

LATVIJAS UNIVERSITĀTE  
DATORIKAS FAKULTĀTE

# GLOSĀRIJU INFORMĀCIJAS SISTĒMA

BAKALaura DARBS

Autors: **Artūrs Bugins**

Studenta apliecības Nr.: ab12128

Darba vadītājs: profesors Dr. dat. Guntis Arnicāns

RĪGA 2016

## ANOTĀCIJA

Glosārijs ir alfabētisks saraksts ar terminiem vai vārdiem un to skaidrojumiem, kas atrodami vai ir attiecināmi uz kādu konkrētu priekšmetu, tekstā vai dialektā. Bakalaura darbā tiek pētīta un aprakstīta glosāriju sistēmas izveide, kas spēj uzturēt glosārijus un to terminus. Tiek apskatīti un analizēti dažādi glosāriju pieraksta formāti un tīmeklī pieejamie risinājumi, kā rezultātā tiek iegūti dati par glosārija datu struktūru, izstrādāta datu bāze vadoties pēc iegūtās informācijas un izstrādāta strādājoša glosāriju informāciju sistēma. Sistēma ļauj pievienot un meklēt terminus, veidot jaunus glosārijus, importēt tos no tīmekļa resursiem, kā arī eksportēt tos .DOC un .PDF formātos.

**Atslēgvārdi:** glosārijs, vārds, frāze, sistēma, termins, pieraksta formāti, datu struktūru, eksportēt, importēt.

## **ABSTRACT**

### **Glossary information system**

Glossary is an alphabetical list of terms or words found in or relating to a specific subject, text, or dialect, with explanations. This paper describes development of a glossary system that is be able to maintain glossaries and terminology. Different glossary recording formats and web-based solutions are examined and analyzed. In result data about glossary data structures is attained. Afterwards a data base and a working glossary information system is developed accordingly to the attained data. Developed system allows to add and search terms, to create new glossaries, import them from web resources, as well as export them in .DOC and .PDF formats.

**Keywords:** glossary, words, phrases, system, terms, recording formats, data structures, export, import.

# SATURS

Apzīmējumu saraksts.....	6
Ievads.....	7
1. Glosāriju pieraksta datu struktūras izpēte.....	8
1.1. Glosārijs.....	8
1.2. Termins.....	8
1.3. Strukturēta terminoloģija.....	10
1.4. TBX, MARTIF un OLIF standarti .....	11
1.4.1. TBX pieraksta formāts.....	11
1.4.2. MARTIF pieraksta formāts .....	13
1.4.3. OLIF pieraksta formāts.....	14
1.5. TBX, MARTIF un OLIF kopskats .....	15
1.6. TMF – Terminu marķēšanas ietvars .....	15
1.7. GlossML pieraksta formāts .....	17
1.7.1. GlossML lietojums .....	17
1.7.2. GlossML struktūra.....	17
1.7.3. GlossML piemērs .....	19
1.8. XML formāta glosāriji.....	20
1.8.1. GlossXML elementu lietojums.....	20
1.8.2. GlossXML lietošanas piemērs.....	22
1.9. Secinājumi par glosāriju pieraksta valodu.....	22
1.10. Esošo tīmeklī pieejamo glosāriju un vārdnīcu izpēte .....	23
1.10.1. AkadTerm un Tēzaurs .....	24
1.10.2. Wikipedia .....	25
1.10.3. WhatIs, bis.org un Academic .....	26
1.10.4. Trūkumi un secinājumi.....	27
2. Datu bāzes modelis glosāriju IS .....	28
2.1. Datu bāzes modelis.....	28
3. Glosāriju IS izstrāde .....	31
3.1. Glosāriju izveide, pārvaldīšana un lietotāju autentifikācija.....	31
3.2. Glosārija eksportēšana.....	32
3.2.1. Glosārija eksportēšana .DOC formātā.....	32
3.2.2. Glosārija eksportēšana pdf formātā .....	35
3.3. Glosāriju importēšana.....	36

3.3.1. DOM satura koncepcija .....	37
3.3.2. DOM satura apstrāde .....	38
3.4. Terminu pievienošana un meklēšana.....	40
Rezultāti un secinājumi .....	41
Izmantotā literatūra un avoti.....	43
Pielikumi.....	45
1. pielikums. Manuāla glosārija pievienošanas forma.....	45
2. pielikums. Terminu atlasīšana glosārijam .....	46
3. pielikums. Pievienoto terminu pārskats.....	47
4. pielikums. Glosārija pārskats pirms eksportēšanas .....	47
5. pielikums. Terminu pievienošana. Kreisā puse .....	48
6. pielikums. Terminu pievienošana. Labā puse .....	48
7. pielikums. Terminu meklētājs .....	49

## APZĪMĒJUMU SARAKSTS

**MySQL** -relāciju datu bāzu vadības sistēma, kas nodrošina datu glabāšanu.

**.DOC** - Microsoft Office Word lietojumprogrammas faila paplašinājums.

**.PDF** - failu formāts, kas tiek lietots dokumentu atspoguļošanai neatkarīgi no operētājsistēmas, programmatūras un aparatūras.

**.ODT** - atvērtā dokumenta formāts, kas bāzēts uz XML.

**TBX** - terminu bāzes apmaiņas formāts.

**MARTIF** - mašīnlasāmās terminoloģijas apmaiņas formāts.

**OLIF** - atvērtās leksikas apmaiņas formāts.

**GlossML** - glosārija pieraksta formāts bāzēts uz XML valodas.

**XML** - paplašināmā iezīmēšanas valoda.

**PHP** - atklātā pirmkoda skriptu valoda, kura sākotnēji bija paredzēta servera puses lietojumos dinamiska tīmekļa lapu ģenerēšanai.

**Laravel** - ietvars uz kura var tikt izstrādātas sistēmas. Bāzēts uz PHP programmēšanas valodas.

**CRUD** - datu bāzes vaicājumu grupa: izveidošana, lasīšana, atjaunošana un dzēšana.

**HTTP** - protokols datu pārraidīšanai tīmeklī.

**Unix** -pārnesama, daudz uzdevumu un daudzlietotāju operētājsistēma.

**Windows** - operētājsistēma personālajiem datoriem un serveriem.

**MS Word** - lietojumprogramma, kurā iespējams veidot un apstrādāt tekstuālus dokumentus.

**COM** - tehnoloģija, kas ļauj objektiem mijiedarboties ar procesiem un datoru komponentēm.

**API** - lietojumprogrammu programmēšanas interfeiss. Satur maršrutēšanu un protokolus lietojumprogrammu izveidei.

**DOM** - dokumenta objektu modelis, kas reprezentē ar objektiem HTML, XHTML un XML dokumentus.

**HTML** - hiperteksta iezīmēšanas valoda, kas izstrādāta tīmekļa vietņu informācijas attēlošanai.

**IS** – informāciju sistēma.

## IEVADS

Katrā zināšanu sfērā jeb nozarē tiek formulēti termini, kas apzīmē kādu konkrētu procesu vai darbību. Ja cilvēks nav pazīstams ar šiem terminiem, tad nav iespējams izprast nozares būtību un pētāmās problēmas. Tāpēc tiek veidoti glosāriji konkrētām zināšanu sfērām, kas satur terminus un to skaidrojumus.

Bakalaura darbā tika izvirzīts uzdevums – izveidot glosāriju informāciju sistēmu. Sistēma paredzēta glosāriju glabāšanai, veidošanai un pārvaldīšanai. Realizējot sistēmu tika pētītas glosāriju pieraksta formas, kas nepieciešams korektas datu struktūras izveidei. Sistēmā paredzēts uzturēt ne tikai glosārijus, bet arī atsevišķi terminus, lai lietotājiem būtu iespēja meklēt informāciju par interesējošiem terminiem. Darba veiksmīgai izstrādei tika uzstādīti šādi uzdevumi:

- dažādu glosārija pieraksta formātu izpēte;
- dažādu tīmeklī pieejamo risinājumu apskate, kas palīdzētu sistēmas izveidē;
- datu bāzes struktūras izveide balstoties uz iegūtajiem rezultātiem;
- pētījums un risinājums glosāriju importēšanas un eksportēšanas mehānismu izveidei;
- sistēmas strādājošas versijas izveide.

Pirmā nodaļa satur ievadu par to, kas ir termins un glosārijs. Kā arī pētījumu par glosārija pieraksta formātiem, to lietojumu un secinājumus par to kāda datu struktūra jāatbalsta sistēmā glabājamajiem glosārijiem.

Otrajā nodaļā ir aprakstīts un izveidots datu bāzes modelis, kas spēs uzturēt visu nepieciešamo informāciju. Galvenais uzsvars tika likts uz to, lai datu bāze spētu glabāt terminu datus dažādās valodās un terminiem būtu iespējams pievienot vairākas definīcijas, kas savukārt arī varētu būt vairākās valodās.

Trešajā nodaļā ir aprakstīta darba praktiskā daļa, par to kādas tehnoloģijas un pieejas lietotas, lai izveidotu strādājošu glosāriju informāciju sistēmu. Šī darba daļa tika veidota balstoties uz iegūto informāciju no pirmās un otrās nodaļas.

# 1. GLOSĀRIJU PIERAKSTA DATU STRUKTŪRAS IZPĒTE

Šajā nodaļā apkopota informācija, kas palīdzēs saprast glosārija pieraksta valodas struktūru, kāda būtu jāatbalsta veidojamajā sistēmā. Kā arī, lai spētu veiksmīgi uzprojektēt sistēmas datu bāzi un izveidotu korektus algoritmus, kas ļautu importēt sistēmā glosārijus no ārējiem resursiem, nepieciešams apskatīt un izpētīt dažādus glosāriju XML pieraksta formātus. Saprast kāda ir to struktūra un kādu informāciju vajadzētu glabāt par glosāriju, terminiem un definīcijām.

Papildus ir apskatīti tūmeklī pieejamie resursi, lai saprastu glosāriju sistēmas perspektīvas un ko tai būtu jānodrošina. Kā arī tas dos vizuālu ieskatu, tam kā varētu izskatīties veidojamā sistēma.

## 1.1. Glosārijs

Glosārijs ir alfabētisks tekstā lietotu neparastu, maz pazīstamu, grūti saprotamu vārdu, frāžu un to skaidrojumu vai tulkojumu saraksts, kas parasti mazas vārdnīcas veidā pievienots kādai grāmatai vai rokrakstam [1].

Glosāriju veidi var būt daudz un dažādi, bet ir izdalāmi divi glosāriju pamatveidi. Viens no tiem ir vienvalodas glosārijs, kas sastāv no terminiem un to definīcijām par konkrētu zināšanu sfēru. Šādus glosārijus veido, piemēram, pasniedzēji universitātēs saviem pasniedzamajiem kursiem, lai kursa apmeklētāji spētu saprast kursa vadītāja valodā izmantojamus terminus un spētu veiksmīgi apgūt kursu. Otrs glosāriju veids, ko varētu izdalīt ir tulkošanas glosārijs, kas satur tikai terminus. Vienā kolonā ir termini, piemēram, latviešu valodā, bet otrā kolonā šie termini ir pārtulkoti, piemēram, angļu valodā. Šāda veida glosārijs tiek izmantots terminu tulkošanai un var tikt sagatavots importēšanai kādā no tulkošanas lietojumprogrammām.

## 1.2. Termins

Galvenās glosārija sastāvdaļas ir termins un tā skaidrojums jeb definīcija. Termins ir vārds vai vārdkopa, kas izteic (nosauc un iezīmē) attiecīgās jomas (nozares) noteiktu jēdzienu kā nozares jēdzienu sistēmas sastāvdaļu, piemēram, valodniecības jēdzieni izteikti ar

terminiem vārds, īpašvārds, sugas vārds, vārdkopa. Termina sastāvā var būt simboli, piemēram,  $\beta$  stari. Katram terminam ir sava definīcija un tai jābūt tieši nodefinētai, kas paskaidro termina lietojumu konkrētajā zināšanu sfērā.

Termina struktūra var atšķirties, bet no pieejamajiem rakstiem izdevās noteikt kāda būtu labākā un pilnīgākā termina struktūra veidojot glosāriju. Iegūtā termina struktūra tika balstīta uz izpētīto rakstu par ontoloģiju [2].

- Unikāls identifikators termina vārds, piemēram, “melnās kastes testēšanas paņēmiens”.
- Domēns norāda kurai zināšanu sfērai(nozarei) pieder konkrētais termins, piemēram, bioloģijai vai datorzinātnēm.
- Viena vai vairākas atsauces uz citiem terminiem, kas varētu būt saistīti vai līdzīgi.
- Cits termina identifikators, iespējams terminam ir vairāki identifikatori, ko mēdz izmantot.
- Sinonīmi - cieši saistīti vārdi vai frāzes, kas saistīti ar termina vārda nozīmi.
- Atsauces no kurienes ir ņemts termina skaidrojums, vai informācija par tā nozīmi.
- Komentārs - jebkāda veida papildus informācija, piemēram, termina lietošanas piemēri.
- Novecojis - termina status. Norāda to vai termins vēl tiek aktīvi lietots.

