

KAUNO TECHNOLOGIJOS UNIVERSITETAS
INFORMATIKOS FAKULTETAS

T120B029 Programų sistemų analizės ir projektavimo įrankiai
Projekto ataskaita nr. 3

Tema: Naudotų dalių pardavimo sistema

Studentų komanda:

Rimvydas Neverauskas

Valentinas Kasteckis

Henrikas Juzuitis

Eligijus Kiudys

Dėstytojai:

lekt. Jurgelaitis Mantas

Magylaitė Kristina

Ryselis Karolis

dr. Lina Čeponienė

KAUNAS, 2020

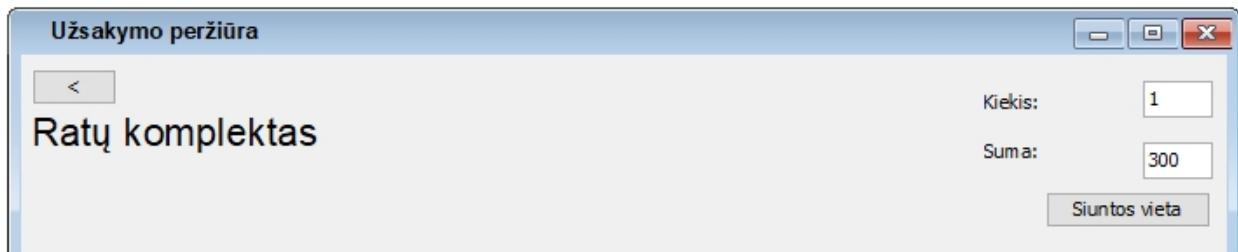
1 Kuriamos sistemos aprašymas

Kuriamos sistemos idėja: naudotų automobilio dalių pardavimo sistemą. Sistemoje bus galima pirkti, bei parduoti automobilių dalis. Sistemoje pirkėjas gali pasirinkti prekes, kurias galima įsidėti į krepšelį. Pasirinkus norimas nusipirkti dalis reikia apmokėti už krepšelį. Dalys yra surenkamos iš įvairių pardavėjų, bei kiekvienai daliai yra suteikiamas siuntos siuntimo kodas. Sékmingai užsisakius dalis, jas galima peržiūrėti užsakymų istorijoje. Taip pat, šių dalių buvimo vietą pirkėjas gali sekti sistemoje su pateiktais siuntos sekimo kodais.

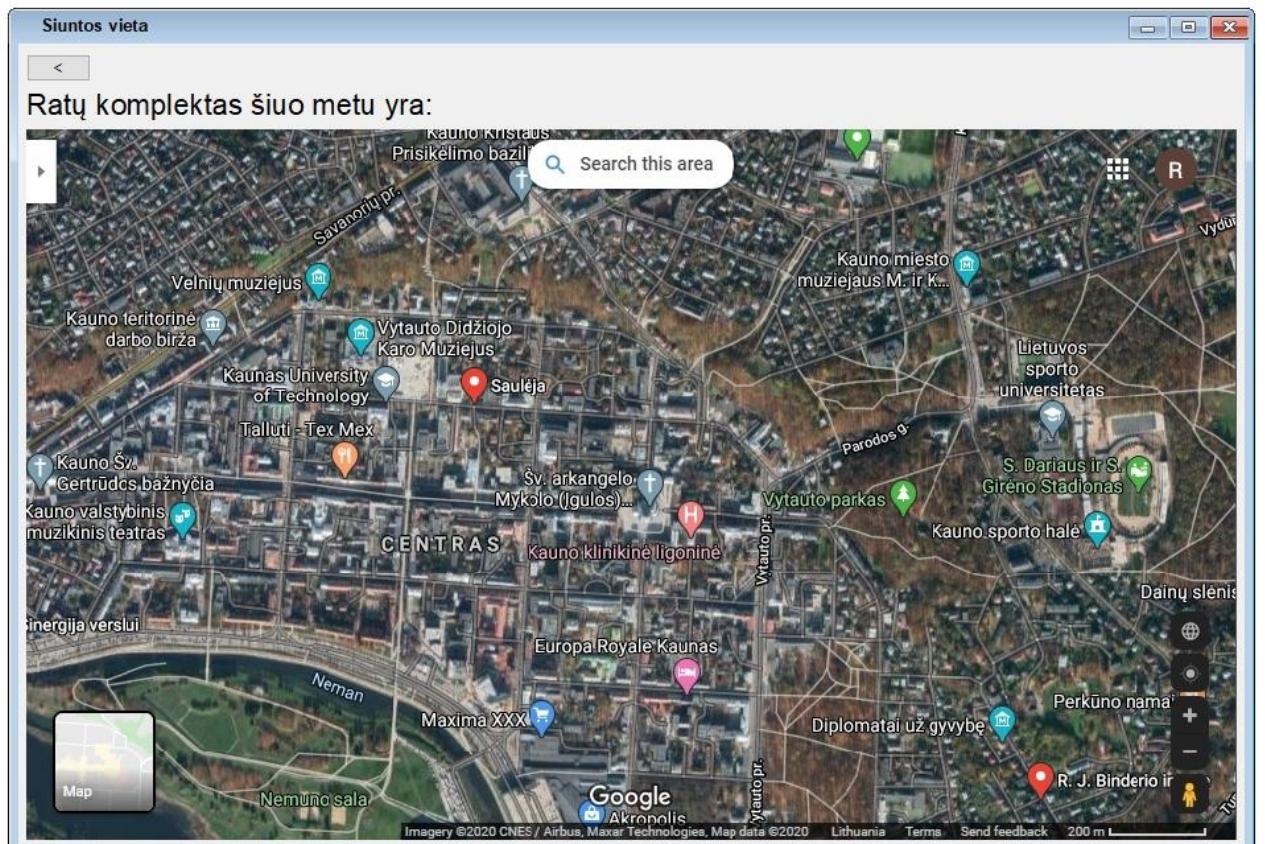
Ši sistema suteikia galimybę susirinkti viso automobilio dalis pagal markę. Pasirinkus automobilio markę, rodomas visų galimų dalių sąrašas. Pirkėjui pasirinkus visas norimas dalis, automobilis gali būti surenkamas mechaniko, kuris yra siūlomas mūsų puslapyje. Mechanikas taip pat turės savo darbo kalendorių, kuriame pirkėjas gali rezervuoti laiką, kada nori surinkti automobilį. Pirkėjas turi galimybę prekės užsakymo metu pasirinkti detalės pakeitimo paslaugą, kur taip pat galima pasirinkti rezervacijos laiką šios detalės pakeitimui. Sistemoje yra šie naudotojai:

- Svečias - gali tik prisiregistruoti ar prisijungti prie sistemos.
- Pirkėjas – gali užsisakyti dalis, jas parduoti, užsisakyti automobilio surinkimą, rezervuoti mechaniko laiką, tam tikros automobilio dalies keitimui.
- Mechanikas – turi darbo grafiką, iš kurio pirkėjas gali rezervuoti.
- Administratorius – patvirtina mechanikus sistemoje.

2 Sistemos naudotojo sasajos eskizai



pav. 1 Užsakymo peržiūros eskizas



pav. 2 Siuntos sekimo eskizas

Užsakymo valdymas

Atgal

Data Apmoketas užsakymas

Daliniai užsakymai

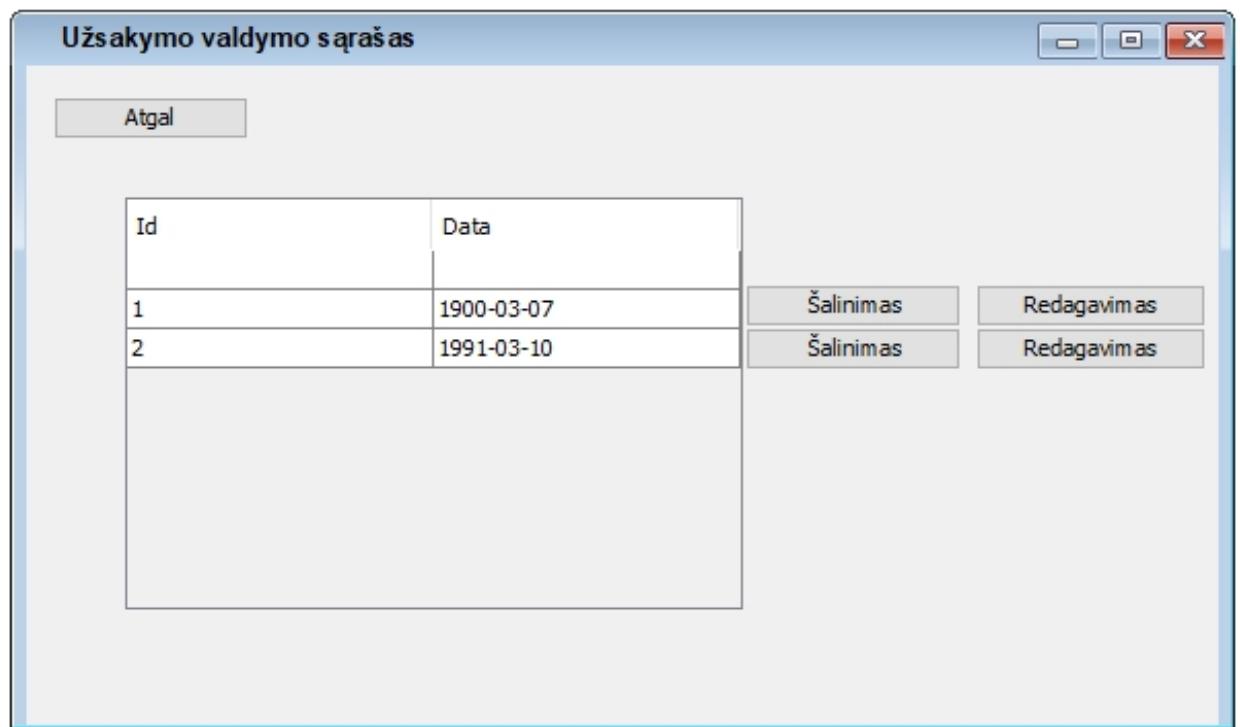
Id	Kaina	Sekimo Kodas

Mechanikas

Id	Kaina	Būsena

Išsaugoti

pav. 3 Užsakymo valdymo eskizas



pav. 4 Užsakymo valdymo sąrašo eskizas

Prekė

< Atgal

Prekės pavadinimas	Bižokas
Kategorija	Variklis
Kaina	20.31
Miestas	Kaunas
Adresas	Savanoriu pr. 13A

Apaščymas

Nuimtas nuo Golf 9. 2.33L 99kW.

Išsaugoti

pav. 5 Prekės eskizas

Prekių sąrašas

Prekė	Kaina	Miestas	Adresas	Patvirtinta
Babinas	12.32	Kaunas	Savanoriu pr. 13	Taip
Kardanas	250.34	Kaunas	Taikos pr. 214	Ne
Račnykas	23.41	Vilnius	Kauno g. 19	Taip
Piridačkas	70.12	Kaunas	Savanoriu pr. 13	Taip

< Atgal

Pridėti **Redaguoti** **Deaktyvuoti**

pav. 6 Prekių sąrašo eskizas

Nuolaidos aktyvacijos diagramma

Atgal

Įveskite nuolaidos kodą

Aktyvuoti

Aktyvuotos nuolaidos

Nuolaidos procentas	Pabaigos data
15%	2021-01-01

pav. 7 Nuolaidos aktyvacijos eskizas

Profilio diagrama

Atgal	Elektroninis paštas <input type="text"/>	Daugkartinės nuolaidos aktyvacija
	Vardas <input type="text"/>	
	Pavardė <input type="text"/>	
	Tel. numeris <input type="text"/>	
	Adresas <input type="text"/>	
Išsaugoti pakeitimus		
	Senas slaptažodis <input type="text"/>	
	Naujas slaptažodis <input type="text"/>	
	Pakartokite naują slaptažodį <input type="text"/>	
Atnaujinti slaptažodį		

pav. 8 Profilio diagramos eskizas

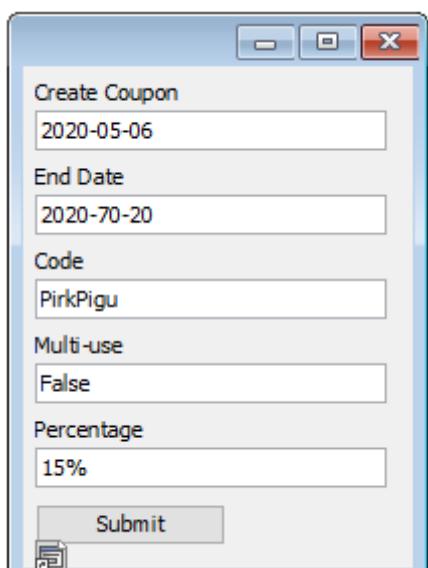
Coupons

[Add New Coupon](#)

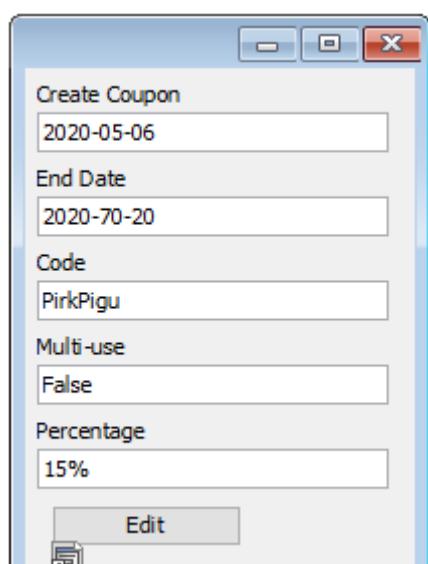
Start Date	End Date	Code	Multi-use	Percentage
2020-03-01	2020-08-28	Pirk	True	10%
2020-05-15	2020-06-28	Ispardavimas	False	25%
2020-02-01	2021-01-01	Pigu	False	15%

[Delete](#) [Edit](#)

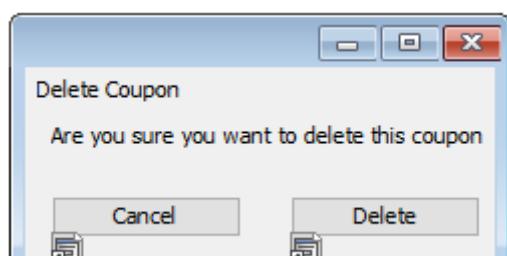
pav. 9 Kupony valdymo eskizas



pav. 10 Kupono pridėjimo eskizas



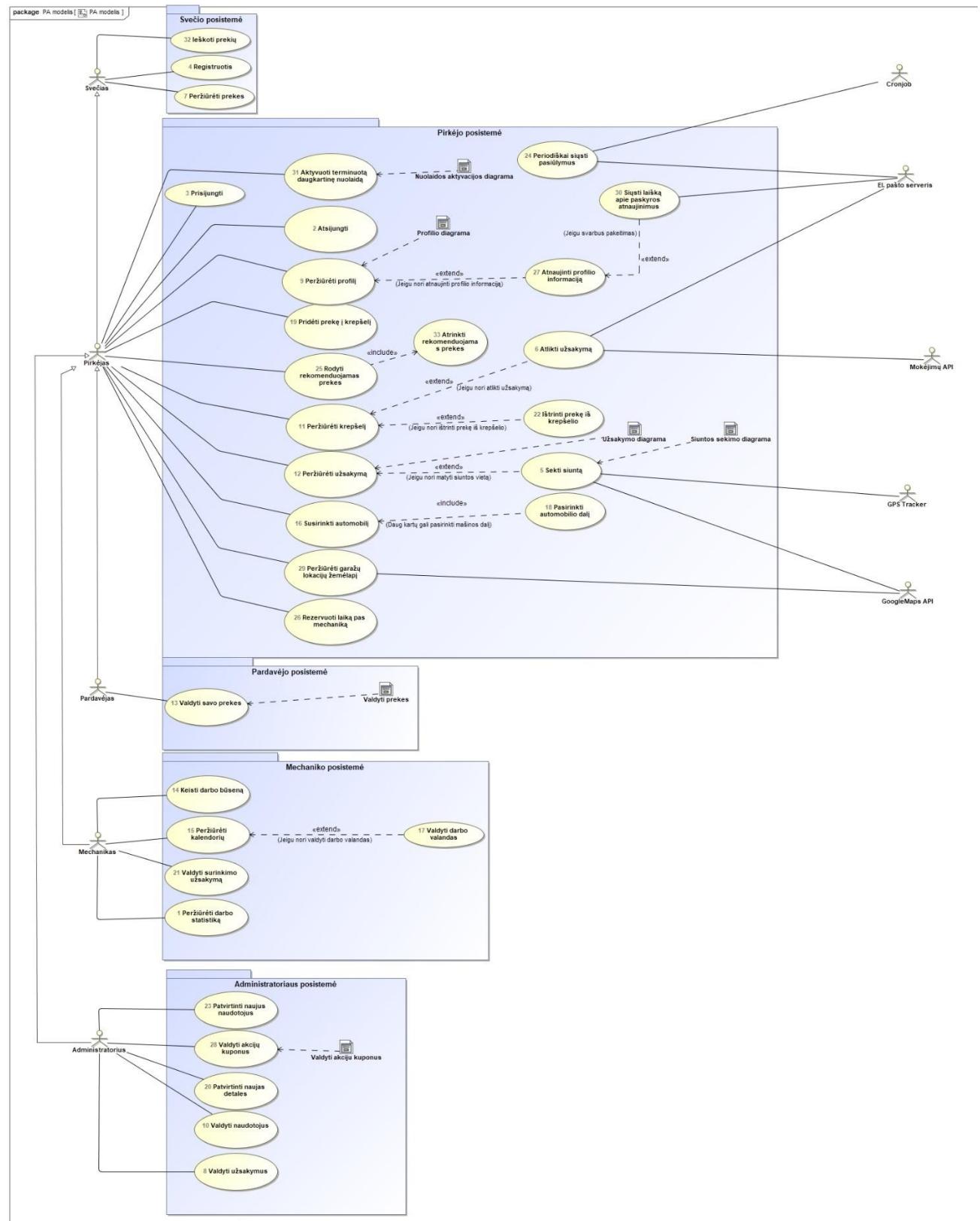
pav. 11 Kupono redagavimo eskizas



pav. 12 Kupono ištrynimo eskizas

3 Sistemos reikalavimų specifikacija

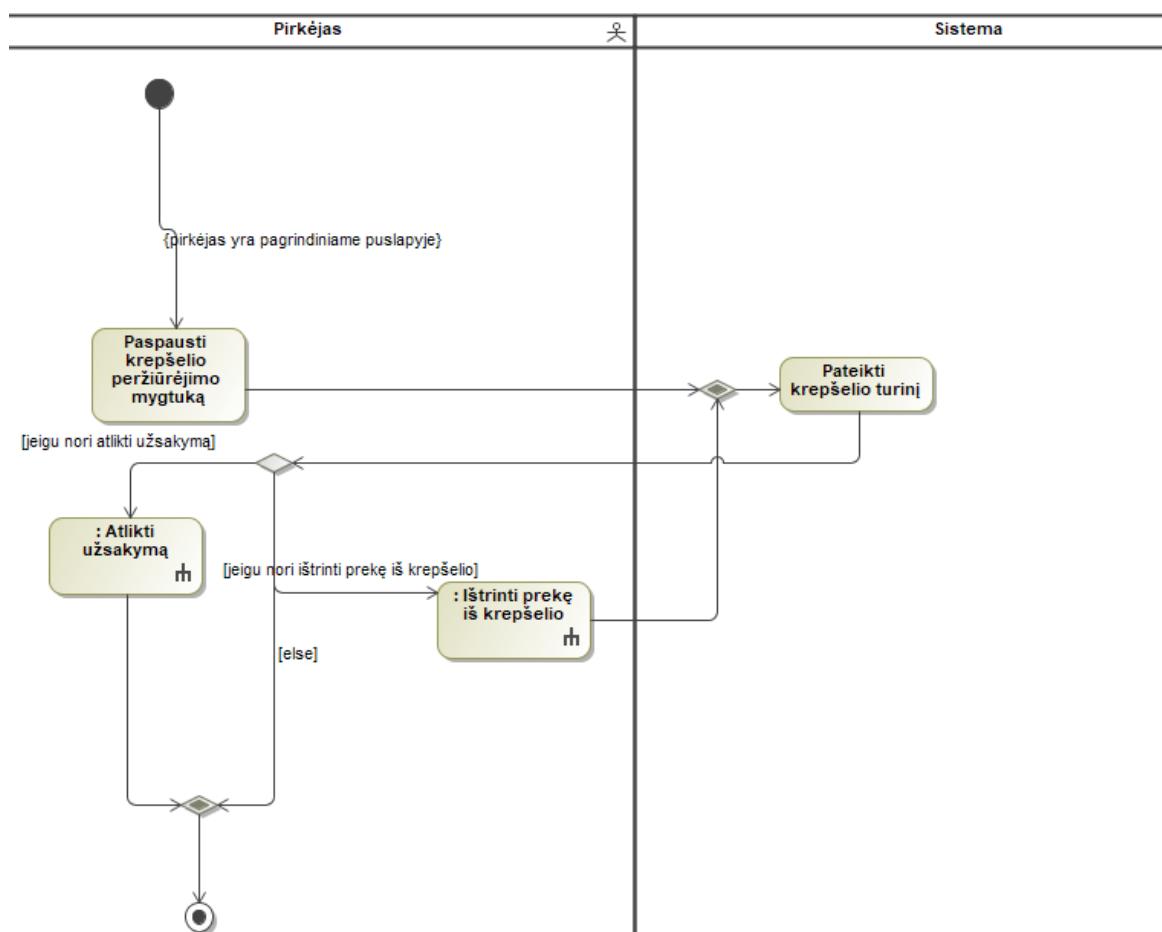
3.1 Panaudojimo atvejų modelis



pav. 13 Panaudojimo atvejų diagrama

Panaudojimo atvejis	11 "Peržiūrėti krepšelj"	Rimvydas Neverauskas
Tikslas.		
Parodyti pirkėjui prekes, kurias idėjo į krepšeli.		
Aprašymas. PA suteikia galimybę pirkėjui peržiūrėti prekių, kurias pasirinko, krepšeli.		
Prieš sąlyga	Pirkėjas yra pagrindiniame puslapyje.	
Aktoriai	Pirkėjas	
Susiję panaudojimo atvejai	Apimami PA Išplečiantys PA Specializuojamai PA	
Po sąlyga	Parodomas pirkėjo prekių krepšelis.	

lentelė 1 "Peržiūrėti krepšelj"

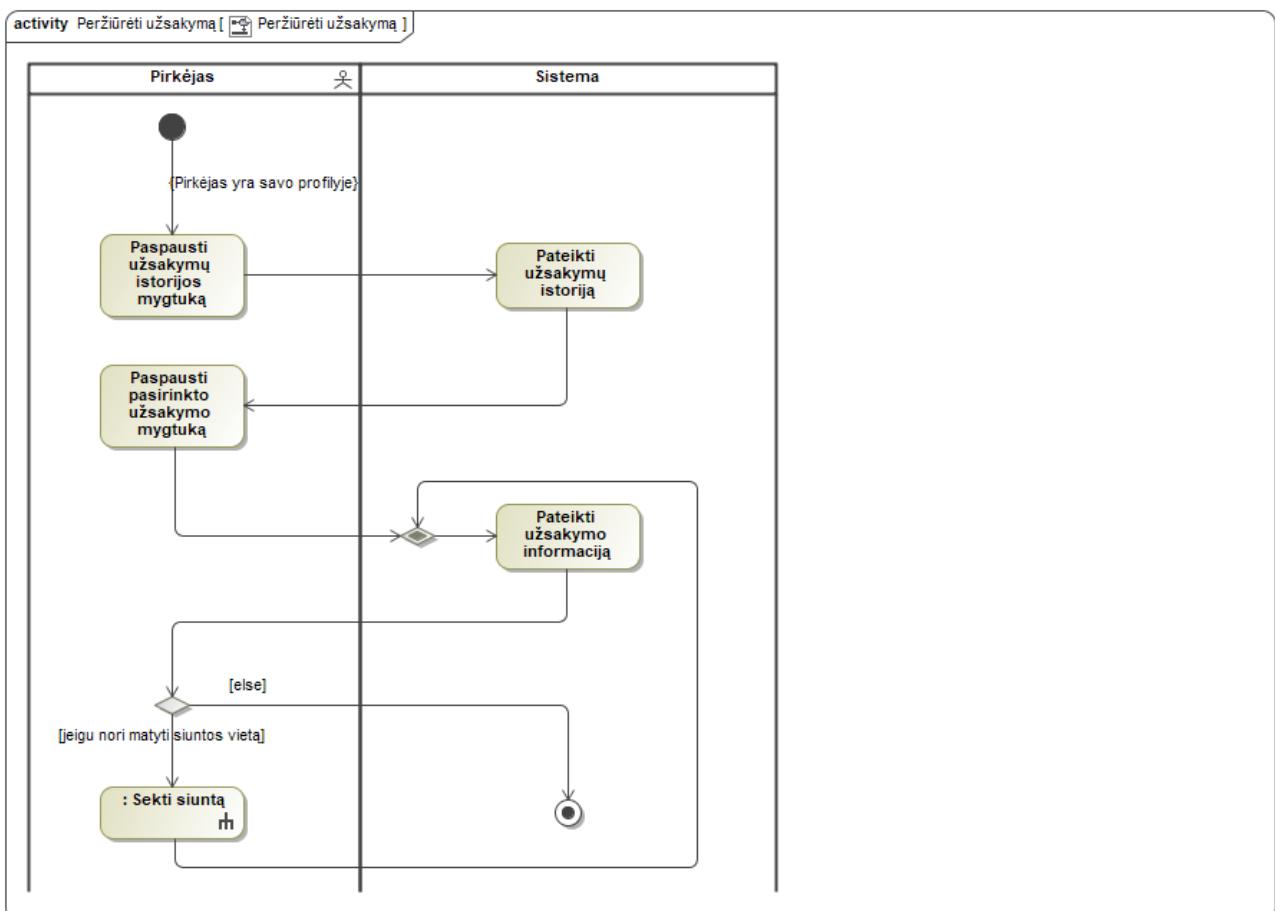


pav. 14 Panaudojimo atvejo "Peržiūrėti krepšelj" scenarijus

Panaudojimo atvejis	12 "Peržiūrėti užsakymą"	Rimvydas Neverauskas
Tikslas.		
Pateikti galimybę peržiūrėti užsakymo istoriją.		
Aprašymas. PA suteikia galimybę peržiūrėti naudotojo padarytą užsakymą.		
Prieš sąlyga	Naudotojas yra užsakymų sąraše.	
Aktoriai	Pirkėjas	
Susiję panaudojimo atvejai	Apimami PA Išplečiantys PA	
	5 "Sekti siunta"	

atvejai	Specializuojami PA	
Po sąlyga		Pateikiama užsakymo informacija.

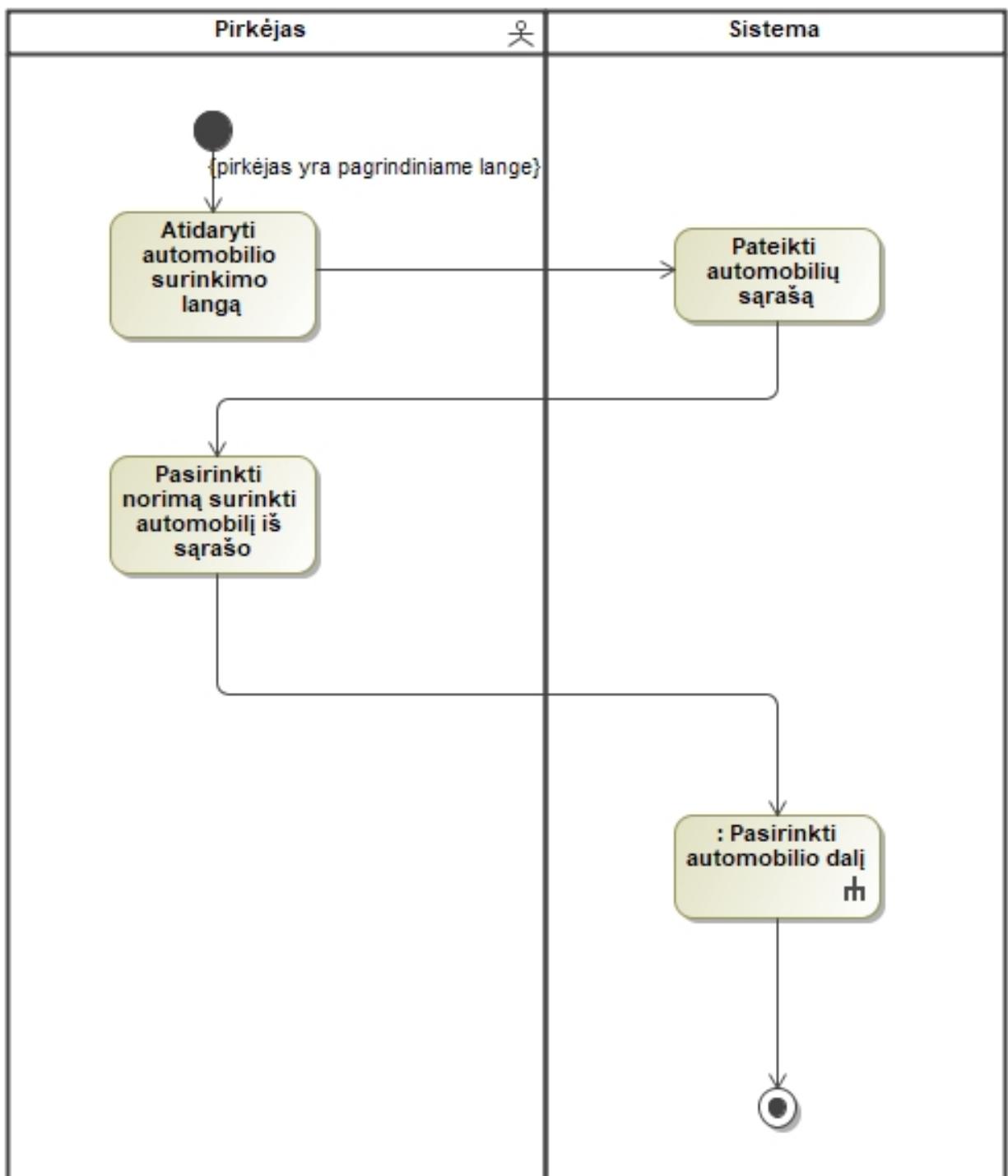
lentelė 2 "Peržiūrēti užsakymai"



pav. 15 Panaujimo atvejo "Peržiūrėti užsakymą" scenarijus

Panaudojimo atvejis	16 "Susirinkti automobilį"	Rimvydas Neverauskas
Tikslas. Suteikti galimybę pirkėjui užsisakyti automobilio surinkimo paslaugą, kurioje visas norimas automobilio dalis pirkėjas pasirinks pats, bei jas sudėti į automobilį.		
Aprašymas. PA suteikia galimybę pirkėjui užsisakyti automobilio surinkimo paslaugą, kurioje visas norimas automobilio dalis pirkėjas pasirinks pats, bei jas sudėti į automobilį.		
Prieš sąlyga		
Aktoriai		Pirkėjas yra pagrindiniame puslapyje.
Susiję panaudojimo atvejai	Apimami PA Išplečiantys PA Specializuojami PA	Pirkėjas 18 "Pasirinkti automobilio dalį"
Po sąlyga		Pirkėjas pasirinks visas norimas automobiliui surinkti dalis.

lentelė 3 "Susirinkti automobilį"

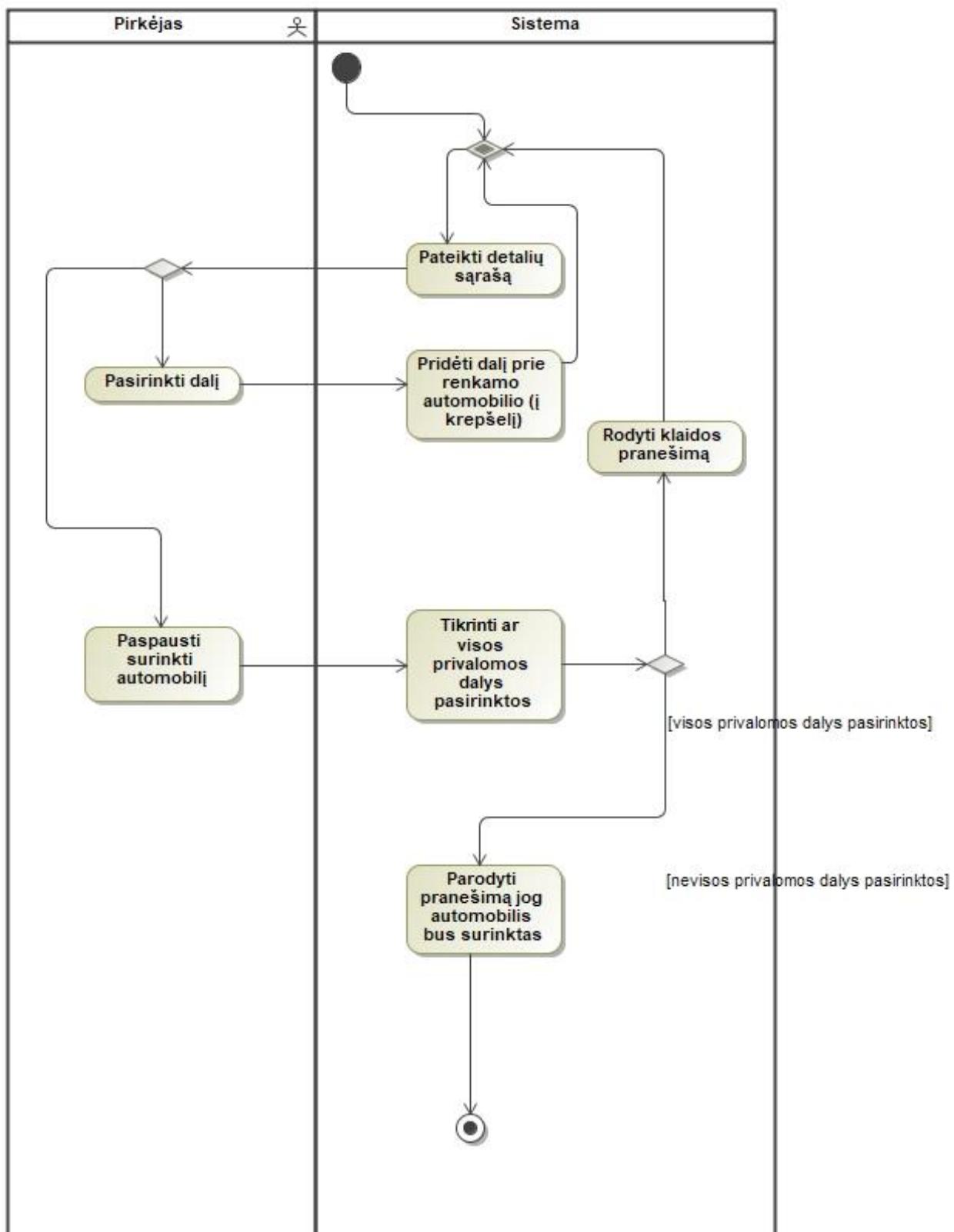


pav. 16 Panaudojimo atvejo "Susirinkti automobilį" scenarijus

Panaudojimo atvejis	18 "Pasirinkti automobilio dalij"	Rimvydas Neverauskas
Tikslas.		
Pirkėjas pasirenka norimą dalij, kuri bus idėta į automobilį.		
Aprašymas. PA suteikia galimybę pirkėjui pridėti norimą dalij prie surenkamo automobilio.		
Prieš sąlyga	Pirkėjas yra automobilio surinkimo puslapyje.	
Aktorai		
Susiję	Apimami PA	

panaudojimo atvejai	Išplečiantys PA	
	Specializuojami PA	
Po sąlyga		Pasirinkta dalis yra pridedama prie pirkėjo surenkamo automobilio.

lentelė 4 "Pasirinkti automobilio dalį"

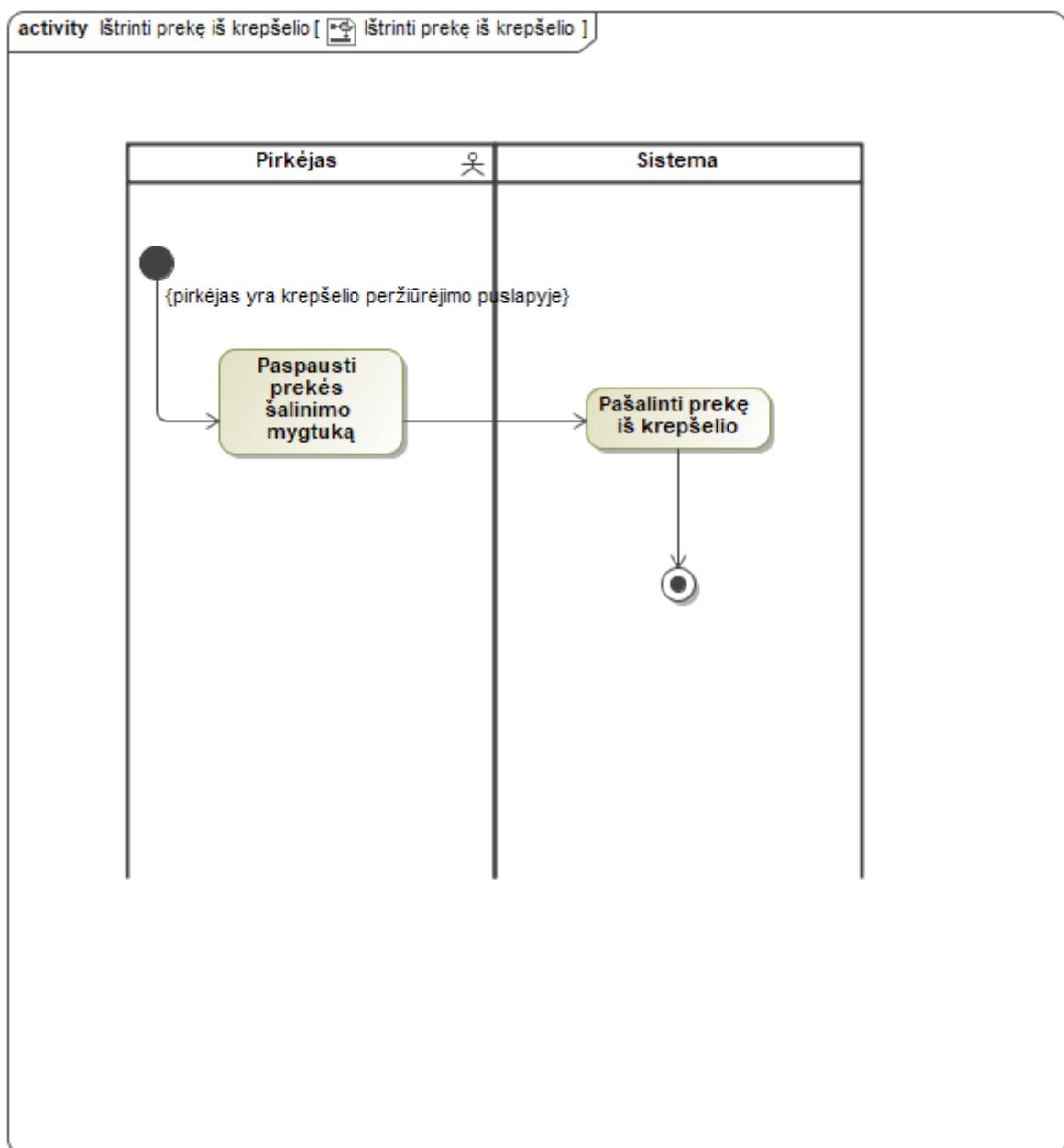


pav. 17 Panaudojimo atvejo "Pasirinkti automobilio dalį" scenarijus

Panaudojimo atvejis	22 "Ištrinti prekę iš krepšelio"	Rimvydas Neverauskas
Tikslas.	Pašalinti prekę iš krepšelio.	

