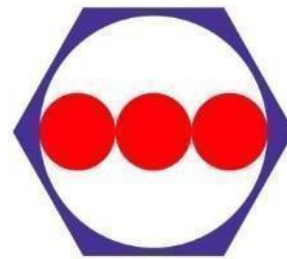




**Univerzitet u Novom Sadu
FAKULTET TEHNIČKIH NAUKA**

**Departman za industrijsko inženjerstvo i inženjerski menadžment
Inženjerstvo informacionih sistema**



Praćenje poslovanja preduzeća za pružanje vulkanizerskih usluga

-SISTEMI BAZA PODATAKA-

Filip Josifljević IT44g2020

Novi Sad, 2023.

SADRŽAJ

UVOD	2
ANALIZA PROGRAMSKOG DOMENA	2
ER MODEL	3
ER MODEL PODŠEME	4
TABELARNI PRIKAZ OBELEŽJA I OGRANIČENJA	5
RELACIONI MODEL	6
DDL	6
DML	6
SQL	6
OBJEKTI	6
UPITI	7
FUNKCIJE	8
PROCEDURE	9
TRIGERI	10
ZAKLJUČAK	11

UVOD

Projekat koji je predstavljen u ovoj dokumentaciji podrazumeva razvoj baze podataka za potrebe rada jednog dela vulkanizerskog preduzeća, odnosno razvoj dela baze podataka koja će korisnicima da pruži mogućnost dodavanja, manipulacije, uređivanja i uklanjanja zahteva za usluge, vrsta usluga, aktivnosti, operacija, koraka, materijala, mašina i radnih naloga.

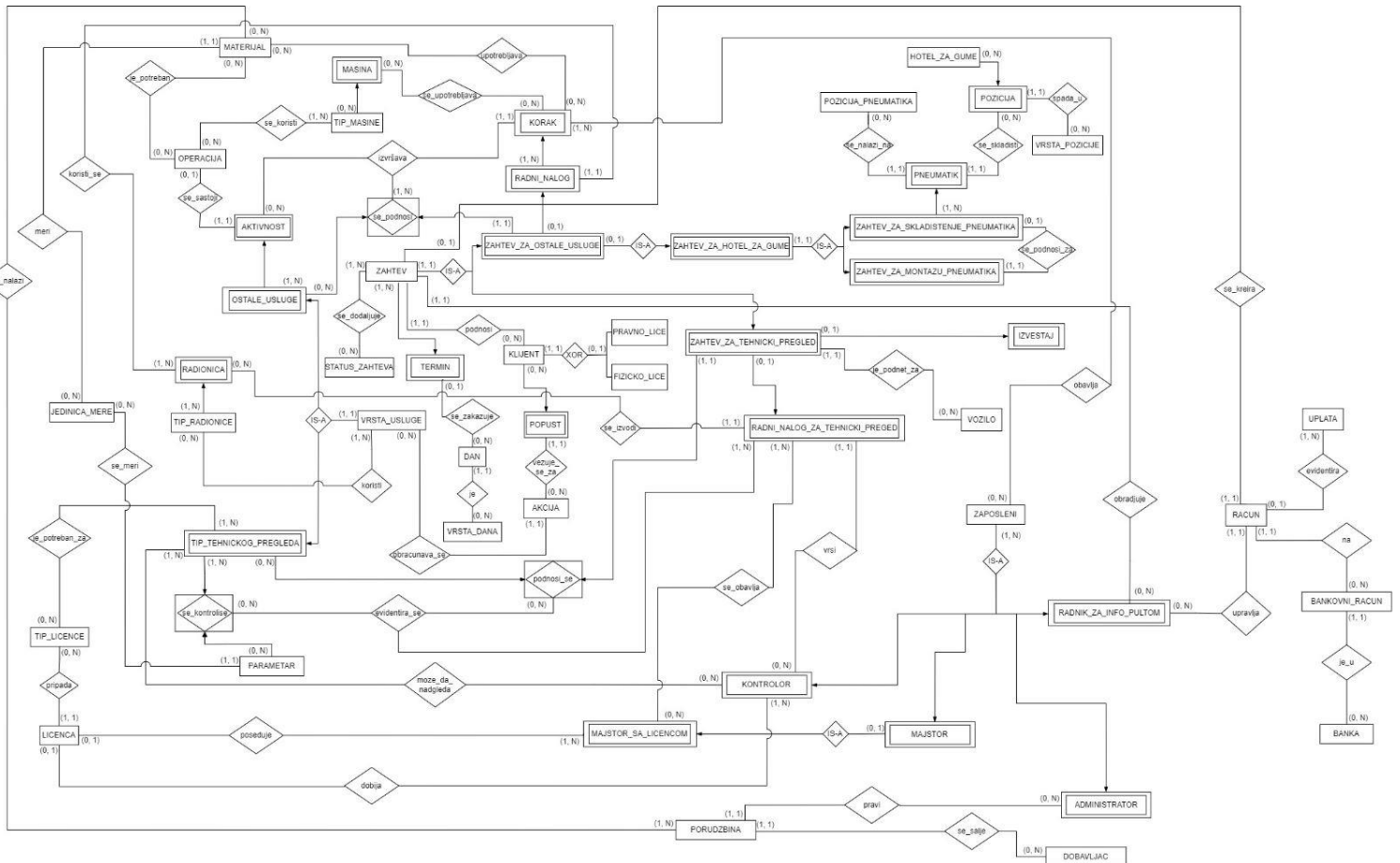
U daljem tekstu, priložen je ER model podataka, čijim prevođenjem je kreiran relacioni model podataka. Takođe, priloženi su i primeri implementacije određenih složenih ograničenja, funkcija, procedura i trigera, kao i propratni tabelarni prikaz svih obeležja relevantnih za realizaciju ovog projekta. Sve ove funkcionalnosti su implementirane nad delom glavne šeme baze podataka, tj. podšemom koja se naziva “MašineMaterijali”.

ANALIZA PROGRAMSKOG DOMENA

Cilj projekta jeste omogućavanje rada vulkanizerske radnje. Bazira se na vođenju evidencije o zahtevima koje korisnici podnose, mogućim vrstama usluga koje vulkanizerska radnja nudi, svim aktivnostima, operacijama i koracima koji se izvršavaju pri odrađivanju radnih naloga kreiranih na osnovu pomenutih zahteva. Takođe se vode evidencije o tipovima mašina, kao i pojedinačnim mašinama, i njihovima karakteristikama, koje su dostupne radnicima radnje za ispunjavanje korisničkih zahteva. Beleže se i podaci o raznim materijalima u skladištu i njihovoj količini kako bi se obezbedio nesmetani rad firme. Svi ovi podaci se beleže zajedno u integrisanoj strukturi koja nam omogućava da pratimo, ažuriramo i obezbedimo efikasno poslovanje jedne višeuslužne vulkanizerske radnje.

