

ŠIAULIŲ UNIVERSITETAS

Informatikos katedra

Arnas Kučinskas

**Vartotojo darbo su kompiuterio klaviatūra analizės
sistema**

Bakalauro baigiamasis darbas

Vadovė dr. Simona Ramanauskaitė

Šiauliai, 2018

ŠIAULIŲ UNIVERSITETAS

Informatikos katedra

TVIRTINU

Informatikos ir matematikos katedros
vedėja doc. dr. V. Garbaliauskienė
2018-05-03

Vartotojo darbo su kompiuterio klaviatūra analizės sistema

Informacinių technologijų bakalauro baigiamasis darbas

Vadovė

Informatikos katedros docentė.
2018 m. gegužės 31 d.

dr. S. Ramanauskaitė

Recenzentė

Informatikos katedros docentė
2018 m. birželio ____ d.

dr. A. Drukteinienė

Autorius

IT-14 gr. studentas
2018 m. gegužės 31 d.

A. Kučinskas

Šiauliai, 2018

TVIRTINU

Informatikos katedros vedėja

doc. dr. V. Garbaliauskienė

2017 m. _____ mėn. _____ d.

Bakalauro baigiamojo darbo užduotis

Studentui **ARNUI KUČINSKUI**

Darbo tema: **VARTOTOJO DARBO SU KOMPIUTERIO KLAVIATŪRA ANALIZĖS SISTEMA
(SYSTEM FOR ANALYSIS OF COMPUTER USER'S WORK WITH THE KEYBOARD)**

1. Sukurti sistemą, skirtą vartotojo darbo su kompiuterio klaviatūra fiksavimui ir analizei:
 - 1.1. Susipažinti egzistuojančiomis sistemomis, skirtomis vartotojų darbo su kompiuterio klaviatūra optimizavimui.
 - 1.2. Atsižvelgiant į egzistuojančius analogus, suprojektuoti sistemą, kuri kaip fone veikianti programa stebėtų vartotojo klaviatūros paspaudimus ir remiantis sukauptais duomenimis, pateiktų atitinkamas rekomendacijas ar bent naudojimo statistiką.
 - 1.3. Realizuoti siūlomos sistemos prototipą.
 - 1.4. Įvertinti sukurtos sistemos veikimo tikslumą bei apkrovą sistemai.
2. Aiškinamojo rašto turinys turi atitikti Informatikos ir matematikos katedros studentų bakalauro baigiamųjų darbų metodinius nurodymus.
3. Bakalauro baigiamasis darbas pateikiamas su įrišta darbo užduotimi, įdėtu vadovės atsiliepimu ir kompaktine plokštele.
4. Kompaktinėje plokštelėje įrašyti visus informacinės sistemos veikimui būtinus failus ir baigiamojo darbo aiškinamąjį raštą.
5. Gynimo metu pateikiama pristatymo pateiktis ir sukurta informacinė sistema.
6. Apginto katedroje paskutinioji baigiamojo darbo pristatymo diena – 2018 m. gegužės 31 d.

Baigiamojo darbo vadovė

dr. Simona Ramanauskaitė

2017 m. rugsėjo 29 d.

Studentas

A. Kučinskas

2017 m. rugsėjo 29 d.

DARBO SANTRAUKA

Tikslas	Sukurti programinę įrangą (toliau PĮ), kuri sektų vartotojo darbą su klaviatūra, analizuotų duomenis ir pateiktų aktualią informaciją vartotojui.
Užduoties pagrindimas	PĮ yra kuriama, kadangi rinkoje esančios programos neturi funkcijų, kurios fiksuotų vartotojo daromas klaidas, o tuo labiau jas analizuotų. PĮ skirta asmenims, norintiems tobulinti savo darbą su klaviatūra.
Kuriamo produkto prototipai	WhatPulse – viena iš populiariausių tokio tipo programų. Leidžia matyti paspaudimų karščio žemėlapi, kiek paspaudimų buvo atliktą skirtingose programose, yra taikomi filtrai pagal datas arba programas. RSIGuard – programa labiau orientuotą į darbo kokybiškumą ir pertraukas darbo su kompiuteriu metu. Turi tris karščio žemėlapius, matuoja darbo su klaviatūra ir pele laiką. Keyboard Optimizer (IBM research) – programa, skirta pagerinti darbą su klaviatūra keičiant OS nustatymus žmonėms, kurie turi sunkumų dirbant su klaviatūra. Tačiau veikia tik su Windows XP operacine sistema. Aplication usage statistics – seka pelės, klaviatūros ir programų statistiką. Tačiau pateikia tik labai „sausą“ statistiką. Keyboard usage – turi karščio žemėlapi ir seka gerai parašyto teksto ir klaidų santykį.
Technologinis sprendimas	C# Programavimo kalba, Microsoft .NET Framework, SQLite duomenų bazė, Win32 API klavišų paspaudimų sekimui.
Panaudota programinė įranga	Visual Studio 2017 – programinio kodo rašymui ir testavimui. DB browser for SQLite – duomenų bazės peržiūrai ir testavimui. MSI Afterburner – programinės įrangos našumo testavimui. Aida64 – programinės įrangos našumo testavimui. Microsoft Office 2013 – dokumentacijos rengimui. Draw.io – diagramų braižymui.. ArgoUML – diagramų braižymui.
Naudotos diagramos	Veiklos – iliustruoja kaip PĮ reaguoja į klavišų paspaudimus. Sekų – iliustruoja tipinį vartotojo veiksmų scenarijų, rašymą ir skaitymą iš duomenų bazės, statistikos atvaizdavimą. Būsenų – iliustruoja sekimo buferio būsenas. Klasių – iliustruoja sistemos statinį vaizdą tarp vidinių ir išorinių komponentų. Panaudos atvejų – iliustruoja vartotojo funkcijas. Duomenų srautų – iliustruoja sistemoje vykstančius duomenų srautus.
Testavimo apimtis	Našumo, duomenų vientisumo, funkcionalumo, duomenų atvaizdavimo, stabilumo testavimas.
Vartotojo dokumentacijos sudėtinės dalys	Vartotojo dokumentacija yra integruota sistemoje. Vartotojui pateikiama dokumentacija su programinės įrangos funkcijų aprašymais.

SUMMARY

System for Analysis of Computer User's Work with the Keyboard

Currently in a market there are a lot of programs which tracks users work with keyboard. But none of those programs puts a focus on mistakes user is making and analysis of those mistakes. This is the main reason why this system has been developed.

Paper analyses programs which tracks user's work with keyboard, looks at best tools available to track keyboard inputs. Describes prepared requirements and system architecture based on which the system has been developed.

System uses heat maps, lists with grammatical mistakes and line graphs to help user identify his most often occurring mistakes and inefficiencies when working with keyboard. Thorough testing, code and database optimizations assures that system works stable and efficiently.

TURINYS

IVADAS.....	7
1. KLAVIATŪRA IR JOS SEKIMUI NAUDOJAMŲ TECHNOLOGIJŲ APŽVALGA	8
1.1 Klaviatūrų kategorijų apžvalga	8
1.2 Klaviatūros išdėstymų apžvalga.....	8
1.3 Metodų, skirtų fiksuoti vartotojo klaviatūros paspaudimus apžvalga	9
1.4 Globalių klaviatūros žinučių kabliukų veikimas	10
1.5 Fone veikiančios programinės įrangos privalumai.....	10
1.6 Analogų palyginimas	11
2. REIKALAVIMŲ SPECIFIKACIJA.....	14
2.1 Sistemos paskirtis.....	14
2.2 Projekto dalyviai	14
2.3 Vartotojai	14
2.4 Įpareigojantys apribojimai	15
2.5 Veiklos kontekstas	15
2.6 Produkto veiklos sfera.....	16
2.7 Funkciniai reikalavimai.....	19
2.8 Nefunkciniai reikalavimai	21
2.9 Galimos sistemos kūrimo rizikos	23
3. SISTEMOS ARCHITEKTŪROS SPECIFIKACIJA	24
3.1 Architektūros specifikacijos paskirtis	24
3.2 Architektūros pateikimas	24
3.3 Architektūros tikslai ir apribojimai	24
3.4 Sistemos dinaminis vaizdas.....	24
3.5 Sistemos statinis vaizdas.....	31
3.6 Duomenų vaizdas	33
3.7 PĮ komponentai	35
3.8 Duomenų srautai PĮ.....	36
3.9 PĮ kokybė	36
4. TESTAVIMAS	37
4.1 Testavimo paskirtis	37
4.2 PĮ funkcijų testavimas	37
4.3 PĮ charakteristikų testavimas.....	45
4.4 Testavimo išvados.....	48
5. VARTOTOJO DOKUMENTACIJA.....	49
5.1 Vartotojo vadovo paskirtis	49
5.2 Sistemos funkcinis aprašymas.....	49
5.3 Sistemos vartotojo vadovas.....	52
5.4 Sistemos diegimo vadovas	52
IŠVADOS	54
LITERATŪRA.....	55
TERMINŲ IR SANTRUMPŲ ŽODYNĖLIS.....	56
PRIEDAI.....	57

IVADAS

Didžioji dalis visų žmonių, kuriems tenka naudotis kompiuterio klaviatūra, su ja dirba ne optimaliai, labai dažnai to net nepastebi ar nenori pastebėti. Į šią grupę patenka visų amžių ir pajėgumų atstovai, nuo vaikų iki senjorų ir nuo pirmą kartą matančių klaviatūra iki IT profesionalų. Todėl sprendimai, kurie tinka vieniems, ne visada gali būti pritaikyti kitiems. Taip pat kiekvienas žmogus individualiai yra pripratęs prie savo rašymo stiliaus, tai galbūt jam neleidžia tobulėti greičiau.

Šiuo metu rinkoje esančios programinės įrangos (ypač Windows operacinei sistemai) neturi funkcijų, kurios padėtų suprasti, kur ir kodėl yra daromos gramatikos klaidos ar su kokiais klavišais vartotojui dažniausiai kyla problemų ir kaip tai ištaisyti. Todėl jų atliekami pakeitimai ne visada būna naudingi. Taip pat dažnai programinė įranga neprisitaiko prie vartotojo, todėl pačiam vartotojui tenka prisitaikyti prie programinės įrangos.

Darbo tikslas – sukurti programinę įrangą, kuri sektų klaviatūros paspaudimų duomenis ir leistų šiuos duomenis apibendrinti ir patogiai atvaizduoti. Taip palengvintų vartotojo darbo su kompiuterine klaviatūra metu kylančių klaidų aptikimą.

Uždaviniai:

1. Išanalizuoti rinkoje esančius sprendimus;
2. Parengti sistemos reikalavimų ir architektūros specifikacijas;
3. Realizuoti suprojektuotą programinę įrangą;
4. Atlikti programinės įrangos testavimą;
5. Parengti vartotojo dokumentaciją

Sukurta programinė įranga galės būti naudojama kaip fone veikianti programa, kuri nereikalauja papildomo vartotojo darbo su kompiuterio klaviatūra, norint surinkti duomenis jų analizei – analizei bus naudojami vartotojo klavišų paspaudimai įprasto darbo su kompiuteriu metu. Tai turėtų palengvinti sistemos pritaikomumą ir analizuojamų duomenų tikslumą.

1. KLAVIATŪRA IR JOS SEKIMUI NAUDOJAMŲ TECHNOLOGIJŲ APŽVALGA

1.1 Klaviatūrų kategorijų apžvalga

Klaviatūra yra prietaisas skirtas perduoti informaciją kompiuteriui. Tai gali būti simbolis ar komanda. Jau 1970m. buvo pradėdamos gaminti klaviatūros, kurios primena ir šių dienų klaviatūras, o 1981m. IBM išleido „IBM Model F“ klaviatūra, iš kurios daugelis funkcinių klavišų išliko ir iki pat šių dienų („NumLock“ su visais skaičiais, „F1-F12“, „Alt“, ir net „Windows“ klavišas) [1].

Klaviatūros dėl savo universalumo paplito labai plačiai ir yra naudojamos daugelyje prietaisų kaip pagrindinis įvesties metodas.

Klaviatūras galima išskirti į dvi kategorijas:

- Virtualios klaviatūros – naudojamos tokiuose prietaisuose, kaip išmanieji telefonai, planšetės, daiktų interneto prietaisai, ar prietaisuose, kurie turi fizinę klaviatūrą, kad jais galėtų naudoti vartotojai turintys negalią.
- Fizinės klaviatūros – jos dar yra grupuojamos į dvi grupes
 - Integruotos klaviatūros – dažniausiai sutinkamos nešiojamuosiuose kompiuteriuose, terminaluose ir specializuotuose prietaisuose.
 - Belaidės ir laidinės klaviatūros – pagrindinės šių klaviatūrų privalumas yra mobilumas. Jų poziciją galima laisvai keisti, ją apriboja tik laido ilgis arba maksimali pasirinktos technologijos (dažniausiai „bluetooth“ arba „Wi-Fi“) distancija. Tokio tipo klaviatūros dažnai yra universalios, su jomis galima valdyti nešiojamus ir personalinius kompiuterius, išmaniuosius telefonus ir planšetes ir daiktų interneto prietaisus.

Abu tipai turi savų trūkumų bei privalumų. Virtualias klaviatūras reikia programuoti kiekvienam prietaisui atskirai, tačiau jos turi didesnę lankstumą iš funkcijų pusės. Fizinės klaviatūros dažnai yra labiau ribotos, nes turi prisitaikyti prie esamų standartų. Nors ir su jomis yra daroma daugiau eksperimentų, pvz.: ergonomiškos, skirtos valdyti vieną ranką, specialių išdėstymų klaviatūros ir pan.. Vienas iš didžiausių fizinių klaviatūrų privalumų yra rašymo greitis, daugelyje atvejų žmonės gali rašyti greičiau su fizine klaviatūra, nei kad su virtualia [2].

1.2 Klaviatūros išdėstymų apžvalga

- QWERTY – išlieka populiariausias šalyse, kur yra naudojami lotyniškų simbolių abėcėlė. Šis išdėstymas buvo sugalvotas vieno iš pirmų asmenų, išradusio rašymo mašinėlę jungtinėse Amerikos valstijose – Cristopher Latham Sholes. Vėliau šis išdėstymas buvo parduotas „Remington“ spausdinimo mašinų kompanijai, kuri siekė jį pakoreguoti. Pakoreguotas išdėstymas didžiąją dalimi išliko toks pat iki šių dienų, tik su nedideliais patobulinimais [3].
- DOVARK – Šis išdėstymas buvo sukurtas August Dovark. Jis tikėjo, kad QWERTY išdėstymas yra neefektyvus ir nepatogus. DOVARK išdėstymas skirtas optimizuoti rašymą su klaviatūra, kad rankos kuo mažiau vaikščiėtų po klaviatūrą [4]. Tačiau pačio

August Dovark atliktos studijos buvo šališkos ir su daugybę kintamųjų. Daugelyje nepriklausomų studijų nebuvo rasta užtikrinamų įrodymų, kad DOVARK išdėstymas yra lengviau suprantamas, efektyvesnis ar patogesnis už QWERTY [5].

- Maltron – Šis išdėstymas buvo sukurtas „Maltron Ltd.“ kompanijos. Ji gamina ergonomiškas klaviatūras, kurių paskirtis yra sumažinti RSI (angl. repetitive strain injury) tikimybę. Specialiai šioms klaviatūroms buvo sukurtas „Maltron“ išdėstymas. Kompanija teigia, kad klavišų išdėstymas yra subalansuotas taip, kad dažniausiai spaudžiami klavišai būtų lengvai pasiekiami [6].
- JCUKEN – išdėstymas skirtas kirilica abėcėlės raidynui. Tai priimtas kaip standartinis išdėstymas Rusijoje [4]. Tačiau žmonės, kurie yra pripratę prie QWERTY išdėstymo dažnai naudoja fonetinį kirilica abėcėlės išdėstymą. Jame kirilica simbolis tiesiogiai atitinka lotynišką simbolį QWERTY išdėstyme.

Šiame poskyryje apžvelgti keli patys populiariausi klaviatūrų išdėstymai. Yra dar daug kitų išdėstymų skirtų atskiroms šalims: Vietnamo, Japonijos, Kinijos, Korėjos, Graikijos ir kiti. Taip pat įvairių mažiau populiarių išdėstymų skirtų anglų kalbai: Neo, Malt, QWERTY variacijos, Colemak, Workman ir kiti.

Programinė įranga yra kuriama anglų kalba, todėl buvo pasirinkti sutelkti dėmesį į QWERTY klaviatūros išdėstymą

1.3 Metodų, skirtų fiksuoti vartotojo klaviatūros paspaudimus apžvalga

Norint sekti klaviatūros klavišų paspaudimus yra trys pagrindinės metodų grupės. Vartotojo lygio (angl. user level), branduolio lygio (angl. kernel level) ir mechaninio lygio (angl. hardware level) [7]. Mechaninis duomenų rinkimas iškart atkrenta, nes jis yra nepraktiškas, kadangi jis reikalauja specialios techninės įrangos. Branduolio lygio duomenų rinkimas dažniausiai yra naudojamas kenkėjiškiems tikslams, kadangi jį yra žymiai sunkiau susekti nei vartotojo lygio duomenų rinkimą. Branduolio lygio duomenų rinkimas reikalauja gilių žemo programavimo kalbų lygio žinių ir dokumentacija yra labai ribota.

Dauguma iš metodų buvo atrasti vien dėl kenkėjiškų priežasčių, todėl nemaža jų dalis kaip privalumą pažymi aptikimo sunkumą, kas kuriamai PĮ nėra aktualu. Todėl belieka pasirinkimas tarp vartotojo lygio duomenų rinkimo metodų. Kadangi nerasta informacijos apie šių metodų našumą, lieka tik pagrindiniai kriterijai – funkcionalumas ir panaudojamumo paprastumas [7].

1 lentelė. Klavišų paspaudimų sekimo metodai

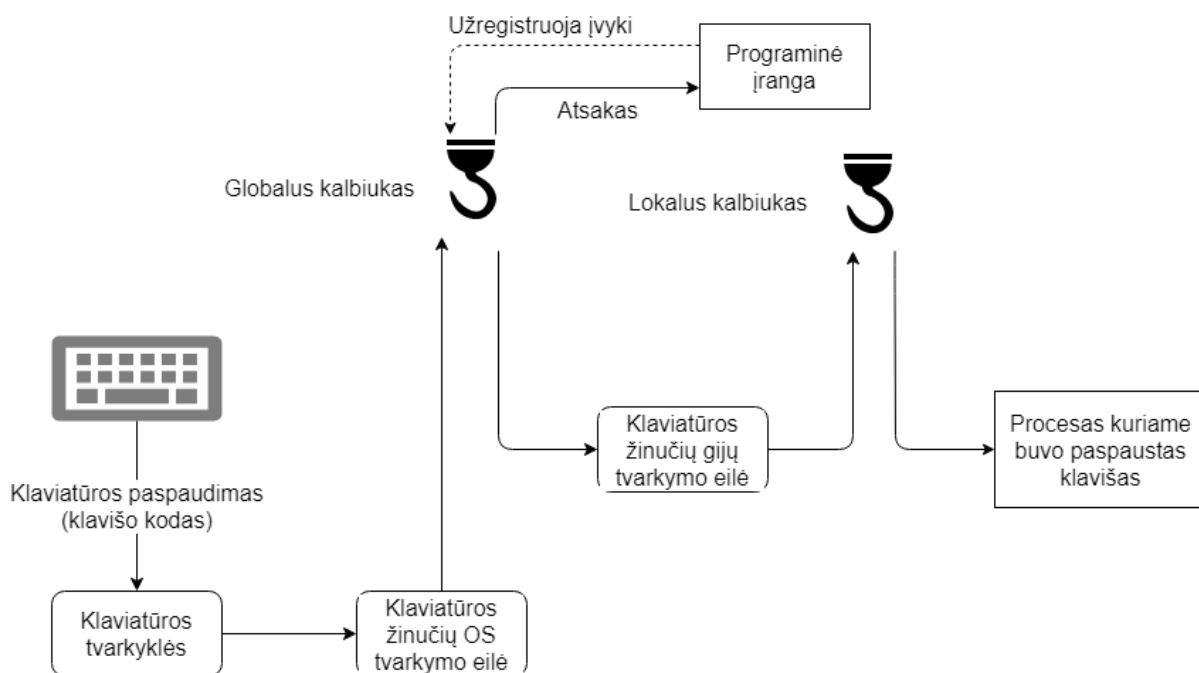
Metodas	Duomenų surinkimas	Sunkumas	Pastabos
Globalūs klaviatūros žinučių kabliukai	Visi duomenys	Paprasta	
Pastovus klaviatūros būklės stebėjimas	Yra šansas, kad ne visi duomenys bus užfiksuoti	Paprasta	Reikalauja papildomos analizės
Į kiekvieną procesą yra įdiegiama klaviatūros žinučių kabliukai injekcijos principu	Yra šansas, kad ne visi duomenys bus užfiksuoti	Sudėtinga	Injekciją į kiekvieną procesą yra didžiulė raudona vėliava antivirusinėms

Neapdorotų įvesčių modelis	Visi duomenys	Paprasta	
----------------------------	---------------	----------	--

Globalūs klaviatūros žinučių kabliukai (angl. global keyboard messasgee hooks) ir neapdorotų įvesčių modelis (angl. raw input model) atrodo tinkamiausi variantai. Abu surenka visus duomenis bei yra lengvai įgyvendinami. Pagrindinis skirtumas tarp šių metodų yra tai, kad kabliukai gali užblokuoti vartotojo įvestis, o neapdorotų įvesčių modelis gali atpažinti, iš kokio prietaiso buvo įvestis atlikta (nešiojamojo kompiuterio klaviatūra, išorinė klaviatūra ar pan.). Tačiau nei viena iš šių funkcijų PĮ nėra reikalinga, todėl pasirinktas klaviatūros žinučių kabliukų metodas, kadangi šio metodo dokumentacija yra labiau išplėtotą.

1.4 Globalių klaviatūros žinučių kabliukų veikimas

Paspaudus klaviatūros klavišą yra sugeneruojamas klavišo kodas ir iškarto perduodamas klaviatūros tvarkyklėms. Jei PĮ nori perrimti klavišų paspaudimus, pirmiausia turi būti užregistruojamas globaliųjų kabliukų įvykis (angl. event) ir tik tada PĮ gali perrimti sugeneruotą kodą naudodama atsako (angl. callback) funkciją [8]. Globalusis kabliukas seka klaviatūros paspaudimus dar OS žinučių tvarkymo eilėje, todėl paspaudimas yra užfiksuojamas, nesvarbu, kurioje programoje buvo paspaustas klavišas. Tuo tarpu lokalusis kabliukas seką paspaudimus tikrai gijų tvarkymo eilėje, todėl perrimti klavišus gali tik iš tos programos, kurioje buvo paspaustas klavišas (žr. 1 pav.).



1 pav. Klaviatūros kabliukų veikimo detalizavimas

1.5 Fone veikiančios programinės įrangos privalumai

Programinė įranga veikianti fone turi du didžiausius privalumus.

- Vartotojas pamiršta, kad programinė įranga yra įjungta. Taip yra gaunami duomenys, kurie atspindi tikrąjį vartotojo darbą su klaviatūra. Jeigu vartotojui reikėtų dirbti tiesiogiai su programine įranga, yra labai didelė tikimybė, kad gauti duomenis būtų surenkami nevisai tikslūs. Vartotojas kruopščiau, greičiau ir kokybiškiau bandytų atlikti reikalingą užduotį.

