

LATVIJAS UNIVERSITĀTE
EKSAKTO ZINĀTŅU UN TEHNOLOĢIJU FAKULTĀTE

**Atmiņas kartīšu lietotne ar domēna specifiskas
valodas izmantošanu
Kvalifikācijas darbs**

Darba autors:

Jorens Štekeļs, js21283

Darba vadītājs:

prof. Dr. sc. comp. Laila Niedrīte

RĪGA 2025

ANOTĀCIJA

Atmiņas kartītes (angl. flashcards) ir izplatīts veids, kā skolēni, studenti un citi macošies cilvēki iegaumē materiālu. To veidošanai, uzturēšanai un izmantošanai izmanto programmatūras risinājumus, taču lielākoties neizmanto pilnīgi caurspīdīgu vienkārša teksta formātu. Darbā tiek aprakstīts risinājums, kas izmanto domēna specifisku valodu ar pilnīgu informācijas glabāšanu to ietvaros.

Atslēgvārdi: atmiņas kartītes, domēna specifiska valoda, vienkārša teksts, apzīmēšanas valoda.

ABSTRACT

Flashcards are a common way for students to students and other studying people memorize the material. To create them, software solutions are used for maintenance and use, however most do not use a fully transparent plain text format. At work a solution that uses a domain-specific language with a complete is described information storage within them.

Keywords: memory cards, domain specific language, simple text, markup language.

SATURS

APZĪMĒJUMU SARAKSTS	5
IEVADS	6
Nolūks	6
Darbības sfēra	6
Saistība ar citiem dokumentiem	7
Pārskats	7
1. VISPĀRĒJAIS APRAKSTS	8
1.1. Esošā stāvokļa apraksts	8
1.2. Pasūtītājs	8
1.3. Produkta perspektīva	8
1.4. Risinājuma lietotāji	8
1.5. Darījumprasības	9
1.6. Pieņēmumi un atkarības	9
2. PROGRAMMATŪRAS PRASĪBU SPECIFIKĀCIJA	10
2.1. Funkcionālās prasības	10
2.1.1. Augsta līmeņa valodas specifikācija	10
2.1.2. Definētās datu strukturas	11
2.1.3. Funkciju sadalījums moduļos	12
2.1.4. Kopīgas procedūras	13
2.1.5. Atmiņas bāzes modulis	15
2.1.6. Mācīšanās modulis	23
2.1.7. Konfigurācijas modulis	28
2.2. Nefunkcionālās prasības	31
2.2.1. Veikspējas prasības	31
2.2.2. Atribūti	31
3. PROGRAMMATŪRAS PROJEKTĒJUMA APRAKSTS	33
IZMANTOTĀ LITERATŪRA UN AVOTI	34

APZĪMĒJUMU SARAKSTS

API – lietojumprogrammu saskarne (angl. Application Program Interface);

Atzīmēšanas valoda – dokumenta formatēšanas veids, kas iekļauj birkas un citus simbolus, kas definē dokumenta struktūru un izskatu;

Markdown – populāra vienkārša teksta atzīmēšanas valoda, kas ir izmantota dokumentu (parasti tīmekļa) veidošanai;

Transpilēšana – Programmēšanas valodas pārveidošana vienā abstrakcijas līmenī citā valodā ar atšķirīgu abstrakcijas līmeni, kā arī valodas teskta pārveidošana uz kādas programmas objektiem;

Vienkāršais teksts – teksta dati, kas reprezentē simbolus bez grafiskas reprezentācijas vai citiem objektiem;

IEVADS

Nolūks

Šī darba nolūks ir specificēt lietotnes programmatūras prasības un aprakstīt tās projektējumu, kas tiek aprakstīta un izstrādāta kvalifikācijas darba ietvaros.

Darbības sfēra

Atmiņas kartītes (angl. flashcards) ir izplatīts veids, kā skolēni, studenti un citi macošies cilvēki iegaumē materiālu. Šādu kartīšu pamatā parasti ir priekša un aizmugura. Priekšā atrodas jautājums vai uzdevums vai cita informācija, kas uzstāda mentālu vai praktisku uzdevumu, iespējams iekļaujot arī mājienu. Aizmugurā atrodas atbildes teksts, kas ar sevi atbild vai parāda pareizu atbildi uz priekšā uzstādītu jautājumu vai uzdevumu.

Mācīšanās nolūkiem tiek veidotas vairākas atmiņas kartītes uz līdzīgu tēmu, veidojas kartīšu kaudzes. Mācīšanas seansā, tiek vērtēts, cik grūti vai viegli jautājumi vai uzdevumi ir attiecīgi atbildāmi vai atrisināmi. Balstoties uz grūtību vai iespēju atbildēt, kartītes tiek grupētas noteiktās grupās, kas, savukārt, nosaka intervālu, pēc kura kartīte figurēs nākamā mācīšanās seansā. Piemēram, grupas - ļoti viegli, vidēji, grūti, neizdevās -, kam atbilst 5 dienas, 3 dienas, 1 diena, 0 dienas attiecīgi. Jo lielāka ir subjektīva piepūle, atbildot uz jautājumu vai risinot uzdevumu, jo tiek uzturēts garāks pārtraukums pirms nākamās reizes, kad šī kartīte būs izmantota. Šī mācīšanās metode ir atkārtošana ar intervāliem. Mācīšanās metode vai līdzīgas metodes var uzlabot atcerēšanas līmeni pēc mācīšanās (Alison Voice and Arran Stirton, 2020).

Veidojot atmiņas kartītes manuāli tās var pazaudēt, var kļūdīties, sagatavojot to tekstu. Tāpēc ir iespējams kartītes veidot un uzturēt, izmantojot tīmekļa vai lokālo lietotni. Aprakstāmās programmatūras mērķis ir vienkāršot atmiņas kartīšu veidošanu ar iespēju veidot, mainīt un glabāt atmiņas kartītes vienkāršā tekstā un mācīties izmantojot atkārtošana ar intervāliem. Atmiņas kartītes ir glabātas teksta failos un glabājas direktorijā, kas pilnībā paliek lietotāja pārvaldē. Vienkārša teksta kartīšu glabāšana paredz, ka lietotājs var brīvi pārvietot un mainīt kartīšu saturu, turpinot izmantot programmatūras mācīšanās funkcionalitāti.

Uzsvars ir uz pilnīgu atmiņas kartīšanu glabāšanas pilnīgu caurspīdīgumu - kartītes glabājas bez slēptas informācijas. Vienkārša teksta izmantošana bez slēptas informācijas atbrīvo lietotāju no atkarības no šīs programmatūras ar iespēju pārslēgties uz citu risinājumu ar dažām teksta izmaiņām.

Programmatūras produkta nosaukums ir „Mmemory“, kas ir darināts nosaukums no „MD“ (atzīmēšanas valodas Markdown abreviatūra) un „memory“ (angl. atmiņa), kas parāda produkta saistību ar atmiņu.

Saistība ar citiem dokumentiem

Pārskats

Ievadā tiek definēts nolūks darbības sfēra, aprakstīta galvenā programmatūras produkta būtība un mērķi, saistītie dokumenti.

