

#### Zadaća 4

Potrebno je kreirati sljedeću funkcionalnost i ukoliko želite možete izmijeniti tematiku stranica. Svaki ispis na ekran mora biti formatiran prema strukturi i CSS uputama definiranim u zadaći 1. Struktura direktorija ostaje ista kao što je bila u zadaći 1.

#### Opće upute koje vrijede za sve stranice

- **Izbaciti JavaScript funkcionalnost ukoliko nije drugačije navedeno u zadatku i HTML5 provjere kako bi se mogla testirati kompletna PHP funkcionalnost.**
- **Obavezno u zadaći koristiti relativne putanje.**
- Svaki ispis na ekran mora biti oblikovan prema strukturi i CSS uputama definiranim u zadaći 1.
- Struktura direktorija i datoteka ostaje ista kao što je bila u zadaći 1.
- Potrebno je preimenovati sve datoteke s ekstenzijom html u datoteke s ekstenzijom php.
- Kreirajte bazu podataka koja ima tablice potrebne za spremanje podataka koji se nalaze u obrascima. **Tablice u nazivu neka imaju prefix DZ4\_ kako bi mogli raditi na Vašim projektima dok nastavnici ocjenjuju zadaću 4.**
- Pripremite barem 5 smislenih unosa u bazi za svaku tablicu u koje se spremaju podaci iz obrazaca.
- U sustavu postoje četiri uloge: neregistrirani korisnik (neprijavljeni), registrirani korisnik, moderator i administrator. Uloge idu redom kako su navedene, a svaka viša uloga ima pravo pristupa svim stranicama niže uloge. Najjača uloga je administrator.
  - Neregistrirani korisnik može pristupiti stranicama: index.php, registracija.php, prijava.php
  - Registrirani korisnik može pristupiti svemu kao i neregistrirani korisnik plus: popis.php (opcija ažuriranja vlastitih podataka), obrazac.php
  - Voditelj može pristupiti svemu kao i registrirani korisnik plus: multimedija.php
  - Administrator može pristupiti svim stranicama kao i voditelj plus: popis.php (opcija ažuriranja svih podataka).
- Ukoliko se u preglednik unese direktno URL do neke stranice, a korisnik nema pravo pristupa toj stranici (prema ranijim uputama) skripta ga odjavljuje i prebacuje ga na stranicu prijave. U dnevnik se upisuje korisnik, uloga, putanju do skripte kojoj je pokušao pristupiti te datum i vrijeme pristupa.
- Sve stranice sadrže u navigaciji opciju 'odjava' ukoliko je korisnik prijavljen ili opciju 'prijava' ako nije prijavljen. Navigacija samo prikazuje poveznice stranica kojima korisnik temeljem svoje uloge ima pravo pristupa.
- Sve stranice moraju biti implementirane uz potpuno i ispravno korištenje Smarty sustava predložaka. Ukoliko stranica nije implementirana u potpunosti i ispravno pomoću Smarty predložaka ne mogu se ostvariti svi bodovi. Ispravno korištenje znači da u PHP datotekama nema nikakvog HTML-a. (Npr. svaki ispis neke liste se u PHP-u sprema u niz i uz pomoć Smarty petlje ispisuje na ekran sa željenim oblikovanjem).
- Preporučujemo da što manje HTML koda kombinirate sa PHP-om i da što više koristite objektno orijentirani pristup kako bi što lakše implementirali Smarty sustav predložaka.
- Implementacija:
  - u kojoj se uopće ne koristi **Smarty** može najviše nositi **7.5 bodova** iz cijele zadaće.
  - u kojoj se uopće ne koristi **Smarty** i **AJAX** može najviše nositi **5 bodova** iz cijele zadaće.
- **Pripazite da ne napravite beskonačnu petlju ili da ne šaljete PHP mail funkcijom puno poruka istovremeno. U oba slučaja poslužitelj barka.foi.hr Vas može blokirati!**

