Izrada i testiranje sustava za dogovaranje zamjena na poslu

Miksad, Pavel

Undergraduate thesis / Završni rad

2018

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **Josip Juraj Strossmayer University of Osijek, Faculty of Electrical Engineering, Computer Science and Information Technology Osijek / Sveučilište Josipa Jurja Strossmayera u Osijeku, Fakultet elektrotehnike, računarstva i informacijskih tehnologija Osijek**

Permanent link / Trajna poveznica: https://urn.nsk.hr/urn:nbn:hr:200:711546

Rights / Prava: In copyright / Zaštićeno autorskim pravom.

Download date / Datum preuzimanja: **2024-05-04**

Repository / Repozitorij:

Faculty of Electrical Engineering, Computer Science and Information Technology Osijek

SVEUČILIŠTE JOSIPA JURJA STROSSMAYERA U OSIJEKU ELEKTROTEHNIČKI FAKULTET

Stručni studij

IZRADA I TESTIRANJE SUSTAVA ZA DOGOVARANJE ZAMJENA NA POSLU

Završni rad

Pavel Miksad

Osijek, 2017.

Sadržaj

1. UVOD 3
1.1. Zadatak završnog rada 3
2. PRIMIJENJENE TEHNOLOGIJE I ALATI 4
2.1. HTML 4
2.2. CSS 4
2.3. PHP 5
2.4. JavaScript 5
2.5. MySQL 6
2.6. Bootstrap 6
2.7. Sublime Text 3 6
2.8. XAMPP 7
3. REALIZACIJA SUSTAVA 8
3.1. Baza podataka 8
3.2. Dizajn sučelja i funkcionalnost aplikacije 9
3.2.1. Frontend 9
3.2.2. Backend 18
4. TESTIRANJE APLIKACIJE 21
5. ZAKLJUČAK 23
LITERATURA 24
SAŽETAK 25
ABSTRACT 26
ŽIVOTOPIS 27
PRILOZI 28

1. UVOD

Cilj ovog završnog rada je izraditi i testirati sustav za dogovaranje zamjena smjena na poslu. Sustav bi trebao omogućiti zaposlenicima pretraživanje po smjenama, dogovaranje putem poruka te prihvaćanje ili odbijanje zamjene. Sustav također sadrži i administratorski račun, koji može pregledati, uređvati i brisati sve zaposlenike, smjene i zahtjeve. Zaposlenici bi trebali moći putem tablice naći smjenu koja im odgovara u tekućem tjednu, te pokrenuti zahtjev za zamjenu. Pri pokretanju zahtjeva, drugi zaposlenik je obaviješten porukom, te se porukom mogu dogovoriti i potvrditi ili odbiti zamjena. Zaposlenik može tražiti druge zamjene u istom danu dok se ne potvrdi zamjena. Nakon izrade sustava, potrebno je testirati da li svi alati i sve funkcije rade ili se tokom obavaljanja nekih funkcija javljaju pogreške.

1.1. Zadatak završnog rada

Objasniti problem dogovaranja zamjena na radnom mjestu. Predložiti odgovarajuće tehnologije za razvoj aplikacije za dogovaranje zamjena. Opisati tehnologije koje će se koristiti u izradi rješenja kao i sam postupak izrade rješenja. Izrađenu aplikaciju je potrebno testirati.

2. PRIMIJENJENE TEHNOLOGIJE I ALATI

Tokom izrade ovog sustava korištene su HTML, CSS, PHP, JavaScript tehnologije te MySQL za izradu baze podataka.

2.1. HTML

HTML ili *HyperText Markup Language* je prezentacijski jezik za izradu web stranica. HTML jezikom oblikuje se sadržaj i stvaraju se hiperveze *hipertext* dokumenta [1]. HTML je korišten za izradu cijele strukture aplikacije te hiperveza koje vode na ostale dijelove aplikacije.

```
<!doctype html>
<html>
<head>
  <meta charset="utf-8">
  <title>Zamjena Smjena</title>
  <link rel="stylesheet" href="css/styles.css?v=1.0">
</head>
<body>
  <script src="js/scripts.js"></script>
</body>
</html>
```

Kod 2.1.Primjer osnovna strukture HTML koda

2.2. CSS

CSS ili *Cascading Style Sheets* je stilski jezik kojim se opisiju prezentacije dokumenta napisanog pomoću HTML jezika [2]. Korišten je za dodatno stiliziranje aplikacije koje je bilo potrebno doraditi, a da već nije navedeno u Bootstrapovom izvornom CSS-u.

```
#poruke{
	border: 1px solid #DEE2E6;
	border-radius: 10px;
	overflow-y: auto;
	overflow-x: hidden;
	height: 400px;
}
```

Kod 2.2. Odsječak koda korištenog za stiliziranje prozora koji prikazuje poruke

2.3. PHP

PHP ili PHP: Hypertext Preprocessor je programski skriptni jezik namijenjen programiranju dinamičnih web stranica te za izvršavanje skripti na serverskoj strani [3]. Korišten je za obavljanje svih matematičkih i logičkih operacija unutar aplikacije, te povezivanja, dohvaćanja i postavljanje podataka s MySQL bazom podataka.

```
$stmt=$conn->prepare("insert into poruke (id, primio, poslao, sadrzaj, datum_vrijeme, procitana) values (null, :primio, :poslao, :sadrzaj, now(), null)");
$stmt->execute(array(
    "primio"=>$_POST["prima"],
    "poslao"=>$_POST["salje"],
    "sadrzaj"=>$_POST["sadrzaj"]
));
```

Kod 2.3. Odsječak PHP koda korištenog za postavljanje poslanih poruka u bazu podataka

2.4. JavaScript

JavaScript je skriptni programski jezik koji se izvršava u web pregledniku na strani korisnika. JavaScript dopušta pravljenje interaktivnih elemenata u aplikaciji, koji se promijene ili unaprijede korištenjem nekog događaja od strane korisnika [4]. Osim već gotovih JavaScript skripti koje su korištene pomoću Bootstrapovog JavaScripta i jQuery korištene su i dvije dodatne skripte. Prva postavlja traku za pomicanje na dno u prozoru koji prikazuje poruke:

```
$('#poruke').scrollTop($('#poruke')[0].scrollHeight);
```

Kod 2.4. Odsječak koda za postavljanje trake za pomicanje

Druga pruža dodatne mogućnosti u izborniku datuma (Bootsrap Datepicker):

```
$('#date').datepicker({
        format: "dd.mm.yyyy",
        todayBtn: "linked",
        clearBtn: true,
        orientation: "bottom auto",
        autoclose: true
        });
```

