



Kauno technologijos universitetas

Informatikos fakultetas

Ekstremalaus vairavimo simulatorius

Baigiamasis bakalauro projektas

Artūras Latvys

Projekto autorius

Prof. Egidijus Kazanavičius

Vadovas

Kaunas, 2026



Kauno technologijos universitetas

Informatikos fakultetas

Ekstremalaus vairavimo simulatorius

Baigiamasis bakalauro projektas

Informatikos inžinerija (6121BX014)

Artūras Latvys

Projekto autorius

Prof. Egidijus Kazanavičius

Vadovas

Jaun. asist. Algirdas Dobrovolskis

Recenzentas

Kaunas, 2026



Kauno technologijos universitetas

Informatikos fakultetas

Artūras Latvys

Ekstremalaus vairavimo simulatorius

Akademinio sąžiningumo deklaracija

Patvirtinu, kad mano, Artūro Latvio, baigiamasis projektas tema „Ekstremalaus vairavimo simulatorius“ yra parašytas visiškai savarankiškai ir visi pateikti duomenys ar tyrimų rezultatai yra teisingi ir gauti sąžiningai. Šiame darbe nei viena dalis nėra plagijuota nuo jokių spausdintinių ar internetinių šaltinių, visos kitų šaltinių tiesioginės ir netiesioginės citatos nurodytos literatūros nuorodose. Įstatymų nenumatytų piniginių sumų už šį darbą niekam nesu mokėjęs.

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

(vardą ir pavardę įrašyti ranka)

(parašas)

**1. Informatikos
fakultetas,***Kompiuterių katedra*

2026 m. vasario 6 d.

BAKALAURO BAIGIAMOJO PROJEKTO UŽDUOTIS**1. Bakalauro baigiamojo projekto tema:**

Ekstremalaus vairavimo simulatorius

2. Bakalauro baigiamojo projekto tikslas:

Sukurti sistemą, kurios pagalba galima analizuoti žmogaus reakciją, poveikį jam ir jo priimamiems sprendimams.

3. Bakalauro baigiamojo projekto uždaviniai:

Sukurti sistemą, kurios pagalba galima nuskaityti žmogaus duomenis.

Surinktus duomenis perduoti į duomenų surinkimo ir pateikimo sistemą.

Suderinti sistemą su išorinėmis programinėmis įrangomis, kuriose bus vykdomi vairavimo scenarijai.

Sukurti scenarijus ir ištestuoti sistemos veikimą.

4. Bakalauro baigiamojo projekto pateikimo terminai:

Vadovui (8 d. d. prieš gynimą katedroje): 2026 m. gegužės __ d.

Recenzentui (6 d. d. prieš gynimą katedroje): 2026 m. gegužės __ d.

Komisijai (3 d. d. prieš gynimą katedroje): 2026 m. gegužės __ d.

5. Funkciniai reikalavimai kuriamam objektui arba sistemai:

Sistema turi rinkti vairuotojo valdymo telemetrijos duomenis;

Sistema turi realiu laiku rinkti biometrinius duomenis;

Sistema turi perduoti surinktus duomenis belaidžiu ryšiu;

Sistema turi saugoti surinktus duomenis numatytoje duomenų bazėje;

Sistema turi užtikrinti nuolatinį duomenų rinkimą;

Sistema turi vizualizuoti gautus duomenis.

6. Nefunkciniai reikalavimai kuriamam objektui arba sistemai:Duomenys turi būti renkami laiko intervalais, kurie yra ne >0.1 sekundės;

Sistema turi palaikyti pastovų, nepertraukiamą duomenų srautą simuliacijos metu;

Duomenys turi būti išsaugomi duomenų bazėje be naudotojo įsikišimo;

Duomenų perdavimas turi vykti saugiu ir patikimu ryšio protokolu;

Sistema neturi kelti pavojaus naudotojui ar kitiems asmenims;

Sistema turi būti lengvai integruojama su fizine simulatoriaus dalimi;
Sistemos jutikliai negali trukdyti naudotojui vykdyti simuliacijos scenarijus.

7. Kiti reikalavimai (operacinei sistemai, programinei įrangai, techninei įrangai, testavimui):

Sistema turi veikti Windows arba Linux operacinėje sistemoje;
Operacinė sistema turi palaikyti internetinio tinklo ryšį;
Sistema turi palaikyti simuliacijos programinę įrangą, kurioje bus vykdomi scenarijai;
Sistema turi turėti galimybę palaikyti internetinį ryšį;
Sistema turi būti kompaktiška ir lengvai pritaikoma;
Sistemos komponentai turi būti lengvai prieinami ir pakeičiami;
Sistema turi būti testuojama imituojant realius vairavimo scenarijus;
Testavimo metu turi būti patikrintas duomenų vientisumas, tikslumas;
Testavimo metu turi būti užtikrinta, jog sistema gali veikti be kritinių klaidų.

Vadovas: Prof. Egidijus Kazanavičius

(vardas, pavardė, parašas)

Data: 2026 m. vasario 6 d.

(suderinimo data)

Studentas: Artūras Latvys

(vardas, pavardė, parašas)

Data: 2026 m. vasario 6 d.

(suderinimo data)

Autoriaus Latvys, Artūras. Ekstremalaus vairavimo simulatorius. Bakalauro krypties studijų baigiamasis projektas vadovas prof. Egidijus Kazanavičius; Kauno technologijos universitetas, informatikos fakultetas.

Studijų kryptis ir sritis (studijų krypčių grupė): Informatikos inžinerija.

Reikšminiai žodžiai: vairavimo, simulatorius.

Kaunas, 2026. 67 p.

Santrauka

Šiame darbe vykdoma analizė apie ekstremalaus vairavimo simulatorių reikšmingumą. Planuojamas prototipas lyginamas su esamais konkurentais. Aprašomas prototipo įgyvendinimas pagal UML tipo lenteles.

Įgyvendintas prototipas yra patikrinamas pagal funkcinis ir nefunkcinius reikalavimus. Aprašomas naudotojo vadovas. Aprašomos išvados apie tai, ar prototipas buvo sėkmingai įgyvendintas ir ar prototipas gali būti vystomas toliau.

Latvys, Artūras. Extreme driving simulator. Bachelor's Professional Studies, supervisor prof. Kazanavičius Egidijus; Faculty of Informatics, Kaunas University of Technology.

Study field and area (study field group): Computer Engineering.

Keywords: driving, simulator.

Kaunas, 2026. 67 pages.

Summary

This paper analyzes the significance of extreme driving simulators. The planned prototype features are compared with existing competitors on commercial market. The implementation of the prototype is shown using UML tables.

The implemented prototype is tested against functional and non-functional requirements. A user manual is created for the prototype. Conclusions are described regarding the success of the prototype's implementation and whether there is use in further prototype development.

Turinys

Lentelių sąrašas	9
Paveikslų sąrašas	10
Santrumpų ir terminų sąrašas	11
Įvadas.....	12
1. Darbo tikslas	13
1.1. Darbo uždaviniai	13
2. Problemos analizė.....	14
2.1. Projekto vizija.....	15
2.2. Konkurentų analizė.....	16
2.3. Programinės įrangos analizė.....	18
2.3.1. Duomenų bazių variantų palyginimas	18
2.3.2. Komunikacijos protokolų variantų palyginimas	18
2.3.3. Simuliacijos programinė įranga.....	19
2.3.4. Akių stebėjimo programinė įranga	20
2.4. Fizinės įrangos analizė	21
2.4.1. Mikrovaldiklių variantų palyginimas	21
2.4.2. Širdies veiklos matavimo sensorių palyginimas.....	21
2.4.3. Streso matavimo sensorių palyginimas	22
2.5. Ekstremalaus vairavimo simulatoriaus komponentų analizės išvados.....	23
3. Ekstremalaus vairavimo simulatoriaus projektavimas	24
3.1. Ekstremalaus vairavimo simulatoriaus funkciniai reikalavimai	24
3.1.1. Ekstremalaus vairavimo simulatoriaus panaudojimo atvejai	25
3.1.2. Ekstremalaus vairavimo simulatoriaus nefunkciniai reikalavimai.....	32
4. Ekstremalaus vairavimo simulatoriaus programinės įrangos projektas	33
4.1.1. Ekstremalaus vairavimo simulatoriaus programinės įrangos architektūra.....	33
4.1.2. Ekstremalaus vairavimo simulatoriaus programinio kodo sandara.....	34
4.1.3. Ekstremalaus vairavimo simulatoriaus programinės įrangos elgsena.....	37
4.1.4. Ekstremalaus vairavimo simulatoriaus naudotojo sąsajos projektas.....	44
4.1.5. Ekstremalaus vairavimo simulatoriaus techninės įrangos projektas	46
5. Ekstremalaus vairavimo simulatoriaus prototipo realizacija ir testavimas	48
5.1. Ekstremalaus vairavimo simulatoriaus prototipo realizacija ir testavimas	48
5.1.1. Ekstremalaus vairavimo simulatoriaus techninės įrangos projektas	48
5.2. Ekstremalaus vairavimo simulatoriaus prototipo testavimas	50
5.2.1. Ekstremalaus vairavimo simulatoriaus prototipo testavimo planas	50
5.2.2. Ekstremalaus vairavimo simulatoriaus prototipo testavimo planas	50
6. Ekstremalaus vairavimo simulatoriaus dokumentacija	59
6.1. Ekstremalaus vairavimo simulatoriaus diegimo vadovas	59
6.2. Ekstremalaus vairavimo simulatoriaus naudotojo vadovas.....	59
6.3. Ekstremalaus vairavimo simulatoriaus stebėjimo įrankio naudotojo vadovas.....	60
6.4. Ekstremalaus vairavimo simulatoriaus analizės įrankio naudotojo vadovas	63
Išvados	65
Literatūros sąrašas	66

Lentelių sąrašas

1 lentelė. Konkurentų palyginimo kriterijai	17
2 lentelė. Duomenų bazių palyginimas pagal iškeltus kriterijus	18
3 lentelė. Komunikacijos protokolų palyginimas	19
4 lentelė. Simuliacijos programų palyginimo lentelė.....	20
5 lentelė. Akių stebėjimo programinės įrangos palyginimas	20
6 lentelė. Mikrovaldiklių palyginimas	21
7 lentelė. Širdies veiklos matavimo sensorių palyginimas.....	22
8 lentelė. Streso matavimo sensorių palyginimas	22
9 lentelė. Panaudojimo atvejo V.1 aprašymas	25
10 lentelė. Panaudojimo atvejo V.2 aprašymas	25
11 lentelė. Panaudojimo atvejo V.3 aprašymas	26
12 lentelė. Panaudojimo atvejo SP.1 aprašymas	27
13 lentelė. Panaudojimo atvejo S.1 aprašymas	28
14 lentelė. Panaudojimo atvejo S.1 aprašymas	28
15 lentelė. Panaudojimo atvejo S.2 aprašymas	29
16 lentelė. Panaudojimo atvejo S.3 aprašymas	30
17 lentelė. Panaudojimo atvejo S.4 aprašymas	31
18 lentelė. „BeamNG.Drive“ vykdomojo failo pasirinkimo ir paleidimo testavimo procesas	50
19 lentelė. „Beam Eye Tracker“ vykdomojo failo pasirinkimo ir paleidimo testavimo procesas	51
20 lentelė. Reikalingų programinių įrangų procesų tikrinimo testavimo procesas	52
21 lentelė. „InfluxDB“ duomenų bazės pasiekiamumo testavimo procesas.....	52
22 lentelė. MQTT brokerio pasiekiamumo testavimo procesas.....	53
23 lentelė. Stebėjimo sistemos funkcijų pasiekiamumo testavimo procesas	54
24 lentelė. Duomenų formavimo ir siuntimo į duomenų bazę testavimo procesas	54
25 lentelė. Analizės įrankio duomenų failo įkėlimo testavimo procesas	55
26 lentelė. Analizės įrankio įkeltų matavimų modifikavimo apsaugos testavimo procesas	56
27 lentelė. Analizės įrankio etaloninių matavimų pagal scenarijų failo įkėlimo testavimo procesas	56
28 lentelė. Analizės įrankio palyginimo rezultatų failo sukūrimo testavimo procesas.....	57
29 lentelė. Analizės įrankio išėjimo mygtuko testavimo procesas	57

Paveikslų sąrašas

1 pav. Ekstremalaus vairavimo simulatoriaus projekto vizija	16
2 pav. Ekstremalaus vairavimo simulatoriaus panaudojimo atvejų diagrama	24
3 pav. Ekstremalaus vairavimo simulatoriaus programinės įrangos architektūra.....	33
4 pav. Ekstremalaus vairavimo simulatoriaus mikrovaldiklio programinio kodo klasės	34
5 pav. Ekstremalaus vairavimo simulatoriaus analizės įrankio klasės.....	35
6 pav. Ekstremalaus vairavimo simulatoriaus duomenų bazės lentelės	36
7 pav. Ekstremalaus vairavimo simulatoriaus analizės įrankio klasės.....	37
8 pav. Ekstremalaus vairavimo simulatoriaus simuliacijos stebėsenos sekų diagrama (S.1)	38
9 pav. Ekstremalaus vairavimo simulatoriaus duomenų gavimo iš duomenų bazės sekų diagrama (SĮ.1)	38
10 pav. Ekstremalaus vairavimo simulatoriaus gautų scenarijaus duomenų analizės sekų diagrama (S.2)	39
11 pav. Ekstremalaus vairavimo simulatoriaus gautų scenarijaus duomenų analizės sekų diagrama (S.3)	40
12 pav. Ekstremalaus vairavimo simulatoriaus gautų scenarijaus duomenų analizės sekų diagrama (S.4)	41
13 pav. Ekstremalaus vairavimo simulatoriaus duomenų perdavimo sekų diagrama (SP.1)	42
14 pav. Ekstremalaus vairavimo simulatoriaus scenarijaus atlikimo sekų diagrama (V.1).....	43
15 pav. Ekstremalaus vairavimo simulatoriaus scenarijaus pradžios sekų diagrama (V.2)	43
16 pav. Ekstremalaus vairavimo simulatoriaus scenarijaus stabdymo sekų diagrama (V.3)	44
17 pav. Ekstremalaus vairavimo simulatoriaus simuliacijos naudotojo sąsaja.....	45
18 pav. Ekstremalaus vairavimo simulatoriaus simuliacijos naudotojo sąsaja.....	45
19 pav. Ekstremalaus vairavimo simulatoriaus stebėtojo naudotojo sąsajos projektas	46
20 pav. Ekstremalaus vairavimo simulatoriaus schema sujungus elementus	47
21 pav. Paruoštas prototipas su USB jungtimi.....	49
22 pav. AD8232 sensoriaus jutiklių jungimo schema.....	60
23 pav. Stebėjimo įrankio „Testing“ skiltis	61
24 pav. Stebėjimo įrankio „General“ skiltis.....	62
25 pav. Klaviatūros mygtukas žaidimo konsolės langui atidaryti.....	63
26 pav. Analizės įrankio išvaizda	64

Santrumpų ir terminų sąrašas

Terminai:

MQTT – komunikacijos protokolas, kuris siunčia žinutes tarp kompiuterių naudojantis prenumeravimo–viešinimo principą.

UML – vieninga modeliavimo kalba (Unified Modeling Language) skirta atvaizduoti ir sukurti objektais orientuotų programų dokumentus.

IoT – daiktų interneto technologija, kuri jungia prietaisus, turinčius prisijungimo prie tinklo funkciją į bendrą prietaisų tinklą, kurį galima pasiekti per internetą.

Ivadas

Transporto priemonių valdymas ekstremaliose situacijose reikalauja greitai ir tiksliai reaguoti į netikėtas ir pavojus keliančias situacijas: staigiai pasirodžiusios kliūtys, padangų sukibimo praradimas, kitų eismo dalyvių įvykius, nepalankias oro sąlygas ir daugiau [1]. Tokios situacijos reikalauja ne tik tinkamos įrangos ir komponentų, bet ir techninių vairavimo įgūdžių bei gebėjimo išlaikyti ramų, šalto būdo mąstymą bei koncentraciją.

Įprastai tiek vairavimo mokyklos, tiek AB „Regitra“ teorinio ar praktinio egzamino metu nereikalauja vairuotojų žinių ir gebėjimų, kaip tinkamai priimti sprendimus ekstremaliose situacijose, kurios gali susidaryti keliuose. Dėl to vairuotojai realiose sąlygose gali būti nepakankamai pasiruošę netikėtoms ir stresą keliančioms situacijoms. Nors egzistuoja ekstremalaus vairavimo mokymai, tačiau jie yra brangūs, reikalauja specialiai paruoštos aplinkos, automobilių, instruktorių.

Vairavimo simulatorių naudojimas suteikia galimybę įgauti pradinį įgūdžių ir tobulinti esančius įgūdžius saugioje, kontroliuojamoje aplinkoje, kur vairuotojo elgesys gali būti stebimas ir analizuojamas realiu laiku [2]. Daugelis simulatorių, kurie yra komerciškai parduodami, yra skirti laisvalaikio praleidimui ir be galimybės bei įrangos stebėti vairuotojo fiziologinius, bei elgesio duomenis, kurių pagalba būtų galima įvertinti ir pastebėti klaidas ekstremaliose situacijose.

Dėl šių problemų, kyla poreikis sukurti sistemą, kuri galėtų ne tik atkurti realaus vairavimo sąlygas, bet ir leistų objektyviai matuoti vairuotojo reakciją, emocinę būklę pasitelkiant jutiklius ir duomenų analizę. Tai suteiktų galimybę sukurti mokymosi scenarijus, atlikti fizinius ir psichologinius tyrimus bei tobulinti vairavimo įgūdžius prieš pritaikant šias žinias realiame pasaulyje.

1. Darbo tikslas

Sukurti ekstremalaus vairavimo simulatoriaus prototipą, kuris galėtų, naudojantis biometriniais jutikliais, telemetrijos duomenimis ir programinės įrangos analize, įvertinti vairuotojo reakciją, streso lygį bei vairavimo gebėjimus virtualiose ekstremaliose situacijose ir visą tai padaryti kiek įmanoma prieinamose sąlygose.

1.1. Darbo uždaviniai

Toliau yra aprašomi projekto darbo uždaviniai siekiant įgyvendinti šį projektą:

1. Išanalizuoti esamus ekstremalaus vairavimo simulatorius, jų galimybes, prieinamumą. Palyginti gautus rezultatus.
2. Parengti sistemos architektūrą, apimančią fizinį simulatorių, jutiklius ir valdiklius bei duomenų apdorojimą.
3. Įdiegti pagrindinius biometrinius jutiklius, kuriais galima būtų stebėti ir realiu laiku rinkti duomenis apie vairuotoją.
4. Sukurti duomenų bazę ir ją sujungti su fizine simulatoriaus sistema, siekiant išsaugoti duomenis ir juos analizuoti.
5. Ištestuoti sudarytos sistemos veikimą bei pašalinti pastebėtas klaidas duomenų rinkime, perdavime, analizėje.

2. Problemos analizė

Šiuolaikinis kelių eismas pasižymi dideliu intensyvumu, kintančiomis oro ir kelio sąlygomis, kuris reikalauja nuolatinio vairuotojų dėmesio ir reakcijos [3]. Nepaisant to, dauguma vairuotojų įgūdžius lavina tik per standartinius teorijos mokymus, kuriuose yra pasakojama apie kelių eismo taisykles ir ženklus, ir nustatytą kiekį praktinių važiavimų. Po teisinio leidimo vairuoti automobilį, įgūdžiai priklauso nuo pravažiuojamų kilometrų kiekio kasdienio eismo sąlygomis. Toks žinių ir įgūdžių pagrindas nepakankamai paruošia vairuotojus, ypač tuos, kurie neturi daug patirties ir nėra susidūrę su prastomis oro sąlygomis ar kitomis pavojų keliančiomis situacijomis, kuomet per kelias akimirkas yra būtina priimti tinkamiausią sprendimą, lemiantį saugumą ir sveikatą ekstremalioje situacijoje.

Tokiose aplinkybėse itin svarbu tampa vairuotojo psichologinė būsena, gebėjimas susidoroti su stresu, emocijomis, greitas mąstymas, aplinkos analizė ir koncentracija.

Daugelyje šalių, taip pat ir Lietuvoje, vairavimo mokymo programos neparuošia vairuotojų kaip elgtis:

- Praradus sukibimą priekinių, galinių bei visų varomų ratų transporto priemonėje;
- Avarinėje situacijoje, reikalaujančio stabdymo bei greito kliūčių apvažiavimo;
- Kitam eismo dalyviui įvykdžius netikėtą manevrą;
- Riboto matomumo ir nepalankių oro sąlygų metu.

Dėl to vairuotojai dažnai nėra pasirengę ir nežino, kaip reaguoti į tokio tipo įvykius, o jų elgesys gali būti nulemtas streso, panikos, lėtos reakcijos.

Šią problemą taip pat patvirtina kelių eismo įvykių statistiką, kuri įrodo, jog daugelis avarių kyla ne dėl teorinių žinių trūkumo, bet dėl neteisingos reakcijos į susidariusią pavojingą situaciją.

Nors egzistuoja ekstremalaus vairavimo mokymai, kuriuos dažniausiai organizuoja vairavimo centrai, turintys prieigą prie vairavimo aikštelių, tačiau jų prieinamumas yra ribotas, reikalaujantis gyvo instruktoriaus priežiūros, specializuotos fizinės transporto priemonės, techninės priežiūros centro. Dėl to, tokio tipo mokymai yra brangūs, reikalaujantys bent vienos laisvos dienos skirtos vairavimui, siekiant pagerinti vairavimo įgūdžius [4].

Standartiniai vairavimo simulatoriai suteikia galimybę saugiai atkartoti realias situacijas, analizuoti vairuotojo veiksmus ir reakciją. Dažniausiai yra kaupiami šie duomenys:

- pedalų paspaudimai;
- pavarų keitimai;
- vairo pasukimo kampas;
- automobilio greitis;
- gravitacinės jėgos.

Tačiau egzistuojantys komerciniai simulatoriai neturi:

- vairuotojo elgsenos analizės;
- biometrinių jutiklių integracijos ir duomenų rinkimo;
- akių judesių sekimo.

Tokios ribos lemia, jog esami simulatoriai gali rinkti duomenis tik vienpusiu būdu, tai, kas yra renkama pačios programinės įrangos, kurioje vykdoma simuliacija. Norint pridėti vairuotojo psichofiziologinių aspektų analizę, yra būtina pridėti papildomų sensorių, bei sukurti programinę įrangą, kuri galėtų rinkti ir analizuoti tokio tipo duomenis. Dažniausiai tokio tipo simulatoriai yra uždari, reikalauja daug paruošimo bei jų kaina gali viršyti 10000 eurų.

Išanalizavus šią informaciją, akivaizdu, jog rinka reikalauja prieinamos sistemos, kuri galėtų patenkinti šiuos punktus:

- gali atkartoti realias ekstremalias situacijas;
- leidžia matuoti vairuotojo reakcijos laiką;
- stebi fiziologinius rodiklius;
- fiksuoja akių žvilgsnio kryptį ir dėmesio koncentraciją;
- pateikia duomenų analizę realiu laiku.

Tokio tipo sistema galėtų būti naudojama ne tik mokymuose, bet ir tyrimuose, kurių pagalba galima tobulinti vairuotojų teorinius ir praktinius gebėjimus.