## Upute po skriptama

1. PHP datoteku (baza.class.php) s klasom za rad s bazom podataka koja minimalno 4 funkcije (čiji parametri su proizvoljni):
  - a. Funkcija za spajanje na bazu.
  - b. Funkcija za izvršavanje SELECT MySQL upita koja vraća neformatirani rezultat izvršenog upita.
  - c. Funkcija za izvršavanje ostalih vrsta MySQL upita koja vraća status izvršenog upita (uspjeh/neuspjeh).
  - d. Funkcija za prekidanje veze prema bazi.
2. PHP datoteka **registracija.php** šalje podatke samoj sebi i provjerava ispravnost unesenih podataka (validacija na strani poslužitelja) te u slučaju greške ispisuje informaciju o grešci. Skripta se koristi za dodavanje i ažuriranje podataka.
  - a. Ako se na stranicu dođe direktno putem poveznice u navigaciji tada se može dodati novi korisnik:
    - i. Provjeriti na slanje obrasca da li u polju korisničko ime već postoji takav zapis u bazi. Ako ne postoji sve je u redu te labela i element postaju zelene boje. U protivnom su crvene boje.
    - ii. Provjerite ispravnost formata datuma rođena (**bez korištenja regularnih izraza**). Format je „dd.mm.gggg.“, gdje su za datum „d“, „m“ i „g“ brojke od 0-9 uz iznimke da prva „d“ brojka može biti od 0-3, a prva „m“ brojka 0 ili 1. Korisnik mora biti stariji od 18 godina.
    - iii. Provjerite da je u elementu višestrukog odabira, odabrana najmanje jedna opcija za dozvole kolačića: uvjeti korištenja, zapamti me, popuni podatke, redoslijed prikazivanja. Ako neke od opcija nisu označene ne mogu se koristiti.
  - b. Ukoliko se na stranicu dođe preko tablice sa skripte *popis.php* tada se može raditi ažuriranje za id koji je dobiven:
    - i. Prije nego što se omogućuje ažuriranje potrebno je unijeti u polje korisničko ime postojeće korisničko ime i u polje lozinka postojeću lozinku tog korisnika. Ostala polja su automatski popunjena dobivenim podacima i ostaju onemogućena do trenutka unosa ispravnog korisničkog imena i lozinke. Ukoliko je korisnik administrator on ne mora unositi korisničko ime i lozinku već su odmah sva polja omogućena za izmjenu. Koristite JavaScript funkcije koje izlaskom iz polja šalju **AJAX zahtjev** Vašoj PHP skripti koja provjerava da postoji korisničko ime i lozinka već u bazi i vraća odgovor u ispravnom JSON formatu.
    - ii. Provjerite dozvole kolačića u elementu višestrukog odabira da je odabrana najmanje jedna opcija. Ako se neka opcija više ne koristi potrebno je očistiti prethodno korištene kolačiće.
  - c. Zajedničko:
    - i. Svi podaci svi moraju biti uneseni.
    - ii. U slučaju dodavanja novog unosa ili ažuriranja zapisuje novog korisnika u bazu podataka (koristeći funkcije iz prve točke). Lozinka mora biti zapisana u čitljivom (ovo je veliki sigurnosni propust i radimo ga isključivo za predmet i za potrebe ocjenjivanja zadaća) i kriptiranom obliku (korištenje SHA256 algoritma i dinamične soli).