Kod 2.5. Odsječak koda za izbornik datuma

2.5. MySQL

MySQL je besplatan sustav za upravljanje bazom podataka otvorenog tipa. MySQL baze su relacijskog tipa koji se pokazao kao najbolji način skladištenja i pretraživanja velikih količina podataka [5]. MySQL je korišten kao baza podataka u ovoj aplikacije, te je kao jezik korišten za kreiranje čitave baze podataka, i u kombinaciji sa PHP-om je korišten za sav pristup bazi podataka iz aplikacije.

```
create table poruke(
      id int not null primary key auto_increment,
      primio int not null,
      poslao int not null,
      sadrzaj varchar(500) not null,
      datum_vrijeme datetime not null,
      procitana bool
);
```

Kod 2.6. Odsječak koda korištenog za izradu tablice koja sprema poruke

2.6. Bootstrap

Bootstrap je najpopularnije HTML, CSS i JavaScript okruženje za izradu responzivnih web stranica. Bazira se na 12 stupaca u kojima elemente na stranici možemo dijeliti u širine željenog broja stupaca te ih tako slagati na web stranici [6]. U ovom radu korištena je Bootstrap verzija 4.

```
<div class="row">
      <div class="col-6">
            ...
      </div>
      <div class="col-6">
            ...
      </div>
</div>
```

Kod 2.7. Primjer korištenja Bootstrapovog sistema od 12 stupaca

2.7. Sublime Text 3

Sublime Text 3 je uređivač teksta za kod, označavanje i prozu pomoću kojeg je pisan čitav kod za ovu aplikaciju. Korištena je vremenski ograničena inačica uređivača.

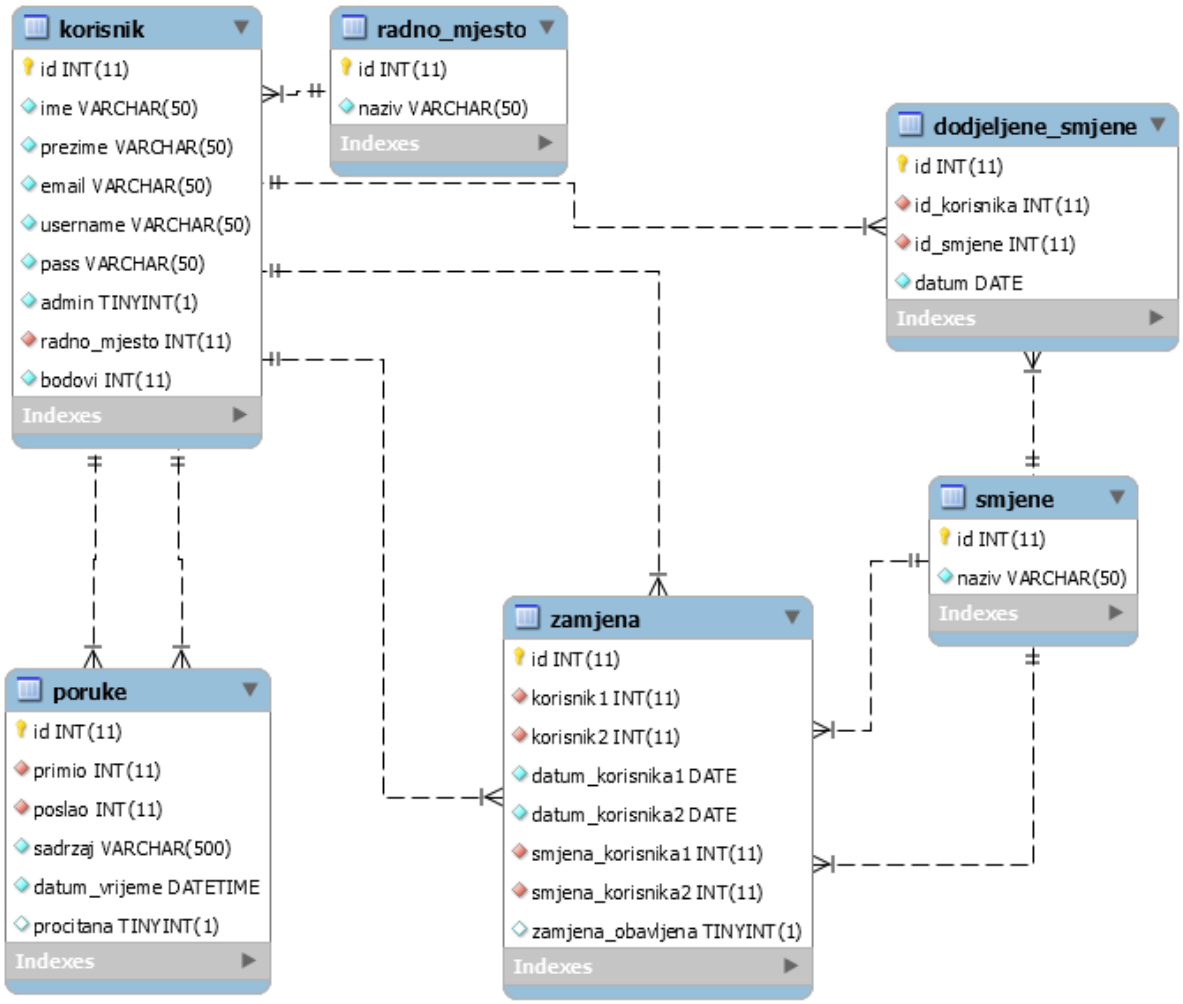
2.8. XAMPP

XAMPP je besplatni multi-platformni paket otvorenog tipa za izradu lokalnog poslužitelja kojeg su razvili Apache Friends, uglavnom se sastoji od Apache HTTP poslužitelja, MariaDB baze podataka te tumača za skripte napisane u PHP i Perl programskim jezicima [7].

3. REALIZACIJA SUSTAVA

U ovom poglavlju opisana je realizacija sustava. Navedena je struktura baze podataka, te izgled i način rada aplikacije podijeljeno u frontend i backend dio aplikacije.

3.1. Baza podataka



Slika 3.1. Baza podataka aplikacije

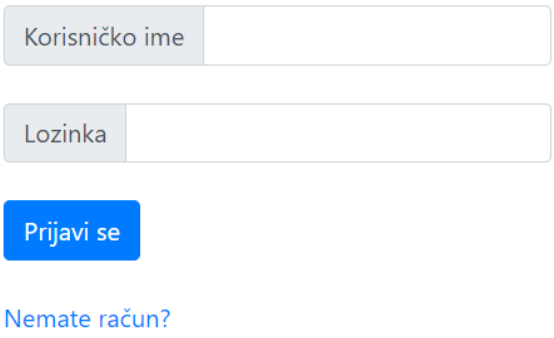
Baza podataka se sastoji od 6 tablica koje su povezane. Primarni i strani ključevi se koriste za povezivanje tablica. Strani ključevi u nekoj tablici gledaju na primarni ključ iz druge tablice. Tablica *korisnik* sprema sve potrebne informacije o pojedinim radnicima, povezuje se s tablicom *radno_mjesto* koja sprema informacije o mogućim radnim mjestima. Tablica *dodjeljene_smjene*

je također povezana s tablicom korisnik i sprema sve smjene koje su dodjeljene određenog datuma, povezuje se s tablicom *smjene* koja sprema vrste smjena. Tablica *poruke* sprema sve poruke poslane i primljene među korisnicima, povezuje se s tablicom *korisnik* kako bi se mogle vidjeti informacije o korisnicima koji su poslali/primili poruku. Tablica *zamjena* sprema sve zahtjeve upućene koje je korisnik uputio drugom korisniku za zamjenu smjene određenog datuma, povezuje se s tablicom *korisnik* kako bi se mogle vidjeti informacije o korisnicima koji sudjeluju u zamjeni i povezuje se s tablicom smjene kako bi se mogao biti dohvaćen naziv smjene.

