

VYTAUTO DIDŽIOJO UNIVERSITETAS
ŠVIETIMO AKADEMIJA

Linas Katakinas

**DEMONSTRACINĖ ELEKTRONINIO BANKO SISTEMA
SUAUGUSIŲJŲ FINANSINIO RAŠTINGUMO UGDYMU**

Bakalauro baigiamasis darbas

Švietimas ir informacinės technologijos studijų programa, valstybinis kodas 612X30004
Andragogikos ir taikomosios informatikos studijų kryptis

Vadovas (-ė). Doc. dr. Elena Trepulė _____
(Moksl. laipsnis, vardas, pavardė) (Parašas) (Data)

Apginta Lina Kaminskienė _____
(Švietimo akademijos kanclerė) (Parašas) (Data)

Kaunas, 2020

TURINYS

SANTRUMPŲ IR TERMINŲ ŽODYNAS	3
SANTRAUKA.....	4
ABSTRACT	5
1. ĮVADAS.....	6
1.1 Darbo tikslas	6
1.2 Uždaviniai.....	6
2. PATYRIMINIO MOKYMOSI NAUDA FINANSINIAM RAŠTINGUMUI.	7
2.1 Patyriminis mokymasis.....	7
2.2 Finansinis raštingumas	11
2.3 Finansinį raštingumą lemiantys veiksniai ir poreikiai	12
2.4 Finansinio švietimo padėtis	13
2.5 Finansinio švietimo rekomendacijos	15
3. DEMONSTRACINIŲ ELEKTRONINIŲ BANKŲ APŽVALGA	17
3.1 Lietuvos elektroniniai bankai	17
3.2 Užsienyje veikiantys demonstraciniai elektroniniai bankai	19
3.3 Demonstracinių elektroninių bankų palyginimas	23
3. SISTEMOS KŪRIMAS.	25
3.1 Prototipavimas	26
3.2 Sistemos vartotojo sąsajos kūrimas	27
3.3 Naudojami dizaino elementai	28
3.4 Programinės įrangos pasirinkimas.....	30
3.5 Demonstracinio interneto banko programavimas	36
4. TESTAVIMAS.	54
5. PRITAIKOMUMAS.....	56
6. ATILIEPIMAI IR ĮVERTINIMAI.....	57
7. REZULTATAI IR IŠVADOS.....	59
7.1 Rezultatai	59
7.2 Išvados.....	59
8. LITERATŪROS SĄRAŠAS	60
9. PRIEDAI.....	62
9.1 Demonstracinio elektroninio banko programinis kodas	62

SANTRUMPŲ IR TERMINŲ ŽODYNAS

AJAX – asinchroninis JavaScript ir XML programavimas.

SEPA – bendra mokėjimų eurais erdvė (angl. Single Euro Payments Area).

API – aplikacijų programavimo sąsaja (angl. Application Programming Interface).

CSS – stiliaus žymėjimo kalba.

HTML – hiperteksto žymėjimo kalba.

Javascript – programavimo kalba.

Modal – iššokantis langas.

OS – operacinė sistema.

Serveris – specialios paskirties kompiuteris ir/arba programa, skirta kitų kompiuterių aptarnavimui.

SQL – programavimo kalba (angl. Structured Query Language).

URL – interneto svetainės adresas.

SANTRAUKA

Autorius	Linas Katakinas
Pavadinimas	Demonstracinė elektroninio banko sistema suaugusiųjų finansinio raštingumo ugdymui
Vadovas	Doc. dr. Elena Trepulė
Darbas pristatytas	Vytauto Didžiojo Universitetas, Švietimo Akademija, Kaunas 2021 01 14
Puslapių skaičius	62
Paveikslėlių skaičius	30
Lentelių skaičius	3
Priedų skaičius	1

Sparčiai plėtojant skaitmenines technologijas vis daugiau bankinių paslaugų perkeliamos į virtualią erdvę. Žmonės turintys menkesnius skaitmeninius gebėjimus nemoka ir bijo naudotis elektroniniais bankais. Be to mokymosi procesuose pastebimas praktinių įrankių trūkumas. Šio darbo tikslas – sukūrti demonstracinę elektroninį banką. Jis skirtas išmokti naudotis bazinėmis elektroninio banko paslaugomis ir išdrįsti išmėginti tikrus bankus.

Bakalauro darbe nagrinėjama patirtinio mokymosi nauda finansinio raštingumo ugdyme. Ažvelgiama Lietuvoje veikiančių elektroninių bankų demonstracinės versijos ir mokymosi medžiagos. Taip pat palyginami užsienio šalių bankų sistemų demonstracinės versijos. Aprašomas demonstracinio banko kūrimas, pasirinktos technologijos, testavimas ir pasiekti rezultatai.

ABSTRACT

Author	Linas Katakinas
Title	Demonstrational electronic banking system for adult financial literacy training
Supervisor	Doc. dr. Elena Trepulė
Presented at	Vytauto Didžiojo Universitetas, Švietimo Akademija, Kaunas 2020 01 14
Number of pages	62
Number of images	30
Number of tables	3
Number of appendices	1

With the rapid development of digital technologies, more and more banking services are being moved to the virtual space. People with lower digital literacy are unable and avoid using electronic banking. Also there is a lack of practical tools in the learning process. The aim of this work is to create a demonstrational electronic bank. It's designed to learn the basic services of e-banking and dare to try real banks.

The bachelor's thesis reviews the benefits of experiential learning in educating financial literacy. In addition, demonstration versions and learning materials of Lithuanian electronic banks are reviewed. As well as comparable demonstrational versions of foreign e-banking systems. It also describes the development of a demonstrational bank, the technologies chosen, the testing and the results achieved.

1. ĮVADAS

Vykstant sparčiam skaitmeninių technologijų plėtojimui, su jomis vienokia ar kitokia forma susiduria kiekvienas žmogus. Kompiuteris ar išmanus telefonas jau yra tapusi būtinybe, kurią turi tiek vaikai, tiek suaugusieji. Žmonės tikisi vis geresnių technologijų, mato kaip jos gerina ar lengvina gyvenimą ir buitį. Neabejojama tolesne plėtros sparta ir vis didesniu paslaugų perkėlimu į internetą. Per gana trumpą laiką stipriai pasikeitė prekių, paslaugų, informacijos dalijimosi būdai ir bendravimas. Prisitaikyti prie tokio spartaus virsmo tampa iššūkiu, ypač vyresnio mažiau žmonėms, pas kuriuos jau susiformavę įpročiai. Švietimas yra vienintelis kelias sumažinantis šią atskirtį.

Viena iš sričių, kur pastebimas ypač ryškus pasikeitimas yra mažmeninis finansinis sektorius. Vien per 2020 metų trečią ketvirtį žmonių turimų bankinių kortelių skaičius padvigubėjo (Lietuvos Bankas, 2020). Per paskutinius 14 metų bankinių kortelių skaičius svyravęs tarp 3-4 mln., šiemet pasiekė 7 mln. Tai rodo ne tik, kad žmonės turi po kelias bankines korteles, bet ir parodo vis dažnesnį skaitmeninės bankininkystės naudojimą. Taip pat prie to prisideda spartus bankų skyrių ir bankomatų uždarymas, todėl žmonėms nelieka kitos galimybės, tik išmokti naudotis elektroniniais bankais. Patys bankai suinteresuoti kuo didesniu elektroninių bankų naudojimu, tačiau neskiria pakankamai dėmesio žmonių švietimui kaip jais naudotis. Vyresniems žmonėms (virš 51 metų) tai rimta problema dėl jų menkesnių skaitmeninių įgūdžių. Nevyriausybinių organizacijų ir valstybės globojamos programos organizuoja gyventojų skaitmeninio raštingumo mokymus, kartu ir finansinio raštingumo srityje. Tokių mokymų metu išryškėjo poreikis mokymuose naudoti demonstracinę elektroninę banką kaip patirtinio mokymosi pagrindą. Šiame darbe apžvelgiama patirtinio mokymosi nauda finansinio raštingumo ugdyme, demonstraciniai bankai Lietuvoje ir užsienyje. Aprašomas demonstracinio elektroninio banko kūrimas, pasirinktos technologijos, testavimas, pritaikomumas kitoms šalims, atsiliepimai, bei statistinė informacija.

1.1 Darbo tikslas

Sukurti demonstracinę elektroninio banko sistemą suaugusiųjų finansinio raštingumo ugdymui.

1.2 Uždaviniai

1. Apžvelgti patyriminio mokymosi naudą finansiniam raštingumui.
2. Apžvelgti Lietuvoje veikiančius demonstracinius elektroninius bankus.
3. Sukurti ir ištestuoti demonstracinio banko sistemą.
4. Apžvelgti sistemos naudojimo rezultatus ir pritaikomumą kitoms šalims.

2. PATYRIMINIO MOKYMOSI NAUDA FINANSINIAM RAŠTINGUMUI.

2.1 Patyriminis mokymasis.

Susiduriant su naujomis technologijomis ar skaitmeninėmis aplinkomis sunku jas perprasti vien tik teoriniais metodais. Skaitmeninės technologijos kuriamos taip, kad jas naudotų kuo daugiau žmonių. Siekiama jas supaprastinti ir padaryti aiškesnes, tačiau turint menkesnius skaitmeninius gebėjimus iššūkiu tampa net paprasti veiksmai. Kiekvieno besimokančiojo noras yra kuo greičiau išbandyti naujas technologijas ar bent jau pamatyti kaip jos veikia. Labai svarbiu tampa patirtinis mokymasis. Išbandyti, suprasti ir susigyventi su naujomis technologijomis yra būtina norint išnaudoti naujovių teikiamas galimybes.

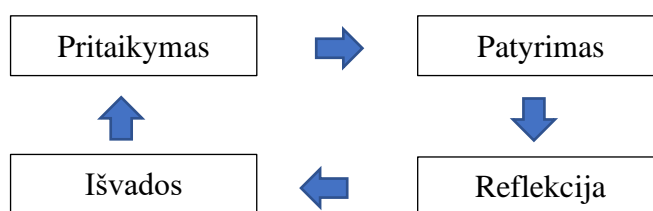
Norint suprasti kas yra patirtinis mokymasis, būtina nusistatyti kas yra patirtis. Patirtį G. Jackūnas (2008) apibrėžia kaip specifinį semantinį išankstinį supratimą. Jis pabrėžia, kad svarbu, nemaišyti patirties su patyrimu. Patyrimą apibrėžė kaip kažkokio veiksmo rezultata, kurį suvokia iškart, kontekstą interpretuoja iš karto, vertindamas esančias aplinkybes. Patirtis atsiranda iš patyrimo. Tai reikalauja apmąstymo, atsiribojimo nuo staigių emocijų ir platesnio suvokimo. Patyrimas susiformuoja iš pasikartojančios patirties, išskiriant tai kas yra aktualiausia asmeniui ar bendruomenei. Patirties pastovumą ar kaitą lemia patyrimo pobūdis. „Patyrimė besireiškiančios prasmės yra konkrečios, partikuliarios, fenomeniškos, kontekstiškos, o patirtinės prasmės yra apibendrintos, abstrahuotos, dekontekstualizuotos bei defenomenalizuotos“ (Jackūnas, 2008). Patirtis leidžia interpretuoti patirtus reiškinius kaip suprantamus ar nesuprantamus. „Reikėtų kuo aiškiau ir nuoširdžiau pripažinti, kad mokymas vyksta patiriant, pasinaudojant patirtimi ir dėl patirties. Tuomet bus vis svarbiau suformuluoti aiškia koncepcija, kas yra patirtis“ (Dewey, 1938). Patirtimi suformuojama praktika. Patirtį mokymosi procesuose akcentavo daugelis mokslininkų. Kaip esminį žinių konstravimo pamatą jį išskyrė Dewey (1938), Kolb (1984), Knowles (1975), Boud ir Miller (1996). Gana sunku tiksliai apibrėžti patyriminio mokymosi paaiškinimą, tačiau neretai praktikoje naudojamas toks apibrėžimas: „Patirtinis ugdymas – tai organizuotas procesas, kurio metu tiesioginio patyrimo dėka dalyviai konstruoja žinias ir įgūdžius, kurias vėliau gali pritaikyti gyvenime. Patirtinis ugdymas įvyksta tuomet, kai asmuo įsitraukia į įvairias veiklas, po to apmąsto savo patirtį ir šios analizės dėka pasiekia naudingų įžvalgų, kurias integruoja į savo besikeičiantį mąstymą bei elgesį“ (Association of Experiential Education, 1995). Patirtinis mokymas gali būti apibūdinamas kaip ugdymo strategija, kur esminis mokymosi išteklius yra patirtis, įgūdžiai susiformuoja iš tiesioginės asmeninės patirties. Pastebima, kad patirtis gali būti suformuojama betarpiškai stebint kitų patirtis (Association of Experiential Education, 1995). Pabrėžiama, kad patirtinio mokymosi pamatas yra ne kitų įperšama nuomonė, o asmeninis savo ar kitų patirčių vertinimas ir supratimas. Todėl toks

mokymasis nėra sąlygotas iš anksto nustatytų taisyklių ir formulių, kiekvienas individas sukuria savo „pasaulį“ besidalindamas patirtimi. Toks mokymosi būdas yra labai asmeniškasis ir autentiškasis. Labai svarbiu tampa aktyvus dalyvavimas ir įsitraukimas, nes nėra prasmės tikėtis automatinio žinių priėmimo, kuomet reikalingas aiškus įsisąmonimas. Tik stipri asmeninė paskata keičia elgesį.

Patirtinis mokymasis yra gana abstrakti sąvoka, sujungianti įvairius mokymosi būdus ir metodus. Dažnai juo apibūdinamos mokymosi veiklos, kur didesnis dėmesys skiriamas besimokančiojo asmenybei, o ne mokymosi turiniui. Turinys nėra koncentruotas į vieną apibrėžimą, o pateikiamas kaip įvairių požiūrių visuma.

Patirtinis mokymasis yra neatskiriama modernaus švietimo dalis, ypač neformaliajame švietime. Jis lengvai taikomas kartu su kitomis mokymo(si) formomis ir strategijomis (pavyzdžiui su probleminiu, autentišku, bendradarbiaujančiu ir kt.).

Nagrinėti patirtinį mokymąsi įprasta nuo D. A. Kolb sudaryto patirtinio mokymosi rato (žr. pav. 1). „Kolbo patirtinio mokymosi metodika apibendrina patirtinio mokymosi struktūrą, suskirstydama mokymosi procesą į keturis vienas po kito einančius blokus: patirtį, jos aptarimą ar refleksiją, apibendrinimą ir iš jo kylantį išvadą bei šių išvadų pritaikymą naujai patirčiai“ (Jakubė, 2012).



Pav. 1. D. A. Kolbo mokymosi ratas

Pats D. A. Kolb ratą sudėrė remdamasi pastebėjimu, kad „individas gali labai daug pasimokyti iš savo veiklos, nes sąmoningai ar nesąmoningai pereidamas nuolatinio ciklo etapus, laiko spaudžiamas, jis yra priverstas daryti sprendimus. Čia mokymosi šaltinis yra patirtis, ir tokie metodai, kaip diskusijos, aptarimai, problemų sprendimai, padeda „įdarbinti“ besimokančiojo patirtį. Šitoks mokymas leidžia įgyti patirties ir, kaip edukacinis procesas, yra nenutrūkstamas spiralinis veiksmas“ (Grigaitienė, 2005). Cikliškumas yra labai svarbus mokymosi veiksnys. Žmogus mokymosi procesą gali pradėti bet kuriame ciklo etape, tačiau toliau visi likę etapai pereinami iš eilės ir kartojami. Patyrimo ratas tapo plačiai naudojamu patirtinio ugdymo modeliu. Jis susideda iš tokių dalių:

Patyrimas. Tai konkreti veikla, kuri yra viso mokymosi pagrindas. Patyrimas suteikia išteklius tolimesniems veiksams. Svarbu atkreipti dėmesį, kad patyrimas gali turėti tiek teigiamą, tiek neigiamą atspalvį, jie nėra vienpusiai. P. Godvadas, G. Jasienė ir A. Malinauskas (2014) išskyrė tris skirtingas būsenas: komforto, įtampos/augimo, panikos. Visos trys būsenos yra susijusios su

ugdymu. Ugdymas formuojasi pereinant iš komforto būsenos į augimo. Nuolatinis toks perėjimas praplečia komforto ribas ir tokiu būdu žmogus tampa stipresnis, užtikrintesnis savimi ir galintis abstrahuoti naujas žinias. Perėjimas per būsenas automatiškai sukelia savirefleksines mintis, verčia pamatyti save iš šono, stebėtis ir analizuoti savo jausmus ir elgesį.

Refleksija. Pagal dabartinės lietuvių kalbos žodyną, refleksija – tai savo psichikos stebėjimas ir mąstymas. Refleksija yra susidariusios įtampos išdava. Patyrimai transformuojasi į patirtį pereidami refleksijos etapą. Veiklos metu žmogus apmąsto tai kas vyksta dabar, remiasi anksčiau įgytomis patirtimis ir jausmais, derina su teorinėmis žiniomis. Refleksijoje apie veiksmą apmąstomi taikyti veiksmai. Net neįprastose situacijose ieškoma skirtumų ir panašumų, analizuojamos pasekmės. Tokiu būdu gimsta naujos idėjos, jos yra analizuojamos ir įsisavinamos. Ypač svarbi refleksija yra mokantis savarankiškai. Labai svarbus netik patyriminis eksperimentavimas, bet ir refleksijos eksperimentai, naujų įžvalgų konstravimai. Refleksija yra būtina norint išmokti mokytis ir tai ypač pasireiškia savivadžiam mokymesi. Labai svarbu išmokti atpažinti savus trūkumus ir emocijas ir sugebėti pažvelgti į save iš šono.

Išvados. Patyrimas įvardintas refleksijose veda prie išvadų formulavimo. V. Lukošienė (2011) išskiria tris įtakos šakas skatinančias besimokančiųjų tobulėjimą: siekimas to paties tikslo ateityje; ankstesnio tikslo konkretinimas ir naujų tikslų kėlimas. Išvadų formulavimas yra būtinas etapas refleksijų apibendrinimui ir taikymui kitoms ateities situacijoms. Išvadų etape žmogus priima patirtį, ją suvokia giliau ir ji tampa sava (Godvadas, Jasienė ir Malinauskas, 2014).

Pritaikymas. Šiame etape asmuo suvokia kodėl patirtis yra tokia, ir žino kaip ją paaiškinti. Jis žino ko išmoko ir gali pritaikyti ateityje. Žmogus gali priimti atsakomybę už patirtis ir drąsiai imtis kitų patirčių, keisti elgesį. Įsisąmoninta patirtis sukuria naujus elgesio modelius, skatina juos pakartoti ir taikyti įvairiose srityse. Pažymėtina, kad asmuo gali pats nuspręsti ar taikyti susikurtus naujus elgesio modelius ar eksperimentuoti toliau ir tikrinti naujas iškilusias hipotezes.

Patyriminis mokymas plačiai naudojamas švietime, nes jo ypatumai labai dera su tokiais mokymosi principais: nuolatinis naujų žinių ir patirčių kaupimas; individualiu žinių konstravimu turimos patirties pagrindu; nuolatinis cikliškumas; struktūruotu pagrindu; teorinės medžiagos ir praktikos sinteze; naudojamų priemonių įvairumu patirčiai gauti. Kiekvienas besimokantysis ugdo savo mokėjimo mokytis kompetenciją.

„Be saugumo jausmo suteikimo, egzistuoja dar vienas svarbus aktyvaus dalyvavimo veikloje stimulus – veiklos patrauklumo ir įdomumo principas, kuris yra vienas iš galingiausių motyvacijos veiksnių. Taikydamas motyvacinio ir emocinio nuteikimo strategijas, dėstytojas gali ne tik padrašinti ugdytinius siekti naujos patirties, bet ir paprasčiausiu pažadu, jog jis darys viską, kad ta tema, kuri siejasi su gana menka ar net neigiama ankstesne patirtimi, taptų iššūkiu ir pozityviai nuteiktų besimokančiuosius“ (Grigaitienė, 2005). Greičiausiai išmokstame tai kas yra aktualu. Patyriminis

mokymosi būdas netik leidžia pabandyti dabar ir čia, bet ir suprasti, bei įsisąmoninti ko gali reikėti ar ką reikėtų ieškoti.

G. Strazdienė, R. Geležinienė (2007) pabrėžia, kad patirtinio mokymosi metodai sumažina mokytojo poreikį ir perleidžia mokymosi kontrolę pačiam besimokančiajam. Vis dėlto kiti autoriai pabrėžia mokytojo svarbą, ypač refleksijos etapuose. Apibendrinant galima išskirti tokius pagrindinius punktus į ką turėtų atkreipti dėmesį tiek pats besimokantysis, tiek mokymų organizatoriai:

- Tiesioginis patyrimas. Esminis faktorius yra pačio asmens patyrimas ir susiformavusi patirtis. Mokymosi iš kitų patirties neužtenka, ji tik gali papildyti mokymosi procesą.
- Žinių konstravimas. Žmogus pats susikonstruoja mokymosi organizavimo būdus, terminus, susidėlioja autentiškas įžvalgas ir išvadas.
- Patirties analizė. Patirtinis mokymasis neįsivaizduojamas be apmąstymų ir reflektavimo. Skirtingi žmonės skirtingai supranta ir vertina situacijas, todėl jų mokymosi patirtis yra skirtinga.
- Išvadų įgalinimas. Perėjus refleksijos etapą, patirtiniame mokymesi labai svarbu nesustoti ir suformuotas įžvalgas ir išvadas pritaikyti tolimesnėje veikloje, keisti savo elgesį ar reakcijos būdus.
- Organizuotas procesas. Kryptingas ir aktyvus proceso organizavimas padidina mokymosi efektyvumą.
- Kartoavimo sąlygų sudarymas. Skirtumai ir pasiekimai geriausiai pastebimi per kartojimą ir lyginimą.
- Išorinės patirties pritaikymas veiklose. Svarbu ne tik matyti tiesioginę mokymosi naudą, bet ir atpažinti (reflektuojant) kas ugdymo procese padėjo ir kas trukdė.
- Atvirumas. Ypač grupinėse veiklose būtinas bendradarbiavimas, nuoširdumas ir kitų patirčių priėmimas. Patyrimas konstruojamas lyginant ir gretinant savo asmeninį matymą su kitų žmonių matymu.

