



DOKUMENTŲ VALDYMO SISTEMOS

(Lietuvos IS DVS) PROJEKTAS

Testavimo planas

Paruošė: Kristina Dubikovienė T18/1

Aurimas Vasauskas T18/1

Justina Mockaitytė T18/1

2019

Ižanga

Ižangoje apžvelgiamas Lietuvos **dokumentų valdymo sistemos** (toliau **DVS IS**) testavimo planas, kuris apima testuojamą programinę įrangą ir su ja susijusias sąsajos dalis.

Testavimo plane pateikiamos testavimo strategijos, kurios apima šias veiklas: *vienetų testavimą, integravimo testavimą, sistemos testavimą, regresinį testavimą, priėmimo testavimą, dokumentacijos testavimą, našumo testavimą* (tik keliais aspektais), *grafinės vartotojo sąsajos testavimą* (panaudojamumo, tinkamumo aspektu).

Kita svarbi veikla aptariama šiame dokumente yra **dokumentų valdymo sistemos** testavimo procedūros, žvelgiant iš testavimo procedūrų perspektyvos. Testavimo plane aprašomos pateiktų testavimo metodų (*vienetų testavimo, integravimo testavimo, priėmimo testavimo, sistemos testavimo, regresinio testavimo, dokumentacijos testavimo, našumo testavimo* (tik keliais aspektais), *grafinės vartotojo sąsajos testavimo*) procedūros.

Testavimo procedūrų skyriuje aprašoma kaip bus paskirstyti testavimo resursai testavimo etapo metu bei kaip bus kaupiami testavimo rezultatai.

Testavimo plane trumpai apžvelgiamas ir programinės įrangos kūrimo procesas, kuris remiasi iteratyviuoju - inkrementiniu (angl. Agile) metodologijos principu.

1.1. Tikslai ir objektai

Pasirinkti testavimo metodai turi užtikrinti, kad kuriama programinė įranga (Lietuvos DVS IS):

- atitinka reikalavimų specifikaciją;
- surastos ir ištaisytos programavimo bei logikos klaidos;
- patikrintas atskirų sistemos dalių funkcionalumas;
- sumažintas rastų klaidų skaičius iki minimumo;
- paprastas sistemos naudojimas (intuityvi vartotojui);
- sistema yra efektyvi realiame laike;
- sistema atitinka saugumo lygį;
- korektiškai vyksta bendravimas su kitomis sistemomis (duomenų perdavimas naudojant HTTP protokolą paslaugų tiekėjui).

Išskiriami du pagrindiniai **testavimo tikslai**: **patikrinti** (angl. verify) ar dokumentų valdymo sistema veikia teisingai ir **patvirtinti** (angl. validate), kad dokumentų valdymo sistema atlieka tai, ko iš jos tikimasi.

Pažymima, kad bus taikomi tokie testavimo atvejai, kurie leis užtikrinti minimalų informacinės sistemos klaidų kiekį ir aukštą kuriamos programinės įrangos kokybę.

Rastas klaidas įsipareigojama taisyti viso projekto įgyvendinimo metu (2 mėn.)

1.2. Testavimo srities apibrėžimas

Lietuvos DVS IS programinė įranga bus testuojama 7 skirtingais būdais:

- vienetų testavimas;
- integravimo testavimas;
- sistemos testavimas;
- regresinis testavimas;
- priėmimo testavimas;
- dokumentacijos testavimas;
- našumo testavimas (tik keliais aspektais).

Testavimo metu bus nustatyta:

- ar patogus sistemos valdymas;
- ar vartotojui lengva orientuotis sistemoje;
- ar duomenų skaitymas iš duomenų bazės vyksta be sutrikimų;
- ar korektiškai į sistemą talpinamas ir atvaizduojamas dokumento turinys;
- ar prieėjimas prie dokumentų yra galimas tik autorizuotiems vartotojams.
- ar naudojami duomenų šifravimo algoritmai.

1.3. Pagrindiniai sistemos reikalavimai ir apribojimai

1.3.1. Sistema sukurta naudojant:

- Java (8+) programavimo kalbą;
- Spring (5+) karkasą (užklausų valdymo serveris);
- React (16+) karkasą (sistemos vartotojo sąsaja);
- MySQL realiacinę duomenų bazę (duomenų saugojimas);
- Maišos funkciją (ne mažesnė nei SHA-256 saugumo lygio);

Su Lietuvos DVS IS kita susijusi programinė įranga, testavimo įrankiai:

- Windows operacinė sistema;
- Internetinis ryšys;
- Chrome internetinė naršyklė;
- Selenium (automatinis įrankis);
- JUnit (standartinis testavimo karkasas);
- Spiretest (rankinis testavimas, klaidų registravimas);
- Apache JMeter™ (našumo testavimas);
- Testpad (rankinis testavimas);

1.3.2. Pagrindiniai apribojimai:

Testavimas vykdomas Kauno informacinių technologijų mokykloje, paruoštoje tam skirtoje aplinkoje. Darbo aplinka reikalauja turėti tam tikrų techninių įgūdžių ir testavimui skirtų žinių bei specialių testavimo įrankių. Testavimo laikas ribojamas, testuojama darbo dienomis nuo 8 val. iki 17 val. Kitu laiku Kauno informacinių technologijų mokykla būna uždaryta.

2.1. Testuojama programinė įranga

2.1.1. Sąsajos

Pagrindinis langas:

Langas skirtas parodyti pagrindines sistemos funkcijas.

Tikrinama ar vartotojas gali patekti iš pagrindinio sistemos lango į kitus sistemos langus. Ar vartotojas gali peržiūrėti tik jam leistus prieiti duomenis, ar gali prieiti ir prie duomenų, kurių prieigai nėra autorizuotas?

Registracijos langas:

Sąsaja skirta vartotojui prisiregistruoti prie sistemos.

Tikrinama, ar sėkmingai vartotojas gali prisiregistruoti prie sistemos. Ar visi formos laukeliai turi būti užpildyti? Ar sistema perspėja, jei trūksta įvedamų duomenų? Ar sistema tikrina įvestų duomenų korektiškumą? Ar sistema tikrina konkrečius nustatytus apribojimus įvedamiems duomenims?

Prisijungimo langas:

Sąsaja skirta vartotojui prisijungti prie sistemos.

Tikrinama ar sėkmingai vartotojas prisijungia. Ar negali prie sistemos prisijungti neautentifikuoti vartotojai?

