

**VIŠJA STROKOVNA ŠOLA ACADEMIA
MARIBOR**

PRETOČNE GLASBENE VSEBINE IN BIG DATA

Kandidatka: Tina Očkerl

Vrsta študija: študentka izrednega študija

Študijski program: Informatika

Mentor predavatelj: mag. Ervin Schaff

Mentor v podjetju: Petra Munda, dipl. inž. inf.

Lektor: Tadeja Šef, dipl. slov.

Maribor, 2024

IZJAVA O AVTORSTVU DIPLOMSKEGA DELA

Podpisana Tina Očkerl sem avtorica diplomskega dela z naslovom Pretočne glasbene vsebine in Big Data, ki sem ga napisala pod mentorstvom mag. Ervina Schaffa.

S svojim podpisom zagotavljam, da:

- je predloženo delo izključno rezultat mojega dela,
- sem poskrbela, da so dela in mnenja drugih avtorjev, ki jih uporabljam v predloženi nalogi, navedena oz. citirana skladno s pravili Višje strokovne šole Academia Maribor,
- se zavedam, da je plagiatorstvo – predstavljanje tujih del oz. misli kot moje lastne, kaznivo po Zakonu o avtorski in sorodnih pravicah (Uradni list RS, št. 16/07 – uradno prečiščeno besedilo, 68/08, 110/13, 56/15 in 63/16 – ZKUASP); prekršek pa podleže tudi ukrepom Višje strokovne šole Academia Maribor skladno z njenimi pravili,
- skladno z 32.a členom ZASP dovoljujem Višji strokovni šoli Academia Maribor objavo diplomskega dela na spletnem portalu šole.

Selnica ob Muri, avgust 2024

Podpis študenta:

ZAHVALA

Zahvalila bi se svojemu mentorju mag. Ervinu Schaffu za potrpežljivost, strokovno svetovanje in spodbudo pri nastajanju diplomske naloge.

Zahvalila se bi tudi svoji družini, katera mi je stala ob strani in me spodbujala, da sem lahko zraven službe in majhnega otroka opravila študij in diplomsko nalogo.

Iskrena hvala vsem.

POVZETEK

V diplomski nalogi sem raziskovala pretočne glasbene vsebine (Spotify in YouTube Music) ter Big Data (veliko podatkovje) in povezavo med le-temi. V prvem delu naloge so opisane glavne lastnosti Big Data ter obdelovanje in shranjevanje teh podatkov.

Ker se veliki podatki delijo na strukturirane in nestrukturirane podatke, razložim zakaj in na kakšen način katere shranimo, ter katere sisteme shranjevanja poznamo. Razložen je sistem za obdelavo ter podroben opis posameznih faz obdelave. Za lažje razumevanje sem narisala sliko sistema.

V drugem delu naloge se osredotočim na glasbena ponudnika, in sicer na Spotify in YouTube Music. Opisano je delovanje teh glasbenih ponudnikov, njuno delovanje in prioritete. Njune podobnosti in razlike so predstavljene v tabeli.

Sledi raziskava o tem, kako Spotify in YouTube Music uporabljata velike podatke, da razvijata prilagojeno vsebino za uporabnike s ciljem, da jih zadovoljita na način, da potem ostanejo ali postanejo dolgoletne stranke. Ugotavljam, da ponudnika Spotify in YouTube Music pridobita podatke posameznikov, nato pa na podlagi teh podatkov ustvarita nove glasbene liste, glede na njihovo dosedanje poslušanje skladb.

Naredila sem skripto, ki omogoča izvoz glasbenega seznama uporabnika iz Spotifya. Le-ta se lahko z uporabo te skripte prenese na katerega drugega ponudnika, na primer na Apple Music, kar je zelo priročno.

Postavila sem štiri hipoteze, katere sem na koncu potrdila, delno potrdila ali ovrgla. V hipotezah sem se vprašala ali sta glasbena ponudnika primerljiva in pa določila število glavnih lastnosti Big Data, kjer sem naletela na različne rešitve, jih med seboj primerjala in prišla do zaključka.

Ključne besede: »Big Data, Spotify, YouTube Music, Spotify API, skripta glasbenega seznama«

ABSTRACT

»STREAMING MUSIC CONTENT AND BIG DATA«

In my thesis, I focused on Big Data, Spotify and YouTube Music as well as the connection between all three. I describe the main characteristics of Big Data. Next I focus on how Big Data is processed and stored. Since Big Data is divided into structured and unstructured data, I explain why and in what way we store which, and which storage systems we know.

When we processing data I explain the processing system and describe the processing phases in more details. For a better understanding I drew a picture of the system, how it works.

In the second part of the thesis I focus on music providers Spotify and YouTube Music.

I describe how the two music providers work, what their priorities are. I write down how similar/different they are. I made a table for an easier comparison, where the data are collected in a condensed form.

I continue a research how Spotify and YouTube Music use Big Data to develop personalized content for users with goal of satisfying them so that they then remain or become long-term customers. I find that Spotify and YouTube Music take data from individuals and use that data to create new playlists that are meant to give people new ideas based on their listening history.

I made a script that enables the export of a user's playlist from Spotify. This can be transferred to some other provider using this script, for example Apple Music, which is very convinient.

I set up four hypotheses, which in the end I comfirmed, partially confirmed or rejected. In the hypotheses I asked myself whether the two music providers are comparable and determined the number of main characteristics of Big Data, where I came across different solutions, compared them with each other and came to a conclusion.

Keywords: »Big Data, Spotify, YouTube Music, Spotify API, Playlist Script«

KAZALO VSEBINE

1	UVOD	9
1.1	OPIS PODROČJA IN OPREDELITEV PROBLEMA	9
1.2	NAMEN, CILJI IN OSNOVNE TRDITVE	9
1.3	PREDPOSTAVKE IN OMEJITVE	10
1.4	UPORABLJENE RAZISKOVALNE METODE	10
2	BIG DATA	12
2.1	SHRANJEVANJE.....	14
2.1.1	<i>Shranjevanje strukturiranih podatkov:</i>	14
2.1.2	<i>Shranjevanje nestrukturiranih podatkov</i>	15
2.1.3	<i>Shranjevanje v oblaku</i>	17
2.2	OBDELAVA:.....	18
2.2.1	<i>Hadoop in MapReduce</i>	19
2.2.2	<i>Bloomov filter</i>	20
2.2.3	<i>PageRank:</i>	21
2.3	LASTNOSTI	22
2.3.1	<i>Volume (Obseg)</i>	22
2.3.2	<i>Velocity (Hitrost)</i>	23
2.3.3	<i>Variety (Raznolikost)</i>	23
2.3.4	<i>Veracity (Verodostojnost)</i>	24
2.3.5	<i>Value (Vrednost)</i>	24
2.3.6	<i>Vizualization (vizualizacija)</i>	24
2.4	VELIKO PODATKOVJE IN MEDICINA.....	26
3	SPOTIFY	27
3.1	LASTNOSTI	27
3.2	SPOTIFY IN BIG DATA	28
3.3	SPOTIFY IN PROGRAMSKI VMESNIKI (API).....	29
4	YOUTUBE MUSIC	30
4.1	LASTNOSTI	30
4.2	YOUTUBE MUSIC IN BIG DATA	30
4.3	YOUTUBE IN PROGRAMSKI VMESNIKI (API).....	31
5	PRIMERJAVA MED OBEMA PONUDNIKOMA	32
5.1	PRIMERJAVA UPORABNIŠKIH LASTNOSTI.....	32
5.2	PRIMERJAVA V POVEZAVI Z BIG DATA	34
6	SEZNAM GLASBE PRI PONUDNIKU	35
6.1	RAZVOJNO OKOLJE IN JAVASCRIPT	35
6.2	SKRIPTA ZA IZVOZ GLASBENEGA SEZNAMA	35
6.3	UGOTOVITVE	43

6.4	NAPOTKI ZA NADALJNJE DELO.....	44
7	SKLEP	45
8	SEZNAM UPORABLJENIH VIROV	47

KAZALO SLIK

SLIKA 1: BIG DATA.....	12
SLIKA 2 SOCIALNA OMREŽJA	13
SLIKA 3: SHRANJEVANJE V OBLAKU	17
SLIKA 4: HADOOP	18
SLIKA 5: MAP.....	19
SLIKA 6: BLOOMOV FILTER	21
SLIKA 7: PAGERANK	22
SLIKA 8: 4V'S VELIKEGA PODATKOVJA	25
SLIKA 9: VELIKI PODATKI IN MEDICINA.....	26
SLIKA 10: SPOTIFY	27
SLIKA 11: SPOTIFY API.....	29
SLIKA 12: YOUTUBE MUSIC	30
SLIKA 13: SPOTIFY VS YOUTUBE MUSIC	34
SLIKA 14: KODA JAVASCRIPT.....	36
SLIKA 15: SCOPES	36
SLIKA 16: KODA JAVASCRIPT.....	37
SLIKA 17: CLIENT ID IN CLIENT SECRET	37
SLIKA 18: KODA JAVASCRIPT	38
SLIKA 19: KODA JAVASCRIPT.....	38
SLIKA 20: KODA JAVASCRIPT.....	39
SLIKA 21: PRIKAZ PO ZAGONU INDEX.JS.....	39
SLIKA 22: KODA JAVASCRIPT	40
SLIKA 23: KODA JAVASCRIPT.....	40
SLIKA 24: KODA JAVASCRIPT.....	41
SLIKA 25: KODA JAVASCRIPT	41
SLIKA 26: PRIKAZ V KONZOLNEM OKNU	42
SLIKA 27: PRIKAZ V KONZOLNEM OKNU	42

KAZALO TABEL

TABELA 1: PRIMERJAVA MED PONUDNIKOMA	32
--	----

1 UVOD

1.1 Opis področja in opredelitev problema

V diplomski nalogi z naslovom Pretočne glasbene vsebine in Big Data, sem se osredotočila na svetovno znana glasbena ponudnika Spotify in YouTube Music. Raziskovala sem, kaj so Big Data, kako se shranjujejo in obdelujejo ter njihove lastnosti. V nadaljnjem raziskovanju sem ugotavljala kako glasbena ponudnika uporabljata Big Data. Razlog za izbor teme je predvsem ta, da želim pridobiti novo in dodatno znanje glede velikih podatkov. Izbrano področje se mi zdi pomembno, saj so Big Data del tehnologije, ki sega v prihodnost. Organizacijam omogočajo več vpogledov, večjo učinkovitost, ustvarjanje prihodkov, hitrejši razvoj, kar je zelo pomembno. Big Data je perspektivna tema, o kateri menim, da je dobro vedeti kaj več. Verjamem, da bodo strokovnjaki na tem področju iskani in potrebovani.