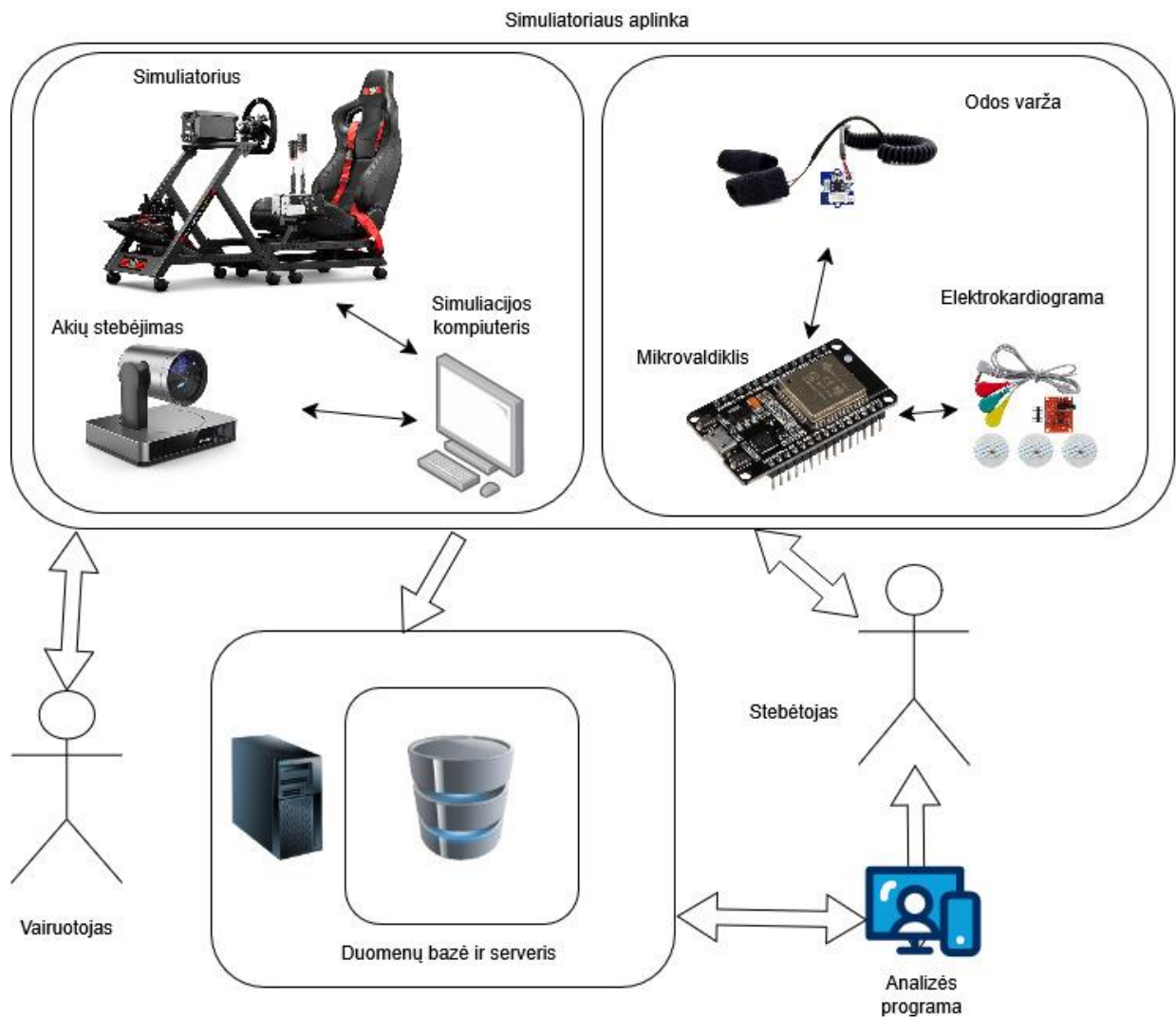
Kuriamam projektui, kuris galėtų patenkinti šiuos punktus yra reikalingi:

- mikrovaldiklis – mažos kainos, didelio funkcionalumo modulis, galintis rinkti duomenis iš kelių sensorių vienu metu ir perduoti juos;
- biometriniai jutikliai – leidžia matuoti širdies ritmą, sekti streso indikaciją;
- akių sekimo modulis – leidžia fiksuoti žvilgsnio kryptį;
- telemetrijos programa – leidžia realiu laiku gauti automobilio valdymo ir fizikinius duomenis;
- duomenų bazės ir analizės įranga – leidžia apdoroti ir interpretuoti surinktus duomenis.

Šių technologijų derinimas leidžia sukurti prieinamą, išplečiamą ir tyrimams tinkamą sistemą, kuri gali būti labiau prieinama nei jau esantys komerciniai sprendimai.

2.1. Projekto vizija

Šioje vizijoje siekiama atvaizduoti šio projekto veikimo realizacija tiek iš programinės pusės, tiek iš pagrindinių fizinių komponentų (žr. **1 pav.**).



1 pav. Ekstremalaus vairavimo simulatoriaus projekto vizija

2.2. Konkurentų analizė

Nors Lietuvoje tokio tipo sistemų nėra, turime lyginti projekto idėją su užsienyje esančiais produktais, kurie yra komerciškai pasiekiami. Beveik visi tokio tipo projektai buvo rasti Jungtinėse Amerikos Valstijose, todėl integracijos, kaip elektros tiekimas, techninis palaikymas, ir importavimo iššūkiai, kaip muitai, siuntimo mokesčiai, gerokai pasunkina konkurentų produktų diegimą Lietuvoje. Nepriklausomai nuo šių logistinių ir finansinių problemų, sugalvotas produktas bus lyginamas su konkurentais pagal jų funkcionalumą ir kainas, kuriomis galima juos įsigyti iš tiekėjo.

Pagrindiniai rasti konkurentai:

Drive Square Simulation System [5] – JAV įmonės produktas, kurio išskirtinis bruožas – mokymasis/kvalifikavimasis su realioje transporto priemonėje įdiegta valdymo sistema. Simuliacijos scenarijai gali būti paruošti naudojantis jų programine įranga, vairuotojas viską mato per specialiai paruoštus virtualios realybės akinius. Pradinė kaina prasideda nuo 10000 JAV dolerių.

SimXperience ESR-1 [6] – aukšto lygio lenktynių simulatorius su galimybe papildomai imituoti automobilio judesius. Šis simulatorius nėra apribotas programinės įrangos, todėl galima taikyti tiek

pramogoms, tiek specializuotiems mokymams. Šis simulatorius yra labiau orientuotas į naudotojus, kurie ieško lenktynių simulatoriaus, kurį galima naudoti be papildomų paruošimų. Bazinė kaina prasideda nuo 9999 JAV dolerių, neįtraukiant automobilio judesių imitacijos, pavarų ar vaizduoklio tvirtinimo elementų.

Simulator Systems International (Risk mitigation) [7] – pats paprasčiausias sprendimas, kurio pagalba galima mokyti vairuotojus vairavimo įgūdžių. Lyginant su kitais konkurentais, visos sistemos veikimas yra viename kompiuteryje, be galimybės matyti duomenis realiu laiku nepertraukiant simuliacijos. Kartu su simulatoriumi yra bazinė programinė įranga, kurioje yra paruošti scenarijai ir galimybė koreguoti automobilio elektroninės pagalbos sistemas bei automobilio parametrus. Bazinė kaina prasideda nuo 10000 eurų.

Tecknotrove Car Driving Simulator [8] – arčiausiai rastas sprendimas, kurio tikslas mokyti įvairių gebėjimų vairuotojus. Turi iš anksto paruoštus scenarijus, telemetriją, rezultatų analizę, scenarijaus koregavimą gyvu laiku, galimybę sujungti kelis simulatorių į vieną bendrą scenarijų. Kainos nėra nurodyta, tačiau pagal programinę bei techninę įrangą su kuria yra paruoštas šis projektas, galima teigti, jog bazinis variantas kainuos bent 10000 eurų.

1 lentelė. Konkurentų palyginimo kriterijai

Sistema	Komerciškai prieinama?	Palaiko akių stebėjimą?	Palaiko biometrinių duomenų matavimą?	Turi telemetriją?	Ar galima naudoti be specializuotos programinės įrangos?	Kaina iki 1000 eurų?
Drive Square EVO	+	+	-	+	-	-
SimXperience ESR-1	+	-	-	+	+	-
Simulator Systems International (Risk mitigation)	+	-	-	+	-	-
Tecknotrove Car Driving Simulator	+	-	-	+	-	-
Ekstremalaus vairavimo simulatorius	+	+	+	+	+	+

Galima matyti, kad nei vienas iš komercinių sprendimų nesiūlo biometrinių duomenų matavimo ir tik vienas iš variantų palaiko akių stebėjimą. Kadangi visi, išskyrus „SimXperience ESR-1“, turi savo specializuotas programines įrangas, tai gali apriboti laisvę norint integruoti naujus scenarijus ar funkcionalumus siekiant suteikti geresnę ir pilnesnę patirtį naudotojui. Visų konkurentų kainos viršija mūsų simulatoriaus kainą bent 10 kartų, tai reiškia, kad kuriamas sprendimas yra gerokai labiau prieinamas kainos atžvilgiu.

2.3. Programinės įrangos analizė

Šioje dalyje analizuosime programinės įrangos variantus, atrinksime pagrindinius konkurentus ir, pagal iškeltus kriterijus, pagrįsime, kodėl pasirinktas variantas yra tinkamiausias.

2.3.1. Duomenų bazių variantų palyginimas

Norint efektyviai rinkti, analizuoti ir skirstyti duomenis pagal vairavimo užduotis ar bandymus, reikalinga duomenų bazė, kuri galėtų saugoti reikiamus duomenis. Saugomi duomenys turės būti pasiekiami iš mūsų pasirinktos aplinkos. Pagrindiniai kriterijai palyginimui yra:

- ar duomenys gali būti perduodami greičiau nei 0.1 sekundės – turimi mikrovaldiklių komunikacijos protokolai, gali perduoti informaciją greičiau nei 0.1 sekundės;
- integruota duomenų išsaugojimo funkcija – tai leidžia išsaugoti norimų bandymų individualius rezultatus, kuriuose būtų pateikiama pilna informacija, net jei komunikacija tarp pasirinktos aplinkos ir duomenų bazės nutrūktu;
- integracija tarp duomenų bazės ir daiktų interneto technologijų (IoT) komponentų;
- kaina – siekiant sudaryti kuo mažesnius kaštus, kas paverčia projektą komerciškai patrauklesniu.

Pagrindiniai variantai, kurie galėtų atitikti mūsų reikalavimus yra „InfluxDB“, „MongoDB“ ir „Cassandra“.

InfluxDB [9] – specializuota duomenų bazė, sukurta nuolatiniam ir greitam duomenų rašymui. Ši duomenų bazė yra sukurta laiko eilučių pagrindu.

MongoDB [10] – dokumentinė, „NoSQL“ duomenų bazė, kuri saugo informaciją „JSON“ tipo dokumentuose.

Cassandra [11] – paskirstyta „NoSQL“ duomenų bazė, sukurta dirbti su didelio masto duomenų kiekiais ir užtikrinti pastovų prieinamumą.

2 lentelė. Duomenų bazių palyginimas pagal iškeltus kriterijus

Pavadinimas	Ar duomenys gali būti perduodami greičiau nei 0.1s?	Ar integruota automatinė duomenų įrašymo ir išsaugojimo funkcija?	Ar yra tiesioginis IoT komponentų palaikymas?	Ar nemokama?
InfluxDB	+	+	+	+
MongoDB	+	-	-	+
Cassandra	+	+	-	+

Matome, kad tinkamiausias variantas yra „InfluxDB“, kadangi jis atitinka visus mūsų kriterijus.

2.3.2. Komunikacijos protokolų variantų palyginimas

Komunikacijos protokolai yra pagrindas, kaip projekto komponentai galės komunikuoti su analizės priemonėmis. Parinkti kriterijai:

- galimybė siųsti duomenis bent kas 0.1 sekundę, siekiant palaikyti pastovų duomenų naujinimą ir sudaryti galimybę kuo tiksliau išanalizuoti scenarijų įvykdymus;
- užtikrinamas duomenų gavimas, kadangi tiksliais matavimams reikia, kad duomenys būtų perduodami pastoviai ir susidarius nuokrypiams, nebūtu paveikti galutiniai rezultatai;
- integracija į mūsų pasirinktą duomenų bazę, šiuo atveju „InfluxDB“.

Atrinkti variantai buvo MQTT, HTTP, WebSocket.

MQTT [12] – prenumeravimo–viešinimo principu paremtas tinklo protokolas, sukurtas perduoti duomenis tarp ribotus resursus turinčių įrenginių.

HTTP [13] – kliento–serverio principu paremtas protokolas, skirtas perduoti įvairaus tipo informaciją. Užklauskos vykdomos tekstiniu formatu.

WebSocket [14] – dvikryptis, kliento–serverio principu paremtas protokolas skirtas keistis duomenimis realiu laiku.

3 lentelė. Komunikacijos protokolų palyginimas

Pavadinimas	Ar duomenys perduodami greičiau nei 0.1 sekundės?	Ar užtikrinamas duomenų gavimas?	Ar „InfluxDB“ turi integruotą palaikymą?
MQTT	+	+	+
HTTP	-	+	+
WebSocket	+	+	-

Pagal iškeltus kriterijus, geriausiai atitinkantis komunikacijos protokolas tarp mikrovaldiklio ir duomenų bazės yra MQTT.

2.3.3. Simuliacijos programinė įranga

Siekiant įgyvendinti visus simulatoriaus elementus, reikalinga programinė įranga, kurioje bus atliekamos simuliacijos, sukuriami scenarijai. Pasirinkti kriterijai:

- kaina – siekiant palaikyti projekto kaštus kiek įmanoma žemesnius buvo nuspręsta, kad kaina negali viršyti 25 eurų;
- galimybė kurti savo scenarijus – tai padės plėsti mokymosi katalogą, kurį galima pateikti vairuotojams praktikuotis;
- fizikos realistiškumas – simulatorius turi turėti kiek įmanoma tikroviškesnę fizikinių reiškinių ir jėgų simuliaciją;
- telemetrija – norint gauti tam tikrus duomenis iš pačių scenarijų, reikalinga komunikacija tarp simuliacijos ir duomenų bazės.

Atrinkti variantai „City car driving“, „BeamNG.drive“ ir „OpenDS“.

City car driving [15] – realistiškas miesto vairavimo simulatorius, orientuotas į kasdienio eismo situacijų atkūrimą.

BeamNG.drive [16] – pažangus vairavimo simulatorius, žinomas dėl savo fizikos simuliacijos, palaikantis deformacijas ir dinamišką modeliavimą.

OpenDS [17] – atvirojo kodo vairavimo simulatorius, sukurtas moksliniams tyrimams.

4 lentelė. Simuliacijos programų palyginimo lentelė

Pavadinimas	Ar yra galimybė kurti scenarijus?	Ar fizikinės jėgos bei reiškiniai yra atvaizduojami?	Ar yra palaikoma telemetrija?	Ar kaina neviršija 25 eurų?
City car driving	+	-	-	+
BeamNG.drive	+	+	+	+
OpenDS	+	-	+	+

Matoma, jog vienintelis pasirinkimas tampa „BeamNG.drive“ simulatorius.

2.3.4. Akių stebėjimo programinė įranga

Siekiant integruoti akių stebėjimą, reikalinga programinė įranga kamerai, kuri atitiktų šiuos kriterijus:

- kaina – siekiant palaikyti projekto kaštus kiek įmanoma mažesnius, siekiama, kad kaina būtų iki 100 eurų;
- galimybė taikyti su skirtingų gamintojų kameromis – siekiant išlaikyti projekto prieinamumą ir galimybę keisti komponentus be papildomų kodo pakeičių;
- tikslumas – norint tinkamai stebėti, į kurią tašką vairuotojas kreipia dėmesį, reikalingas maksimaliai 2 laipsnių tikslumas;
- dažnis – siekiant pastoviai ir tiksliai perduoti informaciją apie akių stebėjimą reikalinga, jog duomenų nuskaitymo dažnis būtų nemažiau kaip 0.1 sekundės.

Atrinkti pasirinkimai „EyeWare Beam edition“, „OpenCV“, „Tobii Eye Tracking“.

EyeWare Beam edition [18] – programinė įranga, kuri naudoja įprastą internetinę kamerą žvilgsnio sekimui ir galvos judesių stebėjimui.

OpenCV [19] – atvirojo kodo kompiuterinės regos biblioteka, naudojama vaizdų apdorojimui, objektų atpažinimui ir realaus laiko analizei.

Tobii Eye Tracking [20] – profesionali žvilgsnio sekimo technologija, naudojanti specializuotus infraraudonųjų spindulių jutiklius, kad tiksliai nustatytu akių judesius.

5 lentelė. Akių stebėjimo programinės įrangos palyginimas

Pavadinimas	Ar yra galimybė taikyti su skirtingomis kameromis?	Ar tikslumo yra didesnis nei 2 laipsniai?	Ar duomenys yra atnaujinami bent iki 0.1 sekundės?	Ar kaina yra mažesnė nei 100 eurų?
EyeWare Beam Edition	+	+	+	+
OpenCV	+	-	-	+
Tobii Eye Tracking 5	-	+	+	-

Pagal kriterijus geriausiai tinka „EyeWare Beam edition“.

2.4. Fizinės įrangos analizė

Šioje dalyje analizuosime fizinę įrangą pagal kategorijas, pateiksime jų palyginimo kriterijus ir galutinai pasirinktus elementus.

2.4.1. Mikrovaldiklių variantų palyginimas

Norint sėkmingai nuskaityti ir perduoti duomenis reikalingi:

- mikrovaldiklis turi turėti pakankamai operatyvios atminties, kodo vykdymui, duomenų nuskaitymui ir jų perdavimui komunikacijos protokolais;
- svarbu, kad mūsų mikrovaldiklis turėtų integruotą „Wi-Fi“, siekiant turėti galimybę be papildomų pakeitimų iškart perduoti duomenis MQTT protokolu;
- reikalingas ADC keitiklis, kadangi mums reikalingi sensoriai duomenis perduoda analoginiu formatu;
- kaina negali viršyti 15 eurų, kadangi tai yra vidutinė programuojamų mikrovaldiklių kaina rinkoje.

Atrinkti pasirinkimai – „ESP32-S3“, „Arduino Mega“, „Raspberry Pi 4“.

ESP32-S3 [21] – mikrovaldiklis su integruotu „Wi-Fi“ ir „Bluetooth“. Turi dviejų branduolių procesorių. Yra galimybė prijungti didelį kiekį papildomų modulių.

Arduino Mega [22] – mikrovaldiklis turintis 54 skaitmeninių įvesčių ir 16 analoginių įvesčių kiekį, tačiau neturi integruoto „Wi-Fi“ ar „Bluetooth“, yra vieno branduolio procesorius.

Raspberry Pi 4 [23] – mini kompiuteris, skirtas sudėtingesnėms užduotims, vaizdo apdorojimui, serveriams, duomenų analizei. Turi keturių branduolių procesorių, tačiau nepalaiko realaus laiko jutiklių be papildomų priemonių.

6 lentelė. Mikrovaldiklių palyginimas

Mikrovaldiklis	Ar integruotos atminties dydis didesnis nei 512 kB?	Ar yra integruotas „Wi-Fi“?	Ar yra integruotas ADC keitiklis?	Ar kaina yra mažesnė nei 15 eurų?
ESP32-S3	+	+	+	+
Arduino Mega	-	-	-	+
Raspberry Pi 4	+	+	-	-

Pagal kriterijus, galime matyti, jog geriausias pasirinkimas yra „ESP32-S3“.

2.4.2. Širdies veiklos matavimo sensorių palyginimas

Širdies veiklos matavimo sensorių pasirinkimas priklauso nuo šių kriterijų:

- atsparus judesiams, kadangi vykdant scenarijus, kūnas gali judėti, o tikslams matavimams reikalingas pastovus matavimas ir kontaktas;
- širdies veiklos matavimo tikslumas, kadangi mums reikalinga turėti kuo tikslesnius galutinius rezultatus;

- matavimo greitis – lėtas duomenų perdavimas nesuteikia tikslių duomenų analizuojant veiksmų eigos, taikysime, kad greitis negali būti lėtesnis nei 0.1 sekundės;
- kaina, siekiant turėti kuo mažesnius kaštus, iki 10 eurų.

Atrinkti pasirinkimai – MAX30102, KY-039, AD8232.

MAX30102 [24] – integruotas optinis jutiklis, matuojantis širdies ritmą ir kraujo prisotinimą deguonimi. Reikalingas pastovus kontaktas su oda.

KY-039 [25] – optinis pulso jutiklis, naudojantis infraraudonųjų spindulių LED ir foto tranzistorių.

AD8232 [26] – analoginis elektrokardiogramos signalų stiprintuvas ir filtravimo modulis.

7 lentelė. Širdies veiklos matavimo sensorių palyginimas

Sensorius	Ar atsparus judesiams?	Ar matavimo paklaida yra iki 2%?	Ar duomenų perdavimo greitis didesnis nei 0.1 sekundės?	Ar kaina yra mažesnė nei 10 eurų?
MAX30102	-	+	+	+
KY-039	-	-	+	+
AD8232	+	+	+	+

Geriausiai mūsų kriterijus atitinka AD8232 sensorius.

2.4.3. Streso matavimo sensorių palyginimas

Streso matavimo sensorių pasirinkimas yra pagal šiuos kriterijus:

- informacijos skaitymas – mums reikalinga informacijos perdavimo sparta iš sensoriaus į mikrovaldiklį, kaip ir prieš tai, optimaliausiai skaičiuojame, jog greitis turi būti mažesnis nei 0.1 sekundės;
- lengva integracija su simulatoriumi, nors skamba abstrakčiai, tačiau sensorius turi turėti galimybę būti suderinamas ir naudojamas su sistema, kuri turės daug judesių;
- matavimo tikslumas – kaip ir širdies veiklos matavimo kriterijus, čia mums reikalingas kuo tikslesnis galutinis matavimų rezultatas;
- kaina, kuri negali viršyti 15 eurų.

Atrinkti pasirinkimai – „Grove GSR“, „FSR-402“, „DHT11“.

Grove GSR [27] – matuoja odos elektrinį laidumą, kuris kinta priklausomai nuo žmogaus streso, emocijų ar prakaitavimo lygio.

FSR-402 [28] – spaudimui jautrus rezistorius, kurio varža mažėja didėjant spaudimui.

DHT11 [29] – skaitmeninis temperatūros ir drėgmės jutiklis.

8 lentelė. Streso matavimo sensorių palyginimas

Sensorius	Ar duomenų perdavimo greitis didesnis nei 0.1 sekundės?	Ar paprasta integruoti?	Ar matavimo paklaida yra iki 2%?	Ar kaina yra mažesnė nei 15 eurų?
-----------	---	-------------------------	----------------------------------	-----------------------------------

Grove GSR	+	+	+	+
FSR-402	+	-	+	+
DHT11	-	+	-	+

Geriausiai kriterijus atitinka „Grove GSR“ sensorius.

2.5. Ekstremalaus vairavimo simulatoriaus komponentų analizės išvados

Atlikus analizę, nustatytos tinkamiausios programinės įrangos bei fizinių komponentų technologiniai sprendimai. Atlikus duomenų bazės technologijų analizę, komunikacijos protokolo technologiniai sprendimai tapo tiesiogiai susiję. Taip pat turint programinę įrangą ir komunikacijos protokolą buvo konkretizuotas pasirinkimas ties galimais mikrovaldiklio pasirinkimais, kadangi komunikacijos protokolų palaikymas tampa sąsajos dalis tarp veikiančios sistemos ir stebėtojo stebimos analizės įrankio.

3. Ekstremalaus vairavimo simulatoriaus projektavimas

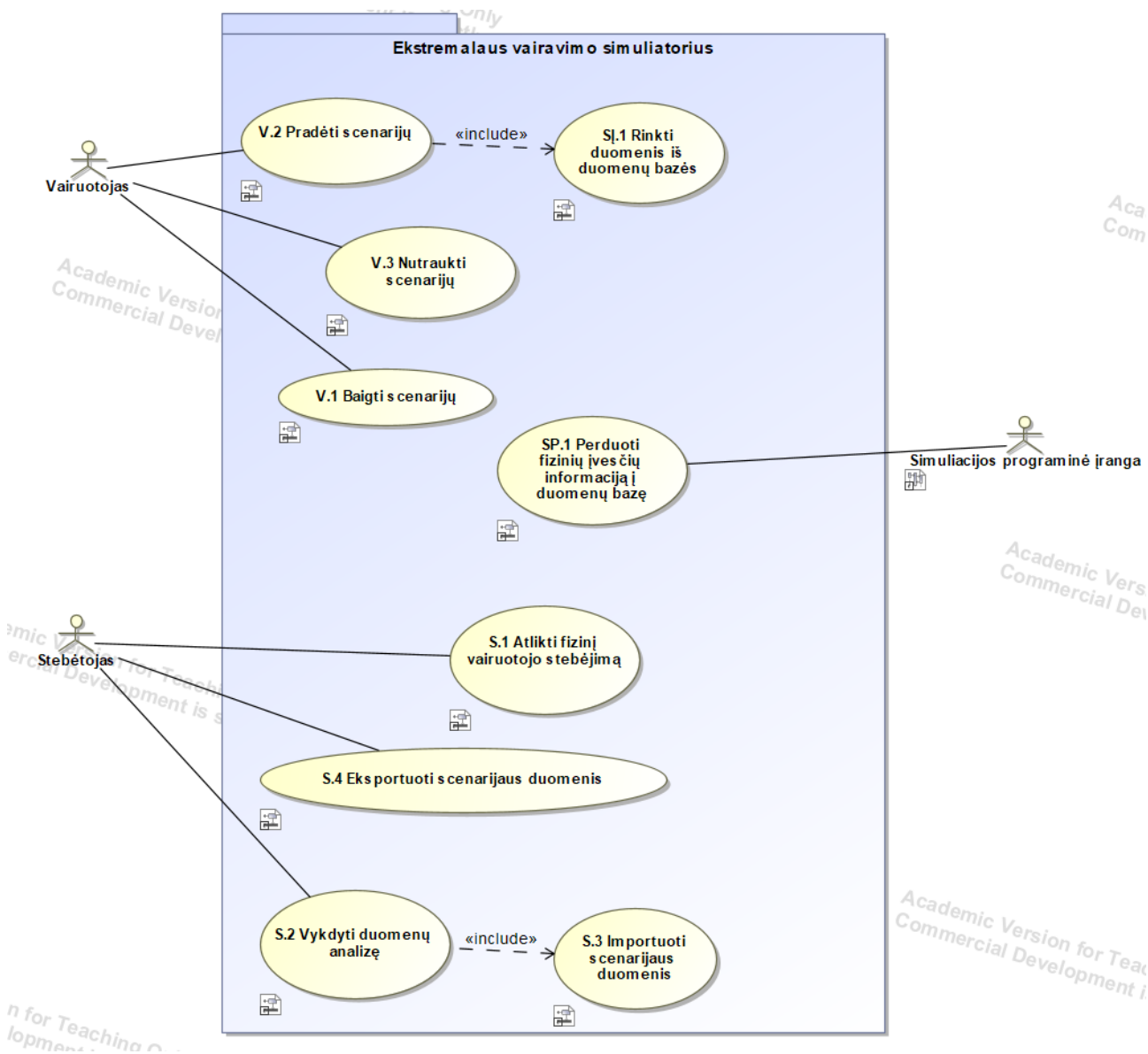
Šiame skyriuje aprašoma ekstremalaus vairavimo simulatoriaus projektavimo specifikacija.