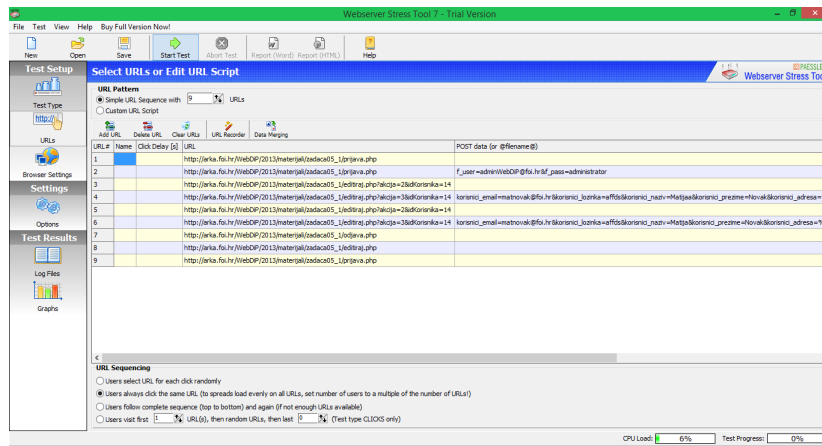
- iii. Ukoliko se radi o dodavanju osvježava se stranica i inicijalizira se obrazac. Moguće je popuniti obrazac testnim podacima ako se označi opcija “popuni podatke” na stranici registracije.
  - iv. Ako se radi o ažuriranju ispisuje se poruka „**Korisnik pod ID-om je uspješno ažuriran!**“, gdje je ID broj zapisa korisnika u bazi.
3. PHP datoteku **prijava.php** šalje podatke samoj sebi i provjerava korisničke podatke. **Ispod obrasca napisati jedno valjano korisničko ime i lozinku u čitljivom obliku za svaku ulogu korisnika (admin, moderator i registrirani korisnik).**
- a. Provjerite da je skripti pristupljeno preko https protokola. Ukoliko nije ili se URL manipulacijom promijeni protokol skripta automatski preusmjerava na https.
  - b. Provjerite je li odabrana opcija „Zapamti me“. Ukoliko da i ako je dopušteno korištenje na stranici registracije kreira se kolačić za pamćenje korisničkog imena. Prilikom sljedeće prijave unaprijed se popuni! U protivnom briše se kolačić za pamćenje korisničkog imena.
  - c. Provjeriti na slanje obrasca da li već postoji korisničko ime i lozinka u bazi. Ako ne postoji sve labele i elementi promijene boju u crvenu. Koristite JavaScript funkcije koje šalju **AJAX zahtjev** Vašoj PHP skripti koja provjerava da postoji korisničko ime i lozinka u bazi i vraća odgovor u ispravnom XML formatu.
    - i. Ako postoji korisničko ime i lozinka je pogrešna, bez osvježavanja ekrana ispisuje se poruka: „**Lozinka nije ispravna. Pokušajte ponovo! Broj preostalih pokušaja je X**“. Pri čemu je X broj neuspješnih pokušaja a ukupan broj pokušaja 5. Nakon 5 neuspješnih pokušaja status korisničkog računa postane blokiran i više se ne može prijaviti.
    - ii. Ako se unesu ispravno korisničko ime i lozinka koji postoje u bazi, postavlja se broj neuspješnih pokušaja na 0 te se kreira sesija i korisnika se preusmjerava na stranicu *popis.php*.
4. PHP datoteku **popis.php** koja prikazuje 2 tablična prikaza svih podataka sa stranice *registracija.php* i zapis iz dnevnika koji su uneseni u bazu. Ovisno o prijavljenoj ulozi neki dijelovi stranice nisu vidljivi.
- a. Tablica obrazac:
    - i. Svaki redak tablice ima poveznicu id korisnika. Dodatno se administratoru prikazuje i opcija briši. Korisnici koji nisu administrator vide samo svoje podatke u tablici. Administrator vidi sve podatke korisnika. Klikom na id korisnika prebacuje se na skriptu *registracija.php* uz slanje id-a retka koji se želi ažurirati. Automatski se popune sva polja sa podacima odabranog retka.
    - ii. Moguće je sortirati podatke (uzlazno/silazno) temeljem: id korisnika, uloge korisnika i statusa korisničkog računa. Inicijalno su podaci sortirani silazno.
    - iii. Klikom na opciju obriši, briše se korisnik iz baze podatka (koristeći funkcije iz prve točke) i radi se preusmjerava na istu skriptu (*popis.php*).
  - b. Tablica dnevnik (vidljiva administratoru):
    - i. Stupci tablice dnevnika su: korisnik, uloga, apsolutna putanja skripte, datum i vremena pristupa. Podaci su grupirani po korisniku.
    - ii. Implementirati straničenje podataka (prva, prethodna, sljedeća) bez osvježavanja ekrana. Koristite JavaScript funkcije koje šalju **AJAX zahtjev** Vašoj PHP skripti koja provjerava broj stranice u bazi i vraća odgovor u ispravnom JSON formatu.