Vispārējā apraksta sadaļā tiek aprakstīts esošais stāvoklis, pāsūtītājs, produkta perspektīva, galvenās darījumprasības, pieņēmumi un atkarības.

Programmatūras prasību specifikācijas sadaļā tiek aprakstīts risinājumā izmantots karītes formāts un ar to saistītas datu struktūras, risinājuma datu plūsmas, funkcijas un risinājuma nefunkcionālās prasības.

1. VISPĀRĒJAIS APRAKSTS

1.1. Esošā stāvokļa apraksts

Esošie atmiņas kartīšu risinājumi galvenokārt atbalsta atmiņas kartīšu importu ar vienkāršu tekstu. Taču lielākā daļa nedod iespēju uzturēt kartītes bez vienkāršā tekstā, jo risinājumi glabā kartītes savā, pielāgotā formātā.

Daži risinājumi iekļauj arī vairākus specializētus jautājumu veidus, kā grupēšanas jautājumi, vairāku atbilžu jautājumus un secības jautājumus. Tādu risinājumu piemērs ir „Quizlet” lietotne. Minētā importa funkcija parasti nav pieejama esošos risinājumos. To importa funkcionalitāte realizē vienkāršu importu visbiežāk ar mērķi migrēt atmiņas kartītes vienas lietotnes ietvaros.

Diezgan tuvs pēc funkcionalitātes mērķiem ir „Emacs” teksta editora papildinājums „org-drill”. Tas piedāvā vienkārša teksta atmiņas kartīšu veidošanu un glabāšanu ar vienkāršu tekstu ar augstu pielāgojamību. Toties tas pieprasa „Emacs” teksta editora izmantošanu un ir domāta tehniskākiem lietotājiem mazāk lietotājiem draudzīgas funkcionalitātes dēļ.

Pašlaik neviens no esošiem risinājumiem neatbalsta minēto specializēto atmiņas kartīšu veidu importu.

1.2. Pasūtītājs

Programmatūras produkts tiek izstrādāts pēc darba autora iniciatīvas kvalifikācijas darba ietvaros.

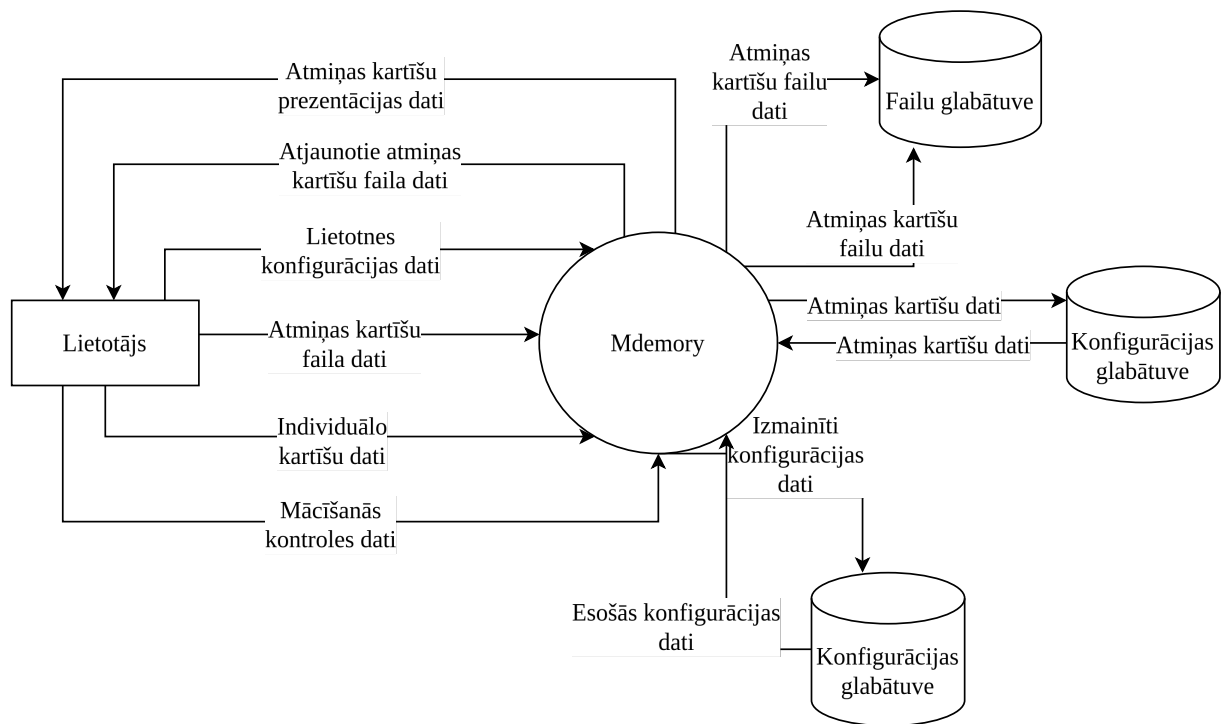
1.3. Produkta perspektīva

Risinājums ir patstāvīga darbvirsmas lietotne. Lietotnes izstrādē ir izmantotas lietotnes saskarnes bibliotēkas un programmēšanas valodas standarta bibliotēka. Risinājums lasīs un izmainīs teksta failus no lietotāja norādītās direktorijas.

1.4. Risinājuma lietotāji

Risinājumam ir viena lietotāju grupa, kam ir pieejamas visas risinājuma iespējas un funkcijas.

Risinājuma augsta līmeņa datu plūsmas ir attēlota 0. līmeņa datu plūsmas diagrammā (DPD; skat. 1.1. attēls). Lietotāju galvenā datu apmaiņa ir atmiņas kartīšu dati un dati par saskarni ar kartītēm.



1.1. att. 0.līmeņa datu plūsmas diagramma

1.5. Darījumprasības

- 1) Kartīšu failu apstrāde, izmantojot definēto domēna specifikas valodu;
- 2) Kartīšu saraksta veidošana;
- 3) Definēto jautājumu veidu tekstuāla atspoguļošana;
- 4) Definēto jautājumu veidu interaktīva mācīšanas funkcija;
- 5) Mācīšanās ar atkārtotanu ar intervāliem;
- 6) Kartīšu pievienošana, modifikācija un dzēšana sarakstā;
- 7) Vairāku kartīšu failu apstrāde direktorijā;
- 8) Kartīšu saglabāšana no saraksta.

1.6. Pieņēmumi un atkarības

- 1) Lietotnes var tikt uzbūvēta darbam uz Windows vai Linux sistēmām, uz kurām ir pieejamas risinājumā izmantotas saskarnes bibliotēkas;
- 2) Lietotājs atbild par atmiņas kartīšu failu glabāšanu;

2. PROGRAMMATŪRAS PRASĪBU SPECIFIKĀCIJA

2.1. Funkcionālās prasības

2.1.1. Augsta līmeņa valodas specifikācija

Kartišu veidošanai vienkāršā tekstā tiek izmantota vienkārša domēna specifiska valoda. Ir 4 jautājumu veidi, kuru struktūra ir līdzīga ar dažām atšķirībām. Katrs jautājums sākas ar priekšu, kam var būt skaitlis, kas apzīmē intervālu stundās, pēc kura tas būs atkārtoti izmantots.

- Ka sauc Latvijas galvaspilsētu? >

[...]