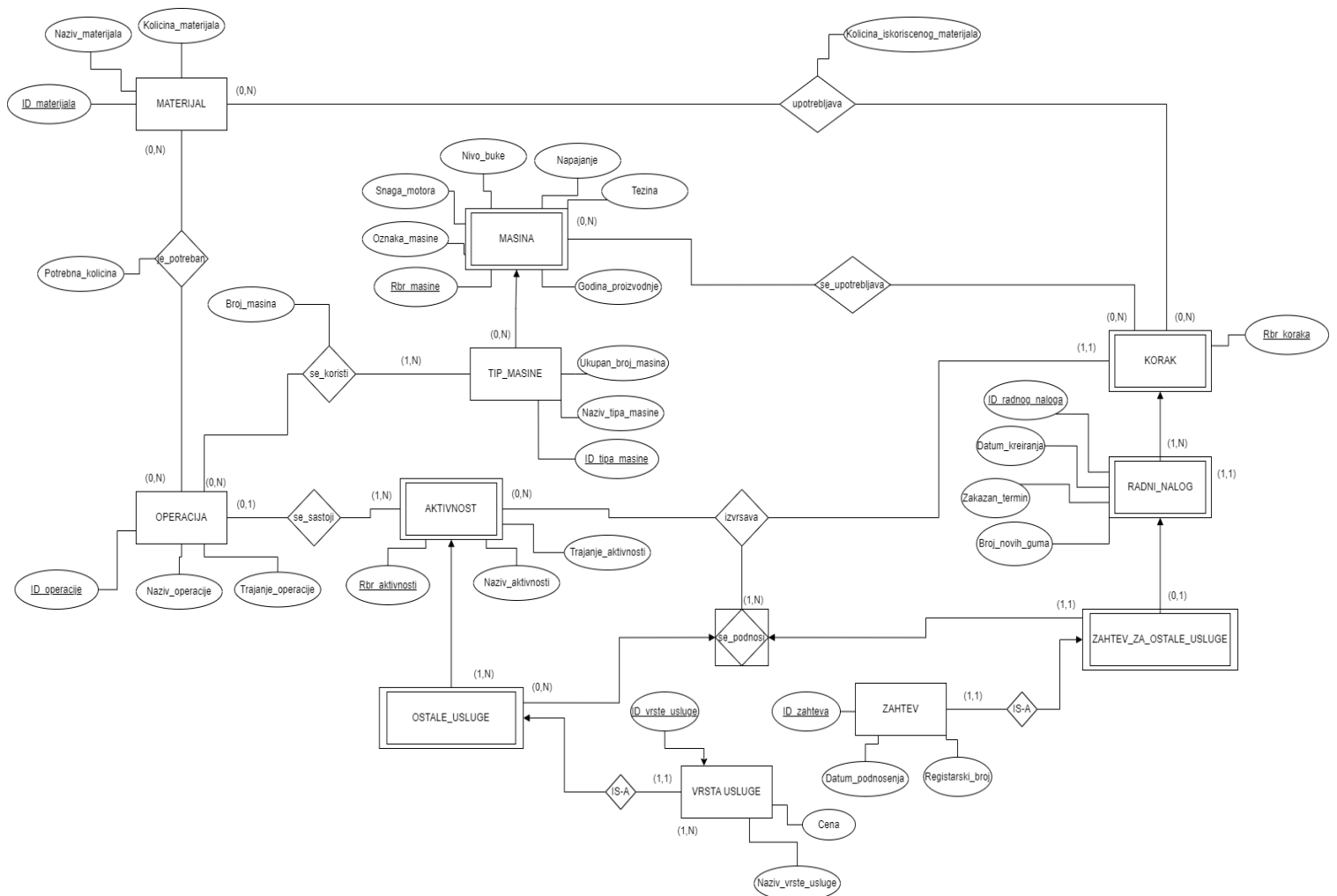
ER MODEL

Prikaz er modela vulkanizerske radnje.



ER MODEL PODŠEME

Prikaz ER modela podšeme MašineMaterijali



Podšema prikazuje strukturu koja pruža mogućnost evidencije podataka o sprovedenim uslugama na osnovu podnetih zahteva klijenata, kao i o njihovim koracima i aktivnostima. U koracima se upotrebljavaju mašine određenog tipa, koji se u isto vreme koristi i u raznim operacijama. Operacije zahtevaju određene materijale za svoje izvršenje, dok se ti isti materijali koriste i u koracima definisanim u radnim nalogima.

Radni nalozi se formiraju na osnovu podnetih zahteva za usluge koje ne uključuju tehnički pregled, oni sadrže svoj datum kreiranja koji ne sme biti raniji od datuma podnošenja zahteva, kao i zakazani termin u kojem će se radni nalog realizovati uz upotrebu 0 ili više guma.

TABELARNI PRIKAZ OBELEŽJA I OGRANIČENJA

1. *Tabela 3. 1 – Vrsta usluge*

Naziv obeležja	Opis obeležja	Tip podataka	Dužina podataka	Null	Uslov
<i>ID_vrste_usluge</i>	Jedinstvena šifra vrste usluge	INTEGER	8	⊥	Δ
<i>naziv_vrste_usluge</i>	Naziv vrste usluge	VARCHAR	30	⊥	Δ
<i>cena</i>	Cena usluge	INTEGER	8	⊥	Δ
Ne mogu postojati 2 ista naziva vrste usluga					
KLJUČ	<i>ID_vrste_usluge</i>				

Tabela 3. 2 – Ostale usluge

Naziv obeležja	Opis obeležja	Tip podataka	Dužina podataka	Null	Uslov
<i>ID_vrste_usluge</i>	Jedinstvena šifra vrste usluge	INTEGER	8	⊥	Δ
KLJUČ	<i>ID_vrste_usluge</i>				

Tabela 3. 3 - Operacija

Naziv obeležja	Opis obeležja	Tip podataka	Dužina podataka	Null	Uslov
<i>ID_operacije</i>	Jedinstvena šifra operacije	INTEGER	5	⊥	d > 0
<i>naziv_operacije</i>	Naziv operacije	VARCHAR	20	⊥	Δ
<i>trajanje_operacije</i>	Trajanje operacije u minutima	INTEGER	3	⊥	d > 0
KLJUČ	<i>ID_operacije</i>				

