

LATVIJAS UNIVERSITĀTE
DATORIKAS FAKULTĀTE

**PLĀNOŠANAS UN LAIKA PĀRVALDĪBAS TĪMEKĻA
LIETOJUMPROGRAMMA ORGANIZĀCIJAS DARBĪBAI UN
PERSONĪGAI LIETOŠANAI**

KVALIFIKĀCIJAS DARBS

Autors: **Roberts Gordins**

Studenta apliecības Nr.: rg19031

Darba vadītājs: Dr. dat. Maksims Kravcevs

RĪGA 2022

ANOTĀCIJA

Darba *Plānošanas un laika pārvaldības tīmekļa lietojumprogramma organizācijas darbība un personīgai lietošanai* ietvaros tika izstrādāta sistēma, kurā ir iespējams pārvaldīt organizācijas izstrādes procesus un noteiktam organizācijas dalībniekam uzskaitīt laiku, kas tika atvēlēts sistēmā reģistrētam uzdevumam. Tīmekļa lietojumprogrammu ir iespējams izmantot personīgā nolūkā, veidojot savu organizāciju un laika uzskaiti uz izveidotiem uzdevumiem. Sistēmas vispārīgs nolūks ir atvieglot organizācijas komandas vai organizācijas dalībnieka izstrādes un produktivitātes analīzi, kuru veic organizācijas īpašnieks vai komandas vadītājs.

Atslēgvārdi: laika pārvaldība, tīmekļa lietojumprogramma, Sapper, Svelte, ASP.NET API

ABSTRACT

SCHEDULING AND TIME MANAGEMENT WEB-BASED APPLICATION FOR ORGANIZATION'S OPERATIONS AND PERSONAL USE

Within the framework of the work *Scheduling and time management web-based application for organization's operations and personal use*, a system was developed in which organization's development process can be managed and organization member's time that was spent on task completion can be registered. The web application can be used for personal purposes by creating your own organization and time records for registered tasks. The overall purpose of the system is to ease the development and productivity analysis of the organization's team or member by the organization owner or team leader.

Keywords: Time Management, Single Page Application, Sapper, Svelte, ASP.NET API

SATURS

APZĪMĒJUMU SARAKSTS	5
IEVADS	7
Nolūks.....	7
Darbības sfēra	7
Saisība ar citiem dokumentiem	7
Pārskats	7
1. VISPĀRĪGS APRAKSTS	9
1.1. Esošā stāvokļa apraksts	9
1.2. Pasūtītājs.....	9
1.3. Produkta perspektīvas.....	9
1.4. Darījumprasības.....	9
1.5. Sistēmas lietotāji.....	10
1.6. Vispārējie ierobežojumi.....	13
1.7. Pieņēmumi un atkarības.....	13
2. PROGRAMMATŪRAS PRASĪBU SPECIFIKĀCIJA	15
2.1. Konceptuālais datu bāzes apraksts	15
2.2. Funkcionālās prasības.....	18
2.2.1. Vispārējās nodaļas, kas saistītas ar funkciju aprakstīšanu.....	18
2.2.2. Funkciju sadalījums pa moduļiem.....	20
2.2.3. Organizācijas struktūras veidošanas modulis	23
2.2.4. Izstrādes procesa vadības modulis.....	35
2.2.5. Sazināšanas modulis	49
2.2.6. Izstrādes procesa attēlošanas modulis	56
2.3. Nefunkcionālās prasības	61
2.3.1. Veiktspējas prasības	61
2.3.2. Lietojamības prasības	61
2.3.3. Drošības prasības	62
2.3.4. Pieejamības prasības.....	63
2.3.5. Uzturēšanas prasības.....	63
3. PROGRAMMATŪRAS PROJEKTĒJUMA APRAKSTS.....	64
3.1. Datu bāzes projektējums.....	64
3.1.1. Loģiskais datu bāzes apraksts.....	64
3.1.2. Fiziskais datu bāzes apraksts	66
3.2. Daļējs funkciju projektējums.....	79
3.2.1. Programmatūras izvietojuma diagramma.....	79

3.2.2. Uzdevuma izpildes procesa aktivitāšu diagramma.....	81
3.2.3. Lomas uzdevuma izveidošanas secību diagramma	83
3.2.4. Izstrādes procesa uzdevumu stāvokļu pāreju diagramma.....	85
3.3. Daļējs lietotāja saskarņu projektējums	89
3.3.1. Lietotāju saskarnes pārvietošanas diagramma.....	89
3.3.2. Saskarnes skati.....	91
4. TESTĒŠANAS DOKUMENTĀCIJA.....	100
4.1. Testēšanas Apraksts.....	100
4.2 Testpiemēru saraksts.....	100
5. PROJEKTA ORGANIZĀCIJA.....	114
5.1. Kvalitātes nodrošināšana	114
5.2. Konfigurāciju pārvaldība.....	115
5.3. Darbietilpības novērtēšana	115
REZULTĀTI UN DISKUSIJA	119
SECINĀJUMI	120
IZMANTOTĀ LITERATŪRA UN AVOTI	121
PIELIKUMI.....	122
1. pielikums. Lietojumprogrammas saskarnes servisu konfigurācija.....	122
2. pielikums. Komunikācijā starp tīmekļa vietni un lietojumprogrammas saskarni. 1.daļa...	124
3. pielikums. Komunikācijā starp tīmekļa vietni un lietojumprogrammas saskarni. 2.daļa...	128

APZĪMĒJUMU SARAKSTS

AAA (*Arrange-Act-Assert*) – vienībtestēšanas šablons, kas nedefinē testpiemēra struktūru.

API (*Application Programming Interface*) – lietojumprogrammas saskarne.

API Endpoint (*Application Programming Interface Endpoint*) – lietojumprogrammas saskarnes galapunkts. Tiek izmantots kā saziņas kanāls starp lietojumprogrammas saskarni un citu sistēmu.

AspNet Core Identity – ASP.NET satvara lietojumprogrammas saskarne, kas piedāvā funkcionalitāti lietotāju kontu, paroles, lomu, pretenziju, e-pasta apstiprinājumu pārvaldīšanai.

CSRF (*Cross site request forgery*) – starpvietņu pieprasījumu viltošanas uzbrukums.

CodeFirst – DDD pieeja, kas ļauj izveidot datu bāzi, izmantojot domēnā ieviestās klases un konfigurācijas.

DDD (*Domain-driven design*) – programmatūras izstrādes pieeja, kas koncentrējas uz domēna programmatūras modelēšanu, lai tā atbilstu domēna eksperta ievadei.

EF (*EntityFramework*) – .NET satvara objektrelāciju kartēšanas lietojumprogrammas saskarne.

FK (*Foreign Key*) – identifikators, kas veido savstarpējās saiknes starp tabulām datu bāzē bez jaunu tabulu izveides.

HTTPS (*Hypertext Transfer Protocol Secure*) – lietojumslāņa protokols, kas paredzēts hipertekstu datu apmaiņai. Protokols izmanto SSL/TLS protokolu šifrēšanai un autentifikācijai.

IIS (*Internet Information Services*) – paplašināma tīmekļa servera programmatūra, kura ir izveidota Windows NT saimes operētājsistēmām.

JSON (*JavaScript Object Notation*) – atvērta standarta faila un datu apmaiņas formāts, kas izmanto cilvēkiem lasāmu tekstu, lai uzglabātu un pārsūtītu datu objektus, kas sastāv no atribūtu-vērtību pāriem un masīviem.

ORM (*Object-relational mapping*) – programmēšanas tehnika datu konvertēšanai starp nesavienotām sistēmām, kas izmanto objektorientētās programmēšanas valodas.

PK (*Primary Key*) – unikāls identifikators, pēc kura ir iespējams identificēt ierakstu noteiktā datu bāzes tabulā

SPA (*Single-page application*) – tīmekļa lietojumprogramma vai vietne, kas mijiedarbojas ar lietotāju, dinamiski pārrakstot tīmekļa lapu ar jauniem datiem no tīmekļa servera.

XSS (*Cross-site scripting*) – starpvietņu skriptēšanas uzbrukums.

xUnit – .Net platformas vienībtestēšanas satvars.

IEVADS

Nolūks

Izveidots dokuments ir izstrādāts, lai dokumentētu plānošanas un laika pārvaldības tīmekļa sistēmas programmatūras prasību specifikāciju, programmatūras projektējuma aprakstu, izmantotas tehnoloģijas, veikto sistēmas testēšanu un projekta organizāciju. Dokuments tiek izmantots, lai iepazīstinātu pasūtītāju un potenciālos lietotājus ar sistēmas prasībām.

Darba autoram dokument ir paredzēts kā noslēguma darbs studiju programmā *Programmēšanas un datortīklu administrēšana*.

Darbības sfēra

Plānošanas un laika pārvaldības sistēma ir informācijas sistēma, kuru galvenais uzdevums ir piedāvāt sistēmas lietotājiem iespēju uzskaitīt un apskatīt laiku, cik tika atvēlēti noteiktā organizācijas/komandas uzdevuma pildīšanā. Sistēmā ļauj izveidot organizācijas un piedāvā funkcionalitāti organizāciju struktūras modificēšanai, tas ir, funkcionalitāte, kas ļauj izveidot organizācijas lietotāju lomas un komandas. Sistēma arī piedāvā funkcionalitāti noteiktas komandas izstrādes procesa pārvaldīšanai, veidojot uzdevumus komandas dalībniekiem un apskatīt uzdevuma izpildes procesu. Plānošanas un laika pārvaldības sistēma nodrošina saziņu starp sistēmas lietotājiem organizācijas un komandas ietvaros.

Saistība ar citiem dokumentiem

Dokumenta noformēšanā ievērotas standartu *LVS 68:1996 Programmatūras prasību specifikācijas ceļvedis* [1], *LVS 72:1996 Ieteicamā prakse programmatūras projektējuma aprakstīšanai* [2] un *LVS 70:1996 Programmatūras testēšanas dokumentācija* [3] prasības.

Standarta *ISO/IEC 19505-2:2012* [4] noteiktie apzīmējumi tika ievēroti un izmantoti UML diagrammu veidošanas procesā.

Pārskats

Dokumenta 1. sekcijā tiek aprakstīta ievadinformācija, kas satur dokumenta nolūku, mērķi, definīciju skaidrojumu, citu dokumentu izmantošanu dokumenta tapšanas laikā. Nodaļas mērķis ir vispārīgi noformulēt sistēmas prasības, ierobežojumus un raksturot to funkcionalitāti.

Dokumenta 2. sekcijā tiek aprakstīta programmatūras prasību specifikācija, kas satur sistēmas funkcionālo prasību sadalījumu pa moduļiem un aprakstus, datu plūsmas diagrammas, kā arī sistēmas nefunkcionālās prasības. Nodaļas mērķis ir apkopot un aprakstīt sistēmas funkcionalitāti, lietotāja ievaddatus un iegūstamo rezultātu.

Dokumenta 3. sekcijā tiek aprakstīts programmatūras projektējuma apraksts, kas satur datu bāzes projektējumu, daļēju funkciju projektējumu un daļēju saskarņu projektējumu. Nodaļas mērķis ir modelēt kā aprakstītās prasības tiks realizētas izvēlētajā vidē.

Dokumenta 4. sekcijā tiek aprakstīts programmatūras testēšana, kas satur testēšanas aprakst. Sekcijā tiek aprakstīts kādā veidā testēšana tika veikta un kādi testpiemēri tika izmantoti testēšanas laikā.

Dokumenta 5. sekcijā tiek aprakstīta projekta organizācija, kur tiek aprakstīta kvalitātes nodrošināšana, konfigurācijas pārvaldība un darbietilpības novērtēšana.

1. VISPĀRĪGS APRAKSTS

1.1. Esošā stāvokļa apraksts

Darba plānošanas un laika pārvaldības rīki pastāv tīmeklī lielā daudzumā, tomēr katrs rīks ir izveidots, lai atbilst noteiktas nozares prasībām un uzņēmuma operācijām. Kā arī rīki sevī ietver atsevišķu servisu kopumu, piemēram *Microsoft Outlook* ietver sevī funkcionalitāti tieši laika plānošanai, bet *Toggl* lietojumprogramma balstās tieši uz laika uzskaiti un lietojumprogrammas darbībai ir nepieciešama savienība ar plānošanas/izstrādes vadības platformām (piemēram, *Trello*, *Github* vai *Jira*).

Pastāv vairāki servisi, kuri koncentrējas uz spējas dzīves cikla izstrādi un piedāvā vairākas funkcijas vienas platformas ietvaros, kas palīdzēs izstrādes komandai izsekot visām izmaiņām un jaunumiem izstrādes procesā. Tādi servisi kā *Azure DevOps Services*, *Bitbucket*, *Jira*, *Github*, *Gitlab* piedāvā vairākas plānošanas iespējas.

Sistēma, kas tiks izstrādāta kvalifikācijas projekta ietvaros, implementēs plānošanas un laika pārvaldības rīku, kurš izstrādāts pirmkārt priekš laika uzskaites bez savienojuma ar minētām platformām. Sistēmas unikalitāte ir lomu uzdevumu izveidošana un uzdevumu ķēdes veidošana izstrādes procesa plānošanas stadijā.

1.2. Pasūtītājs

Plānošanas un laika pārvaldības sistēma ir izstrādāta pēc darba autora iniciatīvas noslēguma darba studiju programmā *Programmēšanas un datortīklu administrēšana* ietvaros.

1.3. Produkta perspektīvas

Plānošanas un laika pārvaldības sistēma ir pašpietiekams produkts, tomēr pastāv iespēja sistēmas papildināšanai, lai izveidotu sadarbību ar spējas izstrādes platformām (piemēram *Azure DevOps Services*).

1.4. Darījumasprasības

Plānošanas un laika pārvaldības sistēmai ir jānodrošina sekojošas funkcijas:

- Lietotāju autentifikācijas un autorizācijas mehānismu izveidošana, savukārt piedāvāt neautenticētam lietotājam sistēmas reģistrēšanas vai pieteikšanās iespējas.

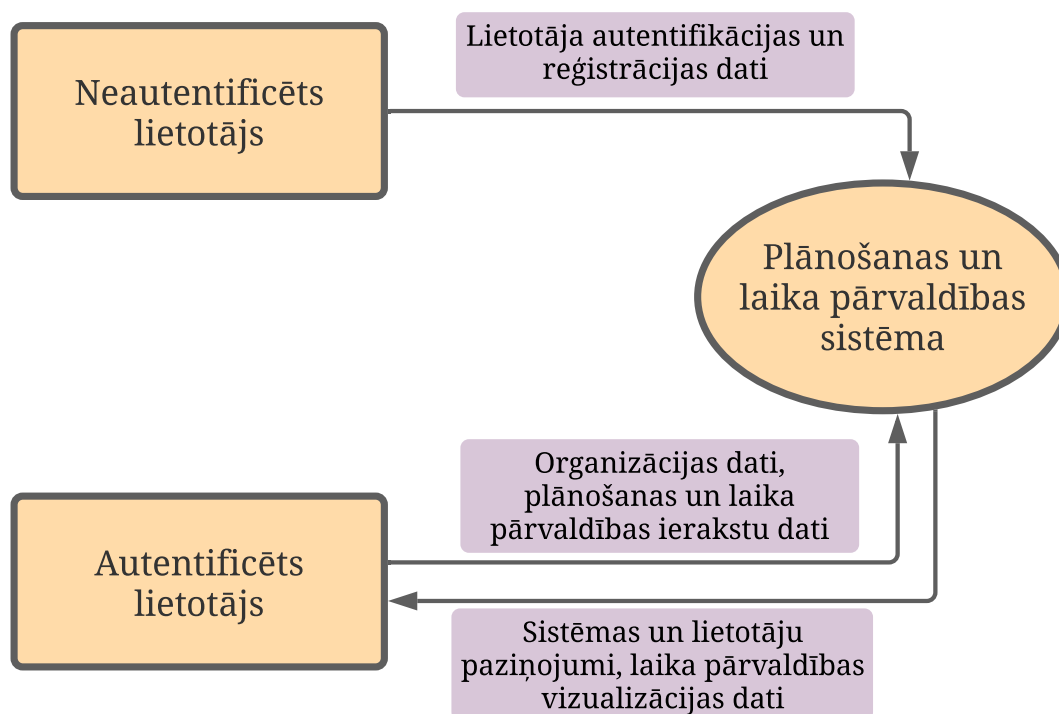
- Jaunas organizācijas struktūras izveidošanas funkcionalitāte, tas ir, organizācijas dalībnieku uzaicināšana, organizācijas lomu un organizācijas komandu izveidošana un to datu rediģēšana. Komandas un lomu piešķiršana organizācijas dalībniekiem un tiesību regulēšana noteiktas organizācijas un komandas ietvaros.
- Izstrādes procesa izveidošanas funkcionalitāte, tas ir, komandas izstrādes pazīmju un uzdevumu izveidošana. Uzdevumu piešķiršana izvēltam komandas dalībniekam vai komandas lomai.
- Izstrādes procesa regulēšanas funkcionalitāte, tas ir, lomas uzdevumu pieņemšanas iespēja komandas lomas dalībniekam, uzdevumu ķēdes atsaucis izveidošana uzdevumu ķēdes izpildei, uzdevuma aizvēršanas un uzdevuma pārskata funkcionalitāte.
- Sazināšanas funkcionalitāte, tas ir, ziņojumu izveidošana par piešķirto lomu vai komandu organizācijas ietvaros. Ziņojumu izveidošana par piešķirto uzdevumu vai lomas uzdevumu izvēltas komandas procesa izstrādes ietvaros. Organizācijas uzaicinājuma paziņojuma izveide un atbildes apstrāde.
- Laika pārvaldības reģistrēšanas funkcionalitāte, tas ir, iespēja komandas dalībniekam reģistrēt cik laika tika atvēlēts izvēltam komandas uzdevumam.
- Laika pārvaldības attēlošanas funkcionalitāte, tas ir, iespēja organizācijas īpašniekam vai noteiktas komandas vadītājam apskatīt izvēltas komandas, komandas lomas vai organizācijas dalībnieka laika pārvaldību.

1.5. Sistēmas lietotāji

Plānošanas un laika pārvaldības sistēma iedala sistēmas lietotājus divos veidos: autentificēts un neautentificēts lietotājs (sk. 1.1. att.).

Neautentificēts lietotājs ir sistēmas lietotājs, informācija par kuru netika piedāvāta sistēmai. Tā kā neautentificētam lietotājam nav iespējams pārbaudīt piekļuves tiesības sistēmas vietnēm un datiem, lietotājam tiek piešķirta piekļuve tikai sistēmas publiskām vietnēm, kas ir sistēmas mājas, reģistrēšanas un pieteikšanās vietnes. Neautentificētam lietotājam ir iespēja pieteikties sistēmai, piedāvājot personas akreditācijas datus, kuri atbilst izveidotam lietotāju kontam, kas eksistē sistēmas ietvaros. Neautentificēts lietotājs var reģistrēties sistēmā, piedāvājot nepieciešamo informāciju sistēmas lietotāja konta izveidei.

Autentificēts lietotājs ir sistēmas lietotājs, informācija par kuru tiek glabāta sistēmā un kurš veiksmīgi izmantoja sistēma autentifikācijas un autorizācijas mehānismu, piedāvājot sistēmai lietotāja akreditācijas datus. Autentificētam lietotājam tiek piedāvāta iespēja izveidot jaunas organizācijas vai kļūt par eksistējošas organizācijas dalībnieku, pieņemot cita autentificēta lietotāja uzaicinājumu.



1.1. att. Konceptuālā datu plūsmas diagramma izstrādātai sistēmai.

Plānošanas un laika pārvaldības sistēmā tikai autentificētam lietotājam tika piedāvāta iespēja veidot organizāciju struktūras un mijiedarboties ar organizācijas aktivitātēm (uzdevuma veikšanas procesa reģistrēšana, paziņojumu izveidošana un saņemšana, komandas izstrādes procesa attēlošana u.t.t.). Vienas organizācijas ietvaros autentificēts lietotājs tiek iedalīts trīs veidos (sk. 1.2. att.):

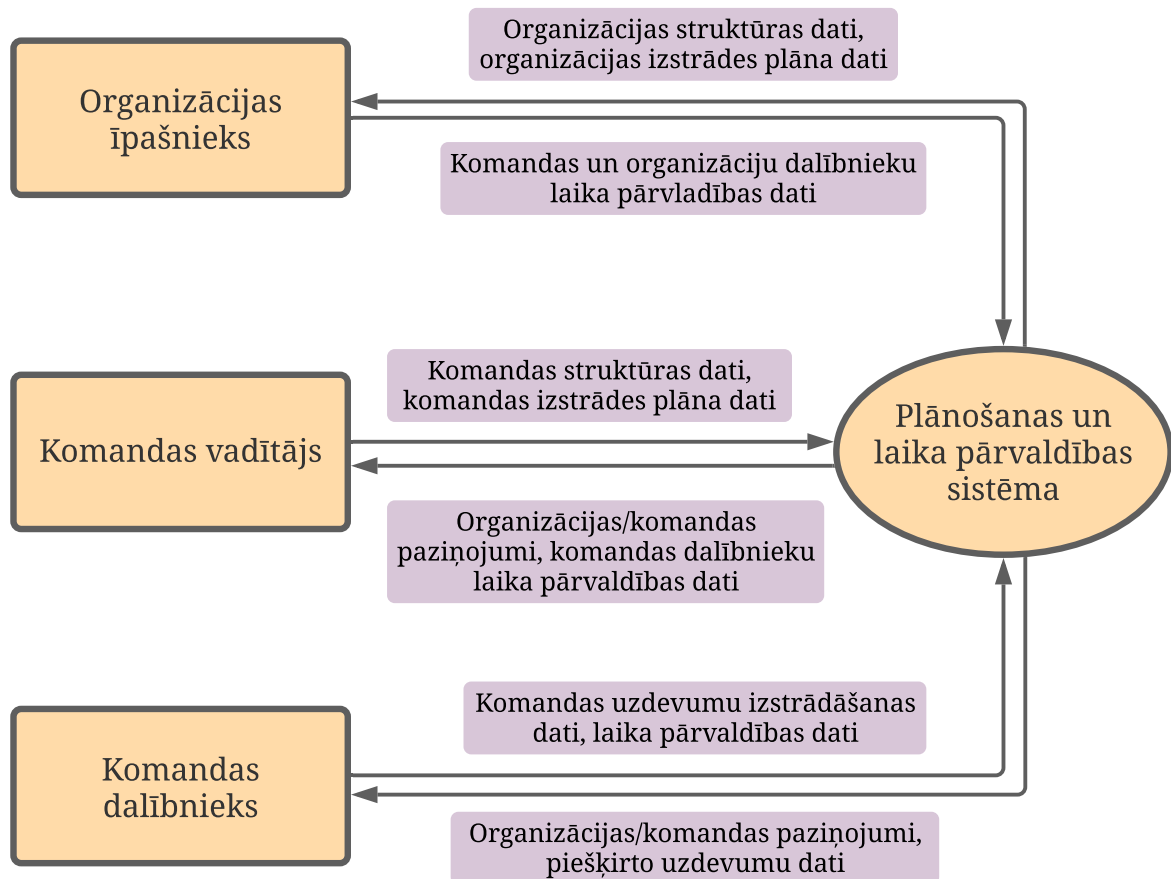
- Organizācijas īpašnieks
- Komandas vadītājs
- Komandas dalībnieks

Organizācijas īpašniekam tiek piedāvāta funkcionalitāte organizācijas struktūras veidošanai, tas ir, organizācijas lomu izveidošana un piešķiršana organizācijas dalībniekiem, organizācijas komandu izveidošana, uzaicinājumu veidošanas ārpus organizācijas lietotājiem ar piedāvājumu kļūt par organizācijas dalībnieku. Organizācijas īpašniekam ir tiesības piekļūt

un modificēt visus datus, kas tiek reģistrēti organizācijas ietvaros, tajā skaitā komandas izstrādāto pazīmju datiem un organizācijas dalībnieka kalendāru ierakstiem. Organizācijas īpašnieks var veidot vizuālo izstrādes procesa attēlošanu visām organizācijas komandām. Organizācijas īpašnieks sevī ietver tiesības kā arī sistēmas funkcijas, kas tiek piedāvātas komandas vadītājam un komandas dalībniekam.

Komandas vadītājam tiek piedāvāta funkcionalitāte komandas izstrādes plāna veidošanai, tas ir, komandas pazīmju izveidošana, komandas uzdevumu izveidošana un piešķiršana noteiktam komandas dalībniekam, komandas dalībnieku lomu modificēšana. Komandas vadītājam ir tiesības piekļūt un modificēt visus datus, kas tiek reģistrēti vadītas komandas ietvaros. Komandas vadītājs var veidot vizuālo izstrādes procesa attēlošanu vadītai komandai. Komandas vadītājs sevī ietver tiesības kā arī sistēmas funkcijas, kas tiek piedāvātas komandas dalībniekam.

Komandas dalībniekam tiek piedāvāta funkcionalitāte kalendāru ierakstu izveidošanai, kur tiek attēlots piešķirto uzdevumu izpildes process (patērētais laiks). Komandas dalībniekam ir tiesības piekļūt savas komandas datiem, bet modificēt ir iespējams tikai datus, kuri tika piešķirti komandas dalībniekam (piemēram, uzdevums, par kuru izpildi ir atbildīgs komandas dalībnieks).



1.2. att. Organizācijas dalībnieku konceptuālā datu plūsmas diagramma izstrādātai sistēmai.

1.6. Vispārējie ierobežojumi

Plānošanas un laika pārvaldības sistēmā pastāv sekojošie vispārējie ierobežojumi:

- Sistēmas darbība netiek nodrošināta bez interneta savienojuma.
- Sistēma netiek atbalstīts reaģējošs tīmekļa vietnes dizains
- Sistēma atbalsta tikai tīmekļa pārlūkus, kuri atbalsta aktuālos tīmekļa standartus (HTML5 [5] , CSS3 [6], ECMA-262 [7]). Sistēma atbalsta sekojošu tīmekļa pārlūkus: Microsoft Edge, Google Chrom, Firefox, Opera.
- Sistēmas autentifikācijas un autorizācijas mehānismi ir pieejami tikai tad, kad tiek nodrošināts HTTPS protokols.
- Sistēmas darbība notiek vienas laika joslas ietvaros.

1.7. Pieņēmumi un atkarības

Plānošanas un laika pārvaldības sistēmas izstrādē tika pieņemts:

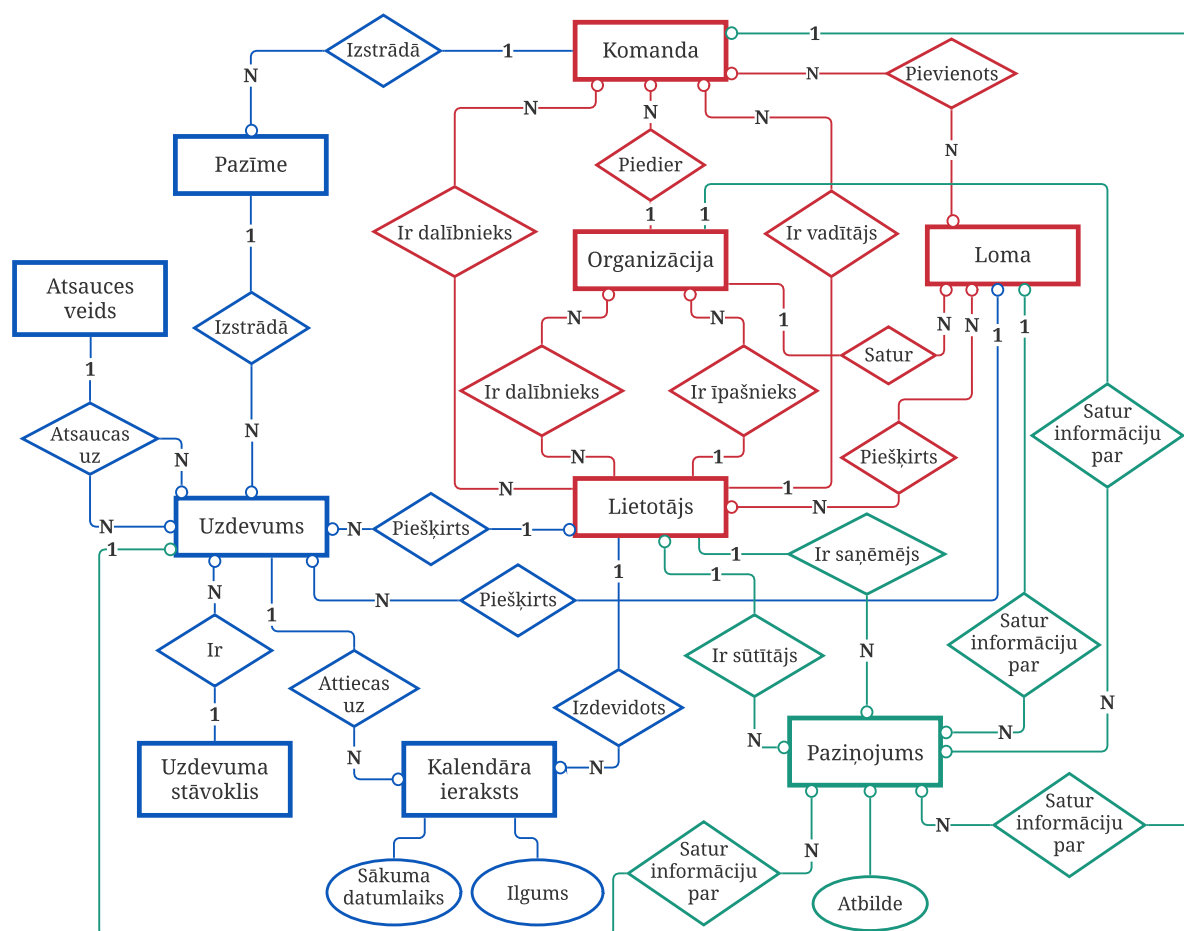
- Lietotājam pastāv e-pasts, lai reģistrēties sistēmai.
- Lietotājs nemainīs tīmekļa vietnes kodu, izmantojot pārlūka izstrādes instrumentus.
- Sistēmas nolūka izpildei ir pieņemts, ka komandas dalībnieki uzskaitīs laiku, kas tika atvelēts uzdevumu izpildei un neveidos nekorektus datus. Sistēmā nepastāv iespēja validēt komandas dalībnieka izveidoto laika uzskaites ieraksta atbilstību, jo sistēmā nav informācijas par lietotāju aktivitātēm un darbībām ārpus sistēmas.
- Tiek pieņemts, ka izstrādes vadītājs aizvērs uzdevumu vai pazīmju izpildi, kad uzdevuma vai pazīmes izstrāde tika pabeigta. Sistēmai nav iespēju validēt aizvēšanas pieprasījumu, jo sistēmā nav informācijas par izstrādes procesu, kas tiek veidots ārpus sistēmas.

Plānošanas un laika pārvaldības sistēma ir atkarīga no:

- Savienojums ar datu bāzes serveru.
- Savienojums ar sistēmas lietojumprogrammas saskarni.
- SSL sertifikāta, kas tiks izmantots HTTPS savienojuma veidošanai.

2. PROGRAMMATŪRAS PRASĪBU SPECIFIKĀCIJA

2.1. Konceptuālais datu bāzes apraksts



2.1. att. Konceptuālais datu bāzes modelis izstrādātai sistēmai.

Konceptuālais datu bāzes modelis (sk. 2.1. att.) ar krāsas palīdzību ir sadalīts trīs daļās, kas reprezentē noteikto sistēmas moduli. Entītijas ir piešķirtas noteiktam pamata modulim, kurā tiek veikta entītiju datu reģistrēšana un modificēšana, tomēr parēju sistēmas moduļu ietvaros pastāv iespēja izmantot un apstrādāt entītiju datus. Konceptuālajā datu bāzes modelī ir attēloti sekojošie moduļi:

- Organizācijas struktūras veidošanas modulis (sarkanā krāsa)
- Izstrādes procesa vadības modulis (zilā krāsa)
- Sazināšanas modulis (zaļā krāsa)

Organizācijas struktūras veidošanas modulis sastāv no entītijām *Lietotājs*, *Organizācija*, *Loma* un *Komanda*.

Entītijā *Lietotājs* tiek glabāti lietotāju akreditācijas dati, kas tiek izmantoti sistēmas autentifikācijas un autorizācijas mehānismos. Tiek uzskatīts ka katram entītijas *Lietotājs* ierakstam, kas izveidots sistēmas ietvaros, atbilst tieši viens lietotāja konts, caur kuru lietotājs mijiedarbojas ar sistēmu.

Entītijā *Organizācija* tiek glabāti dati, kas ir saistīti ar izveidotas organizācijas aprakstu (nosaukums, apraksts u.t.t.). Organizācijai ir tieši viens īpašnieks (lietotājs, kurš reģistrē organizāciju) un neierobežots skaits organizācijas dalībnieku. Lietotājam ir iespēja izveidot vairākas organizācijas kā arī būt vairāku organizāciju dalībniekam. Pēc lietotāja konta reģistrēšanas lietotājs nav nevienas organizācijas īpašnieks un nav nevienas organizācijas dalībnieks.

Entītijā *Loma* tiek glabāti dati, kas ir saistīti ar izveidotas lomas informāciju (nosaukums, apraksts, lomas krāsa u.t.t.). Katrā organizācijā pastāv iespēja izveidot savu lomu kopumu, bet izveidotā loma ir pieejama tikai organizācijas ietvaros, kurā tā tika reģistrēta. Tas nozīmē, ka sistēmā nepastāv iespēja pārnest lomu no vienas organizācijas uz citu. Lietotājam var piešķirt vairākas lomas kā arī viena loma var tiks piešķirta vairākiem lietotājiem, kas ir organizācijas dalībnieki, kur noteikta loma tika izveidota.

Entītijā *Komanda* tiek glabāti dati, kas ir saistīti ar izveidotas komandas informāciju (nosaukums, apraksts u.t.t.). Līdzīgi kā lomas izveidošanai, katrā organizācijā pastāv iespēja izveidot savu komandu kopumu, kur katra izveidota komanda ir pieejama tikai organizācijas ietvaros. Reģistrējot jaunu komandu ir nepieciešams izvēlēties komandas vadītāju no organizācijas dalībnieku kopas. Komandai ir iespējams pievienot neierobežotu skaitu organizācijas lomu un organizācijas dalībnieku. Organizācijas dalībnieks var tikt pievienots vairākās komandās (kā arī vadīt vairākas komandas) un organizācijas loma var tikt izmantota vairākās komandās.

Izstrādes procesa vadības modulis sastāv no entītijām *Pazīme*, *Uzdevums*, *Uzdevuma stāvoklis*, *Atsauces veids*, *Kalendāra ieraksts*.

Entītijā *Pazīme* tiek glabāti dati, kas ir saistīti ar izveidotas pazīmes informāciju (nosaukums, apraksts, pazīmes izstrādes plānotais sākums, pazīmes izstrādes plānotais ilgums u.t.t.). Sistēmā par pazīmi tiek uzskatīts uzdevumu kopums, kuru nepieciešams paveikt organizācijas komandai, lai sasniegtu uzstādītu starpposma mērķi (piemēram, ieviest organizācijas produktā jaunu moduli). Komanda var izstrādāt vairākas pazīmes, bet noteiktu pazīmi izstrādā tieši viena komanda.

Entītijā *Uzdevums* tiek glabāti dati, kas satur izveidoto uzdevumu informāciju (uzdevuma kods, nosaukums, apraksts) un kas attiecās uz izstrādātas pazīmes funkcionalitātes daļu, tāpēc vienas pazīmes ietvaros ir iespējams izveidot vairākus uzdevumus, bet noteikts uzdevums pieder tieši vienai pazīmei. Entīcija *Uzdevumu stāvoklis* ir saistīts tikai ar entītijā *Uzdevums* un tajā tiek glabāts noteikta uzdevuma stāvoklis.

Sistēmā tiek izveidoti sekojošie uzdevuma stāvokļi:

- Izpildītāja piešķiršana – stāvoklis, kas tiek piešķirts uzdevumam, kad izstrādes vadītājs reģistrē jaunu uzdevumu sistēmā un uzdevuma izpildītājs netika piešķirts.
- Saistīts – stāvoklis, kas tiek piešķirts uzdevumam, kad izveidotais uzdevums ir saistīts ar citu komandas izstrādes procesa uzdevumu. Ja uzdevums pieņem stāvokli *Saistīts*, tad uzdevumu nav iespējams pildīt/izstrādāt kamēr saistītais uzdevums netiks izpildīts un pieņems uzdevuma stāvokli *Aizvērts* (tiek pieminēts turpmāk sarakstā)
- Piešķirts – stāvoklis, kas tiek piešķirts uzdevumam, kad uzdevuma izpildītājs tika piešķirts uzdevumam un visi saistītie uzdevumi pieņēma stāvokli *Aizvērts*.
- Aktīvs – stāvoklis, kas tiek piešķirts uzdevumam, kad uzdevuma izpildītājs atzīmēja uzdevumu izpildīšanas laiku ar pirmo izveidoto kalendāra ierakstu.
- Pārskats – stāvoklis, kas tiek piešķirts uzdevumam, kad uzdevuma izpildītājs pabeidza uzdevuma izpildi/izstrādi un atzīmēja sistēmā, ka uzdevums ir gatavs būt pārskatīts izstrādes vadītājam.
- Aizvērts – stāvoklis, kas tiek piešķirts uzdevumam, kad komandas vadītājs pārskatīja uzdevumu un akceptēja uzdevuma izpildīšanu.

Izveidojot uzdevumu, tam nepieciešams piešķirt lomu, kuru dalībnieks turpmāk izpildīs to, vai uzreiz izvelēties komandas dalībnieku, kurš izpildīs to. Komandas lomu vai komandas dalībnieku ir iespējams izvelēties pēc uzdevuma reģistrācijas sistēmā. Reģistrējot uzdevumu, pastāv iespēja atsaukties uz iepriekš ieviesto sistēmā uzdevumu. Šis funkcionāls ir ieviests, lai izveidotu uzdevumu izpildes ķēdi vai veikt informatīvo atsauci uz iepriekš paveikto darbu. Atsauces veidu informācija tiek glabāta entītijā *Atsauces veids*. Uzdevums var atsaukties uz vairākiem citiem uzdevumiem kā arī uz vienu uzdevumu var atsaukties daudzi citi uzdevumi. Veidojot atsauci ir nepieciešams norādīt atsauces veidu.

Sistēmā tiek izveidoti sekojošie atsauces veidi:

- Informatīva atsauce – atsauces veids, kuru nolūks ir informēt uzdevuma izpildītāju, ka uzdevums ir saistīts ar pazīmes izpildīto vai aktīvo uzdevumu.
- Ķēdes veidošanas atsauce – atsauces veids, kuru nolūks ir saistīt izveidoto uzdevumu ar pazīmes aktīvo uzdevumu. Ja uzdevumam ir izveidota *Ķēdes veidošanas atsauce* un uzdevumam, uz kuru izveidota atsauce, nav stāvoklis *Aizvērts*, tad uzdevuma izpildītājam nav iespējas uzsākt uzdevuma izpildi. Uzdevums kļūst pieejams izpildei tad un tikai tad, kad visi uzdevumi, uz kuriem ir izveidota *Ķēdes veidošanas atsauce*, pieņems uzdevuma stāvokli *Aizvērts*.

Entītijā *Kalendāra ieraksts* tiek glabāta informācija par laiku, ko noteikts lietotājs ir pavadījis, pildot izvēlēta uzdevuma daļu. Reģistrējot kalendāra ierakstu ir nepieciešams norādīt uzdevumu, lietotāju, sākuma datumu laiku un beigu datumu laiku. Vienam uzdevumam kā arī vienam lietotājam var būt vairāki kalendāra ieraksti, tomēr katrs kalendāra ieraksts attiecas tieši uz vienu organizācijas lietotāju un pazīmes uzdevumu.

