

LATVIJAS UNIVERSITĀTE
DATORIKAS FAKULTĀTE

Tulkošanas kvalitātes vērtēšanas platforma

KVALIFIKĀCIJAS DARBS

Autors: **Kaspars Rumkovskis**

Studenta apliecības Nr.: kr19043

Darba vadītājs: B.dat. Artūrs Stafanovičs

RĪGA 2020

ANOTĀCIJA

Mašīntulkošanas modeļa izstrādē ir svarīgi noteikt, vai jaunizstrādātais modelis ir labāks par iepriekšējo versiju. Visprecīzākais veids, kā to noteikt, ir ļaut cilvēkvērtētājiem vērtēt katra modeļa tulkošanas kvalitāti, bet cilvēkvērtēšana ir dārgs un laikietilpīgs process, tāpēc šī novērtēšana ir jāveic pēc iespējas efektīvi.

Kvalifikācijas darba mērķis ir izstrādāt māšīntulkošanas kvalitātes vērtēšanas platformu, kas ļautu efektīvi un ātri cilvēkvērtētājiem novērtēt mašīntulka tulkotos teikumus un ar šīs informācijas palīdzību uzlabot mašīntulkošanas kvalitāti. Izstrādātā platforma ļauj pārvaldīt novērtēšanas kopas, veikt tiešo cilvēkvērtēšanu, analizēt, un eksportēt rezultātus. Izstrādātā sistēma atvieglos modeļu izstrādi un rezultātu analīzi, kā arī samazinās novērtēšanas izmaksas.

Atslēgas vārdi: mašīntulkošanas kvalitāte, cilvēkvērtēšanas platforma, statistiskā analīze, ASP.NET Core.

ABSTRACT

TRANSLATION QUALITY EVALUATION PLATFORM

During the machine model's creation it is important to know if the new and improved model is better than the last version. The most precise way to tell is to let human evaluators evaluate each model's translation quality but human evaluation is a long and expensive process, so this evaluation must be done as efficiently as possible.

The goal of this qualification work is to develop a machine translation quality evaluation platform which allows human evaluators to quickly and effectively evaluate machine translated sentences and with this information improve the machine translation quality. The created platform allows evaluation mark moderation, direct human evaluation and result analyzation and exportation. The created system will make model creation and data analysis easier and will decrease evaluation expenses.

Keywords: machine translation quality, human evaluation platform, statistical analysis, ASP.NET Core.

SATURS

1. IEVADS	7
2. APZĪMĒJUMI UN DEFINĪCIJAS	9
3. PROGRAMMATŪRAS PRASĪBU SPECIFIKĀCIJA	10
3.1. Ievads	10
3.1.1. Nolūks.....	10
3.1.2. Darbības sfēra	10
3.1.3. Definīcijas.....	10
3.1.4. Saistība ar citiem dokumentiem.	11
3.1.5. Pārskats.....	11
3.2. Vispārējs apraksts	11
3.2.1. Produkta perspektīva	11
3.2.2. Produkta funkcijas	12
3.2.3. Lietotāja raksturierzīmes.....	12
3.2.4. Vispārējie ierobežojumi.....	13
3.2.5. Pieņēmumi un atkarības.....	13
3.3. Funkcionālās prasības	14
3.3.1. Ievads.....	14
3.3.2. Lietotāju stāsti	14
3.4. Nefunkcionālās prasības	20
4. PROGRAMMATŪRAS PROJEKTĒJUMA APRAKSTS	21
4.1. Ievads	21
4.1.1. Nolūks.....	21
4.1.2. Darbības sfēra	21
4.1.3. Definīcijas.....	21
4.1.4. Saistības ar citiem dokumentiem.....	21

4.2.	Dekompozīcijas apraksts	21
4.2.1.	Paralēlo korpusu modulis	21
4.2.2.	Statistikas modulis	21
4.2.3.	Vērtēšanas uzdevumu modulis	22
4.3.	Detalizēts projektējums.....	22
4.3.1.	Funkciju sadalījums moduļos	22
4.3.2.	Paralēlo korpusu modulis	23
4.3.3.	Statistikas modulis.....	31
4.3.4.	Vērtēšanas uzdevumu modulis	35
4.3.5.	Nejaušu teikumu izvēles varbūtība.....	45
4.4.	Datubāzes projektējums	46
4.4.1.	Datubāzes konceptuālais entītiju relāciju modelis.....	46
4.4.2.	Fiziskais modelis	48
4.4.3.	Datubāzes tabulu apraksts	49
4.4.4.	Entītiju apraksts	49
4.5.	Saskarnes projektējums.....	55
4.5.1.	Ievads.....	55
4.5.2.	Sākumlapas skats.....	57
4.5.3.	Autentifikācijas lapas skats	58
4.5.4.	Vērtēšanas uzdevumu lapas skats.....	59
4.5.5.	Vērtēšanas uzdevuma pievienošanas lapas skats	60
4.5.6.	Paralēlā korpusa pievienošanas lapas skats	61
4.5.7.	Paralēlā korpusa pārvaldīšanas lapas skats.....	62
4.5.8.	Korpusa daļas pievienošanas lapas skats.....	63
4.5.9.	Teikumu faila pievienošanas lapas skats	64
4.5.10.	Korpusa daļu pārvaldīšanas lapas skats.....	65

4.5.11. Visu teikumu pārvaldīšanas lapas skats.....	66
4.5.12. Vērtēšanas uzdevuma pārvaldīšanas lapas skats	67
4.5.13. Vērtēšanas apakšuzdevuma pievienošanas lapas skats	68
4.5.14. Vērtēšanas apakšuzdevuma pārvaldīšanas lapas skats	69
4.5.15. Vērtēšanas uzdevuma statistikas lapas skats	70
4.5.16. Visu vērtēšanas uzdevumu statistikas lapas skats	72
4.5.17. Teikumu vērtēšanas lapas skats	73
5. TESTĒŠANAS DOKUMENTĀCIJA	74
5.1. Testēšanas metodika	74
5.2. Vienībtestēšana	74
5.3. Manuālie testi.....	75
6. PROJEKTA PĀRVALDĪBA	76
6.1. Projekta organizācija.....	76
6.2. Kvalitātes nodrošināšana	76
6.3. Konfigurāciju pārvaldība	76
6.4. Darbietilpības novērtējums	77
7. SECINĀJUMI	78
8. IZMANTOTĀ LITERATŪRA	79
9. PIELIKUMS	81

1. IEVADS

Mašīntulkošana ir automatizēta teksta tulkošana no avotvalodas uz mērķvalodu, kas mūsdienās pārsvarā tiek realizēts ar neironu tīklu modeļiem. Lai apmācītu neironu tīklu modeli, tam tiek dotas divas kopas – kopa ar avotvalodas tekstu un kopa ar atbilstošu cilvēkveidotu tulkojumu. Plašāk lietotie mašīntulkošanas modeļi strādā teikumu līmenī, kas nozīmē, ka ieejā tas saņem vienu teikumu un atgriež tā tulkojumu. Kvalifikācijas darbs apskata šādu teikuma-līmeņa neironu mašīntulkošanas sistēmu kvalitātes novērtēšanu.

Mašīntulkošanas modelis tiek iteratīvi trenēts līdz tas sasniedz kādu kritēriju, piemēram, tā kvalitāte neuzlabojas uz atsevišķi nodalītas izstrādes kopas. Viens veids, kā novērtēt modeļa kvalitāti, ir rēķināt automātiskās metrikas, piemēram, BLEU, kas ieejā saņem tulkojumu un referenci (“pareizais” cilvēka veidotais tulkojums) un rezultātā iegūst vērtību, kas novērtē mašīntulkošanas sistēmas kvalitāti. Modeļa apmācīšanai tiek lietotas automātiskās novērtēšanas metrikas tāpēc, ka tas ir ātrs un lēts novērtēšanas veids. Taču pētījumi [1] rāda, ka automātiskās metrikas slikti korelē ar cilvēkvērtējumiem augstas kvalitātes sistēmām, it īpaši tās metrikas, kuras ir balstās uz n-grammu salīdzināšanu. Tipiskākie automātisko metriku pielietošanas scenāriji ir sekošana kvalitātei apmācības laikā, kā arī modeļu salīdzināšana savā starpā, ja nav pieejami resursi dārgākiem novērtēšanas veidiem.

Mašīntulkošanas galvenais mērķis ir spēt tulkot tekstu cilvēka līmenī. Cilvēka tulkojumi tiek uzskatīti kā zelta standarts. Salīdzinot apmācītas sistēmas, ir vajadzība noskaidrot, kura no tām ir labākā – kura pēc iespējas vairāk līdzinās cilvēktulkojumam. Lai to izdarītu, tiek piesaistīti cilvēkvērtētāji, kas novērtē tulkošanas sistēmas tulkošanas kvalitāti. Cilvēkvērtēšana dod iespēju pateikt ar lielāku pārliecību, kuras mašīntulkošanas sistēma ir labāka, kā arī pamanīt niansētākas atšķirības, izstrādājot jaunas mašīntulkošanas metodes. Cilvēkvērtēšana ir ļoti dārga un lēna, tāpēc ir vajadzība resursus izlietot pēc iespējas efektīvi. Šim nolūkam tiek veidota novērtēšanas platforma.

Kvalifikācijas darba ietvaros Valodu tehnoloģiju uzņēmumam SIA “Tilde” tika izstrādāta “Vērtēšanas platforma”, kuras galvenie mērķi ir nodrošināt modernu un efektīvu cilvēkvērtēšanas metodoloģiju, ļaut iegūtos datus analizēt, kā arī aprēķināt automātiskās metrikas un tās salīdzināt ar cilvēkvērtējumiem.

Iepriekš šim nolūkam tika izmantota SIA “Tilde” izstrādātā Banzai platforma. Banzai ir sistēma, kurā cilvēkvērtētāji vērtē teikumu kvalitāti, izvēloties labāko starp diviem dotajiem tulkojumiem. Banzai cilvēkvērtēšana neatbilda WMT mašīntulkošanas sacensību mūsdienu metodoloģijai [2], kā arī trūka ērta datu pārvaldība, rezultātu vizualizācija un integrācija ar Tildes ekosistēmu. Kvalifikācijas darba ietvaros tika izstrādāta novērtēšanas platforma, kura ietver šādu funkcionalitāti:

- WMT stila cilvēkvērtēšana (direct assessment),
- Novērtēšanas kopu pārvaldība,
- Cilvēkvērtēšanas vizualizācija,
- Modernu automātisko metriku integrācija,
- Neuzticamu vērtētāju identificēšana,
- Statistiskā analīze:
 - Statistiskā nozīmība,
 - Automātisko metriku korelācija ar cilvēkvērtējumiem,
 - Vērtētāju vienprātība,
 - Grafīku attēlošana.

Programmatūra tika izstrādāta, ievērojot spējās programmatūras izstrādes metodoloģiju. Programmkoda izstrāde tika veikta Angular un C# vidē. Datu glabāšanai un uzturēšanai tika lietota MySQL datubāze. Lietotāju autentifikācijai tika lietota Keycloak autentificēšanās sistēma [3], kura glabā lietotājus savā datubāzē un lietotāja autentifikācijas datus glabā JWT html galvenē. Automātisko vērtēšanas metriku aprēķināšanas serveris tika izstrādāts Python programmatūras valodā un tas veido savienojumu, izmantojot websocket.

2. APZĪMĒJUMI UN DEFINĪCIJAS

2.1. tabula Apzīmējumi un definīcijas

Termins	Definīcija
AVMAS	Automātiskās vērtēšanas metriku aprēķināšanas serveris.
JWT	JSON tīkla marķieris (angl. JSON web token).
GUID	Globāli īpašs identifikators.
Hypothesis	Hipotēze - Ar mašīntulku iegūtais avotvalodas teikuma tulkojums.
Reference	Cilvēktulkota korpusa daļa, atsaucis korpusa daļa, reference.
Source	Avotvaloda
Likert	Likerta skala.
Direct Assessment	Tiešais novērtējums.
BLEU	BLEU (<i>Bilingual Evaluation Understudy</i>) ir novērtēšanas metrika, kas salīdzina vārdu pārklājumu tulkojumā un dotajā atsaucis tekstā
CHRF	CHRF (<i>character n-gram F-score</i>) ir novērtēšanas metrika, kas salīdzina simbolu n-grammas atsaucis teikumā ar hipotēzi.
TER	TER (<i>Translation error rate</i>) ir novērtēšanas metrika, kura mēra, cik labojumi jāveic tulkojumā, salīdzinot ar doto atsaucis tekstu.
COMET	COMET ir novērtēšanas metrika, kuru aprēķina, izmantojot mašīnāpmācītu novērtēšanas modeli.
Websocket	Atkārtoti uzturēts savienojums.

3. PROGRAMMATŪRAS PRASĪBU SPECIFIKĀCIJA

3.1. Ievads

3.1.1. Nolūks

Programmatūras prasību specifikācija (turpmāk tekstā – PPS) nolūks ir aprakstīt izstrādātās sistēmas funkcionālās un nefunkcionālās prasības. Dokuments paredzēts projektēšanas un programmēšanas darba grupai.

3.1.2. Darbības sfēra

Sistēma “Tulkošanas kvalitātes vērtēšanas platforma” ir teikumu uzglabāšanas, vērtēšanas un statistikas attēlošanas sistēma, kas ļauj vērtētājiem novērtēt mašintulkotus teikumus un ļauj lietotājiem apskatīties statistiku, piemēram, vidējais vērtējums, statistiskā nozīmība utt., gan par vērtētāju dotajiem vērtējumiem, gan par automātisko metriku vērtējumiem. Sistēmas administratori ir atbildīgi par paralēlā korpusa ielādēšanu sistēma, kas sastāv no avotvalodas korpusa daļas, mērķvalodas atsauces korpusa daļu un hipotēzes (sistēmas) korpusa daļas. Šis paralēlais korpus tiek piesaistīts vērtēšanas uzdevumam un tiek izveidoti vērtēšanas apakšuzdevumi, kuri tiek izdalīti vērtētājiem kā hipersaite. Vērtētāju uzdevums ir novērtēt dotajā skalā iztulkotā teikuma kvalitāti.

Sistēma sastāv no piecām daļām:

- Ārējās saskarnes,
- Datu apstrādes serveris,
- Datubāzes,
- AVMAS,
- Autentifikācijas serveris.

Sistēmai vairs nebūs pielietojuma, kad mašintulkošana tiks attīstīta līdz cilvēka tulka līmenim vai tiks atrasti jauni veidi, kā uzlabot mašintulkošanas darbību.

3.1.3. Definīcijas

Skatīt nodaļu Apzīmējumi un definīcijas

3.1.4. Saistība ar citiem dokumentiem.

Dokumenta noformēšanā izmantots Latvijas standarts LVS 68:1996 PPS “Programmatūras prasību specifikācijas ceļvedis” [4].

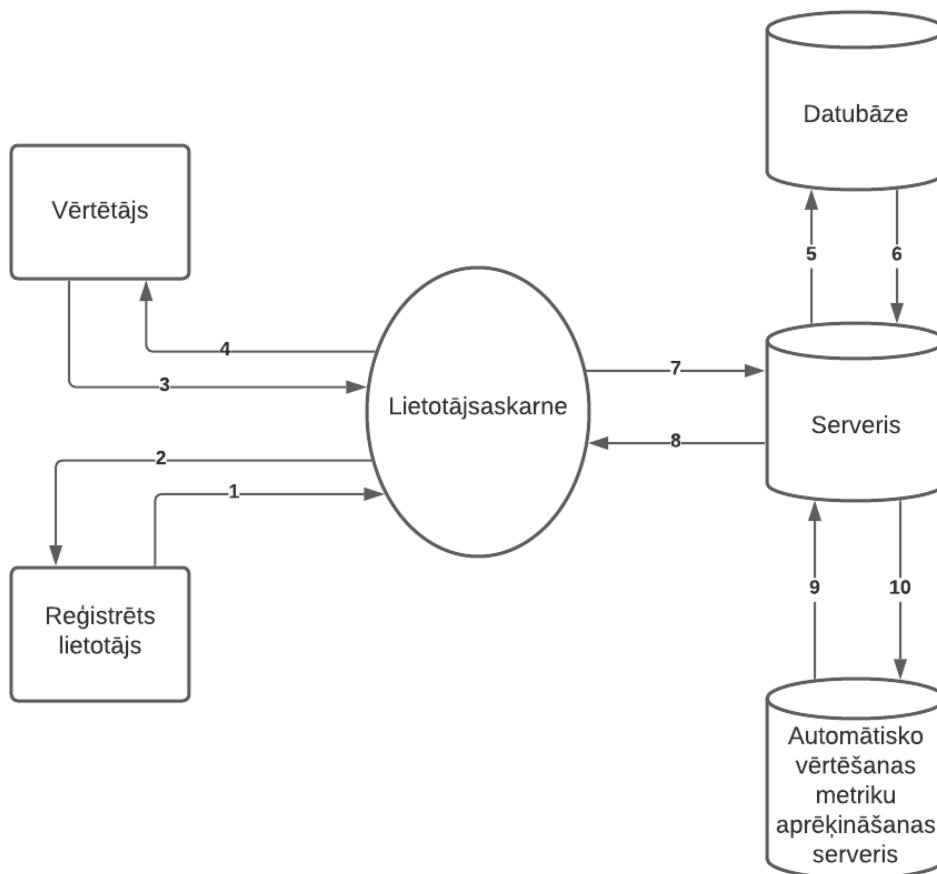
3.1.5. Pārskats

Programmatūras prasību specifikācijas sadaļas, ietver ievadu, vispārējo aprakstu, funkcionālās un nefunkcionālās prasības.

3.2. Vispārējs apraksts

3.2.1. Produkta perspektīva

Vērtēšanas platformas piekļuve notiek caur tīmekļa lietotni. Platforma ir paredzēta reģistrētiem lietotājiem, kas atbild par vērtēšanas uzdevumiem, un neregistrētiem lietotājiem (vērtētājiem), kas atbild par teikumu pareizu novērtēšanu. Vērtētāja piekļuve sistēmai ir limitēta un viņš sistēmai var piekļūt tikai caur saiti, kas viņam ir iedota no kāda reģistrēta lietotāja.



3.1. attēls. 0. līmeņa datu plūsmas diagramma

1. Lietotājs ievada jaunu informāciju
2. Saskaņā rāda lietotājam aktuālāko informāciju par sistēmu.
3. Vērtētājs novērtē doto teikumu attiecīgajā skalā
4. Saskaņā atgriež lietotājam avotvalodas teikumu ar tā piederīgo, iztulkoto teikumu
5. Serveris nodod informāciju par jaunu ierakstu izveidi.
6. Datubāze atgriež servera pieprasītos datus.
7. Saskaņā pieprasa serverim aktuālāko informāciju par sistēmu.
8. Serveris atgriež saskaņas pieprasīto informāciju un datus.
9. AVMAS atgriež izrēķinātas metrikas no servera padotajiem teikumiem.

3.2.2. Produkta funkcijas

Produkta funkcijas ir sadalītas pa lietotāju grupām:

- Reģistrēts lietotājs spēj:
 - Pievienot paralēlo korpusu sistēmai,
 - Izveidot, pārvaldīt un dzēst vērtēšanas uzdevumu, kā arī izslēgt iespēju novērtēt teikumus vērtēšanas uzdevumā,
 - Pārreķināt automātiskās vērtēšanas metrikas vērtības,
 - Apskatīties statistiku pa vērtēšanas uzdevumu,
 - Vidējos vērtējumus,
 - Automātiskās novērtēšanas metrikas,
 - Salīdzināt cilvēkvērtējumu sadalījumu grafikā,
 - Automātisko metriku-cilvēkvērtējumu korelāciju.
 - Lejuplādēt vērtēšanas uzdevuma vērtējumus un statistiku.
- Neregistrēts lietotājs spēj:
 - Novērtēt teikumus vērtēšanas uzdevuma dotajā skalā, izmantojot viņam iedotu unikālu saiti.

3.2.3. Lietotāja raksturiezīmes

Lai izmantotu sistēmu, lietotājam jābūt pamatzināšanām izmantot datoru un pārlūkprogrammu. Reģistrētam lietotājam jāpārziņ paralēlo korpusu struktūra.

3.2.4. Vispārējie ierobežojumi

Sistēmu izmantot un tai piekļūt būs iespējams reģistrēti lietotāji, kā arī neregistrēti lietotāji, t. i., vērtētāji. Vērtētājiem būs pieeja sistēmai tikai no personalizētām hipersaitēm (apakšuzdevumiem), ko viņi iegūs no kāda reģistrēta lietotāja. Katram apakšuzdevumam ir piesaistīts viens un tikai viens vērtētājs.

Vērtēšanas uzdevumam vajadzētu būt piesaistītiem vienādam vai lielākam skaitam vērtētāju kā mašīntulkoto sistēmu skaitam, lai abas sistēmas varētu vienlīdzīgi novērtēt.

AVMAS spēj strādāt tikai uz Linux operētājsistēmas, jo automātiskās novērtēšanas metrikas COMET bibliotēka atbalsta tikai Linux operētājsistēmu [5].

3.2.5. Pieņēmumi un atkarības

Lai darbinātu sistēmu ar aktuāliem datiem, tai jābūt pieejai datu apstrādes serverim, kā arī jābūt savienojumam ar autentifikācijas serveri, kas atjauno autentifikācijas marķieri.

Sistēmai jādarbojas četrās populārākajās tīmekļa pārlūkprogrammās: *Google Chrome*, *Apple Safari*, *Mozilla Firefox*, un *Microsoft Edge*.

Izstrādātā programmatūra ir atkarīga sistēma; tai ir saistība ar citām sistēmām, kas ir atbildīgas par lietotāju autentifikāciju, izmantojot JWT kā autentifikācijas pārbaudi, un bez tās reģistrēts lietotājs nespēs piekļūt klāt sistēmai.

3.3. Funkcionālās prasības

3.3.1. Ievads

Sistēmā “Tulkošanas kvalitātes vērtēšanas platforma” tiks nodrošināts darbs ar diviem lietotājiem:

- Reģistrēts lietotājs
- Neregistrēts lietotājs (vērtētājs)

Sistēmai ir trīs moduļi. Paralēlo korpusu modulis atbild par paralēlā korpusa, tā korpusa daļu un tā teikumu pievienošanu un pārvaldīšanu, kā arī sazināšanos ar automātiskās metrikas aprēķināšanas serveri, lai aprēķinātu metrikas katram paralēlajam korpusam. Statistikas modulis atbild par vērtēšanas uzdevuma statistikas aprēķināšanu. Lai statistika būtu aktuāla, tā netiek glabāta datubāzē, bet gan izrēķināta uz katru pieprasījumu. Vērtēšanas uzdevumu modulis atbild par vērtēšanas uzdevumu un apakšuzdevumu izveidi, pārvaldību un teikumu novērtēšanu.

Katrai reģistrēta lietotāja pieejamai funkcijai ir pieeja lietotāja JWT, kas nosaka lietotāja vārdu un identifikatoru, tāpēc tas pie funkciju ievaddatiem netiek atzīmēts.