Del, kjer raziskujem Spotify in YouTube Music je zame zanimiv, saj rada poslušam glasbo in je velik del mojega življenja. V ta namen sem se odločila, da bom naredila skripto, katera bi olajšala uporabo, oz. prinesla dodatno vrednost pri poslušanju glasbe. V računalništvu je skript kratek in preprost nabor navodil. Skripta omogoča izvoz seznama glasbe iz glasbenega ponudnika.

1.2 Namen, cilji in osnovne trditve

Tema, ki jo raziskujem je zelo obširna, a sem se osredotočila na nekaj ključnih stvari. Da bi lahko dobro opisala delovanje velikega podatkovja, je bilo treba raziskati kaj sploh so Big Data, njihove lastnosti, shranjevanje in obdelovanje. Postavila sem prvo hipotezo, ki se glasi:

H1: Big Data imajo 5V's - 5 glavnih lastnosti.

V nadaljevanju sem se osredotočila na glasbena ponudnika Spotify in YouTube Music. Raziskovala sem, za kaj se uporabljata, njune razlike in podobnosti in za katere uporabnike sta priporočljiva. Nato sem raziskovala, kako uporabljata velike podatke.

Hipotezi, ki sem si ju tukaj postavila sta:

H2: Spotify in YouTube Music sta primerljiva s stališča uporabnika.

H3: Spotify in YouTube Music sta primerljiva s stališča Big Data.

Nazadnje sem izdelala skripto, katera omogoča izvoz glasbenega seznama. Tukaj je bil cilj predvsem to, da lahko dobro narejena skripta olajša oz. doprinese dodatno vrednost pri poslušanju glasbe. Hipoteza:

H4: Izdela se lahko skripta za izvoz seznama glasbe iz glasbenega ponudnika.

1.3 Predpostavke in omejitve

Pri pisanju diplomske naloge sem se omejila na literaturo in spletne vire. Skripto sem pisala v WebStorm razvojnem okolju, in sicer v JavaScript jeziku. Pregledati sem morala delovanje Spotify API-ja in se naučiti, kako ga vključiti v razvoj skripte za izvoz glasbenega seznama. Narediti sem morala svojo testno aplikacijo na Spotifyu in s tem dobila dostop do knjižnic, ki jih Spotify API ponuja. Ker sem prvič delala preko te aplikacije, sem potrebovala kar nekaj časa, da sem ugotovila kako deluje. Preko YouTube sem si ogledala nekaj posnetkov, ki so mi pomagali pri razvoju.

Omejitev glede anket nisem imela, saj nisem delala konkretno z ljudmi. Z verodostojnostjo virov ni težav, saj sem uporabljala uradne strani in knjigo.

1.4 Uporabljene raziskovalne metode

Prvi del diplomskega dela bo teoretičen, v katerem sem si pomagala z literaturo (strokovni članki in knjige) ter z osebnimi zapiski. S pomočjo literature sem predstavila lastnosti Big Data, njihovo shranjevanje in obdelovanje ter opisala zakaj so pomembni. Prav tako sem prišla do ugotovitve, na kakšen način uporabljata Big Data, znana glasbena ponudnika.

Uporabila sem primerjalno študijo, s katero sem primerjala glasbena ponudnika Spotify in YouTube Music. Primerjala sem ju iz stališča uporabnika in iz stališča Big Data. Naredila sem tabelo, iz katere je razvidno v čem sta si ponudnika Spotify in YouTube Music podobna in v čem ne.

Uporabila sem metodo abstrakcije. Za to metodo je značilno, da upoštevamo samo dejstva, ki so bistvena, vse ostalo zanemarimo. Ker je tema, ki jo raziskujem zelo obširna, je bilo pomembno, da uporabim samo podatke, ki so za mojo nalogo bistveni.

Pomembna je bila tudi metoda analize, pri kateri sem se sistematično lotila naloge in imela objektivni pristop k razumevanju podatkov. S tem sem izluščila bistvo in kvalitetnejše podatke.

Pri razvoju skripte sem uporabila metodo branja literature, katera mi je omogočila razumevanje kode in me podučila o raznih knjižnicah, ki so pomembne pri razvoju.

Ker sem se prvič srečala s Spotify API-jem, je bilo potrebnega veliko eksperimentiranja in učenja. Pregledala sem veliko podatkov na njihovi uradni strani in preučila njegovo delovanje. Za delovanje skripte sem se morala registrirati na njihovi uradni strani Spotify API for developers (Spotify API za razvijalce), kjer sem si ustvarila svoj račun, naredila testno aplikacijo in pregledala njihove knjižnice, katere sem nato vključila v skripto.

2 BIG DATA

Zaradi IoT (internetnih) naprav, pametnih mobilnih telefonov, brezpilotnih vozil se je ustvarilo ogromno podatkov, katerih število se večja/raste eksponentno.

Big Data so zbirka podatkov, ki so tako veliki in kompleksni, da jih nobeno tradicionalno orodje ni zmožno shranjevali ali procesirati. Vemo, da veliko podatkovje vključuje veliko podatkov ampak kakšna količina je to? Po Forbesu je 2,5 kvintiljona bajtov podatkov ustvarjenih vsak dan. Torej vprašanje je, kako se Big data orodja spoprijemajo s tako veliko količino podatkov?

Dandanes razvoj zelo in eksponentno napreduje. Razvijajo se vedno boljši pametni mobilni telefoni, televizije, IoT naprave, avtomobili ... Že z našimi pametnimi telefoni, preko katerih pošljemo npr. posnetek preko WhatsAppa ali kakšne druge aplikacije, ustvarimo podatke. Ti podatki niso v takšnem formatu, da jih naša baza podatkov lahko obravnava. Npr. tudi novi brezpilotni avtomobili snemajo vsako minuto podrobnosti, kot so velikost avtomobila, razdalja do ovire itd., nato se odločijo kako bodo reagirali. Da lahko reagirajo, je po vsakem prevoženem kilometru ustvarjenih veliko podatkov.

(Edureka, 2023)

(Simplilearn, 2019)



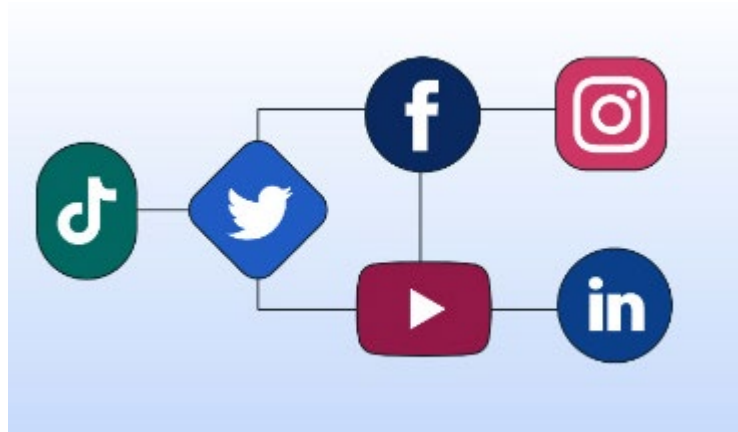
Slika 1: Big data

Vir: (<https://www.analyticsvidhya.com>)

Pri razvoju velikega podatkovja oz. Big Data so najbolj pomemben del socialna omrežja, kot so Facebook, Instagram, YouTube, Spotify, Tweeter ... Dandanes so socialna omrežja del življenja večine ljudi. Imajo naše osebne podatke kot so ime, starost, fotografije, posnetke. Vse to skupaj z vsakim všečkom, delitvijo fotografije ali posnetka, ustvari veliko podatkov. Ti podatki niso strukturirani in so zelo veliki. Tudi pri spletnih trgovinah se ustvarja veliko podatkov. Vse kar iščemo (vsa zgodovina iskanja, katere artikle smo iskali, katere strani obiskujemo ...) se shrani v obliki podatkov.

Torej Big Data je izraz zbirke podatkovnih nizov, ki so tako velike in kompleksne, da jih je pretežno procesirati na sistemskih orodjih baz podatkov ali tradicionalnih aplikacijah za obdelavo podatkov.

(Edureka, 2023)



Slika 2 Socialna omrežja

vir: (<https://sproutsocial.com>)

2.1 Shranjevanje

Ustvarjeni podatki se približno na vsaki dve leti podvojijo. V bistvu rast podatkov sovпада z zmožnostjo shranjevanja podatkov, saj vsi ustvarjeni podatki, ki so uporabni in jih potrebujemo morajo biti nekje shranjeni. Zaradi te eksponentne rasti podatkov, imamo pametne telefone, Netflix, računalništvo v oblaku, IoT.

(Holmes, 2019)

2.1.1 Shranjevanje strukturiranih podatkov:

Strukturirani podatki do katerih dostopamo preko pametnih telefonov ali računalnikov, so shranjeni v relacijski podatkovni bazi. *Relational Database Management System* (RDBMS) je sistem, kateri se uporablja za dostopanje, manipuliranje, ohranjanje in ustvarjanje podatkov. Najprej je potrebno izdelati podatkovno bazo, in sicer ustvariti shemo oziroma strukturo podatkovne baze (poznati in porazdeliti je potrebno podatkovna polja v tabele in identificirati razmerja med tabelami). Ko je to narejeno, s strukturiranim povpraševalnim jezikom (SQL) opravimo poizvedbo.

Danes ob tako veliki količini podatkov prihaja do problemov, in sicer gre za problem razširljivosti. Relacijske podatkovne baze postajajo nezanesljive in počasne. To se dogaja zaradi neprestanega dodajanja podatkov, ki so na enem samem strežniku. Razširljivost lahko dosežemo z večjo močjo računalnika, kar pa ima svoje meje.

Relacijske podatkovne baze se še vedno uporabljajo predvsem zaradi ACID, saj so te lastnosti zelo zaželjene. ACID je:

A (*atomicity*) – atomarnost: podatkovna baza ne more biti posodobljena ob nepopolni transakciji.

C (*consistency*) – konsistentnost: izločanje podatkov, ki so neveljavni.

I (*isolation*) – izolacija: transakcije se ne mešajo med sabo.

D (*durability*) – trajnost: posodobitev podatkovne baze pred naslednjo transakcijo.

(Holmes, 2019)

2.1.2 Shranjevanje nestrukturiranih podatkov

Nestrukturirani podatki zahtevajo drugačen pristop od RDBMS. Ta sistem je neprimeren, saj je organizacija nestrukturiranih podatkov v tabele/stolpce nemogoča, in sicer zato, ker je shemo relacijske podatkovne baze, ko je enkrat ustvarjena težko spremeniti.

Te velike množice podatkov so shranjene na več strežnikih. Zaradi okvar strežnikov je pomembno, da obstajajo identične kopije le-teh in tudi te morajo biti shranjene na več strežnikih.

(Holmes, 2019)

2.1.2.1 Hadoop:

Hadoopov porazdeljeni datotečni sistem DFS omogoča, da je veliko podatkovje shranjeno na več računalnikih zanesljivo in učinkovito.