Sazināšanas modulis sastāv no vienīgas entītijas *Paziņojums*. Entītijā sevī ietver ziņojuma informāciju, kuru nepieciešams attēlot noteiktam lietotājam. Reģistrējot ziņojumu sistēmā, ir nepieciešams norādīt ziņojuma saņēmēju un organizāciju, ar kuru ziņojuma informācija ir saistīta (visas darbības, kuras tiek veiktas sistēmā ir saistītas ar noteikto organizāciju, izņemot sistēmas autorizācijas un autentifikācijas mehānismus). Ja ziņojumu veido noteikts lietotājs, tad šis lietotājs ir atzīmēts kā ziņojuma sūtītājs. Ja paziņojums satur informāciju par konkrēto lomu, uzdevumu vai komandu, tad entītijas ieraksts satur atsauci uz noteiktu lomu, uzdevumu vai komandu, kas tiek pieminēta ziņojuma saturā. Sistēmā pastāv iespēja izveidot paziņojumu, kurš pieprasa saņēmēju pozitīvi vai negatīvi atbildēt uz jautājumu. Atbilde tiek saglabāta entītijas ierakstā.

2.2. Funkcionālās prasības

2.2.1. Vispārējās nodaļas, kas saistītas ar funkciju aprakstīšanu

2.2.1.1. Funkciju aprakstīšanas struktūra

Moduļa funkcijas apraksts tiks iedalīts uz sekojošām daļām:

- Ievads – moduļa funkcijas apraksta sadaļa, kas sevī ietver izstrādātas funkcijas galveno uzdevumu un mērķi

- Ievaddati – moduļa funkcijas apraksta sadaļa, kas sevī ietver funkcijai nepieciešamo ievaddatu aprakstu. Ievaddatu sadaļā tiks specificēts vai funkcijai dati tiks padoti, izmantojot formu ar ievadlaukiem vai izmantojot citas funkcijas izsaukumu.
- Apstrāde – moduļa funkcijas apraksta sadaļa, kas sevī ietver funkcijas ievaddatu apstrādi. Apstrādes sadaļā tiks specificētas papildus ievaddatu pārbaudes un kļūdu ziņojumu izsaukumi. Tiks specificēts datu apstrādes process un entītijas, kur dati tiks saglabāti.
- Izvaddati – moduļa funkcijas apraksta sadaļa, kas sevī ietver funkcijas izvaddatu aprakstu. Izvaddatu sadaļā arī tiks specificēti ziņojumi, kas tiks attēloti lietotājam pēc funkcijas izpildes, un vietnes, kur lietotājs tiks novirzīts, ja funkcija tika veiksmīgi izpildīta.
- Kļūdu ziņojumi – moduļa funkcijas apraksta sadaļa, kas sevī ietver kļūdu ziņojumus, kuri tiek izveidoti funkcijas izpildes laikā.

2.2.1.2. Kopīgie kļūdu ziņojumi un ievadlauku ierobežojumi

Ja funkcijas apstrādes laikā netiek atrasts nepieciešamais datu bāzes ieraksts (situācija, kad datu bāzes administrators izdzēsa ierakstu vai ierakstam tiek piekļauts ārpus sistēmas piedāvāto lietotāju saskarni), tad sistēmā tiek izvadīts kļūdas paziņojumu nr. 1.

Katrai lietotāja aizpildāmai ievadformai pirms un pēc iesniegšanas tiek pārbaudīts, vai aizpildīti visi obligāti aizpildāmie lauki (norādīts funkcionālo prasību aprakstos sadaļā ievaddati). Ja kaut viens obligāts lauks nav aizpildīts, tad, spiežot formas iesniegšanas pogu, forma netiek iesniegta un zem atbilstošajiem laukiem tiek izvadīts kļūdas paziņojums nr. 2.

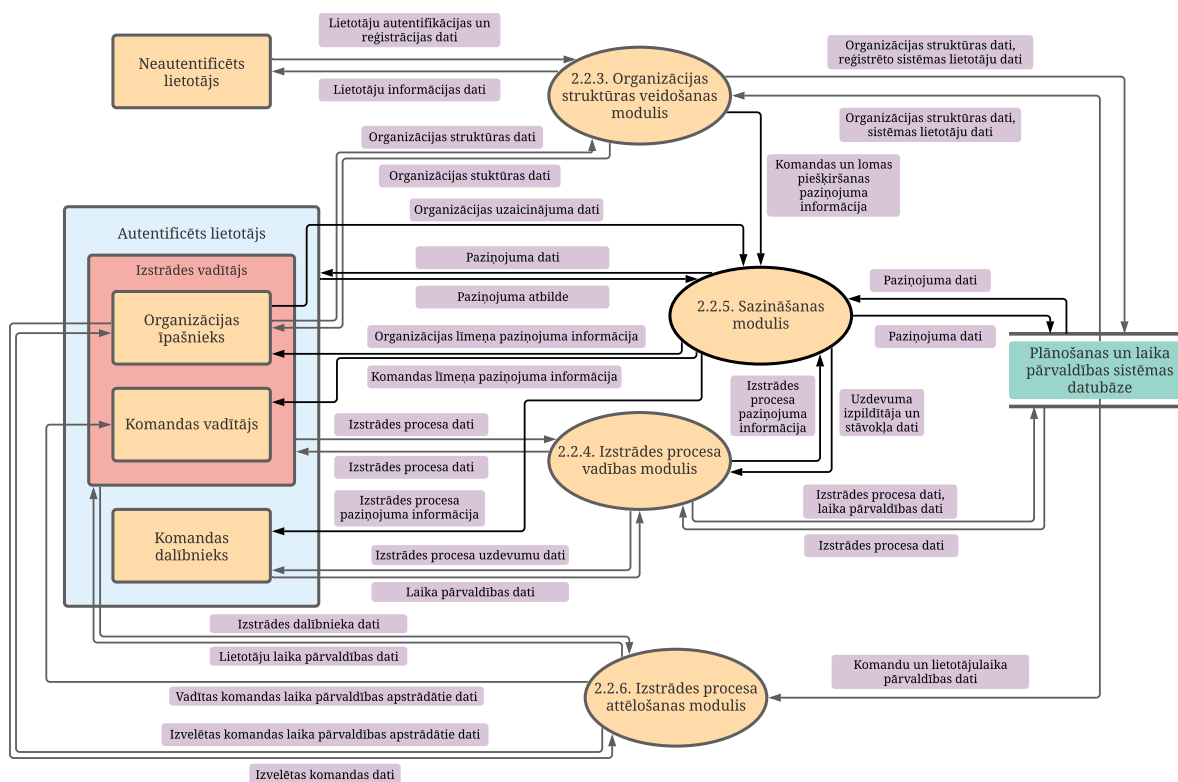
Apstrādājot ievadformas teksta lauku, tiek pārbaudīts ievadīto simbolu skaits. Funkcionālo prasību aprakstos ievaddatu sadaļā tiek specificēts minimālais un maksimālais simbolu skaits. Pēc formas iesniegšanas pogas uzspiešanas, tiek izvadīts kļūdu ziņojums nr. 3. ja simbolu skaits pārsniedz maksimālo simbolu skaitu. Ja simbolu skaits nesasniedz norādīto minimālo simbolu skaitu, tad tiek izvadīts kļūdu ziņojums nr. 4. Ja ievadformā kaut viens teksta ievadlauks satur kļūdas ziņojumu, tad forma netiek iesniegta un zem teksta lauka ar nepareizo formātu tiek attēlots kļūdas ziņojums.

Datumu un laiku ievaddatus var izvēlēties no kalendāra vai pulksteņa, spiežot uz ikonas blakus teksta formāta ievadlaukam. Ja ievadlaukā rakstītā vērtība neatbilst datuma un laika formātam (MM/dd/yyyy HH:mm:ss vai MM/dd/yyyy), tad tiek izvadīts kļūdas ziņojumu nr. 5.

Kopīgie kļūdu ziņojumi

1. [Entītijas nosaukums] nav atrasta sistēmas ietvaros.
2. Obligāti aizpildāms lauks.
3. Maksimālais atļautais teksta garums: [skaitlis].
4. Minimālais atļautais teksta garums: [skaitlis].
5. Neatbilstošs datuma un laika formāts. Ievaddatiem ir nepieciešams atbilst sekojošam formātam: dd/MM/yyyy HH:mm:ss (datumam: dd/MM/yyyy)

2.2.2. Funkciju sadalījums pa moduļiem



2.2. att. 1. līmeņa datu plūsmas diagramma plānošanas un laika pārvaldības sistēmai

Plānošanas un laika pārvaldības sistēma ir iedalīta četros moduļos, kas ir attēloti 1. līmeņa datu plūsmas diagrammā (sk. 2.2. att.). Diagramma sevī ietver sekojošus moduļus.

- Organizācijas struktūras veidošanas modulis

- Izstrādes procesa vadības modulis
- Sazināšanas modulis
- Izstrādes procesa attēlošanas modulis

Organizācijas struktūras veidošanas modulis ir atbildīgs par organizācijas struktūru veidošanu, tas ir, organizācijas izveidošana sistēmā, komandu un lomu izveidošana un to piešķiršana organizācijas dalībniekiem. Kā arī modulis ietver autentifikācijas un autorizācijas mehānismus.

Izstrādes procesa vadības modulis ir atbildīgs par noteiktas komandas izstrādes procesa vadību, tas ir, pazīmju un to uzdevumu izveidošana, uzdevumu piešķiršana noteiktam komandas dalībniekam, patērētā laika reģistrēšana sistēmā, kuru komandas dalībnieks patērēja, izpildot tam piešķirto uzdevumu.

Sazināšanas modulis tiek izmantots organizācijas struktūras veidošanas modulī un izstrādes procesa vadības modulī ar mērķi nosūtīt paziņojumu par izveidotu pasākumu. Ja paziņojumam ir nepieciešama saņēmējā atbilde un turpmāka sistēmas darbība ir atkarīga no padotas atbildes (piemēram, lietotāja uzaicināšana organizācijā), tad funkcijas, kas apstrādā padoto atbildi un veic darbības, pamatojoties uz atbildes vērtības, tiek implementētas sazināšanas moduļa ietvaros.

Izstrādes procesa attēlošanas modulis sevī ietver noteiktas komandas izstrādes procesa attēlošanas un vizualizēšanas funkcijas. Modulis nemodificē un neveido jaunus ierakstus datu bāzē un izmanto informāciju, kas tika reģistrēta sistēmā darbības laikā.

Katra moduļa funkcijas tiek aprakstītas moduļu funkcijas tabulā (sk. 2.1. tabulu)

2.1. tabula

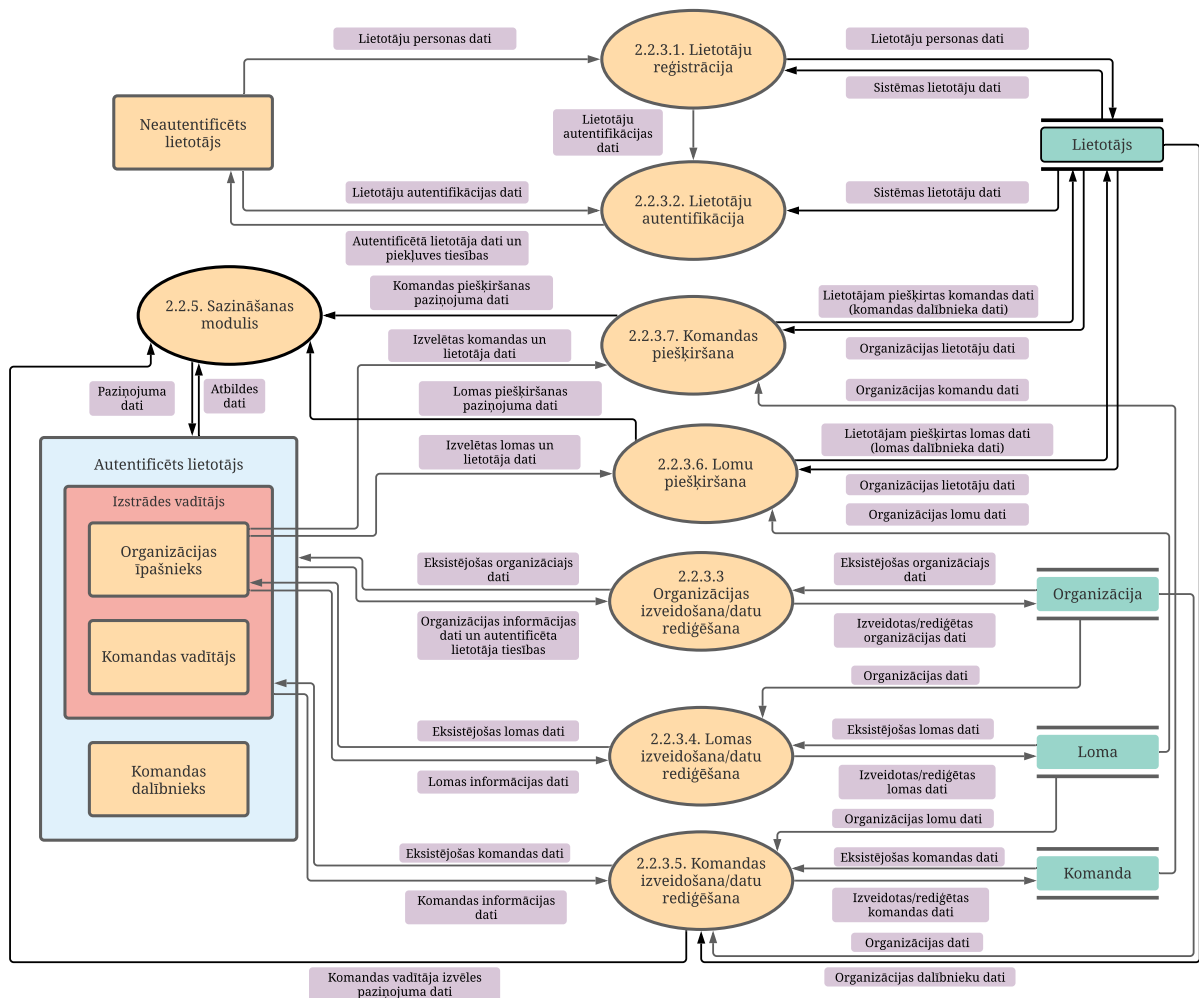
Plānošanas un laika pārvaldīšanas sistēmas moduļu funkcijas saraksts

<i>Modulis</i>	<i>Funkcija</i>	<i>Sistēmas lietotājs</i>
<i>2.2.3. Organizācijas struktūras veidošanas modulis</i>	<i>2.2.3.1. Lietotāju reģistrācija</i>	<i>Neautenticēts lietotājs</i>
	<i>2.2.3.2. Lietotāju autentifikācija</i>	<i>Neautenticēts lietotājs</i>

<i>Modulis</i>	<i>Funkcija</i>	<i>Sistēmas lietotājs</i>
	<i>2.2.3.3 Organizācijas izveidošana / datu rediģēšana</i>	<i>Autentificēts lietotājs</i>
	<i>2.2.3.4. Lomas izveidošana / datu rediģēšana</i>	<i>Organizācijas īpašnieks</i>
	<i>2.2.3.5. Komandas izveidošana / datu rediģēšana</i>	<i>Izstrādes vadītājs: Organizācijas īpašnieks, Komandas vadītājs</i>
	<i>2.2.3.6. Lomu piešķiršana</i>	<i>Organizācijas īpašnieks</i>
	<i>2.2.3.7. Komandas piešķiršana</i>	<i>Organizācijas īpašnieks</i>
	<i>2.2.4.1. Pazīmes izveidošana / datu rediģēšana</i>	<i>Izstrādes vadītājs: Organizācijas īpašnieks, Komandas vadītājs</i>
	<i>2.2.4.2. Uzdevuma izveidošana / datu rediģēšana</i>	<i>Izstrādes vadītājs: Organizācijas īpašnieks, Komandas vadītājs</i>
<i>2.2.4. Izstrādes procesa vadības modulis</i>	<i>2.2.4.3. Uzdevuma izpildītāja iestatīšana</i>	<i>Izstrādes vadītājs: Organizācijas īpašnieks, Komandas vadītājs</i>
	<i>2.2.4.4. Uzdevuma stāvokļa iestatīšana</i>	<i>Organizācijas īpašnieks, Komandas vadītājs, Komandas dalībnieks</i>
	<i>2.2.4.5. Kalendāra ieraksta izveidošana / datu rediģēšana</i>	<i>Organizācijas īpašnieks, Komandas vadītājs, Komandas dalībnieks</i>
	<i>2.2.5.1. Lietotāja paziņojuma izveide</i>	<i>Organizācijas īpašnieks, Komandas vadītājs, Komandas dalībnieks</i>
<i>2.2.5. Sazināšanas modulis</i>	<i>2.2.5.2. Komandas lomas paziņojuma izveide</i>	<i>Organizācijas īpašnieks, Komandas vadītājs</i>
	<i>2.2.5.3. Lomas uzdevuma pieņemšana</i>	<i>Organizācijas īpašnieks, Komandas vadītājs, Komandas dalībnieks</i>
	<i>2.2.5.4. Lietotāja uzaicināšana organizācijā</i>	<i>Organizācijas īpašnieks</i>

Modulis	Funkcija	Sistēmas lietotājs
	2.2.5.5. Organizācijas uzaicinājuma pieņemšana	Autenticēts lietotājs
2.2.6. Izstrādes procesa attēlošanas modulis	2.2.6.1. Komandas dalībnieka laika pārvaldības apskate	Izstrādes vadītājs: Organizācijas īpašnieks, Komandas vadītājs
	2.2.6.2. Komandas lomas laika pārvaldības apskate	Izstrādes vadītājs: Organizācijas īpašnieks, Komandas vadītājs
	2.2.6.3. Komandas laika pārvaldības apskate	Izstrādes vadītājs: Organizācijas īpašnieks, Komandas vadītājs

2.2.3. Organizācijas struktūras veidošanas modulis



2.3. att. 2. līmeņa datu plūsmas diagramma organizācijas struktūras datu veidošanas moduļim

Organizācijas struktūras datu veidošanas moduļa datu plūsmas diagrammā (sk. 2.3. att.) tiek attēlotas datu plūsmas starp sistēmas lietotājiem, moduļa funkcijām un entītijām, kas tiek izmantoti moduļa ietvaros. Modulis izmanto entītijas *Lietotājs*, *Organizācijas*, *Loma* un *Komanda*. Diagrammā papildus tiek attēlotas datu plūsmas, kas ir saistītas ar sazināšanas moduļa funkcijām (sk. 2.2.5. *Sazināšanas modulis*).

2.2.3.1. Lietotāju reģistrācija

Ievads

Funkcija nodrošina iespēju neautenticētam lietotājam reģistrēt sistēmā jaunu lietotāju kontu, izmantojot neautenticētā lietotāja personas datus.

Ievaddati

Neautenticētam lietotājam tiek piedāvāta forma, kur ir nepieciešams norādīt sekojošo informāciju:

- Lietotāja vārds. Simbolu virkne, 1-40 simboli. Obligāts.
- Lietotāja otrais vārds. Simbolu virkne, 1-40 simboli. Neobligāts.
- Lietotāja uzvārds. Simbolu virkne, 1-40 simboli. Obligāts.
- Lietotājvārds, Simbolu virkne, 1-40 simboli. Obligāts. Unikāls.
- E-pasts. Simbolu virkne, 1-255 simboli. Obligāts. Unikāls.
- Lietotāju parole. Simbolu virkne, 8-255 simboli. Obligāts.
- Atkārtota lietotāju parole. Simbolu virkne, 8-255 simboli. Obligāts.

Apstrāde

Tiek pārbaudīts, vai formas ievadlauki atbilst sistēmas vispārējiem ierobežojumiem, kas ir nodefinēti sekcijā *Ievaddati* un aprakstīti sekcijā 2.2.1.2. *Kopīgie kļūdu ziņojumi un ievadlauku ierobežojumi*.

Tiek pārbaudīts, vai sistēmā netika reģistrēts lietotāja konts ar norādīto lietotājvārdu. Ja sistēmā pastāv lietotājs ar norādīto lietotājvārdu, tad tiek izvadīts kļūdas paziņojums nr. 1.

Tiek pārbaudīts, vai ievadītais e-pasts satur e-pasta simbolu @. Ja e-pasta ievadītais lauks nesatur e-pasta simbolu, tad tiek izvadīts kļūdas paziņojums nr. 2.

Tiek pārbaudīts, vai sistēmā netika reģistrēts lietotāja konts ar norādīto e-pastu. Ja sistēmā pastāv lietotājs ar norādīto e-pastu, tad tiek izvadīts kļūdas paziņojums nr. 3.

Tiek pārbaudīts, vai ievadīta parole satur vismaz vienu mazo burtu, vienu lielo burtu, vienu ciparu un vismaz vienu speciālo simbolu. Ja lietotāja parole neatbilst kaut vienam minētam prasībām, tad tiek izvadīts kļūdas paziņojums nr. 4.

Tiek pārbaudīts, vai ievadīta lietotāju parole sakrīt ar atkārtoti ievadīto lietotāju paroli. Ja tika ievadītas atšķirīgas simbolu virknes (atšķirīgs simbolu skaits, simbolu vērtība noteiktā vietā simbolu virknē, simbolu reģistrs, simbolu valoda), tad tiek izvadīts kļūdas paziņojums nr. 5.

Ja netika izvadīts kļūdas paziņojums, tad lietotāja parole tiek šifrēta ar jaucējfunkcijas palīdzību. Sistēmā tiek izveidots jaunā lietotāja konts un datu bāzē tiek saglabāta neautenticēta lietotāja padotie un apstrādātie personas dati. Tiek izsaukta funkcija 2.2.3.2. *Lietotāju autentifikācija*, kurai tiek padoti izveidota lietotāja konta autentifikācijas/pieteikšanas dati.

Izvaddati

Ja apstrādes gaitā sistēmā tika reģistrēta kļūda, tad tiek izvadīts attiecīgais kļūdas paziņojums.

Ja apstrādes gaitā sistēmā netika reģistrēta kļūda, tad nekas netiks izvadīts, jo apstrādes gaitā tiek izsaukta cita moduļa funkcija.

Kļūdu ziņojumi

1. Lietotājs ar norādītu lietotājvārdu jau pastāv sistēmas ietvaros.
2. E-pasts ir ievadīts nekorekti.
3. Lietotājs ar norādītu e-pastu jau pastāv sistēmas ietvaros
4. Ievadīta parole neatbilst sistēmas prasībām. Paroles simbolu virknē ir nepieciešams ietvert vismaz vienu mazu burtu, vienu lielu burtu, vienu ciparu un vismaz vienu speciālu simbolu.
5. Atkārtoti ievadīta parole nesakrīt ar ievadīto paroli.

2.2.3.2. Lietotāju autentifikācija

Ievads

Funkcija nodrošina iespēju neautenticētam lietotājam pieteikties sistēmai.

Ievaddati

Neautenticētam lietotājam tiek piedāvāta forma, kur ir nepieciešams norādīt sekojošo informāciju:

- Lietotāja pieteikuma vārds. Simbolu virkne, 1-40 simboli. Obligāts.
- Lietotāju parole. Simbolu virkne, 1-255 simbolu. Obligāts.

Apstrāde

Tiek pārbaudīts, vai formas ievadlauki atbilst sistēmas vispārējiem ierobežojumiem, kas ir nodefinēti sekcijā *Ievaddati* un aprakstīti sekcijā *2.2.1.2. Kopīgie kļūdu ziņojumi un ievadlauku ierobežojumi*.

Tiek pārbaudīts vai sistēmā ir reģistrēts lietotājs ar norādīto lietotājvārdu. Ja lietotājs ar norādīto lietotājvārdu nepastāv sistēmā, tad tiek izvadīts kļūdas ziņojums nr. 1.

Ja lietotājs tika atrasts sistēmas ietvaros, tad ievadīta lietotāju parole tiek šifrēta ar jaucējfunkcijas palīdzību un tiek salīdzinātā ar sistēmā glabāto šifrēto paroli. Ja šifrēto simbolu virkņu vērtības nesakrīt, tad tiek izvadīts kļūdas paziņojums nr. 1.

Ja netika izvadīts kļūdas paziņojums, lietotājas veiksmīgi pieticies sistēmai.

Izvaddati

Ja apstrādes gaitā sistēmā tika reģistrēta kļūda, tad tiek izvadīts attiecīgais kļūdas paziņojums.

Ja apstrādes gaitā sistēmā netika reģistrēta kļūda, tad lietotāja sesija krātuvē tiks modificēta sesijas informācija par veiksmīgu pieteikšanos sistēmā un lietotāja tiesībām.

Kļūdu ziņojumi

1. Tika ievadīts nepareizs pieteikuma vārds vai parole.

2.2.3.3 Organizācijas izveidošana/datu rediģēšana

Ievads

Funkcija nodrošina iespēju autenticētam lietotājam izveidot jaunu organizāciju vai rediģēt izveidotas organizācijas informāciju.

Ievaddati

Autentificētam lietotājam tiek piedāvāta forma, kur ir nepieciešams norādīt sekojošu informāciju:

- Organizācijas nosaukums. Simbolu virkne, 1-40 simboli. Obligāts. Unikāls.
- Organizācijas apraksts. Simbolu virkne, 1-1000 simboli. Neobligāts.

Kopā ar formas norādītajiem ievaddatiem funkcijai tiek padoti paslēpti lauki:

- Autentificēta lietotāja identifikators. Vesels skaitlis. Obligāts.

Ja autentificētais lietotājs rediģē eksistējošo organizāciju, tad ievaddatu laukos kā noklusētas vērtības ir iestatītas rediģētas organizācijas aktuāla informācija. Kā arī funkcijai tiek padots sekojošs paslēpts lauks:

- Organizācijas identifikators. Vesels skaitlis. Obligāts.

Apstrāde

Tiek pārbaudīts, vai formas ievadlauki atbilst sistēmas vispārējiem ierobežojumiem, kas ir nodefinēti sekcijā *Ievaddati* un aprakstīti sekcijā *2.2.1.2. Kopīgie kļūdu ziņojumi un ievadlauku ierobežojumi*.

Tiek pārbaudīts, vai sistēmā pastāv organizācija ar norādīto organizācijas nosaukumu. Ja sistēmā pastāv organizācija ar norādīto organizācijas nosaukumu un tā nav šī rediģējamā organizācija, tad tiek izvadīts kļūdas paziņojums nr. 1.

Ja netika izvadīts kļūdas paziņojums un autentificēts lietotājs izvēljas izveidot jaunu organizāciju, tad organizācija tiek izveidota sistēmas ietvaros un datu bāzē tiek izveidots ieraksts ar organizācijas aktuāliem datiem. Par organizāciju īpašnieku tiek uzstādīts autentificēts lietotājs, kurš izveidoja organizāciju.

Ja autentificēts lietotājs izvēljas rediģēt eksistējošas organizācijas informāciju, tad, izmantojot padoto organizācijas identifikatoru, datu bāzē tiek atrasts organizācijas ieraksts. Tiek pārbaudīts, vai autentificētais lietotājs ir atrastas organizācijas īpašnieks. Ja autentificēts lietotājs nav atrastas organizācijas īpašnieks, tad tiek izvadīts kļūdas paziņojums nr. 2.

Ja autentificēts lietotājs ir organizācijas īpašnieks un organizācijas jaunais nosaukums ir unikāls sistēmas ietvaros, tad organizācijas informācija un datu bāzes ieraksts tiek atjaunoti.

Izvaddati

Ja apstrādes gaitā sistēmā tika reģistrēta kļūda, tad tiek izvadīts attiecīgais kļūdas paziņojums.

Ja apstrādes gaitā sistēmā netika reģistrēta kļūda, notiek pāreja uz organizāciju saraksta skatu, izvadot paziņojumu par veiksmīgu organizācijas izveidošanu.

Kļūdu ziņojumi

1. Organizācija ar norādīto organizācijas nosaukumu pastāv sistēmā.
2. Nav tiesību rediģēt organizācijas [Organizācijas nosaukums] datus.

2.2.3.4. Lomas izveidošana/datu rediģēšana

Ievads

Funkcija nodrošina iespēju organizācijas īpašniekam izveidot jaunu organizācijas lomu vai rediģēt izveidotas organizācijas lomas informāciju.

Ievaddati

Organizācijas īpašniekam tiek piedāvāta forma, kur ir nepieciešams norādīt sekojošo informāciju:

- Lomas nosaukums. Simbolu virkne, 1-40 simboli. Obligāts. Unikāls organizācijas ietvaros.
- Lomas apraksts. Simbolu virkne, 1-1000 simboli. Neobligāts.
- Lomas krāsa. Simbolu virkne, 6 simboli. Satur krāsas kodu heksadecimālā formātā. Obligāts.

Kopā ar formas norādītajiem ievaddatiem funkcijai tiek padoti paslēpti lauki:

- Organizācijas identifikators. Vesels skaitlis. Obligāts.

Ja organizācijas īpašnieks rediģē eksistējošo organizācijas lomu, tad ievaddatu laukos kā noklusētas vērtības ir iestatītas rediģētas organizācijas lomas aktuāla informācija. Kā arī funkcijai tiek padots sekojošs paslēpts lauks:

- Lomas identifikators. Vesels skaitlis. Obligāts.

Apstrāde

Tiek pārbaudīts, vai formas ievadlauki atbilst sistēmas vispārējiem ierobežojumiem, kas ir nodefinēti sekcijā *Ievaddati* un aprakstīti sekcijā 2.2.1.2. *Kopīgie kļūdu ziņojumi un ievadlauku ierobežojumi*.

Izmantojot organizācijas identifikatoru, sistēmas datu bāzē tiek atrasts izvēlētas organizācijas ieraksts. Tiek pārbaudīts, vai autentificētais lietotājs ir organizācijas īpašnieks. Ja autentificētais lietotājs nav organizācijas īpašnieks, tad tiek izvadīts kļūdas paziņojums nr. 1.

Ja autentificētais lietotājs ir organizācijas īpašnieks, tiek pārbaudīts, vai izvēlētas organizācijas ietvaros pastāv loma ar norādīto lomas nosaukumu. Ja organizācijas ietvaros pastāv loma ar norādīto nosaukumu un tas nav šī rediģējamā loma, tad tiek izvadīts kļūdas paziņojums nr. 2.

Ja netika izvadīts kļūdas paziņojums un autentificēts lietotājs izvēlējās izveidot jaunu lomu, tad loma tiek izveidota organizācijas ietvaros un datu bāzē tiek izveidots ieraksts ar lomas aktuāliem datiem.

Ja organizācijas īpašnieks izvēlējās rediģēt eksistējošas lomas informāciju, tad, izmantojot padoto lomas identifikatoru, datu bāzē tiek atrasts rediģētās lomas ieraksts. Ja netika izvadīts kļūdas paziņojums, tad lomas informācija un datu bāzes ieraksts tiek atjaunoti.

Izvaddati

Ja apstrādes gaitā sistēmā tika reģistrēta kļūda, tad tiek izvadīts attiecīgais kļūdas paziņojums.

Ja apstrādes gaitā sistēmā netika reģistrēta kļūda, tad notiek pāreja uz lomas dalībnieku piešķiršanas skatu, izvadot paziņojumu par veiksmīgu lomas izveidošanu.

Kļūdu ziņojumi

1. Nav tiesību veidot jaunas lomas organizācijā [Organizācijas nosaukums].
2. Loma ar norādīto lomas nosaukumu eksistē organizācijas ietvaros.

2.2.3.5. Komandas izveidošana/datu rediģēšana

Ievads

Funkcija nodrošina iespēju organizācijas īpašniekam izveidot jaunu organizācijas komandu un dot iespēju izstrādes vadītājam (organizācijas īpašnieks vai komandas vadītājs) rediģēt izveidotas organizācijas komandas informāciju.

Ievaddati

Izstrādes vadītājam tiek piedāvāta forma, kur ir nepieciešams norādīts sekojošo informāciju:

- Komandas nosaukums. Simbolu virkne, 1-40 simboli. Obligāts. Unikāls organizācijas ietvaros.
- Komandas apraksts. Simbolu virkne, 1-1000 simboli. Neobligāts.
- Komandā iesaistītās lomas (lomas identifikatoru masīvs). Veselu skaitļu masīvs. Neobligāts. Formā tiek piedāvāta izvēlne ar organizācijas lomu nosaukumiem.

Organizācijas īpašniekam ir nepieciešams piedāvāt sekojošo informāciju (komandas vadītājam sekojošie ievaddatu lauki tiek paslēpti):

- Komandas vadītāja identifikators. Vesels skaitlis. Obligāts. Formā tiek piedāvāta izvēlne ar organizācijas dalībnieku vārdiem un uzvārdiem.

Kopā ar formas norādītajiem ievaddatiem funkcijai tiek padots paslēpti lauki:

- Organizācijas identifikators. Vesels skaitlis. Obligāts.

Ja izstrādes vadītājs rediģē eksistējošo organizācijas komandu, tad ievaddatu laukos kā noklusētas vērtības ir iestatītas rediģētas organizācijas komandas aktuāla informācija. Kā arī funkcijai tiek padots sekojošs paslēpts lauks:

- Komandas identifikators. Vesels skaitlis. Obligāts.

Apstrāde

Tiek pārbaudīts, vai formas ievadlauki atbilst sistēmas vispārējiem ierobežojumiem, kas ir nodefinēti sekcijā *Ievaddati* un aprakstīti sekcijā *2.2.1.2. Kopīgie kļūdu ziņojumi un ievadlauku ierobežojumi*.

Izmantojot organizācijas identifikatoru, sistēmas datu bāzē tiek atrasts izvēlētas organizācijas ieraksts.

Ja autentificētais lietotājs izvēlējās izveidot jaunu komandu organizācijas ietvaros, tad tiek pārbaudīts, vai autentificētais lietotājs ir organizācijas īpašnieks. Ja autentificētais lietotājs nav organizācijas īpašnieks, tad tiek izvadīts kļūdas paziņojums nr. 1.

Ja autentificētais lietotājs izvēlējās rediģēt eksistējošas komandas informāciju, tad, izmantojot organizācijas komandas identifikatoru, sistēmas datu bāzē tiek atrasts rediģētas

komandas ieraksts. Tiek pārbaudīts, vai autentificētais lietotājs ir organizācijas īpašnieks vai izvēlētas komandas vadītājs. Ja autentificētais nav izstrādes vadītājs rediģētai komandai, tad tiek izvadīts kļūdas paziņojums nr. 2.

Tiek pārbaudīts, vai izvēlētas organizācijas ietvaros pastāv komanda ar norādīto komandas nosaukumu. Ja organizācijā pastāv komanda ar norādīto nosaukumu un tas nav šī rediģējamā komanda, tad tiek izvadīts kļūdas paziņojums nr. 3.

Ja funkcijas parametros tiek padots jauns komandas vadītāja identifikators un autentificētais lietotājs nav organizācijas īpašnieks, tad tiek izvadīts kļūdas paziņojums nr. 4.

Ja netika izvadīts kļūdas paziņojums un autentificētais lietotājs izvēlējās izveidot jaunu komandu, tad komanda tiek izveidota organizācijas ietvaros un datu bāzē tiek izveidots ieraksts ar komandas aktuāliem datiem. Tiek izveidots *2.2.5.1. Lietotāja paziņojuma izveide* funkcijas izsaukums, kuram tiek padots izvēlētais komandas vadītāja identifikators, izveidotas komandas identifikators un paziņojuma informācija par jauno komandas vadītāju izvēli.

Ja izstrādes vadītājs izvēlējās rediģēt eksistējošas komandas informāciju, tad, izmantojot padoto komandas identifikatoru, datu bāzē tiek atrasts rediģētās komandas ieraksts. Ja netika izvadīts kļūdas paziņojums, tad komandas informācija un datu bāzes ieraksts tiek atjaunoti. Ja komandu vadītāja identifikators tika mainīts, tad tiek izveidots *2.2.5.1. Lietotāja paziņojuma izveide* funkcijas izsaukums, kuram tiek padots jauns komandas vadītāja identifikators, izveidotas komandas identifikators un paziņojuma informācija par jauno komandas vadītāju izvēli.

Izvaddati

Ja apstrādes gaitā sistēmā tika reģistrēta kļūda, tad tiek izvadīts attiecīgais kļūdas paziņojums.

Ja apstrādes gaitā sistēmā netika reģistrēta kļūda, tad notiek pāreja uz komandas dalībnieku piešķiršanas skatu, izvadot paziņojumu par veiksmīgu komandas izveidošanu.

Kļūdu ziņojumi

1. Nav tiesību veidot jaunas komandas organizācijā [Organizācijas nosaukums].
2. Nav tiesību rediģēt izvēlēto komandu: [Komandas nosaukums].
3. Komanda ar norādīto komandas nosaukumu eksistē organizācijas ietvaros.
4. Nav tiesību izvēlēties komandas vadītāju.

2.2.3.6. Lomu piešķiršana

Ievads

Funkcija nodrošina iespēju organizācijas īpašniekam piešķirt organizācijas lomu noteiktiem organizācijas dalībniekiem.

Ievaddati

Organizācijas īpašniekam tiek piedāvāta forma, kur ir nepieciešams norādīts sekojošo informāciju:

- Lomas identifikators. Vesels skaitlis. Obligāts. Formā tiek piedāvāta izvēlne ar organizācijas lomu nosaukumiem.
- Organizācijas dalībnieki (lietotāju identifikatoru masīvs). Veselu skaitļu masīvs. Obligāts. Formā tiek piedāvāta izvēlne ar organizācijas dalībnieku vārdiem un uzvārdiem. Izvēlnē jau tiek izvēlēti organizācijas dalībnieki, kuriem izvēlēta loma ir piešķirta.

Kopā ar formas norādītajiem ievaddatiem funkcijai tiek padots paslēpti lauki:

- Organizācijas identifikators. Vesels skaitlis. Obligāts.

Apstrāde

Tiek pārbaudīts, vai formas ievadlauki atbilst sistēmas vispārējiem ierobežojumiem, kas ir nedefinēti sekcijā *Ievaddati* un aprakstīti sekcijā *2.2.1.2. Kopīgie kļūdu ziņojumi un ievadlauku ierobežojumi*.

Izmantojot organizācijas identifikatoru, sistēmas datu bāzē tiek atrasts izvēlētas organizācijas ieraksts. Tiek pārbaudīts, vai autentificētais lietotājs ir organizācijas īpašnieks. Ja autentificētais lietotājs nav organizācijas īpašnieks, tad tiek izvadīts kļūdas paziņojums nr. 1.

Tiek pārbaudīts vai izvēlēta loma pastāv organizācijas ietvaros. Ja izvēlēta loma neeksistē organizācijas lomu sarakstā, tad tiek izvadīts kļūdas paziņojums nr. 2.

Katram lietotāja identifikatoram sistēmas datu bāzē tiek atrasts attiecīgai lietotājs un tiek pārbaudīts vai katrs sistēmas lietotājs ir organizācijas dalībnieks. Ja kaut viens lietotājs nav izvēlētas organizācijas dalībnieks, tad tiek izvadīts kļūdas paziņojums nr. 3.

Ja netika izvadīts kļūdas paziņojums, tad katrs izvēlētais organizācijas dalībniekam tiek piešķirta izvēlēta organizācijas loma. Ja organizācijas dalībniekam loma nebija piešķirta, tad

tiek izveidots 2.2.5.1. *Lietotāja paziņojuma izveide* funkcijas izsaukums, kuram tiek padoti lietotāja identifikators, lomas dati un paziņojuma informācija par lomas piešķiršanu.

Ja organizācijas dalībnieks netika iekļauts padotā lietotāju sarakstā un lietotājam ir piešķirta izvēlēta loma, tad saistība starp lietotāju un lomu tiek dzēsta. Tiek izveidots 2.2.5.1. *Lietotāja paziņojuma izveide* funkcijas izsaukums, kur tiek padoti lietotāja identifikators, lomas dati un paziņojuma dati par lomas saistības dzēšanu.