- Nėra švaistomas vartotojo laikas. Jeigu vartotojui pastoviai reikėtų dirbti su programine įranga, yra rizikuojama parasti vartotoją. Nusibodus dirbti su PĮ vartotojo rezultatai gali suprastėti arba programinė įranga gali būti pašalinta iš kompiuterio.

Programinė įranga skirta stebėti ir fiksuoti vartotojo darbą su klaviatūra dažnai yra naudojama tyrimuose. Pavyzdžiui, programinė įranga, kuri fiksavo paspaudimo laiką, buvo naudojama tyrime, kuris bandė nustatyti ar darbas su klaviatūra gali indukuoti ankstyvos stadijos Parkinsono ligą. Žmonėms, kuriems buvo nustatyta ankstyvos stadijos Parkinsono liga spaudinėjo du klavišus pakaitomis ir vieną klavišą pastoviai lėčiau, nei tie žmonės, kuriems ši liga nebuvo nustatyta[9]. Kitas tyrimas bandė nustatyti, ar sekant vartotojo darbą su klaviatūra galima aptikti motorinių funkcijų sutrikimus, kurie indukuotų ankstyvos stadijos Alzheimerio ligą. Tyrimas nustatė, kad vien senstant žmonių darbo kokybė ir sparta su klaviatūra sparčiai lėtėja, tačiau tų asmenų, kuriems dar yra nustatyta ankstyvos stadijos Alzheimerio sindromas, darbo sparta sulėtėja dar labiau[10].

1.6 Analogų palyginimas

Yra trys pagrindinės priežastis kodėl yra kuriama programinė įranga.

- Rinkoje nėra to, ką galėtų pasiūlyti kuriama programinė įranga.
- Rinkoje esanti programinė įranga yra neišbaigta, kuriama PĮ bus kokybiškesnė.
- Rinkoje esanti programinė įranga yra per brangi, siekiama gauti vartotojų žemesnėje kainų grupėje ar išvis platinant PĮ nemokamai.

Apžvelgus analogus paaiškėjo, kad tokio tipo PĮ yra laisvos vietos. Vieni analogai yra gana neišbaigti, kiti yra mokami arba turi mokamų funkcijų.

2 lentelė. Analogų palyginimas

	Application usage statistics [11]	WhatPulse [12]	RSIGuard [13]	Keyboard usage [14]	Keyboard optimizer (ibm research) [15]
Veikia kompiuterio fone (angl. background)	Taip	Taip	Taip	Taip	Ne
Nemokama	Taip	Dalinai. Yra „Premium“ versija. 1,5\$/mėn.	Ne. Kaina 65\$	Taip	Taip
Platus Windows OS palaikymas	Taip	Taip	Taip	Taip	Ne
Android ar IOS palaikymas	Ne	Ne	Ne	Ne	Ne
Kontaktuoja su išoriniu serveriu	Ne	Taip	Taip	Ne	Ne
Karščio žemėlapis	Ne	Taip	Taip. 3 tipų	Taip.	Ne
Rašymo greičio matavimas	Taip	Ne	Ne	Ne	Ne
Naudojamų	Taip	Taip	Ne	Ne	Ne

programų sekimas					
Pelės sekimas	Taip	Taip	Taip	Ne	Ne
Gramatinių klaidų sekimas	Ne	Ne	Ne	Dalinai. Fiksuoja tik proporciją tarp daromų klaidų ir paprasto teksto	Dalinai. Atsižvelgia ar vartotojas nespaudžia to pačio mygtuko daugiau kaip kartą
Duomenų filtravimas	Ne	Taip, datos filtravimas, tačiau programų filtravimas yra mokamas	Taip, datos filtravimas	Ne	Ne
Keičia OS nustatymus	Ne	Ne	Ne	Ne	Taip. Keičia klaviatūros rašymo dažnumą laikant klavišą

Iš šių programų tik dvi fiksuoja duomenis, kurie primena daromas gramatines klaidas darbo su klaviatūra metu. „Keyboard usage“ fiksuoja tiksliai proporciją tarp rašymo, klaidų darymo ir redagavimo. Tuo tarpu „Keyboard optimizer“ fiksuoja klaidas pagal vartotojui duotą parašyti tekstą ir pagal tai pakeičia OS nustatymus. Ši programa yra skirta su žmonėms kurie turi problemų dirbant su klaviatūra.

Taip pat svarbu paminėti, kad dauguma iš šių programų pateikia darbo su klaviatūra statistiką, tačiau ji yra ganėtinai ribota. „WhatPulse“ pateikia karščio žemėlapi su paspaustų mygtukų kiekiu, laiką kiek jis praleido dirbant su klaviatūra ar pele, tačiau vien iš tokios informacijos vartotojui yra neįmanoma padaryti išvadas, ar jis nors kiek patobulėjo dirbant su klaviatūra.

„RSIGuard“ yra orientuota į vartotojo darbo su klaviatūra kokybę, kad vartotojas darytų reguliarias pertraukas darbo metu. Tačiau ji turi kelias naudingas funkcijas ir darbo su klaviatūra atžvilgiu. Šios programos karščio žemėlapių statistiką rodo aktualią informaciją – mygtukų paspaudimo dažnumą, laiko tarpą tarp paspaudimų bei jėgą, kokia buvo paspaustas mygtukas. Nors ir pateiktos statistikos nėra daug, tačiau vartotojas panaudojęs šiuos duomenis gali padaryti išvadas, kur reiktų patobulėti.

„Application usage statistics“ fiksuoja klaviatūros, pelės ir programų statistiką. Jos yra gana nemažai, tačiau ji yra „sausą“. Paspaudimų kiekis, praleistas laikas dirbant su programa, paspaudimų greitis, dažnumas. Tačiau iš šių duomenų nėra atliekama jokia analizė. Naudojantis tokiais duomenimis vartotojas negalės padaryti išvadų apie savo darbo su klaviatūra tobulėjimą ar daromas klaidas.

Šiuo metu rinkoje esanti programinė įranga turi daug minusų tobulinant vartotojo darbą su klaviatūra. Kai kurios programos pateikia nemažai statistikos, tačiau tik labai maža dalis yra aktuali

virtotojui, kuris nori tobulinti savo darbu su klaviatūra. Taip pat vienos programos yra mokamos, o kitos pateikia tik nedidelę dalį statistikos.

2. REIKALAVIMŲ SPECIFIKACIJA

2.1 Sistemos paskirtis

2.1.1 Projekto kūrimo pagrindas

Klaviatūros darbo sekimo servisų Windows aplinkoje yra gana nemažai, tačiau retas turi funkcijų, kurios padėtų optimizuoti darbą su klaviatūra. Ir dar retesni atvejai, kada šie servisai daro gilią surinktų duomenų analizę. O ir jų daroma analizė nėra orientuota į vartotojo darbo su klaviatūra optimizavimą ir dažnai tokia PĮ yra mokama ar bent jau dalis jos funkcijų yra mokamos.

2.1.2 Sistemos tikslai

Sistemos tikslas yra sekti vartotojo darbą su kompiuterio klaviatūra fone. Fiksuoti gaunamus duomenis ir juos analizuoti taip, kad vėliau pateikti duomenys vartotojui būtų aktualūs ir padėtų jam tobulėti, dirbat su kompiuterio klaviatūra.

Sistemos savybės:

- Paprastumas – sistemos pagrindinės funkcijos turi būti lengvai suprantamos visų amžių bei įgūdžių žmonėms.
- Informacijos aktualumas – sistema turi pateikti vartotojui, kuris nori tobulėti dirbti su klaviatūra, aktualią informaciją, kuri galėtų jam tai padėti.
- Funkcionalumas – sistema turi pateikti duomenis lengvai suprantamu formatu bei leisti juos filtruoti.
- Našumas – sistema turi veikti vartotojo kompiuteryje nepastebimai. Ji neturi sukelti jokių našumo sutrikimų.
- Stabilumas – sistema turi būti stabili. Pati neišsijunginėti, vartotojui dirbant su klaviatūra neturi rodyti jokių klaidų pranešimų ar kaip nors kitaip trukdyti vartotojo darbą.

2.2 Projekto dalyviai

2.2.1 Užsakovas

Simona Ramanauskaitė,

El. Paštas simram@it.su.lt,

Tel. Nr. +370 614 37184

2.2.2 Vykdytojas

Arnas Kučinskas,

El. paštas arnaskucinskas@gmail.com,

Tel. Nr. +370 626 71459

2.3 Vartotojai

Sistemoje numatyta tik viena vartotojų kategorija – vartotojas. Jo savybės aprašytos 3-ioje lentelėje.

3 lentelė. Vartotojo „Vartotojas“ detalizavimas

Vartotojo kategorija	Vartotojas
Vartotojo sprendžiami uždaviniai	Programos kalibravimas, klaidų, analizės ir statistikos peržiūra.
Patirtis dalykinėje srityje	Nereikalinga
Patirtis informacinėse technologijose	Naujokas – specialistas. Rekomenduojama gebėti atskirti skirtingas programas, procesus operacinėje sistemoje.
Papildomos vartotojo charakteristikos	Anglų kalbos įgūdžiai.
Apsimokymo poreikis	Minimalus
Amžiaus grupė	10-80

2.4 Įpareigojantys apribojimai

2.4.1 Apribojimai sprendimui ir diegimo aplinka

- Apribojimai sprendimui - C# programavimo kalba, Visual Studio 2017 IDE, Microsoft .NET Framework 4.5.2 ir naujesnės versijos, SQLite duomenų bazė
- Diegimo aplinka – Windows 7 ir naujesnės OS versijos, Microsoft .NET Framework 4.5.2. ir naujesnės versijos.

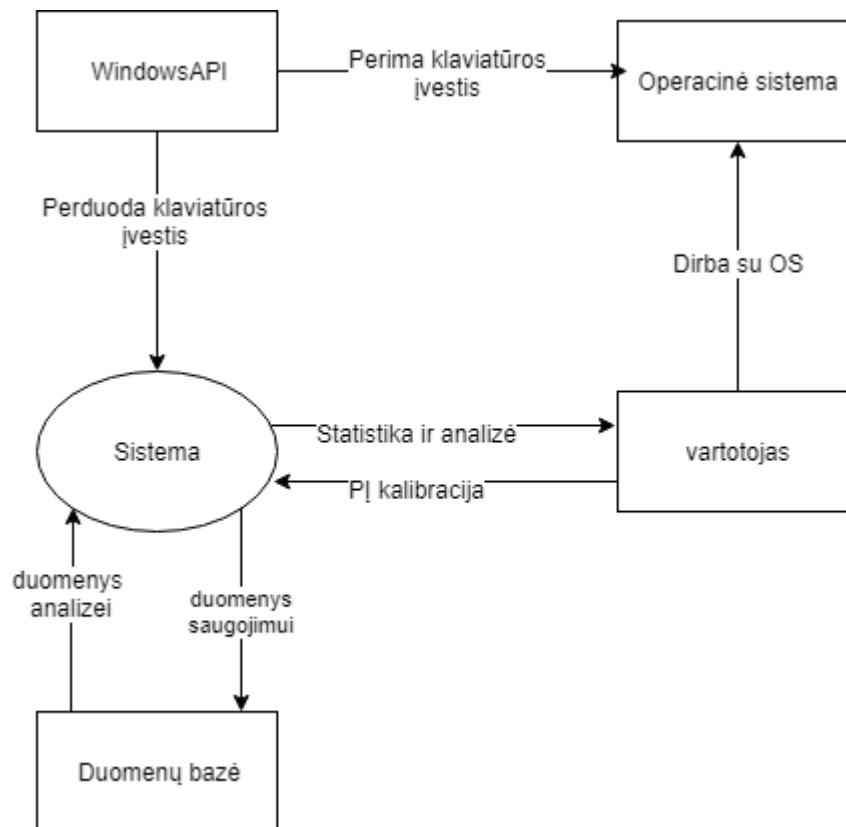
2.4.2 Bendradarbiaujančios sistemos

Programinė įranga nefunkcionuos teisingai neveikiant šioms sistemoms:

- Microsoft .NET karkasas
- Win32 API
- Windows OS

2.5 Veiklos kontekstas

Vartotojas dirba su operacine sistema. Sistema naudodama Win32 API gali perrimti visus vartotojo paspaudimus. Kiekvienas paspaudimas ar paspaudimų seka yra atitinkamai fiksuojama, klasifikuojama ir išsaugoma duomenų bazėje. Vėliau sistema iš duomenų bazės išanalizuoja duomenis, kurie yra pateikiami vartotojui (žr. 2 pav.).



2 pav. Veiklos konteksto diagrama

Vartotojas turi atlikti kalibravimą, norint, kad sistema veiktų. Kalibravimas nustato, kiek laiko yra sekami vartotojo paspaudimai, iki kada sistema traktuos sekantį paspaudimą kaip pirmąjį, tai reiškia, kad sekančio klavišo „paspaudimo laikas tarp klavišų“ bus 0 ms. Jei šis laiko tarpas yra per mažas, tada didžioji dalis vartotojo paspaudimų bus užfiksuojami kaip pirmieji, jeigu laiko tarpas yra per didelis, vartotojui sustojus rašyti ir po tam tikro laiko vėl pradėjus rašyti vis dar bus fiksuojamas paspaudimo laikas tarp klavišų.. Todėl kalibravimas turi būti atliekama kiekvienam vartotojui individualiai.

Duomenys iš Win32 API gaunami tada, kai vartotojas naudoja kompiuterio klaviatūrą. Tuo tarpu duomenys į duomenų bazę gali būti siunčiami ne būtinai kiekvienam duomenų gavimui iš Win32 API, t.y. galimas duomenų srauto optimizavimas, kada duomenys kaupiami sistemoje ir į duomenų bazę įrašomi dalimis arba papildomi dalinai apdorotų duomenų, lengvesnei jų analizei ir didesniai PĮ našumui.

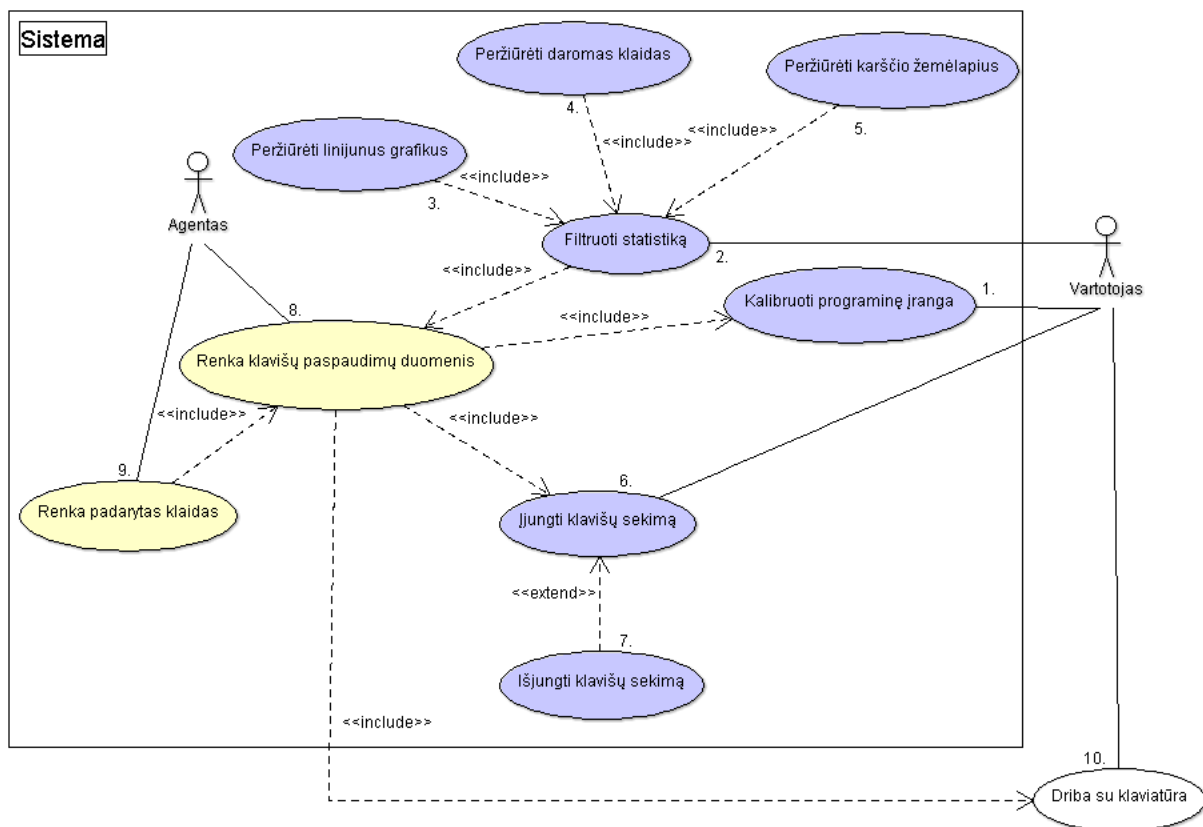
2.6 Produkto veiklos sfera

2.6.1 Sistemos ribos

Gilinantį į vartotojo galimas funkcijas su sistema, pastebima, kad vartotojas praktiškai niekaip nepastebi duomenų mainų tarp Win32 API ir sistemos bei sistemos ir jos duomenų bazės. Vartotojo funkcijų realizacija apima tik sekimo įjungimą/išjungimą, PĮ kalibravimą bei sukauptų duomenų peržiūrą ir filtravimą.

Diagramoje pavaizduotas veikėjas vardu „Agentas“, tai iliustruoja, kad sistema stebi vartotojo darbą su klaviatūra, net kai jis yra iš sistemos ribų t.y., kai vartotojas tiesiogiai

nesinaudoja jokiais PĮ funkcijomis. Visi duomenys, kurie yra naudojami statistikoje, yra surenkami agento (žr. 3 pav.).



3 pav. Sistemos vartotojai ir funkcijų tarpusavio priklausomybės

2. 6. 2 Panaudos atvejų sąrašas

4 lentelė. Panaudojimo atvejo „Kalibruoja programinę įrangą“ detalizavimas

1. PANAUDOJIMO ATVEJIS:	Kalibruoti programinę įrangą
Vartotojas/Aktorius:	Vartotojas
Aprašas:	Vartotojas kalibruoja PĮ, kad ji būtų pritaikyta jo darbo stiliui
Prieš sąlyga:	-
Sužadinimo sąlyga:	Kalibravimas atliekama pirmą kartą įjungus PĮ. Arba bet kada vartotojui panorėjus perkalibruoti PĮ
Po-sąlyga:	PĮ nustatymai yra pritaikomi pagal kalibravimo rezultatus

5 lentelė. Panaudojimo atvejo „Filtruoja statistiką“ detalizavimas

2. PANAUDOJIMO ATVEJIS:	Filtruoti statistiką
Vartotojas/Aktorius:	Vartotojas
Aprašas:	Vartotojas gali filtruoti statistiką pagal OS programas, kuriose dirbo su klaviatūra, ir datą, datos intervalą ir nustatytus filtrus.
Prieš sąlyga:	Vartotojas atsidaro langą, kuriame rodoma statistika.
Sužadinimo sąlyga:	Vartotojas nori matyti išfiltruotus duomenis.
Po-sąlyga:	Vartotojas mato filtruotus duomenis.

6 lentelė. Panaudojimo atvejo „Peržiūrėti linijinius grafikus“ detalizavimas

3. PANAUDOJIMO ATVEJIS:	Peržiūrėti linijinius grafikus
Vartotojas/Aktorius:	Vartotojas
Aprašas:	Vartotojas gali peržiūrėti statistiką linijinių grafikų forma.
Prieš sąlyga:	Statistika turi būti išfiltruojama (jei vartotojas nenurodo kitaip, sistema rodo paskutinių 30 dienų duomenis)

Sužadinimo sąlyga:	Vartotojas nori pamatyti statistiką, susijusią su rašymo greičiu ir klaidomis
Po-sąlyga:	Vartotojas mato linijinius grafikus, kurie vaizduoja statistiką, susijusią su rašymo greičiu ir klaidomis

7 lentelė. Panaudojimo atvejo „Pamatyti daromas klaidas“ detalizavimas

4. PANAUDOJIMO ATVEJIS:	Peržiūrėti daromas klaidas
Vartotojas/Aktorius:	Vartotojas
Aprašas:	Vartotojui rodoma, kokias klaidas, kokiuose žodžiuose ir kaip dažnai jis daro
Prieš sąlyga:	Statistika turi būti išfiltruojama
Sužadinimo sąlyga:	Vartotojas daro klaidas dirbdamas su klaviatūra
Po-sąlyga:	Vartotojas mato daromas klaidas

8 lentelė. Panaudojimo atvejo

5. PANAUDOJIMO ATVEJIS:	Peržiūrėti karščio žemėlapius
Vartotojas/Aktorius:	Vartotojas
Aprašas:	Vartotojas gali peržiūrėti dažniausiai spaudžiamų klavišų ir laiko tarpo tarp klavišų paspaudimų karščio žemėlapius
Prieš sąlyga:	Statistika turi būti išfiltruojama
Sužadinimo sąlyga:	Vartotojas dirba su kompiuterio klaviatūra
Po-sąlyga:	Vartotojas mato klaviatūros, dieninį ir savaitinius karščio žemėlapius.

9 lentelė. Panaudojimo atvejo „Išjungti klavišų sekimą“ detalizavimas

6. PANAUDOJIMO ATVEJIS:	Išjungti klavišų sekimą
Vartotojas/Aktorius:	Vartotojas
Aprašas:	Vartotojas turi turėti galimybę išjungti klaviatūros sekimą neišjungus PĮ
Prieš sąlyga:	Sekimas turi būti įjungtas
Sužadinimo sąlyga:	Noras, kad duomenys būtų nustojami sekti.
Po-sąlyga:	Klaviatūros sekimas išjungiamas.

