

# SBNZ

## Librarian

Sistem za preporuku knjiga za čitanje

### Članovi tima:

- Strahinja Sekulić SV34/2020
- Lazar Milanović SV15/2020

### Motivacija

Često nakon završene knjige ne znamo šta sledeće da čitamo. Ponekad želimo slične knjige, a nekad možda nešto malo drugačije. Uvek je teško odlučiti se za tu sledeću knjigu, jer ni u jednom slučaju nam niko ne garantuje da se u toj knjizi nalazi ono što baš nas zanima. Ili ćemo odabrati knjigu poznatog autora koja ne mora biti ni slična već pročitanim, ili biramo nasumično knjigu koja može, ali i ne mora, biti zanimljiva. Problem može da nastane i kod kupovine knjige za poklon jer često ne znamo šta bismo kupili osobi iako znamo šta voli da čita i šta je čitao/la ranije. Neophodan nam je dovoljno dobar način da odaberemo sledeću knjigu.

### Pregled problema

Jedna opcija je otići u biblioteku gde bi nam stručna osoba preporučila knjigu, ali mnogi ljudi knjige kupuju preko interneta te im je ova opcija van dometa. Takođe, mnogo ljudi prosto ne želi da troši vreme na odlazak u biblioteku samo kako bi mu neko preporučio knjigu. Na internetu postoje i reader's advisory baze znanja koje preporučuju knjige, ali ili nisu dovoljno precizne, ili se plaćaju.

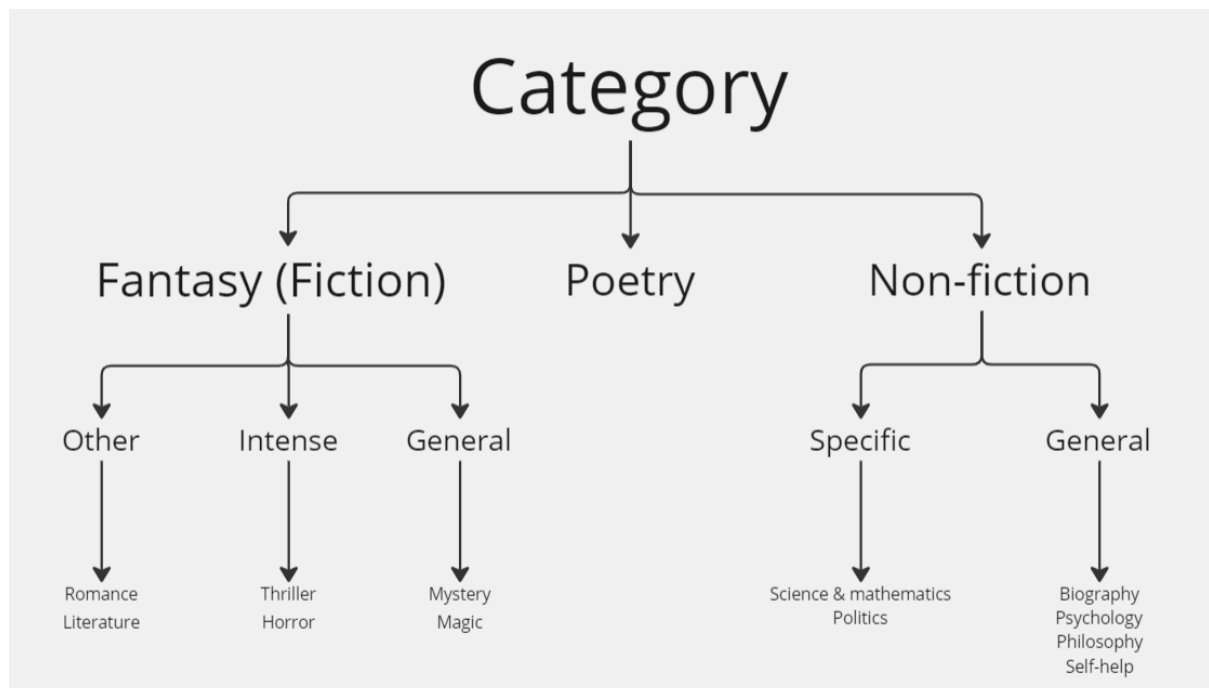
Naše rešenje ima za cilj da korisnicima olakša izbor sledeće knjige za čitanje. Sistem o korisniku prikuplja podatke. Od osnovnih podataka tu su pol i godište kako bi se vršile preporuke na osnovu slične demografije. Od korisnika se takođe zahteva da selektuje svoje preference - omiljenu kategoriju kao i neke od omiljenih ključnih pojmova. Korisnik takđe može da navede omiljene pisce kao i koja grupa godišta knjiga ga zanima kako bi dodatno suzio pretragu. Pored ovoga gledaju se i pročitane knjige kao i koji od pojmova iz njih je korisnik označio kao posebno dobre.

