

Kauno technologijos universitetas

Informatikos fakultetas

**Modulinė „Moodle“ virtualiosios mokymosi aplinkos
naudotojo sąsajos sistema**

Baigiamasis magistro studijų projektas

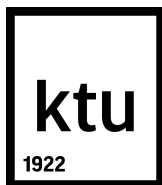
Aistė Čiurlytė Kolesničenko

Projekto autorė

lekt. dr. Ramūnas Kubiliūnas

Vadovas

Kaunas, 2019



Kauno technologijos universitetas

Informatikos fakultetas

**Modulinė „Moodle“ virtualiosios mokymosi aplinkos
naudotojo sąsajos sistema**

Baigiamasis magistro studijų projektas

Informatikos inžinerija (B04)

Aistė Čiurlytė Kolesničenko

Projekto autorė

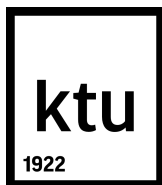
lekt. dr. Ramūnas Kubiliūnas

Vadovas

doc. dr. Danguolė Rutkauskienė

Recenzentė

Kaunas, 2019



Kauno technologijos universitetas

Informatikos fakultetas

Aistė Čiurlytė Kolesničenko

Modulinė „Moodle“ virtualiosios mokymosi aplinkos naudotojo sąsajos sistema

Akademinio sąžiningumo deklaracija

Patvirtinu, kad mano, Aistės Čiurlytės Kolesničenko, baigiamasis projektas tema „Modulinė „Moodle“ virtualiosios mokymosi aplinkos naudotojo sąsajos 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)

Kaunas, 2019

Čiurlytė Kolesničenko, Aistė. Modulinė „Moodle“ virtualiosios mokymosi aplinkos naudotojo sąsajos sistema. Magistro studijų baigiamasis projektas / vadovas lekt. dr. Ramūnas Kubiliūnas; Kauno technologijos universitetas, Informatikos fakultetas.

Studijų kryptis ir sritis: Informatikos mokslai, Informatikos inžinerija (B04).

Reikšminiai žodžiai: naudotojo sąsaja, naudotojo sąsajos pasiekiamumas, dizaino sistema, Moodle.

Kaunas, 2019. 82 p.

Santrauka

Virtualioji mokymosi aplinka nuotoliniame mokymesi tampa pagrindiniu mokymosi proceso dalyvių bendravimo bei mokymosi medžiagos pateikimo kanalu. Mobilųjų technologijų plėtra, įrenginių įvairovė bei skirtingi naudotojų poreikiai VMA naudotojo sąsajos kūrėjams kelia naujų iššūkių. Siekiant suderinti skirtingus naudotojų poreikius, technologines aplinkybes bei pritaikymą naudotojams su negalia, kyla universalios, transformuojamos, modulinės, pasiekiamumo standartus atitinkančios sąsajos poreikis.

VMA „Moodle“ sąsajos kūrimas – sudėtingas procesas. Temos šablono redagavimas ir testavimas „Moodle“ aplinkoje reikalauja daug laiko dėl specifinių sistemos savybių.

Atsižvelgiant į tai, tiriamajame projekte buvo suprojektuota modulinė naudotojo sąsajos dizaino sistema, kurią sudaro blokai ir elementai, jungiami tarpusavyje. Sistemos komponentams pritaikyta pakopinių stiliaus šablonų metodika, nustatanti jų atvaizdavimo taisykles.

VMA naudotojo sąsaja vertinta pagal nustatytus pasiekiamumo kriterijus, kurie suformuluoti atsižvelgiant į Pasaulinio saityno organizacijos pateikiamas padalinio WCAG gaires. Projekto eigoje atliktas „Moodle“ pagrindinės „Boost“ temos naudotojo sąsajos komponentų pasiekiamumo testavimas, identifikuojami pasiekiamumo trūkumai bei pateikiamas patobulintas komponento kodas. Dizaino sistemos įgyvendinimui pasitelkta „Fractal“ priemonė. Patobulinti ir „Fractal“ priemonėje išbandyti komponentai perkelti į „Boost“ paveldinčiąją temą, tokiu būdu pagerinant „Moodle“ naudotojo sąsajos pasiekiamumą.

Dizaino sistemos pagalba patobulintas „Moodle“ naudotojo sąsajos kūrimo, testavimo bei atnaujinimo procesas, leidžiantis kurti, atnaujinti, išbandyti pasiekiamumą bei perkelti į „Moodle“ temos šabloną komponento kodą.

Čiurlytė Kolesničenko, Aistė. A modular system of the user interface of Moodle virtual learning environment. Master's Final Degree Project / supervisor lect. dr. Ramūnas Kubiliūnas; The Faculty of Informatics, Kaunas University of Technology.

Study field and area: Computing, Informatics engineering (B04).

Keywords: user interface, accessibility of the user interface, design system, Moodle.

Kaunas, 2019. 82.

Summary

A virtual learning environment has become the main learners' communication and content learning delivery channel in distance education. Evolution of mobile technologies, diversity of devices, used for accessing the web, and different users' needs bring new challenges for the developers of the user interface of a virtual learning environment. With different users' needs and technological possibilities comes a necessity of universal, transformable, modular and accessible user interface.

Creation of the user interface for virtual learning environment „Moodle“ is a complex process. The editing and testing of theme template in „Moodle“ environment is time consuming and it is determined by the system's architectural features.

Considered that, a modular user interface design system containing blocks and elements, that can be joined together, were planned for the project. A CSS methodology that determines a rules of display of the components was used.

The evaluation of the virtual learning environment was made according to W3C WCAG accessibility guidelines. The evaluation of the accessibility of „Moodle's“ main theme „Boost“ user's interface components was made. New improved code was introduced for the components where accessibility fails were detected. For the creation and management of the design system a „Fractal“ tool was adopted. New components with improved accessibility were transferred from „Fractal“ to the „Boost“ child theme. By adopting the design system the process of the creation, testing and updating of the „Moodle“ user interface was improved.

Turinys

Lentelių sąrašas	9
Paveikslų sąrašas	10
Santrumpų ir terminų sąrašas	12
Įvadas	13
1. Virtualiosios mokymosi aplinkos samprata ir naudojimas	15
1.1. Virtualusis mokymasis ir virtualiosios mokymosi aplinkos samprata	15
1.2. Virtualiosios mokymosi sistemos naudojamumo reikšmė	15
1.2.1. Naudojamumas ir naudotojo sąsaja	15
1.2.2. Naudotojo sąsajos testavimas ir įvertinimas	16
1.3. Mokymosi aplinkos naudotojo sąsajos kūrimo problematika	16
1.4. Mobilųjų technologijų plėtra	17
1.4.1. Mobilųjų įrenginių naudojimas VMA	17
1.4.2. Naudojamų įrenginių įvairovė	19
1.4.3. Prielaidos dizaino sistemų atsiradimui ir jų naudojimo pavyzdžiai	20
1.5. Naudotojo sąsajos pasiekiamumas	21
1.5.1. Pagalbinė sąveikos su naudotojo sąsaja programinė įranga	23
1.5.2. Universalus sąsajos dizainas	23
1.6. Besimokančiųjų VMA dalyvių poreikių analizė	24
1.6.1. VMA dalyvių vaidmenys	24
1.6.2. Besimokančiųjų VMA dalyvių poreikiai	24
1.6.3. Besimokančiųjų dalyvių funkcijų panaudojimo atvejų modelis	25
1.7. „Moodle“ aplinkos temos architektūra	28
1.8. Pakopinių stilių šablonų metodikos grafinėje naudotojo sąsajoje	29
1.8.1. Pakopinių stilių rekomendacijos „Moodle“ temai	29
1.8.2. Block Element Modifier (BEM) metodika	30
1.8.3. Scalable modular architecture CSS (SMACSS)	31
1.8.4. Atominio dizaino architektūra	32
1.8.5. CSS metodikų apžvalga ir palyginimas	32

1.9. Sass preprocesorius	33
1.10. „Bootstrap“ karkasas	33
1.11. CSS stilių pateikimo ateities perspektyvos	34
1.12. Išvados	34
2. „Moodle“ naudotojo sąsajos vertinimo kriterijai	36
2.1. HTML kodo semantika	36
2.2. Spalvos kontrastas ir tekstas	37
2.3. Navigacija	38
2.4. Prisitaikantis dizainas	38
2.5. Išvados	39
3. „Moodle“ aplinkos naudotojo grafinės sąsajos dizaino sistema	40
3.1. Dizaino sistemos kūrimas	40
3.2. „Moodle“ temos dizaino sistemos projektavimas	40
3.3. Dizaino sistemos kūrimo priemonės	43
3.4. Dizaino sistemos įrankio „Fractal“ diegimas	44
3.5. Dizaino sistemos kūrimas „Fractal“ priemonėje	45
3.5.1. „Moodle“ naudotojo sąsajos komponentų kūrimas	48
3.6. Navigacijos komponentai	48
3.6.2. Naršymas klaviatūra ir fokusavimas	49
3.6.3. Šoninė navigacija	51
3.6.4. Navigacijos seka	51
3.6.5. Greitosios nuorodos	53
3.7. Tekstas ir spalvų kontrastas	53
3.8. Semantinė puslapio HTML struktūra	55
3.9. Prisitaikančio dizaino realizavimas	55
3.10. Dizaino sistemos elementų perkėlimas į „Moodle“ temos šabloną	56
3.11. Temos atnaujinimas kartu su „Moodle“ sistemos atnaujinimu	57
3.12. Išvados	57
Išvados	59
Literatūra	60

Priedai	63
1. priedas. KTU NMIT kurso studentų apklausa	63
2. priedas. VMA kurso puslapio HTML komponentų požymių diagrama	66
3. priedas. Straipsnis „Application of modular design system in virtual learning environment“ ALTA'18 konferencijai	67
4. Priedas. Šoninės navigacijos testavimas su „Voice Over“ balso asistentu	73
5. priedas. Komponento atvaizdavimas „Fractal“ priemonės aplinkoje	74
6. priedas. „Boost“ temos atitikimo pasiekiamumo gairėms rezultatai	77
7. priedas. Komponento perkėlimas į VMA „Moodle“ temą	79

Lentelių sąrašas

<u>1 lentelė. Specialiųjų poreikių pavyzdžiai pagal jų pastovumą</u>	22
<u>2 lentelė. Besimokančiųjų reikalavimai pagal VMA sistemas</u>	25
<u>3 lentelė. CSS metodikų modulių skaidymo būdai</u>	33
<u>4 lentelė. Semantinio HTML kodo gairės</u>	37
<u>5 lentelė. Spalvos ir teksto pasiekiamumo gairės</u>	37
<u>6 lentelė. Navigacijos pasiekiamumo gairės</u>	38
<u>7 lentelė. Prisitaikančio dizaino pasiekiamumo gairės</u>	38
<u>8 lentelė. Ekranų dydžiai, sugrupuoti pagal raišką</u>	39
<u>9 lentelė. VMA puslapio pagrindinių blokų žymėjimas</u>	41
<u>10 lentelė. Kurso bloką sudarančių blokų ir elementų žymėjimas</u>	41
<u>11 lentelė. Navigacijos bloką sudarančių blokų ir elementų žymėjimas</u>	41
<u>12 lentelė. Įrankių dizaino sistemai kurti palyginimas</u>	44
<u>13 lentelė. Komponentų kūrimo kontekstinio grafo mazgų ir veiksmų detalizacija</u>	48

Paveikslų sąrašas

<u>1 pav. Mokymosi aplinkos naudotojo sąsajos kūrimo problemų medis</u>	16
<u>2 pav. Mobiliuoju įrenginiu atliekamos „Moodle“ veiklos</u>	18
<u>3 pav. Pasyvių ir aktyvių veiklų atlikimo įpročiai mobiliajame įrenginyje</u>	18
<u>4 pav. Ekranų, naudojamų naršymui internete, raiškos statistika. Lietuva, 2018 m. spalio mėn.</u>	19
<u>5 pav. Įrenginiai, naudoti naršymui. Lietuva, 2018 m. spalio mėn.</u>	20
<u>6 pav. Su besimokančiais susijusių kurso valdymo veiklų panaudojimo atveju diagrama</u>	26
<u>7 pav. Su besimokančiais susijusių mokymosi turinio pateikimo veiklų panaudojimo atveju diagrama</u>	26
<u>8 pav. Su besimokančiais susijusių vertinimo veiklų panaudojimo atveju diagrama</u>	27
<u>9 pav. Su besimokančių bendravimu susijusių veiklų panaudojimo atveju diagrama</u>	27
<u>10 pav. BEM metodo blokų struktūra HTML puslapyje</u>	30
<u>11 pav. SMACSS metodo kategorijos</u>	31
<u>12 pav. Atominio dizaino koncepcijos modulių kategorijos</u>	32
<u>13 pav. VMA kurso puslapio skaidymas blokais</u>	40
<u>14 pav. Kurso bloko skaidymas blokais ir elementais</u>	41
<u>15 pav. Navigacijos bloko skaidymas blokais ir elementais</u>	42
<u>16 pav. Dizaino sistemos įrankio pradžios langas</u>	45
<u>17 pav. Bylos „./src/assets“ struktūra</u>	46
<u>18 pav. Bylos „./src/components“ struktūra</u>	46
<u>19 pav. Komponento išteklių (SCSS kodo) atvaizdavimas dizaino sistemoje</u>	46
<u>20 pav. Komponento puslapio šablonas</u>	47
<u>21 pav. „Gulp“ eksporto proceso konfigūravimo funkcija</u>	47
<u>22 pav. „Gulp build“ eksporto procesas</u>	47
<u>23 pav. Komponentų kūrimo kontekstinis grafas</u>	48
<u>24 pav. „Boost“ temos interaktyvių elementų fokusavimas</u>	50
<u>25 pav. Dizaino sistemos interaktyvių elementų vieningas fokusavimo stilius</u>	50
<u>26 pav. Fokusavimo deklaracija naudojant universalųjį identifikatorių</u>	50

<u>27 pav. Fokusavimo judėjimo tvarka „Boost“ viršutiniame meniu</u>	52
<u>28 pav. „Boost“ temos teksto, nuorodų ir fono kontrasto testas</u>	54
<u>29 pav. Pakoreguoto teksto, nuorodų ir fono kontrasto testas</u>	54

Santrumpų ir terminų sąrašas

Santrumpos:

VMA – virtualioji mokymosi sistema;

MVS – mokymosi valdymo sistema;

HTML – hiperteksto žymėjimo kalba (angl. *Hyper text markup language*);

CSS – pakopiniai stilių šablonai (angl. *Cascading style sheets*);

W3C – Pasaulinio saityno konsorciumas (angl. *The World Wide Web Consortium*);

WCAG – Saityno turinio pasiekiamumo grupė (angl. *Web Content Accessibility Group*);

Terminai:

Naudotojo sąsaja – (angl. *user interface*) priemonių visuma, skirta sistemos valdymui.

Naudojamumas – (angl. *usability*) naudotojo sąsajos kokybės įvertinimas pagal tai, kaip lengva ja naudotis.

Pasiekiamumas – (angl. *accessibility*) sistemos dizainas, pritaikytas specialiųjų poreikių turintiems asmenims.

Universali naudotojo sąsaja – naudotojo sąsaja, pritaikyta visiems naudotojams, nepriklausomai nuo jų negalios ar poreikių.

Dizaino sistema – sistema, apjungianti naudotojo sąsajos komponentų biblioteką, stiliaus gaires bei dokumentaciją.

VMA tema – VMA papildinio tipas, nustatantis naudotojo sąsajos atvaizdavimo taisykles.

Paveldinčioji VMA tema – VMA tema, paveldinti nurodytos pagrindinės temos naudotojo sąsajos atvaizdavimo taisykles.

Prisitaikantis dizainas – naudotojo sąsajos dizainas, dinamiškai prisitaikantis prie naudotojo įrenginio ekrano dydžio bei orientacijos.

Išvadas

Temos aktualumas. Virtualiajame nuotoliniame mokymesi susitikimus auditorijoje pakeičia virtualiosios mokymosi sistemos. Tai tampa vienu iš pagrindinių dėstytojo ir besimokančiųjų bendravimo kanalu. Tradiciniame mokymesi virtualioji mokymosi aplinka taip pat atlieka mokymosi medžiagos talpinimo, tarpusavio bendravimo, administracines, informacinę ir kitas funkcijas. O nuo naudotojo sąsajos patogumo priklauso besimokančiojo įsitraukimo lygis, informacijos randamumas bei naudojimosi sistema efektyvumas.

Atitinkamai mobiliųjų įrenginių paplitimas, prisitaikančio dizaino atsiradimas bei naudotojo sąveikos su programine įranga būdų įvairovė skatina programinės įrangos kūrėjus keisti požiūrį į naudotojo ir sistemos tarpusavio sąveiką iš „naudotojas išmoksta naudotis sistema“ į „sistema prisitaiko (pritaikoma) naudotojo poreikiams“. Programinės ir techninės įrangos dažni atnaujinimai bei su tuo susiję naudotojo sąsajos pakeitimai sąlygoja naudotojo pasimetimą, veiklos efektyvumo sumažėjimą. VMA besimokantysis naudotojas tokiu atveju gali nerasti pateiktos medžiagos, aktualios informacijos ar net praleisti atsiskaitymo terminus, o dalyko mokytojas ar dėstytojas gaišta laiką instruodamas naudotojus apie sistemos naudotojo sąsajos ypatybes.

Plati naudotojų įrenginių įvairovė lėmė prisitaikančio dizaino atsiradimą, ir tai atnešė naujų iššūkių žiniatinklio sistemų naudotojo sąsajos projektuotojams ir vystytojams: nebeužtenka teisiog sukurti fiksuoto dydžio interneto puslapių rinkinį. Naudotojai naudoja įvairius įrenginius, skirtingose situacijose, kurios priklauso ir nuo įrenginio tipo, bet ir nuo naudotojo individualių savybių, skirtingai sąveikauja su sistemos sąsaja.

Be to, Europos Sąjungos lygiu patvirtintos taisyklės, reglamentuoja viešojo sektoriaus institucijų interneto svetainių ir mobiliųjų programų prieinamumą. Juose pabrėžiama universalus, pritaikyto visiems naudotojams, nepriklausomai nuo jų fizinių ar kognityvinių galimybių, sąsajos dizaino svarba.

Darbo tikslas – sukurti moduliarią grafinės naudotojo sąsajos dizaino sistemą, kuri padėtų kurti ir prižiūrėti virtualiajai mokymosi aplinkai „Moodle“ skirtą temą, tenkinančią aplinkos dalyvių funkcinius ir nefunkcinius reikalavimus.

Darbo uždaviniai:

1. ištirti virtualiosios mokymosi aplinkos sąsajos naudojimo ypatumus bei jai keliamus reikalavimus;
2. išanalizuoti besimokančiųjų funkcinius ir nefunkcinius poreikius virtualiojoje mokymosi aplinkoje;
3. išnagrinėti virtualiosios mokymosi aplinkos Moodle naudotojo sąsajos tobulinimo galimybes, būdus ir techninius reikalavimus;
4. sukurti moduliarią dizaino sistemą, padėsiančią gerinti virtualiosios mokymosi aplinkos naudotojo sąsajos pasiekiamumą;

5. įvertinti virtualiosios mokymosi aplinkos „Moodle“ naudotojo sąsajos pasiekiamumo parametrus bei galimybes ir būdus juos pagerinti;
6. įvertinti dizaino sistemos pritaikymo galimybes virtualiosios mokymosi aplinkos „Moodle“ naudotojo sąsajai.

Darbo objektas: mokymosi valdymo sistemos „Moodle“ naudotojo sąsajos kūrimo būdai bei priemonės.

Darbo rezultatas: virtualiosios mokymosi aplinkos „Moodle“ moduliari dizaino sistema, skirta kurti naudotojo sąsajos temą, kuri atitiktų besimokančiųjų sistemos dalyvių nefunkcinius reikalavimus.

Darbo struktūra. Darbą sudaro įvadas, analitinė dalis, projektinė bei realizacijos ir eksperimentinė dalys, išvados. Pirmojoje dalyje tiriama mobiliųjų technologijų plėtros įtaka naudojimuisi virtualiosiomis mokymosi aplinkomis, VMA besimokančiųjų dalyvių poreikiai, naudotojo sąsajos pasiekiamumo reikalavimai bei modulinės sąsajos dizaino įgyvendinimo būdai. Antrajame skyriuje nustatomi pasiekiamumo gairės ir reikalavimai kuriamai sistemai. Trečioji dalis, atsižvelgiant į projekto specifiką, apjungia projektinę bei realizacijos ir eksperimentinę dalis. Sukuriamas dizaino sistemos projektas: puslapio struktūra išskaidoma į blokus bei elementus, kuriems pritaikoma CSS metodika. Projektas įgyvendinamas nuosekliai testuojant pasirinktos bazinės „Moodle“ temos komponentų atitikimą antrajame skyriuje nustatytiems pasiekiamumo reikalavimams bei pateikiant patobulintą komponento struktūrą. Skyrius užbaigiamas dizaino sistemos komponentų perkėlimo į „Moodle“ temos struktūrą proceso aprašymu. Kiekviena dalis apibendrinama skyriaus išvadomis. Darbe pateikiami 29 paveikslai bei 13 lentelių, iliustruojančių ir papildančių tiriamąjį projektą.

Darbo aprobacija. Darbo tyrimo tematika pristatyta tarptautinėje konferencijoje „ALTA'18“ „Pažangios mokymosi technologijos ir aplikacijos. Žaidybinimas švietime“. Parengtas straipsnis anglų kalba „Application of modular design system in virtual learning environment“ (liet. „Modulinės dizaino sistemos taikymas virtualiojoje mokymosi aplinkoje“) paskelbtas konferencijos leidinyje.

1. Virtualiosios mokymosi aplinkos samprata ir naudojimas

1.1. Virtualusis mokymasis ir virtualiosios mokymosi aplinkos samprata

Virtualusis mokymasis – tai virtualioje erdvėje vystantis mokymosi procesas, kurio būtina sąlyga yra internetas ir kompiuteris. [1]

Masinių atvirųjų internetinių kursų (angl. *MOOC*) atsiradimas ir sparti plėtra prisidėjo prie susidomėjimo internetinių technologijų suteikiamomis mokymosi galimybėmis bei virtualaus mokymosi populiarinimo. Pradėtos plėtoti virtualiosios mokymosi aplinkos bei programinė įranga, skirta jas kurti ir naudoti.

Virtualioji mokymosi aplinka (VMA) – tai informacinių ir komunikacinių technologijų priemonėmis pagrįsta sistema, skirta mokymosi medžiagos pateikimui, mokymosi organizavimui, besimokančiųjų bendravimui ir bendradarbiavimui [1, 2]. VMA realizuojama pasiteliant pagal dalyvių poreikius pritaikytas programines priemones – mokymosi valdymo sistemas (MVS). VMA sudaro galimybę mokytis individualiu tempu, patogiu laiku ir nepriklausomai nuo vietos.

1.2. Virtualiosios mokymosi sistemos naudojamumo reikšmė

Mokymosi įstaigoms įsidiegiant virtualiąsias mokymosi sistemas ir į jas perkeltiant daugiau veiklų bei funkcijų, kinta mokytojo ar dėstytojo ir besimokančiųjų bendravimo vieta ir būdas. Virtualioji mokymosi aplinka atlieka mokymosi medžiagos talpyklos, žinių patikrinimo, tarpusavio bendravimo ir kitas funkcijas. Mokymosi aplinkos ir besimokančiojo tarpusavio sąveika vyksta naudotojo sąsajos pagalba. Netinkamai organizuota, neintuityvi, neatitinkanti pasiekiamumo standartų naudotojo sąsaja gali lemti studento motyvacijos ir išitraukimo į mokymosi procesą sumažėjimą. Virtualiosiomis mokymosi sistemomis naudojamas mokantis tradiciniu būdu ir virtualiai. Mokantis nuotoliniu būdu virtualiai, virtualiosios mokymosi aplinkos reikšmė smarkiai išauga.

1.2.1. Naudojamumas ir naudotojo sąsaja

Naudojamumas (angl. *usability*) gali būti apibrėžiamas kaip naudotojo sąsajos kokybės įvertinimas pagal tai, ar ja yra lengva naudotis, ar ji yra intuityvi, savaime suprantama.

Naudojamumas yra apibūdinamas šiais kokybės komponentais [3]:

- **Išmokstamumas.** Ar naudotojams, pirmą kartą besinaudojantiems sąsaja, lengva atlikti pagrindines užduotis?
- **Efektyvumas.** Kaip naudotojas, susipažinęs su sąsajos dizainu, lengvai gali atlikti užduotis?
- **Įsimenamumas.** Kaip greitai naudotojas, kurį laiką nesinaudojęs konkrečia sąsaja, gali prisiminti naudojimosi eigą?
- **Klaidos.** Kiek klaidų naudotojas daro? Kaip dažnai, ir kaip lengvai jis gali šias klaidas ištaisyti?
- **Pasitenkinimas.** Ar naudotojo sąsają yra patraukli naudoti?

1.2.2. Naudotojo sąsajos testavimas ir įvertinimas

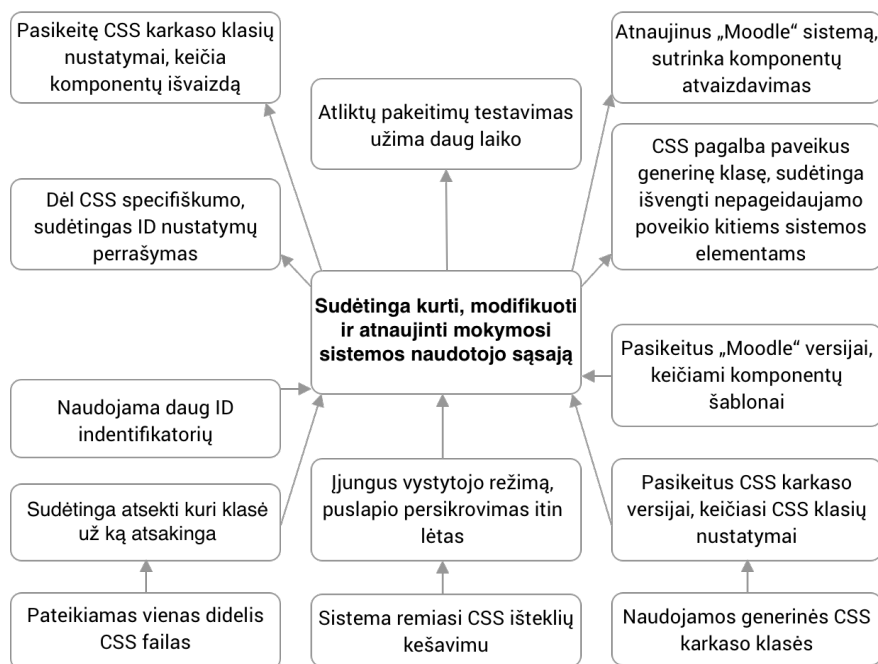
Naudotojo sąsajos įvertinimui dažniausiai pasitelkiami asmenys, atitinkantys tikslinės auditorijos aprašymą. Gali būti testuojamas sąsajos prototipas, modulis, dar neįdiegta sąsaja, veikianti esama sąsaja, planuojami sąsajos pakeitimai. Naudojamos testavimo technikos [4]:

- interviu: interviu (pokalbis) su naudotoju po užduoties atlikimo;
- klausimynai: Naudotojas atsako į pateiktus klausimus apie jo naudojimosi įpročius;
- „mąstymas garsiai“: naudotojas, atlikdamas užduotis, garsiai įvardija tai, ką jis galvoja;
- tiesioginis stebėjimas: stebėtojas stebi naudotoją, atliekantį užduotis;
- įrašytas stebėjimas: naudotojo elgsena fiksuojama vaizdo kameros pagalba;
- ekrano įrašas: fiksuojama naudotojo elgsena ekrano įrašymo pagalba;
- naudotojų elgsenos duomenų rinkimas: automatizuotai renkami sistemos analitiniai duomenys apie naudotojų lankymosi trukmę, paspaudimus;
- fokus grupė: moderuojama diskusija tarp potencialių naudotojų;
- akių judesių sekimas (angl. *eye tracking*): akių judėjimo ekranu fiksavimas specialios įrangos pagalba.

Kaip atskirą testavimo sritį derėtų išskirti naudotojo sąsajos pasiekiamumo vertinimą. Tai svarbus etapas siekiant užtikrinti sąsajos prieinamumą visiems naudotojams nepriklausomai nuo jo negalios, fizinių savybių, situacijos ar naudojamo įrenginio. Šiai testavimo sričiai yra sukurti oficialūs Pasaulinio žiniatinklio konsorciumo kriterijai, pagal kuriuos galima įvertinti sąsajos komponentų pasiekiamumą.

1.3. Mokymosi aplinkos naudotojo sąsajos kūrimo problematika

Norėdami kurti, modifikuoti ir atnaujinti VMA naudotojo sąsają, vystytojai susiduria su sunkumais (1 pav.).