10 lentelė. Panaudojimo atvejo „Įjungti klavišų sekimą“ detalizavimas

7. PANAUDOJIMO ATVEJIS:	Įjungti klavišų sekimą
Vartotojas/Aktorius:	Vartotojas
Aprašas:	Vartotojas turi turėti galimybę įjungti klaviatūros sekimą neperkraunant PĮ
Prieš sąlyga:	Sekimas išjungtas
Sužadinimo sąlyga:	Noras, kad būtų pradedami rinkti vartotojo darbo su klaviatūra duomenys
Po-sąlyga:	Sekimas įjungtas

11 lentelė. Panaudojimo atvejo „Renka klavišų paspaudimų duomenis“ detalizavimas

8. PANAUDOJIMO ATVEJIS:	Renka klavišų paspaudimų duomenis
Vartotojas/Aktorius:	Agentas
Aprašas:	PĮ seką ir fiksuoją vartotojo darbą su klaviatūra duomenis
Prieš sąlyga:	Sekimas įjungtas ir vartotojas dirba su klaviatūra
Sužadinimo sąlyga:	Vyksta vartotojo darbas su klaviatūra
Po-sąlyga:	Fiksuojami paspaustų klavišų duomenys

12 lentelė. Panaudojimo atvejo „Renka padarytas klaidas“ detalizavimas

9. PANAUDOJIMO ATVEJIS:	Renka padarytas klaidas
Vartotojas/Aktorius:	Agentas
Aprašas:	Iš surinktų vartotojo darbo su klaviatūra duomenų yra fiksuojamos padarytos klaidos
Prieš sąlyga:	Yra renkami klavišų paspaudimų duomenys

Sužadavimo sąlyga:	Vartotojas ištrina simbolį ar žodį
Po-sąlyga:	Klaida yra užfiksuojama

13 lentelė. Panaudojimo atvejo „Dirba su klaviatūra“ detalizavimas

10. PANAUDOJIMO ATVEJIS:	Dirba su klaviatūra
Vartotojas/Aktorius:	Vartotojas
Aprašas:	Vartotojas dirba su klaviatūra už sistemos ribų.
Prieš sąlyga:	-
Sužadavimo sąlyga:	Vartotojui reikia pasinaudoti klaviatūra
Po-sąlyga:	Vartotojas dirba su klaviatūra

2.7 Funkciniai reikalavimai

14 lentelė. Funkcinio reikalavimo „PĮ kalibravimas“ detalizavimas

Reikalavimas nr.:	1.	Reikalavimo tipas:	2.7	Įvykis/panaudojimo atvejo nr.:	1
Aprašymas:	PĮ kalibravimas				
Pagrindimas:	Programinė įranga turi būti sukalibruota atsižvelgiant į vartotojo pajėgumą dirbant su klaviatūra. Kalibravimas turi būti kuo paprastesnis procesas vartotojui, tačiau tai turi būti galima padaryti ir rankiniu būdu.				
Šaltinis:	Vartotojas				
Tinkamumo kriterijus:	Atlikus kalibravimą duomenų bazėje turi pasikeisti „vartotojo lygio“ reikšmė.				
Užsakovo tenkinimas:	4	Užsakovo netenkinimas:	3		
Priklausomybės:	–	Konfliktai:	Nėra		

15 lentelė. Funkcinio reikalavimo „Statistikos filtravimas“ detalizavimas

Reikalavimas nr.:	2.	Reikalavimo tipas:	2.7	Įvykis/panaudojimo atvejo nr.:	2
Aprašymas:	Statistikos filtravimas				
Pagrindimas:	Surinktą statistiką vartotojas turi galėti filtruoti pagal datą, datos intervalą, tai leidžia peržiūrėti aktualius duomenys. Taip pat pagal programas, kuriuose statistika buvo užfiksuota, tai leidžia atmesti programas kurios „suteršia“ statistiką. Išfiltruotų programų duomenys toliau turi būti nebesekami programinės įrangos.				
Šaltinis:	Vartotojas				
Tinkamumo kriterijus:	Pasirinkus filtrus turi būti atvaizduojama statistiką tik kuri buvo užfiksuota atitinkamoje programoje ar datą, kad įvykis buvo užfiksuotas. Išfiltravus programą duomenys iš jos nebėra sekami.				
Užsakovo tenkinimas:	5	Užsakovo netenkinimas:	5		
Priklausomybės:	8	Konfliktai:	Nėra		

16 lentelė. Funkcinio reikalavimo „Linijinių grafikų peržiūrėjimas“ detalizavimas

Reikalavimas nr.:	3.	Reikalavimo tipas:	2.7	Įvykis/panaudojimo atvejo nr.:	3
Aprašymas:	Linijinių grafikų peržiūrėjimas				
Pagrindimas:	Linijiniai grafikai turi atvaizduoti statistiką kuri susijusi su vartotojo rašymo greičiu ir daromomis klaidomis. Su linijiniais grafikais galima identifikuoti vartotojo optimalų rašymo greitį, kiek laiko užima ištaisyti klaidas, kiek klaidų yra daroma priklausomai nuo rašymo greičio, vidutinį rašymo greitį tam tikram laiko intervalui. Tai padėtų vartotojui suvokti jo rašymo stilių, kokius greičius jam geriausia yra dirbti su klaviatūra bei kada jis daro daugiausiai gramatikos klaidų.				
Šaltinis:	Vartotojas				
Tinkamumo kriterijus:	Vartotojas atsidaręs linijinių grafikų langą gali pasirinkti atvaizduoti vieną iš keturių tipų linijinius grafikus.				
Užsakovo tenkinimas:	5	Užsakovo netenkinimas:	5		
Priklausomybės:	2	Konfliktai:	Nėra		

17 lentelė. Funkcinio reikalavimo „Daromų gramatinių klaidų peržiūrėjimas“ detalizavimas

Reikalavimas nr.:	4.	Reikalavimo tipas:	2.7	Įvykis/panaudojimo atvejo nr.:	4
Aprašymas:	Daromų gramatinių klaidų peržiūrėjimas				
Pagrindimas:	Klaidos turi būti atvaizduojamos dviem tipais: dažniausiai ištrinami simboliai ir žodžiai, kuriuose kyla daugiausiai klaidų. Tai padės vartotojui identifikuoti jo dažniausiai daromas gramatines klaidas t.y., kurie simboliai yra dažniausiai trunami bei kuriuos žodžius yra sunkiausiai rašyti.				
Šaltinis:	Vartotojas				
Tinkamumo kriterijus:	Vartotojas atsidaręs daromų klaidų sąrašus gali pasirinkti dviejų tipų sąrašus.				
Užsakovo tenkinimas:	5	Užsakovo netenkinimas:	5		
Priklausomybės:	2	Konfliktai:	Nėra		

18 lentelė. Funkcinio reikalavimo „Karščio žemėlapių peržiūrėjimas“ detalizavimas

Reikalavimas nr.:	5.	Reikalavimo tipas:	2.7	Įvykis/panaudojimo atvejo nr.:	5
Aprašymas:	Karščio žemėlapių peržiūrėjimas				
Pagrindimas:	Karščio žemėlapiai turi būti dviejų tipų: klavišų paspaudimų skaičiaus bei laiko tarpo tarp paspaudimų. Tai leis vartotojui identifikuoti jo lėčiausiai spaudžiamus klavišus, dienas ir valandas, kada jis dirba lėčiausiai ar daugiausiai.				
Šaltinis:	Vartotojas				
Tinkamumo kriterijus:	Vartotojas atsidaręs karščio žemėlapių langą turi matyti klaviatūros, dienos ar savaitės karščio žemėlapius sužymėtus spalvomis, nuo baltos iki raudonos.				
Užsakovo tenkinimas:	5	Užsakovo netenkinimas:	5		
Priklausomybės:	2	Konfliktai:	Nėra		

19 lentelė. Funkcinio reikalavimo „Įjungti klavišų sekimą“ detalizavimas

Reikalavimas nr.:	6.	Reikalavimo tipas:	2.7	Įvykis/panaudojimo atvejo nr.:	6
Aprašymas:	Įjungti klavišų sekimą				
Pagrindimas:	Vartotojui turi būti aiškiai nurodoma, kad jis įjungia klaviatūros sekimą. Kad PĮ nepradėtų veikti fone jam to pilnai nesuvokiant.				
Šaltinis:	Vartotojas				
Tinkamumo kriterijus:	Įjungus klavišų sekimą programa pradeda rinkti vartotojo darbą su klaviatūra.				
Užsakovo tenkinimas:	3	Užsakovo netenkinimas:	3		
Priklausomybės:	–	Konfliktai:	Nėra		

20 lentelė. Funkcinio reikalavimo „Išjungti klavišų sekimą“ detalizavimas

Reikalavimas nr.:	7.	Reikalavimo tipas:	2.7	Įvykis/panaudojimo atvejo nr.:	7
Aprašymas:	Išjungti klaviatūros sekimą				
Pagrindimas:	Vartotojas gali norėti laikinai sustabdyti jo darbą su klaviatūra sekimą. Ši funkcija skirta, kad tam padaryti nereiktų išjungti visos PĮ				
Šaltinis:	Vartotojas				
Tinkamumo kriterijus:	Išjungus klavišų sekimą sistema nebefiksuoja jokių vartotojo darbo su klaviatūra duomenų				
Užsakovo tenkinimas:	3	Užsakovo netenkinimas:	3		
Priklausomybės:	6	Konfliktai:	Nėra		

21 lentelė. Funkcinio reikalavimo „Klavišų paspaudimų duomenų rinkimas“ detalizavimas

Reikalavimas nr.:	8.	Reikalavimo tipas:	2.7	Įvykis/panaudojimo atvejo nr.:	8
Aprašymas:	Klavišų paspaudimų duomenų rinkimas				
Pagrindimas:	Norint atlikti bet kokią analizę iš pat pradžių reikia turėti duomenų, ši funkcija skirta surinkti visus duomenis kurie vėliau yra naudojami analizei.				
Šaltinis:	Agentas				
Tinkamumo kriterijus:	Vartotojo darbo su klaviatūra metu yra renkami tokie duomenys (paspaustas klavišas, sugeneruotas simbolis, data, kada klavišas buvo paspaustas ir programa kurioje klavišas buvo paspaustas.).				
Užsakovo tenkinimas:	5	Užsakovo netenkinimas:	5		

Priklausomybės:	1, 6, 10	Konfliktai:	Nėra
------------------------	----------	--------------------	------

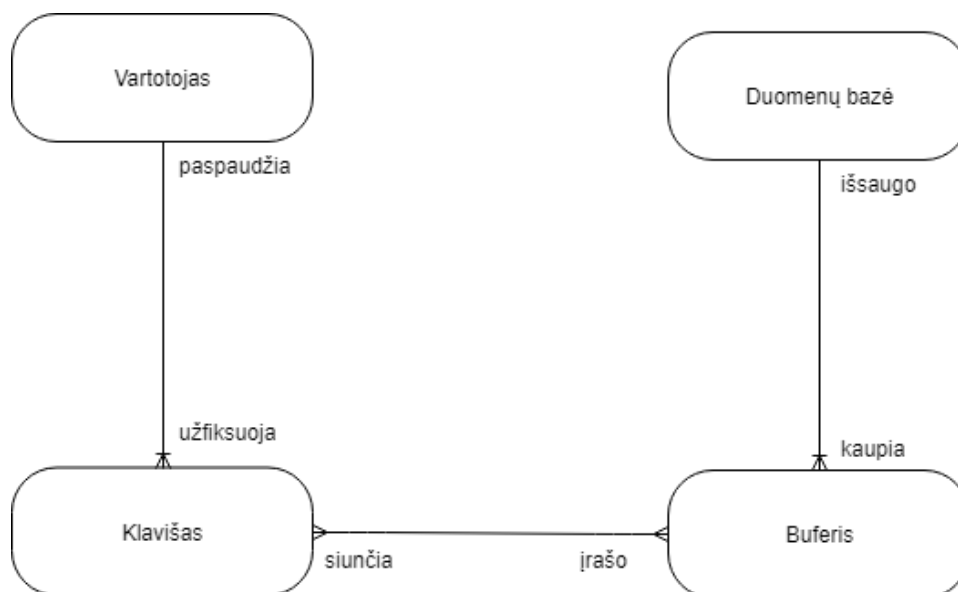
22 lentelė. Funkcinio reikalavimo „Padarytų klaidų rinkimas“ detalizavimas

Reikalavimas nr.:	9.	Reikalavimo tipas:	2.7	Įvykis/panaudojimo atvejo nr.:	9
Aprašymas:	Padarytų klaidų rinkimas				
Pagrindimas:	Norint atvaizduoti panaudos atvejo nr. 4 statistiką turi būti papildomai renkamos padarytos klaidos				
Šaltinis:	Vartotojas				
Tinkamumo kriterijus:	Vartotojui trinant simbolius ir juos keičiant kitais tai turi būti užfiksuota sistemoje.				
Užsakovo tenkinimas:	3	Užsakovo netenkinimas:	3		
Priklausomybės:	8	Konfliktai:	Nėra		

2. 7. 1 Reikalavimai duomenims

Esybių ryšiu diagramoje galime matyti supaprastintą duomenų vaizdą (žr. 4 pav.).

Vartotojas bet kuriuo metu gali paspausti klavišą, jų kombinacijas ar kelis klavišus iš karto. Sistema juos užfiksuoja ir įrašo į buferį arba buferius. Priklausomai nuo nuspausto klavišo ir esamos sistemos būsenos klavišas gali būti įrašytas daugiau nei vieną buferį. Tai daroma paprastesniam duomenų rūšiavimui bei našumui pagerinti. Buferiai kaupia klavišų paspaudimus, jų sekas, ištrintus bei pakeistus žodžius. Įvykus tam tikram įvykiui ar praėjus tam tikram laiko kiekiui (pvz.: vartotojas ištaiso klaidą arba klavišų duomenys jau yra buferyje 10 sekundžių) iš buferių duomenys yra išsaugomi duomenų bazėje.



4 pav. Sistemos duomenų loginė struktūra

2. 8 Nefunkciniai reikalavimai

2. 8. 1 Reikalavimai sistemos išvaizdai

23 lentelė. Nefunkcinio reikalavimo „Neįkyri sąsaja“ detalizavimas

Reikalavimas nr.:	13.	Reikalavimo tipas:	08	Įvykis/panaudojimo atvejo nr.:	–
Aprašymas:	Neįkyri sąsaja				
Pagrindimas:	Siekiant patobulinti vartotojo darbą vienoje srityje, negalima jo apkrauti kitoje. Jeigu PĮ bus įkyri, vartotojas gali labai greitai tiesiog ištrinti programą be jokių dvejonų				
Šaltinis:	Vartotojas				

Tinkamumo kriterijus:	PĮ neturi rodyti klaidos pranešimų, išsijunginėti ar kitaip trukdyti vartotojo darbą su klaviatūra.		
Užsakovo tenkinimas:	4	Užsakovo netenkinimas:	4
Priklausomybės:	–	Konfliktai:	Nėra

2. 8. 2 Reikalavimai panaudojamumui

24 lentelė. Nefunkcinio reikalavimo „Paprastai panaudojama“

Reikalavimas nr.:	14.	Reikalavimo tipas:	08	Įvykis/panaudojimo atvejo nr.:	–
Aprašymas:	Paprastai panaudojamos pagrindinės funkcijos vidutinio asmens be apsimokymo				
Pagrindimas:	Programa skirta labai plačiai amžiaus grupei, todėl ja turi galėti naudotis kiekviena iš jų be jokių problemų				
Šaltinis:	Vartotojas				
Tinkamumo kriterijus:	Pagrindinėmis programos funkcijomis be jokių paaiškinimų turi sugebėti pasinaudoti bent 90% žmonių nurodytose amžiaus grupėse				
Užsakovo tenkinimas:	4	Užsakovo netenkinimas:	4		
Priklausomybės:	–	Konfliktai:	Nėra		

2. 8. 3 Reikalavimai vykdymo charakteristikoms

25 lentelė. Nefunkcinio reikalavimo „Vartotojo sąsajos sparta“ detalizavimas

Reikalavimas nr.:	15.	Reikalavimo tipas:	08	Panaudojimo atvejo nr.:	–
Aprašymas:	Grafinė vartotojo sąsaja turi atsidaryti greitai				
Pagrindimas:	Programa iš vartotojo pusės neturi daugybės funkcijų, todėl ir turėtų atsidaryti ne ilgiau kaip per 250 ms				
Šaltinis:	Vartotojas				
Tinkamumo kriterijus:	Grafinė vartotojo sąsaja turi atsidaryti ne ilgiau kaip per 250 ms				
Užsakovo tenkinimas:	3	Užsakovo netenkinimas:	3		
Priklausomybės:	–	Konfliktai:	Nėra		

26 lentelė. Nefunkcinio reikalavimo „Duomenų atnaujinimas“ detalizavimas

Reikalavimas nr.:	16.	Reikalavimo tipas:	08	Panaudojimo atvejo nr.:	–
Aprašymas:	Duomenys sąsajoje turi atsinaujinti perkrovus formą				
Pagrindimas:	Vartotojas atsidaręs formą iš naujo nori matyti naujausius duomenis.				
Šaltinis:	Vartotojas				
Tinkamumo kriterijus:	Iš naujo atidarius formą yra rodomi patys naujausi (ne senesni kaip 10 sekundžių)				
Užsakovo tenkinimas:	3	Užsakovo netenkinimas:	2		
Priklausomybės:	–	Konfliktai:	Nėra		

27 lentelė. Nefunkcinio reikalavimo „PĮ našumas“ detalizavimas

Reikalavimas nr.:	17.	Reikalavimo tipas:	08	Panaudojimo atvejo nr.:	–
Aprašymas:	Programa turi veikti fone nepastebimai				
Pagrindimas:	Vartotojas neturi pajusti jokio sistemos sulėtėjimo susirašius programą. Kitu atveju didžioji dalis vartotojų programa nesinaudos				
Šaltinis:	Vartotojas				
Tinkamumo kriterijus:	Programa veikianti fone neturi vartoti daugiau kaip 5% CPU resursų. Rašymai į diską turi būti kuo retesni (tik kas 10 s, išjungiant programą ir vartotojui padarius klaidą)				
Užsakovo tenkinimas:	5	Užsakovo netenkinimas:	5		
Priklausomybės:	–	Konfliktai:	Nėra		

28 lentelė. Nefunkcinio reikalavimo „PĮ stabilumas“ detalizavimas

Reikalavimas nr.:	17.	Reikalavimo tipas:	08	Panaudojimo atvejo nr.:	–
Aprašymas:	Programa turi veikti stabiliai				
Pagrindimas:	PĮ turi būti stabilus, neišsijunginėti ir nestrikinėti, ji neturi trukdyti vartotojui dirbti su klaviatūra.				
Šaltinis:	Vartotojas				
Tinkamumo kriterijus:	Įjungus programinę įrangą ji turi išsijunkti tik vartotojui ją išjungus. Vartotojui dirbant su klaviatūra neturi būti rodoma jokių klaidų pranešimų. Šios sąlygos turi būti patenkintos PĮ veikiant fone ne mažiau kaip savaitę.				
Užsakovo tenkinimas:	5	Užsakovo netenkinimas:	5		
Priklausomybės:	–	Konfliktai:	Nėra		

2. 8. 4 Reikalavimai saugumui

29 lentelė. Nefunkcinio reikalavimo „Konfidencialumas“ detalizavimas

Reikalavimas nr.:	18.	Reikalavimo tipas:	08	Panaudojimo atvejo nr.:	–
Aprašymas:	Konfidencialumas				
Pagrindimas:	Vartotojo duomenys ir statistika nėra siunčiama į jokią išorinę serverį. Taip vartotojas jausis žymiai saugiau naudojantis PĮ				
Šaltinis:	Vartotojas				
Tinkamumo kriterijus:	PĮ nesiunčia jokių duomenų į išorinius serverius ir išvis niekada nesinaudoja interneto prieiga.				
Užsakovo tenkinimas:	4	Užsakovo netenkinimas:	4		
Priklausomybės:	–	Konfliktai:	Nėra		

2. 9 Galimos sistemos kūrimo rizikos

30 lentelė. Sistemos kūrimo rizikos

Rizikos faktorius	Tikimybinis įvertinimas, %
Reikalavimų specifikacijos pasikeitimai realizavimo fazėje	10
Sistema pradės per lėtai veikti	20
Duomenų bazės struktūros kitimas realizavimo metu	15
Nebus įgyvendintos numatytos funkcijos	15

3. SISTEMOS ARCHITEKTŪROS SPECIFIKACIJA

3.1 Architektūros specifikacijos paskirtis

Sistemos architektūros specifikacijos paskirtis yra atvaizduoti bendrą ir detalų sistemos funkcionavimą, veikimą, apribojimus statinį ir dinaminį vaizdą aprašant būsenų, sekų, klasių ir kitas diagramas.

3.2 Architektūros pateikimas

- Veiklos diagramos – iliustruoja, kaip sistemoje vyksta klavišų fiksavimas, duomenų saugojimas buferiuose ir įrašymas į duomenų bazę.
- Sekų diagramos – iliustruoja, kaip yra įrašomi duomenys esant vienai iš dažniausiai vykstančių stadijų sistemoje, kaip vyksta skaitymo ir rašymo operacijos duomenų bazėje bei kaip vyksta statistikos atvaizdavimas.
- Būsenų diagramos – iliustruoja, kaip kinta buferių būseną nuo vartotojo paspaustų ar atleidžiamų klavišų.
- Klasių diagramos – iliustruoja vartotojo darbo su klaviatūra sekimo bei atvaizdavimo struktūras.
- Komponentų diagrama – iliustruoja, kaip sistemos komponentai sąveikauja tarpusavyje ir su išoriniais komponentais.

3.3 Architektūros tikslai ir apribojimai

1. Programa yra skirta konkrečiam vartotojui. Jeigu kompiuteriu naudosis du ar daugiau asmenų sistema nerodys teisingų duomenų.
2. Skirta plačiausiai naudojamoms x86 operacinėms sistemoms. Windows 7 (reikia Microsoft .NET Framework 4.5.2 arba SP1), Windows 8, Windows 10.
3. Naudojamas .NET Framework 4.5.2.

3.4 Sistemos dinaminis vaizdas

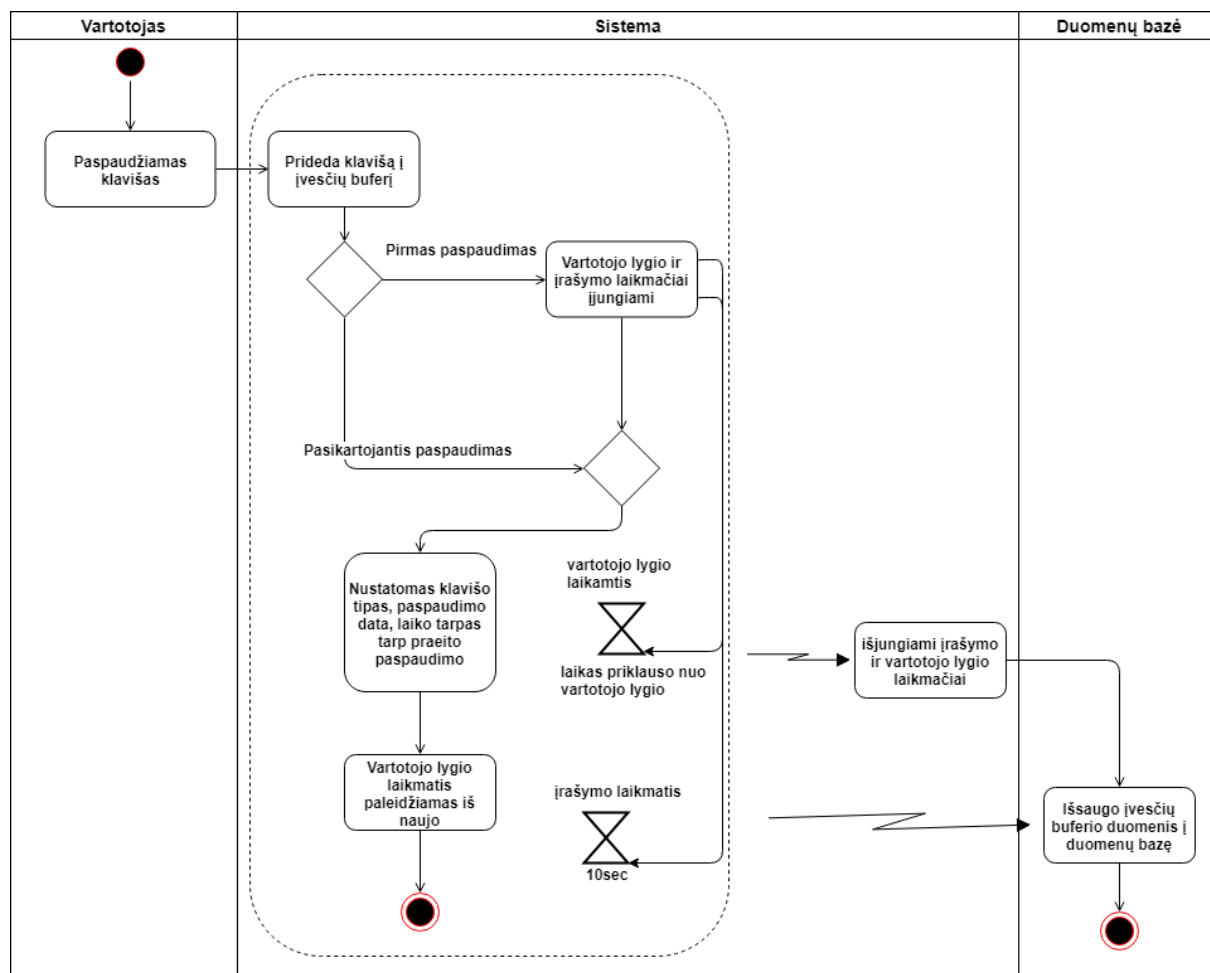
Šio punkto diagramose yra vaizduojama skirtingų tipų buferiai ir laikmačiai.

