

## ULOGA I ZNAČAJ BAZA PODATAKA ZA IZRADU POSLOVNIH APLIKACIJA

### THE ROLE AND SIGNIFICANCE OF DATABASES FOR THE CREATION OF BUSINESS APPLICATIONS

**Aleksandra Zečević**

Ekonomski fakultet, Univerzitet u Beogradu, Srbija  
aleksandra.zecevic@ekof.bg.ac.rs  
ORCID: 0000-0003-2955-8254

**Đorđe Stakić**

Ekonomski fakultet, Univerzitet u Beogradu, Srbija  
djordje.stakic@ekof.bg.ac.rs  
ORCID: 0000-0002-3241-4289

**Danilo Đurđić**

Ekonomski fakultet, Univerzitet u Beogradu, Srbija  
danilo.djurdjic@ekof.bg.ac.rs  
ORCID: 0009-0001-3438-8846

**Apstrakt:** Poslovni sistemi koji podržavaju aplikacije koje zahtevaju rad sa velikim brojem podataka nalaze se u situaciji da moraju doneti odluku o tome kako podatke najefikasnije skladištiti i raditi sa njima. Jedna od najvažnijih odluka je određivanje najbolje platforme koja će se koristiti za skladištenje i rad sa podacima u okviru aplikacije. Nekada se većina poslovnih sistema odlučivala za korišćenje SQL baze podataka zbog njihove sposobnosti da zaštite podatke i obezbede njihov integritet. Ubrzan rast korišćenja interneta i cloud tehnologija doveli su do toga da su se mnogi poslovni sistemi okrenuli NoSQL bazama podataka, velikim delom zato što mogu bolje da rukuju velikim brojem nestrukturiranih i polustrukturiranih podataka. Obe vrste baza podataka imaju svoje prednosti i nedostatke, ali se razlikuju po tome kako se prave, kako se preuzimaju i skladište podaci i kako im aplikacije pristupaju. Razumevanjem ovih razlika poslovna organizacija može doneti efikasniju odluku o tome koja će vrsta baza podataka najviše odgovarati njihovom poslovanju. U ovom radu će se razmotriti ključne razlike ovih vrsta baza podataka ali i njihov značaj u izradi poslovnih aplikacija.

**Ključne riječi:** baze podataka, SQL, NoSQL, poslovne aplikacije

**JEL klasifikacija:** C80, M15

**Abstract:** Business systems supporting applications that require working with large amounts of data find themselves in a situation where they have to make decisions about how to store and work with data more efficiently. One of the most important decisions is determining the best platform which will be used for store and work with data within the application. In the past, most business systems have opted to use SQL databases because of their ability to protect data and ensure its integrity. The rapid growth of Internet use and cloud technologies have led many business systems turn to great extent to NoSQL databases because they can better handle large amounts of unstructured and semi-structured data. Both types of databases have their advantages and disadvantages, but differ by how they are created, how data is retrieved and stored, and how applications access it. By understanding these differences, a business organization can make a more effective decision about which type of database will best suit their business. This paper will consider the key differences between these types of databases, as well as their importance in the development of business applications.

**Key Words:** databases, SQL, NoSQL, business applications

**JEL classification:** C80, M15

## 1. UVOD

Svaki poslovni sistem, firma, radi sa većom ili manjom količinom podataka. Na menadžmentu te firme je da proceni kako će najkvalitetnije tim podacima da se upravlja. Od velike pomoći i koristi su standardizovani sistemi za upravljanje bazama podataka. Ovi sistemi se grubo mogu podeliti na relacione i nerelacione i u nastavku će o njima biti više reči.

U radu (Nance, Lossner, Iype, & Harmon, 2013) dato je poređenje ove dve vrste sistema, kao i ukazivanje na to u kojim situacijama je koji sistem bolji za korišćenje. U radu (Venkatraman, Fahd, Kaspi, & Venkatraman, 2016) autori analiziraju vezu oba ova sistema iz ugla analitike velikih podataka. U radu (Hammes, Medero, & Mitchell, 2014) porede se ova dva sistema na cloud platformama. Poređenje NoSQL i SQL baza podataka dato je u radovima (Sareen, & Kumar, 2015) i (Patel, & Eltaieb, 2015), naročito iz ugla skladištenja podataka i rada sa velikim podacima. O migraciji sa SQL na hibridni SQL/NoSQL model podataka može se više pronaći u radu (Sokolova, Gómez, & Borisoglebskaya, 2020).