Napisan je v programskem jeziku Java in omogoča shranjevanje ter zagotavlja platformo za analizo podatkov (nestrukturiranih in delno strukturiranih). Pri uporabi Hadoop DFS sistema so podatki porazdeljeni v podatkovnih centrih, po več deset tisoč vozliščih.

Osnovno gručo nekega Hadoop DFS sistema sestavljata eno imensko vozlišče ali z drugim poimenovanjem gospodar (angl. *NameNode*) ter veliko podatkovnih vozlišč oz. sužnjev (angl. *DataNodes*).

Torej *NameNode* je glavno vozlišče, ki vsebuje metapodatke o shranjenih podatkih. Podatki so shranjeni na podatkovnih vozliščih, ki so del strojne opreme v porazdeljenem okolju.

V *NameNodu* so metapodatki o podatkih, ki so shranjeni v *DataNode*, npr. za kateri podatkovni blok gre, kje se hranijo replikacije podatkovnega bloka itd. Imensko vozlišče porazdeljuje prostor shranjevanja, spremlja razpoložljivost shranjevanja, se ukvarja z poizvedbami iz odjemalčevega računalnika, skrbi za odpiranje/zapiranje datotek – torej za osnovne datotečne operacije. Dostop do podatkov je nadzorovan.

DataNodes oziroma podatkovna vozlišča se ukvarjajo z ustvarjanjem, brisanjem in repliciranjem blokov, saj so odgovorna za shranjevanje podatkov. Najpomembneje je, da so ustvarjene kopije v drugih podatkovnih vozliščih, saj če pride do težave oz. do ne delovanja nekega vozlišča, druga prevzamejo naloge, tako se podatki ne izgubijo. Glavno vozlišče ugotavlja nedelovanje podatkovnih vozlišč na način, da jim pošilja srčni utrip (angl. *Heartbeat*).

(Bakratsas, Basaras, Katsaros, & Tassiulas, 2017), (Holmes, 2019)

Če ga od podatkovnega vozlišča ne dobi nazaj, se smatra, da to vozlišče ne deluje, zato drugo podatkovno vozlišče, kjer je kopija ne delujočega vozlišča postane vozlišče teh podatkov. Če pa se zgodi, da imensko vozlišče več ne deluje, pa se uporabijo vgrajeni varnostni sistemi.

Razlika in tudi prednost Hadoop sistema pred relacijskimi bazami je ta, da lahko neprestano dodajamo velike količine podatkov. Dodajamo lahko tudi nove podatke za katere ne rabimo vedeti zakaj točno jih bomo potrebovali.

(Edureka, 2023)

2.1.2.2 NoSQL

NoSQL pomeni Ne samo SQL in se uporablja za nerelacijske podatkovne baze. Ne uporaba SQL-a v nerelacijskem modelu obstaja zaradi omogočanja neprestanega dodajanja novih podatkov. Gre za lastnosti nerelacijskega modela, ki so nujne za upravljanje z velikimi podatki. Te lastnosti so razpoložljivost, zmogljivost in razširljivost. CAP izrek pravi, da zadostiti kriterijem uspe le dvema od treh (kjer se podatki delijo, se pravi v porazdeljenem računalniškem sistemu). Torej sistem je konsistenten in odporen, odporen in razpoložljiv ali konsistenten in razpoložljiv.

V RDMS-u pa zadovolji konsistentnosti in razpoložljivosti in ker omrežje ni razdeljeno, prideta v poštev le ta dva kriterija.

V NoSQL-u pa moramo izbrati med razpoložljivostjo in konsistentnostjo, saj je razdelitev obvezna. Če se odločimo za konsistentnost lahko počakamo, dokler ni le-ta dosežena. Če pa se odločimo za razpoložljivost, se bodo med strežniki podatki včasih razlikovali.

Glavni cilj pa je dejansko vse kriterije optimizirati in jih ne opustiti, temveč najti kompromis.

(Holmes, 2019)

2.1.3 Shranjevanje v oblaku

Pri oblaku gre za način, kjer se sklicujemo na mrežo (med seboj povezani strežniki, ki so povezani po vsem svetu v podatkovnih centrih). Hub oziroma podatkovno zvezdišče so podatkovni centri, kjer so shranjeni veliki podatki. Uporabo teh strežnikov si preko interneta za plačilo delimo za uporabljanje, shranjevanje datotek, zaganjanje aplikacij ...

Če imamo določeno programsko opremo za dostop do oblaka, si lahko ogledamo svoje dokumente kjer koli smo in enako dajemo dovoljenje tudi drugim. Količina informacij, ki so shranjene v oblaku je gromozanska. Amazon je največji ponudnik storitev (v oblaku), vendar količina podatkov, s katerimi upravlja poslovna skrivnost.

Leta 2017 je Amazon doživel izpad (trajal je cca. 5 ur) vendar je bilo nedostopnih veliko strani, kot so Netflix, Expedia, borza ZDA ... Rečeno je bilo, da naj bi za izpad bila kriva človeška napaka (nehoten odklop strežnikov). Da so se ti ogromni sistemi spet ponovno zagnali, je trajalo dlje kot je bilo pričakovano, vendar uspešno. Vseeno pa je ta dogodek primer, kako je internet dovzeten za izpad, ne glede na to ali je šlo za dejansko človeško napako ali za zlomameren vdor.

(Holmes, 2019)



Slika 3: Shranjevanje v oblaku

vir: (<https://androidayuda.com>)

Zaradi močne rasti podatkov, so na mestu vprašanja, katere podatke je potrebno sploh hraniti, koliko kopij je potrebnih in kako dolgo jih hraniti. Ne moremo pa določiti, kateri podatki so pomembni ali pa nam bodo koristili v prihodnosti. Zato je pomembno podatke stiskati in tako zmanjšati razna podvajanja, zmanjšati potrebe po shranjevanju in izboljšati hitrost pretakanja (v primeru videov). Ločimo izgubno in brezizgubno stiskanje. Za besedila je koristno brezizgubno stiskanje, saj so s tem podatki ohranjeni. Primer so datoteke, ki imajo končnico ZIP, te ohranijo vse informacije, so pa stisnjene. ASCII je standardni nabor za izmenjavo informacij, z njim pa kodiramo podatke. Črke nekega niza dobijo z algoritmom krajše kode. Npr. beseda IZITI ima izvorno 40 bitov, s stiskanjem pa le 7 bitov.

Izgubno stiskanje: izgubno stiskanje pa pri zvočnih in slikovnih datotekah odstrani podrobnosti, in sicer tako, da ne vpliva pretirano na to kako mi vidimo ali slišimo neko datoteko. Podatke teh datotek za vedno odstrani že v prvotni datoteki, tako pridobimo na prostoru. Brezizgubno stiskanje pri teh datotekah ne pride v poštev, saj razmerje stiskanja ni dovolj visoko.

(Holmes, 2019)

2.2 Obdelava:

Mnoge organizacije kot so npr. Google, Amazon, Facebook uporabljajo za obdelavo in shranjevanje informacij Hadoop.

(Edureka, 2023)



Slika 4: Hadoop

vir: (<https://www.linkedin.com>)

2.2.1 Hadoop in MapReduce

MapReduce je primer sistema za analizo velikega podatkovja, torej za obdelavo velikih podatkov. Ta sistem je Hadoopov osrednji funkcionalni del.

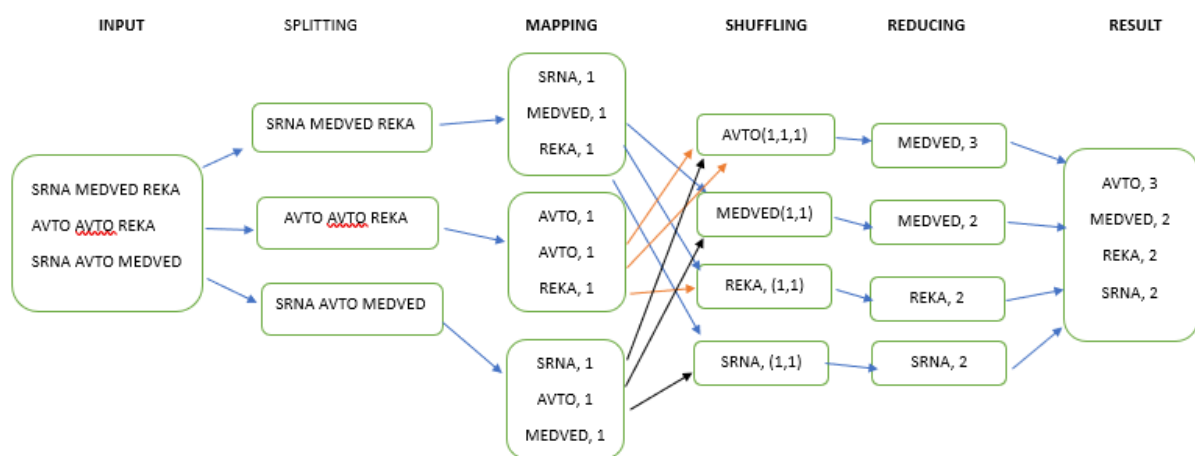
Pri MapReduce je vključenih ogromno procesorjev. Z velikim podatkovjem se spoprijema na način, da podatke razdeli na manjše kose in jih obdela posamezno. Omogoča lokalno obdelavo podatkov, vsako vozlišče deluje z nekim delom.

Za opis kako deluje MapReduce, bom podala zelo pomanjšan primer, ki bo prikazoval proces, kot je pri velikem podatkovju.

Imamo komponente *map* (zemljevid), *shuffle* (premešati) in *reduce* (zmanjšati).

(Holmes, 2019), (Edureka, 2023)

Prva faza je map:



Slika 5: Map

Vir: (Lasten vir)

V fazi *map* se podatki razporedijo. Omogočeno je, da se lahko vsak vhodni dokument bere ločeno, vrstico za vrstico. Datoteke so shranjene v datotečnem sistemu (Hadoop), in sicer ključ-vrednost. Ta skripta vrne za vsako posamezno vrstico pare ključ-vrednost.

(Holmes, 2019)

Faza *shuffling*:

Algoritem premeša in razporedi pare ključ-vrednost. Datoteke razporedi po abecedi. Rezultat faze *shuffling* je poslan naslednji datoteki, ki je pripravljena na skripto *reduce*.

Faza *reduce*:

Tukaj so rezultati prve in druge faze, torej faze map in *shuffling* združeni in vsaka datoteka je poslana v ločeno datoteko, nato pa je omogočen izračun posameznih enot.

Ti rezultati se potem pošljejo končni izhodni datoteki in se shranijo v DFS (so v obliki parov ključ-vrednost).

(Holmes, 2019)

2.2.2 Bloomov filter

Bloomov filter je tehnika (razvita je bila v 70. letih 20. stoletja) oz. zelo koristna metoda za rudarjenje velikih podatkov. Temelj te tehnike je teorija verjetnosti. Uporablja se predvsem tam, kjer je problem shranjevanja in kjer so sezname podatkov.