- Klaidų buferis – skirtas saugoti vartotojo šiuo metu daromai gramatiniai klaidai. Svarbiausi saugomi duomenys yra ištrinti simboliai, pakeisti simboliai bei galutinis žodis.
- Įvesčių – skirtas sekti įvestus klavišus. Iš šio buferio duomenų vėliau yra daromos linijiniai grafikai ir karščio žemėlapiai.
- Sekimo – skirtas sekti vartotojo dabartinę kursoriaus poziciją ir šiuo metu vartotojui matomą tekstą. Jei vartotojas ištrina simbolį tas simbolis yra ištrinamas ir iš sekimo buferio.
- Vartotojo lygio laikmatis – šis laikmatis yra pritaikytas kiekvienam vartotojui individualiai pagal jo atliktą kalibravimo testą.

- Įvesčių laikmatis – šis laikmatis visada skaičiuoja iki 10 sekundžių. Jis yra skirtas buferių įrašymui ir išvalymui nustatytu laiko tarpu. Tai yra daroma dėl našumo, kad atmintyje neužsiliktų visos dienos duomenys, kurie vėliau vienu metu būtų rašomi į duomenų bazę.

3. 4. 1 Klaviatūros klavišų paspaudimo fiksavimas

Nesvarbu kada, kokioje stadijoje ir kokį klavišą vartotojas paspaudė, klavišas visada bus užfiksuotas įvesčių buferyje, išskyrus atvejus, kai klavišas buvo paspaustas programoje, kurioje vartotojas išjungė sekimą. Jeigu paspaudimas yra pirmas (paspaudimas yra skaitomas pirmas, kai vartotojo lygio laikmatis yra išjungtas), sistema įjungia vartotojo lygio ir įrašymo laikmačius. Kitu atveju sistema iškarto nustato klavišo tipą, paspaudimo datą ir laiko tarpą tarp praeito klavišo paspaudimo (šis laikas yra matuojamas naudojant vartotojo lygio laikmatį). Tada klavišas yra pridodamas į įvesčių buferį ir vartotojo lygio laikmatis yra paleidžiamas iš naujo. Kai vienas iš laikmačių pasiekia savo ribą, buferio duomenys yra išsaugomi į duomenų bazę. Jeigu vartotojo lygio laikmatis pasiekia savo ribą, abu laikmačiai yra išjungiami ir sekantis vartotojo paspaudimas bus traktuojamas kaip pirmas paspaudimas. (žr. 5 pav.).

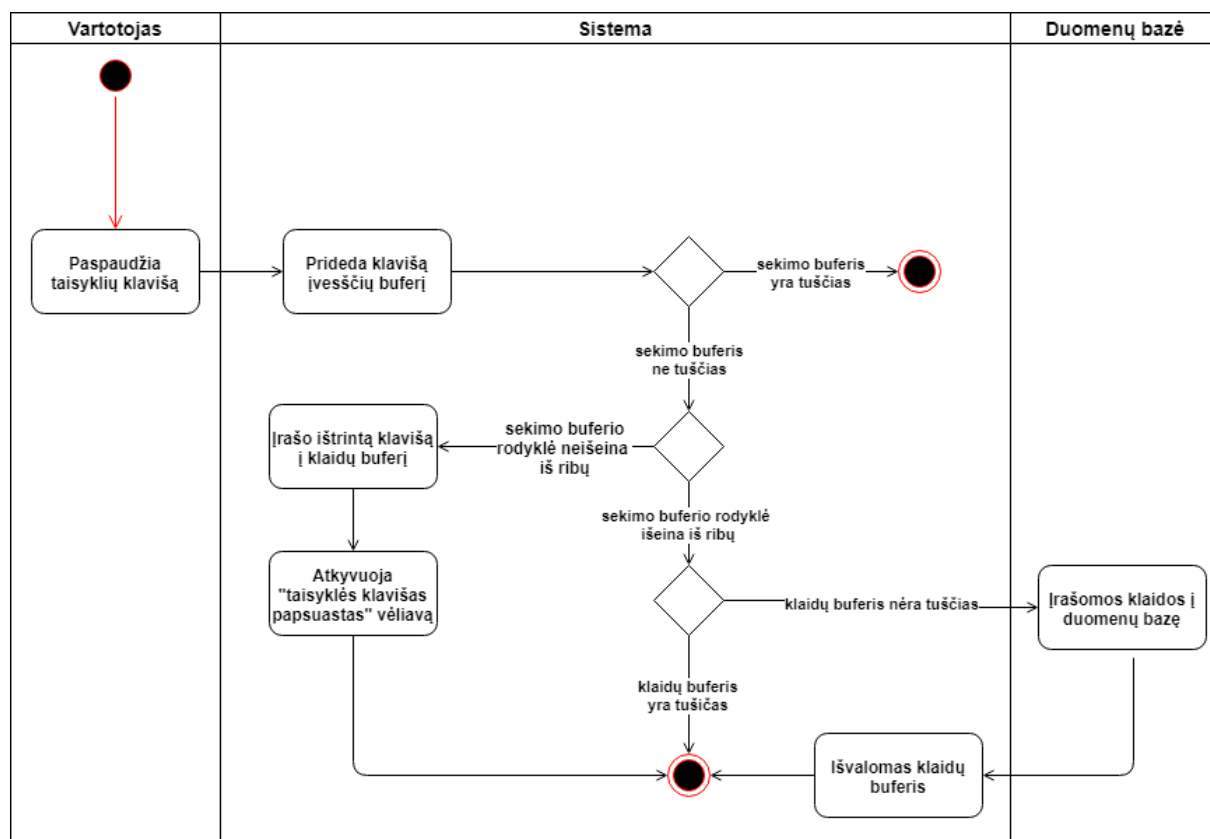


5 pav. Klavišo paspaudimo veiklos diagrama

3. 4. 2 Taisyklių klavišo paspaudimo fiksavimas

Paspaustas mygtukas yra įrašomas į įvesčių buferį. Jeigu sekimo buferis yra tuščias, toliau jokie veiksmai nėra vykdomi, kadangi iš šio buferio yra renkami vartotojo paspaudimai. Kitu atveju

yra patikrinama, ar sekimo buferio rodyklė neišeina iš ribų, jeigu ne – ištrintas klavišas yra įrašomas į klaidų buferį ir aktyvuojama taisyklės „klavišas paspaustas“ vėliava. Jeigu sekimo buferio rodyklė išeina iš ribų ir klaidų buferis nėra tuščias, klaidos yra įrašomos į duomenų bazę ir buferis išvalomas. Jei klaidų buferis yra tuščias, jokie veiksmai toliau nėra vykdomi (žr. 6 pav.).

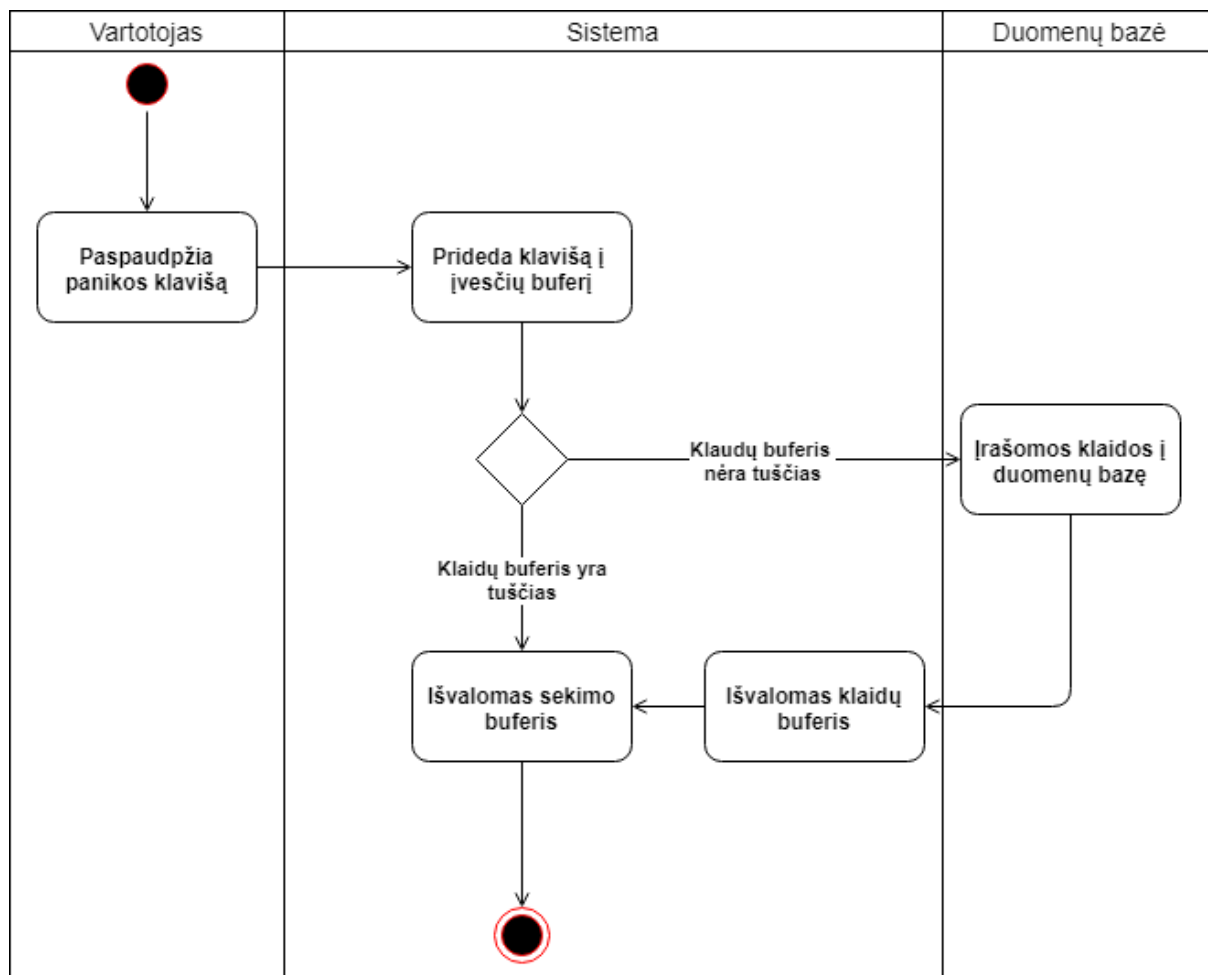


6 pav. Taisyklių klavišo paspaudimo veiklos diagrama

3. 4. 3 Panikos klavišo paspaudimo fiksavimas

Paspaudus panikos klavišą užfiksuotos klaidos yra įrašomos į duomenų bazę (jei tokių yra) ir yra išvalomas sekimo buferis, taip vartotojo klavišų sekimas prasideda iš naujo, kadangi po panikos mygtuko paspaudimo nebegalima atsekti, kurioje vietoje dabar yra kursoriai. (žr. 7 pav.).

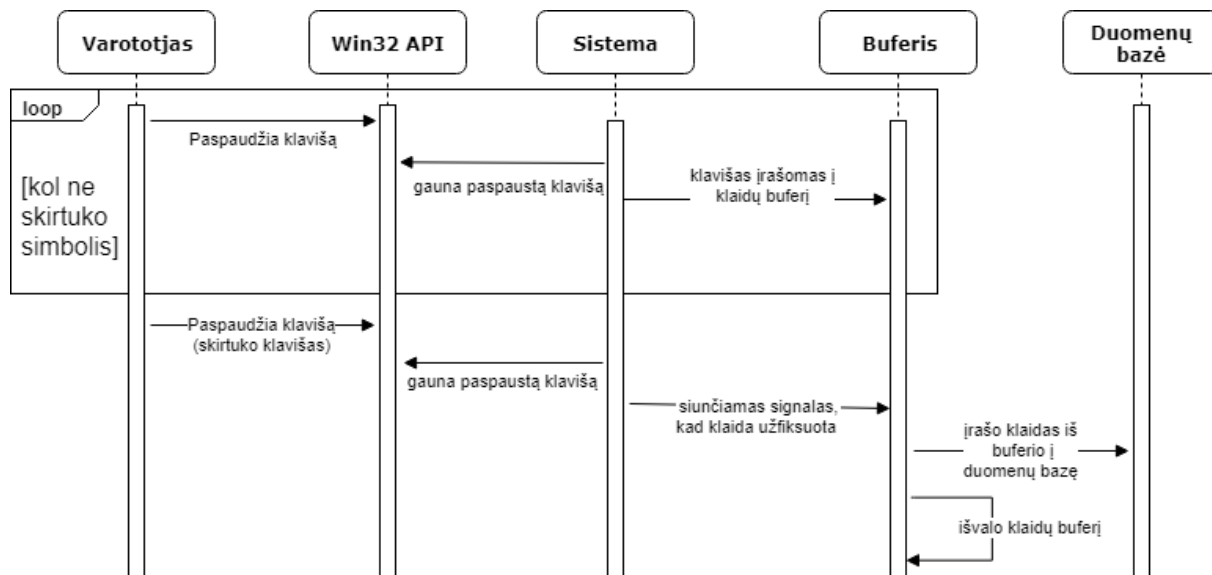
Taip pat kaip panikos klavišo paspaudimas yra skaitomas pelės kairėjo ir dešiniojo mygtuko paspaudimai. Galima daryti prielaidą, kad po jų paspaudimo kursoriai buvo perkeltas į kitą vietą.



7 pav. Panikos klavišo paspaudimo veiklos diagrama

3. 4. 4 Tipinio vartotojo atliekamo veiksmo fiksavimas

Diagrama parodo, kaip sistema elgsis esant vienai iš dažniausių pasitaikančių situacijų – klaidų taisymui. Diagramoje (žr. 8 pav.) yra rodoma situacija, kai vartotojas baigia taisyti žodį, parašo paskutinę raidę ir padeda simbolį, kuris reiškia žodžio pabaigą. Paspaudus klavišą yra patikrinama, ar žodis nėra baigtas redaguoti, tai atliekama tikrinant ar paspaustas klavišas nebuvo skirtuko simbolis t.y. tarpelis, klaustukas, šauktukas, kabutės ir pan.. Jeigu žodis dar nėra baigtas redaguoti, vartotojo įvestas simbolis yra įrašomas į klaidų buferį, jeigu žodis buvo baigtas redaguoti, duomenys iš klaidų buferio yra įrašomi į duomenų bazę ir buferis išvalomas.



8 pav. Tipinio vartotojo veiksmo sekų diagrama

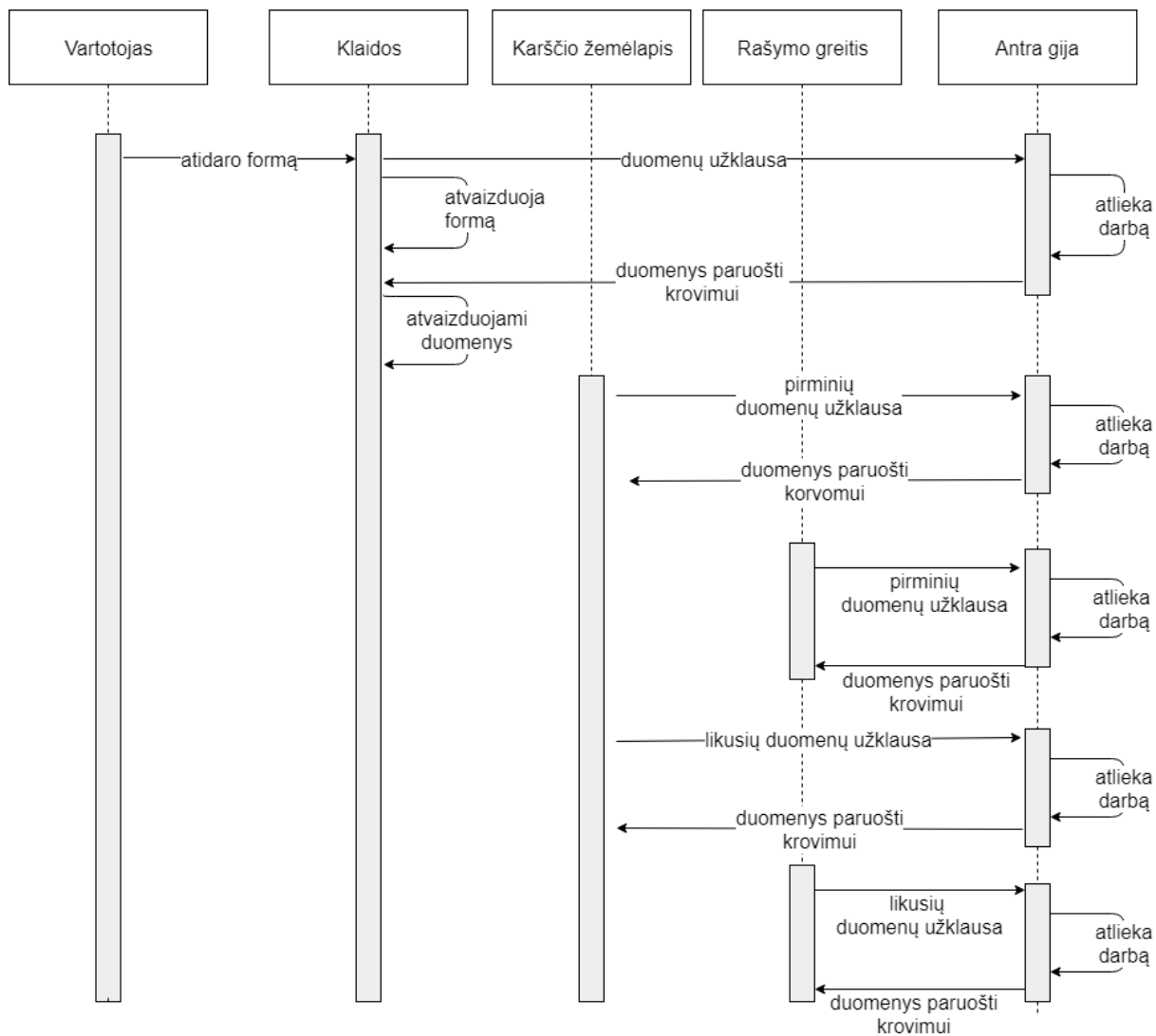
3. 4. 5 Statistikos atvaizdavimas

Didėjant duomenų kiekiui sistemoje analizės trukmė užtruks vis ilgiau. Todėl duomenų analizavimui bei atvaizdavimui buvo panaudota lygiagretusis programavimas.

Vartotojui atidarius pirmą statistikos langą iš karto yra užduodama užklausa, kuri užkrauna ir atvaizduoja pačius pirmus duomenis, kuriuos vartotojas matys. Pirmi atvaizduojami duomenys yra parinkti tokie, kuriuos lengva apskaičiuoti net esant dideliui duomenų kiekiui. Iš karto po to yra analizuojami kitų formų pirminiai duomenys (t.y. patys pirmi duomenys, kuriuos paspaudęs formą pamatys vartotojas). Kai visų formų pirminiai duomenys yra paruošti, baigiama analizuoti likusieji duomenys (žr. 9 pav.).

Tokiu būdu vartotojui jau atidarius pirmą langą atrodo, kad visi duomenys jau yra užkrauti. Kadangi prima yra užkraunami pirminiai duomenys, vartotojui atidarius kitą langą jaus bus rodoma statistiką, o kol vartotojas pasirinks kokio tipo statistiką jis nori matyti, atvaizdavimui bus paruošti ir likę duomenys.

Kadangi užkrauti kai kuriuos duomenis gali užtrukti ilgiau kaip 0,5 s, norint išlaikyti PĮ interaktyvumą skaičiavimai yra iškeliami į antrą giją. Tokios operacijos yra vadinamos „priklausančios nuo procesoriaus“ (angl. CPU-bound) operacijos [16].

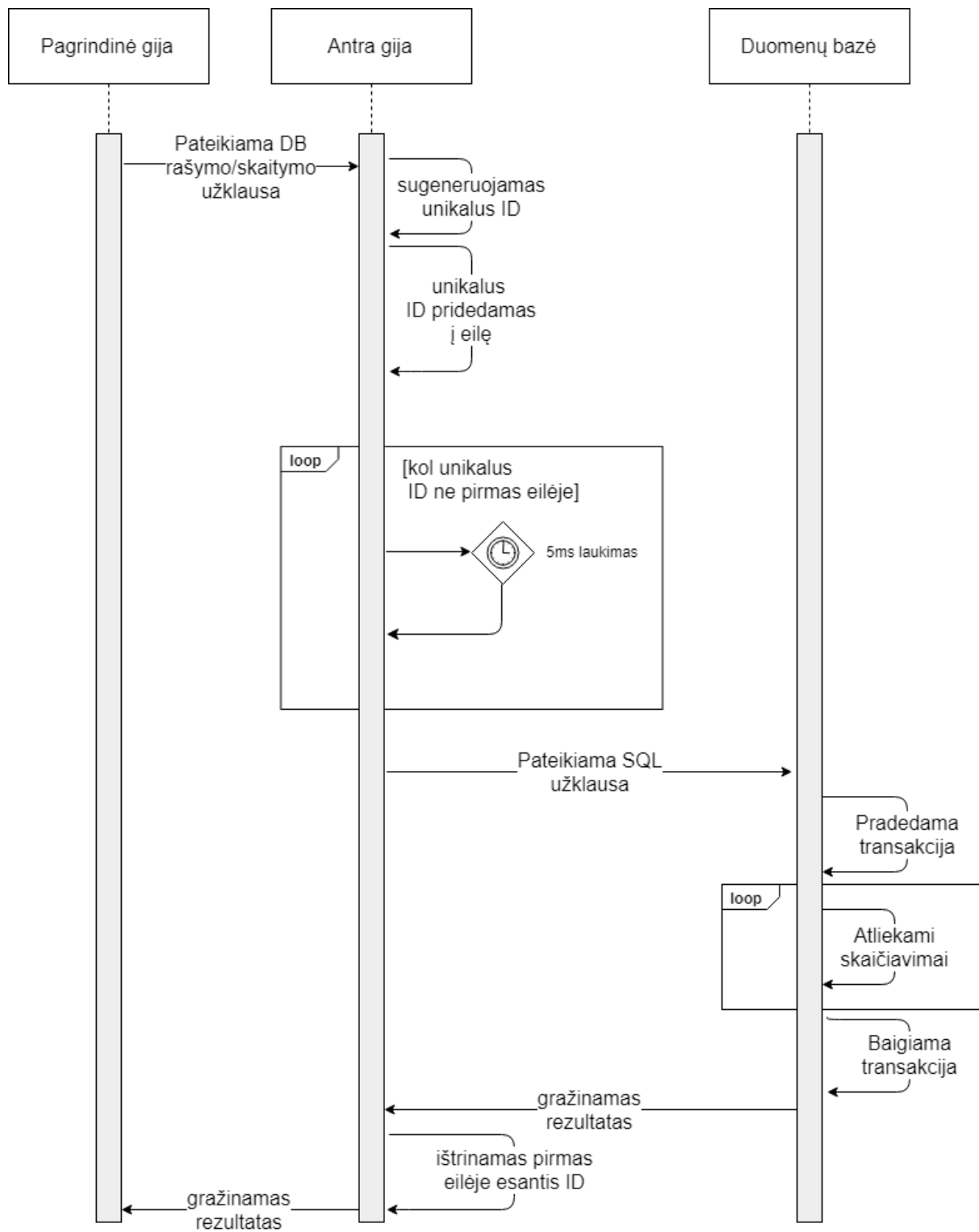


9 pav. Statistikos atvaizdavimo sekų diagrama

3. 4. 6 Skaitymo ir rašymo operacijos duomenų bazėje

SQLite duomenų bazė turi vieną didelį trūkumą, vienu metu iš tos pačios lentelės gali skaityti ar rašyti tik vienas klientas [17]. Kadangi PĮ naudojamas lygiagretusis programavimas rašymui ir skaitymui yra didelė tikimybė, vienu metu rašyti ar skaityti bandys daugiau kaip vienas procesas. Kadangi kiekvienas procesas prisijungia kaip naujas klientas, yra didelė tikimybė, kad būtų bandoma atlikti kelias duomenų bazės operacijas vienu metu. Ši problema buvo išspręsta vykdant procesų užklausas po vieną.

