

**KAUNO TECHNOLOGIJOS UNIVERSITETAS
PROGRAMŲ INŽINERIJOS KATEDRA**

**POŽYMIŲ DIAGRAMŲ TEKSTINIO SPECIFIKAVIMO
KALBŲ ANALIZĖ IR TYRIMAS**

Testavimo Planas

Vadovas:

Robertas Damaševičius

Užsakovas:

Vytautas Štuikys

Robertas Damaševičius

Autorius:

IFM-9/2 gr.stud.

Kęstutis Matonis

KAUNAS, 2010

Turinys

Turinys.....	2
1. Įžanga.....	3
1.1. Tikslai ir objektai	3
1.2. Terminų žodynas	3
1.3. Testavimo Technologijos	3
1.4. Reikalavimai testavimui	3
1.5. Testavimo prioritetai	4
1.6. Rolės ir atsakomybės	4
1.7. Testavimo Aplinka	4
1.7.1. Techninė įranga	4
1.7.2. Kliento programinė įranga, Operacinė sistema	4
1.1.1. Specifiniai reikalavimai:.....	4
1.8. Testavimo Strategijos	5
1.8.1. Vartotojo sąsajos testavimas	5
1.8.2. Sistemos veikimo testavimas	5
1.8.3. Streso Testavimas	6
2. Testų aprašymas	7
2.1. Vartotojo sąsajos testavimas.....	7
2.1.1. Projekto sukūrimas.....	7
2.1.2. Diagramos elemento įtraukimas į diagramą	7
2.1.3. Ryšio tarp diagramos elementų įtraukimas.....	7
2.1.4. Diagramos elemento priskyrimas UML diagramos elementui	8
2.1.5. Diagramos išsaugojimas XML formatu	8
2.1.6. Diagramos išsaugojimas OWL formatu.....	9
2.1.7. Diagramos išsaugojimas FDL formatu.....	9
2.1.8. Diagramos išsaugojimas paveikslėlio formatu.....	10
2.1.9. UML diagramos generavimas	10
2.2. Streso testavimas.....	11
2.3. Testavimo rezultatų kaupimas.....	11

1. Įžanga

Šiame skyriuje yra pateikta apibendrinta požymių diagramų modeliavimo įrankio plėtinio programinės įrangos testų specifikacija.

1.1. Tikslai ir objektai

Pateikti vartotojui teisingai funkcionuojantį produktą.

Šiame dokumente pateikiamai testavimo atvejai, padedantys užtikrinti aukštą įrankio programinės įrangos kokybę ir minimalų defektų kiekį.

1.2. Terminų žodynas

lentelė 1.1. Terminų žodynas

Terminas	Apibūdinimas
Eclipse programavimo aplinka su modeliavimo įrankių komplektu	Programa, priedų platforma, kurią galima papildyti individualiais priedais.
XML	Extensible Markup Language
FDL	Feature Description Language
OWL	Web Ontology Language

1.3. Testavimo Technologijos

Požymių diagramų plėtinio komponentui kokybės užtikrinimui bus atlikti:

- Naudotojo sąsajos testavimas;
- Komponento testavimas;
- Integracijos tarp komponentų testavimas;
- Visos sistemos testavimas;
- Sistemos pertestavimas po klaidų pataisymo.

1.4. Reikalavimai testavimui

Norint atlikti programinės įrangos testavimą reikalinga:

- Pilna programinės įrangos specifikacija išreikšta panaudojimo atvejų diagramomis ir panaudojimo scenarijais.
- Pilna ir testavimui parengta programinė įranga.
- Nustatyta procedūra, ką daryti su aptiktomis klaidomis.
- Klaidos ataskaitos šablonas.

1.5. Testavimo prioritetai

Atliekant sistemos testavimą pagrindinis dėmesys bus kreipiamas į šiuos faktorius. Jie pateikti svarbumo mažėjimo tvarka.

- Funkcionalumas – ar reikalingos programinės įrangos funkcijos yra ir atliekam joms priskirtas roles taip kaip reikia (atitinka specifikaciją)
- Panaudojamumas – kaip vartotoja sąsaja yra patogi vartotojui ir draugiška.
- Saugumas – ar programa ir vartotojas gali sugadinti pradinius duomenis.