Prisijungimo lange bus bandoma prisijungti teisingais ir neteisingais vartotojų vardais ir slaptažodžiais. Sistema turi leisti prisijungti tik su egzistuojančiu duomenų bazėje vartotojo vardo ir slaptažodžio kombinacija. Kitu atveju sistema turi išvesti pranešimą apie klaidą ir leisti pakartotinai įvesti prisijungimo duomenis.

DVS dokumentų pateikimas:

Sąsaja skirta vartotojui pateikti dokumentus į DVS IS.

Ar sėkmingai ir korektiškai į sistemą yra talpinami DVS dokumentai? Ar pateiktus į sistemą dokumentus gali matyti tik tam tikras teises turintys vartotojai? Ar yra dokumentų filtravimo paieškos funkcija vartotojo patogumui?

Administratoriaus pagrindinis langas:

Sąsaja skirta administruoti naudotojus ir grupes.

Tikrinama, ar administratorius turi galimybę sukurti, šalinti vartotojų grupes?

Tikrinama, ar administratorius turi galimybę pridėti, šalinti vartotojus grupėse?

Sąsaja skirta dokumentų šablonams kurti.

Tikrinama, ar administratorius turi galimybę sukurti, šalinti dokumentų šablonus?

2.1.2. Kita įranga

Duomenų perdavimas/gavimas į sistemą:

Komponentas skirtas automatiniam duomenų užkrovimui į kitas sistemas.

Reikia ištestuoti ar sėkmingai perduodami duomenys (naudojant POST metodą) į tarnybinę stotį.

Reikia ištestuoti ar sėkmingai gaunami duomenys (naudojant GET metodą) į tarnybinę stotį.

2.2. Testavimo strategija

2.2.1. Vienetų testavimas

Atskiriems programinės įrangos moduliams/komponentams bus naudojamas ***vienetų testavimas***.

Testai bus rašomi dviejų tipų: „***baltos dėžės***“ – kai bus atsižvelgiama į testavimo įrankio programinį kodą ir „***juodos dėžės***“ – kai testai bus sudaromi pagal specifikaciją.

Kiekvienam komponentui/moduliui bus įvedami įvairūs testiniai duomenys ir stebimas tų duomenų korektiškumas (ar komponentas/modulis veikia teisingai).

Tai bus atliekama su standartiniu testavimo karkasu – ***JUnit***, kuris yra pritaikytas naudoti Java objektiniai programavimo kalbai.

2.2.2. Integravimo testavimas

Atlikus vienetų testavimą, vienetai bus apjungiami į bendrą sistemą ir testuojama jų sąveika. Integracinio testavimo metu bus tikrinama tarpusavio integracija tarp vartotojo sąsajos. Visi rasti programinės įrangos nesuderinamumai bus užfiksuojami.

2.2.3. Sistemos testavimas

Atlikus integracinį testavimą ir ištaisius rastus defektus bus atliekami sistemos testai, kad būtų nustatyta ar sistema atitinka funkcinius ir nefunkcinius reikalavimus.

2.2.4. Regresinis testavimas

Regresinis testavimas bus vykdomas po programinės įrangos modifikavimo, pakartotinai panaudojant jau sukurtus testus. Testų kartojimas vykdomas siekiant patvirtinti, jog sistemoje neatsirado naujų defektų nekeistose vietose (išskyrus pakeitimus, kurie buvo numatyti modifikuojant PĮ). Toks modifikuotas programos testavimas leidžia sumažinti laiko ir išteklių kiekį bei užtikrinti programinės įrangos kokybę.

2.2.5. Priėmimo testavimas

Priėmimo testavimas bus atliekamas, kai visa programinė įranga jau sukurta. Testavimui į pagalbą pasitelkiamas „***juodos dėžės***“ principas. Sistema bus nuodugniai ištestuota atsižvelgiant į reikalavimų specifikaciją, patikrinant ar sistema atitinka vartotojo poreikius. Jei programinė įranga specifikacijos neatitiks, tuomet bus pildoma klaidų ataskaita.

Kai programinę įrangą pradės testuoti būsimi sistemos DVS IS naudotojai, tuomet jie galės pateikti apie sukurta sistemą savo pastebėjimus ir siūlymus.

2.2.6. Dokumentacijos testavimas

Vartotojas galės naudotis paruošta vartotojo dokumentacija („*vartotojo vadovo instrukcija*“), kuri leis geriau suprasti ir išmokyti dirbti su naujai sukurta Lietuvos DVS IS sistema.

2.2.7. Našumo testavimas (tik keliais aspektais)

Našumo testavimas bus atliekamas bent kartą tam, kad būtų įsitikinta, jog programinė įranga veikia teisingai. Toks testavimo būdas turėtų parodyti pokytį tarp reakcijos laiko ir našumo santykio.

2.3. Atsakomybės ir rolės

Sistemos testavimą atlieka testuotojų komanda:

- testavimo specialistas – Aurimas Vasauskas (vykdo sistemos testus, atsakingas už testavimo techninį realizavimą ir gautus rezultatus, bendravimą su programuotojų komanda, dokumentacijos rengimą);
- testavimo specialistė – Kristina Dubikovienė (vykdo sistemos testus, atsakinga už testavimo plano rengimą, testavimo įrankių paieškos analizę ir sėkmingą jų naudojimą, dokumentacijos rengimą);
- testavimo specialistė – Justina Mockaitytė (vykdo sistemos testus, atsakinga už testavimo atvejų rengimą ir sėkmingą jų vykdymą, dokumentacijos rengimą).

2.4. Testavimo grafikas

Nr.	Testavimo etapas	Rezultatai	Terminas
1.	Testavimo planas	Parašytas testavimo planas	Vasario 15
2.	Pradedamas testavimas	Ištestuota sistema, užregistruoti defektai	Vasario 15
3.	Programinės įrangos rastų defektų šalinimas	Atnaujinta PĮ.	Vasario 18- Kovo 22
4.	Pakartotinis testavimas	Ištestuota sistema. Užregistruoti defektai, parašyta ataskaita.	Kaskart atnaujinus IS (sprintai)
5.	Sistemos testavimo rezultatų apibendrinimas ir užbaigimas	Užpildyta galutinė testavimo ataskaita, sistema paruošta naudojimui.	Kovo 28

2.5. Testų rezultatų sekimas ir matavimas

Visi randami defektai fiksuojami, aprašomi bei pateikiamos galutinės testinių atvejų ataskaitos. Testų efektyvumas bus matuojamas atsižvelgiant į rastų defektų ir testo vykdymo laiko santykį. Tam bus vedamas testavimo protokolas (test-log), kurio struktūra bus sudaryta iš:

- Testavimo datos;
- Asmens atlikusio testus;
- Kokie testai buvo atliekami (jų pavadinimai/numeriai);
- Testuoto produkto ir jo konfigūracijos;
- Kiekvieno testo pradžios (laikas);
- Kiekvieno testo pabaigos (laikas);
- Aptiktų defektų skaičiaus.