3.2. Dizajn sučelja i funkcionalnost aplikacije

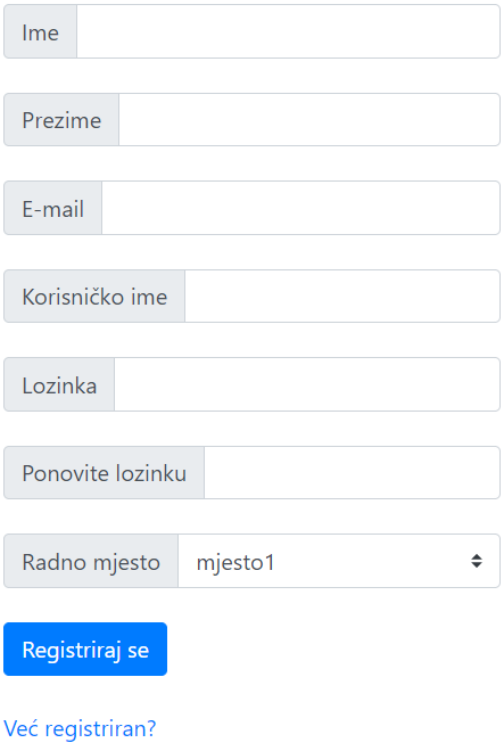
3.2.1. Frontend

Otvaranjem web aplikacije se pokreće stranica za prijavu ili registraciju korisnika.



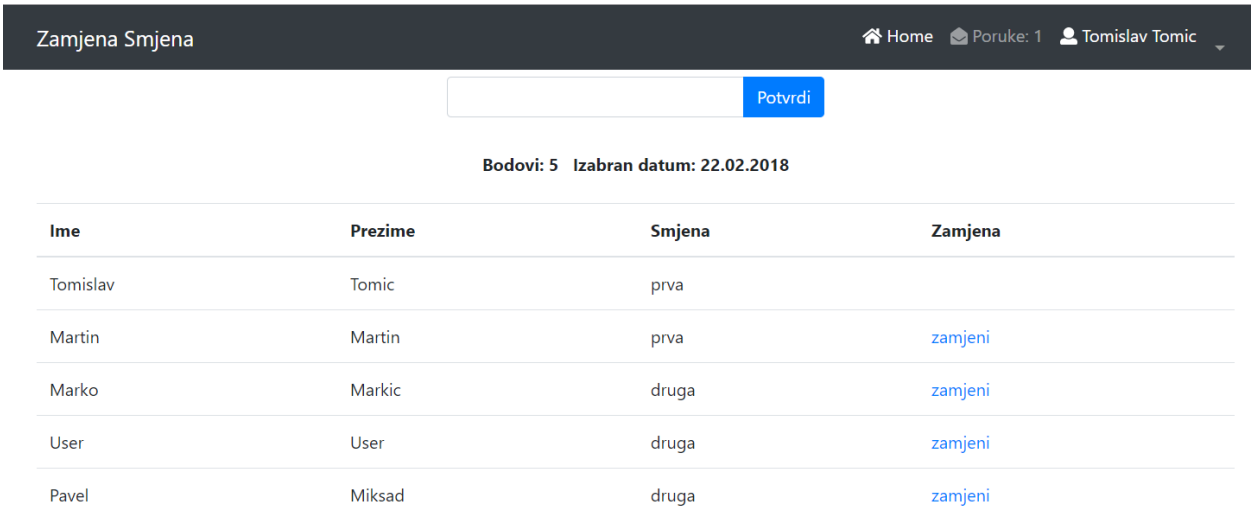
Slika 3.2. Prozor za prijavu

Na ovoj stranici unose se korisničko ime i lozinka, te pritiskom na gumb *Prijavi se* ulazi u stranicu ako njegov korisnički račun postoji u bazi podataka. Ukoliko korisnik nema korisnički račun, može se registrirati pritiskom na gumb *Registriraj se*.

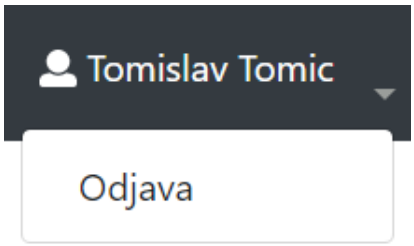


Slika 3.3. Prozor za izradu računa

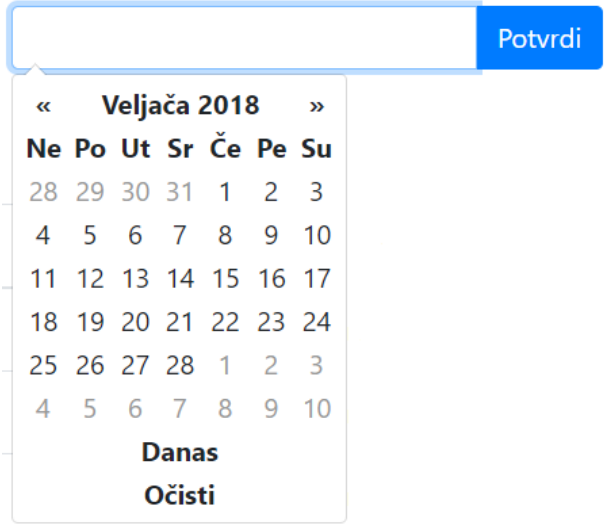
Korisnik nakon upisa svih potrebnih informacija i izbora radnog mjesta pritiskom na gumb *Registriraj se* izrađuje korisnički račun za pristup aplikaciji. Nakon toga se može izvršiti prijaviti pomoću prozora sa slike 3.2.