3.3.2. Lietotāju stāsti

3.3.2.1. #53983 – Paralēlā korpusa izveide un pārvaldība

Stāsts
Es kā reģistrēts lietotājs Vēlos pievienot paralēlo korpusu sistēmai, Lai to varētu novērtēt cilvēkvērtētāji un automātiskās metrikas sistēmu salīdzināšanai.
Akceptējošie kritēriji
<ul style="list-style-type: none">• Iespēja pievienot paralēlo korpusu, norādot tā nosaukumu, avotvalodas un mērķvalodas kodus, korpusa daļu nosaukumus, tipu un failu ar teikumiem.• Iespējai atsevišķi pievienot korpusa daļu, norādot nosaukumu, tipu “Target Reference” vai “Hypothesis” un failu ar teikumiem.• Iespēja apskatīt paralēlo korpusu• Mērķvalodai un avotvalodas izvēlnē jābūt ISO-639-1 valodu kodi [6].

3.3.2.2. #54015 – Teikumu pievienošana ar teksta failu palīdzību

Stāsts
Es kā reģistrēts lietotājs Vēlos pievienot korpusa daļai failu ar teikumiem, Lai varētu pievienot vairākus teikumus vienlaicīgi.
Akceptējošie kritēriji
<ul style="list-style-type: none">• Ielādēto failu jāuztver kā teksta failu, kur katrs teikums ir savā rindā.• Failu ielādes poga• Faili var būt ar dažādiem formātiem, piemēram, .txt, .en, .lt utt.

3.3.2.3. #54068 – Lietotāja autentificēšanās ar Keycloak

Stāsts
Es kā reģistrēts lietotājs Vēlos autentificēties sistēmai, Lai norādītu, kam pieder pievienotais paralēlais korpuss.
Akceptējošie kritēriji
<ul style="list-style-type: none">• Autentificēšanās jānodrošina ar SIA “Tilde” autentificēšanās serveri Keycloak.• Visas sistēmas funkcijas ir pieejamas tikai autentificētam lietotājam.

3.3.2.4. #54076 – Vērtēšanas uzdevumu izveide un pārvaldība

Stāsts
Es kā reģistrēts lietotājs Vēlos izveidot jaunu vērtēšanas uzdevumu, Lai norādīt vērtēšanas kritērijus paralēlā korpusa novērtēšanai.
Akceptējošie kritēriji
<ul style="list-style-type: none">• Iespēja pievienot vērtēšanas uzdevumu, norādot nosaukumu, paralēlo korpusu, vērtēšanas tipu starp “Direct Assessment” un “Likert”, skalu, kas nav obligāta (“Direct Assessment” skalai noklusējumā ir vērtība 101 un “Likert” - 7), aprakstu, kas tiks parādīts vērtētājam, un norādīt cik vērtētājus izveidot automātiski.• Iespējai atsevišķi pievienot vērtētājus, norādot tā nosaukumu.

- Apskatīt vērtēšanas uzdevumu un vērtētājus.
- Iespēja apskatīt vērtētāja vērtējumus un informāciju par katru vērtētāju – hipotēzes teikums, avotvalodas teikums, vērtējuma vērtība un vai teikums ir kontroles teikums un kāds.
- Vērtēšanas uzdevums sastāv no apakšuzdevumiem, kuriem katram ir unikāls identifikators. Katram apakšuzdevumam pieder viens vērtētājs un vērtētājs varēs tikt pie apakšuzdevuma ar saiti, ko viņam iedos reģistrēts lietotājs.

3.3.2.5. #54099 – Lietotāja piesaistīšana pievienotajiem datiem

Stāsts
<p>Es kā reģistrēts lietotājs Vēlos, pievienojot datus, Lai tie būtu pieejami tikai man - lietotājam, kas tos pievienoja.</p>
Akceptējošie kritēriji
<ul style="list-style-type: none"> • Izveidotie paralēlie korpusi un vērtēšanas uzdevumi būtu jāredz tikai lietotājam, kas tos pievienoja.

3.3.2.6. #54200 – Teikumu novērtēšana uzdevumā

Stāsts
<p>Es kā vērtētājs Vēlos novērtēt teikumus vērtēšanas uzdevumā, Lai izvērtēt, kura sistēma nodrošina labāku tulkošanas kvalitāti.</p>
Akceptējošie kritēriji
<ul style="list-style-type: none"> • Vērtētājam ir jāredz avotvalodas teikums un hipotēzes teikums. • Iespēja novērtēt teikumu vērtēšanas uzdevuma noteiktajā skalā. • Teikumu izvēlei jānotiek pēc noteiktas struktūras (skat. #54220 – Kontroles teikumu kontrole). • Vērtētājam ir jāredz vērtēšanas uzdevuma apraksts.

3.3.2.7. #54220 – Kontroles teikumu kontrole

Stāsts
<p>Es kā reģistrēts lietotājs</p> <p>Vēlos vērtētājiem dod atkārtotus teikumus un sabojātus teikumus,</p> <p>Lai noteiktu, vai vērtētāji godīgi vērtē un lai pārbaudītu vērtētāju atjautību.</p>
Akceptējošie kritēriji
<ul style="list-style-type: none">• Pirmie 25 teikumi, kas tiek doti vērtētājiem, tiek doti vienādi ar pārējiem vērtētājiem.• Pēc 25 teikumiem tiek doti teikumi, kuru neviens vērtētājs nav novērtējis, bet tā teikuma vieta korpusā ir vienāda ar kāda cita vērtētāja novērtētu teikuma vietu korpusā.• Pēc 40 teikumiem:<ul style="list-style-type: none">○ vērtētājiem ar varbūtību 0,1 tiek dots teikums, ko vērtētājs jau ir novērtējis.○ vērtētājiem ar varbūtību 0,1 tiek dots teikums, ko ir novērtējis kāds cits vērtētājs.○ vērtētājiem ar varbūtību 0,1 tiek dots teikums, kas ir sabojāts [7]. Sabojātam teikumam tiek izņemti vārdi atkarībā no vārdu skaitā teikumā:<ul style="list-style-type: none">▪ 2-3 vārdu teikumā tiek izņemts 1 vārds.▪ 4-5 vārdu teikumā tiek izņemti 2 vārdi.▪ 6-8 vārdu teikumā tiek izņemti 3 vārdi.▪ 9-15 vārdu teikumā tiek izņemti 4 vārdi.▪ 16-20 vārdu teikumā tiek izņemti 5 vārdi.▪ Teikumiem ar vārdu daudzumu $n > 20$ tiek izņemti $\lfloor n/5 \rfloor$ vārdi.○ Vērtētājam ar 0,7 tiek dots teikums, kuru neviens vērtētājs nav novērtējis, bet tā teikuma vieta korpusā ir vienāda ar kāda cita vērtētāja novērtētu teikuma vietu korpusā.

3.3.2.8. #54225 – Standarta statistikas rādītāji

Stāsts
<p>Es kā reģistrēts lietotājs</p> <p>Vēlos apskatīt standarta statistikas rādītājus,</p> <p>Lai varētu veikt vienkāršu sistēmu analīzi.</p>
Akceptējošie kritēriji
<ul style="list-style-type: none">• Vidējā vērtība vērtējumiem

- Standartnovirze
- Pārliecības intervāls
- Vērtējuma standartizētā vērtība
- Z vērtība – vērtība, kas apzīmē vidējo vērtību standartizētajiem vērtējumiem

3.3.2.9. #54229 – Statistiskā nozīmība

Stāsts
<p>Es kā reģistrēts lietotājs</p> <p>Vēlos apskatīt statistisko nozīmības matricu,</p> <p>Lai noteiktu, cik uzticami ir vērtējumi.</p>
Akceptējošie kritēriji
<ul style="list-style-type: none"> • P-vērtību matrica pa vērtētājiem, kas nosaka viņu godīgumu. • P-vērtības >0.05 iekrāsoas zālā krāsā, jo tad vērtējumi var tikt uzskatīti par uzticamiem. • P-vērtības <0.05 iekrāsotas sarkanā krāsā jo tad vērtējumi tiek uzskatīti par neuzticamiem. • Izvēlnē var izvēlēties, kuras sistēmas parādās tabulā. • Rēķinot p-vērtību tiek izmantots bootstrap resampling [8].

3.3.2.10. #54232 – Vērtētāju vienprātība

Stāsts
<p>Es kā reģistrēts lietotājs</p> <p>Vēlos noteikt vērtētāju vienprātību,</p> <p>Lai uzzinātu, vai vērtētāji vērtē līdzīgi.</p>
Akceptējošie kritēriji
<ul style="list-style-type: none"> • Direct Assessment skalai ir jāizveido Spearman matrica, kas nosaka vērtētāju vienprātību. • Izvēlnē var izvēlēties, kuri vērtētāji parādās tabulā. • Likert skalai ir jāizveido Fleiss kappa.

3.3.2.11. #54252 – Vērtējumu attēlošana diagrammā

Stāsts
Es kā reģistrēts lietotājs Vēlos vērtējumus kā diagrammas, Lai varētu vienkārši apskatīties salīdzinājumu starp sistēmu vērtējumiem.
Akceptējošie kritēriji
<ul style="list-style-type: none">• Divas stabiņu diagrammas: standardizētiem vērtējumiem un normāliem vērtējumiem.• Vērtējumi ir jāsadala pa sistēmām.• Uzbraucot ar peles kursoru uz kādu no diagrammas stabiņiem, saskarnei ir jāattēlo papildus informācija par attiecīgo vērtējumu, t.i., vērtējums, vērtējumu skaits un procentuālā daļa no visiem vērtējumiem.• Jābūt iespējai lejuplādēt stabiņu diagrammas kā bildes.

3.3.2.12. #54280 – Automātisko metriku aprēķināšana katram teikumam

Stāsts
Es kā reģistrēts lietotājs Vēlos redzēt automātiskās novērtēšanas metrikas, Lai varētu veikt korelācijas ar cilvēkvērtējumiem.
Akceptējošie kritēriji
<ul style="list-style-type: none">• Katram teikumam ir jāaprēķina BLEU, CHRF, TER.• Katram teikumam ir jāaprēķina COMET pret katru references korpusa daļas teikumu.

3.3.2.13. #54325 – Vērtējumu un statistikas eksportēšana CSV faila formātā

Stāsts
Es kā reģistrēts lietotājs Vēlos lejuplādēt vērtējumus un statistiku CSV faila formātā, Lai tos varētu lietot ārpus sistēmas ietvariem.
Akceptējošie kritēriji
<ul style="list-style-type: none">• Poga, kas ļauj lejuplādēt vērtējumus kā CSV failu.• Poga, kas ļauj lejuplādēt vērtēšanas uzdevuma statistiku kā CSV failu

3.3.2.14. #54343 – Korelācija starp cilvēkvērtējumiem un automātiskajām novērtēšanas metrikām

Stāsts
Es kā reģistrēts lietotājs Vēlos redzēt korelāciju cilvēkvērtējumiem un automātiskajām novērtēšanas metrikām, Lai uzzinātu, vai cilvēkvērtētāji vērtē līdzīgi automātiskajām novērtēšanas metrikām.
Akceptējošie kritēriji
<ul style="list-style-type: none">• Jāizveido Spearman korelācijas matrica [9] ar visiem iespējamiem vērtējumiem vērtēšanas uzdevumos pret automātiskajām novērtēšanas metrikām..• Jāizveido Kendall Tau korelācijas matrica [10] ar visiem iespējamiem vērtējumiem vērtēšanas uzdevumos pret automātiskajām novērtēšanas metrikām.• Jābūt iespējai izvēlēties vērtēšanas uzdevumus, kuri tiks iekļauti vērtējumos.• Matrica jāizveido gan normāliem, gan standartizētiem vērtējumiem.

3.4. Nefunkcionālās prasības

Veiktspējas prasības:

- Ātrdarbība ir atkarīga no servera veiktspējas.
- Sistēmai ir jānodrošina vismaz 500 vienlaicīgi lietotāji.
- Ierakstu skaits datubāzē var pārsniegt vairākus simtu tūkstošus.
- AVMAS ātrdarbība ir atkarīga no korpusa daļu daudzuma paralēlajā korpusā, kā arī no teikumu daudzuma katrā korpusa daļā.
- Pēc katra novērtējuma sistēmai jāatgriež jauns vērtējams teikumus laikā, kas mazāks par 1 sekundi.

Drošība:

- Katram vērtētājam tiek nodrošināta anonimitāte, tam piešķirot hipersaiti uz vērtēšanas apakšuzdevumu.
- Tiek saglabāta datu integritāte.

4. PROGRAMMATŪRAS PROJEKTĒJUMA APRAKSTS

4.1. Ievads

4.1.1. Nolūks

PPA ir izveidots, lai detalizētu PPS aprakstīto moduļu un prasību realizāciju. Šis dokuments ir paredzēts moduļa izstrādātājiem un saskarņu projektētājiem.

4.1.2. Darbības sfēra

“Teikumu kvalitātes novērtēšanas platformas” mērķis ir ļaut cilvēkvērtētājiem efektīvi un ātri novērtēt mašīntulkotus teikumus, lai varētu noteikt mašīntulku sistēmu kvalitāti. Vērtēšanu veiks anonīmi cilvēkvērtētāji, kuriem reģistrēti lietotāji iedots saiti uz vērtēšanas apakšuzdevumu, un vērtēšanas uzdevumus pievienos un pārvaldīs reģistrēti lietotāji. Izmantojot šos vērtējumus, reģistrēti lietotāji varēs noteikt, kura sistēma ir labāka, tādējādi atvieglojot sistēmu administratoru darbu un atļaujot uzlabot mašīntulku sistēmas.

4.1.3. Definīcijas

Skatīt nodaļu Apzīmējumi un definīcijas.

4.1.4. Saistības ar citiem dokumentiem

Dokumenta noformēšanā izmantots Latvijas standarta LVS 72:1996 PPS “Ieteicamā prakse programmatūras projektējuma aprakstīšanai” 12[11].

4.2. Dekompozīcijas apraksts

4.2.1. Paralēlo korpusu modulis

Modulis ir paredzēts, lai pārvaldītu paralēlos korpusu, ko lietotājs ir pievienojis. Paralēlais korpus ir atbildīgs par visu korpusa daļu glabāšanu un korpusa daļas ir atbildīgas par teikumu glabāšanu. Modulis paredz, ka sistēmai pievienotais paralēlais korpus bez vērtēšanas uzdevuma izveides, tāpēc paralēlā korpusa pārvaldīšana notiek caur izveidotu vērtēšanas uzdevumu.

4.2.2. Statistikas modulis

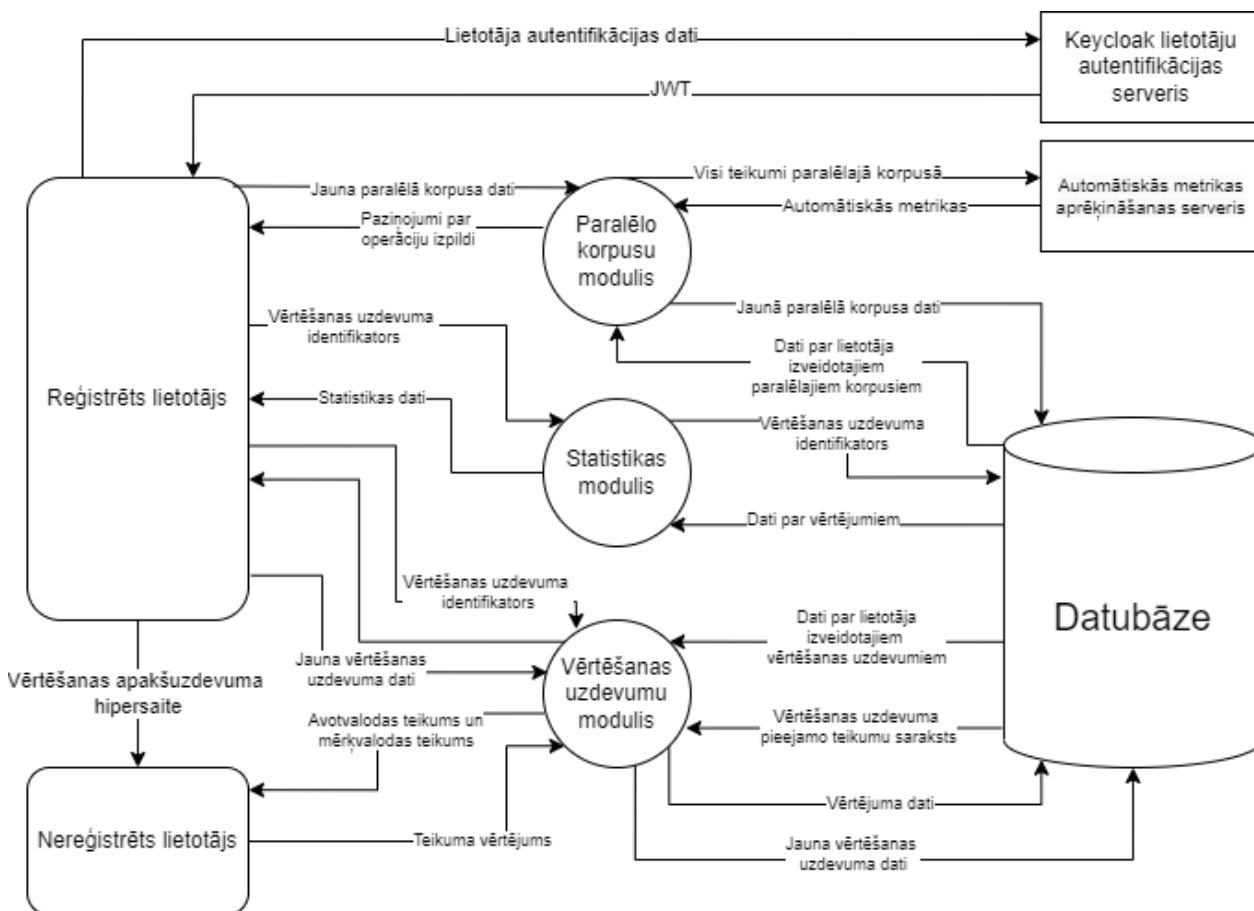
Modulis ir paredzēts, lai pārvaldītu visu statistiku par vērtēšanas uzdevumiem. Modulis spēj izrēķināt un pārrēķināt automātiskās vērtēšanas metrikas, kā arī atgriezt visas nepieciešamās metrikas, kas ir redzamas vērtēšanas uzdevuma statistikas skatā.

4.2.3. Vērtēšanas uzdevumu modulis

Modulis ir paredzēts, lai pārvaldītu visus vērtēšanas uzdevumus, ko lietotājs ir pievienojis. Vērtēšanas uzdevumam ir piesaistīts viens paralēlais korpuss un vairāki vērtētāji, kuru skaits parasti ir vienāds vai lielāks par hipotēžu korpusa daļu skaitu, kas atrodas piesaistītajā paralēlajā korpūsā. Reģistrēta lietotāja atbildība ir nosūtīt vērtētājiem vērtēšanas apakšuzdevuma hipersaites, kur vienu vērtēšanas apakšuzdevumu vērtē viens un tikai viens vērtētājs.

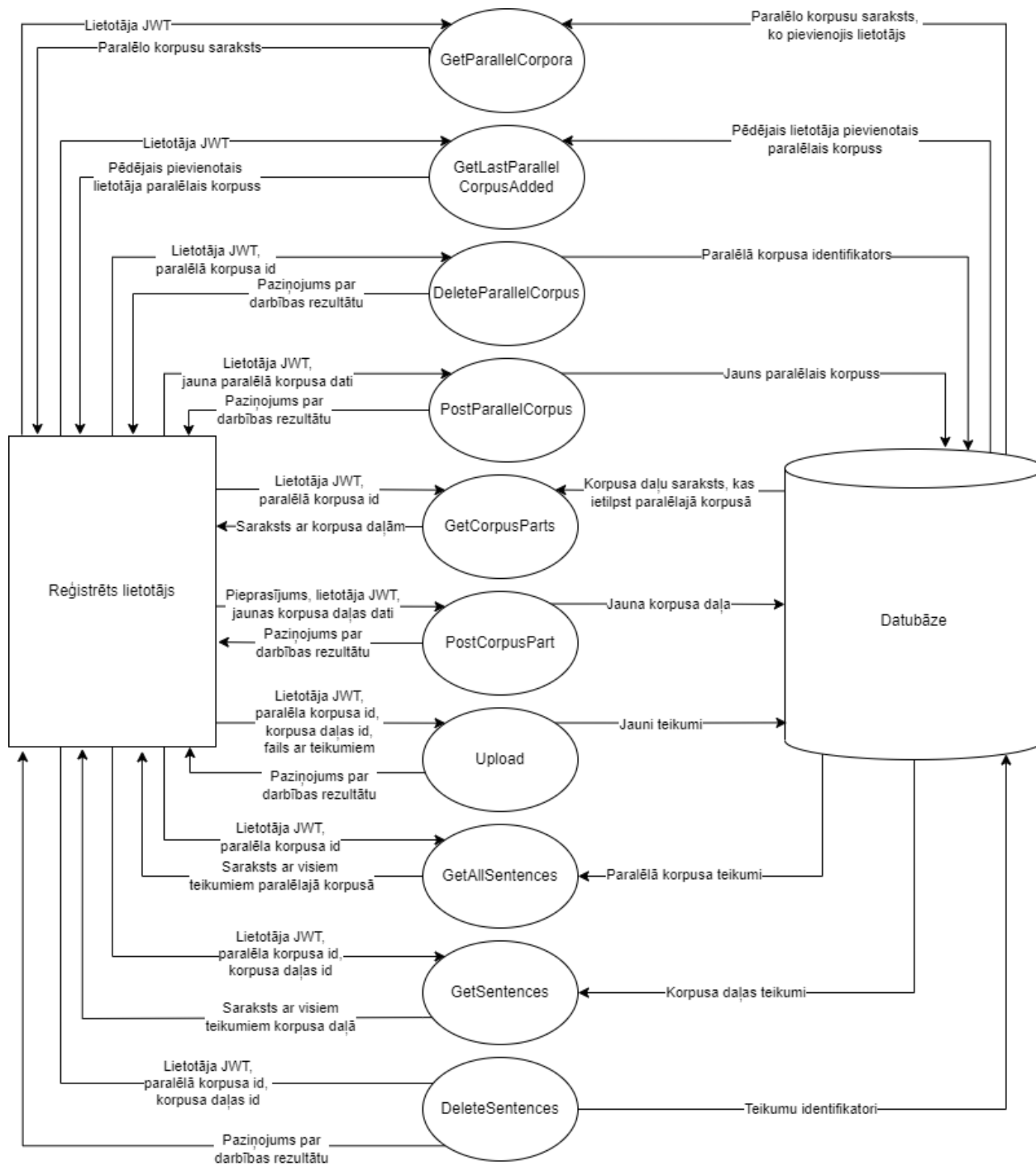
4.3. Detalizēts projektējums

4.3.1. Funkciju sadalījums moduļos



4.1. attēls. 1. līmeņa datu plūsmas diagramma

4.3.2. Paralēlo korpusu modulis



4.2. attēls. 2. līmeņa datu plūsmas diagramma

PPS prasība: 3.3.2.1
Funkcija: GetParallelCorpora
Mērķis: Jāatgriež visi paralēlie korpusi, kurus lietotājs varētu izvēlēties, veidojot jaunu paralēlo korpusu.
Ievaddati: Nav.
Apstrāde: Serveris no datubāzes atgriež visus paralēlos korpusus, kuriem nav piesaistīti vērtēšanas uzdevumi un kurus ir pievienojis autentificētais lietotājs.
Izvaddati: JSON objekts ar paralēlo korpusu sarakstu.
Kļūdu paziņojumi: Ja lietotājs nav autentificējies sistēmai, tad viņš saņems autentifikācijas kļūdu 401.