3.1. Ekstremalaus vairavimo simulatoriaus funkciniai reikalavimai

Pagal darbo funkcinis reikalavimus, naudojantis UML specifikacija, buvo sukurta panaudojimų atvejų diagrama (žr. 2 pav.). Ji apibrėžia pagrindinius ekstremalaus vairavimo simulatoriaus funkcionalumus.

Panaudojimo atvejų diagramoje, veiksmus atlieka šie aktoriai:

- stebėtojas – tai asmuo, kuris vykdo sistemos ir vairuotojo stebėjimą ir vertinimą;
- vairuotojas – tai asmuo, kuris vykdo nustatytus scenarijus, kol yra stebimas stebėtojo ir jutiklių;
- simuliacijos programinė įranga – vykdo simuliaciją, kurioje vairuotojas atlieka scenarijus;
- stebėjimo įrankis – atvaizduoja surinktus duomenis iš duomenų bazės stebėjimo įrankyje.



2 pav. Ekstremalaus vairavimo simulatoriaus panaudojimo atvejų diagrama

Tolimesniuose poskyriuose detalizuojami panaudojimo atvejai aprašymais ir veiklos diagramomis pagal sudarytą panaudojimo atvejų diagramą.

3.1.1. Ekstremalaus vairavimo simulatoriaus panaudojimo atvejai

Šiame poskyryje pateikiami ir detalizuojami ekstremalaus vairavimo simulatoriaus panaudojimo atvejai lentelėmis ir aprašais.

9 lentelė. Panaudojimo atvejo V.1 aprašymas

Panaudojimo atvejis	V.1 Pradėti scenarijų
Aktoriai	Vairuotojas ir simuliacijos programinė įranga.
Aprašymas	Vairuotojas supažindinamas su scenarijaus tikslu. Susipažinęs ir pasiruošęs vykdyti scenarijų, vairuotojas spaudžia mygtuką „Start“.
Sąlyga - prieš	Stebėtojas užtikrina, kad vairuotojas ir jo stebimi duomenys yra korektiški. Simuliacija paruošta vykdymui.
Sąlyga - po	Vairuotojas pradeda vykdyti scenarijų.
Grafinė sąsaja	
Veiklos diagrama	

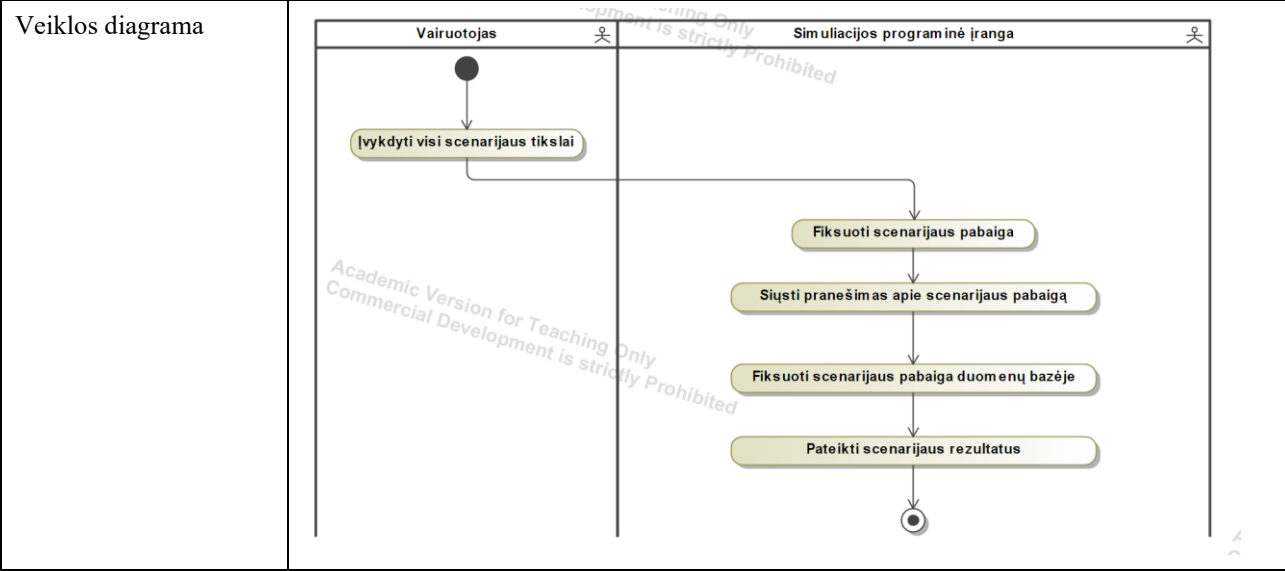
10 lentelė. Panaudojimo atvejo V.2 aprašymas

Panaudojimo atvejis	V.2 Nutraukti scenarijų
---------------------	-------------------------

Aktoriai	Vairuotojas ir simuliacijos programinė įranga.
Aprašymas	Esant poreikiui vairuotojas gali nutraukti scenarijaus vykdymą.
Sąlyga - prieš	Pradėtas vykdyti scenarijus.
Sąlyga - po	Scenarijus sustabdytas.
Veiklos diagrama	<pre> graph TD subgraph Vairuotojas Start(()) --> S1[Sustabdyti scenarijų] end subgraph Simuliacijos_programinė_įranga S1 --> S2[Siųsti stabdymo pranešimą] S2 --> S3[Perduoti stabdymo pranešimą į duomenų bazę] S3 --> S4[Sustabdyti scenarijaus vykdymą] S4 --> S5[Sustabdyti duomenų rinkimą] S5 --> End((())) end </pre>

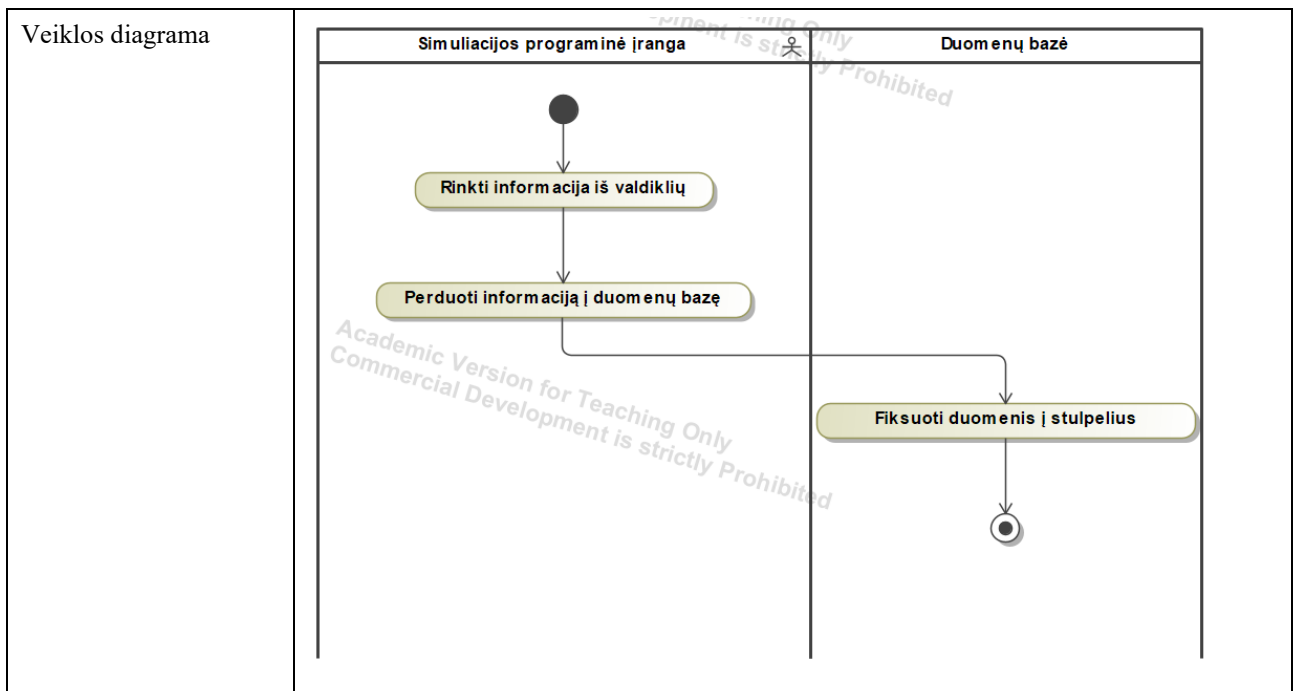
11 lentelė. Panaudojimo atvejo V.3 aprašymas

Panaudojimo atvejis	V.3 Baigti scenarijų
Aktoriai	Vairuotojas ir simuliacijos programinė įranga.
Aprašymas	Simuliacijos programa užfiksuoja scenarijaus pabaigą pagal iš anksto nurodytą galutinio tikslo pasiekimą ar kelių tikslų įvykdymą.
Sąlyga - prieš	Pradėtas vykdyti scenarijus.
Sąlyga - po	Scenarijaus vykdymas baigtas, duomenys nebėra fiksuojami analizei.
Grafinė sąsaja	



12 lentelė. Panaudojimo atvejo SP.1 aprašymas

Panaudojimo atvejis	SP.1 Perduoti fizinių įvesčių informaciją į duomenų bazę
Aktoriai	Simuliacijos programinė įranga.
Aprašymas	Simuliacijos programinė įranga perduoda vairo padėties, pedalų padėties, pavarų svirties padėties informaciją į duomenų bazę.
Sąlyga - prieš	Scenarijus paruoštas vykdymui.
Sąlyga - po	Scenarijaus vykdymas baigtas.
Grafinė sąsaja	

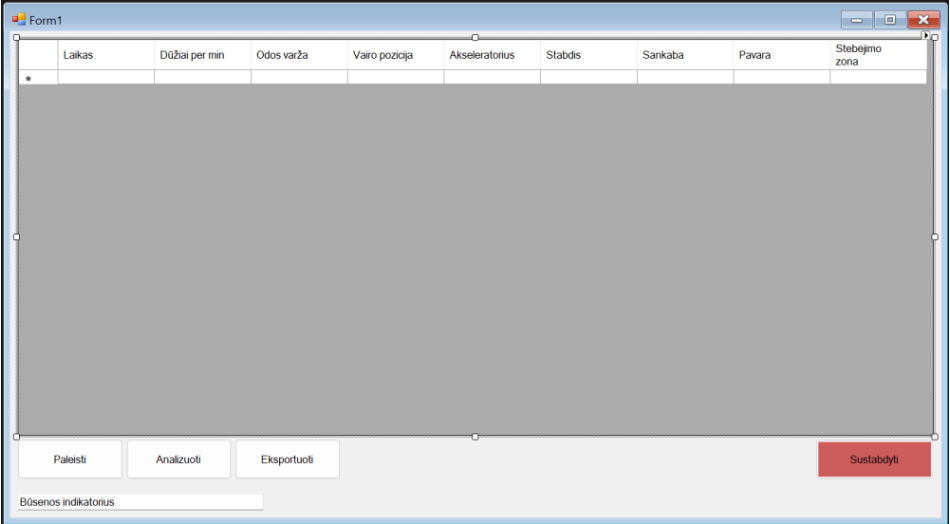
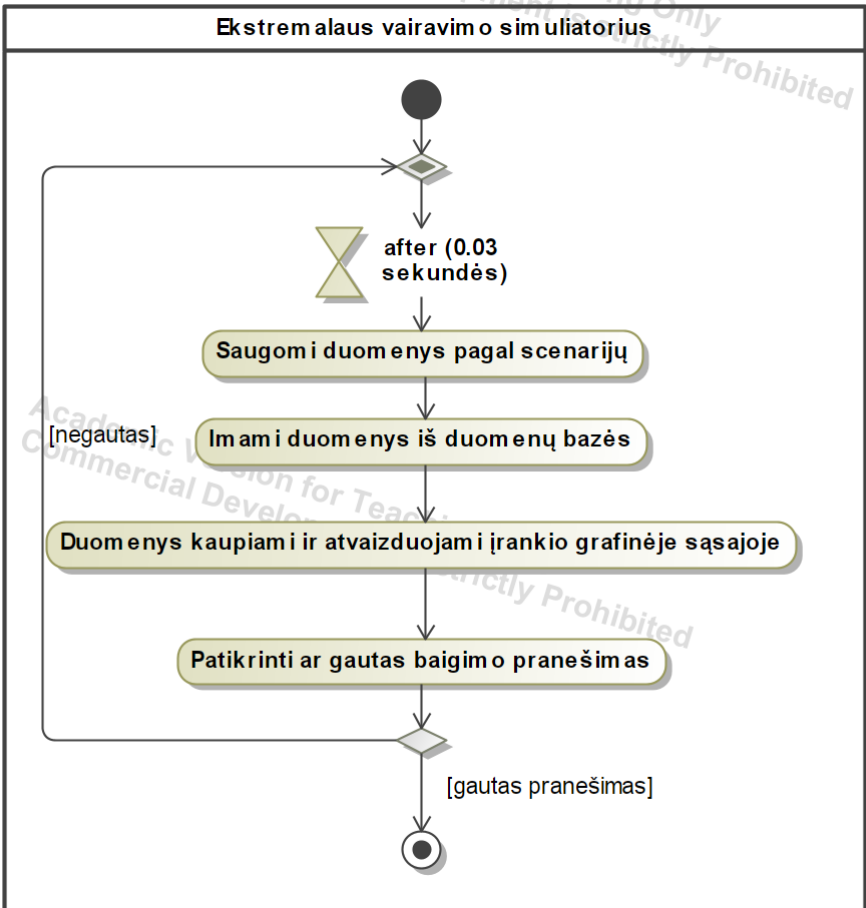


13 lentelė. Panaudojimo atvejo S.1 aprašymas

Panaudojimo atvejis	S.1 Atlikti simuliacijos stebėseną
Aktoriai	Stebėtojas.
Aprašymas	Vykdo vairuotojo stebėjimą prieš, per ir po scenarijaus įvykdymo laikotarpį.
Sąlyga - prieš	Vairuotojas yra supažindintas su sistema ir yra pasiruošęs vykdyti scenarijų.
Sąlyga - po	Vairuotojas atliko iškeltus scenarijaus arba scenarijų tikslus ir baigia veiklą su simuliacija.
Veiklos diagrama	<pre> graph TD Start(()) --> Stebėti[Stebėti duomenų eigą] Stebėti --> D1{ } D1 -- "[ne]" --> Patikrinti1[Patikrinti ar duomenų siuntimas sustabdytas] D1 -- "[taip]" --> Patikrinti2[Patikrinti ar duomenys siunčiami taisyklingai] Patikrinti1 --> D2{ } D2 -- "[ne]" --> Sustabdyti[Duomenys sustabdyti] D2 -- "[taip]" --> D3{ } Patikrinti2 --> D3 D3 -- "[ne]" --> Ignoruoti[Duomenys ignoruoti] D3 -- "[taip]" --> D4{ } Ignoruoti --> D4 Sustabdyti --> D4 D4 --> Nutraukti[Nutraukti simuliaciją] Nutraukti --> End((())) </pre>

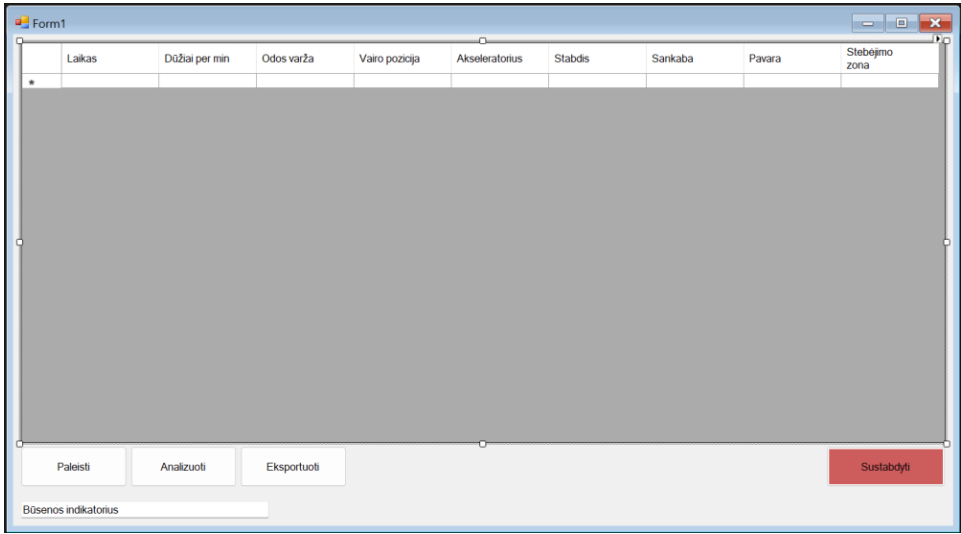
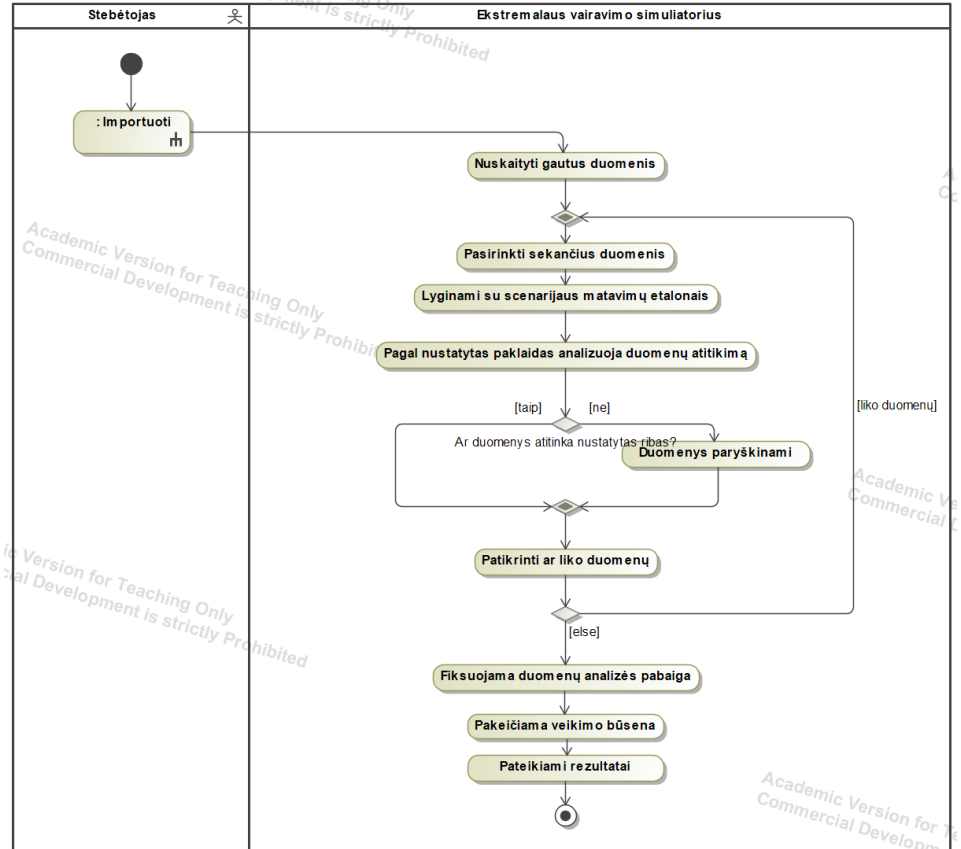
14 lentelė. Panaudojimo atvejo S.1 aprašymas

Panaudojimo atvejis	S.1 Rinkti duomenis iš duomenų bazės
---------------------	--------------------------------------

Aktoriai	Ekstremalaus vairavimo simulatorius.
Aprašymas	Įrankis vykdo duomenų rinkimą iš duomenų bazės, kurie bus analizuojami pagal vykdomą scenarijų.
Sąlyga - prieš	Vairuotojas pradėjo vykdyti scenarijų.
Sąlyga - po	Vairuotojas baigė vykdyti scenarijų.
Grafinė sąsaja	
Veiklos diagrama	

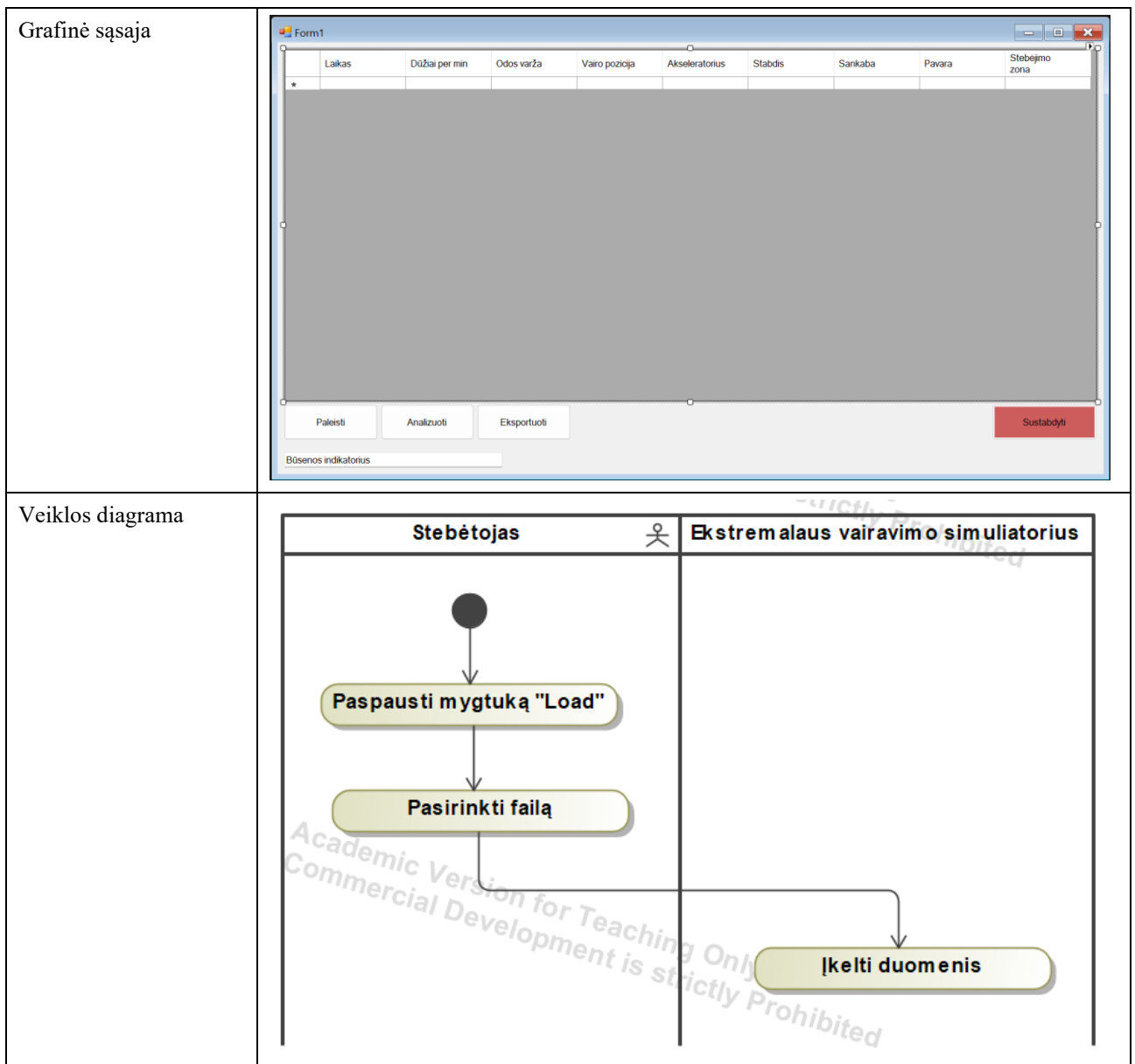
15 lentelė. Panaudojimo atvejo S.2 aprašymas

Panaudojimo atvejis	S.2 Vykdyti duomenų analizę
Aktoriai	Ekstremalaus vairavimo simulatorius.

