



UNIVERZITET U SARAJEVU
ELEKTROTEHNIČKI FAKULTET SARAJEVO
ODSJEK RAZVOJ SOFTVERA



Projekat iz predmeta Skladišta podataka

Tema: Apoteka

Profesor:

Vanr.prof.dr. Samir Omanović, dipl.ing.el

Studenti:

Dajana Prašo

Neira Piranić

Emina Palalić

Denis Destović

Sarajevo, august 2022.

Sadržaj

Uvod	3
Analiza	4
Skladišta podataka (data warehouse)	5
3.1 Pristup od dna prema gore (eng. bottom-up)	5
3.2 Pristup od vrha prema dnu (eng. top-bottom)	6
ETL proces	8
UML reprezentacija relacijske baze podataka	9
5.1 ER dijagram	10
5.2 Use Case dijagram	11
5.3 Activity dijagram	12
Implementacija skladišta podataka	13
Kreiranje kocke	21
8. Analiza	26
9. Zaključak	29
10. Literatura	29

1. Uvod

Ovaj projekat je obrađen u sklopu predmeta “Skladišta podataka” stručnog studija Razvoj softvera na Elektrotehničkom fakultetu Univerziteta u Sarajevu. Kroz ovaj projekat bit će obrađena upotreba analitičke obrade podataka primjenjene nad organizovanim skladištem podataka apotekarske ustanove sa prodajnim jedinicama na teritoriji Bosne i Hercegovine.

Skladišta podataka (eng. Datawarehouse) su centralna spremišta integrisanih podataka iz jednog ili više različitih izvora koja omogućavaju brzo i jednostavno pretraživanje velikih količina podataka.

2. Analiza

Kao i svaka apoteka, i naša apoteka ima upravitelja. Upravitelj apoteke odlučuje o nabavci i prodaji robe, zapošljava i nadgleda rad radnika, ali i vodi brigu o apoteci općenito. Firma tj. prodavač od kojeg upravitelj naručuje proizvode, upravitelju izdaje fakturu, a dobavljač mu dostavlja narudžbu. Apotekari se bave poslovima kao što su preuzimanje narudžbe, odlaganje proizvoda u magacin, slaganje i raspoređivanje proizvoda po policama, ažuriranje podataka o proizvodima. Oni također informišu kupce o samim proizvodima i cijenama istih te ih prodaju kupcima i izdaju im račune.



Slika 1. Apotekar raspoređuje proizvode

3. Skladišta podataka (data warehouse)

Skladišta podataka (eng. Data warehouse) su centralna spremišta integrisanih podataka iz jednog ili više različitih izvora koja omogućavaju brzo i jednostavno pretraživanje velikih količina podataka. Skladištenje podataka je složen proces. Može se normalizirati ili denormalizirati ovisno o potrebama. U pozadini može se raditi o relacijskoj bazi podataka ili višedimenzionalnoj bazi podataka. Podaci u skladištu podataka često se mijenjaju te se skladišta podataka često moraju usredotočiti na određenu aktivnost ili entitet.

Postoje dva osnovna pristupa u arhitekturi skladišta podataka:

1. Od dna prema gore
2. Od vrha prema dolje

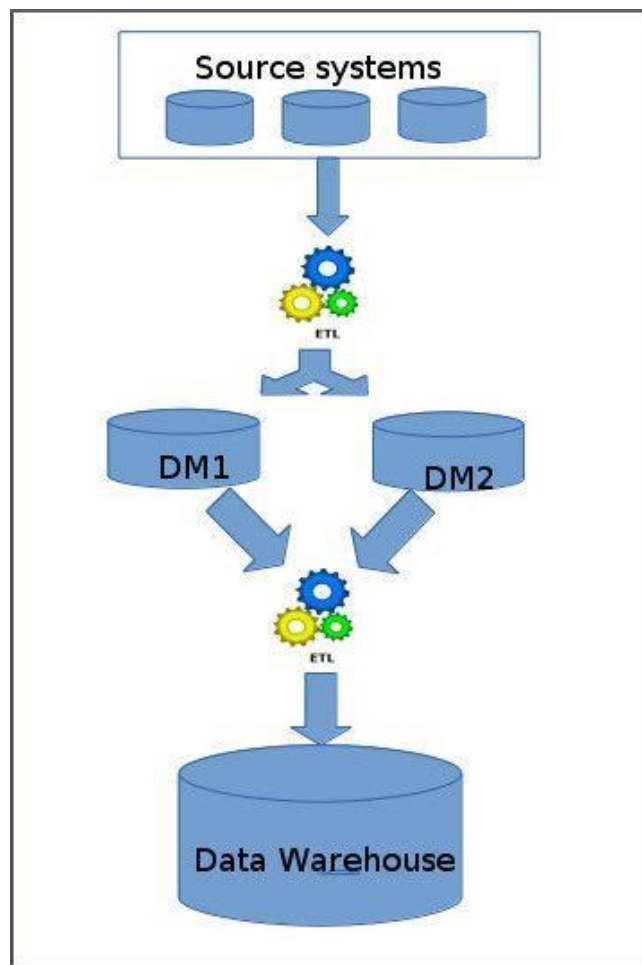
3.1 Pristup od dna prema gore (eng. bottom-up)

Začetnik ovog pristupa je Ralph Kimball, te prema njegovoj teoriji skladišta podataka moraju biti razumljiva i brza. Kod ove arhitekture, područno skladište sadrži detaljne i historijske podatke. Prednost je brza izgradnja sistema, ali mana mu je da problem integracije sistema dolazi u centar scene te je potrebno planirati zajednički razvojni okvir koji obuhvata sve dijelove sistema skladištenja podataka.

Ovaj pristup dohvaća i transformira sve podatke, ali ih prvo sprema u područna skladišta pa tek ona u skladište podataka. Za razliku od pristupa od vrha prema dolje, ovaj pristup je brži jer ne sprema odmah sve podatke u skladište već ih dijeli.

Koraci realizacije ovog pristupa su:

1. Tok podataka u pristupu odozdo prema gore počinje od ekstrakcije podataka iz različitih izvornih sistema u područje faze gdje se obrađuju i učitavaju u podatkovne martove koja upravljaju specifičnim poslovnim procesima.
2. Nakon osvježavanja podatkovnih martova, trenutni podaci se još jednom izdvajaju u području scene i primjenjuju se transformacije za kreiranje podataka u strukturi podatkovnog marta. Podaci su ekstraktovani iz Data Mart-a u scensko područje se agregiraju, sumiraju i tako dalje učitavaju u EDW(Enterprise data warehouse), a zatim se stavljaju na raspolaganje krajnjem korisniku za analizu i omogućavaju kritične poslovne odluke.

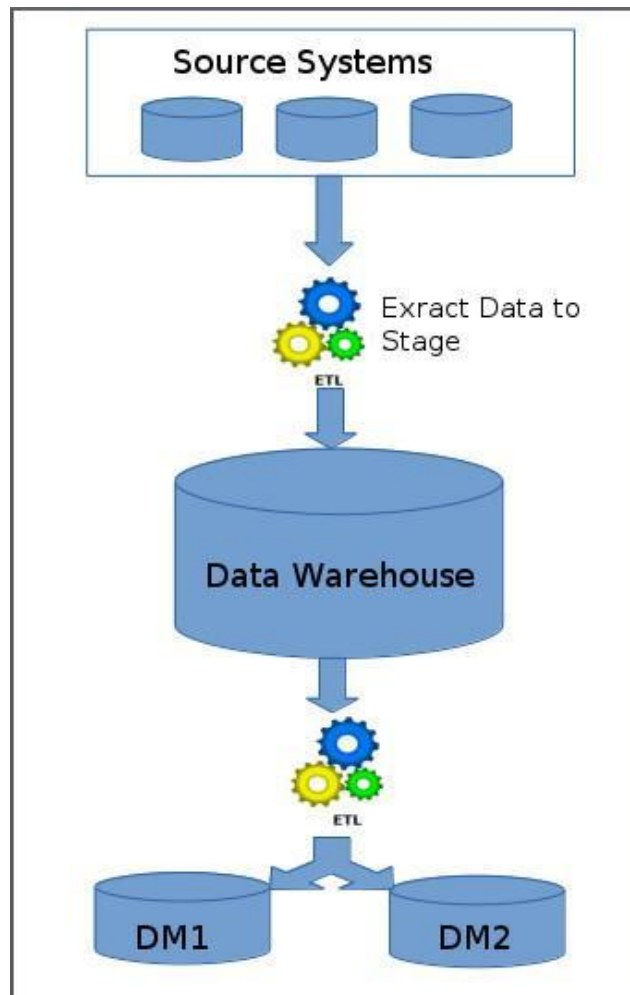


Slika 2. Prikaz pristupa od dna ka vrhu

3.2 Pristup od vrha prema dnu (eng. top-bottom)

Začetnik ovog pristupa je Bill Inmon. Prema njegovoj teoriji u skladištu podataka prevladavaju normalizirani modeli podataka te za pojedine grupe pitanja koje će krajnji korisnici često postavljati, kreiraju se mali dimenzijski modeli. Skladište podataka sadrži najdetaljnije, a područno skladište podataka sumirane podatke. No, ovakva izgradnja sistema je dugotrajna i skupa.

Pristup od vrha prema dnu je prikazan na slici 3.