Tabela 3. 4 - Aktivnost

Naziv obeležja	Opis obeležja	Tip podataka	Dužina podataka	Null	Uslov
<i>ID_vrste_usluge</i>	Jedinstvena šifra vrste usluge	INTEGER	8	⊥	Δ
<i>rbr_aktivnosti</i>	Redni broj aktivnosti u usluzi	INTEGER	5	⊥	d > 0
<i>naziv_aktivnosti</i>	Naziv aktivnosti	VARCHAR	30	⊥	Δ
<i>trajanje_aktivnosti</i>	Trajanje aktivnosti u minutima	INTEGER	3	⊥	d > 0
<i>ID_operacije</i>	Jedinstvena šifra operacije	INTEGER	5	⊥	Δ
KLJUČ	<i>ID_vrste_usluge + rbr_aktivnosti</i>				

Tabela 3. 5 – Tip mašine

Naziv obeležja	Opis obeležja	Tip podataka	Dužina podataka	Null	Uslov
<i>ID_tipa_masine</i>	Jedinstvena šifra tipa mašine	INTEGER	2	⊥	d > 0
<i>naziv_tipa_masine</i>	Naziv tipa mašine	VARCHAR	30	⊥	Δ
<i>Ukupan_broj_masina</i>	Ukupan broj mašina tog tipa	INTEGER	30	⊥	d > 0
KLJUČ	<i>ID_operacije</i>				

Tabela 3. 6 – Mašina

Naziv obeležja	Opis obeležja	Tip podataka	Dužina podataka	Null	Uslov
<i>ID_tipa_masine</i>	Jedinstvena šifra tipa mašine	INTEGER	2	⊥	Δ
<i>rbr_masine</i>	Redni broj mašine tog tipa	INTEGER	5	⊥	$d > 0$
<i>oznaka_masine</i>	Oznaka modela mašine	VARCHAR	20	⊥	Δ
<i>snaga_motora</i>	Snaga motora mašine	INTEGER	3	T	$d \geq 0$
<i>nivo_buke</i>	Nivo buke mašine u dB	INTEGER	3	⊥	$d > 0$
<i>napajanje</i>	Napajanje mašine u V	INTEGER	3	T	$d \geq 0$
<i>tezina</i>	Težina mašine u kg	INTEGER	4	T	$d > 0$
<i>godina_proizvodnje</i>	Godina proizvodnje mašine	INTEGER	4	⊥	$d > 0$
KLJUČ	<i>ID_tipa_masine</i> + <i>rbr_masine</i>				

Tabela 3. 7 – Se koristi

Naziv obeležja	Opis obeležja	Tip podataka	Dužina podataka	Null	Uslov
<i>ID_tipa_masine</i>	Jedinstvena šifra tipa mašine	INTEGER	2	⊥	Δ
<i>ID_operacije</i>	Jedinstvena šifra operacije	INTEGER	5	⊥	Δ
<i>broj_masina</i>	Broj mašina koje se koriste	INTEGER	2	T	$d \geq 0$
KLJUČ	<i>ID_operacije</i> + <i>ID_tipa_masine</i>				

Tabela 3. 8 – Materijal

Naziv obeležja	Opis obeležja	Tip podataka	Dužina podataka	Null	Uslov
<i>ID_materijala</i>	Jedinstvena šifra materijala	INTEGER	5	⊥	$d > 0$
<i>naziv_materijala</i>	Naziv materijala	VARCHAR	20	⊥	Δ
<i>kolicina_materijala</i>	Količina materijala	VARCHAR	3	⊥	Δ
KLJUČ	<i>ID_operacije + ID_tipa_masine</i>				

Tabela 3. 9 – Je potreban

Naziv obeležja	Opis obeležja	Tip podataka	Dužina podataka	Null	Uslov
<i>ID_materijala</i>	Jedinstvena šifra materijala	INTEGER	5	⊥	Δ
<i>ID_operacije</i>	Jedinstvena šifra operacije	VARCHAR	20	⊥	Δ
<i>potrebna_kolicina</i>	Potrebna količina materijala	VARCHAR	8	⊥	Δ
KLJUČ	<i>ID_operacije + ID_tipa_masine</i>				

Tabela 3. 10 – Zahtev

Naziv obeležja	Opis obeležja	Tip podataka	Dužina podataka	Null	Uslov
<i>ID_zahteva</i>	Jedinstvena šifra zahteva	INTEGER	8	⊥	d > 0
<i>registarski_broj</i>	Registarski broj vozila	VARCHAR	10	T	Δ
<i>datum_podnosenja</i>	Datum podnošenja zahteva	DATE	Δ	⊥	Δ
KLJUČ	<i>ID_operacije + ID_tipa_masine</i>				

Tabela 3. 11 – Zahtev za ostale usluge

Naziv obeležja	Opis obeležja	Tip podataka	Dužina podataka	Null	Uslov
<i>ID_zahteva_za_ostale_usluge</i>	Jedinstvena šifra zahteva za ostale usluge	INTEGER	8	⊥	Δ
<i>ID_vrste_usluge</i>	Jedinstvena šifra vrste usluge	INTEGER	8	T	Δ
<i>datum_podnosenja</i>	Datum podnošenja zahteva	DATE	Δ	⊥	Δ
KLJUČ	<i>ID_zahteva_za_ostale_usluge</i>				

Tabela 3. 12 – Radni nalog

Naziv obeležja	Opis obeležja	Tip podataka	Dužina podataka	Null	Uslov
<i>ID_radnog_naloga</i>	Jedinstvena šifra radnog naloga	INTEGER	8	⊥	$d > 0$
<i>ID_zahteva_za_ostale_usluge_rn</i>	Jedinstvena šifra zahteva za ostale usluge radnog naloga	INTEGER	8	T	Δ
<i>datum_kreiranja</i>	Datum kreiranja radnog naloga	DATE	Δ	⊥	Δ
<i>zakazan_termin</i>	Vreme zakazanog termina	TIME	Δ	⊥	Δ
<i>broj_novih_guma</i>	Broj guma korišćenih u realizaciji radnog naloga	INT	2	T	$d \geq 0$
Ne smeju postojati 2 ista ID-ja radnog naloga					
KLJUČ	<i>ID_zahteva_za_ostale_usluge_rn</i>				

