„**IT AKADEMIJA“ SARAJEVO**

**SMJER : PHP**

**Profesor : gosp.Amar Bešlija**

**Šabović Amel**

**APLIKACIJA „FIRMA“**

**ZAVRŠNI RAD**

**SARAJEVO 2019**

**UVOD**

Aplikacija „Firma“ služi da bi zaposlenici određene firme ili kompanije izrađivali dokumente potrebne u poslovanju (prijem robe, računi, naruđbe) , imali uvid u stanje artikala na zalihi, unijeli nove artikle te upravljali podacima u ostalim tablicama. Svi podaci koje zaposleni unose udmah se ažuriraju i na raspolaganju su na uvid ostalim zaposlenim jer svi koriste istu bazu podataka.

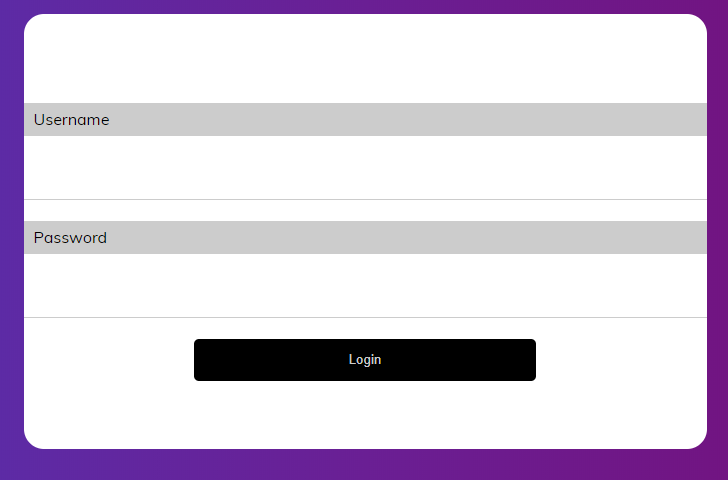
Za pristup ovoj aplikaciji može da se koristi bilo koja vrsta računara, samo je bitno da ima pristup internetu.

Za izradu ove aplikacije koristio sam PHP programski jezik i na njemu se zasniva većina koda ove aplikacije. Pored PHP-a koristio sam još Javascript, SQL te baza podataka koja je rađena u Phpmyadmin-u.

Korišten je PHP objektno orijentirani programski jezik sa PDO (PHP data object) konekcijom koja služi za lahak i jednostavan pristup bazama. Jedna od najboljih pogodnosti kod PDO je try/catch blok koji je odličan metod za upravljanje greškama i takođe je korišten.

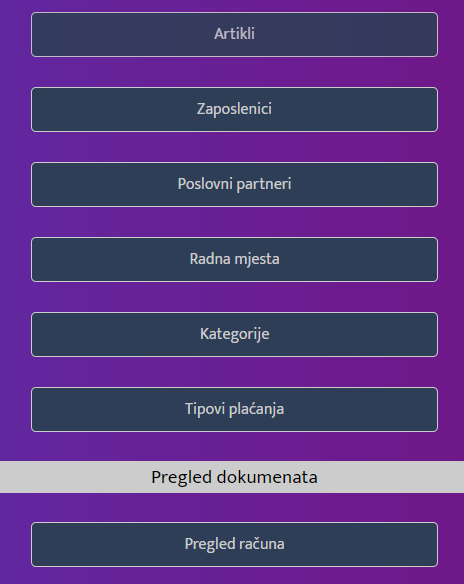
Aplikacija sadrži dvije vrste korisnika : **administrator i zaposlenik**. Jedina razlika je u tome što administrator ima pristup svim tabelama unutar aplikacije i bazi podataka dok zaposlenik ima limitiran pristup sa dozvolom na pregled naruđbi, pregled primki, pregled računa, unos računa i unos primke robe.

Nakon što se aplikacija pokrene otvara se forma za prijavu koja zahtjeva korisnički username i password. Nakon što se unese ispravan i username i password te pritisne na tipku login šalje se zahtjev u bazu podataka u tablicu zaposleni. Ukoliko je unesen pogrešan username ili pogrešan password neće se moći ulogovati u aplikaciju.