Aprašymas. PA suteikia galimybę pašalinti prekę iš pirkėjo krepšelio.	
Prieš sąlyga	Pirkėjas yra krepšelio peržiūrėjimo puslapyje.
Aktorai	Pirkėjas
Susiję panaudojimo atvejai	Apimami PA
	Išplečiantys PA
	Specializuojamai PA
Po sąlyga	Pirkėjas pašalina prekę iš krepšelio.

Lentelė 5 "Ištrinti prekę iš krepšelio"

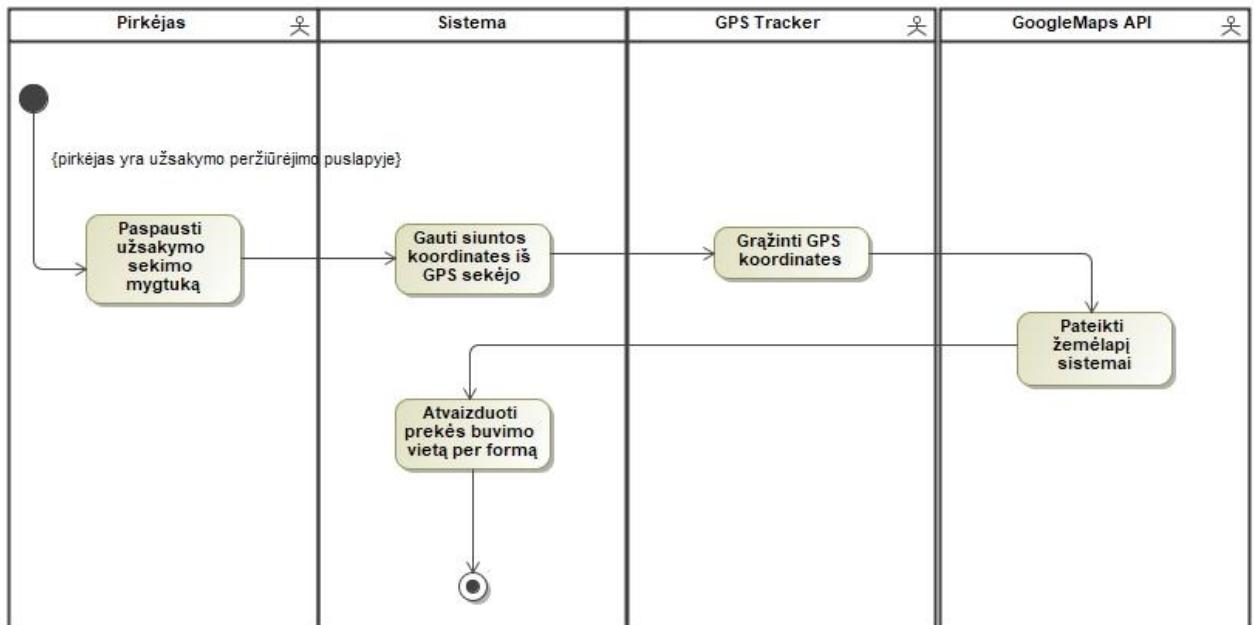


pav. 18 Panaudojimo atvejo "Ištrinti prekę iš krepšelio" scenarijus

Panaudojimo atvejis	5 "Sekti siuntą"	Rimvydas Neverauskas
Tikslas.		
Parodyti pirkėjui siunto buvimo vietą.		
Aprašymas. PA suteikia galimybę parodyti pirkėjui siunto buvimo vietą -emėlapyje.		
Prieš sąlyga	Pirkėjas yra užsakymo peržiūrėjimo puslapyje.	

Aktorai		GoogleMaps API GPS Tracker
Susiję panaudojimo atvejai	Apimami PA	
	Išplečiantys PA	
	Specializuojamai PA	
Po sąlyga		Parodoma siuntos buvimo vieta žemėlapyje.

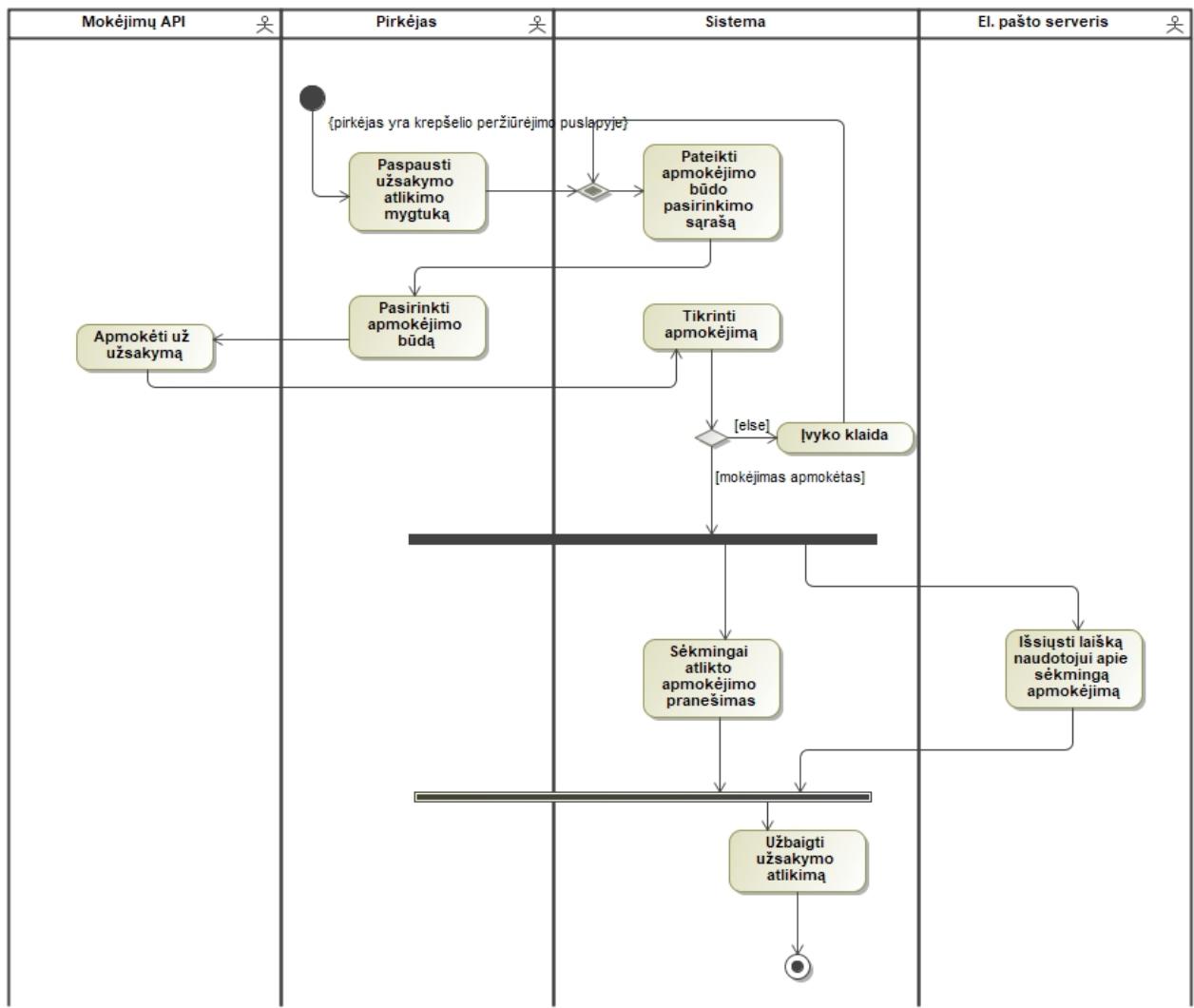
lentelė 6 "Sekti siuntą"



pav. 19 Panaudojimo atvejo "Sekti siuntą" scenarijus

Panaudojimo atvejis	6 "Atlikti užsakymą"	Rimvydas Neverauskas
Tikslas. Apmokėti už krepšelyje esančias prekes.		
Aprašymas. PA suteikia galimybę pirkėjui, atlikti apmokėjimą už krepšelyje esančias prekes.		
Prieš sąlyga Pirkėjas yra krepšelio peržiūrėjimo puslapyje.		
Aktorai	El. pašto serveris Mokėjimų API	
Susiję panaudojimo atvejai	Apimami PA	
	Išplečiantys PA	
	Specializuojamai PA	
Po sąlyga	Pirkėjas atlieka apmokėjimą už krepšelyje esančias prekes.	

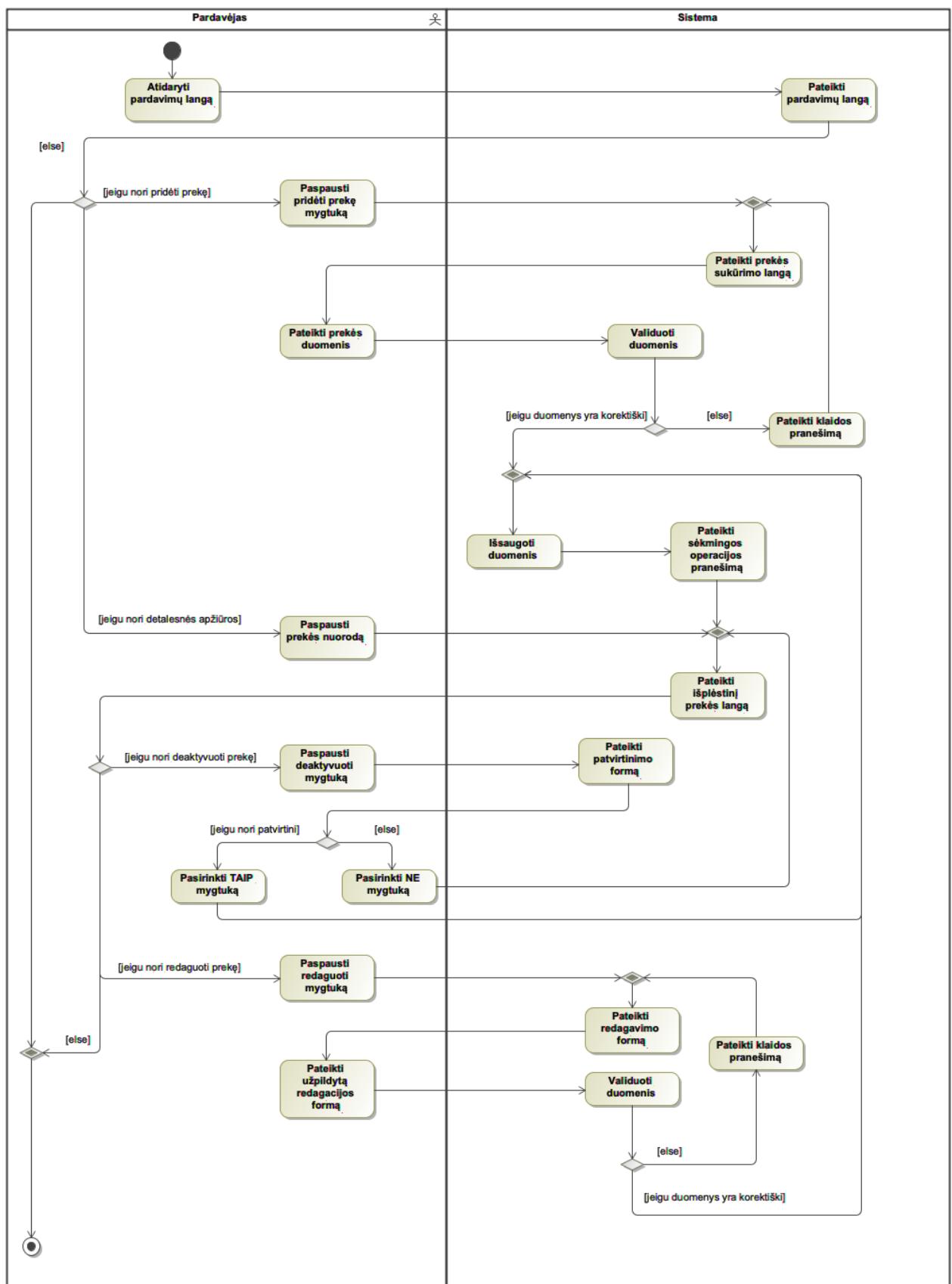
lentelė 7 "Atlikti užsakymą"



pav. 20 Panaudojimo atvejo "Atlikti užsakymą" scenarijus

Panaudojimo atvejis	13 "Valdyti savo prekes"	Henrikas Juzuitis
Tikslas.		
Suteikti pardavėjui galimybę peržiūrėti savo prekes, pridėti naujas prekes, redaguoti savo prekes ir deaktyvuoti prekės pardavimą		
Aprašymas. Pardavėjas atsidarės pardavimų langą gali detaliau peržiūrėti savo prekes, pridėti naujas prekes, deaktyvuoti savo prekės pardavimą ir redaguoti parduodamas prekes. Naujai pridėtos prekės turės sulaukti administratoriaus patvirtinimo.		
Prieš sąlyga		Pardavėjas yra prisijungęs, patvirtintas ir pagrindiniame puslapyje
Aktorai		Pardavėjas
Susiję panaudojimo atvejai	Apimami PA	
	Išplečiantys PA	
	Specializuojami PA	
Po sąlyga		Patvirtintos prekės bus patalpinamos į parduodamų prekių sąrašą. Deaktyvuotos prekės yra išimamos iš parduodamų prekių sąrašo. Redaguotos prekės yra atnaujintos.

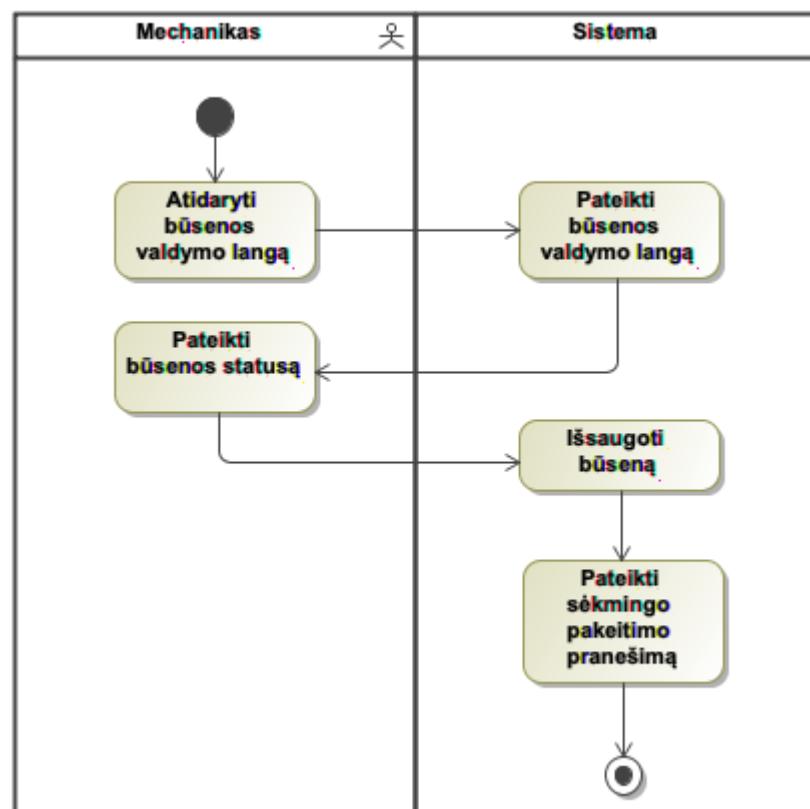
Lentelė 8 Valdyti savo prekes



pav. 21 Panaudojimo atvejo "Valdyti savo prekes" scenarijus

Panaudojimo atvejis	14 "Keisti darbo būseną"	Henrikas Juzuitis
Tikslas.		
Suteikti mechanikui galimybę pakeisti savo darbo būseną		
Aprašymas. Mechanikas gali pakeisti savo darbo būseną į kitą iš sąrašo.		
Prieš sąlyga	Mechanikas yra prisijungęs, patvirtintas ir darbo puslapyje	
Aktorai	Mechanikas	
Susiję panaudojimo atvejai	Apimami PA Išplečiantys PA Specializuojamieji PA	
Po sąlyga	Mechaniko darbo būsena yra atnaujinta	

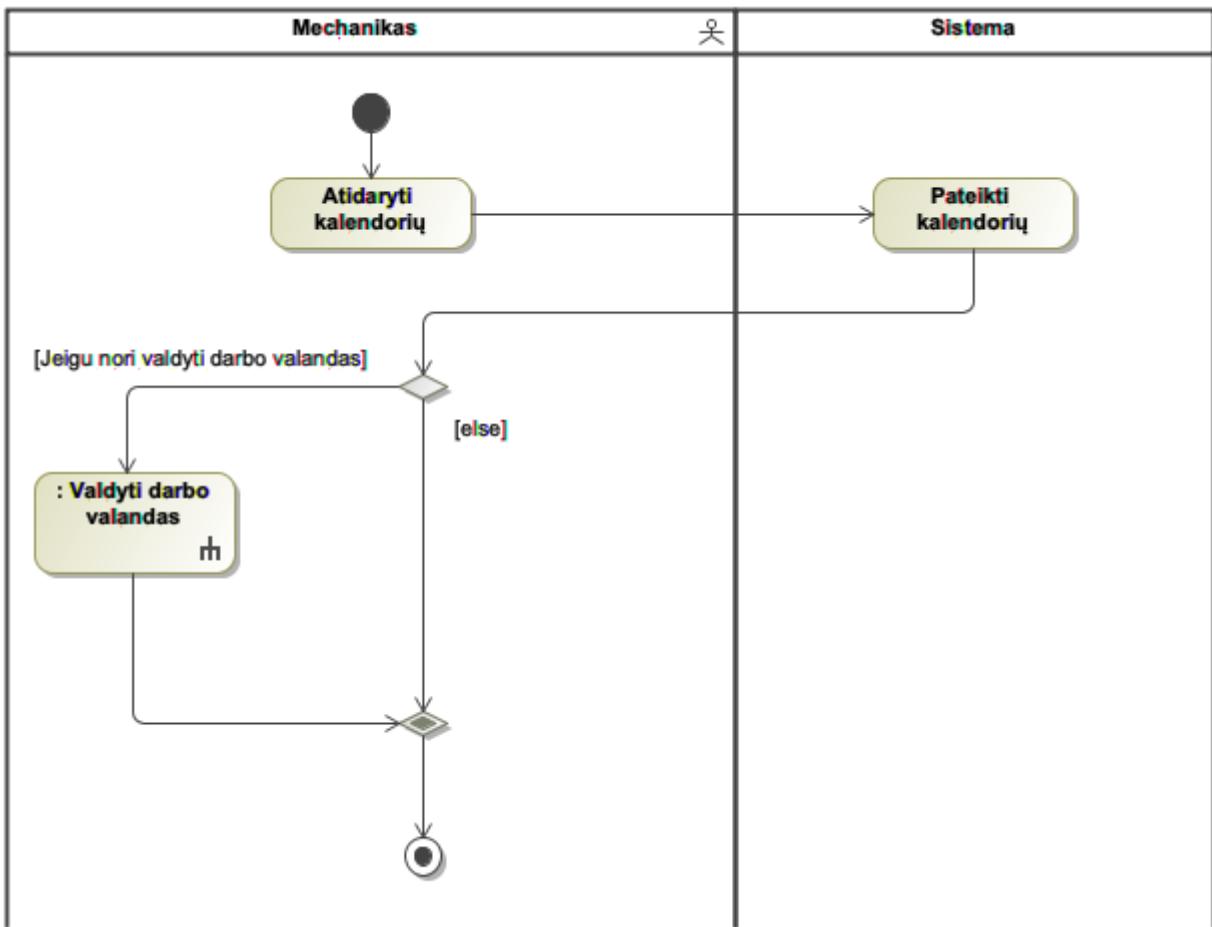
Lentelė 9 Keisti darbo būseną



pav. 22 Panaudojimo atvejo "Valdyti darbo būseną" scenarijus

Panaudojimo atvejis	15 "Peržiūrėti kalendorių"	Henrikas Juzuitis
Tikslas.		
Suteikti mechanikui galimybę peržiūrėti savo darbo kalendorių		
Aprašymas. Mechanikas gali peržiūrėti savo darbo kalendorių		
Prieš sąlyga	Mechanikas yra prisijungęs, patvirtintas ir darbo puslapyje	
Aktorai	Mechanikas	
Susiję panaudojimo atvejai	Apimami PA Išplečiantys PA Specializuojamieji PA	
Po sąlyga	Mechaniko kalendorius peržiūrėtas	

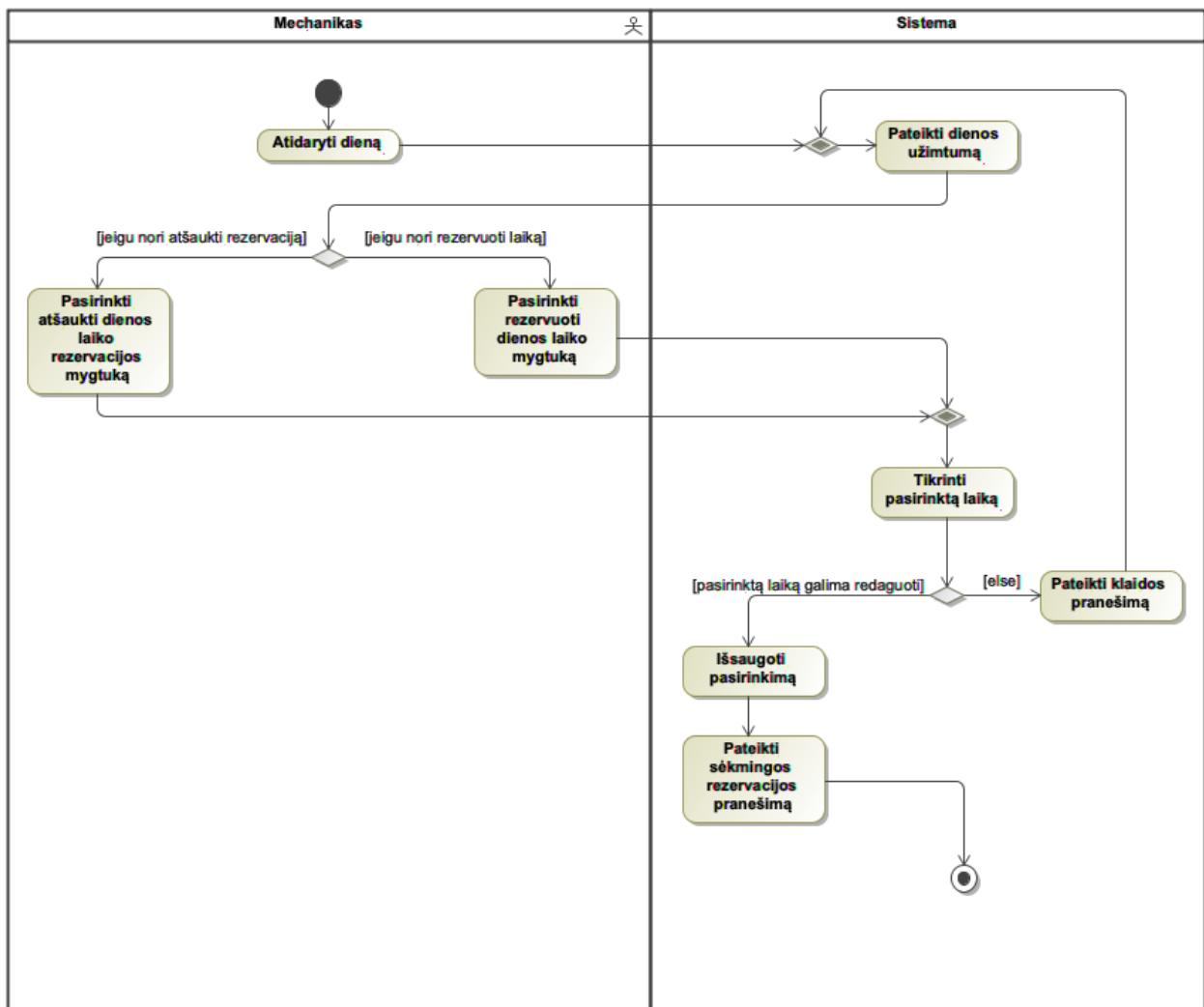
Lentelė 10 Peržiūrėti kalendorių



pav. 23 Panaudojimo atvejo "Peržiūrėti kalendorių" scenarijus

Panaudojimo atvejis	17 "Valdyti darbo valandas"	Henrikas Juzuitis
Tikslas.	Suteikti galimybę mechanikui valdyti savo darbo valandas	
Aprašymas.	Mechanikas gali valdyti savo darbo valandas. Redaguotos darbo valandos bus pridėtos arba pašalintos iš laisvų darbo valandų sąrašo.	
Prieš sąlyga	Mechanikas yra prisijungęs, patvirtintas ir atsidaręs savo kalendorių	
Aktorai	Mechanikas	
Susiję panaudojimo atvejai	Apimami PA Išplečiantys PA Specializuojamai PA	
Po sąlyga	Mechaniko darbo kalendorius yra atnaujintas	

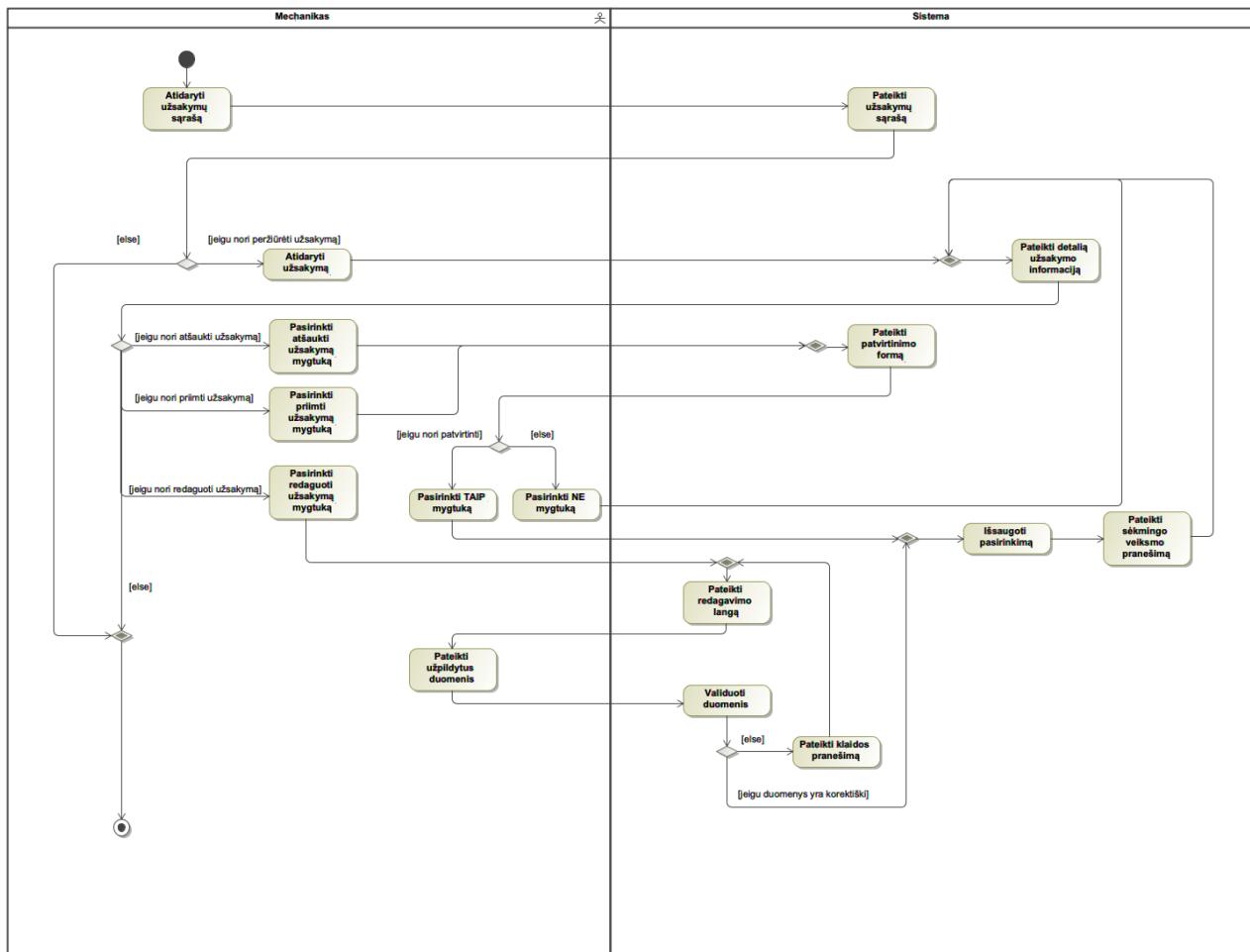
lentelė 11 Valdyti darbo valandas



pav. 24 Panaudojimo atvejo "Valdyti darbo valandas" scenarijus

Panaudojimo atvejis	21 "Valdyti surinkimo užsakymą"	Henrikas Juzuitis
Tikslas. Suteikti mechanikui galimybę peržiūrėti, redaguoti, priimti ir atšaukti užsakymą		
Apaščias. Mechanikas gali detaliau peržiūrėti užsakymus. Gali užsakymus redaguoti, atšaukti ir priimti.		
Prieš sąlyga Mechanikas yra prisijungęs, patvirtintas ir darbo puslapyje		
Aktorai	Mechanikas	
Susiję panaudojimo atvejai	Apimami PA Išplečiantys PA Specializuojami PA	
Po sąlyga	Priimti, atšaukti ir redaguoti užsakymai yra atnaujinti	

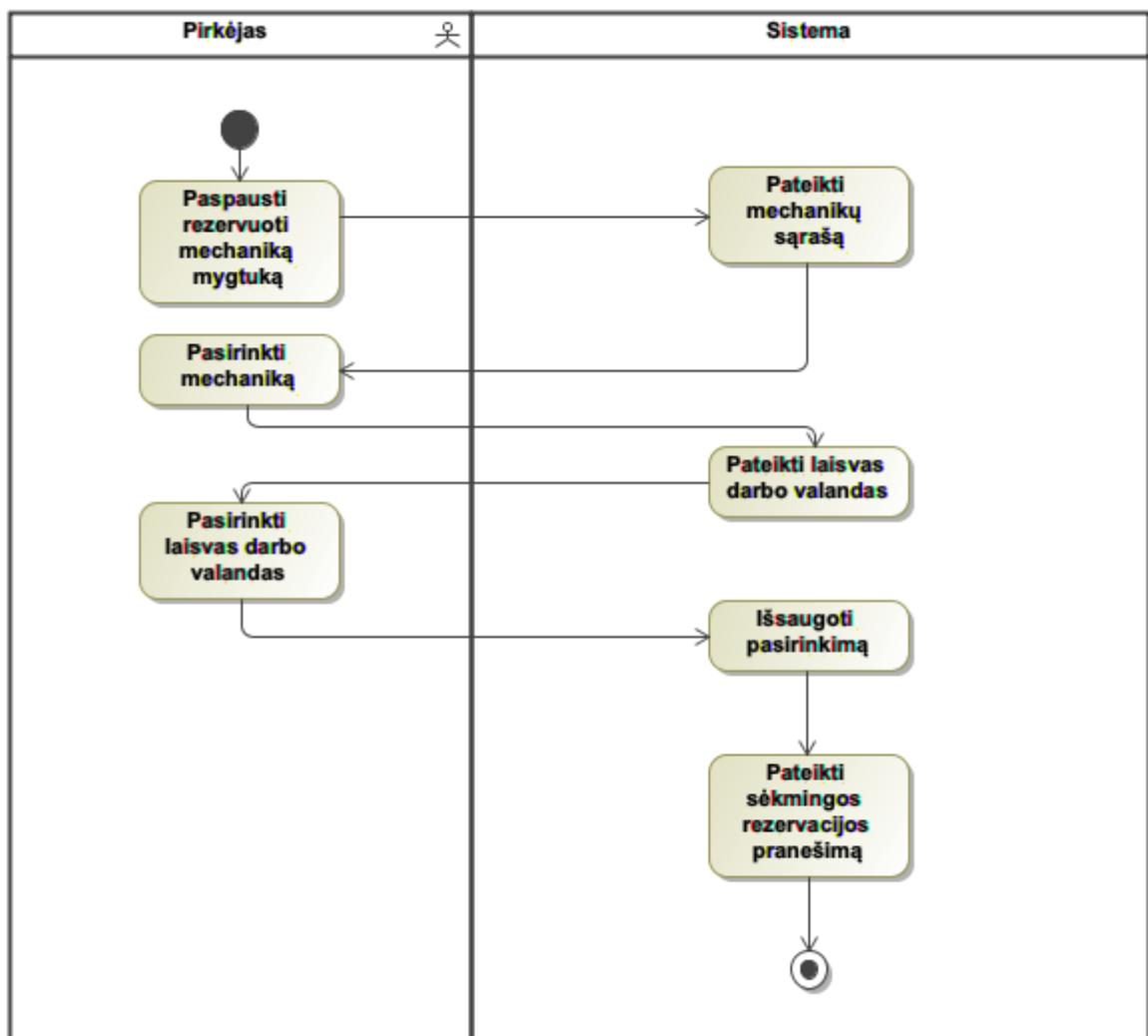
lentelė 12 Valdyti surinkimo užsakymą



pav. 25 Panaudojimo atvejo "Valdyti surinkimo užsakymą" scenarijus

Panaudojimo atvejis	26 "Rezervuoti laiką pas mechaniką"	Henrikas Juzuitis
Tikslas.		
Suteikti galimybę pirkėjui užsirezervuoti laiką pas mechaniką		
Aprašymas. Pirkėjas gali iš sąrašo pasirinkti mechaniką. Pasirinkęs mechaniką, galės rezervuoti laisvas darbo valandas iš sąrašo. Tuomet parinktos darbo valandas mechaniko, bus pažymėtos kaip užimtos.		
Prieš salyga	Pirkėjas yra prisijungęs ir pagrindiniame puslapyje	
Aktoriai	Pirkėjas	
Susiję panaudojimo atvejai	Apimami PA	
	Išplečiantys PA	
	Specializuojamai PA	
Po salyga	Mechaniko darbo valandas yra pažymėtos kaip užimtos	