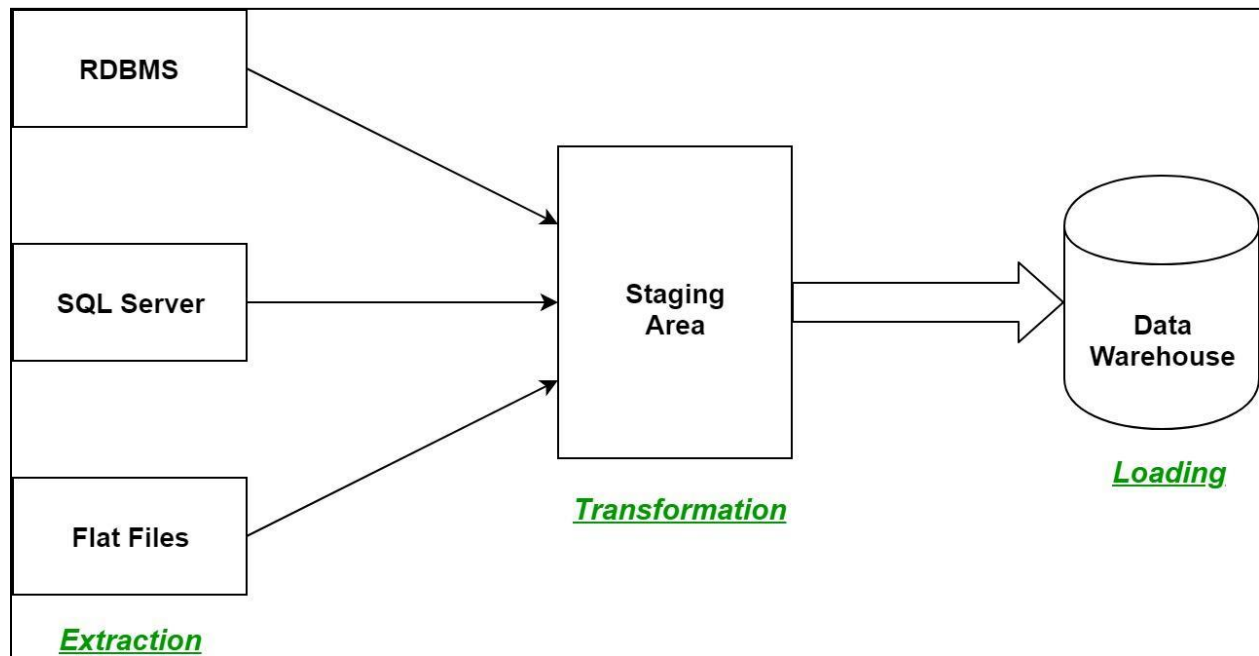
Slika 3. Prikaz pristupa od dna ka vrhu

Koraci realizovanja pristupa od vrha prema dnu:

1. Podaci se izvlače iz različitih izvornih sistema. Ekstrakti se učitavaju i validiraju u području scene. Validacija je potrebna da bi se osiguralo da su ekstraktovani podaci tačni i tačni. Možete koristiti ETL alate ili pristup za izdvajanje i guranje u skladište podataka.
2. Podaci se izvlače iz skladišta podataka redovno u sceni. U ovom koraku ćete primijeniti različite tehnike agregacije, ljetne obrade na ekstrahirane podatke i učitavanje natrag u skladište podataka.
3. Kada se agregacija i objedinjavanje dovrše, različita tržišta podataka izdvajaju te podatke i primjenjuju još neke transformacije kako bi se napravila struktura podataka kako je definirana od strane podatkovnih martova.

4. ETL proces

ETL je proces u skladištu podataka i označava izdvajanje, transformaciju i učitavanje. To je proces u kojem ETL alat izdvaja podatke iz različitih sistema izvora podataka, transformiše ih u scenskom području, a zatim ih konačno učitava u sistem skladišta podataka.



Slika 4. Etl proces

Kao što vidimo na slici 3 etl proces se sastoji od tri koraka: ekstrakcija, transformacija i učitavanje. Sada ćemo objasniti svaki od ova tri koraka.

1. Ekstrakcija

Prvi korak ETL procesa je ekstrakcija. U ovom koraku ekstraktuju se podaci iz različitih izvornih sistema koji mogu biti u različitim formatima kao što su relacijske baze podataka, bez SQL-a, XML-a i ravnih datoteka u scensko područje. Važno je izdvojiti podatke iz različitih izvornih sistema i prvo ih pohraniti u scenu, a ne direktno u skladište podataka, jer su ekstraktovani podaci u različitim formatima i mogu biti oštećeni. Zato ga učitavanje direktno u skladište podataka može oštetiti i vraćanje će biti mnogo teže. Zbog toga je ovo jedan od najvažnijih koraka ETL procesa.

2. Transformacija

Drugi korak ETL procesa je transformacija. U ovom koraku, skup pravila ili funkcija se primjenjuje na ekstrahirane podatke kako bi se pretvorili u jedinstveni standardni format. Može uključivati sljedeće procese/zadatke:

1. Filtriranje – učitavanje samo određenih atributa u skladište podataka
2. Čišćenje – popunjavanje NULL vrijednosti nekim zadanim vrijednostima
3. Razdvajanje – razdvajanje jednog atributa na više atributa
4. Sortiranje – sortiranje tuple-ova na osnovu nekog atributa (obično ključ-atribut)

3. Učitavanje

Treći i posljednji korak ETL procesa je učitavanje. U ovom koraku, transformirani podaci se konačno učitavaju u skladište podataka. Ponekad se podaci ažuriraju učitavanjem u skladište podataka vrlo često, a ponekad se to radi u dužim, ali pravilnim intervalima. Brzina i period učitavanja isključivo zavise od zahtjeva i variraju od sistema do sistema.

ETL alati: Najčešće korišteni ETL alati su Hevo, Sybase, Oracle Warehouse builder, CloverETL i MarkLogic

5. UML reprezentacija relacijske baze podataka

UML(Unified Modeling Language) je vizuelni jezik za specificiranje, konstruisanje i dokumentovanje artefakata sistema. Uloga UML-a je specifikacija, vizualizacija, konstrukcija i dokumentacija razvoja softvera.

UML se sastoji od različitih tipova dijagrama kao što su: er dijagram, dijagram toka, dijagram aktivnosti, dijagram slučaja i slično. U sljedećoj sekciji ćemo predstaviti er dijagram namijenjen za naše skladište podataka.

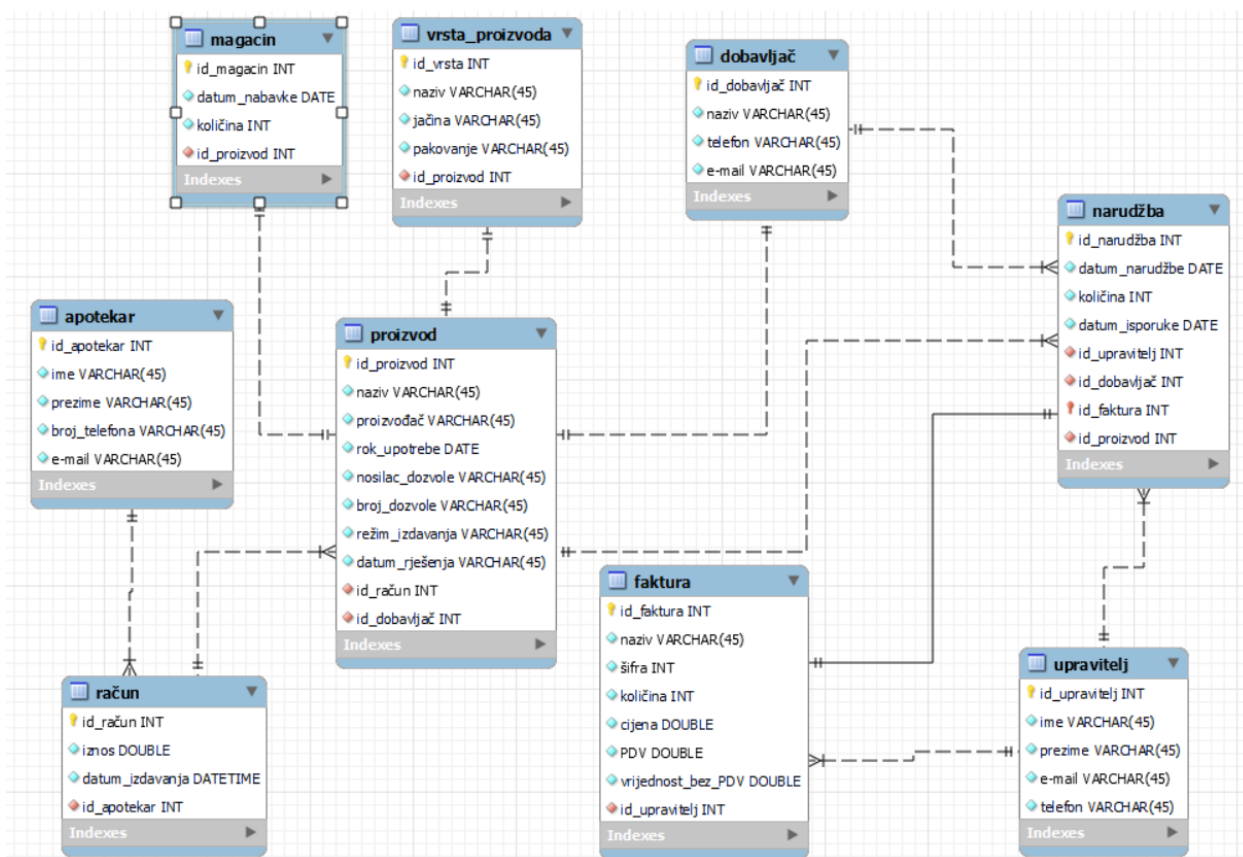
5.1 ER dijagram

Entity-Relationship (ER) model ili model objekti-veze (MOV) predstavlja detaljan logički prikaz podataka preko skupa entiteta, njihovih atributa i međusobnih veza.

