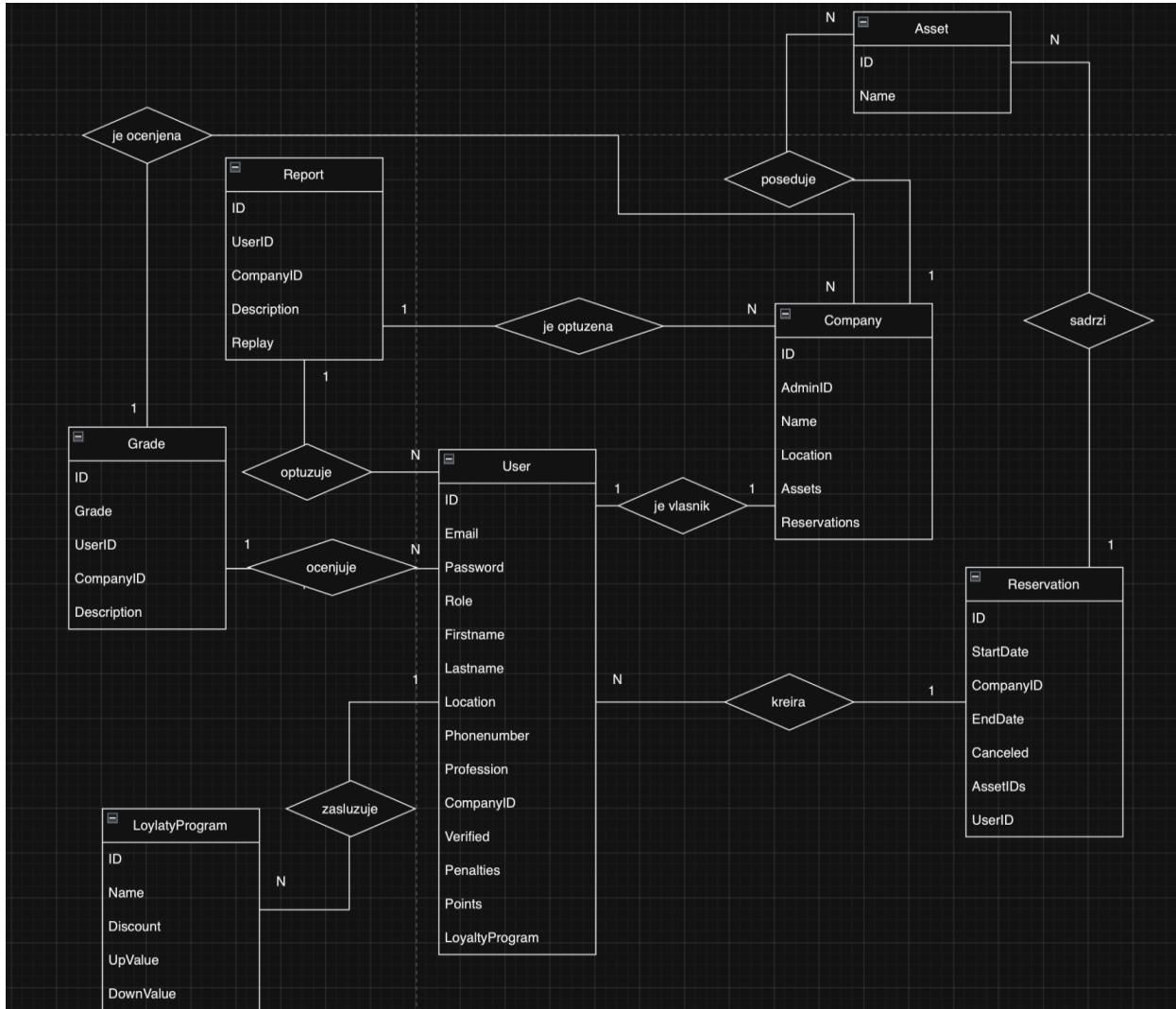


# MEDcom

Autor: Nikola Kojic

## 1. Dizajn Šeme Baze Podataka



## 2. Strategija za Partitionisanje Podataka

Vertikalno Partitionisanje

- **Rezervacije:** Kako imam osnovnu tabelu Reservations sa ključnim informacijama kao što su ReservationID, UserID, AssetID, i drugu tabelu ReservationDetails sa dodatnim informacijama kao što su StartDate, EndDate, Status.
- **Zakazani Termini:** Slično, mogu razdvojiti tabelu Appointments u osnovne podatke i detalje o terminima.