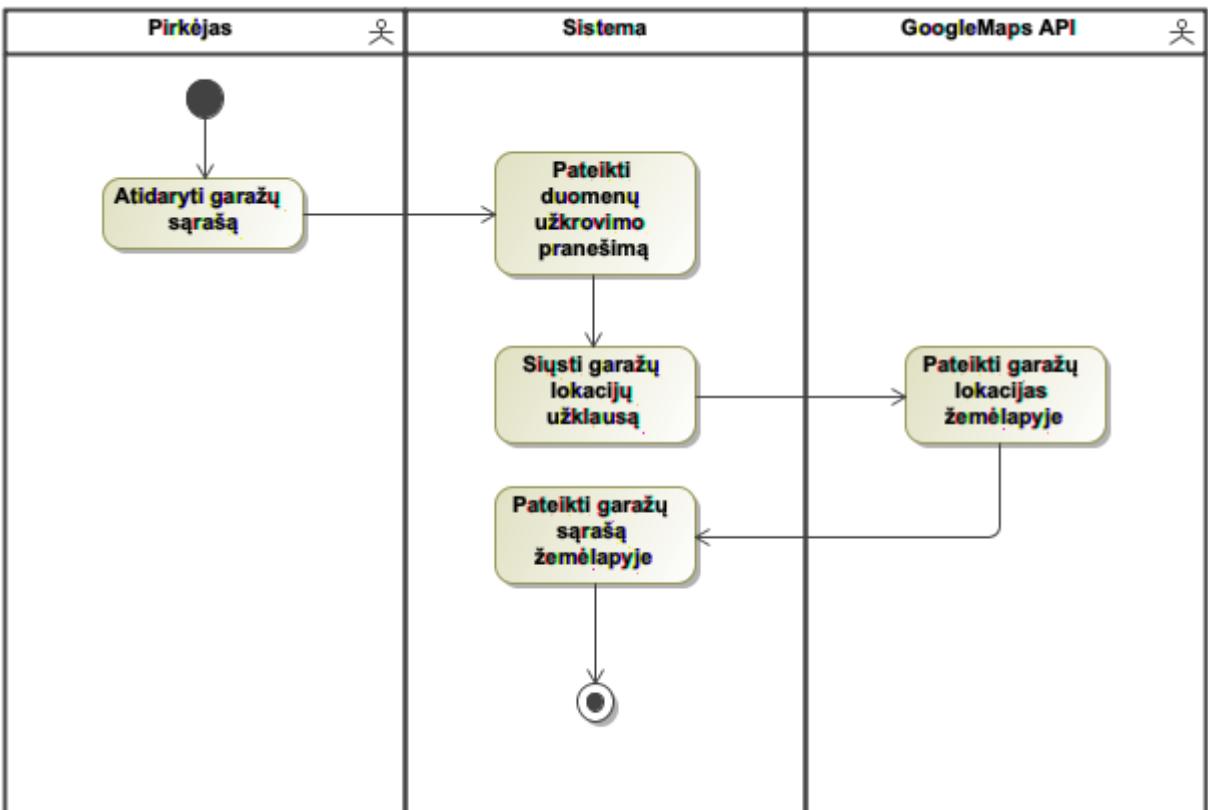
lentelė 13 Rezervuoti laiką pas mechaniką



pav. 26 Panaudojimo atvejo "Rezervuoti laiką pas mechaniką" scenarijus

Panaudojimo atvejis	29 "Peržiūrėti garažų lokacijų žemėlapį"	Henrikas Juzuitis
Tikslas.		
Leisti pirkėjui peržiūrėti garažų lokacijas žemėlapyje		
Aprašymas. Pirkėjas gali pamatyti visus garažus žemėlapyje		
Prieš sąlyga	Pirkėjas yra prisijungęs ir pagrindiniame puslapyje	
Aktorai	GoogleMaps API Pirkėjas	
Susiję panaudojimo atvejai	Apimami PA	
	Išplečiantys PA	
	Specializuojami PA	
Po sąlyga	Garažų lokacijos žemėlapyje yra peržiūrėtos	

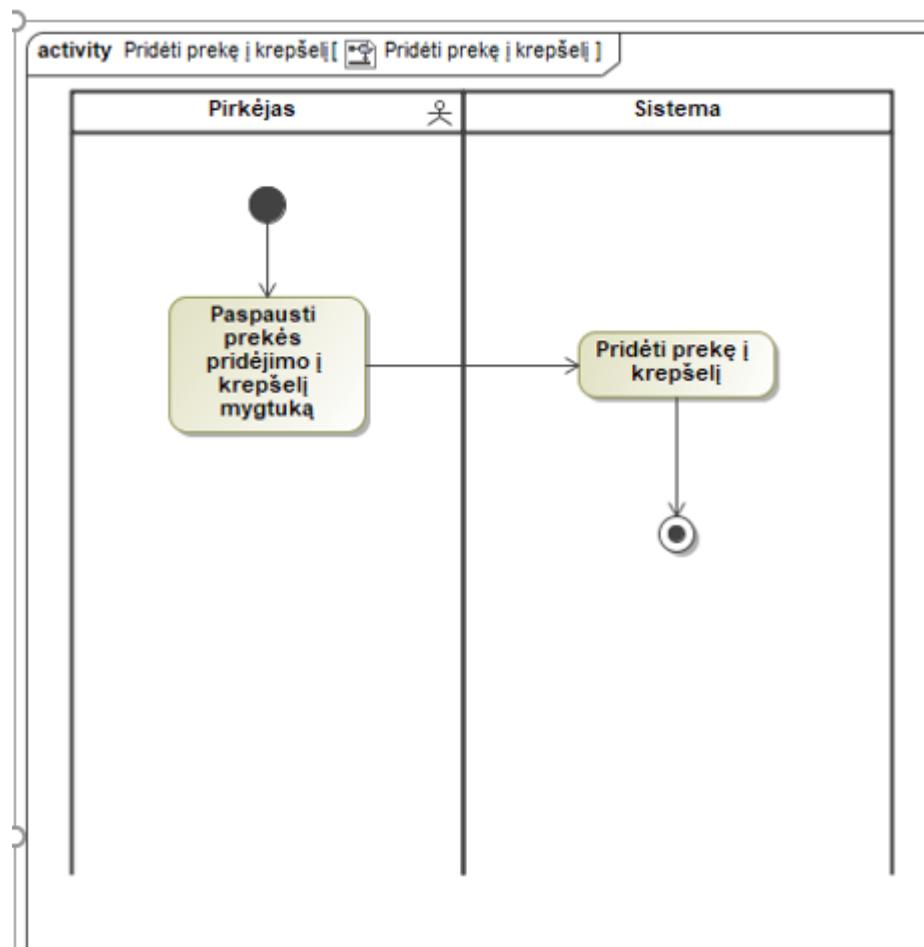
Lentelė 14 Peržiūrėti garažų lokacijų žemėlapį



pav. 27 Panaudojimo atvejo "Peržiūrėti garažų lokacijų žemėlapį" scenarijus

Panaudojimo atvejis	19 "Pridėti prekę į krepšelį"	Valentinas Kasteckis
Tikslas.		
Suteikti galimybę naudotojui prisidėti norimas prekes prie krepšelio		
Aprašymas. PA atvėjai suteikia galimybę naudotojui prisidėti prekes prie krepšelio		
Prieš sąlyga		Naudotojas turi būti prisijungęs
Aktorai		Pirkėjas
Susiję panaudojimo atvejai	Apimami PA	
	Išplečiantys PA	
	Specializuojami PA	
Po sąlyga		Prekė pridedama prie krepšelio

lentelė 15 Pridėti prekę į krepšelį

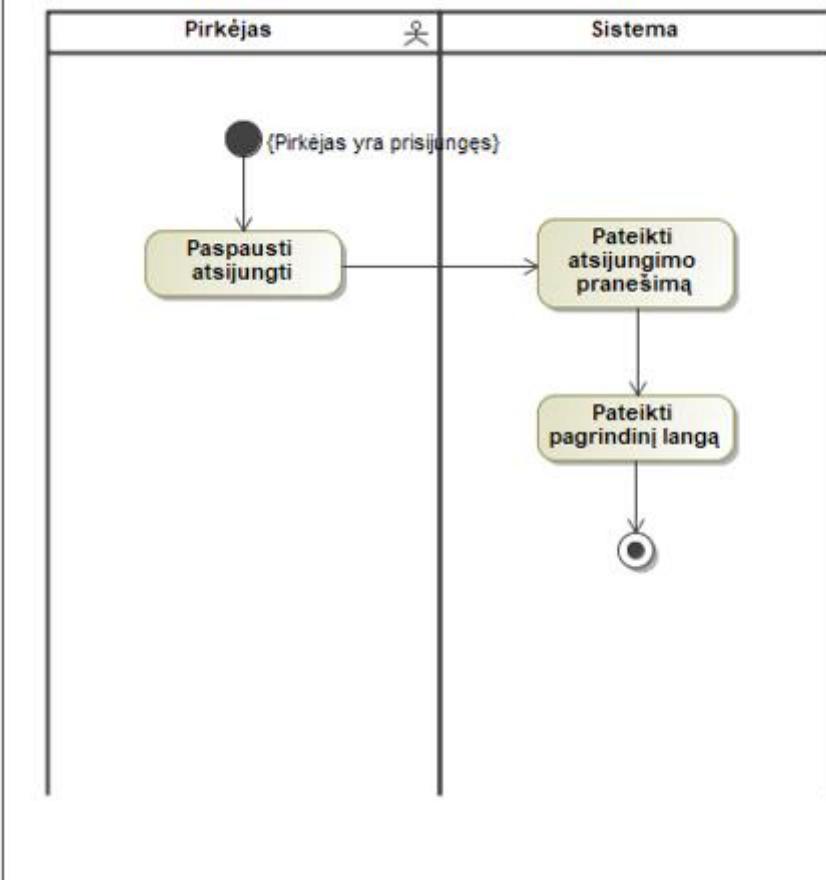


pav. 28 Panaudojimo atvejo "Pridėti prekę į krepšelį" scenarijus

Panaudojimo atvejis	2 "Atsijungti"	Valentinas Kasteckis
Tikslas.		
Suteikti galimybę atsijungti nuo sistemos		
Aprašymas. PA suteikia galimybę naudotojui atsijungti nuo sistemos		
Prieš sąlyga		Naudotojas turi būti prisijungęs.
Aktorai		Pirkėjas
Susiję panaudojimo atvejai	Apimami PA	
	Išplečiantys PA	
	Specializuojami PA	
Po sąlyga		Naudotojas yra atjungiamas nuo sistemos, jis tampa Svečiu.

lentelė 16 Atsijungti

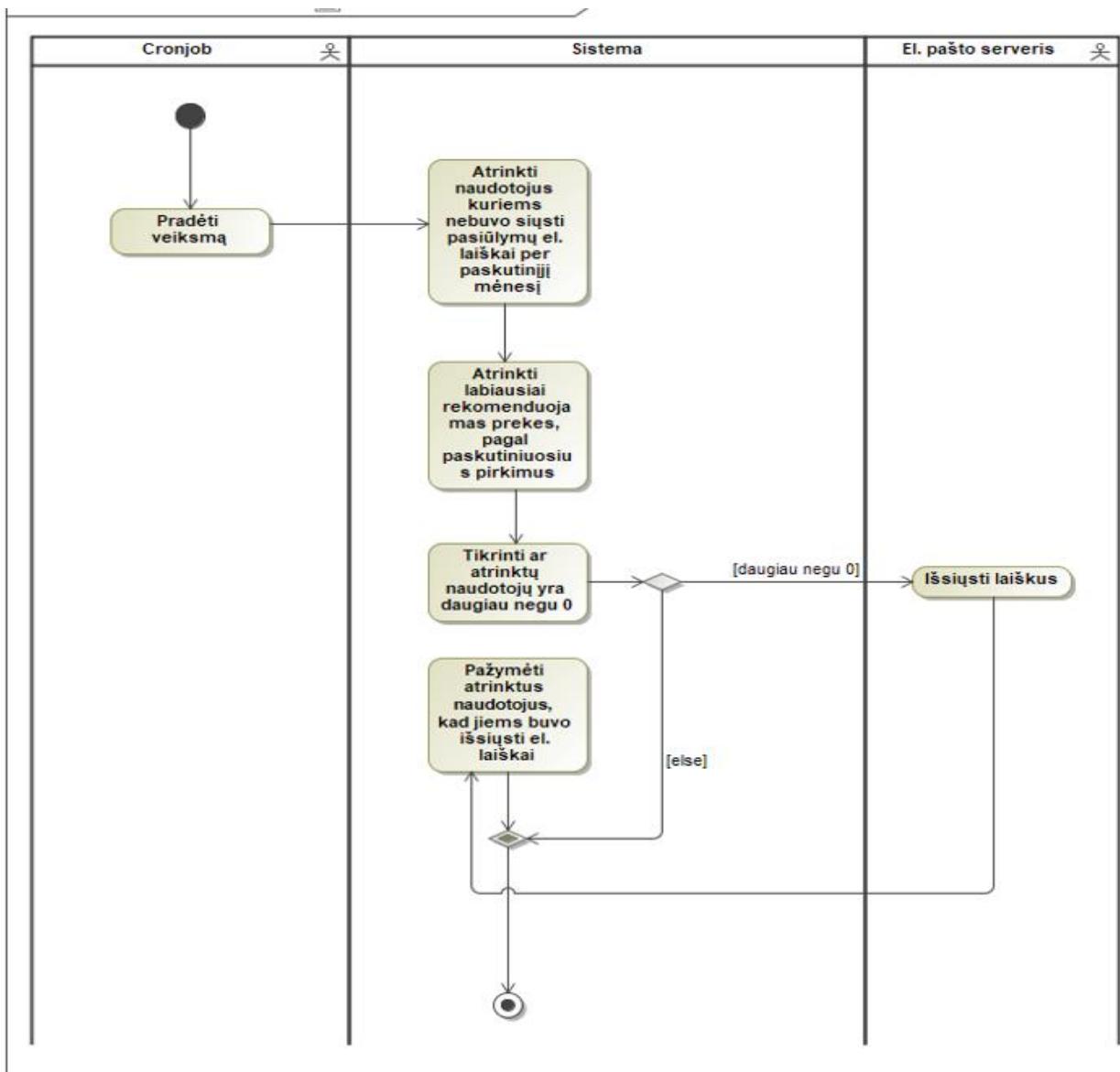
activity Atsijungti [ Atsijungti]



pav. 29 Panaudojimo atvejo "Atsijungti" scenarijus

Panaudojimo atvejis	24 "Periodiškai siųsti pasiūlymus"	Valentinas Kasteckis
Tikslas.		
Suteikti galimybę sistemai periodiškai siųsti marketinginius pasiūlymus naudotojams į el. paštą		
Aprašymas. PA suteikia galimybę siųsti periodiškai el. laiškus naudotojams į el. paštą		
Prieš sąlyga		Laikmatis praneša apie darbo pradžią
Aktorai		Cronjob El. pašto serveris
Susiję panaudojimo atvejai	Apimami PA	
	Išplečiantys PA	
	Specializuojami PA	
Po sąlyga		Išsiunčiami pasiūlymai naudotojams

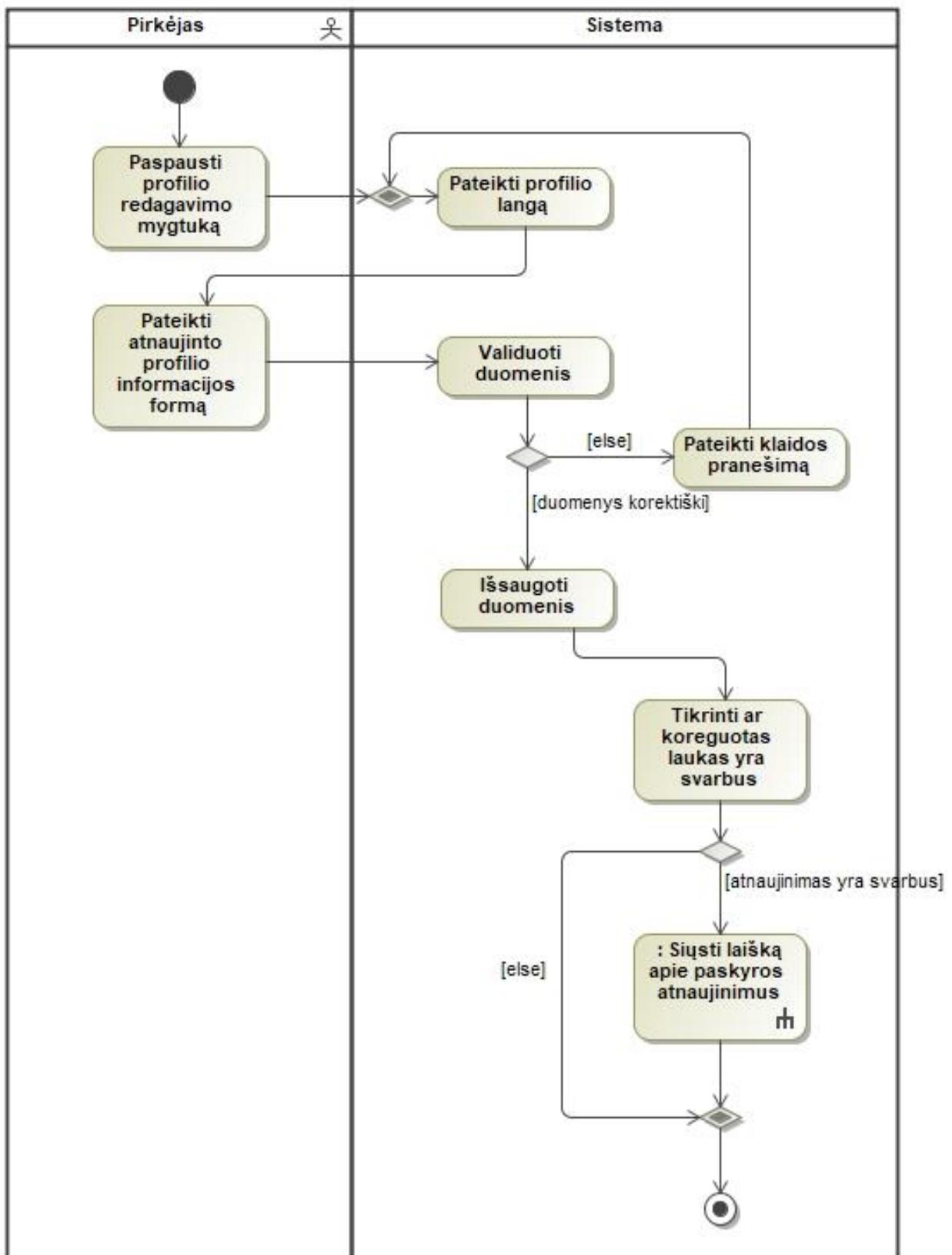
lentelė 17 Periodiškai siųsti pasiūlymus



pav. 30 Panaudojimo atvejo "Periodiškai siųsti pasiūlymus" scenarijus

Panaudojimo atvejis	27 "Atnaujinti profilio informaciją"	Valentinas Kasteckis
Tikslas.		
Suteikti galimybę naudotojui atnaujinti savo profilį		
Aprašymas. PA suteikia galimybę naudotojui atnaujinti savo profilį		
Prieš sąlyga	Naudotojas turi būti prisijungęs prie sistemos	
Aktorai		
Susiję panaudojimo atvejai	Apimami PA	
	Išplečiantys PA	30 "Siųsti laišką apie paskyros atnaujinimus"
	Specializuojamai PA	
Po sąlyga	Naudotojas mato savo atnaujintą profilį	

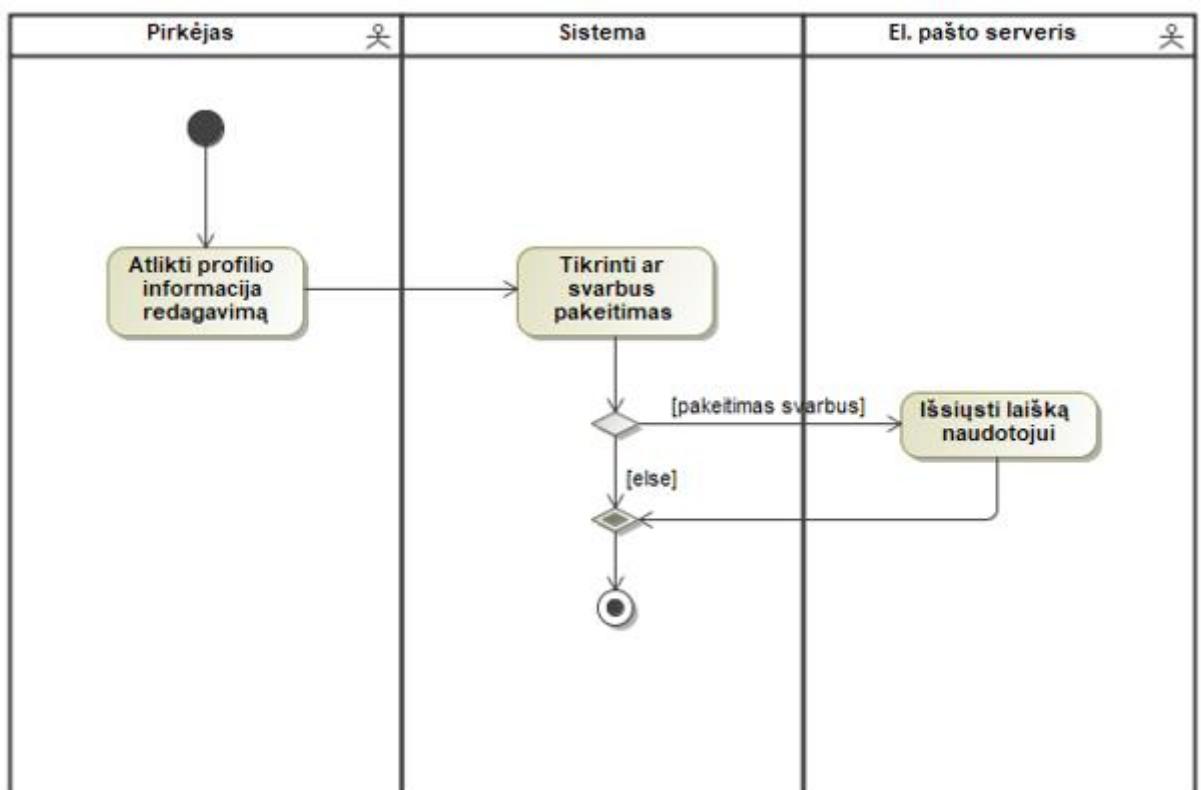
lentelė 18 Atnaujinti profilio informaciją



pav. 31 Panaudojimo atvejo "Atnaujinti profilio informaciją" scenarijus

Panaudojimo atvejis	30 "Siųsti laišką apie paskyros atnaujinimus"	Valentinas Kasteckis
Tikslas.		
Suteikti galimybę naudotojui gauti el. laiškus apie svarbius profilio informacijos pakeitimus		
Aprašymas. PA atvėjis suteikia galimybę naudotojui gauti svarbius laiškus apie jo profilio informacijos svarbius pakeitimus.		
Prieš sąlyga		Naudotojas atnaujino svarbų informacijos lauką
Aktorai		El. pašto serveris
Susiję panaudojimo atvejai	Apimami PA	
	Išplečiantys PA	
	Specializuojami PA	
Po sąlyga		Išsiųstas laiškas naudotojui į el. paštą

Lentelė 19 Siųsti laišką apie paskyros atnaujinimus



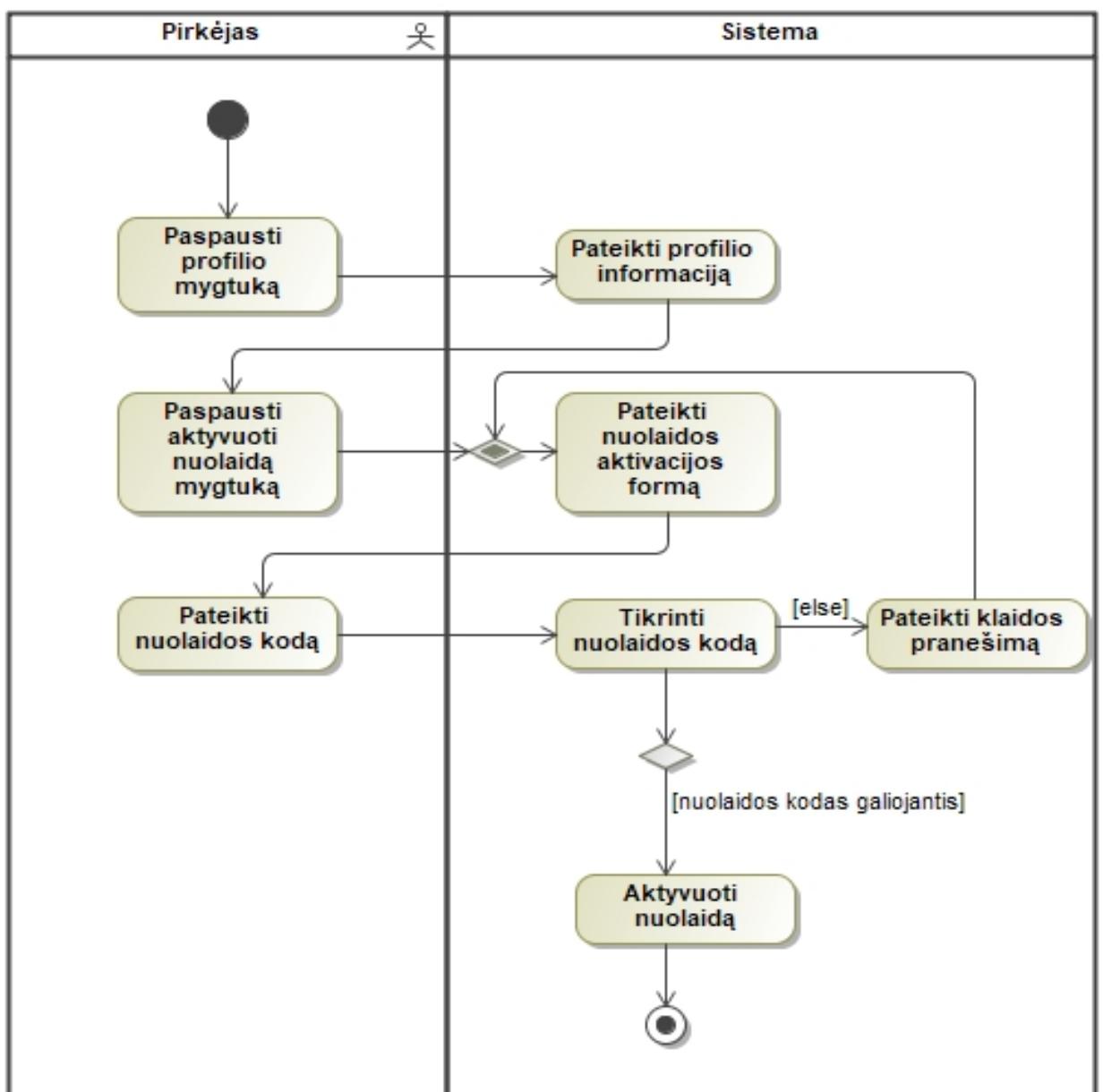
pav. 32 Panaudojimo atvejo "Siųsti laišką apie paskyros atnaujinimus" scenarijus

Panaudojimo atvejis	31 "Aktyvuoti terminuotą daugkartinę nuolaidą"	Valentinas Kasteckis
Tikslas.		
Suteikti galimybę aktyvuotis daugkartinę nuolaidą naudotojui		

Aprašymas. PA suteikia galimybę naudotojui įvesti daugkartinę nuolaidą (jeigu jis žino kupono kodą) bei naudotis ilgalaike nuolaida prekėms.

Prieš sąlyga	Naudotojas turi būti prisijungęs.
Aktoriai	Pirkėjas
Susiję panaudojimo atvejai	Apimami PA
	Išplečiantys PA
	Specializuojamai PA
Po sąlyga	Aktyvuojama daugkartinė nuolaidą naudotojui.

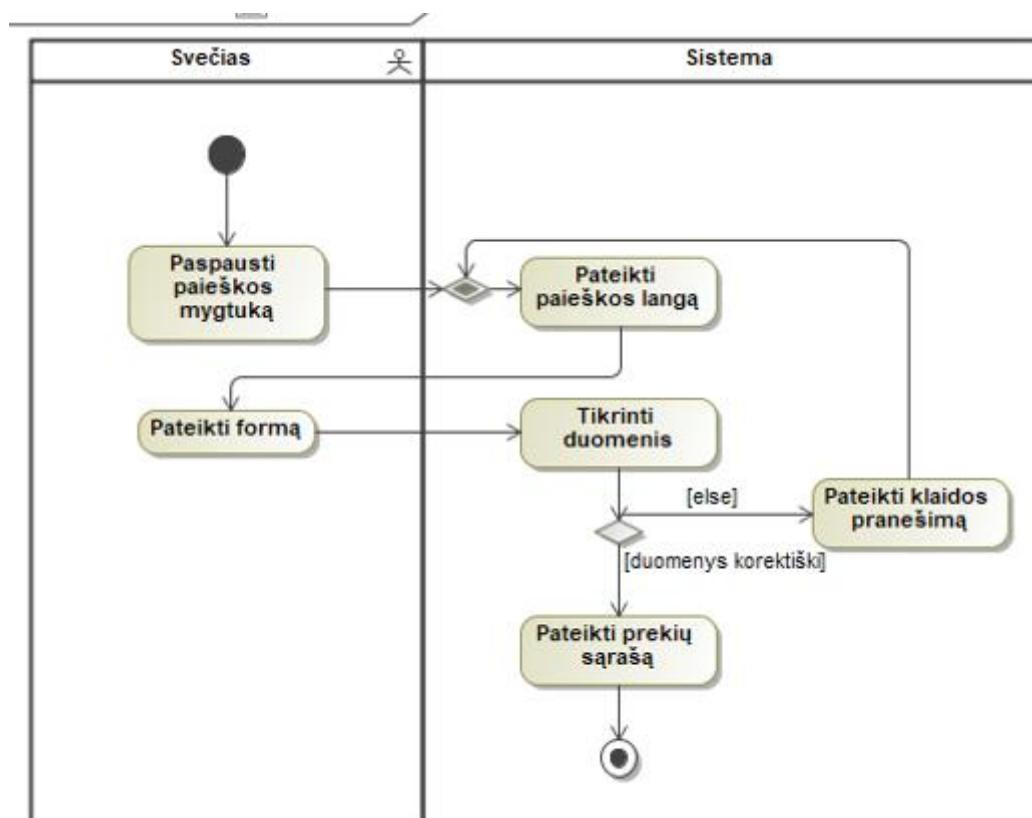
Lentelė 20 Aktyvuoti terminuotą daugkartinę nuolaidą



pav. 33 Panaudojimo atvejo "Aktyvuoti terminuotą daugkartinę nuolaidą" scenarijus

Panaudojimo atvejis	32 "leškoti prekių"	Valentinas Kasteckis
Tikslas.		
Suteikti galimybę naudotojui ieškoti prekių		
Aprašymas. PA atvėjis suteikia galimybę naudotojui ieškoti prekių		
Prieš sąlyga		Naudotojas įsijungęs sistemą
Aktorai		Svečias
Susiję panaudojimo atvejai	Apimami PA	
	Išplečiantys PA	
	Specializuojami PA	
Po sąlyga		Pateikti atfiltruotą prekių sąrašą naudotojui

lentelė 21 leškoti prekių

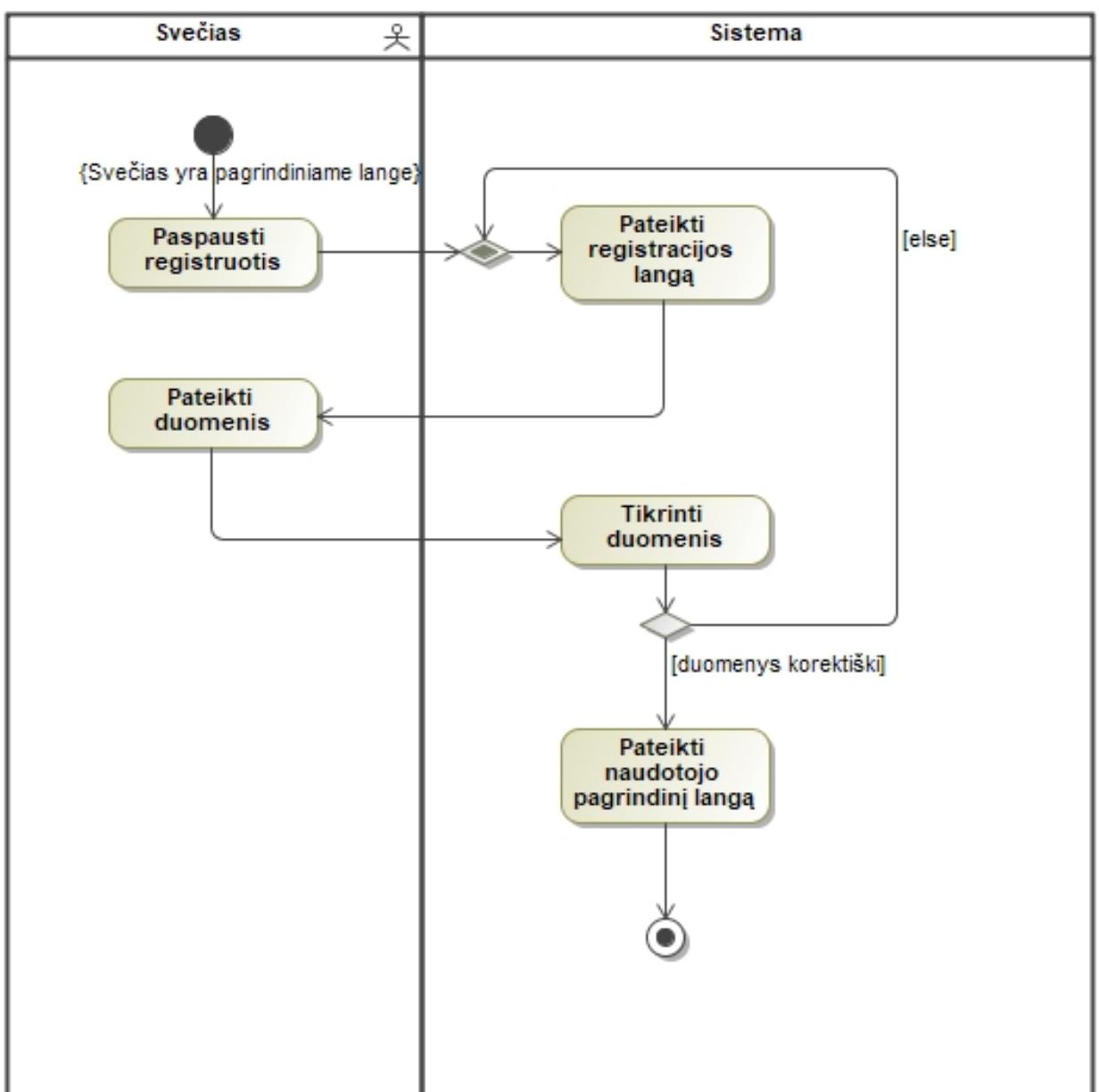


pav. 34 Panaudojimo atvejo "leškoti prekių" scenarijus

Panaudojimo atvejis	4 "Registruotis"	Valentinas Kasteckis
Tikslas.		
Suteikti galimybę svečiui užsiregistrnuoti prie sistemos		
Aprašymas. PA atvėjis suteikia galimybę svečiui prisiregistrnuoti prie sistemos		
Prieš sąlyga		Būti neprisijungus prie sistemos

Aktorai		Svečias
Susiję panaudojimo atvejai	Apimami PA	
	Išplečiantys PA	
	Specializuojamai PA	
Po sąlyga		Praneši naudotojui apie sėkmingai užregistruotą naudotoją

Lentelė 22 Registrutotis

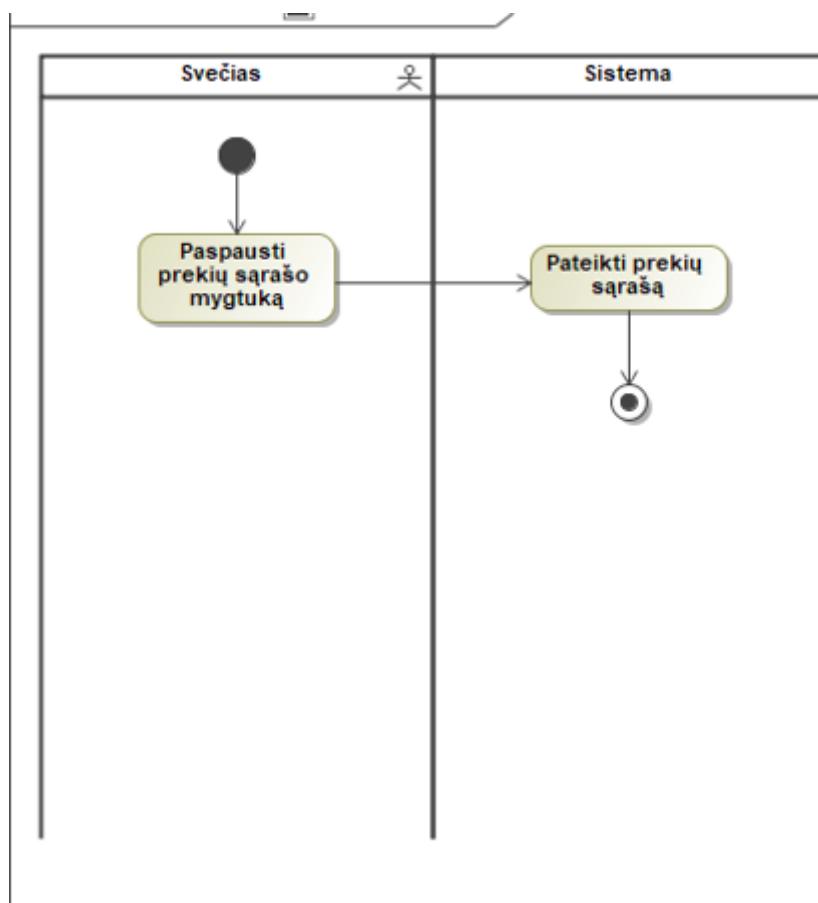


pav. 35 Panaudojimo atvejo "Registrutotis" scenarijus

Panaudojimo atvejis	7 "Peržiūrėti prekes"	Valentinas Kasteckis
Tikslas.		