Slika 3.4. Glavni prozor

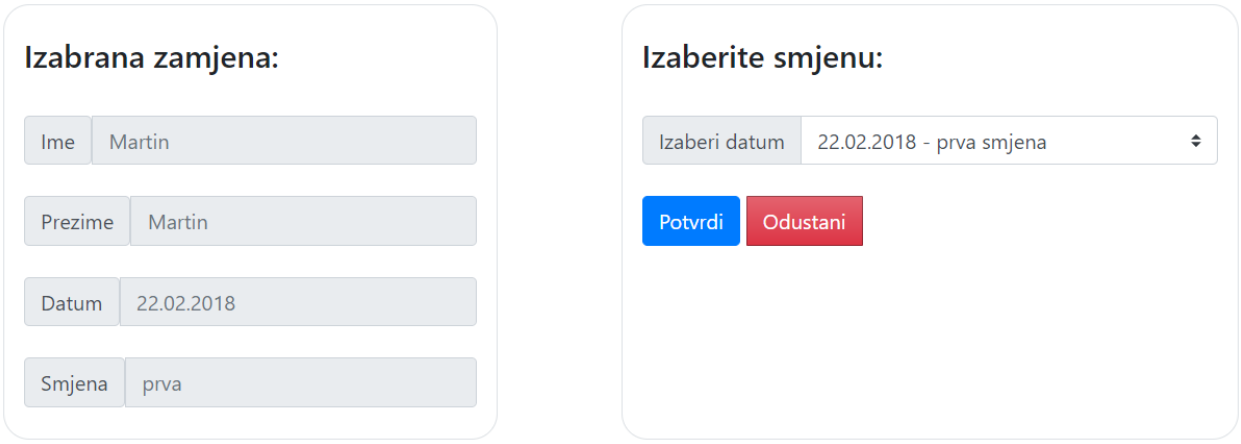


Slika 3.5. Gumb za odjavu



Slika 3.6. Izbor datuma

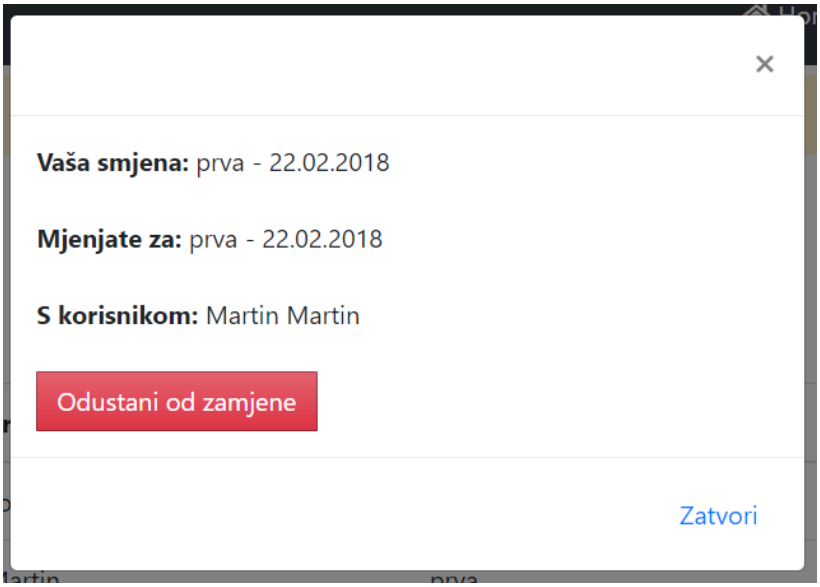
Na glavnom prozoru, idući od gore lijevo, nalazi se navigacijska traka na kojoj se nalaze gumb za povratak na ovu stranicu, gumb za otvaranje stranice s porukama, ime i prezime prijavljenog korisnika te klikom na strelicu prema dolje se pojavljuje gumb za odjavu sa aplikacije (Slika 3.2.4.). Ova stranica služi za pregled dodijeljenih smjena pojedinim korisnicima za određene datume. Pomoću izbornika za datum (Slika 3.6.) se izabire datum za koji će se dodjeljene smjene prikazivati. Veza zamjeni u određenom retku tablice otvara formu za zamjenu smjene s radnikom u tom retku.



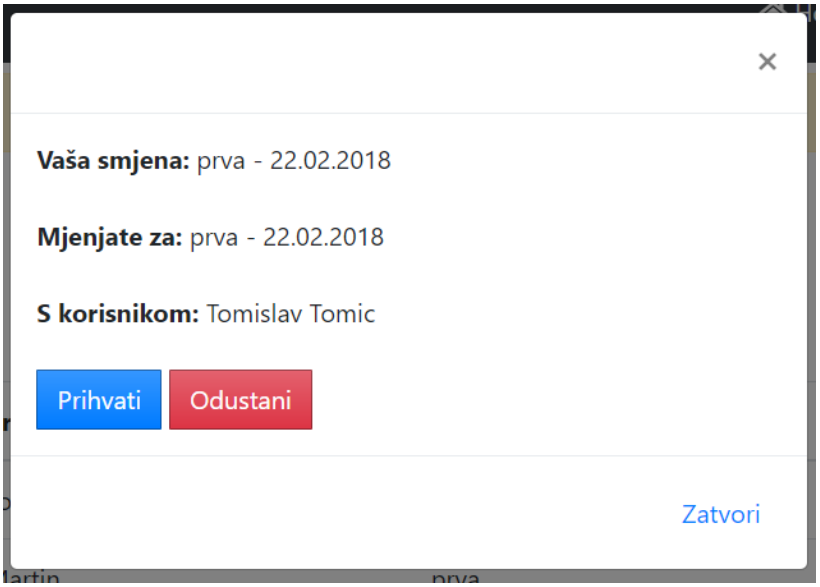
Slika 3.7. Prozor za zamjenu

U prozoru za zamjenu nalaze se ime i prezime korisnika s kojim će se izvršiti zamjena, njegova smjena za izabran datum te datum i dodijeljena smjena korisnika koji podnosi zahtjev za zamjenu.

Nakon prijavljenog zahtjeva ponuda se može i poništiti u pregledu zamjene u tijeku dok je drugi korisnik ne prihvati (Slika 3.8.).



Slika 3.8.



Slika 3.9. Prozor primljenog zahtjeva

U ovom prozoru primatelj zahtjeva vidi sve podatke o zamjeni. Može se prihvatiti zamjena gdje se onda dodaje 1 bod primatelju zahtjeva i oduzima 1 bod pošiljatelju zahtjeva, može se i odbiti zamjena.