Aprašymas	Iš duomenų bazės gautus duomenis išanalizuoja pagal scenarijaus iškeltus tikslus.
Sąlyga - prieš	Vairuotojas baigė vykdyti scenarijų.
Sąlyga - po	Analizės įrankis atliko duomenų analizę pagal scenarijų.
Grafinė sąsaja	
Veiklos diagrama	

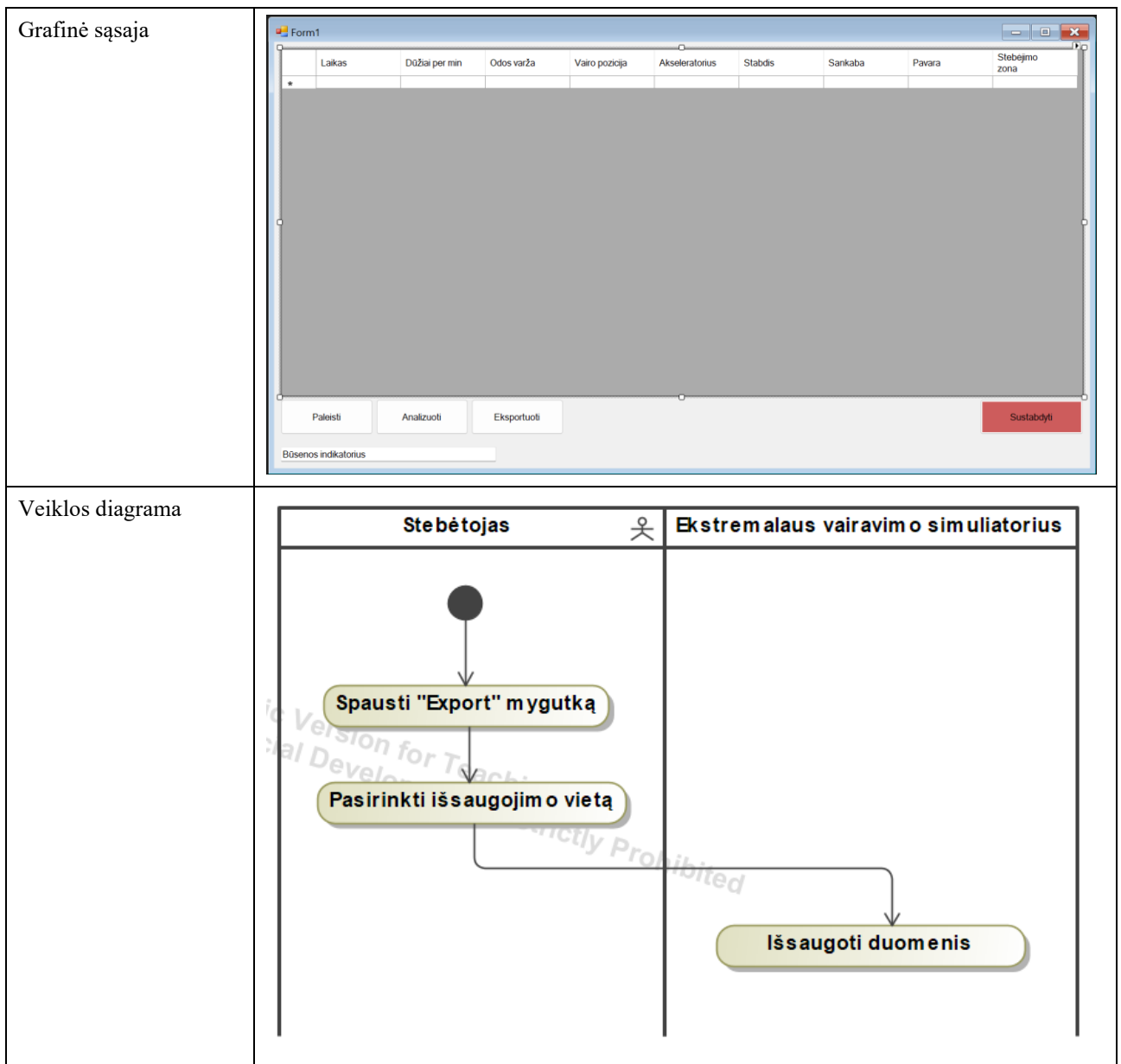
16 lentelė. Panaudojimo atvejo S.3 aprašymas

Panaudojimo atvejis	S.3 Importuoti scenarijaus duomenis
Aktoriai	Ekstremalaus vairavimo simulatorius.
Aprašymas	Įkelti duomenų rinkinį į analizės įrankį.
Sąlyga - prieš	Analizės įrankyje nėra įkeltų duomenų.
Sąlyga - po	Analizės įrankis turi įkeltus duomenis.



17 lentelė. Panaudojimo atvejo S.4 aprašymas

Panaudojimo atvejis	S.4 Eksportuoti scenarijaus duomenis
Aktoriai	Ekstremalaus vairavimo simulatorius.
Aprašymas	Išanalizuoti duomenys išsaugomi lokaliai.
Sąlyga - prieš	Analizės įrankis atliko duomenų analizę pagal scenarijų.
Sąlyga - po	Analizės įrankis išsaugo gautus rezultatus.



3.1.2. Ekstremalaus vairavimo simulatoriaus nefunkciniai reikalavimai

Ekstremalaus vairavimo simulatoriaus projekto užduotyje iškelti tokie nefunkciniai reikalavimai:

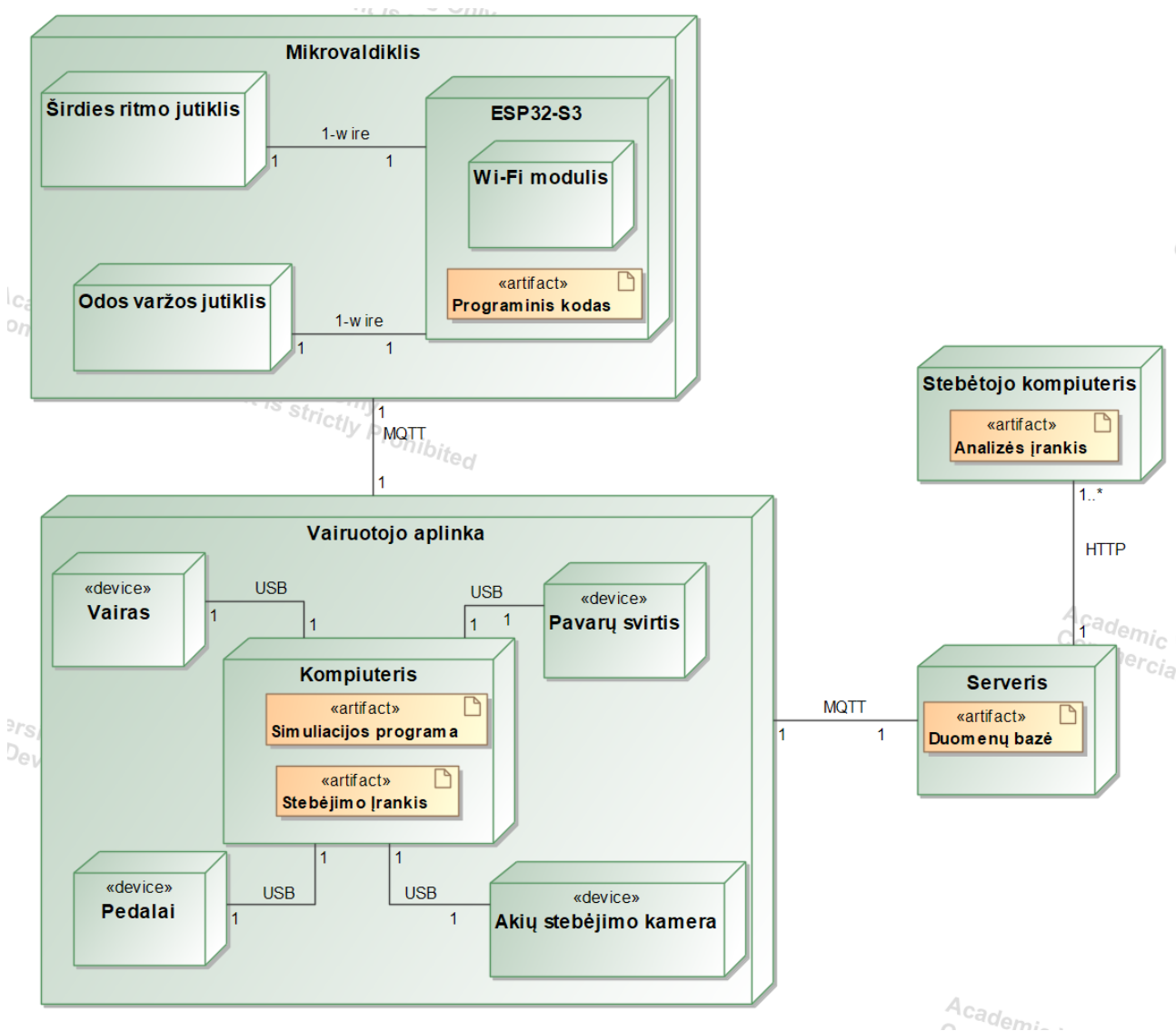
- perduoti duomenys turi būti išsaugomi duomenų bazėje be naudotojo įsiterpimo;
- sistema turi palaikyti pastovų, nepertraukiamą duomenų srautą simuliacijos metu;
- sistema neturi kelti pavojaus naudotojui ar kitiems asmenims;
- sistemos jutikliai negali trukdyti naudotojui vykdyti simuliacijos scenarijų.

4. Ekstremalaus vairavimo simulatoriaus programinės įrangos projektas

Šiame skyriuje ekstremalaus vairavimo simulatoriaus programinės įrangos projektas detalizuotas UML diagramomis.

4.1.1. Ekstremalaus vairavimo simulatoriaus programinės įrangos architektūra

Pateiktoje ekstremalaus vairavimo simulatoriaus diegimo diagramoje (žr. 3 pav.) yra atvaizduojami sąryšiai tarp projekto sistemos dalių. Asmens stebėjimo jutikliai (širdies ritmo ir odos varžos) yra prijungti prie ESP32-S3 mikrovaldiklio. Pats mikrovaldiklis turi WiFi modulį, kuris yra integruotas ir, naudojantis MQTT, pagal parašytą programinį kodą, perduoda informaciją į stebėjimo įrankį. Naudotojo sistemoje yra prijungtas vairas, pedalai, pavarų svirtis ir kamera. Šie elementai perduoda duomenis apie savo pozicijas ir scenarijaus eigą į tą stebėjimo įrankį. Stebėjimo įrankis suformuoja duomenų paketus, kurie, per MQTT, yra perduodami į duomenų bazę. Atlikus scenarijų, stebėtojas atsišiuoia scenarijaus duomenų rinkinį ir įkelia į analizės įrankį, norint pamatyti scenarijaus duomenis.



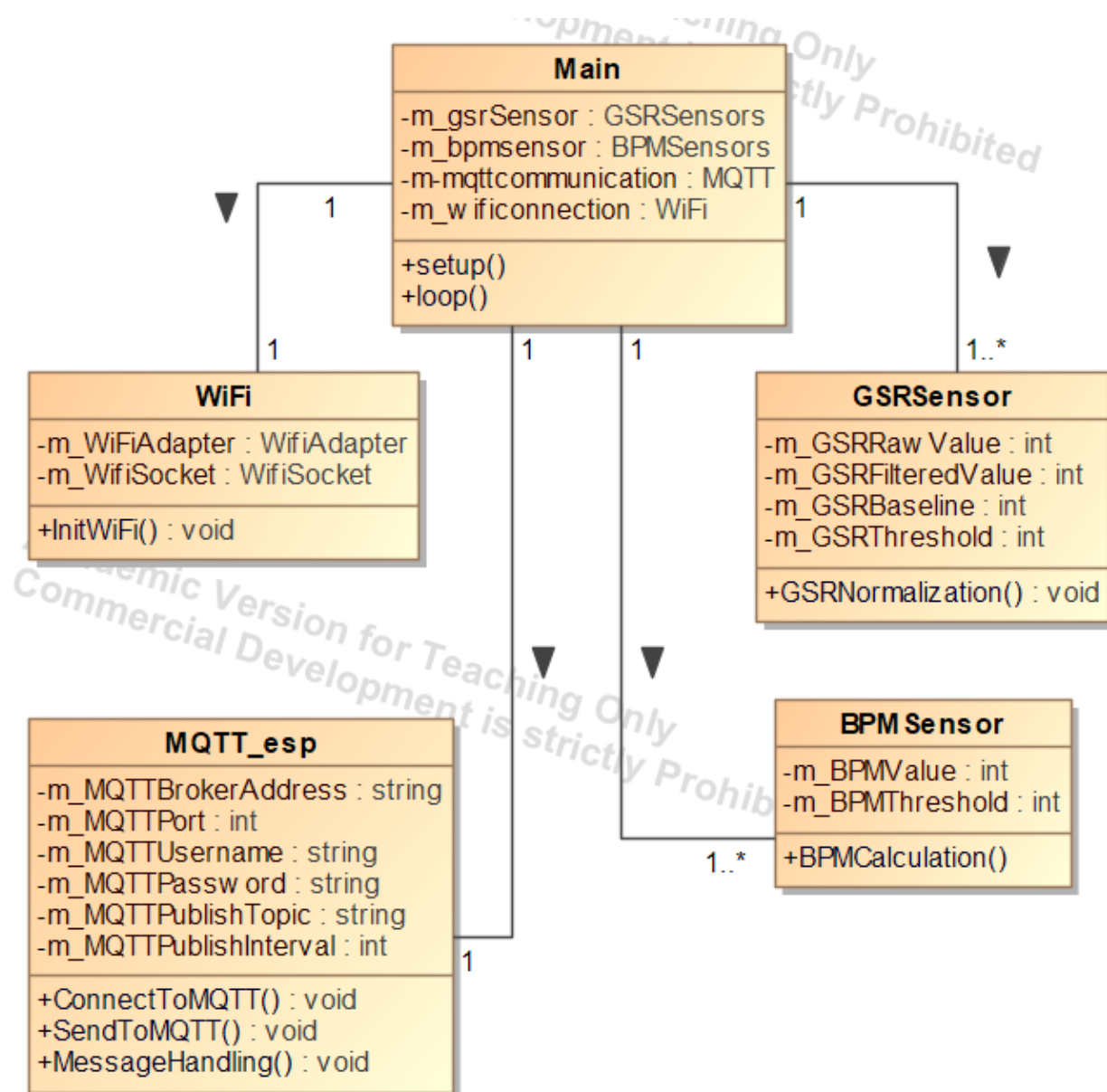
3 pav. Ekstremalaus vairavimo simulatoriaus programinės įrangos architektūra

4.1.2. Ekstremalaus vairavimo simulatoriaus programinio kodo sandara

Mikrovaldiklio programinį kodą sudaro šios klasės:

- *Main* klasė yra pagrindinė klasė, kuri yra atsakinga už jutiklių suteikiamų duomenų apdorojimą;
- *MQTT* klasė yra skirta MQTT komunikacijos vykdymui;
- *WiFi* klasė yra skirta WiFi tinklui prisijungti prie interneto, kad perduoti duomenis MQTT protokolu;
- *GSRSensor* klasė yra skirta odos varžos gaunamų duomenų apdorojimui ir apskaičiavimui į aiškiai nusakomą vertę;
- *BPMSensor* klasė yra skirta širdies ritmo apdorojimui ir apskaičiavimui į aiškiai nusakomą vertę.

Klasių sąryšiai ir struktūros yra atvaizduojamos mikrovaldiklio klasių diagramoje (žr. 4 pav.).

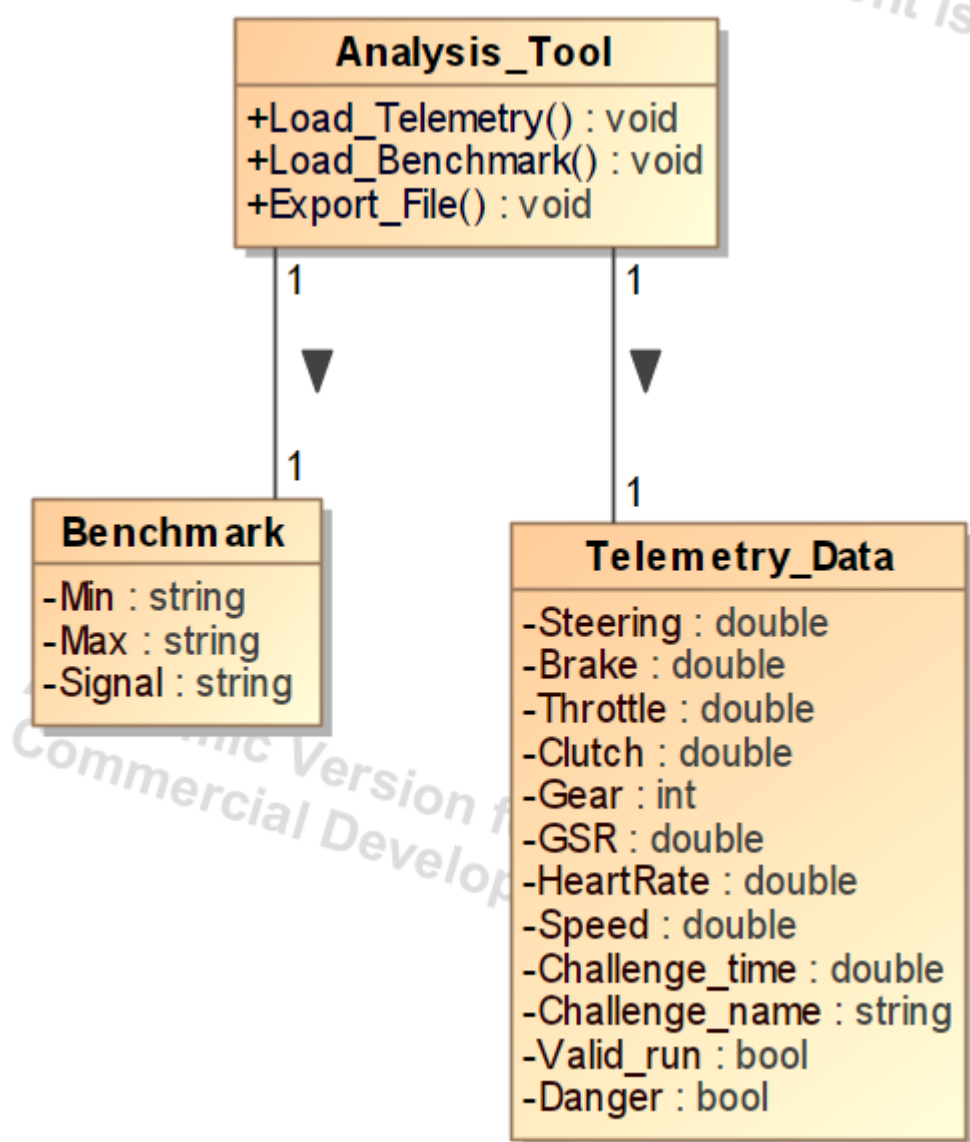


4 pav. Ekstremalaus vairavimo simulatoriaus mikrovaldiklio programinio kodo klasės

Analizės įrankio programinį kodą sudaro šios klasės:

- *Analysis_tool* klasė yra pagrindinė klasė, kuri atsakinga už atliekamus veiksmus, kaip duomenų rinkinio failo įkėlimas, duomenų palyginimo failo įkėlimas ir bendro rezultato eksportavimas.
- *Benchmark* klasė yra atsakinga taisyklingai nustatyti duomenų palyginimo failo kintamuosius ir jų ribas.
- *Telemetry_Data* klasė yra skirta duomenų rinkinio skirtingų rūšių duomenis nuskaityti ir įkelti į lenteles.

Klasių sąryšiai ir struktūros yra atvaizduojamos analizės įrankio klasių diagramoje (žr. **5 pav.**)



5 pav. Ekstremalaus vairavimo simulatoriaus analizės įrankio klasės

Duomenų bazė susideda iš vienos lentelės, kadangi tai yra laiko eilučių tipo duomenų bazė, kur kiekviena nauja lentelė yra duomenų rinkinys. Duomenų bazės lentelė yra atvaizduojama ryšių diagramoje (žr. **6 pav.**).

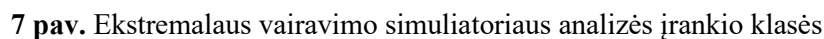
Telemetry_dataset
-Timestamp : datetime
-Valid_run : bool
-Challenge_Name : string
-Challenge_Time : float
-Checkpoint : int
-Speed : int
-Steering : double
-Throttle : double
-Brake : double
-Clutch : double
-Gear : int
-Gaze_Location : string
-HeartRate : int
-GSR : int
-Danger_flag : bool

6 pav. Ekstremalaus vairavimo simulatoriaus duomenų bazės lentelės

Stebėjimo įrankio programinį kodą sudaro šios klasės:

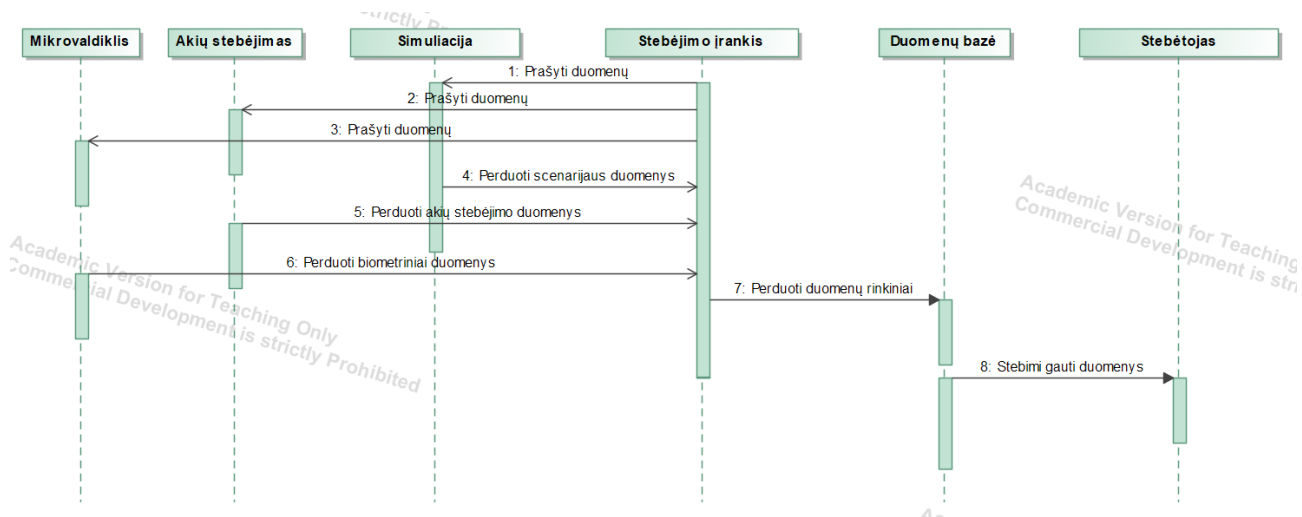
- *Controller* klasė, kurioje yra valdoma stebėjimo įrankio grafinė sąsaja, pranešimų generavimas, komponentų prijungimo ir atjungimo logika, duomenų siuntimo įjungimas ir išjungimas, duomenų išsaugojimas;
- *Eye_Tracking* klasė yra atsakinga už taisyklingą duomenų perdavimo įjungimą, išjungimą ir valdymą;
- *ESP32* klasė yra atsakinga už taisyklingą informacijos apdorojimą iš mikrovaldiklio per MQTT protokolą;
- *BeamNG* klasė yra atsakinga už duomenų rinkimą iš „BeamNG“ programinės įrangos;
- *MQTT* klasė yra atsakinga už taisyklingą komunikaciją tarp stebėjimo įrankio ir MQTT brokerio;
- *BeamNG_Packet* klasė yra atsakinga už korektišką duomenų formavimą iš „BeamNG“ programinės įrangos;
- *Vehicle_Data* klasė yra atsakinga už reikalingų kintamųjų apie automobilio duomenis saugojimą;
- *Scenario_Data* klasė yra atsakinga už reikalingų kintamųjų apie scenarijaus duomenis saugojimą;
- *Telemetry_Event* klasė yra atsakinga už duomenų apdorojimą iš fizinių įrenginių;
- *Biometric* klasė yra atsakinga už biometrinių duomenų nuskaitymą;
- *Gaze_Packet* klasė yra atsakinga už akių stebėjimo zonos nuskaitymą;
- *Telemetry* klasė yra atsakinga už taisyklingą duomenų atvaizdavimą stebėjimo įrankyje;
- *Bio_Data* klasė yra atsakinga už taisyklingą biometrinių duomenų atvaizdavimą stebėjimo įrankio *Telemetry* klasėje;
- *Health_Status* klasė yra atsakinga už taisyklingą programinių įrangų ir komunikacijos būsenos atvaizdavimą stebėjimo įrankio *Telemetry* klasėje;

- Klasių sąryšiai ir struktūros yra atvaizduojamos analizės įrankio klasių diagramoje (žr. 7 pav.)



