



Kauno technologijos universitetas
Informatikos fakultetas

Mokyklos administracinė sistema

Baigiamasis bakalauro

Vytenis Giedraitis
Projekto autorius

Jurgita Arnastauskaitė
Vadovė

Kaunas, 2021



Kauno technologijos universitetas
Informatikos fakultetas

Mokyklos administracinė sistema
Baigiamasis bakalauro projektas
6121BX014

Vytenis Giedraitis
Projekto autorius

Jurgita Arnastauskaitė
Vadovė

Lekt. Vardenis Pavardenis
Recenzentas / Recenzentė

BAKALAURO BAIGIAMOJO PROJEKTO UŽDUOTIS

1. Bakalauro baigiamojo projekto tema:

Mokyklos administracinė sistema

2. Bakalauro baigiamojo projekto tikslas:

Suprojektuoti ir sukurti administracinės sistemos prototipą, kuris palengvintų bei pagreitintų tam tikrus administracinius procesus mokykloje.

3. Bakalauro baigiamojo projekto uždaviniai:

- a. Išanalizuoti mokyklos administracinių darbų specifiką.
- b. Išsiaiškinti administracinių sistemų veikimo principus, atlikti panašių informacinių sistemų bei naujai sistemai realizuoti naudosimų technologijų analizę.
- c. Paruošti administracinės sistemos projektą.
- d. Parengti vartotojo vadovą.

4. Bakalauro baigiamojo projekto pateikimo terminai:

Vadovui (8 d. d. prieš gynimą katedroje): 2021 m. gegužės 17 d.
Recenzentui (6 d. d. prieš gynimą katedroje): 2021 m. gegužės 19 d.
Komisijai (6 d. d. prieš gynimą katedroje): 2021 m. gegužės 19 d.

5. Funkciniai reikalavimai kuriamam objektui arba sistemai:

- a. Vartotojų prisijungimas
- b. Naujų vartotojų registravimas bei valdymas
- c. Moksleivių sąrašo įkėlimas iš failo
- d. Naujų moksleivių įvedimas bei redagavimas
- e. Moksleivių skirstymas į klases bei srautus
- f. Klasių priskyrimas mokytojams (pagal mokomuosius dalykus)
- g. Kabinėtų paskyrimas mokytojams
- h. Tvarkaraščio sudarymas bei redagavimas
- i. Tvarkaraščio eksportavimas nurodytu formatu
- j. Prašymų rašymas
- k. Automatinis moksleivių tėvų informavimas elektroniniu paštu

6. Nefunkciniai reikalavimai kuriamam objektui arba sistemai:

- a. Puslapis prisitaikantis prie ekrano dydžio
- b. Sistema lietuvių kalba

- c. Įvykus klaidai vartotojui rodomi klaidos pranešimai
- d. Neįvedus privalomų duomenų, formos neturėtų būti išsiunčiamos
- e. Duomenų bazėje svarbiausi duomenys turi būti šifruojami

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

- a. Serverio sistema turi palaikyti technologijas pasirinktas sistemos realizacijai
- b. Duomenų perdavimui turi būti naudojamas SSL šifravimas

Vadovas: Jurgita Arnastauskaitė

(vardas, pavardė, parašas)

Data: 2021 m. vasario 5 d.

(suderinimo data)

Studentas: Vytenis Giedraitis

(vardas, pavardė, parašas)

Data: 2021 m. vasario 5 d.

(suderinimo data)

Kaunas, 2021



Kauno technologijos universitetas

Informatikos fakultetas

Vytenis Giedraitis

Mokyklos administracinė sistema

Akademinio sąžiningumo deklaracija

Patvirtinu, kad mano, Vytenio Giedraičio, baigiamasis projektas tema „Mokyklos administracinė sistema“ 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)



Kauno technologijos universitetas
Fakulteto pavadinimas

Baigiamojo bakalauro / magistro projekto užduotis (pagal poreikį)

Projekto tema

Reikalavimai ir sąlygos
(tikslinti pavadinimą
pagal poreikį)

Vadovas / Vadovė

(vadovo pareigos, vardas, pavardė, parašas)

(data)

Autoriaus pavardė, vardas. Baigiamojo projekto pavadinimas. Bakalauro / Magistro / Profesinių studijų / Gretutinės krypties studijų (pasirinkite) baigiamasis projektas / vadovas / vadovė doc. dr. Vardenis Pavardenis; Kauno technologijos universitetas,(įrašykite) fakultetas.

Studijų kryptis ir sritis (studijų krypčių grupė):(įrašykite).

Reikšminiai žodžiai:(įrašykite).

Miestas, 20XX. XX p.

Santrauka

Lorem ipsum dolor sit amet, eam ex decore persequeris, sit at illud lobortis atomorum. Sed dolorem quaerendum ne, prompta instructor ne pri. Et mel partiendo suscipiantur, docendi abhorreant ea sit. Recteque imperdiet eum te.

Eu eum decore inimicus consetetur, cu usu habeo corpora intellegam. Ut antiopam efficiendi deterrisset sit. Mel sint eirmod id, qui quot virtute id, dolor nemore forensibus usu id. Fugit dolore voluptatum cu vim. An vix veniam graecis insolens, sit posse iusto id. Ut vim ceteros percipit, id quo ubique recusabo, eum sint lucilius ea. In sumo inani numquam has.

Author's surname, name. Title of the Final Degree Project. Bachelor's / Master's Final Degree Project / Final Degree Project of Minor Studies / Professional Studies (choose one) / supervisor abbreviation of the position, name and surname of the supervisor; Name of the Faculty, Kaunas University of Technology.

Study field and area (study field group): (type here).

Keywords: (type here).

Town, Year. Number of pages.

Summary

Lorem ipsum dolor sit amet, eam ex decore persequeris, sit at illud lobortis atomorum. Sed dolorem quaerendum ne, prompta instructor ne pri. Et mel partiendo suscipiantur, docendi abhorreant ea sit. Recteque imperdiet eum te.

Eu eum decore inimicus consetetur, cu usu habeo corpora intellegam. Ut antiopam efficiendi deterruisset sit. Mel sint eirmod id, qui quot virtute id, dolor nemore forensibus usu id. Fugit dolore voluptatum cu vim. An vix veniam graecis insolens, sit posse iusto id. Ut vim ceteros percipit, id quo ubique recusabo, eum sint lucilius ea. In sumo inani numquam has.

Turinys

Lentelių sąrašas	10
Paveikslų sąrašas	11
Santrumpų ir terminų sąrašas	12
Ivadas.....	13
1. Analizė	14
2. Technologijų, kurios bus naudojamos administracinės sistemos kūrimui analizė.....	16
3. Projektas.....	21
4. Dokumentacija naudotojui	44
Išvados	45
Literatūros sąrašas	46
Informacijos šaltinių sąrašas	49
Priedai (pagal poreikį)	51

Lentelių sąrašas

Lentelė 1 Panašių sistemų palyginimas	15
Lentelė 2 Serverio dalies karkasų palyginimas	17
Lentelė 3 Kliento dalies karkasų palyginimas	19
Lentelė 4 Duomenų bazių palyginimas	20
Lentelė 5 Prisijungimo veiklos diagrama	23
Lentelė 6 Vartotojų valdymo veiklos diagrama	24
Lentelė 7 Mosleivių sąrašo įvedimo veiklos diagrama	25
Lentelė 8 Klasijų sudarymo veiklos diagrama	26
Lentelė 9 Tvarkaraščio sudarymo veiklos diagrama	27
Lentelė 10 Funkciniai reikalavimai	28
Lentelė 11 Nefunkciniai reikalavimai	28
Lentelė 12 Klasijų diagramos paaiškinimas (1).....	30
Lentelė 14 Duomenų bazės projekto paaiškinimas	31

Paveikslų sąrašas

Pav. 3.1 Konceptija (Klasių sudarymas).....	21
Pav. 3.2 Konceptija (Tvarkaraščio sudarymas)	21
Pav. 3.3 Konceptija (Kitos funkcijos).....	22
Pav. 3.4 Panaudos atvejų diagrama	22
Pav. 3.5 PA-01 Prisijungimas.....	23
Pav. 3.6 PA-02 Vartotojų valdymas	24
Pav. 3.7 PA-03 Moksleivių sąrašo įvedimas	25
Pav. 3.8 PA-04 Klasių sudarymas	26
Pav. 3.9 PA-05 Tvarkaraščio sudarymas.....	27
Pav. 3.10 Klasių diagrama.....	29
Pav. 3.13 Tvarkaraščio sudarymo sekos diagrama.....	30
Pav. 3.14 Duomenų bazės projektas.....	31
Pav. 3.16 Vartotojo sąsajos struktūra	32
Pav. 3.17 Langas - Prisijungimas	33
Pav. 3.18 Langas – Slaptažodžio atstatymas	34
Pav. 3.19 Langas – Pagrindinis sistemos langas (neprisijungus)	35
Pav. 3.20 Langas – Pagrindinis sistemos langas (prisijungus).....	36
Pav. 3.21 Langas – Mokytojai.....	37
Pav. 3.22 Langas – Moksleiviai	38
Pav. 3.23 Langas – Klasės.....	39
Pav. 3.24 Langas – Srautai	40
Pav. 3.25 Langas – Tvarkaraštis.....	41
Pav. 3.26 Langas – Moksleivio paskyrimas	42
Pav. 3.27 Diegimo diagrama	43

Santrumpų ir terminų sąrašas

Santrumpos:

Lekt. – lektorius;

Doc. – docentas;

Prof. – profesorius.

Terminai:

Saityno analitika – lorem ipsum dolor sit amet, eam ex decore persequeris, sit at illud lobortis atomorum. Sed dolorem quaerendum ne, prompta instructor ne pri. Et mel partiendo suscipiantur, docendi abhorreant ea sit. Recteque imperdiet eum te.

Tinklaraštis – lorem ipsum dolor sit amet, eam ex decore persequeris, sit at illud lobortis atomorum. Sed dolorem quaerendum ne, prompta instructor ne pri. Et mel partiendo suscipiantur, docendi abhorreant ea sit. Recteque imperdiet eum te.

Ivadas

Paskutiniais dešimtmečiais, žmonės vis labiau ieško būdų, kaip supaprastinti ir palengvinti procesus, kurie kelia daug iššūkių ir naudoja daug laiko jiems vykdyti. Dažnai įmonės šiems procesams valdyti bei problemas spręsti naudoja kompiuterines sistemas. Ne išimtis ir mokyklos, kurios susiduria su daugybe problemų, kurias galėtų padėti spręsti kompiuterinės sistemos.

Vienos didesnių ir sudėtingesnių problemų mokyklose – administracinės, nes yra didelis kiekis moksleivių bei mokytojų, o kartu ir didelis duomenų kiekis apie juos. Visa tai tvarko ir apdoroja mokyklos administracija. Tačiau iki šios dienos įvairūs administraciniai procesai, kaip naujų moksleivių skirstymas į klases ar pamokų tvarkaraščio sudarymas, vis dar yra atliekami rankiniu būdu ir neretai užtrunka ilgai bei juos atliekant lieka klaidų. Pritaikant tam tikras sistemas būtų galima palengvinti ir pagreitinti mokyklų administravimo procesą.

Darbo tikslas – suprojektuoti ir sukurti administracinės sistemos prototipą, kuris palengvintų bei pagreitintų tam tikrus administracinius procesus mokykloje.

Darbo uždaviniai:

1. Išanalizuoti mokyklos administracinių darbų specifiką.
2. Išsiaiškinti administracinių sistemų veikimo principus, atlikti panašių informacinių sistemų bei technologijų, kurios bus naudojamos naujos sistemos kūrimui, analizę.
3. Paruošti administracinės sistemos projektą.
4. Parengti vartotojo vadovą.

1. Analizė

Šiuo metu mokykloje administravimui nėra naudojama jokia sistema. Mokyklos administracijai tenka rankiniu būdu suskirstyti naujus mokinius į klases, kas užtrunka labai daug laiko. Taip pat suskirsčius moksleivius duomenys yra perduodami registru, iš kurio moksleivių informacija yra patalpinama į manodienynas.lt ir moodle sistemas, tačiau šis procesas dažnai vėluoja ir mokinius suvesti tenka rankiniu būdu.

Kita stambi problema, su kuria yra susiduriama — tvarkaraščio sudarymas. Net kompiuteriui ši problema yra sudėtinga ir užima nemažai laiko, o mokyklos administracijai sudarant tvarkaraštį rankiniu būdu, tai gali užtrukti iki kelių savaičių. Taip pat yra paliekama klaidų, todėl net prasidėjus mokslo metams dažnai tvarkaraštis dar keičiasi.

