



KAUNO TECHNOLOGIJOS UNIVERSITETAS
INFORMATIKOS FAKULTETAS

SONATA JACKYTĖ

ONTOLOGIJŲ VERIFIKAVIMO GALIMYBIŲ TYRIMAS

Baigiamasis magistro projektas

Vadovė
doc. dr. R. Butkienė

KAUNAS, 2017

KAUNO TECHNOLOGIJOS UNIVERSITETAS
INFORMATIKOS FAKULTETAS

ONTOLOGIJŲ VERIFIKAVIMO GALIMYBIŲ TYRIMAS

Baigiamasis magistro projektas
Informacinių sistemų inžinerijos studijų programa (kodas 621E15001)

Vadovė

doc. dr. R. Butkienė
2017-05-22

Recenzentas

dr. Mantas Lukoševičius
2017-05-22

Projektą atliko

Sonata Jackytė
2017-05-22

KAUNAS, 2017



KAUNO TECHNOLOGIJOS UNIVERSITETAS
INFORMATIKOS FAKULTETAS

(Fakultetas)

Sonata Jackytė

(Studento vardas, pavardė)

Informacinių sistemų inžinerijos studijų programa, 621E15001

(Studijų programos pavadinimas, kodas)

Baigiamojo projekto „Ontologijų verifikavimo galimybių tyrimas“

AKADEMINIO SAŽININGUMO DEKLARACIJA

20 ____ m. ____ d.

Kaunas

Patvirtinu, kad mano, **Sonatos Jackytės**, baigiamasis projektas tema „Ontologijų verifikavimo galimybių tyrimas“ yra parašytas visiškai savarankiškai ir visi pateikti duomenys ar tyrimų rezultatai yra teisingi ir gauti sąžiningai. Šiame darbe nei viena dalis nėra plagijuota nuo jokių spausdintinių ar internetinių šaltinių, visos kitų šaltinių tiesioginės ir netiesioginės citatos nurodytos literatūros nuorodose. Įstatymų nenumatytų piniginių sumų už šį darbą niekam nesu mokėjęs.

Aš suprantu, kad išaiškėjus nesąžiningumo faktui, man bus taikomos nuobaudos, remiantis Kauno technologijos universitete galiojančia tvarka.

(vardą ir pavardę įrašyti ranka)

(parašas)

Jackytė, Sonata. Ontologijų verifikavimo galimybių tyrimas. Magistro baigiamasis projektas / vadovas doc. dr. Rita Butkienė; Kauno technologijos universitetas, Informatikos fakultetas.
Mokslo kryptis ir sritis: Informatikos inžinerija, technologijos mokslai
Reikšminiai žodžiai: *semantinis tinklas, ontologija, OWL 2, RDF, verifikavimas*.
Kaunas, 2017. 71 p.

SANTRAUKA

Šiame tiriamajame darbe analizuojamas bazinis semantinio tinklo komponentas – ontologijos ir jų aprašymas. Pristatomos bendrosios ontologijos elementų aprašo tikrinimo taisyklės.

Tyrimas remiasi atlikta esamų ontologijų verifikavimo sistemų ir gerųjų praktikų analize. Analizės metu išskirti pagrindiniai ontologiją sudarantys elementai ir jų aprašymas. Remiantis bendrosiomis ontologijos aprašo taisyklėmis sukurtas ontologijų verifikavimo įrankio prototipas, kurio paskirtis identifikuoti ontologijos elementus prieštaraujančius bendrosioms ontologijos elementų aprašymo taisyklėms. Verifikavimo įrankio pagalba ontologijų modeliavimo ekspertai gali pastebėti ontologijos kūrimo metu padarytus pažeidimus. Eksperimento metu patikrinta dešimt ontologijų ir ištirtos ontologijų verifikavimo įrankio galimybės.

Tiriamąjį darbą metu gauti rezultatai parodė, kad įrankis gali aptikti ontologijoje esančius pažeidimus ir suformuoti rezultatų failą, kuris gali būti perkeltas į ontologijų redaktorių tolimesniam ontologijos tobulinimui.

Jackytė, Sonata. *Study of Feasibility to Verify Ontology*: Master's thesis in Information Systems Engineering / supervisor assoc. prof. Rita Butkienė. The Faculty of Informatics, Kaunas University of Technology.

Research area and field: Informatics Engineering, Technology Science
Key words: *Semantic Web, ontology, OWL 2, RDF, verification*.
Kaunas, 2017. 71 p.

SUMMARY

The purpose of this research paper is to analyze the base component of semantic network - ontology and its description, as well as present ontology description verification rules based on the ontology creation process.

This research is based on the analysis of existing systems and best known practices. The analytical part of this research distinguishes the basic ontology forming elements and their corresponding descriptions. Using the general description of ontology rules this paper introduces a generated ontology verification tool prototype whose aim is to identify the ontology elements that do not correspond to the before mentioned rules. The purpose of this verification tool is to help ontology modeling experts identify infringements that arise when ontologies are created. Designed experiment tested ten ontologies to investigate options of the proposed verification tool.

Data collected at the end of the research show that verification tool can detect flaws in ontologies and generate a results file that then can be used by the ontology editor for further improvements.

TURINYS

| | |
|--|----|
| LENTELIŲ SĄRAŠAS | 7 |
| PAVEIKSLŲ SĄRAŠAS | 8 |
| TERMINŲ IR SANTRUPŲ ŽODYNAS | 9 |
| ĮVADAS | 10 |
| 1. ONTOLOGIJŲ KŪRIMO ANALIZĖ | 12 |
| 1.1. Analizės tikslas | 12 |
| 1.2. Tyrimo objektas, sritis ir problema | 12 |
| 1.3. Ontologijos apibrėžimas | 12 |
| 1.4. Semantinio tinklo apibrėžimas ir jam išreikšti naudojamų kalbų analizė..... | 13 |
| 1.5. Esamų sprendimų analizė | 18 |
| 1.6. Siekiamo sprendimo apibrėžimas | 20 |
| 1.7. Analizės išvados..... | 21 |
| 2. ONTOLOGIJŲ VERIFIKAVIMO ĮRANKIO REIKALAVIMŲ SPECIFIKACIJA IR PROJEKTAVIMAS..... | 23 |
| 2.1. Reikalavimų specifikacija ontologijų verifikavimo įrankiui | 23 |
| 2.2. Ontologijų verifikavimo dalykinės srities modelis | 25 |
| 2.3. Formalus sprendimo aprašas..... | 26 |
| 2.4. Reikalavimų apibendrinimas | 32 |
| 3. ONTOLOGIJŲ VERIFIKAVIMO ĮRANKIO REALIZACIJOS PROJEKTAS | 33 |
| 3.1. Sprendimo pagrindimas ir esminiai principai | 33 |
| 3.2. Ontologijų verifikavimo įrankio architektūra | 33 |
| 3.3. Sistemos elgsenos modelis..... | 34 |
| 3.4. Detalus projektavimas..... | 38 |
| 3.5. Programinių komponentų architektūra | 39 |
| 4. ONTOLOGIJOS VERIFIKAVIMO ĮRANKIO REALIZACIJA IR TESTAVIMAS | 41 |
| 4.1. Sprendimo realizacijos ir veikimo aprašas | 41 |
| 4.2. Testavimo modelis | 44 |
| 4.3. Testavimo duomenys ir rezultatai | 44 |
| 4.4. Testavimo rezultatų apibendrinimas | 46 |
| 5. EKSPERIMENTINIS ONTOLOGIJŲ VERIFIKAVIMO SISTEMOS TYRIMAS..... | 48 |
| 5.1. Eksperimento planas | 48 |
| 5.2. Eksperimento rezultatai | 48 |
| 6. REZULTATŲ APIBENDRINIMAS IR IŠVADOS..... | 68 |
| 7. LITERATŪRA..... | 69 |
| 8. PRIEDAI..... | 72 |
| 8.1. priedas. Įspėjimų ir klaidų anotacijų ontologijos failas | 72 |
| 8.2. priedas. Politikos ontologijos verifikavimo metu sugeneruotas rezultatų failas..... | 75 |

8.3. priedas. Tikrintų ontologijų detali statistika78

LENTELIŲ SĄRAŠAS

| | |
|---|----|
| 1.1 lentelė. Esamų sprendimų palyginimas..... | 20 |
| 2.1 lentelė. Įspėjimų ir klaidų anotacijų ontologijos struktūra..... | 31 |
| 3.1 lentelė. Ontologijos verifikavimo valdiklio operacijos..... | 39 |
| 3.2 lentelė. Ontologijos įkėlimo valdiklio operacijos | 39 |
| 4.1 lentelė. Ontologijų verifikavimo įrankio pasirinkimai..... | 42 |
| 4.2 lentelė. Politikos ontologijos charakteristikos | 45 |
| 4.3 lentelė. Ontologijų verifikavimo įrankio testavimo metu gauti rezultatai | 45 |
| 5.1 lentelė. Eksperimento metu naudotos ontologijos | 48 |
| 5.2 lentelė. Laiko ontologijos charakteristikos | 49 |
| 5.3 lentelė. Laiko ontologijos verifikavimo rezultatai | 49 |
| 5.4 lentelė. Bendravimo ontologijos charakteristikos..... | 50 |
| 5.5 lentelė. Bendravimo ontologijos verifikavimo rezultatai..... | 51 |
| 5.6 lentelė. Vynų ontologijos charakteristikos..... | 52 |
| 5.7 lentelė. Vynų ontologijos verifikavimo rezultatai..... | 52 |
| 5.8 lentelė. Viešojo administravimo ontologijos charakteristika..... | 53 |
| 5.9 lentelė. Viešojo administravimo ontologijos verifikavimo rezultatai..... | 54 |
| 5.10 lentelė. Asmenvardžių ir organizacijų pavadinimų ontologijos verifikavimo rezultatai | 55 |
| 5.11 lentelė. Asmenvardžių ir organizacijų pavadinimų ontologijos verifikavimo rezultatai | 55 |
| 5.12 lentelė. Įvykių ontologijos charakteristikos | 57 |
| 5.13 lentelė. Įvykių ontologijos verifikavimo rezultatai | 58 |
| 5.14 lentelė. Vietovardžių ontologijos charakteristikos..... | 59 |
| 5.15 lentelė. Vietovardžių ontologijos verifikavimo rezultatai..... | 60 |
| 5.16 lentelė. Agentų ontologijos charakteristikos | 61 |
| 5.17 lentelė. Agentų ontologijos verifikavimo rezultatai..... | 61 |
| 5.18 lentelė. Abstrakčių objektų ontologijos charakteristikos | 62 |
| 5.19 lentelė. Abstrakčių objektų ontologijos verifikavimo rezultatai | 63 |
| 5.20 lentelė. SemLt ontologijos charakteristikos | 64 |
| 5.21 lentelė. SemLt ontologijos verifikavimo rezultatai..... | 64 |
| 5.22 lentelė. Eksperimento rezultatai..... | 66 |

PAVEIKSLŲ SĄRAŠAS

| | |
|---|----|
| 1.1 pav. <i>RDF</i> grafo struktūra | 13 |
| 1.2 pav. <i>RDF</i> sintaksės pavyzdys..... | 14 |
| 1.3 pav. <i>OWL 2</i> meta-modelio struktūra [15] | 14 |
| 1.4 pav. <i>OWL 2</i> ontologijos struktūra [17]..... | 15 |
| 1.5 pav. <i>OWL 2</i> esybės ir jų deklavimas [17] | 15 |
| 1.6 pav. Objektinių savybių sudarymo struktūros pavyzdys | 16 |
| 1.7 pav. Duomenų savybių sudarymo struktūros pavyzdys..... | 16 |
| 1.8 pav. Anotacijos <i>label_sbvr</i> sudarymo struktūros pavyzdys..... | 17 |
| 1.9 pav. Ontologijos kūrimo žingsniai | 17 |
| 1.10 pav. <i>OWL 2</i> į <i>SBRV</i> transformavimo procesas įtraukiant ontologijos verifikavimą..... | 21 |
| 2.1 pav. Panaudojimo atvejų diagrama | 23 |
| 2.2 pav. Panaudojimo atvejo <i>įkelti ontologiją</i> veiklos diagrama | 24 |
| 2.3 pav. Panaudojimo atvejo <i>tikrinti peržiūrint</i> veiklos diagrama..... | 24 |
| 2.4 pav. Panaudojimo atvejo <i>tikrinti išsaugant</i> veiklos diagrama | 25 |
| 2.5 pav. Ontologijų verifikavimo įrankio klasių modelis | 25 |
| 3.1 pav. Robastiškumo diagrama | 33 |
| 3.2 pav. Loginė sistemos architektūra..... | 33 |
| 3.3 pav. Vartotojo posistemė..... | 34 |
| 3.4 pav. Ontologijos verifikavimo posistemė | 34 |
| 3.5 pav. Panaudojimo atvejo <i>įkelti ontologiją</i> sekų diagrama | 35 |
| 3.6 pav. <i>Kalbos pasirinkimo</i> sekų diagrama | 35 |
| 3.7 pav. Panaudojimo atvejo <i>tikrinti peržiūrint</i> sekų diagrama | 36 |
| 3.8 pav. Panaudojimo atvejo <i>tikrinti išsaugant</i> sekų diagrama..... | 36 |
| 3.9 pav. Ontologijos verifikavimo proceso diagrama | 37 |
| 3.10 pav. Detalaus ontologijos verifikavimo proceso diagrama..... | 37 |
| 3.11 pav. Panaudojimo atvejo <i>įkelti ontologiją</i> realizacijos klasių diagrama..... | 38 |
| 3.12 pav. Panaudojimo atvejo <i>tikrinti peržiūrint</i> realizacijos klasių diagrama..... | 38 |
| 3.13 pav. Panaudojimo atvejo <i>tikrinti išsaugant</i> realizacijos klasių diagrama | 38 |
| 3.14 pav. Ontologijų verifikavimo sistemos komponentų diagrama | 39 |
| 3.15 pav. Komponento <i>pagrindinis</i> diagrama..... | 39 |
| 3.16 pav. Komponento <i>įkėlimas</i> diagrama..... | 40 |
| 3.17 pav. Komponento <i>verifikatorius</i> diagrama..... | 40 |
| 3.18 pav. Komponento <i>išsaugojimas</i> diagrama | 40 |
| 3.19 pav. Ontologijų verifikavimo sistemos diegimo modelis | 40 |
| 4.1 pav. Pradinis ontologijų verifikatoriaus langas..... | 41 |
| 4.2 pav. Ontologijos įkėlimo inicijavimas | 41 |
| 4.3 pav. Kalbos pasirinkimas | 42 |
| 4.4 pav. Morfologinės analizės rezultatų atvaizdavimas | 42 |
| 4.5 pav. Rasti pažeidimai prieštaraujantys ontologijos tikrinimo taisyklėms..... | 43 |
| 4.6 pav. Tikrinti ontologiją ją išsaugant..... | 44 |
| 4.7 pav. Verifikuotos <i>politikos</i> ontologijos rezultatų failo fragmentas..... | 44 |
| 4.8 pav. Verifikavimo metu sugeneruoto rezultatų failo atvaizdavimas ontologijų redaktoriuje..... | 47 |

TERMINŲ IR SANTRUPŲ ŽODYNAS

| | |
|--|--|
| Duomenys (angl. <i>Data</i>) | Faktai ar teiginiai, kurie yra ar gali būti surinkti, išsaugoti, apdoroti, tačiau nėra organizuoti ar pateikti kontekste. |
| Informacija (angl. <i>Information</i>) | Duomenų prasmė, reikšmė, numatyta interpretacija. Duomenų organizavimas, susiejimas asociacijomis, apribojimas, sudarantis galimybę juos panaudoti. |
| Informacinė sistema (angl. <i>Information System</i>) | Automatizuota ar rankinė sistema, apimanti informacijos surinkimo, apdorojimo, saugojimo, persiuntimo, pateikimo, platinimo, panaudojimo priemones ir komponentus. |
| Ontologija (angl. <i>Ontology</i>) | Tam tikros srities bendrai naudojamos sąvokų/konceptų, esybių tipų, jų tarpusavio priklausomybių, sąryšių, aksiomų, dėsningumo ir kt. visumos formalus aprašas. |
| Ontologijos validavimas (angl. <i>Ontology validation</i>) | Procesas, kurio metu tikrinamas ontologijos teisingumas dalykinės srities atžvilgiu. |
| Ontologijos verifikavimas (angl. <i>Ontology verification</i>) | Procesas, kurio metu tikrinamas ontologijos sintaksės korektiškumas. |
| OWL (angl. <i>Web Ontology Language</i>) | Ontologijai aprašyti skirta kalba su formaliai apibrėžta semantika. |
| RDF (angl. <i>Resource Description Framework</i>) | Išteklių aprašymo kalba. |
| SBVR (angl. <i>Semantics of Business Vocabulary and Business Rules</i>) | Standartas apibrėžiantis žodyną bei taisykles veiklos žodynų, faktų ir taisyklių aprašymui. SBVR skirtas deklaratyviai esybių aprašymui formalia ir natūralia kalba. |
| XML (angl. <i>Extensible Markup Language</i>) | Išplečiamoji žymėjimo kalba, naudojama duomenims aprašyti. |
| XML Schema (angl. <i>Extensible Markup Language</i>) | Apibūdina XML dokumento struktūrą. |

IVADAS

XXI a. žmogui tinkamai apdorota informacija yra didis ginklas, tačiau jos pateikimas ir supratimas aktualus ne tik žmogui, bet ir kompiuteriams. Semantinis pasaulinis tinklas – tai naujos internetinės eros laikotarpis, kai informacija bus suprantama tiek žmonėms, tiek programinei įrangai. Visa to sudarymo pagrindas – informacija ir jos pateikimas ne tik sintaksiniame, bet ir semantiniame lygmenyje.

Ontologijos apibrėžiamos kaip pagrindinis semantinio tinklo elementas, kuris nusako dalykinės srities savybes ir tarpusavio ryšius. Ontologijos aprašomos formalia ontologijų kalba *OWL* (angl. *Web Ontology Language*), o *OWL 2* yra vėlesnė ir patobulinta versija, ontologijų kūrimo kalba perteikia mašinoms bazines žinias apie dalykinę sritį. Šias žinias gali apdoroti kompiuterinės programos.

OWL 2 ontologijos elementai, tokie kaip klasės, objektinės ir duomenų savybės yra saugomos kaip semantinio saityno reikšmės. *OWL 2* poaibis yra vadinamas *OWL 2* deskriptyviaja logika (DL) (angl. *Description Logic*), kuri remiasi pirmosios eilės logika. Dėl to sukurtos ontologijos loginis teisingumas gali būti patikrintas *OWL 2* tikrinimo įrankiais (angl. *Reasoners*). Siekiant užtikrinti ne tik loginį, bet ir struktūrinį ontologijos teisingumą reikalingas įrankis galintis verifikuoti ontologiją t. y. patikrinti ontologijos struktūrą. Tokio įrankio sukūrimas gali palengvinti ontologijų kūrėjo darbą ir pagerinti ontologijos kokybę. Įrankio pagrindinė idėja remtusi rekomenduojamomis *OWL 2* ontologijos tikrinimo taisyklėmis, be to įrankis galėtų išsaugoti ontologijos tikrinimo metu rastus pažeidimus ir anotacijų pagalba atvaizduotų juos kaip išpėjimus kitose ontologijų redagavimo ir kūrimo sistemose. Ontologijų kūrimo specialistas matydamas išpėjimus galėtų juos ištaisyti. Šiam tikslui įgyvendinti siekiama apibrėžti bendrąsias ontologijų tikrinimo taisykles, kuriomis remiantis ontologijų verifikavimo įrankis galėtų patikrinti ontologiją.

Darbo problematika ir aktualumas

Šiame darbe sprendžiama aktuali problema – ontologijų struktūros tikrinimas/verifikavimas siekiant išvengti žmogiškosios klaidos. Ontologijos sukūrimo procesas reikalauja eksperto atidumo ir kruopštumo, tačiau visada išlieka žmogiškosios klaidos tikimybė. Struktūriškai neteisinga ontologija negali būti panaudojama tolimesniuose kūrimo etapuose arba ontologijos transformacijose. Vienas iš tokių atvejų ontologijos transformavimas į *SBVR* (angl. *Semantics Of Business Vocabulary And Rules*) modelį. Transformacijos metu svarbu užtikrinti ontologijos elementų korektiškumą, nes kitokiu atveju transformacija gali būti nevykdoma arba vykdoma neteisingai. Šiai problemai spręsti planuojama sukurti automatizuoto *OWL 2/RDF* ontologijų verifikavimo įrankį, kuris galės patikrinti ontologijos struktūrą prieš ją transformuojant arba panaudojant kituose kūrimo procesuose. Verifikavimo įrankis remsis ontologijų tikrinimo taisyklėmis, kurios bus struktūrizuotos ir aprašytos šiame darbe.

Darbo tikslas ir uždaviniai

Šio darbo tikslas yra sudaryti galimybes verifikuoti *OWL 2/RDF* formatu aprašytas ontologijas sukuriant kompiuterizuotą ontologijų verifikavimo sistemą. Tiriamojo darbo uždaviniai:

1. išanalizuoti ontologijų *RDF* ir *OWL 2* kalbų modelius, ontologijų kūrimo įrankius ir egzistuojančius sprendimus;
2. sudaryti ontologijų verifikavimo taisykles, suprojektuoti realizuojantį įrankį;
3. realizuoti prototipą, leidžiantį atlikti ontologijų verifikavimą, patikrinti įrankio prototipo vykdomo verifikavimo korektiškumą;
4. atlikti eksperimentą prototipo tinkamumui įvertinti;
5. apibendrinti tyrimo rezultatus.

Darbo rezultatai ir jų svarba

Ontologijų verifikavimo įrankis kuriamas taip, kad potencialus vartotojas galėtų jį įsidiegti į savo asmeninį kompiuterį. Pagrindinis įrankio tikslas pritaikant pagrindines ontologijos tikrinimo taisykles verifikuoti vartotojo įkeltą ontologiją. Verifikavimo įrankio langas galėtų būti padalintas į du laukus, viename iš jų atvaizduojant įkeltą ontologiją, kitame – nustatytų išpėjimų sąrašą. Ontologijos tikrinimo metu vartotojas būtų informuojamas apie rastus taisyklių pažeidimus juos išryškinant skirtinga spalva. Siekiant užtikrinti įrankio patogumą ontologijoje įrankio pagalba rastas struktūras, kurios neatitinka pagrindinių

ontologijos tikrinimo taisyklių būtų galima peržiūrėti iš karto tikrinant ontologiją arba ją išsaugoti pasirenkant verifikavimo taisyklės, kurių pažeidimai yra aktualiausi vartotojui. Ontologijos verifikavimo rezultatų ontologijos failas būtų sudaromas importuojant pagrindinę ontologiją ir įspėjimų ir klaidų anotacijos failą.

Ontologijos verifikavimo įrankio tinkamumui įvertinti buvo atliktas eksperimentas, kurio metu buvo verifikuota 10 skirtingų ontologijų.

Darbo struktūra

- Pirmajame tiriamojo darbo skyriuje apibrėžta tyrimo sritis, objektas ir problema, atlikta *RDF* ir *OWL 2* kalbų analizė, aptarti pagrindiniai komponentai ir jų savybės, pateikti pavyzdžiai. Susipažinta su ontologijos kūrimo principais. Remiantis ontologijos į *SBVR* transformacijos analize ir gerosiomis praktikomis išskirti svarbiausi elementai sudarantys ontologiją, suformuotos siekiamos pagrindinės ontologijos tikrinimo taisyklės. Atlikta esamų įrankių analizė ir palyginimas. Suformuota būsimos sistemos vizija ir veikimo principai.
- Antrajame skyriuje pateikti ontologijų verifikavimo įrankio funkciniai ir nefunkciniai reikalavimai, panaudojimo atvejų diagrama, klasių modelis, apibrėžtos ir predikatų logika aprašytos pagrindinės ontologijos tikrinimo taisyklės, suformuotos ir aprašytos įspėjimų ir klaidų anotacijos.
- Trečiajame skyriuje atlikta reikalavimų analizė, sudaryta loginė sistemos architektūra. Sekų diagramomis atvaizduotas sistemos elgsenos modelis, veiklos diagrama parodytas verifikatoriaus tikrinimo algoritmo veikimas. Panaudojimo atvejai realizuoti klasių diagramomis, aprašyti metodai, kurie naudojami pagrindiniuose įrankio valdikliuose. Sudaryta programinių komponentų architektūra ir diegimo modelis.
- Ketvirtajame skyriuje apibendrinti gauti rezultatai, pateiktas sistemos veikimo aprašymas, aprašyti galimi ontologijos verifikavimo pasirinkimai. Pateikti testavimo metu gauti rezultatai.
- Penktajame skyriuje suformuotas eksperimento apbrėžimas, tikslas, aprašyti eksperimento metu gauti rezultatai ir išvados.

1. ONTOLOGIJŲ KŪRIMO ANALIZĖ

1.1. Analizės tikslas

Šios analizės tikslas yra išsiaiškinti ontologijoms naudojamų kalbų sąvokas bei sintaksę. Perprasti ontologijos kūrimo principus ir įrankius, atlikti analizatorių ir esamų sprendimų palyginimus, suformuoti bendrąsias ontologijų tikrinimo taisykles, suformuoti kuriamo įrankio viziją ir aprašyti analizės išvadas.

Siekiant atlikti išsamų tyrimą analizuojami pasirinktos dalykinės srities literatūros šaltiniai. Problemos sprendimui įgyvendinti analizuojamos ne tik ontologijos, *OWL 2* kalba, bet ir esami įrankiai, kuriais tikrinamos ontologijos.

1.2. Tyrimo objektas, sritis ir problema

Pagrindinis tyrimo objektas yra ontologijos tikrinimo taisyklių išgryninimas ir įrankio joms patikrinti sukūrimas.

Magistrinio darbo tyrimo sritis yra semantinio tinklo kalbos: *RDF* ir *OWL*.

RDF (angl. *Resource Description Framework*) – tai standartinis duomenų modelis skirtas duomenų mainams internetiniame tinkle atlikti, jis grindžiamas tripletais. Kiekvieną tripletą sudaro subjektas, predikatas ir objektas, kurių aibė per bendrai naudojamus subjektus formuoja orientuotą grafą [1].

OWL (angl. *Web Ontology Language*) – tai semantinio tinklo kalba sukurta pateikti žinias apie daiktus, jų grupes ir ryšius tarp jų. Dabartinė *OWL* versija yra vadinama *OWL 2*. *OWL 2* kalba aprašytos ontologijos yra paremtos *RDF* grafais. *OWL 2* ontologijos išreiškia klases, savybes, individus ir duomenų reikšmes. Ontologijos yra saugomos kaip semantinio tinklo dokumentas *RDF/XML* formatu [2].

Tyrimo sritis – ontologijų tikrinimo taisyklių sudarymas ir jų technologinis įgyvendinimas sukuriant ontologijų verifikavimo įrankį. Semantinis tinklas vis greičiau skverbiasi į medicinos, verslo ir akademinės bendruomenės pasaulį ir sparčiai tobulėja. Semantinį tinklą sudarančios ontologijos yra naudojamos iš įvairių šaltinių, kurie reikalauja skirtingų tikslų ir kokybės kriterijų. Ontologijos verifikavimas yra svarbus siekiant plačiai naudoti ontologiją. Išskiriami trys scenarijai, kurie reikalauja, kad ontologija būtų struktūriškai teisinga [3]:

- ontologijoje esančios klaidos ir elementų neišbaigtumas gali sukelti ne pilną ontologijos elementų pasiekiamumą. Todėl tvarkingai sukurtos ontologijos gali turėti aukštesnį duomenų panaudojimo laipsnį ir geriau bendradarbiauti su kitų dalykinių sričių ontologijomis.
- žmonėms, kuriantiems ontologijas, reikia būdo galinčio įvertinti jų rezultatus ir taisyklių apibrėžiančių ontologijos kūrimo principus. Tai leistų ontologijų inžinieriams pasiekti norimų rezultatų.
- lokalūs pokyčiai kuriamoje ontologijoje gali turėti įtakos ir kitoms lygiagrečiai modeliuojamoms ontologijoms. Ontologijų verifikavimo taisyklės ir įrankis leistų struktūrizuoti kūrimo procesą, ir automatiškai patikrinti klasių ir savybių vardus, domeno ir srities naudojimą savybių apibūdinime, anotacijų panaudojimą kitose ontologijos elementuose, remiantis tyrimo metu struktūrizuotomis taisyklėmis.

Ontologijos logikai patikrinti yra naudojami analizatoriai, tačiau ontologijos struktūrą tikrinančių įrankių nėra daug be to jų funkcionalumas skirtingas. Viena iš sričių, kuriai reikalingos struktūriškai taisyklingos ontologijos sukūrimas yra ontologijos elementų transformacija į *SBVR*, kurio metu siekiama transformuoti kuo daugiau ontologijos elementų. Todėl kiekviena klaida padaryta ontologijų kūrimo metu gali turėti neigiamos įtakos tolimesniems darbams arba transformacijoms, kuriose bus naudojama ontologija. Šiai problemai spręsti bus siekiama suformuoti pagrindines ontologijos tikrinimo taisykles, kuriomis remiantis vėliau būtų galima patikrinti jau sukurtą ontologiją, šiam tikslui sukuriant ontologijų verifikavimo įrankį.

1.3. Ontologijos apibrėžimas

1990 metais prasideda ontologijų kalbos užuomazga, kuri buvo žinių kalbos (angl. *Knowledge Representation*) evoliucijos rezultatas. Interneto atsiradimas paskatino ontologijų kalbos plėtimąsi tinkle.

Ontologija yra žinių modelis, kuris nusako dalykinės srities sąvokas, jų savybes bei tarpusavio ryšius. Pasak Thom'o Gruberg'o: „Ontologija – tai aiškiai išreikšta bendrai naudojamos konceptualizacijos specifikacija“ [5]. Taip pat ontologija dar vadinama tam tikros srities sąvokų visumos specifikavimas, išreikštas pavidalu. Egzistuoja įvairūs ontologijų sąvokos formavimo būdai, tačiau kiekvienas jų gali būti aprašomas formaliu komponentų rinkiniu [2].

Ontologijos sąvoka informatikoje naudojama keliomis prasmėmis. Minėtas, T. Gruberg'o, ontologijos apibūdinimas, sukuria tarpusavyje bendradarbiaujančių kategorijų sistemą, kuri naudojama vienokiems, ar kitokiems pasaulio fragmentams išreikšti ir apibūdinti jų tarpusavio ryšius.

Informacinių sistemų inžinerijoje ontologija aprašo konkretų kompiuterizuojamos veiklos fragmentą, tačiau tam gali būti naudojamos ir skirtingos ontologijos. Kuriant informacinę sistemą, vienos jos dalys būna sukuriamos vadovaujantis vienomis ontologijos nuostatomis, kitos visai kitomis nuostatomis. To pasekoje, kuriama sistema tampa nevienoda, nes jos dalis sujungti į bendrą, darniai veikiančią, sistemą yra labai sunku arba visai neįmanoma [3].

Ontologijų naudojimas ne tik padeda išreikšti dalykinę sritį tiek programinei įrangai, tiek žmonėms suprantama kalba, didinant naudojamos terminijos vienareikšmiškumą, bet tuo pačiu palengvina žinių naujose srityse struktūrizavimą. Taip pat suteikia sąveikų tarp skirtingų komponentų palengvinimą, išsiaiškinant atitikmenis tarp jų vartojamų terminų. Ontologija palengvina žinių platinimą ir pakartotinį panaudojimą. Be to, ontologijos naudoja aksiomas tam tikroms sąvokoms apriboti.

1.4. Semantinio tinklo apibrėžimas ir jam išreikšti naudojamų kalbų analizė

Semantinis tinklo apibrėžimas

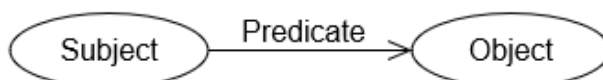
Siekiant suprasti *OWL 2* ontologijų panaudojimo ir aprašymo galimybes šiame skyriuje aptarsime semantinio tinklo sąvoką, ir pagrindinį tikslą.

Semantinio tinklo apibrėžimą sugalvojo Tim'as Bernersas Lee, pasak jo, semantinis tinklas yra dabartinio saityno išplėtimas, kuriame informacija turi aiškiai apibrėžtą prasmę ir užtikrina žmonių bei kompiuterių darnų tarpusavio bendradarbiavimą [5]. Sukūrus pasaulinį tinklą (angl. *World Wide Web*), pastarasis leido kompiuteriams tik atvaizduoti pateiktą informaciją nesigilinant į jos prasmę. Dabar kompiuteris gali ne tik atvaizduoti pateiktą informaciją monitoriuje, bet ir naudoti ją automatizavimo, integracijos ir pakartotinio panaudojimo tikslais įvairiose programose. Pagrindinis semantinio tinklo uždavinys – sukurti universalią terpę keistis duomenimis [6]. Semantinis pasaulinis tinklas yra ne atskiras, bet dabartinio pasaulinio tinklo išplėtimas, kuriame pateikiama informacija ir paslaugos turi aiškiai apibrėžtą prasmę, tokiu būdu leidžiant kompiuteriams ir žmonėms dirbti bendradarbiaujant. Semantinio pasaulinio tinklo kūrimu ir plėtojimu rūpinasi *W3C* (angl. *World Wide Web Consortium*), kuris skatina platų atvirų standartų tokių kaip *RDF* ir *OWL 2* naudojimą pramonės srityje ir akademinėje bendruomenėje [7]. Semantinio tinklo technologija leidžia kurti duomenų saugyklas, žodynus ir duomenų aprašymo taisykles [8].

RDF analizė

RDF yra viena iš sudedamųjų semantinio tinklo dalių, suteikianti prasmę internete naudojamam turiniui [1]. Kitaip tariant, tai informacijos keitimosi standartas, kuris aprašomas *XML* (angl. *Extensible Markup Language*) kalba ir yra naudojamas semantiniame tinkle.