Tabela 3. 13 – Korak

Naziv obeležja	Opis obeležja	Tip podataka	Dužina podataka	Null	Uslov
<i>rbr_koraka</i>	Redni broj koraka u radnom nalogu	INTEGER	5	⊥	$d > 0$
<i>ID_zahteva_za_ostale_usluge_rn</i>	Jedinstvena šifra zahteva za ostale usluge radnog naloga	INTEGER	8	T	Δ
<i>ID_vrste_usluge</i>	Jedinstvena šifra vrste usluge	INTEGER	8	⊥	Δ
<i>rbr_aktivnosti</i>	Redni broj aktivnosti	INTEGER	5	⊥	Δ
<i>ID_zahteva_za_ostale_usluge</i>	Jedinstvena šifra zahteva za ostale usluge	NUMERIC	8	⊥	Δ
KLJUČ	<i>ID_zahteva_za_ostale_usluge_rn_ + rbr_koraka</i>				

Tabela 3. 14 – Se upotrebljava

Naziv obeležja	Opis obeležja	Tip podataka	Dužina podataka	Null	Uslov
<i>rbr_koraka</i>	Redni broj koraka u radnom nalogu	INTEGER	5	⊥	Δ
<i>ID_zahteva_za_ostale_usluge_rn</i>	Jedinstvena šifra zahteva za ostale usluge radnog naloga	INTEGER	8	⊥	Δ
<i>ID_tipa_masine</i>	Jedinstvena šifra tipa mašine	INTEGER	2	⊥	Δ
<i>rbr_masine</i>	Redni broj mašine	INTEGER	5	⊥	Δ
KLJUČ	<i>ID_zahteva_za_ostale_usluge_rn_ + rbr_koraka + ID_tipa_masine + rbr_masine</i>				

Tabela 3. 15 – Upotrebljava

Naziv obeležja	Opis obeležja	Tip podataka	Dužina podataka	Null	Uslov
<i>rbr_koraka</i>	Redni broj koraka u radnom nalogu	INTEGER	5	⊥	Δ
<i>ID_zahteva_za_ostale_usluge_rn</i>	Jedinstvena šifra zahteva za ostale	INTEGER	8	⊥	Δ

	usluge radnog naloga				
<i>Kolicina_iskoriscenog_mat</i>	Količina iskorišćenog materijala	INTEGER	10	⊥	Δ
<i>ID_materijala</i>	Jedinstvena šifra materijala	INTEGER	5	⊥	Δ
KLJUČ	<i>ID_zahteva_za_ostale_usluge_rn_ + rbr_koraka + ID_materijala</i>				

RELACIONI MODEL

Opis šta je to relacioni model i relaciona baza podataka (osnovni koncepti).

Relacioni model baze podataka predstavlja konceptualni model koji se koristi za definisanje i modelovanje strukture i organizacije podataka. Podaci se predstavljaju u formi tabela gde redovi predstavljaju torke a kolone predstavljaju obeležja. Nastao je 1970. God. I njegovim stvoriteljem se smatra Edgar F. Codd. Veze između tabela se ostvaruju pomoću ključeva koji vrše migracije iz jedne tabele u drugu po principu strogo definisanih pravila. Relaciona baza podataka predstavlja bazu zasnovanu na relacionom modelu.

```
AKTIVNOST( {Rbr_aktivnosti, Naziv_aktivnosti, Trajanje_aktivnosti, ID_operacije},  
            {ID_vrste_usluge + Rbr_aktivnosti})  
  
AKTIVNOST[ID_vrste_usluge]  $\subseteq$  OSTALE_USLUGE[ID_vrste_usluge]  
OSTALE_USLUGE[ID_vrste_usluge]  $\subseteq$  AKTIVNOST[ID_vrste_usluge]  
AKTIVNOST[ID_operacije]  $\subseteq$  OPERACIJA[ID_operacije]  
Null(AKTIVNOST, ID_operacije) = 1
```

4.1 Primer tipa entiteta prevedenog u relacioni model zajedno sa ograničenjima

DDL

DDL predstavlja Data Definition Language, tj. Jezik za definiciju podataka. On se koristi za izgradnju i upravljanje strukturom ili šemom baze podataka. Neke od glavnih komandi su:

- CREATE - za kreiranje šeme ili tabela zajedno sa svim njihovim obeležjima, tipovima podataka i ograničenjima
- ALTER - za modifikaciju već kreiranih tabela ili drugih objekata baze podataka
- DROP - za brisanje objekata kao što su tabele, pogledi, šeme, indeksi itd.
- TRUNCATE - za brisanje svih podataka iz tabele ostavljajući njenu strukturu i ograničenja netaknutim

Primer koda za kreiranje tabele :

```
CREATE TABLE MasineMaterijali.materijal  
(id_materijala NUMERIC(5) PRIMARY KEY,  
naziv_materijala VARCHAR(20) NOT NULL,  
kolicina_materijala VARCHAR(3) NOT NULL,  
CONSTRAINT CHK_id_materijala_value CHECK(id_materijala>0))
```

Primer koda za brisanje tabele :

DROP TABLE MasineMaterijali.materijal

Primer koda za modifikaciju tabele :

ALTER TABLE MasineMaterijali.materijal
ADD <column-name>
OR
DROP<column-name>

DML

DML predstavlja Data Manipulation Language koji se koristi za manipulaciju i čitanje podataka iz baze. Omogućava korisnicima da koriste glavne funkcionalnosti kao što su insert-ovanje podataka u bazu, njihovo update-ovanje i brisanje. Glavne komande su : **INSERT, UPDATE, DELETE.**

Primer INSERT komande :

INSERT INTO
MasineMaterijali.zahtev(id_zahteva,registarski_broj,datum_podnosenja)
VALUES(1,'NS-1071-FH','2019-07-21');
(primer unosa bez sekvence)
INSERT INTO MasineMaterijali.materijal(id_materijala, naziv_materijala,
kolicina_materijala)
VALUES(NEXT VALUE FOR MasineMaterijali.materijal_sequence, 'Bakar', 50);
(primer unosa sa sekvencom)

Primer UPDATE komande :

UPDATE MasineMaterijali.radni_nalog
SET datum_kreiranja='2020-01-01'
WHERE id_radnog_naloga=1

Primer DELETE komande :

DELETE FROM MasineMaterijali.zahtev WHERE id_zahteva=11

SQL

SQL predstavlja standardizovani upitni jezik koji se koristi za upravljanje i manipulaciju bazama podataka. Sadrži skup komandi koji omogućavaju korisnicima da vrše interakciju sa bazom i time obavljaju niz funkcionalnosti kao što su upis, modifikacija, brisanje i selekcija podataka iz baze. SQL ima široku primenu kao i niz alternativnih modifikovanih varijanti od strane raznih proizvođača kao što su Microsoft(SQL Server), Oracle, MySQL, PostgreSQL...