Izvaddati

Ja apstrādes gaitā sistēmā tika reģistrēta kļūda, tad tiek izvadīts attiecīgais kļūdas paziņojums.

Ja apstrādes gaitā sistēmā netika reģistrēta kļūda, tad tiek izvadīts paziņojums par veiksmīgu lomas dalībnieku modificēšanu.

Kļūdu ziņojumi

1. Nav tiesību organizācijas dalībniekiem piešķirt lomas organizācijā [Organizācijas nosaukums].
2. Izvēlēta loma nepastāv organizācijas ietvaros.
3. Lietotājs [Lietotājs vārds] [Lietotāja uzvārds] nav organizācijas [Organizācijas nosaukums] dalībnieks.

2.2.3.7. Komandas piešķiršana

Ievads

Funkcija nodrošina iespēju organizācijas īpašniekam piešķirt organizācijas komandu noteiktiem organizācijas dalībniekiem. Organizācijas dalībnieki kļūst par izvēlētas komandas dalībniekiem.

Ievaddati

Organizācijas īpašniekam tiek piedāvāta forma, kur ir nepieciešams norādīts sekojošo informāciju:

- Komandas identifikators. Vesels skaitlis. Obligāts. Formā tiek piedāvāta izvēlne ar organizācijas komandu nosaukumiem.
- Organizācijas dalībnieki (lietotāju identifikatoru masīvs). Veselu skaitļu masīvs. Obligāts. Formā tiek piedāvāta izvēlne ar organizācijas dalībnieku vārdiem un

uzvārdiem. Izvēlnē jau tiek izvēlēti organizācijas dalībnieki, kuri ir izvēlētas komandas dalībnieki.

Kopā ar formas norādītajiem ievaddatiem funkcijai tiek padots paslēpti lauki:

- Organizācijas identifikators. Vesels skaitlis. Obligāts.

Apstrāde

Tiek pārbaudīts, vai formas ievadlauki atbilst sistēmas vispārējiem ierobežojumiem, kas ir nodefinēti sekcijā *Ievaddati* un aprakstīti sekcijā *2.2.1.2. Kopīgie kļūdu ziņojumi un ievadlauku ierobežojumi*.

Izmantojot organizācijas identifikatoru, sistēmas datu bāzē tiek atrasts izvēlētas organizācijas ieraksts. Tiek pārbaudīts, vai autentificētais lietotājs ir organizācijas īpašnieks. Ja autentificētais lietotājs nav organizācijas īpašnieks, tad tiek izvadīts kļūdas paziņojums nr. 1.

Tiek pārbaudīts vai izvēlēta komanda pastāv organizācijas ietvaros. Ja izvēlēta komanda neeksistē organizācijas komandu sarakstā, tad tiek izvadīts kļūdas paziņojums nr. 2.

Katram lietotāja identifikatoram sistēmas datu bāzē tiek atrasts attiecīgai lietotājs un tiek pārbaudīts vai katrs sistēmas lietotājs ir organizācijas dalībnieks. Ja kaut viens lietotājs nav izvēlētas organizācijas dalībnieks, tad tiek izvadīts kļūdas paziņojums nr. 3.

Ja netika izvadīts kļūdas paziņojums, tad katrs izvēlētais organizācijas dalībniekam tiek piešķirta izvēlēta organizācijas komanda. Ja organizācijas dalībniekam komanda nebija piešķirta, tad tiek izveidots *2.2.5.1. Lietotāja paziņojuma izveide* funkcijas izsaukums, kur tiek padoti lietotāja identifikators, komandas dati un paziņojuma informācija par komandas piešķiršanu.

Ja organizācijas dalībnieks netika iekļauts padotā lietotāju sarakstā un lietotājam ir piešķirta izvēlēta komanda, tad saistība starp lietotāju un komandu tiek dzēsta. Tiek izveidots *2.2.5.1. Lietotāja paziņojuma izveide* funkcijas izsaukums, kur tiek padoti lietotāja identifikators, komandas dati un paziņojuma informācija par komandas saistības dzēšanu.

Izvaddati

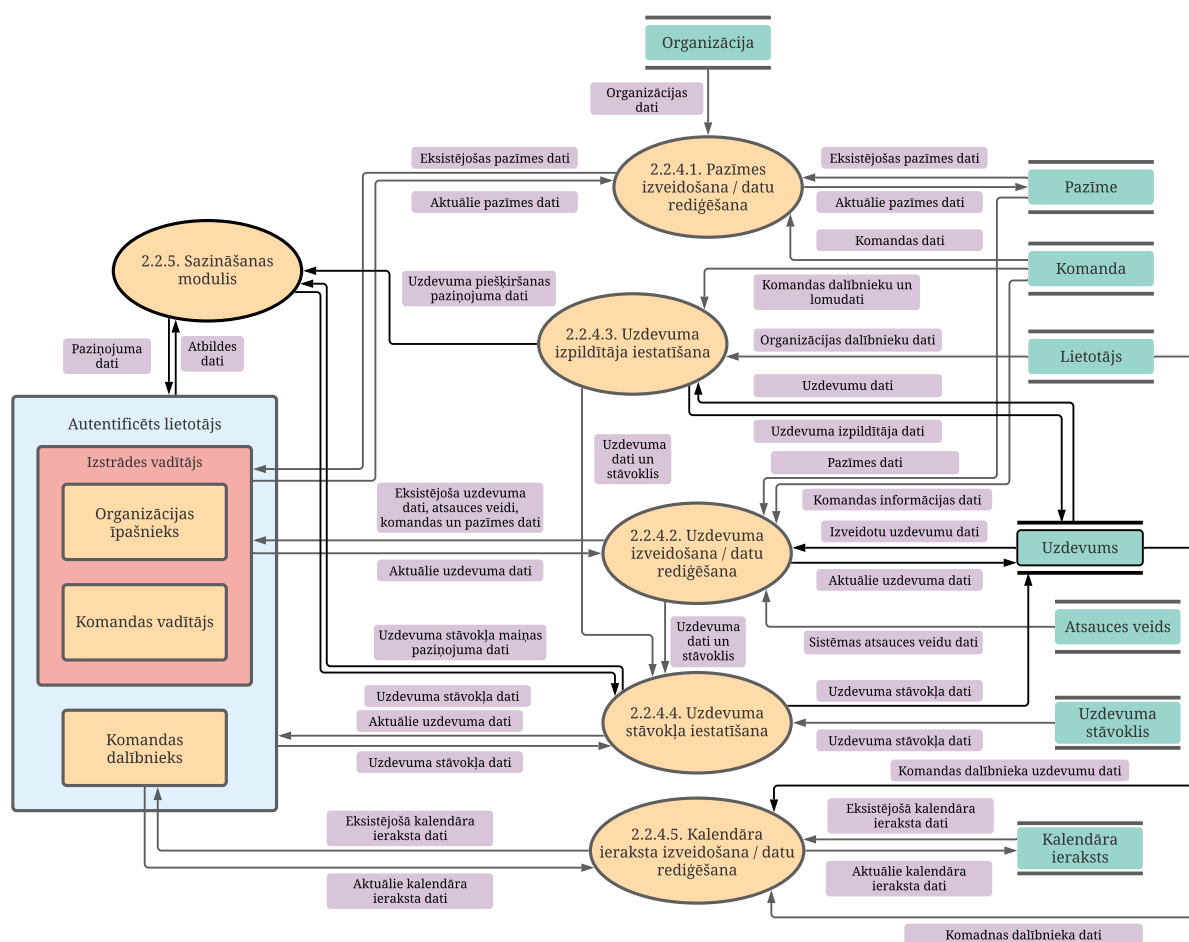
Ja apstrādes gaitā sistēmā tika reģistrēta kļūda, tad tiek izvadīts attiecīgais kļūdas paziņojums.

Ja apstrādes gaitā sistēmā netika reģistrēta kļūda, tad tiek izvadīts paziņojums par veiksmīgu komandas dalībnieku modificēšanu.

Kļūdu ziņojumi

1. Nav tiesību organizācijas dalībniekiem piešķirt komandas organizācijā [Organizācijas nosaukums].
2. Izvēlēta komanda nepastāv organizācijas ietvaros.
3. Lietotājs [Lietotājs vārds] [Lietotāja uzvārds] nav organizācijas [Organizācijas nosaukums] dalībnieks.

2.2.4. Izstrādes procesa vadības modulis



2.4. att. 2. līmeņa datu plūsmas diagramma izstrādes procesa vadības moduļim

Izstrādes procesa vadības moduļa datu plūsmas diagrammā (sk. 2.4. att.) tiek attēlotas datu plūsmas starp sistēmas lietotājiem, moduļa funkcijām un entītijām, kas tiek izmantoti moduļa ietvaros. Modulis izmanto entītijas *Organizācija*, *Pazīme*, *Komanda*, *Lietotājs*, *Uzdevums*, *Atsauces veids*, *Uzdevuma stāvoklis*, *Kalendāra ieraksts*. Moduļa ietvaros tiek izstrādāta funkcionalitāte, kurā sevī ietver pazīmes, uzdevuma un kalendārā ieraksta izveidošanu, uzdevuma izpildītāja izvēli un uzdevuma stāvokļa maiņu ar turpmāku paziņojuma

izveidi izstrādes vadītājam. Diagrammā papildus tiek attēlotas datu plūsmas, kas ir saistītas ar sazināšanas moduļa funkcijām (sk. 2.2.5. *Sazināšanas modulis*).

2.2.4.1. Pazīmes izveidošana/datu rediģēšana

Ievads

Funkcija nodrošina iespēju izstrādes vadītājam (organizācijas īpašniekam vai noteiktas komandas vadītājam) izveidot vai rediģēt pazīmi noteiktas komandas izstrādes procesam.

Ievaddati

Izstrādes vadītājam tiek piedāvāta forma, kur ir nepieciešams norādīt sekojošu informāciju:

- Pazīmes kods. Simbolu virkne, 1-8 simboli. Obligāts. Unikāls izvēlētas komandas ietvaros
- Pazīmes nosaukums. Simbolu virkne, 1-40 simboli. Obligāts.
- Pazīmes apraksts. Simbolu virkne, 1-10000 simboli. Neobligāts.
- Pazīmes izstrādes sākuma datums. Datuma vērtība. Obligāts.
- Pazīmes izstrādes ilgums darbadienās. Vesels skaitlis. Obligāts.

Kopā ar formas norādītajiem ievaddatiem funkcijai tiek padots paslēpti lauki:

- Organizācijas identifikators. Vesels skaitlis. Obligāts.
- Izvēlētas komandas identifikators. Vesels skaitlis. Obligāts. Izstrādes vadītājam ir iespējams piekļūt uzdevuma izveides formā tikai no organizācijas komandas pārvaldības skata (sk. 3.3.1. *Lietotāju saskarnes pārvietošanas diagramma* nodaļu). Tātad formā pastāv izvēlētas komandas identifikators.

Ja izstrādes vadītājs rediģē eksistējošo komandas pazīmi, tad ievaddatu laukos kā noklusētas vērtības ir iestatītas rediģētas komandas pazīmes aktuāla informācija. Kā arī funkcijai tiek padots sekojošs paslēpts lauks:

- Pazīmes identifikators. Vesels skaitlis. Obligāts.

Apstrāde

Tiek pārbaudīts, vai formas ievadlauki atbilst sistēmas vispārējiem ierobežojumiem, kas ir nodefinēti sekcijā *Ievaddati* un aprakstīti sekcijā 2.2.1.2. *Kopīgie kļūdu ziņojumi un ievadlauku ierobežojumi*.

Izmantojot organizācijas identifikatoru, sistēmas datu bāzē tiek atrasts izvēlētas organizācijas ieraksts. Tiek pārbaudīts, vai izvēlēta komanda pastāv organizācijas ietvaros. Ja izvēlēta komanda nepastāv organizācijas ietvaros, tad tiek izvadīts kļūdas paziņojums nr. 1.

Tiek pārbaudīts, vai autentificētais lietotājs ir organizācijas īpašnieks vai izvēlētas komandas vadītājs. Ja autentificētais lietotājs nav organizācijas īpašnieks un nav izvēlētas komandas vadītājs, tad tiek izvadīts kļūdas paziņojums nr. 2.

Tiek pārbaudīts, vai izvēlētas komandas izstrādes procesa ietvaros pastāv pazīme ar norādīto pazīmes kodu. Ja izvēlētas komandas izstrādes procesa ietvaros pastāv pazīme ar norādīto pazīmes kodu, tad tiek izvadīts kļūdas paziņojums nr. 3.

Tiek pārbaudīt, vai pazīmes izstrādes ilguma vērtība ir vesels pozitīvs skaitlis. Ja pazīmes izstrādes ilguma vērtība nav vesels pozitīvs skaitlis, tad tiek izvadīts kļūdas paziņojums nr. 4.

Ja netika izvadīts kļūdas paziņojums, tad:

- Ja autentificētais lietotājs izvēlējās izveidot jaunu pazīmi, tad pazīme tiek izveidota izvēlētas komandas izstrādes procesa ietvaros un datu bāzē tiek izveidots ieraksts ar pazīmes aktuāliem datiem.
- Ja izstrādes vadītājs izvēlējās rediģēt eksistējošas pazīmes informāciju, tad, izmantojot padoto pazīmes identifikatoru, datu bāzē tiek atrasts rediģētas pazīmes ieraksts. Ja netika izvadīts kļūdas paziņojums, tad pazīmes informācija un datu bāzes ieraksts tiek atjaunoti.

Izvaddati

Ja apstrādes gaitā sistēmā tika reģistrēta kļūda, tad tiek izvadīts attiecīgais kļūdas paziņojums.

Ja apstrādes gaitā sistēmā netika reģistrēta kļūda, tad tiek izvadīts paziņojums par veiksmīgu pazīmes izveidošanu.

Kļūdu ziņojumi

1. Izvēlēta komanda nepastāv organizācijas ietvaros.
2. Nav tiesību veidot jaunu pazīmi izvēlētai komandai: [Komandas nosaukums]

3. Pazīme ar norādīto pazīmes kodu eksistē izvēlētas komandas izstrādes procesa ietvaros.
4. Pazīmes izstrādes laiks darbadienās ir vesels pozitīvs skaitlis.

2.2.4.2. Uzdevuma izveidošana/datu rediģēšana

Ievads

Funkcija nodrošina iespēju izstrādes vadītājam (organizācijas īpašniekam vai noteiktas komandas vadītājam) izveidot vai rediģēt uzdevumu noteiktas komandas izstrādes procesam un izvēlētas pazīmes ietvaros.

Ievaddati

Izstrādes vadītājam tiek piedāvāta forma, kur ir nepieciešams norādīt sekojošu informāciju:

- Uzdevuma kods. Simbolu virkne, 1-12 simboli. Obligāts. Unikāls izvēlētas komandas ietvaros. Pēc pazīmes izvēles pazīmes kods tiek iestatīts kā koda priedēklis, bet lietotājam ir iespējams izmainīt noklusēto vērtību.
- Uzdevuma nosaukums. Simbolu virkne, 1-255 simboli. Obligāts.
- Uzdevuma apraksts, 1-10000 simboli. Neobligāts.
- Veidojot vai rediģējot uzdevumu ir iespējams izveidot atsauces uz eksistējošo komandas uzdevumu, kura atrodas izvēlētas pazīmes ietvaros. Atsauces nav obligātas. Lai izveidotu uzdevuma atsauci ir nepieciešams norādīt sekojošus laukus katrai izveidotai atsaucēi:
 - Atsauces uzdevuma identifikators. Vesels skaitlis. Obligāts. Formā tiek piedāvāta izvēlne ar komandas uzdevumiem izvēlētas pazīmes ietvaros.
 - Atsauces veida identifikators. Vesels skaitlis. Obligāts. Formā tiek piedāvāta izvēlne ar sistēmas nedefinētiem atsauces veidiem.
 - Atsauces apraksts. Simbolu virkne, 1-1000 simboli. Neobligāts.

Kopā ar formas norādītajiem ievaddatiem funkcijai tiek padots paslēpti lauki:

- Organizācijas identifikators. Vesels skaitlis. Obligāts.

- Izvēlētas komandas identifikators. Vesels skaitlis. Obligāts. Izstrādes vadītājam ir iespējams piekļūt uzdevuma izveides formā tikai no organizācijas komandas pārvaldības skata (sk. 3.3.1. *Lietotāju saskarnes pārvietošanas diagramma* nodaļu). Tātad formā pastāv izvēlētas komandas identifikators.
- Izvēlētas pazīmes identifikators. Vesels skaitlis. Obligāts. Līdzīgi kā komandas identifikatoram, izstrādes vadītājam ir iespējams piekļūt uzdevuma izveides formā tikai izvēlētas pazīmes ietvaros.

Ja izstrādes vadītājs redīgē eksistējošo komandas uzdevumu, tad ievaddatu laukos kā noklusētas vērtības ir iestatītas redīgētas komandas pazīmes aktuāla informācija. Un priekš katras uzdevumu atsauces tiek izveidots ieraksts uzdevumu atsauces formas sekcijā (sk. 3.3.2.2. *Komandas uzdevuma veidošanas/redīgēšanas skats* sekciju 13. skata sastāvdaļu). Kā arī funkcijai tiek padots sekojošs paslēpts lauks:

- Uzdevuma identifikators. Vesels skaitlis. Obligāts.

Apstrāde

Tiek pārbaudīts, vai formas ievadlauki atbilst sistēmas vispārējiem ierobežojumiem, kas ir nedefinēti sekcijā *Ievaddati* un aprakstīti sekcijā 2.2.1.2. *Kopīgie kļūdu ziņojumi un ievadlauku ierobežojumi*.

Izmantojot organizācijas identifikatoru, sistēmas datu bāzē tiek atrasts izvēlētas organizācijas ieraksts. Tiek pārbaudīts, vai izvēlēta komanda pastāv organizācijas ietvaros. Ja izvēlēta komanda nepastāv organizācijas ietvaros, tad tiek izvadīts kļūdas paziņojums nr. 1.

Tiek pārbaudīts, vai autentificētais lietotājs ir organizācijas īpašnieks vai izvēlētas komandas vadītājs. Ja autentificētais lietotājs nav organizācijas īpašnieks un nav izvēlētas komandas vadītājs, tad tiek izvadīts kļūdas paziņojums nr. 2.

Izmantojot pazīmes identifikatoru, sistēmas datu bāzē tiek atrasts izvēlētas pazīmes ieraksts. Tiek pārbaudīts, vai izvēlēta pazīmē pastāv izvēlētas komandas izstrādes procesa ietvaros. Ja izvēlēta pazīme nepastāv komandas izstrādes procesa ietvaros, tad tiek izvadīts kļūdas paziņojums nr. 3.

Tiek pārbaudīts, vai izvēlētas komandas izstrādes procesa ietvaros pastāv uzdevums ar norādīto uzdevuma kodu. Ja izvēlētas komandas izstrādes procesa ietvaros pastāv pazīme ar norādīto uzdevuma kodu, tad tiek izvadīts kļūdas paziņojums nr. 4.

Ja tika norādītas uzdevuma atsauces, tad katrai atsaucei tiek pārbaudīts, vai izvēlētais atsauces uzdevums pastāv izvēlētas pazīmes ietvaros. Ja izvēlētais atsauces uzdevums neeksistē pazīmes ietvaros, tad tiek izvadīts kļūdas paziņojums nr. 5.

Katrai izveidotai uzdevumu atsaucei tiek pārbaudīts, vai izvēlētais atsauces veids pastāv sistēmas ietvaros. Ja izvēlētais atsauces veids nepastāv sistēmas ietvaros, tad tiek izvadīts kļūdas paziņojums nr. 6.

Ja netika izvadīts kļūdas paziņojums un autentificētais lietotājs izvēlējās izveidot jaunu uzdevumu, tad uzdevums tiek izveidots izvēlētas pazīmes un izvēlētas komandas izstrādes procesa ietvaros un datu bāzē tiek izveidots ieraksts ar uzdevuma aktuāliem datiem.

Ja izstrādes vadītājs izvēlējās rediģēt eksistējoša uzdevuma informāciju, tad, izmantojot padoto uzdevuma identifikatoru, datu bāzē tiek atrasts rediģēta uzdevuma ieraksts. Ja netika izvadīts kļūdas paziņojums, tad uzdevuma informācijas un datu bāzes ieraksts tiek atjaunoti.

Ja izveidotam vai rediģētam uzdevumam pastāv uzdevumu atsauce ar veidu *Ķēdes veidošanas atsauce* un kaut vienam atsauces uzdevumam nav uzdevuma stāvoklis *Aizvērts*, tad tiek izveidots 2.2.4.4. *Uzdevuma stāvokļa iestatīšana* funkcijas izsaukums, kuram tiek padots izveidotā vai rediģētā uzdevuma identifikators un uzdevuma stāvokļa identifikators, kas atbilst uzdevuma stāvoklim *Saistīts*.

Ja izveidotam uzdevumam nepastāv uzdevumu atsauce ar veidu *Ķēdes veidošanas atsauce*, tad tiek izveidots 2.2.4.4. *Uzdevuma stāvokļa iestatīšana* funkcijas izsaukums, kuram tiek padots izveidotā vai rediģētā uzdevuma identifikators un uzdevuma stāvokļa identifikators, kas atbilst uzdevuma stāvoklim *Izpildītāja piešķiršana*.

Ja izveidotam uzdevumam nepastāv uzdevumu atsauce ar veidu *Ķēdes veidošanas atsauce*, tad uzdevuma stāvoklis netiek mainīts.

Izvaddati

Ja apstrādes gaitā sistēmā tika reģistrēta kļūda, tad tiek izvadīts attiecīgais kļūdas paziņojums.

Ja apstrādes gaitā sistēmā netika reģistrēta kļūda, tad notiek pāreja uz uzdevuma izpildītāja piešķiršanas skatu, izvadot paziņojumu par veiksmīgu uzdevuma izveidošanu.

Kļūdu ziņojumi

1. Izvēlēta komanda nepastāv organizācijas ietvaros.

2. Nav tiesību veidot jaunu uzdevumu izvēlētai komandai: [Komandas nosaukums]
3. Izvēlēta pazīme nepastāv komandas izstrādes procesa ietvaros.
4. Uzdevums ar norādīto uzdevuma kodu eksistē izvēlētas komandas izstrādes procesa ietvaros.
5. Izvēlētais atsauces uzdevums neeksistē izvēlētas pazīmes ietvaros
6. Izvēlētais uzdevumu atsauces veids neeksistē sistēmas ietvaros

2.2.4.3. Uzdevuma izpildītāja iestatīšana

Ievads

Funkcija nodrošina iespēju izstrādes vadītājam (organizācijas īpašniekam vai noteiktas komandas vadītājam) iestatīt izvēlētam uzdevumam izpildītāju.

Ievaddati

Izstrādes vadītājam tiek piedāvāta forma, kad sistēmā tika reģistrēts jauns uzdevums vai izstrādes vadītājs izvēlējās rediģēt noteikta uzdevuma informāciju. Formā ir nepieciešams norādīt sekojošu informāciju:

Lietotāja vai lomas identifikators. Vesels skaitlis. Obligāts. Formā tiek piedāvāta izvēlne ar komandas dalībniekiem un komandas lomām, kuru komandai ir piešķirts attiecīgais uzdevums. Ja uzdevumam pastāv uzdevuma izpildītājs, tad ievadlaukā tiek ievietota uzdevuma izpildītāja aktuālā informācija.

Kopā ar formas norādītajiem ievaddatiem funkcijai tiek padots paslēpti lauki:

- Organizācijas identifikators. Vesels skaitlis. Obligāts.
- Uzdevuma identifikators. Vesels skaitlis. Obligāts.
- Izpildītāja veids. Simbolu virkne, 1-40 simboli. Obligāts. Ja tika izvēlēts komandas dalībnieks, lauks tiek automātiski aizpildīts ar simbolu virkni *Persona*. Ja tika izvēlēts komandas loma, lauks tiek automātiski aizpildīts ar simbolu virkni *Loma*.

Apstrāde

Tiek pārbaudīts, vai formas ievadlauki atbilst sistēmas vispārējiem ierobežojumiem, kas ir nodefinēti sekcijā *Ievaddati* un aprakstīti sekcijā 2.2.1.2. *Kopīgie kļūdu ziņojumi un ievadlauku ierobežojumi*.

Izmantojot organizācijas identifikatoru, sistēmas datu bāzē tiek atrasts izvēlētas organizācijas ieraksts un, izmantojot uzdevuma identifikatoru, sistēmas datu bāzē tiek atrasts uzdevuma ieraksts. Tiek pārbaudīts, vai autentificētais lietotājs ir organizācijas īpašnieks vai komandas vadītājs komandai, kurai ir piešķirts attiecīgais uzdevums. Ja autentificētais lietotājs nav organizācijas īpašnieks vai komandas vadītājs, tad tiek izvadīts kļūdas paziņojums nr. 1.

Ja izpildītāja veids ir *Persona*, tad piedāvātam lietotāja identifikatoram sistēmas datu bāzē tiek atrasts attiecīgais lietotājs un tiek pārbaudīts vai sistēmas lietotājs ir komandas dalībnieks. Ja lietotājs nav komandas dalībnieks, tad tiek izvadīts kļūdas paziņojums nr. 2.

Ja izpildītāja veids ir *Loma*, tad piedāvātam lietotāja identifikatoram sistēmas datu bāzē tiek atrasta attiecīgā loma un tiek pārbaudīts vai loma ir komandas lomu kopā, un komandā pastāv vismaz viens dalībnieks ar norādīto lomu. Ja piedāvāta loma nav komandas lomu kopā, tad tiek izvadīts kļūdas paziņojums nr. 3. Ja izvēlēta komandā nav neviena komandas dalībnieka ar piedāvāto lomu, tad tiek izvadīts kļūdas paziņojums nr. 4.

Ja netika izvadīts kļūdas paziņojums un pirms apstrādes uzdevumam tika iestatīts uzdevuma izpildītājs ar izpildītāja veidu *Persona*, tiek pārbaudīts vai padotais uzdevuma izpildītājs atšķiras no esoša izpildītāja. Ja uzdevuma izpildītāji atšķiras, tad tiek izveidots funkcijas 2.2.5.1. *Lietotāja paziņojuma izveide* izsaukums, kuram tiek padoti esoša uzdevumu izpildītāja lietotāja identifikators, uzdevuma identifikators un paziņojums par uzdevuma izpildīšanas atcelšanu. Tiek izveidots funkcijas 2.2.4.4. *Uzdevuma stāvokļa iestatīšana* izsaukums, kuram tiek padots uzdevuma identifikators un uzdevuma stāvokļa identifikators, kas atbilst uzdevuma stāvoklim *Piešķiršana*.

Ja netika izvadīts kļūdas paziņojums un izpildītāja veids ir *Persona*, tad tiek izveidots funkcijas 2.2.5.1. *Lietotāja paziņojuma izveide* izsaukums, kuram tiek padoti lietotāja identifikators, uzdevuma identifikators un paziņojums par uzdevuma piešķiršanu. Ja uzdevumam pirms apstrādes netika pievienots uzdevuma izpildītājs, tad tiek izveidots funkcijas 2.2.4.4. *Uzdevuma stāvokļa iestatīšana* izsaukums, kuram tiek padots uzdevuma identifikators un uzdevuma stāvokļa identifikators, kas atbilst uzdevuma stāvoklim *Piešķirts*. Kā uzdevumu izpildītājs uzdevumam tiek iestatīts padotais komandas dalībnieka identifikators.

Ja netika izvadīts kļūdas paziņojums un izpildītāja veids ir *Loma*, tad tiek izveidots funkcijas 2.2.5.2. *Komandas lomas paziņojuma izveide* izsaukums, kuram tiek padoti lomas identifikators, uzdevuma identifikators un paziņojums par uzdevuma piešķiršanu. Kā uzdevumu izpildītājs uzdevumam tiek iestatīts padotais komandas lomas identifikators.

Izvaddati

Ja apstrādes gaitā sistēmā tika reģistrēta kļūda, tad tiek izvadīts attiecīgais kļūdas paziņojums.

Ja apstrādes gaitā sistēmā netika reģistrēta kļūda, tad tiek izvadīts paziņojums par veiksmīgu uzdevuma izpildītāju modificēšanu.

Kļūdu ziņojumi

1. Nav tiesību uzstādīt uzdevumu izpildītājus komandai: [Komandas nosaukums]
2. Izvēlētais lietotājs nav komandas [Komandas nosaukums] dalībnieks.
3. Izvēlēta loma nepastāv komandas [Komandas nosaukums] lomu kopā.
4. Komandā [Komandas nosaukums] nepastāv komandas dalībnieku ar izvēlēto lomu.

2.2.4.4. Uzdevuma stāvokļa iestatīšana

Ievads

Funkcija nodrošina iespēju izstrādes dalībniekam (organizācijas īpašniekam, noteiktas komandas vadītājam vai komandas dalībniekam) iestatīt izvēlētam uzdevumam noteiktu uzdevuma stāvokli.

Ievaddati

Funkcija tiek izsaukta citu funkciju apstrādes gaitā vai lietojot tīmekļa vietnes saskarni. Forma netiek izveidota. Funkcijai ir nepieciešams padot sekojošu informāciju:

- Uzdevuma identifikators. Vesels skaitlis. Obligāts.
- Uzdevuma stāvokļa identifikators. Vesels skaitlis. Obligāts.

Apstrāde

Tiek pārbaudīts, vai ievaddati atbilst sistēmas vispārējiem ierobežojumiem, kas ir nedefinēti sekcijā *Ievaddati* un aprakstīti sekcijā 2.2.1.2. *Kopīgie kļūdu ziņojumi un ievadlauku ierobežojumi*.

Izmantojot uzdevuma identifikatoru, sistēmas datu bāzē tiek atrasts uzdevuma ieraksts un, izmantojot uzdevuma stāvokļa identifikatoru, sistēmas datu bāzē tiek atrasts uzdevuma stāvokļa ieraksts.

Tiek pārbaudīts, vai autentificētais lietotājs ir organizācijas īpašnieks vai komandas dalībnieks, kuru komandai ir piešķirts attiecīgais uzdevums. Ja autentificētais lietotājs nav organizācijas īpašnieks vai komandas dalībnieks, tad tiek izvadīts kļūdas paziņojums nr. 1.

Ja padotais uzdevuma stāvoklis ir *Izpildītāja piešķiršana*, tad:

- Tiek pārbaudīts, vai uzdevuma stāvoklis ir *Izpildītāja piešķiršana*, *Saistīts*, *Piešķirts* vai uzdevumam nav uzdevuma stāvoklis. Ja uzdevumam nav neviens no minētiem stāvokļiem un stāvoklis pastāv, tad tiek izvadīts kļūdas paziņojums nr. 2.
- Tiek pārbaudīts vai autentificētais lietotājs ir izstrādes vadītājs. Ja autentificētais lietotājs nav izstrādes vadītājs, tiek izvadīts kļūdas paziņojums nr. 1.
- Ja netika izvadīts kļūdas paziņojums, tad:
 - Tiek pārbaudīts, vai uzdevuma stāvoklis ir *Saistīts*. Ja uzdevuma stāvoklis ir *Saistīts*, tad uzdevuma stāvoklis netiek mainīts.
 - Ja uzdevumam nav stāvoklis *Saistīts*, tad izvēlam uzdevumam tiek piešķirts stāvoklis *Izpildītāja piešķiršana*.

Ja padotais uzdevuma stāvoklis ir *Saistīts*, tad:

- Tiek pārbaudīts, vai uzdevuma stāvoklis ir *Izpildītāja piešķiršana*, *Saistīts* vai *Piešķirts*. Ja uzdevumam nav neviens no minētiem stāvokļiem, tad tiek izvadīts kļūdas paziņojums nr. 2.
- Tiek pārbaudīts vai autentificētais lietotājs ir izstrādes vadītājs. Ja autentificētais lietotājs nav izstrādes vadītājs, tiek izvadīts kļūdas paziņojums nr. 1.
- Ja netika izvadīts kļūdas paziņojums, tad izvēlam uzdevumam tiek piešķirts stāvoklis *Saistīts*.

Ja padotais uzdevuma stāvoklis ir *Piešķirts*, tad:

- Tiek pārbaudīts, vai uzdevuma stāvoklis ir *Izpildītāja piešķiršana*, *Saistīts* vai *Piešķirts*. Ja uzdevum nav neviens no minētiem stāvokļiem, tad tiek izvadīts kļūdas paziņojums nr. 2.
- Tiek pārbaudīts vai autentificētais lietotājs ir izstrādes vadītājs. Ja autentificēts lietotājs nav izstrādes vadītājs, tiek izvadīts kļūdas paziņojums nr. 1.
- Ja netika izvadīts kļūdas paziņojums, tad:
 - Tiek pārbaudīts, vai uzdevuma stāvoklis ir *Saistīts*. Ja uzdevuma stāvoklis ir *Saistīts*, tad uzdevuma stāvoklis netiek mainīts.
 - Ja uzdevumam nav stāvoklis *Saistīts*, tad izvēlam uzdevumam tiek piešķirts stāvoklis *Piešķirts*.

Ja padotais uzdevuma stāvoklis ir *Aktīvs*, tad:

- Tiek pārbaudīts, vai uzdevumam ir stāvoklis *Piešķirts* vai *Aktīvs*. Ja uzdevum nav neviens no minētiem stāvokļiem, tad tiek izvadīts kļūdas paziņojums nr. 3.
- Tiek pārbaudīts, vai autentificētais lietotājs ir uzdevuma izpildītājs. Ja autentificētais lietotājs nav uzdevumu izpildītājs, tad tiek izvadīts kļūdas paziņojums nr. 4.
- Ja netika izvadīts kļūdas paziņojums, tad izvēlam uzdevumam tiek piešķirts stāvoklis *Aktīvs*.

Ja padotais uzdevuma stāvoklis ir *Pārskats*, tad:

- Tiek pārbaudīts, vai uzdevumam ir stāvoklis *Aktīvs*. Ja uzdevumam nav stāvoklis *Aktīvs*, tad tiek izvadīts kļūdas paziņojums nr. 5.
- Ja netika izvadīts kļūdas paziņojums, tad:
 - Tiek izveidots funkcijas 2.2.5.1. *Lietotāja paziņojuma izveide* izsaukums, kuram tiek padoti komandas vadītāja lietotāja identifikators, uzdevuma identifikators un paziņojuma informācija par uzdevuma pārskatu nepieciešamību.
 - Izvēlam uzdevumam tiek piešķirts stāvoklis *Pārskats*.

Ja padotais uzdevuma stāvoklis ir *Aizvērts*, tad:

- Ja uzdevumam ir stāvoklis *Aktīvs*, tiek pārbaudīts vai autentificētais lietotājs ir izstrādes vadītājs. Ja uzdevumam ir stāvoklis *Aktīvs* un autentificēts lietotājs nav izstrādes vadītājs, tiek izvadīts kļūdas paziņojums nr. 6.
- Tiek pārbaudīts vai uzdevumam ir stāvoklis *Pārskats* un autentificētais lietotājs ir izstrādes vadītājs. Ja uzdevumam nav stāvoklis *Pārskats* vai autentificētais lietotājs nav izstrādes vadītājs, tad tiek izvadīts kļūdas paziņojums nr. 7.
- Ja uzdevumam ir stāvoklis, kas netika minēts iepriekš un autentificēts lietotājs nav izstrādes vadītājs, tad tiek izvadīts kļūdas paziņojums nr. 8.
- Ja netika izvadīts kļūdas paziņojums, tad:
 - Tiek pārbaudīts vai pastāv tādi uzdevumi, kuri atsaucas uz pašreizējo uzdevumu. Ja pastāv tādi uzdevumi, tad katram uzdevumam tiek pārbaudīts, vai pašreizējais uzdevums ir pēdējais atsaucis uzdevums ar atsaucis veidu *Ķēdes veidošanas atsaucis*. Ja tas ir pēdējais uzdevums, tad uzdevumam, kurš veidoja atsauci, tiek piešķirts stāvoklis *Izpildītāja piešķiršana* vai *Piešķirts* atkarībā no uzdevuma izpildītāja pastāvēšanas. Ja uzdevuma izpildītājs pastāv, tad tiek izveidots *2.2.5.1. Lietotāja paziņojuma izveide* funkcijas izsaukums, kuram tiek padoti atvērtā uzdevuma izpildītāja lietotāja identifikators, atvērtā uzdevuma identifikators un paziņojuma informācija par saistīto uzdevumu aizvēršanu.
 - Tiek izveidots *2.2.5.1. Lietotāja paziņojuma izveide* funkcijas izsaukums, kuram tiek padoti uzdevuma izpildītāja lietotāja identifikators, uzdevuma identifikators un paziņojuma informācija par uzdevuma aizvēršanu.
 - Izvēlētam uzdevumam tiek piešķirts stāvoklis *Aizvērts*.

Izvaddati

Ja apstrādes gaitā sistēmā tika reģistrēta kļūda, tad tiek izvadīts attiecīgais kļūdas paziņojums.

Kļūdu ziņojumi

1. Nav tiesību uzstādīt uzdevumu stāvokļus komandai: [Komandas nosaukums]

2. Uzdevumam nav iespējams pievienot stāvokli [Uzdevuma stāvokļa nosaukums], jo uzdevums tiek izpildīts vai ir aizvērts.
3. Nav iespējams uzsākt uzdevumu, ja uzdevumam netika piešķirts uzdevuma izpildītājs vai saistītie uzdevumi netika aizvērti.
4. Tikai uzdevuma izpildītājs var uzsākt uzdevuma izpildi.
5. Uz pārskatu ir iespējams nosūtīt tikai uzdevumu ar uzdevuma stāvokli *Aktīvs*.
6. Nav tiesības aizvērt uzdevumu bez izpildīta uzdevuma pārskata.
7. Nav tiesību pārskatīt un aizvērt izvēlēto uzdevumu.
8. Nav tiesības aizvērt uzdevumu.

2.2.4.5. Kalendāra ieraksta izveidošana/datu rediģēšana

Ievads

Funkcija nodrošina izstrādes dalībniekam (organizācijas īpašniekam, noteiktas komandas vadītājam vai komandas dalībniekam) izveidot vai rediģēt kalendāra ierakstu.

Ievaddati

Funkcija tiek izsaukta izmantojot tīmekļa vietnes saskarni (sk. 3.3.2.3. *Kalendāra ieraksta reģistrēšanas skats* sekciju). Funkcijai ir nepieciešams padot sekojošu informāciju:

- Uzdevuma identifikators. Vesels skaitlis. Obligāts.
- Lietotāja identifikators. Vesels skaitlis. Obligāts.
- Uzdevuma izstrādes posma sākuma laiks. Datums un laiks. Obligāts.
- Uzdevuma izstrādes posma ilgums. Datums un laiks. Obligāts.

Ja izstrādes dalībnieks rediģē eksistējošo kalendāra ierakstu, tad funkcijai tiek papildus padots sekojošie dati:

- Kalendāra ieraksta identifikators. Vesels skaitlis. Obligāts.

Apstrāde

Tiek pārbaudīts, vai formas ievadlauki atbilst sistēmas vispārējiem ierobežojumiem, kas ir nedefinēti sekcijā *Ievaddati* un aprakstīti sekcijā 2.2.1.2. *Kopīgie kļūdu ziņojumi un ievadlauku ierobežojumi*.

Tiek pārbaudīts vai autentificētais lietotājs ir uzdevuma izpildītājs. Ja autentificētais lietotājs nav uzdevuma izpildītājs, tad tiek izvadīts kļūdas paziņojums nr. 1.

Tiek pārbaudīts vai uzdevuma stāvoklis ir *Aktīvs* vai *Piešķirts*. Ja uzdevuma stāvoklis nav neviens no minētiem uzdevuma stāvokļiem, tad tiek izvadīts kļūdas paziņojums nr. 2.