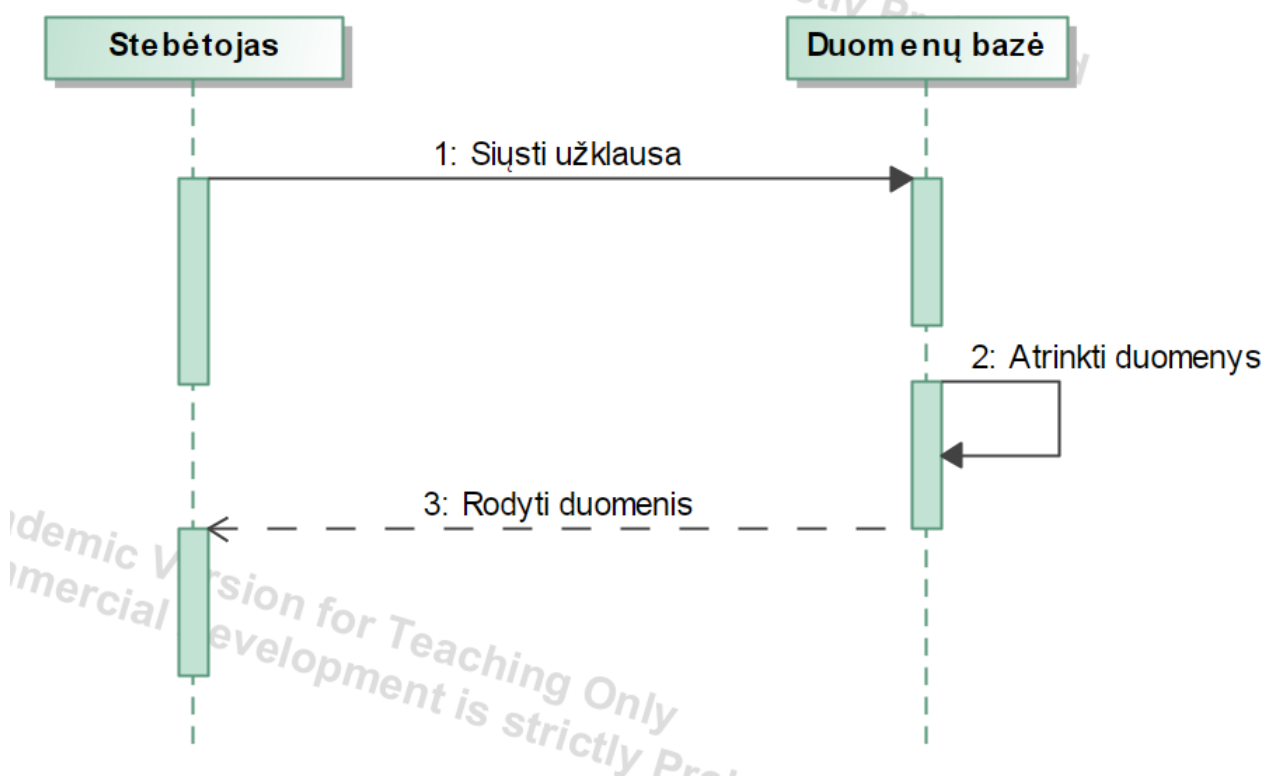
Šiame skyriuje yra detalizuojamos ekstremalaus vairavimo simulatoriaus veiklų diagramos, naudojantis sekų diagramomis.

37



8 pav. Ekstremalaus vairavimo simulatoriaus simuliacijos stebėsenos sekų diagrama (S.1)

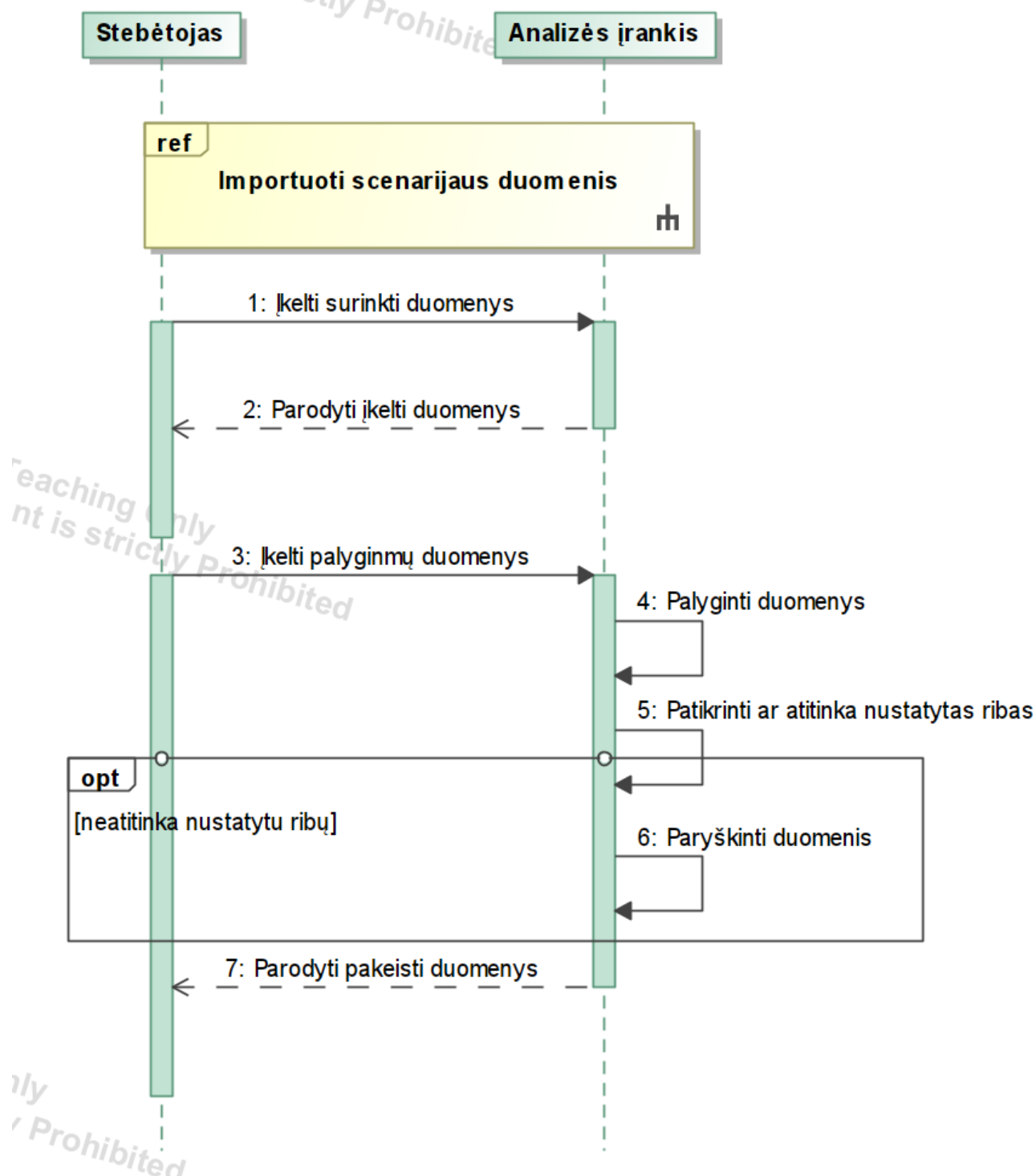
Stebėtojas po scenarijaus atlikimo siunčia užklausą dėl norimo duomenų rinkinio. Duomenų bazė atlieka duomenų atrinkimą pagal norimą duomenų rinkinio identifikatorių ir leidžia stebėtojui matyti, kokie duomenys yra įtraukti į norimą duomenų rinkinį. Ši veiksmų seka yra atvaizduota sekų diagramoje (žr. **9 pav.**).



9 pav. Ekstremalaus vairavimo simulatoriaus duomenų gavimo iš duomenų bazės sekų diagrama (S1.1)

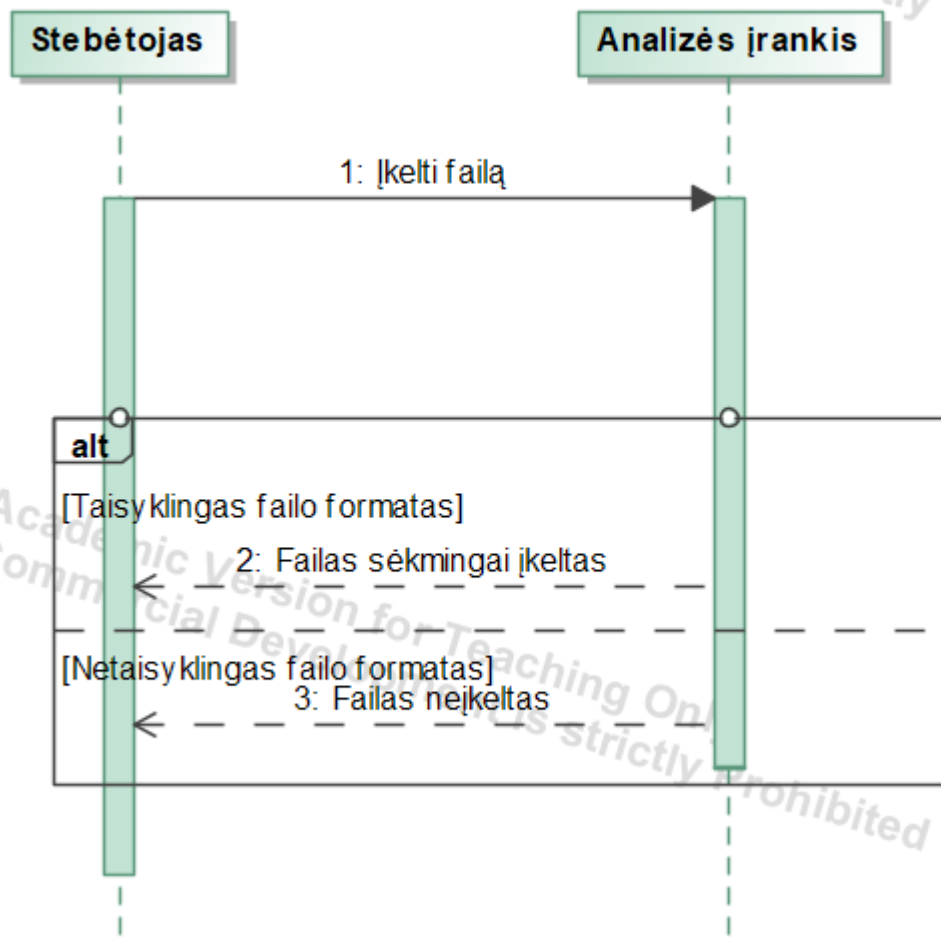
Sudarytas duomenų rinkinys, stebėtojo prašymu, yra atsiunčiamas į jo kompiuterį. Atsiųsti duomenys yra įkeliami į analizės įrankį, siekiant atlikti tikslesnę duomenų peržiūrą. Stebėtojas pagal atitinkamą scenarijų gali įkelti palyginimo duomenų failą į analizės įrankį, pagal kurį bus tikrinami esami

duomenys. Atlikus duomenų palyginimą, rezultatas yra parodomas analizės įrankyje stebėtojiui. Ši veiksmų seka yra atvaizduota sekų diagramoje (žr. 10 pav.).



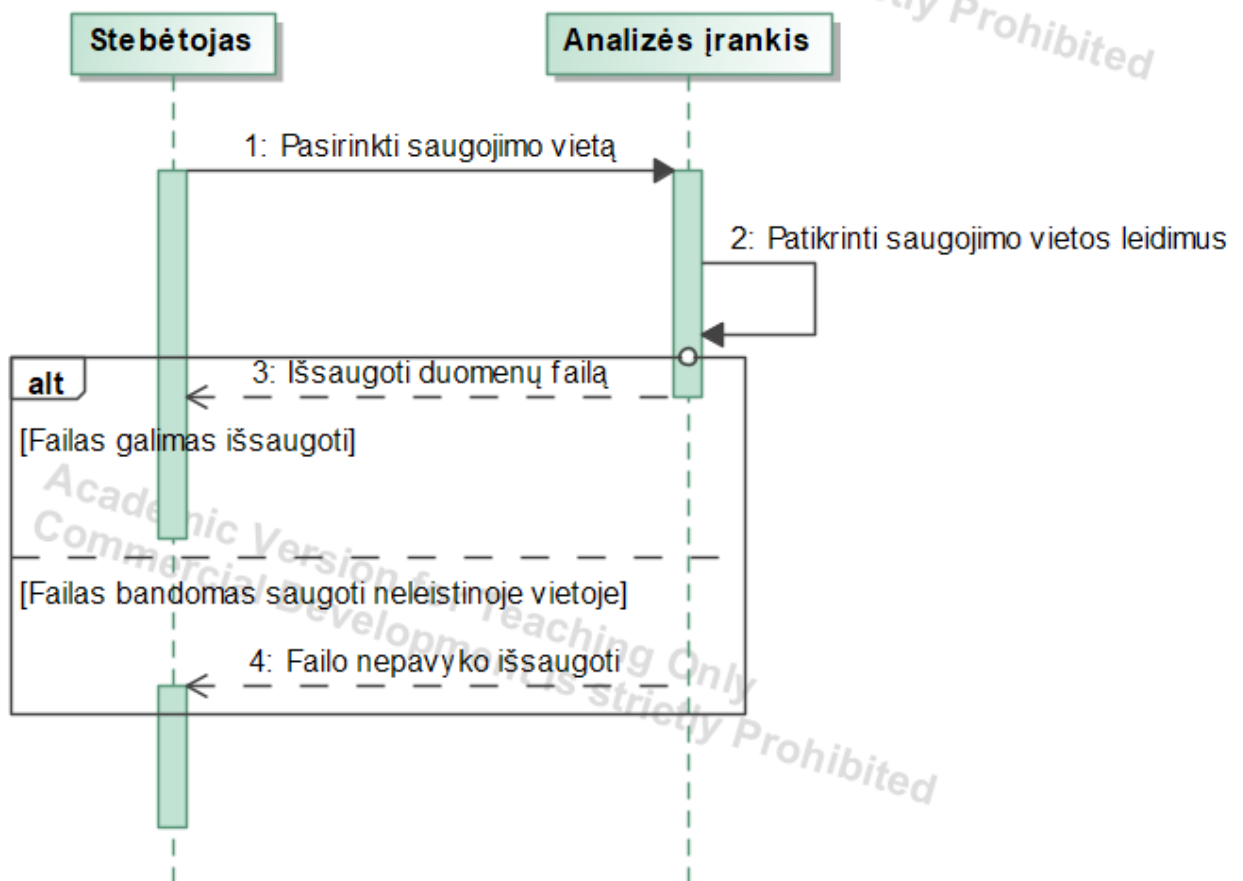
10 pav. Ekstremalaus vairavimo simulatoriaus gautų scenarijaus duomenų analizės sekų diagrama (S.2)

Stebėtojas gali įkelti norimą duomenų rinkinio failą į analizės įrankį. Jei įkeliamas failas yra taisyklingo formato, jis įkeliamas sėkmingai. Jei failas yra netaisyklingo formato, failas yra neįkeliamas į analizės įrankį. Ši veiksmų seka yra atvaizduota sekų diagramoje (žr. 11 pav.).



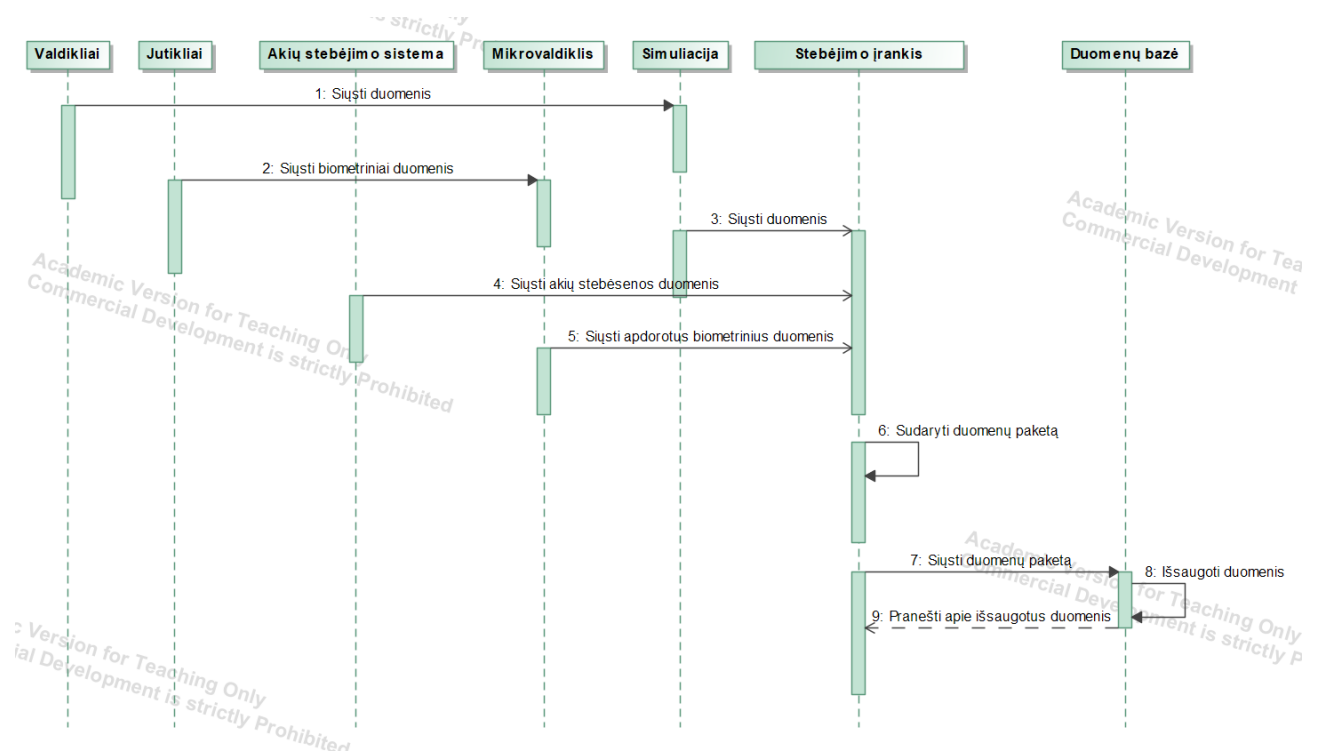
11 pav. Ekstremalaus vairavimo simulatoriaus gautų scenarijaus duomenų analizės sekų diagrama (S.3)

Stebėtojas gali išsaugoti išanalizuotą duomenų rinkinio failą iš analizės įrankio. Išsaugojimui stebėtojas turi pasirinkti tinkamą vietą failo saugojimui. Jei pasirinktoje vietoje failas gali būti saugomas, failas yra įrašomas. Jei pasirinktoje vietoje nėra pakankamai leidimų, failas yra neišsaugomas ir prašoma pakeisti saugojimo vietą. Ši veiksmų seka yra atvaizduota sekų diagramoje (žr. 12 pav.).



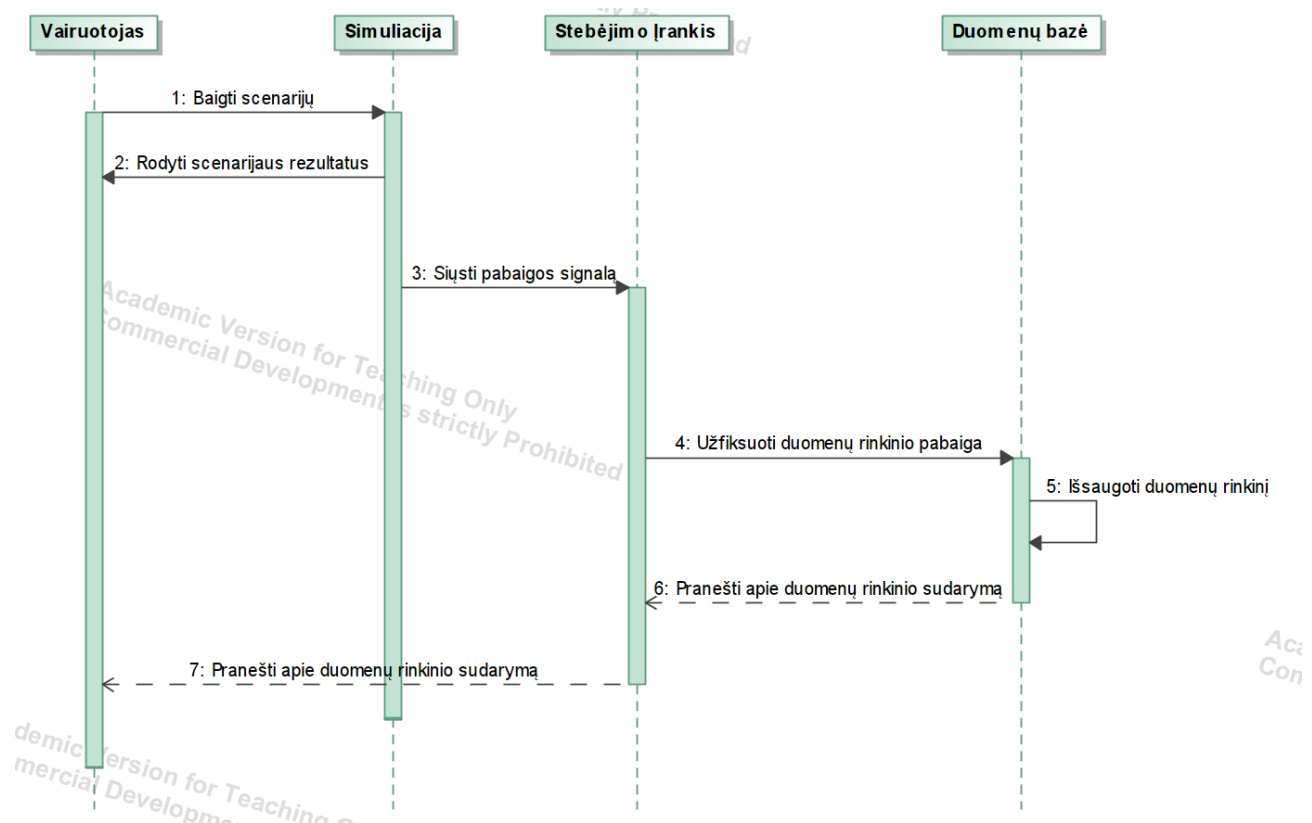
12 pav. Ekstremalaus vairavimo simulatoriaus gautų scenarijaus duomenų analizės sekų diagrama (S.4)

Scenarijaus metu, valdikliai siunčia duomenis į simuliacijos aplinką. Jutikliai siunčia duomenis į mikrovaldiklį. Simuliacija, akių stebėjimo sistema ir mikrovaldiklis siunčia duomenis į stebėjimo įrankį. Stebėjimo įrankis atlieka duomenų paketo sudarymą ir jo persiuntimą į duomenų bazę su duomenų rinkinio identifikatoriumi. Duomenų bazė išsaugoja duomenų paketus į atitinkamą duomenų rinkinį pagal pateikiamą duomenų rinkinio identifikatorių. Ši veiksmų seka yra atvaizduota sekų diagramoje (žr. **13 pav.**).