Suteikti galimybę betkuriam naudotojui peržiūrėti prekes	
Apaščimas. PA suteikia galimybę naudotojui peržiūrėti prekių sąrašą	
Prieš sąlyga	Būti įsijungus sistemą
Aktorai	Svečias
Susiję panaudojimo atvejai	Apimami PA
	Išplečiantys PA
	Specializuojamai PA
Po sąlyga	Pateikiamas prekių sąrašas

Lentelė 23 Peržiūrėti prekes

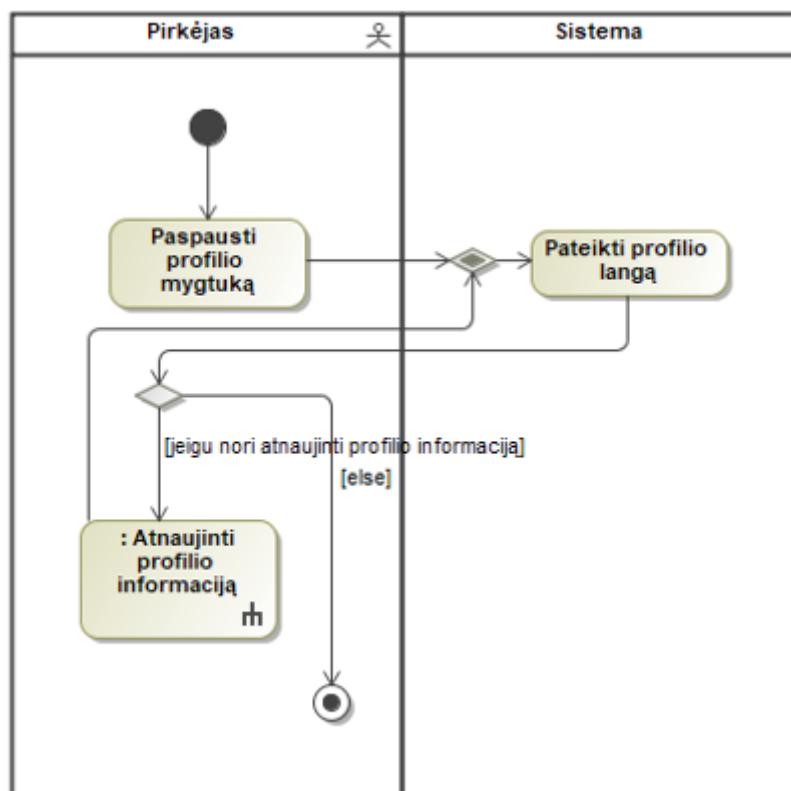


pav. 36 Panaudojimo atvejo "Peržiūrėti prekes" scenarijus

Panaudojimo atvejis	9 "Peržiūrėti profilį"	Valentinas Kasteckis
Tikslas.		
Suteikti galimybę naudotojui peržiūrėti jo profilį		
Apaščimas. PA suteikia galimybę peržiūrėti profilį naudotojui		
Prieš sąlyga	Naudotojas turi būti prisijungęs	
Aktorai	Pirkėjas	

Susiję panaudojimo atvejai	Apimami PA	
	Išplečiantys PA	27 "Atnaujinti profilio informaciją"
	Specializuojamiai PA	
Po sąlyga		Pateikiamas naudotojui jo profilis

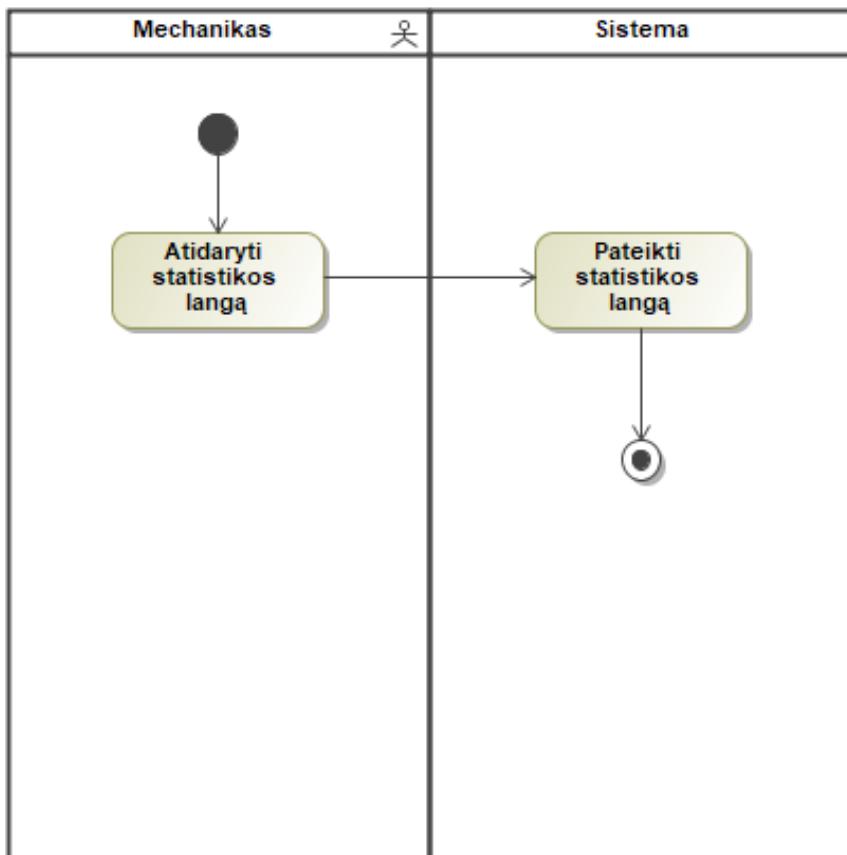
Lentelė 24 Peržiūrēti profilių



pav. 37 Panaudojimo atvejo "Peržiūrēti profilių" scenarijus

Panaudojimo atvejis	1 "Peržiūrēti darbo statistiką"	Eligijus Kiudys
Tikslias.		
Suteikti mechanikui peržiūrēti darbo statistiką		
Aprašymas. Mechanikas gali peržiūrėti savo dirbtą darbo statistiką.		
Prieš sąlyga		Mechanikas yra prisijungęs, patvirtintas ir darbo puslapyje
Aktorai		Mechanikas
Susiję panaudojimo atvejai	Apimami PA	
	Išplečiantys PA	
	Specializuojamiai PA	
Po sąlyga		Mechaniko statistika peržiūrėta

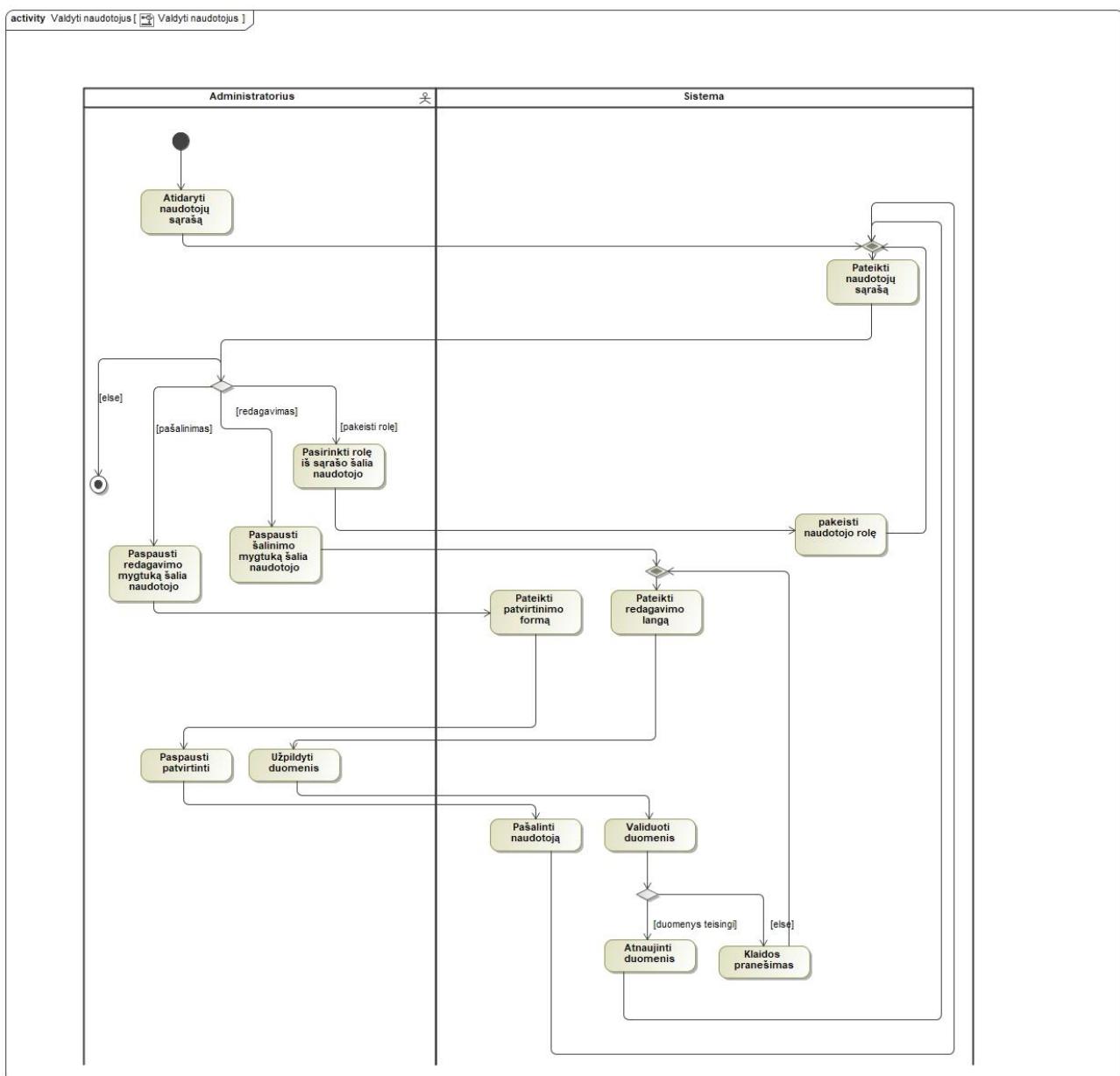
Lentelė 25 Peržiūrēti darbo statistiką



pav. 38 Panaudojimo atvejo "Peržiūrėti darbo statistiką" scenarijus

Panaudojimo atvejis	10 "Valdyti naudotojus"			
Tikslas.				
Leisti administratoriui ištrinti arba redaguoti naudotojus				
Apaščias. Naudotojų valdymas suteikia administratoriui naudotojų valdymą: redagavimą bei šalinimą.				
Prieš sąlyga		Administratorius yra prisijungęs ir yra administravimo puslapyje		
Aktorai		Administratorius		
Susiję panaudojimo atvejai	Apimami PA			
	Įspiečiantys PA			
	Specializuojamai PA			
Po sąlyga		Ištrinti naudotojai arba redaguoti naudotojai yra atnaujinti		

lentelė 26 Valdyti naudotojus

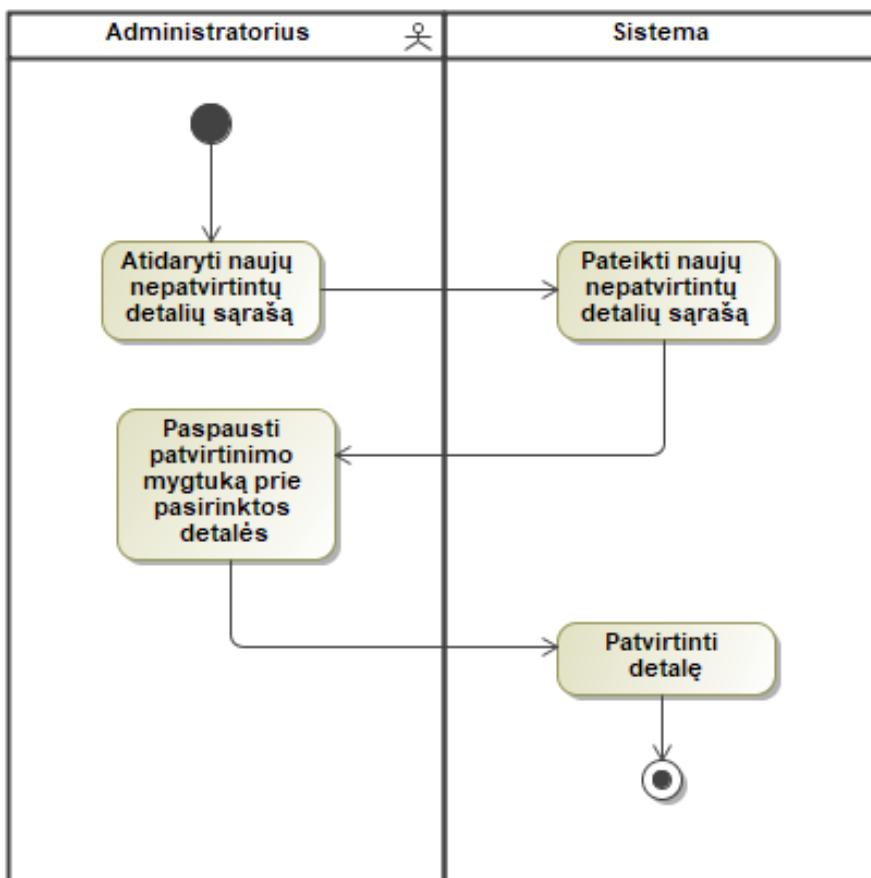


pav. 39 Panaudojimo atvejo "Valdyti naudotojus" scenarijus

Panaudojimo atvejis	20 "Patvirtinti naujas detales"	Eligijus Kiudys
Tikslias.		
Suteikti galimybę administratoriui patvirtinti prekes		
Aprašymas. Naujų detalių patvirtinimas suteikia administratoriui galimybę patvirtinti pasirinktą nepatvirtintą prekę.		
Prieš sąlyga		Administratorius yra prisijungęs ir yra administravimo puslapyje
Aktorai		Administratorius
Susiję panaudojimo atvejai	Apimami PA	
	Įspėjiantys PA	
	Specializuojami PA	

Po sąlyga	Patvirtintos naujos prekės
------------------	----------------------------

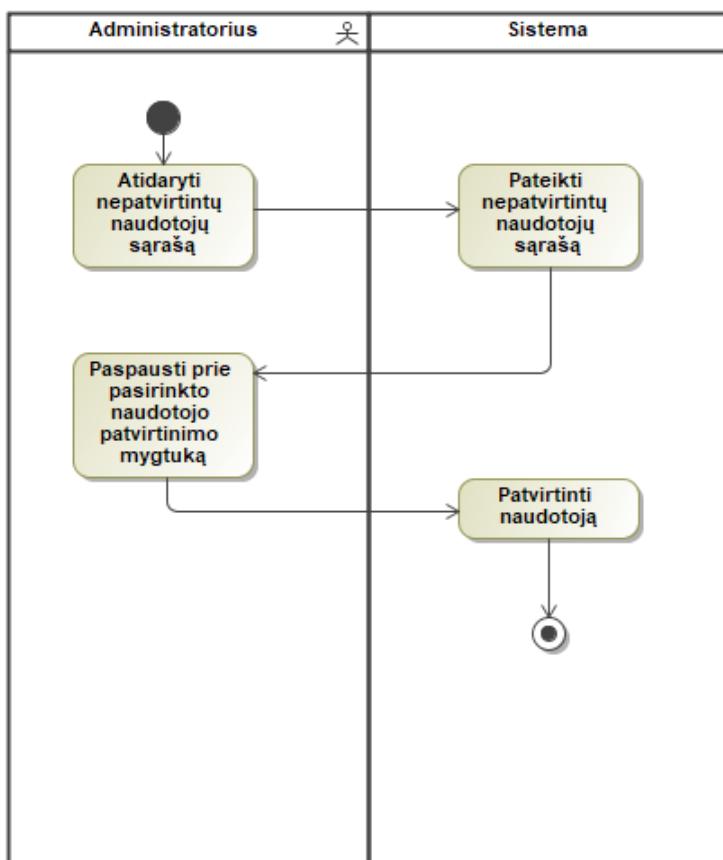
lentelė 27 Patvirtinti naujas detales



pav. 40 Panaudojimo atvejo "Patvirtinti naujas detales" scenarijus

Panaudojimo atvejis	23 "Patvirtinti naujus naudotojus"	Eligijus Kiudys
Tikslas.		
Suteikti administratoriui galimybę patvirtinti naujus naudotojus		
Apaščymas. Administratoriui suteikiama galimybė aktyvuoti pasirinktą naudotoją.		
Prieš sąlyga		Administratorius yra prisijungęs ir yra administravimo puslapyje
Aktorai		Administratorius
Susiję panaudojimo atvejai	Apimami PA	
	Išplečiantys PA	
	Specializuojamai PA	
Po sąlyga		Patvirtintas naujas naudotojas

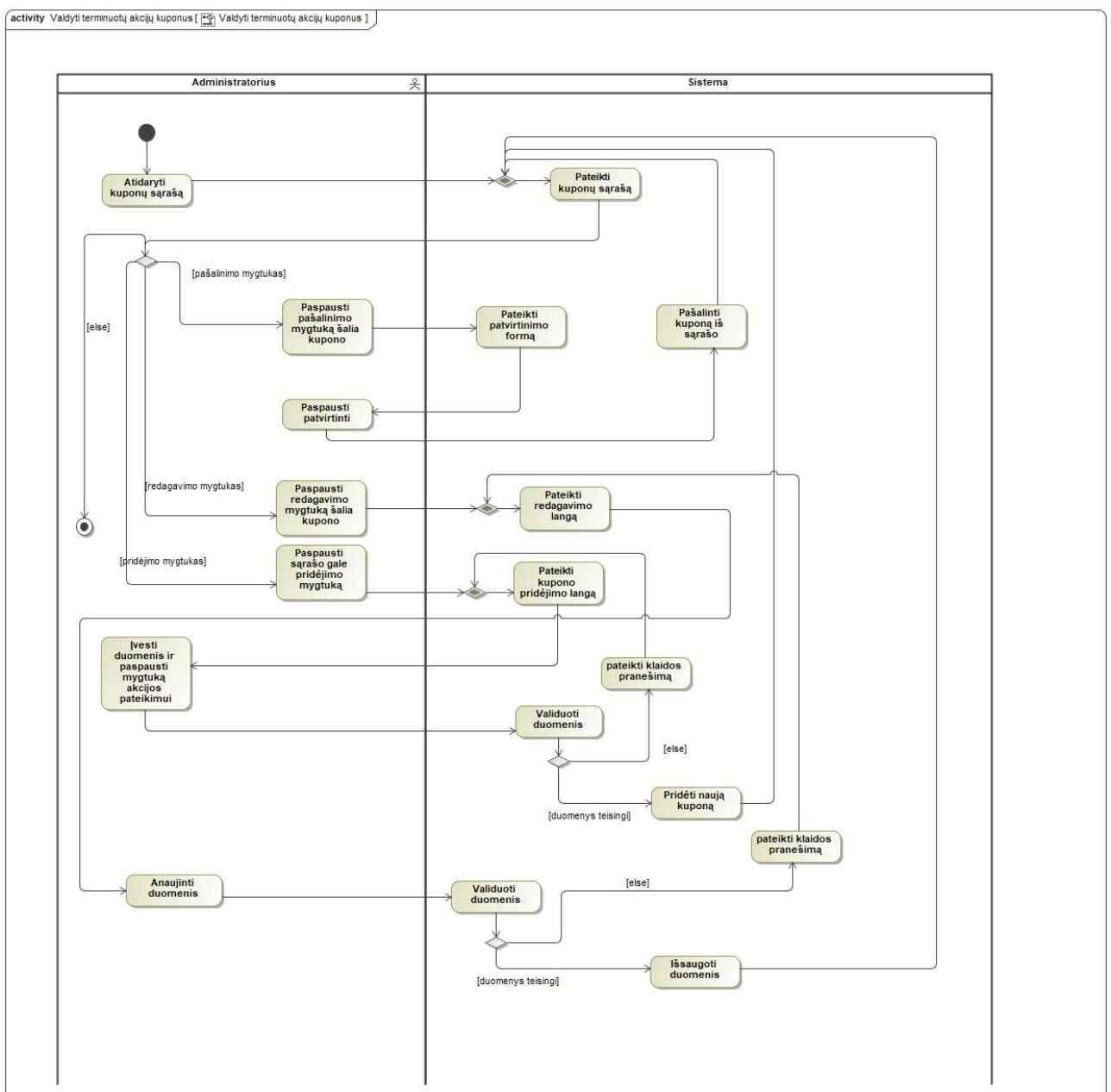
lentelė 28 Patvirtinti naujus naudotojus



pav. 41 Panaudojimo atvejo "Patvirtinti naujus naudotojus" scenarijus

Panaudojimo atvejis	28 "Valdyti terminuotų akcijų kuponus"	Eligijus Kiudys
Tikslas.		
Suteikti administratoriui galimybę redaguoti, ištrinti ir sukurti treminuotų akcijų kuponus		
Aprašymas. Terminuotų akcijų kuponų valdymas suteikia administratoriui terminuotų akcijų kuponų valdymą: redagavimą, šalinimą bei pridėjimą.		
Prieš sąlyga		Administratorius yra prisijungęs ir yra administravimo puslapyje
Aktorai		Administratorius
Susiję panaudojimo atvejai	Apimami PA	
	Išplečiantys PA	
	Specializuojamai PA	
Po sąlyga		Sukurti, ištrinti ir redaguoti terminuotu akcijų kuponai yra atnaujinti

Lentelė 29 Valdyti terminuotų akcijų kuponus

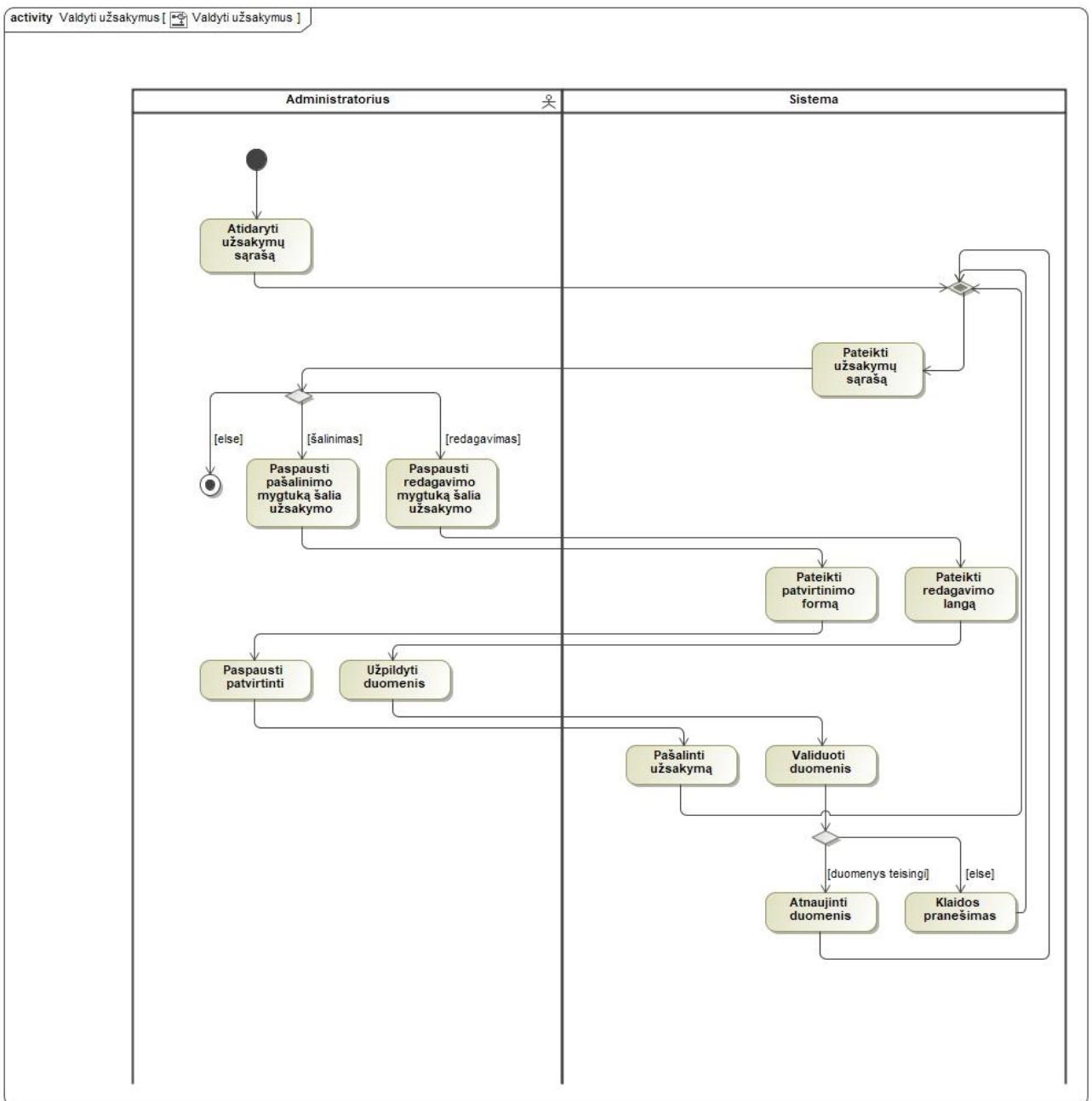


pav. 42 Panaudojimo atvejo "Valdyti terminuotų akcijų kuponus" scenarijus

Panaudojimo atvejis	8 "Valdyti užsakymus"	Eligijus Kiudys
Tikslas. Leisti administratoriui ištrinti arba redaguoti užsakymus		
Aprašymas. Užsakymų valdymas suteikia administratoriui tokias galimybes: užsakymų redagavimą bei užsakymų šalinimą.		
Prieš sąlyga Administratorius yra prisijungęs ir yra administravimo puslapyje		
Aktoriai Administratorius		
Susiję panaudojimo atvejai	Apimami PA	
	Išplečiantys PA	

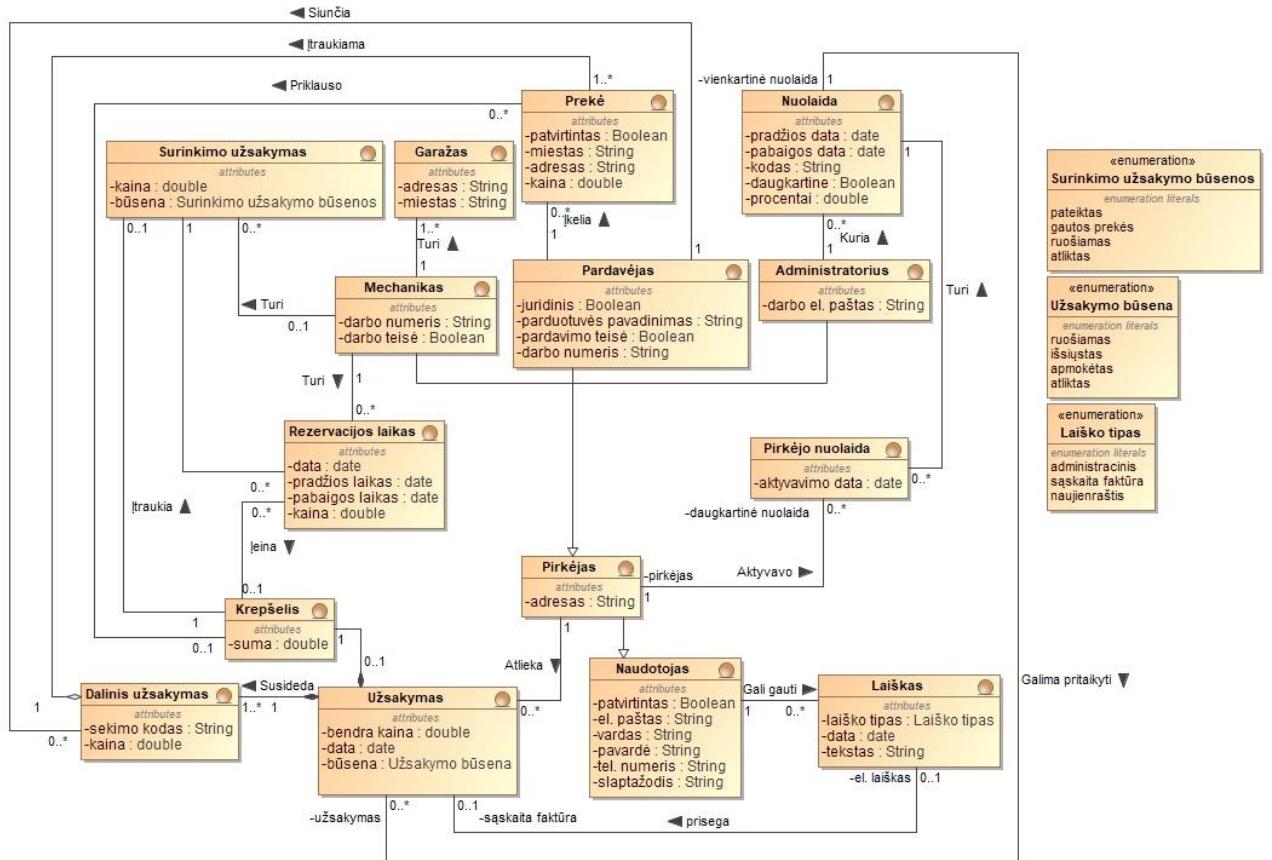
	Specializuojami PA	
Po sąlyga	Ištrinti užsakymai arba redaguoti užsakymai yra atnaujinti	

Lentelė 30 Valdyti užsakymus



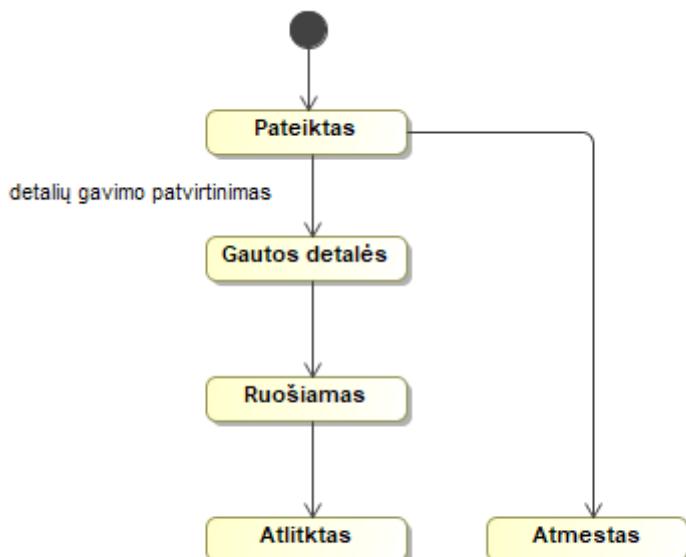
pav. 43 Panaudojimo atvejo "Valdyti užsakymus" scenarijus

3.2 Dalykinės srities modelis

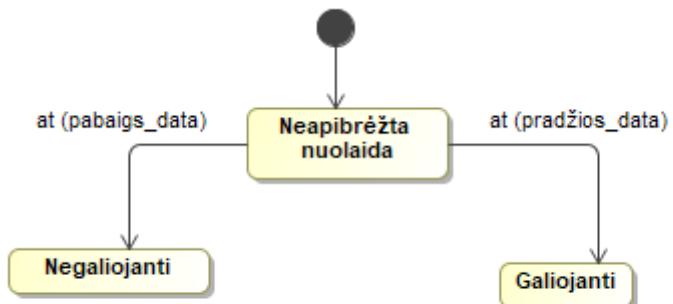


pav. 44 Dalykinės srities modelis

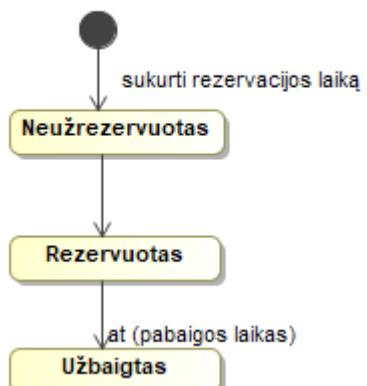
3.3 Būsenų diagramos



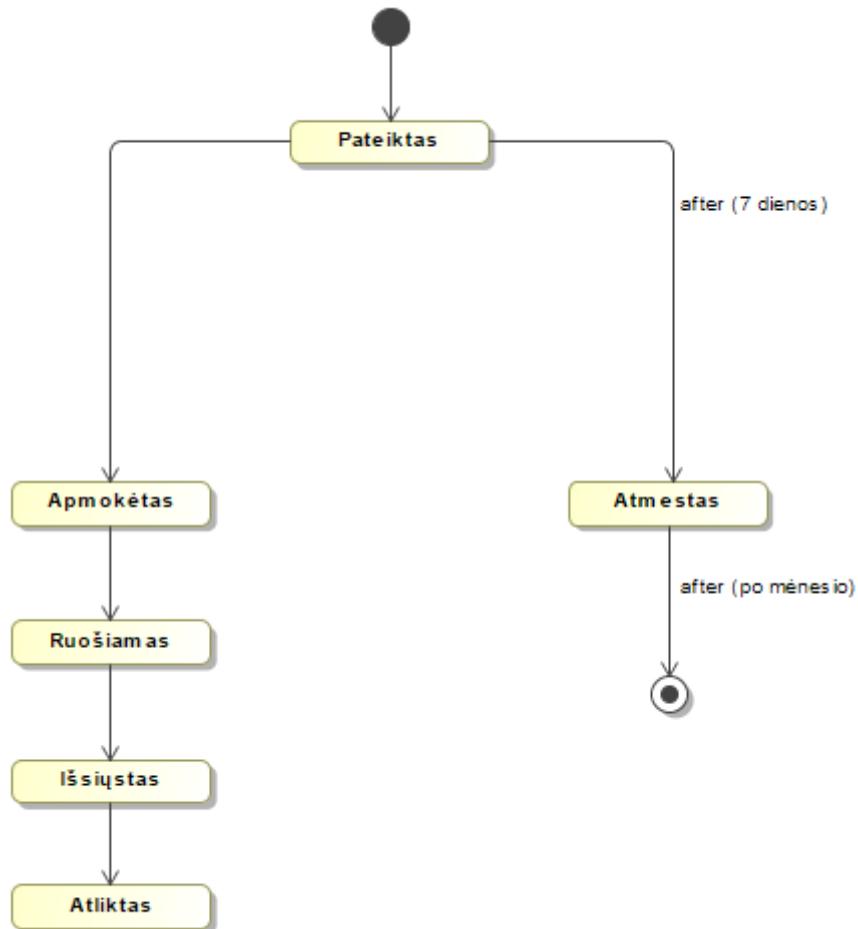
pav. 45 Surinkimo užsakymo būsenos



pav. 46 Nuolaidos užsakymo būsena



pav. 47 Rezervacijos laikas būsena



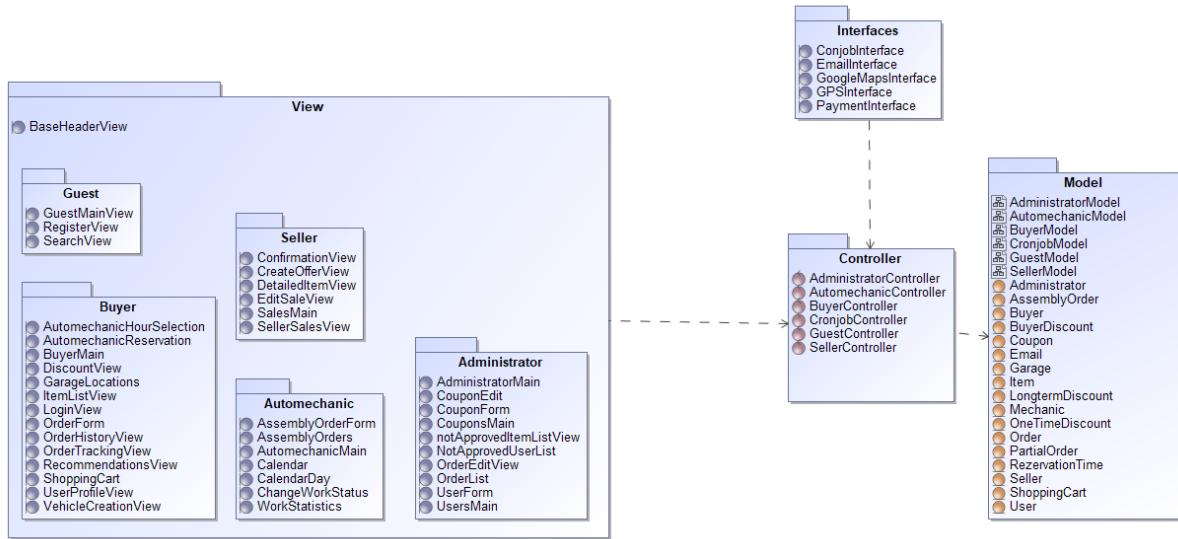
pav. 48 Užsakymo būsenų diagrama

3.4 Pasirinkta programavimo kalba, technologija

Šios sistemos kūrimui naudosime C# programavimo kalbą. Duomenų bazės projektavimui naudojame „MagicDraw“ modeliavimo įrankį. Naudosime .NET Core karkasą.