- [12.5] Ka sauc Igaunijas galvaspilsētu? >

[...]

Vienkāršs atbilžu jautājums ir jautājums, kam ir viena vai vairākas pareizas atbildes, katra atbilde sākas ar „-“.

- Ka sauc Lietuvas galvas pilsētu? >

- Viļņus

- Nosauciet Baltijas valstis >

- Latvija

- Igaunija

- Lietuva

Izvēles jautājums ir jautājums, kam ir viens vai vairākas pareizas atbildes, pareizās atbildes sākas ar „+“, nepareizās – ar „-“.

- Kuras no valstīm ir Baltijas valstis >

+ Latvija

- Rumānija

+ Lietuva

- Spānija

Secības jautājums.

- Sakārtojiet zinātniskās metodes soļus pareizā secībā >

-^ Uzdod jautājumu

-^ Izvirza hipotēzi

-^ Veic eksperimentu

-^ Analizē datus

2.1.1.1. Grupēšanas jautājums

- Savienojiet planētas ar to īpašībām >

- Zeme:

- Satur dzīvi

- Jupiters:

- Lielākā planēta

- Ir gredzeni

- Venēra:

- Merkurs:

- Mazākā

- Saturns:

- Ir gredzeni

- Neptūns:

- Tālākā no saules

- Ir gredzeni

2.1.2. Definētās datu struktūras

2.1.2.1. Jautājumi

Šajā apakšsadaļā ir aprakstītas datu struktūras, kas tiks izmantotas, aprakstot risinājuma funkcijas.

Jautājumu datu struktūra satur informāciju par atmiņas kartīti ar papildus informāciju, kas atbilst noteiktam jautājuma veidam.

Pamatā jautājumam ir:

- Teksts – simbola virkne;
- Ilgums, cik kartīte nebūs izmantota atkāļ intervālu metodē – pozitīvs daļu skaitlis;

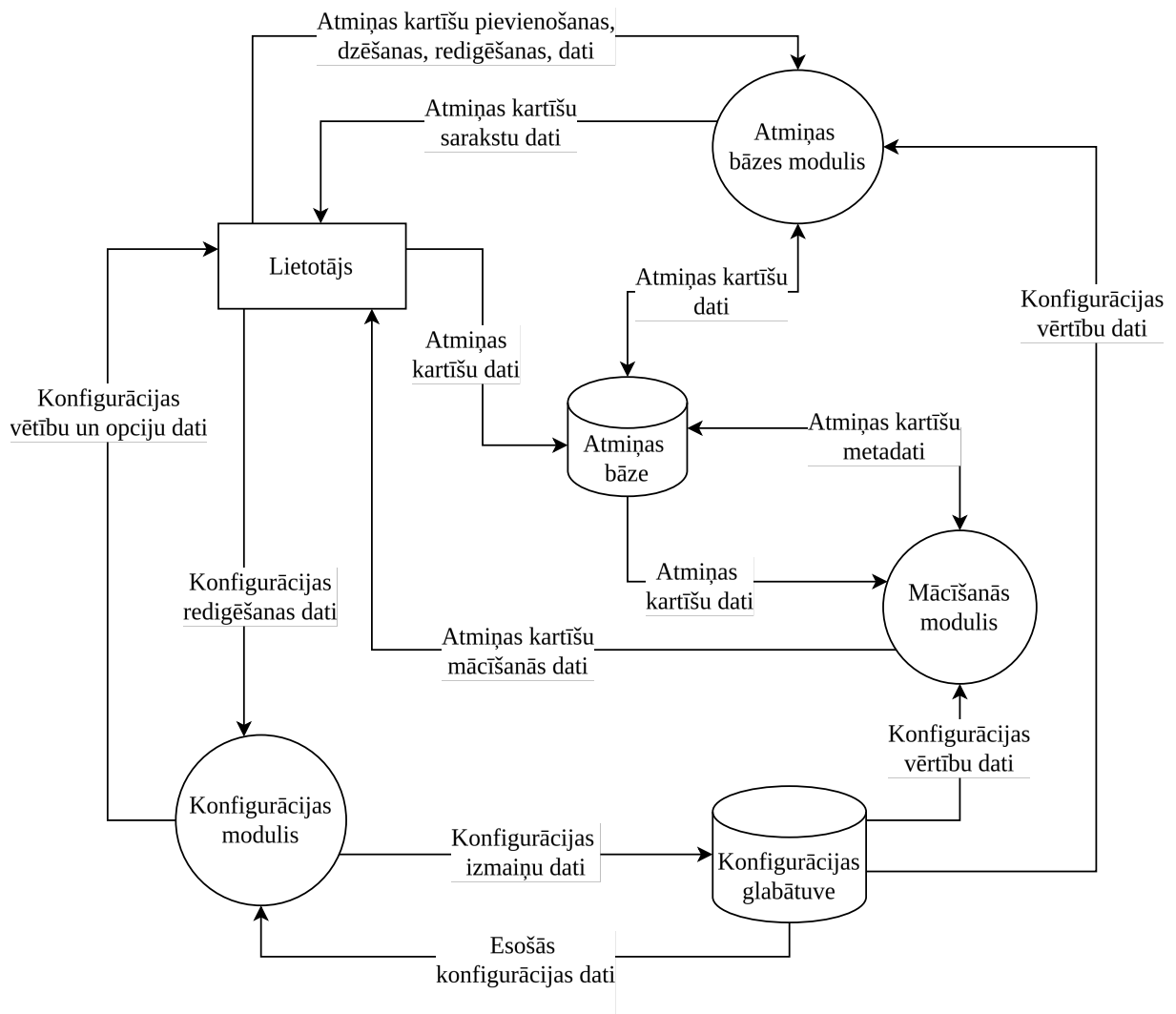
Papildus informācija iedalās vairākos veidos.

1) Vairāku elementu veids:

- Veids – uzskaitījums, kurš no - vairāku variantu, secības, vienkāršs - jautājums ir;
- Saraksts ar objektiem, kas sastāv no simbolu virknes un korektuma karodziņu;

2) Grupas veids:

- Saraksts ar objektiem, kas sastāv no:
 - Nosaukuma – simbolu virkne;
 - Elementiem – saraksts ar simbolu virknēm.



2.1. att. 1. līmeņa datu plūsmas diagramma

Lappuse ir ir saraksts ar referencēm uz vairākiem jautājuma objektiem, kam ir noteikts lappuses skaits.

Datumam un laikam tiek izmantots sekojošs formāts - .. : -, piemēram, „13.05.2024 12:02“.

2.1.3. Funkciju sadalījums moduļos

Risinājuma moduļi ar to saistītām datu plūsmām un datu glabātuvēm ir attēlots 1. līmeņa DPD (skat. 2.1. attēls). Funkciju sadalījums moduļos ir apraksts sadalījuma tabulā (skat. 2.1. tabula).