Primer selekcije imena, prezimena i plate radnika iz tabele Radnik :

```
SELECT Ime, Prz, Plt FROM Radnik
```

OBJEKTI

Objekti koji su korišćeni pri realizaciji ovog projekta jesu : šema, tabele, sekvence, funkcije, procedure, kursori i trigeri.

Šema je korišćena za implementaciju podšeme pod nazivom “MašineMaterijali” nad kojom su kreirani svi ostali objekti.

Sintaksa za kreiranje šeme :

```
CREATE SCHEMA MasineMaterijali
```

Tabele su kreirane na osnovu relacione šeme baze podataka koja je odrađena za celu šemu i sadrži sve definicije, obeležja, tipove podataka i ograničenja.

Sintaksa za kreiranje tabele :

```
CREATE TABLE table_name (  
    column1 datatype,  
    column2 datatype,  
    column3 datatype,  
    ...  
);
```

Dodavanje ograničenja pomoću ALTER komande :

```
ALTER TABLE table_name  
ADD CONSTRAINT constraint_name <constraint>
```

Sintaksa za kreiranje sekvenci :

```
CREATE SEQUENCE sequence_name  
  START WITH initial_value  
  INCREMENT BY increment_value  
  [ MINVALUE min_value | NOMINVALUE ]  
  [ MAXVALUE max_value | NOMAXVALUE ]  
  [ CYCLE | NOCYCLE ]  
  [ CACHE cache_size | NOCACHE ];
```

Sekvenca je upotrebljena za inkrementiranje obeležja primarnog ključa u tabelama Materijal i Operacija.

Sintaksa za kreiranje funkcija :

```
CREATE FUNCTION schema_name.function_name (@parameter1 datatype,  
@parameter2 datatype, ...)  
  RETURNS return_datatype  
  [ WITH { ENCRYPTION | SCHEMABINDING | RETURNS NULL ON NULL INPUT  
| CALLED ON NULL INPUT } ]  
  [ AS ]  
  BEGIN  
    -- Function body and logic  
    RETURN expression  
  END
```

Sintaksa za kreiranje procedura :

```
CREATE PROCEDURE schema_name.procedure_name
    @parameter1 datatype,
    @parameter2 datatype,
    ...
AS
BEGIN
    -- Procedure body and logic
END
```

Procedure i funkcije su u projektu iskorišćene za kreiranje složenih tipova ograničenja i funkcionalnosti koje se nisu mogli implementirati deklarativnim načinom.

Sintaksa za kreiranje kursora :

```
DECLARE cursor_name CURSOR [LOCAL | GLOBAL] [STATIC | KEYSET |
DYNAMIC | FAST_FORWARD | FORWARD_ONLY]
FOR
SELECT column1, column2, ...
FROM table_name
WHERE condition;

OPEN cursor_name;
FETCH NEXT FROM cursor_name INTO @variable1, @variable2, ...;

WHILE @@FETCH_STATUS = 0
BEGIN
    -- Cursor processing logic

    FETCH NEXT FROM cursor_name INTO @variable1, @variable2, ...;
END

CLOSE cursor_name;
DEALLOCATE cursor_name;
```

Sintaksa za kreiranje trigeru :

```
CREATE TRIGGER schema_name.trigger_name  
ON table_name  
[AFTER | INSTEAD OF] {INSERT | UPDATE | DELETE}  
AS  
BEGIN  
    -- Trigger body and logic  
END
```

UPITI

Prvi upit

Ovaj upit nam izlistava podatke(redni broj mašine sa alijasom “Redni broj”, naziv tipa sa alijasom “Naziv tipa mašine”, oznaku mašine kao “Oznaka”, težinu mašine kao “Težina” i godinu proizvodnje kao “Godina proizvodnje”) o svim neelektričnim mašinama proizvedenim posle 2016. God. Ovi podaci se mogu dobiti spajanjem tabele Mašina sa tabelom Se upotrebljava preko zajedničke kombinacije obeležja *id_tipa_masine+rbr_masine*, zatim spajanjem sa tabelom Korak preko zajedničke kombinacije obeležja *id_zahteva_za_ostale_usluge_rn+rbr_koraka*, zatim spajanjem sa tabelom Radni nalog preko zajedničkog obeležja *id_zahteva_za_ostale_usluge_rn* i na kraju spajanjem sa tabelom Tip Mašine preko zajedničkog obeležja *id_tipa_masine* i sortiranje po godini proizvodnje rastuće i težini opadajuće.

	Redni broj	Naziv tipa mašine	Oznaka	Težina	Godina proizvodnje
1	2	Dizalica za teretne gume	TX12004	122	2017
2	2	Dizalica za teretne gume	TX12004	122	2017
3	2	Dizalica za teretne gume	TX12004	122	2017
4	2	Dizalica za teretne gume	TX12004	122	2017
5	2	Dizalica za teretne gume	TX12004	122	2017
6	1	Alat za demontažu guma	TRK60001	50	2017
7	1	Alat za demontažu guma	TRK60001	50	2017
8	1	Alat za demontažu guma	TRK60001	50	2017
9	1	Alat za demontažu guma	TRK60001	50	2017
10	1	Alat za demontažu guma	TRK60001	50	2017
11	2	Alat za demontažu guma	TRS70006	53	2018
12	2	Alat za demontažu guma	TRS70006	53	2018
13	2	Alat za demontažu guma	TRS70006	53	2018

4.1 Rezultujuća tabela prvog upita

Drugi upit

Ovaj upit nam izlistava ukupan broj mašina lakših od 80kg svakog tipa kao “Broj mašina”, nazive njihovih tipova kao “Naziv tipa mašine”, njihovu težinu kao “Težina” i godinu kao “Godina proizvodnje” koje su korištene u zahtevima podnetim pre 2020-01-01. Ovo smo postigli spajanjem tabele Mašina sa tabelom Tip mašine preko zajedničkog obeležja *id_tipa_masine*, zatim spajanjem sa tabelom Se upotrebljava preko zajedničke kombinacije obeležja *id_tipa_masine+rbr_masine*, zatim spajanjem sa tabelom Korak preko zajedničke kombinacije obeležja *id_zahteva_za_ostale_usluge_rn+rbr_koraka*, zatim spajanjem sa tabelom Radni nalog preko zajedničkog obeležja *id_zahteva_za_ostale_usluge_rn* i na kraju spajanjem sa tabelom Zahtev preko zajedničkog obeležja *id_zahteva*. Mašine smo grupisali prema nazivu njihovog tipa, težini i godini proizvodnje.