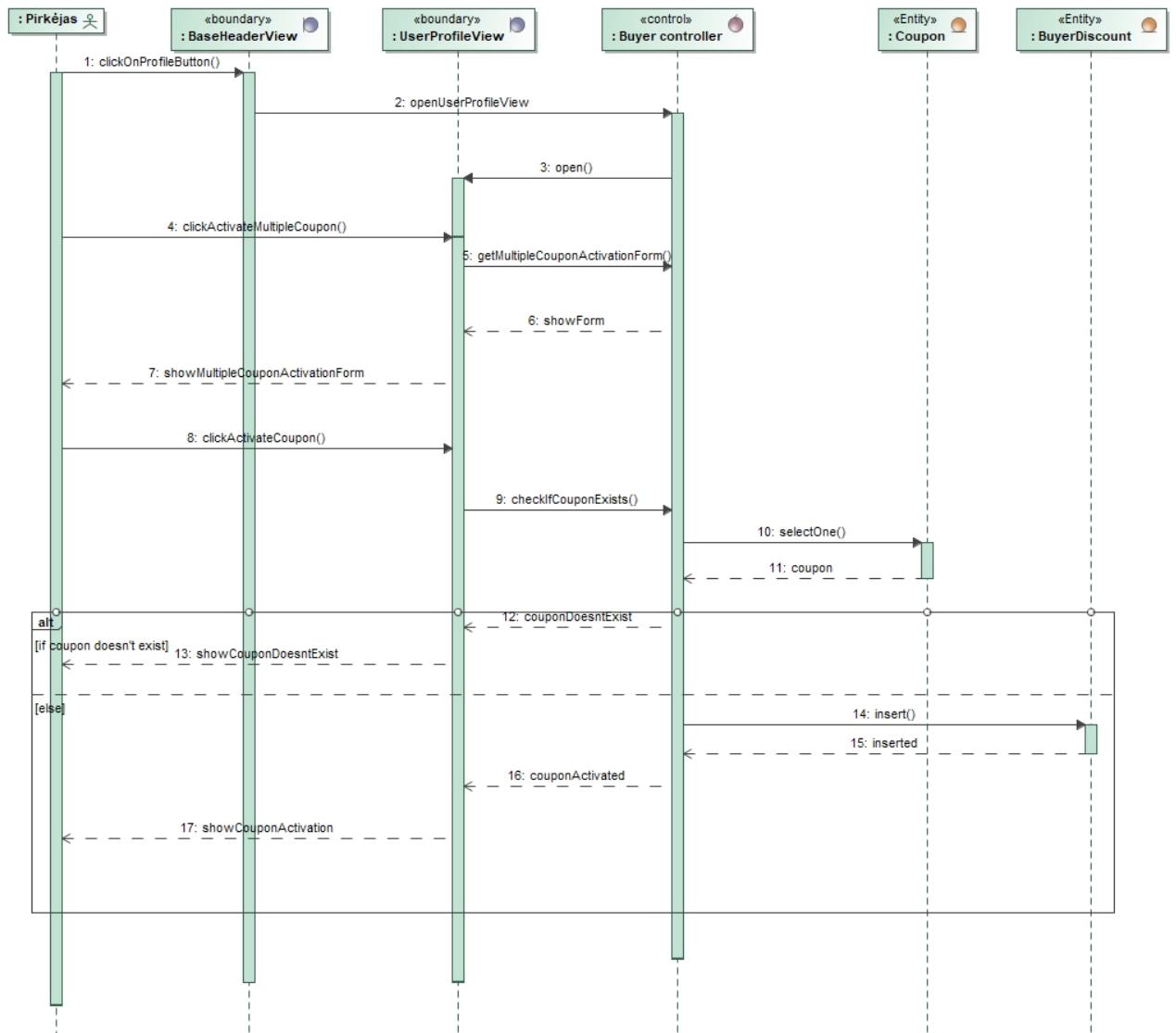
4 Sistemos projekto modelis

4.1 Sistemos loginė architektūra

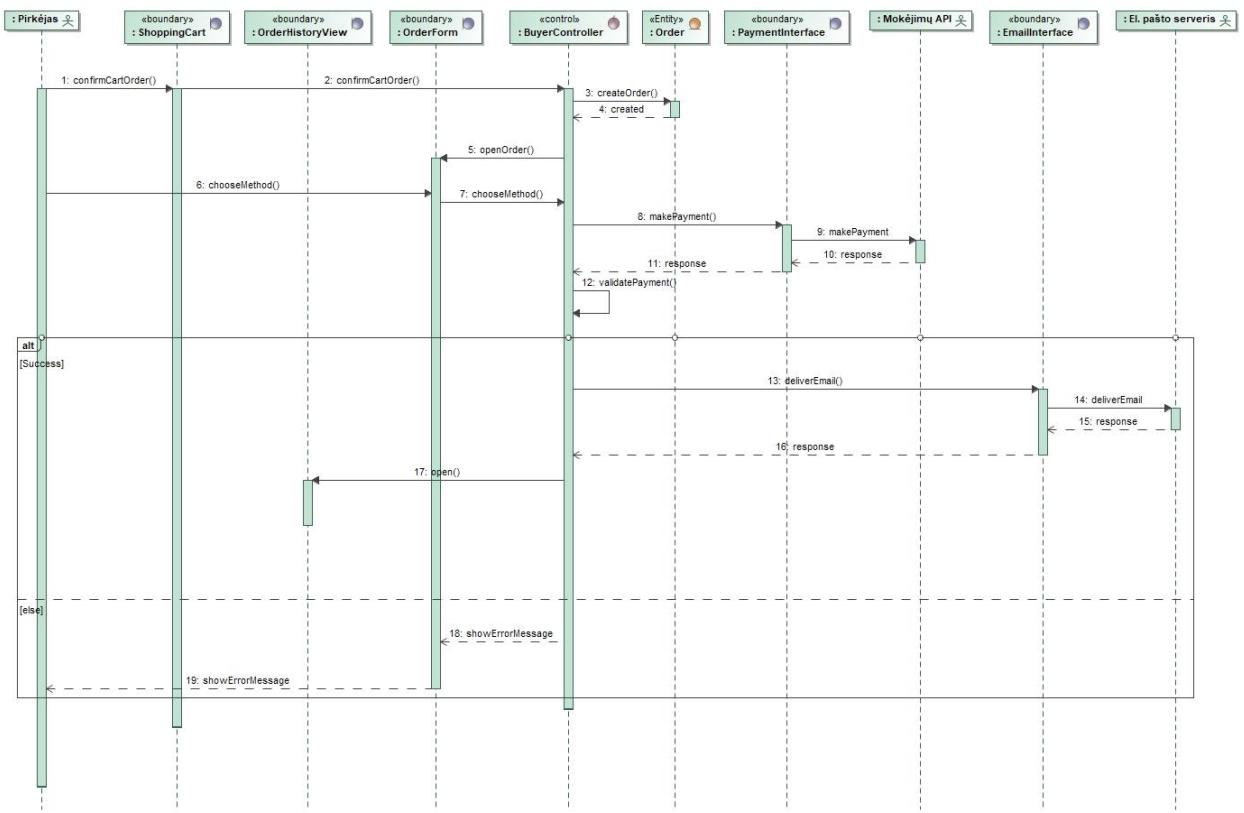


Pav. 49 loginė sistemos architektūra

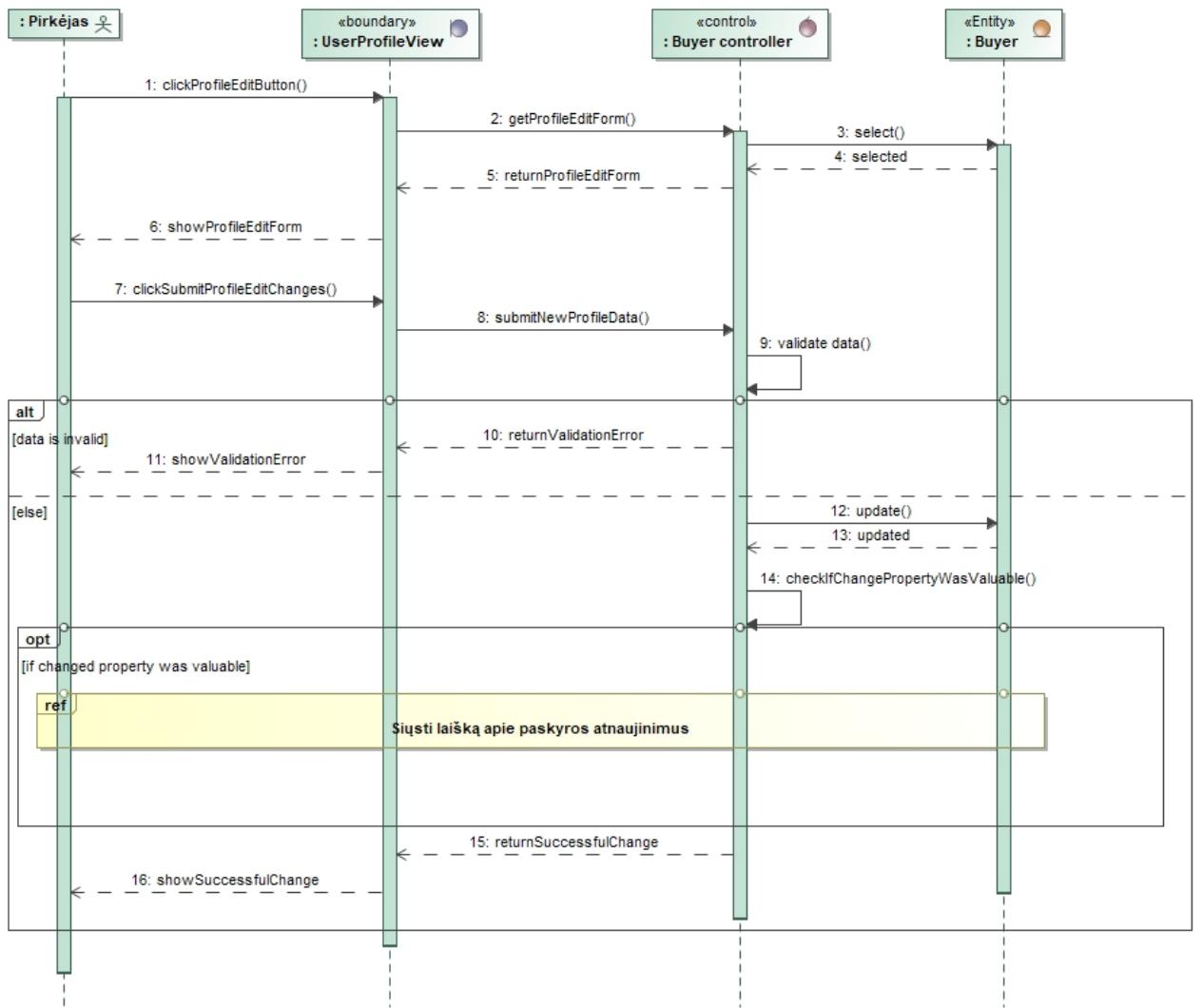
4.2 Panaudojimo atvejų sekų diagramos



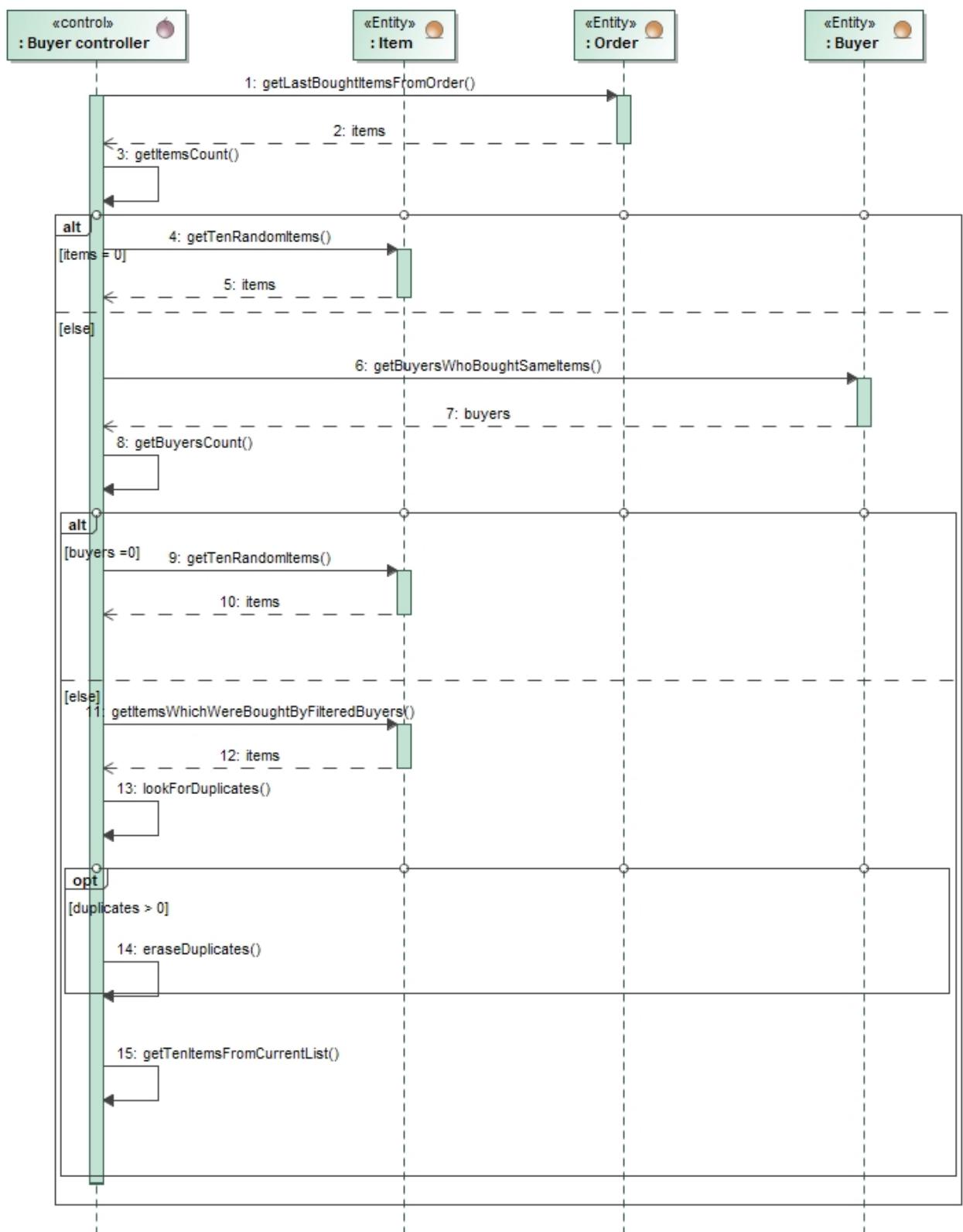
Pav. 50 Terminuotos nuolaidos sekų diagrama



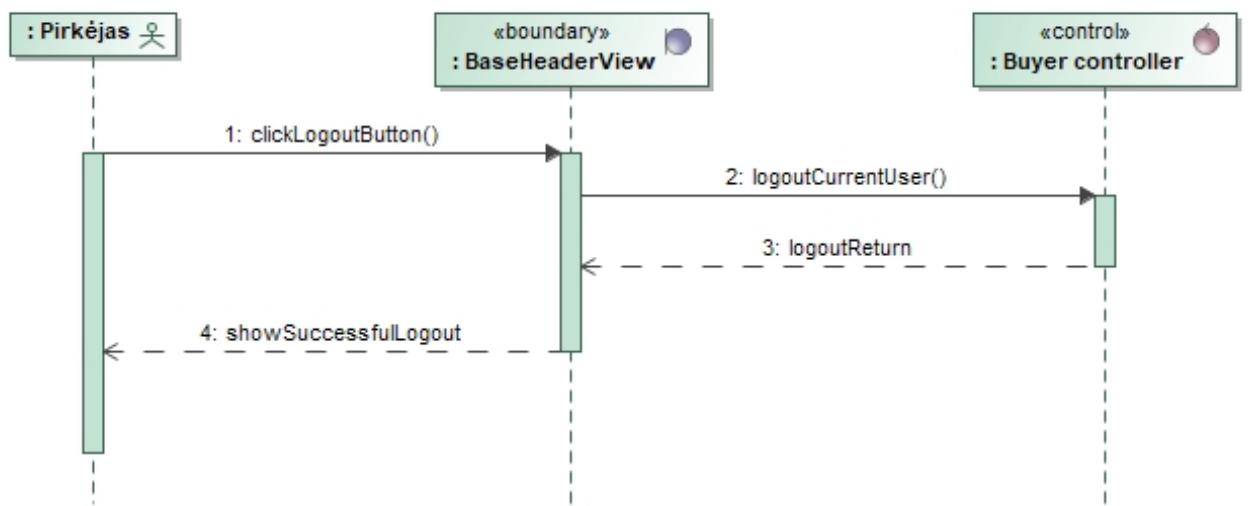
Pav. 51 Atlīkti uzsakymu sekūdiagramma



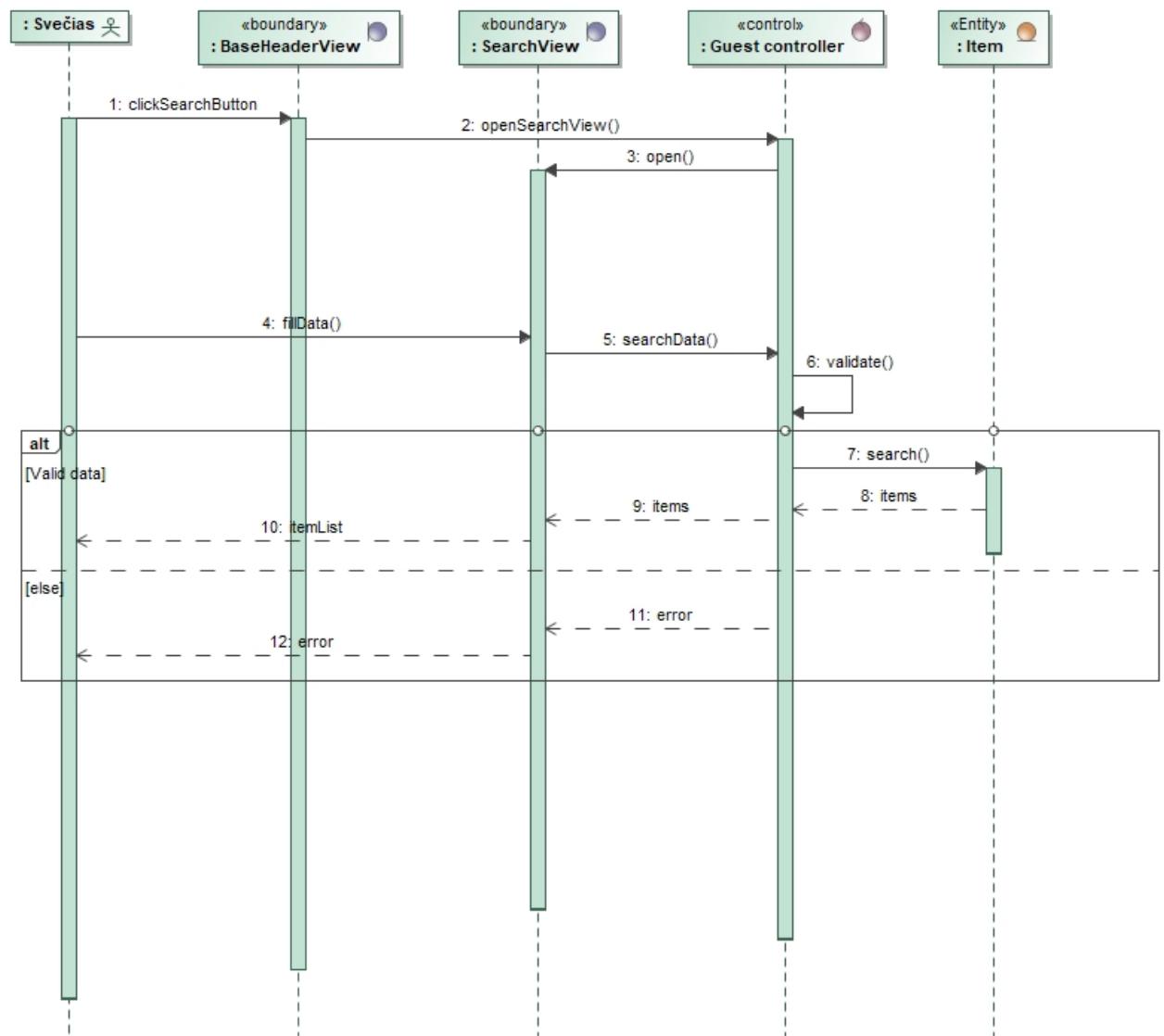
Pav. 52 Atnaujinti profilio informaciją sekų diagrama



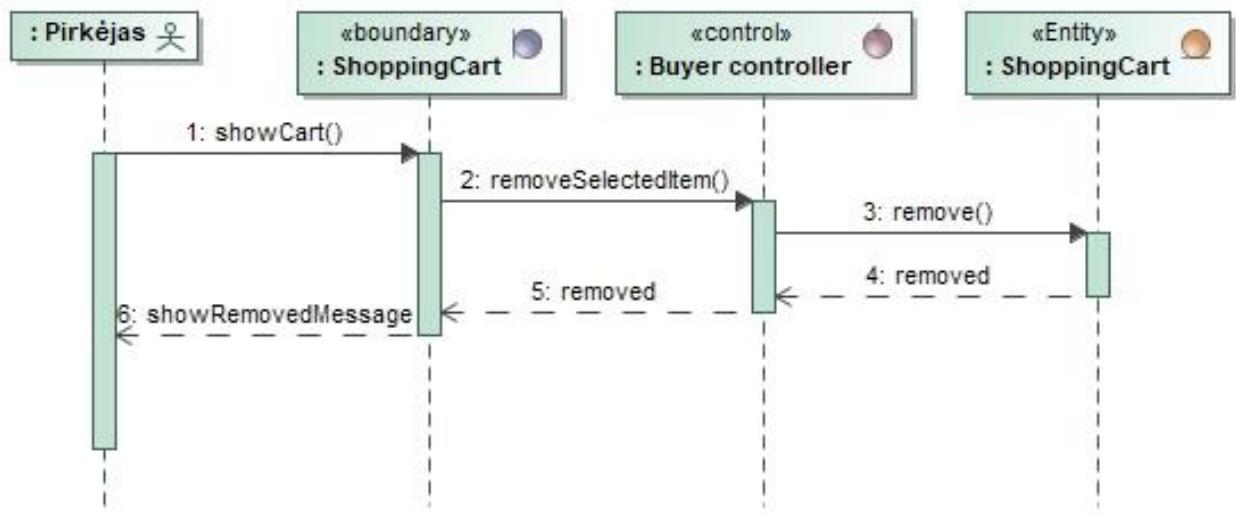
Pav. 53 Atrinkti rekomenduojamų prekes sekų diagrama



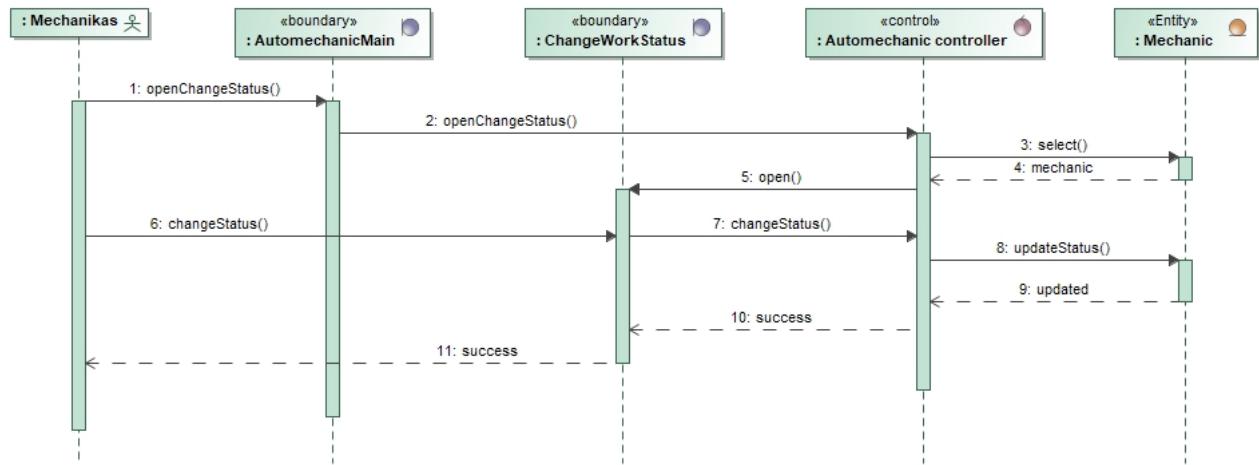
Pav. 54 Atsijungti sekų diagrama



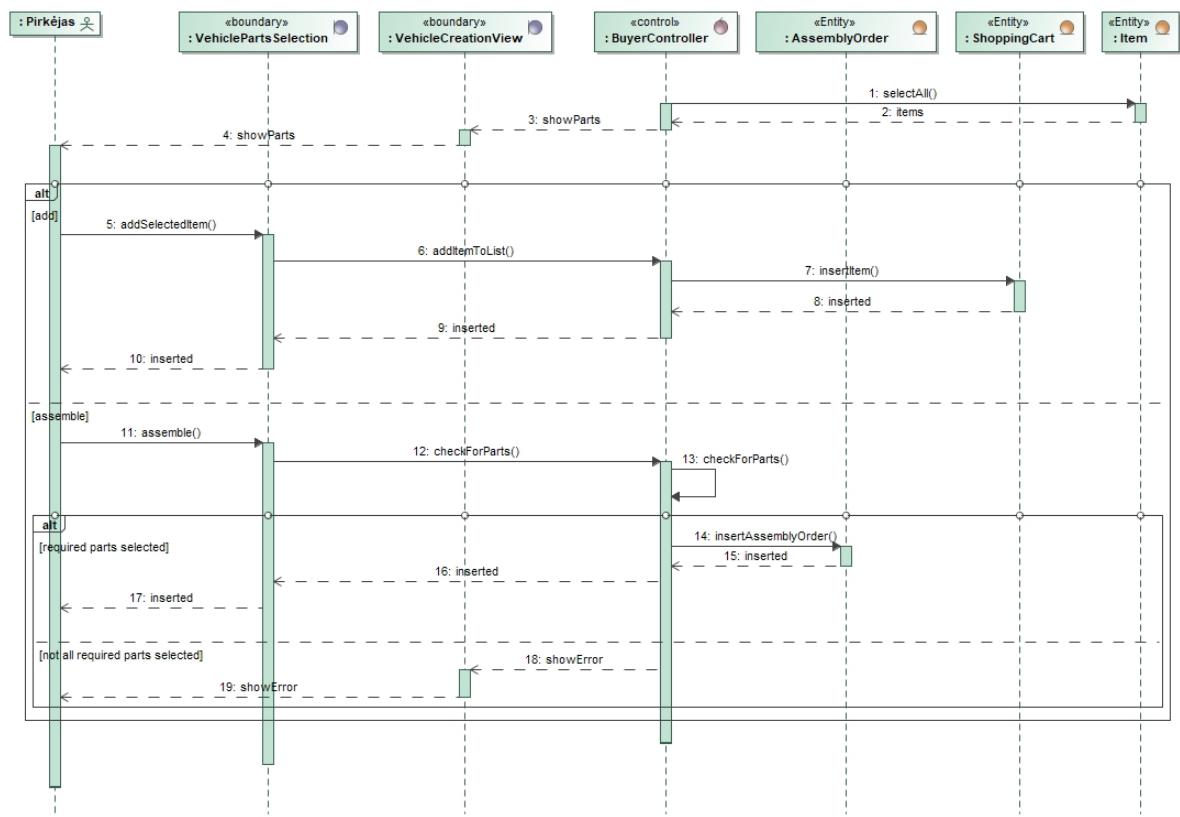
Pav. 55 ieškoti prekių sekų diagrama



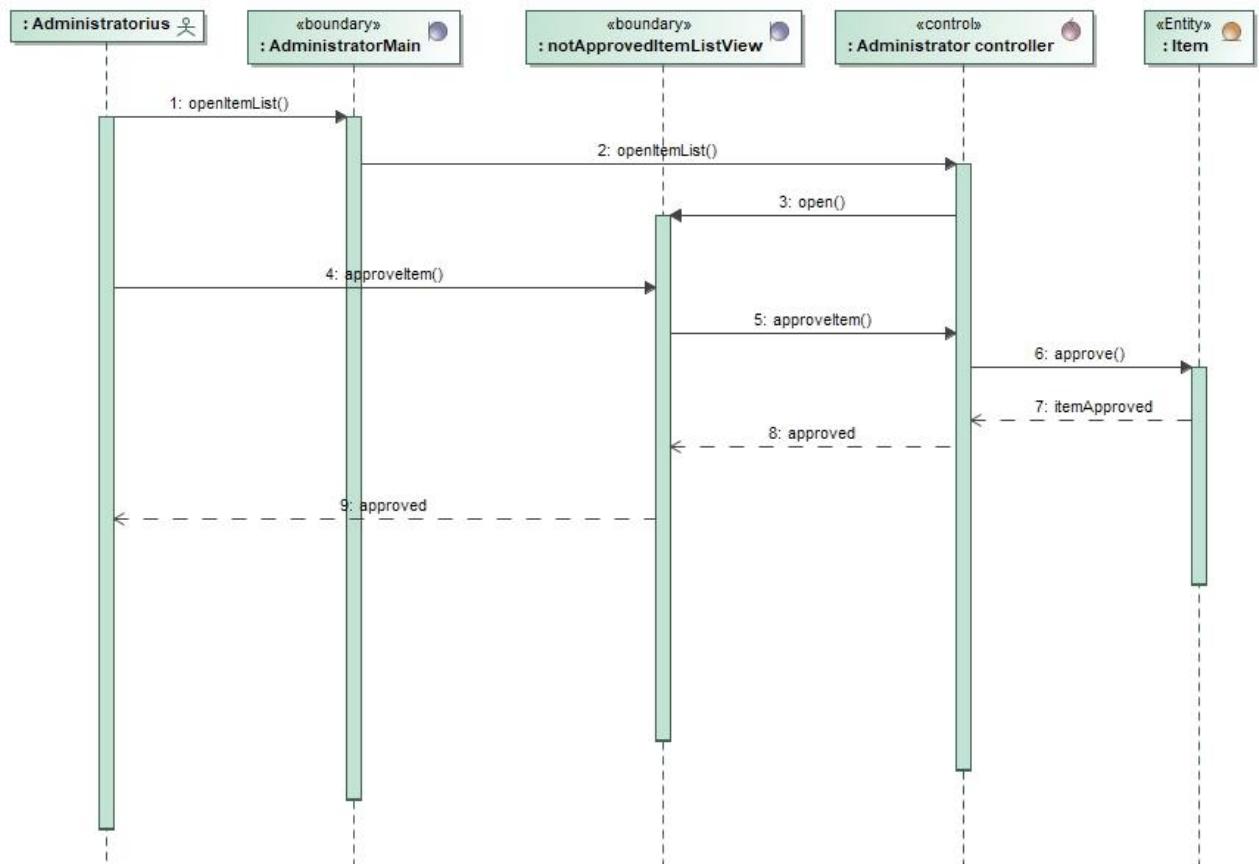
Pav. 56 Ištrinti prekę iš krepšelio sekų diagrama



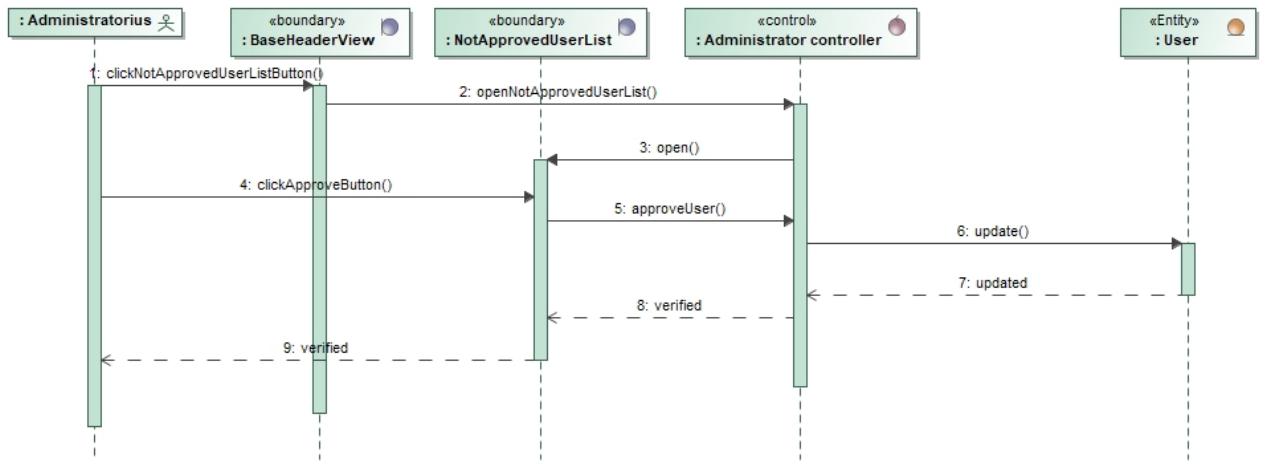
Pav. 57 Keisti darbo būseną sekų diagrama