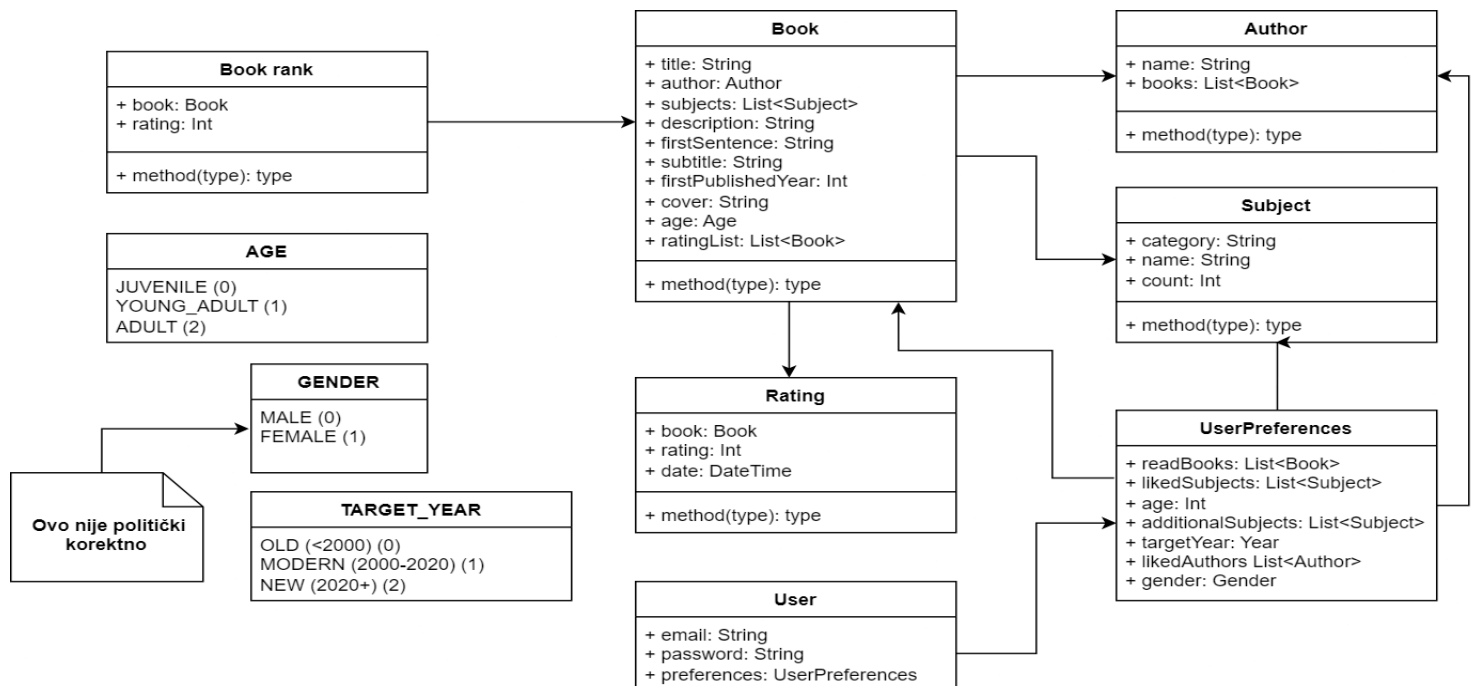
Unošenje većine podataka obavlja se pri registraciji korisnika te se kasnija preporuka knjiga svodi na prost izbor - da li korisnik želi sličnu knjigu već pročitanim, ili želi nešto drugačije. Sistem uzima sve činjenice, grubo filtrira knjige, ocenjuje ih i korisniku daje top 10 knjiga koje bi mogao sledeće da čita.

