

Računarski praktikum 2 – Druga zadaća: Chat

Ponukan rastućom potrebom za alatima za online komunikaciju, Mirko je došao na *originalnu* ideju za web-aplikaciju koja implementira chat. U toj aplikaciji:

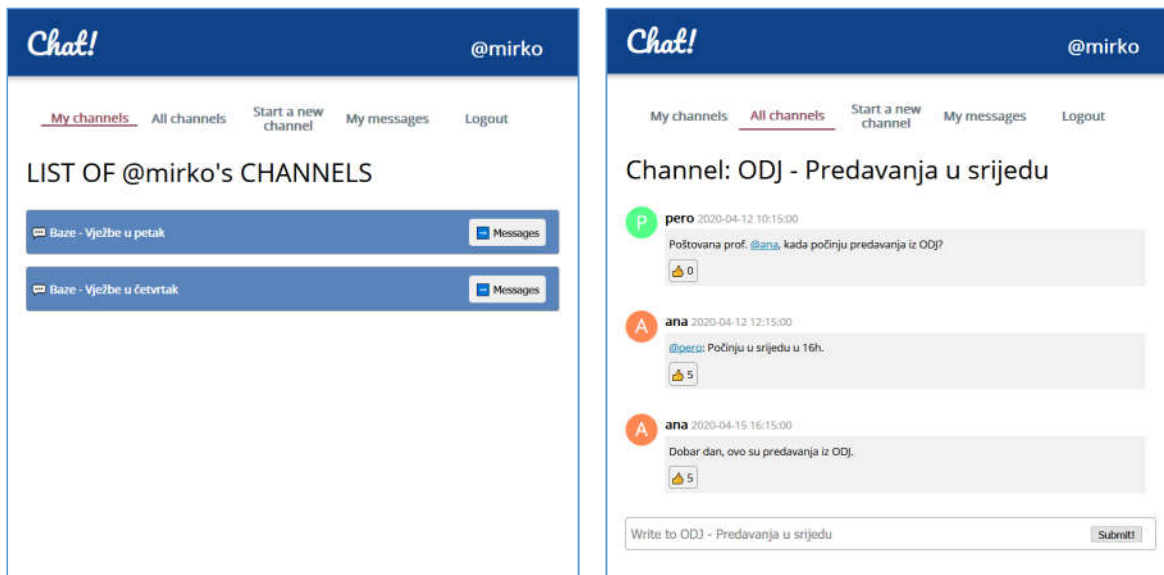
- Korisnici mogu pokretati kanale za raspravu o nekoj temi.
- Unutar pojedine rasprave, korisnici mogu objavljivati poruke. Za svaku poruku je vidljivo tko ju je objavio, kada ju je objavio, kao i njezin sadržaj. Poruci je moguće dati „palac gore“.
- Poruke mogu sadržavati korisnička imena, koja imaju oblik @username.

Pomozite Mirku da implementira ovu web-aplikaciju u PHP-u.

Konkretno, potrebno je implementirati sljedeće:

- Na adresi
https://web.math.pmf.unizg.hr/nastava/rp2d/slideovi/2020/create_tables.php.txt
možete skinuti datoteku `create_tables.php` koja inicijalizira tri tablice u bazi podataka.
 1. `dz2_users` sadrži podatke o korisnicima (id, korisničko ime, hashirani password, e-mail, niz za registraciju, te oznaku o uspješnoj registraciji).
 2. `dz2_channels` sadrži podatke o kanalima (id, id korisnika koji ga je pokrenuo, naziv kanala).
 3. `dz2_messages` sadrži podatke o porukama (id, id kanala, id korisnika koji ju je objavio, sadržaj poruke, vrijeme objavljivanja, broj palaca gore).
- Na adresi
https://web.math.pmf.unizg.hr/nastava/rp2d/slideovi/2020/seed_tables.php.txt
možete skinuti datoteku `seed_tables.php` koja popunjava te tablice nekim testnim podacima.
- Prilagodite podatke za spajanje na bazu, te inicijalizirajte tablice unutar svoje baze na rp2 serveru tako da izvršite skripte `create_tables.php` i `seed_tables.php`.
- Kada korisnik pristupi početnoj stranici aplikacije, prikazuje mu se forma za ulogiravanje.
- Po uspješnom ulogiravanju (provjera username / password), korisniku se prikazuje izbornik u kojem može odabrati:
 1. Popis svih kanala koje je on pokrenuo. Klikom na ime nekog kanala iz popisa prikazuju se poruke unutar tog kanala, kronološkim redom.
 2. Popis svih kanala koji postoje. Klikom na ime nekog kanala prikazuju se poruke unutar tog kanala, kronološkim redom.
 3. Stvaranje novog kanala. Za novi kanal je potrebno unijeti samo njegovo ime.
 4. Popis svih poruka koje je korisnik objavio u bilo kojem kanalu. Poruke trebaju biti ispisane silaznim kronološkim redom. Kraj svake poruke treba biti link koji vodi na popis svih poruka unutar pripadnog kanala.
 5. Odlogiravanje korisnika.
- Prilikom popisivanja poruka unutar nekog kanala, za svaku poruku je navedeno tko ju je napisao i kada, zatim sadržaj poruke, te broj „palaca gore“ za tu poruku. Palčevi su „klikabilni“, tj. posjetitelj web-stranice može za svaki post po volji mnogo puta povećati taj brojač (čak i za svoje poruke!). Na dnu popisa treba biti forma koja omogućava unos nove poruke u tom kanalu ulogiranom korisniku. Klikom na palac ili unosom nove poruke se ponovno učitava ista stranica s popisom svih poruka tog kanala.
- Nakon ulogiravanja se (osim izbornika) prikazuje ono što je opisano u točki 1.
- Bonus za dodatni bod: Kad god se u nekoj poruci pojavljuje podstring @username, taj podstring treba prikazati kao link koji vodi na prikaz svih poruka korisnika s imenom username. Taj prikaz napravite kao onaj iz točke 4.