1 pav. Mokymosi aplinkos naudotojo sąsajos kūrimo problemų medis

VMA „Moodle“ naudotojo sąsajos išvaizdos nustatymai yra pateikiami papildinių - temų pagalba. Kartu su sistema, instaliuojamos standartinės temos, kurios gali būti naudojamos kaip pagrindas naujos sąsajos kūrimui ar modifikavimui. „Moodle“ sistema naudoja CSS išteklių kešavimą, kas leidžia pagerinti sistemos puslapių veikimo naršyklėje spartą, tačiau norint keisti bei išbandyti atliktus pakeitimus sistemos išvaizdoje, turi būti įjungtas temos vystytojo režimas (angl. *theme designers mode*). Tai itin sulėtina puslapio atsisiuntimo iš naujo spartą bei apsunkina naudotojo sąsajos kūrimo procesą.

Standartinės temos naudoja CSS karkasus bei labai plačiai taiko karkasų klases HTML kode. Tai padeda užtikrinti patenkinamą sąsajos išvaizdą ir elementų išdėstymą prisitaikančiame dizaine. Tačiau, karkaso atnaujinimas reikalauja atnaujinimo temos kode. Taip pat, temos kode dažnai naudojamas ID identifikatorius, kurį perrašyti, panaudojant CSS kalbos pakopiškumą yra gan sudėtinga. Todėl, „Moodle“ naudotojo sąsajos kūrimas tampa gana sudėtingu bei laiko sąnaudų reikalaujančiu procesu.

1.4. Mobilųjų technologijų plėtra

2017 metų pabaigoje pasaulyje mobiliųjų įrenginių naudojimas prieigai prie interneto sudarė 52,48% visų prisijungimų. 2016 metų pabaigoje prieiga prie interneto per mobiliuosius įrenginius viršijo prieigą per asmeninius kompiuterius, kuri 2017 gruodžio mėnesį sudarė 43,26% visų prisijungimų prie interneto [5].

2015 metais atliktas mobiliųjų įrenginių ir asmeninių kompiuterių naudojimo Virtualiojoje mokymosi aplinkoje tyrimas [6] tarp Liublijanos verslo mokyklos studentų (tradicinių ir nuotolinių studijų modulių). Mobilųjų įrenginių studentai naudotų daugiausiai: tvarkaraščių paieškai, sutikrinimui (48%), pažymiams, įvertinimams patikrinti (43%). Asmeninių kompiuterių daugiausiai naudotų elektroninio mokymosi turinio parsisiuntimui (83%) bei interaktyvioms mokymosi veikloms (83%).

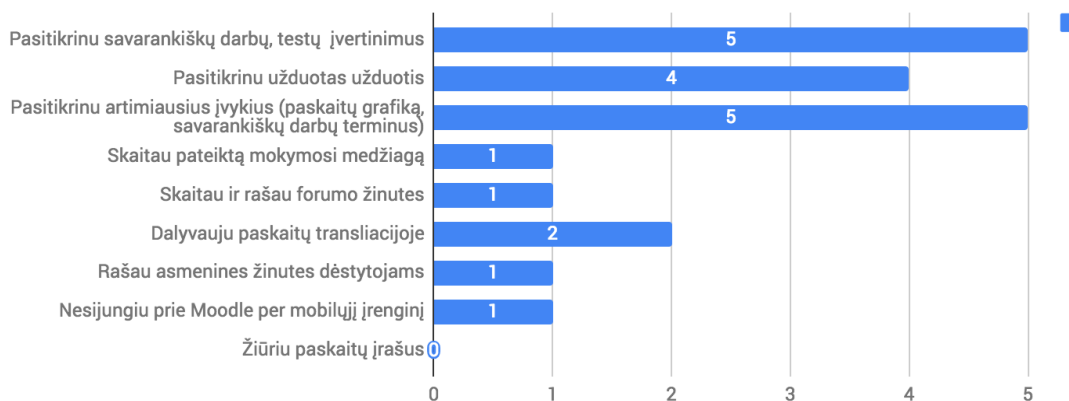
1.4.1. Mobilųjų įrenginių naudojimas VMA

2017 metais atliktas tyrimas apie mobiliųjų įrenginių naudojimą virtualiojoje mokymosi aplinkoje „Moodle“ atskleidė, kad mobilajam telefonui nebuvo teikiama pirmenybė jungiantis prie virtualiosios mokymosi aplinkos [7]. Nepaisant to, kad išmaniųjų telefonų techninės savybės, tokios kaip ekrano raiška ir įrenginio sparta, nuolat tobulėja, o prisijungimas per mobilųjį interneto ryšį pinga, išmanusis įrenginys studentų nurodomas tik kaip pagalbiniė prisijungimo prie virtualiosios mokymosi aplinkos priemonė, skirta atlikti nesudėtingoms ir daug laiko neužimančioms užduotims [7].

2011 metais atlikta Vienos ekonomikos ir verslo universiteto Virtualios mokymosi aplinkos „Learn@WU“ naršymo statistikos analizė [8] parodė, kad vidutinis naudotojo užklausų skaičius mobilajame įrenginyje buvo 10,47, kai tuo tarpu asmeniniame kompiuteryje – 44,09. Tai reiškia, mobiliųjų įrenginių naudotojai naudoja VMA efektyviau ir tikslingiau [8]. Remiantis minėtu tyrimu, galima teigti, kad mobiliųjų įrenginių naudotojai buvo linkę labiau susitelkti į konkrečią informaciją (egzaminų rezultatai, kalendorius, paskaitų kambarys ir pan.). Palyginti kompiuterių naudotojai labiau koncentravosi ties modulių informacija ir mokymosi veiklomis (forumo veikla, mokymosi medžiagos parsisiuntimas, užduotys, egzaminai ir pan.) [8].

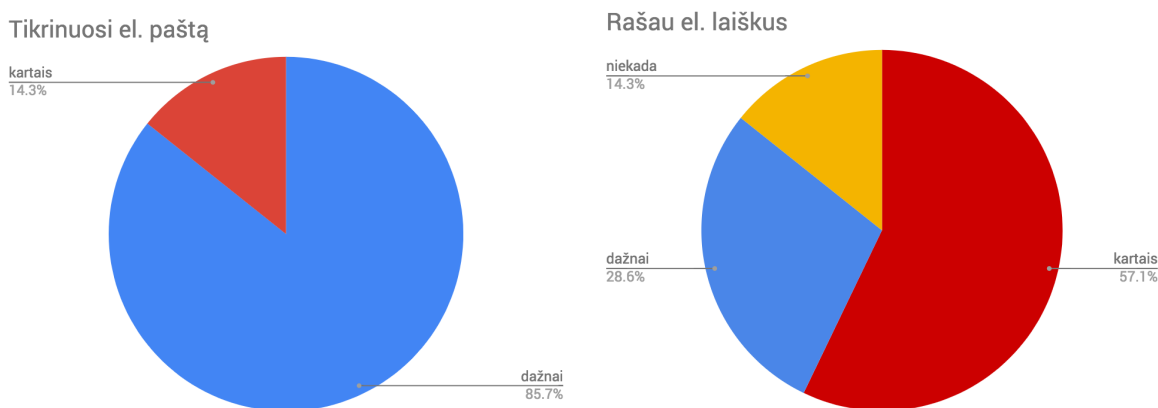
2018 metais apklausti Kauno technologijos universiteto Nuotolinio mokymosi informacinių technologijų studijų programos I kurso magistrantai, (1 priedas) patvirtino anksčiau pateiktas prielaidas apie mobiliųjų įrenginių naudojimo įpročius kurso virtualiojoje mokymosi aplinkoje. 71,4% apklaustų dalyvių prisijungę prie „Moodle“ aplinkos mobiliuoju įrenginiu pasitikrina darbų įvertinimus bei peržiūri artimiausius kurso įvykius (2 pav.).

Kokias veiklas atliekate prisijungę prie Moodle mobiliajame įrenginyje?



2 pav. Mobiliuoju įrenginiu atliekamos „Moodle“ veiklos

Apklausos metu buvo teirautasi apie mobiliojo įrenginio naudojimo įpročius: dominuojančios mobiliajame įrenginyje veiklos yra nereikalaujančios koncentracijos ir sudėtingų atsakomųjų veiksmų. Tai iliustruoja e. pašto naudojimo pavyzdys: didžioji dalis apklaustųjų tikrinosi e. paštą telefone, tačiau mažiau nei pusė dažnai rašė ar atsakinėjo į e. laiškus mobiliajame įrenginyje. (3 pav.)

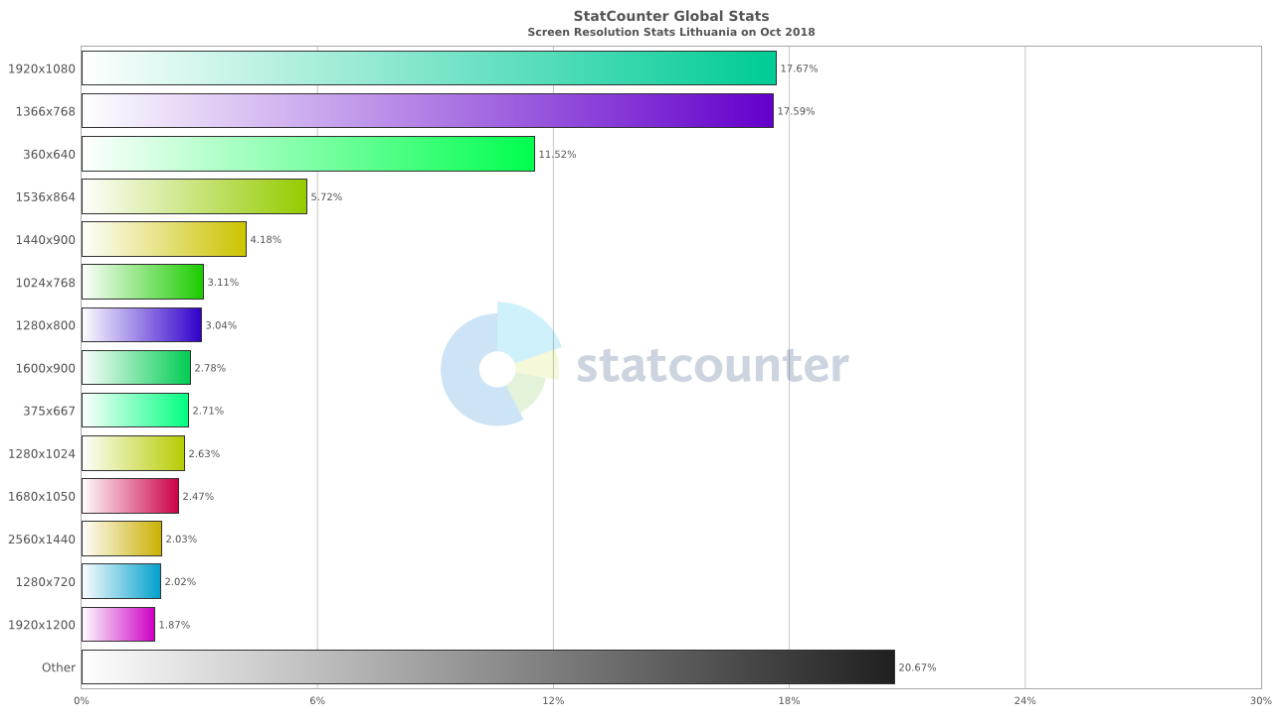


3 pav. Pasyvių ir aktyvių veiklų atlikimo įpročiai mobiliajame įrenginyje

Apibendrinant galima teigti, kad pritaikant sistemos naudotojo sąsają mobiliesiems įrenginiams, neužtenka vien pritaikyti puslapių išvaizdą ekrano dydžiui. Atsižvelgiant į pasikeitusias veiklas, naudotojams informaciją reikia pateikti remiantis jų prioritetais, atitinkančiais naršymo įpročių skirtumais.

1.4.2. Naudojamų įrenginių įvairovė

Kartu su mobiliųjų technologijų plėtra bei su tuo susijusiu prisitaikančio dizaino atsiradimu, atsirado naujų grafinės naudotojo sąsajos projektavimo bei kūrimo iššūkių. Naudotojų iš Lietuvos ekranų raiškos statistiniai duomenys, surinkti „Statcounter“ (4 pav.) [9], rodo platų naudotojų galinių įrenginių spektrą. Iš pateiktų duomenų galima manyti, kad mažesnio dydžio bei portretinio formato ekranai (360x640) yra išmanieji telefonai. Kiti įrenginiai, kurių ekrano raiška varijuoja nuo itin didelių 2560x1440 iki 1024x768 pikselių gali reikšti tradicinius ar planšetinius kompiuterius. Didžiausia dalis (20,67%) nepapuoia į nė vieną kategoriją, grafike nurodyta kaip „Kita“.

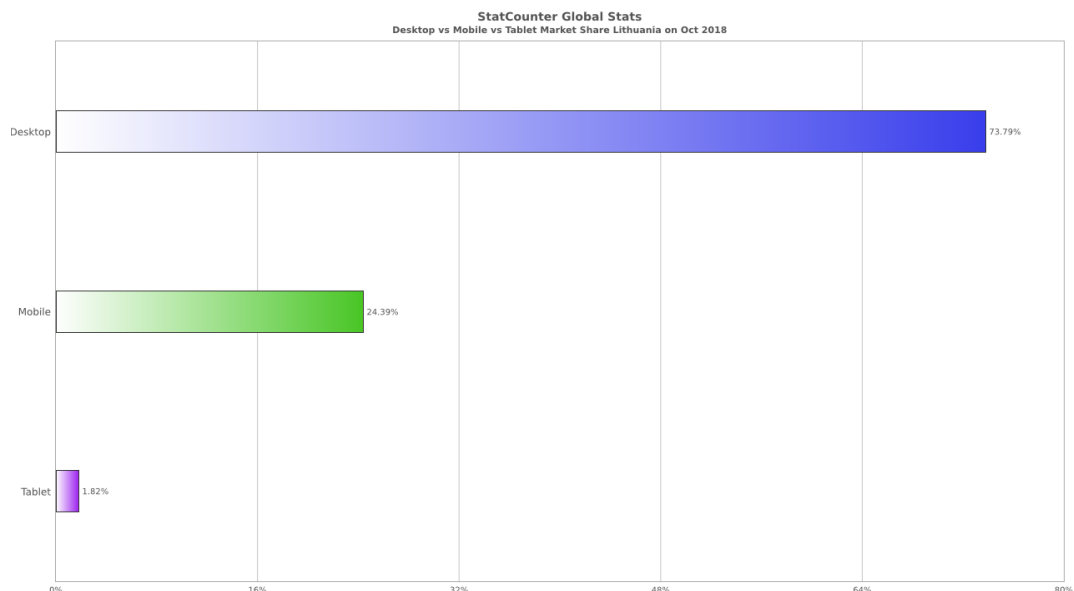


4 pav. Ekranų, naudojamų naršymui internete, raiškos statistika. Lietuva, 2018 m. spalio mėn.

Naudotojai skirtingai sąveikauja su sistema: priklausomai nuo jų fizinių, kognityvinių bei technologinių galimybių ir esamos situacijos jie naudojami pele, jutikliniu kilimėliu, klaviatūra ar kitu įvesties įrenginiu. Naudojantis išmaniuoju įrenginiu, naudotojų veiksmai dažniausiai atliekami liečiant ekraną. Derėtų paminėti, kad nėra griežtų ribų tarp naudotojų sąveikos būdų: kompiuteriai gali turėti lietimui jautrų ekraną, išmanieji įrenginiai gali būti naudojami juos liečiant įvesties priemone (pieštuku). Lietuvos saityno naudotojų statistika [5] (5 pav.) rodo, kad 2018 m. spalio mėn. duomenimis, 73,79 proc. naudotojų naršė internete kompiuterio pagalba, 24,39 proc. mobiliuoju telefonu ir 1,82 proc. – planšetiniu kompiuteriu.

Remiantis pateiktais duomenimis naudotojų įrenginius galima sugrupuoti pagal šiuos požymius:

1. Pagal dydį (maži, vidutiniai, dideli).
2. Pagal sąveikos būdą (liečiama, pelė, klaviatūra).



5 pav. Įrenginiai, naudoti naršymui. Lietuva, 2018 m. spalio mėn.

Naudojamų įrenginių įvairovė lemia prisitaikančio sąsajos dizaino būtinybę: naudotojo sąsaja prisitaiko ne tik prie naudotojo įrenginio, bet ir turi būti universali sąveikos būdo atžvilgiu.

1.4.3. Prielaidos dizaino sistemų atsiradimui ir jų naudojimo pavyzdžiai

Prisitaikančio dizaino atsiradimas, sąlygotas mobiliųjų įrenginių plėtros bei įvairovės, paskatino naudotojo sąsajos kūrėjus naujai pažvelgti į sąsajos architektūros projektavimo ir kūrimo procesą. Sąsajos komponentų išdėstymas prisitaikančiame dizaine ekrane nėra statiškas ir nekintantis. Prisitaikančiame dizaine sąsajos komponentų pateikimo būdas ir eiliškumas priklauso nuo naudotojo įrenginio dydžio bei orientacijos. Tinklalo išvaizda įgyja daug variantų, dėl to sąsajos dizaino kūrimas vis sudėtingėja. Atsirado būtinybė sukurti tokią sistemą, kuri leistų kurti, tobulinti, keisti, apjungti, dokumentuoti, išbandyti skirtingose aplinkose naudotojo sąsajos komponentus bei jų tarpusavio sąveiką.

M. Godbolt savo knygoje sąsajos architektūros sąvoką apibrėžia kaip įrankių bei procesų rinkinį, skirtą pagerinti sąsajos programiniam kodui bei efektyvesnei darbo eigai [10]. Išskiriama keletas sąsajos dizaino architektūros sąvokų: stiliaus gairės, šablonų biblioteka ir dizaino sistema.

Stiliaus gairės (angl. *style guides*). John Toman savo straipsnyje nurodo, kad stiliaus gairės yra bendros prekės ženklo ar produkto vizualaus pateikimo taisyklės [11]. Sąsajos projektavimui gali būti pritaikytos bendrosios stiliaus gairių taisyklės, taikomos kitiems organizacijos ar produkto pateikimo atvejams (spaudoje, korespondencijoje, atributikoje).

Šablonų biblioteka (angl. *pattern library*). Kitaip dar dažnai vadinama komponentų biblioteka, sudaroma iš suprogramuotų HTML, CSS ir JavaScript komponentų bei dokumentacijos. Kaip savo straipsnyje, palyginančiame aptariamas sąvokas, pabrėžia Jack Davies, skirtingai nuo stiliaus gairių, šablonų biblioteka dažniau taikoma tik naudotojo sąsajai kurti [12]. Šablonų bibliotekos dažnai yra sudaromos iš tokių grafinės naudotojo sąsajos elementų kaip: mygtukai, nuorodos, paveikslai, galerijos, navigacija.

Dizaino sistema (angl. *design system*). Dizaino sistema apjungia stiliaus gaires bei šablonų biblioteką į vieningą naudotojo sąsajos sistemą, kuri apima [12]:

- dizainą;
- sistemos pasiekiamumą;
- grafinės sąsajos architektūrą ir pateikimo principus.

Dizaino sistemos diegiamos įmonėse, vystančiose įvairius internetinius projektus ar platformas. Tai palengvina vystytojų ir skirtingų padalinių bendradarbiavimą, užtikrina vieningą struktūrą, atitikimą pasiekiamumo standartams. Sukurtos sistemos neretai pateikiamos viešai. Dizaino sistemas yra sukūrę bei nuolat atnaujina „AirBnB“, „IBM“, „Lonely Planet“, „BBC“. JAV Bendrųjų paslaugų administracija (angl. *U.S. General Services Administration*) sukūrė atvirą dizaino sistemą (angl. *U.S. Web Design System*), skirtą visoms vyriausybinėms organizacijoms, siekiant užtikrinti pasiekiamumo standartus [13].

Virtualios mokymosi sistemos dizaino sistemą sukūrė ir nuolat tobulina „Future Learn“, „The Open University“ padalinys. „Future Learn“ pateikia pagrindinius savo kuriamos dizaino sistemos tikslus [14]:

- sukurti bendrą sąvokų žodyną, naudojamą šablonams dokumentuoti;
- užtikrinti nuoseklumą tarp šablonų ir grafinės sąsajos komponentų;
- pateikti gaires teisingam sąsajos komponentų naudojimui;
- skatinti kodo pakartotinį panaudojimą, sumažinti bereikalingą perrašymą.

„Future Learn“ sistemos kūrėjai pabrėžia, kad dizaino sistema yra nuolat besikeičiantis dokumentas, kuris yra atnaujinamas keičiant šablonus ar papildant sistemą naujais sąsajos elementais [14].

1.5. Naudotojo sąsajos pasiekiamumas

Virtualiosioms mokymosi aplinkoms pasiekti naudojamos interneto naršyklės (ang. *web browser*), interpretuojančios HTML, CSS ir JavaScript kalbą ir atvaizduojančios turinį vizualiai, naudotojams suprantamu formatu. Tačiau ne visi naudotojai turi vienodas galimybes pasiekti, pamatyti, suprasti pateiktą turinį.

Terminas interneto turinio pasiekiamumas (angl. *web accessibility*) reiškia, kad asmenys, su specialiaisiais poreikiais gali naudotis internetu. Pasaulinio saityno konsorciumo (W3C) tinklalapyje apie interneto turinio pasiekiamumą, detalizuojama, kad anksčiau minėtieji asmenys, gali suvokti, suprasti, naršyti ir bendrauti padedant saitynui. Konsorciumo tinklalapyje pažymima, kad pritaikytu, pasiekiamu turiniu gali naudotis ir asmenys, neturintys negalios, įskaitant ir senstančius žmones, kurių gebėjimai su amžiumi gali kisti [15].

Pasaulinis saityno konsorciumas (W3C) įsteigė visuomeninę organizaciją (*Web Accessibility Initiative, WAI*), kuri rengia gaires interneto pasiekiamumui gerinti. Tai savanoriška organizacija, pateikianti rekomendacijas internetinių svetainių, mobiliųjų programėlių asmenims su specialiaisiais poreikiais pritaikymui.

„WAI“ organizacija, bendradarbiaudama su organizacijomis partnerėmis, siekia gerinti saityno pasiekiamumą [16]:

- pateikia rekomendacijas, kurios pripažintos tarptautiniu saityno pasiekiamumo standartu;
- teikia papildomą medžiagą, padedančią suprasti ir diegti saityno pasiekiamumą, bei bendradarbiauja kuriant pasiekiamumo informacinius išteklius;
- edukuoja ir viešina pasiekiamumo svarbą;
- koordinuoja tyrimus, kurie gali pakeisti ir pagerinti saityno pasiekiamumą;
- dalyvauja pasiekiamumo gerinimo priemonių vystyme ir vertinime.

Pasaulinis saityno konsorciumo gairėse specialieji poreikiai ir sutrikimai klasifikuojami į pagrindines kategorijas [15]:

- regos;
- klausos;
- fizinius, motorinius;
- kalbos;
- kognityvinius;
- neuorologinius.

Specialieji poreikiai gali būti grupuojami pagal jų pastovumą [17]: vieni poreikiai yra nuolatiniai, kiti laikini (1 lentelė). Tai padeda suprasti, kad specialieji poreikiai nebūtinai yra siejami su negalia. Pasitaiko situaciniai, išnykstantys poreikiai, kurie gali iškilti tik tam tikrų situacijų metu ar patyrus traumas. Dažnai su amžiumi susijusių poreikių: senatvinis regos silpnėjimas, motoriniai sutrikimai.

1 lentelė. Specialiųjų poreikių pavyzdžiai pagal jų pastovumą

	Situaciniai	Laikini	Nuolatiniai	Amžiaus
Regos		Smegenų sukrėtimas	Aklumas Daltonizmas	Geltonosios dėmės degeneracija Senatvinė toliaregystė
Motoriniai	Rankose laikomas kūdikis	Lūžusi, patempta ranka Riešo kanalo sindromas	Cerebrinis paralyžius	Parkinsono liga
Klausos	Triukšminga aplinka		Kurtumas	Silpstanti klausa
Kognityviniai		Smegenų sukrėtimas	Disleksija Autizmo spektro sutrikimai	Demencija

WAI išleido „Saityno turinio pasiekiamumo gairių“ 2.0 versiją (angl. *Web Content Accessibility Guidelines 2.0, WCAG*), kuriose pateiktos gairės bei gerosios praktikos, naudotinos kaip priemonė sistemų pasiekiamumui gerinti [18].

Gairės sudarytos atsižvelgiant į keturis principus [18]:

- **Suvokimas:** naudotojai gali suvokti turinį. Tai padeda suprasti, kad jei didelė dauguma naudotojų gali pasinaudoti kažkuriuo jausmu (pvz. rega), tai negali būti laikoma standartu visiems naudotojams.
- **Valdymas:** naudotojai gali naudotis sąsajos elementais bei naršyti. Pavyzdžiui, veiksmai, kurie galimi tik užvedus pelę ant interaktyvaus elemento yra neįmanomi tiems naudotojams, kurie nenaudoja pelės ar lietimui jautraus įrenginio.
- **Aiškumas:** naudotojai gali suprasti turinį. Naudotojai gali suvokti kaip naudotis sąsaja ir ar ji nėra klaidinanti.
- **Universalumas:** sistemos turinys gali būti naudojamas pasitelkiant skirtingas naršyklės. Veikia, naudojantis pagalbinėmis sąveikos su sąsaja technologijomis.

1.5.1. Pagalbinė sąveikos su naudotojo sąsaja programinė įranga

Siekdami pagerinti naudotojo naršymo galimybes, ypač tais atvejais kai neužtenka standartinių naudotojo sąsajos pritaikymo sprendimų, naudotojai gali pasitelkti papildomą programinę ar techninę įrangą, padedančią naršyti internete. Programinė įranga gali būti diegiama ar integruota į naršyklę, operacinę sistemą.

Balso asistentai konvertuoja skaitmeninį tekstą į susintetintą kalbą. Tai leidžia naudotojams girdėti turinį bei naršyti klaviatūros pagalba. Ši technologija padeda akliesiems ar asmenims su sutrikusia rega naudotis technologijomis savarankiškai ir be kitų pagalbos [19]. „WebAim“ organizacija taip pat pabrėžia, kad balso asistentų technologija padeda naudotojams, turintiems tam tikrų kognityvinių ar mokymosi sutrikimų [19].

„**ChromeVox**“. Google Chrome papildinys, įgarsinantis naršomą turinį, skirtas asmenims, turintiems regėjimo negalią [20].

„**Voice Over**“. Į Mac OS bei iOS integruotas įrankis, įgarsinantis naršomą turinį.

Virtualūs asistentai (pvz. „Siri“, „Alexa“). Gali pagelbėti esant situaciniams ar laikiniams specialiesiems poreikiams.

1.5.2. Universalus sąsajos dizainas

Pasaulinio žiniatinklio konsorciumo gairės taip pat apibūdina universalus dizaino (angl. *inclusive design*) idėją, kuri reiškia, kad interneto turinys gali būti pateikiamas universaliu formatu, pritaikytu visiems naudotojams, nepriklausomai nuo jų negalios ar poreikių, kas atitinka Jungtinių Tautų neįgaliųjų teisių konvencijoje pateikiamą apibrėžtį, kad aprėpiantis dizainas – gaminių, aplinkos, programų ir paslaugų, skirtų naudoti visiems žmonėms kuo platesniu mastu, dizainas, kai nėra pritaikymo ar specializuoto dizaino būtinybės [21].

ES lygmeniu yra priimta direktyva dėl viešojo sektoriaus institucijų interneto svetainių ir mobiliųjų programų prieinamumo nustato taisykles, kuriomis vadovaujantis valstybės narės užtikrintų, jog viešojo sektoriaus institucijų interneto svetainės atitiktų nustatytus pasiekiamumo reikalavimus [22].

Pasaulinis žiniatinklio konsorciumas išskiria tris pritaikymo lygius [15]:

- **A lygis:** minimalus pasiekiamumo lygmuo. Tinklalapis atitinka A lygio kriterijus arba yra pateikiama pritaikyta alternatyvi tinklalapio versija.
- **AA lygis:** standartinis pritaikymo lygmuo. Tinklalapis atitinka visus A ir AA lygio kriterijus, arba pateikiama atitinkama AA lygio alternatyvi versija.
- **AAA lygis:** tinklalapis atitinka visus A, AA ir AAA kriterijus, arba pateikiama AAA lygio alternatyvi versija.

Pasaulinio žiniatinklio konsorciumo dokumentacijoje pažymima, kad nėra būtina pritaikyti AAA lygiu visą tinklalapio turinį, kadangi ne visą turinį įmanoma pritaikyti šiam lygmeniui [15]. Pabrėžtina, kad pritaikymas pagal pasiekiamumo reikalavimus įtraukia ne tik sąsajos bei turinio adaptavimą naudotojams, turintiems specialiųjų poreikių (nuolatinių ar laikinųjų), tačiau ir naudotojams, besinaudojantiems skirtinga technine įranga.

1.6. Besimokančiųjų VMA dalyvių poreikių analizė

1.6.1.VMA dalyvių vaidmenys

VMA dalyviai gali atlikti skirtingus vaidmenis sistemoje. Jiems taip pat gali būti priskirtos papildomos teisės ar suteiktos naudojimosi sistema teisės, atsižvelgiant į dalyvio atliekamas funkcijas ir poreikius.

Pagrindiniai ir dažniausiai naudojami „Moodle“ sistemos dalyvių vaidmenys yra:

- administratorius;
- dėstytojas (mokytojas);
- asistentas;
- studentas (besimokantysis, mokinys);
- svečias.