2.1. tabula Funkciju sadalījums pa moduļiem

Modulis	Funkcija	Identifikators
Atmiņas bāzes modulis	Kartīšu faila ielāde	AB01
	Kartīšu saraksta lappušu izveidošana	AB02
	Atmiņas bāzes lappuses iegūšana	AB03
	Kartīšu pievienošana	AB04
	Kartītes pirmteksta iegūšana	AB05
	Kartītes redigēšana	AB06
	Kartītes dzēšana	AB07
	Kartītes faila saglabāšana	AB08
Mācīšanās modulis	Nākamās kartītes noteikšana	MA01
	Izvēles kartītes pārbaude	MA02
	Kārtošanas kartītes pārbaude	MA03
	Grupēšanas kartītes pārbaude	MA04
	Intervāla grupas atbildes sniegšana	MA05
Konfigurācijas modulis	Konfigurācijas vērtības iegūšana	KF01
	Konfigurācijas vērtību iegūšana	KF02
	Konfigurācijas vērtības izmaiņa	KF03

2.1.4. Kopīgas procedūras

Vairākas prasībās definētās funkcijas izmanto kopīgas procedūras, kas tiek definētas tabulās:

PR01, PR02.

2.2. tabula Teksta jautājumu transpilēšana

Procedūras nosaukums
Teksta jautājumu transpilēšana
Procedūras identifikators
PR01
Procedūras apraksts
Procedūra apstrādā simbolu virkni, iegūstot jautājumu datu struktūru objektus. Ir sagaidāms, ka simbolu virkne atbilst specifificētai valodai. Ja simbolu virkne valodai neatbilst, tiek izvadīta kļūda. Simbolu virkne tiek apstrādāta vairākos posmos.
Ievade
1) Simbolu virkne;
Apstrāde
1) Tiek nolasīti simboli, sagrupējot to tekstvienībās;

2) Tiek veikta virspusēja gramatikas pārbaude; a) Ja tiek sastapta tekstvienība, kas nav sagaidāma kādā kontekstā, beidz apstrādi, izvadē pievienojot kļūdu ar tās pozīciju; 3) Tiek nolasīti jautājumi, nosakot to veidu pēc to struktūras: a) Jautājuma sākums (jeb kartītes priekša); b) Veidojot jautājumus tiek pielasīts arī intervāla laiks, ja tāds ir; c) Pēc jautājuma sākuma ir viens vai vairākas atbildes jeb apakšpunkti; i) Vairāki apakšpunkti ar „-“ – atbilžu jautājums; ii) Vairāki apakšpunkti ar „+“ – izvēles jautājums; iii) Vairāki apakšpunkti ar „-^“ – izvēles jautājums; iv) Vairāki apakšpunkti, kas beidzas ar „:“, kam seko apakšpunkti ar „-“ (iespējams tos nenorādīt); d) Jautājumu veidi ir nosakāmi bez tiešas vieda norādīšanas; e) Ja nolasīšanas laikā parādās nesagaidīta tekstvienība, beidz apstrādi, izvadē pievienojot kļūdu ar tās pozīciju;
Izvade
1) Saraksts ar jautājumiem; 2) Kļūda ar tekstu un tās pozīciju saturā;

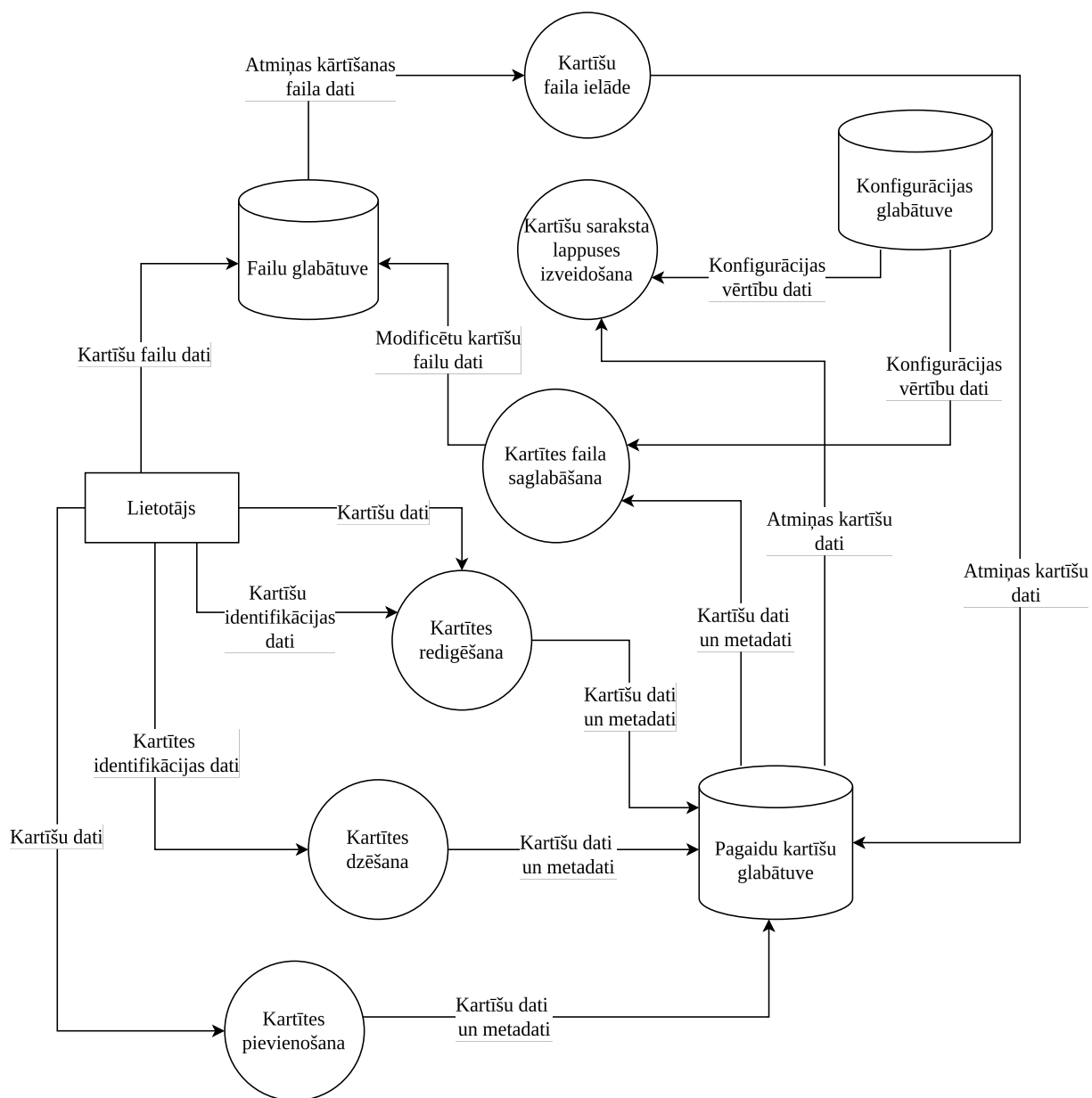
2.3. tabula Teksta jautājumu detranspilēšana

Procedūras nosaukums
Teksta jautājumu detranspilēšana
Procedūras identifikators
PR02
Procedūras apraksts
Procedūra apstrādā jautājumu sarakstu un izveido tam atbilstošu simbolu virkni, kas atbilst specificētai valodai. Papildus sākumā tiek pievienots datums un laiks, ja tas tiek padots, kas ir daļa no metadatiem.