PPS prasības: 3.3.2.1.
Funkcija: GetLastParallelCorpusAdded
Mērķis: Lietotājam, pievienojot jaunu paralēlo korpusu, to automātiski jāattēlo vērtēšanas uzdevuma izveidošanas lapā kā izvēlēto paralēlo korpusu.
Ievaddati: Nav.
Apstrāde: Serveris no datubāzes atgriež pēdējo paralēlo korpusu, kuru pievienojis autentificētais lietotājs.
Izvaddati: JSON objekts ar paralēlo korpusu.
Kļūdu paziņojumi: Ja lietotājs nav autentificējies sistēmai, tad viņš saņems autentifikācijas kļūdu 401.

PPS prasības: 3.3.2.1.
Funkcija: DeleteParallelCorpus
Mērķis: Lietotājam jābūt iespējai izdzēst paralēlo korpusu, ja ir tāda nepieciešamība.
Ievaddati: Obligātie: <ul style="list-style-type: none"> ParallelCorpusId – paralēlā korpusa identifikators
Apstrāde: Serveris nosūta datubāzei pieprasījumu dzēst paralēlo korpusu ar norādīto idenfikatoru.
Izvaddati: Lietotājam tiek atgriezts paziņojums par darbības rezultātu
Kļūdu paziņojumi: Ja lietotājs nav autentificējies sistēmai, tad viņš saņems autentifikācijas kļūdu 401. Ja paralēlais korpus netika atrasts, tad lietotājs saņems atrašanas problēmas kļūdu 404.

PPS prasības: 3.3.2.1.
Funkcija: PostParallelCorpus
Mērķis: Paralēlo korpusu ir jā saglabā datubāzē.
Ievaddati: Obligātie: <ul style="list-style-type: none"> Name – paralēlā korpusa nosaukums, SourceLanguageCode – paralēlā korpusa avotvalodas kods. TargetLanguageCode – paralēlā korpusa mērķvalodas kods.
Apstrāde:

Serveris, saņemot pieprasījumu, pievieno paralēlajam korpusam autentificētā lietotāja identifikatoru. Ja datubāzē nav šis lietotājs, tad serveris nosūta datubāzei pieprasījumu pievienot šo lietotāju. Serveris nosūta datubāzei pieprasījumu pievienot jaunu paralēlo korpusu.

Izvaddati:

Lietotājam tiek atgriezts paziņojums par darbības rezultātu

Kļūdu paziņojumi:

Ja lietotājs nav autentificējies sistēmai, tad viņš saņems autentifikācijas kļūdu 401.

4.5. tabula

PPS prasības: 3.3.2.1.

Funkcija:

GetCorpusParts

Mērķis:

Lietotājam jābūt iespējai redzēt visas korpusa daļas paralēlajā korpusā.

Ievaddati:

Obligātie:

- ParallelCorpusId – paralēlā korpusa identifikators.

Apstrāde:

Serveris no datubāzes atgriež visas korpusa daļas paralēlajā korpusā.

Izvaddati:

JSON objekts ar visām korpusa daļām.

Kļūdu paziņojumi:

Ja lietotājs nav autentificējies sistēmai vai viņš vēlas piekļūt paralēlajam korpusam, ko viņš nav izveidojis, tad viņš saņems autentifikācijas kļūdu 401. Ja neeksistē tāds paralēlais korpus ar tādu identifikatoru, tad lietotājs saņem neatrašanas kļūdu 404.

PPS prasības: 3.3.2.1.
Funkcija: PostCorpusPart
Mērķis: Lietotājam jābūt iespējai pievienot korpusa daļu paralēlajam korpusam.
Ievaddati: Obligātie: <ul style="list-style-type: none"> • ParallelCorpusId – paralēlā korpusa identifikators. • Name – Sistēmas nosaukums • Type – korpusa daļas tips.
Apstrāde: Serveris nosūta datubāzei pieprasījumu pievienot jaunu korpusa daļu.
Izvaddati: Lietotājam tiek atgriezts paziņojums par darbības rezultātu
Kļūdu paziņojumi: Ja lietotājs nav autentificējies sistēmai, tad viņš saņems autentifikācijas kļūdu 401.

PPS prasības: 3.3.2.2.
Funkcija: Upload
Mērķis: Lietotājam jābūt iespējai, pievienojot jaunu korpusa daļu, pievienot arī failu ar teikumiem, kas atradīsies korpusa daļā
Ievaddati: Obligātie: <ul style="list-style-type: none"> • ParallelCorpusId – paralēlā korpusa identifikators. • CorpusPartId – korpusa daļas identifikators. • File – fails ar teikumiem
Apstrāde:

Serveris, saņemot pieprasījumu, failu saglabā pagaidu mapē un to atver, un tam iterē cauri pa rindiņai, kur katra rindiņa ir teikums. Serveris katru teikumu individuāli sūta uz datubāzi kā pieprasījumu pievienot jaunu teikumu datubāzei konkrētajai korpusa daļai. Kad visi teikumi ir nosūtīti, serveris saglabāto failu izdzēš no pagaidu mapes.

Izvaddati:

Lietotājam tiek atgriezts paziņojums par darbības rezultātu.

Kļūdu paziņojumi:

Ja lietotājs nav autentificējies sistēmai, tad viņš saņems autentifikācijas kļūdu 401. Ja neeksistē tāds paralēlais korpus ar tādu identifikatoru vai neeksistē tāda korpusa daļa ar tādu identifikatoru, tad lietotājs saņem neatrašanas kļūdu 404.

4.8. tabula

PPS prasības: 3.3.2.1.

Funkcija:

GetAllSentences

Mērķis:

Lietotājam jābūt iespējai apskatīt visus teikumus paralēlajā korpusā un lai pārliecinātos, ka visi teikumi ir pievienoti pareizi, t.i., katrs teikums ir piesaistīts pie pareizā avotvalodas teikuma.

Ievaddati:

Obligātie:

- ParallelCorpusId – paralēlā korpusa identifikators.

Apstrāde:

Serveris no datubāzes atgriež visus teikumus paralēlajā korpusā un sakārto tos tā, lai mērķvalodas puses teikumi būtu piekārtoti pareizajam avotvalodas teikumam pēc to vietas korpusā (NumberInCorpus).

Izvaddati:

JSON objekts ar visu teikumu sarakstu.

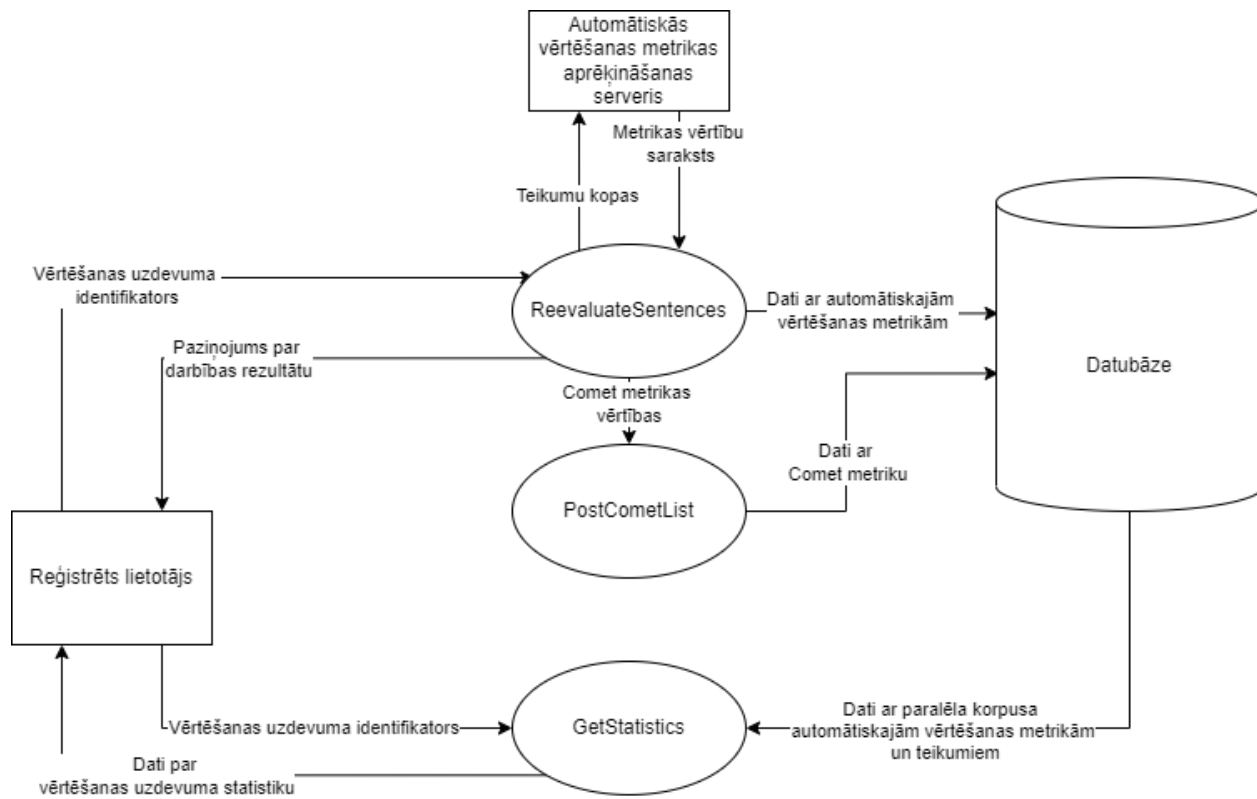
Kļūdu paziņojumi:

Ja lietotājs nav autentificējies sistēmai vai viņš vēlas piekļūt paralēlā korpusa datiem, ko viņš nav izveidojis, tad viņš saņems autentifikācijas kļūdu 401. Ja neeksistē tāds paralēlais korpus ar tādu identifikatoru, tad lietotājs saņem neatrašanas kļūdu 404.

<p>PPS prasības: 3.3.2.1.</p> <p>Funkcija: GetSentences</p>
<p>Mērķis: Lietotājam jābūt iespējai apskatīt visus teikumus korpusa daļā.</p>
<p>Ievaddati: Obligātie:</p> <ul style="list-style-type: none"> • ParallelCorpusId – paralēlā korpusa identifikators. • CorpusPartId – korpusa daļas identifikators.
<p>Apstrāde: Serveris no datubāzes atgriež visus teikumus korpusa daļā.</p>
<p>Izvaddati: JSON objekts ar teikumu sarakstu.</p>
<p>Kļūdu paziņojumi: Ja lietotājs nav autentificējies sistēmai vai viņš vēlas piekļūt paralēlā korpusa datiem, ko viņš nav izveidojis, tad viņš saņems autentifikācijas kļūdu 401. Ja neeksistē tāds paralēlais korpus ar tādu identifikatoru vai neeksistē tāda korpusa daļa ar tādu identifikatoru, tad lietotājs saņem neatrašanas kļūdu 404.</p>

PPS prasības: 3.3.2.1.
<p>Funkcija: DeleteSentences</p>
<p>Mērķis: Lietotājam pievienojot jaunu failu ar teikumiem, ir jāizdzēš iepriekšējie teikumi, kas atradās korpusa daļā.</p>
<p>Ievaddati: Obligātie:</p> <ul style="list-style-type: none"> • CorpusPartId – korpusa daļas identifikators.
<p>Apstrāde: Serveris nosūta pieprasījumu izdzēst no datubāzes visus teikumus, kas atrodas esošajā korpusa daļā.</p>
<p>Izvaddati: Lietotājam tiek atgriezts paziņojums par darbības rezultātu.</p>
<p>Kļūdu paziņojumi: Ja lietotājs nav autentificējies sistēmai, tad viņš saņems autentifikācijas kļūdu 401.</p>

4.3.3. Statistikas modulis



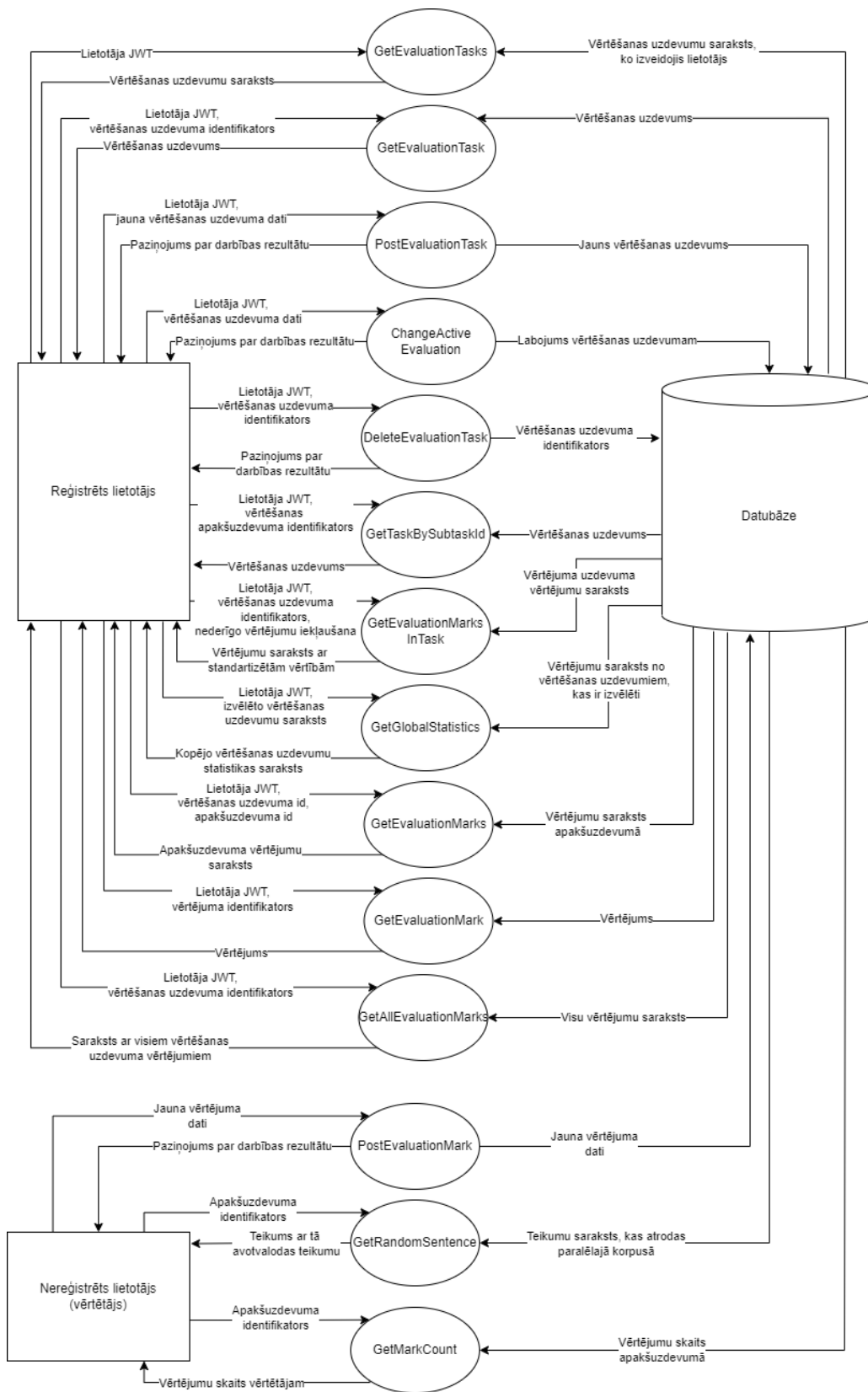
4.3. attēls. 2. līmeņa datu plūsmas diagramma

<p>PPS prasības: 3.3.2.8.</p> <p>Funkcija: GetStatistics</p>
<p>Mērķis: Lietotājam ir nepieciešams apskatīt statistiku par vērtēšanas uzdevumu.</p>
<p>Ievaddati:</p> <p>Obligātie:</p> <ul style="list-style-type: none"> • taskId – vērtēšanas uzdevuma identifikators • includeInvalidMarks – boolean vērtība, kura nosaka nederīgo vērtējumu iekļaušanu statistiskās aprēķinos
<p>Apstrāde:</p> <p>Izmantojot ievades datus, serveris no datubāzes atgriež visus vērtējumus vērtēšanas uzdevumā un izfiltrē tos atkārtībā no includeInvalidMarks vērtības. No šīs izfiltrētās kopas tiek aprēķināta visa nepieciešamā statistika, kas ir:</p> <ul style="list-style-type: none"> • BLEU • CHRF • TER • COMET • Vidējā vērtība • Standardnovirze • Z vērtība • Fleiss kappa • Spearman ranks • Pārliecības intervāls • P vērtību matrica
<p>Izvaddati:</p> <p>JSON objekts ar augstāk nosauktām statistikas vērtībām.</p>
<p>Kļūdu paziņojumi:</p> <p>Ja lietotājs nav autentificējies sistēmai vai viņš vēlas iegūt statistiku no vērtēšanas uzdevuma, ko viņš nav izveidojis, tad viņš saņems autentifikācijas kļūdu 401.</p>

<p>PPS prasības: 3.3.2.10.</p> <p>Funkcija: ReevaluateStatistics</p>
<p>Mērķis: Pievienojot jaunu paralēlo korpusu, tam ir nepieciešams aprēķināt automātiskās novērtēšanas metrikas.</p>
<p>Ievaddati: Obligātie:</p> <ul style="list-style-type: none"> • ParallelCorpusId – paralēla korpusa identifikators
<p>Apstrāde: No paralēlā korpusa pēc kārtas tiek ņemti hipotēzes teikumi un tam attiecīgais avotvalodas teikums un atsaucis teikumi. Tie tiek pārveidoti par JSON objektu un ar WebSocket palīdzību tiek nosūtīti AVMAS. Kad visi teikumi ir nosūtīti un aprēķināti, no iegūtajām metrikām tiek ņemta vidējā vērtība un nosūtīti datubāzei.</p>
<p>Izvaddati: Paziņojums par darbības rezultātu.</p>
<p>Kļūdu paziņojumi:</p> <ul style="list-style-type: none"> • Ja lietotājs nav autentificējies sistēmai, tad viņš saņems autentifikācijas kļūdu 401. • Ja serveris nespēj izveidot savienojumu ar AVMAS, tad lietotājs tiek informēts ar kļūdas paziņojumu. • Ja lietotājs cenšas iegūt automātiskās novērtēšanas metrikas paralēlajam korpusam, kuram nav pievienota atsaucis korpusa daļa, tad lietotājs tiek informēts ar kļūdas paziņojumu.

<p>PPS prasības: 3.3.2.10.</p> <p>Funkcija: PostCometList</p>
<p>Mērķis: COMET vērtības ir jāsavienā datubāzē.</p>
<p>Ievaddati:</p> <p>Obligātie:</p> <ul style="list-style-type: none"> • comet – masīvs ar COMET vērtībām • corpusPart – hipotēzes korpusa daļa • referenceAndSourceCorpusParts – masīvs ar avotvalodas un atsauces korpusa daļām.
<p>Apstrāde:</p> <p>COMET vērtība tiek aprēķināta pret katru atsauces korpusa daļu, tāpēc atšķirībā no BLEU, CHRf un TER tai var būt vairākas vērtības vienai hipotēzes korpusa daļai. Ievaddati tiek apstrādāti par “comets” objektu un nosūtīti datubāzei entītijas atjaunošanai, ja korpusa daļai jau bija izrēķināts COMET, vai jaunas entītijas izveidošanai.</p>
<p>Izvaddati:</p> <p>Paziņojums par darbības rezultātu</p>
<p>Kļūdu paziņojumi:</p> <p>Ja datubāze nespēj atjaunot vai izveidot entītiju, tā atgriež kļūdas paziņojumu ar kļūdu 503.</p>

4.3.4. Vērtēšanas uzdevumu modulis



4.4. attēls. 2. līmeņa datu plūsmas diagramma

PPS prasības: 3.3.2.4.
Funkcija: GetEvaluationTasks
Mērķis: Jāatgriež visi vērtēšanas uzdevumi, kurus lietotājs ir pievienojis.
Ievaddati: Nav.
Apstrāde: Serveris no datubāzes atgriež visus vērtēšanas uzdevumus, kurus ir pievienojis autentificētais lietotājs.
Izvaddati: JSON objekts ar vērtēšanas uzdevumu sarakstu.
Kļūdu paziņojumi: Ja lietotājs nav autentificējies sistēmai, tad viņš saņems autentifikācijas kļūdu 401.

PPS prasības: 3.3.2.4.
Funkcija: PostEvaluationTask
Mērķis: Vērtēšanas uzdevumu ir jā saglabā datubāzē.
Ievaddati: Obligātie: <ul style="list-style-type: none"> • Name – paralēlā korpusa nosaukums, • Description – vērtēšanas uzdevuma apraksts, kas tiks parādīts vērtētājam, piemēram, kāda veida uzdevums, pēc kādiem kritērijiem jāvērtē utt. • EvaluationType – vērtēšanas uzdevuma tips (Likert vai Direct Assessment) • Neobligātie:

<ul style="list-style-type: none"> Scale – vērtēšanas uzdevuma skala (noklusējumā Likert skala ir 0-6 un Direct Assessment skala ir 0-100).
<p>Apstrāde:</p> <p>Serveris, saņemot pieprasījumu, pievieno vērtēšanas uzdevumam autentificētā lietotāja identifikatoru un pieliek ActiveEvaluation vērtību kā True. Ja datubāzē nav šis lietotājs, tad serveris nosūta datubāzei pieprasījumu pievienot šo lietotāju. Serveris nosūta datubāzei pieprasījumu pievienot jaunu vērtēšanas uzdevumu.</p>
<p>Izvaddati:</p> <p>Lietotājam tiek atgriezts paziņojums par darbības rezultātu.</p>
<p>Kļūdu paziņojumi:</p> <p>Ja lietotājs nav autentificējies sistēmai, tad viņš saņems autentifikācijas kļūdu 401.</p>

4.16. tabula

<p>PPS prasības: 3.3.2.4.</p> <p>Funkcija:</p> <p>ChangeActiveEvaluation</p>
<p>Mērķis:</p> <p>Lietotājam jābūt iespējai izslēgt vai ieslēgt vērtēšanu vērtēšanas uzdevumam</p>
<p>Ievaddati:</p> <p>Obligātie:</p> <ul style="list-style-type: none"> task – vērtēšanas uzdevums.
<p>Apstrāde:</p> <p>Serveris nosūta datubāzei pieprasījumu nomainīt vērtēšanas uzdevuma vērtību ActiveEvaluation.</p>
<p>Izvaddati:</p> <p>Lietotājam tiek atgriezts paziņojums par darbības rezultātu.</p>
<p>Kļūdu paziņojumi:</p> <p>Ja lietotājs nav autentificējies sistēmai, tad viņš saņems autentifikācijas kļūdu 401.</p>

PPS prasības: 3.3.2.4.
Funkcija: DeleteEvaluationTask
Mērķis: Lietotājam jābūt iespējai izdzēst vērtēšanas uzdevumu, ja ir tāda nepieciešamība.
Ievaddati: Obligātie: <ul style="list-style-type: none"> • id – vērtēšanas uzdevuma identifikators.
Apstrāde: Serveris nosūta datubāzei pieprasījumu izdzēst vērtēšanas uzdevumu ar norādīto idenfikatoru.
Izvaddati: Lietotājam tiek atgriezts paziņojums par darbības rezultātu
Kļūdu paziņojumi: Ja lietotājs nav autentificējies sistēmai, tad viņš saņems autentifikācijas kļūdu 401. Ja vērtēšanas uzdevums netika atrasts, tad lietotājs saņems atrašanas problēmas kļūdu 404.