Šiame darbe nagrinėjama naudotojo sąsaja, atsižvelgiant į besimokančiojo poreikius, tačiau dalyvių funkcijos dažnai yra susijusios ir poreikiai neretai sutampa.

1.6.2.Besimokančiųjų VMA dalyvių poreikiai

VMA dalyvių poreikiai skirstomi į funkcinius ir nefunkcinius reikalavimus. Funkciniai poreikiai nurodo ką reikia atlikti, o nefunkciniai reikalavimai apibūdina kaip tai turi būti atlikta. Naudotojo sąsajos atvaizdavimas, patogumas ir pasiekiamumas dažniausiai atspindi sistemos dalyvių nefunkcinius reikalavimus. Nefunkciniai reikalavimai gali skirtis priklausomai nuo sistemos posistemės ar panaudojimo atvejo atliekamos funkcijos (2 lentelė).

Aplinkos dalyviai, atsižvelgiant į jų funkcijas, turi skirtingus poreikius. Dažnai skirtingų dalyvių poreikiai sutampa. Projektuojant sistemą, būtina atsižvelgti į visų jos dalyvių reikalavimus. Projektuojant naudotojo sąsają, pagrindiniai reikalavimai formuluojant kriterijus buvo būtent besimokančiojo poreikiai.

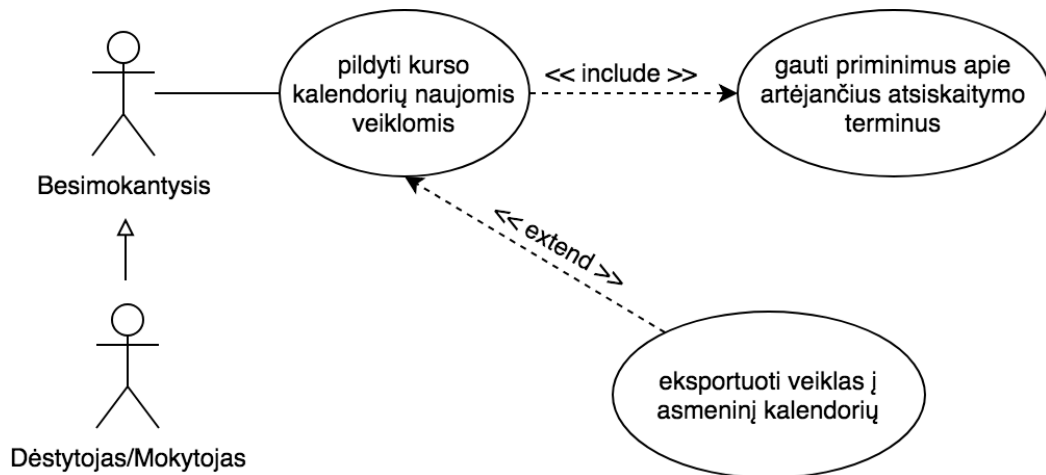
1.6.3. Besimokančiųjų dalyvių funkcijų panaudojimo atvejų modelis

Nagrinėjami besimokančiųjų sistemos dalyvių funkcijų panaudojimo atvejai. Modelis papildomas dėstytojo panaudojimo atvejais tuo atveju, kai besimokančiųjų panaudojimo atvejis yra susijęs su dėstytojo vaidmens dalyviu tarpusavio ryšiais.

2 lentelė. Besimokančiųjų reikalavimai pagal VMA posistemas

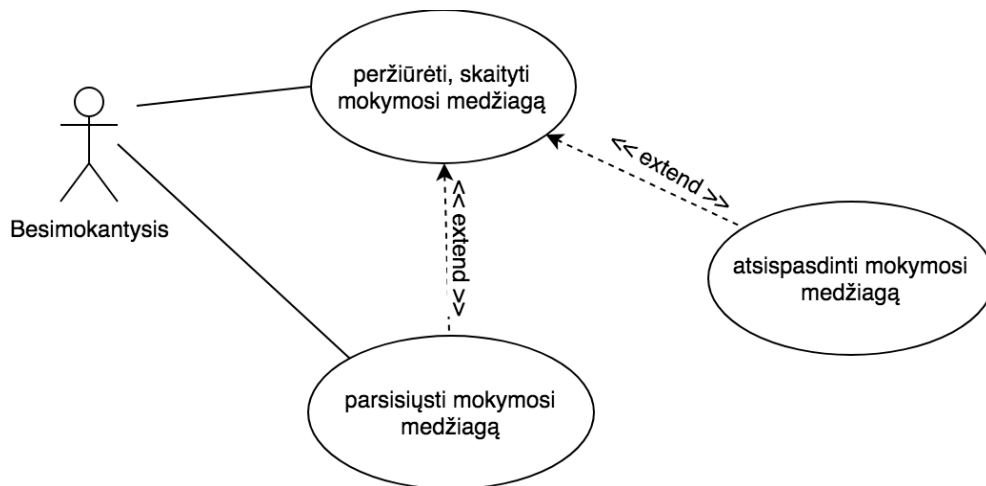
VMA posistemė	Funkciniai reikalavimai	Nefunkciniai reikalavimai
Kurso valdymas	<ul style="list-style-type: none"> ○ pildyti kurso kalendorių veiklomis; ○ eksportuoti kalendoriaus veiklas į asmeninį kalendorių; ○ gauti priminimus apie artėjančius atsiskaitymo terminus. 	<ul style="list-style-type: none"> ○ adaptyvus, pritaikytas naudotojo įrenginiui dizainas; ○ geras informacijos matomumas.
Mokymosi turinio pateikimas	<ul style="list-style-type: none"> ○ peržiūrėti, skaityti mokymosi medžiagą; ○ išsisaugoti mokymosi medžiagą; ○ atsispausdinti mokymosi medžiagą. 	<ul style="list-style-type: none"> ○ pasiekiamas, universalus mokymosi medžiagos formatas; ○ galimybė individualizuoti turinio pateikimą, pvz. pasididinti skaitomo teksto šriftą; ○ patogus ir aiškus mokymosi medžiagos išdėstymas; ○ pritaikyta asmenims su specifiniais poreikiais.
Vertinimas	<ul style="list-style-type: none"> ○ galima atlikti įvertinimo, savitikros veiklas; ○ gauti grįžtamąjį ryšį; ○ stebėti pažangą. 	<ul style="list-style-type: none"> ○ aplinka pritaikyta įvairiems įrenginiams; ○ patogi sąsaja, informacijos randamumas.
Bendravimas ir bendradarbiavimas	<ul style="list-style-type: none"> ○ susisiekti su dėstytojais; ○ bendrauti su kitais dalyviais; ○ parašyti žinutę; ○ bendradarbiauti su kitais dalyviais; ○ ruošti bendrus dokumentus. 	<ul style="list-style-type: none"> ○ interaktyvumas; ○ patogi paieška.

Kurso valdymo posistemė. Kurso valdymo posistemė (6 pav.) apima mokymosi veiklų organizavimo funkcijas. Ši posistemė dažniausiai atitinka dėstytojo, administratoriaus ar kito kurso valdytojo funkcijas. Besimokančiųjų atžvilgiu, ši posistemė atlieka veiklų priminimo, proceso organizavimo funkcijas. Dėstytojas pildo kurso kalendorių bendromis kurso veiklomis, o besimokantysis gali papildyti kalendorių savo individualiomis veiklomis, nustatyti asmeninius parametrus, eksportuoti pateiktas veiklas į asmeninį kalendorių.



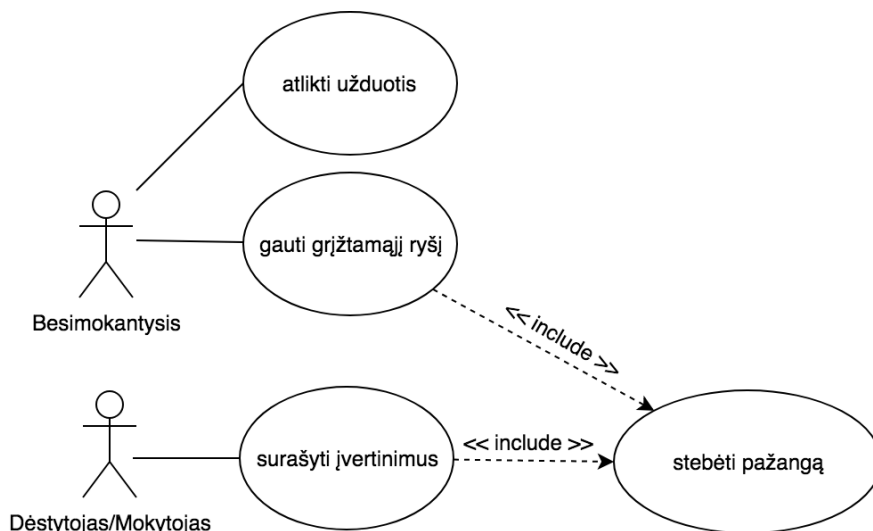
6 pav. Su besimokančiaisiais susijusių kurso valdymo veiklų panaudojimo atvejų diagrama

Mokymosi turinio pateikimo posistemė. Ši posistemė apima kurso mokymosi turinio pateikimo, naudojimo, atnaujinimo veiklas (7 pav.). Tai posistemė, kurioje vyksta didžioji dalis mokymosi veiklų ir yra viena svarbiausių ir dažniausiai naudojamų VMA. Kurso turinio paruošimo veiklų dalyviai dažniausiai būna kurso dėstytojai, asistentai.



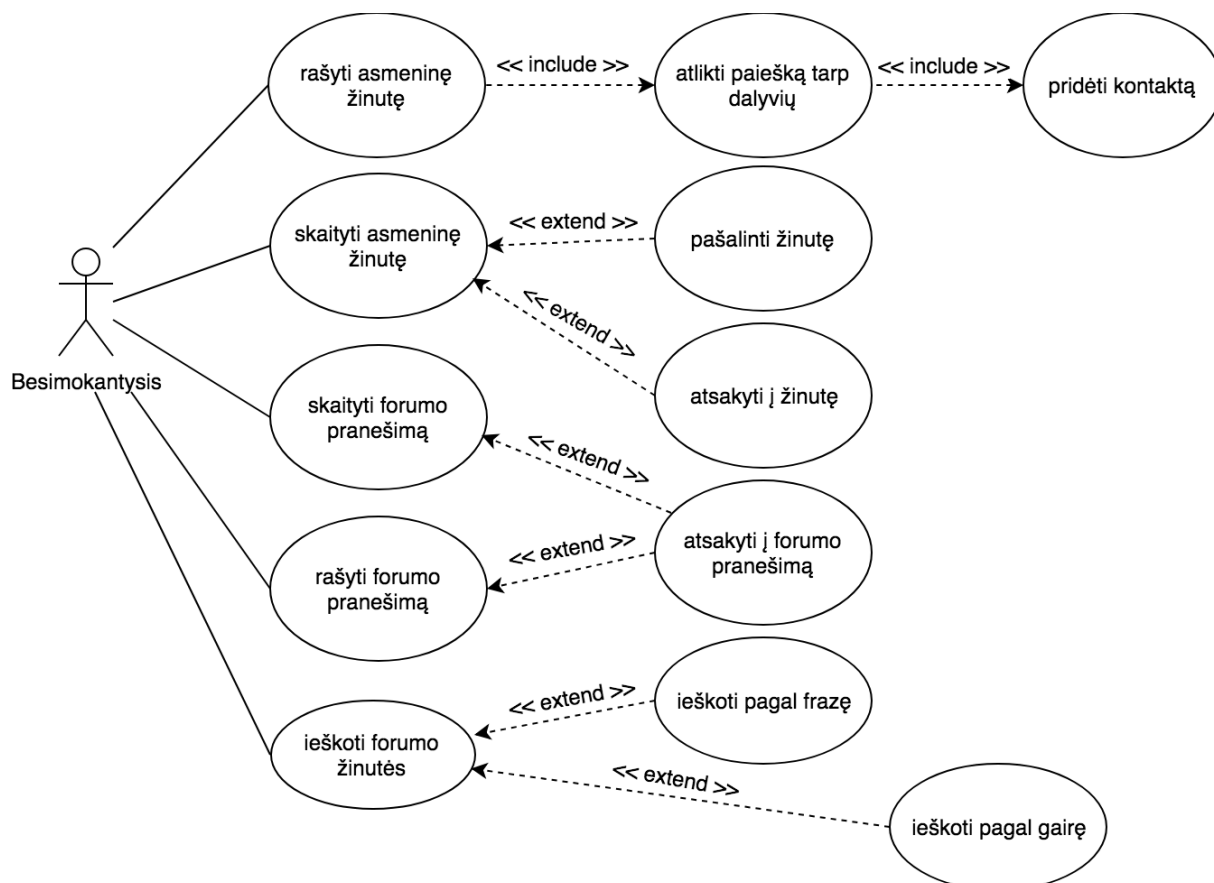
7 pav. Su besimokančiaisiais susijusių mokymosi turinio pateikimo veiklų panaudojimo atvejų diagrama

Vertinimo posistemė. Vertinimo posistemė apima veiklas susijusias su besimokančiojo žinių ir gebėjimų patikrinimu bei įvertinimu (8 pav.).



8 pav. Su besimokančiais susijusių vertinimo veiklų panaudojimo atvejų diagrama

Bendravimo ir bendradarbiavimo posistemė. Bendravimas ir bendradarbiavimas mokymesi yra svarbi savarankiško darbo dalis. Kai tradiciniam bendravimui nėra galimybių, ypač mokantis nuotoliniu būdu, technologijos gali išspręsti bendradarbiavimo problemas, kylančias besimokantiems dėl atstumo, kultūrinių skirtumų.



9 pav. Su besimokančiųjų bendravimu susijusių veiklų panaudojimo atvejų diagrama

Bendravimo veikla (9 pav.) apima įvairius bendravimo būdus VMA: susirašinėjimą žinutėmis, bendravimą forumuose su dėstytoju ar mokytoju bei kitais besimokančiais.

1.7. „Moodle“ aplinkos temos architektūra

„Moodle“ – viena populiariausių pasaulyje atvirojo kodo, modulinė, lanksčiai pritaikoma mokymosi valdymo sistema, leidžianti kurti ir valdyti virtualiąją mokymosi aplinką.

„Moodle“ temų pagalba galima keisti puslapių išdėstymą ir naudotojo sąsają. „Moodle“ tema, tai sistemos papildinio tipas, kuris nustato „Moodle“ elementų atvaizdavimą [23].

Kartu su pradiniu diegimo paketu pateikiamos standartinės pasirenkamos temos: „Boost“, „Clean“ ir „More“. Moodle.org svetainės dokumentacijoje nurodoma, kad standartinė „Boost“ tema, parengta su „Bootstrap 4“ karkasu ir turinti integruotą Sass kompiliatorių, yra skirta lengvam aplinkos pritaikymui naudotojų poreikiams [24].

Kartu su „Moodle“ 3.2 versija buvo pristatyta nauja prisitaikančio dizaino „Bootstrap 4alpha“ karkaso pagrindu sukurta „Boost“ tema, kuri pateikiama kaip pagrindinė aplinkos naudotojo sąsajos tema. „Moodle“ atvirojo kodo vystymo koordinatorius Sander Bangma patvirtino [25], kad tobulinant „Boost“ temą bus siekiama suteikti skirtingų įrenginių naudotojams (išmaniųjų įrenginių, kompiuterių) vieningą patirtį.

Buvo taip pat patvirtinta, kad temos „Clean“ ir „More“ nebebus palaikomos. Planuojama išleisti „Classic“ tema bus taip pat sukurta karkaso „Bootstrap 4“ pagrindu. „Moodle“ vystytojai dokumentacijoje akcentuoja, kad „Boost“ bus plėtojama kaip pagrindinė sistemos tema.

„Boost“ temoje naudojami HTML šablonai (angl. *templates*), generuojami JavaScript kalbos pagalba. Moduliams naudojama „Mustache“ šablonų generavimo kalba. „Mustache“ kalboje HTML moduliai įterpiami į HTML šabloną dvigubų riestinių skliaustų (`{{ ... }}`) pagalba [26]. Šablonų generavimo kalbos naudojimas palengvina modulinio dizaino sistemos kūrimą ir modulių pakartotinio panaudojimo galimybę.

Modulinė „Moodle“ sistemos struktūra leidžia kurti, diegti, keisti, atnaujinti ir pritaikyti dalyvių poreikiams naudotojo sąsajos temas. Esant poreikiui, specifiniam mokymo kurso formatui ar ypatingiems dalyvių poreikiams, specifinė sąsajos tema gali būti pritaikyta ne visai aplinkai, bet tik konkrečiam pasirinktam mokymosi kursui bei jo besimokančiųjų dalyvių poreikiams.

Tačiau, „Moodle“ naudotojo sąsajos temą tenka nuolat atnaujinti. Tai sąlygoja:

- besikeičianti, nuolat atnaujinama „Moodle“ sistema;
- atsinaujinančios naršyklės bei HTML, CSS, JavaScript kalbų atsinaujinimas ir jų palaikymas naršyklėse;
- nauji naudotojų įrenginių formatai (ekrano dydžiai), techniniai parametrai (ekrano raiška, interneto sparta) bei virtualios mokymosi aplinkos pasiekimo būdų ir galimybių įvairovė.

Atnaujinant „Moodle“ sistemą, dėl pasikeitusių techninių parametru, tenka keisti ir naudotojo sąsajos temą, kuri keičia puslapių atvaizdavimą ir neretai sutrikdo naudotoją. Naudojant standartinę naudotojo sąsajos temą, mokymo įstaigai gali būti sudėtinga susikurti vieningą vizualinį identitetą.

„Moodle“ temos kūrimas bei tobulinimas – sudėtingas ir laikui imlus procesas. Tai sąlygoja minėtos mokymosi valdymo sistemos architektūra, kuri remiasi CSS stilių bei JavaScript kešavimu, kas apsunkina bei ženkliai pailgina „Moodle“ temos kūrimui skiriamą laiką. Siekiant lengviau ir efektyviau pritaikyti naudotojo sąsają besimokančiųjų poreikiams ir pasirengti būsimiems sistemos atnaujinimams, siūloma sukurti vieningą mokymo sistemos elementų dizaino rinkinį, kuris būtų pritaikomas naudojantis standartinės „Moodle“ pateiktos temos pagrindu.

1.8. Pakopinių stilių šablonų metodikos grafinėje naudotojo sąsajoje

Pakopiniai stilių šablonai (angl. *Cascading Style Sheets*) – kalba, skirta aprašyti kita kalba aprašyto dokumento (dažniausiai HTML ar XML) atvaizdavimą. Ji leidžia adaptuoti puslapio išvaizdą skirtingiems įrenginių tipams ir ekranams [27]. CSS pavadinimas atspindi kalbos interpretavimo būdą: pakopomis, eilės tvarka vykdomos aprašytos stiliaus taisyklės, atsižvelgiant į taisyklės eiliškumą, paveldėjimą ir identifikatoriaus atributą.

CSS atskyrimas nuo HTML dokumento leidžia atskirti puslapio struktūrą nuo atvaizdavimo bei taikyti vieningas stiliaus taisykles skirtingiems puslapiams, atskiroms sistemos dalims ar visai sistemai. Tačiau CSS globalumas sukuria problemą, su kuria susiduria besiplečiančių, augančių bei moduliarių sistemų kūrėjai: CSS stilių taisyklių neįmanoma izoliuoti ir taikyti tik konkrečiai grafinės naudotojo sąsajos (angl. *graphical user interface, GUI*), sričiai. CSS globalumas lemia tai, kad, sistemai plečiantis, jau esantys stiliai gali neprognozuojama linkme paveikti naujai kuriamus elementus, o veikimo pakopiškumas sukelti nepageidaujamus efektus dėl taisyklių paveldėjimo ar jų persidengimo (perrašymo) [28].

Minėtų CSS savybių, kurios saityno sistemos vystytojui gali suteikti nemažai privalumų ir lankstumo, kuriant grafinės naudotojo sąsajos atvaizdavimą, šalutinis neprognozuojamas poveikis dideles internetines sistemas plėtojančias kompanijas bei nepriklausomus vystytojus paskatino kurti metodikas, skirtas sistemos CSS architektūros tobulinimui. Šios savybės (globalumas, specifiškumas, pakopiškumas), puikiai tiko internetinių svetainių dizainui, nedidelių sistemų vystytojams, kuomet visus CSS stilius galima pateikti viename faile. Tačiau, sistemoms plečiantis, CSS failo eilučių kiekis gali siekti kelis ar net keliasdešimt tūkstančių. Tokiu atveju, tampa sudėtinga palaikyti, koreguoti, atnaujinti bei tobulinti sistemą. Grafinės sąsajos vystytojai pradėjo ieškoti sprendimų ir kurti metodikas, organizuojančias CSS architektūrą išnaudojant stipriąsias šios kalbos savybes.

1.8.1. Pakopinių stilių rekomendacijos „Moodle“ temai

„Moodle“ dokumentacijoje pateikiamos esminės pakopinių stilių (CSS) struktūros rekomendacijos. Rekomenduojama nustatyti vieningus standartus temos projektui bei laikytis paprastumo ir aiškumo principų [29].

Identifikatorių pavadinimų pasirinkimui patariama [29]:

- naudoti mažąsias raides ir brūkšnius (-) ar žemesnius brūkšnius (_) žodžių atskyrimui;
- vardams naudoti paprastus angliškus žodžius;
- naudoti semantinius pavadinimus: identifikatorių vardai turėtų būti suprantami ir maksimaliai iliustruoti elemento paskirtį;

- vengti ID identifikatorių naudojimo.

1.8.2. Block Element Modifier (BEM) metodika

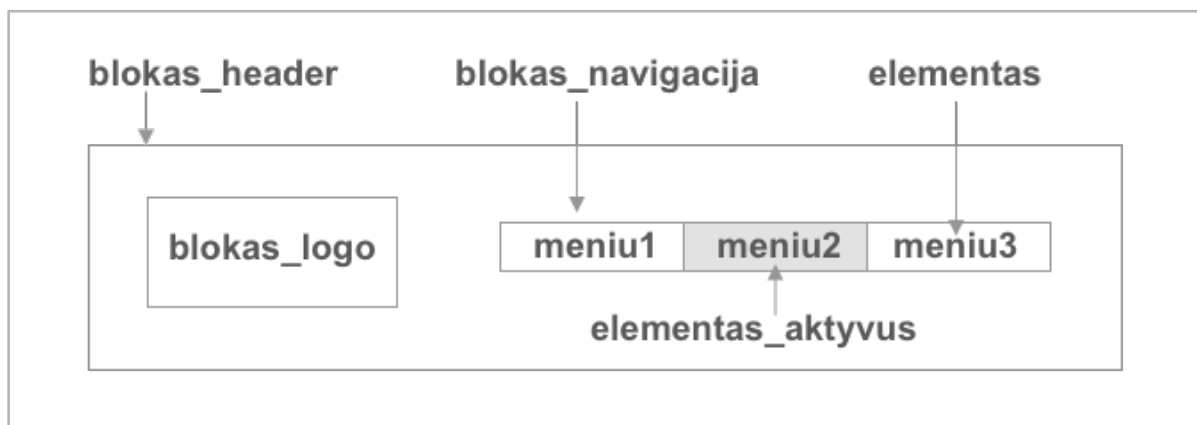
BEM metodika sukurta įmonėje „Yandex“ ir šiuo metu yra viena iš populiariausių CSS metodikų. 2006 įmonė, kuri valdė paieškos sistemą, pradėjo vystyti didelio masto projektus: „Yandex.Music“ ir „Ya.Ru“. Šie projektai, kuriuos sudarė dešimtys internetinių puslapių, atskleidė tradicinio požiūrio į vystymą trūkumus [30]:

- bet koks pakeitimas, atliktas vieno puslapio kode, paveikia kitų puslapių kodą;
- labai sudėtinga tinkamai parinkti klasių pavadinimus.

Didelio projekto vystymas galėjo tapti nevaldomas, tad reikėjo nustatyti taisykles, kaip kurti ar keisti klases, žymes, vizualius komponentus [30]. Iškilė būtinybė standartizuoti ir dokumentuoti CSS architektūrą bei vystymo procesą.

Buvo pristatyta blokų koncepcija [30]:

- blokas yra HTML puslapio fragmentas, su tam tikru žymėjimu ir stiliais;
- blokai, gali turėti savyje kitus blokus.



10 pav. BEM metodo blokų struktūra HTML puslapyje

Blokas – logiškai ir funkcionaliai nepriklausomas puslapio komponentas. Blokas savyje inkapsuliuoja stilius (CSS), išdėstymą, veikimą (JavaScript). Blokai yra nepriklausomi ir gali būti daug kartų panaudojami skirtingose projekto vietose (10 pav.) [31].

Blokai, kaip nepriklausomi elementai, gali būti atvaizduojami skirtingose puslapio ar projekto vietose, taip pat gali būti perkeltami ir panaudojami kituose projektuose, nekeičiant jų CSS atvaizdavimo stilių ar JavaScript kodo [31].

Elementas – bloko sudėtinė dalis, kurios negalima panaudoti kitame kontekste nepakeitus jo išdėstymo ar atvaizdavimo stiliaus (10 pav.) [31].

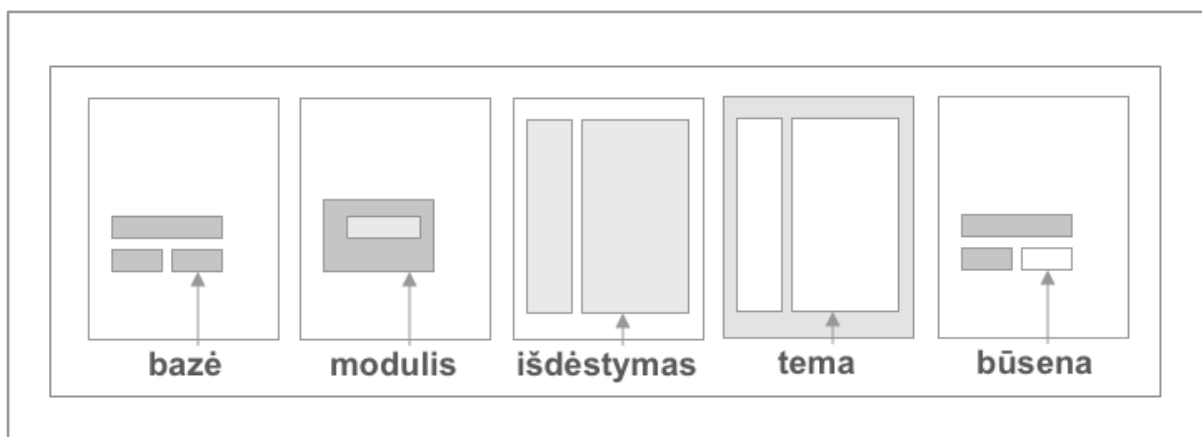
Modifikatorius – nustato bloko ar elemento veikimą ar atvaizdavimo būdą. Modifikatoriaus naudojimas nėra privalomas [31].

1.8.3. Scalable modular architecture CSS (SMACSS)

Metodo kūrėjas J. Snook apibūdina labiau kaip rekomendacijų rinkinį, ne sistemą ar griežtą karkasą. SMACSS metodas, tai būdas peržiūrėti savo sistemos dizaino procesą ir pritaikyti jos taisykles pagal sistemos poreikius [32]. Ši metodika buvo kuriama ir pirmą kartą buvo pritaikyta „Yahoo Mail“ platformoje.

Puslapio elementus siūloma priskirti į penkias kategorijas einant nuo globalių link specifiškesnių taisyklių (11 pav.) ir pagal tai organizuoti savo CSS struktūrą [33]:

- bazinės (angl. *base*);
- išdėstymo (angl. *layout*);
- modulio (angl. *module*);
- būsenos (angl. *state*);
- temos (angl. *theme*).



11 pav. SMACSS metodo kategorijos

Bazinės taisyklės. Pagrindinių HTML puslapio elementų (<body>, <main>, <section>, , ir kt.) stiliai. Šioje kategorijoje daugiausiai naudojami HTML žymių ir atributų identifikatoriai, panaudojama CSS kalbos globalumo savybė [34].

Išdėstymo taisyklės. Nustato globalių elementų išdėstymo (<header>, <footer>, <sidebar> ir kt.) tvarką ir skirsto puslapį į sritis (skiltis), kurios apjungia skirtingus modulius [33, 34]. Autorius metodikoje nerekomenduoja ID identifikatorių, tačiau jei tai yra neišvengiama, siūlo naudoti tik šioje kategorijoje ir tik su sąlyga, jei ID identifikatorius nepasikartos puslapyje [34].

Modulio taisyklės. Moduliai, tai blokai, kurie gali būti panaudojami kelis kartus skirtingose puslapio vietose. Rekomenduojama naudoti tik klasių identifikatorius [34].

Būsenos taisyklės. Nustato, kaip moduliai ir išdėstymo taisyklės atrodys tam tikroje būsenoje (paslėptas, išskleistas, aktyvus, neaktyvuotas). Taip pat čia aprašoma, kaip jie atrodys skirtinguose ekranuose [33].

