



VILNIAUS UNIVERSITETAS
MATEMATIKOS IR INFORMATIKOS FAKULTETAS
INFORMATIKOS INSTITUTAS
KOMPIUTERINIO IR DUOMENŲ MODELIAVIMO KATEDRA

Bakalauro baigiamasis darbas

**Efektyvi apklausų ir balsavimo sistema universiteto fakulteto
bendruomenei**

Atliko:

Martynas Giržadas

parašas

Vadovas:

Lekt. Kazimieras Mickus

Vilnius

2018

Turinys

Anotacija.....	4
Summary.....	5
Įvadas.....	6
1. Panašių internetinių svetainių analizė.....	8
1.1. Panašaus pobūdžio internetinės svetainės.....	8
1.1.1. www.manoapklausa.lt svetainės analizė	8
1.1.2. www.manoapklausa.lt svetainės analizė mobiliajame telefone.....	9
1.1.3. www.apklausa.lt svetainės analizė	9
1.1.4. www.apklausa.lt svetainės analizė mobiliajame telefone	10
1.1.5. www.mentimeter.com svetainės analizė	10
1.1.6. www.mentimeter.com svetainės analizė mobiliajame telefone.....	11
1.1.7. forms.google.com svetainės analizė	11
1.1.8. forms.google.com svetainės analizė mobiliajame telefone	12
1.2. Analizės apibendrinimas.....	12
1.3. Kuriamai sistemai iškelti reikalavimai	14
2. Projektavimas	16
2.1. Svetainės panaudos atvejų diagrama	16
2.2. Svetainės veiksmų diagrama.....	17
2.3. Agile - Lean	18
2.4. Bendra sistemos architektūra	20
2.5. Technologijos.....	21
2.5.1 React	21
2.5.2 Redux.....	21
2.5.3 MongoDB NoSQL duomenų bazė	23

3. Programavimas	25
3.1. Programavimo aplinkos paruošimas	25
3.2. React Router	25
3.3. Pagrindinių aplikacijos komponentų kūrimas.....	26
3.4. Redux konteineriai	27
3.5. NoSQL duomenų bazės struktūra	27
3.6. Autentifikacija su Auth0.....	28
4. Sistemos apžvalga	29
Išvados ir rekomendacijos	30
Literatūros sąrašas	31
Priedai.....	32

Anotacija

Baigiamojo bakalaurinio darbo tikslas – sukurti efektyvią balsavimo ir apklausų sistemą fakulteto bendruomenei panaudojant neeilines inovatyvias technologijas. Sistema turi būti lengva naudotis, ji turi būti orientuota į kuo patogesnę momentinės apklausos sukūrimą tam tikrai žmonių grupei. Darbo tikslui pasiekti buvo iškelti šie uždaviniai:

- Išanalizuoti analogiškas sistemas;
- Apsibrėžti reikalavimus, pasirinkti tinkamas technologijas;
- Sukurti lengvai naudojamą ir patogią sistemą;
- Dirbti iteracijomis inkrementiškai auginant produktą.

Darbo metu pasiekti rezultatai – sukurta balsavimo ir apklausų sistema, naudojant React ir Redux technologijas. Sistemos serveris sukurtas panaudojant Express karkasą, kuriame veikia sistemai sukurta API sąsaja darbui su MongoDB duomenų baze. Sistemos vartotojai autentifikuojami panaudojant modernią tapatybės platformą Auth0.

Summary

Effective Poll and Voting System for Community of the Faculty of the University

The purpose of this bachelor's thesis is to create an effective poll and voting system for community of the faculty of the university while using innovative technologies. The system in question should be easy-to-use and oriented to the main objective – creating an instant poll for a particular group of respondents. Some tasks were defined to reach the goal of the thesis:

- To analyze similar systems;
- Define requirements, choose appropriate technologies;
- To create a system which is easy-to-use and convenient;
- Work with iterations while incrementally building the system.

Results achieved – poll and voting system was created using React and Redux. The server ran on Express.js framework, which also contains an API specifically created for this system to work with MongoDB database. Users of this system are authenticated using a modern identity platform Auth0.

Ivadas

Šiuolaikiniame pasaulyje yra begalė įvairių sistemų skirtingiems vartotojų poreikiams patenkinti, iš kurių beveik kiekvienai sistemai yra sukurtos bent kelios alternatyvos. Alternatyviose sistemose stengiamasi akcentuoti pirmosios sukurtos sistemos trūkumus ir pasitelkti juos kuriant naują - geresnį produktą. Ne išimtis yra ir balsavimo bei apklausų sistemos, kurių yra įvairiausių tipų, akcentuojančių įvairias sistemos veikimo dalis. Kai kurios sistemos orientuotos į patogų rezultatų vaizdavimą, kai kurios į patogų duomenų išgavimą tolimesniam duomenų apdorojimui. Trečios sukurtos kaip visavertės platformos, kurios yra plačiai konfigūruojamos ir gali atitikti platų spektrą vartotojų poreikių. Esant milžiniškam balsavimo ir apklausų sistemų pasirinkimui, atsirado poreikis sukurti į fakulteto bendruomenę orientuotą apklausų ir balsavimo sistemą, kuria būtų paprasta naudotis ir kurios pagalba galima būtų greitai ir patogiai apklausti, pavyzdžiui, studentų grupę.

Šio baigiamojo bakalaurnio darbo tikslas - yra sukurti efektyvią apklausų ir balsavimo sistemą fakulteto bendruomenei. Šiam tikslui pasiekti buvo iškelti šie uždaviniai:

1. Išanalizuoti keturias skirtingas balsavimo ir apklausų sistemas;
2. Palyginti analizuojamų sistemų privalumus;
3. Įvardinti analizuojamų sistemų skirtumus;
4. Formuoti reikalavimus naujai kuriamai sistemai;
5. Sukurti mobiliems įrenginiams patogią vartotojo sąsają;
6. Įgyvendinti patogų ir greitą apklausos kūrimo bei dalyvavimo joje veikimo principą;
7. Sukurti sistemą lietuvių kalba.

Baigiamojo bakalaurnio darbo metu buvo sukurta apklausų ir balsavimo sistema dedikuota fakulteto bendruomenei. Darbo gynimui paruoštoje sistemos versijoje veikia apklausos kūrimo, keitimo, balsavimo, trynimo funkcijos, vienas svarbiausių funkcionalumų – slaptažodžių sistema, sukurta siekiant sukurti patogią prieigą mobiliems įrenginiams. Autentifikuotam vartotojui sukūrus privačią apklausą su 6 skaitmenų slaptažodžiu, respondentai gali patogiai į jas atsakyti neprisijungę prie sistemos, o tiesiog įvedę teisingą apklausos slaptažodį. Vienas iš pagrindinių neįgyvendintų funkcionalumų – Auth0 tapatybės platformos dalinis implementavimas. Darbo metu nespėta sukonfigūruoti platformos vartotojų duomenų tikrinimui su fakulteto duomenų baze naudojant SAML kalbą. Šiuo metu sistemoje įgyvendinta paprasta autentifikacija – galimybę prisijungti naudojant Google paskyrą arba kuriant vartotoją Auth0 platformoje.

Sistema yra patalpinta Heroku (<http://herokuapp.com>) debesų platformoje, o duomenų bazė patalpinta mLab (<http://www.mlab.com>) specialioje MongoDB duomenų bazės kaip paslaugos (DbaaS) platformoje. Sistema yra pasiekama adresu <http://mif-vote-app.herokuapp.com>.

1. Panašių internetinių svetainių analizė

1.1. Panašaus pobūdžio internetinės svetainės

Siekiant sukurti optimalų universiteto fakulteto bendruomenės apklausų sistemos sprendimą, pirmiausia buvo analizuojamos kitos apklausų sistemos, jų privalumai ir trūkumai. Apklausų sistemos yra labai skirtingos ir panaudojamos skirtinguose scenarijuose prisitaikant prie vartotojo poreikių. Kai kurios sistemos yra orientuotos į vartotojus, besinaudojančius mobiliaisiais įrenginiais, kai kurios akcentavo rezultatų pateikimo ir vizualizavimo procesą baigus vykdyti apklausą, dar kitos yra visavertės apklausų kūrimo platformos, stengiančios apimti visus vartotojų poreikius. Analizės metu pasirinktos keturios sistemos, kurias analizuojant siekta įgyti žinių kaip kurti panašų projektą, išmokti taikyti gerąsias praktikas ir nekartoti esamų klaidų vartotojo sąsajoje, kuriant lietuvišką, greitą, patogią ir efektyvią balsavimo ir apklausų sistemą universiteto fakulteto bendruomenei. Pagrindiniai kriterijai, kuriais buvo lyginamos apklausų svetainės buvo išskirti remiantis kokybiniu tyrimu metodu, kuomet tyrėjas pats gali nuspręsti ar išskirti jam svarbius pagrindinius kriterijus, kuriais remiantis įmanoma palyginti skirtingas sistemas.

1.1.1. www.manoapklausa.lt svetainės analizė

Užėjus į analizuojamą svetainę iš karto pastebimas gražus ir šiuolaikiškas svetainės dizainas, kuris suformuoja gerą įspūdį vartotojui [8]. Svetainė siūlo registruotis bei prisijungti, pasirinkti tarp dviejų apklausų tipų – verslo ir studentų apklausų. Svetainėje nurodoma, kad naudotis ja yra greita ir paprasta, į akį krenta dizaino vientisumas, jokių stiliaus išsikraipymų, tiek kompiuteryje tiek mažesniuose ekranuose stilius vientisas ir patrauklus. Ši sistema yra nemokama, su galimybe nusipirkti papildomų paslaugų paketą metų laikotarpiui.