PPS prasības: 3.3.2.4.
Funkcija: GetTaskBySubtaskId
Mērķis: Kad vērtētājs atver vērtēšanas apakšuzdevumu, viņa klientam ir redzams tikai vērtēšanas apakšuzdevuma identifikators, bet ir jānoskaidro, vai šis vērtēšanas uzdevums ir aktīvs vai neaktīvs.
Ievaddati: Obligātie: <ul style="list-style-type: none"> • subtaskId – vērtēšanas apakšuzdevuma identifikators.
Apstrāde: Serveris nosūta datubāzei pieprasījumu atrast vērtēšanas uzdevumu, kam pieder norādītais vērtēšanas apakšuzdevums.

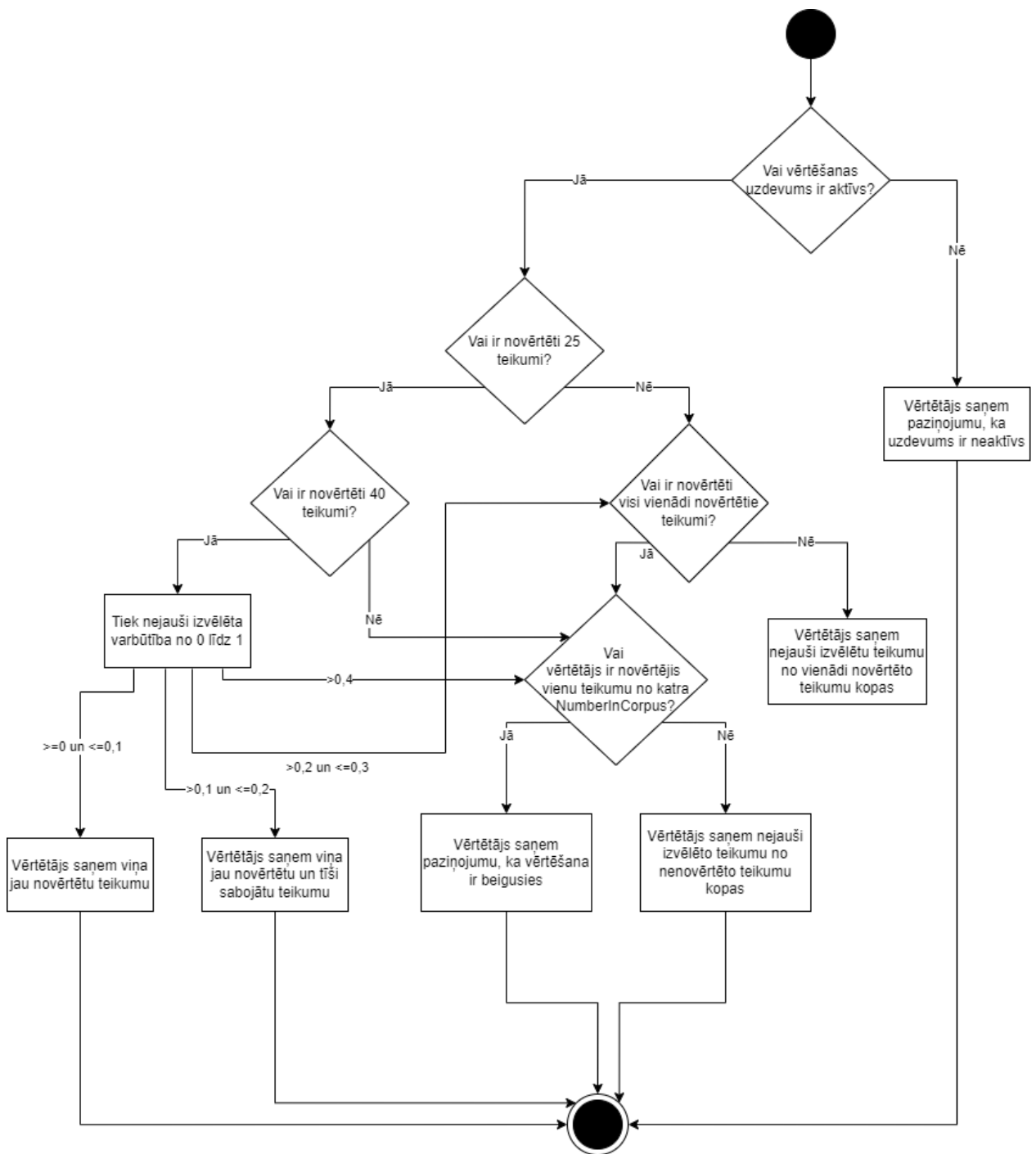
Izvaddati: Vērtēšanas uzdevums.
Kļūdu paziņojumi: Ja vērtēšanas uzdevums netika atrasts, tad lietotājs saņems atrašanas problēmas kļūdu 404.

4.19. tabula

PPS prasības: 3.3.2.9.
Funkcija: GetEvaluationMarksInTask
Mērķis: Lietotājam ir jāredz grafiki ar visiem vērtējumiem vērtēšanas uzdevumā.
Ievaddati: Obligātie: <ul style="list-style-type: none"> • taskId – vērtēšanas uzdevuma identifikators. • includeInvalidMarks – boolean vērtība, kas nosaka, vai iekļaut vērtējumus, kurus veikuši neuzticami vērtētāji.
Apstrāde: Serveris nosūta datubāzei pieprasījumu atgriezt visus vērtējumus vērtēšanas uzdevumā vai visus vērtējumus vērtēšanas uzdevumā, neiekļaujot neuzticamu vērtētāju vērtējumus.
Izvaddati: JSON objekts ar vērtējumu sarakstu.
Kļūdu paziņojumi: Ja lietotājs nav autentificējies sistēmai, tad viņš saņems autentifikācijas kļūdu 401.

Lai noteiktu, kurš vērtētājs ir neuzticams, tiek izmantoti atkārtoti un sabojāti teikumi. Labs vērtētājs vienādiem teikumiem neliks ievērojami atšķirīgus vērtējumus un sabojātiem teikumiem vērtējums būs ievērojami zemāks nekā tā nesabojātā teikuma vērtējums. Izmantojot šīs vērtības, sistēma izmanto Wilcoxon Rank sum test, kam nulles hipotēze ir, ka atkārotu teikumu vērtējumu starpība ir mazāka nekā sabojātā un nesabojāta teikumu vērtējuma starpība. Ja p vērtība ir lielāka par 0.05, tad vērtētājs tiek uzskatīts par neuzticamu vērtētāju.

<p>PPS prasības: 3.3.2.6., 3.3.2.7.</p> <p>Funkcija: GetRandomSentence</p>
<p>Mērķis: Vērtētājam, sākot vērtēt un novērtējot katru nākamo teikumu, ir jāparāda teikums, kuru viņam ir jānovērtē ar tā avotvalodas teikumu.</p>
<p>Ievaddati: Obligātie:</p> <ul style="list-style-type: none"> • evaluationSubtaskId – vērtēšanas apakšuzdevuma identifikators.
<p>Apstrāde: Izmantojot nejaušas varbūtības ģeneratoru (skat. Nejaušu teikumu izvēles varbūtība), serveris izvēlas teikumu, kuru pieprasīt no datubāzes kopā ar tā avotvalodas teikumu.</p>
<p>Izvaddati: JSON objekts ar nejauši izvēlētu teikumu un tā avotvalodas teikumu.</p>
<p>Kļūdu paziņojumi: Ja vērtētājs ir novērtējis visus teikumus, ko viņš drīkst novērtēt, tad viņš saņems tukšuma kļūdu 204 un paziņojumu, ka visi teikumi ir novērtēti.</p>



4.5. attēls. Funkcijas “GetRandomSentence” projektējuma algoritma blokshēma.

PPS prasības: 3.3.2.12.
Funkcija: GetGlobalStatistics
Mērķis: Lietotājam ir jāredz visu viņa izveidotu vērtēšanas uzdevumu statistiku.
Ievaddati: Obligātie: <ul style="list-style-type: none"> • activeTaskId – izvēlēto vērtēšanas uzdevumu identifikatoru saraksts.
Apstrāde: Serveris nosūta datubāzei pieprasījumu atgriezt visas automātiskās metriku vērtības, kas ietilpst izvēlēto vērtēšanas uzdevumu sarakstā, kā arī visus vērtējumus, kas ietilpst vērtēšanas uzdevumos activeTaskId identifikatoru sarakstā. No šīm vērtībām tiek izrēķināta Spearman korelācijas matrica un Kendall Tau korelācijas matrica starp vērtējumiem un katru automātisko metriku.
Izvaddati: JSON objekts ar automātisko vērtību vidējo vērtību sarakstu.
Kļūdu paziņojumi: Ja lietotājs nav autentificējies sistēmai, tad viņš saņems autentifikācijas kļūdu 401.

PPS prasības: 3.3.2.4.
Funkcija: GetEvaluationMarks
Mērķis: Lietotājam ir jāredz visi vērtējumi, ko ir novērtējis vērtētājs.
Ievaddati: Obligātie: <ul style="list-style-type: none"> • taskId – vērtēšanas uzdevuma identifikators. • subtaskId – vērtēšanas apakšuzdevuma identifikators.
Apstrāde:

Serveris nosūta datubāzei pieprasījumu atgriezt visus vērtējumus, kas ietilpst vērtēšanas apakšuzdevumā. Serveris pēc vērtējumu iegūšanas apstrādā tos un atzīme, kurš vērtējums tika vērtēts uz kontroles teikuma, uz atkārtota teikuma vai uz sabojāta teikuma.

Izvaddati:

JSON objekts ar vērtējumu sarakstu.

Kļūdu paziņojumi:

Ja lietotājs nav autentificējies sistēmai, tad viņš saņems autentifikācijas kļūdu 401.

4.23. tabula

PPS prasības: 3.3.2.11.

Funkcija:

GetAllEvaluationMarks

Mērķis:

Lietotājam ir jābūt iespējai lejuplādēt csv failu ar visiem vērtējumiem vērtēšanas uzdevumā.

Ievaddati:

Obligātie:

- taskId – vērtēšanas uzdevuma identifikators.

Apstrāde:

Serveris nosūta datubāzei pieprasījumu atgriezt visus vērtējumus, kas ietilpst vērtēšanas uzdevumā.

Izvaddati:

JSON objekts ar vērtējumu sarakstu.

Kļūdu paziņojumi:

Ja lietotājs nav autentificējies sistēmai, tad viņš saņems autentifikācijas kļūdu 401.

4.24. tabula

PPS prasības: 3.3.2.6.

Funkcija:

PostEvaluationMark

Mērķis:

Vērtētājam, novērtējot teikumu, jābūt iespējai saglabāt vērtējumu datubāzē.
<p>Ievaddati:</p> <p>Obligātie:</p> <ul style="list-style-type: none"> • subtaskId – vērtēšanas apakšuzdevuma identifikators. • sentenceId – novērtētā teikuma identifikators. • mark – vērtējums. • isDegraded – vai teikums tika sabojāts.
<p>Apstrāde:</p> <p>Serveris nosūta datubāzei pieprasījumu pievienot vērtējumu.</p>
<p>Izvaddati:</p> <p>Paziņojums par darbības rezultātu.</p>
<p>Kļūdu paziņojumi:</p> <p>Ja vērtētājs cenšas ievadīt vērtējumu, kas neietilpst vērtēšanas skalā, tad viņš saņems slikta pieprasījuma kļūdu 400.</p>

4.25. tabula

<p>PPS prasības: 3.3.2.6.</p> <p>Funkcija:</p> <p>GetMarkCount</p>
<p>Mērķis:</p> <p>Vērtētājam ir jāredz, cik teikumus viņš ir novērtējis.</p>
<p>Ievaddati:</p> <p>Obligātie:</p> <ul style="list-style-type: none"> • subtaskId – vērtēšanas apakšuzdevuma identifikators.
<p>Apstrāde:</p> <p>Serveris nosūta datubāzei pieprasījumu atgriezt visu apakšuzdevuma vērtējumu skaitu.</p>
<p>Izvaddati:</p> <p>JSON objekts ar vērtējumu skaitu.</p>
<p>Kļūdu paziņojumi:</p> <p>Nav.</p>

4.3.5. Nejaušu teikumu izvēles varbūtība

Vērtētājam teikumi tiks doti nejaušā secībā, bet pēc konkrētas struktūras. Pirmie 25 teikumi, ko vērtētājs vērtēs, būs vienādi ar citiem vērtētājiem vērtēšanas uzdevumā, lai noskaidrotu vērtētāju vienprātību. Pēc 25 teikumiem vērtētājam tiks doti nejauši izvēlēti, nenovērtēti teikumi no teikumu kopas, kur teikuma vieta korpusā ir vienāda ar kāda cita novērtēta teikuma vietu korpusā. Ja šī kopa ir tukša, tad vērtētājam tiks doti nejauši izvēlēti, nenovērtēti teikumi. Šis tiek darīts, lai, salīdzinot divas vai vairākas sistēmas, būtu novērtēti vienādi teikumi, nevis vienai sistēma vieglāki, otrai grūtāki.



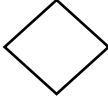
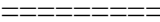

Pēc 40 teikumiem vērtētājam tiks doti teikumi pēc šādas struktūras:

- Ar varbūtību 0,1 vērtētājam tiks dots teikums, kuru viņš jau ir novērtējis.
- Ar varbūtību 0,1 vērtētājam tiks dots teikums, kuru viņš jau ir novērtējis, bet šis teikums tiks tīši sabojāts (skat. #54220 – Kontroles teikumu kontrole)
- Ar varbūtību 0,1 vērtētājam tiks dots teikums, kuru ir novērtējis kāds cits vērtētājs. Ja tāda teikuma nav, tad viņam tiek dots nejauši izvēlēts, nenovērtēts teikums.
- Ar varbūtību 0,7 vērtētājam tiks dots nejauši izvēlēts, nenovērtēts teikums.

4.4. Datubāzes projektējums

4.4.1. Datubāzes konceptuālais entītiņu relāciju modelis

4.26. tabula Konceptuālā modeļa apzīmējumu tabula

Apzīmējums	Paskaidrojums
	Īpašība
	Entītiņa
	Piederība
	Ārējās atslēgas piederība
	Īpašības piederība
<u>Id</u>	Primārā atslēga entītiņai

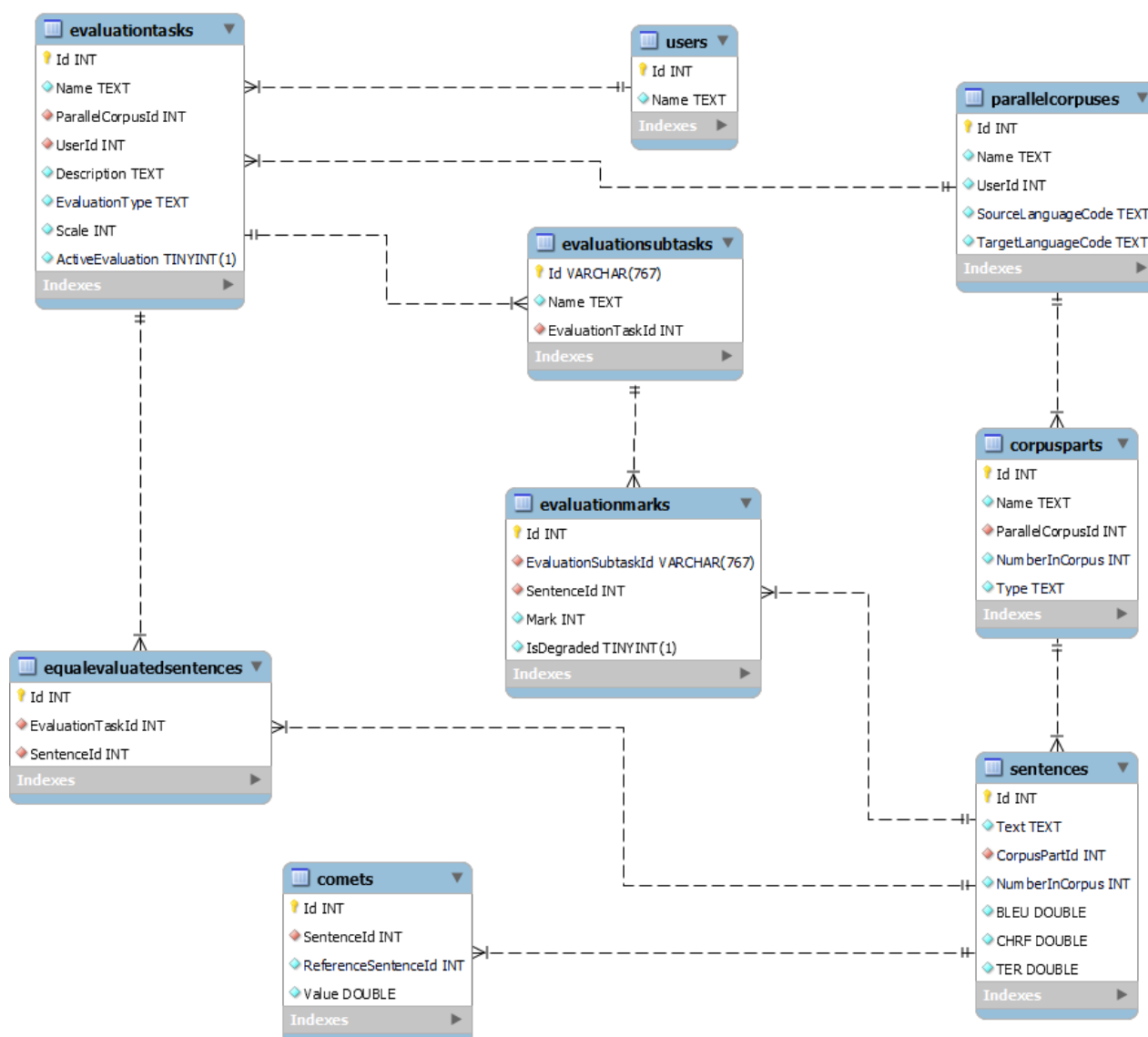


4.6. attēls. Konceptuālais modelis

Galvenās entītijas ir “Paralēlais korpus”, “Vērtēšanas uzdevums” un “Lietotājs”. Entītija “Paralēlais korpus” satur informāciju lietotnē ievadītajiem paralēlajiem korpusiem. Pie katra paralēlā korpusa ir piesaistītas vismaz divas korpusa daļas, kas pēc tipa ir avotvaloda (source) un mašintulkots (hypothesis). Katrai korpusa daļai pieder vienāds skaits teikumu un katram teikumam pieder aprēķinātas comet metrikas, kas tiek aprēķinātas no katras atsauces korpusa daļas paralēlajā korpusā. Entītija “Vērtēšanas uzdevums” satur informāciju par paralēlā korpusa novērtēšanas

uzdevumu. Vērtēšanas uzdevumam ir vērtēšanas apakšuzdevumi, kas tiek doti vērtētājiem kā hipersaites. Vērtēšanas apakšuzdevumi novērtē teikumus un veido vērtējumus, kas satur informāciju par vērtējumu un teikumu degradāciju. Vērtēšanas uzdevumam ir piesaistīti vienādi novērtēti teikumi, kas tiek doti katram vērtētājam, lai noteiktu viņu vienprātību. Entīcija “Lietotājs” satur informāciju ar lietotāja vārdu un kādus vērtēšanas uzdevumus viņš ir izveidojis.

4.4.2. Fiziskais modelis



4.7. attēls. Fiziskais datubāzes modelis

4.4.3. Datubāzes tabulu apraksts

4.27. tabula Datubāzes tabulu apraksts

Datubāzes tabulas nosaukums	Tabulas apraksts
parallelcorpus	Paralēlā korpusa dati.
corpusparts	Korpusa daļu dati.
sentences	Teikuma dati.
comets	Viena no automātiskajām novērtēšanas metrikām teikumam pret katru atsaucē korpusa daļas teikumu.
users	Reģistrētu lietotāju dati.
evaluationtasks	Vērtēšanas uzdevums.
evaluationsubtasks	Vērtēšanas apakšuzdevums.
evaluationmarks	Vērtētāja vērtējums par teikumu.
equalevaluatedsentences	Glabā teikumus, kurus dos katram vērtētājam uzdevumā, lai katrs vērtētājs uzdevumā būtu novērtējis vienādu, fiksētu skaitu teikumu.

4.4.4. Entītiņu apraksts

Datubāžu tabulu aprakstam izmantoju šādu struktūru:

- Lauka nosaukums
- Datu tips
- Cita informācija (NOT NULL, NULL, FOREIGN KEY utt.)
- Apraksts

4.4.4.1. Paralēlo korpusu tabula

4.28. tabula Datubāzes tabula “parallelcorpus”

Lauka nosaukums	Datu tips	Cita informācija	Apraksts
Id	INT	PRIMARY KEY IDENTITY	Identifikators
Name	TEXT	NOT NULL	Paralēlā korpusa nosaukums
SourceLanguageCode	TEXT	NOT NULL	Avotvalodas kods
TargetLanguageCode	TEXT	NOT NULL	Mērķvalodas kods
UserId	INT	NOT NULL	Lietotāja identifikators

4.4.4.2. Korpusa daļu tabula

4.29. tabula Datubāzes tabula “corpusparts”

Lauka nosaukums	Datu tips	Cita informācija	Apraksts
Id	INT	PRIMARY KEY IDENTITY	Identifikators
Name	TEXT	NOT NULL	Korpusa daļas nosaukums
ParallelCorpusId	INT	FOREIGN KEY	Ārējā atslēga entītijai “parallelcorpus”
NumberInCorpus	INT	NOT NULL	Kārtas numurs paralēlajā korpusā
Type	TEXT	NOT NULL	Korpusa tips (avotvaloda, atsauces vai sistēmas tulkojums)

4.4.4.3. Automātiskās metrikas comet tabula

4.30. tabula Datubāzes tabula “comets”

Lauka nosaukums	Datu tips	Cita informācija	Apraksts
Id	INT	PRIMARY KEY IDENTITY	Identifikators
SentenceId	INT	FOREIGN KEY	Ārēja atslēga entītijai “sentences”
ReferenceSentenceId	INT	NOT NULL	Atsauces korpusa daļas teikuma identifikators
Value	DOUBLE	NOT NULL	Automātiskās metrikas COMET vērtība

Comet metrikai ir nepieciešama atsevišķa tabula, jo tā tiek rēķināta pret katru atsauces korpusa daļu, tāpēc izveidojas

4.4.4.4. Lietotāju tabula

4.31. tabula Datubāzes tabula “users”

Lauka nosaukums	Datu tips	Cita informācija	Apraksts
Id	INT	PRIMARY KEY IDENTITY	Identifikators
Name	TEXT	NOT NULL	Lietotājvārds

Šajā tabulā tiek glabāti visi lietotāji, kas kaut ko sistēmai kādreiz ir pievienojuši vai izmainījuši, nevis visi lietotāji, kas var pieslēgties sistēmai.