Planuojant demonstracinį elektroninį banką buvo atsižvelgiama į patyriminio mokymosi aspektus. Atsižvelgiant į juos numatyta galimybė naudotis demonstraciniu banku ne tik grupinių mokymu metu, bet ir individualiai, patogioje aplinkoje ir norimu laiku. Vienu svarbiausiu reikalavimu tapo būtinybė pačiam besimokančiajam surasti reikiamas funkcijas, įprasti prie skaitmeninės aplinkos, bei veiksmų iteracijų. Taip pat sudarytos galimybės bandyti finansines operacijas su pagalba (paaiškinimais) ir be jos. Be to sudarytos bendradarbiavimo, grupinių užsiėmimo ar dialogo galimybės.

2.2 Finansinis raštingumas

Finansinio raštingumo ugdymo nauda neabejojama, o sparčiai plečiantis finansinėms priemonėms ir technologijoms, tampa itin svarbiu greitas jų įsisavinimas. Ekonominio bendradarbiavimo ir plėtros organizacija (angl. OECD – Organisation for Economic Co-operation and Development; toliau EBPO) teigia, kad vis sudėtingesnėje visuomenėje, finansinis raštingumas yra laikomas esminiu ir todėl būtinu švietimo elementu. Be abejonės su finansais susiduria kiekvienas žmogus, daugėja įvairių finansinių produktų ir paslaugų, kinta jų formos, todėl finansinių operacijų išmanymas tiesiogiai sąlygoja gyvenimo kokybę. Finansinis raštingumas glaudžiai susijęs su mokymusi visą gyvenimą ir patirtiniu mokymusi.

Vieni autoriai glaudžiai apibrėžia finansinį raštingumą, kaip finansinių produktų išmanymą (Hathaway ir Khatiwada, 2008). Kiti apibrėžia įvairiau, labiausiai akcentuodami asmens gebėjimus ir asmeninį veikimą: „Pagrindinių ekonominių ir finansinių sąvokų žinojimas, taip pat gebėjimas tas žinias panaudoti kartu su kitais finansiniais įgūdžiais, siekiant efektyviai naudoti finansinius išteklius visą gyvenimą dėl finansinės gerovės“ (Hung, Parker ir Yoong, 2009). EBPO pažymi, kad finansinis raštingumas glaudžiai siejasi su galimybe veikti. Finansinių sąvokų žinojimas ir supratimas apie rizikas bei motyvacija ir pasitikėjimas savo jėgomis yra veiksmingų sprendimų pamatas leidžiantis pagerinti asmens ir visuomenės gerovę. B. Frijns, A. Gilbert, A. Tourani-Rad (2013) savo tyrimuose atskleidžia patirties svarbą finansiniame raštingume. Jų išvados rodo, kad žmonės, turintys didesnę finansinę patirtį, įgyja daugiau finansinių žinių per savišvietą ir imliau naudojami įvairiomis švietimo programomis.

Nors su finansais susiduria kiekvienas žmogus nuo vaikystės, tačiau gilesnis finansinis supratimas nesuformuojamas, dažniausiai dėl menko tėvų ar senelių finansinio raštingumo. Nesant kokybiško švietimo šeimoje, ypač sustiprėja švietimo institucijų ar programų reikšmė. Hilgert (2003) pažymi, kad vienos pagrindinių finansinio švietimo programų nesėkmių priežastys yra drąsos trūkumas panaudoti žinias, palaikymo trūkumas ir nepakankamas įsisaugojimas. Dažniais atvejais finansinių instrumentų išbandymą riboja seni šeimos įsitikinimai.

Apibendrinama įvairių mokslinų tyrimus EBPO (2014) išskiria tris sritis apibrėžiančias finansinį raštingumą:

1. **Žinios ir supratimas.** Norint priimti sprendimus, būtina išmanyti terminologiją, suprasti procesus ir paprasčiausiai mokėti skaičiuoti. Žinios turi leisti numanyti pasekmes ir žmonių elgesį kokios būna priėmus vienus ar kitus sprendimus. Tačiau pakeisti elgesį vien tik žinių neužtenka (Hilgert, 2003; Perry ir Morris, 2005), todėl reik atsižvelgti į kitus aspektus.

2. **Įgūdžiai ir elgesys.** Tam, kad pakeisti savo elgesį būtina praktiškai įvaldyti naujus procesus. Neužtenka vien tik žinoti, reikia daryti nuolatos.

3. **Požiūris ir pasitikėjimas savimi.** Kad galėtumėte pritaikyti tam tikrus sprendimus už įprastinės situacijos ribų, būtina turėti pasitikėjimo ir drąsos (Bandura, 1997). Būtina išsiauginti reikiamą motyvaciją.

2.3 Finansinį raštingumą lemiantys veiksniai ir poreikiai

Pagal G. Kvieskienę (2016) kompleksinių (holistinių) tyrimų nagrinėjančių finansinio raštingumo veiksnius trūksta, tačiau galima pastebėti įvairiose šalyse atsikartojančius dėsningumus. Finansinis raštingumas didesnis pas turinčius aukštesnį išsilavinimą, gyvenančius mieste, nepriklausantiems tautinėms mažumoms, vyrams, vidutinio amžiaus. Jaunų asmenų finansinio raštingumo lygis varijuoja įvairiose šalyse, tuo tarpu vyresnių stabiliai žemesnis. Finansinį raštingumą lemiančius veiksnius galima suskirstyti į tris sritis: geografinį, demografinį ir socialinį (ekonominį).

Geografiniai veiksniai. PISA 2018 finansinio raštingumo tyrimas parodė, kad kuo gyvenamoje vietovėje yra mažiau žmonių, tuo mažesni raštingumo lygiai. Atitinkamai tuo daugiau pastangų turi įdėti žmogus pabaigęs mokyklą.

Demografiniai veiksniai. M. Navicko, T. Gudaičio ir E. Krajnakovos (2014) tyrimas parodė kad tik 53,8% jaunų šeimų ūkių yra finansiškai raštingi. Finansinis raštingumas pagal amžių išsiskiria į tris dalis, nepilnamečius, suaugusius ir vyresnio amžiaus. Suaugusiųjų raštingumo pikas yra apie 40 metų (Cole ir Liebenberg, 2010). Iki tol tolygiai didėja, vėliau mažėja. Nepilnamečių ir vyresnio amžiaus žmonių menkas finansinis raštingumas susijęs su praktiniu finansinių paslaugų išbandymų stoka.

Socialiniai (ekonominiai) veiksniai. Be abejonės, aukštesnį finansinį raštingumą turi aukštesnių pajamų žmonės, kurie gali išbandyti įvairius produktus, gali lengviau rizikuoti turimomis lėšomis. Tokiu būdu ir supratimas apie juos yra didesnis. Didesnės pajamos ir finansinis raštingumas abipusiai susijęs, be vieno nebūtų ir kito, tačiau ne absoliučiai. Negalima teikti, kad kuo didesnė pajamos, tuo aukštesnis finansinio raštingumo lygis. Pajamos turi pasiekti tam tikrą lygį, vėliau jos nebedaro esminio poveikio raštingumui. Kalbų mokėjimas, didesni gebėjimai mokytis ir platesnis pasaulio suvokimas palengvina įvairių finansinių instrumentų suvokimą. Dalinimasis žiniomis ir aktyvesnis socialinis gyvenimas padeda praplėsti asmeninį akiratį ir suteikia motyvacijos išbandyti finansines paslaugas.

Lietuvos bankas dar 2012 metais išskyrė tokias pagrindines kliūtis trukdančias priimti finansinius sprendimus:

- Finansinių žinių stoka. Ypač vyresni ir mažamečiai žmonės nežino kasdieninių finansinių procesų (nevertinant paprasto, fizinių pinigų naudojimo).

- Naujų finansinių priemonių sudėtingumas. Technologijų pažanga ir spartus elektroninių paslaugų diegimas apunkina menkesnius gebėjimus turinčių žmonių prisitaikymą prie pasikeitusių sąlygų. Be to naujausi, išvestiniai, finansiniai produktai yra sudėtingi, reikalaujantys plataus finansinės sistemos supratimo. Suprasti finansinių produktų taisykles neužtenka net vidutinio finansinio raštingumo, dažnai reikia specialistų konsultacijos. Be to produktai patys savaime turi įvairių sąlygų ir reguliavimų, kuriuos reikia išmanyti. Taip pat aprašymai surašyti specifine kalba, kurią eiliniam vartotojui sunku perprasti. Sparti bankinių paslaugų skaitmenizacija ir uždaromi bankų skyriai verčia žmones naudotis elektroniniais bankais ir finansinėmis paslaugomis internete. Ypač mažesniuose Lietuvos miestuose nebeliko bankų skyrių ir net mažinamas bankomatų kiekis. Daug nepatogumų sukelia 65-74 metų amžiaus žmonėms, iš kurių internetu naudojasi tik 40%, todėl jiems labai menkai prieinamos bankinės paslaugos (Lietuvos statistikos departamentas, 2019).
- Informacijos ir pasiūlos gausa. Per didelis pasirinkimas mažina žmonių motyvaciją pasirinkti finansinį produktą ar paslaugą. Dažnai trūksta gebėjimų susirinkti informaciją.
- Laiko efektas ir spaudimas. Dažname finansiniame produkte esminį vaidmenį atlieka laikas. Žmonėms trūksta supratimo kaip keičiasi rinkos, kaupiamos pensijos ar išmokamos pensijos ir kokį poveikį tai daro asmeniniams finansams.
- Per didelis arba per mažas pasitikėjimas. Dažnai žmonės nuvertina arba neįvertina savo padėties priimdami sprendimus. Nelinkę kritiškai žvelgti į finansinę situaciją ar socialinius bei ekonominius veiksnius.
- Savikontrolės stoka. Finansiniai sprendimai dažnai priimami emociniu pagrindu. Neįvertinamas įsipareigojimų svarbumas ir vengiama ieškoti alternatyvių pasiūlymų ar sprendimų. Lietuvos gyventojams sunkiai sekasi tvarkyti asmeninius ar namų ūkio finansus. 34 proc. apklaustųjų nurodė kad nenusistatė biudžeto (Lietuvos bankas, 2020) ir tik 37 proc. pasitiki savo jėgomis valdydami šeimos ar asmeninį biudžetą. Nors apie finansinių paslaugų ir produktų įvairovę žino dažnas Lietuvos gyventojas, tačiau jais naudojasi retas. Apie einamąją sąskaita žino 84 proc. gyventojų, o turi - 80 proc. Apie akcijas žino 59 proc., bet naudojasi tik 2 proc. (Lietuvos bankas, 2020).

2.4 Finansinio švietimo padėtis

Ilgą laiką finansinio raštingumo švietimas susidarė iš atskirų pamokų mokyklose, neformaliojo ir savaiminio ugdymo. 2012 metais Lietuvos bankas sukūrė finansinio švietimo koncepciją ir pradėjo organizuoti įvairias veiklas. Anksčiau vykę finansiniai mokymai apsiribodavo pavienėmis nevyriausybinėse organizacijose, valstybinių institucijų ir privačių asmenų iniciatyvomis. Šiuo metu vykdomas 2017 metais Finansų ministerijos kartu su Švietimo, mokslo ir sporto

ministerija, Sodra ir Lietuvos banku susitartas ir iki 2021 metų įgyvendinamas finansinio švietimo planas. Jis orientuotas į vaikų finansinį raštingumą formaliojo ir neformaliojo ugdymo įstaigose, suaugusiųjų žinių apie finansinius produktus ir paslaugas padidinimą, mokamų mokesčių naudą ir kaupimą ateičiai. Nevyriausybiniam sektoriuje išsiskiria Lietuvos „Junior Achievement“ iniciatyva siūlanti įvairias finansinio raštingumo programas mokiniams. Įvairiose veiklose aktyviai veikia dauguma Lietuvos aukštųjų mokyklų. Projektas „Prisijungusi Lietuva“ vykdo mokymus visoje Lietuvoje ugdydamas skaitmeninį raštingumą. Jų tikslinė grupė apie 500 tūkst. asmenų, daugiausia regionuose. Projekte vykdomuose mokymuose kelios mokymo programos skirtos didinti finansinį raštingumą. Privačiame sektoriuje yra organizuojami įvairūs neformalaus švietimo finansinio raštingumo kursai. Taip pat aktyviai veikia bankai. Swedbank įkūrė Finansų institutą ir laboratoriją, kurios skirtos vaikų ir suaugusiųjų finansinio raštingumo ir verslumo ugdymui. SEB bankas inicijuoja jaunimo veiklas ir programas.

Nors valstybė ir privatus sektorius deda pastangas stiprinant finansinį švietimą, tačiau Lietuva vertinama žemiau nei vidutiniškai. Lietuvoje atliktas 18-30 metų žmonių finansinio raštingumo matavimas (Navickas et al., 2014) buvo arčiausiai A. Lusardi ir S. O. Mitchell (2004–2013), bei EBPO (2012) atliekamų tyrimų. Atsakymai nurodė, kad apie pusę respondentų pavyko įvykdyti užduotis ir tai rodė vidutiniškus rodiklius. Klausimus apie investavimą ar infliaciją atsakė tik 60%, tuo tarpu JAV 75% gyventojų atsakė teisingai (Lusardi ir Mitchell, 2004–2013). Šio tyrimo vidurkis parodė 53,8% finansinį raštingumą tarp 18-30 metų žmonių Lietuvoje. A. Lusardi ir S. O. Mitchell tyrimai atskleidė gana žemą finansinį raštingumą visame pasaulyje. EBPO atlikti tyrimai (2017) nustatė žemą suaugusiųjų finansinio švietimo lygį ir rekomendavo finansinio raštingumo švietimą pradėti kuo ankstesniame amžiuje. Iš galimo 21 taško, Lietuva surinko 13,2 taško, tuo tarpu EBPO šalių vidurkis – 13,7 taško.

Kitas svarbus aspektas reikalaujanti spartesnio finansinio švietimo yra sparti skaitmenizacija ir banko paslaugų perkėlimas į elektroninę erdvę. Žmonėms, turintiems menkesnius skaitmeninius gebėjimus, tai tampa rimta problema, sumažina finansinių paslaugų prieinamumą ir didina atskirtį. Atsiskaitymas bankinėmis kortelėmis nuosekliai didėja kiekvienais metais. Tuo tarpu bankomatų ir kortelių skaitytuvų skaičius gana ženkliai krenta. 2012 metais Lietuvoje buvo daugiausia, 1610 bankomatų, o 2020 metais jų liko tik 894 ir tai yra mažiau nei 2006 metais kai pradėta rinkti tokia statistika (Lietuvos bankas, 2020). Taip pat pastebimas labai ryškus mokėjimų paslaugas teikiančių vietų mažėjimas, ypačingai mažuosiuose miestuose. Nuo 2009 metų kuomet buvo 967 tokios vietos, 2020 m. liko tik 407 ir toliau mažėja (Lietuvos Bankas, 2020). Panaši tendencija ir su grynujų pinigų naudojimu, nuo 2016 metų, kasmet grynujų pinigų operacijų sumažėja apie 10 proc. (Lietuvos Bankas, 2020). Per pastarąjį dešimtmetį grynujų operacijų sumažėjo 7 kartus, tai rodo būtinybę kuo greičiau įsisavinti elektroninius mokėjimus ir kitas bankines operacijas. Vis dėlto tai nėra paprasta

padaryti, nes tik 56 proc. Lietuvos gyventojų turi minimalius ar aukštesni skaitmeninius gebėjimus (Eurostat, 2019). Tai rodo, kad didelė dalis žmonių yra priklausomi nuo kitų žmonių pagalbos ir nesijaučia užtikrinti valdydami savo finansus.

2.5 Finansinio švietimo rekomendacijos

Apibendrinant finansinio švietimo situaciją ir kitų šalių patirtis galima išskirti tokius pagrindinius finansinio švietimo plėtojimo principus:

- Finansinis švietimas turi būti teikiamas sąžiningai ir nešališkai.
- Vyriausybinių ir nevyriausybinių iniciatyvos turėtų būti koordinuojamos.
- Švietimo programos turi atsižvelgti į nacionalines valstybių aplinkybes.
- Turėtų būti skatinamas finansinių įstaigų vaidmuo švietime, svarbu užtikrinti koordinaciją ir nešališkumą.
- Finansinio švietimo programos turėtų būti rengiamos atsižvelgiant į jų tikslinės auditorijos poreikius ir finansinio raštingumo lygį, taip pat turėtų atspindėti, kaip jų tikslinė auditorija nori gauti finansinę informaciją. Mokymų programos ir medžiaga turėtų konkrečiai atliepti vartotojų pogrupių poreikius, atsižvelgti į individualias aplinkybes.
- Finansinis švietimas turėtų būti laikomas visą gyvenimą trunkančiu, tęstiniu procesu, visų pirma siekiant atsižvelgti į padidėjusį rinkų sudėtingumą, skirtingus poreikius skirtinguose gyvenimo etapuose ir vis sudėtingesnę informaciją.
- Išskiriamas būtinas būsimų pensininkų informuotumas apie būtinybę įvertinti dabartinių valstybinių ir privačių pensijų sistemų finansinį tinkamumą ir didinti motyvaciją imtis tinkamų veiksmų.
- Dar mokykloje turėtų būti gaunamas visavertis finansinis švietimas.
- Reikėtų skatinti specialias interneto svetaines, kad visuomenei būtų teikiama tinkama, vartotojui patogi finansinė informacija. Turėtų būti plėtojamos nemokamos informacijos paslaugos. Turėtų būti skatinamos vartotojų, profesionalių ar kitų organizacijų įspėjimo sistemos dėl didelės rizikos klausimų, galinčių pakenkti finansų vartotojų interesams (įskaitant sukčiavimo atvejus).
- Plėtoti tarptautinį bendradarbiavimą finansinio švietimo srityje, keistis informacija apie naujausią nacionalinę patirtį finansinio švietimo srityje.
- Finansų institucijos turėtų teikti informaciją, kuri yra aiški ir patogi vartotojams. Atsisakyti mažų šriftų ir komplikotos dokumentacijos politikos.
- Finansų įstaigos turėtų būti skatinamos tobulinti savo darbuotojų finansinio švietimo mokymą ir parengti elgesio kodeksus taip, kad būtų teikiamos bendros konsultacijos dėl investicijų ir skolinimosi, neakcentuojant konkretaus produkto. Finansų įstaigos turėtų būti skatinamos

aiškiai atskirti finansinį švietimą nuo finansinės informacijos ir „komercinių“ finansinių konsultacijų.

- Skatinti tobulinti esamas finansinio švietimo programas, metodikas. Reikėtų apsvarstyti galimybę oficialiai pripažinti finansinius švietimo programas, kurios atitinka atitinkamus kriterijus. Investuoti į mokytojų ir lektorių išsilavinimą ir kompetencijų didinimą. Plėtojamos instruktorių rengimo programos.

Atsižvelgiant į šiuos principus buvo kuriamas demonstracinis elektroninis bankas, kaip finansinio švietimo priemonė, kuri gali būti naudojama nepriklausomai nuo institucijų, laisvai prieinama, nereikalaujanti licencijų ar papildomų išteklių.

3. DEMONSTRACINIŲ ELEKTRONINIŲ BANKŲ APŽVALGA

3.1 Lietuvos elektroniniai bankai

Demonstracinis elektroninis bankas yra skirtas Lietuvos gyventojams, todėl pagrindinė jo paskirtis yra išmokinti ir įgalinti naudotis Lietuvoje veikiančiais elektroniniais bankais. Prieš projektuojant demonstracinį banką būtina įvertinti realių bankų padėtį, jų suteikiamas mokymosi galimybes ir funkcionalumą, bei vartotojo sąsaja. Apžvalgai pasirinkti didžiausi Lietuvoje veikiantys bankai (Lietuvos bankas, 2020): Swedbank, SEB bankas, Luminor bankas ir Šiaulių bankas.

Swedbank. Būdamas didžiausiu Lietuvoje veikiančiu banku, Swedbank neturi sukūręs realiai veikiančios demonstracinio banko versijos. Pagalbinė ir mokomoji medžiaga yra pateikiama animuotais vaizdo įrašais. Vaizdo įrašai yra atraktyvūs ir apimantys visas pagrindines bankininkystės temas, tačiau naudotojui nesudaroma galimybė pamatyti kaip atrodo bankas prieš tampant banko klientu. Naujam naudotojui reikia fiziškai atvykti į banką arba konsultuotis telefonu dėl iškilusių problemų ar neaiškumų. Swedbank šiek tiek daugiau dėmesio skyrė mobiliajai bankinei aplikacijai, tačiau veikianči bandomoji versija yra neprieinama. Gana sudėtinga surasti mokomąją medžiagą. Nėra greitai randamų nuorodų, todėl, ypač menkesnių skaitmeninių gebėjimų žmonės, gali nerasti informacijos.

SEB bankas. Šis bankas skyrė daugiau dėmesio, todėl turi viešai prieinamą demonstracinę versiją ir mokomąją medžiagą. Ši medžiaga ir demonstracinė versija yra lengvai randama, viešinama startiniame puslapyje prie interneto banko nuorodos. Mokomoji medžiaga pasižymi išsamumu, apima visas bankines funkcijas. Vaizdo įrašų trūkumą kompensuoja gausios iliustracijos. Demonstracinė SEB banko versija atitinka realų banką, todėl yra informatyvi (žr. 2 pav.). Tačiau neturi pilnai veikiančių funkcionalumų, pavyzdžiui, suteikiama galimybė formuoti pavedimą ir bankas parodys, kad jis sėkmingai išsiųstas, tačiau pinigų kiekis nepasikeis ir naudotojas nematys kas atsitiko su pinigais. Taip pat dažnai pasitaiko niekur nevedančių mygtukų. Šie esminiai trūkumai neleidžia naudoti kaip pilnavertės mokymosi priemonės. SEB banko demonstracinė versija labiau skirta vizualiniam supažindinimui su elektroniniu banku.

