



LATVIJAS UNIVERSITĀTE

DATORIKAS FAKULTĀTE

**PROGRAMMATŪRAS LIETOJUMKVALITĀTES METRIKAS
UN TESTĒŠANAS VEIDI**

BAKALaura DARBS

Autors: Sintija Ustupe

Studenta apliecības Nr.: su11017

Darba vadītājs: docente, Dr.dat. Vineta Arnicāne

RĪGA 2019

ANOTĀCIJA

Darba mērķis bija izpētīt ISO/IEC 25022 standarta lietojumkvalitātes atribūtu metrikas un veidus, kā varētu veikt testēšanu dažādajiem lietojumkvalitātes atribūtiem, kā arī izpētīt to popularitāti interneta vietnēs starp zinātniskajiem rakstiem un starp bloga rakstiem. Ir apkopota informācija par izmantojamajām testēšanas metodēm un nepieciešamo datu ieguves veidiem.

Darbā ir izpētīti lietojumkvalitātes atribūtu testēšanas veidi un pielietojamās metrikas programmatūras lietojumkvalitātes novērtēšanai. Ir tikusi iegūta statistika popularitātes pieaugumam gan zinātniskajos avotos, gan bloga rakstos galvenajiem lietojumkvalitātes atribūtiem – rezultativitāte, efektivitāte, apmierinātība, riskbrīvība un kontekstu pārklājums – un to apakšatribūtiem.

Atslēgvārdi: testēšana, lietojumkvalitāte, metrikas, atribūti

ABSTRACT

Testing methods and metrics of software quality in use

The aim of the thesis was to research ISO/IEC 25022 standard quality in use attribute metrics and the ways in which the testing can be done for different attributes of quality in use as well as to research their popularity in internet sites between scientific articles and blog articles. Information has been gathered about the usable testing methods and the required data collection methods.

In the thesis research has been carried out on the quality in use attribute testing methods and the usable metrics for software quality in use grading. There have been statistics acquired for the popularity growth in both scientific articles and blog articles of the most important quality in use attributes – effectiveness, efficiency, satisfaction, freedom from risk and context coverage – and their subattributes.

Keywords: testing, quality in use, metrics, attributes

SATURS

APZĪMĒJUMI.....	6
IEVADS.....	7
1. LIETOJUMKVALITĀTE.....	10
2. REZULTATIVITĀTE.....	13
2.1. Uzdevumu pabeigtības līmenis.....	15
2.2. Sasniegtie mērķi.....	16
2.3. Kļūdu skaits.....	16
2.4. Kļūdaini veiktie uzdevumi.....	17
2.5. Uzdevumu kļūdu intensitāte.....	17
3. EFEKTIVITĀTE.....	19
3.1. Uzdevuma laiks.....	20
3.2. Laika efektivitāte.....	21
3.3. Izmaksu lietderība.....	22
3.4. Produktīvā laika attiecība.....	22
3.5. Nevajadzīgās darbības.....	22
3.6. Noguruma sekas.....	22
4. APMIERINĀTĪBA.....	24
4.1. Vispārējā apmierinātība.....	25
4.2. Noderīgums.....	26
4.2.1. Apmierinātība ar iespējām.....	27
4.2.2. Diskrētā lietošana.....	27
4.2.3. Funkciju izmantošana.....	28
4.2.4. Neapmierināto lietotāju daļa.....	28
4.3. Uzticēšanās.....	28
4.4. Patika.....	30
4.5. Ērtums.....	32
5. RISKBRĪVĪBA.....	34
5.1. Ekonomiskā riska mazinājums.....	36
5.1.1. Ienākumi no investīcijām.....	37
5.1.2. Uzņēmējdarbības sniegums.....	38
5.1.3. IT ieguldījumu priekšrocības.....	38
5.1.4. Klientu apkalpošana.....	38
5.1.5. Tīmekļa vietnes piesaistītie klienti.....	38
5.1.6. Ienākumi no katra klienta.....	39

5.1.7. Kļūdas ar finansiālām sekām	39
5.2. Veselības un drošības riska mazinājums	39
5.2.1. Lietotāju veselības problēmu ziņošanas biežums	41
5.2.2. Ietekme uz lietotāju veselību un drošību	41
5.2.3. Ietekmējošo lietotāju drošība	41
5.3. Vides risku mazinājums	42
6. KONTEKSTU PĀRKLĀJUMS	44
6.1. Kontekstu pilnīgums	45
6.2. Pielāgojamība	47
6.2.1. Pielāgojamība lietošanas kontekstiem	48
6.2.2. Produkta pielāgojamība	48
6.2.3. Prasmju neatkarība	49
REZULTĀTI	50
SECINĀJUMI	53
IZMANTOTĀ LITERATŪRA UN AVOTI	55
PIELIKUMI	57
1. pielikums. Pirmkods statistikas datu ieguvei valodā Python	57
2. pielikums. Meklēšanai izmantojamās frāzes	58

APZĪMĒJUMI

1. IT – Informācijas tehnoloģija
2. ISO – Starptautiskā standartizācijas organizācija
3. IEC – Starptautiskā elektrotehniskā komisija
4. ROI – Ienākumi no investīcijām
5. BPM – Biznesa snieguma pārvaldība
6. Quality in use – Lietojumkvalitāte
7. Effectiveness – Rezultatīvitate
8. Efficiency – Efektivitate
9. Satisfaction – Apmierinatiba
10. Usefulness – Noderīgums
11. Trust – Uzticēšanās
12. Pleasure – Patika
13. Comfort – Ērtums
14. Freedom from risk – Riskbrīvība
15. Economic risk mitigation – Ekonomiskā riska mazinājums
16. Health and safety risk – Veselības un drošības riska mazinājums
17. Environmental risk mitigation – Vides risku mazinājums
18. Context coverage – Kontekstu pārklājums
19. Context completeness – Kontekstu pilnīgums
20. Flexibility – Pielāgojamība

IEVADS

Lietojumkvalitātes jēdziens IT nozares speciālistu vidū ir reti dzirdēts un lietots, taču, lai nodrošinātu kvalitatīvu sistēmu, būtu vērts veikt lietojumkvalitātes testēšanu, kas ietver vairāku aspektu jeb atribūtu testēšanu.

Lietojumkvalitāte ir tādu aspektu kopums kā sistēmas rezultativitāte un efektivitāte, lietotāju apmierinātība, sistēmas riskbrīvība, kā arī dažādu kontekstu pārklājums. Katru no šiem atribūtiem iespējams testēt atsevišķi, iegūstot dažādas metrikas, kuras vēlāk salīdzināt gan ar izmaiņām laikā, gan veikt dažādu sistēmu salīdzinājumu, balstoties uz iegūtajām metrikām.

Pašlaik vēsturiski lielākais apkopojums par šo tēmu ir 2011. gadā izveidotais standarts ISO/IEC 25010 [1], taču jaunākais ir 2016. gadā izveidotais ISO/IEC 25022 standarts [2], kurš apraksta, kas ir lietojumkvalitāte, kā arī veidus, kā to varētu testēt, savukārt ISTQB glosārijā [3] ir atrodamas definīcijas dažādām testēšanas metodēm, tehnikām, pieejām, kas var palīdzēt veikt lietojumkvalitātes atribūtu testēšanu.

Šī bakalaura darba mērķis ir izpētīt gan ISO/IEC 25022 standartā [2] pieejamo informāciju, gan ISTQB glosārijā [3] aprakstītos testēšanas veidus, un apkopot informāciju no dažādiem citiem avotiem, kas sniegtu plašāku ieskatu tajā, kas ir lietojumkvalitāte un kā to būtu labāk testēt, kā arī iegūt statistikas rādītājus, kā pieaug lietojumkvalitātes popularitāte gan zinātniskajos rakstos, gan bloga rakstos, kas norāda, cik lielā mērā par katru no lietojumkvalitātes atribūtiem un tā testēšanu ir atrodama un kļūst pieejamāka informācija interneta avotos.

Statistikas rādītāji ir iegūti galvenajiem lietojumkvalitātes atribūtiem, kas redzami attēlā 1.1., neveicot dziļāku apakšatribūtu izpēti darba ierobežotā apjoma dēļ.

Lai sasniegtu izvirzīto mērķi, bakalaura darba uzdevumi bija:

- izpētīt un apkopot informāciju, kādas ir līdzības un atšķirības starp dažādiem avotiem, kas apraksta, kā veikt testēšanu dažādiem lietojumkvalitātes atribūtiem;
- reizi nedēļā veikt statistikas datu iegūvi, cik meklēšanas rezultāti atrodami zinātniskajos avotos, izmantojot *Google Scholar* vietni [4], un bloga rakstos, izmantojot *Blog Search Engine* vietni [5].

Darbā izstrādes procesā tika pielietotas vairākas metodes gan lietojumkvalitātes atribūtu un metriku informācijas ieguvei un apkopošanai, gan statistikas datu ieguvei.

Lai izpētītu lietojumkvalitātes atribūtu popularitāti interneta vietnēs, kas norāda, cik daudz informācija par katru atribūtu un tā testēšanu ir pieejama internetā, tika izvēlētas dažāda rakstura meklēšanas tīmekļa vietnes, no kurām viena veic meklēšanu starp zinātniskajiem avotiem, savukārt otra – starp bloga rakstiem, kas ir neformālāka rakstura.

Zinātnisko avotu apstrādei tika izvēlēta Google Scholar meklēšanas vietne tās popularitātes un uzticamības dēļ [6] un ērtās automatizētās datu savākšanas dēļ. Kā citas alternatīvas ir izmantojamas tādas vietnes kā CiteSeerx [7] vai Microsoft Academic Research [8]. Daudziem no meklēšanas avotiem ir redzams teksts ‘Powered by Google Custom Search’, kas liecina, ka tie avoti nemaz neveic atsevišķu meklēšanu, bet gan izmanto modificētu produktu Google.

Meklēšanas vietne bloga rakstiem tika izvēlēta Blog Search Engine, kurā tiek veikta meklēšana starp dažādiem rakstiem, kurus var būt veidojuši jebkuri cilvēki un tiem nav jābūt oficiāli apstiprinātiem. Tā kā autoram jau ir bijusi pieredze ar šo tīmekļa vietnes izmantošanu, meklējot bloga rakstus, tāpēc arī tika izvēlēta konkrēti šī vietne.

Sākotnēji meklēšana tika veikta tikai pēc angļiskā testēšanas atribūta nosaukuma, taču, veicot iegūto rezultātu analīzi, nejauši izvēloties kādu no atrastajiem rakstiem, tika iegūti tādi rezultāti, kas atbilst ķīmijai vai bioloģijai, vai medicīnai, bet neatbilst programmatūras testēšanai. Tādēļ tika veiktas modifikācijas meklējamās frāzes uzbūvē.

Meklēšana abās vietnēs tika veikta, liekot meklējamo atribūtu pēdējās un pievienojot tekstu aiz tām – software testing. Tas tika darīts tādēļ, lai atlasītu tādus rakstus, kas saistīti tieši ar programmatūras testēšanu, nevis, piemēram, medicīnu. Meklēšanai izmantotās frāzes redzamas 2. pielikumā.

Statistikas datu iegūšanas atvieglošanai ir izveidota programma valodā Python, kas redzama 1. pielikumā. Valoda Python tika izvēlēta tās vienkāršības un pieejamo bibliotēku dēļ. Izstrādātā programma automātiski nolasa meklēšanas rezultātus no tīmekļa vietnes Google Scholar [4], izvadot masīvu ar skaitļiem, kurus pēc tam var ievietot sev vēlamajā vietā.

Tālāk dati tika glabāti programmas Microsoft Excel failā, jo šī programma ļauj ērti pārskatīt un kārtot datus, kā arī iespējams veidot vizuālas datu attēlošanas diagrammas.

Datu ieguve notika katru svētdienu periodā no 2018. gada 30. oktobra līdz 2019. gada 5. maijam, lai vērotu progresu, cik lielā mērā informācijas apjoms pieaug par to, kā varētu veikt testēšanu un salīdzināt iegūtos rezultātus dažādiem atribūtiem.

Bakalaura darba struktūra ir veidota, izmantojot standartā ISO/IEC 25022 [2] aprakstīto lietojumkvalitātes atribūtu hierarhijas modeli, no kura tika iegūti pētāmie lietojumkvalitātes atribūti un to apakšatribūti.

Lai veiksmīgāk varētu veikt šo atribūtu testēšanu, tika meklētas testēšanas metodes, kuras būtu izmantojamas konkrētā lietojumkvalitātes aspekta novērtēšanai. Kā viens no izsmeļošākajiem avotiem, kurā ir aprakstītas testēšanas metodes, bija ISTQB glosārijs [3], kurā ir alfabēta kārtībā izveidots saraksts ar testēšanas terminiem, no kuriem tika apkopoti tie, kuri būtu atbilstošie katram no lietojumkvalitātes atribūtiem.

Tika veikta dažādu citu informācijas avotu apkopošana un izzināšana, taču tajos atrodamā informācija vairākos gadījumos bija balstīta uz jaunāko standartu ISO/IEC 25022 [2] vai arī senākiem. Plašāks klāsts ar pieejamajiem avotiem bija tādiem lietojumkvalitātes atribūtiem, kuri jau bija atrodami standartā ISO/IEC 9126 [9]. Veicot dažādo avotu izpēti, tika iegūts plašs apskats par lietojumkvalitātes atribūtiem un to metrikām.

Bakalaura darbs sastāv no vairākām daļām, kur sākotnēji tiek dots vispārējs lietojumkvalitātes apraksts, bet tālāk seko nodaļas, kas ir veltītas katram no lietojumkvalitātes atribūtiem - sistēmas rezultativitātei, efektivitātei, lietotāju apmierinātībai, sistēmas riskbrīvībai, kā arī sistēmas kontekstu pārklājumam, apskatot arī konkrēto atribūtu metrikas.

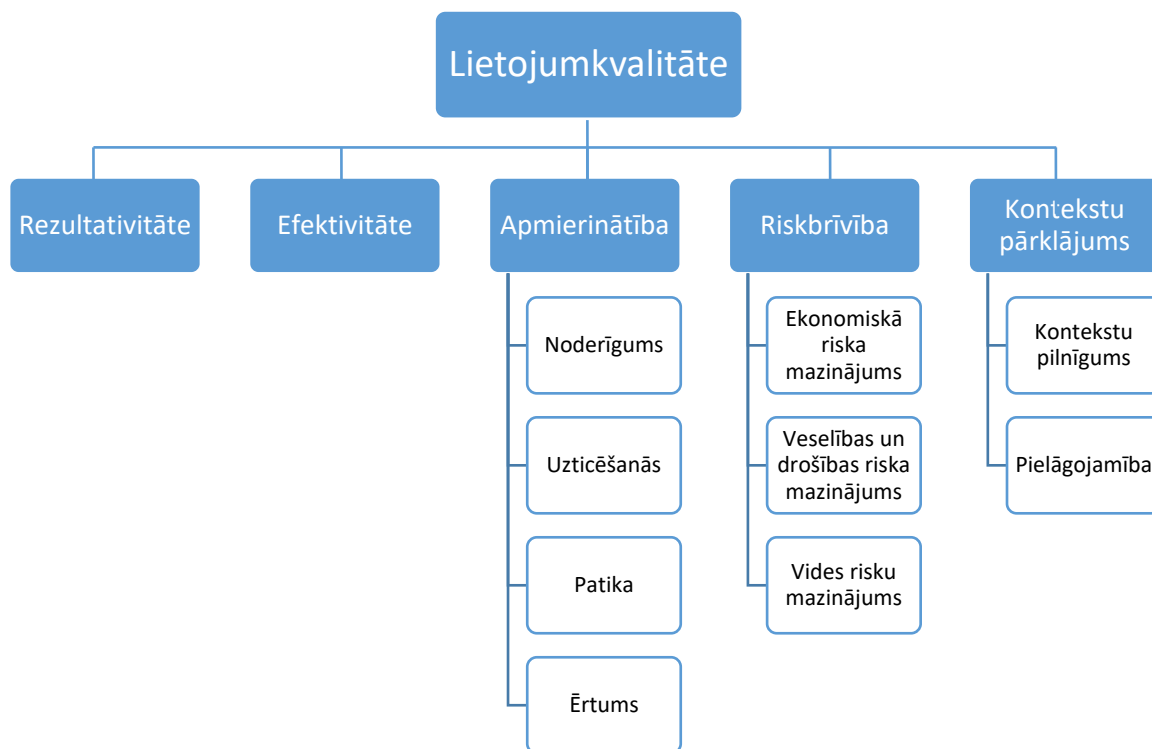
1. LIETOJUMKVALITĀTE

Termins “lietojumkvalitāte” ir radies no agrākā apzīmējuma “lietojamība”, kas ietvēra tādus aspektus kā noderīgumu, rezultativitāti, efektivitāti, apgūstamību, apmierinātību, pieklūstamību [10]. Mūsdienās lietojamības jēdziens pieder pie lietojumkvalitātes, ietverot arī papildus aspektus kā riskbrīvību un kontekstu pārklājumu.

Standartā ISO/IEC 25022 [2] minēts, ka lietojumkvalitāte ir pakāpe, kurā produktu vai sistēmu var lietot konkrēti lietotāji, apmierinot savas vajadzības, lai ar rezultativitāti, efektivitāti, apmierinātību un riskbrīvību noteiktos lietojumkontekstos sasniegtu konkrētus mērķus.

Programmatūras produkta vai sistēmas lietojumkvalitāti var izmērīt, novērtējot mērķa sistēmas vai programmatūras produkta efektu, kad lietotāji implementēto sistēmu ir izmantojuši, vai ir tikusi veikta alfa un beta testēšana, kas atšķiras ar testētāju grupu, kur alfa testēšanas laikā to dara pašu izstrādātāju testētāju komanda, savukārt beta testēšanas fāzē testēšanu veic izvēlēta lietotāju grupa, kas ir līdzīgāka galalietotājam [11].

Lai precīzāk novērtētu lietojumkvalitāti, tā tiek sīkāk dalīta atribūtos, no kuriem daži dalīti vēl smalkāk, kurus var izmērīt atsevišķi, palielinot lietojumkvalitātes novērtēšanas precizitāti un adekvātumu. Sīkāks dalījums ir redzams attēlā 1.1.

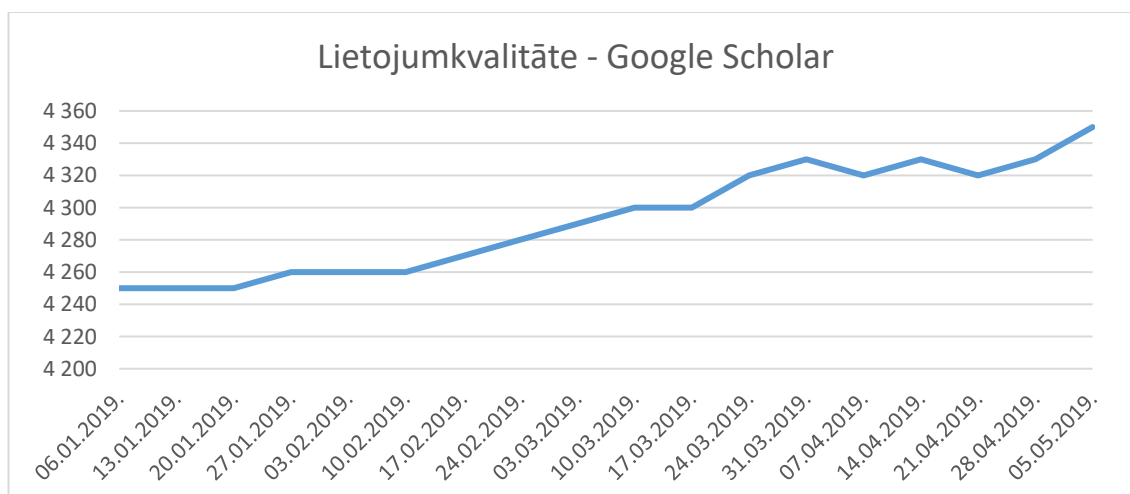


1.1. att. Lietojumkvalitātes hierarhija standartā ISO/IEC 25022 [2]

Tiek uzskatīts, ka lietojumkvalitātei iedala piecus atribūtus – rezultativitāti, efektivitāti, apmierinātību, riskbrīvību un kontekstu pārklājumu, bet dažiem no tiem vēl ir arī apakšatribūti.

Lai novērtētu, cik sistēma ir piemērota lietotājiem, ir svarīgi izvēlēties testa dalībniekus, kas nav no izstrādes komandas [12], jo izstrādātāji parasti sistēmu zina no galvas, tādēļ nav vērts izmantot zinošus cilvēkus, lai iegūtu objektīvus datus par lietojumkvalitāti. Novērtēt, vai sistēma ir intuitīvi saprotama, var tikai tādi lietotāji, kuri nav pie sistēmas pieraduši, bet vislabāk šādu aspektu var novērtēt cilvēki, kas pirmo reizi sistēmu redz.

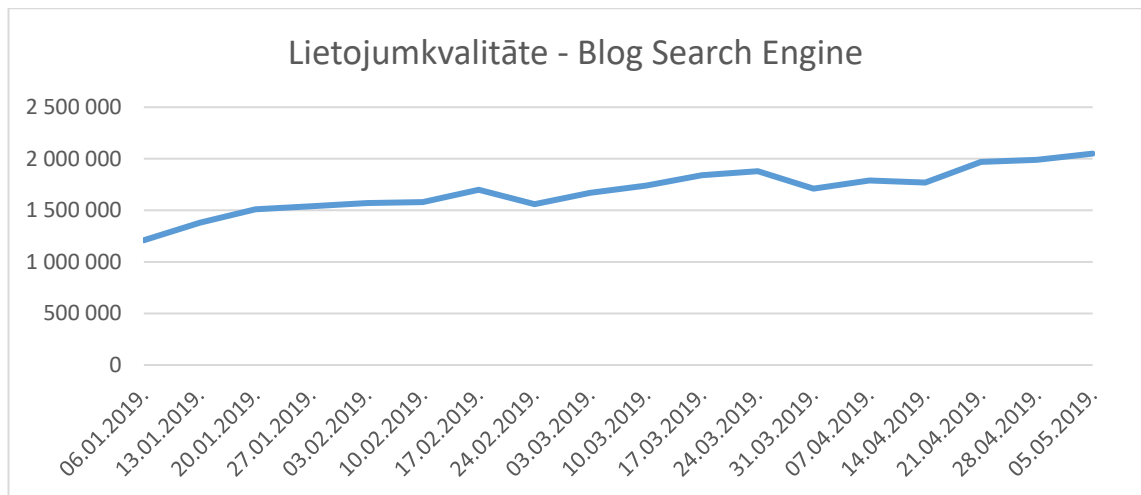
Termins “lietojumkvalitāte” kļūst aizvien populārāks arī dažādās informācijas meklēšanas vietnēs. Pieaugums vērojams gan starp zinātniskajiem avotiem, gan bloga rakstiem, kas liecina par to, ka šī tēma cilvēkus sāk interesēt aizvien vairāk.



