaidinančių nuorodų.	Būtina
NFR 14.3	Ikonos turi atspindėti mygtuko panaudojimą.	Būtina

### 2.2.4. Aptarnavimo ir priežiūros reikalavimai

20 lentelė. Nefunkciniai reikalavimai. Aptarnavimo ir priežiūros reikalavimai.

Kodas	Reikalavimas	Svarba
Aptarnavimo ir priežiūros reikalavimai		
NFR 15.1	Atsiradęs naujas funkcionalumas turi būti įdiegtas per 5 darbo dienas.	Būtina
NFR 15.2	Rasta klaida turi būti ištaisyta per 2 darbo dienas.	Būtina
NFR 15.3	Į naudotojo laiškus su pastebėjimais ir skundais atsakyti reikia per 3 darbo dienas.	Pageidautina
NFR 15.4	Jei dėl planuojamo atnaujinimo reikės trumpam sustabdyti sistemos veiklą, naudotojai turi būti iš anksto įspėti ne mažiau nei prieš 24 val.	Pageidautina

### 2.2.5. Tiražuojamumo reikalavimai

21 lentelė. Nefunkciniai reikalavimai. Tiražuojamumo reikalavimai.

Kodas	Reikalavimas	Svarba
Tiražuojamumo reikalavimai		
NFR 16.1	Internetinė svetainė turi veikti bet kuriame įrenginyje, kuris turi naršyklę ir interneto ryšį.	Būtina

### 2.2.6. Apsaugos reikalavimai

22 lentelė. Nefunkciniai reikalavimai. Apsaugos reikalavimai.

Kodas	Reikalavimas	Svarba
Apsaugos reikalavimai		
NFR 17.1	Naudotojui prisijungiant prie sistemos vykdoma jo identifikacija.	Būtina
NFR 17.2	Atsarginės DB kopijos daromos ne rečiau nei kas savaitę.	Būtina
NFR 17.3	Jei naudotojas neaktyvus ilgiau nei 30 minučių, jis turi būti automatiškai atjungiamas nuo sistemos.	Būtina

### 2.2.7. Juridiniai reikalavimai

23 lentelė. Nefunkciniai reikalavimai. Juridiniai reikalavimai.

Kodas	Reikalavimas	Svarba
Juridiniai reikalavimai		
NFR 18.1	Kuriant sistemą projekto komanda neturi naudotis nelegalia programine įranga.	Būtina
NFR 18.2	Internetinėje svetainėje turi būti galimybė peržiūrėti naudojimosi sąlygas.	Būtina

## 2.3. VARTOTOJO SĄSAJOS REIKALAVIMAI

3 skyriuje pateikiami vartotojo sąsajos reikalavimai, kuriuose pateikiama informaciją apie sistemos grafinį vaizdą, kurį mato vartotojas. Nagrinėjami naudotojui matomi puslapiai, ikonos, simboliai bei mygtukai, pavaizduoti prototipuose (žr. 1 priedas). Taip pat aprašomos jų funkcijos, paskirtys bei svarbumas.

### 2.3.1. Dalykinės srities metaforos reikalavimai

24 lentelė. Vartotojo interfeiso reikalavimai. Dalykinės srities metaforos reikalavimai.

1. Dalykinės srities metaforos reikalavimai		
Kodas	Reikalavimas	Svarba
VIR 1.1	Pagrindinio puslapio atidarymas arba perkrovimas yra vaizduojamas puslapio logotipu	Būtinai
VIR 1.2	Renginio arba naujienos įkėlimas yra vaizduojamas pliuso simboliu.	Pageidautina
VIR 1.3	Renginio arba naujienos redagavimas yra vaizduojamas pieštuko simboliu.	Pageidautina
VIR 1.4	Renginiai išdėstyti nuo eilės tvarka nuo artimiausios datos iki tolimiausios	Būtina

25 lentelė. Vartotojo interfeiso reikalavimai. Užduočių reikalavimai

2. Užduočių reikalavimai		
Kodas	Reikalavimas	Svarba
Neprisiregistravusio naudotojo sąsajos užduotys		
VIR3.1	Prisiregistruoti prie aplikacijos	Būtinai
VIR2.2	Susipažinti su puslapio taisyklėmis	Būtinai
Dėstytojo sąsajos užduotys		
VIR3.1	Prisijungti prie savo paskyros	Būtinai
VIR3.2	Redaguoti savo prisijungimo slaptažodį	Būtinai
VIR3.3	Atsijungti iš savo paskyros	Būtinai
VIR3.4	Pridėti naujieną	Būtinai
VIR3.5	Redaguoti renginį	Būtinai
VIR3.6	Ištrinti renginį	Būtinai
VIR3.7	Pridėti renginį	Būtinai
VIR3.8	Redaguoti renginį	Būtinai
VIR3.9	Ištrinti renginį	Būtinai
VIR3.10	Siųsti žinutę	Būtinai
VIR3.11	Peržiūrėti visas žinutes	Būtinai
VIR3.12	Gauti žinutes	Būtinai
VIR3.13	Peržvelgti visas naujienas	Būtinai
VIR3.14	Peržvelgti visus renginius	Būtinai

VIR3.15	Peržiūrėti D.U.K.	Būtinai
VIR3.16	Pridėti D.U.K.	Būtinai
VIR3.17	Peržiūrėti D.U.K.	Būtinai
VIR3.18	Peržvelgti visus dėstytojus	Būtinai
VIR3.19	Redaguoti asmeninį tinklalapį	Būtinai
Studento sąsajos užduotys		
VIR4.1	Prisijungti prie savo paskyros	Būtinai
VIR4.2	Redaguoti savo prisijungimo slaptažodį	Būtinai
VIR4.3	Atsijungti iš savo paskyros	Būtinai
VIR4.4	Siųsti žinutę	Būtinai
VIR4.5	Peržiūrėti visas žinutes	Būtinai
VIR4.6	Gauti žinutes	Būtinai
VIR4.7	Peržvelgti visas naujienas	Būtinai
VIR4.8	Peržvelgti visus renginius	Būtinai
VIR4.9	Peržiūrėti D.U.K.	Būtinai
VIR4.10	Peržiūrėti D.U.K.	Būtinai
VIR4.11	Peržvelgti visus dėstytojus	Būtinai
VIR4.12	Peržiūrėti konkretaus dėstytojo puslapį ir informaciją jame	Būtinai
VIR4.13	Peržiūrėti visus konspektus	Būtinai
VIR4.14	Įkelti konspektą	Būtinai
Administratoriaus sąsajos reikalavimai		
VIR5.1	Pridėti naujienas/renginius/D.U.K.	Būtinai
VIR5.2	Peržiūrėti naujienas/renginius/D.U.K. ir pašalinti nebeaktualius	Būtinai
VIR5.3	Kurti sąrašus duomenų bazėje	Būtinai
VIR5.4	Atnaujinti tinklalapį	Būtinai
VIR5.5	Patikrinti konspektų turinį	Būtinai
VIR5.6	Taisyti tinklalapio klaidas	Būtinai
VIR5.7	Blokuoti naudotojus	Būtinai
Bendri reikalavimai		
VIR6.1	Puslapio viršuje visada esantis meniu	Būtinai
VIR6.2	Teksto įvedimo formų laukai	Būtinai
VIR6.3	Ikonos	Būtinai
VIR6.4	Matomas atsijungimo mygtukas	Būtinai

### 2.3.2. Užduočių formulavimo kalbos reikalavimai

26 lentelė. Vartotojo interfeiso reikalavimai. Užduočių formulavimo kalbos reikalavimai

3. Užduočių formulavimo kalbos reikalavimai		
Kodas	Reikalavimas	Svarba
Įrankiai skirti naudotojui naudotis aplikacija		
VIR7.1	Grafinis meniu – vartotojo sąsaja su tinklalapiu	Būtinai
VIR7.2	Mygtukai – naudojami siekiant patekti į kitus tinklalapio langus	Būtinai
VIR7.3	Ikonos – interfeise naudojamos piktogramos	Būtinai
VIR7.4	Patvirtinimo langai - langai prašantys naudotojo dar kartą patvirtinti tam tikrą svarbų veiksmą	Būtinai
VIR7.5	Įvedimo laukai – naudojami naudotojui įvesti tekstinius duomenis	Būtinai
VIR8.1	Naudotojo prisijungimo vardas turi būti validus el. pašto adresas egzistuojantis VU sistemoje	Būtinai
VIR9.1	Naudotojo slaptažodis turi būti sudarytas iš raidžių (didžiųjų ir mažųjų), skaitmenų ir specialių simbolių	Būtinai
VIR9.2	Slaptažodis turi būti ne trumpesnis nei 8 simboliai	Būtinai

### 2.3.3. Užduočių formulavimo protokolo reikalavimai

27 lentelė. Vartotojo interfeiso reikalavimai. Užduočių formulavimo protokolo reikalavimai

4. Užduočių formulavimo protokolo reikalavimai		
Kodas	Reikalavimas	Svarba
Prisiregistravimas prie tinklalapio		
VIR10.1	Norėdamas prisiregistruoti naudotojas turi paspausti mygtuką „Registruotis“. Paspaudus jį išmetamas registracijos langas, kuriame naudotojas suveda savo duomenis (el. paštas, vardas, pavardė, slaptažodis, pasirinkama dėstytojo/studento kategorija, pasirinkus dėstytoją suvedamas dėstytojo identifikacijos kodas)	Būtinai
VIR10.2	Paspaudus mygtuką „Registruotis“ registracijos lange tikrinama ar duomenys suvesti teisingai ir ar tokio naudotojo dar nėra duomenų bazėje, jei viskas gerai, naudotojas prijungiamas prie paskyros. Kitu atveju į ekraną išmetamos žinutės prie tų laukų, kurie yra suvesti klaidingai	Būtinai
Prisijungimas prie tinklalapio		
VIR11.1	Prisijungti gali tik registruotas naudotojas. Tai padaryti gali paspaudęs mygtuką „Prisijungti“ ir išmestame lange suvedęs savo prisijungimo duomenis (el.paštą, slaptažodį)	Būtinai

VIR11.2	Paspaudus tvirtinantį prisijungimą mygtuką „Prisijungti“ duomenys yra patikrinami duomenų bazėje ir, jei viskas teisinga, naudotojas yra prijungiamas. Jei prisijungimas klaidingas, naudotojui išmetama žinutė, kad prijungimas nepavyko, prašoma patikrinti suvestus laukus	Būtinai
VIR11.3	Prisijungiant galima pažymėti varnelę prie „Prisiminti mane“ ir kitą kartą naudotojas bus prijungiamas automatiškai, nevedant duomenų iš naujo	Būtinai
Pamirštas slaptažodis		
VIR12.1	Pamiršus slaptažodį galima paspausti ant mygtuko „Pamiršau slaptažodį“. Atsiradusiame lange reikia įvesti el.paštą, kuriuo naudotojas prisijungia.	Būtinai
VIR12.2	Sistema radusi tokį el.paštą duomenų bazėje išsiunčia nuorodą nurodytu el.pašto adresu nukreipiančiu į formą leidžiančią pasikeisti slaptažodį	Būtinai
Pagrindinis juostinis meniu		
VIR13.1	Vaizduojamas viršutinėje puslapio dalyje, matomas kiekviename pasirinkimų lange	Būtinai
VIR13.2	Kairiame kampe vaizduojamas tinklalapio logotipas ir pavadinimas „SocialVU“	Būtinai
VIR13.3	Dešiniame kampe rodomas naudotojo el.paštas, kuris nukreipia į asmeninį profilį, kuriame galima rasti prisijungimo informaciją, ir atsijungimo mygtukas „Atsijungti“ bei „Žinutės“	Būtinai
VIR13.4	Prisijungus dėstytojo aplinkoje šone atsiranda mygtukas „Mano puslapis“, kuris nukreipia į asmeninį dėstytojo puslapį	Būtinai
VIR13.5	„Pagrindinis“ - atverčiamas pagrindinis, naujienų, puslapis	Būtinai
VIR13.6	„D.U.K.“ - atverčiami dažniausiai užduodami klausimai su atsakymais	Būtinai
VIR13.7	„Dėstytojai“ - atverčiamas visų dėstytojų sąrašas	Būtinai
VIR13.8	„Renginiai“ - atverčiamas visų renginių sąrašas	Būtinai
„SocialVU“ logotipas		
VIR14.1	„Renginiai“ - atverčiamas visų renginių sąrašas	Būtinai
„Renginiai“		
VIR15.1	Pateikiamas pilnas renginių sąrašas su pavadinimu, data, trumpa informacija, kaina	Būtinai
VIR15.2	Paspaudus ant padidinamo stiklo aktyvuojamas įvesties langas, kuriame galima ieškoti renginio suvedant tai, ko ieškoma	Būtinai
VIR15.3	Paspaudus ant tekstinio lauko, kuriame galima pasirinkti datą, renginiai filtruojami pagal pasirinktą dieną	Būtinai
VIR15.4	Paspaudus ant konkretaus renginio, išmetama papildoma informacija apie jį	Būtinai
VIR15.5	Paspaudus „+“ (matoma tik dėstytojui) išmetama renginio pridėjimo forma	Būtinai
„Dėstytojai“		



VIR16.1	Pateikiamas pilnas dėstytojų sąrašas (vardas, pavardė, dėstomų dalykų sąrašas, nuotrauka)	Būtinai
VIR16.2	Paspaudus ant padidinamo stiklo aktyvuojamas įvesties langas, kuriame galima ieškoti dėstytojo suvedant tai, ko ieškoma	Būtinai
VIR16.3	Paspaudus ant konkretaus dėstytojo, naudotojas nukreipiamas į dėstytojo puslapį	Būtinai
„D.U.K.“		
VIR17.1	Pateikiamas pilnas D.U.K. sąrašas su atsakymais	Būtinai
VIR17.2	Paspaudus ant padidinamo stiklo aktyvuojamas įvesties langas, kuriame galima ieškoti klausimo suvedant tai, ko ieškoma	Būtinai
VIR17.3	Paspaudus „+“ (matoma tik dėstytojui) išmetama renginio pridėjimo forma	Būtinai
„Žinutės“		
VIR18.1	Pateikiamas pilnas naudotojo gautų žinučių sąrašas	Būtinai
VIR18.2	Paspaudus „Išsiųstos“ pateikiamas pilnas naudotojo išsiųstų žinučių sąrašas	Būtinai
VIR18.3	Paspaudus „Visos“ pateikiamas pilnas naudotojo ir gautų, ir išsiųstų žinučių sąrašas	Būtinai
VIR18.4	Paspaudus „Sukurti“ išmetama naujos žinutės forma, kurioje pasirenkamas gavėjas, bei užpildomi žinutės duomenys	Būtinai
VIR19.4	Paspaudus ant padidinamo stiklo aktyvuojamas įvesties langas, kuriame galima ieškoti žinutės suvedant tai, ko ieškoma	Būtinai
„Konspektai“ (pasiekiami tik studentui)		
VIR19.1	Pateikiamas pilnas dėstomų dalykų sąrašas	Būtinai
VIR19.2	Pasirinkus konkretų dalyką atidaromas sukeltų konspektų sąrašas, su trumpa informacija (autorius vardas, pavardė, sukūrimo data)	Būtinai
VIR19.3	Pasirinkus konkretų konspektą atidaromas failo turinys	Būtinai
VIR19.4	Paspaudus ant padidinamo stiklo aktyvuojamas įvesties langas, kuriame galima ieškoti konspekto suvedant tai, ko ieškoma	Būtinai
VIR19.5	Paspaudus „+“ išmetama konspekto pridėjimo forma	Būtinai
„Mano puslapis“ (pasiekiamas tik dėstytojui)		
VIR20.1	Pateikiamas naudotojo asmeninis puslapis, kuriame informaciją gali pateikti pats	Būtinai
„Šoninė juosta“		
VIR21.1	Matoma dešinėje sistemos pusėje, matoma kiekviename pasirinkimų lange	Būtinai
„El. paštas“		
VIR22.1	Paspaudus išmetama informacija apie naudotoją	Būtinai
VIR22.2	Paspaudus ant mygtuko „Pakeisti slaptažodį“ naudotojui išmetama forma, kurioje galima pasikeisti savo prisijungimo slaptažodį	Būtinai