## 2. RELACIONE BAZE PODATAKA

Relaciona baza podataka predstavlja kolekciju strukturiranih podataka koji se čuvaju u tabelama. Svaka tabela se sastoji iz strogo definisanih kolona i redova. Zbog svoje jasno definisane strukture, nad ovakvim bazama je moguće vršiti veoma kompleksne operacije uz spajanje više tabela. Osnovni gradivni blok svake relacione baze je

tabela. Iz tog razloga je potrebno vrlo pažljivo vršiti organizaciju samih podataka i izrazito voditi računa o strukturi baze. U kolonama su definisane osobine, dok se u redovima nalaze instance tih podataka, i takav oblik je potrebno zadržati bez obzira na upotrebu. Za komunikaciju sa bazom koristi se strukturni upitni jezik (Structured Query Language). SQL je programski jezik koji se upotrebljava za postavljanje upita nad podacima. U zavisnosti od potreba korisnika, ti upiti mogu biti jednostavni, kao što CRUD operacije (Create, Read, Update, Delete), ili kompleksniji koji podrazumevaju upotrebu podataka iz više tabela.

Poslednjih decenija, mnoge kompanije razvile su svoje okruženje za upravljanje relacionim bazama podataka. Jedno od najčešće korišćenih okruženja je MySQL, koji je izvorno nastao 1995. godine. Prvenstveno se istakao zbog svoje stabilnosti, jednostavnosti i brojnih funkcija. Još jedan primer upravljačkog sistema baze podataka je PostgreSQL koji je poznat po svojoj pouzdanosti, visokom kapacitetu i podršci za upotrebu SQL visokih performansi. Oracle Corporation razvila je Oracle Database kao jedan od najviše upotrebljivanih alata za upravljanje SQL bazama podataka. Koriste ga brojna preduzeća, ali i vladine i nevladine organizacije širom sveta. Jedna od najvećih svetskih kompanija Microsoft razvila je Microsoft SQL Server koji ima bogate alate i funkcije za razvoj i upravljanje bazama podataka koji se široko koriste u Windows okruženju. Svako od ovih okruženja je prepoznatljivo po svojoj pouzdanosti, delotvornosti i podršci za bezbednost poslovanja i samih podataka, stoga je važno pažljivo razmotriti potrebe projekta i odabrati najprikladnije rešenje u zavisnosti od potreba korisnika.

Relacioni model se primenjuje decenijama i ima ključnu ulogu u poslovnim procesima mnogih preduzeća. MySQL je jedan primer od mnogih licenciranih relacionih baza podataka otvorenog koda koje su dokazano pouzdane za primenu kod poslovnih aplikacija. Pošto su svoje mesto pronašli u razvoju velikog broja aplikacija, navedena okruženja su postala sinonim za pouzdanost i visoke performanse u svetu baza podataka. Sa svojim jasno definisanim šemama i moćnim SQL upitima, relacione baze podataka pružaju stabilno i konzistentno okruženje za skladištenje i upravljanje podacima.

Kako bi aplikacija uspešno funkcionisala, potrebno je voditi računa o strukturi baze. Glavni cilj prilikom projektovanja baze podataka je smanjenje frekvencije ponavljanja podataka. Konačno, ukoliko se ispoštuju sva pravila, moguća je laka i efikasna manipulacija podacima. Organizacija relacionih baza je u skladu sa ACID svojstvima (Atomicity, Consistency, Isolation, Durability). Ova pravila omogućavaju doslednost prilikom rada sa podacima, što obezbeđuje efikasno izvršavanje upita i konzistentnost podataka. ACID pravila je potrebno strogo poštovati kako bi se očuvao integritet podataka.

Normalizacija podataka je ključan uslov kako bi se obezbedila nesmetana funkcionalnost. To je proces organizovanja podataka koji ima za cilj efikasnost, očuvanje integriteta i minimiziranje redundantnosti podataka. Kako bi obezbedile integritet, SQL baze podataka oslonjene su na matematičku teoriju funkcionalne zavisnosti.