Labai funkcionalus ir patrauklus apklausos rezultatų apipavidalinimas bei pateikimas. Svetainėje sukurta atskira skiltis rezultatams pristatyti, nes rezultatų apžvalga yra bene stipriausia iš visų bandytų apklausų sistemų. Galimybė pasirinkti bet kokio tipo grafiką kiekvienam klausimui – skritulinę diagramą, kreivę, rezultatų išdėstymą, spalvas ir t.t. Patogus duomenų eksportavimas pageidaujamu formatu. Yra atskira skiltis kiekvieno respondento atsakymų analizei individualiai.

1.1.2. www.manoapklausa.lt svetainės analizė mobiliajame telefone

Analizuojama apklausų sistemos [manoapklausa.lt](http://www.manoapklausa.lt) rezultatų skiltis yra stiprioji šios sistemos pusė, tačiau išbandžius apklausų pavyzdžius telefone, akivaizdu, jog sistemos silpnoji pusė yra mobiliaisiais įrenginiais apklausas norintys atsakinėti vartotojai. Nors stilius yra vientisas ir toks pats gražus kaip ir kompiuteriu prieinamoje internetinėje svetainėje, atsakinėjant į apklausos klausimus mobiliajame įrenginyje tekstas yra gerokai per mažas, sunku pasirinkti tinkamą atsakymo variantą, yra daug laisvos vietos, kurią galima būtų išnaudoti geresniam vartotojo patyrimui. Kita vertus, kitos apklausų sistemos skiltys yra gerai atvaizduojamos ir mobiliajame – anksčiau minėta rezultatų skiltis yra taip pat patogi, kaip ir kompiuterinė jos versija.

1.1.3. www.apklausa.lt svetainės analizė

Sekanti analizuojama sistema buvo [apklausa.lt](http://www.apklausa.lt), kurios dizainas yra žymiai paprastesnis ir senoviškesnis, lyginant su prieš tai analizuota sistema [9]. Bene visa svetainė yra juodai balta, su keliomis spalvotomis ikonomis, tačiau šis svetainės paprastumas dizaino prasme sukuria dalykiškumo, estetiškumo ir lengvumo įspūdį. Viskas yra konkretu ir profesionalu – vos užėjus į svetainę matoma trumpa ir aiški naudojimosi instrukcija, trumpa statistika apie svetainėje sukurtas apklausas ir jų respondentus, rekomenduojamos apklausos bei naujausios viešos apklausos. Ši sistema yra nemokama, tačiau jos viduje yra taikoma kreditų sistema apklausoms reklamuoti.

Apklausų kūrimas šioje sistemoje yra ypatingai greitas ir patogus. Turint paruoštus klausimus ir atsakymus pakanka visą tekstą įklijuoti į „Greitas apklausos kūrimas“ laukelį, priskirti specialiuosius simbolius ir tuomet sugeneruojama apklausa.

Puiki funkcija yra galimybė patalpinti anketą kito puslapio turinyje įdiegiant į jį „iframe“ elementą su nuoroda į apklausą.

Sistemoje pateikiami rezultatai taip pat yra labai paprasti, apklausos puslapyje spustelėjus ant rezultatų ikonos pateikiama kiekvieno klausimo skritulinė diagrama, pateikiami procentai, patogiu ir greitu panaudoti diagramą iškerpant ją ir įklijuojant savo moksliniame darbe. Yra galimybė rezultatus eksportuoti vartotojui priimtinu formatu.

1.1.4. www.apklausa.lt svetainės analizė mobiliajame telefone

Mobilioji šios svetainės versija taip pat remiasi to paties paprasto ir dalykiško dizaino principais. Vos užėjus į svetainę su mobiliuoju įrenginiu, ekrane dideliu šriftu pasirodo naudojimosi instrukcijos ir visa kita informacija, kuri pasirodo ir kompiuterinėje versijoje. Ši apklausų sistema turi bene geriausiai mobiliesiems įrenginiams pritaikytą dizainą, nes šriftas yra didžiulis, todėl didėja ir paspaudimo ant atsakymo plotas, sumažėja vartotojų daromų klaidų ir taip gerinama bendra vartotojų patirtis svetainėje.

1.1.5. www.mentimeter.com svetainės analizė

Šią svetainę analizuoti buvo nuspręsta, kai su darbo vadovu sumąstėme kiek kitokį apklausų sistemos veikimo principą. Mūsų aptartoje sistemoje apklausos būtų vykdomos įvairių prezentacijų metu, gyvai dalinantis tam tikru slaptažodžiu arba nuoroda ir taip pasiekiant auditorijoje esančius respondentus. Būtent [mentimeter.com](http://www.mentimeter.com) siūlo panašiausią įrankį, kuris ir buvo analizuojamas siekiant panaudoti gerąsias praktikas [10]. Pati svetainė yra moderni, stilinga, suderintos spalvos ir šriftai, maloni akiai vartotojo sąsaja. Sistemoje yra privalomas prisijungimas, tačiau vėliau pasitinka puiki platformos sąsaja, kurioje galima kurti naują prezentaciją. Siūloma daugybė įvairių šablonų ir panaudojimo atvejų, pavyzdžiui galimybė visiems respondentams palikti komentarus tiesiai skaidrėje, įvairios skalės, kelių atsakymų klausimai, matricos ir greitos formos. Tai puikus ir universalus įrankis orientuotas į apklausų kūrimą prezentacijose ir kitose auditorijose. Šis įrankis yra mokamas ir tik ribotas kiekis apklausų gali būti sukurtas testuojant šią sistemą, vėliau yra taikomi šios paslaugos mokesčiai.

Ši platforma nėra skirta įprastoms, daug klausimų turinčioms, apklausoms atlikti ir nėra orientuota į duomenų eksportą tolimesniam duomenų apdorojimui. Surinktų duomenų eksportavimo norimu formatu paslauga šioje sistemoje yra mokama, tačiau dažniausiai šioje sistemoje atliktų apklausų metu surinktų duomenų niekur eksportuoti nereikia, duomenys svarbūs tik realiu laiku, prezentacijos ar paskaitos metu, visi duomenys atvaizduojami pagrindiniame apklausos ekrane, po kiekvieno balso vis atnaujinant vizualiai pateiktus duomenis.

1.1.6. www.mentimeter.com svetainės analizė mobiliajame telefone

Visa apklausų sistema yra suprojektuota tokiu principu – prezentacijos vedėjas sukuria prezentaciją mentimeter.com platformoje savo kompiuteryje, o respondentai atsako į klausimus www.menti.com svetainėje, šį adresą pasiekę mobiliuosiuose telefonuose esančiomis naršyklėmis. Atvėrus atsakymams skirtą www.menti.com svetainę, joje pasirodo laukelis apklausos kodui įvesti. Įvedus teisingą kodą, atverčiama prezentacijos metu sukurta apklausa ir apklausos tipą atitinkantys laukeliai atsakymams pateikti. Apklausos atsakymus pasirinkti yra labai lengva, erdvė paspaudimui yra didelė, todėl vartotojas gali patogiai ir lengvai pasirinkti norimą atsakymo variantą. Vartotojui atsakius į klausimą, pagrindinis apklausos langas atsinaujina realiu laiku ir vis papildo grafikus naujais duomenimis. Visa sistema puikiai apgalvota, orientuota į savo tikslą, išlaiko vientisą stilių ir suteikia vartotojui gerą patirtį naudojantis ja.

1.1.7. forms.google.com svetainės analizė

Vienareikšmiškai pats universaliausias, labiausiai techniškai ir vizualiai konfigūruojamas ir bet kokius kliento lūkesčius galintis atitikti įrankis [11]. Apklausų sistema yra puikiai suderinta, savyje talpina didžiulį kiekį kintamų parametrų ir vistiek išlieka maloni akiai ir nevarginanti vartotojo. Google Forms sitemoje viskas aišku, vartotojas prieš pradėdamas kurti apklausą gali pasirinkti iš anksto sukurtą šabloną, kurį vėliau gali keisti. Vartotojui suteikiama galimybė pasirinkti vieną iš begalės klausimų tipų („trumpas atsakymas“, „žymimieji langeliai“, „kelių pasirinkimų“ ir t.t.). Kuriant formą juntamas gigantės „Google“ užnugaris, nes vartotojams suteikiama galimybė įkelti failus, kurie įkeliami į kitą Google platformą – Google Drive, taip panaudojant ir kito jų sukurto produkto resursus ir galimybes. Ankščiau analizuotų apklausų sistemų kontekste Google Forms turi unikalią galimybę įdiegti sukurtą apklausą tiesiai į kitos svetainės turinį, įdiegiant „iframe“ komponentą į svetainės HTML kodą ir nurodant unikalią apklausos nuorodą kaip šaltinį.

Rezultatus iš formos taip pat patogiu apdoro ti naudojant dar vieną Google produktą – Google Skaičiuokles (angl. „Google Spreadsheets“). Paliekama ir standartinė galimybė atsisiųsti duomenis .CSV formatu. Abi šios funkcijos leidžia patogiai naudotis surinktais rezultatais, juos toliau apdorojant darant įvairius skaičiavimus moksliniams darbams ir kitiems panaudojimo atvejams. Pritrūko funkcionalumo leidžiančio pasirinkti skirtingus apklausos rezultatų grafikus, pasirenkant kaip atvaizduoti jau surinktus apklausos duomenis.

1.1.8. forms.google.com svetainės analizė mobiliajame telefone

Apklausų sistema Google Forms taip pat puikiai suderinta ir mobiliajame telefone. Mobilioji sąsaja atitinka ir kompiuterinę svetainės versiją, todėl nesunku rasti jau anksčiau kompiuterio naršyklėje matytas ikonas. Sistemos privalumas yra funkcionalumo nepriklausymas nuo platformos, tiek kompiuteryje, tiek mobiliajame, jis išlieka toks pats, todėl net sudėtingą apklausą vartotojas nesunkiai gali sukurti naudodamasis mobiliuoju telefonu, kai tuo tarpu panašios svetainės net nevisada suteikia galimybę patogiai atsakyti į apklausos klausimus mobiliuosiuose.