1.1. Panašios administracinės sistemos mokykloms

Sistemų analizei pagrindiniai kriterijai yra:

1. Galimybė moksleivius skirstyti į klases/srautus
2. Galimybė moksleivių sąrašą suvesti į manodienynas.lt ir moodle sistemas
3. Galimybė sudaryti tvarkaraštį
4. Galimybė redaguoti tvarkaraštį
5. Sistemos kalba

1.1.1. Timetable Generator GHC

Programa skirta tvarkaraščio sudarymui. Tačiau norint sudaryti tinkamą tvarkaraštį reikia suvesti labai daug duomenų bei apribojimų. Taip pat ši sistema veikia tik lokaliaje įrenginyje, todėl neįmanoma mokyklos mokytojams pateiktų jų asmeninius tvarkaraščius. Sistemoje nėra galimybės taisyti tvarkaraštį, reikia pakeitus nustatymus generuoti iš naujo. Taip pat kadangi ši sistema yra skirta tik tvarkaraščio sudarymui, joje nėra galimybės skirstyti moksleivius į klases bei srautus, taip pat suvesti į naudojamas kitas sistemas. Taip pat didelis trūkumas, sistema tik anglų kalba.

1.1.2. Prime Timetable

Programa taip pat skirta tik tvarkaraščio sudarymui. Šia sistema norint sugeneruoti tvarkaraštį taip pat reikia suvesti daugybę duomenų iš anksto. Tačiau priešingai nei ankstesnė sistema, sugeneruotą tvarkaraštį galima redaguoti ir jį redaguojant yra gaunamas grįžtamasis ryšys ar nesikerta dalykas su kitu dalyku tuo laiku. Sistemoje nėra galimybės skirstyti mokinius. Sistema veikia internetinėje erdvėje, todėl mokytojai gali peržiūrėti jiems paskirtus tvarkaraščius. Tačiau sistema yra gana brangi ir tik anglų kalba.

1.1.3. Alma

Sistema skirta mokyklų administravimui. Šia sistema galima rankiniu būdu sudaryti tvarkaraštį bei jį redaguoti. Sistemoje yra galimybė skirstyti moksleivius į klases. Tačiau sistemoje yra labai daug neaktualių funkcijų dėl kurių sistema yra labai didelė ir nepatogi naudotis. Taip pat sistemos kainą reikia derinti su kūrėju ir sistema yra anglų klaba.

1.2. Analizuotų sistemų palyginimas su kuriama sistema

	Dabar mokykloje (Excel)	Timetable Generator GHC	Prime Timetable	Alma	Kuriama sistema
Moksleivių skirstymas į klases/srautus	Rankiniu būdu, nenaudojant papildomų technologijų	Nėra	Nėra	Yra	Yra
Moksleivių suvedimas į manodienynas.lt ir moodle sistemas	Rankiniu būdu arba laukiant, kol bus surašyti iš registro	Nėra	Nėra	Nėra	Yra
Tvarkaraščio sudarymas	Rankiniu būdu, nenaudojant papildomų technologijų	Generuojamas automatiškai	Generuojamas automatiškai	Rankiniu būdu	Yra
Tvarkaraščio redagavimas	Rankiniu būdu, neužtikrinant, kad nebus įvelta naujų klaidų	Reikia generuoti iš naujo	Yra	Yra	Yra
Sistemos kalba	Lietuvių	Anglų	Anglų	Anglų	Lietuvių

Lentelė 1 Panašių sistemų palyginimas

2. Technologijų, kurios bus naudojamos administracinės sistemos kūrimui analizė

Mokykla šiuo metu naudoja internetinių svetainių talpinimo paslauga „Hostex“ platformoje. Tai apriboja galimus serverio dalies (angl. back-end) bei vartotojo sąsajos (angl. front-end) programavimo kalbų bei karkasų pasirinkimus. Šiuo atveju serveriui galima naudoti tik PHP programavimo kalba paremtus karkasus.

2.1. PHP karkasai skirti aplikacijos serverio pusei realizuoti

Pagrindiniai reikalavimai serverio daliai PHP karkasams:

1. Bendruomenės dydis (populiarumas)
2. Plėtinių skaičius
3. Vidutinis atsako greitis
4. Asmeninė patirtis

2.1.1. Lumen

„Laravel“ yra galingas karkasas, kurį nėra sudėtinga naudoti. Dėl ganėtinai nesudėtingos sintaksės, kuri padeda nesudėtingai struktūrizuoti projekto kodą bei „Eloquent ORM“ naudojimo, kuris susieja duomenų bazės lenteles su sistemoje veikiančiu „modeliu“, šis karkasas yra vienas akivaizdžiausių pasirinkimų realizuojant tiek didelį, tiek mažą projektą. Dėl daugybės nemokamai prieinamų plėtinių, ganėtinai griežtos struktūros, ir neįtikėtinai trumpo užklausų, kurių karkasas gali „atlaikyti“ net iki šimtų milijonų per mėnesį, atsako laiko, projektus realizuotus šiuo karkasu yra nesudėtinga plėsti. Tačiau vienas didžiausių „Laravel“ karkaso privalumų yra bendruomenės dydis. Viena populiariausių programuotojų siekiantiems pagalbos puslapių „Stack Overflow“ klausimų su žyma „laravel“ yra per 170 tūkstančių, kas indikuoja, kad su bendruomenės pagalba atsakymą į iškilusį klausimą rasti bus ganėtinai nesudėtinga.

„Lumen“ yra „Laravel“ karkasu paremtas mikro karkasas sukurtas specialiai aplikacijų programavimo sąsajai (API) kurti. Pagrindinis „Lumen“ karkaso tikslas yra kuo įmanoma labiau padidinti atsako greitį, išsaugant esmines „Laravel“ karkaso savybes, todėl didžiausi plusai persiduoda ir „Lumen“ karkasui. Didžiausi skirtumai tarp „Lumen“ ir „Laravel“ karkasų yra tai, jog „Laravel“ yra vadinamasis „full-stack“ arba serverio bei kliento dalių karkasas, o „Lumen“ yra tik serverio dalies. Tačiau tai reiškia, kad šis karkasas yra smulkesnis, veikia greičiau ir gali apdoroti net dar daugiau užklausų, nei „Laravel“ (Padamkar, be datos). Taigi, pagrindinis klausimas išlieka, kaip greitai „Lumen“ gali apdoroti užklausas. Atlikus testavimą, buvo nustatyta, jog vidutinis šalto starto, kai serveris nėra pradėjęs vykdyti sistemos operacijų, buvo 140.17ms, sunaudojant 87mb atminties, kai tuo tarpu „Laravel“ užtruko net 3 kartus ilgiau ir sunaudojo 100mb atminties (Iglis, 2020).

Tačiau kartu su plusais, „Lumen“ karkasas paveldi ir „Laravel“ karkaso trūkumus. „Eloquent ORM“ yra galingas įrankis, tačiau, kadangi tai yra beveik „dieviškoji“ klasė, kuri gali labai palengvinti darbą, kam nors neteisingai suveikus, gali būti labai sudėtinga surasti, kur slypi klaida, mes labai didelės „Eloquent“ veikimo dalies mes nematome. Kitas trūkumas yra, kadangi „Lumen“ yra mikro karkasas viena populiariausių „Laravel“ funkcijų, tai yra vartotojų autentifikavimas bei autorizavimas naudojant „passport“ buvo pašalintas, bei negalėtų būti naudojamas, kadangi „Lumen“ nepalaiko sesijos. Todėl norint realizuoti vartotojų autentifikaciją bei autorizaciją, tenka naudoti trečiųjų šalių sprendimus arba realizuoti patiems.

2.1.2. Guzzle

Karkasas sukurtas HTTP užklausoms ir mikro servisų integracijai. Šis karkasas yra vienas populiariausių siekiant realizuoti REST API dėl savo plataus pritaikymo galimybių bei ganėtinai nesudėtingos implamentacijos. „Guzzle“ gali gerokai palengvinti užklausų apdorojimą. Su tinkamomis tarpinėmis klasėmis, „Guzzle“ konstruktorius, naudojant JWT žetoną autentifikacijai, gali nesudėtingai nustatyti ar vartotojui turėtų būti suteikta prieiga, ar galioja naudojamas žetonas ir jei reikia, jį atnaujinti (Brad, 2017). Kita didelė šio karkaso stiprybė yra galimybė perduoti ir apdoroti didelius duomenų srautus, įkeliant bei parsisiunčiant duomenis. Taip pat šis karkasas palaiko tiek sinchronines, tiek asinchronines užklausas, naudojant tą pačią sąsają.

Tačiau, šis karkasas yra ganėtinai abstraktus ir nemaža dalis kodo yra interpretuojama. Tai sukelia riziką, jog programa gali veikti nenuspėjamai ar ne taip, kaip yra siekiama. Kiek kitokiame karkaso testavime, nei „Lumen“ karkasui, buvo atliekama 30000 nesudėtingų užklausų, kurias „Guzzle“ apdorojo per 1.8905s (Evgeniy Zyubin, Tobias Nyholm, 2021). Šis testas buvo atliekamas ir karkasui, kuris bus nagrinėjamas po šio. Pasak puslapio „Stack Overflow“ šio karkaso bendruomenė yra gerokai mažesnė, nei „Laravel“, puslapyje užduota šiek tiek daugiau nei 2000 klausimų.

2.1.3. Slim Framework

Karkasas sukurtas web aplikacijoms ir API kūrimui. Šis karkasas yra minimalistiškiausias iš trijų ir neabejotinai reikalauja daugiausia darbo norint naudotis. Šis karkasas yra panašesnis į pagalbą PHP programuotojui, o ne pilną karkasą. Tačiau tai padaro jį labai lanksčiu ir labai geru pasirinkimu patyrusiems PHP programuotojams, kurie norėdami dirbti efektyviau, gali pasitelkti šį karkasą. Kadangi šis karkasas reikalauja nemažai patirties turinčių programuotojų, nenuostabu, jog puslapyje „Stack Overflow“ su žyma apie šį karkasą, klausimų yra kiek mažiau nei 3000. To paties testavimo, kaip ir „Guzzle“ karkaso metu, taip pat atliekant 30000 užklausų, šis karkasas su jomis susidorojo per 2.4757s, kas leidžia suprasti, jog šis karkasas yra gerokai lėtesnis už „Guzzle“.

2.1.4. Apibendrinimas

	Lumen	Guzzle	Slim Framework
Bendruomenės dydis (populiarumas)	Didelė	Maža	Maža
Plėtinių skaičius	Didelis	Didelis	Mažas
Vidutinis atsako greitis	Didelis	Didelis	Didelis
Asmeninė patirtis	Vidutinė	Nėra	Nėra

Lentelė 2 Serverio dalies karkasų palyginimas

Nuspręsta, kad naudojamas bus „Lumen“ karkasas.

2.2. Vartotojo sąsajai realizuoti skirti karkasai

Naudojamame serveryje, vartotojo sąsajai realizuoti galima pasitelkti bet kokią HTML ir JavaScript/TypeScript karkasą. Siekiant užtikrinti kuo įmanoma patogesnę sąsają naudojamas bus vieno puslapio aplikacijos karkasas.

Pagrindiniai reikalavimai karkasams:

1. TypeScript palaikymas
2. Sudėtingumas išmokti
3. Būsenų valdymo sistema
4. Asmeninė patirtis

2.2.1. React

Kompanijos „Facebook“ sukurtas karkasas, yra vienas populiariausių pasaulyje, šiuo metu. „React“ yra JavaScript kalba sukurtas vieno puslapio aplikacijos karkasas, kurio veikimas paremtas komponentų sistema, kai kiekviena sistemos dalis yra atskiras komponentas. Tai gali būti nuo viso puslapio maketo iki vieno mygtuko, kuris bus naudojamas keliose vietose, taip leisdamas vartotojo sąsajai būti dinamiškesnei. „React“ pasitelkia vartotojo aplikacijos atmintyje esantį virtualų objektų-dokumentų modelį (vDOM), kuris yra sinchronizuojamas su realiu objektų-dokumentų modeliu (DOM) pasitelkiant biblioteką, kaip „ReactDOM“. Tai leidžia realiajam objektų-duomenų modeliui būti manipuliuojamu neperkraunant puslapio. Kas kart įvykus pakeitimui vDOM pasitelkiant DOM manipuliavimo biblioteką yra atnaujinamas ir realusis DOM (Emekoma, 2021).

Karkase yra naudojama būsenų valdymo sistema. Ši sistema leidžia realizuoti aplikacijai veikti kaip įprastai kompiuterinei programai, kurioje įvykus pakitimui nereikia perkrovinti lango ir yra iš kart parodomas pranešimas (Buyse, 2020).

Vienas didžiausių šio karkaso trūkumų yra sudėtingumas naudoti neturint patirties. Karkase yra naudojama sintaksė, kuri nėra naudojama kituose karkasuose. Dažnai yra panaši į įprastą HTML, kas reiškia, kad suvokti kodą nėra itin sudėtinga, tačiau parašyti jį gali būti šioks toks iššūkis.

2.2.2. Vue.js

Vienas jaunesnių karkasų, sukurtas ir palaikomas bendruomenės. Beveik visomis prasmėmis yra labai panašus į „React“ karkasą. Tačiau priešingai nei „React“, šį karkasą įvaldyti yra ganėtinai nesudėtinga. Tam didelė dalimi įtaką daro, jog sintakse ir funkcijomis rūpinasi bendruomenė, o ne viena korporacija. Todėl šiame karkase sintaksė yra beveik identiška įprastai HTML. Šis karkasas yra labai responsyvus ir greitai veikiantis. Taip pat didelis plusas yra tai, jog nuo trečiosios šio karkaso versijos, jis yra parašytas naudojant TypeScript ir jame TypeScript yra naudojama, kaip numatytoji kalba be jokių papildomų konfigūracijų. Tai padeda apsaugoti kuriamą aplikaciją nuo nenumatytų duomenų įvedimo.