1.2. att. Google Scholar vietnes meklēšanas rezultāti laikā frāzei - "Quality in use" software testing

Google Scholar vietnē [4] 6. janvārī 2019. gadā bija nedaudz virs 4 tūkstošiem ierakstu, taču 2019.gada 5. maijā ir bijuši jau par aptuveni 100 ierakstiem vairāk. Pa šo laiku pieaugums ir bijis 2,3 %. Attēlā 1.2. ir attēlota līkne, kā pieaudzis atrasto rakstu skaits. Ir redzams, ka 2019. gada pirmajā ceturksnī rakstu skaits ir pieaudzis daudz straujāk nekā pēc tam, kas liecina, ka šis periods ir bijis mazāk aktīvs jaunu publikāciju vai grāmatu tapšanā par pirmo ceturksni, savukārt no aprīļa beigām ir vērojams atkal straujāks rakstu pieaugums, kas nozīmē, ka ir bijusi lielāks intereses kāpums par lietojumkvalitātes testēšanas metrikām un veidiem.

Vietnē Google Scholar [4] atlasot rakstus pa gadiem, ir redzams, ka kopš 2014. gada jaunu rakstu skaits gadā ir pieaudzis par apmēram 100 rakstiem vairāk, kas liecina par to, ka šī testēšanas joma kļūst aizvien populārāka.



1.3. att. Blog Search Engine vietnes meklēšanas rezultāti laikā frāzei - "Quality in use" software testing

Blog Search Engine vietnē [5] 6. janvārī 2019. gadā bija virs 1 miljona atrasto ierakstu, kas ir daudzkārt vairāk kā Google Scholar vietnē, taču 2019. gada 5. maijā ierakstu skaits ir sasniedzis jau 2 miljonus. Pa šo laiku pieaugums ir bijis 41 %, kas liecina par to, ka aizvien vairāk cilvēku interesējas un apspriež šo tēmu dažādās interneta vietnēs.

Attēlā 1.3. ir redzama līkne ir vienmērīgāka nekā tas bijis Google Scholar [4] vietnē, kas nozīmē, ka visu aplūkoto periodu cilvēku interese par šo tēmu ir bijusi līdzīga, taču jāņem vērā arī fakts, ka atrasto ierakstu skaits ir bijis daudz lielāks, kas nozīmē, ka arī iegūtā līkne ir mazāk precīza, tāpēc lietojama kopainas izpratnei.

Ja pēta rakstu skaita pieaugumu pa gadiem, tad ir redzamas līdzīgas iezīmes senajiem termiņiem un līdzīgas – jaunajiem. Vietnē Google Scholar [4] ir vērojams, ka rakstu skaita pieaugums kopš 2013. gada ar katru gadu sarūk (skatīt rezultātu sadaļā 1. attēlu), savukārt jauno terminu rakstu skaita pieaugumam ir tendence palielināties (skatīt rezultātu sadaļā 2.attēlu). Tā kā termini ir jauni, tad arī pieejamā informācija par to testēšanu ir niecīga, kas varētu būt par iemeslu, kādēļ ik gadu rodas arvien vairāk jaunu rakstu. Cilvēki sāk testēt jaunus sistēmas kvalitātes aspektus un ar informāciju dalās publiskajā telpā.

Turpmākajās nodaļās tiek sniegts plašs apskats par katru no lietojumkvalitātes galvenajiem atribūtiem, kas attēloti attēlā 1.1. – rezultativitāti, efektivitāti, apmierinātību, riskbrīvību un kontekstu pārklājumu – un to iegūstamajām metrikām un testēšanas veidiem.

2. REZULTATIVITĀTE

Nereti rezultativitātes un efektivitātes jēdzieni tiek uzskatīti par vienu un to pašu, taču būtiski izprast, ka tās ir atšķirīgas lietas.

Latviešu valodas skaidrojošajā vārdnīcā “Tēzaurus” [13] ir teikts, ka rezultativitāte ir atdeve jeb efektivitātes mērs, ko nosaka kā starpību starp ieguldījumu un sasniegtajiem rezultātiem.

Savukārt, balstoties uz ISO/IEC 25022 standartu [2], rezultativitāte ir precizitāte un pabeigtība, ar kādu lietotāji ir sasnieguši konkrētus mērķus. Šī metrika ļauj salīdzināt dažādu sistēmu rezultativitāti, sniedzot ieskatu tajā, cik daudz no vēlamajām sistēmas sniegtajām iespējām izdosies izmantot.

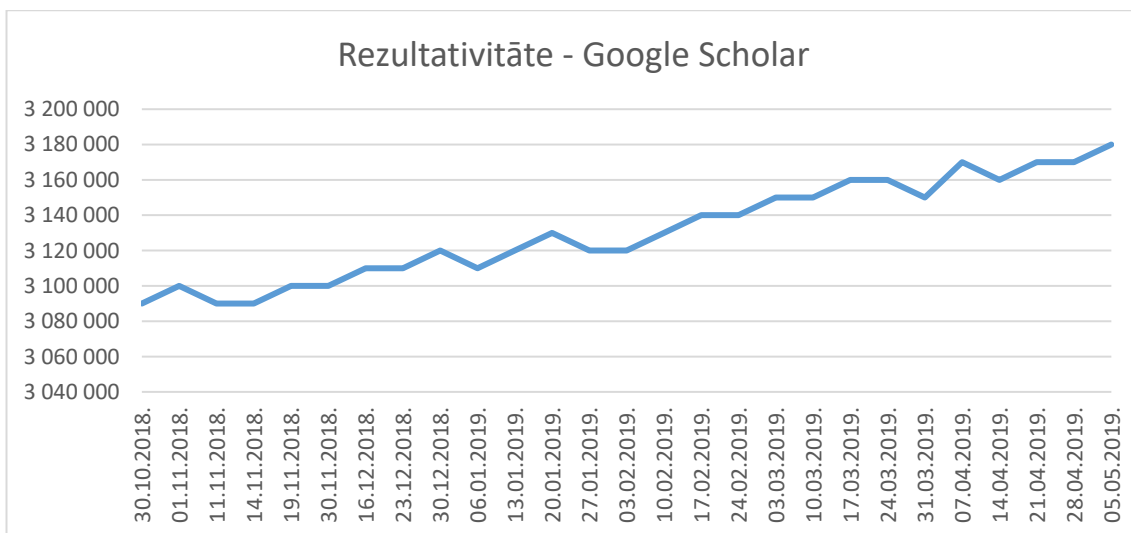
Rezultativitātes metrikas neņem vērā, kā mērķi tika sasniegti, bet gan, cik daudz tie tika sasniegti. Ja mērķa sasniegšanai nepieciešams nospiegt divreiz vairāk pogu, kā iepļānots, tad šis aspekts neietekmē rezultativitāti, jo mērķis ir ticis sasniegts, iegūstot korektu rezultātu tik un tā.

ISTQB glosārijā [3] teikts, ka rezultativitāte ir pareizu un pilnīgu mērķu sasniegšanas apjoms. Tas ir salīdzināms lielums, kas var kalpot kā bāzes punkts, veidojot jaunas vai uzlabojot esošās sistēmas. Glosārijā minēts, ka rezultativitātei ir saistība ar efektivitāti, kas tiek sīkāk aprakstīta 3. nodaļā.

Rezultativitātes testēšanai var pielietot lietotājcentrētu testēšanas pieeju, kuras mērķis ir uzlabot lietotāju saskarni [12]. Pirmkārt, ir nepieciešams izvēlēties tādu personu, kura atbilst mērķa lietotāja profilam, bet šī persona nebūtu izstrādes komandas dalībnieks, lai viedoklis par sistēmu būtu maksimāli adekvāts. Otrkārt, nepieciešams šo testa dalībniekam pastāstīt nelielu testēšanas scenāriju, uzsverot, ka nav nepareizu atbilžu, lai izvēlētajai personai nebūtu neērti vai nepatīkami teikt negatīvos sistēmas aspektus. Kamēr lietotājs veic nepieciešamās darbības, ar viņu nav ieteicams sarunāties, arī, ja rodas kādi jautājumi lietotājam, jāpaziņo, ka jums nav tiesību viņu nekādi ietekmēt.

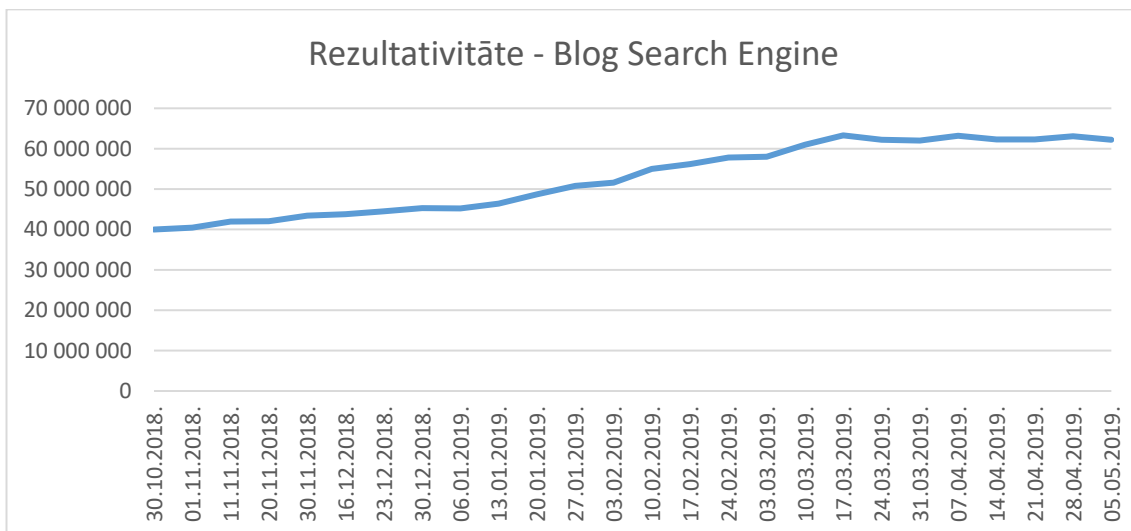
Šādi vērojot, cik ilgi lietotājs uzdevumu pilda vai cik nevajadzīgas darbības lietotājs veic, var saprast, ko nepieciešams sistēmā pamainīt, lai uzlabotu tās efektivitāti. Šo pašu testēšanas veidu var pielietot arī 2. – 4. nodaļā aprakstīto lietojumkvalitātes atribūtu novērtēšanai.

Pārraugot ierakstu skaitu dažādās meklēšanas vietnēs, ir redzams, ka nemitīgi tiek papildināts gan zinātniskās literatūras klāsts, gan arī cilvēki bloga vietnēs apspriež šo tēmu arvien biežāk.



2.1. att. Google Scholar vietnes meklēšanas rezultāti laikā frāzei - "Effectiveness" software testing

Google Scholar vietnē [4] 30. oktobrī 2018. gadā bija nedaudz virs 3 miljoniem ierakstu, taču 2019. gada 5. maijā rakstu skaits sasniedza 3,18 miljonus. Pa šo laiku pieaugums ir bijis 2,8 %. Attēlā 2.1. ir redzamā līkne attēlo atrasto rakstu skaita pieaugumu, kas ir pieaudzis lineāri, bez strauju kāpumu vai nemainības periodiem.



2.2. att. Blog Search Engine vietnes meklēšanas rezultāti laikā frāzei - "Effectiveness" software testing

Blog Search Engine vietnē [5] 30. oktobrī 2018. gadā bija aptuveni 40 miljoni ierakstu, taču 2019. gada 5. maijā datumā ir bijuši jau ap 62 miljoniem ierakstu. Pa šo laiku pieaugums ir bijis nedaudz virs 35 %. Attēlā 2.2. ir redzama līkne, kā pieaudzis atrasto rakstu skaits. Laika posmā līdz 2019. gada marta vidum ir vērojams, ka cilvēki par šo testēšanas aspektu ir interesējušies un pieminējuši to dažādos rakstos vairāk, nekā tas ir bijis pēc marta vidus.

Turpmākajās apakšnodaļās tiek aprakstītas dažādas rezultatīvātes metrikas, kā noteikt sistēmas rezultatīvāti un kādus rādītājus tam nepieciešams iegūt.

2.1. Uzdevumu pabeigtības līmenis

Par sistēmas rezultativitāti liecina, cik uzdevumus ir izdevies veikt veiksmīgi no visiem uzsāktajiem [2]. Uzdevumu pabeigtības līmenis ir aprēķināms, piešķirot tikai divas iespējamās izvēles testa dalībniekam. Ja testa dalībnieks spēj uzdevumu izpildīt, tad atbilde ir "1", bet gadījumā, kad uzdevums nav paveicams, tiek sniegta atbilde "0". Tādējādi ir viegli noteikt proporciju veiksmīgajiem uzdevumiem pret visiem uzdevumiem kopumā.

Vērtēšanas vienkāršības dēļ šī metrika ir viegli saprotams rādītājs, tādēļ ieguvis tik lielu popularitāti izstrādātāju vidū. Šo vērtēšanas metodi var pielietot gan individuālam lietotājam, gan arī lietotāju grupai. Ja kāds no uzdevumiem ir ar augstāku sarežģītības pakāpi, tad var pielietot katram uzdevumam koeficientu, jo, ja grūtākos uzdevums var lietotāji paveikt un sasniegt uzstādīto gala mērķi, tad rezultativitāte ir augstāka, jo uzdevumam tiek piešķirts lielāks koeficients. Turklāt šos rezultātus var iegūt jebkurā sistēmas izstrādes stadijā.

Tādējādi rezultativitāti var attēlot procentuāli, izmantojot vienkāršu vienādojumu, kurā tiek dalīts visu to uzdevumu skaits, kas ir paveikti veiksmīgi, ar visiem tiem uzdevumiem, kas ir tikuši uzsākti, iegūto vērtību reizinot ar 100, tiek iegūta procentuālā vērtība.

Lai gan teorētiski vienmēr jācenšas sasniegt 100% pabeigtības līmeni, balstoties uz Jeff Sauro pētījumu [2], vidēji uzdevumu pabeidz 78% testa dalībnieku (balstīts uz 1 100 uzdevumu analīzi). Tajā pat pētījumā tika arī novērots, ka pabeigtības līmenis ir ļoti atkarīgs no novērtējamā uzdevuma konteksta.

Vēlams ir veidot tādu sistēmu, kur visi lietotāji varēs izpildīt uzdevumu, taču šis pētījums norāda uz to, ka praktiski neiespējami ir sasniegt 100% pabeigtības līmeni, kas ir teorētisks apgalvojums.

Pētījumā iegūtos 78% var izmantot, vērtējot un salīdzinot iegūtos testēšanas rezultātus, kā arī veidojot nefunkcionālās prasības [14] jaunu sistēmu ieviešanai.

Piemēram, ja testā piedalās pieci dalībnieki, kas izpilda vienu un to pašu uzdevumu vienā un tajā pašā sistēmā, un testa beigās trīs lietotāji ir veiksmīgi sasnieguši vēlamo rezultātu, kamēr divi atlikušie dalībnieki nav nonākuši pie rezultāta, pielietojot augstāk minēto vienādojumu, kopumā sistēmas rezultativitāti mēra, dalot trīs ar pieci, tad sareizinot ar 100, iegūstot rezultātu 60, tātad var secināt, ka šajā konkrētajā piemērā sistēmas rezultativitāte ir 60% [15], kas ir mazāka par 78%, ko varētu uzskatīt par robežšķirtni, tātad sistēmai būtu uzskatāms nepietiekams rezultativitātes līmenis.

2.2. Sasniegtie mērķi

Nenoliedzami rezultativitāte vislielākajā mērā asociējas ar paveiktā daudzumu jeb sasniegtajiem mērķiem. Jo vairāk mērķi tiek sasniegti no visiem uzstādītajiem, jo produktīvāka darbošanās ir notikusi.

Kā aprakstīts ISO/IEC 25022 standartā [2], tad rezultativitāti var mērīt, nosakot proporciju, kuri uzdevumu mērķi ir sasniegti korekti, neizmantojot nekāda veida palīdzību. To var aprēķināt pēc formulas:

$$\{Rezultativitāte = 1 - \sum A_i \vee rezultativitāte \geq 0\} \quad (2.1)$$

A_i šajā gadījumā apzīmē proporcionālo vērtību katram palaistam garām vai nesasniegtam mērķim uzdevumu izvadē (maksimālā vērtība = 1, ja netiek sasniegts neviens mērķis).

Piemēram, lietotājam ir viens uzdevums ar četriem mērķiem:

- pierēģistrēties lietotnē;
- atrast konkrētu preci, ievadot nosaukumu meklētājā;
- atrast konkrētu preci, izvēloties no saraksta;
- nopirkt abus atrastos produktus.

Pirmo mērķi lietotājam izdodas paveikt veiksmīgi, tātad šajā gadījumā proporcionālā vērtība A_1 ir nulle dalīts ar četri, jo no četriem uzdevumiem šajā nebija nevienas problēmas. Savukārt, otro mērķi lietotājam neizdodas sasniegt, tātad A_2 vērtība ir viens pret četri, kas ir 0,25, bet trešais mērķis nesagādā nekādas grūtības, tāpēc A_3 vērtība ir nulle. Tā kā lietotājs varēja atrast tikai vienu no divām precēm, tad nopirkt arī bija iespējams tikai vienu atrasto preci, kas nozīmē, ka ceturta mērķa A_4 ir 50% no vienas ceturtdaļas, kas ir 0,125.

Šo visu saskaitot kopā, iegūst formulu:

$$\begin{aligned} Rezultativitāte &= 1 - (A_1 + A_2 + A_3 + A_4) = \\ &= 1 - (0 + 0,25 + 0 + 0,125) = 1 - 0,375 = 0,625 \end{aligned} \quad (2.2)$$

Tātad šajā piemērā pēc formulas (2.2) var spriest, ka sistēmas rezultativitāte ir 0,625 jeb 62,5% procentos gadījumu sistēmas darbojas rezultatīvi.

2.3. Kļūdu skaits

Cits mērījums ietver problēmu skaita noteikšanu, ko veic testa dalībnieki, mēģinot pabeigt uzdevumu. Mēģinot izpildīt uzdevumu, problēmas var rasties dažādas – neplānotas darbības, kļūdas vai darbību izlaišana.

Ideālā gadījumā vajadzētu pievienot īsu aprakstu atklātajām problēmām, smaguma pakāpes, kā arī klasificēt katru problēmu attiecīgā kategorijā. Lai gan tas var būt laikietilpīgi, problēmu skaita noteikšana nodrošina teicamu diagnostikas informāciju.

Pamatojoties uz 719 uzdevumu analīzi, kas tika veikta, izmantojot patērētāju un biznesa programmatūru, Jeff Sauro [2] nonāca pie secinājuma, ka vidējais kļūdu skaits uz vienu uzdevumu ir 0,7 kļūdas.

Tikai 10% no novērotajiem uzdevumiem tika veikti bez jebkādam kļūdām, tādējādi tika nonākts pie secinājuma, ka lietotājiem ir pilnīgi normāli kļūdoties, veicot uzdevumus [15].

No vienas puses, pēc šī pētījuma var secināt, ka nevar uzstādīt prasību, ka jāizveido tāda programmatūra vai sistēma, kuru lietojot cilvēks nekad nepieļaus kļūdas, taču no otras puses – programmatūrai ir jāsniedz cilvēkam iespēju nekļūdīgi veikt nepieciešamās darbības.

Šos 10% var izmantot, vērtējot un salīdzinot iegūtos testēšanas rezultātus, kā arī veidojot nefunkcionālās prasības [14] jaunu sistēmu ieviešanai.

2.4. Kļūdaini veiktie uzdevumi

Tā kā mēdz būt situācijas, kad ir būtiski zināt, cik daudz uzdevumu ir veikts kļūdaini, tad ir ieviesta arī šāda metrika, ar kuru noteikt rezultativitātes līmeni vai pakāpi [2].

Šis mērījums un iegūtā vērtība ir pretēja uzdevumu pabeigtības līmenim, jo mēra proporciju tiem uzdevumiem, kas veikti kļūdaini, attiecībā pret visu uzdevumu skaitu. Formula ir dalījums kļūdainajiem uzdevumiem ar visiem uzdevumiem, kas savukārt sareizināts ar 100%, lai galarezultātā tiktu iegūta procentuāla vērtība, ko salīdzināt dažādās izstrādes stadijās, neprasot sevišķu piepūli.

Ja lietotājs, piemēram, ir kļūdījies četros no pieciem uzdevumiem, tad sanāk, ka kļūdaini veikto uzdevumu īpatsvars ir 80% no visiem uzdevumiem.

Kļūdas var būt dažāda rakstura, gan sākot ar sīkām neprecizitātēm, gan beidzot ar kritiskām sistēmas kļūdām, tomēr tās visas uzskatāmas par neveiksmēm, kas ietekmē kļūdaini veikto uzdevumu procentuālo vērtību.

2.5. Uzdevumu kļūdu intensitāte

Dažkārt kļūda gadās vienam no miljons cilvēkiem, bet citreiz kļūda atkārtojas gandrīz katram lietotājam.

Atkarībā no tā, cik lietotājiem kļūda atkārtojas, nosaka kļūdas labošanas prioritāti. Ar augstāko prioritāti tiek apzīmēti tādi kļūdu pieteikumi, kas atkārtojas vairumam cilvēku, taču ar zemāko prioritāti – tādi, kas ir ļoti retos gadījumos. Pieteikuma labošana nenesīs peļņu, ja tas palīdzēs vienam lietotājam, turpretī ienesīgumu palielinās kļūdu novēršana, kas traucē sistēmas lietošanu vairumam cilvēku.

Papildus nepieciešams arī novērtēt kļūdas nozīmīgumu jeb smagumu, jo, piemēram, medicīnas vai aviācijas sistēmās katra kļūda var būt kritiska un beigties letāli kādam cilvēkam.

Tad uzdevumu kļūdu intensitātes mērījumus nebūtu ieteicams izmantot, taču šī metrika ir lieliski piemērojama dažādām izklaidējošām vai mazāk nozīmīgāk sistēmām un lietotnēm.

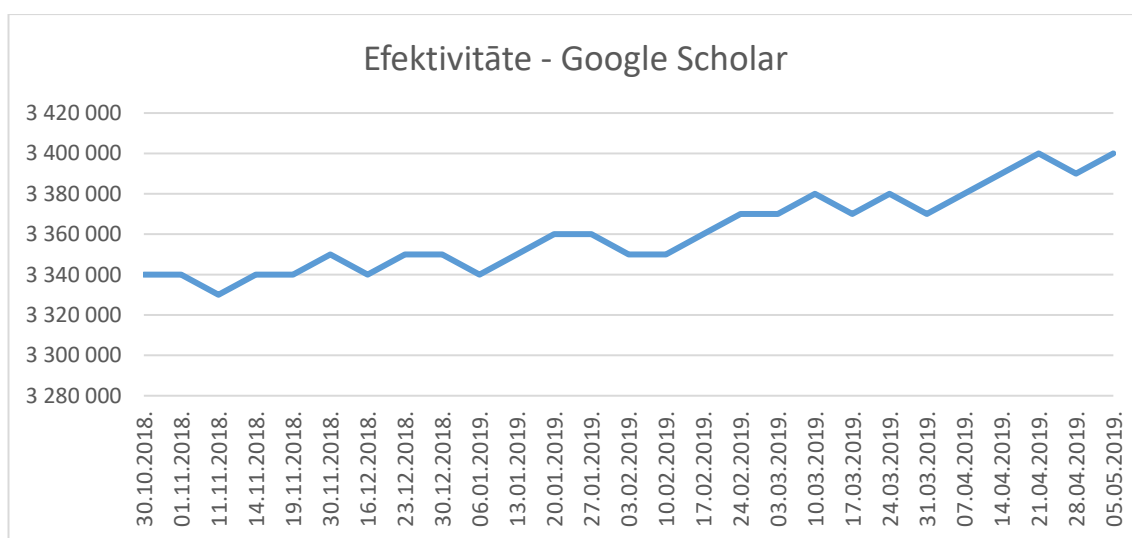
Šādu metriku izmanto, lai noteiktu, cik daudz lietotāju pieļauj kļūdas. Formula šai metrikai ir dalījums lietotāju skaitam, kas pieļāvuši kļūdu, ar visu to lietotāju skaitu, kas uzdevumu veica, to visu reizinot ar 100%.