Temos taisyklės. Panašios į būsenos taisyklės, tik nustato kaip moduliai ir išdėstymas gali atrodyti skirtinguose nustatymuose. Ši kategorija yra naudojama pagal poreikį, jei tai yra būtina pagal projekto specifikaciją [33].

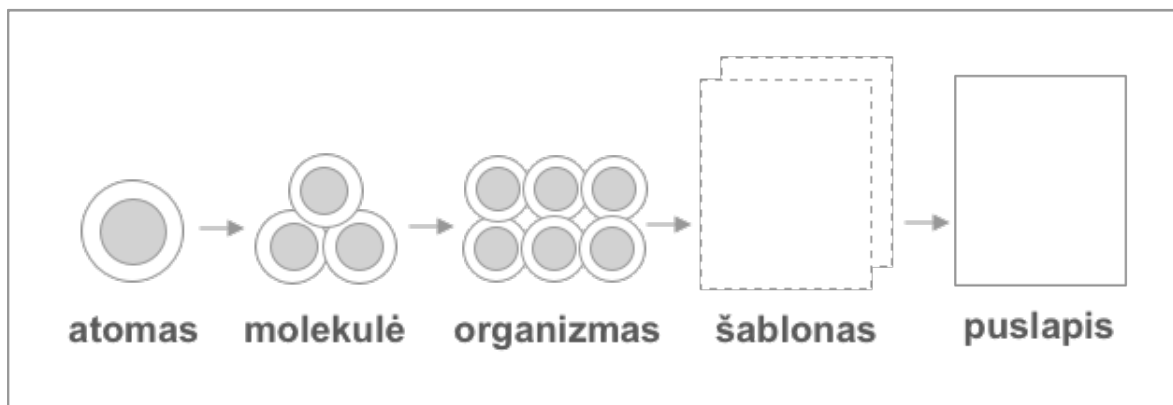
1.8.4. Atominio dizaino architektūra

Pristatyta B. Frost, remiasi elementų, panašių į elementus, esančius gamtoje pavadinimais [35]:

- atomai;
- molekulės;
- organizmai.

Juos apjungia:

- šablonai;
- puslapiai.



12 pav. Atominio dizaino koncepcijos modulių kategorijos

Šis požiūris atspindi ir kitų metodikų pagrindinę moduliarumo nuostatą: blokai, elementai, moduliai, jų sąryšiai ir jungimas pagal tam tikras taisykles (12 pav.). Šios koncepcijos išskirtinumas yra, kad atsižvelgdami į elemento kategoriją (atomas, molekulė), galime suprasti elemento svarbą bei granuliacijos lygį.

Atominio dizaino architektūros koncepcija virtualiai mokymosi aplinkai naudojama „FutureLearn“ platformoje: atomų sąvoka naudojama neskaidomiems, smulkiausiems elementams [36]. Atomai sudaro molekules, o molekulės – organizmus. Savo straipsnyje „FutureLearn“ sąsajos dizainerė A. Kholmatova [36] siūlo sistemos modulius skirstyti pagal tai, ar elementas gali egzistuoti sistemoje savarankiškai, nepriklausydami aukštesniam „tėviniam“ moduliui. Aukštesnio savarankiškumo modulis gali būti traktuojamas kaip nepriklausomas modulis. Todėl, norėdami skirstyti modulius pagal jų savarankiškumo lygmenį, „FutureLearn“ dizaino sistemos kūrėjai pasirinko naudoti tik atomų ir molekulių koncepcijas iš atominio dizaino metodikos [37].

1.8.5. CSS metodikų apžvalga ir palyginimas

Iš aprašytų metodikų matome, kad visos HTML elementus klasifikuojančios CSS metodikos taiko panašų požiūrį į elementų skaidymą, skiriasi tik elementų įvardinimas ir granuliacijos lygmuo.

Lentelėje (2 lentelė) pateikiami modulių lygmenys surikiuoti nuo smulkausio, neskaidomo elemento bei artimiausi jų atitikmenys kitose metodikose.

3 lentelė. CSS metodikų modulių skaidymo būdai

	Neskaidomi elementai	Apjungiantys elementai	Elementų dariniai	Funkcionalūs dariniai	Būsenos
Atominis dizainas	Atomas	Molekulė, Organizmas	Šablonas	Puslapis	
BEM	Elementas	Blokas			Modifikatorius
SMACSS	Bazė	Modulis	Išdėstymo taisyklės	Tema	Būsena

Iš duomenų, pateiktų lentelėje matome, kad visose sutinkami dariniai yra mažiausiai skaidomi moduliai. Kuriamai dizaino sistemai bus naudojamos elementų ir blokų sąvokos.

1.9. Sass preprocesorius

Sass yra pakopinių stilių šablonų kalba, kompiliuojama į CSS. Ji suteikia galimybę naudoti kintamuosius, funkcijas, fragmentuoti bei importuoti stiliaus šablonus, atlikti matematinius veiksmus ir pilnai palaiko CSS sintaksę. Sass padeda suskaidyti ir organizuoti didelius stiliaus šablonų failus [38]. Sass failai yra su .sass plėtiniu ir yra dažnai vadinami SCSS failais. Sass tai preprocesorius, paverčiantis SCSS į CSS failus.

Sass palaiko šias funkcijas:

- kintamuosius;
- importavimą bei skaidymą;
- stiliaus deklaracijų įtraukimą vienu į kitas;
- matematinius veiksmus.

„Moodle“ sistema turi integruotą Sass kalbos kompiliatorių, tad „Boost“ temos pakopinių stilių šablonai yra išskaidyti į atskirus failus pagal jų paskirtį. Įdiegtoje temoje SCSS failų fragmentai sujungiami ir paverčiami vienu CSS failu.

Sass ruošiniai (angl. *presets*) „Boost“ temoje. Ruošinys yra SCSS failas, papildantis „Boost“ arba paveldinčią temą. Šių ruošinių pagalba galima pakeisti temos stiliaus nuostatus. Ruošinių failus galima įkelti bei pasirinkti Moodle sistemos administravimo dalyje.

1.10. „Bootstrap“ karkasas

„Bootstrap“ – naudotojo sąsajos karkasas, sukurtas „Twitter“ socialiniam tinklui. 2011 metais karkasas pateiktas viešam naudojimui ir tapo vienu populiariausiu naudotojo sąsajos karkasu [39]. Šiuo metu vyrauja „Bootstrap“ 4-oji versija kurioje įvykdyta keletas svarbių architektūrinių pakeitimų: perėjimas prie Sass preprocesoriaus bei CSS „flexbox“ savybės naudojimo.

„Bootstrap“ karkaso naudojimas gerokai pagreitina bei supaprastina naudotojo sąsajos kūrimą, taip pat padeda vystytojams, neturintiems pakankamai HTML semantikos, CSS, pasiekiamumo įgyvendinimo technologijos žinių, pakankamai paprastai pasiekti siekiamą sąsajos išvaizdos rezultatą. Tačiau, karkaso naudojimas turi ir trūkumų:

- siunčiamas didelis CSS kodo kiekis, kuris nebūtinai yra reikalingas;
- HTML kodas perkraunamas daugybe klasių;
- karkaso atnaujinimas gali sukelti nepageidaujamus dizaino pasikeitimus.

1.11. CSS stilių pateikimo ateities perspektyvos

JavaScript karkasai. React.js, Vue.js ir kiti sparčiai populiarėjantys JavaScript karkasai remiasi „CSS-in-JS“ požiūriu, kai konkretaus modulio CSS stilius aprašomas JavaScript modulyje, izoliuojant ir inkapsuliuojant stilių šabloną. Šiuo atveju modulio struktūra ir dizainas pateikiami kartu, kas prieštarauja atsakomybių atskyrimo principui, tačiau suteikia galimybę užtikrinti CSS stilių inkapsuliaciją.

HTTP/2 protokolas. Perėjus prie šio protokolo, nereikės pateikti pilno CSS failo <head> srityje. CSS stiliaus šablonai gali būti išskaidomi ir pateikiami skirtingose puslapio <body> srityse [40]. Tai leis CSS stilius suskaidyti į atskirus modulius ir juos taikyti pagal poreikį, bei atsisakyti vieno didelio CSS failo siuntimo, siekiant sumažinti HTTP užklausų kiekį, kadangi HTTP užklausos neįtakos puslapio atvaizdavimo naršyklėje greičio.

1.12. Išvados

1. Naudotojų, naršančių mobiliuoju įrenginiu informacijos prioritetai skiriasi, tad būtina atsižvelgti ne tik į pritaikymą ekrano dydžiui, bet ir informacijos pateikimą sistemoje.
2. Naudotojų įrenginių ekrano dydžių plati gama bei sąveikos būdų įvairovė sukuria poreikį kurti naudotojo sąsają remiantis naudotojų poreikiais bei sąveikos būdais, o ne naudojamais įrenginiais.
3. Projektuojant ir kuriant virtualiosios mokymosi aplinkos sąsają, būtina atsižvelgti į Pasaulinio žiniatinklio konsorciumo pasiekiamumo gaires ir Europos Sąjungos viešojo sektoriaus institucijų pasiekiamumo reikalavimus bei projektuoti sąsajos elementus, atitinkančius AA arba A lygmeniu. Sąsajos pasiekiamumas gerina naudotojo patirtį.
4. Vieningos naudotojo sąsajos dizaino sistemos sukūrimas gali pagerinti ir paspartinti standartinio Moodle temos šablono pritaikymą jos naudotojų poreikiams.
5. Pritaikant virtualios mokymosi aplinkos sąsają besimokantiesiems, gerėja naudotojų patirtis ir kitiems sistemos dalyviams.
6. „Moodle“ sistemos atnaujinimas gali pareikalauti naudotojo sąsajos šablono keitimo, kas gali sutrikdyti virtualiosios mokymosi aplinkos vizualinį identitetą ir pabloginti naudojamumo kokybę.
7. Moduliarios sistemos struktūra sukuria prielaidas moduliarių CSS metodikų taikymui.

8. Metodikas galima taikyti pagal poreikį, panaudojant sistemai ir organizacijai priimtinius metodus. Galima apjungti metodus iš kelių skirtingų metodikų.
9. CSS metodikas galima modifikuoti, jas pritaikant vystomai sistemai.

2. „Moodle“ naudotojo sąsajos vertinimo kriterijai

Naudotojo sąsajos dizainas gali būti įvertintas remiantis skirtingais požiūriais ir būdais. J. Nielsen ir R. Molich straipsnyje išskyrė keturis pagrindinius naudotojo sąsajos įvertinimo būdus [41]:

- formalus, taikant tam tikrą analizę ar techniką;
- automatinis, naudojant kompiuterizuotą procedūrą;
- empirinis, atliekant eksperimentus su tikslinėmis naudotojų grupėmis;
- euristinis, pateikiant savo įžvalgas bei savo nuomonę.

Minėtame straipsnyje detaliau nagrinėjamas euristinis vertinimo būdas, jo privalumai bei trūkumai. Tyrimo metu atlikti eksperimentai parodė, kad individualūs vertintojai identifikavo nuo 20 proc. iki 51 proc. įvertintos sąsajos naudojamumo trūkumų [41]. Empirinis vertinimas reikalauja tikslinių sistemos naudotojų profilių nustatymo, kas reikštų, kad vertinama mokymosi sistemos, kaip galutinio produkto, sukurto konkrečiai mokymosi institucijai ar organizacijai, naudotojo sąsaja.

Automatiniam sąsajos vertinimui skirtos priemonės nuolat tobulėja. Naudojamoms ar kuriamoms sistemoms įvertinti pritaikomos naudotojų elgsenos sekimo analitinės programos. Sąsajos vystytojams lengvai prieinami naršyklėse diegiami papildiniai, tokie kaip „Google Chrome Lighthouse“, kuris procentine išraiška pateikia sąsajos pasiekiamumo įvertinimą bei papildomas rekomendacijas sąsajai tobulinti. Tačiau automatizuotos sistemos taip pat turi ir trūkumų, kadangi pasiekiamumo sprendimai dažnai priklauso nuo konteksto ir sistemos paskirties. Automatizuotas testavimo būdas gali būti taikomas kaip papildoma priemonė naudotojo sąsajos pasiekiamumo vertinime. Šiame darbe nagrinėjami būdai ir priemonės virtualiosios mokymosi aplinkos „Moodle“ naudotojo sąsajai kurti, todėl pasirinktas formalus sąsajos vertinimo metodas, paremtas Pasaulinio žiniatinklio konsorciumo parengtomis „WCAG“ (angl. *Web Content Accessibility Guidelines*) gairėmis [18] bei „WebAim“ pasiekiamumo kriterijų sąrašu [42].

„WCAG“ pateikia tinklalapių turinio pasiekiamumo gaires, kurios skirstomos į šias pagrindines naudotojo sąsajos bei turinio savybes [18]:

- suvokiama (angl. *perceivable*);
- valdoma (angl. *operable*);
- aišku, suprantama (angl. *understandable*);
- universalu (angl. *robust*).

Minėtos „WCAG“ gairės taip pat pažymėtos raidėmis A, AA ir AAA. Viena raidė reiškia bazinį (būtiną) pasiekiamumo lygmenį. Didesnis A raidžių skaičius reiškia mažesnę gairės svorį pasiekiamumo vertinime. Vertinimo kriterijai parengti atsižvelgiant į virtualiosios mokymosi sistemos „Moodle“ bazinę (angl. *core*) sąsają bei pavyzdinį kurso „Diskusijų ir bendradarbiavimo priemonė Piazza“ turinį.

2.1. HTML kodo semantika

Semantinis kodas suteikia tinklalapiui struktūrą bei reikšmę HTML elementams. Tai palengvina naudojimąsi tinklalapiu ne tik asmenims su specialiais poreikiais, naudotojams, naršantiems klaviatūros pagalba ir besinaudojantiems balso asistentu.

4 lentelė. Semantinio HTML kodo gairės

Lygmuo	Kriterijai
A	Antraštėms (<h1>), sąrašams (, ir <dl>), tinklalapio dalims (<main>, <nav>) ir pan. naudojamas semantinis žymėjimas. Semantinis žymėjimas panaudotas teisingai, pagal paskirtį.
A	Skaitymo ir navigacijos seka yra logiška ir intuityvi.
A	Puslapis turi informatyvų pavadinimą (angl. <i>title</i>).
AA	Tinklalapio orientacija nėra apribota (pvz. tik vertikali ar gulsčia).
AAA	Specialūs HTML5 elementai (<main>, <nav>, <section>, <article>) ar ARIA žymės (role="banner", role="navigation", role="main")
AAA	Kur reikalinga, naudojamas WAI ARIA žymėjimas, siekiant geriau identifikuoti sąsajos komponentų paskirtį.

Lentelėje (3 lentelė) pateikti tinklalapio kodo semantikos kriterijai, skirti įvertinti HTML kodo pasiekiamumą.

2.2. Spalvos kontrastas ir tekstas

Spalva ir šriftas – vieni pagrindinių vizualaus dizaino išraiškos priemonių, tačiau derėtų prisiminti, kad ne visi naudotojai vienodai skiria spalvas ar gerai mato tekstą. Lentelėje (4 lentelė) išvardinti kriterijai padeda užtikrinti galimybes naudotojams matyti ir suprasti pateiktą turinį.

5 lentelė. Spalvos ir teksto pasiekiamumo gairės

Lygmuo	Kriterijai
A	Spalva nėra vienintelis matomas vizualinių elementų išryškinimo būdas.
A	Nuorodų išskyrimui iš aplinkos nėra naudojama vien tik spalva (pvz. naudojamas pabraukimas), nebent kontrastas tarp nuorodos tekste ir aplinkinio teksto yra santykiu 3:1 a bei atsirandantis papildomas išskyrimas (pabraukimas) nuorodą fokusuojant ar užvedant pelę.
AA	Tekstas kontrastuoja su fonu santykiu 4,5:1 ir daugiau.
AA	Didesnis tekstas (nuo 18pt) arba 14pt pajuodintas tekstas kontrastuoja su fonu santykiu 3:1 ar daugiau.
AA	Puslapio tekstas yra įskaitomas bei funkcionalus, jį padidinus (angl. <i>zoom</i>) iki 200%.
AAA	Teksto blokai, ilgesni nei viena eilutė: <ul style="list-style-type: none">- ne ilgesni nei 80 ženklų;- nenaudojama abipusė lygiuotė;- tarpas tarp eilučių yra mažiausiai ½ teksto dydžio;- tarpas tarp paragrafų yra 1,5 karto didesnis už tarpą tarp eilučių.

Pažymėtina, kad spalvos ir teksto pateikimo kriterijai svarbūs ne tik asmenims, turintiems regos ar spalvų matymo sutrikimų, bet ir kognityvinių ar mokymosi sutrikimų turintiems naudotojams (pvz. disleksija).

2.3. Navigacija

Galimybė naršyti bei interaktyviai sąveikauti su sistema naudotojo sąsajos pagalba yra vienas svarbiausių pasiekiamumo veiksnių, užtikrinančių naudotojui valdymo galimybę.

6 lentelė. Navigacijos pasiekiamumo gairės

Lygmuo	Kriterijai
A	Visos puslapio funkcijos veikia naudojantis klaviatūra. Tai netaikoma toms funkcijoms, kurios negali būti pasiektos klaviatūros pagalba (pvz. piešimas).
A	Kiekviename puslapyje pasikartojantiems elementams (pvz. navigacijai) praleisti, pateikiamos greitosios nuorodos.
A	Nuorodų, formų elementų, kitų interaktyvių elementų navigacijos tvarka yra intuityvi ir logiška.
AA	Fokusavimas: vizualiai akivaizdu, kuris puslapio elementas yra fokusuotas ir matomas.
AA	Navigacijos nuorodos, pasikartojančios kiekviename puslapyje (pvz. meniu), naviguojant per puslapius nekeičia savo išdėstymo tvarkos.
AAA	Interaktyvūs elementai yra mažiausiai 44px aukščio ir 44px pločio, nebent yra pateikta alternatyvi nuoroda, interaktyvus elementas yra teksto dalis (interaktyvi nuoroda tekste), interaktyvus elementas negali būti modifikuotas (standartinis formos „check“ laukelis).

Lentelėje (5 lentelė) pateiktos gairės, padėsiančios įvertinti tinklalapio pasiekiamumą naudotojams, besinaudojantiems skirtingais įrenginiais ir navigacijos būdais.

2.4. Prisitaikantis dizainas

Prisitaikantis dizainas vertinamas imituojant skirtingus pasirinktus ekrano dydžius, naudojantis naršyklėje įdiegtais vystytojo įrankiais (ang. *web developer tools*). Naudotojo sąsajos dizaino vertinimui pasitelkta Google Chrome naršyklėje įdiegta priemonė „Developer Tools“ leidžia pasirinkti testuojamus ekrano dydžius arba rinktis iš populiariausių naudotojų įrenginių modelių sąrašo (pvz.: „iPhone“, „iPad“, „Galaxy S5“ ir kt.)

7 lentelė. Prisitaikantis dizaino pasiekiamumo gairės

Lygmuo	Kriterijai
AA	Sumažinus ekrano dydį iki 320px neprarandamas funkcionalumas, neatsiranda horizontali slankjuostė. Horizontalios slankjuostės išimtis taikomos turiniui, kuriam yra būtina horizontali peržiūra, pvz. duomenų lentelės, grafai, žemėlapiai.
AA	Elementams, kurie savyje turi teksto elementų, nenustatomas aukštis pikseliais.

Prisitaikančio dizaino taisyklės nustatomos medijų užklausų (angl. *media query*) pagalba. Pateikiama minimalaus arba maksimalaus ekrano pločio užklausa.

8 lentelė. Ekranų dydžiai, sugrupuoti pagal raišką

Orientacija	Ekranų raiška (px)	Populiarumas, proc.
Horizontali	1536, 1600, 1680, 1920, 2560	32,54
Horizontali	1024, 1280, 1366, 1440	32,57
Vertikali	360, 375	14,23
Kita (nežinoma)	nežinoma	20,67

Remiantis anksčiau tyrime pateikiama „Globalstats“ ekranų raiškos statistika [9], naudotojų ekranų dydžiai buvo sugrupuoti į mažus (mobilaus įrenginio), vidutinius ir didelius ekranus. Pabrėžtina, kad būtina įvertinti naudotojų galimybę naudotis ir keliais įrenginiais (pvz. išmaniuoju telefonu ir kompiuteriu).

2.5. Išvados

1. Naudotojo sąsajos vertinimas atliekamas atsižvelgiant į tinklalapio kontekstą, paskirtį bei pobūdį.
2. Naudotojo sąsajos komponentų pasiekiamumui įvertinti naudotinos oficialios Pasaulinio žiniatinklio konsorciumo parengtos gairės, kuriose vadovaujama ar puslapio turinys gali būti suvokiamas, valdomas, aiškus bei universalus.
3. Prisitaikančio dizaino realizacija nepsiriboja tik tinklalapio pritaikymu mobiliems įrenginiams. Atkreiptinas dėmesys ir į naudotojus, besinaudojančius dideliais ar itin dideliais ekranais bei sąsajos dizaino pritaikymą visiems naudotojams.
4. Naudotojai gali naudotis daugiau nei vienu įrenginiu, tad to paties naudotojo poreikiai skirtingomis aplinkybėmis bus skirtingi.

3. „Moodle“ aplinkos naudotojo grafinės sąsajos dizaino sistema

3.1. Dizaino sistemos kūrimas

Kaip apibrėžia A. Kholmatova knygoje „Dizaino Sistemos“ [43], dizaino sistema – tai tarpusavyje susijusių modelių (angl. *patterns*) ir jiems skirtų taisyklių rinkinys, nusakantis skaitmeninio produkto atvaizdavimo būdą. Modeliai – tai pasikartojantys elementai (mygtukai, teksto laukai, simboliai, spalvos, teksto struktūros), kuriuos sujungus kuriama naudotojo sąsaja [43]. Dizaino sistemos pagalba inventorizuoja sistemos elementai, dokumentuoja jų struktūrą, atvaizdavimo būdą, būsenas. Tai padeda išlaikyti vieningą HTML ir CSS žymėjimo stilių, laikytis dokumento struktūros, hierarchijos reikalavimų, taip pat palaikyti virtualios mokymosi aplinkos išvaizdos vieningumą, vizualinį identitetą.

Modelių bibliotekos (angl. *pattern library*) kūrimas nėra apribotas konkrečios technologijos ar įrankio rėmuose. Aprašyti sistemos modelius ir pateikti naudojimo pavyzdžius galima ir „Google Docs“ arba kita, dokumentuoti bei bendradarbiauti skirta priemone, taip pat galima naudoti specializuotas vaizdines dokumentavimo priemones [43].

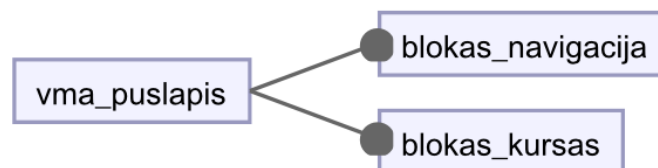
Siekiant kuriamą dizaino sistemą maksimaliai priartinti prie Moodle terminologijos, sistemoje bus naudojamos dvi – blokų ir elementų sąvokos. Elementai atitiktų smulkesnius, aukštesnės granuliacijos sistemos komponentus, kurie jungiami sudaro blokus. Blokai būtų sudaryti iš apjungtų elementų ir kitų blokų.

3.2. „Moodle“ temos dizaino sistemos projektavimas

„Moodle“ aplinkos struktūra remiasi modulių koncepcija: pagrindinė sistemos dalis (angl. *core*) ir diegiami papildiniai, kurių pagalba sistema tampa lanksti ir pritaikoma įvairiems naudotojų poreikiams.

Sudarytas Moodle kurso puslapio informacinis modelis. HTML puslapio komponentai išskaidyti pagal požymius ir išreikšti grafiškai požymių diagramos pagalba. Požymių diagrama braižoma programos „Eclipse“ pagalba. Modulių HTML ir klasių žymėjimas pateiktas lentelėje: žymima semantinę reikšmę turinčiomis HTML kalbos žymėmis (<main>, <nav>), kurios pagerina puslapio pritaikymą specialiesiems poreikiams.

Virtualiosios mokymosi aplinkos puslapis skaidomas į struktūrinius blokus (13 pav.): navigacija ir kursas (nagrinėjamas VMA puslapis yra kurso puslapis).

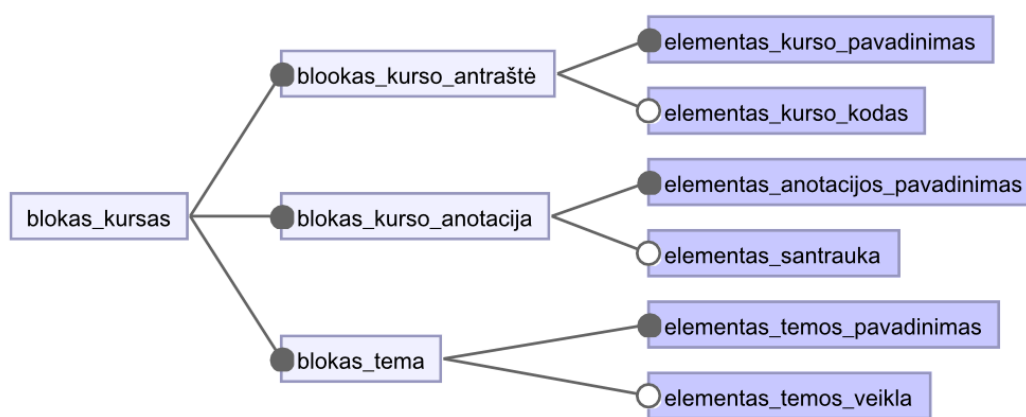


13 pav. VMA kurso puslapio skaidymas blokais

9 lentelė. VMA puslapio pagrindinių blokų žymėjimas

Modulis	HTML žymėjimas	CSS Klasė	Rūšis
vma puslapis	<body>	.b-page	Blokas
navigacija	<nav> – pagrindinė navigacija <div> – kita navigacija	.b-navigation	Blokas
kursas	<main>	.b-course	Blokas

Kurso blokas (14 pav.) skaidomas į blokus ir juos sudarančius smulkesnius elementus.



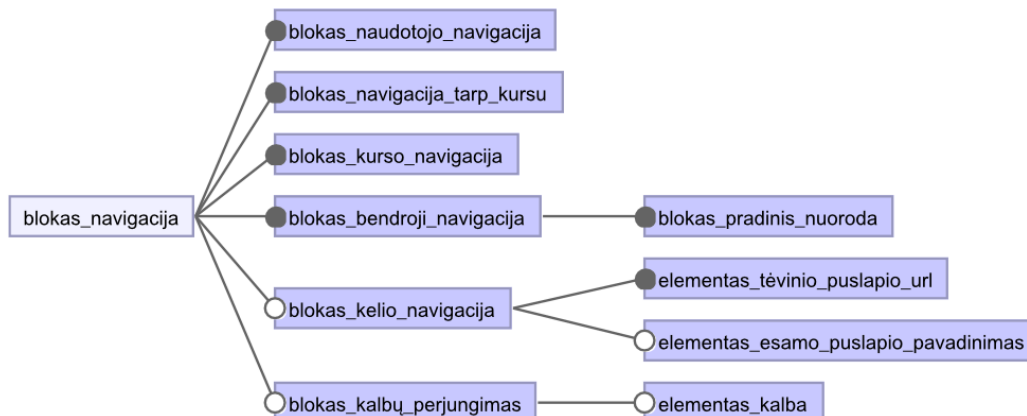
14 pav. Kurso bloko skaidymas blokais ir elementais

10 lentelė. Kurso bloką sudarančių blokų ir elementų žymėjimas

Modulis	HTML žymėjimas	CSS Klasė	Rūšis
kurso antraštė	<header>	.b-header-main	Blokas
kurso anotacija	<section>	.b-summary	Blokas
tema	<article>	.b-topic	Blokas
kurso pavadinimas, kurso kodas	<h1>	.e-header-main_title	Elementas
anotacijos pavadinimas	<h2>	.e-summary_title	Elementas
santrauka	<div>	.e-summary_text	Elementas
temos pavadinimas	<h2>	.e-topic_title	Elementas

temos veikla	<div>	.e-topic_activity	Elementas
--------------	-------	-------------------	-----------

Navigacijos blokas (15 pav.) skaidomas į blokus ir juos sudarančius smulkesnius elementus.



15 pav. Navigacijos bloko skaidymas blokais ir elementais

11 lentelė. Navigacijos bloką sudarančių blokų ir elementų žymėjimas

Modulis	HTML žymėjimas	CSS Klasė	Rūšis
naudotojo navigacija	<div> 	.b-nav-user	Blokas
navigacija tarp kursų	<nav> 	.b-nav-courses-main	Blokas
kurso navigacija	<nav> 	.b-nav-course	Blokas
bendroji navigacija	<nav> 	.b-nav-general	Blokas
kelio navigacija		.b-nav-breadcrumbs	Blokas
kalbų perjungimas	<div> 	.b-nav-lang	Blokas
pradinis nuoroda	<div>	.b-home-link	Blokas
tėvinio puslapio url	 <a>	.e-breadcrumb-parent	Elementas
esamo puslapio pavadinimas	 	.e-breadcrumb-current	Elementas
kalba	 <a>	.e-lang	Elementas

Požymių diagramoje į komponentus išskaidytas virtualiosios mokymosi aplinkos kurso puslapis, lentelėje (10 lentelė) pateikti HTML žymėjimo variantai, sudaryti CSS klasių pavadinimai remiantis BEM klasių įvardinimo taisyklėmis bei nurodytas komponento granuliacijos lygis. Sudaryta požymių diagrama (2 priedas) bus remiamasi kuriant bei jungiant komponentus dizaino sistemoje.