Prvi korak koji bi trebalo razmotriti prilikom dizajniranja relacionih baza je plan normalizacije podataka. Ova pravila, bazirana na matematičkim principima, definišu veze između atributa u tabelama. Praćenjem pravila normalizacije, moguće je smanjiti pojave redundantnosti i anomalija u podacima. To za rezultat ima bolju strukturu i obezbeđuje efikasnije upite. Razvijen je set pravila koji se koriste u kreiranju relacionih baza podataka kako bi se postigao visok stepen kvaliteta i efikasnosti.

Relacione baze podataka zasnovane su na teoriji skupova i algebri. Definisane same strukture je moguće pomoću matematičke teorije skupova. Osnovni upiti se mogu posmatrati kroz matematičke principe unije, preseka i razlike skupova. Operacije filtriranja (select), projekcije, spajanja (join), unije i preseka se mogu jednako primeniti na matematičkim skupovima, kao i u bazi podataka. Konačno, kako bi se definisale operacije nad relacijama, koristi se relaciona algebra.

Najveći problem kod relacionih baza podataka jesu troškovi vertikalnog skaliranja. Tradicionalno, relacioni model je bio zasnovan na jednom serveru. To podrazumeva nabavku hardvera u slučaju potrebe za proširenjem baze, što se ispostavilo kao izuzetno skupa i nefleksibilna opcija. Tržište, kao i dostupna količina potrebnih informacija ubrzano rastu, mnoge relacione baze su dodale mogućnost horizontalne raspodele pomoću particionisanja, i time omogućile rad sa velikom količinom podataka. Podaci se pomoću horizontalnog skaliranja smeštaju na nekoliko servera, što smanjuje rizik gubitka i oštećenja podataka. Na taj način je smanjeno opterećenje i rizik, jer umesto jednog servera, sada baza može da bude smeštena na nekoliko međusobno nezavisnih lokacija. Horizontalnim skaliranjem je omogućen rad sa velikim skupovima podataka (Big Data). Ipak, praksa je pokazala da su NoSQL baze prikladnije rešenje za rad sa BigData. Fokus većine SQL baza podataka ostaje doslednost i dostupnost u odnosu na toleranciju particija.

### **3. NERELACIONE BAZE**

NoSQL (Not Only SQL) baze podataka su razvijene kao alternativa relacionim bazama podataka. Glavni fokus NoSQL baza jeste rad sa podacima koji nisu mogli efikasno da se čuvaju i analiziraju u SQL bazama. NoSQL baze obrađuju velike količine posluksstrukturiranih, nestruksukturiranih podataka, ali i podatake koji se vrlo često menjaju (dinamične). NoSQL baze su dosta fleksibilnije po pitanju postojanja striktno strukture i ne zahtevaju postojanje tabela, samim tim su jednostavnije za kreiranje i pogodnije za čuvanje velike količine podataka. Pored prilagodljive strukture i nepostojanja ograničenja po pitanju oblika skladištenja podataka, ipak je važno voditi računa o konzistentnosti same strukture.

Horizontalno strukturiranje je jedno od najvećih prednosti nerelacionih baza. Najviše se ističe sposobnost analize velikih količina podataka. Kako bi se to postiglo, sami NoSQL upiti su prilagođeniji za rad sa velikim setovima podataka nego što je slučaj kod SQL jezika. Ovakve performanse su moguće zato što nerelacione baze prihvataju određene ustupke u pogledu doslednosti i referencijalnog integriteta podataka.

Nerelacione baze su, poput relacionih, efikasne prilikom čuvanja strukturiranih podataka u formatima kao što su XML i JSON. Ipak najčešća upotreba NoSQL baza je čuvanje polustrukturiranih i nestrukturiranih podataka zbog nepostojanja striktnih pravila normalizacije koji bi je ograničili u tome. Međutim, ta sloboda donosi određene probleme. Nerelacione baze su veoma limitirane po pitanju kompleksnih upita zbog česte nemogućnosti spajanja više tabela. Iz tog razloga, glavna prednost jeste u čuvanju podataka u tekstualnom formatu, kao i multimedijalnih sadržaja. Velika prednost NoSQL baza u odnosu na SQL jeste u čuvanju dinamičnih podataka. Najbolji primer za to su audio zapisi, kao i logovi dobijeni iz bezbedonostnih, ali i ostalih IoT (Internet of Things) uređaja. Takvi podaci su veoma promenljivi u svojoj strukturi i veličini. Zato nerelacione baze predstavljaju idealno rešenje jer nije potrebno posebno planiranje kako uklopiti podatke u strukturu, kao što je to slučaj sa relacionim bazama.