#### 2.3.4. Pranešimo formulavimo reikalavimai

28 lentelė. Vartotojo interfeiso reikalavimai. Pranešimo formulavimo reikalavimai

5. Pranešimo formulavimo reikalavimai		
Kodas	Reikalavimas	Svarba
VIR23	Pranešimai turi būti parašyti laikantis gramatikos ir skyrybos taisyklių	Būtinai
VIR24	Pranešimai turi būti aiškūs, suprantami, kuo trumpesni bei vienareikšmiški. Aprašo tik tą sritį, dėl kurios yra išmetami naudotojui	Būtinai
VIR25	Patvirtinimai turi būti aiškūs, suprantami bei vienareikšmiški. Klausia tik patvirtinimo reikalingo užduočiai patvirtinti arba nutraukti	Būtinai

#### 2.3.5. Interfeiso darnos ir standartizavimo reikalavimai

29 lentelė. Vartotojo interfeiso reikalavimai. Interfeiso darnos ir standartizavimo reikalavimai

6. Interfeiso darnos ir standartizavimo reikalavimai		
Kodas	Reikalavimas	Svarba
VIR26	Visi grafiniai objektai turi derėti tarpusavyje. Visi mygtukai, lentelės pranešimai, ikonos derančios išvaizdos.	Būtinai
VIR27	Tinklapyje naudojamos vienos paletės spalvos ir lengvai įskaitomas šriftas	Būtinai

#### 2.3.6. Interfeiso individualizavimo reikalavimai

30 lentelė. Vartotojo interfeiso reikalavimai. Interfeiso individualizavimo reikalavimai

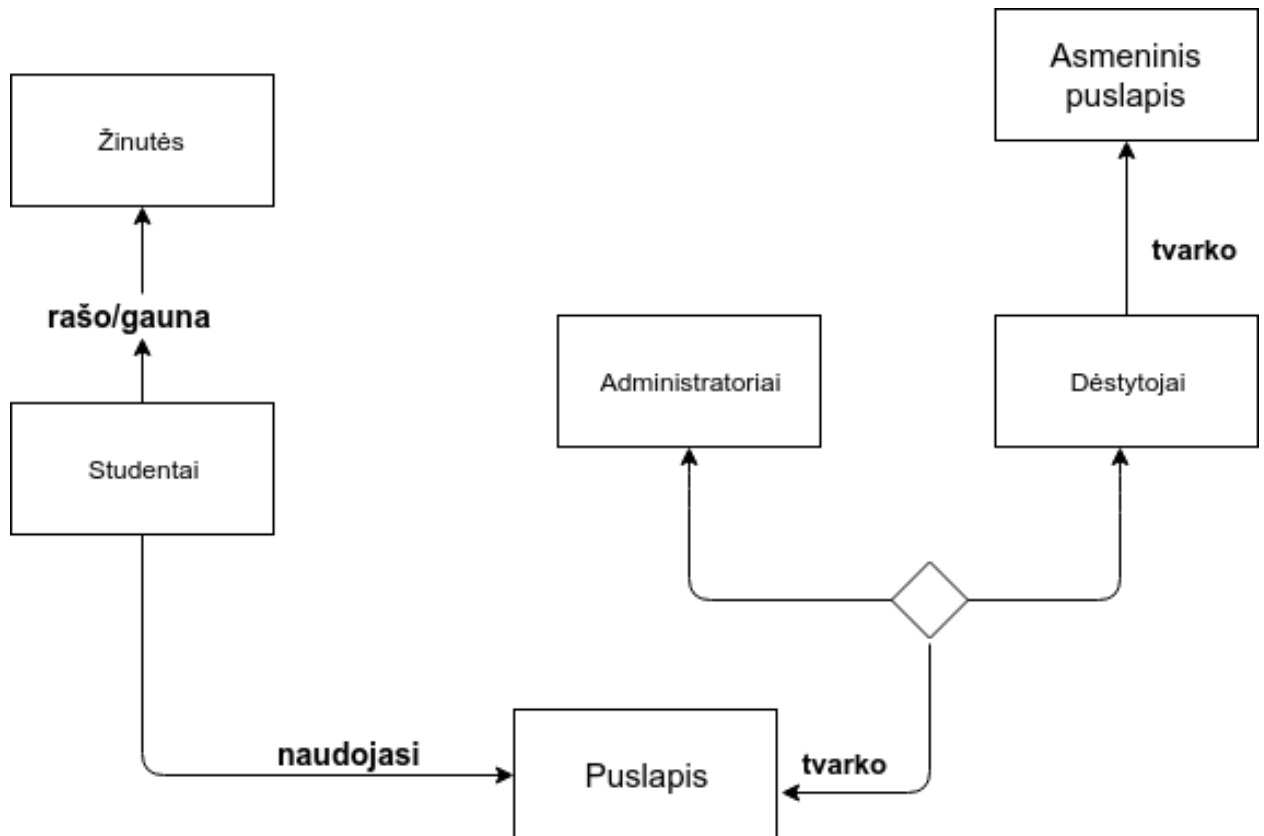
7. Interfeiso individualizavimo reikalavimai		
Kodas	Reikalavimas	Svarba
VIR28	Sistemos spalvų pasirinkimas	Pageidautinas
VIR29	Kalbos pasirinkimas sistemoje	Pageidautinas

### 3. KURIAMOS SISTEMOS ARCHITEKTŪRA

#### 3.1. LOGINIS PJŪVIS

Loginį pjūvį sudaro klasių diagramos, kurios naudojamos pavaizduoti sistemos architektūros projektavimo etapus.

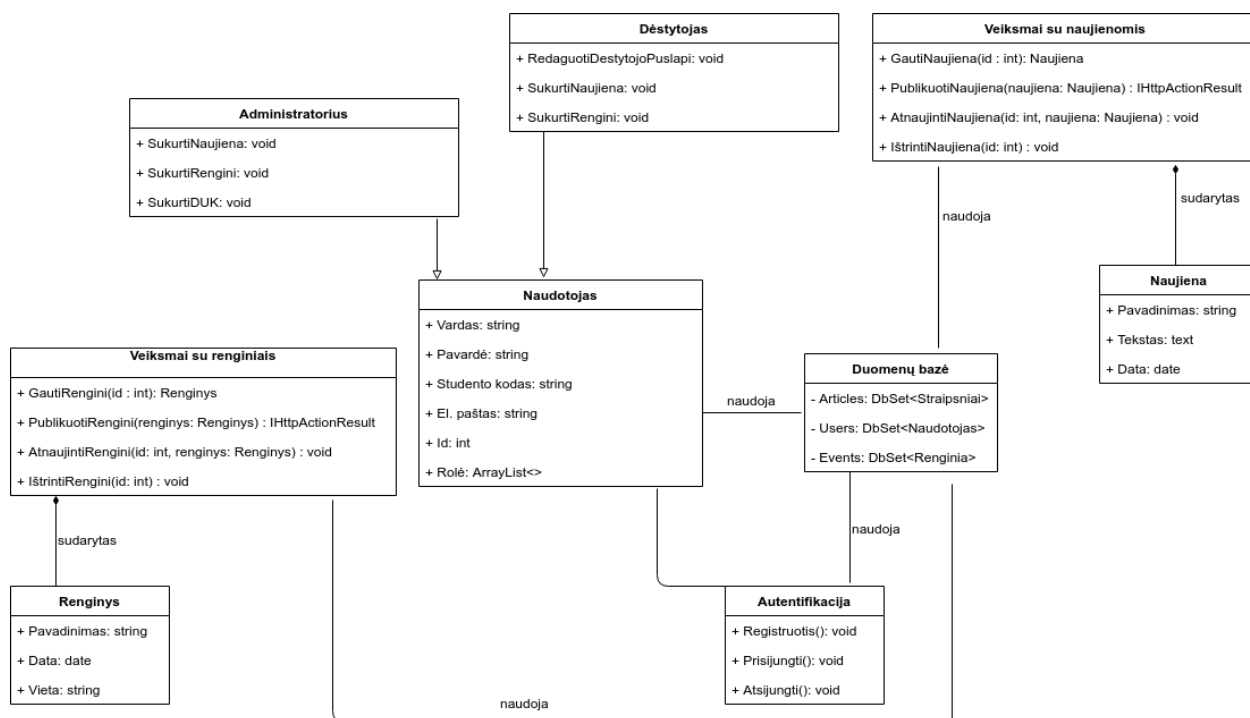
##### 3.1.1. Esybių klasių diagrama (nulinis lygis)



18 pav. Dalykinė srities UML diagrama

18 pav. esybių diagramoje vaizduojamos esybių sąsajos. Pagrindinė esybė Naudotojas, kuris gali būti Studentas, Dėstytojas arba Administratorius. Studentas turi galimybę naudotis pagrindinėmis puslapio funkcijomis, o dėstytojai ir administratoriai pateikti naudingą studentams medžiagą. Taip pat studentai bei dėstytojai gali komunikuoti tarpusavyje nesinaudojant trečiųjų šalių komunikacinėmis priemonėmis. Administratoriai, savo ruožtu, pateikia informaciją apie renginius, naujienas ir D.U.K.

### 3.1.2. Klasių diagrama (pirmas lygis)



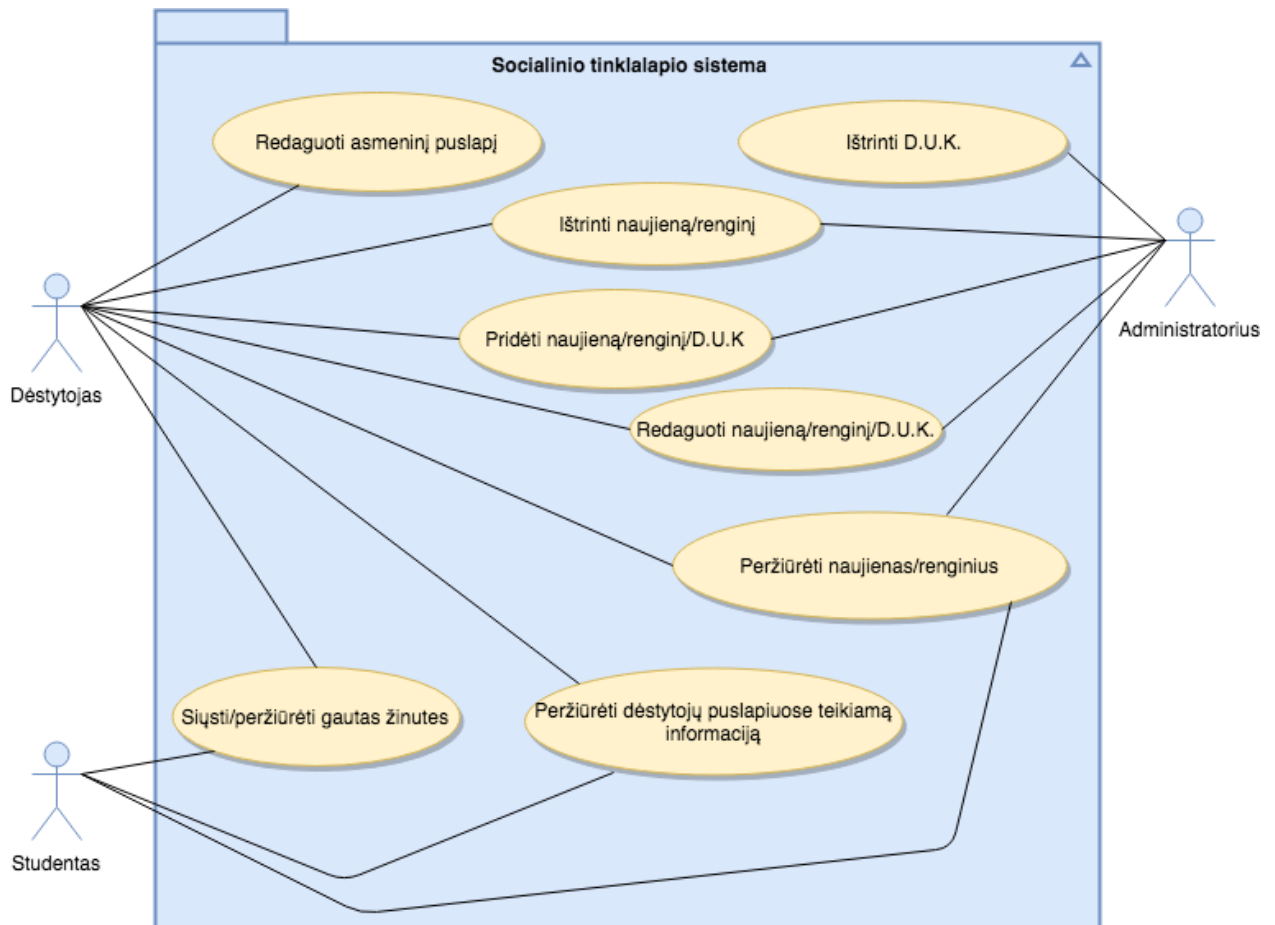
19 pav. Dalykinė srities UML diagrama

Pagrindinį programos funkcionalumą užtikrina šios klasės: Studentas, Dėstytojas, Administratorius, Reitingas, Dėsytojo puslapis, Naujienos, Autentifikacija, Duomenų bazė, D.U.K., Renginiai. Veikimą įgyvendinačių klasių tarpusavio bendradarbiavimas vaizduojamas asociacija, generalizacija, kompozicija bei kardinalumus (19 pav.).

## 3.2. UŽDUOČIŲ PJŪVIS

Šiame skyriuje aprašomas kuriamo socialinio tinklalapio galimi panaudojimo atvejai. Pasaudojant užduočių diagrama pateikiami studento ir dėstytojo (SocialVU naudotojų) tikslai socialiniam tinklalapiui. Kiekvienai užduočiai pateikiamas scenarijus, kuris parodo, kaip užduotis įgyvendinama.