Pagrindinė *RDF* kalbos idėja yra suskaidyti informaciją į trumpas frazes, kurių pagrindinės sudedamosios dalys būtų apibrėžiamos formatu: subjektas (angl. *Subject*), predikatas (angl. *Predicate*) ir objektas (angl. *Object*). Toks formatas, kitaip dar vadinamas tripletu (angl. *Triple*). Subjektai ir objektai nurodo realaus pasaulio išteklius, o predikatai žymi santykį tarp objektų ir subjektų. Minėtoji aibė sudaro kryptinį ir žymėtąjį *RDF* grafą (1.1 pav.). Pilnas *RDF* grafas atspindi faktų visumą, galiojančią tam tikroje dalykinėje srityje. Subjektai gali būti traktuojami kaip ištekliai (angl. *Resource*), o predikatai dar yra vadinami savybėmis (angl. *Properties*), objektai įgyja savybių reikšmės pavadinimą (angl. *Property Value*). Todėl *RDF* sakinyje gali turėti tokią struktūrą: ištekliai (angl. *Resource*), savybės (angl. *Properties*), savybių reikšmė (angl. *Property Value*) [10].



1.1 pav. *RDF* grafo struktūra

RDF grafe gali būti trijų tipų viršūnės: *URI*, *literal* ir *blank node*, plačiau aptarsime pirmąsias dvi, kurios ir bus naudojamos kuriant ontologijų verifikavimo įrankį. *URI* (angl. *Uniform Resource Identifier*) *RDF* kalboje yra naudojamas ištekliams ir savybėms internetiniame tinkle identifikuoti. Jie leidžia nepasimesti vardų ir metodų įvairovėje, kurie yra naudojami *HTTP*, *FTP* ir elektroniniame pašte [11]. *RDF* kalboje naudojamos *literal* reikšmės naudojamos tokiems dydžiams kaip kalba, data ar skaičiai nusakyti [12]. *RDF* sintaksės pavyzdys atvaizduojamas 1.2 paveiksle.

```

<owl:Class rdf:ID="Daiktas"/>
<owl:Class rdf:ID="Vieta">
<owl:Class rdf:ID="Panaudojimo sferos">

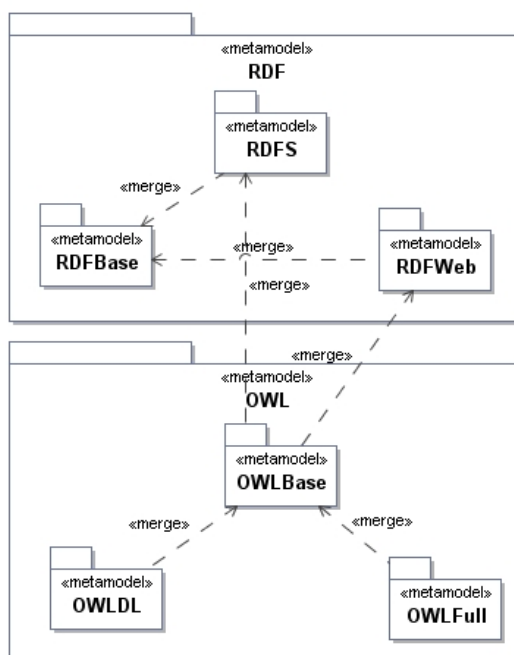
```

1.2 pav. *RDF* sintaksės pavyzdys

RDF tripletą sudarantys terminai neperteikia konkrečios dalykinės srities semantikos. Šiam tikslui naudojamas *RDF* plėtinys – *RDF Schema* (RDFS).

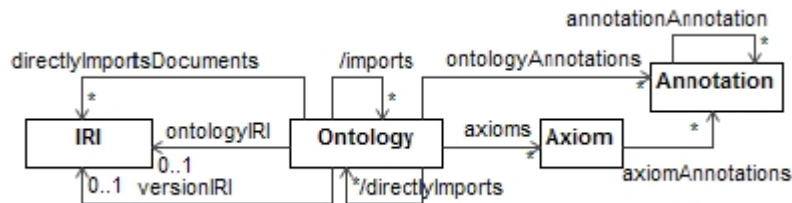
OWL 2 analizė

OWL 2 kalba skirta ontologijų kūrimui ir jų atvaizdavimui semantiniame tinkle [13]. *OWL 2* yra *RDF* žodyno pratęsimas, taigi bet koks *RDF* grafas formuoja *OWL 2* ontologiją. *OWL 2* meta-modelio struktūra ir jos priklausomybė nuo *RDF* meta-modelio yra pateikiama 1.3 paveiksle.



1.3 pav. *OWL 2* meta-modelio struktūra [15]

OWL 2 pasižymi turinio apdorojimo galimybėmis lyginant ją su kitomis ontologijų kalbomis, tokiomis kaip *RDF*, *RDF Schema* ar *XML* [14]. *OWL* buvo sukurta tam, kad atvaizduotų informaciją apie objektų kategorijas, jų tarpusavio ryšius. Tačiau dabar ši ontologijų kalba naudojama siekiant aiškiai pateikti žodynuose esančias išraiškas, nusakant jų prasmę ir tarpusavio ryšius. Detali ontologijos struktūra pateikiama 1.4 paveiksle.



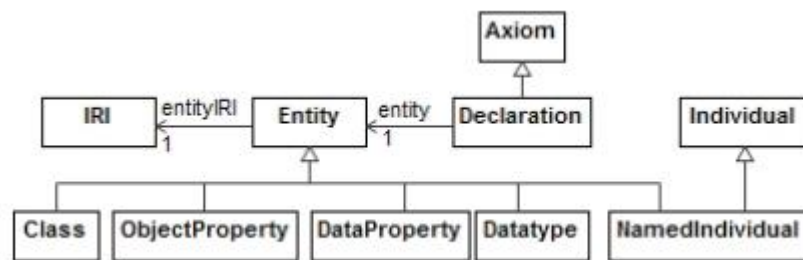
1.4 pav. OWL 2 ontologijos struktūra [17]

OWL 2 leidžia aprašyti žinias apie nagrinėjamą sritį naudojant šiuos konstruktus:

- esybės (angl. *Entities*): klases, individus, objektines ir duomenų savybes;
- savybių apribojimus (angl. *Property Restriction*);
- klasių išraiškas (angl. *Class Expressions*);
- duomenų diapazoną (angl. *Data Ranges*): duomenų tipus;
- aksiomas (angl. *Axioms*): klasių, savybių, teiginių;
- anotacijos (angl. *Annotations*);
- kitų ontologijų importavimo failus.

Esybės

Esybės išreiškia elementų vardus, kurie buvo panaudoti kuriant ontologiją. Visos esybės yra identifikuojamos per jos *IRI* (angl. *Internationalized Resource Identifier*) ir kiekviena esybė turi būti aprašyta naudojantis deklaracijos aksioma (angl. *Declaration Axiom*) [8]. OWL 2 esybės ir jų deklaravimas pateikiamas 1.5 paveiksle.



1.5 pav. OWL 2 esybės ir jų deklaravimas [17]

Klasės

Klasės yra pagrindinis ontologijos elementas. Jos apibūdina dalykinės srities idėją. Pavyzdžiui *Kelionės* ontologija pristato visus keliavimo būdus [17]. Klasės elementas yra sąvokų aibė, kurios nusako koks bus turinys. Be to sąvokų aibė apibrėžia nagrinėjamoje dalykinėje srityje vartojamą terminiją. Pavyzdžiui, kelionės apibrėžimą būtų galima nusakyti tam tikrais terminais, kurie susiję su minėta dalykine sritimi: vieta (*miestas, kaimas*), sustojimai (*viešbutis, poilsiavietė*) ir transportas (*lėktuvas, automobilis, keltas, traukinys, laivas*). Klasės dažniausiai aprašomos taksonomijomis¹, per kurias gali būti taikomas paveldėjimo principas. [2]

Klasės identifikatorius apibūdina klasę per klasės vardą, kuri pristatoma kaip *IRI* nuoroda. Klasės elementai apibrėžiami naudojant struktūrą: `owl:Class`, tai yra `rdfs:Class` poklasė. Klasė *kelionėLektuvu* gali būti aprašyta tokiu būdu:

¹ taksonomija - (takso- + gr. nomos - dėsnis): 1. mokslas, tiriantis taksonus - bet kurio hierarchinio lygio objektus, išskiriamus remiantis tam tikrais kriterijais, jų grupuotes, subordinaciją; 2. mokslas, tiriantis pačius klasifikavimo principus. Matematiškai taksonomiją galima apibrėžti kaip klasifikaciją, kurioje visi objektai yra išdėstomi į tam tikrą medžio struktūrą. Šio medžio viršūnę paprastai sudaro vienintelis klasifikacijos elementas, kuriam priklauso visi kiti šios klasifikacijos objektai. Taksonas, einantis žemiau už viršūnę, yra labiau specifinis ir išskiria tam tikrą klasifikuojamų objektų aibę pagal tam tikrą požymį.

```

<owl:Class rdf:ID="kelionėLėktuvu">
  <rdfs:subClassOf rdf:resource="#keliavimoBūdas"/>
</owl:Class>

```

OWL 2 kalboje apibrėžiamos dvi pagrindinės klasės `owl:Thing`, kuri yra bendra ir įtraukia viską. Kiekviena aprašoma klasė yra `owl:Thing` poklasis.

Pavyzdžiui, kelionės ontologijai aprašyti, galimas trijų klasių sukūrimas: *Kelionė*, *KelionėsPaskirtis* ir *KeliavimoBūdas*:

```

<owl:Class rdf:ID="Kelionė"/>
<owl:Class rdf:ID="KelionėsPaskirtis"/>
<owl:Class rdf:ID="KeliavimoBūdas"/>

```

Svarbu paminėti, kad klasės gali būti ir sudėtinės, kurioms naudojami šie OWL 2 konstruktai:

- `owl:intersectionOf` - loginis AND (konjunkcija);
- `owl:unionOf` – loginis OR (disjunkcija);
- `owl:complementOf` – loginis NOT (neigimas).

Trijų tipų klasės aprašymai pristato pažangųjį klasės aprašymą, kuriame naudojama deskriptyvioji logika. Išraiškos sankirta, sąjunga, papildymas gali būti reprezentuojami kaip *ir* (angl. *And*), *arba* (angl. *Or*), ir *ne* (angl. *Not*) klasių operatoriai. Šių klasių aprašymui naudojamos išraiškos: `owl:intersectionOf`, `owl:unionOf`, `owl:complementOf`.

Savybių elementai

Kuriant ontologiją yra išskiriamos dvi pagrindinės kategorijų savybės:

- objektinės savybės (angl. *Object properties*), sujungia vieną objektą su kitu objektu. Tokiu būdu nurodoma, kokia savybė priklauso tam tikram objektui. Objektinių savybių pavyzdys pateikiamas 1.6 paveiksle.

```

<owl:ObjectProperty rdf:about="#economy;has_model">
  <rdfs:label xml:lang="lt">produktas turi modelį</rdfs:label>
  <ann:label_sbvr xml:lang="lt">produktas turi modelį</ann:label_sbvr>
  <ann:label_lt xml:lang="lt">produktas turi modelį</ann:label_lt>
  <rdfs:range rdf:resource="#economy;model"/>
  <rdfs:domain rdf:resource="#economy;product"/>
</owl:ObjectProperty>

```

1.6 pav. Objektinių savybių sudarymo struktūros pavyzdys

Nurodytame pavyzdyje ryšių aibė apibrėžia asociacijas tarp srities sąvokų. Ontologijos dažniausiai aprašomos dvejetainiais ryšiais. Pirmasis argumentas nusako domeną (angl. *Domain*), o antrasis – sritį (angl. *Range*). Taigi šiuo atveju *model* yra sritis, *product* yra domenas, *has_model* – ryšio pavadinimas [4].

- duomenų savybės (angl. *Datatype Properties*) individualius elementus suriša su duomenų tipo reikšmėmis. Surišimo rezultatas nurodomas ryšiais tarp klasės ir RDF aprašo arba XML schemas. Duomenų savybių pavyzdys pateikiamas 1.7 paveiksle.

```

<owl:DatatypeProperty rdf:about="#events;size_decimal_value">
  <rdfs:label xml:lang="lt">dydis dešimtaine reikšme</rdfs:label>
  <ann:label_lt xml:lang="lt">dydis dešimtaine reikšme</ann:label_lt>
  <ann:label_sbvr xml:lang="lt">dydis dešimtaine reikšme</ann:label_sbvr>
  <rdfs:domain rdf:resource="#events;size"/>
  <rdfs:range rdf:resource="#xsd;decimal"/>
</owl:DatatypeProperty>

```

1.7 pav. Duomenų savybių sudarymo struktūros pavyzdys

Anotacijos

Kuriant ontologijas dažnai reikia nurodyti vienokią ar kitokią informaciją (komentarą (*rdfs: comment*), kalbą (*rdfs:label xml:lang="en"*) ir pan.), kuri leidžia užtikrinti, kad ontologija bus suprantama ne tik kompiuteriui, bet ir ją nagrinėjančiam žmogui. Nepriklausomai nuo pagrindinės ontologijos struktūros, šiam tikslui naudojamos anotacijos. Pastarosios susideda iš anotacijos savybės (angl. *Annotation Property*) ir anotacijos reikšmės (angl. *Annotation Value*) (*literal arba IRI*) [8].

Kurdamas ontologiją vartotojas gali praplėsti anotacijas t. y. suformuodamas naujas. Tokiu būdu buvo suformuota anotacija *label_sbvr* pritaikant *SBVR* standarto elementų pavadinimų struktūrą: *domenas + veiksmazodis + sritis*. *SBVR* nuo ontologijų skiriasi tuo, kad aprašo dalykinę sritį natūralia kalba, kuri yra suprantama ne tik kompiuteriams, bet ir žmonėms. Anotacijos *label_sbvr* pavyzdys pateikiamas 1.8 paveiksle. Pavyzdyje vaizduojama *label_sbvr* struktūra *dokumente minimas objektas* sudaroma panaudojant domeną – *dokumentas* (angl. *Document*), veiksmazodį – *minimas* ir sritį – objektas (angl. *Object*).

```
<owl:ObjectProperty rdf:about="&semlt;refers_to_object">
  <rdfs:label xml:lang="lt">dokumente minimas objektas</rdfs:label>
  <ann:label_lt xml:lang="lt">dokumente minimas objektas</ann:label_lt>
  <ann:label_sbvr xml:lang="lt">dokumente minimas objektas</ann:label_sbvr>
  <rdfs:domain rdf:resource="&semlt;document"/>
  <owl:inverseOf rdf:resource="&semlt;is_referred_in_document"/>
  <rdfs:range rdf:resource="&semlt;object"/>
</owl:ObjectProperty>
```

1.8 pav. Anotacijos *label_sbvr* sudarymo struktūros pavyzdys

Ontologijos elementų vardų stilius

Kuriant ontologijos elementų vardus svarbu nenaudoti tokių simbolių kaip: kabutės, grotelės ir pan. Klasių ir savybių vardai turi būti išreiškiami aiškiai suprantama ir perskaitoma forma [18].

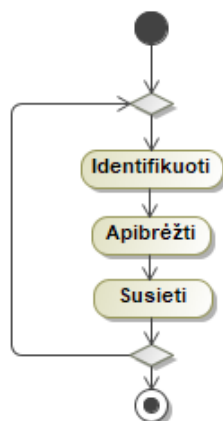
Remiantis gerosiomis praktikomis kuriant ontologijų elementų vardus svarbu išlaikyti vieningą stilių t. y. žodžių junginiams naudoti gyvatės (angl. *Snake Case*) arba kupranugario stilių (angl. *Camel Case*) ir vengti trumpinių naudojimo, pavyzdžiui: geriau yra naudoti pilną žodį *Kaunas*, negu trumpinį *Kn*.

Ontologijų kūrimas

Kaip paaiškėjo ontologijų kalbos analizės metu iš nagrinėtų šaltinių nėra teisingiausio kelio ar metodikos kurti pasirinktos dalykinės srities ontologijas. Tačiau remiantis gerosiomis praktikomis galime išskirti svarbiausias ontologijos kūrimo žingsnius [17] (pav. 1.9):

- objekto identifikavimas;
- klasių apibrėžimas;
- elementų susiejimas.

Objektus, esančius ontologijoje, galima išreikšti daiktavardžiais, o juos susiejančius ryšius – veiksmazodžiais. To pasekoje galime priėti išvados, kad pagrindinius ontologijų kūrimo principus galima aprašyti taisyklėmis.



1.9 pav. Ontologijos kūrimo žingsniai

Ontologijų kūrimo ir redagavimo įrankiai vartotojams leidžia vizualiai manipuluoti, patikrinti ir kurti ontologijas. Vienas pagrindinių ontologijų kūrimo įrankių – *Protégé*.

Protégé yra grafinis *OWL* kalbos redaktorius, kurio pagalba kuriama ontologija formuojama hierarchijų principu. Redaktorius sukurtas *Java* programavimo kalba, taip pat yra lanksčios architektūros, todėl turi daug įskiepių, kurių dėka jo funkcijos yra praplečiamos. *Protégé* įrankiu sukurtos ontologijos gali būti eksportuojamos į tokius formatus kaip *RDF(S)*, *OWL* ir *XML Schema*. Ontologijos loginį teisingumą *Protégé* įrankyje užtikrina įskiepiai: *Pellet*, *Fact++*, *RacerPro*.

1.5. Esamų sprendimų analizė

Tikslingas ontologijos įvertinimas nėra išnagrinėtas iki galo. Dauguma sistemų geba tikrinti ontologijos loginę struktūrą (*Pellet*, *Fact++*, *RacerPro*), tačiau ne jos sintaksinę struktūrą. Šiame tiriamajame darbe įvertinta ontologijos struktūra remiasi elementų transformavimo į *SBVR* modelį suderinamumu [17]. Siekiant sukurti struktūriškai ir stilistiškai teisingą ontologiją buvo apibrėžtos bendrinės ontologijos tikrinimo taisyklės, kuriomis remiantis ontologijos kūrimo specialistas gebėtų pritaikyti praktikoje. Tačiau vertinant žmogiškosios klaidos faktorių, buvo atsižvelgta į tai, kad ontologijos kūrėjas gali suklysti kurdamas ontologiją, todėl siekiama sukurti įrankį, kuris išpėtų vartotoją apie padarytas klaidas. Gautą rezultatą būtų galima išsaugoti ir toliau nagrinėti ontologijų redaktoriuje nepažeidžiant pagrindinės ontologijos struktūros t. y. pridėdant išpėjimų ir klaidų anotaciją.

Atsižvelgiant į jau esamus sprendimus buvo išskirti dar du įrankiai, kurie geba įvertinti ontologijos struktūrą. Vienas iš jų tai *Protégé* įrankio įskiepis – *OntoCheck*, o kitas – *OOPS ! (Ontology Pitfall Scanner!)*.

OntoCheck analizė

Ontologijų kūrimo politika reikalauja nurodyti metaduomenų atributus ir pilnai aprašyti ontologijoje naudojamų elementų vardus, tačiau ontologijoje naudojamų metaduomenų išsamumo vis dar trūksta. Todėl buvo sukurtas *Protégé* įskiepis *OntoCheck*, kuris galėtų patikrinti ontologijos struktūros nuoseklumą ir kokybę. Įrankis yra pristatomas kaip ontologijų validatorius, kuris tikrina ontologijos teisingumą dalykinės srities atžvilgiu, tačiau turi keletą funkcijų, kurios tikrina/verifikuoja ontologijos elementų struktūrą [25].

Įrankis yra lengvai importuojamas kaip *Protégé* įskiepis. Todėl rasti pažeidimai gali būti greitai ištaisomi. Įrankio lange vartotojas gali pasirinkti subjektą pateiktą atidarytoje ontologijoje, pavyzdžiui: *rdfs:label*, *dc:comment*. Visas subjektų egzistavimas gali būti tikrinamas t. y. vartotojas gali nurodyti, kad visos klasės turi turėti bent vieną *label* anotaciją arba bent vieną natūralios kalbos apibrėžimą. Klasėms trūkstamų reikiamų duomenų jie gali būti papildomi tiesiogiai *Protégé* redaktoriuje. Įrankis leidžia tikrinti minimalų ir maksimalų simbolių ir žodžių naudojimą, pavyzdžiui, žodis yra trumpesnis nei 2 simboliai arba aptikti elementų vardus, kuriuose yra naudojami žodžiai sudaryti, pavyzdžiui, iš 50 simbolių. Įrankis geba patikrinti skyrybą ir didžiųjų bei mažųjų raidžių naudojimą ontologijoje. *OntoCheck* suteikia galimybę palyginti reikšmes, kurios yra naudojamos anotacijose ir metaduomenyse su reikšmėmis esančiose skirtingose klasėse, pavyzdžiui: *rdf:ID* gali būti pažymėtas jei atitinka *rdfs:label* naudojamą toje pačioje klasėje arba kitame operatoriuje. Įrankis taip pat atlieka ir ontologijos vertinimą, jis parodo procentinį susikirtimo taškų skaičių esantį nagrinėjamoje ontologijoje [22]. Pagrindiniai *OntoCheck* įrankio privalumai ontologijų verifikavime:

- naudojamas kaip įskiepis ontologijų redagavimo įrankyje *Protégé*, turintis perprantamą vartotojo sąsają;
- ontologijos elementų rašymo stiliaus tikrinimas, t. y. elementų rašymas tik didžiosiomis arba tik mažosiomis raidėmis, iš didžiosios arba tik iš mažosios raidės;
- ontologijos elementų vardų atskyrimo stiliaus tikrinimas, t. y. žodžiai atskiriami brūkšniu apačioje, tašku arba nenaudojamas joks simbolis atskirti žodžiams;
- skaitinių išraiškų tikrinimas *label* elementuose, pavyzdžiui: ieškant skirtingų verčių arba iškvietimo indikatorių;
- reguliarių išraiškų (angl. *Regex*) tikrinimas t. y. *prefix*, *infix* ir *postfix*;
- ontologijoje naudojamų vardų minimumo ir maksimumo reikšmių tikrinimas.

OOPS!

Dar vienas įrankis galintis patikrinti ontologijos struktūrą, tai *OPPS!*, jis padeda ontologijų specialistui aptikti įvairias anomalijas ir užtikrina ontologijos kokybę aptikdamas dažniausiai pasitaikančias klaidas [26].

Įrankis gali [24]:

- kurti daugiareikšmius elementus t. y. ontologijos elementai, kurių vardai turi skirtingas prasmes yra įtraukiami į ontologiją pažymint, kad jie turi daugiau nei vieną konceptualią idėją. Pavyzdžiui, žodis *kinas* gali būti vieta ir meninė disciplina;
- kurti sinonimus kaip klasių elementus t. y. kelios klasės, kurios yra identifikuojamos kaip sinonimai ir sujungiamos ryšiu *owl:equivalentClass*, pavydžiui: klasės *automobilis*, *mašina*, *mechanizmas* gali būti išreiškiamos kaip lygiavertės;
- kurti ryšius panaudojant *is* ryšį vietoje *rdfs: subclassOf*, *rdf:type* arba *owl:sameAs*. Ryšis *is* kuriamas tam, kad nenaudoti pirmykščių ryšių nusakančių poklasių santykį (*rdfs: subclassof*), tipą (*rdf:type*) arba lygybes (*owl:sameAs*);
- kurti nesusijusius ontologijos elementus t. y. ontologijos elementai (klasės, ryšiai ar atributai) gali būti sukurti nesusiejant jų su tikrinama ontologija;
- aptikti klaidingus ryšius, kurie žymį atvirkštinį ryšį (angl. *Inverse Relationships*) t. y. kai dvi sąlygos yra sujungiamos atvirkštiniu ryšiu nors iš tikrųjų toks ryšio panaudojimas klaidingas. Pavyzdžiui: kažkas gali būti perkama ir kažkas gali būti parduodama, tad šiuo atveju yra *Perkama* ir yra *Parduodama* negali būti naudojamas kaip atvirkštinis ryšys;
- aptikti elementus, kuriuose nėra panaudojamos arba panaudojamos ne pagal ontologijos prasmę anotacijos tokios kaip: *rdfs: label*, *lemon: LexicalEntry*; *skos: prefLabel*, *skos: altLabel*, *rdfs: comment* arba *dc: description*;
- tikrinti ar ontologijos savybių elementai turi arba neturi nurodytą domeną ir sritį;
- nurodyti jei ontologijos elementų hierarchija (angl. *Max Depth*) yra per didelė t. y. ontologijos hierarchija yra kuriama taip, kad galutinis elementas negali turėti papildomų atvejų (angl. *Instances*), nes jie patys gali būti traktuojami kaip naujos dalykines srities ontologijos, užuot kuriami kaip klasių elementai;
- aptinka skirtingu stiliumi užrašytų ontologijų elementus t. y. kai ontologijos elementai nėra užrašyti pagal tą pačią konvenciją, pavyzdžiui: *kupranugario* stiliumi;
- aptikti klases su tokiais pat *label* atributais, pavyzdžiui: dvi arba daugiau klasių turi tokį patį anotacijos pavadinimą kalbai *rdfs: label annotation*;
- aptinka nedeklaruotas klases t. y. kai ontologijos elementas yra naudojamas kaip klasė, tačiau nedeklaruotas naudojant išsireiškimų *owl:Class* arba *rdfs Class*;
- aptinka nedeklaruotas savybes t. y. kai ontologijos elementai yra naudojami kaip savybės, tačiau neaprašomos naudojant struktūrą *rdf: Property*, *owl:ObejectProperty* arba *owl: Datatypeproperty*;
- taip pat įrankis gali patikrinti ontologijos licenciją, *URI* egzistavimą, dviprasmišką vardų naudojimą.

Įrankiu vertinimui pasirinktos tos savybės, kurios naudojamos ontologijų verifikavimui. Iš atliktos įrankių charakteristikos analizės galima teigti, kad ontologijų verifikavimo srityje pranašesnis yra *OOPS!* verifikatorius. Toliau aptarsime kokios savybės gali būti naudojamos naujai kuriamame sprendime (1.1 lentelė).

1.1 lentelė. Esamų sprendimų palyginimas

| Savybė \ Įrankis | <i>OntoCheck</i> | <i>OPPS!</i> | Siekiamas įrankis |
|---|------------------|----------------|-------------------|
| Ontologijų kalba | <i>OWL/RDF</i> | <i>OWL/RDF</i> | <i>OWL/RDF</i> |
| Vartotojo sąsajos palaikymas | ✓ | ✓ | ✓ |
| Reikalingas instaliavimo procesas | ✓ | ✗ | ✓ |
| Tikrina savybių domeną ir sritį | ✗ | ✓ | ✓ |
| Nustato ontologijos hierarchijos dydį | ✗ | ✓ | ✓ |
| Tikrina ontologijos elementų vardų užrašymo stilių | ✗ | ✓ | ✓ |
| Tikrina nedeklaruotas klases | ✗ | ✓ | ✗ |
| Aptinka klases su tokiais pat <i>label</i> elementais | ✗ | ✓ | ✗ |
| Atlieka klasių vardų rašymo morfologinę analizę | ✗ | ✗ | ✓ |
| Atlieka savybių vardų rašymo morfologinę analizę | ✗ | ✗ | ✓ |
| Tikrina kalbos anotacijos egzistavimą | ✗ | ✗ | ✓ |
| Tikrina <i>label_sbvr</i> egzistavimą ir struktūrą | ✗ | ✗ | ✓ |

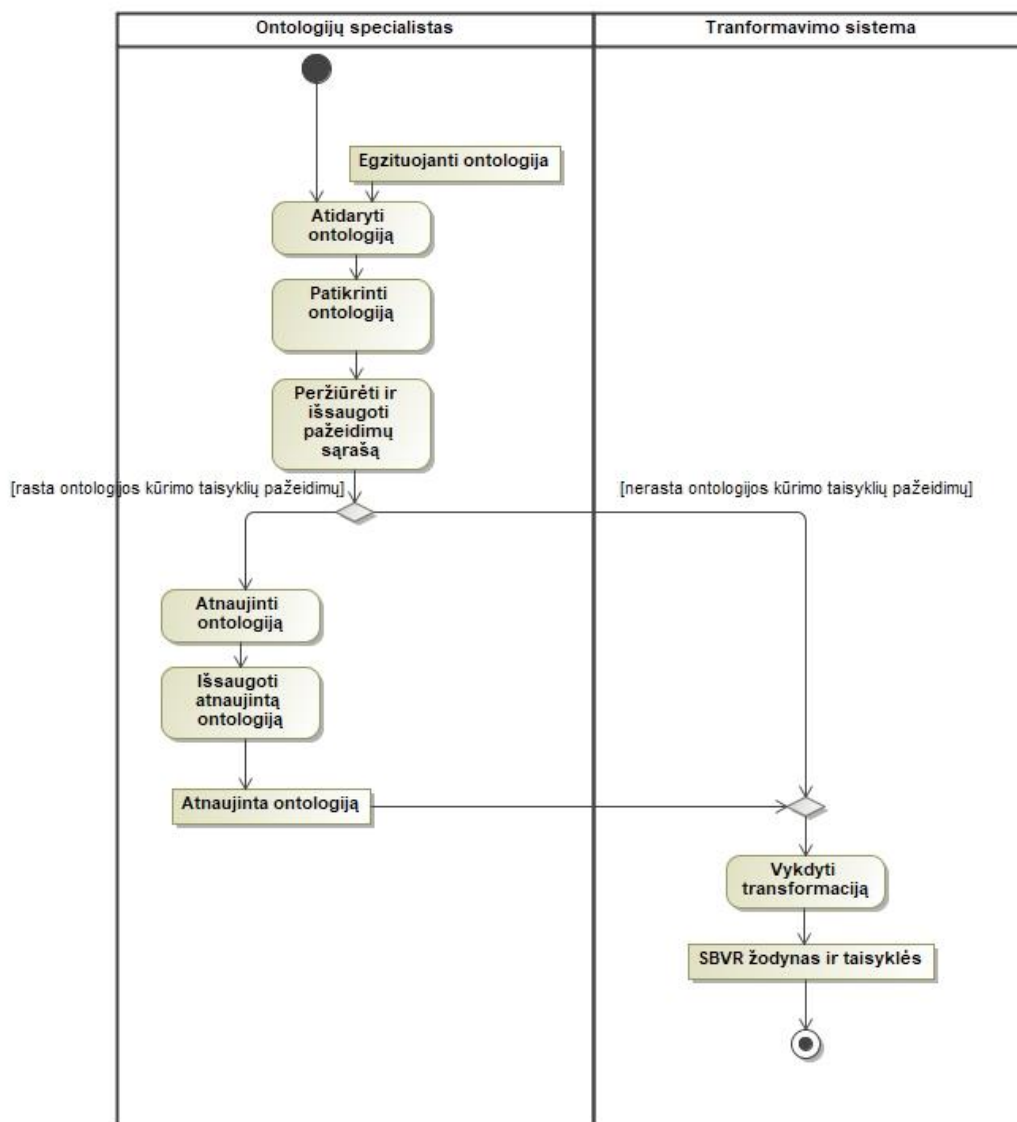
✗ - įrankis neišpildo reikalavimo

✓ - įrankis išpildo reikalavimą

1.6. Siekiamo sprendimo apibrėžimas

Siekiami sprendimai: sukurti ontologijos tikrinimo taisykles remiantis atlikta ontologijos elementų analize, esamais sprendimais, sukurti metodiką, kuri būtų naudojama kuriant ontologijas ir jas tikrinant. Sukurti ontologijų verifikavimo įrankį, didžiausią dėmesį skiriant ontologijos tikrinimo taisyklių išpildymui. Išanalizuoti ontologijų tikrinimo galimybes panaudojant sukurtą įrankį: ištirti dažniausiai daromas klaidas, pateikti gautus rezultatus.

Viena iš sričių, kuriai yra svarbus ontologijos verifikavimo procesas – tai jos transformacija į *SBVR* modelį. Preliminarus transformacijos proceso vykdymas pateikiamas 1.10 paveiksle. Norėdamas atlikti transformaciją ontologijų specialistas įkelia norimą transformuoti ontologiją. Siekiant užtikrinti ontologijos korektiškumą transformuojama ontologija patikrinama verifikavimo įrankyje. Dalykinės srities specialistas turi galimybę peržiūrėti klaidų sąrašą, išsaugoti jį ir pagal rekomendacijas atnaujinti ontologiją. Ontologijos verifikavimo įrankiu patikrinta ir struktūriškai teisinga ontologija įkeliamą į transformavimo sistemą, ko pasėkoje gaunamas korektiška ontologijos elementų transformacija į *SBRV* žodyną ir taisykles.



1.10 pav. OWL 2 į SBVR transformavimo procesas įtraukiant ontologijos verifikavimą

1.7. Analizės išvados

1. Literatūros analizė parodė, kad ontologijų patikrinimas, prieš vykdant transformaciją, yra svarbus siekiant gauti tikslius transformacijos rezultatus. Taip pat ontologijos patikrinimas yra svarbus siekiant išvengti žmogiškosios klaidos faktoriaus ontologijų kūrimo metu.
2. Atlikus *RDF* ir *OWL 2* kalbų analizę buvo prieita prie išvados, kad *OWL 2* ne tik papildo, bet ir praplečia *RDF* kalbos galimybes, taip suteikdama platesnes paieškos galimybes semantiniame tinkle.
3. Ontologijos elementų analizė parodė, kad nėra vieningos metodikos aprašančios ontologijos kūrimo procesą. Tačiau galima kurti ontologijas laikantis bendrinio ontologijos tikrinimo aprašo taisyklių.
4. Egzistuojančių sprendimų analizė parodė, kad yra sistemų galinčių analizuoti ontologijos struktūrą, tačiau jie neužtikrina jos morfologinio nagrinėjimo, negali patikrinti kalbos anotacijos egzistavimo, taip pat jų rezultatai negali būti perkelti į ontologijų redaktorių norint ištaisyti gautus pažeidimus. Egzistuojantys įrankiai netikrina *label_sbvr* anotacijos egzistavimo ir jos sudarymo struktūros, kuri yra ypač svarbi ontologijos transformacijai į *SBVR* modelį. Šiai spragai užpildyti siekiama sukurti įrankį gebantį atlikti visus minėtus veiksmus.

5. Ontologijų verifikavimo įrankio realizavimui, dėl patogaus karkaso ir plataus taikymo, buvo pasirinkta *C#* programavimo kalba. Ontologijų redagavimui ir klaidų anotacijų kūrimui buvo pasirinktas dažniausiai atnaujinamas ontologijų redagavimo įrankis *Protégé*.