### Poboljšanje Performansi

**Brža Pretraga i Upiti:** Kada se podaci particionišu po vremenskim periodima, upiti se mogu ograničiti na relevantne particije, čime se smanjuje broj redova koje treba pregledati. Na primer, upit za rezervacije u januaru ne mora da pretražuje podatke iz svih meseci, već samo iz particije za januar. **Brža Pretraga i Upiti:** Kada se podaci particionišu po vremenskim periodima, upiti se mogu ograničiti na relevantne particije, čime se smanjuje broj redova koje treba pregledati. Na primer, upit za rezervacije u januaru ne mora da pretražuje podatke iz svih meseci, već samo iz particije za januar.

## 3. Strategija za Replikaciju i Otpornost na Greške

### MongoDB Replikacioni Setovi

- **Replikacioni set** u MongoDB je grupa MongoDB servera (čvorova) koja se koristi za obezbeđivanje visoke dostupnosti i otpornosti na greške. Svaki replikacioni set se sastoji od jednog primarnog čvora i jednog ili više sekundarnih čvorova.
- **Primarni Čvor:** Svaki zapis (ili “operacija”) se prvo upisuje u primarni čvor. Primarni čvor je jedini koji prihvata pisanje podataka.
- **Sekundarni Čvorovi:** Sekundarni čvorovi redovno replikovani podatke sa primarnog čvora. Oni mogu preuzeti ulogu primarnog čvora ako trenutni primarni čvor otkaže.

### Backup i Oporavak

- **Redovni Backup:** Implementirajte strategiju redovnih sigurnosnih kopija baze podataka kako biste obezbedili da podaci mogu biti obnovljeni u slučaju ozbiljnih grešaka ili gubitka podataka.
- **Testiranje Oporavka:** Redovno testirajte procedure za oporavak kako biste osigurali da su sigurnosne kopije ispravne i da se podaci mogu brzo i tačno obnoviti.

## 4. Strategija za Keširanje Podataka

### Keširanje Upita i Statickih Podataka

- **Upiti:** Keširajte rezultate složenih ili često korišćenih upita kako biste smanjili opterećenje baze podataka. Na primer, rezultate pretrage ili često korišćene statističke podatke.
- **Staticki Podaci:** Keširajte statičke podatke koji se retko menjaju, kao što su korisničke konfiguracije ili postavke aplikacije.

### Smanjenje Opterećenja Baze Podataka

- **Smanjenje Broja Upita:** Keširanjem često korišćenih podataka smanjuje se broj upita koji se šalju glavnoj bazi podataka, što smanjuje opterećenje na bazi i poboljšava njene performanse.
- **Brži Odgovori:** Podaci koji su pohranjeni u kešu mogu se dohvatiti znatno brže nego iz glavne baze podataka, čime se smanjuje latencija i poboljšava korisničko iskustvo.

## Poboljšanje Performansi Aplikacije

- **Brži Pristup Podacima:** Keširanje omogućava brži pristup podacima, što poboljšava performanse aplikacije i korisničko iskustvo.
- **Manje Opterećenje na Serverima:** Keširanjem se smanjuje potreba za čestim pristupom glavnoj bazi podataka, što može smanjiti troškove i opterećenje na serverima.

## 5. Procena Hardverskih Resursa

- **Ukupan broj korisnika:** 100 miliona
- **Broj rezervacija mesečno:** 500.000
- **Sistem mora biti skalabilan i visoko dostupan**

**Trenutna veličina baze podataka za 1 godinu:** Podaci po korisniku: 0,2 KB (korisnički podaci) i Podaci po rezervaciji: 0,1 KB (podaci o rezervaciji)

Prvi proračuni:

- **Korisnički podaci:**
- 100 miliona korisnika \* 0,2 KB = 20 GB za korisničke podatke u prvoj godini
- **Rezervacije:**
- 500.000 rezervacija mesečno \* 12 meseci \* 0,1 KB = 0,6 GB za rezervacije u prvoj godini

**Ukupno za prvu godinu:**

- 20 GB (korisnici) + 0,6 GB (rezervacije) = **20,6 GB**

### 2. Projekcija rasta podataka za 5 godina:

Pretpostavljeni godišnji rast od 10% za korisničke podatke, dok broj rezervacija ostaje stabilan.