Postoji nekoliko vrsta nerelacionih baza podataka. Izdvajaju se dokument orijentisane, ključ-vrednost orijentisane, graf orijentisane i kolona orijentisane baze podataka. Izbor tipa baze će zavisi od strukture i tipa podataka koji se čuvaju, kao i od potrebe korisnika. Dokument orijentisane baze podataka skladište podatke u formatu JSON, XML ili BSON. Ključ-vrednost orijentisane baze podataka čuvaju podatke kao parove ključ-vrednost dok su kolona orijentisane baze podataka izrazito efikasne za analitiku i rad sa velikim skupovima podataka. Graf orijentisane baze podataka omogućavaju skladištenje i upravljanje podacima kod kojih postoji izražena međusobna relacija. Izbor vrste NoSQL baze podataka zavisi od prirode i specifičnih potreba aplikacije.

U poslednjih nekoliko decenija mnoge kompanije su razvile sopstvene NoSQL baze. MongoDB je jedna od najpopularnijih nerelacionih baza podataka, njen fokus je na skladištenju podataka u JSON formatu, što znači da je dokument orijentisana baza podataka. MongoDB je poznat je po svojoj skalabilnosti i lakoći upravljanja podacima. Cassandra je još jedna popularna NoSQL baza podataka, koja je kolonski orijentisana. Dizajnirana je specijalno za obradu velikih i složenih setova podataka. Glavna primena Cassandre je unutar sistema i aplikacija koje zahtevaju veliku brzinu upisivanja i čitanja podataka. Redis je istaknuta ključ-orijentisana baza podataka koja se koristi za keširanje podataka u memoriji. Iz tog razloga je odličan alat za brzo i efikasno za učitavanje i pisanje podataka koji se koriste i analiziraju u realnom vremenu, kao i kod mašina sa visokim performansama. Neo4j je graf orijentisana baza koja je koncentrisana na unos i upravljanje podacima koji se mogu predstaviti grafovima.

Velika prednost nerelacionog modela baza podataka su troškovi implementacije i održavanja. Nerelacione baze mogu biti ekonomičnija opcija u odnosu na tradicionalne baze podataka iz nekoliko razloga. Prvo, skalabilnost je ključna prednost jer je omogućeno horizontalno širenje. Dodavanje jeftinih servera, kao i mogućnost upotrebe cloud tehnologije, značajno utiče na smanjenje ukupnih troškova. Plaćanje samo za stvarnu upotrebu i skaliranje resursa prema potrebi može doneti znatnu uštedu, pogotovo za nova preduzeća koja još uvek nemaju jasnu predstavu o ukupnom obimu podataka. Drugo, izbegava se problem vertikalnog skaliranja jednog servera. Za razliku od relacionih baza, nije potrebno ulagati veliki novac u nabavku hardvera,

a omogućava se efikasna obrada velikih količina podataka. Treće, moguće je izbeći troškove licenciranja. Neke od najpoznatijih NoSQL baza podataka su otvorenog koda i potpuno su besplatne. To znači da je moguće u potpunosti izbeći troškove licenciranja. Zato je nerelacioni model baza podataka privlačan izbor za mnoge organizacije koje žele da smanje troškove i efikasno upravljaju podacima. Nerelacione baze su, kao rezultat svojih osobina, našle primenu u savremenim aplikacijama koje zahtevaju pristup u realnom vremenu velikim setovima podataka. Veoma su pogodne za aplikacije elektronske trgovine i društvenih mreža, za analizu velikih skupova podataka i IoT (Internet of Things) aplikacije. Ipak, ne može se reći da su nerelacione baze bolje rešenje od SQL baza za razvoj aplikacija. Bolji zaključak bi bio da su to komplementarna rešenja, koja međusobno nadopunjuju svoje nedostatke. Iz tog razloga je potrebno razmotriti koje su tačno potrebe organizacije pre odabira tipa baze podataka.