### 3.2.1. Sistemoje vykdomos užduotys



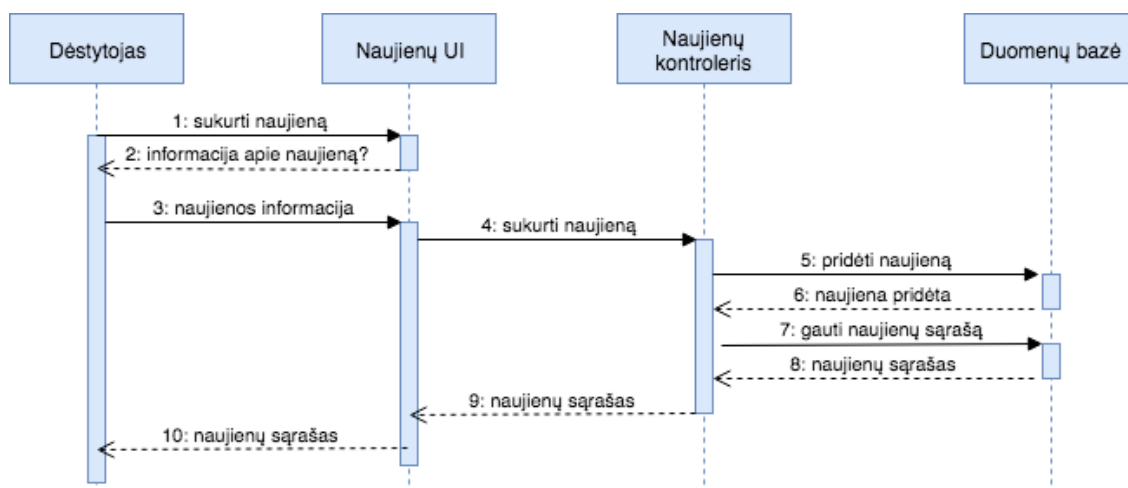
20 pav. Socialiniame tinklalapyje vykdomos užduotys

Sistemoje vykdomos pagrindinės užduotys: studentas gali peržiūrėti pasirinkto dėstytojo puslapius. Dėstytojams sudaroma galimybė pateikti aktualias naujienas, informaciją apie renginius, redaguoti savo asmeninius tinklalapius, kuriuose gali talpinti informaciją apie savo dėstomus dalykus bei kitą naudingą informaciją, kurią matys jų studentai. Tiek dėstytojai, tiek studentai gali gauti informaciją apie juos dominančius renginius, matyti aktualias naujienas, gauti bei siųsti žinutes.(20 pav.).

### 3.2.2. Užduočių vykdymo scenarijai

Užduočių vykdymo scenarijai, atvaizduoja agentų, šiuo atveju studento ir dėstytojo, įmanomų įvykdyti užduočių veiksmus paeiliui, nuo pradžios iki užduoties vykdymo pabaigos.

### 3.2.3. Užduoties "Pridėti naujieną" scenarijus

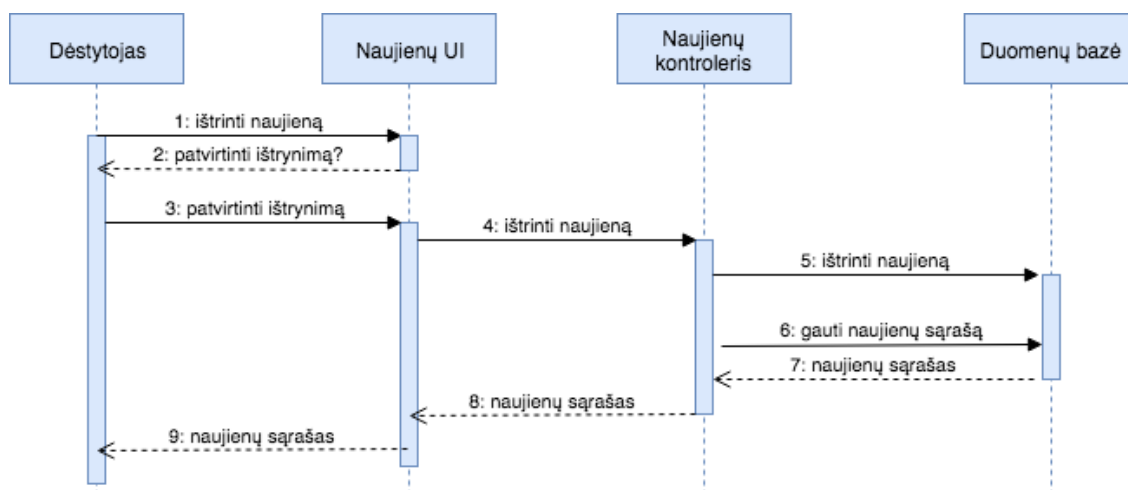


21 pav. Užduoties "Pridėti naujieną" scenarijus

#### Žingsnių seka (21pav.)

1. Sukurti naujieną - dėstytojas paspaudžia ant nuorodos leidžiančios sukurti naujieną.
2. Informacija apie naujieną? - naujienų UI išmeta dėstytojui naujienos formą užpildymui.
3. Naujienos informacija - naudotojas išsiunčia užpildytą formą naujienų UI.
4. Sukurti naujieną - įvesta naujienos informacija yra siunčiama kontrolieriui.
5. Pridėti naujieną - darbų UI įdeda naujieną į duomenų bazę.
6. Naujiena pridėta - duomenų bazė parsiunčia naujienos patalpinimo patvirtinimą.
7. Gauti naujienų sąrašą - naujienų kontrolieris prašo duomenų bazės pateikti naują naujienų sąrašą.
8. Naujienų sąrašas - naujienų sąrašas keliauja iš duomenų bazės iki dėstytojo.

### 3.2.4. Užduoties "Ištrinti naujieną" scenarijus

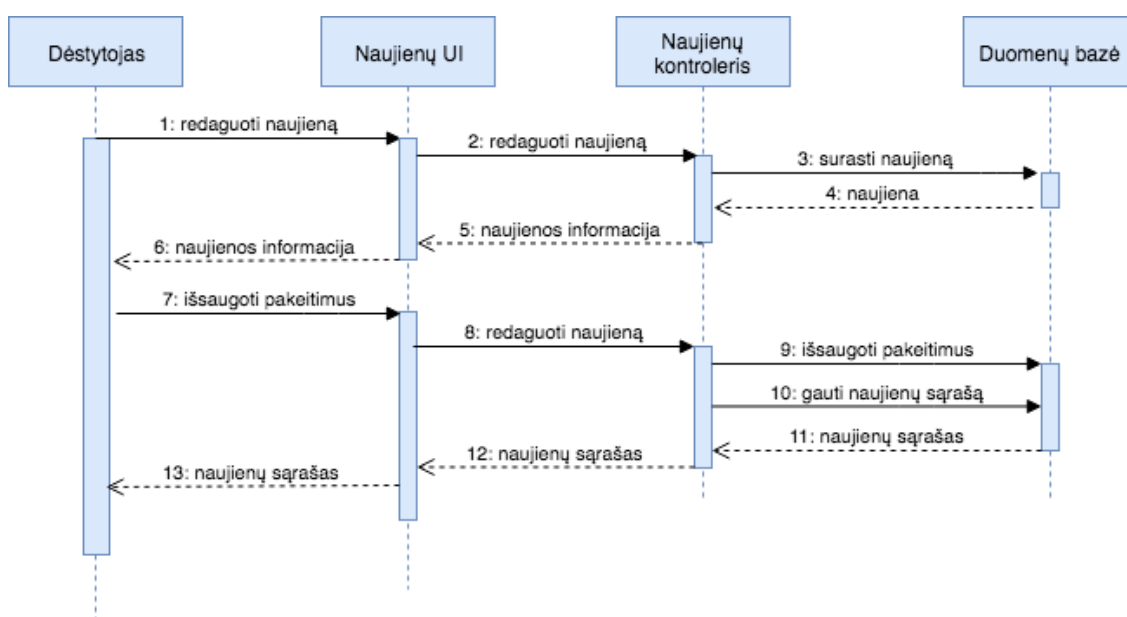


22 pav. Užduoties "Ištrinti naujieną" scenarijus

### Žingsnių seka (22pav.)

1. Ištrinti naujieną - dėstytojas paspaudžia nuorodą ištrinančią naujieną.
2. Patvirtinti ištrynimą? - naudotojas prašomas patvirtinti ištrynimą.
3. Patvirtinti ištrynimą - dėstytojas patvirtina ištrynimą.
4. Ištrinti naujieną - pateikiama naujiena ištrynimui įvykdyti.
5. Ištrinti naujieną - prašoma duomenų bazės surasti ir ištrinti naujieną.
6. Gauti naujienų sąrašą - siunčiamas prašymas naujienų sąrašui gauti iš duomenų bazės.
7. Naujienų sąrašas - atnaujintas sąrašas keliauja iki dėstytojo.

### 3.2.5. Užduoties "Redaguoti naujieną" scenarijus

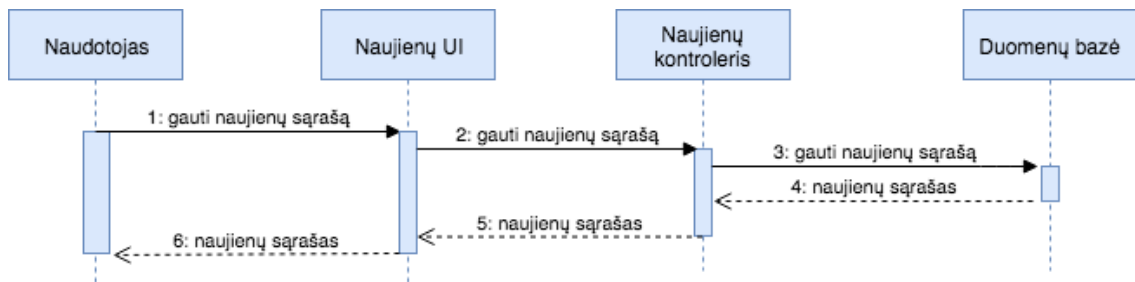


23 pav. Užduoties "Redaguoti naujieną" scenarijus

### Žingsnių seka (23pav.)

1. Redaguoti naujieną - dėstytojas paspaudžia nuorodą leidžiančią redaguoti naujieną.
2. Redaguoti naujieną - pateikiama naujiena naujienų kontrolieriui.
3. Surasti naujieną - naujiena ieškoma duomenų bazėje.
4. Naujiena - duomenų bazė pateikia naujieną kontrolieriui.
5. Naujienos informacija - informacija apie naujieną keliauja iki naudotojo.
6. Naujienos informacija - informacija apie naujieną keliauja iki naudotojo.
7. Išsaugoti pakeitimus - naudotojas prašo išsaugoti įvykdytus pakeitimus.
8. Redaguoti naujieną - naujiena siunčiama redagavimui.
9. Išsaugoti pakeitimus - pakeitimai išsaugomi duomenų bazėje.
10. Gauti naujienų sąrašą - kontrolieris prašo duomenų bazės gauti atnaujintą naujienų sąrašą.
11. Naujienų sąrašas - naujienų sąrašas keliauja iš duomenų bazės iki dėstytojo.

### 3.2.6. Užduoties "Peržiūrėti naujienas" scenarijus

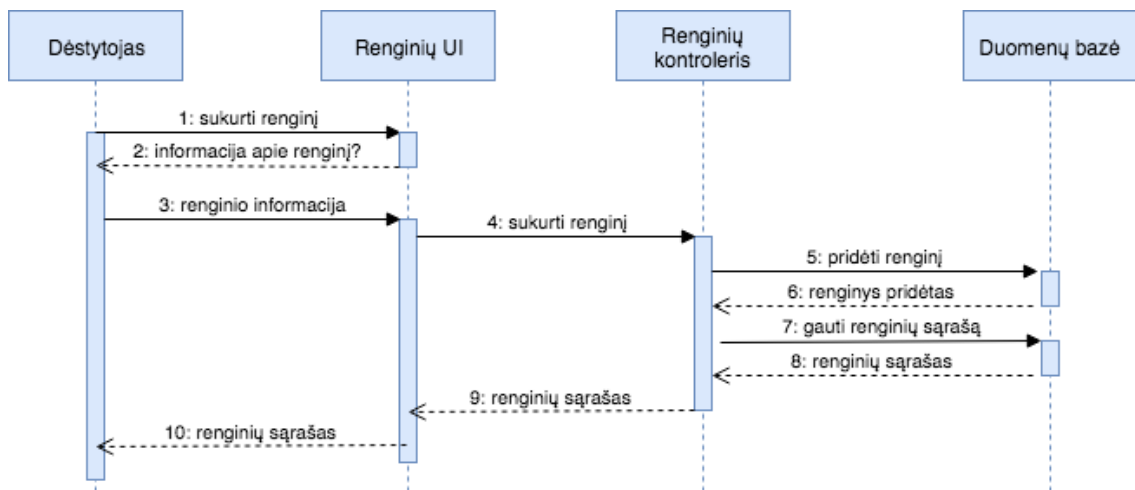


24 pav. Užduoties "Peržiūrėti naujienas" scenarijus

#### Žingsnių seka (24pav.)

1. Gauti naujienų sąrašą - dėstytojas paspaudžia nuorodą į naujienų sąrašą.
2. Gauti naujienų sąrašą - prašymas gauti naujienų sąrašą keliauja iki duomenų bazės.
3. Naujienų sąrašas - naujienų sąrašas iš duomenų bazės keliauja iki naudotojo.

### 3.2.7. Užduoties "Pridėti renginį" scenarijus



25 pav. Užduoties "Pridėti renginį" scenarijus

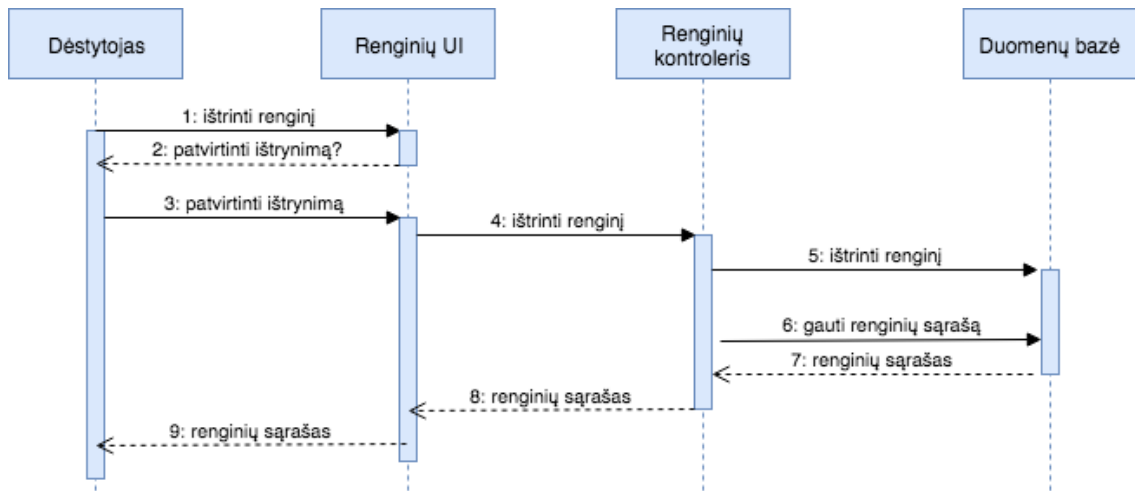
#### Žingsnių seka (25pav.)