5. PHP datoteka **obrazac.php** šalje podatke iz obrasca samoj sebi kako bi dodala novi podatak u bazu podataka. Skripta provjerava ispravnost unesenih podataka (validacija na strani poslužitelja) i u slučaju greške ispisuje informaciju o grešci i elemente koje ima grešku označeno je drugom CSS osobinom. Provjerite:
- Element padajućeg izbornika u kojem se odabire redoslijed prikaza elemenata (vodoravno/okomito) da nije odabrana opcija "Odaberi". Napomena: Dodati opciju u padajući izbornik ako ne postoji od ranije. Moguće je u kolačić spremi željeni odabir (ako je dopušteno korištenje na stranici registracije). Inicijalno je odabrana opcija "Odaberi" osim ako prethodno nije već u kolačiću spremljen preferirani način prikaza.
  - Element za prijenos datoteke (**bez korištenja regularnih izraza**) da je ekstenzija (pdf, png, jpg, mp3, mp4) i da datoteka nije veća od 1 MB. Koristite JavaScript funkcije koje šalju **AJAX zahtjev** Vašoj PHP skripti koja provjerava postoji li već ta datoteka u bazi i vraća odgovor u ispravnom JSON formatu. U slučaju da datoteka postoji u bazi potrebno je promijeniti naziv datoteke. Napomena: U bazu se sprema putanja do datoteke.
  - Element za unos datuma i vremena da je u ispravnom formatu (**obavezno korištenje regularnih izraza**). Format je „dd.mm.gggg. hh:mm:ss“. Gdje su za datum „d“, „m“ i „g“ brojke od 0-9 uz iznimke da prva „d“ brojka može biti od 0-3, a prva „m“ brojka 0 ili 1. Gdje su za vrijeme „h“, „m“ i „s“ brojke od 0-9 uz iznimke da prva „h“ brojka može biti 0-2, a prva „h“ i „s“ brojka 0-5.
  - U slučaju ispravnog unosa primjenjuje se željeni raspored prikaza elemenata, zapisuje se novi podatak u bazu podataka (koristeći funkcije iz prve točke) i radi se preusmjerava na skriptu *multimedija.php*.
6. PHP datoteka **multimedija.php** prikazuje sve zapise iz skripte *obrazac.php* grupirano po ekstenziji datoteke.

### Testiranje web aplikacije

Testirajte vašu zadaću. Sva testiranja napravite na zadaći 4, no ako niste ništa napravili u zadaći 4 napravite testiranje na zadaći 3 ili zadaći 1 da dobijete bodove iz ovog dijela. **Oprez, testiranje se može smatrati napadom i možete biti blokirani sa barke. Sva testiranja radite lokalno na vašem lokalnom xampp poslužitelju!**

- Kreirajte novu stranicu plan.html u korijenski direktorij zadaće i dodajte poveznicu u navigaciju.
- Napravite plan testa, izvršite ga korištenjem alata siege ili web stress tool. Plan mora sadržavati minimalno 20 poveznica koji prolazi kroz sve vaše stranice barem jednom. Ako koristite siege priložite datoteku sa url adresama koje ste testirali i postavite poveznicu na nju. Ako ste koristili web stress tool napravite sliku ekrana u kojoj se vidi popis vaših 20 poveznica koje testirate kao na slici ispod.