- **Korisnici:**
- Godina 1: 20 GB
- Godina 2: 22 GB
- Godina 3: 24,2 GB
- Godina 4: 26,62 GB
- Godina 5: 29,28 GB

Ukupno korisnički podaci kroz 5 godina:

**20 GB + 22 GB + 24,2 GB + 26,62 GB + 29,28 GB = 122,1 GB**

- **Rezervacije:**
- 0,6 GB godišnje \* 5 godina = **3 GB** za rezervacije kroz 5 godina

### Ukupna veličina podataka za 5 godina:

- 122,1 GB (korisnici) + 3 GB (rezervacije) = **125,1 GB**

### Overhead (indeksi, replikacija, backup):

- **Indeksi:** Pretpostavimo dodatnih 30% za indekse:

$125,1 \text{ GB} * 0,30 = \mathbf{37,53 \text{ GB}}$  za indekse

- **Replikacija:** Koristeći MongoDB sa primarnom i sekundarnom replikom:

$(125,1 \text{ GB} + 37,53 \text{ GB}) * 2 = \mathbf{325,26 \text{ GB}}$  za replikaciju

- **Backup prostor:** Pretpostavimo dodatnih 10% za backup:

$125,1 \text{ GB} * 0,10 = \mathbf{12,51 \text{ GB}}$  za backup

### Ukupni potrebni hardverski resursi:

- **Osnovni podaci + indeksi:**  $125,1 \text{ GB} + 37,53 \text{ GB} = \mathbf{162,63 \text{ GB}}$
- **Replikacija:** 325,26 GB
- **Backup:** 12,51 GB

### Ukupan prostor:

$325,26 \text{ GB (za replikaciju)} + 12,51 \text{ GB (za backup)} = \mathbf{337,77 \text{ GB}}$  za narednih 5 godina.

Zaključak: Za sledećih 5 godina, potrebni prostor za skladištenje podataka sa ovim novim vrednostima iznosi približno **337,77 GB**.

## 6. Strategija za Load Balancer

### Između aplikativnih servera i baze podataka

Kako sistem koristi više replika baza podataka (primarna i sekundarna, kao što je predloženo u replikacionoj strategiji), load balancer može da rasporedi upite ka bazi na različite instance baze podataka. Na primer, read-only upiti mogu da idu na sekundarne instance kako bi se smanjilo opterećenje na primarnu bazu.

- **Primer primene:** Upiti koji prikupljaju statistike ili učitavaju podatke za prikaz rezervacija mogu ići na repliku baze, dok upiti za kreiranje novih rezervacija idu na primarnu bazu. Load balancer ovde optimizuje raspodelu upita.

## 7. Praćenje Operacija Korisnika

### 1. Prijavljivanje korisnika (Login)

- **Zašto pratiti?:** Problemi sa prijavljivanjem mogu direktno uticati na korisničko iskustvo i percepciju pouzdanosti sistema. Praćenjem vremena potrebnog za prijavljivanje i učestalost neuspešnih pokušaja, možeš identifikovati probleme sa autentifikacijom ili sistemске greške.
- **Šta pratiti?**
- Učestalost neuspešnih prijava.

- Vreme potrebno za uspešno prijavljivanje.
- Broj aktivnih korisnika u određenom vremenskom periodu.

## 2. Rezervacije

- **Zašto pratiti?:** Ovo je ključna funkcionalnost sistema i problemi sa kreiranjem rezervacija mogu značajno narušiti korisničko iskustvo. Potrebno je pratiti i uspešne i neuspešne zahteve za rezervacije, kao i vreme potrebno za odgovor sistema.
- **Šta pratiti?**
- Broj uspešnih i neuspešnih rezervacija.
- Vreme potrebno za kreiranje rezervacije.
- Korisničke sesije koje često otkazuju ili prekinu proces rezervacije.

## 3. Pregledanje i filtriranje podataka (pretraga dostupnih entiteta, kompanija, i resursa)

- **Zašto pratiti?:** Ako korisnici često pretražuju resurse ili kompanije, ali ne prave rezervacije ili preduzimaju sledeći korak, to može ukazivati na problem sa dostupnošću ili kvalitetom podataka. Možda je potrebno unaprediti način prikazivanja informacija.
- **Šta pratiti?**
- Broj pretraga u odnosu na broj rezervacija.
- Vreme potrebno za prikaz rezultata pretrage.
- Korisničke obrasce u pretrazi (najčešće pretrage, korišćeni filteri).

## 4. Podnošenje pritužbi i ocenjivanje kompanija (Reporting i Grading)

- **Zašto pratiti?:** Visoka stopa pritužbi ili loših ocena može ukazivati na probleme sa određenim kompanijama ili resursima. Praćenjem tih operacija, možeš proaktivno otkriti probleme pre nego što utiču na reputaciju sistema.
- **Šta pratiti?**
- Učestalost pritužbi i ocena po korisniku i po kompaniji.
- Vreme potrebno za rešavanje pritužbi.
- Trendovi u ocenjivanju kompanija.

# 8. Crtež Dizajna Arhitekture