1. Sukurti renginį - dėstytojas paspaudžia ant nuorodos leidžiančios sukurti renginį.
2. Informacija apie renginį? - renginių UI išmeta dėstytojui renginio formą užpildymui.
3. Renginio informacija - naudotojas išsiunčia užpildytą formą renginių UI.
4. Sukurti renginį - įvesta renginio informacija yra siunčiama kontrolieriui.
5. Pridėti renginį - renginių UI įdeda renginį į duomenų bazę.
6. Renginys pridėtas - duomenų bazė parsisiunčia renginio patalpinimo patvirtinimą.
7. Gauti renginių sąrašą - renginių kontrolieris prašo duomenų bazės pateikti naują renginių sąrašą.



8. Renginių sąrašas - renginių sąrašas keliauja iš duomenų bazės iki dėstytojo.

### 3.2.8. Užduoties "Ištrinti renginį" scenarijus

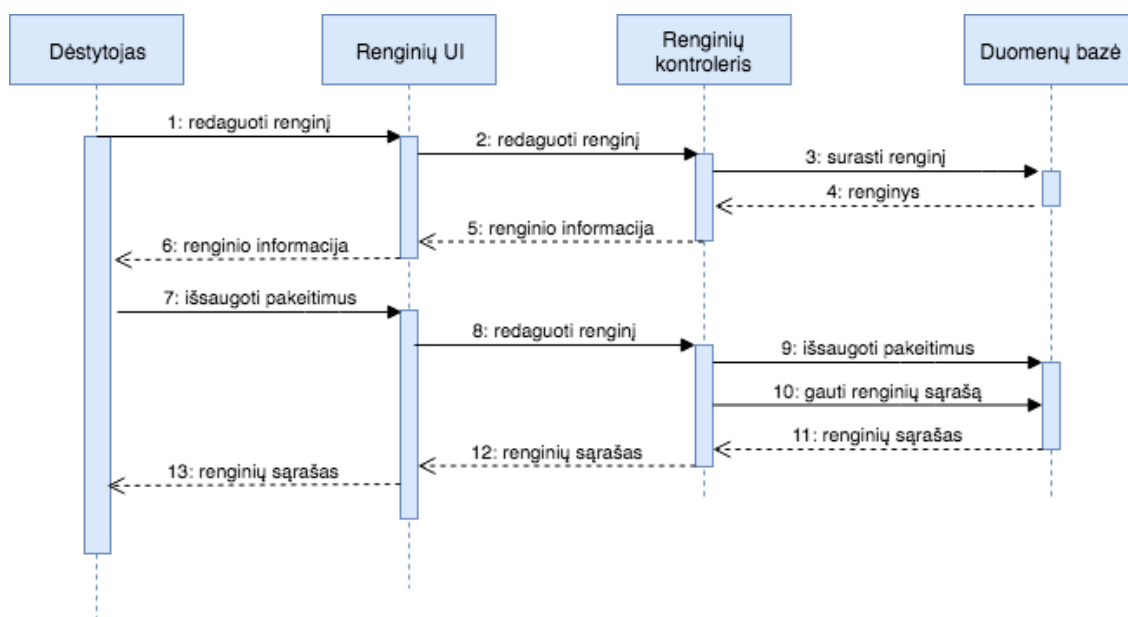


26 pav. Užduoties "Ištrinti renginį" scenarijus

#### Žingsnių seka (26pav.)

1. Ištrinti renginį - dėstytojas paspaudžia nuorodą ištrinančią renginį.
2. Patvirtinti ištrynimą? - naudotojas prašomas patvirtinti ištrynimą.
3. Patvirtinti ištrynimą - dėstytojas patvirtina ištrynimą.
4. Ištrinti renginį - pateikiamas renginys ištrynimui įvykdyti.
5. Ištrinti renginį - prašoma duomenų bazės surasti ir ištrinti renginį.
6. Gauti renginių sąrašą - siunčiamas prašymas renginių sąrašui gauti iš duomenų bazės.
7. Renginių sąrašas - atnaujintas sąrašas keliauja iki dėstytojo.

### 3.2.9. Užduoties "Redaguoti renginį" scenarijus

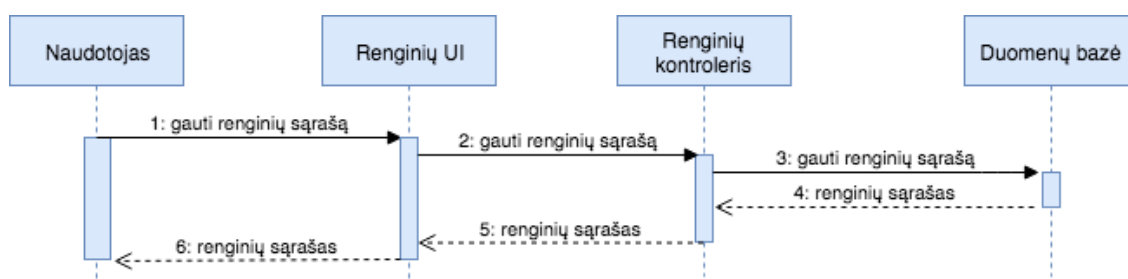


27 pav. Užduoties "Redaguoti renginį" scenarijus

#### Žingsnių seka (27pav.)

1. Redaguoti renginį - dėstytojas paspaudžia nuorodą leidžiančią redaguoti renginį.
2. Redaguoti renginį - pateikiamas renginys renginių kontrolieriui.
3. Surasti renginį - renginys ieškomas duomenų bazėje.
4. Renginys - duomenų bazė pateikia renginį kontrolieriui.
5. Renginio informacija - informacija apie renginį keliauja iki naudotojo.
6. Renginio informacija - informacija apie renginį keliauja iki naudotojo.
7. Išsaugoti pakeitimus - naudotojas prašo išsaugoti įvykdytus pakeitimus.
8. Redaguoti renginį - renginys siunčiamas redagavimui.
9. Išsaugoti pakeitimus - pakeitimai išsaugomi duomenų bazėje.
10. Gauti renginių sąrašą - kontrolieris prašo duomenų bazės gauti atnaujintą renginių sąrašą.
11. Renginių sąrašas - renginių sąrašas keliauja iš duomenų bazės iki dėstytojo.

### 3.2.10. Užduoties "Peržiūrėti renginius" scenarijus

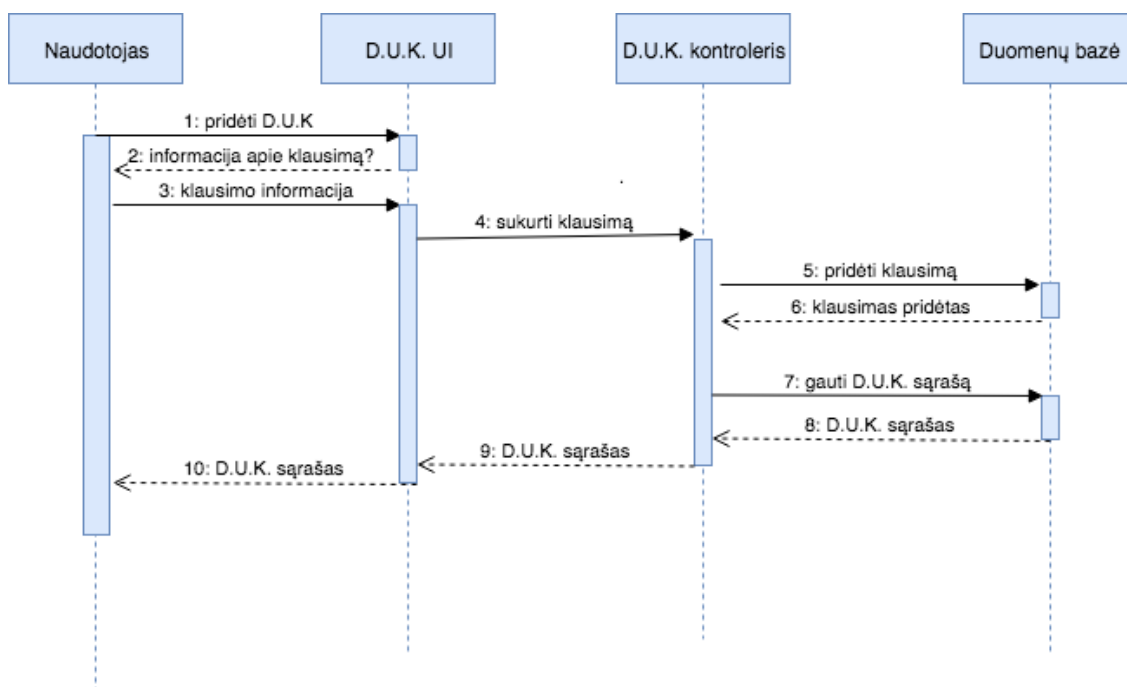


28 pav. Užduoties "Peržiūrėti renginius" scenarijus

### Žingsnių seka (28pav.)

1. Gauti renginių sąrašą - dėstytojas paspaudžia nuorodą į renginių sąrašą.
2. Gauti renginių sąrašą - prašymas gauti renginių sąrašą keliauja iki duomenų bazės.
3. Renginių sąrašas - renginių sąrašas iš duomenų bazės keliauja iki naudotojo.

### 3.2.11. Užduoties "Pridėti D.U.K." scenarijus

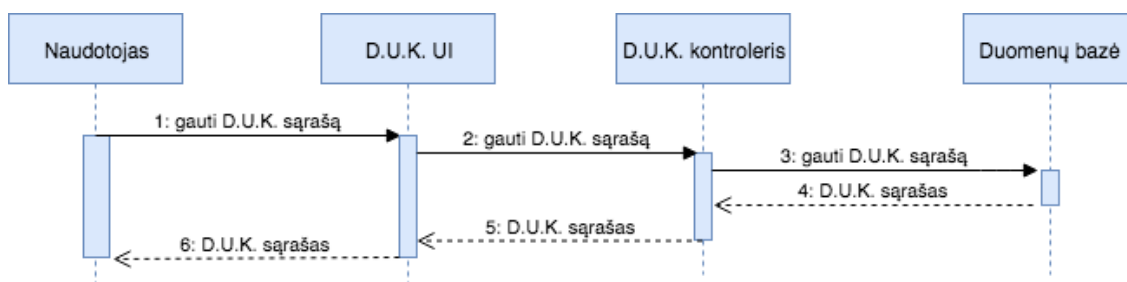


29 pav. Užduoties "Pridėti D.U.K." scenarijus

### Žingsnių seka (29pav.)

1. Pridėti D.U.K. - dėstytojas paspaudžia ant nuorodos leidžiančios sukurti D.U.K.
2. Informacija apie klausimą? - D.U.K. UI išmeta dėstytojui D.U.K. formą užpildymui.
3. Klausimo informacija - naudotojas išsiunčia užpildytą formą D.U.K. UI.
4. Sukurti klausimą - įvesta klausimo informacija yra siunčiama kontrolieriui.
5. Pridėti klausimą - D.U.K. UI įdeda klausimą į duomenų bazę.
6. Klausimas pridėtas - duomenų bazė parsunčia klausimo patalpinimo patvirtinimą.
7. Gauti D.U.K. sąrašą - D.U.K. kontroleris prašo duomenų bazės pateikti naują D.U.K. sąrašą.
8. D.U.K. sąrašas - D.U.K. sąrašas keliauja iš duomenų bazės iki dėstytojo.

### 3.2.12. Užduoties "Peržiūrėti D.U.K." scenarijus

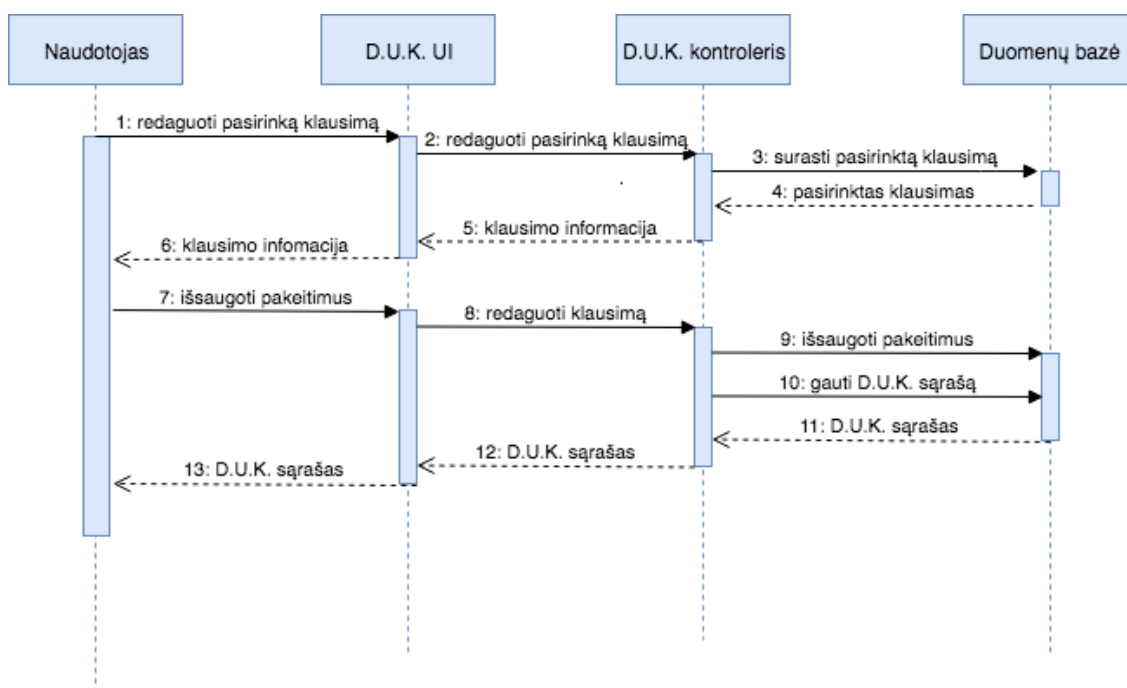


30 pav. Užduoties "Peržiūrėti D.U.K." scenarijus

#### Žingsnių seka (30pav.)

1. Gauti D.U.K. sąrašą - dėstytojas paspaudžia nuorodą į D.U.K. sąrašą.
2. Gauti D.U.K. sąrašą - prašymas gauti D.U.K. sąrašą keliauja iki duomenų bazės.
3. D.U.K. sąrašas - D.U.K. sąrašas iš duomenų bazės keliauja iki naudotojo.