4.4.4.5. Vērtēšanas uzdevumu tabula

4.32. tabula Datubāzes tabula “evaluationtasks”

Lauka nosaukums	Datu tips	Cita informācija	Apraksts
Id	INT	PRIMARY KEY IDENTITY	Identifikators
Name	TEXT	NOT NULL	Vērtēšanas uzdevuma nosaukums
ParallelCorpusId	INT	NOT NULL	Ārējā atslēga entītijai “parallelcorpus”
Description	TEXT	NULL	Vērtēšanas uzdevuma apraksts
EvaluationType	TEXT	NOT NULL	Vērtēšanas veids(Likert vai Direct Assessment)
Scale	INT	NOT NULL	Vērtēšanas skalas diapozona (no nulle līdz Scale)
ActiveEvaluation	BOOLEAN	NOT NULL	Apzīmē, vai vērtēšanas uzdevums ir aktīvs
UserId	INT	FOREIGN KEY	Ārēja atslēga entītijai “users”

Katru vērtēšanas uzdevumu var izslēgt, lai nevarētu pievienot vēl vērtējumus. Ja uzdevums ir izslēgts, tad katram vērtējamam vērtēšanas logā parādīsies paziņojums, ka uzdevums ir izslēgts, un viņš nespēs novērtēt nevienu teikumu.

4.4.4.6. Vērtēšanas apakšuzdevumu tabula

4.33. tabula Datubāzes tabula “evaluationstasks”

Lauka nosaukums	Datu tips	Cita informācija	Apraksts
Id	VARCHAR(767)	PRIMARY KEY IDENTITY	Identifikators GUID formātā
Name	TEXT	NOT NULL	Vērtēšanas apakšuzdevuma nosaukums
EvaluationTaskId	INT	NOT NULL	Ārējā atslēga entītijai “evaluationtasks”

Katram vērtētājam tiks izveidots viens vērtēšanas apakšuzdevums un tam būs unikāla primārā atslēga GUID formātā. Ar šīs atslēgas palīdzību lietotāja saskarnē reģistrētajam lietotājam būs iespēja nokopēt pilnu hipersaiti, kas tiks iedota vērtētājam, kurā vērtētājs varēs novērtēt teikumus vērtēšanas uzdevumā.

4.4.4.7. Vērtējumu tabula

4.34. tabula Datubāzes tabula “evaluationmarks”

Lauka nosaukums	Datu tips	Cita informācija	Apraksts
Id	ID	PRIMARY KEY IDENTITY	Identifikators
EvaluationSubtaskId	VARCHAR(767)	NOT NULL	Ārējā atslēga entītijai “evaluationstasks”
SentenceId	INT	NOT NULL	Ārējā atslēga entītijai “sentences”
Mark	INT	NOT NULL	Vērtējums
IsDegraded	BOOLEAN	NOT NULL	Apzīmē, vai ir novērtēts sabojāts teikums

4.4.4.8. Teikumu tabula

4.35. tabula Datubāzes tabula “sentences”

Lauka nosaukums	Datu tips	Cita informācija	Apraksts
Id	ID	PRIMARY KEY IDENTITY	Identifikators
Text	TEXT	NOT NULL	Teikums
CorpusPartId	INT	NOT NULL	Ārējā atslēga entītijai “corpusparts”
NumberInCorpus	INT	NOT NULL	Kārtas numurs korpusā
BLEU	DOUBLE	NOT NULL	Automātiskā metrika BLEU
CHRF	DOUBLE	NOT NULL	Automātiskā metrika CHRF
TER	DOUBLE	NOT NULL	Automātiskā metrika TER

4.4.4.9. Vienādi novērtēto teikumu tabula

4.36. tabula Datubāzes tabula “equalevaluatedsentences”

Lauka nosaukums	Datu tips	Cita informācija	Apraksts
Id	INT	PRIMARY KEY IDENTITY	Identifikators
EvaluationTaskId	INT	NOT NULL	Ārējā atslēga entītijai “evaluationtasks”
SentenceId	INT	NOT NULL	Ārējā atslēga entītijai “sentences”

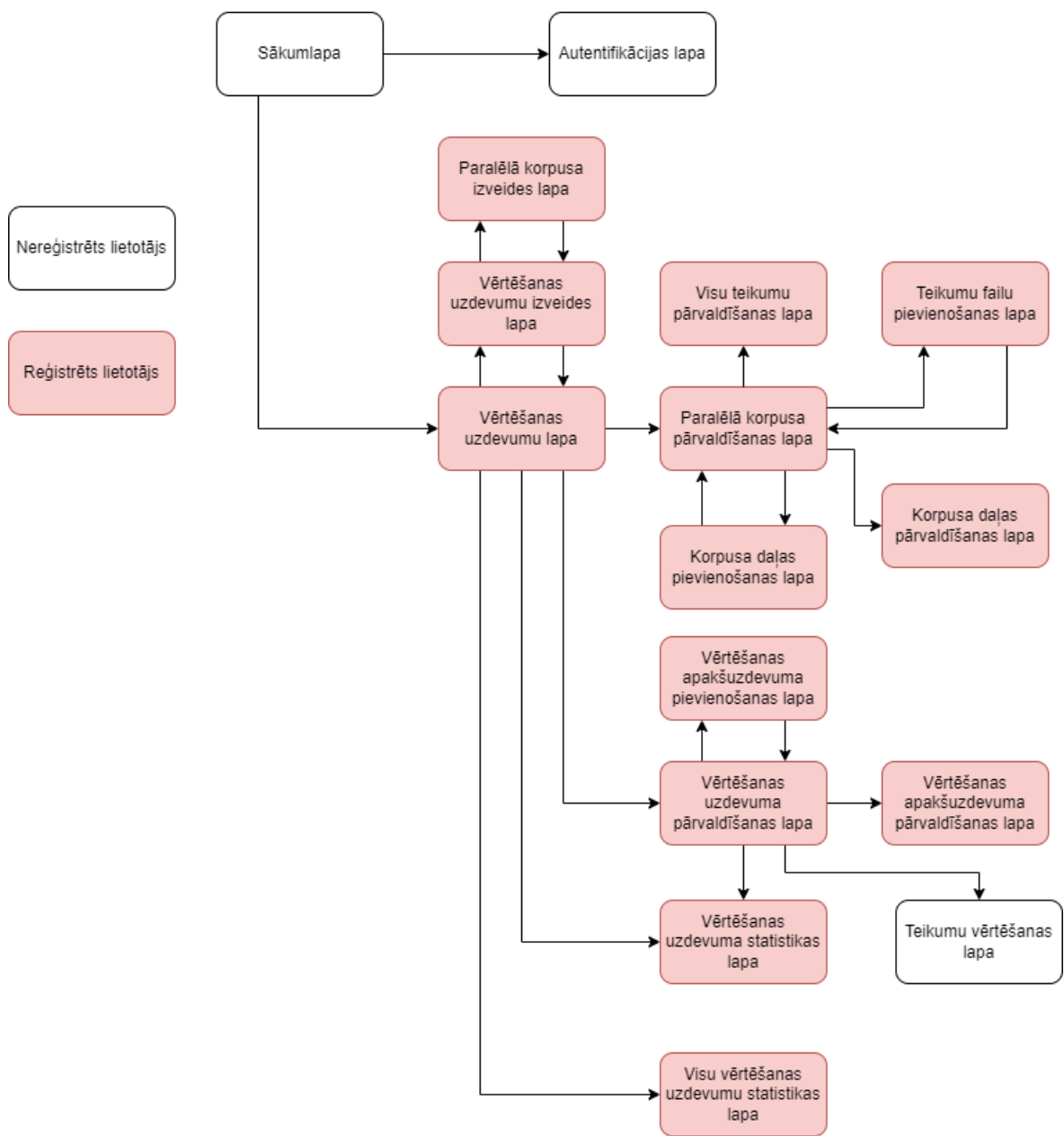
Katram vērtētājam vērtēšanas pirmie teikumi (pēc noklusētās vērtības 25 teikumi) tiks doti vienādi, lai varētu novērtēt vērtētāju korelāciju. Pa vērtēšanas procesu ar noteiktu varbūtību (skat. Nejausu teikumu izvēles varbūtība) arī tiks doti teikumi, ko vērtēja kāds cits vērtētājs vai ko vērtēs kāds cits vērtētājs, ja tādi nav pieejami.

4.5. Saskarnes projektējums

4.5.1. Ievads

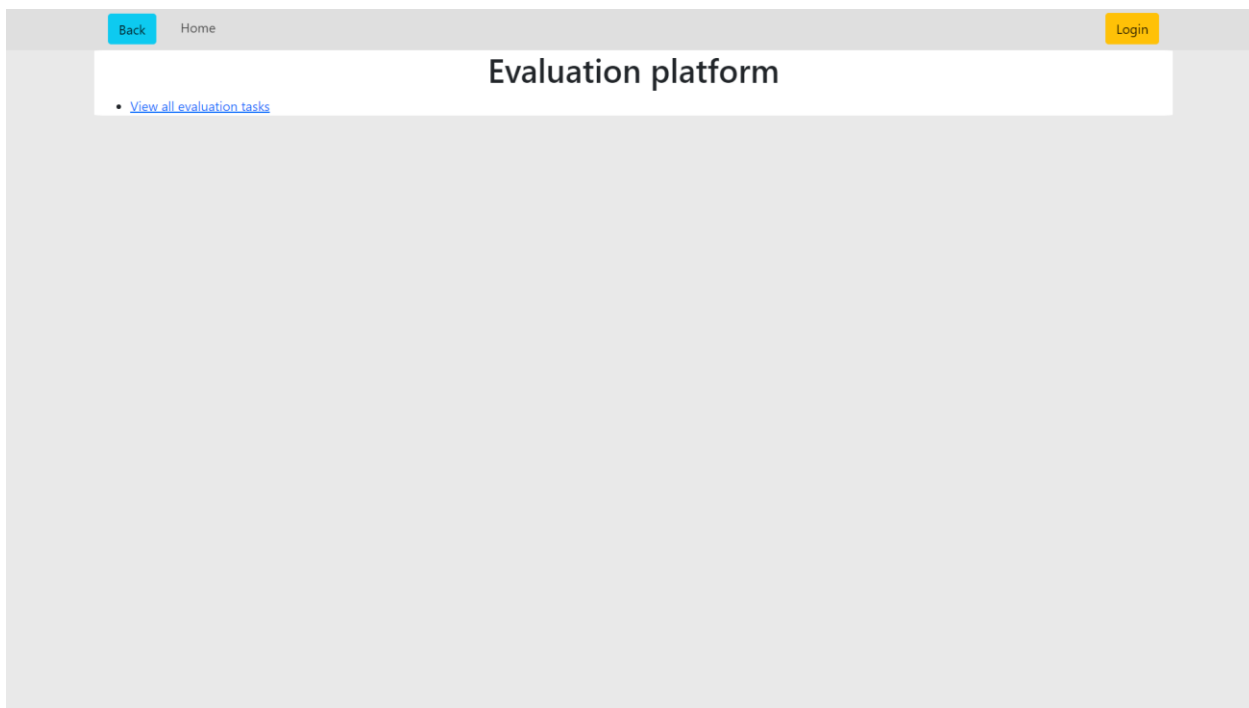
Katrā saskarnes projektējumā ir redzama navigācijas josla, kurā atrodas poga, kas aizvedīs atpakaļ, autentificēšanas poga un navigācijas pogas, kas aizvedīs atpakaļ uz attiecīgo skatu. Reģistrēts lietotājs, autentificējoties sistēmā, nokļūs vērtēšanas uzdevumu skatā, no kura viņš varēs nokļūt tālāk. Ja reģistrēts lietotājs centīsies nokļūt viņam nepieejamā skatā, viņam parādīsies paziņojums par to, ka viņam nav piekļuve šim skatam.

Neregistrēts lietotājs var redzēt tikai sākumlapu un teikumu vērtēšanas skatu. Ja neregistrēts lietotājs centīsies nokļūt viņam nepieejamā skatā, sistēma viņu nogādās uz autentifikācijas skatu.



4.8. attēls. Saskaņņu dialoga diagramma

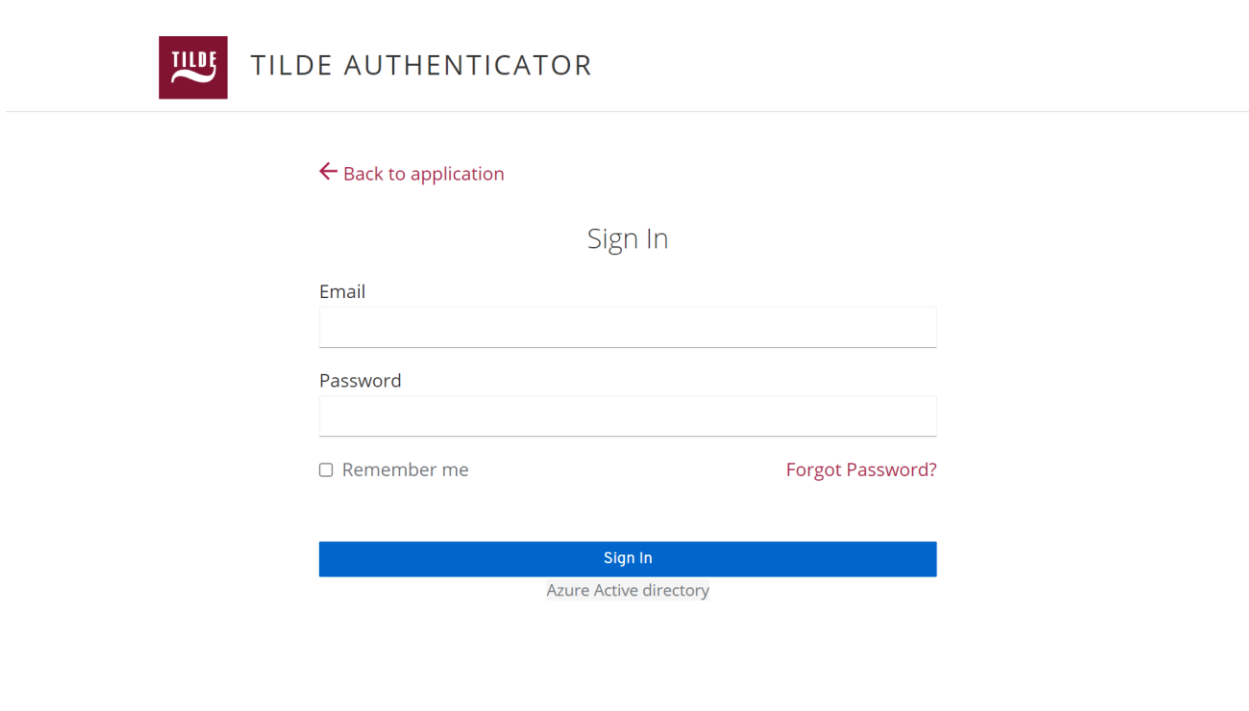
4.5.2. Sākumlapas skats



4.9. attēls. Sākumlapas skats

Sākumlapu var redzēt jebkurš lietotājs, gan reģistrēti lietotāji, gan vērtētāji. Reģistrēts lietotājs var autentificēties divos veidos: nospiežot pogu “Login”, kas atrodas navigācijas joslas labajā pusē, vai nospiežot saiti “View evaluation tasks”, kas pēc autentificēšanas viņu aizvedīs uz vērtēšanas uzdevumu lapu.

4.5.3. Autentifikācijas lapas skats



TILDE AUTHENTICATOR

← Back to application

Sign In

Email

Password

Remember me [Forgot Password?](#)

Sign In

Azure Active directory

4.10. attēls. Autentificēšanas lapas skats

Šajā skatā ir redzama autentificēšanas ievades lauki: epasts un parole. Kad lauki ir aizpildīti un nospiesta poga "Sign in", lietotājs tiek atgriezts atpakaļ uz vērtēšanas uzdevumu lapā kā autentificējies lietotājs.

4.5.4. Vērtēšanas uzdevumu lapas skats

The screenshot shows the 'Evaluation platform' interface. At the top, there are 'Back' and 'Logout' buttons. Below the title, there is a link 'View all evaluation tasks'. The main section is titled 'Evaluation Tasks' and contains two buttons: 'Add a task' and 'Automatic metric correlation with human judgements'. Below this is a table with the following columns: ID, Name, Type, Scale, Parallel corpus id, Source language code, Target language code, Parallel corpus, Evaluators, Statistics, Task deletion, and Active evaluation. The table contains six rows of tasks.

ID	Name	Type	Scale	Parallel corpus id	Source language code	Target language code	Parallel corpus	Evaluators	Statistics	Task deletion	Active evaluation
1	Ciešāks labs slikts starp sistēmām	Direct Assessment	100	1	lv	en	View parallel corpus	View evaluators	View task statistics	Delete task	<input checked="" type="checkbox"/>
2	Labs slikts starp sistēmām	Likert	6	2	lv	en	View parallel corpus	View evaluators	View task statistics	Delete task	<input checked="" type="checkbox"/>
4	Tiešais vērtējums	Direct Assessment	100	5	lv	en	View parallel corpus	View evaluators	View task statistics	Delete task	<input checked="" type="checkbox"/>
5	PValue test	Direct Assessment	100	6	lv	en	View parallel corpus	View evaluators	View task statistics	Delete task	<input checked="" type="checkbox"/>
6	Artūra faili	Direct Assessment	100	7	en	lv	View parallel corpus	View evaluators	View task statistics	Delete task	<input checked="" type="checkbox"/>

4.11. attēls. Vērtēšanas uzdevumu lapas skats

Vērtēšanas uzdevumu lapā ir redzami visi autentificētā lietotāja izveidotie vērtēšanas uzdevumi un informācija par tiem. Virs vērtēšanas uzdevumu saraksta ir iespējams nospiegt pogu “Add a task”, kas aizvedīs uz vērtēšanas uzdevumu izveides lapu, kā arī pogu “Automatic metric correlation with human judgements”, kas aizvedīs uz visu vērtēšanas uzdevumu lapu. Pie katra vērtēšanas uzdevuma ir to informācija un pogas “View parallel corpus”, kas aizvedīs uz vērtēšanas uzdevuma piesaistītā paralēlā korpusa pārvaldīšanas lapu, “View evaluators”, kas aizvedīs uz vērtēšanas uzdevuma pārvaldīšanas lapu, un “View task statistics”, kas aizvedīs uz vērtēšanas uzdevuma statistikas lapu. Poga “Delete task” izdzēsīs vērtēšanas uzdevumu un izvēles rūtiņa atzīmē, vai vērtēšanas uzdevuma vērtēšana ir aktīva vai nē, ko var arī nospiegt un nomainīt.

4.5.5. Vērtēšanas uzdevuma pievienošanas lapas skats

The screenshot shows a web interface for creating an evaluation task. At the top, there is a navigation bar with a 'Back' button, the path 'Home / Evaluation tasks / Add', and a 'Logout' button. The main heading is 'Evaluation platform'. Below it, there is a link to 'View all evaluation tasks'. The form is titled 'Evaluation task Creation' and contains the following fields:

- Name***: A text input field with the placeholder text 'Name'.
- Parallel Corpus***: A dropdown menu with the placeholder text '-- Choose a parallel corpus --'.
- Evaluation Type***: A dropdown menu with the placeholder text '-- Choose an evaluation type --'.
- Scale (Mark count. If left at 0, default count for Likert will be 7 (0-6), Direct Assessment - 101 (0-100))**: A text input field with the value '0'.
- Description***: A large text area with the placeholder text 'Description'.
- Auto-generated subtask count***: A text input field with the value '0'.

A blue 'Submit' button is located at the bottom left of the form.

4.12. attēls. Vērtēšanas uzdevuma pievienošanas lapas skats

Vērtēšanas uzdevuma pievienošanas lapā ir redzams forma ar dažādiem ievades laukiem:

- Name – vērtēšanas uzdevuma nosaukums,
- Parallel Corpus – izvēlne ar pieejamiem paralēlajiem korpusiem,
- Evaluation type – izvēlne ar Likert un Direct Assessment,
- Scale – vērtēšanas skalas diapozons,
- Description – vērtēšanas uzdevuma apraksts, kas tiks parādīts vērtētājam, piemēram, pēc kādiem kritērijiem ir jāvērtē vērtēšanas uzdevums.
- Auto-generated subtask count – automātiski uzģenerēto vērtēšanas apakšuzdevumu skaits.

Paralēlā korpusa izvēlnē ir opcija “Add a new parallel corpus”, kurš aizvedīs uz paralēlā korpusa izveides lapu.

Aizpildot visus laukus un nospiežot pogu “Submit”, sistēma aizvedīs atpakaļ uz vērtēšanas uzdevumu lapu.

4.5.6. Paralēlā korpusa pievienošanas lapas skats

[Back](#) Home / Parallel Corpus / Add [Logout](#)

Evaluation platform

- [View all evaluation tasks](#)

Parallel Corpus Creation

Parallel Corpus

Name*

Source language code*

Target language code*

Corpus part 1

System name: Source
Type: Source

 No file chosen

Corpus part 2

System name*

Type: Hypothesis

 No file chosen

Corpus part 3

System name*

Type*

 No file chosen

4.13. attēls. Paralēlā korpusa pievienošanas lapas skats

Paralēlā korpusa pievienošanas lapā ir redzams forma ar paralēlā korpusa informācijas ievadlaukiem un korpusa daļu ievadlaukiem. Paralēlajam korpusam ir 3 ievadlauki:

- Name – paralēlā korpusa nosaukums,
- Source language code – avotvalodas kods,
- Target language code – mērķvalodas kods.

Pirmā korpusa daļa, kas atrodas paralēlajā korpusā vienmēr ir avotvalodas korpusa daļa, kur ir nepieciešams augšuplādēt teikumu failu. Otrā korpusa daļa vienmēr ir hipotēze, kurai ir nepieciešams ievadīt korpusa daļas vārdu un augšuplādēt teikumu failu. Lapas apakšpusē ir redzama poga “Add Corpus Part”, kas jaunu tukšu formu ar ievadlaukiem:

- System name – sistēmas nosaukums,
- Type – izvēlne ar Hypothesis (hipotēze) un Target reference (atsauce).
- Upload file – faila augšuplāde ar teikumiem.
- Remove corpus part – poga, kas noņems esošo korpusa daļu no formas.

Aizpildot visus laukus un nospiežot pogu “Submit”, sistēma aizvedīs atpakaļ uz vērtēšanas uzdevumu pievienošanas lapu ar izvēlētu tikko pievienoto paralēlo korpusu, kā arī sāks rēķināt automātiskās metrikas teikumiem.