Ievade
1) Jautājumu saraksts; 2) Datums un laiks;
Apstrāde
1) Iegūst konfigurācijas vērtības, kas nosaka cik garš ir tabulācijas simbols un kāds ir maksimāls rindiņas platums pirms tiek veidots jaunās rindas simbols; 1) Ja datums un laiks ir padots, pievieno datumu un laiku atbilstoši formātam; 2) Katram jautājumam veic sekojošo: a) Pievieno jautājuma sākumu (jeb priekšu) ar intervāla skaitli, ja tāds ir; b) Pievieno apakšpunktus atbilstoši tā veidam; i) Vairāki apakšpunkti ar „-“ – atbilžu jautājums; ii) Vairāki apakšpunkti ar „+“ – izvēles jautājums; iii) Vairāki apakšpunkti ar „-^“ – izvēles jautājums; iv) Vairāki apakšpunkti, kas beidzas ar „:“, kam seko apakšpunkti ar „-“; c) Veidojot jautājumus ievēro iegūtās konfigurācijas vērtības;
Izvade
1) Simbolu virkne;

2.1.5. Atmiņas bāzes modulis

Atmiņas bāze ir direktorijs ar atmiņas kartīšu failiem. Modulis atbild par mijiedarbību ar atmiņas šiem failiem un atmiņas kartīšu prezentēšanu. Šī moduļa funkcijas ir izmantotas atmiņu kartīšu saraksta funkcionalitātei. Funkcijas mijiedarbojas ar pagaidu glabātuvē esošām atmiņas kartītēm un failu glabātuves failiem, kas satur atmiņu kartīšu informāciju. Ar funkcijām saistītas datuplūsmas ir attēlotas moduļa 2. līmeņa DPD (skat. 2.2. attēlu). Moduļa funkcijas tiek definētas tabulās: AB01, AB02, AB03, AB04, AB05, AB06, AB07, AB08.



2.2. att. Atmiņas bāzes moduļa 2.līmeņa datu plūsmas diagramma

Funkcijas nosaukums
Kartīšu faila ielāde
Funkcijas identifikators
AB01
Apraksts
Funkcija apstrādā faila saturu. Iegūstot objektus ar jautājumu dati un ar to saistītiem metadatiem. Ieejas failam jāatbilst valodas specifikācijai.
Ievade
1) Ievades faila ceļš ar noteiktu paplašinājumu;
Apstrāde
1) Faila saturs tiek ielādēts atmiņā; 2) Ja faila ielāde ir neveiksmīga, parāda 1. paziņojumu. 3) Tiek nolasīts datums un laiks no faila sākuma, ja tāds eksistē: a) Ja datuma un laika formāts nav korekts, parāda 2. paziņojumu ar atbilstošu iemeslu un kļūdas pazīmi, beidz apstrādi. 4) Tiek iegūti jautājuma objekti, izmantojot procedūru PR01; a) Ja radās kļūda, parāda 2. paziņojumu ar kļūdas informāciju, beidz apstrādi. 5) Ja kādā no apstrādes posmiem radās kļūda, parāda 2. paziņojumu ar atbilstošu iemeslu un kļūdas pozīciju, beidz apstrādi.
Izvade
1) Saraksts ar jautājumiem; 2) Faila reģistrēts datums un laiks;
Paziņojumi
1) Ielādes kļūda: neizdevās atvert failu; 2) Ielādes kļūda: [kļūdas iemesls] [kļūdas rinda]:[kļūdas kolonna];

2.5. tabula **Kartīšu saraksta lappušu izveidošana**

Funkcijas nosaukums
Kartīšu saraksta lappušu izveidošana
Funkcijas identifikators
AB02
Apraksts
Funkcija izveido vairākas lappuses, sadalot atmiņā esošus jautājumu objektus intervālos. Lappušu izmērs ir noteikts ar konfigurācijas parametrus.
Ievade
1) Jautājumu objektu saraksts;
Apstrāde
1) No konfigurācijas tiek iegūts jautājumu skaits lappusē, un rādāmo lappušu pogu skaits; 2) Jautājumi tiek sadalīti lappusēs, veidojot lappušu objektus, katrā tiek ievietotas references uz jautājumiem. Lappušu pogu skaits atbilst konfigurācijas vērtībai;
Izvade
1) Lappušu objektu saraksts;
Paziņojumi
Funkcijai nav paziņojumu.

2.6. tabula **Atmiņas bāzes lappuses iegūšana**

Funkcijas nosaukums
Atmiņas bāzes lappuses iegūšana
Funkcijas identifikators
AB03
Apraksts
Funkcija iegūst jautājumu objektu intervālu no pagaidu glabātuves.
Ievade
1) Lappuses numurs;

Apstrāde
1) No pagaides atmiņas kartīšu glabātuves tiek iegūtas atmiņas kartīšu lappuses; 2) Tiek meklēts lappušu objekts ar norādīto numuru; a) Ja lappuse tiek atrasta, saistītie atmiņas kartīšu objekti tiek atgriezti; b) Ja tāda lappuse netiek atrasta, parāda 1. paziņojumu; 3)
Izvade
1) Saraksts ar lappuses atmiņas kartīšu objektiem;
Paziņojumi
1) Pieprasītā lappuse neeksistē;

2.7. tabula **Kartīšu pievienošana**

Funkcijas nosaukums
Kartīšu pievienošana
Funkcijas identifikators
AB04
Apraksts
Funkcija pieņem lietotāja ievadīto tekstu un pārveido to jautājumu objektos, kas pievieno tos jau esošiem jautājumiem.
Ievade
1) Simbolu virkne;
Apstrāde
1) Ja ievadīts teksts, parāda 1. paziņojumu, beidz apstrādi; 2) Apstrādā simbolu virkni, izmantojot PR01; a) Ja tiek izvadīta kļūda, parāda 2. paziņojumu ar kļūdas informāciju, beidz apstrādi;
Izvade
Izvades jautājuma objekti tiek pievienoti jautājumu pagaidu glabātvē. Saskarnē parādās indikators, ka atmiņas kartīšu fails tika izmainīts. 1) Lappušu objektu saraksts;

Paziņojumi
1) Nav ievadīts neviens jautājums;
2) Ielādes kļūda: [kļūdas iemesls] [kļūdas rinda]:[kļūdas kolonna];

2.8. tabula **Kartītes pirmteksta iegūšana**

Funkcijas nosaukums
Kartītes pirmteksta iegūšana
Funkcijas identifikators
AB05
Apraksts
Funkcija iegūst kartītes pirmtekstu, kas atbilst valodas specifikācijai. Funkcija ir specializēts procedūras izmantošanas gadījums, kurā tiek detranspilēts tikai viens jautājums. Funkcija ir izmantota redigēšanas iespējai.
Ievade
1) Jautājuma objekts;
Apstrāde
1) Izmantojot procedūru PR01, jautājums tiek detranspilēts pirmtekstā;
Izvade
1) Simbolu virkne – jautājuma pirmteskts;
Paziņojumi
Funkcijai nav paziņojumu. Saskarnē parādās indikators, ka atmiņas kartīšu fails tika izmainīts.