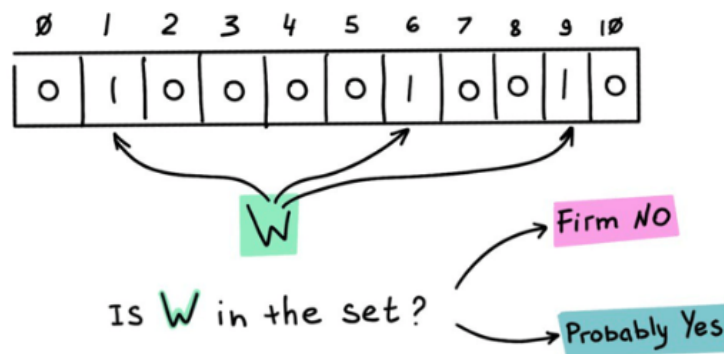
Ta sistem bi naj odgovoril na vprašanje: Ali je nek Y na tem seznamu? Pri velikih podatkih, ki so res ogromni, se pojavi težava, da bi po vseh teh množicah iskali podatke, saj je lahko iskanje zelo počasno. Bloomov filter je za to uporaben, vendar ni 100% natančen (nek element je uvrščen na seznam, čeprav tja ne spada). Vseeno pa je hitro, zanesljivo in prostorsko učinkovito.

Primer Bloomovega filtra je preverjanje spletnih strani. Danes lahko spletne strani vodijo do škodljivih. Filter preveri (natančno in hitro) ali je neka spletna stran lahko škodljiva. Je kot neka črna lista znanih spletnih strani, ki jih preverja, če so le-te varne ali ne.

Drug primer je preverjanje e-poštnih sporočil, če gre morda za nezaželeno pošto, zabljanje ali ribarjenje.

(Holmes, 2019), (Edureka, 2023)

Bloom filter



Slika 6: Bloomov filter

vir: (<https://luminousmen.com>)

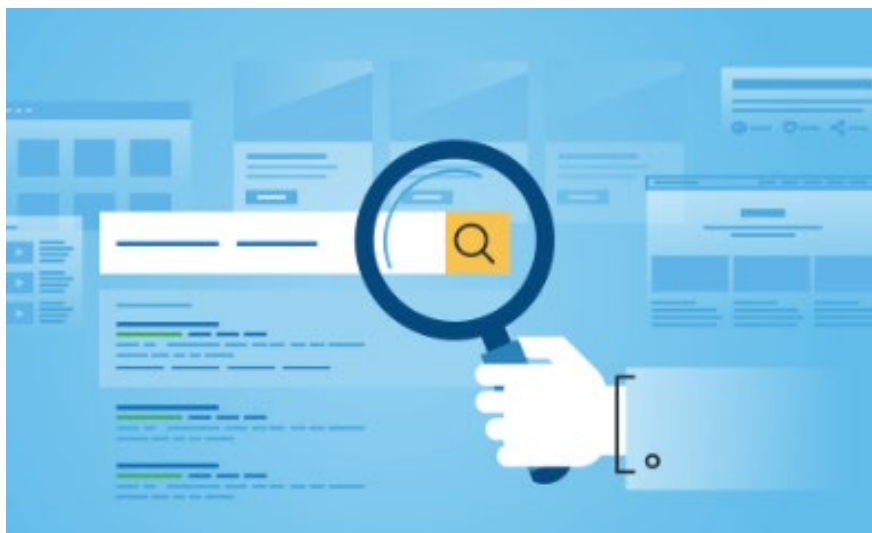
2.2.3 PageRank:

Je metoda, ki razvršča spletne strani glede na iskalne izraze. PageRank uporablja Google in z njim doseže razporeditev strani. Algoritem je narejen tako, da računa vrednost PageRanka glede na število povezav, ki vodijo do spletne strani (več kot je teh povezav, višji je zadetek in višji je rezultat). Razvršča se od 1 do 10. Večinoma ljudje, ko v Google vpišejo željeno vsebino, ki si jo želijo prebrati, odpirajo do prvih štirih zadetkov.

Da je spletna stran tako visoko v Google brskalniku, mora biti PageRank zelo visok. To pomeni, da mora imeti veliko povezav. Zaradi tega je pripeljalo do trgovanja s povezavami (umetno rangiranje oz. prodaja povezav).

Statistik George Box je dejal, da so vsi statistični in znanstveni modeli napačni, vendar so nekateri uporabni. Gre torej zato, da ti modeli koristijo pri neki predstavi, nam podajo neko sliko, s pomočjo katere potem izoblikujemo napovedi in izpeljemo zaključke. Pri analizi malega podatkovja imamo znanstvene metode, pri katerih je nujno vključen človek, kar pa ne drži za veliko podatkovje, kjer je v prevladi stroj in ne znanstvenik. Pri rudarjenju velikih podatkov iščemo koristne informacije, vzorce in korelacije. Te korelacije so med glavnimi težavami, saj zraven uporabnih nastanejo tudi lažne, zgrešene korelacije.

(Holmes, 2019)



Slika 7: PageRank

vir: (<https://www.elegantthemes.com>)

2.3 Lastnosti

V zadnjih letih so bili veliki podatki najprej definirani s »3Vs«, zdaj pa obstaja »6Vs« velikih podatkov, ki so označeni tudi kot značilnosti velikih podatkov:

2.3.1 Volume (obseg)

Že samo ime Big Data je povezano z ogromno velikostjo, torej res ogromno količino podatkov, ki si jo težko predstavljamo. Gre pravzaprav za vse hitreje naraščajočo količino elektronskih podatkov, ki so zbrani in shranjeni. Že samo ime Big Data pove, da je obseg velik, vprašanje pa je kako velik. Ker zajemanje podatkov silno hitro narašča, ne moremo tako zlahka postaviti točne velikosti, saj kar je pred nekaj leti veljalo za ogromno, danes več ne velja. Drugi razlog zakaj ne more obstajati dokončna meja, je shranjevanje podatkov in vrsta podatkov. Kriterij za obseg je zadoščen, ko je podatkov toliko, da jih s tradicionalnimi računalniški in statističnimi metodami ne moremo shraniti, analizirati in zbrati. Pri določitvi vrednosti nekih podatkov ima velikost teh podatkov ključno vlogo. Pri obravnavi velikih podatkov je potrebno upoštevati torej količino.

2.3.2 *Velocity (hitrost)*

Hitrost se nanaša na visoko hitrost kopičenja podatkov. Gre za neprestano pretakanje velikih podatkov iz virov, kot so stroji, omrežja, senzorji, družbeni mediji, pametni telefoni ... Hitrost in obseg sta med seboj zelo povezana, saj hitreje kot so podatki ustvarjeni, več jih je. Na primer sporočila na družabnih omrežjih obidejo svet zelo hitro, saj kar objavimo, si lahko v zelo hitrem času naši prijatelji ogledajo in to objavo delijo dalje.

Pomembno je prav tako omeniti hitrost pretoka podatkov. Gre za to, da so v nekem času nekatere strani bolj obremenjene, saj je obiskanost večja. Takrat so okvare na računalniških sistemih bolj pogoste. Število uporabnikov Facebooka se vsako leto poveča za približno 22%.

Pri hitrosti gre tudi za to kako hitro so podatki elektronsko obdelani. Dober primer so podatki iz avtonomnih vozil, torej podatki iz senzorjev. Ti podatki morajo biti ustvarjeni v realnem času, zelo hitro morajo biti analizirani in brezžično poslani. Tudi navodila morajo biti pravočasno poslana nazaj. Le tako lahko avtonomna vozila delujejo zanesljivo.

2.3.3 *Variety (raznolikost)*

Raznolikost se nanaša na podatke, ki so lahko strukturirani, nestrukturirani ali delno strukturirani. Vsak, ki ima povezavo do medmrežja in seveda računalnik, lahko dostopa do raznih podatkov in komunicira z ljudmi preko družbenih omrežij, e-pošte ... Na spletu se srečujemo s kaotično zbirko podatkov bodisi iz zanesljivih ali dvomljivih virov (ti so nagnjeni k napakam in podvajanju). Večina podatkov iz spleta je nestrukturiranih (besedila, slike, videi itd.) in jih ni mogoče shraniti v obliki vrstic in stolpcev. Na primer sporočila se analizira z namenom, da se razbere izraženo razpoloženje nekoga bodisi pozitivno bodisi negativno, kar ima veliko tržno vrednost. Za takšno kompleksno analizo so potrebne posebne tehnike, ki so zelo razvite, torej uporablja se analiza velikega podatkovja.

2.3.4 Veracity (verodostojnost)

Gre za kakovost zbranih podatkov, natančneje za natančne in zanesljive podatke. Podatki so lahko včasih neurejeni, kakovost in zanesljivost pa je težko nadzorovati. Zgodi se, da pri nekaterih podatkih ne gre zlahka izluščiti pomembnih informacij. Prav tako ni nujno oziroma je na primer na družabnih omrežjih resničnost podatkov zelo vprašljiva. Ti podatki so pogosto vprašljivi, netočni in nezanesljivi. Pri tem lahko pomaga obseg, saj je zbranih več podatkov. Vendar vseeno ni tako enostavno saj lahko pridemo do nasprotnega rezultata (poljubno število naključnih korelacij).

Zelo obsežni podatki lahko povzročijo zmedo, medtem ko lahko manjša količina podatkov povzroči nepopolne ali polovične informacije.

2.3.5 Value (vrednost)

Večina podatkov na primer podjetju ne koristi, saj nimajo vrednosti, razen če jih spremenijo v koristne. Vrednost se uporablja za opis, kako drugim podjetjem podatke prodajajo neka komercialna podjetja. Nato podjetja, ki so jih odkupila, podatke obdelajo z lastno analitiko in jih spremenijo v koristne podatke.

Pri vrednosti gre torej za kakovost rezultatov, kateri so dobljeni z analizo velikega podatkovja. Podatki sami po sebi niso koristni ali pomembni, vendar jih je potrebno pretvoriti v nekaj dragocenega za pridobivanje informacij.