4.5.7. Paralēlā korpusa pārvaldīšanas lapas skats

The screenshot shows the 'Corpus parts' management interface. At the top, there are navigation links: 'Back', 'Home / Evaluation tasks / Corpus Parts', and 'Logout'. The main heading is 'Evaluation platform'. Below the heading, there is a link 'View all evaluation tasks'. The 'Corpus parts' section contains three buttons: 'Add a corpus part', 'View sentences in parallel corpus', and 'Reevaluate sentences'. A table lists the corpus parts with columns for System Name, Number in corpus, Type, and Sentences. Each row has 'View sentences' and 'Upload with file' buttons.

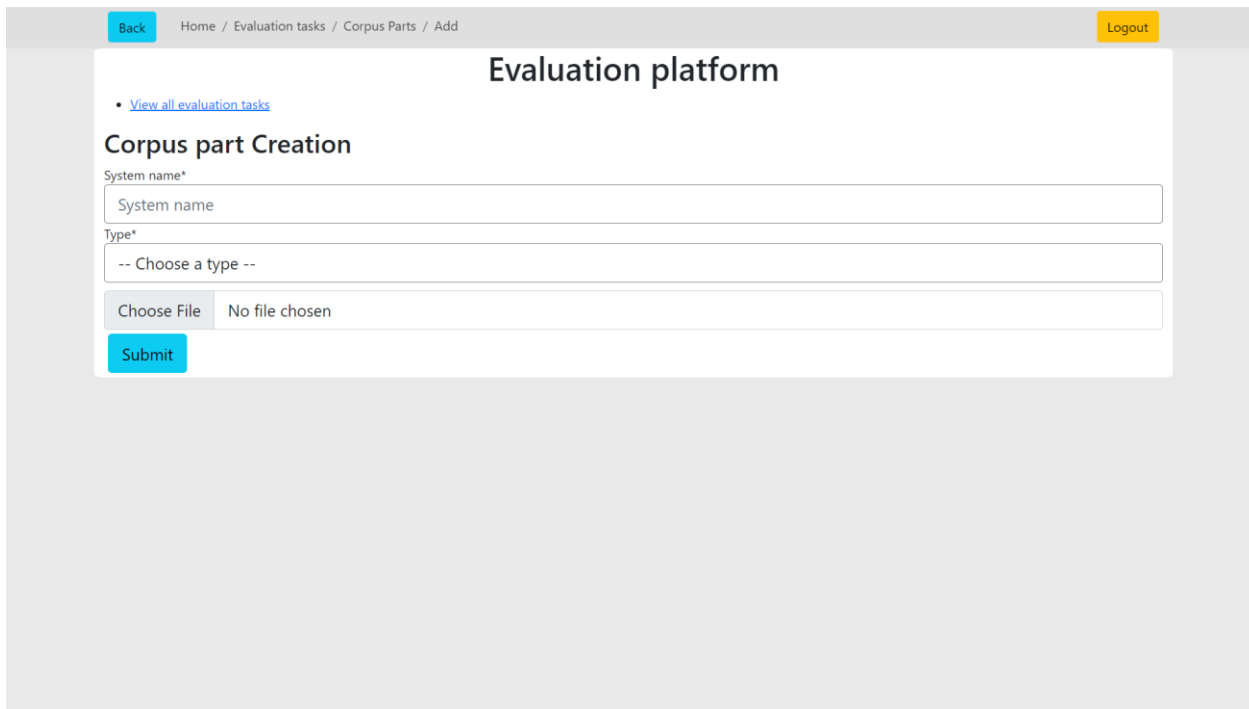
System Name	Number in corpus	Type	Sentences
Source	1	Source	View sentences Upload with file
Baseline	2	Hypothesis	View sentences Upload with file
Improved	3	Hypothesis	View sentences Upload with file
Reference	4	Target	View sentences Upload with file

4.14. attēls. Paralēlā korpusa pārvaldīšanas lapas skats

Paralēlā korpusa pārvaldīšanas lapā ir redzami visas korpusa daļas, kas atrodas paralēlajā korpusā. Virs korpusa daļu saraksta ir iespējams nospiegt pogas “Add a corpus part”, kas aizvedīs uz korpusa daļas pievienošanas lapu, “View sentences in parallel corpus”, kas aizvedīs uz visu teikumu pārvaldīšanas skatu, un “Reevaluate sentences”, kas pārrēķinās paralēlajam korpusam

automātiskās metrikas. Pie katras korpusa daļas ir to informācija un pogas “View sentences”, kas aizvedīs uz korpusa daļas pārvaldīšanas lapu, un poga “Upload sentences”, kuru nospiežot, tiks izdzēsti visi teikumi korpusa daļā un aizvedīs uz teikumu failu pievienošanas lapu.

4.5.8. Korpusa daļas pievienošanas lapas skats



The screenshot shows a web interface titled "Evaluation platform". At the top left, there is a "Back" button and a breadcrumb trail: "Home / Evaluation tasks / Corpus Parts / Add". At the top right, there is a "Logout" button. Below the title, there is a link: "• [View all evaluation tasks](#)". The main section is titled "Corpus part Creation" and contains the following form elements:

- A text input field labeled "System name*" with the placeholder text "System name".
- A dropdown menu labeled "Type*" with the placeholder text "-- Choose a type --".
- A file upload area with a "Choose File" button and the text "No file chosen".
- A blue "Submit" button.

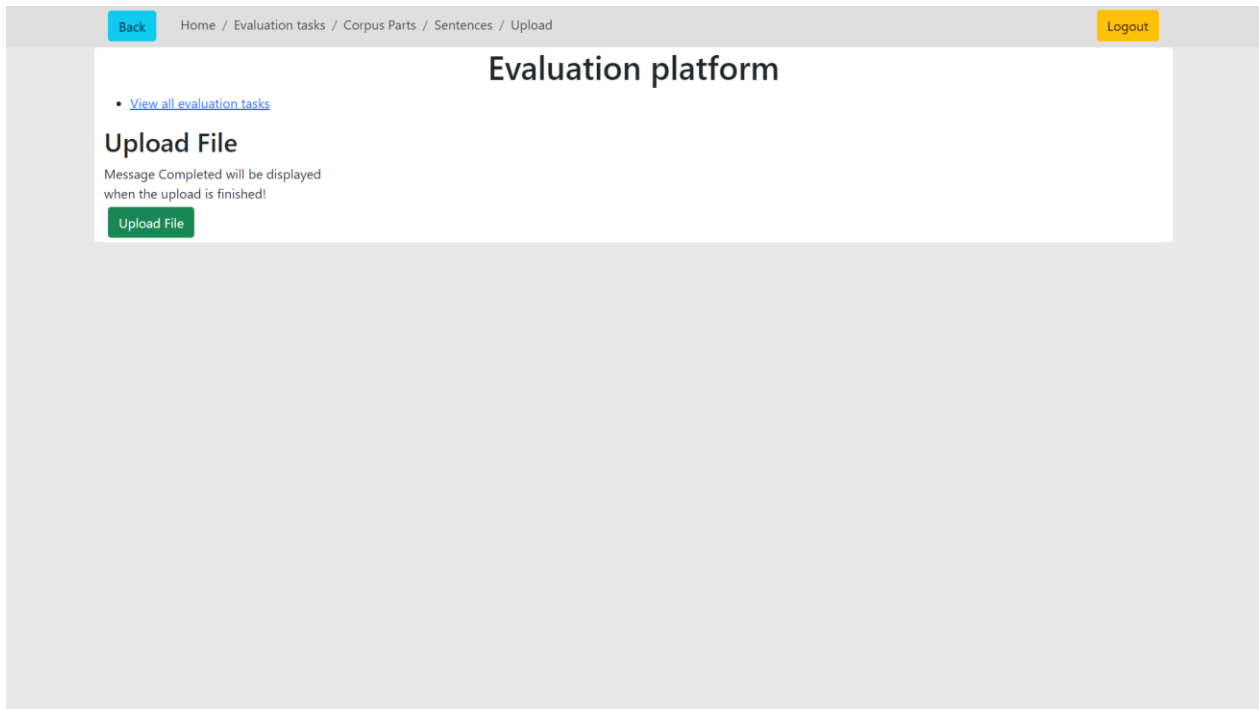
4.15. attēls. Korpusa daļas pievienošanas lapas skats

Vērtēšanas uzdevuma pievienošanas lapā ir redzams forma ar dažādiem ievades laukiem:

- System name – sistēmas nosaukums,
- Type – izvēlne ar Hypothesis (hipotēze) un Target reference (atsauce).
- Upload file – faila augšuplāde ar teikumiem.

Aizpildot visus laukus un nospiežot pogu “Submit”, sistēma aizvedīs atpakaļ uz paralēlā korpusa pārvaldīšanas lapu, kā arī sāks pārrēķināt automātiskās metrikas teikumiem.

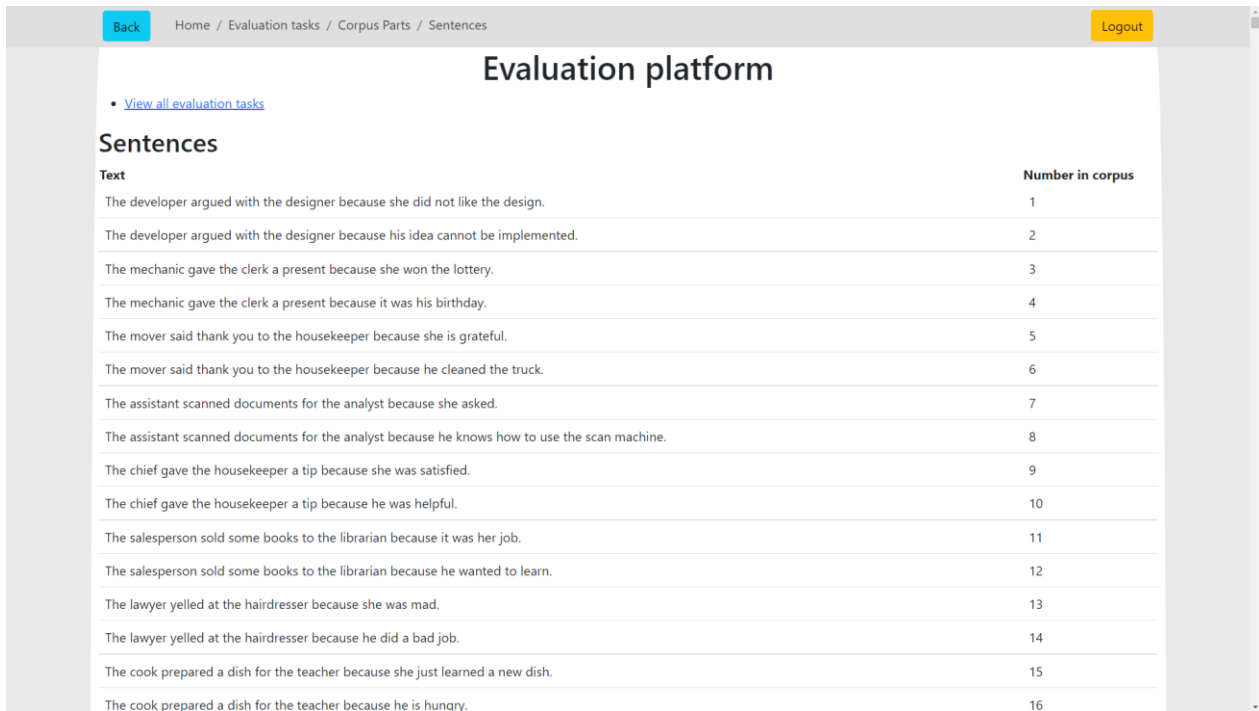
4.5.9. Teikumu faila pievienošanas lapas skats



4.16. attēls. Teikumu faila pievienošanas lapas skats

Šajā skatā ir redzama poga “Upload file”, kas augšuplādēs failu ar teikumiem korpusa daļā. Kad ielāde ir pabeigusies, sistēma aizvedīs atpakaļ uz paralēlā korpusa pārvaldīšanas lapu, kā arī sāks pārreķināt automātiskās metrikas teikumiem.

4.5.10. Korpusa daļu pārvaldīšanas lapas skats



The screenshot shows a web interface for an evaluation platform. At the top, there is a navigation bar with a 'Back' button, a breadcrumb trail 'Home / Evaluation tasks / Corpus Parts / Sentences', and a 'Logout' button. The main heading is 'Evaluation platform'. Below it, there is a link 'View all evaluation tasks'. The main content is a table titled 'Sentences' with two columns: 'Text' and 'Number in corpus'. The table lists 16 sentences, each with a corresponding number from 1 to 16.

Text	Number in corpus
The developer argued with the designer because she did not like the design.	1
The developer argued with the designer because his idea cannot be implemented.	2
The mechanic gave the clerk a present because she won the lottery.	3
The mechanic gave the clerk a present because it was his birthday.	4
The mover said thank you to the housekeeper because she is grateful.	5
The mover said thank you to the housekeeper because he cleaned the truck.	6
The assistant scanned documents for the analyst because she asked.	7
The assistant scanned documents for the analyst because he knows how to use the scan machine.	8
The chief gave the housekeeper a tip because she was satisfied.	9
The chief gave the housekeeper a tip because he was helpful.	10
The salesperson sold some books to the librarian because it was her job.	11
The salesperson sold some books to the librarian because he wanted to learn.	12
The lawyer yelled at the hairdresser because she was mad.	13
The lawyer yelled at the hairdresser because he did a bad job.	14
The cook prepared a dish for the teacher because she just learned a new dish.	15
The cook prepared a dish for the teacher because he is hungry.	16

4.17. attēls. Korpusa daļu pārvaldīšanas lapas skats

Korpusa daļu pārvaldīšanas lapā ir redzams saraksts ar visiem teikumiem korpusa daļā un to vietu korpusā daļā.

4.5.11. Visu teikumu pārvaldīšanas lapas skats

Home / Evaluation tasks / Corpus Parts / Sentences

Logout

Evaluation platform

- [View all evaluation tasks](#)

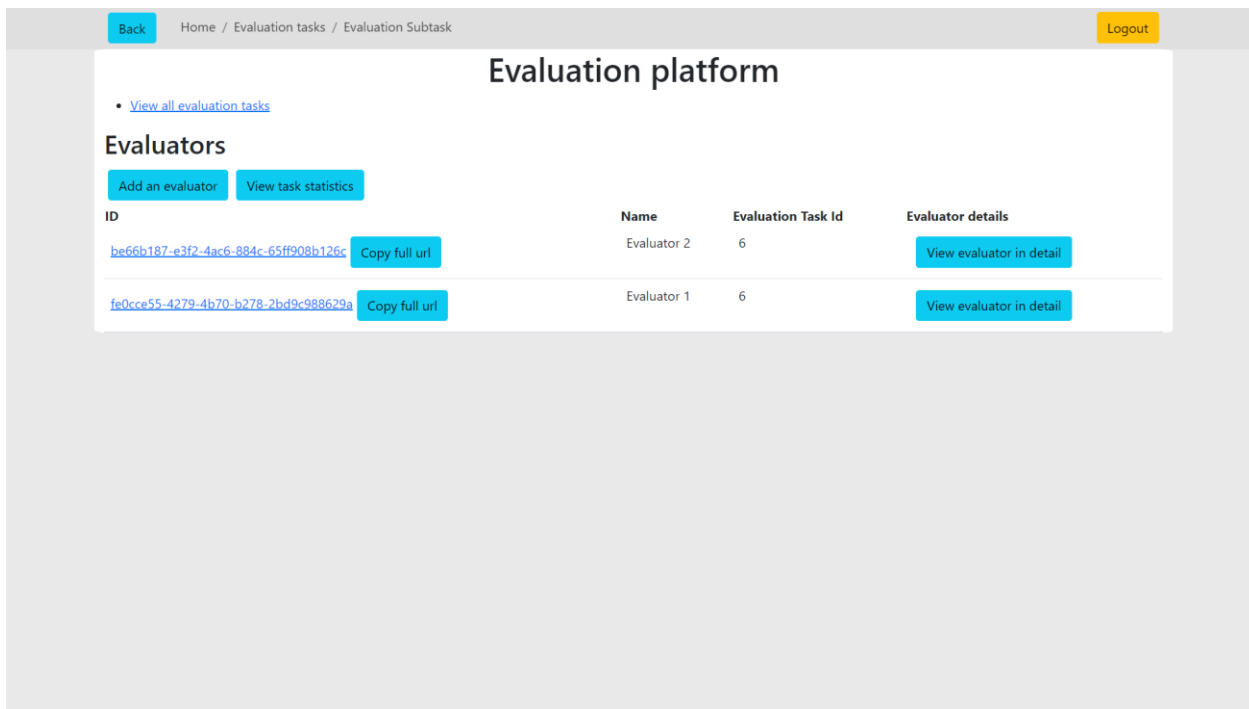
Sentences

Number in corpus	Source	Baseline	Improved	Reference
1	The developer argued with the designer because she did not like the design.	attīstītājs strīda ar dizaineru , jo viņai nepatika dizains .	izstrādātājs strīdējās ar dizaineri , jo viņai šis dizains nepatika .	Šarlote strīdējās ar dizaineri , jo viņai šis dizains nepatika .
2	The developer argued with the designer because his idea cannot be implemented.	attīstītājs strīda ar dizaineri , jo viņa ideju nevar īstenot .	izstrādātājs strīdējās ar dizaineri , jo viņa ideju nav iespējams īstenot .	viņš strīdējās ar dizaineri , jo viņa ideju nevar īstenot .
3	The mechanic gave the clerk a present because she won the lottery.	mehāniķis atdeva pulksteni , jo viņa uzvarēja loterijā .	mehāniķis iedeva pārdevējam dāvanu , jo viņa uzvarēja loterijā .	mehāniķe pasniedza pārdevējam dāvanu , jo viņa uzvarēja loterijā .
4	The mechanic gave the clerk a present because it was his birthday.	mehāniķis atdeva pulksteni , jo tā bija viņa dzimšanas diena .	mehāniķis iedeva pārdevējam dāvanu , jo bija viņa dzimšanas diena .	mehāniķis iedeva ierēdnim dāvanu , jo bija viņa dzimšanas diena .
5	The mover said thank you to the housekeeper because she is grateful.	pūlis pateicās māsaiņiecei , jo viņa ir pateicīga .	pārzinis pateicās saimniecības vadītājai , jo viņa ir pateicīga .	māte pateicās saimniecības vadītājai , jo viņa ir pateicīga .
6	The mover said thank you to the housekeeper because he cleaned the truck.	pūlis teica paldies māsaiņiekam , jo viņš notīrīja kravas mašīnu .	namzīnis pateicās saimniecības vadītājai , jo iztīrīja mašīnu .	pārvietotājs pateicās saimniecības vadītājai , jo tīrīja mašīnu .
7	The assistant scanned documents for the analyst because she asked.	viņa vaicāja , ka palīdzē skenēja uz analītiķa plociem rakstītus dokumentus .	asistents skenēja analītiķa dokumentus , jo viņa to jautāja .	asistente skenēja dokumentus analītiķei , jo jautāja .
8	The assistant scanned documents for the analyst because he knows how to use the scan machine.	asistents skenēja analītiķa dokumentus , jo viņš zina , kā izmantot smlīšu mašīnu .	asistents skenēja analītiķa dokumentus , jo zina , kā izmantot skenēšanas iekārtu .	asistents skenēja dokumentus analītiķim , jo zina , kā izmantot skenēšanas iekārtu .
9	The chief gave the housekeeper a tip	the chief gave the house keeper a tip	virsaitis deva mājkalpotājai padomu , jo	virsaitis pasniedza saimniecības

4.18. attēls. Visu teikumu pārvaldīšanas lapas skats

Visu teikumu pārvaldīšanas lapā ir redzams saraksts ar visiem teikumiem paralēlajā korpusā un kuriem teikumiem tie atbilst pēc vietas korpusa daļā.

4.5.12. Vērtēšanas uzdevuma pārvaldīšanas lapas skats



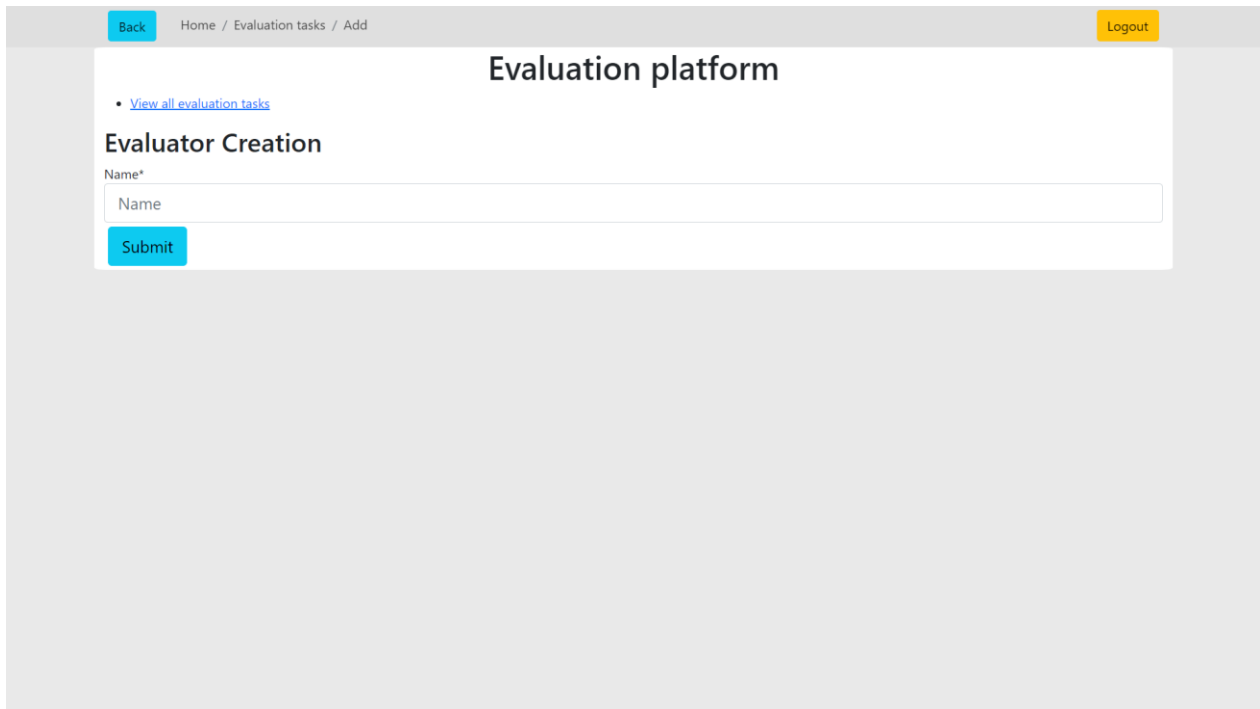
The screenshot displays the 'Evaluation platform' interface. At the top, there is a navigation bar with a 'Back' button, a breadcrumb trail 'Home / Evaluation tasks / Evaluation Subtask', and a 'Logout' button. Below the navigation bar, the main content area is titled 'Evaluation platform'. Underneath, there is a link to 'View all evaluation tasks'. The 'Evaluators' section contains two buttons: 'Add an evaluator' and 'View task statistics'. Below these buttons is a table listing evaluators. The table has columns for 'ID', 'Name', 'Evaluation Task Id', and 'Evaluator details'. Two evaluators are listed: 'Evaluator 2' and 'Evaluator 1', both associated with 'Evaluation Task Id' 6. Each ID is accompanied by a 'Copy full url' button, and each row has a 'View evaluator in detail' button.

