

Specifikacija projekta

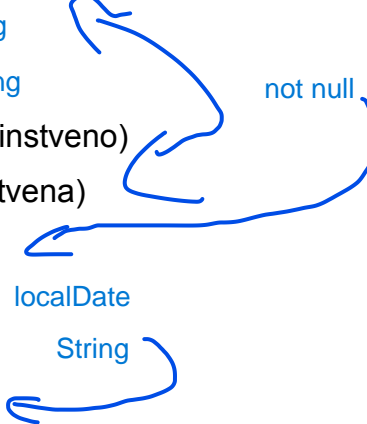
Web programiranje 2022/2023

Potrebno je kreirati web aplikaciju koja korisnicima omogućava praćenje pročitanih knjiga, pisanje recenzija, kao i pravljenje različitih lista knjiga (polica) - po uzoru na [Goodreads](https://www.goodreads.com/).

Pristup aplikaciji imaju i prijavljeni i neprijavljeni korisnici. Prijavljeni korisnici mogu biti: čitaoci, autori i administratori.

Model podataka


Korisnik

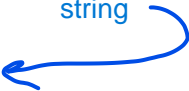
- Ime String
 - Prezime String
 - Korisničko ime (jedinstveno)
 - Mejl adresa (jedinstvena)
 - Lozinka
 - Datum rođenja LocalDate
 - Profilna slika String
 - Opis
 - Uloga (Čitalac, Autor, Administrator) enum @Enumerated
- 

Ako je korisnik Autor dodatno ima

- Da li je nalog aktivan (boolean) extends User
- Spisak knjiga HashSet

Knjiga

- Naslov string
 - Naslovna fotografija
 - ISBN (međunarodni identifikator knjige)
- 

- Datum objavljivanja `localDate`
 - Broj strana `integer`
 - Opis `string`
 - Žanr
 - Ocena `BigDecimal`
- 

Žanr

- Naziv `String`

Polica

- Naziv `String`
- Oznaka da li je primarna `bool`
- Stavka police


Stavka police

- Recenzija
- Knjiga

Recenzija

- Ocena `integer`
- Tekst `String`
- Datum recenzije `LocalDate`
- Korisnik

Zahtev za aktivaciju naloga autora

- Email `String`
 - Telefon
 - Poruka
 - Datum `localDate`
 - Status (na čekanju/odobren/odbijen) `Enum`
- 

Funkcionalnosti

Neprijavljen korisnik

- Prva stranica koju neprijavljeni korisnik vidi jeste početna stranica, ovde korisnik može da se prijavi na sistem ili registruje, ukoliko nema aktivan nalog.
- Neprijavljeni korisnik takođe ima mogućnost pretrage knjiga, kao i njihovog pregleda.
- Osim knjiga, neprijavljeni korisnik ima mogućnost pregleda: recenzija, žanrova, profila korisnika, kao i njihovih polica – lista knjiga. Za svaku knjigu moguće je videti njene recenzije.
- Neprijavljeni korisnik može još i da podnese zahtev za aktivaciju naloga autora, o čemu će biti reči posle.
- Da bi se korisnik registrovao potrebno je da unese svoje ime, prezime, jedinstveno korisničko ime, mejl adresu i lozinku, koju takođe treba da potvrdi ponovnim unosom. Uneta mejl adresa takođe treba da bude jedinstvena u sistemu. Uspešnom registracijom otvara se profil čitaoca.
- Administratori se programski učitavaju iz baze i ne mogu se naknadno dodati. Autora može kreirati samo administrator. Čitalac ne može da postane autor.
- Korisnik se prijavljuje tako što unosi svoju mejl adresu i lozinku. Početna stranica prijavljenog korisnika prikazuje listu njegovih polica.

Čitalac

- Čitalac odmah nakon otvaranja naloga ima tri primarne police: *Want to Read*, *Currently Reading* i *Read*. Ove police se ne mogu obrisati. One predstavljaju liste knjiga koje korisnik želi da pročita, trenutno čita i onih koje je već pročitao.
- Korisnik ima mogućnost dodavanja drugih polica, koje kasnije može i obrisati. Korisnik može da napravi novu policu samo ukoliko već nema policu sa izabranim imenom.
- Čitalac može da stavi određenu knjigu na najviše jednu od tri primarne police i na proizvoljan broj polica koje je sam kreirao. Međutim, da bi dodao knjigu na neku od polica koje je sam napravio mora je prvo dodati na jednu od primarnih polica.