Osnovni elementi modela su:

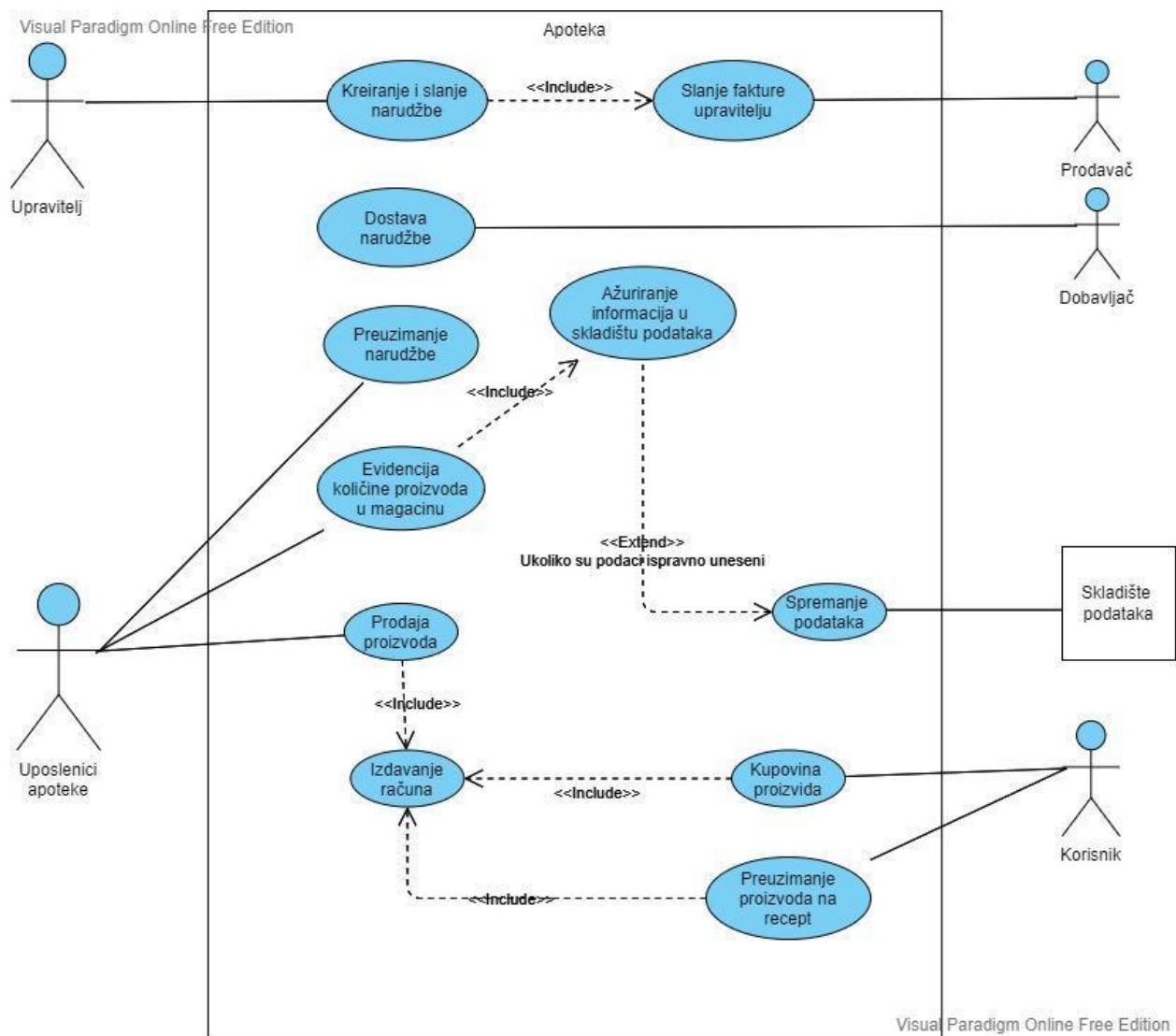
1. entiteti (objekti)
2. atributi
3. veze

ER dijagram za naše skladište podataka sastoji se od devet entiteta, tj. dimenzija, a to su: Upravitelj, Dobavljač, Apotekar, Narudžba, Faktura, Magacin, Proizvod, Vrsta proizvoda i Račun. Svaka od ovih entiteta ima svoje atribute. Najvažniji atribut u svakoj tabeli je id, tj. jedinstveni broj pomoću kojeg razlikujemo sve podatke i koji nam služi za povezivanje podataka u skladištu.



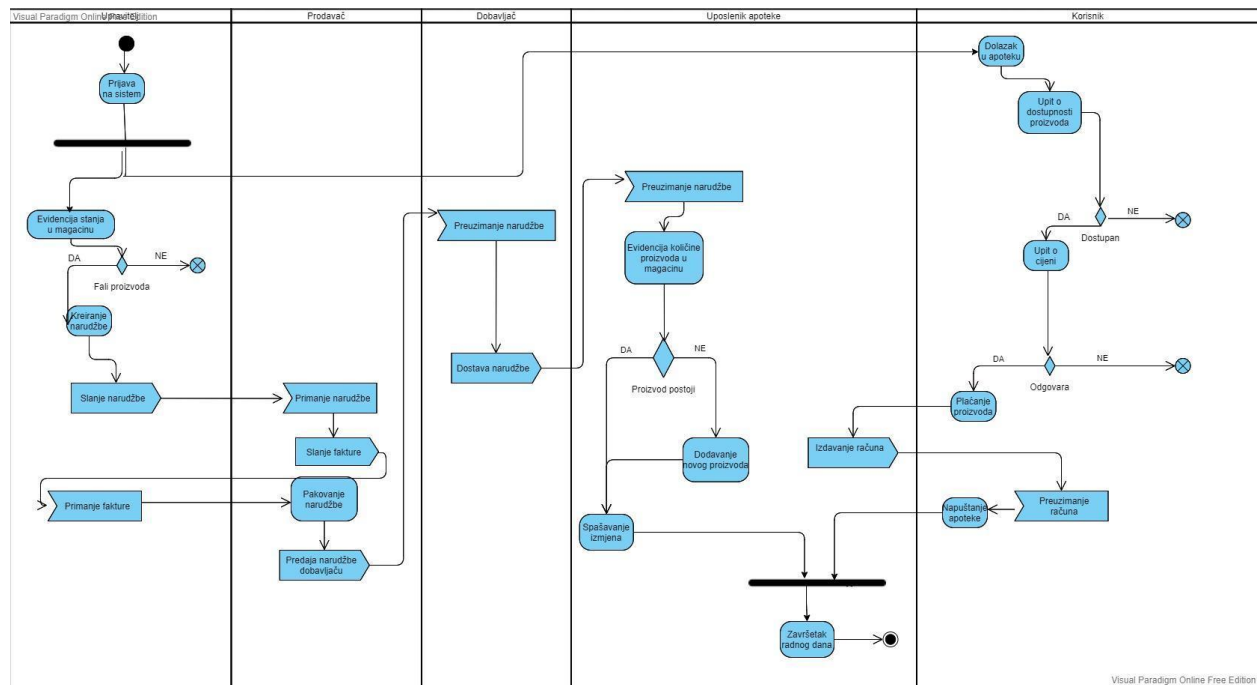
Slika 5. ER dijagram skladišta podataka za apoteku

5.2 Use Case dijagram



Slika 6. Use Case dijagram

5.3 Activity diagram



Slika 7. Activity dijagram

6. Implementacija skladišta podataka

Implementacija skladišta podataka se sastoji od slijedećih koraka:

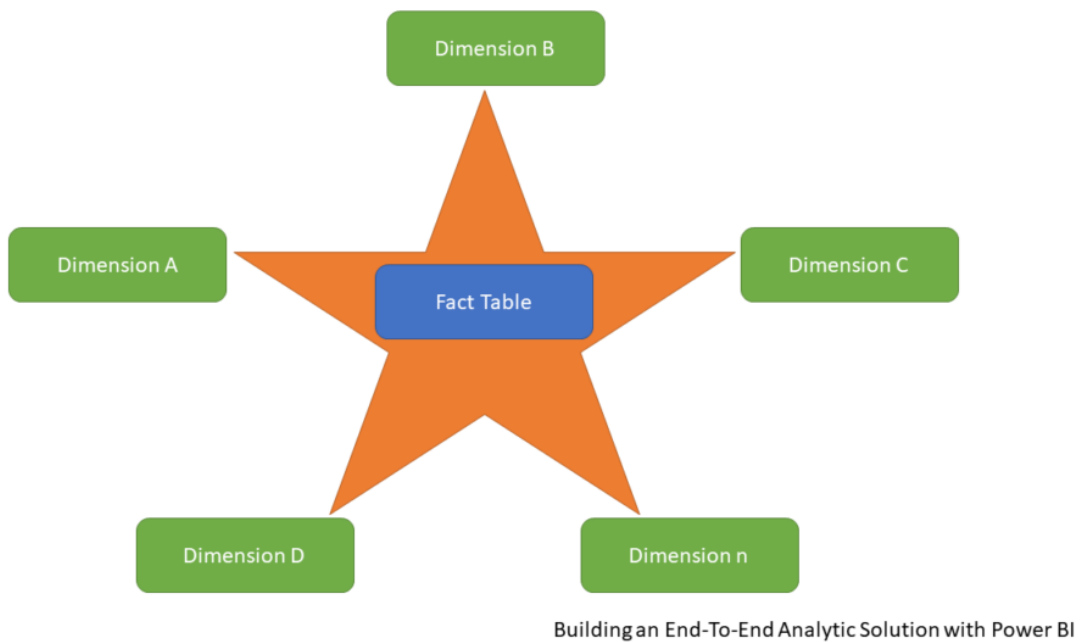
1. Proces izgradnje započinje definiranjem korisničkih zahtjeva.
2. Analiziraju se zahtjevi i postojeći podaci u operacijskoj bazi.
3. Izgrađuje se logički model skladišta podataka.
4. Podaci se logički transformiraju prilikom izvlačenja iz transakcijske baze i učitavanja u skladište podataka.
5. Podaci u skladištu podataka se modeliraju u dimenzijski model podataka pogodan za izvođenje miliona upita istovremeno.
6. Podaci se zatim fizički dimenzioniraju (pročišćavaju i pojednostavljaju) što osigurava razumijevanje pojedinih podataka.
7. Osigurava se učitavanje novih podataka u dogovorenim/potrebnim vremenskim intervalima.
8. Izgradnja skladišta je gotova, ali je skladište potrebno nadgledati i upravljati njime, te ako
9. se javi potreba i rekonstruisati/modificirati (djelimično ili potpuno).