-forma za prijavu-

Ako se korisnik aplikacije nalazi u tablici „zaposlenik“ koja se nalazi u bazi podataka otvara se početna forma aplikacije koja je prikazana na slici ispod.



Na početnoj formi, ako ste prijavljeni kao zaposlenik vidjet ćete slijedeća dugmad koja su podijeljena u dvije grupe :

1. Pregled dokumenata

-Pregled narudzbi

-Pregled primki

-Pregled računa

2. Unos dokumenata

-Unos računa

-Unos primke

U koliko ste prijavljeni kao administrator otvara se forma sa svim dugmadima unutar aplikacije koja su podijeljena u tri grupe a sadrži slijedeće :

1. Pregled, unos i izmjena podataka

-Artikli

-Zaposlenici

-Poslovni partneri

-Radna mjesta

-Kategorije

-Tipovi plaćanja

1. Pregled dokumenata

-Pregled računa

-Pregled narudzbi

-Pregled primki

1. Unos dokumenata

-Dodaj zaposlenika

-Narudzbe

-Dodaj artikal

-Dodaj partnera

-Dodaj novo radno mjesto

-Dodaj novu kategoriju

-Dodaj novi tip plaćanja

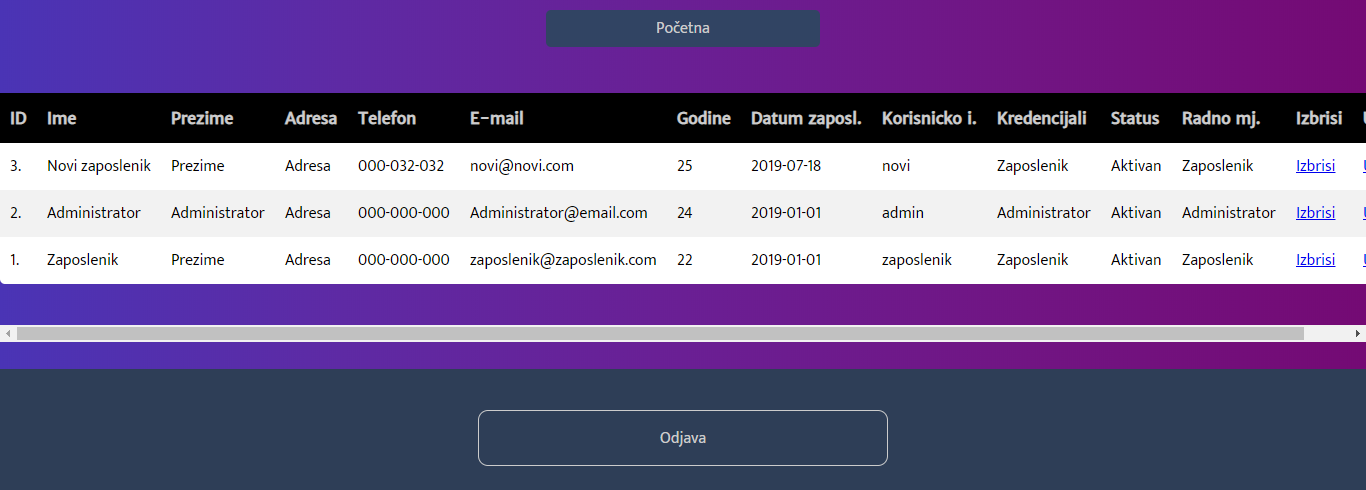
**ARTIKLI**

Svrha ove forme jeste da korisnik vidi stanje svih artikala unutar baze sa nazivom svakog artikla, kojoj kategoriji artikl pripada, cijenom i zalihom. Takođe korisnik moze da uredi svaki artikl te ga izbriše.

Ulaskom u samu formu automatski se šalje upit u bazu te se izlistavaju svi artikli.

**ZAPOSLENICI**

Ova forma služi za prikaz svih zaposlenih unutar firme te ona sadrži : ime zaposlenika, prezime, adresu , telefon , e-mail, godine zaposlenika , datum zaposlenja , korisničko ime, kredencijal , status , radno mjesto i takođe nudi administratoru da uredi podatke za svakog zaposlenika ili da ga izbriše.