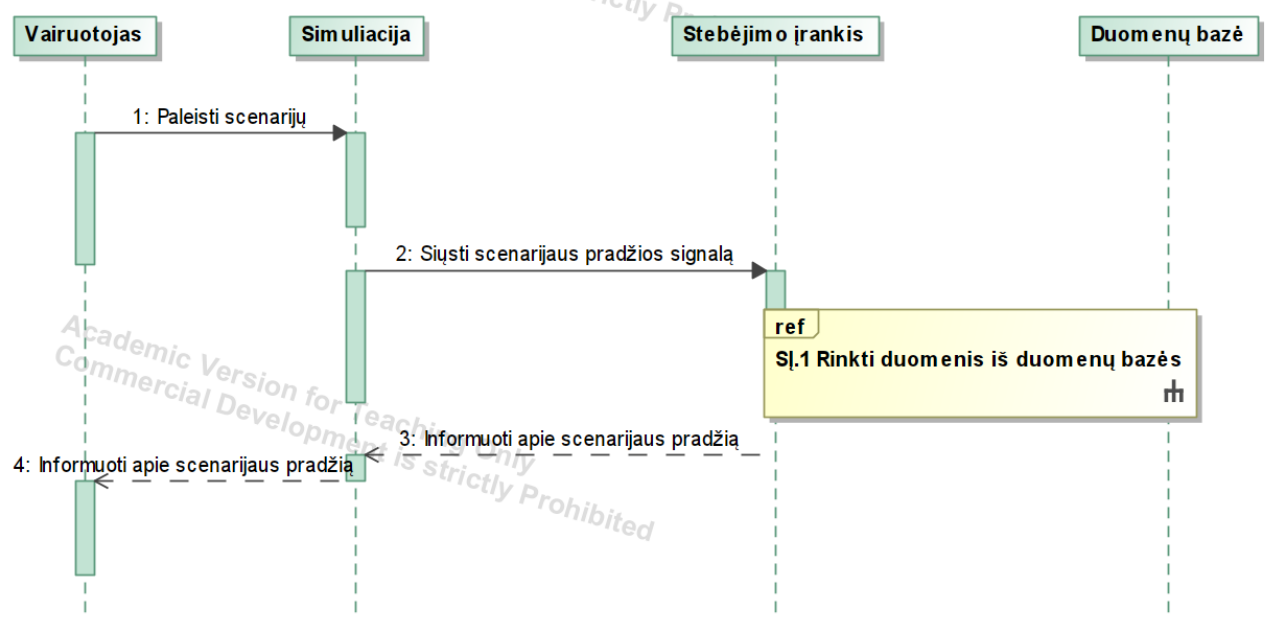
13 pav. Ekstremalaus vairavimo simulatoriaus duomenų perdavimo sekų diagrama (SP.1)

Simuliacijos aplinka, kuomet vairuotojas baigia scenarijaus vykdymą, siunčia užbaigimo signalą į stebėjimo įrankį. Stebėjimo įrankis išsiunčia duomenų rinkinio pabaigos signalą, taip užbaigdamas duomenų paketų įrašymą į sugeneruotą duomenų rinkinį. Duomenų bazė, gavusi duomenų rinkinio pabaigos signalą išsaugo esamus duomenis sugeneruotame duomenų rinkinyje ir perduoda informaciją apie tai, jog gali priimti sekančius duomenų paketus į naujai sugeneruojamą duomenų rinkinį. Atlikus šiuos veiksmus, simuliacijos aplinka pateikia vairuotojui informaciją apie scenarijaus rezultatus. Ši veiksmų seka yra atvaizduota sekų diagramoje (žr. **14 pav.**).



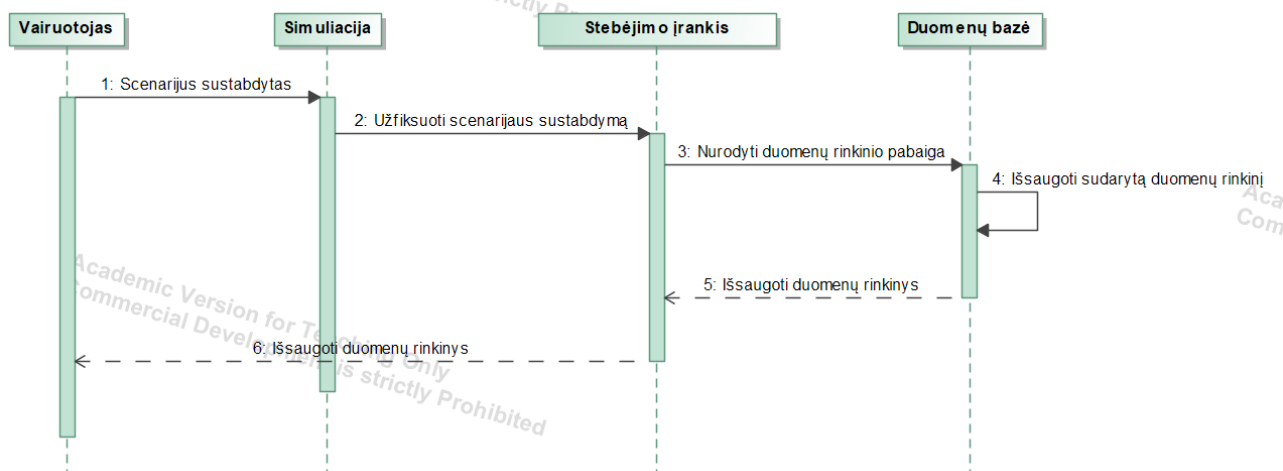
14 pav. Ekstremalaus vairavimo simulatoriaus scenarijaus atlikimo sekų diagrama (V.1)

Vairuotojui pradėjus scenarijų, simuliacija siunčia signalą į stebėjimo įrankį, kad scenarijus yra pradedamas, reikalinga pradėti formuoti duomenų paketus ir juos saugoti į duomenų bazę pagal atitinkamai sugeneruojamą duomenų rinkinio indikatorius. Duomenų bazė gavusi signalą apie scenarijaus pradžią, sukuria duomenų rinkinio lentelę pagal pateiktą identifikatorių ir pradeda duomenų paketų saugojimą. Simuliacija pradeda scenarijaus eigą. Ši veiksmų seka yra atvaizduota sekų diagramoje (žr. 15 pav.).



15 pav. Ekstremalaus vairavimo simulatoriaus scenarijaus pradžios sekų diagrama (V.2)

Vairuotojui sustabdžius scenarijų, simuliacija išsiunčia scenarijaus sustabdymo pranešimą į stebėjimo įrankį. Stebėjimo įrankis, gavęs šį pranešimą, nurodo, jog duomenų bazėje esantis duomenų rinkinys yra baigtas. Duomenų bazė išsaugo esamą duomenų rinkinį. Ši veiksmų seka yra atvaizduota sekų diagramoje (žr. 16 pav.).



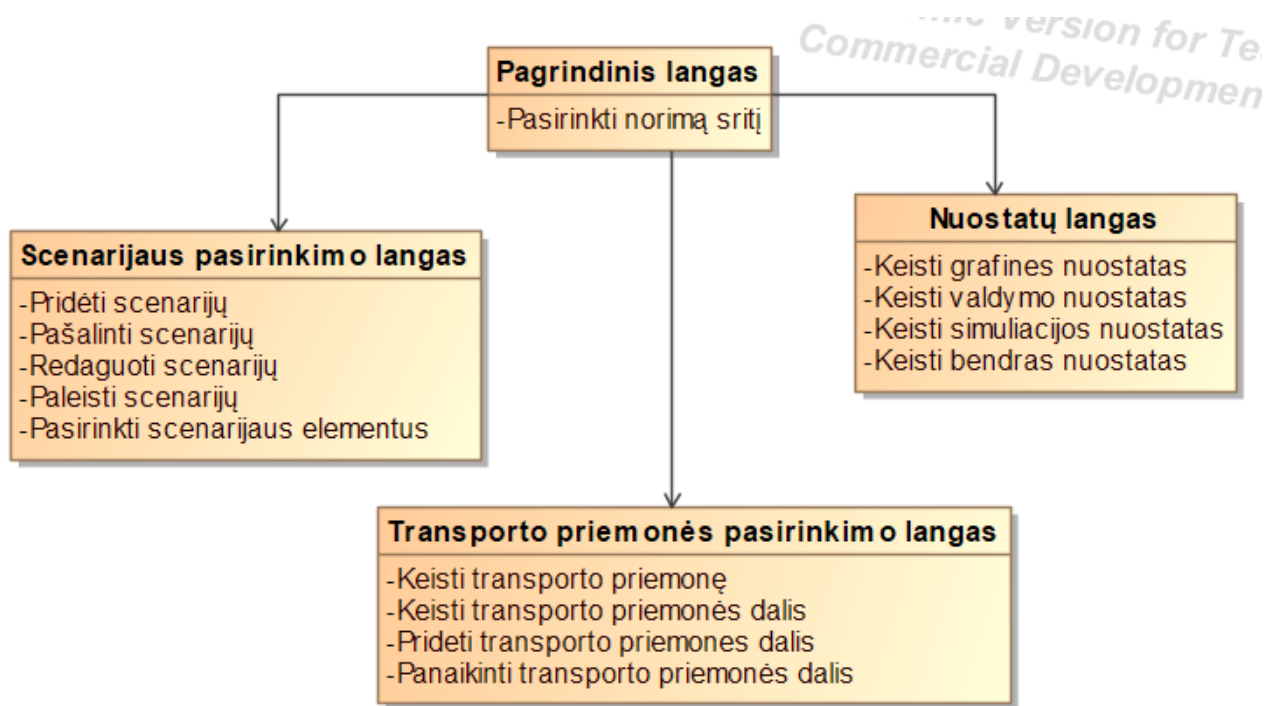
16 pav. Ekstremalaus vairavimo simulatoriaus scenarijaus stabdymo sekų diagrama (V.3)

4.1.4. Ekstremalaus vairavimo simulatoriaus naudotojo sąsajos projektas

Ekstremalaus vairavimo simulatoriaus naudotojo sąsaja yra skirstoma į 3 pagrindines dalis. Simuliacijos aplinka, kurioje yra:

- scenarijaus pasirinkimo langas – šiame lange yra pasirenkamas scenarijus kuris bus vykdomas simuliacijos metu;
- transporto priemonės pasirinkimo langas – šiame lange suteikiama galimybė interaktyviai išbandyti galimas transporto priemonių kombinacijas, kurios gali būti naudojamos scenarijuje;
- nuostatų langas – šiame lange yra valdomos simuliacijos bendros nuostatos, valdiklių nuostatos, grafinės nuostatos, fizikos nuostatos.

Simuliacijoje pasirenkami langų sąryšiai pateikiami tolimesnėje diagramoje (žr. 17 pav.)

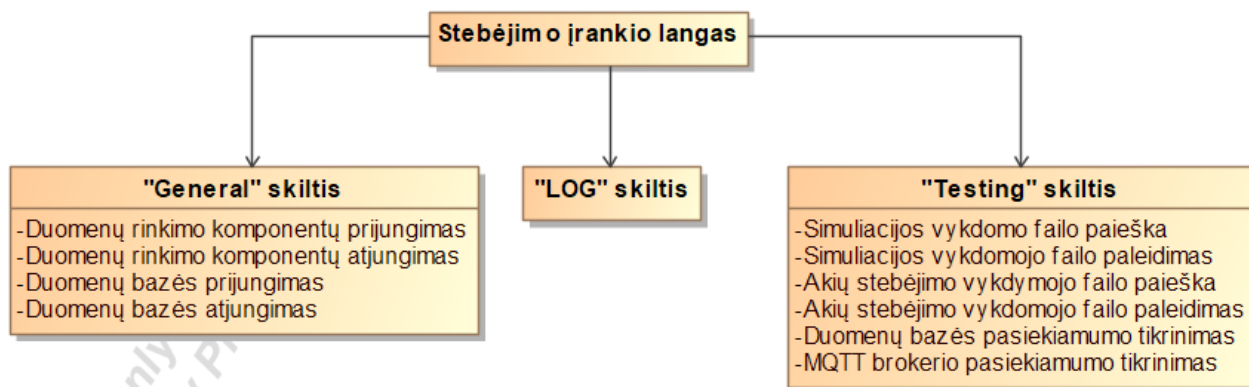


17 pav. Ekstremalaus vairavimo simulatoriaus simuliacijos naudotojo sąsaja

Stebėjimo įrankis, kurio pagrindiniame lange yra:

- bendroji skiltis („General“), kurioje yra rodoma informacija apie renkamus duomenis, dabartinę duomenų rinkimo komponentų prisijungimo būseną, duomenų bazės informacija apie paskutinį duomenų rinkinį nuo paleidimo;
- pranešimų skiltis („LOG“), kurioje yra rodoma bendra informacija apie tai, kas šiuo metu vyksta stebėjimo įrankio aplinkoje ir detalūs komunikacijos pranešimai;
- tikrinimo skiltis („Testing“), kurioje yra rodoma derinimo ir tikrinimo informacija susijusi su simuliacijos aplinkos paleidimu, akių stebėjimo paleidimu, duomenų bazės komunikacijos nustatymu, MQTT komunikacijos nustatymu. Taip yra siekiant užtikrinti taisyklingą veikimą prieš bandant vykdyti scenarijus.

Simuliacijoje pasirenkami langų sąryšiai pateikiami tolimesnėje diagramoje (žr. **18 pav.**)

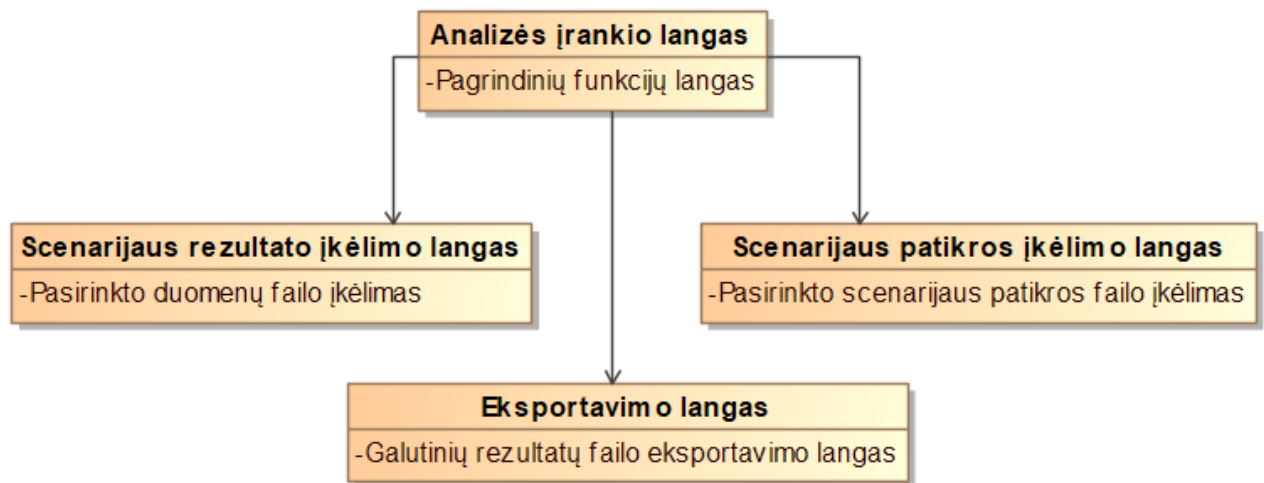


18 pav. Ekstremalaus vairavimo simulatoriaus simuliacijos naudotojo sąsaja

Analizės įrankis, kuriame yra:

- pagrindinis analizės įrankio langas – čia vaizduojami visi galimi pasirinkimai ir nustatymai norint atlikti norimus veiksmus;
- scenarijaus rezultato įkėlimo langas – langas skirtas pasirinkti scenarijaus duomenų rinkinio įkėlimą į analizės įrankį;
- scenarijaus patikros įkėlimo langas – langas skirtas pasirinkti scenarijaus etalono duomenų rinkinio įkėlimą, norint patikrinti gautus scenarijaus rezultatus;
- eksportavimo langas – langas skirtas pasirinkti kur išsaugomi išanalizuoti rezultatai po apdorojimo ir atlikto palyginimo.

Stebėtojai suteikiamų valdymo langų sąryšiai pateikiami tolimesnėje diagramoje (žr. **19 pav.**)



19 pav. Ekstremalaus vairavimo simulatoriaus stebėtojo naudotojo sąsajos projektas

4.1.5. Ekstremalaus vairavimo simulatoriaus techninės įrangos projektas

Ekstremalaus vairavimo simulatoriaus techninės įrangos principinėje schemoje vaizduojami reikiami fiziniai sujungimai tarp skirtingų simulatoriaus fizinių komponentų (žr. **20 pav.**).

5. Ekstremalaus vairavimo simulatoriaus prototipo realizacija ir testavimas

Šiame skyriuje aprašomos programinės ir techninės įrangos prototipo realizacija ir testavimas. Prototipo realizacija atliekama pagal projektavimo skyrių. Atlikta prototipo realizacija pateikiama aprašymais, paveikslėliais ir diagramomis. Testavimas atliekamas pagal sudarytą testavimo planą aprašytame sekančiame poskyryje.

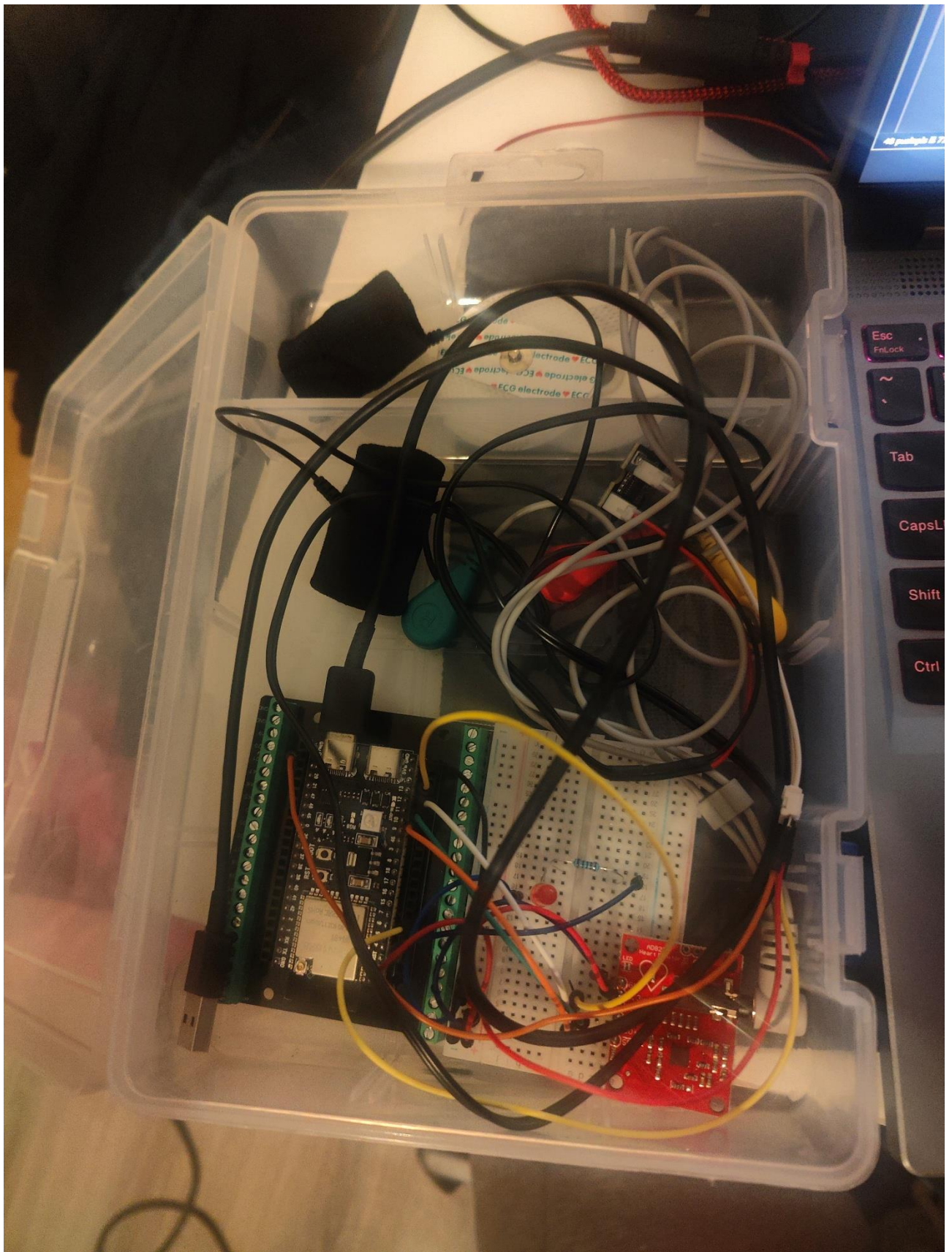
5.1. Ekstremalaus vairavimo simulatoriaus prototipo realizacija ir testavimas

Prototipo programinės įrangos realizavimo metu sukurtos dvi atskiros C# pagrindo programos su vartotojo sąsajomis. Pirmoji skirta užtikrinti veikiančią komunikaciją tarp sensorių, jutiklių, „BeamNG.Drive“ programinės įrangos bei „Beam Eye Tracker“ programinės įrangos, duomenų surinkimu, MQTT paketų konstravimu ir jų perdavimu į duomenų bazę. Antroji skirta apdoroti atsiųstą duomenų failą, kuriame saugomi duomenys apie įvykdytą scenarijų ir palyginti su etaloniniais matmenimis, paruoštais pagal teorinius žingsnius. Programų įgyvendinimui panaudota „Visual Studio Code“ programavimo aplinka.

Mikrovaldiklio programinei įrangai panaudota „Visual Studio Code“ programavimo aplinka, „PlatformIO“ ir „ESP-IDF“ karkasas. Realizacijos metu sukonfigūruotas mikrovaldiklis, paruošti ir sukonfigūruoti sensoriai. Parašytas programinis kodas, kuris atlieka suprojektuotus panaudojimo atvejus.

5.1.1. Ekstremalaus vairavimo simulatoriaus techninės įrangos projektas

Ekstremalaus vairavimo simulatoriaus techninė įranga realizuota pagal techninės įrangos projektą. Prototipo pagrindinei plokštei panaudota eksperimentinė plokštė maketavimui, jungiamieji laideliai bei atitinkami komponentai pagal techninės įrangos projektą. Žemiau matomas sujungtas prototipas paruoštas naudojimui ir įdėtas į transportavimo dėžutę (žr. **pav. 21**).



21 pav. Paruoštas prototipas su USB jungtimi

Kiti techninės įrangos komponentai jungiami prie kompiuterio, kuriame bus vykdomi vairavimo scenarijai.

5.2. Ekstremalaus vairavimo simulatoriaus prototipo testavimas

Šiame skyriuje aprašomas prototipo testavimo planas ir rezultatai. Testavimo metu patikrinama, ar sistemos prototipas atitinka išskeltus funkcinius ir nefunkcinius reikalavimus.

5.2.1. Ekstremalaus vairavimo simulatoriaus prototipo testavimo planas

Ekstremalaus vairavimo simulatoriaus testavimui pasirenkamas funkcinis testavimas. Šis testavimas tikrina, ar sistema atitinka funkcinius ir nefunkcinius reikalavimus ir ar ji veikia taip, kaip yra numatyta.

Funkcinio testavimo metu ekstremalaus vairavimo prototipas bandomas realiomis sąlygomis. Šie testai atliekami rankiniu būdu, naudojantis prototipo technine ir programine įranga.

Sistemos testavimui turi būti paruošta tinkama testavimo aplinka scenarijaus vykdymo kompiuteryje:

- turi būti įdiegta „Beam Eye Tracker“ programinė įranga;
- turi būti įdiegta „BeamNG.Drive“ programinė įranga.

Ekstremalaus vairavimo prototipo testavimui buvo naudojami 2 asmeniniai kompiuteriai. Pirmasis kompiuteris atlieka scenarijaus vykdymo veiksmus bei scenarijaus analizės veiksmus, antrasis kompiuteris atlieka serverio vaidmenį, kuriame yra paleista MQTT brokeris, „InfluxDB“, „Telegraf“ ir „Grafana“.

5.2.2. Ekstremalaus vairavimo simulatoriaus prototipo testavimo planas

Šiame poskyryje pateikiamos testavimo lentelės, kurios aprašo testavimo atvejį, žingsnius, numatomus rezultatus ir gautus rezultatus. Taip pat pateikiamos testavimo išvados ar pataisymai.

18 lentelė. „BeamNG.Drive“ vykdomojo failo pasirinkimo ir paleidimo testavimo procesas

Testavimo atvejis	„BeamNG.Drive“ vykdomojo failo pasirinkimas ir paleidimas			
Aprašymas	Testuojamas „BeamNG.Drive“ vykdomojo failo nurodymas ir įjungimas naudojantis stebėjimo įrankiu			
Sąlyga – prieš	Stebėjimo įrankis neturi duomenų apie „BeamNG.Drive“ vykdomojo failo vietą ir negali jo paleisti			
Sąlyga – po	Stebėjimo įrankis įrašo „BeamNG.Drive“ vykdomojo failo vietą ir jį paleidžia			
Testavimo žingsniai				
Nr.	Duomenys	Veiksmas	Numatomas rezultatas	Gautas rezultatas
1.	-	Stebėjimo įrankyje pasirenkama „Testing“ skiltis	Atidaromas nuostatų tikrinimo langas	Nuostatų langas atidarytas
2.	-	„APPS SETUP“ grupėje, pirmoje eilutėje, nuspaudžiamas „Browse“ mygtukas	Atidaromas langas, kuriame atliekamas vykdomojo failo pasirinkimas	Atidarytas vykdomojo failo pasirinkimo langas

3.	<Naudotojo BeamNG.drive įrašytas aplankalas>\BeamNG.Drive\Bin64\BeamNG.drive.x64.exe	Pasirenkamas „BeamNG.drive.x64.exe“ failas ir spaudžiama atidaryti	Pasirinkto vykdomojo failo esama vieta išsaugojama paleidimui stebėjimo įrankyje	Pasirinkto vykdomojo failo esama vieta išsaugota paleidimui stebėjimo įrankyje
4.	-	Nuspaudžiamas mygtukas „Launch“	Pasirinktas vykdomasis failas yra paleidžiamas	Pasirinktas vykdomasis failas buvo paleistas

Atlikus „BeamNG.Drive“ paleidimo testavimą matoma, kad „BeamNG.Drive“ vykdomasis failas yra nurodomas taisyklingai ir vykdomojo failo paleidimas veikia korektiškai. Atliktas alternatyvus testas, kuriame vykdyti tie patys žingsniai, tačiau nurodomas nekorektiškas failas. Atliekant failo paleidimą buvo atidarytas pranešimas, jog tai nėra taisyklingas „BeamNG.Drive“ vykdomojo failo pavadinimas. „BeamNG.Drive“ vykdomojo failo vietos nurodymas ir paleidimas veikia taisyklingai.