Naše skladište podataka će koristiti zvjezdastu shemu. Zvjezdasta shema podrazumijeva da je svaka dimenzija za analizu u potpunosti denormalizirana, dok pahuljičasta shema nije u potpunosti denormalizirana. Denormalizacija se radi kako bi broj tablica koje se dohvaćaju u korisničkom upitu bio što manji, te kako bi upiti radili brže. Dimenzijsko oblikovanje jest tehnika logičkog oblikovanja koja ima za cilj predstaviti podatke u jednostavnom, intuitivnom obliku koji dopušta vrlo učinkovit pristup (pregled). Dakle, dimenzijski model pokušava riješiti nedostatak relacijskog modela (u analitičkom smislu) i pružiti krajnjim korisnicima brz i jednostavan pregled podataka.

Dimenzijska tablica odgovara subjektu/objektu koji sudjeluje u procesu koji se prati u 22 skladištu podataka. Dimenzijske tablice daju kontekst, odnosno objašnjavaju i opisuju činjenice pohranjene u činjeničnoj tablici. Dimenzijske tablice nisu normalizirane.

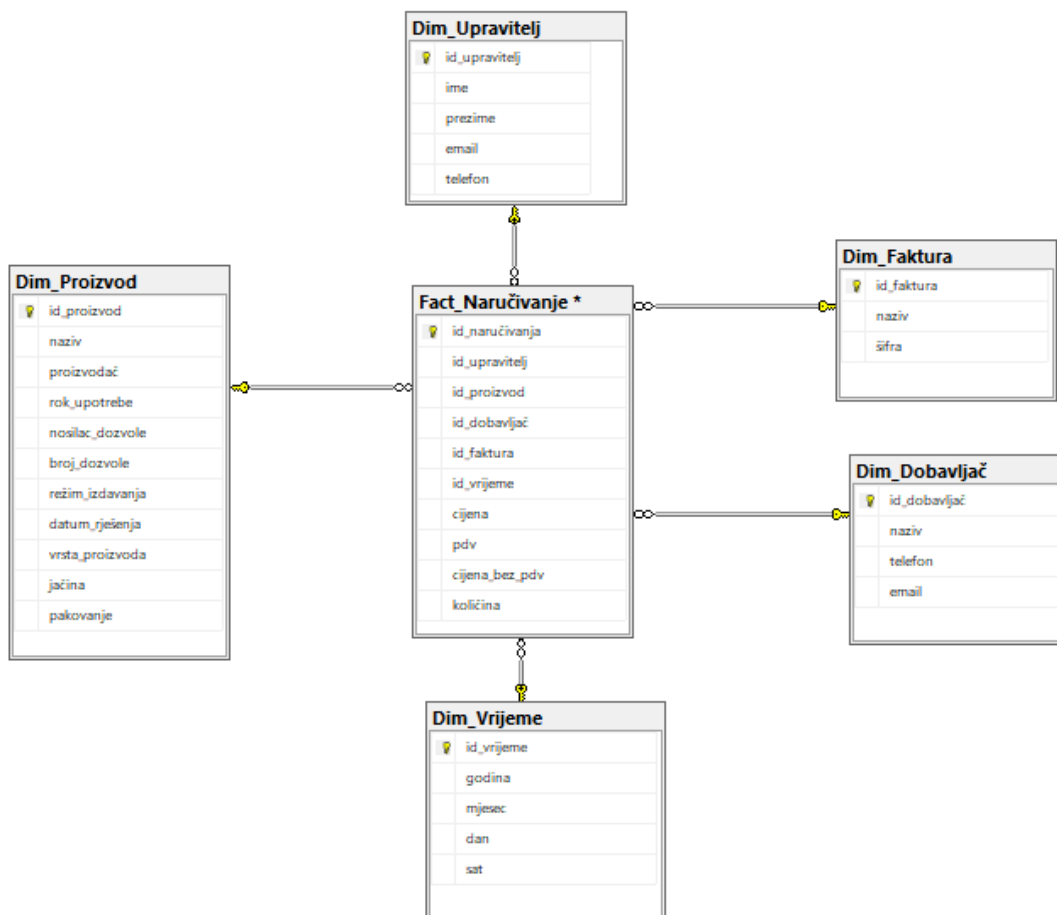
U središtu zvjezdastog modela je činjenična tablica koja odgovara procesu koji se prati u skladištu podataka. U nekom skladištu podataka možemo naravno imati N zvjezdastih spojeva i N činjeničnih tablica. Činjenična tablica sadrži dvije skupine numeričkih atributa:

- Ključevi dimenzijskih tablica
- Mjere (eng. measures)



Slika 8. Kreiranje star sheme

Naša DWH se sastoji od dvije star sheme koje će biti predstavljene u nastavku kao i krajnje skladište podataka. Atribute primarne, surogatne ključeve te njihove veze možete veoma lahko analizirati sa dijagramima.

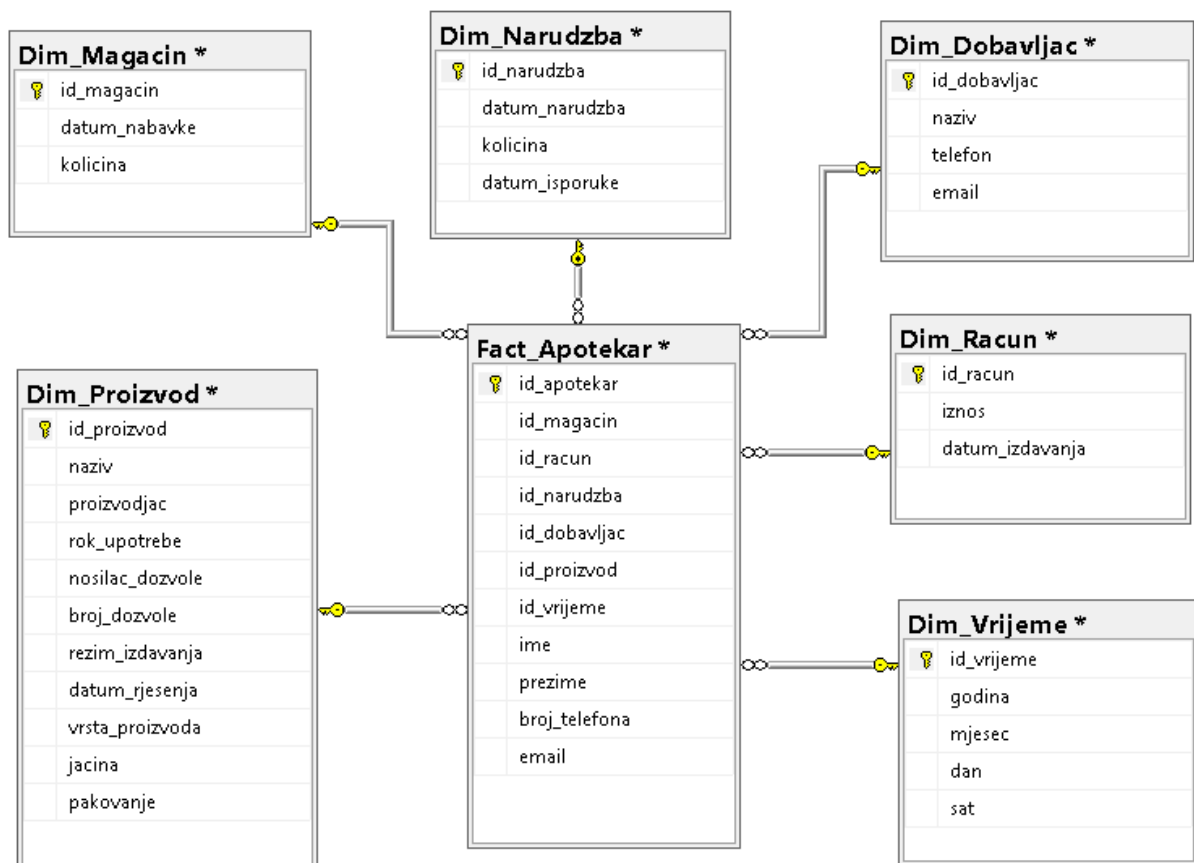


Slika 9. Star schema 1 - Fact_Naručivanje

Naša prva star schema sastoji se od tabele činjenica Fact_Naručivanje koja oko sebe grupiše dimenzije upravitelj, proizvod, faktura, dobavljač i vrijeme.

Tabela činjenica naručivanje nam pored toga daje informacije o cijeni naručenog proizvoda, kao i o količini koja je naručena.

Razlog zašto smo izabrali prethodno navedene dimenzije je taj što smo htjeli uvijek da imamo podatke o upravitelju koji je izvršio narudžbu, kao i proizvodu koji je naručen. Da bi podaci bili još tačniji i pouzdaniji, dodali smo dimenziju faktura i dimenziju vrijeme, koje nam tačno prikazuju malo detaljnije informacije o narudžbi, dok podatke u dobavljaču čuvamo u dimenziji dobavljač.

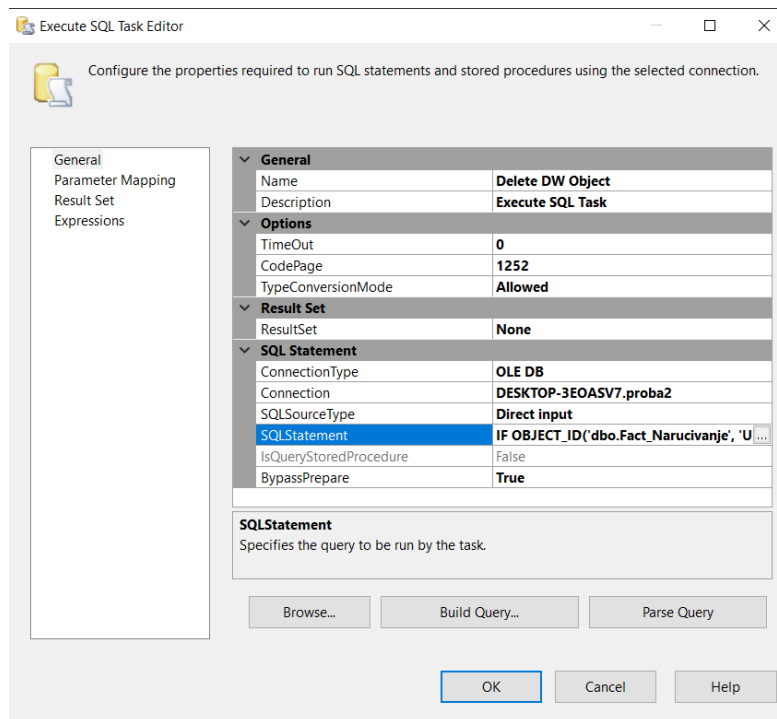


Slika 10. Star schema 2 - Fact_Apotekar

Druga star schema sastoji se od tabele činjenica Fact_Apotekar koja je okružena tabelama dimenzija: narudžba, dobavljač, račun, vrijeme, proizvod i magacin.

