

LATVIJAS UNIVERSITĀTE
DATORIKAS FAKULTĀTE

Kvalifikācijas darbs
„Company Event Manager”

Darba autors:
Maksims Pavlenko
Apl.nr.: mp17060

Darba vadītājs: Dr.dat. Ivo Odītis

ANOTĀCIJA

Sekojošais dokuments ir noslējuma darbs programminženierijas kursā. Darba autori: Maksims Pavļenko. Dokuments ir programmatūras prasību specifikācija produktam „Company Event Manager”, kura mērķis ir lielo uzņēmumu darbinieku profesionālo prasmju uzlabošana, sniegt informāciju par attīstības iespējām, kā arī palīdzētu orientēties ziņu plūsmā, laika plānošanai pašrealizācijai un uzlabot komandas darbu uzņēmumā.

Atslēgvārdi: izglītība, komandas darbs, pasākumi.

Abstract

The following document is the final work of the programme engineering course. The authors of the work: Maksims Pavlenko. The document is a specification of the software requirements for the “Company Event Manager”, which aims to improve the professional skills of employees of large companies by providing information on development opportunities, as well as helping you to orient yourself in the news flow and time planning for self-realization and to improve teamwork in the company.

Keywords: education, teamwork, events.

Satura rādītājs

Ievads	6
Nolūks	6
Darbības sfēra	6
Saistība ar citiem dokumentiem	6
Pārskats	6
Vārdnīca	7
1. Vispārējs apraksts	8
1.1. Esošā stāvokļa apraksts	8
1.2. Pasūtītājs.....	8
1.3. Produkta perspektīva	8
1.4. Darījumu prasības	8
1.5. Sistēmas lietotāji	9
1.6. Vispārējie ierobežojumi	10
1.7. Sistēmas pieņēmumi un atkarības	10
2 Konceptuālais datu bāzes apraksts	11
2.1. Konceptuālais datu bāzes apraksts.....	12
2.2. Funkcionālā vienību lietoti identifikācijas apzīmējumi.....	13
2.3. Funkcionālās prasības.....	14
2.3.1. Autentifikācijas funkcija	14
2.3.2. Lietotāja izveidošanas funkcija	14
2.3.3. Lietotāju dzēšanas funkcija.....	15
2.3.4. Paroles maiņas funkcija	15
2.3.5. Lietotāja datu maiņas funkcija.....	16
2.3.6. Paroles atjaunošanas funkcija.....	16
2.3.7. Notikumu pievienošanas funkcija.....	17
2.3.8. Notikumu saraksta atspoguļošanas funkcija.....	17
2.3.9. Komandas pievienošanas funkcija.....	18
2.3.10. Komandas saraksta atspoguļošanas funkcija.....	18
2.3.11. Biroja pievienošanas funkcija.....	19
2.3.12. Biroja saraksta atspoguļošanas funkcija.....	19
2.3.13. Lietotāju saraksta atspoguļošanas funkcija.....	20
2.3.14. Lietotāja profilā informācijas atspoguļošanas funkcija.....	20
2.3.15. Cita lietotāja profila informācija atspoguļošanas funkcija.....	21
2.3.16. Biroju informācijas atspoguļošanas funkcija.....	21
2.3.17. Komandu informācijas atspoguļošanas funkcija.....	22
2.3.18. Notikumu informācijas atspoguļošanas funkcija.....	22
2.4. Ievadlauku ierobežojumi.....	23
2.5. Nefunkcionālās prasības	24
2.5.1. Veiktspējas prasības	24
2.5.2. Izmantojamība	24
2.5.3. Drošības prasības	24
2.5.4. Pieejamības prasības	24
2.5.5. Uzturēšanas prasības	24
2.5.6. Uzticamības prasības	24

3. Programmatūras projektējuma apraksts	25
3.1. Datu bāzes projektējums	25
3.1. Datu bāzes loģiskais modelis	25
3.1.1 Pāreja no konceptuāla uz loģisko modeli.....	26
3.1.2. Datu bāzes fiziskais modelis	27
3.1.2.1 Datu bāzes tabulu apraksts	27
3.1.2.1.1 Tabulas User apraksts.....	27
3.1.2.1.2 Tabulas TeamEvent apraksts.....	28
3.1.2.1.3 Tabulas Team apraksts.....	28
3.1.2.1.4 Tabulas Office apraksts.....	29
3.1.2.1.5 Tabulas GlobalEvent apraksts.....	29
3.1.2.1.6 Tabulas EventDescription apraksts.....	30
3.1.2.1.7 Tabulas LocalEvent apraksts.....	30
3.1.2.1.8 Tabulas Roles apraksts.....	31
3.1.2.1.9 Tabulas Event_types apraksts.....	31
3.2 Daļējs funkciju projektējums	32
3.2.1 Eventa pievienošana.....	32
3.2.2 Lietotāja paroles maiņa.....	33
3.2.3 Sistēmas pamat funkcionalitātes klašu diagramma.....	34
3.3. Skatu projektējums	35
3.3.1 Lietotāja autorizācija.....	35
3.3.2 Autorizācijas mājas lapa.....	35
3.3.3 Notikumu saraksts.....	36
3.3.4 Komandu saraksts.....	36
3.3.5 Biroju saraksts.....	37
3.3.6 Lietotāju saraksts.....	37
3.3.7 Lietotāja izveides logs.....	38
3.3.8 Biroja izveidošanas logs.....	38
3.3.9 Notikumu izveides logs.....	39
3.3.10 Komandu izveides logs.....	39
4. Pielikums.....	40
4.1 Testēšanas dokumentācija.....	40
4.1.0 Lietotāju autorizācijas dati(Test roles).....	40
4.1.1 Autorizācijas tests.(login test).....	40
4.1.2 Lietotāja veidošanas tests. (create user test).....	40
4.1.3 Biroju veidošanas tests. (create office test).....	41
4.1.4 Notikumu veidošanas tests. (creat eeent test).....	41
4.1.5 Komandas veidošanas tests. (create team test).....	42
4.1.6 Atļauja veidot lietotājus tests. (create user access test).....	42
4.1.7Atļauja veidot birojus tests. (create offices access test)	43
4.1.8Atļauja veidot komandu tests. (create offices access test)	43
4.2 Programmatūras projekta organizācija.....	44
4.3 Kvalitātes nodrošināšana.....	45
4.4Darbietilpības novērtējums.....	46
4.5Secinājumi	47
4.6Izmantotā literatūra	48
4.7 Programmatūras pirmkods	49–53

IEVADS

Nolūks

Dokuments apraksta funkcionalitāti un prasības "Company Event Manager" produktam. Dokuments ietver sistēmas specifikāciju, prasības, ierobežojumus izmantošanu un uzturēšanu procesus, projektējumu utt.

Darbības sfēra

Sistēma „Company Event Manager” ir rīks lai uzlabotu darbinieku profesionālās prasmes un viņu daudzpusējo attīstību. Sistēma ir orientēta uz “Teambuilding” uzņēmuma ietvaros, uzlabot komunikāciju uzņēmuma un palielināt produktivitāti.

Saistība ar citiem dokumentiem

Dokuments sastādīts saskaņā ar standartu dokumentiem LVS 68:1996 „Programmatūras prasību specififikācijas ceļvedis” un LVS 72:1996 “Ieteicamā prakse programmatūras projektējuma aprakstīšanai”.

Pārskats

Dokuments sastāv no trim nodaļām.

1. Pirmajā nodaļā - produkta vispārīgs apraksts funkcionalitātes izskats un ierobežojumi un paskaidrots sistēmas lietotāju lomu. detalizēta informācija par funkciju un to funkcionālam iespējam.
2. Otrajā nodaļā tiek aprakstīts prasību specififikācija, kas ietver:
 - Konceptuālais modeli tā aprakstu
 - Funkcionālās prasības- detalizēta informācija par funkciju un to funkcionālam iespējam.
 - Nefunkcionālās prasības.
3. Otrajā nodaļā tiek aprakstīts datu bāzes projektējums, loģiskais un fiziskais modelis datu bāzes apraksts.

Vārdnīca

ID – identifikācijas dati.

DB – datu bāze

PK – primārā atslēga (primary key)

FK – ārējā atslēga (foreign key)

SQL – Structured Query Language (strukturēta vaicājumvaloda).

Team Building- ir kolektīvs termins dažāda veida darbībām, ko izmanto, lai uzlabotu sociālās attiecības un definētu lomas grupās, bieži vien iesaistot sadarbības uzdevumus.

Datetime- tips izmantots vērtībām, kas satur gan datuma, gan laika daļas.

PPS – programmatūras prasību specifikācija

String - Simbolu virkne (string) ir simboliskā tipa (piemēram, char) vērtību secība, kas reprezentē tekstuālu informāciju.

MNC - korporatīvā organizācija, kurai pieder vai kura kontrole precu vai pakalpojumu ražošana vismaz viena valstī, kas nav tas mitnes valsts.

Boss - uzņēmuma vadītājs.

1. Vispārējs apraksts

1.1 . Esošā stāvokļa apraksts

Aprakstītā sistēma ir tīmekļa lietojumprogramma, kuras mērķis ir uzlabot uzņēmuma iekšējo organizāciju, sniedzot informāciju par aktivitātēm. Sistēma “Company Event Manager” ir izstrādāta, lai ar to strādātu lielās organizācijās un uzlabotu „Team Building” un sniegtu vairāk filtrētu informācijas plūsmu par iekšējām darbībām uzņēmumā. “Company Event Manager” ļauj veidot notikumus, pamatojoties uz personas lomu uzņēmumā. Atšķirībā no citam lietojumprogrammām tā sadala darbības tādā veidā, ka lietotājam tiek nodrošinātas 3 sadaļas: globālie pasākumi, vietējie pasākumi un komandas pasākumi. Globālie notikumi tiek veidoti, lai atbalstītu ziņas uzņēmuma iekšienē. Šis veids atšķiras ar to, ka jebkādas ziņas un aktivitātes, kas notiek šādā veidā, ietekmēs visu uzņēmumu. Vietējie notikumu veidi ļauj pārvaldīt šīs grupas darbības reģiona (valsti), kuras mērķis ir nodrošināt lietotāju darbību atkarībā no reģiona, kurā tas atrodas (paredzēts cilvēkiem, kuri strādā ārvalstīs). Komandas notikumu mērķis ir uzlabot komandas darbu un attīstīt profesionālās prasmes. Šai sistēmai ir daudz analoģu, piemēram, “Eventbase”, “Whoa”, “Guidebook”, kas sniedz analogisku “Company Event Manager” funkcionalitāti aktivitātēm, izveidojot un pārvaldot tos, taču tas atšķiras ar to informācijas sniegšanas veidu un struktūru.

1.2 . Pasūtītājs

Sistēma tika izstrādāta pēc autora iniciatīvas, nākotnē potenciālais klients var kļūt par starptautisks uzņēmums.

1.3 . Produkta perspektīva

Sistēma nodrošina attiecīgās informācijas izplatīšanu un uzlabo komandas veidošanu

1.4 . Darījumu prasības

Sistēmas pamatprasījums ir organizāciju aktivitāšu glabāšana un apstrāde.

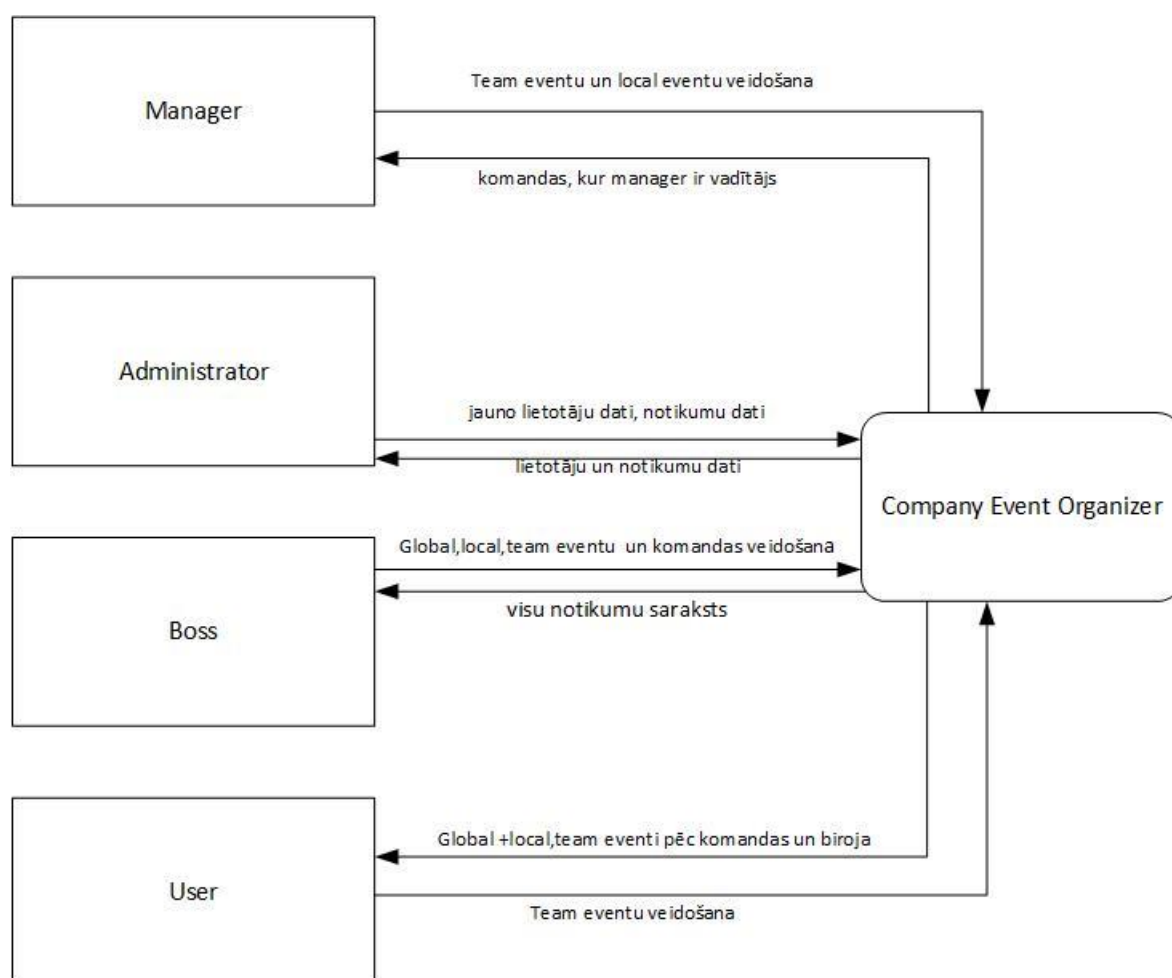
Šī pamatprasība ietver:

- Iespēju izveidot aktivitātes pamatojoties uz lomu uzņēmumā.
- Skatīt esošās aktivitātes atbilstoši savai grupai.
- Rediģēt personiskos pasākumus.
- Pievienot personiskos pasākumus.
- Glabāt aktivitātes atbilstoši savai grupai.