ID	Name	Evaluation Task Id	Evaluator details
be66b187-e3f2-4ac6-884c-65ff908b126c Copy full url	Evaluator 2	6	View evaluator in detail
fe0cce55-4279-4b70-b278-2bd9c988629a Copy full url	Evaluator 1	6	View evaluator in detail

4.19. attēls. Vērtēšanas uzdevuma pārvaldīšanas lapas skats

Vērtēšanas uzdevuma pārvaldīšanas lapā ir redzami visi vērtēšanas apakšuzdevumi un informācija par tiem. Virs saraksta ir pogas “Add an evaluator”, kas aizvedīs uz vērtēšanas apakšuzdevuma pievienošanas lapu, un “View task statistics”, kas aizvedīs uz vērtēšanas uzdevuma statistikas lapu. Katru identifikatoru sarakstā var nospiegt un nokļūt uz attiecīgo vērtēšanas apakšuzdevuma vērtēšanas lapu. Blakus šim identifikatoram ir poga “Copy full url”, kas nokopēs vērtēšanas apakšuzdevuma saiti uz vērtēšanas skatu, ko reģistrēts lietotājs var sūtīt vērtētājam. Pie katra vērtēšanas apakšuzdevuma ir redzama poga “View evaluator in detail”, kas aizvedīs uz vērtēšanas apakšuzdevuma pārvaldīšanas lapu.

4.5.13. Vērtēšanas apakšuzdevuma pievienošanas lapas skats

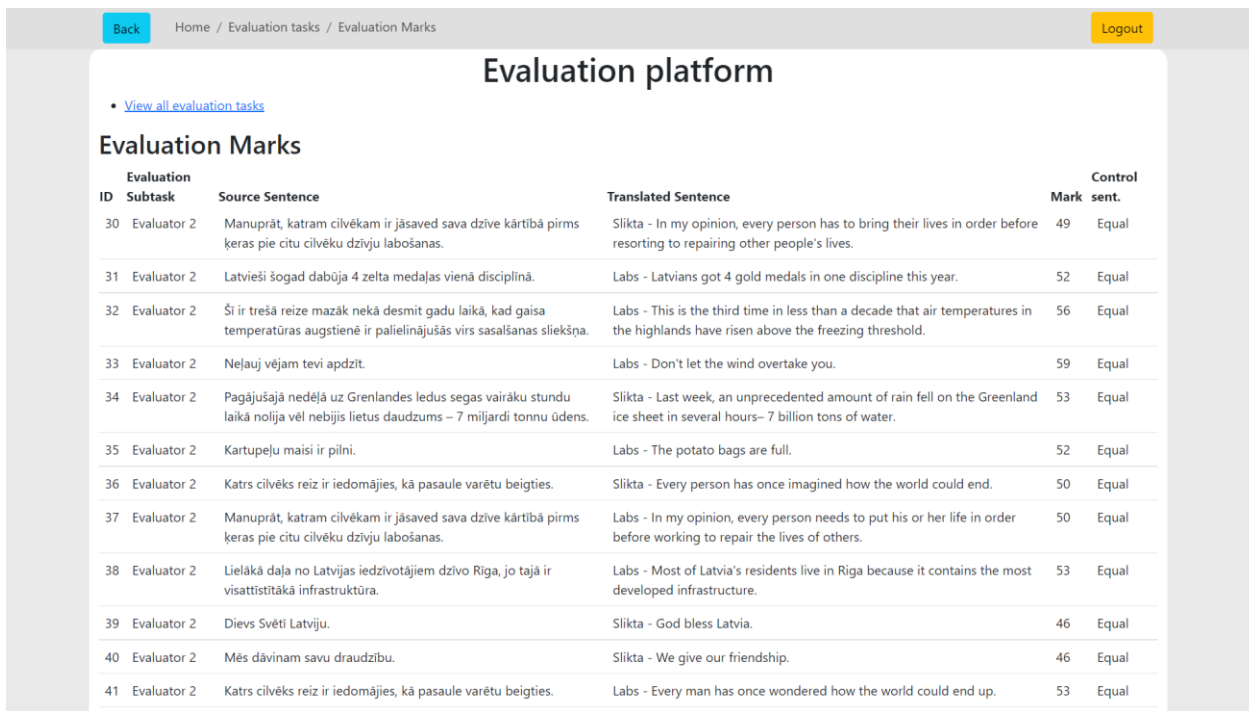


The screenshot displays a web interface for an evaluation platform. At the top, there is a navigation bar with a 'Back' button on the left and a 'Logout' button on the right. The main heading is 'Evaluation platform'. Below the heading, there is a link: '• [View all evaluation tasks](#)'. The primary section is titled 'Evaluator Creation'. It contains a form with a label 'Name*' and a text input field with the placeholder text 'Name'. Below the input field is a blue 'Submit' button.

4.20. attēls. Vērtēšanas apakšuzdevuma pievienošanas lapas skats

Vērtēšanas apakšuzdevuma pievienošanas skatā ir redzama forma, kur ir jāievada vērtētāja nosaukums. Kad ievada nosaukumu un nospiež pogu “Submit”, sistēma aizved atpakaļ uz vērtēšanas uzdevuma pārvaldīšanas skatu.

4.5.14. Vērtēšanas apakšuzdevuma pārvaldīšanas lapas skats



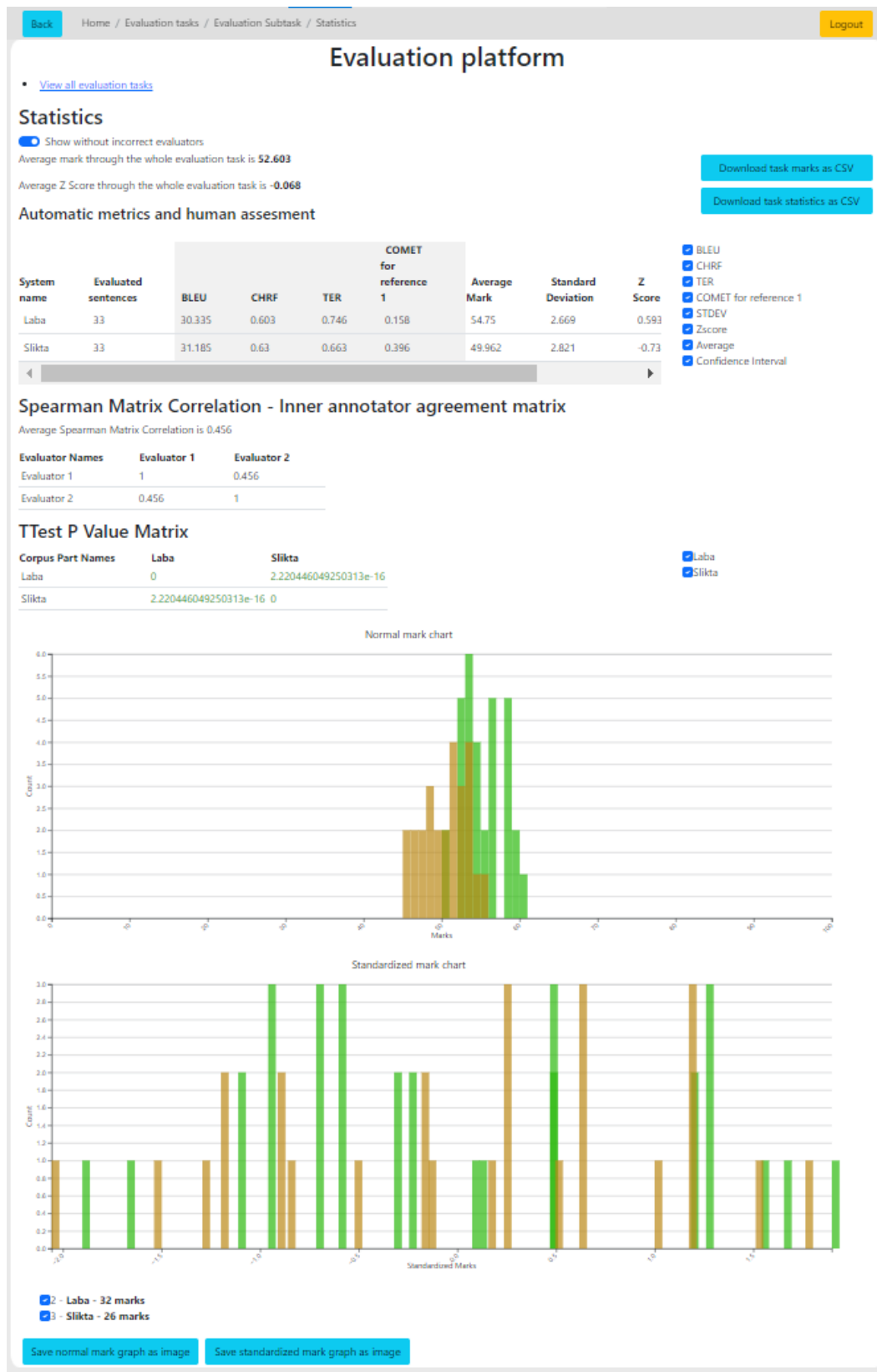
The screenshot shows a web interface for an evaluation platform. At the top, there are navigation links for 'Back' and 'Logout'. The main heading is 'Evaluation platform'. Below it, there is a link to 'View all evaluation tasks'. The main content is a table titled 'Evaluation Marks' with the following columns: 'Evaluation ID', 'Subtask', 'Source Sentence', 'Translated Sentence', 'Mark', and 'Control sent.'. The table contains 12 rows of data, each representing an evaluation task with its ID, subtask, source sentence, translated sentence, mark, and control status.

Evaluation ID	Subtask	Source Sentence	Translated Sentence	Mark	Control sent.
30	Evaluator 2	Manuprāt, katram cilvēkam ir jāsaved sava dzīve kārtībā pirms ķeras pie citu cilvēku dzīvju labošanas.	Slihta - In my opinion, every person has to bring their lives in order before resorting to repairing other people's lives.	49	Equal
31	Evaluator 2	Latvieši šogad dabūja 4 zelta medaļas vienā disciplīnā.	Labs - Latvians got 4 gold medals in one discipline this year.	52	Equal
32	Evaluator 2	Ši ir trešā reize mazāk nekā desmit gadu laikā, kad gaisa temperatūras augstienē ir palielinājušās virs sasalšanas sliekšņa.	Labs - This is the third time in less than a decade that air temperatures in the highlands have risen above the freezing threshold.	56	Equal
33	Evaluator 2	Nelauj vējam tevi apdzīt.	Labs - Don't let the wind overtake you.	59	Equal
34	Evaluator 2	Pagājušajā nedēļā uz Grenlandes ledus segas vairāku stundu laikā nolija vēl nebijis lietus daudzums – 7 miljardi tonnu ūdens.	Slihta - Last week, an unprecedented amount of rain fell on the Greenland ice sheet in several hours – 7 billion tons of water.	53	Equal
35	Evaluator 2	Kartupeļu maiši ir pilni.	Labs - The potato bags are full.	52	Equal
36	Evaluator 2	Katrs cilvēks reiz ir iedomājies, kā pasaule varētu beigties.	Slihta - Every person has once imagined how the world could end.	50	Equal
37	Evaluator 2	Manuprāt, katram cilvēkam ir jāsaved sava dzīve kārtībā pirms ķeras pie citu cilvēku dzīvju labošanas.	Labs - In my opinion, every person needs to put his or her life in order before working to repair the lives of others.	50	Equal
38	Evaluator 2	Lielākā daļa no Latvijas iedzīvotājiem dzīvo Rīgā, jo tajā ir visattīstītākā infrastruktūra.	Labs - Most of Latvia's residents live in Riga because it contains the most developed infrastructure.	53	Equal
39	Evaluator 2	Dievs Svēti Latviju.	Slihta - God bless Latvia.	46	Equal
40	Evaluator 2	Mēs dāvinām savu draudzību.	Slihta - We give our friendship.	46	Equal
41	Evaluator 2	Katrs cilvēks reiz ir iedomājies, kā pasaule varētu beigties.	Labs - Every man has once wondered how the world could end up.	53	Equal

4.21. attēls. Vērtēšanas apakšuzdevuma pārvaldīšanas lapas skats

Vērtēšanas apakšuzdevuma pārvaldīšanas lapā ir redzams saraksts ar visiem vērtēšanas apakšuzdevuma vērtējumiem. Pie katra vērtējuma ir arī informācija par teikumu, kuru vērtēja, avotvalodas teikums un vai tas teikums bija kontroles teikums. Uzbraucot ar kursoru uz Control sent. parādās papildinformācija lietotājam, kas ir kontroles teikums.

4.5.15. Vērtēšanas uzdevuma statistikas lapas skats



4.22. attēls. Vērtēšanas uzdevuma statistikas lapas skats

Vērtēšanas uzdevuma statistikas lapā ir apkopota visa statistika par vērtēšanas uzdevumu. Lapas augšdaļā ir redzama izvēles rūtiņa, kas nosaka, vai statistikā tiks iekļauti vērtētāji, kam pieder nederīgi vērtējumi. Nospiežot šo rūtiņu, tiks pārrēķināta visa statistikas lapa bez vai ar šiem vērtējumiem. Lapas labajā malā ir divas pogas:

- Download task marks as CSV – lejuplādēt csv failu ar visiem vērtēšanas uzdevuma vērtējumiem.
- Download task statistics as CSV - lejuplādēt csv failu ar visu vērtēšanas uzdevuma statistiku.

Pirmā redzamā tabula, cik novērtēti teikumi ir katrai sistēmai vidējo vērtību BLEU, CHRF, TER, katrai COMET vērtībai un vērtējumiem, standartnovirzi, Z vērtību un pārlicības intervālu. Labajā pusē tabulai ir iespējams paslēpt katru vērtību ar izvēles rūtiņu. Zem šīs tabulas ir redzama Spearman korelācijas matrica, TTest vērtību matrica, Kendall tau matrica un diagrammas ar vērtējumiem un standartizētajiem vērtējumiem. Uz katru stabiņu diagrammā ir iespējams uzbraukt ar kursoru un tiks izgaismota attiecīgās korpusa daļas vērtējumi un dota papildinformācija par to vērtējumu kopu. Zem diagrammām ir iespējams paslēpt korpusa daļas ar izvēles rūtiņu.

4.5.16. Visu vērtēšanas uzdevumu statistikas lapas skats

The screenshot shows the 'Evaluation platform' interface. At the top, there are navigation links: 'Back' (blue), 'Home / Evaluation tasks / Global statistics' (grey), and 'Logout' (yellow). The main content area is titled 'Evaluation platform' and includes a link to 'View all evaluation tasks'. Below this, the section 'Automatic metric correlation with human judgement' is displayed. It contains two tables: 'Spearman matrix' and 'Kendall Tau matrix'. Each table has columns for 'Mark type', 'BLEU', 'CHRF', 'TER', and 'COMET'. The 'Spearman matrix' shows values for 'Normal marks' (0.0658, 0.0658, 0.0658, -0.0626) and 'Standardized marks' (0.0103, 0.0103, 0.0103, -0.2063). The 'Kendall Tau matrix' shows values for 'Normal marks' (-1.6202, -1.6202, -1.6202, -1.8174) and 'Standardized marks' (-2.0693, -2.0693, -2.0693, -2.2967). At the bottom, there is a list of checkboxes for filtering the data: 'Ciešāks labs slihts starp sistēmām', 'Labs slihts starp sistēmām', 'Tiešais vērtējums', 'PValue test', and 'Artūra faili'.

Back Home / Evaluation tasks / Global statistics Logout

Evaluation platform

[View all evaluation tasks](#)

Automatic metric correlation with human judgement

Spearman matrix

Mark type	BLEU	CHRF	TER	COMET
Normal marks	0.0658	0.0658	0.0658	-0.0626
Standardized marks	0.0103	0.0103	0.0103	-0.2063

Kendall Tau matrix

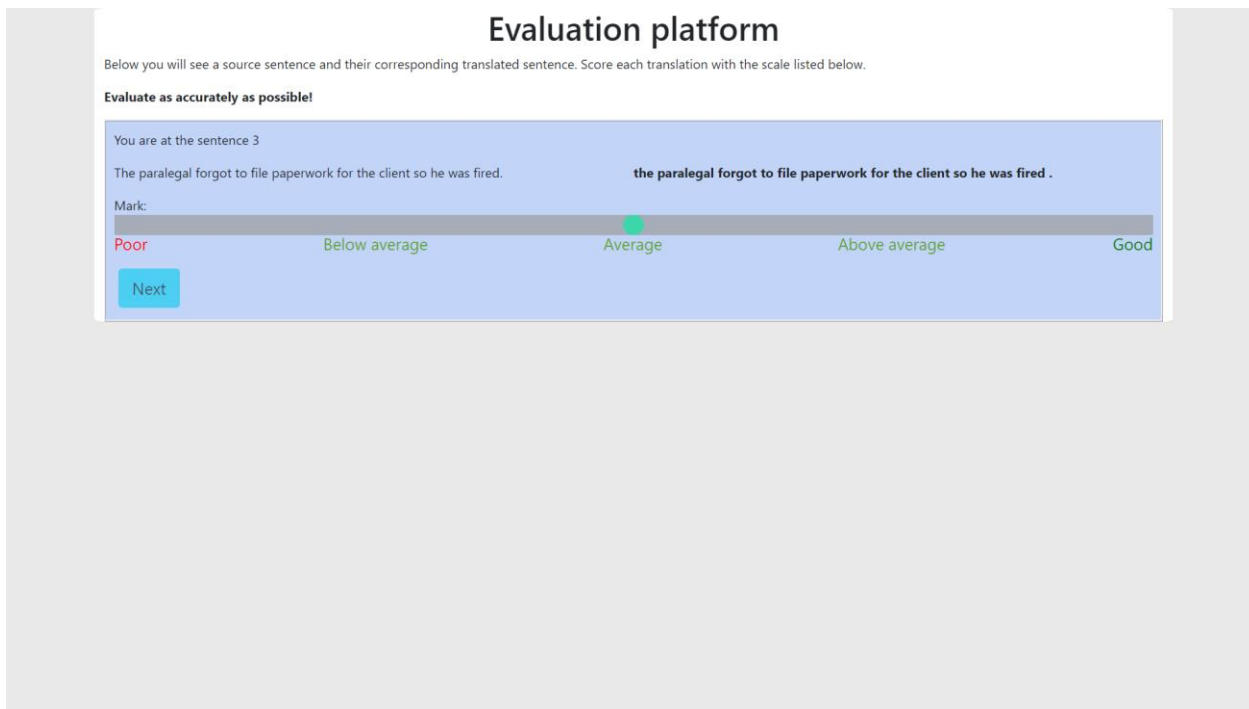
Mark type	BLEU	CHRF	TER	COMET
Normal marks	-1.6202	-1.6202	-1.6202	-1.8174
Standardized marks	-2.0693	-2.0693	-2.0693	-2.2967

- Ciešāks labs slihts starp sistēmām
- Labs slihts starp sistēmām
- Tiešais vērtējums
- PValue test
- Artūra faili

4.23. attēls. Visu vērtēšanas uzdevumu statistikas lapas skats

Visu vērtēšanas uzdevumu statistikas lapā ir redzama Spearman korelācijas matrica ar Kendall Tau korelācijas matricu. Zem tabulām ir saraksts ar vērtēšanas uzdevumiem un izvēles rūtiņām. Reģistrētais lietotājs var izvēlēties, kurus vērtēšanas uzdevumus viņš vēlas iekļaut matricu aprēķināšanā.

4.5.17. Teikumu vērtēšanas lapas skats



4.24. attēls. Teikumu vērtēšanas lapas skats

Teikumu vērtēšanas lapai nav pieejama navigācijas josla, jo tā vērtētājam nav nepieciešama. Skatā ir redzams vērtētājam iedotā instrukcija, ko ir iedevusi gan sistēma, gan reģistrēts lietotājs, kas izveidoja vērtēšanas uzdevumu. Skatā ir redzams, kurā teikumā ir vērtētājs, kreisajā pusē ir redzams avotvalodas teikums un labajā pusē treknrakstā ir redzams mašīntulkots teikums, kuru vērtētājam ir jānovērtē, izmantojot skalu un slīdni, kas atrodas tieši zem teikumiem. Kustinot to, virs skalas dinamiski atjaunojas vērtētāja dotais vērtējums un, kad vērtētājs ir izvēlējies novērtēt teikumu, nospiežot pogu “Next”, viņam parādās jauna teikumu kopa, ko viņam atkal ir jānovērtē.

5. TESTĒŠANAS DOKUMENTĀCIJA

5.1. Testēšanas metodika

Kvalifikācijas darbā izstrādātais programmatūras kods tiks testēts vairākās metodēs. Katras jaunas funkcionalitātes izstrādes beigās tika manuāli testēta jaunā funkcionalitāte, izmantojot manuāli vadītus testus, lai pārbaudītu, vai tā darbojas pareizi. Funkcijas un formas tika testētas gan ar korektiem, gan ar nekorektiem datiem.

Automātiskie vienībtesti sistēmai tika rakstīti, izmantojot Microsoft.NET.Test.Sdk bibliotēku. Katrs vienībtests tika rakstīts, lai pārbaudītu, vai funkcija pareizi pievieno datus datubāzei. Manuālie testi tika veikti uz forma aizpildīšanu un korektu vai nekorektu datu ievadi, lai lietotājam tiktu doti pareizie paziņojumi un pareizais rezultāts pēc ievaddatu aizpildīšanas.

5.2. Vienībtestēšana

Nr.p.k.	Nosaukums	Gaidāmais rezultāts	Statuss
1.	Should_Add_Parallel_Corpus	Pievienot paralēlo korpusu datubāzei.	OK
2.	Mark_randomizer_should_give_close_to_sentence_amount	Nejauši izvēlēto teikumu skaitam vajadzētu būt tuvam kopējam teikumu skaitam, ja kopējais teikumu skaits ir vienāds ar izvēlēto teikumu daudzumu.	OK
3.	Should_upload_files	Vajadzētu pievienot failu ar teikumiem korpusa daļai.	OK
4.	Should_not_add_marks_if_out_of_range	Nevajadzētu pievienot vērtējumus, kas ir ārpus vērtēšanas skalas.	OK
5.	Should_add_marks	Vajadzētu pievienot vērtējumus.	OK
6.	Should_give_average_statistics	Vajadzētu atgriezt pareizu vidējo vērtību.	OK
7.	Should_give_some_same_sentences_for_two_evaluators	Vajadzētu iedot diviem vērtētājiem vienādus teikumus.	OK
8.	Should_make_one_degraded_sentence	Vajadzētu iedot vērtētājam vismaz vienu degradētu teikumu.	OK

5.3. Manuālie testi

Manuālie testi tika atkārtoti ar aktuāliem datiem katru reizi, kad kāda no testa funkcionalitātēm mainījās.