Šādas termina datu struktūra ir ļoti noderīga, tā sniedz lielāku ieskatu par pašu terminu un tā lietojumu. Kā arī pievienotās atsauces un sinonīmi varētu būt noderīgi gadījumos, ja tiek pētīta termina rašanās vēsture. Attēlā 1.1. redzami terminu pieraksta piemēri balstīti uz augstāk aprakstīto struktūru.

Identifikators: operētājsistēma  
Domēns: datorzinātne  
Citi termin: operētājsistēma DOS, operētājsistēma UNIX  
Sinonīmi:  
Atsauces: AkadTerm: <http://termini.lza.lv/>  
Komentārs: Tiek lietots, lai raksturotu datorprogrammu kompleksu, kas nodrošina datora darbību.  
Novēcojis: nav

Identifikators: lietvārds  
Domēns: valoda  
Citi termin: norises lietvārds, kopības lietvārds  
Sinonīmi: substantīvs  
Atsauces: AkadTerm: <http://termini.lza.lv/>  
Komentārs: Teikumā - Saule lec, lietvārds ir saule.  
Novēcojis: nav

### 1.1. att. Termina struktūra

Attēlā redzamais terminu pieraksts ir labi strukturēts un ir pieejama pilnīga informācija par aprakstīto terminu. Aplūkojot šādi pierakstītus terminus rodas lielāka skaidrība par termina nozīmi un lietojumu pat neredzot definīciju.

## 1.3. Strukturēta terminoloģija

Termina bāzē jēdziena ieraksti parasti ir iedalīti šādās sadaļās: sadaļa, kas apraksta koncepciju, sadaļa priekš katras valodas un sadaļa priekš katra termina. Visas šīs sadaļas ir vērstas uz terminu [3].

Termina sadaļa sastāv no viena termina un informācijas par to:

- termina tips, forma, akronīms, abreviatūra un cits;
- termina piederība, piemēram, lietvārds, darbības vārds, īpašības vārds un citi;
- kontekstuāls termina piemērs teikumā, lai parādītu tā lietojumu;
- veidotājs, ja kāds cilvēks veidojis terminu specifiski sev, kādā darbības sfērā;
- atbildīgie, ja pie termina izveides strādājuši vairāki cilvēki;
- atsauce uz citu terminu, ja ir sastopams līdzīgs pēc nozīmes;
- lietošanas piemērs.

Valodu sadaļas sastāv no vienas vai vairākām terminu sadaļām un informācijas par konkrēto valodu:

- valodas lietošanas noteikumi konkrētajā valodas sadaļā;

- koncepcijas definīcija.

Koncepcijas sadaļa:

- domēns jeb zināšanu sfēra, kurai pieder koncepcija;
- atsauce uz koncepcijas ilustrējošu attēlu.

Apskatot sīkāk katru sadaļu redzams, ka visa minētā informācija sadaļās ir attiecināma uz terminu, kas arī pierāda to orientāciju uz pašu terminu.

## **1.4. TBX, MARTIF un OLIF standarti**

Ir vairāki standarti kā apmainīties ar terminoloģijas informāciju, piemēram, TBX “TermBase eXchange”, MARTIF “Machine-Readable Terminology Interchange Format - ISO 12200” un OLIF “Open Lexicon Interchange Format”. Visi šie ir terminu pieraksta formāti, kas veidoti glosārija datu apmaiņai. Ar atbilstošajiem rīkiem ir iespējams apstrādāt šāda veida glosārijus, jo ir zināma to struktūra, tādējādi šie formāti nodrošina to, ka glosāriji var tikt importēti, piemēram, tulkošanas lietojumprogrammās.

### **1.4.1. TBX pieraksta formāts**

TBX ir internacionāls standarts, lai atspoguļotu un apmainītos ar terminoloģiskiem datiem. Šis standarts definē formātu, struktūru un datu kategorijas, kā atspoguļot informāciju par terminiem, to zināšanu apgabalu jeb domēnu [4].

Galvenais formāta nolūks ir veidot datu struktūru tādu, lai atbilstošā lietojumprogramma varētu piekļūt termina datiem, atspoguļot tos, manīt vai kā citādi apstrādāt. Tas nodrošina ērtu terminu datu izmantošanu dažādos tulkošanas darbos.

TBX tika izstrādāts ar Localization Industry Standards Association (LISA) un International Organization for Standardization (ISO) 2008. gadā kā ISO 30042:2008 standarts. Kā arī šim standartam ir vairāki dialekti tādi, kā “TBX-Default”, kas paredzēts pēc iespējas pilnākas informācijas glabāšanai par terminu. “TBX Basic” - tika izstrādāts, lai efektīvi glabātu lielu daudzumu glosāriju XML formātā. Šo formātu var izmantot vienvalodas, vairākvalodu un divvalodu glosāriju glabāšanai. “TBX-Min” - ir salīdzinoši jaunizstrādāts dialekts, kas paredzēts vienvalodas vai divvalodas glosāriju glabāšanai. Pamatā šo dialektu ērti izmantot tulkiem, kas nodarbojas ar terminu tulkošanu [5].

TBX pieraksta formātā katram termina ierakstam jeb “termEntry” jāsaturs atribūts “langSet” jeb valoda kādā rakstīts termins. Terminam iespējams norādīt papildus informāciju, tādu kā avots, vārdsķira. Šī informācija ievietoama elementā “termNote”. Un termina definīcija ierakstāma “definition” elementā [14]. Attēlā 1.2. redzams piemērs kā noformēt failu TBX pieraksta formātā.

```

<?xml version='1.0'?>
<martif type='TBX' xml:lang='lv'>
  <martifHeader>
    <fileDesc>
      <sourceDesc>
        <p>Izveidojis Artūrs Bugins</p>
      </sourceDesc>
    </fileDesc>
    <encodingDesc>
      <p type='DCSName'>System "TBXDCSv05c.xml"</p>
    </encodingDesc>
  </martifHeader>
  <text>
    <body>
      <termEntry id='c1'>
        <descrip type='subjectField'>Testēšana</descrip>
        <descrip type='conceptPosition' ></descrip>
        <descrip type='definition'>
          Programmatūras un aparatūras darbības pārbaude izmantojot testa datus.
        </descrip>
        <langSet xml:lang='lv'>
          <ntig id='t1'>
            <termGrp>
              <term>Testēšana</term>
            </termGrp>
          </ntig>
        </langSet>
        <termNote></termNote>
      </termEntry>
    </body>
  </text>
</martif>

```

1.2. att. TBX piemērs

Šādi izskatās korekti noformēts fails, kas veidots pēc TBX standarta. Lietotā pieraksta valoda ir XML. TBX pieraksts sastāv no galvenes, kas apraksta failu, pēc kā var sekot vairāki terminu ieraksti, kas apzīmēti ar <termEntry>, katrā termina ierakstā ir apraksta elementi <descrip>, kas satur atribūtus norādošus vai apraksts ir termina definīcija vai koncepcijas pozīcija. Elements <langSet> satur terminu konkrētajā valodā, kas norādīta elementa valodas atribūtā.

Piemērā pierakstīts termins testēšana latviešu valodā ar definīciju - programmatūras un aparatūras darbības pārbaude, izmantojot testdatus.

## 1.4.2. MARTIF pieraksta formāts

Šis standarts ir atvase no TEI “Text Encoding Initiative” un LISA”Localisation Industry Standards Association”. Standartā lietotas ap 150 datu kategorijas, kas standartizētas kā ISO 12620. MARTIX izstrādāts priekš sistēmām, kas balstītas uz koncepciju, tāpēc šis standarts ir vairāk koncepcijas orientēts [6].

MARTIF spēj nodrošināt ērtu un elastīgu terminu datu apmaiņu starp dažādām terminoloģiskām datu bāzu menedžmenta sistēmām. Šo pieraksta formātu var izmantot ne tikai, lai apmainītos ar datiem, bet arī gadījumos, ja nepieciešams izmainīt datu bāzes struktūru un pāriet uz citu datu formātu, jo MARTIF nodrošina datu konvertāciju un pārveidi [17].

Attēlā 1.3. redzams MARTIF lietošanas piemērs.

```
<martif>
  <martifHeader>
    <fileDesc>
      <titleStmt>
        <title>Piemērs martin dokumenta izveidē</title>
      </titleStmt>
    </fileDesc>
  </martifHeader>
  <text>
    <body>
      <termEntry>
        <descrip type='subjectField'>Testēšana</descrip>
        <ntig lang=lv>
          <termGrp>
            <term>Testēšana</term>
            <termNote type='partOfSpeech'>darbības vārds</termNote>
            <termNote type='gen'>s</termNote>
          </termGrp>
          <descripGrp type='definition'>
            Programmatūras un aparatūras darbības pārbaude izmantojot testa datus.
          </descripGrp>
        </ntig>
      </termEntry>
    </body>
  </text>
</martif>
```

1.3. att. MARTIF lietošanas piemērs

MARTIF lietošanas piemērā redzams, ka galvene satur informāciju par failu. Katram terminam ir savs ierakst <termEntry>, kas satur informāciju par terminu, tādu kā termina valoda, definīcija, dzimums un vārdšķira.

Konkrētajā piemērā pierakstīts termins testēšana. Šī termina vārdšķira ir darbības vārds un dzimte ir sieviešu. Definīcija: programmatūras un aparatūras darbības pārbaude izmantojot testa datus.

### 1.4.3. OLIF pieraksta formāts

OLIF “Open Lexicon Interchange Format” standarts piedāvā plaša spektra atbalstu valodas datu atspoguļošanā un apmaiņā ar XML valodas palīdzību. OLIF nodrošina: terminoloģisko datu apmaiņu un pārvaldīšanu, terminu ieguvu, kontrolētu valodu, informācijas atspoguļošanu, glosāriju izstrādi un ontoloģijas [7].

OLIF vienvalodas glosārija datu piemērs attēlots attēlā 1.4. Ieraksta dati tiek grupēti pēc lingvistiskās, leksiskās un terminoloģiskās informācijas.

```
<entry>
  <mono>
    <keyDC>
      <canForm>table</canForm>
      <language>en</language>
      <ptOfSpeech>noun</ptOfSpeech>
      <subjField>general</subjField>
      <semReading>86</semReading>
    </keyDC>
    <monoDC>
      <monoAdmin>
        <originator>Weber</originator>
        <adminStatus>ver</adminStatus>
      </monoAdmin>
      <monoMorph>
        <inflection>like book,books</inflection>
      </monoMorph>
      <monoSyn>
        <synType>cnt</synType>
        <synFrame>[gencomp-opt]</synFrame>
      </monoSyn>
      <monoSem>
        <definition>
          An arrangement of words, numbers, or signs or combinations of them,
          as in parallel columns, to exhibit a set of facts or relations in a definite,
          compact, and comprehensive form.
        </definition>
        <semType>inform</semType>
      </monoSem>
    </monoDC>
  </mono>
</entry>
```

1.4. att. OLIF lietošanas piemērs

Elements “keyDC” satur datus par kanonisko formu, valodu, vārdšķiru, subjektu un semantiku. Administratīvā datu kategorija “monoAdmin” satur informāciju par autoru, administratīvo statusu un ģeogrāfisko lietojumu. Morfoloģiskā datu kategorija apraksta

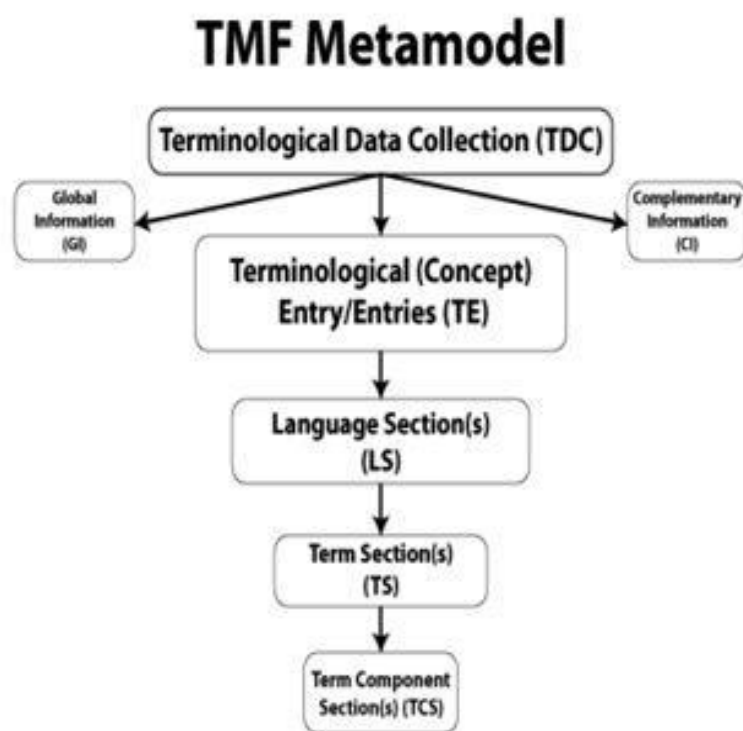
termina morfoloģisko struktūru. Semantiskā datu kategorija “monoSem” satur definīciju un citus semantiskos datus [15].

