

Sveučilište Jurja Dobrile u Puli
Fakultet informatike u Puli

Ivana Pleše

Australske prometne nesreće sa smrtnim ishodom 1989-2021

Seminarski rad

Pula, svibanj 2024.

Sveučilište Jurja Dobrile u Puli
Fakultet informatike u Puli

Ivana Pleše

Australske prometne nesreće sa smrtnim ishodom 1989-2021

Seminarski rad

JMBAG: 0303100674

Studijski smjer: Informatika

Kolegij: Skladišta i rudarenje podataka

Mentor: doc.dr.sc. Goran Oreški

Pula, svibanj 2024.



IZJAVA O AKADEMSKOJ ČESTITOSTI

Ja, dolje potpisana Ivana Pleše, ovime izjavljujem da je ovaj seminarski rad rezultat isključivo mogega vlastitog rada, da se temelji na mojim istraživanjima te da se oslanja na objavljenu literaturu kao što to pokazuju korištene bilješke i bibliografija. Izjavljujem da niti jedan dio seminarskog rada nije napisan na nedozvoljen način, odnosno da je prepisan iz kojega necitiranog rada, te da ikoji dio rada krši bilo čija autorska prava. Izjavljujem, također, da nijedan dio rada nije iskorišten za koji drugi rad pri bilo kojoj drugoj visokoškolskoj, znanstvenoj ili radnoj ustanovi.

Student

Pula, svibanj 2024.

SADRŽAJ

1.	POSLOVNA INTELIGENCIJA.....	1
2.	UVOD.....	4
3.	ODABIR SKUPA PODATAKA	5
3.1.	OSNOVNA ANALIZA PODATAKA.....	5
4.	RELACIJSKI MODEL PODATAKA	7
4.1.	IZRADA BAZE PODATAKA.....	7
4.2.	POPUNJAVANJE BAZE PODATAKA.....	8
4.3.	EER DIJAGRAM	9
5.	DIMENZIJSKI MODEL PODATAKA.....	11
5.1.	STAR SCHEMA.....	12
5.2.	ETL PROCES.....	13
5.2.1.	DIMENZIJSKA TABLICA STATE.....	13
5.2.2.	DIMENZIJSKA TABLICA HOLIDAY	14
5.2.3.	DIMENZIJSKA TABLICA HOUR.....	15
5.2.4.	DIMENZIJSKA TABLICA SPEED	15
5.2.5.	DIMENZIJSKA TABLICA VICTIM.....	16
5.2.6.	TABLICA ČINJENICA FACT_CRASH	17
6.	OLAP ANALIZA	19
6.1.	VIZUALIZACIJA PODATAKA.....	20
7.	ZAKLJUČAK	25
8.	LITERATURA	26

1. POSLOVNA INTELIGENCIJA

Poslovna inteligencija¹ (Business Intelligence, BI) predstavlja proces prikupljanja, analiziranja, tumačenja i prikazivanja podataka za podršku donošenju poslovnih odluka. Ovaj koncept integrira tehnologiju, alate i postupke za prikupljanje podataka iz mnogih izvora, njihovu transformaciju u korisne informacije i njihovu prezentaciju u svrhu podrške strateškom, taktičkom i operativnom donošenju odluka u poslovnom okruženju. Cilj poslovne inteligencije je podržati bolje poslovno odlučivanje i omogućiti organizacijama da postignu konkurentsku prednost. Poslovna inteligencija obuhvaća nekoliko ključnih komponenti:

Prikupljanje podataka: Proces prikupljanja podataka iz različitih izvora, uključujući interne izvore poput baza podataka, ERP sustava, CRM sustava i vanjske izvore poput tržišnih istraživanja, društvenih mreža i javnih baza podataka.

Skladištenje podataka: Prikupljeni podaci se pohranjuju u skladišta podataka (Data Warehouses) ili manje skladišne jedinice poznate kao data marts. Skladišta podataka omogućuju centraliziranu pohranu podataka i olakšavaju pristup i analizu podataka.

ETL proces: Extract, Transform, Load (ETL) procesi uključuju izdvajanje podataka iz izvora, njihovu transformaciju u odgovarajući format i učitavanje u skladište podataka. ETL procesi osiguravaju kvalitetu podataka i njihovu pripremljenost za analizu.

Analiza podataka: Korištenjem različitih analitičkih tehnika, poput statističke analize, rudarenja podataka (Data Mining), prediktivne analitike i strojnog učenja, organizacije mogu otkriti obrasce, trendove i korelacije u podacima.

Izvjestavanje i vizualizacija podataka: Izvještaji, nadzorne ploče (dashboards) i vizualizacije podataka predstavljaju rezultate analiza na razumljiv i interaktivan način. Alati poput Power BI, Tableau i Qlik omogućuju korisnicima da stvaraju dinamičke vizualizacije i prilagođene izvještaje. Što se tiče funkcioniranja procesa poslovne inteligencije, prvo je važna integracija i učitavanje podataka: Podaci iz izvornog sustava integrirani su i učitani u skladište podataka ili drugi repozitorij. Priprema podataka za analizu: Skupovi podataka organizirani su u analitičke modele podataka ili OLAP kocke kako bi se pripremili za analizu. Analitički upiti: BI analitičar, ostali analitičari i poslovni korisnici pokreću analitičke upite prema dostupnim podacima. Vizualizacija i izvještavanje: Rezultati upita ugrađeni su u podatke, vizualizaciju, nadzorne

¹ Stedman, C. „What is BI?“, <https://www.techtarget.com/searchbusinessanalytics/definition/business-intelligence-BI>

ploče, izvješća i online portale. Korištenje informacija: Rukovoditelji i radnici koriste informacije za bolje donošenje odluka i strateško planiranje.

Prednosti poslovne inteligencije i njene primjene donose brojne pogodnosti organizacijama, uključujući poboljšano donošenje odluka: BI omogućuje menadžerima i poslovnim korisnicima donošenje odluka temeljenih na podacima, smanjujući subjektivnost i povećavajući točnost i pouzdanost odluka. Povećanje učinkovitosti: Analiza podataka može identificirati neučinkovitosti i uska grla u poslovnim procesima, omogućujući organizacijama da optimiziraju resurse i povećaju operativnu učinkovitost. Bolje razumijevanje tržišta i kupaca: BI pruža duboke uvide u ponašanje i preferencije kupaca, omogućujući organizacijama da prilagode svoje proizvode i usluge kako bi bolje zadovoljile potrebe kupaca. Povećanje konkurentne prednosti: Organizacije koje uspješno primjenjuju BI mogu brže reagirati na tržišne promjene, identificirati nove poslovne prilike i ostati ispred konkurencije. Unapređenje financijske učinkovitosti: BI omogućuje detaljnu analizu financijskih podataka, što pomaže u identificiranju područja za smanjenje troškova i povećanje prihoda.

Poslovna inteligencija koristi širok spektar alata i tehnologija za prikupljanje, pohranu, analizu i vizualizaciju podataka: Alati za prikupljanje podataka: Alati poput Apache NiFi, Talend i Informatica omogućuju integraciju podataka iz različitih izvora. Sustavi za skladištenje podataka: Oracle, Microsoft SQL Server, Amazon Redshift i Google BigQuery su popularni sustavi za skladištenje podataka koji pružaju skalabilne i pouzdane platforme za pohranu podataka. ETL alati: Alati poput Apache NiFi, Talend, Informatica i Pentaho Data Integration olakšavaju ETL procese, omogućujući učinkovitu transformaciju i prijenos podataka u skladišta podataka. Analitički alati: Alati poput R, Python, SAS i SPSS pružaju napredne analitičke mogućnosti za rudarenje podataka, prediktivnu analitiku i strojno učenje. Alati za izvještavanje i vizualizaciju podataka: Power BI, Tableau, Qlik Sense i Looker omogućuju stvaranje interaktivnih izvještaja i vizualizacija koje olakšavaju razumijevanje složenih podataka.