Bodovi: 0 Izabran datum: 22.02.2018

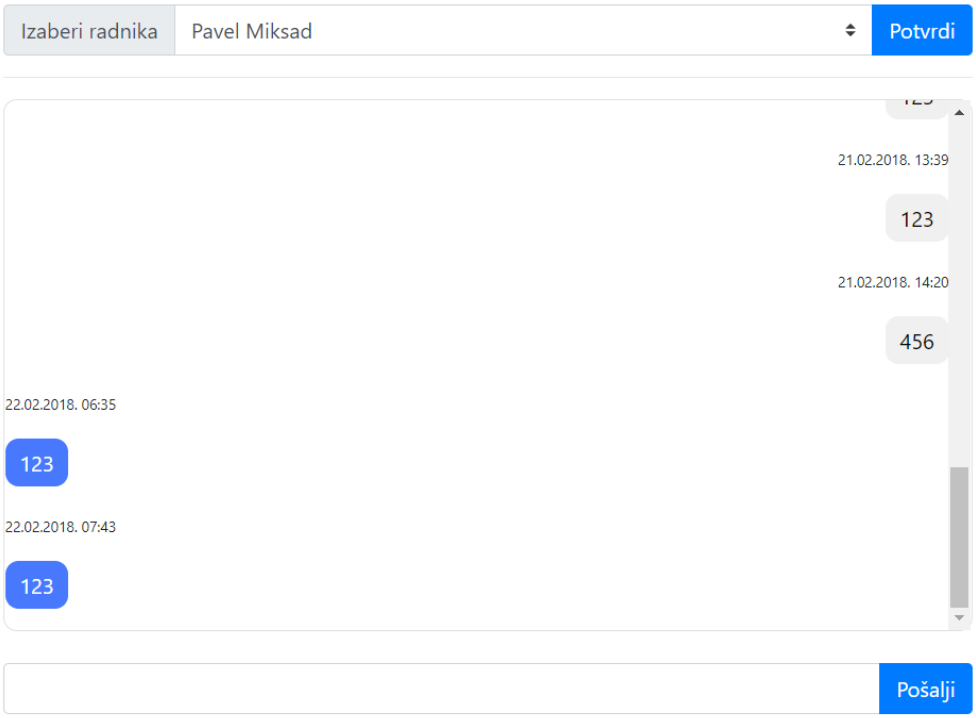
Ime	Prezime	Smjena	Zamjena
Tomislav	Tomic	prva	Nedovljno bodova za zamjenu!
Martin	Martin	prva	Nedovljno bodova za zamjenu!
Marko	Markic	druga	Nedovljno bodova za zamjenu!
User	User	druga	Nedovljno bodova za zamjenu!

Slika 3.10. Nedovoljno bodova

Ukoliko korisnik ima 0 bodova ne može se tražiti zamjena, može se samo prihvaćati zamjena. Prihvaćanjem zamjena se korisniku dodaje bod, te nakon što je korisnik opet ima više od jedan bod, može slati zahtjev za zamjenu.

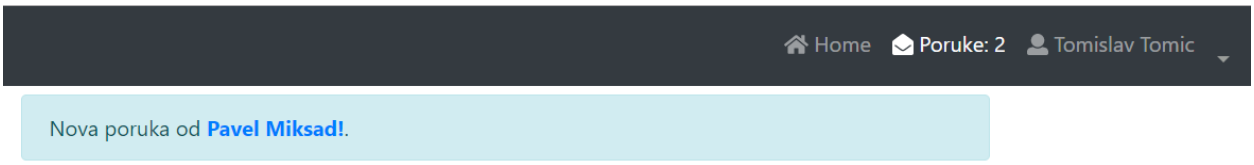
Slika 3.11. Obavjesti

Kao obavijest za nove poruke pojavljuje se broj pored gumba koji vodi do pregleda poruka, te kao obavijest za poruke pojavljuje se informacija na glavnom prozoru uz pomoć koje korisnik može pregledati zamjenu.



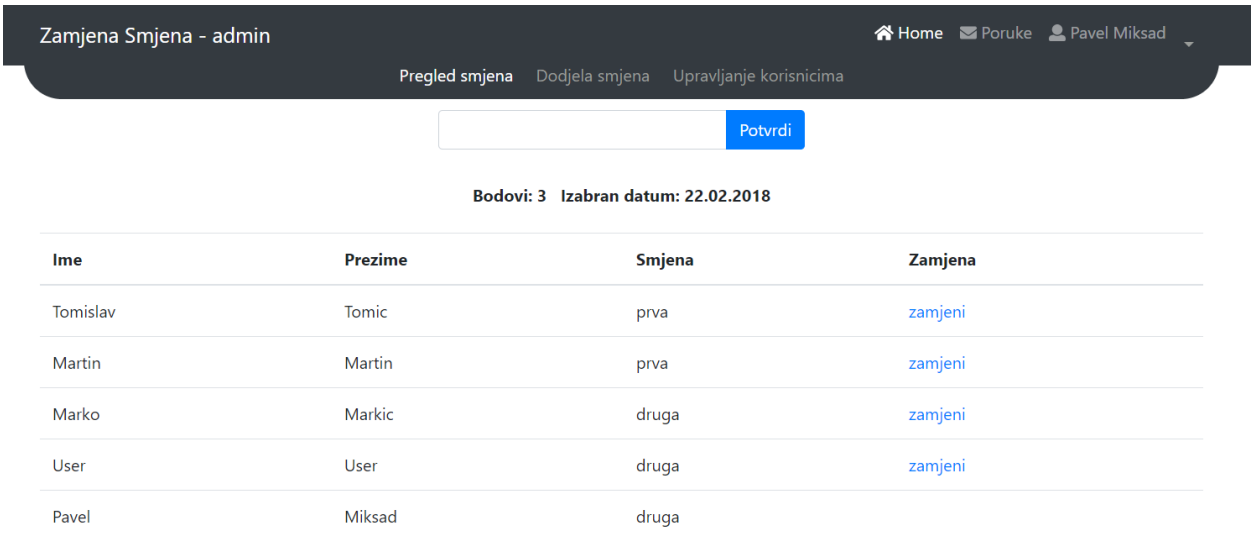
Slika 3.12. Prozor za razmjenu poruka

Iz padajućeg izbornika se odabire korisnik kojem se želi poslati poruka. Ispod izbornika nalazi se prozor za pregled poruka. U polje ispod poruka upisuje se tekst poruke za slanje, te pritiskom na gumb *Enter* na tipkovnici ili mišem na gumb *Pošalji*, šalje se poruka odabranom korisniku.