- Primjer prikaza stranice iz točke 1 je lijevo, a primjer prikaza svih poruka iz nekog kanala je desno (ne morate napraviti identičan; izgled nije bitan nego gore opisana funkcionalnost):



Bonus za još 2 dodatna boda (povrh onog jednog gore):

- Uočite da gore opisani zadatak ne zahtijeva dodavanje novih korisnika. U tablici `dz2_users` vam zapravo trebaju samo prva tri stupca (`id`, `username`, `password`).
- Za dodatna 2 boda, omogućite da korisnici otvaraju nove korisničke račune na način na koji je to opisano u Zadatku 6 iz Zadataka za vježbu. Za to će vam trebati i preostala tri stupca u `dz2_users`.

Bonus za još 1 dodatni bod:

- Zadatak implementirajte korištenjem objektno-relacijskog preslikavanja (ORM), kako je opisano u Predavanjima 7 i pripradnom screencastu. ORM trebate implementirati tako da su klase u modelu izvedene iz bazne klase `Model`. U baznoj klasi se ne smijete referirati na pojedine izvedene klase (tj. imati nešto poput `if(imelzvedeneKlase === 'User') ...`), nego ona treba biti posve generička. Smijete se referirati na popis i tipove stupaca (vidi iduću točku).
- U bazoj klasi `Model` smijete uz `$table` dodati i popis atributa `$attributes` kao statičko polje. U tom polju će izvedena klasa navesti popis svih stupaca u pripadnoj tablici i njihovih tipova, npr.

```
class Message extends Model {
    protected static $table = 'dz2_messages';
    protected static $attributes = ['id'=>'int', 'content'=>'string', 'date'=>'datetime', ...];
}
```

Sugestije:

- Jako se preporuča korištenje MVC (jer olakšava organizaciju koda), ali ono nije nužno.
- Svakako koristite *prepared statements* za unos poruka u bazu!
- Za pristup porukama kanala čiji je `id`, na primjer, jednak 237864 možete koristiti linkove poput:
`chat.php?id_channel=237864` ili
`chat.php?rt=channel/show&id_channel=237864`
 pa onda u toj skripti (ili kontroleru) dohvatiti `$_GET['id_channel']`.
- Vezu između tipova podataka za spremanje vremena u MySQL i PHP možete vidjeti npr. ovdje:
<http://www.richardlord.net/blog/php/dates-in-php-and-mysql.html>

Upute za postavljanje aplikacije na web i slanje rješenja:

- Aplikacija se smije sastojati od jedne ili više PHP, HTML, CSS i slikovnih datoteka.
- Za rješavanje nemojte koristiti JavaScript, nego isključivo PHP.
- Sve datoteke od kojih se sastoji aplikacija zapakirajte u datoteku `chat.zip` (čak i u slučaju kad se sastoji samo od jedne datoteke!), te tu datoteku uploadajte kroz sustav za zadaće na adresi: <http://www.math.hr/nastava/rp2d/zadace.php>
- Nadalje, aplikaciju postavite tako da bude dostupna preko rp2-servera kroz ovu adresu (pazite na velika i mala slova!):
<http://rp2.studenti.math.hr/~username/dz2/chat.php>
Prava pristupa za direktorij `/student1/username/public_html/rp2/dz2` prije početka izrade zadaće postavite na `rwx-----x (701)`.
- Koristite svoju MySQL bazu na rp2 serveru.