1.2. Analizės apibendrinimas

Visose keturiose svetainėse privaloma užsiregistruoti arba prisijungti prieš sukuriant apklausą. Taip pat visos apklausos sistemos suteikia galimybę sukurtais apklausomis dalintis respondentams siunčiant unikalią nuorodą į apklausą, arba išsiųsti elektroniniu paštu. Visose keturiose sistemose yra galimybė patogiai apžvelgti respondentų atsakymus ir vizualizuoti duomenis rezultatų grafikuose.

Ketrios analizuotos sistemos yra labai skirtingos ir dažniausiai naudojamos siekiant patenkinti skirtingus vartotojų poreikius. Google Forms yra didžiulė apklausų platforma, turi daugiausiai galimybių skirtingoms stilių, klausimų tipų, duomenų rinkimo ir eksporto konfigūracijoms. Mentimeter taip pat yra didelė platforma, tačiau ji yra daug labiau orientuota į prezentacijas ir nėra suderinta pavyzdžiui rinkti duomenis moksliniams darbams. Kalbant apie mokslinius darbus, apklausa.lt ir manoapklausa.lt abi yra puikiai suderintos rinkti duomenis iš didelio respondentų kiekio, eksportuoti juos ir vėliau panaudoti savo moksliniame darbe arba tolimesniuose skaičiavimuose. Pastarosiose svetainėse yra galimybė apklausas paversti viešomis, taip pritraukiant dar didesnę respondentų auditoriją. Google Forms turi galimybę patalpinti apklausą tiesiai į kitos svetainės turinį, panaudojant „iframe“ komponentą.

Patogiausias ir universaliausias įrankis daugeliui vartotojų yra Google Forms, todėl ši sistema tikrai galėtų tapti siekiama daugeliui kuriamų sudėtingų apklausų kūrimo sistemų. Tačiau universiteto fakulteto bendruomenei kuriama efektyvi balsavimo ir apklausų sistema projektuojama būti panašesne į Mentimeter platformą, kurioje apklausos yra orientuotos į prezentacijas, ar kitas grupės žmonių, kuriose gali būti užduodamas klausimas, o respondentams būtų kuo patogiau pasiekti balsavimą iš kompiuterių arba mobiliųjų telefonų. Sistemos kalba bus lietuvių, todėl buvo analizuojamos kelios lietuviškos svetainės. Apklausa.lt pasiūlė bene geriausią mobiliesiems

telefonams pritaikytą svetainės sprendimą, kuomet didesnis šriftas padidina paspaudimo erdvę ir yra efektyviausias mobiliųjų sprendimas tarp visų bandytų lietuviškų apklausų sistemų.

WEB versija					
	www.manoapklausa.lt	www.apklausa.lt	www.mentimeter.com	forms.google.com	mif-vote-app.herokuapp.com
Dizainas	+	-	+	+	+
Galimybė konfigūruoti apklausą vizualiai	-	-	+	+	-
Naudojimo paprastumas	+	+	+	+	+
Rezultatų pateikimas	+	+	+	+	+
Grafikų pasirinkimas	+	-	+	-	-
Duomenų eksportas	+	+	-	+	-
Failų įkėlimas	-	-	+	+	-
Apklausa „iframe“	-	+	-	+	-
Nemokama sistema	+	+	-	+	+

1 lentelė. Analizuojamos sistemos kompiuterio naršyklėje (sudaryta autoriaus).

Iš 1 lentelės matome, kad visos nagrinėtos apklausų sistemos turi pagrindinius jas vienijančius bruožus: surašyti +. Taip pat analizėje išryškėjo ir pagrindinis šių tinklalapių skirtumas, kuris gali turėti neigiamos įtakos vartotojui. Pavyzdžiui apklausa.lt ir manoapklausa.lt neturi galimybės konfigūruoti apklausos išvaizdą. Tai gali atliepti vartotojo naudojimo patirtį, nes esant poreikiui pritaikyti apklausą pagal tam tikrus stiliaus reikalavimus, bus pasirinkta kita apklausos sistema, suteikianti šį funkcionalumą. Mentimeter.com sistemos pagrindiniai trūkumai yra tai, jog sistema yra mokama ir nesuteikia galimybės eksportuoti duomenų. Šios sistemos prezentacijų taip pat negalima įkelti į kitos svetainės kodą iframe elemento pagalba, palyginimui apklausa.lt ir Google Forms sistemos turi šį funkcionalumą. Iš lentelėje pateiktų duomenų, matome, jog failų įkėlimo funkcionalumas įdiegtas tik Mentimeter ir Google Forms sistemose, kitos apklausų sistemos šios galimybės neturi. Silpniausias kompiuteriu pasiekiamos apklausų sistemos dizainas yra apklausa.lt sistemoje. Vizualinė apklausos konfigūracijos galimybė suteikiama tik Mentimeter ir Google Forms sistemose, likusios sistemos neturi tokio funkcionalumo.

Mobile versija				
	Stilius	Paspaudimas	Teksto dydis	Rezultatų skiltis
manoapklausa.lt	+	-	-	+
apklausa.lt	-	+	+	+
mentimeter.com	+	+	+	+
forms.google.com	+	+	+	+
mif-vote-app.herokuapp.com	+	+	+	+

2 lentelė. Analizuojamos sistemos mobiliuosiuose įrenginiuose (sudaryta autoriaus).

Lentelėje Nr. 2 vaizduojami nagrinėtų apklausos sistemos mobiliosios versijos analizės rezultatai. Testuojant mobiliąsias sistemos versijas buvo išsikelti labai konkretūs reikalavimai, nes dažniausiai sistemos visiškai atitinka kompiuterines savo versijas ir tik stiliai pritaikomi mažesniems ekranams. Vienas iš pagrindinių kriterijų yra paspaudimo patogumas mobiliajame telefone. Visos nagrinėtos sistemos turėjo pakankamai dideles erdves paspaudimui, todėl buvo pakankamai lengva pasirinkti tinkamą variantą, išskyrus manoapklausa.lt sistemą, kurios mobilioji versija yra silpna, ir paspaudimo erdvė buvo pakankamai maža, kad būtų lengva pasirinkti ne tą variantą, kurio tikėjosi vartotojas. Visose sistemose išlieka galimybė ir mobiliajame telefone patogiai peržiūrėti apklausų metu surinktus duomenis informatyviuose grafikuose.

1.3. Kuriamai sistemai iškelti reikalavimai

Susitikimuose su dėstytoju ir pasitelkiant analizės metu surinktus duomenis apie keturias visiškai skirtingas sistemas buvo išsikelti kriterijai naujai kuriamai efektyviai balsavimo ir apklausų sistemai fakulteto bendruomenei. Vienas iš susiformavusių kriterijų yra kuo paprastesnis apklausos atsakymas apklausų sistemoje, todėl kuriamoje sistemoje vienas iš reikalavimų yra galimybė atsakyti į apklausas neprisijungus prie sistemos. Tokiu atveju nebus galimybės atskirti ar vartotojas balsavo, todėl kitas reikalavimas yra skaityti vartotojo IP adresą ir pridėti jį prie konkrečioje apklausoje balsavusiųjų sąrašo. Kitas reikalavimas sistemai yra patogus valdymas mobiliesiems įrenginiams, tikslas yra kuo labiau padidinti balsavimo atsakymų užimamą erdvę, taip sumažinant vartotojų daromas klaidas ir pagerinant vartotojų patirtį naudojantis balsavimo sistema. Dar vienas reikalavimas yra kuriamą sistemą sukurti lietuvių kalba. Taip pat išsikeltas sistemos reikalavimas yra lengvas ir patogus jos naudojimas, todėl apklausos turėtų būti sukuriamos greitai ir į jas atsakyti privalo būti lengva. Taipogi,

turi būti užtikrinama kuo paprastesnė prieiga prie apklausų vartotojams su mobiliaisiais telefonais ir patogios vartotojo sąsajos sukūrimas.

2. Projektavimas

2.1. Svetainės panaudos atvejų diagrama

Šiame poskyryje žemiau (1 pav.) pateiksiu panaudos atvejų schemą.