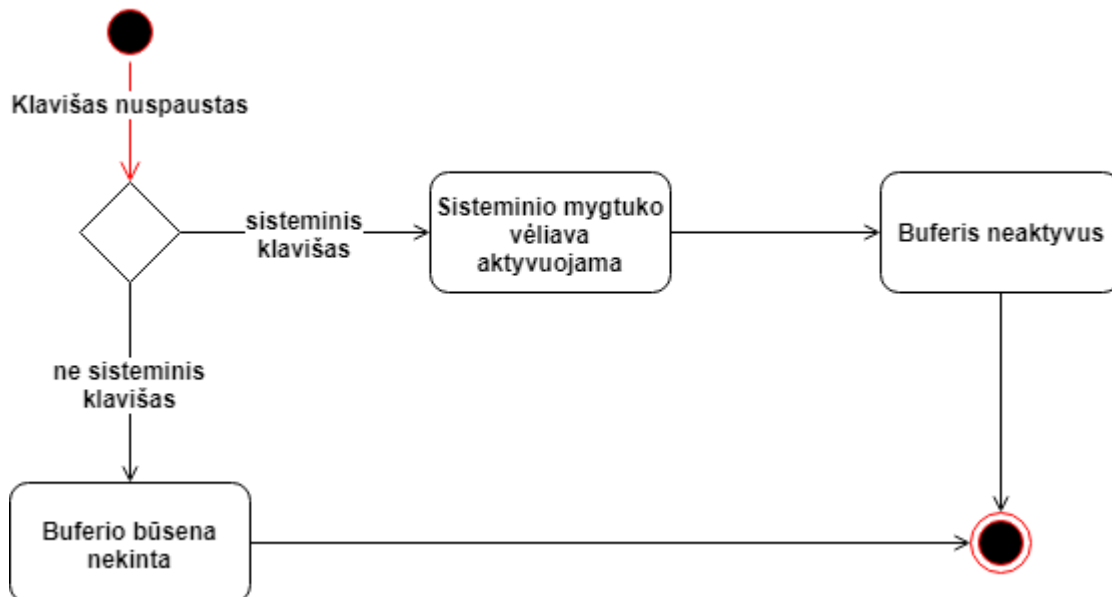
Kiekvienam procesui prieš kreipimąsi į duomenų bazę yra sugeneruojamas unikalūs identifikacijos kodas. Šis kodas yra pridamas į eilę. Tada procesas tikrina, ar jo unikalūs identifikacijos kodas yra pirmas eilėje, jeigu ne, tikrinimas vėl vyksta po 5 ms, jeigu taip, į duomenų bazę yra pateikiamos užklausos. Prieš atliekant bet kokius skaičiavimus yra pradedama transakcija, taip visos užklausos bus vykdomas kaip viena, o ne kaip atskiros transakcijos. Tokiu būdu yra žymiai pagerinamas našumas, pvz.: 50 užklausų įrašymo laikas sumažėjo iš 4 s į 0,03 s. Baigus skaičiavimus yra uždaroma transakcija, gražinami rezultatai bei ištrinamas proceso unikalūs identifikavimo kodas iš eilės. Taip sekantis procesas eilėje žino, kada gali pradėti operacijas su duomenų baze (žr. 10 pav.).



10 pav. Skaitymo ir rašymo operacijų duomenų bazės sekų diagrama

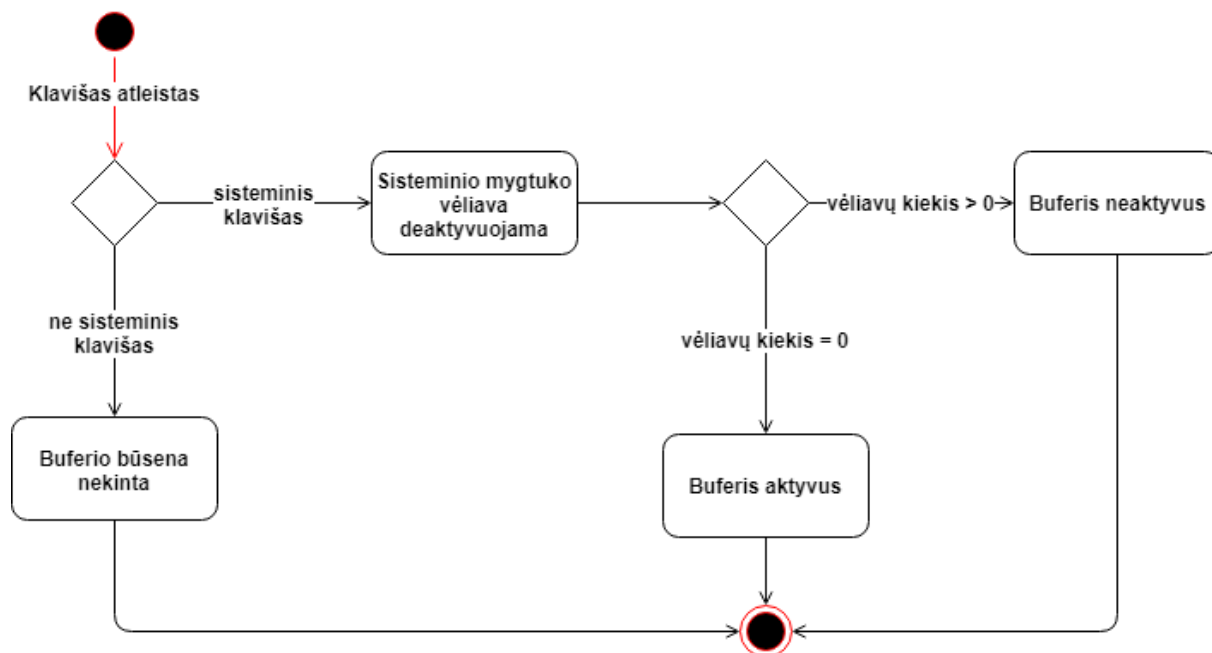
3. 4. 7 Sekimo buferių būsenos kitimas

Diagrama vaizduoja, kaip kinta sekimo buferio būsena nuspaudus klavišą. Jeigu yra nuspaustas sisteminis mygtukas, yra aktyvuojama vėliava ir automatiškai buferis palieka neaktyvų (žr. 11 pav.).



11 pav. Sekimo buferio būseną nuspaudus klavišą būsenų diagrama

Diagrama vaizduoja, kaip keičiasi buferio būseną, kai klavišas yra atleidžiamas. Jei atleistas buvo sisteminis klavišas, yra patikrinama, ar liko dar aktyvuotų vėliavų, jei ne – buferis aktyvuojamas, tačiau jei vis dar yra aktyvuotų vėliavų, vadinasi, yra nuspaustas kurs nors kitas sisteminis mygtukas, todėl buferis išlieka neaktyvus. Ne sisteminio klavišo paspaudimo metu buferio būseną nekinta. (žr. 12 pav.).



12 pav. Sekimo buferio būseną atleidus klavišą būsenų diagrama

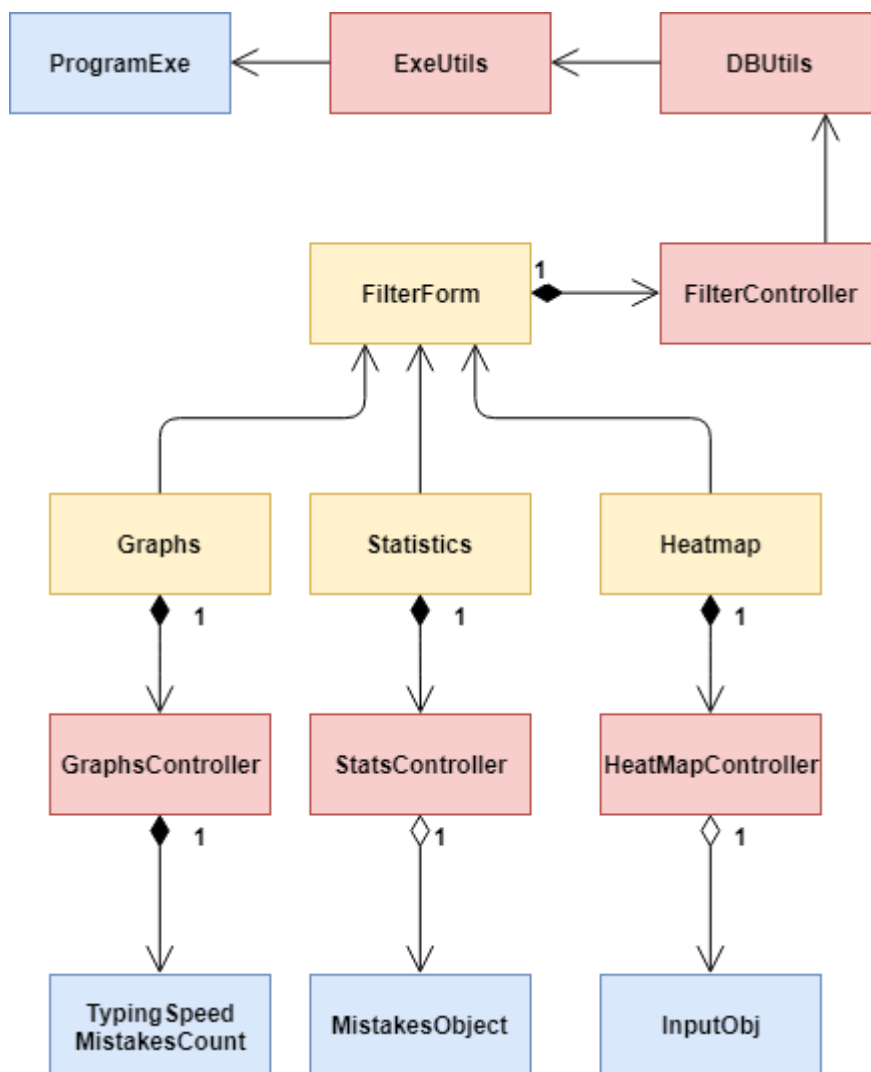
3.5 Sistemos statinis vaizdas

Klasės diagramoje yra pažymėtos spalvos kodais, kad būtų lengviau jas suprasti

- Raudona – klasės, skirtos saugoti, skaityti, kontroliuoti ir manipuluoti duomenis.
- Geltona – tai yra formos, skirtos atvaizduoti duomenis.
- Mėlyna – duomenų struktūros nusakančios objektą.
- Violetinė – klasė, skirta Win32 API funkcijų importavimui.

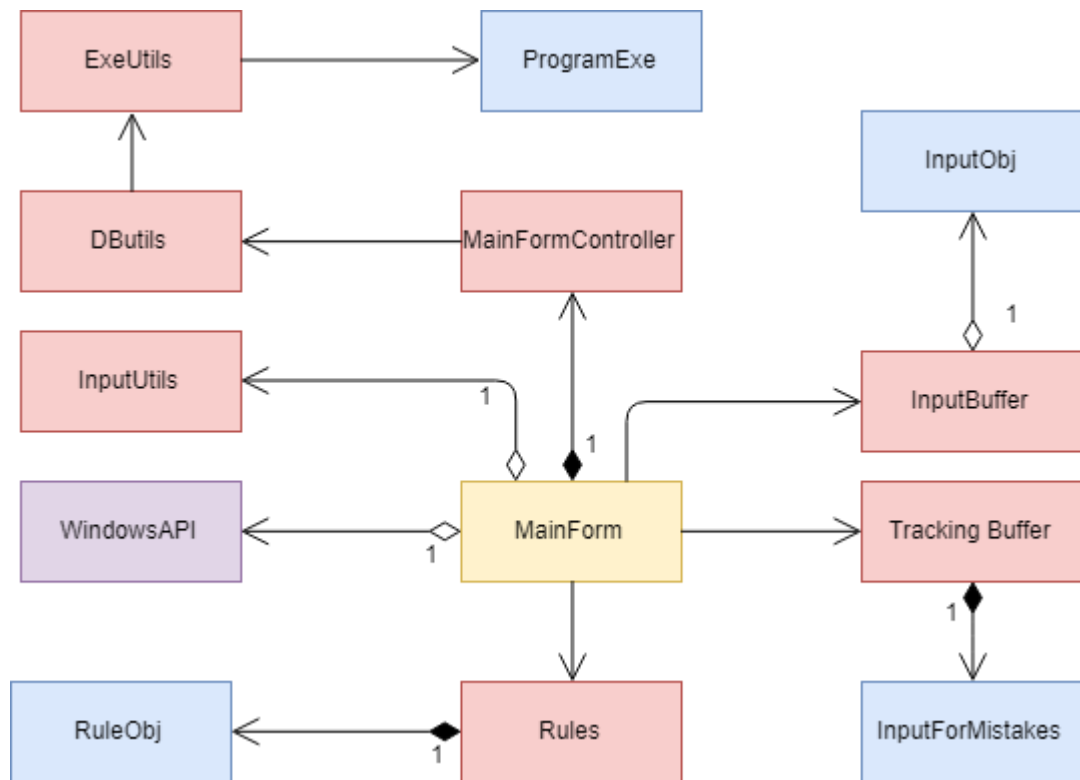
Klasės diagrama vaizduoja kaip veikia duomenų ir statistikos atvaizdavimas. Kiekviena iš duomenų atvaizdavimo formų naudoja po kontrolerį duomenų surinkimui ir manipuliavimui. Kiekvienas kontroleris turi po objektą arba tų objektų sąrašą, kuris saugo duomenis kurie būna atvaizduojami formoje.

Esant norui filtruoti duomenims kiekviena forma, skirta atvaizduoti statistiką gali kreiptis į filtro formą. Filtro kontroleris naudodamasis duomenų bazės klase atnaujiną duomenis pagal nustatytą filtrą kurie buvo atsiųsti iš konkrečios formos (žr. 13 pav.).



13 pav. Sistemos duomenų ir statistikos atvaizdavimo klasių diagrama

Pagrindinė sistemos forma yra atsakinga už vartotojo duomenų sekimą. Naudodama „WindowsAPI“ klasę yra importuojama Win32 API funkcijos skirtos simbolių konvertavimui ir klaviatūros ir programų sekimui. Ši ir „HookNative“ klasės yra atsakingos už klavišų paspaudimų sekimą. Forma naudoja „InputBuffer“ ir „Tracking buffer“ klases įvesčių ir sekimo ir klaidų buferiams saugoti. „Rules“ klasė padeda klasifikuoti paspaustus taisyklių klavišus, o „InputUtils“ konvertuoti paspaustus klavišus į simbolius. „ExeUtils“ klasė atsakinga už sekimą, kurioje programoje buvo paspaustas klavišas, o „MainFormController“ yra skirtas duomenų įrašymui į duomenų bazę. (žr. 14 pav.).



14 pav. Sistemos duomenų stebėjimo ir rinkimo klasių diagrama

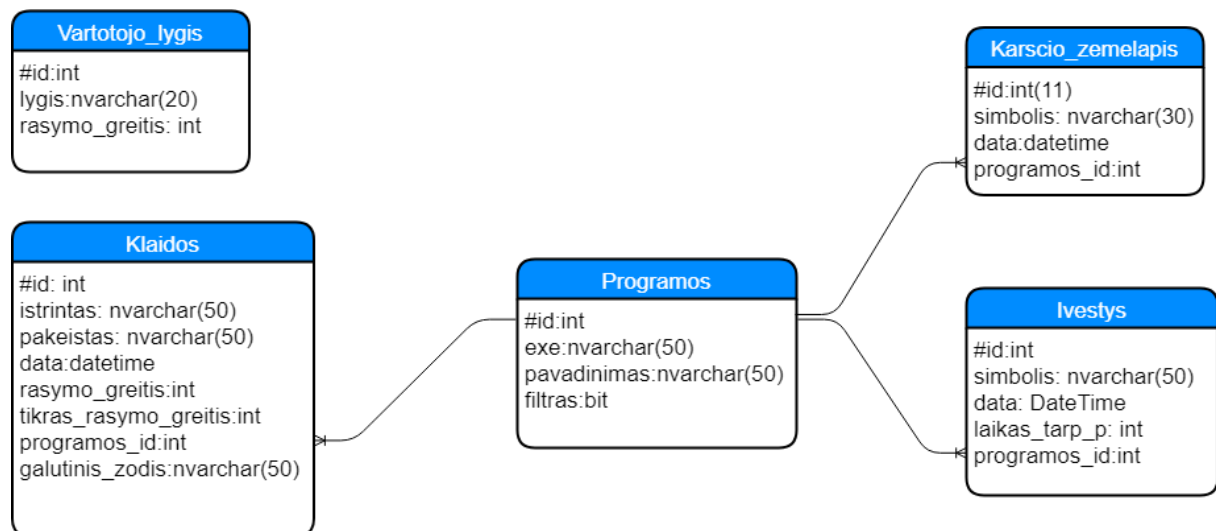
3.5.1 Klasių detalizavimas

Klasių detalizavimas pateikiamas 1 priedas

3.6 Duomenų vaizdas

PĮ naudoja SQLite duomenų bazę. Kadangi sistemoje yra saugomi dideli duomenų kiekiai operacijos su XML ar JSON failais būtų per lėtos (ypač skaitymo operacijos), todėl buvo pasirinkta naudoti duomenų bazę.

Iš duomenų bazės vaizdo (žr. 15 pav.) matosi, kad visos lentelės, skirtos rinkti duomenims, yra susietos per „Programos“ lentelę. Tai yra darome dėl dviejų priežasčių – filtravimo bei duomenų susiejimo su programa. Vartotojui bus patogiau, jeigu programų filtras išsisaugos jo pasirinkimus ir jam nereikės kiekvieną kartą iš naujo rinkti programas, iš kurių jis nori ar nenori matyti statistiką.



15 pav. Duomenų bazės vaizdas

3. 6. 1 Lentelių detalizavimas

31 lentelė. Lentelės „Programos“ detalizavimas

Pavadinimas	Programos
Aprašymas	Saugoma visos programos kuriose vartotojas kada nors yra įvedęs bent vieną simbolį
Struktūra	Id: int Exe: nvarchar(50) Pavadinimas: nvarchar(50) Filtrai: bit
Apribojimai	Skaityti – sistema, vartotojas Rašyti – sistema Atnaujinti – sistema, vartotojas (tikrai filtras reikšmę)

32 lentelė. Lentelės „Vartotojo_lygis“ detalizavimas

Pavadinimas	Vartotojo_lygis
Aprašymas	Saugo vartotojo nustatytą lygį
Struktūra	Id:int Lygis: nvarchar(20) Rasyto_greitis: int
Apribojimai	Skaityti – sistema Rašyti – sistema Atnaujinti – sistema

33 lentelė. Lentelės „Ivestys“ detalizavimas

Pavadinimas	Ivestys
Aprašymas	Saugoja visus vartotojo įvestus simbolius, paspaudimo laiką tarp jų ir datą
Struktūra	Id: int Simbolis: nvarchar(50) Data: DateTime Laikas_tarp_p: int Program_id: int
Apribojimai	Skaityti – sistema Rašyti – sistema Atnaujinti – sistema
Sąsaja	Programos_id > Programos:id

34 lentelė. Lentelės „Klaidos“ detalizavimas

Pavadinimas	Klaidos
Aprašymas	Saugo vartotojo ištrintus ir pakeistus kitais simbolius ir žodžius
Struktūra	Id: int Strintas: nvarchar(50) Pakeistas: nvarchar(50) Data: datetime Rasyto_greitis: int Tikras_rasyto_greitis: int Programos_id: int Galutinis_zodis: nvarchar(50)
Apribojimai	Skaityti – sistema Rašyti – sistema Atnaujinti – sistema
Sąsaja	Programos_id > Programos:id

35 lentelė. Lentelės „Karščio_zemelapis“ detalizavimas

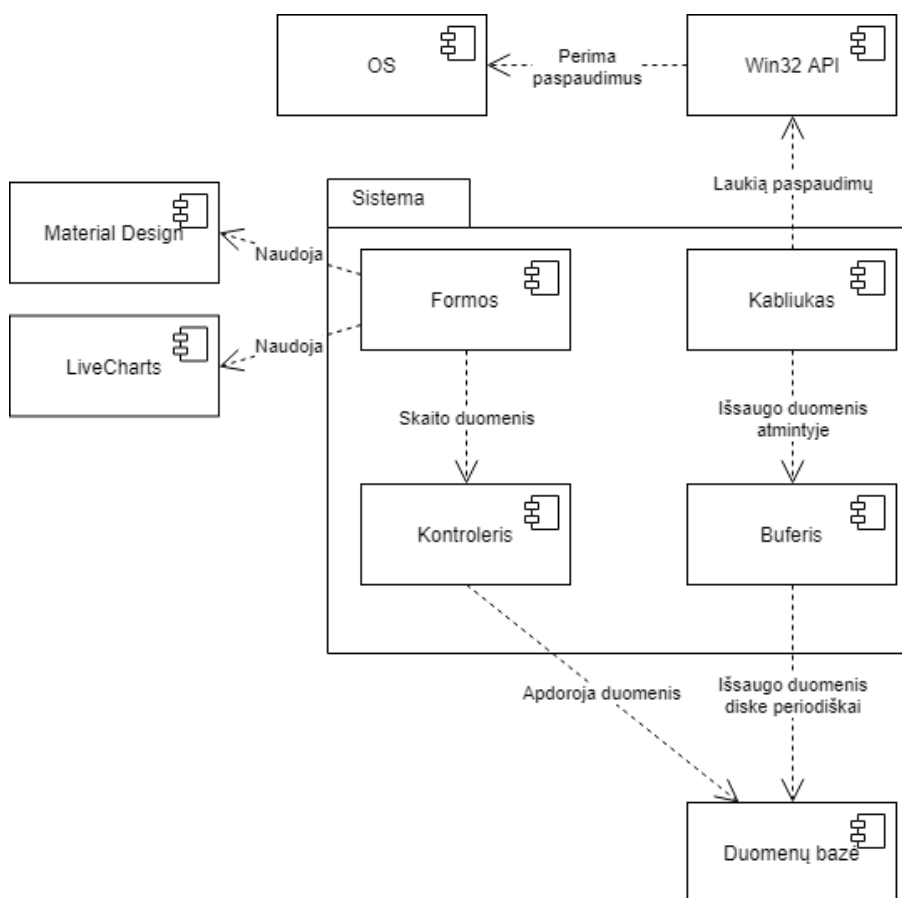
Pavadinimas	Karščio_zemelapis
Aprašymas	Saugo visus paspaustus klavišus

Struktūra	Id: int Simbolis: nvarchar(30) Data: datetime Programos_id: int
Apribojimai	Skaityti – sistema Rašyti – sistema Atnaujinti – sistema
Sąsaja	Programos_id > Programos:id

Vartotojas tiesiogiai negali peržiūrėti jokių lentelių, išskyrus lentelę „Programos“. Duomenys vartotojui iš kitų lentelių yra pateikiami per virtualias lenteles (angl. views).