Taip pat „Vue.js“ gali veikti pasitelkiant API, kas leidžia šiam karkasui būti labai dinamiškam ir nesunkiai būti integruojamam su kitais karkasais, kaip „Laravel“ ar net „React“ (Vue.js: Benefits and Drawbacks, 2021).

2.2.3. Angular

„Google“ sukurtas karkasas plačiai naudojamas jų kurtuose produktuose. Kaip ir anksčiau minėti karkasai, „Angular“ yra vieno puslapio aplikacijoms kurti skirtas karkasas. Parašytas naudojant TypeScript, bei šią kalbą naudoja kaip numatytąją, kas padaro aplikacijas gana saugiomis nuo

netinkamų duomenų tipų. Tačiau, kaip ir „React“ tai yra korporacijos vidinis produktas ir nors ir yra ganėtinai populiarus, išmokti naudoti šį karkasą gali būti ganėtinai sudėtinga. Taip pat jis yra pritaikytas dideliems projektams, todėl kuriamas aplikacijas yra nesudėtinga išplėsti, tačiau juo naudotis yra patogiausia su didesne komanda. Dirbant vienam arba nedidelėje komandoje, susitvarkyti su kodu gali būti kiek sudėtinga.

2.2.4. Apibendrinimas

	React	Vue.js	Angular
TypeScript palaikymas	Yra	Yra	Yra
Sudėtingumas išmokti	Didelis	Mažas	Vidutinis
Būsenų valdymo sistema	Yra, diegiama papildomai	Yra, diegiama papildomai	Yra
Asmeninė patirtis	Maža	Didelė	Nėra

Lentelė 3 Kliento dalies karkasų palyginimas

Nuspręsta, kad naudojamas bus „Vue.js“ karkasas.

2.3. Duomenų bazės

Pagrindiniai kriterijai duomenų bazėms:

1. Saugumas
2. Mokyklos jau naudojama
3. Ar duomenų bazė reliacinė

2.3.1. MySQL

Kadangi buvo nuspręsta dirbti su „Lumen“ karkasu, ši duomenų bazė tam tikrų idealiai, nes yra viena iš numatytųjų „Lumen“ palaikomų duomenų bazių ir su ja kuriant projektą nereikalinga papildoma konfigūracija. Tai yra reliacinė duomenų bazė. Ši duomenų bazė yra viena senesnių, sukurta 1994 metais, nuo 2010 metų priklauso kompanijai „Oracle“. Tai yra didelis plusas, nes per tokį ilgą laiką, buvo pašalinta daugybė pažeidžiamumų. Taip pat grafinė sąsaja „phpMyAdmin“ yra labai patogi ir lengvai naudojama. Taip pat šią duomenų bazę mokykla naudoja kartu su savuoju puslapiu. Vienas didžiausių šios duomenų bazės trūkumų yra ribotos plečiamumo galimybės, nes jei duomenų bazėje yra labai daug duomenų, ji gali tapti lėta ir nepatikima.

2.3.2. PostgreSQL

Kaip ir „MySQL“, tai yra reliacinė duomenų bazė. Priešingai, nei „MySQL“, ši duomenų bazė yra visiškai atvirojo kodo, todėl pasižymi gana neblogu saugumu. Naudojantis šia duomenų baze, sintaksė gali pasirodyti sudėtinga. Nors grafinė sąsaja yra ganėtinai nuvilianti, kaip ir „MySQL“, ši

duomenų bazė veikia be papildomos konfigūracijos su „Lumen“ karkasu. Taip pat ji geriau susitvarko su dideliais duomenų kiekiais.

2.3.3. MongoDB

Priešingai nei anksčiau minėtosios, ši duomenų bazė yra dokumentinė, tai gali ją padaryti greičiausia iš trijų, nes ji geriausiai susitvarko su dideliais duomenų kiekiais. Kadangi ši duomenų bazė naudoja NoSQL schemas bei užklaudas, norint naudoti šią duomenų bazę kartu su „Lumen“ karkasu, reikalingi trečiųjų šalių plėtiniai (Almeida, 2017). Taip pat šios duomenų bazės saugumas yra vertinamas kaip gana žemas (Preecha Noiumkar, Tawatchai Chomsiri, 2014).

2.3.4. Apibendrinimas

	MySQL	PostgreSQL	MongoDB
Saugumas	Vidutinis	Vidutinis	Mažas
Naudojama mokykloje	Taip	Ne	Ne
Reliacinė	Taip	Taip	Ne

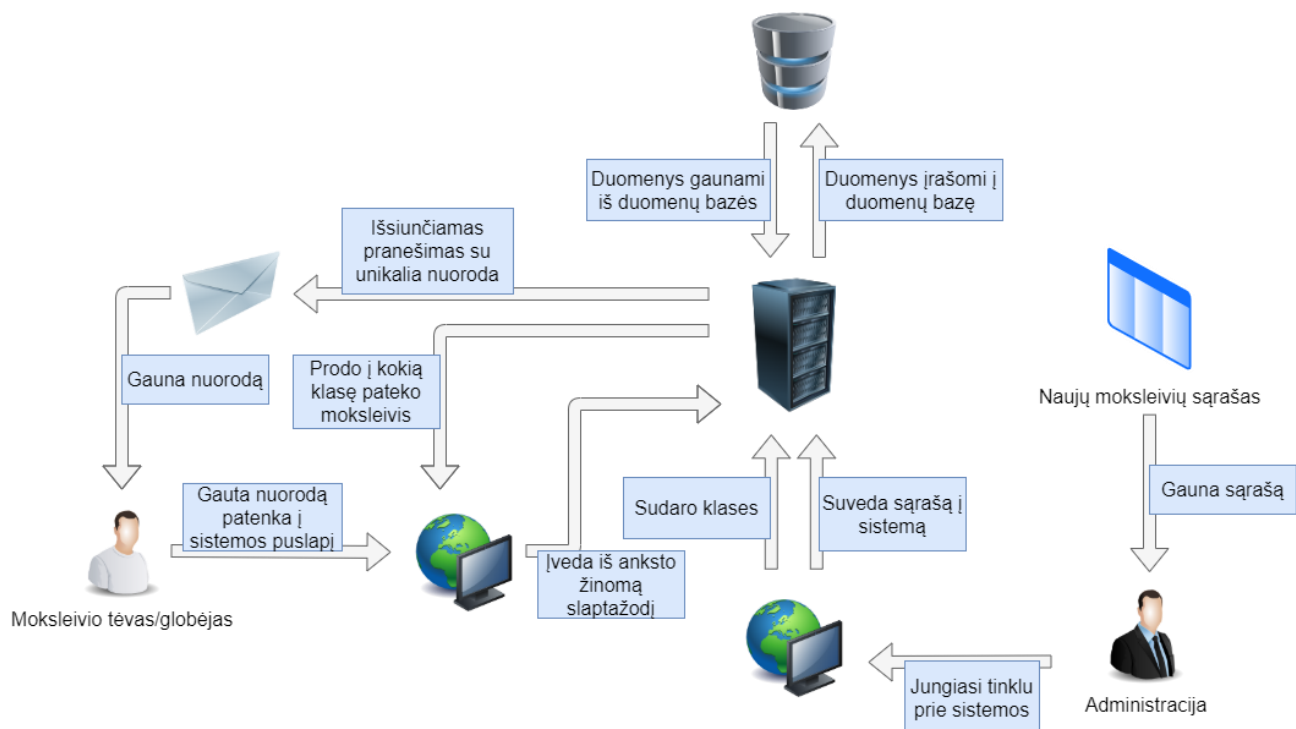
Lentelė 4 Duomenų bazių palyginimas

2.4. Vartotojų autentifikavimas

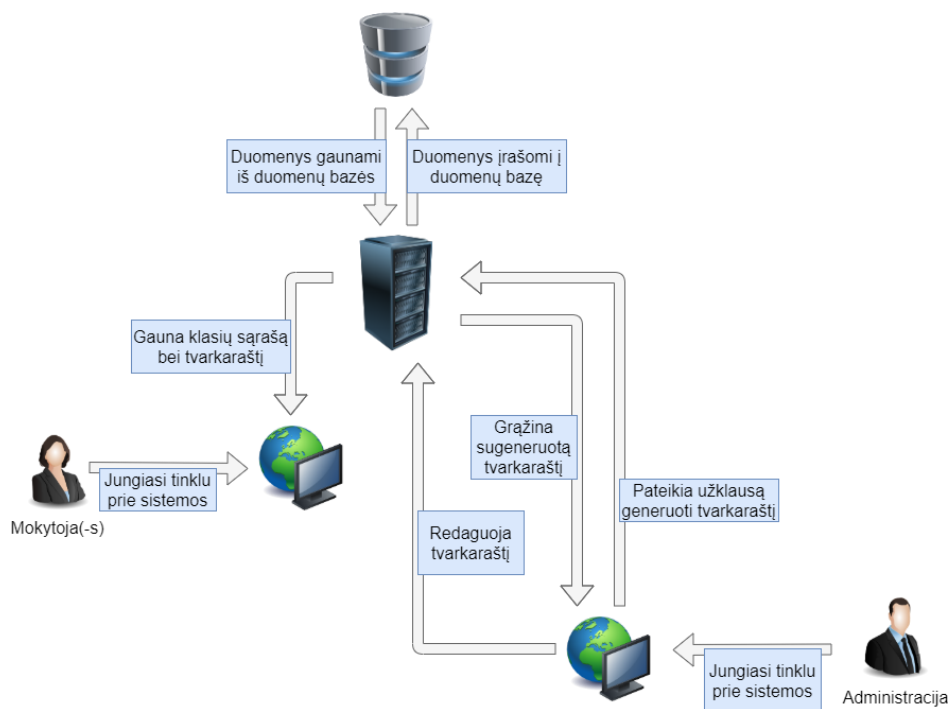
„Lumen“ karkase nėra numatytosios vartotojų autentifikavimo galimybės, todėl reikia pasitelkti trečiųjų šalių realizaciją. Vienas paprasčiausių autentifikavimo realizacijos būdų šiuo atveju yra naudoti OAuth sistemą. Autentifikavimo paslaugų tiekėjų yra keletas, tačiau tik „auth0“ suteikia galimybę naudotis šia paslauga nemokamai iki 7000 vartotojų. Taip pat jis yra lengvai suderinamas su pasirinktais naudoti karkasais „Lumen“ ir „Vue.js“. Ši paslauga taip pat pasirūpina vartotojų autorizavimu, kas yra naudinga, jei kuriamoje sistemoje planuojama turėti kelias skirtingas vartotojų roles. Naudojant „auth0“ taip pat yra užtikrinamas saugumas bei yra galimybė naudoti kelių žingsnių prisijungimo funkciją.

3. Projektas

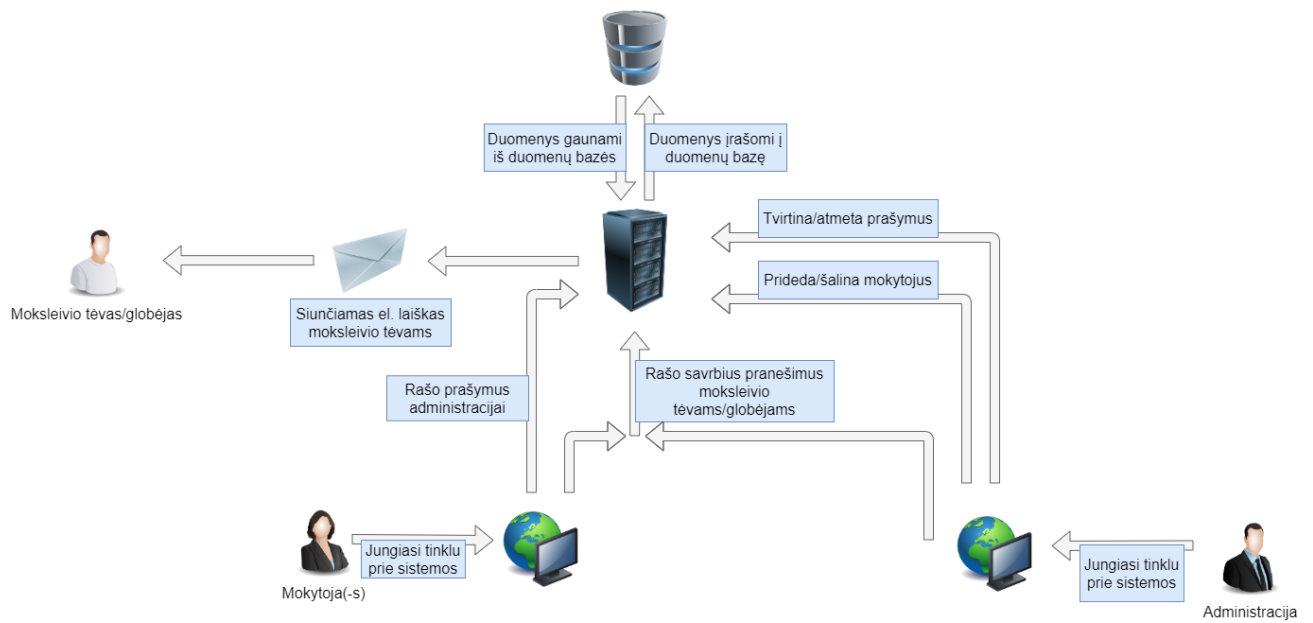
3.1. Konceptcija



Pav. 3.1 Konceptcija (Klasių sudarymas)