3.3. Dizaino sistemos kūrimo priemonės

Egzistuoja nemažai atvirojo kodo dizaino sistemų kūrimo priemonių, kurias galima naudoti kuriant bei tobulinant savo komponentų biblioteką ar dizaino sistemą. Dažniausiai tai yra parsisiunčiamos ir diegiamos lokaliai priemonės, generuojančios statinius HTML dokumentus. Vieni įrankiai labiau orientuoti į grafinės naudotojo sąsajos kūrimą HTML ir CSS priemonėmis, kiti labiau orientuoti į JavaScript karkasų, tokių kaip React ar Vue.js naudojimą.

3.3.1.,,Pattern Lab“

„Pattern Lab“, tai statinių HTML puslapių generatorius (generuojantis PHP arba Node.js kalbų pagalba), apjungiantis sukurtus dizaino sistemos naudotojo sąsajos komponentus [44]. „Pattern Lab“ įrankis Node.js su Node paketų valdymo programa (ang. *Node Package Manager*, sutrumpintai NPM) arba PHP kalbos pagalba. Pattern Lab generuoja HTML, CSS ir JavaScript kalbos failus. Komponentai organizuojami ir jungiami atominio dizaino principu. Galima peržiūrėti komponento HTML kodą, testuoti skirtingiems ekrano dydžiams. Yra nustatyti S, M, L, XL ekrano dydžiai testavimui. Pateikta išsami dokumentacija ir diegimo bei naudojimo instrukcijos.

3.3.2.,,Devbridge living styleguide“

„Devbridge“ įmonės sukurta priemonė „Devbridge styleguide“, skirta dizaino sistemos kūrimui, tobulinimui, dalinimuisi ir bendradarbiavimui. Devbridge styleguide kompiliuoja SCSS kintamuosius į CSS kodą [45]. Sistemoje galima redaguoti HTML kodą tiesiogiai sistemoje, pridėti CSS stilius komponentui bei išsaugoti redaguotą kodą. Taip pat yra galimybė sistemoje kurti naujus komponentus. Sistemoje yra nustatyti ekrano dydžiai (pikseliais) testavimui. Pateikta bazinė instaliacijos ir naudojimo informacija.

3.3.3. „Fractal“

„Fractal“, tai priemonė, kurios pagalba kuriamos, diegiamos ir dokumentuojamos komponentų bibliotekos bei dizaino sistemos. Dizaino sistemos leidžia modularizuoti interneto svetaines, suskaidant jas į nedidelius, pakartotinai panaudojamus modulius, kurie gali būti jungiami ir derinami tarpusavyje [46]. „Fractal“ gali būti konfigūruojama bei derinama atisžvelgiant dizaino sistemos vystytojo naudojamas programines priemones. Yra pateikta labai išsami dokumentacija. Nėra galimybės redaguoti tiesiogiai sistemoje – visi pakeitimai vykdomi kodo redaktoriaus pagalba. Galima rankiniu būdu didinti ar mažinti komponento atvaizdavimo langą arba atsidaryti komponentą atskirame naršyklės lange. Kartu su komponentu pateikiami su juo susieti CSS ar SCSS stilių failai. Yra galimybė peržiūrėti su komponentu susietus išteklius (CSS, SCSS ar JavaScript failus) sistemoje.

3.3.4. Dizaino sistemos kūrimo priemonių palyginimas

Palyginus pasirinktas priemones, skirtas dizaino sistemos kūrimui, matomi jų skirtumai. Pagrindiniai kriterijai, renkant sistemą, apibendrinti lentelėje (11 lentelė).

12 lentelė. Įrankių dizaino sistemai kurti palyginimas

	“Pattern Lab”	“Devbridge Styleguide”	“Fractal”
Galima peržiūrėti komponento HTML kodą sistemoje (naršyklėje)	taip	taip	taip
Galima peržiūrėti su komponentu susijusį CSS/SCSS kodą sistemoje (naršyklėje)	ne	ne	taip
Galima redaguoti komponento HTML ir CSS kodą sistemoje (naršyklėje)	ne	taip	ne
Išsami dokumentacija	taip	ne	taip
Nustatyti ekrano dydžiai prisitaikančio dizaino testavimui	taip	taip	ne

Nė viena iš priemonių pilnai neatitiko visų reikalavimų, todėl buvo renkama pagal tai, kurie reikalavimai aptariamam atveju, t.y. dizaino sistemos „Moodle“ VMA kūrimui, yra svarbiausi. „Moodle“ temos kūrimo atveju, svarbu yra matyti kurie CSS stiliai yra susiję su redaguojamu komponentu, todėl tai buvo vienas svarbiausių veiksnių pasirenkant „Fractal“ įrankį. Taip pat pasirinkimą nulėmė ir gana išsami sistemos dokumentacija.

3.4. Dizaino sistemos įrankio „Fractal“ diegimas

„Fractal“ įrankis, tai Node.js pagrindu veikianti sistema, diegiama kompiuteryje (Windows, Mac arba „Linux“ operacinėse sistemose). Diegimo instrukcija sukurta remiantis oficialia „Fractal“ svetainėje pateikiama informacija [46].

Reikalavimai sistemai. „Fractal“ įrankio diegimui reikalinga:

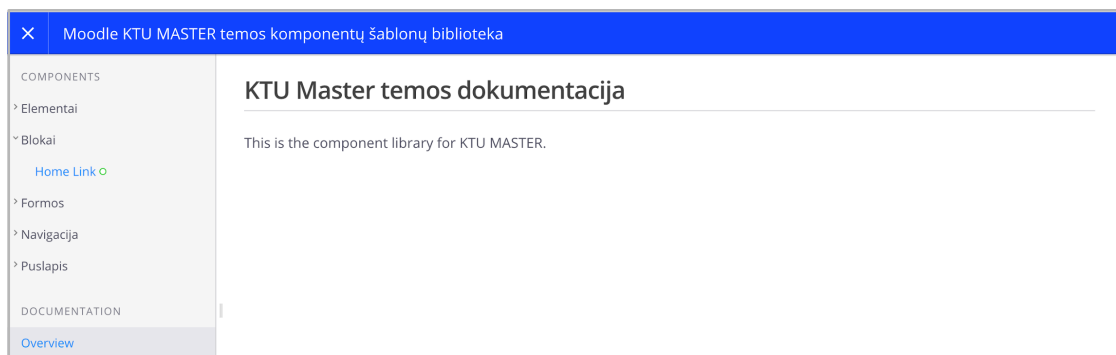
- Node.js 4.4.7 ar naujesnė versija;
- Node Package Manager (npm);
- Fractal CLI, Gulp, Webpack ar kito procesų automatizavimo įrankio.

Darbui su „Fractal“ įrankiu reikalinga komandinės eilutės (Windows) arba „Terminal“ (Mac) programa kurios pagalba bus paleidžiamas ir sustabdomas serverio veikimas, taip pat atliekami „Gulp“ veiksmai, generuojami statiniai HTML failai.

Komponentams kurti, redaguoti bei Fractal įrankio konfigūravimui būtinas kodo redaktorius. Sistemai realizuoti naudojamas „Microsoft Visual Studio Code“ kodo redagavimo programa. Procesų automatizavimui sukonfigūruotas „Gulp“ įrankis.

Diegimas:

1. „Fractal“ įrankis atsisiunčiamas iš <https://github.com/frctl/fractal>.
2. Kompiuteryje, kuriame diegiamas įrankis, sukuriama byla, į kurią nukopijuojami visi parsisiųsti failai.
3. Komandinės eilutės lange, komandos “cd” pagalba einama į sukurtą byla su „Fractal“ įrankio failais.
4. Komandinės eilutės lange įvedama komanda: “npm install --save @frctl/fractal”.
5. Procesų automatizavimui, diegiamas Gulp bei jo priklausiniai. Pagrindinėje projekto byloje sukuriamas failas gulpfile.js.
6. Komandinėje eilutėje, komandos “npm install gulp gulp-sass gulp-sass-glob gulp-plumber gulp-notify” pagalba instaliuojamas „Gulp“ bei priklausiniai.
7. Komandinėje eilutėje „gulp“ komandos pagalba paleidžiamas serverio veikimas.
8. Sistema veikia naršyklės lange suvedus adresą <http://localhost:3000> (16 pav.).

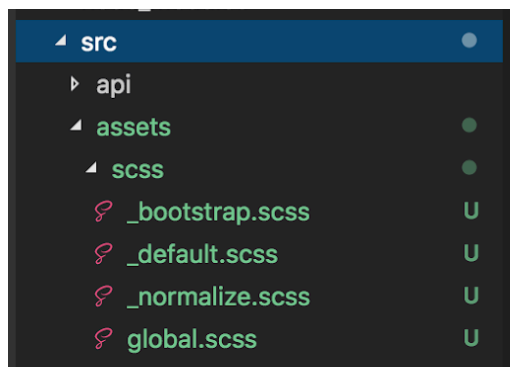


16 pav. Dizaino sistemos įrankio pradžios langas

3.5. Dizaino sistemos kūrimas „Fractal“ priemonėje

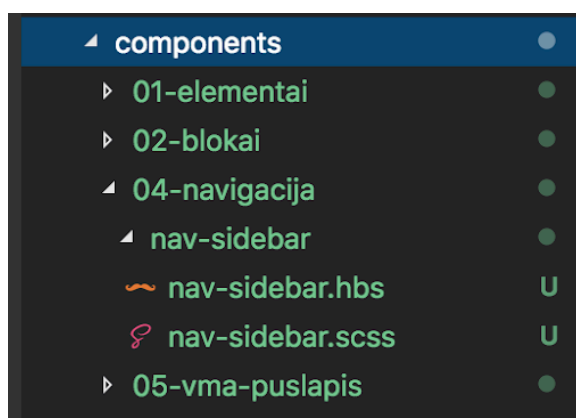
Bendrujų CSS failų įtraukimas. Byloje „./src“ sukuriama byla pavadinimu „./assets“. Šioje byloje bus talpinami visi bendrieji SCSS failai (17 pav.).

Failai, prieš pavadinimą turintys apatinį brūkšnį, yra daliniai failai ir yra importuojami į global.scss failą, bei „Gulp“ pagalba apjungiami ir transformuojami į CSS failą.



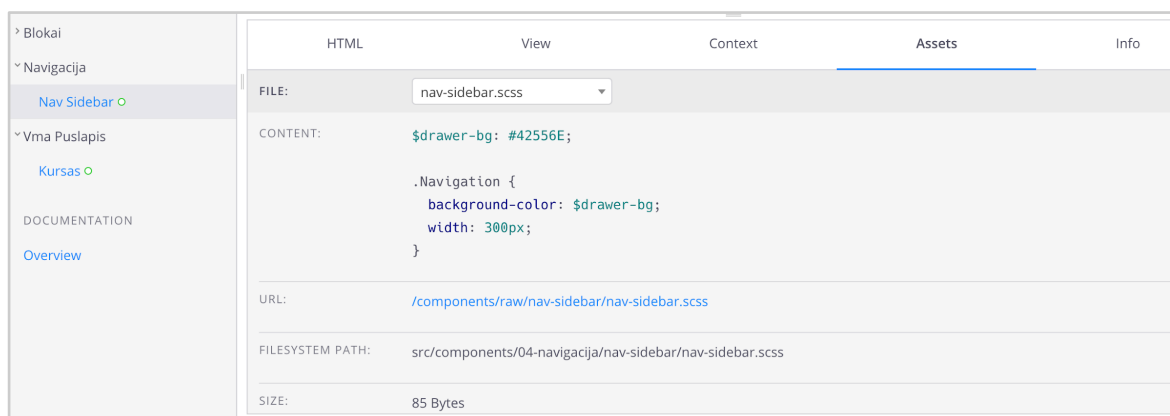
17 pav. Bylos „./src/assets“ struktūra

Komponentų kūrimas. Byloje „./src“ sukuriama „./components“ byla, kurioje bus kuriami visi dizaino sistemos komponentai. Norint sugrupuoti komponentus pagal požymius, byloje „./components“ sukuriamos sunumeruotos bylos, atitinkančios komponentų grupes. Šiose bylose kuriamos komponentų (blokų ir elementų) bylos, turinčios pavadinimus, atitinkančius komponento pavadinimą. Tam, kad su komponentu susiję ištekliai būtų atvaizduoti tinkamai, visi komponento failų pavadinimai turi sutapti su bylos pavadinimu.



18 pav. Bylos „./src/components“ struktūra

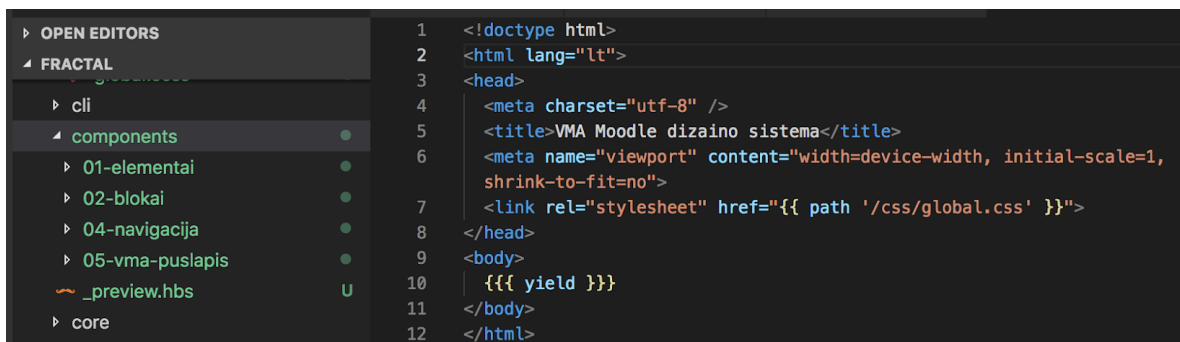
Komponentas peržiūrai pasirenkamas šoniniame navigacijos meniu. Visa su sukurtu komponentu susijusi informacija atvaizduojama dizaino sistemos lange: HTML kodas, SCSS kodas, kiti ištekliai (19 pav.). Kartu su HTML, SCSS kodu lange atvaizduojamas komponentas.



19 pav. Komponento išteklių (SCSS kodo) atvaizdavimas dizaino sistemoje

Komponentų jungimas. „Fractal“ priemonės pagalba, galima įtraukti vienus komponentus į kitus. Tokiu būdu, galima lengvai jungti elementus į blokus, įtraukti vienus blokus į kitus. Norint įtraukti sub-komponentą, reikalinga nurodyti to sub-komponento pavadinimą komponento HTML kode. Mygtuko, pavadinimu „button“ įtraukimas į komponentą atliekamas deklaruojant `{{> @button }}` HTML kode.

Puslapių šablonų kūrimas. Fractal priemonėje sugeneruotas komponentas yra HTML kodo fragmentas. Norint pateikti komponentą peržiūrėti puslapio kontekste, sukuriamas puslapio šablonas, kuriame bus atvaizduojami komponentai (19 pav.). Sukuriamas puslapio šablonas, kurio HTML kode `<body>` žymėje nurodoma `{{{ yield }}}` deklaracija, žyminti vietą, kurioje turi būti įterptas atvaizduojamas komponentas.



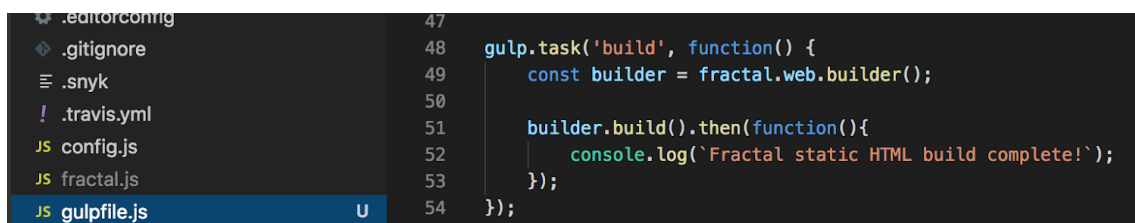
```
1 <!doctype html>
2 <html lang="lt">
3 <head>
4   <meta charset="utf-8" />
5   <title>VMA Moodle dizaino sistema</title>
6   <meta name="viewport" content="width=device-width, initial-scale=1,
7     shrink-to-fit=no">
8   <link rel="stylesheet" href="{{ path '/css/global.css' }}">
9 </head>
10 <body>
11   {{{ yield }}}
12 </body>
13 </html>
```

20 pav. Komponento puslapio šablonas

Pagrindinė „Gulp“ funkcija paleidžiama komandinėje eilutėje suvedus komandą „gulp“ kuri atlieka keletą funkcijų:

- paleidžia serverio „localhost“ veikimą;
- stebi pakeitimus komponentų failuose ir automatiškai perkrauna naršyklę jiems pasikeitus;
- kompiluoja .scss į .css failus.

Dizaino sistemos eksportas (angl. *build*). Eksporto procese vykdomas statinių HTML, CSS ir JavaScript failų generavimas. Sugeneruoti failai talpinami „/build“ byloje. Generavimo procesas atliekamas „Gulp“. Konfigūruojamas failas „gulpfile.js“, kuriame aprašoma „gulp build“ funkcija (21 pav.), atliekama iš komandinės eilutės.



```
47
48 gulp.task('build', function() {
49   const builder = fractal.web.builder();
50
51   builder.build().then(function(){
52     console.log('Fractal static HTML build complete!');
53   });
54 });
```

21 pav. „Gulp“ eksporto proceso konfigūravimo funkcija

```
Karoliss-MacBook-Pro-2:fractal Karolis$ gulp build
[14:54:52] Using gulpfile ~/Sites/fractal/gulpfile.js
[14:54:52] Starting 'build'...
Fractal static HTML build complete!
```

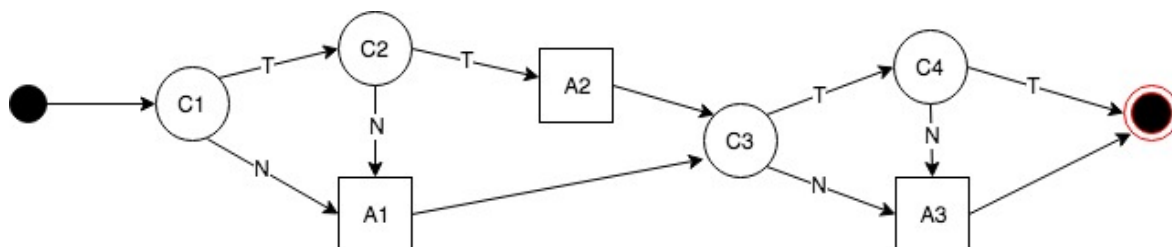
22 pav. „Gulp build“ eksporto procesas

„Gulp build“ funkcija paleidžiama komandinėje eilutėje suvedus „gulp build“ komandą (22 pav.). Sugeneruoti failai iš sistemos „/build“ bylos FTP ar SSH protokolo pagalba gali būti perkelti į pasirinktą serverį. Dizaino sistema veikia be duomenų bazės. Pakeitus ar papildžius komponentus, „build“ procesą galima atlikti iš naujo, pasikeitusius failus serveryje.

3.5.1. „Moodle“ naudotojo sąsajos komponentų kūrimas

Projekto metu kuriama „Moodle“ mokymosi valdymo sistemos dizaino sistema, paremta pagrindinės temos „Boost“ savybėmis ir paveldinti šios temos nustatymus. Todėl, yra nagrinėjamas „Boost“ temos komponentų atitikimas pasiekiamumo gairėms, bei teikiami pasiūlymai ir kuriamas patobulintas komponentų dizainas, labiau atitinkantis pasiekiamumo standartus.

Komponentai dizaino sistemos aplinkoje kuriami iš atskirų HTML ir CSS kodo fragmentų, susijusių tik su šiuo elementu. Komponentui taip pat taikomi ir globalūs stiliaus nustatymai. Sistemoje gali būti kuriami nauji, bei tobulinami anksčiau sukurti komponentai.



23 pav. Komponentų kūrimo kontekstinis grafas

13 lentelė. Komponentų kūrimo kontekstinio grafo mazgų ir veiksmų detalizacija

Sąlyga	Detalizacija	Veiksmai	Detalizacija
C1	Ar komponento HTML atitinka jam taikomas pasiekiamumo gaires?	A1	Sukuriamas HTML kodas komponentui
C2	Ar tėvinės temos komponento HTML struktūra tinkama?	A2	Panaudojama tėvinės temos komponento HTML struktūra
C3	Ar tėvinės temos komponento CSS stilius atitinka pasiekiamumo gaires?	A3	Komponentui perrašomas CSS stilius
C4	Ar numatytasis tėvinės temos komponento CSS stilius tinkamas?		

Kontekstiniame grafe (23 pav.) pavaizduota komponentų kūrimo paveldinčiai temai seka. Pirmiausia atsižvelgiama į tėvinės temos komponento atitikimą pasiekiamumo gairėms bei tinkamumą kuriamam mokymosi sistemos dizainui.

3.6. Navigacijos komponentai

Galimybė sąveikauti su naudotojo sąsaja bei naršyti tinklalapyje tarpusavyje susijusių nuorodų pagalba susieja sistemos HTML puslapius bei sukuria interaktyvumą. Todėl galimybė naršyti ir judėti navigacijos bei nuorodų pagalba yra viena svarbiausių internetinės svetainės savybių. Tačiau,

kaip jau buvo apžvelgta, naudotojų naršymo būdai bei galimybės skiriasi. Su naudotojo sąsaja galima sąveikauti pelės, klaviatūros, kito įvesties įrenginio ar lietimio pagalba. Naudojimosi scenarijai, naudojantis skirtingomis priemonėmis, taip pat skiriasi.

3.6.1. Naršymas pele

Naršymas pelės pagalba yra suvokiamas kaip standartinis ir dažniausiai naudojamas naršymo būdas. Kompiuterinės pelės pagalba, naudotojas lengvai gali judėti ir pasiekti peržiūrimo tinklalapio interaktyvius elementus, taip pat pataikyti net ir į labai smulkius grafinius elementus. Dauguma naudotojo sąsajos elementų yra pirmiausia projektuojami standartiniam naudotojui, sąveikaujančiam su naudotojo sąsaja pelės pagalba.

3.6.2. Naršymas klaviatūra ir fokusavimas

Naršant klaviatūros pagalba, naudojamosi klavišų iššaukiamomis komandomis. Pagrindinis būdas judėti per puslapio interaktyvius elementus yra TAB klavišo pagalba fokusuojami (angl. *focus*) aktyvūs HTML elementai. Šį naršymo būdą dažniausiai pasitelkia naudotojai, turintys motorinių (laikinių ar nuolatinių) sutrikimų bei asmenys su regos sutrikimais, kartu pasitelkiantys ir balso asistento priemonę.

Tam, kad naršymas būtų efektyvus ir neklaidintų, naršyklėse nustatytas elementų fokusavimo stilius, kurio pagalba fokusuotas elementas paryšklinamas apvadu, žyminčiu jo būseną ir aiškiai išskiriančiu jį iš kitų elementų. Naršymas vyksta linijiniu būdu, einant iš eilės HTML kodo pateikimo tvarka, per visus interaktyvius tinklalapio elementus. Skirtingos naršyklės naudoja skirtingus fokusavimo stilius, tad siekdami suvienodinti bei pritaikyti savo kuriamam dizainui, nemažai sąsajos kūrėjų CSS pagalba išjungia ar pakeičia fokusavimo funkciją.

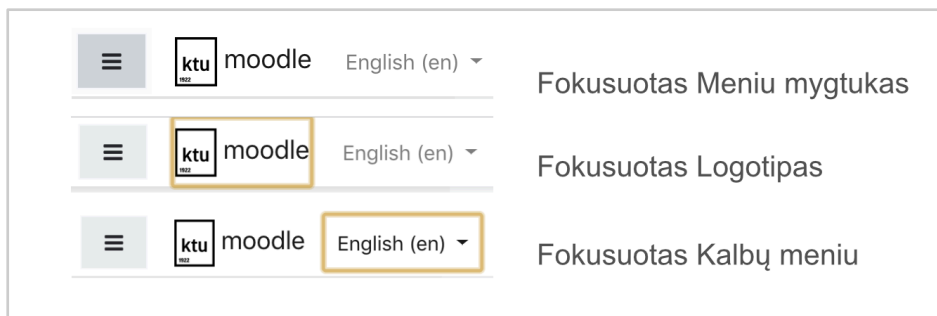
„Moodle“ mokymosi sistemos „Boost“ temoje fokusavimo stilius yra pakeistas, tačiau yra keletas esminių trūkumų:

- fokusavimo stilius pakeistas ne visiems, o tik kai kuriems elementams (24 pav.);
- pakeistas fokusavimo stilius yra beveik nepastebimas (lengvai pasikeičia fokusuoto elemento fonas), kai tuo tarpu pagrindinis reikalavimas fokusavimui yra tai, kad fokusuotas elementas išsiskirtų iš aplinkos (24 pav.).

Fokusavimo stiliaus nenuoseklumas yra sąlygotas CSS stiliaus, kuris pašalina naršyklėje numatytą apvadą nuo mygtukų:

```
.btn:focus, .btn.focus {  
  
    outline: 0;  
  
}
```

Kiti interaktyvūs elementai, tuo tarpu išlaiko naršyklėje numatytąjį stilių. „WebAIM“, organizacija, tirianti žiniatinklio pasiekiamumą, nurodo vengti „outline: 0“ ar panašių CSS stiliaus deklaracijų, kadangi tai reikšmingai pablogina tinklalapio pasiekiamumą [47].



24 pav. „Boost“ temos interaktyvių elementų fokusavimas

Siekiant pagerinti sistemos pasiekiamumą bei naudotojų, naršančių klaviatūra patirtį, kuriame dizaino sistemoje, nuspręsta suvienodinti fokusavimo stilių visiems interaktyviems elementams. Pasirinktas su fonu kontrastuojančios spalvos vientisas apvadas (25 pav.).



25 pav. Dizaino sistemos interaktyvių elementų vieningas fokusavimo stilius

Fokusavimo stilius nustatomas visiems elementams, universaliojo identifikatoriaus „*“ pagalba, nurodant pseudo-identifikatorių „:focus“. CSS kalbos specifiškumo taisyklės nustato, kad universaliojo identifikatoriaus taisyklės taikomos tik tada, jei nėra kitų, labiau specifinių taisyklių. Todėl, siekiant, kad fokusavimo stiliaus deklaracija paveiktų tinklalapio HTML elementus, prie deklaracijos nurodoma „!important“ svarbos žymė (26 pav.).

```

*:focus {
  outline: 2px solid #CED33A !important;
}

```

26 pav. Fokusavimo deklaracija naudojant universalųjį identifikatorių

Komponento CSS stiliaus pakeitimai, atliekami klasių identifikatoriaus pagalba, nepasiekia tikslo, kadangi Boost temoje CSS stilių failuose plačiau naudojami ID identifikatoriai, kurie turi aukštesnį specifiškumą nei klasės identifikatoriai. Išspręsti šią problemą gali būti „!important“ deklaracija, naudojama šalia CSS savybės reikšmės, tačiau pagal CSS metodikų nuostatas, tai laikoma vengtina praktika.

3.6.3.Šoninė navigacija

Šoninė navigacija „Moodle“ naudojama kaip nuorodų sąrašas navigacijai tarp kursų, bei navigacijai tarp kurso skyrių. „Boost“ temoje šis šoninis meniu pateikiamas tokiu HTML kodu:

```
<nav>
  <a href="...">...</a>
  <a href="...">...</a>
  <a href="...">...</a>
  ...
</nav>
```

Tokia navigacija tinkama naudotojams, naršantiems įprastu būdu, tačiau gali sukelti nepatogumų asmenims su specialiaisiais poreikiais, naudojantiems papildomą įgarsinimo įrangą ar virtualų asistentą. Įsijungus „Voice Over“ ir pasiekus šoninį navigacijos meniu, įgarsinimo priemonė iš karto vienas po kito pradeda skaityti visi nuorodų pavadinimus. Nėra aišku, kiek nuorodų yra pateikta meniu sąrašė. Tai ypač pablogina asmenų su regėjimo negalia naršymo patirtį.

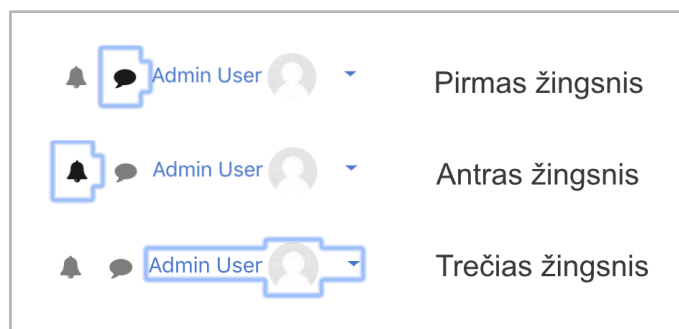
Kuriamoje dizaino sistemoje pateikiamas toks šoninio meniu HTML kodas:

```
<nav>
  <ul>
    <h3 hidden>Meniu pavadinimas</h3>
    <li><a href="...">...</a></li>
    <li><a href="...">...</a></li>
    <li><a href="...">...</a></li>
  </ul>
</nav>
```

Navigacijos pateikimas nenumeruoto sąrašo pavidalu suteikia galimybę naršant balso asistento pagalba išgirsti kiek meniu punktų yra sąrašė. Testuojant su „Voice Over“ priemone, fokusavus pirmąjį meniu esantį sąrašo punktą, balso asistentas praneša: „Nuoroda. Sąrašas. 6 vienetai“. Tokiu būdu balso asistento pagalba naršantis naudotojas žino, kokio dydžio yra fokusuotas meniu sąrašas.