Implementacija poslovne inteligencije nije bez izazova. Neki od glavnih izazova uključuju: Kvaliteta podataka: Loša kvaliteta podataka može dovesti do netočnih analiza i loših odluka. Osiguravanje točnosti, potpunosti i konzistentnosti podataka ključno je za uspjeh BI projekata. Integracija podataka: Integracija podataka iz različitih izvora može biti složena, posebno kada se radi o velikim količinama podataka ili različitim formatima podataka. Sigurnost i privatnost: Zaštita podataka i osiguravanje privatnosti podataka su kritični aspekti poslovne inteligencije,

posebno s obzirom na sve strože regulative o zaštiti podataka. Kultura podataka: Uvođenje BI rješenja zahtijeva promjenu kulture unutar organizacije, gdje svi zaposlenici trebaju biti educirani i motivirani za korištenje podataka u svakodnevnim odlukama. Unatoč izazovima, uspješna primjena BI-a može donijeti značajne koristi, omogućujući organizacijama da budu agilnije, efikasnije i bolje informirane.

2. UVOD

S obzirom da količina podataka generiranih svakodnevno raste eksponencijalno, upravljanje, analiza i vizualizacija tih podataka postali su ključni aspekti za donošenje informiranih poslovnih odluka. U sklopu kolegija "Skladišta i rudarenje podataka", cilj ovog projekta bio je primijeniti stečena teorijska znanja na samostalni praktičan primjer, koristeći stvarne podatke.

Projekt započinje preuzimanjem skupa podataka s Kaggle platforme, renomirane baze podataka koja nudi raznovrsne skupove podataka za analizu. Nakon preuzimanja, podaci su analizirani i prilagođeni u Jupyter Notebook okruženju, što je omogućilo detaljnu preglednost i pripremu podataka za daljnju obradu. Sljedeći korak uključivao je izradu baze podataka u MySQL-u putem Python skripti. Ova faza je obuhvatila kreiranje baza podataka i tablica te njihovo popunjavanje podacima iz CSV datoteke. Nakon toga, prikazani su Entitetsko-atributski dijagram (EER) i Star schema model, koji su osigurali vizualni prikaz strukture baze podataka i međusobnih odnosa među podacima. U Pentaho Data Integration alatima, kreirane su dimenzijske tablice i tablica činjenica, što je omogućilo strukturirano skladištenje podataka i njihovu pripremu za analitičke procese. Na kraju, korištenjem Microsoft Power BI alata, izrađene su vizualizacije koje omogućuju jednostavno i intuitivno pretraživanje, analizu i interpretaciju podataka.

Ovaj projekt integrira više faza procesa rada s podacima, od ekstrakcije i transformacije podataka, preko skladištenja, do analize i vizualizacije. Kroz ove korake, demonstrirana je važnost skladištenja podataka, pravilne analize te vizualizacije kao završne faze koja omogućuje krajnjim korisnicima donošenje utemeljenih odluka na temelju analiziranih podataka.

3. ODABIR SKUPA PODATAKA

Podatci za ovaj projekt preuzeti su sa službene web stranice Kaggle, pod nazivom “Australian Fatal Road Accident 1989-2021”.² U opisu podataka stoji: “Australaska baza podataka o smrtnim slučajevima u prometnim nesrećama pruža osnovne pojedinosti o smrtnim slučajevima u prometnim nesrećama u Australiji koje policija svaki mjesec prijavljuje državnim i teritorijalnim tijelima za sigurnost na cestama. Smrtni slučajevi na cestama iz posljednjih mjeseci su preliminarni i serija je podložna reviziji.” Nakon detaljnog proučavanja ovih podataka, izabrani su jer sadrže sve potrebno za napraviti ovaj projekt. CSV datoteka sadrži više od 54 tisuće redaka i 23 stupca, što je dovoljno podataka za daljnje korake. Važno je da ima i vremensku dimenziju, od godine do detaljnijih stupaca koji opisuju koji dan je u tjednu, koje vrijeme je, je li noć ili dan i radi li se o blagdanskom periodu. Također, važni su kvantitativni i kvalitativni podatci koji također ovdje postoje. Nedostajućih vrijednosti nema previše, no svakako će biti eliminirane u daljnjoj izradi. Imena svih stupaca prvotne csv datoteke su: Crash ID, State, Month, Year, Dayweek, Time, Crash Type, Bus Involvement, Heavy Rigid Truck Involvement, Articulated Truck Involvement, Speed Limit, Road User, Gender, Age, National Remoteness Areas, SA4 Name 2016, National LGA Name 2017, National Road Type, Christmas Period, Easter Period, Age Group, Day of week i Time of day.

3.1. OSNOVNA ANALIZA PODATAKA

Osnovna analiza podataka provedena je u Jupyteru kako bi uopće bilo moguće raditi sa preuzetim podacima. Prvo su učitani podatci iz csva u dataframe pomoću importanog pandas. Korišten je `data = pd.read_csv(PATH...)`. Zamjenjene su prazne vrijednosti sa NaN vrijednosti. Zatim je bitno učitati pregled prvih 5 redaka svakog stupca kako bismo vidjeli je li sve dobro namješteno zasad (`print(data.head())`). Zatim je potrebno učitati sve podatke radi analize kako bi dobili ispis dimenzija skupa podataka (`data.shape`), čime je dobiven rezultat od 52843 redaka i 23 stupaca. Sljedeći korak bio je ispis imena stupaca i nedostajućih

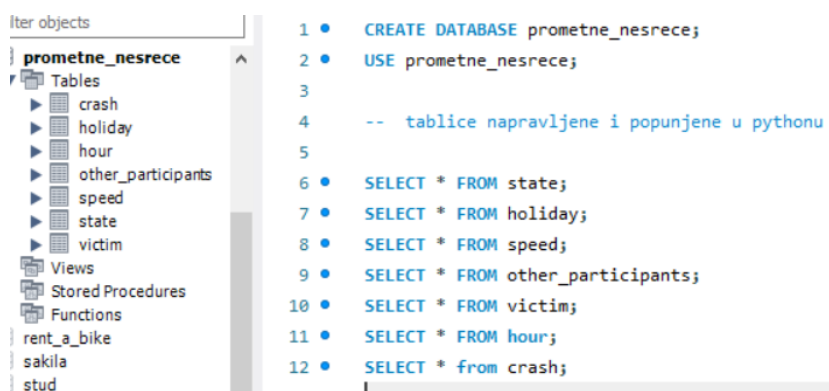
² Australian Fatal Road Accident 1989-2021; <https://www.kaggle.com/datasets/deepcontractor/australian-fatal-car-accident-data-19892021>

vrijednosti. Slijedi ispis broja jedinstvenih vrijednosti po stupcima, ispis tipova podataka po stupcima (data.dtypes) odnosno analiza kvantitativnih i kvalitativnih podataka, brisanje tri stupca koji se neće koristiti (SA4 Name 2016, National LGA Name 2017, National Road Type) , dodavanje novog stupca Penalty Price (dodatno izrađen stupac sa random brojevima od 30 do 400 sa korakom od 10), ponovni pregled svih stupaca te zatim spremanje u novu csv datoteku. Također, skup je podijeljen na dva dijela, jedan od 80% podataka i drugi od 20%. Pregledani su i tipovi svih podataka u pojedinom stupcu te je napokon datoteka bila spremna za raditi sa njom. Po završetku analize csv datoteka sa kojom će se izraditi projekt sastoji se od idućih stupaca: Crash ID, Penalty Price, Crash Type, State, Month, Year, Day week, Time, Crash Type, Bus Involvement, Heavy Rigid Truck Involvement, Articulated Truck Involvement, Speed Limit, Road User, Gender, Age, Christmas Period, Easter Period, Age Group, Day of week, Time of day.

4. RELACIJSKI MODEL PODATAKA

4.1. IZRADA BAZE PODATAKA

Nakon analize skupa podataka, idući korak bio je stvoriti smisleni relacijski model koji bi omogućio učinkovito skladištenje i obradu podataka. Za stvaranje baze podataka napisana je Python skripta povezana sa MySQL programom. MySQL je relacijski sustav za upravljanje bazama podataka (RDBMS) koji se koristi za pohranu i upravljanje podacima, a izabran je zbog svoje učinkovitosti, pouzdanosti i široke primjene u industriji. Analizom podataka identificirane su ključne relacije i definirane kardinalnosti među njima. Na temelju ovih analiza, stvoren je konceptualni model od osam relacija koje su definirane u skladu s analiziranim podacima. Svaka tablica je dizajnirana tako da podržava specifične attribute i veze. Konkretno, izrađena je baza podataka pod imenom `prometne_nesrece`, a u njoj su izrađene tablice: `crash`, `state`, `speed`, `victim`, `hour`, `other_participants` i `holiday`. (Slika 1). Svi stupci iz csv datoteke poredani su u attribute tih tablica, te su dodana ograničenja na tablicama poput PRIMARY KEY i FOREIGN KEY kako bi se moglo povezati entitete.