2. ONTOLOGIJŲ VERIFIKAVIMO ĮRANKIO REIKALAVIMŲ SPECIFIKACIJA IR PROJEKTAVIMAS

Daugelis ontologijos kūrimo būdų reikalauja didžiulių pastangų ir subjektyvaus vertinimo kuriant ontologiją bei ją transformuojant. Kita vertus gebėjimas projektuoti ir laikytis ontologijos tikrinimo taisyklių reikalauja patirties ir žinių tiek iš dalykinės srities eksperto, tiek iš ontologijų modeliavimo kalbos. Tačiau dėl praktinio pritaikymo ontologijų skaičius sparčiai auga, jos tampa didesnės apimties, todėl vis sunkiau užtikrinti jų taisyklingumą. Ontologijų modeliuotojams jų korektiškumas yra pagrindinis prioritetasis, todėl tokie sunkumai kaip įvairių anomalijų pasirodymas ontologijoje arba techninės klaidos, vaidina svarbų vaidmenį ontologijos kūrimo eigoje.

Taigi pagrindinis projektuojamo įrankio tikslas užtikrinti galimybę tikrinti ontologiją prieš atliekant kitus veiksmus, t. y. prieš vykdant vienokią ar kitokią transformaciją įrankis padės pastebėti ontologijos struktūros nekorektiškumą. Ontologijų verifikatorius padės sumažinti galimų klaidų skaičių tokių kaip: neapibrėžtų savybių ir klasių, netaisyklingai suformuotų duomenų savybių, gramatinių ir kt.

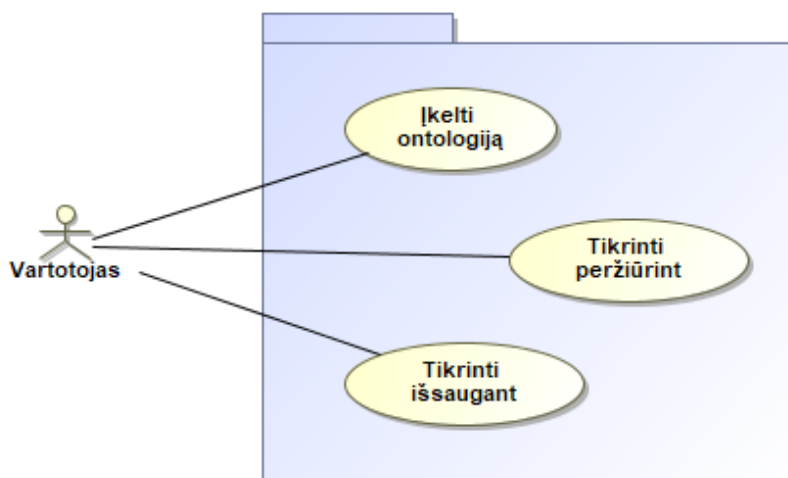
2.1. Reikalavimų specifikacija ontologijų verifikavimo įrankiui

Funkciniai reikalavimai

Funkciniai reikalavimai nusako, pagrindinį sistemos funkcionalumą. Todėl naujai kuriamas ontologijų verifikavimo įrankis turi leisti atlikti tokias funkcijas:

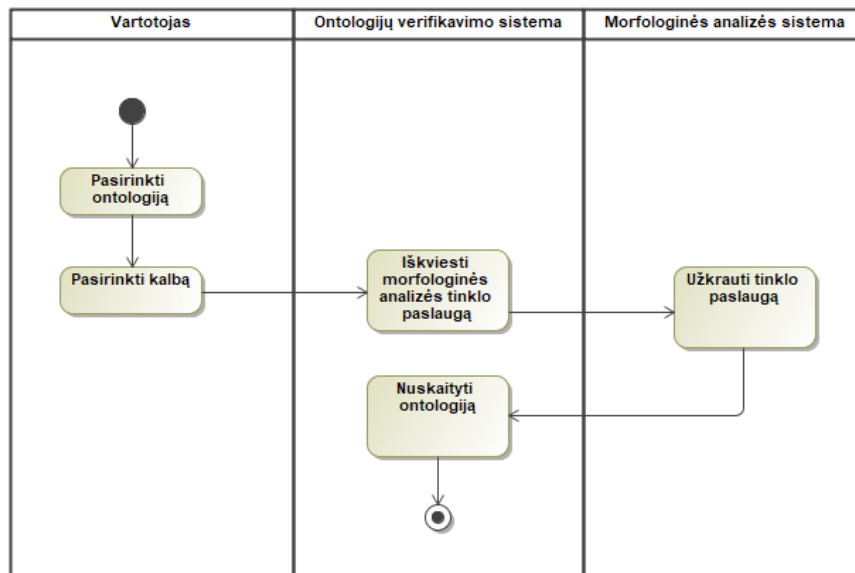
1. Vartotojui atsidarius sistemos langą turi būti suteikiama galimybė įkelti norimą nagrinėti ontologiją, kuri aprašyta *RDF/OWL* kalbomis;
2. Įrankis, vartotojo inicijuotas, pradeda ontologijos tikrinimo procesą, vartotojui turi būti suteikiama galimybė pasirinkti kokius ontologijos elementus norima patikrinti: objektines savybes, duomenų savybes, klases ir jų skaičių, ontologijos elementų vardų rašymo stilių, objektinių ir duomenų savybių sudarymo struktūrą. Vartotojui turi būti suteikiama galimybė peržiūrėti ontologijoje rastus įspėjimus tiesiogiai programoje. Rastų klaidų atvaizdavimas nurodomas eilučių tikslumu.
3. Vartotojui turi būti suteikiama galimybė pasirinkti, kuriuos ontologijos elementų tikrinimo rezultatus norima peržiūrėti išsaugant. Tokiu atveju ontologijos verifikavimo įrankis sukuria naują failą, kuriame surašomi tikrinti elementai su juose rastomis klaidomis, į kurį papildomai importuojama klaidų ir įspėjimų anotacijos failas ir originali ontologija, kuri buvo tikrinta. Išsaugotą ontologijos failą turi būti galima peržiūrėti ir redaguoti įsikėlus jį į ontologijų redaktorių.

Kompiuterizuojamų panaudojimo atvejų diagramos, kuri vaizduoja ontologijos verifikavimo procesą, veikimas pateikiamas 2.1 paveiksle. Sistemoje gali leisti atlikti tokias funkcijas: *įkelti ontologiją, tikrinti peržiūrėti, tikrinti išsaugant*.



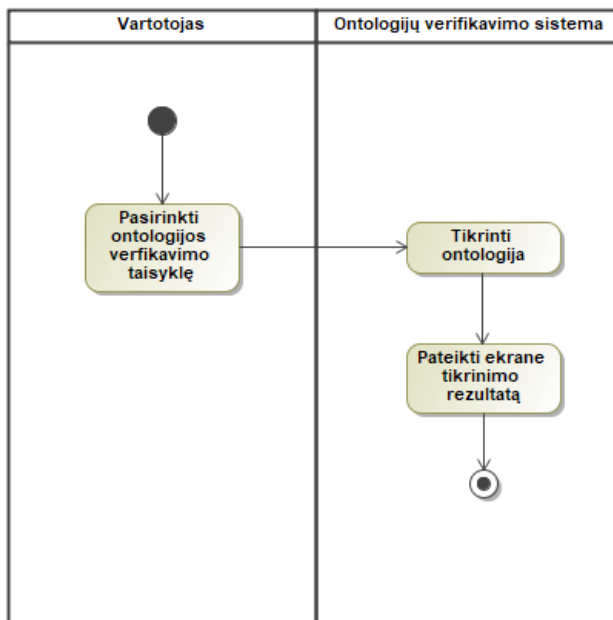
2.1 pav. Panaudojimo atvejų diagrama

Plačiau aptarsime panaudojimo atvejus, kurių diagrama pavaizduota 2.1 paveiksle. Panaudojimo atvejui *įkelti ontologiją* vartotojui yra suteikiama galimybė pasirinkti ontologiją, tada prašoma nurodyti kalbą. Tokiu būdu įrankis iškviečia morfologinės analizės tikrinimo paslaugą, kuri atliks morfologinę ontologijos elementų vardų analizę. Ontologijos įkėlimo metu vyksta tinklo paslaugos užkrovimas jam įvykus įrankis pradeda ontologijos nuskaitymo procesą. Panaudojimo atvejo *įkelti ontologiją* veiklos diagrama pateikiama 2.2 paveiksle.



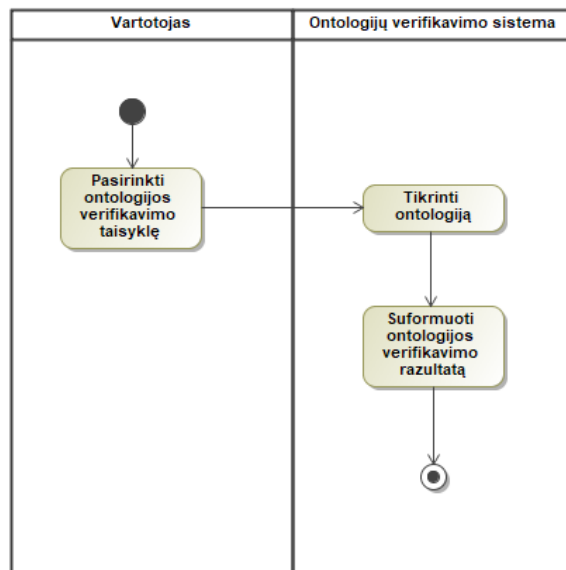
2.2 pav. Panaudojimo atvejo *įkelti ontologiją* veiklos diagrama

Įkėlus ontologiją vartotojui turi būti suteikta galimybė ją peržiūrėti ontologijų verifikavimo įrankyje. Panaudojimo atvejo *tikrinti peržiūrėti* metu vartotojas iš duoto sąrašo pasirenka elementų grupę, kurią nori patikrinti, o klaidas tiesiogiai peržiūrėti verifikavimo programoje. (Pav. 2.3)



2.3 pav. Panaudojimo atvejo *tikrinti peržiūrėti* veiklos diagrama

Verifikavimo įrankis turi suteikti galimybę įkeltą ontologiją tikrinti ją išsaugant. Panaudojimo atvejo *tikrinti išsaugant* veiklos diagrama pateikta 2.4 paveiksle.



2.4 pav. Panaudojimo atvejo *tikrinti išsaugant* veiklos diagrama

Nefunkciniai reikalavimai

Nefunkciniai reikalavimai nusako savybes, kuriomis turėtų pasižymėti būsimas ontologijų verifikavimo įrankis.

Verifikatoriaus paskirtis: ontologijų verifikavimo įrankis turi būti naudojamas ontologijų tikrinimo taisyklių tyrimui. Svarbiausias tyrimo akcentas yra bendrinių ontologijos tikrinimo taisyklių sukūrimas ir jų realizavimas įrankyje.

Naudojamos ontologijos: eksperimentui turi būti naudojamos viešai prieinamos ontologijos. Verifikavimo rezultatų išsaugojimo metu turi būti naudojama įspėjimų ir klaidų anotacijų ontologija.

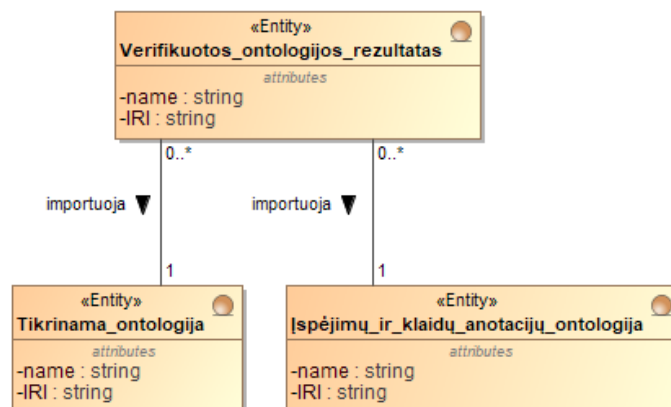
Reikalavimas spendimui: ontologijų verifikatorius turi būti kuriamas naudojant *C#* programavimo kalbą.

Diegimo aplinka: *Windows 7/8/10* operacinė sistema.

2.2. Ontologijų verifikavimo dalykinės srities modelis

Dalykinės srities modeliavimas yra priemonė leidžianti apibrėžti modeliuojamo įrankio informacinius poreikius. Modeliavimo metu identifikuojamos pagrindinės savybės, jų atributai ir ryšiai, kuriais jie siejasi tarpusavyje.

Ontologijų verifikavimo įrankio dalykinės srities diagrama pateikiama 2.5 paveiksle.



2.5 pav. Ontologijų verifikavimo įrankio klasių modelis

2.3. Formalus sprendimo aprašas

Norint sukurti struktūriškai teisingą ontologiją analizės dalyje buvo aptarti pagrindiniai elementai ir atributai, kurie sudaro ontologiją. Remiantis atlikta analize struktūrizuotos ir išskirtos devynios taisyklės, kuriomis remiantis ontologijų kūrimo specialistas galėtų sukurti taisyklingai aprašytą (pasirinktų nuostatų atžvilgiu) ontologiją. Kiekviena taisyklė yra išskiriama skaičiais, kurie bus naudojami reprezentuojant taisyklės tolimesniuose skyriuose.

Taisyklės išskirstytos į keturias grupes: elementų vardų, savybių, anotacijų ir stiliaus formavimo. Taisyklių tekstinis aprašymas, jų pateikimas pirmos eilės predikatų logika ir pavyzdžiai nurodomi tolimesniame tekste.

Ontologijos elementų vardų formavimo taisyklės

Ontologijos pagrindą sudaro klasės. Klasės apibūdina dalykinės srities bendrą supratimą. Todėl jų vardų nurodymas yra privalomas kiekvienoje ontologijoje. Remiantis gerosiomis praktikomis klasių vardai ontologijoje nurodomi daiktavardžio forma [18]. To pasėkoje galime išskirti pagrindinę ontologijos tikrinimo taisyklę:

Kiekvienos ontologijos klasės vardas turi būti *daiktavardis* arba *daiktavardinė frazė*.

$$\forall x \in X \ v \in V [klasės_vardas(x, v) \Rightarrow \exists v1 \in V [daiktavardis(v1) \wedge klasės_vardo_struktūra(v, v1)].] \quad (1)$$

čia X – klasių aibė, V – žodžių aibė

Pavyzdžiui:

Taisyklė laikoma nepažeista, kai klasės vardas yra daiktavardinė frazė:

x = http://www.semanticweb.org/ontologies/pvz/knygos#universiteto_biblioteka;

v = universiteto_biblioteka;

v1 = biblioteka.

Taisyklė laikoma pažeista, kai klasės vardas nėra daiktavardinė frazė:

x = http://www.semanticweb.org/ontologies/pvz/knygos#raiškiai_skaitė;

v = raiškiai_skaitė;

v1 = skaitė.

Vienos klasės aprašymas ontologijoje nebus toks informatyvus apibūdinant aprašomą dalykinę sritį. Nurodžius klasių elementus svarbu aprašyti ir juos jungiančias savybes, pavyzdžiui: spalvą, formą, dydį. Kiekviena ontologijoje sukurta savybė turi būti priskirta klasei, kurią apibūdina. Ontologijos dalykinės srities sąvoką turi atspindėti nagrinėjamos srities objektus ir ryšius tarp jų. Remiantis gerosiomis praktikomis ontologijose aprašančiose tam tikrą dalykinę sritį daiktavardžiais apibūdinami objektai, o veiksmažodžiais ryšiai tarp objektų [18]. Iš to seka antroji ir trečioji ontologijos tikrinimo taisyklės, nurodančios savybių, naudojamų ontologijoje vardų sudarymo būdus:

Kiekvienos objektinės savybės (angl. *Object Property*) vardas turi atitikti struktūrą: *veiksmažodis* + *daiktavardis*.

$$\forall x \in X \ v \in V [objektinės_savybės_vardas(x, v) \Rightarrow \exists v1, v2 \in V [veiksmažodis(v1) \wedge daiktavardis(v2) \wedge objektinės_savybės_vardo_struktūra(v, v1, v2)].] \quad (2)$$

čia X – objektinių savybių aibė, V – žodžių aibė

Pavyzdžiui:

Taisyklė laikoma nepažeista, kai objektinės savybės vardas yra sudaromas naudojant struktūrą *veiksmazodis + daiktavardis*:

x = http://www.semanticweb.org/ontologies/pvz/knygos#priklauso_bibliotekai;
v = priklausos_bibliotekai;
v1 = priklausos;
v2 = bibliotekai.

Taisyklė laikoma pažeista, kai objektinės savybės vardas yra nesudaromas naudojant struktūrą *veiksmazodis + daiktavardis*:

x = http://www.semanticweb.org/ontologies/pvz/knygos#skaitovas_skaito;
v = skaitovas_skaito;
v1 = skaitovas;
v2 = skaito.

Kiekvienos duomenų savybės (angl. *Data Property*) vardas turi atitikti struktūrą: *turi + daiktavardis*.

$$\forall x \in X \forall v \in V [duomenų_savybės_vardas(x, v) \Rightarrow \exists v1, v2 \in V [turi(v1) \wedge daiktavardis(v2) \wedge duomenų_savybės_vardo_struktūra(v, v1, v2)]]. \quad (3)$$

čia X – objektinių savybių aibė, V – žodžių aibė

Pavyzdžiui:

Taisyklė laikoma nepažeista, kai duomenų savybės vardas yra sudaromas naudojant struktūrą *turi + daiktavardis*:

x = http://www.semanticweb.org/ontologies/pvz/knygos#turi_numerį;
v = turi_numerį;
v1 = turi;
v2 = numerį.

Taisyklė laikoma pažeista, kai duomenų savybės vardas yra nesudaromas naudojant struktūrą *veiksmazodis + daiktavardis*:

x = http://www.semanticweb.org/ontologies/pvz/knygos#priklauso_bibliotekai;
v = priklausos_bibliotekai;
v1 = priklausos;
v2 = bibliotekai.

Ontologijos savybių formavimo taisyklės

Klasės savybių detalesniam aprašymui yra naudojama sritis. O klasei, kuriai yra nurodyta savybė ar klasei, kurią apibūdina savybė yra naudojamas kitas ontologijos elementas – domenas. Svarbu paminėti, kad nėra būtina nurodyti savybių domeno, tačiau rekomenduojama [18]. Iš to galima formuoti taisykles:

Kiekvienai objektiniai savybei (angl. *Object Property*) turi būti specifikuotas domenas (angl. *Domain*) ir sritis (angl. *Range*).

$$\forall x \in X [objektinė_savybė(x) \Rightarrow \exists y, z \in Y [domenas(x, y) \wedge sritis(x, z)]]. \quad (4)$$

čia X – objektinių savybių aibė, Y – ontologijos klasių aibė.

Pavyzdžiui:

Taisyklė laikoma nepažeista, kai objektinei savybei yra specifikuotas domenai ir sritis:

```
<ObjectProperty rdf:about="http://www.semanticweb.org/ontologies/pvz/knygos#dirba_bibliotekoje">
  <inverseOf rdf:resource="http://www.semanticweb.org/ontologies/pvz/knygos#idarbina_darbuotoja"/>
  <rdf:type rdf:resource="http://www.w3.org/2002/07/owl#FunctionalProperty"/>
  <rdfs:domain rdf:resource="http://www.semanticweb.org/ontologies/pvz/knygos#Asmuo"/>
  <rdfs:range rdf:resource="http://www.semanticweb.org/ontologies/pvz/knygos#Biblioteka"/>
</ObjectProperty>
```

x = http://www.semanticweb.org/ontologies/pvz/knygos#dirba_bibliotekoje;
y = asmuo;
z = biblioteka.

Taisyklė laikoma pažeista, kai objektinei savybei nėra specifikuotas domenai ir sritis:

```
<ObjectProperty rdf:about="http://www.semanticweb.org/ontologies/pvz/knygos#dirba_bibliotekoje">
  <inverseOf rdf:resource="http://www.semanticweb.org/ontologies/pvz/knygos#idarbina_darbuotoja"/>
  <rdf:type rdf:resource="http://www.w3.org/2002/07/owl#FunctionalProperty"/>
</ObjectProperty>
```

x = http://www.semanticweb.org/ontologies/pvz/knygos#dirba_bibliotekoje;
y = nėra reikšmės;
z = nėra reikšmės.

Kiekvienai duomenų savybei (angl. *Datatype Property*) turi būti specifikuotas domenai (angl. *Domain*) ir sritis (angl. *Range*).

$\forall x \in X [duomenų_savybė(x) \Rightarrow \exists y, z \in Y [domenas(x, y) \wedge sritis(x, z)]]$. (5)
čia X – duomenų savybių aibė, Y – ontologijos klasių aibė

Pavyzdžiui:

Taisyklė laikoma nepažeista, kai duomenų savybei yra specifikuotas domenai ir sritis:

```
<DatatypeProperty rdf:about="http://www.semanticweb.org/ontologies/pvz/knygos#turi_gražinimo_data">
  <rdfs:domain rdf:resource="http://www.semanticweb.org/ontologies/pvz/knygos#Užsakymas"/>
  <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#date"/>
</DatatypeProperty>
```

x = http://www.semanticweb.org/ontologies/pvz/knygos#turi_gražinimo_data;
y = užsakymas;
z = date.

Taisyklė laikoma pažeista, kai duomenų savybei nėra specifikuotas domenai ir sritis:

```
<DatatypeProperty rdf:about="http://www.semanticweb.org/ontologies/pvz/knygos#turi_gražinimo_data">
</DatatypeProperty>
```

x = http://www.semanticweb.org/ontologies/pvz/knygos#turi_gražinimo_data;
y = nėra reikšmės;
z = nėra reikšmės.

Ontologijos anotacijų formavimo taisyklės

Anotacijos ontologijoje, iš vartotojo pusės, pagerina dalykinės srities supratimą ir panaudojimo galimybes. Todėl siekiant sukurti struktūriškai taisyklingą ontologiją buvo išskirtos trys anotacijos:

- *label* anotacija, papildo ontologijos elementų aprašymus, padeda suprasti elementų prasmę;
- *label* kalbos anotacija, nurodo ontologijoje naudojamų elementų kalbą, tokiu būdu palengvina ontologijos elementų morfologinę analizę;
- *label_sbvr* anotacija, reikalinga siekiant atlikti ontologijos transformaciją į *SBVR* modelį.

Kiekvienas ontologijos elementas turi turėti *label* anotaciją.

$$\forall x \in X [esybe(x) \Rightarrow \exists y \in Y [label(x, y)].$$

čia X – ontologijos elementų aibė, Y – label anotacijų aibė

(6)

Pavyzdžiui:

Taisyklė laikoma nepažeista, kai ontologijos klasių ir savybių elementams yra specifikuota *label* anotacija:

```
<Class rdf:about="http://www.semanticweb.org/ontologies/pvz/knygos#Biblioteka">
  <rdfs:comment xml:lang="lt">Biblioteka, kurioje yra saugomos knygos</rdfs:comment>
  <rdfs:label xml:lang="en">Library</rdfs:label>
  <rdfs:label xml:lang="lt">Biblioteka</rdfs:label>
  <ann:label_lt xml:lang="lt">Biblioteka</ann:label_lt>
  <ann:label_sbvr xml:lang="lt">Biblioteka</ann:label_sbvr>
</Class>
```

Kiekviena ontologijos elemento *label* anotacija turi turėti kalbą.

$$\forall x \in X y \in Y [label(x, y) \Rightarrow \exists z \in Z [kalba(y, z)].$$

čia X – ontologijos elementų aibė, Y – label anotacijų aibė, Z – kalbų aibė

(7)

Taisyklė laikoma nepažeista, kai ontologijos klasių ir savybių elementams yra specifikuota *label* anotacijos kalba:

```
<Class rdf:about="http://www.semanticweb.org/ontologies/pvz/knygos#Biblioteka">
  <rdfs:comment xml:lang="lt">Biblioteka, kurioje yra saugomos knygos</rdfs:comment>
  <rdfs:label xml:lang="en">Library</rdfs:label>
  <rdfs:label xml:lang="lt">Biblioteka</rdfs:label>
  <ann:label_lt xml:lang="lt">Biblioteka</ann:label_lt>
  <ann:label_sbvr xml:lang="lt">Biblioteka</ann:label_sbvr>
</Class>
```

Kiekvienoje objektinėje savybėje esanti anotacija *label_sbvr*, turi būti tokios struktūros: srityje esančios klasės vardas, kuris įrašytas tos klasės *label_sbvr* anotacijoje + veiksmoždis + domene esantis klasės vardas, kuris įrašytas tos klasės *label_sbvr* anotacijoje.

$$\forall x \in X; v, v1, v3 \in V y, z \in Y [objektinės_savybės_label_sbvr(x, v) \wedge \text{domenas}(x, y) \wedge klases_label_sbvr(y, v1) \wedge sritis(x, z) \wedge klases_label_sbvr(x, v3)] \Rightarrow \exists v2 \in V [veiksmoždis(v2) \wedge objektinės_savybės_label_sbvr_struktūra(v, v1, v2, v3)].$$

čia X – objektinių savybių aibė, V – žodžių aibė, Y – ontologijos klasių aibė

Pavyzdžiui:

Taisyklė laikoma nepažeista, kai ontologijos objektinių savybių elementams yra specifikuota *label_sbvr* anotacijos, o jos sudarymui naudojama struktūra *domenas* + *veiksmoždis* + *sritis*:

```

<owl:ObjectProperty rdf:about="http://semantika.lt/ns/Politics#klausytojui_kalbėjo_kalbėjimą">
  <rdfs:label xml:lang="lt">klausytojui kalbėjo kalbėjimą</rdfs:label>
  <ann:label sbvr xml:lang="lt">klausytojui kalbėjo kalbėjimą</ann:label_sbvr>
  <ann:label_lt xml:lang="lt">klausytojui kalbėjo kalbėjimą</ann:label_lt>
  <rdfs:domain rdf:resource="&agents;klausytojas"/>
  <owl:inverseOf rdf:resource="http://semantika.lt/ns/Politics#kalbėjimą_kalbėjo_klausytojui"/>
  <rdfs:range rdf:resource="http://semantika.lt/ns/Politics#kalbėjimas"/>
</owl:ObjectProperty>

```

$x = \text{http://semantika.lt/ns/Politics\#klausytojui_kalbėjo_kalbėjimą};$
 $v1 = \text{klausytojas};$
 $v2 = \text{kalbėjo};$
 $v3 = \text{kalbėjimas}.$

Ontologijos elementų rašymo stiliaus formavimo taisyklės

Dalykinės srities, kurią aprašo ontologijos, suprantamumui pagilinti ir modeliavimo kokybei pagerinti svarbu ontologijos vardu užrašymo stiliaus vientisumas. Šiam tikslui įgyvendinti buvo apsibrėžtos ontologijos stiliaus tikrinimo taisyklės, kurios nusako, kad:

Ontologijos elementų vardai, kuriuos sudaro keli žodžiai, turi būti užrašyti pagal vieną iš pateiktų stilių:

- vardą sudarančių žodžių atskyrimui turi būti naudojamas pabraukimo simbolis apačioje pvz.: *gyvatės_stilius* (angl. *Snake case*)
- vardą sudarančių žodžių atskyrimui turi būti naudojamos didžiosios ir mažosios raidės pvz.: *kupranugarioStilius* (angl. *Camel case*)

$$\forall x \in X \forall v \in V [\text{vardas}(x,y)] \Rightarrow \text{snake_case}(v) \vee \text{camel_case}(v)] \quad (9)$$

čia X – ontologijos elementų aibė, V – žodžių aibė

Pavyzdžiui:

Taisyklė laikoma nepažeista, kai ontologijos elementų vardai yra rašomi vieningu stiliumi t. y. gyvatės arba kupranugario, pavyzdyje, kuris yra pateikiamas žemiau elementų vardai sujungiami laikantis gyvatės stiliaus jungimo principų:

```

<ObjectProperty rdf:about="http://www.semanticweb.org/ontologies/pvz/knygos#užsako_universiteto_bibliotekos_knyga">
  <rdf:type rdf:resource="http://www.w3.org/2002/07/owl#FunctionalProperty"/>
  <rdfs:domain rdf:resource="http://www.semanticweb.org/ontologies/pvz/knygos#Užsakymas"/>
  <rdfs:range rdf:resource="http://www.semanticweb.org/ontologies/pvz/knygos#Universiteto_bibliotekos_knyga">
</ObjectProperty>

```

Taisyklė laikoma pažeista, kai ontologijos elementų vardai yra rašomi nevieningu stiliumi, pateiktame pavyzdyje vienas iš ontologijos elementų vardu užrašytas laikantis gyvatės stiliaus (pabraukta žalia spalva), kitas – kupranugario (pabraukta raudona spalva):

```

<ObjectProperty rdf:about="http://www.semanticweb.org/ontologies/pvz/knygos#užsako_universiteto_bibliotekos_knyga">
  <rdf:type rdf:resource="http://www.w3.org/2002/07/owl#FunctionalProperty"/>
  <rdfs:domain rdf:resource="http://www.semanticweb.org/ontologies/pvz/knygos#Užsakymas"/>
  <rdfs:range rdf:resource="http://www.semanticweb.org/ontologies/pvz/knygos#universitetoBibliotekosKnyga"/>
</ObjectProperty>

```

Ontologijų verifikavimo įrankis tikrins įkeltą ontologiją, pagal ontologijos kūrimui aprašytas taisykles. Siekiant išsaugoti verifikavimo metu aptiktus taisyklių pažeidimų sąrašą ir jį atvaizduoti ontologijų redagavimo įrankyje buvo sukurtas išpėjimų ir klaidų anotacijų ontologija.

Taisyklių pažeidimų tipų sąrašas užtikrina rastų išpėjimų ir klaidų atvaizdavimą ontologijos redagavimo sistemoje *Protégé*. Šiame įrankyje jos atvaizduojamos kaip anotacijos, todėl nedaro įtakos dalykinės srities aprašymui. Rezultato generavimui verifikavimo įrankis importuos tikrintą ontologiją ir anotacijų failą. Išpėjimų ir klaidų anotacijos ontologijos elementai, jų aprašymai ir ontologijos tikrinimo taisyklių reprezentavimas pateikiamas 2.1 lentelėje, o visą išpėjimų ir klaidų anotacijų ontologiją galima rasti 8.1 skyriuje.

2.1 lentelė. Įspėjimų ir klaidų anotacijų ontologijos struktūra.

| Įspėjimų ir klaidų anotacijų ontologijos elementai | Apibūdinimas | Ontologijos tikrinimo taisyklė |
|--|--|--------------------------------|
| warning_class_not_noun | Įspėjamoji anotacija, kad klasės elementas prieštarauja taisyklei t. y. ontologijos klasė turi būti <i>daiktavardis</i> arba <i>daiktavardinė frazė</i> . | 4 |
| error_class_without_name | Klaidos anotacija, kuri praneša, kad klasės elementas neturi vardo. | 4 |
| warning_data_no_child_elements | Įspėjamoji anotacija pranešanti, kad duomenų savybė nėra pilnai išbaigta t. y. neturi anotacijos, domeno ir srities, nenurodyta kalba. | 2 |
| warning_data_not_have/has/had_plus_noun | Įspėjamoji anotacija, informuojanti, kad duomenų savybė neatitinka struktūros <i>turi (have/has/had) + daiktavardis</i> . | 6 |
| warning_data_without_attribute_about | Įspėjamoji anotacija, informuojanti, kad duomenų savybė yra aprašyta be atributo <i>about</i> , tačiau atributas gali būti apkeistas kitu atributu pvz. ID, tačiau būna situacijų kai dėl įvairių modifikacijų elementas praranda pagrindinį atributą, todėl anotacija yra užuomina patikrinti duomenų savybę, kurioje nėra <i>about</i> atributo. | 2 |
| warning_data_without_domain | Įspėjamoji anotacija, kad duomenų savybė neturi domeno. | 2 |
| error_data_without_name | Klaidos anotacija, informuojanti, kad duomenų savybė yra nurodyta, tačiau neturi vardo. | 2 |
| warning_data_without_range | Įspėjamoji anotacija, informuojanti, kad duomenų savybė neturi srities. | 2 |
| warning_data_without_resource | Įspėjamoji anotacija informuojanti, kad duomenų savybė nėra išbaigta t. y. nėra nurodyta pagrindinės ontologijos URI adresas (esantis <i>resource</i> atribute). | 2 |
| warning_label_is_missing | Įspėjamoji anotacija informuojanti, kad klasėje/duomenų savybėse/objektinėse savybėse nėra nurodyta <i>label</i> anotacija . | 7 |
| warning_label_is_missing_selected_language | Tikrinimo metu, vartotojo nurodyta kalba, nebuvo rasta nei viena kalbos anotacija. | 8 |
| warning_label_sbvr_is_missing | Įspėjimas, informuojantis vartotoją apie, tai, kad rastas ontologijos elementas neturi <i>label_sbvr</i> anotacijos. | 9 |
| warning_label_sbvr_strukture_is_not_correct | Įspėjimas, informuojantis vartotoją apie, tai, kad ontologijos objektinių savybių anotacija <i>label_sbvr</i> buvo suformuota nesilaikant taisyklės t. y. <i>objektinės savybės domenai + veiksmazodis + objektinės savybės sritis</i> . | 9 |
| warning_object_no_child_elements | Įspėjamoji anotacija pranešanti, kad objektinė savybė nėra pilnai išbaigta t. y. neturi anotacijos, domeno ir srities, nenurodyta kalba. | 1 |

| | | |
|--|--|---|
| warning_object_not_verb_plus_noun | Įspėjamoji anotacija, informuojanti, kad duomenų savybė neatitinka struktūros <i>veiksmazodis + daiktavardis</i> . | 5 |
| warning_object_without_attribute_about | Įspėjamoji anotacija, informuojanti, kad objektinė savybė yra aprašyta be atributo <i>about</i> , tačiau atributas gali būti pakeistas kitu atributu pvz. <i>ID</i> , tačiau būna situacijų kai dėl įvairių ontologijos modifikacijų elementas praranda pagrindinį atributą, todėl anotacija yra užuomina patikrinti duomenų savybę, kurioje nėra <i>about</i> atributo. | 1 |
| warning_object_without_domain | Įspėjamoji anotacija, kad objektinė savybė neturi domeno. | 1 |
| error_object_without_name | Klaidos anotacija, informuojanti, kad objektinė savybė yra nurodyta, tačiau neturi vardo. | 1 |
| warning_object_without_range | Įspėjamoji anotacija, kad objektinė savybė neturi srities. | 1 |
| warning_object_without_resource | Įspėjamoji anotacija informuojanti, kad objektinė savybė nėra išbaigta t. y. nėra nurodyta pagrindinės ontologijos <i>URI</i> adresas (esantis <i>resource</i> atribute). | 1 |
| warning_style_not_camel_case | Tikrinamos ontologijos elementų vardai yra parašyti nesilaikant kupranugario stiliaus žodžių jungimo principo. | 3 |
| warning_style_not_snake_case | Tikrinamos ontologijos elementų vardai yra parašyti nesilaikant gyvatės stiliaus žodžių jungimo principo. | 3 |

2.4. Reikalavimų apibendrinimas

Šiame skyriuje buvo aptarti funkciniai ir nefunkciniai reikalavimai, išskirtos ir formaliai aprašytos devynios taisyklės, kuriomis remiantis galima patikrinti ontologijos struktūrą. Pagrindinės sistemos funkcijos: ontologijos įkėlimas, ontologijos tikrinimas peržiūrint ir ontologijos tikrinimas išsaugant.

