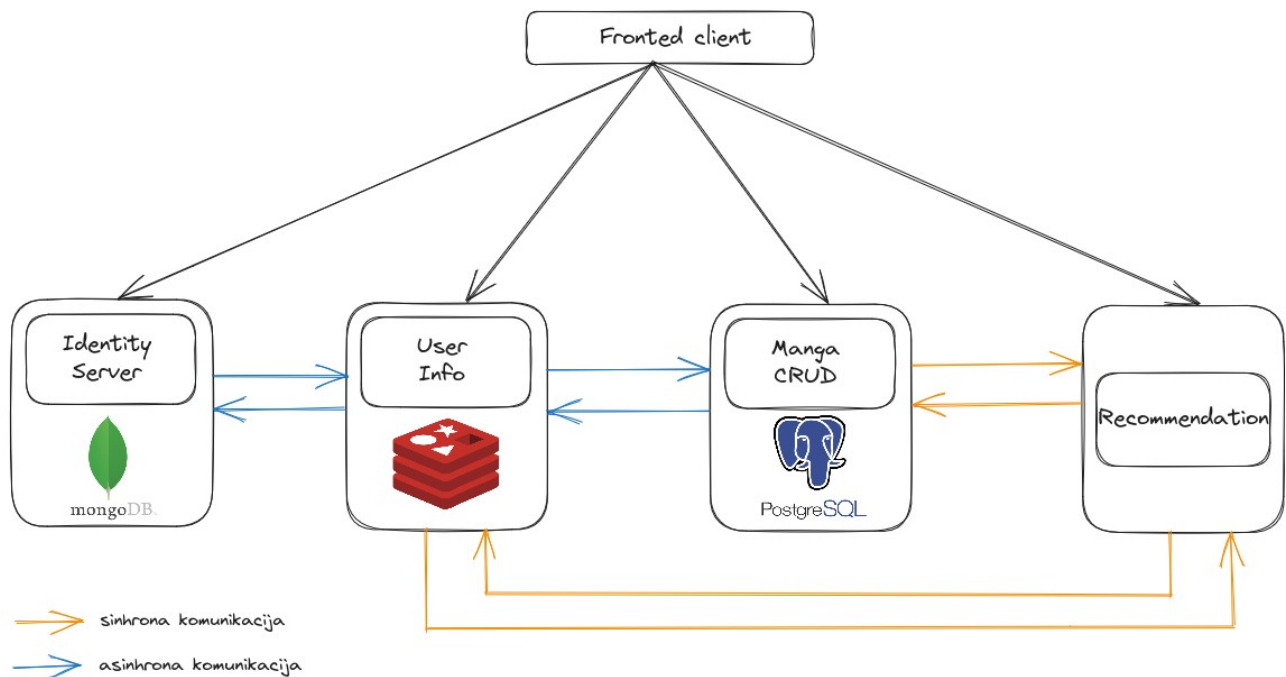


Mikroservisna arhitektura

Konačni izgled arhitekture mikroservisne aplikacije:



Aplikacija se sastoji od 4 mikroservisa, a to su IdentityServer servis, UserInfo servis, Manga CRUD servis i Recommendation servis.

Mikroservis IdentityServer pruža usluge autorizacije i autentikacije korisnika. Autentikacija se zasniva na JWT tokenima (access token, refresh token). Takođe, ovaj servis upravlja korisničkim nalogom, tj. omogućava promenu podataka u korisničkom nalogu. Koristi Mongo bazu podataka.

UserInfo servis je servis koji ima više funkcionalnosti. Kako samo ime kaže, ovaj servis obezbeđuje razne informacije o korisniku. Za konkretnog korisnika ovaj servis čuva informacije o svim mangama koje ikada čitao, koja je njegova lista želja za čitanje (tj. korisnik može da napravi listu koje mange želi sledeće da čita, kao podsetnik), kao i informaciju koja je poslednja manga koju je korisnik čitao. Na osnovu tih informacija Recommendation servis daje preporuke mangi korisniku na osnovu različitih kriterijuma koje korisnik može da unese, na osnovu autora čije je mange čitao, na osnovu žanra koji je već čitao i slično. Koristi se Redis baza podataka. Komunicira sa IdentityServer servisom, kada se kreira novi korisnik, IdentityServis posalje informaciju UserInfo servisu o tom događaju, pa se i u UserInfo servisu kreira novi objekat za