Planuojama į regresijos testų paketą įtraukti: visus testus, kurie anksčiau yra aptikę defektų, ir visus testus, kurie tikrino tas sistemos sritis, kurios buvo modifikuotos vėlesniuose programinės įrangos kūrimo cikluose.

2.6. Stabdymo ir grįžimo kriterijai

Jeigu atlikus 5 pirmus testuojamo modulio atvejus, randama daugiau kaip 10 defektų, tuomet testavimas laikomas sustabdytu ir tęsiamas tik tuomet, kai aptikti sistemos defektai ištaisomi kuriančių atsakingų programuotojų.

2.7. Kokybės kriterijai ir vertinimas

Norint nustatyti ir įvertinti programinės įrangos kokybę bus atsižvelgiama į tarptautinį **ISO/IEC 25010:2011** programinės įrangos kokybės standartą, kuris apima aštuonias reikalavimų ir vertinimo kokybės charakteristikas, tokias kaip:

- Funkcionalumas (Ar PĮ-oje realizuotos visos funkcijos?);
- Patikimumas (Kiek patikima yra PĮ-a);
- Panaudojamumas (Ar lengva naudotis PĮ-a);
- Veikimo našumas (Ar pakankamas programos veikimo greitis?);
- Palaikomumas (Kaip lengva pakeisti PĮ-ą?);

- Perkeliamumas (Kaip lengva PĮ-ą perkelti į kitą aplinką?);
- Saugumas (Ar pakankamas saugumo lygis?);
- Suderinamumas (Ar yra galimybė naudotis kitomis PĮ sistemos versijomis?).

Šiame testavimo plane išsamiau bus analizuojama kokybės **panaudojamumo** charakteristika (Kiek pastangų reikia įdėti naudojant PĮ), atsižvelgiant į **panaudojamumo rekomendacijos standartą ISO 9241-11** (standarto „Žmogaus ir kompiuterių sąveikos ergonomika“ 11-a dalis) bei aptariant panaudojamumo kriterijus.

Laikomasi šių panaudojamumo kriterijų:

- Išmokstamumas (nusako, ar lengva naudotis sistema);
- Lankstumas (rodo, kaip naudotojo sąsaja prisitaiko prie skirtingų naudotojo įgūdžių lygių);
- Našumas (rodo, naudotojo sąsajos atžvilgiu užduoties atlikimo greitį);
- Atsakas, grįžtamasis ryšys („neatsilikti nuo naudotojo“, pateikti esamą būseną ir neleisti naudotojui laukti nežinant, kas vyksta);
- Įsimenamumas (parodo, ar lengva sistemos naudotojui atsiminti, kokius veiksmus atlikti,);
- Saugumas (rodo, ar sistema yra atspari klaidingiems naudotojo veiksams ir ar lengvai atstatoma sistemos būsena);
- Estetika (nusako spalvų suderinamumą, unikalių grafinių elementų vaizdavimą, išdėstymą);
- Natūralumas (apibrėžia, kiek naudojama veiksmų seka, tekstai, metaforos ir t.t. yra įprasti ir natūralūs naudotojui);
- Portabilumas (parodo, sistemos išvaizdos vienodumą ir vientisumą nepriklausomai nuo įrenginio).

Programinės įrangos kokybės kriterijai bus įvertinami pagal:

- Programinės įrangos **naudotojų vartojamumo** apklausą;
- Programinės įrangos kūrėjų apklausą („**Savęs paties įvertinimas**“).

„**Naudotojų vartojamumo įvertinimas**“ naudojamas kaip vienas iš testavimo įrankių, kurio pagalba bus siekiama suprasti ir įvertinti naudotojo patirtį dirbant su programine įrangos sistema. Tam bus pasitelkiamas programinės įrangos kokybės standarto modelis - **ISO/IEC 25010:2011**, kuris leis tinkamai paruošti anketos klausimus. Galiausiai, remiantis gautais anketos rezultatais bus atliekamas išsamios analizės tyrimas ir pateikiami siūlymai programinės įrangos sistemos tobulinimui.

Taip pat naudotojams bus paruoštos nesudėtingos užduotys, kurių dėka bus galima įvertinti ir pamatyti sistemoje atsiradusius neatitikimus ir juos ištaisyti kuo ankstesniame programinės įrangos kūrimo etape.

Programinės įrangos *kūrėjų apklausa „Savęs paties įvertinimas“*, tai pačių kūrėjų kuriamos programinės įrangos įsivertinimas, kuris kaip ir naudotojų vartojamumo apklausos atveju, remiasi programinės įrangos kokybės standarto *ISO/IEC 25010:2011* modeliu. Rezultatus gautus iš „*Savęs paties įvertinimas*“ apklausos bus galima palyginti su gautais rezultatais iš „*Naudotojų vartojamumo įvertinimas*“ apklausos.

Testavimo metu rasti neatitikimai/nesuderinamumai bus klasifikuojami į:

- Kritinius;
- Svarbius;
- Nereikšmingus.

Laikomasi prielaidos, kad jeigu per testavimą yra randama mažiau kaip 7 nereikšmingi defektai ar nė vieno kritinio defekto, tai produktas turi nedidelių trūkumų.

Laikomasi prielaidos, kad jeigu per testavimą randama 7 - 15 nereikšmingi defektai ar keli svarbūs defektai, bet jų randama ne daugiau kaip 5, bet nė vieno kritinio defekto, tai produktas turi trūkumų, bet juo pilnai galima naudotis.

Laikomasi prielaidos, kad jeigu per testavimą yra randama daugiau kaip 15 nereikšmingų defektų ir 1 ar daugiau kritinių defektų, bet ne daugiau kaip 5 svarbūs defektai, tai produktas laikomas netinkamu naudotis.