Ontologijų verifikavimo įrankio dalykinės srities klasių modelis buvo pateikiamas rezultatų failo formavimui pasirinkus tikrinamos ontologijos ir įspėjimų ir klaidų anotacijų ontologijos failų importavimą (2.1 paveikslas). Lentelėje 2.1 buvo pateikti įspėjimų ir klaidų anotacijų ontologijos, ir jų aprašymai, tikrinama ontologija priklausys nuo to, kokią ontologiją pasirinko vartotojas.

Išskirtos devynios formalios ontologijų tikrinimo taisyklės, kuriomis remiantis galima sukurti struktūriškai teisingą ontologiją, šiomis taisyklėmis remiasi ir ontologijų verifikavimo įrankis.

Šiame skyriuje buvo specifiukuota pagrindinė ontologijų verifikavimo įrankio idėja – leisti įkelti, patikrinti ir išsaugoti ontologiją, jos tikrinimui pritaikant ontologijų tikrinimo taisykles.

3. ONTOLOGIJŲ VERIFIKAVIMO ĮRANKIO REALIZACIJOS PROJEKTAS

Siekiant sukurti funkcionalų ir tikrinimo taisyklėmis paremtą ontologijų verifikavimo įrankį buvo atliktas detalus įrankio projektavimas. Šiame skyriuje atlikta reikalavimų analizė, nubraižyta robastiškumo diagrama, sukurta loginė įrankio architektūra, specifikuotos sekų diagramos, pateiktas ontologijų verifikavimo algoritmas ir diegimo diagrama.

3.1. Sprendimo pagrindimas ir esminiai principai

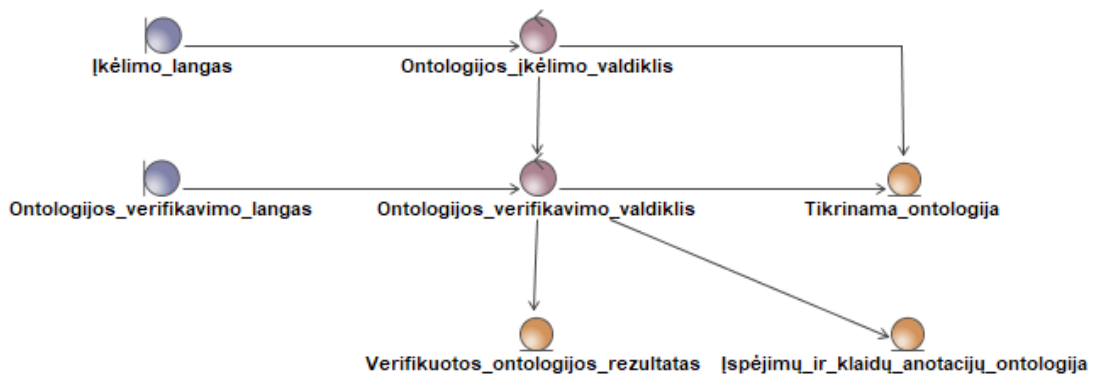
Ontologijų verifikavimo įrankis yra ontologijos kūrimo metodikos dalis. Šis įrankis tikrina ontologiją remdamasis projektavimo metu išskirtomis ontologijų tikrinimo taisyklėmis, išsaugotas rezultatas naudojamas dažniausiai pasitaikančių klaidų analizei ir tolimesniam ontologijos redagavimui.

3.2. Ontologijų verifikavimo įrankio architektūra

Reikalavimų analizė

Pagrindinis projekto tikslas yra suprojektuoti ontologijų verifikavimo sistemą panaudojant *CASE* įrankius. Apsibrėžti vartotojo sąsajos pagrindinį langą, nustatyti programinių langų skaičių, realizuoti sistemos logiką ir diegimo modelį.

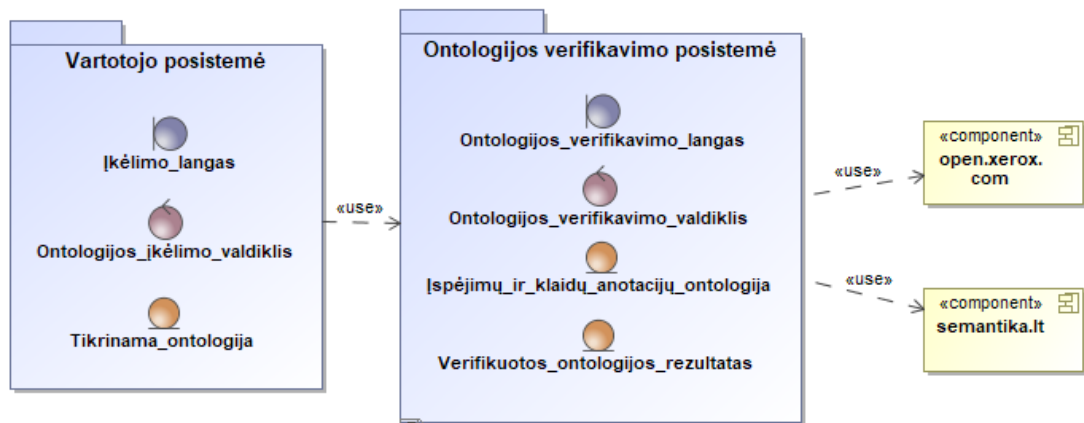
Robastiškumo diagrama padeda atskleisti kokios pagrindinės vartotojo sąsajos turi būti įgyvendintos, kiek ir kokie turi būti valdikliai užtikrinantys programos veikimo logiką, analizės diagramos parodo kokie turi būti įėjimo ir išėjimo failai. Robastiškumo diagrama pateikiama 3.1 paveiksle.



3.1 pav. Robastiškumo diagrama

Loginė sistemos architektūra

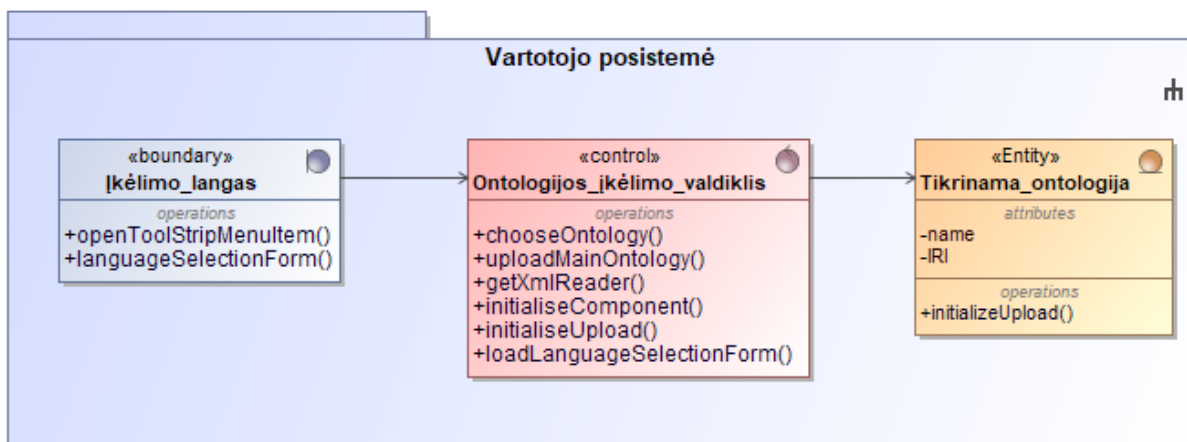
Loginė sistemos architektūra susideda iš dviejų pagrindinių paslaugų t. y. vartotojo posistemės ir ontologijų verifikavimo posistemės, kuri papildomai iškviečia tinklo paslaugą morfologinei analizei atlikti. Ontologijų verifikavimo sistemos loginė architektūra pateikiama 3.2 paveiksle.



3.2 pav. Loginė sistemos architektūra

Vartotojo posistemė

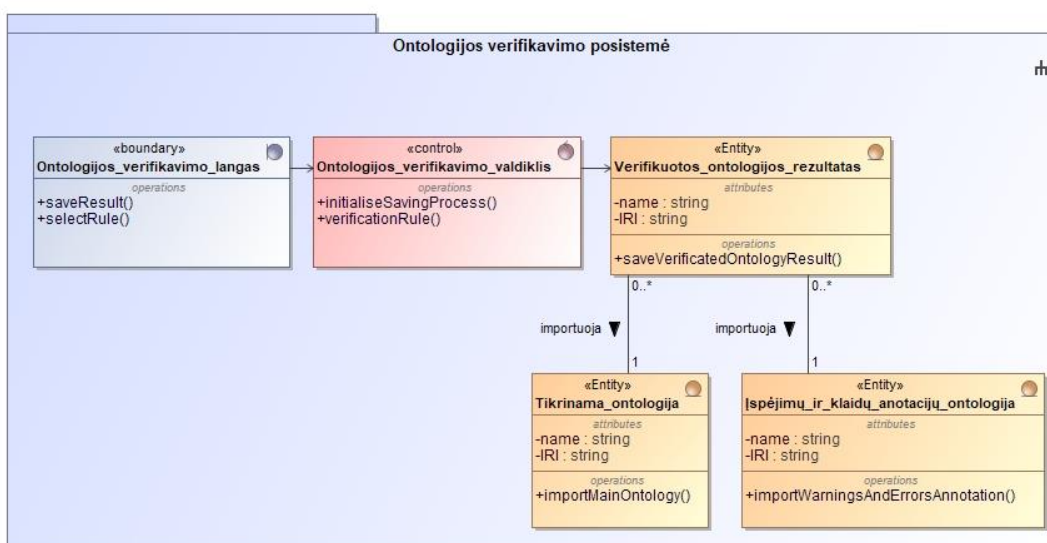
Atsidarius ontologijų verifikavimo sistemą vartotojas pirmiausia patenka į vartotojo posistemę, iš kurios mato įkėlimo langą ir gali įkelti naują ontologiją, ontologijos įkėlimo valdiklis kontroliuoja įkėlimo procesą ir galiausiai vartotojas pagrindiniame įrankio lange gali matyti tikrinamą ontologiją. Vartotojo posistemės schema pateikiama 3.3 paveiksle.



3.3 pav. Vartotojo posistemė

Ontologijų verifikavimo posistemė

Detalus verifikavimo sistemos posistemės modelis vaizduoja naudojamas klases ir metodus. Vartotojui įkėlus ontologiją įrankis galės pradėti ontologijos tikrinio procesą vos tik vartotojas pasirinks elementus, kuriuos nori patikrinti. Ontologijos verifikavimo posistemės veikimo modelis yra pateiktas 3.4 paveiksle.

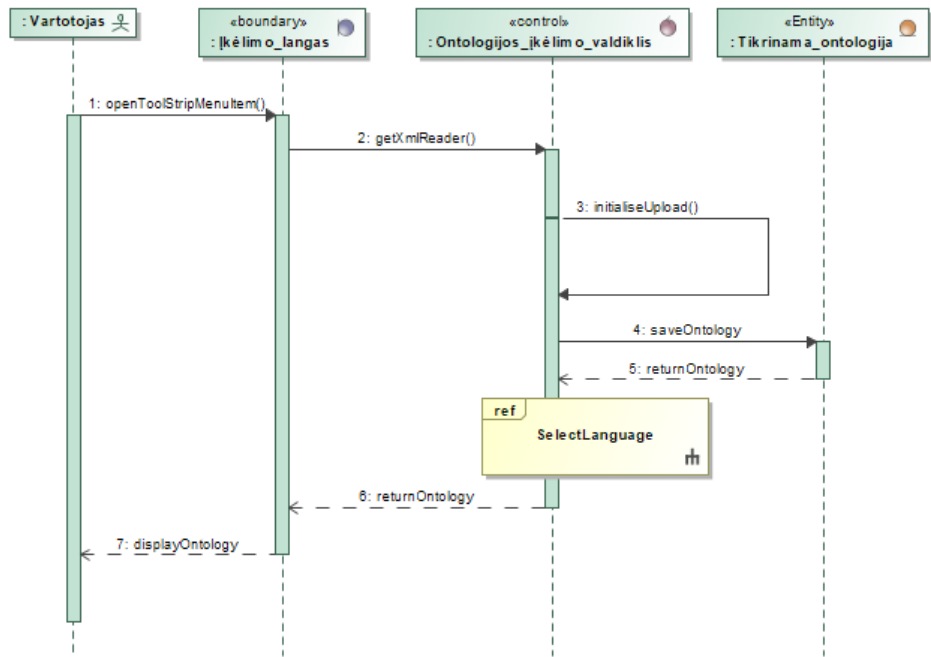


3.4 pav. Ontologijos verifikavimo posistemė

3.3. Sistemos elgsenos modelis

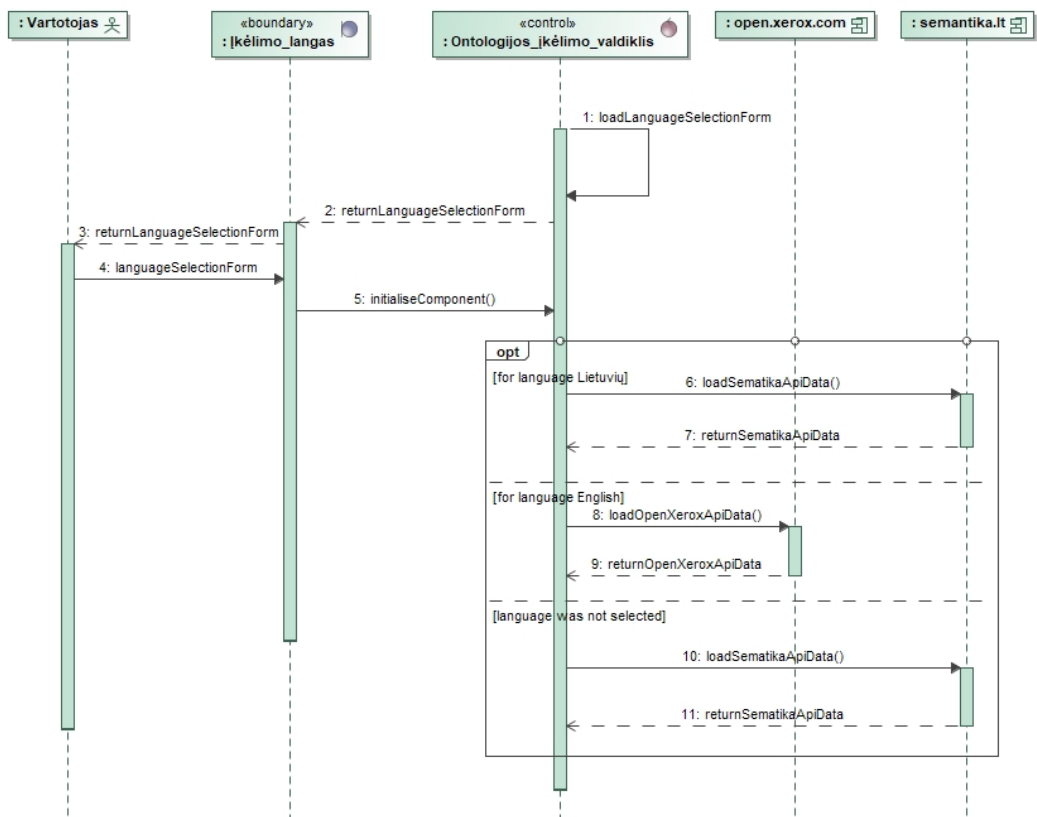
Šiame skyriuje pateikiamos sekų diagramos, kurios parodo sistemos veikimą tarp sistemos langų, valdiklių ir klasių. Diagramose parodyti metodai, kurie manipuliuoja gautais duomenimis ir užtikrina galutinį rezultatą, kurio buvo tikimasi.

Panaudojimo atvejo *įkelti ontologiją* sekų diagramos vaizduojamos 3.5 ir 3.6 paveiksluose. Pirmojoje diagramoje vaizduojamas ontologijos įkėlimas į kurį įsiterpia ir kalbos pasirinkimo sekų diagrama (Pav. 3.5).



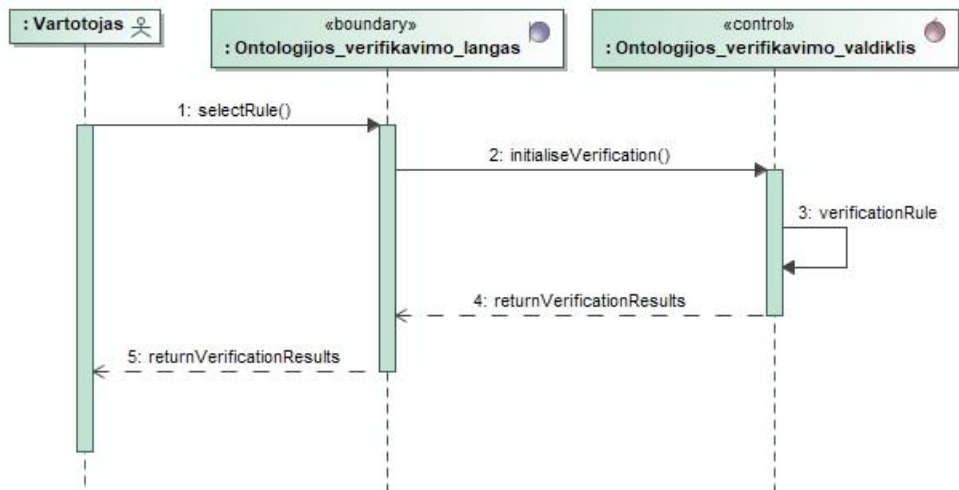
3.5 pav. Panaudojimo atvejo *įkelti ontologiją* sekų diagrama

Įrankiui iniciavus įkėlimo procesą vartotojo prašoma nurodyti kalbą, kuria yra aprašyta įkeliama ontologija. Įvykdžius pasirinkimą pradedamas tinklo paslaugos užrovimo procesas, detali sekų diagrama atvaizduojama 3.6 paveiksle.



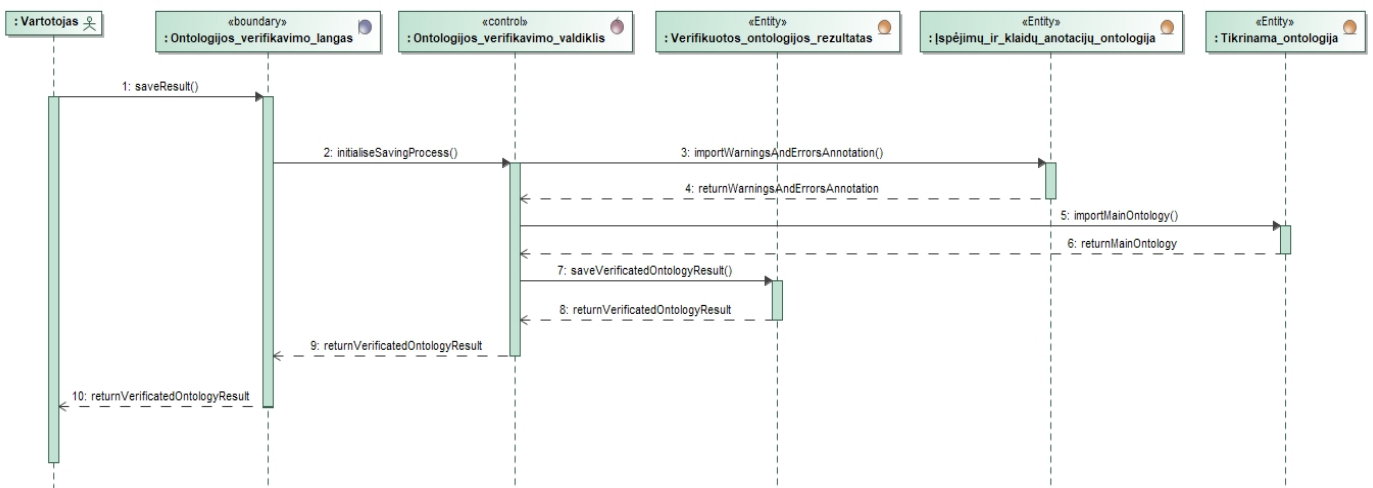
3.6 pav. Kalbos pasirinkimo sekų diagrama

Panaudojimo atvejo *tikrinti peržiūrint* sekų diagrama atvaizduojama 3.7 paveiksle.



3.7 pav. Panaudojimo atvejo *tikrinti peržiūrint* sekų diagrama

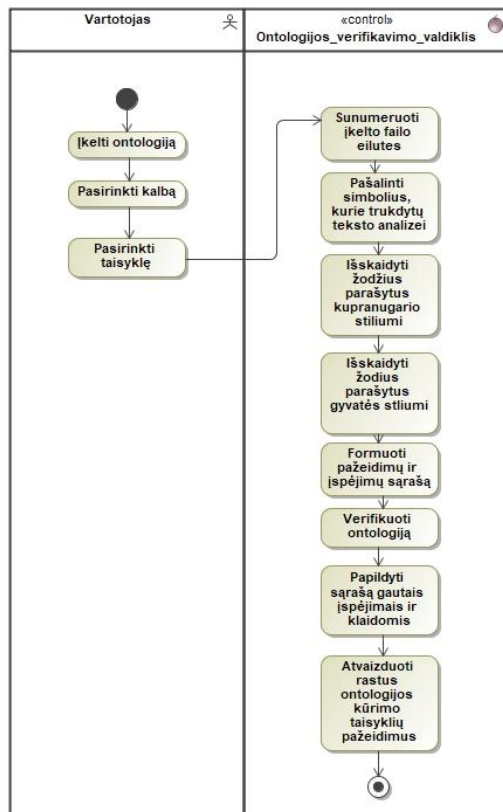
Panaudojimo atvejo *tikrinti išsaugant* sekų diagrama atvaizduojama 3.8 paveiksle.



3.8 pav. Panaudojimo atvejo *tikrinti išsaugant* sekų diagrama

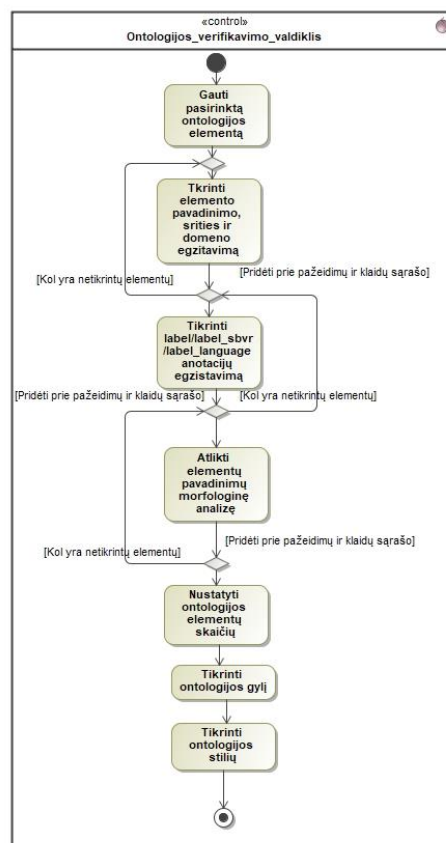
Verifikavimo įrankio pagrindė funkcija yra ontologijos tikrinimas remiantis ontologijos tikrinimo taisyklėmis. Veiklos diagramos, kuriuose vaizduojamas algoritmo veikimo principas atvaizduojamos 3.9 ir 3.10 paveiksluose.

Pirmojoje diagramoje, kuri yra pavaizduota 3.9 paveiksle, parodytas vartotojo ir ontologijos verifikavimo valdiklio sąveika.



3.9 pav. Ontologijos verifikavimo proceso diagrama

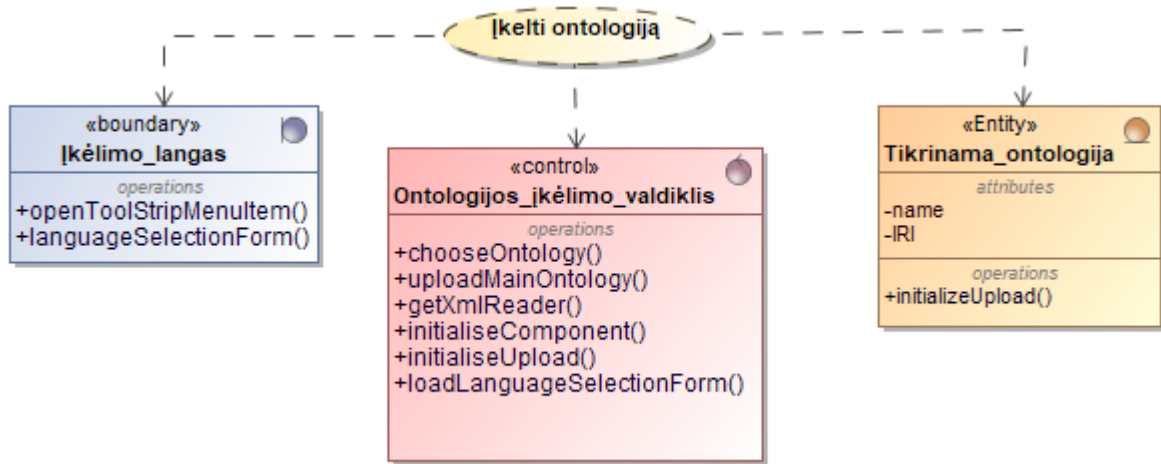
Antroji diagrama parodo vidinius verifikatoriaus veiksmus, kai vartotojas įkelia ontologiją ir pasirenka kalbą. (Pav. 3.10)



3.10 pav. Detalaus ontologijos verifikavimo proceso diagrama

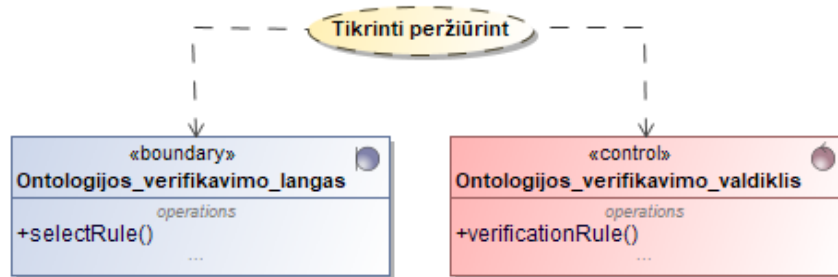
3.4. Detalus projektavimas

Panaudojimo atvejai yra įgyvendinami juos kompiuterizuojant. Panaudojimo atvejo *įkelti ontologiją* realizacijos diagrama yra pateikiama 3.11 paveiksle.



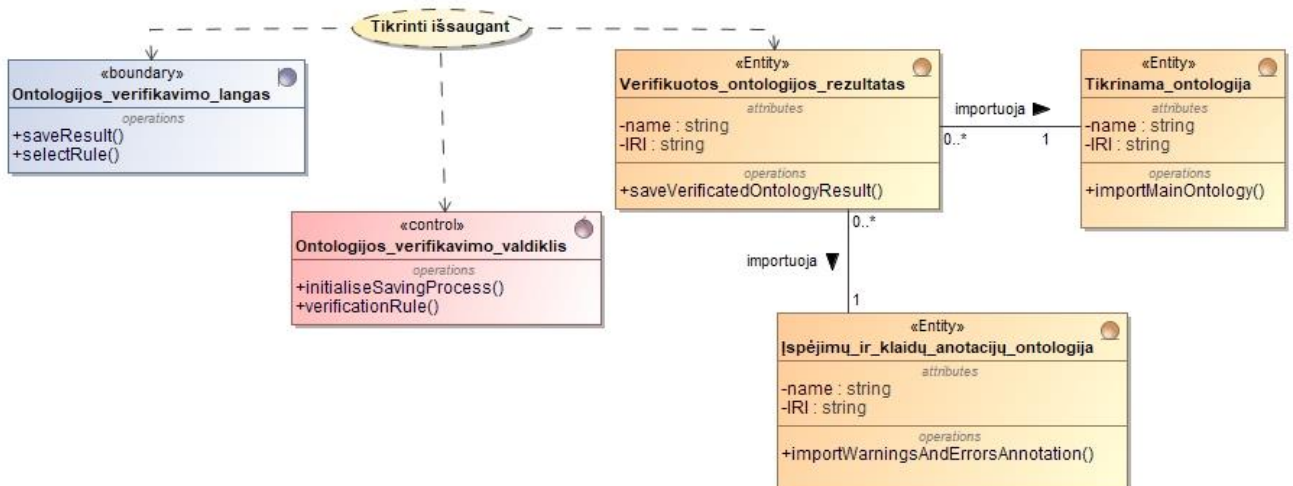
3.11 pav. Panaudojimo atvejo *įkelti ontologiją* realizacijos klasių diagrama

Panaudojimo atvejo *tikrinti peržiūrint* realizacijos diagrama yra pateikiama 3.12 paveiksle.



3.12 pav. Panaudojimo atvejo *tikrinti peržiūrint* realizacijos klasių diagrama

Panaudojimo atvejo *tikrinti išsaugant* realizacijos diagrama yra pateikiama 3.13 paveiksle.



3.13 pav. Panaudojimo atvejo *tikrinti išsaugant* realizacijos klasių diagrama

Valdikliuose naudotų metodų aprašymas ir jų operacijos yra pateikiamos 3.1 ir 3.2 lentelėse. Ontologijos verifikavimo valdiklio operacijos pateikiamos 3.1 lentelėje.

3.1 lentelė. Ontologijos verifikavimo valdiklio operacijos

| Metodas | Paskirtis |
|---------------------------|--|
| initialiseSavingProcess() | Išsaugoti rezultatų failą |
| verificationRule() | Verifikuoti ontologiją pasirinkta taisykle |

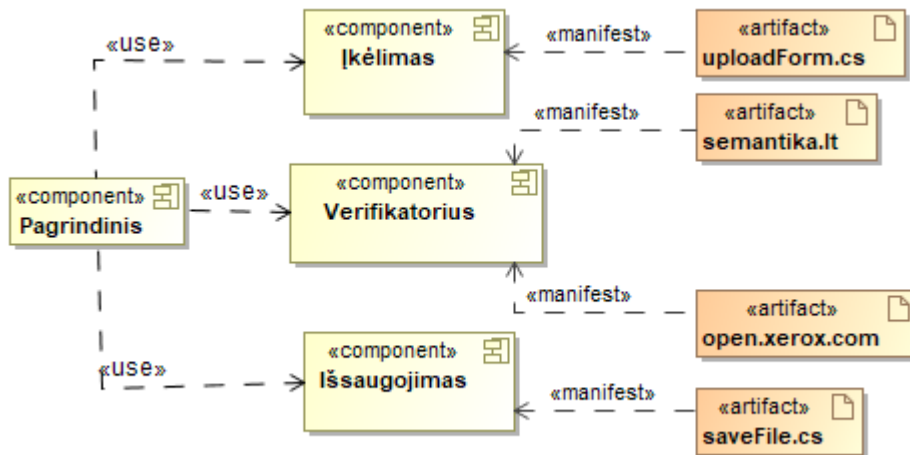
Ontologijos įkėlimo valdiklio naudojamos operacijos pateikiamos 3.2 lentelėje.

3.2 lentelė. Ontologijos įkėlimo valdiklio operacijos

| Metodas | Paskirtis |
|-----------------------------|---|
| chooseOntology() | Pasirinkti norimą įkelti ontologiją |
| uploadMainOntology() | Įkelti tikrinamą ontologiją |
| getXmlReader() | Iškvieti XML failo nuskaitymo metodą |
| initialiseUpload() | Pradėti įkėlimo procesą |
| initialiseComponent() | Inicijuoti kalbos analizės paslaugos iškvietimą iš tinklo |
| loadLanguageSelectionForm() | Užkrauti kalbos pasirinkimo langą |

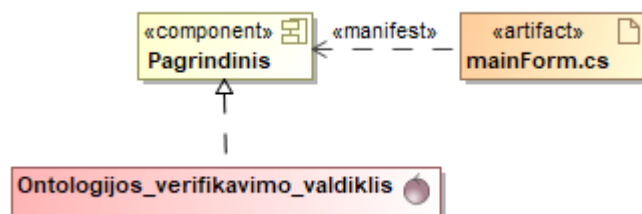
3.5. Programinių komponentų architektūra

Komponentų, komponentų realizavimo ir diegimo diagramos sudaro realizacijos modelį. Komponentų diagrama parodo iš kokių komponentų susideda ontologijų verifikavimo įrankis, diagrama pavaizduota 3.14 paveiksle.



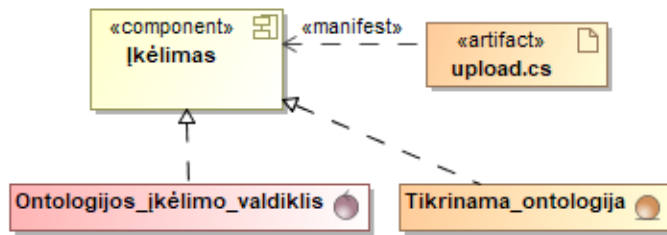
3.14 pav. Ontologijų verifikavimo sistemos komponentų diagrama

Toliau vaizduojamos komponentų realizavimo diagramos parodo detalias, kurios sudaro komponentus. Komponento *pagrindinis* diagrama vaizduojama 3.15 paveiksle.