Sistem postaje sve precizniji što se više koristi jer on polako sve bolje uči šta korisnika zanima.

## Metodologija rada

Svaka knjiga ima jednu ili više kategorija. Odabrali smo nekoliko "grubih" kategorija poput fikcije, dokumentarne literatura (non-fiction) i poezije. Te kategorije su dalje razložene na grupe kategorija. Konačno, na dnu stabla se nalaze precizne kategorije poput romantike, magije i biografije koje dosta blisko opisuju knjige. Uz ove kategorije, u našem skupu smo isfiltrirali oko 9000 ključnih pojmova koji se pominju u knjigama poput "truck drivers", "friendship", "mass media", "monsters", "zombie" i slično. Ključni pojmovi se sigurno nalaze u jednoj ili više pomenutih kategorija (imaju kategoriju kao parent-a). Na ovaj način ćemo omogućiti korisniku da preciznije bira pojmove koje bi želeo da čita, ili da u knjizi koju je pročitao obeleži pojam koji mu se najviše dopada. Takođe, na ovaj način smo u stanju da računamo sličnost između pojmova i tako preporučujemo knjige sa sličnim motivima.





- Book - Svaka knjiga ima naslov i autora. Poneke knjige imaju opis, prvu recenicu, sliku i podnaslov koji će nam pomoći u prikazivanju knjige na frontend-u aplikacije. firstPublishedYear je godina na osnovu koje ćemo primeniti pravila vezane za godine. Ako korisnik želi da čita najnovija izdanja ili starija izdanja to će moći da bira. Age je starosna grupa knjige. Knjiga može biti specifično za decu, za tinejdžere ili za sve uzraste. Uz knjigu je pridružena lista ključnih pojmova kao i lista ocena za tu knjigu.
- Author - Autor ima svoj naziv i listu knjiga koju je napisao. Taj podatak nam pomaže prilikom preporuke knjiga od istog autora.
- Subject - Ovde spadaju glavne kategorije i svi ključni pojmovi. Category predstavlja roditeljsku kategoriju dok naziv predstavlja naziv ključnog pojma. Uz subject je pridružen i broj ponavljanja koji predstavlja "relevantnost" motiva u odnosu na to koliko puta se taj motiv pojavljuje u ostalim knjigama u bazi podataka
- Book rank - Objekte ove klase će insertovati i menjati sam Drools gde će im dodavati relevantnost ili oduzimati relevantnost na osnovu pravila koje budemo definisali.
- Rating - Ovo je ocena knjige. Poneke knjige će imati ocenu čitaoca na osnovu kojih će im se povećavati relevantnost prilikom preporučivanja. Uvek ćemo prednost davati bolje ocenjenim knjigama
- User - Korisnik ima email i password na osnovu kojih se loguje na sistem. Po potrebi ćemo dodati i admina koji ima CRUD prava nad svim klasama aplikacije. Korisnik takođe ima i user preferences koje predstavljaju sviđanja tog korisnika.
- UserPreferences - Ovaj objekat nam služi da zaključimo koje knjige ćemo uzeti u obzir prilikom preporučivanja, a koje ćemo odbaciti.