**POSLOVNI PARTNERI**

Forma koja služi za listu svih poslovnih partnera sa kojima firma sarađuje te njihove podatke : naziv poslovnog partnera , adresa , telefon , telefax , e-mail , od kada je partner i koji je status firme.

**RADNA MJESTA**

Radno mjesto se koristi vezano za tablicu zaposlenika , označava funkciju koju zaposlenik obavlja u firmi .Sadrži podatke o nazivu i opisu radnog mjesta.

**KATEGORIJE**

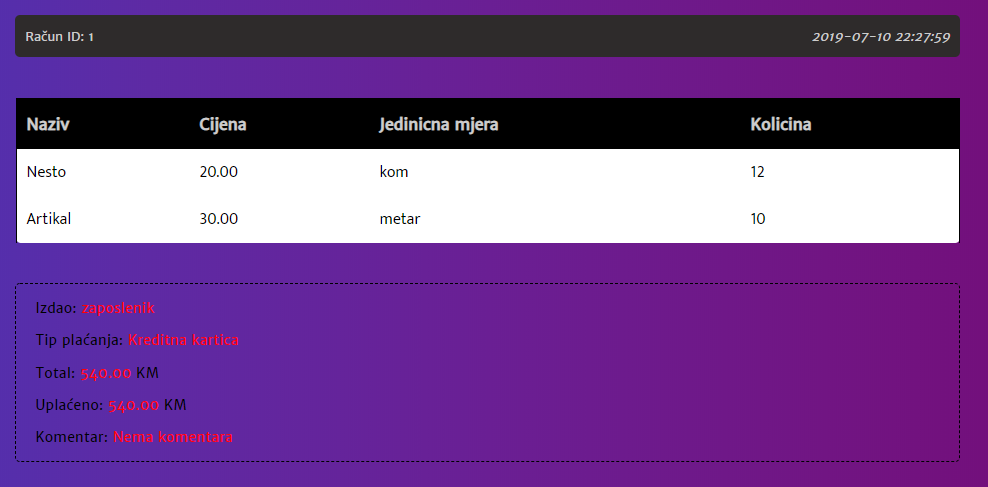
Služi za listu svih kategorija artikala na stanju sa nazivima kategorija te nudi mogućnost uređivanja ili brisanja kategorije.

**TIPOVI PLAĆANJA**

Sadrži tipove plaćanja koje koristimo kod unosa plaćanja .

**PREGLED RAČUNA**

Pokretanjem ove forme šalje se upit bazi za izlistavanjem svih računa sa id-om svakog računa, artiklima na računu , datumom i vremenom kada je račun napravljen i imenom zaposlenika koji je račun napravio , tipom plaćanja , koliko je uplaćeno i komentarom u koliko je potrebno.



**PREGLED NARUDZBI**

Na ovoj formi možemo vidjeti podatke koji su vezani uz dokument narudzbu , poslovnog partnera od kojeg smo odredjeni artikl naručili , naziv artikla , količinu , da li je artikl primljen i datum prijema.

**PREGLED PRIMKI**

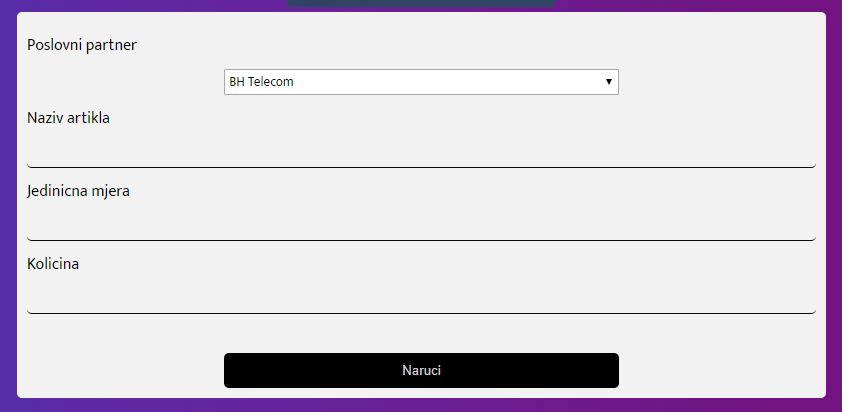
Kao i u ostalim formama pregleda u ovoj formi se prikazuju podaci iz narudzbe sa id-om naruđbe , poslovnim partnerom , nazivom artikla i količinom.