Piemēram, ja uzdevumā četri no 10 lietotājiem kļūdījās, tad kļūdu intensitāte ir 40%, kas nozīmē, ka 60% lietotāju var veiksmīgi izpildīt uzdevumu.

3. EFEKTIVITĀTE

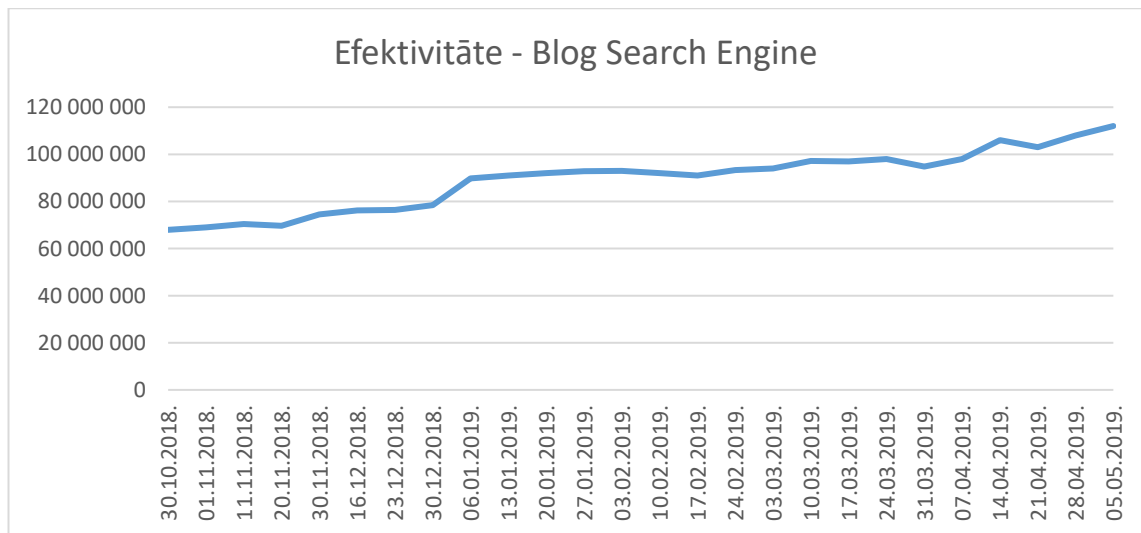
Gan ISTQB glosārijā [3], gan ISO/IEC 25010 standartā [1], gan ISO/IEC 25022 standartā [2] minētā informācija sakrīt, ka efektivitāte ir izlietotie resursi saistībā ar to, cik lielā mērā lietotāji sasniedz noteiktus mērķus. Ne tikai laiks, bet arī izmaksas vai veselība, vai nepieciešamie materiāli ir resursi, kas var būt nepieciešami uzdevuma izpildei un mērķu sasniegšanai.

Efektivitāte visbiežāk tiek mērīta uzdevuma izpildei nepieciešamā laika ziņā. Tas ir laiks sekundēs vai minūtēs, kas nepieciešams, lai veiksmīgi izpildītu uzdevumu. Laika posms uzdevuma veikšanai var tikt aprēķināts, atņemot no uzdevuma beigu laika sākuma laiku, iegūstot precīzu laiku, ko lietotājs ir patērējis.



3.1. att. Google Scholar vietnes meklēšanas rezultāti laikā frāzei - "Efficiency" software testing

Tā kā efektivitāte ir termins, kas pastāv jau ilgu laiku, tad atrasto ierakstu skaits ir mērāms miljonos arī starp zinātniskajiem rakstiem. Google Scholar vietnē [4] 30. oktobrī 2018. gadā bija aptuveni 3,33 miljoni ierakstu, taču 2019. gada 5. maijā ir bijuši tikai par 1,8 % rakstiem vairāk, kas redzams attēlā 3.1. Nelielais pieaugums skaidrojams ar to, ka efektivitāte ir plaši pētīta un lietota testēšanā jau senāk, kas nozīmē, ka liela daļa no efektivitātes testēšanas metodēm un tehnikām jau ir aprakstītas līdz šim.



3.2. att. Blog Search Engine vietnes meklēšanas rezultāti laikā frāzei - "Efficiency" software testing

Blog Search Engine vietnē [5] 30. oktobrī 2018. gadā bija gandrīz 70 tūkstoši ierakstu, bet 2019. gada 5. maijā šis skaits ir sasniedzis 112 tūkstošus. Pa šo laiku pieaugums ir nepilni 40 %. Attēlā 3.2. ir redzama līkne, kā pieaudzis atrasto rakstu skaits, kurā redzams, ka ap gadu miju rakstu skaits ir pieaudzis straujāk nekā pārējā periodā, kas liecina, ka šajā laikā cilvēku interese par efektivitātes testēšanu ir bijusi lielākā kā citā laikā

Turpmākajās apakšnodaļās tiek aprakstītas dažādas efektivitātes metrikas, kā noteikt sistēmas efektivitāti un kādus rādītājus tam nepieciešams iegūt.

3.1. Uzdevuma laiks

Viena no vienkāršākajām efektivitātes metrikām, kas tiek aprakstītas ISO/IEC 25022 standartā [2], ir uzdevuma laiks. Tiek veikts mērījums, cik ilgi konkrēto uzdevumu lietotājs pilda, līdz uzdevumu pabeidz ar veiksmīgu gala iznākumu.

Veiksmes kritērijus nosaka katram uzdevumam atsevišķi, atkarībā no mērķa, kuru ir iecere sasniegt, izpildot konkrētu uzdevumu.

Bieži problēmas rada meklētāji, ja tie nav veidoti, ņemot vērā lietotāju vēlmi uzdevumu veikt pēc iespējas ātrāk. Veicot vairāku Latvijā pieejamo sistēmu izpēti, ir redzams, ka eksistē tādi interneta veikali, kuros nav iespējams preces kārtot pēc dažādiem kritērijiem vai veikt filtrēšanu. Tas vairākkārtīgi paaugstina laiku, kas nepieciešams, lai atrastu precīzi, ar dažādiem ierobežojumiem, piemēram, nepieciešama galda lampa, kura cenas ziņā nepārsniegtu EUR 10.00, kā arī krāsa būtu balta. Ja ir nepieciešams caurskatīt visas pieejamās preces lampu kategorijā, tas aizņem krietni ilgāku laiku, nekā, ja ir iespējams filtrā atzīmēt sev vēlamos parametrus. Kā labs piemērs Latvijas tirgū ir interneta veikals *1a.lv* [16], kur dažādām precēm ir dažādi atlasē filtru, kas ļoti īsā laika posmā ļauj atrast sev interesējošās preces, turpretī

neveiksmīgs piemērs ir interneta veikals *cenuklubs.lv* [17], kurā atlasas filtri ir ļoti minimāli, kārtošanu nav iespējams veikt, kas ļoti paildzina preces atrašanu, dažkārt novedot pie cita veikala meklējumiem.

3.2. Laika efektivitāte

Tā ir tāda veida efektivitāte, kas nosaka, ar kādu efektivitātes līmeni vai pakāpi lietotāji sasniedz savus mērķus sistēmas lietošanas laikā. Efektivitāte pieaug, pieaugot rezultativitātei un samazinoties nepieciešamajam laikam uzdevuma veikšanai. Ja uzdevumi ir ar dažādu grūtības pakāpi, tad būtu nepieciešams tiem pievienot koeficientus atkarībā no tās.

Laika efektivitāti mēra kā sasniegtos mērķus attiecībā pret laiku. Piemēram, lietotājs uzdevumu pilda divas minūtes un sasniedz piecus mērķus, tad šādā gadījumā laika efektivitāte ir pieci mērķi divās minūtēs [2].

Šos rezultātus ir iespējams viegli salīdzināt, jo galā ir iegūstamas proporcijas, kuras var matemātiski salīdzināt, nosakot dažādu uzdevumu grupējumu efektivitāti.

Šī veida efektivitātes aprēķināšanai izmantojama arī komplicētāka formula (3.1), ja nepieciešams mērīt laika efektivitāti vairākiem lietotājiem ar vairākiem uzdevumiem:

$$Uz\ laiku\ balstītā\ efektivitāte = \frac{\sum_{j=1}^R \sum_{i=1}^N \frac{n_{ij}}{t_{ij}}}{NR}, kur \quad (3.1)$$

- N apzīmē kopējo vēlamu uzdevu skaitu, piemēram, viens uzdevums,
- R apzīmē testa dalībnieku skaitu, piemēram, četri dalībnieki,
- n_{ij} apzīmē uzdevuma i rezultātu dalībniekam j,
- t_{ij} apzīmē laiku, kas nepieciešams dalībniekam j uzdevuma i izpildei [15].

Ja lietotājs j veiksmīgi izpilda uzdevumu i, tad N_{ij} ir viens, pretējā gadījumā N_{ij} ir nulle. Ja uzdevums nav ticis veiksmīgi izpildīts, tad laiks tiek mērīts līdz momentam, kad lietotājs uzdevumu pārtrauc.

Piemērā, kur pirmais lietotājs paveic uzdevumu vienā sekundē, otrais lietotājs divās, bet trešais lietotājs trijās sekundēs, savukārt ceturtais lietotājs pilda uzdevumu sešas sekundes un beigās vispār neizpilda uzdevumu, sanāk, ka uz laiku balstīto efektivitāti var aprēķināt šādi:

$$Uz\ laiku\ balstītā\ efektivitāte = \frac{\left(\frac{1}{1} + \frac{1}{2} + \frac{1}{3} + \frac{0}{6}\right)}{1 \cdot 4} = 0,46\ mērķi/sekundē \quad (3.2)$$

Pēc formulas (3.2) var sacīt, ka sistēmas efektivitāte ir 0,46 mērķi vienas sekundes laikā. Šīs pašas darbības var veikt atkārtoti pēc nākamā programmatūras izstrādes cikla [18] un salīdzināt rezultātus, vai ir tikuši veikti uzlabojumi vai ir notikusi sistēmas regresija.

3.3. Izmaksu lietderība

Ne vienmēr par izmaksām uzskata laiku. Vispārīgā gadījumā izmaksu lietderību aprēķina kā kopējās izmaksas uzdevuma veikšanai pret sasniegtajiem uzdevuma mērķiem.

Finansiālās izmaksas ieņem lielu lomu biznesa vadībā, tādā gadījumā, ja par izmaksām uztver naudas līdzekļus, šī metrika ir līdzīga laika efektivitātei. Tad formula ir apgriezti proporcionāla laika efektivitātes formulai, jo tiek dalītas kopējās izmaksas ar sasniegtajiem mērķiem, iegūstot proporciju, kas nosaka, cik naudas nepieciešams viena uzdevuma veiksmīgai izpildei.

Tāpat kā citās metrikās, arī šajā var pielietot dažādas grūtības pakāpes uzdevumus, tikai tādā gadījumā jāpiešķir koeficienti katram uzdevumam, tos izvērtējot.

3.4. Produktīvā laika attiecība

Ne viss laiks, ko lietotājs pavada uzdevuma izpildei, ir produktīvs, piemēram, gadījumos, kad nav skaidrs, kura poga spiežama, ja nav uzvedinoši attēli vai teksti vai arī tie ir maldinoši.

Balstoties uz ISO/IEC 25022 standartu [2], produktīvā laika attiecība ir laika, kas lietotājam nepieciešams produktīvu darbību veikšanai, attiecība pret visu patērēto laiku uzdevuma veikšanai.

Par neproduktīvu laiku uzskatāms, piemēram, palīdzības meklēšana dažādu neskaidrību gadījumos vai atgūšanās no kļūdām, taču neietverot obligāti veicamās aktivitātes uzdevuma veiksmīgai izpildei.

3.5. Nevajadzīgās darbības

Šī efektivitātes mērīšanas metode tiek izdalīta atsevišķi no produktīvā laika attiecības tādēļ, ka nevajadzīgās darbības ir vieglāk izmērīt, kaut gan ļoti līdzinās produktīvā laika attiecības metodei.

Šajā gadījumā tiek mērīta attiecība nevajadzīgajām darbībām pret visu veikto darbību skaitu, lai lietotājs veiktu uzdevumu.

Vislabāk šo metodi ir izmantot gadījumos, kad uzdevums ir izpildāms, izdarot izvēles – ar datora peli, pieskaroties vai ar balss komandām.

3.6. Noguruma sekas

Šādu aspektu nepieciešams mērīt tādēļ, ka cilvēki pēc ilgstošas uzdevumu pildīšanas ir nogurušāki, kas izraisa dažādas neuzmanības kļūdas. Mērīts tiek pirmās uzdevuma veikšanas reizes sniegums, kas tiek uzskatīts par atskaites punktu, kā arī vēlāko uzdevumu pildīšanas reižu sniegums. Teorētiski pēc vairāku reižu pildīšanas sniegums samazinās noguruma dēļ.

Šis rādītājs tiek aprēķināts pēc formulas:

$$\text{Noguruma sekas} = 1 - \frac{\text{Pašreizējais sniegums}}{\text{Sniegums 1. pildīšanas reizē}} \quad (3.3)$$

Jo šis rādītājs ir tuvāk nullei, jo izstrādātajai sistēmai ir lielāka efektivitāte. Bet jāņem vērā, ka šo metriku vislabāk būtu pielietot, izmantojot pieredzējušus lietotājus rādītāja aprēķināšanai nepieciešamo datu ieguvei, jo, ja sistēma ir lietojama ilgstoši, visticamāk, ka lietotāji to būs ļoti labi apguvuši, piemēram, kasieri, administratori un citi tāda veida speciālisti, kas vairākas stundas dienā sistēmu lieto.

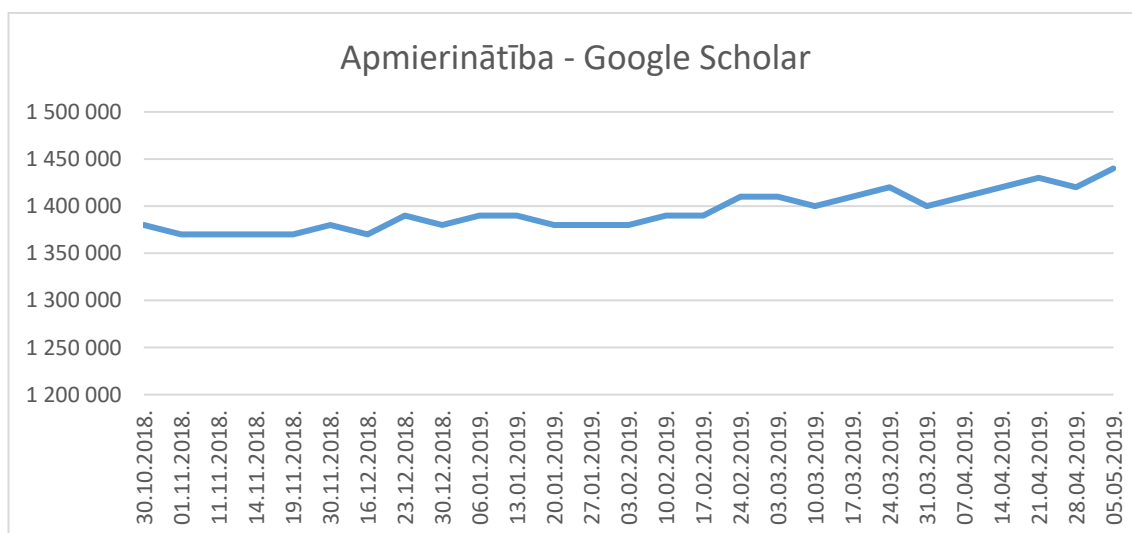
4. APMIERINĀTĪBA

Apmierinātība ir pakāpe vai apmērs, kurā lietotāja vēlmes un nepieciešamības ir apmierinātas, izmantojot sistēmu konkrētā lietošanas kontekstā. Visbiežāk to mēra, aptaujājot lietotājus, lai uzzinātu, cik lielā mērā viņu vajadzības un vēlmes sistēma apmierina [2].

Anketēšanas metode ir pielietojama visās turpmākajās apakšnodaļās, kurās tiek sīkāk aplūkoti dažādi apmierinātības aspekti.

Anketas cilvēkiem reti patīk pildīt cītīgi, tādēļ tā ir viena no lielākajām problēmām, ar ko saskaras testētāji. Dati, ko sniedz respondenti, var būt nekorekti emociju dēļ, kas bieži ir slinkums pildīt aptaujas vai nevēlēšanās koncentrēties uz lietām, kas cilvēkiem šķiet, ka nedod tūlītēju labumu.

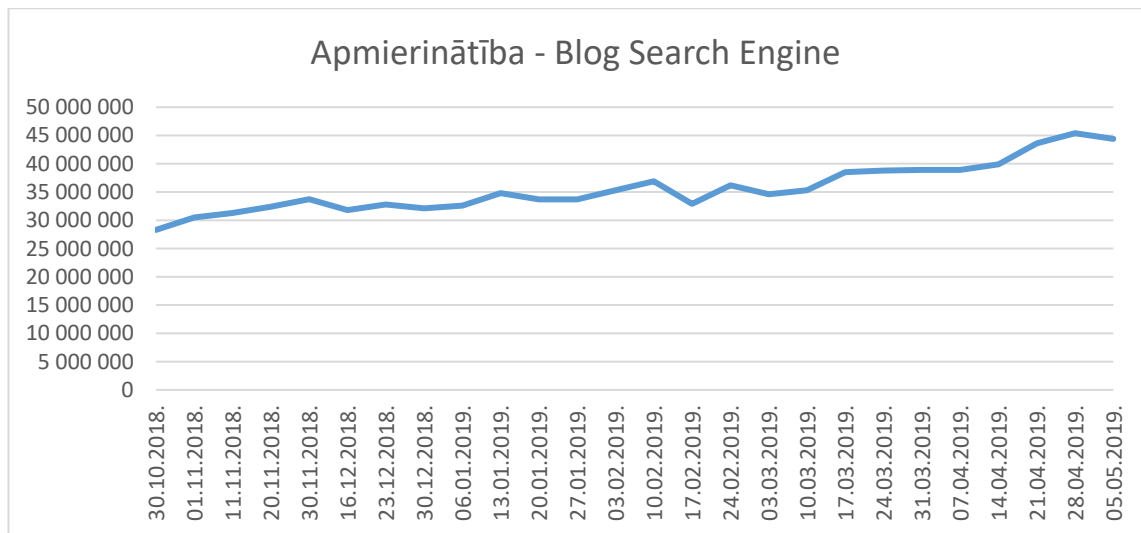
Katrs cilvēks ir individuāls, katrs cilvēks spriež pēc saviem principiem, tādēļ nereti gadās, ka anketas ir pilnīgi pretrunīgas. Šādos gadījumos ieteicams ir neiekļaut statistikas datu apkopošanā labāko un sliktāko vērtējumu, lai iegūtu vispārīgāku priekšstatu par lietotājiem kā vienu veselumu.



4.1. att. Google Scholar vietnes meklēšanas rezultāti laikā frāzei -
"Satisfaction" software testing

Google Scholar vietnē [4] 30. oktobrī 2018. gadā bija 1,38 miljoni rakstu, koros minēta apmierinātība programmatūras testēšanas kontekstā ierakstu, taču 2019. gada 5. maijā ir bijuši 1,44 miljoni ierakstu. Pa šo laiku pieaugums ir bijis aptuveni 4,2 %.

Attēlā 4.1. ir redzamajā līknē nav vērojami strauju kāpumu vai ilgstošu nemainības periodu, jo apmierinātība ir viena no galvenajām lietām, kas tiek testēta un ir zināma jau vairākus gadus.



4.2. att. **Blog Search Engine vietnes meklēšanas rezultāti laikā frāzei - "Satisfaction" software testing**

Blog Search Engine vietnē [5] 30. oktobrī 2018. gadā bija gandrīz 30 tūkstoši rakstu, taču 2019. gada 5. maijā ir bijuši jau nepilni 45 tūkstoši rakstu. Šajā laika periodā rakstu skaita pieaugums ir bijis aptuveni 36%. Attēlā 4.2. ir redzama līkne, kā pieaudzis atrasto rakstu skaits, kas ir līdzīga no dinamikas viedokļa kā zinātnisko avotu vidū, kā redzams attēlā 4.1. tas liecina, ka gan zinātniskajās aprindās, gan ikdienas dzīvē cilvēkiem ir nemainīga interese par šo lietojumkvalitātes atribūtu

4.1. Vispārējā apmierinātība

ISO/IEC 25022 standartā [2] minēts, ka var mērīt vispārējo apmierinātību, tomēr šī vērtēšanas metode ir ļoti vispārīga, jo balstās uz lietotāja subjektīvo viedokli. Lai šo rādītāju noteiktu, tiek aptaujāti lietotāji un apkopoti rezultāti, cik kopumā apmierināti tie jūtas, lietojot sistēmu.

Lai noteiktu lietotāju apmierinātību ar sistēmu, parasti tiek pielietota anketēšanas metode. Anketu veidi ir dažādi gan pēc satura, gan laika, kad anketēšana tiek veikta. Lai novērtētu apmierinātību, retos gadījumos tiek veikti mērījumi, apskatot dažādus apmierinātības aspektus sīkāk, biežāk lietotāju apmierinātība ar sistēmu tiek vērtēta kopumā.

Bieži vien tiek izmantota skala no 0 līdz 100, kur 0 ir pavisam neapmierinoši, bet 100 izcili. Lietotājs izvēlas, cik apmierinoša sistēma viņam šķiet atbilstoši saviem standartiem.