Insert keyword Ieškoti 8 Pranešimai ir e. sąskaitos | Kontaktai | Vardenis Pavardenis

SEB Demo

Pradžiai Mokėjimai Sąskaitos ir kortelės Taupymas ir pensija Kreditai ir lizingas Vertybiniai popieriai Paraiškos Kitos paslaugos ir nustatymai

[Naujas mokėjimas](#)

Patvirtinti mokėjimus 1 mokėjimas (-ai) laukia patvirtinimo

Einamosios sąskaitos	Balansinis likutis	Rezervuota	Disponuojamas likutis	
SEB BANKAS	1 524,27	113,60	1 411,25	EUR
LT00000000000000000000 VARDENIS PAVARDENIS SĄSKAITA EUR	123,10	113,60	9,50	EUR
LT00000000000000000000 VARDENIS PAVARDENIS SĄSKAITA EUR	1 401,75		1 401,75	USD

Nebaigtos paraiškos ir sutartys

Lengvojo transporto lizingas

Kreditas įkeičiant turta 10,00 EUR

Kredito kortelių sąskaitos	Kredito limitas	Balansinis likutis	Rezervuota	Disponuojamas likutis	
LT00 0000 0000 0000 0000 Vardenis Pavardenis sąskaita EUR	2 222,00			1 044,00	EUR
Disponuojamas likutis				1 044,00	EUR

Indėlio pavadinimas	Pabaigos data	Palūkanų norma	Palūkanos	Suma	
► Terminuotasis indėlis	2019.08.20	0,00%	0,00	200,00	EUR
► Kaupiamasis indėlis	Neterminuotas	5,00%	0,00	120,00	EUR
Iš viso terminuotųjų indėlių			0,00	200,00	EUR
Iš viso indėlių			0,00	320,00	EUR

Norėdami pakeisti nustatymus, spustelėkite [Pagrindinio puslapio nustatymai](#)

Pav. 2. SEB banko demonstracinė versija

Luminor bankas. Šis bankas taip pat turi demonstracinę versiją, tačiau be mokomosios medžiagos (žr. 3 pav.). Kaip ir SEB banke galima pamatyti kaip atrodo elektroninis bankas iš vidaus, tačiau neleidžiama atlikti operacijų ir duomenys nėra išsaugomi. Tai pat kai kuri informacija demonstracinėje versijoje yra pasenusi ir neaktuali, be to pasitaiko dingusių paveikslėlių. Nors galima pamatyti kaip atrodo elektroninis bankas ir vidaus, susidaro įspūdis, kad versija pasenusi arba neišbaigta. Pradedančiajam naudotojui reikia kuo paprastesnės ir aiškesnės sistemos, kurios šiuo atveju trūksta. Vienas iš šio demonstracinio banko privalumų yra galimybė matyti visas elektroninio banko funkcijas.

Demonstracinė versija Vardenis Pavardenis Vardenis Pavardenis Laiškai (119) Nustatymai English Atsijungti

Luminor

Sąskaitos/Kortelės Mokėjimai Paskolos Taupymas/investavimas Paraiškos M. bankas El. paslaugos

Pradžia

Greitasis meniu

- Paslaugų krepšeliai
- Pervedimas eurais
- Mokėjimas už paslaugas
- Pervedimas į savo sąskaitą
- Mano ruošiniai
- E. sąskaitos
- Mokėjimo kortelės

Vardenis Pavardenis

Paskutinį kartą jungėtės – 2015.08.13 10:53

Banko sąskaitos

Sąskaita	Valiuta	Kredito dydis	Rezervuota suma	Nuosavos lėšos	Likutis (Likutis EUR)
LT0840XXXXXXXXXX89	EUR	200,00	150,00	1.450,00	1.500,00
LT0840XXXXXXXXXX89	USD	0,00	15.000,00	15.019,00	19,00 (13,90 EUR)
LT0840XXXXXXXXXX80	EUR	100,00	0,00	300,00	400,00
Suma					1.913,90

Sąskaitos išrašas

Pasirinkite sąskaitą... LT0840XXXXXXXXXX80

400,00 – EUR

Laikotarpis Šiandien praėjęs mėnuo šis mėnuo

Rodyti

Banke laikomos lėšos sąskaitose ir indėliai yra draudžiami VĮ „Indėlių ir investicijų draudimas“. Atkreipiame dėmesį, kad apie atvejus, kai lėšos sąskaitose ir indėliai nėra draudžiami ar jiems taikomi draudimo išmokų mokėjimo apribojimai, galite [susipažinti čia](#).

Pav. 3. Luminor banko demonstracinė versija

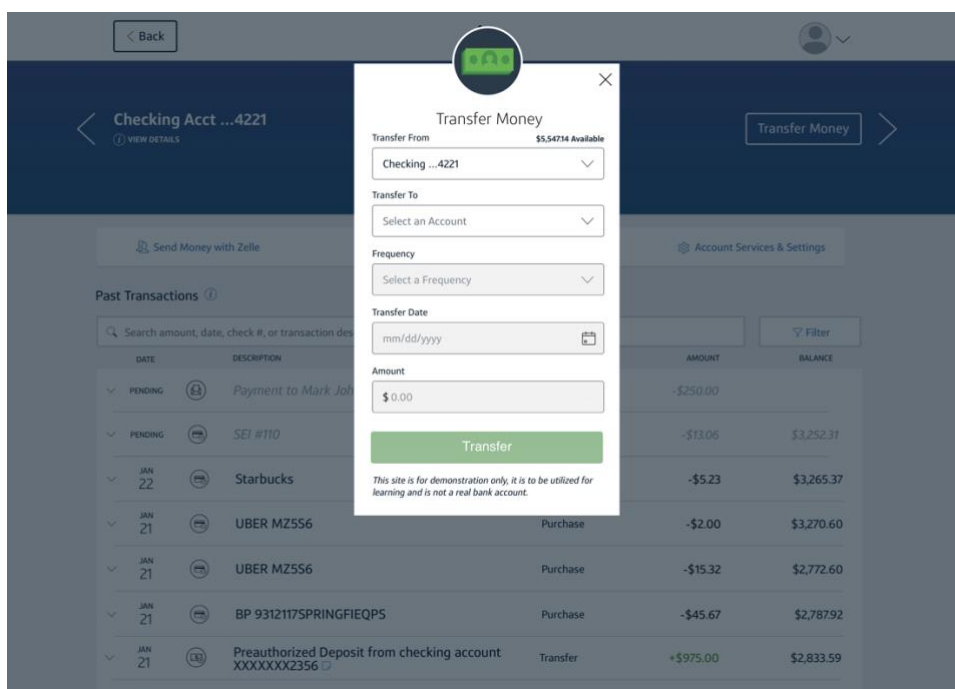
Šiaulių bankas. Šis bankas skyrė mažiausią dėmesį pradedantiesiems elektroninio banko naudotojams. Nėra demonstracinės versijos ir mokomoji medžiaga yra labai minimali, apsiriboja tik atsakymais į dažniausiai užduodamus klausimus be iliustracijų ar vaizdo paaiškinimų.

3.2 Užsienyje veikiantys demonstraciniai elektroniniai bankai

Nors kuriamas produktas skirtas Lietuvos rinkai, tačiau svarbu ir naudinga išnagrinėti užsienio gerąsias patirtis ir naudojamus sprendimus. Reikia pastebėti, kad ir užsienio bankuose nėra savaime suprantama įdiegti demonstracinę versiją, netgi nemažai didžiųjų bankų neturi tokių sprendimų ir apsiriboja tekstinė mokomąja medžiaga ar vaizdo įrašais.

Ready, Set, Bank. Sukurtas JAV, vienintelis iš apžvelgiamų sistemų yra ne tikras bankas, o mokymo platforma orientuota į vyresnio amžiaus žmonių finansinį švietimą (žr. 4 pav.). Tai yra vienintelis pavyzdys, kuris panašus į kuriamą demonstracinį banką savo paskirtimi ir tiksline auditorija. Ši sistema turi gerai paruošta mokymosi medžiagą, tiek tekstinę, tiek vaizdo įrašais. Sistema labai tikslingai orientuota į senjorus. Šios sistemos neigiamas aspektas yra tas, kad neleidžiama laisvai vaikščioti po imitacinį banką, o tik atlikti užduotis pagal vedlį. Pavyzdžiui, kuriant mokėjimą ir užpildžius mokėjimo formą, langai užsidaro ir vietoj sąskaitos apžvalgos

grįžtama į mokymų sistemos vaizdo įrašų sąrašą. Tokiu būdu nesukuriamas realaus banko įspūdis, ir sistema neleidžia individualiai surasti reikiamus kelius ir išbandyti įvairius variantus. Taip pat trūksta galimybės naršyti mokėjimų išrašus, bei neišsaugomi atlikti mokėjimai.



Pav. 4. „Ready, Set, Bank“ mokymų sistema.

Westpac bank. Vienas iš seniausių ir didžiausių Australijos bankų turi aiškia, neperkrautą demonstracinę versiją (žr. 5 pav.). Kaip ir kuriamas demonstracinis bankas, *Westpac* demonstracinė versija yra orientuota į bazines bankines paslaugas (sąskaitų apžvalgą, mokėjimus ir išrašus). Šioje sistemoje neišsaugomas padarytas mokėjimas išrašuose, tačiau yra įgyvendintas sąskaitos likučio sumažėjimas. Taip pat pastebimas pagalbos arba mokomosios medžiagos trūkumas. Ši sistema puikiai iliustruoja ir leidžia išbandyti elektroninį banką, tačiau neturint baziniu finansinio raštingumo žinių jis gali būti per sudėtingas.

DEMO SITE

Accounts

Pay & Transfer

Timeline

Payees

Accounts

All accounts

Account Name	Available	Balance
Westpac Everyday 03-0104-0863752-001	\$20,003.00	\$20,003.00
Holiday Savings 03-0104-0863752-002	\$1,232.32	\$1,232.32
Westpac Bonus Saver PIE 03-0104-0863752-003	Available (\$1 = 1 unit) \$3,402.60	Balance (\$1 = 1 unit) \$3,402.60
Term investment 03-0104-0863752-081		Balance \$5,205.00
Home loan 03-0104-0868752-091	Available -\$274,995.00	Balance \$0.00
Platinum Mastercard ****-0576 Payment due 1 Jan 2021	Available \$431.78	Balance -\$1,568.22
KiwiSaver 186115		Balance as at 2020-12-12 \$20,050.10

+ Open/Apply

hotpoints

hotpoints
You have the freedom to choose with hotpoints Pay*
Total hotpoints as at 21 Jan 2020
7,204

Manage my Cards

Feedback Profile Logout

I want to:

- Quick transfer
- Make a payment/transfer
- Learn to use Westpac One
- Setup and manage alerts
- Notify Westpac of my travel
- Manage my Cards
- Open/Apply

Recent & upcoming [See more](#)

Tomorrow

- to Someone new - \$123.45
- to Johnny - \$9.87
- to Power Co - \$1.45

Today

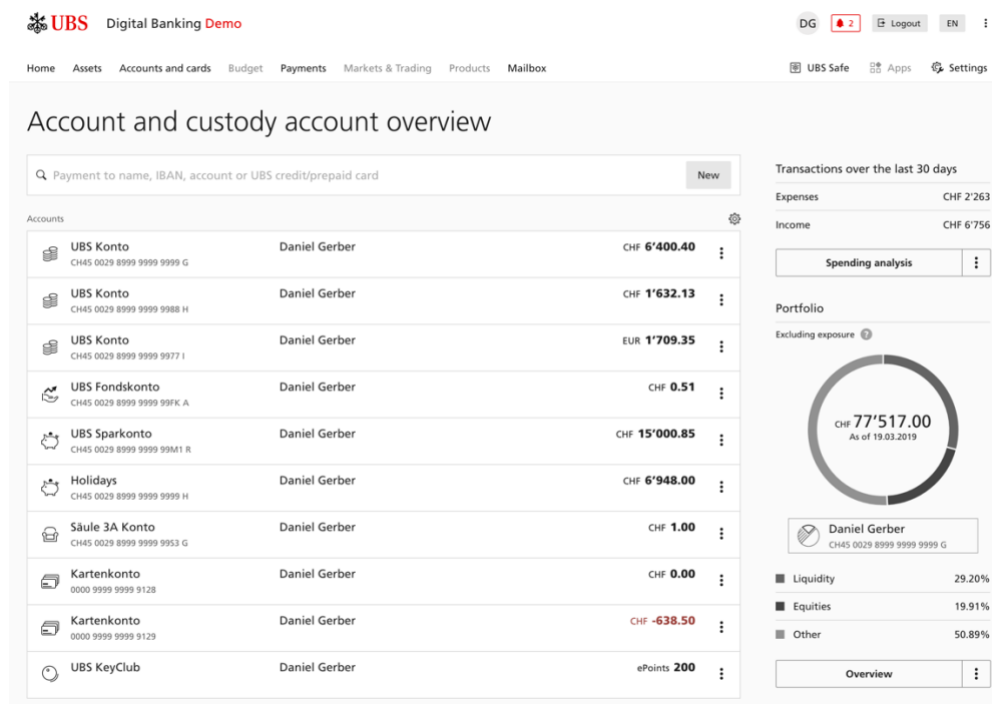
- to Westpac Transaction... - \$256.00
- to Meridian - \$256.00
- to AMALGA TAXIS 3-888... - \$45.00
- to New World Metro - \$66.42
- to New World Metro - \$13.42

Yesterday

- to New World - \$26.50
- to Dick Smiths - \$8.05
- from AUTOMATIC PAYM... + \$72.46

Pav. 5. Westpac banko demonstracinė versija.

UBS bankas. Iš Šveicarijos kilęs pasaulinis bankas turi viešai prieinamą demonstracinę versiją, kurioje apribotos sudėtingesnės funkcijos, o paliktos tik bazinės, aktualios naujiems naudotojams (žr. 6 pav.). Šis bankas taip pat pasižymi pagalbinės medžiagos trūkumu, tačiau gana aiškiai aprašyti būtini užpildyti formų laukeliai. Funkcionalumas apribotas sąskaitų apžvalga, įvairių tipų mokėjimais, išrašais ir pranešimų funkcionalumais, tačiau paliktos neaktyvios nuorodos į sudėtingesnes paslaugas. Pagrindinis trūkumas, kad šiame banke nėra įgyvendintas sąskaitos likučio skaičiavimas atlikus mokėjimą ir mokėjimo išsaugojimas išrašuose.



Pav. 6. UBS banko demonstracinė versija.

Postbank bankas. Šis Vokietijos bankas yra *Deutsche Bank* dalis, įkurtas sujungiant pašto ir bankines veiklas vykdančias įstaigas. Šis bankas specializuojasi į gyventojų aptarnavimą siūlydamas įvairias greitąsias perlaidas, kartu su įprastomis bankinėmis paslaugomis. *Postbank* turi gana plačiai išvystytą demonstracinę banko versiją, tačiau pradedančiajam naudotojui ji yra per daug apkrauta įvairiomis funkcijomis. Šis bankas leidžia priimti ir siųsti pavedimus ne tik įprastu būdu, bet ir įvairiomis perlaidomis, todėl gali susimaišyti teikiamos paslaugos. Taip pat palikta nemažai blaškančių reklaminių banko paslaugų informacinių blokų. Kaip ir kiti bankai šis bankas neišsaugo mokėjimo duomenų išrašuose ir neatnaujina sąskaitos. Tiesa, turi gana plačią ir įvairiapusišką iš anksto įkeltų mokėjimų istoriją. Ši demonstracinė banko versija neturi mokomosios medžiagos, tačiau pagalba suteikiama per aktyvius nuorodas į paslaugų aprašymų puslapius.

The screenshot displays the Postbank online banking interface. At the top, the Postbank logo and navigation menu are visible. The user's name, Petra Pffiffig, and a welcome message are shown. The main section displays the total balance (Gesamtsaldo) of 4.020,00 €. Below this, there are tabs for 'Alle Produkte', 'Girokonten', 'Geldanlagen', and 'Kredite'. A promotional banner for 'Lahmes Sparschwein' is present. The main content area shows a list of accounts for Petra Pffiffig with a total balance of 8.070,00 €. The accounts listed are:

Account Name	IBAN	Balance	Actions
Giro start direkt	DE79 1001 0010 1234 5678 91	950,00 €	⋮
Unterhaltsame Feiertage mit unseren Vorteilsangeboten Zur Vorteilswelt			
Quartal SparCard	DE18 2011 0022 1234 5678 95	1.000,00 €	⋮
Tschüss Soli: Jetzt in einen ETF-Sparplan investieren Zu den ETF-Sparplänen			
Aktiv Sparbuch	DE18 2011 0022 1987 6543 22	1.000,00 €	⋮
Tschüss Soli: Jetzt in einen ETF-Sparplan investieren Zu den ETF-Sparplänen			
Tagesgeld	DE82 1001 0010 1234 5678 93	2.200,00 €	⋮
Unterhaltsame Feiertage mit unseren Vorteilsangeboten Zur Vorteilswelt			

Pav. 7. Postbank banko demonstracinė versija.

3.3 Demonstracinių elektroninių bankų palyginimas

Pastebėta tendencija, kad demonstracinės versijos yra naudojamos kaip interaktyvios iliustracijos, be mokėjimų duomenų išsaugojimo ir automatinių išrašų formavimo. Besimokančiam naujam banko naudotojui sunku suprasti kas įvyksta su jo pinigais kai įvykdomas mokėjimas. Nors kai kurie bankai turi gerai paruoštą pagalbinę medžiagą, tačiau neužtikrina pasitikėjimo ir verčia pagalbos ieškoti pas kitus žmones ar kreiptis į banką. Lyginant Lietuvoje veikiančius bankus atsiskleidė dėmesio trūkumas vyresnio amžiaus žmonėms, jiems nėra išskirta mokomoji medžiaga, neadaptuota pagal jų poreikius. Užsienio bankuose pastebimas didelis dėmesys senjorams, tačiau demonstracinėse versijose dažniausiai nebūna pagalbos funkcijų. Lietuvos bankų naudotojams užsienyje sukurtos finansinio raštingumo ir demonstracinių bankų sistemos dažniausiai netinka dėl kalbos barjero. Taip pat gali klaidinti užsienio bankai, kurie nepriklauso SEPA sistemai. Išsamesnis demonstracinių elektroninių bankų palyginimas atsiskleidžia 1 lentelėje.

1 lentelė. Lietuvos elektroninių bankų demonstracinės versijos palyginimas

Funkcijos	Swedbank	SEB bankas	Luminor bankas	Šiaulių bankas	Ready, Set, Bank programa	Westpac bankas	UBS bankas	Postbank bankas
Mokomoji medžiaga.	Vidutiniška	Išsami	Nėra	Nėra	Išsami	Nėra	Minimali	Vidutiniška
Aiškinantys vaizdo įrašai.	Yra	Nėra	Nėra	Nėra	Yra	Nėra	Nėra	Nėra
Paaiškinimai prie turinio objektų.	Nėra	Nėra	Yra	Nėra	Yra	Nėra	Yra	Nėra
Demonstracinė banko versija.	Nėra	Yra	Yra	Nėra	Yra	Yra	Yra	Yra
Prisijungimo imitavimas.	Nėra	Nėra	Yra	Nėra	Nėra	Yra	Yra	Yra
Aktualūs iš anksto įkelti duomenys.	Nėra	Iš dalies	Iš dalies	Nėra	Iš dalies	Taip	Taip	Taip
Pridėta daug perteklinės ir neaktyvios medžiagos.	Nėra	Taip	Taip	Nėra	Ne	Ne	Ne	Taip
Naudojami reklaminių skydelių.	Nėra	Ne	Ne	Nėra	Taip	Ne	Ne	Taip
Filtruojami išrašai.	Nėra	Yra	Yra	Nėra	Nėra	Yra	Yra	Yra
Galimybė atlikti mokėjimą.	Nėra	Yra	Yra	Nėra	Yra	Yra	Yra	Yra
Pervedimas tarp savo sąskaitų.	Nėra	Yra	Yra	Nėra	Yra	Yra	Yra	Yra
Sąskaitų apžvalga.	Nėra	Yra	Yra	Nėra	Iš dalies	Yra	Yra	Yra
Išplėstinės bankinės paslaugos (kreditai, kortelių užsakymas, lizingas ir t.t.).	Nėra	Yra	Yra	Nėra	Nėra	Nėra	Yra	Yra
Atitinka SEPA taisyklės.	Nėra	Taip	Taip	Nėra	Ne	Ne	Taip	Taip
Lengvai surandamas.	Ne	Taip	Taip	Ne	Taip	Taip	Taip	Taip
Kalba.	Lietuvių	Lietuvių	Lietuvių	Lietuvių	Anglų	Anglų	Anglų	Vokiečių

3. SISTEMOS KŪRIMAS.

Įvertinus poreikį ir išanalizavus jau esamas ir naudojamas technologijas išsikristalizavo reikalingi sistemos funkcionalumai ir reikalavimai programinei įrangai bei naudotojų sąsajai. Kadangi sistema orientuota į pradinio lygio mokymus, todėl nėra būtinybės išskirti sunkumo lygių ar skaidyti funkcionalumus pagal naudotojų skaitmeninio ar finansinio raštingumo lygį. Tai pat nėra poreikio daryti atskirus funkcionalumus mokymų organizatoriams ar lektoriams. Tai sąlygojo noras parodyti, kad visi naudojami tomis pačiomis banko funkcijomis ir nėra paslėptų paslaugų, o mokomoji medžiaga ir veikimo paaiškinimai pateikiami centralizuotai, mokymų organizatorių, o ne specifiskai atskirų lektorių. Sistema orientuota į vieną, banko naudotojo rolę, atsisakant administratorių ar redaktorių rolių. Atsižvelgiant į tai buvo išskirti tokie būtini funkcionalumai.