Pav. 3.2 Konceptcija (Tvarkaraščio sudarymas)



Pav. 3.3 Konceptija (Kitos funkcijos)

3.2. Sistemos reikalavimų analizė

3.2.1. Panaudos atvejų diagramos



Pav. 3.4 Panaudos atvejų diagrama

ID	PA-01
Pavadinimas	Prisijungimas
Aprašymas	Sistemos vartotojas prisijungia, įveddamas savo elektroninį paštą ir slaptažodį.
Aktoriai	Mokytojas, Administracijos narys
Pradinės sąlygos	Vartotojas privalo būti registruotas
Pagrindiniai žingsniai	Prisijungimo lange vartotojas įveda savo elektroninį paštą, slaptažodį ir paspaudžia mygtuką „Prisijungti“
Alternatyvūs žingsniai	Nėra
Išskirtinės situacijos	Jeigu vartotojas įveda neteisingus prisijungimo duomenis vartotojui yra parodomas klaidos pranešimas.
Galutinės sąlygos	Įvesti prisijungimo duomenys yra teisingi, vartotojui suteikia prieigą prie sistemos
Veiklos diagrama	<pre> graph TD Start(()) --> A[Įsijungia prisijungimo puslapį] A --> D1{ } Note1["[siminti prisijungimo duomenys]"] -.-> D1 D1 -- Ne --> B[Įveda prisijungimo duomenis] D1 -- Taip --> F(()) B --> D2{ } Note2["[vesti duomenys teisingi]"] -.-> D2 D2 -- Ne --> C[Parodo klaidos pranešimą] C --> B D2 -- Taip --> F </pre> <p style="text-align: center;">Pav. 3.5 PA-01 Prisijungimas</p>

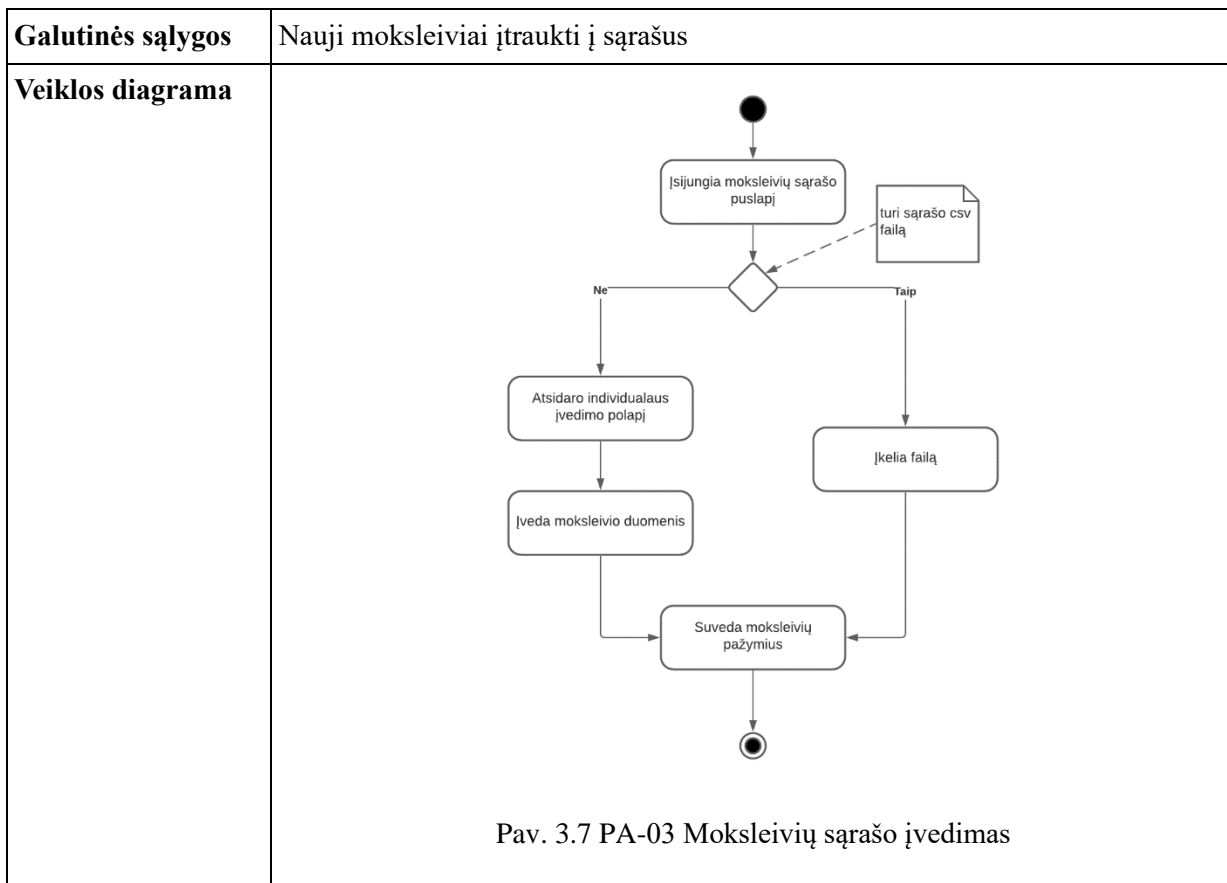
Lentelė 5 Prisijungimo veiklos diagrama

ID	PA-02
Pavadinimas	Vartotojų valdymas
Aprašymas	Valdomi vartotojų duomenys bei įtraukiami nauji vartotojai
Aktoriai	Administracijos narys
Pradinės sąlygos	Vartotojus valdyti gali tam teisę turintys vartotojai

Pagrindiniai žingsniai	Vartotojų valdymo lange pasirenkamas vartotojas ir redaguojami jo duomenys
Alternatyvūs žingsniai	Įvedami naujo vartotojo duomenys
Išskirtinės situacijos	Jeigu paliekami tušti ar neteisingai užpildyti laukai, yra parodomas klaidos pranešimas.
Galutinės sąlygos	Yra pakeisti vartotojo duomenys
Veiklos diagrama	<pre> graph TD Start(()) --> Join([Įsijungia vartotojų valdymo sritį]) Join --> Decision{ } Note1[Nori registruoti naują vartotją] -.-> Decision Note2[Nori redaguoti vartotją] -.-> Decision Decision --> Input([Įveda informaciją apie vartotją]) Decision --> Edit([Atsidaro informaciją apie konkretų vartotją]) Input --> Assign([Paskiria rolę]) Edit --> Update([Atlieka pakeitimus]) Assign --> End((())) Update --> End </pre> <p>Pav. 3.6 PA-02 Vartotojų valdymas</p>

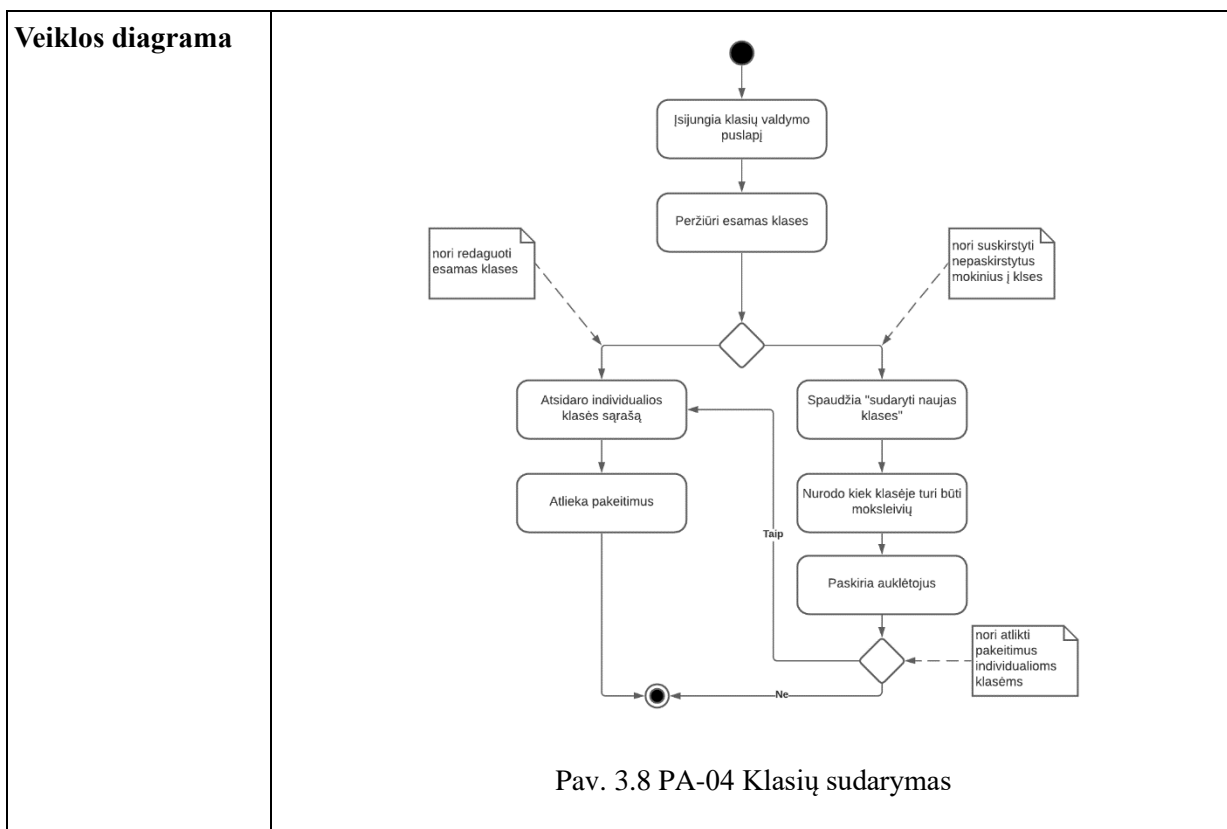
Lentelė 6 Vartotojų valdymo veiklos diagrama

ID	PA-03
Pavadinimas	Moksleivių sąrašo įvedimas
Aprašymas	Įvedami nauji moksleiviai į sistemą
Aktoriai	Administracijos narys
Pradinės sąlygos	Vartotojas privalo būti administracijos narys
Pagrindiniai žingsniai	Atsidaręs moksleivių sąrašo puslapį vartotojas mato į sistemą jau įvestus moksleivius, norėdamas pridėti naują moksleivį vartotojas spaudžia mygtuką pridėti ir suveda reikiamus duomenis.
Alternatyvūs žingsniai	Jei turi failą su moksleivių sąrašu, gali jį įkelti į sistemą
Išskirtinės situacijos	Jeigu paliekami tušti arba neteisingai užpildyti laukai, išvedamas klaidos pranešimas



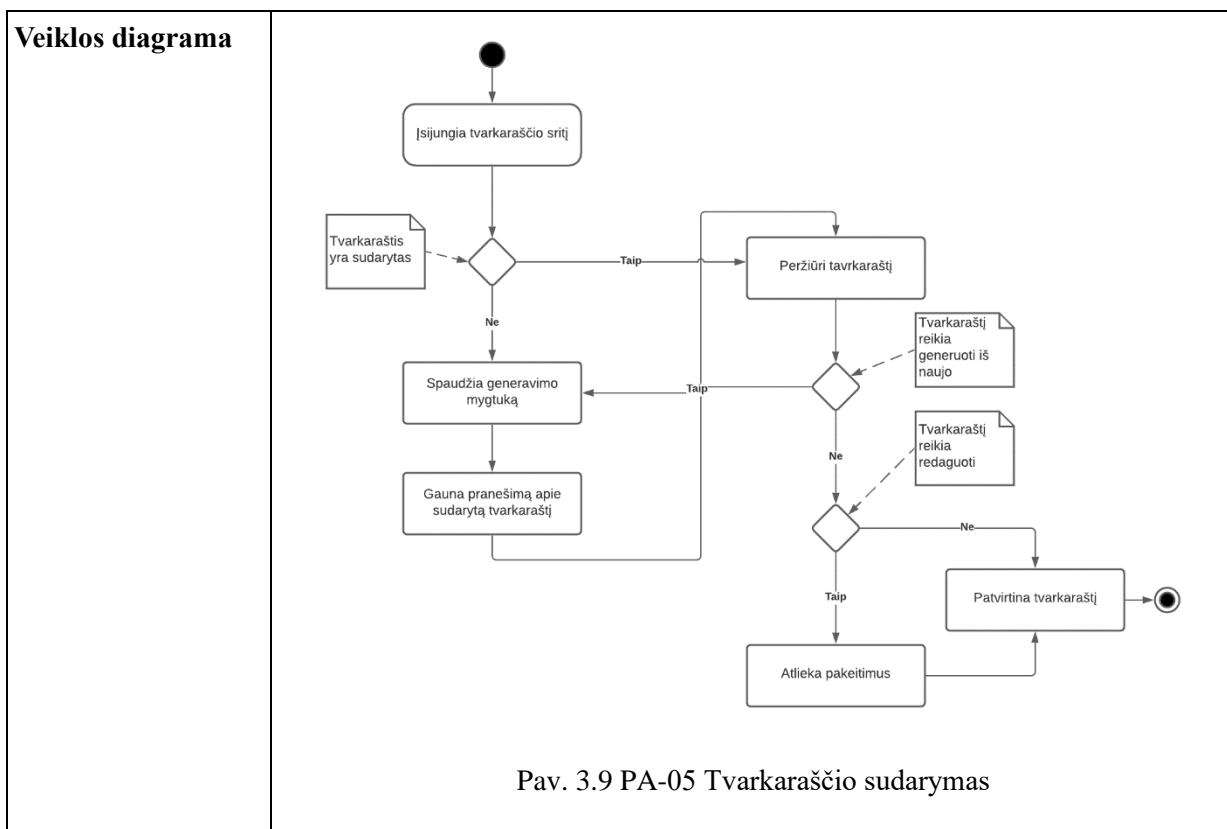
Lentelė 7 Moksleivių sąrašo įvedimo veiklos diagrama