Slika 1: Prikaz baze u MySQL-u

U prvoj tablici; `state`, stvoreni su atributi `id` i `name` koji će biti povezan sa csv stupcem `State`. Sljedećoj tablici `hour`, stvoreni su atributi `id`, `month` koji će biti povezan sa stupcem `Month`, atribut `year` sa stupcem `Year`, atribut `dayweek` sa stupcem `Dayweek`, atribut `time` sa stupcem `Time`, atribut `time_of_day` sa stupcem `Time of Day` i atribut `day_of_week` sa stupcem `Day of Week`. Tablica `holiday` stvorena je sa atributima `id`, `christmas_period` i `easter_period` koji će biti povezani sa csv stupcima `Christmas Period` i `Easter Period`. Tablica `speed` sadrži

atribute id i speed_limit koji će biti povezan sa stupcem Speed Limit. Tablici victim stvoreni su atributi id, road_user koji će biti spojen sa stupcem Road User, gender sa stupcem Gender, age sa stupcem Age, age_group sa stupcem Age Group, i strani ključ other_participants. Tablica other_participants ima attribute id, bus_involvement koji će biti spojen sa stupcem Bus Involvement, heavy_rigid_truck_involvement sa stupcem Heavy Rigid Truck Involvement i atribut articulated_truck_involvement sa stupcem Articulated Truck Involvement. Posljednja tablica bila je crash, koja se sastoji od atributa id, type koji će biti povezan sa stupcem Crash Type, penalty_price koji će biti povezan sa stupcem Penalty Price, te strani ključevi state_fk, hour_fk, victim_fk, speed_fk i holiday_fk koji će služiti za spajanje sa ostalim tablicama. Konačno, stvorena je zadovoljavajuća baza podataka koju treba popuniti.

4.2. POPUNJAVANJE BAZE PODATAKA

Sljedeći korak bio je otvoriti MySQL i pregledati je li baza uspješno stvorena te nalaze li se sve tablice i atributi u njoj. Nadalje se ponovno koristi Python skripta za popunjavanje tablica vrijednostima iz csv datoteke. Prvo povezujemo skriptu sa csv datotekom koja je ranije analizirana i u kojoj se nalazi 80% podataka. Pomoću pd.DataFrame, odabrani su atributi jedan po jedan te povezani sa ranije navedenim nazivom stupaca iz kojih se žele izvući podatci. Sljedeće slike pokazat će par primjera punjenja tablica podacima (slika 2 i 3):

```
33 # Popunjavanje tablice state
34 state_names = df['State'].unique().tolist()
35 state_data = pd.DataFrame({'name': state_names})
36 state_data.to_sql(con=mydb, name='state', if_exists='append', index=False)
37
38 # Popunjavanje tablice holiday
39 holiday_data = pd.DataFrame(
40     {'christmas_period': df['Christmas Period'], 'easter_period': df['Easter Period']})
41 holiday_data.to_sql(con=mydb, name='holiday', if_exists='append', index=False)
42
43
```

Slika 2: Popunjavanje atributa sa vrijednostima iz csv stupaca

```

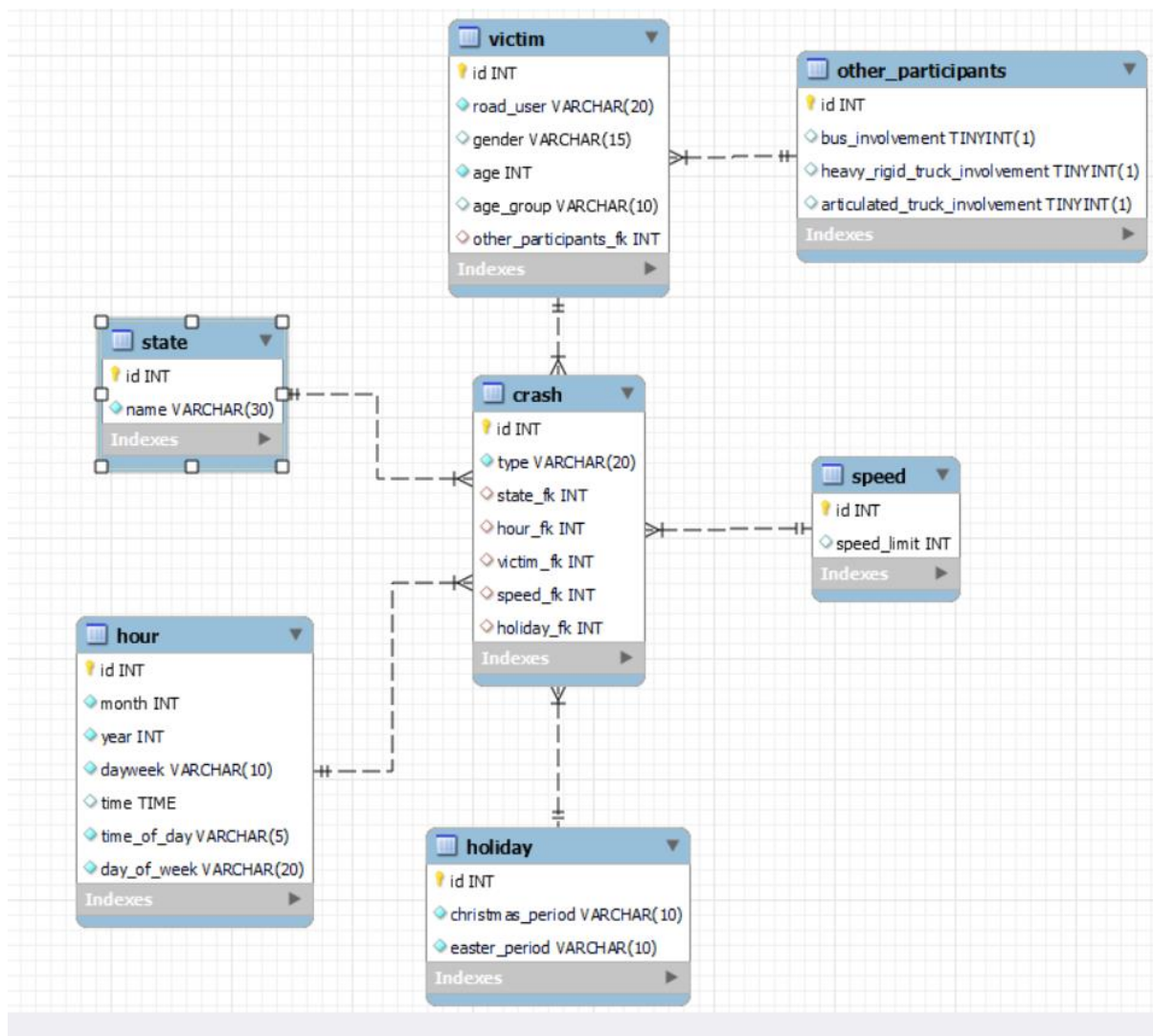
33
34 # Popunjavanje tablice hour
35 hour_data = []
36
37 for i, row in df.iterrows():
38     month = row['Month']
39     year = row['Year']
40     dayweek = row['Dayweek']
41     time = row['Time']
42     time_of_day = row['Time of day']
43     day_of_week = row['Day of week']
44
45     hour_entry = {
46         'month': month,
47         'year': year,
48         'dayweek': dayweek,
49         'time': time,
50         'time_of_day': time_of_day,
51         'day_of_week': day_of_week
52     }
53
54     hour_data.append(hour_entry)
55
56 hour_df = pd.DataFrame(hour_data)
57 hour_df.to_sql(con=mydb, name='hour', if_exists='append', index=False)

```

Slika 3: Popunjavanje atributa sa vrijednostima iz csv stupaca

4.3. EER DIJAGRAM

Pomoću MySQL opcije Reverse Engineer kreiran je EER dijagram. Enhanced Entity-Relationship dijagram je grafički alat koji se koristi za modeliranje i dizajn baza podataka i služi za vizualno prikazivanje strukture baze, uključuje entitete, njihove attribute i odnose. (Slika 4)