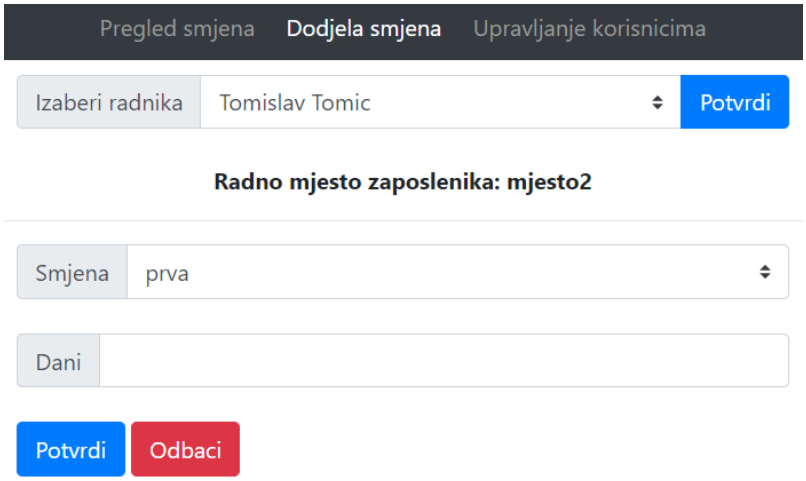
Slika 3.13. Obavijest o novoj poruci

Kada postoji nova nepročitana poruka, u prozoru s porukama će se nalaziti obavijest od koga je ta nova poruka primljena. Klikom na ime korisnika u obavijesti otvara se pregled poruka s tom osobom, te se ta poruka označava kao pročitana.



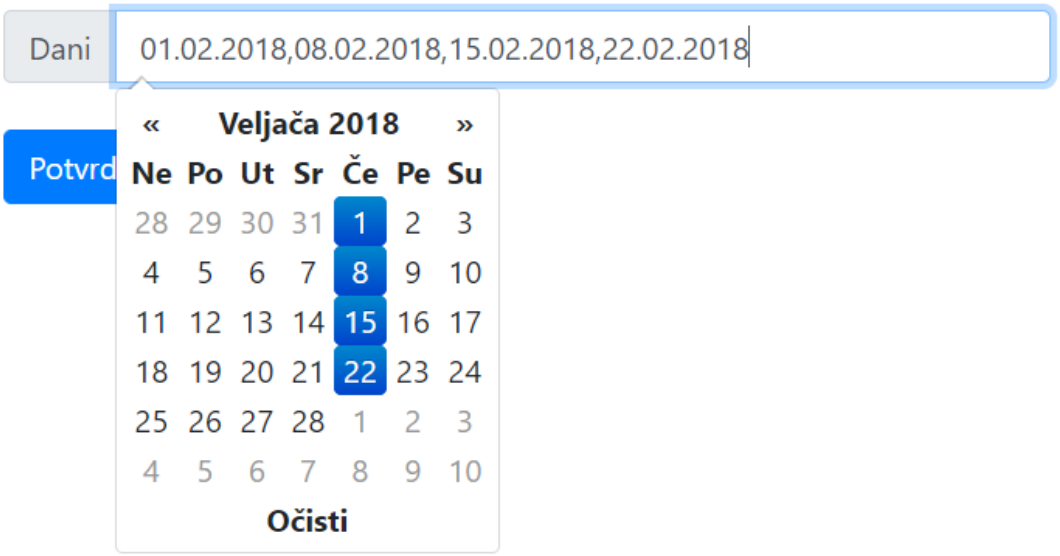
Slika 3.14. Administratorsko sučelje

Ukoliko korisnik ima status administratora nakon prijave na slici 3.2., otvara se administratorsko sučelje. Administratorsko sučelje se sastoji od tri dijela: pregled smjena, dodjela smjena i upravljanje korisnicima. Pregled smjena izgleda gotovo identično kao i sučelje za ostale korisnike gdje se može pregledavati smjene te gdje se traži zamjena smjene.

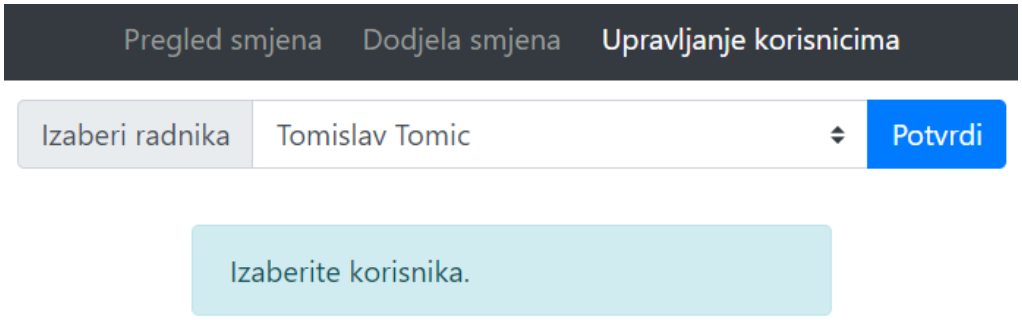


Slika 3.15. Sučelje dodjele smjena

U ovom sučelju se izabire radnik kojem se žele dodjeliti smjene. Pritiskom na gumb potvrdi prikazuju se donja polja za izbor smjene te izbor datuma kojima se želi pridjeliti izabran radnik s određenom smjenom. Moguć je izbor više datuma u jednom unosu.

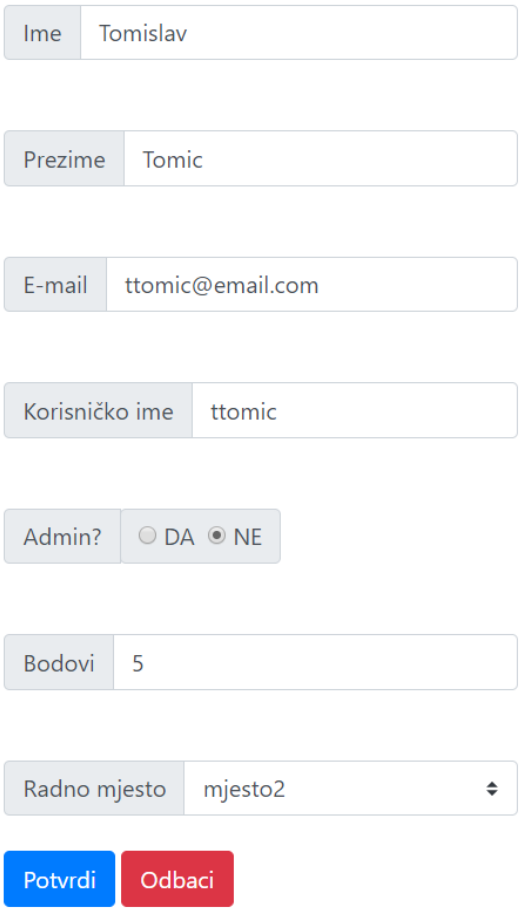


Slika 3.16. Izbor više datuma



Slika 3.17. Sučelje upravljanja korisnicima

U ovom sučelju odabire se radnik kojem se žele promijeniti neke informacije ili dodati administratorska prava. Pritiskom na gumb potvrdi, otvara se obrazac u kojoj se uređuju podatci.



Slika 3.18. Forma za uređivanje podataka

Obrazac dopušta uređivanje osobnih informacija radnika, ima li administratorska prava, broj podova koje radnik posjeduje te radno mjesto na koje je trenutno prijavljen. Pritiskom na gumb *potvrdi* izmjene se spremaju u bazu podataka, a gumb *odbaci* služi za prekidanje i odbacivanje svih izmjena.