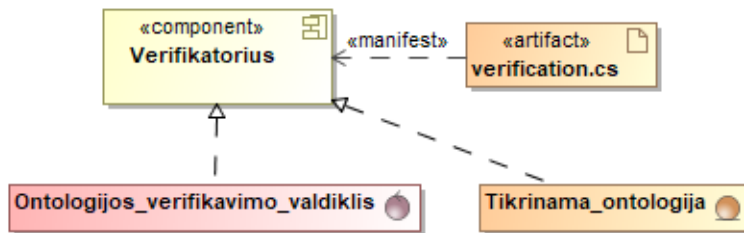
3.15 pav. Komponento *pagrindinis* diagrama

Komponento *įkėlimas* diagrama pavaizduota 3.16 paveiksle.



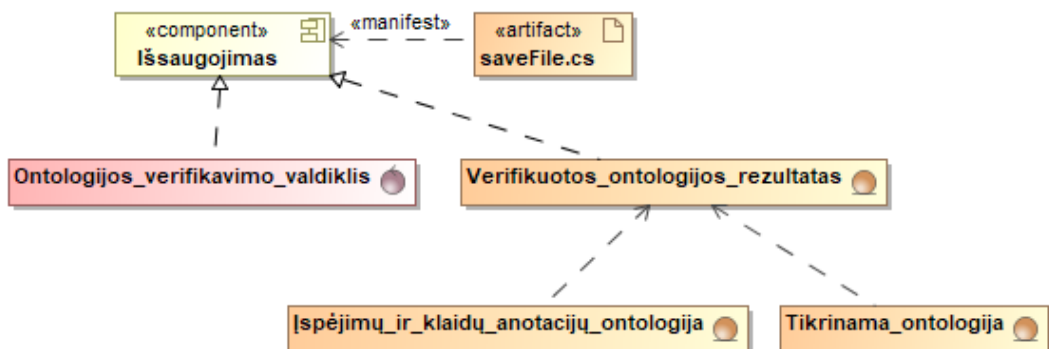
3.16 pav. Komponento *įkėlimas* diagrama

Komponento *verifikatorius* diagrama pateikiama 3.17 paveiksle.



3.17 pav. Komponento *verifikatorius* diagrama

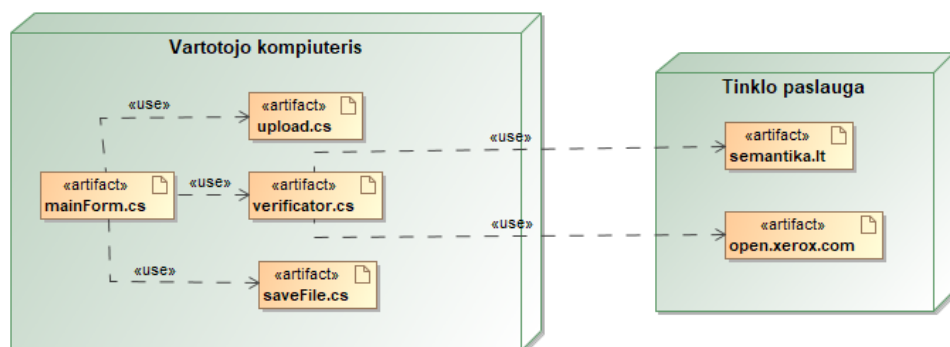
Komponento *išsaugojimas* diagrama atvaizduojama 3.18 paveiksle.



3.18 pav. Komponento *išsaugojimas* diagrama

Diegimo modelis

Ontologijų verifikavimo įrankio diegimo modelis pateiktas 3.19 paveiksle.



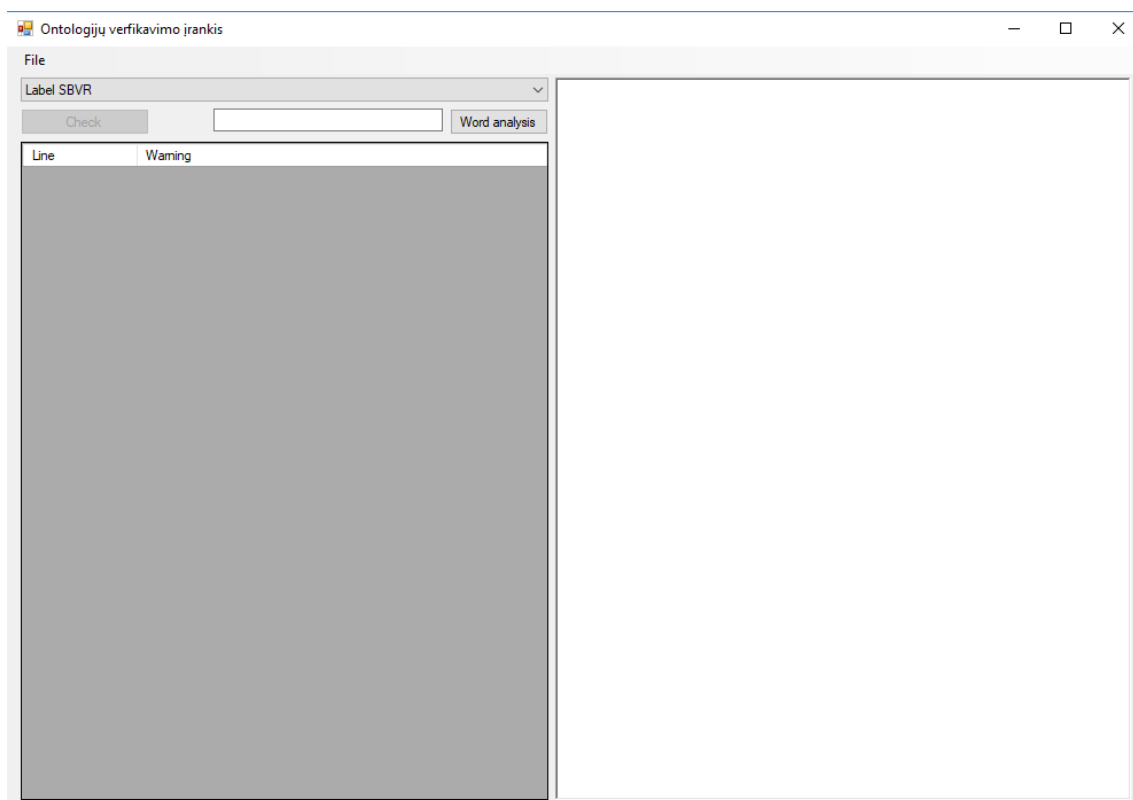
3.19 pav. Ontologijų verifikavimo sistemos diegimo modelis

4. ONTOLOGIJOS VERIFIKAVIMO ĮRANKIO REALIZACIJA IR TESTAVIMAS

Šiame skyriuje pateikiama ontologijų verifikavimo įrankio realizacija ir pristatomas jo veikimas. Įrankis buvo testuojamas pasirinkus *politikos ontologiją*, testavimo metu patikrintas įrankio veikimas, apibendrinti gauti rezultatai.

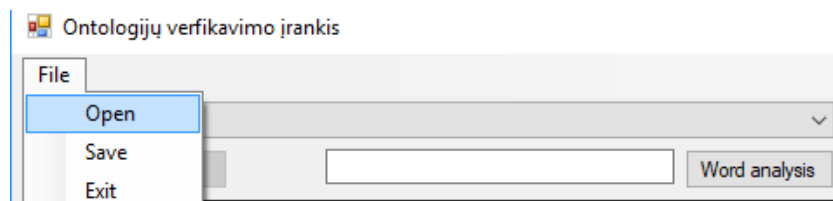
4.1. Sprendimo realizacijos ir veikimo aprašas

Ontologijų verifikavimo sistemos įgyvendinimui buvo sukurta *Windows* forma, kurią vartotojas mato vos pradėjęs programos vykdymą (Pav. 4.1).



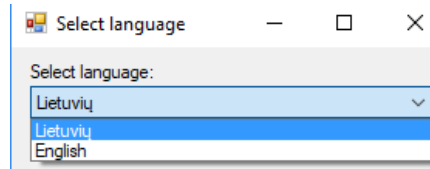
4.1 pav. Pradinis ontologijų verifikatoriaus langas

Vartotojas norėdamas tikrinti sukurtą ontologijos failą, jį *.owl/.rdf* formatu įkelia į ontologijų verifikavimo sistemą. Ontologija yra įkeliamą naudojant mygtukus *File - > Open*, esančius kairiajame aplikacijos kampe (pav. 4.2).



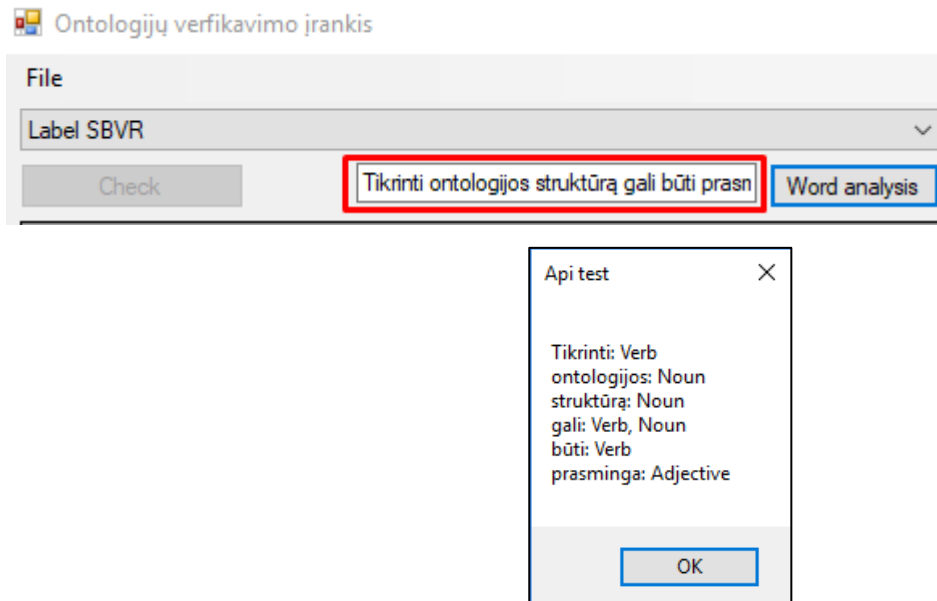
4.2 pav. Ontologijos įkėlimo inicijavimas

Vos įkėlus ontologiją vartotojui pasirodo langas, kuriame prašoma pasirinkti kalbą, tokią kuria ir buvo kuriama ontologija. Ontologijų verifikavimo įrankis leidžia pasirinkti vieną iš dviejų kalbų t. y. anglų ir lietuvių. Kalbos pasirinkimo langas pavaizduotas 4.3 paveiksle.



4.3 pav. Kalbos pasirinkimas

Žodžių morfologinei analizei patikrinti paruoštas specialus laukas, kuriame įvedus žodį/frazę lietuvių kalba galima nustatyti ar tai daiktavardis (angl. *Noun*), veiksmažodis (angl. *Verb*), būdvardis (angl. *Adjective*). Morfologinės žodžių analizės pavyzdys pateikiamas 4.4 paveiksle.



4.4 pav. Morfologinės analizės rezultatų atvaizdavimas

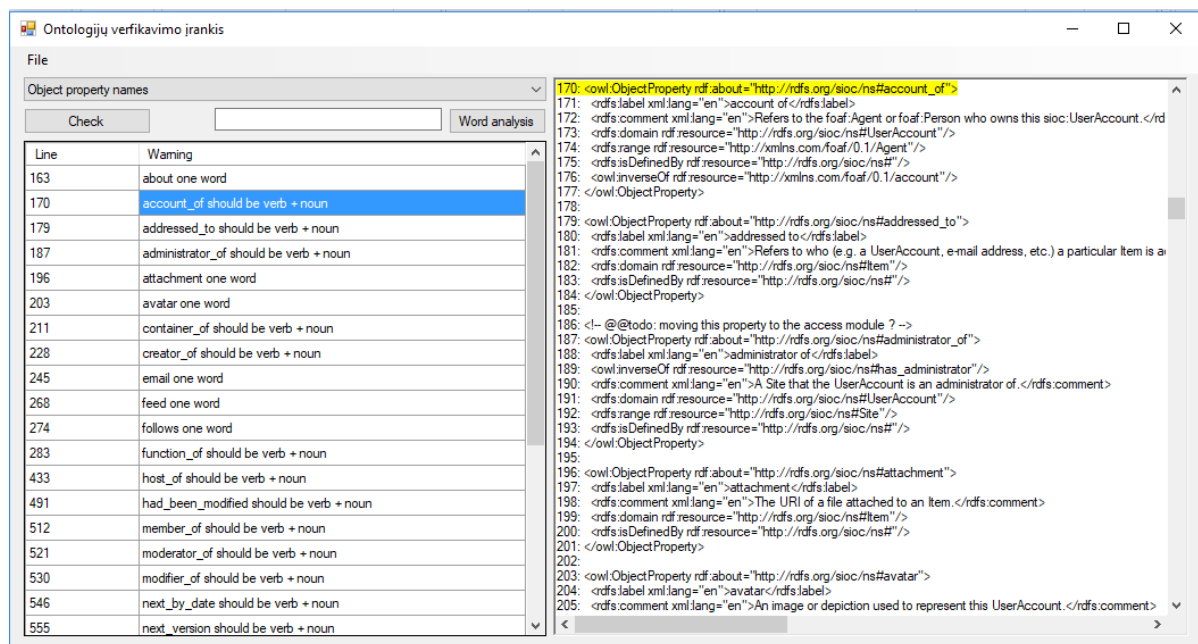
Įkėlus ontologiją į ontologijų verifikavimo įrankį vartotojas gali pasirinkti kuriuos ontologijos elementus norima patikrinti. Ontologijos elementų tikrinimas vykdomas remiantis ontologijų tikrinimo taisyklėmis, kurios buvo išskirtos įrankio projektavimo metu. Ontologijų verifikavimo įrankis leidžia pasirinkti šiuos ontologijos elementų tikrinimo punktus žr. lentelė 4.1.

4.1 lentelė. Ontologijų verifikavimo įrankio pasirinkimai

| Taisyklės tikrinimo funkcija | Aprašymas | Ontologijos tikrinimo taisyklė |
|------------------------------|--|--------------------------------|
| <i>Label</i> | <i>label</i> anotacijos egzistavimo tikrinimas. | 6 |
| <i>Label SBVR</i> | <i>label_sbvr</i> anotacijos egzistavimo ir sudarymo tikrinimas. | 8 |
| <i>Label language</i> | Kalbos anotacijos tikrinimas. | 7 |
| <i>Classes</i> | Klasės elemento užbaigtumo tikrinimas, tikrinama ar klasės elementas turi vardą. | 1 |
| <i>Object properties</i> | Tikrinamas ontologijos objektinių savybių užbaigtumas, srities ir domeno egzistavimas. | 4 |

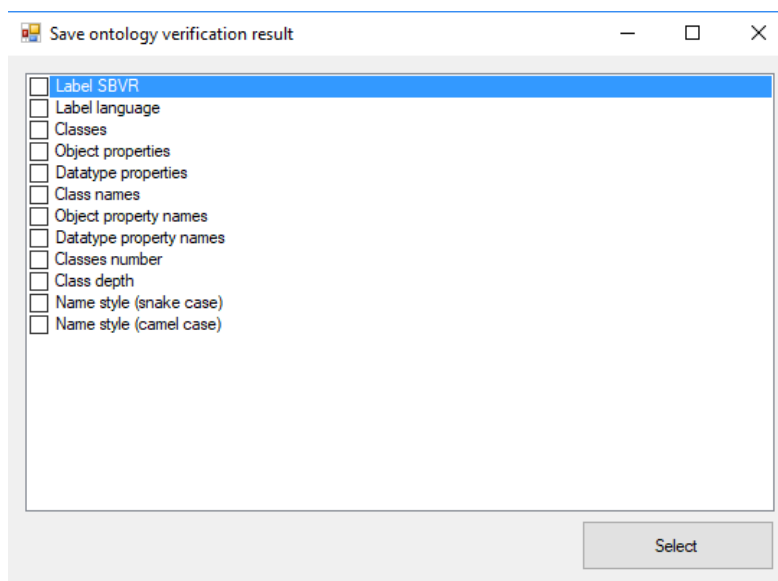
| | | |
|--------------------------------|---|---|
| <i>Datatype properties</i> | Tikrinamas ontologijos duomenų savybių užbaigtumas, jos srities ir domeno egzistavimas. | 5 |
| <i>Class names</i> | Klasės vardų morfologinė analizė, kiekvienas vardas turi būti <i>daiktavardis</i> . | 1 |
| <i>Object property names</i> | Objektinių savybių vardų morfologinė analizė, kiekvienas objektinės savybės vardas turi atitikti struktūrą <i>veiksmažodis + daiktavardis</i> . | 2 |
| <i>Datatype property names</i> | Duomenų savybių vardų morfologinė analizė, kiekvienas duomenų savybės vardas turi atitikti struktūrą <i>turi + daiktavardis</i> . | 3 |
| <i>Classes number</i> | Informacinio pobūdžio verifikatoriaus punktas, kuriame vartotojas informuojamas apie klasių skaičių tikrinamoje ontologijoje. | - |
| <i>Class depth</i> | Klasės gylio nustatymas, tai informacinio pobūdžio punktas, vartotojas yra informuojamas apie ontologijos klasės gylį ir gali nuspręsti ar jis yra per didelis ar ne. | - |
| <i>Name style (snake case)</i> | Ontologijos elementų vardų stiliaus tikrinimas, tikrinama ar visi ontologijos vardai yra parašyti gyvatės stiliumi. | 9 |
| <i>Name style (camel case)</i> | Ontologijos elementų vardų stiliaus tikrinimas, tikrinama ar visi ontologijos vardai yra parašyti kupranugario stiliumi. | 9 |

Ontologijų verifikatorius pritaiko pasirinktą taisyklę ir išnagrinėja ontologijos failą, rezultatą vartotojas mato monitoriaus ekrane. Rasti pažeidimai prieštaraujantys ontologijos tikrinimo taisyklėms yra išskiriami juos pažymint geltona spalva (Pav. 4.5).



4.5 pav. Rasti pažeidimai prieštaraujantys ontologijos tikrinimo taisyklėms

Norėdamas išsaugoti verifikuotos ontologijos rezultato failą, vartotojas pasirenka norimas išsaugoti taisykles. Išsaugojimo langas pavaizduotas 4.6 paveiksle.



4.6 pav. Tikrinti ontologiją ją išsaugant

Rezultatų failas suformuojamas importuojant tikrinamą ontologiją ir išpėjimų ir klaidų anotacijų ontologiją. Papildomų ontologijos failų importavimo eilutės pavaizduotos 4.7 paveiksle. Gautą verifikuotos ontologijos rezultatą galima įkelti į ontologijų redaktorių ir ištaisyti rastus pažeidimus.

```
<owl:Ontology rdf:about="http://semantika.lt/ns/Politics_errors">  
  <owl:imports rdf:resource="http://semantika.lt/ns/Politics" />  
  <owl:imports rdf:resource="http://www.semanticweb.org/sonata/ontologies/2017/3/ispejimu_ir_klaidu_annotacija" />  
</owl:Ontology>
```

4.7 pav. Verifikuotos politikos ontologijos rezultatų failo fragmentas

4.2. Testavimo modelis

Ontologijų verifikavimo įrankio testavimui svarbu pasirinkti ontologiją, kuri būtų viešai prieinama ir naudojama tam tikroje dalykinėje srityse. Šiuo atveju buvo pasirinkta *politikos ontologija*, kuri buvo parsisiūsta iš *Lietuvių kalbos sintaksinės ir semantinės analizės informacinės sistemos*. Ontologijų verifikavimo įrankio algoritmas nuskaito ontologijos klases, duomenų ir objektines savybes, juos verifikuoja pagal ontologijos tikrinimo taisykles ir rastus pažeidimus įrašo į rezultatų failą.

Ontologijų verifikavimo algoritmo testavimas susideda iš šių punktų:

1. ontologijos įkėlimo;
2. kalbos pasirinkimo;
3. taisyklių pasirinkimo;
4. rastų pažeidimų peržiūros, kurie buvo rasti tikrinimo metu;
5. rezultatų išsaugojimo.

4.3. Testavimo duomenys ir rezultatai

Ontologijos apibūdinimas: politikos ontologijos charakteristikos pateikiamos 4.2 lentelėje

4.2 lentelė. Politikos ontologijos charakteristikos

| | |
|-----------------------------|---|
| Pavadinimas | Politikos ontologija (angl. <i>Politics Ontology</i>); |
| URI | http://semantika.lt/ns/Politics |
| Aprašymas | Politikos ontologija skirta politikos įvykiams (sakymams, susitikimams) aprašyti ir sakymams kategorizuoti pagal pateiktą sakymų kategorizavimo schemą. |
| Kalba | en/lt |
| Klasių skaičius | 55 |
| Objektinių savybių skaičius | 17 |
| Duomenų savybių skaičius | 0 |
| Autorius | UAB Sekasoft |
| Skleidėjas | semantika.lt |

Kalbos pasirinkimas: ontologijos elementų vardai aprašomi anglų kalba, tačiau anotacijose naudojamas lietuviškas aprašymas. Todėl kalbos anotacijų tikrinimui kalba buvo pasirinkta lietuvių, o kitų ontologijos elementų tikrinimui pasirinkta anglų kalba.

Testavimo procesas: ontologijų verifikavimo įrankio testavimo metu gauti verifikavimo rezultatai pateikiami lentelėje 4.3. Lentelėje naudojami sutrumpinimai: pažeidimų skaičius (pažeid. sk.), bendras elementų skaičius (bendr. elem. sk.), pažeidimų skaičius išreikštas procentais (pažeid. sk. išreikštas %). Lentelėje naudojama sąvoka *ontologijos eilutė* reprezentuoja ontologijos failo eilutes.

4.3 lentelė. Ontologijų verifikavimo įrankio testavimo metu gauti rezultatai

| Tikrinti elementai | Nustatytų pažeidimų skaičius | | | Rezultatų atvaizdavimas | |
|---------------------|------------------------------|------------------|--------------------------|---|---|
| | | | | Ontologijos eilutė | Nustatytas pažeidimas |
| | Pažeid. sk. | Bendr. elem. sk. | Pažeid. sk. išreikštas % | | |
| Label | 1 | 72 | 1,4 | 372 | <i>Class</i> expected at least one label |
| Label SBVR | 16 | 72 | 22,2 | 101; 116; 139; 166; 205; 231; 298 | <i>Range</i> in <i>label_sbvr</i> is not well formed |
| | | | | 116; 128; 153; 192; 218; 243; 256; 269; 284 | <i>Domain</i> in <i>label_sbvr</i> is not well formed |
| Label language | 1* | 72 | 1,4 | 372 | <i>Class</i> expected to have at least one language label |
| Classes | - | - | - | - | - |
| Object properties | - | - | - | - | - |
| Datatype Properties | - | - | - | - | - |
| Class names | 2 | 55 | 3,6 | 572 | <i>defensive</i> should be noun |

| | | | | | |
|-------------------------|----|----|------|------------------------------|--|
| | | | | 990 | <i>plead</i> should be noun |
| Object property names | - | - | - | - | - |
| Datatype property names | - | - | - | - | - |
| Classes number | 0 | - | - | - | Class count: 55 |
| Class depth | 0 | - | - | - | Ontology depth: 2 |
| Name style (snake case) | 12 | 72 | 16,7 | 113 | Expected snake case: <i>SymmetricProperty</i> |
| | | | | 247 | Expected snake case: <i>topObjectProperty</i> |
| | | | | 2123; 2133; 2136; 2139; 2174 | Expected snake case: <i>AtomList</i> |
| | | | | 2126 | Expected snake case: <i>ClassAtom</i> |
| | | | | 2143 | Expected snake case: <i>DifferentIndividualsAtom</i> |
| | | | | 2152, 2162, 2178 | Expected snake case: <i>IndividualPropertyAtom</i> |
| Name style (camel case) | ** | - | - | - | - |

* - ontologijos egzemplioriai tikrinti lietuvių kalbos tinklo paslauga *semantika.lt*.

** - ontologijos egzemplioriai tikrinti anglų kalbos tinklo paslauga *open.xerox.com*

*** - tikrinamas tik vienas iš dominuojančių stilių.

Verifikuotos ontologijos rezultatų failas: sugeneruotas verifikuotos ontologijos rezultatų failas pridėtas prie priedų 8.2 skyriuje, o skyriuje 8.1 pateikiama importuojamos ontologijos išpėjimų ir klaidų anotacijų ontologija.

4.4. Testavimo rezultatų apibendrinimas

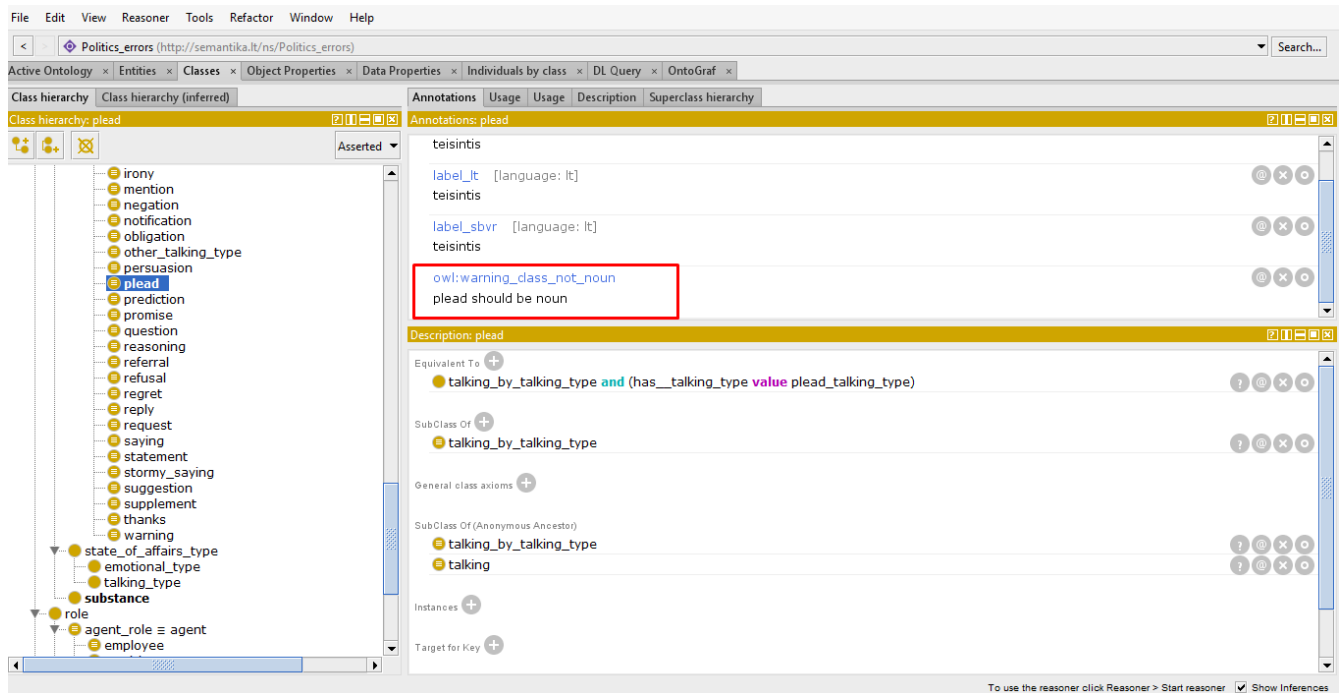
Įrankio testavimui buvo pasirinkta *politikos* ontologija. Ontologijos verifikavimo metu nagrinėti 72 ontologijos elementai. Iš kurių:

- 1,4 % visų nagrinėtų ontologijos elementų neturėjo nurodytos *label* anotacijos, kadangi pažeidimų skaičius palyginus su visu nagrinėtu elementų skaičiumi yra nedidelis, todėl galime teigti, kad tai buvo žmogiška ontologijos kūrimo metu padaryta klaida.
- 22,2 % visų elementų turėjo netaisyklingai suformuluotą *label_sbvr* anotacijos struktūrą. Todėl nerekomenduojama naudoti politikos ontologijos transformacijai į *SBVR* modelį, nes ontologijos elementų transformacija gali būti įvykdyta nekorektiškai.
- 1,4 % visų tikrintų elementų neturėjo anotacijos, kurioje būtų nurodyta ontologijos tikrinimo metu pasirinkta kalba.
- 3,6 % iš visų nagrinėtų klasės elementų buvo suformuluoti netaisyklingai, t. y. klasės vardas buvo suformuotas nepanaudojant *daiktavardžio*. Todėl pažeidžiama pirmoji ontologijos aprašo

tikrinimo taisyklė, siekiant tikslių rezultatų tolimesniuose darbuose su *politikos ontologija* rastus klasių vardų formavimo pažeidimu yra svarbu ištaisyti.

- 16,7 % ontologijos elementų vardų buvo užrašyti nesilaikant gyvatės stiliaus, to pasėkoje ontologija nėra vieningos struktūros, todėl norint ją panaudoti semantinio tinklo kūrime ji gali būti nesuderinama su kitomis ontologijomis.

Patikrinus ontologiją verifikavimo įrankis nustatė joje padarytus pažeidimus, testavimo metu buvo suformuotas verifikavimo rezultatų failas, kuris gali būti įkeltas į ontologijų redagavimo įrankį ir gauti pažeidimai gali būti ištaisyti. Verifikavimo metu suformuoto rezultatų failo atvaizdavimas ontologijų redagavimo programoje pavaizduotas 4.8 paveiksle.



4.8 pav. Verifikavimo metu sugeneruoto rezultatų failo atvaizdavimas ontologijų redaktoriuje

5. EKSPERIMENTINIS ONTOLOGIJŲ VERIFIKAVIMO SISTEMOS TYRIMAS

Šiame skyriuje pateikiamas tiriamojo darbo metu atliktas eksperimentas, kurio tikslas nustatyti dažniausiai daromus ontologijų aprašo taisyklių pažeidimus. Eksperimento metu buvo išnagrinėta 10 skirtingų ontologijų, o rastų pažeidimų skaičius detalizuotas procentais. Aptarti kiekvienos ontologijos eksperimento metu gauti rezultatai, remiantis bendra ontologijų verifikavimo rezultatų lentele apibendrintos išvados.

5.1. Eksperimento planas

Apibrėžimas

Eksperimento tikslas yra verifikuoti viešai prieinamas ontologijas, didžiausią dėmesį skiriant rastų pažeidimų skaičiui. Ontologijų verifikavimo rezultatai parodys, ar sukurtos taisyklės yra korektiškos ir ar gali būti pritaikytos tolimesniuose darbuose, kurie susiję su ontologijų kūrimu.

Planas

Eksperimentas vyksias pasinaudojant magistrinio darbo metu sukurtu ontologijų verifikavimo įrankiu, kurio veikimas yra paremtas suformuluotomis devyniomis ontologijų tikrinimo taisyklėmis. Eksperimentas vykdomas su oficialiomis ontologijomis, kurios yra prieinamos *semantika.lt* ir kitose internetiniuose puslapiuose. Ontologijos pasirinktos iš skirtingų dalykinių sričių, siekiant gauti įvairius rezultatus. Eksperimentas kartojamas su skirtingu dalykinių sričių ontologijomis, įvertinant ontologijos elementų ir gaunamų pažeidimų skaičių.

Ontologijų pasirinkimas vyksias atsitiktinai, nepriklausomai nuo dalykinės srities. Eksperimento metu bus naudojama 10 ontologijų. Eksperimentas vykdomas dviem etapais: pirmiausia patikrinamos ontologijos ontologijų verifikavimo įrankiu, gauti rezultatai ir klaidos surašomos į rezultatų lenteles, antroje eksperimento dalyje verifikavimo rezultatai apibendrinami susumuojant gautų klaidų skaičių visuose ontologijose.

5.2. Eksperimento rezultatai

Atlikus eksperimentą, gauti rezultatai pateikiami atskirose lentelėse. Eksperimentas buvo atliktas kiekvieną ontologiją tikrinant ontologijų verifikavimo įrankiu, į lentelę buvo įrašomas nustatytų pažeidimų skaičius, bendras tikrintų elementų skaičius ir rastų pažeidimų skaičiaus procentinė išraiška. Detali informacija apie rastas klaidas yra pateikiama prieduose 8.3 skyriuje.

Eksperimento metu naudotos ontologijos ir jų vardai pateikiami 5.1. lentelėje. Lentelėse naudojami sutrumpinimai: pažeidimų skaičius (pažeid. sk.), bendras elementų skaičius (bendr. elem. sk.), pažeidimų skaičius išreikštas procentais (pažeid. sk. išreikštas %).

5.1 lentelė. Eksperimento metu naudotos ontologijos

| Nr. | Ontologijos pavadinimas | URI | Kalba |
|-----|--|---|-------|
| 1. | Laiko ontologija | http://semantika.lt/ns/Time | lt/en |
| 2. | Bendravimo ontologija | http://rdfs.org/sioc/ns | en |
| 3. | Vyno ontologija | http://www.w3.org/TR/2003/CR-owl-guide-20030818/wine | en |
| 4. | Viešojo administravimo ontologija | http://semantika.lt/ns/PublicAdministrative | lt/en |
| 5. | Asmenvardžių ir organizacijų pavadinimų ontologija | http://semantika.lt/ns/PersonNames | lt/en |
| 6. | Įvykių ontologija | http://semantika.lt/ns/Events | lt/en |

| | | | |
|-----|-------------------------------|---------------------------------------|-------|
| 7. | Vietovardžių ontologija | http://semantika.lt/ns/Locations | lt/en |
| 8. | Agentų ontologija | http://semantika.lt/ns/Agents | lt/en |
| 9. | Abstrakčių objektų ontologija | http://semantika.lt/ns/AbstractObject | lt/en |
| 10. | SemLT | http://semantika.lt/voc/SemLT | lt/en |

Eksperto metu pirmosios verifikuotos ontologijos charakteristikos pateikiamos 5.2 lentelėje.