Laikomasi prielaidos, kad jeigu testavimo metu nėra rasta kritinių klaidų, o testuojamas funkcionalumas padengia didžiąją dalį (70%-80%) suplanuoto testavimui skirtą funkcionalumą, tai produktas laikomas sėkmingu ir tinkamas naudoti.

3. Testavimo procedūra

3.1. Testuojama programinė įranga

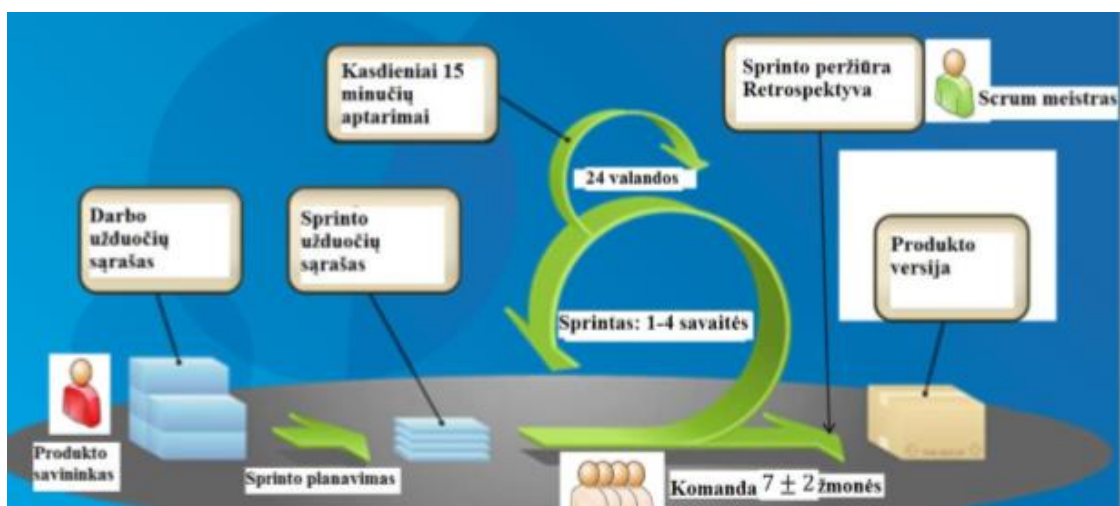
Testuojama įranga aprašyta 2.1. punkte.

3.2. Testavimo procedūros

Šiame etape nurodomos kokios bus testavimo procedūros, testavimo metodai, laukiami ir gaunami rezultatai bei rezultatų saugojimas (ataskaitos) bus pateikiami testavimo plano priede.

Testavimo procesas yra vienas svarbiausių procesų programinės įrangos kūrime, tad norint užtikrinti programinės įrangos testavimo proceso valdymą bei jo automatizavimą, reikia atsižvelgti ne tik į testavimo procesą, bet ir visą programinės įrangos kūrimo procesą. Todėl dažnai prieš kuriant programinę įrangą yra stengiamasi įvertinti ir pasirinkti tinkamiausią programinės įrangos kūrimo metodologiją ir metodus bei jais vadovautis.

Būtent dėl to, kuriant **dokumentų valdymo sistemą** yra numatyta taikyti iteratyvųjį - implementinį (angl. Agile) programinės įrangos sistemos kūrimo metodą, kurio pagrindinis bruožas – kūrimas iteracijomis. Tai tarsi suteikia galimybę didelius ir kompleksinius programinės įrangos kūrimo dalis padalinti į mažesnius etapus ir nuosekliai pereiti nuo vieno etapo prie kito, greitai reaguojant į atsiradusius pokyčius.



Pav. 1 iteratyvusis - inkrementinis (angl. Agile) metodologijos principas

Dažnai kuriant programinę įrangą užsakovai patys inicijuoja įvairius pokyčius ir tai yra Agile metodo galimybė, kuri leidžia greitai prisitaikyti prie pokyčių ir gauti norimus rezultatus. Tai ypač aktualu užsakovui, kurio siekis – lūkesčių išpildymas bei pasitenkinimas turima programine įranga. Agile esminis progreso rodiklis – veikiantys programos komponentai, kurie nuolatos yra vis papildomi. Žinoma, ne visada Agile metodo pasirinkimas yra pats tinkamiausias pasirinkimas, kai kuriais atvejais dėl saugumo yra siūloma rinktis taip vadinamąjį tradiciniu laikomą (Krioklio) metodą.

Kuriant programinę įrangą - Lietuvos DVS (IS) sistemą yra būtent remiamasi pasirinktu iteratyviuoju - implementiniu (ang. Agile) metodologijos principu, kurio laikomasi visos sistemos kūrimo laikotarpiu.

3.2.1. Vienetų testavimas

Vienetų testų komponentai bus testuojami paduodant jiems įėjimo duomenis ir stebint jų išėjimus ir juos lyginant su laukiamais rezultatais, paliekant galimybę juos pakartotinai įvykdyti. Todėl vienetų testai ir netikri objektai bus naudojami kartu. Naudojant netikrus objektus tarsi bus izoliuojami testuojami vienetai nuo priklausomybių, stebimos sąveikos, taip suteikiant galimybę testus leisti norimą kiekį kartų.

Testinis atvejis turėtų susidaryti iš tokių komponentų, kuris leistų iškviešti metodą testo eigos pradžioje ir perrašyti, kai norima sukurti objektus ar inicializuoti kintamuosius (metodas kviečiamas prieš kiekvieną testą). Taip pat testinis atvejis galės būti perrašomas tuomet, kai reikės atlaisvinti resursus naudotus inicializacijoje ar testavimo metu (metodas kviečiamas po kiekvieno testo).

Testų vykdymo metu testų sąlygos, šalutinis efektas ar metodo grąžinami rezultatai bus lyginami su tikėtiniais rezultatais. Tam bus pasinaudota standartinio testavimo karkaso JUnit galimybe, kuri leidžia naudojant `assertEquals()` metodą ir palyginti gautus aktualius rezultatus su tikėtiniais ir iškviešti išimtis, kai sąlygos nėra įvykdytos.

Testuojamas komponentas: serverio sąsaja

Metodas	Įeinantys duomenys	Rezultatas kurio tikimasi	Testavimo rezultatas
createDoc	docName	Boolean: true	Klaidos nėra.