1.5 . Sistēmas lietotāji

Sistēma ir sadalīta trīs 3 grupās: administrators, reģistrēts lietotājs, neregistrēts lietotājs. Grupas ir izveidotas, pamatojoties uz to funkcionālajām iespējām. Pirmās grupas administratori ir atbildīgi par sistēmas uzturēšanu visiem sistēmas datiem. Otrā grupa ne pierēģistrēti lietotāji. Ir iespēja reģistrēties sistēmā, lai to varētu izmantot. Trešā grupa ir reģistrēts lietotāji. Autorizēti lietotāji var izmantot sistēmu atkarībā no personas loma.

Attēlā 1.1. var redzēt 0. līmeņa datu plūsmu diagrammu, kura atspoguļo datu plūsmas starp lietotājiem un sistēmu.



1.1. att. 0. līmeņa **DPD**

Sistēma nodrošina 4 lietotāju lomas:

- Manager
- Administrator
- Boss
- User

Manager - ir iespēja pievienot pasākumus visām kontrolētajām komandām, kā arī spēj izveidot vietējos pasākumus birojam, kuram viņš pieder.

Administrator- ir piekļuve katra lietotāja informācijai un iespējai rediģēt šo informāciju. Dzēst esošos lietotājus. Šīs roles galvenais mērķis ir sistēmas pārvaldība.

Boss - ir pilna piekļuve uzņēmuma aktivitātēm veidošanai , atšķirībā no citām lomām var izveidot globālus eventus visā uzņēmumā.

User – taista eventus pamatojoties uz komandu, kurai viņš pieder, saņem notikumus atkarībā no biroja, kuram viņš pieder, un komandu, kurai viņš ir pievienots.

1.6 Vispārējie ierobežojumi

- Sistēma ierobežo piekļuvi datu bāzei un aktivitāšu skatīšanai atkarībā no lietotāja lomas.
- Sistēmai nav paredzēta neregistrēta piekļuve.

1.7 Pieņēmumi un atkarības

Sistēma darbspēja ir paredzēta “Google Chrome”, “Opera”, “MozillaFirefox,” un “Microsoft Edge” pārlūkprogrammām. Ja viena no iepriekš uzskaitītajām pārlūkprogrammām ir pieejama, programma darboties neatkarīgi no operētājsistēmas.

2. Programmatūras prasību specifikācija

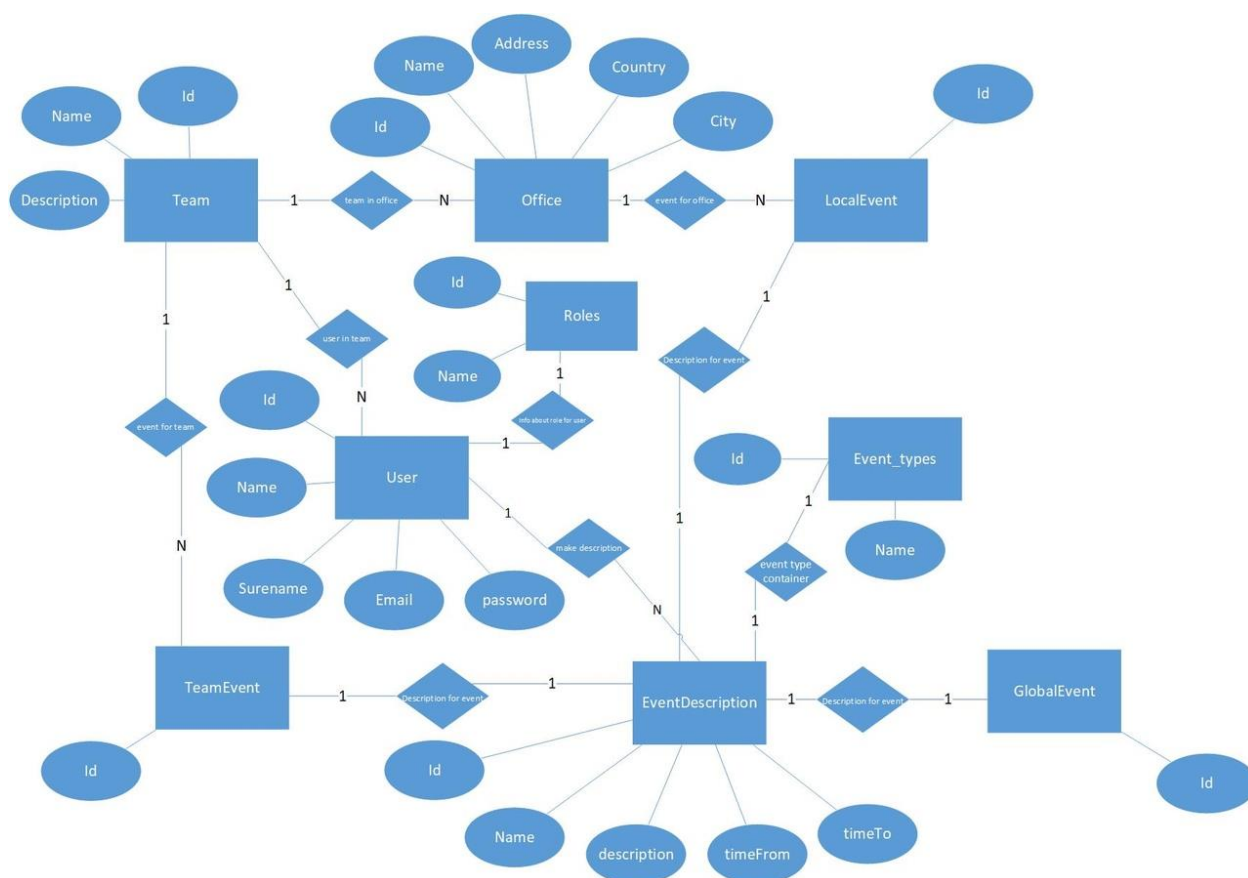
2.1 Konceptuālais datu bāzes apraksts

Datu bāze glabā informāciju par iepriekš minētajām vienībām. Lietotāji (Users) – lietotāju attiecība pret komandu (Team) ir definēta kā N:1, vienā komandā var būt daudz dažādu veidu lietotāju, savukārt lietotājs var tikt pievienots tikai vienai komandai. Komandas (Teams) ir pievienotas birojiem (Office) ar 1:N attiecību, komandai var būt viens birojs, taču viena biroja var būt daudz komandu. Aktivitātes apraksts (EventDescription) – tam ir savienojumi – pirmajam savienojumam ar lietotāju, kas izveido notikuma savienojumu, ir 1:N attiecība, jo lietotājs var izveidot daudzus notikumus atkarībā no viņa piekļuves, no kuras atkarīga notikumu klasifikācija. Klasifikācijas sadalījums tika izveidots ar mērķi sadalīt notikumu veidošanas līmeņus uzņēmumā, lai pastāvētu iespēja veikt izmaiņas, kas neietekmētu citu aktivitāšu funkcionalitāti. No notikumu apraksta tiek izveidotas trīs iepriekš minētās klasifikācijas. Vietējie notikumi (localevent) jeb pirmā klasifikācija ir saistīta ar notikumu aprakstu, attiecība ir 1:1, jo viens vietējais notikums satur vienu aprakstu par sevi. Vietēja notikuma mērķis ir izveidot darbības vadības vienību reģionā (valstī). Valstīs var būt vairāk nekā viena filiāle (programmatūras lietošana ietaupīs laiku, veidojot notikumus birojiem visā reģionā; pasākuma piemērs: valsts svētki). Otrā klasifikācija – komandas notikumi (teamevents) – komandas notikumu attiecība pret notikumu aprakstu 1:1, šāds savienojums ir pamatots ar to, ka komandas aktivitātei var būt tikai viens apraksts. Mērķis – sniegt informāciju tikai tai komandai, kurā lietotājs ir iekļauts. Trešā klasifikācija ir globālas aktivitātes ar attiecību 1:1, šai darbībai ir viens darbības apraksts – satur informāciju visiem uzņēmuma lietotājiem. “EventDescription” attiecība pret globālam aktivitātēm ir definēta kā 1:N, vienam birojam var būt daudz vietējo pasākumu. Komandas attiecība pret komandas aktivitātēm ir definēta kā 1:N, vienā komandā var būt vairākas komandas aktivitātes.

Datu bāzes konceptuālajā modelī ir sekojošas entitātes:

- User
- Team
- Office
- EventDescription
- GlobalEvent
- LocalEvent
- TeamEvent
- Roles
- Event_types

Attēlā 2.1 Demonstrējamā konceptuālu datu bāzes modeli, kas atspoguļo attiecības starp iepriekš minētajām vienībām. Modelis demonstrē datu bāzes vispārējo struktūru.



2.1.att. Datu bāzes konceptuālais modelis

2.2. Funkcionāl vienību lietoti identifikācijas apzīmējumi

Funkciju indikatoru apraksts:

- Sākotnējā daļa satur produkta iniciāļus kā pārliecinājums par vienības atbilstību aprakstāmajam produktam – CEM (Company Event Manager)
- Vienības tipa identifikators – **F** (funkcionāls - functional)
- Noteikta tipa kārtas numurs

2.2. tabula – Lietotāja lomas piekļuve apzīmējumi

Piekļuves apzīmējumi:	Identifikators
Visi lietotāji (all users)	AU
Administratoru grupa (administrator group)	AG
Uzņēmuma administratori un vadītāji (administrator and boss group)	AB
Parastie lietotāji (standart user group)	SA

2.3. tabula – Funkciju saraksts ar funkciju ID

Funkcija	Indefikators
Autentifikācijas funkcijas	CEM_AU_F_01
Lietotāja izveidošanas funkcijas	CEM_AB_F_01
Paroles maiņas funkcijas	CEM_AU_F_02
Lietotāju dzēšanas funkcija	CEM_AG_F_01
Lietotāja datu maiņas funkcija	CEM_AB_F_02
Paroles atjaunošanas funkcija	CEM_AU_F_03
Notikumu pievienošanas funkcija	CEM_AU_F_04
Komandas pievienošanas funkcija	CEM_AG_F_02
Biroja pievienošanas funkcija	CEM_AG_F_03
Notikumu saraksta atspoguļošanās funkcija	CEM_AU_F_05
Komandas saraksta atspoguļošanās funkcija	CEM_AU_F_06
Biroja saraksta atspoguļošanās funkcija	CEM_AU_F_07
Lietotāj saraksta atspoguļošanās funkcija	CEM_AU_F_08
Lietotāja profilā informācijas atspoguļošanās funkcija	CEM_AU_F_09
Cita lietotāja profila informācija atspoguļošanās funkcija	CEM_AU_F_10
Biroju informācijas atspoguļošanās funkcija	CEM_AU_F_11
Komandu informācijas atspoguļošanās funkcija	CEM_AU_F_12
Notikumu informācijas atspoguļošanās funkcija	CEM_AU_F_13

2.3. Funkcionālās prasības

2.3.1 Autentifikācijas funkcija

2.4. tabula – Autentifikācijas funkcijas apraksts

Identifikators	CEM_AU_F_01
Apraksts	
	Funkcija atpazīst reģistrēto lietotāju, ja autentifikācija ir veiksmīga, dod piekļuvi sistēmai.
Lietotāju grupas	
	Reģistrēts lietotājs
Ievaddati	
	E-pasts kā lietotājvārds, parole
Apstrāde	
	Datu pārbaude, ja dati ir pareizi, lietotājs saņem piekļuvi sistēmai
Izvaddati	
	Lietotājs saņem skatu atkarībā no viņa lomas
Kļūdu paziņojumi	
	Ja autentifikācija neizdevās lietotājs saņem ziņojumus, kas izraisa kļūdu

2.3.2 Lietotāja izveidošanas funkcija

2.5. tabula – Lietotāja izveidošanas funkcijas apraksts

Identifikators	CEM_AB_F_01
Apraksts	
	Funkcija ir paredzēta, lai veidotu jaunus sistēmas lietotājus.
Lietotāju grupas	
	Administrators, Boss
Ievaddati	
	Lietotājvārds, parole, e-pasts, loma
Apstrāde	
	Datu pārbaude, ja tie ir korekti tiek izveidots jauns lietotājs
Izvaddati	
	Paziņojums, ka lietotājs ir izveidots
Kļūdu paziņojumi	
	Visiem laukiem jābūt aizpildītiem
	Lauki tiek aizpildīti nepareizi
	Lietotājs ar šo e-pasta adresi jau eksistē

2.3.3Lietotāju dzēšanas funkcija

2.6. tabula – Lietotāju dzēšanas funkcija apraksts

Identifikators	CEM_AG_F_01
Apraksts	
	Funkcija paredzēta lietotāju dzēšanai.
Lietotāju grupas	
	Administrators
Ievaddati	
	Lietotāju saraksts
Apstrāde	
	Izmantojot lietotāju sarakstu, atrod vajadzīgo lietotāju, noklikšķina uz pogu "Dzēst lietotāju" pēc ieraksta saistītā objekta tiek dzēsts administratora izvēlētais lietotājs
Izvaddati	
	Funkcijas izpildes apstiprināšana
Kļūdu paziņojumi	