Ja netika izvadīts kļūdas paziņojums, tad:

- Ja uzdevumam ir uzdevuma stāvoklis *Piešķirts*, tad tiek izveidots 2.2.4.4. *Uzdevuma stāvokļa iestatīšana* funkcijas izsaukums, kuram tiek padots izveidotā vai rediģētā uzdevuma identifikators un uzdevuma stāvokļa identifikators, kas atbilst uzdevuma stāvoklim *Aktīvs*.
- Ja autentificētais lietotājs izvelējas izveidot jaunu pazīmi, tad kalendāra ieraksts tiek izveidots komandas izstrādes procesa ietvaros un datu bāzē tiek izveidots ieraksts ar kalendāra ieraksta aktuāliem datiem.
- Ja uzdevumu izpildītājs izvelējās rediģēt eksistējoša kalendāra ieraksta informāciju, tad, izmantojot padoto kalendāra ieraksta identifikatoru, datu bāzē tiek atrasts rediģētā kalendāra ieraksta ieraksts. Ja netika izvadīts kļūdas paziņojums, tad kalendāra ieraksta informācija un datu bāzes ieraksts tiek atjaunoti.

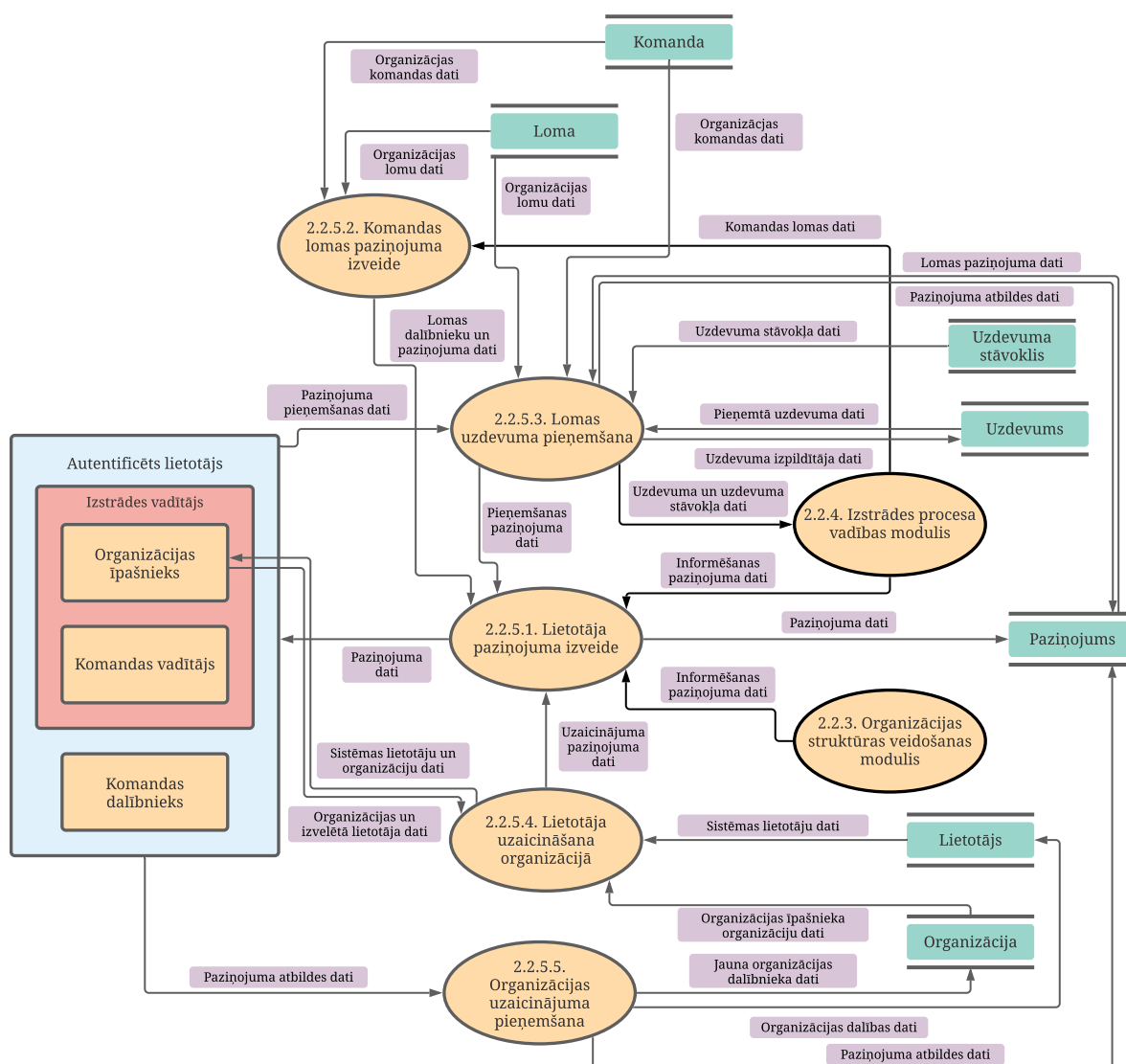
Izvaddati

Ja apstrādes gaitā sistēmā tika reģistrēta kļūda, tad tiek izvadīts attiecīgais kļūdas paziņojums.

Kļūdu ziņojumi

1. Tikai uzdevuma izpildītājam ir tiesības veidot kalendāra ierakstus izvēlamam uzdevumam.
2. Nav iespējams izveidot kalendāra ierakstu, jo uzdevumam ir saistīti uzdevumu vai uzdevums ir aizvērts, vai uzdevums ir pārskatā.

2.2.5. Sazināšanas modulis



2.5. att. 2. līmeņa datu plūsmas diagramma sazināšanas moduļim

Sazināšanas moduļa datu plūsmas diagrammā (sk. 2.5. att.) tiek attēlotas datu plūsmas starp sistēmas lietotājiem, moduļa funkcijām un entītijām, kas tiek izmantoti moduļa ietvaros. Modulis izmanto entītijas *Lietotājs*, *Organizācija*, *Loma*, *Komanda*, *Uzdevums*, *Uzdevuma stāvoklis* un *Paziņojums*. Diagrammā papildus tiek attēlotas datu plūsmas, kas ir saistīti ar organizācijas struktūras veidošanas moduļim (sk. 2.2.3. *Organizācijas struktūras veidošanas modulis*) un izstrādes procesa vadības moduļim (sk. 2.2.4. *Izstrādes procesa vadības modulis*).

2.2.5.1. Lietotāja paziņojuma izveide

Ievads

Funkcijas nodrošina iespēju plānošanas un laika pārvaldības sistēmai izveidot paziņojumus izvēltam lietotājam.

Ievaddati

Lai izsaukt funkciju, ir nepieciešams norādīt sekojošu informāciju:

- Saņēmēja lietotāja identifikators. Vesels skaitlis. Obligāts.
- Sūtītāja lietotāja identifikators. Vesels skaitlis. Neobligāts.
- Paziņojuma teksts. Simbolu virkne 1-1000 simboli. Obligāts.
- Organizācijas identifikators. Vesels skaitlis. Obligāts.
- Lomas identifikators. Vesels skaitlis. Neobligāts.
- Komandas identifikators. Vesels skaitlis. Neobligāts.
- Uzdevuma identifikators. Vesels skaitlis. Neobligāts.
- Paziņojuma kopas vispārēji unikāls identifikators. Simbolu virkne, 1-50 simboli. Neobligāts.
- Informatīvs paziņojums. Loģiskā vērtība (Jā/Nē). Obligāts.

Ja vērtība *Informatīvs paziņojums* pieņem vērtību *Nē*, tad nepieciešams norādīt sekojošu informāciju:

- Atbildes apstrādes funkcijas sistēmas galapunkts. Simbolu virkne, 1-1000 simboli. Obligāts. Sevī ietver sistēmas galapunktu, kurš izsauks saņēmēja atbildes apstrādes funkciju.

Apstrāde

Tiek pārbaudīts, vai funkcijas ievaddati atbilst sistēmas vispārējiem ierobežojumiem, kas ir nodefinēti sekcijā *Ievaddati* un aprakstīti sekcijā *2.2.1.2. Kopīgie kļūdu ziņojumi un ievadlauku ierobežojumi*.

Ja netika reģistrēts kļūdas paziņojums, tad tiek izveidots jaunais paziņojums sistēmas ietvaros un datu bāzē tiek izveidots ieraksts ar padotiem funkcijai paziņojuma datiem.

Izvaddati

Ja apstrādes gaitā sistēmā tika reģistrēta kļūda, tad tiek izvadīts attiecīgais kļūdas paziņojums.

2.2.5.2. Komandas lomas paziņojuma izveide

Ievads

Funkcija nodrošina iespēju plānošanas un laika pārvaldības sistēmai izveidot paziņojumu izvēlētai komandas lomai.

Ievaddati

Lai izsauktu funkciju ir nepieciešams norādīt sekojošu informāciju:

- Paziņojuma teksts. Simbolu virkne 1-1000 simboli. Obligāts.
- Organizācijas identifikators. Vesels skaitlis. Obligāts.
- Komandas identifikators. Vesels skaitlis. Obligāts.
- Lomas identifikators. Vesels skaitlis. Obligāts.
- Uzdevuma identifikators. Vesels skaitlis. Neobligāts.
- Informatīvs paziņojums. Loģiskā vērtība (Jā/Nē). Obligāts.

Ja vērtība *Informatīvs paziņojums* pieņem vērtību *Nē*, tad nepieciešams norādīt sekojošu informāciju:

- Atbildes apstrādes funkcijas sistēmas galapunkts. Simbolu virkne, 1-1000 simboli. Obligāts. Sevī ietver sistēmas galapunktu, kurš izsauks saņēmēja atbildes apstrādes funkciju.

Apstrāde

Tiek pārbaudīts, vai funkcijas ievaddati atbilst sistēmas vispārējiem ierobežojumiem, kas ir nodefinēti sekcijā *Ievaddati* un aprakstīti sekcijā 2.2.1.2. *Kopīgie kļūdu ziņojumi un ievadlauku ierobežojumi*.

Izmantojot organizācijas, komandas, lomas identifikatorus, sistēmas datu bāzē tiek atrasts izvēlētas organizācijas, komandas un lomas ieraksti. Tiek pārbaudīts vai izvēlētajā komandā pastāv izvēlēta loma. Ja izvēlēta loma nepastāv komandas ietvaros, tad tiek izsaukts kļūdas paziņojums nr. 1.

Tiek pārbaudīts, vai izvēlētajā komandā pastāv vismaz viens komandas dalībnieks ar izvēlēto lomu. Ja norādītā komandā nepastāv vismaz viens komandas dalībnieks ar izvēlēto lomu, tad funkcija beidz apstrādi.

Ja komandā pastāv komandas dalībnieks ar izvēlēto lomu, tad katram komandas dalībniekam tiek izveidots funkcijas 2.2.5.1. *Lietotāja paziņojuma izveide* izsaukums, kuram tiek padots komandas dalībnieka lietotāja identifikators, padotais funkcijā paziņojuma teksts, organizācijas, lomas, komandas, uzdevuma identifikatori un atbildes apstrādes 2.2.5.3. *Lomas uzdevuma pieņemšana* funkcijas galapunkts. Izsaukumam papildus tiek padots uzģenerēts lomas paziņojuma unikāls identifikators, kurš ir vienāds priekš katram paziņojumam, kas tiks izveidots katram komandas dalībniekam.

Izvaddati

Ja apstrādes gaitā sistēmā tika reģistrēta kļūda, tad tiek izvadīts attiecīgais kļūdas paziņojums.

Kļūdu ziņojumi

1. Izvēlēta loma nepastāv izvēlētas komandas lomu kopā.

2.2.5.3. Lomas uzdevuma pieņemšana

Ievads

Funkcija nodrošina iespēju komandas dalībniekam, kurš saņēma komandas lomas uzdevuma iesniegšanas paziņojumu, pieņemt lomas uzdevumu izpildei.

Ievaddati

Izmantojot sistēmas saskarni, autentificētais lietotājs izsauc funkciju, kurai ir nepieciešams padot sekojošu informāciju:

- Lomas identifikators. Vesels skaitlis. Obligāts.
- Komandas identifikators. Vesels skaitlis. Obligāts.
- Organizācijas identifikators. Vesels skaitlis. Obligāts.
- Uzdevuma identifikators. Vesels skaitlis. Obligāts.
- Paziņojuma identifikators. Vesels skaitlis. Obligāts.

- Paziņojuma vispārēji unikāls identifikators. Simbolu virkne 1-50 simboli. Obligāts.

Apstrāde

Tiek pārbaudīts, vai funkcijas ievaddati atbilst sistēmas vispārējiem ierobežojumiem, kas ir nodefinēti sekcijā *Ievaddati* un aprakstīti sekcijā 2.2.1.2. *Kopīgie kļūdu ziņojumi un ievadlauku ierobežojumi*.

Izmantojot organizācijas, komandas, lomas, uzdevuma, paziņojuma identifikatorus, sistēmas datu bāzē tiek atrasts izvēlētās organizācijas, komandas, lomas, uzdevuma un paziņojuma ieraksti.

Tiek pārbaudīts vai autentificētais lietotājs ir organizācijas dalībnieks, autentificēts lietotājs ir komandas dalībnieks organizācijas ietvaros un komandas ietvaros autentificētam lietotājam ir piešķirta padota komandas loma. Ja kaut viens minēts nosacījums neizpildās, tad tiek izsaukts kļūdas paziņojums nr. 1.

Tiek pārbaudīts vai izvēlētais uzdevums ir izveidots padotai komandai un izvēlētai komandas lomai. Ja uzdevums neizveidots izvēlētai komandai vai izvēlētai lomai, tad tiek izvadīts kļūdas paziņojums nr. 2.

Tiek pārbaudīts vai izvēlētais uzdevums tika pieņemts citas lomas dalībniekam. Ja uzdevums tika pieņemts citas lomas dalībniekam, tad tiek izvadīts kļūdas paziņojums nr. 3.

Ja netika izvadīts kļūdas paziņojums, tad padotam uzdevumam kā uzdevumu izpildītājs tiek iestatīts autentificētā lietotāja identifikators un tiek izveidots funkcijas 2.2.4.4. *Uzdevuma stāvokļa iestatīšana* izsaukums, kuram tiek padots uzdevuma identifikators un uzdevuma stāvokļa identifikators, kas atbilst uzdevuma stāvoklim *Piešķirts*.

No datu bāzes tiek atlasīti visi paziņojumi ar padoto paziņojuma vispārēju unikālu identifikatoru (paziņojumi, kas tika atsūtīti pārējiem lomas dalībniekiem). Katrs atlasītais paziņojums tiek atcelts (Loģiskā vērtība paziņojuma ieraksta atbildes laukā: Nē) un funkcijas padotam paziņojumam tiek reģistrēta atbilde, par uzdevuma pieņemšanu (Loģiskā vērtība paziņojuma ieraksta atbildes laukā: Jā).

Izvaddati

Ja apstrādes gaitā sistēmā tika reģistrēta kļūda, tad tiek izvadīts attiecīgais kļūdas paziņojums.

Kļūdu ziņojumi

1. Lietotājam nav tiesību pieņemt izvēlētas komandas lomas uzdevumu.
2. Izvēlētais uzdevums netika izveidots izvēlētas komandas lomas ietvaros.
3. Uzdevumu pieņēma cits komandas lomas dalībnieks.

2.2.5.4. Lietotāja uzaicināšana organizācijā

Ievads

Funkcija nodrošina iespēju organizācijas īpašniekam izveidot organizācijas uzaicinājumu sistēmas lietotājam.

Ievaddati

Organizācijas īpašniekam tiek piedāvāta forma, kur ir nepieciešams norādīt sekojošu informāciju:

- Organizācijas identifikators. Vesels skaitlis. Obligāts. Formā tiek piedāvāta izvēlne ar organizācijām, kuriem autentificētais lietotājs ir organizācijas īpašnieks.
- Lietotāju identifikators. Vesels skaitlis. Obligāts. Formā tiek piedāvāts ievadlauks ar automātisko aizpildīšanas pabeigšanas funkciju. Laukam tiek padoti sistēmas lietotāji, kas nav izvēlētas organizācijas dalībnieki.

Apstrāde

Tiek pārbaudīts, vai formas ievadlauki atbilst sistēmas vispārējiem ierobežojumiem, kas ir nodēfīnēti sekcijā *Ievaddati* un aprakstīti sekcijā *2.2.1.2. Kopīgie kļūdu ziņojumi un ievadlauku ierobežojumi*.

Izmantojot organizācijas un lietotāja identifikatorus, sistēmas datu bāzē tiek atrasts izvēlētas organizācijas un lietotāja ieraksti. Tiek pārbaudīts vai autentificētais lietotājs ir izvēlētas organizācijas īpašnieks. Ja izvēlētais lietotājs nav izvēlētas organizācijas īpašnieks, tad tiek izvadīts kļūdas paziņojums nr. 1.

Tiek pārbaudīts vai izvēlētais lietotājs nav izvēlētas organizācijas dalībnieks. Ja izvēlētais lietotājs ir organizācijas dalībnieks, tad tiek izvadīts kļūdas paziņojums nr. 2.

Ja netika izvadīts kļūdas paziņojums, tad tiek izveidots funkcijas *2.2.5.1. Lietotāja paziņojuma izveide* izsaukums, kuram tiek padots izvēlētas organizācijas identifikators, izvēlēta

lietotāja identifikators un atbildes apstrādes 2.5.5.5. *Organizācijas uzaicinājuma pieņemšana* funkcijas galapunkts.

Izvaddati

Ja apstrādes gaitā sistēmā tika reģistrēta kļūda, tad tiek izvadīts attiecīgais kļūdas paziņojums.

Ja apstrādes gaitā sistēmā netika reģistrēta kļūda, tad tiek izvadīts paziņojums par veiksmīgu organizācijas uzaicinājuma izveidošanu.

Kļūdu ziņojumi

1. Nav tiesību izveidot organizācijas uzaicinājumus izvēlētai organizācijai.
2. Izvēlētais lietotājs ir izvēlētas organizācijas dalībnieks.

2.5.5.5. Organizācijas uzaicinājuma pieņemšana

Ievads

Funkcija nodrošina iespēju autentificētam lietotājam, kuram tika atsūtīt organizācijas uzaicinājums, pieņemt uzaicinājumu un kļūst par organizācijas dalībnieku.

Ievaddati

Izmantojot sistēmas saskarni, autentificētais lietotājs izsauc funkciju, kurai ir nepieciešams padot sekojošu informāciju:

- Organizācijas identifikators. Vesels skaitlis. Obligāts.
- Paziņojuma identifikators. Vesels skaitlis. Obligāts.
- Organizācijas uzaicinājums tika pieņemts. Loģiskā vērtība (Jā/Nē). Obligāts.

Apstrāde

Tiek pārbaudīts, vai formas ievadlauki atbilst sistēmas vispārējiem ierobežojumiem, kas ir noteikti sekcijā *Ievaddati* un aprakstīti sekcijā 2.2.1.2. *Kopīgie kļūdu ziņojumi un ievadlauku ierobežojumi*.

Izmantojot organizācijas un paziņojuma identifikatorus, sistēmas datu bāzē tiek atrasts izvēlētas organizācijas un paziņojuma ieraksti. Tiek pārbaudīts vai paziņojuma saņēmējs ir autentificēts lietotājs. Ja paziņojuma saņēmējs nav autentificētais lietotājs, tad tiek izvadīts kļūdas paziņojums nr. 1.

Ja netika izvadīts kļūdas paziņojums, tad:

- Tiek pārbaudīts, vai lietotājs pieņēma organizācijas uzaicinājumu. Ja lietotājs pieņēma organizācijas uzaicinājumu, tad tiek izveidots organizācijas dalībnieku ieraksts datu bāzē, kur tiek norādīts organizācijas identifikators un autentificētā lietotāja identifikators.
- Paziņojuma atbilde tiek reģistrēta paziņojuma ierakstā un aktuāla informācija tiek saglabāta datu bāzē.

Izvaddati

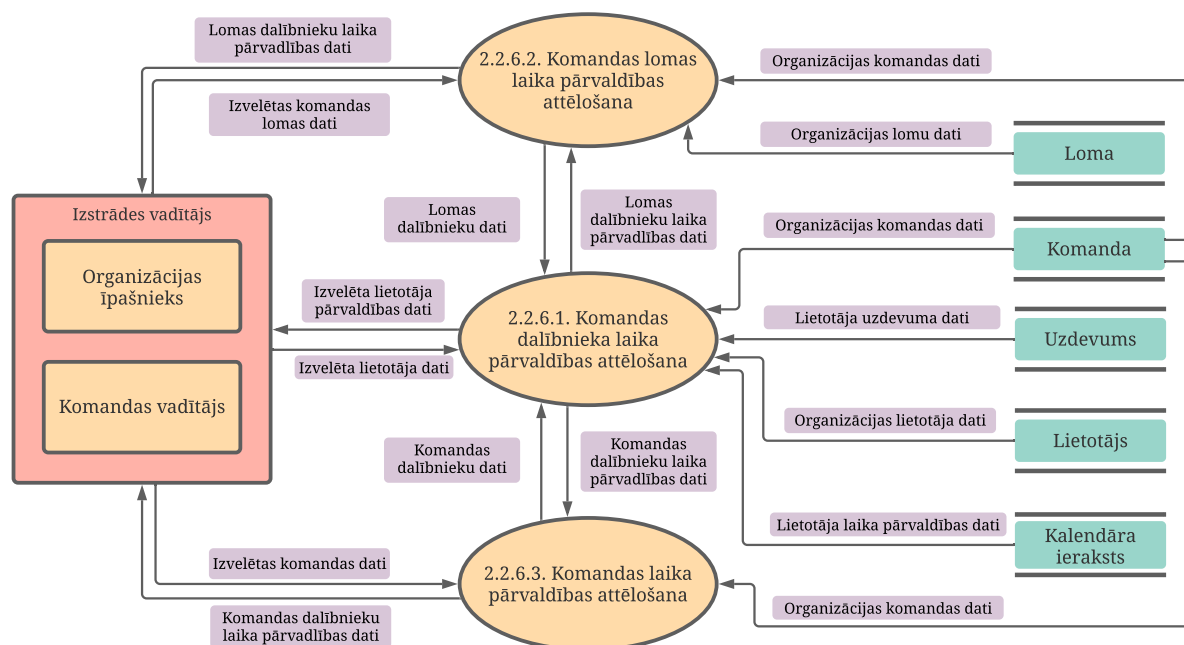
Ja apstrādes gaitā sistēmā tika reģistrēta kļūda, tad tiek izvadīts attiecīgais kļūdas paziņojums.

Ja apstrādes gaitā sistēmā netika reģistrēta kļūda, tad tiek izvadīts paziņojums par veiksmīgu organizācijas uzaicinājuma pieņemšanu.

Kļūdu ziņojumi

1. Nav tiesību atbildēt uz organizācijas uzaicinājumu.

2.2.6. Izstrādes procesa attēlošanas modulis



2.6. att. 2. līmeņa datu plūsmas diagramma izstrādes procesa attēlošanas modulim

Izstrādes procesa attēlošanas moduļa datu plūsmas diagrammā (sk. 2.6. att.) tiek attēlotas datu plūsmas starp organizācijas izstrādes vadītājiem, moduļa funkcijām un entītijām, kas tiek

izmantoti moduļa ietvaros. Modulis izmanto entītijas *Lietotājs, Loma, Komanda, Uzdevums, Kalendāra ieraksts*. Izstrādes procesa attēlošanas modulis ir modulis, kas apstrādā datu bāzē eksistējošos datus (jaunie ieraksti netiek veidoti vai saglabāti datu bāzē).

2.2.6.1. Komandas dalībnieka laika pārvaldības apskate

Ievads

Funkcija nodrošina iespēju izstrādes vadītājam (organizācijas īpašniekam vai noteiktas komandas vadītājam) apskatīt izvēlētas komandas dalībnieka laika pārvaldības datus.

Ievaddati

Izstrādes vadītājam tiek piedāvāta forma, kur ir nepieciešams norādīt sekojošu informāciju (kad funkcija tiek izsaukta moduļa ietvaros, tad forma netiek piedāvāta. Funkcijai tomēr ir nepieciešams padot minēto informāciju):

- Komandas identifikators. Vesels skaitlis. Obligāts. Formā tiek piedāvāta izvēlne ar organizācijas komandām, kurām izstrādes vadītājam ir tiesības vadīt izstrādes procesu. Organizācijas īpašniekam – visas organizācijas komandas, komandas vadītājam – visas komandas, kur organizācijas dalībnieks ir komandas vadītājs.
- Komandas dalībnieka identifikators. Vesels skaitlis. Obligāts. Kad tika izvēlēta komanda, formā tiek piedāvāta izvēlne ar izvēlētas komandas dalībniekiem.

Kopā ar formas norādītajiem ievaddatiem funkcijai tiek padots paslēpti lauki:

- Organizācijas identifikators. Vesels skaitlis. Obligāts.

Apstrāde

Tiek pārbaudīts, vai formas ievadlauki atbilst sistēmas vispārējiem ierobežojumiem, kas ir nodefinēti sekcijā *Ievaddati* un aprakstīti sekcijā *2.2.1.2. Kopīgie kļūdu ziņojumi un ievadlauku ierobežojumi*.

Izmantojot organizācijas, komandas un komandas dalībnieka identifikatorus, sistēmas datu bāzē tiek atrasts izvēlētas organizācijas, komandas un komandas dalībnieka ieraksti. Tiek pārbaudīts, vai izvēlēta komanda pastāv organizācijas ietvaros. Ja izvēlēta komanda nepastāv organizācijas ietvaros, tad tiek izvadīts kļūdas paziņojums nr. 1.

Tiek pārbaudīts, vai autentificētais lietotājs ir organizācijas īpašnieks vai izvēlētas komandas vadītājs. Ja autentificētais lietotājs nav organizācijas īpašnieks un nav izvēlētas komandas vadītājs, tad tiek izvadīts kļūdas paziņojums nr. 2.

Tiek pārbaudīts, vai izvēlētais lietotājs norādītā komandā ir komandas dalībnieks. Ja izvēlētais lietotājs nav komandas dalībnieks, tad tiek izvadīts kļūdas paziņojums nr. 3.

Ja netika izvadīts kļūdas paziņojums, tad no datu bāzes tiek atlasīti visi kalendāra ieraksti, kas ir saistīti ar uzdevumiem, kuri tika izpildīti izvēlētas komandas ietvaros, un ir izveidoti izvēlētam komandas dalībniekam.

Izvaddati

Ja apstrādes gaitā sistēmā tika reģistrēta kļūda, tad tiek izvadīts attiecīgais kļūdas paziņojums.

Ja apstrādes gaitā sistēmā netika reģistrēta kļūda, tad funkcija atgriež visus atlasītus kalendāra ierakstus, kas ir saistīti ar uzdevumiem, kuri tika izpildīti izvēlētas komandas ietvaros, un ar padoto komandas dalībnieku.

Kļūdu ziņojumi

1. Izvēlēta komanda nepastāv organizācijas ietvaros.
2. Nav tiesību apskatīt komandas dalībnieku laika pārvaldības datus komandai:
[Komandas nosaukums]
3. Lietotājs nav izvēlētas komandas dalībnieks.

2.2.6.2. Komandas lomas laika pārvaldības apskate

Ievads

Funkcija nodrošina iespēju izstrādes vadītājam (organizācijas īpašniekam vai noteiktas komandas vadītājam) apskatīt izvēlēta komandas lomas laika pārvaldības datus.

Ievaddati

Izstrādes vadītājam tiek piedāvāta forma, kur ir nepieciešams norādīt sekojošu informāciju:

- Komandas identifikators. Vesels skaitlis. Obligāts. Formā tiek piedāvāta izvēlne ar organizācijas komandām, kurām izstrādes vadītājam ir tiesības vadīt izstrādes

procesu. Organizācijas īpašniekam – visas organizācijas komandas, komandas vadītājam – visas komandas, kur organizācijas dalībnieks ir komandas vadītājs.

- Komandas lomas identifikators. Vesels skaitlis. Obligāts. Kad tika izvēlēta komanda, formā tiek piedāvāta izvēlne ar izvēlētas komandas lomām.

Kopā ar formas norādītajiem ievaddatiem funkcijai tiek padots paslēpti lauki:

- Organizācijas identifikators. Vesels skaitlis. Obligāts.

Apstrāde

Tiek pārbaudīts, vai formas ievadlauki atbilst sistēmas vispārējiem ierobežojumiem, kas ir nodefinēti sekcijā *Ievaddati* un aprakstīti sekcijā 2.2.1.2. *Kopīgie kļūdu ziņojumi un ievadlauku ierobežojumi*.

Izmantojot organizācijas, komandas un komandas lomas identifikatorus, sistēmas datu bāzē tiek atrasts izvēlētas organizācijas, komandas un komandas lomas ieraksti. Tiek pārbaudīts, vai izvēlēta komanda pastāv organizācijas ietvaros. Ja izvēlēta komanda nepastāv organizācijas ietvaros, tad tiek izvadīts kļūdas paziņojums nr. 1.

Tiek pārbaudīts, vai autentificētais lietotājs ir organizācijas īpašnieks vai izvēlētas komandas vadītājs. Ja autentificētais lietotājs nav organizācijas īpašnieks un nav izvēlētas komandas vadītājs, tad tiek izvadīts kļūdas paziņojums nr. 2.

Tiek pārbaudīts, vai izvēlēta loma pastāv norādītā komandas lomu kopā. Ja izvēlēta loma nepastāv norādītās komandas lomu kopā, tad tiek izvadīts kļūdas paziņojums nr. 3.

Ja netika izvadīts kļūdas paziņojums, tad katram komandas dalībniekam, kuram tika piešķirta norādīta loma, tiek izveidots funkcijas 2.2.6.1. *Komandas dalībnieka laika pārvaldības apskate* izsaukums, kuram tiek padots komandas un organizācijas identifikatori kopā ar komandas dalībnieka, kuram tika piešķirta norādīta loma, identifikatoru. Visi kalendāra ieraksti tiek apkopoti un atgriezti kā funkcijas rezultāts.

Izvaddati

Ja apstrādes gaitā sistēmā tika reģistrēta kļūda, tad tiek izvadīts attiecīgais kļūdas paziņojums.

Ja apstrādes gaitā sistēmā netika reģistrēta kļūda, tad funkcija atgriež visus atlasītus kalendāra ierakstus, kas ir saistīti ar uzdevumiem, kuri tika izpildīti izvēlētas komandas ietvaros, un tika izveidoti izvēlētas lomas dalībniekiem.

Kļūdu ziņojumi

1. Izvēlēta komanda nepastāv organizācijas ietvaros.
2. Nav tiesību apskatīt komandas lomas laika pārvaldības datus komandai:
[Komandas nosaukums]
3. Izvēlēta loma nepastāv norādītas komandas lomu kopā.

2.2.6.3. Komandas laika pārvaldības apskate

Ievads

Funkcija nodrošina iespēju izstrādes vadītājam (organizācijas īpašniekam vai noteiktas komandas vadītājam) apskatīt izvēlēta komandas laika pārvaldības datus.

Ievaddati

Izstrādes vadītājam tiek piedāvāta forma, kur ir nepieciešams norādīt sekojošu informāciju:

- Komandas identifikators. Vesels skaitlis. Obligāts. Formā tiek piedāvāta izvēlne ar organizācijas komandām, kurām izstrādes vadītājam ir tiesības vadīt izstrādes procesu. Organizācijas īpašniekam – visas organizācijas komandas, komandas vadītājam – visas komandas, kur organizācijas dalībnieks ir komandas vadītājs.

Kopā ar formas norādītajiem ievaddatiem funkcijai tiek padots paslēpti lauki:

- Organizācijas identifikators. Vesels skaitlis. Obligāts.

Apstrāde

Tiek pārbaudīts, vai formas ievadlauki atbilst sistēmas vispārējiem ierobežojumiem, kas ir nedefinēti sekcijā *Ievaddati* un aprakstīti sekcijā *2.2.1.2. Kopīgie kļūdu ziņojumi un ievadlauku ierobežojumi*.

Izmantojot organizācijas, komandas un komandas lomas identifikatorus, sistēmas datu bāzē tiek atrasts izvēlētas organizācijas, komandas un komandas lomas ieraksti. Tiek pārbaudīts, vai izvēlēta komanda pastāv organizācijas ietvaros. Ja izvēlēta komanda nepastāv organizācijas ietvaros, tad tiek izvadīts kļūdas paziņojums nr. 1.

Tiek pārbaudīts, vai autentificētais lietotājs ir organizācijas īpašnieks vai izvēlētas komandas vadītājs. Ja autentificētais lietotājs nav organizācijas īpašnieks un nav izvēlētas komandas vadītājs, tad tiek izvadīts kļūdas paziņojums nr. 2.

Ja netika izvadīts kļūdas paziņojums, tad katram komandas dalībniekam tiek izveidots funkcijas 2.2.6.1. *Komandas dalībnieka laika pārvaldības apskate* izsaukums, kuram tiek padots komandas un organizācijas identifikatori kopā ar komandas dalībnieka identifikatoru. Visi kalendāra ieraksti tiek apkopoti un atgriezti kā funkcijas rezultāts.

Izvaddati

Ja apstrādes gaitā sistēmā tika reģistrēta kļūda, tad tiek izvadīts attiecīgais kļūdas paziņojums.

Ja apstrādes gaitā sistēmā netika reģistrēta kļūda, tad funkcija atgriež visus atlasītus kalendāra ierakstus, kas ir saistīti ar uzdevumiem, kuri tika izpildīti izvēlētās komandas ietvaros.

Kļūdu ziņojumi

1. Izvēlēta komanda nepastāv organizācijas ietvaros.
2. Nav tiesību apskatīt laika pārvaldības datus komandai: [Komandas nosaukums]

2.3. Nefunkcionālās prasības

2.3.1. Veiktspējas prasības

Plānošanas un laika pārvaldības sistēmai ir jānodrošina sekojošas veiktspējas prasības:

- Sistēmai ir nepieciešams apstrādāt lietotāju pieprasījumus vismaz 3 sekunžu laikā. Tas nozīmē kā pieprasījumi lietojumprogrammas saskarnei, pieprasījums datu bāzei (datu saglabāšanai un datu informācijas iegūšanai) un informācijas atveidošana uz lietotāja ierīces ekrāna ir nepieciešams, lai būtu paveikti dotā laika intervālā.
- Pieprasījumi lietojumprogrammas saskarnei un pieprasījumi datu bāzei datu saņemšanai pēc iespējas ir jābūt asinhroniem.
- Sistēmai ir nepieciešams nodrošināt iespēju paralēli apstrādāt vismaz 1000 lietotāju pieprasījumus paralēli.

2.3.2. Lietojamības prasības

Plānošanas un laika pārvaldības sistēmai ir jānodrošina sekojošas lietojamības prasības:

- Tomēr sistēmā netiek atbalstīts reaģējošs tīmekļa vietnes dizains, tomēr sistēmas saskarnēm ir nepieciešams pielāgoties dažādu klēpj datoru un monitoru izmēriem, kad lapa tiek izmantota pilnekrāna stāvoklī.
- Ja ir paredzēts kā lietotāju vietnes pieprasījuma apstrāde aizņems vairāk par izvēlēto laika intervālu (sk. 2.3.1. *Veiktspējas prasības* sekciju), tad ir nepieciešams vietnē attēlot ielādētāju, kurš piedāvās lietotājam izpratni, ka vietne tiek atveidota.
- Situācijā, kad lietotājam ir nepieciešams ievadīt datus, kas pastāv sistēmas ietvaros, tad lietotājam tiek piedāvāts ievaddatu lauks ar automātisko teksta pabeigšanas funkcionalitāti (Autocomplete). Piedāvātos ievaddatu laukos ir iespējams atvērt nolaižamu sarakstlodziņu, lai izvēlētos datus, kas pastāv sistēmā.
- Sistēmai ir nepieciešams nodrošināt darbību latviešu valodā. Saskarnes nosaukumi, pogas un kļūdu vai informācijas paziņojumi tiek attēloti latviešu valodā.

2.3.3. Drošības prasības

Plānošanas un laika pārvaldības sistēmai ir jānodrošina sekojošas drošības prasības:

- Sistēmai ir jāatbalsta uz sīkdatnēm balstītu autentifikāciju un autorizāciju mehānismus. Autentifikācijas un autorizācijai tiek izmantots lietotāja pieteikuma vārds un lietotāja parole.
- Reģistrējot sistēmā jaunu lietotāju kontu, lietotāja parole tiek šifrēta, izmantojot PBKDF2 algoritmu [8] un šifrētā veidā tiek glabātā datu bāzē. Izmantojot piedāvāto algoritmu, sistēmai ir nepieciešams izveidot aizsardzību pret pārlases uzbrukumiem.
- Sistēmai ir nepieciešams izmantot HTTPS protokolu lietotāju pieprasījuma datu pārsūtīšanas lietojumprogrammas saskarnei.
- Sistēmai ir nepieciešams novērst starpvietņu skriptēšanu (XSS), starpvietņu pieprasījumu viltošanu (CSRF) un SQL iesprašanas uzbrukumus

2.3.4. Pieejamības prasības

Plānošanas un laika pārvaldības sistēmai ir jānodrošina sekojošas pieejamības prasības:

- Sistēmai ir jāatbalsta tīmekļa pārlūkus, kuri implementē un atbalsta aktuālos tīmekļa standartus (HTML5 [5] , CSS3 [6], ECMA-262 [7]). Sistēma atbalsta sekojošu tīmekļa pārlūkus: Microsoft Edge, Google Chrom, Firefox, Opera.
- Sistēmai ir jābūt pieejamai, izņemot laiku, kad tiek uzstātīti sistēmas atjaunojumi.