### 3.2.13. Užduoties "Redaguoti pasirinktą klausimą" scenarijus



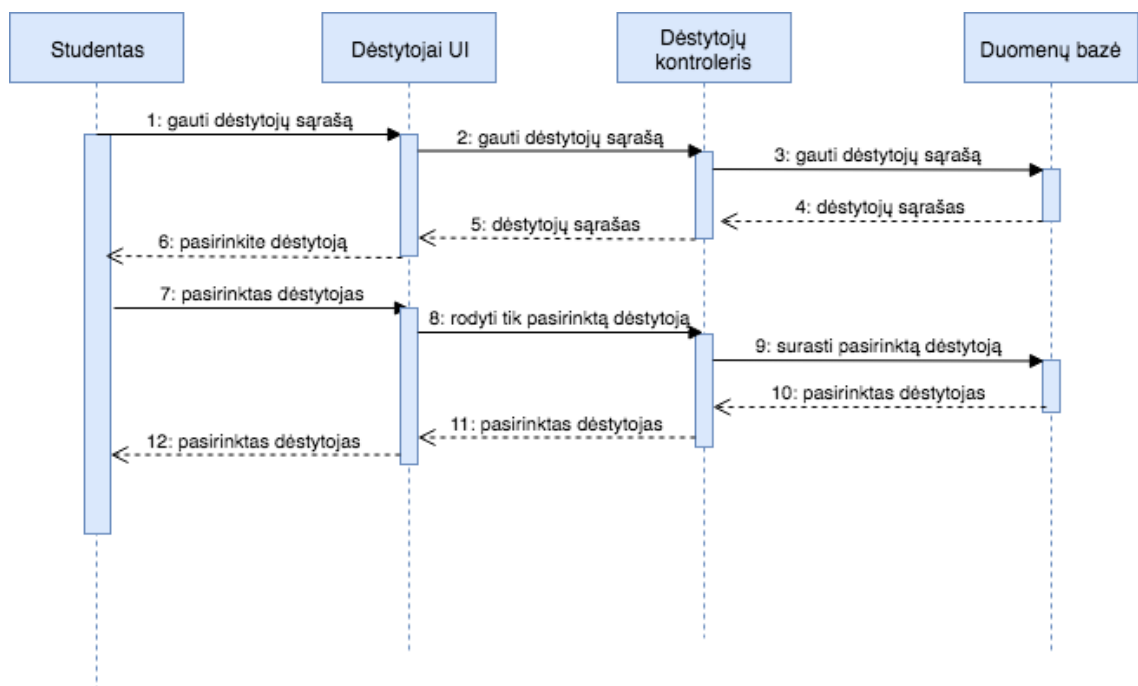
31 pav. Užduoties "Redaguoti pasirinktą klausimą" scenarijus

#### Žingsnių seka (31pav.)

1. Redaguoti pasirinktą klausimą - dėstytojas paspaudžia nuorodą leidžiančią redaguoti klausimą.
2. Redaguoti pasirinktą klausimą - pateikiamas klausimas D.U.K. kontrolieriui.
3. Surasti pasirinktą klausimą - klausimas ieškomas duomenų bazėje.

4. Pasirinktas klausimas - duomenų bazė pateikia klausimą kontrolieriui.
5. Klausimo informacija - informacija apie klausimą keliauja iki naudotojo.
6. Klausimo informacija - informacija apie klausimą keliauja iki naudotojo.
7. Išsaugoti pakeitimus - naudotojas prašo išsaugoti įvykdytus pakeitimus.
8. Redaguoti klausimą - klausimas siunčiamas redagavimui.
9. Išsaugoti pakeitimus - pakeitimai išsaugomi duomenų bazėje.
10. Gauti D.U.K. sąrašą - kontrolieris prašo duomenų bazės gauti atnaujintą D.U.K. sąrašą.
11. D.U.K. sąrašas - D.U.K. sąrašas keliauja iš duomenų bazės iki dėstytojo.

### 3.2.14. Užduoties "Peržiūrėti pasirinkto dėstytojo teikiamą informaciją" scenarijus

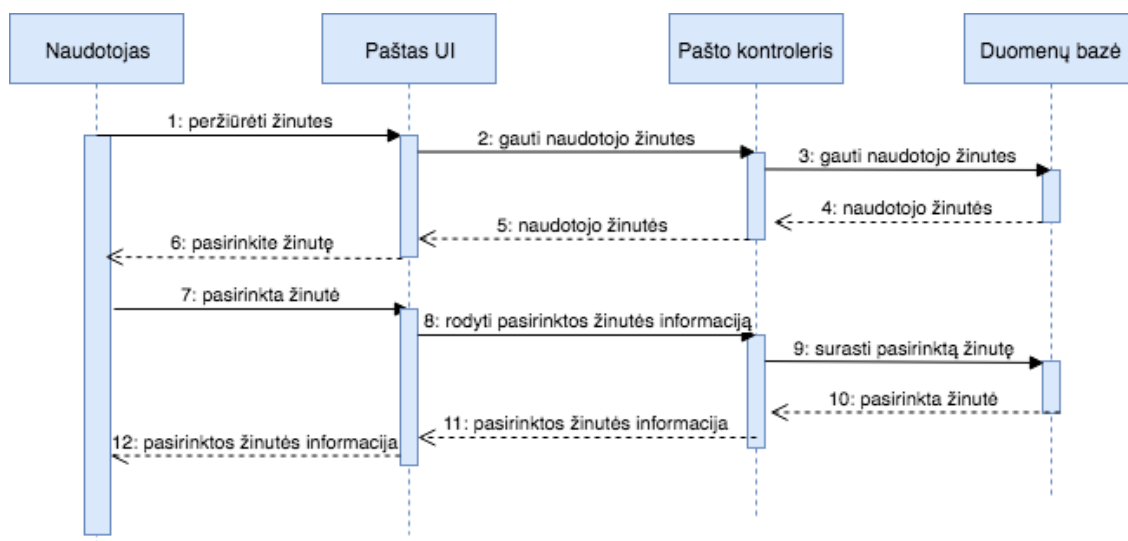


32 pav. Užduoties "Peržiūrėti pasirinkto dėstytojo teikiamą informaciją" scenarijus

### Žingsnių seka (32pav.)

1. Gauti dėstytojų sąrašą - studentas paspaudžia nuorodą į dėstytojų sąrašą.
2. Gauti dėstytojų sąrašą - prašymas gauti dėstytojų sąrašą keliauja iki duomenų bazės.
3. Gauti dėstytojų sąrašą - prašymas gauti dėstytojų sąrašą keliauja iki duomenų bazės.
4. Dėstytojų sąrašas - duomenų sąrašas keliauja iki naudotojo.
5. Dėstytojų sąrašas - duomenų sąrašas keliauja iki naudotojo.
6. Pasirinkite dėstytoją - prašoma pasirinkti, kurio dėstytojo puslapį norima matyti.
7. Pasirinktas dėstytojas - pasirenkamas dėstytojas, kurio informacija domina.
8. Rodyti tik pasirinktą dėstytoją - dėstytojas siunčiamas kontrolieriui.
9. Surasti pasirinktą dėstytoją - D.U.K. kontrolieris prašo duomenų bazės pateikti dėstytojo puslapį.
10. Pasirinktas dėstytojas - dėstytojo puslapis keliauja iš duomenų bazės iki studento.

### 3.2.15. Užduoties "Peržiūrėti pasirinktą žinutę" scenarijus

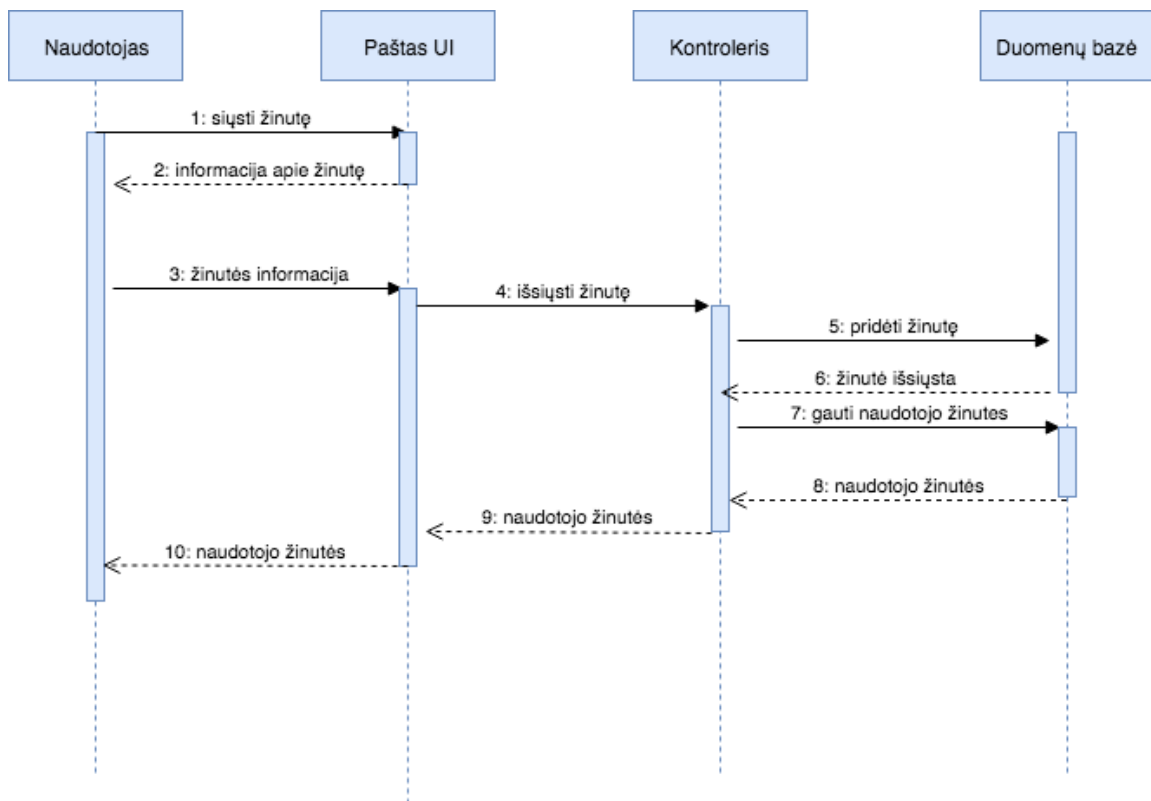


33 pav. Užduoties "Peržiūrėti pasirinktą žinutę" scenarijus

#### Žingsnių seka (33pav.)

1. Peržiūrėti žinutes - naudotojas paspaudžia nuorodą peržiūrėti žinutes.
2. Gauti naudotojo žinutes - prašymas gauti žinutes keliauja iki duomenų bazės.
3. Gauti naudotojo žinutes - prašymas gauti žinutes keliauja iki duomenų bazės.
4. Naudotojo žinutės - žinučių sąrašas keliauja iki naudotojo.
5. Naudotojo žinutės - žinučių sąrašas keliauja iki naudotojo.
6. Pasirinkite žinutę - leidžiama pasirinkti, kurią žinutę norima matyti.
7. Pasirinkta žinutė - pasirenkama žinutė, kurios informacija domina.
8. Rodyti pasirinktos žinutės informaciją - žinutė siunčiama kontrolieriui.
9. Surasti pasirinktą žinutę - Pašto kontrolieris prašo duomenų bazės pateikti žinutę.
10. Pasirinkta žinutė - žinutė keliauja iš duomenų bazės kontrolieriui.
11. Pasirinktos žinutės informacija - žinutė keliauja iš duomenų bazės naudotojui.

### 3.2.16. Užduoties "Siųsti žinutę" scenarijus

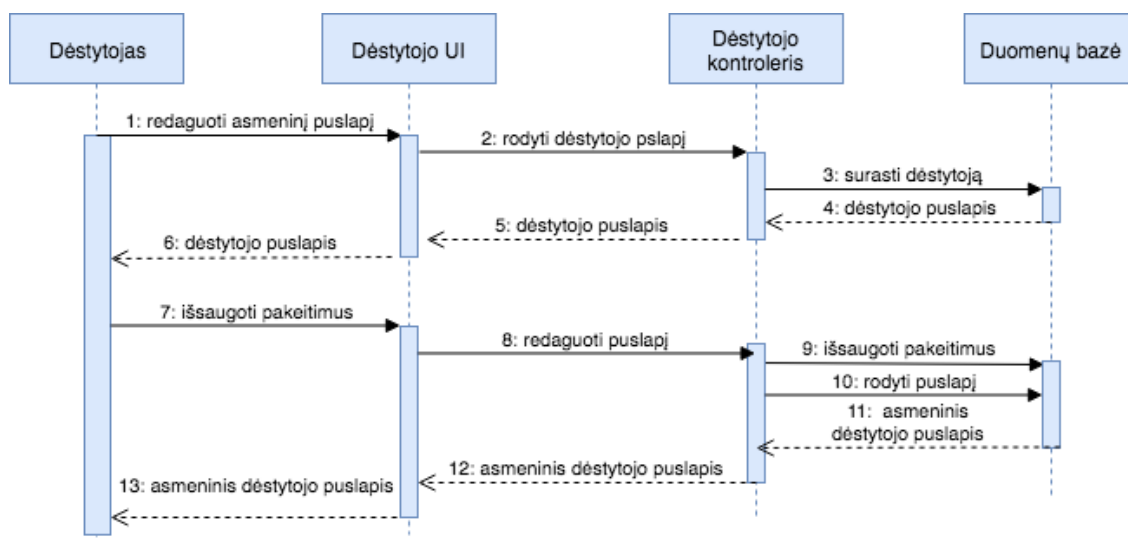


34 pav. Užduoties "Siųsti žinutę" scenarijus

#### Žingsnių seka (34pav.)

1. Siųsti žinutę - naudotojas paspaudžia nuorodą siųsti žinutę.
2. Informacija apie žinutę - pašto UI išmeta naudotojui žinutės formą užpildymui.
3. Žinutės informacija - naudotojas suveda žinutės informaciją ir patvirtina siuntimą.
4. Išsiųsti žinutę - įvesta informacija yra siunčiama kontrolieriui.
5. Pridėti žinutę - pašto UI įdeda žinutę į duomenų bazę.
6. Žinutė išsiųsta - kontroleris gauna patvirtinimą apie žinutės išsiuntimą.
7. Gauti naudotojo žinutes - kontroleris prašo duomenų bazės gauti naudotojo žinutes.
8. Naudotojo žinutės - naudotojo žinučių sąrašas keliauja iš duomenų bazės naudotojui.

### 3.2.17. Užduoties "Redaguoti asmeninį dėstytojo puslapį" scenarijus



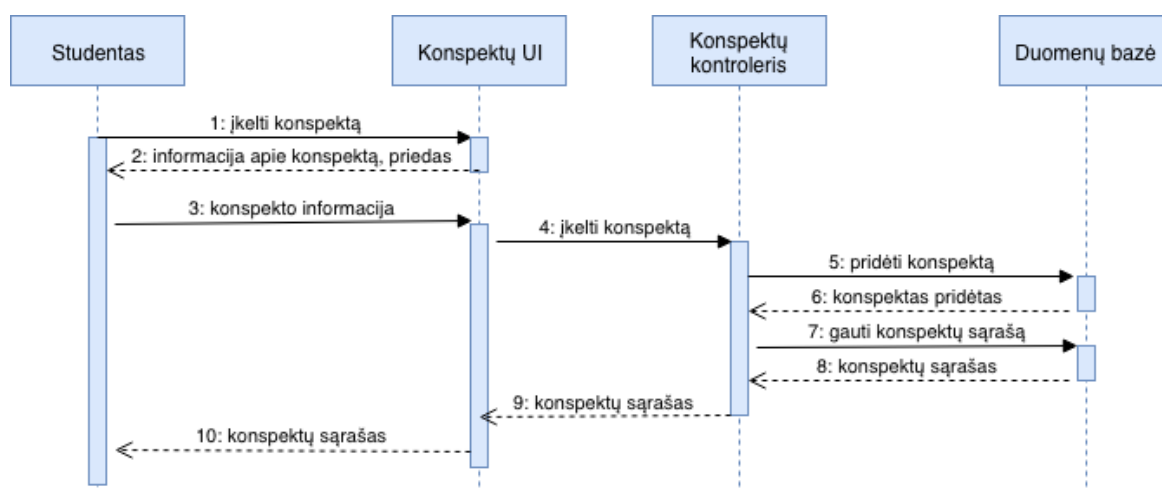
35 pav. Užduoties "Redaguoti asmeninį dėstytojo puslapį" scenarijus

#### Žingsnių seka (35pav.)

1. Redaguoti asmeninį puslapį - dėstytojas paspaudžia nuorodą redaguoti asmeninį puslapį.
2. Rodyti dėstytojo puslapį - prašymas rodyti puslapį keliauja į kontrolierį.
3. Surasti dėstytoją - kontrolieris pateikia prašymą duomenų bazei surasti prisijungusį dėstytoją.
4. Dėstytojo puslapis - rasto dėstytojo puslapio informacija keliauja dėstytojui.
5. Išsaugoti pakeitimus - atlikęs pakeitimus savo puslapyje, dėstytojas paspaudžia nuorodą išsaugoti.
6. Redaguoti puslapį - įvesta informacija yra siunčiama kontrolieriui.
7. Išsaugoti pakeitimus - dėstytojo kontrolieris įdeda pakeistą informaciją į duomenų bazę.
8. Rodyti puslapį - kontrolieris prašo duomenų bazės gauti redaguotą dėstytojo puslapio informaciją.
9. Asmeninis dėstytojo puslapis - dėstytojo puslapio informacija keliauja iš duomenų bazės dėstytojui.



