

# SBNZ ~ Predlog projekta

Milica Đumić, sw~27~2018

## Preporuka knjiga

### Pregled problema

Problem koji se rešava jeste preporuka knjiga korisniku. Parametri koji bi se uzimali u obzir su: žanrovi koji se dopadaju korisniku, prethodna sviđanja i sviđanja korisnika označili da im se sviđa knjiga koja korisnik gleda. Takođe, ne bi svi drugi korisnici aplikacije bili uzeti u obzir podjednako. Sa rastom broj sviđanja, korisnik se uzima više u obzir pri odabiru knjiga za preporuku.

**Žanrovi knjiga:** romantika, istorija, naučna-fantastika, bojanke, enciklopedije, trileri, horor, drama, komedije, sport, klasici,...

Kada se posmatraju knjige koje se sviđaju drugim korisnicima (koji su označili da im se dopada trenutno posmatrana knjiga), rang knjige iz liste sviđanja se dodaje:

Broj knjiga koje se sviđaju korisniku	Glas za knjigu vredi
Manje od $\frac{n}{10}$	0
$\frac{n}{10}$ do $\frac{3*n}{10}$	0.25
$\frac{3*n}{10}$ do $\frac{5*n}{10}$	0.5
$\frac{5*n}{10}$ do $\frac{7*n}{10}$	0.75
Više od $\frac{7*n}{10}$	1

n - najveći broj sviđanja u sistemu

Objašnjenje postojećeg algoritma sam pronašla za preporuku knjiga sa internet sajta Goodreads. Konkretno se u obzir uzimaju prethodna sviđanja i nesviđanja, žanrovi koji se dopadaju ulogovanom korisniku i šta se drugim korisnicima dopada (a da su pročitali posmatranu knjigu). Kao dodatak ovom

algoritmu, smatram da bi pomoglo da se ne uzimaju svi korisnici u obzir jednako. Korisnici koji su malo interagovali sa aplikacijom možda ne poznaju svoj čitalački ukus, te će navesti samo par knjiga bez mnogo zajedničkih karakteristika.

## Motivacija

Kod preporuke bilo kog proizvoda, često dolazi do promašaja. Motivacija za rešavanje ovog problema je uzimanje u obzir šta drugi korisnici (koji su rekli da im se sviša trenutno posmatrana knjiga) označe da im se sviđa. Dobra preporuka bi korisnika navela da koristi više aplikaciju ili je preporuči još nekome, ali naravno primarno je da pomogne korisniku da nađe odgovarajuću knjigu za sebe. Znanje da aktivnost korisnika utiče na dalje preporuke može dodatno motivisati korisnike jer će znati da se uzima u obzir njihovo mišljenje.

## Metodologija rada

**Ulazni parametri:** žanrovi koji se dopadaju korisniku, prethodna svidanja i knjiga koja se trenutno posmatra.

**Izlazni parametri:** 5 knjiga koje su najbolje rangirane za date ulazne podatke.

Da bi sistem pravilno radio potrebne su nam knjige, korisnici i njihova prethodna svidanja. Rešenje bi evaluiralo na osnovu unetih korisničkih svidanja, kao i količine podataka u sistemu. Što više podataka imamo, to će se bolje preporučivati knjige.

## Reprezentativan primer rezonovanja

P1: ulaz je knjiga koja se posmatra, proverava se svaki korisnik sistema i ukoliko je pročitao datu knjigu on se uzima u obzir, izlaz je lista korisnika kojima se sviđa posmatrana knjiga.

P2: ulaz su lista korisnika kojima se sviđa posmatrana knjiga (rezultat iz P1) i najveći broj sviđanja, za svakog se računa njegov udeo (kao u tabeli), izlaz je vrednost koliko svaki od tih korisnika učestvuje u preporuci.

P3: ulaz su lista knjiga koja se sviđa korisniku (korisnici su rezultat iz P1) i vrednost koliko dati korisnik učestvuje u oceni (vrednosti su rezultat iz P2), svakoj knjizi se doda vrednost koliko korisnik učestvuje, izlaz lista svih knjiga sa glasovima.

P4: ulaz su žanrovi koji se dopadaju ulogovanom korisniku, za svaku knjigu u žanru se dodaje 1 na trenutne glasove, izlaz lista svih knjiga sa glasovima.

P5: ulaz je lista knjiga koje se sviđaju ulogovanom korisniku [A] (rezultat pravila P1) i lista svih knjiga sa konačnim glasovima [B] (rezultat pravila P3 i P4), izlaz je razlika ova dva skupa  $B/A$  (ne predlažu se knjige koje se već dopadaju korisniku).

P6: ulaz je rangirana lista knjiga (rezultat pravila P6) i trenutno posmatrana knjiga, izlaz je rangirana lista knjiga bez posmatrane knjige (ne predlažu se knjige koje se već dopadaju korisniku).

P7: ulaz je knjiga lista svih korisnika sistema, proverava se koji korisnik ima najviše sviđanja i ta vrednost se postavlja za najveći broj sviđanja.

**Primer forward-chaining:** pravila P1, P2 i P3 imaju povezane ulaze i izlaze i iz tog razloga su reprezentativan primer.

**Primer template-a:** pravilo P8 i P9 (korisnik označi da mu se sviđa (ili poništi sviđanje) knjiga i sistem po potrebi izmeni najveći broj sviđanja). ???

**Primer akumulatorskih funkcija:** pravilo P1 iterira kroz listu knjiga koje se dopadaju korisniku i računa sumu knjiga koje imaju ISBN isti kao i posmatrana knjiga (opcije su ili 0 ili 1). U slučaju kada je vrednost 1 tada se dati korisnik uzima u obzir.