2.3.5. Uzturēšanas prasības

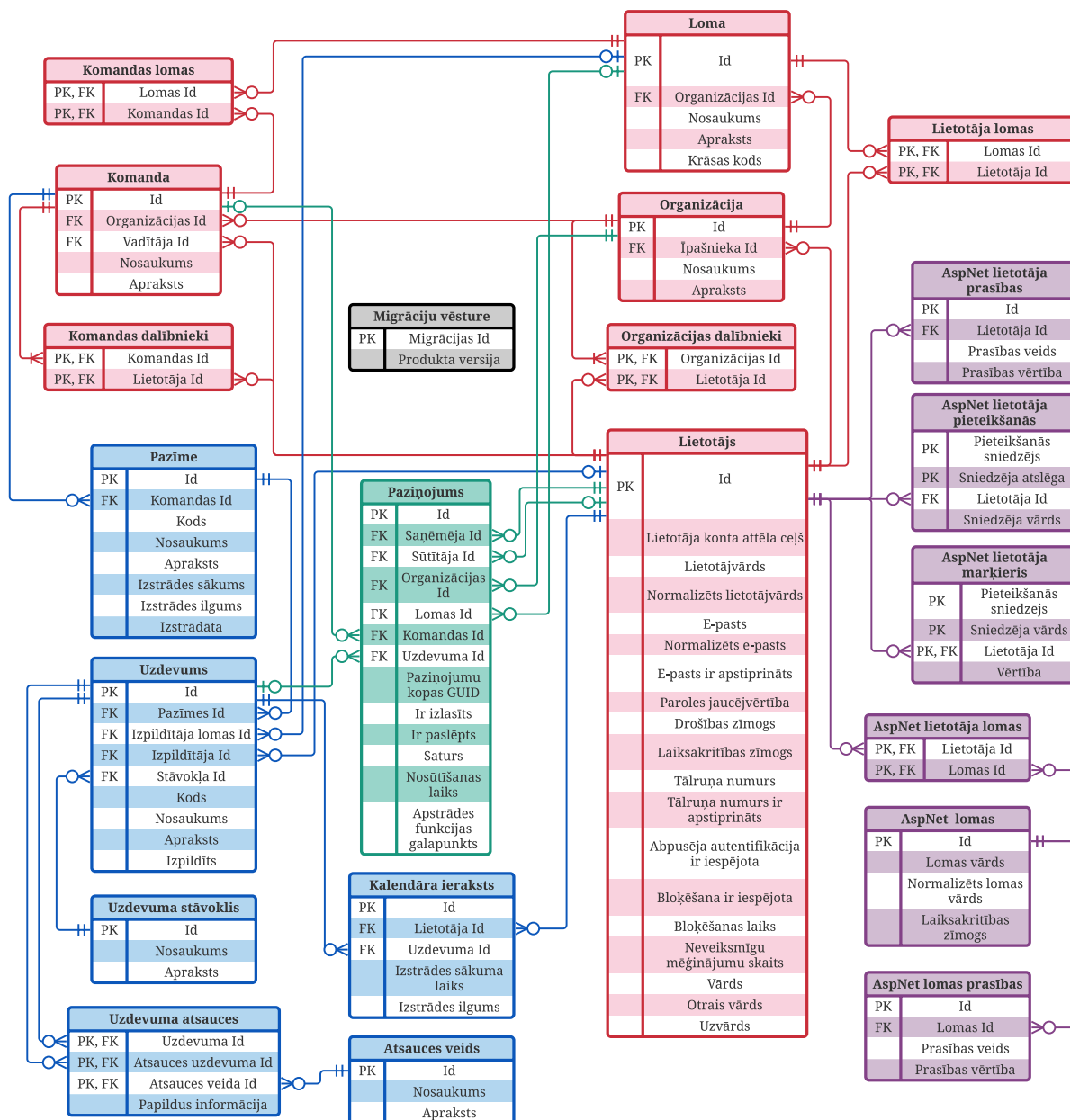
Plānošanas un laika pārvaldības sistēmai ir jānodrošina sekojošas uzturēšanas prasības:

- Ja sistēmā tiks plānots atjauninājums, tad ir nepieciešams izveidot attiecīgu paziņojumu par laiku, kad sistēma nebūs pieejama.
- Sistēmā organizācijas dati tiek glabāti 2 gadus. Ja organizācijas īpašnieks vai organizācijas dalībnieks nepieprasīja organizācijas datus divu gadu intervālā, tad organizācijas dati tiks dzēsti no sistēmas.

3. PROGRAMMATŪRAS PROJEKTĒJUMA APRAKSTS

3.1. Datu bāzes projektējums

3.1.1. Loģiskais datu bāzes apraksts



3.1. att. Loģiskais datu bāzes modelis izstrādātai sistēmai.

Loģiskais datu bāzes modelis (sk. 3.1. att.) atbilst konceptuālajam modelim (sk. 2.1. att.), pārveidotam atbilstoši vārnkāju sintaksei. Līdzīgi kā konceptuālais datu bāzes modelis, loģiskais datu bāzes modelis ar krāsas palīdzību ir sadalīts piecās daļās, kas reprezentē noteikto sistēmas moduli vai tehnoloģiju, kas tiks izmantota sistēmas ietvaros. Entītijas ir piešķirtas noteiktam pamata modulim (vai tagad arī noteiktai funkcionalitātei, kurai ir nepieciešams

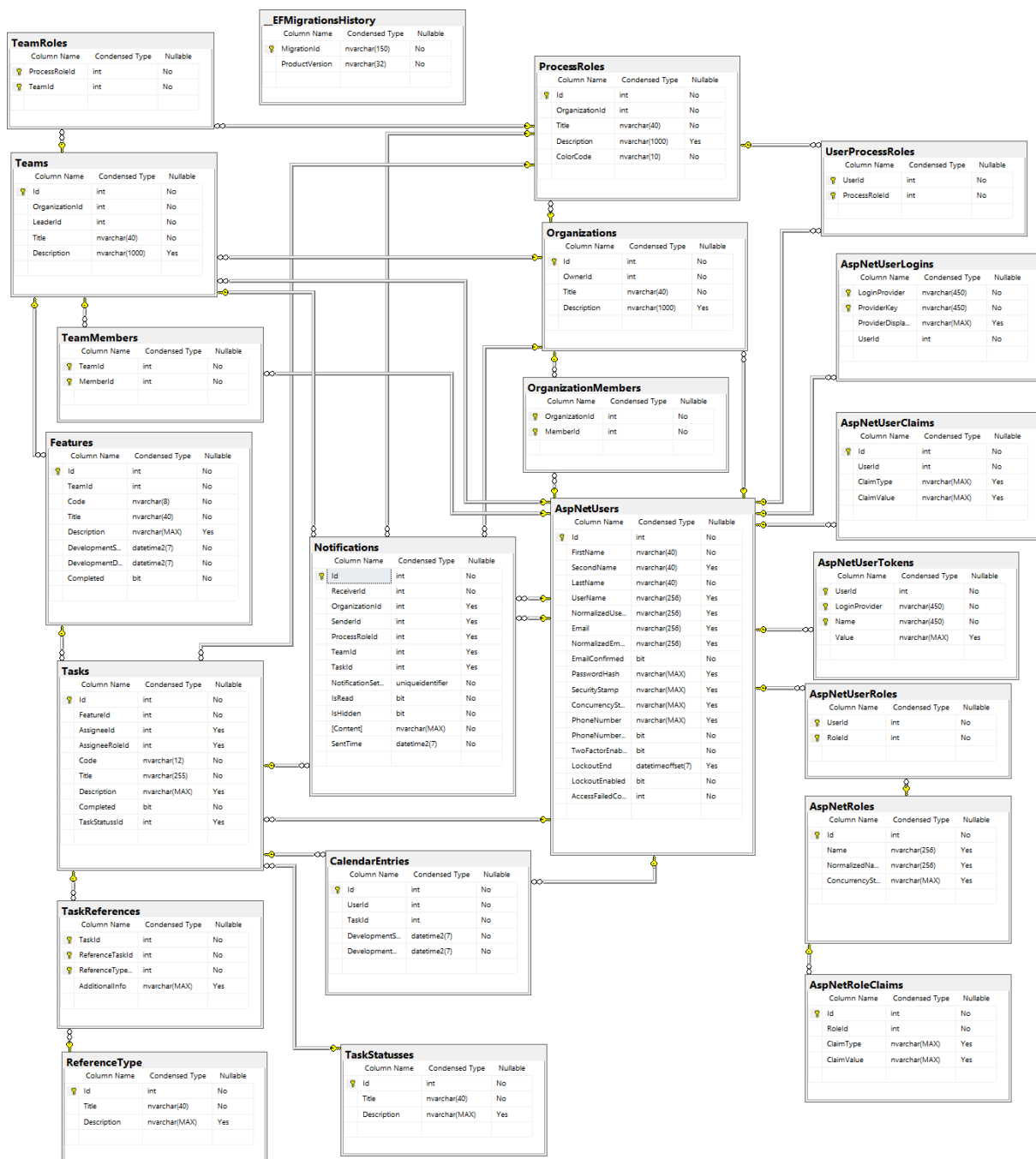
izveidot papildus entītijas), kurā tiek veikta entītijū datu rediģēšana un modificēšana, tomēr parēju sistēmas moduļu ietvaros ir iespējams izmantot entītijas datus. Loģiskā datu bāzē attēloti sekojošas sistēmas daļas:

- Organizācijas struktūras veidošanas modulis (sarkanā krāsa)
- Izstrādes procesa vadības modulis (zilā krāsa)
- Sazināšanas modulis (zaļā krāsa)
- ASP.NET Core Identity autentifikācijas un autorizācijas mehānismi (violetā krāsa)
- EntityFramework CodeFirst migrāciju vēsture (pelēka krāsa)

Loģiskajā datu bāzes modelī katrai entītijai tiek norādīti vairāki atribūti kā arī tiek norādīti primārās un ārējās atslēgas. Tiek ieviestas vairākas entītijas, lai realizētu daudz pret daudz relācijas, kas tika ieviestas konceptuālajā datu bāzes modelī. Tika ieviestas entītijas *Organizācijas dalībnieki*, *Lietotāja lomas*, *Komandas dalībnieki*, *Komandas lomas* un *Uzdevuma atsauces*, kas visas satur ārējas atslēgas uz entītijām, kas veidoja daudz pret daudz relāciju.

Tika ieviestas entītijas *AspNet lietotāja prasības*, *AspNet lietotāja pieteikšanās*, *AspNet lietotāja marķieris*, *AspNet lietotāja lomas*, *AspNet lomas*, *AspNet lomas prasības*, kas tiks izmantotas, lai veiktu lietotāja autentifikāciju un autorizāciju. Projekta ietvaros tiks izmantota ASP.NET Core Identity funkcionalitāte un minētas tabulas ir izveidotas, lai izvairītos no nevēlamam kļūdām jau pārbaudītā un testētā programmatūras bibliotēkā, tomēr entītijas *AspNet lomas* un *AspNet lomas prasības* netiks izmantotas projekta ietvaros.

3.1.2. Fiziskais datu bāzes apraksts



3.2. att. Fiziskais datu bāzes modelis izstrādātai sistēmai.

Fiziskais datu bāzes modelis (sk. 3.2. att.) atbilst loģiskajam modelim (sk. 3.1. att.) un ir realizēts atbilstoši MS SQL Server datu bāzes pārvaldes sistēmai. Tabulas nav sadalītas pa krāsām, bet papildus katram tabulas laukam ir norādīts tā datu tips un īpašība, kas norāda vai lauka vērtība var pieņemt vērtību *null*.

Tabulu aprakstos tika minēti datu tipi, kas tiek iestatīti attiecīgam laukam atbilstoši MS SQL Server datu bāzes pārvaldes sistēmai. Simbols ‘-’ atslēgas kolonā nozīmē kā noteiktam laukam nav primārās un nav ārējas atslēgas.

Fiziskajā datu bāzes modelī tika attēlotas sekojošas tabulas:

Lietotāju tabula (sk. 3.1. tabulu) ir paredzēta lietotāju pamatinformācijas uzturēšanai. Tā kā lietotāja tabulā tika izveidoti lauki, kas tiks izmantoti autentifikācijas un autorizācijas mehānismos, lauku shēma netika modificēta, lai atbilst aprakstītām prasībām, kas tika minēti funkciju aprakstos. Obligātie lauki un lauku maksimālais simbolu skaits tiks pārbaudīts ievadlauku validācijas posmā, pirms datu bāzes ieraksta izveidošanas vai modificēšanas. Tas ir izdarīts ar testēšanas samazināšanas nolūku, jo tiek pieņemts, ka ASP.NET Core Identity funkcionalitāte nesatur kļūdas un netiks testēta projekta izstrādes ietvaros.

3.1. tabula

Datu bāzes sistēmas lietotāja datu tabula (AspNetUsers)

<i>Lauks</i>	<i>Datu tips</i>	<i>Atslēga</i>	<i>Cita informācija</i>	<i>Apraksts</i>
<i>Id</i>	<i>Int</i>	<i>PK</i>	<i>Auto increment Not null</i>	<i>Unikāls identifikators</i>
<i>FirstName</i>	<i>Nvarchar(40)</i>	-	<i>Not null</i>	<i>Lietotāja vārds</i>
<i>SecondName</i>	<i>Nvarchar(40)</i>	-	<i>Nullable</i>	<i>Lietotāja otrais vārds</i>
<i>LastName</i>	<i>Nvarchar(40)</i>	-	<i>Not null</i>	<i>Lietotāja uzvārds</i>
<i>UserName</i>	<i>Nvarchar(256)</i>	-	<i>Nullable</i>	<i>Lietotāja pieteikuma vārds</i>
<i>NormalizedUserName</i>	<i>Nvarchar(256)</i>	-	<i>Nullable</i>	<i>Normalizēts lietotāja pieteikuma vārds</i>
<i>Email</i>	<i>Nvarchar(256)</i>	-	<i>Nullable</i>	<i>Lietotāja e-pasts</i>
<i>NormalizedEmail</i>	<i>Nvarchar(256)</i>	-	<i>Nullable</i>	<i>Normalizēts lietotāja e-pasts</i>

<i>Lauks</i>	<i>Datu tips</i>	<i>Atslēga</i>	<i>Cita informācija</i>	<i>Apraksts</i>
<i>EmailConfirmed</i>	<i>Bit</i>	-	<i>Not null</i>	<i>Ir lietotāja e-pasts apstiprināts</i>
<i>PasswordHash</i>	<i>Nvarchar(max)</i>	-	<i>Nullable</i>	<i>Lietotāja paroles šifrēšanas vērtība</i>
<i>SecurityStamp</i>	<i>Nvarchar(max)</i>	-	<i>Nullable</i>	<i>Drošības zīmogs</i>
<i>ConcurrencyStamp</i>	<i>Nvarchar(max)</i>	-	<i>Nullable</i>	<i>Laiksakritības zīmogs</i>
<i>PhoneNumber</i>	<i>Nvarchar(max)</i>	-	<i>Nullable</i>	<i>Lietotāja telefona numurs</i>
<i>PhoneNumberConfirmed</i>	<i>Bit</i>	-	<i>Not null</i>	<i>Lietotāja telefona numurs ir apstiprināts</i>
<i>TwoFactorEnabled</i>	<i>Bit</i>	-	<i>Not null</i>	<i>Abpusēja autentifikācija ir iespējota</i>
<i>LockoutEnd</i>	<i>Datetimeoffset(7)</i>	-	<i>Nullable</i>	<i>Autentifikācijas bloķēšanas laiks</i>
<i>LockoutEnabled</i>	<i>Bit</i>	-	<i>Not null</i>	<i>Autentifikācijas bloķēšana ir iespējota</i>
<i>AccessFailedCount</i>	<i>Int</i>	-	<i>Not null</i>	<i>Autentifikācijas neveiksmīgu mēģinājumu skaits</i>

Lietotāja pieteikšanās datu tabula (sk. 3.2. tabulu) ir paredzēta lietotāja pieteikšanās sniedzēju pamatinformācijas uzturēšanai. Tabula ir izveidota autentifikācijas un autorizācijas mehānismiem, kas tiek izmantoti ASP.NET Core Identity.

3.2. tabula

Datu bāzes ASP.NET Core Identity lietotāja pieteikšanās datu tabula (AspNetUserLogins)

<i>Lauks</i>	<i>Datu tips</i>	<i>Atslēga</i>	<i>Cita informācija</i>	<i>Apraksts</i>
<i>LoginProvider</i>	<i>Nvarchar(450)</i>	<i>PK</i>	<i>Not null</i>	<i>Pieteikšanās sniedzējs</i>
<i>ProviderKey</i>	<i>Nvarchar(450)</i>	<i>PK</i>	<i>Not null</i>	<i>Pieteikšanās sniedzēja atslēga</i>
<i>ProviderDisplayName</i>	<i>Nvarchar(max)</i>	-	<i>Nullable</i>	<i>Pieteikšanās sniedzēja vārds</i>
<i>UserId</i>	<i>Int</i>	<i>FK</i>	<i>Not null</i>	<i>Lietotāja unikāls identifikators</i>

Lietotāja autentifikācijas marķieru datu tabula (sk. 3.3. tabulu) ir paredzēta lietotāja autentifikācijas marķieru un to pieteikšanās marķieru sniedzēju pamatinformācijas uzturēšanai. Tabula ir izveidota autentifikācijas un autorizācijas mehānismiem, kas tiek izmantoti ASP.NET Core Identity.

3.3. tabula

Datu bāzes ASP.NET Core Identity lietotāja autentifikācijas marķieru datu tabula (AspNetUserTokens)

<i>Lauks</i>	<i>Datu tips</i>	<i>Atslēga</i>	<i>Cita informācija</i>	<i>Apraksts</i>
<i>UserId</i>	<i>Int</i>	<i>PK, FK</i>	<i>Not null</i>	<i>Lietotāja unikāls identifikators</i>
<i>LoginProvider</i>	<i>Nvarchar(450)</i>	<i>PK</i>	<i>Not null</i>	<i>Pieteikšanās sniedzējs</i>
<i>Name</i>	<i>Nvarchar(450)</i>	<i>PK</i>	<i>Not null</i>	<i>Marķiera nosaukums</i>
<i>Value</i>	<i>Nvarchar(max)</i>	-	<i>Nullable</i>	<i>Marķiera vērtība</i>

Lietotāja autentifikācijas prasības datu tabula (sk. 3.4. tabulu) ir paredzēta lietotāja autentifikācijas prasību pamatinformācijas uzturēšanai. Tabula ir izveidota autentifikācijas un autorizācijas mehānismiem, kas tiek izmantoti ASP.NET Core Identity.

Datu bāzes ASP.NET Core Identity lietotāju autentifikācijas prasības datu tabula (AspNetUserClaims)

<i>Lauks</i>	<i>Datu tips</i>	<i>Atslēga</i>	<i>Cita informācija</i>	<i>Apraksts</i>
<i>Id</i>	<i>Int</i>	<i>PK</i>	<i>Auto increment Not null</i>	<i>Lietotāja prasību unikāls identifikators</i>
<i>UserId</i>	<i>Int</i>	<i>FK</i>	<i>Not null</i>	<i>Lietotāju unikāls identifikators</i>
<i>ClaimType</i>	<i>Nvarchar(max)</i>	-	<i>Nullable</i>	<i>Prasību veids</i>
<i>ClaimValue</i>	<i>Nvarchar(max)</i>	-	<i>Nullable</i>	<i>Prasību vērtība</i>

ASP.NET lomu datu tabula (sk. 3.5. tabulu) ir paredzēta ASP.NET lietotāju lomu pamatinformācijas uzturēšanai. Tabula ir izveidota autentifikācijas un autorizācijas mehānismiem, kas tiek izmantoti ASP.NET Core Identity.

Datu bāzes ASP.NET Core Identity lietotāju lomu datu tabula (AspNetRoles)

<i>Lauks</i>	<i>Datu tips</i>	<i>Atslēga</i>	<i>Cita informācija</i>	<i>Apraksts</i>
<i>Id</i>	<i>Int</i>	<i>PK</i>	<i>Auto increment Not null</i>	<i>ASP.NET lomas unikāls identifikators</i>
<i>Name</i>	<i>Nvarchar(256)</i>	-	<i>Nullable</i>	<i>Lomas nosaukums</i>
<i>NormalizedName</i>	<i>Nvarchar(256)</i>	-	<i>Nullable</i>	<i>Normalizēts lomas nosaukums</i>
<i>ConcurrencyStamp</i>	<i>Nvarchar(max)</i>	-	<i>Nullable</i>	<i>Laiksakritības zīmogs</i>

ASP.NET lomas prasību datu tabula (sk. 3.6. tabulu) ir paredzēta ASP.NET lietotāju lomu prasību pamatinformācijas uzturēšanai. Tabula ir izveidota autentifikācijas un autorizācijas mehānismiem, kas tiek izmantoti ASP.NET Core Identity.

3.6. tabula

Datu bāzes ASP.NET Core Identity lietotāju lomas prasību datu tabula (AspNetRoleClaims)

<i>Lauks</i>	<i>Datu tips</i>	<i>Atslēga</i>	<i>Cita informācija</i>	<i>Apraksts</i>
<i>Id</i>	<i>Int</i>	<i>PK</i>	<i>Auto increment Not null</i>	<i>ASP.NET lomas prasību unikāls identifikators</i>
<i>RoleId</i>	<i>Int</i>	<i>FK</i>	<i>Not null</i>	<i>ASP.Net lomas unikāls identifikators</i>
<i>ClaimType</i>	<i>Nvarchar(max)</i>	-	<i>Nullable</i>	<i>Prasību veids</i>
<i>ClaimValue</i>	<i>Nvarchar(max)</i>	-	<i>Nullable</i>	<i>Prasību vērtība</i>

ASP.NET lietotāju lomu datu tabula (sk. 3.7. tabulu) ir paredzēta relācijas daudz pret daudz izveidei starp lietotāju un ASP.NET lomu tabulām. Tabula ir izveidota autentifikācijas un autorizācijas mehānismiem, kas tiek izmantoti ASP.NET Core Identity.

3.7. tabula

Datu bāzes ASP.NET Core Identity lietotāju lomu datu tabula (AspNetUserRoles)

<i>Lauks</i>	<i>Datu tips</i>	<i>Atslēga</i>	<i>Cita informācija</i>	<i>Apraksts</i>
<i>UserId</i>	<i>Int</i>	<i>PK, FK</i>	<i>Not null</i>	<i>Lietotāju unikāls identifikators</i>
<i>RoleId</i>	<i>Int</i>	<i>PK, FK</i>	<i>Not null</i>	<i>ASP.NET lomas unikāls identifikators</i>

EntityFramework migrāciju vēstures datu tabula (sk. 3.8. tabulu) ir paredzēta Code First migrācijas vēstures pamatinformācijas uzturēšanai. Tabula un tabulas ieraksti tiek izveidoti automātiski, migrācijas pielietošanas laikā.

Datu bāzes EntityFramework migrāciju vēstures datu tabula (__EFMigrationsHistory)

<i>Lauks</i>	<i>Datu tips</i>	<i>Atslēga</i>	<i>Cita informācija</i>	<i>Apraksts</i>
<i>MigrationId</i>	<i>Nvarchar(150)</i>	<i>PK</i>	<i>Not null</i>	<i>Migrācijas unikāls identifikators (tiek veidots no laika zīmoga un nosaukuma)</i>
<i>ProductVersion</i>	<i>Nvarchar(32)</i>	-	<i>Not null</i>	<i>Produkta versija</i>

Organizāciju datu tabula (sk. 3.9. tabulu) ir paredzēta sistēmas organizāciju pamatinformācijas uzturēšanai.

Datu bāzes organizāciju datu tabula (Organizations)

<i>Lauks</i>	<i>Datu tips</i>	<i>Atslēga</i>	<i>Cita informācija</i>	<i>Apraksts</i>
<i>Id</i>	<i>Int</i>	<i>PK</i>	<i>Auto increment Not null</i>	<i>Organizācijas unikāls identifikators</i>
<i>OwnerId</i>	<i>Int</i>	<i>FK</i>	<i>Not null</i>	<i>Organizācijas īpašnieka (lietotāja) unikāls identifikators</i>
<i>Title</i>	<i>Nvarchar(40)</i>	-	<i>Not null</i>	<i>Organizācijas nosaukums</i>
<i>Description</i>	<i>Nvarchar(1000)</i>	-	<i>Nullable</i>	<i>Organizācijas apraksts</i>

Organizācijas dalībnieku datu tabula (sk. 3.10. tabulu) ir paredzēta relācijas daudz pret daudz izveidei starp sistēmas lietotājiem un sistēmas organizācijām.

Datu bāzes organizācijas dalībnieku datu tabula (OrganizationMembers)

<i>Lauks</i>	<i>Datu tips</i>	<i>Atslēga</i>	<i>Cita informācija</i>	<i>Apraksts</i>
<i>OrganizationId</i>	<i>Int</i>	<i>PK, FK</i>	<i>Not null</i>	<i>Organizācijas unikāls identifikators</i>
<i>MemberId</i>	<i>Int</i>	<i>PK, FK</i>	<i>Not null</i>	<i>Lietotāja unikāls identifikators</i>

Organizāciju lomu datu tabula (sk. 3.11. tabulu) ir paredzēta noteiktas organizācijas lomu pamatinformācijas uzturēšanai.

3.11. tabula

Datu bāzes organizāciju lomu datu tabula (ProcessRole)

<i>Lauks</i>	<i>Datu tips</i>	<i>Atslēga</i>	<i>Cita informācija</i>	<i>Apraksts</i>
<i>Id</i>	<i>Int</i>	<i>PK</i>	<i>Auto increment Not null</i>	<i>Lomas unikāls identifikators</i>
<i>OrganizationId</i>	<i>Int</i>	<i>FK</i>	<i>Not null</i>	<i>Organizācijas unikāls identifikators</i>
<i>Title</i>	<i>Nvarchar(40)</i>	-	<i>Not null</i>	<i>Lomas nosaukums</i>
<i>Description</i>	<i>Nvarchar(1000)</i>	-	<i>Nullable</i>	<i>Lomas apraksts</i>
<i>ColorCode</i>	<i>Nvarchar(10)</i>	-	<i>Not null</i>	<i>Lomas krāsas kods</i>

Lietotāju lomu datu tabula (sk. 3.12. tabulu) ir paredzēta relācijas daudz pret daudz izveidei starp sistēmas lietotājiem un organizācijas lomām.

3.12. tabula

Datu bāzes lietotāju lomu datu tabula (UserProcessRoles)

<i>Lauks</i>	<i>Datu tips</i>	<i>Atslēga</i>	<i>Cita informācija</i>	<i>Apraksts</i>
<i>UserId</i>	<i>Int</i>	<i>PK, FK</i>	<i>Not null</i>	<i>Lietotāja unikāls identifikators</i>
<i>ProcessRoleId</i>	<i>Int</i>	<i>PK, FK</i>	<i>Not null</i>	<i>Organizācijas lomas unikāls identifikators</i>

Organizāciju komandu datu tabula (sk. 3.13. tabulu) ir paredzēta noteiktas organizācijas komandu pamatinformācijas uzturēšanai.

3.13. tabula

Datu bāzes organizāciju komandu datu tabula (Teams)

<i>Lauks</i>	<i>Datu tips</i>	<i>Atslēga</i>	<i>Cita informācija</i>	<i>Apraksts</i>
--------------	------------------	----------------	-------------------------	-----------------

<i>Id</i>	<i>Int</i>	<i>PK</i>	<i>Auto increment</i> <i>Not null</i>	<i>Komandas unikāls</i> <i>identifikators</i>
<i>OrganizationId</i>	<i>Int</i>	<i>FK</i>	<i>Not null</i>	<i>Organizācijas unikāls</i> <i>identifikators</i>
<i>LeaderId</i>	<i>Int</i>	<i>FK</i>	<i>Not null</i>	<i>Komandas vadītāja</i> <i>(Lietotāja) unikāls</i> <i>identifikators</i>
<i>Title</i>	<i>Nvarchar(40)</i>	-	<i>Not null</i>	<i>Komandas nosaukums</i>
<i>Description</i>	<i>Nvarchar(1000)</i>	-	<i>Nullable</i>	<i>Komandas apraksts</i>

Komandas dalībnieku datu tabula (sk. 3.14. tabulu) ir paredzēta relācijas daudz pret daudz izveidei starp sistēmas lietotājiem un organizācijas komandām.

3.14. tabula

Datu bāzes komandas dalībnieku datu tabula (TeamMembers)

<i>Lauks</i>	<i>Datu tips</i>	<i>Atslēga</i>	<i>Cita informācija</i>	<i>Apraksts</i>
<i>TeamId</i>	<i>Int</i>	<i>PK, FK</i>	<i>Not null</i>	<i>Komandas unikāls</i> <i>identifikators</i>
<i>MemberId</i>	<i>Int</i>	<i>PK, FK</i>	<i>Not null</i>	<i>Komandas dalībnieka</i> <i>(Lietotāja) unikāls</i> <i>identifikators</i>

Komandas lomu datu tabula (sk. 3.15. tabulu) ir paredzēta relācijas daudz pret daudz izveidei starp organizācijas lomām un organizācijas komandām.

3.15. tabula

Datu bāzes komandas lomu datu tabula (TeamRoles)

<i>Lauks</i>	<i>Datu tips</i>	<i>Atslēga</i>	<i>Cita informācija</i>	<i>Apraksts</i>
<i>TeamId</i>	<i>Int</i>	<i>PK, FK</i>	<i>Not null</i>	<i>Komandas unikāls</i> <i>identifikators</i>
<i>ProcessRoleId</i>	<i>Int</i>	<i>PK, FK</i>	<i>Not null</i>	<i>Organizācijas lomas unikāls</i> <i>identifikators</i>

Komandas pazīmju datu tabula (sk. 3.16. tabulu) ir paredzēta noteiktas komandas pazīmju pamatinformācijas uzturēšanai.

3.16. tabula

Datu bāzes komandas pazīmju datu tabula (Features)

<i>Lauks</i>	<i>Datu tips</i>	<i>Atslēga</i>	<i>Cita informācija</i>	<i>Apraksts</i>
<i>Id</i>	<i>Int</i>	<i>PK</i>	<i>Auto increment Not null</i>	<i>Pazīmes unikāls identifikators</i>
<i>TeamId</i>	<i>Int</i>	<i>FK</i>	<i>Not null</i>	<i>Komandas unikāls identifikators</i>
<i>Code</i>	<i>Nvarchar(8)</i>	-	<i>Not null</i>	<i>Pazīmes kods</i>
<i>Title</i>	<i>Nvarchar(40)</i>	-	<i>Not null</i>	<i>Pazīmes nosaukums</i>
<i>Description</i>	<i>Nvarchar(max)</i>	-	<i>Nullable</i>	<i>Pazīmes apraksts</i>
<i>DevelopmentStart</i>	<i>Datetime2(7)</i>	-	<i>Not null</i>	<i>Pazīmes izstrādes uzsākšanas datums</i>
<i>DevelopmentDuration</i>	<i>Datetime2(7)</i>	-	<i>Not null</i>	<i>Pazīmes izstrādes ilgums</i>
<i>Completed</i>	<i>Bit</i>	-	<i>Not null</i>	<i>Ir pazīmes izstrāde pabeigta</i>

Uzdevuma stāvokļu datu tabula (sk. 3.17. tabulu) ir paredzēta uzdevuma stāvokļu pamatinformācijas uzturēšanai.

3.17. tabula

Datu bāzes uzdevuma stāvokļu datu tabula (TaskStatuses)

<i>Lauks</i>	<i>Datu tips</i>	<i>Atslēga</i>	<i>Cita informācija</i>	<i>Apraksts</i>
<i>Id</i>	<i>Int</i>	<i>PK</i>	<i>Auto increment Not null</i>	<i>Uzdevuma stāvokļa unikāls identifikators</i>
<i>Title</i>	<i>Nvarchar(40)</i>	-	<i>Not null</i>	<i>Uzdevuma stāvokļa nosaukums</i>
<i>Description</i>	<i>Nvarchar(max)</i>	-	<i>Nullable</i>	<i>Uzdevuma stāvokļa apraksts</i>

Uzdevumu datu tabula (sk. 3.18. tabulu) ir paredzēta uzdevumu pamatinformācijas uzturēšanai.

3.18. tabula

Datu bāzes uzdevumu datu tabula (Tasks)

<i>Lauks</i>	<i>Datu tips</i>	<i>Atslēga</i>	<i>Cita informācija</i>	<i>Apraksts</i>
<i>Id</i>	<i>Int</i>	<i>PK</i>	<i>Auto increment Not null</i>	<i>Uzdevuma unikāls identifikators</i>
<i>FeatureId</i>	<i>Int</i>	<i>FK</i>	<i>Not null</i>	<i>Pazīmes unikāls identifikators</i>
<i>AssigneeId</i>	<i>Int</i>	<i>FK</i>	<i>Nullable</i>	<i>Uzdevuma izpildītāja (Lietotāja) unikāls identifikators</i>
<i>AssigneeRoleId</i>	<i>Int</i>	<i>FK</i>	<i>Nullable</i>	<i>Uzdevuma izpildītāja lomas unikāls identifikators</i>
<i>TaskStatusId</i>	<i>Int</i>	<i>FK</i>	<i>Not null</i>	<i>Uzdevuma stāvokļa unikāls identifikators</i>
<i>Code</i>	<i>Nvarchar(12)</i>	-	<i>Not null</i>	<i>Uzdevuma kods</i>
<i>Title</i>	<i>Nvarchar(255)</i>	-	<i>Not null</i>	<i>Uzdevuma nosaukums</i>
<i>Description</i>	<i>Nvarchar(max)</i>	-	<i>Nullable</i>	<i>Uzdevuma apraksts</i>
<i>Completed</i>	<i>Bit</i>	-	<i>Not null</i>	<i>Ir uzdevums izpildīts</i>

Uzdevuma atsauces veidu datu tabula (sk. 3.19. tabulu) ir paredzēta uzdevuma atsauces veidu pamatinformācijas uzturēšanai.

3.19. tabula

Datu bāzes uzdevuma atsauces veidu datu tabula (ReferenceTypes)

<i>Lauks</i>	<i>Datu tips</i>	<i>Atslēga</i>	<i>Cita informācija</i>	<i>Apraksts</i>
<i>Id</i>	<i>Int</i>	<i>PK</i>	<i>Auto increment Not null</i>	<i>Uzdevuma atsauces veida unikāls identifikators</i>
<i>Title</i>	<i>Nvarchar(40)</i>	-	<i>Not null</i>	<i>Uzdevuma atsauces veida nosaukums</i>

<i>Description</i>	<i>Nvarchar(max)</i>	-	<i>Nullable</i>	<i>Uzdevuma atsauces veida apraksts</i>
--------------------	----------------------	---	-----------------	---

Uzdevumu atsauču datu tabula (sk. 3.20. tabulu) ir paredzēta relācijas daudz pret daudz izveidei starp uzdevumiem, atsauces uzdevumiem un uzdevuma atsauces veidiem.

3.20. tabula

Datu bāzes uzdevumu atsauču datu tabula (TaskReferences)

<i>Lauks</i>	<i>Datu tips</i>	<i>Atslēga</i>	<i>Cita informācija</i>	<i>Apraksts</i>
<i>TaskId</i>	<i>Int</i>	<i>PK, FK</i>	<i>Not null</i>	<i>Uzdevuma unikāls identifikators</i>
<i>ReferenceTaskId</i>	<i>Int</i>	<i>PK, FK</i>	<i>Not null</i>	<i>Atsauces uzdevuma unikāls identifikators</i>
<i>ReferenceTypeId</i>	<i>Int</i>	<i>PK, FK</i>	<i>Not null</i>	<i>Uzdevuma atsauces veida unikāls identifikators</i>
<i>AdditionalInfo</i>	<i>Nvarchar(max)</i>	-	<i>Nullable</i>	<i>Atsauces papildus informācija</i>

Kalendāra ierakstu datu tabula (sk. 3.21. tabulu) ir paredzēta kalendāra ierakstu pamatinformācijas uzturēšanai.

3.21. tabula

Datu bāzes kalendāra ierakstu datu tabula (CalendarEntries)

<i>Lauks</i>	<i>Datu tips</i>	<i>Atslēga</i>	<i>Cita informācija</i>	<i>Apraksts</i>
<i>Id</i>	<i>Int</i>	<i>PK</i>	<i>Auto increment Not null</i>	<i>Kalendāra ieraksta unikāls identifikators</i>
<i>UserId</i>	<i>Int</i>	<i>FK</i>	<i>Not null</i>	<i>Uzdevuma izpildītāja (Lietotāja) unikāls identifikators</i>
<i>TaskId</i>	<i>Int</i>	<i>FK</i>	<i>Not null</i>	<i>Izpildāmā uzdevuma unikāls identifikators</i>

<i>DevelopmentStart</i>	<i>Datetime2(7)</i>	-	<i>Not null</i>	<i>Izstrādes posma sākums</i>
<i>DevelopmentDuration</i>	<i>Datetime2(7)</i>	-	<i>Not null</i>	<i>Izstrādes posma ilgums</i>

Paziņojumu datu tabula (sk. 3.22. tabulu) ir paredzēta paziņojumu pamatinformācijas uzturēšanai.

3.22. tabula

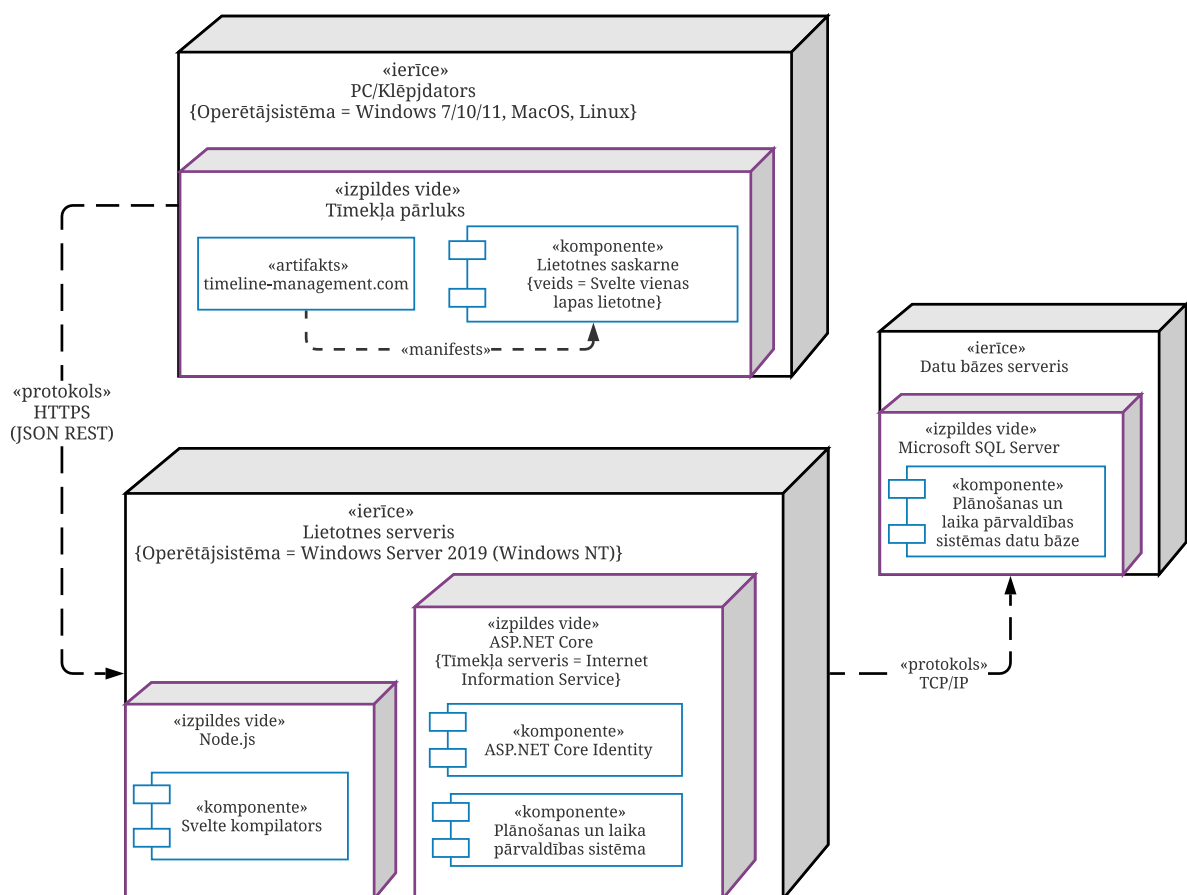
Datu bāzes paziņojumu datu tabula (Notifications)

<i>Lauks</i>	<i>Datu tips</i>	<i>Atslēga</i>	<i>Cita informācija</i>	<i>Apraksts</i>
<i>Id</i>	<i>Int</i>	<i>PK</i>	<i>Auto increment Not null</i>	<i>Paziņojuma unikāls identifikators</i>
<i>ReceiverId</i>	<i>Int</i>	<i>FK</i>	<i>Not null</i>	<i>Saņēmēja (Lietotāja) unikāls identifikators</i>
<i>SenderId</i>	<i>Int</i>	<i>FK</i>	<i>Nullable</i>	<i>Sūtītāja (Lietotāja) unikāls identifikators</i>
<i>OrganizationId</i>	<i>Int</i>	<i>FK</i>	<i>Not null</i>	<i>Organizācijas unikāls identifikators</i>
<i>ProcessRoleId</i>	<i>Int</i>	<i>FK</i>	<i>Nullable</i>	<i>Organizācijas lomas unikāls identifikators</i>
<i>TeamId</i>	<i>Int</i>	<i>FK</i>	<i>Nullable</i>	<i>Organizācijas komandas unikāls identifikators</i>
<i>TaskId</i>	<i>Int</i>	<i>FK</i>	<i>Nullable</i>	<i>Uzdevuma unikāls identifikators</i>
<i>NotificationSetGuid</i>	<i>Uniqueidentifier</i>	-	<i>Not null</i>	<i>Paziņojuma kopas vispārējs unikāls identifikators</i>
<i>IsRead</i>	<i>Bit</i>	-	<i>Not null</i>	<i>Ir paziņojums izlasīts</i>
<i>IsHidden</i>	<i>Bit</i>	-	<i>Not null</i>	<i>Ir paziņojums paslēpts saņēmējam</i>
<i>Content</i>	<i>Nvarchar(max)</i>	-	<i>Not null</i>	<i>Paziņojuma saturs</i>

Lauks	Datu tips	Atslēga	Cita informācija	Apraksts
SentTime	Datetime2(7)	-	Not null	Paziņojuma sūtīšanas laiks
ActionEndpoint	Nvarchar(max)	-	Nullable	Paziņojuma atbildes apstrādes funkcijas galapunkts

3.2. Daļējs funkciju projektējums

3.2.1. Programmatūras izvietojuma diagramma



3.3. att. Plānošanas un laika pārvaldības sistēmas programmatūras izvietojuma diagramma

Programmatūras izvietojuma diagrammā (sk. 3.3. att.) ir attēlotas trīs ierīces, kas savstarpēji mijiedarbosies vienas sistēmas ietvaros, tas ir, lietotāja personālais datordators/klējdatordators (turpmāk lietotāja datordators), lietotņu serveris un datu bāzes serveris.