2.3.6 Vizualization (vizualizacija)

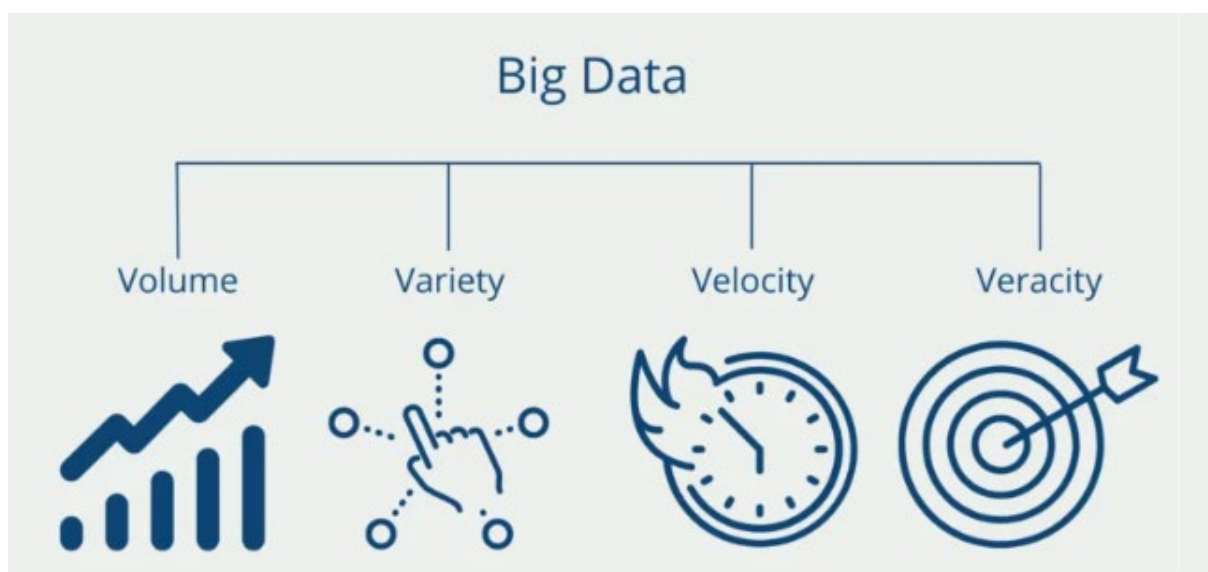
Da lahko razložimo in predstavimo analitične rezultate, potrebujemo vizualizacijo. Za zahtevne podatkovne množice se uporabljajo na primer infografike (katere so statične). Pri tortnem diagramu ali prikazu s stolpci (uporabljata se za vizualizacijo malih podatkovnih množic) je uporabnost omejena pri vizualni interpretaciji velikega podatkovja.

Zaradi kompleksnosti in čedalje večjih podatkovnih množic je bila za opredelitev ključnih lastnosti velikega podatkovja, predlagana uporaba treh v-jev in sicer *volume* (obseg), *variety* (raznolikost) in *velocity* (hitrost) (Doug Laney, 2001). Kasneje je, za postavljanje definicije velikega podatkovja, črka V postala zelo priljubljena. Dodali so izraze *vulnerability* (ranljivost), *viability* (izvedljivost), *value* (vrednost), *visualization* (vizualizacija), *veracity* (verodostojnost). Glavne značilnosti velikega podatkovja pa še vedno ostajajo prvi 4 v-ji in sicer obseg (*volume*), raznolikost (*variety*), hitrost (*velocity*) in verodostojnost (*veracity*). Te štiri značilnosti so velik izziv za upravljanje z velikimi podatki. Na vprašanja, katerih odgovori so naše rešitve teh izzivov, lahko upamo, da dobimo odgovor s pomočjo rudarjenja velikega podatkovja.

(Holmes, 2019)

(Tiao, 2024),

(GeeksForGeeks, 2023)



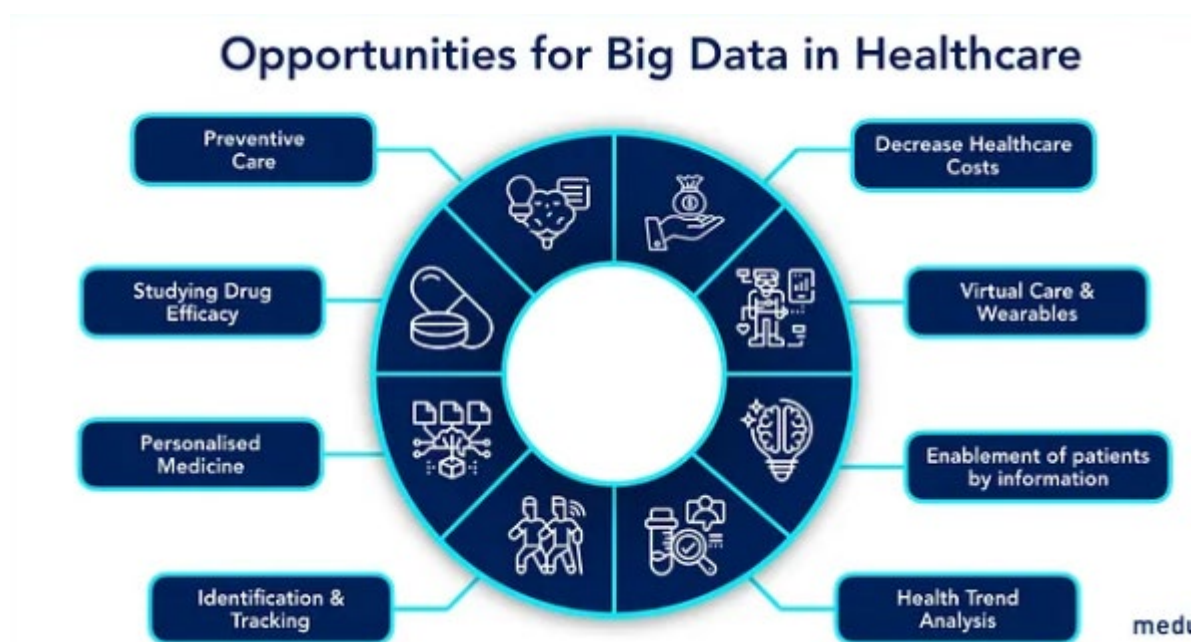
Slika 8: 4V's velikega podatkovja

vir: (<https://databasecamp.de>)

2.4 Veliko podatkovje in medicina

Z analizo velikih podatkov se zdravstveno varstvo pomembno spreminja. Za zdaj njen potencial vključuje napoved epidemij, zdravstveno diagnozo, zmanjševanje stroškov v zdravstvenem sistemu, oceno odzivov javnosti. Medicina uporablja velike podatke predvsem za izboljšanje oskrbe pacientov in zmanjševanje stroškov. Podatki se zbirajo na primer, ko na družabnih omrežjih ljudje govorijo o stranskih učinkih zdravil, s tem pa delijo koristne informacije. Obstajajo spletne strani, kjer strokovnjaki podajajo zdravniške nasvete in ustvarjajo še več informacij, ki so zelo koristne. Ko pacienti obiščejo zdravnika, se s tem zbirajo tudi podatki. Elektronsko zabeležene (elektronske kartoteke) so razne podrobnosti od anamneze, zdravil, testov, rezultatov.

(Holmes, 2019)



Slika 9: Veliki podatki in medicina

Vir: (<https://medium.com>)

3 SPOTIFY

3.1 Lastnosti

Spotify je digitalna glasba, podcast in video storitev, ki omogoča dostop do milijonov pesmi in drugih vsebin ustvarjalcev z vsega sveta. Osnovne funkcije Spotifyja so predvajanje glasbe in so brezplačne, lahko pa se tudi odločimo za nakup oziroma za nadgradnjo na Spotify Premium. Ne glede na to ali to dokupimo, dobimo priporočila glede na naš okus, ustvarimo zbirke glasbe in podcastov. Spotify je na voljo na številnih napravah, kot so računalniki, telefoni, tablični računalniki, zvočniki, televizorji, avtomobili. Spotify omogoča dostop do glasbe in podcastov preko njihove aplikacije. Njihovo licenciranje pomeni, da vsebine ni mogoče izvoziti izven aplikacije. Glasbo lahko delimo na skoraj kateri koli platformi družbenih medijev. Svoj profil na Spotifyu lahko povežemo z različnimi aplikacijami, kot so aplikacije za vstopnice (za priporočanje prihajajočih nastopov), aplikacije za zmenke (povežemo se lahko z ljudmi, ki imajo podoben glasbeni okus). Prednost je predvsem ta, da priporoča glasbo na podlagi naših preteklih glasbenih izbir. Discover Weekly nam ponuja novosti, ki jih lahko preizkusimo glede na naše pretekle izbire poslušanja. Ponuja razne upravljalce, kot so predvajanje/pavza, ponavljanje pesmi/seznama, shranjevanje priljubljenih. Prav tako ima novost, imenovano DJ, ki ustvari radijsko postajo glede na naš glasbeni okus in s tem pomaga uporabnikom pri odkrivanju novih. Iskanje lahko filtriramo glede na zvrst in razpoložanje. Ustvarimo lahko skupne sezname predvajanj (naših in prijateljevih), zato je Spotify imenovan kot najbolj družaben proti drugim aplikacijam za pretakanje glasbe.

(6MonthsLater, 2020), (Spotify, brez datuma)



Slika 10: Spotify

vir: (<https://mashable.com>)

3.2 Spotify in Big Data

Razvijanje prilagojene vsebine:

Spotify uporablja velike podatke (le-te so ustvarili uporabniki) za razvoj vsebine za katero bodo uporabniki menili, da je za njih uporabna oz. se ujema z njihovim okusom. Cilj je zadovoljiti uporabnike, da postanejo dolgotrajne stranke. To je bilo doseženo s sprejetjem različnih algoritmov umetne inteligence.

Leta 2012 je bila uvedena funkcija Discover. Pojavil se je seznam predvajanj skladb, ki so jih izdali priljubljeni umetniki uporabnika. Kasneje se je ta funkcija razvila, in sicer predlagala se je zbirka skladb, ko je uporabnikov seznam predvajanja bil končan.

Najbolje se trenutno obnese Discover Weekly. Sestavljen je iz algoritmov, ki ustvarjajo seznam predvajanj, kateri je ekskluziven za uporabnika. Algoritem pregleda sezname predvajanj drugih uporabnikov, da ugotovi podobnosti med skladbami, nato pa te podatke uporabi za razvoj novega seznama skladb. Ta nov seznam se ujema z uporabnikovimi željami za poslušanje glasbe. Vsak uporabnik ima svoj osebni profil okusa, sestavljen iz mikro zvrsti, ki igra vlogo pri prilagajanju teh seznamov. To deluje tako, da na poslušalec posluša neko skladbo in jo po manj kot 30. sekundah preskoči ali jo ustavi. Algoritem to zazna in te skladbe ali njej podobnih ne doda na seznam. Če pa pesem uporabnik posluša celo ali večkrat, pa algoritem zazna kot pozitiven odziv.

(Rangaiah, 2021)

3.2.1.1 Digitalizacija okusa:

Daily Mixes so Spotifyjevi seznammi predvajanj, ki se razlikujejo od zvrsti, h katerim se uporabniki običajno nagibajo in so običajno sestavljeni iz pesmi, ki jih je uporabnik dodal na svoje sezname predvajanj ali jih shranil. Gre za sveže izvajalce ali albume, ki jih uporabnik ne pozna. Spotify tukaj doda dodatno vrednost, saj doda na seznam nekaj zanimivih pesmi, ki jih uporabnik ne pozna in seznam naredi bolj živahen.