**DODAJ ZAPOSLENIKA**

Pokretanjem ove forme administratoru se nudi da dodaje novog zaposlenika sa njegovim osnovnim podacima te korisničkim imenom i šifrom.Takođe može da dodavanjem novog radnika istog postaviti i kao administratora.

**NARUDZBE**

Forma koja služi za izradu dokumenta naruđba. Ona se šalje dobavljaču kako bi nam dostavio novu robu u skladište , a podaci se spremaju u bazu .Pokretanjem ove forme otvara nam se prozor sa imenom poslovnog partnera od kojeg dobavljamo robu , nazivom artikal kojeg naručujemo , jediničnom mjerom i količinom željene robe.



**DODAJ ARTIKL**

Pritiskom na ovo dugme unutar aplikacije administratoru se nudi dodavanje novog artikla sa nazivom istog , opisom , kategorijom , cijenom , jediničnom mjerom i zalihom tog artikla.

**DODAJ NOVO RADNO MJESTO**

Služi za dodavanje novog radnog mjesta sa njegovim nazivom i opisom.

**DODAJ NOVU KATEGORIJU**

Kao što samo ime kaže ovdje se dodaju nove kategorije artikala.

**BAZA PODATAKA**

Mysql je jedan od najraširenijih relacijskih sistema otvorenog koda za upravljanje bazom podataka. Centralna je komponenta xampp paketa. Pokretanjem Mysql- a unutar xampp control panela moguće je pristupiti bazi podataka.

Baza podataka unutar aplikacije „Firma“ sastoji se od 12 tabela i atributima koje ću objasniti u nastavku :

-firma\_artikl (artikl\_id, artikl\_naziv, artikl\_opis, artikl\_kategorija\_id, cijena , jedinicna\_mjera, zaliha)

-firma\_kategorija (kategorija\_id , kategorija\_naziv)

-firma\_narudzba (narudzba\_id, narudzba\_poslovni\_partner , naziv\_artikla , jedinicna\_mjera, kolicina, narudzba\_status\_id , datum)

-firma\_narudzba-primka (primka\_id, narudzba\_id , kolicina , datum)

-firma-poslovni\_partner (partner\_id , naziv, adresa , telefon , telefax , email , partner\_od , partner\_status\_id)

-firma\_radno\_mjesto ( mjesto\_id, mjesto\_ime, mjesto\_opis)

-firma\_status ( status\_id , status\_ime)

-firma\_tip\_placanja ( placanje\_id , placanje\_naziv)

-kredencijal ( kredencijal\_id , kredencijal\_ime)

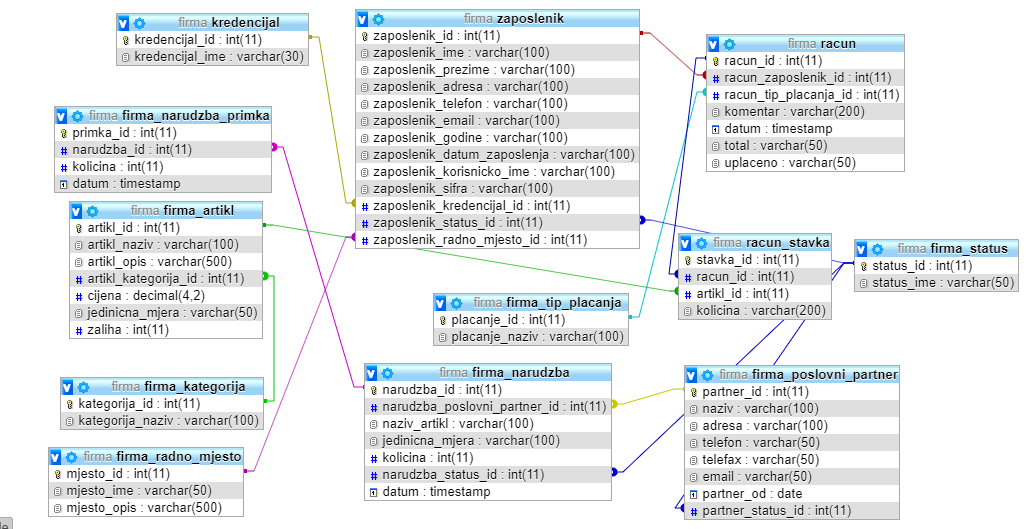
-racun (racun\_id, racun\_zaposlenik\_id, racun\_tip\_placanja\_id , komentar, datum, total , uplaceno)