2.3.4Paroles maiņas funkcija

2.7. tabula – Paroles maiņas funkcijas apraksts

Identifikators	CEM_AU_F_02
Apraksts	
	Funkcija ļauj mainīt paroli savā profilā
Lietotāju grupas	
	Reģistrēts lietotājs
Ievaddati	
	Vecā parole, jaunā parole, atkārtoti ievadīta jauna parole.
Apstrāde	
	Veco paroli pārbaude, pārbaude, vai parole atbilst tās izveides prasībām, pārbaude ka jauna parole nesakrīt ar veco, pārbaude atbilst jaunās paroles rindai
Izvaddati	
	Funkcijas izpildes apstiprināšana
Kļūdu paziņojumi	
	Jauna parole neatbilst drošības prasībām
	Nepareiza vecā parole
	Jaunā parole sakrīt ar veco paroli
	Jaunās paroles apstiprinājuma lauks neatbilst jaunās paroles laukam

2.3.5 Lietotāja datu maiņas funkcija

2.7. tabula – Lietotāja datu maiņas

Identifikators	CEM_AB_F_02
Apraksts	
Funkcija ļauj veikt izmaiņas lietotāja informācijā	
Lietotāju grupas	
Administrator , Boss	
Ievaddati	
Lietotāja ID, tā loma, uzvārds, emails	
Apstrāde	
Lietotājs ievada datus, kurus vēlaties mainīt datu rediģēšanas formā. Tālāk notiek ievadīto datu pārbaude.	
Ja dati ir ievadīti pareizi, dati tiek ierakstīti datu bāzē pēc tam, kad tiek atjaunināti interfeisā.	
Dažām lomām izmaiņu veikšanas forma var atšķirties. Administratora lomu nevar piešķirt Bosam.	
Izvaddati	
C	
Kļūdu paziņojumi	
Nav piekļuves, lai veiktu izmaiņas	
Kļūda lauku aizpildīšanā laika	
Kļūdu uzrādīšana, mēģinot mainīt e-pastu, ja e-pastu izmanto cits lietotājs	

2.3.6 Paroles atjaunošanas funkcija

2.8. tabula – Paroles atjaunošanas funkcija apraksts

Identifikators	CEM_AU_F_03
Apraksts	
Funkcija ir paredzēta, lai piekļūtu sistēmai, ja tiek zaudēta parole	
Lietotāju grupas	
Reģistrēts lietotājs	
Ievaddati	
E-pasts	
Apstrāde	
Izmantojot ievadīto emailu, tiek pievienota jauna automātiski ģenerēta parole vienreizējai autorizācijai pēc autorizācijas lietotājam tiks piedāvāts mainīt paroli	
Izvaddati	
Funkcijas izpildes apstiprināšana, vēstulē uz emailu ar paroli	
Kļūdu paziņojumi	
Lietotājs ar šādu emailu neeksistē	

2.3.7 Notikumu pievienošanas funkcija

2.9. tabula – Notikumu pievienošanas funkcija

Identifikators	CEM_AU_F_04
Apraksts	
Funkcija ļauj izveidot notikumus	
Lietotāju grupas	
Reģistrēts lietotājs	
Ievaddati	
Notikuma apraksts	
Apstrāde	
Datu pārbaude, ja dati ir pareizi, pievieno eventu sistēmai	
Izvaddati	
Funkcijas izpildes apstiprināšana, events izveidots	
Kļūdu paziņojumi	
Visiem laukiem jābūt aizpildītiem	
Lauki ir aizpildīti nepareizi	

2.3.8 Notikumu saraksta atspoguļošanās funkcija

2.10. tabula – Notikumu saraksta atspoguļošanās funkcija

Identifikators	CEM_AU_F_05
Apraksts	
Funkcija atgriež lietotājam notikumu sarakstu	
Lietotāju grupas	
Reģistrēts lietotājs	
Ievaddati	
Filtri notikumu sarakstam	
Apstrāde	
Atbilstoši ievadītajiem atlasē notikuma datiem tiek parādīti pieejamie notikumi	
Izvaddati	
Parādīti pieejamie notikumi	
Kļūdu paziņojumi	

2.3.9 Komandas pievienošanas funkcija

211. tabula – Komandas pievienošanas funkcija

Identifikators	CEM_AG_F_02
Apraksts	
Funkcija ļauj izveidot komandus	
Lietotāju grupas	
Administrators, Boss	
Ievaddati	
Komandas apraksts	
Apstrāde	
Datu pārbaude, ja dati ir pareizi, pievieno komandu sistēmai	
Izvaddati	
Funkcijas izpildes apstiprināšana, komanda izveidota	
Kļūdu paziņojumi	
Visiem laukiem jābūt aizpildītiem	
Lauki ir aizpildīti nepareizi	

2.3.10 Komandas saraksta atspoguļošanās funkcija

2.11. tabula – Komandas saraksta atspoguļošanās funkcija

Identifikators	CEM_AU_F_06
Apraksts	
Funkcija atgriež lietotājam notikumu sarakstu	
Lietotāju grupas	
Reģistrēts lietotājs	
Ievaddati	
Filtri notikumu sarakstam	
Apstrāde	
Atbilstoši ievadītajiem atlasē notikuma datiem tiek parādīti pieejamie notikumi	
Izvaddati	
Parādīti pieejamie notikumi	
Kļūdu paziņojumi	

2.3.11 Biroja pievienošanas funkcija

2.12. tabula – Biroja pievienošanas funkcija

Identifikators	CEM_AG_F_03
Apraksts	
Funkcija ļauj izveidot biroju	
Lietotāju grupas	
Administrators, Boss	
Ievaddati	
Biroja apraksts	
Apstrāde	
Datu pārbaude, ja dati ir pareizi, pievieno biroja sistēmai	
Izvaddati	
Funkcijas izpildes apstiprināšana, biroja izveidota	
Kļūdu paziņojumi	
Visiem laukiem jābūt aizpildītiem	
Lauki ir aizpildīti nepareizi	

2.3.12 Biroja saraksta atspoguļošanās funkcija

2.13. tabula – Biroja saraksta atspoguļošanās funkcija

Identifikators	CEM_AU_F_07
Apraksts	
Funkcija atgriež lietotājam biroju sarakstu	
Lietotāju grupas	
Reģistrēts lietotājs	
Ievaddati	
Filtri biroju sarakstu	
Apstrāde	
Atbilstoši ievadītajiem atlasē notikuma datiem tiek parādīti pieejamie biroji	
Izvaddati	
Parādīti pieejamie biroji	
Kļūdu paziņojumi	

2.3.13 Lietotāj saraksta atspoguļošanās funkcija

2.14. tabula – Lietotāj saraksta atspoguļošanās funkcija

Identifikators	CEM_AU_F_08
Apraksts	
Funkcija atgriež lietotājam lietotāju sarakstu	
Lietotāju grupas	
Reģistrēts lietotājs	
Ievaddati	
Filtri lietotāju sarakstam	
Apstrāde	
Atbilstoši ievadītajiem atlasē notikuma datiem tiek parādīti pieejamie notikumi	
Izvaddati	
Tiek rādīti lietotāji, pamatojoties uz filtriem	
Kļūdu paziņojumi	

2.3.14 Lietotāja profilā informācijas atspoguļošanās funkcija

2.15. tabula – Informācija lietotāja profilā

Identifikators	CEM_AU_F_09
Apraksts	
Funkcija atgriež lietotājam informācija par savu profilu	
Lietotāju grupas	
Reģistrēts lietotājs	
Ievaddati	
Reģistrējoties izmantotie dati	
Apstrāde	
Datu ielāde saskaņā ar lietotāja ID	
Izvaddati	
Izmantotā lietotāja profila dati	
Kļūdu paziņojumi	

2.3.15 Cita lietotāja profila informācija atspoguļošanās funkcija

2.16. tabula – Cita lietotāja profila informācija

Identifikators	CEM_AU_F_10
Apraksts	
Funkcija atgriež lietotājam informācija par cita lietotāja profilu	
Lietotāju grupas	
Reģistrēts lietotājs	
Ievaddati	
Profils, kas tiks skatīts	
Apstrāde	
Datu ielāde saskaņā ar lietotāja ID	
Izvaddati	
Izmantotā lietotāja profila dati	
Kļūdu paziņojumi	

2.3.16 Biroju informācijas atspoguļošanās funkcija

2.17. tabula – Informācija par biroju

Identifikators	CEM_AU_F_11
Apraksts	
Funkcija atgriež lietotājam informācija par izvēlēto biroju	
Lietotāju grupas	
Reģistrēts lietotājs	
Ievaddati	
Dati, kas tika aizpildīti, izveidojot biroju	
Apstrāde	
Datu ielāde saskaņā ar biroju ID	
Izvaddati	
Informācija par biroju <input type="checkbox"/>	
Kļūdu paziņojumi	

2.3.17 Komandu informācijas atspoguļošanās

2.18. tabula – Informācija par komandu

Identifikators	CEM_AU_F_12
Apraksts	
Funkcija atgriež lietotājam informācija par izvēlēto komandu	
Lietotāju grupas	
Reģistrēts lietotājs	
Ievaddati	
Dati, kas tika aizpildīti, izveidojot komandu	
Apstrāde	
Datu ielāde saskaņā ar komandas ID	
Izvaddati	
Informācija par komandu <input type="checkbox"/>	
Kļūdu paziņojumi	

2.3.18 Notikumu informācijas atspoguļošanās funkcija

2.20. tabula – Informācija par notikumu

Identifikators	CEM_AU_F_13
Apraksts	
Funkcija atgriež lietotājam informācija par izvēlēto notikumu	
Lietotāju grupas	
Reģistrēts lietotājs	
Ievaddati	
Dati, kas tika aizpildīti, izveidojot notikumu	
Apstrāde	
Datu ielāde saskaņā ar notikumu ID	
Izvaddati	
Informācija par izvēlēto notikumu <input type="checkbox"/>	
Kļūdu paziņojumi	

2.4. Ievadlauku ierobežojumi.

2.4.1 . tabula. Ievadlauku ierobežojumi.

Lauka nosaukums	Ierobežojumi <input type="checkbox"/>
Lietotāja vārds	1-30 simboli
Lietotāja uzvārds	Nesatur simbolus : !,@,#,\$%,^,&,* utt. Nesatur ciparus
E-pasts	5-100 simboli Jābūt @ Nesatur simbolus : !,#,\$%,^,&,* utt.
Parole	8-30 simboli Jāsatur vismaz 1 cipars Jāsatur vismaz 1 liels burts Jāsatur vismaz 1 simbols : ! ,<,>?,@,#,\$%,^,&,* utt.
Vecā parole	Jāsatur pašlaik aktīva parole
Jaunā parole	Nav vienāds ar veco paroli
Jaunā paroles apstiprinājums	Vienāds ar jaunu paroli
Notikumu nosaukums	1-30 simboli
Notikumu apraksts	10-190 simboli
Notikumu sākuma datums	Nav (standartizēta ievade, veicot datumu izvēli)
Notikumu pabeigšanas datums	Nav (standartizēta ievade, veicot datumu izvēli)
Biroja nosaukums	1-50 simboli
Valsts, kurā atrodas birojs	1-60 simboli Nesatur simbolus : !,@,#,\$%,^,&,* utt. Nesatur ciparus
Pilsēta, kurā atrodas birojs	1-60 simboli Nesatur simbolus : !,@,#,\$%,^,&,* utt. Nesatur ciparus
Pilsēta, kurā atrodas birojs	1-60 simboli Nesatur simbolus : !,@,#,\$%,^,&,* utt.
Komandas nosaukums	1-50 simboli
Komandas apraksts	10-190 simboli
Notikumu tips	1-30 simboli Nesatur simbolus : !,@,#,\$%,^,&,* utt.
Lietotāja loma	1-30 simboli Nesatur simbolus : !,@,#,\$%,^,&,* utt.

2.5. Nefunkcionālās prasības

2.5.1. Veiktspējas prasības

- Sistēmai jānodrošina paralēla aktivitāšu (eventu) lejupielādi ar intervālu lietotāja interfeisā, kas nepārsniedz 5 sekundes.
- Jānodrošina sistēmas darbību vismaz 5000 lietotājiem paralēli.
- Prasības neattiecas uz tīkla veiktspēju.

2.5.2. Ērtības un vienkāršība

- 95% jauno lietotāju vajadzētu izmantot sistēmas funkcionalitāti profilā bez trešās puses palīdzības.
- Sistēmai ir jāpielāgojas dažādiem ierīces izmēriem.
- Jābūt pieejamai starptautiskai valodai (angļu valodā).
- Ievades laukiem ir jābūt norādījumiem par ievades iespējām. Lietojumprogrammas dizainam jābūt vizuāli cik vien iespējams vienkāršam, kas lietotājam ļaus to izmantot pēc iespējas koncentrētāk.
- Sistēmai jānodrošina lietotājam iespēja labot radītos notikumus.

2.5.3. Drošības prasības

- Piekļuve sistēmai jāietver parole, neregistrētu lietotāju piekļuve sistēmai nav iespējama.
- Piekļuve ar nepareizi ievadītiem autorizācijas datiem nav iespējama.
- Dati no citiem birojiem uz komandām nedrīkst būt redzami personām, kas nav iekļautas attiecīgajos biroja un komandās.
- Piekļuve rediģēšanas lietotājiem ir pieejama tikai administratoram.
- Pasākumu izveidei vajadzētu būt tikai pieejamībai, pamatojoties uz uzņēmuma lomu.
- Parolēm jābūt šifrētām.

2.5.4. Pieejamības prasības

- Sistēmas atjauninājumi jāveic laikā no 02:00 līdz 05:00, jo šajā laikā ir paredzēta vismazākā lietotāju aktivitāte.
- Sistēmai jābūt pieejamai 93% laika (sistēma var nebūt pieejama 11.2 stundas mēnesī) .
- Sistēmai jāstrādā un pareizi jāparādās “Google Chrome”, “Opera”, “MozillaFirefox,” un “Microsoft Edge” pārlūkprogrammām.