#### **4. IZAZOVI PRIMENE NOSQL MODELA PODATAKA**

Pre implementacije NoSQL baza podataka potrebno je i sagledati mane ovog modela. Nedostaci NoSQL sistema se odnose na određene karakteristike kao i na konkretne probleme koji nastaju prilikom upotrebe. Fleksibilnost strukture podataka je jedna od glavnih prednosti ovog tipa baze podataka ali takođe, dovodi do velikih problema prilikom modeliranja nerelacione baze. Usled nerigoroznog pristupa, potpuna odgovornost osmišljavanja i održavanja konzistentosti je na programeru. Problemi nastaju i zbog toga što nerelacione baze ne moraju da poštuju ACID (Atomicity, Consistency, Isolation, Durability) pravila prilikom definisanja modela podataka. Nedostatak konzistencije stvara dodatne poteškoće jer specifična struktura zahteva potpuno nova rešenja. Iz tih razloga su često neophodni dodatni alati i uputstva kako bi se moglo efikasno upravljati bazom podataka.

Potrebno je razmotriti i pitanje upravljanja podacima u NoSQL bazama. Zbog svoje nekonzistentnosti nastaje niz poteškoća prilikom pisanja upita. Kod relacionih modela spajanje različitih tabela relacionih baza je relativno jednostavno. Jasna struktura i konzistentnost tipova podataka omogućava jednostavno snalaženje prilikom postavljanja kompleksnih upita. S druge strane, korisnici nerelacionih baza mogu naići na problem kod izvlačenja složenih zaključaka iz postojećih podataka. Često je potrebno uložiti veliki trud kako bi se postigao isti rezultat kao kod primene SQL jezika, što može biti vremenski i resursno zahtevno.

NoSQL baze su relativno nove, tako da baza korisnika i dalje nije velika kao kod relacionog modela. Ovo je jedan od razloga zašto ne postoji razvijena podrška prilikom upotrebe NoSQL baza. S obzirom da relacione baze postoje godinama, razvijeni su alati i resursi za rešavanje specifičnih problema. Sa druge strane, kod NoSQL baza neretko dolazi do problema za koje još uvek nije pronađeno rešenje i pored postojeće dokumentacije. Različite tehnologije i veliki broj vendora utiče na to da postoji veliki broj potpuno različitih pristupa u razvoju baze što utiče na rešavanje nastalih problema. Takođe, kako su NoSQL baze još uvek novina na tržištu, postoje različiti vendori koji imaju sopstvene patentirane tehnologije što dodatno otežava organizacijama da pređu sa jednog vendedora na drugi. Ovo se naziva "vendor lock-in" ili vezivanje za dobavljača. Još jedan izazov može biti migracija sa relacionih na

nerelacione modele, zbog međusobno potpuno različitih struktura i načina modeliranja. Mogući problem može biti potreba za potpunom konverzijom i preoblikovanjem podataka koja podrazumeva detaljno poznavanje obe vrste sistema. NoSQL baze se koriste za obradu velike količine podataka u realnom vremenu što znači da je neophodno obezbediti horizontalno skaliranje. Skaliranje NoSQL baza je kompleksan proces i na duži rok doprinosi smanjenju troškova, ali je neophodno veoma detaljno planiranje kako bi se taj proces obavio ispravno. Moguća rešenja horizontalnog skaliranja su segmentacija i partitionisanje, mada je neophodno razmotriti uticaj ovih odluka na performanse upita i održavanje na konzistentnosti podataka pre implementacije. Kod NoSQL baza takođe postoji mnogo veća opasnost pri zaštiti podataka nego kod tradicionalnih baza. Jedan od glavnih razloga za to je nedovoljna podrška prilikom autentifikacije i enkripcije podataka, manjak standardizacije i problemi sa skaliranjem što doprinosi povećanju ranjivosti na napade, teže održavanje bezbednosti i manju konzistentnost podataka. Ipak konstantna poboljšanja sigurnosnih aspekata i specijalizovane tehnologije doprinose rešavanju ovih problema. Kao što je rečeno, NoSQL baze su relativno nova tehnologija koja se i dalje razvija i unapređuje, tako da se smatra da će uticaj navedenih problema u budućnosti biti minimiziran. Kompanije koje pružaju usluge nerelacionih baza podataka, konstantno rade na unapređivanju bezbednosti putem enkripcije komunikacije, autentifikacije korisnika i zaštite podataka. Uz prave alate i adekvatnu implementaciju moguće je garantovati potpunu sigurnost ovih baza podataka.