Funkciniai reikalavimai:

- naudotojas turi galėti perskaityti kokius veiksmus gali daryti sistemoje (kur spaudžiama, kaip atrodo nuorodos, į ką atkreipti dėmesį);
- rodomas sistemos funkcionalumus pristatantis filmukas;
- visuomet matomas „pagalbos mygtukas“, kurį paspaudus rastų informaciją apie sistemos veikimą tekstiniu ir/arba vaizdo būdu;
- naudotojo galimybė pasirinkti rodyti paaiškinimus ar ne;
- galimybė anuliuoti (ištrinti istoriją) visus bandytus veiksmus ir pradėti iš pradžių;
- iteracijų ir mokymosi metu padaryti pavedimai yra išsaugomi ir vėliau matomi (bent 2 savaitės);
- galimybė palikti atsiliepimus apie demonstracinio banko sistemą ir/arba užduoti klausimą;
- bent dviejų sąskaitų naudojimas;
- matomos jau sugeneruotos paskutinio mėnesio bankinės operacijos;
- automatinis sąskaitų balanso (išlaidų/pajamų) skaičiavimas atsižvelgiant į padarytus pavedimus;
- išrašų filtravimas pagal datą, operacijų tipus ir sąskaitas;
- naudojama reali operacijų data ir iš anksto nustatytų operacijų datų perskaičiavimas;
- mokėjimų formavimas pagal SEPA taisykles;
- automatinis mokėjimų klaidų tikrinimas pagal SEPA taisykles;
- mokėjimų formavimas ir pavedimas tarp sistemoje naudojamų sąskaitų;
- patvirtinimo kodo įvedimas (be generavimo);
- pranešimų apie įvykdytas operacijas rodymas.

Nefunkciniai reikalavimai:

- pritaikomumas kitoms šalims ir/arba kalboms;
- vartotojo sąsaja kaip įmanoma paprastesnė ir orientuota į mažo skaitmeninio raštingumo naudotojus;
- mygtukai ir nuorodos aiškios ir lengvai suprantamos;
- dizaino (stiliaus) sistemos naudojimas pagal pagrindines dizaino taisykles;
- sistema turi veikti visose pagrindinėse naršyklėse (pasiekti bent 96% populiariausių naršyklių aprėptį) ir kompiuterių ir mobiliųjų įrenginių operacinėse sistemose;
- sistemos serveriai neturi reikalauti specifinės ir sunkiai (brangiai) perkamos programinės įrangos;
- visos bankinės sistemos funkcijos turi veikti be interneto ryšio.

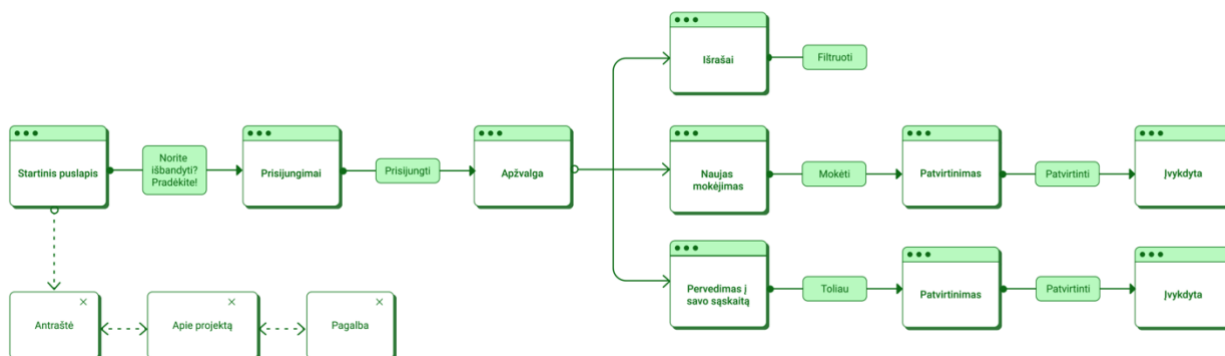
3.1 Prototipavimas

Prototipavimas - tai numatytų funkcionalumų vizualizavimas, tam kad lengviau būtų galima nustatyti klaidas ir palengvintų programavimą ir architektūros sudarymą. Prototipavimas reikalingas netik lengvesniam ir sklandesniam sistemos kūrimo organizavimui, bet ir lengvesniam sistemos funkcionalumų derinimui su mokymų lektorais ir mokymų organizatoriais. Teorinis problemų aptarimas ir galimų sprendimų paieška be abejo yra naudinga ir būtina, tačiau tik išvydus konkrečius, vizualius pavyzdžius mokymų organizatoriai ir lektoriai gali geriau įsivaizduoti kaip bus galim pritaikyti demonstracinį banką savo mokymuose, bei išgryninti norus ir pageidavimus. Pats patyriminis mokymosi procesas reikalauja galimybės išbandyti ar pamatyti veikiančius produktus, todėl svarbu užtikrinti kuo glaudesnę bendradarbiavimą ir kuo ankstyvesnę produkto kūrimo stadijoje. Kadangi kuriama sistema naudojama ne tik individualių žmonių bet ir grupiniuose mokymuose, svarbu sužinot lektorių patirtis ir įžvalgas. Lektoriai betarpiškai dirbantys su besimokančiais ir žinantys jiems kylančias problemas, geriausiai žino su kokiais sunkumais jie susiduria ir taip padeda tobulinti pačią sistemą. Taip pat prototipavimas leidžia sutaupyti laiką ir išlaidas išvengiant klaidų ar sistemos logikos ir funkcionalumų pakeitimų.

Prototipavimas gali būti organizuojamas įvairiais būdais: naudojamos grafinės kūrybos programos, specializuotos prototipavimo programos, piešiama ir aprašoma ranka, klijuojami aprašantys lipnūs lapeliai ir pan. Technologija pasirenkama atsižvelgiant į sistemos kūrimo etapą, kūrėjo gebėjimus ir naudotojų patogumą. Demonstracinio banko sistemos ankstyvasis prototipavimo etapas buvo vykdomas raštu (funkcionalumų derinimas) ir pašant ranka (pirminiai dizaino ir turinio architektūros eskizai), nes etape buvo konsultuojamasi su nedideliu žmonių kiekiu (1-3 žmonės). Vėliau, įtraukiant mokymų organizatorius iškilo poreikis kokybiškiems prototipams, todėl buvo

pasitelkta „Figma“ prototipavimo programa. Ji pasižymi patogiu prototipų kūrimu ir galimybe visiems komandos nariams matyti pakeitimus realiu laiku.

Vadovaujantis išnagrinėtais poreikiais ir mokymų organizatorių atsiliepimais, buvo sudarytas toks naudotojų mokymosi kelias ir turinio diagrama (žr. 8 pav.)



Pav. 8. Sistemos turinio diagrama.

3.2 Sistemos vartotojo sąsajos kūrimas

Pagrindinis sistemos dizaino tikslas yra efektyvus mokymasis tiek grupėje, tiek individualiai. Skirtingai nuo įprastų interneto svetainių ar programų, šioje nėra pagrindinis tikslas išsiskirti ir įtvirtinti prekės ženklą. Nekuriamas demonstracinio banko įvaizdis, o atvirkščiai, fokusuojamasi į funkcionalumus, siekiama, kad bankas būtų „beveidis“. Sistemos planavimo ir prototipavimo metu buvo išgryninti tokie pagrindiniai dizaino ir stiliaus aspektai:

- Dizaino vieningumas su kitais projekte vystoma ar naudojama grafine medžiaga ir stiliumi. Kadangi demonstracinio banko sistema yra vykdomų testinių mokymų projekto dalis, todėl turi būti išlaikytas, kiek įmanoma, vieningas stiliaus ar bent stiliaus elementai su kitomis projekto svetainėmis ar marketinginėmis priemonėmis. Šiuo atveju buvo pasirinktas naudoti tie patys tekstų šriftai, spalvos, mygtukų dizainas.
- Dizainas turi būti kuo paprastesnis, be išskirtinių, dėmesį atitraukiančių vizualizacijų, paveikslukų ar animacijų. Reikėjo maksimaliai išgryninti bankines funkcijas, be papildomų pagražinimų.
- Kadangi pati demonstracinio banko paskirtis yra parodyti kaip atrodo bankas, todėl ir jo išvaizda turi būti kuo panašesnė į realius elektroninius bankus. Kita vertus labai svarbu buvo išvengti panašumų su konkrečiu banku. Vienas iš svarbiausių mokymų aspektų yra išmokyti besimokantįjį susiorientuoti ir atlikti bazines operacijas bet kokiame Lietuvoje veikiančiame elektroniniame banke. Todėl kuriant dizainą ir informacijos architektūrą buvo įvertinti visi

Lietuvos elektroniniai bankai ir pagal juos sudaryta navigacija ir formų veikimo principai. Reikėjo naudoti realiuose bankuose dažniausiai naudojamus grafinius valdiklius, tokius kaip išsiskleidžiančius pasirinkimus (angl. dropdown select), iššokančius modalinius langus (angl. modal window), kalendoriaus datų pasirinkimus (angl. datetimepicker). Jie skirti tam, kad besimokantysis galėtų kuo geriau perprasti realiuose bankuose naudojamas navigacijos ir informacijos pateikimo priemones.

Kuriant demonstracinio banko sistemą vadovautasi pagrindinėmis dizaino kūrimo taisyklėmis ir dėsniais, kurie paremti žmogaus psichologijos eksperimentais:

- Jakob'o taisyklė. Siekiama, kad dizainas būtų atpažįstamas ir suprantamas. Neieškoma labai įmantrių dizaino sprendimų.
- Hick'o taisyklė (arba Hick-Hyman'o taisyklė). Siekiama sumažinti sprendimų skaičių, kuriuos turi priimti sistemos naudotojas ją naudodamas.
- Miller'io taisyklė. Kurti kuo paprastesnes, neapkrautas sistemas, kad būtų lengviau orientuotis.
- Gestalt'o principai. Siekiant intuityvaus vartotojos sąsajos naudojimo, elementai turi būti grupuojami, kartotis, atskirti tuščia erdve.
- Von Restorff'o taisyklė. Pagal šią taisyklę teigiama, kad žmonės geriausiai atsimena tuos objektus, kurie pasikartojančių objektų sekoje išsiskiria.

3.3 Naudojami dizaino elementai

Spalvos

Teorinius spalvų psichologijos nagrinėjimo pamatus padėjo dar J. N. Gėtė knygoje „Spalvų teorija“ (1810). Šiais laikais neabejoja spalvų svarba. Pagal Birren (2006), spalvos turi svarbią emocinę reikšmę priimant sprendimus, išskiriant jas pagal spalvų temperatūrą, intensyvumą, atspalvį.

Demonstracinio banko sistemos stiliaus spalvinė gama yra šviesi, skirtas sudaryti švarumo, lengvumo ir paprastumo įspūdį. Baltame fone lengviau išlaikyti kontrastą, ypač svarbų regėjimo sutrikimus turintiems žmonėms (žr. 9 pav.).

Pagrindinė spalva yra žalia, pasirinkta pagal projekto „Prisijungusi Lietuva logotipą. Tiek pačio projekto „Prisijungusi Lietuva“ (kurio mokymuose dažniausiai naudojama demonstracinio banko sistema) svetainėje, tiek įvairioje marketingo ar viešinimo leidiniuose dažniausiai akcentuojama žalia spalva. Tačiau žalios spalvos atspalvis nėra identiškas projekto logotipui, nes netinka atspalvio ryškumas ir intensyvumas. Šioje sistemoje naudojamas atspalvis yra šiek tiek žemiškesnis, artimesnis Lietuvos gamtos žalumui, kartu ramesnis ir ne erziantis. Toks pasirinkimas

padarytas atsižvelgiant į ramybės, patrauklumo ir aiškumo svarbą mokymosi procese. Žalia spalva apibūdinama kaip harmonijos ir sveikatos spalva, balansuojanti emocijas, siejama su gamta, augimu. Manoma, kad žalia spalva simbolizuoja saugumą, viltį ir sėkmę.

Rudos spalvos atspalviai pagrindinėje iliustracijoje simbolizuoja stabilumą ir patikimumą. Ji simbolizuoja žemiškumą, paprastumą, suteikia stabilumo efektą. Neintensyvi ruda spalva leidžia išlaikyti stiliaus lengvumo įspūdį.

Pilkos spalvos atspalviai naudojami grafiniu elementų formavime ir elementų išskyrimo. Pilka spalva išskiriami objektai, kurie nėra svarbūs, tačiau reikia parodyti, kad tokie yra.

Tekstui parinkta įprasta juoda spalva dėl maksimalaus kontrasto išlaikymo. Naudojamų spalvų kontrastai atitinka WCAG 2.0 reikalavimus.



Pav. 9. Sistemos spalvos ir kodai.

Logotipas ir pagrindinė iliustracija.

Atsižvelgiant į demonstracinio banko dizaino reikalavimus buvo pasirinktas kuo paprastesnis ir neišsiskiriantis logotipas. Kaip ir realiuose bankuose yra numatyta vieta puslapio viršuje paveikslėliui, tačiau pasirinktas ne reklaminis, ne žinutę perduodantis, o modernus, neutralus, tačiau kartu dinaminis linijų paveikslukas. Harmoningas ir kartu vingiuotas linijų raštas simbolizuoja kelią ar topografinius kalnelius, kurie iliustruoja mokymosi kelią. Pagrindinėje iliustracijoje tikslingai atsisakyta, realiuose bankuose dažnai naudojamu, žmonių nuotraukų, vengiant dėmesio nustelbimo ir stengimosi išlaikyti neutralumą. Iliustracijos spalvos pasirinktos derinant jas su pagrindine žalia spalva, kartu neišsišokant ir neblaškant dėmesio (žr. 10 pav.).



10 pav. Sistemos logotipas ir pagrindinė iliustracija.

Piktogramos

Piktogramų naudojimas yra patogus įrankis paspartinti nuorodų ir informacijos suvokimą, bei išlaikyti stiliaus vieningumą. Sistemoje naudojama „FontAwesome“ piktogramų biblioteka.

Šriftai

Vienas iš pagrindinių, bet kokios dizaino sistemos, dalių yra teksto atvaizdavimas. Antraštėms pasirinktas „Bariol“ šriftas, taip išlaikant ryšį su bendru projekto stiliumi. Suapvalintos šrifto formos suteikia dizainui švelnumo ir ramumo. Likusiuose tekstuose naudojami sisteminiai neserifiniai šriftai, priklausomai nuo kompiuterio operacinės sistemos tai būtų „Segoe UI“, „Roboto“, „Helvetica Neue“ arba „Arial“ šriftai. Šie šriftai yra kurti specialiai skaitmeninei erdvei, todėl jų įskaitomumas ir raidžių supratimas yra didelis, bei šriftai yra gerai pažįstami. Įskaitomumas yra vienas svarbiausių patogios interneto sistemos kriterijų, todėl visoje sistemoje išlaikomas minimalus 16 px dydžio šriftas. Be to korektiškai veikia tekstų didinimas (angl. zoom) naudojant interneto naršyklės priemones, todėl silpniau matantys žmonės gali naudotis naršyklės neįgaliųjų funkcijomis.

3.4 Programinės įrangos pasirinkimas

Tam, kad pasiektų užsibrėžtus tikslus demonstracinis elektroninis bankas ne tik turi atvaizduoti grafiškai, bet ir funkcionuoti. Tam naudojamos technologijos įvairios programinio kodo kūrimo technologijos. Vienas iš esminių užduoties dalių yra būtinybė sistemai veikti nepriklausomai nuo interneto ryšio ir nereikalauti specifinių serverių ar duomenų saugyklų. Įgyvendinant šiuos tikslus buvo pasirinktos tokios technologijos:

- HTML
- SASS, kaip CSS generavimo variklis;
- Gulp;
- Javascript;
- localStorage, kaip lokali (naršyklės) duomenų bazė.

HTML

HTML (hiperteksto žymėjimo kalba) yra visų internetinių sistemų pagrindas. Ši kalba aprašo internetinių puslapių ar programų dalis pagal nustatytas taisykles, tokiu būdu visos naršyklės vienodai interpretuoja aprašytus duomenis. Pavyzdžiui, HTML kalba nurodo, kuris tekstas yra antraštė, paragrafas, citata, vaizdo ar garso įrašas. Interneto naršyklės neatvaizduoja HTML kodo, tik pagal jį identifikuoja informaciją. HTML kalbos raidą prižiūri ir taisykles nustato W3C konsorciumas.

Šiuo metu naudojama 5 HTML versija, kuri pasižymi platesniu ir detalesniu informacijos atvaizdavimo galimybėmis. Pridėta daugiau dinaminių elementų (vaizdo ir garso grotuvai), turinio architektūros elementų (antraštės, poraštės, straipsnių, citatų ir įrašų išskyrimai) ir specifinių aplikacijų programavimo sąsajų (angl. API). HTML5 pasižymi tokiomis savybėmis:

- Semantika. Leidžia tiksliau apibūdinti turinį.
- Jungiamumas. Leidžia susijungti su serveriais naujausiomis technologijomis.
- Saugyklos. Leidžia saugoti duomenis kliento pusėje ir efektyviai veikti neprisijungus prie serverio.
- Multimedia. Įgalina atvaizduoti vaizdo ir garso įrašus be papildomų aplikacijų.
- 2D ir 3D palaikymas. Leidžia atvaizduoti specifinius 2D ir 3D efektus.
- Našumas ir integracija. Užtikrina didesnę spartos optimizavimą ir efektyvesnę kompiuterio resursų naudojimą.
- Įvesties ir išvesties įrenginiai. Leidžia prijungti įvairesnius įvesties ir išvesties prietaisus.
- Stilius. Suderintas su naujausiais CSS stiliais.
- Optimizuota paieška. HTML kalba yra viena iš draugiškiausių paieškos sistemoms, dėl savo universalumo ir populiarumo.
- Atviro kodo kalba.

Nors HTML yra universali ir standartizuota, tačiau kai kurios naršyklės gali nekorektiškai atvaizduoti turinį. Taip pat turi saugumo spragų jungiantis prie išorinių aplikacijų ar įrenginių. Be to yra labai jautri rašybos klaidoms. HTML atvaizdavimas priklauso nuo interneto naršyklių, todėl skirtingų gamintojų naršyklėse pasitaiko skirtingų HTML kalbos interpretavimų.

Demonstracinio banko sistemoje naudojama HTML5 versija, kartu su *Bootstrap 4* biblioteka, tam, kad išvengtų Versijų nesuderinamumo skirtingose interneto naršyklėse, greitesnio programavimo ir kodo interpretavimo.

CSS

Norint informaciją ne tik aprašyti, bet ir vizualizuoti, reikia naudoti CSS kalbą. CSS – pakopiniai stilių šablonai, skirti nusakyti duomenų išvaizdą. CSS naudojama kartu su HTML kalba aprašydama HTML žymas. Standartizuotų stilių naudojimas suteikia interneto svetainei ar aplikacijai tokius privalumus:

- Sutaupo laiko ją kuriant. Tuos pačius stilius galima naudoti daugeliui elementų, ir nereikia kas kartą jų aprašinėti.
- Dokumentai užkraunami greičiau. CSS kodą galima pateikti kaip atskirą bylą ar integruoti į HTML kodą. Taip pat išskaidyti jį dalimis. Tokiu būdu galima minimalų kodą siųsti užklauskos pradžioje, o likusį vėliau.
- Lengva priežiūra. Užtenka pakeisti mažą dalį CSS kodo, kad tai įtakotų visą aplikaciją. Be to vieningos sistemos naudojimas įgalina skirtingus programuotojus greitai suprasti ir taisyti kodą.

- Platus stilių pasirinkimas. Plačiausiai naudojama, CSS 3 versija palaiko turi virš 500 įvairių nuostatų, nuo elementų išdėstymo, iki šriftų keitimo ir paveikslukų manipuliacijų.
- Platus suderinamumas. Didžioji dauguma naršyklių, įvairiose operacinėse sistemose, supranta CSS kodą ir jį vienodai interpretuoja.
- Atviras kodas.
- Visuotinas standartas.

Nepaisant įtikinamų CSS privalumų, kodo rašymas išlieka imlus laikui. Be to CSS nebūdama pilna vertė programavimo kalba, neturi galimybės naudoti kintamųjų kode ar atlikti matematinius skaičiavimus. Tam buvo sukurti CSS preprocesoriai. CSS preprocesoriai (SASS, Less, Stylus) yra programavimo kalbos, kurios sugeneruoja CSS kodą pagal nustatytas taisykles. CSS preprocesoriai prideda tokį funkcionalumą:

- Kintamųjų naudojimas. Nustatyti kintamieji gali būti naudojami visose bylose, taip greitinant programavimą ir kodo tvarką, bei įsimenamumą.
- Išlaikoma CSS sintaksė. CSS preprocesoriuose galima rašyti CSS sintaksę ir kartu ją jungti su preprocesorių funkcijomis.
- Naudojama lizdinė sintaksė. Skirtingai nuo CSS, nereikia kiekvienam HTML objektui rašyti atskiro CSS kodo, juos galima grupuoti. Tokiu būdu sumažinamas kodo kiekis, jis rašomas natūraliau ir lengviau organizuoti kodą.
- Funkcijų (angl. mixin) naudojimas. Dažnu atveju kintamųjų neužtenka, reikia turėti galimybę įterpti kodą pagal teikiamus parametrus.
- Kodo moduliavimas ir importavimas. Galimybė išskaidyti kodą į atskiras bylas.
- Preprocesoriaus sukompiliuotas kodas atitinka visus CSS standartus.

Demonstracinio banko sistemoje pasirinktas SASS preprocesorius dėl savo universalumo, plačios dokumentacijos, greičio ir naudojimo Bootstrap bibliotekoje.

Javascript

Javascript programavimo kalba yra dažniausiai naudojama interneto technologijose. Sunku surasti tokią interneto svetainę ar aplikaciją, kurios funkcionalumo užtikrinimui nebūtų panaudota Javascript kalba. Javascript yra populiariausia programavimo kalba (Statista, 2020), išsiskirianti savo universalumu. Pagrindiniai Javascript kalbos privalumai yra šie:

- Mažiau iteracijų su serveriu. Javascript sukurta taip, kad veiktų ne tik serveryje, bet ir interneto naršyklėje, naudodama lankytojo kompiuterio resursus. Tokia savybė leidžia kurti ir pateikti dinaminį turinį realiu laiku, be būtinybės susisiekti su serveriu. Javascript suteikia galimybę atlikti skaičiavimus ar duomenų manipuliacijas naudotojo kompiuteryje, o siųsti į serverį jau sugeneruotus duomenis. Tai sumažina ryšio ir serverių apkrovimus.