### **1.5. TBX, MARTIF un OLIF kopskats**

Apskatot šos glosārija pieraksta formātus darba autors novēroja, ka ievietotie dati un struktūra ir līdzīga. Visa pamatā ir glosārijs kā saknes elements, par kuru var tikt ievietota tāda informācija kā, valoda, komentārs un autors. Katrā glosārija ir termina ieraksti, kas glabā informāciju par to kādā valodā rakstīts termins, termina identifikators, termina definīcija, avoti un lingvistikā informācija par terminu.

### **1.6. TMF - Terminoloģijas marķēšanas ietvars**

TMF ir abstrakts datu modelis, kas definēts ISO 16642 standartā. TMF struktūra attēlota attēlā 1.2. TMF metamodelis. Šis datu modelis neparedz kādi ir obligātie datu atribūti kuriem jāparādās veidotajā terminu bāzē. Vienīgās lietas, kas ir obligātas ir termins un valoda kādā tas rakstīts. To varētu uzskatīt par TMF stipro pusi, jo tādā veidā šis ietvars var atbalstīt daudzveidīgas terminu bāzes [3]. TMF modelis redzams attēlā 1.5.



1.5. att. TMF metamodelis

Modelis sevī iekļauj sekojošus līmeņus:

- Valodas sadaļa: Šī sadaļa nosaka to, ka terminiem konkrētās valodas blokā jābūt sagrupētiem pēc to zināšanu sfēras.
- Termina sadaļa: Katras valodas sadaļas iekšienē katram terminam ir sava sadaļa. Tas nozīmē, ka katram terminam dati tiek definēti atsevišķi.
- Termina komponentes: Dažreiz termins, kas sastāv no vairākiem vārdiem, piemēram, melnās kastes testēšanas princips, ir sadalīts atsevišķi pa vārdiem un šie vārdi tiek glabāti atsevišķi terminu bāzē. Tas nepieciešams priekš tulkošanas rīkiem, jo viens un tas pats termins tekstā var parādīties dažādās formās, un lai rīks spētu sniegt informāciju par terminu nepieciešama lingvistiskā informācija. Un ja termins tiek sadalīts pa vārdiem, tad par katru vārdu iespējams glabāt nepieciešamo lingvistisko informāciju.
- Globālā informācija: Attiecas uz visu termina bāzi kopumā. Tā satur termina bāzes nosaukumu, izcelsmi un veidotāju.
- Papildu informācija: Var saturēt informāciju, par cilvēkiem, kas izstrādāja terminu bāzi. Kā arī informāciju par izmantotajām grāmatām un tīmekļa resursiem, kas tika izmantoti terminu un terminu bāzes veidošanai.

Apskatot TMF modeli saskatāma šāda datu struktūra, katram terminam ir jābūt domēnam, jeb zināšanu apgabalam, katrs termins tiek definēts atsevišķi un ir pieejama informācija par termina izcelsmi un lingvistiskajiem datiem.

## **1.7. GlossML pieraksta formāts**

GlossML ir XML bāzēta vārdnīca, kas veidota priekš glosāriju atspoguļošanas XML formātā. GlossML var uzturēt vienalodas un daudzvalodu glosārijus, to terminus un definīcijas [9]. Šāda veida formāti tikai veidoti viena iemesla pēc, lai būtu iespēja ērti apmainīties ar glosārijiem. Atbilstošās lietojumprogrammas spēj apstrādāt XML valodu un dod lietotājam pilnu kontroli pār glosāriju. Lietotājs var mainīt, dzēst vai pielikt jaunus terminus esošajam glosārijam.

### **1.7.1. GlossML lietojums**

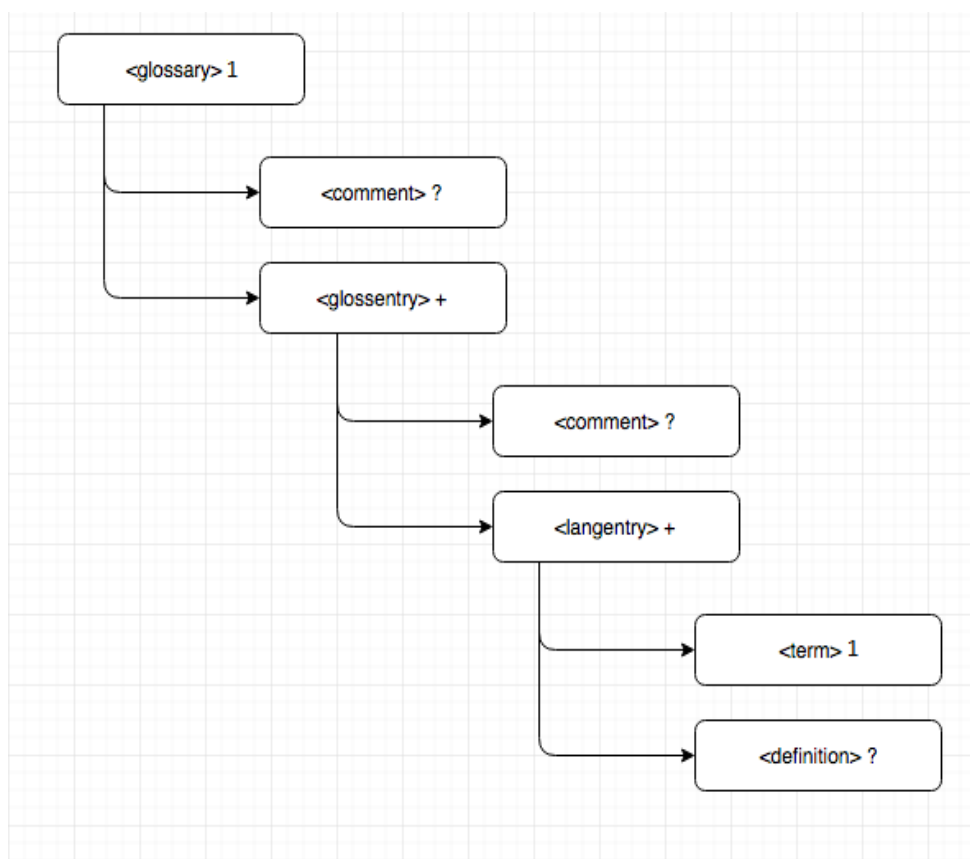
“Glossary Markup Language” - Glosāriju iezīmēšanas valoda. Šī iezīmēšanas valoda veidota ar XML jeb paplašināto iezīmēšanas valodu. Šī glosāriju pieraksta valoda paredzēta vienalodas vai daudzvalodu glosāriju glabāšanai. Uzsvars tiek likts uz tādiem atribūtiem kā valoda, termins un definīcija [8].

GlossML ir ļoti vienkāršs un nekomplicēts formāts, kas sastāv no 6 elementiem un 4 atribūtiem. Pieraksta formāts ir tik vienkāršs, jo tas paredzēts glosāriju pierakstam, kur pamatā ir tikai termins un tā skaidrojums. Visi GlossML glosāriji var tikt papildināti ar XML atribūtiem, bet tiem jābūt korekti nodefinētiem XML failā un šiem atribūtiem jābūt pieejamiem validācijai [9].

### **1.7.2. GlossML struktūra**

GlossML glosārija sakne ir <glossary> elements un galvenā lietojamā valoda glosārijā apzīmēta ar “srclang” atribūtu. <glossary> elements var saturēt <comment> un <glossentry> elementus. <glossentry> elements satur terminu un tā tulkojumus vienā vai vairākās valodās. Termins tiek glabāts <langentry> elementā, kur termina vārds tiek glabāts <term> elementā, bet tā definīcija jeb skaidrojums tiek glabāts <definition> elementā.

GlossML elementu hierarhiska struktūra attēlota attēlā 1.6.



1.6. att. GlossML struktūra

Attēlotajā shēmā redzama GlossML formāta hierarhiskas elementu savstarpējas attiecības. Pie elementiem atrodami simboli “1”, “?” un “+”. “1” nozīmē tieši viens, “?” nozīmē nulle vai vairāk un “+” nozīmē viens vai vairāk. Tas nozīmē, ka veidojot glosāriju GlossML formātā vienā failā drīkst definēt tikai vienu glosāriju, glosārijam var būt vairāki vai neviens komentārs par pašu glosāriju. Veidojamajā glosārijā jābūt vismaz vienam ierakstam, tas ir `<glossentry>` un katrā ierakstā glabājama informācija par termina valodu, paša termina vārds un tā skaidrojums. Termina valoda un pats termins ir obligāti norādāmie lauki bez kuriem glosārijs nebūs korekti veidots.

### 1.7.3. GlossML piemērs

Neliels piemērs, tam kā būtu jāizskatās GlossML formātā veidotam glosārijam redzams attēlā 1.7.

```
<?xml version="1.0" encoding="UTF-8"?>
<glossary version="1.0" srclang="lv-LV">
  xmlns="http://www.maxprograms.com/gml"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.maxprograms.com/gml GlossML.xsd">
  <comment>Piemērs daudzvalodu glosārijam</comment>
  <glossentry>
    <langentry xml:lang="en">
      <term>term</term>
      <definition source="AkadTerm">
        A word or phrase that stands for a concept used in a particular subject area.
      </definition>
    </langentry>
    <langentry xml:lang="lv">
      <term>termins</term>
    </langentry>
  </glossentry>
  <glossentry>
    <comment from="Arturs">
      Komentāra piemērs termina sadaļai
    </comment>
    <langentry xml:lang="en">
      <term>statistic</term>
      <definition source="Merriam Webster">
        a numerical fact or datum, esp. one computed from a sample
      </definition>
    </langentry>
    <langentry xml:lang="es">
      <term>estadística</term>
      <definition source="Larousse">cuadro numérico de un hecho que se
        presta a la estadística</definition>
    </langentry>
  </glossentry>
</glossary>
```

#### 1.7.att. GlossML piemērs

Piemērs parāda to, kā pierakstīt glosāriju GlossML formātā, ko pēc tam būtu iespējams nodot kādai apstrādes programmai, kas balstoties uz XML pieraksta valodu spētu manipulēt ar glosārija datiem.

Redzams, ka sākumā tiek definēts glosārijs, kuram pielikts komentārs un valoda kādā veidots glosārijs, kas konkrētajā gadījumā būtu latviešu valoda. Glosārijs sastāv no diviem ierakstiem, kur glabājas termins un tā informācija. Katram terminam pielikts valodas atribūts, definīcija un definīcijas avots. Konkrētajā gadījumā pierakstīts termins statistika latviešu valodā ar definīciju - Zinātnes nozare, kas zinātniskos un praktiskos nolūkos pētī cilvēku

sabiedrības, tautsaimniecības attīstības kvantitatīvās pārmaiņas. Definīcija izgūta no AkadTerm vietnes.

## 1.8. GlossXML formāta glosāriji

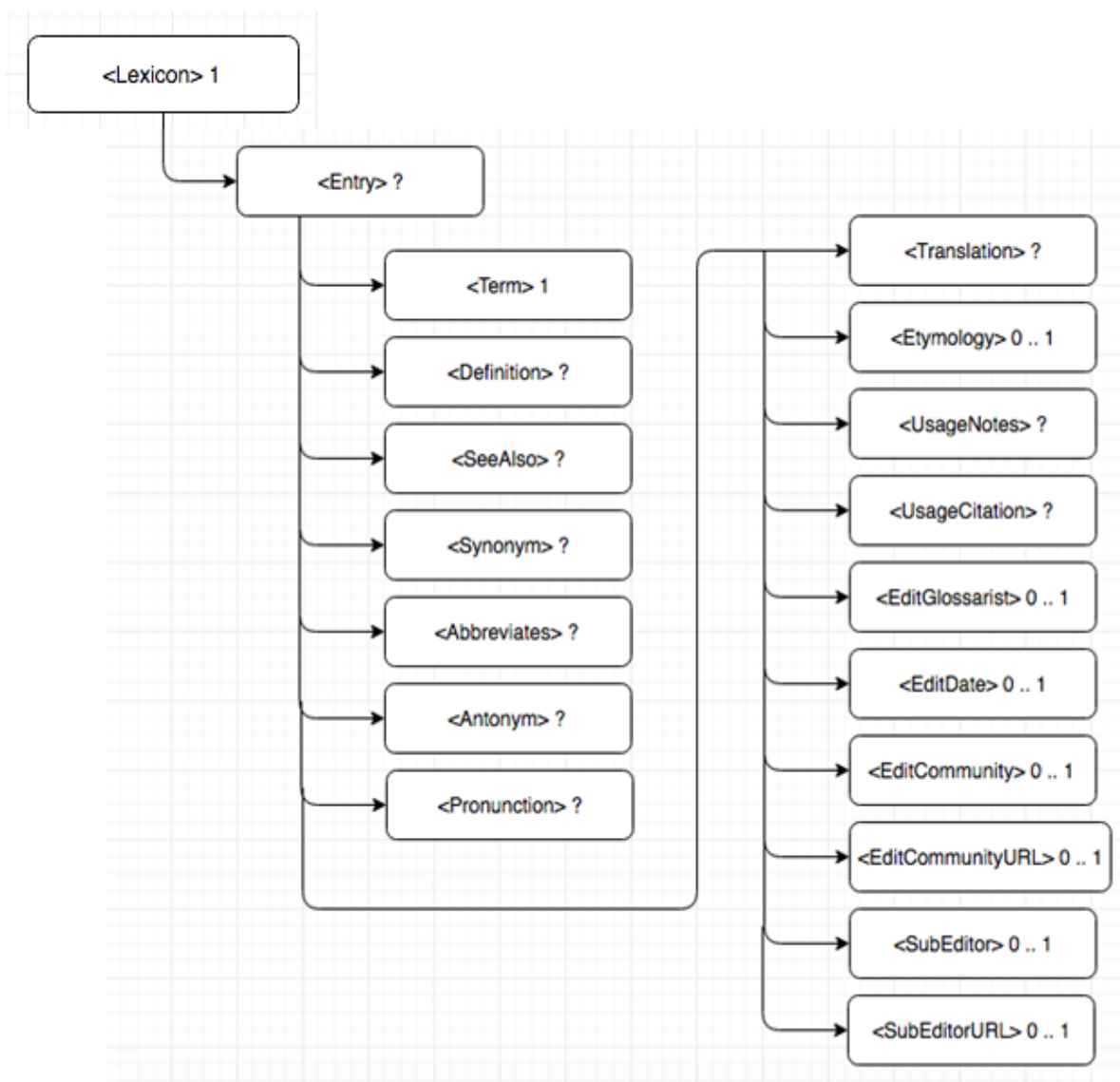
Glosāriju un terminu pārvaldei nepieciešams ērts formāts, tā lai tas būtu saprotams un ērti maināms. Un XML pierakstīšanas valoda sniedz šādas iespējas. Un ir izveidots GlossXML pieraksta formāts, kas pēc būtības ir līdzīgs GlossML pieraksta formātam, tomēr atšķirība ir glosārija datu struktūrā un detalizācijā. Ja GlossML saturēja tikai pamatinformāciju, tas ir valodu, terminu un tā definīciju, tad GlossXML ļauj glabāt arī tādus datus kā, piemēram, citi termini, kas saistīti ar konkrēto terminu, sinonīmus, termina izrunu, abreviatūru, lietošanas piemērus un cita veida informāciju. Šāda veida glosāriji varētu būt populāri valodnieku starpā, jo glosārijā pieejama visa valodniekiem nepieciešamā informācija.

