

Zadaća 4

Potrebno je kreirati sljedeću funkcionalnost i **izbaciti JavaScript funkcionalnost i HTML5 provjere**:

1. PHP datoteku (baza.class.php) s klasom za rad s bazom podataka koja inicijalno ima 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 datoteku (novi_proizvod.php) koja je kopija novi_proizvod.html iz zadatke 1 i šalje podatke iz obrasca sama sebi kako bi dodala novi proizvod:
 - a. Dolaskom na stranicu u kolačić se sprema vrijeme pristupa stranici ako on ne postoji. Kolačić vrijedi 10 minuta.
 - b. Provjerava ispravnost unesenih podataka (validacija na strani servera) i u slučaju greške ispisuje informaciju o grešci:
 - i. Svi podaci moraju biti uneseni i da ne sadrže sljedeće znakove: () { } ' ! # " \ /
 - ii. Provjera da li naziv proizvoda ima najmanje 5 znakova i počinje velikim slovom.
 - iii. Provjera datum proizvodnje da li je manji ili jednak od trenutnog datuma i da li je u obliku dd.mm.gggg. Napomena: Promijenite tip datuma u tekstualno polje u HTML dokumentu.
 - iv. Provjerava da li je označena barem jedna kategorija proizvoda kod slanja obrasca.
 - v. Klikom na slanje provjerava da li je prošlo 5 minuta od trenutka posjeta stranice, ako jest javlja se greška i sva polja se isključuju (engl. disable) uključujući i gumb za slanje i korisnik mora čekati do isteka kolačića koji se kreirao dolaskom na stranicu da bi osvježavanjem stranice dobio ponovo mogućnost dodavanja novog proizvoda.
 - vi. Svako polje koje je neispravno ima u labeli nakon naziva znak ! crvene boje, ako je ispravno znaka više nema. Uz taj znak iznad obrasca ispisuju se odgovarajuće poruke o greškama. Prilikom ponovnog pokušaja unosa „stare“ poruke o greškama treba maknuti kako se ne bi duplicirale.
 - c. U slučaju ispravnog unosa:
 - i. Zapisuje novi proizvod u bazu podataka (koristeći funkcije iz prve točke).
 - ii. Ispisuje vrijednosti zapisane u kolačiću na ekran te briše kolačić.
 - iii. Zapisuje u dnevnik informaciju da je ispravno unesen proizvod, datum i vrijeme unosa kao i korisnika koji je to unio.

3. PHP datoteku (registracija.php) koja je kopija registracija.html iz zadatke 1 sa dodatnim elementom (element da li želi koristiti prijavu u dva koraka) i šalje podatke iz obrasca sama sebi kako bi dodala novog korisnika:
 - a. Provjerava ispravnost unesenih podataka (validacija na strani servera) i u slučaju greške ispisuje informaciju o grešci:
 - i. Svi podaci moraju biti uneseni i da ne sadrže sljedeće znakove: () { } ' ! # " \ /
 - ii. Provjeriti dali lozinka sadrži barem dva velika slova, dva mala slova, jedan broj i duljine je od 5 do 15 znakova korištenjem regularnih izraza.
 - iii. Provjeriti da li su lozinka i potvrda lozinke isti.
 - iv. Email mora biti oblika (nesto@nesto.nesto) - provjeriti korištenjem regularnih izraza.
 - v. Email adresa i korisničko ime ne smije biti zauzeto (da već postoji korisnik s istim korisničkim imenom ili emailom).
 - vi. Obrazac za registraciju mora imati captcha sustav (preporuka [reCaptcha](#)).
 - b. U slučaju ispravnog unosa:
 - i. Zapisuje novog korisnika u bazu podataka (koristeći funkcije iz prve točke) s time da lozinka mora biti zapisana u čitljivom i kriptiranom obliku (korištenje SHA1 algoritma i soli).
 - ii. Generira se nasumični aktivacijski kod te a temelju trenutnog datuma i vremena i korisničkog imena (svaki kod se smije generirati samo jednom te je potrebno pamtiti u bazi već generirane kodove) te šalje poruku sa linkom za aktivaciju korisničkog računa na email adresu unesenu u obrazac za registraciju. Također ispisuje link na ekran (isključivo radi lakšeg testiranja zadatke).
4. PHP datoteku (aktivacija.php) koja aktivira korisnički račun pomoću gore navedenog linka te pritom:
 - a. Provjerava aktivacijski link i aktivira korisnički račun u bazi i preusmjerava na skriptu pod 5.
5. PHP datoteku (prijava.php) koja je kopija prijava.html iz zadatke 1 i šalje podatke obrasca sama sebi kako bi provjerila da li postoji aktivirani korisnik s tim podacima. **Ispod obrasca napisati jedno valjano korisničko ime i lozinku u čitljivom obliku.** Prijava se sastoj od jednog ili dva koraka (ovisno o odabiru prilikom registracije). Prvi korak prijave je unos korisničkog imena i lozinke. U slučaju tri neuspješna unosa (za redom), korisniku se zaključava pristup sustavu (u bazu podataka se zapiše da je on zaključan); u tom slučaju se aktiviranje korisničkog računa obavlja putem skripte opisane pod 6. Ako je prvi korak ispravan i ako je uključena prijava u dva koraka slijedi drugi korak gdje se korisniku na email šalje jedinstveni kod prijave koji se također ispiše i na ekranu (isključivo radi lakšeg testiranja zadatke). Kada korisnik unese ispravan kod ispisuje se poruka „Prijavljeni ste!“.
6. PHP datoteku (otklucavanje_korisnika.php) koja ispisuje sve zaključane korisnike sa mogućnošću da se isti otključa nakon odabira jednog po jednog korisnika (realizirano u ovoj skripti).
7. PHP datoteka (dnevnik.php) ispisuje sve podatke iz dnevnika sortirano po datumu i vremenu.

Svaki ispis na ekran mora biti formatiran prema strukturi i CSS uputama definiranim u zadatci 1.

Način predaje:

Rješenje postaviti na poslužitelj barka.foi.hr do dogovorenog roka, u mapu /var/www/WebDiP/2016/zadaca_04/{LDAP_korisnicko_ime}/ i nakon postavljanja dodijeliti prava (-rwxrwx---) za direktorij (**chmod 770**). **Ne poštivanje ovih pravila smatra se nepredavanjem zadatka što rezultira sa 0 bodova.**

U direktoriju smiju biti **samo datoteke i poddirektoriji vezane uz zadatak 4**, sve ostalo treba obrisati što je iz prošlih zadataka.

Zadaci smije sadržavati datoteke: *baza.class.php, novi_proizvod.php, registracija.php, prijava.php, otključavanje_korisnika.php, aktivacija.php, dnevnik.php, recaptcha direktorij (smještaju se sve skripte skinute za rad reCaptcha sustava), {korisnicko_ime}.css, slike se nalaze unutar poddirektorija /slike, datoteke vizualnog stila se nalaze u poddirektoriju /css unutar glavnog rješenja (polazne mape).*

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

Rješenju pristupate preko adrese:

http://barka.foi.hr/WebDiP/2016/zadaca_04/{LDAP_korisnicko_ime} pomoću Web preglednika. 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 sa linkom koji radi validaciju stranice (<http://validator.w3.org/check?uri={uri}>) i CSS ikonu koja radi validaciju stranice CSS-a.

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

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

Literatura:

- www.php.net