

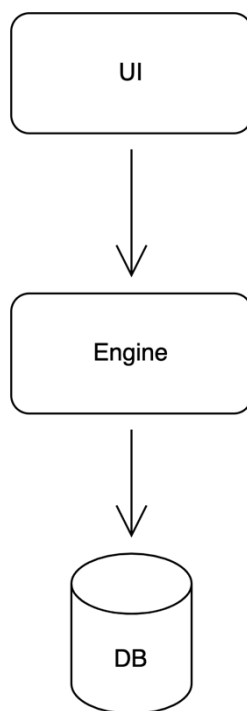
Projektni zadatak DRES

Drustvena mreža

Implementirati projekat koji simulira društvenu mrežu sličnu današnjim.

Implementacija treba da sadrži 3 komponente:

1. Korisnicki interfejs (UI)
2. Servis za obradu zahteva i podataka (Engine)
3. Bazu podataka (DB)



Skica sistema

Korisnicki interfejs (UI)

Korisnicki interfejs je Flask web aplikacija koja treba da opsluzi korisnika .

Inicijalno aplikacija treba da sadrži administratorski nalog. Administrator ima svoje:

1. Ime
2. Prezime
3. Adresa
4. Grad
5. Drzava
6. Broj telefona
7. Email
8. Lozinka
9. Korisničko ime

Prva od opcija koja je omogućena administratoru nakon prijave je kreiranje novog korisničkog naloga. Nakon što administrator doda korisnički nalog korisnik može da se prijavi. Za kreiranje korisnika potrebno je popuniti iste podatke kao što ima i administrator. Prilikom kreiranja korisnika email i korisničko ime moraju biti jedinstveni.

Kada administrator registruje korisnika, korisnik na email dobija poruku da je registrovan. U poruci se nalaze i njegovi kredencijali (korisničko ime i lozinka).

Korisnik treba da se prijavi na svoj nalog, a zatim nakon prve prijave administrator treba da dobije obavještenje putema mail-a o prijavi korisnika.

Obični korisnik treba da podrži sledeće mogućnosti:

1. Prijava na nalog (korisničko ime i lozinka)
2. Izmena korisničkog naloga (svih polja)
3. Mogućnost postavljanja objave na svoj nalog, izmjene i brisanja svojih objava.
4. Mogućnost pretrage drugih korisnika uz pomoć email adrese, korisničkog imena, imena, prezimena, adrese, grada...
5. Mogućnost da korisnik doda u prijatelje neki drugi nalog.
6. Korisnik može da prihvati zahtjeve za prijateljstvo i izbriše prijatelje i zahtjeve.
7. Korisnik može da postavi objavu. Objava može da sadrži sliku i tekst.
8. Korisnik može da pregleda objave samo svojih prijatelja.
9. Objave treba da budu sortirane po vremenu objavljivanja.

Korisnike se loguje putem:

- Korisničkog imena i
- Lozinke

Proces objave statusa:

Korisnik unosi svoju objavu i šalje je. Objave se izlistavaju kod administratora koji ih odobrava ili odbija. Kada korisnik kreira objavu potrebno automatski poslati mail administratoru da je objava kreirana, i isto tako obavijestiti korisnika da li je objava prihvaćena ili odbijena. Administrator uživo može da prati pristizanje zahtjeva za odobravanje objava (bez osvježavanja stranice, upotrijebiti **websocket**). Ako se korisniku objava odbije, vraća mu se na ispravku. Korisnik može kreirati više objava i poslati ih na provjeru. Ukoliko korisniku bude odbijeno više od 3 objave automatski mu se blokira pristup nalogu i šalje obavještenje putem maila. Administrator ima listu blokiranih korisnika i može

da ih odblokira.

Servis za obradu zahteva I podataka (Engine)

Engine je servis implementiran kao flask API aplikacija. Engine ima svoje endpointe koje prikazuje eksternom svetu (UI aplikaciji) za koriscenje. UI deo poziva endpointe Engine-a radi obrade raznih zahteva I podataka. Pri tome samo Engine komunicira sa bazom, a UI sa Engine-om.

Baza podataka (DB)

Baza podataka je u komunikaciji sa Engine-om za svrhu skladistenja podataka o aplikaciji. U bazi se skladiste svi esencijalno bitni podaci za rad aplikacije.

Model baze kao I tip baze (NoSQL, SQL) je proizvoljan (ne može biti SQLite).