3.2.2. Integravimo testavimas

Šis testavimas bus atliekamas su atskirais sistemos komponentais, kurie po vienetų testavimo bus apjungiami į didesnius komponentus. Pasirinktas naudoti „iš apačios į viršų“ (angl. „bottom-top“) integravimo testavimas. Integraciniame testavime dalyvaus vartotojo pusės kodo ir serverio pusės integracija. Vartotojo sąsaja yra atsakinga už bendradarbiavimą su serveriu, todėl reikės ištestuoti, ar serveriui pateiktų užklausų duomenys yra tinkamai pateikiami ir atvaizduojami vartotojo sąsajoje get ir post metodais.

Pirmiausiai, bus sujungti ir ištestuoti registracijos ir prisijungimo komponentai.

3.2.3. Sisteminis testavimas

Sisteminis testavimas bus atliekamas po integracijos testavimo, kai visi kuriamos sistemos komponentai ir moduliai bus apjungti tarpusavyje. Sisteminis testavimas – tai visa apimančių

komponentų sąveikos testų vykdymas, norint patikrinti, ar apjungus visus komponentus ir modulius sistema veikia kaip visuma, ar atitinka visus jai keliamus reikalavimus.

Sisteminis testavimas bus atliekamas „juodosios dėžės“ principu, testuojant iš vartotojo perspektyvos ir ignoruojant programos kodą. Todėl sisteminį testavimą galima pradėti užbaigus ir patvirtinus dokumentaciją, pagal kurią jau galima parašyti testus.

Sisteminio testavimo metu bus:

- dar kartą peržiūrima dokumentacija (ar esanti sistema atitinka jai keliamus reikalavimus);
- tikrinamas sistemos našumo veikimas;
- tikrinamas sistemos saugumas (ar duomenys yra apsaugoti);
- tikrinama vartotojo sąsaja (ar integruoti Lietuvos DVS IS sistemos komponentai veikia pagal specifikaciją);
- tikrinama, ar vartotojui nekyla sunkumų naudojantis kuriama sistema;
- tikrinamas sistemos atsparumas stresui ir veikimas esant skirtingam jos apkrovimui.

Šis testavimo etapas yra vienas svarbiausių, nes būtent šio testavimo metu randama daugiausiai nesutapimų tarp sistemai keliamų reikalavimų, specifikacijų ir kodo įgyvendinimo. Rasti defektai bus registruojami ir perduoti atsakingiems asmenims. Ištaisius visas klaidas, sistema bus dar kartą ištestuota.

3.2.4. Regresinis testavimas

Kiekvieną kartą ištaisius rastus DVS sistemos defektus ar įdiegus naują funkcionalumą bus atliekamas regresinis testavimas, norint įsitikinti, kad po atliktų pakeitimų kuriama sistema tinkamai veikia ir ar neatsirado naujų defektų (nebuvo paveiktas senas kodas). Šiam tikslui pasiekti atliekami pakartotini testai, tokie kaip:

- testavimo atvejai, kuriuose buvo rasta defektų ir yra ištaisyti;
- ribinių reikšmių testavimas;
- sėkmingai praėję testai.

3.2.5. Priėmimo testavimas

Priėmimo testavimas bus atliekamas jau sukūrus DVS IS sistemą ir atitikus nustatytus testavimo plano kokybės kriterijus. Priėmimo testavimas bus vykdomas „juodosios dėžės“ principu dalyvaujant trečiai nepriklausomai šaliai pagal iš anksto paruoštus testavimo scenarijus, nuosekliai

atliekant visus nurodytus žingsnius. Tokiu būdu DVS IS sistema bus nuodugniai ištestuota pagal reikalavimų specifikaciją ir patikrinta, ar sistema atitinka vartotojo poreikius ir sistemai keliamus reikalavimus.

Priėmimo testavimo metu bus orientuojamasi į sistemos funkcinius ir nefunkcinius reikalavimus. Šis testavimo metodas labai panašus į sisteminį testavimą. Pagrindinis skirtumas tas, kad priėmimo testavimas vykdomas natūralioje (ne testuojamoje) aplinkoje nepriklausomų vartotojų naudojant kuo realesnius duomenis.

3.2.6. Dokumentacijos testavimas

Bus analizuojamas visų sistemos savybių/funkcijų aprašymas dokumentacijoje bei kaip suprantamai pateikiamas turinys kiekvienam sistemos vartotojui.

Rengiant vartotojo dokumentaciją bus atsižvelgiama į:

- Dokumento struktūrą (ar dokumente išdėstytas produkto turinys ir ką vartotojas gali daryti su šiuo produktu?);
- Skaitomumą (ar dokumentas lengvai skaitomas?);
- Suprantamumą (ar žmonės neturintys specifinių žinių gali suprasti tai, kas parašyta dokumente?).

3.2.7. Našumo testavimas (tik keliais aspektais)

Bus tikrinama, kaip greitai sistema ir sistemos komponentai (DB) atlieka įvairias jai paskirtas operacijas, tokias kaip:

- Įrašo išsaugojimas;
- Įrašo ištrynimasis;
- Įrašo redagavimas;
- Perėjimas iš vieno sistemos lango į kitą.

Taip pat bus tikrinama, kaip keičiasi našumas esant šioms būsenoms:

- Performance - įprastinė kasdieninė apkrova;
- Load - pikinė apkrova;
- Stress - didesnė nei numatoma pikinė apkrova.

3.2.8. Kokybės kriterijai ir vertinimas

Apklauso testavimas atliekamas su iš anksto paruoštais klausimais sistemos naudotojams ir sistemos kūrėjams, įtraukiant į klausimyną kokybės kriterijus atitinkančius klausimus.

Žemiau yra pateikiami dokumentų valdymo sistemos kūrėjų - „*Savęs paties įvertinimas*“ ir „*Naudotojų vartojamumo įvertinimas*“ (iš 2 – jų dalių) klausimynai, kurie apima *ISO/IEC 25010:2011* programinės įrangos kokybės standarto charakteristikų grupes, tokias kaip: *funkcionalumas, patikimumas, panaudojamumas, veikimo našumas, palaikomumas, perkeliamumas, saugumas, suderinamumas*.