Slika 4: Prikaz EER dijagrama

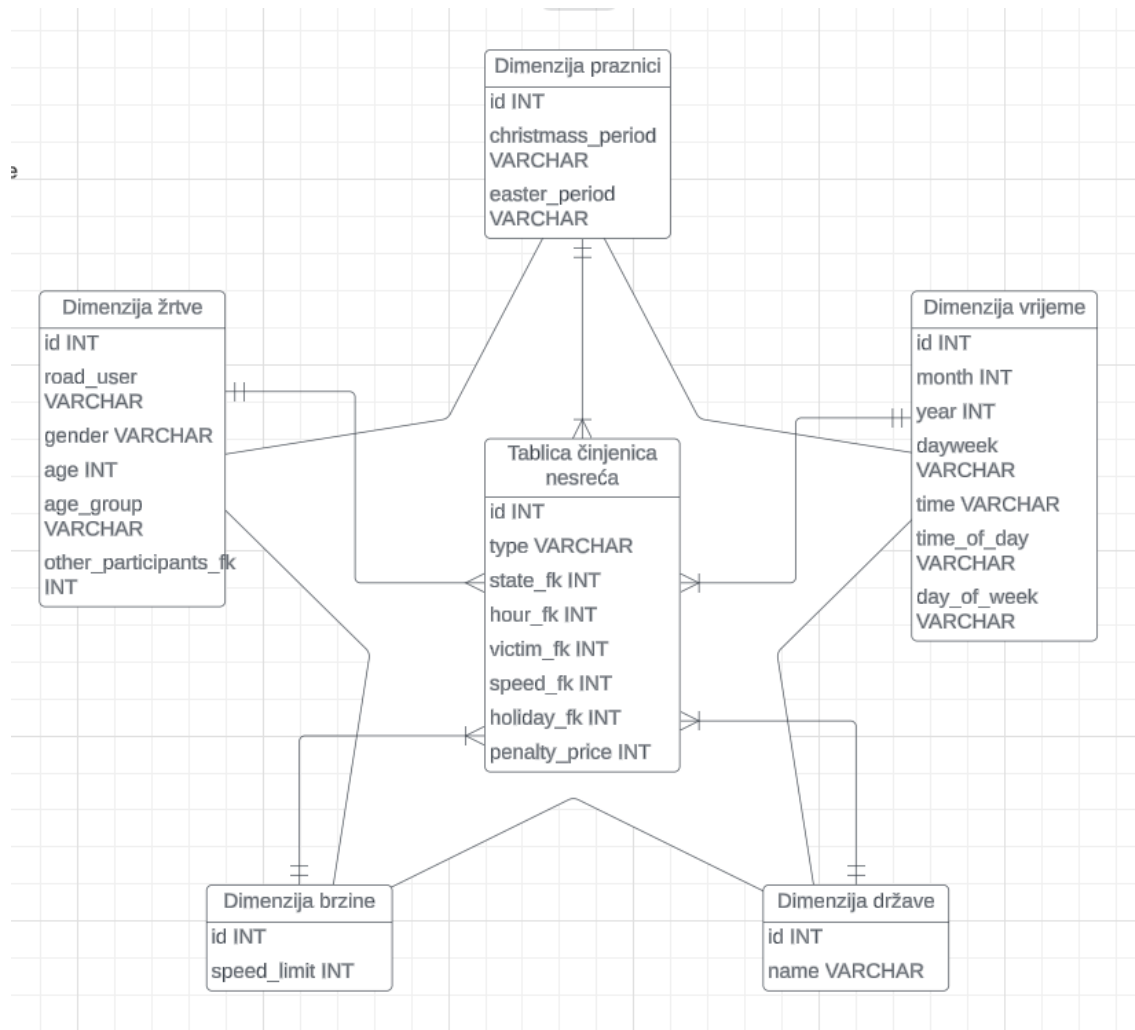
5. DIMENZIJSKI MODEL PODATAKA

Dimenzijsko modeliranje³ ključna je tehnika koja se koristi u skladištenju podataka kako bi se strukturirali i rasporedili podaci na način koji olakšava analizu i ekstrakciju informacija. Dimenzijsko modeliranje omogućava organizaciju podataka u formatu koji je optimiziran za brzo i efikasno izvršavanje analitičkih upita te je razumljiv krajnjim poslovnim korisnicima.

Dakle dimenzijski model podataka je pristup dizajnu baze podataka koji optimizira pristup podacima za analitičke svrhe. Umjesto normaliziranih struktura koje se koriste u operativnim bazama podataka, dimenzijski model koristi denormalizirane tablice kako bi se poboljšali upiti. Dimenzijski model sastoji se od dvije osnovne vrste tablica: tablice činjenica i dimenzijskih tablica. Tablica činjenica nalazi se u središtu modela i sadrži numeričke podatke ili mjerenja koja želimo analizirati, poput iznosa prodaje, broja narudžbi ili količine proizvoda. U ovom primjeru to će biti `penalty_price` atribut. Dimenzijske tablice okružuju tablicu činjenica i sadrže opisne attribute koji daju kontekst numeričkim podacima, kao što su vrijeme, lokacija, proizvod ili kupac. Ovaj raspored poznat je kao shema zvijezde (slika 5) , gdje tablica činjenica ima ključne veze prema dimenzijskim tablicama. Jedna od prednosti dimenzijskog modela je upotreba tehničkih ključeva (surrogate keys), koji djeluju kao jedinstveni identifikatori za retke tablica. Ovi ključevi omogućuju učinkovito povezivanje podataka između tablica, neovisno o prirodnim ključevima iz izvornog sustava. Tehnički ključevi pomažu u održavanju integriteta podataka i pojednostavljaju upravljanje podacima.

³ Javatpoint; What is dimensional modeling? ; <https://www.javatpoint.com/data-warehouse-what-is-dimensional-modeling>

5.1. STAR SCHEMA



Slika 5: Prikaz star scheme izrađene u Lucidchartu

5.2. ETL PROCES

Za kreiranje i punjenje dimenzijskog modela koristio se alat Pentaho s kojim smo se upoznali u sklopu predavanja ovog kolegija. Pentaho pruža grafičko sučelje koje olakšava proces pretvorbe relacijskog modela u dimenzijski model. Prilikom kreiranja dimenzijskog modela, važno je slijediti nekoliko pravila kako bi se osigurala učinkovitost i točnost analitičkih procesa. Prvi korak u stvaranju dimenzijskog modela je odvajanje tablice činjenica od dimenzija. Tablica činjenica sadrži mjere poslovanja koje želimo pratiti, dok dimenzijske tablice pružaju kontekst za te mjere. Moguće je i spajanje sličnih dimenzija kako bi se smanjila kompleksnost modela. U tom slučaju povezane relacije koje opisuju slične entitete mogu se spojiti u jednu dimenziju. Ova redundancija olakšava kasniju analizu podataka i omogućuje brže izvršavanje upita. Sljedeći korak je punjenje tablica podacima. Ovaj proces uključuje izdvajanje podataka iz relacijskih tablica, njihovu transformaciju i unos u dimenzijske i tablicu činjenica.

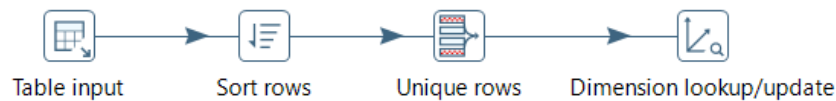
U alatu Pentaho, ovaj proces se provodi kroz ETL (Extract, Transform, Load) procese, koji omogućuju učinkovitu obradu i migraciju podataka iz izvora u dimenzijski model. ETL proces započinje ekstrakcijom podataka iz izvornog sustava, nakon čega slijedi transformacija podataka kako bi se uklonile nepravilnosti, konsolidirali duplikati i pripremili podaci za unos u skladište podataka. Transformirani podaci se zatim unose u odgovarajuće dimenzijske tablice i tablicu činjenica.