**Primer međusobno isključivih pravila:** dodela vrednosti na osnovu tabele će biti delegirana u više pravila i biće moguće ispuniti samo jedan uslov od svih tih.

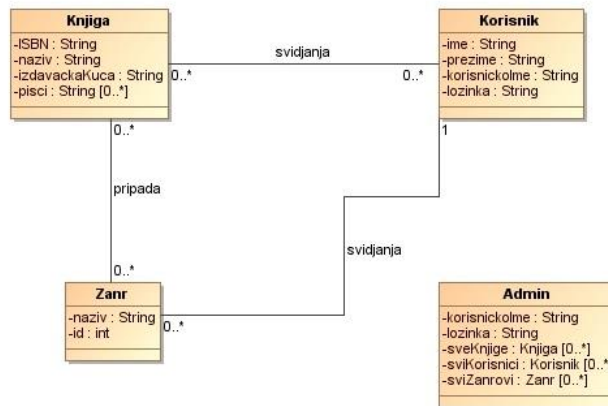
P2.1 korisnik ima  $k$  svidjanja i  $k < \frac{n}{10}$ , vraća vrednost 0.

P2.2 korisnik ima  $k$  svidjanja i  $k \in [\frac{n}{10}, \frac{3*n}{10})$ , vraća vrednost 0.25.

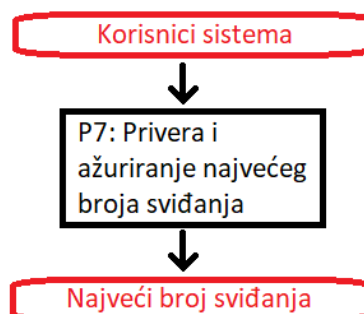
P2.3 korisnik ima  $k$  svidjanja i  $k \in [\frac{3*n}{10}, \frac{5*n}{10})$ , vraća vrednost 0.5.

P2.4 korisnik ima  $k$  svidjanja i  $k \in [\frac{5*n}{10}, \frac{7*n}{10})$ , vraća vrednost 0.75.

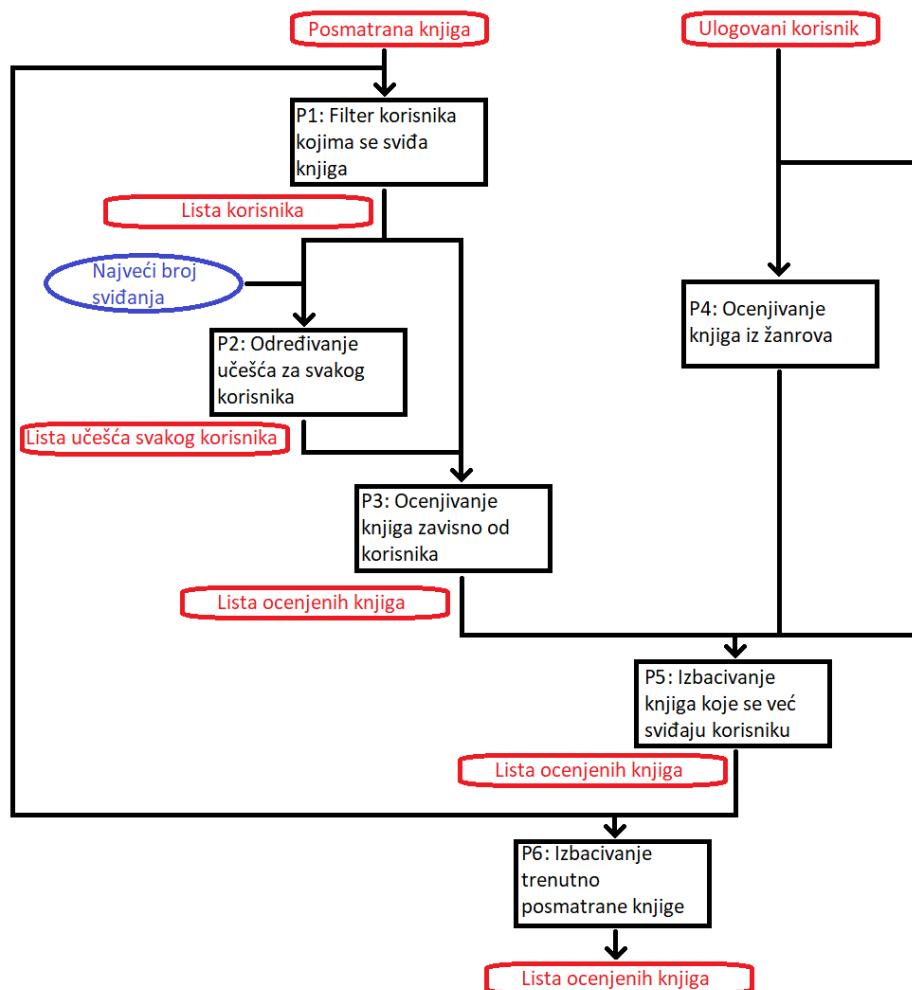
P2.5 korisnik ima  $k$  svidjanja i  $k > \frac{7*n}{10}$ , vraća vrednost 1.



Slika prikazuje klasni dijagram



Slika prikazuje pravilo P7



Slika prikazuje povezanost pravila P1-P6

## Literatura

<https://www.goodreads.com/blog/show/303-announcing-goodreads-personalized-recommendations#:~:text=The%20Goodreads%20Recommendation%20Engine%20combines,will%20want%20to%20read%20next.>