Dokumentų valdymo sistemos kūrėjų apklauso („Savęs paties įvertinimas“) klausimynas:

1. Ar dokumentų valdymo sistema remiasi į žmogų orientuota sistemos kūrimo koncepcija?
2. Ar sistema grindžiama iteratyviuoju-inkrementiniu (angl. Agile) sistemos kūrimo principu?
3. Ar dokumentų valdymo sistemos užsakovui yra pateikiami imitacinio pobūdžio scenarijai (prototipai)?
4. Ar kuriant dokumentų valdymo sistemą yra remiamasi programinės įrangos kokybės modelio standartu pagal *ISO/IEC 25010:2011* reikalavimus ir vertinimus?
5. Ar galioja trijų paspaudimų principas, norint pasiekti reikiamą informaciją dokumentų valdymo sistemos naudotojo pagrindiniame puslapyje?
6. Ar dokumentų valdymo sistemos struktūra leidžia sugrįžti iš bet kurio puslapio į pagrindinį naudotojo sistemos puslapį?
7. Ar dokumentų valdymo sistemoje egzistuoja klaidos pranešimo perspėjimas?
8. Ar dokumentų valdymo sistemoje naudojamos spalvos netrukdo informacijos skaitomumui?
9. Ar dokumentų valdymo sistemoje yra galimybė nesudėtingai integruoti naujus funkcinis modulius?
10. Ar dokumentų valdymo sistemoje esančios funkcijos veikia tvarkingai ir atlieka savo pirminę paskirtį?
11. Ar dokumentų valdymo sistema atitinka minimalius saugumo reikalavimus?
12. Ar dokumentų valdymo sistemoje yra vidinė paieškos sistema, jei norima surasti konkrečią informaciją?
13. Ar pateiktų dokumentų peržiūros laikas ir būseną yra stebimi ir kontroliuojami dokumentų valdymo sistemoje?
14. Ar teisinė informacija, tokia kaip autorių teisės ir privatumo politika yra numatyta dokumentų valdymo sistemoje?

15. Ar dokumentų valdymo sistemoje publikuojama informacija nepažeidžia kitų asmenų teisių bei atitinka Lietuvos Respublikos įstatymus apie draudžiamą turinį, nepageidaujamą turinį?

Dokumentų valdymo sistemos „Naudotojų vartojamumo apklausa“ (užduotis, 1 dalis)

Norint gauti kuo išsamesnius rezultatus iš naudotojų vartojamumo pusės, bus pateikiamos papildomos užduotys, tokios kaip: **sukurti naują dokumentą**, jį **išsaugoti** ir **pateikti** už dokumentus atsakingai grupei. Vėliau, kai dokumento statusas įgiję būseną - „Priimtas“, naudotojai turi patikrinti ar jų pateikti dokumentai atsirado tarp „Pateikti“ dokumentų sąrašo. Tokios užduotys padeda išsiaiškinti ar dokumentas yra korektiškai sukuriamas ir išsaugojamas, ar pasiekiamas tikslinga auditorija, kuri atsakinga už konkrečių dokumentų tvarkymą. Taip pat svarbu įvertinti, kiek laiko naudotojas užtrunka atlikdamas vieną ar kitą užduotį. Toks testavimas tarsi leidžia atsakyti į klausimą: **ar dokumentų valdymo sistema remiasi į žmogų orientuota sistemos kūrimo koncepcija?**

Dokumentų valdymo sistemos „Naudotojų vartojamumo apklausa (klausimynas, 2 dalis)

1. Ar Jūs susiduriate su sunkumais pirmą kartą registruojantis prie dokumentų valdymo sistemos?
2. Ar Jums prisijungus prie dokumentų valdymo sistemos yra nesudėtinga naudotis sistema?
3. Ar Jums pradedant dirbti su sistema yra pateikiama vartotojo naudojimosi instrukcija, kaip reikia dirbti su sistema?
4. Ar Jus tenkina dokumentų valdymo sistemoje esanti paieškos(filtravimo) galimybė, kai norite surasti tam tikro turinio dokumentą?
5. Ar susiduriate su situacija, kuomet esate pagrindiniame naudotojo puslapyje ir negalite patekti į kitus dokumentų valdymo sistemos puslapius?
6. Ar Jums patogiu dokumentų valdymo sistemoje - sukurti, išsaugoti, redaguoti, bei pateikti dokumentus atitinkamai dokumentus priimančiai grupei ar konkrečiam tos grupės nariui?
7. Ar Jums patogiai suprantama ir pateikiama informacija, tokia kaip: spalvinė gama, šrifto dydis, grafika?
8. Ar Jūs intuityviai randate dokumentų valdymo sistemoje esančias paslaugas: prisijungimas, naujo dokumento sukūrimas, sukurto dokumento suradimas?
9. Ar Jūsų naudojamoje naršyklėje yra neiškraipomas dokumentų valdymo sistemos vaizdas?
10. Ar dokumentų valdymo sistemoje pateikiama informacija yra korektiška, be klaidų?

11. Ar Jums ilgai tenka laukti, kol bus atveriamas Jūsų sukurtas, išsaugotas ar kitaip pateiktas dokumentas?
12. Ar dokumentų valdymo sistemoje Jums pateikta informacija yra aiški, lengvai suprantama ir tenkina Jūsų poreikius bei lūkesčius?
13. Ar Jūs turite galimybę dokumentų valdymo sistemoje susisiekti su sistemos administratoriumi, jeigu Jums iškyla kokių nors nesklandumų?
14. Ar yra priežasčių dėl ko Jūs nenorėtumėte dirbti su dokumentų valdymo sistema?
15. Jūsų manymu, ar dokumentų valdymo sistemoje yra reikalinga spausdinimo funkcijos galimybė norimo dokumento atsispausdinimui?
16. Jūsų pastabos, pasiūlymai, kaip būtų galima patobulinti dokumentų valdymo sistemą?

Gauti apklausų rezultatai bus pateikiami paruoštoje ataskaitoje.

DOKUMENTŲ VALDYMO SISTEMOS REIKALAVIMAI

Funkciniai reikalavimai

FR 1. Sistemoje egzistuoja vartotojai

FR 1.1. Sistema turi leisti užsiregistruoti vartotojams.

FR 1.2. Sistema turi leisti prisijungti prie sistemos, įvedant vartotojo vardą ir slaptažodį.

FR 1.3. Sistemos administratoriui apribotos teisės kurti sistemos vartotojus.

FR 2. Sistemoje egzistuoja administratoriaus rolė

FR 2.1. Sistemoje egzistuoja speciali vartotojų rolė - administratorius.

FR 2.2. Sistemos administratorius gali kurti vartotojų grupes.

FR 2.3. Sistemos administratorius gali priskirti vartotojus į grupes.

FR 2.4. Sistemos vartotojai nepriskirti konkrečiai vartotojų grupei, neturi teisės matyti tos grupės atitinkamų dokumentų.