1.6. Rolės ir atsakomybės

Atliekant programinės įrangos testavimą yra apibrėžtos tokios rolės ir jų atsakomybės:

- Vadovas – atsakingas už testavimo planus ir sėkmingą jų vykdymą.
- Testuotojas – atsakingas už testų vykdymą pagal sudarytą planą.
- Produkto vadovas – užtikrina, kad testai yra įvykdyti sėkmingai žiūrint iš vartotojo perspektyvos.

1.7. Testavimo Aplinka

1.7.1. Techninė įranga

Testavimas turi būti atliekamas naudojant techninę įrangą, kuri atitiktų reikalavimus:

- Personalinio kompiuterio rekomenduojama minimali specifikacija: 1.5 GHz spartos procesorius, 512MB darbinės atminties,
- Rezoliucija ne mažesnė nei 1024 x 768.

1.7.2. Kliento programinė įranga, Operacinė sistema

Komponentas turi veikti operacinėse sistemose:

- Windows XP
- Windows 2000 profesional/server
- Windows 2003
- Windows Vista
- Linux

1.1.1. Specifiniai reikalavimai:

- Įrankis turi būti įkomponuotas į Eclipse Modeling Tools

1.8. Testavimo Strategijos

Kuriant testavimo atvejus ir testuojant Požymių diagramų braižymo komponentą turi būti naudojamos šios strategijos:

- Vartotojo sąsajos testavimas;
- Streso testavimas.
- Vienetų testavimas
- Integravimo testavimas
- Priemimo testavimas
- Aukšto lygio testavimas

1.8.1. Vartotojo sąsajos testavimas

Vartotojo sąsajos testavimo metu yra tikrinama ar vartotojas gali pasiekti visas reikiamas funkcijas. Tikrinama ar UI yra intuityvi..

Testo tikslas:	Patikrinti ar visi vartotojo interfeiso komponentai yra pasiekiami. Ar galima pereiti nuo lango prie lango, nuo lauko prie lauko naudojantis klaviatūra, pelės pagalba ir sparčiaisiais mygtukais. Visi langų objektai, tokie kaip meniu, objektų dydžiai, vieta lange ir kiti, atitinka numatytus standartus.
Technologija:	Sukurti ir įvykdyti testavimo atvejus kiekvienam sistemos langui ir visiems esantiems objektams jame.
Užbaigimo kriterijai:	Vartotojas gali exportuoti diagramas į visus saugojimo formatus

Kuriant vartotojo sąsajos testavimo atvejį reikia numatyti tokią informaciją:

- *Testavimo atvejo pavadinimas.* Pavadinimą rekomenduojama rašyti tokį patį kaip ir testuojamo lango pavadinimas.
- *Testavimo atvejo scenarijus.* Testavimo scenarijų rekomenduojama rašyti lentelės forma.

Žemiau nurodyta lentelėje turi būti pateikiami testavimo atvejai:

Nr	Objektas	Testavimo sąlygos, atvejis	Laukiami rezultatai

1.8.2. Sistemos veikimo testavimas

Sistemos veikimo testas yra naudojamas, norint nustatyti ir įvertinti funkcijų vykdymo laikus, reakcijos greičius ir kitus skaičiuojamus parametrus. Šio testo tikslas yra patikrinti ar sistema veikia

pagal nustatytus sistemos veikimo reikalavimus. Veikimo testas turi būti parašytas ir įvykdytas pagal nurodytus sistemos veikimo techninius reikalavimus.

Žemiau nurodyta lentelėje turi būti pateikiami testavimo atvejai:

Nr	Atvejis	Matavimo parametras

1.8.3. Streso Testavimas

Streso testavimas yra vienas iš sistemos veikimo testų, kurio tikslas patikrinti sistemos veikimą esant mažiems resursams arba dirbant dideliu kiekiu sistemos naudotojų. Esant atminties ar vietos diske trūkumui gali būti aptinkamos klaidos, kurios nėra matomos normaliomis sąlygomis.