### 1.8.1. GlossXML elementu lietojums

<Lexicon> - Dokumenta sākuma elements, kas var saturēt vairākus <Entry> elementus. <Entry> - Vārds vai frāze glosārijā. Elements saturēs visus elementus, kas attieksies uz terminu. <Term> - Vārds vai frāze, kas var būt tikai viena katram ierakstam. <Definition> - Termina definīcija teksta formātā. Var būt nulle vai vairākas. <SeeAlso> - Citi termini, kas varētu būt saistīti ar konkrēto terminu. Var būt vairāki, vai nebūt vispār. <Synonym> - Neviens vai vairāki sinonīmi terminam. <Abbreviates> termina abreviatūra vai abreviatūras, ja tādas eksistē un galvenais termins ir saīsinājums vai akronīms uz kādu citu terminu. <Antonym> - Termina antonīms, var nebūt, vai būt vairāki. <Pronunciation> - Termina izruna. <Translation language=*lang*> - Termina tulkojums citā valodā. Termins iztulkots valodā, kas norādīta "language" atribūtā. Valoda jānorāda salīdzinājumā, kas atbilst ISO 639-1/2 standartam, piemēram, lav-lv, kas būtu termina tulkojums latviešu valodā. <Etymology> - Elements, kas satur piezīmes par termina etimoloģiju. <UsageNotes> - Elements satur termina lietojuma piezīmes. <UsageCitation> - Elements satur termina lietojuma piemērus. <EditGlossarist> - Satur ierakstu par to kurš pēdējais veicis izmaiņas termina ierakstā. Var būt tikai viens elements ar šādu ierakstu. <EditDate> - Datums, kad pēdējo reizi mainīts ieraksts. <EditCommunity> - Kopiena, kas pēdējā strādājusi pie termina izmaiņām vai veidošanas. <EditCommunityUrl> - Kopienas saite, kur termins pēdējo reizi

mainīts vai publicēts. <SubEditor> - Ja vēl kāds strādājis pie termina. <SubEditorUrl> - Saite uz līdzizstrādātāju [16].

Attēlā 1.8. redzama kopaina elementu lietojumam GlossXML pieraksta formātā.



1.8. att. GlossXML struktūra

Shēmā attēlota GlossXML hierarhiska elementu struktūra, kāda jāveido pierakstot glosāriju šajā formātā. Simboli “1” nozīmē tieši viens, “0 .. 1” nozīmē nulle vai viens, “?” nozīmē nulle vai vairāk.

Veidojot glosāriju GlossXML formātā failā drīkst parādīties tikai viens <Lexicon> elements, kas apzīmē glosāriju. Glosārijā var būt vairāki ieraksti, kur katrā ierakstā ir informācija par konkrēto terminu. Vienā ierakstā nedrīkst parādīties vairāki termini tomēr katrā ierakstā var nodefinēt vairākus <Translation> elementus, kas satur termina tulkojumu citās valodās.

## 1.8.2. GlossXML lietošanas piemērs

Neliels piemērs, tam kā būtu jāizskatās GlossXML formātā veidotam glosārijam redzams attēlā 1.9.

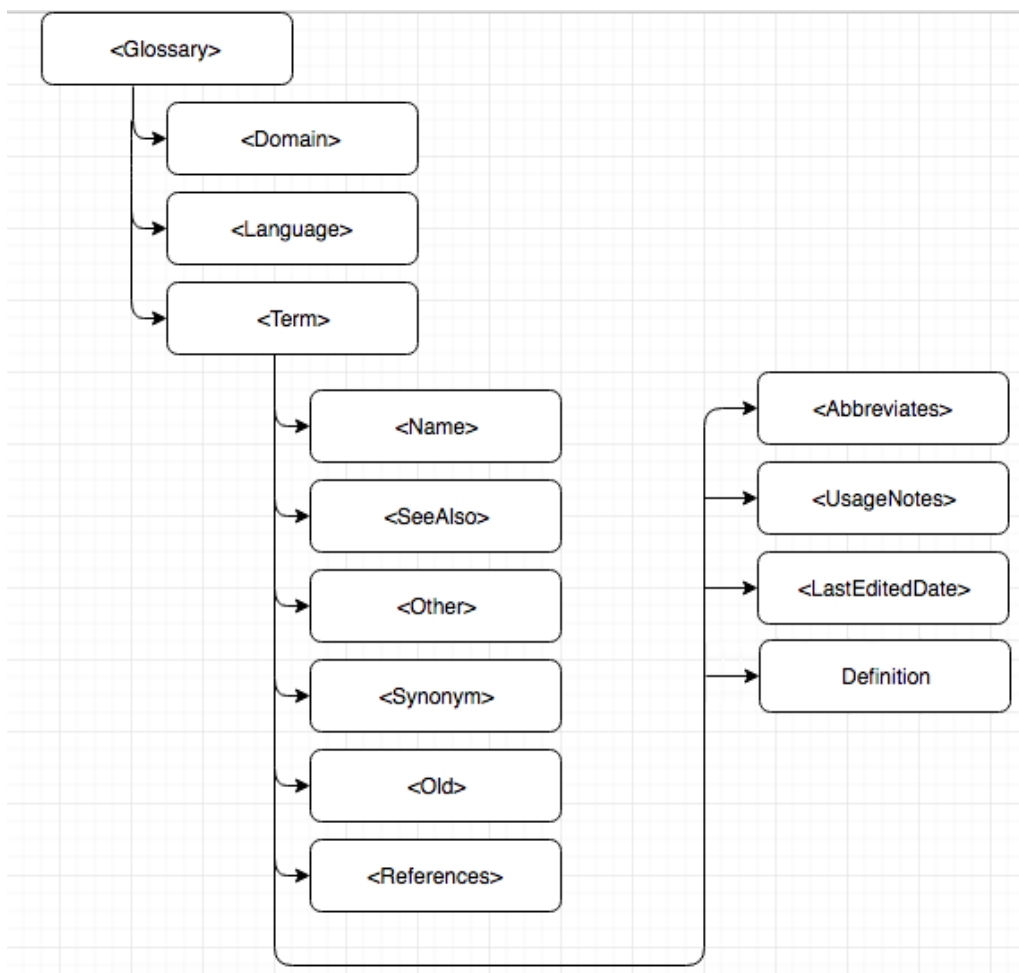
```
<Lexicon>
  <Entry>
    <Terms>Termins</Term>
    <Definition>
      Konkrētas tematiskās jomas, speciālās nozares jēdzina vārdiskis apzīmējums.
    </Definition>
    <SeeAlso>Šeit ievieto līdzīgus terminus</SeeAlso>
    <SeeAlso>Šeit ievieto līdzīgus terminus</SeeAlso>
    <Synonym>Šeit ievietojams sinonīms</Synonym>
    <Translation language=en>Term</Translation>
  </Entry>
  <Entry>
    ...
  </Entry>
</Lexicon>
```

1.9. att. GlossXML lietojuma piemērs

Piemērs parāda to kādam jābūt korekti nodefinētam glosārijam GlossXML pieraksta formātā, lai šo glosāriju varētu izmantot attiecīgajās apstrādes lietojumprogrammās.

## 1.9. Secinājumi par glosāriju pieraksta valodu

Ir pieejamas dažādas terminu un glosāriju pieraksta formas, bet visās formās novērojama kopēja struktūra. Balstoties uz izpētītajiem formātiem izdevās izcelt elementus, kas nepieciešami datu struktūrā veidojot glosāriju, tie attēloti attēlā 1.10. Kā arī sistēmas mērķis nav uzturēt glosārijus, kas būtu paredzēti valodniekiem, tāpēc struktūrā neparādās tādas lietas kā termina izruna, piederība vai vārdšķira. Bet datu bāzes struktūra tiks plānota tā, lai nākotnē ja tas būs nepieciešams nerastos problēmas papildināt glosāriju struktūru.



1.10. att. Svarīgākie elementi glosārija struktūrā

Attēlotajā shēmā redzams, ka visa pamatā ir glosārijs, kas veidots kādā konkrētā zināšanu sfērā jeb domēnā. Tāpēc glosārijam pieejami valodas un domēna elementi. Glosārija pamatelements ir termins, kas satur informāciju par tā definīciju, termina vārdu, saistītajiem terminiem, sinonīmiem, atsaucēm, abreviatūru, lietošanas piemēriem un pēdējās atjaunošanas datumu.

## 1.10. Esošo tīmeklī pieejamo glosāriju un vārdnīcu izpēte

Sadaļā tiks aprakstīti dažādi tīmeklī pieejami glosāriji un vārdnīcas, tas tiek darīts, lai labāk izprastu glosāriju struktūru un to kādai vajadzētu būt veidojamajai sistēmai, lai tā spētu konkurēt ar jau esošajiem tīmeklī pieejamajiem risinājumiem apskatot esošo risinājumu trūkumus un stiprās puses.

## 1.10.1 AkadTerm un Tēzaurs

AkadTerm [1] ir Latvijas Zinātņu akadēmijas Terminoloģijas komisijas izstrādāto un apstiprināto terminu sakopojums, kuram pievienoti arī citi terminu krājumi. Tajā ievietoti 1 055 533 termini, kas atrodami 6 valodās. Vietne atbalsta Latviešu, Krievu, Angļu, Vācu, Franču un Latīņu valodas. Piemērs vietnē meklētam terminam redzams attēlā 1.11.

### Akadēmiskā terminu datubāze *AkadTerm*

glosārijs LV latviešu Meklēt [Palīdzība](#)  
Izmantojiet zvaigznīti \* vārda daļu meklēšanai (piemēram, dator\* vai \*pratība)

Jūs meklējat **glosārijs**

Atrasts 1 termins

• **glosārijs**

**EN** glossary  
**LV** glosārijs  
**RU** глоссарий  
**DE** Glossar[ium]

1. **EN** glossary  
**LV** glosārijs  
**RU** глоссарий

*Definīcija:* Bieži lietojamu frāžu un teksta fragmentu kopums, kas tiek veidots ar tekstapstrādes programmu starpniecību un ko vajadzības gadījumā var gatavā veidā iespraust sagatavojamajā dokumentā.  
LZA TK ITTEA terminu datubāze

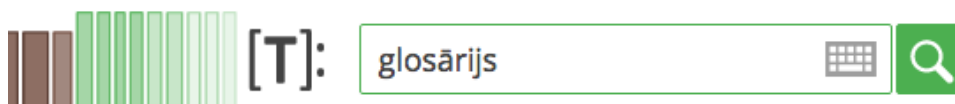
2. **EN** glossary  
**LV** glosārijs  
**RU** глоссарий  
**DE** Glossar[ium]

*Definīcija:* Alfabētisks tekstā lietotu neparastu, maz pazīstamu, grūti saprotamu vārdu, frāžu (arī terminu) un to skaidrojumu vai tulkojumu (glosu) saraksts, kas parasti mazas vārdnīcas veidā pievienots kādai grāmatai vai rokrakstam.  
Valodniecības pamatterminu skaidrojošā vārdnīca. — R., 2007

#### 1.11. att. Termina meklēšanas rezultāts AkadTerm vietnē

Attēlā redzams meklēšanas rezultāts terminam – glosārijs. Lietotājam tiek parādīti termina skaidrojumi, kas seko viens pēc otra, kur korektākais skaidrojums atrodas saraksta augšgalā. Katram skaidrojumam pieejama atsauce, kur atrodams skaidrojums.