2.5.5. Uzturēšanas prasības

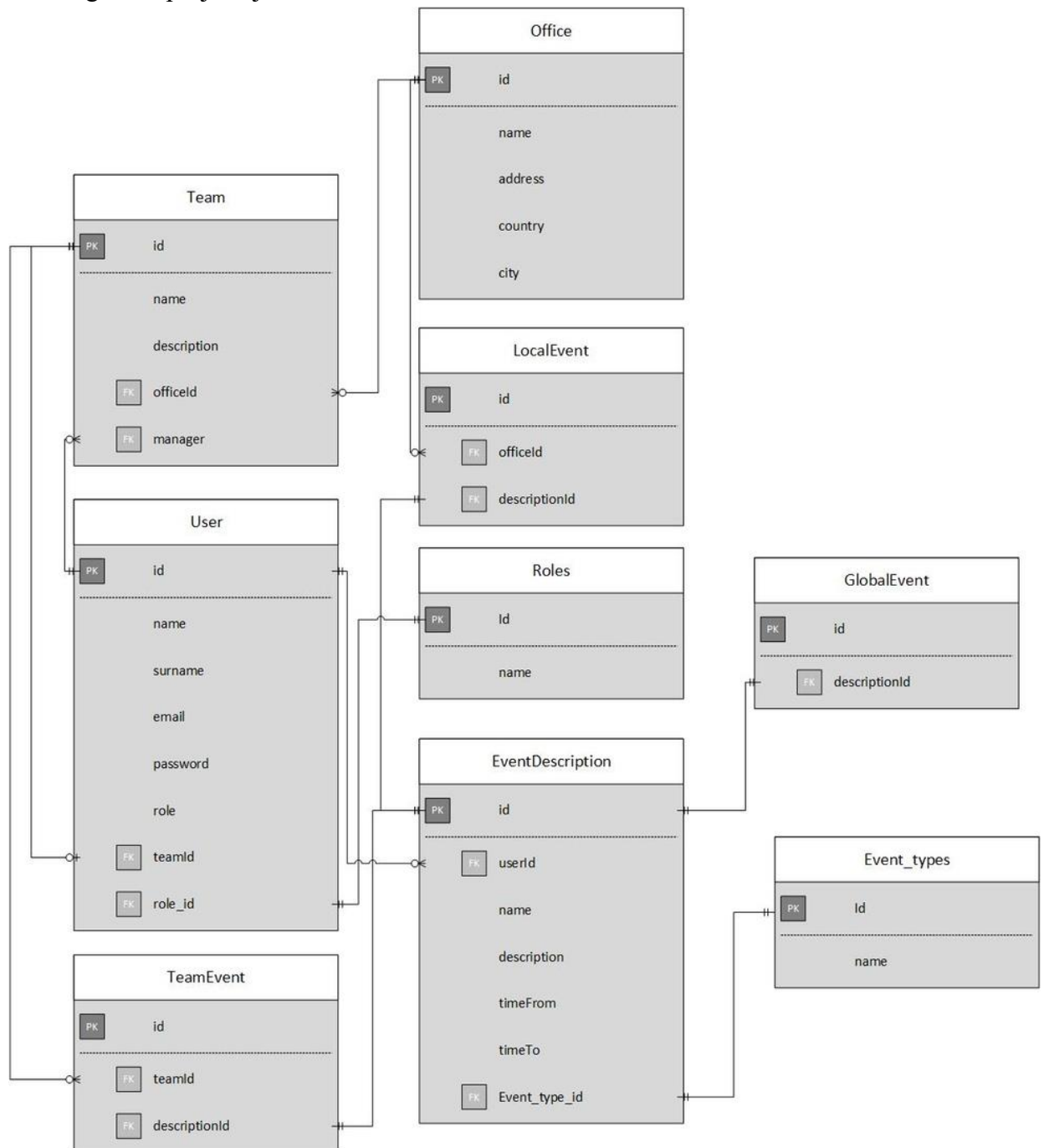
- Sistēma ir jāatjaunina vismaz divas reizes mēnesī.
- Sistēmai jāveido rezerves kopija divas reizes mēnesī.

2.5.6. Uzticamības prasības

- Gadījumā, ja tiek pārtraukta servera darbs, dati nevar tikt pazaudēti.

3 PROGRAMMATŪRAS PROJEKTĒJUMA APRAKSTS

3.1. Loģiskais projektējums



3.1. att. Datu bāzes loģiskais modelis

Attēlā 3.1. redzams datu bāzes loģiskais modelis. Primārās atslēgas (primary key) atzīmētas (PK), ārējās atslēgas (foreign key) atzīmētas (FK).

3.1.1 Pāreja no konceptuāla uz loģisko modeli

Veicot pāreju no konceptuālā datu bāzes modeļa (attēls 2.1) uz loģisko datu bāzes (attēls 3.1) modeli attēlotajām entītijām (User, Team, Office, EventDescription, GlobalEvent, LocalEvent, TeamEvent) ir izveidotas attiecīgās tabulas ar attiecībām 1 pret 0-N vai 1:1 un norādītas to primārās atslēgas.

Vārnu kāju apzīmējumu paskaidrojums:

- User – Team ir N:1 attiecība un, iespējams 0, nulles vērtība būs gadījumā, ja lietotājs nav pievienots komandai.
- User – Roles ir 1:1 attiecība lietotājam var būt viena un tikai viena loma.
- TeamEvent būtība satur komandu un pasākuma aprakstu. Komandai var nebūt notikumu, taču katram pasākumam ir viena un tikai viena komanda arī katram notikumam ir tikai viens apraksts.
- User – EventDescription ir 1:N attiecība un, iespējams 0. Katrs notikuma apraksts ir unikāls, tāpēc pasākuma aprakstam ir tikai viens lietotājs, kas to izveidoja, arī lietotājam var nebūt izveidoto eventu, šajā gadījumā apraksts būs tukšs attiecībā pret lietotāju.
- GlobalEvent - katram pasākumam ir viens un tikai viens apraksts.
- Office – Team ir 1:N attiecība un, iespējams, 0 komanda vienmēr ir saistīta ar biroju, birojam var būt daudz komandu, arī birojam var nebūt komandas.
- Office – LocalEvent 1:N attiecība un, iespējams, 0. Birojā var būt daudz pasākumu, kā arī to skaits var būt 0, bet katram vietējam pasākumam ir birojs / biroji, kuriem tas ir pievienots.
- LocalEvent – EventDescription, katram vietējam notikumam ir viens un tikai viens apraksts.
- EventDescription - ir 1:1 attiecība, jo vienam notikuma aprakstam var piešķirt vienu un tikai vienu lomu.

3.1.2. Datu bāzes fiziskais modelis

Pāreja no loģiskās datu bāzes modeļa (3.1. Attēls) uz fizisku datu bāzi.

Fiziskais modelis parāda ģenerēto datu kopu un tā elementus: varchar, bigint, fk, pk,timestamp, datetime. Kas ir parādīts sadaļas (3.1.2.1) tabulās Datu bāzes tabulu apraksta.

3.1.2.1 Datu bāzes tabulu apraksts

3.1.2.1.1. Tabula. Tabulas User apraksts.

Tabulā Usertiek glabāti usera dati. Tabula sastāv no biroja identifikācijas numura, vārds, uzvārds, parole, emails, emaila apstiprināšanas laiks, drošības marķieris (token for cookie), lietotāja izveides datums, lietotāja pēdējās atjaunināšanas datums. Ir divas ārējās atslēgas (role_id), (team_id)) kas savieno Usertabulu (skat. 3.1.2.1.1.) Team tabulu (skat. 3.1.2.1.3. Tabulu) un Roles tabulu(skat. 3.1.2.1.8).

Tabulā 3.1.2.1.1. glabā User kolonnas nosaukumus, to datu tipus, apraksts un citu informāciju (PK, FK, NOT NULL).

3.1.2.1.1tabula. Tabulas User apraksts.

Nosaukums	Datu tips	Papild informācija	Apraksts
ID	Bigint(20)	NOT NULL,PRIMARY KEY	Lietotāja indentifikācijas numurs
Name	Varchar(191)	NOT NULL	Lietotāja vārds
Surname	Varchar(191)	NOT NULL	Lietotāja uzvārds
Password	Varchar(191)	NOT NULL	Lietotāja parole
Email	Varchar(191)	NOT NULL	Lietotāja e-pasts
email_verified_at	Timestamp	NULLABLE	ģenerēts frameworka
remember_token	Varchar(100)	NOT NULL	ģenerēts frameworka
created_at	Timestamp	NULLABLE	ģenerēts frameworka
updated_at	Timestamp	NULLABLE	ģenerēts frameworka
role_id	Bigint(20)	FOREIGN KEY	Savienojums ar tabulu Roles
team_id	Bigint(20)	FOREIGN KEY	Savienojums ar tabulu Team

3.1.2.1.2. tabula. Tabulas Team Event apraksts.

Tabulā TeamEvent tiek glabāti komandas eventa dati. Tabula sastāv no personīga identifikācijas numura, šim objektam (TeamEvent) ir divas ārējās atslēgas (descriptionid),(team_id) kas savieno TeamEvent tabulu (skat. 3.1.2.1.2.) ar EventDescription tabulu (skat. 3.1.2.1.6. Tabulu) un Team tabulu (skat. 3.1.2.1.3. Tabulu).

Tabulā 3.1.2.1.2. ir tabulas GlobalEvent kolonnu nosaukumi, to datu tipi, apraksts un cita informācija (PK, FK, NOT NULL).

3.1.2.1.2 tabula. Tabulas TeamEvent apraksts.

Nosaukums	Datu tips	Papild informācija	Apraksts
ID	Bigint(20)	NOT NULL, PRIMARY KEY	TeamEvent identifikācijas numurs
created_at	Timestamp	NULLABLE	ģenerēts frameworka
descriptionid	Timestamp	NULLABLE	ģenerēts frameworka
team_id	Bigint(20)	NOT NULL, FOREIGN KEY	Savienojums ar Komandu
description_id	Bigint(20)	NOT NULL, FOREIGN KEY	Savienojums ar eventa aprakstu

3.1.2.1.3. tabula. Tabulas Team apraksts.

Tabulā Team tiek glabāti komandas dati. Tabula sastāv no biroja identifikācijas numura, izveides datums, pēdējo komandas datu izmaiņu datums, biroja nosaukums, apraksts. Ir divas ārējās atslēgas (office_id), (manager_id) kas savieno ar Office tabulu (skat. 3.1.2.1.5.) un User tabulu (skat. 3.1.2.1.1)

Tabulā 3.1.2.1.3. glabā Team kolonnas nosaukumus, to datu tipus, apraksts un citu informāciju (PK, FK, NOT NULL).

3.1.2.1.3 tabula. Tabulas Team apraksts.

Nosaukums	Datu tips	Papild informācija	Apraksts
ID	Bigint(20)	NOT NULL, PRIMARY KEY	Identifikācijas numurs
created_at	Timestamp	NULLABLE	ģenerēts frameworka
updated_at	Timestamp	NULLABLE	ģenerēts frameworka
Name	Varchar(191)	NOT NULL	Lietotāja vārds
description	Varchar(191)	NOT NULL	Lietotāja eventa apraksts
office_id	Bigint(20)	FOREIGN KEY	Savienojums ar tabulu Office
manager_id	Bigint(20)	FOREIGN KEY	Savienojums ar User

3.1.2.1.4. tabula. Tabulas Office apraksts.

Tabulā Office tiek glabāti biroja dati. Tabula sastāv no biroja identifikācijas numura, izveides datums, pēdējo birojā izmaiņu datums, biroja nosaukums, adrese, pilsēta kur atrodas birojs un valsts.

Tabulā 3.1.2.1.4. glabā Office kolonnas nosaukumus, to datu tipus, apraksts un citu informāciju (PK, FK, NOT NULL).

3.1.2.1.4 tabula. Tabulas Office apraksts.

Nosaukums	Datu tips	Papild informācija	Apraksts
ID	Bigint(20)	NOT NULL,PRIMARY KEY	Office - Indentifikācijas numurs
created_at	Timestamp	NULLABLE	ģenerēts frameworka
updated_at	Timestamp	NULLABLE	ģenerēts frameworka
Name	Varchar(191)	NOT NULL	Biroja nosaukums
Address	Varchar(191)	NOT NULL	Biroja adrese
country	Varchar(191)	NOT NULL	Valsts nosaukums
city	Varchar(191)	NOT NULL	Pilsētas nosaukums

3.1.2.1.5. tabula. Tabulas GlobalEvent apraksts.

Tabulā GlobalEvent tiek glabāti globālie notikumu dati. Tabula sastāv no personīga identifikācijas numura, šim objektam (GlobalEvent) ir viena ārējā atslēga (descriptionid), kas savieno GlobalEvent tabulu (skat. 3.1.2.1.5.) ar EventDescription tabulu (skat. 3.1.2.1.6. Tabulu).

Tabulā 3.1.2.1.5. ir tabulas GlobalEvent kolonnu nosaukumi, to datu tipi, apraksts un cita informācija (PK, FK, NOT NULL).

3.1.2.1.5 tabula. Tabulas GlobalEvent apraksts.

Nosaukums	Datu tips	Papild informācija	Apraksts
ID	Bigint(20)	NOT NULL,PRIMARY KEY	Indentifikācijas numurs
created_at	Timestamp	NULLABLE	ģenerēts frameworka
updated_at	Timestamp	NULLABLE	ģenerēts frameworka
description_id	Varchar(191)	NOT NULL	Apraksta indentifikācijas numurs

3.1.2.1.6. tabula. Tabulas EventDescription apraksts.