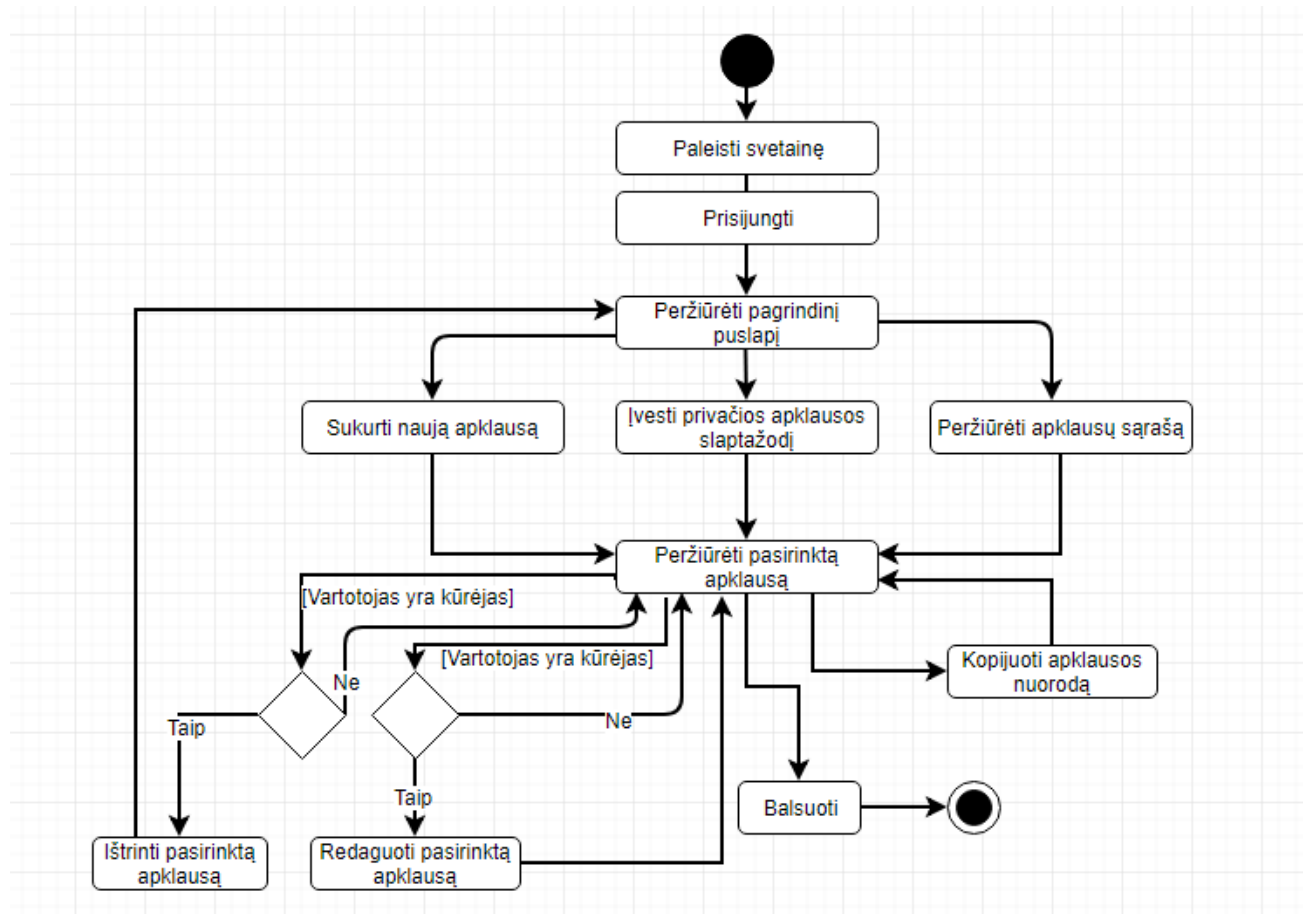


1 pav. Panaudos atvejų schema (sudaryta autoriaus).

Paveikslėlyje Nr. 1 matome, kad prisijungęs naudotojas gali – balsuoti, registruotis, peržiūrėti (tiek viešą tiek privačią bei apklausų sąrašą), ištrinti, sukurti, apklausą, atsijungti ir redaguoti apklausą. Kitaip yra su neautentifikuotais vartotojais – jie gali tik peržiūrėti viešas bei privačias apklausas, prisijungti/registruotis ir balsuoti. Neautentifikuotas vartotojas pasiekti privačias slaptažodžiu pasiekiamas apklausas gali tik žinodamas 6 skaitmenų kodą. Autentifikuotas vartotojas gali peržiūrėti visas savo sukurtas apklausas spustelėjęs ant pagrindiniame puslapyje esančio skirtuko „Mano apklauso“. Spustelėjus ant savo sukurtos apklausos, autentifikuotas vartotojas turi galimybę ją redaguoti arba ištrinti.

2.2. Svetainės veiksmų diagrama

Šiame poskyryje žemiau (2 pav.) pateikiama svetainės veiksmų diagramą autentifikuotam vartotojui.



2 pav. Svetainės veiksmų diagrama autentifikuotam vartotojui (sudaryta autoriaus).

Paveikslėlyje Nr. 2 matome visos svetainės veiksmų diagramą iš autentifikuoto vartotojo perspektyvos. Pirmą kartą paleidus svetainę, vartotojas prisijungia ir mato pagrindinį svetainės puslapį kuriame matomos viešos apklausos. Vartotojas turi galimybę įvesti privačios apklausos slaptažodį ir peržiūrėti viešų ir jo sukurtų apklausų sąrašus atskirtus skirtukais (angl. tabs). Jei vartotojas suveda privačios apklausos slaptažodį, arba paspaudžia ant mygtuko „Įvesti privačios apklausos slaptažodį“ ir suveda slaptažodį teisingai, arba sukuria naują apklausą - yra pasiekiamas apklausos peržiūrėjimo puslapis. Pasiekęs bet kurią apklausą, vartotojas turi galimybę balsuoti, tačiau jei apklausa yra sukurta jo paties, jis taip pat gali apklausą redaguoti, ją ištrinti arba kopijuoti apklausos nuorodą.

2.3. Agile - Lean

Vis daugiau projektų pasaulyje yra vystomi remiantis Agile Manifesto teiginiais ir principais. Agile iš tiesų yra bendrinis terminas kuris apibūdina daug iteracinių - inkrementinių projektų valdymo metodų, kurie atitinka pagrindinius manifeste aprašytus principus [4]. Projektuose vadovaujantis Agile principais siekiama palengvinti jų kūrimo procesą, prisitaikyti prie sparčiai kintančių reikalavimų joms. Konkretus Agile projektų valdymo metodas, naudotas kuriant balsavimo ir apklausų sistemą fakulteto bendruomenei buvo Lean, kuris naudojami tokiais Lean metodikos principais, kaip vykstančių darbų ribojimas rezultatų tobulinimui. Vadovaujantis Lean principais, susikoncentruojama į konkrečius dienos/savaitės darbus ir stengiamasi neatsiplėšti nuo numatyto plano ir dirbti ties suplanuotais darbais sistemiškai tobulinant kuriamą produktą. Lean sistema padeda šalinti nuostolius, atsisakyti nereikalingos ar nereikšmingos veiklos, padeda mažinti išteklius ir sukurti kuo tobulesnį, konkurencingesnį produktą [5].

Tipiškas Agile projektų valdymo principais vykdomo projekto programų kūrimo ciklas trunka nuo 1 iki 4 savaičių, kiekvieno ciklo pabaigoje įvertinant rezultatus, formuojant ateities prioritetus. Naudojant Agile principus balsavimo ir apklausų sistemos fakulteto bendruomenei darbų planas buvo skaidomas į mažesnes dalis ir padalinamas į savaitės trukmės iteracijas kuriant konkretų tolimesnių veiksmų planą ir dalinant planuojamus darbus į atitinkamas dalis. Kiekvienos savaitės iteracijos metu, naudojant Lean metodus buvo koncentruojamasi į tik tos savaitės darbus ir stengiamasi susikoncentruoti į kuo geresnį produktą. Visgi projekto vystymo metu buvo daug pokyčių, kintančių prioritetų ir prisitaikymo prie technologinių galimybių, todėl planai iteracijose buvo keičiami realiu laiku. Daugiau informacijos apie iteracijas, darbų pasiskirstymą projekto eigoje atvaizduojama 3 lentelėje apačioje.

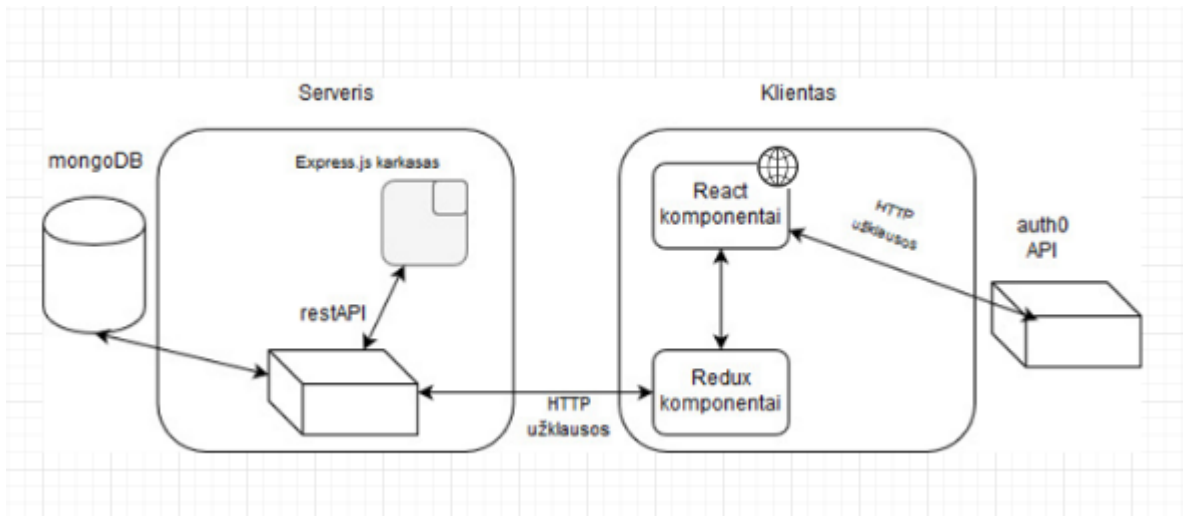
Projektas padalintas į savaitės trukmės iteracijas		
2018-03-04	2018-03-11	Susitikimas su dėstytoju temos aptarimui, baigiamojo bakalaurnio darbo temos registravimas, administraciniai darbai.
2018-03-11	2018-03-18	Apklausų sistemų kompiuterinių ir mobiliųjų versijų analizė, kriterijų nustatymas, darbo struktūros formavimas.
2018-03-18	2018-03-25	React + Redux mokymasis, pavyzdinės programos, apklausų sistemos kūrimo pradžia, šabloninis kodas, pirmieji komponentai, ES6, JSX sintaksė, babel, webpack įrankiai, darbas su aplikacijos būsena.