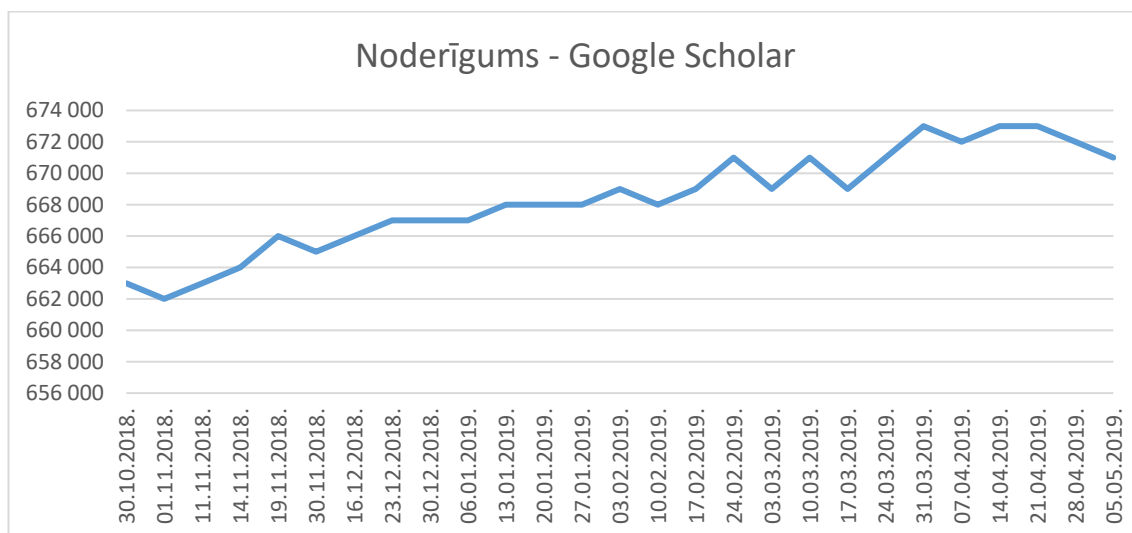
Lai precīzāk mērītu apmierinātību, tā tiek iedalīta sīkākās kategorijās, kuras izmērīt ir vienkāršāk, nekā noteikt vispārējo apmierinātību. Turpmākajās nodaļās tiek aprakstīts, kā novērtēt lietotāju apmierinātību un kādus rādītājus nepieciešams iegūt, lai to noteiktu.

4.2. Noderīgums

Noderīguma mērījumi ļauj novērtēt, cik lielā mērā lietotājs ir apmierināts ar pragmatisko mērķu sasniegšanu, ieskaitot arī lietošanas rezultātus un sekas [2].

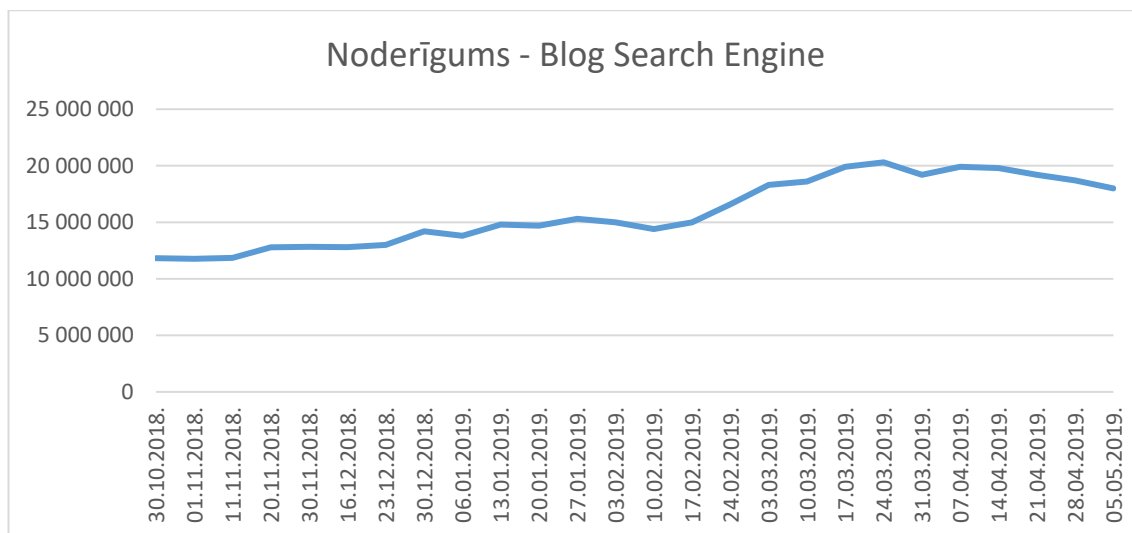
Sistēmai ir jāsniedz lietotājam kāda vērtība. Ne tikai zināšanas, bet arī izklaide var būt vērtība pēc garas darbadienas. Dažādas lietotnes ir paredzētas, lai atvieglotu cilvēku ikdienu, piemēram, iepirkumu saraksts mobilajā viedtālrunī vai izmaksu kontroles lietotnes, kas palīdz plānot savus izdevumus, neļaujot pārtērēt naudas līdzekļus uz lietām, kuras nemaz nav nepieciešamas.

Dažkārt mēdz būt gadījumi, kad noteiktu sistēmas funkcionalitāti nemaz nevar sasniegt, tādēļ pirms noderīguma lietotājiem testēšanas būtu ieteicams veikt koda pārklājuma testēšanu, kad tiktu izmantota analīzes metode, lai noteiktu, kuras programmatūras daļas ir tikušas izpildītas jeb pārklātas konkrētā testa plānā un kuras daļas vai nav vispār tikušas izpildītas [3]. Tas var liecināt, ka lietotājiem šāda funkcionalitāte nemaz nebūs pieejama, tādēļ to neviens nevarēs lietot.



4.3. att. Google Scholar vietnes meklēšanas rezultāti laikā frāzei -
"Usefulness" software testing

Google Scholar vietnē [4] 30. oktobrī 2018. gadā bija aptuveni 663 000 rakstu, taču 2019. gada 5. maijā ierakstu skaits pat nesasniedz 1,2 % pieaugumu. Kā redzams attēlā 4.3., šis termins ir bijis populārs un aprakstīts jau pirms bakalaura darba uzsākšanas, jo līdzīgi kā iepriekš apskatītie lietojumkvalitātes atribūti, kā rezultativitāte, efektivitāte un apmierinātība, arī noderīgums ir viens no testēšanas aspektiem, kas zināms jau ļoti ilgu laika periodu.



4.4. att. Blog Search Engine vietnes meklēšanas rezultāti laikā frāzei - "Usefulness" software testing

Blog Search Engine vietnē [5] 30. oktobrī 2018. gadā bija gandrīz 12 miljoni ierakstu, taču 2019. gada 5. maijā ir bijuši 18 miljoni ierakstu. Pa šo laiku pieaugums ir bijis nedaudz virs 34 %. Attēlā 4.4. ir redzama līkne vērojams, ka straujāku popularitātes kāpums ir bijis no marta sākuma līdz aprīļa vidum.

Turpmākajās apakšnodaļās tiek aprakstītas dažādas noderīguma metrikas, kā noteikt sistēmas noderīgumu un kādus rādītājus tam nepieciešams iegūt.

4.2.1. Apmierinātība ar iespējām

Viena no apmierinātības īpašībām, kas aprakstīta ISO/IEC 25022 standartā [2], ir apmierinātība ar iespējām. Ja lietotājam sistēmas sniegtās iespējas šķiet apmierinošas, tātad var secināt, ka sistēma ir noderīga lietotājam.

Vērtēšanas metode, kas tiek izmantota, lai to noteiktu, ir anketēšana, kur lietotājs sniedz atbildes uz jautājumiem, kas ir saistīti ar konkrētām iespējām sistēmā. Parasti tā ir tāda veida anketa, kurā tiek izmantota Likerta skala, kas ir balsfīta uz intervāliem, kur sākuma punkts ir, piemēram, "pilnībā nepiekrīt" vai "pilnīgi neapmierinoši", bet beigu punkts intervālā ir "pilnībā piekrīt" vai "pilnīgi apmierinoši".

Tāpat kā citās lietojumkvalitātes mērīšanas metodēs, arī šai metodei iespējams pievienot jautājumiem svaru, ja kāds no tiem ir svarīgāks par citiem [2].

4.2.2. Diskrētā lietošana

Lai notestētu, vai sistēmā ieviestā funkcionalitāte ir noderīga lietotājiem, tiek mērīta proporcija lietotāju skaitam, kas lieto konkrētu īpašību, pret potenciālo lietotāju skaitu, kas varētu lietot konkrēto īpašību.

Gadījumos, kad tiek iegūta proporcija, kas tuvojās nullei, ir vērts padomāt, vai šāda funkcionalitāte sistēmā vispār ir nepieciešama, ja to praktiski neviens nelieto. Šādā veidā nebūtu jāuztur tādas sistēmas daļas, kuras tikpat kā neviens neizmanto, ietaupot resursus.

Veicot augstāk aprakstītās darbības, tiek noteikta lietotāju uzvedība, kas var noderēt tālākā izstrādes procesā, lai tiekotos maksimāli tuvu kvalitatīvas sistēmas izveidei [2].

4.2.3. Funkciju izmantošana

Funkciju izmantošanu mēra, nosakot attiecību starp lietotāju skaitu, kas izmanto specifisku funkciju vai īpašību, un lietotāju skaitu noteiktā sistēmas lietotāju grupā. Lietotāji šīs metrikas gadījumā tiek iedalīti noteiktās grupās pēc līdzības pazīmēm vai veicamajām darbībām.

Funkcijas var definēt dažādos detalizācijas līmeņos – sākot ar atsevišķām funkcijām, beidzot ar sistēmas apakškopu

Zema iegūtā vērtība varētu liecināt par to, ka funkcija nav lietderīga vai ir noderīga tikai kādai konkrētai sistēmas lietotāju apakšgrupai, kā arī tas varētu norādīt uz to, ka lietotāji nesaprot, kā funkciju izmantot vai nezina, ka tāda eksistē [2].

4.2.4. Neapmierināto lietotāju daļa

Būtiski ir veikt testēšanu, lai noteiktu proporciju, cik lietotāji no kopējā skaita ir sūdzējušies par sistēmas darbību. Šo metodi var izmantot, lai noteiktu, cik liela daļa ir neapmierināta un sūdzas par kādu konkrētu funkciju no visiem tiek, kas ir sūdzējušies par funkcijām kopumā.

ISO/IEC 25022 standartā [2] tiek aplūkots arī šīs metodes paplašināts variants, veicot mērījumus par konkrētām funkcijām sistēmā, nevis visām kopumā. Piemēram, ja pieci cilvēki sūdzējušies par vienu un to pašu funkciju un kopā ir 20 cilvēku sūdzējušies par jebkādām funkcijām sistēmā, tad proporcija tiek veidota kā 5 pret 20 jeb ceturtdaļa ir neapmierināta ar konkrēto funkciju no tiem, kas ir neapmierināti.

4.3. Uzticēšanās

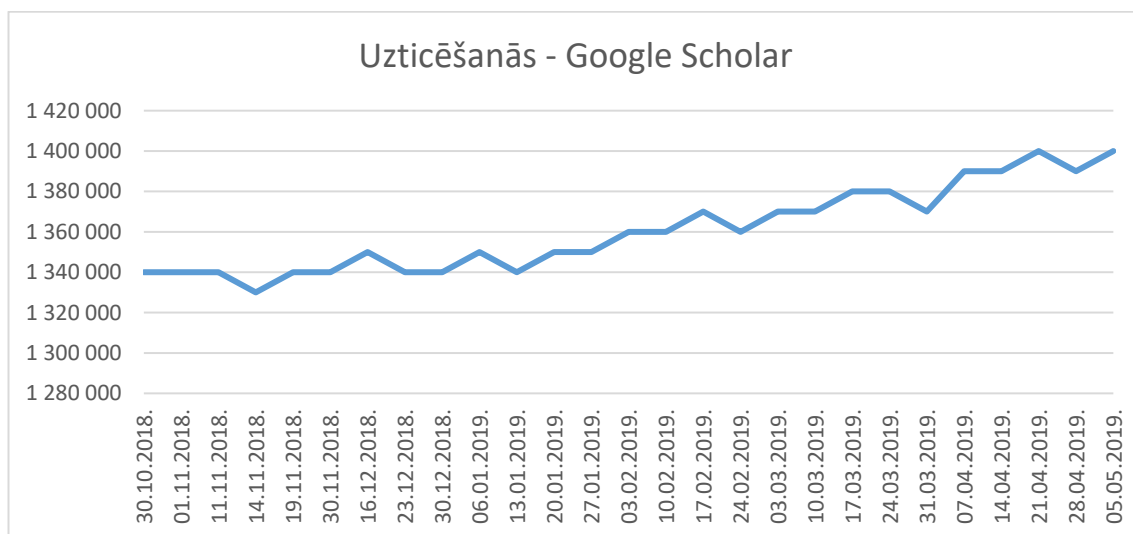
Bieži vien ir svarīgi, lai lietotājs var paļauties uz to rezultātu, ko sistēma sniedz, taču pat viena kļūda gadā, kas šķiet rets gadījums, bet lietotājam izraisa kritiskas sekas, var novest pie tā, ka lietotājs sistēmai vairs neuzticēsies.

Tā kā izstrādātājiem var būt problēmas noteikt, vai lietotājs sistēmai uzticas, jo viņi zina lielu daļu no funkcionalitātes un veidiem, kā tā darbojas, tādēļ ir nepieciešams testēt arī šo programmatūras aspektu.

Lietotāju uzticēšanās mērījumi nosaka pakāpi, kurā lietotājs vai cita ieinteresētā persona ir pārliecināta, ka produkts vai sistēma uzvedīsies kā parasti. Lai to noteiktu, tiek uzdoti dažādi psiholoģiska rakstura jautājumi lietotājiem, no kā var izsecināt par pakāpi vai līmeni, kādā lietotājs uzticas sistēmai [2].

Šie ir diezgan grūti salīdzināmi lielumi, jo katrs lietotājs ir personība ar savu viedokli un attieksmi pret lietām. Palielinot jautājumu skaitu un iespējamus atbilžu variantus, var paaugstināt mērījumu precizitāti, taču tādā gadījumā lietotājiem aptaujas var šķist par garu un nogurdinošu, kas novestu pie ne visai precīziem mērījumiem. Tāpat arī ir nozīme dienas laika izvēlei, jo agros rītos vai vēlos vakaros koncentrēšanās spējas ir daudz zemākas, tāpēc ieteicams izvērtēt, kad mērījums veikt, lai tie būtu precīzāki.

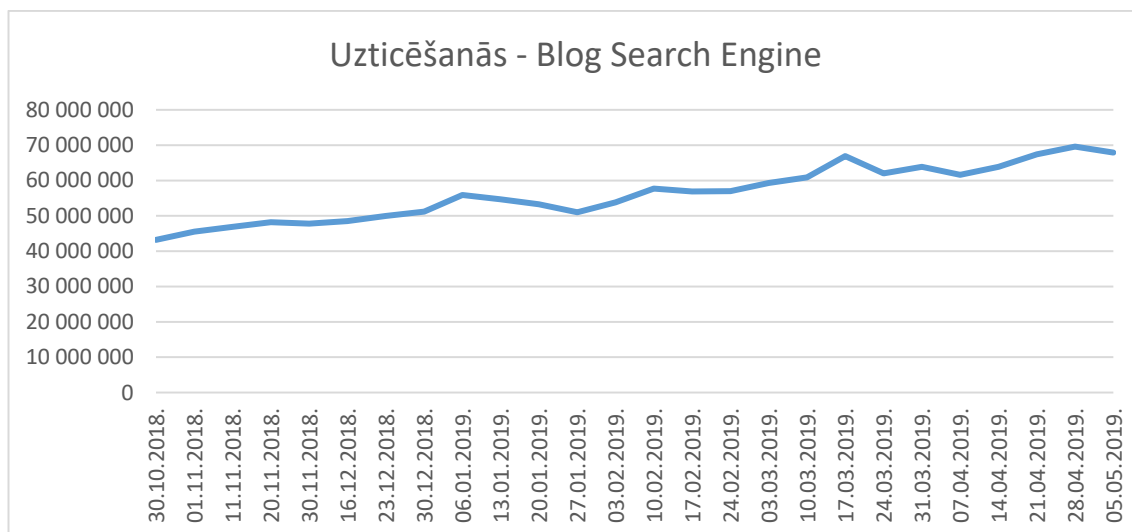
Ne tikai kļūdas var ietekmēt lietotāju uzticēšanos sistēmai, bet arī dažādi papildus paziņojumi, kas attur lietotāju no netīšu darbību veikšanas. Piemēram, lietotājs netīšām nospiež kādu pogu, kas izsauc datu dzēšanu, tad būtu labi, ja šādā situācijā tiktu vēlreiz pārvaicāts, vai tiešām lietotājs vēlas šo darbību veikt, jo sistēmas uzdevums ir atvieglot un palīdzēt lietotājam veikt kādas darbības. Tas rosinās vēlmi uzticēties sistēmai, jo lietotājs jutīsies pasargāts no dažādu nevēlamu darbību izpildes.



4.5. att. Google Scholar vietnes meklēšanas rezultāti laikā frāzei - "Trust" software testing

Google Scholar vietnē [4] 30. oktobrī 2018. gadā bija apmēram 1 340 000 ierakstu, taču 2019. gada 5. maijā atrasto rezultātu skaits pieaudzis līdz aptuveni 1 400 000, kas redzams attēlā 4.5. Tas nozīmē, ka top aizvien jauni raksti par šo tēmu. Tā kā uzticēšanos bieži jauc ar uzticamību, tad tas varētu būt iemesls, kādēļ rakstu skaits ir pieaudzis par vairāk kā 4 %, jo cilvēkiem ir vēlme ieviest skaidrību arī šobrīd starp līdzīgajiem terminiem.

Tomēr, pētot Google Scholar [4] vietnē rakstu skaitu noteiktos gados, ir redzams, ka pēdējo desmit gadu laikā 2013. un 2014. gadā ir tapuši visvairāk rakstu, bet pēc tam rakstu skaita pieaugumam ar gadiem ir tendence sarukt.



4.6. att. **Blog Search Engine vietnes meklēšanas rezultāti laikā frāzei - "Trust" software testing**

Blog Search Engine vietnē [5] 30. oktobrī 2018. gadā bija nedaudz virs 43 miljoniem rakstu, taču 2019. gada 5. maijā ir bijuši par 36% vairāk rakstu, kas ir tuvu 70 miljoniem. Attēlā 4.6. ir redzama līkne, kā pieaudzis atrasto rakstu skaits. Nav novērojamas straujas svārstības līknē, kā arī pieaugums ir bijis pārsvarā vienmērīgs visā apskatītajā laika periodā, tātad cilvēkiem interese ir vienmērīga saistībā ar uzticēšanās testēšanu.

4.4. Patika

Mūsdienās daudzas sistēmas tiek lietotas viedtālruņos, kuros ir pieejamas dažādas alternatīvas, tādēļ lietotāju noturēšanai ir nepieciešams sistēmu izstrādāt skaistu, pievilcīgu un lietotājam patīkamu.

Ja lietotājs neredz kādas pogas vai ir teksti norauti, visticamāk, ka lietotājs izvēlēsies citu alternatīvu, lai apmierinātu savas vajadzības un vēlmes. Kritiski tas ir uz mazajiem ekrāniem, jo cilvēki ir pieraduši, ka viss viedtālruņos notiek uzreiz un tūlīt. Ja kādas darbības veikšana aizņem ilgāku laiku, tad lietotājam ir patīkamāk, ja viņš redz kādu indikatoru, ka darbība tiek veikta, nevis sistēma ir pārtraukusi darboties.

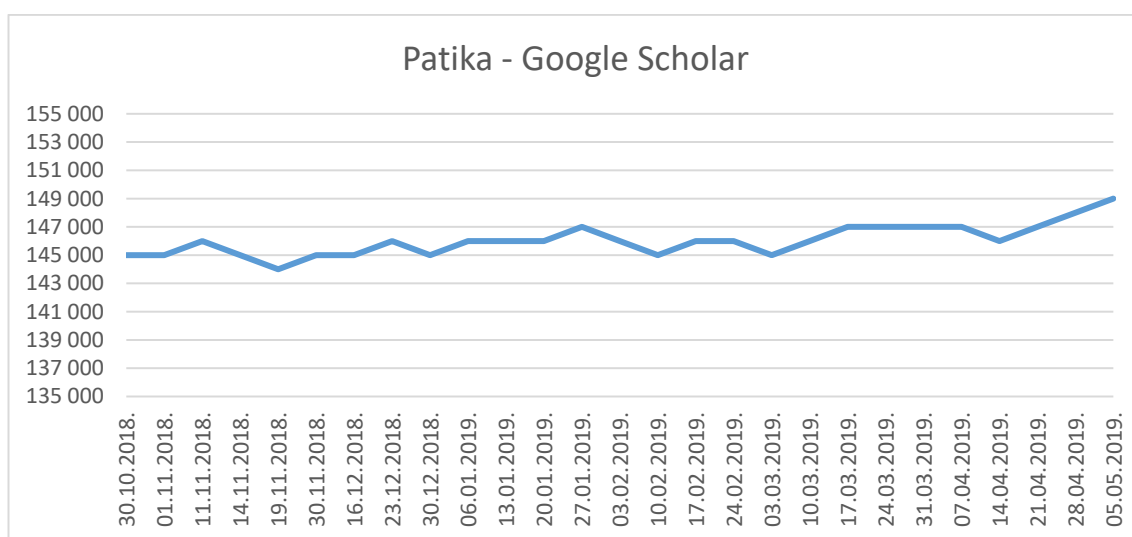
Šādu iemeslu dēļ nepieciešams veikt testēšanu uz dažādu izmēru ekrāniem un viedtālruņu modeļiem, lai novērtētu, kā sistēma pielāgojas atšķirīgu izmēru ekrāniem un kā ir savietojama ar dažādām operētājsistēmām [19].

Patikas mērījumi nosaka pakāpi, kādā lietotāju vēlmes un vajadzības ir apmierinātas.

Lietotāju vajadzības var ietvert arī jaunu zināšanu un prasmju ieguvu, patīkamu mirkļu atsaukšanu atmiņā, kā arī citas lietotājam būtiskas lietas, kas veicina patiku ar sistēmu.

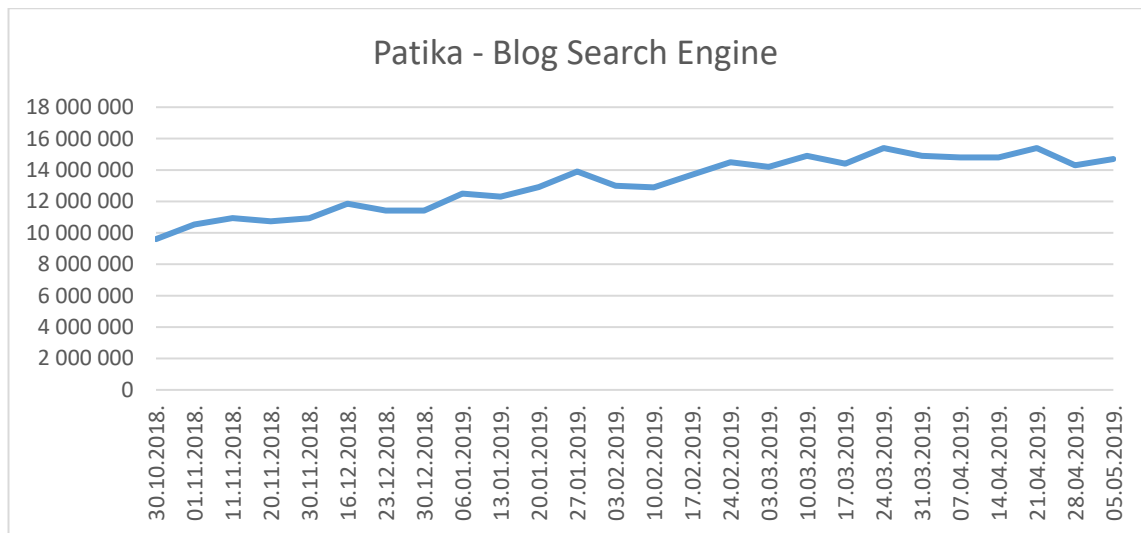
Līdzīgi kā uzticēšanās mērīšanai tiek izmantota psiholoģiska rakstura anketa, kur lietotājiem ir iespēja atzīmēt, cik lielā mērā sistēma lietotāju apmierina. Visbiežāk tā ir skala, sākot ar “pavisam nepatīk” un beidzot ar “ļoti patīk” [2].