### 3.2.18. Užduoties ”Įkelti konspektą” scenarijus

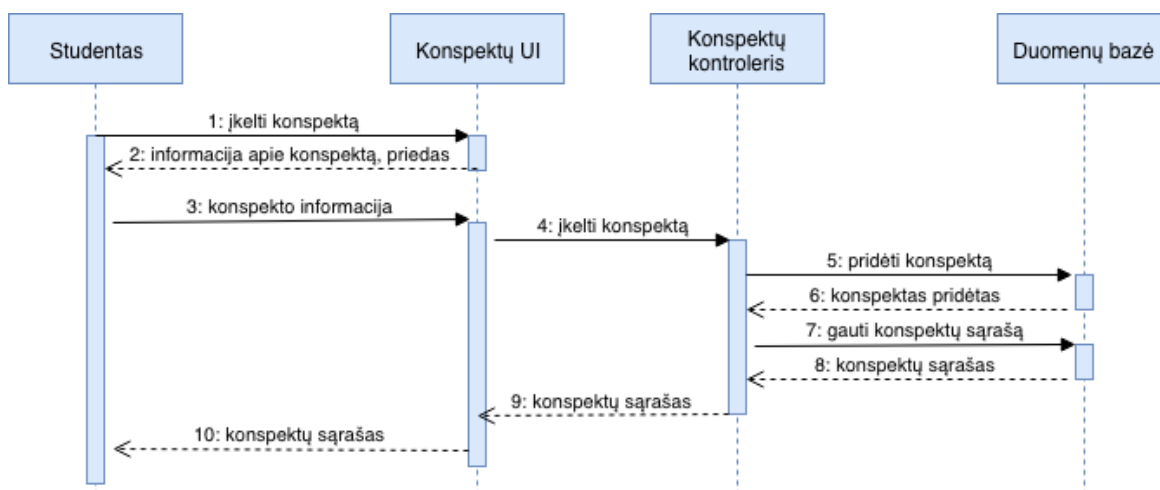


36 pav. Užduoties ”Įkelti konspektą” scenarijus

#### Žingsnių seka (36pav.)

1. Įkelti konspektą - studentas paspaudžia nuorodą įkelti konspektą.
2. Informacija apie konspektą, priedas - konspektų UI išmeta studentui konspekto formą užpildymui, bei pasirinkimui failo, kurį norima pridėti.
3. Konspekto informacija - naudotojas išsiunčia užpildytą formą Konspektų UI.
4. Įkelti klausimą - pateikta konspekto informacija yra siunčiama kontrolieriui.
5. Pridėti konspektą - Konspektų UI įdeda konspektą į duomenų bazę.
6. Konspektas pridėtas - duomenų bazė parsiunčia konspekto patalpinimo patvirtinimą.
7. Gauti konspektų sąrašą - konspektų kontrolieris prašo duomenų bazės pateikti naują konspektų sąrašą.
8. Konspektų sąrašas - Konspektų sąrašas keliauja iš duomenų bazės iki studento.

### 3.2.19. Užduoties "Peržiūrėti konspektus" scenarijus



37 pav. Užduoties "Peržiūrėti konspektus" scenarijus

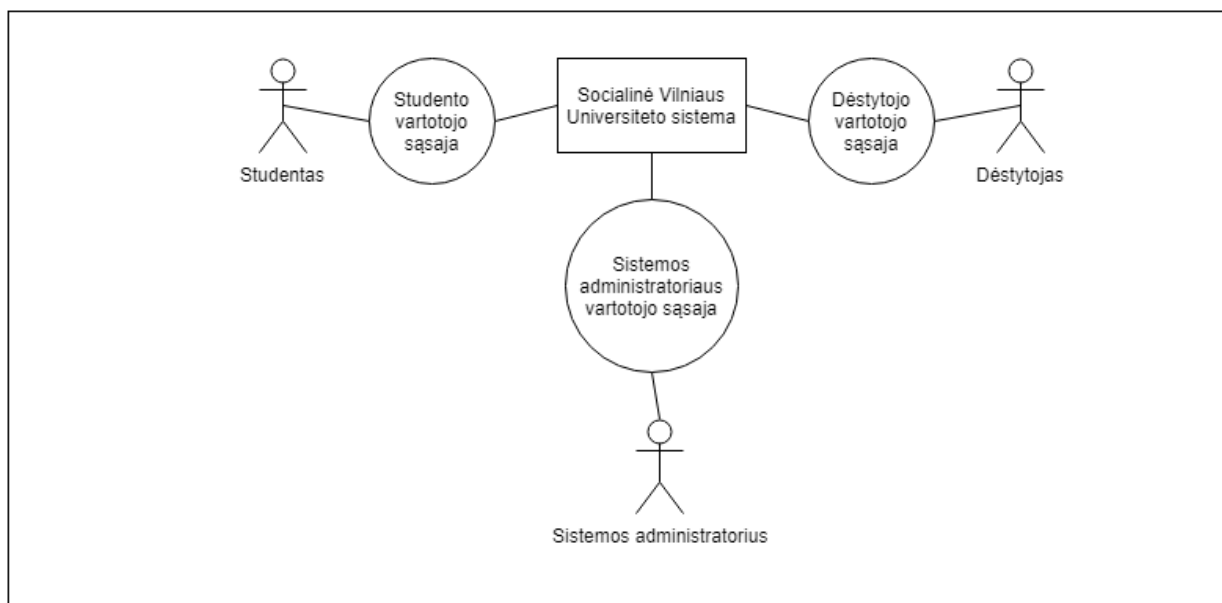
#### Žingsnių seka (37pav.)

1. Gauti konspektų sąrašą - studentas paspaudžia nuorodą į konspektų sąrašą.
2. Gauti konspektų sąrašą - prašymas gauti konspektų sąrašą keliauja iki duomenų bazės.
3. Konspektų sąrašas - konspektų sąrašas iš duomenų bazės keliauja iki naudotojo.

### 3.3. KŪRIMO PJŪVIS

Programų sistemos komponentai yra vaizduojami trimis lygmenimis: nuliniu, pirmuoju ir antruoju. Toks komponentų pateikimas leidžia išsamiau apibrėžti sistemos fizinius komponentus, jų konfigūraciją bei tarpusavio ryšius. Komponentų diagramos, atvaizduodamos struktūrą, priklausomybes bei sąsajas, leidžia susidaryti fizinį sistemos vaizdą. Taip pat suteikia galimybę apžvelgti išoriškai matomą komponentų elgseną. Komponentai atvaizduojami naudojant UML komponentų diagramas.

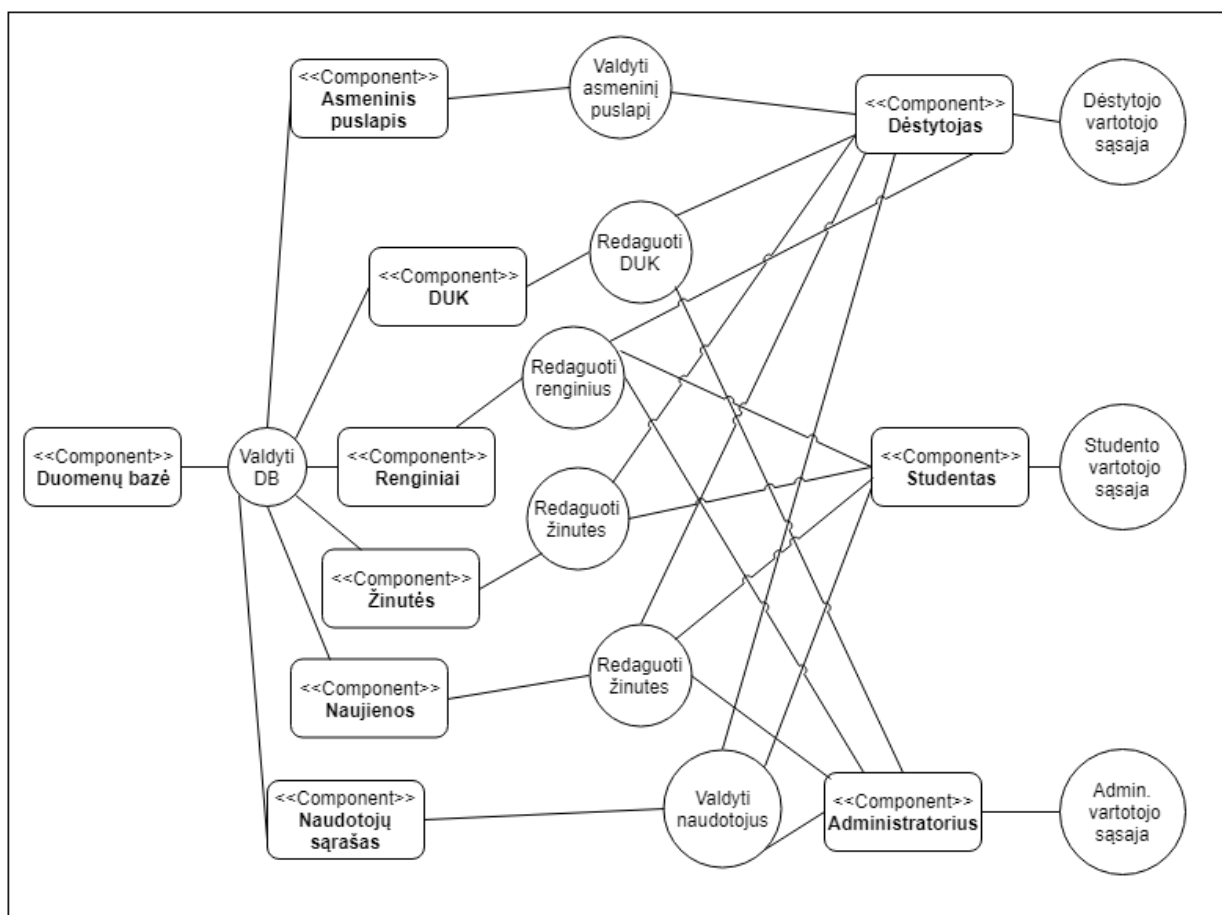
#### 3.3.1. Komponentų diagramos nulinis lygmuo



38 pav. Komponentų diagramos nulinis lygmuo

Komponentų diagramos nuliniame lygmenyje (19 pav.) vaizduojamas bendras komponentų vaizdas. Pagrindinis ir vienintelis šio lygio komponentas yra "Socialinė Vilniaus Universiteto sistema". Šis komponentas sąveikauja su keliomis vartotojo sąsajomis. Studento ir dėstytojo grafinė vartotojo sąsaja įgalina šiuos vartotojus naudotis sistema.

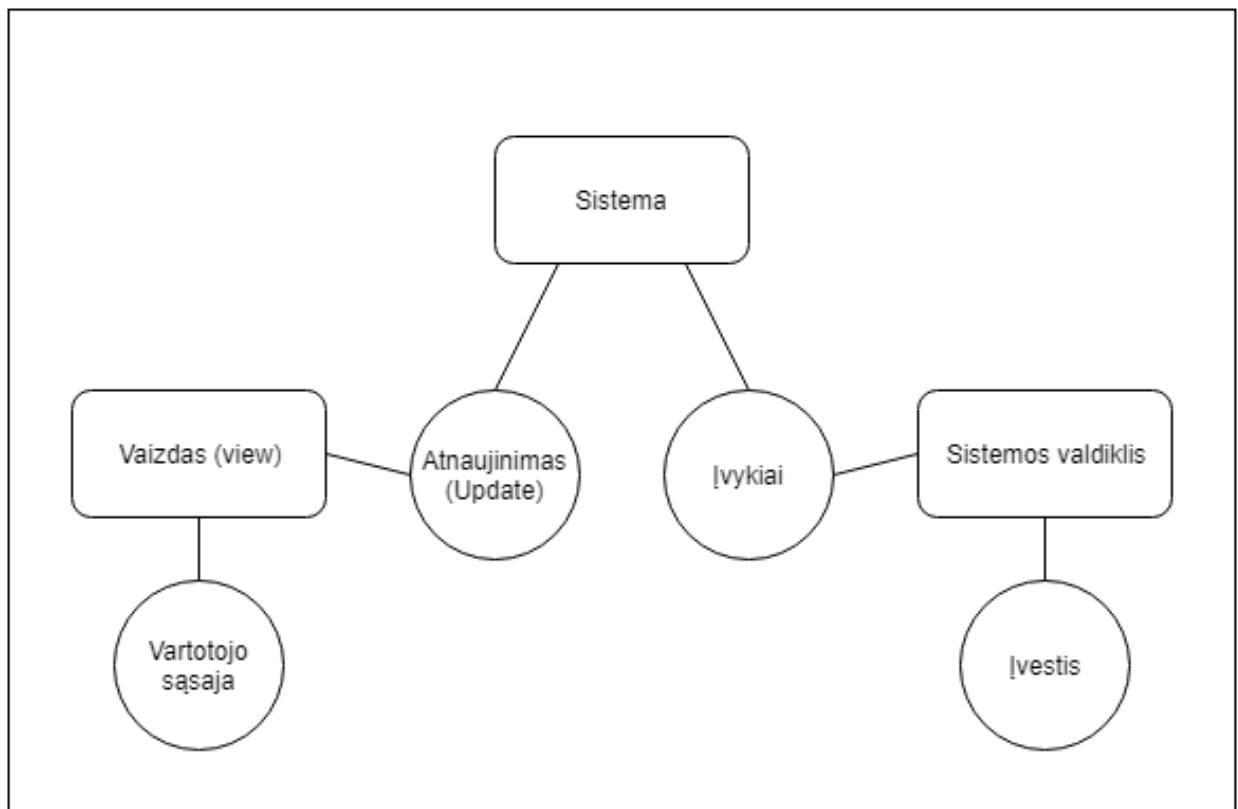
### 3.3.2. Komponentų diagramos pirmasis lygmuo



39 pav. Komponentų diagramos pirmasis lygmuo

Komponentų diagramos pirmame lygmenyje komponentų diagrama yra suskaidoma. Socialinė Vilniaus universiteto sistema yra suskaidoma į šiuos komponentus: Duomenų bazė, Asmeninis puslapis, DUK, Renginiai, Žinutės, Naujienos, Naudotojų sąrašas. Šiame lygyje kiekviena vartotojo sąsaja turi už ją atsakingus komponentus. Taip pat kiekvienas komponentas turi sąsajas su kitais komponentais tam, jog galėtų vykti sąveika ir keitimasis paslaugomis. Tokiu būdu yra užtikrinama visapusiška komponentų realizacija bei tarpusavio darna.