Prvo je bilo potrebno spojiti Pentaho na MySQL bazu podataka na kojoj se nalaze svi podatci, dakle prometne_nesrece. Također, u MySQL je potrebno i stvoriti novu bazu podataka koja će služiti kao skladište podataka, odnosno u nju će biti smještene sve dimenzijske tablice. Ona je nazvana pentaho_nesrece. Stvoren je dimenzijski model od pet dimenzija i jedne tablice činjenica, što je prikazano ranije u Star Schemi. Prvo su kreirane dimenzije jedna po jedna ali na vrlo slične načine, pomoću Table Input, Sort Rows, Unique Rows, Select Values i Dimension Lookup/Update opcija. Po redu:

5.2.1. DIMENZIJSKA TABLICA STATE

Prvo je korišten Table input-u u kojem se bira se tablica state iz stvorene konekcije na bazu podataka, te se odabire SQL select statement koji ispisuje odabranu tablicu. Zatim u Sort rows

koraku odabire se opcija Get fields i sortirani su podatci prema id-u, uzlazno. U unique rows biran je atribut name. U dimension lookup/update koraku prvo je potrebno odabrati target schemu, u ovom slučaju pentaho_nesrece. Zatim target table; dim_state koja će se naposljetku izraditi. U keys odabiru biramo ponovno id, a u Fieldsu sve ostale attribute te dimenzije. Najvažnije; dodan je tehnički ključ za dimenzijsku tablicu state_tk.



Slika 6: Izrada dim_state

5.2.2. DIMENZIJSKA TABLICA HOLIDAY

Za dimenzijsku tablicu holiday vrijedi sličan proces. Prvo je Table input u kojem se bira tablica holiday iz stvorene konekcije na bazu podataka, te se odabire SQL select statement koji ispisuje odabranu tablicu. Zatim u Sort rows koraku odabire se opcija Get fields i sortirani su podatci prema id-u, uzlazno. Ti odabrani podatci idu u dimension lookup/update. Prvo je potrebno odabrati target schemu, ponovno pentaho_nesrece. Zatim target table; dim_holiday koja će se naposljetku izraditi. U keys odabiru biramo ponovno id, a u Fieldsu sve ostale attribute te dimenzije. Najvažnije; dodan je tehnički ključ za dimenzijsku tablicu tk_holiday.



Slika 7: Izrada dim_holiday

5.2.3. DIMENZIJSKA TABLICA HOUR

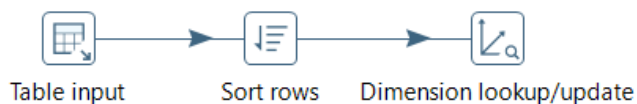
Sljedeća dimenzija je hour. Pomoću Table input dohvaćamo tablicu hour iz mysqla, u Sort rows koraku odabire se opcija Get fields i sortiraju se podatci uzlazno prema id-u, u Select Values opet biramo Get fields i na kraju u Dimension lookup/update prvo je potrebno odabrati target schemu, ponovno pentaho_nesrece. Zatim target table; dim_hour koja će se naposljetku izraditi. U keys odabiru biramo ponovno id, a u Fieldsu sve ostale attribute te dimenzije. Najvažnije; dodan je tehnički ključ za dimenzijsku tablicu tk_hour.



Slika 8: Prikaz izrade dim_hour

5.2.4. DIMENZIJSKA TABLICA SPEED

Za dimenziju speed vrijede isti procesi. Table input bira SQL statement, dodaje se tablica speed, u Sort Rows pomoću Get fields opcije dohvaćaju se podatci i sortiraju. Sve odlazi u Dimension lookup/update gdje se spaja na novu target schemu pentaho_nesrece, stvara se nova target table dim_speed i dodan je tehnički ključ speed_tk koji ćemo kasnije koristiti u fact table.



Slika 9: Prikaz izrade dim_speed

5.2.5. DIMENZIJSKA TABLICA VICTIM

Za dimenziju victim prvo je korišten Table input-u u kojem se bira se tablica victim iz SQL select statementa koji ispisuje odabranu tablicu. Zatim u Sort rows koraku odabire se opcija Get fields i sortirani su podatci prema id-u, uzlazno. U Select values biraju se podatci u Get fields i u dimension lookup/update koraku prvo je potrebno odabrati target schemu, pentaho_nesrece. Zatim target table; dim_victim koja će se naposljetku izraditi. U keys odabiru biraмо ponovno id, a u Fieldsu sve ostale attribute te dimenzije. Najvažnije; dodan je tehnički ključ za dimenzijsku tablicu victim_tk.



Slika 10: Prikaz izrade dim_victim

Finalne dimenzije ovakvog su izgleda:

Result Grid

Filter Rows:

Edit:

Export/Import:

Wrap Cell Cont

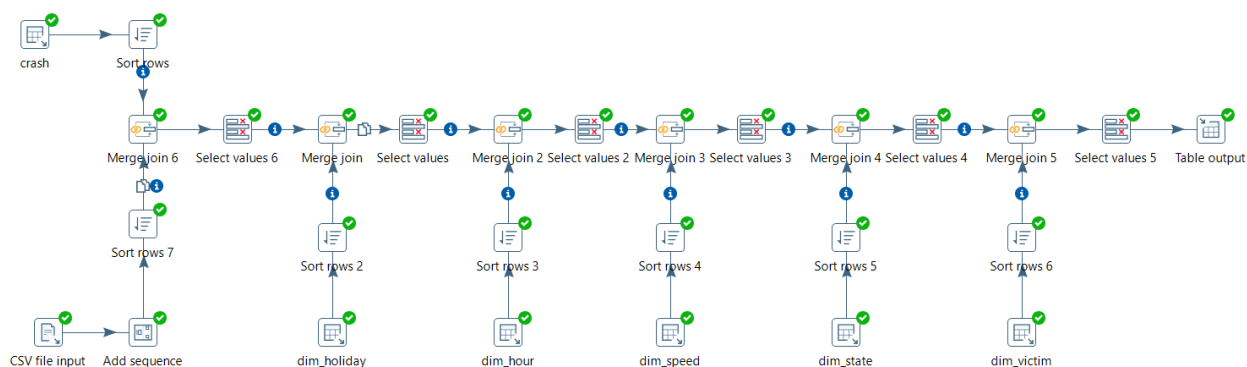
	tk_holiday	version	date_from	date_to	id	christmas_period	easter_period
0	1						
1	1		1900-01-01 00:00:00	2200-01-01 00:00:00	1	No	No
2	1		1900-01-01 00:00:00	2200-01-01 00:00:00	2	No	No
3	1		1900-01-01 00:00:00	2200-01-01 00:00:00	3	No	No
4	1		1900-01-01 00:00:00	2200-01-01 00:00:00	4	No	No
5	1		1900-01-01 00:00:00	2200-01-01 00:00:00	5	No	No
6	1		1900-01-01 00:00:00	2200-01-01 00:00:00	6	No	No
7	1		1900-01-01 00:00:00	2200-01-01 00:00:00	7	No	No
8	1		1900-01-01 00:00:00	2200-01-01 00:00:00	8	No	No
9	1		1900-01-01 00:00:00	2200-01-01 00:00:00	9	No	No
10	1		1900-01-01 00:00:00	2200-01-01 00:00:00	10	No	No
11	1		1900-01-01 00:00:00	2200-01-01 00:00:00	11	No	No

dim_holiday 1

Slika 11: Prikaz dimenzije holiday

5.2.6. TABLICA ČINJENICA FACT_CRASH

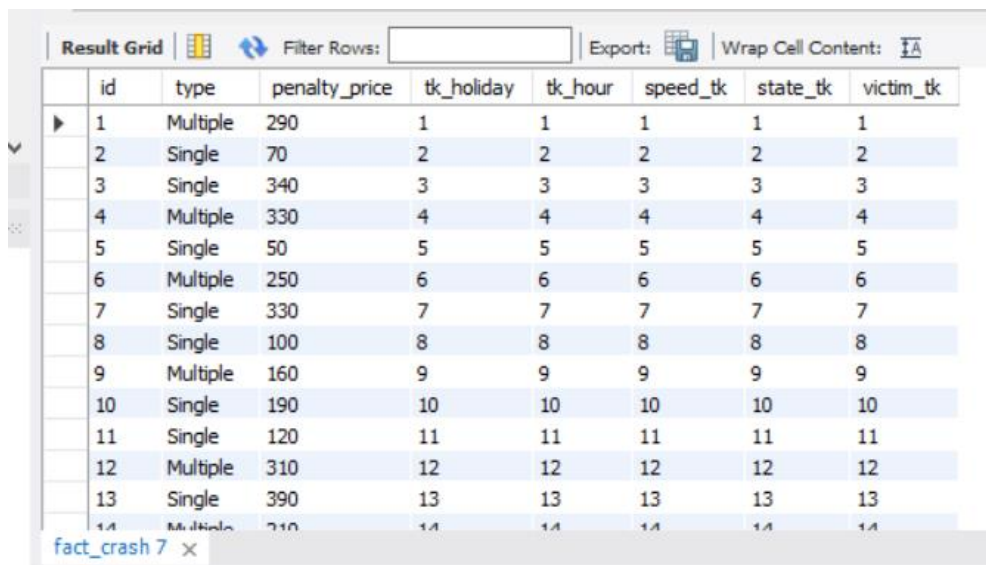
Slijedi najkompleksniji postupak, stvaranje tablice činjenica fact_crash. Prvo slijedi njegova slika (slika 12), a zatim tekst koji objašnjava korake preko slike.