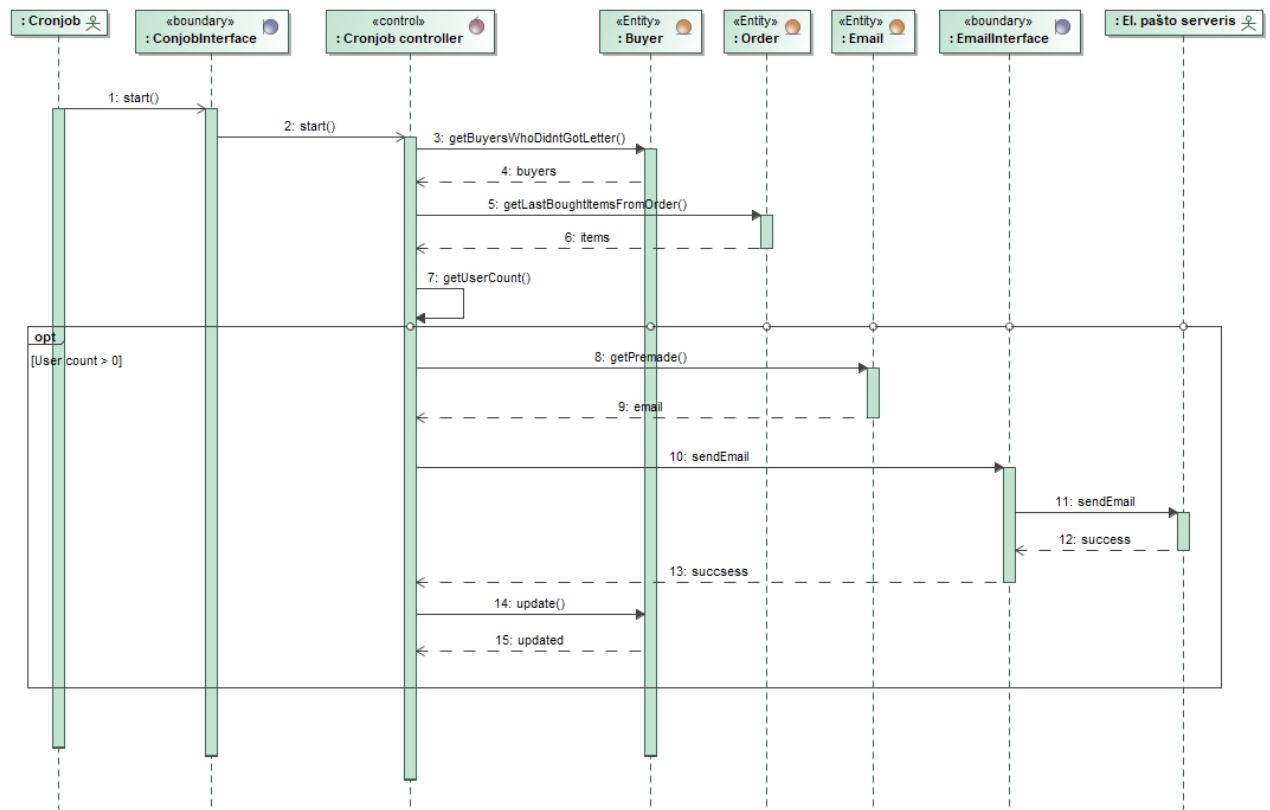
Pav. 58 Pasirinkti automobilio dalį sekų diagrama



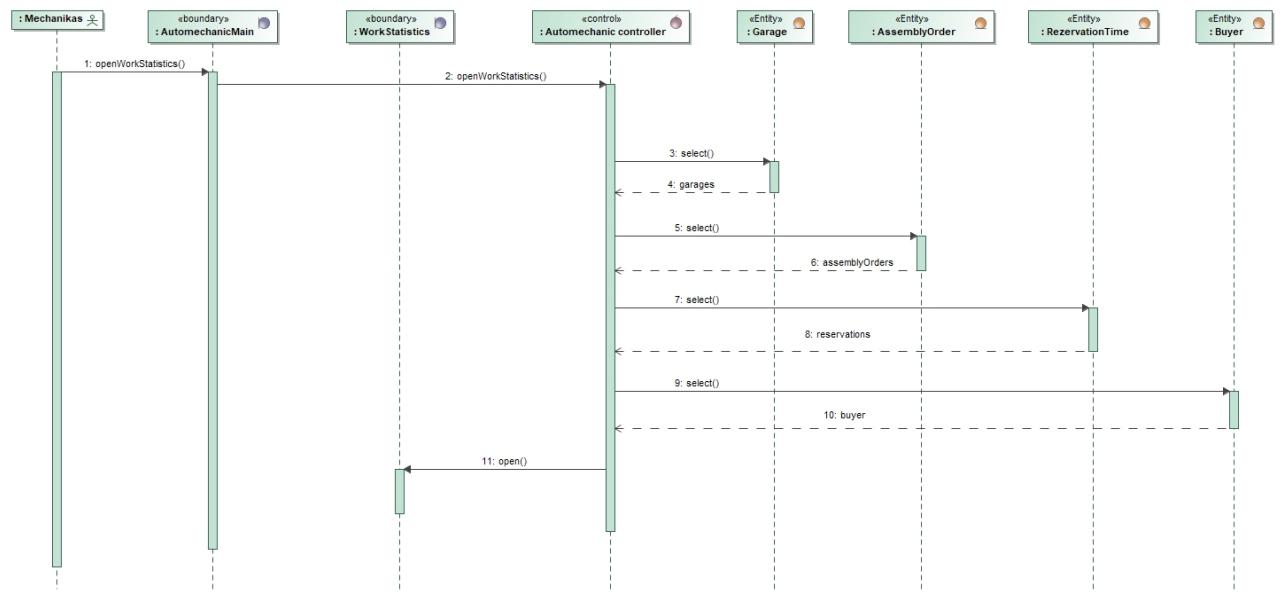
Pav. 59 Patvirtinti naujas detales sekų diagrama



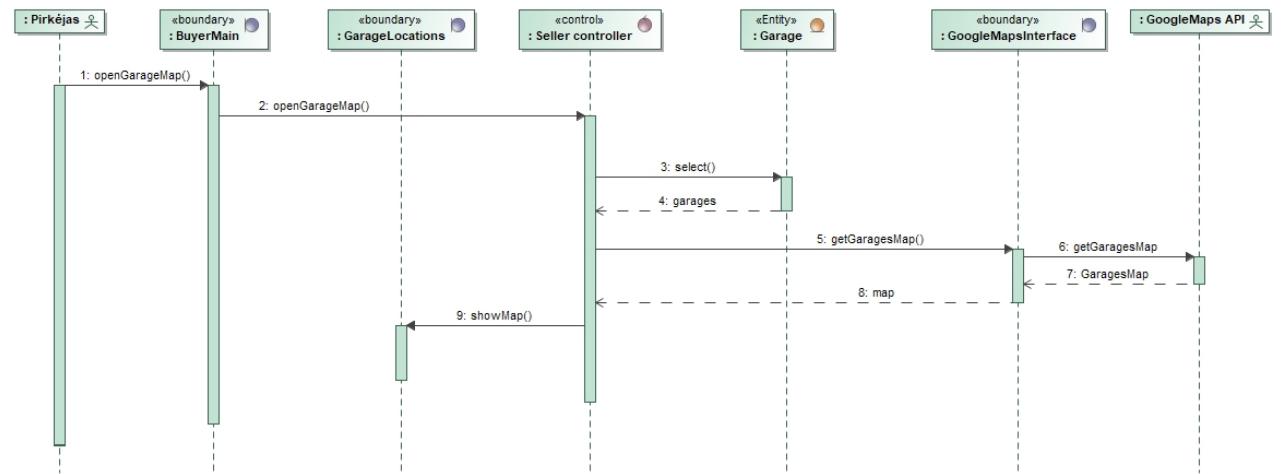
Pav. 60 Patvirtinti naujus naudotojus sekų diagrama



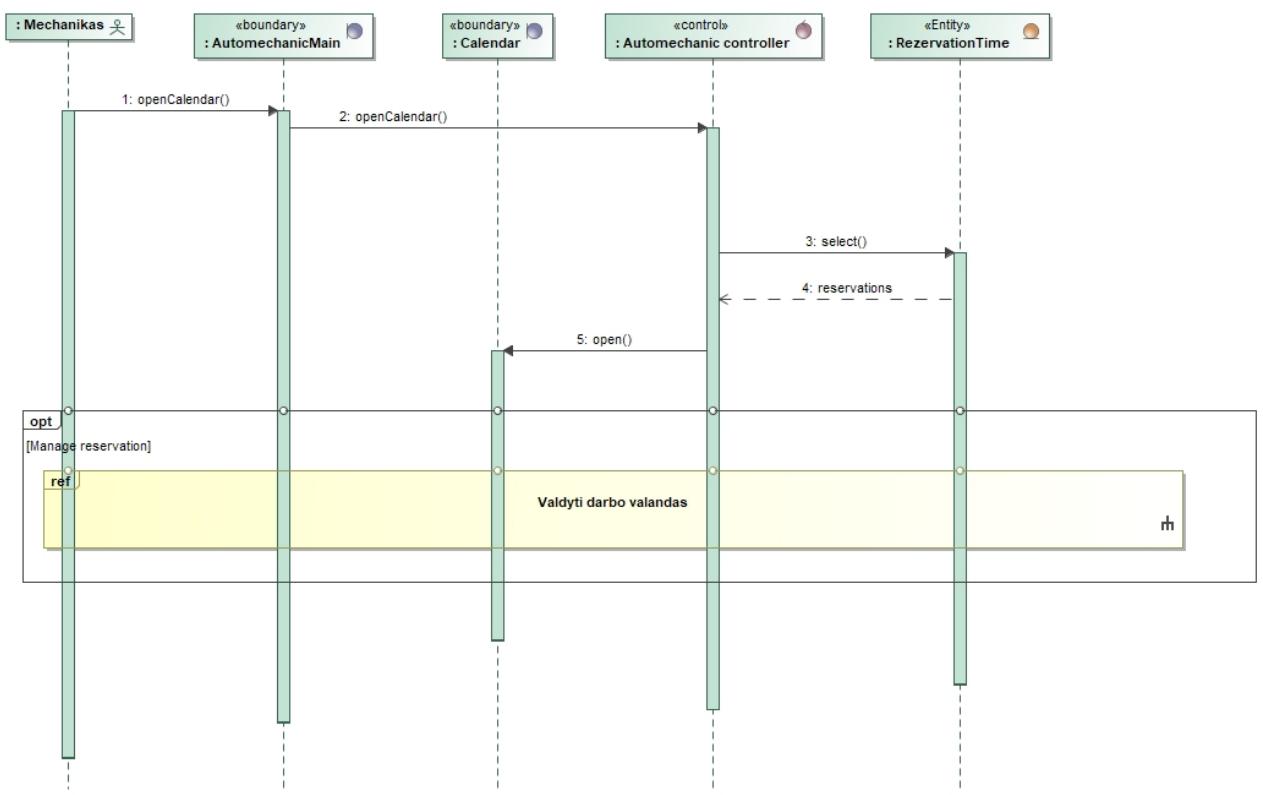
Pav. 61 Periodiškai siųsti pasiūlymus sekų diagrama



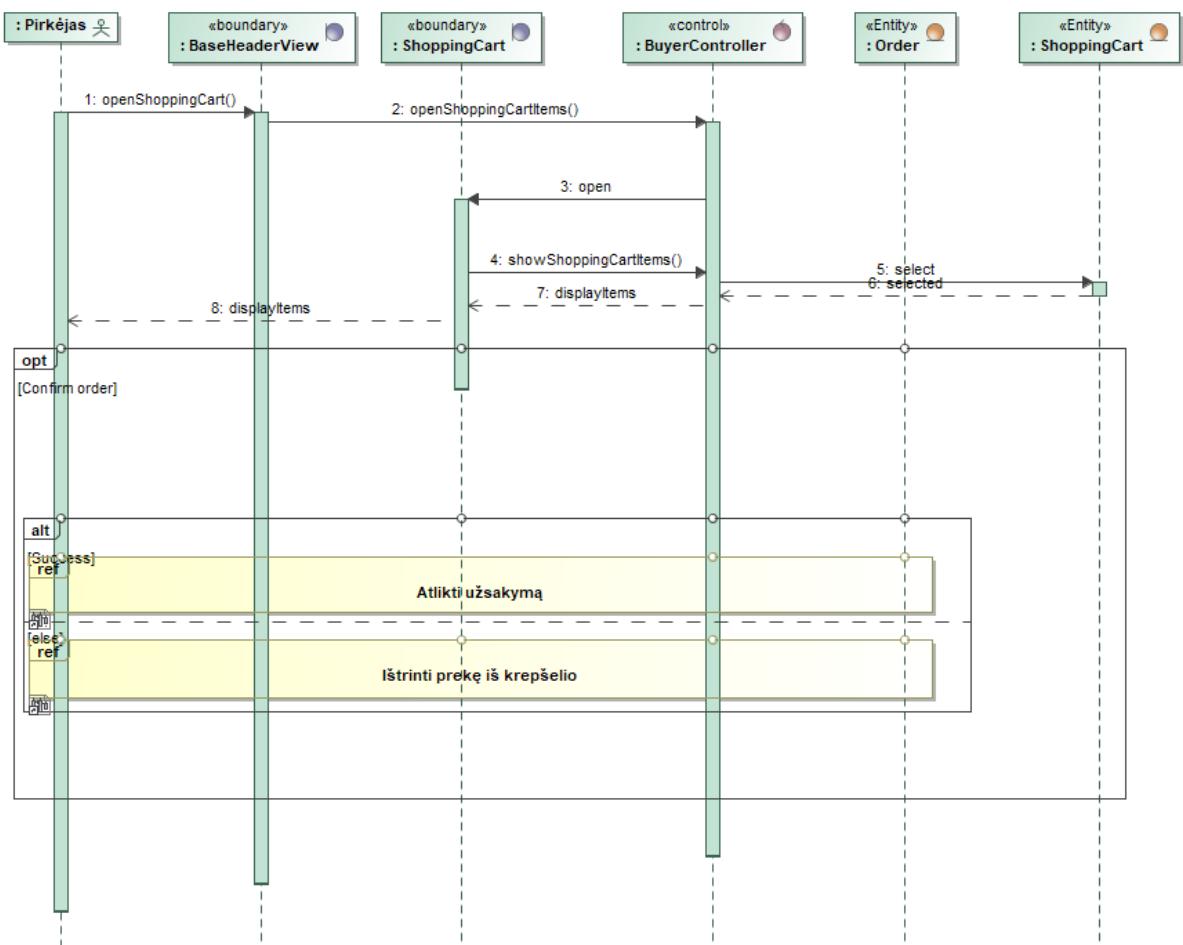
Pav. 62 Peržiūrėti darbo statistiką sekų diagrama



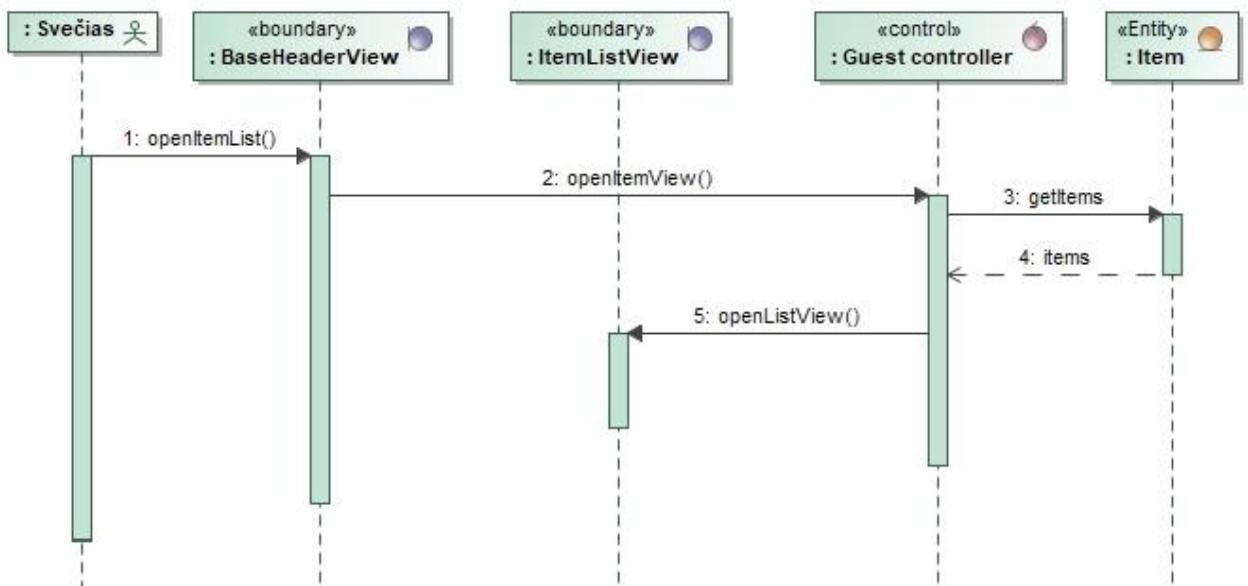
Pav. 63 Peržiūrėti garažų lokacijų žemėlapį sekų diagrama



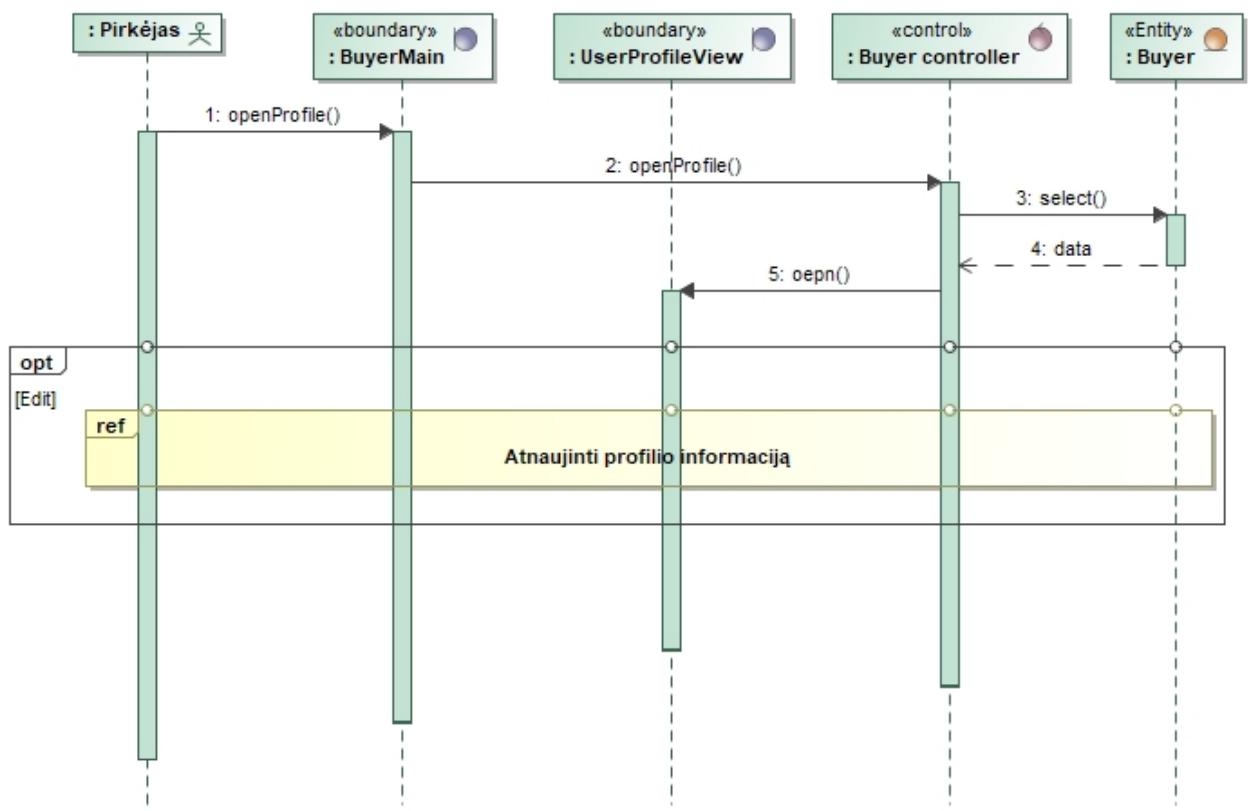
Pav. 64 Peržiūrėti kalendorių sekų diagrama



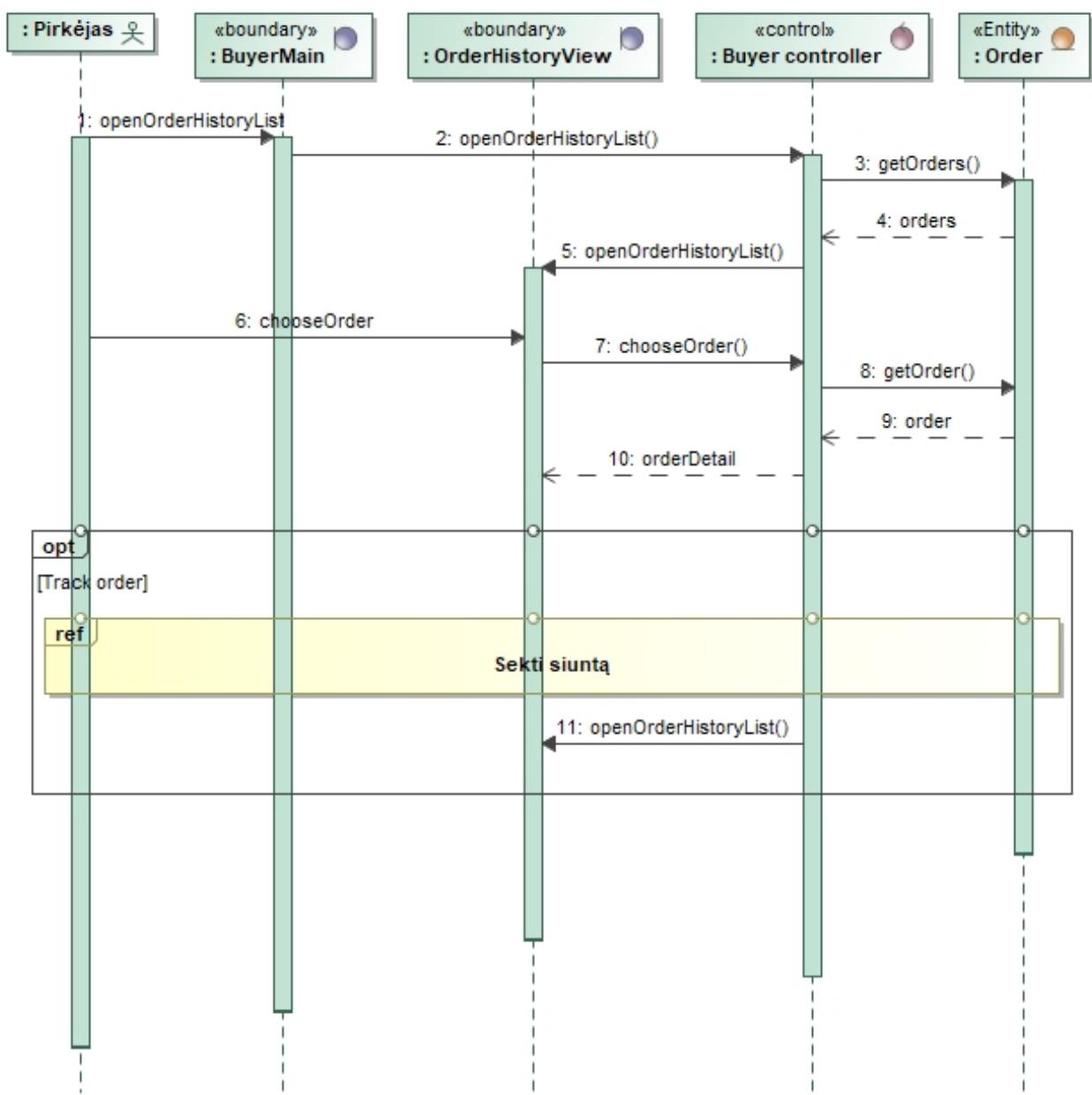
Pav. 65 Peržiūrėti krepšelį sekų diagrama



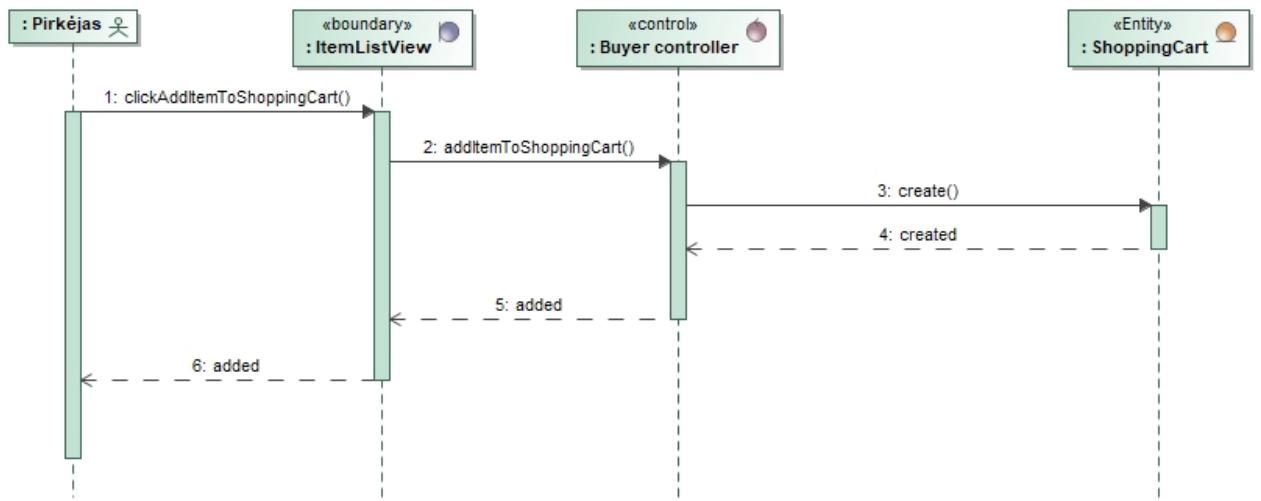
Pav. 66 Peržiūrėti prekes sekų diagrama



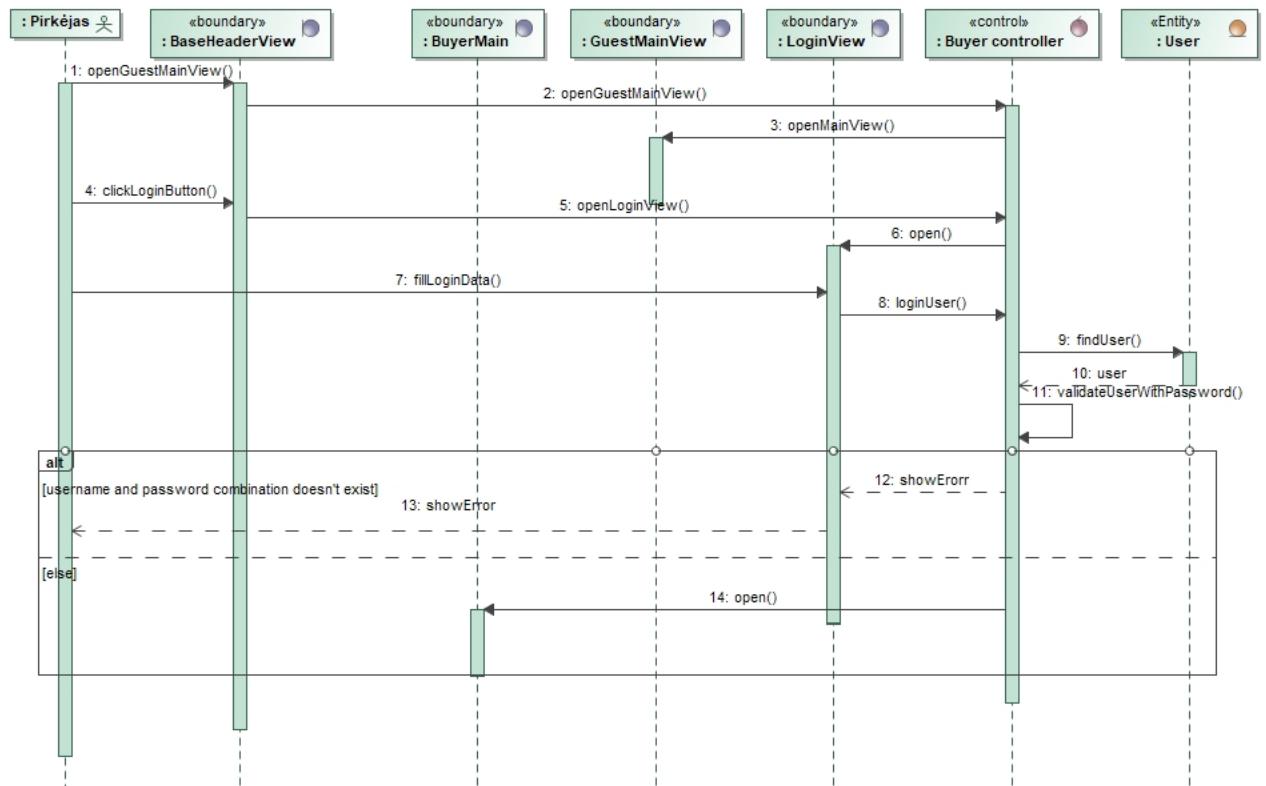
Pav. 67 Peržiūrėti profilį sekų diagrama



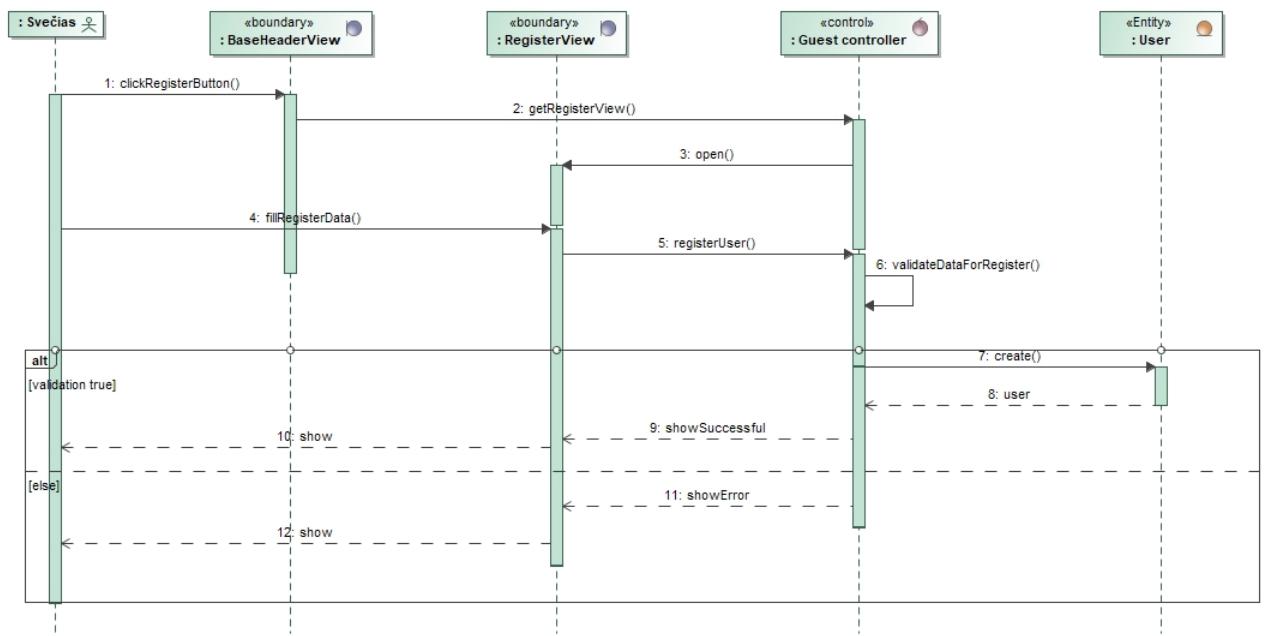
Pav. 68 Peržiūrėti užsakymą sekų diagrama



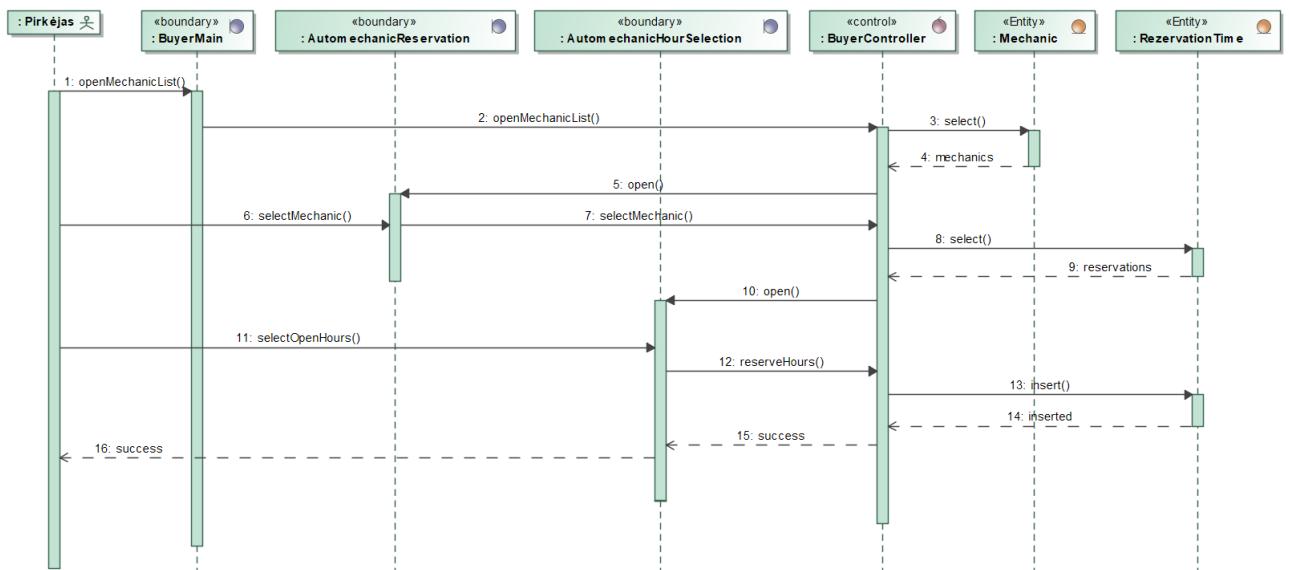
Pav. 69 Pridėti prekę į krepšelį sekų diagrama