Tēzaurs [12] ir plaša vārdnīca, kas pieejama tīmekļa resursos. Šajā vārdnīcā atrodami terminu skaidrojumi tikai latviešu valodā. Kā arī terminu datu bāze nav plaša un daudzi termini nav atrodami. Piemērs vietnē meklētam terminam redzams attēlā 1.12.



## glosārijs

1. Reti lietotu, grūti saprotamu vārdu skaidrojumu vai tulkojumu (glosu) saraksts, kas parasti mazas vārdnīcas veidā pievienots kādai grāmatai vai rokrakstam.
2. inf. Bieži lietojamu frāžu un teksta fragmentu kopums, kas tiek veidots ar tekstapstrādes programmu starpniecību un ko vajadzības gadījumā var gatavā veidā iespraust sagatavojamajā dokumentā.

AVOTI: SV99

MORFOLOĢIJA: lietvārds, vīriešu dzimte, 1. deklinācija +

### 1.12. att. Termina meklēšanas rezultāts Tēzaurs vietnē

Attēlā redzams vietnes atgrieztais rezultāts meklētajam terminam - glosārijs. Tiek parādīti vairāki skaidrojumi saraksta veidā. Zemāk redzami avoti, kas ir atsaucis redzamajiem skaidrojumiem. Un kā pati pēdējā informācija tiek parādīta termina vai vārda lingvistiskā informācija.

Abos resursos pieejams meklētājs, kur lietotājs var ievadīt savu meklēto terminu. Meklētājs atgriež sistēmā pieejamos terminu skaidrojumus. Katram terminam iespējami vairāki skaidrojumi, kas sagrupēti pēc to precīzuma un atbilstības terminam. Katrai definīcijai pievienota atsauce, no kurienes iegūts skaidrojums.

Struktūra, kas novērota šajos resursos: termins, definīcijas, atsaucis uz definīcijas avotu.

### 1.10.2. Wikipedia

Brīvā enciklopēdija, kur var atrast visāda veida informāciju, tai skaitā iespējams atrast arī glosārijus. Piemēram, glosārijs par sistēmas teoriju, kas pieejams [22]. Attēlā 1.13. redzama vietnes glosārija struktūra.

**Contents :**

Top · 0–9 · A · B · C · D · E · F · G · H · I · J · K · L · M · N · O · P · Q · R · S · T · U · V · W · X · Y · Z

**A** [ edit ]

- **Adaptive capacity:** An important part of the **resilience** of systems in the face of a **perturbation**, helping to minimise loss of **function** in individual human, and collective social and biological systems.
- **Allopoiesis** is the process whereby a system produces something other than the system itself.
- **Allostasis** is the process of achieving stability, or **homeostasis**, through physiological or behavioral change.
- **Autopoiesis** is the process by which a system regenerates itself through the self-reproduction of its own elements and of the network of interactions that characterize them. An **autopoietic** system renews, repairs, and replicates or reproduces itself in a flow of matter and energy. Note: from a strictly Maturanian point of view, autopoiesis is an essential property of biological/living systems.

*1.13. att. Glosārijs Wikipedia vietnē*

Glosārija struktūra ir samērā vienkārša, nav pieejams meklētājs, bet gan navigācijas josla ar burtiem, kas sakārtoti alfabētiskā secībā. Tādējādi pārvietojoties pa navigācijas joslu lietotājam ir iespēja meklēt terminus. Termini un to skaidrojumi atrodami tikai vienā valodā, kā arī katram terminam ir tikai viena definīcija.

Wikipēdija ļauj pievienot terminus un definīcijas, bet netiek dota nekāda informācija par to izcelsmi jeb avotu. Struktūra, kas novērota ir primitīva: termins un tā skaidrojums vienā valodā.

**1.10.3. WhatIs, bis.org un Akademic**

WhatIs [10] vietne ir līdzīga vārdnīcai, kur iespējams meklēt terminus un to skaidrojumus. Terminiem ir pievienotas atsauces uz rakstiem, kas saistīti ar konkrēto terminu.

Academic [13] vietne ir plaša vārdnīcas krātuve, kas satur vārdu skaidrojumus krievu valodā. Vietnē pieejams meklētājs, kas atgriež sistēmā glabātos terminu skaidrojumus. Katram skaidrojumam ir atsauce uz avotu, no kura iegūts šis skaidrojums.

bis.org ir maksājumu un mārketinga glosārijs [11], kuru apkopojusi mārketinga komiteja. Meklējot terminus meklētājs ne tikai parāda termina skaidrojumu, bet dokumentu no kura iegūts skaidrojums(source document). Dokumenta nosaukums ir kā saite, kas ved uz resursu tīmeklī. Attēlā 1.14. redzams termina meklēšanas rezultāts.

---

## book entry

The transfer of securities and other financial assets which does not involve the physical movement of paper documents or certificates (for example, the electronic transfer of securities).

Updated: 16 Apr 2012

Related terms: [book-entry system](#), [dematerialisation](#), [immobilisation](#), [securities settlement system \(SSS\)](#)

### Source documents

Date	Title
16 Apr 2012	<a href="#">Principles for financial market infrastructures</a>

---

#### 1.14. att. Termina meklēšanas rezultāts Bis.org vietnē

Attēlā redzams meklēšanas rezultāts mārketinga glosārijā. Meklētāja atgrieztā informācija ir ļoti labi noformēta, piemēram, saistītie termini ir kā saites, kas katra ved uz attiecīgo terminu. Un tādējādi lietotājs var ērti pāriet uz saistīto terminu un aplūkot tā informāciju.

Novērotā struktūra apskatītajās vietnēs: termins, saistītie termini, atsauce uz terminu, definīcija un atsauce uz definīciju jeb skaidrojumu.

#### 1.10.4. Trūkumi un secinājumi

Šobrīd nav pieejams vienots glosāriju sistēmas risinājums, kas nodrošinātu glosāriju importēšanu, eksportēšanu un pārvaldi. Apskatītajās vietnēs novēroti šādi trūkumi: lietotājs nevar pievienot jaunus skaidrojumus esošiem terminiem, nevar veidot savus glosārijus, nav iespējas atlasīt konkrēti interesējošus terminus un eksportēt tos, piemēram, word failā. Nav iespējams savilkt paralēles, piemēram, apskatīt līdzīgu terminu skaidrojumus vienlaicīgi. Glosāriji pārsvarā ir vienā valodā, izņemot AkadTerm vietni.

Tāpēc, lai glosāriju sistēma būtu efektīva jānodrošina glosāriju veidošana un pārvalde. Jāspēj ļaut lietotājam importēt sistēmā jau gatavus glosārijus, bet šiem glosārijiem būtu jāseko kādam no glosāriju pieraksta formātiem, tas varētu būt GlossML vai GlossXML, vai pašizdomāts pieraksta formāts, ko būtu iespējams adaptēt esošajiem glosārijiem. Sistēmas datu struktūrai jābūt tādai, lai būtu iespējas glabāt terminu un to skaidrojumus daudzās valodās. Datu struktūra nedrīkst būt primitīva, jo tad nevarēs nodrošināt sistēmas elastīgumu un tā ne ar ko neatšķirsies no parastiem internetā pieejamajiem glosārijiem.

## 2. DATU BĀZES MODELIS GLOSĀRIJU IS

Šajā nodaļā aprakstīta datu bāzes struktūras izveide, kas parādīs glabājamo informāciju par glosārijiem, terminiem un definīcijām. Atbilstošās datu bāzes struktūras izveide ir ļoti svarīga, jo bez korekti izplānotas datu bāzes nebūs iespējams izveidot labu glosāriju informāciju sistēmu, kas uzturēs glosārijus un nodrošinās to pārvaldi.

Sistēmai jāspēj ļaut lietotājam pievienot terminus, veidot glosārijus, importēt glosārijus un eksportēt glosārijus, piemēram, pdf vai doc formātā. Balstoties uz izpētītajiem materiāliem par glosāriju pieraksta valodu pirmajā nodaļā un nepieciešamajām sistēmas funkcionālajām prasībām tika veidota datu bāzes struktūra.

### 2.1. Datu bāzes modelis

Uzprojektētā datu bāze sastāv no sešām tabulām, kas uzturēs visu sistēmas nepieciešamo informāciju. Sistēmas datu glabāšanai tiks lietota MySQL datu bāze, jo darba autors ar to iepriekš strādājis un ir labi pazīstams

Tabulā “users” jeb lietotāji paredzēts glabāt lietotājus, kas reģistrēti sistēmā. Šī tabula nepieciešama, lai sistēmā būtu iespējams piesaistīt glosārijus tieši kādam lietotājam un šis lietotājs spētu pārvaldīt veidotos glosārijus, tas nodrošinās glosāriju privātumu gadījumos, ja lietotājs nevēlēsies, lai citiem sistēmas lietotājiem un viesiem būtu piekļuve šiem glosārijiem.

Tabulā “glossary” jeb glosārijs paredzēts glabāt glosārijus. Šī tabula nepieciešama, lai sistēmā būtu iespējams pārvaldīt lietotāja veidotus glosārijus. Kā arī tabula saturēs glosārija pamatinformāciju tādu kā nosaukums, apraksts, zināšanu apgabals, atsauces un valoda kādā tas veidots. Šāda veida informācija nepieciešama, lai nākotnē sistēma spētu ne tikai eksportēt glosārijus pdf vai doc formātā, bet arī, piemēram, kādā no glossXML pieraksta formātiem.

Tabulā “glossContent” jeb glosārija saturs ir starptabula, kas glabā glosārija saturu, tas ir glosārijam pievienotos terminus. Šādā veidā tiks nodrošināta ērta datu atlase konkrētajam glosārijam.

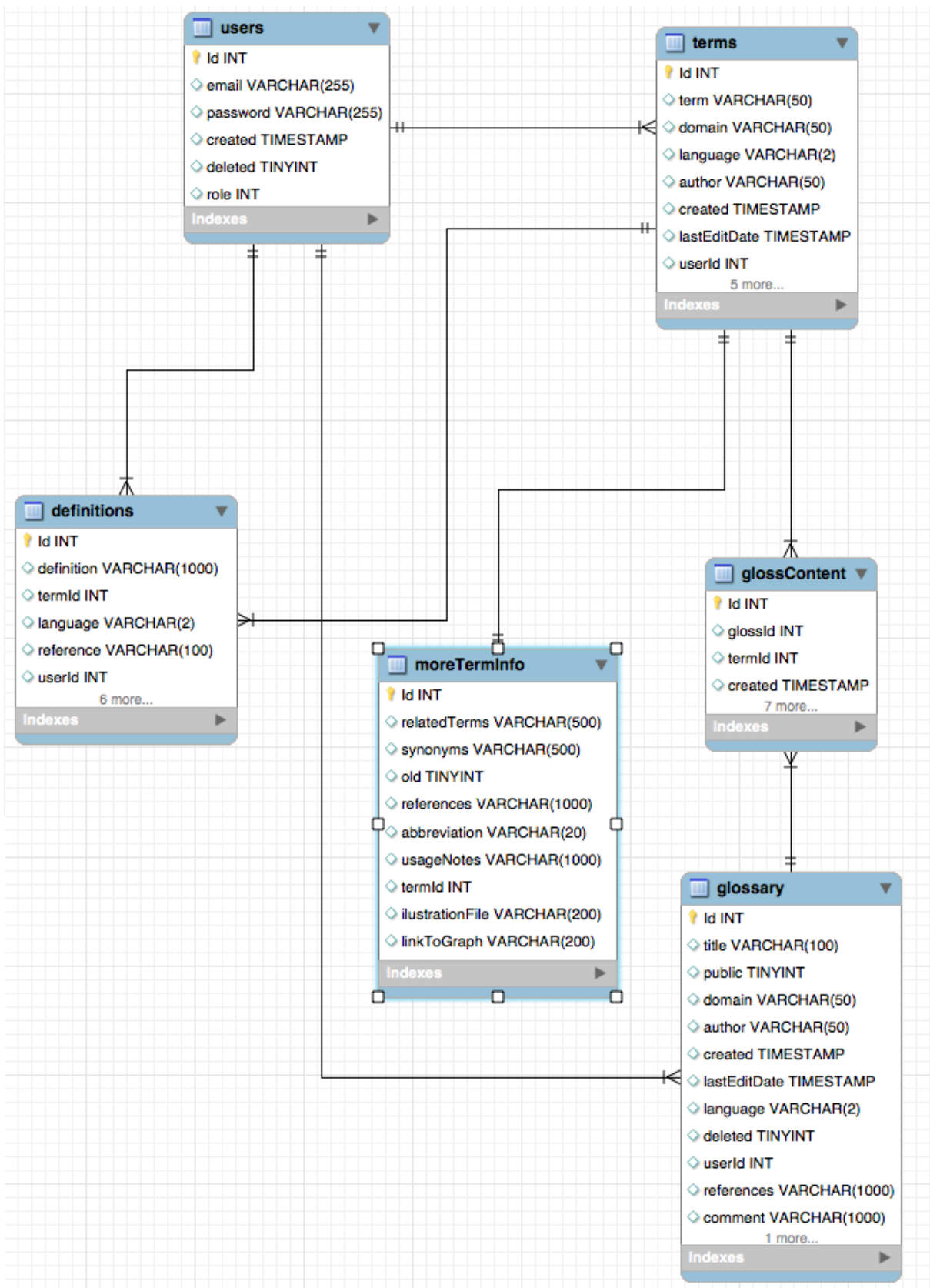
Tabulā “terms” jeb termini paredzēts glabāt sistēmā pielikto terminu datus. Tabula satur informāciju par termina vārdu, zināšanu apgabalu un valodu. Šāda tabulas struktūra nodrošinās ērtu terminu meklēšanas mehānismu, kur lietotājs varēs meklēt terminu ne tikai pēc tā vārda, bet varēs meklētājā norādīt zināšanu sfēru un termina valodu. Kā arī tabulā parādās informācija par termina pievienošanas datumu un pēdējās atjaunošanas datumu.