2018-03-25	2018-04-01	Index komponentas, jame instaliuojamas ReactRouter paketo naudojimas, aplikacijos pirmojo puslapio komponentų kūrimas: Header, Footer, Content.
2018-04-01	2018-04-08	Aplikacijos perkėlimas į Express.js karkasą, darbas su karkasu, srautų paskirstymas, nuorodų kelių konfigūravimas, nustatymas React aplikacijos teisingam veikimui (visada grąžinamas index failas) ir kiti konfigūraciniai darbai, mokomasi axios paketo tolimesniam darbui su API.
2018-04-08	2018-04-15	MongoDB įdiegimas sistemoje, demonstracinių duomenų kūrimas, redux veiksmų kūrėjai (angl. "action creators") duomenims gauti, kuriami duomenų modeliai, sukurtas API darbui su duomenimis.
2018-04-15	2018-04-22	Naujo balsavimo kūrimo komponentas, atitinkami redux veiksmų kūrėjai, redux-form komponentas ir jo konfigūracija, atitinkamų laukelių kūrimas, validacija, kuriamas naujo balsavimo objektas, išsaugomas mongodb su reikalingais duomenimis.
2018-04-22	2018-04-29	Balsavimo ištrynimo funkcionalumas, klaidų taisymas, balsavimo peržiūros puslapio kūrimas, vizualizacijų su Chart.js komponentu implementavimas, duomenų perdavimas, duomenų bazės struktūros keitimas, pridėta spalva prie pasirinkimų, balsų kiekis, balsavimo galimybė, IP adreso nuskaitymas ir pridėjimas prie balsavusiųjų sąrašo.
2018-04-29	2018-05-06	Rašto darbo formavimas, projektavimo ir programavimo dalių aprašai, pritaikoma struktūra, pildomas šablonas, pildomi literatūros sąrašai, išnašos, kuriamos UML diagramos.
2018-05-06	2018-05-13	Auth0 autentifikacijos implementavimas sistemoje, komponentų tobulinimas, klaidų taisymas, redux-form laukelių validacijos tobulinimas, klaidų gaudymas, privačių anketų funkcionalumas, slaptažodžio įvedimo komponentas, modalai.
2018-05-13	2018-05-20	Galutinis susitikimas su dėstytoju, darbo aptarimas, klaidų taisymas, administraciniai darbai, rašto darbo užbaigimas, ateities planų ir rekomendacijų sąrašo sudarymas.

3 lentelė. Projektas padalintas į savaitės trukmės iteracijas (sudaryta autoriaus).

2.4. Bendra sistemos architektūra

Šiame poskyryje pateiksiu (3 pav.) bendrą sistemos architektūros schemą.



3 pav. Bendra sistemos architektūros schema (sudaryta autoriaus).

Paveikslėlyje Nr. 3 vaizduojama bendra sukurtos apklausų ir balsavimo sistemos – serverio ir kliento - architektūra. Kliento dalyje tarpusavyje susiję React ir Redux komponentai atsakingi už visą vartotojui matomą sąsają. React komponentai formuoja vartotojui matomų HTML elementų visumą, kai tuo tarpu Redux komponentai yra atsakingi už sistemos būseną, bendrauja su sukurtu API servisu serverio dalyje. Redux komponentams atnaujinus sistemos būseną, atitinkami React komponentai yra vaizduojami iš naujo su naujais duomenimis. React komponentai bendraudami su išoriniu autentifikacijos įrankio Auth0 API, gauna vartotojo prisijungimo simbolį, taip autentifikuodami vartotoją ir vaizduodami atitinkamus autentifikuoto vartotojo komponentus. Serverio dalyje naudojamas Express karkasas, kuriame taip pat sukurtas sistemos API servisas. Klientui darant užklausą į Express serverį, pagal užklauskos nuorodą nusprendžiama ar turėtų būti pasiektas API servisas, ar grąžintas vieno puslapio HTML dokumentas, kuriame React yra atsakingas už vaizduojamus komponentus. Jei Express serveris gauna užklausą į sukurtą API servisą, tuomet pagal užklauskos tipą duomenys gaunami JSON formatu iš mongoDB serverio, kuriame yra patalpintos visos apklausos. Rezultatai rasti mongoDB serveryje per API servisą yra siunčiami atgal Redux komponentams ir tuomet React dinamiškai vaizduoja atitinkamus komponentus matomus vartotojui.

2.5. Technologijos

2.5.1 React

Kuriamos apklausų ir balsavimo sistemos dalis, su kuria tiesiogiai interakcijomis bendrauja vartotojas, toliau darbe vadinama aplikacija. Šiai aplikacijai kurti, buvo pasirinkta Facebook inžinierių sukurta ir palaikoma React biblioteka [1]. Ši biblioteka yra paremta JavaScript kalba ir yra skirta vartotojų sąsajų kūrimui. Nors React nėra naujovė ir programavimo rinkoje yra jau kiek daugiau nei tris metus, bibliotekos populiarumas kyla ir programuotojai prisitaiko prie tendencijų taip tik didindami React įtaką JavaScript bendruomenėje. React nėra karkasas, todėl negalime sukurti sudėtingos aplikacijos naudojant tik šią biblioteką. Ši biblioteka yra paremta komponentais (angl. „component-based“), kaip ir keletas jos konkurentų: AngularJS ir Vue.js. React aplikacija, iš tiesų yra vienas didžiulis komponentas, sudarytas iš daug mažų, tarpusavyje susijusių, komponentų.

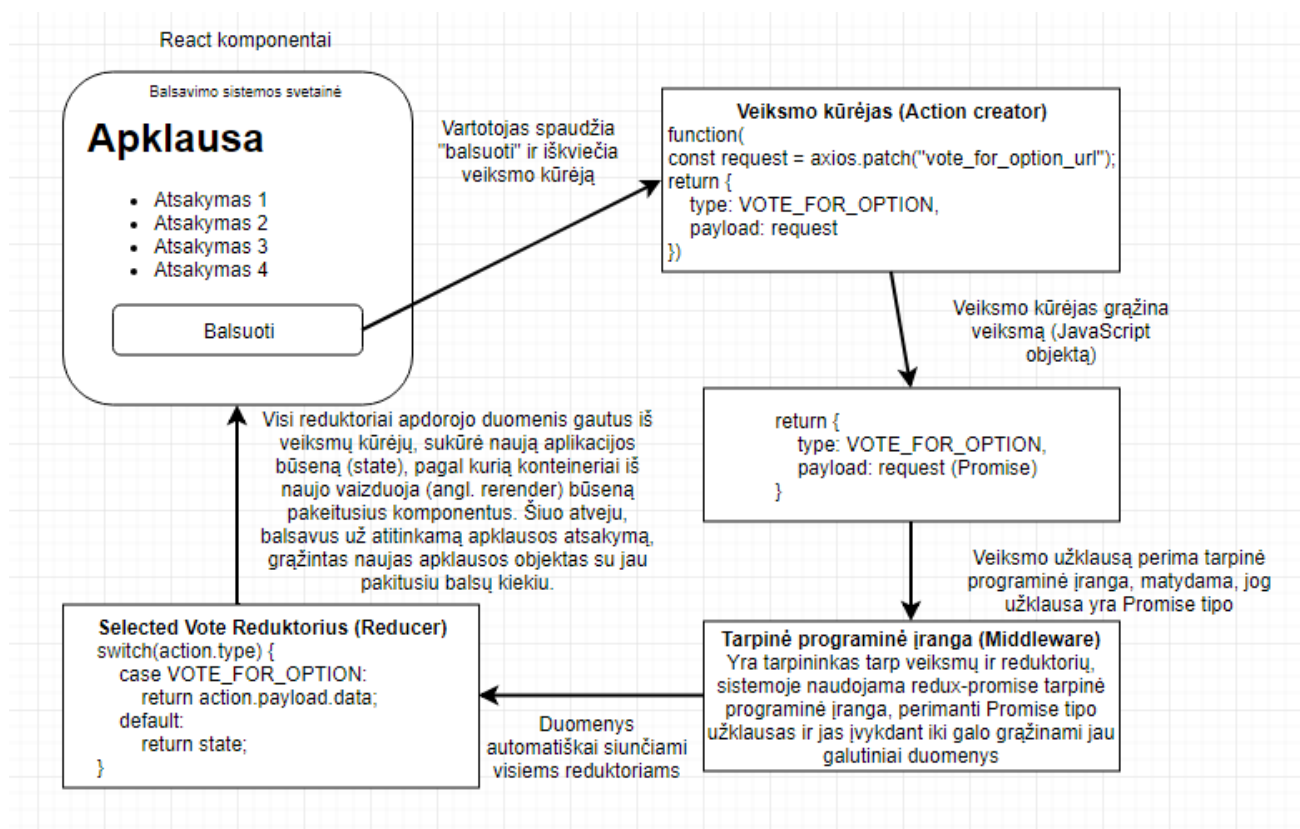
Vienas iš pagrindinių React bibliotekos naudojimo pranašumų yra virtualus DOM objektas. DOM („Document Object Model“) manipuliacija modernioje, interaktyvioje aplikacijoje yra butelio kakliukas (angl. „bottleneck“), nes yra lėtesnis nei didžioji dauguma JavaScript operacijų. Blogiausia, jog JavaScript aplikacijose DOM yra atnaujinamas žymiai dažniau, nei to iš tikrųjų reikėtų ir tai tapo labai opia problema. Būtent React virtualaus DOM pagalba buvo išspręsta ši problema, nes joje kiekvienas DOM objektas turi savo virtualų atitikmenį, kopiją. Manipuliuojant virtualias kopijas veiksmai atliekami žymiai greičiau, nes komponentai nėra atvaizduojami ekrane. Visi pakeitimai aplikacijoje pakeičia visą virtualaus DOM objektą, tačiau vėliau šis virtualus objektas yra palyginamas su prieš pakeitimą buvusiu virtualiu DOM objektu, ir tuomet pakeitimai atvaizduojami tik tuose komponentuose, kurie buvo pakeisti. Trumpai tariant, React dėka galime keisti tik tuos komponentus, kurie buvo pakeisti, be reikalo neperkraudant visų likusių nepakitusių elementų. Tai leidžia sutaupyti labai daug resursų ir žymiai paspartinti aplikacijos veikimą.