3.6.4.Navigacijos seka

„WCAG“ pasiekiamumo gairėse [48] nurodyta, kad navigacijos bei skaitymo seka (nulemta kodo sekos) yra logiška ir intuityvi. „Boost“ temoje bandant naviguoti klaviatūros TAB klavišo pagalba, iš eilės nuo viršutinio kairiojo krašto yra fokusuojami interaktyvūs elementai: meniu atvėrimo/užvėrimo mygtukas, nuoroda į pradinį puslapį, kalbų perjungimo meniu. Tačiau, pasiekus naudotojo profilio navigaciją, pirmiausiai fokusuojamas žinučių mygtukas, po to grįžtama atgal prie pranešimų mygtuko (27 pav.). Tai reiškia, kad puslapio HTML yra išdėstytas tokia tvarka, kokia fokusuojami elementai, o vizualus išdėstymas pakeistas CSS pagalba.



27 pav. Fokusavimo judėjimo tvarka „Boost“ viršutiniame meniu

Su TAB mygtuku praėjus visą viršutinį meniu, fokusavimo žymė nepapuola ant šoninio meniu, nors pagal vizualią skaitymo tvarką (iš kairės į dešinę) tai atrodytų kaip logiška navigacijos seka. Užuoat patkus į šoninį meniu, fokusuojami pagrindiniame (kurso turinio) lange esantys elementai ir nuosekliai judant kurso turiniu, pasiekiami šoniniai dešinės pusės pagalbiniai blokai (pvz. kalendorius, informaciniai skydeliai). Tik praėjus visą kurso puslapio turinį iki apačios, peršokama į šoninę kairiąją navigaciją. Taip yra todėl, kad „Boost“ temos HTML struktūroje šoninės navigacijos komponentas seka po pagrindinio turinio komponento.

Esama „Boost“ navigacijos pateikimo struktūra HTML kode:

```
<nav class="navbar"> ... </nav>      – viršutinis meniu
<div id="page"> ... </div>          – kurso pagrindinis turinys
<div id="nav-drawer"> ... </div>    – šoninės navigacijos skydelis
```

Šoninė navigacija „Moodle“ sistemoje naudojama naudotojo kursų sąrašui pateikti bei kurso nuorodoms į įverčių, pasiekimų, kalendoriaus posistemas ir yra dažniausiai naudojama navigacijai tarp skirtingų kursų ar to paties kurso skyrių. Puslapyje šoninė navigacija CSS pagalba vizualiai išdėstyta kairiajame šone, tačiau norint ją pasiekti TAB pagalba, reikia judėti nuosekliai per visą puslapio HTML kodą tol, kol pasiekiamas šoninės navigacijos meniu.

Siekiant padaryti naršymo klaviatūra patirtį patogesne, galimi keli sprendimai:

- perkelti šoninės navigacijos komponentą prieš pagrindinio turinio komponentą ir išlaikyti logišką HTML elementų seką;
- prieš kurso pagrindinio turinio komponentą pateikti tik naršantiems klaviatūra matomą greitąją nuorodą (ang. *skip link*), nukreipiančią į šoninės navigacijos elementą.

„WebAIM“ straipsnyje analizuojamos naršymo klaviatūros pagalba problemos, galimi sprendimo būdai. Vienas iš nagrinėjamų pavyzdžių yra navigacijos pateikimas po pagrindinio turinio, siekiant sumažinti TAB mygtuko paspaudimų kiekį. Tačiau iškyla kitas klausimas: ar toks pateikimo būdas reiškia, kad reikia pateikti greitąją nuorodą, sutikeikiančią galimybę vienu paspaudimu pereiti prie navigacijos. „WebAIM“ straipsnyje pabrėžiama, kad greitosios nuorodos, nukreipiančios į navigaciją nėra įprastas reiškinys ir gali sutrikdyti naudotoją [49].

Remiantis minėtame straipsnyje išsakyta nuomone [49], dizaino sistemoje šoninė navigacija išdėstoma prieš pagrindinį turinį, pateikiant greitąją nuorodą „Į pagrindinį turinį“:

`<nav class="navbar"> ... </nav>` – viršutinis meniu

`<div>Į pagrindinį turinį </div>`

`<div id="nav-drawer"> ... </div>` – šoninės navigacijos skydelis

`<div id="page"> ... </div>` – kurso pagrindinis turinys

3.6.5. Greitosios nuorodos

Norėdamas pasiekti norimą vietą puslapyje, naudotojas, naršantis klaviatūros pagalba, juda pirmyn ar atgal per interaktyvius elementus TAB pagalba. Siekiant sutrumpinti kelią iki norimos vietos puslapyje, HTML kode pateikiamos greitosios nuorodos, leidžiančios naudotojui peršokti į tam tikrą nurodytą elementą. Greitosios nuorodos pateikiamos prieš komponentą, kurį galima praleisti, ir CSS pagalba paslepiamos taip, kad būtų matomos tik naršantiems klaviatūros ar balso asistento pagalba.

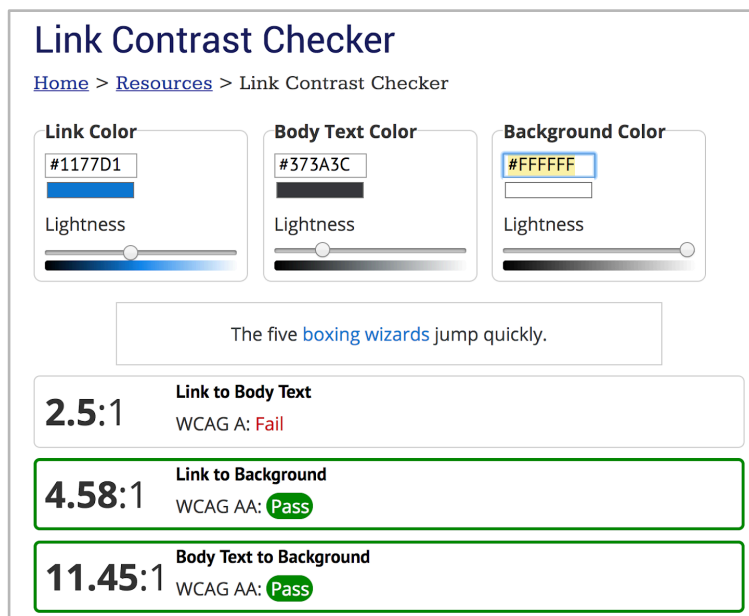
Greitosios nuorodos „Boost“ temoje naudojamos praleisti viršutinį navigacijos meniu bei papildomus informacinius skydelius („Kalendorius“, „Naujausia veikla“ ir pan.), yra pateiktos tinkamai ir padeda naudotojui greitai peršokti į kitą puslapio vietą. Dizaino sistemoje papildomai pateikiama šoninės navigacijos greitoji praleidimo nuoroda.

3.7. Tekstas ir spalvų kontrastas

Virtualiojoje mokymosi aplinkoje viena iš pagrindinių dalių yra mokymosi medžiagos pateikimo posistemė. Mokymosi turinys yra pateikiamas kursais, kuriuose vyksta įvairi su pateikto kurso turiniu susijusi mokymosi veikla. Mokymosi medžiaga pateikiama įvairiomis formomis: tekstine, vaizdo, garso bei nuorodomis.

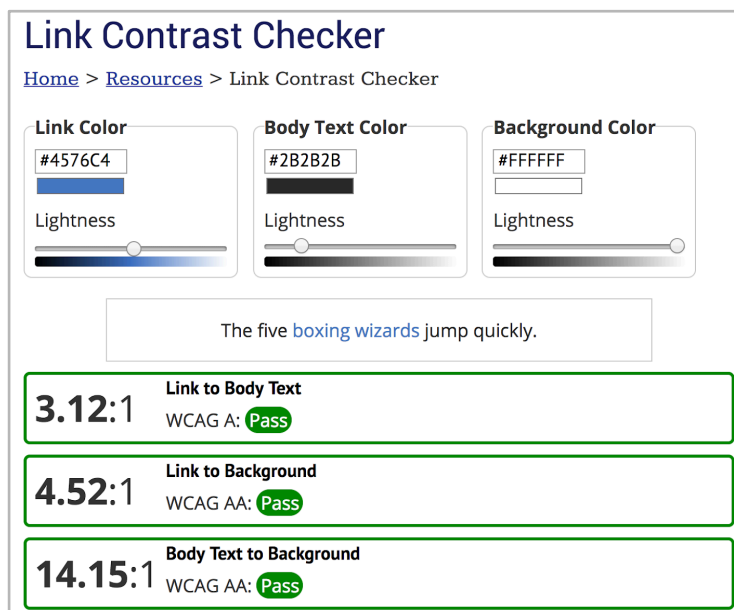
Pagrindinis `<body>` elemento teksto dydis, kurį paveldi ir kiti tekstiniai, paragrafų elementai, „Boost“ temoje yra nustatytas 0.9375rem, kas atitinka 15 pikselių teksto dydį, tarpas tarp eilučių yra 1,5. Tarpas tarp paragrafų yra 1rem (16px). Šie teksto dydžio nustatymai pilnai atitinka tekstui keliamus pasiekiamumo vertinimo kriterijus.

Spalvų kontrasto pasiekiamumui patikrinti, naudojamas „WebAim Color Checker“ spalvų pasiekiamumo testavimo įrankis. Teksto ir nuorodų spalvos su aplinkiniu fonu santykis turi būti lygus ar didesnis negu 4,5:1 tam, kad tenkintų WCAG AA lygio standartą. „Boost“ temos standartinių nustatymų teksto ir nuorodų spalvų kontrastas su fonu yra atitinkamai 11,45:1 ir 4,58:1. Tai atitinka AA lygio standartą. Tačiau, nuorodų ir teksto spalvų kontrastas nėra pakankamas ir neužtikrina pasiekiamumo A lygio standarto (28 pav.). Optimaliai naudotojo patirčiai bei pasiekiamumui užtikrinti, nuorodos turi būti pabrauktos. Jei nuorodos nėra pabrauktos ir išskiriamos tik spalvos pagalba, teksto ir nuorodų tarpusavio spalvų kontrastas turi būti ne mažesnis nei 3:1.



28 pav. „Boost“ temos teksto, nuorodų ir fono kontrasto testas

Temos pasiekiamumo pagerinimui, dizaino sistemoje teksto ir nuorodų spalvų nuostatos buvo pakoreguotos taip, kad atitiktų AA lygio reikalavimus (29 pav.).



29 pav. Pakoreguoto teksto, nuorodų ir fono kontrasto testas

Derėtų atkreipti dėmesį, kad mokymosi valdymo sistemos „Moodle“ administratoriai tu galimybę keisti temos spalvų gamą „Boost“ temos ruošiniais. Keičiant temos teksto, nuorodų ir fono nustatymus, labai svarbu pasitikrinti planuojamų pakeitimų atitikimą pasiekiamumo standartams, bei siekti, kad visi parametrai atitiktų A arba AA pasiekiamumo gairių lygį.

3.8. Semantinė puslapio HTML struktūra

„WCAG“ gairėse nurodoma, kad tinklalapio dalims turi būti naudojamas semantinis prasmingas žymėjimas, naudojami specialios HTML5 bei ARIA žymės. „Boost“ temoje pagrindinių puslapio dalių žymėjimas HTML yra:

```

<nav> ... </nav>                – viršutinė navigacija
<div id="page">                  – pagrindinė puslapio dalis
  <header id="page-header">      – antraštinė dalis
  ...
  </header>
</div>

```

Iš pateikto pavyzdžio matoma, kad yra semantinis žymėjimas panaudotas viršutinei navigacijai, pagrindinės dalies antraštinei daliai, tačiau nėra pagrindinei puslapio daliai nėra panaudota `<main>` žymė. Taip pat žymėjime nėra naudojami ARIA „role“ atributai.

Siūloma pakeisti puslapio struktūros HTML žymėjimą į:

```

<nav role="navigation"> ... </nav>                – viršutinė navigacija
<main role="main" id="page">                      – pagrindinė puslapio dalis
  <header role="banner" id="page-header">          – antraštinė dalis
  ...
  </header>
</main>

```

Panaudotos HTML5 žymės atskiria kodo sritis pagal vaidmenis ir suteikia daugiau prasmingumo negu `<div>` konteinerio žymė.

3.9. Prisitaikančio dizaino realizavimas

Prisitaikantis puslapio dizainas realizuojamas identifikuojant naudotojo ekrano plotį ir pritaikant naudotojo sąsają nurodytam režiuui. Lūžio taškai (angl. *breakpoint*) nustatomi dažniausiai CSS pagalba, pateikiant užklausą apie ekrano plotį ir nustatant elemento pateikimo taisykles:

```

@media(min-width: ekrano dydis pikseliais arba santykiniais vienetais) {
  ... (taisyklės šiai parinkčiai)
}

```

„Boost“ temoje naudojami „Bootstrap4“ karkaso lūžio taškai:

- mažiausia reikšmė yra 576 px (mobilus telefonas);
- vidutinė reikšmė – 768 px (planšetinis kompiuteris);
- didelė – 992 px;
- labai didelė 1200 px.

Iš „GlobalStats“ pateiktos statistikos matome, kad naudotojų ekranai ne tik mažėja, bet ir didėja (didėja pikselių kiekis), 32,54% naudotojų naršė ekranuose, didesniuose nei 1536px. Reikia pažymėti, kad naudotojų ekranai taip pat gali skirtis ir pikselių tankiu bei kitais parametrais. Tai reiškia, kad kuriant lanksčias sistemas, dydžius dizaino sistemoje siūloma nustatyti ne absoliučiais (pikseliais), o santykiniais dydžiais (pvz. em, rem).

Nustatyti tokie lūžio taškai, paliekant mažo dydžio taškus nepakeistus, koreguojant didelės raiškos reikšmes bei paverčiant jas į santykinius dydžius:

- mažiausia reikšmė yra 36rem (mobilus telefonas);
- vidutinė reikšmė – 48rem (planšetinis kompiuteris);
- didelė – 72rem;
- labai didelė – 96rem.

Nustatyti lūžio taškai santykiniais „rem“ vienetais, kurie priklauso nuo pagrindinio <html> elemento nustatymų.

3.10. Dizaino sistemos elementų perkėlimas į „Moodle“ temos šabloną

„Moodle“ tema sukuriama byloje „/theme“ sukuriant bylą naujos temos pavadinimu. Kuriamos temos failų nustatymai atlikti pagal „Moodle“ puslapyje pateiktą dokumentaciją [50].

Sukurtame „config.php“ temos konfigūravimo faile nurodomas kuriamos temos pavadinimas. Pasirinktas pavadinimas yra „Master“. Taip pat nurodoma, kad ši tema paveldės „Boost“ temos nustatymus.

```
$THEME->name = 'master';
```

```
$THEME->parents = ['boost'];
```

Byloje „/templates“ pateikiami temos šablonai .moustache failų formatu, kuriuose pateikiamas komponentų HTML kodas. Kuriant paveldinčią temą, galima pasirinkti norimą pakeisti komponentą pagrindinėje temoje ir nukopijuoti jį tuo pačiu pavadinimu į paveldinčios temos „/templates“ bylą. Naujai sukurtame komponento šablone galima perkelti HTML kodą iš dizaino sistemos, paliekant specialius „Moustache“ intarpus, išskirtus dvigubais ar trigubais riestiniais skliaustais.

Byloje „/scss“ sukuriamas „pre.scss“ failas, kuriame pateikiamas CSS kodas, einantis prieš paveldimą „tėvinės“ temos CSS kodą, ir „post.scss“ failas, kuriame pateikiamas CSS kodas, einantis po paveldimo kodo. Siekiant išlaikyti modulinę struktūrą, dizaino sistemos SCSS failai perkeliama „/scss“ byla ir importuojami į „post.scss“ failą pateikus deklaraciją @import bei dalinio failo pavadinimą.

Iš dizaino sistemos kopijuojamas komponento HTML kodas, perkeliama į atitinkamą komponento temos šabloną. Sukurtos naujoms komponento klasėms taikomas iš dizaino sistemos perkeltas CSS kodas. Sistemos tvariam veikimui užtikrinti, paliekamos ir pagrindinėje temoje taikomos komponento klasės bei ID, kadangi nuo jų gali priklausyti pagrindinėje temoje esančio JavaScript kodo veikimas.

3.11. Temos atnaujinimas kartu su „Moodle“ sistemos atnaujinimu

Kaip ir bet kuri prižiūrima bei techniškai palaikoma sistema, „Moodle“ nuolat pateikia programinius atnaujinimus, kurie neišvengiami norint atitikti saugumo, atsinaujinančių naršyklių, naudotojų reikalavimus. Tiriamojo projekto metu, buvo atliktas „Moodle“ atnaujinimas iš 3.2 versijos į 3.6 versiją ir tiriami su temos dizainu susiję pokyčiai.

Atnaujinus „Moodle“ 3.2 į 3.6 versiją, atsinaujino ir „Boost“ tema, kurios pagrindu sukurta paveldinčioji „Master“ tema. Vienas pagrindinių pakeitimų temoje buvo „Bootstrap4 alpha“ karkaso versijos pakeitimas į „Bootstrap4“, kas sąlygojo tam tikrų naudotų klasių pavadinimų pakeitimus.

„Bootstrap“ karkase yra sukurtos pagalbinės (angl. *utility*) klasės, atliekančios paprastas funkcijas. „Boost“ temoje plačiai naudojamos m- ar p- klasės, kurios prideda paraštes ir įtraukas. Pavyzdžiui, „mb-3“ klasė atlieka apatinės paraštės funkciją:

```
.mb-3 { margin-bottom: 1rem!important; }
```

„Bootstrap 4 alpha“ versijoje naudojamos „m-b-“, „m-t-“, „m-r-“, „m-l-“ klasės „Bootstrap 4“ karkaso versijoje pakeistos į „mb-“, „mt-“, „mr-“ ir „ml-“ klases. Taikant šias pagalbines klases vienai funkcijai, atsinaujinus karkasui, tenka keisti visuose HTML šablonuose. Užuot naudojus karkaso klases, kurios atlieka vienintelę nedidelę funkciją, optimaliau sukurti SCSS kintamąjį, nustatantį paraštės dydį (pvz.: \$margin: 1rem;) ir jį taikyti komponento SCSS kode.

Atnaujinimo metu „Boost“ temoje pakeistas „header.mustache“ šablonas į „navbar.mustache“. Siekiant, kad tema veiktų tinkamai, paveldinčioje temoje turėjo būti atlikti atitinkami minėtų šablonų pakeitimai.

Apibendrinant galima teigti, kad sistemos atnaujinimai neišvengiamai susiję ir su naudotojo sąsaja formuojančios temos peržiūrėjimu, atnaujinimu bei suderinimu su „Moodle“ sistemos funkcijomis. Tai reiškia, kad naudotojo sąsajos dizaino kūrimas yra cikliškas procesas, derinantis prie virtualios mokymosi sistemos veikimo principų, nuolat vykstančių technologinių pokyčių ir besikeičiančių bei labai įvairių naudotojų poreikų.

3.12. Išvados

1. „Moodle“ temos šablono naudotojo sąsajos projektavimui galima pritaikyti CSS metodikos taisykles.
2. Remiantis BEM metodika bei siekiant kuriamą dizaino sistemą priartinti prie Moodle terminologijos, projektuojamoje sistemoje komponentai turėtų būti skirstomi į blokus ir elementus.
3. Dizaino sistemos realizavimui pasirinkta „Fractal“ sistema dėl išsamios dokumentacijos bei galimybės CSS failus atvaizduoti sistemoje kartu komponentu.
4. Naudotojo sąsajos pasiekiamumas gali būti ženkliai pagerintas nedaug pakeitus puslapio HTML kodą, jo išdėstymą ar CSS stiliaus nuostatas.
5. Sukūrus paveldinčią „Moodle“ temą, reikalinga nuolatinė paveldinčiosios temos techninė priežiūra ir atnaujinimas pagal naujus „tėvinės“ temos nustatymus, todėl tai gali pareikalauti nemažai resursų iš organizacijos, plėtojančios paveldinčią „Moodle“ temą.
6. „Bootstrap“ karkaso naudojimas temoje pagreitina bei temos kūrimo procesą padaro paprastesniu, tačiau karkaso atnaujinimas gali įtakoti komponentų atvaizdavimą. „Bootstrap“ karkaso naudojimas gali pasitarnauti pakankamai vieningam sistemos komponentų, ypač

naudojant įvairius papildinius, atvaizdavimui. Specifinėms komponentų atvaizdavimo taisyklėms patartina naudoti CSS stiliaus nuostatas, pritaikytas konkretaus komponento identifikatoriui, bei SCSS kintamuosius, užuot pridėjus papildomą bendrinę karkaso klasę HTML elementui.

Išvados

1. Vieno iš VMA dalyvių - besimokančiojo galimybes veikia daug aplinkybių: technologinės aplinkybės, naudotojo fiziniai ar kognityviniai poreikiai, situacijos, kurių metu naudotojas naudojami sistema. Visa tai apibendrintai galima pavadinti naudotojų poreikiais. Šių poreikių visuma yra labai plati, tad netikslinga kurti atskirų posistemių, pritaikytų tam tikriems specialiesiems poreikiams ar technologijoms. Prioritetu turėtų būti laikoma universalios sistemos, tenkinančios įvairius naudotojų poreikius, kūrimas, vadovaujantis suvokiamumo, valdomumo, aiškumo bei universalumo principais.
2. Priklausomai nuo virtualiosios mokymo aplinkos naudojimo pobūdžio, sistema turi skirtingas posistemas, kurioms naudotojų funkciniai ir nefunkciniai reikalavimai gali skirtis. Vienų posistemių veikla labiau susijusi su informacijos suvokimu, peržiūra (turinio pateikimo posistemė), kitų – su interaktyvia ir dėmesio reikalaujančia veikla (vertinimas). Tai nulemia skirtingus naudotojo sąsajos pateikimo ir pritaikymo būdus.
3. Moduliari „Moodle“ struktūra leidžia papildyti mokymosi valdymo sistemą papildiniais. Viena iš papildinių rūšių yra naudotojo sąsają modeliuojanti tema, kurios pagalba galima pakeisti aplinkos išvaizdą bei pritaikyti sąsają naudotojo poreikiams. „Moodle“ sistemos architektūra palanki sąsajos kūrimui ir suteikia galimybę išnaudoti pagrindinės temos funkcines galimybes bei nustatymus kuriant paveldinčią sąsajos temą. Vis dėlto, temos keitimas ar kūrimas tiesiogiai „Moodle“ sistemoje – sudėtingas ir daug laiko sąnaudų reikalaujantis procesas. Siekiant šį procesą padaryti paprastesniu, nuspręsta kurti bei saugoti sistemos komponentus specializuotoje dizaino sistemos kūrimo priemonėje.
4. Virtualiosios mokymosi aplinkos komponentams pritaikyti CSS metodologijos principai, kurie leidžia kurti „Moodle“ naudotojo sąsajos dizainą nenaudojant perteklinių CSS karkaso klasių HTML kode bei užtikrinant sąsajos komponento stiliaus nuostatų inkapsuliaciją. Tai padeda kurti semantinį bei geriau pasiekiamumo standartus atitinkantį HTML kodą.
5. Įvertinus virtualiosios mokymosi aplinkos „Moodle“ prisitaikančio dizaino temą „Boost“ nustatyta naudotojo sąsajos pasiekiamumo trūkumų, kuriais galima ištaisyti bei ženkliai pagerinti naudotojo sąsajos pasiekiamumą koreguojant HTML kodą ir CSS stiliaus nuostatas. Realizuojant modulinę sistemą įsitikinta, kad dizaino sistema tarnauja kaip aplinka komponentų kūrimui, testavimui, pasiekiamumo tobulinimui.
6. Dizaino sistemos komponentai perkeliama į „Moodle“ aplinkos temą paverčiant komponentus temos šablonais bei importuojant stiliaus failus į temos failų bylą. Projekto įgyvendinimo metu nustatyta, kad dizaino sistema palengvina „Moodle“ temos kūrimą bei pritaikymą prie „Moodle“ sistemos atnaujinimo metu pakeistos pagrindinės temos parametrų.

Literatūra

1. TARGAMADZĖ, Aleksandras. *Technologijomis grįsto mokymosi priemonės ir sistemos. Mokomoji knyga*. Kaunas: Technologija, 2011. [žiūrėta 2018-01-16]. e. ISBN 978-609-433-074-2. Prieiga per: <https://www.ebooks.ktu.lt>
2. Lietuvos Respublikos terminų bankas [interaktyvus]. [žiūrėta 2018-01-16]. Prieiga per: <http://terminai.vlkk.lt/pls/tb/>
3. Nielsen, Jacob. *Introduction to usability* [interaktyvus]. 2012.01.04. [žiūrėta: 2018-01-18]. Prieiga per: <https://www.nngroup.com/articles/usability-101-introduction-to-usability/>
4. RAMAKRISNANA, Prasanna et al. Evaluation of User Interface Design for Learning Management System (LMS): Investigating Student's Eye Tracking Pattern and Experiences. *Procedia - Social and Behavioral Sciences*, 2012, 67. Prieiga per: doi: <https://doi.org/10.1016/j.sbspro.2012.11.357>
5. *Desktop vs Mobile vs Tablet Market Share Worldwide* [interaktyvus]. [žiūrėta: 2018-01-16]. Prieiga per: <http://gs.statcounter.com/platform-market-share/desktop-mobile-tablet>
6. LAPUH Bele, Julija et al. User and mobile-friendly Learning Management System design [interaktyvus]. [žiūrėta: 2018-01-16]. *2015 XXV International Conference on Information, Communication and Automation Technologies*, 2015, pp.1-6. Prieiga per: <http://ieeexplore.ieee.org/document/7340513/?arnumber=7340513>
7. PAPADAKIS, Stamatios et al. Access Moodle Using Smart Mobile Phones. A case study in a Greek University. *2nd EAI International Conference on Design, Learning & Innovation*. [interaktyvus]. 2017 [žiūrėta: 2018-01-16]. Prieiga per: https://www.researchgate.net/publication/322021557_Access_Moodle_Using_Smart_Mobile_Phones_A_case_study_in_a_Greek_University
8. MÖDRITSCHER, Felix, Gustaf NEUMANN and Claudia BRAUER. Comparing LMS usage behavior of mobile and web users [interaktyvus]. *IEEE 12th International Conference on Advanced Learning Technologies*. 2102 [žiūrėta: 2018-01-18]. Prieiga per: <http://ieeexplore.ieee.org/document/6268200/>
9. *Screen Resolution Stats Lithuania* [interaktyvus]. 2018 [žiūrėta: 2018-10-15] Prieiga per: <http://gs.statcounter.com/screen-resolution-stats>
10. GODBOLT, Micah. *Frontend Architecture for Design Systems: A Modern Blueprint for Scalable and Sustainable Websites*. O'Reilly Media. 2016 [žiūrėta: 2019-05-14]. ISBN 978-1491926789. Prieiga per: Kindle.
11. *Design systems, style guides, pattern libraries* [interaktyvus]. 2017 [žiūrėta: 2019-05-15]. Prieiga per: <https://medium.com/boomhaus/design-systems-style-guides-all-those-libraries-what-the-hell-is-the-difference-4c2741193fdc>
12. *Pattern Libraries* [interaktyvus]. 2018 [žiūrėta: 2019-05-15]. Prieiga per: <https://medium.com/@whatjackhasmade/pattern-libraries-abcc45c6144c>
13. *U.S. Web Design System* [interaktyvus]. [žiūrėta: 2019-05-15]. Prieiga per: <https://designsystem.digital.gov>