## **5. PRIMENA SQL I NOSQL BAZA PODATAKA U KOMPANIJAMA**

Baze podataka imaju veliku ulogu u poslovanju svih velikih kompanija. Smatra se da preduzeća koja ne koriste dostupne informacije o kupcima, tržištu i ekonomskim prilikama, neće moći da prežive moderne tržišne uslove. Iz tog razloga, mnoge svetske kompanije se odlučuju za neke od dostupnih relacionih i nerelacionih sistema baza podataka. Izbor će svakako zavisiti od individualnih potreba poslovanja i ne postoji konkretan odgovor na pitanje koji tip baza podataka je bolji ili lošiji. Neka preduzeća zapravo koriste i relacione i nerelacione baze u različitim sektorima poslovanja.

SQL je široko rasprostranjen jezik i kao takav je 2021. godine proglašen za najviše upotrebljavan jezik za upravljanje podacima. Takođe, prema istom istraživanju, SQL se nalazi na četvrtom mestu najpoznatijih programskih jezika (<https://insights.stackoverflow.com/survey/2021#most-popular-technologies-database-prof>). Tome doprinosi i činjenica da mnoge globalne organizacije koriste SQL za sopstvene potrebe čuvanja, obrade i analize podataka. Ovaj jezik se, pored skladištenja i primene u analizi podataka, koristi u svim segmentima poslovanja. Vrlo je pogodan za praćenje i analizu tržišnih uslova, kao i za sastavljanje finansijskih izveštaja. Pošto su takvi podaci najčešće strukturirani, relacione baze imaju veoma dobre alate koji mogu pomoći prilikom donošenja poslovnih odluka.

Mnoge svetske kompanije koriste SQL u svrhe analize podataka. Na primer, Accenture je svetska kompanija koja pruža IT i konsultantske usluge. Ova organizacija koristi SQL za pomoć klijentima u domenu razumevanja i testiranja Oracle sistema. Poznata internacionalna kompanija, koja pruža usluge digitalnog marketinga Adobe Systems aktivno koristi SQL prilikom izgradnje i unapređenja svojih veb aplikacija. Cigna je svetska organizacija koja je specijalizovana u oblasti zdravstvenog osiguranja pojedincima i preduzećima. Veliki deo poslovanja te kompanije jeste sistem upravljanja bazom podataka, kako bi uspešno vodili evidenciju svojih klijenata. Facebook jeste jedna od najpopularnijih društvenih mreža. Kompanija koristi MySQL za upravljanje dela korisničkih podataka. Aplikacija je kreirana u PHP programskom jeziku uz podršku MySQL baze. Microsoft, internacionalna korporacija koja pruža softverske usluge, razvila je sopstveni server za čuvanje podataka koji koristi SQL jezik. Microsoft SQL koriste Windows, Linux i Docker sistemi. Seagate, svetska kompanija specijalizovana u proizvodnji hardvera, koristi SQL kako bi postigla izveštavanje u realnom vremenu. Konačno, i SAP HANA predstavlja relacionu bazu podataka koja u potpunosti podržava upotrebu SQL. Ipak, relacione baze podataka nisu jedinstveno rešenje za svaku firmu. NoSQL je bolje rešenje ukoliko je fokus kompanije na obradi velike količine podataka i analiza u realnom vremenu. Mogu biti bolji izbor kod preduzeća koja brzo rastu zbog svoje prilagodljive strukture. Kompanija LinkedIn koristi Espresso, NoSQL bazu podataka koja pruža mogućnost fleksibilnije strukture čuvanja podataka. Google je razvio sopstvenu nerelacionu bazu podataka pod nazivom Google Cloud Datastore i upotrebljava se za skladištenje podataka sa različitih servisa kao što su Gmail, GooglePlay i Google drive. Netflix koristi Apache Cassandra i DynamoDB nerelacione baze podataka, kako bi čuvali podatke o korisnicima, preporukama i sadržaju. Takođe, Amazon je razvio svoju NoSQL bazu DynamoDB koja služi kao podrška unutar Amazon Web Services. Twitter kao nerelacionu bazu podataka koristi ApacheHbase za skladištenje tvitova, profila i interakcija između korisnika. Bitno je istaći da Facebook sve više migrira na nerelacione baze podataka. Oni koriste bazu podataka koju je razvio Amazon kako bi čuvali podatke o objavama i interakcijama između korisnika.