2.5.2 Redux

Redux yra nuspėjamos būsenos konteineris JavaScript aplikacijoms [2]. Šis įrankis padeda kurti stabilias sistemas nuspėjant jų elgesį, paleisti jas skirtingose aplinkose, lengvai testuoti. Šis įrankis

yra dažnai naudojamas React aplikacijose, nes puikiai atlieka savo funkciją ir yra atsakingas už visos aplikacijos būseną.

Mano kuriamoje balsavimo sistemoje, Redux užima tarpininko poziciją tarp React aplikacijos (matomos vartotojui) ir duomenų. Jis yra atsakingas už aplikacijos būseną, tokius laukelius kaip šiuo metu pasirinktas balsavimas ir t.t. Redux pagalba kuriame įvairius veiksmus („action creators“), kurie grąžina paprastą JavaScript objektą su tipu ir duomenimis, kai to reikia. Tuomet reduktorių (angl. „reducers“) pagalba priskiriame duomenis su atitinkamu laukeliu Redux būsenoje, kurią galime pasiekti bet kuriame React aplikacijos komponente. Norint komponente naudoti atitinkamus reduktorių laukelius ir juose esančią informaciją, turime juos importuoti į React komponento būseną tokių metodų pagalba kaip: `mapStateToProps` ir `mapDispatchToProps`. Toliau aplikacijoje nesunkiai implementuojami atitinkami „actions.js“ faile aprašomi veiksmai „actions“, kuriuos galime surišti su atitinkamais veiksmiais aplikacijoje, pvz. mygtuko paspaudimu.



4 pav. Redux veikimo kuriamoje balsavimo sistemoje diagrama (sudaryta autoriaus).

Paveikslėlyje Nr. 4 matome esminį Redux veikimo ir jo bendravimo su React komponentais principą. Vartotojas spustelėjęs ant tam tikro mygtuko arba atlikęs bet kokią kitą veiksmą, iškviečia

prie to veiksmo priskirtą veiksmo kūrėją (angl. action creator). Visi reikalingi veiksmų kūrėjai buvo importuoti į komponento JavaScript failą ir priskirti prie atitinkamų komponentų. Šiuo atveju pavyzdinei diagramai buvo pasirinkta balsavimo funkcija, kai šalia veiksmo kūrėjo yra su Axios biblioteka formuojama užklausa į mūsų sukurtą API servisą, kuris dirba su MongoDB duomenų baze. Veiksmo kūrėjas grąžina paprastą (angl. plain) JavaScript objektą su veiksmo tipu ir Axios užklausoje į norimus duomenis suformuotu Promise objektu. Vėliau paveikslėlyje atspindimas tarpinės programinės įrangos elgesys – atpažįstamas Promise tipo objektas, todėl įdiegta Redux-Promise tarpinė programinė įranga perima objektą ir užtikrina galutinį duomenų išgavimą ir priskyrimą tam pačiam payload laukeliui veiksmo kūrėjo objekte. Galiausiai suformuojamas objektas savyje turi tipą ir užklausoje grąžintus duomenis. Šis objektas yra siunčiamas per visus reduktorius, kurie pagal objekto tipą nustato ar šis objektas yra skirtas jiems. Kadangi šiame paveikslėlyje nurodyta balsavimo ciklo diagrama, sistemoje sukurtas reduktorius Selected Vote yra atsakingas už šį objektą ir priskiria naujai gautus balsavimo duomenis prie jam priskirto Redux State laukelio selectedVote. Vėliau šis laukelis React VoteDetail komponente yra priskiriamas prie React būsenos (angl. state) su mapStateToProps funkcijos pagalba. Pasikeitus React būsenai, atitinkamas komponentas VoteDetail yra iš naujo vaizduojamas (angl. rerender) pakeičiant tik pakitusius elementus, šiuo atveju perpiešiamas duomenų grafikas.

2.5.3 MongoDB NoSQL duomenų bazė

Šiuolaikinės įmonės vis ieško lankščių ir greitų technologinių sprendimų įgauti pranašumui prieš jų konkurentus. Būtent tai ir yra viena iš priežasčių lėmusių NoSQL duomenų bazių kilimą prieš standartines reliacines duomenų bazes. NoSQL duomenų bazės gali būti skirtingų tipų: Stulpelių, Dokumentų, Rakto, Grafų ir Įvairių modelių mišinys [Str17]. Mano kuriamoje balsavimo sistemoje bus naudojama dokumentų tipo MongoDB duomenų bazė. Kiekvienas įrašas naudojamoje duomenų bazėje bus atskiras JSON formato objektas.

NoSQL duomenų bazių pranašumas prieš tradicines reliacines duomenų bazes – jei kuriami duomenys turi gilesnę hierarchinę struktūrą, gali būti labai sudėtinga patalpinti tokius duomenis į tradicinę reliacinę duomenų bazę, todėl naudojant dokumentų tipo NoSQL duomenų bazę galima lengvai implementuoti ir pačią sudėtingiausią hierarchinę struktūrą [3]. Antrasis NoSQL duomenų bazių pranašumas yra dinamiška duomenų struktūra/hierarchija, nes norint pakeisti reliacinės duomenų bazės struktūrą reikia keisti visą schemą. Trečiasis, ir bene svarbiausias ilgalaikėje

perspektyvoje, NoSQL duombazių pranašumas prieš reliacines duomenų bazes yra plėtimo galimybės. Didėjant duomenų kiekiui, reliacinės duomenų bazės į užklausas atsako vis lėčiau, ir vienintelė išeitis yra plėsti sistemą vertikalčiai: tobulinti serverį į geresnį, kai tuo tarpu NoSQL duomenų bazių sistema gali būti tobulinama horizontaliai: didinant serverių kiekį. Toks sprendimas yra gerokai pigesnis ir paprastesnis.

3. Programavimas

3.1. Programavimo aplinkos paruošimas

Balsavimo ir apklausų sistemos fakulteto bendruomenei kūrimo pirmasis programavimo žingsnis yra pradinis sistemos kodas. Siekiant sumažinti pradinio sistemos kūrimo etapo laiką, internete buvo rasta Stephen Grider GitHub repozitorija, kurioje implementuotos standartinės priklausomybės (angl. „dependencies“). Šis kodas yra visiškai standartinė React aplikacija, kurioje kodas bus rašomas naujuoju ES6 standartu (oficialiai vadinamu ECMAScript 2015) ir naudojant JSX – šiek tiek kitokią JavaScript sintaksę. JSX sintakse rašomas kodas yra lengvai suprantamas ir vizualiai patrauklus programuotojams, todėl jo pagalba lengvai galima kurti HTML komponentus, o „babel“ įrankis transpiliuoja kuriamą JSX kodą į standartinį JavaScript. Ankščiau minėtas ES6 standartas taip pat nėra visiškai palaikomas naršyklėse, todėl React aplikacijoje rašomas ES6 standarto kodas nebūtinai veiks naršyklėje. Tam yra naudojami „webpack“ ir „babel“ įrankiai, kurie padeda rašoma kodą transpiliuoti ir pritaikyti naršyklėms, naudojant pačius naujausius standartus. Transpiliuotas kodas yra sutraukiamas į vienintelį bundle.js JavaScript failą. Pradinis kodas, atsisiųstas iš Stephen Grider repozitorijos, gali būti instaliuotas ir paleistas naudojant npm („node package manager“). Paleidžiant komandą npm install yra įrašomos visos nurodytos priklausomybės („dependencies“) ir įrankiai. Visi programoje naudoti paketai ir komponentai iš trečiųjų šalių taip pat buvo įdiegiami panaudojant npm. Paleidus komandą npm start, lokaliai paleidžiamas node aplinkoje veikiantis serveris, kuriame ir veikia kuriama React aplikacija. Paleistame serveryje „webpack“ ir „babel“ įrankiai transpiliuoja aplikacijos kodą į naršyklei suprantamą standartą, suformuoja bundle.js JavaScript failą ir paleidžia testavimo aplinką vietiniame serveryje „localhost“.

3.2. React Router

Ankščiau internetinės svetainės veikė vartotojui bendraujant su serveriu – vartotojas siunčia puslapio užklausą į serverį, o serveris į užklausą atsako su atitinkamu nuorodai HTML puslapiu. Vartotojui atlikus veiksmą puslapyje su jame patalpinta nuoroda, jis siųsdavo naują užklausą, į kurią serveris atsako kraudamas kitą HTML puslapį. Trumpai tariant, vartotojas naršyklėje keičia nuorodą, o serveris krauna atitinkamą dokumentą.

React Router veikimo principas yra kitoks – vartotojui atlikus interakciją su puslapiu, šis komponentas perima nuorodą, neleidžia siųsti užklausos į serverį, ir vietoj to pats nusprendžia dinamiškai parodyti nuorodai priskirtą komponentų rinkinį [Gri17]. React Router naudoja History paketą, kuris veikia programos užkulisiuose ir perima vartotojo pakeistą nuorodą ir grąžina ją React Router komponentui, kuris ir nusprendžia kuriuos komponentus atvaizduoti pagal gautą nuorodą. Ši architektūra yra kitaip vadinama vieno puslapio aplikacija (angl. SPA – „Single Page Application“). Tai reiškia, jog vartotojas nebegauna vis skirtingų HTML puslapių po kiekvienos užklausos į serverį, vietoj to, visą laiką dirbama tik su vienu HTML dokumentu, kuriame React Router dėka dinamiškai išdėstomi atitinkami komponentai. Kiti papildomi stilių ir scenarijų failai kraunami kartu su HTML dokumentu vieną kartą.