Testo tikslas:	<p>Patikrinti ar sistemos funkcijos teisingai ir be klaidų veikia esant stresinėms situacijoms:</p> <ul style="list-style-type: none"> • Esant mažai ar trūkstant kompiuterio atminties (RAM). • Daug kartų kartojant nekorektiškas funkcijų veikimo situacijas bei jas keičiant. <p>Pastaba: streso testo metu taip pat gali būti aprašomos situacijos, kurioms esant sistema veiks neteisingai.</p>
Technologija:	<ul style="list-style-type: none"> • Panaudoti testavimo atvejus aprašytus veikimo ir apkrovos testuose. • Norint patikrinti sistemos veikimą esant mažiems kompiuterio resursams, testas gali būti vykdomas viename kompiuteryje. • Testuojant keliems sistemos naudotojams, testo duomenys gali būti naudojami tie patys. Šiuo atveju gali būti nustatomos situacijos, kuriose sistema veikia nekorektiškai.
Užbaigimo kriterijai:	Testais yra skaitomas baigtas, kai visi suplanuoti testai yra suvykdyti ir aprašyti visi apribojimai sistemos veikimui (pateiktos situacijos, kurioms esant komponentas neveiks).

Kuriant naudotojo sąsajos testavimo atvejį reikia numatyti tokią informaciją:

- *Testavimo atvejo pavadinimas.*
- *Tikslas.* Trumpai aprašyti testavimo atvejo tikslą.
- *Pradinės sąlygos.* Aprašyti pradinę testavimo situaciją.
- *Testavimo atvejo scenarijus.* Testavimo scenarijų rekomenduojama rašyti lentelės forma.
- *Laukiami rezultatai.* Aprašyti laukiamą sistemos reakciją.

Žemiau nurodyta lentelėje turi būti pateikiami testavimo atvejai:

Nr	Naudotojo veiksmas	Laukiami rezultatai

2. Testų aprašymas

2.1. Vartotojo sąsajos testavimas

2.1.1. Projekto sukūrimas

Tikslas: Sukurti naują požymių diagramą.

Pradinės sąlygos: Sistema prieinama vartotojui; Vartotojo techninė bei programinė įranga atitinka reikalavimus

Susiję testavimo atvejai: Visi

Veikėjai: Visi

Scenarijus:

Nr	Naudotojo veiksmas	Laukiami rezultatai
1	Pasileisti Eclipse programą	Atidaroma programa ir užkraunamas požymių diagramos braižymo komponentas.
2	Atsidariusioje programoje susikurti standartinį projekto katalogą File->New Project -> General -> Project	Programos Project Explorer lange sukuriamas naujas projekto katalogas
3	Spaudžiamas dešinysis pelės klavišas ant sukurto projekto katalogo ir jam priskiriama požymių diagrama New -> Exsample... -> Fd2 diagram	Sukuriama požymių braižymo diagrama ir atidaros braižymo langas

2.1.2. Diagramos elemento įtraukimas į diagramą

Tikslas: Įtraukti į diagramą naują požymių diagramos elementą.

Pradinės sąlygos: Sukurtas naujas projektas.

Susiję testavimo atvejai: Visi

Veikėjai: Visi

Scenarijus:

Nr	Naudotojo veiksmas	Laukiami rezultatai
1	Išsirenkamas norimas diagramos elementas šoninėje įrankių juostoje ir pažymimas pele	Norimas įtraukti elementas nuspalvinamas oranžine spalva, taip parodant kad jis yra aktyvus
2	Diagramos brėžimo lange, norimoje vietoje spustelima pelės kairiuoju klavišu	Naujas diagramos elementas atvaizduojamas diagramos brėžimo lange
3	Įvedamas elemento pavadinimas	Požymių diagramos elementas atvaizduojamas su nauju pavadinimu.

2.1.3. Ryšio tarp diagramos elementų įtraukimas

Tikslas: Sujungti du diagramos elementu ryšio elementu.

Pradinės sąlygos: Įterpti mažiausiai du diagramos elementai.