Tabulā EventDescription tiek glabāti primārie dati, lai izveidotu aktivitāti sistēmas ietvaros. Tabula sastāv no personīga identifikācijas numura, tabulai ir divas ārējās atslēgas (user_id),(event_type_id)kas kā savieno EventDescription tabulu (skat. 3.1.2.1.6.) ar tabulu User (skat. 3.1.2.1.1. Tabulu) un EventTypes(skat. 3.1.2.1.9. Tabulu). Tabulā 3.1.2.1.6. ir tabulas EventDescription kolonnu nosaukumi, to datu tipi, apraksts un cita informācija (PK, FK, NOT NULL).

3.1.2.1.6 tabula. Tabulas GlobalEvent apraksts.

Nosaukums	Datu tips	Papild informācija	Apraksts
ID	Bigint	NOT NULL,PRIMARY KEY	Apraksta indentifikācijas numurs
created_at	Timestamp	NULLABLE	ģenerēts frameworka
updated_at	Timestamp	NULLABLE	ģenerēts frameworka
user_id	Bigint	FOREIGN KEY	Lietotāja indentifikācijas numurs
Name	Varchar(191)	NOT NULL	Eventa nosaukums
description	Varchar(191)	NOT NULL	Eventa apraksts
timeFrom	DateTime	NOT NULL	Sākuma datums
timeTo	DateTime	NOT NULL	Nobeiguma datums
event_type_id	Bigint	FOREIGN KEY	Lietotāja indentifikācijas numurs

3.1.2.1.7. tabula. Tabulas LocalEvent apraksts.

Tabulā LocalEvent tiek glabāti lokalo notikumu dati. Tabula sastāv no personīga identifikācijas numura, šim objektam (LocalEvent) ir divas ārējā atslēgas (description_id), (office_id), kas savieno ar tabulāmEventDescription (skat. 3.1.2.1.6. Tabulu) un Office (skat. 3.1.2.1.4.Tabulu) Tabulā 3.1.2.1.7. ir tabulas LocalEvent kolonnu nosaukumi, to datu tipi, apraksts un cita informācija (PK, FK, NOT NULL).

3.1.2.1.7 tabula. Tabulas LocalEvent apraksts

Nosaukums	Datu tips	Papild informācija	Apraksts
ID	Bigint(20)	NOT NULL,PRIMARY KEY	Lokālo eventu Indentifikācijas numurs
created_at	Timestamp	NULLABLE	ģenerēts frameworka
updated_at	Timestamp	NULLABLE	ģenerēts frameworka
office_id	Bigint(20)	FOREIGN KEY	Savienojums ar Office tabulu
description_id	Bigint(20)	FOREIGN KEY	Savienojums ar apraksta tabulu

3.1.2.1.8. tabula. Tabulas Roles apraksts.

Tabulā Roles tiek glabāti lomu dati. Tabula sastāv no lomu identifikācijas numura, nosaukuma. Tabulā 3.1.2.1.8. glabā Roles kolonnas nosaukumus, to datu tipus, apraksts un citu informāciju (PK, FK, NOT NULL).

3.1.2.1.8 tabula. Tabulas Roles apraksts.

Nosaukums	Datu tips	Papild informācija	Apraksts
ID	Bigint(20)	NOT NULL, PRIMARY KEY	Lomu indentifikācijas numurs
name	varchar(191)	NOT NULL	Pieejamās lomas

3.1.2.1.9. tabula. Tabulas Event_types apraksts.

Tabulā Event_types tiek glabāti eventu tipu dati. Tabula sastāv no eventa tipa identifikācijas numura un eventa nosaukuma.

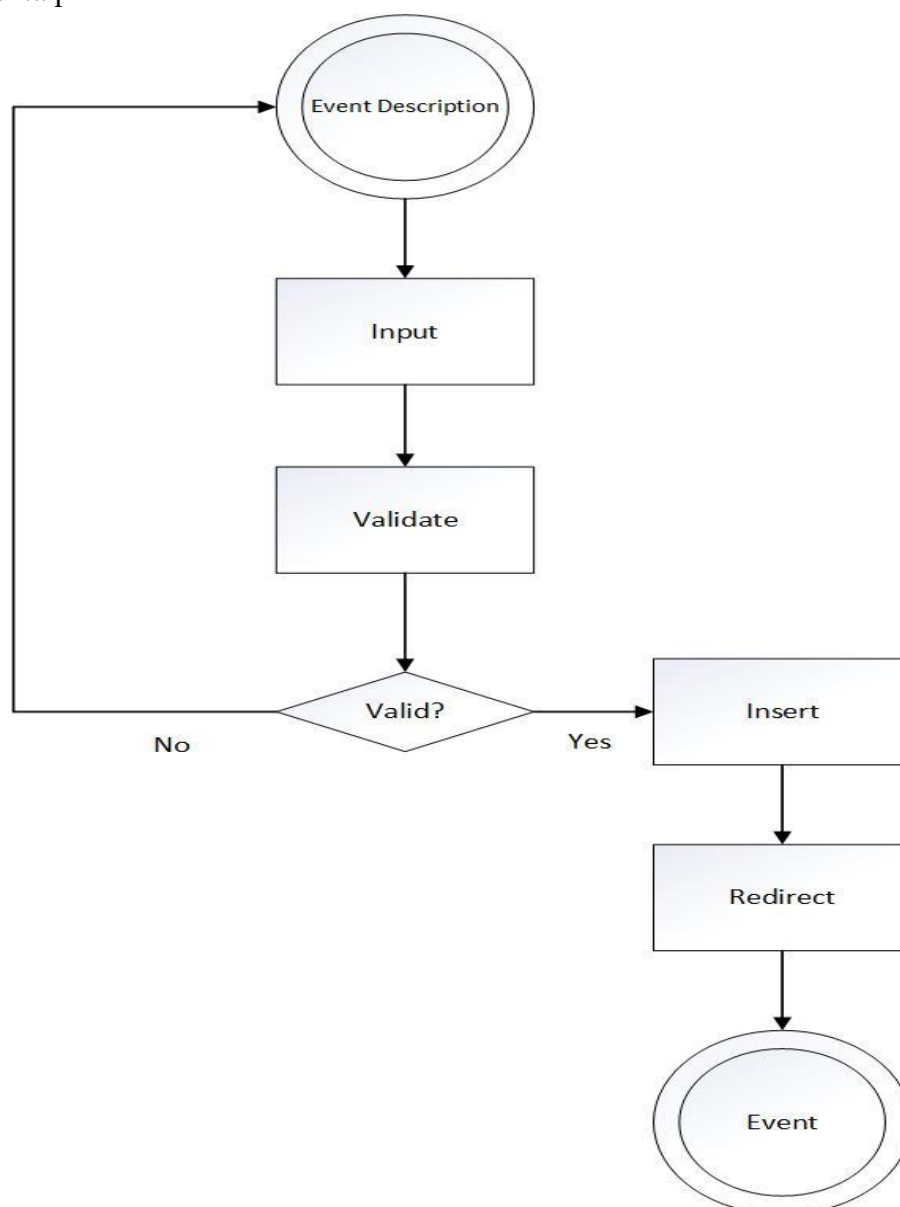
Tabulā 3.1.2.1.9. glabā Even_types kolonnas nosaukumus, to datu tipus, apraksts un citu informāciju (PK, FK, NOT NULL).

3.1.2.1.9 tabula. Tabulas Event_types apraksts.

Nosaukums	Datu tips	Papild informācija	Apraksts
ID	Bigint(20)	NOT NULL, PRIMARY KEY	Lomu indentifikācijas numurs
name	varchar(191)	NOT NULL	Pieejamās lomas

3.2 Daļējs funkciju projektējums

3.2.1 Eventa pievienošana



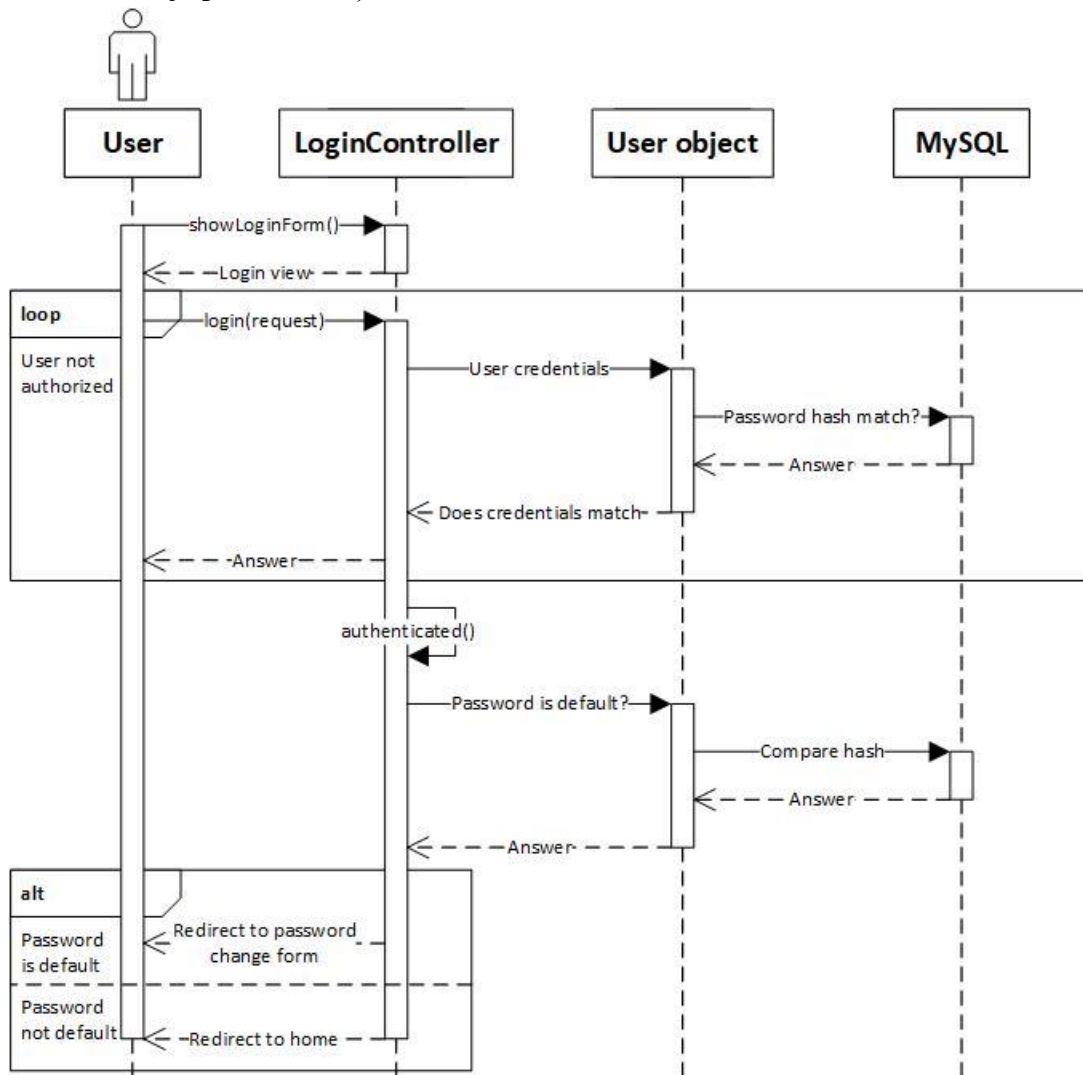
3.1. att. Atrisinājuma pievienošanas aktivitāšu diagramma

Attēlā 3.1. Var redzēt aktivitāšu diagramma eventa pievienošanas momentu.

Eventa pievienošanā:

- Ir redzama eventdescription forma.
- Forma aizpildīšanā.
- Formas fieldus validācija.
- Pēc pārbaudījumā, ja ievadītajā konteksta ir kļūda lietotājs atgriežas pie eventdescription aizpildīšanas formas.
- Ja viss ir korekti aizpildīts tad veidojas aktivitāte (events).

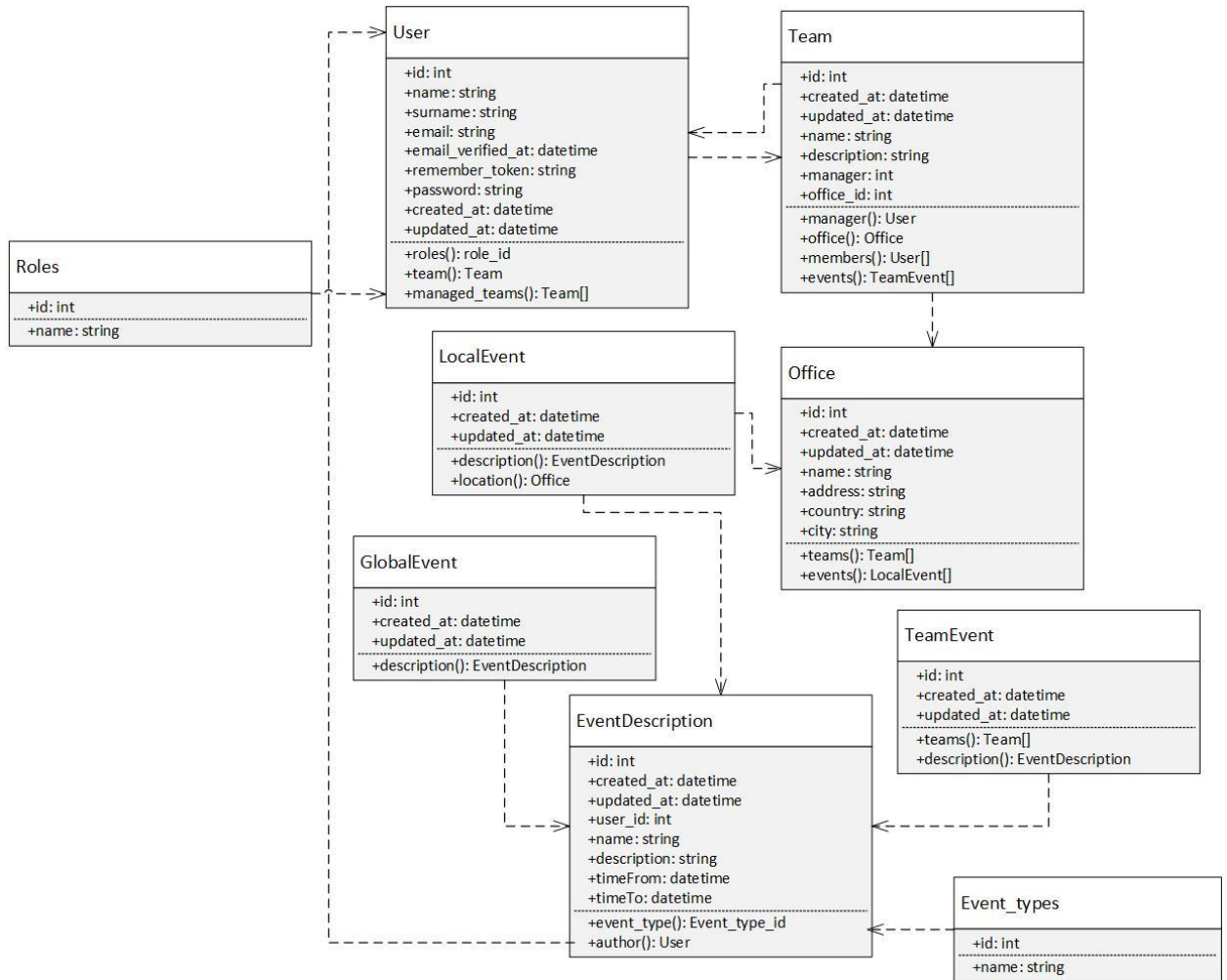
3.2.2 Lietotāja paroles maiņa



3.2. att. Paroles maiņas secību diagramma.