(Rangaiah, 2021)

3.2.1.2 Ciljni oglasi

Spotify je dobil ogromne količine podatkov, katere je pridobil od uporabnikov medtem ko, je izboljševal izkušnje strank. Ti podatki so pomagali, da bi posodabljali svoje oglaševalske akcije in ciljali na stranke na kar se da privlačen način.

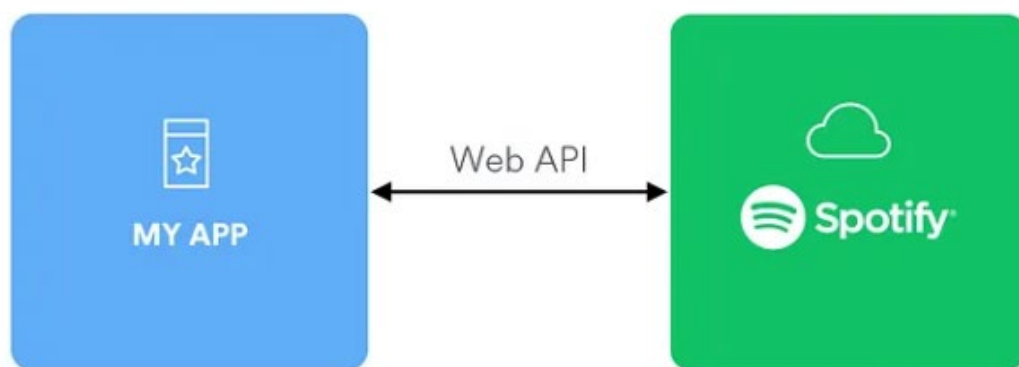
Platforma preuči znanje, ki ga pridobi s pomočjo svojih poslušalcev in to uporabi za ustvarjanje oglasov, ki so ciljno usmerjeni na ciljne poslušalce.

(Rangaiah, 2021)

3.3 Spotify in programski vmesniki (API)

Spletni Spotify API ponuja široko paleto funkcij za razvijalce. Pridobimo lahko podatke svojega najljubšega izvajalca, albuma, oddaje. Poiščemo lahko vsebino Spotify, nadziramo in komuniciramo s predvajanjem, predvajamo in nadaljujemo, poiščemo položaj ali pridobimo svojo čakalno vrsto. Upravljamo lahko svojo osebno knjižnico tako da, ustvarimo nov seznam predvajanja in mu dodamo skladbe.

(Spotify, brez datuma)



Slika 11: Spotify API

Vir: (<https://medium.com>)

4 YOUTUBE MUSIC

4.1 Lastnosti

YouTube Music je spletna storitev za pretakanje glasbe in videa. Uporabljamo jo lahko na računalniku ali mobilnih napravah, vendar pa ne vsebuje neposredno podkastov. Googlovi podatki sporočajo, da je trenutno več kot 80 milijonov uradnih skladb. Brskamo lahko med izvajalci, žanri in skladbami. Možnost je tudi, da lahko naložimo naše lastne skladbe iz računalnika na YouTube Music. Osnovna spletna storitev je brezplačna, lahko pa jo nadgradimo s plačljivo naročnino. Ta nam omogoča dodatne možnosti, kot so predvajanje v ozadju, ko imamo napravo izklopljeno ali ko uporabljamo katero drugo aplikacijo. Dodatno je tudi to, da se nam med poslušanjem glasbe ne pojavljajo oglasi. Skladbe si lahko shranimo in jih poslušamo tudi ko nismo povezani na splet. (Rangaiah, 2021), (Youtube Music, brez datuma)



Slika 12: YouTube Music

vir: (<https://www.alphr.com>)

4.2 YouTube Music in Big Data

Ko predvajamo neko glasbeno vsebino na YouTube Music, ustvarimo podatke. Te podatke YouTube uporabi za svoje poslovanje in zagotavljanje boljših uporabniških izkušenj. Ta

spletna storitev ima možnost komentiranja skladb, kar doprinese ustvarjalcem neke odgovore, kaj ljudje želijo poslušati. Vendar pa so nekateri komentarji ustvarjeni z namenom, da bi preusmerili promet iz YouTube na svoje spletne strani. YouTube je to težavo rešil z uporabo modela strojnega učenja, za odkrivanje neželenih komentarjev v realnem času. Ta model torej razvrsti komentarje.

Veliki podatki se uporabljajo predvsem za to, da ugotovijo, glede na našo poslušano zgodovino, kaj nam odgovarja, kakšen okus za pesmi imamo in nam tako priporoča naslednjo skladbo.

Ker so glavni vir dohodka oglasi, YouTube prikazuje oglase na podlagi naših preteklih zanimanj, zgodovine ogledov in iskanja. (NewsHour, 2014)

4.3 YouTube in programski vmesniki (API)

YouTube API nam omogoča, da lahko vključimo funkcije, ki se običajno izvajajo na spletnem mestu YouTube, v svoje spletno mesto ali aplikacijo. Vsebuje sezname, ki določajo različne vrste virov, katere lahko pridobimo z uporabo tega API-ja. Podpira tudi metode za posodabljanje, vstavljanje in brisanje številnih teh virov.

Imamo vodnik, ki nam pojasnjuje kako uporabiti API za izvajanje vseh teh operacij. Vir predstavlja vrsto elementa, ki je del YouTube, kot je videoposnetek, seznam predvajanja ali naročnina. Viri so predstavljeni kot objekti JSON. (Youtube, brez datuma)

5 PRIMERJAVA MED OBEMA PONUDNIKOMA

5.1 Primerjava uporabniških lastnosti

Največja razlika med ponudnikoma so naprave, ki jih podpirajo.

Spotify je na več napravah kot YouTube Music. Na primer Amazonove naprave se trenutno povezujejo samo s Spotifyem in ne tudi z YouTube Music. Spotify ima prav tako več interakcij z bolj nenavadnimi napravami, kot je na primer Tesla.

Obe storitvi ponujata brezplačno različico. YouTube Music nam za razliko od Spotifya omogoča predvajanje katere koli skladbe, preskočimo lahko poljubno število skladb.

Spotify omogoča predvajanje glasbe v ozadju, ko se na telefonu zaklene ekran ali zapremo aplikacijo. YouTube v enakih situacijah glasbo ustavi.

Pri obeh ponudnikih pa imamo tudi možnost dokupa dodatnih storitev, pri katerih se pri obeh izognemo oglasom (Premium). Spotify nam omogoča, da lahko poslušamo karkoli, preskočimo poljubno število skladb. YouTube pa nam omogoči, da lahko poslušamo glasbo v ozadju. Pri obeh ponudnikih lahko prenesemo glasbo na napravo. Na Spotifyu za razliko od YouTube Music lahko poslušamo podkaste. YouTube Music in Spotify sta si zelo podobna, z nekaj razlikami. Uporabnik se odloči glede na njegovo potrebo. Glavna prednost YouTube Music je, da v svojo knjižnico naložimo glasbo medtem ko, Spotify ne omogoča nalaganja datotek neposredno v oblak, vendar ima funkcijo lokalnih datotek, katera omogoča predvajanje glasbe v naši napravi preko aplikacije. Glavna prednost Spotifya pa je, da glede na to, kar mi radi poslušamo ustvarja glasbene mikse oz. ustvari listo glasbe, katera je zasnovana po naši zgodovini poslušanja glasbe. (6MonthsLater, 2020)

Tabela 1: Primerjava med ponudnikoma

	SPOTIFY	YOUTUBE	SPOTIFY	YOUTUBE MUSIC
--	---------	---------	---------	------------------

		MUSIC	PREMIUM	PREMIUM
Preskok skladb	6 skladb/1uro	DA	DA	DA
Predvajanje v ozadju	DA	NE	DA	DA
Preskok oglasov	NE	Da, po 5. sekundah	DA	DA
Poslušanje/prenos brez povezave	NE	NE	DA	DA
Stroški (Slovenija)	/	/	6,49€ mesečno	5,99€ mesečno
Stroški (Anglija)	/	/	£11.97 mesečno (14,09€)	£12 mesečno (14,10€)
Stroški (Nemčija)	/	/	5,99€ mesečno	11,99€ mesečno

Cene veljajo od 2024 naprej in se lahko spreminjajo.

5.2 Primerjava v povezavi z *Big Data*

Pri obeh (YouTube Music in Spotify) se Big Data uporabljajo, da ponudnika naredita uporabnikom čim boljše izkušnje. Cilj je torej, da so uporabniki zadovoljni in da postanejo dolgoletne stranke. Big Data pomaga pri tem, da ponudnika pridobita podatke posameznikov in s tem oba ustvarjata glasbene liste, ki so namenjene, da ljudje dobijo nove ideje kaj poslušati, glede na njihov okus prejšnjih skladb. Pri obeh je torej bistvo uporabe Big Data enako.



Slika 13: Spotify vs YouTube Music

vir: (<https://createbytes.medium.com>)

6 SEZNAM GLASBE PRI PONUDNIKU

6.1 Razvojno okolje in JavaScript

Pri izdelavi skripte sem kodo pisala v JavaScript jeziku, in sicer v WebStorm razvojnem okolju.

Za WebStorm in JavaScript sem se v prvi vrsti odločila zato, saj sem se z njima srečala že v času študija. WebStorm je integrirano razvojno okolje, ki vsebuje vse kar potrebujemo pri razvoju z JavaScript.

Je pameten urejevalnik z dokončevanjem kode in sprotnim odkrivanjem napak.

Ima navigacijo in iskalnik (po celotni kodni bazi), kar nam prihrani čas, še posebej pri večjih projektih.

6.2 Skripta za izvoz glasbenega seznama

Skripto sem napisala z namenom, da lahko iz Spotifya prenesem seznam skladb v JSON file.

Kasneje se lahko uporabi za prenos seznama sklad na primer na Apple Music. Sicer že takšne rešitve obstajajo, vendar so plačljive.

Za to skripto sem uporabila Spotify API. Napisala sem skripto v Node.js.

Spotify API ima veliko primerov in knjižnic, s katerimi lahko napišemo skripto.

Uporabila sem knjižnico *spotify-web-api-node*, katera nam dovoljuje vse, kar potrebujemo, da ustvarimo skripto.

Dovoljuje nam, da pridobimo uporabnikov seznam skladb, vključno s podrobnostmi skladb, nato pa jih zapiše v JSON datoteki.

Najprej sem si na Spotify API for Developers brezplačno ustvarila svoj račun. Dobila sem ID (identifikacijsko številko) in skrivni ključ (angl. *client secret*). Prav tako sem morala nastaviti *RedirectURI* (<http://localhost:8888/callback>) zaradi gostovanja. V packet json imamo *express* in *spotify-web-api-node*.

```

1  {
2    "name": "spotify_playlist_export",
3    "version": "1.0.0",
4    "description": "",
5    "main": "index.js",
6    "scripts": {
7      "test": "echo \"Error: no test specified\" && exit 1"
8    },
9    "keywords": [],
10   "author": "",
11   "license": "ISC",
12   "dependencies": {
13     "express": "^4.19.2",
14     "spotify-web-api-node": "^5.0.2"
15   }
16 }

```

Slika 14: Koda JavaScript

Vir: (Lasten vir)

Naredila sem skripto index.js. Aplikacija vsebuje *scopes*, s katerimi dostopamo do uporabnika.