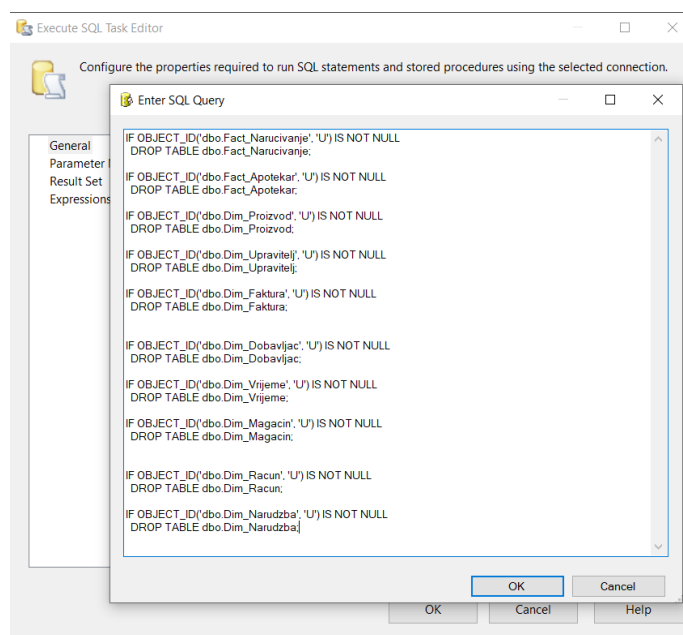
Tabela činjenica apotekar nam omogućava da u bilo kojem trenutku imamo uvid o proizvodima te njihovom stanju u magacinu. Apotekara smo spojili sa tabelama narudžba i dobavljač kako bi svaki podatak bio ažuran te poznato vrijeme dolaska određenog proizvoda. Pored toga dodate su još dvije tabele, vrijeme i račun kako bi nam iznos i datum izdavanja bio poznat.

Dodali smo Execute SQL Task u Control flow:



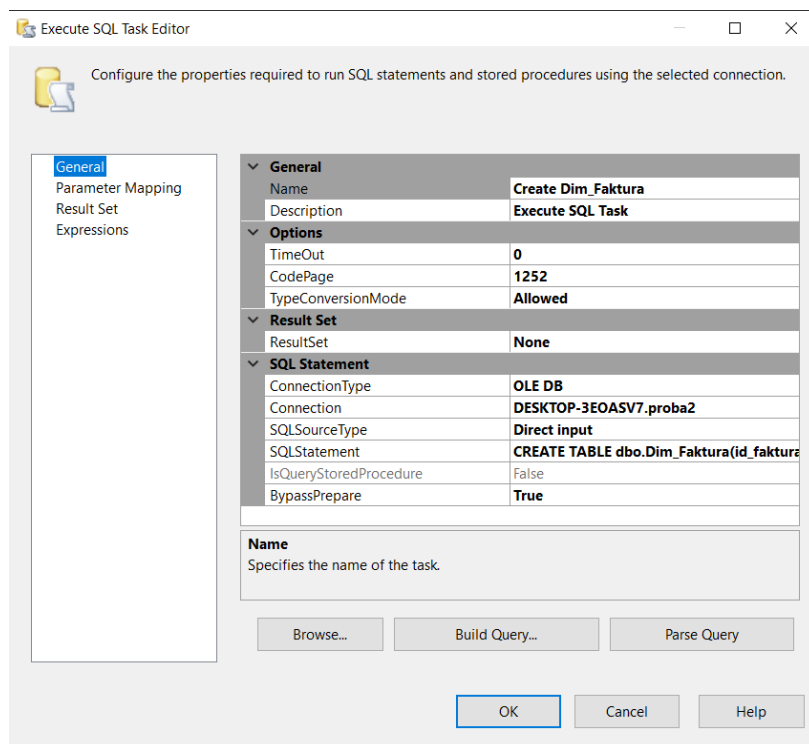
Slika 11. Execute SQL Task

Na slici ispod vidimo da smo napisali SQLStatement koji će izvršiti provjeru da li tabele postoje te ukoliko već postoje, izvršit će se njihovo brisanje.

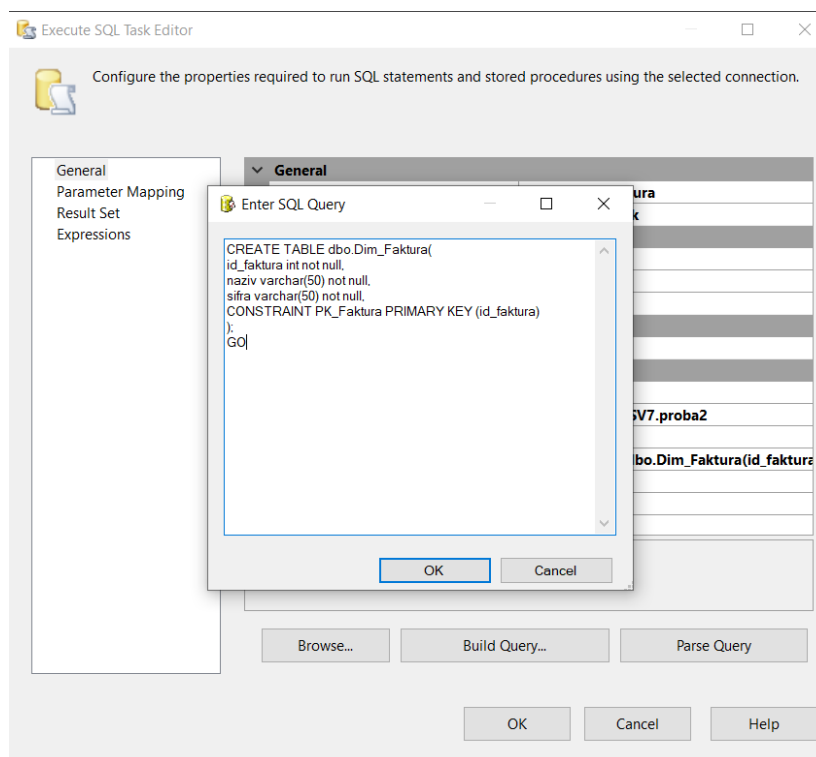


Slika 12. SQL statement - provjera da li tabele postoje

Naredni korak je kreiranje svih dimenzija Star schema. Primjer za Dim_Faktura:

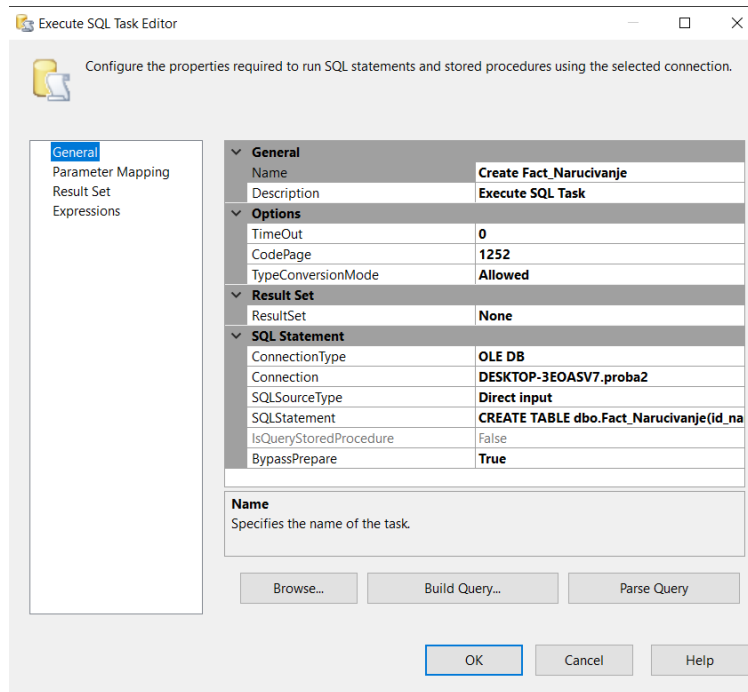


Slika 13. Execute SQL Task - Dim_Faktura

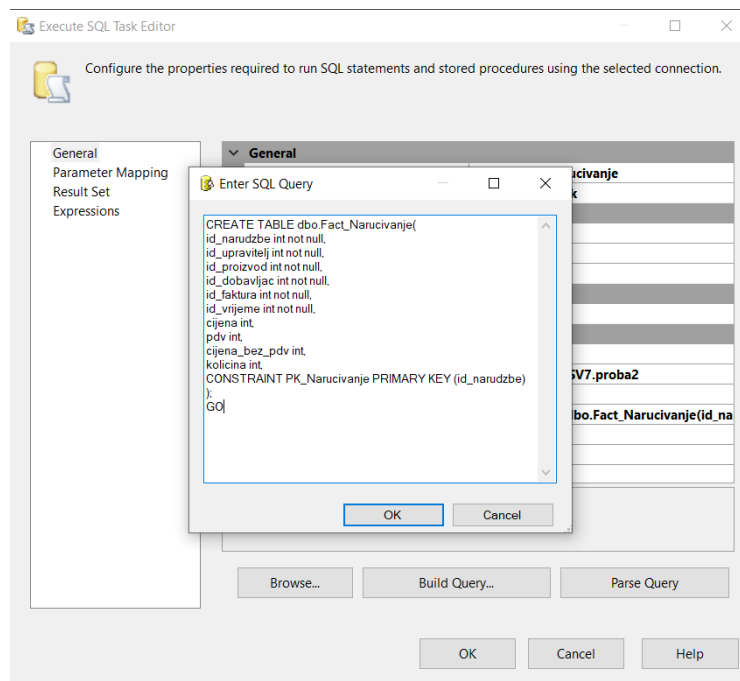


Slika 14. Kreiranje Dim_Faktura tabele

Zatim smo na isti način kreirali tabele činjenica te alter tabele činjenica za dodavanje stranih ključeva. Primjer za Fact_Naručivanje:

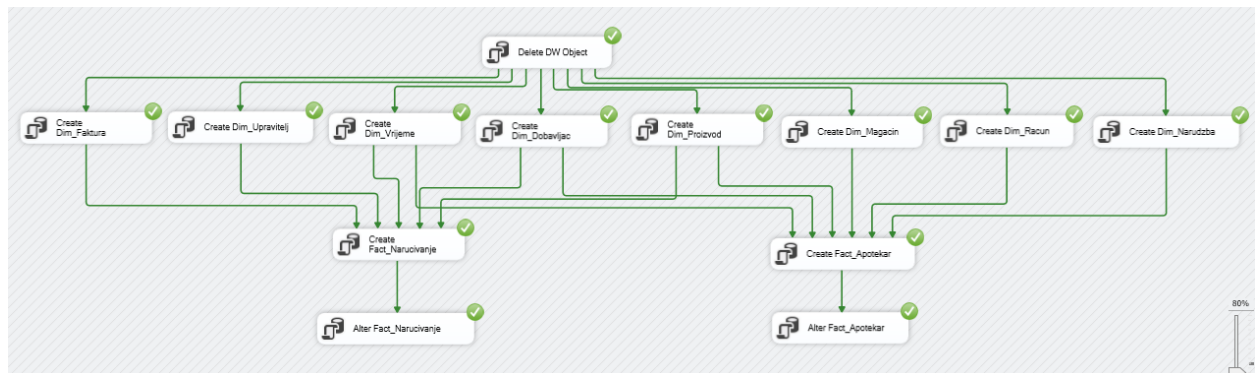


Slika 15. Execute SQL Task - Fact_Naručivanje



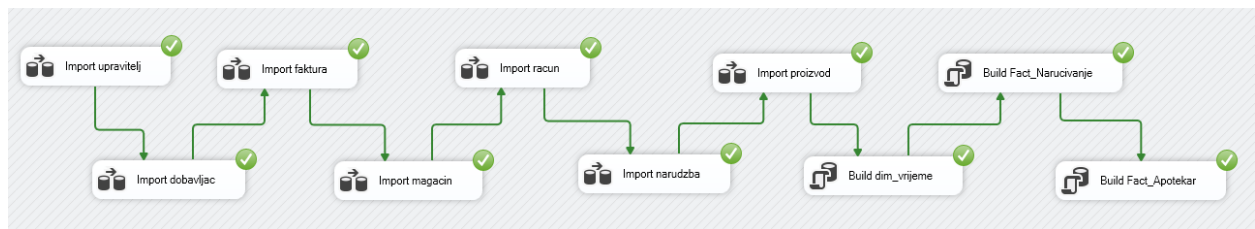
Slika 16. Kreiranje Fact_Naručivanje tabele

Control flow ETL procesa kojim se kreira skladište:



Slika 17. Control flow

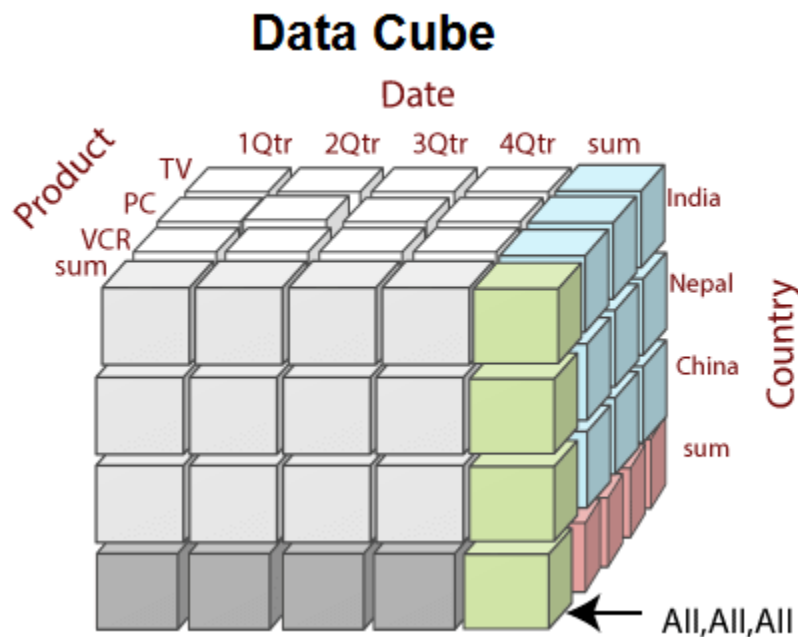
Data flow ETL procesa kojim se importuju podaci u skladište:



Slika 18. Data flow

7. Kreiranje kocke

OLAP je pristup analizi i izvještavanju koji omogućuje korisniku da lako i selektivno izdvaja i pregledava podatke s različitih stajališta temeljeno na multidimenzijskoj strukturi podataka zvanom kocka (engl. cube). Kocka tipično odgovara jednom zvjezdastom spoju i tipično se unutar OLAP sistema definira uzevši postojeći zvjezdasti spoj kao izvor podataka.



Slika 19. Kocka

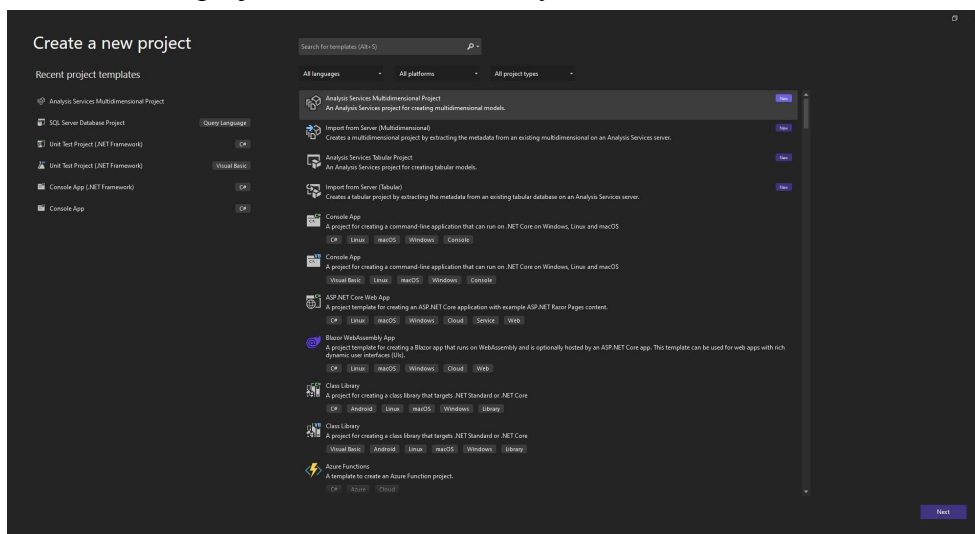
OLAP kocka predstavlja multidimenzionalni niz podataka, odnosno skupinu ćelija podataka raspoređenih po dimenzijama. Ako govorimo o OLAP kocki podrazumijeva se da postoje tri dimenzije kojima se definiraju podaci. Multidimenzionalni niz podataka može imati i više od tri dimenzije, ali takav niz je teško predočiti i naziva se hiper-kocka.

Operacije koje se mogu izvoditi nad kockom su sljedeće:

1. Roll-up / Drill up – operacija koja se sastoji od agregiranja odnosno sakupljanja podataka u kocki čime ovu operaciju možemo nazvati agregiranjem.
2. Roll-down / Drill down – operacija koja omogućuje korisniku analizu podataka prema većoj razini detalja u hijerarhiji koncepata.
3. Slice and Dice – operacije koje pokreću korisnici, a uključuje interaktivnu navigaciju kroz pogled stranice, kroz specifikaciju kriški (slices) 5 pomoću rotacija i agregiranja i detaljiziranja. Postoji mogućnost horizontalnog i vertikalnog analiziranja podataka.
4. Pivot – operacija rotacije koja omogućava rotiranje osi, mijenjajući određene dimenzije kako bi se stekao drugi pogled na multidimenzionalnu kocku.

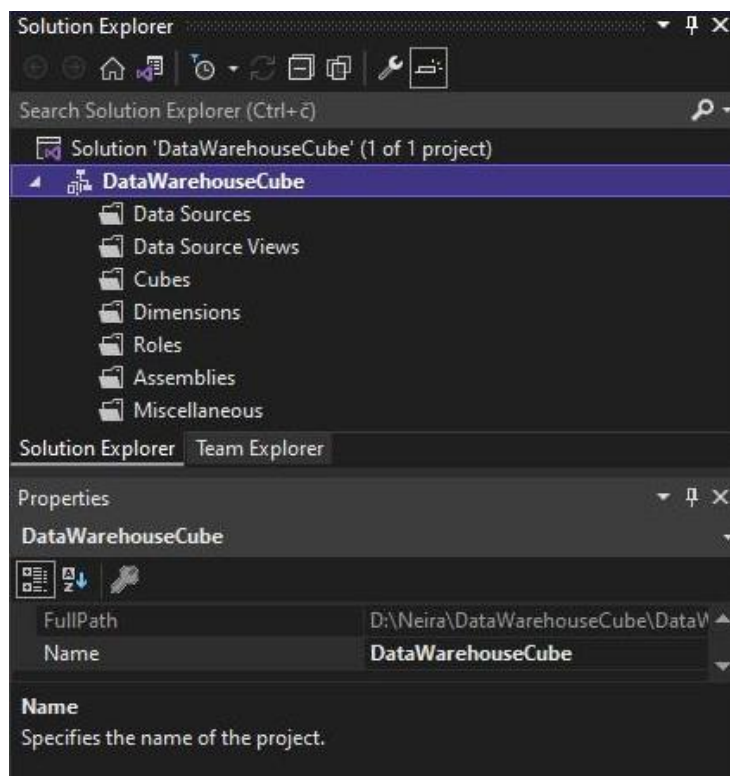
Kreiranje kocke vršimo u Microsoft Visual Studio-u.