	Broj mašina	Naziv tipa mašine	Težina	Godina proizvodnje
1	5	Alat za demontažu guma	50kg	2017
2	3	Alat za demontažu guma	53kg	2018
3	6	Brusilica za poliranje	4kg	2021
4	3	Mašina za balansiranje točkova	78kg	2018
5	6	Perac pod pritiskom	42kg	2020

4.2 Rezultujuća tabela drugog upita

Treći upit

Ovaj upit nam izlistava ukupnu količinu materijala korišćenog u svim radnim nalogima gde nisu menjane gume. Ovi podaci se izlistavaju u formatu : Upotrebljeno je <kol> materijala <naz> u radnom nalogu <ID_radnog_naloga>. Ovo se postiglo spajanjem tabele Upotrebljava sa tabelom Materijal preko zajedničkog obeležja *id_materijala* a zatim spajanjem sa tabelom Radni nalog preko zajedničkog obeležja *id_zahteva_za_ostale_usluge_rn*. Materijali su grupisani po nazivu i *id_radnog_naloga* a željeni format ispisa se dobio korišćenjem ugrađene funkcije CONCAT i navođenjem potrebnih parametara.

	Ukupna količina
1	Upotrebljeno je 5 materijala aktivna pena u radnom nalogu 1
2	Upotrebljeno je 4 materijala kapica ventila u radnom nalogu 3
3	Upotrebljeno je 4 materijala ulozak ventila u radnom nalogu 3
4	Upotrebljeno je 32 materijala olovni tegovi u radnom nalogu 5
5	Upotrebljeno je 2 materijala aktivna pena u radnom nalogu 8

4.3 Rezultujuća tabela trećeg upita

Četvrti upit

Ovaj upit nam služi za islistavanje svih zahteva odrađenih za vozila koja su registrovana u istom gradu kao i vozilo koje je platilo najjeftiniju uslugu. Prikazuju se podaci *id_zahteva* kao Zahtev, *datum_podnosenja* kao Datum podnošenja i *registarski_broj* kao Registracija. Podaci su iščitani spajanjem tabele Zahtev sa tabelom Zahtev za ostale usluge po obeležju *id_zahteva*, i tabelom Vrsta usluge po obeležju *id_vrste_usluge*. Da bi se uverili da su vozila registrovana u istom gradu izdvajamo prva 2 znaka registarskog broja vozila funkcijom LEFT sa parametrom 2 karaktera. Registarski broj vozila koje je platilo najskuplju uslugu se dobija tako što se spoje tabele Zahtev, Zahtev za ostale usluge i Vrsta usluge na isti način kao i pre i uzme se jedan registarski broj gde je cena jednaka najmanjoj ceni plaćenih usluga (funkcija TOP 1).

	Zahtev	Datum podnošenja	Registracija
1	1	2019-07-21	NS-1071-FH
2	2	2019-08-15	NS-9899-JI
3	3	2019-10-22	NS-2234-GG

4.4 Rezultujuća tabela četvrtog upita

Peti upit

Ovaj upit nam vraća nazive aktivnosti, usluga u kojima se one sadrže za sve usluge u kojima su korišćene mašine glasnije od 10dB. Naziv_vrste_usluge ima alijas "Usluga", naziv_aktivnosti ima alijas "Aktivnost", trajanje aktivnosti je zapisano u formatu : Ukupno su trajale <ukupno_trajanje> minuta – sa alijasom "Ukupno trajanje". Podatke smo izvukli spajanjem tabele Vrsta usluge, Zahtev za ostale usluge, Radni nalog, Se Upotrebljava i Mašina na iste načine kao i u prethodnim primerima. Na kraju smo sve grupisali po nazivima vrsta usluga i nazivima aktivnosti i sortirali ih po ukupnom trajanju aktivnosti opadajuće.

	Usluga	Aktivnost	Ukupno trajanje
1	Skidanje-nameštanje tocka	Odvrtanje šrafova	Ukupno su trajale : 11 minuta
2	Unutrašnje pranje	Brisanje volana	Ukupno su trajale : 6 minuta
3	Unutrašnje pranje	Prskanje tecnosti	Ukupno su trajale : 4 minuta
4	Skidanje-nameštanje tocka	Skidanje tocka	Ukupno su trajale : 1 minuta

4.5 Rezultujuća tabela petog upita

FUNKCIJE

Prva funkcija

Funkcija `Vrati_udeo` služi da vrati udeo određene usluge u ukupnoj ceni svih izvršenih usluga pre određenog vremenskog perioda. Kao ulazne parametre prima `id_vrste_usluge` i datum. Unutar funkcije su definisane pomoćne promenljive koje u sebi pamte podatke o ukupnoj ceni svih usluga pre prosleđenog vremenskog perioda i o ukupnoj ceni usluge sa prosleđenim id-jem pre istog vremenskog perioda. Udeo te usluge se računa kao odnos između suma cena prosleđene usluge i suma cena svih usluga. Kao izlaz iz funkcije imamo string u tačno definisanom formatu korišćenjem funkcije `CONCAT` gde udeo predstavljamo u procentima. Ukoliko usluga nije izvršena u tom periodu, odnosno njen udeo ne postoji, prosleđivaće se 0.

```
SELECT MasineMaterijali.vrati_udeo(2, '2020-01-01') AS Udeo
```

5.1 Poziv funkcije `vrati_udeo`

	Udeo
1	Usluga 9 sa nazivom Skidanje-nameštanje tocka ima udeo od 6.00% od svih usluga izvršenih pre 2020-01-01

5.2 Ispis iz funkcije za uslugu koja ima udela

	Udeo
1	Usluga 9 sa nazivom Skidanje-nameštanje tocka ima udeo od 6.00% od svih usluga izvršenih pre 2020-01-01

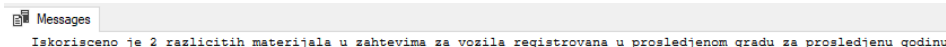
5.3 Ispis iz funkcije za uslugu koja nema udela

Druga funkcija

Funkcija `Vrati_materijal` nam vraća broj različitih materijala koji su iskorišćeni za sva vozila registrovana u prosleđenom gradu u prosleđenoj godini. Kao ulazne parametre prima ime grada i godinu. Koristimo pomoćnu promenljivu čija će vrednost postati skraćenica grada korišćena za registraciju na osnovu grada koji je prosleđen. Pomoćna promenljiva se poredi sa prva dva karaktera vrednosti obeležja *registarski_broj* iz tabele *Zahtev*. Izlaz iz funkcije nam predstavlja sumiranu količinu različitog materijala koji smo dobili upitom primenjujući pomenute uslove.

```
PRINT CONCAT('Iskorisceno je ',MasineMaterijali.vrati_materijal('beograd',2019),' razlicitih materijala u zahtevima za vozila registrovana u prosledjenom gradu za prosledjenu godinu')
```