- Pagreitina naudotojų veiksmų atsako laiką. Aplikacijų naudotojams nereikia laukti kol persikraus puslapis kol gaus atsakymą.
- Padidina interaktyvumą. Nors pastaraisiais metais sparčiai progresuoja HTML ir CSS galimybės, tačiau jos neprilygsta Javascript suderinamumu su įvairiais įvesties ir išvesties įrenginiais.
- Turtingesnė vartotojų sąsaja. Javascript pagalba galima įgyvendinti dinامينius elementus, suteikti galimybę atlikti specifinius vartotojų sąsajos veiksmus (pavyzdžiui, paišyti, tempti objektus, animuoti, manipuliuoti nuotraukomis ir t.t.).

Internetinėse svetainėse ir aplikacijose Javascript atsakinga už duomenų tikrinimą ir transportavimą, dinامينius grafikos elementus, kurių negali įgyvendinti CSS priemonėmis, bei naudotojų veiksmų iteracijas.

Kaip kiekviena programavimo kalba, Javascript galimybės yra plečiamos įvairių bibliotekų pagalba. Demonstraciniame elektroniniame banke naudojama populiariausia jQuery biblioteka. jQuery – tai greita, maža, veikianti įvairiose platformose ir pilna funkcionalumų biblioteka. Ji sukurta palengvinti kurti internetinių svetainių ar aplikacijų naudotojų sąsajas. Ji palengvina Javascript kalbos naudojimą automatizuodama dažniausias užduotis. Taip pat jQuery supaprastina duomenų perdavimą AJAX technologija. Pagrindiniai jQuery privalumai:

- Propaguoja sintaksės paprastumą. Sutrumpinami komandų rašymai leidžia greičiau parašyti kodą ir jį skaityti. Didina kodo panaudojamumą.
- Atviro kodo politika leidžia ne tik nemokamai dalintis programiniu kodu, bet ir jį populiarinti, tapdama viena populiariausių Javascript bibliotekų. Dėka populiarumo ir atvirumo įvairios interneto naršyklės sugeba teisingai interpretuoti svetaines ir aplikacijas, taip pateikiamą informaciją darydama prieinamesne ir populiariesne.
- Suderinamumas. Skirtingos interneto naršyklės skirtingai interpretuoja Javascript kalbą, kad išvengti ilgo testavimo, jQuery suvienodina Javascript interpretaciją. Tai reiškia, kad galima programuoti nesijaudinant dėl skirtingų interneto naršyklių gamintojų ar versijų.
- Kodo lengvumas ir mažumas. Palyginimui, standartinė jQuery biblioteka yra mažesnė už paprastą paveiksliuką interente. Be to kai kurie jQuery funkcionalumai yra iškelti kaip priedai ir gali būti prijungiami tik esant būtinybei.
- jQuery apjungia CSS ir HTML žymų galimybes, taip leisdamas sukurti įvairius efektus ir animacijas, kurios neįmanomos vien tik CSS priemonėmis.
- Automatizuotas duomenų siuntimas ir gavimas. jQuery supaprastina AJAX siunčiamas užklausas, jas automatizuodamas ir nereikalaudamas ilgo kodo rašymo.

Dėka plataus naudojimo, išvystytos dokumentacijos ir lengvo išmokymo jQuery biblioteką naudoja 77% interneto svetainių (W3Techs, 2020).

Bootstrap

Tam, kad dar labiau optimizuoti programavimo procesą demonstraciniame elektroniniame banke naudojama Bootstrap biblioteka. Tai atviro kodo vartotojo sąsajos kūrimui skirta biblioteka, kuri apjungia HTML5, CSS (per SASS) ir Javascript (per jQuery) funkcionalumus. Pagrindiniai Bootstrap privalumai yra:

- Laiko taupymas. Bootstrap sudarytas iš kodų rinkinių, kurie atsakingi už tam tikras funkcijas. Nereikia kas kart rašyti savo komandų, užtenka naudoti nustatytas taisykles.
- Paprastumas. Bootstrap neturi specifinės kalbos, užtenka mokėti įprastas HTML ir CSS žymas.
- Prisitaikantis dizainas. Ypač daug laiko praleidžiama taikant interneto svetainę ar aplikaciją skirtingiems ekranų dydžiams. Bootstrap supaprastina šį etapą ir leidžia greitai pritaikyti vartotojo sąsajas įvairiems ekranams, nuo mobiliųjų įrenginių, iki didelių ekranų.
- Suderinamumas. Kadangi Bootstrap naudojama jQuery biblioteka taip yra išsprendžiama suderinamumo klausimai.
- Lankstumas. Bootstrap sudarytas iš daugelio priedų ir funkcionalumų, kuriuos galima pasitelkti esant poreikiui.
- Vieninga vartotojo sąsaja. Kiekviena interneto naršyklė skirtingai atvaizduoja vartotojo sąsaja (skirtingi mygtukai, iššokančios lentelės ir t.t.), kurias Bootstrap suvienodina ir leidžia pradėti rašyti kodą nuo vieningo šablono.

Šiame demonstraciniame banke naudojama Bootstrap 4 versija.

Gulp

Anksčiau išvardinto technologijos yra susijusios, tačiau reikalauja daug pasikartojančių užduočių, kaip CSS kompiliavimas ar kodo optimizavimas ir tikrinimas. Tam, kad to nereikėtų daryti kaskart atnaujinus programinį kodą, naudojami automatizatoriai. Gulp yra automatinis užduočių vykdytojas. Kiekvieną kartą modifikavus programinį kodą, paleidžiama Gulp programa, kuri surenka kodą, atlieką jam nustatytus veiksmus, testuoja, sukopijuoja bylas, sukuria virtualų serverį ir viską atvaizduoja naršyklėje. Gulp gali atlikti įvairias užduotis. Gulp iš kitų panašių programų išsiskiria ekosistemos išvystymu (daugiau nei 3000 plėtinių), platformų itegracija (siejama su įvairomis programavimo kalbomis ir programomis) ir paprastumu. Šiame demonstraciniame banke buvo pasirinktas tokių užduočių automatizavimas:

- gulp-sass: sukompiluoja SAAS kodą į CSS.
- browser-sync: sukuria virtualų serverį.
- gulp-concat: sujungia vieno tipo bylas į vieną bylą.

- gulp-rev: prideda unikalų kodą prie bylos pavadinimo, kad naršyklės neatvaizduotų atmintyje likusio seno kodo.
- gulp-rev-rewrite: perrašo unikalius kodus.
- del: ištrina senas bylas.
- gulp-autoprefixer: automatiškai prideda specifinius priešdėlius pagal interneto naršykles.
- gulp-sourcemaps: sugeneruoja programinio kodo planus.
- gulp-uglify: sumažina bylas ištrindamas nenaudojamus tarpus, simbolius ir komentarus Javascript kode.
- gulp-clean-css: išvalo nereikalingus tarpus, simbolius ir komentarus CSS kode.
- gulp-imagemin: sumažina PNG, JPEG, GIF ir SVG bylas jas suspausdamas ir optimizuodamas.
- gulp-file-include: automatiškai įterpia nustatytą HTML kodą į kitus puslapius.
- gulp-inject: įterpia sugeneruotus Javascript ir CSS kodus į HTML kodą.

Pagal nustatytas komandas Gulp generuoja dalinai sukompiliuotą kodą testavimui ir pilnai sukompiliuotą viešinimui. Toks automatizavimas paspartina kodo rašymą ir leidžia koncentruotis į sistemos kūrimą.

Localstorage

Dauguma interneto svetainių ir aplikacijų naudoja duomenų bazes. Dažniausiai jos būna paremtos SQL kalba, tačiau šiuo atveju įprastos „MySQL“, „Oracle“ ar „Microsoft SQL“ netiko. Vienas iš pagrindinių demonstracinio banko sistemos reikalavimų buvo galimybė sistemai veikti be interneto ryšio ir papildomos programinės įrangos. Tokiu atveju serveriuose veikiančios duomenų bazės yra nepanaudojamos arba jas reikia diegti kiekviename kompiuteryje. Lieka galimybė duomenis saugoti pačioje interneto naršyklėje. HTML5 technologija ir modernios naršyklės suteikia galimybę saugoti įvairius duomenų formatus. Be to duomenų saugojimas lokaliai yra gerokai greitesnis nes nereikia duomenų siųsti į nutolusius serverius. Taip pat sumažėja pamestų duomenų tikimybė. Yra 3 įmanomų lokalaus duomenų saugojimo būdų pasirinktas *localStorage*. Šis būdas pasirinktas dėl:

- patogaus „raktas-reikšmė“ saugojimo eilutėje būdo.
- galiojimo laiko neturėjimo. Duomenys saugomi atmintyje tol kol yra ištrinami pačio naudotojo arba aplikacijos.
- veikia vieno šaltinio principu, t.y. localStorage saugomus duomenis gali pasiekti tik leidimus turinčios programos. Taip apsaugoma nuo duomenų nutekėjimo trečiosioms šalims.
- nereikalauja nuotolinio serverio.

Deja yra ir vienas neigiamas aspektas – tik 10 MB talpos atmintis, tačiau demonstracinio elektroninio banko sistemoje užtenka tokios atminties. Kitų alternatyvų buvo atsisakyta: *Session Storage* (dėl duomenų ištrynimo uždarius naršyklės langą), *Cookie* (dėl 4 KB atminties ir riboto galiojimo), *IndexedDB* (nors ir gali saugoti gerokai daugiau duomenų ir veikia asinchroniškai, tačiau nepalaikomas senesnėse naršyklėse).

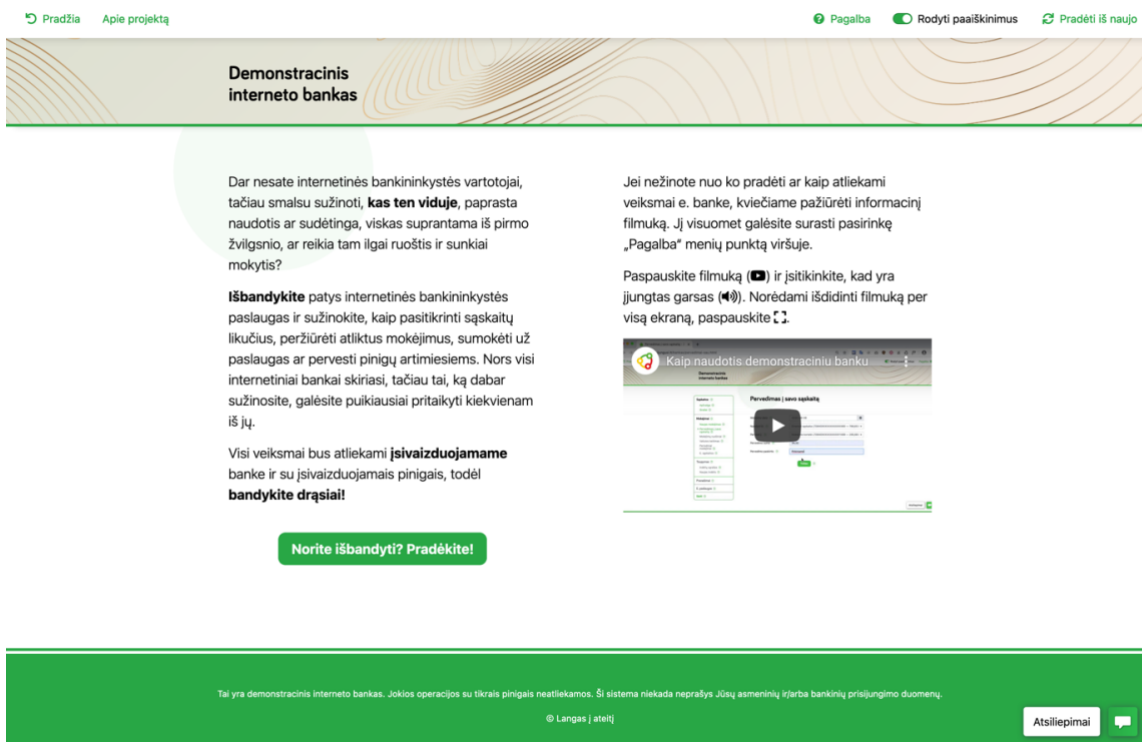
Su *localStorage* pavyko pasiekti, veiktų netik įprastiniame serveryje, bet ir paprastame kompiuterio standžiajame diske ar nešiojamame atmintuke. Tai padeda nenutraukti finansinio raštingumo mokymosi proceso dėl sutrikusio interneto ryšio ar vesti mokymus kur jo visai nėra.

3.5 Demonstracinio interneto banko programavimas

Planavimo ir prototipavimo stadijose buvo nuspręsta dėl demonstracinio elektroninio banko sistemos funkcionalumų ir nupaišytas dizainas, vėliau išsirinktos technologijos, kuriomis programuojama. Tolesniame etape visi sprendimais sujungiami į vieną veikiančią sistemą.

Pradinis puslapis

Pradinis puslapis – tai startinis aplikacijos puslapis, kurio pagrindinė paskirtis supažindinti su aplikacija. Startiniame puslapyje trumpai aprašoma, kodėl buvo kurta tokia sistema ir kokią naudą iš jos gaus naudotojas. Kadangi sistema orientuota ne tik į grupinių mokymų dalyvius, bet ir į pavienius naudotojus, todėl yra sukurtas ir įdėtas aiškinamasis vaizdo įrašas (žr. 11 pav.). Susipažinęs su aprašymu ir veikimu, naudotojas turi atlikti vienintelį veiksmą – paspausti mygtuką „Norite išbandyti? Pradėkite!“, tam kad galėtų pereiti prie banko funkcionalumų mokymosi.



11 pav. Sistemos pradinis puslapis.

Šis puslapis sukurtas naudojant *Bootstrap* biblioteką pritaikant numatytam dizainui. Vaizdo įrašas talpinamas *youtube.com* platformoje, o į puslapį įterpiamas `<iframe>` žymos pagalba. Tokiu būdu nereikia laikyti vaizdo įrašo serveryje ir rūpintis ryšio kokybe, be to įrašų talpinimas socialiniuose tinkluose prisideda prie sistemos populiarinimo.

Verta atskirai išskirti antraštės funkcionalumus, nes jie nekinta ir kartojasi visuose puslapiuose (žr. 12 pav.). Antraštė skirta pagalbiniai informacijai talpinti. Joje talpinama informacija turi būti visada matoma, tačiau nesimaišyti su bankinėmis funkcijomis. Antraštėje talpinama demonstracinio banko funkcionalumai, kurių nėra, arba jie kitaip pateikiami, realiuose elektroniniuose bankuose. Antraštėje be aprašymų yra svarbūs funkcionalumai, be kurių būtų sunku mokytis: sugrįžimo į pradžią, pagalbos, paaiškinimų rodymo pasirinkimo ir pradėjimo iš naujo (istorijos ištrynimo) funkcijos.



12 pav. Sistemos antraštė.

Pagalbos skiltyje pateikiamas vaizdo įrašas kaip naudotis demonstraciniu elektroniniu banku (tas pats įrašas kaip ir pradiniam puslapyje). Jį galima žiūrėti esant bet kuriame sistemos puslapyje

neprarandant duomenų ir neperkraudant puslapio, tai svarbu, norint patikrinti ir/arba sužinoti ką reikia daryti toliau ar ko tikėtis.

Sugrįžimo į pradžią ir pradėjimo iš naujo funkcijos yra gana panašios, tačiau pirmosios atveju naudotojas tik nukreipiamas į pradžią, kur gali vėl jungtis prie demonstracinio banko ir bandyti jo funkcijas. Visi veiksmai, kuriuos jis atliko anksčiau bus matomi ir išlikę. Tuo tarpu pasirinkęs „pradėti iš naujo“ funkciją naudotojas netik gražinamas į pradžią, bet ir ištrinama visa jo naudojimosi istorija. Naudotojas jungiasi kaip naujas naudotojas ir vėl mato tik iš anksto nustatytas reikšmes. Kad per klaidą neištrintų savo istorijos, tam padarytas iššokantis langas, vieta, kur būtų galima perskaityti kas laukia, bei reikalautų daugiau paspaudimų ir suteiktų laiko supratimui. Taip sumažinama tikimybė „sugadinti“ sistemą, kurios neretai bijo besimokantieji.

Paaškinimų rodymas yra labai svarbus mokymosi procese. Paaškinamai matomi iššokančiuose langeliuose, kai užvedama pelė arba paspaudžiamas pasirinktas klaustuko simbolis. Mokymų lektoriai, naudodami šį demonstracinį banką, rekomenduoja pirmą kartą bandyti su įjungtais paaškinimais, o kai išbandomos visos funkcijos, juos išjungti. Anot jų, vertinga pabandyti ir kitą dieną, individualiai, be paaškinimų ir vaizdo įrašų, išbandyti visas galimas bankininkystės funkcijas.

Antraštės iššokantys langai suprogramuoti pagal *Bootstrap* rekomendacijas, *modal* biblioteką. Iššokančių langų funkcionalumas pasirinktas dėl galimybės neišeiti į kitą puslapį, kad išvengtų pasiklydimo. Paaškinimų ir istorijos ištrynimui papildyti *Javascript* kodu (žr. 12 pav.).

```
1 //tooltips
2 $(function () {
3   $('[data-toggle="tooltip"]').tooltip()
4 });
5
6 // show tooltips
7 if(document.querySelector('body').classList.contains('index-page') === true ||
8 document.querySelector('body').classList.contains('prisijungimai-page') === true) {
9   $('.top-nav .nav-item .form-check').hide();
10 };
11 $('#tooltips').on('change', function() {
12   if ($('#tooltips').is(":checked")) {
13     $('.pastaba').css('display', 'inline');
14   } else {
15     $('.pastaba').css('display', 'none');
16   }
17 });
18 // reset history
19 $("#reset").click(function(){
20   localStorage.removeItem('tempSauStorage');
21   localStorage.removeItem('accCard');
22   localStorage.removeItem('accHistory')
23 });
```

12 pav. Paaškinimų ir istorijos ištrynimo kodas.

Kitas elektroninio demonstracinio banko funkcionalumas matomas nuolatos yra galimybė parašyti atsiliepimus. Iššokančiame lange užtenka įrašyti savo vardą, elektroninio pašto adresą ir atsiliepimą. Funkcionalumas realizuotas naudojant *purechat.com* sistemą.

Prisijungimo puslapis

Prisijungimo prie elektroninė bankininkystės mokymas yra vienas iš sunkiausių etapų. Nors pradinėje sistemos planavimo stadijoje buvo planuojamas elektroninio prisijungimo imitavimas, bet jo buvo atsisakyta dėl per sudėtingo paaiškinimo ir per didelio informacijos kiekio pradedančiajam elektroninės bankininkystės naudotojui, be to SMS pranešimų siuntimas reikalautų finansinių išteklių. Mokymo procese mokoma, kad elektroninės bankininkystės naudotojas tikrame banke gaus prisijungimo duomenis, o demonstracinėje sistemoje užtenka papasakoti kokie jie yra ir kaip jais naudotis. Nesvarbu kuri būdą pasirinktų, paspaudus „prisijungti“ mygtuką naudotojas nukreipiamas į demonstracinį banką.

Apžvalgos puslapis

Apžvalgos puslapio paskirtis yra supažindinti su pagrindinėmis bankinėmis funkcijomis ir apžvelgti ko reikia ieškoti/tikėtis elektroniniuose bankuose. Šis puslapis susideda iš trijų pagrindinių dalių: navigacijos juostos, sąskaitų apžvalgos ir paskutinių operacijų sąrašo (žr. 13 pav.).

Pradžią Aple projektą Pagalba Rodyti paaiškinimus Pradėti iš naujo

Demonstracinis interneto bankas

Sąskaitos

- Apžvalga
- Išrašai

Mokėjimai

- Naujas mokėjimas
- Pervedimas į savo sąskaitą
- Mokėjimų ruošiniai
- Valiutos keitimas
- Periodiniai mokėjimai
- E. sąskaitos

Taupymas

- Indėlių sąrašas
- Naujas indėlis

Pranešimai

E. paslaugos

Išėiti

Apžvalga

Sąskaitos	Valiuta	Kredito likutis	Rezervuota suma	Nuosavos lėšos	Likutis
Mokėjimo kortelė LT0840XXXXXXXXXX89	EUR	0,00	31,46	265,68	265,68
Einamoji sąskaita LT0840XXXXXXXXXX80	EUR	0,00	0,00	766,65	766,65
Suma					1.032,33

Paskutinės operacijos

Data	Mokėtojas/Gavėjas	Paskirtis	Suma
2020-12-05	UAB „Kita parduotuvė“	Už prekes	-31,46
2020-12-03	AB Bankas	Terminuotas indėlis	-150,00
2020-12-02	UAB „Žalias varnas“	Darbo užmokestis	+450,00
2020-12-01	UAB „Miesto turgus“	Už pirkinius	-23,71
2020-11-29	Jonas Petraitis	Skolos gražinimas	+150,00
2020-11-28	UAB „Maisto parduotuvė“	Operac. Nr. 24012019031	-56,39
2020-11-26	ATM Nr. 215	Grynujų įmokėjimas bankomate	+300,00
2020-11-24	UAB „Geras knygynas“	Už knygas	-21,33
2020-11-21	Valstybinė mokesčių inspekcija prie LR FM	Žemės mokestis už metus	-25,90
2020-11-20	Bankomatas CASH,VILNIUS,LTU	ATM234 CASH,VILNIUS,LTU	-40,00

Tai yra demonstracinis interneto bankas. Jokios operacijos su tikrais pinigais neatliekamos. Ši sistema niekada nepratys Jūsų asmeninių ir/arba bankinių prisijungimo duomenų.