Ovo znači da ukoliko se knjiga ukloni sa jedne od primarnih polica ona će se ukloniti i sa svih drugih korisnikovih polica.

- Ukoliko se knjiga doda na *Read* policu otvara se dijalog za dodavanje recenzije. Ovde korisnik može da izabere na koje još od svojih polica želi da doda knjigu, može da izabere jedan ili više datuma kada je knjiga pročitana, da dodeli ocenu knjizi, kao i da podeli svoje mišljenje o njoj. Recenzije se mogu izmeniti. Ukoliko se knjiga ukloni sa *Read* police, korisnikova recenzija se briše.
- Korisnik može da ažurira svoj profil. Osim što ima opciju da izmeni svoje ime i prezime, može da doda svoj datum rođenja, profilnu sliku, kao i da napiše nešto o sebi. Ukoliko želi da promeni svoju lozinku ili mejl adresu neophodno je da prvo ponovo unese svoju trenutnu lozinku.

Autor

- Autor ima pristup svim funkcionalnostima koje su dostupne čitaocima.
- Autor može da doda nove knjige koje je napisao, kao i da ažurira postojeće.
- Korisnika ovog tipa u sistem može dodati samo administrator.

Administrator

- Administrator dodaje novog autora u sistem. Autorov nalog prvobitno nije aktivan. Neprijavljeni korisnik na profilu autora može kliknuti na dugme za podnošenje zahteva za aktivaciju naloga. Pritiskom na dugme otvara se dijalog gde je potrebno uneti mejl adresu i broj telefona, takođe se može ostaviti dodatna poruka.
- Administrator ima uvid u sve zahteve za aktivaciju koji nisu pregledani. Zahtev je moguće prihvatiti ili odbiti. Ukoliko administrator odbije zahtev, biće poslat mejl na unetu adresu koji obaveštava korisnika o donetoj odluci. Međutim, ako se zahtev prihvati, nalog autora će se aktivirati – biće kreirane primarne police i na mejl adresu iz zahteva će biti poslat mejl sa lozinkom od novootvorenog naloga.
- Osim rada sa zahtevima za aktivaciju naloga, administratori mogu i da dodaju nove žanrove u aplikaciju, kao i da dodaju, ažuriraju i brišu knjige. Međutim, samo knjige bez recenzija mogu biti obrisane. Administratori takođe mogu da ažuriraju profile neaktiviranih naloga autora.

Ocenjivanje projektnog zadatka

- Za verzioniranje potrebno je koristiti git. Napraviti **privatan** repozitorijum na <https://github.com> i dodati **WebProgramiranjeIN** kao kolaboratora.
- Potrebno je prijaviti timove u dokumentu na [linku](#).
- U git istoriji se mora jasno videti tok razvoja u koji moraju biti uključeni svi članovi tima. Svaki član tima na svakoj kontrolnoj tački (osim prve) mora da ima **minimum 2 grane i 5 komita** koji se odnose na funkcionalnosti za tu kontrolnu tačku. **Neće biti priznati projekti čiji su svi komiti napravljeni 2 dana pre odbrane kontrolne tačke**, mora se jasno videti kontinuitet rada na projektu. Za svaku kontrolnu tačku potrebno je da se vidi da je rađeno **minimum nedelju dana**.
- Dizajn korisničkog interfejsa kao i stilovi su prepušteni studentu. Za olakšanje formiranja layouta stranica studenti mogu koristiti neki od postojećih framework-a (Bootstrap, MaterializeCSS...)
- Serverski deo aplikacije treba da bude realizovan korišćenjem Java programskog jezika i Spring Boot radnog okvira.
- Zadatak se sastoji iz tri kontrolne tačke:
 1. Predstavlja model (napravljene klase i veze između njih)
 2. Predstavlja funkcionalnosti backenda (Java Spring Boot deo)
 3. Predstavlja uvezivanje sa frontendom (HTML, CSS, JS i opciono Vue.js)