3.3. Pagrindinių aplikacijos komponentų kūrimas

Vartotojui paleidus svetainę yra rodomi atitinkami nuorodai komponentai, kurie buvo kuriami inkrementiškai pridedant aplikacijai vis daugiau funkcionalumo. Pirmasis sistemai sukurtas komponentas buvo app, kuriame, pagal nuorodą visada yra rodomi Header ir Footer komponentai, bei turinys tarp jų. Taip sukuriamas įprastos aplikacijos pojūtis, kai vartotojui visada yra pasiekiami tokie Header komponentai kaip prisijungimo galimybė ir vartotojo profilio informacija. Tuomet buvo sukurtas pagrindinis aplikacijos puslapio komponentas – IndexPage. Šiame komponente atsiranda vartotojui aktuali informacija, nurodymai kaip naudotis sistema. Sekantis kuriamas komponentas buvo VoteList, kuriame vaizduojami visi duomenų bazėje esantys apklausų dokumentai. Pagrindinio svetainės puslapio komponentų išdėstymo paveikslas yra įkeltas į šio darbo priedus (1 Priedas). Pagrindiniame puslapyje esantys komponentai išdėstomi priklausomai nuo to, ar vartotojas yra prisijungęs. Vartotojui prisijungus, atsiranda skirtuko komponentas, kuriame galima naviguoti tarp viešų ir vartotojo sukurtų apklausų sąrašų. Sekantis kuriamas komponentas buvo VoteNew, kuris sukuria apklausos įrašą duomenų bazėje (2 Priedas). Įgyvendinant komponentą įdiegtas redux-form papildinys, kuris leidžia valdyti visus formoje esančius langelius, jų reikšmes. Papildinio dėka sukurta funkcija dinamiškai pridėti daugiau atsakymų pasirinkimų apklausos objektui. Vartotojui teisingai suvedus formos duomenis, suformuojamas apklausos objektas, kuris kreipiantis į sistemoje sukurtą API servisą sukuria įrašą duomenų bazėje. Vėliau buvo kuriamas VoteDetail komponentas, kuris vaizduoja visą informaciją apie duomenų bazėje sukurtą įrašą (4 Priedas). Vaizduojant šį komponentą, vartotojas turi galimybę balsuoti už atitinkamą pasirinkimą, o jei vartotojas yra apklausos kūrėjas,

suteikiama galia šią apklausą ištrinti arba redaguoti. Vėliau, sistemoje įdiegta slaptažodžių sistema, todėl sukurtas VotePrivate React komponentas, pasinaudojant jau anksčiau minėtu redux-form papildiniu, leidžiantis vartotojui įvesti 6 skaitmenų privačios apklausos slaptažodį (3 Priedas). Jei slaptažodis teisingas, vartotojas nukreipiamas į jau anksčiau sukurtą VoteDetail komponentą su atitinkamos apklausos duomenimis. VoteDetail komponente taip pat naudojamosi paketu atsisiųstu su npm (angl. node package manager) pagalba ir įdiegtu kuriamoje aplikacijoje. Šio atsisiųsto paketo dėka sukurtoje diagramoje, gražiai vaizduojami iš duomenų bazės gauti apklausos duomenys – apklausos pasirinkimų variantai, balsų kiekis už kiekvieną pasirinkimą bei apklausos metu pasirinkimui priskirta spalva.

3.4. Redux konteineriai

Paralelėje su aplikacijoje kuriamais React komponentais, buvo įgyvendinamos ir Redux kodo dalys. Susiejant Redux būseną su React komponentais, React komponentai paaukštinami į konteinerius. Konteineriai, naudojantys Redux būseną kuriamoje aplikacijoje: VoteDetail, VoteList, VoteNew, VotePrivate. Redux su React siejantys metodai yra mapStateToProps ir mapDispatchToProps importuoti iš su npm pagalba įdiegto react-redux paketo. Eksportuojant sukurtą konteinerį anksčiau minėti metodai yra sujungiami connect() funkcijos pagalba. Taip sujungus React komponentus su Redux vadovaujama būsena, React komponentai su Redux pagalba turi galimybę gauti duomenų bazėje esančius duomenis. Redux funkcijos yra aprašomos actions aplanke esančiame index.js faile. Jame esančios funkcijos su Axios paketo pagalba formuoja užklausas į sistemoje sukurtą API galines nuorodas, kuriose ir vyksta bendravimas su duomenų baze priklausomai nuo užklausos tipo. Užklausų, kurios yra apsaugotos autentifikacijos simbolio (angl. token) patikrinimu, antraštės (angl. headers) yra papildomos autentifikacijos simboliu, taip užtikrinant, kad tik autentifikuoti vartotojai gali pasiekti apsaugotus resursus.

3.5. NoSQL duomenų bazės struktūra

Kuriamoje sistemoje buvo naudojama dokumentų tipo NoSQL duomenų bazė. Joje buvo talpinamos apklausos, kiekvienai apklausai sukuriant atskirą dokumentą. Duomenų bazė yra patalpinta mLab platformoje (<http://www.mlab.com/>), taip užtikrinant, kad duomenų bazė išoriniame serveryje yra visada pasiekiamą. Duomenų bazėje saugomo dokumento laukeliai nurodomi sąraše:

- `_id` – Numatytasis mongodb laukelis kuriame yra unikali 12 baitų dokumento identifikacinė reikšmė.
- `IPsVoted` – Masyvas, kuriame renkami balsavusių vartotojų IP adresai.
- `Options` – Apklausos atsakymų objektų masyvas. Vienas atsakymas masyve yra objektas, savyje laikantis patį atsakymą, balsų skaičių ir atvaizdavimo grafike spalvą.
- `Name` – String tipo apklausos klausimo pavadinimas.
- `User` – String duomenų tipo klausimą sukūrusio vartotojo pavadinimas.
- `createdAt` – Numatytasis mongodb laukelis su dokumento sukūrimo tikslu laiku.
- `updatedAt` – Numatytasis mongodb laukelis su tikslu dokumento paskutinio atnaujinimo laiku.

3.6. Autentifikacija su Auth0

Kuriamos sistemos autentifikacijai užtikrinti buvo pasirinktas Auth0 įrankis. Jo dėka vartotoją kuriamoje sistemoje galima autentifikuoti daugybe būdų: kurti savo vartotojų duombazę, prisijungti su socialinių tinklų duomenimis ir t.t. Pagrindinis kriterijus buvo tai, jog Auth0 turi galimybę integracijai su MIF fakulteto vartotojų duomenimis, tačiau šis funkcionalumas nebuvo įdiegtas baigiamojo darbo gynimui. Tokia galimybė vistiek yra, ir yra įtraukta į tolimesnių darbų sąrašą, kurie bus vykdomi netolimoje ateityje ir įtraukti į kuriamą sistemą. Auth0 įrankis bus sukonfigūruotas kaip tapatybės teikėjas (angl. „Identity Provider“), ir SAML kalba galės tikrinti vartotojo duomenis su MIF vartotojų duomenų baze. Patikrai grąžinus teigiamą rezultatą, balsavimo sistemai bus siunčiamas autentifikacijos simbolis (angl. „token“), tačiau sistema ir toliau veiks taip pat kaip ir baigiamojo darbo gynimo metu.

Autentifikacijai užtikrinti buvo sukurta atskira klasė, kurioje yra formuojamas sujungimas su Auth0 platforma. Ten nurodyti teisingi platformos duomenys užtikrina susijungimą su išorine platformos duomenų baze. Vartotojas prisijungimo metu yra nukreipiamas į atitinkamą nuorodą, kurioje yra autentifikuojamas. Į balsavimo ir apklausų sistemą yra grąžinamas vartotojo prisijungimo simbolis, kuris ir užtikrina, jog vartotojas sėkmingai prisijungė.

4. Sistemos apžvalga

Baigiamojo bakalaurinio darbo gynimui paruošta balsavimo ir apklausų sistema fakulteto bendruomenei turi įdiegtą privačių apklausų kūrimo funkcionalumą. Tokiu būdu prisijungę vartotojai gali kurti privačias apklausas, kurios nepasirodo bendrajame viešų apklausų sąrašė ir yra pasiekiamos dviem būdais: dalinantis apklausos nuroda su unikalia apklausos identifikacine reikšme arba dalinantis šešių skaitmenų slaptažodžiu. Slaptažodis neatlieka apsauginės funkcijos, tai yra apklausos sukūrimo metu sugeneruota šešių skaitmenų reikšmė, padedanti respondentams patogiai pasiekti apklausą iš bet kurio prietaiso, labiausiai orientuojantis į mobiliuosius įrenginius. Slaptažodžio funkcionalumas padeda sumažinti laiką vartotojui ieškant apklausos dideliame sąrašė, nes pakanka prisijungti į pagrindinį sistemos puslapį, spustelti mygtuką „įvesti privačios apklausos slaptažodį“, ir atsakyti į užduotą klausimą. Visose apklausose gali dalyvauti ir neautentifikuoti vartotojai, eliminuojant prisijungimo poreikį buvo sumažintas interakcijų kiekis taip pagerinant vartotojo patirtį dirbant su sistema.

Sistema atitinka darbo pradžioje išsikeltus kriterijus ir tikslus, nes ja yra paprasta ir patogiu naudotis, ji yra lietuviška ir pritaikyta greitam apklausos atlikimui. Sistemos mobiliojoje versijoje yra patogiu paspausti ant norimo atsakymo varianto. Visuose sistemos įvesties laukeliuose yra įdiegta patikra, todėl visi laukeliai yra tikrinami ar yra įvesti teisingai, pvz. slaptažodis turi būti iš 6 skaitmenų. Įvedus klaidingą informaciją į laukelius, atvaizduojamas klaidos pranešimas apie konkretų neatitikimą.