ID	PA-04
Pavadinimas	Klasių sudarymas
Aprašymas	Moksleiviai suskirstomi į klases, klasėms priskiriami auklėtojai
Aktoriai	Administracijos narys
Pradinės sąlygos	Vartotojas privalo būti administracijos narys
Pagrindiniai žingsniai	Vartotojas atsidaręs klasių redagavimo skiltį, peržiūri esamas klases ir spaudžia „sudaryti naujas klases“ mygtuką, nurodo kiek moksleivių turi būti klasėse. Sistema iš įvestų moksleivių, kurie nėra suskirstyti, sudaro klasių sąrašus, vartotojas jiems priskiria auklėtojus, tuomet gali patvirtinti arba redaguoti individualių klasių sąrašus.
Alternatyvūs žingsniai	Vartotojas atsidaro esamos klasės sąrašą, jame atlieka pakeitimus.
Išskirtinės situacijos	Jeigu priskiriamas auklėtojas jau turi auklėjamą klasę, išvedamas klaidos pranešimas
Galutinės sąlygos	Moksleiviai suskirstyti į klases



Lentelė 8 Klasių sudarymo veiklos diagrama

ID	PA-05
Pavadinimas	Tvarkaraščio sudarymas
Aprašymas	Administracija sudaro pamokų tvarkaraštį
Aktoriai	Administracijos narys
Pradinės sąlygos	Vartotojas privalo būti administracijos narys
Pagrindiniai žingsniai	Vartotojas atsidaręs tvarkaraščio sudarymo skiltį, jei tvarkaraštis nėra sudarytas spaudžia mygtuką „Generuoti“, tuomet sulaukęs pranešimo apie sudarytą tvarkaraštį, gali jį redaguoti
Alternatyvūs žingsniai	Jei tvarkaraštis jau sudarytas, vartotojas gali jį redaguoti
Išskirtinės situacijos	Jeigu mokytojui dubliuojasi pamokos arba ne visoms klasėms yra paskirtos visos pamokos, rodomas klaidos pranešimas
Galutinės sąlygos	Sudarytas patvirtintas pamokų tvarkaraštis



Lentelė 9 Tvardarašcio sudarymo veiklos diagrama

3.2.2. Funkciniai reikalavimai

Nr.	Reikalavimas	Susiję panaudos atvejai	Sudėtingumas	Nuoroda į papildomą informaciją
FR-01	Vartotojai turi galėti prisijungti prie sistemos	PA-01	Nesudėtingas	-
FR-02	Administracija turi turėti galimybę registruoti naujus vartotojus bei juos redaguoti	PA-02	Vidutinis	-
FR-03	Naujų mokinių sąrašą turi būti įmanoma įkelti iš failo	PA-03	Vidutinis	-
FR-04	Naujus moksleivius turi būti įmanoma įvesti bei juos redaguoti rankiniu būdu	PA-03	Vidutinis	-
FR-05	Moksleiviai turi būti suskirstomi į klases automatiškai	PA-04	Sudėtingas	-
FR-06	Klasės turi būti priskiriamos mokytojams	PA-04	Nesudėtingas	-
FR-07	Kabinetai priskiriami mokytojams	PA-02	Nesudėtingas	-

FR-08	Tvarkaraščio sudarymas bei redagavimas	PA-05	Sudėtingas	-
FR-09	Tvarkaraščio eksportavimas nurodytu formatu	PA-05	Vidutinis	-
FR-10	Prašymų rašymas	-	Nesudėtingas	-
FR-11	Automatinis moksleivių tėvų informavimas elektroniniu paštu	-	Vidutinis	-
FR-03	Įvykus klaidai vartotojui rodomi klaidos pranešimai.	-	Vidutinis	-

Lentelė 10 Funkciniai reikalavimai

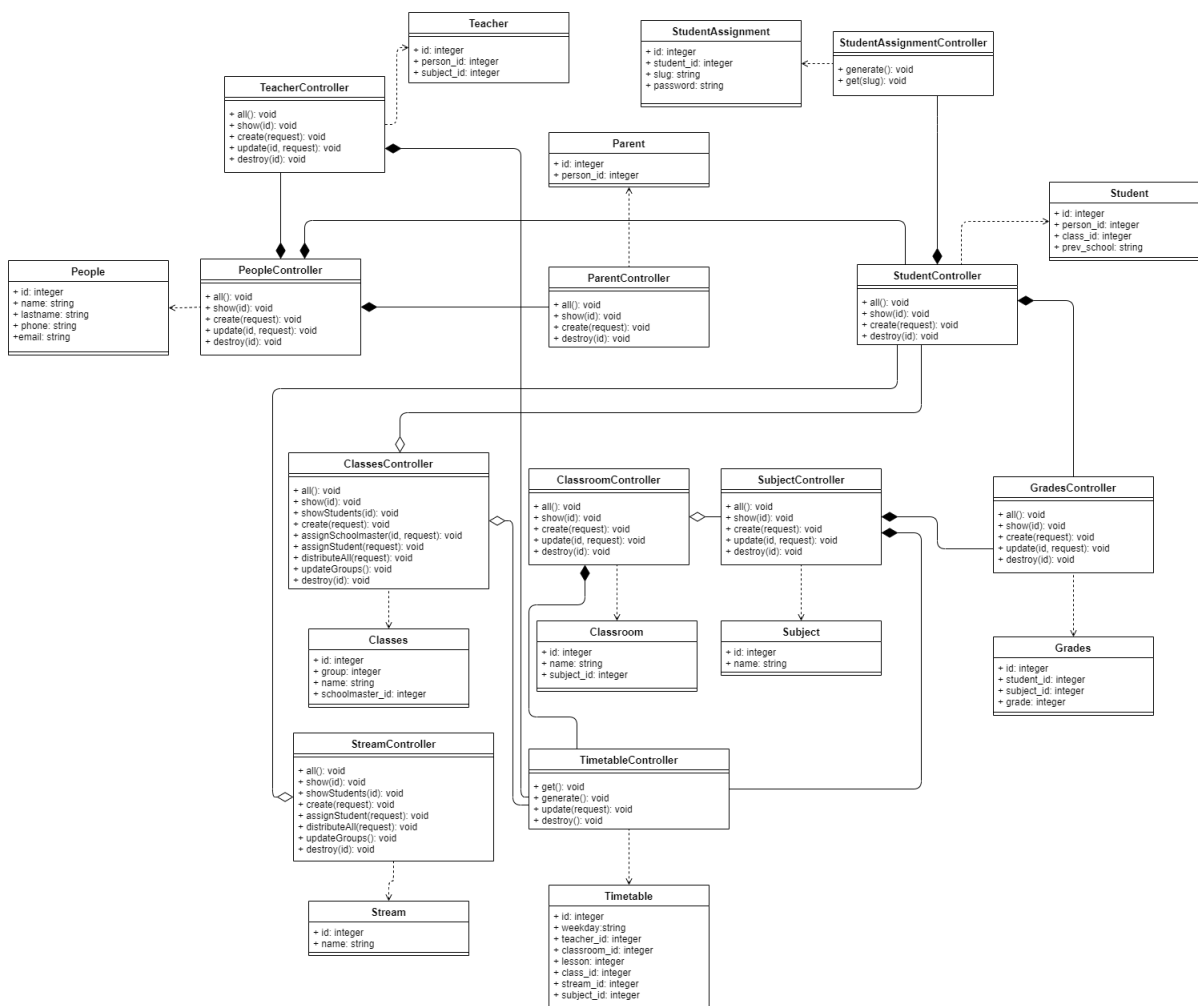
3.2.3. Nefunkciniai reikalavimai

Nr.	Reikalavimas	Reikalavimų grupė	Panaudos atvejis	Sudėtingumas	Nuoroda į papildomą informaciją
NR-01	Sistemos vartotojo sąsaja turi būti lietuvių kalba.	Vartotojo sąsajai	-	Nesudėtingas	-
NR-02	Puslapis prisitaikantis prie ekrano dydžio.	Vartotojo sąsajai	-	Vidutinis	-
NR-04	Neįvedus privalomų duomenų, formos neturėtų būti išsiunčiamos.	-	-	Nesudėtingas	-
NR-05	Duomenų bazėje svarbiausi duomenys turi būti šifruojami.	Saugumo	-	Vidutinis	-

Lentelė 11 Nefunkciniai reikalavimai

3.3. Informacinės posistemės projektas

3.3.1. Klasių diagramos



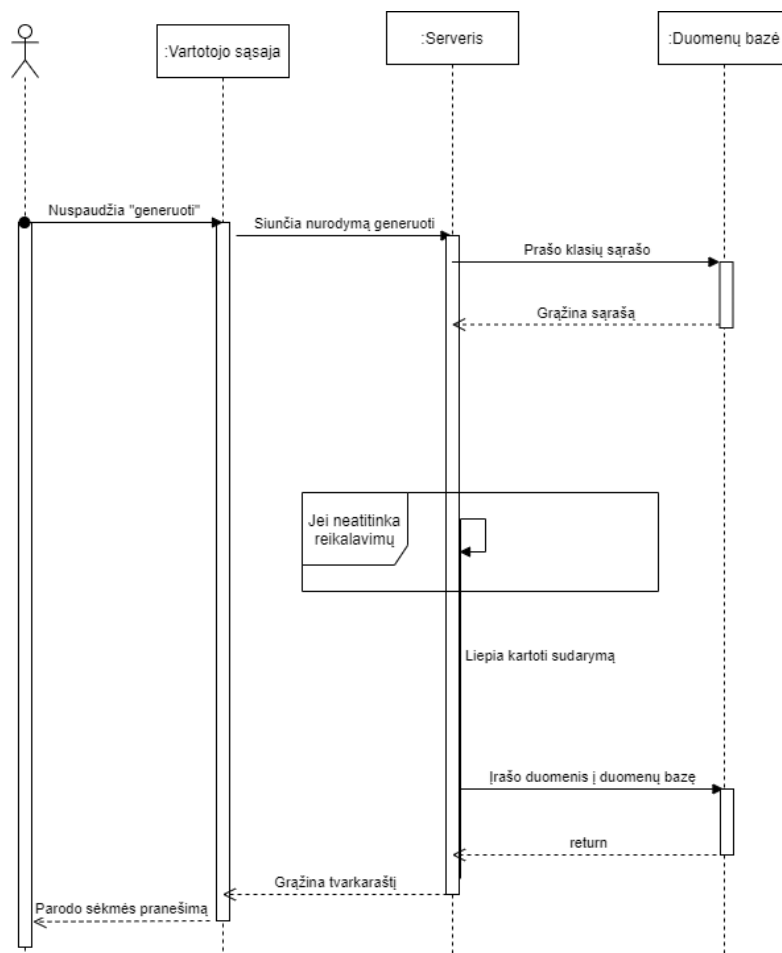
Pav. 3.10 Klasių diagrama

Nr.	Pavadinimas	Aprašymas
1.	People	Bendra klasė nurodanti, asmenų vardus, pavardes ir kontaktinius duomenis
2.	Teacher	Klasė nurodanti, kurie asmenys yra mokytojai
3.	Parent	Klasė nurodanti, kurie asmenys yra globėjai
4.	Student	Klasė nurodanti, kurie asmenys yra mokiniai
5.	Classes	Nurodo, kuriai klasei priklauso mokinys
6.	Subject	Suvesti visi mokomieji dalykai
7.	Grades	Moksleivio pažymių sąrašas reikalingas skirstant į klases bei srautus
8.	Classroom	Kabinetų sąrašas
9.	Stream	Moksleivių srautų sąrašas
10.	Timetable	Tvarkaraštis
11.	StudentChoices	Moksleivių pasirinkimai 3 klasei
12.	StudentAssignment	Moksleivių paskirstymo į klases sąrašas
13.	PeopleController	Asmenų valdymo klasė