2.9. tabula **Kartītes redigēšana**

Funkcijas nosaukums
Kartītes redigēšana
Funkcijas identifikators

AB06
Apraksts
Funkcija izmaina esošo jautājumu, pēc to pirmteksta, kas atbilst valodas specifikācijai.
Ievade
1) Jautājuma objekts; 2) Jautājuma pirmteksts;
Apstrāde
1) Ja ievades teksts ir tukšs, parāda 1. paziņojumu, beidz apstrādi; 2) Izmantojot PR01, tiek iegūts jautājuma objekts no sniegtā pirmteksta; 1) Ja tiek izvadīta kļūda, parāda 1. paziņojumu ar atbilstošu kļūdas informāciju, beidz apstrādi; 2) Padotais jautājuma objekts tiek atjaunots ar iegūta objekta datiem;
Izvade
1) Jautājuma objekts;
Paziņojumi
Izvades jautājuma objekts tiek atjaunots atbilstošam ierakstu jautājumu pagaidu glabātuvē. Saskarnē parādās indikators, ka atmiņas kartīšu fails tika izmainīts. 1) Nav ievadīts neviens jautājums; 2) Redigēšanas kļūda: [kļūdas iemesls] [kļūdas rinda]:[kļūdas kolonna];

2.10. tabula Kartītes dzēšana

Funkcijas nosaukums
Kartītes dzēšana
Funkcijas identifikators
AB07
Apraksts
Funkcija izdēš kartīti no pagaidu glabātuves.
Ievade

1) Jautājuma reference;
Apstrāde
1) Jautājums tiek sameklēts un izdzēsts;
Izvade
Funkcijai nav izvades.
Paziņojumi
Funkcijai nav paziņojumu.

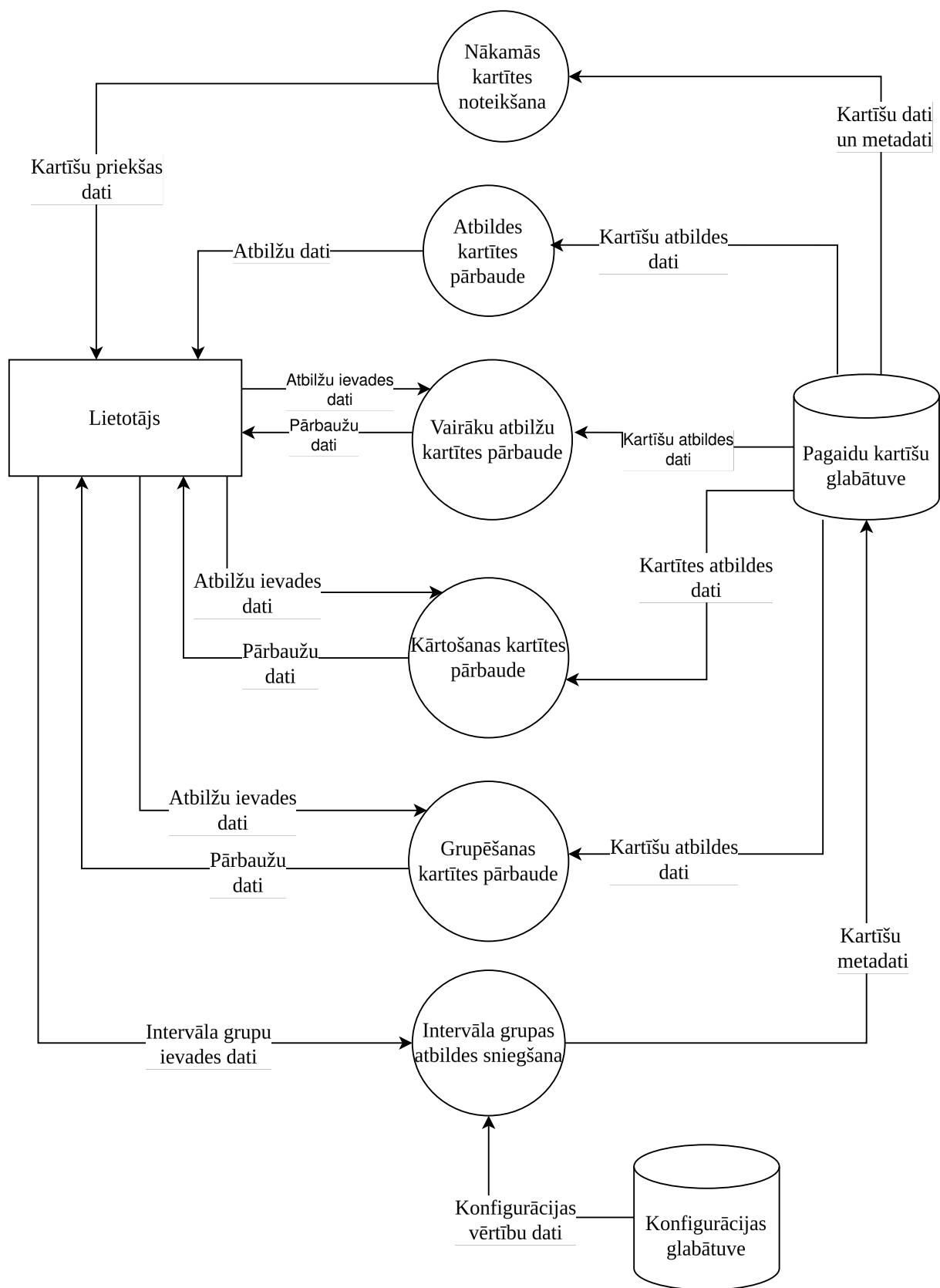
2.11. tabula **Kartītes faila saglabāšana**

Funkcijas nosaukums
Kartītes faila saglabāšana
Funkcijas identifikators
AB08
Apraksts
Funkcija saglabā kartīšu jautājumus un metadatus tam atbilstošā failā, pārveidojot tos valodai atbilstošā formātā.
Ievade
1) Faila atrašanās ceļš; 2) Jautājumu saraksts; 3) Datums un laiks;
Apstrāde
1) Iegūst pirmtekstu padotajiem objektiem, izmantojot procedūru PR02; 2) Ja tiek izvadīta kļūda, parāda 2. paziņojumu ar kļūdas informāciju, beidz apstrādi; 3) Izveido vai atver failu padotā ceļā; a) Ja atveršana vai izveidošana neizdevās, parāda 1. paziņojumu; 4) Pievienot iegūto saturu failam un saglabā to;
Izvade
Fails tiks atjaunots failu sistēmā. Saskarnē pazūd indikators, ka atmiņas kartīšu fails tika izmainīts.

Paziņojumi
1) Neizdevās atvērt vai izveidot atmiņas kartīšu failu; 2) Redigēšanas kļūda: [kļūdas iemesls] [kļūdas rinda]:[kļūdas kolonna];

2.1.6. Mācīšanās modulis

Mācīšanās risinājuma kontekstā ir kartīšu prezentēšana ar interaktīvu atbildes funkcionalitāti, atbilstoši atmiņas kartītes tipam. Šī moduļa funkcijas ir izmantotas mācīšanās loga funkcionalitātei. Funkcijas ietver darbības ar pagaidu glabātnē esošām atmiņas kartītēm. Ar funkcijām saistītas datuplūsmas ir attēlotas moduļa 2. līmeņa DPD (skat. 2.3. attēlu). Moduļa funkcijas tiek definētas tabulās: MA01, MA02, MA03, MA04, MA05.