Viena no testēšanas metodēm, kā novērtēt lietotāju apmierinātību ar sistēmu ir pārtraukumu testēšana [20]. Pārtraukumi var rasties dažādi, bet lietotājs nebūs apmierināts, ja viņa ievadītie dati būs pazuduši pēc nelieliem pārtraukumiem vai pēkšņi kāda informācija vispār būs pazudusi no redzesloka. Izmantojot mobilos datus bieži ir tīkla pārrāvumi, kas pat var ilgt tikai dažas sekundes daļas, tāpēc ir jāpārlicinās, ka pie nelieliem tīkla traucējumiem lietotājs to nemanīs vai būs redzams kāds atbilstošs paziņojums par problēmām ar tīklu, ja pārrāvums ilgst lielāku laika periodu. Iespēju robežās ir jācenšas dati saglabāt.



4.7. att. Google Scholar vietnes meklēšanas rezultāti laikā frāzei - "Pleasure" software testing

Google Scholar vietnē [4] 30. oktobrī 2018. gadā bija 145 tūkstoši ierakstu, taču 2019. gada 5. maijā to ir bijis 149 tūkstoši. Šajā laika periodā pieaugums ir bijis nepilni 3 %. Attēlā 4.7. ir redzama līkne, kā pieaudzis atrasto rakstu skaits. Liela dinamika nav vērojama, jo patikas testēšanu mūsdienās zina vairums testētāju, kā arī tas tiek mācīts un apgūts universitātēs vai dažādos testēšanas semināros.



4.8. att. **Blog Search Engine vietnes meklēšanas rezultāti laikā frāzei - "Pleasure" software testing**

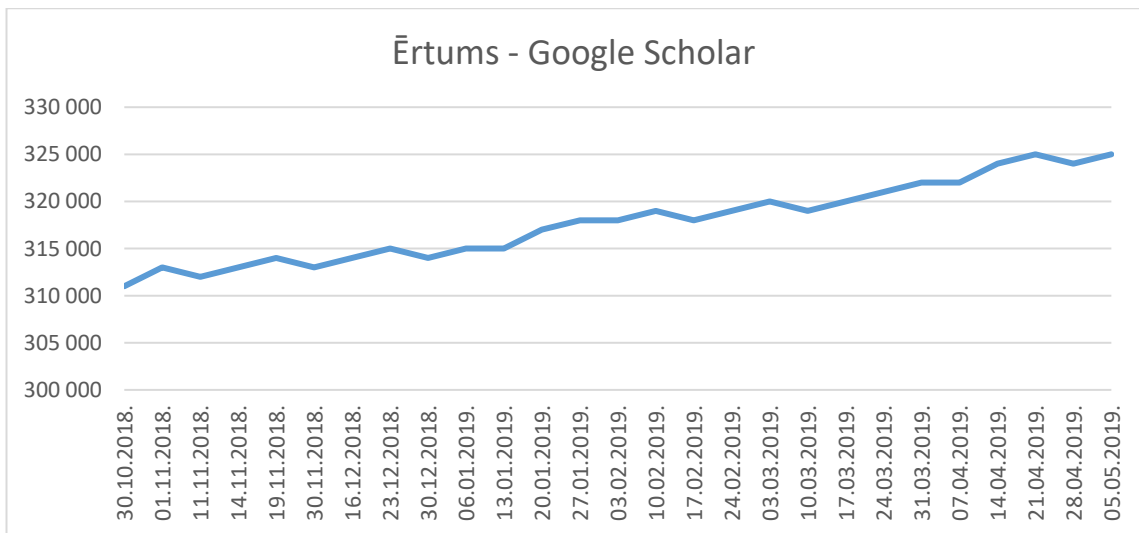
Blog Search Engine vietnē [5] 30. oktobrī 2018. gadā bija gandrīz tuvu 10 miljoniem rakstu, taču 2019. gada 5. maijā atrasto rakstu skaits ir bijis gandrīz 15 miljoni. Pa šo laiku pieaugums ir bijis apmēram 36 %. Attēlā 4.8. ir redzama līkne, kā pieaudzis atrasto rakstu skaits. Pieaugums un līdz ar to cilvēku interese par šo lietojumkvalitātes atribūta testēšanu nedaudz ir sarukusi no marta vidus.

4.5. Ērtums

Cits aspekts, pēc kā noteikt apmierinātību, ir ērtums jeb pakāpe, kādā lietotāju vajadzības pēc fiziska komforta ir apmierinātas. Būtisku ietekmi rada arī vide, kādā sistēma tiek lietota, kā arī gan lietotāja pozīcija, gan veicamās darbības [2].

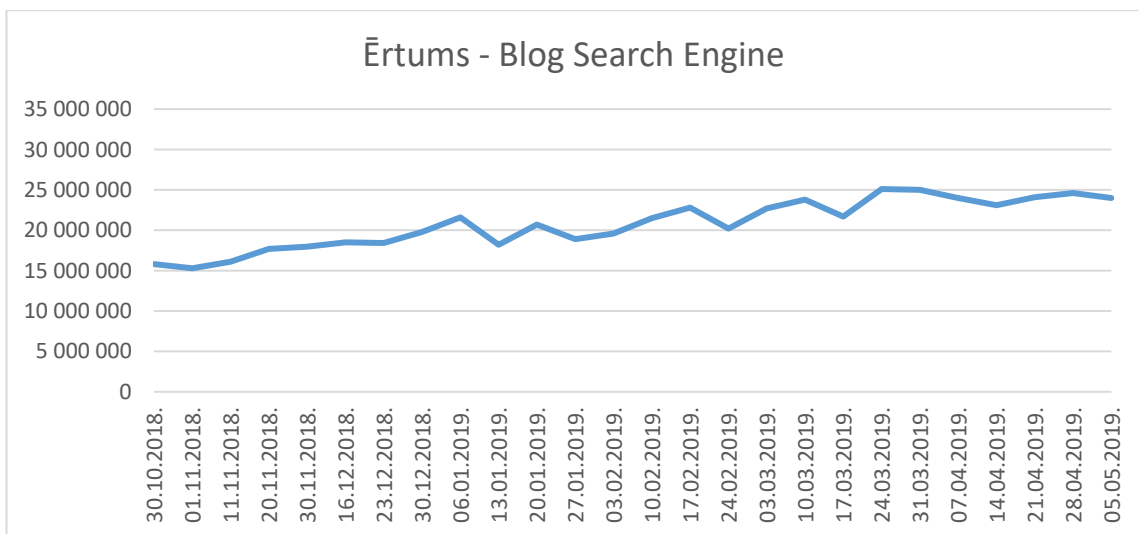
Ērtuma mērīšanai tiek izmantotas psiholoģiska rakstura aptaujas ar jautājumiem, kur atbilde jāizvēlas no skalas, cik daudz lietotājs izjūt konkrētas iezīmes, sistēmas lietošanas laikā vai arī pēc.

Ja burtiņi sistēmā ir tik sīki, ka lietotājam ir jāpieliecas tuvāk ekrānam, lai saskatītu tekstu, tad šī sistēma lietotājam var šķist neērta, it īpaši, ja tāda ir jālieto ilgstoši, saliecot kaklu nedabiskā pozā vai lieki piepūlējot acs muskuļus. Ja nepieciešams lielu daudzumu informācijas tomēr attēlot, tad vērts ir ieviest kādus dialoglogus vai ritjoslas, kas varētu ļaut burtu lielumu palielināt, taču samazināt aizņemto vietu, tādējādi padarot sistēmu lietotājam ne tikai patīkamāku pēc izskata, bet arī ērtāku lietošanai.



4.9. att. Google Scholar vietnes meklēšanas rezultāti laikā frāzei - "Comfort" software testing

Google Scholar vietnē [4] 30. oktobrī 2018. gadā bija aptuveni 310 tūkstoši ierakstu, taču 2019. gada 5. maijā ir bijuši 325 tūkstoši ierakstu. Pa šo laiku pieaugums ir bijis 4,3 %. Attēlā 4.9. ir redzama līkne, kā pieaudzis atrasto rakstu skaits.



4.10. att. Blog Search Engine vietnes meklēšanas rezultāti laikā frāzei - "Comfort" software testing

Blog Search Engine vietnē [5] 30. oktobrī 2018. gadā bija gandrīz 16 miljoni rakstu, taču 2019. gada ir bijuši par nepilniem 35% vairāk rakstu Attēlā 4.10. ir redzama līkne, kā pieaudzis atrasto rakstu skaits. Tāpat kā līkne attēlā 4.9., tās abas ir diezgan vienmērīgas, kas liecina, ka nav bijis straujš pieaugums cilvēku interesei ne zinātniskajos rakstos, ne bloga rakstos.

5. RISKBRĪVĪBA

Risks ir faktors, kas var rezultēties nākotnē ar negatīvām sekām. ISTQB glosārijā [3] ir aprakstīti dažāda veidi riski – produkta, kas ietekmē produkta kvalitāti, projekta, kas ietekmē projekta izdošanos, kvalitātes riski, kas tiešā veidā ir saistīti ar produkta kvalitātes ietekmi.

Lai mazinātu potenciālos draudus, tiek pielietota risku analīze, kas ir vispārīgs process risku identificēšanai un novērtēšanai. Risku kvalitatīvos un kvantitatīvos mērījumus, ko nosaka risku ietekme un iespējamība, var izteikt kā riska līmeņus.

Viena no testēšanas metodēm, ko piemin ISTQB organizācija, ir regresā testēšana, kas izmanto dažādas tehnikas, lai pārvaldītu regresijas riskus, piemēram, izstrādājot atkārtoti lietojamas testprogrammatūras un automatizējot testēšanas procesus vienā vai vairākos testēšanas līmeņos.

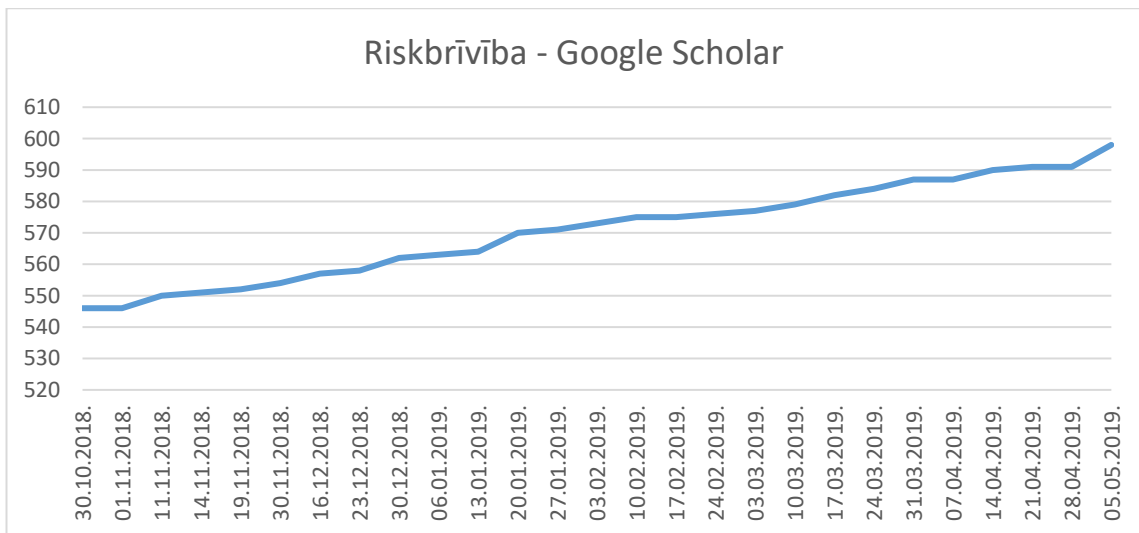
Cita metode ir riskbāzēta testēšana, kurā risku pārvaldība, prioritizēšana un testēšanas aktivitāšu, kā arī resursu lietošana ir balstīta uz atbilstošajiem riska tipiem un līmeņiem. Lai to būtu vieglāk veikt, tad riski tiek grupēti pēc līdzības faktora.

Lai izvairītos no riskiem, ir ieteicams jau sistēmas plānošanas brīdī veikt analītisko testēšanu, kuras pamatā ir sistemātiska analīze tādiem aspektiem kā, piemēram, produkta riskiem vai prasībām [3].

ISO/IEC 25022 standartā [2] aprakstīts, ka riskbrīvība nosaka pakāpi, kurā produkta vai sistēmas kvalitāte mazina vai novērš potenciālos riskus lietotājam, organizācijai vai projektam, iekļaujot ekonomiskā stāvokļa riskus, draudus dzīvībai un veselībai, kā arī apdraudējumu videi.

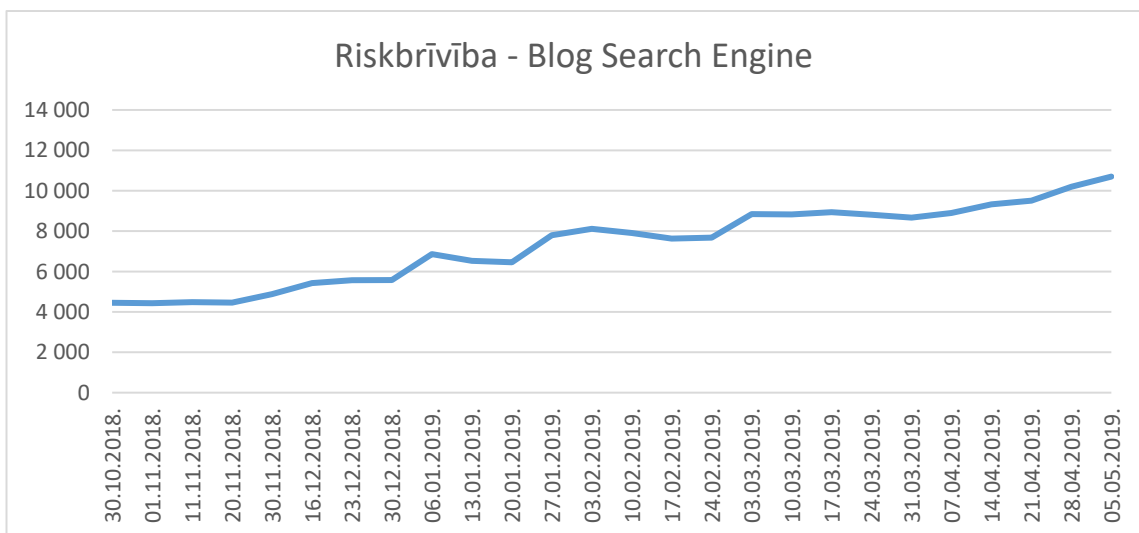
IT nozarē riskus bieži vien palielina integrācijas problēmas starp dažādām sistēmām, piemēram, versiju nesakritības izmantotajiem moduļiem klienta pusē ar izmantoto versiju izstrādes komandas pusē vai situācijas, kad tiek pievienotas jaunas ārējās sistēmas, ar kurām ir problēmas saintegrēties. Šādu iemeslu dēļ ieteicams ir veikt integrācijas testēšanu, kas ir tāda veida testēšana, lai atklātu defektus saskarnē un integrēto komponentu vai sistēmu mijiedarbībā [3].

Integrācijas testēšanai ir izmantojamas dažādas programmatūras vai testēšanas ietvari. Kā vienas no populārākajām programmatūrām ir *VectorCAST/C++*, *VectorCAST/Ada*, *Citrus*, *Integration Testing* un *LDRA, SMART INTEGRATION TEST ACCELERATOR (SITA)* [21]. Šajos testēšanas rīkos pieejamas arī cita veida testēšanas metodes, lai uzlabotu sistēmas kvalitāti.



5.1. att. Google Scholar vietnes meklēšanas rezultāti laikā frāzei - "Freedom from risk" software testing

Google Scholar vietnē [4] 30. oktobrī 2018. gadā bija nepilni 550 raksti, taču 2019. gada 5. maijā rakstu skaits ir bijis tuvu 600 robežai, kā redzams attēlā 5.1. Pieaugums ir bijis gandrīz 9 %, kas ir vairākas reizes lielāks kā iepriekš aplūkoto terminu pieaugums, ko varētu skaidrot ar faktu, ka riskbrīvība ir jaunāks jēdziens par tiem. Tā kā rakstu skaits ir bijis krietni mazāks, tad arī statistikas dati ir daudz precīzāki, jo meklēšanas vietne Google Scholar [4] veic atrasto ierakstu skaita noapaļošanu, ja tie pārsniedz 1 000 atrastos rakstus, atstājot redzamus pirmos trīs ciparus no skaitļa, bet atlikusī daļa ir nulles.



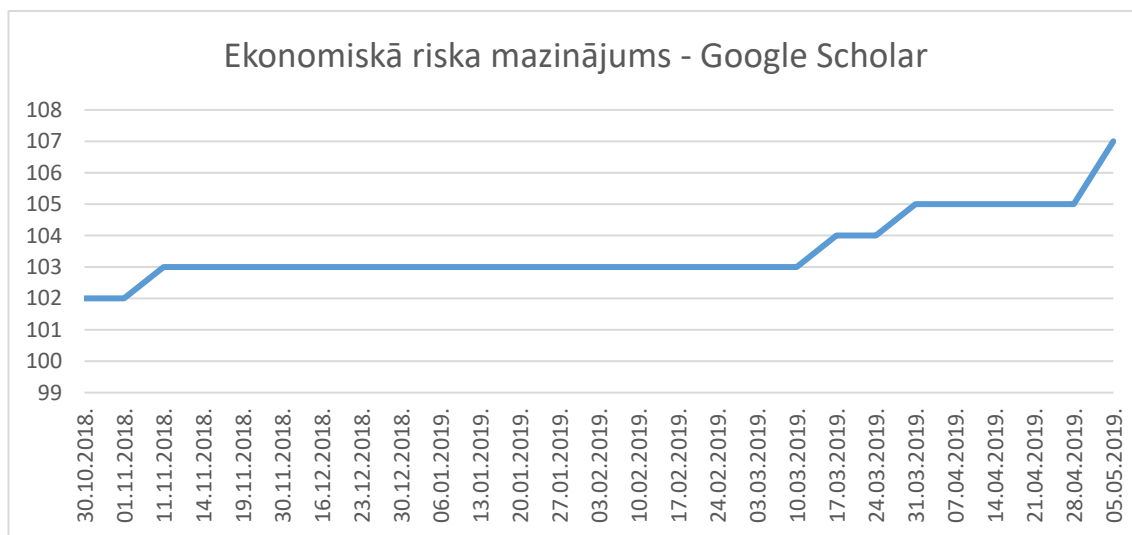
5.2. att. Blog Search Engine vietnes meklēšanas rezultāti laikā frāzei - "Freedom from risk" software testing

Blog Search Engine vietnē [5] 30. oktobrī 2018. gadā bija gandrīz 4,45 tūkstoši ierakstu, taču 2019. gada 5. maijā ir bijuši jau 10,7 tūkstoši rakstu. Pa šo laiku pieaugums šajā meklēšanas vietnē arī ir bijis krietni lielāks kā iepriekš aplūkotajiem terminiem, tas sasniedzis gandrīz 60 %. Attēlā 5.2. ir redzama līkne, kā pieaudzis atrasto rakstu skaits.

5.1. Ekonomiskā riska mazinājums

Tā kā lielākā daļa sistēmu tiek veidotas peļņas nolūkos, tad svarīgi ir veikt testēšanu, lai noteiktu potenciālos riskus un maksimāli samazinātu tos riskus, kas var negatīvi ietekmēt peļņu.

Kā aprakstīts ISO/IEC 25022 standartā [2], tad ekonomiskā riska mazinājums nosaka kvalitātes ietekmi uz ekonomiskajiem mērķiem, kas ir saistīti ar finansiālo stāvokli, efektīvu darbību, komerciāliem īpašumiem, reputāciju vai citiem resursiem, kas varētu būt apdraudēti vai arī sniegt iespējas.

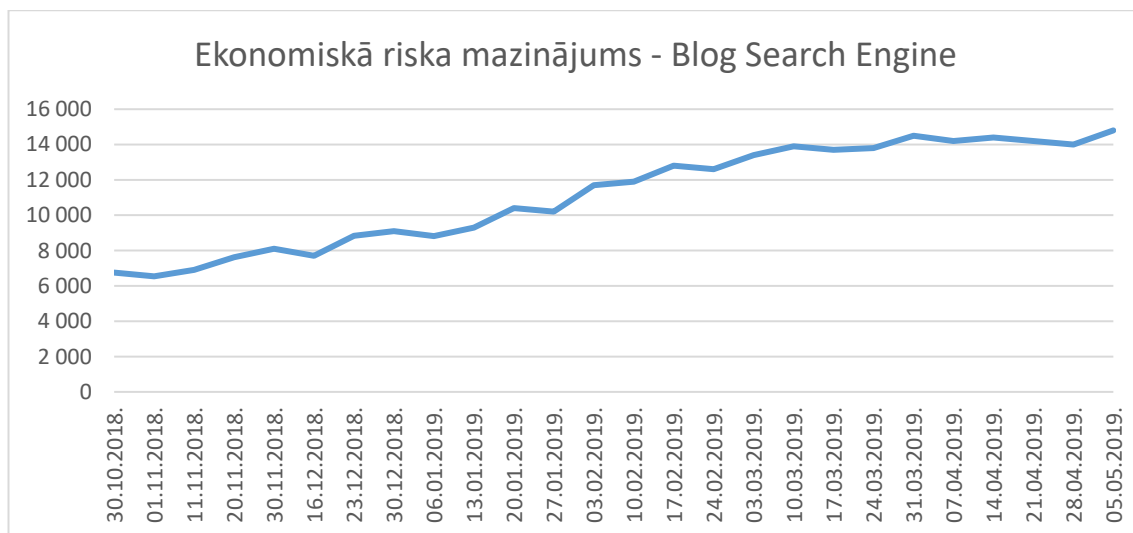


5.3. att. Google Scholar vietnes meklēšanas rezultāti laikā frāzei -
"Economic risk mitigation" software testing

Tā kā šāds lietojumkvalitātes atribūts pastāv tikai dažus gadus, tad arī rakstu skaits ir minimāls, kas nozīmē, ka potenciālos ekonomiskos riskus nosaka retos gadījumos, tāpēc pieejamā informācija ir neliela apjoma.

Google Scholar vietnē [4] 30. oktobrī 2018. gadā bija atrasti tikai 102 raksti, taču 2019. gada 5. maijā rakstu skaits bija tikai nedaudz lielāks – 107. Ja pēta šos datus no pieauguma procentuālā ziņā, tad tas ir bijis 4,7 %.