Ja lietotājs vēlas izmantot sistēmu, tad ir sagaidīts kā lietotāja datorā ir uzinstalēts tīmekļa pārlūks, kurš turpmāk kalpo kā izpildes vide. Ievadot adreses joslā tīmekļa lapaspuses URL, lietotājam tīmekļa pārlūka tiek ielādēta Svelte vienas lapas lietotne. Projekta ietvaros tiks izmantota Svelte tehnoloģija tīmekļa vietnes atveidošanai.

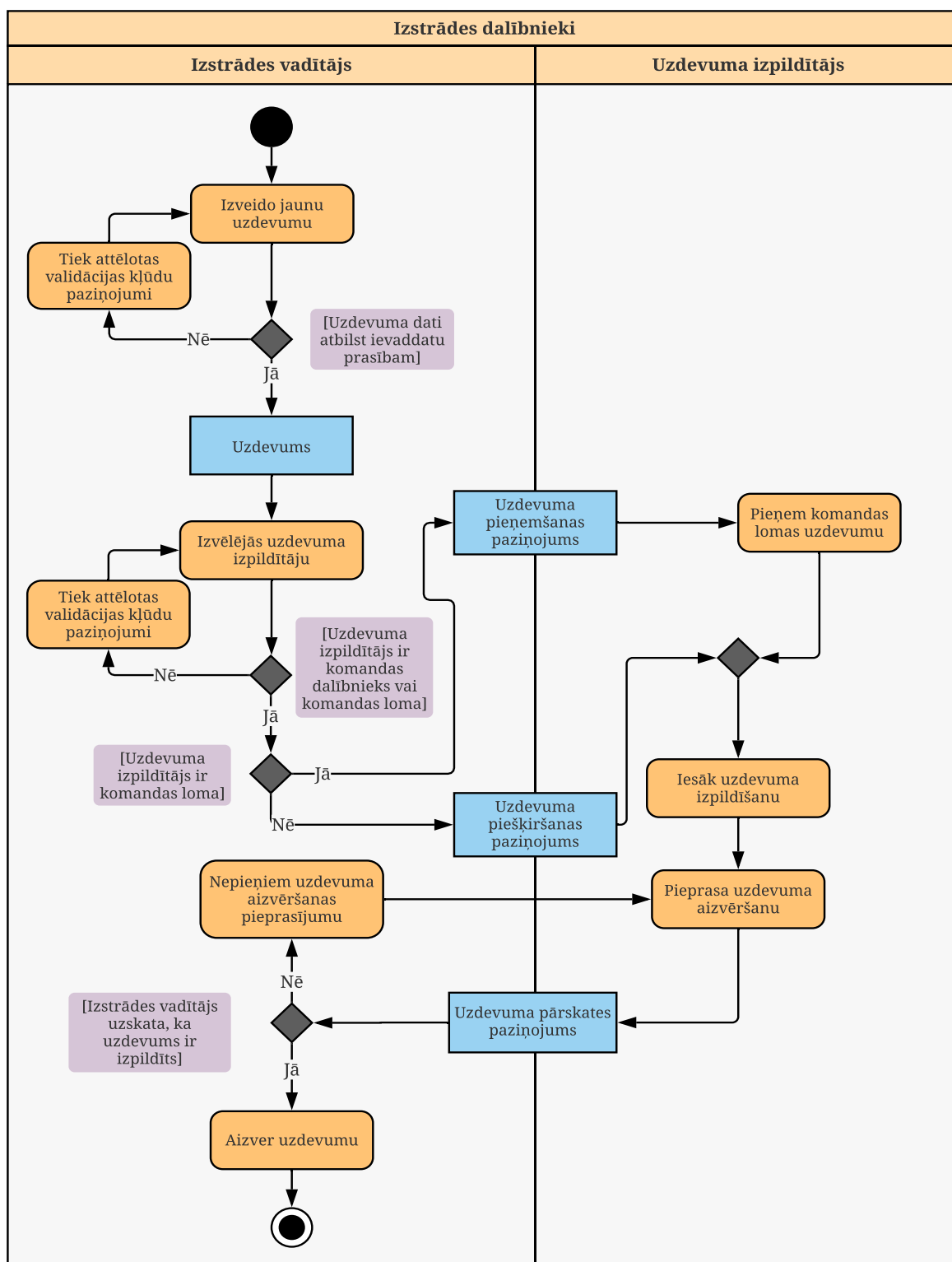
Izmantojot piedāvātu lietotnes saskarni, lietotājam ir iespējams veidot pieprasījumus lietotnes serverim, lai pieprasītu informāciju (piemēram, pieprasīt informāciju par noteiktu organizāciju, lomu, komandu u.t.t) kā arī lai piedāvātu informāciju (piemēram, piedāvāt sava lietotāja konta pieteikšanas vārdu un paroli, lai piekļūtu sistēmai). Informācija tiek padota lietotnes serverim, izmantojot HTTPS protokolu, kur pieprasījuma saturs tiks formatēts attiecīgi JSON formātā. Lietotnes saskarne ir atkarīga no informācijas, kas tiks piedāvāta no lietotnes servera, un ja lietotnes saskarnei nav iespējas saņemt informāciju no lietotnes servera, tad lietotājam nav iespējams mijiedarboties ar sistēmas funkcionalitāti.

Lietotnes serveris ir ierīce, kura veiks datu apstrādi. Lietotnes serveris satur divas izpildes vides. Pirmā izpildes vide ir Node.js vide, kur Svelte tīmekļa vietne tiks kompilēta un atsūtīta lietotājam. Otrā izpildes vide ir ASP.NET Core, kur tiks realizēts autentifikācijas un autorizācijas mehānisms ASP.NET Core Identity un tiks realizēta sistēmas datu apstrādes loģika.

Lietotnes serveris datu apstrādes gaitā veido vairākus pieprasījumus datu bāzes serverim, izmantojot TCP/IP protokolu. Lietotnes serveris ir atkarīgs no informācijas, kas tiek glabāta datu bāzes serverī, tāpēc ja lietotnes serverim nav iespējas saņemt informāciju no datu bāzes servera, tad lietotnes serveris neveiks datu apstrādi.

Datu bāzes serverī tiek izmantota Microsoft SQL Server izpildes vide, lai glabātu plānošanas un laika pārvaldības sistēmas informāciju. Datu bāzes serveris nav atkarīgs no sistēmas ierīcēm.

3.2.2. Uzdevuma izpildes procesa aktivitāšu diagramma



3.4. att. Uzdevuma izpildes procesa aktivitāšu diagramma

Aktivitāšu diagrammā (sk. 3.4. att.) ir attēlota uzdevuma izveidošanas un izpildes gaita. Aktivitāšu diagrammā ir iekļauti divi aktori: izstrādes vadītājs un uzdevuma izpildītājs. Abi aktori ir vienas organizācijas komandas dalībnieki. Uzdevuma izpildes procesa aktivitāšu

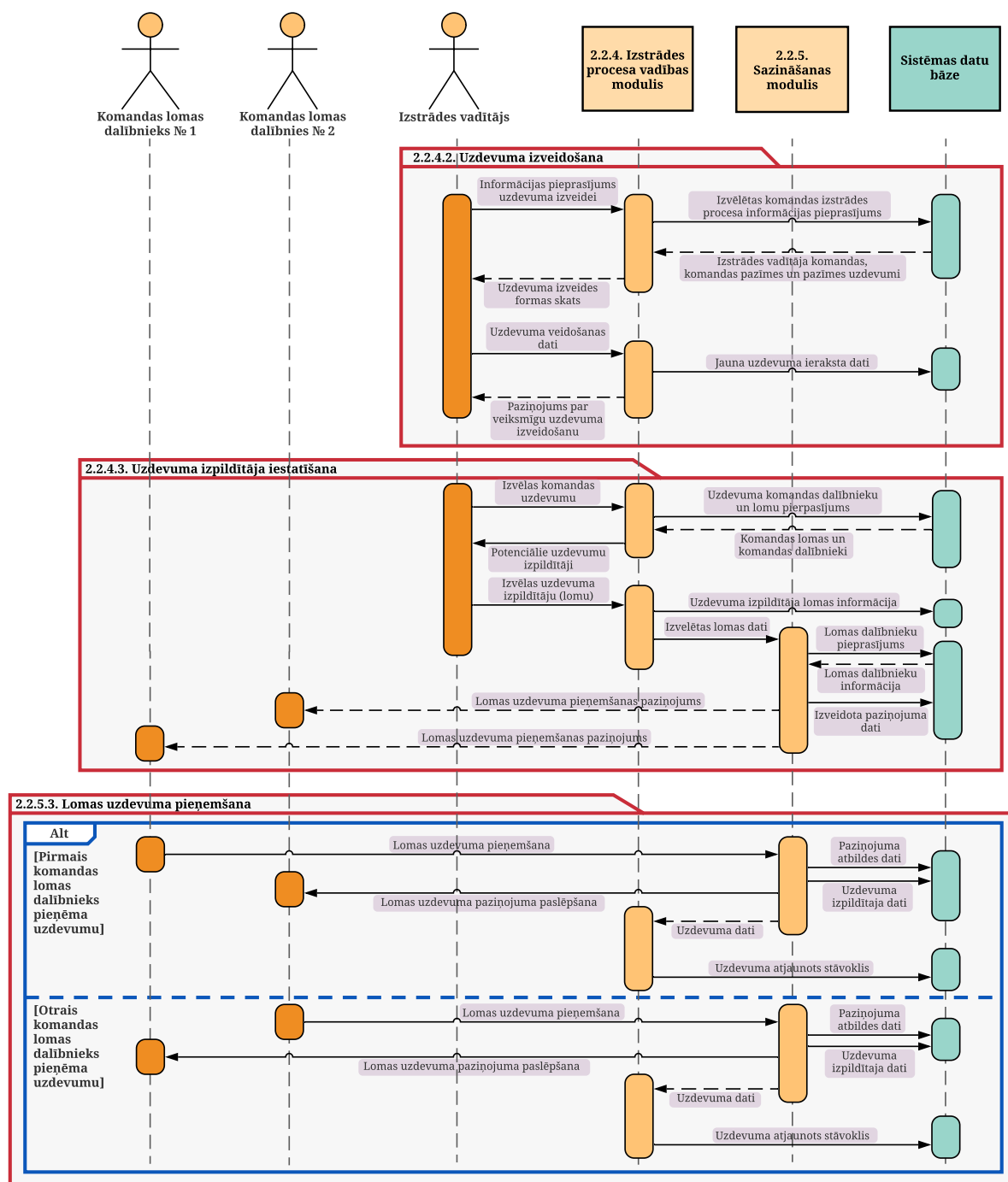
diagrammā tiek izmantoti 2.2.4. *Izstrādes procesa vadības modulis* un 2.2.5. *Sazināšanas modulis*.

Lai izveidotu uzdevumu, izstrādes vadītājs izmanto funkciju 2.2.4.2. *Uzdevuma izveidošana / datu rediģēšana* (tiek izmantota tieši izveidošanas funkcijas funkcionalitāte). Uzdevuma izpildītāja izvēlei tiek izmantota funkcija 2.2.4.3. *Uzdevuma izpildītāja iestatīšana*. Izstrādes vadītājs saņem paziņojumu par uzdevuma aizvēršanu no funkcijas 2.2.5.1. *Lietotāja paziņojuma izveide* un aizver uzdevumu, ja uzskata, ka uzdevums ir izpildīts, izmantojot funkciju 2.2.4.4. *Uzdevuma stāvokļa iestatīšana*.

Ja izstrādes vadītājs izvelējās kā uzdevuma izpildītāju iestatīt komandas lomu, tad paziņojums tiek izveidots, izmantojot 2.2.5.2. *Komandas lomas paziņojuma izveide* un komandas lomas dalībnieks (aktivitāšu diagrammā tas ir uzdevuma izpildītājs) pieņem uzdevumu, izmantojot funkciju 2.2.5.3. *Lomas uzdevuma pieņemšana*. Ja izstrādes vadītājs izvelējās kā uzdevuma izpildītāju iestatīt komandas dalībnieku, tad komandas dalībniekam (aktivitāšu diagrammā tas ir uzdevuma izpildītājs) tiek izveidots paziņojums par uzdevuma piešķiršanu, izmantojot funkciju 2.2.5.1. *Lietotāja paziņojuma izveide*. Uzdevuma izpildītājs iesāc uzdevuma izpildi, izveidojot kalendāra ierakstu noteiktajam uzdevumam. Uzdevuma izpildītājs paveic šo darbu, izmantojot funkciju 2.2.4.5. *Kalendāra ieraksta izveidošana /datu rediģēšana* (tiek izmantota tieši izveidošanas funkcijas funkcionalitāte). Uzdevuma izpildītājs pieprasa uzdevuma aizvēršanu, izsaucot funkciju 2.2.4.4. *Uzdevuma stāvokļa iestatīšana*.

Aktivitāšu diagrammā tiek izveidoti trīs paziņojuma objekti, tomēr tas ir vienas entītijas ieraksti, kur atšķiras satura lauks.

3.2.3. Lomas uzdevuma izveidošanas secību diagramma



3.5. att. Lomas uzdevuma izveidošanas secību diagramma

Secību diagrammā (sk. 3.5. att.) ir attēlota izstrādes procesa izveidošanas gaita, kad izstrādes vadītājs izvēlās uzdevumu izveidot noteiktai komandas lomai.

Secību diagrammā ir iekļauti trīs aktori: komandas lomas dalībnieks №1, komandas lomas dalībnieks №2 un izstrādes vadītājs. Pirmie divi aktori ir noteiktas komandas dalībnieki, kuriem tika piešķirta viena un tā paša loma, savukārt izstrādes vadītājs ir organizācijas īpašnieks

vai izvēlētas komandas vadītājs. Secību diagrammas process ir attēlots vienas organizācijas komandas ietvaros.

Secību diagrammā arī tiek attēloti tādi objekti kā *2.2.4 Izstrādes procesa vadības modulis*, *2.2.5. Sazināšanas modulis* un plānošanas un laika pārvaldības sistēmas datu bāze. Diagramma tika sadalīta trīs daļās, kur katra daļa attēlo noteiktas funkcijas datu apstrādes procesu. Lomas uzdevuma izveidošanas secību diagrammā tika attēlotas sekojošas funkcijas:

1. *2.2.4.2. Uzdevuma izveidošana / datu rediģēšana*

- Secību diagrammā tika attēlots tieši uzdevuma izveidošanas funkcionalitāte un netika iekļauti kļūdu ziņojumi, kas tiks nosūtīti lietotājam, ja ievaddatu apstrādes gaitā tiks konstatēts, ka ievaddati neatbilst funkcijas ievaddatu prasībām.
- Funkcijas izpildē piedalās tieši viens aktors, izstrādes vadītājs, un uzdevumu izveidošanai ir nepieciešams mijiedarboties ar izstrādes procesa vadības moduli un sistēmas datu bāzi.

2. *2.2.4.3. Uzdevuma izpildītāja iestatīšana*

- Secību diagrammā kā izveidota uzdevuma izpildītāju tika izvēlēts noteiktas komandas loma, kura ir piešķirta diviem aktoriem (komandas lomas dalībnieks №1, komandas lomas dalībnieks №2).
- Diagrammā netika iekļauti kļūdu ziņojumi, kas tiks nosūtīti lietotājam, ja ievaddatu apstrādes gaitā tiks konstatēts, ka ievaddati neatbilst funkcijas ievaddatu prasībām.
- Funkcijas izpildē piedalās visi trīs aktori un visi attēlotie sistēmas objekti. Izstrādes vadītājs ar izstrādes procesa vadības moduļa palīdzību iestāta uzdevuma izpildītāju un, izmantojot sazināšanas moduli, paziņojumi par lomas pieņemšanu ir atsūtīti komandas lomas dalībniekiem.

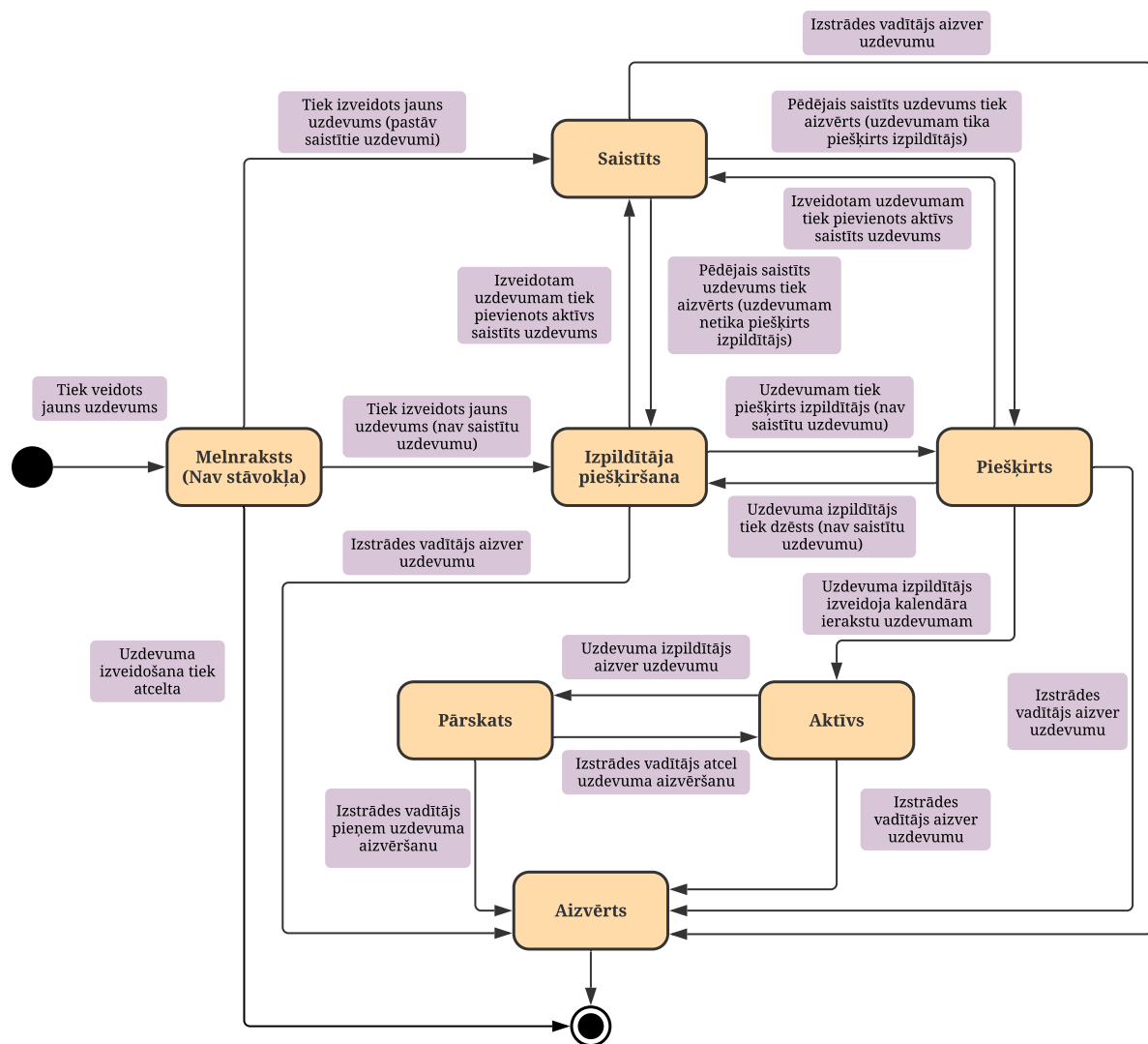
3. *2.2.5.3. Lomas uzdevuma pieņemšana*

- Secību diagrammā ir attēlotas divas līdzīgas situācijas, kad lomas uzdevumu pieņem izpildei pirmais vai otrais komandas lomas dalībnieks.
- Funkcijas izpildē piedalās divi aktori (komandas lomas dalībnieki) un visi attēlotie sistēmas objekti. Kad komandas lomas dalībnieks pieņem uzdevumu, sazināšanas modulis reģistrē paziņojuma atbildi un uzdevuma izpildītāju un

slēpj paziņojumu par lomas uzdevuma pieņemšanu, kas tika atsūtīts pārējiem komandas lomas dalībniekiem. Ar izstrādes procesa vadības palīdzību datu bāzē tika atjaunots pieņemtā uzdevuma stāvoklis.

- Diagrammā netika iekļauti kļūdu ziņojumi, kas tiks nosūtīti lietotajam, ja ievaddatu apstrādes gaitā tiks konstatēts, ka ievaddati neatbilst funkcijas ievaddatu prasībām.

3.2.4. Izstrādes procesa uzdevumu stāvokļu pāreju diagramma



3.6. att. Izstrādes procesa uzdevumu stāvokļu pāreju diagramma

Izstrādes procesa uzdevumu stāvokļu pāreju diagramma (sk. 3.6. att.) attēlo stāvokļu pārejas, kas tika aprakstītas entītijai *Uzdevumu stāvoklis* sekcijā 2.1. *Konceptuālais datu bāzes apraksts* un kuru piešķiršanas procesa gaita tika aprakstīta sekcijā 2.2.4.4. *Uzdevuma stāvokļa*

iestatīšana. Stāvokļu pāreja diagramma satur septiņus stāvokļus: *Melnraksts (Nav stāvokļa)*, *Saistīts*, *Izpildītāja piešķiršana*, *Piešķirts*, *Aktīvs*, *Pārskats*, *Aizvērts*.

3.2.4.1 Uzdevuma stāvoklis: Melnraksts (Nav stāvokļa)

Sākumstāvoklī izstrādes vadītājam ir iespēja izvēlēties izveidot jaunu izstrādes procesa uzdevumu, kas stāvokļu pāreju diagrammā izveido pāreju uz stāvokli *Melnraksts (Nav stāvokļa)*. Minētā stāvoklī izstrādes vadītājam tiek piedāvāta uzdevuma izveidošanas forma, kur nepieciešams norādīt informāciju, kura atbilst funkcijas 2.2.4.2 *Uzdevuma izveidošana/datu rediģēšana* ievaddatiem, lai turpmāk izveidotu uzdevumu. Stāvoklim tika pievienots pēcraksts *Nav stāvokļa*, jo uzdevuma entītijas ierakstā stāvokļa lauks netiek aizpildīts un sistēmā neeksistē attiecīga stāvokļa, kurš tiek iestatīts uzdevumam melnrakstu veidošanas posmā.

Ja, aizpildot un iesniedzot formu, izstrādes vadītājs izveidoja atsauces uz aktīviem uzdevumiem ar atsauces veidu *Ķēdes veidošanas atsauce*, tad stāvokļu pāreju diagramma notiek pāreja no stāvokļa *Melnraksts (Nav stāvokļa)* uz stāvokli *Saistīts*. Ja atsauce ar minētu atsauces veidu netika pievienota, tad notiek pāreja no stāvokļa *Melnraksts (Nav stāvokļa)* uz stāvokli *Izpildītāja piešķiršana*.

Ja izstrādes vadītājs izvelējas atcelt uzdevuma izveidošanu, tad no stāvokļa *Melnraksts (Nav stāvokļa)* notiek pāreja uz beigu stāvokli.

3.2.4.2. Uzdevuma stāvoklis: Izpildītāja piešķiršana

Stāvoklī *Izpildītāja piešķiršana* uzdevums tika izveidots, reģistrēts sistēmā un sistēmas datu bāzē, bet uzdevumam vēl netika piešķirts uzdevuma izpildītājs. Ja uzdevums atrodas minētā stāvoklī tas nozīmē, ka uzdevumam nepastāv atsauces uz aktīviem uzdevumiem (atsauces veids: *Ķēdes veidošanas atsauce*).

Ja izstrādes vadītājs izvelējas rediģēt uzdevuma informāciju un ir pievienojis jaunu ķēdes veidošanas atsauci uz uzdevumu, kuram nav stāvoklis *Aizvērts*, tad stāvokļu pāreju diagrammā notiek pāreja no stāvokļa *Izpildītāja piešķiršana* uz stāvokli *Saistīts*.

Ja izstrādes vadītājs pievienoja uzdevuma izpildītāju, izmantojot funkciju 2.2.4.3 *Uzdevuma izpildītāja iestatīšana*, un uzdevumam nav saistītu uzdevumu, tad:

- Ja uzdevums tika piešķirts noteiktam komandas dalībniekam, tad pēc funkcijas izpildes stāvokļu pāreju diagrammā notiek pāreja no stāvokļa *Izpildītāja piešķiršana* uz stāvokli *Piešķirts*.

- Ja uzdevums tika piešķirts noteiktai komandas lomai, tad pāreja no stāvokļa *Izpildītāja piešķiršana* uz stāvokli *Piešķirts* notiek tikai tad, kad komandas lomas dalībnieks izvēlējas pieņemt uzdevumu izpildi.

Ja izstrādes vadītājs izvēlējas aizvērt uzdevumu, tad notiek pāreja no stāvokļa *Izpildītāja piešķiršana* uz stāvokli *Aizvērts*.

3.2.4.3. Uzdevuma stāvoklis: Piešķirts

Stāvoklī *Piešķirts* uzdevumam tika izvēlēts uzdevuma izpildītājs, bet uzdevuma izpildītājs vēl neregistrēja sistēmā kalendāra ierakstu ar izvēlēto uzdevumu. Ja uzdevums atrodas minētā stāvoklī tas nozīmē, ka uzdevumam nepastāv atsaucis uz aktīviem uzdevumiem (atsauces veids: *Ķēdes veidošanas atsaucis*)

Ja izstrādes vadītājs izvēlējas rediģēt uzdevuma informāciju un ir pievienojis jaunu ķēdes veidošanas atsaucis uz uzdevumu, kuram nav stāvoklis *Aizvērts*, tad stāvokļu pāreju diagrammā notiek pāreja no stāvokļa *Piešķirts* uz stāvokli *Saistīts*.

Ja izstrādes vadītājs izvēlējas izdzēst uzdevuma izpildītāju, tad stāvokļu pāreju diagrammā notiek pāreja no stāvokļa *Piešķirts* uz stāvokli *Izpildītāja piešķiršana* (Uzdevumam nepastāv saistītu uzdevumu).

Ja uzdevumu izpildītājs sistēmā registrēja kalendāra ierakstu, izmantojot funkciju 2.2.4.5 *Kalendāra ieraksta izveidošana/datu rediģēšana*, tad notiek pāreja no stāvokļa *Piešķirts* uz stāvokli *Aktīvs*.

Ja izstrādes vadītājs izvēlējas aizvērt uzdevumu, tad notiek pāreja no stāvokļa *Piešķirts* uz stāvokli *Aizvērts*.

3.2.4.4. Uzdevumu stāvoklis: Saistīts

Stāvoklī *Saistīts* uzdevumam ir izveidotas atsaucis ar atsaucis veidu *Ķēdes veidošanas atsaucis* uz uzdevumiem, kuriem stāvoklis nav *Aizvērts*. Tas nozīmē, kā uzdevums vēl nav atvērts izpildei un atrodas gaidīšanas režīmā, kamēr izpildīsies (aizvērsies) uzdevumi, uz kuriem tika izveidota atsaucis.

Ja pēdējais saistīts uzdevums tika aizvērts, tad tiek pārbaudīts, vai uzdevumam pastāv uzdevuma izpildītājs. Ja uzdevuma izpildītājs pastāv, tad notiek pāreja no stāvokļa *Saistīts* uz stāvokli *Piešķirts* un uzdevums ir atvērts izpildei. Ja uzdevuma izpildītājs nepastāv vai

uzdevuma izpildītājs ir komandas loma, tad notiek pāreja no stāvokļa *Saistīts* uz stāvokli *Izpildītāja piešķiršana*.

Ja izstrādes vadītājs izvelējas aizvērt uzdevumu, tad notiek pāreja no stāvokļa *Saistīts* uz stāvokli *Aizvērts*.

3.2.4.5. Uzdevumu stāvoklis: Aktīvs

Stāvoklī *Aktīvs* uzdevumu izpildītājs izpilda izvelēto uzdevumu. Tas nozīmē kā uzdevumu izpildītājs izveidoja vismaz vienu kalendāra ierakstu ar izvelēto uzdevumu.

Ja uzdevumu izpildītājs izvelējas aizvērt uzdevumu (Uzdevumu izpildītājs uzskata, ka uzdevums ir izpildīts), tad stāvokļu pāreju diagrammā notiek pāreja no stāvokļa *Aktīvs* uz stāvokli *Pārskats*.

Ja izstrādes vadītājs izvelējas aizvērt uzdevumu, tad notiek pāreja no stāvokļa *Aktīvs* uz stāvokli *Aizvērts*.

3.2.4.6. Uzdevuma stāvoklis: Pārskats

Stāvoklī *Pārskats* uzdevums ir nodots izstrādes vadītājam uz pārbaudi. Ja izstrādes vadītājs uzskata, ka uzdevums ir izpildīts, tad izstrādes vadītājs aizver uzdevumu. Stāvokļu pāreju diagramma tiek izveidota pāreja no stāvokļa *Pārskats* uz stāvokli *Aizvērts*.

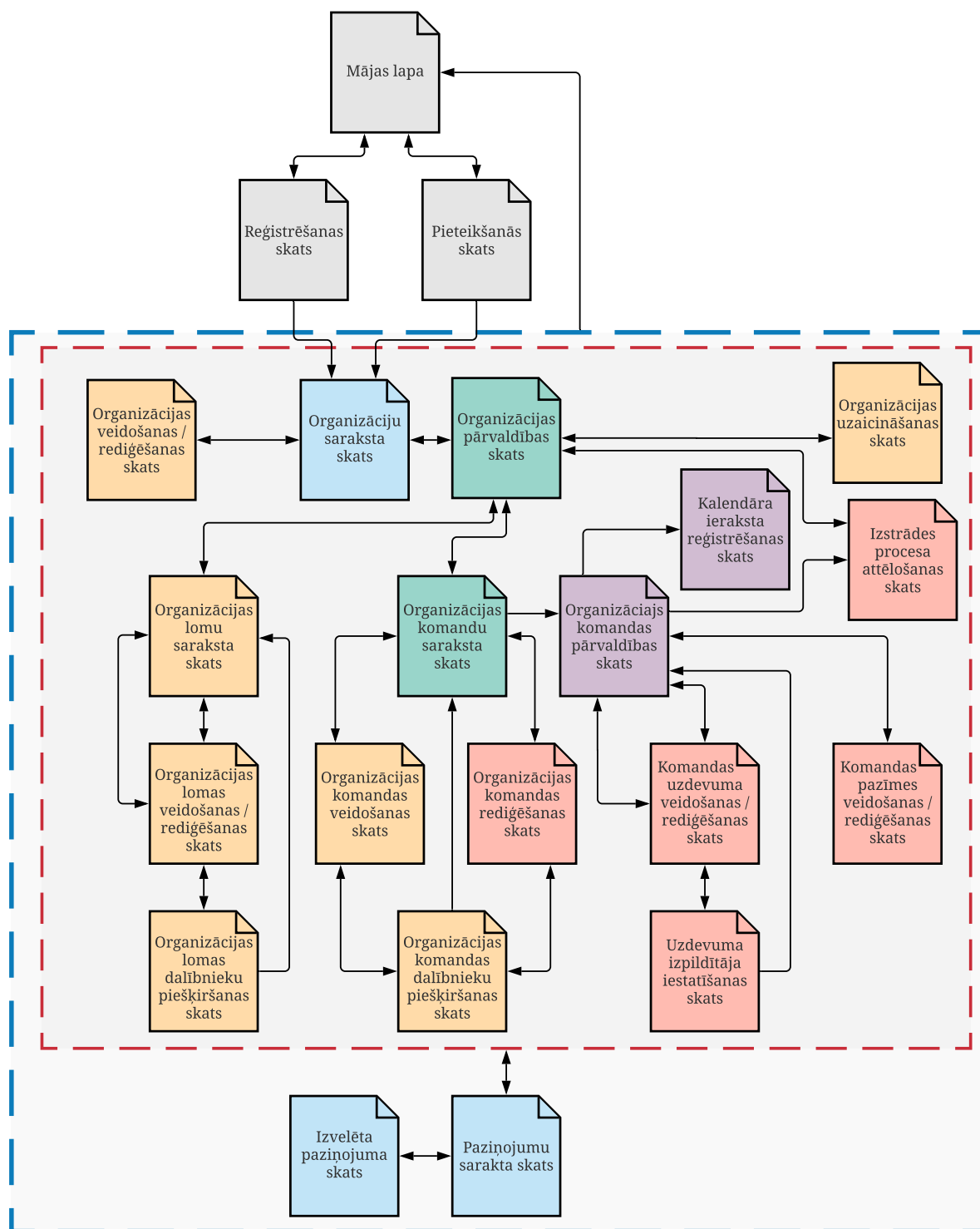
Ja izstrādes vadītājs uzskata, ka uzdevums nav izpildīts, tad izstrādes vadītājs atcel uzdevuma aizvēšanas pieprasījumu un stāvokļu pāreju diagrammā tiek izveidota pāreja no stāvokļa *Pārskats* uz stāvokli *Aktīvs*.

3.2.4.7. Uzdevuma stāvoklis: Aizvērts

Stāvoklī *Aizvērts* uzdevums tiek uzskatīts par izpildītu (arī kad uzdevums tika aizvērts stāvoklī *Piešķirts*, *Izpildītājas piešķiršana*, *Saistīts*) un nonākot stāvoklī stāvokļu pāreju diagramma uzreiz notiek pāreja uz beigu stāvokli.

3.3. Daļējs lietotāja saskarņu projektējums

3.3.1. Lietotāju saskarnes pārvietošanas diagramma



3.7. att. Lietotāju saskarnes pārvietošanas diagramma

Lietotāju saskarnes diagramma (sk. 3.7. att.) ir attēlota diagramma vienas sistēmas, vienas organizācijas un vienas komandas ietvaros. Un katrs skats ir sadalīts, ņemot vērā sistēmas

lietotāju, to attiecību ar attēloto organizāciju un attiecību ar organizācijas komandu. Lietotāju saskarnes pārvietošanas diagramma plānošanas un laika pārvaldības sistēmai satur sekojošu krāsu kodējumu:

- Neautenticēts lietotājs (pelēkā krāsa)
- Autenticēts lietotājs (gaiši zilā krāsa)
- Organizācijas īpašnieks (gaiši oranžā krāsa)
- Organizācijas dalībnieks (zaļā krāsa)
- Komandas izstrādes procesa vadītājs (sarkanā krāsa)
- Komandas dalībnieks (violetā krāsa)

Ja lietotājs neatbilst atzīmētas krāsas nosacījumam, tad lietotājam nav tiesību apmeklēt izvēlēto skatu.

Lietotāju saskarnes pārvietošanas diagramma ir sadalīta trīs daļās, izmantojot divas pārtrauktās līnijas. Tas tika izdarīts lai demonstrētu, ka lietotājam ir iespēja izlogoties no sistēmas visos skatos, kas ietilpst taisnstūrī ar zilu pārtrauktu līniju, un ir iespējams piekļūt pie paziņojuma saraksta skata no visiem sistēmas skatiem, kas ietilpst taisnstūrī ar sarkanu pārtrauktu līniju.

Neautenticēts lietotājs vispirms tiek piekļūts mājas lapai, kur neautenticētam lietotājam ir iespēja reģistrēties vai pieteikties sistēmai, izmantojot reģistrācijas vai pieteikšanās skatus.

3.3.2. Salkarnes skati

3.3.2.1. Organizācijas komandas pārvaldības skats

Organizācijas komandas pārvaldības skats
http://timeline-management.com/organization/1/team/3

Plānošanas un laika pārvaldības sistēma

1 Organizācijas 2 Komandas 3 Laika pārvaldība 4 Pārskats 5 Atteikties 6

Komandas informācija 7

Komandas nosaukums: Bug Busters
Komandas vadītājs: Igors Jurkovs
Komandas apraksts: Bug Busters komanda ir atbildīga par līgumu moduļa izstrādi kompanijas BBS Solutions produktā

Komandas aktīvas pazīmes 8

P21-S: Paralelu uzdevumu īstenošana (no 20.12.2021 līdz 24.12.2021) Aizvērt

Aktīvie uzdevumi 9

P21-S-2: Paralelu uzdevumu reģistrēšana	Izpildītājs: Roberts Gordins	Stāvoklis: Aktīvs	Aizvērt
P21-S-US-1: Lietotāja stāsta prasību atjaunošana	Izpildītājs: Sistēmas arhitektori	Stāvoklis: Izpildītāja piešķiršana	Aizvērt

Izveidot jaunu uzdevumu 10

Aizvērtie uzdevumi 11

P21-S-DB-1: Datu bāzes tabulu atjaunošana	Izpildītājs: Roberts Gordins	Stāvoklis: Aizvērts
P21-S-US-0: Lietotāja stāsta izveidošana	Izpildītājs: Igors Jurkovs	Stāvoklis: Aizvērts

Izveidot jaunu pazīmi 12

Komandas aizvērtas pazīmes 13

P20-S: Grupas uzdevuma izveidošana (no 04.12.2021 līdz 16.12.2021)

Aizvērtie uzdevumi

P20-S-DB-1: Datu bāzes tabulu atjaunošana	Izpildītājs: Roberts Gordins	Stāvoklis: Aizvērts
P20-S-US-1: Lietotāja stāsta izveidošana	Izpildītājs: Igors Jurkovs	Stāvoklis: Aizvērts
P20-S-C-1: Grupas uzdevuma izveidošanas skats	Izpildītājs: Toms Ešmils	Stāvoklis: Aizvērts

3.8. att. Organizācijas komandas pārvaldības skats

Organizācijas pārvaldības skatā (sk. 3.8. att.) tiek attēlota noteiktas organizācijas komandas pārvaldības iespējas. Minētais skats ir attēlots no izstrādes vadītāja puses, bet visiem norādītas komandas dalībniekiem ir tiesības piekļūt izvēlētam skatam. Izvēlētā skatā izstrādes vadītājam ir iespējams apskatīt izvēlētās komandas informāciju (komandas nosaukumu, vadītāju un aprakstu) kā arī apskatīt komandas izstrādes procesa informāciju (aktīvas un aizvērtas pazīmes, kur ir norādītas aktīvie un aizvērtie uzdevumi). Izstrādes vadītājs var aizvērt noteiktu pazīmi vai uzdevumu un komandas vadītājs var pieprasīt uzdevuma aizvēršanu. Skata attēlā (sk. 3.8. att.) ir ar sarkano apli un skaitli ir atzīmēti sekojošas skata daļas:

Navigācijas josla (sevī ietver skaitļus no 1. līdz 6. un navigācijas josla tiks aprakstīta tikai sekojošā skatā):

1. Navigācijas joslas poga *Organizācijas*. Poga, kas novirzīs autentificētu lietotāju līdz organizācijas saraksta skatam, kur tiks attēlotas visas organizācijas, kur autentificēts lietotājs ir organizācijas dalībnieks.
2. Navigācijas joslas poga *Komandas*. Poga, kas novirzīs autentificētu lietotāju līdz noteiktas organizācijas komandas saraksta skatam, kur tiks attēlotas visas komandas, kur autentificēts lietotājs ir komandas dalībnieks. Poga tiks paslēpta no navigācijas joslas, ja netiks izvēlēta organizācija. Poga *Komandas* ir izvēlēta, jo sekojošā skatā lietotājs mijiedarbojas ar noteiktas komandas pārvaldību.
3. Navigācijas joslas poga *Laika pārvaldība*. Poga, kas novirzīs autentificētu lietotāju līdz izvēlētās organizācijas komandas kalendāra ieraksta reģistrēšanas skatu (sk. 3.3.2.3. *Kalendāra ieraksta reģistrēšanas skats* sekciju). Poga tiks paslēpta no navigācijas joslas, ja netiks izvēlēta noteikta organizācijas komanda.
4. Navigācijas joslas poga *Pārskats*. Poga, kas novirzīs autentificētu lietotāju līdz izvēlētās organizācijas komandas izstrādes procesa attēlošanas skatu. Poga ir attēlota tikai tad, kad autentificētam lietotājam ir izvēlētās komandas izstrādes vadītāja tiesības (komandas vadītājs vai organizācijas īpašnieka tiesības). Poga arī ir paslēpta tad, kad noteikta organizācijas komanda nav attēlota.
5. Navigācijas joslas poga *Atteikties*. Poga, kas ļauj autentificētam lietotājam atteikties no sistēmas. Lietotājs tiks novirzīts uz mājas lapas skatu. I pieejama tikai autentificētam lietotājam.
6. Navigācijas joslas aploksnis attēla poga. Poga, kas novirzīs autentificētu lietotāju pie paziņojuma saraksta skata. Ir pieejama tikai autentificētam lietotājam.