2.3. att. Mācīšanās moduļa 2.līmeņa datu plūsmas diagramma

Funkcijas nosaukums
Nākamās kartītes noteikšana
Funkcijas identifikators
MA01
Apraksts
Funkcija nosaka nākamo kartīti, kas tiks parādīta lietotājam mācīšanas procesā. Kartīšu secību nosaka mācīšanās algoritms.
Ievade
1) Pašreizējais kartīte – kartītes objekts; 2) Algoritms – viens no vertībām: „fizisks“, „nejaušs“, „intervālu“;
Apstrāde
1) Ja pagaidu kartīšu glabāvē nav nevienas kartītes, beidz apstrādi, neatgriežot kartīti; 1) Ja padotais algoritms ir „fizisks“; 2) Ja pašreizējā kartīte ir padota, izvēlās nākamo kartīti starp pagaidu kartīšu glabātuves; 3) Citādi izvēlas pirmo kartīti no pagaidu kartīšu glabātuves; 4) Ja padotais algoritms ir „nejaušs“; a) Izvēlas nākamo kartīti nejaušā kārtībā starp pagaidu kartīšu glabātuves; 5) Ja padotais algoritms ir „intervālu“; a) No pagaidu kartīšu glabātuves izvēlas tikai to, kur nav norādīts intervāls vai intervāls ir 0; b) No šī saraksta izvēlas nākamo kartīti nejaušā kārtībā;
Izvade
1) Nākamās kartītes objekts (iespējams nulles objekts);
Paziņojumi
Funkcijai nav paziņojumu.

Funkcijas nosaukums
Izvēles kartītes pārbaude
Funkcijas identifikators
MA02
Apraksts
Veic jautājuma atbildes pārbaudi izvēles kartītes jautājuma veidam.
Ievade
1) Izvēlēto atbilžu simbolu virkņu saraksts ar pazīmi, vai atbilde ir izvēlēta;
2) Izvēles jautājuma objekts;
Apstrāde
1) Izskata abus sarakstus paralēli;
2) Ja atbildes un jautājuma objektu izvēles elementa pazīmes sakrīt (piemēram, „pareizs“ un „pareizs“);
a) Atbilžu elementam uzstāda pareizuma pazīmi kā „pareizs“;
b) Citādi uzstāda atbildes pareizuma atbildi kā „nepareizs“;
Izvade
1) Izvēlēto atbilžu simbolu virkņu saraksts ar pareizuma karodziņu vērtībām;
Paziņojumi
Funkcijai nav paziņojumu.

2.14. tabula Kārtošanas kartītes pārbaude

Funkcijas nosaukums
Kārtošanas kartītes pārbaude
Funkcijas identifikators
MA03
Apraksts
Veic jautājuma atbildes pārbaudi kārtošanas kartītes jautājuma veidam.
Ievade
1) Atbilžu simbolu virkņu saraksts;

2) Jautājuma objekts;
Apstrāde
1) Izskata abus sarakstus paralēli; 2) Ja atbildes un jautājuma objektu izvēles elementa sakrīt, atbilžu elementam uzstāda pareizuma pazīmi kā „pareizs“; 3) Citādi uzstāda atbildes pareizuma atbildi kā „nepareizs“;
Izvade
1) Atbilžu simbolu virkņu saraksts ar pareizuma karodziņu vērtībām;
Paziņojumi
Funkcijai nav paziņojumu.

2.15. tabula Grupēšanas kartītes pārbaude

Funkcijas nosaukums
Grupēšanas kartītes pārbaude
Funkcijas identifikators
MA04
Apraksts
Veic jautājuma atbildes pārbaudi grupēšanas kartītes jautājuma veidam.
Ievade
1) Negrupēto elementu simbolu virkņu saraksts; 2) Grupētu simbolu virkņu sarakstu saraksts; 3) Jautājuma objekts;
Apstrāde
1) Katram nesagrupētam elementam uzstāda pareizuma pazīmi uz „nepareizs“; 2) Katru grupai atbilžu un jautājuma objekta saraksta grupu izskata paralēli; a) Elementiem, kas sakrīt abos sarakstos, atbilžu sarakstā uzstāda pareizuma pazīmi uz „pareizs“; b) Elementiem, kas nav jautājuma atbildes grupā pareizuma pazīmi uz „nepareizs“;
Izvade

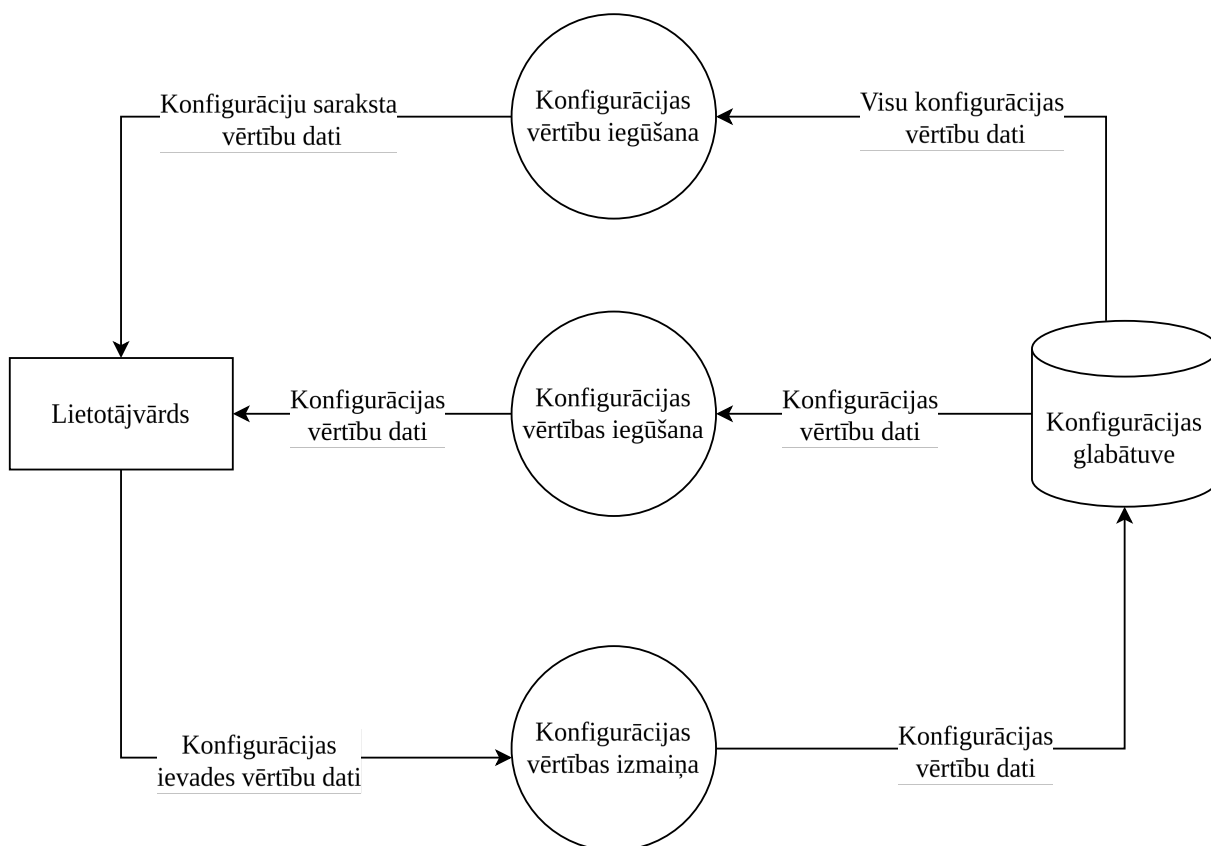
1) Negrupēto elementu simbolu virkņu saraksts ar pareizuma karodziņu vērtībām;
2) Grupētu simbolu virkņu sarakstu saraksts ar pareizuma karodziņu vērtībām;
Paziņojumi
Funkcijai nav paziņojumu.