-racun\_stavka (stavka\_id , racun\_id , artikl\_id , kolicina)

-zaposlenik (zaposlenik\_id, zaposlenik\_ime , zaposlenik\_prezime, zaposlenik\_adresa, zaposlenik\_telefon, zaposlenik\_godine , zaposlenik\_datum\_zaposlenja, zaposlenik\_korisnicko\_ime, zaposlenik\_sifra, zaposlenik\_kredencijal\_id, zaposlenik\_status\_id, zaposlenik\_radno\_mjesto\_id)

Jedan od najraširenijih sistema za upravljanje relacionim bazama podataka je MYSQL , sistem otvorenog koda koji se pokreće na serveru i koji je u mojoj aplikaciji korišten. On takođe pruža pristup većem broju korisnika što je isto tako izuzetno važno.

Tabele su međusobno povezane što nam omogućava lakši ispis podataka iz baze pomoću SQL joina.



-model baze podataka „Firma“-

**IMPLEMENTACIJA APLIKACIJE**

Sama aplikacija je rađena uz instalirane programske pakete Visual Studio Code , Apache HTTP server koji je otvoreni web server i najkorišteniji server na internetu.

Instalacijom i pokretanjem Apache web servera automatski je omogućen i pristup bazi podataka te je moguć rad sa PHP programskim jezikom.Sam PHP je jedan od najpopularnijih open source skriptnih jezika za web development. Razlikuje se od drugih programskih jezika jer se izvršava na serveru za razliku od npr Javascripta koji se izvršava u browseru. Jezik koji je pogodan za početnike , brzo se uči ali isto tako pogodan i za iskusne programere. Može se koristiti za prikupljanje podataka iz obrazaca, dinamičko generiranje sadržaja stranice ili za slanje ili primanje cookiea.

Jedna od najjačih i najznačajnijih mogućnosti PHP jeste njegova podrška za različite baze podataka. Jednostavno je nadograditi web aplikaciju korištenjem specifičnih ekstenzija poput Mysqli , PDO koji sam ja u svojoj aplikaciji koristio ili neko drugo spajanje(ODBC).

Pored svega navedenog veoma je važno što se PHP može koristiti na svim operativnim sistemima od Linuxa, Unixa do Microsoft Windowsa i Mac-a.

Prvi korak u implementaciji projekta , nakon što se napravi baza jeste sama konekcija na bazu . Konekcija je rađena kroz PDO .

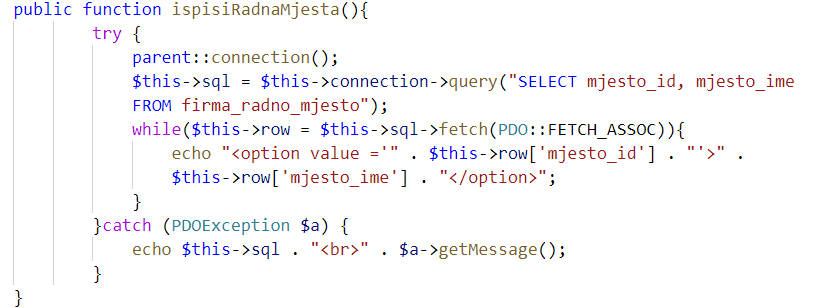
Veza se uspostavlja stvaranjem instance PDO bazne klase. Sama klasa u sebi sadrži parametre za određivanje izvora baze podataka, hosta na kojem je implementirana aplikacija , username i passworda .