Slika 12: Svi koraci korišteni za izradu fact_crash tablice

Za tablicu činjenica bilo je potrebno spojiti sve prethodno izrađene dimenzijske tablice kako bi dobili potrebne podatke iz njih. Prvo su izdvojeni svi podatci iz potrebne csv datoteke, zatim sa Add sequence korakom izabran je value id, Sort rows korak sortira podatke upravo po id-u. Zatim kroz Table input korak spajamo i crash tablicu iz MySQL-a. U koraku Sort rows, sortiram po atributima id, type i penalty_price s obzirom da su oni jedini stupci koji mi trebaju iz ranije baze podataka prometne_nesrece. Sortirani su uzlazno. Napokon dolazimo do prvog Merge Joina, u kojem koristeći Inner join spajam csv datoteku i crash tablicu po ključu id. Sljedeći korak je Select values, gdje biram Remove opciju i ukljanjam sve nepotrebne stupce. Zatim kroz Table input dodajem dimenziju dim_holiday, sortira se po id-u ta spaja putem Merge joina sa prethodnim korakom Select Values. Ključevi spajanja su sada id i tk_holiday koji je jedini potreban atribut iz holiday tablice. Sada u koraku Select Values opcijom Remove ponovno brišem sve nepotrebne podatke, što znači da sad ostaju id, type, penalty_price i holiday_tk. Ovaj postupak ponovljen je za svaku dimenziju identično, dakle iz svake je izvučen njen technical key. Redom to su : tk_holiday, tk_hour, speed_tk, state_tk, victim_tk. Posljednji korak je Table output u kojem se koristi Connection mysql, target schema pentaho_nesrece i target table fact_crash.

Slijedi prikaz tablice činjenica. (Slika 13)



	id	type	penalty_price	tk_holiday	tk_hour	speed_tk	state_tk	victim_tk
▶	1	Multiple	290	1	1	1	1	1
	2	Single	70	2	2	2	2	2
	3	Single	340	3	3	3	3	3
	4	Multiple	330	4	4	4	4	4
	5	Single	50	5	5	5	5	5
	6	Multiple	250	6	6	6	6	6
	7	Single	330	7	7	7	7	7
	8	Single	100	8	8	8	8	8
	9	Multiple	160	9	9	9	9	9
	10	Single	190	10	10	10	10	10
	11	Single	120	11	11	11	11	11
	12	Multiple	310	12	12	12	12	12
	13	Single	390	13	13	13	13	13
	14	Multiple	210	14	14	14	14	14

fact_crash 7 x

Slika 13: Prikaz tablice činjenica

Ovim korakom skladište podataka je gotovo te su sve dimenzije vidljive u MySQL bazi pentaho_nesrece (Slika 14).

```
1 • CREATE DATABASE pentaho_nesrece;
2 • USE pentaho_nesrece;
3
4 • SELECT * FROM dim_holiday;
5 • SELECT * FROM dim_hour;
6 • SELECT * FROM dim_state;
7 • SELECT * FROM dim_speed;
8 • SELECT * FROM dim_victim;
9 • SELECT * FROM fact_crash;
```

Slika 14: Pregled novostvorene baze

6. OLAP ANALIZA

OLAP (Online Analytical Processing) je tehnologija koja omogućava analizu podataka u više dimenzija, pružajući korisnicima brz i interaktivan način za analizu velikih količina podataka. Ova tehnologija je ključna za poslovnu inteligenciju, jer omogućava organizacijama da donose informisane odluke zasnovane na detaljnoj analizi podataka. OLAP sistemi se sastoje od nekoliko ključnih komponenti;

1. OLAP Kocke: OLAP kocke su višedimenzionalne strukture koje omogućavaju brz pristup podacima za analizu. Kocke se stvaraju od nekoliko dimenzija, kao što su vrijeme, zemljopis i crte proizvoda, sa sažetim podacima kao što su brojke prodaje.
2. Dimenzije: Dimenzije predstavljaju različite aspekte podataka koje korisnici žele analizirati. Primeri dimenzija uključuju vrijeme, lokaciju, proizvod, itd. Svaka dimenzija može imati hijerarhiju, kao što je godina, kvartal, mesec i dan u slučaju vremenske dimenzije.
3. Mjere: Mjere su kvantitativni podaci koji se analiziraju, kao što su prodaja, prihod, količina, itd. One predstavljaju vrijednosti koje su predmet analize i obično su u tablici činjenica.
4. Član: Član može biti jedinstven, odnosno jedinstvena stavka u hijerarhiji koja predstavlja jedno ili više pojavljivanje podataka.
5. Izračunati član: Član dimenzije čija se vrijednost izračunava pomoću izraza. Izračunate vrijednosti članova mogu se izvesti iz vrijednosti drugih članova.
6. Hijerarhija: Hijerarhija je logička struktura koja organizira članove dimenzije tako da svaki član ima jednog nadređenog člana i nula ili više podređenih članova. Dakle dijete je član na sljedećoj nižoj razini hijerarhije koja je izravno povezana s trenutnim članom. Nadređeni član je na sljedećoj višoj razini a ona mora biti izravno povezana s trenutnim članom.

OLAP također omogućava različite operacije koje pomažu u detaljnoj analizi podataka. Neke od njih su Roll-up, operacija koja konsolidira podatke, prelazeći sa niže na višu razinu. Drill-down operacija koja je suprotna roll-up operaciji, omogućuje detaljniju analizu podataka, prelazeći sa više na nižu razinu detalja. Na primjer, korisnik može detaljno analizirati podatke prelazeći sa mjesečnog na dnevni opseg podataka. Slice operacija omogućava selekciju jedne