Išvados ir rekomendacijos

Šio darbo metu sukurta balsavimo ir apklausų fakulteto bendruomenei sistema, tačiau visuomet yra galimybė bet kokią sistemą tobulinti. Ne išimtis yra ir šis projektas, kuriam jau nuo pat projekto pradžios buvo formuojamas darbų sąrašas, kuris buvo nuolat pildomas. Vieni iš rekomenduojamų prioritetinių šios sistemos papildymo ir patobulinimo darbų yra šie:

- Autentifikacijos galimybė tik fakulteto nariams. Panaudoti jau įdiegtą Auth0 platformą ir sukonfigūruoti ją kaip tapatybės tiekėją. Tuomet SAML protokolo dėka tikrinti duomenis su MIF duomenų baze ir gauti informaciją apie vartotojus.
- Galimybė filtruoti apklausas pagal fakulteto nario vardą (jei yra galimybė gauti tokia informaciją iš MIF duombazės per SAML).
- Sistemos orientavimas į mobiliųjų telefonų vartotojus.
- Rezultatų vizualizavimo konfigūracijos. Galimybė pasirinkti skirtingus grafikus, pagal vartotojo poreikius.
- Klausimų įvairovės diegimas. Galimybė sukurti kelių atsakymų klausimus, iškrentančio atsakymų sąrašo klausimus, atsakymų eilutės tekstu klausimus, matricų klausimus.
- Kelių klausimų apklausos funkcionalumas.
- Rezultatų eksportavimo galimybė.

Sąrašas yra sudarytas pagal preliminarinius prioritetus, į pirmąją vietą darbų sąrašė iškeliant autentifikacijos fakulteto nariams galimybę. Nors visos sistemos veikimo principas ir nepasikeis, tačiau ši sistema yra skirta fakulteto bendruomenei, todėl autentifikacija turėtų būti tik su fakulteto prisijungimo duomenimis. Visi kiti punktai yra antriniai ir gali būti vykdomi norint patobulinti sistemos veikimą, siekiant pagerinti vartotojo patirtį apklausų ir balsavimo sistemoje fakulteto bendruomenei.

Sistema yra patalpinta Heroku (<http://herokuapp.com>) debesų platformoje, o duomenų bazė patalpinta mLab (<http://www.mlab.com>) specialioje MongoDB duomenų bazės kaip paslaugos (DbaaS) platformoje. Sistema yra pasiekama adresu <http://mif-vote-app.herokuapp.com>.

Literatūros sąrašas

- [Amb11] Scott W. Ambler and Associates. Examining the Agile Manifesto. Ambyssoft Inc, 2011.
- [Gri17] Stephen Grider. Modern React with Redux – React Router + Redux Form, 2017.
- [Str17] Christof Strauch. NoSQL Databases, 2017. <http://www.christof-strauch.de/nosql dbs.pdf>
- [1] React. Oficiali React dokumentacija, <https://reactjs.org/docs>.
- [2] Redux. Oficiali Redux dokumentacija, <https://redux.js.org>.
- [3] Medium, The SQL vs NoSQL Difference: MySQL vs MongoDB, 2017, <http://www.medium.com>.
- [4] Agile Lietuva, <http://www.agile.lt/straipsnis/agile-projektu-valdymas-41>, 2018.
- [5] Lean Lietuva, <http://www.lean.lt/kas-yra-lean/lean/>, 2018.
- [6] Heroku – platforma kaip paslauga (angl. Platform as a Service). <http://herokuapp.com>, 2018.
- [7] mLab – duomenų bazė kaip paslauga (angl. Database as a Service). <http://mlab.com>, 2018.
- [8] Mano Apklausą., <http://www.manoapklausą.lt>, 2018.
- [9] Apklausą, <http://www.apklausą.lt>, 2018.
- [10] Mentimeter, <http://www.mentimeter.com>, <http://www.menti.com>, 2018.
- [11] Google Forms, <http://forms.google.com>, 2018.

Priedai

MIF Balsavimo sistema

Sveiki, labas@rytas.lt

Atsijungti

Scenarijus: Dėstytojas ant lentos užrašo šios svetainės nuorodą bei šešių skaitmenų slaptažodį. Paspaudus mygtuką 'Įvesti privačios apklausos slaptažodį' ir įvedę teisingą slaptažodį, būsite nukreipti į dėstytojo apklausą. Pavyzdinis slaptažodis: 122905

Pastaba: Šiuo metu sistema sukonfigūruota testavimui. Toks funkcionalumas kaip vartotojo IP tikrinimas yra sukurtas, tačiau išjungtas, todėl balsuoti galima daug kartų.

Sveiki atvykę!

Jūs sveikina įvadinis MIF balsavimo sistemos fakulteto bendruomenei puslapis. Galite rinktis bet kurį šiuo metu vykstantį viešą balsavimą iš sąrašo apačioje, arba spausti mygtuką žemiau ir įvesti privačios apklausos slaptažodį. Prisijungti galite paspaudę žalią mygtuką viršuje, tuomet turėsite galimybę sukurti naujas apklausas.

Sukurti naują apklausą

Įvesti privačios apklausos slaptažodį

Viešos apklausos

Mano apklausos

Pasirinkite apklausą iš sąrašo apačioje.

Įvertinkite šių metų studijas skalėje nuo 1 iki 5.
Viešojoje apklausa: Ar manote, jog šiukšlių skandalas sumažino Remigijaus Šimašiaus reitingus?
Diplomų įteikimo data
Koks yra pirmojo masyvo elemento indeksas?
Ar sutinkate atsiskaitymą perkelti viena savaitę ankščiau?

1 priedas. Prisijungusio vartotojo sąsaja. Rodomas įvadinis puslapis, viešų apklausų ir vartotojo sukurtų apklausų sąrašai, kuriuos galima pasiekti paspaudžiant ant atitinkamo skirtuko.

[Grįžti atgal](#)

Sukurkite naują apklausą

Atsakymų blokas

Ištrinti

Ištrinti

[Pridėti atsakymą](#)☐ Generuoti 6 skaitmenų slaptažodį apklausai[Sukurti apklausą](#)[Iš naujo](#)

2 priedas. Autentifikuotas vartotojas, pradiname puslapyje paspaudęs „Sukurti apklausą“, iškviečia apklausos sukūrimo puslapį ir gali sukurti apklausą.

[Grįžti atgal](#)

Įveskite 6 skaitmenų privačios apklausos slaptažodį:

[Pateikti](#)

3 priedas. Apklausos kūrimo metu sugeneravus 6 skaitmenų slaptažodį, apklausa yra pasiekiamo pagrindiniame puslapyje paspaudus „Įvesti privačios apklausos slaptažodį“ mygtuką ir suvedus teisingą jos slaptažodį.

Grįžti atgal

Kopijuoti nuorodą

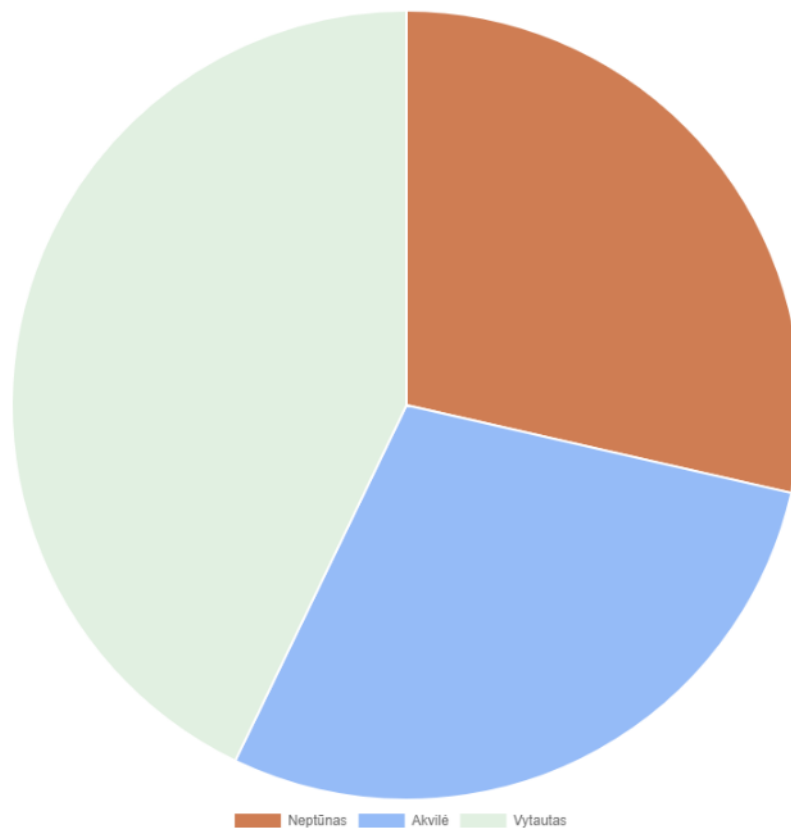
Redaguoti apklausą

Ištrinti apklausą

Mėgstamiausias vanduo

Pasirinkite Jums labiausiai tinkantį atsakymo variantą iš sąrašo apačioje:

Neptūnas
Akvilė
Vytautas



4 priedas. Paspaudus ant bet kurios apklausos sąrašė, arba įvedus teisingą privačios apklausos slaptažodį, vartotojas nukreipiamas į apklausos informacijos puslapį. Šiame puslapyje apklausos kūrėjas turi galimybę ją redaguoti, ištrinti, balsuoti joje. Visi kiti vartotojai gali joje balsuoti.