1. Kreiramo novi projekat i odaberemo Analysis Services Multidimensional Project:



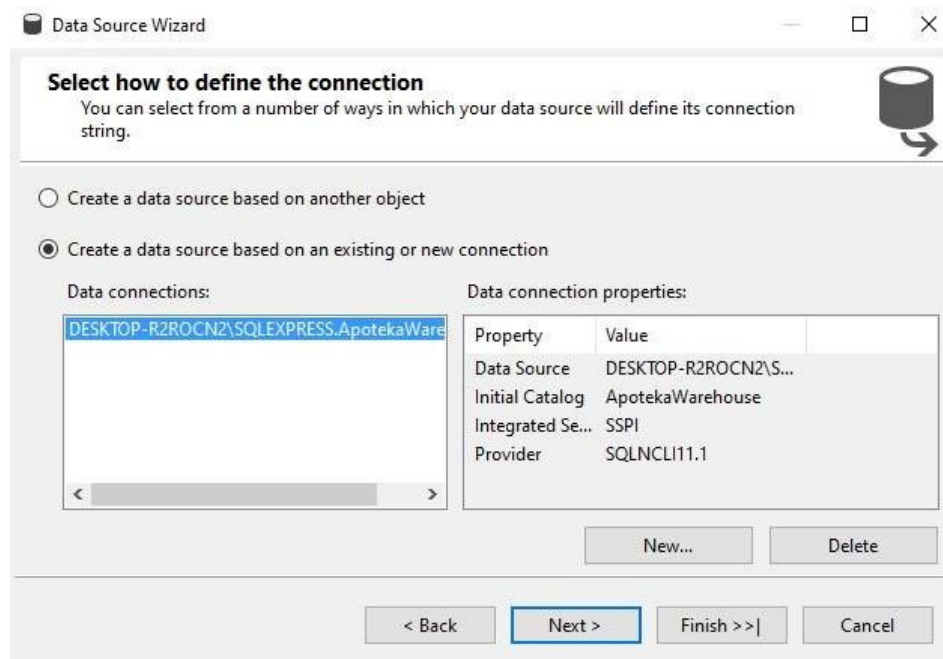
Slika 20. Kreiranje novog projekta u Visual Studiu

2. Kreiramo Data Source: Desni klik na Data Sources - New Data Source



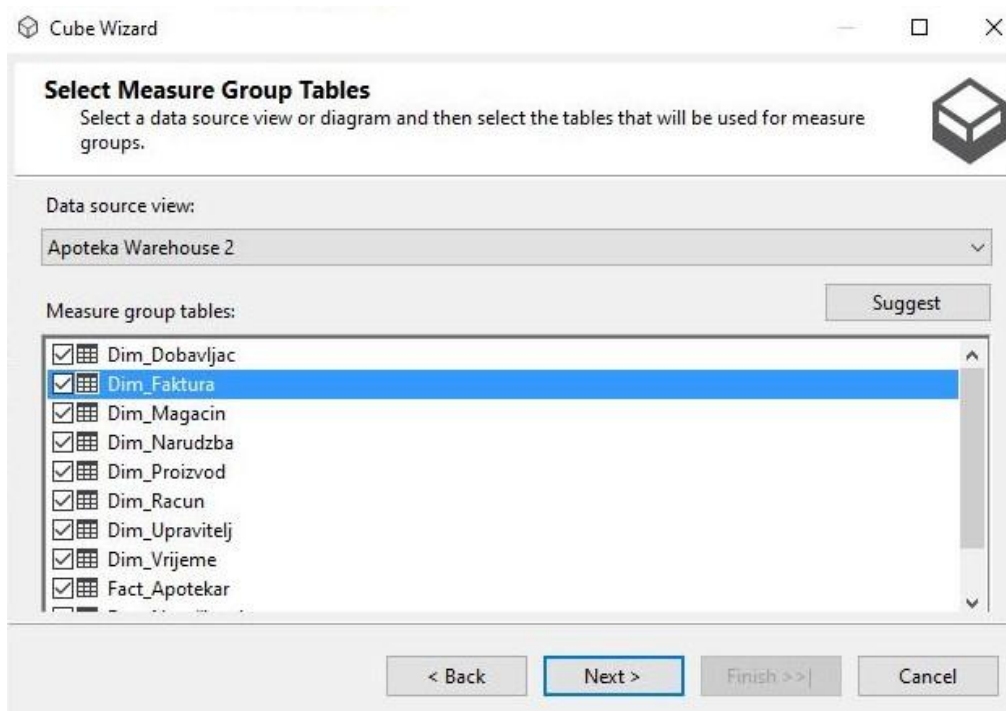
Slika 21. Prikaz lokacije kreiranja Data Source (Solution Explorer)

3. Konektujemo se na server i odaberemo bazu s kojom želimo da manipuliramo
- 4.

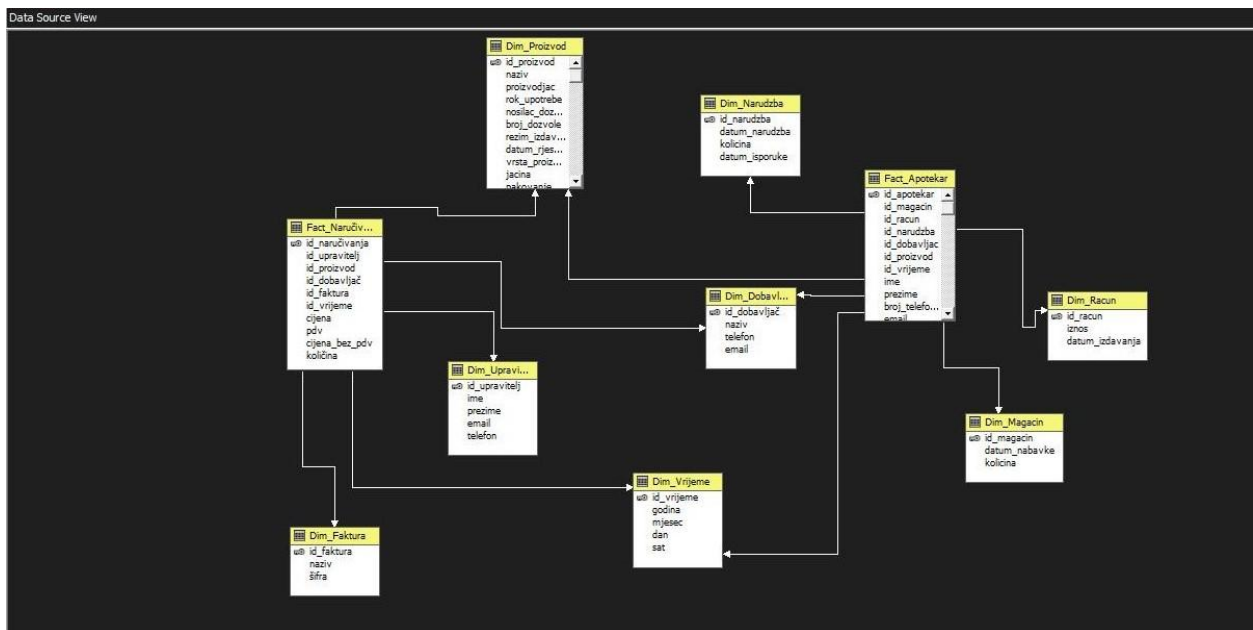
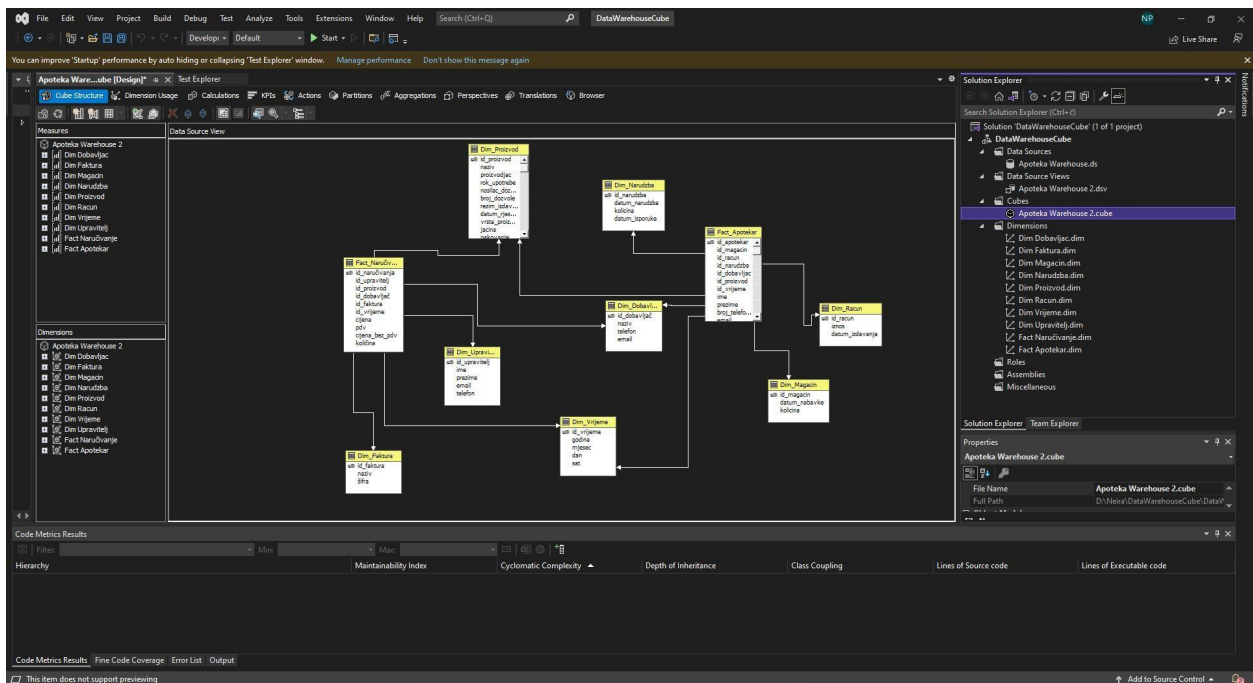


Slika 22. Kreiranje data source-a

5. Nakon što smo kreirali Data Source, potrebno je kreirati Data Source View: Desni klik na Data Source View - New Data Source View. Odabiremo fact tabele, i dimenzije povezane sa datim fact tabelama.
6. Kreiranje kocke: Desni klik na cubes - New Cube. Za kreiranje kocke nam je potreban Data Source View.