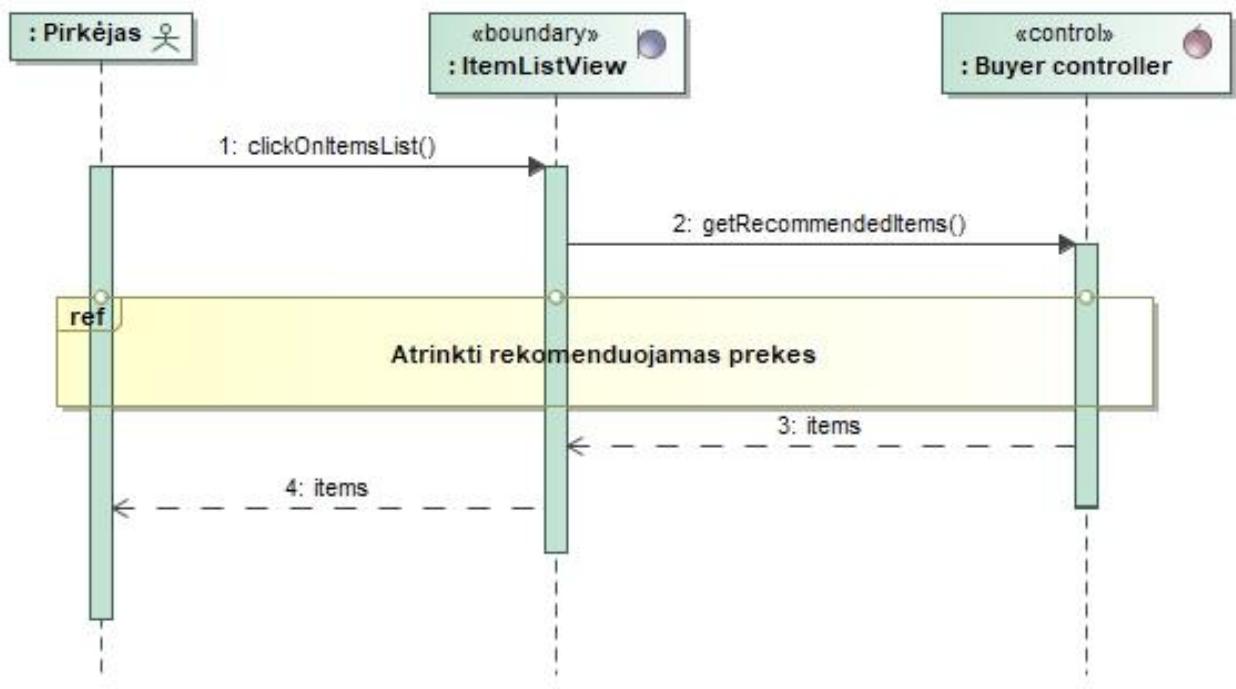
Pav. 70 Prisijungti sekų diagrama



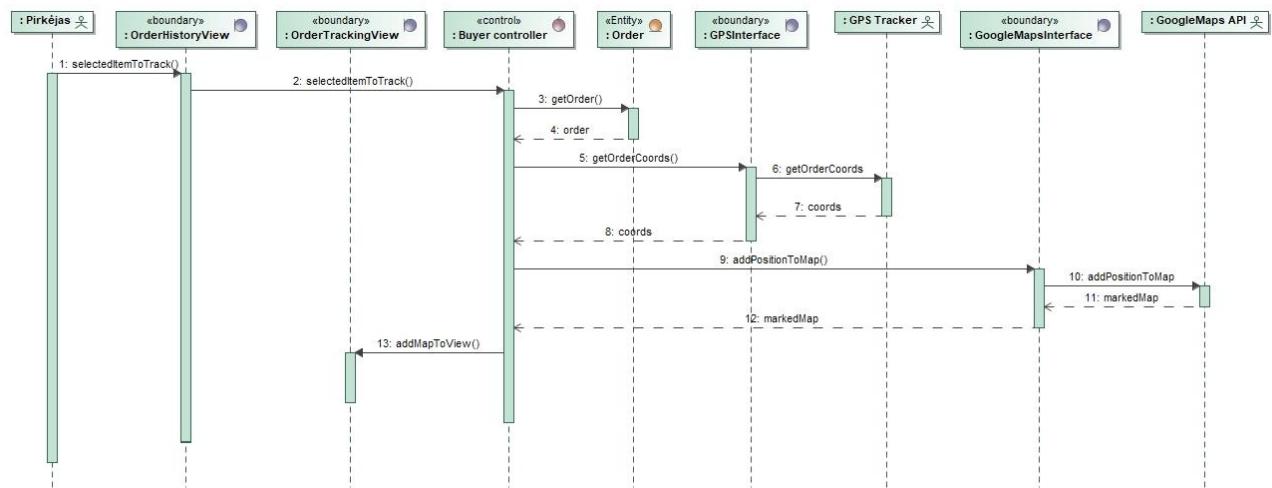
Pav. 71 Registruotis sekų diagrama



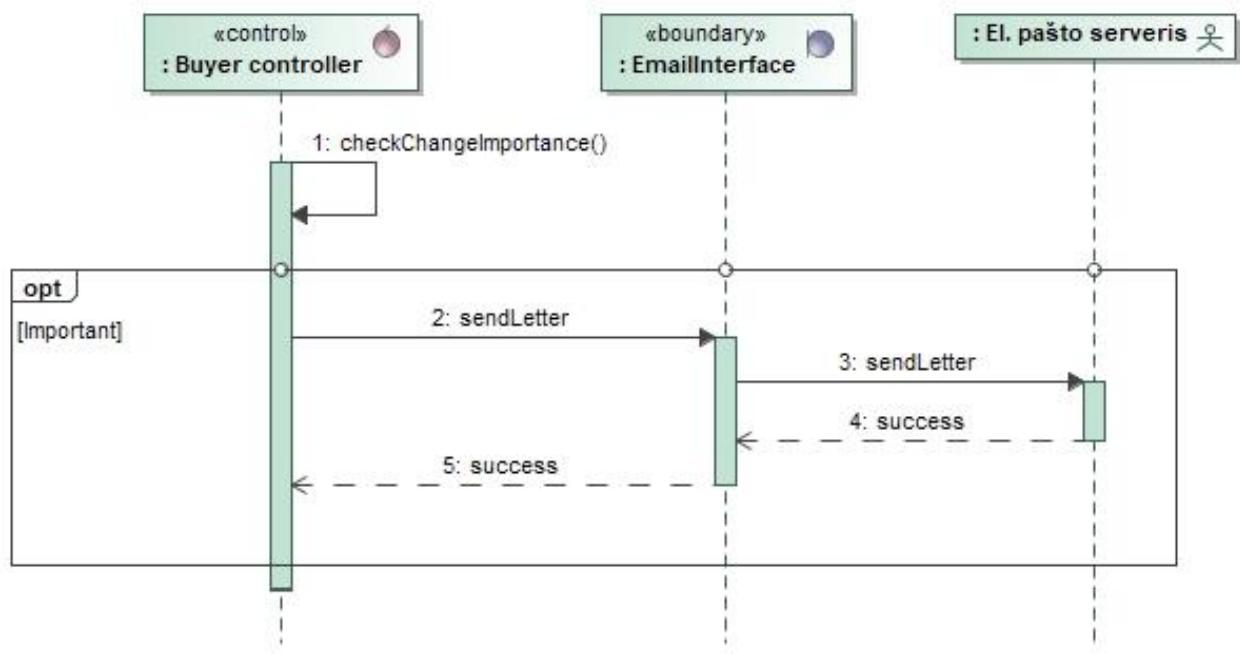
Pav. 72 Rezervuoti laiką pas mechaniku sekų diagrama



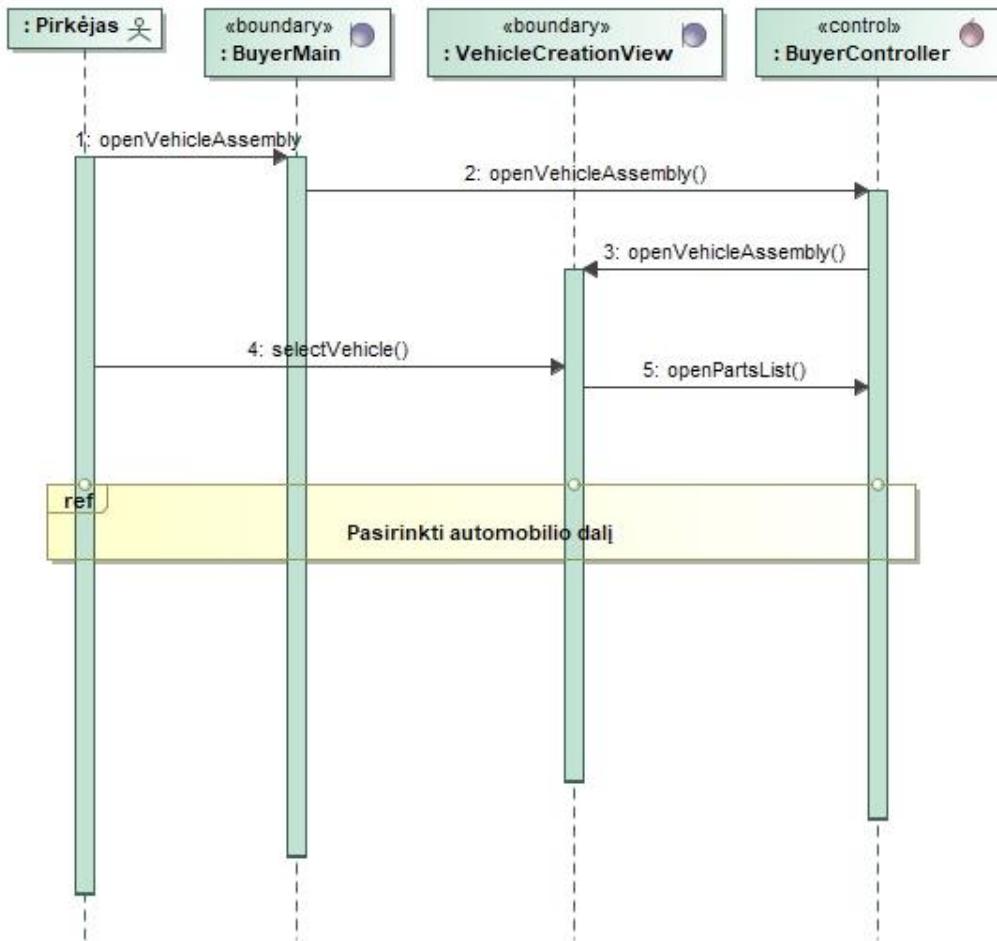
Pav. 73 Rodyti rekomenduojamas prekes sekų diagrama



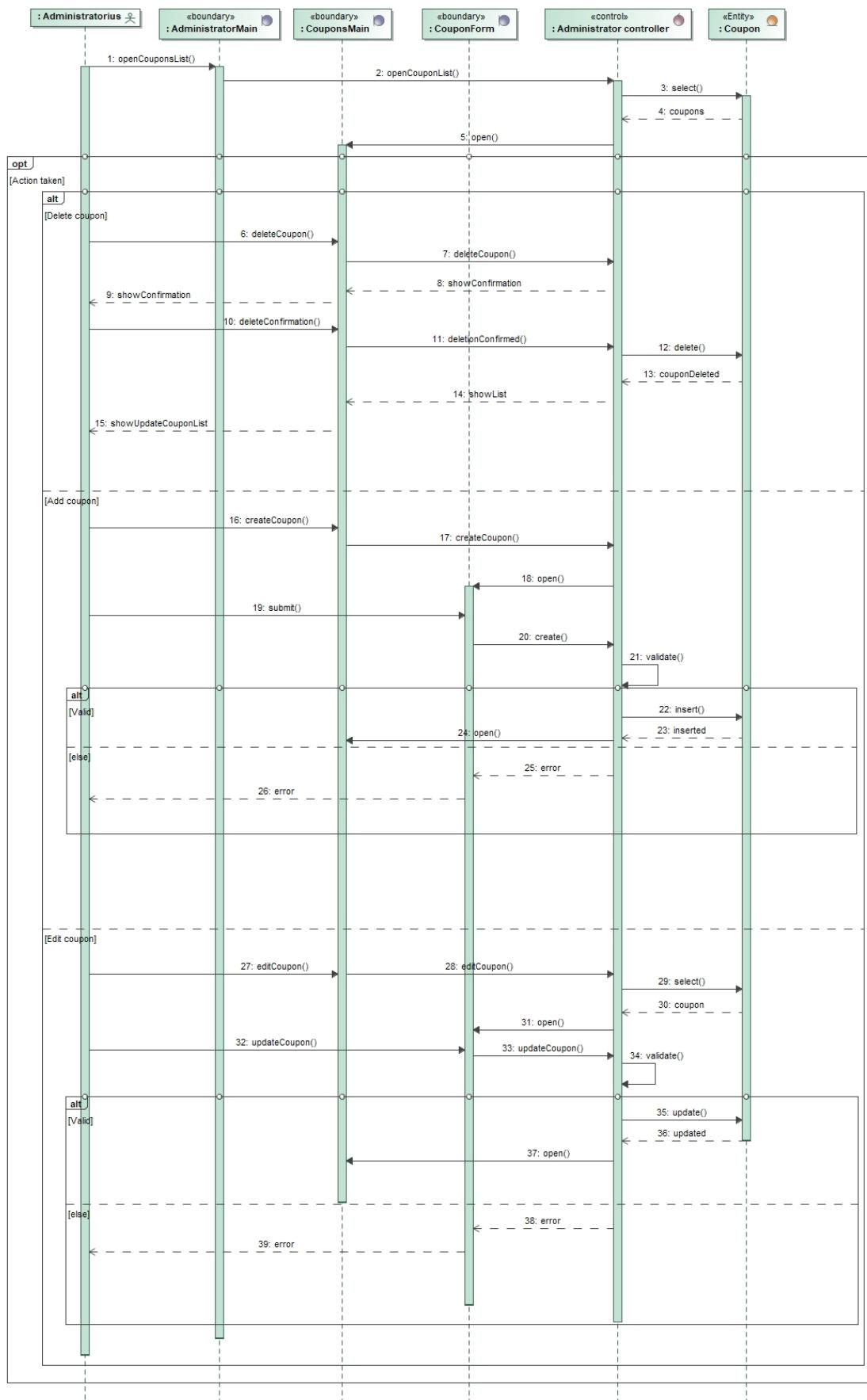
Pav. 74 Sekti siuntą sekų diagrama



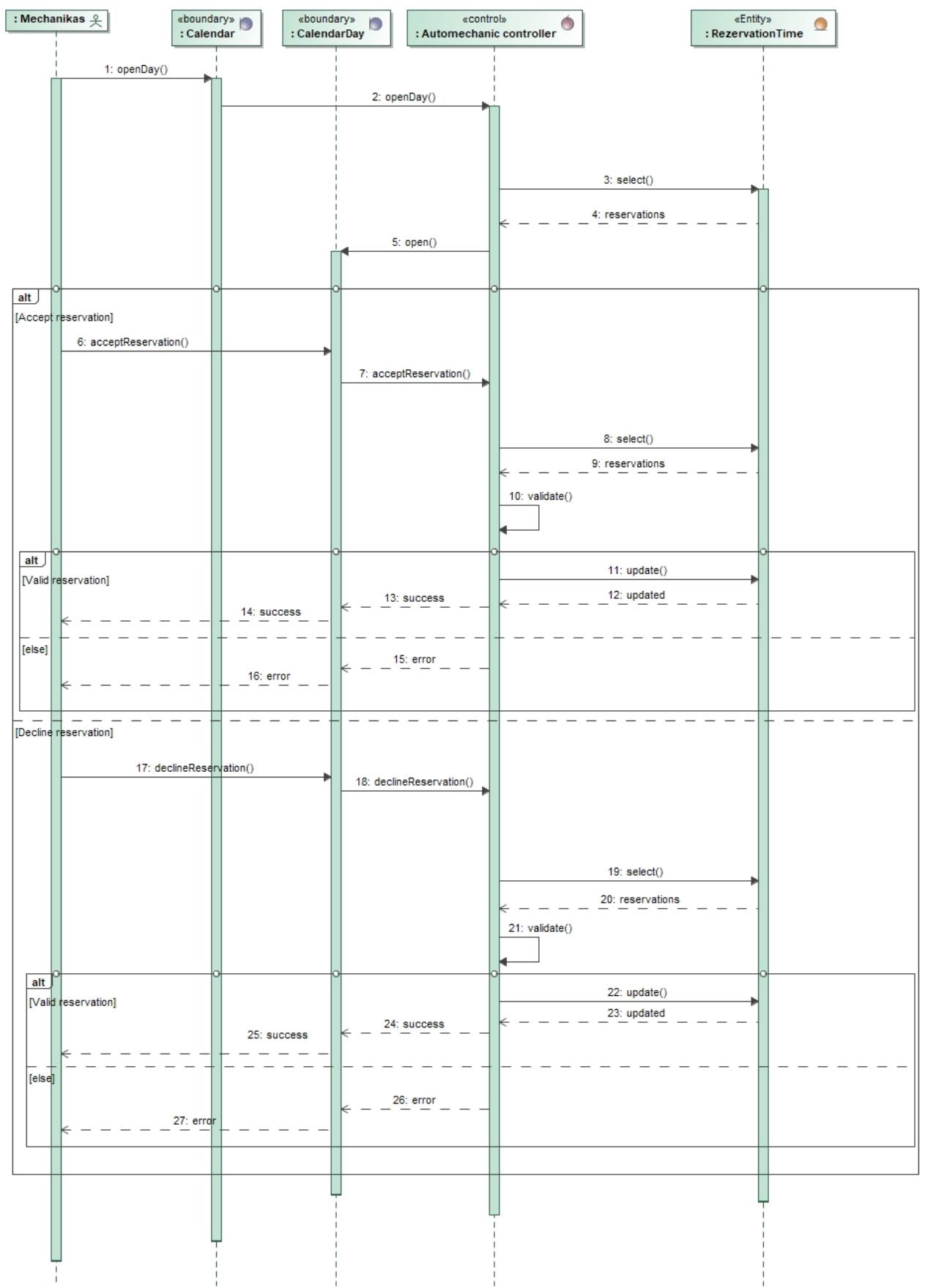
Pav. 75 Siųsti laišką apie paskyros atnaujinimus sekų diagrama



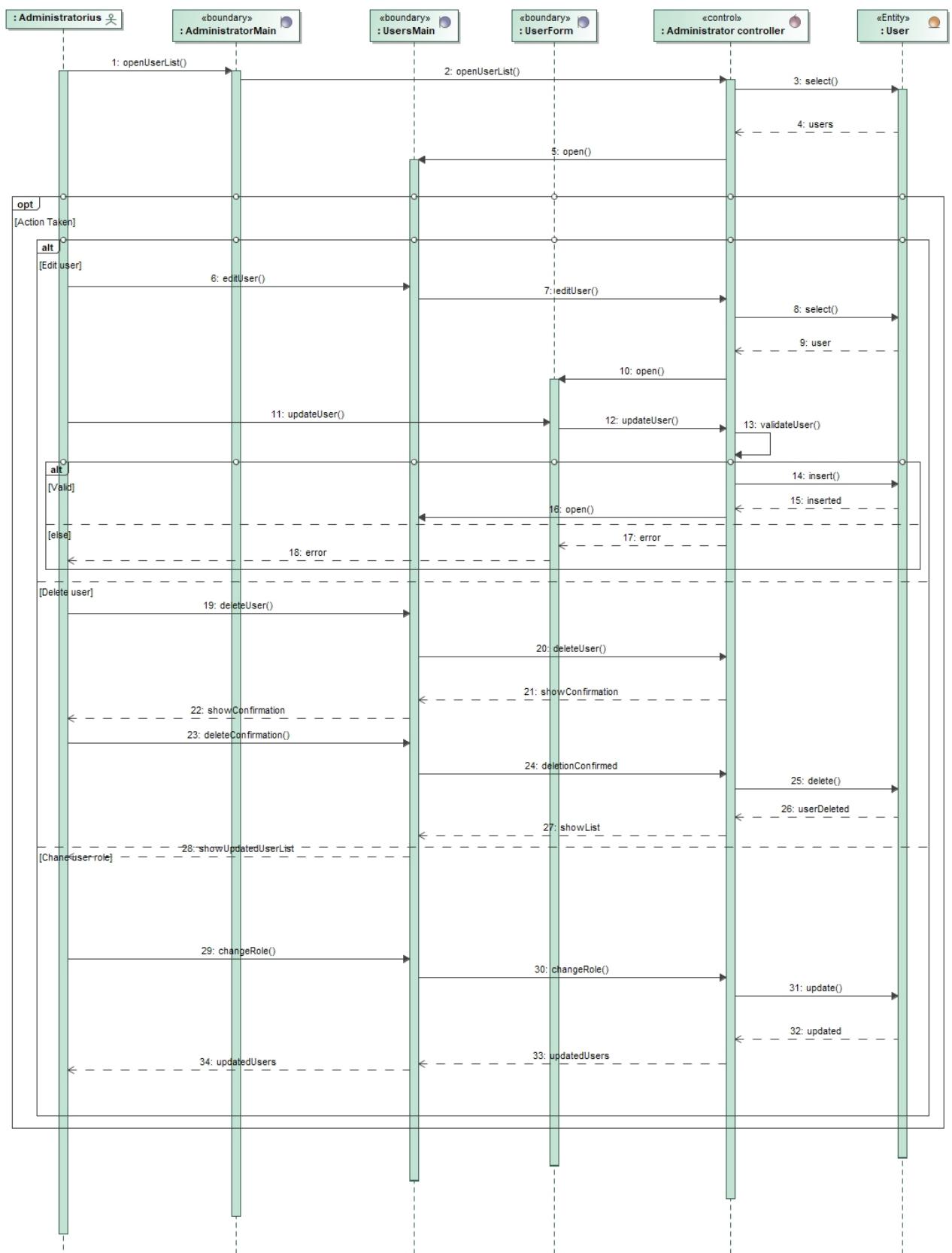
Pav. 76 Susirinkti automobilij sekų diagrama



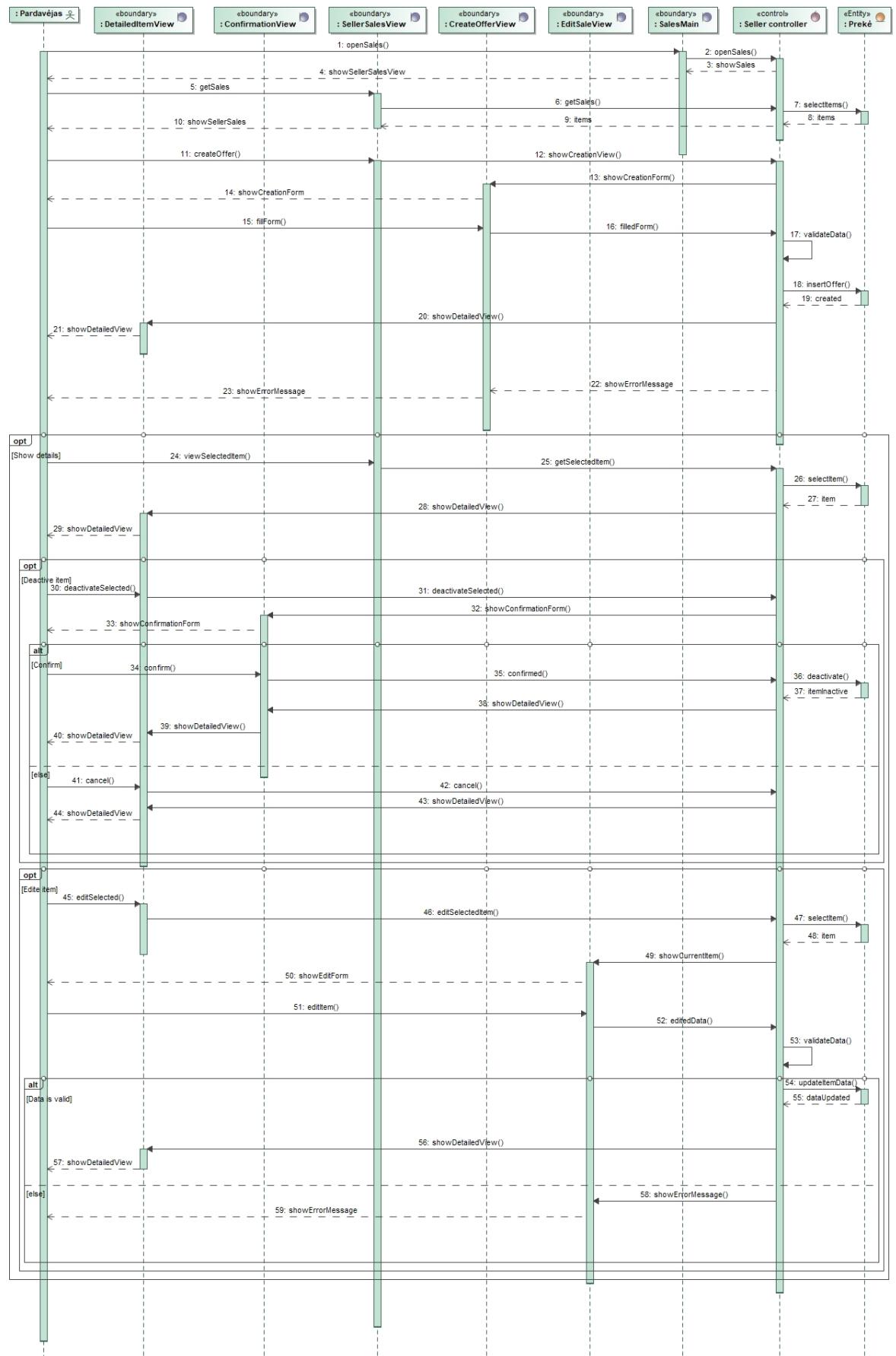
Pav. 77 Valdyti akcijų kuponus sekų diagrama



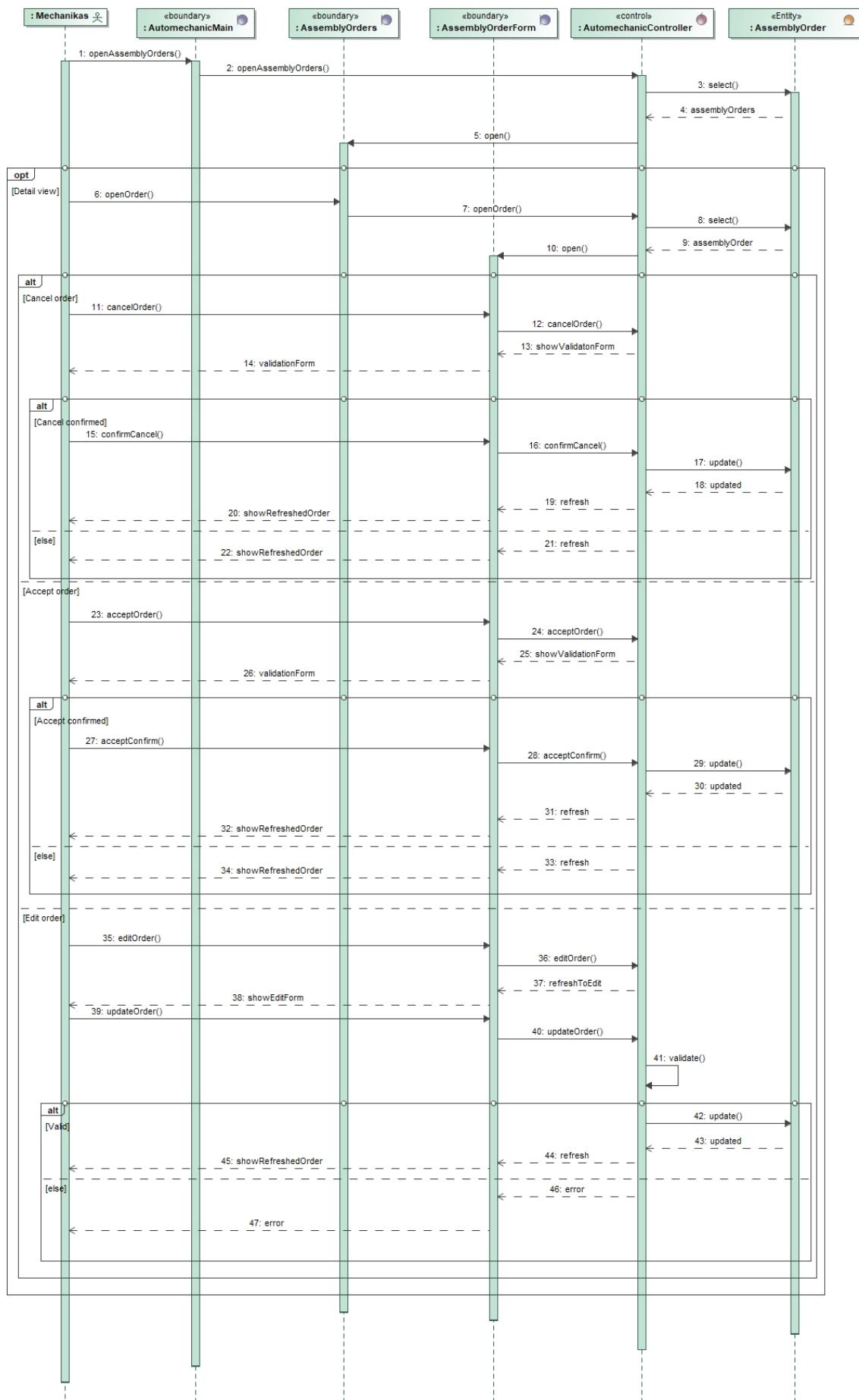
Pav. 78 Valdyti darbo valandas sekų diagrama



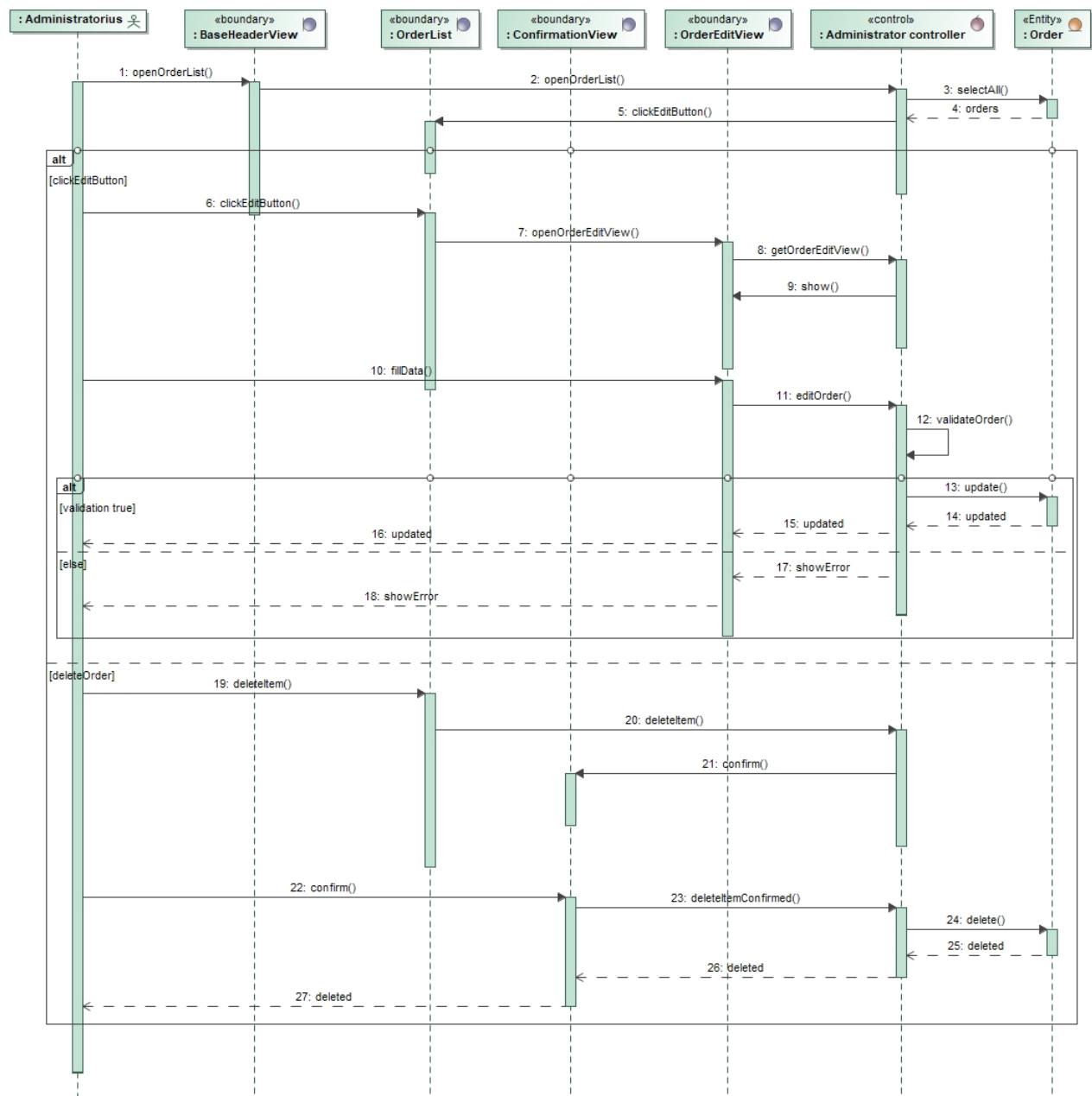
Pav. 79 Valdyti naudotojus sekų diagrama



Pav. 80 Valdyti savo prekes sekly diagrama

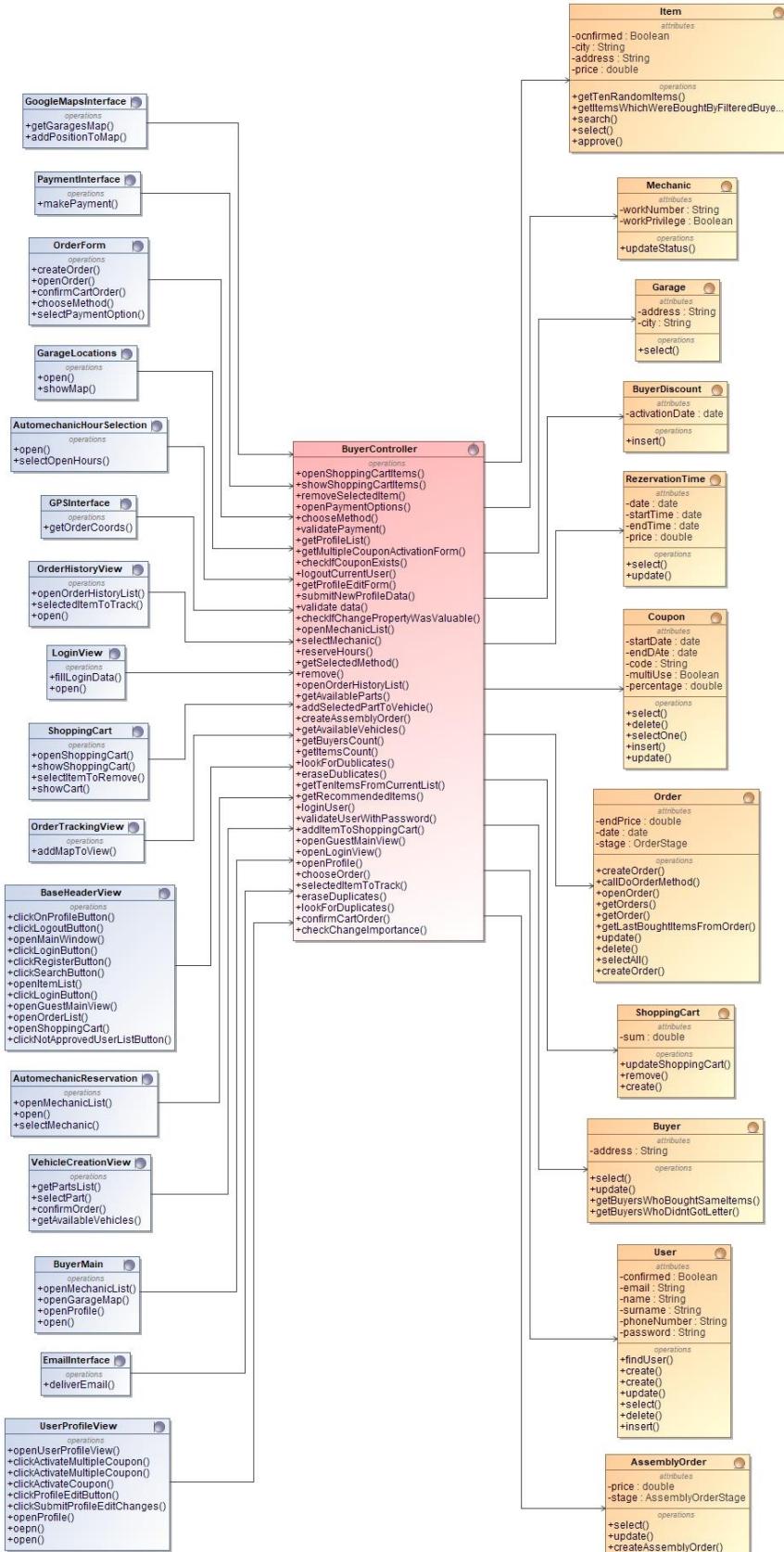


Pav. 81 Valdyti surinkimo užsakymų sekų diagrama



Pav. 82 Valdyti užsakymus sekų diagrama

4.3 Sistemos klasių diagrama

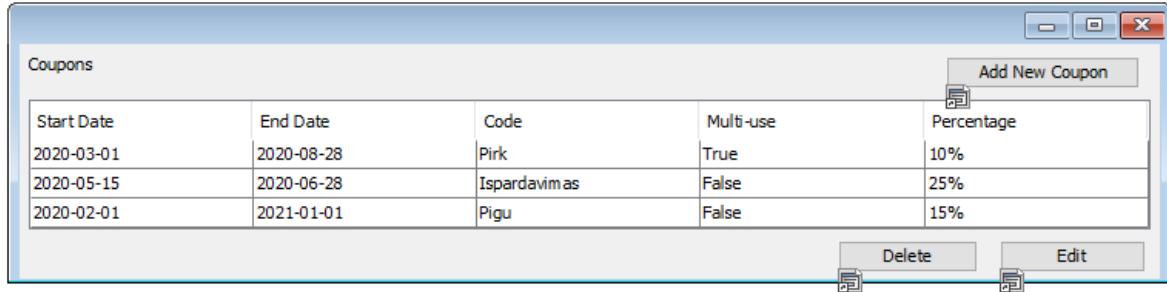


Pav. 83 Sistemos klasių diagrama

4.4 Sistemos prototipas

Realizuosime „Valdyti akcijų kuponus“ CRUD panaudojimo atvejį.