Susiję testavimo atvejai: Visi

Veikėjai: Visi

Scenarijus:

Nr	Naudotojo veiksmas	Laukiami rezultatai
1	Išsirenkamas norimas ryšio elementas šoninėje įrankių juostoje ir pažymimas pele	Norimas įtraukti ryšio elementas nuspalvinamas oranžine spalva, taip parodant kad jis yra aktyvus
2	Diagramos brėžimo lange pasirenkamas pirmas diagramos elementas ir paspaudžiamas, po to išsirenkamas antras elementas ir pažymimas	Tarp dviejų diagramos elementų nubrėžiamas ryšys

2.1.4. Diagramos elemento priskyrimas UML diagramos elementui

Tikslas: Priskirti UML diagramos elementą.

Pradinės sąlygos: Įterpti mažiausiai du diagramos elementai.

Susiję testavimo atvejai: Visi

Veikėjai: Visi

Scenarijus:

Nr	Naudotojo veiksmas	Laukiami rezultatai
1	Išsirenkamas diagramos elementas, spaudžiamas dešinysis pelės klavišas ir pasirenkama priskirti UML diagramos elementą.	Atidaromas UML diagramos pasirinkimo langas
2	Išsirenkama UML diagrama ir nurodomas elementas kuris bus priskirtas ir spaudžiama OK	Prie diagramos elemento priskiriamas UML diagramos elementas ir atvaizduojamas paveikslukas šalia elemento.

2.1.5. Diagramos išsaugojimas XML formatu

Tikslas: Išsaugoti požymių diagramą XML formatu.

Pradinės sąlygos: Nubrėžta požymių diagrama.

Susiję testavimo atvejai: Visi

Veikėjai: Visi

Scenarijus:

Nr	Naudotojo veiksmas	Laukiami rezultatai
1	Pasirenkama diagrama Project Explorer lange ir spaudžiamas dešinysis pelės klavišas ir pasirenkama Export	Atidaromas diagramos eksportavimo langas
2	Atsidariusiame lange pasirenkamas norimas eksportavimo formatas (XML) ir	Pasirenkamas XML saugojimo formatas

	spaudžiama Next	
3	Pasirenkama norima eksportuoti diagrama ir spaudžiama Next	Pasirenkama eksportuojama diagrama
4	Atsidariusiame dokumento pavadinimo lange įvedamas norimas dokumento pavadinimas	Sukuriamas norimu pavadinimu XML dokumentas

2.1.6. Diagramos išsaugojimas OWL formatu

Tikslas: Išsaugoti požymių diagramą OWL formatu.

Pradinės sąlygos: Nubrėžta požymių diagrama.

Susiję testavimo atvejai: Visi

Veikėjai: Visi

Scenarijus:

Nr	Naudotojo veiksmas	Laukiami rezultatai
1	Pasirenkama diagrama Project Explorer lange ir spaudžiamas dešinysis pelės klavišas ir pasirenkama Export	Atidaromas diagramos eksportavimo langas
2	Atsidariusiame lange pasirenkamas norimas eksportavimo formatas (OWL) ir spaudžiama Next	Pasirenkamas OWL saugojimo formatas
3	Pasirenkama norima eksportuoti diagrama ir spaudžiama Next	Pasirenkama eksportuojama diagrama
4	Atsidariusiame dokumento pavadinimo lange įvedamas norimas dokumento pavadinimas	Sukuriamas norimu pavadinimu OWL dokumentas

2.1.7. Diagramos išsaugojimas FDL formatu

Tikslas: Išsaugoti požymių diagramą FDL formatu.

Pradinės sąlygos: Nubrėžta požymių diagrama.