Testēšanas datums	Nosaukums, ievadītie dati	Gaidāmais rezultāts	Statuss
16.06.2021.	<i>Paralēlā korpusa pievienošana,</i> name, language code	Pievienos paralēlo korpusu	OK
17.06.2021.	<i>Vērtēšanas beigas,</i> mark, poga "Next".	Veicot pēdējo vērtējumu, vērtētājam jāparādās ziņojums, ka vērtēšana ir beigusies.	OK
08.07.2021.	<i>Failu augšuplāde,</i> fails ar teikumiem	Pareizi ievadīti teikumi datubāzē	NOK (izlabots 08.07.2021.)
09.08.2021.	<i>Automātisko metriku aprēķināšana,</i> poga "Reevaluate sentences"	Paziņojums, ka sākas automātisko metriku aprēķināšana un darbības rezultāta paziņojums.	OK
29.09.2021.	<i>Vērtēšanas uzdevuma formas validācija,</i> parallel corpus, evaluation type, scale, description, auto-generated subtask count.	Ziņojums, ka visi lauki nav aizpildīti.	OK
15.12.2021.	<i>Lietotāja piekļuve citu lietotāju vērtēšanas uzdevumiem.</i>	Ziņojums, ka lietotājam nav piekļuves.	NOK (izlabots 16.12.2021.)

6. PROJEKTA PĀRVALDĪBA

6.1. Projekta organizācija

Kvalifikācijas darba izstrādātā sistēma tika veidota pēc spējās programmatūras izstrādes metodoloģijas. Visa sistēma tika pakāpeniski papildināta ar prasībām, ko noteica uzņēmums. Pirms sistēmas izstrādes bija jāapgūst programmēšanas valoda Angular, Keycloak autentifikācijas serveris un MySQL datubāzes dokumentācija.

Uzdevumi sistēmas izstrādei tika noteikti TFS versiju kontroles serverī izmantojot lietotāju stāstus, kurus veidoja projekta vadītājs. Pirms nodošanas izstrādei lietotāju stāstus izskatīja ekspertu komanda un deva laika novērtējumu. Projektu izstrādāja viens cilvēks. Koda apskati veica nozares eksperti. Ik rītu notika sapulces par darba progresu un katra mēneša beigās tika veidota sapulce, kurā tika apkopots mēneša darbs.

Sistēmas izstrādes laikā tika veidoti vienībtesti un veikti manuālie testi, kas pārbaudīja gan veiktspēju, gan sistēmas kvalitāti.

6.2. Kvalitātes nodrošināšana

Kvalifikācijas darba programmas kods tika rakstīts pēc katras valodas “Labā koda” metodikas. Komentāri tika rakstīti angļu valodā vietās, kas koda rakstītājam likās sarežģītas. Programmas koda rakstīšanai tika lietota versiju kontrole, kas ļāva viegli atgriezties vecākas versijas kodā, ja izveidotas kāda kļūme. Pirms koda augšuplādēšanas tika palaists katrai valodai koda labotājs, kas izmainīja visas koda vietas, kas tam likās nepārskatāmas, piemēram, pārāk garas rindas, nevajadzīgas atstarpes, nevajadzīgi mainīgie u.c. Izstrādes laikā notika koda pārskati, kur nozares eksperti izskatīja uzrakstīto programmatūras kodu un norādīja nepilnības, kļūdas, nekorektu formatējumu un komentāru trūkumu kodā. Lai kolēģi varētu apskatīties, testēt un novērtēt sistēmas kvalitāti saskarnes līmenī, pa izstrādes procesu sistēma tika uzstādīta uz testa servera, kas bija pieejams SIA “Tilde” lokālajā tīklā. Šis testa serveris tika periodiski atjaunots ar jaunākajām izmaiņām.

6.3. Konfigurāciju pārvaldība

Programmatūras versiju kontrolei tika lietots Git un TFS repozitoriju glabātuve. Koda izmaiņas tika augšuplādētas regulāri, lai nodrošinātu koda aktualitāti repozitorijā, kā arī lai sadalītu augšuplādes funkcionalitātes daļās ar komentāru, ko šī koda augšuplāde ir izmainījusi. Katram lietotājistāstam tika veidots savs zars, kurš pēc tam tika sapludināts izstrādes zarā (izstrādes zars

tika likts testa vidē). Git atļāva pārbaudīt izmaiņas pa koda rindiņām un nodrošināja koda piekļuvi citiem cilvēkiem uzņēmumā, kā arī atļāva nozares ekspertiem viegli un vienkārši veikt koda kvalitātes pārskatus, koda lasāmības un modularitātes uzlabošanai, kad izstrādes zaru sapludināja ar galveno zaru.

6.4. Darbietilpības novērtējums

Darbietilpības novērtējums tika veikts pēc ekspertu metodēm. Katra lietotāja stāsta funkcionalitāte tika novērtēta funkcijpunktos. Katrs funkcijpunkts ir pielīdzināms vienai cilvēka darbadienai jeb 8 darba stundām. Funkcijpunktu ietvaros tika ņemts vērā laiks patērēts mācoties programmēšanas valodas un pētīt pareizos risinājumus statistiku aprēķināšanai, bet nav ņemts vērā laiks patērēts uz dokumentācijas izstrādi.

6.1. tabula

Lietotāju stāstu identifikators	Darbietilpības novērtējums funkcijpunktos	Reālais darbietilpības funkcijpunktos
53983	6	7
54015	2	2
54068	4	4
54076	6	6
54099	2	2
54200	5	7
54220	8	10
54241	16	20
54252	6	6
54280	6	6
54325	1	1
54343	5	5
Kopā	67	76

Izmantojot šo tabulu, var secināt, ka izstrāde prasīja vairāk laika nekā tika iepļānots, kas ir raksturīgi spējās metodoloģijai, un sasniedza 3 personmēnešu darbietilpību.

7. SECINĀJUMI

Kvalifikācijas darba ietvaros tika izstrādāta platforma, kas ļauj cilvēkvērtētājiem novērtēt mašīntulkošanas sistēmas tulkošanas kvalitāti, kā arī reģistrētiem lietotājiem pārvaldīt vērtēšanas uzdevumus un sistēmas, kuras tiks novērtētas, un apskatīt novērtēto sistēmu statistiku, no kuras var secināt tulkošanas kvalitāti.

Programmatūra tika izstrādāta pēc spējās metodoloģijas, tāpēc daudzas lietas izstrādes procesā mainījās, lai precīzi pielāgotu pasūtītāja ieskatam. Jebkurā brīdī varēja precizēt pasūtītāja ieskatus, lai izstrādātā funkcionalitāte atbilstu tiktu izveidota precīzi.

Kvalifikācijas darbā tika apgūtas jaunas programmēšanas valodas, kā arī nostiprinātas iepriekšējās zināšanas par citām programmēšanas valodām. Kopā pa darbu tika apgūtas zināšanas ar šādām tehnoloģijām:

- C# programmēšanas valodas kopā ar .NET ietvaru,
- LINQ vaicājumi .NET ietvarā,
- JSON objektu struktūra,
- Angular ietvars,
- Typescript programmēšanas valoda,
- MySQL datubāze,
- Python programmēšanas valoda,
- Git versiju kontrole,
- TFS projektu pārvaldības rīks,
- HTML programmēšanas valoda,
- CSS programmēšanas valoda,
- Bootstrap ietvars.

Kvalifikācijas darba rezultāts ir ne tikai programmatūra, bet arī pieredze izstrādātājam strādāt komandā, uzlabotās dokumentācijas rakstīšanas prasmes un mašīntulkošanas teorētiskās zināšanas.

8. IZMANTOTĀ LITERATŪRA

1. Kocmi, Tom, et al. "To ship or not to ship: An extensive evaluation of automatic metrics for machine translation." arXiv preprint arXiv:2107.10821, 2021. Pieejams: <https://arxiv.org/pdf/2107.10821.pdf>
2. Barrault, Loïc, et al. "Findings of the 2019 conference on machine translation (wmt19)." Proceedings of the Fourth Conference on Machine Translation (Volume 2: Shared Task Papers, Day 1), 2019. Pieejams: <https://aclanthology.org/W19-5301.pdf>
3. Keycloak dokumentācija – [atsauce 03.01.2021.] Pieejams: <https://www.keycloak.org/documentation>
4. Latvijas Nacionālais standartizācijas un metroloģijas centrs. LVS 68:1996 PPS. PROGRAMMATŪRAS PRASĪBU SPECIFIKĀCIJAS CEĻVEDIS.– [atsauce 29.12.2021.] Pieejams: <https://estudijas.lu.lv/mod/resource/view.php?id=131427>
5. Automātiskās novērtēšanas metrikas COMET dokumentācija – [atsauce 08.01.2022.] – Pieejams: <https://unbabel.github.io/COMET/html/index.html>
6. ISO-639 valodas kodi – [atsauce 05.01.2022] – Pieejams: https://en.wikipedia.org/wiki/List_of_ISO_639-1_codes
7. GRAHAM, Y., BALDWIN, T., MOFFAT, A., & ZOBEL, J. (2017). Can machine translation systems be evaluated by the crowd alone. Natural Language Engineering, 23(1), 3-30. doi:10.1017/S1351324915000339 - Pieejams: <https://www.cambridge.org/core/journals/natural-language-engineering/article/can-machine-translation-systems-be-evaluated-by-the-crowd-alone/E29DA2BC8E6B99AA1481CC92FAB58462>
8. Koehn, Philipp. "Statistical significance tests for machine translation evaluation." Proceedings of the 2004 conference on empirical methods in natural language processing. 2004. Pieejams: <https://aclanthology.org/W04-3250.pdf>
9. Spearman korelācijas koeficients – [atsauce 03.01.2022.] Pieejams: https://en.wikipedia.org/wiki/Spearman%27s_rank_correlation_coefficient
10. Kendall Tau korelācijas koeficients – [atsauce 09.01.2022.] Pieejams: https://en.wikipedia.org/wiki/Kendall_rank_correlation_coefficient
11. Latvijas Nacionālais standartizācijas un metroloģijas centrs. LVS 72:1996 PPS. IETEICAMĀ PRAKSE PROGRAMMATŪRAS PROJEKTĒJUMA APRAKSTĪŠANAI.

– [atsauce 29.12.2021.] Pieejams:

<https://estudijas.lu.lv/mod/resource/view.php?id=131428>

12. Akadēmiskā terminu datubāze – Pieejams: <http://termini.lza.lv/term.php?>

Teikumu failu augšuplādes funkcijas

```

/// <summary>
/// Uploads a file with sentences
/// </summary>
/// <remarks>Uploads a file with sentences, which will be added to the specific
corpus part.
/// Method creates a temporary file in the system, which will then be used in the
called "UploadSentencesToDatabase" method.
/// File format is a text file and each sentence is in a new line.</remarks>
// GET: api/corpus/5/upload
[HttpPost("{corpusId}/upload"), DisableRequestSizeLimit]
public async Task<IActionResult> Upload(int corpusId)
{
    try
    {
        string username = _utilities.GetUsername(HttpContext);

        IFormFile file = Request.Form.Files[0];
        string pathToSave = Path.GetTempPath();
        if (file.Length > 0)
        {
            string fileName = Path.GetRandomFileName() + ".txt";
            string dbPath = Path.Combine(pathToSave, fileName);
            //Gets the authenticated user
            User newUser = await _context.Users.Where(x => x.Name ==
username).FirstOrDefaultAsync();
            if (newUser == null)
            {
                newUser = new User
                {
                    Name = username
                };
                _ = _context.Users.Add(newUser);
                _ = await _context.SaveChangesAsync();
                newUser = await _context.Users.Where(x => x.Name ==
username).FirstOrDefaultAsync();
            }
            //Creates a temp file in the temp directory
            await using (FileStream stream = new(dbPath, FileMode.Create))
            {
                await file.CopyToAsync(stream);
            }
            _ = await UploadSentencesToDatabase(corpusId, fileName);
            FileInfo di = new(dbPath);
            di.Delete(); //After sentence upload is done, deletes the file
            return Ok("Sentences saved!");
        }
        return BadRequest();
    }
    catch (Exception ex)
    {

```

```

        return StatusCode(500, $"Internal server error: {ex}");
    }
}
/// <summary>
/// Uploads the sentences to the database
/// </summary>
/// <remarks>Takes the specified temporary file filepath and uploads the sentences in
them to the database.</remarks>
[NonAction]
public async Task<IActionResult> UploadSentencesToDatabase(int corpusId, string
filepath)
{
    string dir = Path.GetTempPath();
    string fullPath = Path.Combine(dir, filepath);
    int highestSentenceNumber = 0;
    if (await _context.Sentences.Where(x => x.CorpusPartId == corpusId).AnyAsync())
    {
        highestSentenceNumber = await _context.Sentences.Where(x => x.CorpusPartId ==
corpusId).MaxAsync(x => x.NumberInCorpus);
    }
    highestSentenceNumber++;
    try
    {
        using (StreamReader sr = new(fullPath))
        {
            //Goes through the file where each line is a sentence and uploads them to
the database
            while (!sr.EndOfStream)
            {
                string sentences = await sr.ReadLineAsync();
                Sentence sentence = new()
                {
                    Text = sentences,
                    NumberInCorpus = highestSentenceNumber,
                    CorpusPartId = corpusId
                };
                _ = _context.Sentences.Add(sentence);

                highestSentenceNumber++;
                _ = await _context.SaveChangesAsync();
            }
        }
        return Ok("Sentences saved!");
    }
    catch (Exception e)
    {
        Console.WriteLine(e);
        return NoContent();
    }
}

```

2. Pielikums

Nejaušu teikumu izvēles funkcija

```
/// <summary>
```

```

/// Retrieves a random sentence
/// </summary>
/// <remarks>Retrieves a random sentence depending on the evaluation subtask. If the
mark count is higher than 40, it has a chance to retrieve a sentence for another
evaluation.</remarks>
// GET: api/tasks/subtasks/5/sentences
[HttpGet("subtasks/{evaluationSubtaskId}/sentences")]
[AllowAnonymous]
public async Task<ActionResult<IEnumerable<Sentence>>> GetRandomSentence(string
evaluationSubtaskId)
{
    Random rand = new();
    int parallelCorpusId = await
_context.EvaluationSubtasks.Include("EvaluationTask")
    .Where(x => x.Id == evaluationSubtaskId).Select(x =>
x.EvaluationTask.ParallelCorpusId).FirstOrDefaultAsync();
    int taskId = await _context.EvaluationSubtasks.Where(x => x.Id ==
evaluationSubtaskId)
    .Select(x => x.EvaluationTask.Id).FirstOrDefaultAsync();
    Sentence sentenceTranslated;
    Sentence sentenceSource;
    int count = await _context.EvaluationMarks.Where(x => x.EvaluationSubtaskId ==
evaluationSubtaskId)
    .Select(x => x.SentenceId).CountAsync();

    List<int> evaluatedSentences = await _context.EvaluationMarks.Where(x =>
x.EvaluationSubtaskId == evaluationSubtaskId)
    .OrderByDescending(x => x.Id).Select(x =>
x.SentenceId).Take(40).ToListAsync();

    double repeatedEqualSentences =
UtilityClass.GetValueFromConfig("RepeatedEqualSentences");

    if (count < UtilityClass.GetValueFromConfig("RepeatedEqualSentences"))
    {
        sentenceTranslated = await GetEqualEvaluatedSentences(evaluationSubtaskId,
evaluatedSentences, repeatedEqualSentences);
        if (sentenceTranslated == null)
        {
            return NoContent();
        }
    }

    //If there haven't been 40 marks already submitted, the randomizer will give
sentences that are still not evaluated.
    else if (count < 40)
    {
        List<Sentence> sentences = await GetEvaluationReadySentenceList(taskId,
parallelCorpusId, evaluationSubtaskId);

        sentenceTranslated = SentencePicker.GetRandomNotEvaluatedSentence(sentences);
        if (sentenceTranslated == null)
        {
            return NoContent();
        }
    }
}

```

```

else //When there have been at least 40 marks submitted, it will include already
evaluated sentences
{
    double evaluatedOrNot = rand.NextDouble(); // Generate a random floating
point number in the interval [0, 1)
    if (evaluatedOrNot <= 0.1) //Happens with a probability of 0.1 (10%)
    { //Takes a random already evaluated sentence.
        List<Sentence> sentences = await
_context.EvaluationMarks.Include("Sentence.CorpusPart")
//Gets evaluated sentences
.Where(x => x.EvaluationSubtaskId == evaluationSubtaskId).Select(x =>
x.Sentence).ToListAsync();
        sentenceTranslated =
SentencePicker.GetRandomEvaluatedSentence(sentences);
        sentenceTranslated.IsDegraded = false;
    }
    else if (evaluatedOrNot <= 0.2) //Happens with a probability of 0.1 (10%)
    { //Takes a random already evaluated sentence which gets degraded by
removing words from it.
        List<Sentence> sentences = await
_context.EvaluationMarks.Include("Sentence.CorpusPart")
//Gets evaluated sentences
.Where(x => x.EvaluationSubtaskId == evaluationSubtaskId).Select(x =>
x.Sentence).ToListAsync();
        sentenceTranslated =
SentencePicker.GetRandomEvaluatedSentence(sentences);
        SentencePicker.RemoveWords(sentenceTranslated);
        sentenceTranslated.IsDegraded = true;
    }
    else if (evaluatedOrNot <= 0.3) //Happens with a probability of 0.1 (10%)
    { //Takes a random sentence that has been evaluated by any other evaluator.
        sentenceTranslated = await
GetEqualEvaluatedSentences(evaluationSubtaskId, evaluatedSentences,
repeatedEqualSentences);
        if (sentenceTranslated == null)
        {
            return NoContent();
        }
    }
    else
    { //Takes a random sentence that has not been evaluated by the current
evaluator
        List<Sentence> sentences = await GetEvaluationReadySentenceList(taskId,
parallelCorpusId, evaluationSubtaskId);

        sentenceTranslated =
SentencePicker.GetRandomNotEvaluatedSentence(sentences);
        if (sentenceTranslated == null)
        {
            return NoContent();
        }
    }
}
sentenceSource = await _context.Sentences.Include("CorpusPart")
.Where(x =>
x.NumberInCorpus == sentenceTranslated.NumberInCorpus

```

```

        && x.CorporusPart.ParallelCorpusId ==
sentenceTranslated.CorporusPart.ParallelCorpusId
        && x.CorporusPart.NumberInCorpus == 1
    ).FirstOrDefaultAsync();
    List<Sentence> twoSentences = new() { sentenceSource, sentenceTranslated };
    return twoSentences;
}

```

3. Pielikums

AVMAS Python programmēšanas valodā

```

import json
from comet.models import download_model

import asyncio
import websockets
import sacrebleu

#Downloads and/or loads the COMET evaluation model in 'models/' directory
model = download_model("wmt-large-da-estimator-1719", "models/")
print("Ready!")

def Comet_Calculation(reference_List, system_List, source_List):
    comet_For_All_references = []
    reference_List = list(zip(*reference_List))
    for referenceSentence in range(0, len(reference_List)): #For cycle is for each
reference corpus
        #https://unbabel.github.io/COMET/html/running.html
        data = {"src": source_List, "mt": system_List, "ref":
reference_List[referenceSentence]}
        data = [dict(zip(data, t)) for t in zip(*data.values())]
        _, COMET = model.predict(data, cuda=True)
        comet_For_All_references.append(sum(COMET)/len(COMET))
    return comet_For_All_references

def Sentence_Metrics_Without_Comet(sentence_Id, system, references):
    BLEU = sacrebleu.sentence_bleu(system, references)
    CHRF = sacrebleu.sentence_chrf(system, references)
    TER = sacrebleu.sentence_ter(system, references)
    data = {
        "SentenceId":sentence_Id,
        "BLEU":BLEU.score,
        "CHRF":CHRF.score,
        "TER":TER.score
    }
    return json.dumps(data)

```

```

def Sentence_Metrics_With_Comet(sentence_Id, system, references,
references_For_Comet, system_For_Comet, source_For_Comet):
    BLEU = sacrebleu.sentence_bleu(system, references)
    CHRF = sacrebleu.sentence_chrf(system, references)
    TER = sacrebleu.sentence_ter(system, references)
    comet_For_All_references = Comet_Calculation(references_For_Comet,
system_For_Comet, source_For_Comet)
    data = {
        "SentenceId":sentence_Id,
        "BLEU":BLEU.score,
        "CHRF":CHRF.score,
        "TER":TER.score,
        "COMET":comet_For_All_references
    }
    return json.dumps(data)

def Calculate_Remaining_Sentence_Comets(sentence_Count, references_For_Comet,
system_For_Comet, source_For_Comet):
    response=""
    if(sentence_Count!=0):
        comet_For_All_references = Comet_Calculation(references_For_Comet,
system_For_Comet, source_For_Comet)
        for cometValue in range(0, len(comet_For_All_references)):
            response+= " " + str(comet_For_All_references[cometValue])
    return response

async def echo(websocket, path):
    references=""
    system=""
    sentence_Id=0
    references_For_Comet = []
    system_For_Comet=[]
    source_For_Comet=[]
    sentence_Count=0
    async for message in websocket:
        request = json.loads(message)
        #If the server recieves a message "Done", which indicates that all sentences
        have been sent, it calculates COMET for the remaining sentences
        #and sends a message "Close", which allows to close the connection.
        if (request=="Done"):
            response=Calculate_Remaining_Sentence_Comets(sentence_Count,
references_For_Comet, system_For_Comet, source_For_Comet)
            await websocket.send("Close" + response)
        else:
            #request is a object that has been serialized in JSON format
            sentence_Id = request["SentenceId"]
            source_For_Comet.append(request["Source"])

```

```

system = request["Hypothesis"]
system_For_Comet.append(request["Hypothesis"])
references = request["Target"]
references_For_Comet.append(request["Target"])
sentence_Count+=1
#COMET metric is calculated every 10 sentences because calculating
individual sentences is longer than in batches
#This value can be changed if it is necessary, but shouldnt be changed to
a large value to not hold too many sentences in memory
if(sentence_Count>=1):
    json_data = Sentence_Metrics_With_Comet(sentence_Id, system,
references, references_For_Comet, system_For_Comet, source_For_Comet)
    references_For_Comet = []
    system_For_Comet=[]
    source_For_Comet=[]
    sentence_Count=0
else:
    json_data = Sentence_Metrics_Without_Comet(sentence_Id, system,
references)
    await websocket.send(json_data)

asyncio.get_event_loop().run_until_complete(
    websockets.serve(echo, '0.0.0.0', 8766))
#Keeps the server running
asyncio.get_event_loop().run_forever()

```

Kvalifikācijas darbs „*Tulkošanas kvalitātes vērtēšanas platforma*” izstrādāts Latvijas Universitātes Datorikas fakultātē.

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

Autors: *Kaspars Rumkovskis* _____ .01.2022.

Rekomendēju darbu aizstāvēšanai

Darba vadītājs: *B.dat., Artūrs Stafanovičs* _____ .01.2022.

Recenzents: *Dr.sc.comp, Edgars Diebelis*

Darbs iesniegts 10.01.2022.

Kvalifikācijas darbu pārbaudījumu komisijas sekretārs: _____

Darbs aizstāvēts kvalifikācijas darbu pārbaudījuma komisijas sēdē

____.01.2022. prot. Nr. _____

Komisijas sekretārs(-e): _____