Tabulā “definitions” jeb definīcijas glabāsies informācija par termina skaidrojumu jeb definīciju. Pētot dažādus glosārijus un terminu vārdnīcas darba autors sastapās ar terminiem, kuriem pievienoti vairāki skaidrojumi. Tāpēc lai nodrošinātu, to ka sistēma spētu uzturēt vairākus terminu skaidrojumus ir izveidota atsevišķa tabula termina skaidrojumiem.

Tabula “moreTermInfo” jeb vairāk informācijas par terminu. Šī tabula paredzēta, lai glabātu papildus datus par terminu, ja tādi pieejami. Informācija par termina sinonīmiem, citiem saistītiem terminiem, abreviatūra, lietošanas piemēri, atsauces un vai termins ir/nav novecojis. Šāda tabulas struktūra nodrošina to, ka informācija par terminu ir detalizēta un ļauj importēt sistēmā glosārijus, kas rakstīti kādā no glossXML pieraksta formātiem.

Visās tabulās, kur izmantots lauks “language” valodas norādīšanai tiek izmantots ISO 639-1 standarts, kas apzīmē valsti ar diviem burtiem, piemēram, Latvija “lv”, Vācija “de”. Šāds standarts tiek lietots TBX, MARTIF un OLIF pieraksta formātos.

Datu bāzes modelis apskatāms attēlā 2.1.



2.1. att. Datubāzes modelis

### 3. GLOSĀRIJU IS IZSTRĀDE

Šajā nodaļā aprakstīta bakalaura darba praktiskā daļa, tas ir glosāriju sistēmas izstrāde balstoties uz pirmās un otrās nodaļas iegūto informāciju. Praktiskajā daļā darba autors uzstādīja šādus uzdevumus:

- izstrādāt reģistrāciju un pieteikšanos sistēmā;
- izstrādāt glosāriju pārvaldi, kur lietotājs var apskatīt gan publiski pieejamos glosārijus, gan paša lietotāja privātos glosārijus;
- izstrādāt glosāriju importēšanu sistēmā no tīmekļa resursiem nolasot un apstrādājot vietnes DOM saturu;
- izstrādāt glosāriju eksportēšanu, kas tiktu atbalstīta diviem failu tipiem, pdf un doc;
- izstrādāt manuālu terminu un glosāriju izveidi sistēmā;
- izstrādāt meklētāju, kas meklētu terminu informāciju sistēmas datu bāzē.

Tālākajās apakšnodaļās tiks aprakstīti veiktie uzdevumi un iegūtie rezultāti.

#### 3.1. Glosāriju izveide, pārvaldīšana un lietotāju autentifikācija

Viens no uzdevumiem, kas jāveic sistēmai ir glosāriju pārvaldīšana. Zem pārvaldīšanas ir paredzēts, ka lietotājiem būs iespēja redzēt veidotos glosārijus un manipulēt ar to datiem. Šādu iespēju nodrošināšanai katram glosārijam nepieciešams unikāls veidotāja identifikators pēc, kura vadoties ļaut lietotājam mainīt vai dzēst glosāriju saturu. Tāpēc tika pieņemts lēmums, ka sistēmai jānodrošina lietotāju izveide, kur katram lietotājam var tikt piesaistīti glosāriji, kas nodrošinātu glosārija pārvaldes mehānisma izstrādi.

Tā kā darba autors ir labi pazīstams ar PHP programmēšanas valodu sistēma tika bāzēta uz laravel ietvara. No viesiem pieejamajiem PHP ietvariem šobrīd laravel ir populārākais un viens no labākajiem ietvariem. Tāpēc tika izvēlēts tieši šis ietvars. Laravel nodrošina jau gatavu lietotāju autentifikācijas mehānismu un darba autoram bija nepieciešams tikai pielāgot šo mehānismu sistēmas datu bāzei. Kā arī ietvars atbalsta ērtu CRUD pieprasījumu veikšanu uz datu bāzi. Tādējādi ļaujot ērti veikt datu manipulācijas.

Glosāriju manuāla izveide tika nodrošināta ar formas palīdzību, kas apskatāmā 1. pielikumā “Manuāla glosārija pievienošanas forma”. Glosāriji tika iedalīti divās grupās, tas ir publiski pieejamie glosāriji un privātie glosāriji. Privātie glosāriji ir glosāriji, kas redzami un

pieejami tikai glosārija veidotājam, bet savukārt publiskie glosāriji ir pieejami apskatei viesiem sistēmas lietotājiem. Terminu atlase parādīta 2. pielikumā “Terminu atlasīšana glosārijam” un pievienoto terminu apskate parādīta 3. pielikumā “Pievienoto terminu pārskats”.

Rezultātā tika paveikti šādi uzdevumi: pilnībā izstrādāta reģistrācija un autentifikācija, daļēji izstrādāta glosāriju pārvalde un izstrādāta manuāla glosāriju pievienošana.

## **3.2. Glosārija eksportēšana**

Apskatot un pētot tīmekļa resursos pieejamos glosārijus un skaidrojošās vārdnīcas, tika novērots, ka neviens no apskatītajiem resursiem nepiedāvā eksportēšanas iespējas. Šāda funkcionalitāte izceltu glosāriju sistēmu starp esošajiem risinājumiem. Tika nolemts, ka sistēmai nepieciešams nodrošināt glosāriju eksportēšanu doc un pdf formātos. Šādi formāti tika izvēlēti, jo tie ir populārākie failu formāti, kas tiek lietoti dokumentu veidošanā.

### **3.2.1. Glosārija eksportēšana .DOC formātā**

Ir pieejamas vairākas metodes, kā ar PHP programmēšanas valodu veidot doc formāta failus. Darba autors apskatīja populārākos veidus, lai izvēlētos labāko un piemērotāko glosāriju informācijas sistēmai.

Viena no metodēm kā izveidot doc formāta failu būtu izmantojot HTTP galvenes vērtības. Metode lietojama gan priekš Unix gan priekš Windows operētājsistēmām. Viens no trūkumiem ir ļoti minimālas formatēšanas iespējas [21]. Attēlā 3.1. redzams piemērs, kā veidot doc formāta failus ar HTTP metodi.

```

<?php
header("Content-type: application/vnd.ms-word");
header("Content-Disposition: attachment;Filename=document_name.doc");

echo "<html>";
echo "<meta http-equiv='Content-Type' content='text/html; charset=Windows-1252'>";
echo "<body>";
echo "<b>My first document</b>";
echo "</body>";
echo "</html>";
?>

```

### 3.1. att. Doc faila veidošanas piemērs ar HTTP metodi

HTTP galvenes metodē tiek definēts faila formāts, kas konkrētajā gadījumā ir ms-word. Pēc tam kad galvenes nedefinētas seko dokumenta saturs. Kā redzams attēlā 3.1. šīs metodes teksta formatēšanas iespējas ir ļoti minimālas. Nav iespējams specificēt teksta fontu vai kā citādi ietekmēt teksta izkārtojumu.

Nākamā metode, ko vārētu izmantot doc failu veidošanai ir COM objekts - tehnoloģija, kas nodrošina objektu mijiedarbību ar datoru. Lai veidotu dokumentu ar šo metodi nepieciešams, lai uz servera, kur tiek darbināts kods būtu uzstādīta MS Word lietojumprogramma. Sākumā tiek veidots COM objekts un definēts jauns dokuments, tālāk šim dokumentam tiek pievienots vēlamais saturs pēc kā dokuments ir gatavs un var tikt lejupielādēts. Galvenais metodes trūkums ir tas, ka metode atbalsta tikai Windows operētājsistēmas [21].

Dokumentu izveide iespējama arī ar OpenOffice veidnēm. Nepieciešams manuāli izveidot veidni, kas saturēs vietturus, piemēram, [%aizvietojamā-vērtība%]. Pēc datu ievietošanas veidnē nepieciešams veikt vairākas konvertācijas, lai gala rezultātā iegūtu nepieciešamo doc formāta failu. Galvenais metodes trūkums ir konvertāciju veikšana, kas liela apjoma failiem nav efektīva un var būt laikietilpīga [21].

Apskatot doc formāta failu veidošanas metodes darba autors atrada bibliotēku bāzētu uz PHP programmēšanas valodu. Bibliotēkas nosaukums ir PHPWord, kas atrodas zem LGPL licences un ir pieejama brīvai lietošanai un ievietošanai sistēmās. Bibliotēka nodrošina plašas iespējas dokumenta formatēšanā. Ar šo bibliotēku dokumentā iespējams veidot klikšķināmas saites, kas ir ļoti liels plus tā kā termini un definīcijām pieejamas atsauces. Galvenās priekšrocības ir dinamiska dokumentu veidošana izmantojot PHP programmēšanas valodu un pieejama laba dokumentācija pēc kuras vadoties bez grūtībām iespējams izveidot doc formāta failu ar vēlamo formatējumu [18].

Rezultātā pēc apskatītajām metodēm tika izveidota glosāriju eksportēšana doc formāta failā ar PHPWord bibliotēku. Attēlā 3.2. redzams koda piemērs datu eksportēšanai. Un 4.

pielikumā “Glosārija pārskats pirms eksportēšanas” attēlots glosārija pārskats, kas atbilst eksportējamajai struktūrai dokumentā. Izvēloties kādu no formātiem lietotājs iegūs eksportētu glosāriju norādītajā formātā.

```
$phpWord = new \PhpOffice\PhpWord\PhpWord();
$section = $phpWord->addSection();
$textRun = $section->addTextRun();

$textRun->addText(
    htmlspecialchars($title),
    array('name' => 'TimesNewRoman', 'size' => 14, 'bold' => true),
    array('align' => 'center')
);
$textRun->addTextBreak(3);

$prevId = null;
foreach($query as $key => $val) {
    if($prevId != $val->termId) {
        if($prevId != null) {
            $textRun->addTextBreak(2);
        }
        $textRun->addText(
            htmlspecialchars('Termins: '.$val->term),
            array('name' => 'TimesNewRoman', 'size' => 12)
        );
        $textRun->addTextBreak(2);
        $textRun->addText(
            htmlspecialchars('Definīcija: '.$val->definition),
            array('name' => 'TimesNewRoman', 'size' => 12)
        );
    } else {
        $textRun->addTextBreak(2);
        $textRun->addText(
            htmlspecialchars('Definīcija: '.$val->definition),
            array('name' => 'TimesNewRoman', 'size' => 12)
        );
        $textRun->addTextBreak();
    }
    $prevId = $val->termId;
}

if(!file_exists(public_path().'glossary')) {
    mkdir(public_path().'glossary', 0777, true);
}
$objWriter = \PhpOffice\PhpWord\IOFactory::createWriter($phpWord, 'Word2007');
$fileName = $id.'-'.time().'.docx';
$objWriter->save(public_path().'glossary/'.$fileName);
```

### 3.2. att. Piemērs doc formāta glosārija eksportēšanai

Piemērā redzams kā eksportēt glosāriju doc formātā. Sākumā tiek izveidota dokumenta daļa, kura saturēs vairākus teksta ierakstus. Pirmais tiek ievietots nosaukums ar “TimesNewRoman”, 14pt un izcēluma formatējumu. Tālāk dokumentā tiek ievietoti glosārija termini un definīcijas. Kad viss saturs ievietots fails tiek saglabāts servera mapē, no kurienes

jau lietotājs var lejupielādēt eksportēto failu. Glosārija formāts šajā gadījumā būtu: glosārija nosaukums, terminu un to definīciju saraksts.

### **3.2.2. Glosārija eksportēšana pdf formātā**

Ir vairākas iespējas kā eksportēt glosārijus pdf formātā. Darba autors apskatīs vairākus risinājumus, lai izvēlētos labāko glosāriju informācijas sistēmai.