3.2.2. Backend

3.2.2.1. Povezivanje s bazom podataka

Aplikacija ne može raditi ako nije ostvarena veza s bazom podataka. Za sve upite u bazu podataka postavljena je varijabla u jednoj datoteci, gdje se ta varijabla definira, te se onda ta datoteka pozove gdje je potrebno uspostaviti vezu s bazom podataka.

```
$conn = new PDO(
      "mysql:host=localhost;dbname=zavrsnirad",
      "root",
      "",
      array(
            PDO::ATTR_EMULATE_PREPARES=> false,
            PDO::ATTR_ERRMODE => PDO::ERRMODE_EXCEPTION,
            PDO::MYSQL_ATTR_INIT_COMMAND => "SET CHARACTER SET utf8",
            PDO::MYSQL_ATTR_INIT_COMMAND => "SET NAMES utf8"
      )
);
```

Kod 3.1. Odsječak koda za povezivanje s bazom bodataka

3.2.2.2. Upiti bazi podataka

Svaki dio aplikacije gdje se čitaju ili pišu podaci na bazu podataka obavlja upit bazi podataka. Upit je kombinacija MySQL upita i PHP koda.

```
$stmt=$conn->prepare("select * from korisnik");
$stmt->execute();
$res2 = $stmt->fetchAll(PDO::FETCH_OBJ);
```

Kod 3.2. Odsječak koda za preuzimanje podataka o korisnicima iz baze podataka

Ovaj upit čita sve podatke o korisnicima i sprema ga u varijablu *res2* kao red, koji se onda može koristiti za ispis informacija ili neke druge operacije.

3.2.2.3. Navigacijska traka

Navigacijska traka se također nalazi u zasebnoj datoteci gdje se onda poziva na svakoj stranici, kako bi se smanjilo ponavljanje istog koda. Kako bi se ostvarilo da navigacijska traka radi na svakoj stranici dodan je *if* (ako) upit kod svake stvake kako bi se moglo utvrditi na kojoj stranici se trenutno nalazi.

```
<a class="nav-link <?php
if(basename($_SERVER['PHP_SELF'])=="poruke.php"){echo 'active';} ?>"
href="poruke.php">
  <?php if($brojporuka>0){ ?>
  <i class="fas fa-envelope-open"></i> Poruke: <?php echo $brojporuka;
?></a>
<?php }else{ ?>
  <i class="fas fa-envelope"></i> Poruke</a>
<?php } ?>
```

Kod 3.3. Odsječak koda navigacijske trake

3.2.2.4. Polja za unos

Zbog velikog broja polja za unos, napravljena je funkcija u koju se unose tekst polja, naziv polja te tip, te njenim pozivom se ispisuje polje u HTML kodu.

```
<?php function inputPolje($polje,$naziv,$tip){ ?>
<div class="input-group">
      <div class="input-group-prepend">
            <span class="input-group-text" id=""><?php echo $polje;
?></span>
      </div>
      <input type="<?php echo $tip; ?>" class="form-control" name="<?php
echo $naziv; ?>"
      <?php if(isset($_POST[$naziv])): ?> value="<?php echo $_POST[$naziv]
?>"
      <?php endif; ?><?php if(isset($_POST["izabran2"])): ?> value="
      <?php echo $_POST["izabran2"]->$naziv; ?>" <?php endif;
if($tip=="password")
      { echo 'minlength="6"';} ?>  <?php if(!base-
name($_SERVER['PHP_SELF'])=="upravljanje.php")
      {  echo "required"; } ?>>
</div>
<br>
<?php }  ?>
```

Kod 3.4. Odsječak koda funkcije za polja za unos

Kako bi se funkcija pozvala u trenutku kada je potrebno polje za unos, koristi se:

```
<?php inputPolje("Korisničko ime","user","text") ?>
```

Kod 3.5. Odsječak koda za pozivanje funkcije za polja za unos

3.2.2.5. Provjera povezanog korisnika

Kako bi se zaštitila aplikacija od ne željenih ulazaka koriste se dvije provjere. Provjera je li korisnik koji pristupa stranici aplikacije prijavljen.

```
if(!isset($_SESSION["connected"])){
      header("location: ../logout.php");
}
```

Kod 3.6. Odsječak koda provjere povezan li korisnik

I provjera da povezani korisnik ima dodjeljena administratorska prava, pomoću podataka o korisniku koje su preuzete iz baze podataka pomoću koda 3.2.:

```
$user=$_SESSION["connected"][0];
if($user->admin!=1){
      header("location: ../logout.php");
}
```

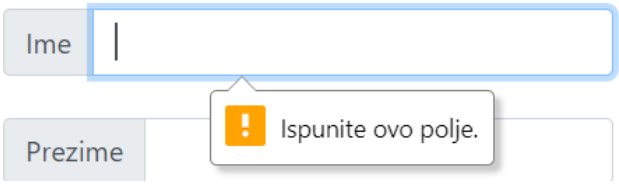
Kod 3.7. Odsječak koda provjere administratorskih prava

Ukoliko provjera završi s odgovorom „NE“, šalje korisnika na stranici *logout.php* koja korisnika šalje na prozor za prijavu.

4. TESTIRANJE APLIKACIJE

Testiranje aplikacije će se provoditi na svim poljima za unos podataka na registraciji. Budući da na obrazcu za uređivanje korisnika su korištene iste funkcije kao kod registracije, pretpostavljeno je da će test imati jednak rezultat. Obrazac za registraciju sastoji se od 7 polja za unos te jednog padajućeg izbornika.

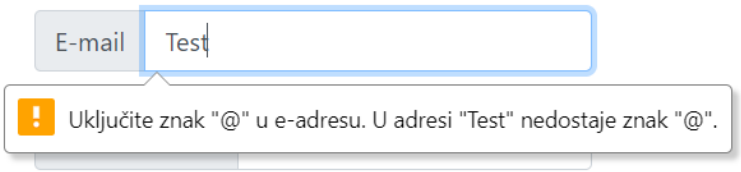
Prvim testom je provjereno jesu li sva polja moraju biti popunjena kako bi se korisnik uspješno registrirao.



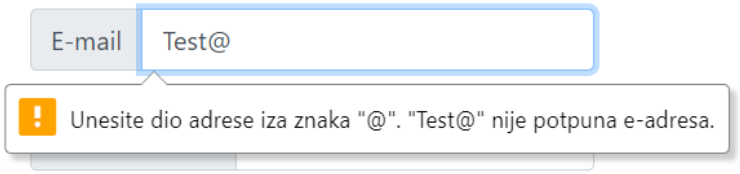
Slika 4.1. Ne popunjeno polje

Registracija se ne može ostvariti dok sva polja nisu popunjena.