19 lentelė. „Beam Eye Tracker“ vykdomojo failo pasirinkimo ir paleidimo testavimo procesas

Testavimo atvejis	„Beam Eye Tracker“ vykdomojo failo pasirinkimas ir paleidimas			
Aprašymas	Testuojamas „Beam Eye Tracker“ vykdomojo failo nurodymas ir įjungimas naudojantis stebėjimo įrankiu			
Sąlyga – prieš	Stebėjimo įrankis neturi duomenų apie „Beam Eye Tracker“ vykdomojo failo vietą ir negali jo paleisti			
Sąlyga – po	Stebėjimo įrankis įrašo „Beam Eye Tracker“ vykdomojo failo vietą ir jį paleidžia			
Testavimo žingsniai				
Nr.	Duomenys	Veiksmai	Numatomas rezultatas	Gautas rezultatas
1.	-	Stebėjimo įrankyje pasirenkama „Testing“ skiltis	Atidaromas nuostatų tikrinimo langas	Nuostatų langas atidarytas
2.	-	„APPS SETUP“ grupėje, antroje eilutėje, nuspaudžiamas „Browse“ mygtukas	Atidaromas langas, kuriame atliekamas vykdomojo failo pasirinkimas	Atidarytas vykdomojo failo pasirinkimo langas
3.	<Naudotojo „Beam Eye Tracker“ įrašytas aplankalas>\BeamEyeTracker \ BeamEyeTracker.exe	Pasirenkamas „BeamEyeTracker.exe“ failas ir spaudžiama atidaryti	Pasirinkto vykdomojo failo esama vieta išsaugojama paleidimui stebėjimo įrankyje	Pasirinkto vykdomojo failo esama vieta išsaugota paleidimui stebėjimo įrankyje
4.	-	Nuspaudžiamas mygtukas „Launch“	Pasirinktas vykdomasis failas yra paleidžiamas	Pasirinktas vykdomasis failas buvo paleistas

Atlikus „Beam Eye Tracker“ paleidimo testavimą matoma, kad „Beam Eye Tracker“ vykdomasis failas yra nurodomas taiskykliai ir vykdomojo failo paleidimas veikia korektiškai. Atliktas alternatyvus testas, kuriame vykdyti tie patys žingsniai, tačiau nurodomas nekorektiškas failas. Atliekant failo paleidimą buvo atidarytas pranešimas, jog tai nėra taiskykliai „Beam Eye Tracker“ vykdomojo failo pavadinimas. „Beam Eye Tracker“ vykdomojo failo vietos nurodymas ir paleidimas veikia taiskykliai.

20 lentelė. Reikalingų programinių įrangų procesų tikrinimo testavimo procesas

Testavimo atvejis	„Beam Eye Tracker“ ir „BeamNG.Drive“ paleistų vykdomųjų failų tikrinimas			
Aprašymas	Testuojama funkcija tikrinanti ar paleistų programinių įrangų vykdomųjų failų tikrinimas veikia tinkamai			
Sąlyga – prieš	Nėra rodoma duomenų kiek procesų veikia pagal nurodytus pavadinimus			
Sąlyga – po	Rodoma, kiek procesų veikia pagal nurodytus pavadinimus			
Testavimo žingsniai				
Nr.	Duomenys	Veiksmas	Numatomas rezultatas	Gautas rezultatas
1.	-	Stebėjimo įrankyje pasirenkama „Testing“ skiltis	Atidaromas nuostatų tikrinimo langas	Nuostatų langas atidarytas
2.	-	„APPS SETUP“ grupėje nuspaudžiamas „Test Process“ mygtukas	Prie „BeamNG:“ ir „Beam Eye Tracker:“ tekstų yra parodomas skaičius, kiek nurodytos programos procesų yra paleistų	Prie „BeamNG:“ ir „Beam Eye Tracker:“ tekstų rodomas skaičius, kiek nurodytos programos procesų yra paleistų

Atlikus procesų tikrinimo testavimą matoma, kad „BeamNG.Drive“ ir „Beam Eye Tracker“ vykdomieji procesai yra nurodomi taiskykliai. Po „BeamNG.Drive“ paleidimo matoma, jog yra daugiau negu vienas veikiantis procesas su tokiu pavadinimu. Taip yra dėl pačios „BeamNG.Drive“ programinės įrangos paleidimo. Atlikti alternatyvūs testai, kurių metu viena iš programinių įrangų nebuvo paleista arba abi nebuvo paleistos. Po to buvo tikrintas paleistų procesų kiekis. Nepriklausomai nuo to ar viena iš programinių įrangų nebuvo paleista ar abi, atitinkamose skiltyse, vietoje skaičių, buvo rodomas tekstas „NOT RUNNING“, indikuojantis, jog procesai yra neveikiantys ar nepaleisti.

21 lentelė. „InfluxDB“ duomenų bazės pasiekiamumo testavimo procesas

Testavimo atvejis	„InfluxDB“ duomenų bazės pasiekiamumo tikrinimas
Aprašymas	Testuojama funkcija tikrinanti ar „InfluxDB“ duomenų bazė yra pasiekiamas su nurodytais duomenimis
Sąlyga – prieš	„InfluxDB“ būseną nėra rodoma „Testing“ skiltyje
Sąlyga – po	„InfluxDB“ būseną yra rodoma „Testing“ skiltyje
Testavimo žingsniai	

Nr.	Duomenys	Veiksmas	Numatomas rezultatas	Gautas rezultatas
1.	-	Stebėjimo įrankyje pasirenkama „Testing“ skiltis	Atidaromas nuostatų tikrinimo langas	Nuostatų langas atidarytas
2.	IP: 192.168.1.135 Port: 8181 Database: Vairavimo_simulatorius Token: apiv3_	„INFLUXDB SETUP“ grupėje užpildomi „IP“, „Port“, „Database“, „Token“ laukai	Taisyklingai užpildyti „InfluxDB“ duomenų laukai	Taisyklingai užpildyti „InfluxDB“ duomenų laukai
3.	-	„INFLUXDB SETUP“ grupėje nuspaužiamas „Test“ mygtukas	Atliekama funkcija patikrinanti ar duomenų bazė yra pasiekama	Parodoma dabartinė „InfluxDB“ duomenų bazės veikimo būseną – „OK“

Atlikus „InfluxDB“ pasiekiamumo tikrinimą matoma, jog duomenų bazės pasiekiamumas veikia tinkamai. Atliktas alternatyvus testas, kuomet duomenų bazė nebuvo paleista ir buvo atliktas pasiekiamumo tikrinimas. Gauta veikimo būseną „TIMEOUT“, kuri leidžia suprasti, jog duomenų bazė neatsako, taip parodoma, jog ji neveikia arba yra nepasiekama.

22 lentelė. MQTT brokerio pasiekiamumo testavimo procesas

Testavimo atvejis	MQTT brokerio pasiekiamumo tikrinimas			
Aprašymas	Testuojama funkcija tikrinanti ar MQTT brokeris yra pasiekiamas su nurodytais duomenimis			
Sąlyga – prieš	MQTT brokerio būsena nėra rodoma „Testing“ skiltyje			
Sąlyga – po	MQTT brokerio būsena yra rodoma „Testing“ skiltyje			
Testavimo žingsniai				
Nr.	Duomenys	Veiksmas	Numatomas rezultatas	Gautas rezultatas
1.	-	Stebėjimo įrankyje pasirenkama „Testing“ skiltis	Atidaromas nuostatų tikrinimo langas	Nuostatų langas atidarytas
2.	Broker IP: 192.168.1.135 Broker Port: 1883 ESP32 topic: esp32/sensors InfluxDB3 topic: telemetry/session	„MQTT SETUP“ grupėje užpildomi „Broker IP“, „Broker Port, „ESP32 topic“, „InfluxDB3 topic“ laukai	Taisyklingai užpildyti MQTT brokerio duomenų laukai	Taisyklingai užpildyti MQTT brokerio duomenų laukai
3.	-	„MQTT SETUP“ grupėje nuspaužiamas „Test“ mygtukas	Atliekama funkcija patikrinanti ar duomenų bazė yra pasiekama	Parodoma dabartinė MQTT duomenų bazės veikimo būsena – „OK“

Atlikus MQTT brokerio pasiekiamumo tikrinimą matoma, jog brokerio pasiekiamumas veikia tinkamai. Atliktas alternatyvus testas, kuomet MQTT brokeris nebuvo paleistas ir buvo atliktas pasiekiamumo tikrinimas. Gauta veikimo būsena „FAILED“, kuri leidžia suprasti, jog brokeris neatsako, taip parodoma, jog jis neveikia arba yra nepasiekiamas.

23 lentelė. Stebėjimo sistemos funkcijų pasiekiamumo testavimo procesas

Testavimo atvejis	Stebėjimo sistemos funkcijų pasiekiamumo tikrinimas			
Aprašymas	Testuojama, ar stebėjimo sistemos funkcijų pasiekiamumo laukai rodo korektiškas būsenas apie „BeamNG.Drive“, „Beam Eye Tracker“, ESP32 ir MQTT pasiekiamumą			
Sąlyga – prieš	Stebėjimo sistema rodo, jog visos dabartinės sistemos yra atjungtoje būsenoje – „Disconnected“			
Sąlyga – po	Stebėjimo sistema rodo, jog visos dabartinės sistemos yra prijungtoje būsenoje – „Connected“			
Testavimo žingsniai				
Nr.	Duomenys	Veiksmas	Numatomas rezultatas	Gautas rezultatas
1.	-	Stebėjimo įrankyje pasirenkama General skiltis	Atidaromas bendrinis sistemų būsenų langas	Bendrinis sistemų būsenų langas atidarytas
2.	-	Grupėje „SYSTEM STATUS“ nuspaudžiamas „Connect“ mygtukas	Atliekamos funkcijos, prisijungiančios prie „BeamNG.Drive“, „Beam Eye Tracker“, ESP32 mikrovaldiklio ir MQTT brokerio	Sėkmingai prisijungta prie „BeamNG.Drive“, „Beam Eye Tracker“, ESP32 mikrovaldiklio ir MQTT brokerio

Atlikus stebėjimo funkcijų pasiekiamumo tikrinimą matoma, jog funkcija veikia tinkamai. Atliktas alternatyvus testas, kuomet nei viena iš šių funkcijų nebuvo paleistos. Tikrinant funkcijų pasiekiamumą būsena buvo pažymėta kaip „Disconnected“.

24 lentelė. Duomenų formavimo ir siuntimo į duomenų bazę testavimo procesas

Testavimo atvejis	Duomenų formavimo ir siuntimo į duomenų bazę tikrinimas			
Aprašymas	Testuojama, ar prie suformuotų duomenų yra pridamas ID kintamasis ir ar duomenų perdavimo būsena rodoma taisyklingai			
Sąlyga – prieš	Stebėjimo sistema rodo, jog ID nėra nurodytas, o būsena pažymėta „STOPPED“			
Sąlyga – po	Stebėjimo sistema rodo nurodytą ID ir būsena pažymėta „RUNNING“			
Testavimo žingsniai				
Nr.	Duomenys	Veiksmas	Numatomas rezultatas	Gautas rezultatas
1.	-	Stebėjimo įrankyje pasirenkama General skiltis	Atidaromas bendrinis sistemų būsenų langas	Bendrinis sistemų būsenų langas atidarytas

2.	-	Grupėje „SYSTEM STATUS“ nuspaudžiamas „Connect“ mygtukas	Atliekamos funkcijos, prisijungiančios prie „BeamNG.Drive“, „Beam Eye Tracker“, ESP32 mikrovaldiklio ir MQTT brokerio	Sėkmingai prisijungta prie „BeamNG.Drive“, „Beam Eye Tracker“, ESP32 mikrovaldiklio ir MQTT brokerio
3.	-	Grupėje „SESSION CONTROL“ nuspaudžiamas „Start“ mygtukas	Atliekama funkcija, kuri prisijungia prie InfluxDB3 duomenų bazės ir prie siunčiamų duomenų prideda lentelės ID	Su nauju ID sukuriamą lentelę, kurioje yra saugomi surinkti ir suformatuoti duomenys iš „BeamNG.Drive“, „Beam Eye Tracker“, ESP32 mikrovaldiklio ir MQTT brokerio

Atlikus duomenų formavimo ir siuntimo į duomenų bazę tikrinimą matoma, jog funkcija veikia tinkamai. Atliktas alternatyvus testas, kuomet duomenų bazė nebuvo paleista. Tikrinant funkcijos veikimą būseną buvo pažymėta kaip „Disconnected“.

25 lentelė. Analizės įrankio duomenų failo įkėlimo testavimo procesas

Testavimo atvejis	Analizės įrankio duomenų failo įkėlimo tikrinimas			
Aprašymas	Testuojama, ar failo įkėlimas atliekamas korektiškai			
Sąlyga – prieš	Analizės įrankis neturi įkeltų duomenų			
Sąlyga – po	Analizės įrankis turi taisyklingai suformuotus duomenis			
Testavimo žingsniai				
Nr.	Duomenys	Veiksmas	Numatomas rezultatas	Gautas rezultatas
1.	-	Analizės įrankyje spaudžiamas mygtukas „Load measurements“	Atidaromas failo pasirinkimo langas	Atidarytas failo pasirinkimo langas
2.	-	Pasirenkamas *.csv failas	Į analizės įrankį yra įkeltas scenarijaus rezultatų failas	Į analizės įrankį įkeltas scenarijaus rezultatų failas
3.	-	Failo įrašai automatiškai atvaizduojami lentelėje	Failo įrašai atvaizduojami lentelėje pagal iš anksto suformuotus stulpelius	Failo įrašai atvaizduojami lentelėje pagal iš anksto suformuotus stulpelius

Atlikus analizės įrankio duomenų failo įkėlimo testavimą, matoma, jog failo pasirinkimo langas yra atidaromas korektiškai, failas yra įkeliamas pagal nustatytus stulpelius. Alternatyvus testas buvo to

pačio failo įkėlimas. Gautas rezultatas – įkeltas failas perrašo esančius duomenis rodomus lentelėje. Taip užtikrinama, jog duomenys nėra papildomai įtraukiami į lentelę, sugadinant matavimų rodmenis.

26 lentelė. Analizės įrankio įkeltų matavimų modifikavimo apsaugos testavimo procesas

Testavimo atvejis	Analizės įrankio įkeltų matavimų modifikavimo apsaugos tikrinimas			
Aprašymas	Testuojama, ar įkelti duomenys gali būti pakeisti analizės įrankyje rankiniu būdu			
Sąlyga – prieš	Analizės įrankis turi įkeltus duomenis			
Sąlyga – po	Analizės įrankis leidžia koreguoti tik komentarų skiltį			
Testavimo žingsniai				
Nr.	Duomenys	Veiksmas	Numatomas rezultatas	Gautas rezultatas
1.	-	Pasirenkamas laukas, kuris nėra stulpelyje „Comments“	Pasirinktas langelis yra paryškinamas	Pasirinktas langelis yra paryškinamas
2.	-	Įvedamas pasirinktas simbolis į nurodytą langelį	Langelis, kuris priklauso įkeltam failui nėra leidžiamas modifikuoti	Langelis, kuris priklauso įkeltam failui nėra leidžiamas modifikuoti

Atlikus analizės įrankio įkeltų matavimų modifikavimo apsaugos tikrinimą, matoma, jog įkelti duomenys nėra leidžiami modifikuoti naudotojui tiesiogiai. Jie yra keičiami tik pakeitus įkeltą duomenų failą arba įkelto palyginimo failo. Vienintelis laukas, kurį naudotojas gali modifikuoti yra „Comments“, kuriame gali būti pažymėti komentarai.

27 lentelė. Analizės įrankio etaloninių matavimų pagal scenarijų failo įkėlimo testavimo procesas

Testavimo atvejis	Analizės įrankio etaloninių matavimų pagal scenarijų failo įkėlimo tikrinimas			
Aprašymas	Testuojama, ar etaloninių matavimų failo įkėlimas atliekamas korektiškai			
Sąlyga – prieš	Analizės įrankis turi įkeltus pradinis duomenis			
Sąlyga – po	Analizės įrankis turi taisyklingai išanalizuotus duomenis			
Testavimo žingsniai				
Nr.	Duomenys	Veiksmas	Numatomas rezultatas	Gautas rezultatas
1.	-	Analizės įrankyje spaudžiamas mygtukas „Load benchmark“	Atidaromas failo pasirinkimo langas	Atidarytas failo pasirinkimo langas
2.	-	Pasirenkamas *.csv failas	Į analizės įrankį yra įkeltas scenarijaus rezultatų failas	Į analizės įrankį įkeltas scenarijaus rezultatų failas
3.	-	Atliekama duomenų analizė pagal nustatytus scenarijaus taškus	Pagal nustatytus taškus atliekamas palyginimas ir neatitinkantys ribos	Pagal nustatytus taškus ir atliktą palyginimą parodomi duomenys, kurie

			duomenys yra paryškunami	neatitinka nustatytų ribų
--	--	--	--------------------------	---------------------------

Atlikus analizės įrankio etaloninių matavimų failo įkėlimo testavimą, matoma, jog failo pasirinkimo langas yra atidaromas korektiškai, failas yra įkeliamas pagal nustatytus stulpelius. Alternatyvus testas buvo to pačio failo įkėlimas. Gautas rezultatas – įkeltas failas perrašo esančius duomenis rodomus lentelėje. Taip užtikrinama, jog duomenys nėra papildomai įtraukiami į lentelę, sugadinant matavimų rodmenis.

28 lentelė. Analizės įrankio palyginimo rezultatų failo sukūrimo testavimo procesas

Testavimo atvejis	Analizės įrankio rezultatų palyginimo failo sukūrimo tikrinimas			
Aprašymas	Testuojama, ar rezultatų palyginimo failas yra išsaugomas korektiškai			
Sąlyga – prieš	Analizės įrankis turi išanalizuotų kintamųjų lentelę			
Sąlyga – po	Analizės įrankis sėkmingai išsaugo išanalizuotų kintamųjų lentelę			
Testavimo žingsniai				
Nr.	Duomenys	Veiksmas	Numatomas rezultatas	Gautas rezultatas
1.	-	Analizės įrankyje spaudžiamas mygtukas „Export data“	Atidaromas failo išsaugojimo langas	Atidarytas failo išsaugojimo langas
2.	-	Įvedamas *.csv failo pavadinimas, kuriuo bus išsaugotas duomenų failas	Įvedus pavadinimą ir paspaudus „Save“ mygtuką, failas yra išsaugomas	Įvedus pavadinimą ir paspaudus „Save“ mygtuką, failas yra išsaugomas
3.	-	Atsiradusiame pranešimo lange su tekstu „Export completed!“ spaudžiama OK	Failas yra išsaugomas, naudotojui yra parodomas pranešimas apie sėkmingai išsaugotą failą	Failas yra išsaugotas, naudotojui yra parodytas pranešimas apie sėkmingai išsaugotą failą

Atlikus analizės įrankio rezultatų palyginimo failo sukūrimą, matoma, jog failo išsaugojimo langas yra atidaromas taisyklingai, failas yra išsaugomas taisyklingai ir naudotojui suteikiamas pranešimas apie failo išsaugojimą. Alternatyvus testas buvo duomenų įrašymas į aplanką, kuriame negalimas failų įrašymas. Bandant tai atlikti rodomas pranešimas, jog naudotojas neturi tinkamų teisių įrašyti norimą failą į norimą aplanką.

29 lentelė. Analizės įrankio išėjimo mygtuko testavimo procesas

Testavimo atvejis	Analizės įrankio išėjimo mygtuko tikrinimas			
Aprašymas	Tikrinama, ar analizės įrankio išėjimo mygtukas veikia korektiškai			
Sąlyga – prieš	Analizės įrankis yra atidarytas			
Sąlyga – po	Analizės įrankis yra uždarytas			

Testavimo žingsniai				
Nr.	Duomenys	Veiksmas	Numatomas rezultatas	Gautas rezultatas
1.	-	Spaudžiamas mygtukas „Exit“	Analizės įrankis yra uždaromas	Analizės įrankis yra uždarytas

Atlikus analizės įrankio išėjimo mygtuko tikrinimą, matoma, jog programa uždaroma taisyklingai.

Atlikus ekstremalaus vairavimo prototipo funkcionalumo testavimą nebuvo pastebėta trūkumų, kurie paveiktų prototipo veikimą. Galima teigti, kad prototipas teisingai atlieka numatytas funkcijas.

6. Ekstremalaus vairavimo simulatoriaus dokumentacija

Šiame skyriuje pateikiama ekstremalaus vairavimo diegimo ir naudotojo vadovai.

6.1. Ekstremalaus vairavimo simulatoriaus diegimo vadovas

Norint naudotis ekstremalaus vairavimo simulatoriumi, reikalinga sukompiliuoti ir įdiegti mikrovaldiklio, analizės įrankio, stebėjimo įrankio programinius kodus, taisyklingai paleisti duomenų bazės atvaizdavimo ir įrašymo komponentus, įrašyti papildomą programinę įrangą bei paruošti techninę įrangą.

Šis diegimo vadovas skirtas Windows kompiuterių operacinėms sistemoms. Programinės įrangos projektai reikalauja nedidelių konfigūracijos pakeitimų po kurių jie yra paruošti naudojimui.

Mikrovaldiklio programinio kodo diegimui naudojama „PlatformIO Core“ aplinka, kuri yra paruošiama pagal „PlatformIO“ tinklalapyje[31] prieinamas instrukcijas. Įsidiegus „PlatformIO“ įrankius, vykdomi tolimesni žingsniai:

1. atidaromas projektas „ESP32-sensors“ aplankale;
2. prie kompiuterio prijungiamas mikrovaldiklis;
3. pakeičiama informacija apie naudojama bevielį tinklą ir jo duomenis;
4. komandinėje eilutėje vykdoma komanda „pio run -target upload“.

Įvykdžius šiuos žingsnius, programinis kodas yra sukompiliuojamas ir įkeliamas į mikrovaldiklį. Jei mikrovaldiklio komunikacijos „COM“ įvadas yra neaptinkamas, reikalinga papildoma modifikacija „platform.ini“ faile. Nurodytame faile pridedama eilutė „upload_port = /dev/ttyUSBx“, kurioje „x“ turi būti pakeistas atitinkamo įvado skaičius.

Stebėjimo įrankio kopija yra sukompiliuota ir paruošta naudojimui. Jei norima pačiam naudotojui susikompiliuoti, tuomet programinio kodo diegimui naudojama „Visual Studio Code“ aplinka ir „.NET 10.0“ karkasas ir „HiveMQTT“ „NuGet“ paketas. Įsidiegus šiuos elementus projektas gali būti sukompiliuotas.

Analizės įrankio kopija taip pat yra sukompiliuota ir paruošta naudojimui. Jei norima pačiam naudotojui susikompiliuoti, tuomet programinio kodo diegimui naudojama „Visual Studio Code“ aplinka ir „.NET 10.0“ karkasas. Įsidiegus šiuos elementus projektas gali būti sukompiliuotas.

Vairuotojo sistemoje turi būti įdiegtos „BeamNG.Drive“ ir „Beam Eye Tracker“ programinių įrangų kopijos.

„BeamNG.Drive“ turi būti bent vieną kartą paleista, kad būtų sugeneruoti naudotojo failai. Sugeneruotuose failuose, kurie įprastai randami „C:\Users\<naudotojo vardas>\AppData\Local\BeamNG\BeamNG.drive\current“. Šiame aplankale turi būti sukurtas aplankalas pavadinimu „lua“. Šiame aplankale yra patalpinami failai „telemetry.lua“ ir „telemetryvehicle.lua“. Šie failai bus naudojami komunikacijos paleidimui rankiniu būdu.