Pašā PHP programmēšanas valodā ir pieejama PDFlib bibliotēka, kas nodrošina pdf failu veidošanu. Bibliotēka ir novecojusi un tās dokumentācija nav pietiekama, lai saprastu cik sarežģītus pdf failus iespējams veidot. Kā izrādījās ar šo bibliotēku iespējams veidot primitīvus pdf dokumentus. Dokumentos iespējams ievietot tikai tekstu un bildes, nekāda cita veida dokumenta formatēšana netiek atbalstīta. Šis nebūtu labākais risinājums pdf dokumentu veidošanai.

Pētot pieejamos risinājumus pdf failu veidošanā tika atrastas divas bibliotēkas: FPDF un tFPDF, abas šīs bibliotēkas ir pieejamas brīvā izmantošanā [18], [19]. Bibliotēkas nodrošina pilnu kontroli pār veidojamo pdf failu un tajos attēlojamo saturu. Pēc funkcionalitātes bibliotēkas var teikt ir vienādas. Taču pēc tam, kad darba autors izmēģināja abas bibliotēkas pdf failu ģenerēšanā tika secināts, ka FPDF neatbalsta UTF-8 kodējuma formātu, līdz ar to glosāriji latviešu valodā netika korekti atspoguļoti. Savukārt tFPDF bibliotēka nodrošina UTF-8 kodējumu. Šis ir ļoti svarīgs moments, jo sistēma saturēs vairāku valodu terminus, kas nozīmē, ka veidojamajiem pdf failiem jāatbalsta visu valodu simbolu kodējums. Kā vēl vienu tFPDF bibliotēkas pārākumu var pieminēt, to ka dokumentā iespējams ievietot klikšķināmas saites uz ārējiem avotiem.

Pēc veiktajiem pētījumiem un mēģinājumiem darba autors izvēlējās izmantot tFPDF bibliotēku glosāriju eksportēšanai pdf formātā. Zemāk attēlā 3.3. redzams glosārija eksportēšanas piemērs.

---

```

$pdf = new \FPDF('P','mm','A4');
$pdf->AddPage();
$pdf->AddFont('DejaVu', '', 'DejaVuSansCondensed.ttf', true);
$pdf->SetFont('DejaVu', '', 14);
$pdf->Cell(0, 10, $title, 0, 1, 'C');
$pdf->Ln(10);

$prevId = null;
$pdf->SetFont('DejaVu', '', 12);
foreach($query as $key => $val) {
    if($prevId != $val->termId) {
        if($prevId != null) {
            $pdf->Ln(6);
        }
        $pdf->MultiCell(0, 10, 'Termins: '.$val->term, 0, 1, "");
        $pdf->Ln(8);
        $pdf->MultiCell(0, 10, 'Definīcija: '.$val->definition, 0, 1, "");
        $pdf->Ln(6);
    } else {
        $pdf->MultiCell(0, 10, 'Definīcija: '.$val->definition, 0, 1, "");
    }
    $prevId = $val->termId;
}

if(!file_exists(public_path().'glossary')) {
    mkdir(public_path().'glossary', 0777, true);
}

$fileName = $id.'-'.time().'.pdf';
$pdf->Output(public_path().'glossary/'.$fileName, 'F');

```

### 3.3. att. Piemērs pdf formāta glosārija eksportēšanai

Piemērā redzams kā eksportēt glosāriju pdf formātā. Sākumā tiek izveidots pdf dokumenta objekts. Dokumentam tiek nodefinēts teksta formatējums un ievietots glosārija nosaukums, pēc tam tiek ievietoti termini un to definīcijas. Kad pdf faila saturs ievietots, fails tiek saglabāts servera mapē un lietotājs var lejupielādēt eksportēto glosāriju.

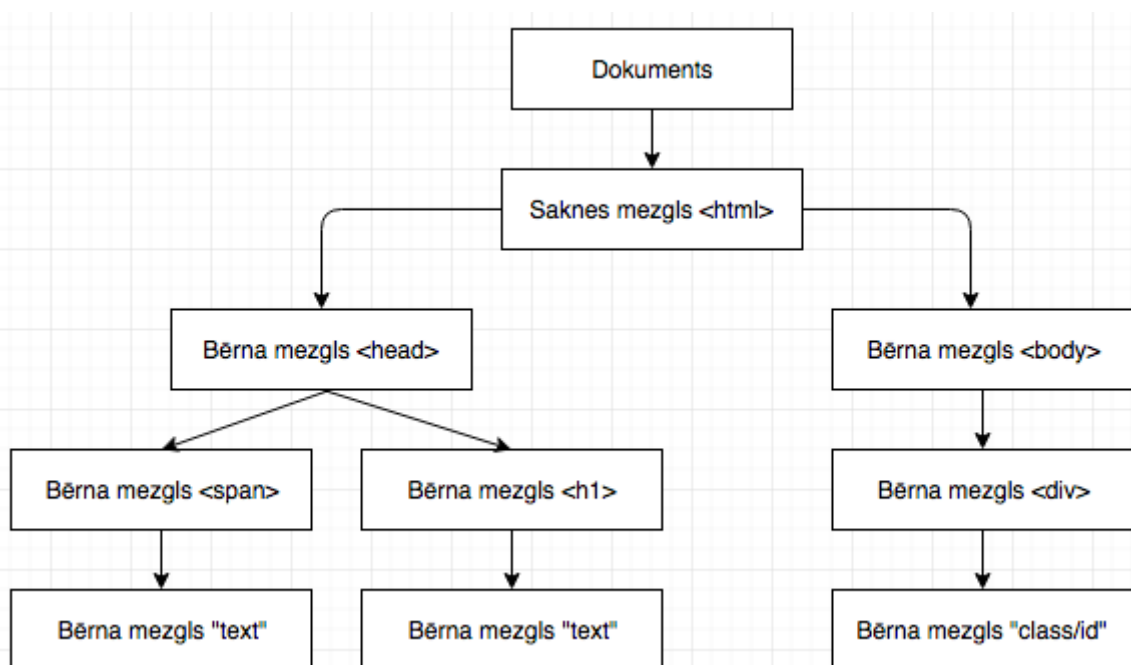
### 3.3. Glosāriju importēšana

Sistēmas datu bāzes papildināšanai un lietotāju ērtībām viens no uzdevumiem bija izpētīt un izveidot glosāriju importēšanas mehānismu, kas spētu apstrādāt glosārijus. Tā kā vairums glosāriju un terminu pieejami tīmekļa resursos tika nolemts izveidot importēšanas mehānismu, kas spētu apstrādāt tīmekļa vietnes. Tas nozīmē, ka importēšanas mehānismam jāspēj apstrādāt DOM elementus izgūstot nepieciešamo informāciju un saglabājot to datu bāzē. Šī mehānisma izstrāde nav triviāla, to ietekmē vairāki faktori, tīmekļa resursos ievietotie glosāriji netiek veidoti pēc vienota standarta, DOM saturs ne vienmēr ir korekts un derīgs

apstrādei, nepieciešami atbilstoši rīki DOM elementu apstrādei un nav iespējams pilnīgi automatizēt glosāriju importēšanu no tīmekļa resursiem.

### 3.3.1. DOM saturs koncepcija

Lai spētu strādāt ar DOM elementiem nepieciešams saprast to, ka viss DOM saturs ir veidots no mezgliem. DOM dokumentam ir hierarhiska koka struktūra, kas sastāv no mezgliem. Tas sastāv no saknes mezgla, kurš var saturēt bērnu mezglus un kur katrs bērns mezgls var saturēt citus bērnu mezglus. Attēlā 3.4. redzams ilustrējums DOM dokumenta kokam.



3.4. att. DOM elementu koks

Dokuments sastāv no elementiem, kā piemēram, `<html>`, `<head>`, `<body>`, `<div>`, `<span>` un `<h1>`. Katrs elements var saturēt atribūtus vai tekstu. Dotajā piemērā atribūts ir “class/id” `<div>` elementam, bet teksts ir `<h1>` un `<span>` elementiem, diagrammā apzīmēts ar “text”.

### 3.3.2. DOM satura apstrāde

Darba autors apskatīja dažādas bibliotēkas DOM elementu apstrādei, kas bāzētas uz PHP programmēšanas valodas. Lielākā daļa bibliotēku jau iebūvētas pašā PHP un nav nepieciešams uzstādīt nekādus papildinājumus, lai apstrādātu DOM saturu.

Populārākās no tām ir DOM, XMLReader, XML Parser un SimpleXML. Šīs bibliotēkas ļauj apstrādāt DOM saturu un piekļūt meklētajiem elementiem, ja ir zināms atlasāmais identifikators. Taču pētot un mēģinot bibliotēkas DOM elementu apstrādē, tika atklātas divas problēmas. Ne visas bibliotēkas spēja korekti izgūt tīmekļa vietnes saturu līdz ar to tās nespēja apstrādāt DOM elementus. Nekorektums izpaudās tajā, ka HTML elementiem trūka aizverošie elementi, kas noveda pie tā ka mezglu koks veidojās nekorekts. Viena no bibliotēkām DOM spēja apstrādāt arī nekorektu HTML kodu, bet darba autors nolēma papētīt vēl citus pieejamos risinājumus.

Rezultātā tika atrasta PHPHtmlParser [20] bibliotēka, kas ir viegli saprotama un elastīga izstrādē. Šī bibliotēka spēj apstrādāt, kā korektu tā arī nekorekti neformētu HTML kodu. Tā nodrošina ērtu pārvietošanos pa DOM elementiem uztverot tos, kā mezglus. Funkcijas nodrošina datu nolasi gan no failiem, gan no tīmekļa vietnēm. Tāpēc tika pieņemts lēmums veidot importēšanas mehānismu ar PHPHtmlParser bibliotēku, jo nākotnē glosāriju informāciju sistēma varētu atbalstīt arī glosāriju importēšanu no failiem, kas noformēti kādā no XML pieraksta formātiem.

Kā jau nodaļas sākumā tika minēts importējot glosārijus no tīmekļa resursiem nav iespējams visu automatizēt, jo glosāriji nav veidoti ne pēc viena no XML glosārija pieraksta formātiem. Tāpēc lietotājam nepieciešams patstāvīgi ievadīt šādus datus: informāciju par glosāriju, vietni no kuras importējams glosārijs, glosārija struktūra un galvenais DOM elements, kas satur paša glosārija saturu. Attēlā 3.5. redzams glosārija importēšanas koda fragments.

```

if(filter_var($url, FILTER_VALIDATE_URL) && count($structure) > 1) {
    if($glossary = Glossary::create($form)) {
        $dom = new Dom();
        $dom->loadFromUrl($url);
        $dom->setOptions([
            'removeScript' => true,
            'preserveLineBreaks' => true,
            'removeStyles' => true
        ]);

        $main = $dom->getElementById($content);
        $tag = $this->getTag($main->innerHTML);
        $arr = explode($tag, $main->innerHTML);

        foreach($arr as $key => $val) {
            $save = true;
            if(strlen($val) >= 7) {
                $elem = $dom->loadStr($val, []);
                $term = [];
                foreach($structure as $k => $v) {
                    if($k == 0) {
                        $elem = $this->getFirstNode($elem);
                        if($v != 'definition') {
                            $term[$v] = $elem->innerHTML;
                            if($term[$v] != strip_tags($term[$v])) {
                                $save = false;
                            }
                        } else {
                            $def[$v] = $elem->innerHTML;
                            if($def[$v] != strip_tags($def[$v])) {
                                $save = false;
                            }
                        }
                    } else {
                        $elem = $this->getNextNode($elem);
                        ...
                    }
                }
            }
        }
    }
}

```

3.5. att. Glosārija importēšanas koda fragments

Attēlā attēlots fragments no glosārija importēšanas mehānisma. Sākumā tiek validēta norādītā saite no kuras jāimportē glosārijs. Tālāk tiek ielādēts saites DOM saturs un izgūstot galveno saturu tiek atfiltrēts viss liekais, tas ir skripti, formatējumi un tukšās rindas. Tālāk ejot cauri visam saturam tiek veidoti dati saglabāšanai, tas tiek panākts ar funkcijām `getFirstNode` un `getNextNode`. Tādējādi pārvietojoties pa DOM koku. Termina atribūti tiek izgūti no masīva `$structure`, ko iepriekš nedefinē lietotājs pirms glosārija importēšanas.

### 3.4. Terminu pievienošana un meklēšana

Sistēmas izstrādē tika uzstādīts uzdevums par terminu pievienošanu un meklēšanu sistēmā. Terminu pievienošanai tika izveidota forma, kas satur visus informācijas laukus, kurus atbalsta un uztur sistēmas datu bāze. Lietotājs manuālā ceļā var pievienot jaunus terminus sistēmā. Pievienošanas forma apskatāma 5. Pielikumā „Terminu pievienošana. Kreisā puse” un 6. Pielikumā „Terminu pievienošana. Labā puse”, kā redzams lietotājs var ievadīt pamatinformāciju par terminu un ja nepieciešams var norādīt termina papildus informāciju izvēloties pogu „Papildus informācija”. Papildus informācijā lietotājs var norādīt termina avotus, sinonīmus un citu informāciju. Labajā pusē lietotājs var pievienot definīcijas, kā jau tas tika minēts sistēma atbalsta un uztur vairākus terminu skaidrojumus.