FR 2.5. Sistemos vartotojai nepriskirti konkrečiai vartotojų grupei, turi teisę matyti kitas sistemoje esančias grupes.

FR 2.6. Sistemos administratorius gali kurti dokumentų (tipų) šablonus.

FR 2.7. Sistemos administratorius gali priskirti grupes, kurios turi teisę kurti dokumentus.

FR 2.8. Sistemos administratorius gali priskirti grupes, kurios turi teisę priimti dokumentus.

FR 2.9. Sistemos administratorius gali priskirti grupes, kurios turi teisę atimesti dokumentus.

FR 3.0. Sistemos administratorius gali užblokuoti sistemos vartotoją.

FR 3. Sistemoje egzistuoja vartotojai suskirstyti į grupes

FR 3.1. Sistemos vartotojai gali priklausyti vienai sistemos vartotojų grupei.

FR 3.2. Sistemos vartotojai gali priklausyti kelioms sistemos vartotojų grupėms.

FR 3.3. Sistemos administratorius gali *priskirti* sistemos vartotojus į grupes. (kartojasi).

FR 4. Sistemoje galima pateikti įvairių tipų dokumentus

FR 4.1. Sistemos vartotojai gali pateikti dokumentus.

FR 4.2. Sistemoje pateikiamas kiekvienas dokumentas yra tam tikro tipo.

FR 4.3. Sistemos administratorius kuria dokumentų tipus.

FR 4.4. Sistemos administratorius nustato, kokių tipų dokumentus leidžiama kurti vartotojų grupėms.

FR 4.5. Sistemos vartotojai turi teisę kurti neribotą kiekį skirtingų tipų dokumentų.

FR 4.6. Sistemos dokumentų savybės gali kisti iš vienos būsenos į kitą.

FR 5. Dokumentai gali būti peržiūrimi, priimami ir atmetami

FR 5.1. Sistemos vartotojas, kuriantis dokumentą gali jį pateikti.

FR 5.2. Sistemos vartotojo sukurtas ir išsaugotas dokumentas įgyja būseną „Sukurtas“.

FR 5.3. Sistemos vartotojo dokumentas įgijęs būseną „Sukurtas“ matomas tik dokumento autoriui.

FR 5.4. Sistemos vartotojai priklausantys priimančiai grupei gali peržiūrėti vartotojo pateiktą dokumentą.

FR 5.5. Sistemos vartotojai priklausantys priimančiai grupei gali priimti vartotojo pateiktą dokumentą.

FR 5.6. Sistemos vartotojai priklausantys priimančiai grupei gali atmesti vartotojo pateiktą dokumentą.

FR 5.7. Dokumentą galima pateikti konkrečiam sistemos vartotojui.

FR 5.8. Dokumentą galima pateikti konkrečiai sistemos vartotojų grupei.

FR 5.9. Pateikus dokumentą konkrečiam sistemos vartotojui, tik jis gali priimti dokumentą.

FR 5.10. Pateikus dokumentą konkrečiam sistemos vartotojui, tik jis gali atmesti dokumentą.

FR 5.11. Pateikus dokumentą konkrečiai sistemos grupei, bet kuris tos grupės vartotojas gali priimti dokumentą.

FR 5.12. Pateikus dokumentą konkrečiai sistemos grupei, bet kuris tos grupės vartotojas gali atmesti dokumentą.

FR 5.13. Dokumentas sistemoje gali būti priimtas tik vieną kartą iš to paties vartotojo.

FR 5.14. Dokumentas sistemoje gali būti atmestas tik vieną kartą iš to paties vartotojo.

FR 5.15. Sistemos administratorius gali papildomai leisti sistemos vartotojui pateikti dokumentą.

FR 6. Dokumentų savybės

FR 6.1. Sistemos kiekvienas dokumentas turi turėti dokumento numerį.

- FR 6.2. Sistemos kiekvienas dokumentas turi turėti dokumento autorių.
- FR 6.3. Sistemos kiekvienas dokumentas turi turėti dokumento pavadinimą.
- FR 6.4. Sistemos kiekvieną dokumentą turi sudaryti dokumento turinio aprašas.
- FR 6.5. Sistemos kiekvienam dokumentui nurodomas dokumento tipas.
- FR 6.6. Sistemos kiekvienas dokumentas turi turėti dokumento pateikimo datą.
- FR 6.7. Sistemos kiekvienas dokumentas turi turėti dokumento patvirtinimo datą.
- FR 6.8. Sistemos kiekvienas dokumentas turi turėti dokumento atmetimo datą.
- FR 6.9. Sistema turi leisti vartotojui matyti dokumento būseną, kai dokumentas sukurtas.
- FR 6.10. Sistema turi leisti vartotojui matyti dokumento būseną, kai dokumentas priimtas.
- FR 6.11. Sistema turi leisti vartotojui matyti dokumento būseną, kai dokumentas pateiktas.
- FR 6.12. Sistema turi rodyti koks vartotojas priėmė dokumentą.
- FR 6.13. Sistema turi rodyti koks vartotojas atmetė dokumentą.
- FR 6.14. Sistema turi rodyti pateikto dokumento atmetimo priežastį.

FR 7. Vartotojai gali peržiūrėti sukurtus dokumentus

- FR 7.1. Sistemos vartotojai gali peržiūrėti savo sukurtus ir dar nepateiktus dokumentus.
- FR 7.2. Sistema turi leisti vartotojui peržiūrėti sukurtus dokumentus naudojant filtravimo funkciją.
- FR 7.3. Sistemos vartotojas turi turėti galimybę savo sukurtus dokumentus filtruoti pagal dokumento tipą.
- FR 7.4. Sistemos vartotojas turi turėti galimybę savo sukurtus dokumentus filtruoti pagal datą

FR 8. Vartotojai gali peržiūrėti savo pateiktus dokumentus

- FR 8.1. Sistemos vartotojas gali peržiūrėti savo pateiktus dokumentus.
- FR 8.2. Sistemos vartotojai gali matyti savo pateiktus dokumentus, kai būseną yra „Pateiktas“.
- FR 8.3. Sistemos vartotojai gali matyti savo pateiktus dokumentus, kai būseną yra „Priimtas“.
- FR 8.4. Sistemos vartotojai gali matyti savo pateiktus dokumentus, kai būseną yra „Atmestas“.
- FR 8.5. Sistemos vartotojas turi turėti galimybę savo pateiktus dokumentus filtruoti pagal dokumento tipą.
- FR 8.6. Sistemos vartotojas turi turėti galimybę savo pateiktus dokumentus filtruoti pagal gavėją.
- FR 8.7. Sistemos vartotojas turi turėti galimybę savo pateiktus dokumentus filtruoti pagal būseną.
- FR 8.8. Sistemos vartotojas turi turėti galimybę savo pateiktus dokumentus filtruoti pagal datą.