# Baza znanja

Prvi nivo - Ovde grubo filtriramo knjige na osnovu uzrasta korisnika i glavne kategorije.  
Pravila otprilike izgledaju ovako:

```
when age > 22  
then insert (flag_adult)
```

```
when age > 12  
then insert (flag_ya)
```

```
when age > MIN_AGE  
then insert (flag_juvenile)
```

```
when flag_juvenile and not flag_adult and not flag_ya  
then punishNonJuvenileBooks() and insert (flag_no_horror)
```

```
when flag_ya and flag_juvenile and not flag_adult  
then punishAdultBooks()
```

```
when fiction  
then punishAllPoetry() and punishAllNonFiction()
```

```
when poetry  
then punishAllFiction() and punishAllNonFiction()
```

```
when nonfiction  
then punishAllFiction() and punishAllPoetry()
```

```
when fiction and not literature  
then punishLiterature()
```

```
when fiction and not romance  
then punishRomance()
```

```
when fiction and not thriller  
then punishAllThriller()
```

```
when fiction and (not horror or flag_no_horror)  
then punishAllHorror()
```

```
when fiction and not mystery  
then punishAllMystery()
```

```
when fiction and not magic  
then punishAllMagic()
```

```
when nonfiction and not sciencemathematics
then punishAllScienceAndMathematics()
```

```
when nonfiction and not politics
then punishAllPolitics()
```

```
when nonfiction and not biography
then punishAllBiography()
```

```
when nonfiction and not psychology
then punishAllPsychology()
```

```
when nonfiction and not philosophy
then punishAllPhilosophy()
```

```
when nonfiction and not selfhelp
then punishAllSelfHelp()
```

Drugi nivo - Ovde ćemo svaku knjigu modifikovati na osnovu preference godina izdavanja knjige, autora koji se korisniku sviđaju i kategorija koje mu se sviđaju. Izbacićemo knjige koje je korisnik pročitao. Otprilike ovako:

```
when target_year
then addPointsForPubilshedYear(target_year)
```

```
when liked_subjects
then addPointsForSubjects(liked_subjects)
```

```
when liked_authors
then addPointsForAuthors(liked_authors)
```

```
when read_books
then removeBooks(read_books)
```

Treći nivo - Ovde modifikujemo knjige na osnovu njihovih ocena iz baze podataka koje su dali drugi korisnici, pružajući prednost knjigama sa boljom ocenom. Takođe, ovde ćemo razmotriti i dodatne kategorije i ključne reči koje je korisnik uneo da bi voleo da postoje u knjigama koje ćemo mu preporučiti.

Četvrti nivo - Preostalo je da prikupimo podatke od drugih korisnika koji su slični datom korisniku po godistu i polu i da vidimo kakve kategorije oni vole i koje su knjige čitali. Dodelićemo više poena tim knjigama i tim kategorijama kako bismo suzili naš izbor.

Peti nivo - Na kraju biramo X najbolje rangiranih knjiga i Y nasumično dodatih knjiga kako bismo pružili veći izbor korisniku.

## Primer za backward

Za opciju koja pronalazi nekoliko sličnih knjiga na osnovu zadate tačno jedne knjige bez gledanja korisnikovih preferenci, koristićemo backwards chaining za pronalazak sličnosti po kategorijama. Kategorije su organizovane u stablo kategorija pa ćemo na osnovu toga da vidimo sličnost jedne kategorije sa drugim kategorijama. Akumulacijom ovih podataka i odabira top X najsličnijih kategorija, možemo preporučiti korisniku knjige koje sadrže baš te kategorije.

Napravićemo nekoliko vrsta izveštaja koristeći backwards chaining:

Prva vrsta izveštaja će biti "o tebi" odnosno o korisniku (slično spotifyevom godišnjem izveštaju za muziku koju smo slušali tokom cele godine). Tu ćemo prikazivati njegove omiljene kategorije, autore i knjige koje su mu se najviše svidеле.

Druga vrsta izveštaja predstavlja najpopularnije kategorije, autore i knjige na nivou cele platforme posmatrajući sve korisnike. Prikazivaćemo statistiku po polu i uzrastu.

## Primer za template

Grube filtracije po kategorijama bi bili template-i pri čemu bismo generalizovali po kategoriji. Filtracija po godinama bi takodje bila template.

Na nivou gde modifikujemo naš BookRank objekat bismo napravili template pravila za dodavanje ili oduzimanje poena kod nekih jednostavnijih pravila koja funkcionišu na sličan način - proveravaju da li se parametar XYZ nalazi ili ne nalazi u knjizi i na osnovu toga dodaju ili oduzimaju poene.