4 Poziv funkcije `vrati_materijal`



Messages
Iskorisceno je 2 razlicitih materijala u zahtevima za vozila registrovana u prosledjenom gradu za prosledjenu godinu

5.5 Ispis količine materijala iz funkcije

PROCEDURE

Prva procedura

Procedura `usluge_procedura` nam za prosleđenu godinu vraća sve usluge i aktivnosti odrađene u toj godini. Kao ulazni parametar prihvata godinu tipa podatka `INTEGER`. Definišemo pomoćnu promenljivu *brusluga* koja dobija vrednost ukupnog broja svih vrsta usluga odrađenih u prosleđenoj godini. U slučaju da je broj usluga 0, imamo ispis da u godini nije odrađena nijedna usluga. Ako je broj veći od 0 onda deklariramo novi kursor koji će nam vraćati šifre vrste usluga, njihove nazive i cene za svaku odrađenu uslugu. Unutar tog kursora definišemo novu pomoćnu promenljivu *braktivnosti* koja predstavlja ukupan broj aktivnosti koji se odvijaju za tu uslugu. Ako je broj aktivnosti 0 ispisaće se odgovarajuća poruka, a u slučaju da je veći od 0 prelazi se na sledeći korak. Sada definišemo novi kursor koji nam služi za vraćanje svih pojedinačnih aktivnosti za usluge iz spoljašnjeg kursora. Za aktivnosti vraćamo podatke o rednom broju, nazivu i trajanju i ispisujemo ih u odgovarajućem obliku. Aktivnosti se unutar usluga sortiraju u rastućem poretku po njihovom rednom broju. Nakon ispisa odgovarajućih podataka prvo se zatvara i dealocira kursor za aktivnosti, a zatim i kursor za usluge. Na kraju smo dobili kompletan ispis svih usluga i u njima ugnježđenih aktivnosti.

```
EXEC MasineMaterijali.usluge_procedura
@godina = 2019 --ispravna

EXEC MasineMaterijali.usluge_procedura
@godina = 2020 --usluge koje nemaju aktivnosti

EXEC MasineMaterijali.usluge_procedura
@godina = 2018 -- nema usluga
```

6.1 Mogući pozivi procedure

U godini 2019 odradjene su sledece usluge :

Usluga 1 sa nazivom Unutrašnje pranje i cenom 5000rsd ima sledece aktivnosti :

1. Prskanje tecnosti traje 2min.
2. Brisanje volana traje 3min.

Usluga 3 sa nazivom Rucno pranje i cenom 10000rsd ima sledece aktivnosti :

1. Usisavanje sedišta traje 5min.
2. Prebrisanje sedišta traje 7min.
3. Sušenje sedišta traje 3min.

Usluga 4 sa nazivom Zamena pneumatika i cenom 12000rsd nema definisanih aktivnosti!

Usluga 5 sa nazivom Poliranje i cenom 11000rsd ima sledece aktivnosti :

1. Stavljanje paste traje 3min.

Usluga 6 sa nazivom Balansiranje gume i cenom 250rsd ima sledece aktivnosti :

1. Skidanje gume sa tocka traje 3min.
2. Stavljanje tegova traje 1min.
3. Balansiranje traje 3min.
4. Montiranje gume traje 6min.

Usluga 9 sa nazivom Skidanje-nameštanje tocka i cenom 2500rsd ima sledece aktivnosti :

1. Odvrtnje šrafova traje 11min.
2. Skidanje tocka traje 1min.

Usluga 10 sa nazivom Brušenje i presovanje fleke i cenom 500rsd nema definisanih aktivnosti!

Usluga 10 sa nazivom Brušenje i presovanje fleke i cenom 500rsd nema definisanih aktivnosti!

6.2 Uspešan ispis procedure

U godini 2020 odradjene su sledece usluge :

Usluga 4 sa nazivom Zamena pneumatika i cenom 12000rsd nema definisanih aktivnosti!

Usluga 8 sa nazivom Cepovanje guma i cenom 3000rsd nema definisanih aktivnosti!

6.3 Slučaj kada usluge nemaju aktivnosti

U godini 2018 nije izvršena nijedna usluga

6.4 Slučaj kada nije izvršena nijedna usluga

Druga procedura

Procedura `masine_procedura` nam na osnovu prosleđene šifre zahteva vraća sve mašine određenih tipova korišćene u zahtevu. Kao parametar se prosleđuje šifra zahteva tipa `INTEGER`. U slučaju da se šifra zahteva ne nalazi u tabeli Se upotrebljava, to znači da zahtev još nije obrađen i ispisuje se odgovarajuća poruka. Ako zahtev jeste obrađen definiše se novi kursor koji nam vraća podatke o šifri tipa mašine i njenom nazivu. Unutar tog kursora se pravi još jedan koji nam vraća podatke o svim mašinama tog tipa. Podaci koji se vraćaju jesu redni broj mašine, oznaka, snaga motora, nivo buke, napajanje, težina i godina proizvodnje. Nakon ispisa svake mašine tog tipa prelazi se na sledeći tip i tako sve dok se ne pokriju svi korišćeni tipovi. Na kraju se konačno zatvaraju i dealociraju oba kursora i dobijamo listu podataka koja sadrži sve tipove sa svojim mašinama.

```
EXEC MasineMaterijali.masine_procedura
@idzahteva = 2 --nije obradjen
EXEC MasineMaterijali.masine_procedura
@idzahteva = 4 --obradjen
```

6.5 Mogući pozivi procedure `masine_procedure`

Iskoriscen je tip masine : 3. Dizalica za teretne gume

1. TX12002 sa snagom 0,nivoom buke 10dB, napajanjem OV, tezinom 120kg, proizvedena 2016. godine
2. TX12004 sa snagom 0,nivoom buke 10dB, napajanjem OV, tezinom 122kg, proizvedena 2017. godine
3. TX13003 sa snagom 0,nivoom buke 10dB, napajanjem OV, tezinom 116kg, proizvedena 2013. godine
4. TX22006 sa snagom 0,nivoom buke 10dB, napajanjem OV, tezinom 127kg, proizvedena 2016. godine

Iskoriscen je tip masine : 4. Alat za demontažu guma

1. TRK60001 sa snagom 0,nivoom buke 10dB, napajanjem OV, tezinom 50kg, proizvedena 2017. godine
2. TRS70006 sa snagom 0,nivoom buke 10dB, napajanjem OV, tezinom 53kg, proizvedena 2018. godine

6.6 Slučaj kada je zahtev obrađen

Zahtev jos nije obradjen u radnom nalogu!