### 3.3.3. Komponentų diagramos antrasis lygmuo



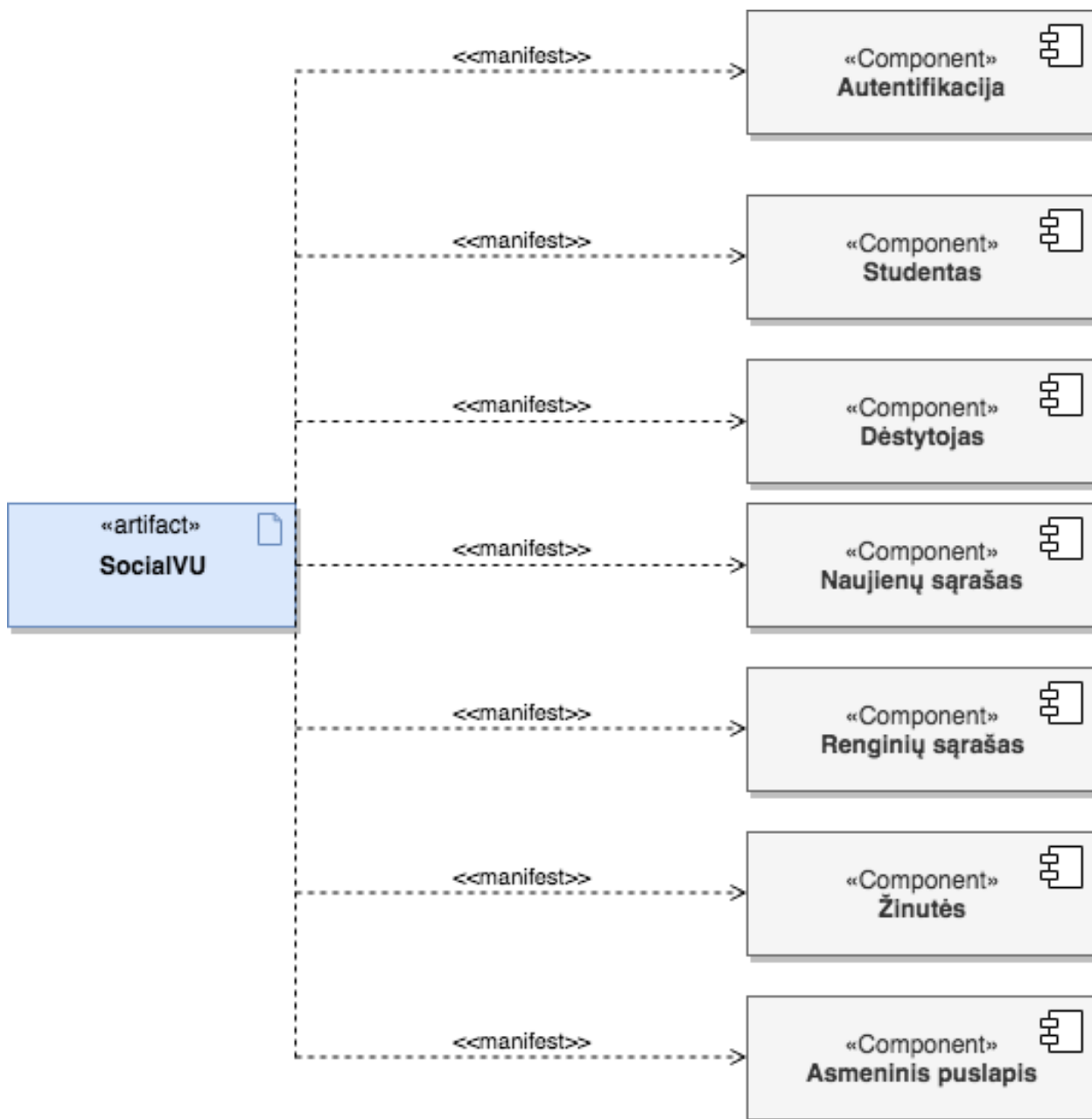
40 pav. Komponentų diagramos antrasis lygmuo

Antrame lygmenyje dekomponavimui buvo pasitelktas MVC dizaino šablonas. Šis modelis buvo pasirinktas dėl jo paprastumo, universalumo ir populiarumo. MVC šabloną sudaro trys pagrindiniai komponentai: Model, View ir Controller. Model (šiuo atveju, mūsų Sistema) – pagrindinis šablono komponentas, jis atsakingas už visos sistemos elgesį problemineje situacijoje, nepriklausomai nuo vartotojo sąsajos, taip pat jis atsakingas už duomenis, logiką ir taisykles. Vaizdas (view) – komponentas, kuris yra atsakingas už informacijos atvaizdavimą, jos atnaujinimą. Valdiklis – komponentas, kuris rūpinasi duomenų įvestimi ir įvesties apdorojimu.

### 3.4. FIZINIS PJŪVIS

Fizinis pjūvis sudarytas iš dislokavimo diagramų. Šiose diagramose vaizduojamas programos komponentų išdėstymas tinkle bei komunikacijos protokolai tarp jų.

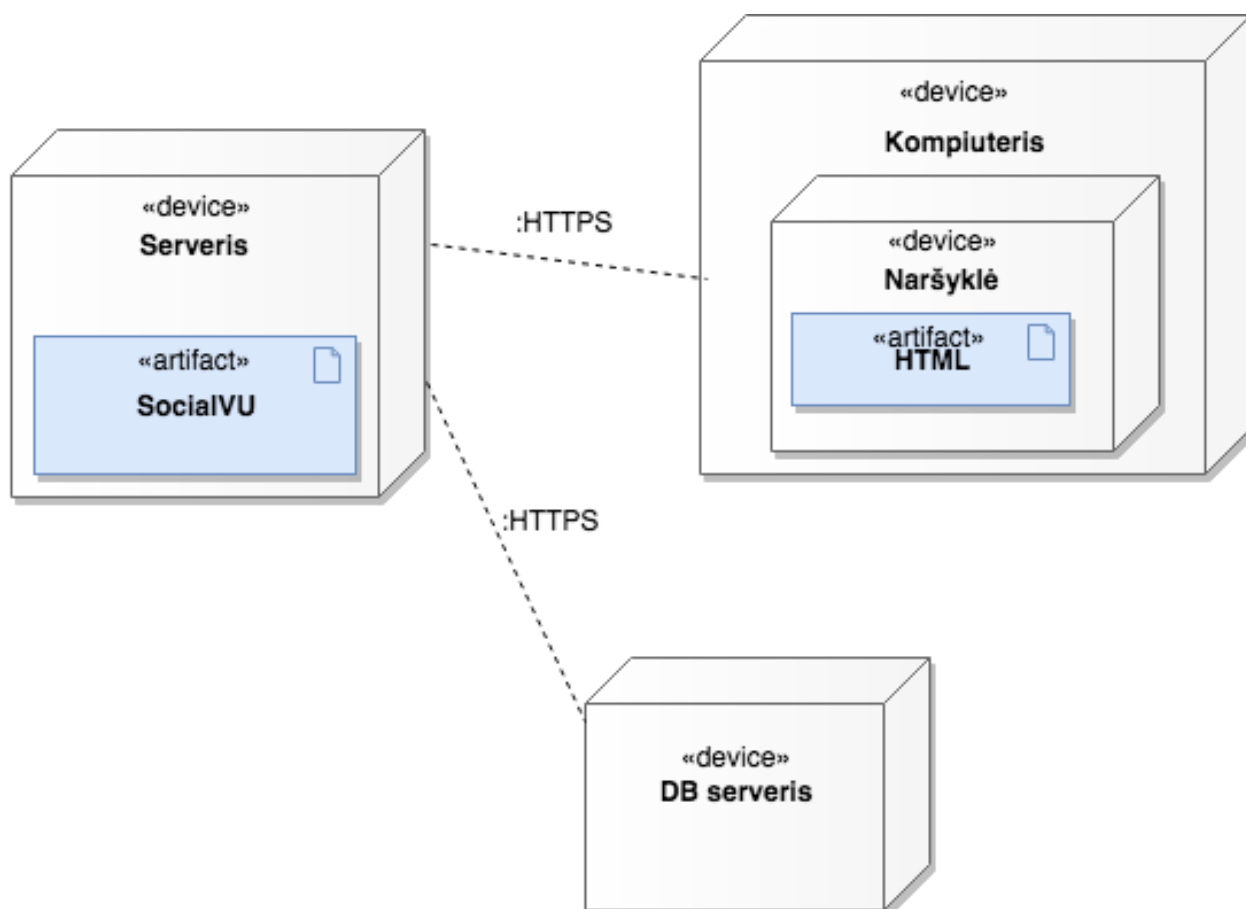
#### 3.4.1. Dislokavimo diagrama nr. 1 (komponentų ir artefaktų ryšių diagrama)



41 pav. Komponentų ir artefaktų ryšių diagrama

Komponentų ir artefaktų ryšių diagramoje (41pav.) vaizduojamas artefaktas SocialVU, kuris įgyvendina šiuos programos komponentus: Autentifikacija, Studentas, Dėstytojas, Naujienų sąrašas, Renginių sąrašas, Žinutės, Asmeninis puslapis.

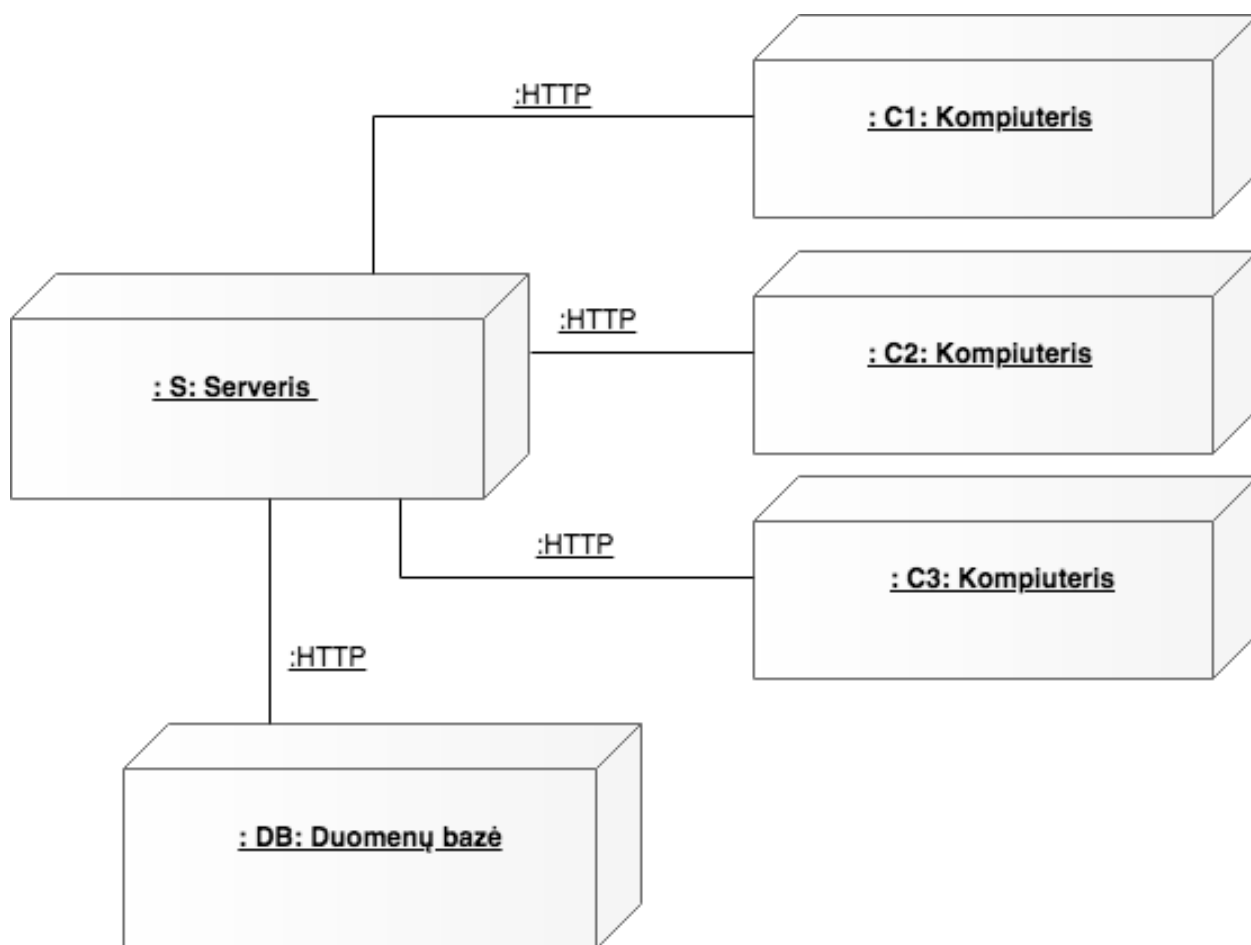
### 3.4.2. Dislokavimo diagrama nr. 2 (mazgų ir artefaktų ryšių diagrama)



42 pav. Mazgų ir artefaktų ryšių diagrama

Mazgų ir artefaktų diagramoje (42pav.) parodo, kad tinklalapis yra serveryje, kuris bendrauja http protokolu su SQL serveriu, kuriame saugoma duomenų bazė. Naudotojai turi galimybę pasiekti sistemos teikiamas paslaugas savo pasirinkta naršykle, kuri palaiko http protokolą.

### 3.4.3. Dislokavimo diagrama nr. 3 (mazgų ir artefaktų egzempliorių diagrama)



43 pav. Mazgų ir artefaktų egzempliorių diagrama

43pav. vaizduojamas įrenginių (mazgų) išsidėstymas tinkle. Serveris turi tiesioginį ryšį su duomenų baze, o naudotojai gali prisijungti prie serverio. Tačiau naudotojai negali tiesiogiai pasiekti duomenų bazės ir joje saugomų duomenų.