-primjer metode unutar klase za konekciju na bazu podataka-

Takođe u metodi unutar klase za konekciju postavljen je charset za bazu sa UTF-8 kolacijom(Unicode Transformation Format 8 ).

Klasa koju sam nazvao DatabaseClass sadrži sve metode potrebne za ispis podataka iz baze podataka. Navest cu nekoliko primjera metoda iz klase sa kodom :



Metoda koja ispisuje podatke iz tabele u bazi pod imenom „Radna mjesta“ .Kroz try blok poziva se konekcija na bazu i zatim putem SQL query selektuju kolone koje želimo da dobavimo . While petlja je korištena za ispis podataka . Broj ponavljanja kod while petlje nije unaprijed poznat , nego zavisi od ispunjenja zadanog uslova. Kod ove vrste petlje uvjet se ispituje na početku , a blok naredbi petlje ponavlja se dok se uvjet ne ispuni. Može se dogoditi da se uvjet iz petlje i ne izvrši niti jednom .

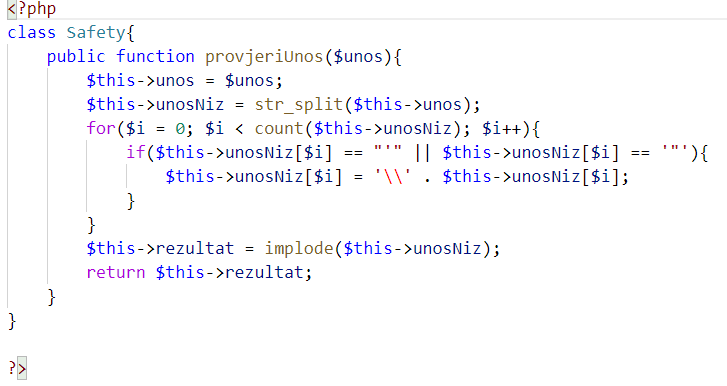
Nakon što je „pokušano“ dobaviti podatke iz baze , ako je uslov ispunjen kroz catch blok se hvataju ključne riječi koje su nabrojane u query-u .

Primjer while petlje za dobavljanje podataka iz tabele „Partneri“ :



Sama DataBaseClass-a ima 45 metoda koje rade sa bazom podataka .

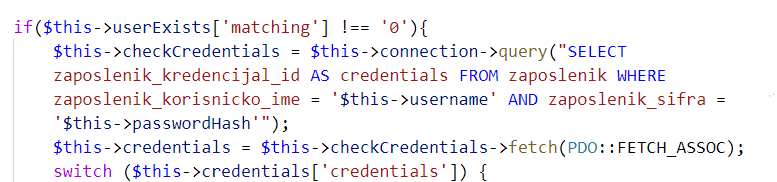
Safety Class je klasa koja radi provjeru unosu specijalnih znakova u formu. Metoda provjeri unos i konvertuje string u niz putem PHP ugrađene funkcije str\_split().



Pored ove dvije navedene klase postoji jos jedna klasa nazvana UserClass koja vrši provjeru da li postoji korisnik u bazi da bi se mogao ulogovati. Metoda prima dva parametra username i password koji je hashovan u bazi.



Nakon što se kroz if else statement provjeri da li postoji korisnik u bazi , ako postoji provjeri se korisnikov kredencijal :



Ako je kredencijal\_id = 1 ulogovani ste kao zaposlenik i šalje vas na panel za zaposlenika a u koliko je kredencijal\_id = 2 redirektovani ste na administratorski panel.

Primjer metode koja uređuje podatke unutar baze i koja prima parametre :



Mislim da je aplikacija veoma jednostavna za korištenje , sa kodom i metoda koje same za sebe govore koja im je svrha. Veoma lahko putem PHP koda ispisuje se željeni sadržaj na stranicu i uređuje.

**PRIMJENA**

Aplikacija se može primjeniti u mnogo malih obrta i firmi koji nemaju prevelike zahtjeve. PHP je uveliko olakšao samu implementaciju. Svaka firma može da koristi ovu aplikaciju bez dodatnih troškova i sa minimalnim naporima uz visoku sigurnost podataka , brzinu i učinkovitost.