© Langas | ateiti

Atsiliepimai

13 pav. Sistemos apžvalgos puslapis.

Navigacijos juostoje yra nuorodos į pagrindines bankines funkcijas, bei iliustruojamos kitos galimos funkcijos. Jų realizacijos atsisakyta dėl poreikio nebūvimo pas mokymų organizatorius ir koncentravimosi į bazines bankų funkcijas.

Sąskaitų apžvalgos dalyje rodomos dvi fiktyvios sąskaitos su iš anksto nustatytu pinigų kiekiu. Prie sąskaitų rodomas valiutos tipas, likęs kredito likutis (jeigu sąskaita turi kreditą), rezervuota suma, nuosavo lėšos ir likutis (nuosavų lėšų ir kredito suma). Rezervuota suma – yra paskutinių atliktų mokėjimų suma, taip imituojamas pinigų užlaikymas realiuose bankuose, kol vyksta pinigų transakcija. Ji atsinaujina dinamiškai, priklausomai nuo atliktų bankinių veiksmų.

Dvi sąskaitos pasirinktos, kad būtų galima patogiau iliustruoti sąskaitų rūšis bankuose: mokėjimo kortelės sąskaita kaip kasdieninio naudojimosi sąskaita, o einamoji sąskaita, kaip taupomoji (kaupiamoji) sąskaita (žr. 14 pav.). Mokėjimo kortelei yra numatyta kredito funkcija. Dėl

to, kad dažniausiai mokymuose dalyvauja senjorai, kredito likutis sumažintas iki nulio. Taip pat gana dažnai senjorai vengia turėti arba jiems neišduodamos kreditinės kortelės. Šis funkcionalumas geriau yra matomas programiniame kode.

```
1 // saskaitos
2 const accCard = [
3   {
4     update_time: new Date(),
5     expire_time: current_date.setHours(current_date.getHours()+168)
6   },
7   {
8     tipas: 'Mokėjimo kortelė',
9     saskaita: 'LT0840XXXXXXXXXX89',
10    valiuta: 'EUR',
11    kredito_likutis: 0.00, // kreditas 200.00
12    rezervuota_suma: 31.46,
13    nuosavos_lesos: 265.68
14  },
15  {
16    tipas: 'Einamoji sąskaita',
17    saskaita: 'LT0840XXXXXXXXXX00',
18    valiuta: 'EUR',
19    kredito_likutis: 0.00,
20    rezervuota_suma: 0.00,
21    nuosavos_lesos: 766.65
22  },
23 ];
```

14 pav. Numatytas sąskaitų masyvas.

Kiekviena kartą naudotojui prisijungus prie sistemos yra tikrinama ar jis jau turi sąskaitas ir ar jos nepasenusios. Tam yra generuojama galiojimo laiko data. Konsultuojantis su mokymų organizatoriais buvo nuspręsta, kad po savaitės tikslinga panaikinti duomenis ir leisti, net šiek tiek versti, pradėti nuo pradžių. Funkcionalumas realizuotas išsaugant *localStorage* naudotojo prisijungimo datą ir sugeneruojant galiojimo pabaigos datą (168 valandos – 7 paros) (žr. 15 pav.).

```
26 if (localStorage.getItem('accCard') === null) {
27   localStorage.setItem('accCard', JSON.stringify(accCard));
28 } else {
29   let itemStr = JSON.parse(localStorage.getItem('accCard'))
30   let now = new Date()
31   let expire_date = itemStr[0].expire_time
32   if (now.getTime() > itemStr[0].expire_time) {
33     console.log('expire time pasibaigė')
34     localStorage.setItem('accCard', JSON.stringify(accCard));
35   }
36 }
37 const data = JSON.parse(localStorage.getItem('accCard'));
38 // apzvalga
39 function displayApzvalga(){
40   let nuosavos = data.map(a => a.nuosavos_lesos);
41   let kreditas = data.map(a => a.kredito_likutis);
42   let apzvalgaTb = document.getElementById('apzvalga')
43   document.getElementById('suma').innerHTML = currencyFormatLT(nuosavos[1] + kreditas[1] + nuosavos[2] + kreditas[2])
44   apzvalgaTb.rows[1].cells[5].innerHTML = currencyFormatLT(nuosavos[1] + kreditas[1])
45   apzvalgaTb.rows[2].cells[5].innerHTML = currencyFormatLT(nuosavos[2] + kreditas[2])
46   apzvalgaTb.rows[1].cells[0].innerHTML = data[1].tipas + '<br><span class="text-muted">' + data[1].saskaita + '</span><i
class="far fa-question-circle pastaba" data-toggle="tooltip" data-html="true" title="Banko kortelė (kitaip mokėjimo
kortelė) – Banko išduota mokėjimo priemonė, kuria galite sumokėti už prekes arba paslaugas bei išsiimti grynųjų
bankomate. "></i>'
47   apzvalgaTb.rows[2].cells[0].innerHTML = data[2].tipas + '<br><span class="text-muted">' + data[2].saskaita + '</span><i
class="far fa-question-circle pastaba" data-toggle="tooltip" data-html="true" title="Einamoji sąskaita (kitaip
atsiskaitomoji sąskaita) – Banko sąskaita, kurioje vartotojas laiko lėšas ir gali bet kada jomis pasinaudoti."></i>'
48   apzvalgaTb.rows[1].cells[1].innerHTML = data[1].valiuta
49   apzvalgaTb.rows[2].cells[1].innerHTML = data[2].valiuta
50   apzvalgaTb.rows[1].cells[2].innerHTML = currencyFormatLT(data[1].kredito_likutis)
51   apzvalgaTb.rows[2].cells[2].innerHTML = currencyFormatLT(data[2].kredito_likutis)
52   apzvalgaTb.rows[1].cells[3].innerHTML = currencyFormatLT(data[1].rezervuota_suma)
53   apzvalgaTb.rows[2].cells[3].innerHTML = currencyFormatLT(data[2].rezervuota_suma)
54   apzvalgaTb.rows[1].cells[4].innerHTML = currencyFormatLT(nuosavos[1])
55   apzvalgaTb.rows[2].cells[4].innerHTML = currencyFormatLT(nuosavos[2])
56 }
57 if(document.querySelector('body').classList.contains('apzvalga-page') === true) {
58   displayApzvalga()
59 }
```

15 pav. Sąskaitų duomenų įrašymas į *localStorage* ir *html* kodą.

Suformuotų sąskaitų duomenys atvaizduojami lentelėje ir apskaičiuojama jų suma. Suma skaičiuojama, kaip ir realiuose bankuose, kreditą įtraukiant į bendrą sumą. Taip pat skaičiavimuose naudojamus skaičius reikia konvertuoti pagal lietuvių kalboje naudojamus skaičių skyrybos ženklų taisykles, tam naudojama funkcija *currencyFormatLT()* (žr. 16 pav.).

```
1 // perasom float i LT skaicius
2 function currencyFormatLT(num) {
3   return (
4     num
5     .toFixed(2)
6     .replace('.', ',')
7     .replace(/(\d)?=(\d{3})+(?!\d)/g, '$1.')
8   )
9 }
```

16 pav. Skaičių konvertavimas.

Paskutinių operacijų skiltis skirta atvaizduoti paskutines 10 operacijų. Šioje skiltyje atvaizduojamos ir naudotojo atliktos operacijos. Jei naudotojas darė pavedimus tarp savo sąskaitų, tuomet atvaizduojamas ir įskaitymas ir nurašymas. Realiuose bankuose šis funkcionalumas realizuojamas skirtingai, todėl buvo pasirinktas paprasčiausias ir aiškiausias variantas (žr. 17 pav.).

```
1 function displayIsrasas(){
2   let dataHist = JSON.parse(localStorage.getItem('accHistory'));
3   const lookup = dataHist.reduce((a, e) => {
4     a[e.suma] = ++a[e.suma] || 0;
5     a[e.paskirtis] = ++a[e.paskirtis] || 0;
6     a[e.data] = ++a[e.data] || 0;
7     return a;
8   }, {});
9   var filtered = dataHist.filter(e => lookup[e.suma] && lookup[e.data] && lookup[e.paskirtis] && e.operacija === '+');
10  for(var i = 0; i < filtered.length; i++){
11    for( var y = 0; y < dataHist.length; y++){
12      if ( dataHist[y] === filtered[i] ) {
13        dataHist.splice(y, 1);
14        y--;
15      }
16    }
17  }
18  var filteredMinus = dataHist.filter(e => lookup[e.suma] && lookup[e.data] && lookup[e.paskirtis] && e.operacija === '-');
19  for(var i = 0; i < filteredMinus.length; i++){
20    for( var y = 0; y < dataHist.length; y++){
21      if ( dataHist[y] === filteredMinus[i] ) {
22        dataHist[y].operacija = '';
23      }
24    }
25  }
26  dataHist.sort(function compare(a, b) {
27    var dateA = new Date(a.data);
28    var dateB = new Date(b.data);
29    return dateA - dateB;
30  });
31  dataHist.reverse()
32  if (dataHist.length > 10) {
33    dataHist.length = 10;
34  }
35  tbl = document.getElementById('paskutines');
36  var tb = tbl.createTBody()
37  for(var i = 0; i < dataHist.length; i++){
38    var tr = tb.insertRow(); // sukuria tr
39    td = tr.insertCell()
40    td.appendChild(document.createTextNode(date_format(dataHist[i].data)));
41    td = tr.insertCell()
42    td.appendChild(document.createTextNode(dataHist[i].gavejas));
43    td = tr.insertCell()
44    td.appendChild(document.createTextNode(dataHist[i].paskirtis));
45    td = tr.insertCell()
46    td.appendChild(document.createTextNode(dataHist[i].operacija + currencyFormatLT(dataHist[i].suma)));
47  }
48 }
49 if(document.querySelector('body').classList.contains('apzvalga-page') === true) {
50   displayIsrasas()
51 }
```

17 pav. Paskutinių operacijų atvaizdavimo programinis kodas.

Naujai prisijungęs naudotojas jau mato fiktyvius pavedimus su nedaugiau kaip mėnesio senumo datomis. Jiems suformuoti naudojamas automatinis datų generavimas (žr. 18 pav.). Fiktyvūs pavedimai išdėstomi praėjusio mėnesio laikotarpyje, kad sudaryti aktualumo įspūdį.

```
1 function histdate(numb){
2   var currentMil = new Date().setHours(0, 0, 0, 0);
3   var dayMil = 86400000 * (numb-1)
4   var d = currentMil - dayMil
5   e = new Date(d)
6   return date_format(e.toISOString())
7 }
```

18 pav. Skaičių konvertavimas.

Bankinių operacijų istorija formuojama stengiantis ją padaryti keik įmanoma panašesnę į realius žmonių pavadinimus, tačiau neperkraunant per dideliu kiekiu informacijos. Stengiamasi atvaizduoti įvairius tipinius mokėjimus, pavyzdžiui: už pirkimą internetu, pinigų išsiėmimą, pirkimą įprastoje parduotuvėje, mokėjimai už transporto paslaugas, pinigų įmokėjimą bankomate, įvairių mokesčių mokėjimą, labdaros pavedimą ir indėlio padėjimą. Mokėjimų istorija padalina po abidvi sąskaitas, tam , kad naudotojas galėtų matyti skirtumus sąskaitų išrašuose. Visa mokėjimų istorija saugoma interneto naršyklės *localStorage* (atskiroje eilutėje nuo sąskaitų).

Išrašų puslapis

Vienas iš pagrindinių elektroninės bankininkystės sričių, kurių mokomi pradedantieji elektroninių bankų naudotojai, yra orientavimasis banko sąskaitų išrašuose. Gebėjimas matyti visą finansinių transakcijų istoriją ir mokėjimų paskirtis leidžia ne tik prisiminti kur išleisti pinigai, bet ir matyti visą finansinių operacijų „paveikslą“. Pasidaro lengviau planuoti finansus, pastebėti kam daugiausia išleidžiama, o kur galima būtų sutaupyti. Išrašų puslapyje, skirtingai nuo paskutinių operacijų funkcijos, atvaizduojamos visos finansinės transakcijos (žr. 19 pav.).

Sąskaitos ?

Apžvalga ?

► Išrašai ?

Mokėjimai ?

Naujas mokėjimas ?

Pervedimas į savo sąskaitą ?

Mokėjimų ruošiniai ?

Valutos keitimas ?

Periodiniai mokėjimai ?

E. sąskaitos ?

Taupymas ?

Indėlių sąrašas ?

Naujas indėlis ?

Pranešimai ?

E. paslaugos ?

Išeiti ?

Išrašai

Sąskaitos ?

Laikotarpis ? -

Operacijos tipas ? Visos operacijos ? Išlaidos ? Įplaukos ?

Filtruoti ?

Data ?	Mokėtojas/Gavėjas ?	Paskirtis ?	Suma ?
2020-11-08	UAB „Mano vaistinė“	Transakcijos nr.: 12124456	-8,43
2020-11-09	UAB „Kabelinė TV“	Mokėjimas už kabelinę televiziją	-8,90
2020-11-12	UAB „Kaimo krautuvėlė“	Už prekes Nr.: 234233512	-18,55
2020-11-16	UAB „Teleryšys“	Už mob. Ryšio paslaugas	-9,32
2020-11-16	AB „Ignitis“	Už dujas	-15,45
2020-11-16	AB „Komunalinis ūkis“	Už komunalines paslaugas	-37,50
2020-11-16	UAB „Ignitis“	Už elektrą	-16,50
2020-11-20	Bankomatas CASH,VILNIUS,LTU	ATM234 CASH,VILNIUS,LTU	-40,00
2020-11-21	Valstybinė mokesčių inspekcija prie LR FM	Žemės mokestis už metus	-25,90
2020-11-24	UAB „Geras knygynas“	Už knygas	-21,33
2020-11-26	ATM Nr. 215	Grynųjų įmokėjimas bankomate	+300,00

19 pav. Išrašų puslapis.

Esminis išrašo puslapio funkcionalumas yra galimybė filtruoti mokėjimų istoriją pagal sąskaitas, laikotarpį ir operacijų tipą. Šie pasirinkimai yra sudėti į HTML užklauso formą ir naudotojui atlikus pasirinkimus ir paspaudus mygtuką „Filtruoti“ duomenys išsiunčiami *GET* metodu. *GET* metodo privalumas prieš *POST* metodą yra pasirinkimų atvaizdavimas puslapio adreso eilutėje (angl. *url*). Mokėjimų atvaizdavimo funkcija paima (naudodama *querystring* funkciją) nustatymų duomenis ir pagal juos išfiltruoja *localStorage* saugomus duomenis ir juos atvaizduoja (žr. 20 pav.).

```

1 const filterSask = querystring('sasisrasai');
2 const filterStart = querystring('start');
3 const filterEnd = querystring('end');
4 const filterTipas = querystring('operacijos');
5 // operaciju simboli pavercia zodziu
6 function wordToSymbol(i) {
7   if (i == 'islaidos') {
8     return '-'
9   } else if (i == 'iplaukos') {
10    return '+'
11  } else {
12    return 'visos'
13  }
14 }
15 function FullIsrasaiDisplay() {
16   var data1 = JSON.parse(localStorage.getItem('accHistory'));
17   // surikiavimas pagal data
18   data1.sort(function compare(a, b) {
19     var dateA = new Date(a.data);
20     var dateB = new Date(b.data);
21     return dateA - dateB;
22   });
23   // Formuojama lentele
24   var fullIsrasaiTb = document.getElementById('full-israsai');
25   tbl = document.getElementById('full-table');
26   var tb = tbl.createTbody()
27   // filtruoti pagal operacijos tipa
28   if (wordToSymbol(filterTipas[0]) == 'visos' ) {
29     var data = data1;
30   } else {
31     // filtruojam pagal reikiama reikame is url
32     var data = data1.filter(i => i.operacija == wordToSymbol(filterTipas[0]) );
33   }
34   // Likuciu masyvas
35   var myPajamos = new Array;
36   var myIslaidos = new Array;
37   var likutis = new Array;
38
39   for(var i = 0; i < data.length; i++){
40     var saskStatus = false;
41     if (filterSask == data[i].saskaita || filterSask[0] === undefined || filterSask[0] == 'Visos' ) {
42       var saskStatus = true;
43     }
44     let transactionDate = new Date(date_format(data[i].data)).setHours(0, 0, 0, 0);
45     let startDate = new Date(filterStart).setHours(0, 0, 0, 0);
46     let endDate = new Date(filterEnd).setHours(0, 0, 0, 0);
47
48     if (transactionDate <= endDate && transactionDate >= startDate && saskStatus == true ){
49       var tr = tb.insertRow();
50       td = tr.insertCell(); td.appendChild(document.createTextNode(date_format(data[i].data)));
51       td = tr.insertCell()
52       td.appendChild(document.createTextNode(data[i].gavejas));
53       td = tr.insertCell()
54       td.appendChild(document.createTextNode(data[i].paskirtis));
55       td = tr.insertCell()
56       td.appendChild(document.createTextNode(data[i].operacija + currencyFormatLT(data[i].suma)));
57       //LIKUCIAI
58       if (data[i].operacija == '+') {
59         myPajamos.push(data[i].suma)
60       } else {
61         myIslaidos.push(-Math.abs(data[i].suma))
62       }
63     }
64     // jei nepasirinkte data ir atitinka saksaito, tai rodomos paskutinio menesio transakcijos
65     if (!isNaN(startDate) && saskStatus == true ) {
66       let date = new Date();
67       // month ago convert to milliseconds
68       let monthago2 = new Date(date.getFullYear(), date.getMonth()-1, date.getDate()).setHours(0, 0, 0, 0);
69       if (transactionDate > monthago2 ) {
70         var tr = tb.insertRow();
71         td = tr.insertCell(); td.appendChild(document.createTextNode(date_format(data[i].data)));
72         td = tr.insertCell()
73         td.appendChild(document.createTextNode(data[i].gavejas));
74         td = tr.insertCell()
75         td.appendChild(document.createTextNode(data[i].paskirtis));
76         td = tr.insertCell()
77         td.appendChild(document.createTextNode(data[i].operacija + currencyFormatLT(data[i].suma)));
78         //LIKUCIAI
79         if (data[i].operacija == '+') {
80           myPajamos.push(data[i].suma)
81         } else {
82           myIslaidos.push(-Math.abs(data[i].suma))
83         }
84       }
85     }
86     //Visas sumas sudadam i likuti
87     if (data[i].operacija == '+') {
88       likutis.push(data[i].suma)
89     } else if (data[i].operacija == '-') {
90       // paverciam i neigiamu reiksme
91       likutis.push(-Math.abs(data[i].suma))
92     }
93   }
94   fullIsrasaiTb.appendChild(tbl);
95   let sumPajamos = myPajamos.reduce((a, b) => a + b, 0)
96   let sumIslaidos = myIslaidos.reduce((a, b) => a + b, 0)
97   //viska susumuojam
98   let sumLikutis = likutis.reduce((a, b) => a + b, 0)
99   let sumLikutisPries = sumLikutis - sumPajamos - sumIslaidos
100  //Irasoma laikotrapio islaidos
101  document.getElementById('likutis-islaidos').innerHTML += currencyFormatLT(sumIslaidos);
102  //Irasoma laikotrapio pajamos
103  document.getElementById('likutis-pajamos').innerHTML += '+' + currencyFormatLT(sumPajamos);
104  //Likutis pries
105  document.getElementById('likutis-pries').innerHTML += currencyFormatLT( sumLikutisPries );
106  //Likutis po
107  document.getElementById('likutis-po').innerHTML += currencyFormatLT(sumLikutisPries+sumPajamos+sumIslaidos);
108  if (tbl.rows[1] == null) {
109    document.getElementById("Israsai-message").innerHTML = 'Pagal nurodytus kriterijus duomenų nėra';
110  }
111 }

```

20 pav. Išrašų generavimo ir atvaizdavimo kodas.

Laikotarpio pasirinkimas realizuotas naudojant *datepicker* biblioteką (žr. 21 pav.). Laikotarpis automatiškai pritaikomas pagal naudotojo prisijungimo dieną ir rodomas paskutinis mėnuo nuo tos dienos. Naudotojas gali nusistatyti bet kokią pageidautiną laikotarpio rėžį.

```
125 if(document.querySelector('body').classList.contains('israsai-page') === true) {
126     FullIsrasaiDisplay()
127     if (filterStart[0] == null || filterStart[0] == '') {
128         var date = new Date();
129         var today = new Date(date.getFullYear(), date.getMonth(), date.getDate());
130         var monthago = new Date(date.getFullYear(), date.getMonth() - 1, date.getDate());
131
132         $(document).ready(function () {
133             $('#datestart').datepicker({format: 'yyyy-mm-dd', language: "lt", });
134             $('#datestart').datepicker( 'setDate', monthago );
135             $('#dateend').datepicker({format: 'yyyy-mm-dd', language: "lt", });
136             $('#dateend').datepicker( 'setDate', '0' );
137         })
138     }
139 }
140 //datepicker range
141 $(document).ready(function () {
142     $('#input-daterange').datepicker({
143         todayBtn: "linked",
144         language: "lt",
145         todayHighlight: true,
146         setDate: "today",
147         format: 'yyyy-mm-dd'
148     }).on('changeDate', function(e) {
149
150     });
151     if ($('#input-daterange')[0]){
152         $('#datestart').datepicker('setDate', filterStart);
153         $('#dateend').datepicker('setDate', filterEnd);
154     }
155     if ($('#sasisrasai')[0]){
156         $('#sasisrasai').val(filterSask);
157     }
158     if (!$('#sasisrasai').val()){
159         $('#sasisrasai').val('Visos');
160     }
161     if (wordToSymbol(filterTipas[0]) == '-' ) {
162         $('#opislaidos').prop("checked", true);
163     } else if (wordToSymbol(filterTipas[0]) == '+' ) {
164         $('#opiplaukos').prop("checked", true);
165     } else {
166         $('#opvisos').prop("checked", true);
167     }
168 });
```

21 pav. Laikotarpio pasirinkimo kodas.