- Na stranicu plan.html stavite slike ekrana sažetog izvještaja iz Siege alata ili slike sažetog izvještaja kojeg je generirao Web stress tool.
- U pregledniku Firefox odaberite mogućnost responsive design i izmjerite koliko je trajanje otvaranje svake vaše stranice sa postavkom „Good 2G“. Prije mjerenja obrišite cache.
- Kao izvještaj kreirajte stranicu testBrzine.html u korijenski direktorij zadaće i dodajte poveznicu u navigaciju. Stranica sadrži tablicu sa dvije kolone, naziv skripte i vrijeme učitavanja. Poredajte rezultate skripata onim redom kojim ste ih mjerili.
- Optimirajte početnu stranicu da se učitava maksimalno 2 sekunde sa postavkom „Good 2G“ uz prethodno očišćen cache. Čitavi sadržaj mora se učitavati sa barke. Dodajte informaciju o optimalnom vremenu učitavanja na stranicu testBrzine.html sa kratkim opisom što ste točno sve promijenili da bi postigli traženo vrijeme.

### Način predaje:

Rješenje postaviti na poslužitelj barka.foi.hr do dogovorenog roka, u mapu /var/www/WebDiP/2021/zadaca\_04/{LDAP\_korisnicko\_ime}/ i nakon prijenosa postaviti prava (-rwxrwx---) za taj direktorij (chmod 770). Prava datoteka i poddirektorija unutar tog direktorija nisu bitna. **Ne poštivanje ovih pravila smatra se ne predavanjem zadaće što rezultira s oduzimanje do 50% ostvarenih bodova.**

U direktoriju smiju biti **samo datoteke i poddirektoriji vezane uz zadaću 4 i smije imati sve iz prethodnih zadaća (1-3)**, sve ostalo treba obrisati. Zadaća nema ograničenja oko nazivlja/strukture datoteke i direktorija osim jednog. Potrebno je sve vanjske datoteke (datoteke koje niste samostalno kreirali) staviti u direktorij pod nazivom „vanjske\_biblioteke“. Izuzetak od tog pravila jest direktorij kojeg kreira sam sustav predložaka i koji mora biti u korijenskom direktoriju (npr. templates\_c u smarty predlošku). U direktoriju „vanjske\_biblioteke“ nemojte stavljati svoje skripte jer se taj direktorij neće gledati.

Napomena: Poslužitelj **barka.foi.hr** prepoznaje velika i mala slova (eng. case sensitive) kod naziva direktorija, datoteka i ekstenzija.

Ukoliko **se ne pridržava zadane strukture direktorija i nazivlja datoteka** student se kažnjava sa **oduzimanjem do 50% od ostvarenih bodova.**

Rješenju pristupate u pregledniku preko adrese:

[http://barka.foi.hr/WebDiP/2021/zadaca\\_04/{LDAP\\_korisnicko\\_ime}](http://barka.foi.hr/WebDiP/2021/zadaca_04/{LDAP_korisnicko_ime})

Sve stranice moraju ispravno prikazivati hrvatske znakove i prolaziti HTML5 validaciju - koristiti službeni W3 validator na stranici <http://validator.w3.org/> i CSS validaciju (<http://jigsaw.w3.org/css-validator/>). Nakon validacije postaviti HTML5 ikonu s linkom koji radi validaciju stranice (<http://validator.w3.org/check?uri={uri}>) i CSS ikonu koja radi validaciju stranice CSS-a (<https://jigsaw.w3.org/css-validator/validator?uri={uri}>).

**Potrebno je ispuniti anketu vezanu uz zadaću na Moodle sustavu.** Ukoliko **se ne ispuni anketa ili se ne ispuni do kraja ili sadrži pogrešne informacije** student se kažnjava sa **oduzimanjem do 50% od ostvarenih bodova**. Napomena: ovi bodovi se akumuliraju na negativne bodove o strukturi direktorija.

**Zabranjeno je prepisivanje** zadaća; ukoliko se utvrdi da je zadaća prepisana od drugog studenta **obojica dobivaju 0 bodova**.

#### Literatura:

- [www.php.net](http://www.php.net)
- <http://www.smarty.net/docs/en/>
- <http://twig.sensiolabs.org/>
- <https://github.com/ewwink/siege-windows>
- <https://linux.die.net/man/1/siege>