Attēlā 5.3. ir redzama līkne, kā pieaudzis atrasto rakstu skaits. Vērojams, ka ilgu laika periodu jauni raksti neparādījās starp atrastajiem rezultātiem, taču no marta vidus ir vērojams straujāks kāpums, kas nozīmē, ka interese ir pieaugusi.



**5.4. att. Blog Search Engine vietnes meklēšanas rezultāti laikā frāzei -
"Economic risk mitigation" software testing**

Blog Search Engine vietnē [5] 30. oktobrī 2018. gadā bija gandrīz 7 tūkstoši ierakstu, bet 2019. gada 5. maijā ir bijuši jau gandrīz 15 tūkstoši rakstu. Pa šo laiku pieaugums ir bijis tuvu 55 %. Attēlā 5.4. ir redzama līkne, kā pieaudzis atrasto rakstu skaits. Līdz martam rakstu skaita kāpums ir bijis lielāks, taču pēc tam cilvēku interese un pieejamās informācijas pieaugums ir nedaudz samazinājies.

Turpmākajās apakšnodaļās tiek aprakstītas ekonomiskā riska mazinājuma metrikas, kā novērtēt potenciālos ekonomiskos riskus un kādus rādītājus nepieciešams ņemt vērā, lai mazinātu ekonomiskos riskus.

5.1.1. Ienākumi no investīcijām

Investori ir ieinteresēti ieguldīt līdzekļus, ja ir potenciāls gūt peļņu, tāpēc šis jēdziens ROI (*return on investment*) ir plaši izmantots ekonomikā, aprēķinot iegūto peļņu no katra veiktā ieguldījuma. Mērķis ir samazināt ieguldījumu izmaksas [22].

Lai to noteiktu, tiek izmantota formula, kurā tiek atņemts no papildus iegūtajiem labumiem tā summa, kas ir tikusi ieguldīta, tad tiek iegūtais rezultāts izdalīts ar ieguldīto naudas summu. Par iegūtajiem labumiem uzskatāms, piemēram, personāla izdevumu samazināšana vai krājumu sarukums, vai materiālu izmaksu mazinājums, veicot koncentrētus pirkumus.

Var mērīt arī laiku, kas nepieciešams, lai sasniegtu iepriekš prognozētos ienākumus no veiktajiem ieguldījumiem. Šo metriku var izmantot, lai salīdzinātu reālo situāciju biznesā ar to, kāds būtu pieņemamais laiks uzstādīto mērķu sasniegšanai ienākumu ziņā no investīcijām [2].

ROI formula ir plaši izmantota ne tikai ekonomikas nozarē, bet arī veicot programmatūras testēšanu, jo tā ir vienkārša formula, kas to padara saprotamu arī speciālistiem, kam ir vājākas zināšanas ekonomikā.

5.1.2. Uzņēmējdarbības sniegums

Cita metrika, kas nosaka ekonomiskos riskus ir uzņēmējdarbības sniegums (*business performance*). Ekonomikas ziņā plašāk lietots jēdziens ir uzņēmējdarbības snieguma pārvaldība (*business performance management (BPM)*). Biznesa vārdnīcā [23] atrodams skaidrojums, ka tā ir pieeja uzņēmējdarbības vadīšanai, kas uztver biznesu kopumā, nevis daļa dažādos līmeņos, nosakot, kā labāk var sasniegt uzstādītos mērķus. Tā kā BPM cenšas apkopot pieejamo informāciju, vadītāji kļūst vairāk informēti par uzņēmuma stāvokli un var pieņemt labākus lēmumus.

Lai noteiktu biznesa sniegumu tiek mērīta rentabilitāte vai reālie pārdošanas apjomi salīdzinājumā pret uzstādīto pārdošanas apjoma mērķi. Iegūto rādītāju var salīdzināt ar IT ieguldījumu apjomiem vai citu uzņēmumu pārdošanas apjomiem [2].

5.1.3. IT ieguldījumu priekšrocības

Šo metriku nosaka, mērot reālās IT ieguldījumu priekšrocības pret uzstādīto mērķi, tādējādi iegūstot proporciju [2].

Viens no variantiem, ko izmantot, ir līdzsvarotās vadības karte [24]. Tas ir kopums, kas sevī ietver gan pārvaldību, gan stratēģisko plānošanu, gan vadības sistēmu, pielietojot biznesa, valsts pārvaldes un nevalstiskajās organizācijās. No šīs kartes var iegūt informāciju par iekšējiem procesiem un gala rezultātiem, jo tiek analizētas tādas lietas kā klienti, finanses, procesi, inovācijas un mācīšanās, ko var izmantot kā bāzes punktu mērījumiem.

5.1.4. Klientu apkalpošana

Daudzas sistēmas gūst peļņu tieši no apkalpoto klientu skaita. Jo vairāk apkalpoto klientu, jo lielāki ienākumi, tādēļ šis ir viens no aspektiem, ko būtu vēlams ņemt vērā, veicot sistēmas plānošanu un izstrādi.

Šī metrika nosaka, cik lielā mērā ir sasniegts plānotais apkalpoto klientu līmenis. Lai to noteiktu, daļa reālo apkalpoto klientu skaitu ar to, kāds ir bijis plānotais. Iegūtā vērtība var pārsniegt vienu, ja esošais klientu apkalpošanas līmenis pārsniedz vēlamo.

Kā viens no piemēriem varētu būt noteikt, cik lielā mērā tiek aizkavētas piegādes un vidējais gaidīšanas laiks, lai sasniegtu noteiktu klientu apkalpošanas līmeni [2].

5.1.5. Tīmekļa vietnes piesaistītie klienti

Mūsdienās klientu piesaistei nereti tiek izmantotas krāšņas mājaslapas, kas vilina tās lietotājus iegādāties kādu preci vai sākt lietot kādu sistēmu, tādēļ mājaslapas ir viens no veidiem, kā iegūt vai pārvilināt no konkurentiem klientus, kas nestu lielāku peļņu.

ISO/IEC 25022 standartā [2] teikts, lai noteiktu ekonomiskā riska mazinājumu, jāmēra proporcija, cik vienas vai vairāku noteiktu vietņu apmeklētāji ir kļuvuši par klientiem no visiem apmeklētājiem kopā. Šo attiecību var noteikt gan konkrētai tīmekļa lapai, gan visai tīmekļa vietnei kopumā, atkarībā no testēšanas vēlmēm.

5.1.6. Ienākumi no katra klienta

Ne vienmēr ir svarīgi tieši kopējā peļņa. Dažkārt lielāka nozīme ir katram lietotājam vai klientam atsevišķi.

Novērtējot šo lietojumkvalitātes aspektu, tiek analizēti ienākumi, ko rada katrs klients atsevišķi, nevis visi cilvēki kopumā. Ir vairāki veidi, kā klienti var tikt iedalīti, piemēram, esošie un jaunie klienti, kas var tikt izmantoti, lai novērtētu iespējamus zaudējumus, kas varētu rasties, ieviešot jaunu funkcionalitāti sistēmā [2]. Pastāv iespējamība, ka jaunie klienti nenes peļņu, jo izmaksas jaunu funkciju ieviešanai sistēmā krietni pārsniedz ienākumus, ko jaunie klienti varētu radīt.

5.1.7. Kļūdas ar finansiālām sekām

Biznesam lielākās problēmas rada tādas kļūdas, kas nes zaudējumus, tādēļ ir nepieciešams izvērtēt to, kādas ir potenciālās kļūdas un kādus preventīvos pasākumus varētu veikt, lai to samazinātu.

Lai noteiktu šī atribūta vērtību, tiek mērīts lietošanas situāciju īpatsvars, kurās ir cilvēka vai sistēmas kļūdas, kas izraisa finansiālas sekas no visām lietošanas situācijām kopumā.

Kļūdām var pievienot svaru, kas atkarīgs no ietekmes uz finansiālo stāvokli. Lietošanas situācijas var definēt kā transakcijas vai laiku, bet kļūdas var ietvert, piemēram, datu bojāšanu. Ekonomiskās sekas var atstāt iespaidu gan uz organizāciju, gan uz individuālu lietotāju.

Piemēram, finansiālas sekas indivīdam, kurš nespēj efektīvi un rezultatīvi sasniegt savu mērķi iegādāties paredzēto transporta biļeti, kas var novest pie nespējas ceļot, kā bijis iepļānots, vai nepieciešamības iegādāties jaunu biļeti.

5.2. Veselības un drošības riska mazinājums

Sistēmas tiek izstrādātas dažādiem nolūkiem, piemēram, medicīniskām procedūrām. Sevišķi bīstamas ir kļūdas smagi slimu pacientu apskatei vai procedūrām paredzēto iekārtu sistēmās, jo pat sīkākās kļūdas var novest pacientu līdz letālam iznākamam. Vēsturiski šādas kļūdas ir bijušas īpašos gadījumos, kad pacienta situācija nav bijusi standartsituācija un tikusi saņemta nāvējoša starojuma deva.

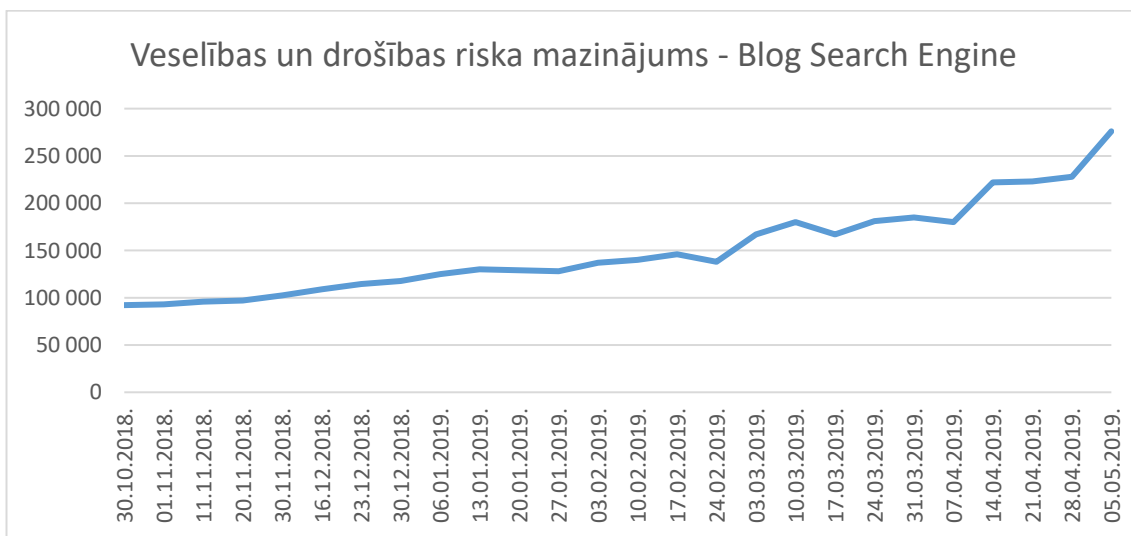
Balstoties uz ISO/IEC 25022 standartu [2], veselības un drošības riska mazinājuma mērījumi nosaka kvalitātes ietekmi uz veselības un drošības aspektiem, piemēram, izvairīšanās no atkārtotiem deformācijas ievainojumiem.

Lai noteiktu drošības problēmas sistēmā, var veikt uzbrukumu bāzēto testēšanu. Kā aprakstīts ISTQB glosārijā, tad tā ir uz pieredzi balstīta testēšanas tehnika, kas izmanto programmatūras uzbrukumus, lai mākslīgi izraisītu sistēmas kļūdas, it īpaši tādas, kas ir saistītas ar drošību [3].



5.5. att. Google Scholar vietnes meklēšanas rezultāti laikā frāzei - "Health and safety risk" software testing

Google Scholar vietnē [4] 30. oktobrī 2018. gadā bija apmēram 1 720 rakstu, taču 2019. gada 5. maijā rakstu skaits pieaudzis par 8%, sasniedzot 1 870 lielu skaitu.. Attēlā 5.5. ir redzama līkne, kā pieaudzis atrasto rakstu skaits.



5.6. att. Blog Search Engine vietnes meklēšanas rezultāti laikā frāzei - "Health and safety risk" software testing

Blog Search Engine vietnē [5] 30. oktobrī 2018. gadā bija gandrīz 100 tūkstoši ierakstu, taču 2019. gada 5. maijā tas pieaudzis līdz 276 tūkstošiem. Pa šo laiku pieaugums ir bijis gandrīz 67 %. Attēlā 5.6. ir redzama līkne, kā pieaudzis atrasto rakstu skaits. Līknei ir paraboliskas iezīmes, kas nozīmē, ka cilvēku interese par šo lietojumkvalitātes atribūtu arvien pieaug.

Turpmākajās apakšnodaļās tiek aprakstītas dažādas veselības un drošības riska mazinājuma metrikas, kā noteikt sistēmas riskbrīvību no kaitējumiem veselībai un drošībai, kā arī kādus rādītājus tam nepieciešams iegūt.

5.2.1. Lietotāju veselības problēmu ziņošanas biežums

Tā kā sistēmās mēdz būt nepilnības, tad ir gadījumi, kad tas nodara būtisku kaitējumu veselībai, kas var atstāt paliekošas sekas.

Šis ir statistiskas mērījums, ko veic, lai noteiktu proporciju lietotājiem, kas ziņojuši par veselības problēmām, kuras izraisījusi sistēmas lietošana, no visa kopējā lietotāju skaita.

Veselības problēmas var būt tādas kā ķermeņa deformācija, nogurums, galvassāpes un dažādas citas [2]. Bieži vien nogurumu var radīt košās krāsas vai nepiemērots fonta lielums, tādēļ lietotāji var atteikties šo sistēmu lietot vispār, ja šāda opcija ir iespējama.

Šos mērījumus varētu traucēt iegūt fakts, ka ne vienmēr cilvēki ziņo par to, ka tiem ir nodarīts kāds kaitējums veselībai, lietojot izstrādāto programmatūru. Arī problēmas var rasties vēlāk pēc ārsta apmeklējuma vai pēc ilgstošas sistēmas lietošanas, ko lietotājam ir grūti sasaistīt tieši ar sistēmas lietošanu, jo tas var nebūt acīmredzami.

5.2.2. Ietekme uz lietotāju veselību un drošību

Ietekmējoši aspekti var būt gan mentālie veselības traucējumi, gan arī fiziskie veselības traucējumi un draudi drošībai. Šīs problēmas mēdz izraisīt, piemēram, stress, kurš radies no grūtībām lietot sistēmu, jo tai ir vāji izstrādāta lietotāja saskarne.

Lai to izmērītu, var izmantot ISO/IEC 25022 standartā [2] atrodamo formulu, kurā tiek savstarpēji reizinātas divas lietas – summa no katra lietotāja laika ilguma, kurā tas ticis ietekmēts, reizinājuma ar problēmas nozīmīguma līmeni, cik lielā mērā persona tikusi ietekmēta, un daļījums starp 1 un sistēmas funkcijas lietošanas ilgumu.

5.2.3. Ietekmējošo lietotāju drošība

Tiek mērīta attiecība starp tiem lietotājiem, kuri ir nokļuvuši apdraudētā situācijā, un visiem lietotājiem kopumā, kas tikuši ietekmēti, izmantojot sistēmu. Tomēr dažkārt labāk varētu būt mērīt konkrētu skaitu lietotāju, kas tikuši ietekmēti, nevis proporciju.

Viens no piemēriem ir pacientu skaits, kas saņēmuši nekorektu ārstēšanu pret visiem pacientiem kopumā [2].

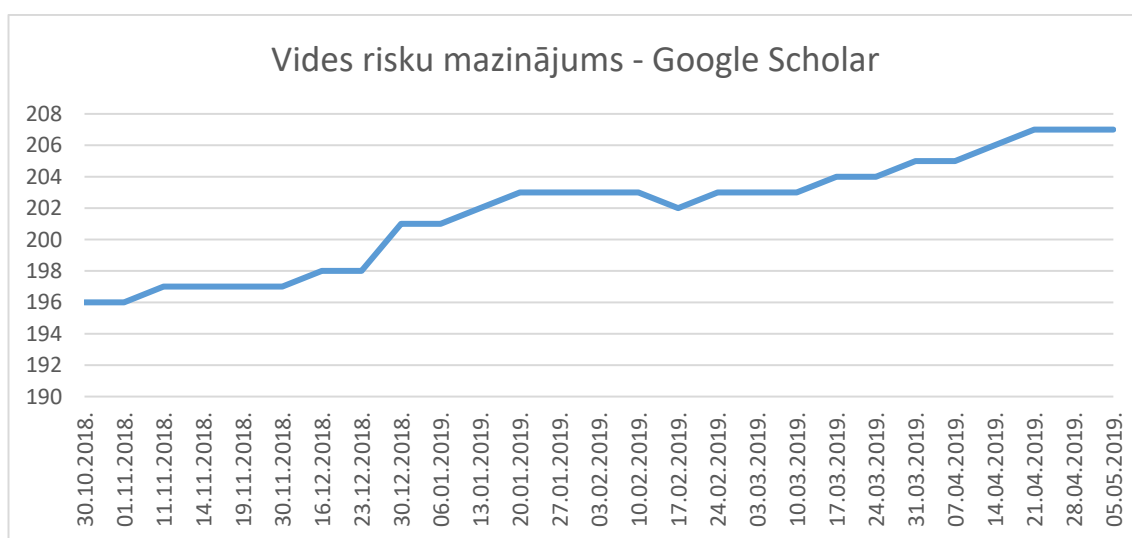
5.3. Vides risku mazinājums

Ir ieteicams novērtēt esošo produkta vai sistēmas ražošanas un lietošanas ietekmi uz vidi, salīdzinot ar uzstādīto mērķi [2].

Ietekme uz vidi varētu būt, piemēram, piesārņojums, traucējošas skaņas vai pat globālā sasilšana, atkarībā no izstrādātās sistēmas tipa. Bieži draudus rada lielo rūpnīcu sistēmas, kas nepiemērotā brīdī atver slūžas, ļaujot dabā izplūst vēl nefiltrētam ūdenim.

Problēmas var radīt ne tikai defekti sistēmā, bet arī lietotāju radītās kļūdas, tāpēc nepieciešams pastiprināti notestēt gadījumus, kā varētu nekorekti lietot sistēmu un kas varētu šādās situācijās notikt, kas palīdzēt rast risinājumus, lai videi nodarīto kaitējumu preventīvi samazinātu.

Tāpat kā, lai novērtētu veselības un drošības riskus, arī vides risku noteikšanai izmantojama testēšanas metode, apkopojot lietotāju ziņojumus par nodarītajiem kaitējumiem videi, taču uz to vien nevar paļauties, jo nav nekādu garantiju, ka lietotāji visos gadījumos ziņos par videi nodarīto kaitējumu.



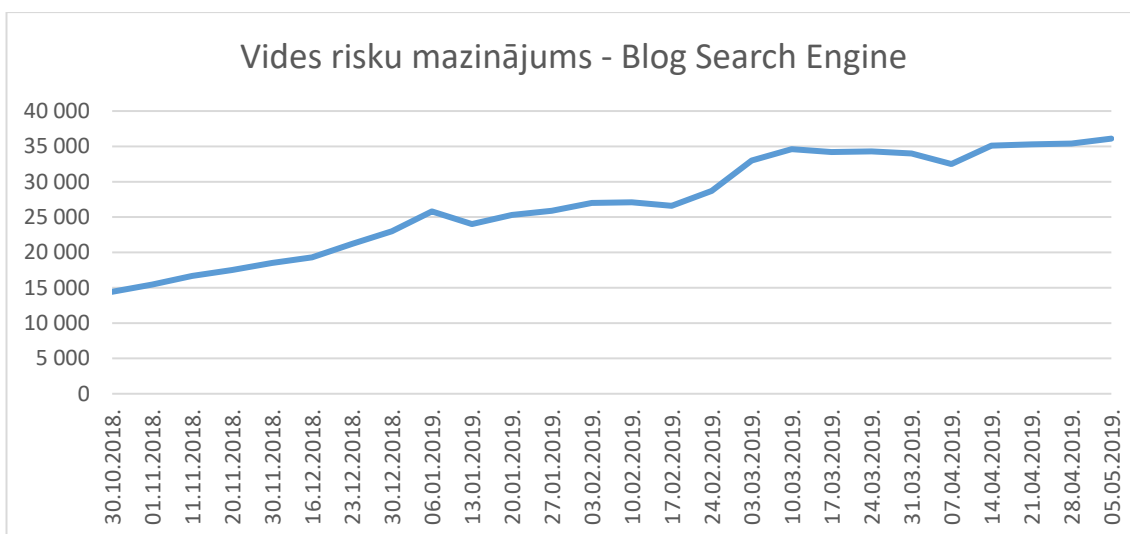
5.7. att. Google Scholar vietnes meklēšanas rezultāti laikā frāzei -

"Environmental risk mitigation" software testing

Google Scholar vietnē [4] 30. oktobrī 2018. gadā bija gandrīz 196 ieraksti, taču 2019. gada 5. maijā ir bijuši jau 207 ieraksti. Pa šo laiku pieaugums ir bijis vairāk kā 5 %.

Attēlā 5.7. ir redzama līkne, kā pieaudzis atrasto rakstu skaits. Laika periodā no janvāra vidus līdz marta vidum pieaugums ir bijis minimāls, taču pārējā laika periodā jauni raksti ir parādījušies biežāk.

Periodisko interesi par vides risku samazinājumu varētu saistīt ar to, ka ik pa laikam notiek dažādas vides akcijas, kas pievērš pastiprinātu cilvēku uzmanību vides problēmām, ko nereti izraisa dažādas mūsdienu tehnoloģijas, tādēļ arī programmatūras izstrādes procesā šī tēma kļūst aktuālāka periodiski.



5.8. att. Blog Search Engine vietnes meklēšanas rezultāti laikā frāzei - "Environmental risk mitigation" software testing

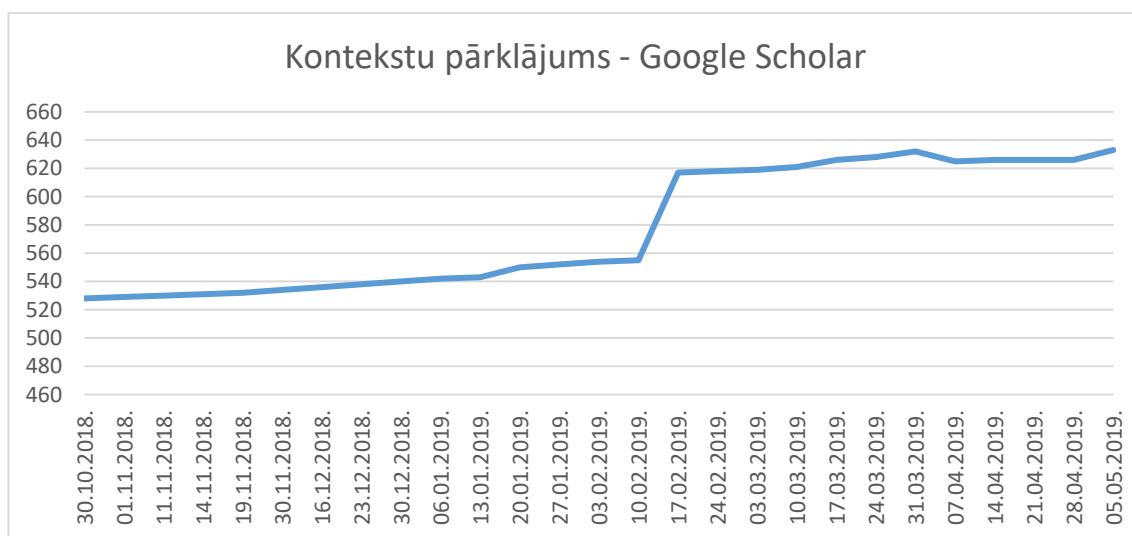
Blog Search Engine vietnē [5] 30. oktobrī 2018. gadā bija gandrīz 15 tūkstoši ierakstu, taču 2019. gada 5. maijā ir bijuši jau 36 tūkstoši ierakstu. Pa šo laiku pieaugums ir bijis 60 %.