Slika 23. Kreiranje kocke



Slika 24. i 25. Prikaz kocke

8. Analiza

Kreirali smo par upita kako bi analizirali sam sistem:

```
SELECT naziv,
       proizvođač,
       rok_upotrebe,
       nosilac_dozvole,
       režim_izdavanja,
       datum_rješenja
FROM   apoteka2.proizvod
WHERE  rok_upotrebe BETWEEN '2022-09-05' AND '2023-07-14'
```

100 %

Results Messages

	naziv	proizvođač	rok_upotrebe	nosilac_dozvole	režim_izdavanja	datum_rješenja
1	LINEX	LEK farmacevtska družba d.d., Slovenija	2023-03-21	Novartis BA d.o.o.	Rp - Lijek se izdaje uz ljekarski recept	28.12.2020
2	FLUIMUKAN PLUS	LEK farmacevtska družba d.d., Slovenija	2023-07-14	Novartis BA d.o.o.	BRp - Lijek se izdaje bez ljekarskog recepta	10.06.2019
3	FLUIMUKAN za djecu	LEK farmacevtska družba d.d., Slovenija	2023-07-14	Novartis BA d.o.o.	BRp - Lijek se izdaje bez ljekarskog recepta	27.08.2019
4	FLUIMUKAN AKUT	LEK farmacevtska družba d.d., Slovenija	2023-07-14	Novartis BA d.o.o.	BRp - Lijek se izdaje bez ljekarskog recepta	09.09.2020

Slika 22. Upit koji nam prikazuje proizvode čiji je rok upotrebe između dva zadana datuma

```
SELECT p.naziv,
       p.proizvođač,
       r.iznos,
       r.datum_izdavanja
FROM   apoteka2.proizvod p, apoteka2.račun r
WHERE  r.iznos > 100
```

100 %

Results Messages

	naziv	proizvođač	iznos	datum_izdavanja
1	ACIPAN	LEK farmacevtska družba d.d., Slovenija	300	2022-04-01 00:00:00
2	LINEX	LEK farmacevtska družba d.d., Slovenija	300	2022-04-01 00:00:00
3	FLONIDAN	LEK farmacevtska družba d.d., Slovenija	300	2022-04-01 00:00:00
4	FLUIMUKAN PLUS	LEK farmacevtska družba d.d., Slovenija	300	2022-04-01 00:00:00
5	FLUIMUKAN za djecu	LEK farmacevtska družba d.d., Slovenija	300	2022-04-01 00:00:00
6	FLUIMUKAN AKUT	LEK farmacevtska družba d.d., Slovenija	300	2022-04-01 00:00:00

Slika 23. Upit koji nam prikazuje proizvode i račun čiji je iznos veći od 100

```

SELECT d.naziv,
       d.[e-mail],
       n.datum_narudžbe,
       n.količina,
       n.datum_isporuke
FROM apoteka2.dobavljač d, apoteka2.narudžba n
WHERE n.datum_narudžbe
      BETWEEN '01.02.2022' AND '03.02.2022'

```

	naziv	e-mail	datum_narudžbe	količina	datum_isporuke
1	EuroExpress	sektorkomercijale@euroexpress.ba	2022-02-01	100	2022-01-04
2	XExpress	info@x-express.ba	2022-02-01	100	2022-01-04
3	Express One	office@expressone.ba	2022-02-01	100	2022-01-04
4	EuroExpress	sektorkomercijale@euroexpress.ba	2022-02-02	200	2022-01-10
5	XExpress	info@x-express.ba	2022-02-02	200	2022-01-10
6	Express One	office@expressone.ba	2022-02-02	200	2022-01-10

Slika 24. Upit koji nam prikazuje dobavljače i narudžbe čiji je datum isporuke između dva zadana datuma

```

SELECT f.šifra,
       f.količina,
       f.cijena,
       f.PDV,
       f.vrijednost_bez_PDV,
       p.naziv,
       p.proizvođač,
       p.rok_upotrebe,
       p.režim_izdavanja
FROM apoteka2.faktura f
INNER JOIN apoteka2.proizvod p ON f.id_faktura = p.id_proizvod
ORDER BY f.količina ASC

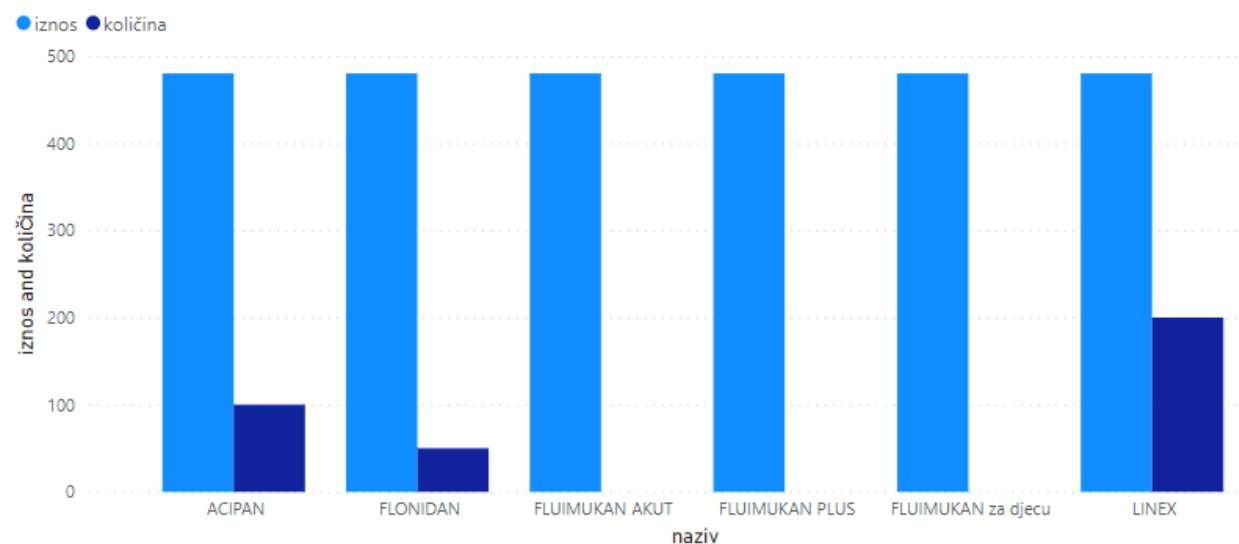
```

	šifra	količina	cijena	PDV	vrijednost_bez_PDV	naziv	proizvođač	rok_upotrebe	režim_izdavanja
1	7589	10	200	10	180	LINEX	LEK farmacevtska družba d.d., Slovenija	2023-03-21	Rp - Lijek se izdaje uz ljekarski recept
2	6374	20	200	10	180	ACIPAN	LEK farmacevtska družba d.d., Slovenija	2024-09-01	Rp - Lijek se izdaje uz ljekarski recept

Slika 25. Upit za koji smo koristili inner join između fakture i proizvoda

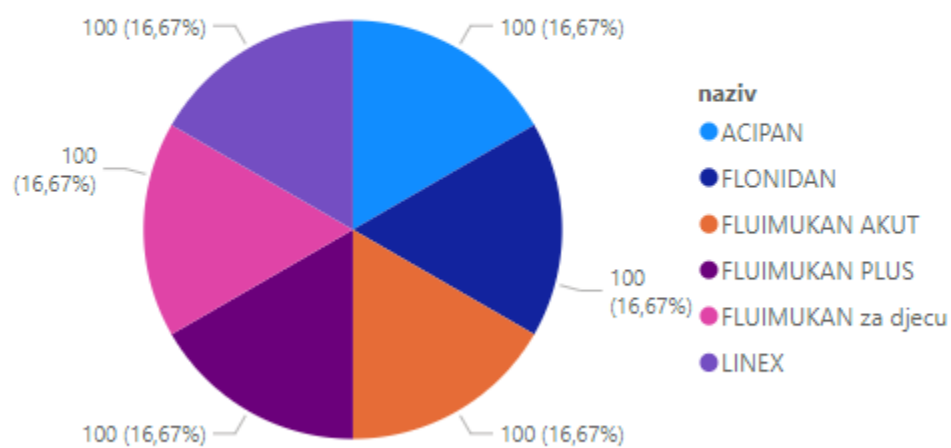
Za kreiranje izvještaja koristili smo alat Power BI, koji nam omogućava lako i efikasno prikazivanje raznih tipova izvještaja kao i vizualizaciju istih.

iznos and količina by naziv



Slika 26. Iznos i količina proizvoda po nazivu proizvoda

količina by naziv



Slika 27. Ukupna količina proizvoda po magacinu

9. Zaključak

Vođenje poslova “po sjećanju” nije dovoljno za stabilno poslovanje i rast. Poznavanje osnova statičke analize, ali i vladanje njom, menadžeru omogućava bolji uvid u stanje i procese na tržištu na kojima djeluje. Donošenje odluka u poslovanju, unatoč velikom broju dostupnih podataka nije ništa lakše nego prije. Potrebna je osnovna statička pismenost za razumijevanje i komuniciranje onoga što podaci govore. Zato je potrebno omogućiti što bolji, efikasniji i brži sistem upravljanja skladištem podataka kao i samo skladište podataka. Sve ovo bi trebalo dovesti do boljeg poslovanja kompanije, održavanja kvalitete proizvoda te većeg zadovoljstva klijenata, što i jeste glavni i osnovni cilj.

10. Literatura

1. <https://dwgeek.com/various-data-warehouse-design-approaches.html/>
2. <https://www.geeksforgeeks.org/etl-process-in-data-warehouse/>