Organizācijas komandas pārvaldības skata sastāvdaļas:

7. Komandas informācijas sekcija. Sekcija, kur tiek attēlots komandas nosaukums, komandas vadītāja vārds un uzvārds kā arī komandas apraksts. Izstrādes vadītājam nav iespējams rediģēt izvēlēto informāciju, tāpēc komandas pārvaldības skatā netiek izveidota pāreja un komandas informācijas rediģēšanas skatu.
8. Komandas aktīvu pazīmju sekcija. Sekcija, kur tiek attēlotas pazīmes, kuras atrodas izstrādes procesā. Visi komandas dalībnieki ir pieejams informācija par komandas pazīmēm, bet tika izstrādes vadītājam ir pieejama poga *Aizvērt* pie pazīmes nosaukuma. Nospiežot minēto pogu, pazīme tiek aizvērta un pārvietota sekcijā 13. Uzspiežot uz pazīmes nosaukumu, izstrādes vadītājs tiks pārvietots uz komandas pazīmes veidošanas/rediģēšanas skatu.
9. Aktīvo uzdevumu sekcija. Pazīmes sekcija, kur tiek attēloti visi izveidoti uzdevumi, kuriem nav stāvoklis *Aizvērts*. Izstrādes vadītājam ir iespējams aizvērt uzdevumu, uzspiežot uz pogu *Aizvērt*, kas atrodas pa labi no aktīva uzdevuma. Komandas dalībniekam ir iespējams pieprasīt aizvērt uzdevumu, ja komandas dalībnieks ir iestatīts kā izvēlēta uzdevuma izpildītājs. Uzspiežot uz uzdevuma nosaukumu, izstrādes vadītājs tiks pārvietots uz komandas uzdevuma veidošanas / rediģēšanas skatu (sk. 3.3.2.2. *Komandas uzdevuma veidošanas / rediģēšanas skats* sekciju).
10. Poga *Izveidot jaunu uzdevumu* ir pieejama tikai izstrādes vadītājam. Uzspiežot uz minēto pogu, tā novirzīs izstrādes vadītāju uz komandas uzdevuma veidošanas / rediģēšanas skatu (sk. 3.3.2.2. *Komandas uzdevuma veidošanas / rediģēšanas skats* sekciju).
11. Aizvērtu uzdevumu sekcija. Pazīmes sekcija, kur tiek attēloti visi izveidoti uzdevumi, kuriem ir stāvoklis *Aizvērts*. Aizvērtam uzdevumam joprojām ir iespējams rediģēt tā informāciju, uzspiežot uz tā nosaukumu (izstrādes vadītājs tiks novirzīts uz komandas uzdevuma veidošanas/rediģēšanas skatu).
12. Poga *Izveidot komandas pazīmi* ir pieejama tikai izstrādes vadītājam. Uzspiežot un minēto pogu, tā novirzīs izstrādes vadītāju uz komandas pazīmes veidošanas / rediģēšanas saktu, kur izstrādes vadītājam ir iespējams izveidot jaunu aktīvo pazīmi izvēlētai komandai.

13. Aizvērtu pazīmju sekcija, kur tiek attēlotas pazīmes, kurus izstrādes vadītājs aizvēra. Aizvērtas pazīmes un aizvērtie uzdevumi nevar tikt atvērti.

3.3.2.2. Komandas uzdevuma veidošanas/redīgēšanas skats

Komandas uzdevuma veidošanas/redīgēšanas skats
http://timeline-management.com/organization/1/team/3/task/create

Plānošanas un laika pārvaldības sistēma

Organizācijas **Komandas** Laika pārvaldība Pārskats Atteikties

Komandas uzdevuma izveide 2 Izveidot 3 Atcelt

Uzdevuma kods: 4 P21-S-2

Uzdevuma nosaukums: 5 Paralēlu uzdevumu reģistrēšana

Uzdevuma apraksts: 6 Uzsākot vizēšanas procesu, lietotājam ir iespējams izvēlēties vizēt izvēlēto dokumentu starp visiem vizētājiem paralēli.

Komandas uzdevuma atsauču izveide 7

Uzdevuma izpildītājs 8 Uzdevuma kods 9 Atsauces veids 10 12 Pievienot

Papildus informācija 11

Roberts Gordins P21-S-DB-1 13 X Ķēdes veidošanas atsauce

Nepieciešams modificēt konfigurācijas parametrus, lai pievienot iespēju iespējot paralēlu izpildīšanu

Igors Jurkovs P21-S-US-0 X Informatīva atsauce

Lietotāja stāsts, kur uzdevuma ipzilde ir aprakstīta

3.9. att. Komandas uzdevuma veidošanas/redīgēšanas skats

Komandas uzdevuma veidošanas/redīgēšanas skatā (sk. 3.9. att.) tiek attēlota noteiktas organizācijas komandas pazīmes uzdevuma izveidošana vai redīgēšana. Skats ir pieejams tikai izvēlētas komandas izstrādes vadītājam un pie tā var nokļūt uzspiežot pogu *Izveidot jaunu uzdevumu* (sk. 3.8. att. 10. skata sastāvdaļa).

Komandas uzdevuma veidošanas/rediģēšanas skata sastāvdaļas:

1. Navigācijas joslā ir aktīva sekcija *Komandas*, jo uzdevumu izveidošanas skats ir pieejams tad un tikai tad, ja ir izvēlēta noteikta komanda no komandu saraksta.
2. Poga *Izveidot* ir nepieciešama, lai uzsāktu formas validāciju. Ja validācijas vai apstrādes gaitā tika reģistrēta kļūda, tad kļūda tiks attēlota zem izvēlēta ievaddatu lauka (ja kļūda attiecas un noteiktu lauku) vai zem formas (ja kļūda attiecas uz vairākiem ievadlaukiem). Ja validācijas un apstrādes laikā netika reģistrēta kļūda, tad izstrādes vadītājs tiek novirzīts uz uzdevuma izpildītāju iestatīšanas skatu izveidotam uzdevumam. Pogas nosaukums mainās atkarībā no uzdevuma veidošanas vai rediģēšanas. Nosaukums *Izveidot* tiek attēlots uzdevuma veidošanas gaitā un nosaukums *Rediģēt* tiks attēlots uzdevuma rediģēšanas gaitā. Poga izsauc funkciju, kas ir aprakstīta 2.2.4.2. *Uzdevuma izveidošana/datu rediģēšana* sekcijā.
3. Poga *Atcelt* ir nepieciešama, lai atcelt uzdevuma rediģēšanu vai izveidošanu. Izstrādes vadītājs tiek novirzīts uz organizācijas komandas pārvaldības skatu (sk. 3.3.2.1. *Organizācijas komandas pārvaldības skats* sekciju).
4. Uzdevuma koda ievadlauks. Obligāts un unikāls izvēlētas komandas ietvaros. Ja ievadlauks satur validācijas kļūdas, tad tās tiks attēlotas zem teksta ievadlauka.
5. Uzdevuma nosaukumu ievadlauks. Obligāts lauks. Ja ievadlauks satur validācijas kļūdas, tad tās tiks attēlotas zem teksta ievadlauka.
6. Uzdevuma apraksts. Neobligāts lauks.
7. Komandas uzdevuma atsauču izveides sekcija. Sekcija, kur ir iespējams izveidot atsauces uz eksistējošiem pazīmes uzdevumiem.
8. Uzdevuma izpildītāja iezīmes lauks, kur tiks attēlots izvēlēta uzdevuma izpildītāja vārds un uzvārds. Lauka informāciju nav iespējams rediģēt un tā tiek atjaunota, kad tiek izvēlēts cits uzdevumu kods (9. skata sastāvdaļa).
9. Uzdevuma koda izvēlne. Izstrādes vadītājam tiek piedāvāta izvēlne ar izvēlētas pazīmes uzdevumiem (aktīvie un aizvērtie). Kad izstrādes vadītājs izvēlas noteiktu uzdevumu, tad uzdevumu izpildītāja laukā tiek atjaunots uzdevuma izpildītāja vārda un uzvārda vērtības, kas atbilst izvēlēta uzdevuma izpildītājam.

Ja uzdevuma izpildītājs ir loma, tad tiek attēlots lomas nosaukums. Ja netika izvēlēts uzdevuma izpildītājs, tad tiek attēlots tukšums.

10. Uzdevuma atsaucē veida izvēlne. Izstrādes vadītājam tiek piedāvāta izvēlne ar sistēmas uzdevuma atsaucē veidiem.
11. Papildus informācijas ievadlauks. Nav obligāts. Ievadlauka nolūks ir norādīt nepieciešamo informāciju, lai paskaidrotu atsaucē izveidošanas iemeslu.
12. Poga *Pievienot* ir nepieciešama, lai pievienotu uzdevumam izveidoto atsaucē. Nospiežot pogu, tiek veikta jaunas atsaucē validācija un ja validācijas gaitā tiek reģistrēta kļūda, tad kļūdas paziņojums tiks attēlots zem atsaucē veidošanas formas (komponentes). Ja validācijas gaitā netika reģistrēta kļūda, tad tiek izveidots jauns atsaucē objekts, kas tiks novietots zem veidošanas formas (13. skata sastāvdaļa).
13. Saraksts ar izveidotām atsaucē objektiem. Katrai atsaucē tiek pievienots krusta attēls, ar kuru palīdzību ir iespējams izdzēst izveidoto atsaucē objektu. Izveidotam atsaucē objektam joprojām ir iespējams veidot izmaiņas.

3.3.2.3. Kalendāra ieraksta reģistrēšanas skats

Kalendāra ieraksta reģistrēšanas skats
http://timeline-management.com/organization/1/calendar-entries

Plānošanas un laika pārvaldības sistēma

Organizācijas Komandas **Laika pārvaldība** Atteikties

Komandas uzdevumu laika pārvaldība

Ierp. Nāk. Šodien 2021. g. 20 – 26. dec. Mēnesis **Nedēļa** Diena Dienas kārtība

Komandas pazīmes:

- P21-S: Paralelu uzdevumu īstenošana (no 20.12.2021 līdz 24.12.2021)
- Aktīvie uzdevumi**
 - P21-S-2: Paralelu uzdevumu reģistrēšana
 - P21-S-US-2: Lietotāja stāsta prasību validēšana

Ned. 51.	pirmd. 20.12.	otrd. 21.12.	trešd. 22.12.	ceturd. 23.12.	piektd. 24.12.	sestd. 25.12.	svētd. 26.12.
08							
09							
10	P20-S-DB-1: Datu bāzes tabulu atjaunošana			P21-S-US-2: Lietotāja stāsta prasību validēšana			
11		P21-S-2: Paralelu uzdevumu reģistrēšana					
12							
13							
14	P20-S-DB-1: Datu bāzes tabulu atjaunošana			P21-S-2: Paralelu uzdevumu reģistrēšana			
15							
16							
17		P21-S-2: Paralelu uzdevumu reģistrēšana					
18							
19							
20							

3.10. att. Kalendāra ieraksta reģistrēšanas skats

Kalendāra ieraksta reģistrēšanas skatā (sk. 3.10. att.) tiek attēlota komandas dalībnieka laika pārvaldības kalendāra ierakstu reģistrēšana. Skats ir attēlots no komandas dalībnieka puses, kuram izvēlēta komandā nav izstrādes vadītāja tiesības.

Kalendāra ieraksta reģistrēšanas skata sastāvdaļas:

1. Navigācijas joslā ir aktīva sekcija *Laika pārvaldība*, jo autentificēts lietotājs atrodas laika pārvaldības skatā, kur tiks reģistrēti kalendāra ieraksti aktīviem uzdevumiem.
2. Aktīvo pazīmju attēlošana, kur lietotājam ir aktīvie uzdevumi. Līdzīgi kā organizācijas komandas pārvaldības skatā, uzdevumi tiek iekļauti noteiktajā pazīmē, tomēr tiek attēlotas tikai tādas pazīmes, kur autentificētam lietotājam ir kaut viens aktīvais uzdevums.
3. Aktīvie uzdevumi. Iezīmes ar vilkšanas un nomešanas īpašībām (drag&drop). Lai izveidotu kalendāra ierakstu, nepieciešams izvēlēties uzdevuma iezīmi un pārvietot to uz laika pārvaldības kalendāru. Ar katru pārvietošanu tiek izveidots funkcija 2.2.4.5. *Kalendāra ieraksta izveidošana/datu rediģēšana* izsaukums.
4. Uzdevuma kalendāra ieraksts. Kalendāra ieraksts noteikti aizņem vismaz vienu stundu kalendārā. Autentificētam lietotājam ir iespējams rediģēt tikai aktīva uzdevuma kalendāra ierakstus. Ja uzdevums tika aizvērts vai uzdevums ir pārskatā, tad pēc kalendāra rediģēšanas tiks izvadīts pārlūka noklusētais modālais logs ar kļūdas paziņojumu.

3.3.2.4. Izstrādes procesa attēlošanas skats

Izstrādes procesa attēlošanas skats
http://timeline-management.com/organization/1/overview

Plānošanas un laika pārvaldības sistēma

Organizācijas Komandas Laika pārvaldība **Pārskats** Atteikties

Organizācijas izstrādes procesa pārskats

Komandas nosaukums: **2** Bug Busers
Lomas nosaukums: **3** Programmētāji
Komandas dalībnieks: **4** Roberts Gordins
Brajans Grants
Toms Ešmils

Pārskatīt **5**

Ierp. Nāk. Šodien 2021. g. 20 – 26. dec. Mēnesis Nedēļa Diena Dienas kārtība

Ned. 51.	pirmd. 20.12.	otrd. 21.12.	trešd. 22.12.	ceturd. 23.12.	piektd. 24.12.	sestd. 25.12.	svētd. 26.12.
08							
09	P20-S-DB-1: Datu bāzes tabulu atjaunošana 6	P21-S-2: Paralēlu uzdevumu reģistrēšana		P21-S-US-2: Lietotāja stāsta prasību validēšana			
10							
11							
12							
13							
14	P20-S-DB-1: Datu bāzes tabulu atjaunošana	P21-S-2: Paralēlu uzdevumu reģistrēšana	P21-S-2: Paralēlu uzdevumu reģistrēšana				
15							
16							
17	P21-S-2: Paralēlu uzdevumu reģistrēšana						
18							
19							
20							

3.11. att. Izstrādes procesa attēlošanas skats

Izstrādes procesa attēlošanas skatā (sk. 3.11. att.) tiek attēlotas izvēlētās organizācijas izstrādes procesa kalendāra ieraksti. Skats ir pieejams tikai autentificētam lietotājam ar izstrādes procesa vadītāja tiesībām (autentificētam lietotājam ir nepieciešams būt vismaz vienas komandas vadītājam) vai organizācijas īpašnieka tiesībām.

Izstrādes procesa attēlošanas skata sastāvdaļas:

1. Navigācijas joslā ir aktīva sekcija *Pārskats*, jo autentificēts lietotājs atrodas izstrādes procesa attēlošanas skatā, kur tiks veikts organizācijas dalībnieku laika pārvaldības pārskats.
2. Organizācijas komandas izvēlne. Autentificētam lietotājam tiek piedāvāta izvēlne ar organizācijas komandām, kur lietotājam ir izstrādes vadītāja tiesības.
3. Komandas lomas izvēlne. Autentificētam lietotājam tiek piedāvāta izvēlne ar organizācijas lomām, kas tika piešķirtas izvēlētai komandai (sk. 2. skata sastāvdaļu).

4. Organizācijas dalībnieku izvēlne. Autenticētam lietotājam tiek piedāvāta izvēlne ar organizācijas dalībniekiem. Ja lietotājs izvēlējas organizācijas komandu (sk. 2. skata sastāvdaļu), tad izvēlnē tiek attēloti tikai komandas dalībnieki. Ja lietotājs izvēlējas organizācijas lomu (sk. 3. skata sastāvdaļu), tad izvēlnē tiek attēloti komandas dalībnieki, kuriem ir piešķirta attiecīga loma.
5. Poga *Pārskatīt* ir nepieciešama, lai skata kalendārā ielādētu pieprasītos kalendāra ierakstus. Atkarībā no aizpildītiem ievadlaukiem (sk. 2.-4. skata sastāvdaļas) tiek kalendāra ierakstu ielādēšanai tiek izveidoti attiecīgu funkciju izsaukumi:
 - i. Ja tika izvēlēts organizācijas dalībnieks, tad tiek izveidots funkcijas *2.2.6.1. Komandas dalībnieka laika pārvaldības apskate* izsaukums
 - ii. Ja tika izvēlēta komandas loma, tad tiek izveidots funkcijas *2.2.6.2. Komandas lomas laika pārvaldības apskate* izsaukums.
 - iii. Ja tika izvēlēta organizācijas komanda, tad tiek izveidots funkcijas *2.2.6.3. Komandas laika pārvaldības apskate* izsaukums.
6. Uzdevuma kalendāra ieraksts. Kalendāra ieraksts noteikti aizņem vismaz vienu stundu kalendārā. Autenticētam lietotājam nav iespējams rediģēt kalendāra ierakstus.

4. TESTĒŠANAS DOKUMENTĀCIJA

4.1. Testēšanas apraksts

Testēšanas dokumentācijā ir aprakstītas plānošanas un laika pārvaldības sistēmas vienībtestēšanas process, lai pārliecinātos, ka izstrādātā programmatūra atbilst izveidotai programmatūras prasību specifikācijai. Plānošanas un laika pārvaldības sistēma ir sadalīta divās daļās: tīmekļa vietne un lietojumprogrammas saskarne. Tīmekļa vietnes funkcionalitāte tika testēta manuāli, bet lietojumprogrammas saskarnes testēšanā tika izmantots vienībtestēšanas satvars xUnit.

Katrs testpiemērs ir sadalīts trīs daļās pēc AAA vienībtestēšanas metodes (Sakārtošana, Darbība, Apgalvošana), kur katrā daļā tiek veiktas attiecīgas darbības. Sakārtošanas daļā tiek sagatavoti dati, kas tiks izmantoti testpiemērā. Darbības daļā testējamā funkcija veic darbību. Apgalvošanas daļā funkcijas rezultāts tiek pārbaudīts un tiek reģistrēts testa rezultāts.

Datu bāzes simulēšanai lietojumprogrammas saskarnes vienībtestēšanā tika izmantota *EntityFramework Core In-Memory Database Provider*, kas izmanto testēšanas ierīces atmiņu, kur tiks saglabāti dati testēšanas nolūkā.

4.2 Testpiemēru saraksts

Testpiemēru sarakstā tiek attēloti daži funkcijas vienībtestēšanas scenāriji. Līdzīgi testi tiek palaisti ar citu ievadlauku vērtībām.

Tabulā 4.1. ir aprakstīts funkcijas 2.2.3.1. *Lietotāju reģistrācija* vienībtestēšana.

4.1. tabula

Funkcijas 2.2.3.1. Lietotāju reģistrācija vienībtestēšana

<i>Testpiemēra datu sagatave</i>	<i>Ievaddati</i>	<i>Sagaidāmais rezultāts</i>
-	<i>Lietotāja vārds: Roberts Lietotāja otrais vārds: - Lietotāja uzvārds: Gordins Lietotājvārds: Conwyn E-pasts: rgordins@gmail.com Lietotāju parole: Par0le!</i>	<i>Lietotājs veiksmīgi tika reģistrēts sistēmā. Atbildes kods: 200 (Ok) Tiek atgriezts: Id: 1 (identifikators, kas atbilsts datu bāzes ierakstam)</i>

		<p><i>Lietotāja vārds: Roberts</i></p> <p><i>Lietotāja otrais vārds: -</i></p> <p><i>Lietotāja uzvārds: Gordins</i></p> <p><i>Lietotājvārds: Conwyn</i></p> <p><i>E-pasts:</i></p> <p><i>rgordins@gmail.com</i></p>
<p><i>Sistēma ir izveidots lietotājs ar lietotājvārdu: Conwyn</i></p>	<p><i>Lietotāja vārds: Roberts</i></p> <p><i>Lietotāja otrais vārds: -</i></p> <p><i>Lietotāja uzvārds: Gordins</i></p> <p><i>Lietotājvārds: Conwyn</i></p> <p><i>E-pasts: rgordins@gmail.com</i></p> <p><i>Lietotāju parole: Par0le!</i></p>	<p><i>Atbildes kods: 409 (Validation Error)</i></p> <p><i>Kļūdu paziņojums: Lietotājs ar norādītu lietotājvārdu jau pastāv sistēmas ietvaros.</i></p>
<p><i>Sistēma ir izveidots lietotājs ar e-pastu: rgordins@gmail.com</i></p>	<p><i>Lietotāja vārds: Roberts</i></p> <p><i>Lietotāja otrais vārds: -</i></p> <p><i>Lietotāja uzvārds: Gordins</i></p> <p><i>Lietotājvārds: Conwyn</i></p> <p><i>E-pasts: rgordins@gmail.com</i></p> <p><i>Lietotāju parole: Par0le!</i></p>	<p><i>Atbildes kods: 409 (Validation Error)</i></p> <p><i>Kļūdu paziņojums: Lietotājs ar norādītu e-pastu jau pastāv sistēmas ietvaros</i></p>
-	<p><i>Lietotāja vārds: Roberts</i></p> <p><i>Lietotāja otrais vārds: -</i></p> <p><i>Lietotāja uzvārds: Gordins</i></p> <p><i>Lietotājvārds: Conwyn</i></p> <p><i>E-pasts: rgordins@gmail.com</i></p> <p><i>Lietotāju parole:test</i></p>	<p><i>Atbildes kods: 409 (Validation Error)</i></p> <p><i>Kļūdu paziņojums: Ievadīta parole neatbilst sistēmas prasībām. Paroles simbolu virknē ir nepieciešams ietvert vismaz vienu mazu burtu, vienu lielu burtu, vienu ciparu un vismaz vienu speciālu simbolu</i></p>

Tabulā 4.2. ir aprakstīts funkcijas 2.2.3.2. *Lietotāju autentifikācija* vienībtestēšana.

4.2. tabula

Funkcijas 2.2.3.2. Lietotāju autentifikācija vienībtestēšana

<i>Testpiemēra datu sagatave</i>	<i>Ievaddati</i>	<i>Sagaidāmais rezultāts</i>
-	<p><i>Lietotāja pieteikuma vārds:</i> Conwyn</p> <p><i>Lietotāja parole: ParOle!</i></p>	<p><i>Atbildes kods: 409</i> <i>(Validation Error)</i></p> <p><i>Kļūdu paziņojums: Tika ievadīts nepareizs pieteikuma vārds vai parole.</i></p>
<p><i>Sistēma pastāv lietotājs:</i> <i>Id: 1</i> <i>Lietotāja vārds: Roberts</i> <i>Lietotāja otrais vārds: -</i> <i>Lietotāja uzvārds:</i> <i>Gordins</i> <i>Lietotājvārds: Conwyn</i> <i>E-pasts:</i> <i>rgordins@gmail.com</i> <i>Lietotāju parole: ParOle!</i></p>	<p><i>Lietotāja pieteikuma vārds:</i> Conwyn</p> <p><i>Lietotāja parole: OtraParOle!</i></p>	<p><i>Atbildes kods: 409</i> <i>(Validation Error)</i></p> <p><i>Kļūdu paziņojums: Tika ievadīts nepareizs pieteikuma vārds vai parole.</i></p>
<p><i>Sistēma pastāv lietotājs:</i> <i>Id: 1</i> <i>Lietotāja vārds: Roberts</i> <i>Lietotāja otrais vārds: -</i> <i>Lietotāja uzvārds:</i> <i>Gordins</i> <i>Lietotājvārds: Conwyn</i> <i>E-pasts:</i> <i>rgordins@gmail.com</i> <i>Lietotāju parole: ParOle!</i></p>	<p><i>Lietotāja pieteikuma vārds:</i> Conwyn</p> <p><i>Lietotāja parole: ParOle!</i></p>	<p><i>Lietotājs veiksmīgi pieteicās sistēmā.</i></p> <p><i>Atbildes kods: 200 (Ok)</i></p> <p><i>Tiek atgriezts:</i> <i>Id: 1</i> <i>Lietotāja vārds: Roberts</i> <i>Lietotāja otrais vārds: -</i> <i>Lietotāja uzvārds: Gordins</i> <i>Lietotājvārds: Conwyn</i> <i>E-pasts:</i> <i>rgordins@gmail.com</i></p>

Tabulā 4.3. ir aprakstīts funkcijas 2.2.3.3 *Organizācijas izveidošana / datu rediģēšana* vienībtestēšana.

4.3. tabula

Funkcijas 2.2.3.3 Organizācijas izveidošana / datu rediģēšana vienībtestēšana

<i>Testpiemēra datu sagatave</i>	<i>Ievaddati</i>	<i>Sagaidāmais rezultāts</i>
-	<p><i>Organizācijas nosaukums: Testa organizācija</i></p> <p><i>Organizācijas apraksts: -</i></p> <p><i>Lietotāja identifikators: 1</i></p>	<p><i>Tiek izveidota jauna organizācija.</i></p> <p><i>Atbildes kods: 200 (Ok)</i></p> <p><i>Tiek atgriezts:</i></p> <p><i>Id: 1 (identifikators, kas atbilsts izveidotam datu bāzes ierakstam)</i></p> <p><i>Organizācijas nosaukums: Testa organizācija</i></p> <p><i>Organizācijas apraksts:-</i></p> <p><i>Organizācijas īpašnieka identifikators: 1</i></p>
<p><i>Sistēmā pastāv organizācija ar nosaukumu: Testa organizācija</i></p>	<p><i>Organizācijas nosaukums: Testa organizācija</i></p> <p><i>Organizācijas apraksts: -</i></p> <p><i>Lietotāja identifikators: 1</i></p>	<p><i>Atbildes kods: 409 (Validation Error)</i></p> <p><i>Kļūdu paziņojums:</i></p> <p><i>Organizācija ar norādīto organizācijas nosaukumu pastāv sistēmā.</i></p>
<p><i>Sistēmā pastāv organizācija ar:</i></p> <p><i>Organizācijas nosaukums: Testa organizācija</i></p> <p><i>Organizācijas īpašnieka identifikators: 1</i></p> <p><i>Id: 1</i></p>	<p><i>Organizācijas nosaukums: Jauna testa organizācija</i></p> <p><i>Organizācijas apraksts: Jauns apraksts</i></p> <p><i>Lietotāja identifikators: 1</i></p> <p><i>Organizācijas identifikators: 1</i></p>	<p><i>Organizācijas dati tika rediģēti.</i></p> <p><i>Atbildes kods: 200 (Ok)</i></p> <p><i>Tiek atgriezts:</i></p> <p><i>Id: 1</i></p> <p><i>Organizācijas nosaukums: Jauna testa organizācija</i></p> <p><i>Organizācijas apraksts: Jauns apraksts</i></p>

		<i>Organizācijas īpašnieka identifikators: 1</i>
<i>Sistēmā pastāv organizācija ar: Organizācijas nosaukums: Testa organizācija Organizācijas īpašnieka identifikators: 1 Id: 1</i>	<i>Organizācijas nosaukums: Jauna testa organizācija Organizācijas apraksts: Jauns apraksts Lietotāja identifikators: 2 Organizācijas identifikators: 1</i>	<i>Atbildes kods: 409 (Validation Error) Kļūdu paziņojums: Nav tiesību rediģēt organizācijas Testa organizāciju datus</i>

Tabulā 4.4. ir aprakstīts funkcijas 2.2.3.4. Lomas izveidošana / datu rediģēšana vienībtestēšana.

4.4. tabula

Funkcijas 2.2.3.4. Lomas izveidošana / datu rediģēšana vienībtestēšana

<i>Testpiemēra datu sagatave</i>	<i>Ievaddati</i>	<i>Sagaidāmais rezultāts</i>
<i>Sistēma ir izveidota organizācija ar identifikatoru 1. Organizācijas īpašnieka identifikators: 1</i>	<i>Lomas nosaukums: Testa loma Lomas apraksts: - Lomas krāsa: ff0000 Organizācijas identifikators: 1 Lietotāja identifikators: 1</i>	<i>Tiek izveidota jauna loma. Atbildes kods: 200 (Ok) Tiek atgriezts: Id: 1 (identifikators, kas atbilsts izveidotam datu bāzes ierakstam) Lomas nosaukums: Testa loma Lomas apraksts: - Lomas krāsa: ff0000 Organizācijas identifikators: 1</i>
<i>Sistēma ir izveidota organizācijas ar identifikatoru 1, 2. Organizāciju īpašnieka identifikators: 1</i>	<i>Lomas nosaukums: Testa loma Lomas apraksts: - Lomas krāsa: ff0000 Organizācijas identifikators: 1 Lietotāja identifikators: 1</i>	<i>Tiek izveidota jauna loma. Atbildes kods: 200 (Ok) Tiek atgriezts:</i>

<p><i>Organizācijā ar identifikatoru 2 pastāv sekojošā loma:</i></p> <p><i>Id: 1</i></p> <p><i>Lomas nosaukums: Testa loma</i></p> <p><i>Lomas apraksts: -</i></p> <p><i>Lomas krāsa: ff0000</i></p> <p><i>Organizācijas identifikators: 2</i></p>		<p><i>Id: 2 (identifikators, kas atbilsts izveidotam datu bāzes ierakstam)</i></p> <p><i>Lomas nosaukums: Testa loma</i></p> <p><i>Lomas apraksts: -</i></p> <p><i>Lomas krāsa: ff0000</i></p> <p><i>Organizācijas identifikators: 1</i></p>
<p><i>Sistēma ir izveidota organizācijas ar identifikatoru 1.</i></p> <p><i>Organizācijas īpašnieka identifikators: 1</i></p> <p><i>Organizācijā pastāv sekojošā loma:</i></p> <p><i>Id: 1</i></p> <p><i>Lomas nosaukums: Testa loma</i></p> <p><i>Lomas apraksts: -</i></p> <p><i>Lomas krāsa: ff0000</i></p> <p><i>Organizācijas identifikators: 1</i></p>	<p><i>Lomas nosaukums: Testa loma</i></p> <p><i>Lomas apraksts: -</i></p> <p><i>Lomas krāsa: ff0000</i></p> <p><i>Organizācijas identifikators: 1</i></p>	<p><i>Atbildes kods: 409 (Validation Error)</i></p> <p><i>Kļūdu paziņojums: Loma ar norādīto lomas nosaukumu eksistē organizācijas ietvaros</i></p>
<p><i>Sistēma ir izveidota organizācijas ar identifikatoru 1.</i></p> <p><i>Organizācijas īpašnieka identifikators: 1</i></p> <p><i>Organizācijā pastāv sekojošā loma:</i></p> <p><i>Id: 1</i></p> <p><i>Lomas nosaukums: Testa loma</i></p>	<p><i>Lomas identifikators: 1</i></p> <p><i>Lomas nosaukums: Jauna testa loma</i></p> <p><i>Lomas apraksts: Jauns apraksts</i></p> <p><i>Lomas krāsa: ff00ff</i></p> <p><i>Organizācijas identifikators: 1</i></p>	<p><i>Lomas dati tika rediģēti.</i></p> <p><i>Atbildes kods: 200 (Ok)</i></p> <p><i>Tiek atgriezts:</i></p> <p><i>Id: 1</i></p> <p><i>Lomas nosaukums: Jauna testa loma</i></p> <p><i>Lomas apraksts: Jauns apraksts</i></p> <p><i>Lomas krāsa: ff00ff</i></p> <p><i>Organizācijas identifikators: 1</i></p>

<i>Lomas apraksts: -</i> <i>Lomas krāsa: ff0000</i> <i>Organizācijas</i> <i>identifikators: 1</i>		
--	--	--

Tabulā 4.5. ir aprakstīts funkcijas 2.2.3.5. *Komandas izveidošana / datu rediģēšana vienībtestēšana.*

4.5. tabula

Funkcijas 2.2.3.5. Komandas izveidošana / datu rediģēšana vienībtestēšana

<i>Testpiemēra datu sagatave</i>	<i>Ievaddati</i>	<i>Sagaidāmais rezultāts</i>
<i>Sistēmā pastāv organizācija:</i> <i>Organizācijas identifikators: 1</i> <i>Organizācijas īpašnieka identifikators: 1</i> <i>Organizācijas pastāv lomas ar identifikatoriem: 1, 2, 3</i> <i>Organizācijā pastāv dalībnieki ar identifikatoriem: 1, 2, 3</i>	<i>Komandas nosaukums: Testa komanda</i> <i>Komandas apraksts: -</i> <i>Komandas iesaistītās lomas: [1, 2]</i> <i>Komandas vadītāja identifikators: 2</i> <i>Organizācijas identifikators: 1</i> <i>Lietotāja identifikators: 1</i>	<i>Tiek izveidota jauna komanda.</i> <i>Atbildes kods: 200 (Ok)</i> <i>Tiek atgriezts:</i> <i>Id: 1 (identifikators, kas tika piešķirts datu bāzes ierakstam)</i> <i>Komandas nosaukums: Testa komanda</i> <i>Komandas apraksts: -</i> <i>Komandas iesaistītās lomas: [1, 2]</i> <i>Komandas vadītāja identifikators: 2</i> <i>Organizācijas identifikators: 1</i>
<i>Sistēmā pastāv organizācija:</i> <i>Organizācijas identifikators: 1</i> <i>Organizācijas īpašnieka identifikators: 1</i> <i>Organizācijas pastāv lomas ar identifikatoriem: 1, 2, 3</i>	<i>Komandas nosaukums: Testa komanda</i> <i>Komandas apraksts: -</i> <i>Komandas iesaistītās lomas: [1, 2]</i> <i>Komandas vadītāja identifikators: 2</i> <i>Organizācijas identifikators: 1</i> <i>Lietotāja identifikators: 1</i>	<i>Atbildes kods: 409 (Validation Error)</i> <i>Kļūdu paziņojums: Komanda ar norādīto komandas nosaukumu eksistē organizācijas ietvaros</i>

<p><i>Organizācijā pastāv dalībnieki ar identifikatoriem: 1, 2, 3</i></p> <p><i>Organizācijā pastāv komanda ar nosaukumu: Testa komanda</i></p>		
<p><i>Sistēmā pastāv organizācija: Organizācijas identifikators: 1 Organizācijas īpašnieka identifikators: 1 Organizācijas pastāv lomas ar identifikatoriem: 1, 2, 3 Organizācijā pastāv dalībnieki ar identifikatoriem: 1, 2, 3</i></p> <p><i>Citā organizācijā pastāv komanda ar nosaukumu: Testa komanda</i></p>	<p><i>Komandas nosaukums: Testa komanda</i></p> <p><i>Komandas apraksts: -</i></p> <p><i>Komandas iesaistītās lomas: [1, 2]</i></p> <p><i>Komandas vadītāja identifikators: 2</i></p> <p><i>Organizācijas identifikators: 1</i></p> <p><i>Lietotāja identifikators: 1</i></p>	<p><i>Tiek izveidota jauna komanda. Atbildes kods: 200 (Ok)</i></p> <p><i>Tiek atgriezts: Id: 1 (identifikators, kas tika piešķirts datu bāzes ierakstam)</i></p> <p><i>Komandas nosaukums: Testa komanda</i></p> <p><i>Komandas apraksts: -</i></p> <p><i>Komandas iesaistītās lomas: [1, 2]</i></p> <p><i>Komandas vadītāja identifikators: 2</i></p> <p><i>Organizācijas identifikators: 1</i></p>
<p><i>Sistēmā pastāv organizācija: Organizācijas identifikators: 1 Organizācijas īpašnieka identifikators: 1 Organizācijas pastāv lomas ar identifikatoriem: 1, 2, 3 Organizācijā pastāv dalībnieki ar</i></p>	<p><i>Komandas nosaukums: Jauna testa komanda</i></p> <p><i>Komandas apraksts: Jauns apraksts</i></p> <p><i>Komandas iesaistītās lomas: [1]</i></p> <p><i>Komandas vadītāja identifikators: 2</i></p> <p><i>Organizācijas identifikators: 1</i></p> <p><i>Lietotāja identifikators: 1</i></p> <p><i>Komandas identifikators: 1</i></p>	<p><i>Komandas dati tiek veiksmīgi rediģēti. Atbildes kods: 200 (Ok)</i></p> <p><i>Tiek atgriezts: Id: 1</i></p> <p><i>Komandas nosaukums: Jauna testa komanda</i></p> <p><i>Komandas apraksts: Jauns apraksts</i></p> <p><i>Komandas iesaistītās lomas: [1]</i></p>

<p><i>identifikatoriem: 1, 2, 3</i></p> <p><i>Organizācijā pastāv komanda:</i></p> <p><i>Komandas nosaukums: Testa komanda</i></p> <p><i>Komandas apraksts: - Komandas iesaistītās lomas: [1, 2]</i></p> <p><i>Komandas vadītāja identifikators: 2</i></p> <p><i>Organizācijas identifikators: 1</i></p>		<p><i>Komandas vadītāja identifikators: 2</i></p> <p><i>Organizācijas identifikators: 1</i></p>
--	--	---

Tabulā 4.6. ir aprakstīts funkcijas 2.2.4.1. *Pazīmes izveidošana / datu rediģēšana vienībtestēšana.*

4.6. tabula

Funkcijas 2.2.4.1. Pazīmes izveidošana / datu rediģēšana vienībtestēšana

<i>Testpiemēra datu sagatave</i>	<i>Ievaddati</i>	<i>Sagaidāmais rezultāts</i>
<p><i>Sistēmas pastāv organizācija ar organizācijas identifikatoru 1 un tajā ir izveidota komanda ar komandas identifikatoru 1</i></p>	<p><i>Pazīmes kods: VT01</i></p> <p><i>Pazīmes nosaukums: Vienībtestēšana</i></p> <p><i>Pazīmes apraksts: - Pazīmes izstrādes sākuma datums: 12/12/2021</i></p> <p><i>Pazīmes izstrādes ilgums darbadienās: 5</i></p> <p><i>Komandas identifikators: 1</i></p> <p><i>Organizācijas identifikators: 1</i></p>	<p><i>Jauna pazīme ir izveidota sistēmas ietvaros.</i></p> <p><i>Atbildes kods: 200 (Ok)</i></p> <p><i>Tiek atgriezts:</i></p> <p><i>Id: 1 (Pazīmes ieraksta identifikators datu bāzē)</i></p> <p><i>Pazīmes kods: VT01</i></p> <p><i>Pazīmes nosaukums: Vienībtestēšana</i></p> <p><i>Pazīmes apraksts: - Pazīmes izstrādes sākuma datums: 12/12/2021</i></p>

		<p><i>Pazīmes izstrādes ilgums darbadienās: 5</i></p> <p><i>Komandas identifikators: 1</i></p> <p><i>Organizācijas identifikators: 1</i></p>
<p><i>Sistēmas pastāv organizācija ar organizācijas identifikatoru 1 un tajā ir izveidota komanda ar komandas identifikatoru 1. Sistēmā komandas ietvaros pastāv pazīme ar pazīmes kodu: VT01</i></p>	<p><i>Pazīmes kods: VT01</i></p> <p><i>Pazīmes nosaukums: Vienībtestēšana</i></p> <p><i>Pazīmes apraksts: -</i></p> <p><i>Pazīmes izstrādes sākuma datums: 12/12/2021</i></p> <p><i>Pazīmes izstrādes ilgums darbadienās: 5</i></p> <p><i>Komandas identifikators: 1</i></p> <p><i>Organizācijas identifikators: 1</i></p>	<p><i>Atbildes kods: 409 (Validation Error)</i></p> <p><i>Kļūdu paziņojums: Pazīme ar norādīto pazīmes kodu eksistē izvēlētas komandas izstrādes procesa ietvaros</i></p>
<p><i>Sistēmas pastāv organizācija ar organizācijas identifikatoru 1 un tajā ir izveidota komanda ar komandas identifikatoru 1. Sistēmā komandas ietvaros pastāv pazīme ar pazīmes identifikatoru 1</i></p>	<p><i>Id: 1</i></p> <p><i>Pazīmes kods: VT01</i></p> <p><i>Pazīmes nosaukums: Atkārtota vienībtestēšana</i></p> <p><i>Pazīmes apraksts: jauns apraksts</i></p> <p><i>Pazīmes izstrādes sākuma datums: 12/12/2021</i></p> <p><i>Pazīmes izstrādes ilgums darbadienās: 5</i></p> <p><i>Komandas identifikators: 1</i></p> <p><i>Organizācijas identifikators: 1</i></p>	<p><i>Pazīmes dati tika rediģēti.</i></p> <p><i>Atbildes kods: 200 (Ok)</i></p> <p><i>Tiek atgriezts:</i></p> <p><i>Id: 1 (Pazīmes ieraksta identifikators datu bāzē)</i></p> <p><i>Pazīmes kods: VT01</i></p> <p><i>Pazīmes nosaukums: Atkārtota vienībtestēšana</i></p> <p><i>Pazīmes apraksts: jauns apraksts</i></p> <p><i>Pazīmes izstrādes sākuma datums: 12/12/2021</i></p> <p><i>Pazīmes izstrādes ilgums darbadienās: 5</i></p> <p><i>Komandas identifikators: 1</i></p> <p><i>Organizācijas identifikators: 1</i></p>

Tabulā 4.7. ir aprakstīts funkcijas 2.2.4.2. *Uzdevuma izveidošana / datu rediģēšana vienībtestēšana*. Testpiemēru datu sagatavē tiek pieņemts kā pastāv organizācijas komanda, kurai tika piešķirti organizācijas dalībnieki un organizācijas lomas. Komandas ietvaros tika izveidota pazīme. *Informatīva atsauce* datu bāzē satur identifikatoru 1 un *Ķēdes veidošanas atsauce* datu bāzē satur identifikatoru 2.