14.	TeacherController	Mokytojų valdymo klasė
15.	ParentContoller	Tėvų valdymo klasė
16.	StudentController	Moksleivių valdymo klasė
17.	ClassesController	Klasių valdymo klasė
18.	SubjectController	Mokomųjų dalykų valdymo klasė
19.	GradesController	Pažymių valdymo klasė
20.	ClassroomController	Kabinetų valdymo klasė
21.	StreamController	Srautų valdymo klasė
22.	TimetableController	Tvarkaraščio valdymo klasė
23.	StudentAssignmentController	Moksleivių paskyrimų valdymo klasė
24.	StudentChoicesAssignment	Moksleivių 3 klasės pasirinkimų valdymo klasė

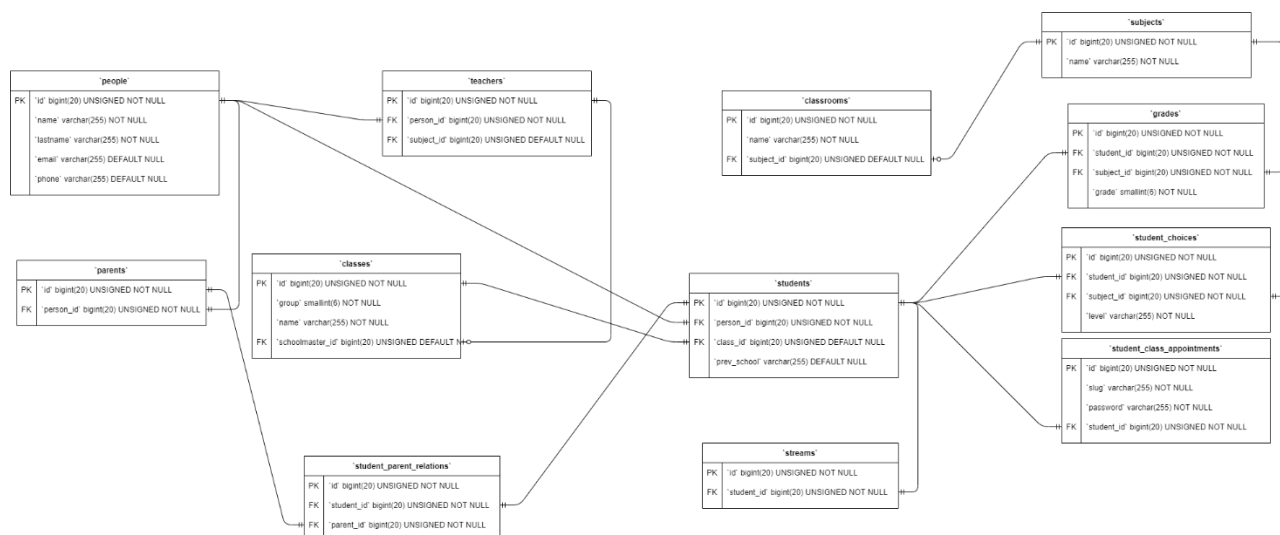
Lentelė 12 Klasių diagramos paaiškinimas (1)

3.3.2. Sekų diagramos



Pav. 3.11 Tvarkaraščio sudarymo sekos diagrama

3.3.3. Duomenų bazės projektas



Pav. 3.12 Duomenų bazės projektas

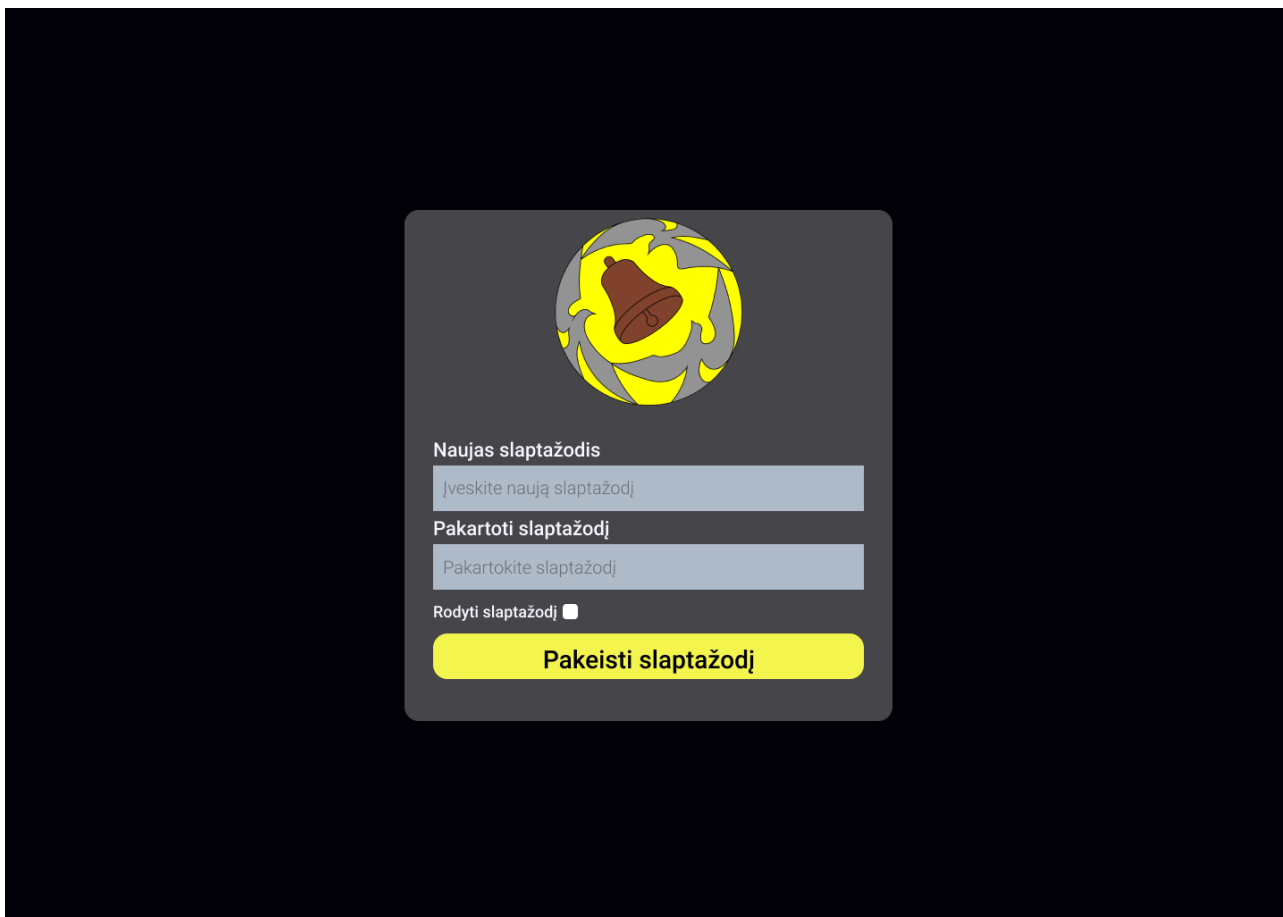
Nr.	Pavadinimas	Duomenų tipai	Aprašymas
1.	people	id:int name:string lastname:string phone:string email:string	Visi sistemoje įvestų asmenų pagrindiniai duomenys
2.	parents	id:int student_id:int	Nurodomas asmuo, kuris yra mokinio globėjas
3.	students	id:int class_id:int	Nurodomas asmuo, kuris yra mokinys, nurodomas auklėtojas bei klasė
4.	teachers	Id:int	Nurodomas asmuo, kuris yra mokytojas
5.	classes	Id:int Pavadinimas:string	Nurodoma klasė sudaryta iš mokinių
6.	classrooms	Id:int Pavadinimas:string	Kabinetai, kuriuose vyksta pamokos
7.	grades	Id:int Mokinys:int Dalykas:string Pažymys:int	Nurodomi mokių pažymiai ateinant į mokyklą
8.	timetable	Diena:string Mokytojas:int Kabinetas:int Pamoka:string Klasė:int	Pamokų tvarkaraštis

Lentelė 13 Duomenų bazės projekto paaiškinimas

moksleivius arba įkelti csv failą su moksleivių informacija. Nuspaudus mygtukus „Eksportuoti į moodle“ ir „Eksportuoti į manodienynas“ yra sugeneruojami atitinkami csv failai, taip suformatuoti, kaip reikalinga atitinkamai sistemai norint įvest naujus naudotojus. Paspaudus mygtuką „Skirstyti į klases“, visi moksleiviai, kurie neturi jiems priskirtos klasės yra automatiškai paskirstomi ir vartotojas yra nukreipiamas į „Klasės“ langą. Lange „Klasės“ yra lentelė su pagrindine informacija apie klasę bei mygtukais jai redaguoti, kuriuos nuspaudus atsidaro forma, kurioje galima paskirti naują auklėtoją ar perkelti moksleivius į kitas klases. Nuspaudus „Šalinti“ mygtuką, klasė yra panaikinama. Paspaudus mygtuką „Pridėti naują“ yra sukuriamas tuščia klasė. Nuspaudus nuorodą „Srautai“, vartotojas nukreipiamas į srautų langą, kuriame 3 ir 4 klasės moksleiviai yra suskirstomi į srautus pagal jų pasirinktus dalykus. Lange „Tvarkaraštis“ yra lentelė su redaguojamais teksto laukais, šiuose laukuose yra sudedami kokiai klasei kokiu metu vyksta pamoka. Laukuose atlikus pakeitimų ir paspaudus mygtuką „Saugoti“ pakeitimai yra išsaugomi. Paspaudus mygtuką „Generuoti“ yra sugeneruojamas naujas tvarkaraštis vietoj esamo. Langas „Moksleivio paskyrimas“ yra skirtas moksleivio tėvams/globėjams. Šiame lange yra forma, kurioje įvedus pirmas moksleivio vardo ir pavardės raides yra parodomas pranešimas, kokiai klasei jis buvo paskirtas.

Pav. 3.14 Langas - Prisijungimas

Lange yra viena forma su dviem laukais: el. paštas bei slaptažodis. Šiuose laukuose įvedus prisijungimo duomenis ir nuspaudus mygtuką „Prisijungti“, vartotojas yra autentifikuojamas ir prijungiamas prie sistemos. Paspaudus mygtuką „Pamiršote slaptažodį“, jei el. pašto laukas yra užpildytas, šiuo adresu yra išsiunčiama slaptažodžio atkūrimo nuoroda.



Pav. 3.15 Langas – Slaptažodžio atstatymas

Lange yra viena forma su dviem laukais: naujas slaptažodis ir pakartoti slaptažodį. Užpildžius šiuos laukus ir paspaudus mygtuką „Pakeisti slaptažodį“ yra pakeičiamas vartotojo slaptažodis.



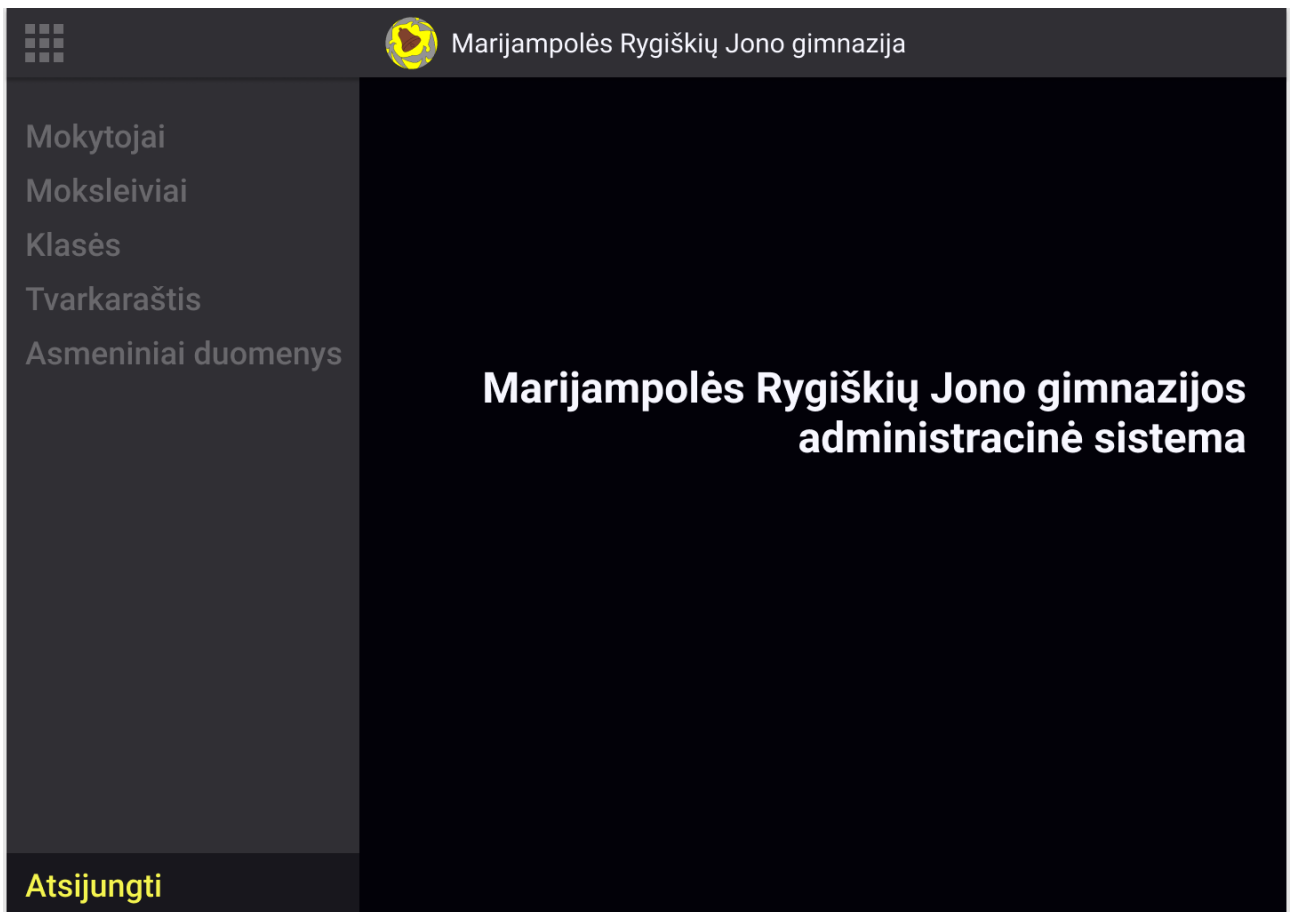
Marijampolės Rygiškių Jono gimnazijos administracinė sistema

Jei esate MRJG bendruomenės narys

PRISIJUNGTI

Pav. 3.16 Langas – Pagrindinis sistemos langas (neprisijungus)

Neprisijungus prie sistemos, pagrindiniame lange yra vienas mygtukas, nukreipiantis į prisijungimo langą.