Papildomas naudingas funkcionalumas – ataskaitų lentelės rūšiavimas pagal eiliškumą. Lentelės antraštės, paspaudus ant rodyklių, galima keisti rezultatų išdėstymą didėjimo arba mažėjimo tvarka. Numatytas nustatymas yra rikiavimas pagal datą. Vartotojas gali pasirinkti rikiavimą pagal datą, mokėtojo ar gavėjo pavadinimą, paskirtį ir sumą (žr. 22 pav.)

```

171 // table sort
172 function sortTable(f,n){
173     var rows = $('#full-israsai table tbody tr').get();
174     rows.sort(function(a, b) {
175         var A = getVal(a);
176         var B = getVal(b);
177         if(A < B) {
178             return -1*f;
179         }
180         if(A > B) {
181             return 1*f;
182         }
183         return 0;
184     });
185
186     function getVal(elm){
187         var v = $(elm).children('td').eq(n).text().toUpperCase();
188         // currency apmazinimas ir vertimas.
189         if (v.startsWith("$") && v.slice(-3, -2) == ',' || v.startsWith("-") && v.slice(-3, -2) == ',') {
190             v = parseFloat(v.replace(',',''));
191         }
192         if ($.isNumeric(v)){
193             v = parseInt(v,10);
194         }
195         return v;
196     }
197     $.each(rows, function(index, row) {
198         $('#full-israsai table').children('tbody').append(row);
199     });
200 }
201 var cs = 1;
202 $('".column-sort").click(function(){
203     cs *= -1;
204     var n = $(this).prevAll().length;
205     sortTable(cs,n);
206 });

```

22 pav. Išrašų rezultatų lentelės rūšiavimas.

Tai pat pridėtas automatinis pinigų likučių prieš ir po nustatytą datą skaičiavimas, bei išlaidų ir pajamų skirstymas. Tai suteikia patogumo naudotojui, nes nereikia papildomai skaičiuoti. Jis yra integruotas į mokėjimo istorijos atvaizdavimo programinį kodą ir nėra išskirtas kaip atskira funkcija.

Naujo mokėjimo puslapis

Pagrindinė elektroninio banko paskirtis yra atlikti mokėjimus į kitas sąskaitas. Tai besimokančiajam yra pati jautriausia elektroninio banko dalis. Vienas iš pagrindinių stabdžių trukdančių naudotis elektronine bankininkyste yra nežinojimas kas vyksta su pinigais, kokie veiksmai gali lemti pinigų praradimą. Ši demonstracinio banko skiltis sukurta taip, kad būtų rodomi pagrindiniai ir būtini mokėjimo atlikimo laukeliai. Nuo 2016 metų Lietuvoje veikia SEPA mokėjimų taisyklės. SEPA (angl Single Euro Payment Area) – tai 52 Europos valstybes ir teritorijas apimanti erdvė, kurioje mokėjimai eurais atliekami vienodomis sąlygomis (Lietuvos Bankas, 2020). Toks vieningo standarto naudojimas suvienodino bankų reikalavimus mokėjimo duomenims ir leidžia naudoti lietuviškas raides. Išmokus formuluoti mokėjimus vienoje sistemoje, automatiškai tai bus galima daryti kitose Lietuvos ar kitų SEPA šalių narių bankuose. Šis demonstracinis elektroninis bankas taip pat paremtas SEPA taisyklėmis.

Vizualiai mokėjimo atlikimo sistema susideda iš trijų žingsnių: mokėjimo formos užpildymo, patvirtinimo kodo įvedimo ir patvirtinimo pranešimo. Mokėjimo forma susideda iš septynių punktų, iš kurių keturis būtina užpildyti sistemos naudotojui, o du gali būti užpildomi automatiškai (žr. 23 pav.).

Mokėjimas

Mokėjimo data ?	2020-12-06 📅
Iš sąskaitos ?	Mokėjimo kortelė LT0840XXXXXXXXXX89 — 854,68 EU +
Gavėjo sąskaita ?	LT45268569541236
Gavėjo vardas ?	Vardas Pavardė
Gavėjo kodas ?	48296541256
Mokėjimo suma ?	7.30
Mokėjimo paskirtis ?	<input checked="" type="radio"/> Laisvas tekstas <input type="radio"/> Įmokos kodas ?
	Už dienos pietus
Mokėti ?	

23 pav. Mokėjimo forma.

Mokėjimo formoje data priskiriama automatiškai, tačiau leidžiama datą pasirinkti ir iš kalendoriaus (*datepicker* pagalba). Prie sąskaitų pavadinimo ir numerio automatiškai pridedamas pinigų likutis (žr. 24 pav.)

```

1 // sumu rodymas prie sąskaitu
2 function sumapriesaskaitos(id) {
3     for (i = 0; i < id.length; i++) {
4         var x = id.options[i].text
5         let index = kurisObjektas(data, splitSaskaita(x).toString())
6         let lesos = data[index].nuosavos_lesos + data[index].kredito_likutis
7         let pilnas = x + ' - ' + currencyFormatLT(lesos) + ' EUR';
8         id.options[i].innerHTML = pilnas;
9     }
10 }

```

24 pav. Sąskaitos likučio rodymo kodas.

Gavėjo sąskaita, gavėjo vardas ir kodas palikti laisvais laukeliais į kuriuos galima vesti bet kokią informaciją. Gavėjo kodo įvedimas, net ne būtinas. Mokėjimo paskirtis taip pat pasirenkama laisvai, įmokos kodai nėra tikrinami. Būtina užpildyti gavėjo sąskaitos, gavėjo vardo, mokėjimo sumos ir mokėjimo paskirties laukelius. Šie laukeliai yra tikrinami. Jei naudotojas jų neužpildo, išmetama klaida ir neleidžiama tęsti. Verta išskirti mokėjimo sumos validavimą, nes čia tikrinama ne tik laukelio užpildymas reikiamais simboliais, bet ir paskaičiuoja ar pasirinkus nurodytą sumą užteks pinigų pasirinktoje sąskaitoje ir ar neviršys numatyto kredito. Tai pat neleidžiama įvesti kitokių simbolių nei skaičiai ir ne daugiau dviejų skaitmenų po taško. Laukelių validacija realizuota naudojant dalį *Bootstrap* bibliotekos ir papildomą kodą (žr. 25 pav.).


```

1 (function() {
2   'use strict';
3   window.addEventListener('load', function() {
4     var forms = document.getElementsByClassName('needs-validation');
5     var validation = Array.prototype.filter.call(forms, function(form) {
6       form.addEventListener('submit', function(event) {
7         if (form.checkValidity() === false) {
8           event.preventDefault();
9           event.stopPropagation();
10          }
11          form.classList.add('was-validated');
12        }, false);
13      });
14    });
15  });
16  function check(input) {
17    let saskaita = saskaitaInput.options[saskaitaInput.selectedIndex].value
18    let indeksas = data.findIndex(x => x.saskaita === saskaita);
19    let lesos = data[indeksas].nuosavos_lesos + data[indeksas].kredito_likutis
20    let suma = parseFloat(input.value.replace(',', '.'))
21
22    if (suma <= 0) {
23      input.setCustomValidity('Bloga suma');
24      document.getElementById("suma-error").innerHTML = "Pakeiskite blogai įrašytą sumą";
25    } else if (suma > lesos) {
26      input.setCustomValidity('Neužtenka lėšų. ');
27      document.getElementById("suma-error").innerHTML = "Neužtenka lėšų";
28    } else {
29      // input is fine — reset the error message
30      input.setCustomValidity('');
31    }
32  }
33  function setInputFilter(textbox, inputFilter) {
34    ["input", "keydown", "keyup", "mousedown", "mouseup", "select", "contextmenu", "drop"].forEach(function(event) {
35      textbox.addEventListener(event, function() {
36        if (inputFilter(this.value)) {
37          this.oldValue = this.value;
38          this.oldSelectionStart = this.selectionStart;
39          this.oldSelectionEnd = this.selectionEnd;
40        } else if (this.hasOwnProperty("oldValue")) {
41          this.value = this.oldValue;
42          this.setSelectionRange(this.oldSelectionStart, this.oldSelectionEnd);
43        } else {
44          this.value = "";
45        }
46      });
47    });
48  }

```

25 pav. Formų laukelių validacijos kodas.

Užpildžius formą naudotojas turi paspausti mygtuką „Mokėti“. Tuomet sistema sukuria naują laikiną *tempStorage* įrašą, kuriama saugoma visą į formą įvesta informacija (žr. 26 pav.). Atlikus šiuos veiksmus naudotojas nukreipiamas į patvirtinimo puslapį.

```

1 const moketi = document.getElementById("moketi")
2 const saskaitaInput = document.getElementById("saskaita")
3 const sumaInput = document.getElementById("mok-suma")
4 const gavSaskInput = document.getElementById("gav-saskaita")
5 const gavVardasInput = document.getElementById("gav-vardas")
6 const gavKodasInput = document.getElementById("gav-kodas")
7 const gavPaskirtisInput = document.getElementById("paskirtis-kodas")
8
9 function mokejimas() {
10   let dataNew = {
11     data: new Date,
12     saskaita: saskaitaInput.options[saskaitaInput.selectedIndex].value,
13     suma: parseFloat(sumaInput.value.replace(',', '.')),
14     gav_sask: gavSaskInput.value,
15     gavejas: gavVardasInput.value,
16     gav_kodas: gavKodasInput.value,
17     pask_tipas: document.querySelector('[name="paskirtisOptions"]:checked').value,
18     paskirtis: gavPaskirtisInput.value,
19     operacija: '-',
20   };
21   localStorage.setItem('tempStorage', JSON.stringify(dataNew));
22 }
23 // on moketi submit
24 if(document.querySelector('body').classList.contains('pervedimai-page') === true) {
25   sumapriesaskaitos(saskaitaInput);
26   var mokejimasForm = document.getElementById("mokejimas");
27   mokejimasForm.addEventListener("submit", mokejimas);
28   setInputFilter(document.getElementById("mok-suma"), function(value) {
29     return /^~?d*([,]?\d{0,2})$/i.test(value); });
30 }

```

26 pav. Formos duomenų įrašymas į laikiną saugyklą .

Patvirtinimo puslapyje sistemos naudotojas turi įvesti patvirtinamąjį kodą ir gali pasitikrinti ar teisingą informaciją parašė pildydamas formą. Laukelyje jau yra įrašytas pagalbinis kodas „XXXXXX“, tačiau naudotojui leidžiama įvesti bet kokį kodą (kodas netikrinamas, svarbu, kad būtų kažkas įvesta). Kadangi sistema saugo visus duomenis, naudotojas gali paspausti interneto naršyklės mygtuką „atgal“ ir matys užpildytą formą. Esant būtinybei galės atlikti pataisymus. Po mygtuko „Patvirtinimas“ paspaudimo atliekamos esminės mokėjimo funkcijos dalys (žr. 27 pav.):

- mokėjimo duomenys pridedami prie mokėjimų istorijos;
- perskaičiuojamos ir atnaujinamos sąskaitų sumos (jei reikia ir nuo kredito);
- prirašoma, jei reikia, susumuojama rezervuojama suma prie sąskaitų apžvalgos;
- visi pakeitimai išsaugomi *localStorage* duomenų bazėje;
- ištrinama laikina mokėjimo duomenų saugykla.

```

1 // Patvirtinimas
2 function confirmationDisplay() {
3   let array = JSON.parse(localStorage.getItem('tempStorage'));
4   let confirmationTb = document.getElementById("confirmation")
5   confirmationTb.rows[0].cells[1].innerHTML = getSaskName(array.saskaita)
6   confirmationTb.rows[1].cells[1].innerHTML = currencyFormatLT(array.suma) + " EUR"
7   confirmationTb.rows[2].cells[1].innerHTML = array.gav_sask
8   confirmationTb.rows[3].cells[1].innerHTML = array.gavejas
9   confirmationTb.rows[4].cells[1].innerHTML = array.gav_kodas
10  confirmationTb.rows[5].cells[1].innerHTML = array.paskirtis
11 }
12 function confirmation() {
13   let confirmInput = document.querySelector("#confirm");
14   if (confirmInput.checkValidity() === true) {
15     let arrayTemp = JSON.parse(localStorage.getItem('tempStorage'));
16     a = JSON.parse(localStorage.getItem('accHistory'));
17     a.push(arrayTemp);
18     localStorage.setItem('accHistory', JSON.stringify(a));
19     const acc = JSON.parse(localStorage.getItem('accCard'));
20     var saskaitaTemp = arrayTemp.saskaita
21     var sumaTemp = arrayTemp.suma
22     switch (splitSaskaita(saskaitaTemp)) {
23       case splitSaskaita(acc[1].saskaita) :
24         acc[1].rezervuota_suma = acc[1].rezervuota_suma + sumaTemp
25         if (acc[1].nuosavos_lesos >= sumaTemp) {
26           acc[1].nuosavos_lesos = acc[1].nuosavos_lesos - sumaTemp
27         } else {
28           let skirtumas = sumaTemp - acc[1].nuosavos_lesos
29           acc[1].nuosavos_lesos = acc[1].nuosavos_lesos - (sumaTemp - skirtumas)
30           acc[1].kredito_likutis = acc[1].kredito_likutis - skirtumas
31         }
32         localStorage.setItem('accCard', JSON.stringify(acc));
33         localStorage.removeItem('tempStorage');
34         break;
35       case splitSaskaita(acc[2].saskaita) :
36         acc[2].rezervuota_suma = acc[2].rezervuota_suma + sumaTemp
37         if (acc[2].nuosavos_lesos >= sumaTemp) {
38           acc[2].nuosavos_lesos = acc[2].nuosavos_lesos - sumaTemp
39         } else {
40           let skirtumas = sumaTemp - acc[2].nuosavos_lesos
41           acc[2].nuosavos_lesos = acc[2].nuosavos_lesos - (sumaTemp - skirtumas)
42           acc[2].kredito_likutis = acc[2].kredito_likutis - skirtumas
43         }
44         localStorage.setItem('accCard', JSON.stringify(acc));
45         localStorage.removeItem('tempStorage');
46         break;
47       default :
48         console.log('nera saskaitu')
49     }
50     localStorage.removeItem('tempStorage');
51   }
52 }
53 if(document.querySelector('body').classList.contains('patvirtinimas-page') === true) {
54   confirmationDisplay()
55   const patvirtinimasForm = document.getElementById("patvirtinimas");
56   patvirtinimasForm.addEventListener("submit", confirmation);
57 }

```

27 pav. Mokėjimo patvirtinimo programinis kodas .

Atlikus visus veiksmus sistema naudotojui parodo sėkmingos transakcijos pranešimą ir dar kartą parodo mokėjimo duomenis.

Pervedimas į savo sąskaitą puslapis

Turint daugiau negu vieną sąskaita, o ypač jei viena iš jų kreditinė, neišvengiamai reikia daryti pinigų pervedimus tarp savo sąskaitų. Šis funkcionalumas veikia panašiai kaip paprastas mokėjimas. Esminiai skirtumai nuo įprasto mokėjimo yra tai kad nereikia didžiosios dalies formos laukelių, užtenka tik pasirinkti sąskaitas iš kurios į kuria pervedama, įrašyti sumą ir paskirtį (žr. 28 pav.).

Pervedimas į savo sąskaitą

Mokėjimo data 2020-12-07

Nurašyti iš: Einamoji sąskaita LT0840XXXXXXXXXXXX80 – 112,65 EUR

Pervesti į: Mokėjimo kortelė LT0840XXXXXXXXXXXX89 – 192,84 EUR

Pervedimo suma 50

Pervedimo paskirtis Papildymas

Toliau

28 pav. Pervedimo į savo sąskaitą puslapis .

Šis funkcionalumas realizuotas panašiai kaip paprasto mokėjimo, naudotos tos pačios funkcijos (klaidų tikrinimo, pinigų sumų atvaizdavimo ir t.t.). Pervedant pinigus tarp savo sąskaitų nebereikia įvesti patvirtinimo kodo, užtenka paspausti patvirtinimo mygtuką.

Prie jau suprogramuotų funkcijų prisideda tikrinimas ar pervedami pinigai tarp skirtingų sąskaitų ir ar jose užtenka pinigų (žr. 29 pav.). Taip pat tikrinama ar ne pervedami kredito pinigai, nes realiuose bankuose pervedinėti kredito pinigų, iš vienos sąskaitos į kitą to pačio asmens sąskaitą, negalima.

```

93 function checkSau(input) {
94   let saskaita = isSauInput.options[isSauInput.selectedIndex].value
95   let indeksas = data.findIndex(x => x.saskaita === saskaita);
96   let lesos = data[indeksas].nuosavos_lesos
97   let suma = parseFloat(input.value.replace(',', '.'))
98   if (suma <= 0) {
99     input.setCustomValidity('Bloga suma');
100    document.getElementById("suma-sau-error").innerHTML = "Pakeiskite blogai įrašytą sumą";
101  } else if (suma > lesos) {
102    input.setCustomValidity('Neužtenka lėšų. ');
103    document.getElementById("suma-sau-error").innerHTML = "Neužtenka lėšų";
104  } else {
105    input.setCustomValidity('');
106  }
107  if (isSauInput.options[isSauInput.selectedIndex].text === isSauInput.options[isSauInput.selectedIndex].text) {
108    isSauInput.setCustomValidity('Pasirinkite kitą sąskaitą');
109    document.getElementById("sask-error").innerHTML = "Pasirinkite kitą sąskaitą";
110  } else {
111    isSauInput.setCustomValidity('');
112  }
113 }
114 if (document.querySelector('body').classList.contains('pervedimai-sau-page') === true) {
115   sumaSauInput.addEventListener('change', (event) => {
116     checkSau(sumaSauInput)
117   });
118   isSauInput.addEventListener('change', (event) => {
119     checkSau(sumaSauInput)
120   });
121   iSauInput.addEventListener('change', (event) => {
122     checkSau(sumaSauInput)
123   });
124   if (isSauInput.options[isSauInput.selectedIndex].text === isSauInput.options[isSauInput.selectedIndex].text) {
125     isSauInput.setCustomValidity('Pasirinkite kitą sąskaitą');
126     document.getElementById("sask-error").innerHTML = "Pasirinkite kitą sąskaitą";
127   } else {
128     isSauInput.setCustomValidity('');
129   }
130 }

```

29 pav. Pavedimo į savo sąskaitą puslapis.

Tai pat skiriasi pavedimo informacijos įrašymas į mokėjimų istoriją. Kadangi nuo vienos sąskaitos pinigai nuskaitomi, o į kitą įrašomi, mokėjimo istorijoje atvaizduojamos abidvi operacijos. Tai svarbu nagrinėjant sąskaitų išrašus, kad prie kiekvienos sąskaitos matytųsi pinigų sumų pokytis. Šio tipo mokėjimai taip pat sukuriama laikinas *localStorage* įrašas, atliekami skaičiavimai, pakeičiamos visos sąskaitų duomenų reikšmės, išsaugomi duomenys ir naudotojui pranešama apie sėkmingą operaciją (žr. 30 pav.).

```

4 const moketiSau = document.getElementById("moketi-sau")
5 const dataSauInput = document.getElementById("data-sau")
6 const isSauInput = document.getElementById("saskaitaissau")
7 const iSauInput = document.getElementById("saskaitaisau")
8 const sumaSauInput = document.getElementById("mok-suma-sau")
9 const paskSauInput = document.getElementById("paskirtis-kodas-sau")
10
11 function pervedimasSau() {
12   let dataNew = {
13     data: dataSauInput.value,
14     is: isSauInput.options[isSauInput.selectedIndex].value,
15     i: iSauInput.options[iSauInput.selectedIndex].value,
16     suma: parseFloat(sumaSauInput.value.replace(',', '.')),
17     paskirtis: paskSauInput.value,
18   };
19   localStorage.setItem('tempSauStorage', JSON.stringify(dataNew));
20 }
21 if (document.querySelector('body').classList.contains('pervedimai-sau-page') === true) {
22   sumapriesaskaitos(isSauInput);
23   sumapriesaskaitos(iSauInput);
24   var mokejimasForm = document.getElementById("mokejimassau");
25   mokejimasForm.addEventListener("submit", pervedimasSau);
26   setInputFilter(document.getElementById("mok-suma-sau"), function(value) {
27     return /^~?d*[.,]?d{0,2}$/.test(value); });
28 }

```

```

29 function confirmationSauDisplay() {
30   let array = JSON.parse(localStorage.getItem('tempSauStorage'));
31   let confirmationTb = document.getElementById("confirmation-sau")
32   confirmationTb.rows[0].cells[1].innerHTML = getSaskName(array.is)
33   confirmationTb.rows[1].cells[1].innerHTML = getSaskName(array.i)
34   confirmationTb.rows[2].cells[1].innerHTML = currencyFormatLT(array.suma) + " EUR"
35   confirmationTb.rows[3].cells[1].innerHTML = array.paskirtis
36 }
37 function confirmationSau() {
38   let arrayTempSau = JSON.parse(localStorage.getItem('tempSauStorage'));
39   let arrayCards = JSON.parse(localStorage.getItem('accCard'));
40   let indexIs = arrayCards.findIndex(x => x.saskaita === arrayTempSau.is);
41   let indexI = arrayCards.findIndex(x => x.saskaita === arrayTempSau.i);
42   arrayCards[indexIs].nuosavos_lesos = arrayCards[indexIs].nuosavos_lesos - arrayTempSau.suma
43   arrayCards[indexI].nuosavos_lesos = arrayCards[indexI].nuosavos_lesos + arrayTempSau.suma
44   localStorage.setItem('accCard', JSON.stringify(arrayCards));
45   let tempSauTemp = [{0: {
46     data: arrayTempSau.data,
47     saskaita: arrayTempSau.is, //is
48     suma: arrayTempSau.suma,
49     gav_sask: arrayTempSau.i, //i
50     gavejas: "Pervedimas į savo sąskaitą",
51     gav_kodas: "",
52     pask_tipas: "tekstas",
53     paskirtis: arrayTempSau.paskirtis,
54     operacija: "-"
55   }, 1: {
56     data: arrayTempSau.data,
57     saskaita: arrayTempSau.i, //is
58     suma: arrayTempSau.suma,
59     gav_sask: arrayTempSau.is, //i
60     gavejas: "Pervedimas į savo sąskaitą",
61     gav_kodas: "",
62     pask_tipas: "tekstas",
63     paskirtis: arrayTempSau.paskirtis,
64     operacija: "+"
65   }
66   });
67   c = JSON.parse(localStorage.getItem('accHistory'));
68   c.push(tempSauTemp[0]);
69   c.push(tempSauTemp[1]);
70   localStorage.setItem('accHistory', JSON.stringify(c));
71 }
72
73 if(document.querySelector('body').classList.contains('patvirtinimas-sau-page') === true) {
74   confirmationSauDisplay()
75   const patvirtinimasForm = document.getElementById("patvirtinimas-sau");
76   patvirtinimasForm.addEventListener("submit", confirmationSau);
77 }
78
79 function successSauDisplay() {
80   let successTb = document.getElementById("success-sau")
81   let array = JSON.parse(localStorage.getItem('tempSauStorage'));
82   successTb.rows[0].cells[1].innerHTML = getSaskName(array.is)
83   successTb.rows[1].cells[1].innerHTML = getSaskName(array.i)
84   successTb.rows[2].cells[1].innerHTML = currencyFormatLT(array.suma) + " EUR"
85   successTb.rows[3].cells[1].innerHTML = array.paskirtis
86 }
87
88 if(document.querySelector('body').classList.contains('success-sau-page') === true) {
89   successSauDisplay()
90 }

```

30 pav. Pervedimo į savo sąskaitą puslapis.