5.2 lentelė. Laiko ontologijos charakteristikos

| | |
|-----------------------------|--|
| Numeris | 1 |
| Pavadinimas | Laiko ontologija (angl. <i>Time Ontology</i>); |
| URI | http://semantika.lt/ns/Time |
| Aprašymas | Laiko ontologija skirta laiko klasėms ir jų sąryšiams aprašyti. Siekiant daugkartinio naudojimo, sukurta modulinio ontologijų kūrimo principu. |
| Kalba | en/lt |
| Klasių skaičius | 5 |
| Objektinių savybių skaičius | 0 |
| Duomenų savybių skaičius | 1 |
| Autorius | UAB Sekasoft |
| Skleidėjas | semantika.lt |

Laiko ontologijos verifikavimo rezultatai pateikiami 5.3 lentelėje. Ontologijos verifikavimo metu gauti rezultatai parodė, kad 16,7 % visų tikrintų ontologijos elementų neturėjo *label* anotacijos, toks pat procentas elementų neturėjo ir kalbos anotacijos. Duomenų savybė, kuri buvo panaudota ontologijoje buvo suformuota nesilaikant struktūros *turi + daiktavardis*. Stiliaus analizė parodė, kad ontologijos elementai yra parašyti naudojant gyvatės stilių, tačiau aptiktas vienas elemento vardas, kuris buvo suformuotas kitu stiliumi.

5.3 lentelė. Laiko ontologijos verifikavimo rezultatai

| Ontologijos tikrinimo taisyklė | Nustatytų pažeidimų skaičius | | | Pažeidimo apibrėžimas |
|--|------------------------------|------------------|--------------------------|--|
| | Pažeid. sk. | Bendr. elem. sk. | Pažeid. sk. išreikštas % | |
| Kiekvienas ontologijos elementas turi turėti <i>label</i> anotaciją. | 1 | 6 | 16,7 | <i>Datatype Property</i> expected at least one label |
| Kiekvienoje objektinėje savybėje esanti anotacija <i>label_sbvr</i> , turi būti tokios struktūros: <i>objektinės savybės sritis + veiksmavardis + objektinės savybės domenai</i> | - | - | - | - |
| Kiekviena ontologijos elemento anotacija turi turėti kalbą | 1* | 6* | 16,7* | <i>DatatypeProperty</i> expected to have at least one language label |
| Kiekvienai objektiniai savybei (angl. <i>Object Property</i>) turi būti specifikuotas domenai (angl. <i>Domain</i>) ir sritis (angl. <i>Range</i>) | - | - | - | - |

| | | | | |
|---|-----|---|------|---|
| Kiekvienai duomenų savybei (angl. <i>Datatype Property</i>) turi būti specifikuotas domenas (angl. <i>Domain</i>) ir sritis (angl. <i>Range</i>) | - | - | - | - |
| Kiekvienos ontologijos klasės vardas turi būti <i>daiktavardis</i> | - | - | - | - |
| Kiekvienos objektinės savybės (angl. <i>Object Property</i>) vardas turi atitikti struktūrą: <i>veiksmažodis + daiktavardis</i> | - | - | - | - |
| Kiekvienos duomenų savybės (angl. <i>Data Property</i>) vardas turi atitikti struktūrą: <i>turi + daiktavardis</i> | 1 | 1 | 100 | <i>Time_value</i> should be have/has/had + noun |
| Klasių skaičius | - | - | - | <i>Class count: 5</i> |
| Klasių gylis | - | - | - | Ontology depth: 2 |
| Ontologijos elementų vardai, kuriuos sudaro keli žodžiai, turi būti užrašyti: gyvatės stiliumi | 1 | 6 | 16,7 | Expected snake case: <i>dateTime</i> |
| Ontologijos elementų vardai, kuriuos sudaro keli žodžiai, turi būti užrašyti: kupranugario stiliumi | *** | - | - | - |

* - ontologijos egzemplioriai tikrinti lietuvių kalbos tinklo paslauga *semantika.lt*.

** - ontologijos egzemplioriai tikrinti anglų kalbos tinklo paslauga *open.xerox.com*

*** - tikrinamas tik vienas iš dominuojančių stilijų.

Eksperimento metu antrosios verifikuotos ontologijos charakteristikos pateikiamos 5.4 lentelėje.

5.4 lentelė. Bendravimo ontologijos charakteristikos

| | |
|-----------------------------|--|
| Numeris | 2 |
| Pavadinimas | Bendravimo ontologija (angl. <i>Socially Communities Ontology</i>); |
| URI | http://rdfs.org/sioc/ns |
| Aprašymas | Bendravimo ontologija aprašo bendravimo struktūras, išsireiškimus, vietas. |
| Kalba | en |
| Klasių skaičius | 12 |
| Objektinių savybių skaičius | 49 |
| Duomenų savybių skaičius | 14 |
| Autorius | - |
| Skleidėjas | rdf.org |

Bendravimo ontologijos verifikavimo metu gauti rezultatai pateikiami 5.5 lentelėje. Ontologijos verifikavimo metu gauti rezultatai parodė, kad visi ontologijos objektinių savybių elementai neturėjo nurodytos *label_sbvr* anotacijos. Todėl ši ontologija negalėtų būti naudojama transformacijoje į *SBVR* modelį, nes ontologijos elementai būtų transformuoti nekorektiškai. 2,7 % visų nagrinėtų ontologijos elementų neturėjo nurodytos kalbos anotacijos. 18,4 % objektinių savybių nebuvo specifikuotas domenai ir sritis, ir 61,2 % objektinių savybių vardų buvo suformuoti nesilaikant struktūros *veiksmažodis + daiktavardis*. 85,7 % ontologijos duomenų savybių buvo sudaryti nesilaikant struktūros *turi + daiktavardis*. Verifikavimo metu nustatyta, kad bendravimo ontologijos elementai suformuoti laikantis vieningo kupranugario stiliaus.

5.5 lentelė. Bendravimo ontologijos verifikavimo rezultatai

| Ontologijos tikrinimo taisyklė | Nustatytų pažeidimų skaičius | | | Pažeidimo apibrėžimas |
|---|------------------------------|------------------|--------------------------|--|
| | Pažeid. sk. | Bendr. elem. sk. | Pažeid. sk. išreikštas % | |
| Kiekvienas ontologijos elementas turi turėti <i>label</i> anotaciją. | - | - | - | - |
| Kiekvienoje objektinėje savybėje esanti anotacija <i>label_sbvr</i> , turi būti tokios struktūros: <i>objektinės savybės sritis</i> + <i>veiksmazodis</i> + <i>objektinės savybės domenai</i> . | 49 | 49 | 100 | <i>ObjectProperty</i> is missing <i>label_sbvr</i> |
| Kiekviena ontologijos elemento anotacija turi turėti kalbą | 2** | 75** | 2,7** | <i>DatatypeProperty</i> is missing selected language |
| Kiekvienai objektinei savybei (angl. <i>Object Property</i>) turi būti specifikuotas domenai (angl. <i>Domain</i>) ir sritis (angl. <i>Range</i>) | 9 | 49 | 18,4 | <i>ObjectProperty</i> is missing range |
| Kiekvienai duomenų savybei (angl. <i>Datatype Property</i>) turi būti specifikuotas domenai (angl. <i>Domain</i>) ir sritis (angl. <i>Range</i>) | - | - | - | <i>DatatypeProperty</i> is missing domain |
| | | | | <i>DatatypeProperty</i> is missing range |
| Kiekvienos ontologijos klasės vardas turi būti <i>daiktavardis</i> | - | - | - | - |
| Kiekvienos objektinės savybės (angl. <i>Object Property</i>) vardas turi atitikti struktūrą: <i>veiksmazodis</i> + <i>daiktavardis</i> | 30 | 49 | 61,2 | <i>Name of Object Property</i> should be verb + noun |
| Kiekvienos duomenų savybės (angl. <i>Data Property</i>) vardas turi atitikti struktūrą: <i>turi</i> + <i>daiktavardis</i> | 12 | 14 | 85,7 | <i>Name of Datatype Property</i> should be have or has or had + noun |
| Klasių skaičius | - | - | - | <i>Class count</i> : 12 |
| Klasių gylis | - | - | - | <i>Class depth</i> : 2 |
| Ontologijos elementų vardai, kuriuos sudaro keli žodžiai, turi būti užrašyti: gyvatės stiliumi | - | - | - | - |
| Ontologijos elementų vardai, kuriuos sudaro keli žodžiai, turi būti užrašyti: kupranugario stiliumi | *** | - | - | - |

* - ontologijos egzemplioriai tikrinti lietuvių kalbos tinklo paslauga *semantika.lt*.

** - ontologijos egzemplioriai tikrinti anglų kalbos tinklo paslauga *open.xerox.com*

*** - tikrinamas tik vienas iš dominuojančių stilių.

Eksperimento metu trečios verifikuotos ontologijos charakteristikos pateikiamos 5.6 lentelėje.

5.6 lentelė. Vynų ontologijos charakteristikos

| | |
|-----------------------------|---|
| Numeris | 3 |
| Pavadinimas | Vynų ontologija (angl. <i>Wine Ontology</i>); |
| URI | http://www.w3.org/TR/2003/CR-owl-guide-20030818/wine |
| Aprašymas | Vynų ontologija skirta vynų aprašymams, ši ontologija naudojama kaip ontologijų kūrimo mokymo priemonė. |
| Kalba | en |
| Klasių skaičius | 74 |
| Objektinių savybių skaičius | 12 |
| Duomenų savybių skaičius | 1 |
| Autorius | W3 |
| Skleidėjas | w3.org |

Vyno ontologijos verifikavimo metu gauti rezultatai pateikiami 5.7 lentelėje. Gauti rezultatai parodė, kad 98,9 % visų tikrintų elementų neturi nurodytos *label* anotacijos, to pasekoje neegzistuoja ir kalbos anotacija. 100 % objektinių savybių buvo aprašytos nenaudojant *label_sbvr* anotacijos, todėl ši ontologija negali būti naudojama transformacijai į *SBVR* modelį. 58,3 % objektinių savybių elementų neturėjo nurodytos srities ir domeno, 1,4 % klasių vardų buvo sudaromi nesilaikant klasių vardų sudarymo taisyklės, kuri sako, kad klasės vardas turi būti *daiktavardis*. Tačiau, tai galime traktuoti, kaip netikrą pažeidimą, nes klasė *CabernetSauvignon* yra specifinė vyno rūšis, todėl šio žodžių junginio morfologinės analizės paslauga, kuri yra naudojama ontologijų verifikavimo įrankyje, negali atpažinti. 8,3 % objektinių savybių vardų buvo sudaryti nesilaikant struktūros *veiksmazodis + daiktavardis*. Vynų ontologija sudaryta laikantis vieningo gyvatės stiliaus. Vynų ontologija yra pateikiama kaip viena iš pavyzdžių būsimam ontologijų kūrimo specialistui, tačiau didžioji dalis jos elementų yra suformuoti nesilaikant ontologijos aprašymo taisyklių.

5.7 lentelė. Vynų ontologijos verifikavimo rezultatai

| Ontologijos tikrinimo taisyklė | Nustatytų pažeidimų skaičius | | | Pažeidimo apibrėžimas |
|---|------------------------------|------------------|--------------------------|--|
| | Pažeid. sk. | Bendr. elem. sk. | Pažeid. sk. išreikštas % | |
| Kiekvienas ontologijos elementas turi turėti <i>label</i> anotaciją. | 86 | 87 | 98,9 | <i>Class</i> expected at least one label |
| | | | | <i>Object Property</i> expected at least one label |
| | | | | <i>Data Property</i> expected at least one label |
| Kiekvienoje objektinėje savybėje esanti anotacija <i>label_sbvr</i> , turi būti tokios struktūros: <i>objektinės savybės sritis + veiksmazodis + objektinės savybės domenai</i> . | 12 | 12 | 100 | <i>ObjectProperty</i> is missing <i>label_sbvr</i> |

| | | | | |
|---|------|------|--------|---|
| Kiekviena ontologijos elemento anotacija turi turėti kalbą | 86** | 87** | 98,9** | DatatypeProperty expected to have at least one language label |
| | | | | ObjectProperty expected to have at least one language label |
| | | | | DatatypeProperty expected to have at least one language label |
| Kiekvienai objektiniai savybei (angl. <i>Object Property</i>) turi būti specifikuotas domenas (angl. <i>Domain</i>) ir sritis (angl. <i>Range</i>) | 7 | 12 | 58,3 | ObjectProperty is missing domain |
| | | | | ObjectProperty is missing range |
| Kiekvienai duomenų savybei (angl. <i>Datatype Property</i>) turi būti specifikuotas domenas (angl. <i>Domain</i>) ir sritis (angl. <i>Range</i>) | - | - | - | - |
| Kiekvienos ontologijos klasės vardas turi būti <i>daiktavardis</i> | 1 | 74 | 1,4 | <i>CabernetSauvignon</i> should be noun |
| Kiekvienos objektinės savybės (angl. <i>Object Property</i>) vardas turi atitikti struktūrą: <i>veiksmažodis</i> + <i>daiktavardis</i> | 1 | 12 | 8,3 | <i>adjacentRegion</i> should be verb + noun |
| Kiekvienos duomenų savybės (angl. <i>Data Property</i>) vardas turi atitikti struktūrą: <i>turi</i> + <i>daiktavardis</i> | 1 | 1 | 100 | <i>yearValue</i> should be have or has or had + noun |
| Klasių skaičius | - | - | - | <i>Class</i> count: 74 |
| Klasių gylis | - | - | - | Ontology depth: 2 |
| Ontologijos elementų vardai, kuriuos sudaro keli žodžiai, turi būti užrašyti: gyvatės stiliumi | *** | - | - | - |
| Ontologijos elementų vardai, kuriuos sudaro keli žodžiai, turi būti užrašyti: kupranugario stiliumi | - | - | - | - |

* - ontologijos egzemplioriai tikrinti lietuvių kalbos tinklo paslauga *semantika.lt*.

** - ontologijos egzemplioriai tikrinti anglų kalbos tinklo paslauga *open.xerox.com*

*** - tikrinamas tik vienas iš dominuojančių stilių.

Eksperto metu verifikuotos ketvirtosios ontologijos charakteristikos pateikiamos 5.8 lentelėje.

5.8 lentelė. Viešojo administravimo ontologijos charakteristika

| | |
|-------------|---|
| Numeris | 4 |
| Pavadinimas | Viešojo administravimo ontologija (angl. <i>Public Administrative Ontology</i>); |
| URI | http://semantika.lt/ns/PublicAdministrative |
| Aprašymas | Viešojo administravimo ontologija skirta nelaimių, nusikaltimų, paieškos ir pasiūlos įvykiams aprašyti. Šiuos įvykius atpažįsta semantinis anotatorius. |

| | |
|-----------------------------|---|
| Kalba | en/lt |
| Klasių skaičius | 6 |
| Objektyvių savybių skaičius | 16 |
| Duomenų savybių skaičius | 0 |
| Autorius | Kauno Technologijos universitetas, Informacijos sistemų katedra |
| Skleidėjas | semantika.lt |

Viešojo administravimo ontologijos verifikavimo metu gauti rezultatai pateikiami 5.9 lentelėje. Rezultatai parodė, kad iš egzistuojančių 16 objektyvių savybių buvo rasta 18 pažeidimų. Pažeidimų skaičius didesnis už objektyvių savybių skaičių, nes įrankis kiekvieną žodį, kuris priklauso objektyvios savybės sričiai, domenui ar veiksmožodžiui tikrina atskirai nors jis ir priskiriamas, tai pačiai objektyviai savybei. Viešojo administravimo ontologijos elementai aprašyti laikantis gyvatės stiliaus.

5.9 lentelė. Viešojo administravimo ontologijos verifikavimo rezultatai

| Ontologijos tikrinimo taisyklė | Nustatytų pažeidimų skaičius | | | Pažeidimo apibrėžimas |
|--|------------------------------|------------------|--------------------------|---|
| | Pažeid. sk. | Bendr. elem. sk. | Pažeid. sk. išreikštas % | |
| Kiekvienas ontologijos elementas turi turėti label anotaciją. | - | - | - | - |
| Kiekvienoje objektyvioje savybėje esanti anotacija <i>label_sbvr</i> , turi būti tokios struktūros: <i>objektyvios savybės sritis + veiksmožodis + objektyvios savybės domenai</i> . | 18 | 16 | 112,5 | Domain is not well formed Range is not well formed |
| Kiekviena ontologijos elemento anotacija turi turėti kalbą | - | - | - | - |
| Kiekvienai objektyviai savybei (angl. <i>Object Property</i>) turi būti specifikuotas domenai (angl. <i>Domain</i>) ir sritis (angl. <i>Range</i>) | - | - | - | - |
| Kiekvienai duomenų savybei (angl. <i>Datatype Property</i>) turi būti specifikuotas domenai (angl. <i>Domain</i>) ir sritis (angl. <i>Range</i>) | - | - | - | - |
| Kiekvienos ontologijos klasės vardas turi būti daiktavardis | - | - | - | - |
| Kiekvienos objektyvios savybės (angl. <i>Object Property</i>) vardas turi atitikti struktūrą: <i>veiksmožodis + daiktavardis</i> | - | - | - | - |
| Kiekvienos duomenų savybės (angl. <i>Data Property</i>) vardas turi atitikti struktūrą: <i>turi + daiktavardis</i> | - | - | - | - |
| Klasių skaičius | - | - | - | Class count: 6 |

| | | | | |
|---|-----|---|---|-------------------|
| Klasių gylis | - | - | - | Ontology depth: 2 |
| Ontologijos elementų vardai, kuriuos sudaro keli žodžiai, turi būti užrašyti: gyvatės stiliumi | - | - | - | - |
| Ontologijos elementų vardai, kuriuos sudaro keli žodžiai, turi būti užrašyti: kupranugario stiliumi | *** | - | - | - |

* - ontologijos egzemplioriai tikrinti lietuvių kalbos tinklo paslauga *semantika.lt*.

** - ontologijos egzemplioriai tikrinti anglų kalbos tinklo paslauga *open.xerox.com*

*** - tikrinamas tik vienas iš dominuojančių stilių.

Eksperto metu verifikuotos penktosios ontologijos charakteristikos pateikiamos 5.10 lentelėje.

5.10 lentelė. Asmenvardžių ir organizacijų pavadinimų ontologijos verifikavimo rezultatai

| | |
|-----------------------------|---|
| Numeris | 5 |
| Pavadinimas | Asmenvardžių ir organizacijų pavadinimų ontologija (angl. <i>Person Names Ontology</i>); |
| URI | http://semantika.lt/ns/PersonNames |
| Aprašymas | Lietuvių kalbos asmenvardžių ir organizacijų pavadinimų ontologija |
| Kalba | en/lt |
| Klasių skaičius | 7 |
| Objektinių savybių skaičius | 4 |
| Duomenų savybių skaičius | 4 |
| Autorius | UAB Fotonika |
| Skleidėjas | semantika.lt |

Asmenvardžių ir organizacijų pavadinimų ontologijos verifikavimo metu gauti rezultatai pateikiami 5.11 lentelėje. Verifikavimo metu gauti rezultatai parodė, kad iš 15 ontologijos elementų nei vienas neturėjo aprašytos *label* anotacijos, 4 objektinės savybės iš 4 neturėjo *label_svr* anotacijos. 25 % visų tikrintų objektinių savybių vardų buvo sudaryti nesilaikant struktūros *veiksmažodis + daiktavardis*. 100 % duomenų savybių vardų buvo aprašyti nesilaikant tiriamojo darbo metu išskirtų bendrų ontologijos formavimo taisyklių. Ontologijos stiliaus tikrinimo metu buvo rasti 73,3 % pažeidimų, kurie informavo, kad ontologijos vardų elementai suformuoti nesilaikant vieningo gyvatės stiliaus. Toks ontologijos bendrų ontologijos aprašo taisyklių nesilaikymas neleidžia atlikti ne tik ontologijos transformacijos, bet ir jos panaudojimas kuriant semantinį tinklą gali būti nekorektiškas ir nesuderinamas su kitomis ontologijomis.

5.11 lentelė. Asmenvardžių ir organizacijų pavadinimų ontologijos verifikavimo rezultatai

| Ontologijos tikrinimo taisyklė | Nustatytų pažeidimų skaičius | | | Pažeidimo apibrėžimas |
|--|------------------------------|------------------|--------------------------|---|
| | Pažeid. sk. | Bendr. elem. sk. | Pažeid. sk. išreikštas % | |
| Kiekvienas ontologijos elementas turi turėti <i>label</i> anotaciją. | 15 | 15 | 100 | <i>ObjectProperty</i> expected at least one label |

| | | | | |
|---|-----|-----|------|---|
| | | | | <i>DatatypeProperty</i> expected to have at least one label |
| | | | | <i>Class</i> expected to have at least one label |
| Kiekvienoje objektinėje savybėje esanti anotacija <i>label_sbvr</i> , turi būti tokios struktūros: <i>objektinės savybės sritis + veiksmazodis + objektinės savybės domenas</i> . | 4 | 4 | 100 | <i>ObjectProperty</i> is missing <i>label_sbvr</i> |
| Kiekviena ontologijos elemento anotacija turi turėti kalbą | 15* | 15* | 100* | <i>ObjectProperty</i> is missing selected language (lt) |
| | | | | <i>DatatypeProperty</i> is missing selected language (lt) |
| | | | | <i>Class</i> is missing selected language (lt) |
| Kiekvienai objektiniai savybei (angl. <i>Object Property</i>) turi būti specifikuotas domenas (angl. <i>Domain</i>) ir sritis (angl. <i>Range</i>) | - | - | - | - |
| Kiekvienai duomenų savybei (angl. <i>Datatype Property</i>) turi būti specifikuotas domenas (angl. <i>Domain</i>) ir sritis (angl. <i>Range</i>) | - | - | - | - |
| Kiekvienos ontologijos klasės vardas turi būti <i>daiktavardis</i> | - | - | - | - |
| Kiekvienos objektinės savybės (angl. <i>Object Property</i>) vardas turi atitikti struktūrą: <i>veiksmazodis + daiktavardis</i> | 1 | 4 | 25 | <i>Language</i> only one word, should be verb + noun |
| Kiekvienos duomenų savybės (angl. <i>Data Property</i>) vardas turi atitikti struktūrą: <i>turi + daiktavardis</i> | 4 | 4 | 100 | <i>Country</i> only one word, should be have/has/had + noun |
| | | | | <i>Full_name</i> should be have or has or had + noun |
| | | | | <i>Industry</i> only one word, should be |

| | | | | |
|---|-----|----|------|---|
| | | | | have/has/had + noun |
| | | | | <i>Lithuanian</i> only one word, should be have/has/had + noun |
| Klasių skaičius | - | - | - | <i>Class count: 7</i> |
| Klasių gylis | - | - | - | Ontology depth: 2 |
| Ontologijos elementų vardai, kuriuos sudaro keli žodžiai, turi būti užrašyti: gyvatės stiliumi | 11 | 15 | 73,3 | Expected snake case: <i>FunctionalProperty</i> |
| | | | | Expected snake case: <i>Person_Name</i> |
| | | | | Expected snake case: <i>NameKind</i> |
| Ontologijos elementų vardai, kuriuos sudaro keli žodžiai, turi būti užrašyti: kupranugario stiliumi | *** | - | - | - |

* - ontologijos egzemplioriai tikrinti lietuvių kalbos tinklo paslauga *semantika.lt*.

** - ontologijos egzemplioriai tikrinti anglų kalbos tinklo paslauga *open.xerox.com*

*** - tikrinamas tik vienas iš dominuojančių stilių.

Eksperto metu šeštosios verifikuotos ontologijos charakteristikos pateikiamos 5.12 lentelėje

5.12 lentelė. Įvykių ontologijos charakteristikos

| | |
|-----------------------------|---|
| Numeris | 6 |
| Pavadinimas | Įvykių ontologija (angl. <i>Events Ontology</i>); |
| URI | http://semantika.lt/ns/Events |
| Aprašymas | Įvykių ontologija skirta aprašyti bendriniam įvykiams |
| Kalba | en/lt |
| Klasių skaičius | 26 |
| Objektinių savybių skaičius | 54 |
| Duomenų savybių skaičius | 3 |
| Autorius | UAB Sekasoft |
| Skleidėjas | semantika.lt |

Įvykių ontologijos verifikavimo rezultatai pateikiami 5.13 lentelėje. Verifikavimo rezultatai parodė, kad 2,4 % tikrintų elementų neturi *label* anotacijos. 18,5 % objektinių savybių elementų turi *label_sbvr* anotaciją, tačiau ji yra suformuota klaidingai. Eksperimento metu pastebėta, kad domenai arba sritis ne visada pateikiami juos aprašant atskira klase. Todėl įrankis neradęs klasės, kuri turėtų būti, objektinėje savybėje *label_sbvr* struktūroje panaudotą domeną ir sritį traktuoja kaip netinkamus. 2,4 % tikrintų elementų neturi nurodytos *label* anotacijos kalbos. 3,7 % visų objektinių savybių neturi nurodytos srities arba domeno, o 9,3 % objektinių savybių vardų suformuoti nesilaikant struktūros *veiksmazodis + daiktavardis*. Visi įvykių ontologijoje naudojami duomenų savybių vardai buvo suformuoti nesilaikant rekomenduojamų formavimo taisyklių *turi + veiksmazodis*. Įrankis nustatė, kad 5,6 % ontologijos elementų vardų buvo sukurti nesilaikant gyvatės stiliaus.

5.13 lentelė. Įvykių ontologijos verifikavimo rezultatai

| Ontologijos tikrinimo taisyklė | Nustatytų pažeidimų skaičius | | | Pažeidimo apibrėžimas |
|---|------------------------------|------------------|--------------------------|--|
| | Pažeid. sk. | Bendr. elem. sk. | Pažeid. sk. išreikštas % | |
| Kiekvienas ontologijos elementas turi turėti <i>label</i> anotaciją. | 2 | 83 | 2,4 | <i>Class</i> expected to have at least one label |
| Kiekvienoje objektinėje savybėje esanti anotacija <i>label_sbvr</i> , turi būti tokios struktūros: <i>objektinės savybės sritis + veiksmazodis + objektinės savybės domenai</i> . | 10 | 54 | 18,5 | <i>Domain</i> is not well formed |
| | | | | <i>Range</i> is not well formed |
| Kiekviena ontologijos elemento anotacija turi turėti kalbą | 2* | 83* | 2,4* | <i>Class</i> expected to have at least one language label |
| Kiekvienai objektiniai savybei (angl. <i>Object Property</i>) turi būti specifikuotas domenai (angl. <i>Domain</i>) ir sritis (angl. <i>Range</i>) | 2 | 54 | 3,7 | <i>ObjectProperty</i> is missing <i>domain</i> |
| | | | | <i>ObjectProperty</i> is missing <i>range</i> |
| Kiekvienai duomenų savybei (angl. <i>Datatype Property</i>) turi būti specifikuotas domenai (angl. <i>Domain</i>) ir sritis (angl. <i>Range</i>) | - | - | - | - |
| Kiekvienos ontologijos klasės vardas turi būti <i>daiktavardis</i> | - | - | - | - |
| Kiekvienos objektinės savybės (angl. <i>Object Property</i>) vardas turi atitikti struktūrą: <i>veiksmazodis + daiktavardis</i> | 5 | 54 | 9,3 | <i>has__position_held</i> should be verb + noun |
| | | | | <i>is_assigned_to__position_held</i> should be verb + noun |

| | | | | |
|---|-----|----|-----|---|
| | | | | <i>provide__position_held</i> should be verb + noun |
| | | | | <i>representation_in__event</i> should be verb + noun |
| | | | | <i>representation_of__organization</i> should be verb + noun |
| Kiekvienos duomenų savybės (angl. <i>Data Property</i>) vardas turi atitikti struktūrą: <i>turi + daiktavardis</i> | 3 | 3 | 100 | <i>size_decimal_value</i> should be have or has or had + noun |
| | | | | <i>size_integer_value</i> should be have or has or had + noun |
| | | | | <i>size_string_value</i> should be have or has or had + noun |
| Klasių skaičius | - | - | - | Class count: 26 |
| Klasių gylis | - | - | - | Ontology depth: 2 |
| Ontologijos elementų vardai, kuriuos sudaro keli žodžiai, turi būti užrašyti: gyvatės stiliumi | 3 | 54 | 5,6 | Expected snake case: <i>FunctionalProperty</i> |
| Ontologijos elementų vardai, kuriuos sudaro keli žodžiai, turi būti užrašyti: kupranugario stiliumi. | *** | - | - | - |

* - ontologijos egzemplioriai tikrinti lietuvių kalbos tinklo paslauga *semantika.lt*.

** - ontologijos egzemplioriai tikrinti anglų kalbos tinklo paslauga *open.xerox.com*

*** - tikrinamas tik vienas iš dominuojančių stilių.

Eksperto metu septintosios verifikuotos ontologijos charakteristikos pateikiamos 5.14 lentelėje.

5.14 lentelė. Vietovardžių ontologijos charakteristikos

| | |
|-----------------------------|---|
| Numeris | 7 |
| Pavadinimas | Vietovardžių ontologija (angl. <i>Locations Ontology</i>); |
| URI | http://semantika.lt/ns/Events |
| Aprašymas | Vietovardžių ontologija skirta aprašyti vietovėms. |
| Kalba | en/lt |
| Klasių skaičius | 13 |
| Objektinių savybių skaičius | 4 |
| Duomenų savybių skaičius | 0 |
| Autorius | UAB Fotonija |
| Skleidėjas | semantika.lt |

Vietovardžių ontologijos verifikavimo metu gauti rezultatai pateikiami 5.15 lentelėje. Ontologijos verifikavimo rezultatai parodė, kad ontologija iš dalies išpildo tiriamojo darbo metu išskirtas bendrąsias ontologijų aprašo tikrinimo taisykles. Verifikavimo metu įrankis nustatė 3 pažeidimus *label_svr* formavime, tačiau juos peržiūrėjus buvo pastebėta, kad domenai ir sritis nebuvo aprašyti klasių elementuose, todėl įrankis negali patikrinti lietuviškos žodžio reikšmės (domenas ir sritis ontologijoje aprašyti anglų kalba). To pasekoje ontologija gali būti naudojama transformacijai į *SBVR* modelį, taip pat ši ontologija gali būti panaudojama ir semantinio tinklo kūrimui, nes yra struktūriškai teisinga.

5.15 lentelė. Vietovardžių ontologijos verifikavimo rezultatai

| Ontologijos tikrinimo taisyklė | Nustatytų pažeidimų skaičius | | | Pažeidimo apibrėžimas |
|--|------------------------------|------------------|--------------------------|----------------------------------|
| | Pažeid. sk. | Bendr. elem. sk. | Pažeid. sk. išreikštas % | |
| Kiekvienas ontologijos elementas turi turėti <i>label</i> anotaciją. | - | - | - | - |
| Kiekvienoje objektinėje savybėje esanti anotacija <i>label_svr</i> , turi būti tokios struktūros: <i>objektinės savybės sritis + veiksmožodis + objektinės savybės domenai</i> . | 3 | 4 | 75 | <i>Range</i> is not well formed |
| | | | | <i>Domain</i> is not well formed |
| Kiekviena ontologijos elemento anotacija turi turėti kalbą | - | - | - | - |
| Kiekvienai objektiniai savybei (angl. <i>Object Property</i>) turi būti specifikuotas domenai (angl. <i>Domain</i>) ir sritis (angl. <i>Range</i>) | - | - | - | - |
| Kiekvienai duomenų savybei (angl. <i>Datatype Property</i>) turi būti specifikuotas domenai (angl. <i>Domain</i>) ir sritis (angl. <i>Range</i>) | - | - | - | - |
| Kiekvienos ontologijos klasės vardas turi būti daiktavardis | - | - | - | - |
| Kiekvienos objektinės savybės (angl. <i>Object Property</i>) vardas turi atitikti struktūrą: <i>veiksmožodis + daiktavardis</i> | - | - | - | - |
| Kiekvienos duomenų savybės (angl. <i>Data Property</i>) vardas turi atitikti struktūrą: <i>turi + daiktavardis</i> | - | - | - | - |
| Klasių skaičius | - | - | - | <i>Class</i> count: 13 |
| Klasių gylis | - | - | - | Ontology depth: 2 |
| Ontologijos elementų vardai, kuriuos sudaro keli žodžiai, turi būti užrašyti: gyvatės stiliumi | - | - | - | - |

| | | | | |
|---|-----|---|---|---|
| Ontologijos elementų vardai, kuriuos sudaro keli žodžiai, turi būti užrašyti: kupranugario stiliumi | *** | - | - | - |
|---|-----|---|---|---|

* - ontologijos egzemplioriai tikrinti lietuvių kalbos tinklo paslauga *semantika.lt*.

** - ontologijos egzemplioriai tikrinti anglų kalbos tinklo paslauga *open.xerox.com*

*** - tikrinamas tik vienas iš dominuojančių stilių.

Ekspimento metu aštuntosios verifikuotos ontologijos charakteristikos pateikiamos 5.16 lentelėje.

5.16 lentelė. Agentų ontologijos charakteristikos

| | |
|-----------------------------|--|
| Pavadinimas | Agentų ontologija (angl. <i>Agents Ontology</i>); |
| URI | http://semantika.lt/ns/Agents |
| Aprašymas | Agentų ontologija skirta aprašyti asmenų, organizacijų ir grupių konceptams ir jų sąryšiams, kurie naudojami specialiuųjų sričių ontologijų rinkiniuose. |
| Kalba | en/lt |
| Klasių skaičius | 49 |
| Objektinių savybių skaičius | 4 |
| Duomenų savybių skaičius | 0 |
| Autorius | UAB Sekasoft |
| Skleidėjas | semantika.lt |

Agentų ontologijos verifikavimo metu gauti rezultatai pateikiami 5.17 lentelėje. Verifikavimo metu gauti rezultatai parodė, kad 1,9 % visų tikrintų ontologijos elementų neturėjo nurodytos *label* anotacijos. Ontologija parašyta jos elementus formuojant gyvatės stiliumi, tačiau nustatyti trys elementai, kurie buvo suformuluoti kupranugario stiliumi. Remiantis gautais rezultatais galime teigti, kad ontologija gali būti naudojama transformacijai arba kitose veiklose susijusiomis su ontologijų kūrimu.