Pav. 3.17 Langas – Pagrindinis sistemos langas (prisijungus)

Prisijungus prie sistemos, pagrindiniame lange yra 5 nuorodos į kitus langus bei mygtukas „Atsijungti“. Nuspaudus šį mygtuką, vartotojas yra atjungiamas nuo sistemos ir grąžinamas į pagrindinį puslapį neprisijungus.


Marijampolės Rygiškių Jono gimnazija

Pridėti naują

Vardas	Pavardė	Mokomasis dalykas	Kab.	Klasė	Veiksmai
Vardenis	Pavardenis	Chemija	106	-	Redaguoti Šalinti
Vardenis	Pavardenis	Anglų k.	108	1A	Redaguoti Šalinti

Pav. 3.18 Langas – Mokytojai

Šiame lange yra lentelė, kurioje nurodomi pagrindiniai mokytojų duomenys bei mygtukai juos redaguoti ar šalinti. Taip pat yra mygtukas „Pridėti naują“ kurį nuspaudus atsidaro forma, kurioje reikia įvesti pagrindinius mokytojo duomenis, tai yra vardas, pavardė ir el. paštas. Naujai įvestas mokytojas gauna laišką su nuoroda į slaptažodžio keitimo puslapį, kur susikuria slaptažodį.


Marijampolės Rygiškių Jono gimnazija


Skirstyti į klases
Pridėti naują

Vardas	Pavardė	Klasė	Telefonas	El. paštas	Globėjai	Veiksmai
Vardenis	Pavardenis	1A	+3706...	vardas@g...	Vardas...	Redaguoti Šalinti
Vardenis	Pavardenis	1B	+3706...	vardas@g...	Vardas...	Redaguoti Šalinti

Eksportuoti į:
Mano Dienynas
Moodle

Pav. 3.19 Langas – Moksleiviai

Šiame puslapyje lentelėje nurodomi pagrindiniai moksleivių duomenys bei yra mygtukai jiems redaguoti bei šalinti. Paspaudus mygtuką „Pridėti naują“ Atsidaro forma, kurioje reikia įvesti duomenis apie individualų moksleivį arba įkelti sąrašą. Mygtukais eksportuoti į Mano Dienynas bei Moodle sugeneruojami csv failai reikalingi vartotojų įvedimui į šias sistemas. Mygtuku „Skirstyti į klases“, moksleiviai yra suskirstomi ir vartotojas yra nukreipiamas į „Klasės“ langą.


Marijampolės Rygiškių Jono gimnazija

Srautai
Pridėti naują

Klasių grupė: 1-os

	1A	1B	1C	1D	1E
Auklėtojas	Vardenis Pavardenis	Vardenis Pavardenis	Vardenis Pavardenis	Vardenis Pavardenis	Vardenis Pavardenis
1.	Vardenis Pavardenis	Vardenis Pavardenis	Vardenis Pavardenis	Vardenis Pavardenis	Vardenis Pavardenis
2.	Vardenis Pavardenis	Vardenis Pavardenis	Vardenis Pavardenis	Vardenis Pavardenis	Vardenis Pavardenis
Veiksmai	Redaguoti Šalinti	Redaguoti Šalinti	Redaguoti Šalinti	Redaguoti Šalinti	Redaguoti Šalinti

Pav. 3.20 Langas – Klasės

Šiame lange pasirinkus klasių grupę yra lentelė, kurioje yra duomenys apie visas klases taip pat mygtukai jų redagavimui ir naikinimui. Paspaudus mygtuką „Pridėti naują“ yra sukurama tuščia klasė.



Pav. 3.21 Langas – Srautai

Šiame lange yra lentelė su informacija apie srautus bei mygtukai leidžiančiais pamatyti daugiau informacijos. Paspaudus mygtuką „Sudaryti“, 3-4 klasių mokiniai yra suskirstomi į srautus pagal jų pasirinkimus.



Slaptažodis

Įveskite moksleivio vardo ir pavardės 3 pirmas raides

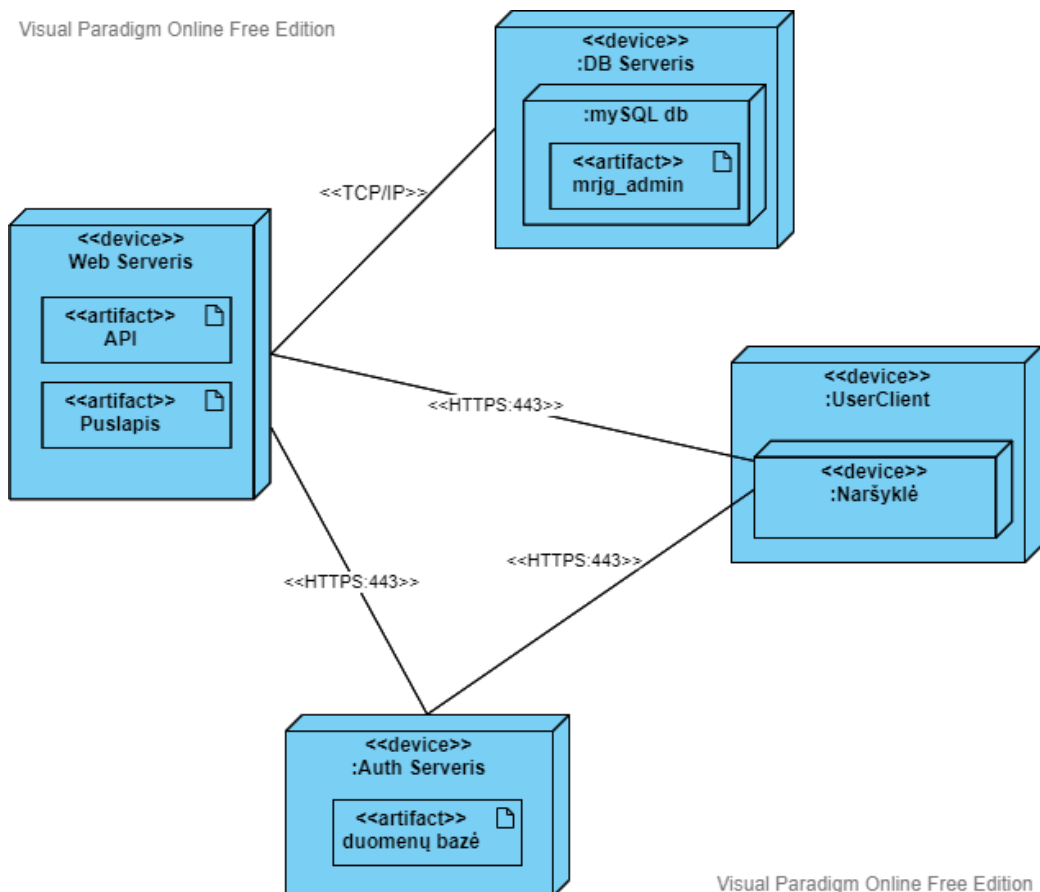
Rodyti slaptažodį ☐

RODYTI

Pav. 3.23 Langas – Moksleivio paskyrimas

Šiame lange yra forma, kurioje įvedus moksleivio pirmas vardo ir pavardės raides ir paspaudus mygtuką „Rodyti“ yra parodomas pranešimas, kokiai klasei jis buvo paskirtas.

3.4. Diegimo projektas



Pav. 3.24 Diegimo diagrama

Sistema turėtų būti patalpinta į vieną web serverį abi serverio ir kliento dalys. Serverio dalis turi komunikuoti su duomenų baze, todėl reikalingas ir duomenų bazės serveris, su kuriuo web serveris galėtų komunikuoti TCP/IP protokolu. Taip pat norint, jog vartotojai galėtų naudotis sistema reikalingas įrenginys su interneto prieiga ir interneto naršykle. Tarp web serverio ir kliento įrenginio komunikacija turėtų vykti naudojant HTTPS protokolą siekiant užtikrinti duomenų perdavimo saugumą. Kadangi autentifikacijai naudojamas išorinis tapatybės tiekėjas, reikalingas ir atskiras autentifikavimo serveris, kuris turi komunikuoti su abiem ir web serveriu, ir kliento įrenginiu. Taip pat siekiant duomenų perdavimo saugumo, įrenginiai turėtų komunikuoti naudodami HTTPS protokolą.

3.5. Aparatūrinė posistemė

Mokykla naudoja web hostingo paslauga, todėl sistema turėtų būti patalpinta į web hostingo paslaugos tiekėjo serverį.

3.6. Realizacija ir testavimas

4. Dokumentacija naudotojui

4.1. Diegimo vadovas

Norint įdiegti sistemą reikia į web serverį patalpinti serverio dalies (back-end) ir vartotojo dalies (front-end) projektus. Rekomenduojama vartotojo dalį talpinti pirmą, tuomet joje sukurti aplanką „api“ ir jame patalpinti serverio dalies projektą. Tuomet nustatyti dns nustatymus, jog aplankas „api/public“ būtų pasiekiamas domenu „api.pvz.lt“, o kliento dalis pasiekiamą domenu „pvz.lt“. Tuomet prisijungus prie „Auth0“ vartotojo konsolės nueiti į „Applications“ skiltį ir pasirinkti mrjg aplikaciją ir joje nurodyti „Allowed Callback URLs“, „Allowed Logout URLs“, „Allowed Web Origins“, „Allowed Origins (CORS)“ visos nuorodos turėtų būti „pvz.lt“. Tuomet nuėjus į „APIs“ skiltį sukurti api, kurio „Identifier“ turėtų būti „api.pvz.lt“.

Tuomet prisijungus prie duomenų bazės reikia sukurti naują duomenų bazę. Ir prisijungus prie web serverio atsidarius vartotojo dalies projektą atsidaryti .env failą, kuriame reikia nurodyti „Auth0“ „Domain“ ir „ClientID“. Taip pat reikia nurodyti „api_domain“, kuris turėtų būti „api.pvz.lt“. Atsidarius serverio dalies projektą, jame reikia atsidaryti .env failą, jame reikia nurodyti prisijungimo prie duomenų bazės duomens bei nurodyti api domeną.

Jei galima pasiekti projektą naudojantis konsolę, nuėjus į serverio dalie projekto aplanką reikia paleisti funkciją „php artisan migrate“. Taip bus sukurtos lentelės duomenų bazėje. Jei projekto pasiekti naudojant konsolę neįmanoma, reikia prie serverio dalies projekto pridėti „db.sql“ failą importuoti į turimą duomenų bazę.

4.2. Vartotojo vadovas

Išvados

1. Lorem ipsum dolor sit amet, eam ex decore persequeris, sit at illud lobortis atomorum. Sed dolorem quaerendum ne, prompta instructor ne pri. Et mel partiendo suscipiantur, docendi abhorreant ea sit. Recteque imperdiet eum te.
2. Eu eum decore inimicus consetetur, cu usu habeo corpora intellegam. Ut antiopam efficiendi deterruisset sit. Mel sint eirmod id, qui quot virtute id, dolor nemore forensibus usu id. Fugit dolore voluptatum cu vim. An vix veniam graecis insolens, sit posse iusto id. Ut vim ceteros percipit, id quo ubique recusabo, eum sint lucilius ea. In sumo inani numquam has.