Scopes

Scopes provide Spotify users using third-party apps the confidence that only the information they choose to share will be sent to the app.

Pre-requisites

Scopes are needed when implementing some of the authorization grant types. Make sure you have read the [Authorization](#) page.

List of Scopes

- Images
 - [ugc-image-upload](#)
- Spotify Connect
 - [user-read-playback-state](#)
 - [user-modify-playback-state](#)
 - [user-read-currently-playing](#)
- Playback
 - [app-remote-control](#)
 - [streaming](#)
- Playlists
 - [playlist-read-private](#)
 - [playlist-read-collaborative](#)
 - [playlist-modify-private](#)
 - [playlist-modify-public](#)

Slika 15: Scopes

(Spotify, brez datuma)

```

let SpotifyWebApi = require('spotify-web-api-node');
const express = require('express')

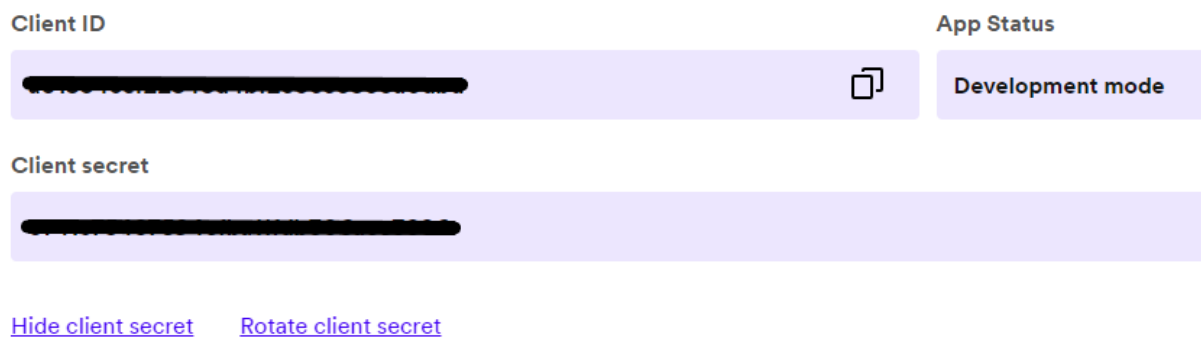
const scopes = [
  'ugc-image-upload',
  'user-read-playback-state',
  'user-modify-playback-state',
  'user-read-currently-playing',
  'streaming',
  'app-remote-control',
  'user-read-email',
  'user-read-private',
  'playlist-read-collaborative',
  'playlist-modify-public',
  'playlist-read-private',
  'playlist-modify-private',
  'user-library-modify',
  'user-library-read',
  'user-top-read',
  'user-read-playback-position',
  'user-read-recently-played'
]

```

Slika 16: Koda JavaScript

Vir: (Lasten vir)

Potem potrebujemo client ID in client Secret. Oboje najdemo na našem *Dashboardu*.



Slika 17: Client ID in Client secret

Vir: (Lasten vir)

```
let spotifyApi = new SpotifyWebApi({ credentials: {
  clientId: 'a...',
  clientSecret: '...',
  redirectUri: 'http://localhost:8888/Callback'
}});
```

Slika 18: Koda Javascript

Vir: (Lasten vir)

get endpoint: prvi korak je zahtevati pooblastilo od uporabnika tako, da lahko naša aplikacija dostopa do virov Spotifyja. Za to mora naša aplikacija zgraditi in poslati zahtevo *GET*.

Koda JavaScript implementira metodo *login* z uporabo orodja *Express*, za sprožitev zahteve za avtorizacijo:

```
const app = express();

app.get('/login', (req, res) => {
  res.redirect(spotifyApi.createAuthorizeURL(scopes));
});

app.get('/callback', (req, res) => {
  const error = req.query.error;
  const code = req.query.code;
  // const state = req.query.state;

  if (error) {
    console.error('Callback Error:', error);
    res.send(`Callback Error: ${error}`);
    return;
  }
}
```

Slika 19: Koda JavaScript

Vir: (Lasten vir)

V primeru *errora* izpišemo *error*.

```

spotifyApi
  .authorizationCodeGrant(code)
  .then(data => {
    const access_token = data.body['access_token'];
    const refresh_token = data.body['refresh_token'];
    const expires_in = data.body['expires_in'];

    spotifyApi.setAccessToken(access_token);
    spotifyApi.setRefreshToken(refresh_token);

    console.log('access_token:', access_token);
    console.log('refresh_token:', refresh_token);

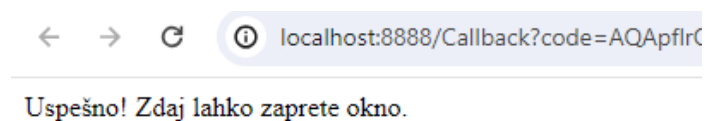
    console.log(
      `Uspešno pridobljen access token. Poteče v ${expires_in} s.`
    );
    res.send( body: 'Uspešno! Zdaj lahko zaprete okno.' );
  });

```

Slika 20: Koda JavaScript

Vir: (Lasten vir)

Ko zaženemo index.js, se v našem konzolnem oknu prikaže *http://localhost:8888/login*, na katerega kliknemo. Če deluje pravilno, se nam mora v brskalniku prikazati napis: *»Uspešno! Zdaj lahko zaprete okno«*.



Slika 21: Prikaz po zagonu index.js

Vir: (Lasten vir)

Nato se v naši konzolni okni prikaže *access token*, katerega moramo kopirati. Poteče v eni uri.

Naredimo novo skripto *getMe.js*. Tukaj sem uporabila naš *access token*.

```
const fs = require('fs')
const SpotifyWebApi = require('spotify-web-api-node');
const token = "BQDeH8sEkREJqVme4c5Wi4YYA1txhgkF7tD_d5MvR2h6f81zdVduBDYNJg40s7t6cEZW6I3X_s5-gNyh61kc";

const spotifyApi = new SpotifyWebApi();
spotifyApi.setAccessToken(token);
let me;
```

Slika 22: Koda JavaScript

Vir: (Lasten vir)

Uporabili bomo *fs* (file system), da bomo shranili datoteke v JSON.

Metoda *getMyData*: Tukaj pokličemo Spotify API da ustvari vse podatke o našem uporabniku.

```
display_name: 'Tina',
external_urls: {
  spotify: 'https://open.spotify.com/user/316q3owk6xz73ta5pv3vqocy4ffm'
},
href: 'https://api.spotify.com/v1/users/316q3owk6xz73ta5pv3vqocy4ffm',
id: '316q3owk6xz73ta5pv3vqocy4ffm',
images: [
  {
    url: 'https://platform-lookaside.fbsbx.com/platform/profilepic/?asid=10229',
    height: 64,
    width: 64
  },
  {
    url: 'https://platform-lookaside.fbsbx.com/platform/profilepic/?asid=10229',
    height: 300,
    width: 300
  }
],
type: 'user',
uri: 'spotify:user:316q3owk6xz73ta5pv3vqocy4ffm',
followers: { href: null, total: 0 },
country: 'SI',
product: 'free',
explicit_content: { filter_enabled: false, filter_locked: false },
email: 'tinaock@hotmail.com'
```

Slika 23: Koda JavaScript

Vir: (Lasten vir)

V nadaljevanju je asinhrona funkcija, *getUserPlaylist*, katera izpiše ime albuma in ID albuma uporabnika. Vse to shrani v JSON datoteko.

```
async function getUserPlaylists(userName) {
    const data = await spotifyApi.getUserPlaylists(userName)

    console.log("-----")
    let playlists = []

    for (let playlist of data.body.items) {
        console.log(playlist.name + " " + playlist.id)
        let tracks = await getPlaylistTracks(playlist.id, playlist.name);
        // console.log(tracks); Preveč podatkov, zato ne rabim

        const tracksJSON = { tracks }
        let data = JSON.stringify(tracksJSON);
        fs.writeFileSync( file: playlist.name+'.json', data); //s tem shranim v json file
    }
}
```

Slika 24: Koda JavaScript

Vir: (Lasten vir)

Naslednja funkcija je asinhrona funkcija *getPlaylistTracks*. Deluje tako, da pod ime albuma izpisuje ime skladbe in njenega izvajalca. Število skladb je omejeno na 100.

```
async function getPlaylistTracks(playlistId, playlistName) {

    const data = await spotifyApi.getPlaylistTracks(playlistId, { options: {
        offset: 1,
        limit: 100,
        fields: 'items'
    }})

    let tracks = [];

    for (let track_obj of data.body.items) {
        const track = track_obj.track
        tracks.push(track);
        console.log(track.name + " : " + track.artists[0].name)
    }

    console.log("-----")
    return tracks;
}

getMyData();
```

Slika 25: Koda JavaScript

Vir: (Lasten vir)

Asinhrono funkcijo uporabimo, ko zahtevamo podatke iz omrežja in jih želimo shraniti v datotečni sistem. Async beseda pretvori funkcijo v asinhrono, zaradi katere vrne *promise*. Beseda *await* se uporabi v asinhroni funkciji z namenom, da začasno ustavi njeno izvajanje in počaka, da se *promise* razreši, preden nadaljuje.

Ko zaženemo *getMe.js* se nam v konzolnem oknu prikažejo podatki uporabnika, albumi, katere skladbe vsebujejo in njihove izvajalce.

```
display_name: 'Tina',
external_urls: {
  spotify: 'https://open.spotify.com/user/316q3owk6xz73ta5pv3vqocy4ffm'
},
href: 'https://api.spotify.com/v1/users/316q3owk6xz73ta5pv3vqocy4ffm',
id: '316q3owk6xz73ta5pv3vqocy4ffm',
images: [
  {
    url: 'https://platform-lookaside.fbsbx.com/platform/profilepic/?asid=10229193013187043&height=500&width=500',
    height: 64,
    width: 64
  },
  {
    url: 'https://platform-lookaside.fbsbx.com/platform/profilepic/?asid=10229193013187043&height=300&width=300',
    height: 300,
    width: 300
  }
],
type: 'user',
uri: 'spotify:user:316q3owk6xz73ta5pv3vqocy4ffm',
followers: { href: null, total: 0 },
country: 'SI',
product: 'free',
explicit_content: { filter_enabled: false, filter_locked: false },
email: 'tinaock@hotmail.com'
```

Slika 26: Prikaz v konzolnem oknu

Vir: (Lasten vir)

```
-----
My Playlist #2 3b1FCxRDQ5se72xNaA2isV
Klinik : Siddharta
Klinik : Siddharta
Na Soncu : Siddharta
Platina : Siddharta
Na Soncu : Siddharta
Sam' prjatla : Big Foot Mama
Nastalo bo : Siddharta
Od Višine Se Zvrtil : Vlado Kreslin
-----
Hi 1M5URsjMrzdLUSgk0PkPk3
OE0 : Azet
Heart Attack : Noizy
Karussell : Zuna
Jede Nacht : Schubi AKpella
-----
```

Slika 27: Prikaz v konzolnem oknu

Vir: (Lasten vir)

6.3 Ugotovitve

Ker sem se prvič spoprijemala z WEB API-jem, sem potrebovala nekaj časa, da sem ugotovila kako in zakaj se uporablja.