Lietotājs ir autorizācijas formā, ievada savus lietotāja datus (login, parole) notiek logins, kurā tiek pārbaudīta defolta parole (password) pašā tabulas apakšdaļā ir redzami divi notikumu varianti. Pirmais variants – parole sakrīt ar defoltu. Lietotājs pāriet uz paroles reģistrācijas veidlapu. Otrais variants - parole nav defolta, lietotājs saņem piekļuvi.

3.2.3 Sistēmas pamat funkcionalitātes klašu diagramma



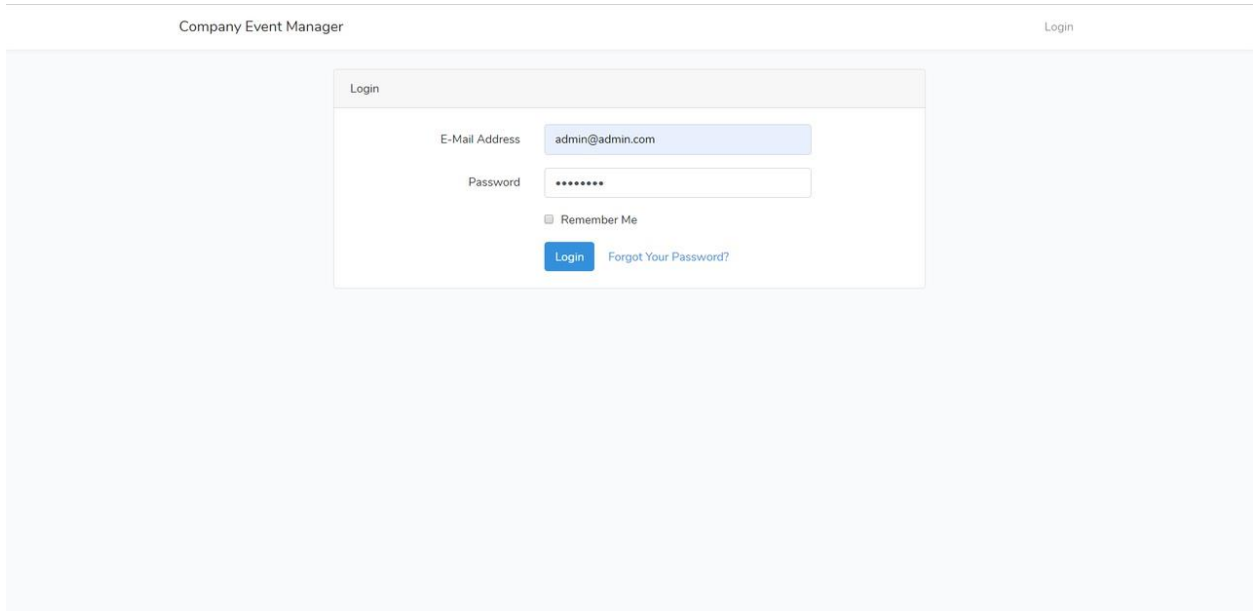
3.3. att. **Klašu diagramma**

Klases diagramma, parāda atkarību starp klasēm TeamEvent, GlobalEvent, LocalEvent, Event_types no EventDescription. Team, LocalEvent no Office. Team, Roles, EventDescription no User. User no Team.

3.3. Skatu projektējums

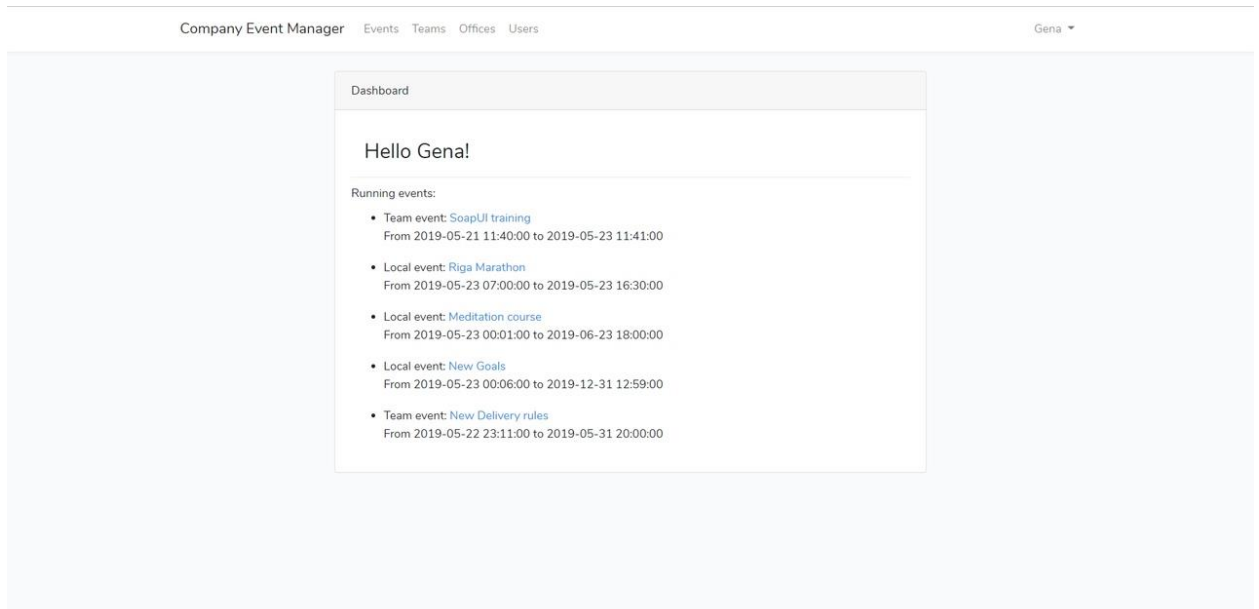
3.3.1 Lietotāja autorizācija.

3.4. att. Lietotāja autorizācija.



3.3.2 Autorizācijas mājas lapa.

3.5. att. Autorizācijas mājas lapa.



3.3.3 Notikumu saraksts.

3.6. att. Notikumu saraksts.

Company Event Manager Events Teams Offices Users Admin

Event list

Create event Per page: 50

#	Name	Running	Type	Activity	From	To	Members		
1	New goals	No	Global	Outdoor activity	2019-05-22 23:36:00	2019-05-22 23:37:00	Everyone	Edit	Delete
2	SoapUI training	Yes	Team	Education	2019-05-21 11:40:00	2019-05-23 11:41:00	Team Automation Team	Edit	Delete
6	Riga Marathon	Yes	Local	Outdoor activity	2019-05-23 07:00:00	2019-05-23 16:30:00	Office RigaOffice	Edit	Delete
7	Meditation course	Yes	Local	Outdoor activity	2019-05-23 00:01:00	2019-06-23 18:00:00	Office RigaOffice	Edit	Delete
8	New Goals	Yes	Local	Outdoor activity	2019-05-23 00:06:00	2019-12-31 12:59:00	Office BerlinOffice	Edit	Delete
9	New Delivery rules	Yes	Team	Work meeting	2019-05-22 23:11:00	2019-05-31 20:00:00	Team Zergs	Edit	Delete

1 / 1

3.3.4 Komandu saraksts.

3.7. att. Komandu saraksts.

Company Event Manager Events Teams Offices Users Admin

Teams

Create team Per page: 50

#	Name	Manager	Office	Members		
1	Legion	Kvintilij Var	BerlinOffice	1	Edit	Delete
2	Zergs	Maksims Pavlenko	RigaOffice	1	Edit	Delete
3	Automation Team	Maksims Pavlenko	BerlinOffice	1	Edit	Delete

1 / 1

3.3.5 Biroju saraksts.

3.8. att. Biroju saraksts.

Company Event Manager Events Teams Offices Users Admin ▾

Offices

[Create office](#) Per page:

#	Name	City	Country	
		<input type="text" value="All"/>	<input type="text" value="All"/>	
2	BerlinOffice	Berlin	Germany	Edit Delete
3	RigaOffice	Riga	Latvia	Edit Delete

[←](#) / 1 [→](#)

3.3.6 Lietotāju saraksts.

3.9. att. Lietotāju saraksts.

Company Event Manager Events Teams Offices Users Admin ▾

Users

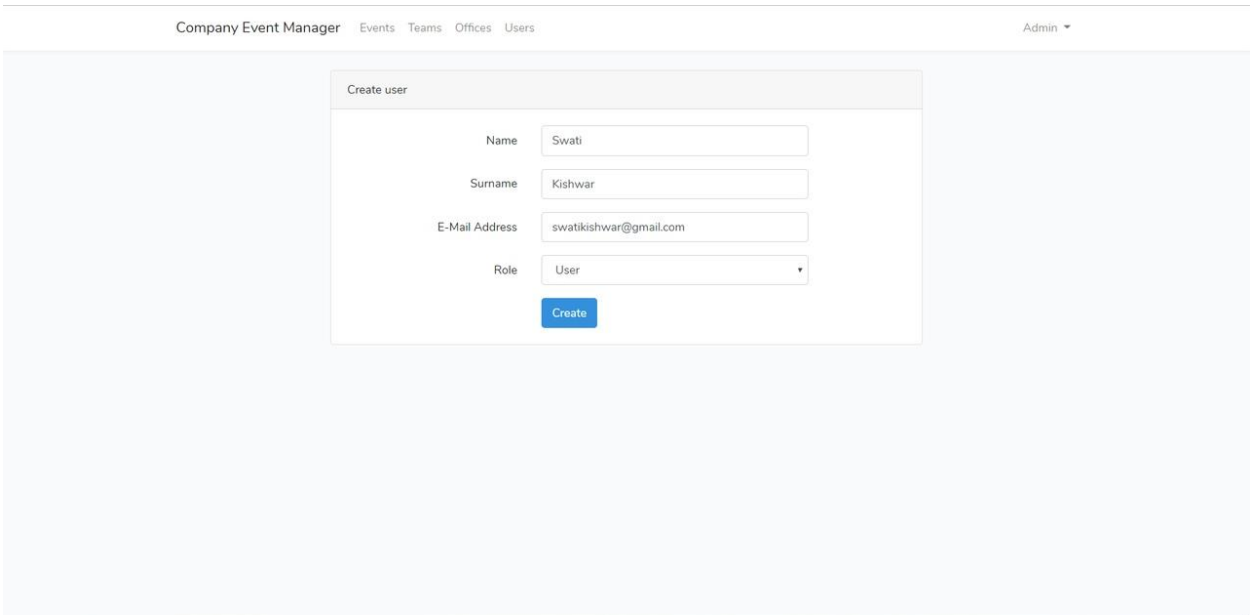
[Create user](#) Per page:

#	Name	Surname	E-mail	Role	Team	Created at	
				<input type="text" value="All"/>	<input type="text" value="All"/>		
2	Albert	Marks	marksalbert@gmail.com	User	Legion	2019-05-22 08:02:03	Edit Delete
3	Kvintilij	Var	kvintilijvar@gmail.com	Manager		2019-05-22 08:02:27	Edit Delete
4	Kishwar	Swati	swatikishwar@gmail.com	User	Zergs	2019-05-22 08:04:02	Edit Delete
5	Maksims	Pavlenko	pavlenkomaxims@gmail.com	Manager		2019-05-22 08:05:00	Edit Delete
6	Gena	Bukin	gena@bukin.com	Boss		2019-05-22 08:05:30	Edit Delete
7	Farukh	Trivedi	farukh@gmail.com	User	Automation Team	2019-05-22 08:31:14	Edit Delete

[←](#) / 1 [→](#)

3.3.7 Lietotāja izveides logs.

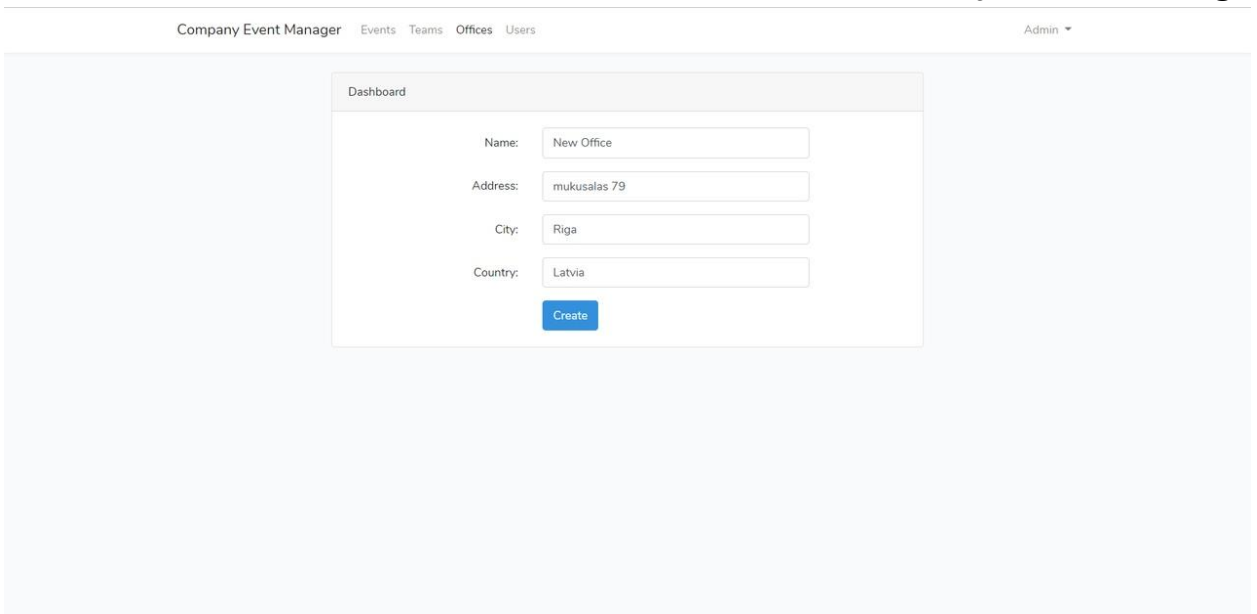
3.10. att. Lietotāja izveides logs.