6.7 Slučaj zahtev nije obrađen

TRIGERI

Prvi trigger

Radni_nalog_trigger se okida nakon upisa u tabelu radni_nalog. Služi za proveru da li je datum kreiranja radnog naloga kasniji od datuma podnošenja zahteva. U slučaju da je raniji treba izvršiti ažuriranje tabele radni_nalog i vrednost upisanog polja za datum kreiranja postaviti na datum koji je za dan kasniji od datuma podnošenja odgovarajućeg zahteva. To je postignuto pomoću DATEADD funkcije gde za datum radnog naloga koji smo prethodno izdvojili iz tabele inserted stavljamo vrednost datuma podnošenja odgovarajućeg zahteva povećanog za jedan dan. Pored toga se i ispisuje odgovarajuća poruka.

```
INSERT INTO MasineMaterijali.radni_nalog(id_radnog_naloga, id_zahteva_za_ostale_usluge_rn, datum_kreiranja, zakazan_termin, broj_novih_guma) --dobar datum
VALUES(11,11,'2021-03-24',' 12:30:00',0)
INSERT INTO MasineMaterijali.radni_nalog(id_radnog_naloga, id_zahteva_za_ostale_usluge_rn, datum_kreiranja, zakazan_termin, broj_novih_guma) --los datum
VALUES(11,11,'2021-02-05',' 12:30:00',0)
```

7.1 Primeri u kojima će se pokrenuti trigger

	id_radnog_naloga	id_zahteva_za_ostale_usluge_m	datum_kreiranja	zakazan_termin	broj_novih_guma
1	1	1	2020-03-22	09:30:00.0000000	0
2	7	2	2020-07-09	16:30:00.0000000	0
3	5	3	2020-07-05	10:00:00.0000000	0
4	3	4	2020-03-21	09:30:00.0000000	0
5	8	5	2020-08-08	17:00:00.0000000	0
6	2	6	2020-03-15	14:30:00.0000000	0
7	6	7	2020-03-12	09:00:00.0000000	4
8	9	8	2020-10-20	11:30:00.0000000	0
9	4	9	2020-04-27	08:00:00.0000000	0
10	10	10	2020-08-17	14:30:00.0000000	0
11	11	11	2021-03-24	12:30:00.0000000	0

7.2 Primer upisa dobrog datuma

Datum kreiranja radnog naloga ne može biti raniji od datuma podnošenja zahteva!
Datum će biti podešen na sledeći mogući.

7.3 Ispis poruke kod unosa lošeg datuma

	id_radnog_naloga	id_zahteva_za_ostale_usluge_m	datum_kreiranja	zakazan_termin	broj_novih_guma
1	1	1	2020-03-22	09:30:00.0000000	0
2	7	2	2020-07-09	16:30:00.0000000	0
3	5	3	2020-07-05	10:00:00.0000000	0
4	3	4	2020-03-21	09:30:00.0000000	0
5	8	5	2020-08-08	17:00:00.0000000	0
6	2	6	2020-03-15	14:30:00.0000000	0
7	6	7	2020-03-12	09:00:00.0000000	4
8	9	8	2020-10-20	11:30:00.0000000	0
9	4	9	2020-04-27	08:00:00.0000000	0
10	10	10	2020-08-17	14:30:00.0000000	0
11	11	11	2021-03-06	12:30:00.0000000	0

7.2 Primer upisa lošeg datuma

Drugi trigger

Trigger `materijal_trigger` se poziva u slučaju modifikovanja vrednosti obeležja `kolicina_materijala` u tabeli `Materijal`. Njegova uloga je da zabeleži staro i novo stanje količine materijala i koliko njega je dodato/oduzeto. Za potrebe ovog triggera smo kreirali i pomoćnu tabelu `evidencija_mat` koja ima obeležja `id_materijala`, `stara_kolicina`, `nova_kolicina` i `datum_izmene`. U slučaju ažuriranja vrednosti kolone `kolicina_materijala` Treba zabeležiti staru i novu količinu u pomoćnim promenljivama koje će svoje vrednosti dobiti iz tabela `inserted` i `deleted`. U slučaju da su stara i nova količina iste ispisaće se poruka da je količina nepromenjena. Ako nisu iste onda se vrši dalja provera. U slučaju da je nova količina manja od stare ispisaće se poruka koliko materijala je oduzeto, a u slučaju da je nova količina veća od stare ispisaće se poruka koliko materijala je dodato na stanje. Ako je pak nova količina jednaka 0 ispisaće se poruka da nema više tog materijala na skladištu.

```
Kolicina materijala Bakar je nepromenjena!
```

7.3 Primer upisa iste količine materijala

```
Dodato je 30 jedinica materijala Bakar u skladiste.
```

7.4 Primer upisa veće količine materijala

```
Oduzeto je 20 jedinica materijala Bakar sa skladista.
```

7.5 Primer upisa manje količine materijala

	id_materijala	stara_kolicina	nova_kolicina	datum_izmene
1	1	50	80	2023-06-04 16:14:16.317
2	1	80	60	2023-06-04 16:14:27.163
3	1	60	40	2023-06-04 16:14:48.260
4	1	40	100	2023-06-04 16:14:54.193
5	1	100	30	2023-06-04 16:14:59.940
6	1	30	50	2023-06-04 16:15:05.410

7.6 Izgled tabele `evidencija_mat`

```
UPDATE MasineMaterijali.materijal
SET kolicina_materijala=50
WHERE id_materijala=1
```

7.6 Primer modifikacije kada se aktivira trigger

ZAKLJUČAK

Na kraju ovog projekta možemo sumirati i zaključiti da je uspešno implementirana podšema MašineMaterijali i da su realizovana složena ograničenja sistema. Koristeći DDL definisana je sama struktura kao i tipovi podataka i deklarativna ograničenja svih tabela. Pomoću DML-a unešeni su svi potrebni podaci kako bi se omogućio dalji rad na projektu. Realizovano je 5 upita koji koriste neke od kompleksnijih funkcija SQL-a i ažurno i tačno prikazuju sve tražene podatke. Kreirane su po 2 funkcije i procedure koje nam na osnovu prosleđenih podataka vraćaju željene rezultate dobijene izvršavanjem unutrašnje logike. Na kraju su definisana i 2 trigera koji nakog određenih radnji efektivno izvršavaju svoje namerene funkcije i pomažu nam u održavanju integriteta samog sistema. Zaključno sa ovom dokumentacijom ovaj projekat predstavlja jednu uspešnu realizaciju zamišljene ideje i potencijalno osnovu za dalje unapređenje baze podataka Vulkanizer.