14. *Futurelearn Design System* [interaktyvus]. [žiūrėta: 2019-05-15]. Prieiga per: <https://design-system.futurelearn.com/>
15. *The World Wide Web Consortium (W3C) Web Accessibility Initiative (WAI)* [interaktyvus]. [žiūrėta 2017-11-05]. Prieiga per: <https://www.w3.org/WAI/intro/accessibility.php>.
16. *About W3C WAI* [interaktyvus]. [žiūrėta: 2019-05-10]. Prieiga per: <https://www.w3.org/WAI/about/>
17. *Accessibility* [interaktyvus]. [žiūrėta: 2019-05-14]. Prieiga per: <https://developers.google.com/web/fundamentals/accessibility/>
18. *Web Content Accessibility Guidelines (WCAG) 2.0* [interaktyvus]. 2008 [žiūrėta: 2019-04-20]. Prieiga per: <https://www.w3.org/TR/WCAG20/>
19. *Designing for Screen Reader Compatibility* [interaktyvus]. 2017 [žiūrėta: 2019-05-10]. Prieiga per: <https://webaim.org/techniques/screenreader/>
20. *Chromevox* [interaktyvus]. [žiūrėta: 2019-05-10]. Prieiga per: <https://www.chromevox.com/>
21. JUNGŲTINĖS TAUTOS. *Jungtinių Tautų neįgalųjų teisių Konvencija ir jos fakultatyvus protokolai, 2006-12-13*. Valstybės žinios, 2010-06-19, Nr. 71-3561.
22. EUROPOS PARLAMENTAS. *Europos Parlamento ir Tarybos direktyva (ES) 2016/2102 2016 m. spalio 26 d. dėl viešojo sektoriaus institucijų interneto svetainių ir mobiliųjų programų prieinamumo* [interaktyvus]. [žiūrėta 2017-11-05]. Prieiga per: <http://eur-lex.europa.eu/legal-content/LT/TXT/PDF/?uri=CELEX:32016L2102&from=EN>
23. *Creating Theme based on Boost* [interaktyvus]. [žiūrėta 2019-01-17]. Prieiga per: https://docs.moodle.org/dev/Creating_a_theme_based_on_boost
24. *Boost Theme* [interaktyvus]. [žiūrėta 2019-01-17]. Prieiga per: https://docs.moodle.org/35/en/Boost_theme
25. *Moodle Core: plan for themes* [interaktyvus]. 2018 [žiūrėta 2019-01-17]. Prieiga per: <https://moodle.org/mod/forum/discuss.php?d=373899>
26. *What is a Template* [interaktyvus]. [žiūrėta 2019-01-17]. Prieiga per: https://docs.moodle.org/dev/Templates#What_is_a_template.3F
27. *HTML & CSS* [interaktyvus]. [žiūrėta 2018-12-15]. Prieiga per: <https://www.w3.org/standards/webdesign/htmlcss>
28. FRAIN, Ben. *The ten commandments of sane style sheets* [interaktyvus]. 2015 [žiūrėta 2018-12-15]. Prieiga per: <https://benfrain.com/the-ten-commandments-of-sane-style-sheets/>
29. *CSS Coding Style* [interaktyvus]. [žiūrėta 2019-01-17]. Prieiga per: https://docs.moodle.org/dev/CSS_Coding_Style#Correct_2
30. *The history of BEM* [interaktyvus]. [žiūrėta 2018-12-17]. Prieiga per: <https://en.bem.info/methodology/history/>
31. *BEM. Key concepts* [interaktyvus]. [žiūrėta 2018-12-17]. Prieiga per: <https://en.bem.info/methodology/key-concepts/>
32. *Scalable and Modular Architecture for CSS. A flexible guide to developing sites small and large* [interaktyvus]. [žiūrėta 2018-12-18]. Prieiga per: <https://smacss.com/>
33. *Categorizing CSS Rules* [interaktyvus]. [žiūrėta 2018-12-18]. Prieiga per: <https://smacss.com/book/categorizing>

34. *Methods to Organize CSS* [interaktyvus]. [žiūrėta 2018-12-18]. Prieiga per: <https://css-tricks.com/methods-organize-css/>
35. *Atomic design* [interaktyvus]. [žiūrėta 2018-12-20]. Prieiga per: <http://bradfrost.com/blog/post/atomic-web-design>
36. KHOLMATOVA, Alla. *Making sense of atomic design: molecules and organisms*. [interaktyvus]. [žiūrėta 2019-01-18]. Prieiga per: <https://about.futurelearn.com/blog/atomic-design-molecules-organisms>
37. *FutureLearn design system* [interaktyvus]. [žiūrėta 2019-01-17]. Prieiga per: <https://design-system.futurelearn.com/core-styles>
38. *SASS Documentation* [interaktyvus]. [žiūrėta 2019-05-14]. Prieiga per: <https://sass-lang.com/documentation>
39. *Bootstrap Framework* [interaktyvus]. [žiūrėta 2019-05-15]. Prieiga per: <https://getbootstrap.com/docs/4.3/about/overview/>
40. *Managing CSS & JS in an HTTP/2 World* [interaktyvus]. [žiūrėta 2018-12-20]. Prieiga per: <https://www.viget.com/articles/managing-css-js-http-2/>
41. NIELSEN, J. and R. MOLICH. Heuristic evaluation of user interfaces. *Proceeding CHI '90 Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. 1990, 249-256. Prieiga per: <https://dl.acm.org/citation.cfm?id=97281>
42. *WebAIM's WCAG 2 Checklist* [interaktyvus]. [žiūrėta 2019-05-15]. Prieiga per: <https://webaim.org/standards/wcag/checklist>
43. KHOLMATOVA, Alla. *Design Systems*. Smashing Media AG, Freiburg, Germany, 2017 [žiūrėta 2018-01-16]. ISBN (ePUB): 978-3-945749-59-3. Prieiga per: iBooks.
44. *Pattern Lab Documentation* [interaktyvus]. [žiūrėta 2019-01-17]. Prieiga per: <https://patternlab.io/>
45. *Devbridge Styleguide* [interaktyvus]. [žiūrėta 2019-04-20]. Prieiga per: <https://devbridge.github.io/Styleguide/>
46. *Fractal User Guide* [interaktyvus]. [žiūrėta 2019-04-20]. Prieiga per: <https://fractal.build/guide/>
47. *The plague of outline:0* [interaktyvus]. [žiūrėta 2019-05-10]. Prieiga per: <https://webaim.org/blog/plague-of-outline-0/>
48. *Success Criterion 1.3.2 Meaningful Sequence (Level A)* [interaktyvus]. [žiūrėta 2019-05-14]. Prieiga per: <https://www.w3.org/TR/WCAG21/#meaningful-sequence>
49. „Skip Navigation“ Links [interaktyvus]. [žiūrėta 2019-05-14]. Prieiga per: <https://webaim.org/techniques/skipnav/>
50. *Creating a theme based on boost* [interaktyvus]. [žiūrėta 2019-05-15]. Prieiga per: https://docs.moodle.org/dev/Creating_a_theme_based_on_boost

Priedai

1. priedas. KTU NMIT kurso studentų apklausa

Moodle aplinkos patogumas vartotojui

1. Kaip įvertintumėte Moodle aplinkos naudojimo patogumą? Pasirinkite teiginius, kurie Jums atrodo teisingi. *

Check all that apply.

- Naudotis gana patogiu
- Iš pradžių naudotis nebuvo patogiu, tačiau įpratau prie sąsajos
- Greitai surandu viską ko reikia
- Kartais reikia gerokai paieškoti, kad surasčiau to ko ieškau
- Visada pavyksta sklandžiai užbaigti norimas užduotis
- Kartais nesuprantu ką reiškia spausti norint atlikti veiksmą

2. Naudodamiesi Moodle aplinka, ką dažniausiai renkatės, norėdami pasiekti Moodle pradžios puslapį? *

Mark only one oval.

- Mano Pagrindinis
- Pagrindinis
- Spaudžiate ant Moodle logotipo viršutinėje ekrano dalyje
- Neprisimenu

3. Kaip dažniausiai skaitote Moodle pateiktą tekstinę mokymosi medžiagą? *

Mark only one oval per row.

	niekada	kartais	dažniausiai
Skaitote ekrane	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Išsaugojate ir skaitote ekrane	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Išsisaugote ir atsispausdinate	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

4. Kaip dažnai pasiekiate Moodle per mobilųjį įrenginį (telefoną, planšetę)? *

Mark only one oval.

- Kasdien
- Maždaug kartą per savaitę
- Maždaug kartą per mėnesį
- Maždaug kartą per semestrą
- Niekada nesijungiu per mobilųjį įrenginį

5. **Kaip pasiekiate Moodle mobiliajame įrenginyje? ***

Mark only one oval.

- Įsidičiau Moodle programėlę
- Per naršyklę
- Mobiliajame įrenginyje Moodle nenaudoju visai

6. **Kokias veiklas internetu dažniausiai atliekate mobiliojo įrenginio pagalba? ***

Mark only one oval per row.

	niekada	kartais	dažnai
Naršau socialiniuose tinkluose	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Bendrauju asmeninėmis žinutėmis	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Naršau internetiniuose portaluose	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Skaitau tinklaraščius	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Skaitau el. knygas	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Žiūriu vaizdo įrašus	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Klausausi tinklalaidžių (podcasts)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Tikrinuosi el. pašta	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Rašau el. laiškus	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

Moodle aplinkos veiklos

7. **Kokias veiklas atliekate prisijungę prie Moodle mobiliajame įrenginyje? ***

Check all that apply.

- Pasitikrinu savarankiškų darbų, testų įvertinimus
- Pasitikrinu užduotas užduotis
- Pasitikrinu artimiausius įvykius (paskaitų grafiką, savarankiškų darbų terminus)
- Skaitau pateiktą mokymosi medžiagą
- Žiūriu paskaitų įrašus
- Dalyvauju paskaitų transliacijoje
- Rašau asmenines žinutes dėstytojams
- Skaitau ir rašau forumo žinutes
- Nesijungiu prie Moodle per mobilųjį įrenginį

8. **Kaip planuojate savarankiškų užduočių atlikimo terminus? ***

Mark only one oval per row.

	niekada	kartais	dažniausiai
Susivedate užduotis į Moodle kalendorių	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Naudojatės dėstytojo suvestomis į kalendorių užduotimis	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Eksportuojate kalendorių Moodle į savo asmeninį kalendorių (Outlook, Google ar pan.)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Susivedate patys užduočių terminus į savo asmeninį kalendorių (Outlook, Google ar pan.)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

9. Kokiū būdu virtualiai bendraujate su dėstytojais? *

Mark only one oval per row.

	niekada	kartais	dažniausiai
Rašau el. laišką	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Rašau asmeninę žinutę per Moodle	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Užduodu klausimą Moodle forume	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

10. Kokiū būdu virtualiai bendraujate su kitais besimokančiais? *

Mark only one oval per row.

	niekada	kartais	dažnai
Moodle forume	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
El. paštu	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Facebook grupėje	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Asmeninėmis žinutėmis per Messenger	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Asmeninėmis žinutėmis per Moodle	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

Informacija apie Jus

Jūsų informacija yra nuasmeninta ir niekaip negali būti susieta su anketą užpildžiusio asmens tapatybe. Amžiaus ir lyties duomenys naudojami apibendrinti demografinę besimokančiųjų virtualios mokymosi aplinkos dalyvių informaciją. Jei nepageidaujate atskleisti savo amžiaus ar lyties pasirinkite "Nepageidauju atskleisti".

11. Jūsų lytis *

Mark only one oval.

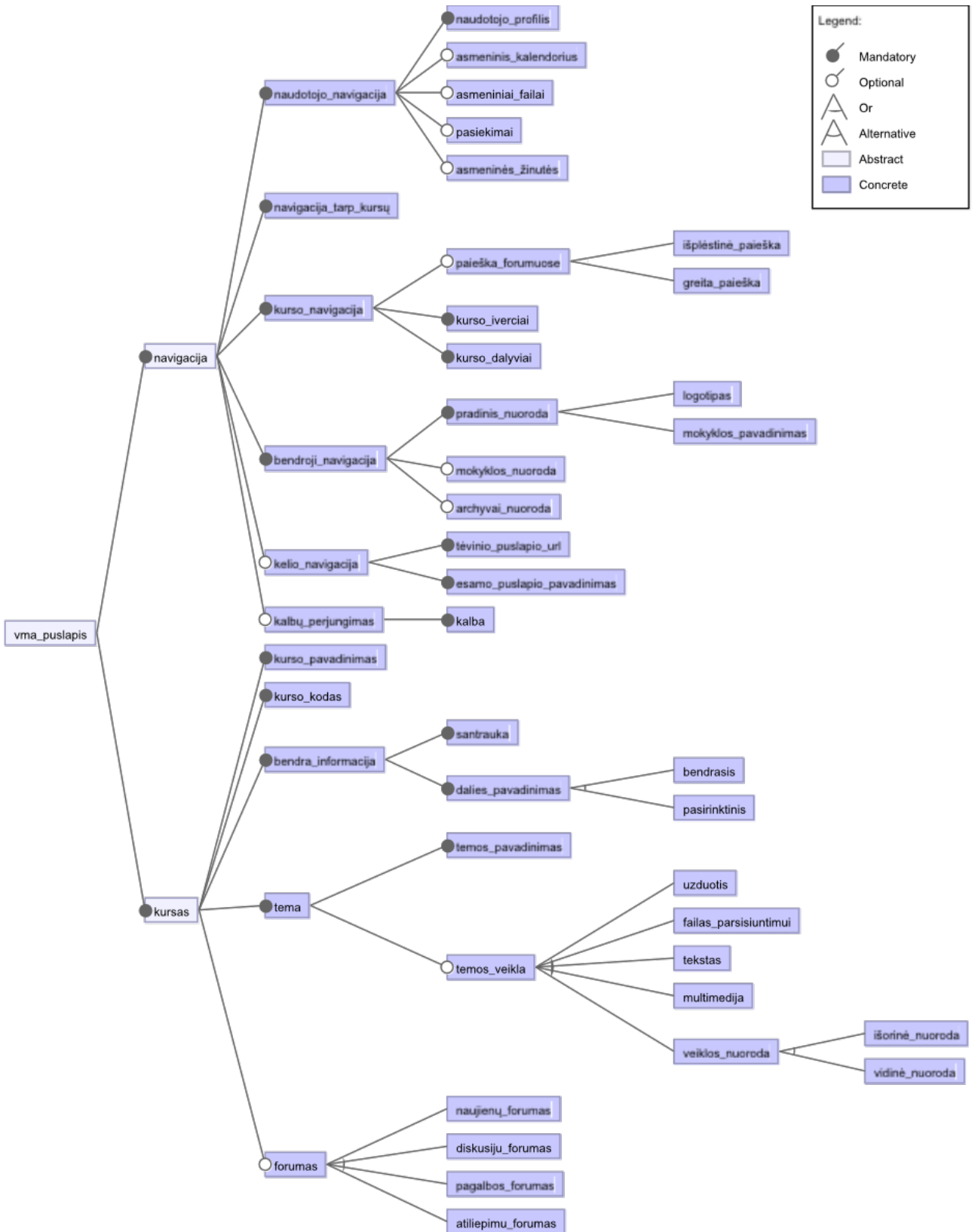
- Moteris
- Vyras
- Nepageidauju atskleisti

12. Jūsų amžius *

Mark only one oval.

- 18 - 22 m.
- 23 - 27 m.
- 28 - 32 m.
- 33 - 37 m.
- 38 - 42 m.
- 43 m. ar daugiau
- Nepageidauju atskleisti

2. priedas. VMA kurso puslapio HTML komponentų požymių diagrama



Application of modular design system in virtual learning environment

Aistė Čiurlytė Kolesničenko¹, Ramūnas Kubiliūnas¹

¹ *Kaunas University of Technology, Lithuania*

Abstract. Web-based learning technologies are experiencing an evolutionary process from e-learning tools, providing course material online to open online courses and learning social networks. Virtual learning environments are created by educational institutions, schools, universities or even independent course providers and usually are based on Learning management systems (LMS). Many educational institutions maintain their own systems based on popular open source LMS, such as Moodle. The article analyzes a structure of LMS Moodle page and presents a process of modular design system creation for presentational layer of the popular learning management system Moodle.

Keywords: learning management system, Moodle, modular design system, graphical user interface

1.1 INTRODUCTION

Web-based learning technologies are experiencing an evolutionary process from e-learning tools, providing course material online to open online courses and learning social networks. Virtual learning environments are created by educational institutions, schools, universities or even independent course providers and usually are based on Learning management systems (LMS). Some institutions choose to purchase paid “Software as a Service” solutions, many others though maintain their own systems based on popular open source LMS, such as Moodle. LMS maintainers and administrators tend to use standard core theme or some third-party theme for Moodle, but the common situation is when generic solutions don't fit course users' specific needs.

The article analyzes a structure of LMS Moodle HTML page and creation principles of a modular design system for the presentational layer of the popular learning management system Moodle.

1.2 RESPONSIVENESS OF USER INTERFACES

1.2.1 Responsiveness of web-based system design

As of Dec 2108, according to Moodle site, there were 203 Moodle powered sites registered in Lithuania (Registered Moodle sites, 2018). Moodle, as the creators describe, is a highly customizable modular open-source system that can be tailored to their users' needs (About Moodle, 2018).

Though high flexibility of the LMS brings necessity to its maintainers undertake responsibility for proper configuration of functional and presentational features of the system in order to sustain systems user experience and engagement of their learners. The challenges that are faced at the presentational level of the LMS are:

- A wide range of screen sizes and devices used for web access makes it difficult to adapt to every size.
- Different users' interaction tools and scenarios on different devices.
- Making user interfaces accessible.

Responsiveness of web-based systems brought new challenges to the developers of the presentational layer. Screen resolution statistics (Screen Resolution Stats Lithuania, 2018) presented in Figure 1 visualize a wide range of screen resolutions of devices used to access the Web. We can assume that the smaller pixel size and portrait viewport orientation screens (360x640) are mobile phones. Others ranging from 2560x1440 pixels to 1024x768 pixels could be desktop or tablet computers. The biggest category (20,67%) though is “Other”: non-standard or unknown screen resolution.

There’s also a difference between interaction possibilities and habits between users of different groups of devices: users interact with desktop computers by using a mouse, keyboard, trackpad or another input device, while on mobile device primary interaction is fulfilled by touch. Worth to mention though that there’s some overlap between tools of interaction: desktop computers may have a touch screen and a mobile device (tablet, phone) could be used with an input device. Lithuanian web user statistics (Desktop vs Mobile vs Tablet Market Share, 2018) showed that 73,79% of users were entering the web on a desktop computer, 24,39% on a mobile phone and a small fraction of 1,82% on a tablet (Figure 2).

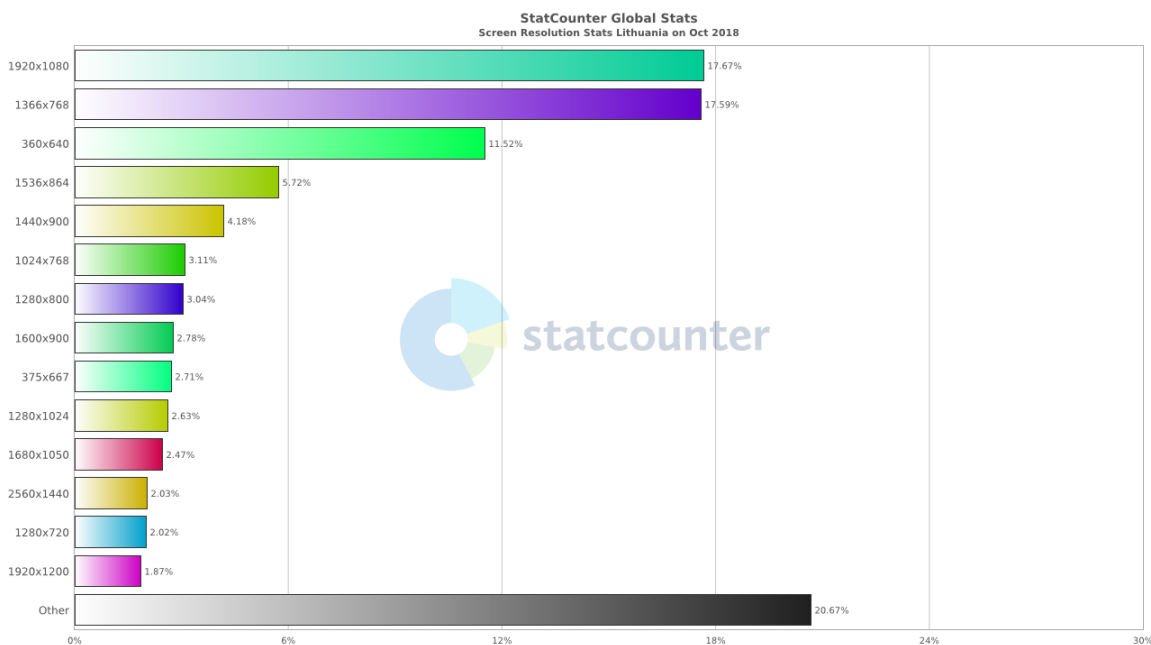


Figure 1. Range of screen sizes accessing the Web. Lithuania, Oct. 2018 (Statcounter, 2018)

Despite some fuzziness and overlap in categorization, we could group users’ devices:

- By size (small, medium, large).
- Interaction tools (touch, mouse, keyboard).

Possible solutions for achieving responsiveness may encounter:

- Detect mobile device and present a mobile version of the theme.
- Suggest users install a mobile application of the LMS.
- Use a responsive LMS theme.

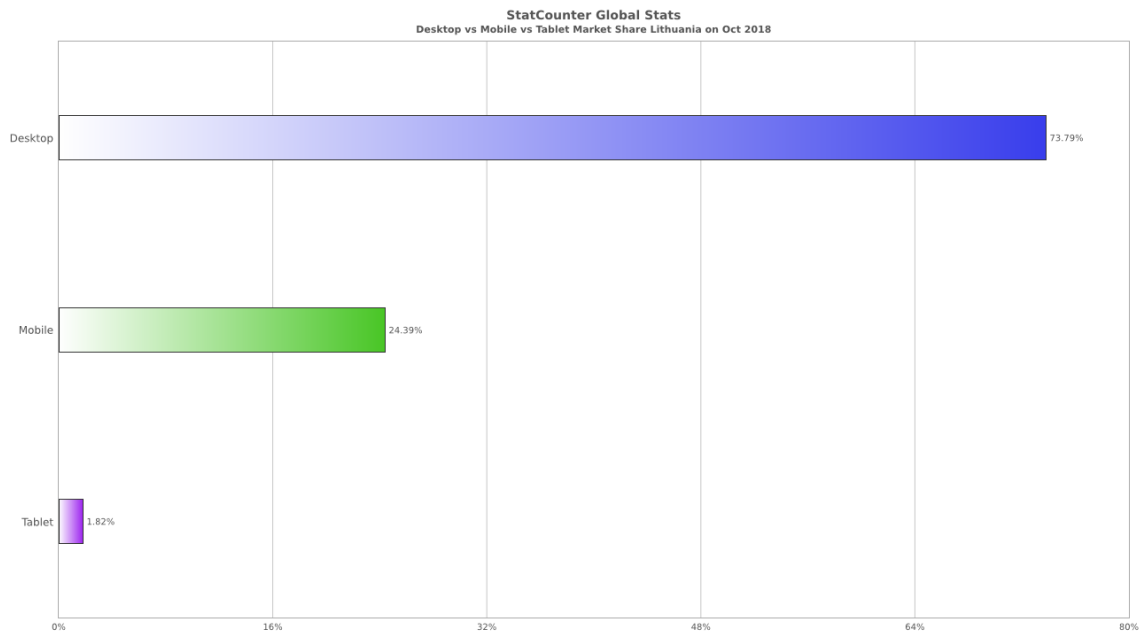


Figure 2. Devices used for accessing the Web. Lithuania, Oct. 2018 (Statcounter, 2018)

Many ways of using learning content and different solutions of presenting it to the user upbrings a need for design patterns that can be used for the creation and maintenance of the responsive, adaptive and accessible LMS presentational layer.

1.2.2 Accessibility and inclusion of user interfaces

Accessibility is emphasized by Web World Wide Web Consortium (W3C) as key feature of the web-based system (Web Accessibility Initiative). European Union in 2016 released directive (The European Parliament and the Council of the European Union, 2016) which states that all websites and mobile applications of public sector bodies should comply with the accessibility requirements.

According to Web Accessibility Initiative there are significant overlaps between accessibility, usability and inclusion (Accessibility, Usability, and Inclusion: Related Aspects of a Web for All, 2016) so we can affirm that making user interfaces more accessible to people with special needs we are making them more usable for all users in general. Accessibility on the web is achieved by using semantic structure for HTML document outline, WAI-ARIA landmark roles, properly marked links and forms, font settings, color and contrast.

1.3 MODULAR DESIGN PATTERNS FOR LMS

The modular pattern library for LMS model is being developed to help its maintainers at schools and other educational institutions to provide guidance for proper setup, configuration and modification of the elements of the user interface on web-based LMS Moodle. Proposed model is based on semantic HTML structure enhanced with accessibility features such as WAI-ARIA roles and defined CSS naming conventions. Modules are going to include HTML markup together with CSS rules relevant to the module ensuring accessibility of the module as well as media queries for its responsiveness and adaptability to different devices or ways of users' interaction.

As of version 3.2 Moodle introduced a new theme called Boost that is based on Bootstrap v.4 framework. Moodle HQ's Open Source Development Coordinator Sander Bangma announced plans to remove "Clean" and "More" themes from Moodle's core and concentrate on developing Boost theme (Bangma, 2018). LMS administrators and maintainers of the institutions that are using the aforementioned themes will have to make decisions upon theme change. Choosing another theme can mean losing improvements and style overrides made for the current theme.

CSS methodologies such as BEM or SMACSS were created to form a structure for the presentational layer of the web-based systems. According to the BEM, one of the mentioned methodologies (The history of BEM, n.d.), the presentational layer of the web-based system is approached as a Lego building blocks structure containing smallest components which are connected and reused for building modular structures. LMS as a complex web-based system can benefit from the modular approach by improving its maintainability.

For creating Moodle's modular design structure we decided to use two types of modules: blocks and elements.

- Elements: modules with a high level of granularity that usually can't be used as a standalone item and must be a part of some block context.
- Blocks: modules that can contain elements and other blocks as well.

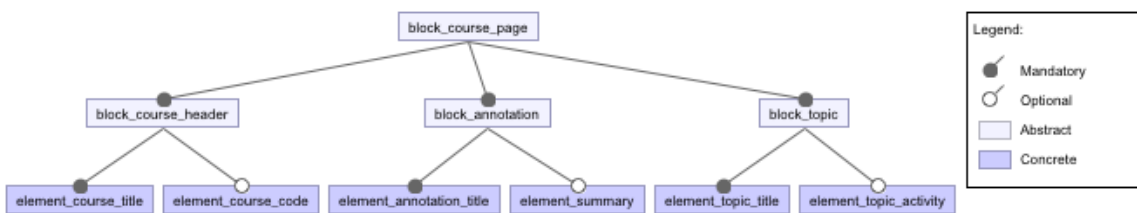


Figure 3. Feature diagram of reusable LMS design components on a course page

A feature diagram of course block (Figure 3) containing other blocks and child elements. We created a library of reusable modules by encapsulating semantic HTML markup, accessibility features, CSS that is relevant to the module and media queries used for responsiveness (Table 1).

1. MODULES OF A COURSE PAGE

Module	HTML markup	CSS class	ARIA landmark	Type
Course page	<main>	.b-course	role="main" (one per page)	Block
Course header	<header>	.b-header-main		Block
Course annotation	<section>	.b-summary		Block
Course topic	<article>	.b-topic	role="article"	Block
Course title, Course code	<h1>	.e-header-main_title	role="heading"	Element
Annotation title	<h2>	.e-summary_title		Element

Annotation summary	<div>	.e-summary_text		Element
Topic title	<h2>	.e-topic_title		Element
Topic activity	<div>	.e-topic_activity		Element

Table 1 shows a table of modules containing recommended HTML tags, CSS selectors and WAI-ARIA roles for accessibility. CSS selectors share a naming convention where it is created with .b- or .e- prefix standing for “block” and “element” respectively. The modules will be further used for creating style guides and a component library for the user interface of a Moodle theme. Further studies of web-based LMS on performance and accessibility improvement have to be made by performing series of page speed and accessibility tests of the created Moodle theme, based on the model.

1.4 CONCLUSIONS

Users of web-based systems are accessing the Web on a vast range of devices that can have different screen sizes and interaction methods.

Making a system more accessible to people with disabilities we also provide a better experience for all users of a system.

A modular design system serve as an asset of reusable components and documentation for the presentational layer of LMS. Design system defines modules as basic building blocks that encapsulate HTML, CSS code, and WAI-ARIA accessibility rules relevant to the component.

Library of well documented modular user interface components should provide the maintainers of LMS design patterns for developing semantic HTML structure and accessibility best practices of the LMS theme.