Kā redzams attēlā 5.8., līdz martam interese par šo tēmu ir augusi straujāk, nekā pēc tam, jo tad rakstu skaita kāpums nav vērojams tik straujš.

6. KONTEKSTU PĀRKLĀJUMS

Vērtējot sistēmu, ir jāņem vērā konteksts, kādā sistēma tiks lietota, jo tas var mainīt un būtiski ietekmēt nepieciešamo sistēmas funkcionalitāti vai izskatu. Kā piemēram, grāmatvežiem mazāk svarīgs ir sistēmas izskats, bet gan funkcionālās iespējas, savukārt lietotnei, kas paredzēta izklaidei, kā svarīgākais aspekts ir izskats.

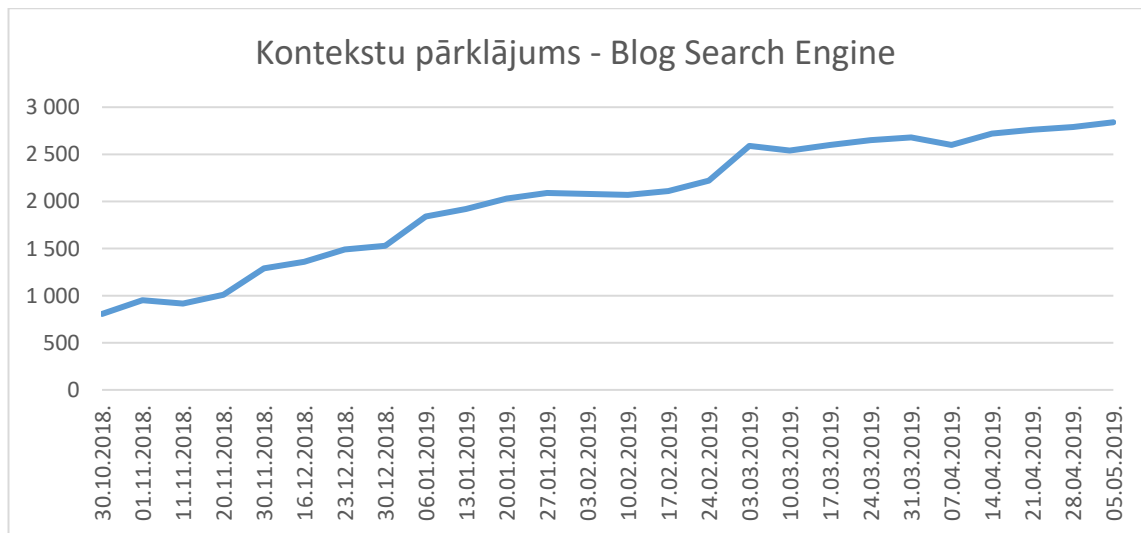
Kā teikts ISO/IEC 25022 standartā [2], kontekstu pārklājuma mērījumi nosaka pakāpi, kādā produkts vai sistēma var tikt efektīvi, rezultatīvi, lietotājus apmierinoši un ar maksimāli mazu riska līmeni lietota gan noteiktos lietošanas kontekstos, gan ārpus konkrēti paredzētiem gadījumiem.



6.1. att. Google Scholar vietnes meklēšanas rezultāti laikā frāzei -
"Context coverage" software testing

Google Scholar vietnē [4] 30. oktobrī 2018. gadā bija gandrīz 530 ierakstu, taču 2019. gada 5. maijā ir bijuši jau 633 ieraksti. Pa šo laiku pieaugums ir bijis gandrīz 17 %, kas ir lielākais no visiem lietojumkvalitātes atribūtu rakstu pieaugumiem.

Attēlā 6.1. ir redzama likne, kā pieaudzis atrasto rakstu skaits. Neparasta situācija vērojama ap februāra vidu, kad rakstu skaits pēkšņi pieaudzis ļoti strauji, taču pārējā periodā gan pirms, gan pēc tam liels kāpums nav vērojams. Tam varētu būt vairāki iemesli – gan kādas izmaiņas rakstu meklēšanas algoritmā, gan straujais interese pieaugums vai arī kādi organizēti semināri un apmācības.



6.2. att. Blog Search Engine vietnes meklēšanas rezultāti laikā frāzei - "Context coverage" software testing

Blog Search Engine vietnē [5] 30. oktobrī 2018. gadā bija gandrīz 806 ierakstu, kas ir tikai nedaudz vairāk kā Google Scholar [4] vietnē, taču 2019. gada 5. maijā ir bijuši jau 2, 84 tūkstoši ierakstu. Pa šo laiku pieaugums ir bijis arī lielākais no visiem lietojumkvalitātes atribūtu rakstu skaita pieaugumiem, tie ir bijuši gandrīz 72 %. Attēlā 6.2. ir redzama līkne, kā pieaudzis atrasto rakstu skaits. Tā ir diezgan vienmērīga, salīdzinājumā ar attēlā 6.1. redzamo līkni.

Turpmākajās apakšnodaļās tiek aprakstīts, kā novērtēt kontekstu pārklājumus un kādus rādītājus nepieciešams iegūt tā noteikšanai.

6.1. Kontekstu pilnīgums

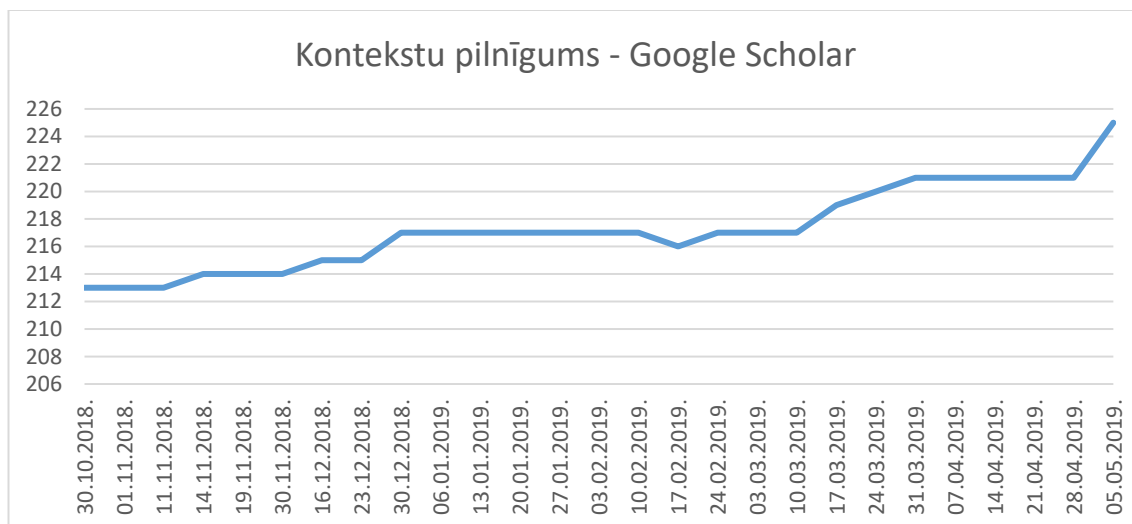
ISO/IEC 25022 [2] standartā minēts, ka konteksta pilnīgums ir pakāpe, kurā produkts vai sistēma var tikt lietota ar nepieciešamo rezultativitātes līmeni, efektivitāti, apmierinātību un riskbrīvību katrā konkrētajā lietošanas kontekstā.

Lai noteiktu kontekstu pilnīgumu, tiek mērīta proporcija kontekstiem ar pieņemamu lietojamību un risku pret kopējo kontekstu lietojumu skaitu, kas ir nepieciešami un savstarpēji atšķirīgi. Analīze un lietotāju testēšana ir pielietojama, nosakot, vai produktam vai sistēmai ir pieņemama lietojamība visās paredzētajās lietotāju tipu, uzdevumu un vides veidu kombinācijās.

Prasības pret kontekstu pārklājumu var būt dažādas atkarībā no situācijām, kurās sistēmu paredzēts lietot. Var būt prasība, ka jānoklāj pilnīgi visi konteksti, 50% no visiem vai kāds konkrēts skaits. Tas viss balstās uz pasūtītāja vēlmēm vai likumdošanas, ja tā regulē kādu no jomām, nosakot precīzus kritērijus.

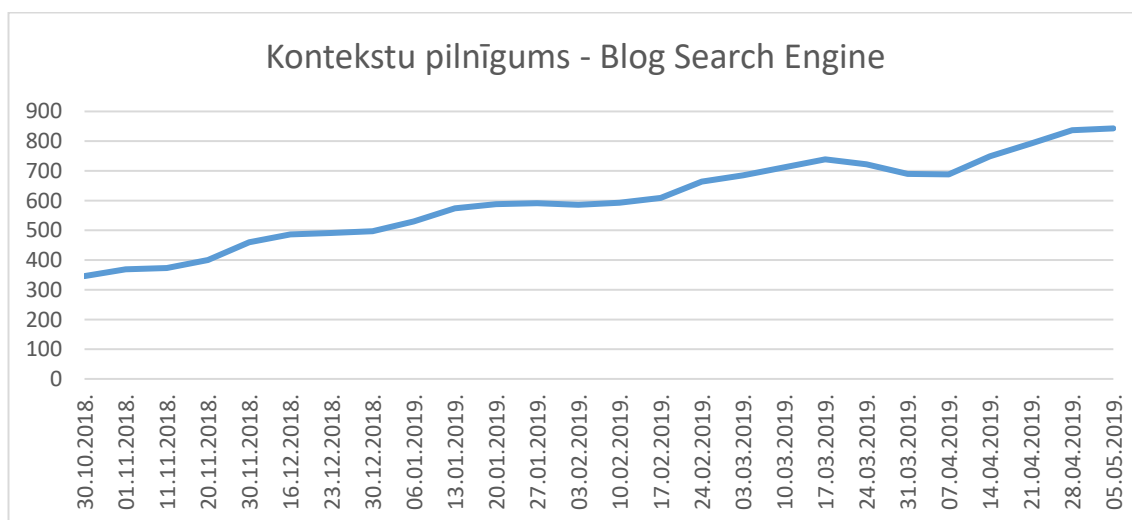
Ja daži no kontekstiem kādu iemeslu dēļ ir svarīgāki, tad tiem var tikt piemērots lielāks svars, tādējādi ietekmējot iegūto mērījuma rezultātu spēcīgāk, nekā mazāk svarīgāki lietošanas konteksti.

Nosakot dažādus lietojamības kontekstus, iespējams izveidot dažādus testpiemērus, ko var izmantot tādu testēšanas metodi kā izsmeļošā testēšana, kas ir testēšanas pieeja, kurā testkomplekts ietver visas ieejas vērtību un priekšnosacījumu kombinācijas [3]. Tādējādi iespējams noteikt, vai visi konteksti un potenciālie lietošanas gadījumi strādātu korekti un sistēmā ir pārklāti.



6.3. att. Google Scholar vietnes meklēšanas rezultāti laikā frāzei - "Context completeness" software testing

Google Scholar vietnē [4] 30. oktobrī 2018. gadā bija aptuveni 213 ierakstu, taču 2019. gada 5. maijā ir bijuši jau 225 ieraksti. Pa šo laiku pieaugums ir bijis aptuveni 5 %, taču ir bijuši periodi, kuros tik pat kā jaunu rakstu nav bijis, bet citos periodos pieaugums ir bijis straujāks. Attēlā 6.3. ir redzama līkne, kā pieaudzis atrasto rakstu skaits.



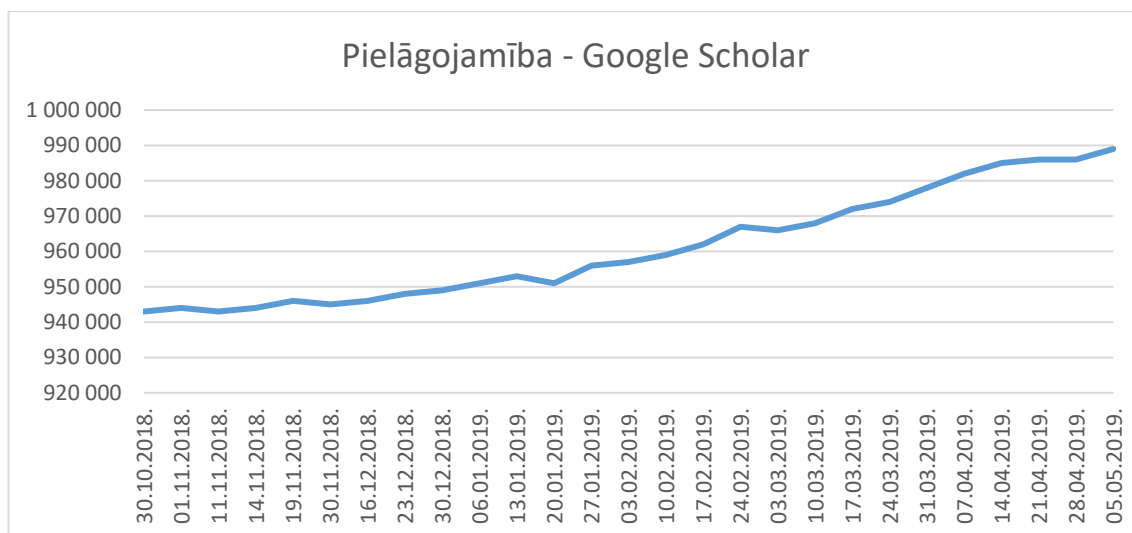
6.4. att. Blog Search Engine vietnes meklēšanas rezultāti laikā frāzei - "Context completeness" software testing

Blog Search Engine vietnē [5] 30. oktobrī 2018. gadā bija gandrīz 350 ierakstu, taču 2019. gada 5. maijā atrasto rakstu skaits ir pieaudzis par aptuveni 60%, pietuvojoties 850 atrastajiem rakstiem. Attēlā 6.4. ir redzama līkne, kā pieaudzis atrasto rakstu skaits. Līknei nav izteiktu kāpumu, kas norāda uz vienmērīgu cilvēku interesi par šo tēmu. Tātad informācijas kļūst aizvien vairāk un plašāk pieejamas, taču bez straujiem kāpumiem vai ilgstošiem miera periodiem.

6.2. Pielāgojamība

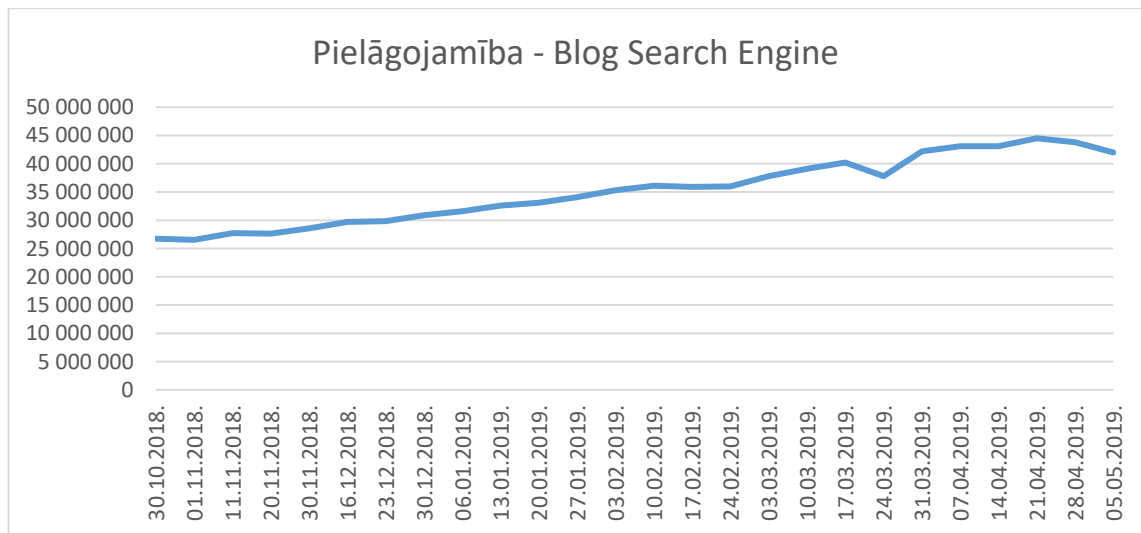
Līdzīgi kā kontekstu pilnīguma gadījumā, arī pielāgojamības mērījumi nosaka pakāpi, kurā produktu vai sistēmu var lietot ar akceptējamu efektivitātes līmeni, rezultativitāti, patiku un riskbrīvību katrā konkrētajā lietošanas kontekstā, kas ir paredzēti.

Pielāgojamība ļauj ņemt vērā potenciālās sekas, iespējas un individuālās vēlmes, kas, iespējams, nav bijušas paredzētas iepriekš. Lai to izmērītu, var analizēt produkta īpašības un lietošanas kontekstu, nosakot apjomu, kādā produktu varētu lietot papildus lietotāju tipi, lai sasniegtu dažādu papildus tipu mērķus, ar efektivitātes līmeni, rezultativitāti, patiku un riskbrīvību dažādos lietojuma kontekstos, vai arī testēt produktu ar lietotājiem šajos lietošanas kontekstos, vai noteikt produkta pielāgošanās spējas jauniem lietotāju tipiem, uzdevumiem un vidēm, kā arī piemērotību individualizācijai [2].



6.5. att. Google Scholar vietnes meklēšanas rezultāti laikā frāzei - "Flexibility" software testing

Google Scholar vietnē [4] 30. oktobrī 2018. gadā bija aptuveni 943 tūkstoši ierakstu, taču 2019. gada 5. maijā ir bijuši gandrīz 990 tūkstoši ierakstu. Pa šo laiku pieaugums ir bijis nepilni 5%. Attēlā 6.5. ir redzama līkne, kā pieaudzis atrasto rakstu skaits, kurai ir neliela tendence uz parabolisku līkni, kas rosina domāt, ka interese par šo tēmu pieaug aizvien straujāk, kaut gan pielāgojamība ir populārs testēšanas virziens.



6.6. att. Blog Search Engine vietnes meklēšanas rezultāti laikā frāzei - "Flexibility" software testing

Blog Search Engine vietnē [5] 30. oktobrī 2018. gadā bija ap 26 miljoniem ierakstu, taču 2019. gada 5. maijā ir bijuši par aptuveni 36% rakstu vairāk, kas ir 42 miljoni. Attēlā 6.6. ir redzama līkne, kā pieaudzis atrasto rakstu skaits.

Turpmākajās apakšnodaļās tiek aprakstītas dažādas pielāgojamības metrikas, kā noteikt sistēmas pielāgojamību un kādus rādītājus tam nepieciešams iegūt.

6.2.1. Pielāgojamība lietošanas kontekstiem

Pielāgojamība lietošanas kontekstiem ir apjoms, kādā produkts var tikt lietots bez nekādām vai ļoti minimālām modifikācijām papildus lietošanas kontekstos, no kuriem iespējamie varianti ir tādi kā dažādi lietotāju tipi, kam raksturīgi specifiskāki ieradumi, kas var palīdzēt nestandartsituāciju testēšanā, arī atšķirīgi uzdevumi var sniegt plašāku ieskatu tajā, kādos kontekstos sistēma potenciāli tiks lietota, un vides atšķirības, kas var mainīt lietošanu.

Nelielas izmaiņas ir domātas kā tādas, kuras lietotājs var veikt pats, vai tādas, kas ir saistītas ar teksta vai datu labošanu, kas sevišķi daudz sistēmu neizmaina, saglabājot funkcionalitāti, kāda tā bijusi pirms labošanas.

Lai šī metrika būtu salīdzināma, tiek dalīts papildus kontekstu skaits, kuros produktu var lietot ar pieņemamu lietojumkvalitātes līmeni, ar kopējo papildus kontekstu skaitu, kurā produktu vispār varētu lietot, iegūstot proporciju. Jo iegūtā proporcija vairāk tiecas uz vērtību viens, jo labāka sistēmas pielāgojamība [2].

6.2.2. Produkta pielāgojamība

Produkta pielāgojamība nosaka, cik vienkārši un viegli produktu var mainīt un modificēt, lai tas atbilstu papildus lietotāju prasībām. Jo vieglāk to izdarīt, jo sistēma uzskatāmāka labāk pielāgojama.

Šīs metrikas aprēķināšanai pielietojama formula, kurā tiek reizinātas divas lietas – dalījums vieniniekam ar visu kopējo lietotāju prasību skaitu un summa no katras prasības modificējamības [2].

6.2.3. Prasmju neatkarība

Prasmju neatkarība ir apjoms, kādā produktu var lietot cilvēki bez īpašām zināšanām, prasmēm, iemaņām vai pieredzes. To nosaka, mērot proporciju papildus lietotāju grupu skaitam, kuru dalībniekiem nav īpašu zināšanu, prasmju vai pieredzes produkta lietošanā, pret kopējo potenciālo lietotāju grupu skaitu bez kādām specifiskām zināšanām, iemaņām un pieredzes.

Šis rādītājs ir būtisks gadījumos, kad sistēma tiek izstrādāta zinošai lietotāju grupai, taču to potenciāli varētu lietot plaša spektra lietotāji, kuriem šādu zināšanu varētu nebūt. Tāpat arī bieži ir situācijas, kad sistēmas testētāji ir augsti kvalificēti speciālisti, kas var liegt iespēju testēt sistēmu kā lietotājam, kam ir maza pieredze vai zināšanas konkrētajā jomā, tādējādi novedot pie tā, ka sistēma lietotājiem šķitīs sarežģīta un nelietojama.

Lai to novērtētu, var izmantot lietotājcentrēto pieeju [12], kad tiek izvēlēti tādi lietotāji, kuri nav no izstrādes komandas un sistēmu nepārzina. Tādā veidā var tikt iegūta informācija, kā cilvēki bez īpašām zināšanām sistēmu lietotu.

Ja iegūtā proporcija tuvojas nullei, tad produktam ir tieksme būt sarežģītākam un to iespējams lietot ir īpaši zinošiem vai prasmīgiem lietotājiem. Turpretī gadījumā, kad iegūtais rezultāts tiecas uz vieninieku, tad ir ticams, ka sistēmu spēs lietot jebkura līmeņa lietotāji – gan vairāk, gan mazāk zinoši lietotāji [2].

REZULTĀTI

Bakalaura darbā ir iegūts plašs pārskats no dažādu avotu skatu punkta par lietojumkvalitāti un tās galvenajiem atribūtiem, kas ir rezultativitāte, efektivitāte, apmierinātība, riskbrīvība un kontekstu pārklājums, aplūkojot arī to apakšatribūtus.