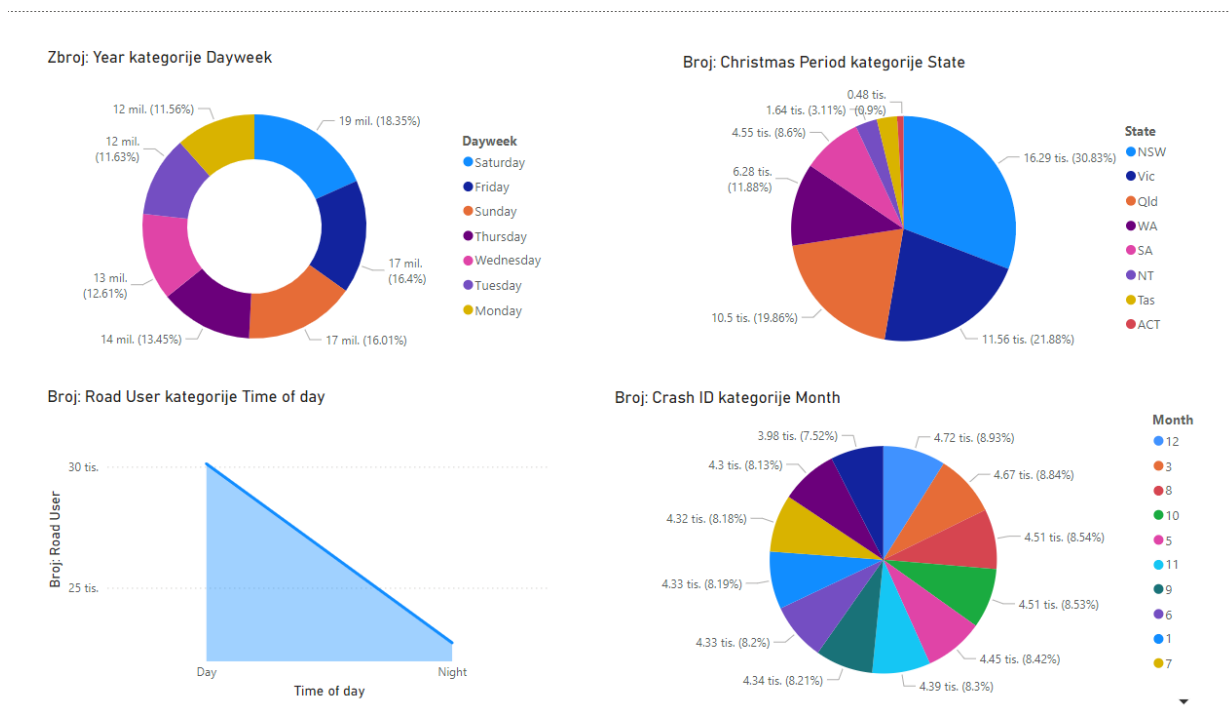
dimenzije kocke, kreirajući podkocku, što je korisno ako se želi na primjer analizirati samo jedan mjesec. Dice operacija omogućava odabir dvije ili više dimenzija kocke, kreirajući podkocku. U tom slučaju može se analizirati određeni mjesec i određeni npr proizvod. Također, postoje tri glavna tipa OLAP sistema, a to su MOLAP (Multidimensional OLAP) koji koristi unaprijed kreirane multidimenzionalne kocke za brzo izvođenje analiza. Prednost MOLAP-a je brzina. ROLAP (Relational OLAP) koji koristi relacijske baze podataka za čuvanje podataka i izvodi upite koristeći SQL. ROLAP je fleksibilniji kada je u pitanju veličina podataka, ali može biti sporiji u izvršavanju složenih upita. HOLAP (Hybrid OLAP) kombinira prednosti MOLAP-a i ROLAP-a, koristeći prednosti oba pristupa. Podaci se mogu čuvati u multidimenzionalnim kockama i relacionim bazama, ovisi o potrebi. Najveće prednosti OLAP-a su brz pristup podacima za analizu, intuitivno korištenje za krajnje korisnike i sposobnost analize podataka iz više dimenzija. Neki od nedostataka su to što implementacija može biti složenija i skupa.

6.1. VIZUALIZACIJA PODATAKA

Za vizualizaciju podataka izabran je alat Power BI preuzet sa službene Microsoft stranice⁴. Radi se o izuzetno moćnom alatu za poslovnu inteligenciju (BI) i vizualizaciju podataka koji omogućava korisnicima da prikupljaju, analiziraju i vizualizuju podatke iz različitih izvora na intuitivan i interaktivan način. Power BI nudi niz funkcionalnosti koje pomažu organizacijama da bolje razumiju i vizualiziraju svoje podatke.

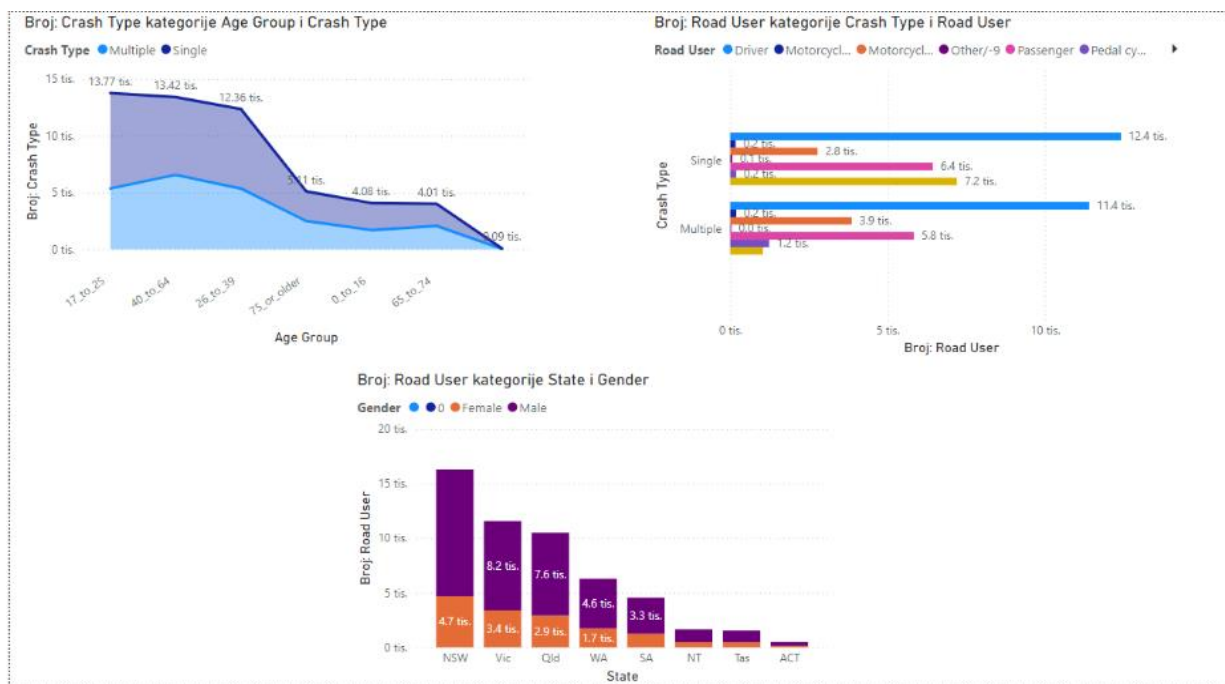
Prvi izrađen Dashboard 1 (slika 15) orijentiran je na vremenske motive. Prvi prikaz je prstenasti grafikon koji prikazuje koliko je u godini bilo nesreća u pojedinom danu u tjednu što je objašnjeno legendom. Pored njega prikazan je tortni grafikon gdje se vidi koliko je u pojedinoj državi u Australiji dogodilo nesreća u Božićnom razdoblju. Zatim slijedi površinski grafikon gdje se vidi da je većina nesreća bila po danu. Zadnje, također pomoću tortnog grafikona prikazan je broj i postotak nesreći u svakom pojedinom mjesecu.

⁴ Power BI, Data Visualization; <https://www.microsoft.com/en-us/power-platform/products/power-bi>



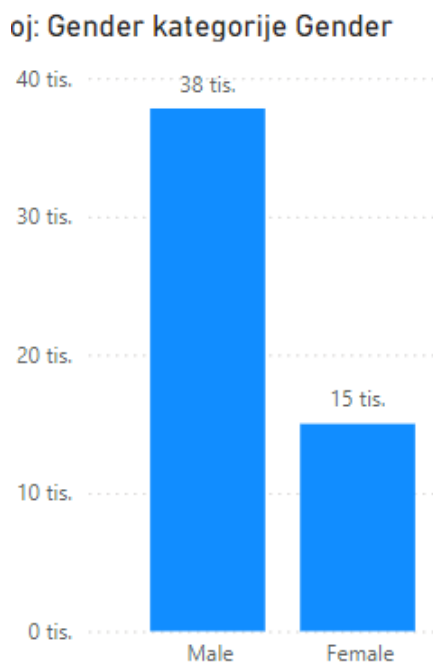
Slika 15: Dashboard 1

Dashboard 2 (slika 16) orijentiran je na ljude poginule u nesrećama. Prvi grafikon složenog područja prikazuje broj nesreća u pojedinoj kategoriji godina onesrećenih. Jasno se vidi da mladi najviše stradaju. Zatim grupirani trakasti grafikon pokazuje radi li se o single ili multiple nesreći te u svakoj od njih imamo broj koliko nesreća je u svakoj kategoriji sudionika u prometu. Kao i očekivano, najviše stradaju vozač i suvozač. Zadnja stavka dashboarda je složeni stupčasti grafikon u kojem se vidi koliko je muškaraca i koliko je žena stradalo u svakoj pojedinoj državi.