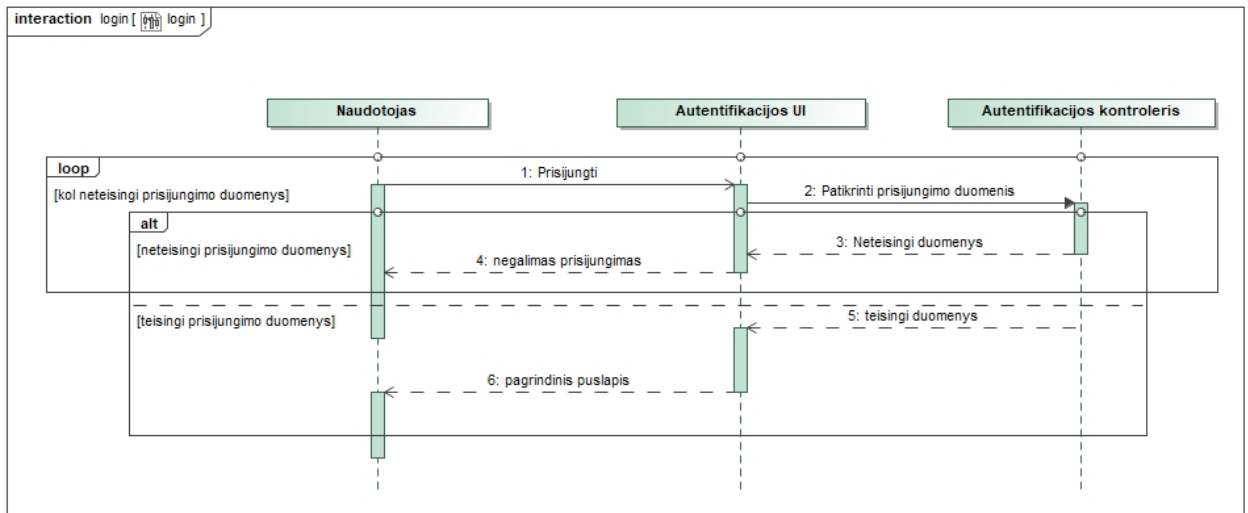


### 3.5. PROCESO PJŪVIS

Procesų pjūvis sudarytas iš sekų ir veiklos diagramų. Diagramose parodoma, kokie procesai vyksta sistemoje bei išreiškiama komunikacija tarp jų.

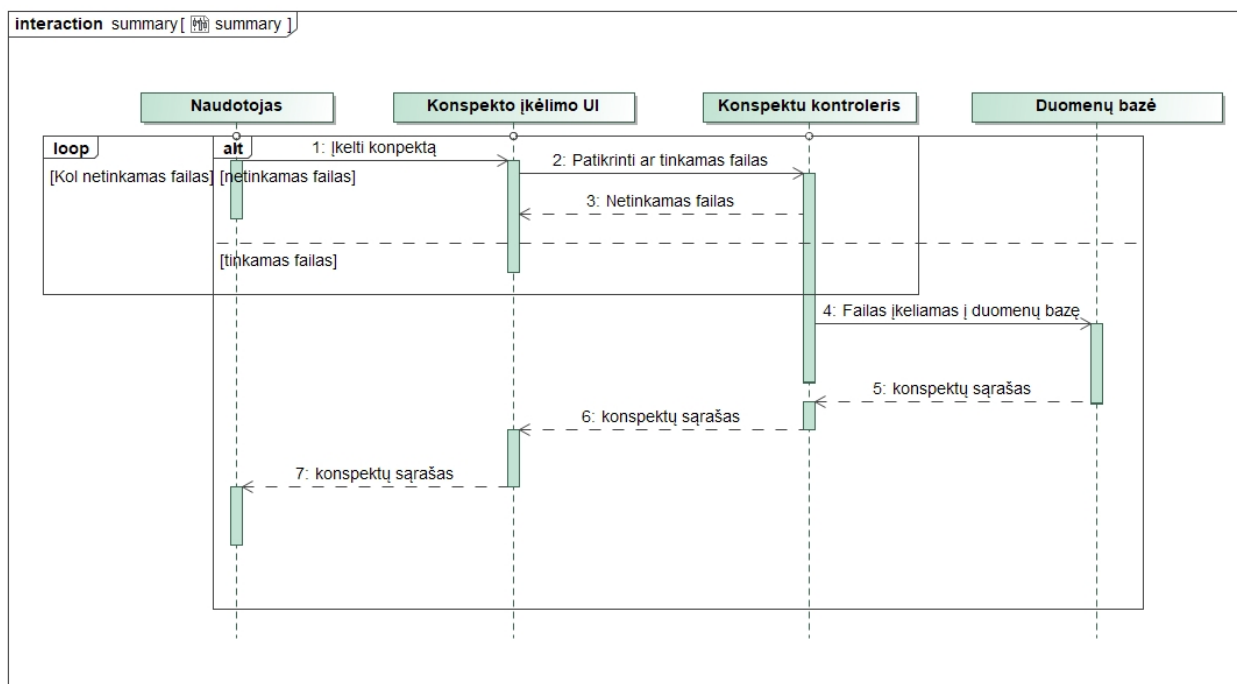
#### 3.5.1. Proceso sekų diagramos

Procesų sekų diagramose, atspindi procesai, kurie yra vykdomi sistemoje. Iš proceso sekų diagramos galima matyti, kokie komponentai dalyvauja vykdyje, kaip procesas vykdomas.



44 pav. Proceso „Prisijungimas” sekų diagrama

Pagal 44 pav. diagramą matoma, kad procesas prasideda naudotojo paspaudimu ant nuorodos įgalinančios prisijungimą. Autentifikacijos UI gautus duomenis siunčia patikrinimui į autentifikacijos kontrolierį. Iš kontrolierio gaunamas atsakymas, ar duomenys teisingi ar ne. Jei duomenys klaidingi, naudotojui išmetamas pranešimas, jog prisijungti negalima ir jis vėl gali kartoti prisijungimo procesą. Jei duomenys teisingi, naudotojas yra nukreipiamas į pagrindinį puslapį.

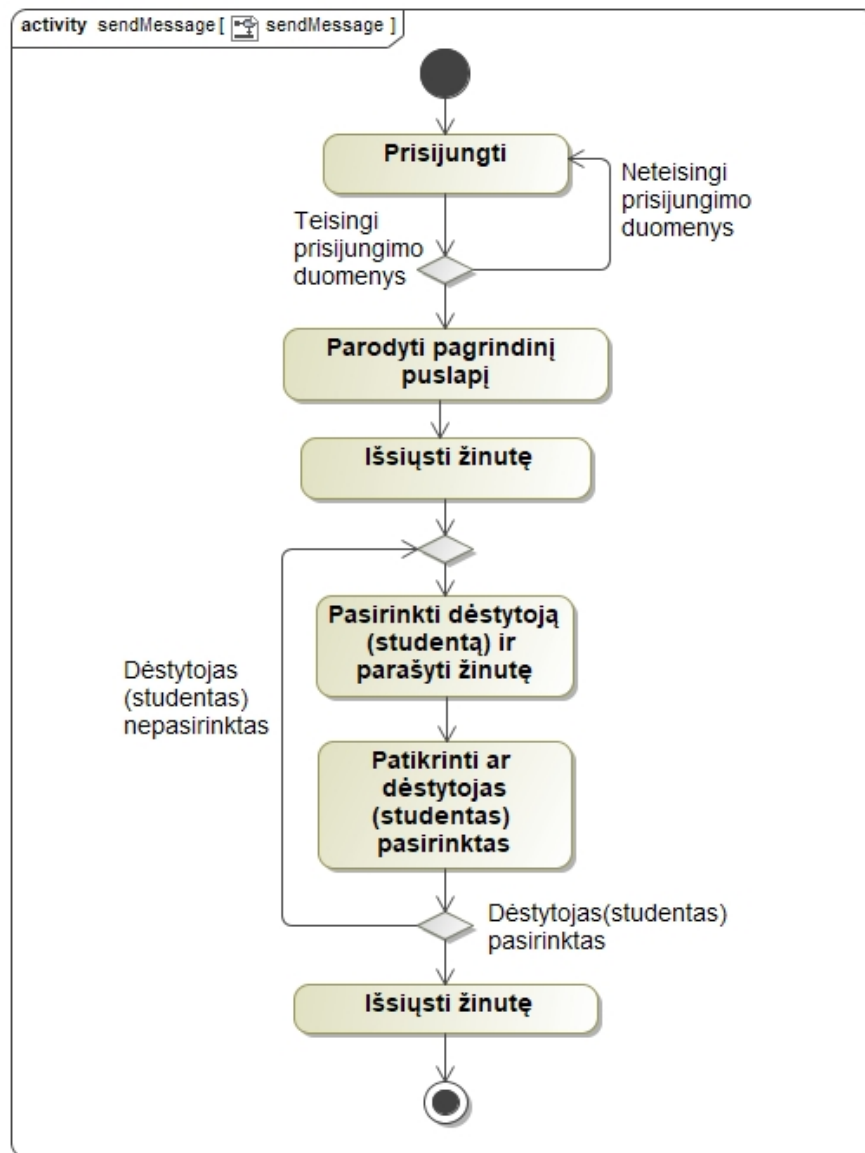


45 pav. Proceso „Konspekto įkėlimas” sekų diagrama

Pagal 45 pav. diagramą matoma, kad procesas prasideda naudotojo paspaudimu ant nuorodos įgalinančios konspekto įkėlimą. Konspekto įkėlimo UI gautus duomenis siunčia patikrinimui į konspektų kontrolierį. Iš kontrolierio gaunamas atsakymas, ar failas tinkamas ar ne. Jei failas netinkamas, naudotojui išmetamas pranešimas, jog failas netinkamas ir jis vėl gali kartoti konspekto įkėlimo procesą. Jei failas tinkamas, jis yra įkeliamas į duomenų bazę, o naudotojas yra nukreipiamas į konspektų puslapį.

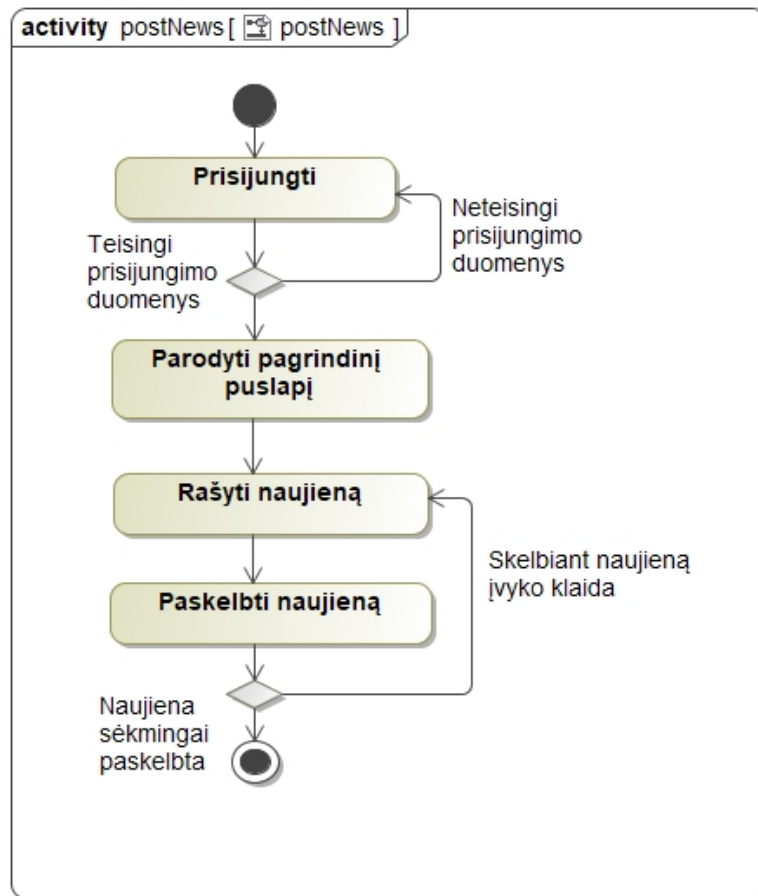
### 3.5.2. Veiklos diagramos

Veiklos diagramos padeda suprasti dinaminį sistemos veikimą, parodo, kokie veiksmai atliekami vykdant konkrečią veiklą, galimus vykdymo atvejus.



46 pav. Žinutės išsiuntimo veiklos diagrama

46 pav. diagramoje matomas, žinutės išsiuntimo dėstytojui procesas. Procesas prasideda naudotojo nuorodos paspaudimu, kreipiančios į žinutės išsiuntimo formą. Naudotojas pateikia parašo žinutę bei pasirenka dėstytoją, tuomet duomenys siunčiami patikrinimui. Jei dėstytojas nebuvo pasirinktas, naudotojas nukreipiamas atgal į žinutės siuntimo formą. Jei duomenys atitinka visus reikalavimus, tada žinutė išsiunčiama.



47 pav. Naujienos paskelbimo veiklos diagrama

47 pav. diagramoje matomas, naujienos paskelbimo procesas. Procesas prasideda naudotojo prisijungimu. Neteisingai suvedus prisijungimo duomenis naudotojas vėl nukreipiamas į prisijungimą, kitu atveju jis nukreipiamas į pagrindinį puslapį bei pasirenka naujienos paskelbimo nuorodą. Naudotojas parašo naujieną bei ją paskelbia. Jei skelbiant naujieną įvyksta klaida jis nukreipiamas į naujienos rašymo formą, kitu atveju naujiena paskelbiama.

## SAVOKOS

Šiame skyriuje pateikiami specifiniai mūsų projekte naudotų žodžių paaiškinimai.

**Sistema** – fakulteto vidinis socialinis tinklas, dokumente vadinamas „SocialVU“.

**Studentas** – vartotojas, kuris turim ribotas galimybes turinio valdymo atžvilgiu.

**Dėstytojas** – vartotojas, kuris turi galimybę redaguoti visą socialinio tinklo turinį.

**Administratorius** – vartotojas, kuris turi galimybę redaguoti visą socialinio tinklo turinį, išskyrus dėstytojus.

## REZULTATAI

Dokumente pateiktas verslo proceso aprašas, kuriamai sistemai atlikta išorinė bei vidinė verslo proceso analizė. Iškelti tikslai, jog sistema naudotųsi bent pusė Vilniaus universiteto studentų. Pateiktos UML diagramos dalykinei sričiai, užduočių veiklai, klasėms.

Atlikus įgyvendinamumo ir naudos analizę paaiškėjo, jog preliminarus metinis sistemos pelnas turėtų būti 31284€. atsižvelgiant į sistemos kūrimui ir palaikymui reikalingas išvadas, sistema turėtų tapti pelninga po 1.76 metų.

Pateikti sistemos naudojimo scenarijai. Nurodyti pagrindinių funkcijų modeliai, atvaizduojantys, kaip pagrindiniai sistemos agentai (dėstytojai, studentai bei administratoriai) naudosis sistema.

Pateikti aiškiai sunumeruoti ir apibrėžti kuriamo socialinio tinklapio funkciniai reikalavimai, o jų tarpusavio sąveika atvaizduota sekų diagramomis.

Pateikti aiškiai sunumeruoti ir apibrėžti kuriamo socialinio tinklapio funkciniai reikalavimai, o jų tarpusavio sąveika atvaizduota sekų diagramomis. Suformuluoti nefunkciniai reikalavimai. Reikalavimai išskirti į šias skiltis: vidinių interfeisų, veikimo, diegimo, aptarnavimo ir priežiūros, tiražuojamumo, apsaugos bei juridiniai. Vartotojo sąsajos reikalavimai apibrėžti metaforos reikalavimų lentele bei suformuluotomis užduotimis įvairiems scenarijams, taip pat pateikiami darnos ir standartizavimo bei pranešimų formulavimo reikalavimai.

Kuriamos sistemos architektūra išnagrinėta naudojant UML4+1 požiūrių rinkinį. Iš loginio, užduočių, kūrimo, fizinio bei procesų pjūvių matomas sistemos funkcionalumas, užduotys, kurias gali įgyvendinti naudotojas, sąsajos tarp atskirų sistemos komponentų, sistemos įranga bei dinaminis sistemos modelis.