3.7 PĮ komponentai

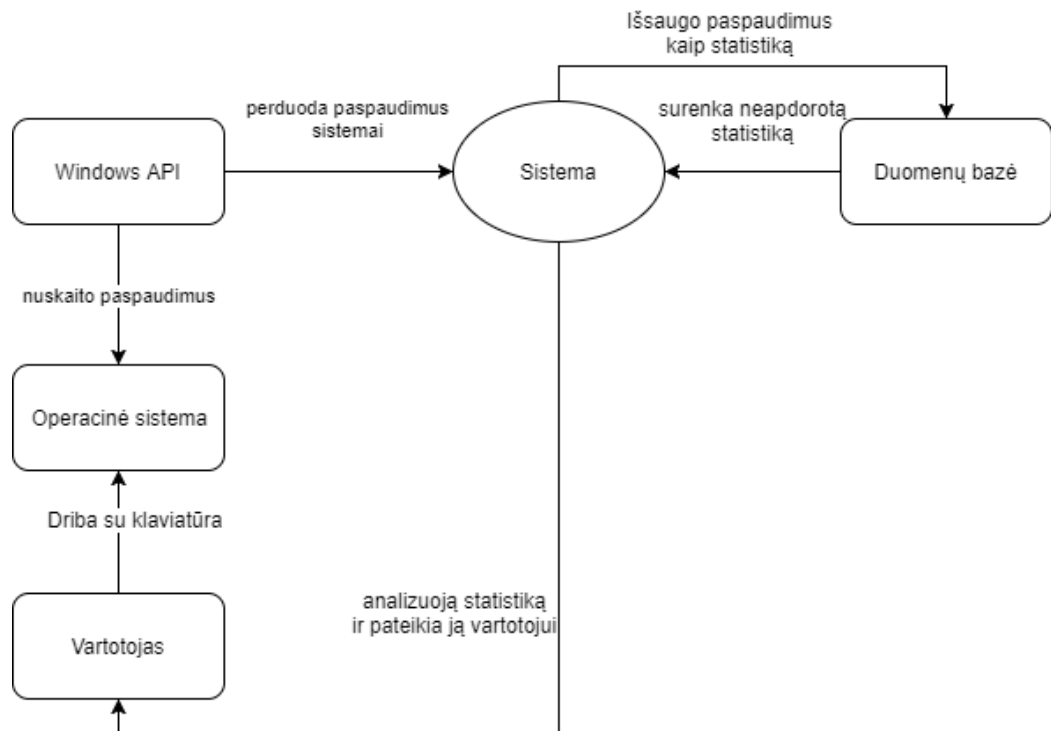
Viskas prasideda operacinės sistemos lygmenyje. Kai OS užfiksuoja paspaustą klavišą, jį iškart perima Win32 API. Kuriamoje sistemoje yra kabliukas, kuris laukia, kol Win32 API perrims klavišo paspaudimą. Sistema perrėmusi klavišo paspaudimą su kabliuku apdoroja klavišą ir padeda į buferį, iš kurio periodiškai yra saugoma į duomenų bazę. Duomenys nėra saugoma tiesiai į duomenų bazę dėl našumo, kadangi būtų atliekama labai daug mažų rašymo operacijų diske. Vėliau vartotojui panorėjus ką nors peržiūrėti duomenys iš duomenų bazės yra nuskaitomi ir apdorojami kontrolerio. Šiuos duomenis iš kontrolerio naudoja formos statistikos atvaizdavimui. Formos naudoja „Material Design“ gražesnei vartotojo sąsajai ir „LiveCharts“ linijinių diagramų atvaizdavimui. (žr. 16 pav.).



16 pav. Sistemos komponentų tarpusavio sąveika

3.8 Duomenų srautai PĮ

Vartotojas dirba su klaviatūra operacinėje sistemoje. Win32 API perima vartotojo paspaudimus ir juos perduoda sistemai. Sistema juos išsaugo į duomenų bazę. Atėjus laikui tie duomenys yra paimami ir išanalizuojami bei pateikiami vartotojui (žr. 17 pav.).



17 pav. Aukščiausio lygio sistemos duomenų srautų diagrama

3.9 PĮ kokybė

- PĮ yra pritaikyta plečiamumui. kadangi sistema naudoja „Windows Forms“, plėsti sistemą yra gana nesudėtinga. Duomenų atvaizdavimo struktūra yra parašyta taip, kad po duomenų manipuliacijos vaizdai būtų atnaujinami su ta pačia funkcija arba su nedideliais jos pakeitimais. Todėl papildyti įvairios statistikos funkcijas yra gana lengva. Taip pat filtrai su papildomomis statistikos atvaizdavimo funkcijomis veiks automatiškai.
- PĮ našumas yra optimizuotas. Į duomenų bazę duomenys yra įrašomi tik tada, kai reikia, kad nevyktų pastovių rašymo operacijų. Tačiau niekada nebus tokios situacijos, kad duomenys nebuvo įrašyti į duomenų bazę ilgiau kaip 10 s, tai užtikrina, kad operatyvioji atmintis niekada nebus perpildyta. Sunkūs skaičiavimai yra iškeliami į atskiras gijas, kad PĮ išliktu aktyvi net vykstant skaičiavimams.

4. TESTAVIMAS

4.1 Testavimo paskirtis

Testavimas yra skirtas aprašyti PĮ testavimo atvejus. Rastus netikslumus ir kritines klaidas, taip pat programinės įrangos įtaką vartotojo sistemai

4.1.1 Testavimo tikslai

- Stabilumas – patikrinti ar sistema yra stabili galutiniam darbui su vartotojui, ar neišsijunginės, nepradės rodyti klaidų pranešimų ar kitaip trukdyti vartotojo darbą.
- Duomenų tikrumas – patikrinti, ar visi duomenys įrašomi į duomenų bazę yra tie, kuriuos vedė bei taisė vartotojas.
- Našumas – nustatyti, kaip programa veikia esant didelėms apkrovoms, ar ji pati nesukelia bereikalingos apkrovos vartotojo sistemai.
- Įsitikinti, kad sistema veikia tvarkingai ne tik su angliskais ir lietuviškais klaviatūrų žodynais bei išdėstymais.
- Įsitikinti, kad PĮ atvaizduoja teisingą informaciją.
- Nustatyti, ar funkcijos grąžina teisingus rezultatus.

4.1.2 Testavimo resursai

- Kompiuteris: i5 4200M, 8GB RAM, GT 750M
- Operacinė sistema: Windows 10 Build 16299
- Microsoft .NET Framework 4.7
- Visual Studio Community edition 2017 Version 15.1
- AIDA 64 Extreme 5.95.4500
- MSI Afterburner 4.4.0

4.1.3 Pagrindiniai apribojimai

- Windows 7 ir naujesnės operacinės sistemos
- Microsoft .NET Framework 4.5.2 ir naujesnės versijos
- Intel Ivy Bridge arba AMD Jaguar arba naujesnių mikroarchitektūrų procesoriai
- 5400 RPM HDD
- 4GB DDR3 1333Mhz atmintis arba greitesnė

4.2 PĮ funkcijų testavimas

36 lentelė. Žodžių ir simbolių rašymo į duomenų bazę testavimas

Testo tikslas:		Patikrinti ar PĮ įveda teisingus duomenys į duomenų bazę			
Testo ID	Reikalavimai/tikslai	Įvykis/įvestis	Laukiamas rezultatas	Gautas rezultatas	T/N
1.	Tuščias laukas	Bandoma trinti tuščią lauką	Klaida nebuvo užfiksuota	Klaida nebuvo užfiksuota	T
2.	Ilgas žodis (50+ simbolių)	Laaaaaaaaaaaaa aaaaaaaaaaaaaa aaaaaaaaaaaaaa	Žodis nebuvo įrašytas į duomenų bazę kaip klaida	Žodis nebuvo įrašytas į duomenų bazę	T

		aaaaaaaaabab		kaip klaida	
3.		Laaaaaaaaa aaaaaa'aaaaaa aaaaa';aaaa+a aaaaa@aaabažs	Žodis nebuvo įrašytas į duomenų bazę kaip klaida	Žodis nebuvo įrašytas į duomenų bazę kaip klaida	T
4.	Trumpas žodis ar simbolis	Labas	Labas	Labas	T
5.		l	l	l	T
6.		labas	labas	labas	T
7.		'labas:--	'labas:--	'labas:--	T

37 lentelė. Dviejų klaviatūrų suderinamumo testavimas

Testo tikslas:		Patikrinti kaip PĮ reaguoja kai vartotojas naudoja dvi klaviatūras iš karto				
Testo ID	Reikalavimai/tikslai	Įvykis/įvestis	Laukiamas rezultatas	Gautas rezultatas	T/N	Pastabos
8.	Vedami vienodi žodžiai su viena ir kita klaviatūra Kb1: labas	Abiejose klaviatūrose aktyvuotas „CapsLock“	labas labas	labas labas	T	
9.	Kb2: labas	Vienoje klaviatūroje laikomas „Shift“ kitoje rašomas žodis	LABAS	LABAS	T	
10.		Vienoje klaviatūroje nuspaustas „CTRL“, kitoje rašomas žodis	Rašymas nevyksta	Rašymas nevyksta	T	
11.	vienoje klaviatūroje nuspausta karinė rodyklė kitoje dešininė rodyklė	Qwertyuiopasdfghjkl zxcv bnm (rodyklė:g) Kb1:dešininė rodyklė Kb2: kairinė rodyklė	qwertyuiopasdfghj klzxcvbnm (rodyklė:w)	qwertyuiopasdfg hjlzxcvbnm(ro dyklė:w)	T	Iš kiekvienos klaviatūros duomenys neateina vienodais intervalais tempu, todėl kursorius po mažų slenkasi į dominuojančios klaviatūros nuspausto klavišo pusę
12.		Qwertyuiopasdfghjkl zxcv bnm (rodyklė:g) Kb1:kairinė rodyklė Kb2:dešininė rodyklė	qwertyuiopasdfghj klzxcvbnm (rodyklė:n)	qwertyuiopasdfg hjlzxcvbnm (rodyklė:n)	T	
13.	laikomi du klavišai nuspausti vienu metu	Kb1:h Kb2:l	hlhlhlhlhlhlhl	hlhlhlhlhlhlhl	T	
14.	matuojamas laiko tarpas tarp vienu metu paspaustų klavišų	Kb1:q Kb2:z	Laiko tarpas tarp paspaudimų (0-2 ms)	Stabiliai pavyksta gauti 1ms	T	Dėl to, kad testas vykdomas iš vartotojo pusės galime tikėtis ~2 ms paklaidos

38 lentelė. Simbolių įvesčių į duomenų bazę testavimas

Testo tikslas:		Patikrinti ar PĮ teisingai įveda įvairius simbolius į duomenų bazę			
Testo ID	Reikalavimai/tikslai	Įvykis/įvestis	Laukiamas rezultatas	Gautas rezultatas	T/N
16.	Įvedami simboliai	<>?:{} ,./;[]\	<>?:{} ,./;[]\	<>?:{} ,./;[]\	T
17.		"	"	"	T
18.		„labas“	„labas“	„labas“	T
19.		„'labas“	„'labas“	„'labas“	T
20.		@/	@/	@/	T
21.		““	““	““	T
22.		/	/	/	T
23.		‘	‘	‘	T

39 lentelė. Kursoriaus judėjimo testavimas

Testo tikslas:		Patikrinti ar PĮ teisingai seka vartotojo įvestą tekstą				
Testo ID	Reikalavimai/tikslai	Įvykis/įvestis	Laukiamas rezultatas	Gautas rezultatas	T/N	Pastabos
24.	Rodyklės nustumimas iš ribų į kairę	Įvestis:labas x5 kairinės rodyklė	Indeksas išėjo iš ribų	Indeksas išėjo iš ribų	T	Indeksas išėjo iš ribų
25.	Rodyklės nustumimas iš ribų į dešinę	Įvestis:labas x5 dešinės rodyklės	Indeksas išėjo iš ribų	Indeksas išėjo iš ribų	T	
26.	Iš naujo pradedama rašyti po ištumtos rodyklės	Indeksas išėjęs iš ribų → labas	labas	labas	T	
27.	Laikomas nuspaustas „alt“ ir spaudžiama viršutinė rodyklė	Labas (rodyklė: s)	Indeksas išėjo iš ribų	Indeksas išėjo iš ribų	T	
28.	Įvestas žodis: LABas Rodyklė: B	Spaudžiama kairinė rodyklė	Rodyklė: A	Rodyklė: A	T	
29.		Spaudžiama dešininė rodyklė	Rodyklė: a	Rodyklė: a	T	

40 lentelė. Sekimo buferio ribų testavimas

Testo tikslas:		Patikrinti ar PĮ teisingai reaguoja į klavišų kombinacijas			
Testo ID	Reikalavimai/tikslai	Įvykis/įvestis	Laukiamas rezultatas	Gautas rezultatas	T/N
30.	Sisteminiai klavišai + paprasti	Ctrl + A	Indeksas išėjo iš ribų	Indeksas išėjo iš ribų	T
31.		Alt + W	Indeksas išėjo iš ribų	Indeksas išėjo iš ribų	T
32.		Kairysis Alt + Dešinysis Alt + X	Indeksas išėjo iš ribų	Indeksas išėjo iš ribų	T
33.		Ctrl + Alt + J	Indeksas išėjo iš ribų	Indeksas išėjo iš ribų	T
34.	Klavišų kombinacijos su OS funkcijomis	Alt + Tab	Indeksas išėjo iš ribų	Indeksas išėjo iš ribų	T
35.		Ctrl + Alt + Delete	Indeksas išėjo iš ribų	Indeksas išėjo iš ribų	T
36.		Ctrl + Win + C	Indeksas išėjo iš ribų	Indeksas išėjo iš ribų	T
37.		Win + Tab	Indeksas išėjo iš ribų	Indeksas išėjo iš ribų	T

Testo ID	Reikalavimai/tikslai	Įvykis/įvestis	Laukiamas rezultatas	Gautas rezultatas	T/N	Pastabos
54.	Ar sistema atpažįsta paspaustą klavišą	Key.PageUp	Panic	Panic	T	
55.		Key.A	Normal	Normal	T	
56.		Key.Alt	System	System	T	
57.		Key.Play	NoType	NoType	T	Sistema klavišams kurių nėra standartinėse klaviatūrose priskiria „Bet tipo“ tipą
58.		Key.F19	NoType	NoType	T	
59.		Key.Up	Pointer	Pointer	T	

45 lentelė. Konvertavimo iš klavišo į raidę testavimas

Testo tikslas:		Patikrinti ar PĮ teisingai konvertuoja paspaustą klavišą į raidę			
Testo ID	Reikalavimai/tikslai	Įvykis/įvestis	Laukiamas rezultatas	Gautas rezultatas	T/N
60.	Anglų kalba	Shift+Key.A	A	A	T
61.		Key.Z	z	z	T
62.		Key.Oem1	;	;	T
63.		Shift + Key.Oem1	:	:	T
64.	Lietuvių kalba	Key.D0	ą	ą	T
65.		Shift + Key.D5	Į	Į	T
66.	Rusų kalba	Shift + Key.B	И	И	
67.		Key.B	и	и	T
68.		Key.Z	я	я	T

46 lentelė. Karščio žemėlapių atvaizdavimo testavimas

Testo ID	Reikalavimai/tikslai	Įvykis/įvestis	Laukiamas rezultatas	Gautas rezultatas	T/N	Pastabos
70.	Spalvų atvaizdavimas Kai didžiausia vertė: 5975	Pilka spalva RGB – (200,200,200)	Klavišai su 0 paspaudimų	F10 – 0 PageDown – 0 Num5 -0	T	Iš realios duomenų bazės yra paimami po tris variantus kiekvienam testui
71.		Mažiausia reikšmė, balta spalva RGB – (255,255,255)	Klavišai su 1 - 11 paspaudimų	F1 – 2 \" - 10 ` - 3	T	
72.		Didžiausia reikšmė, raudona spalva RGB – (255,0,0)	Klavišai su 5963 – 5975 paspaudimų	Space - 5975	T	
73.	Paspaudimų kiekio procentų skaičiavimas. Kai iš viso yra paspaudimų: 27440	0 paspaudimų	0%	0%	T	
74.		1 paspaudimas	0%	0%	T	
75.		2 paspaudimai	0.01%	0.01%	T	
76.		543 paspaudimai	1.98%	1.98%	T	

77.		2616 paspaudimai	9.53%	9.53%	T	
-----	--	---------------------	-------	-------	---	--

47 lentelė. Linijinių grafikų atvaizdavimo ir skaičiavimų testavimas

Test o ID	Reikalavimai/tikslai	Įvykis/įvestis	Laukiamas rezultatas	Gautas rezultatas	T/ N	Pastabos
78.	Vidutinis rašymo greitis	Įvesti klavišai tarp kurių paspaudimo laikas 0 ms	Grafikas neatvaizduoja jokių duomenų	Grafikas neatvaizduoja jokių duomenų		
79.		Įvesti įrašai kurių paspaudimo laikas tarp klavišų 200 ms	Vidutinis rašymo greitis: 5 paspaudimai per sekundę	Vidutinis rašymo greitis: 5 paspaudimai per sekundę		
80.		Įvesti įrašai kurių paspaudimo laikas tarp klavišų 5000 ms	Vidutinis rašymo greitis: 0,2 paspaudimai per sekundę	Vidutinis rašymo greitis: 0,2 paspaudimai per sekundę		
81.		Įvesti įrašai kurių paspaudimo laikas tarp klavišų 0ms – 2018ms	Vidutinis rašymo greitis: 4,43 paspaudimai per sekundę	Vidutinis rašymo greitis: 4,43 paspaudimai per sekundę		Duomenys paimti iš duomenų bazės ir apskaičiuoti rankiniu būtu
82.	Laiko užtrunkančio taisyti klaidas skaičiavimas (kai iš viso sugaišta 59089ms)	Klaidų kiekis: 1 Vidutinis sugaištas laikas: 200 ms	0.34%	0.34%		
83.		Klaidų kiekis: 200 Vidutinis sugaištas laikas: 1 ms	0.34%	0.34%		
84.		Klaidų kiekis: 1 Vidutinis sugaištas laikas: 59089 ms	100%	100%		
85.		Klaidų kiekis: 34 Vidutinis sugaištas laikas: 357 ms	20.53%	20.53%		
86.		Klaidų kiekis: 52	5.54%	5.54%		

		Vidutinis sugaištas laikas: 63 ms				
87.		Klaidų kiekis: 51 Vidutinis sugaištas laikas: 25 ms	2.16%	2.16%		
88.	Klaidų kiekio priklausomai nuo rašymo greičio skaičiavimas (kai iš viso yra 460 klaidų)	Klaidų kiekis: 0	Neatvaizduojama s grafike (0%)	Neatvaizduojama s grafike (0%)		Atvaizduojama yra tik tos reikšmės kurios siekia bent 0.5%
89.		Klaidų kiekis: 1	Neatvaizduojama s grafike (0.21%)	Neatvaizduojama s grafike (0.21%)		
90.		Klaidų kiekis: 100	21.73%			
91.		Klaidų kiekis: 34	7.4%	6.4%		
92.		Klaidų kiekis: 80	17.4%	17.4%		

48 lentelė. Klaidų sąrašo atvaizdavimo testavimas

Testo ID	Reikalavimai/tikslai	Įvykis/įvestis	Laukiamas rezultatas	Gautas rezultatas	T/N	Pastabos
93.	Galutinio žodžio fiksavimas kai trinamas žodis	_Labas	Labas	Labas	T	„_“ testuose yra skirtas atvaizduoti tarpelį
94.		Labas_	Labas	Labas	T	
95.		Lab_as	Du žodžiai: Lab ir as	Du žodžiai: Lab ir as	T	
96.		;Labas	Labas	Labas	T	
97.		Labas;	Labas	Labas	T	
98.		„‘Labas][,	Įrašytas tik žodis „Labas“	Įrašytas tik žodis „Labas“	T	

49 lentelė. Statistikos filtravimo testavimas

Testo ID	Reikalavimai/tikslai	Įvykis/įvestis	Laukiamas rezultatas	Gautas rezultatas	T/N	Pastabos
99.	Linijinių grafikų, klaidų sąrašo ir karščio žemėlapių statistikų filtravimas	Paskutinės 24 valandos	Gauti įrašai nuo dabartinės dienos ir valandos -24h	Gauti įrašai nuo dabartinės dienos ir valandos -24h	T	
		Paskutinės 7 dienos	Gauti įrašai nuo dabartinės dienos ir valandos -7d	Gauti įrašai nuo dabartinės dienos ir valandos -7d	T	
		Paskutinės 30 dienų	Gauti įrašai nuo dabartinės dienos ir valandos -30d	Gauti įrašai nuo dabartinės dienos ir valandos -30d	T	
		Šiandienos data	Gauti įrašai nuo šiandienos 00:00:00 val. iki 23:59:59	Gauti įrašai nuo šiandienos 00:00:00 val. iki 23:59:59	T	
		Dviejų dienų intervalas	Gauti įrašai nuo YYYY-mm-01 00:00:00 iki	Gauti įrašai nuo YYYY-mm-01	T	

			YYYY-mm-02 23:59:59	00:00:00 iki YYYY-mm- 02 23:59:59		
		Visos programos nepažymėtos	Gauti visi įrašai duomenų bazėje	Gauti visi įrašai duomenų bazėje	T	
		Nepažymėta „Microsoft Word“	Gauti visi įrašai išskyrus įrašus kurie buvo parašyti „Microsoft Word“	Gauti visi įrašai išskyrus įrašus kurie buvo parašyti „Microsoft Word“	T	
		Pažymimas atgal „Microsoft Word“ po darbo su klaviatūra.	Gauti visi įrašai išskyrus tie, kurie buvo parašyti tuo metu, kai „Microsoft Word“ buvo nepažymėtas	Gauti visi įrašai išskyrus tie, kurie buvo parašyti tuo metu, kai „Microsoft Word“ buvo nepažymėtas	T	Testo paskirtis patikrinti, ar PĮ tikrai nebeseka vartotojo klavišų kai programa yra išfiltruota