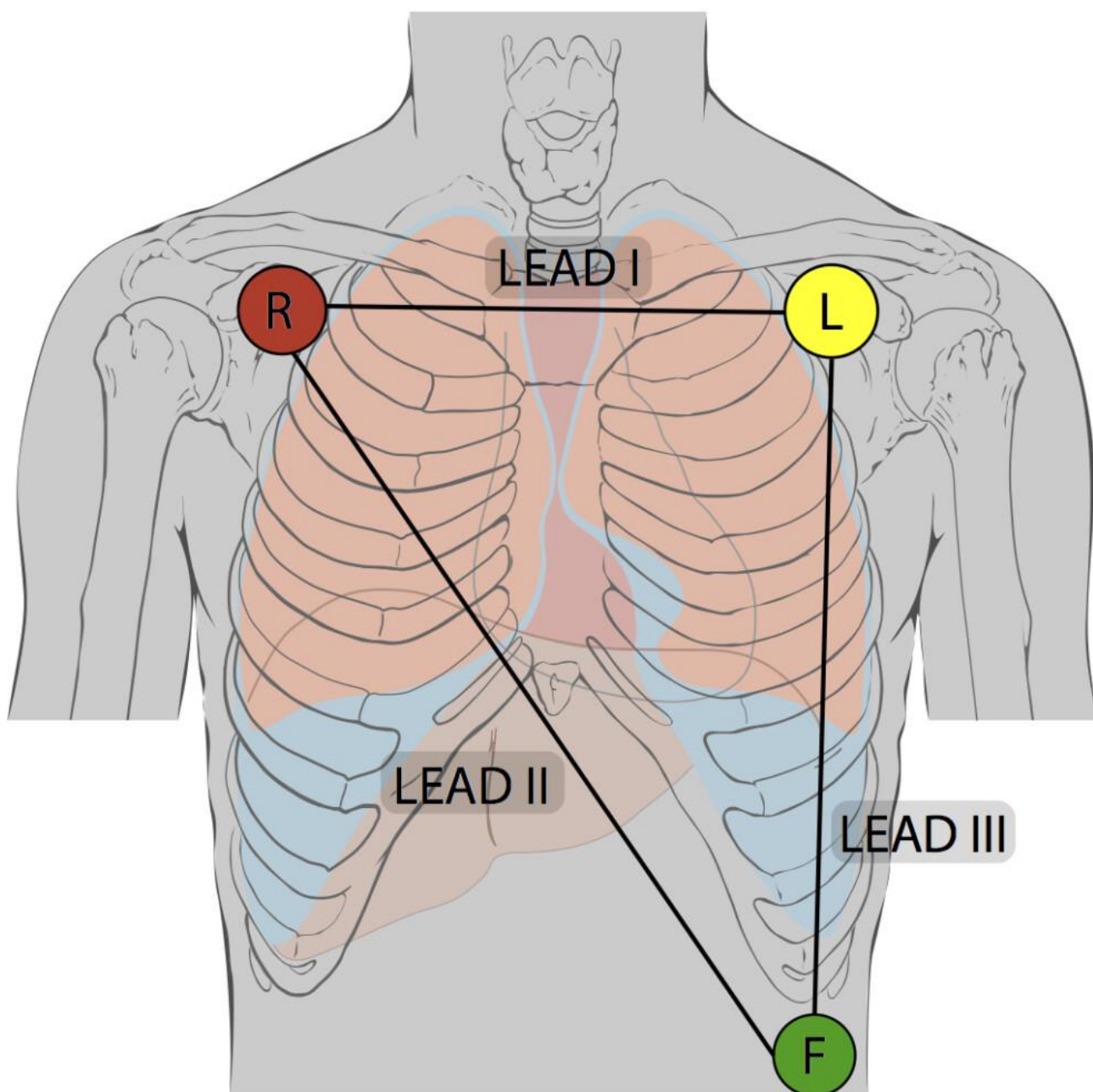
6.2. Ekstremalaus vairavimo simulatoriaus naudotojo vadovas

Šis poskyris yra išskaidytas į dvi dalis, siekiant išaiškinti naudojimosi žingsnius vairuotojo ir stebėtojo pusėse.

6.3. Ekstremalaus vairavimo simulatoriaus stebėjimo įrankio naudotojo vadovas

Stebėtojas pirmiausia turi užtikrinti, jog visi elementai yra paruošti naudojimui:

1. vairas, pedalai bei pavarų svirtis yra prijungti prie kompiuterio;
2. ESP32 yra prijungtas prie kompiuterio;
3. AD8232 sensoriai ant vairuotojo yra uždėti pagal šią schemą (žr. **pav. 22**). Spalva turi atitikti poziciją nurodytą paveikslėlyje;



22 pav. AD8232 sensoriaus jutiklių jungimo schema

4. GSR sensoriai uždėti ant vairuotojo rodomojo ir vidurinio pirštų arba vidurinio ir bevardžio rankos pirštų.

Atlikus fizinių elementų patikrinimą ir konfigūraciją pereinama prie programinės įrangos dalies. Pirmiausia reikia paleisti stebėjimo įrankį.

Paleistame įrankyje reikia patikrinti visą reikalingą komunikaciją tarp kitų elementų kaip „BeamNG.Drive“, „Beam Eye Tracker“, „InfluxDB 3“ ir „MQTT“ brokerio. Skiltyje „Testing“ (žr. **pav. 23**) galima rasti konfigūracijos informaciją.

23 pav. Stebėjimo įrankio „Testing“ skiltis

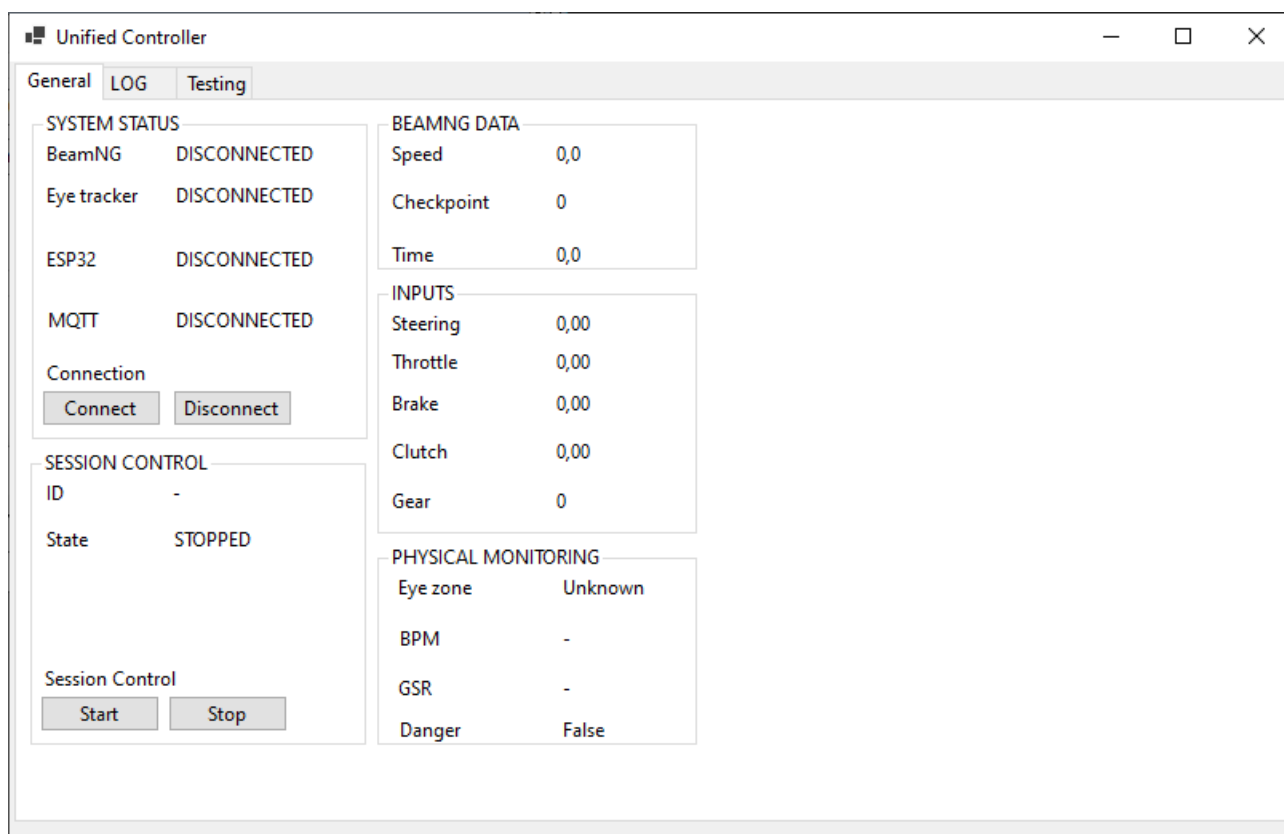
Grupėje „APPS SETUP“, paspaudus mygtuką „Browse“ pagal savo įrašytų programų vietą reikalinga pasirinkti atitinkamus „BeamNG.Drive“ ir „Beam Eye Tracker“ paleisties failus. Išsaugojus šių failų vietas spaudžiami mygtukai „Launch“, kurie atitinkamai randami „BeamNG path“ ir „Beam Eye Tracker path“ eilučių dešinėje pusėje. Atlikus šiuos veiksmus rekomenduojama paspausti „Test Process“, kad patikrinti procesų veikimą.

Siekiant užtikrinti taisyklingą veikimą su „InfluxDB 3 Core“ duomenų baze, naudotojas nustato serverio IP adresą, duomenų bazės prievado numerį, duomenų bazės pavadinimą ir API žetoną grupėje „INFLUXDB SETUP“. Suvedus šiuos duomenis rekomenduojama paspausti „Test“. Taip bus patikrinta, ar komunikacija tarp programos ir duomenų bazės yra vykdoma taisyklingai.

Grupėje „MQTT SETUP“ yra kintamieji dėl „MQTT“ brokerio „IP“ adreso, prievado, „ESP32“ ir duomenų bazės temų („topic“). Suvedus šiuos duomenis rekomenduojama paspausti „Test“. Taip bus patikrinta, ar komunikacija tarp programos ir „MQTT“ brokerio yra vykdoma taisyklingai.

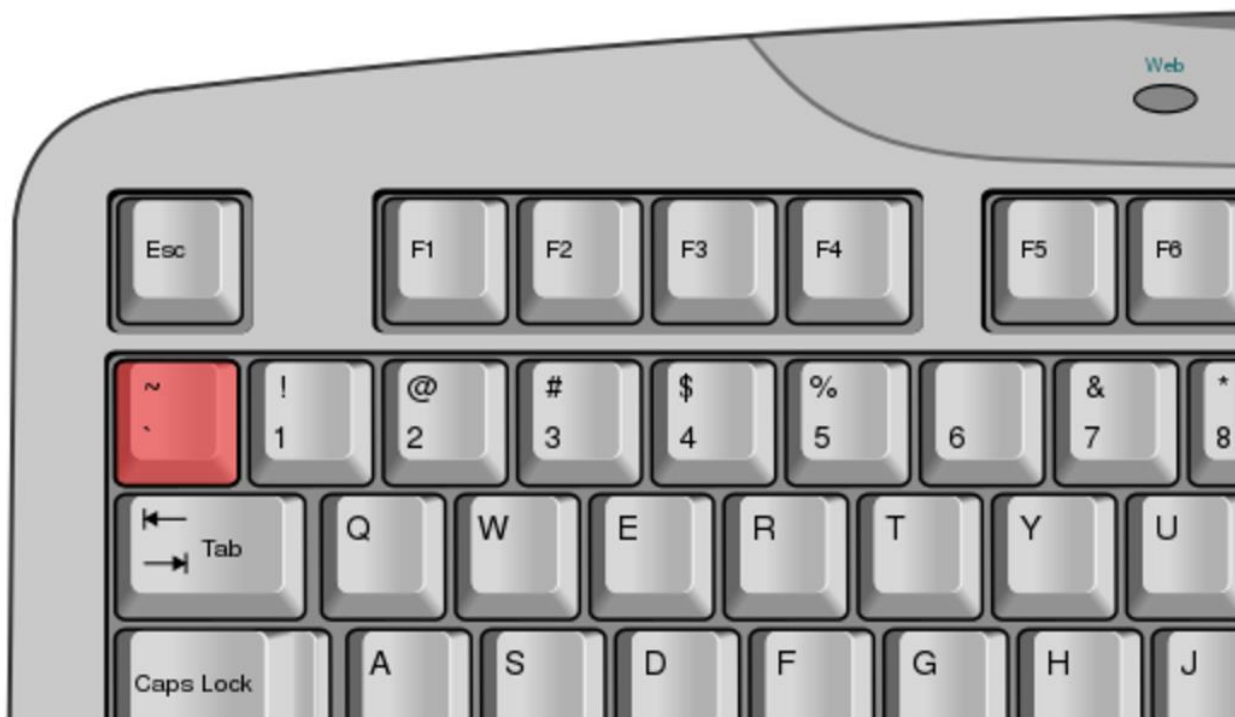
Paleidus „BeamNG.Drive“ ir „Beam Eye Tracker“ bei užtikrinus komunikaciją tarp duomenų bazės ir brokerio, pasirenkama skiltis „General“ (žr. **pav. 24**). Šioje skiltyje surandama grupė „System

status“. Joje reikia paspausti „Connect“ mygtuką, siekiant pradėti komunikaciją su esamais komponentais.



24 pav. Stebėjimo įrankio „General“ skiltis

„BeamNG.Drive“ programoje reikalinga atlikti papildomą žingsnį. Paspausdus „`“ mygtuką (žr. **pav. 25**), kuris atidarys žaidimo konsolę. Joje turi būti įvesta komanda „extensions.load(„telemetry“)“. Ši komanda leidžia perduoti reikalingus duomenis apie scenarijaus ir automobilio būsenas.



25 pav. Klaviatūros mygtukas žaidimo konsolės langui atidaryti

Teisingai atlikus nurodytus žingsnius, grupėse „BEAMNG DATA“, „INPUTS“ ir „PHYSICAL MONITORING“ bus matomi duomenys apie gaunamus duomenis iš prijungtų komponentų.

Toje pačioje skiltyje „General“ yra grupė „SESSION CONTROL“. Joje prieš pradedant scenarijų reikia paspausti „Start“. Šis mygtukas atlieka surinktų duomenų iš komponentų perdavimą pagal sugeneruotą identifikacinį kintamąjį, taip užtikrinant, jog nebūs 2 vienodų įrašų duomenų bazėje.

Teisingai atlikus nurodytus žingsnius, duomenys yra perduodami į duomenų bazę saugoti. Scenarijai gali būti pradėti vykdyti.

Baigus darbą su stebėjimo įrankiu pirmiausia „SESSION CONTROL“ grupėje spaudžiame mygtuką „Stop“, taip sustabdydami duomenų įrašymą į duomenų bazę. Tuomet „SYSTEM STATUS“ grupėje spaudžiame mygtuką „Disconnect“. Tokiu būdu užtikriname, jog duomenys yra nustojami rinkti stebėjimo įrankyje. Atlikus šiuos veiksmus, galima saugiai išjungti „BeamNG.Drive“ bei „Beam Eye Tracker“ ir atjungti „ESP32“ mikrovaldiklį.

6.4. Ekstremalaus vairavimo simulatoriaus analizės įrankio naudotojo vadovas

Atlikus norimus scenarijus atsiunčiamas norimas duomenų bazės įrašas. Stebėtojas, prisijungęs prie duomenų bazės per „Grafana“ vizualinę sąsają pasirenka norimą duomenų bazės lentelę pagal sugeneruotą identifikacinį kintamąjį. Atsidarius norimai lentelei spaudžiamas mygtukas „Export“, kuris atveria papildomą pasirinkimo langą, kuriame prašoma nurodyti saugojimo vietą.

Išsaugojus duomenis, paleidžiamas analizės įrankis (žr. **pav. 26**). Įrankiui pasileidus, spaudžiamas mygtukas „Load measurements“. Atsivėrusiame lange pasirenkamas tinkamas duomenų failas, kuris turi būti „CSV“ formato. Pasirinkus failą jis yra įkeliamas į analizės įrankį.



26 pav. Analizės įrankio išvaizda

Analizės įrankyje norint įkelti palyginimo failą, spaudžiamas mygtukas „Load benchmark“. Atsivėrusiame lange pasirenkamas tinkamas duomenų failas, kuris turi būti „CSV“ formato. Pasirinktas failas yra įkeliamas į analizės įrankį. Įkeltas failas pažymi scenarijaus kritinių sekcijų veiksmus, kurie neatitinka nurodytų veiksmų, spalvomis. Stebėtojas, esant poreikiui, gali papildomai pasirašyti komentarus prie kiekvienos duomenų eilutės.

Atlikus analizę, spaudžiamas mygtukas „Export data“. Atsivėrusiame lange pasirenkame vietą, kurioje bus išsaugomas išanalizuotas failas.

Baigus darbą su analizės įrankiu spaudžiamas mygtukas „Exit“.

Išvados

1. Atlikus probleminės srities ir esamų vairavimo simulatorių rinkos analizę, išsiaiškinta, jog daugelis komerciškai prieinamų sprendimų yra specializuoti, reikalaujantys naudoti jų pačių programinę įrangą ir dažniausiai be galimybės keisti ar pridėti funkcionalumus patiems.
2. Atlikus ekstremalaus vairavimo simulatoriaus prototipo programinės įrangos ir techninės įrangos kūrimo technologijų analizę, apibrėžti reikalavimai reikalingi prototipo realizacijai.
3. Sėkmingai sukurtas ekstremalaus vairavimo simulatoriaus prototipas, kuris gali atlikti biometrinių duomenų matavimus, akių stebėjimą, simuliacijos duomenų surinkimą, duomenų paketų suformavimą ir jų saugojimą duomenų bazėje.
4. Atlikti bandymai rodo, jog sistema gali būti taikoma įvairių scenarijų tipams, leidžianti palaikyti universalumą tarp skirtingų taikymo tikslų ir, jei reikalinga, naujai sukurti scenarijai reikalauja tik palyginimo duomenų sudarymo rezultatų analizei.
5. Prototipas gali palaikyti įvairaus tipo valdiklius, leidžiančius lengvesnei integracijai su esamomis sistemomis.
6. Prototipas, lyginant su komerciškai prieinamais sprendimais, gali palaikyti įvairaus tipo akių stebėjimo komponentus ir biometrinius stebėjimus, leidžiančius lengvesnei integracijai su esamomis sistemomis.
7. Pagrindinis projekto trūkumas – analizės įrankio duomenų tikrinimas su etaloniniais duomenimis. Siekiant pagerinti matavimo tikslumą, reikalinga tolimesnė analizė palyginimų procesui tobulinti.
8. Galimi prototipo patobulinimai – „BeamNG“ LUA kodo paleidimo automatizacija ir duomenų perdavimo automatizacija į analizės įrankį. Tai palengvintų viso prototipo taikymą atliekant tyrimus.
9. Buvo susisiekiama su „BeamNG“ įmone, kuri išreiškė susidomėjimą projektu ir jo eiga.

Literatūros sąrašas

1. Effect of driving experience on visual behavior and driving performance under different driving conditions – Larry B. Nabitlan, Fereydoun Aghazadeh, Ashish D. Nimbarte, Craig C. Harvey, Suman K. Chowdhury – 2011-08-02. [žiūrėta 2026-02-10]. Prieiga per internetą <<https://link.springer.com/article/10.1007/s10111-011-0184-5>>
2. Driver Behavior Analysis for Safe Driving – Sinan Kaplan, Mehmet Amac Guvensan, Ali Gokhan Yavuz, Yasin Karalurt – 2015-18-16. [žiūrėta 2026-02-15]. Prieiga per internetą <<https://ieeexplore.ieee.org/abstract/document/7225158>>
3. Drivers' reaction time research in the conditions in the real traffic - Paweł Drożdziel, Sławomir Tarkowski, Iwona Rybicka, Rafał Wrona - 2020-01-31. [žiūrėta 2026-02-15]. Prieiga per internetą <<https://www.degruyterbrill.com/document/doi/10.1515/eng-2020-0004/html>>
4. Effects of driving experience and sensation-seeking on drivers' adaptation to road environment complexity - Christina M. Rudin-Brown, Jessica Esquist, Michael G. Lenné – 2014-02. [žiūrėta 2024-02-15]. Prieiga per internetą <<https://www.sciencedirect.com/science/article/abs/pii/S0925753513001963>>
5. Drive Square Simulation System - DrivingSimulators. [žiūrėta 2026-02-01]. Prieiga per internetą <<https://drivingsimulator.com/products/Drive%20Square%20DS-565%20Data%20Sheet.pdf>>
6. ESR-1 Racing Simulator - SimXperience [interaktyvus]. [žiūrėta 2026-02-01]. Prieiga per internetą <<https://www.simxperience.com/shop/esr-1-racing-simulator-1170?category=4#attr=208,263>>
7. Desktop Models for Risk Mitigation Simulators - Simulator Systems International [interaktyvus]. [žiūrėta 2026-02-01]. Prieiga per internetą <<https://www.simulatorsystems.com/risk-mitigation-simulators>>
8. Car driving simulator - Tecknotrove [interaktyvus]. [žiūrėta 2026-02-01]. Prieiga per internetą <<https://tecknotrove.com/industries/automobile/car-driving-simulator/>>
9. InfluxDB 3 Core documentation - Influxdata [interaktyvus]. [žiūrėta 2026-02-04]. Prieiga per internetą <<https://docs.influxdata.com/influxdb3/core/>>
10. MongoDB Docs - MongoDB [interaktyvus]. [žiūrėta 2026-02-04]. Prieiga per internetą <<https://www.mongodb.com/docs/>>
11. Cassandra documentation - Cassandra [interaktyvus]. [žiūrėta 2026-02-04]. Prieiga per internetą <<https://cassandra.apache.org/doc/latest/>>
12. HTTP Documentation - HTTP [interaktyvus]. [žiūrėta 2026-02-04]. Prieiga per internetą <<https://mqtt.org/mqtt-specification/>>
13. MQTT Specifications - MQTT.org [interaktyvus]. [žiūrėta 2026-02-04]. Prieiga per internetą <<https://httpwg.org/specs/>>
14. WebSocket Guides - WebSocket.org [interaktyvus]. [žiūrėta 2026-02-04]. Prieiga per internetą <<https://websocket.org/guides/road-to-websockets>>
15. City Car Driving - Advantages and features – Forward Development [interaktyvus]. [žiūrėta 2024-02-05]. Prieiga per internetą <<https://citycardriving.com/features>>
16. OpenDS Features – boomaa23 [interaktyvus]. [žiūrėta 2026-02-05]. Prieiga per internetą <<https://boomaa23.github.io/open-ds/#features>>
17. BeamNG Documentation - BeamNG [interaktyvus]. [žiūrėta 2026-02-05]. Prieiga per internetą <<https://documentation.beamng.com/>>

18. Beam Eye Tracker - Swiss Software [interaktyvus]. [žiūrėta 2026-02-05]. Prieiga per internetą <<https://beam.eyeware.tech/developers/>>
19. OpenCV modules - OpenCV [interaktyvus]. [žiūrėta 2026-02-05]. Prieiga per internetą <<https://docs.opencv.org/4.x/index.html>>
20. Tobii Pro SDK documentation - Tobii AB [interaktyvus]. [žiūrėta 2026-02-05]. Prieiga per internetą <<https://developer.tobii.com/>>
21. ESP32-S3 Datasheet - Espressif. [žiūrėta 2026-02-06]. Prieiga per internetą <https://documentation.espressif.com/esp32-s3_technical_reference_manual_en.pdf>
22. Arduino Mega 2560 Rev3 - Arduino [interaktyvus]. [žiūrėta 2026-02-06]. Prieiga per internetą <<https://docs.arduino.cc/hardware/mega-2560/#suggested-libraries>>
23. Raspberry Pi 4 Model B Datasheet – Raspberry Pi. [žiūrėta 2026-02-06]. Prieiga per internetą <<https://pip-assets.raspberrypi.com/categories/545-raspberry-pi-4-model-b/documents/RP-008341-DS-1-raspberry-pi-4-datasheet.pdf?disposition=inline>>
24. MAX30102 – Maxim Integrated Products Inc. . [žiūrėta 2026-02-06]. Prieiga per internetą <<https://www.analog.com/media/en/technical-documentation/data-sheets/MAX30102.pdf>>
25. SEN-KY039HS Heartbeat sensor - Joy it. [žiūrėta 2026-02-06]. Prieiga per internetą <<https://joy-it.net/files/files/Produkte/SEN-KY039HS/SEN-KY039HS-Manual-21-05-19.pdf>>
26. AD8232 – Analog devices Inc. . [žiūrėta 2026-02-06]. Prieiga per internetą <<https://www.analog.com/media/en/technical-documentation/data-sheets/AD8232.pdf>>
27. Grove - GSR - seeedstudio. [žiūrėta 2026-02-06]. Prieiga per internetą <https://raw.githubusercontent.com/SeeedDocument/Grove-GSR_Sensor/master/res/Lm324.pdf>
28. FSR 400 Series Data Sheet - Interlink Electronics. [žiūrėta 2026-02-06]. Prieiga per internetą <https://files.seeedstudio.com/wiki/Grove-Round_Force_Sensor_FSR402/res/FSR402.pdf>
29. DHT11 Humidity & Temperature Sensor – Mouser Electronics. [žiūrėta 2026-02-06]. Prieiga per internetą <<https://www.mouser.com/datasheet/2/758/DHT11-Technical-Data-Sheet-Translated-Version-1143054.pdf>>
30. EasyEda standard editor – JiaLiChuang (HongKong) Co., Limited. [žiūrėta 2026-04-02]. Prieiga per internetą <<https://easyeda.com/editor#>>
31. PlatformIO install – PlatformIO Labs [interaktyvus]. [žiūrėta 2026-05-02]. Prieiga per internetą <<https://platformio.org/install>>