FR 9. Priimančios grupės vartotojai gali matyti jiems pateiktus dokumentus

FR 9.1. Sistemos priimančios grupės vartotojai gali matyti visų pateiktų konkretaus tipo dokumentų sąrašą.

FR 9.2. Sistemos priimančios grupės vartotojai turi teisę peržiūrėti vartotojų dokumentus.

FR 9.3. Sistemos priimančios grupės vartotojai turi teisę patvirtinti vartotojų dokumentus.

FR 9.4. Sistemos priimančios grupės vartotojai turi teisę atmesti vartotojų dokumentus.

FR 10. Priimančios grupės vartotojai gali matyti dokumentų pateikimo, priėmimo ir atmetimo statistiką

FR 10.1. Priimančios grupės vartotojai priimančys tam tikrų tipų dokumentus, gali matyti tų dokumentų tipų pateikimo statistiką.

FR 10.2. Sistema turi parodyti kiek *pateikta* tam tikro tipo dokumentų per nurodytą laikotarpį.

FR 10.3. Sistema turi parodyti kiek *priimta* tam tikro tipo dokumentų per nurodytą laikotarpį.

FR 10.4. Sistema turi parodyti kiek *atmesta* tam tikro tipo dokumentų per nurodytą laikotarpį.

FR 10.5. Sistema turi leisti priimančiai grupei matyti lentelėje pateiktą dokumentų statistiką pagal dokumentų būseną „Pateiktas“.

FR 10.6. Sistema turi leisti priimančiai grupei matyti lentelėje pateiktą dokumentų statistiką pagal dokumentų būseną „Priimtas“.

FR 10.7. Sistema turi leisti priimančiai grupei matyti lentelėje pateiktą dokumentų statistiką pagal dokumentų būseną „Atmestas“.

FR 10.8. Sistema turi leisti priimančiai grupei atvaizduoti grafike pateiktą dokumentų statistiką pagal dokumentų būseną „Pateiktas“.

FR 10.9. Sistema turi leisti priimančiai grupei atvaizduoti grafike pateiktą dokumentų statistiką pagal dokumentų būseną „Priimtas“.

FR 10.10. Sistema turi leisti priimančiai grupei atvaizduoti grafike pateiktą dokumentų statistiką pagal dokumentų būseną „Atmestas“.

Nefunkciniai reikalavimai

NFR 1. Sistema įgyvendinta naudojant Java technologijas

NFR 1.1. Sistema turi būti įgyvendinta naudojant Java (8+) programavimo kalbą, Spring (5+) karkasą.

NFR 2. Vartotojo sąsajai įgyvendinti naudojamas React (16+) karkasas

NFR 2.1. Sistemos vartotojo sąsajai įgyvendinti naudojamas React (16+) karkasas

NFR 3. Sistemos duomenys saugomi reliacinėje duomenų bazėje

NFR 3.1. Sistemos duomenys turi būti saugomi reliacinėje duomenų bazėje.

NFR 4. Vartotojų slaptažodžiai apsaugoti maišos funkcija

NFR 4.1. Sistemos vartotojų slaptažodžiai turi būti užkoduoti.

NFR 4.2. Sistemos vartotojų slaptažodžiai privalo būti saugomi naudojant maišos funkciją.

NFR 4.3. Pasirinkta maišos funkcija turi būti ne mažesnio nei SHA-256 saugumo lygio.

NFR 4.4. Sistema privalo naudoti papildomas šias apsaugos technikas, kaip *“salt”* ir *“pepper”*.

NFR 5. Visi veiksmai sistemoje kaupiami įvykių žurnale

NFR 5.1. Visi veiksmai sistemoje turi būti matomi įvykių žurnale.

NFR 5.2. Visi veiksmai sistemoje turi būti registruojami įvykių žurnale.

NFR 5.3. Sistemoje kiekvienas įvykis turi turėti registravimo laiką.

NFR 5.4. Sistemoje kiekvienas įvykis turi turėti vartotojo id.

NFR 5.5. Sistemoje kiekvienas įvykis turi turėti unikalų registravimo numerį.

NFR 5.6. Kiekvienas įvykis turi turėti įvykio tipą.

NFR 5.7. Sistemos žurnale registruojamas įvykių registravimo laikas.

NFR 5.8. Sistemos žurnale registruojamas įvykių id.

NFR 5.9. Sistemos žurnale registruojamas vartotojų id.

NFR5.10. Sistemos žurnale registruojamas įvykių tipas.

NFR 6. Kartu su sistema pateikiama sistemos administratoriaus ir vartotojo instrukcijos

NFR 6.1. Turi būti pateikta sistemos administratoriaus naudojimosi sistema instrukcija.

NFR 6.2. Turi būti pateikta sistemos vartotojo naudojimosi sistema instrukcija.

NFR 6.3. Sistemos naudojimo dokumentacija turi aprašyti pagrindinių sistemos funkcijų naudojimą.

NFR 6.4. Sistemos diegimo dokumentacija aprašo kaip dokumentų valdymo sistema turi būti įdiegta konkrečioje aplinkoje.

NFR 6.5. Sistemos diegimo dokumentacija aprašo kaip dokumentų valdymo sistema turi būti paleista konkrečioje aplinkoje.

PRIEDAS

„Savęs paties įsivertinimo rezultatai“ pagal ISO/IEC 25010:2011 kokybės standartą	Įvertinimas balais
Funkcionalumas	7
Patikimumas	8
Panaudojamumas	9
Veikimo našumas	9
Palaikomumas	8
Perkeliamumas	8
Saugumas	10
Suderinamumas	8

„Naudotojų vartojamumo apklausos statistika“ pagal ISO/IEC 25010:2011 kokybės standartą	Įvertinimas balais
Funkcionalumas	8
Patikimumas	9
Panaudojamumas	8
Veikimo našumas	10
Palaikomumas	7
Perkeliamumas	9
Saugumas	10
Suderinamumas	9