The screenshot shows the 'Create user' form within the 'Company Event Manager' application. The navigation bar at the top includes 'Company Event Manager', 'Events', 'Teams', 'Offices', 'Users', and 'Admin'. The form itself is titled 'Create user' and contains the following fields: 'Name' with the value 'Swati', 'Surname' with 'Kishwar', 'E-Mail Address' with 'swatikishwar@gmail.com', and 'Role' with a dropdown menu set to 'User'. A blue 'Create' button is positioned at the bottom of the form.

3.3.8 Biroja izveidošanas logs.

3.11. att. Biroja izveidošanas logs.



The screenshot displays the 'Dashboard' form for creating a new office. The navigation bar at the top shows 'Company Event Manager', 'Events', 'Teams', 'Offices', 'Users', and 'Admin'. The form is titled 'Dashboard' and includes the following fields: 'Name' with 'New Office', 'Address' with 'mukusalas 79', 'City' with 'Riga', and 'Country' with 'Latvia'. A blue 'Create' button is located at the bottom of the form.

3.3.9 Notikumu izveides logs.

3.12. att. Notikumu izveides logs.

Company Event Manager Events Teams Offices Users Admin ▾

Create event

Name:

Description:

Activity type:

Time from:

Time to:

Event for:

Team:

3.3.10 Komandu izveides logs.

3.13. att. Komandu izveides logs.

Company Event Manager Events Teams Offices Users Admin ▾

Edit team #5

Name:

Description:

Office:

Manager:

4 Pielikums.

4.1 Testēšanas dokumentācija

4.1.0 Lietotāju autorizācijas dati(Test roles).

User Role	E-Mail Address	Password
Administrator	admin@admin.com	1q2w3e4r
Boss	gena@bukin.com	1q2w3e4r
Manager	kvintilijvar@gmail.com	1q2w3e4r
User	swatikishwar@gmail.com	1q2w3e4r

4.1.1 Autorizācijastests.(login test).

Step Name	Description	Expected Result
Step 1	Ielogoties kā Administrators	veiksmīga autorizācija
Step 2	Ielogoties kā Boss	veiksmīga autorizācija
Step 3	Ielogoties kā Manager	veiksmīga autorizācija
Step 4	Ielogoties kā User	veiksmīga autorizācija

4.1.2 Lietotāja veidošanas tests. (create user test).

Step Name	Description	Expected Result
Step 1	Ielogoties kā Administrators	Veiksmīga autorizācija
Step 2	Navigācija uz Users	Users sarakstu skats ir atvērts
Step 3	Izveidjiet lietotāju (Aizpildiet visas rindas)	Lietotājs ir izveidots
Step 4	Navigācija uz Users	Izveidots lietotājs eksistē
Step 5	Ielogoties kā Boss	Veiksmīga autorizācija
Step 6	Navigācija uz Users	Users sarakstu skats ir atvērts
Step 7	Izveidjiet lietotāju (Aizpildiet visas rindas) (nav varianta izveidot lietotāju "boss")	Lietotājs ir izveidots
Step 8	Navigācija uz Users	Izveidots lietotājs eksistē

4.1.3 Biroju veidošanas tests. (create office test).

Step Name	Description	Expected Result
Step 1	Ielogoties kā Administrators	Veiksmīga autorizācija
Step 2	Navigācija uz Offices	Offices sarakstu skats ir atvērts
Step 3	Izveidojiet biroju (Aizpildiet visas rindas)	Birojs ir izveidots
Step 4	Navigācija uz Offices	Izveidots birojs eksistē
Step 5	Ielogoties kā Boss	Veiksmīga autorizācija
Step 6	Navigācija uz Offices	Offices sarakstu skats ir atvērts
Step 7	Pārbaudīt biroju no step 4	Birojs eksistē
Step 8	Izveidojiet biroju (Aizpildiet visas rindas)	Birojs ir izveidots
Step 9	Navigācija uz Offices	Offices sarakstu skats ir atvērts
Step 10	Pārbaudīt biroju no step 8	Birojs eksistē

4.1.4 Notikumu veidošanas tests. (create event test).

Step 4	Navigācija uz Events	Izveidots notikums (Step 3) eksistē
Step 5	Ielogoties kā Boss	Veiksmīga autorizācija
Step 6	Navigācija uz Events	Events sarakstu skats ir atvērts
Step 7	Izveidojiet notikumu (Eventu) (Aizpildiet visas rindas) Event for: eksistē 3 notikumu tipi	Notikums ir izveidots
Step 8	Navigācija uz Events	Izveidots notikums (Step 7) eksistē
Step 9	Ielogoties kā Manager	Veiksmīga autorizācija
Step 10	Navigācija uz Events	Events sarakstu skats ir atvērts
Step 11	Izveidojiet notikumu (Eventu) (Aizpildiet visas rindas) Event for: eksistē 2 notikumu tipi (Local, Team)	Notikums ir izveidots
Step 12	Navigācija uz Events	Izveidots notikums (Step 11) eksistē
Step 13	Ielogoties kā User	Veiksmīga autorizācija
Step 14	Navigācija uz Events	Events sarakstu skats ir atvērts
Step 15	Izveidojiet notikumu (Eventu) (Aizpildiet visas rindas) Event for: noklusējuma stāvoklis noklusējuma stāvoklis = (lietotāja komanda)	Notikums ir izveidots
Step 16	Navigācija uz Events	Izveidots notikums (Step 11) eksistē

4.1.5 Komandas veidošanas tests. (create team test).

Step Name	Description	Expected Result
Step 1	Ielogoties kā Administrators	Veiksmīga autorizācija
Step 2	Navigācija uz Users	Users sarakstu skats ir atvērts
Step 3	Izveidojiet divus lietotājus (Manager) <input type="checkbox"/>	Lietotāji ir izveidoti
Step 4	Navigācija uz Offices	Biroju sarakstu skats ir atvērts
Step 5	Izveidojiet divus birojus	Biroji ir izveidoti
Step 6	Navigācija uz Teams	Komandas sarakstu skats ir atvērts
Step 7	Izveidojiet komandu (Aizpildiet visas rindas) Office: pirmais izveidots birojs no Step 5 Manager: pirmais izveidots Manager no Step 3	Komanda ir izveidota
Step 8	Navigācija uz Teams	Komanda no step 7 eksistē
Step 9	Ielogoties kā Boss	Veiksmīga autorizācija
Step 10	Navigācija uz Teams	Komandas sarakstu skats ir atvērts
Step 11	Izveidojiet komandu (Aizpildiet visas rindas) Office: otrais izveidots birojs no Step 5 Manager: otrais izveidots Manager no Step 3	Komanda ir izveidota
Step 12	Navigācija uz Teams	Komanda no step 11 eksistē

4.1.6 Atļauja veidot lietotājus tests. (create user access test).

Step Name	Description	Expected Result
Step 1	Ielogoties kā Administrators	Veiksmīga autorizācija
Step 2	Navigācija uz Users	Users sarakstu skats ir atvērts
Step 3	Ir atļauja izveidot lietotāju <input type="checkbox"/>	Poga "Create user" eksistē
Step 4	Ielogoties kā Boss	Veiksmīga autorizācija
Step 5	Navigācija uz Users	Users sarakstu skats ir atvērts
Step 6	Ir atļauja izveidot lietotāju	Poga "Create user" eksistē
Step 7	Ielogoties kā Manager	Veiksmīga autorizācija
Step 8	Navigācija uz Users	Users sarakstu skats ir atvērts
Step 9	Nav atļauja izveidot lietotāju	Poga "Create user" neeksistē
Step 10	Ielogoties kā User	Veiksmīga autorizācija
Step 11	Navigācija uz Users	Users sarakstu skats ir atvērts
Step 12	Nav atļauja izveidot lietotāju	Poga "Create user" neeksistē

4.1.7 Atļauja veidot birojus tests. (create offices access test).

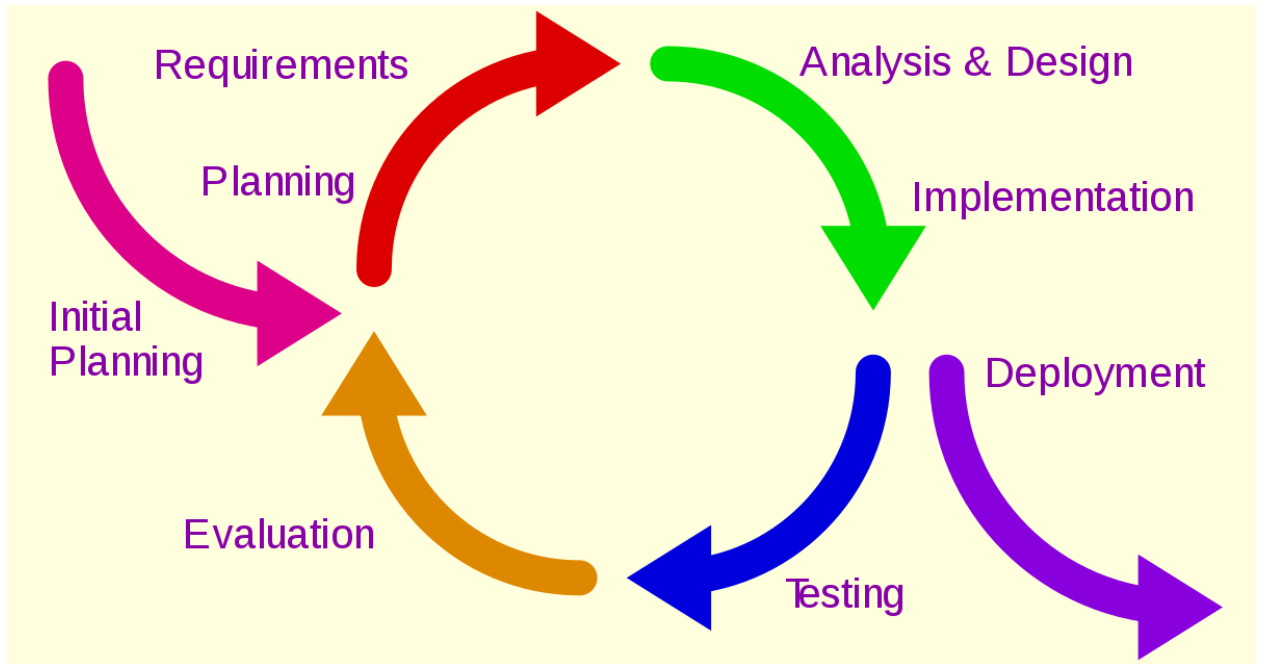
Step Name	Description	Expected Result
Step 1	Ielogoties kā Administrators	Veiksmīga autorizācija
Step 2	Navigācija uz Offices	Biroju sarakstu skats ir atvērts
Step 3	Ir atļauja izveidot biroju <input type="checkbox"/>	Poga "Create offices" eksistē
Step 4	Ielogoties kā Boss	Veiksmīga autorizācija
Step 5	Navigācija uz Offices	Biroju sarakstu skats ir atvērts
Step 6	Ir atļauja izveidot biroju	Poga "Create offices" eksistē
Step 7	Ielogoties kā Manager	Veiksmīga autorizācija
Step 8	Navigācija uz Offices	Biroju sarakstu skats ir atvērts
Step 9	Nav atļauja izveidot biroju	Poga "Create offices" neeksistē
Step 10	Ielogoties kā User	Veiksmīga autorizācija
Step 11	Navigācija uz Offices	Biroju sarakstu skats ir atvērts
Step 12	Nav atļauja izveidot biroju	Poga "Create offices" neeksistē

4.1.8 Atļauja veidot komandu tests. (create offices access test).

Step Name	Description	Expected Result
Step 1	Ielogoties kā Administrators	Veiksmīga autorizācija
Step 2	Navigācija uz Teams	Komandas sarakstu skats ir atvērts
Step 3	Ir atļauja izveidot komandus <input type="checkbox"/>	Poga "Create team" eksistē
Step 4	Ielogoties kā Boss	Veiksmīga autorizācija
Step 5	Navigācija uz Teams	Komandas sarakstu skats ir atvērts
Step 6	Ir atļauja izveidot komandu	Poga "Create team" eksistē
Step 7	Ielogoties kā Manager	Veiksmīga autorizācija
Step 8	Navigācija uz Teams	Komandas sarakstu skats ir atvērts
Step 9	Nav atļauja izveidot komandus	Poga "Create team" neeksistē
Step 10	Ielogoties kā User	Veiksmīga autorizācija
Step 11	Navigācija uz Teams	Komandas sarakstu skats ir atvērts
Step 12	Nav atļauja izveidot komandu	Poga "Create team" neeksistē

4.2 Programmatūras projekta organizācija

Sistēmas izstrāde tika veikta pēc iteratīvā modeļa. Bija nostādīts un noteikts galamērķis izveidot sistēmu lieliem uzņēmumiem, uz lietotāju lomu pamata. Sistēmas funkcionālais mērķis ir uzlabot teambuilding un attīstīt profesionālās prasmes caur eventiem par tuvākajām studiju iespējām. Darbs tika veikts, palielinot sistēmas funkcionalitāti. Šis izstrādes tips bija izvēlēts tā, kā sistēma tika uzskatīta par lielu projektu, realizācijas detaļas evolucionēja laika gaitā. Šie faktori ir vislabāk piemēroti izstrādei pēc iteratīvam modelim.