REFERENCES

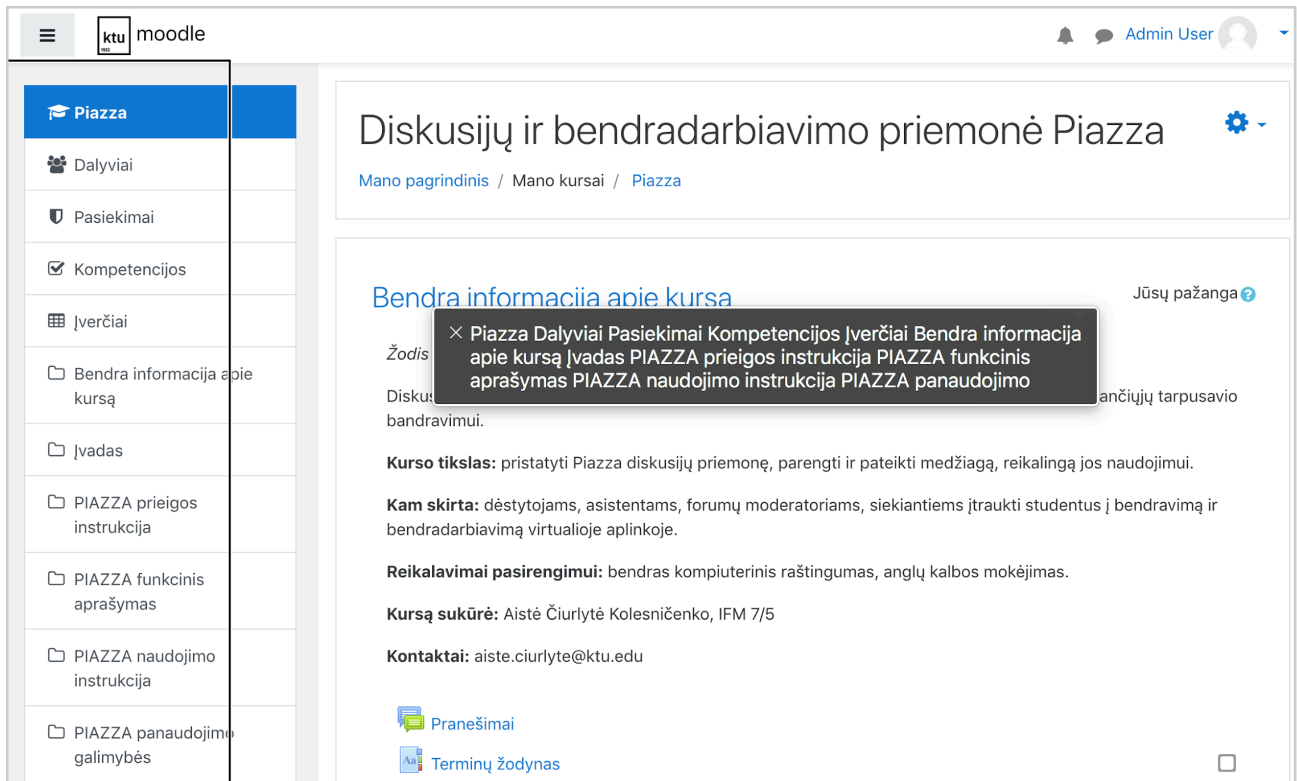
- About Moodle*. (n.d.). From https://docs.moodle.org/36/en/About_Moodle
- Accessibility, Usability, and Inclusion: Related Aspects of a Web for All*. (2016, May 6). From <https://www.w3.org/WAI/intro/usable>
- Accessible Rich Internet Applications (WAI-ARIA)*. (2017, December 14). From https://www.w3.org/TR/wai-aria-1.1/#role_definitions
- Bangma, S. (2018, July 27). *Moodle Core: plan for themes*. From <https://moodle.org/mod/forum/discuss.php?d=373899>
- Desktop vs Mobile vs Tablet Market Share*. (2018, October 15). From <http://gs.statcounter.com/platform-market-share/desktop-mobile-tablet>
- Registered Moodle sites*. (2019, January 17). From <https://moodle.net/sites/>
- Screen Resolution Stats Lithuania*. (2018, October 15). From <http://gs.statcounter.com/screen-resolution-stats>
- The European Parliament and the Council of the European Union. (2016). *Directive on the accessibility of the websites and mobile applications of public sector bodies*. (Official Journal of the European Union No. L 327/13). From <https://eur-lex.europa.eu/legal-content/EN/TXT/PDF/?uri=CELEX:32016L2102&from=EN>

MODULIARIOS DIZAINO SISTEMOS TAIKYMAS VIRTUALIOJOJE MOKYMOSI APLINKOJE

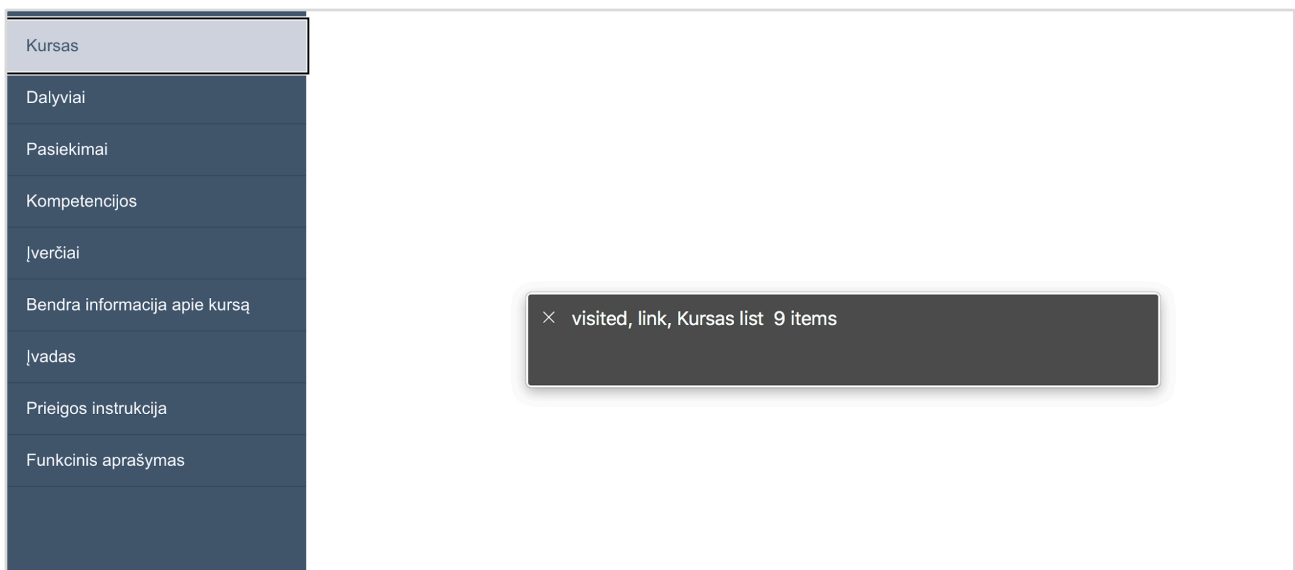
Internetinės mokymosi technologijos šiuo metu evoliucionuoja nuo e. mokymosi priemonių, kurso mokymosi medžiagos teikimo internetu link atvirųjų internetinių kursų bei socialinių tinklų, skirtų mokymuisi. Mokymosi įstaigos, mokyklos, universitetai ar net įvairūs kursų rengėjai kuria virtualiąsias mokymosi aplinkas populiarių, atvirojo kodo mokymosi valdymo sistemų (MVS), tokių kaip Moodle, pagrindu. Straipsnyje pristatomas MVS Moodle modulinis grafinės naudotojo sąsajos modelis, apjungiantis semantinę HTML struktūrą, žymėjimą, užtikrinantį pasiekiamumą neįgaliesiems, CSS kalbos klasių pavadinimų formavimo metodą.

4. Priedas. Šoninės navigacijos testavimas su „Voice Over“ balso asistentu

VMA „Boost“ temos šoninės navigacijos testavimas. Balso asistentas be pauzių skaito visas nuorodas, esančias šoniniame skydelyje. Atsivėrusiame „Voice Over“ lange matosi tekstas, kurį skaito balso asistentas.

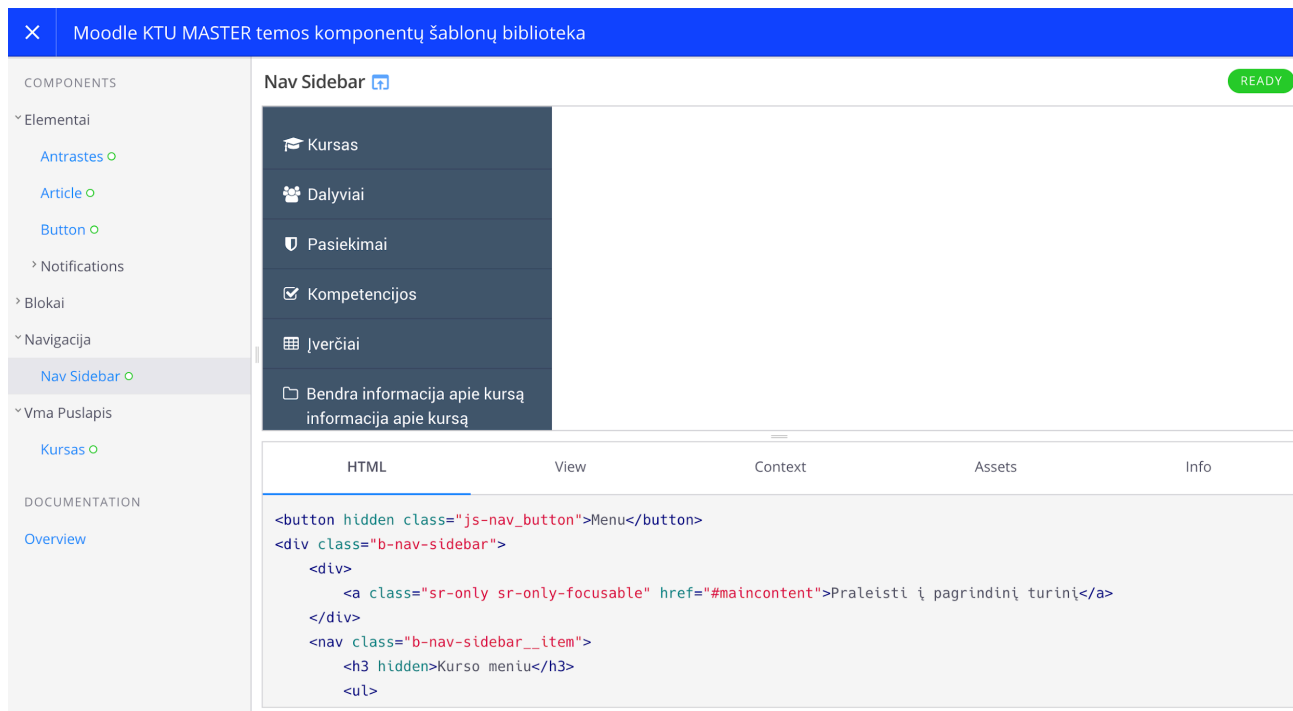


Šoninės navigacijos, sudarytos iš nenumeruoto sąrašo nuorodų testavimas balso asistentu „Fractal“ priemonėje. Balso asistentas nurodo fokusuotos nuorodos pavadinimą bei sąrašo esančių nuorodų skaičių.



5. priedas. Komponento atvaizdavimas „Fractal“ priemonės aplinkoje

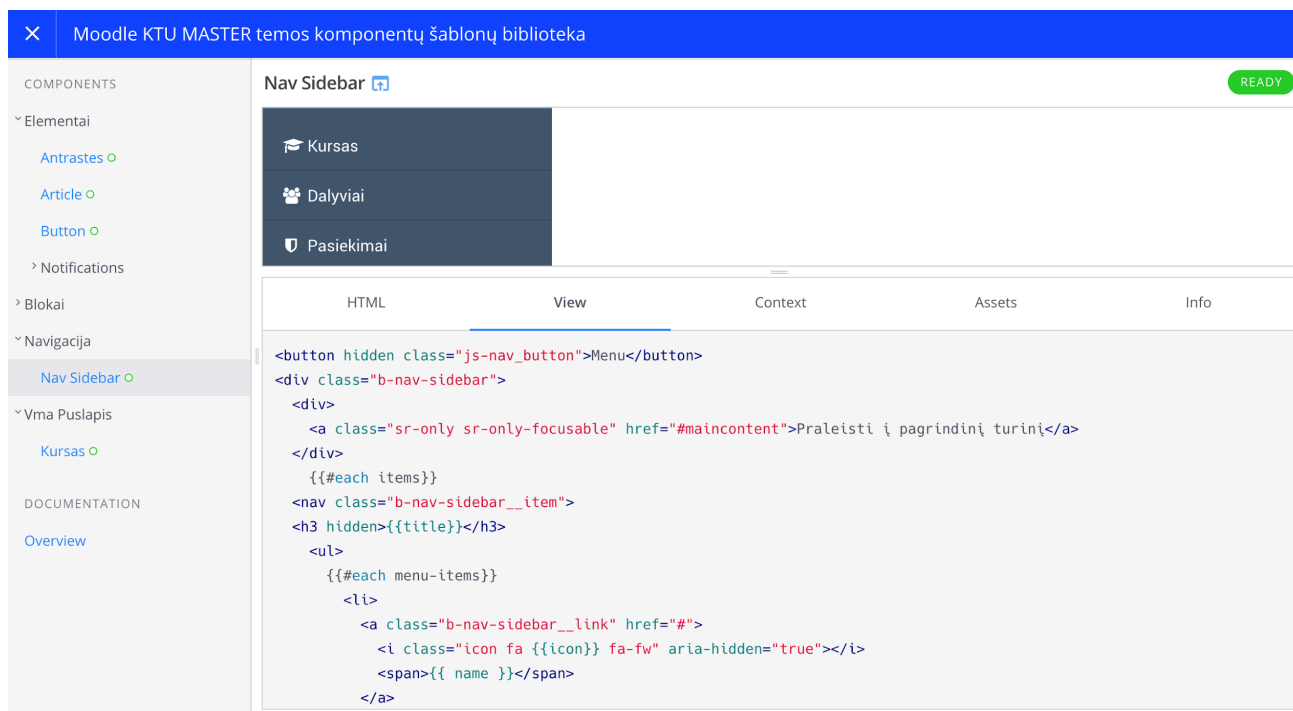
Šoninės navigacijos atvaizdavimas „Fractal“ priemonės aplinkoje. Apatiniame skydelyje pasirinktas HTML kodo blokas.



The screenshot shows the Moodle KTU MASTER components library interface. The 'Nav Sidebar' component is selected, and its HTML code is displayed in the 'HTML' tab. The code includes a hidden button for the menu and a list of navigation items.

```
<button hidden class="js-nav_button">Menu</button>
<div class="b-nav-sidebar">
  <div>
    <a class="sr-only sr-only-focusable" href="#maincontent">Praleisti į pagrindinį turinį</a>
  </div>
  <nav class="b-nav-sidebar__item">
    <h3 hidden>Kurso meniu</h3>
    <ul>
```

„Fractal“ apatiniame skydelyje rodomas .hbs failo kodas su kintamaisiais iš kontekstinio JSON failo.



The screenshot shows the Moodle KTU MASTER components library interface. The 'Nav Sidebar' component is selected, and its HBS code is displayed in the 'View' tab. The code includes a hidden button for the menu and a list of navigation items, with placeholders for dynamic content from a JSON file.

```
<button hidden class="js-nav_button">Menu</button>
<div class="b-nav-sidebar">
  <div>
    <a class="sr-only sr-only-focusable" href="#maincontent">Praleisti į pagrindinį turinį</a>
  </div>
  {{#each items}}
  <nav class="b-nav-sidebar__item">
    <h3 hidden>{{title}}</h3>
    <ul>
      {{#each menu-items}}
      <li>
        <a class="b-nav-sidebar__link" href="#">
          <i class="icon fa {{icon}} fa-fw" aria-hidden="true"></i>
          <span>{{ name }}</span>
        </a>
      </li>
    </ul>
  </nav>
  {{/each}}
</div>
```

Kontekstinio failo peržiūra „Fractal“ priemonėje.

Moodle KTU MASTER temos komponentų šablonų biblioteka

COMPONENTS

- Elementai
 - Antrastes
 - Article
 - Button
- Notifications
- Blokai
- Navigacija
 - Nav Sidebar
- Vma Puslapis
 - Kursas

DOCUMENTATION

Overview

Nav Sidebar READY

Kursas
Dalyviai
Pasiekimai

HTML View Context Assets Info

```
{
  "items": [
    {
      "title": "Kurso meniu",
      "menu-items": [
        {
          "icon": "fa-graduation-cap",
          "name": "Kursas"
        },
        {
          "icon": "fa-users",
          "name": "Dalyviai"
        },
        {
          "icon": "fa-shield",

```

CSS kodo peržiūra „Fractal“ priemonės aplinkoje.

Moodle KTU MASTER temos komponentų šablonų biblioteka

COMPONENTS

- Elementai
 - Antrastes
 - Article
 - Button
- Notifications
- Blokai
- Navigacija
 - Nav Sidebar
- Vma Puslapis
 - Kursas

DOCUMENTATION

Overview

Nav Sidebar READY

Kursas
Dalyviai
Pasiekimai

HTML View Context Assets Info

FILE: nav-sidebar.scss

CONTENT:

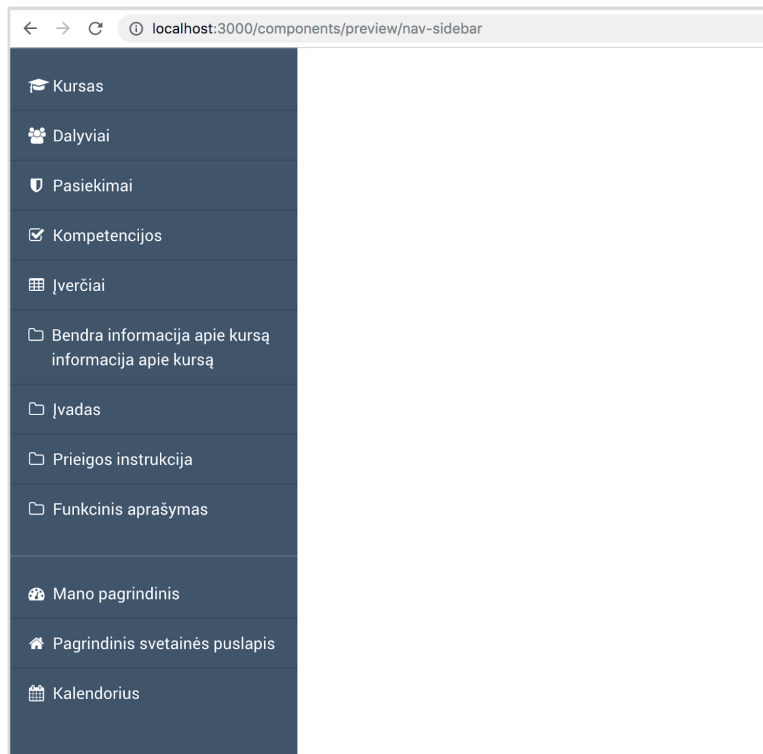
```
$drawer-bg: #42556E;

.b-nav-sidebar {
  display: block;
  font-family: 'Roboto', sans-serif;
  position: fixed;
  width: 285px;
  height: 100%;
  overflow-y: scroll;
  -webkit-overflow-scrolling: touch;
  transition: right .5s ease, left .5s ease;
  z-index: 999;
  background-color: $drawer-bg;

```

Komponento peržiūra „Fractal“ komponento peržiūros (angl. *preview*) aplinkoje. Sukurtas šoninės navigacijos komponentas „Moodle“ VMA, atitinkantis pasiekiamumo standartus. Pakeistas meniu

atvaizdavimo stilius, pateikiant kontrastuojantį su aplinka šoninį meniu bei atskiriant pirmą (šviesus turinio fonas) ir antrą (tamsus šoninio meniu fonas) planą puslapyje.



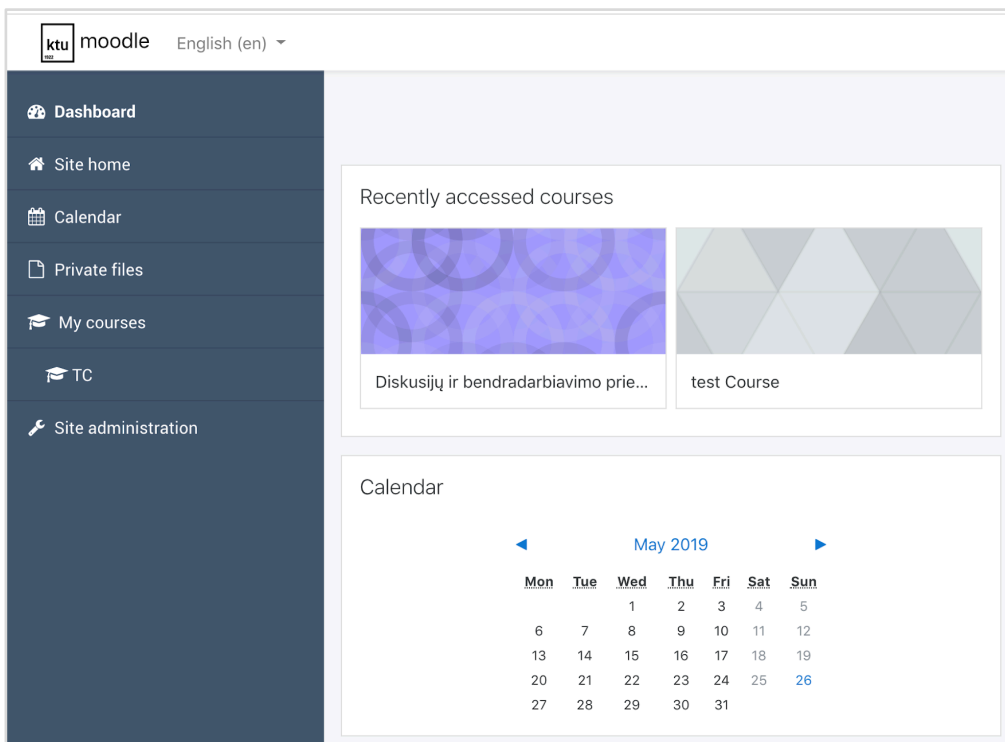
6. priedas. „Boost“ temos atitikimo pasiekiamumo gairėms rezultatai

Lygmuo	Kriterijai	„Boost“ rezultatai
	Semantinis HTML	
A	Antraštėms (<h1>), sąrašams (, ir <dl>), tinklalapio dalims (<main>, <nav>) ir pan. naudojamas semantinis žymėjimas. Semantinis žymėjimas panaudotas teisingai, pagal paskirtį.	+/-
A	Skaitymo ir navigacijos seka yra logiška ir intuityvi.	-
A	Puslapis turi informatyvų pavadinimą (angl. <i>title</i>).	+
AA	Tinklalapio orientacija nėra apribota (pvz. tik vertikali ar gulsčia).	+
AAA	Specialūs HTML5 elementai (<main>, <nav>, <section>, <article>) ar ARIA žymės (role="banner", role="navigation", role="main")	+/-
AAA	Kur reikalinga, naudojamas WAI ARIA žymėjimas, siekiant geriau identifikuoti sąsajos komponentų paskirtį.	+
	Spalvos kontrastas ir tekstas	
A	Spalva nėra vienintelis matomas vizualinių elementų išryškinimo būdas.	+/-
A	Nuorodų išskyrimui iš aplinkos nėra naudojama vien tik spalva (pvz. naudojamas pabraukimas), nebent kontrastas tarp nuorodos tekste ir aplinkinio teksto yra santykiu 3:1 a bei atsirandantis papildomas išskyrimas (pabraukimas) nuorodą fokusuojant ar užvedant pelę.	-
AA	Tekstas kontrastuoja su fonu santykiu 4,5:1 ir daugiau.	+
AA	Didesnis tekstas (nuo 18pt) arba 14pt pajuodintas tekstas kontrastuoja su fonu santykiu 3:1 ar daugiau.	+
AA	Puslapio tekstas yra įskaitomas bei funkcionalus, jį padidinus (angl. <i>zoom</i>) iki 200%.	+
AAA	Teksto blokai, ilgesni nei viena eilutė: <ul style="list-style-type: none"> - ne ilgesni nei 80 ženklų; - nenaudojama abipusė lygiuotė; - tarpas tarp eilučių yra mažiausiai ½ teksto dydžio; - tarpas tarp paragrafų yra 1,5 karto didesnis už tarpą tarp eilučių. 	- - + +
	Navigacija	
A	Visos puslapio funkcijos veikia naudojantis klaviatūra. Tai netaikoma toms funkcijoms, kurios negali būti pasiektos klaviatūros pagalba (pvz. piešimas).	+
A	Kiekviename puslapyje pasikartojantiems elementams (pvz. navigacijai) praleisti, pateikiamos greitosios nuorodos.	+/-

A	Nuorodų, formų elementų, kitų interaktyvių elementų navigacijos tvarka yra intuityvi ir logiška.	-
AA	Fokusavimas: vizualiai akivaizdu, kuris puslapio elementas yra fokusuotas ir matomas.	-
AA	Navigacijos nuorodos, pasikartojančios kiekviename puslapyje (pvz. meniu), naviguojant per puslapius nekeičia savo išdėstymo tvarkos.	-
AAA	Interaktyvūs elementai yra mažiausiai 44px aukščio ir 44px pločio, nebent yra pateikta alternatyvi nuoroda, interaktyvus elementas yra teksto dalis (interaktyvi nuoroda tekste), interaktyvus elementas negali būti modifikuotas (standartinis formos „check“ laukelis).	-
	Prisitaikantis dizainas	
AA	Sumažinus ekrano dydį iki 320px neprarandamas funkcionalumas, neatsiranda horizontali slankjuostė. Horizontalios slankjuostės išimtys taikomos turiniui, kuriam yra būtina horizontali peržiūra, pvz. duomenų lentelės, grafai, žemėlapiai.	+
AA	Elementams, kurie savyje turi teksto elementų, nenustatomas aukštis pikseliais.	+

7. priedas. Komponento perkėlimas į VMA „Moodle“ temą

Šoninė navigacija, perkelta iš „Fractal“ priemonės į VMA „Moodle“ paveldinčią „Boost“ temą.



Modifikuoti šablonų failai:

- nav-drawer.mustache;
- flat-navigation.mustache.

```
nav-drawer.mustache x flat_navigation.mustache .../theme_boost flat_navi
```

```
1 <div id="nav-drawer" data-region="drawer" class="b-nav-sidebar hidden-print
moodle-has-zindex {{^navdraweropen}}closed{{/navdraweropen}}" aria-hidden="{
{{#navdraweropen}}false{{/navdraweropen}}{{^navdraweropen}}true{{/
navdraweropen}}" tabindex="-1">
2   <div>
3     <a class="sr-only sr-only-focusable" href="#maincontent">Praleisti į
pagrindinį turinį</a>
4   </div>
5   {{> theme_boost/flat_navigation }}
6 </div>
7
```

```

nav-drawer.mustache  flat_navigation.mustache x  _nav.scss
<nav class="b-nav-sidebar__item">
  <h3 hidden>Kurso meniu</h3>
  <ul>
    {{# flatnavigation }}
      {{#action}}
        <li>
          <a class="m-l-{{get_indent}} b-nav-sidebar__link {{#isactive}}
            font-weight-bold{{/isactive}}" href="{{action}}" data-key="{{key}}"
            data-isexpandable="{{isexpandable}}" data-indent="{{get_indent}}"
            data-showdivider="{{showdivider}}" data-type="{{type}}" data-nodetype="{{
            nodetype}}" data-collapse="{{collapse}}" data-forceopen="{{forceopen}}"
            data-isactive="{{isactive}}" data-hidden="{{hidden}}" data-preceedwithhr="{{
            preceedwithhr}}" {{#parent.key}}data-parent-key="{{.}}"{{/parent.key}}>
              {{#icon.pix}}
                {{#pix}}>{{icon.pix}}, {{icon.component}}, {{icon.alt}}{
                /pix}
              {{/icon.pix}}
              <span class="{{#isactive}}font-weight-bold{{/isactive}}>{{text}}
            </span>
          </a>
        </li>
      {{/action}}
      {{^action}}
        <li class="b-nav-sidebar__header">
          <div class="m-l-{{get_indent}}">
            {{#icon.pix}}
              {{#pix}}>{{icon.pix}}, {{icon.component}}, {{icon.alt}}{
              /pix}
            {{/icon.pix}}
            <span>{{text}}</span>
          </div>
        </li>
      {{/action}}
    {{/ flatnavigation }}
  </ul>
</nav>

```

Iš „Fractal“ priemonės perkeltas _nav.scss stilių failas ir importuotas į post.scss failą:

```

$drawer-bg: #42556E;
.b-nav-sidebar {
  display: block;
  font-family: 'Roboto', sans-serif;
  padding: 0;
  position: fixed;
  width: 285px;
  height: 100%;
  overflow-y: scroll;
  -webkit-overflow-scrolling: touch;
  transition: right .5s ease, left .5s ease;
  z-index: 999;
}

```

```

        background-color: $drawer-bg;
ul {
    list-style: none;
    padding: .75rem 0;
}
li {
    border-bottom: 1px solid rgba(0,0,0,.125);
}
li:last-child {
    border-bottom: none;
}
a {
    color: #fff;
    padding: .75rem 1.25rem;
    width: 100%;
    &:focus, &:hover {
        color: $drawer-bg;
        background-color: lighten($color: $drawer-bg, $amount: 90);
        text-decoration: none;
    }
}
.icon {
    font-size: 16px;
    width: 16px;
    height: 16px;
    margin: 0;
    padding: 0;
    margin-right: .5rem;
}
.b-nav-sidebar__item {
    margin-bottom: 10px;
}
.b-nav-sidebar__item:not(:first-child) {
    border-top: 1px solid rgba(255,255,255, .2);
}
.b-nav-sidebar__header {
    color: #fff;
    padding: .75rem 1.25rem;
    width: 100%;
}
.b-nav-sidebar__link {
    display: flex;

```

```
        align-items: baseline;
span {
    display: block;
}
}

@media screen and (max-width: 780px) {
    .b-nav-sidebar {
        display: none;
    }
    .is-active {
        display: block;
    }
}
```