## ZAKLJUČAK

Poslovni sistemi rade sa sve više podataka koje je potrebno na kvalitetan način pretraživati, povezivati i preuzimati iz njih potrebne informacije. Relacione baze podataka su standardan model koji može da manipuliše podacima poštujući visoke standarde relacione algebre i koristeći SQL jezik za rad sa bazom podataka uključujući i njeno modeliranje, strukturiranje, ažuriranje, kao i kreiranje upita. Međutim u novije vreme kako se pojavljuju velike količine podataka koje je potrebno obraditi, standardne relacione baze podataka predstavljaju kruto ograničenje za rad sa podacima. Iz navedenih razloga u velikim poslovnim sistemima sve se više koristi NoSQL model baza podataka, koji sa jedne strane ublažava ograničenja koja su prisutna kod relacionih baza, a sa druge strane pomaže da se što jednostavnije manipuliše većim količinama podataka.



**LITERATURA**

- [1] Amazon: <https://aws.amazon.com/dynamodb/>
- [2] Channel futures: <https://www.channelfutures.com/cloud-2/the-limitations-of-nosql-database-storage-why-nosqls-not-perfect>
- [3] Codd, E. F. (1983). A relational model of data for large shared data banks, Communications of the ACM
- [4] Developer survey: <https://insights.stackoverflow.com/survey/2021#most-popular-technologies-database-prof>
- [5] Google cloud database: <https://bluexp.netapp.com/blog/gcp-cvo-blg-google-cloud-nosql-firestore-datastore-and-bigtable>
- [6] Hammes, D., Medero, H., & Mitchell, H. (2014). Comparison of NoSQL and SQL Databases in the Cloud.
- [7] Java point: <https://www.javatpoint.com/dbms-functional-dependency>
- [8] Nance, C., Losser, T., Iype, R., & Harmon, G. (2013). Nosql vs rdbms-why there is room for both.
- [9] Netflix: <https://netflixtechblog.com/nosql-at-netflix-e937b660b4c?gi=563445acfba4>
- [10] Patel, T., & Eltaieb, T. (2015). Relational database vs NoSQL. Journal of Multidisciplinary Engineering Science and Technology (JMEST), 2(4), 691-695.
- [11] Sareen, P., & Kumar, P. (2015). NoSQL Database and its comparison with SQL Database. International Journal of Computer Science & Communication Networks, 5(5), 293-298.
- [12] Sokolova, M. V., Gómez, F. J., & Borisoglebskaya, L. N. (2020). Migration from an SQL to a hybrid SQL/NoSQL data model. Journal of Management Analytics, 7(1), 1-11.
- [13] Tutorials point: <https://www.tutorialspoint.com/challenges-of-nosql>
- [14] Venkatraman, S., Fahd, K., Kaspi, S., & Venkatraman, R. (2016). SQL versus NoSQL movement with big data analytics. International Journal of Information Technology and Computer Science, 8(12), 59-66.

**SUMMARY**

*The pace of data growth has accelerated due to the rapid development of the digital world, which has resulted in an increase in the amount and diversity of data. The amount of data gathered from various sources has typically grown to a point where it cannot be effectively used or managed by standard data management techniques, such as conventional search engines or RDBMS, or relational database management systems. In order to address upcoming difficulties and track the development of new effective methodologies for data engineering, the rapid expansion of unstructured data necessitates a full paradigm change for the new era. From the above description of database models and their various applications, the challenges of applying the NoSQL model, as a model that works with all types of databases, are clarified. All of the above is a way that will approximate the application of these models in various business systems.*



This work is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License