2.16. tabula Intervāla grupas atbildes sniegšana

Funkcijas nosaukums
Intervāla grupas atbildes sniegšana
Funkcijas identifikators
MA05
Apraksts
Izmaina jautājuma objektu atbilstoši izvēlētam grūtības vai izpildījuma vērējumam veicot kartītes atbildi. Atbilstošie intervāli tiek iegūti no konfigurācijas.
Ievade
1) Intervāla atbilde – viens no: „nav atbildēts“, „grūti“, „vidēji“, „viegli“, „ļoti viegli“; a) Jautājuma objekts;
Apstrāde
1) No konfigurācijas glabātuves tiek iegūts intervāls, kas atbilst sniegtai atbildei; 2) Jautājumam tiek uzstādīts intervāls;
Izvade
1) Izmainīts jautājuma objekts;
Paziņojumi
Funkcijai nav paziņojumu.

2.1.7. Konfigurācijas modulis

Konfigurācijas modulis iekļauj parametrus, ko var personalizēt saistībā ar mācīšanās procesu, prezentāciju uc. Šī moduļa funkcijas ir izmantotas konfigurējot vērtības, kas tiek glabātas konfigurācijas failā un tiek izmantotas risinājuma funkcionalitātē, kur tās aspektiem piemīt



2.4. att. Konfigurācijas moduļa 2.līmeņa datu plūsmas diagramma

konfigurējamas vērtības. Ar funkcijām saistītas datuplūsmas ir attēlotas moduļa 2. līmeņa DPD (skat. 2.4. attēlu). Moduļa funkcijas tiek definētas tabulās: KF01, KF02, KF03.

2.17. tabula Konfigurācijas vērtības iegūšana

Funkcijas nosaukums
Konfigurācijas vērtības iegūšana
Funkcijas identifikators
KF01
Apraksts
Funkcijas iegūst vienu definēto konfigurācijas vērtību.
Ievade
1) Vērtības atslēga – simbolu virkne;
Apstrāde
1) Ja konfigurācija vēl nebija ielādēta, tiek ielādēts konfigurācijas fails;
2) Ielādētā konfigurācijā tiek sameklēts konfigurācijas vērtības ieraksts;
Izvade

1) Konfigurācijas vērtība – viens no: skaitlis, simbolu virkne, karodziņš;
Paziņojumi
1) Konfigurācijas fails nav korekts, nevar nolasīt konfigurāciju;

2.18. tabula Konfigurācijas vērtību iegūšana

Funkcijas nosaukums
Konfigurācijas vērtību iegūšana
Funkcijas identifikators
KF02
Apraksts
Funkcija iegūst visas definētās konfigurācijas vērtības.
Ievade
1) Vērtības atslēga – simbolu virkne;
Apstrāde
1) Ja konfigurācija vēl nebija ielādēta, tiek ielādēts konfigurācijas fails;
2) Ielādētā konfigurācijā tiek iegūtas vērtības;
Izvade
1) Konfigurācijas vērtību saraksts – viens no: skaitlis, simbolu virkne, karodziņš;
Paziņojumi
1) Konfigurācijas fails nav korekts, nevar nolasīt konfigurāciju;

2.19. tabula Konfigurācijas vērtības izmaiņa

Funkcijas nosaukums
Konfigurācijas vērtības izmaiņa
Funkcijas identifikators
KF03
Apraksts
Funkcija izmaina vērtību no definētām konfigurācijas vērtībām.
Ievade

1) Vērtības atslēga – simbolu virkne; a) Konfigurācijas vērtība – viens no: skaitlis, simbolu virkne, karodziņš;
Apstrāde
1) Ja konfigurācija vēl nebija ielādēta, tiek ielādēts konfigurācijas fails; a) Konfigurācijas vērtība ar ievadīto atslēgu tiek izmainīta uz ievadīto konfigurācijas vērtību;
Izvade
1) Izmainīta konfigurācijas faila saturs;
Paziņojumi
1) Konfigurācijas fails nav korekts, nevar rediģēt konfigurāciju;

2.2. Nefunkcionālās prasības

Šajā sadaļā tiks apskatītas risinājuma nefunkcionālās prasības.

2.2.1. Veikspējas prasības

Uz dinamisko veikspēju ir sekojošas prasības:

- 1) Failu transpilēšanas ātrums ir vismaz 50000 rindas sekundē;
- 2) Failu detranspilēšanas (saglabāšanas) ātrums ir vismaz 50000 rindas sekundē;

2.2.2. Atribūti

2.2.2.1. Lietojamība

Uz risinājuma lietojamību ir sekojošas prasības:

- 1) Visiem datiem jābūt maināmiem un redzamiem no vienkārša teksta failiem;
- 2) Bieži izmantojamām darbībām jābūt atbilstošiem tastatūras taustiņu īsceļiem;

2.2.2.2. Pirmkoda īpašības un uzturamība

Pirmkodam jābūt izstrādātam ar sekojošiem nosacījumiem:

- 1) Risinājuma pirmkods ir realizēts galvenokārt procedurālā stilā;
- 2) Transpilatora funkcionalitātes testu pārklājums ir 90%;
- 3) Saskarnes funkcionalitāte ir notestēta svarīgākās vietās (svarīgākās vietas ir subjektīvs jēdziens, tāpēc to definēsim kā 20%).

2.2.2.3. Pārnesamība

Uz pārnesamību risinājumam ir sekojoša prasība: kartīšu failiem ir jābūt pārnesamiem pārkopējot visu direktoriju uz citu mašīnu. Nosacījums saglabā atgriezinisko savietojamību vienas galvenās versijas ietvaros.

2.2.2.4. Drošība

Uz drošību risinājumam ir sekojošas prasības:

- 1) Risinājuma pirmkods ir iekļauts kopā ar izpildāmo failu;
- 2) Risinājums nemodificē un nelasa failus, izņemot izmantotās bibliotēkas, failu glabātuvi, standarta konfigurācijas vietu;

3. PROGRAMMATŪRAS PROJEKTĒJUMA APRAKSTS

IZMANTOTĀ LITERATŪRA UN AVOTI

- 1) Alison Voice and Arran Stirton Spaced Repetition: towards more effective learning in STEM (2020). Pieejams: https://web.archive.org/web/20171019211402/https://www.loc.gov/standards/datetime/ISO_DIS%208601-1.pdf aplūkots 12.10.2024