4. TESTAVIMAS

Sukurta sistema turi būti testuojama ir rastos klaidos ištaisytos. Testavimas planuojamas atsižvelgiant į numanomus demonstracinio elektroninio banko naudotojus, į jų turimas priemones. Buvo pasirinkta testuoti naujausiose *Google Chrome*, *Firefox*, *Safari* ir *Edge* interneto naršyklėse, taip pat kompiuteriuose su *Windows 10*, *MacOs X* ir planšetėse su *Android* ir *iOs* naujausiomis operacinėmis sistemomis. Pasirinktos naujausios versijos, dėl dažno automatinio programų atnaujinimo naudojimo.

Testavimas sudarytas iš keturių testavimo fazių: funkcinio sistemos testavimo, vartotojo sąsajos testavimo, suderinamumo testavimo ir programinio kodo validumo testavimas. Gauti rezultatai aprašyti lentelėje (žr. 2 lentelę):

2 lentelė. Demonstracinio elektroninio banko testavimas.

Etapas	Testavimas	Rezultatai
Funkcionalumas		
	Navigacija tarp puslapių.	Veikia kaip numatyta
	Vaizdo įrašų rodymas.	Veikia kaip numatyta
	Apžvalgos atvaizdavimas.	Veikia kaip numatyta
	Numatytų duomenų išsaugojimas ir atvaizdavimas.	Veikia kaip numatyta
	Išrašų datos, operacijos tipo ir sąskaitos pasirinkimas.	Veikia kaip numatyta
	Lentelės stulpelių rikiavimas.	Veikia kaip numatyta
	Išrašų filtravimas pagal kiekvieną parametą.	Veikia kaip numatyta
	Mokėjimo formos atvaizdavimas.	Veikia kaip numatyta
	Mokėjimo formos įmanomų formos reikšmių testavimas.	Veikia kaip numatyta
	Mokėjimo formos klaidų tikrinimas.	Veikia kaip numatyta
	Patvirtinimo formos atvaizdavimas.	Veikia kaip numatyta
	Mokėjimo rezultato atvaizdavimas ir išsaugojimas duomenų bazėje.	Veikia kaip numatyta
	Pervedimo į savo sąskaitą atvaizdavimas.	Veikia kaip numatyta
	Pervedimo į savo sąskaitą klaidų tikrinimas.	Veikia kaip numatyta
	Pervedimo į savo sąskaitą atvaizdavimas ir išsaugojimas duomenų bazėje.	Veikia kaip numatyta
	Pagalbos mygtukų veikimas.	Veikia kaip numatyta
	Pagalbos pranešimų įjungimas/išjungimas.	Veikia kaip numatyta
	Naudotojo duomenų ištrynimas.	Veikia kaip numatyta
	Modalinių, iššokančių langų veikimas.	Veikia kaip numatyta
Vartotojo sąsaja		
	Navigacijos aiškumas ir paprastumas.	Tinkama.
	Mygtukų ir nuorodų veikimas.	Tinkama.
	Šriftų atvaizdavimas.	Tinkama.

	Spalvos. Kontrasto tikrinimas.	Tinkama.
	Paveikslukų atvaizdavimas.	Tinkama.
	Formų aiškumas.	Tinkama.
	Tekstų išdėstymas, įskaitomumas. Rašybos klaidos.	Tinkama.
Suderinamumas		
	<i>Windows 10 : Google Chrome, Firefox, Edge.</i>	Esminių skirtumų nerasta.
	<i>MacOs X: Google Chrome, Firefox, Safari.</i>	Esminių skirtumų nerasta.
	<i>Android: Google Chrome.</i>	Esminių skirtumų nerasta.
	<i>iOs: Safari.</i>	Esminių skirtumų nerasta.
Kodo validumas		
	Visų puslapio programinio kodo validavimas naudojant https://validator.w3.org priemones.	Klaidų ir įspėjimų nerasta.

Testavimo metu nerasta esminių sistemos klaidų ir/arba vartotojų nepatogumų, todėl programa perduota naudojimui mokymuose ir paviešinta patalpinant serveryje.

5. PRITAIKOMUMAS

Demonstracinis elektroninis bankas pagal pirminį sumanymą turi imituoti lietuviškus elektrinius bankus. Tačiau esant plačiai emigracijai ir stipriai plečiantis elektroninėms mokėjimų sistemoms atsiranda būtinybė išmokti naudotis įvairesnėmis priemonėmis. Lygiai taip pat svarbus tampa įvairesnių priemonių mokymas. Atsižvelgiant į tai šiame elektriniame demonstraciniame banke numatytos lokalizavimo galimybės. Sistemos lokalizavimas susideda iš trijų pagrindinių dalių: tekstų vertimo, iš anksto numatytos informacijos aktualinimo ir bankinių mokėjimų sistemos atitikimo.

Tekstų vertimas yra viena iš lengviausiai įgyvendinamų užduočių, nes tekstų nėra daug ir jie nereikalauja specifinių vertėjų įgūdžių. Tačiau reikia atsižvelgti į tai, kad verčiami žodžiai tilptų į numatytą plotą (sistema adaptuota didesniai tekstų plotui ir esant būtinybei gali prisitaikyti), bei būtų vartojami atitinkamos šalies bankuose.

Kitas svarbus aspektas yra numatytos informacijos suaktualinimas. Išrašuose matomos operacijos turi būti atitinkančios įprastus tos šalies tikslinės grupės pavedimus, pavyzdžiui, mokymų metu mokinama kaip atskirti kuris pavedimas atitinka komunalinius mokesčius ar pirkinis parduotuvėje. Tokiu būdu mokoma stebėti išlaidas ir pinigų balansą. Banko naudotojas turėtų atpažinti įprastus mokėjimus ir atskirti netipinius. Demonstracinio banko sistemoje šie duomenys yra pateikiami JSON formatu programiniame kode, todėl yra labai patogiu juos pakeisti, o esant būtinybei galima iškelti į atskirą rinkmeną.

Jeigu šalis kuriai norima lokalizuoti programą priklauso SEPA šalims, tuomet nėra būtinybės keisti mokėjimų formą. Vis dėl to jeigu šalis turi specifinių reikalavimų ir apribojimų, tuomet sistemoje reiktų pakeisti apžvalgos, mokėjimų ir išrašų funkcionalumus. Kadangi kiekvienas funkcionalumas ir jų žingsniai išskirti į atskiras funkcijas, vykdyti pakeitimus gali bet kuris programuotojas. Demonstracinio elektrinio banko pritaikomumas kitoms šalims reikalauja daugiau vertėjų, lektorių ar finansininkų indėlio nei programuotojų.

6. ATSILIEPIMAI IR ĮVERTINIMAI.

Kuriant bet kokią taikomąją aplikaciją labai svarbus grįžtamasis ryšys ir įvertinimai. Tai leidžia netik įvertinti sistemos naudą, bet ir ją koreguoti, geriau pritaikyti naudotojams. Kuriant ir tobulinant demonstracinę elektroninę banką buvo renkami įvertinimai ir atsiliepimai iš trijų šaltinių: grįžtamojo ryšio iš mokymų lektorių, besimokančiųjų bendrinės apklausos ir *Google Analytics* lankomumo statistikos. Atsiliepimų ir statistikos rinkimo laikotarpis yra nuo 2020 vasario 14 iki 2020 gruodžio 1 dienos. Atsiliepimai prieš demonstracinio banko naudojimą rinkti nuo 2019 sausio 1 iki 2020 vasario 14.

Kūrimo pirminiuose etapuose buvo atliekama žodinė mokymų, kuriuose planuojamas naudoti elektroninis demonstracinis bankas, lektorių apklausa. Norėta išsiaiškinti kokios apimties ir kokio sudėtingumo elektronines paslaugas norima išmokinti ir kokios priemonės būtų patogiausios. Tokio ryšio palaikymas leido greičiau ir tikslingiau sukurti banką.

Jau sukūrus ir išplatinus demonstracinę banką buvo vykdoma mokymų dalyvių apklausa. Nors nebuvo galimybės modifikuoti klausimus ir juos adaptuoti detalesnei informacijai, galima matyti atsiliepimų pagerėjimą (žr. 3 lentelę). Klausimai turėjo nuo 1 iki 5 vertinimo skalę, kur 1 reiškia mažiausią įvertinimą, o 5 – didžiausią. Labai mažas mokymų dalyvių kiekis (tik 1-2%) vertino neigiamai. Verta pažymėti, kad pradėjus naudoti demonstracinę elektroninę banką už kelių mėnesių prasidėjo karantinas ir dalis mokymų buvo vykdomi nuotoliniu būdu. Nors išaugo tuo nepatenkintų skaičius (nuo 0 iki 6%), tai nesumenkino pasitenkinimo mokymais lygio.

3 lentelė. Skaitmeninio raštingumo temos „Sumaniau apsipirkite ir atsiskaitykite internetu“ temos dalyvių apklausos rezultatai.

Klausimai	Prieš demonstracinio banko naudojimą	Naudojant demonstracinę banką
Suteikė naujų skaitmeninių žinių ir įgūdžių.	90% (4.5 balo)	96% (4.8 balo)
Mokymų turinys Jums aktualus.	94% (4.7 balo)	96% (4.8 balo)
Mokymų turinys aiškus ir gerai suprantamas.	96% (4.8 balo)	96% (4.8 balo)
Šių mokymų žinias ir įgūdžius pritaikysiu kasdieninėje veikloje.	94% (4.7 balo)	96% (4.8 balo)
Praktinių užduočių kiekis.	86% (4.3 balo)	98% (4.9 balo)
Ar esate patenkintas dalyvavimu skaitmeninio raštingumo mokymuose?	Visiškai patenkintas: 75% Patenkintas: 24% Iš dalies: 1% Nepatenkintas: 0%	Visiškai patenkintas: 79% Patenkintas: 20% Iš dalies: 1% Nepatenkintas: 0%

Jau naudojant demonstracinę elektroninę banką apklausoje sudalyvavo 2781 respondentas, o prieš naudojimą – 2154 respondentai. Pusė respondentų savo gyvenama vietoje nurodė miestą, kita pusė – miestelius arba kaimus. Moterys sudarė apie 80% dalyvių, vyrai – 20%. Pagrindines respondentų amžiaus grupės buvo: 51-65 amžiaus 42% ir daugiau nei 65 metų amžiaus – 31%. Daugiau kaip pusė dalyvių turėjo aukštąjį išsilavinimą (54%), 26% turėjo vidurinį su profesine kvalifikacija. Taip pat dominavo nedirbantys asmenys (pensijoje, neįgalūs, namų šeimininkės ir pan.) – 43%, 29% - samdomi darbuotojai ir 17% - bedarbiai.

Paskutinis ir bene tiksliausias atsiliepimų rinkimo būdas yra *Google Analytics* platforma. Ji pasirinkta dėl galimybės lengvai rinkti duomenis be specifinių nuostatų ir reikalavimų, funkcionalumą gausa ir nemokama licencija. Pagrindinis šio statistikos rinkimo įrankio trūkumas yra tai, kad vykstant mokymams izoliuotoje vietoje (kur naudojamas uždaras interneto ryšys) visi kompiuteriai fiksuojami kaip vienas, todėl susidaro situacija kuomet įprasta, dešimties žmonių grupė fiksuojama kaip viena. Taip pat statistika nerenkama, jei demonstraciniu banku naudojasi be interneto ryšio. Kadangi nėra galimybės tokių atvejų išvengti arba atpažinti, juos priskiriama įprastai paklaidai arba ignoruojami. *Google Analytics* surinko tokią naudotojų statistiką:

- Viso apsilankė 2190 naudotojų (3031 seansai). 80% iš jų yra nauji, 20% - grįžtantys pasikartoti.
- Kas savaitę vidutiniškai apsilanko apie 48 naudotojus.
- Parodyta 19439 puslapiai. Per kiekvieną seansą atverčiami 6,41 puslapio, tai reiškia, kad besimokantieji aplanko visus puslapius ir išbando visas funkcijas. Iššokantys pagalbos ar informacijos langai neįsiskaito kaip nauji puslapiai.
- Vidutiniškai praleidžiama beveik 6 minutes.

Apibendrinant gautus atsiliepimus ir statistikos rezultatus galima daryti išvadą, kad ši mokymosi priemonė yra naudinga ir nuolatos naudojama. Jos sukūrimas buvo naudingas, pagerino ir suaktualino mokymąsi organizuojamuose grupiniuose mokymuose. Taip pat pastebėtas šio demonstracinio banko pakartotinis naudojimas, kaip žinių įgilinimo priemonė.

7. REZULTATAI IR IŠVADOS

7.1 Rezultatai

1. Apžvelgta patirtinio mokymosi nauda finansiniame ugdyme.
2. Palyginti Lietuvoje ir užsienyje veikiančios panašios sistemos.
3. Sukurta, ištestuota demonstracinio elektroninio banko sistema. Programa patalpinta adresu: <https://www.prisijungusi.lt/bankas>.
4. Apžvelgti sistemos naudojimo rezultatai ir pritaikomumas kitoms šalims.

7.2 Išvados

1. Analizuojant finansinio raštingumo ugdymo padėtį ir iššūkius išryškėjo poreikis praktiniam elektroninių paslaugų ir produktų išbandymui. Valstybės ir privataus sektoriaus siekis kurti praktines aplikacijas nėra labai aktyvus ir labiau orientuotas į žaidimus vaikams. Patyriminio mokymosi būdas patogus mokantis tiek grupėse, tiek individualiai. Išryškėjo būtinybė besimokančiajam be spaudimo ir streso pažinti nežinomas elektroninės aplinkas, išbandyti ir matyti pokyčius.

2. Lyginant Lietuvoje veikiančius bankus pastebėtas menkas dėmesys nepatyrusiems ar busimiems elektroninio banko naudotojams. Vienintelis SEB bankas turi laisvai prieinamą demonstracinę banko versiją ir mokymosi medžiagą, tuo tarpu dauguma bankų apsiriboja trumpa mokymosi medžiaga arba vaizdo įrašais. Nagrinėtuose užsienio bankuose jaučiamas didesnis dėmesys vyresniems naudotojams, tačiau pasigendama pilnai funkcionuojančių demonstracinių bankų, kur būtų galima bandyti funkcijas nerizikuojant realiais pinigais. Be abejonės esminis užsienyje naudojamų sistemų trūkumas yra užsienio kalbos barjeras.

3. Planuojant demonstracinio elektroninio banko sistemą pastebėtas konsultacijų su mokymų lektorais poreikis, nes jų išvalgos ir besimokančiųjų pažinimas yra būtinas planuojant sistemą. Taip pat svarbus naujausių technologijų panaudojimas, kuris leido paspartinti programavimą ir išvengti suderinamumo iššūkių.

4. Sistemos naudotojų atsiliepimai ir grįžtamasis ryšys iš mokymų lektorių nustebino poreikiu supaprastinti sistemą iki bazinių užduočių. Natūraliai norisi sistemą sukurti su kuo daugiau funkcijų, kuo labiau panašesnę į tikrus bankus, tačiau poreikiai parodė, kad tikslingiau yra gerai išmokyti pagrindinius elektroninės bankininkystės žingsnius, o ne paviršutiniškai supažindinti su visais.

8. LITERATŪROS SĄRAŠAS

1. Association of Experiential Education (2008). *Asociation of Experiential Patirtinio mokymosi filosofija ir teorija. Įvadas*. Via Experientia: Tarptautinė patirtinio mokymosi akademija.
2. Bandura, A. (1997). *Self-efficacy: The exercise of control*. W H Freeman/Times Books/ Henry Holt & Co.
3. Birren, F. (2006). *Color psychology and color therapy: A factual study of the influence of color on human life*. Whitefish, MT: Kessinger.
4. Brodie, A. (n.d.). The basics about color. Prieiga per internetą: <http://www.andrew.cmu.edu/user/abrodie/color/index.html>
5. Cole, Cassandra & Liebenberg, Andre. (2010). Factors Impacting Low Retirement Savings.
6. Dewey, J. (1998). *Experience and education*. West Lafayette, IN: Kappa Delta Pi.
7. EBPO (OECD) (2005). Recommendation on Principles and Good Practices for Financial Education and Awareness. Prieiga per internetą: <https://www.oecd.org/daf/fin/financial-education/35108560.pdf>
8. Eurostat (2020). Individuals who have basic or above basic overall digital skills. Prieiga per internetą: https://ec.europa.eu/eurostat/databrowser/view/tepsr_sp410/default/table?lang=en
9. Frijns B., Gilbert A., Touani-Rad A. (2013). Learning by doing: the role of financial experience in financial literacy. *Journal of Public Policy*, 123-154.
10. Godvadas, P., Jasienė, G., Malinauskas, A. (2014) *Patirtinio refleksyvaus mokymo taikymo vietos bendruomenėse metodika*. Vilnius.
11. Grigaitienė, R. (2005). Studentų aktyvumo skatinimas anglų kalbos komunikacinių pratybių metu, remiantis patirtinio mokymosi teorijos teiginiais. *Mokslo taikomieji tyrimai Lietuvos kolegijose*, 9–14.
12. Hathaway I and Khatiwada S (2008) *Do Financial Education Programs Work?* Working Paper 08-20. Cleveland, OH: Federal Reserve Bank.
13. Hilgert MA, Hogarth JM and Beverly SG (2003) Household financial management: The connection between knowledge and behavior. *Federal Reserve Bulletin* 89(7), 309–322.
14. Jackūnas Ž. (2008). *Patyrimas, patirtis ir supratimas*. Kultūros, filosofijos ir meno institutas *LOGOS* 56 20-28.
15. Jakubė A., Juozaitis A.M. (2012) *Bendrujų kompetencijų ugdymas aukštojoje mokykloje*. Vilnius.
16. Kolb, D. A. (1983) *Experiential Learning: Experience as the Source of Learning and Development*. Baltimore, Prentice Hall Inc.
17. Kvieskienė G. (2016) Finansinio raštingumo samprata ir paradoksai. *Social Partnership and Innovations*, 24–35.

18. Lietuvos Bankas (2020). *Pranešimas apie Lietuvos banko pagrindinio tikslo įgyvendinimą, finansų rinkos būklę ir funkcijų vykdymą*. Vilnius.
19. Lietuvos Bankas (2020). Bankomatai ir kortelių skaitytuvai. Prieiga per internetą: <https://www.lb.lt/lt/bankomatai-ir-korteliu-skaitytuvai-1>
20. Lietuvos Bankas (2020). Bendra mokėjimų eurais erdvė SEPA. Prieiga per internetą: <https://www.lb.lt/lt/bendra-mokejimu-eurais-erdve-sepa#ex-1-2>
21. Lietuvos Bankas. (2020) Grynujų pinigų operacijos klientų aptarnavimo vietose. Prieiga per internetą: <https://www.lb.lt/lt/grynuju-pinigu-operacijos-klientu-aptarnavimo-vietose-1>
22. Lietuvos Bankas (2020). Klientų aptarnavimo vietos. Prieiga per internetą: <https://www.lb.lt/lt/klientu-aptarnavimo-vietos-1>
23. Lietuvos Bankas (2020). Konsoliduoti pagrindiniai bankų veiklos rodikliai. Prieiga per internetą: <https://www.lb.lt/lt/konsoliduoti-pagrindiniai-banku-veiklos-rodikliai>
24. Lietuvos Bankas (2020). *Metų ataskaita 2019*. Vilnius.
25. Lietuvos Bankas (2020). Mokėjimo kortelės. Prieiga per internetą: <https://www.lb.lt/lt/mokejimo-korteles-1>
26. Navickas M., Gudaitis T., Krajnakova E. (2014) Finansinio raštingumo įtaka jaunimo asmeninių finansų valdymui. *Verslas: Teorija ir praktika*, 32–40.
27. Perry VG and Morris MD (2005) Who is in control? The role of self-perception, knowledge, and income in explaining consumer financial behavior. *Journal of Consumer Affairs* 39(2), 299–313.
28. Statista (2020). Most used programming languages among developers worldwide. Prieiga per internetą: <https://www.statista.com/statistics/793628/worldwide-developer-survey-most-used-languages/>
29. Strazdienė, Gražina, & Geležinienė, Renata. (2008). Mokymosi modelių taikymo prioritetai imitacinėje verslo įmonėje = Priorities in applying learning models at a simulation business enterprise. *Profesinis rengimas: tyrimai ir realijos*, 102–115.
30. W3Techs (2020). Usage statistics and market share of jQuery for websites. Prieiga per internetą: <https://w3techs.com/technologies/details/js-jquery/>

9. PRIEDAI

9.1 Demonstracinio elektroninio banko programinis kodas

Visas demonstracinio elektroninio banko programinis kodas pasiekiamas adresu:
<https://www.prisijungusi.lt/demo-bank.zip> .