4.7. tabula

Funkcijas 2.2.4.2. Uzdevuma izveidošana / datu rediģēšana vienībtestēšana

<i>Testpiemēra datu sagatave</i>	<i>Ievaddati</i>	<i>Sagaidāmais rezultāts</i>
<p><i>Sistēmā pastāv: Organizācija ar organizācijas identifikatoru: 1 Organizācijā pastāv komanda ar komandas identifikatoru 1. Komandā ir izveidota pazīme ar pazīmes identifikatoru 1.</i></p>	<p><i>Uzdevuma kods: TS-01 Uzdevuma nosaukums: Testa uzdevums Organizācijas identifikators: 1 Komandas identifikators: 1 Pazīmes identifikators: 1</i></p>	<p><i>Uzdevuma stāvoklis ir Izpildītāja piešķiršana Atbildes kods: 200 (Ok) Tiek atgriezts: Id: 2 (uzdevuma ieraksts, kas tika izveidots datu bāzē) Pazīmes identifikators: 1 Uzdevuma kods: TS-01 Uzdevuma statusa identifikators: 1 (Izpildītāja piešķiršana) Uzdevuma nosaukums: Testa uzdevums Uzdevuma apraksts: 0 Izpildīts: false</i></p>
<p><i>Sistēmā pastāv: Organizācija ar organizācijas identifikatoru: 1 Organizācijā pastāv komanda ar komandas identifikatoru 1. Komandā ir izveidota</i></p>	<p><i>Uzdevuma kods: TS-01 Uzdevuma nosaukums: Testa uzdevums Atsauces: Atsauces uzdevuma identifikators: 1 Atsauces veida identifikators: 1</i></p>	<p><i>Uzdevuma stāvoklis ir Izpildītāja piešķiršana Atbildes kods: 200 (Ok) Tiek atgriezts: Id: 2 (uzdevuma ieraksts, kas tika izveidots datu bāzē) Pazīmes identifikators: 1 Uzdevuma kods: TS-01</i></p>

<p><i>pazīme ar pazīmes identifikatoru 1.</i></p> <p><i>Pazīmē ir izveidots uzdevums ar identifikatoru 1</i></p>	<p><i>Organizācijas identifikators: 1</i></p> <p><i>Komandas identifikators: 1</i></p> <p><i>Pazīmes identifikators: 1</i></p>	<p><i>Uzdevuma statusa identifikators: 1 (Izpildītāja piešķiršana)</i></p> <p><i>Uzdevuma nosaukums: Testa uzdevums</i></p> <p><i>Uzdevuma apraksts: -</i></p> <p><i>Izpildīts: false</i></p>
<p><i>Sistēmā pastāv:</i></p> <p><i>Organizācija ar organizācijas identifikatoru: 1</i></p> <p><i>Organizācijā pastāv komanda ar komandas identifikatoru 1.</i></p> <p><i>Komandā ir izveidota pazīme ar pazīmes identifikatoru 1.</i></p> <p><i>Pazīmē ir izveidots uzdevums ar identifikatoru 1</i></p>	<p><i>Uzdevuma kods: TS-01</i></p> <p><i>Uzdevuma nosaukums: Testa uzdevums</i></p> <p><i>Atsauces:</i></p> <p><i>Atsauces uzdevuma identifikators: 1</i></p> <p><i>Atsauces veida identifikators: 2</i></p> <p><i>Organizācijas identifikators: 1</i></p> <p><i>Komandas identifikators: 1</i></p> <p><i>Pazīmes identifikators: 1</i></p>	<p><i>Uzdevuma stāvoklis ir Saistīts</i></p> <p><i>Atbildes kods: 200 (Ok)</i></p> <p><i>Tiek atgriezts:</i></p> <p><i>Id: 2 (uzdevuma ieraksts, kas tika izveidots datu bāzē)</i></p> <p><i>Pazīmes identifikators: 1</i></p> <p><i>Uzdevuma kods: TS-01</i></p> <p><i>Uzdevuma statusa identifikators: 3 (Saistīts)</i></p> <p><i>Uzdevuma nosaukums: Testa uzdevums</i></p> <p><i>Uzdevuma apraksts: -</i></p> <p><i>Izpildīts: false</i></p>
<p><i>Sistēmā pastāv:</i></p> <p><i>Organizācija ar organizācijas identifikatoru: 1</i></p> <p><i>Organizācijā pastāv komanda ar komandas identifikatoru 1.</i></p> <p><i>Komandā ir izveidota pazīme ar pazīmes identifikatoru 1.</i></p> <p><i>Pazīmē ir izveidots</i></p>	<p><i>Id: 1</i></p> <p><i>Uzdevuma kods: N-TS-01</i></p> <p><i>Uzdevuma nosaukums: Jauns testa uzdevums</i></p> <p><i>Organizācijas identifikators: 1</i></p> <p><i>Komandas identifikators: 1</i></p> <p><i>Pazīmes identifikators: 1</i></p>	<p><i>Uzdevuma dati tika rediģēti</i></p> <p><i>Atbildes kods: 200 (Ok)</i></p> <p><i>Tiek atgriezts:</i></p> <p><i>Id: 2 (uzdevuma ieraksts, kas tika izveidots datu bāzē)</i></p> <p><i>Pazīmes identifikators: 1</i></p> <p><i>Uzdevuma kods: N-TS-01</i></p> <p><i>Uzdevuma statusa identifikators: 1 (Izpildītāja piešķiršana)</i></p> <p><i>Uzdevuma nosaukums: Jauns testa uzdevums</i></p>

<i>uzdevums ar identifikatoru</i> <i>1</i>		<i>Uzdevuma apraksts: -</i> <i>Izpildīts: false</i>
---	--	--

Tabulā 4.8. ir aprakstīts funkcijas 2.2.4.3. *Uzdevuma izpildītāja iestatīšana vienībtestēšana*. Testpiemēru datu sagatavē tiek pieņemts kā pastāv organizācijas komanda, kurai tika piešķirti organizācijas dalībnieki un organizācijas lomas. Komandas ietvaros tika izveidota pazīmē un pazīmes ietvaros tika izveidots uzdevums, kuram tiek piešķirts izpildītājs.

4.8. tabula

Funkcijas 2.2.4.3. Uzdevuma izpildītāja iestatīšana vienībtestēšana

<i>Testpiemēra datu sagatave</i>	<i>Ievaddati</i>	<i>Sagaidāmais rezultāts</i>
<i>Organizācijas identifikators: 1</i> <i>Komandas identifikators: 1</i> <i>Pazīmes identifikators: 1</i> <i>Uzdevuma identifikators: 1</i> <i>Uzdevumam nav saistīti uzdevumu.</i>	<i>Lietotāja identifikators: 2</i> <i>Organizācijas identifikators: 1</i> <i>Uzdevuma identifikators: 1</i> <i>Izpildītāja veids: Persona</i>	<i>Uzdevuma (id: 1) stāvoklis tiek mainīts uz Piešķirts.</i> <i>Uzdevuma (id:1) izpildītāja identifikators: 2</i> <i>Tiek izveidots paziņojums lietotājam ar identifikatoru 2 par uzdevuma piešķiršanu</i> <i>Atbildes kods: 200 (Ok)</i>
<i>Organizācijas dalībnieku identifikatori: [1, 2, 3, 4]</i> <i>Komandas dalībnieku identifikatori: [1, 2, 3]</i> <i>Komandas lomas identifikatori: [8,9]</i> <i>Komandas dalībnieki ar lomu (lomas id: 8): [2, 3]</i>	<i>Lomas identifikators: 8</i> <i>Organizācijas identifikators: 1</i> <i>Uzdevuma identifikators: 1</i> <i>Izpildītāja veids: Loma</i>	<i>Uzdevuma (id: 1) stāvoklis netiek mainīts.</i> <i>Uzdevuma (id:1) izpildītāja lomas identifikators: 8</i> <i>Tiek izveidots paziņojums lietotājam ar identifikatoru 2 un 3 par lomas uzdevuma izveidošanu.</i> <i>Atbildes kods: 200 (Ok)</i>
	<i>Lietotāja identifikators: 4</i> <i>Organizācijas identifikators: 1</i> <i>Uzdevuma identifikators: 1</i>	<i>Atbildes kods: 409 (Validation Error)</i>

	<i>Izpildītāja veids: Persona</i>	<i>Kļūdu paziņojums Izvēlētais lietotājs nav komandas dalībnieks</i>
	<i>Lomas identifikators: 10 Organizācijas identifikators: 1 Uzdevuma identifikators: 1 Izpildītāja veids: Loma</i>	<i>Atbildes kods: 409 (Validation Error) Kļūdu paziņojums Izvēlēta loma nepastāv komandas lomu kopā</i>
	<i>Lomas identifikators: 9 Organizācijas identifikators: 1 Uzdevuma identifikators: 1 Izpildītāja veids: Loma</i>	<i>Atbildes kods: 409 (Validation Error) Kļūdu paziņojums Komandā nepastāv komandas dalībnieku ar izvēlēto lomu</i>

5. PROJEKTA ORGANIZĀCIJA

Plānošanas un laika pārvaldības sistēmas projektējumam un izstrādei tika izmantots ūdenskrituma dzīves cikla modelis ar mērķi nedefinēt sistēmas moduļu sadarbību un īstenot nepieciešamo datu struktūru, kas tiks izmantota moduļa funkcionalitātes īstenošanā. Sistēma ir izstrādāta pēc darba autora iniciatīvas noslēguma darba studiju programmā *Programmēšanas un datortīklu administrēšana* ietvaros. Izstrādes un projektējuma laikā projektā piedalījās darba autors, kurš izstrādāja sistēmas programmatūras dokumentāciju un sistēmas programmatūru, un darba vadītājs, kurš piedāvāja sistēmas programmatūras dokumentācijas labojumus un atzīmēja piezīmes, kas turpmāk ietekmēja sistēmas programmatūras izstrādes procesu. Komunikācija starp darba vadītāju un darba autoru notika ar e-pastu palīdzību, kur tika piedāvāti sistēmas programmatūras dokumentācijas melnraksti.

5.1. Kvalitātes nodrošināšana

Plānošanas un laika pārvaldības sistēmas programmatūras izstrādes kvalitāte tika nodrošināta, ievērojot un veicot sekojošas darbības:

- C# valodas programmatūras koda noformēšanai lietojumprogrammas saskarnes izstrādē tika izmantots Microsoft dokumentācijas nosaukumu piešķiršanas vadlīnijas. [9]
- Lietojumprogrammas saskarnes izstrādē tika izmantots Visual Studio 2019 izstrādes vide, kura veic automātisku koda sintakses pārbaudi kā arī piedāvā koda formatējuma labojumus un uzlabojumus. Koda formatēšanai tika izmantots noklusētājs Visual Studio 2019 formatētājs.
- TypeScript valodas programmatūras koda noformēšanai tīmekļa vietnes izstrādē tika izmantots Prettier formatētājs ar noklusētiem formatējuma iestatījumiem, kas tika pievienots Visual Studio Code izstrādes vidē.
- Tīmekļa vietnes izstrādē tika izmantotas Sapper un Svelte satvari, kas samazina koda daudzumu un piedāvā stilu veidošanu noteiktam vietnes skatam.
- Programmatūras kods tiek komentēts vietās, kur darba autors uzskata, ka mainīgo un metožu nosaukumi pilnībā nedod sapratni par glabātu vērtību vai veikto darbību.

- Sistēmas izstrādes gaitā tika veikts ieviestas funkcionalitātes testēšana un lietojumprogrammas saskarnes izstrādes gaitā tika izstrādāti testpiemēri vienībtestēšanai.

Plānošanas un laika pārvaldības sistēmas programmatūras dokumentācijas kvalitāte tika nodrošināta, ievērojot un veicot sekojošas darbības:

- Sistēmas programmatūras dokumentācija tika izstrādāta ievērojot programmatūras prasību specifikācijas, programmatūras projektējuma apraksta un testēšanas dokumentācijas nozares standartiem un prasībām. [1] [2] [3] [4]
- Sistēmas projektējuma un izstrādes procesa gaitā sistēmas projektējums tika apspriests ar darba vadītāju.

5.2. Konfigurāciju pārvaldība

Plānošanas un laika pārvaldības sistēmas programmatūras koda konfigurācijas pārvaldība tika veikta, izmantojot Git versiju kontroles rīku ar TortoiseGit klientu, kurš piedāvāja saskarni konfigurācijas pārvaldībai. Versiju kontroles rīks piedāvāja iespējas saglabāt un pārvaldīt sistēmas atjaunojumus, kas tika atsūtīti publiskā repozitorijā un kas atrodas GitHub versiju kontroles servisā. Versiju kontroles rīks piedāvāja iespējas veikt izstrādes procesa vēsturi un nodrošināja piekļuvi sistēmas programmatūras kodam ieinteresētam personām.

Datu bāzes konfigurāciju pārvaldības tika izmantota .Net platformas EntityFramework Core lietojumprogrammas saskarne, kas veidoja datu bāžu migrācijas. Migrācijas tika saglabātas Git versiju kontrole un, pēc migrāciju pielietošanas datu bāzes serverī, izmantota migrācijas versija/nosaukums tika atzīmēts tabulā _EFMigrationHistory Microsoft SQL serverī.

Sistēmas izstrādes gaitā tika ieviesti konfigurācijas faili, kur tiek saglabāti konfigurācijas iestatījumi, tas ir, lietojumprogrammas saskarnes adrese, datu bāzes savienojuma virkne u.t.t.

Plānošanas un laika pārvaldības sistēmas programmatūras dokumentācijas konfigurācijas pārvaldība tika veikta, izmantojot OneDrive servisu, kur iespējams apskatīt programmatūras dokumentācijas izmaiņu vēsturi.

5.3. Darbietilpības novērtēšana

Plānošanas un laika pārvaldības sistēmas darbietilpības novērtēšanā tika veikta, izmantojot ekspertu metodi. [10] Izmantojot izveidotas sistēmas darījumas prasības (sk. 1.4.

Darījumasprasības sekciju), tika izveidota tabula (sk. 5.1. tabulu), kur tika prognozēts/aprēķināts nepieciešamais stundu skaits, lai izpildītu minēto darījumasprasību. Tabula ir attēlotas pesimistiskā, reālistiskā un optimistiskā prognoze.

5.1. tabula

Darījumasprasību darbietilpības prognoze

<i>Darījumasprasības apraksts</i>	<i>Darbietilpības prognozes, stunda</i>		
	<i>Pesimistiskā</i>	<i>Reālistiskā</i>	<i>Optimistiskā</i>
<i>Programmatūras dokumentācijas izstrāde:</i>	-	-	-
<i>Autentifikācijas un autorizācijas mehānisma apraksts</i>	24	16	8
<i>Organizācijas struktūras izveidošanas funkcionalitātes apraksts</i>	48	32	24
<i>Izstrādes procesa regulēšanas funkcionalitātes apraksts</i>	48	40	32
<i>Sistēmas sazināšanas funkcionalitātes apraksts</i>	32	24	16
<i>Laika pārvaldības reģistrēšanas funkcionalitātes apraksts</i>	20	16	8
<i>Laika pārvaldības attēlošanas funkcionalitātes apraksts</i>	32	24	16
<i>Datu bāzes modeļu izveidošana</i>	24	16	8
<i>Programmatūras projektējuma apraksts</i>	80	56	40
<i>Saskarņu apraksts</i>	40	32	24
<i>Testpiemēru izveidošana</i>	32	24	16
<i>Programmatūras koda izstrāde:</i>	-	-	-
<i>Autentifikācijas un autorizācijas mehānisms</i>	32	24	16
<i>Organizācijas struktūras izveidošanas funkcionalitāte</i>	64	40	24

<i>Izstrādes procesa regulēšanas funkcionalitāte</i>	72	56	40
<i>Sistēmas sazināšanas funkcionalitāte</i>	24	16	8
<i>Laika pārvaldības reģistrēšanas funkcionalitāte</i>	32	24	16
<i>Laika pārvaldības attēlošanas funkcionalitāte</i>	32	24	16
<i>Datu bāzes struktūras izveidošana</i>	24	16	8
<i>Vienībtestēšana</i>	32	24	16
<i>Kopā</i>	692	504	336

Izmantojot formulu 1., tika prognozēta darbietilpība stundās:

$$\text{Darbietilpība} = \frac{\text{Pesimistiskā} + 4 * \text{Reālistiskā} + \text{Optimistiskā}}{6} \quad (1.)$$

$$\frac{692 + 4 * 504 + 336}{6} = 507.33 \text{ stundas}$$

Tā kā stundu skaits vienā personmēnesī ir vienāds ar 160 stundām, tad prognozētā darbietilpība personmēnešos ir vienāda ar:

$$\frac{507.33}{160} = 3.17 \text{ personmēneši}$$

Izstrādātajā plānošanas un laika pārvaldības sistēmas programmatūras kodā ir 4005 loģiskās rindīņas (sk. 5.1. att.). Balstoties uz QSM (Quantitative Software Management) atklātā piekļuvē izliktām etalontabulām [11], kur tiek attēlota informācija par pabeigtiem projektiem un tiek aprēķinātas minimālās, maksimālās, vidējās vērtības un mediānas, var secināt ka loģisko rindīņu skaits atbilst prognozētai darbietilpībai. Secinājumi balstās uz tabulas *Business Systems Implementation Unit (New and Modified IU) Benchmarks* datiem (3 personmēnešos tiks izstrādāti no 131 līdz 3115 loģisku koda rindīņu).

Language	files	blank	comment	code
JavaScript	2557	154851	66121	1568205
JSON	13	0	0	14097
C#	38	602	116	2431
Svelte	16	197	91	1183
XML	5	16	45	982
C# Designer	2	325	2	929
TypeScript	9	54	24	391
CSS	116	7	0	231
Markdown	2	73	0	87
Visual Studio Solution	1	1	1	23
HTML	1	5	10	19
MSBuild script	1	4	0	19
C# Generated	2	8	20	12
SVG	2	0	0	2
SUM:	2765	156143	66259	1588611

5.1. att. CLOC (Count Lines of Code) rīka izmantošanas rezultāts.

REZULTĀTI UN DISKUSIJA

Kvalifikācijas darba ietvaros tika izstrādāta tīmekļa vietne, kas piedāvā pārvaldīt organizācijas izstrādes procesus un uzskaitīt laiku, kas tika atvēlēts sistēma reģistrētam uzdevumam. Plānošanas un laika pārvaldības sistēma tika izstrādāta, izmantojot C# un TypeScript programmēšanas valodas, ASP.NET lietojumprogrammas saskarni, Sapper un Svelte satvarus tīmekļa vietnes veidošanai. Tika iegūta pieredze projektējuma izstrādes implementēšanā un projekta darbietilpības novērtēšanā. Iegūta pieredze automatizētu vienībtestēšanas izveidošanā, izmantojot xUnit testēšanas satvaru.

SECINĀJUMI

Plānošanas un laika pārvaldības sistēma izpilda savu galveno funkcionalitāti – veic organizācijas dalībnieku laika uzskaiti izvēlētajām uzdevumam un attēlo izstrādes procesu, kas tika reģistrēts sistēmā. Sistēmas izstrādes gaitā tika konstatēti projektējuma trūkumi, kas netika konstatēti dokumentācijas veidošanas procesā, piemēram, sistēma nav izveidota funkcionalitāte uzdevumu/pazīmju versijas saglabāšanai un demonstrēšanai. Programmatūras dokumentācijas posmā netika veidotas ieinteresētu lietotāju aptaujas par sistēmas funkcionalitātēm, kas var potenciāli uzlabot izstrādātas sistēmas kvalitāti.

Darba autors ieguva pieredzi projektējumu izveidošanā ar ūdenskrituma dzīves cikla modeļa palīdzību kā arī apguva tehnoloģijas tīmekļa vietnes, lietojumprogrammas saskarnes un vienībtestēšanas izveidei.

IZMANTOTĀ LITERATŪRA UN AVOTI

- [1] R. Čevere un M. Lučkina, «LVS 68:1996 Programmatūras prasību specifikācijas ceļvedis» 27. Martā 1996.
- [2] R. Čevere un M. Lučkina, «LVS 72:1996 Ieteicamā prakse programmatūras projektējuma aprakstīšanai» 27. Martā 1996.
- [3] R. Čevere un M. Lučkina, «LVS 70: 1996 Programmatūras testēšanas dokumentācija» 27. Martā 1996.
- [4] Object Management Group, «ISO/IEC 19505-2:2012 Information technology — Object Management Group Unified Modeling Language (OMG UML)» Aprīlī 2012.
- [5] Web Hypertext Application Technology Working Group, «HTML Living Standard» 5. Janvārī 2022. [Tiešsaiste]. Pieejams: <https://html.spec.whatwg.org/multipage/>. [Piekļūts 8. Janvārī 2022].
- [6] World Wide Web Consortium, «CSS Snapshot 2021» 31. Decembrī 2021. [Tiešsaiste]. Pieejams: <https://www.w3.org/TR/css-2021/>. [Piekļūts 8. Janvārī 2022].
- [7] Ecma International, «ECMA-262: ECMAScript 2021 Language Specification» Jūnijā 2021. [Tiešsaiste]. Pieejams: https://www.ecma-international.org/wp-content/uploads/ECMA-262_12th_edition_june_2021.pdf. [Piekļūts 8. Janvārī 2022].
- [8] K. Moriarty, B. Kaliski un A. Rusch, «PKCS #5: Password-Based Cryptography Specification Version 2.1» Janvārī 2017. [Tiešsaiste]. Pieejams: <https://datatracker.ietf.org/doc/html/rfc8018>. [Piekļūts 8. Janvārī 2022].
- [9] Microsoft Corporation, «Naming Guidelines» Microsoft Corporation, 15. Septembrī 2021. [Tiešsaiste]. Pieejams: <https://docs.microsoft.com/en-us/dotnet/standard/design-guidelines/naming-guidelines>. [Piekļūts 9. Janvārī 2022].
- [10] R. S. Pressman, «Estimation for Software Projects» *Software Engineering: A Practitioner's Approach*, 7th red., 2010.
- [11] Quantitative Software Management, «QSM Benchmark Tables» [Tiešsaiste]. Pieejams: <https://www.qsm.com/resources/qsm-benchmark-tables>. [Piekļūts 9. Janvārī 2022].

PIELIKUMI

1. pielikums. Lietojumprogrammas saskarnes servisu konfigurācija

Lietojumprogrammas saskarnes servisu konfigurācijas pirmkods, kas demonstrē ASP.NET Identity konfigurācijas iestatījumus, savienojuma konfigurāciju ar Microsoft SQL Server, izmantojot EntityFramework un HTTP pieprasījuma cauruļvada apstrādes process:

```
using Microsoft.AspNetCore.Builder;
using Microsoft.AspNetCore.Hosting;
using Microsoft.AspNetCore.Identity;
using Microsoft.EntityFrameworkCore;
using Microsoft.Extensions.Configuration;
using Microsoft.Extensions.DependencyInjection;
using Microsoft.Extensions.Hosting;
using Microsoft.OpenApi.Models;
using Timeline_Management.Entities;

namespace Timeline_Management
{
    public class Startup
    {
        public Startup(IConfiguration configuration)
        {
            Configuration = configuration;
        }

        public IConfiguration Configuration { get; }

        // This method gets called by the runtime. Use this method to add
        // services to the container.
        public void ConfigureServices(IServiceCollection services)
        {
            //Configuring Microsoft SQL Server connection with the
            //database
            services.AddDbContext<TimelineManagementContext>(config =>
            {
                config.UseSqlServer(Configuration.GetConnectionString("TimelineManagementD
                b"));
            });

            //Configuring ASP.NET Identity for the user authentication and
            //authorization
            services
                .AddIdentity<User, IdentityRole<int>>(options =>
            {
                options.User.RequireUniqueEmail = true;
                options.Password.RequireDigit = true;
                options.Password.RequiredLength = 6;
                options.Password.RequireNonAlphanumeric = true;
                options.Password.RequireLowercase = true;
            });
        }
    }
}
```

```

        options.Password.RequireUppercase = true;
    })
    .AddEntityFrameworkStores<TimelineManagementContext>()
    .AddDefaultTokenProviders();

    services.AddCors();
    services.AddControllers();
    services.AddSwaggerGen(c =>
    {
        c.SwaggerDoc("v1", new OpenApiInfo { Title =
"Timeline_Management", Version = "v1" });
    });
}

// This method gets called by the runtime. Use this method to
configure the HTTP request pipeline.
public void Configure(IApplicationBuilder app, IWebHostEnvironment
env)
{
    if (env.IsDevelopment())
    {
        app.UseDeveloperExceptionPage();
        app.UseSwagger();
        app.UseSwaggerUI(c =>
c.SwaggerEndpoint("/swagger/v1/swagger.json", "Timeline_Management v1"));
    }

    app.UseHttpsRedirection();

    app.UseRouting();

    app.UseCors(options =>
    {
        options
        .SetIsOriginAllowed(origin => true)
        .AllowAnyHeader()
        .AllowAnyMethod()
        .AllowCredentials();
    });

    app.UseAuthentication();
    app.UseAuthorization();

    app.UseEndpoints(endpoints =>
    {
        endpoints.MapControllers();
    });
}
}
}
}

```

2. pielikums. Komunikācijā starp tīmekļa vietni un lietojumprogrammas saskarni. 1.daļa

Tīmekļa vietnes autentifikācijas un autorizācijas serviss, kuru uzdevums ir veidot pieprasījumus lietojumprogrammas saskarnei, lai mijiedarbotos ar sistēmas datu bāzi. Sekojoša pielikumā tiek attēlots pirmkods, kas izpildās lietotāja pārlūkā.

```
import { apiLink } from "../stores/serverInformation";
import { isAuthenticated, authenticatedUser } from
"../stores/authenticated";

const unauthorizedStatusCode = 401;
const validationErrorStatusCode = 409;

class AuthorizationService {
  //Method is used to retrieve user information if current user contains
  authentication cookie
  //Information about user and authentication status will be stored in
  authenticated stores
  isCurrentUserAuthorized = async () => {
    await fetch(apiLink + "/api/auth/currentUser", {
      method: "GET",
      credentials: "include",
      headers: { "Content-Type": "application/json" },
    })
    .then((response) => response.json())
    .then((data) => {
      //If API returns unauthorized response, setting authenticated
      status to false and deleting existing user information
      if (data.status != undefined && data.status ==
      unauthorizedStatusCode) {
        isAuthenticated.set(false);
        authenticatedUser.set(null);
      } else {
        let currentUser = {
          Id: data.id,
          FirstName: data.firstName,
          SecondName: data.secondName,
          LastName: data.lastName,
          Username: data.username,
          Email: data.email,
        };

        authenticatedUser.set(currentUser);
      }
    });
  }
}
```

```

        isAuthenticated.set(true);
    }
})
.catch((error) => {
    console.log(error);
});
});

//Service method that will provide user register functionality and fill
//authenticated stores with registered user information
//Return registration error message. If user successfully registered in
the system then returns empty string
register = async (userFormData) => {
    let errorResponse = "";

    await fetch(apiLink + "/api/auth/register", {
        method: "POST",
        headers: { "Content-Type": "application/json" },
        credentials: "include",
        body: JSON.stringify({
            FirstName: userFormData.firstName,
            SecondName: userFormData.secondName,
            LastName: userFormData.lastName,
            Username: userFormData.username,
            Email: userFormData.email,
            Password: userFormData.password,
        }),
    })
    .then((response) => response.json())
    .then((data) => {
        //If API returns validation error response, return error message
to register form
        if (
            data.status !== undefined &&
            data.status === validationErrorStatusCode
        ) {
            errorResponse = data.detail;
        } else {
            //Map response information to existing User model
            let registeredUser = {
                Id: data.id,
                FirstName: data.firstName,
                SecondName: data.secondName,
            }
        }
    })
}

```

```

        LastName: data.lastName,
        Username: data.username,
        Email: data.email,
    };

    authenticatedUser.set(registeredUser);
    isAuthenticated.set(true);

    errorResponse = "";
  }
})
.catch((error) => {
  console.log(error);
});

return errorResponse;
};

//Service method that will provide user log in functionality and fill
//authenticated stores with user information
//Return authentication error message.If user succesfully logged in the
system then returns empty string
login = async (username, password) => {
  let errorResponse = "";

  await fetch(apiLink + "/api/auth/login", {
    method: "POST",
    headers: {
      "Content-Type": "application/json",
    },
    credentials: "include",
    body: JSON.stringify({
      Username: username,
      Password: password,
    }),
  })
  .then((response) => response.json())
  .then((data) => {
    //If API returns validation error response, return error message
to register form
    if (
      data.status !== undefined &&
      data.status === validationErrorStatusCode

```

```

    ) {
      isAuthenticated.set(false);
      authenticatedUser.set(null);

      errorResponse = data.detail;
    } else {
      let currentUser = {
        Id: data.id,
        FirstName: data.firstName,
        SecondName: data.secondName,
        LastName: data.lastName,
        Username: data.username,
        Email: data.email,
      };

      authenticatedUser.set(currentUser);
      isAuthenticated.set(true);

      errorResponse = "";
    }
  })
  .catch((error) => {
    console.log(error);
  });

  return errorResponse;
};

//Service method that will provide user logout functionality
logout = async () => {
  await fetch(apiLink + "/api/auth/logout", {
    method: "POST",
    headers: { "Content-Type": "application/json" },
    credentials: "include",
  });

  isAuthenticated.set(false);
  authenticatedUser.set(null);
};
}

export const authorizationService = new AuthorizationService();

```

3. pielikums. Komunikācijā starp tīmekļa vietni un lietojumprogrammas saskarni. 2.daļa

Lietojumprogrammas saskarnes autentifikācijas un autorizācijas kontrolleris, kuru uzdevums ir apstrādāt tīmekļa vietnes pieprasījumus, lai modificēt, izveidot vai apskatīt datu bāzē saglabātu informāciju. Sekojoša pielikumā tiek attēlots pirmkods, kas izpildās sistēmas serverī.

```
using Microsoft.AspNetCore.Identity;
using Microsoft.AspNetCore.Mvc;
using System.Threading.Tasks;
using Timeline_Management.Controllers.Dtos;
using Timeline_Management.Entities;

namespace Timeline_Management.Controllers
{
    [Route("api/auth")]
    [ApiController]
    public class AuthController : Controller
    {
        private readonly UserManager<User> _userManager;
        private readonly SignInManager<User> _signInManager;

        public AuthController(UserManager<User> userManager,
            SignInManager<User> signInManager)
        {
            _userManager = userManager;
            _signInManager = signInManager;
        }

        /// <summary>
        /// Registers new user in the system if request meets validation
        requirement
        /// </summary>
        /// <param name="request">User request data</param>
        /// <returns>Created user accoutn information</returns>
        [HttpPost("register")]
        public async Task<ActionResult<UserRegisterResponse>>
            Register(UserRegisterRequest request)
        {
            if (!ModelState.IsValid)
            {
                //Register information does not pass validation
                requirements
                return ValidationProblem("Reģistrācija informācija
                neatbilst reģistrācijas formas validācijas prasībām", statusCode: 409);
            }

            //Validate other request requirements that can't be included
            in model validation
            var validationResult = await IsRegisterRequestValid(request);

            if (!validationResult.IsValid)
            {
```

```

        return ValidationProblem(validationResult.ErrorMessage,
statusCode: 409);
    }

    var user = await CreateUser(request);

    if (user is null)
    {
        //User can't be created because of userManager error (Not
valid password).
        return ValidationProblem("Neveiksmīga lietotāju konta
izveidošana", statusCode: 409);
    }

    await _signInManager.SignInAsync(user, false);

    return Ok(new UserRegisterResponse()
    {
        Id = user.Id,
        FirstName = user.FirstName,
        SecondName = user.SecondName,
        LastName = user.LastName,
        Username = user.UserName,
        Email = user.Email
    });
}

/// <summary>
/// User authentication using provided request information
/// </summary>
/// <param name="request"> User login request information</param>
/// <returns>User account information if user succesfully logged
in the system </returns>
[HttpPost("login")]
public async Task<ActionResult<UserLoginResponse>>
Login(UserLoginRequest request)
{
    if (!ModelState.IsValid)
    {
        //Login information does not pass validation requirements
        return ValidationProblem("Pietekšanās informācija
neatbilst pieteikšanas formas validācijas prasībām", statusCode: 409);
    }

    var result = await
_signInManager.PasswordSignInAsync(request.Username, request.Password,
isPersistent: false, lockoutOnFailure: false);

    if (!result.Succeeded)
    {
        return ValidationProblem("Tika ievadīts nepareizs
pieteikuma vārds vai parole.", statusCode: 409);
    }

    var user = await
_userManager.FindByNameAsync(request.Username);

```

```

        return Ok(new UserLoginResponse()
        {
            Id = user.Id,
            FirstName = user.FirstName,
            SecondName = user.SecondName,
            LastName = user.LastName,
            Username = user.UserName,
            Email = user.Email
        });
    }

    /// <summary>
    /// User logout functionality
    /// </summary>
    [HttpPost("logout")]
    public async Task<IActionResult> Logout()
    {
        if (!this.User.Identity.IsAuthenticated)
        {
            return Unauthorized();
        }

        await _signInManager.SignOutAsync();
        return Ok();
    }

    /// <summary>
    /// Get authenticated user account information
    /// </summary>
    [HttpGet("currentUser")]
    public async Task<ActionResult<GetCurrentUserResponse>>
GetCurrentUser()
    {
        if (!this.User.Identity.IsAuthenticated)
        {
            return Unauthorized();
        }
        var user = await _userManager.GetUserAsync(this.User);

        return Ok(new GetCurrentUserResponse
        {
            Id = user.Id,
            FirstName = user.FirstName,
            SecondName = user.SecondName,
            LastName = user.LastName,
            Username = user.UserName,
            Email = user.Email
        });
    }

    /// <summary>
    /// Creates new User account in the system
    /// </summary>
    /// <param name="request"></param>
    /// <returns></returns>
    private async Task<User> CreateUser(UserRegisterRequest request)

```

```

    {
        var user = new User()
        {
            FirstName = request.FirstName,
            SecondName = request.SecondName,
            LastName = request.LastName,
            UserName = request.Username,
            Email = request.Email
        };

        var result = await _userManager.CreateAsync(user,
request.Password);

        if (result.Succeeded)
        {
            return user;
        }
        else
        {
            //User can't be created, return null
            return null;
        }
    }

    /// <summary>
    /// User registration request validation. Validates if user login
name and email isn't registered in the system
    /// </summary>
    /// <returns>Validation result that consists of validation result
and error message (if error occurred) </returns>
    private async Task<ValidationResult>
IsRegisterRequestValid(UserRegisterRequest request)
    {
        var validationResult = new ValidationResult()
        {
            IsValid = true,
            ErrorMesasge = string.Empty
        };
        var user = await
_userManager.FindByNameAsync(request.Username);

        if (user != null)
        {
            //User with selected username already exists in the system
scope
            validationResult.IsValid = false;
            validationResult.ErrorMessage = "Lietotājs ar norādītu
lietotājevārdu jau pastāv sistēmas ietvaros";

            return validationResult;
        }

        user = await _userManager.FindByEmailAsync(request.Email);

        if (user != null)
        {

```

```
scope           //User with selected Email already exists in the system
                validationResult.IsValid = false;
                validationResult.ErrorMessage = "Lietotājs ar norādītu e-
pastu jau pastāv sistēmas ietvaros";
                return validationResult;
            }
        }
    }
}
```

Kvalifikācijas darbs „Plānošanas un laika pārvaldības tīmekļa lietojumprogramma organizācijas darbībai un personīgai lietošanai” izstrādāts LU Datorikas fakultātē.

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

Autors: Roberts Gordins _____ __.01.2022

Rekomendēju/nerekomendēju darbu aizstāvēšanai

Darba vadītājs: Dr. dat. Maksims Kravcevs _____ __.01.2022

Recenzents: Andris Bozis

Darbs iesniegts 10.01.2022.

Kvalifikācijas darbu pārbaudījumu komisijas sekretārs: _____

Darbs aizstāvēts kvalifikācijas darbu pārbaudījuma komisijas sēdē __.01.2022. prot. Nr.

Komisijas sekretārs(-e): _____