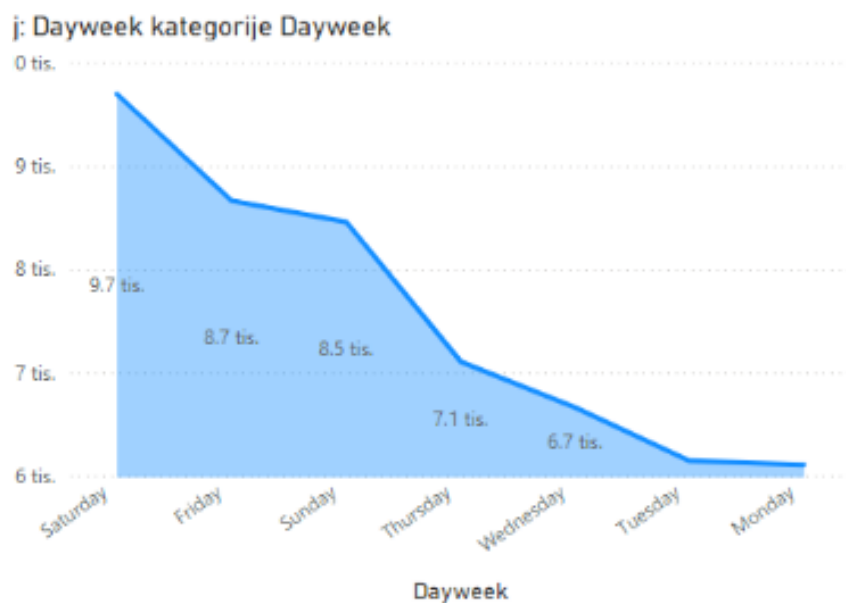
Slika 16: Dashboard 2

Nadalje slijedi prikaz pojedinih podataka. Prvi prikaz podataka (slika 17) je grupirani stupčasti grafikon gdje se vidi da je 38 tisuća onesrećenih bilo muškarci, dok upola manje, 15 tisuća žena.



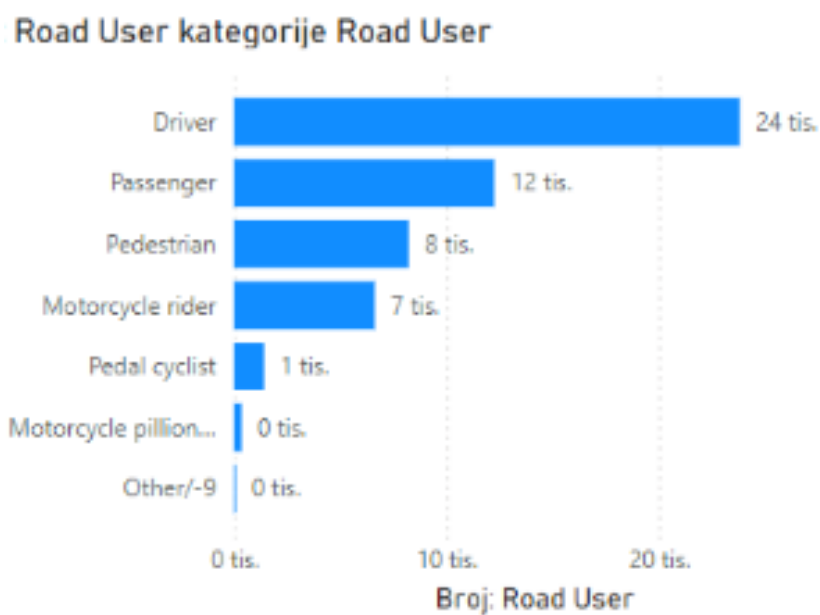
Slika 17: Prikaz podataka 1

Drugi prikaz podataka (slika 18) je grafikon složenog područja, gdje se vide i očekivani podatci. Najviše nesreća subotom, petkom, nedjeljom, pa onda tek ostali radni dani.



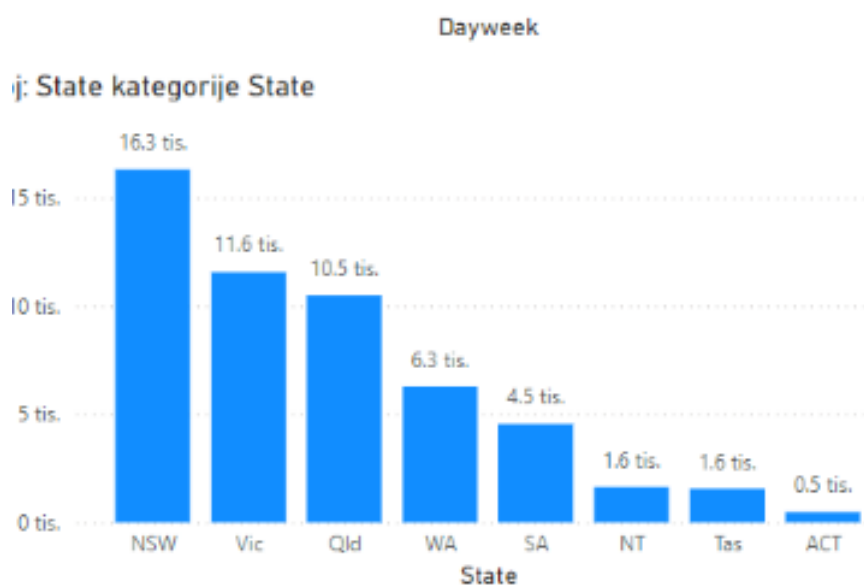
Slika 18: Prikaz podataka 2

Treći prikaz podataka (slika 19) grupirani je trakasti grafikon gdje se vidi koji sudionici u prometu najviše stradaju.



Slika 19: Prikaz podataka 3

Četvrti prikaz podataka (slika 20) je složeni stupčasti grafikon gdje se vidi ukupan broj nesreća u svakoj pojedinoj državi.



Slika 20: Prikaz podataka 4

7. ZAKLJUČAK

Projekt je prvotno objasnio kako poslovna inteligencija olakšava mnoštvo zadataka neke organizacije ili pojedinca. Demonstrirano je sveobuhvatno razumijevanje procesa upravljanja i analize podataka, praktična primjena teorijskih koncepata iz područja skladištenja podataka i rudarenja podataka. Korištenjem različitih alata i tehnika, od prikupljanja i transformacije podataka do njihove pohrane, modeliranja i vizualizacije, pokazano je kako se podaci mogu učinkovito koristiti za poslovnu inteligenciju i podršku u donošenju odluka. Uspješno su povezane različite faze poslovne inteligencije, od prikupljanja i osnovne analize podataka, stvaranje baza podataka i popunjavanje baza, stvaranje skladišta podataka i obrade podataka te njihove vizualizacije. Time se pokazala važnost integriranog pristupa u izgradnji sustava poslovne inteligencije, jasno se može uvidjeti koliko velik značaj imaju kvalitetni podatci i njihova analiza. Upravo to omogućava organizacijama da maksimiziraju vrijednost svojih podataka, osiguravajući precizne i pravovremene uvide koji mogu značajno unaprijediti poslovne procese i strategije. Projekt također ističe važnost kvalitetne pripreme podataka i robusnog dizajna baze podataka kao temelja za uspješnu analitiku i poslovnu inteligenciju. U konačnici, korištenjem podosta različitih alata i ulaganjem velikog broja sati u ovaj jedan mali projekt, zaključujem najbitnije; skladište podataka i njihova obrada vrlo je široko i zahtjevno područje koje zahtjeva preciznost i točnost, a kao ishod nudi jasne i svima razumljive rezultate.

8. LITERATURA

1. Stedman, C. „What is BI?“
<https://www.techtarget.com/searchbusinessanalytics/definition/business-intelligence-BI>
2. Kaggle, Australian Fatal Road Accident 1989-2021;
<https://www.kaggle.com/datasets/deepcontractor/australian-fatal-car-accident-data-19892021>
3. Data Warehousing: ETL, OLAP and OLTP; <https://blog.bismart.com/en/data-warehousing-olap-oltp>
4. Pregled analitičke obrade (OLAP); <https://support.microsoft.com/hr-hr/office/pregled-analiti%C4%8Dke-obrade-na-mre%C5%BEi-olap-15d2cdde-f70b-4277-b009-ed732b75fdd6>