50 lentelė. Kalibracijos testavimas

Testo ID	Reikalavimai/tikslai	Įvykis/įvestis	Laukiamas rezultatas	Gautas rezultatas	T/N	Pastabos
100.	Kalibracijos testas	Atsitiktinis tekstas	Kalibracija neužskaityta	Kalibracija neužskaityta	T	
		Rašymo greitis 2-8 s tarp klavišų paspaudimų	Vartotojo lygis: 2-8 s	Vartotojo lygis: 4,52 s	T	
		Paspaudimo laikas tarp klavišų 10 s+	Kalibracija neužskaityta	Kalibracija neužskaityta	T	Maksimalus leidžiamas vartotojo lygis yra 10 s
		Asmuo 1 testo nr.1	Paklaida ~200 ms	870ms	T	
		Asmuo 1 testo nr.2		760ms		
		Asmuo 2 testo nr.1	Paklaida ~200 ms	652ms	T	
		Asmuo 2 testo nr.2		677ms		
101.	Rankinis kalibravimas	Įvedama reikšmė -1	Kalibracija neužskaityta	Kalibracija neužskaityta	T	
		Įvedama reikšmė: 0	Kalibracija neužskaityta	Kalibracija neužskaityta	T	
		Įvedama reikšmė: 199	Kalibracija neužskaityta	Kalibracija neužskaityta	T	
		Įvedama reikšmė: 200	Vartotojo lygis: 0,2 s	Vartotojo lygis: 0,2 s	T	Minimalus vartotojo lygis: 0,2 s
		Įvedama reikšmė: 10000	Vartotojo lygis: 10 s	Vartotojo lygis: 10 s	T	
		Įvedama	Kalibracija	Kalibracija	T	

		reikšmė: 10001	neužskaityta	neužskaityta		
102.	Automatinis kalibravimas	Greitas klavišų spaudymas	Vartotojo lygis: >0,5 s	0,24 s	T	
		Visi klavišų paspaudimai 0ms	Kalibracija neužskaityta	Kalibracija neužskaityta	T	Toks variantas įmanomas tik tada, jei vartotojas kiekvienam klavišui paspaustu užtrinka daugiau kaip 10s
		Asmuo 1 testo nr.1	Paklaida ~250 ms	1404 ms	T	
		Asmuo 1 testo nr.2		1213 ms		
		Asmuo 2 testo nr.1	Paklaida ~200 ms	998 ms	T	
		Asmuo 2 testo nr.2		870 ms		
		Asmuo 3 testo nr.1	Paklaida ~400 ms	2201 ms	N	Didėjant vartotojo lygio laikmačiui atitinkamai turėtų didėti ir paklaida
		Asmuo 3 testo nr.2		2557 ms		

4.3 PĮ charakteristikų testavimas

51 lentelė. Programinės įrangos stabilumo testavimas

Testo tikslas:		Patikrinti ar PĮ yra stabili esant ištęstiniame darbo su klaviatūra metu			
Testo ID	Reikalavimai/tikslai	Įvykis/įvestis	Laukiamas rezultatas	Gautas rezultatas	T/N
103.	Valanda dirbant su klaviatūra ir atliekant specifiкуotą darbą	Programavimas	PĮ neužstringa ir neišsijungia	PĮ neužstringa ir neišsijungia	T
104.		Susirašinėjimas	PĮ neužstringa ir neišsijungia	PĮ neužstringa ir neišsijungia	T
105.		Kompiuterinių žaidimų žaidimas	PĮ neužstringa ir neišsijungia	PĮ neužstringa ir neišsijungia	T
106.	7 dienos tipinio darbo režimo	Dirbama su įvairiomis pertraukomomis, programomis ir kompiuterio užmigdymu	PĮ neužstringa ir neišsijungia	PĮ neužstringa ir neišsijungia	T

4.3.1 Duomenų fiksavimo esant vartotojo kompiuterio apkrovai testavimas

52 lentelė. Paspaudimo laiko esant vartotojo sistemos apkrovai testavimas

Testo tikslas:		Patikrinti kaip PĮ reaguoja esant didėliai resursų apkrovai vartotojo sistemoje				
Naudojamas įrankis:		Aida64 – sukelti kompiuterio komponentų apkrovą				
Testo ID	Reikalavimai/tikslai	Įvykis/įvestis	Laukiamas rezultatas	Gautas rezultatas	T/N	Pastabos
107.	Paspaudimo laikas tarp klavišų esant	100% CPU	28-30 ms	28-29 ms	T	Yra ilgai nuspaustas

	tam tikrai apkrovai:					vienas klaviatūros mygtukas, kadangi taip yra pašalinamas žmogiškasis faktorius
108.		100% FPU	28-30 ms	23-27 ms	T	Gaunamas rezultatas yra dar geresnis. Tikriausiai tai įvyksta dėl padidėjusio CPU taktinio dažnio
109.		100% FPU ir CPU	28-30 ms	27-30 ms	T	
110.		100% RAM	28-30 ms	0-1 ms	N	Rašymas iš vartotojo pusės pasidaro labai lėtas. Tačiau sistema vis tiek mato visas vartotojo įvestis
111.	Laiko tarpas kol	100% CPU	1-100 ms	1-5 ms	T	
112.	duomenys bus įrašyti	100% FPU	1-100 ms	1-5 ms	T	
113.	į duomenų bazę esant tam tikrai apkrovai:	100% CPU ir FPU	1-100 ms	1-5 ms	T	
114.		100% RAM	1-100 ms	3-10 ms	T	

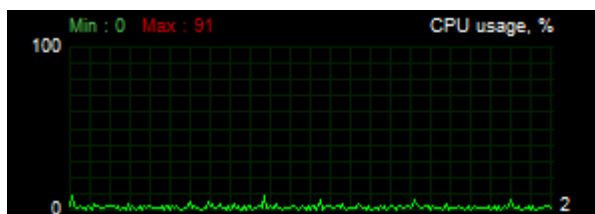
Testo ID: 115.

Testo tikslas: Patikrinti PĮ įtaka vartotojo sistemai

Naudojamas įrankis: MSI Afterburner – CPU aprkovos matavimui

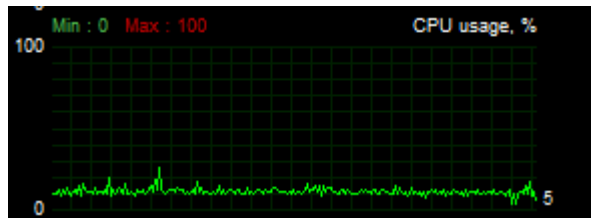
Kadangi vienas iš svarbiausių PĮ apsektų yra nepastebimas veikimas fone, reikia ištestuoti, kokią įtaką PĮ turi vartotojo sistemai. Kiekvienas testas buvo vykdomas penkias minutes.

CPU išnaudojiamumas sistemai esant ramybės būsenoje vidutiniškai svyruoja tarp 1-5% su retais šuoliais iki 10% (žr. 18 pav.).



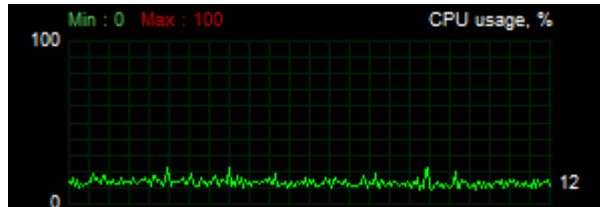
18 pav. CPU išnaudojiamumas ramybės būsenoje

CPU išnaudojiamumas vykstant rašymui, tačiau PĮ esant išjungtai padidėja maždaug iki ~10% su retais šuoliais iki 20% (žr. 19 pav.).



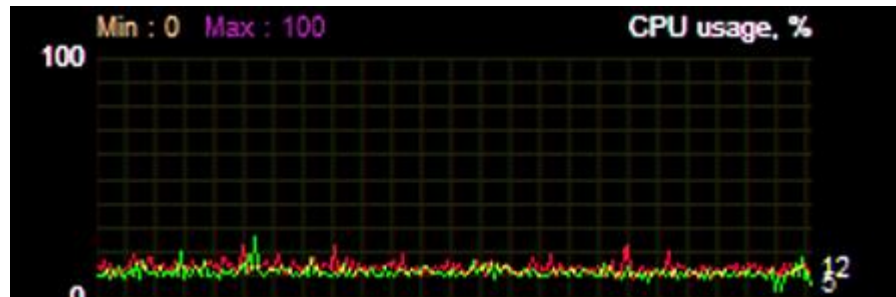
19 pav. CPU išnaudojumas vykstant rašymui

CPU vidutinis išnaudojumas vykstant rašymui ir esant PĮ įjungtai padidėja vidutiniškai tik 2-4% procentais (žr. 20 pav.).



20 pav. CPU išnaudojumas vykstant rašymui su įjungta PĮ

Kad tai matytųsi aiškiau, abi diagramos yra atvaizduotos viena ant kitos (žr. 21 pav.). Raudona spalva vaizduoja grafą, kai PĮ įjungta, o žalia, kai PĮ yra išjungta.



21 pav. CPU išnaudojumo palyginimas su įjungta ir išjungta PĮ

4. 3. 2 Atstatymo testavimas

53 lentelė. Sistemos

Nr.	Situacija	Sprendimas
116.	Netikėtas vartotojo sistemos išsijungimas	Dar neįrašyti duomenys į duomenų bazę bus pradingę. Ilgiausias laiko tarp duomenų įrašymo į DB yra 10 s. Niekada nebus prarandama daugiau kaip 10 s duomenų. Tokių duomenų praradimas nėra kritinis.
117.	Vartotojo sistemos striginėjimas. Esant 100% RAM apkrovai klavišų laikas tarp paspaudimų yra neteisingi.	Vartotojui reikėtų sustabdyti procesus, kurie sukelia tokius spartos sutrikimus.
118.	PĮ išmeta klaidą ir išsijungia.	Įjungti PĮ iš naujo. Gali būti prarasti duomenys iki 10 s.

4. 3. 3 Suderinamumo testavimas

54 lentelė. PĮ, Windows OS ir Microsoft .NET Framework suderinamumo testavimas

Testo tikslas:					
Testo ID	Reikalavimai/tikslai	Įvykis/įvestis	Laukiamas rezultatas	Gautas rezultatas	T/N
119.	Windows 7 32bit .NET 3.5	Paleidžiama PĮ	PĮ nepasileidžia	PĮ nepasileidžia, prašo atnaujinti .NET Framework	T
120.	Windows 10 64bit	Paleidžiama PĮ	PĮ pasileidžia	PĮ pasileidžia	T

	.NET 4.7				
121.	Windows 8.1 32bit .NET 4.5.2	Paleidžiama PĮ	PĮ pasileidžia	PĮ pasileidžia	T
122.	Windows 8 32bit .NET 4.5.2	Paleidžiama PĮ	PĮ pasileidžia	PĮ pasileidžia	T
123.	Windows 7 32bit .NET 4.7	Paleidžiama PĮ	PĮ pasileidžia	PĮ pasileidžia	T
124.	Windows 10 32bit .NET 4.7	Paleidžiama PĮ	PĮ pasileidžia	PĮ pasileidžia	T

4.4 Testavimo išvados

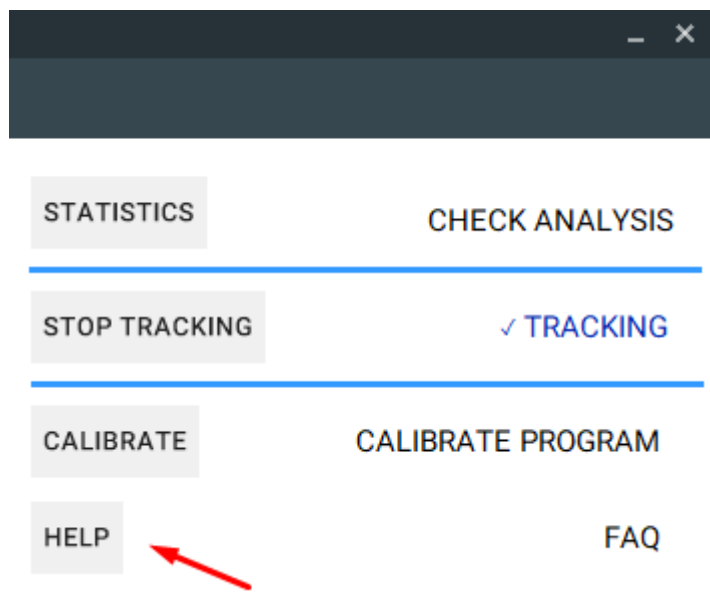
Remiantis atliktais testais galime teigti, kad PĮ veikia stabiliai, atvaizduoja duomenis teisingai. PĮ įsijungia ir veikia su numatytomis OS, veikia našiai ir nesukelia sutrikimų vartotojo kompiuteryje. Duomenų surinkimas yra teisingas daugelyje atvejų, tačiau PĮ neatpažįsta skirtingų klaviatūros išdėstymų, todėl naudojant „DOVARK“ ar kitą išdėstymą karščio žemėlapiu duomenys yra surenkami neteisingai. Tačiau sugeneruoti klavišai ir visa kita statistika yra gaunama teisinga, todėl klaidų statistikos ir linijinių grafikų funkcijos bus atvaizduojamos teisingai.

5. VARTOTOJO DOKUMENTACIJA

5.1 Vartotojo vadovo paskirtis

Vartotojo vadovas yra skirtas leisti vartotojui detaliau susipažinti su PĮ funkcijomis, suprasti jų paskirtį bei kokią naudą jos suteikia.

Vartotojo vadovas yra integruotas programinėje įrangoje, jį galima pasiekti iš pagrindinio lango paspaudus mygtuką „Help“ (žr. 22 pav.).



22 pav. Vartotojo vadovo prieiga sistemoje

5.2 Sistemos funkcinis aprašymas

5.2.1 Statistikos vaizdavimas

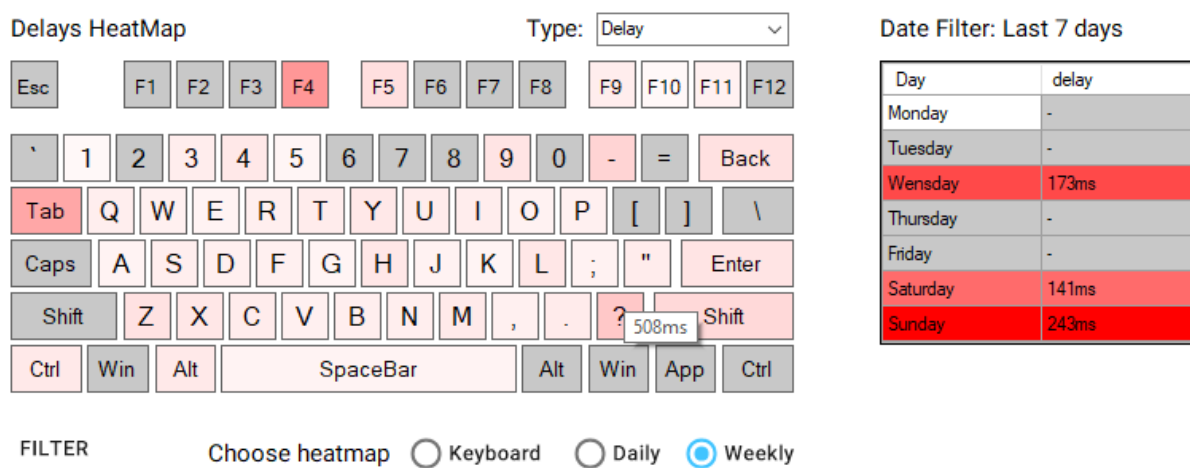
Vartotojos gali peržiūrėti bei filtruoti įvairią informaciją, statistiką apie savo darbą su kompiuterio klaviatūra.

- Klaidų vaizdavimas – vartotojas turi du pasirinkimus, jis gali peržiūrėti savo dažniausiai trinamus bei pakeičiamus simbolius ir kuriuose žodžiuose tai dažniausiai vyksta (žr. 23 pav.). Tokia statistika jam leidžia lengviau identifikuoti su kuriais klavišais ir jų kombinacijomis rašant įvyksta daugiausiai klaidų. Taip pat vartotojas gali peržiūrėti, kuriuose žodžiuose jis daro daugiausiai klaidų, tai leidžia identifikuoti konkrečius žodžius, su kuriais vartotojui iškyla problemų dirbant su klaviatūra.

Deleted	Replaced	Times it happened
s	as	9
a	ą	7
o		
ą		
s		
i		4
hi	hey	4
e	he	3

23 pav. Dažniausiai trinami simboliai ir jų kombinacijos

- Karščio žemėlapiai – vartotojas gali peržiūrėti dviejų tipų karščio žemėlapius – klavišų paspaudimų kiekio bei vidutinės trukmės tarp klavišų paspaudimų. Šiuos karščio žemėlapius jis gali peržiūrėti trimis formatais – klaviatūros, dieniniu bei savaitiniu. Užvedęs pelės rodyklę ant klavišo jis taip pat gali pamatyti detalesnius duomenis. Paveikslėlyje rodomos vidutinės trukmės tarp klavišų paspaudimų karščio žemėlapis klaviatūros bei savaitiniu formatu (žr. 24 pav.). Naudodamas vidutinės trukmės tarp klavišų paspaudimų karščio žemėlapi vartotojas gali identifikuoti, kuriuos mygtukus jis spaudo lėčiausiai, kuriomis valandomis ar savaitės dienomis yra dirbama su klaviatūra lėčiausiai. Klavišų paspaudimo kiekio karščio žemėlapis naudingas norint nustatyti, kuriomis savaitės dienomis bei valandomis vartotojas dirba daugiausiai su klaviatūra.

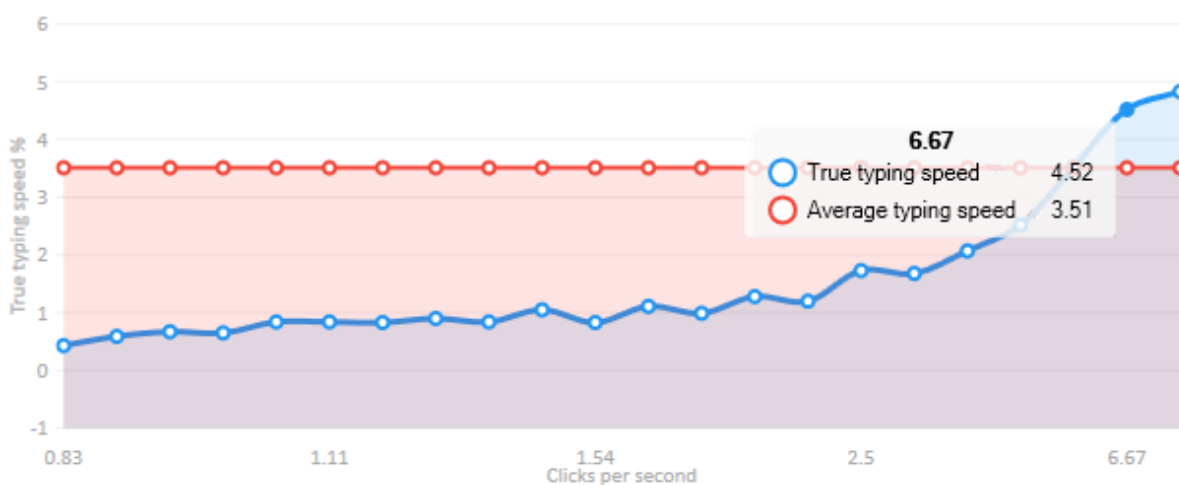


24 pav. Karščio žemėlapių atvaizdavimas

- Linijiniai grafikai – Šie grafikai yra skirti atvaizduoti statistiką, kuri yra susijusi su vidutiniu vartotojo rašymo greičiu bei daromomis klaidomis. Vartotojas gali peržiūrėti keturių tipų grafikus – klaidų santykio priklausomai nuo rašymo greičio, optimalaus rašymo greičio, vidutinio rašymo greičio ir laiko sugaišto taisant klaidas.

- Klaidų santykis pagal rašymo greitį – šis grafikas atvaizduoja vartotojo daromų klaidų kiekį procentais pagal jo rašymo greitį. Pagal tai vartotojas gali identifikuoti, koku greičiu dirbant jis daro daugiausiai klaidų.
- Optimalus rašymo greitis – šis grafikas atsižvelgia į laiką praleistą ne tik rašant bet ir taisant klaidas. Pagal tai galima identifikuoti, koks yra vartotojo optimaliausias rašymo greitis, ar jis nepradeda daryti žymiai daugiau klaidų rašydamas greitai, dėl ko gali sulėtėti teisingai parašyto teksto rašymo greitis. Optimaliausią rašymo greitį vaizduoja aukščiausiai pakilusi mėlyna linija arba tos zonos kurios pakyla virš raudonos linijos. (žr. 25 pav.) .

Optimal typing speed chart



25 pav. Optimaliausio rašymo greičio grafikas

- Vidutinis rašymo greitis – parodo, kaip keičiasi vartotojo vidutinis rašymo greitis per tam tikrą laikotarpį. Naudinga norint pamatyti, ar vartotojo vidutinis rašymo greitis progresuoja ar regresuoja bėgant laikui.
- Laikas sugaištas taisant klaidas – šis grafikas apskaičiuoja, kiek laiko vartotojas sugaišta taisant klaidas priklausomai nuo jo vidutinio rašymo greičio ir klaidų kiekio. Šis laikas yra išreikštas procentais. Taip vartotojas gali identifikuoti, koku rašymo greičiu jis praleidžia daugiausiai laiko taisydamas klaidas.
- Statistikos filtrai – visus duomenys, kurie atvaizduojami statistikoje galima filtruoti dviem būdais – pagal datą bei programas. Naudojant datos filtrą duomenis galima filtruoti pagal datą, datos intervalą arba pagal nustatytus filtrus, pvz.: paskutinės 24 valandos ar paskutinės 7 dienos. Naudojant programų filtrus galima filtruoti pagal norimas programas, kuriose buvo užfiksuotas paspaudimas arba klaida (žr. 26 pav.). Pasirinkus, kad nebūtų rodomi tam tikrų programų duomenis PĮ taip pat nustoja rinktis duomenis iš tų programų. Tokiu būdu vartotojas gali pasirinkti, kuriuose programose jis nenori, kad jo darbas su klaviatūra būtų sekamas.

Name	.exe	Show from
Ease of Access Dialog ...	easeofaccessdial...	☐
Google Chrome	chrome.exe	☒
HookFromV2	hookfromv2.exe	☒
Lightshot	lightshot.exe	☒
Microsoft PowerPoint	powerpnt.exe	☐
Microsoft Visual Studio 2...	devenv.exe	☒
Microsoft Word	winword.exe	☒
Notepad++ : a free (GN...	notepad++.exe	☒
Paint	mspaint.exe	☒

From: Saturday, May 19, 2018 LAST 24 HOURS

To: Saturday, May 19, 2018 LAST WEEK

Selected date filter: **Last 24 hours**

APPLY FILTER

26 pav. Statistikos filtrai

5. 2. 2 Klaviatūros sekimo išjungimas bei įjungimas

Vartotojas gali neišjungęs PĮ reguliuoti, ar jo darbas su klaviatūra yra stebimas ar ne. Ši funkcija reikalinga tam, kad vartotojui nereikėtų kiekvieną kartą išjungti PĮ norint sustabdyti darbo su klaviatūra sekimą.