Literatūros sąrašas

1. Ali, J. (2016 m. gruodis 28 d.). *Using Guzzle and PHPUnit for REST API Testing*. Nuskaityta iš Cloudflare: <https://blog.cloudflare.com/using-guzzle-and-phpunit-for-rest-api-testing/>
2. Almeida, R. (2017 m. vasaris 20 d.). *How to use mongoDB with your Lumen API*. Nuskaityta iš Medium: <https://medium.com/hackernoon/how-to-use-mongodb-with-your-lumen-api-e13f36fa0aa6>
3. Atha, H. (2021 m. sausis 25 d.). *Pros and Cons of Flutter – The Complete Review Updated For 2021*. Nuskaityta iš Moveo: <https://www.moveoapps.com/blog/pros-and-cons-of-flutter/>
4. Bachina, B. (2020 m. liepa 1 d.). *Why State Management is Important For Angular Apps*. Nuskaityta iš Medium: <https://medium.com/bb-tutorials-and-thoughts/why-state-management-is-important-for-angular-apps-e2e7ba8872f1>
5. Brad. (2017 m. liepa 21 d.). *Build A Better API Client with Guzzle Middleware*. Nuskaityta iš Medium: https://medium.com/@brad_brothers/build-a-better-api-client-with-guzzle-middleware-2ace56868dc7
6. Buyse, T. (2020 m. sausis 17 d.). *React State Management in 2020 Do we still need state management frameworks like Redux and MobX?* Nuskaityta iš Better Programming: <https://betterprogramming.pub/react-state-management-in-2020-719d10c816bf>
7. Caputa, M. (2019 m. sausis 27 d.). *8 Reasons Why Vue.js is Worth Considering for Your Next Project*. Nuskaityta iš Netguru: <https://www.netguru.com/blog/why-vue-js>
8. Copes, F. (2018 m. spalio 11 d.). *JWT authentication: When and how to use it*. Nuskaityta iš LogRocket: <https://blog.logrocket.com/jwt-authentication-best-practices/>
9. Daityari, S. (2021 m. kovas 15 d.). *Angular vs React vs Vue: Which Framework to Choose in 2021*. Nuskaityta iš CodeinWP: <https://www.codeinwp.com/blog/angular-vs-vue-vs-react/>
10. Emekoma, H. (2021 m. kovas 2 d.). *Comparing the best new JavaScript frameworks to React*. Nuskaityta iš LogRocket: <https://blog.logrocket.com/comparing-the-best-new-javascript-frameworks-to-react/>
11. Evgeniy Zyubin, Tobias Nyholm. (2021 m. vasaris 20 d.). *Benchmark of PSR-7 and PSR-17 implementation*. Nuskaityta iš GitHub: <https://github.com/devanych/psr-http-benchmark>
12. Ferrill, T. (2020 m. rugsėjis 30 d.). *Microsoft Azure Active Directory Review*. Nuskaityta iš PCmag: <https://www.pcmag.com/reviews/microsoft-azure-active-directory>
13. Hasnany, M. (2021 m. kovas 3 d.). *Flutter web support hits the stable milestone*. Nuskaityta iš Medium: <https://medium.com/flutter/flutter-web-support-hits-the-stable-milestone-d6b84e83b425>
14. Heller, M. (2018 m. rugsėjis 4 d.). *Review: MongoDB takes on the world*. Nuskaityta iš InfoWorld: <https://www.infoworld.com/article/3300619/review-mongodb-takes-on-the-world.html>
15. Iglío, P. (2020 m. spalio 11 d.). *Benchmarking Serverless Laravel vs Lumen*. Nuskaityta iš Medium: <https://igliop.medium.com/benchmarking-serverless-laravel-vs-lumen-with-bref-c3bdca37e5b8>
16. Jakub Rybiński, Norbert Sajdok. (2019 m. lapkritis 5 d.). *Angular state management with Redux Pattern*. Nuskaityta iš The Software Hause: <https://tsh.io/blog/angular-state-management-with-redux-pattern/>

17. JetRuby Agency. (2017 m. rugpjūtis 17 d.). *Vue.JS Review: Can It Beat React.JS?* Nuskaityta iš JetRuby: <https://expertise.jetruby.com/vue-js-review-can-it-beat-react-js-d49999855a5>
18. Kuprenko, V. (2020 m. balandis 22 d.). *Google Flutter Review – Why Mobile App Developers Love Flutter*. Nuskaityta iš freeCodeCamp: <https://www.freecodecamp.org/news/why-mobile-apps-makers-are-in-love-with-flutter/>
19. Kurtto, J. (2017 m. liepa 3 d.). *The Difference Between SAML 2.0 and OAuth 2.0*. Nuskaityta iš ubisecure: <https://www.ubisecure.com/uncategorized/difference-between-saml-and-oauth/>
20. Lloyd, H. (2020 m. birželis 30 d.). *How SAML Authentication Works*. Nuskaityta iš auth0: <https://auth0.com/blog/how-saml-authentication-works/>
21. Mammino, L. (2015 m. balandis 18 d.). *Developing a web application with Lumen and MySQL*. Nuskaityta iš Loige: <https://loige.co/developing-a-web-application-with-lumen-and-mysql/>
22. O'Shaughnessy, K. (2017 m. gegužė 21 d.). *Angular: Getting Started*. Nuskaityta iš Medium: <https://medium.com/@ZombieCodeKill/angular-getting-started-b6316d38c04c>
23. Pedamkar, P. (be datos). *Laravel vs Lumen*. Nuskaityta iš EDUCBA: <https://www.educba.com/laravel-vs-lumen/>
24. Penalara.com. (be datos). *GHC - What is it?* Nuskaityta iš penalara.com: <https://www.penalara.com/en/ES/whatisghc/>
25. Perrers, J. (2020 m. kovas 29 d.). *What is SAML and How Does it Work?* Nuskaityta iš Varonis: <https://www.varonis.com/blog/what-is-saml/>
26. Preecha Noiumkar, Tawatchai Chomsiri. (2014 m. liepa). *A Comparison the Level of Security on Top 5 Open Source NoSQL Databases*. Nuskaityta iš ResearchGate: https://www.researchgate.net/publication/301633978_A_Comparison_the_Level_of_Security_on_Top_5_Open_Source_NoSQL_Databases
27. Putra, R. E. (2020 m. spalio 21 d.). *Simple CRUD REST API With Lumen + MySQL + Eloquent ORM*. Nuskaityta iš dev.to: <https://dev.to/rizqyep/simple-crud-rest-api-with-lumen-mysql-eloquent-orm-dkf>
28. Ratnottar, S. (2020 m. rugsėjis 8 d.). *Angular vs. Vue – Which is Best for Programming in 2020?* Nuskaityta iš freeCodeCamp: <https://www.freecodecamp.org/news/angular-vs-vue-which-is-best-for-programming-in-2020/>
29. Rytov, O. (be datos). *Why I Decided To Embrace Laravel*. Nuskaityta iš Toptal: <https://www.toptal.com/laravel/why-i-decided-to-embrace-laravel>
30. Savoldi, P. (2018 m. vasaris 18 d.). *Why I choose Slim Framework for my PHP web development*. Nuskaityta iš Medium: <https://medium.com/@paooolino/why-i-choose-slim-framework-for-my-php-web-development-3b087e6d09fc>
31. Sethi, K. (2018 m. liepa 25 d.). *MongoDB vs. RDBMS*. Nuskaityta iš Database Zone: <https://dzone.com/articles/mongodb-vs-rdbms>
32. Smallcombe, M. (2020 m. gegužė 26 d.). *PostgreSQL vs MySQL: The Critical Differences*. Nuskaityta iš Xplenty: <https://www.xplenty.com/blog/postgresql-vs-mysql-which-one-is-better-for-your-use-case/>
33. Sweedu.com. (2020 m. spalio 27 d.). *Top benefits of using timetable management at school*. Nuskaityta iš sweedu.com: <https://sweedu.com/blog/timetable-management-benefits/>

34. *The Good and the Bad of Angular Development*. (2020 m. kovo 25 d.). Nuskaityta iš AltexSoft: <https://www.altexsoft.com/blog/engineering/the-good-and-the-bad-of-angular-development/>
35. Vakhnenko, H. (be datos). *Reasons to Use Auth0 for Authentication in Your Application*. Nuskaityta iš Agile: <https://agilie.com/en/blog/reasons-to-use-auth0-for-authentication-in-your-application>
36. *Vue.js: Benefits and Drawbacks*. (2021 m. vasaris 8 d.). Nuskaityta iš The European Business Review: <https://www.europeanbusinessreview.com/vue-js-benefits-and-drawbacks/>
37. Watson, L. (2020 m. vasaris 14 d.). *Best School Scheduling Software*. Nuskaityta iš Medium: <https://medium.com/best-software/school-scheduling-software-411dd0a8f960>
38. Yarosh, K. (2018 m. lapkritis 12 d.). *Understanding Amazon Cognito user pool OAuth 2.0 grants*. Nuskaityta iš aws: <https://aws.amazon.com/blogs/mobile/understanding-amazon-cognito-user-pool-oauth-2-0-grants/>
39. Yen, R. (2019 m. gruodis 19 d.). *How to use PostgreSQL with Laravel*. Nuskaityta iš EnterpriseDB: <https://www.enterprisedb.com/postgres-tutorials/how-use-postgresql-laravel>

Informacijos šaltinių sąrašas

1. Informacijos šaltinis: [Best Alternatives to Peñalara GHC App Free for Windows \(2021\) \(alternativetoapp.com\)](#)
2. Informacijos šaltinis: [GHC Reviews and Pricing - 2021 \(capterra.com\)](#)
3. Informacijos šaltinis: [School scheduling software for Mac, PC, tablet and phone - Prime Timetable](#)
4. Informacijos šaltinis: [User Guide - Prime Timetable](#)
5. Informacijos šaltinis: [Alma Reviews and Pricing - 2021 \(capterra.com\)](#)
6. Informacijos šaltinis: [School Features & Solutions | Alma SIS \(getalma.com\)](#)
7. Informacijos šaltinis: [Getting Started - Ex Libris Knowledge Center \(exlibrisgroup.com\)](#)
8. Informacijos šaltinis: [Why experienced developers consider Laravel as a poorly designed framework? : PHP \(reddit.com\)](#)
9. Informacijos šaltinis: [Installation - Laravel - The PHP Framework For Web Artisans](#)
10. Informacijos šaltinis: [Installation - Lumen - PHP Micro-Framework By Laravel](#)
11. Informacijos šaltinis: [Guzzle, PHP HTTP client — Guzzle Documentation](#)
12. Informacijos šaltinis: [Slim 4 Documentation - Slim Framework](#)
13. Informacijos šaltinis: [Laravel vs. Slim Framework | G2](#)
14. Informacijos šaltinis: [Getting Started – React \(reactjs.org\)](#)
15. Informacijos šaltinis: [Tutorial: Intro to React – React \(reactjs.org\)](#)
16. Informacijos šaltinis: [Introduction | Vue.js](#)
17. Informacijos šaltinis: [What is Vuex? | Vuex \(vuejs.org\)](#)
18. Informacijos šaltinis: [Angular - Introduction to the Angular Docs](#)
19. Informacijos šaltinis: [Flutter documentation - Flutter](#)
20. Informacijos šaltinis: [MySQL :: MySQL Documentation](#)
21. Informacijos šaltinis: [MySQL 8.0 Reference Manual \(oracle.com\)](#)
22. Informacijos šaltinis: <https://www.postgresql.org/files/documentation/pdf/13/postgresql-13-A4.pdf>
23. Informacijos šaltinis: [PostgreSQL Reviews and Pricing | IT Central Station](#)
24. Informacijos šaltinis: [Getting Started — MongoDB Manual](#)
25. Informacijos šaltinis: [Document Database | MongoDB](#)
26. Informacijos šaltinis: [JSON Web Token Introduction - jwt.io](#)
27. Informacijos šaltinis: [Using JWT to authenticate users | API Gateway Documentation \(google.com\)](#)
28. Informacijos šaltinis: [Using JWT to authenticate users | API Gateway Documentation \(auth0.com\)](#)
29. Informacijos šaltinis: [End User Authentication with OAuth 2.0 — OAuth](#)
30. Informacijos šaltinis: [Amazon Cognito Documentation](#)
31. Informacijos šaltinis: [OpenID Connect | Google Identity | Google Developers](#)
32. Informacijos šaltinis: [What is Azure Active Directory? - Azure Active Directory | Microsoft Docs](#)
33. Informacijos šaltinis: [Why Auth0 - Auth0](#)

34. Informacijos šaltinis: Which OAuth 2.0 Flow Should I Use? (auth0.com)

Priedai (pagal poreikį)

1 priedas. Priedo pavadinimas

Lorem ipsum dolor sit amet, eam ex decore persequeris, sit at illud lobortis atomorum. Sed dolorem quaerendum ne, prompta instructor ne pri. Et mel partiendo suscipiantur, docendi abhorreant ea sit. Recteque imperdiet eum te.

Eu eum decore inimicus consetetur, cu usu habeo corpora intellegam. Ut antiopam efficiendi deterruisset sit. Mel sint eirmod id, qui quot virtute id, dolor nemore forensibus usu id. Fugit dolore voluptatum cu vim. An vix veniam graecis insolens, sit posse iusto id. Ut vim ceteros percipit, id quo ubique recusabo, eum sint lucilius ea. In sumo inani numquam has.