5.17 lentelė. Agentų ontologijos verifikavimo rezultatai

| Ontologijos tikrinimo taisyklė | Nustatytų pažeidimų skaičius | | | Pažeidimo apibrėžimas |
|---|------------------------------|------------------|--------------------------|--|
| | Pažeid. sk. | Bendr. elem. sk. | Pažeid. sk. išreikštas % | |
| Kiekvienas ontologijos elementas turi turėti <i>label</i> anotaciją. | 1 | 53 | 1,9 | <i>Class</i> expected to have at least one label |
| Kiekvienoje objektinėje savybėje esanti anotacija <i>label_sbvr</i> , turi būti tokios struktūros: <i>objektinės savybės sritis + veiksmazodis + objektinės savybės domenai</i> . | - | - | - | - |

| | | | | |
|---|-----|----|-----|---|
| Kiekviena ontologijos elemento anotacija turi turėti kalbą | 1* | 53 | 1,9 | <i>Class expected to have at least one language label</i> |
| Kiekvienai objektiniai savybei (angl. <i>Object Property</i>) turi būti specifikuotas domenas (angl. <i>Domain</i>) ir sritis (angl. <i>Range</i>) | - | - | - | - |
| Kiekvienai duomenų savybei (angl. <i>Datatype Property</i>) turi būti specifikuotas domenas (angl. <i>Domain</i>) ir sritis (angl. <i>Range</i>) | - | - | - | - |
| Kiekvienos ontologijos klasės vardas turi būti <i>daiktavardis</i> | - | - | - | - |
| Kiekvienos objektinės savybės (angl. <i>Object Property</i>) vardas turi atitikti struktūrą: <i>veiksmažodis + daiktavardis</i> | - | - | - | - |
| Kiekvienos duomenų savybės (angl. <i>Data Property</i>) vardas turi atitikti struktūrą: <i>turi + daiktavardis</i> | - | - | - | - |
| Klasių skaičius | - | - | - | Class count: 49 |
| Klasių gylis | - | - | - | Ontology depth: 2 |
| Ontologijos elementų vardai, kuriuos sudaro keli žodžiai, turi būti užrašyti: gyvatės stiliumi | 3 | - | - | Expected snake case: <i>topObjectProperty</i> |
| Ontologijos elementų vardai, kuriuos sudaro keli žodžiai, turi būti užrašyti: kupranugario stiliumi | *** | - | - | - |

* - ontologijos egzemplioriai tikrinti lietuvių kalbos tinklo paslauga *semantika.lt*.

** - ontologijos egzemplioriai tikrinti anglų kalbos tinklo paslauga *open.xerox.com*

*** - tikrinamas tik vienas iš dominuojančių stilių.

Eksperto metu verifikuotos devintosios ontologijos charakteristikos pateikiamos 5.18 lentelėje.

5.18 lentelė. Abstrakčių objektų ontologijos charakteristikos

| | |
|-----------------------------|--|
| Pavadinimas | Abstrakčių objektų ontologija (angl. <i>Abstract Object Ontology</i>); |
| URI | http://semantika.lt/ns/AbstractObject |
| Aprašymas | Abstrakčių objektų ontologija skirta viešojo sektoriaus ir verslo poreikiams aktualiems abstrakčių objektų conceptams (turtui, nedarbui, skundams, lėšoms, ekonomikai ir kt.) aprašyti, kategorizuojant juos į grupes. |
| Kalba | en/lt |
| Klasių skaičius | 49 |
| Objektinių savybių skaičius | 4 |
| Duomenų savybių skaičius | 0 |
| Autorius | UAB Sekasoft |
| Skleidėjas | semantika.lt |

Abstrakčių objektų ontologijos verifikavimo metu gauti rezultatai pateikiami 5.19 lentelėje. Ontologijos verifikavimo metu nebuvo rastas nei vienas pažeidimas, kuris prieštarautų tiriamojo darbo metu specifikuotoms bendrinėms ontologijos aprašo taisyklėms. Todėl abstrakčių objektų ontologija gali būti naudojama transformacijai į *SBVR* modelį, taip pat ją galima naudoti ir kitose srityse, kuriose yra nagrinėjamos ontologijos.

5.19 lentelė. Abstrakčių objektų ontologijos verifikavimo rezultatai

| Ontologijos tikrinimo taisyklė | Nustatytų pažeidimų skaičius | | | Pažeidimo apibrėžimas |
|---|------------------------------|------------------|--------------------------|-----------------------|
| | Pažeid. sk. | Bendr. elem. sk. | Pažeid. sk. išreikštas % | |
| Kiekvienas ontologijos elementas turi turėti <i>label</i> anotaciją. | - | - | - | - |
| Kiekvienoje objektinėje savybėje esanti anotacija <i>label_sbvr</i> , turi būti tokios struktūros: <i>objektinės savybės sritis</i> + <i>veiksmažodis</i> + <i>objektinės savybės domenai</i> . | - | - | - | - |
| Kiekviena ontologijos elemento anotacija turi turėti kalbą | - | - | - | - |
| Kiekvienai objektiniai savybei (angl. <i>Object Property</i>) turi būti specifikuotas domenai (angl. <i>Domain</i>) ir sritis (angl. <i>Range</i>) | - | - | - | - |
| Kiekvienai duomenų savybei (angl. <i>Datatype Property</i>) turi būti specifikuotas domenai (angl. <i>Domain</i>) ir sritis (angl. <i>Range</i>) | - | - | - | - |
| Kiekvienos ontologijos klasės vardas turi būti <i>daiktavardis</i> | - | - | - | - |
| Kiekvienos objektinės savybės (angl. <i>Object Property</i>) vardas turi atitikti struktūrą: <i>veiksmažodis</i> + <i>daiktavardis</i> | - | - | - | - |
| Kiekvienos duomenų savybės (angl. <i>Data Property</i>) vardas turi atitikti struktūrą: <i>turi</i> + <i>daiktavardis</i> | - | - | - | - |
| Klasių skaičius | - | - | - | Class count: 17 |
| Klasių gylis | - | - | - | Ontology depth: 2 |
| Ontologijos elementų vardai, kuriuos sudaro keli žodžiai, turi būti užrašyti: gyvatės stiliumi | - | - | - | - |
| Ontologijos elementų vardai, kuriuos sudaro keli žodžiai, turi būti užrašyti: kupranugario stiliumi | *** | - | - | - |

* - ontologijos egzemplioriai tikrinti lietuvių kalbos tinklo paslauga *semantika.lt*.

** - ontologijos egzemplioriai tikrinti anglų kalbos tinklo paslauga *open.xerox.com*

*** - tikrinamas tik vienas iš dominuojančių stilių.

Eksperimento metu verifikuotos dešimtosios ontologijos charakteristikos pateikiamos 5.20 lentelėje.

5.20 lentelė. SemLt ontologijos charakteristikos

| | |
|-----------------------------|--|
| Pavadinimas | SemLT ontologija (angl. <i>SemLt Ontology</i>); |
| URI | http://semantika.lt/ns/SemLT |
| Aprašymas | SemLT ontologija yra viena iš pagrindinių ontologijų. Ji aprašo dokumentus, objektus ir jų sąryšius, kurie naudojami specialiųjų sričių ontologijų rinkiniuose. Siekiant daugkartinio naudojimo, sukurta modulinis ontologijų kūrimo principu. |
| Kalba | en/lt |
| Klasių skaičius | 4 |
| Objektinių savybių skaičius | 5 |
| Duomenų savybių skaičius | 7 |
| Autorius | UAB Sekasoft |
| Skleidėjas | semantika.lt |

SemLt ontologijos verifikavimo metu gauti rezultatai pateikiami 5.21 lentelėje. Rezultatai parodė, kad viena *label_sbvr* anotacija buvo suformuota nesilaikant struktūros *domenas + veiksmazodis + sritis*. Verifikavimo įrankis nustatė, kad visi ontologijos duomenų savybių vardai buvo suformuluoti netaisyklingai, t. y. nenaudojant struktūros *turi + daiktavardis*. Buvo nustatyta, kad ontologija parašyta gyvatės stiliumi, tačiau buvo aptikta, kad 43,8 % visų ontologijos elementų pažeidžia šį stilių, todėl ši ontologija negali būti laikoma struktūriškai teisinga, nes yra sumaišyti du žodžių formavimo stiliai.

5.21 lentelė. SemLt ontologijos verifikavimo rezultatai

| Ontologijos tikrinimo taisyklė | Nustatytų pažeidimų skaičius | | | Pažeidimo apibrėžimas |
|---|------------------------------|------------------|--------------------------|--|
| | Pažeid. sk. | Bendr. elem. sk. | Pažeid. sk. išreikštas % | |
| Kiekvienas ontologijos elementas turi turėti <i>label</i> anotaciją. | - | - | - | - |
| Kiekvienoje objektinėje savybėje esanti anotacija <i>label_sbvr</i> , turi būti tokios struktūros: <i>objektinės savybės sritis + veiksmazodis + objektinės savybės domenas</i> . | 1 | 5 | 20 | <i>Verb was not used in label_sbvr</i> |
| Kiekviena ontologijos elemento anotacija turi turėti kalbą | - | - | - | - |
| Kiekvienai objektiniai savybei (angl. <i>Object Property</i>) turi būti specifikuotas domenas (angl. <i>Domain</i>) ir sritis (angl. <i>Range</i>) | - | - | - | - |

| | | | | |
|---|-----|----|------|---|
| Kiekvienai duomenų savybei (angl. <i>Datatype Property</i>) turi būti specifikuotas domenas (angl. <i>Domain</i>) ir sritis (angl. <i>Range</i>) | - | - | - | - |
| Kiekvienos ontologijos klasės vardas turi būti <i>daiktavardis</i> | - | - | - | - |
| Kiekvienos objektinės savybės (angl. <i>Object Property</i>) vardas turi atitikti struktūrą: <i>veiksmažodis + daiktavardis</i> | - | - | - | - |
| Kiekvienos duomenų savybės (angl. <i>Data Property</i>) vardas turi atitikti struktūrą: <i>turi + daiktavardis</i> | 7 | 7 | 100 | <i>annotation_date</i> should be have or has or had + noun <i>created_date</i> should be have or has or had + noun <i>document_ID</i> should be have or has or had + noun <i>document_URI</i> should be have or has or had + noun <i>is_abstract</i> should be have or has or had + noun <i>source_name</i> should be have or has or had + noun <i>trusted</i> only one word, should be have/has/had + noun |
| Klasių skaičius | - | - | - | Class count: 4 |
| Klasių gylis | - | - | - | Ontology depth: 2 |
| Ontologijos elementų vardai, kuriuos sudaro keli žodžiai, turi būti užrašyti: gyvatės stiliumi | 7 | 16 | 43,8 | Expected snake case: <i>FunctionalProperty</i> Expected snake case: <i>SymmetricProperty</i> Expected snake case: <i>dateTime</i> Expected snake case: <i>dateTime</i> Expected snake case: <i>document_ID</i> Expected snake case: <i>document_URI</i> Expected snake case: <i>anyURI</i> |
| Ontologijos elementų vardai, kuriuos sudaro keli žodžiai, turi būti užrašyti: kupranugario stiliumi | *** | - | - | - |

* - ontologijos egzemplioriai tikrinti lietuvių kalbos tinklo paslauga *semantika.lt*.

** - ontologijos egzemplioriai tikrinti anglų kalbos tinklo paslauga *open.xerox.com*

*** - tikrinamas tik vienas iš dominuojančių stilių.

Ivykdžius eksperimentą buvo sudaryta bendra lentelė (žr. 5.22 lentelė), kurioje pateikiami dažniausiai pasikartoję pažeidimai.

5.22 lentelė. Eksperimento rezultatai

| Ontologijos tikrinimo taisyklė | Taisyklės numeris | Tikrintų ontologijų skaičius | Nepažeistos ontologijos | Neaptikta tikrintų elementų | Elementai netinkamai suformuoti | Bendras pažeidimų skaičius |
|---|-------------------|------------------------------|-------------------------|-----------------------------|---------------------------------|----------------------------|
| Kiekvienas ontologijos elementas turi turėti <i>label</i> anotaciją. | 6 | 10 | 5 | 5 | 0 | 105 |
| Kiekvienoje objektinėje savybėje esanti anotacija <i>label_sbvr</i> , turi būti tokios struktūros: <i>objektinės savybės sritis</i> + <i>veiksmožodis</i> + <i>objektinės savybės domenai</i> . | 8 | 10 | 3 | 3 | 4 | 97 |
| Kiekviena ontologijos elemento anotacija turi turėti kalbą | 7 | 10 | 5 | 5 | 0 | 19*; 93** |
| Kiekvienai objektiniai savybei (angl. <i>Object Property</i>) turi būti specifikuotas domenai (angl. <i>Domain</i>) ir sritis (angl. <i>Range</i>) | 4 | 10 | 7 | 0 | 3 | 18 |
| Kiekvienai duomenų savybei (angl. <i>Datatype Property</i>) turi būti specifikuotas domenai (angl. <i>Domain</i>) ir sritis (angl. <i>Range</i>) | 5 | 10 | 10 | 0 | 0 | 0 |
| Kiekvienos ontologijos klasės vardas turi būti daiktavardis | 1 | 10 | 9 | 0 | 1 | 1 |
| Kiekvienos objektinės savybės (angl. <i>Object Property</i>) vardas turi atitikti struktūrą: <i>veiksmožodis</i> + <i>daiktavardis</i> | 2 | 10 | 6 | 0 | 4 | 37 |
| Kiekvienos duomenų savybės (angl. <i>Data Property</i>) vardas turi atitikti struktūrą: <i>turi</i> + <i>daiktavardis</i> | 3 | 10 | 4 | 0 | 6 | 28 |
| Ontologijos elementų vardai, kuriuos sudaro keli žodžiai, turi būti užrašyti: gyvatės arba kupranugario stiliumi. | 9 | 9 | 4 | 0 | 5 | 25**** |
| Ontologijos elementų vardai, kuriuos sudaro keli žodžiai, turi būti užrašyti: gyvatės arba kupranugario stiliumi. | 9 | 1 | 1 | 0 | 0 | 0*** |

* - ontologijos egzemplioriai tikrinti lietuvių kalbos tinklo paslauga *semantika.lt*.

** - ontologijos egzemplioriai tikrinti anglų kalbos tinklo paslauga *open.xerox.com*

*** - tikrinamas tik vienas iš dominuojančių stilių.

Išnagrinėjus 10 skirtingų ontologijų, anotacija *label* nebuvo aptikta 5 iš jų, tai sudaro 50 % visų tikrintų ontologijų skaičiaus. Kitose 5 ontologijose *label* anotacija buvo aptikta ir suformuota teisingai. Tokios ontologijos kaip *administravimo*, *bendravimo*, *vietovardžių*, *abstrakčių objektų* gali būti lengvai suprantamos ne tik kompiuteriams, bet ir žmonėms, nes jose yra panaudota *label* anotacija.

Anotacija *label_sbvr* buvo aptikta ir tinkamai suformuluota 3 ontologijose, septynios ontologijos neturėjo *label_sbvr* anotacijos arba ji buvo suformuota nesilaikant struktūros. Tokios ontologijos kaip *laiko*, *agentų* ir *abstrakčių objektų* gali būti transformuojamos į *SBVR* modelį korektiškai transformuojant kiekvieną ontologijos objektinę savybę.

Anotacijos *label* kalba buvo nurodyta visose ontologijose, kurios turėjo *label* anotaciją. Todėl galime teigti, kad nurodant *label* anotaciją ontologijoje yra nurodomas ir kalbos elementas. Be to buvo pastebėta, kad tik anglų kalba parašytose ontologijose nėra nurodoma *label* kalbos anotacija, taip yra todėl, kad ontologijų modeliotojai ją laiko kaip tarptautinį standartą naudojamą ontologijų kūrimo, kurio nurodyti neprivaloma.

Patikrinus 10 ontologijų objektinės savybės sritis ir domeną nebuvo aprašyti trijose iš jų. Visų kitų ontologijų objektinės savybės buvo suformuotos panaudojant domeną ir sritį. Tokios ontologijos yra pilnai išbaigtos, todėl gali būti naudojamos tolimesniuose ontologijos kūrimo etapuose.

Duomenų savybių sritis ir domeną buvo korektiškai nurodytas visose tikrintose ontologijose.

Ontologijos klasės vardas netinkamai buvo suformuotas tik 1 iš 10 tikrintų ontologijų. Tačiau išsigilinus į pažeidimą, buvo pastebėta, kad tai specifinė vynu rūšis, todėl įrankis ją traktavo ne kaip daiktavardį, o kaip nežinomą žodį, todėl ekrane išvydome pažeidimo eilutę.

Objektinių savybių vardai taisyklingai t. y. panaudojant struktūrą *veiksmažodis + daiktavardis* buvo suformuoti 6 ontologijose, likusiose 4 ontologijose buvo aptikti vardų formavimo pažeidimai.

Duomenų savybių vardai buvo suformuoti teisingai t. y. panaudojant struktūrą *turi + daiktavardis*, 4 ontologijose, 6 ontologijose buvo rasta pažeidimų.

Patikrinus 10 ontologijų buvo pastebėta, kad 9 ontologijų elementų vardai yra parašyti naudojant gyvatės stilių.

Atlikus eksperimentą paaiškėjo, kad abstrakčių objektų ontologija neturėjo nei vieno pažeidimo. Keletą nežymių pažeidimų turėjo agentų ir laiko ontologijos. Todėl šios ontologijos gali būti transformuojamos į *SBVR* modelį, taip pat jos gali būti naudojamos tolimesniuose ontologijų kūrimo etapuose.

6. REZULTATŲ APIBENDRINIMAS IR IŠVADOS

1. Tiriamajame darbe buvo išskirtos ir formaliai aprašytos devynios ontologijų aprašo tikrinimo taisyklės, pastarosios buvo išskirstytos į keturias grupes: ontologijos elementų vardų formavimo taisyklės, ontologijos savybių formavimo taisyklės, ontologijos anotacijų formavimo taisyklės, ontologijos elementų rašymo stiliaus formavimo taisyklės.
2. Analizės metu buvo nagrinėti esami sprendimai t. y. *OntoCheck* ir *OOPS!*. Nustatyta, kad įrankiai gali tikrinti ontologijos elementų rašymo stilių, savybių sritį ir domeną. Tačiau neatlieka *label_sbvr* anotacijų, ontologijos klasių, objektinių ir duomenų savybių vardų formavimo morfologinės analizės, neinformuoja apie naudojamų klasių skaičių, netikrina kalbos anotacijos egzistavimo. Įrankių perdarymas yra sudėtingas procesas, todėl nuspręsta sukurti naują ontologijų verifikavimo įrankį, kuris remtųsi tiriamojo darbo metu struktūrizuotomis ontologijų tikrinimo taisyklėmis.
3. Tiriamojo darbo metu buvo sukurtas ontologijų verifikavimo įrankis, kuris gali ne tik patikrinti ontologijos struktūrą, elementų pilnumą, bet atlieka ir ontologijos klasių, duomenų ir objektinių savybių vardų struktūros analizę. Įrankis tikrina *label_sbvr* anotacijos egzistavimą ir sudarymo struktūrą, informuoja apie klasių skaičių ir gylį. Ontologijos verifikavimo metu gauti rezultatai gali būti išsaugojami papildomai importuojant tikrintą ir pažeidimų ir klaidų anotacijos ontologijas, tokiu būdu nepažeidžiant tikrintos ontologijos struktūros.
4. Atliktas eksperimentas, su ontologijų verifikavimo įrankiu, parodė, kad ontologijos struktūros išpildymas priklauso nuo tolimesnio ontologijos panaudojimo. Anotacija *label* anglų kalba parašytose ontologijose nėra nurodoma, nes yra laikoma kaip pagrindinė ontologijų modeliavimo kalba. Ontologijų autoriai, kurie kūrė ontologijas nesiekdami jų transformuoti į *SBVR* modelį nenurodo *label_sbvr* anotacijos. Vienas iš dažniausiai aptiktų pažeidimų, tai vieningo stiliaus nesilaikymas, kuris gali daryti neigiama įtaką tolimesniame ontologijos panaudojime. Verifikavimo metu nustatytas, kad ontologijų elementai formuojami laikantis gyvatės stiliaus.
5. Tiriamojo darbo metu sukurtos bendrosios ontologijos aprašo tikrinimo taisyklės ir ontologijų verifikavimo įrankis gali būti pritaikytas kitų ontologijos kūrimo priemonių tyrimui ir tolimesniam ontologijos redagavimui, analizuojant verifikavimo metu nustatytus pažeidimus.

7. LITERATŪRA

- [1] S. W. Standards, „Resource Description Framework,“ W3C, [Tinkle]. Available: <https://www.w3.org/RDF/>. [Kreiptasi 03 balandis 2017].
- [2] B. G. B. M. P. F. Patel-Schneider, „OWL 2 Web Ontology Language“ W3C, 2009. [Tinkle]. Available: <https://www.w3.org/TR/2009/REC-owl2-direct-semantics-20091027/>. [Kreiptasi 03 gegužė 2017].
- [3] D. Vrandečić, „Ontology Evaluation,“ Karlsruher Instituts für Technologie (KIT), 8 Birželis 2010. [Tinkle]. Available: http://simia.net/download/ontology_evaluation.pdf. [Kreiptasi 6 Gegužė 2017].
- [4] T. R. Gruber, „A Translation Approach to Portable Ontology Specifications,“ 1993. [Tinkle]. Available: <http://tomgruber.org/writing/ontolingua-kaj-1993.pdf>. [Kreiptasi 15 kovo 2016].
- [5] Calero, Coral, Ruiz, Francisco, Piattini, Mario (Eds.), *Ontologies for Software Engineering and Software Technology*, Springer, 2006.
- [6] V. University, „Information Science,“ Vilnius University Publishing House, 2003. [Tinkle]. Available: http://www.news.lt/Upload/200402/im_26.pdf. [Kreiptasi 12 12 2015].
- [7] TIM BERNERS-LEE, JAMES HENDLER, ORA LASSILA, „The Semantic Web,“ *Scientific American*, 2001. [Tinkle]. Available: https://www-sop.inria.fr/acacia/cours/essi2006/Scientific%20American_%20Feature%20Article_%20The%20Semantic%20Web_%20May%202001.pdf. [Kreiptasi 10 Sausis 2017].
- [8] Vipul Kashyap, Christoph Bussler, Matthew Moran, *The Semantic Web*, Springer Berlin Heidelberg, 2008.
- [9] Jorge Cardoso, Amit P. Sheth, *Semantic Web Services, Processes and Applications*, Springer US, 2006.
- [10] „W3,“ [Tinkle]. Available: <https://www.w3.org/standards/semanticweb/>. [Kreiptasi 18 sausis 2017].
- [11] L. Yu, *A Developer's Guide to the Semantic Web*, Springer Berlin Heidelberg, 2014.
- [12] „Naming and Addressing,“ w3, 2001. [Tinkle]. Available: <https://www.w3.org/Addressing/>. [Kreiptasi 14 sausio 2017].
- [13] „Concepts and Abstract Syntax,“ w3, 25 vasaris 2014. [Tinkle]. Available: <https://www.w3.org/TR/rdf11-concepts/#dfn-literal>. [Kreiptasi 10 sausis 2017].
- [14] I. Horrocks, P. F. Patel-Schneider, F. van Harmelen., „From SHIQ and RDF to OWL: the making of a web ontology language,“ *Journal of Web Semantics*, p. 7–26, 2003.
- [15] Boris Motik, Ian Horrocks, Zhe Wu, Achille Fokoue, Carsten Lutz, „OWL 2 Web Ontology Language,“ 11 gruodis 2012. [Tinkle]. Available: http://www.w3.org/TR/owl2-profiles/#Class_Expressions. [Kreiptasi 17 sausis 2016].
- [16] J. Z. Pan, I. Horrocks, „OWL-Eu: Adding Customised Datatypes into OWL,“ *Journal of Web Semantics*, pp. 1-20, 2005.
- [17] J. Karpovič, G. Bernotaitytė, L. Ablonskis, L. Nemuratiė, „SUITABILITY OF SEMANTICS OF BUSINESS VOCABULARY AND BUSINESS RULES (SBVR) TO REPRESENT OWL 2 ONTOLOGIES,“ Kaunas.
- [18] Natalya F. Noy, Deborah L. McGuinness, „Ontology Development 101: A Guide to Creating Your First Ontology,“ Stanford University, [Tinkle]. Available: http://protege.stanford.edu/publications/ontology_development/ontology101.pdf. [Kreiptasi 4 gegužė 2017].
- [19] Lina Tankelevičienė, Rimantas Sakalauskas, „DALYKINĖS SRITIES ONTOLOGIJA GRĮSTA INTELEKTUALIOJI GEDIMŲ DIAGNOSTIKOS SISTEMA,“ 2014. [Tinkle]. Available: http://www.su.lt/bylos/mokslo_leidiniai/jmd/08_05_21/14_tankelevicienes.pdf. [Kreiptasi 11 lapkritis 2015].
- [20] D. Schober, I. Tudosse, V. Svatek, M. Boeker, „OntoCheck: verifying ontology naming conventions and metadata completeness in Protégé 4,“ Institute of Medical Biometry and Medical Informatics, [Tinkle]. Available: <https://www.ncbi.nlm.nih.gov/pubmed/23046606>. [Kreiptasi 12 balandis 2017].

- [21] Daniel Schober, Ilinca Tudose, Vojtech Svatek, Martin Boeker, „OntoCheck: verifying ontology naming conventions and metadata completeness in Protégé 4,“ *Journal of Biomedical Semantics*, nr. DOI: 10.1186/2041-1480-3-S2-S4, 2012.
- [22] María Poveda - Villalón, Mari Carmen Suárez - Figueroa, Asunción Gómez - Pérez, „Validating Ontologies with OOPS!,“ Facultad de Informática, Universidad Politécnica de Madrid, [Tinkle]. Available: <http://oa.upm.es/14384/1/mpovedaEtAlCameraReady.pdf#%5B%7B%22num%22%3A71%2C%22gen%22%3A0%7D%2C%7B%22name%22%3A%22XYZ%22%7D%2C122%2C694%2C0%5D>. [Kreiptasi 25 balandis 2017].
- [23] María Poveda-Villalón, Mari Carmen Suárez-Figueroa, Miguel Ángel García-Delgado, Asunción Gómez-Pérez, „Ontology Pitfall Scanner!,“ Ontology Engineering Group Escuela Técnica Superior de Ingenieros Informáticos, Universidad Politécnica de Madrid, Spain, [Tinkle]. Available: http://oa.upm.es/10195/1/OOPS_technical_report_v0.2.pdf. [Kreiptasi 4 gegužė 2017].
- [24] Nicola Guarino, Daniel Oberle, Steffen Staab, „What is an ontology?,“ 2009. [Tinkle]. Available: http://iaoa.org/isc2012/docs/Guarino2009_What_is_an_Ontology.pdf. [Kreiptasi 10 lapkritis 2015].
- [25] W3, „Resource Description Framework,“ 5 sausio 1999. [Tinkle]. Available: <https://www.w3.org/TR/PR-rdf-syntax/>. [Kreiptasi 10 sausio 2017].
- [26] M. Bergman, „The Fundamental Importance of Keeping an ABox and TBox Split,“ [Tinkle]. Available: <http://www.mkbergman.com/489/ontology-best-practices-for-data-driven-applications-part-2/>. [Kreiptasi 10 sausis 2017].
- [27] Dmitry Tsarkov, Ian Horrocks, „FaCT++ Description Logic Reasoner: System Description,“ [Tinkle]. Available: http://link.springer.com/chapter/10.1007%2F11814771_26. [Kreiptasi 12 04 2016].
- [28] Evren Sirin, Bijan Parsia, Bernardo Cuenca Grau, Aditya Kalyanpur, Yarden Katz, „Pellet: A Practical OWL-DL Reasoner,“ [Tinkle]. Available: <http://www.cs.ox.ac.uk/people/bernardo.cuencagrau/publications/PelletDemo.pdf>. [Kreiptasi 2016 04 16].
- [29] K. H. R. M. a. M. W. Volker Haarslev, „The RacerPro Knowledge Representation and Reasoning System,“ [Tinkle]. Available: <http://www.semantic-web-journal.net/content/new-submission-racerpro-knowledge-representation-and-reasoning-system>. [Kreiptasi 12 04 2016].
- [30] M. Koubarakis, „National and Kapodistrian University of Athens,“ [Tinkle]. Available: <http://cgi.di.uoa.gr/~pms547/lectures/intro-to-ontology-development-with-owl2-2spp.pdf>. [Kreiptasi 27 gegužė 2017].
- [31] Aldo Gangemi, Carola Catenacci, Massimiliano Ciaramita, Jos Lehmann, „Modelling Ontology Evaluation and Validation,“ įtraukta *European Semantic Web Conference*, Berlin, Heidelberg, Springer, 2006.
- [32] D. T. S. Strasunskas, „NeOn Methodology for Building Ontology Networks: Specification, Scheduling and Reuse,“ Spain, 2010.
- [33] A. Burton-Jones, V.C. Storey and Sugumaran, and P. Ahluwalia, „A Semiotic Metrics Suite for Assessing the Quality of Ontologies,“ įtraukta *Data and Knowledge Engineering*, 2005.

- [34] „Microsoft,“ [Tinkle]. Available: <https://msdn.microsoft.com/en-us/library/z1zx9t92.aspx>. [Kreiptasi 24 sausis 2017].
- [35] María Poveda-Villalón, Mari Carmen Suárez-Figueroa, „OOPS! - Ontology Pitfalls Scanner!“, Departamento de Inteligencia Artificial Facultad de Informática Universidad Politécnica de Madrid, [Tinkle]. Available: http://oa.upm.es/10195/1/OOPS_technical_report_v0.2.pdf. [Kreiptasi 6 gegužė 2016].