Susiję testavimo atvejai: Visi

Veikėjai: Visi

Scenarijus:

Nr	Naudotojo veiksmas	Laukiami rezultatai
1	Pasirenkama diagrama Project Explorer lange ir spaudžiamas dešinysis pelės klavišas ir pasirenkama Export	Atidaromas diagramos eksportavimo langas
2	Atsidariusiame lange pasirenkamas norimas eksportavimo formatas (FDL) ir spaudžiama Next	Pasirenkamas FDL saugojimo formatas

3	Pasirenkama norima eksportuoti diagrama ir spaudžiama Next	Pasirenkama eksportuojama diagrama
4	Atsidariusiame dokumento pavadinimo lange įvedamas norimas dokumento pavadinimas	Sukuriamas norimu pavadinimu FDL dokumentas

2.1.8. Diagramos išsaugojimas paveikslėlio formatu

Tikslas: Išsaugoti požymių diagramą paveikslėlio formatu.

Pradinės sąlygos: Nubrėžta požymių diagrama.

Susiję testavimo atvejai: Visi

Veikėjai: Visi

Scenarijus A:

Nr	Naudotojo veiksmas	Laukiami rezultatai
1	Diagramos lange spaudžiamas dešinysis pelės klavišas ir pasilenkama File -> Save As Image File..	Atidaromas paveikslėlio saugojimo langas
2	Pasirenkama saugojimo vieta, formatas.	Požymių diagrama išsaugojama pasirinktoje vietoje, pasirinktu formatu.

Scenarijus B:

Nr	Naudotojo veiksmas	Laukiami rezultatai
1	Diagramos lange spaudžiama dešiniuoju pelės klavišu diagramos kairiame viršutiniame kampe ir nepaleidžiant pelės klavišo vedama iki diagramos dešiniojo apatinio kampo ir klavišas atleidžiamas.	Pažymimi diagramos elementai kurie telpa į apvestą langą.
2	Spaudžiamas dešinysis pelės klavišas ir pasirenkama Copy (alternatyva Ctrl+C)	Diagrama nukopijuojama į atmintį ir naudojant įklijavimo funkciją programose kurios priima vaizdinės informacijos įklijavimą, galima įklijuoti pasirinkus reikiamą komandą (arba Ctrl+V)

2.1.9. UML diagramos generavimas

Tikslas: Sugeneruoti UML diagramą iš požymių diagramos.

Pradinės sąlygos: Nubrėžta požymių diagrama ir priskirti UML diagramos elementai.

Susiję testavimo atvejai: Visi

Veikėjai: Visi

Scenarijus:

Nr	Naudotojo veiksmas	Laukiami rezultatai
1	Pasirenkama diagrama Project Explorer lange ir spaudžiamas dešinysis pelės klavišas ir pasirenkama Export	Atidaromas diagramos eksportavimo langas

2	Atsidariusiame lange pasirenkamas norimas eksportavimo formatas (UML) ir spaudžiama Next	Pasirenkamas UML saugojimo formatas
3	Pasirenkama norima eksportuoti diagrama ir spaudžiama Next	Pasirenkama eksportuojama diagrama
4	Atsidariusiame dokumento pavadinimo lange įvedamas norimas dokumento pavadinimas	Sugeneruojama UML diagrama ir sukuriamas diagramos dokumentas.

2.2. Streso testavimas

Tikslas: Patikrinti sistemos veikimą sumažinus atminties kiekį (vykdomas testas LTC# 1)

Pradinės sąlygos: Sistema prieinama vartotojui; Vartotojo techninė bei programinė įranga atitinka reikalavimus; Vartotojas pasileidęs Eclipse sistemą su požymių braižymo komponentu.

Scenarijus: Vykdomas didelės požymių diagramos generavimas su minimaliais techniniais reikalavimais.

Laukiami rezultatai: Sugeneruota UML diagrama

2.3. Testavimo rezultatų kaupimas

Testavimo pabaigoje tikimasi gauti šiuos dokumentus:

- Testavimo planas – šis dokumentas, taip pat jo pakeitimai, jei tokie bus.
- Pakeitimų reikalavimai – klaidų ataskaitos, reikalingų pakeitimų programinei įrangai dokumentai.
- Užbaigimo ataskaita – dokumentas rodantis, kad buvo atliktas sėkmingai priėmimo testavimas

Testų rezultatai bus fiksuojami žemiau nurodytos lentelės formatu.

Klaidos numeris	Klaidos pavadinimas ir aprašymas	Vieta programoje	Būsena
1			
2			
...			