5. 2. 3 Programinės įrangos kalibravimas

Įsidiegus PĮ vartotojas privalo ją sukalibruoti. Kalibravimas yra privalomas, esant neteisingam sukalibravimui PĮ vartotojo darbą su klaviatūra gali interpretuoti neteisingai ir rodyti klaidingus duomenis. Programinę įrangą galima kalibruoti trimis būdais:

- Automatinis kalibravimas – vartotojui nereikia atlikti jokių papildomų veiksmų. Jis gali tęsti savo darbą su klaviatūra. PĮ seka vartotojo klavišų paspaudimus ir surinkusi pakankamai duomenų juos išanalizuoja ir susikalibruoja PĮ automatiškai.
- Rankinis kalibravimas – vartotojas gali nustatyti vartotojo lygi rankiniu būdą. Jam leidžiamas pasirinkimo intervalas nuo 0,2 s iki 10 s.
- Kalibravimo testas – vartotojas gali atlikti kalibravimo testą ir pagal jo rezultatus PĮ susikalibruoja. Kalibravimo teste vartotojui reikia parašyti tekstą, kurį pateikia PĮ.

5. 3 Sistemos vartotojo vadovas

Vartotojo vadovas yra pagrįstas dažnai užduodamų klausimų formatu.

Jis yra suskirstytas į tris pagrindines kategorijas:

- Sekimas – vartotojui paaiškinama, kaip veikia PĮ, kokie duomenys yra renkami, kaip PĮ įtakoja vartotojo kompiuterio našumą bei kaip sustabdyti ar pradėti sekimą.
- Kalibravimas – paaiškinama, kas yra kalibravimas, kodėl tai reikalinga ir kokiais būdais vartotojas gali tai atlikti.
- Statistika ir filtrai – detalizuojama kiekviena iš statistikos funkcijų, kokią naudingą informaciją šios funkcijos gali suteikti ir kas yra filtrai bei kaip jais naudotis.

5. 4 Sistemos diegimo vadovas

PĮ diegimą galima pradėti atidarius „setup.exe“ failą. Vartotojas turi turėti administratoriaus privilegijas savo kompiuteryje arba diegimo metu suteikti prieigas prie jų. Vartotojo sistema turi

atitikti nurodytas kompiuterio specifikacijas 2.4.1 punkte. Jeigu vartotojo kompiuteryje nėra Microsoft .NET Framework 4.5.2 ar naujesnės versijos programinės įrangos, diegimo metu bus automatiškai atsiųsta reikalinga .NET Framework versija.

Sudiegus PĮ lieka tik ją sukalibruoti arba leisti tai atlikti pačiai PĮ naudojant automatinį kalibravimą.

IŠVADOS

1. Analitinėje dalyje buvo surasti bei išrinkti du tinkamiausi metodai fiksuoti klaviatūros klavišų paspaudimus. Abu metodai atitiko PĮ keliamus reikalavimus, o esminiai skirtumai tarp jų buvo tik funkcijos, kurios nėra aktualios kuriamai PĮ. Todėl pasirinktas metodas, kuris turi daugiausiai dokumentacijos. Apžvelgus tokio tipo PĮ analogus paaiškėjo, kad nei vienas nesiūlo funkcijų, kurios atliktų gilią analizės vartotojo darbo su klaviatūra analizę, kuri padėtų vartotojui identifikuoti sunkumus kylančius darbo su klaviatūra metu ir daromas gramatines klaidas. Iš to galime teigti, kad PĮ kūrimo pagrindas yra pagrįstas.
2. Reikalavimų specifikacijoje buvo numatytos pagrindinės PĮ funkcijos ir savybės. Realizuotos tokios funkcijos, kaip karščio žemėlapiai, klaidų sąrašai bei linijiniai grafikai. Atkreiptas papildomas dėmesys į vartotojo darbo kokybę. Šios funkcijos leis atskleisti kuo daugiau problemų, kurios gali kilti vartotojui dirbant su klaviatūra, o pati PĮ netrukdyt vartotojo darbo su klaviatūra.
3. Sistemos architektūra yra įgyvendinta taip, kad būtų įvykdytos funkcijos ir savybės užsibrėžtos reikalavimų specifikacijoje. PĮ užtikrintas našus jos veikimas, nes naudojami lygiagrečiojo programavimo metodai ir duomenų bazės optimizacijos.
4. Testavimai parodė, kad PĮ yra stabili, net veikiant ilgą laiką ir įvairiomis situacijomis. Taip pat sistema yra naši ir sukelia tik minimalų poveikį vartotojo sistemai, kuris neįtakoja vartotojo darbo su klaviatūra. PĮ veikia daugelyje šių laikų Windows operacinėse sistemose bei surenka, atvaizduoja ir apskaičiuoja duomenis teisingai. Testuojant rastos klaidos yra smulkios ir neturi esminės įtakos PĮ veikimui, todėl sistema yra tinkama viešinimui ir platesniam naudojimui.
5. Vartotojo vadovas vartotojui paaiškina kaip veikia sistemos funkcijos, kokią naudą jos suteikia bei kokie duomenys yra sekami. Jis yra lengvai pasiekiamas iš pačios sistemos, o tai leis sumažinti pagalbos gavimo laiką, nes vartotojo vadovas visada bus po ranka ir nereikės gaišti laiko papildomai jo paieškai.

LITERATŪRA

1. COMPUTER HOPE „When was the first keyboard invented?“ [žiūrėta 2018-01-20]. Prieiga per internetą: <<https://www.computerhope.com/issues/ch001802.htm>>
2. CRISTOPHER SHULTZ „OnScreen vs. Physical Keyboards“ [žiūrėta 2018-05-26]. Prieiga per internetą: <<https://litreactor.com/columns/onscreen-vs-physical-keyboards>>
3. DARRYL REH „The first typewriter“ [žiūrėta 2018-01-20]. Prieiga per internetą: <<https://web.archive.org/web/19991103015919/http://home.earthlink.net/~dcrehr/firsttw.html>>
4. CHRIS STOLLER-WALKER „6 Non-QWERT Keyboard Layouts“ [žiūrėta 2018-03-15]. Prieiga per internetą: <<http://mentalfloss.com/article/52483/6-non-qwerty-keyboard-layouts>>
5. LIEBOWITZ, Stan J.; MARGOLIS, Stephen E. The fable of the keys. The Journal of Law and Economics, 1990, 33.1: 1-25.
6. MALTRON LTD „Maltron Letter Layout“ [žiūrėta 2018-05-26]. Prieiga per internetą: <<https://www.maltron.com/the-maltron-letter-layout-advantage.html>>
7. NIKOLAY GREBNIKOV „Keyloggers: Implementig Keyloggers in Windows. Part Two“ [žiūrėta 2018-03-25]. Prieiga per internetą: <<https://securelist.com/keyloggers-implementing-keyloggers-in-windows-part-two/36358/>>
8. SUBRAMANYAM, Kishore; FRANK, Charles E.; GALLI, Donald H. Keyloggers: The overlooked threat to computer security. In: 1st Midstates Conference for Undergraduate Research in Computer Science and Mathematics. 2003.
9. GIANCARDIO, Luca, et al. Computer keyboard interaction as an indicator of early Parkinson's disease. Scientific reports, 2016, 6: 34468.
10. VAN WAES, Luuk, et al. Typing competencies in Alzheimer's disease: An exploration of copy tasks. Computers in Human Behavior, 2017, 73: 311-319.
11. Application Usage Statistics [žiūrėta 2018-03-25]. Prieiga per internetą: <<https://github.com/UsageStats/UsageStats>>
12. „WhatPulse“ [žiūrėta 2018-03-26]. Prieiga per internetą: <<https://whatpulse.org/>>
13. „RSIGuard“ [žiūrėta 2018-03-26]. Prieiga per internetą: <<https://www.rsiguard.com/>>
14. „KeyboardUsage“ [žiūrėta 2018-03-27]. Prieiga per internetą: <<https://github.com/kbilsted/KeyboardUsage>>
15. „Keyboard Optimizer“ [žiūrėta 2018-03-25]. Prieiga per internetą: <https://researcher.watson.ibm.com/researcher/view_group.php?id=1233>
16. MICROSOFT .NET FRAMEWORK DOKUMENTACIJA „Asynchronus Programming“ [žiūrėta 2018-04-19]. Prieiga per internetą: <<https://docs.microsoft.com/en-us/dotnet/csharp/async>>
17. SQLITE DOKUMENTACIJA „Atomic Commit in SQLite“ [žiūrėta 2018-04-22]. Prieiga per internetą: <<https://www.sqlite.org/atomiccommit.html>>

TERMINŲ IR SANTRUMPŲ ŽODYNĖLIS

Buferis	Struktūra skirta saugoti tam tikrus duomenis atmintyje, kad vėliau juos būtų galima įrašyti į duomenų bazę
C#	Aukšto lygio programavimo kalba kuria yra parašyta PĮ
CPU	Kompiuterio procesorius (angl. central proceesing unit)
Kabliukas	Tai funkcija kuri naudodama Win32 API laukia kol vartotojas paspaus klavišą ir jį užfiksuoja.
Klavišas	Tai bet koks klaviatūros mygtukas. (pvz. Home, Enter, Ctrl, A, Num1)
Kontroleris	Klasė skirta valdyti duomenų srautui, manipuliacijai ir saugojimui
Kursorius	Teksto redaktoriaus kursorius, kurioje vietoje jis yra ten bus rašomi sekantys vartotojo simboliai
OS	Operacinė sistema
Paprastas klavišas	Klavišas kurį paspaudus yra atspausdinamas bet koks simbolis.
Panikos klavišas	Klavišas po kurio paspaudimo nebėra įmanoma atsekti vartotojo kursoriaus pozicijos
PĮ	Programinė įranga
RAM	Operatyvinė atmintis(angl. random accses memorry)
Rodyklė	Nurodo vartotojo kursoriaus poziciją PĮ buferyje
Simbolis	Tai simbolis kuris yra sugeneruojamas po klaviatūros klavišo paspaudimo. Vienas klavišas gali sugeneruoti keletą simbolių (pvz. A, a, 1, !).
SQL	Struktūrizuota duomenų bazių užklausų kalba
Taisyklių klavišas	Klavišas po kurio yra pradedama fiksuoti vartotojo tekstas kaip klaida(pvz. delete, backspace).
Vėliava	Komponento būsenos tikrinimui. Jei vėliava yra aktyvi komponentas elgiasi vienaip, jeigu ne aktyvi – kitaip.
Win32 API	User32.dll failas iš kurio yra naudojamos funkcijos klavišų, aktyvių programų sekimui ir klavišų į simbolius konvertavimą.

PRIEDAI

1 priedas. Klasių detalizavimas

55 lentelė. Klasės „FilterController“ detalizavimas

Klasė	FilterController
Apibrėžimas	Klasė skirta valdyti filtro užklausas
Atributai	+List<ProgramExe> programList +StatsController
Metodai	+FilterHubcharts(bool date_changed, bool programs_changed, DateTime from, DateTime to, DataGridView programGrid, Graphs form, GraphsController, graphCoontroller) +FilterHubHeatMap(bool date_changed, bool, programs_changed, DateTime to, DataGridView programGrid, HeatMap heatmapForm, HeatMapController heatController) +FilterHubDataGrid(bool date_changed, bool programs_changed, DateTime from, DateTime to, DataGridView programGrid, DataGridView gridToUpdate) +InitializePrograms() +DateTimeFilterGrid(DateTime from, DateTime to, DataGridView grid) +CheckBoxFilter(DataGridView filterGrid)
Sąveikavimas	FilterForm

56 lentelė. Klasės „GraphsController“ detalizavimas

Klasė	GraphsController
Apibrėžimas	Klasė skirta valdyti duomenų manipuliavimą ir atvaizdavimą Graphs formoje
Atributai	+Dictionary<double, double> mistakesSpeedDict +List<TypingSpeed_MistakesCount> mistakesList +string whereAnd +string where +string selectedList
Metodai	+Run +RunFirst() +Validation(string view) +AllInputsTypingSpeed() +Typing_Speed() +TypingSpeed_MistakesCount() +AverageTimeItTakesToFixMistakes()
Sąveikavimas	Graphs

57 lentelė. Klasės „HeatMapController“ detalizavimas

Klasė	HeatMapController
Apibrėžimas	Klasė skirta valdyti duomenų manipuliavimą ir atvaizdavimą HeatMap formoje
Atributai	+Dictionary<string, int> heatDict +Dictionary<string, int> weeklyHeatmap +Dictionary<string, int> hourlyHeatmap +string selectedHeatMap +string where
Metodai	+Run() +GetTimesPressedHeatmap() +GetDelayHeatmap() -GetWeeklyHeatmap(string query) -GetDailyHeatmap(string query) -GetHeatmap(string query) +StoreHeatMap(List<InputObj> list)
Sąveikavimas	HeatMap

58 lentelė. Klasės „MainFormController“ detalizavimas

Klasė	MainFormController
Apibrėžimas	Klasė skirta valdyti buferius MainForm formoje
Atributai	+Keys lastKeyPressed +List<InputObj> inputBuffer +bool writeInProgress
Metodai	+AddToInputbuffer(Keys vkCode, int stopWatchTime, int exeID, string type) +LastKeyPressed_LiftOff(Keys vkCode) +StoreHeatMapAndInputs()
Sąveikavimas	MainForm

59 lentelė. Klasės „StatsController“ detalizavimas

Klasė	StatsController
Apibrėžimas	Klasė skirta valdyti duomenų manipuliavimą ir atvaizdavimą Statistics formoje
Atributai	-List<MistakesObject> mistList +string where +string selectedStats +List<KeyvaluePair<string, int>> wordList
Metodai	+RunFirst() +Run() +PopulateDataGrid(DataGridView grid) +GetFinalWords(int id)
Sąveikavimas	Statistics

60 lentelė. Klasės „UserController“ detalizavimas

Klasė	UserController
Apibrėžimas	Klasė skirta nuskaityti vartotojo lygį iš duomenų bazės
Atributai	+int userLevel
Metodai	+UserLevelCheck()
Sąveikavimas	MainForm

61 lentelė. Klasės „Input_Buffer“ detalizavimas

Klasė	Input_Buffer
Apibrėžimas	Klasė skirta sekti vartotojo klavišų paspaudimus
Atributai	List<InputObj> inputList
Metodai	+AddItem(Keys vkCode, int programID, int delay) +AddItem(string input, int programID, int delay)
Sąveikavimas	MainForm

62 lentelė. Klasės „InputForMistakes“ detalizavimas

Klasė	InputForMistakes
Apibrėžimas	Klasė kuri apibrėžia struktūra simboliui kuris bus įrašomas į klaidos buferį
Atributai	+string _input +int _delay
Sąveikavimas	MainForm

63 lentelė. Klasės „InputObj“ detalizavimas

Klasė	InputObj
Apibrėžimas	Klasė nusako klavišą ir su juo ateinančius duomenis
Atributai	+string _inputASCI +string _inputKey +DateTime _dateStamp +int _programID +int _delay

Sąveikavimas	MainController, HeatmaController
---------------------	----------------------------------

64 lentelė. Klasės „TrackingBuffer“ detalizavimas

Klasė	TrackingBuffer
Apibrėžimas	Klasė skirta sekti vartotojo rašoma, redaguojama bei matoma tekstą
Atributai	+List<InputForMistakes> buffer -int _length +int pointer +bool ruleKeyPressed
Metodai	+SendRuleKeys(Keys vkCode, bool ctrl) +GetValue() +GetInputForMistakes(Keys vkCode) +ValidateBounds(int key) +AddToBuffer(string vkCode, int delay) +RuleDelete() +RuleBackSpace
Sąveikavimas	MainForm

65 lentelė. Klasės „MistakesObject“ detalizavimas

Klasė	MistakesObject
Apibrėžimas	Klasė vaizduoja vartotojo padarytą klaidą
Atributai	+string replacedValue +int replacedTimes +string deletedValue +List<string> finalWord
Sąveikavimas	StatsController

66 lentelė. Klasės „ProgramExe“ detalizavimas

Klasė	ProgramExe
Apibrėžimas	Klasė apibrėžia duomenis kurie ateina su sekama programa
Atributai	+int _id +string _name +string _exe +bool _filter
Sąveikavimas	FilterControl, DB

67 lentelė. Klasės „RuleObj“ detalizavimas

Klasė	RuleObj
Apibrėžimas	Klasė apibrėžia klavišą kuris buvo ištrintas su juo ateinančius duomenis
Atributai	+string deletedKey +string replacedWithKey +Keys ruleKeyUsed +int timerDeleted +int timerReplaced
Sąveikavimas	Rules

68 lentelė. Klasės „TypingSpeed_MistakesCount“ detalizavimas

Klasė	TypingSpeed_MistakesCount
Apibrėžimas	Klasė nusako vartotojo daromų klaidų kiekį esant konkrečiam greičiui
Atributai	+double speed +int mistakesCount
Sąveikavimas	GraphsController

69 lentelė. Klasės „InputUtils“ detalizavimas

Klasė	InputUtils
Apibrėžimas	Klasė skirta saugoti kintamuosius ir funkcijas kurių prireikia kitoms klasėms
Atributai	+string seperators +Keys[] seperatorKeys +Keys[] systemKeys +Keys[] ruleKeys +Keys[] panicKeys +Keys[] pointerModifierKeys +Keys[] normalKeys
Metodai	+ReturnSystemKey(int i) +GetKeyType(Keys vkCode) +GetCharsFromKeys(Keys keys)
Sąveikavimas	MainForm, Calibration

70 lentelė. Klasės „CalibrationController“ detalizavimas

Klasė	CalibrationController
Apibrėžimas	Klasė skirta nustatyti vartotojo lygį panaudojus kalibracijos testą
Atributai	+string textToWrite -string[] userWords -string[] systemWords -List<int> delayList
Metodai	+Main(Keys vkCode, string last_char, string typeOfPressedKey) -Results() -GetUserLevel(int ajustedAverage)
Sąveikavimas	Calibration

71 lentelė. Klasės „CalibrationAutomaticController“ detalizavimas

Klasė	CalibrationAutomaticController
Apibrėžimas	Klasė skirta nustatyti vartotojo lygį automatiškai vartotojui dirbant su klaviatūra
Atributai	+bool isOn +int begginingID +string lastRowQuery
Metodai	+Hub() +CalculateUserLevel()
Sąveikavimas	MainForm, CalibrationMain

72 lentelė. Klasės „Rules“ detalizavimas

Klasė	Rules
Apibrėžimas	Klasė skirta valdyti duomenis kai vartotojas paspaudžia taisyklių klavišą
Metodai	+RuleKey(Keys vkCode, List<RuleObj> bamboozledlist, TrackingBuffer bufferObj, int timer_deleted) +SortTuppleList(List<RuleObj> bamboozledList, TrackingBuffer buffObj) +OneWordValitadion(Keys vkCode, TrackingBuffer buffer)
Sąveikavimas	MainForm

73 lentelė. Klasės „WindowsAPI“ detalizavimas

Klasė	WindowsAPI
Apibrėžimas	Klasė skirta importuoti reikalingas funkcijas klaviatūros, procesų sekimui, klavišų ir klavišų konvertavimui
Metodai	+GetForegroundWindow() +SetWindowsHookEx(HookType, HookProc callback, IntPtr hMod, uint dwThreadId) +ToAscii(int uVirtKey, int uScanCode, byte[] lpbKeyState, byte[] lpChar, int uFlags)
Sąveikavimas	MainForm

74 lentelė. Klasės „DBUtils“ detalizavimas

Klasė	DBUtils
Apibrėžimas	Klasė skirta atlikti rašymo ir skaitymo operacijas su duomenų baze
Atributai	-string: conectionString -SQLiteConnection con -SQLiteCommand sqlComm -List<string> queList
Metodai	+OpenConection() +CloseConection() +ExecuteQuery(string query) +ExecuteScalar(string query) +ReadMistakes(string query) +ReadPrograms(string query) +GetExeID(Process p)
Sąveikavimas	MainFormController, FilterController, ExeUtils

75 lentelė. Formos „Calibration“ detalizavimas

Forma	Calibration
Atributai	-CalibrationController calibrationController -MainForm mainForm -CalibratonMain cabMain
Metodai	
Sąveikavimas	CalibrationController

76 lentelė. Formos „CalibrationMain“ detalizavimas

Forma	CalibrationMain
Apibrėžimas	Forma reikalinga tam, kad vartotojas iš vienos vietos galėtų pasiekti visus galimus kalibracijos variantus
Atributai	-MainForm mainFrom
Metodai	LoadText()
Sąveikavimas	CalibrationAutomaticController, CalibrationManual, Calibration

77 lentelė. Formos „CalibrationMnual“ detalizavimas

Forma	CalibrationMain
Apibrėžimas	Forma skirta sukalibruoti PĮ rankiniu būtu
Atributai	-MainForm mainFrom -CalibrationMain cbMain
Metodai	ButtonClick()
Sąveikavimas	CalibrationMain

78 lentelė. Formos „FilterForm“ detalizavimas

Forma	FilterForm
Apibrėžimas	Forma skirta atvaizduoti filtravimo variantus
Atributai	-DataGridView gridToUpd -bool programs_changed -bool date_changed -string type
Sąveikavimas	FilterController

79 lentelė. Fromos „Graphs“ detalizavimas

Forma	Graphs
Apibrėžimas	Forma skirta atvaizduoti linijinius grafikus

Atributai	+GraphsController graphocsController
Metodai	+ChartMistakesCount_Loader() +ChartTypingSpeed_Loader()
Sąveikavimas	GraphsController

80 lentelė. Formos „Heatmap“ detalizavimas

Forma	Heatmap
Apibrėžimas	Forma skirta atvaizduoti karščio žemėlapius
Atributai	+MaterialLabel[] itemArray +HeatMapController heatcontrl
Metodai	+HeatmapLoader() +CallendarHeatmapLoader(Dictionary<string, int> dic, DataGridView grid, int highest) -ColorPicker(int value, in highest)
Sąveikavimas	HeatMapController

81 lentelė. Formos „Statistics“ detalizavimas

Forma	Statistics
Apibrėžimas	Forma skirta atvaizduoti vartotojo daromoms klaidoms
Atributai	-Stopwatch -StatsController mistakes
Metodai	-Statistics_Load(object sender, EventArgs e)
Sąveikavimas	StatsController

82 lentelė. Formos „Web“ detalizavimas

Forma	Web
Apibrėžimas	Forma skirta atvaizduoti vartotojo vadovą.
Atributai	-string startupPath -WebBrowser webBrowser1
Sąveikavimas	MainForm

83 lentelė. Formos „MainForm“ detalizavimas

Forma	MainForm
Apibrėžimas	Forma skirta sekti vartotojo klavišų paspaudimus ir valdyti buferius
Atributai	-TrackingBuffer bufferObj -Input_Buffer inputBuffer -List<InputForMistakes> preDeleteBuffer -List<InputForMistakes> postDeleteBuffer -MainFormController mainController -IntPtr hGlobalLLKeyboardHook -HookProc gklobalLLKeyboardHookCallback
Metodai	-SetGlobalLLKeyboardHook() -RemoveGlobalLLKeyboardHook() +LowLevelKeyboardProc(int nCode, IntPtr wParam, IntPtr lParam) -TriggerFlags(Keys vkCode, Keys triggerKey, KeyboardMessage wmKeyboard, int i) +FinalWord() +GetExeID() +WriteAndClearTuple_nTimer()
Sąveikavimas	WindowsAPI, Rules, TrackingBuffer, InputBuffer, MainFormController, CalibrationAutomaticController