8. PRIEDAI

8.1. priedas. Įspėjimų ir klaidų anotacijų ontologijos failas

```
<?xml version="1.0"?>

<!DOCTYPE rdf:RDF [
  <!ENTITY owl "http://www.w3.org/2002/07/owl#" >
  <!ENTITY xsd "http://www.w3.org/2001/XMLSchema#" >
  <!ENTITY rdfs "http://www.w3.org/2000/01/rdf-schema#" >
  <!ENTITY rdf "http://www.w3.org/1999/02/22-rdf-syntax-ns#" >
]>

<rdf:RDF
xmlns="http://www.semanticweb.org/sonata/ontologies/2017/3/ispejimu_ir_klaidu_annotacija#"
xml:base="http://www.semanticweb.org/sonata/ontologies/2017/3/ispejimu_ir_klaidu_annotacija"
xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
xmlns:owl="http://www.w3.org/2002/07/owl#"
xmlns:xsd="http://www.w3.org/2001/XMLSchema#"
xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
<owl:Ontology
rdf:about="http://www.semanticweb.org/sonata/ontologies/2017/3/ispejimu_ir_klaidu_annotacija"/>

  <!--
  ////////////////////////////////////////////////////////////////////
  //
  // Annotation properties
  //
  ////////////////////////////////////////////////////////////////////
  -->

  <!-- http://www.semanticweb.org/sonata/ontologies/2017/3/ispejimu_ir_klaidu_annotacija#
warning_class_depth_too_deep -->

  <owl:AnnotationProperty
rdf:about="&ispejimu_ir_klaidu_annotacija;warning_class_depth_too_deep"/>

  <!-- http://www.semanticweb.org/sonata/ontologies/2017/3/ispejimu_ir_klaidu_annotacija#
warning_class_not_noun -->

  <owl:AnnotationProperty rdf:about="&ispejimu_ir_klaidu_annotacija;warning_class_not_noun"/>
```



```
<!-- http://www.semanticweb.org/sonata/ontologies/2017/3/ispejimu_ir_klaidu_annotacija#
warning_class_not_well_formed -->

<owl:AnnotationProperty
rdf:about="&ispejimu_ir_klaidu_annotacija;warning_class_not_well_formed"/>

<!-- http://www.semanticweb.org/sonata/ontologies/2017/3/ispejimu_ir_klaidu_annotacija#
warning_class_number_too_big -->

<owl:AnnotationProperty
rdf:about="&ispejimu_ir_klaidu_annotacija;warning_class_number_too_big"/>

<!-- http://www.semanticweb.org/sonata/ontologies/2017/3/ispejimu_ir_klaidu_annotacija#
warning_data_no_child_elements -->

<owl:AnnotationProperty
rdf:about="&ispejimu_ir_klaidu_annotacija;warnig_data_no_child_elements"/>

<!-- http://www.semanticweb.org/sonata/ontologies/2017/3/ispejimu_ir_klaidu_annotacija#
warning_data_not_have_plus_noun -->

<owl:AnnotationProperty
rdf:about="&ispejimu_ir_klaidu_annotacija;warning_data_not_have_plus_noun"/>

<!-- http://www.semanticweb.org/sonata/ontologies/2017/3/ispejimu_ir_klaidu_annotacija#
warning_data_without_domain -->

<owl:AnnotationProperty
rdf:about="&ispejimu_ir_klaidu_annotacija;warning_data_without_domain"/>

<!-- http://www.semanticweb.org/sonata/ontologies/2017/3/ispejimu_ir_klaidu_annotacija#
warning_data_without_range -->

<owl:AnnotationProperty
rdf:about="&ispejimu_ir_klaidu_annotacija;warning_data_without_range"/>

<!-- http://www.semanticweb.org/sonata/ontologies/2017/3/ispejimu_ir_klaidu_annotacija#
warning_domain_without_value -->

<owl:AnnotationProperty
rdf:about="&ispejimu_ir_klaidu_annotacija;warning_domain_without_value"/>
```

```
<!-- http://www.semanticweb.org/sonata/ontologies/2017/3/ispejimu_ir_klaidu_annotacija#  
warning_label_empty -->
```

```
<owl:AnnotationProperty rdf:about="&ispejimu_ir_klaidu_annotacija;warning_label_empty"/>
```

```
<!-- http://www.semanticweb.org/sonata/ontologies/2017/3/ispejimu_ir_klaidu_annotacija#  
warning_label_language_empty -->
```

```
<owl:AnnotationProperty  
rdf:about="&ispejimu_ir_klaidu_annotacija;warning_label_language_empty"/>
```

```
<!-- http://www.semanticweb.org/sonata/ontologies/2017/3/ispejimu_ir_klaidu_annotacija#  
warning_object_no_child_elements -->
```

```
<owl:AnnotationProperty  
rdf:about="&ispejimu_ir_klaidu_annotacija;warning_object_no_child_elements"/>
```

```
<!-- http://www.semanticweb.org/sonata/ontologies/2017/3/ispejimu_ir_klaidu_annotacija#  
warning_object_not_verb_plus_noun -->
```

```
<owl:AnnotationProperty  
rdf:about="&ispejimu_ir_klaidu_annotacija;warning_object_not_verb_plus_noun"/>
```

```
<!-- http://www.semanticweb.org/sonata/ontologies/2017/3/ispejimu_ir_klaidu_annotacija#  
warning_object_without_domain -->
```

```
<owl:AnnotationProperty  
rdf:about="&ispejimu_ir_klaidu_annotacija;warning_object_without_domain"/>
```

```
<!-- http://www.semanticweb.org/sonata/ontologies/2017/3/ispejimu_ir_klaidu_annotacija#  
warning_object_without_range -->
```

```
<owl:AnnotationProperty  
rdf:about="&ispejimu_ir_klaidu_annotacija;warning_object_without_range"/>
```

```
<!-- http://www.semanticweb.org/sonata/ontologies/2017/3/ispejimu_ir_klaidu_annotacija#  
warning_range_without_value -->
```

```
<owl:AnnotationProperty
rdf:about="&ispejimu_ir_klaidu_annotacija;warning_range_without_value"/>
```

```
<!--
http://www.semanticweb.org/sonata/ontologies/2017/3/ispejimu_ir_klaidu_annotacija#warning_style_not_camel_case -->
```

```
<owl:AnnotationProperty
rdf:about="&ispejimu_ir_klaidu_annotacija;warning_style_not_camel_case"/>
```

```
<!-- http://www.semanticweb.org/sonata/ontologies/2017/3/ispejimu_ir_klaidu_annotacija#warning_style_not_snake_case -->
```

```
<owl:AnnotationProperty
rdf:about="&ispejimu_ir_klaidu_annotacija;warning_style_not_snake_case"/>
</rdf:RDF>
```

```
<!-- Generated by the OWL API (version 3.4.2) http://owlapi.sourceforge.net -->
```

8.2. priedas. Politikos ontologijos verifikavimo metu sugeneruotas rezultatų failas

```
<?xml version="1.0" encoding="utf-8"?>
<!DOCTYPE rdf:RDF [
  <!ENTITY terms "http://purl.org/dc/terms/" >
  <!ENTITY vann "http://purl.org/vocab/vann/" >
  <!ENTITY time "http://semantika.lt/ns/Time#" >
  <!ENTITY semlt "http://semantika.lt/ns/SemLT#" >
  <!ENTITY labels "http://semantika.lt/ns/Labels#" >
  <!ENTITY agents "http://semantika.lt/ns/Agents#" >
  <!ENTITY events "http://semantika.lt/ns/Events#" >
  <!ENTITY owl "http://www.w3.org/2002/07/owl#" >
  <!ENTITY swrl "http://www.w3.org/2003/11/swrl#" >
  <!ENTITY dc "http://purl.org/dc/elements/1.1/" >
  <!ENTITY swrlb "http://www.w3.org/2003/11/swrlb#" >
  <!ENTITY locations "http://semantika.lt/ns/Locations#" >
  <!ENTITY xsd "http://www.w3.org/2001/XMLSchema#" >
  <!ENTITY ann "http://semantika.lt/ns/Annotations#" >
  <!ENTITY rdfs "http://www.w3.org/2000/01/rdf-schema#" >
  <!ENTITY abstractobject "http://semantika.lt/ns/AbstractObject#" >
  <!ENTITY mu "http://semantika.lt/ns/MeasurementUnits#" >
  <!ENTITY rdf "http://www.w3.org/1999/02/22-rdf-syntax-ns#" >
]>
<rdf:RDF xmlns="http://semantika.lt/ns/Politics#" xml:base="http://semantika.lt/ns/Politics"
xmlns:dc="http://purl.org/dc/elements/1.1/" xmlns:events="http://semantika.lt/ns/Events#"
```

```

xmlns:abstractobject="http://semantika.lt/ns/AbstractObject#"
xmlns:agents="http://semantika.lt/ns/Agents#" xmlns:labels="http://semantika.lt/ns/Labels#"
xmlns:terms="http://purl.org/dc/terms/" xmlns:vann="http://purl.org/vocab/vann/"
xmlns:mu="http://semantika.lt/ns/MeasurementUnits#" xmlns:rdfs="http://www.w3.org/2000/01/rdf-
schema#" xmlns:time="http://semantika.lt/ns/Time#" xmlns:swrl="http://www.w3.org/2003/11/swrl#"
xmlns:semлт="http://semantika.lt/ns/SemLT#" xmlns:locations="http://semantika.lt/ns/Locations#"
xmlns:swrlb="http://www.w3.org/2003/11/swrlb#" xmlns:owl="http://www.w3.org/2002/07/owl#"
xmlns:xsd="http://www.w3.org/2001/XMLSchema#" xmlns:rdf="http://www.w3.org/1999/02/22-rdf-
syntax-ns#" xmlns:ann="http://semantika.lt/ns/Annotations#">
  <owl:Ontology rdf:about="http://semantika.lt/ns/Politics_errors">
    <owl:imports rdf:resource="http://semantika.lt/ns/Politics" />
    <owl:imports
rdf:resource="http://www.semanticweb.org/sonata/ontologies/2017/3/ispejimu_ir_klaidu_anotacija" />
  </owl:Ontology>
  <owl:Class rdf:about="http://semantika.lt/ns/Events#substance">
    <owl:warning_label_sbvr_is_missing>Class expected at least one
label</owl:warning_label_sbvr_is_missing>
  </owl:Class>
  <owl:ObjectProperty rdf:about="http://semantika.lt/ns/Events#is_meeting_of__agent">
    <owl:warning_label_sbvr_is_missing>Range in label_sbvr is not well
formed</owl:warning_label_sbvr_is_missing>
  </owl:ObjectProperty>
  <owl:ObjectProperty rdf:about="http://semantika.lt/ns/Events#met__agent">
    <owl:warning_label_sbvr_is_missing>Range in label_sbvr is not well
formed</owl:warning_label_sbvr_is_missing>
  </owl:ObjectProperty>
  <owl:ObjectProperty rdf:about="http://semantika.lt/ns/Events#met_in__meeting">
    <owl:warning_label_sbvr_is_missing>Domain in label_sbvr is not well
formed</owl:warning_label_sbvr_is_missing>
  </owl:ObjectProperty>
  <owl:ObjectProperty rdf:about="http://semantika.lt/ns/Events#talked__substance">
    <owl:warning_label_sbvr_is_missing>Range in label_sbvr is not well
formed</owl:warning_label_sbvr_is_missing>
  </owl:ObjectProperty>
  <owl:ObjectProperty rdf:about="http://semantika.lt/ns/Events#talked__talking">
    <owl:warning_label_sbvr_is_missing>Domain in label_sbvr is not well
formed</owl:warning_label_sbvr_is_missing>
  </owl:ObjectProperty>
  <owl:ObjectProperty rdf:about="http://semantika.lt/ns/Events#talked_by__speaker">
    <owl:warning_label_sbvr_is_missing>Range in label_sbvr is not well
formed</owl:warning_label_sbvr_is_missing>
  </owl:ObjectProperty>
  <owl:ObjectProperty rdf:about="http://semantika.lt/ns/Politics#has__emotional_type">
    <owl:warning_label_sbvr_is_missing>Domain in label_sbvr is not well
formed</owl:warning_label_sbvr_is_missing>
  </owl:ObjectProperty>
  <owl:ObjectProperty rdf:about="http://semantika.lt/ns/Politics#has__talking_object">
    <owl:warning_label_sbvr_is_missing>Range in label_sbvr is not well
formed</owl:warning_label_sbvr_is_missing>
  </owl:ObjectProperty>
  <owl:ObjectProperty rdf:about="http://semantika.lt/ns/Politics#has__talking_type">
    <owl:warning_label_sbvr_is_missing>Domain in label_sbvr is not well
formed</owl:warning_label_sbvr_is_missing>

```



```

    <owl:warning_style_not_snake_case>Expected snake case:
AtomList</owl:warning_style_not_snake_case>
  </rdf:Description>
</rdf:Description>
    <owl:warning_style_not_snake_case>Expected snake case:
DifferentIndividualsAtom</owl:warning_style_not_snake_case>
  </rdf:Description>
</rdf:Description>
    <owl:warning_style_not_snake_case>Expected snake case:
IndividualPropertyAtom</owl:warning_style_not_snake_case>
  </rdf:Description>
</rdf:Description>
    <owl:warning_style_not_snake_case>Expected snake case:
IndividualPropertyAtom</owl:warning_style_not_snake_case>
  </rdf:Description>
</rdf:Description>
    <owl:warning_style_not_snake_case>Expected snake case:
AtomList</owl:warning_style_not_snake_case>
  </rdf:Description>
</rdf:Description>
    <owl:warning_style_not_snake_case>Expected snake case:
IndividualPropertyAtom</owl:warning_style_not_snake_case>
  </rdf:Description>
</rdf:RDF>
<!-- Generated by the OWL API (version 3.4.2) http://owlapi.sourceforge.net -->

```

8.3. priedas. Tikrintų ontologijų detali statistika

| Ontologija: http://semantika.lt/ns/Time | | | |
|---|--------------------------|-------------------------|--|
| Ontologijos tikrinimo taisyklė | Rastų pažeidimų skaičius | Rezultatų atvaizdavimas | |
| | | Eilutės nr. | Rastas pažeidimas |
| Kiekvienas ontologijos elementas turi turėti anotaciją. | 1 | 89 | <i>Datatype Property</i> expected at least one label |
| struktūros: <i>objektinės savybės sritis</i> + <i>veiksmažodis</i> + <i>objektinės savybės domenai</i> . | - | - | - |
| Kiekviena ontologijos elemento anotacija turi turėti kalbą | 1* | 89 | <i>DatatypeProperty</i> expected to have at least one language label |
| Kiekvienai objektiniai savybei (angl. <i>Object Property</i>) turi būti specifikuotas domenai (angl. <i>Domain</i>) ir sritis (angl. <i>Range</i>) | - | - | - |
| Kiekvienai duomenų savybei (angl. <i>Datatype Property</i>) turi būti specifikuotas domenai (angl. <i>Domain</i>) ir sritis (angl. <i>Range</i>) | - | - | - |
| Kiekvienos ontologijos klasės vardas turi būti daiktavardis | - | - | - |
| Kiekvienos objektinės savybės (angl. <i>Object Property</i>) vardas turi atitikti struktūrą: <i>veiksmažodis</i> + <i>daiktavardis</i> | - | - | - |

| | | | |
|---|-----|--|--|
| Kiekvienos duomenų savybės (angl. <i>Data Property</i>) vardas turi atitikti struktūrą: <i>turi</i> + <i>daiktavardis</i> | 1 | 89 | <i>Time_value</i> should be have/has/had + noun |
| Klasių skaičius | - | 0 | <i>Class</i> count: 5 |
| Klasių gylis | - | 0 | Ontology depth: 2 |
| Ontologijos elementų vardai, kuriuos sudaro keli žodžiai, turi būti užrašyti: gyvatės stiliumi | 1 | 91 | Expected snake case: <i>dateTime</i> |
| Ontologijos elementų vardai, kuriuos sudaro keli žodžiai, turi būti užrašyti: kupranugario stiliumi. | *** | - | - |
| Ontologija: http://www.w3.org/TR/2003/CR-owl-guide-20030818/wine | | | |
| Kiekvienas ontologijos elementas turi turėti anotaciją. | 86 | 86; 95; 99; 109; 119; 136; 143; 160; 167; 182; 189; 205; 219; 109; 95; 284; 298; 339; 341; 349; 353; 362; 371; 380; 391; 401; 416; 426; 436; 466; 500; 536; 550; 564; 587; 627; 637; 657; 667; 677; 696; 716; 744; 764; 784; 836; 870; 910; 926; 978; 1012; 1034; 1066; 1076; 1100; 1112; 1149; 1159; 1173; 1185; 1195; 1212; 1222; 1229; 1264; 1271; 1288; 1304; 1351; 1403; 1457; 1467; 1477; 1529; 1567; 1591; 1601; 1649; 1683; 1693 | <i>Class</i> expected at least one label |
| | | 257; 263; 207; 21; 276; 280; 246; 251; 236; 242; 269; 230 | <i>Object Property</i> expected at least one label |

| | | | |
|---|------|--|--|
| | | 221 | <i>Data Property</i> expected at least one label |
| Kiekvienoje objektinėje savybėje esanti anotacija <i>label_sbvr</i> , turi būti tokios struktūros: <i>objektinės savybės sritis + veiksmožodis + objektinės savybės domenai</i> . | 12 | 207; 21; 230; 236; 242; 246; 251; 257; 263; 269; 276; 280 | <i>ObjectProperty</i> is missing <i>label_sbvr</i> |
| Kiekviena ontologijos elemento anotacija turi turėti kalbą | 86** | 86; 95; 99; 109; 119; 136; 143; 160; 167; 182; 189; 205; 219; 109; 95; 284; 298; 339; 341; 349; 353; 362; 371; 380; 391; 401; 416; 426; 436; 466; 500; 536; 550; 564; 587; 627; 637; 657; 667; 677; 696; 716; 744; 764; 784; 836; 870; 910; 926; 978; 1012; 1034; 1066; 1076; 1100; 1112; 1149; 1159; 1173; 1185; 1195; 1212; 1222; 1229; 1264; 1271; 1288; 1304; 1351; 1403; 1457; 1467; 1477; 1529; 1567; 1591; 1601; 1649; 1683; 1693 | <i>DatatypeProperty</i> expected to have at least one language label |
| | | 213; 242; 230; 246; 251; 257; 263; 269; 276; 207; 26; 280 | <i>ObjectProperty</i> expected to have at least one language label |
| | | 221 | <i>DatatypeProperty</i> expected to have at least one language label |
| Kiekvienai objektiniai savybei (angl. <i>Object Property</i>) turi būti specifikuotas domenai (angl. <i>Domain</i>) ir sritis (angl. <i>Range</i>) | 9 | 242; 263; 251; 257; 276; 280 | <i>ObjectProperty</i> is missing domain |
| | | 242; 276; 280 | <i>ObjectProperty</i> is missing range |
| Kiekvienai duomenų savybei (angl. <i>Datatype Property</i>) turi būti specifikuotas domenai (angl. <i>Domain</i>) ir sritis (angl. <i>Range</i>) | - | - | - |

| | | | |
|---|-----|--|--|
| Kiekvienos ontologijos klasės vardas turi būti daiktavardis | - | - | - |
| Kiekvienos objektinės savybės (angl. <i>Object Property</i>) vardas turi atitikti struktūrą: <i>veiksmožodis + daiktavardis</i> | 1 | 2133 | <i>adjacentRegion</i> should be verb + noun |
| Kiekvienos duomenų savybės (angl. <i>Data Property</i>) vardas turi atitikti struktūrą: <i>turi + daiktavardis</i> | 1 | 221 | <i>yearValue</i> should be have or has or had + noun |
| Klasių skaičius | - | 0 | <i>Class count: 79</i> |
| Klasių gylis | - | 0 | <i>Ontology depth: 4</i> |
| Ontologijos elementų vardai, kuriuos sudaro keli žodžiai, turi būti užrašyti: gyvatės stiliumi. | *** | - | - |
| Ontologijos elementų vardai, kuriuos sudaro keli žodžiai, turi būti užrašyti: kupranugario stiliumi. | - | - | - |
| Ontologija: http://semantika.lt/ns/PublicAdministrative | | | |
| Kiekvienas ontologijos elementas turi turėti anotaciją. | - | - | - |
| Kiekvienoje objektinėje savybėje esanti anotacija <i>label_sbvr</i> , turi būti tokios struktūros: <i>objektinės savybės sritis + veiksmožodis + objektinės savybės domenai</i> . | 18 | 68; 108; 172; 187; 201; 226; 239; 251; 264 | Domain is not well formed |
| | | 82; 94; 121; 134; 147; 160; 201; 226; 239 | Range is not well formed |
| Kiekviena ontologijos elemento anotacija turi turėti kalbą | - | - | - |
| Kiekvienai objektiniai savybei (angl. <i>Object Property</i>) turi būti specifikuotas domenai (angl. <i>Domain</i>) ir sritis (angl. <i>Range</i>) | - | - | - |
| Kiekvienai duomenų savybei (angl. <i>Datatype Property</i>) turi būti specifikuotas domenai (angl. <i>Domain</i>) ir sritis (angl. <i>Range</i>) | - | - | - |
| Kiekvienos ontologijos klasės vardas turi būti daiktavardis | - | - | - |
| Kiekvienos objektinės savybės (angl. <i>Object Property</i>) vardas turi atitikti struktūrą: <i>veiksmožodis + daiktavardis</i> | - | - | - |
| Kiekvienos duomenų savybės (angl. <i>Data Property</i>) vardas turi atitikti struktūrą: <i>turi + daiktavardis</i> | - | - | - |
| Klasių skaičius | - | 0 | <i>Class count: 6</i> |
| Klasių gylis | - | 0 | <i>Ontology depth: 2</i> |
| Ontologijos elementų vardai, kuriuos sudaro keli žodžiai, turi būti užrašyti: gyvatės stiliumi. | - | - | - |
| Ontologijos elementų vardai, kuriuos sudaro keli žodžiai, turi būti užrašyti: kupranugario stiliumi. | *** | - | - |
| Ontologija: http://rdfs.org/sioc/ns | | | |

| | | | |
|---|-----|---|--|
| Kiekvienas ontologijos elementas turi turėti anotaciją. | - | - | - |
| Kiekvienoje objektinėje savybėje esanti anotacija <i>label_sbvr</i> , turi būti tokios struktūros: <i>objektinės savybės sritis + veiksmazodis + objektinės savybės domenai</i> . | 49 | 163; 170; 179; 187; 196; 203; 211; 228; 245; 260; 268; 274; 28; 291; 300; 309; 317; 325; 333; 342; 352; 361; 370; 378; 387; 98; 406; 414; 424; 433; 491; 499; 505; 512; 521; 530; 546; 555; 608; 616; 625; 634; 644; 651; 662; 678; 686; 696; 703 | <i>ObjectProperty</i> is missing <i>label_sbvr</i> |
| Kiekviena ontologijos elemento anotacija turi turėti kalbą | 2** | 449; 457 | <i>DatatypeProperty</i> is missing selected language |
| Kiekvienai objektiniai savybei (angl. <i>Object Property</i>) turi būti specifikuotas domenai (angl. <i>Domain</i>) ir sritis (angl. <i>Range</i>) | 9 | 179; 163; 196; 228; 245; 203; 696; 268; 283; 499; 317; 505; 608; 398; 644; 678; 268; 370; 406; 499; 644; 309; 662; 505; 696; 325 | <i>ObjectProperty</i> is missing range |
| Kiekvienai duomenų savybei (angl. <i>Datatype Property</i>) turi būti specifikuotas domenai (angl. <i>Domain</i>) ir sritis (angl. <i>Range</i>) | - | 442; 457; 474; 539; 565; 572; 587; 601 | <i>DatatypeProperty</i> is missing domain |
| | | 594 | <i>DatatypeProperty</i> is missing range |
| Kiekvienos ontologijos klasės vardas turi būti daiktavardis | - | - | - |
| Kiekvienos objektinės savybės (angl. <i>Object Property</i>) vardas turi atitikti struktūrą: <i>veiksmazodis + daiktavardis</i> | 30 | 163; 170; 179; 187; 196; 203; 211; 228; 245; 268; 274; 283; 433; 491; 512; 521; 530; 546; 555; 608; 616; 625; 634; 644; 651; 662; 678; 686; 696; 703 | <i>Name of Object Property</i> should be verb + noun |
| Kiekvienos duomenų savybės (angl. <i>Data Property</i>) vardas turi atitikti struktūrą: <i>turi + daiktavardis</i> | 12 | 220; 252; 442; 449; 465; 539; 565; 572; 579; 587; 594; 601 | <i>Name of Datatype Property</i> should be have or has or had + noun |

| | | | |
|---|-----|-----------------------------------|---|
| | | | |
| Klasių skaičius | 0 | 0 | <i>Class count: 11</i> |
| Klasių gylis | 0 | 0 | <i>Class depth: 2</i> |
| Ontologijos elementų vardai, kuriuos sudaro keli žodžiai, turi būti užrašyti: gyvatės stiliumi. | - | - | - |
| Ontologijos elementų vardai, kuriuos sudaro keli žodžiai, turi būti užrašyti: kupranugario stiliumi. | *** | - | - |
| Ontologija: http://semantika.lt/ns/PersonNames | | | |
| Kiekvienas ontologijos elementas turi turėti anotaciją. | 15 | 62; 74; 86; 98 | <i>ObjectProperty</i> expected at least one label |
| | | 121; 132; 143; 154 | <i>DatatypeProperty</i> expected to have at least one label |
| | | 176; 196; 219; 238; 257; 279; 301 | <i>Class</i> expected to have at least one label |
| Kiekvienoje objektinėje savybėje esanti anotacija <i>label_sbvr</i> , turi būti tokios struktūros: <i>objektinės savybės sritis + veiksmažodis + objektinės savybės domenai</i> . | 4 | 62; 74; 86; 98 | <i>ObjectProperty</i> is missing label_sbvr |
| Kiekviena ontologijos elemento anotacija turi turėti kalbą | 15* | 62; 74; 86; 98 | <i>ObjectProperty</i> is missing selected language (lt) |
| | | 121; 132; 143; 154 | <i>DatatypeProperty</i> is missing selected language (lt) |
| | | 176; 196; 219; 238; 257; 279; 301 | <i>Class</i> is missing selected language (lt) |
| Kiekvienai objektiniai savybei (angl. <i>Object Property</i>) turi būti specifikuotas domenai (angl. <i>Domain</i>) ir sritis (angl. <i>Range</i>) | - | - | - |
| Kiekvienai duomenų savybei (angl. <i>Datatype Property</i>) turi būti specifikuotas domenai (angl. <i>Domain</i>) ir sritis (angl. <i>Range</i>) | 1 | 89 | |
| Kiekvienos ontologijos klasės vardas turi būti daiktavardis | - | - | - |
| Kiekvienos objektinės savybės (angl. <i>Object Property</i>) vardas turi atitikti struktūrą: <i>veiksmažodis + daiktavardis</i> | 1 | 98 | <i>Language</i> only one word, should be verb + noun - |

| | | | |
|---|-----|-------------------------|--|
| Kiekvienos duomenų savybės (angl. <i>Data Property</i>) vardas turi atitikti struktūrą: <i>turi</i> + <i>daiktavardis</i> | 4 | 121 | <i>Country</i> only one word, should be have/has/had + noun |
| | | 132 | <i>Full_name</i> should be have or has or had + noun |
| | | 143 | <i>Industry</i> only one word, should be have/has/had + noun |
| | | 154 | <i>Lithuanian</i> only one word, should be have/has/had + noun |
| Klasių skaičius | - | 0 | <i>Class count: 7</i> |
| Klasių gylis | - | 0 | <i>Ontology depth: 2</i> |
| Ontologijos elementų vardai, kuriuos sudaro keli žodžiai, turi būti užrašyti: gyvatės stiliumi. | 11 | 63; 75; 87; 99 | Expected snake case: <i>FunctionalProperty</i> |
| | | 67; 79; 103; 244; 279 | Expected snake case: <i>Person_Name</i> |
| | | 78; 219 | Expected snake case: <i>NameKind</i> |
| Ontologijos elementų vardai, kuriuos sudaro keli žodžiai, turi būti užrašyti: kupranugario stiliumi. | *** | - | - |
| Ontologija: http://semantika.lt/ns/Events | | | |
| Kiekvienas ontologijos elementas turi turėti anotaciją. | 2 | 853, 1186 | <i>Class</i> expected to have at least one label |
| Kiekvienoje objektinėje savybėje esanti anotacija <i>label_sbvr</i> , turi būti tokios struktūros: <i>objektinės savybės sritis</i> + <i>veiksmazodis</i> + <i>objektinės savybės domenai</i> . | 10 | 137; 149; 163; 201; 227 | <i>Domain</i> is not well formed |
| | | 163; 176; 187; 201; 215 | <i>Range</i> is not well formed |
| Kiekviena ontologijos elemento anotacija turi turėti kalbą | 2* | 853; 1186 | <i>Class</i> expected to have at least one language label |
| Kiekvienai objektiniai savybei (angl. <i>Object Property</i>) turi būti specifiкуotas domenai (angl. <i>Domain</i>) ir sritis (angl. <i>Range</i>) | 2 | 275 | <i>ObjectProperty</i> is missing <i>domain</i> |
| | | 275 | <i>ObjectProperty</i> is missing <i>range</i> |

| | | | |
|---|-----|---------------|---|
| Kiekvienai duomenų savybei (angl. <i>Datatype Property</i>) turi būti specifikuotas domenas (angl. <i>Domain</i>) ir sritis (angl. <i>Range</i>) | - | - | - |
| Kiekvienos ontologijos klasės vardas turi būti daiktavardis | - | - | - |
| Kiekvienos objektinės savybės (angl. <i>Object Property</i>) vardas turi atitikti struktūrą: <i>veiksmažodis + daiktavardis</i> | 5 | 198 | <i>has__position_held</i> should be verb + noun |
| | | 353 | <i>is_assigned_to__position_held</i> should be verb + noun |
| | | 612 | <i>provide__position_held</i> should be verb + noun |
| | | 739 | <i>representation_in__event</i> should be verb + noun |
| | | 751 | <i>representation_of__organization</i> should be verb + noun |
| Kiekvienos duomenų savybės (angl. <i>Data Property</i>) vardas turi atitikti struktūrą: <i>turi + daiktavardis</i> | 3 | 798 | <i>size_decimal_value</i> should be have or has or had + noun |
| | | 810 | <i>size_integer_value</i> should be have or has or had + noun |
| | | 822 | <i>size_string_value</i> should be have or has or had + noun |
| Klasių skaičius | - | 0 | Class count: 26 |
| Klasių gylis | - | 0 | Ontology depth: 2 |
| Ontologijos elementų vardai, kuriuos sudaro keli žodžiai, turi būti užrašyti: gyvatės stiliumi | 3 | 405; 521; 588 | Expected snake case: <i>FunctionalProperty</i> |
| Ontologijos elementų vardai, kuriuos sudaro keli žodžiai, turi būti užrašyti: kupranugario stiliumi. | *** | - | - |
| Ontologija: http://semantika.lt/ns/Locations | | | |
| Kiekvienas ontologijos elementas turi turėti anotaciją. | - | - | - |
| Kiekvienoje objektinėje savybėje esanti anotacija <i>label_sbvr</i> , turi būti tokios struktūros: <i>objektinės savybės sritis + veiksmažodis + objektinės savybės domenai</i> . | 3 | 76; 113 | <i>Range</i> is not well formed |
| | | 89 | <i>Domain</i> is not well formed |
| Kiekviena ontologijos elemento anotacija turi turėti kalbą | - | - | - |
| Kiekvienai objektiniai savybei (angl. <i>Object Property</i>) turi būti specifikuotas domenas (angl. <i>Domain</i>) ir sritis (angl. <i>Range</i>) | - | - | - |

| | | | |
|---|-----|------|---|
| Kiekvienai duomenų savybei (angl. <i>Datatype Property</i>) turi būti specifikuotas domenas (angl. <i>Domain</i>) ir sritis (angl. <i>Range</i>) | - | - | - |
| Kiekvienos ontologijos klasės vardas turi būti daiktavardis | - | - | - |
| Kiekvienos objektinės savybės (angl. <i>Object Property</i>) vardas turi atitikti struktūrą: <i>veiksmažodis + daiktavardis</i> | - | - | - |
| Kiekvienos duomenų savybės (angl. <i>Data Property</i>) vardas turi atitikti struktūrą: <i>turi + daiktavardis</i> | - | - | - |
| Klasių skaičius | - | 0 | <i>Class count: 13</i> |
| Klasių gylis | - | 0 | <i>Ontology depth: 2</i> |
| Ontologijos elementų vardai, kuriuos sudaro keli žodžiai, turi būti užrašyti: gyvatės stiliumi | - | - | - |
| Ontologijos elementų vardai, kuriuos sudaro keli žodžiai, turi būti užrašyti: kurpranugario stiliumi | *** | - | - |
| Ontologija: http://semantika.lt/ns/Agents | | | |
| Kiekvienas ontologijos elementas turi turėti anotaciją. | 1 | 1066 | <i>Class expected to have at least one label</i> |
| Kiekvienoje objektinėje savybėje esanti anotacija <i>label_sbvr</i> , turi būti tokios struktūros: <i>objektinės savybės sritis + veiksmažodis + objektinės savybės domenai</i> . | - | - | - |
| Kiekviena ontologijos elemento anotacija turi turėti kalbą | 1* | 1066 | <i>Class expected to have at least one language label</i> |
| Kiekvienai objektiniai savybei (angl. <i>Object Property</i>) turi būti specifikuotas domenas (angl. <i>Domain</i>) ir sritis (angl. <i>Range</i>) | - | - | - |
| Kiekvienai duomenų savybei (angl. <i>Datatype Property</i>) turi būti specifikuotas domenas (angl. <i>Domain</i>) ir sritis (angl. <i>Range</i>) | - | - | - |
| Kiekvienos ontologijos klasės vardas turi būti daiktavardis | - | - | - |
| Kiekvienos objektinės savybės (angl. <i>Object Property</i>) vardas turi atitikti struktūrą: <i>veiksmažodis + daiktavardis</i> | - | - | - |

| | | | |
|---|-----|--------------|---|
| Kiekvienos duomenų savybės (angl. <i>Data Property</i>) vardas turi atitikti struktūrą: <i>turi + daiktavardis</i> | - | - | - |
| Klasių skaičius | - | 0 | Class count: 49 |
| Klasių gylis | - | 0 | Ontology depth: 2 |
| Ontologijos elementų vardai, kuriuos sudaro keli žodžiai, turi būti užrašyti: gyvatės stiliumi | 3 | 92, 105, 118 | Expected snake case: <i>topObjectProperty</i> |
| Ontologijos elementų vardai, kuriuos sudaro keli žodžiai, turi būti užrašyti: kurpranugario stiliumi | *** | - | - |
| Ontologija: http://semantika.lt/ns/AbstractObject | | | |
| Kiekvienas ontologijos elementas turi turėti anotaciją. | - | - | - |
| Kiekvienoje objektinėje savybėje esanti anotacija <i>label_sbvr</i> , turi būti tokios struktūros: <i>objektinės savybės sritis + veiksmažodis + objektinės savybės domenas</i> . | - | - | - |
| Kiekviena ontologijos elemento anotacija turi turėti kalbą | - | - | - |
| Kiekvienai objektiniai savybei (angl. <i>Object Property</i>) turi būti specifikuotas domenas (angl. <i>Domain</i>) ir sritis (angl. <i>Range</i>) | - | - | - |
| Kiekvienai duomenų savybei (angl. <i>Datatype Property</i>) turi būti specifikuotas domenas (angl. <i>Domain</i>) ir sritis (angl. <i>Range</i>) | - | - | - |
| Kiekvienos ontologijos klasės vardas turi būti daiktavardis | - | - | - |
| Kiekvienos objektinės savybės (angl. <i>Object Property</i>) vardas turi atitikti struktūrą: <i>veiksmažodis + daiktavardis</i> | - | - | - |
| Kiekvienos duomenų savybės (angl. <i>Data Property</i>) vardas turi atitikti struktūrą: <i>turi + daiktavardis</i> | - | - | - |
| Klasių skaičius | - | 0 | Class count: 17 |
| Klasių gylis | - | 0 | Ontology depth: 2 |

| | | | |
|---|-----|-----|--|
| Ontologijos elementų vardai, kuriuos sudaro keli žodžiai, turi būti užrašyti: gyvatės stiliumi | - | - | - |
| Ontologijos elementų vardai, kuriuos sudaro keli žodžiai, turi būti užrašyti: kupranugario stiliumi | *** | - | - |
| Ontologija: http://semantika.lt/voc/SemLT | | | |
| Kiekvienas ontologijos elementas turi turėti anotaciją. | - | - | - |
| Kiekvienoje objektinėje savybėje esanti anotacija <i>label_sbvr</i> , turi būti tokios struktūros: <i>objektinės savybės sritis + veiksmožodis + objektinės savybės domenai</i> . | 1 | 76 | <i>Verb</i> was not used in <i>label_sbvr</i> |
| Kiekviena ontologijos elemento anotacija turi turėti kalbą | - | - | - |
| Kiekvienai objektiniai savybei (angl. <i>Object Property</i>) turi būti specifikuotas domenai (angl. <i>Domain</i>) ir sritis (angl. <i>Range</i>) | - | - | - |
| Kiekvienai duomenų savybei (angl. <i>Datatype Property</i>) turi būti specifikuotas domenai (angl. <i>Domain</i>) ir sritis (angl. <i>Range</i>) | - | - | - |
| Kiekvienos ontologijos klasės vardas turi būti daiktavardis | - | - | - |
| Kiekvienos objektinės savybės (angl. <i>Object Property</i>) vardas turi atitikti struktūrą: <i>veiksmožodis + daiktavardis</i> | - | - | - |
| Kiekvienos duomenų savybės (angl. <i>Data Property</i>) vardas turi atitikti struktūrą: <i>turi + daiktavardis</i> | 7 | 148 | <i>annotation_date</i> should be have or has or had + noun |
| | | 161 | <i>created_date</i> should be have or has or had + noun |
| | | 174 | <i>document_ID</i> should be have or has or had + noun |
| | | 187 | <i>document_URI</i> should be have or has or had + noun |
| | | 199 | <i>is_abstract</i> should be have or has or had + noun |
| | | 211 | <i>source_name</i> should be have or has or had + noun |

| | | | |
|---|-----|-----|---|
| | | 223 | <i>trusted</i> only one word, should be have/has/had + noun |
| Klasių skaičius | - | 0 | Class count: 4 |
| Klasių gylis | - | 0 | Ontology depth: 2 |
| Ontologijos elementų vardai, kuriuos sudaro keli žodžiai, turi būti užrašyti: gyvatės stiliumi | 7 | 72 | Expected snake case: <i>FunctionalProperty</i> |
| | | 99 | Expected snake case: <i>SymmetricProperty</i> |
| | | 154 | Expected snake case: <i>dateTime</i> |
| | | 167 | Expected snake case: <i>dateTime</i> |
| | | 174 | Expected snake case: <i>document_ID</i> |
| | | 187 | Expected snake case: <i>document_URI</i> |
| | | 192 | Expected snake case: <i>anyURI</i> |
| Ontologijos elementų vardai, kuriuos sudaro keli žodžiai, turi būti užrašyti: kupranugario stiliumi | *** | - | - |