Prototipas akcijų kuponų valdymo lango buvo:



pav. 84 Realizuoto akcijų valdymo prototipo langas

Realizuotas langas:

The screenshot shows a web-based application interface. At the top left are navigation links: "Projektas", "Home", "Admin Main", and "Manage Coupons". The main area has a title "Coupons" and a green "Add New Coupon" button. Below is a table with columns: Start Date, End Date, Code, Is it multi usable?, Percentage, and Actions. The data in the table is as follows:

Start Date	End Date	Code	Is it multi usable?	Percentage	Actions
04-01-2019	05-30-2020	Rimvydo_Garazas	False	2	<button>Edit</button> <button>Delete</button>
04-15-2020	12-10-2020	798798	False	5	<button>Edit</button> <button>Delete</button>
04-01-2020	04-30-2020	Eligijus	False	1	<button>Edit</button> <button>Delete</button>
02-02-2012	03-05-2012	DEMIKO	True	420	<button>Edit</button> <button>Delete</button>
05-05-2015	05-16-2100	YEARS85	False	85	<button>Edit</button> <button>Delete</button>
09-19-2019	09-12-2019	123123	False	99	<button>Edit</button> <button>Delete</button>

pav. 85 Pagrindinis akcijų kuponų valdymo langas

Prototipas kuponų pridėjimo lango:

The screenshot shows a Windows application window titled "Create Coupon". It contains the following fields:

- Start Date: 2020-05-06
- End Date: 2020-07-20
- Code: PirkPigu
- Multi-use: False
- Percentage: 15%

At the bottom is a "Submit" button.

pav. 86 Realizuoto akcijos kūrimo prototipo langas

Realizuotas langas:

Projektas Home Admin Main Manage Coupons

Start Date

End Date

Code

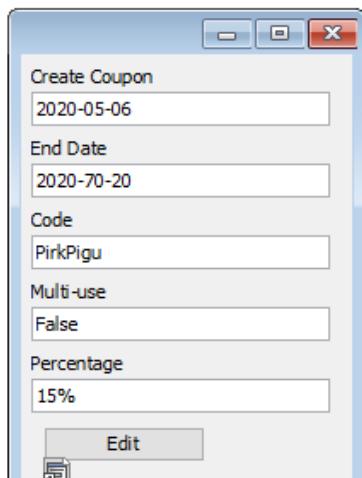
Multi use? True or False

Percentage

Submit

Pav. 87 Akcijos kupono sukūrimo langas

Prototipas kupono redagavimo lango:



pav. 88 Realizuoto akcijos redagavimo prototipo langas

Realizuotas langas:

Projektas Home Admin Main Manage Coupons

Start Date
04-01-2019

End Date
05-30-2020

Code
Rimvydo_Garazas

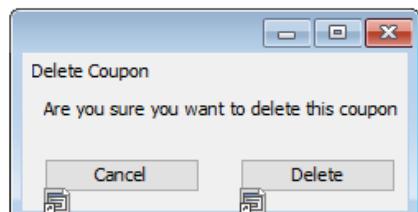
Multi use? True or False
False

Percentage
2

Submit

Pav. 89 Akcijos koregavimo langas

Prototipas kupono ištrynimo lango:



Realizuotas langas:

pav. 90 Realizuoto akcijos ištrynimo prototipo langas

Projektas Home Admin Main Manage Coupons

Coupons

Start Date	End Date	Code	Multi use?	Percentage	Actions
04-01-2019	05-30-2020	Eligijus	False	1	Edit Delete
04-15-2020	12-10-2020	DEMIKO	True	420	Edit Delete
04-01-2020	04-30-2020	YEARS85	False	85	Edit Delete
02-02-2012	03-05-2012	123123	False	99	Edit Delete

Delete Coupon

Are you sure you want to delete these Records?

This action cannot be undone.

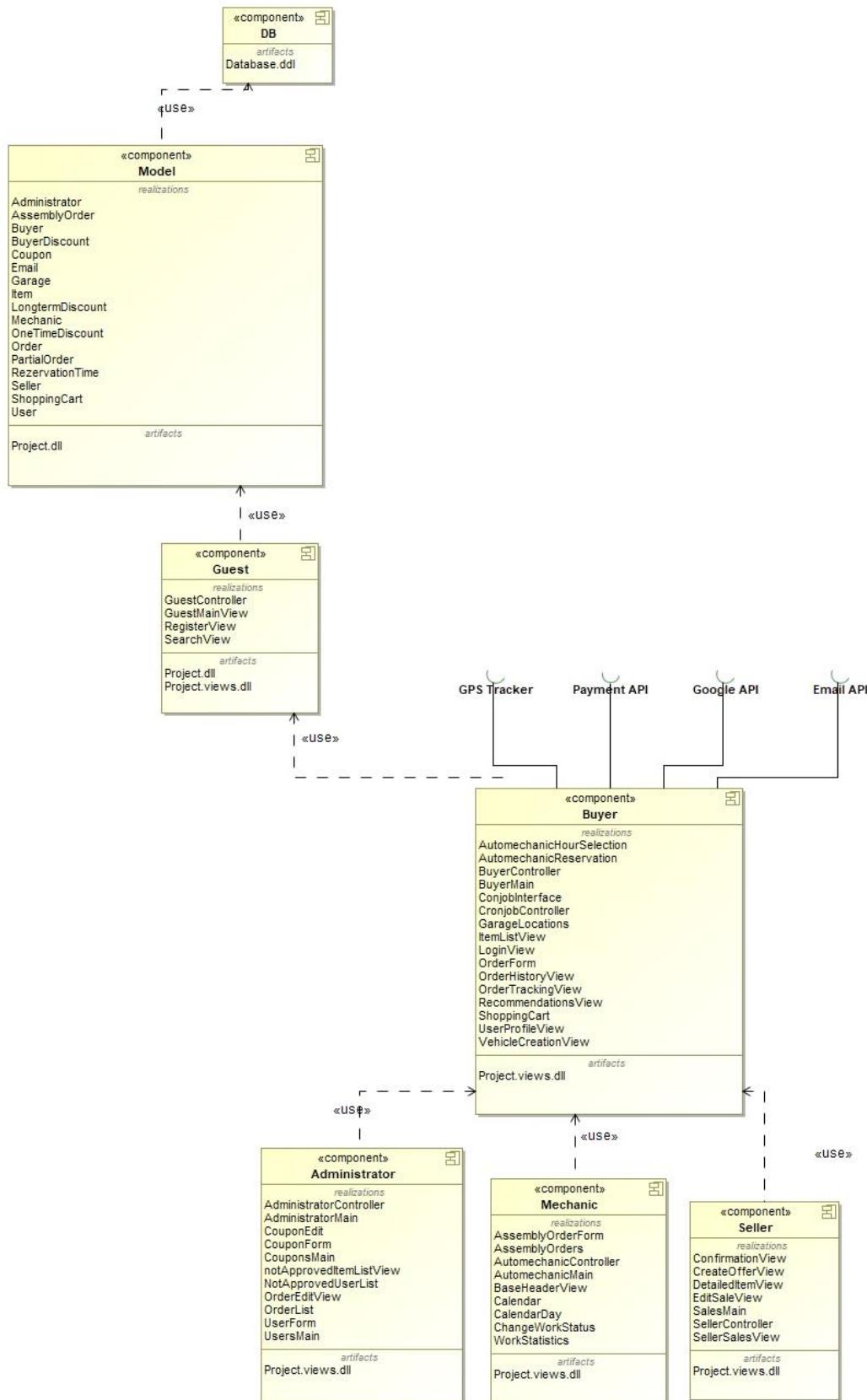
Cancel **Delete**

Pav. 91 Akcijos ištrynimo pranešimas

Iš paveikslėlių matome, kad realizacijos langai funkcionalumo atžvilgiu atitinka prototipus, bet buvo pakeisti kai kurie pavadinimai ir kai kurių elementų pozicijos.

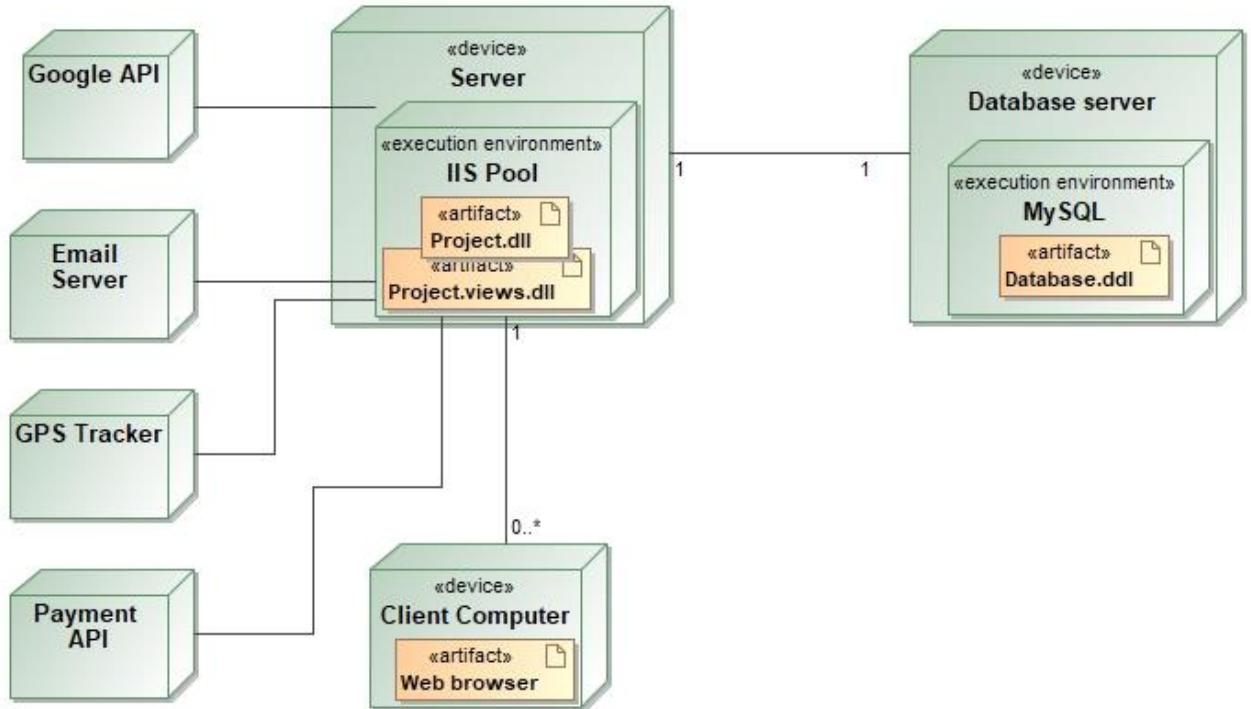
5 Sistemos realizacijos modelis

5.1 Komponentų diagrama



Pav. 92 E-Šrotas komponentų diagrama

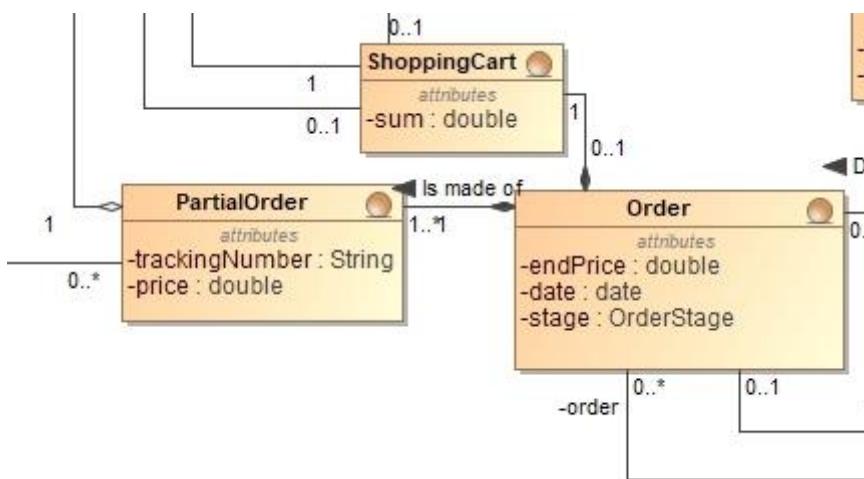
5.2 Diegimo diagrama



Pav. 93 E-Šrotas diegimo diagrama

6 Sistemos realizacija

6.1 Aggregation, Composition, Dependency ryšių atitikimas kodo realizacijoje



Pav. 90 Klasinių diagramos kompozicijos fragmentas

Kompozicija vyksta tarp **PartialOrder** ir **Order** bei **ShoppingCart** ir **Order**. Kitaip sakant, ShopingCart susideda iš Order, o Order susideda iš PartialOrder.

Order entity realizacija.

```
using ...  
  
namespace Projektas.Models  
{  
    #region Usages  
    #region Order  
    public class Order  
    {  
        #region Usages  
        #region Price  
        public string price { get; set; }  
        #endregion  
        #region IdItem  
        public string id_Item { get; set; }  
        #endregion  
        #region Name  
        public string name { get; set; }  
        #endregion  
        #region ShoppingCart  
        public ShoppingCart ShoppingCart { get; set; }  
        #endregion  
        #region PartialOrders  
        public PartialOrder[] PartialOrders { get; set; }  
        #endregion  
        #endregion  
        #region Order  
        public Order()  
        {  
        }  
        #endregion  
        #region Order  
        public Order(string price, string idItem, string name)  
        {  
            this.price = price;  
            id_Item = idItem;  
            this.name = name;  
        }  
    }  
}
```

Pav. 91 Order entity

Kaip matome, šis modelis turi ShoppingCart objektą savyje bei PartialOrders objektų masyvą. Lygiai taip kaip pavaizduota klasių diagramoje.

```
namespace Projektas.Models
{
    public class PartialOrder
    {
        public Order order { get; set; }

        public PartialOrder()
        {

        }

        public PartialOrder(Order order)
        {
            this.order = order;
        }
    }
}
```

Pav. 92 PartialOrder entity

Čia matome, kad PartialOrder turi pagrindinį Order objektą savyje.

```
namespace Projektas.Models
{
    public class ShoppingCart
    {
        public Order order { get; set; }

        public ShoppingCart()
        {

        }

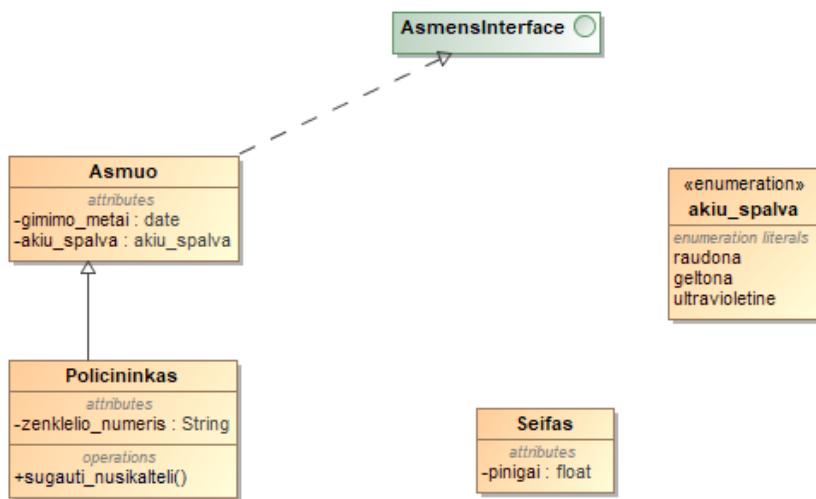
        public ShoppingCart(Order order)
        {
            this.order = order;
        }
    }
}
```

Pav. 93 Order entity

Taip pat čia matome, jog ShoppingCart entity turi savyje Order entity, tai nėra masyvas, nes pagal diagramą ryšis nėra vienas su daug.

6.2 Kodo generavimas

Kodo generavimo demonstracijai buvo sukurta nedidelė nauja klasių diagrama



Pav. 94 Klasių diagramos fragmentas

Iš jos buvo gautas kodas

Akiu_spalva.cs kodas

```
/***
 * @(#) akiu_spalva.cs
 */

namespace Generavimui
{
    public enum akiu_spalva
    {
        raudona,
        geltona,
        ultravioletine,
    }
}

~
~
~
~
```

Pav. 95 akiu spalvos sugeneruotas kodas

AsmensInterface.cs

```
/***
 * @(#) AsmensInterface.cs
 */

namespace Generavimui
{
    public interface AsmensInterface
    {
    }
}
```

Pav. 96 AsmensInterface sugeneruotas kodas

Asmuo.cs

```
/***
 * @(#) Asmuo.cs
 */

namespace Generavimui
{
    public class Asmuo : AsmensInterface
    {
        date gimimo_metai;
        akiu_spalva akiu_spalva;
    }
}
```

Pav. 97 Asmuo sugeneruotas kodas

Policininkas.cs

```
/***
 * @(#) Policininkas.cs
 */

namespace Generavimui
{
    public class Policininkas : Asmuo
    {
        String zenklelio_numeris;

        public void sugauti_nusikalteli( )
        {
        }

    }
}
```

Pav. 98 Policininkas sugeneruotas kodas

Seifas.cs

```
/***
 * @(#) Seifas.cs
 */

namespace Generavimui
{
    public class Seifas
    {
        float pinigai;
    }
}
```

Pav. 99 Seifas sugeneruota klase

Taigi mes nusprendēme pridēti prie Seifas dar kelis atributus, kurie yra:

- Integer seifo_numeris
- String aukso_luitu_gamybos_kodas

```
/***
 * @(#) Seifas.cs
 */

namespace Generavimui
{
    public class Seifas
    {
        float pinigai;
        int seifo_numeris;
        string aukso_luitu_gamybos_kodas;
    }
}
```

Pav. 100 Seifas papildytas kodas

Ir į akiu_spalva buvo pridėtos pora naujų spalvų

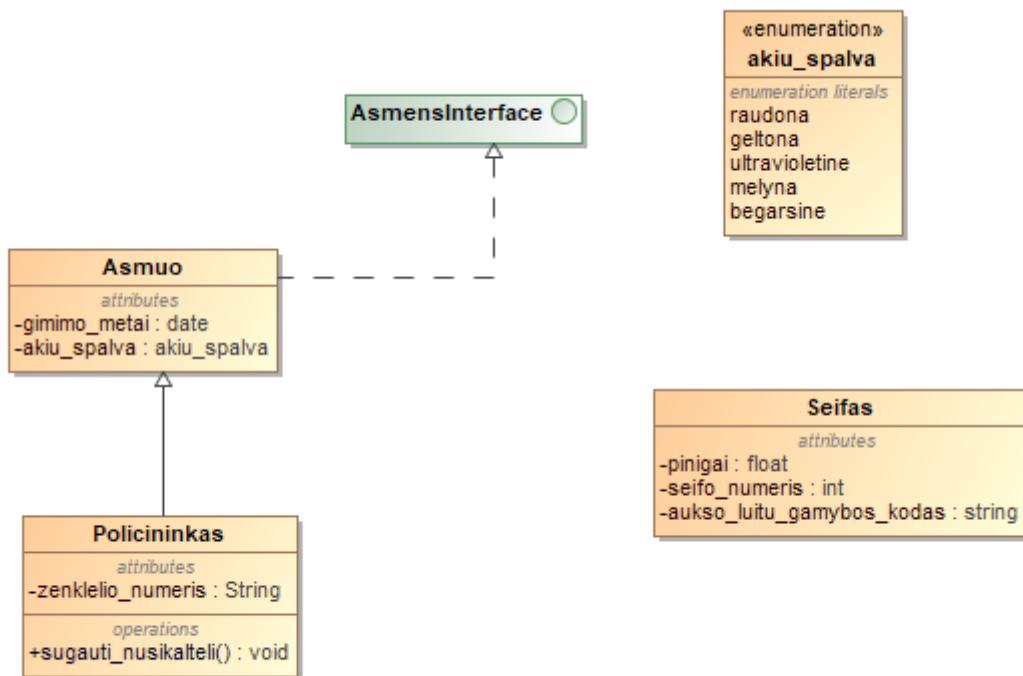
- Melyna
- Begarsine

```
/***
 * @(#) akiu_spalva.cs
 */

namespace Generavimui
{
    public enum akiu_spalva
    {
        raudona,
        geltona,
        ultravioletine,
        melyna,
        begarsine
    }
}
```

Pav. 101 Akiu spalva papildytas kodas

Taigi dabar atliekame reverse-engineering.

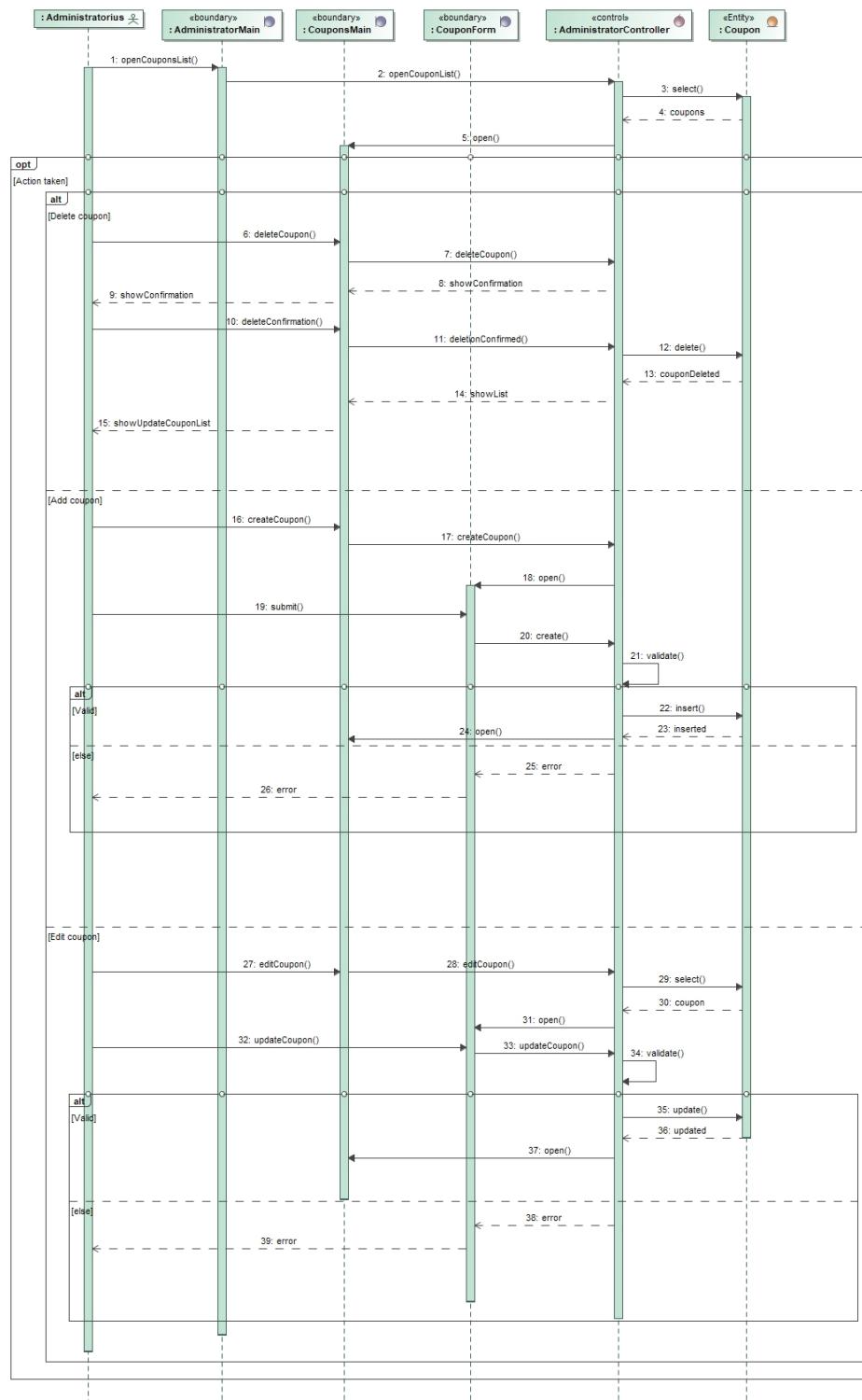


Pav. 102 reverse-engineered is kodo i klasiu diagrama

Taigi kaip matome viskas reverse-engineering atliktas teisingai, todėl galime teigti, jog viskas labai puiku.

6.3 Sekų diagramos kodo realizavimas

Taigi pasirinkome kode pristatyti šią būsenų diagramą



Pav. 103 pasirinkta sekų diagrama

Kupony langas

Coupons

 Add New Coupon

Start Date	End Date	Code	Is it multi usable?	Percentage	Actions
04/01/2019	05/30/2020	Rimvydo_Garazas	True	5	<button>Edit</button> <button>Delete</button>
02/02/2012	05/30/2020	KTU	True	20	<button>Edit</button> <button>Delete</button>
02/02/2012	05/30/2020	LABAS	True	60	<button>Edit</button> <button>Delete</button>

Pav. 104 Kuponų langas

Kuponų pridėjimo langas

Start Date

End Date

Code

Multi use? True or False

Percentage

Submit

Pav. 105 Kuponu pridejimo langas

Kuponų redagavimo langas

Start Date

02/02/2012

End Date

05/30/2020

Code

KTU

Multi use? True or False

True

Percentage

20

Submit

Pav. 106 Kupono redagavimo langas

Kupono ištrinimo langas

The screenshot shows a 'Delete Coupon' modal dialog box centered over a table of coupon records. The dialog contains the text: 'Are you sure you want to delete these Records?' and 'This action cannot be undone.' Below the text are 'Cancel' and 'Delete' buttons. The background table lists four coupon entries with columns: Start Date, End Date, Code, Multi use?, Percentage, and Actions. Each entry has 'Edit' and 'Delete' buttons.

Start Date	End Date	Code	Multi use?	Percentage	Actions
04/01/2019	05/30/2020	LABAS	True	60	Edit Delete
02/02/2012	05/30/2020				Edit Delete
02/02/2012	05/30/2020				Edit Delete

Pav. 107 Kupono istrinimo langas

Realizuotas kodas

```
using Microsoft.AspNetCore.Mvc;
using System;
using System.Collections.Generic;
using System.ComponentModel.DataAnnotations;
using System.Diagnostics;
using System.Linq;
using System.Threading.Tasks;
using Microsoft.AspNetCore.Mvc;
using Microsoft.Extensions.Logging;
using Projektas.Models;
```

```

namespace Projektas.Controllers
{
    public class AdministratorController : Controller
    {
        // GET
        public IActionResult Coupons(bool deletePrompt = false, string id = "-1" )
        {
            Coupon a = new Coupon();
            var data = a.select();
            ViewData.Model = data;
            ViewData["DeleteAlert"] = deletePrompt;
            ViewData["ID"] = id;

            return View();
        }
        public IActionResult createCoupon(string startDate, string endDate, string code, string multiUse,
            string percentage)
        {
            DateTime start;
            DateTime end;
            try
            {
                start = DateTime.Parse(startDate);
                end = DateTime.Parse(endDate);
                if (!validate(code, multiUse, percentage))
                {
                    throw new ValidationException();
                }
            }
            catch
            {
                return RedirectToAction("AddCoupon", new { validationError = "show" });
            }
            Coupon a = new Coupon();
            a.insert(start.ToString("yyyy/MM/dd"), end.ToString("yyyy/MM/dd"), code, multiUse,
            percentage);
            return RedirectToAction("Coupons", new { deletePrompt = false, id = "-1" });
        }
        // If validation is good, it will return true
        bool validate(string code, string multiUse, string percentage)
        {
            if (String.IsNullOrEmpty(code))
            {
                return false;
            }
            if (String.IsNullOrEmpty(multiUse))
            {
                return false;
            }
            if (String.IsNullOrEmpty(percentage))
            {
                return false;
            }
            var isNumeric = int.TryParse(percentage, out _);
            if (!isNumeric)
            {
                return false;
            }
            return true;
        }
        public IActionResult updateCoupon(string id, string startDate, string endDate, string code,
            string multiUse, string percentage)
        {
            DateTime start;

```

```

        DateTime end;
    try
    {
        start = DateTime.Parse(startDate);
        end = DateTime.Parse(endDate);
        if (!validate(code, multiUse, percentage))
        {
            throw new ValidationException();
        }
    }
    catch
    {
        return RedirectToAction("EditCoupon", new { validationError = "show", id = id });
    }
    Coupon a = new Coupon();
    a.update(int.Parse(id), start.ToString("yyyy/MM/dd"), end.ToString("yyyy/MM/dd"), code,
    multiUse, percentage);
    return RedirectToAction("Coupons", new { deletePrompt = false, id });
}

public IActionResult deleteCoupon(string id)
{
    Coupon a = new Coupon();
    int useId = int.Parse(id);
    a.delete(useId);
    return RedirectToAction("Coupons", new { deletePrompt = false, id });
}

public IActionResult showConfirmation(int index)
{
    Coupon a = new Coupon();
    var data = a.select();
    ViewData.Model = data;
    return View("Coupons");
}

public IActionResult EditCoupon(string id = "-1", string validationError = "not show")
{
    Coupon a = new Coupon();
    int intId = int.Parse(id);
    var data = a.selectById(intId);

    var view = View("CouponForm");
    view.ViewData.Model = data;
    // view.ViewData["startDate"] = data.startDate;
    view.ViewData["ID"] = int.Parse(id);
    view.ViewData["validationError"] = validationError;
    return view;
}
// GET
public IActionResult AddCoupon(string validationError = "notShow")
{
    ViewData.Model = validationError;
    var view = View("CouponForm");
    view.ViewData["validationError"] = validationError;
    view.ViewData["Type"] = "ADD";
    return view;
}

// GET
public IActionResult AdministratorMain()
{
    return View();
}

```

```
    }  
}
```

Tada sugeneruotas Coupon.cs modelio entity

```
using System;  
using System.Collections.Generic;  
using System.Configuration;  
using MySql.Data.MySqlClient;  
  
namespace Projektas.Models  
{  
    public class Coupon  
    {  
        public static string cs = "server=auth-  
db240.hostinger.com;userid=u429721638_isp2;password=LEAKED;database=u429721638_isp2";  
        public string startDate { get; set; }  
        public string endDate { get; set; }  
        public string code { get; set; }  
        public string multiUse { get; set; }  
        public string percentage { get; set; }  
        public string id_Discount { get; set; }  
        public string fk_Administrator_Id { get; set; }  
  
        public Coupon()  
        {  
  
        }  
        public Coupon(string startDate, string endDate, string code, string multiUse, string  
percentage, string idDiscount, string fkAdministratorId)  
        {  
            this.startDate = startDate;  
            this.endDate = endDate;  
            this.code = code;  
            this.multiUse = multiUse;  
            this.percentage = percentage;  
            id_Discount = idDiscount;  
            fk_Administrator_Id = fkAdministratorId;  
        }  
  
        public List<Coupon> select()  
        {  
            List<Coupon> selectedList = new List<Coupon>();  
  
            using var con = new MySqlConnection(Coupon.cs);  
            con.Open();  
  
            string sql = "SELECT * FROM Discount";  
            MySqlCommand cmd = new MySqlCommand(sql, con);  
            MySqlDataReader rdr = cmd.ExecuteReader();  
  
            while (rdr.Read())  
            {  
                Coupon tempCoupon = new Coupon(rdr.GetString(0), rdr.GetString(1),  
rdr.GetString(2), rdr.GetString(3), rdr.GetString(4), rdr.GetString(5), rdr.GetString(6));  
                selectedList.Add(tempCoupon);  
            }  
            rdr.Close();  
  
            return selectedList;  
        }  
  
        public Coupon selectById(int id)  
        {  
            Coupon tempCoupon = new Coupon();
```

```

        using var con = new MySqlConnection(Coupon.cs);
        con.Open();

        string sql = String.Format("SELECT * FROM Discount WHERE id_Discount={0};", id);
        MySqlCommand cmd = new MySqlCommand(sql, con);
        MySqlDataReader rdr = cmd.ExecuteReader();

        while (rdr.Read())
        {
            tempCoupon = new Coupon(rdr.GetString(0), rdr.GetString(1), rdr.GetString(2),
rdr.GetString(3), rdr.GetString(4), rdr.GetString(5), rdr.GetString(6));
        }
        rdr.Close();

        return tempCoupon;
    }

    public Coupon selectOne(string couponName)
    {
        Coupon tempCoupon = new Coupon();
        using var con = new MySqlConnection(Coupon.cs);
        con.Open();

        string sql = String.Format("SELECT * FROM Discount WHERE code='{0}' AND multiUse =
1;", couponName);
        MySqlCommand cmd = new MySqlCommand(sql, con);
        MySqlDataReader rdr = cmd.ExecuteReader();

        while (rdr.Read())
        {
            tempCoupon = new Coupon(rdr.GetString(0), rdr.GetString(1), rdr.GetString(2),
rdr.GetString(3), rdr.GetString(4), rdr.GetString(5), rdr.GetString(6));
        }
        rdr.Close();

        return tempCoupon;
    }

    public void insert(string startDate, string endDate, string code, string multiUse, string
percentage)
    {
        using var con = new MySqlConnection(Coupon.cs);
        con.Open();

        string sql = String.Format(
            "INSERT INTO Discount (startDate, endDate, code, multiUse, percentage,
fk_Administrator_id) VALUES ('{0}', '{1}', '{2}', '{3}', '{4}', '{5}');", startDate, endDate,
code, multiUse, percentage, 1);
        MySqlCommand cmd = new MySqlCommand(sql, con);
        cmd.ExecuteNonQuery();
        con.Close();
    }

    public void update(int id, string startDate, string endDate, string code, string
multiUse, string percentage)
    {
        using var con = new MySqlConnection(Coupon.cs);
        con.Open();

        string sql = String.Format("UPDATE Discount SET startDate='{0}', endDate='{1}',
code='{2}', multiUse='{3}', percentage='{4}' WHERE id_Discount={5};", startDate, endDate, code,
multiUse, percentage, id);
        MySqlCommand cmd = new MySqlCommand(sql, con);
        cmd.ExecuteNonQuery();
        con.Close();
    }

```

```

        }

        public void delete(int id)
        {
            using var con = new MySqlConnection(Coupon.cs);
            con.Open();

            string sql = String.Format(
                "DELETE FROM Discount WHERE id_Discount={0};", id);
            MySqlCommand cmd = new MySqlCommand(sql, con);
            cmd.ExecuteNonQuery();

            con.Close();
        }
    }
}

```

Sukurti Views

AdministratorMain

```

@{
    ViewData["Title"] = "Administrator Main Menu";
}

```

Administrator Main Menu

```

<a class="nav-link text-dark" asp-area="" asp-controller="Administrator" asp-
action="Coupons"><button type="button" class="btn btn-primary">Manage Coupons</button></a>

```

Coupons forma

```

@{
    ViewData["Title"] = "Coupons";
}

@if (ViewData["validationError"].ToString() == "show")
{
    <div class="alert alert-danger" role="alert">
        Validation error
    </div>
}

<form method="POST">
    <div class="form-group">
        <label>Start Date</label>
        <input class="form-control" name="startDate"
value="@((ViewData.ModelExplorer.Properties.Count() > 1 ?
Convert.ToDateTime(Model.startDate).ToString("MM/dd/yyyy") : "")" placeholder="Start Date">
    </div>
    <div class="form-group">
        <label>End Date</label>
        <input class="form-control" name="endDate" value="@((ViewData.ModelExplorer.Properties.Count() >
1 ? Convert.ToDateTime(Model.endDate).ToString("MM/dd/yyyy") : "")" placeholder="End Date">
    </div>
    <div class="form-group">
        <label>Code</label>
        <input class="form-control" name="code" value="@((ViewData.ModelExplorer.Properties.Count() >
1 ? Model.code : "")" placeholder="Code">
    </div>
    <div class="form-group">
        <label>Multi use? True or False</label>
    </div>

```

```
<input class="form-control" name="multiUse"
value="@ ViewData.ModelExplorer.Properties.Count() > 1 ? Model.multiUse : ""> placeholder="True
or False">
</div>
<div class="form-group">
    <label>Percentage</label>
    <input class="form-control" name="percentage"
value="@ ViewData.ModelExplorer.Properties.Count() > 1 ? Model.percentage : ""> placeholder="Percentage">
</div>
@if (ViewData["Type"] == (String)"ADD")
{
    <button type="submit" name="button" value="100" class="btn btn-primary" asp-
action="createCoupon">Submit</button>
}
else // edit
{
    <button type="submit" name="id" value="@Model.id_Discount" class="btn btn-primary" asp-
action="updateCoupon">Submit</button>
}
</form>
```

Coupons view

```

        <td>
            @*<a href="#editEmployeeModal" class="edit" data-toggle="modal"><i
            class="material-icons" data-toggle="tooltip" title="Edit">@<#xE254;></i></a>*@
            <form method="POST">
                <button type="submit" name="id" value="@row.id_Discount"
            class="btn btn-primary" asp-action="EditCoupon">Edit</button>
                <button type="submit" name="id" value="@row.id_Discount"
            class="btn btn-primary" asp-action="promptCouponDeletion">Delete</button>
            </form>
        </td>
    </tr>
}

<!-- Delete Modal HTML -->
<div id="deleteEmployeeModal" class="modal fade show">
    <div class="modal-dialog">
        <div class="modal-content">
            <form>
                <div class="modal-header">
                    <h4 class="modal-title">Delete Coupon</h4>
                    <button type="button" class="close" data-dismiss="modal"
aria-hidden="true">&times;</button>
                </div>
                <div class="modal-body">
                    <p>Are you sure you want to delete these Records?</p>
                    <p class="text-warning"><small>This action cannot be
undone.</small></p>
                </div>
                <div class="modal-footer">
                    <input type="button" class="btn btn-default" data-
dismiss="modal" value="Cancel">
                    <button type="submit" name="id" value="@ ViewData["ID"]" asp-
action="deleteCoupon" class="btn btn-danger">Delete</button>
                </div>
            </form>
        </div>
    </div>
</div>

</tbody>
</table>
</div>
</div>

@section Scripts {
    @if (ViewData.ContainsKey("DeleteAlert") && (bool)ViewData["DeleteAlert"])
    {
        <script type="text/javascript">
            $('#deleteEmployeeModal').modal('show');
        </script>
    }
}

```

Realizacija įvykdėta teisingai pagal sekų diagramos, projektas buvo realizuotas pagal sekų diagramas.