Terminu meklēšana sistēmā tika nodrošināta līdzīgā veidā. Lietotājam ir pieejama forma ar trim ievadlaukiem, termina vārds, zināšanu sfēra un valoda. Lietotājam ir iespēja meklēt terminu pēc pāris zīlēm, norādot tā valodu vai zināšanu apgabalu. Atrastie rezultāti tiek attēloti lietotājam ērtā veidā, tiek parādīts saraksts ar atrastajiem terminiem un lietotājam ir iespēja izvēlēties kādu no terminiem un apskatīt detalizētu informāciju par to. Meklētājs apskatāms 7. pielikumā „Terminu meklētājs”. Lietotājs var meklēt terminu pēc frāzes, norādot vēlamu valodu, vai zināšanu apgabalu.

## REZULTĀTI UN SECINĀJUMI

Bakalaura darba mērķis bija izstrādāt strādājošu glosāriju informācijas sistēmu, kas spētu uzturēt glosārijus un to terminus.

Darba izstrādes laikā tika apskatīti un izpētīti šādi glosāriju pieraksta formāti TBX, MARTIF, OLIF, GlossML un GlossXML. Labākai izpratnei par formātu struktūru tika veidotas hierarhiskas shēmas, kas attēloja datus, kurus var saturēt glosārijs. Pēc šī etapa tika apskatīti tīmeklī pieejamie risinājumi glosāriju uzturēšanai. Tika apskatīti risinājumu plusi un mīnusi secinājumu veidā, kas palīdzēja izvirzīt paveicamos uzdevumus praktiskajā daļā. Pirmās nodaļas beigās tika veikta iegūtās informācijas analīze un rezultātā uzzīmēta hierarhiska datu struktūras shēma. Ar iegūtās shēmas un informācijas palīdzību tika izplānota sistēmas datu bāze, kas spētu veiksmīgi uzturēt ne tikai terminus un to skaidrojumus, bet papildus informāciju, tādu kā, termina sinonīmi, avots, lietošanas piemēri un zināšanu apgabals kuram pieder termins.

Datu bāzes struktūra tika plānota tā, lai terminus varētu uzturēt vairākās valodās, ar vairākām definīcijām un tā lai nākotnē nerastos problēmas papildināt datu bāzes tabulas ar citu papildinformāciju. Šīs nodaļas rezultātā tika radīts datu bāzes modelis priekš glosāriju informācijas sistēmas.

Kad tika iegūta visa nepieciešamā informācija darba autors pārgāja pie darba praktiskās daļas, tas ir sistēmas izstrādes. Sistēma tika veidota tā, lai lietotājiem būtu iespēja patstāvīgi veidot glosārijus un terminus. Tika izveidotas atbilstošas formas ar kuru palīdzību lietotājs sistēmā var pievienot jaunus terminus un glosārijus. Glosāriju eksportēšanas mehānisms tika pilnīgi automatizēts un rezultātā lietotājs var iegūt glosārijus .DOC vai .PDF formātā. Veidojot eksportēšanas mehānismu darba autors meklēja bibliotēkas, kas dotu maksimālu kontroli pār dokumentu, ar domu nākotnē piedāvāt dažāda tipa eksporta veidus. Ar glosāriju importēšanu sistēmā gāja grūtāk. Jau pašā sākumā darba autors saskārās ar grūtībām atrast piemērotu PHP bibliotēku, kas spētu apstrādāt DOM elementus. Pēc ilgām meklējumiem tika atrasta piemērota bibliotēka, ar kuras palīdzību izdevās realizēt ne ļoti sarežģītu importēšanas mehānismu. Protams nākotnē būtu nepieciešams piestrādāt pie glosāriju importēšanas un nodrošināt arī importēšanas iespējas ne tikai no tīmekļa resursiem. Kā arī nepieciešams pedomāt pie importēšanas mehānisma kopumā, lai tas būtu komplicētāks un spētu kādā mērā arī kontrolēt kļūdainus glosārijus.

Darba rezultātā tika izveidota strādājoša glosāriju informāciju sistēma, taču tā nav pilnīgi pabeigta un ir nepieciešami uzlabojumi. Kā izrādījās bija jāpatērē diezgan liels laiks,

lai izpētītu glosārija pieraksta formātus, izplānotu datu struktūru un atrastu nepieciešamos rīkus sistēmas izstrādei. Taču visi darba mērķi lielākā vai mazākā mērā tika sasniegti. Izstrādājot bakalaura darbu autors guva izpratni un zināšanas par glosāriju struktūru un to pieraksta veidiem, kā arī izstrādātā sistēma deva papildus pieredzi PHP programmēšanas rīku izmantošanā.

## IZMANTOTĀ LITERATŪRA UN AVOTI

1. Akadēmiskā terminu datubāze AkadTerm. [tiešsaiste]. [atsauce 09.04.2016.]. Pieejams: <http://termini.lza.lv/>
2. Ontology Structure | Gene Ontology Consortium [tiešsaiste]. [atsauce 09.04.2016.]. Pieejams: <http://geneontology.org/page/ontology-structure>
3. Alan K. Melby, Brigham Young University, „Terminology in the age of multilingual corpora”, 2012. g. [tiešsaiste]. [atsauce 09.04.2016.]. Pieejams: [http://www.jostrans.org/issue18/art\\_melby.php](http://www.jostrans.org/issue18/art_melby.php)
4. Introduction to TermBase eXchange (TBX) – TBX overview [tiešsaiste]. [atsauce 12.04.2016.]. Pieejams: <http://www.tbxinfo.net/>
5. Introduction to TermBase eXchange (TBX) – TBX Dialects [tiešsaiste]. [atsauce 12.04.2016.]. Pieejams: <http://www.tbxinfo.net/tbx-dialects/>
6. Glossary of Industry Related Terms - MARTIF [tiešsaiste]. [atsauce 12.04.2016.]. Pieejams: <http://www.creativyst.com/cgi-bin/M/Glos/st/GetTerm.pl?fsGetTerm=MARTIF>
7. The open XML language data standart [tiešsaiste]. [atsauce 14.04.2016.]. Pieejams: <http://www.olif.net/>
8. Rodolfo Raya, „Glossary Markup Language, An XML Representation for Glossaries” [tiešsaiste]. [atsauce 14.04.2016.]. Pieejams: <http://www.maxprograms.com/articles/glossml.html>
9. GlossML 1.0 Specification [tiešsaiste]. [atsauce 14.04.2016.]. Pieejams: <http://www.maxprograms.com/glossml/glossml.pdf>
10. Definitions from WhatIs.com [tiešsaiste]. [atsauce 18.04.2016.]. Pieejams: <http://whatis.techtarget.com/definitions/B/page/4>
11. Bis.org Glossary [tiešsaiste]. [atsauce 18.04.2016.]. Pieejams: <http://www.bis.org/cpmi/publ/d00b.htm?&selection=112&scope=CPMI&c=a&base=term>
12. Tēzauris [tiešsaiste]. [atsauce 18.04.2016.]. Pieejams: <http://www.tezaurs.lv/>
13. Academic.ru [tiešsaiste]. [atsauce 18.04.2016.]. Pieejams: <http://dic.academic.ru/>
14. Convertible TBX [tiešsaiste]. [atsauce 20.04.2016.]. Pieejams: <http://www.ttt.org/tbxg/convertibility/tbxg.html>
15. Convertible OLIF [tiešsaiste]. [atsauce 20.04.2016.]. Pieejams: <http://www.ttt.org/tbxg/convertibility/olf.html>
16. John Repici, „Proposed XML Format for Glossaries” [tiešsaiste]. [atsauce 20.04.2016.]. Pieejams: <http://www.creativyst.com/Prod/Glossary/Doc/XMLOut.htm>

17. Klaus-Dir Schmitz, „Proposed XML Format for Glossaries”, 2015. g. [tiešsaiste]. [atsauce 20.04.2016.]. Pieejams: [https://www.researchgate.net/publication/237303525\\_TERMINOLOGY\\_INTERCHANGE\\_USING\\_MARTIF](https://www.researchgate.net/publication/237303525_TERMINOLOGY_INTERCHANGE_USING_MARTIF)
18. PHPWord bibliotēka [tiešsaiste]. [atsauce 5.05.2016.]. Pieejams: <https://github.com/PHPOffice/PHPWord>
19. fFPDF bibliotēka [tiešsaiste]. [atsauce 5.05.2016.]. Pieejams: <http://www.fpdf.org/en/script/script92.php>
20. PHP parser [tiešsaiste]. [atsauce 12.05.2016.]. Pieejams: <https://github.com/paquetteg/php-html-parser>
21. Creating Word, Excel and CSV files with PHP [tiešsaiste]. [atsauce 12.05.2016.]. Pieejams: [http://webcheatsheet.com/php/create\\_word\\_excel\\_csv\\_files\\_with\\_php.php](http://webcheatsheet.com/php/create_word_excel_csv_files_with_php.php)
22. Glossary of system theory[tiešsaiste]. [atsauce 12.05.2016.]. Pieejams: [en.wikipedia.org/wiki/Glossary\\_of\\_systems\\_theory](en.wikipedia.org/wiki/Glossary_of_systems_theory)

# PIELIKUMI

## 1. pielikums Manuāla glosārija pievienošanas forma

Glosārija nosaukums:

Publiski pieejams:

Glosārija ziāšanu sfēra(Domain):

Glosārija autors:

Glosārija valoda:

Atsauces:

Glosārija apraksts:

Izveidot

## 2. pielikums Terminu atlasīšana glosārijam

Atlasiet terminus priekš jūsu glosārija

Apskatīt atlasītos terminus

Ievadiet meklējamo terminu:

**Termins:** Testēšana

Zināšanu sfēra(Domēns): Testēšana

Autors:

Izvēlēties

**Definīcija:**

1:

Programmatūras un aparatūras darbības pārbaude, izmantojot testdatus. Testēšanas mērķis var būt defekta atklāšana, tā atrašanās vietas lokalizēšana vai arī testējamā objekta dinamisko parametru (piem., ātrdarbības) noskaidrošana.

Avots: LZA TK ITTEA terminu datubāze

**Vairāk par terminu:**

Saistīti termini:

Sinonīmi:

Novecojs: nav

Atsauces:

Abreviatūra:

Lietošanas piemēri:

Ilustrācija: [apskatīt](#)

Grafis:

**Termins:** Akcepttests

Zināšanu sfēra(Domēns): Testēšana

Autors:

Izvēlēties

### 3. pielikums Pievienoto terminu pārskats

Atlasiet terminus priekš jūsu glosārija

Apskatīt atlasītos terminus

**Termins:** Testēšana

Zināšanu sfēra(Domēns): Testēšana

Autors:

Dzēst

**Termins:** Akcepttests

Zināšanu sfēra(Domēns): Testēšana

Autors:

Dzēst

### 4. pielikums Glosārija pārskats pirms eksportēšanas

[Eksportēt kā doc](#) [Eksportēt kā pdf](#)

#### Testēšanas glosārijs

**Termins:** Testēšana

**Definīcija:** Programmatūras un aparatūras darbības pārbaude, izmantojot testdatus. Testēšanas mērķis var būt defekta atklāšana, tā atrašanās vietas lokalizēšana vai arī testējamā objekta dinamisko parametru (piem., ātrdarbības) noskaidrošana.

**Termins:** Akcepttests

**Definīcija:** Programmatūras vai aparatūras galīgā pārbaude, kas parāda izstrādātā produkta funkcionālās iespējas. Šī pārbaude nosaka, vai pārbaudāmais objekts darbojas atbilstoši līgumā fiksētajām prasībām (specifikācijai), t.i., vai tas atbilst akceptēšanas kritērijiem.

**5. pielikums**  
**Terminu pievienošana. Kreisā puse**

Informācija par terminu

Ievadiet terminu:

Zināšanu sfēra:

Izvēlieties valodu:

Termina autors:

+ Papildus informācija

Pievienot

**6. pielikums**  
**Terminu pievienošana. Labā puse**

Informācija par termina skaidrojumu

Ievadiet definīciju:

Izvēlieties valodu:

Definīcijas avots:

+ Pievienot vēl definīciju

## 7. pielikums Terminu meklētājs

Ievadiet meklējamo terminu:

Ievadiet zināšanu apgabalu:

Izvēlieties valodu:

**Termins:** Akcepttests  
**Zināšanu sfēra(Domēns):** Testēšana  
**Autors:**

Bakalaura darbs „Glosāriju informācijas sistēma” izstrādāts LU Datorikas fakultātē.

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

Autors: \_\_\_\_\_ Artūrs Bugins

Rekomendēju/nerekomendēju darbu aizstāvēšanai

Vadītāja: profesors Dr. dat. Guntis Arnicāns \_\_\_\_\_ 30.05.2016.

Recenzents: asociētais profesors Dr. dat. Uldis Straujums

Darbs iesniegts Datorikas fakultātē 30.05.2016.

Dekāna pilnvarotā persona: vecākā metodiķe Ārija Sproģe \_\_\_\_\_

Darbs aizstāvēts bakalaura gala pārbaudījuma komisijas sēdē

\_\_\_\_\_. prot. Nr. \_\_\_\_\_.

Komisijas sekretārs(-e): \_\_\_\_\_