

## Peta domaća zadaća

### o **Zadatak**

Za petu domaću zadaću trebate napraviti stranicu "Kokodakalo". Radi se o stranici sličnoj Twitteru koja služi za razmjenu kratkih poruka.

Poruke se spremaju u bazu i sastoje se od vremena kad je poruka poslana, naslova (max. 80 znakova) i sadržaja (max. 255 znakova). U sadržaj je dopušteno pisati HTML kod, ali ne smije sadržavati `script` tag. Tu provjeru možete napraviti ovako:

```
if (preg_match('/<\s*script[^\>]*>/i', $sadrzaj)) greska();
```

Kokodakalo se sastoji od četiri cjeline:

1. internetska stranica koja prikazuje sve poruke (1 bod),
2. RSS usluga koja vraća 10 najnovijih poruka (3 boda),
3. internetska usluga (API) za dodavanje poruka (3 boda),
4. aplikacija koja koristi API za dodavanje poruka (3 boda).

Internetska stranica prikazuje sve poruke i mora omogućiti straničenje po 10 poruka kako ne bi bila preduga nakon što se pošalje puno poruka. Pretpostavljeno se prikazuje najnovijih 10 poruka. Dodatno, korisnici se moraju moći registrirati preko te stranice.

Stranica se mora referencirati na RSS i sadržavati malu sličicu s RSS logom koja je link na RSS.

Naslov, opis i TTL RSS usluge je proizvoljan. Svaka obavijest (`item`) mora imati naslov, sadržaj, autora i vrijeme objave kao i u poruci iz baze. Za `guid` možete koristiti neku od preporučenih metoda s predavanja.

Jednostavnosti radi, u bazu možete spremati *timestamp* (kao `unsigned integer`) dobiven pozivom na PHP-ovu funkciju `time()`. Vrijeme objave mora biti prema RFC822, a to dobijemo na ovaj način: `$vrObjave = date('r', $timestamp); # $timestamp = time();`

Moraju biti definirani sljedeći API-ji: `login(user, pass)` i `post(key, title, message)`. Parametri u zagradama šalju se preko URL-a. Kao u predavanjima, `login` vraća ključ (*key*) za korištenje API-ja koji zahtijevaju prijavu. `post` zahtijeva prijavu kako bi se znalo tko je poslao poruku i kako ne bi bilo tko mogao slati poruke. Nije bitno da li se koristi XML ili JSON kao zapis odgovora.

Dopušteno je definirati i ostale API-je prema potrebi. Npr. za dohvaćanje poruka ili registraciju.

Aplikacija za dodavanje poruka preko API-ja mora imati neki način prijave korisnika, ali ne smije znati koji korisnici postoje. Prijavu je potrebno provjeravati isključivo preko `login`. Nakon uspješne prijave, ključ je moguće spremati u sjednicu i koristiti ga kasnije prilikom slanja poruke.

Nakon što se korisnik prijavi, potrebno je omogućiti slanje poruke.

### o **Napomena**

Aplikaciju za slanje poruka preko API-ja možete implementirati u **bilo kojem programskom jeziku!** Pokažite svoju kreativnost!

Možete napraviti mobilnu aplikaciju ili za stolno računalo (konzolnu ili grafičku) u Javi, Pythonu, Python + pyGTK, C-u, C++-u, C++ + gtkmm, C++ + wxWidgets, C# ili na bilo koji drugi način.

Ukoliko izvorni kod nije u PHP-u, neće se pregledavati i bitna je samo funkcionalnost. Ako demosi neće moći pregledati taj dio zadatka, pokazat ćete im Vaše rješenje pod predavanjima.