Na Spotify API-ju pridobimo metapodatke iz vsebine Spotify. Omogoča nam, da ustvarjamo aplikacije, ki lahko komunicirajo s storitvijo pretakanja Spotify, kot je pridobivanje metapodatkov vsebine, pridobivanje priporočil, ustvarjanje in upravljanje seznamov predvajanja ali nadzor predvajanja.

Moja skripta je bila ustvarjena s Spotify WEB API for developers, kjer sem uporabila knjižnico *spotify-web-api-node*, katera nam dovoljuje dostop do vsega, kar potrebujemo.

Spotify API je vmesnik, ki ga lahko programi uporabljajo za pridobivanje in upravljanje podatkov Spotify preko interneta. API uporablja protokol http.

Pridobiti moramo kodo za dostop (*access token*), katero lahko program posreduje pri klicih spletnemu API-ju. Razvijalec se mora registrirati na spletnem mestu za razvijalce Spotify. S tem pridobi ID in skrivni ključ. Vse to je potrebno uporabiti v kodi programa. Ko se program izvaja, sta ID in skrivni ključ posredovana storitvi Spotify. Ta preveri dejansko pristnost uporabnika in išče uporabnikovo dovoljenje za dostop do podatkov.

S to skripto sem lahko prenesla seznam skladb v JSON datoteko. Skripta se lahko uporabi za nadaljnji razvoj programa, na primer za prenos seznama skladb iz Spotifyja na Apple Music, kar je lahko za nadaljnjega razvijalca dobra podlaga. Sicer takšne rešitve že obstajajo, vendar so plačljive.

Delo z Spotify WEB API-jem mi je bilo zanimivo, prav tako pa sem pridobila veliko novega znanja.

6.4 Napotki za nadaljnje delo

V diplomski nalogi sem se osredotočila na veliko podatkovje in povezavo dveh glasbenih ponudnikov z le-tem. Veliko podatkovje je izjemno široko področje, zato sem se osredotočila na bistvene podatke in jih zajela tako, da sem lahko delo povezala v celoto v povezavi s pretočnimi glasbenimi vsebinami. To delo je dobra iztočnica za nekoga, ki ga to področje zanima, saj bi lahko temo nadgradil.

Večja poglavja, katera sem bolj kot ne le omenila, so predvsem shranjevanje velikih podatkov, podrobna raziskava o sistemih ter obdelava velikih podatkov. Te zadeve so komplekse in nezaključene, saj se strokovnjaki še dandanes ukvarjajo s sistemi in jih izboljšujejo. Vsako poglavje bi se lahko razčlenilo in podrobno samostojno raziskovalo. Kot vemo, se tehnologija in življenje s tehnologijo nenehno razvija in eksponentno napreduje.

Diplomsko delo je lahko tudi dobra iztočnica pri nadaljevanju oziroma nadgrajevanju skripte za izvoz glasbenega seznama. Napisal bi se lahko program, kateri bi ta seznam glasbe prenesel na novega ponudnika, kot je Apple Music ali YouTube Music..

7 SKLEP

Diplomska naloga je zahtevala poglobljeno raziskovanje velikega podatkovja in programiranja preko WEB Spotify API-ja.

Dandanes tehnologija konstantno napreduje. Večina nas uporablja pametne telefone, računalnike, internet, tablice, pametne ure, pametne televizorje itd. Preko naprav si izmenjujemo podatke, brskamo po spletu, avtomobili snemajo podrobnosti ipd. Pri vsem tem se ustvarjajo ogromne količine podatkov, ki se shranjujejo in obdelujejo. Potrebno je ugotoviti katere podatke je vredno shraniti in za koliko časa. Nekateri podatki so zelo pomembni in koristni.

V nalogi sem postavila štiri hipoteze:

H1: Big Data imajo 5V's (5 glavnih lastnosti). To hipotezo sem delno potrdila v poglavju 2.3.

Ugotovila sem, da so bile lastnosti velikega podatkovja definirane s 3V's, danes pa obstaja že 6V's, ki so označene tudi kot značilnosti. V tem poglavju sem opisala 6 lastnosti, ki definirajo Big Data, in sicer *Volume* (obseg), *Velocity* (hitrost), *Variety* (raznolikost), *Veracity* (verodostojnost), *Value* (vrednost) in *Vizualization* (vizualizacija). Kljub temu, da poznamo več teh lastnosti, ostajajo glavne štiri, in sicer obseg, raznolikost, hitrost in verodostojnost. Upravljanje z velikim podatkovjem še vedno ostaja precej velik izziv.

H2: Spotify in YouTube Music sta primerljiva iz stališča uporabnika. To hipotezo sem delno potrdila, saj je odvisna od tega ali kot uporabnik zakupimo dodatne storitve ali ne. V poglavju 5.1. sem to predstavila s tabelo, iz katere je razvidno, da sta ponudnika primerljiva predvsem, ko ju zakupimo, če pa ju uporabljamo brezplačno pa imata kar nekaj razlik.

H3: Spotify in YouTube Music sta primerljiva iz stališča Big Data. To hipotezo sem potrdila, in sicer sem to podrobneje opisala v poglavju 5.2. Pri obeh je bistveno, da uporabljata velike podatke za razvoj vsebine, ki bo uporabnikom doprinesla dodatno vrednost in izboljšala izkušnjo. To je predvsem zmožnost ugotovitve kaj uporabnik rad posluša, ponudnik pa zmore prilagajati glasbene sezname glede na uporabnikov okus glasbe.

H4: Izdela se lahko skripta za izvoz seznama glasbe iz glasbenega ponudnika. To hipotezo sem potrdila. S to skripto lahko iz Spotifyja prenesem seznam skladb, in sicer v JSON

datoteko. To še ni dokončana različica, temveč gre za osnutek iz katerega lahko razvijalci naredijo dokončno aplikacijo.

Bistvo diplomske naloge je ugotovitev, da so Big Data oz. veliko podatkovje zelo pomembni. Doprinesejo velike razlike v poslovanju, zdravstvu, izboljšujejo odločanje zaradi pridobitve pomembnih vpogledov, tveganj, vzorcev in trendov. Zaradi velikega podatkovja organizacije dobijo ogromno informacij, katere potrebujejo za boljše odločitve. Zajemajo tržne trende, potrošniške navade, stranke in uporabniki pa imajo zaradi njih boljše izkušnje. Izpolnjujejo potrebe ciljnih skupin, izboljšajo se odločitve o razvoju izdelkov, prihranijo stroške, izboljšajo učinkovitost. Velike podatke je mogoče uporabiti za prepoznavanje rešljivih problemov, izboljšanje zdravstvenega varstva in tudi kot boj proti revščini.

Big Data oziroma veliko podatkovje je danes že v ospredju, potreba po koristnih podatkih pa skupaj z razvojem tehnologije še narašča.

Problem obdelave in shranjevanja velikega podatkovja še vedno ostaja, zato strokovnjaki stremijo k temu, da se vedno bolj približajo tem ekstremno kompleksnim procesom in sistemom. Zaradi takšne količine podatkov, ki eksponentno rastejo, se je z razširljivostjo težko spopadati. Večja kot je moč računalnika, večjo razširljivost lahko dosežemo, kar pa ima tudi svoje meje. Baze postajajo počasne in tudi nezanesljive.

Veliki podatki imajo izjemno pomembno vlogo v medicini. S pomočjo le-teh se lahko napovedujejo epidemije, zdravstvene diagnoze, zmanjšajo se lahko stroški v zdravstvenem sistemu in v prvi vrsti izboljša oskrba pacienta.

Ocena pomembnosti velikega podatkovja je zelo visoka.

8 SEZNAM UPORABLJENIH VIROV

- 6MonthsLater. (14. Avgust 2020). *Spotify vs. YouTube Music - Which is Better?* Pridobljeno iz <https://www.youtube.com/watch?v=sEX9xIUPfnM>
- Academia. (18. december 2015). *Gostujoče predavanje sodelavcev časopisa Večer*. Pridobljeno iz Academia Novice: <http://academia.si/novice/gostujoce-predavanje-sodelavcev-casopisa-vecer/>
- Bakratsas, M., Basaras, P., Katsaros, D., & Tassiulas, L. (junij 2017). *Raziskave velikih količin podatkov*. Pridobljeno iz <https://www.sciencedirect.com/science/article/abs/pii/S221457961730014X>
- Bučar, F. (2012). *Rojstvo države*. Ljubljana: Didakta.
- Edureka. (2023). *Big Data&Hadoop Full Course in 12 Hours [2024] | BigData Hadoop Tutoria For Beginners*. Pridobljeno iz <https://www.youtube.com/watch?v=9QxZhapbo0o&t=30603s>
- GeeksForGeeks. (21. februar 2023). *Geeks for geeks*. Pridobljeno iz <https://www.geeksforgeeks.org/5-vs-of-big-data/>
- Holmes, D. E. (2019). *Veliko podatkovje. Zelo kratek uvod*. Krtina: V Ljubljani : Krtina, 2019.
- NewsHour, P. (28. November 2014). *youtube*. Pridobljeno iz How the music industry uses big data to create the next big hit: <https://www.youtube.com/watch?v=R-LTyevbcgo>
- Rangaiah, M. (7. januar 2021). *analyticssteps*. Pridobljeno iz How Spotify is using Big Data to enhance customer experience: <https://www.analyticssteps.com/blogs/how-spotify-using-big-data>
- Simplilearn. (2019). *Big Data In 5 Minutes | What Is Big Data?| Big Data Analytics | Big Data Tutorial | Simplilearn*. Pridobljeno iz <https://www.youtube.com/watch?v=bAyrObl7TYE>
- Spotify*. (brez datuma). Pridobljeno iz Spotify for developers: <https://developer.spotify.com/documentation/web-api>
- Spotify*. (brez datuma). Pridobljeno iz <https://open.spotify.com/>

Tiao, S. (11. marec 2024). *What Is Big Data?* Pridobljeno iz Oracle:

<https://www.oracle.com/uk/big-data/what-is-big-data/>

YouTube. (brez datuma). Pridobljeno iz Data API: <https://developers.google.com/youtube/v3>

YouTube Music. (brez datuma). Pridobljeno iz <https://music.youtube.com/>