4.3 Kvalitātes nodrošināšana

PPS un PPA izstrāde tika veikta saskaņā ar - „LVS 68:1996 Programmatūras prasību specifikācijas ceļvedis” un „LVS 72:1996 Ieteicamā prakse programmatūras projektējuma aprakstīšanai” izstrādes standartiem.

Lai kontrolētu sistēmas versiju, tika izmantots gits. Koda analīzei tika izmantots statistikas

plugins. Izstrādāšanas procesā sistēma tika testēta. Dokumentācijas pamata tika izveidots testējamo sistēmu dokumentācijas saraksts (smoke tests) izstrādātajai sistēmai.

4.4 Darbietilpības novērtējums

4.4.1tabula. COCOMO kalkulators.

COCOMO II - Constructive Cost Model

Models: COCOMO
 Monte Carlo Risk: Off
 Auto Calculate: Off

Software Size Sizing Method: Source Lines of Code

SLOC: 4375
 % Design Modified: 0
 % Code Modified: 0
 % Integration Required: 0
 Assessment and Assimilation (0% - 8%): 0
 Software Understanding (0% - 50%): 0
 Unfamiliarity (0-1): 0

Software Scale Drivers

Precedentedness: Nominal
 Development Flexibility: Nominal
 Architecture / Risk Resolution: Nominal
 Team Cohesion: Nominal
 Process Maturity: Nominal

Software Cost Drivers

Product
 Required Software Reliability: Low
 Data Base Size: Low
 Product Complexity: Low
 Developed for Reusability: Nominal
 Documentation Match to Lifecycle Needs: Low

Personnel
 Analyst Capability: Very High
 Programmer Capability: Very High
 Personnel Continuity: Very High
 Application Experience: High
 Platform Experience: High
 Language and Toolset Experience: High

Platform
 Time Constraint: Nominal
 Storage Constraint: Nominal
 Platform Volatility: Nominal

Project
 Use of Software Tools: Nominal
 Multisite Development: Nominal
 Required Development Schedule: Nominal

Maintenance: Off

Software Labor Rates
 Cost per Person-Month (Dollars):
 Calculate

Results

Software Development (Elaboration and Construction)
 Effort = 3.1 Person-months
 Schedule = 5.3 Months

Staffing Profile
 Your project is too small to display a staffing profile due to truncation.

4.4.2tabula. Statistic plugin.

Statistic

Source File	Total Lines	Source Code Lines	Source Code Lines [%]	Comment Lines	Comment Lines [%]	Blank Lines	Blank Lines [%]
VerificationController.php	41	15	37%	20	49%	6	15%
verify.blade.php	24	21	88%	0	0%	3	12%
VerifyCsrfToken.php	24	9	38%	11	46%	4	17%
web.php	70	43	61%	10	14%	17	24%
welcome.blade.php	84	73	87%	0	0%	11	13%
Total:	5915	4375	74%	867	15%	673	11%

4.5 Secinājumi

Kvalifikācijas darba mērķis bija izveidot tīmekļa lietojumprogrammu ar notikumiem (eventiem) lielos uzņēmumos. Darba gaitā tika pētīta lielo uzņēmumu struktūra. Sistēma tika izstrādāta, pamatojoties uz Cognizant korporatīvo sistēmu, ņemot vērā projekta turpmāko attīstību. Izveidota praktiska notikumu izveides programma. Projekta gaitā tika uzlabotas PHP lietošanas prasmes, front-end prasmes un analītiskā domāšana, veidojot projekta arhitektūru.

4.6 Izmantotā literatūra

1. Rudīte Čevere, LVS 68:1996 „Programmatūras prasību specifikācijas ceļvedis”.
2. Rudīte Čevere, LVS 72:1996 „Ieteicamā prakse programmatūras projektējuma aprakstīšanai”.
3. Tīmekļa izstrādesportāls
Pieejams: <https://www.w3schools.com/>
4. Terminu vārdnīca
Pieejams: <http://termini.lza.lv/term.php>
5. Mācībuportāls
Pieejams: <https://eu.udacity.com/>
6. Mācībuportāls
Pieejams: <https://moodle.cognizant.com/cmoooc/course/view.php?id=123>
Nosaukums: DIGITAL 301 - Web UI Developer Foundation Certification Preparatory
7. Prezentācija “2018 Datubāzes projektēšana - loģiskais un fiziskais modelis”
Pieejams: <https://estudijas.lu.lv/>
8. Online enciklopēdijā “Wikipedia”
Pieejams: https://en.wikipedia.org/wiki/Iterative_and_incremental_development
9. COCOMO II – Constructive Cost Model
Pieejams: <https://csse.usc.edu/tools/COCOMOII.php>
10. PhpShtor – JetBrains (Framework)
Pieejams: https://www.jetbrains.com/phpstorm/specials/phpstorm/phpstorm.html?gclid=Cj0KCQjwz6PnBRCPARIsAN0tCw0WGPcbd6bY52lsXAkY4RzCkbGIHEQqJ30NNOX60HZ17cGkkiQkPJcaAsgTEALw_wcB

4.7 Programmatūras pirmkods

Lietotāju tabula

```
<?php
```

```
use Illuminate\Support\Facades\Schema;
use Illuminate\Database\Schema\Blueprint;
use Illuminate\Database\Migrations\Migration;

class CreateUsersTable extends Migration
{
    /**
     * Run the migrations.
     *
     * @return void
     */
    public function up()
    {
        Schema::create('users', function (Blueprint $table) {
            $table->engine = 'InnoDB';
            $table->bigIncrements('id');
            $table->string('name');
            $table->string('surname');
            $table->string('email')->unique();
            $table->timestamp('email_verified_at')->nullable();
            $table->string('password');
            $table->rememberToken();
            $table->timestamps();
            $table->unsignedBigInteger('role_id')->default(0);
            /**
             * Role values: 0 - user, 1 - manager, 2 - boss, 3 - moderator, 4
             - admin
             */
            $table->unsignedBigInteger('team_id')->nullable();
        });
    }

    /**
     * Reverse the migrations.
     *
     * @return void
     */
    public function down()
    {
        Schema::table('users', function (Blueprint $table) {
            $table->dropForeign('users_team_id_foreign');
            $table->dropForeign('users_role_id_foreign');
        });
        Schema::dropIfExists('users');
    }
}
```

Notikumu apraksta kontrolleris.

```
<?php
```

```
namespace App\Http\Controllers;

use App\EventDescription;
use App\GlobalEvent;
use App\LocalEvent;
use App\Team;
use App\TeamEvent;
use Illuminate\Http\Request;
use Illuminate\Support\Facades\Auth;
use Illuminate\Support\Facades\Validator;

class EventDescriptionController extends Controller
{
    /**
     * EventDescriptionController constructor.
     */
    public function __construct()
    {
        $this->middleware('auth');
    }

    /**
     * Returns view with accessible by user event listing.
     *
     * @return \Illuminate\Contracts\View\Factory|\Illuminate\View\View
     */
    public function index()
    {
        $ids = $this->getReachableIds();

        $events = EventDescription::whereIn('id', $ids)
            ->orderBy('id', 'desc')
            ->get() ?: array();

        return view('event.index', ['events' =>$events, 'all' =>true]);
    }

    /**
     * Returns event view.
     *
     * @param $id
     * @return \Illuminate\Contracts\View\Factory|\Illuminate\View\View
     */
    public function show($id)
    {
        if ($event = EventDescription::find($id)) {
            if ($event->haveAccess(Auth::user())) {
                return view('event.show', ['event' =>$event]);
            } else {
                return redirect()->back()->with('error', 'Not enough permissions to show this!');
            }
        } else {
            return redirect()->back()->with('error', 'Requested event does not exist anymore!');
        }
    }
}

/**
```

```

        * Returns event editing form.
        *
        * @param $id
        * @return \Illuminate\Contracts\View\Factory|\Illuminate\View\View
    */
    public function edit($id)
    {
        if ($event = EventDescription::find($id)) {
            if (Auth::id() === $event->author->id || Auth::user()->isAdmin()) {
                return view('event.edit', ['event' =>$event]);
            } else {
                return redirect()->back()->with('error', 'No permission for editing this event!');
            }
        } else {
            return redirect()->back()->with('error', 'Requested event does not exist anymore!');
        }
    }

    /**
     * Returns event creation form.
     *
     * @return \Illuminate\Contracts\View\Factory|\Illuminate\View\View
    */
    public function create()
    {
        return view('event.create');
    }

    /**
     * Stores new event data.
     *
     * @param Request $request
     * @return \Illuminate\Http\RedirectResponse|\Illuminate\Routing\Redirector
    */
    public function store(Request $request)
    {
        $this->validator($request)->validate();

        $type = (int)$request->get('event_for');
        $id = null;
        if ($type === 1) { //team event
            return TeamEventController::store($request);
        } elseif ($type === 2) { //local event
            return LocalEventController::store($request);
        } elseif ($type === 3) { //global event
            return GlobalEventController::store($request);
        }
        return redirect()->back()->with('error', 'Unknown error, event was not created!');
    }

    /**
     * Validates correct event type.
     *
     * @param Request $request
     * @return \Illuminate\Contracts\Validation\Validator
    */
    public function validator(Request $request)
    {
        return Validator::make($request->all(), [

```

```

'event_for' =>['required', 'min:1', 'max:3'],
]);
}

/**
 * Stores updated event info.
 *
 * @param Request $request
 * @param $id
 */
public function update(Request $request, $id)
{
    $data = $request->all();
    $this->updateValidator($data)->validate();

    EventDescription::where('id', $id)->update([
        'name' =>$data['name'],
        'description' =>$data['description'],
        'event_type_id' =>$data['acttype'],
        'timeFrom' =>$data['timeFrom'],
        'timeTo' =>$data['timeTo'],
    ]);

    return redirect('/event/'.$id)->with('success', 'Event data updated
    successfully!');
}

/**
 * Validates updated event data.
 *
 * @param array $data
 * @return \Illuminate\Contracts\Validation\Validator
 */
public function updateValidator(array $data)
{
    return Validator::make($data, [
        'name' =>['required', 'string', 'min:3', 'max:191'],
        'description' =>['required', 'string', 'min:3', 'max:191'],
        'acttype' =>['required', 'integer', 'exists:event_types,id'],
        'timeFrom' =>['required', 'date', 'before:timeTo'],
        'timeTo' =>['required', 'date', 'after:' . date('d-M-Y H:i')],
    ]);
}

/**
 * Deletes event record by provided EventDescription id.
 *
 * @param $id
 * @return \Illuminate\Http\RedirectResponse
 */
public function destroy($id)
{
    if (Auth::id() === EventDescription::find($id)->user_id
    || Auth::user()->isAdmin()) {
        EventDescription::destroy($id);
        return redirect('/event')->with('success', 'Delete successful!');
    } else {
        return redirect()->back()->with('error', 'Not enough permissions!');
    }
}

/**
 * Returns list of running event ids.

```

```

    *
    * @return array
    */
public static function getActualEvents()
{
    $ids = EventDescriptionController::getReachableIds();

    return EventDescription::whereIn('id', $ids)
->where('timeTo', '>=', date('Y-m-d H:i:s', time()))
->where('timeFrom', '<=', date('Y-m-d H:i:s', time()))
->get() ?: array();
}

/**
 * Return list of accessible by user event ids.
 *
 * @return array
 */
private static function getReachableIds()
{
    $user = Auth::user();

    $ids = array();

    $ids = array_merge($ids, GlobalEvent::whereNotNull('description_id')-
>pluck('description_id')->toArray());

    if ($user->isUser()) {
        if ($user->team) {
            $ids = array_merge(
                $ids,
                LocalEvent::where('office_id', $user->team->office_id)-
                >pluck('description_id')->toArray(),
                TeamEvent::where('team_id', $user->team_id)->pluck('description_id')-
                >toArray()
            );
        }
        elseif ($user->isManager()) {
            $offices = Team::where('manager_id', $user->id)->distinct('office_id')-
            >pluck('office_id')->toArray();
            $teams = Team::where('manager_id', $user->id)->pluck('id')->toArray();
            $ids = array_merge(
                $ids,
                LocalEvent::whereIn('office_id', $offices)->pluck('description_id')-
                >toArray(),
                TeamEvent::whereIn('team_id', $teams)->pluck('description_id')->toArray()
            );
        }
        else {
            $ids = EventDescription::all()->pluck('id')->toArray();
        }
    }

    return $ids;
}
}

```

Kvalifikācijas darbs „*Company Event Manager*” izstrādāts Latvijas Universitātes Datorikas fakultātē.

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

Autors: *Maksims Pavļenko* _____ .05.2019.

Rekomendēju darbu aizstāvēšanai

Darba vadītājs: *Dr.dat. Ivo Odītis* _____ .05.2019.

Recenzents: *Egons Šolmanis*

Darbs iesniegts 27.05.2019.

Kvalifikācijas darbu pārbaudījumu komisijas sekretāre: *Darja Solodovņikova* _____

Darbs aizstāvēts kvalifikācijas darbu pārbaudījuma komisijas sēdē

____.06.2019. prot. Nr. _____

Komisijas sekretārs(-e): _____