Drugim testom je provjereno polje za unos e-mail adrese.



Slika 4.2. Polje za e-mail

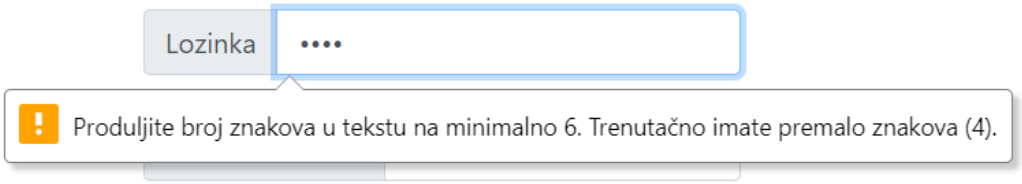


Slika 4.3. Polje za e-mail (2)

Slika 4.4. Polje za e-mail (3)

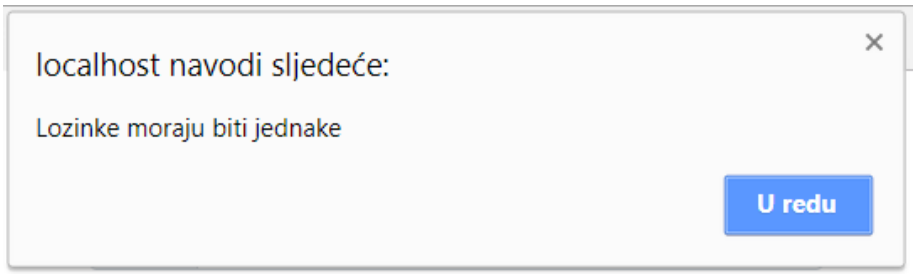
Provjera sadržava li e-mail adresa prvi dio teksta (najčešće ime i prezime), i znak @ rade, provjera drugog dijela adrese prihvaća nepotpununu adresu, bez krajeva kao što su .com, .hr i slično.

Trećim testom provjereno je sadržava li lozinka minimalno 6 znakova. Ukoliko ne sadržava, iskače obavjest koliko znakova lozinka trenutno sadrži.



Slika 4.5. Provjer duljine lozinke

Provjereno je također jednaka li je ponovljena zaporka prvoj. Ako nije izbaci se obavijest da unešene zaporke nisu jednake.



Slika 4.6. Provjera lozinke

Četvrtim testom je bilo provjereno moguće li je imati korisnika s istim korisničkim imenom ili istom e-mail adresom. Postavljena provjera za ove slučajeve nije radila.

5. ZAKLJUČAK

U ovom završnom radu nije predstavljalo napraviti problem funkcionalnost, ali za izradu dizajna bi bila potrebna neka kreativnost. S aplikacijom je moguće pregledavati smjene, slati i primati poruke, tražiti zamjene za smjene te kao administrator dodjeljivati smjene i brisati iste. Ova aplikacija bi se mogla puno bolje realizirati u nekom drugom okruženju sa kojim je moguće koristiti interaktivnost sa bazom podataka, gdje nije potrebno osvježiti web stranicu aplikacije kako bi se prikazale određene promjene u porukama, obavjestima te kako bi aplikacija mogla imati interaktivnu tablicu. Neki dijelovi aplikacije bi se također mogli doraditi kako bi imali više funkcionalnosti, npr. nije bio realiziran način za brisanje loše dodijeljenih smjena. Aplikaciji bi možda više odgovarao i sustav e-maila kao komunikaciju među zaposlenicima umjesto implementiranog sustava razgovora („chat-a"). Neke od provjera unosnih polja bi trebalo također porpaviti jer ne rade svi kako bi trebali.

LITERATURA

[1] HTML, https://hr.wikipedia.org/wiki/HTML, Lipanj 2017.

[2] CSS, https://hr.wikipedia.org/wiki/CSS, Lipanj 2017.

[3] PHP, Programiranje PHP – Rasmus Lerdorf, Kevin Tatroei Peter MacIntyre, 2015. Dobar Plan, Zagreb

[4] JavaScript, https://hr.wikipedia.org/wiki/JavaScript, Lipanj 2017.

[5] MySQL, https://hr.wikipedia.org/wiki/MySQL, Listopad 2015.

[6] Bootstrap, https://getbootstrap.com/

[7] XAMPP, https://www.apachefriends.org/

SAŽETAK

Naslov: Izrada i testiranje sustava za dogovaranje zamjena na poslu

U ovom radu navedene su sve tehnologije i alati potrebni za izradu ovakvog sustava za dogovaranje zamjena na poslu. Sustav je realiziran pomoću baze podataka MySQL a frontend i backend su realizirani pomoću HTML, CSS; PHP te JavaScript programskog jezika gdje se radi lakšeg dizajna koristilo Bootsrap okruženje. Ovakav sustav omogućuje prijavi postojećim korisnicima kao i registraciju novim. Korisnicima ova aplikacija omogućuje zamjenu smjena sa drugim radnicima i izmjenjivanje poruka sa drugim radnicima. Korisnik koji posjeduje administratorska prava unutar sustava može dodjeljivati smjene drugim radnicima, izmjeniti neke od njihovih osobnih informacija i promijeniti im radno mjesto ili da li posjeduju administratorska prava. Svi elementi koji se nalaze u aplikaciji su bili testirani da li rade kako bi trebali. Neki od elemenata nisu radili kako su bili namijenjeni ali to nije spriječilo normalan rad aplikacije.

Ključne riječi: smjena, zamjena, poruke, administracija

ABSTRACT

Title: Designing and testing a workplace shift exchange system

This thesis lists all the technologies and tools needed to create a system for workplace shift exchanges. The system is implemented using a MySQL database. Frontend and backend are realized using HTML, CSS, PHP and JavaScript programing languages where the Bootstrap environment was used for easier design. This system allows users to sign in into their existing accounts or register for new ones. Users of this application can exchange shifts and messages with other workers. A user with administrator rights within the system can assign shifts to other employees, modify some of their personal information, change their workplace or give them administrative rights. The application was tested to see if all the elements are working as they should. Some of the elements did not work as intended but that did not prevent normal application's functionality.

Keywords: shift, exchange, messages, administration

ŽIVOTOPIS

Pavel Miksad rođen je 14.04.1991. godine u Vukovaru, Hrvatskoj. 1998. godine upisuje Osnovnu školu Julija Benešića u Iloku. 2006. godine upisuje Tehničku školu Nikole Tesle smjer Tehničar za računalstvo u Vukovaru. 2010. godine maturira navedenu školu. 2013. godine upisuje sveučilišni studij na Fakultetu elektrotehnike, računarstva i informacijskih tehnologija, stručni studij smjer informatika u Osijeku. Položio je sve ispite i sada se nalazi pred obranom svoga diplomskog rada.

PRILOZI