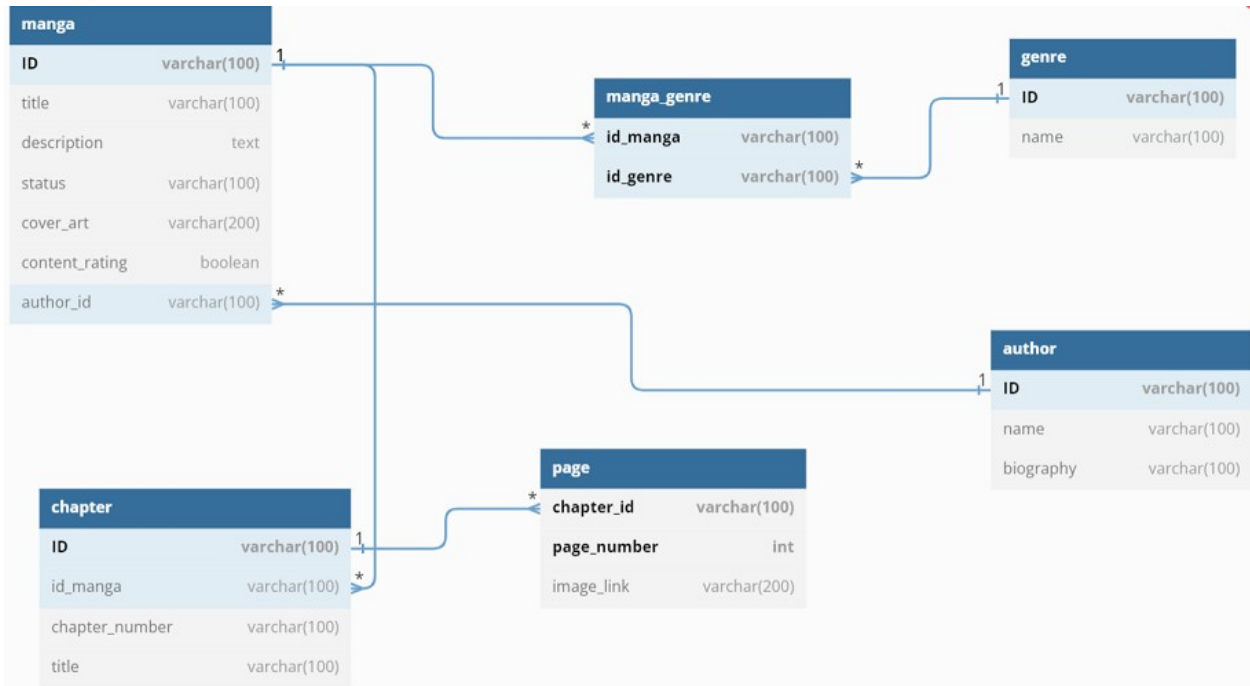
tog korisnika sa praznim svim informacijama koje ovaj servis čuva, koje će se tokom vremena kako korisnik čita razne mange biti dopunjavane. Opisana komunikacija je asinhrona (RabbitMQ). Pored IdentityServer servisa, ovaj servis komunicira i sa Recommendation servisom, kako bi Recommendation servis dobio informacije o korisniku koji traži preporuku za mange. Ova je komunikacija sinhrona(gRPC).

Manga CRUD servis je servis koji skladišti same mange i omogućava manipulaciju njima. Omogućava dodavanje, brisanje, izmenu i čitanje mangi. Koristi PostgreSQL bazu podataka. Komunicira sa UserInfo servisom na dva načina, sinhrono i asinhrono. Asinhrono komunicira sa UserInfo servisom kada mu šalje informacije o radnjama korisnika, u smislu, kada korisnik pročita neku mangu, on asinhrono pošalje UserInfo servisu informaciju o tome da bi UserInfo servis zabeležio to i dodao tu mangu u spisak ikad čitanih mangi tog korisnika i slične akcije. Pored UserInfo servisa, komunicira i sa Recommendation servisom, ta komunikacija je sinhrona. Recommendation servis se obraća Manga servisu kada su mu potrebne konkretne mange koje treba da prikaže kao preporuku.

Recommendation servis, kao što mu ime kaže se bavi davanjem preporuka korisniku koje mange bi mogao da čita sledeće. Te preporuke ovaj servis generiše na osnovu podataka o korisniku, tačnije na osnovu njegove istorije čitanja. Kriterijumi po kojima se daju preporuke mogu biti različiti koje korisnik može da bira. Kao što je pomenuto u opisima prethodnih servisa, ovaj servis komunicira sa IdentityServer servisom na sinhron način i sa Manga CRUD servisom takodje na sinhron način.

Arhitektura baza podataka

Manga database - PostGresql



UserInfo database - Redis



Polje reading_journal se može deserijalizovati u:

```
{  
    string user_id,  
    string last_read_manga_id,  
    List<string> all_read_manga_ids,  
    List<string> wish_list  
}
```

IdentityServer database - Mongo

user	
ID	ObjectId
firstname	text
lastname	text
username	text
email	text
phone_number	text
created_at	Date
modified_at	Date