Ir veikts dažādu testēšanas metožu apkopojums (skatīt 1. tabulu), lai saprastu, ar kādiem testēšanas paņēmieniem katrā gadījumā būtu iespējams labāk un kvalitatīvāk veikt lietojumkvalitātes novērtēšanu, lai atzītu sistēmu par kvalitatīvi lietojamu.

1. tabula

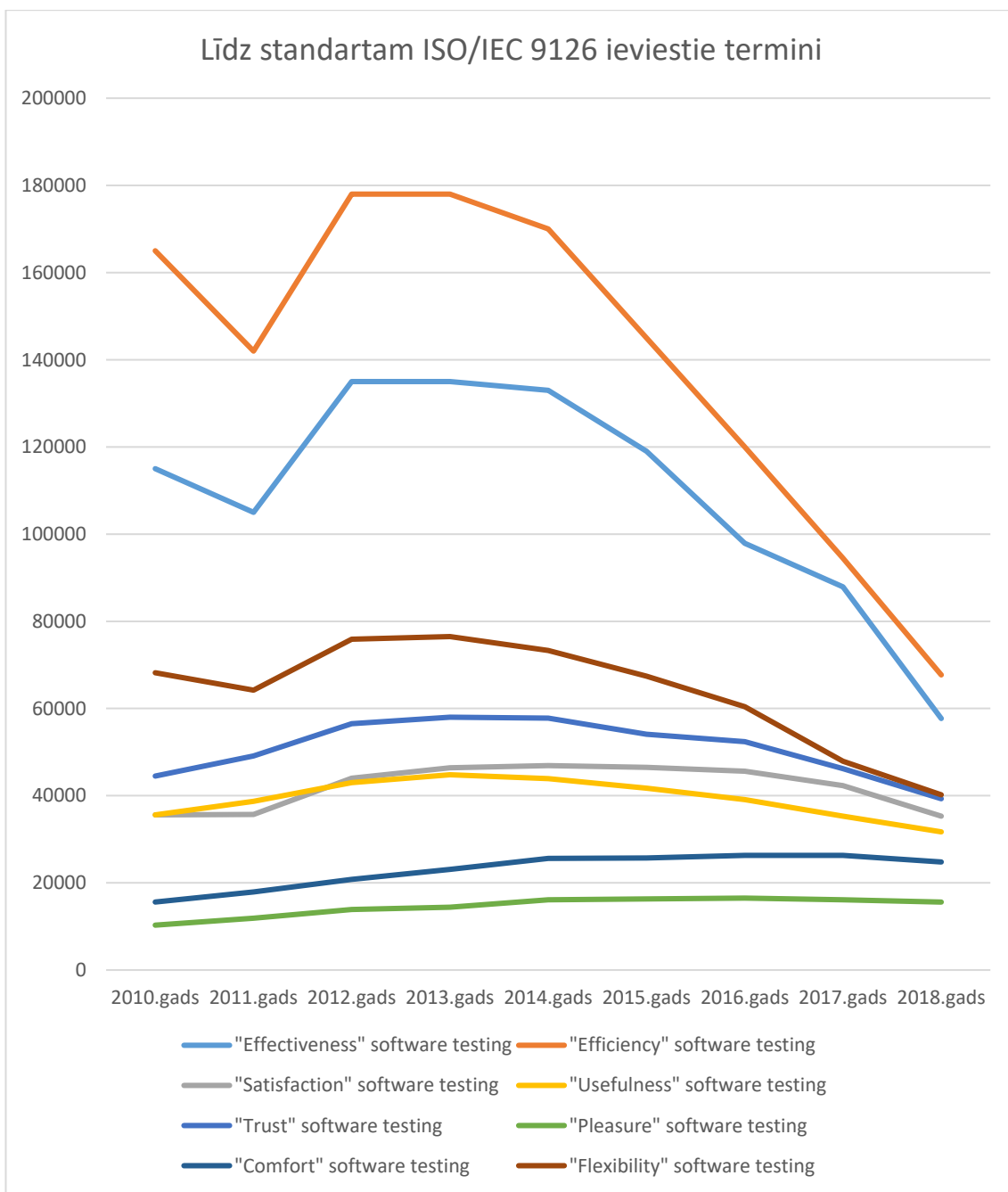
Lietojumkvalitātes atribūtu testēšanas metode

Atribūts/ testēšanas metode	Lietotājcetrētā pieeja	Anketēšanas metode	Regresē testēšana	Riskbāzētā testēšana	Integrācijas testēšana	Uzbrukumu bāzētā testēšana	Lietotāju ziņojumu ieguve	Izsmeljoša testēšana
Rezultativitāte	X							
Efektivitāte	X							
Apmierinātība	X	X						
Noderīgums	X	X						
Uzticēšanās	X	X						
Patika	X	X						
Ērtums	X	X						
Riskbrīvība			X	X	X	X	X	
Ekonomiskā riska mazinājums			X	X	X			
Veselības un drošības risku mazinājums			X	X	X	X	X	
Vides risku mazinājums			X	X	X		X	
Kontekstu pārklājums	X							X
Kontekstu pilnīgums								X
Pielāgojamība	X							

Ir veikta padziļinātākā izpēte veidiem, kā iegūt nepieciešamos datus, lai veiktu dažādu lietojumkvalitātes aspektu testēšanu, no kuriem neprecīzākais veids ir anketēšanas metode, jo balstās uz cilvēku personībām, kas nosaka dažādu lietu nozīmīgumu. Tāpat arī mēdz būt problēmas iegūt dažādus datus par nodarītajiem kaitējumiem gan veselībai, gan videi, jo ne vienmēr cilvēkiem šādas problēmas asociējas ar konkrētas sistēmas lietošanu, tāpēc lietotāju ziņošana par problēmām kā datu ieguves veids nav tāds, uz kuru var 100% paļauties, ka visos gadījumos ziņojumi tiks saņemti.

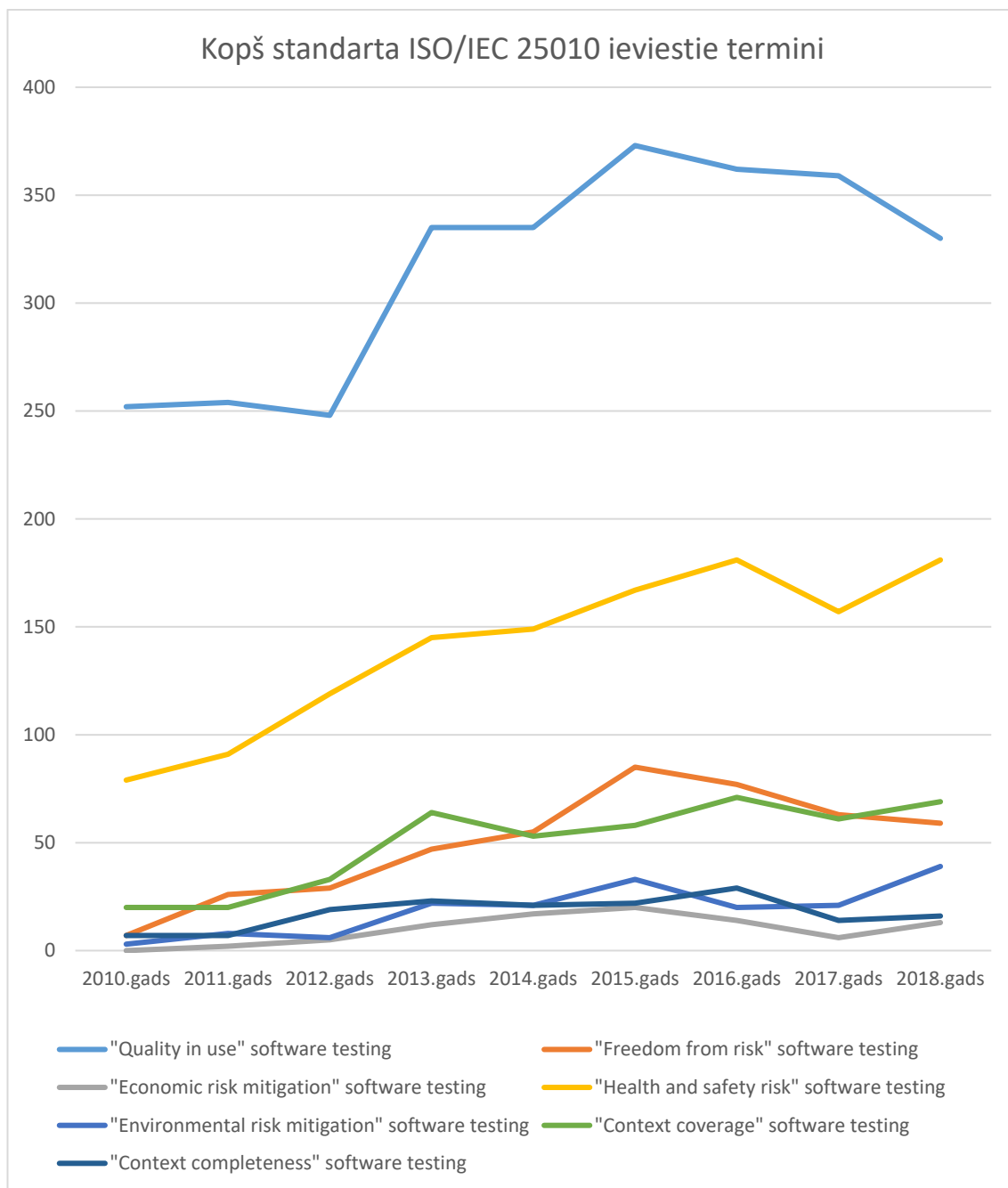
Autors ir ieguvis statistiku par vairāku lietojumkvalitātes atribūtu popularitāti gan zinātniskajos, gan bloga rakstos vairāku mēnešu garumā, no kuriem lielākais rakstu skaita pieaugums ir terminiem “kontekstu pārklājums” un “veselības un drošības riska mazinājums”, bet mazākais pieaugums ir terminam “noderīgums” un “efektivitāte”. Ja pieaugums bija lielāks Google Scholar [4] vietnē starp zinātniskajiem rakstiem, tad pieaugums bija lielāks arī starp bloga rakstiem Blog Search Engine vietnē [5]. Meklējamās frāzes redzamas 2. pielikumā.

Statistikas automatizētai ieguvei ir tikusi izveidota programma valodā Python, kas automātiski nolasa atrasto ierakstu skaitu, kas samazina nepieciešamo laiku datu ieguvei. Programmas kods redzams 1. pielikumā.



1. att. Līdz standartam ISO/IEC 9126 atsevišķi izdalīto terminu pieaugums pa gadiem vietnē Google Scholar

Kā redzams 1. attēlā, tad tiem terminiem, kas tikuši ieviesti līdz standartam ISO/IEC 9126 [9], pieaugumam ir tendence samazināties, taču joprojām rakstu skaita pieaugums ir mērāms desmitos tūkstošu.



2. att. Sākot ar standartu ISO/IEC 25010 atsevišķi izdalīto terminu pieaugums pa gadiem vietnē Google Scholar

Savukārt, kā redzams 2. attēlā, tad jaunākajiem terminiem, kas ieviesti sākot ar standartu ISO/IEC 25010 [1], interesi raksturojošo līkņu pieaugumam ir tendence pieaugt ar gadiem, bet pieaugums šiem terminiem pat nesasniedz 500 rakstus gadā, kas nozīmē, ka pieejamā informācija par šādiem lietojumkvalitātes atribūtiem ir samērā niecīga.

SECINĀJUMI

Lietojumkvalitātes netestēšanas dēļ var rasties zaudējumi sistēmas pasūtītājam, ja tā ir paredzēta lietošanai publiski, jo mūsdienu konkurences apstākļos, lietotājs pārtrauks sistēmu lietot, ja tā viņam nešķitīs vizuāli pievilcīga vai būs neērti lietojama, vai izraisīs kādas paliekošas sekas, kā dēļ lietotājs konkrētajai sistēmai vairs nespēs uzticēties.

Jēdziens “lietojumkvalitāte” aptver plašu testēšanas jomu, kuras precīzākai testēšanai nepieciešams to iedalīt sīkākos apgabalos.

Darbā apkopotās un metrikas un atribūtu novērtējumus var izmantot brīdī, kad sistēmas pasūtītājs veic nefunkcionālo prasību noformēšanu [14]. Sistēma būtu uzskatāma par labu no lietojumkvalitātes viedokļa, ja tā dažādos darbā apskatītajos testos sasniedz noteiktus rādītājus.

Lietojumkvalitāte ir grūti testējama un nosakāma, jo, piemēram, apmierinātības noteikšanai ir nepieciešams liels skaits cilvēku, kas varētu būt testējamās lietotāju grupas dalībnieki. Tā kā vairāku atribūtu testēšana balstās uz anketām, tad ir nepieciešams arī izvēlēties piemērotu laiku, lai anketas cilvēkiem dotu pildīt. Šis process var būt diezgan dārgs, jo kavē daudzu cilvēku darbu, tādēļ ir būtiski izsvērt, cik daudz pasūtītājs ir gatavs ieguldīt līdzekļus, lai uzlabotu sistēmas lietojumkvalitāti un noteiktu nepieciešamās robežšķirtnes.

Tā kā katrs produkts ir atšķirīgs no citiem, nav vienota standarta, kādas metrikas nepieciešams iegūt, lai novērtētu sistēmas lietojumkvalitāti. Katra situācija jāvērtē atsevišķi, ņemot vērā, kas sistēmu lietos un kādi lietojumkvalitātes aspekti konkrētajā situācijā ir vissvarīgākie. Viens no efektīvākajiem testēšanas veidiem ir lietotājcentrētā testēšanas pieeja, kurā tiek izvēlēts testa dalībnieks, kas nav no izstrādes komandas, bet atbilst nepieciešamajam mērķa lietotāja profilam, un tiek novērots darbībā [12].

Vislielākais popularitātes pieaugums starp zinātniskajiem rakstiem meklēšanas vietnē Google Scholar [4] ir vērojams terminiem “kontekstu pārklājums” un “veselības un drošības riska mazinājums”, kas būtu skaidrojams ar to, ka šāds termins ir izdalīts atsevišķi sākot ar ISO/IEC 25010 [1] standartu 2011. gadā, bet plašāks skaidrojums par atribūta testēšanu ir atrodams jaunākajā standartā ISO/IEC 25022 [2], kas tika publicēts 2016. gadā, bet tādiem terminiem, kas bijuši izdalīti atsevišķi jau iepriekšējos standartos, popularitāte aug mazāk strauji.

Google Scholar [4] vietnē pētot pieaugumu pa gadiem (skatīt rezultātu sadaļā 1. un 2. attēlu), ir redzams, ka pieaugums ar katru gadu sarūk tieši tiem terminiem, kas ir tikuši izdalīti kā atsevišķa grupa jau iepriekš. Tie ir tādi zināmi testēšanas aspekti kā rezultativitāte, efektivitāte, apmierinātība, noderīgums, uzticēšanās, patika, ērtums un pielāgojamība.

Terminiem, kas ir ieviesti standartos ISO/IEC 25010 [1] un ISO/IEC 25022 [2] kā atsevišķi atribūti, lielākais pieaugums vērojams 2015. un 2016. gada laikā, kas sakrīt ar to laiku, kad jaunākais standarts tika izveidots un līdz ar to informācija par šo atribūtu testēšanu kļuva pieejamāka. Lielākais pieaugums šajā gadā ir novērojams terminiem – ”riskbrīvība”, “ekonomisko risku mazinājums”, “kontekstu pārklājums”, “kontekstu pilnīgums”.

Savukārt bloga rakstu meklēšanas vietnē Blog Search Engine [5] visiem terminiem ir vērojams līdzīgs pieaugums, kas varētu būt tādēļ, ka cilvēki nemitīgi interesējas par dažādām tēmām, neskatoties, cik senas vai jaunas tās ir.

Vislielākais klāsts ar pieejamo informāciju bija tiem lietojumkvalitātes atribūtiem, kas bija zināmi jau standartā ISO/IEC 9126 [9] un senāk, taču tādiem atribūtiem, kas tika ieviesti sākot ar standartu ISO/IEC 25010 [1], pieejamās informācijas apjoms bija krietni mazāks vai arī pārsvarā balstījās uz jaunākajiem standartiem.

IZMANTOTĀ LITERATŪRA UN AVOTI

1. **ISO/IEC 25010:2011**, *Systems and software engineering Systems and software Quality Requirements and Evaluation (SQuaRE) - System and software quality models*
2. **ISO/IEC 25022:2016**, *Systems and software engineering -Systems and softwrae quality requirements and evaluation (SQuaRE) - Measurement of quality in use*
3. **ISTQB**, *Standard Glossary of Terms used in Software Testing* [online]. [Cited 15.12.2018]. Available: <https://www.istqb.org/downloads/send/20-istqb-glossary/186-glossary-all-terms.html>
4. *Google Scholar* [online]. [Cited 30.10.2018]. Available: <https://scholar.google.lv/>
5. *Blog Search Engine* [online]. [Cited 30.10.2018]. Available: <http://www.blogsearchengine.org>
6. **Alice, K.**, *Top 11 Trusted (And Free) Search Engines For Scientific and Academic Research* [online]. [Cited 30.10.2018]. Available: <https://www.emergingedtech.com/2013/12/top-11-trusted-and-free-search-engines-for-scientific-and-academic-research/>
7. *Citeseerx* [online]. [Cited 30.10.2018]. Available: <http://citeseerx.ist.psu.edu/>
8. *Microsoft Academic* [online]. [Cited 30.10.2018]. Available: <https://academic.microsoft.com/home>
9. **ISO/IEC 9126-1:2001**, *Software engineering -- Product quality -- Part 1: Quality model*
10. **Rubin, J., Chisnell, D.**, *Handbook of Usability Testing: How to Plan, Design, and Conduct Effective Tests*, Indianapolis: Wiley Publishing, Inc., 2008. 348 p., ISBN 978-0-470-18548-3
11. **Agarwal, B. B., Tayal, S. P., Gupta, M.**, *Software Engineering & Testing*, Sudbury: Jones and Bartlett Publishers, 2010. 518 p., ISBN 978-1-934015-55-1
12. **John, V., Done, H.**, *Principles and Practice of Software Testing: Insights into Testing*, Bilaspur: Educreation Publishing, 2018. 172 p., ISBN 978-1-5457-2957-1
13. **Spektors, A.**, *Tēzaurus* [tiešsaiste]. [Skatīts 17.02.2019]. Pieejams: <http://www.tezaurus.lv>
14. **Ingeno, J.**, *Software Architect's Handbook*, Birmingham: Packt Publishing Ltd., 2018. 554 p., ISBN 978-1-78862-406-0
15. **Mifsud, J.**, *Usability Metrics – A Guide To Quantify The Usability Of Any System* [online]. [Cited 08.01.2019]. Available: <https://usabilitygeek.com/usability-metrics-a-guide-to-quantify-system-usability/>
16. *1a.lv*, [tiešsaiste]. [Skatīts 24.04.2019]. Pieejams: <https://www.1a.lv/>
17. *Cenuklubs*, [tiešsaiste]. [Skatīts 24.04.2019]. Pieejams: <https://cenuklubs.lv>

18. **Kneuper, R.**, *Software Processes and Life Cycle Models*, Basel: Springer Nature Switzerland AG, 2018. 344 p. ISBN 978-3-319-98844-3. e-ISBN 978-3-319-98845-0. Available from: doi: 10.1007/978-3-319-98845-0
19. **Kohl, J.**, *Tap Into Mobile Application Testing*, Victoria: Leanpub, 2017. ISBN 978-0-9959823-2-1
20. **Guru99**, *Interrupt Testing in Mobile Application* [online] [Cited 02.03.2019]. Available: <https://www.guru99.com/interrupt-testing.html>
21. **Software testing help**, *Top 10 Integration Testing Tools to Write Integration Tests* [online]. [Cited 06.04.2019]. Available: <https://www.softwaretestinghelp.com/integration-testing-tools/>
22. **Chen, J.**, *Return on Investments* [online]. [Cited 24.02.2019]. Available: <https://www.investopedia.com/terms/r/returnoninvestment.asp>
23. *Business Dictionary* [online]. [Cited 24.02.2019]. Available: <http://www.businessdictionary.com>
24. **Valsts kanceleja**, *Līdzsvarotās vadības karte* [tiešsaiste]. [Skatīts 20.01.2019]. Pieejams: <http://tap.mk.gov.lv/valsts-parvaldes-politika/kvalitates-vadiba/Kvalitates-vadibas-modeli/Lidzsvartas-vadibas-karte/>

PIELIKUMI

1. pielikums

Pirmkods statistikas datu ieguvei valodā Python

```
import requests
from bs4 import BeautifulSoup
import re

terms = [
    "Quality in use" software testing',
    "Effectiveness" software testing',
    "Efficiency" software testing',
    "Satisfaction" software testing',
    "Usefulness" software testing',
    "Trust" software testing',
    "Pleasure" software testing',
    "Comfort" software testing',
    "Freedom from risk" software testing',
    "Economic risk mitigation" software testing',
    "Health and safety risk" software testing',
    "Environmental risk mitigation" software testing',
    "Context coverage" software testing',
    "Context completeness" software testing',
    "Flexibility" software testing'
]

def get_counts():
    for term in terms:
        r = requests.get('https://scholar.google.lv/scholar', params={'q': term})

        soup = BeautifulSoup(r.text)

        n_str = re.findall(r'[\d\s]+', soup.find('div', {'id': 'gs_ab_md'}).text)[0]
        n = int("".join(n_str.split()))

        yield n

counts_scholar = list(get_counts())
print(counts_scholar)
```

Meklēšanai izmantojamās frāzes

Termins	Meklējamā frāze
Quality in use	"Quality in use" software testing
Effectiveness	"Effectiveness" software testing
Efficiency	"Efficiency" software testing
Satisfaction	"Satisfaction" software testing
Usefulness	"Usefulness" software testing
Trust	"Trust" software testing
Pleasure	"Pleasure" software testing
Comfort	"Comfort" software testing
Freedom from risk	"Freedom from risk" software testing
Economic risk mitigation	"Economic risk mitigation" software testing
Health and safety risk	"Health and safety risk" software testing
Environmental risk mitigation	"Environmental risk mitigation" software testing
Context coverage	"Context coverage" software testing
Context completeness	"Context completeness" software testing
Flexibility	"Flexibility" software testing

Bakalaura darbs “Programmatūras lietojumkvalitātes metrikas un testēšanas veidi”
izstrādāts LU Datorikas fakultātē.

Ar savu parakstu apliecinu, ka pētījums veikts patstāvīgi, izmantoti tikai tajā norādītie
informācijas avoti un iesniegtā darba elektroniskā kopija atbilst izdrukai.

Autors: Sintija Ustupe

Rekomendēju/nerekomendēju darbu aizstāvēšanai

Vadītājs: docente Dr.dat. Vineta Arnicāne

24.05.2019.

Recenzents: profesors Dr.dat. Māris Vītiņš

Darbs iesniegts Datorikas fakultātē 24.05.2019.

Dekāna pilnvarotā persona:

Darbs aizstāvēts bakalaura gala pārbaudījuma komisijas sēdē

05.06.2019. prot. Nr.

Komisijas sekretāre: