

# AngularJS aplikacija za vođenje dnevnika rada

---

**Benke, Ivan**

**Undergraduate thesis / Završni rad**

**2016**

*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://um.nsk.hr/um:nbn:hr:200:630187>

*Rights / Prava:* [In copyright](#) / [Zaštićeno autorskim pravom.](#)

*Download date / Datum preuzimanja:* **2024-08-10**

*Repository / Repozitorij:*

[Faculty of Electrical Engineering, Computer Science and Information Technology Osijek](#)



**SVEUČILIŠTE JOSIPA JURJA STROSSMAYERA U OSIJEKU**

**ELEKTROTEHNIČKI FAKULTET OSIJEK**

**Sveučilišni studij računarstva**

**ANGULARJS APLIKACIJA ZA VOĐENJE DNEVNIKA  
RADA**

**Završni rad**

**Ivan Benke**

**Osijek, 2016.**

## SADRŽAJ:

1. UVOD .....	3
1.1. Zadatak završnog rada .....	3
2. ALATI.....	2
2.1. WAMP.....	2
2.2. HTML.....	3
2.3. AngularJS .....	4
2.4. PHP.....	4
3. STRUKTURA I DIZAJN WEB APLIKACIJE.....	6
3.1. Obrazac za prijavu .....	6
3.2. Predaja izvještaja .....	10
3.3. Pregled izvještaja.....	14
3.4. Ocjenjivanje izvještaja.....	15
3.5. Dizajn.....	19
4. ZAKLJUČAK .....	21
LITERATURA.....	22
SAŽETAK.....	23
ABSTRACT .....	24
ŽIVOTOPIS .....	25
PRILOZI.....	27

## **1. UVOD**

Zadatak završnog rada izrada je web aplikacije za vođenje dnevnika rada pomoću AngularJS skriptnog jezika. Web aplikacija podržava dva tipa korisnika: student i mentor. Student koji odrađuje stručnu praksu nakon svakog odrađenog termina predaje izvještaj u kojem se nalazi opis radnog zadatka, datum termina, tvrtka u kojoj odrađuje stručnu praksu te riješeni zadatak. Mentor dobiva na uvid taj izvještaj te može provjeriti u kojoj mjeri je student odradio zadani zadatak te ocjenjuje studenta po pojedinom zadatku. Na kraju se dobiva sveukupna ocjena iz svih ocijenjenih zadataka. Prvo će se obraditi struktura cijele web aplikacije, uključujući prijavljivanje korisnika (mentor/student), predaja izvještaja u bazu podataka kako bi ih mentori mogli ocijeniti te povratna informacija mentora u obliku komentara i ocjene na predane izvještaje. Zatim će biti objašnjen dizajn web aplikacije te koji su se sve gotovi moduli koristili u izradi i dizajniranju web aplikacije.

### **1.1. Zadatak završnog rada**

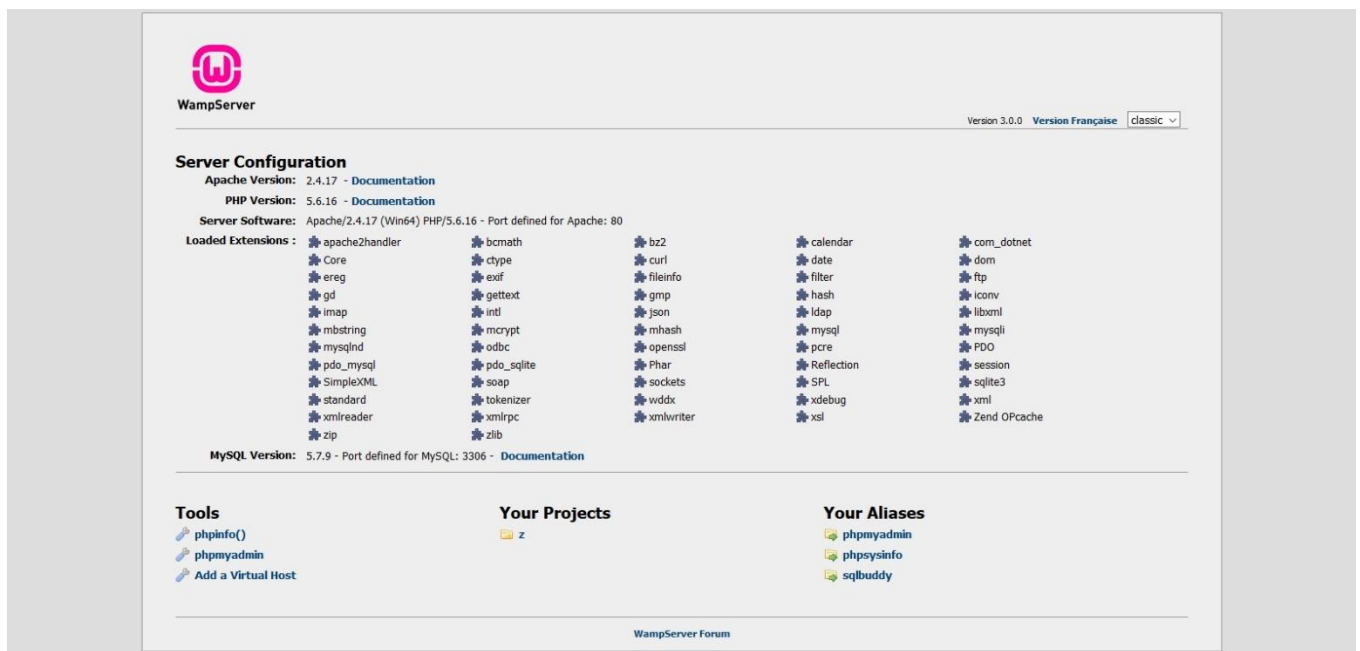
Potrebno je izraditi web aplikaciju pomoću AngularJS skriptnog jezika koja omogućava vođenje dnevnika rada. Studenti, kao i mentori, će se moći prijaviti na stranicu te će studenti moći predati izvještaje u bazu podataka, a mentori će te izvještaje moći ocijeniti i dati komentar na riješene zadatke i obrazloženje ocjene. Na kraju, izračunat će se prosječna ocjena svih riješenih zadataka koja će pomoći mentoru u određivanju konačne ocjene.

## 2. ALATI

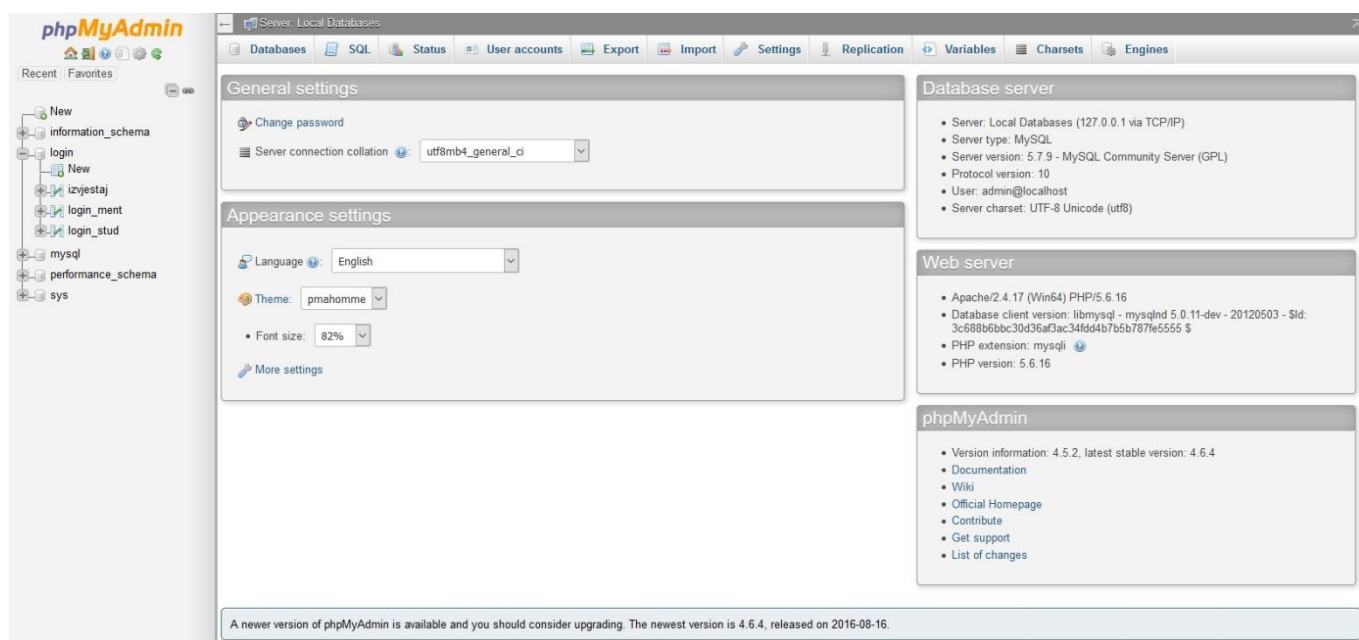
U ovom dijelu završnog rada govorit će se o tome što predstavljaju alati koji su korišteni te kako im izgleda sintaksa. Također, bit će prezentiran i način upotrebe tih alata, ali samo na osnovnoj razini. Naravno, kod korišten u stvaranju završnog rada nije na osnovnoj razini što će se vidjeti i u nastavku.

### 2.1. WAMP

WAMP, skraćeno za Windows, Apache, MySQL, PHP, program je koji se koristi za stvaranje lokalnog servera u svrhu pokretanja server-side skripti koje su potrebne za programske jezike poput PHP-a. Napravljen je za razvijanje i pokretanje dinamičkih web aplikacija, ali isključivo na Windows operacijskom sustavu što je sugerirano i u samom nazivu. Apache je program koji oponaša server na lokalnom računalu na kojem se razvija web aplikacija te se na njemu odvija pokretanje PHP skripti. Osim pokretanja server-side PHP skripti, WAMP omogućuje i lako stvaranje i upravljanje SQL bazama podataka pomoću programa phpMyAdmin. phpMyAdmin programom upravlja se MySQL poslužiteljem baza podataka koji omogućava rad s bazama podataka, ali uz odgovarajuće korisničko ime i lozinku.



#### Sl. 2.1. WAMP localhost



## Sl. 2.2. WAMP phpMyAdmin

## 2.2. HTML

HTML (engl. *Hyper Text Markup Language*) opisni je jezik za stvaranje web sadržaja. Glavno sredstvo izražavanja su oznake ili *tagovi* pomoću kojih se Internet pregledniku daje informacija o vrsti i tipu elemenata koji se koriste. Neki od HTML elemenata su: `<body>`, `<button>`, `<title>`, `<form>` i sl. Postoje dvije verzije HTML elemenata, odnosno otvarajuće i zatvarajuće oznake. Otvarajuće započinju znakom „<“, a završavaju „>“ (`<body>`), dok zatvarajuće započinju znakom „</“ te završavaju „>“ (`</body>`). Također postoje i samozatvarajući elementi kojima nije potrebna zatvarajuća oznaka (`<link ... />`). Svaki element može imati i atribut kojima se pobliže definira taj element te se tako mogu definirati ime elementa (`name=“...“`), klasa (`class=“...“`), ID (`id=“...“`) i drugo. Svi atributi se stavljaju unutar znakova „<“ i „>“ otvarajućih oznaka. HTML nije programski jezik što znači da se njime ne izvršavaju nikakvi zadaci, nego se samo opisuje sadržaj web stranice pa se tako njime mogu stvarati tablice, mijenjati boje i fontovi pojedinih elemenata, mogu se odvajati odlomci. Također se mogu odrediti stilovi, odnosno hoće li tekst biti podebljan, ukošen ili podvučen, te se mogu zadati i različiti tipovi naslova, ovisno o važnosti naslova. „Temeljna zadaća HTML jezika jest uputiti web preglednik kako prikazati hipertext dokument. Pri tome se nastoji da taj dokument izgleda jednako bez obzira o kojemu je web pregledniku, računalu i operacijskom sustavu riječ. HTML datoteke su zapravo obične tekstualne datoteke, ekstenzija im je .html ili .htm. Poveznice

unutar HTML dokumenata povezuju dokumente u uređenu hijerarhijsku strukturu i time određuju način na koji posjetitelj doživljava sadržaj stranica.“ [1]

### 2.3. AngularJS

AngularJS je open-source *client-side* i *server-side framework* koji se koristi za stvaranje web aplikacija. Većim ga dijelom održava Google i zajednica koja se bavi problemima u stvaranju *single-page* aplikacija (SPA). Zasnovan je na model – pogled – upravljač (*Model – View – Controller* (MVC)) arhitekturi. AngularJS radi na način da prvo čita HTML stranicu u kojoj su u oznake umetnuti dodatni atributi poput `ng-model=“...“`, `ng-controller=“...“`, `ng-view=“...“` i slično. AngularJS nadopunjava klasični HTML na način da predstavlja dinamički sadržaj dvosmjernim vezivanjem podataka što omogućuje automatsko osvježavanje modela i pogleda. Time se skida veliki dio radnog tereta sa servera te se pogled generira u čistom HTML-u prema podacima koji su sadržani u *scopeu* modela. \$scope usluga u Angularu očitava promjene na modelu te prema tome izmjenjuje HTML izraze u pogledu pomoću upravljača.

### 2.4. PHP

PHP je server-side skriptni programski jezik napravljen za razvijanje web aplikacija. Može ga se koristiti kao objektno-orijentirani programski jezik, ali i ne mora. Vrlo je sličan C-u i ima mnogo ugrađenih funkcija. Također, korisnici mogu praviti svoje funkcije koje im olakšavaju rad u PHP-u. Osim funkcija, postoje i petlje koje su vrlo slične onima u C-u te omogućavaju rad s poljima. Neke od petlji su *while* petlja, *do ... while* petlja, *for* petlja, *if* petlja i druge. PHP omogućuje rad s bazama podataka i Internet protokolima te omogućuje programiranje dinamičnih web stranica. Podržava sve popularnije baze podataka kao što su MySQL, Oracle, ODBC i druge. Kada se koristi PHP, klijent ne može vidjeti skriptu koja generira sadržaj kojega gleda, već vidi samo čisti HTML kod kojeg je generirala ta PHP skripta. Tako se onemogućava neovlašteni pristup datotekama i moguće rušenje web aplikacija. U početku je PHP bila kratica za *Personal Home Page*, ali je sada rekurzivni akronim PHP: *Hypertext Preprocessor*. Može se koristiti na skoro svim operativnim sustavima i platformama te je besplatan. Može ga se umetnuti unutar HTML koda, a može biti napisan i zasebno. PHP kod, neovisno o tome je li umetnut u HTML ili napisan zasebno, se piše između delimitera „<?php“ i „?>“ ili „<?“ i „?>“. Delimiteri su niz od jednog ili više znakova koji služe za odvajanje dijelova različitog značenja u tekstu, kodu i drugim podatkovnim oblicima. Prva verzija PHP programskog jezika nastala je 1995., a do danas je objavljeno 7 većih verzija sa svojim zakrpama pa je tako verzija 7.1 najnovija. „Danas je PHP jedan od najzastupljenijih programskih jezika za programiranje web

aplikacija.“

[3]



### 3. STRUKTURA I DIZAJN WEB APLIKACIJE

#### 3.1. Obrazac za prijavu

Ulaskom u web aplikaciju, korisniku se pokazuje ekran s naslovom „DNEVNIK RADA“ te gumbom „Prijava“.



*Sl. 3.1. Izgled početne stranice*

Pritiskom na gumb otvara se modal, tj. mali prozor unutar većeg početnog prozora, u kojem se nalazi obrazac za prijavu.

```
$scope.open = function () {  
    var uibModalInstance = $uibModal.open({  
        controller: 'loginCtrl',  
        templateUrl: 'login.html'  
    });  
}
```

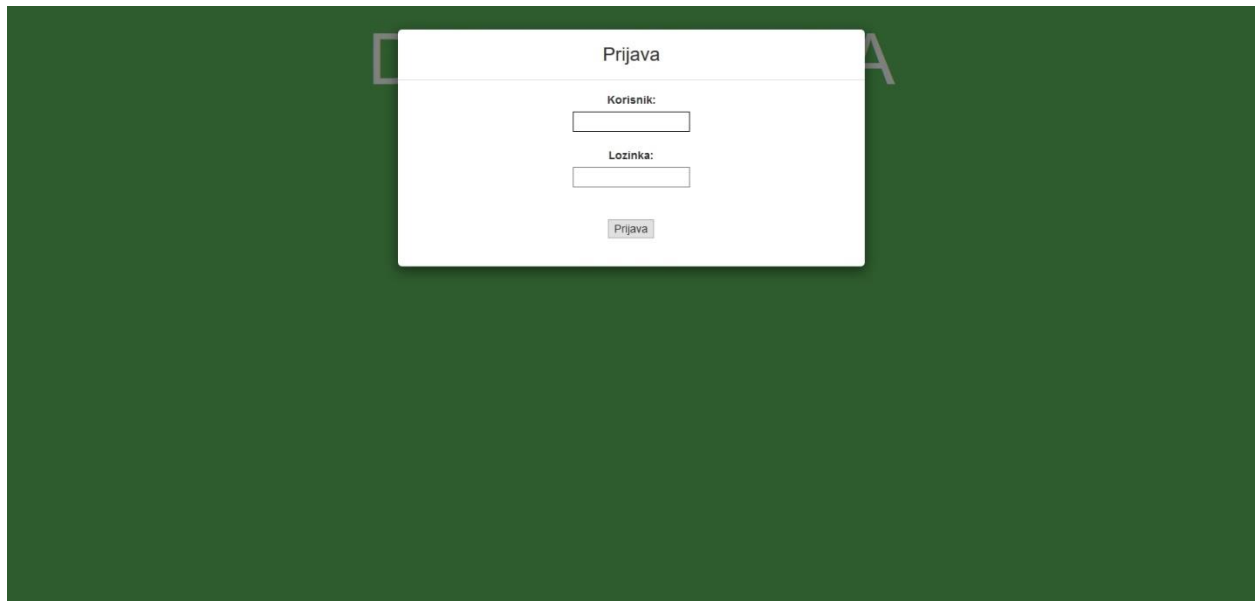
Obrazac za prijavu služi za prijavljivanje korisnika, bilo studenta, bilo mentora, u sustav. Sastoji se od dva polja, jednog za korisničko ime, drugog za lozinku te gumba za prijavu.

```
<form method="post" ng-submit="login()">  
    <label for="user">Korisnik:</label>  
    <br><input type="text" name="user" ng-model="user"><br><br>  
    <label for="password">Lozinka:</label>
```

```

<br><input type="password" name="password" ng-model="password"><br><br>
<input type="submit" value="Prijava"><br><br>
{{message}}
</form>

```



*Sl. 3.2. Izgled obrasca za prijavu*

Obrazac za prijavu napisan je u PHP-u zato što se tako onemogućava neovlašteni pristup Dnevniku rada, ali ima i dio napisan u AngularJS-u koji uzima podatke napisane u poljima za korisničko ime i lozinku te ih pritiskom na gumb „Prijava“ predaje PHP skripti koja zatim provjerava je li upisani korisnik u bazi. Klikom na gumb „Prijava“ također se sprema korisničko ime korisnika u kolačić „user“.

```

$scope.login = function () {
    var data = {
        user: $scope.user,
        password: $scope.password,
    };
    var config = {
        headers : {
            'Content-Type': 'application/x-www-form-urlencoded;charset=utf-8;'
        }
    }
    $cookies.put('user', $scope.user);
}

```

Postoje dvije tablice vezane uz prijavu korisnika unutar baze podataka, „login\_stud“ i „login\_ment“, te nije važno prijavljuje li se student ili mentor jer će prijava biti uspješna sve dok

su podaci unutar jedne od te dvije tablice. U slici ispod su prikazani korisnici sadržani u tablici „login\_stud“ te njihove lozinke, ID brojevi unutar tablice, njihova puna imena te korisnička imena njihovih mentora.

<div><div><div>←</div><div>T</div><div>→</div></div></div>						id	user	password	ime	mentor
<div><div><div><div></div></div></div><div><div><div><div></div></div><div>Edit</div></div><div><div><div><div></div></div><div>Copy</div></div><div><div><div><div></div></div><div>Delete</div></div></div></div></div></div>	1	ddankic	142536	Danko Dankić	aantic					
<div><div><div><div></div></div></div><div><div><div><div></div></div><div>Edit</div></div><div><div><div><div></div></div><div>Copy</div></div><div><div><div><div></div></div><div>Delete</div></div></div></div></div></div>	2	ffranjic	635241	Franjo Franjić	aantic					
<div><div><div><div></div></div></div><div><div><div><div></div></div><div>Edit</div></div><div><div><div><div></div></div><div>Copy</div></div><div><div><div><div></div></div><div>Delete</div></div></div></div></div></div>	3	iivic	415263	Ivan Ivić	aantic					
<div><div><div><div></div></div></div><div><div><div><div></div></div><div>Edit</div></div><div><div><div><div></div></div><div>Copy</div></div><div><div><div><div></div></div><div>Delete</div></div></div></div></div></div>	4	jjosipovic	362514	Josip Josipović	aantic					
<div><div><div><div></div></div></div><div><div><div><div></div></div><div>Edit</div></div><div><div><div><div></div></div><div>Copy</div></div><div><div><div><div></div></div><div>Delete</div></div></div></div></div></div>	5	kkarlovic	654123	Karlo Karlović	bbrankic					
<div><div><div><div></div></div></div><div><div><div><div></div></div><div>Edit</div></div><div><div><div><div></div></div><div>Copy</div></div><div><div><div><div></div></div><div>Delete</div></div></div></div></div></div>	6	llukic	321456	Luka Lukić	bbrankic					
<div><div><div><div></div></div></div><div><div><div><div></div></div><div>Edit</div></div><div><div><div><div></div></div><div>Copy</div></div><div><div><div><div></div></div><div>Delete</div></div></div></div></div></div>	7	mmarkic	32147	Marko Markić	bbrankic					

*Sl. 3.3. Tablica „login\_stud“*

U idućoj slici se nalazi sadržaj tablice „login\_ment“ u kojoj se nalaze korisnička imena mentora, ID brojevi u tablici, njihova puna imena te njihove lozinke.

					id	user	password	ime		
<input type="checkbox"/>		Edit		Copy		Delete	1	aantic	123456	Ante Antić
<input type="checkbox"/>		Edit		Copy		Delete	2	bbrankic	654321	Branko Brankić

*Sl. 3.4. Tablica „login\_ment“*

Zatim imamo PHP skriptu koja se prvo spaja s bazom podataka u kojoj su sadržane tablice „login\_stud“ i „login\_ment“. Nakon toga, skripta dobiva podatke od app.js te ih dekodira funkcijom „*json\_decode*“ kako bi dobila korisničko ime i lozinku koji su potrebni za provjeru autentičnosti. Iduće što skripta radi je provjera je li uneseno korisničko ime i lozinka dio jedne od tih dviju tablica. Ako je, app.js dokumentu se šalje niz znakova, ovisno o tome je li korisnik student ili mentor. Ako je prijavljeni korisnik student, šalje se „dashboard\_stud.php“ niz znakova, a ako je mentor, šalje se „dashboard\_ment.php“ koji se zatim kodiraju funkcijom „*json\_encode*“ kako bi ih app.js mogao pročitati.

```
$data = json_decode(file_get_contents("php://input"));
```

```

$user = $data->user;
$password = $data->password;
$q1="SELECT * FROM login_stud WHERE user='$user' and password='$password'";
$q2="SELECT * FROM login_ment WHERE user='$user' and password='$password'";
$res1 = mysqli_query($conn, $q1);
$res2 = mysqli_query($conn, $q2);
if(!$res1 || mysqli_num_rows($res1)==1) {
    $site = 'dashboard_stud.php';
    header('Content-Type: application/json');
    echo json_encode($site);
} else if (!$res2 || mysqli_num_rows($res2)==1) {
    $site = 'dashboard_ment.php';
    header('Content-Type: application/json');
    echo json_encode($site);
} else {
    $message = false;
    echo json_encode($message);
}

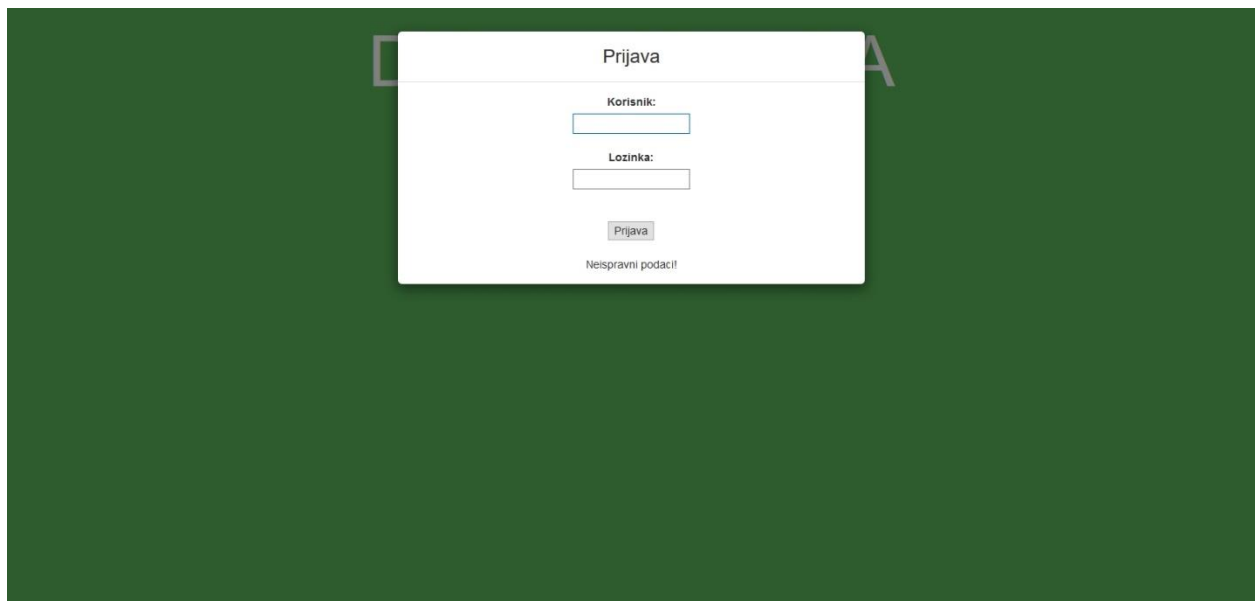
```

Zatim se unutar app.js radi preusmjeravanje korisnika na novu stranicu, ovisno o tome koji je niz znakova primljen od PHP skripte. Ako su podaci za prijavu neispravni, šalje se niz znakova „false“ koji se također kodira funkcijom „*json\_encode*“, ali u app.js se onda ispunjava uvjet kod *if* petlje da su primljeni podaci od PHP skripte jednaki nizu znakova „false“ što se očituje ispisivanjem poruke „Neispravni podaci!“ ispod gumba „Prijava“.

```

$http.post('login.php', data, config)
.then(function successCallback(response) {
    if (response.data == "false") {
        $scope.message = 'Neispravni podaci!';
    } else {
        console.log(response);
        $window.location.href = response.data;
    }
}, function errorCallback(reponse) {
    console.log(response);
});

```



*Sl. 3.5. Prijava pomoću netočnih podataka*

## 3.2. Predaja izvještaja

Kada se student prijavi u Dnevnik rada prikazuje mu se gumb za dodavanje izvještaja, gumb za odjavljivanje iz Dnevnika rada, ime mentora, prosjek ocjena svih ocijenjenih izvještaja te gumbi za pregled svih dodanih izvještaja.

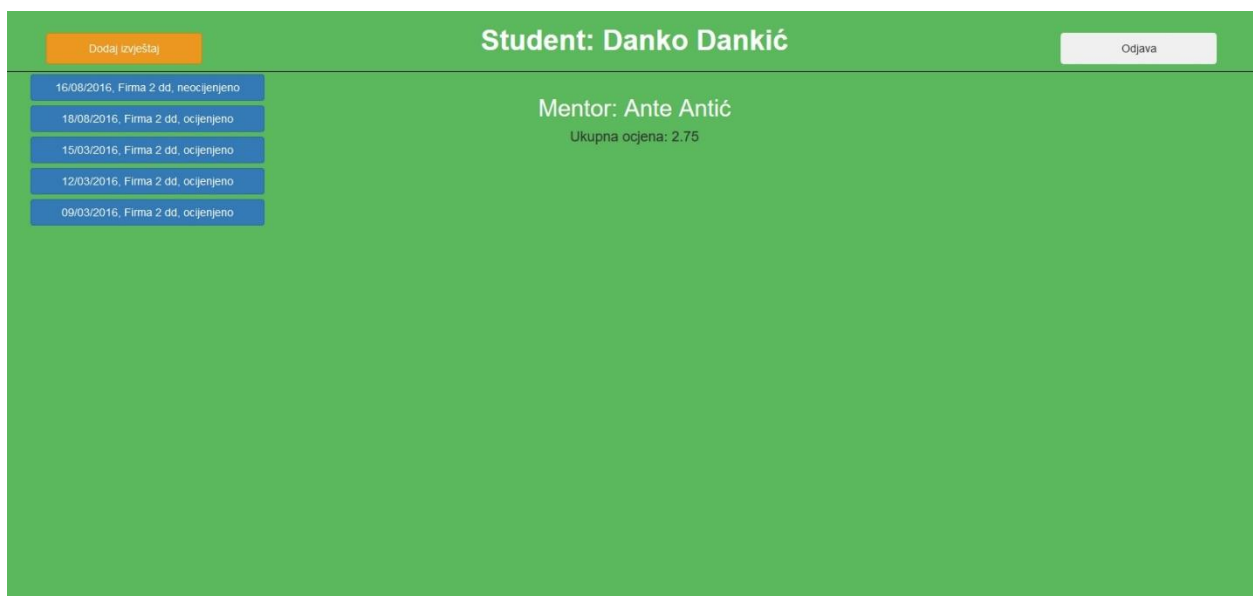
```
<center><h1 class="welcome">Student:
<?php
    while ($row1 = mysqli_fetch_array($res1)) {
        echo $row1['ime'];
    }
?></h1><br><h2 class="mentor">Mentor:
<?php
    while ($row3 = mysqli_fetch_array($res3)) {
        echo $row3['ime'];
    }
?></h2></center><center>
<?php
    while ($row2 = mysqli_fetch_array($res2)) {
        $zbroj+=$row2['ocjena'];
        $br++;
        if ($row2['ocjena'] == 1) {
            $zbroj--;
            $br--;
        }
    }
    if ($zbroj == 0 && $br == 0) {
        echo '<h4>Ukupna ocjena: nije ocijenjen nijedan izvjestaj.</h4>';
```

```

    } else {
        $arit = $zbroj/$br;
        echo '<h4>Ukupna ocjena: '.round($arit, 3).'</h4>';
    }
?>

<?php
while ($row = mysqli_fetch_array($res)) {
    if ($row['komentar'] == 0 AND $row['ocjena'] == 0) {
        echo "<button ng-click='openReview(\".$row['id'].\")' class='btn btn-primary
stud_review'>\".$row['datum'].\", \".$row['tvrтка'].\", neocijeno\".</button><br>";
    } else {
        echo "<button ng-click='openReview(\".$row['id'].\")' class='btn btn-primary
stud_review'>\".$row['datum'].\", \".$row['tvrтка'].\", ocijeno\".</button><br>";
    }
}
?>

```



**SL. 3.6.** Početna stranica student

Klikom na gumb za dodavanje izvještaja, korisniku se otvara modal u kojem se nalaze polja za ispunjavanje datuma, tvrtke u kojoj se praksa odrađuje, opisa zadatka te rješenja zadatka. Ispod forme u koju se upisuju svi potrebni podaci nalazi se gumb „Predaj“ koji služi za predaju izvještaja, tj. za upisivanje izvještaja u tablicu „izvjestaj“.

```

<form method="post" id="uploadForm" ng-submit="upload()">
    <label for="datum">Datum (dd/mm/gggg):</label>
    <input type="text" name="datum" ng-model="datum"><br><br>
    <label for="tvrтка">Tvrтка:</label>
    <input type="text" name="tvrтка" ng-model="tvrтка"><br><br>
    <label for="zadatak">Zadatak:</label>

```

```

        <textarea name="zadatak" ng-model="zadatak"></textarea><br><br>
        <label for="rjesenje">Rješenje:</label>
        <textarea name="rjesenje" ng-model="rjesenje"></textarea><br><br><br>
        <button>Predaj</button><br><br>
    </form>

```

Upisivanje izvještaja u tablicu izvršava se pomoću AngularJS skripte koja uzima podatke iz forme te ih šalje PHP skripti koja ih zatim „INSERT“ funkcijom stavlja u tablicu „izvjestaj“.

```

$scope.upload = function () {
    var data = {
        user: $cookies.get('user'),
        datum: $scope.datum,
        tvrtka: $scope.tvrtka,
        zadatak: $scope.zadatak,
        rjesenje: $scope.rjesenje
    };
    var config = {
        headers : {
            'Content-Type': 'application/x-www-form-urlencoded;charset=utf-8;'
        }
    }
}

```

```

$data = json_decode(file_get_contents("php://input"));
$user = $data->user;
$datum = $data->datum;
$tvrtka = $data->tvrtka;
$zadatak = $data->zadatak;
$rjesenje = $data->rjesenje;
$q="INSERT INTO izvjestaj (user, datum, tvrtka, zadatak, rjesenje, komentar, ocjena) VALUES ('$user',
'$datum', '$tvrtka', '$zadatak', '$rjesenje', '', '0')";
$res = mysqli_query($conn, $q);

```

Ako je prijenos podataka iz AngularJS skripte u PHP skriptu uspješan, te ako je i „INSERT“ funkcija uspješna, stranica se ponovno učitava kako bi student mogao vidjeti izvještaj kojeg je upravo predao.

```

$http.post('dashboard_stud_modal.php', data, config)
.then(function successCallback(response) {
    console.log(response);
    window.location.reload();
}, function errorCallback(reponse) {
    console.log(response);
});

```





```

<button ng-click="logout()" class="btn logout">Odjava</button>

$scope.logout = function () {
    window.location = 'index.php';
    var cookies = $cookies.getAll();
    angular.forEach(cookies, function (v, k) {
        $cookies.remove(k);
    });
}

```

### 3.3. Pregled izvještaja

Kao što je već napomenuto u prijašnjem poglavlju, na početnoj stranici studenta nalaze se gumbi koji služe za pregled već dodanih izvještaja. Na njima pišu datumi svakog izvještaja, tvrtke u kojima je praksa odrađena te jesu li izvještaji ocijenjeni ili neocijenjeni. Klikom na bilo koji od tih gumbova otvara se modal u kojem je izvještaj detaljnije prikazan, tj. prikazani su datum, tvrtka, opis zadatka, rješenje zadatka te komentar i ocjena mentora.

```

<?php
while ($row = mysqli_fetch_array($res)) {
    echo "<b>Datum:</b> ".$row['datum'];
    echo "<br><b>Tvrtka:</b> ".$row['tvrtka'];
    echo "<br><b>Zadatak:</b> ".$row['zadatak'];
    echo "<br><b>Rješenje:</b> ".$row['rjesenje'];
    if ($row['komentar']==NULL) {
        echo "<br><b>Komentar:</b> nije upisan";
    } else {
        echo "<br><b>Komentar:</b> ".$row['komentar'];
    }
    if ($row['ocjena']=='0') {
        echo "<br><b>Ocjena:</b> nije upisana";
    } else {
        echo "<br><b>Ocjena:</b> ".$row['ocjena'];
    }
}
?>

```

Ako komentar i/ili ocjena nisu upisani, u polju za komentar i ocjenu će pisati „nije upisan“/“nije upisana“. Dio AngularJS koda koji se odnosi na pregled dodanih izvještaja nalazi se u „reviewCtrl“, te on prima ID izvještaja iz PHP skripte koji se prenosi klikom na gumb izvještaja kojeg se želi pogledati. U „reviewCtrl“ se taj ID izvještaja sprema u kolačić „id“ kako bi aplikacija znala koji izvještaj treba prikazati.

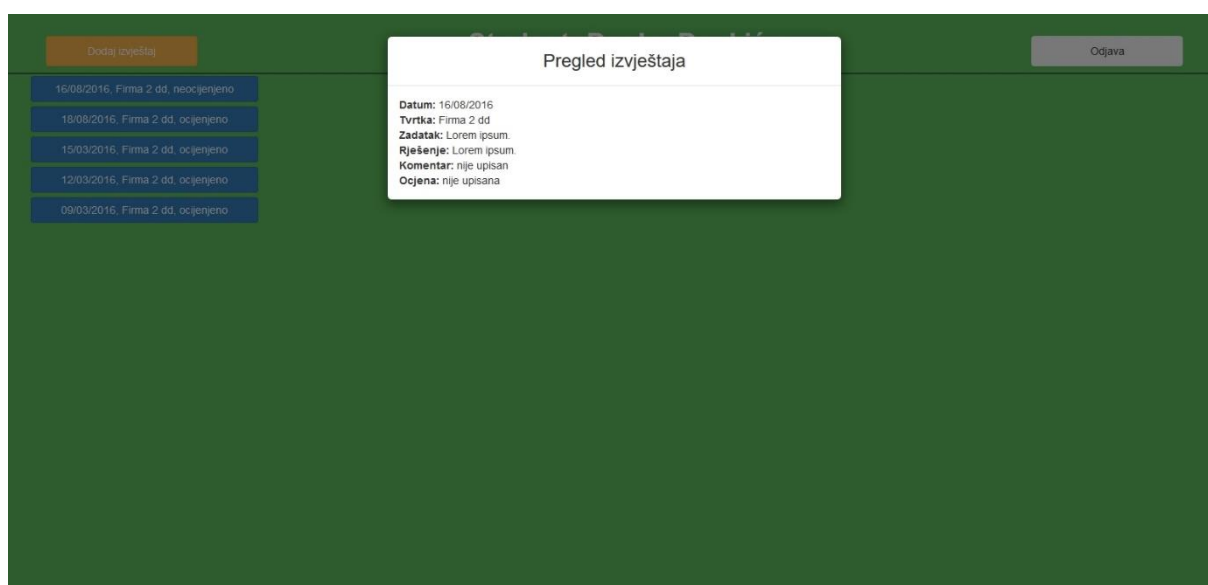
```

$scope.openReview = function (id) {
    $scope.id = id;
    $cookies.put('id', $scope.id);
}

```



*Sl. 3.9. Ocijenjeni izvještaj*



*Sl. 3.10. Neocijenjeni izvještaj*

### 3.4. Ocjenjivanje izvještaja

Kada se mentor prijavi u Dnevnik rada prikaže mu se broj studenata kojima je mentor, gumb za odjavu, te gumbi za pregled izvještaja na kojima pišu imena studenta, datumi izvještaja i jesu li izvještaji ocijenjeni ili ne.

<center><h1 class="welcome">Mentor:

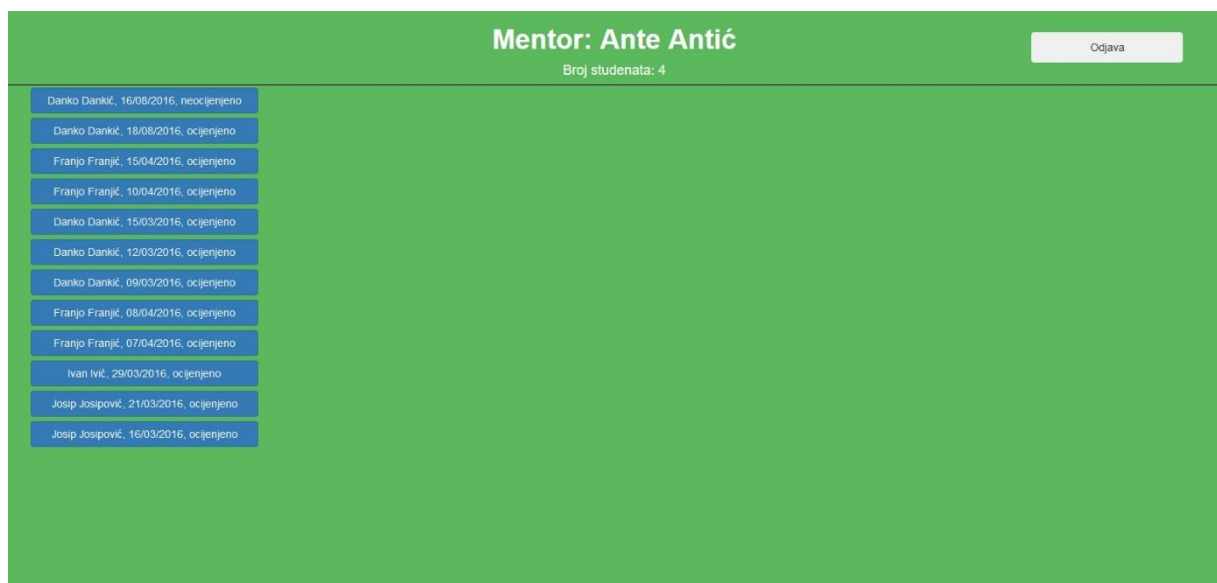
```

<?php
    $q1="SELECT login_ment.ime FROM login_ment WHERE login_ment.user='$user'";
    $res1 = mysqli_query($conn, $q1);
    while ($row = mysqli_fetch_array($res1)) {
        echo $row['ime'];
    }
?>/h1><h4 class="mentor">
<?php
    $q2="SELECT login_stud.user FROM login_stud WHERE login_stud.mentor='$user'";
    $res2 = mysqli_query($conn, $q2);
    $br=0;
    while ($row = mysqli_fetch_array($res2)) {
        $br++;
    }
    echo "Broj studenata: ".$br;
?>/h4></center>

<?php
    $q="SELECT izvestaj.id, login_stud.ime, izvestaj.datum, izvestaj.komentar, izvestaj.ocjena
FROM izvestaj, login_stud WHERE '$user'=login_stud.mentor AND izvestaj.user=login_stud.user ORDER
BY id DESC";
    $res = mysqli_query($conn, $q);
    while ($row = mysqli_fetch_array($res)) {
        if ($row['komentar'] == 0 AND $row['ocjena'] == 0) {
            echo "<button ng-click='openModal(\".$row['id'].\")' class='btn btn-primary
ment_review'>".$row['ime'].", ".$row['datum'].", neocijenjeno."</button><br>";
        } else {
            echo "<button ng-click='openModal(\".$row['id'].\")' class='btn btn-primary
ment_review'>".$row['ime'].", ".$row['datum'].", ocijenjeno."</button><br>";
        }
    }
}
?>

```

Pritiskom na gumb za odjavu, briše se kolačić „user“ koji se stvara prilikom prijave korisnika.



### Sl. 3.11. Početna stranica mentor

Klikom na jedan od gumba za pregled izvještaja otvara se modal u kojem piše detaljnije o izvještaju, tj. ime studenta, datum izvještaja, tvrtka, opis zadatka, rješenje zadatka, komentar te ocjena. Ako komentar nije upisan, ispisat će se „nije upisan“, a ako ocjena nije upisana, pisat će „nije upisana“.

The screenshot shows a web application interface. On the left, there is a list of reports (izvještaji) with columns for name, date, and status. The status is either 'neocijenjeno' (not rated) or 'ocijenjeno' (rated). On the right, a modal titled 'Ocjena izvještaja' (Report Rating) is open. The modal contains the following information:

- Ime: Danko Dankić
- Datum: 16/08/2016
- Tvrtka: Firma 2 dd
- Zadatak: Lorem ipsum
- Rješenje: Lorem ipsum
- Komentar: nije upisan
- Ocjena: nije upisana

Below this information, there are two input fields:

- Komentar: (text area)
- Ocjena: (number input, range 1-5)

At the bottom of the modal is a 'Predaj' (Submit) button. In the top right corner of the modal, there is a button labeled 'Odjava' (Logout).

### Sl. 3.12. Neocijenjeni izvještaj

Klikom na gumb se također prenosi ID izvještaja iz PHP skripte u AngularJS skriptu koja ga zatim sprema u kolačić „id“ kako bi se znalo koji izvještaj treba detaljnije prikazati u modalu.

```
$scope.openModal = function (id) {  
    $scope.id = id;  
    $cookies.put('id', $scope.id);  
}
```

Ispod izvještaja nalazi se forma s poljima za upisivanje komentara i ocjene te gumb za predaju forme.

```
<form method="post" id="reviewForm" ng-submit="review()">  
    <label for="komentar">Komentar:</label>  
    <textarea name="komentar" ng-model="komentar"></textarea><br><br>  
    <label for="ocjena">Ocjena:</label>  
    <input type="number" name="ocjena" min="1" max="5" ng-model="ocjena"><br><br><br>  
    <button>Predaj</button><br><br>  
</form>
```

Klikom na gumb „Predaj“ spremaju se komentar i ocjena u tablicu „izvjestaj“ tako da AngularJS skripta prenosi upisane podatke iz polja PHP skripti koja ih funkcijom „UPDATE“ pridodaje odgovarajućem izvještaju, tj. onom čiji ID odgovara kolačiću „id“.

```
$data = json_decode(file_get_contents("php://input"));
$komentar = $data->komentar;
$ocjena = $data->ocjena;
$user = $_COOKIE['user'];
$id = $_COOKIE['id'];
if ($komentar==NULL AND $ocjena!=NULL) {
    $q="UPDATE izvjestaj SET ocjena='$ocjena' WHERE izvjestaj.id='$id'";
} else if ($komentar!=NULL AND $ocjena==NULL){
    $q="UPDATE izvjestaj SET komentar='$komentar' WHERE izvjestaj.id='$id'";
} else {
    $q="UPDATE izvjestaj SET komentar='$komentar', ocjena='$ocjena' WHERE izvjestaj.id='$id'";
}
$res = mysqli_query($conn, $q);
```

The screenshot shows a web application interface. On the left, there is a list of reports (izvjestaji) with columns for user name, date, and status. The reports are listed in a table-like format. On the right, there is a modal window titled "Ocjena izvještaja" (Rating of report). The modal contains the following information:

- Ime:** Frano Franjić
- Datum:** 15/04/2016
- Tvrtka:** Tvrtka 3 doo
- Zadatak:** Vestibulum a scelerisque dolor. Suspendisse gravida, ante eget ullamcorper ultrices. sem nulla volutpat nisi. in finibus ligula purus et diam. Donec ornare eu mi a hendrerit. Phasellus eget ligula in odio cursus bibendum id eget ex. Nulla at cursus quam
- Rješenje:** Vestibulum a scelerisque dolor. Suspendisse gravida, ante eget ullamcorper ultrices. sem nulla volutpat nisi. in finibus ligula purus et diam. Donec ornare eu mi a hendrerit. Phasellus eget ligula in odio cursus bibendum id eget ex. Nulla at cursus quam
- Komentar:** Dovoljan
- Ocjena:** 2

Below the information, there are two input fields: "Komentar:" and "Ocjena:". The "Ocjena:" field has a dropdown arrow. At the bottom of the modal, there is a "Predaj" button.

**Sl. 3.13.** Ocijenjeni izvještaj

Ako je prijenos podataka iz AngularJS skripte u PHP skriptu uspješan, te ako je dodavanje komentara i ocjene u tablicu funkcijom „UPDATE“ uspješno, stranica se ponovno učitava kako bi mentor mogao vidjeti svoj komentar i ocjenu na upravo ocijenjenom izvještaju.

```
$http.post('dashboard_ment_modal.php', data, config)
.then(function successCallback(response) {
    console.log(response);
    window.location.reload();
    $cookies.remove('id');
}, function errorCallback(response) {
```

```
        console.log(response);
    });
```

### 3.5. Dizajn

Za izradu web aplikacije korišten je Bootstrap *framework* koji se koristi za stvaranje brzih i responzivnijih web stranica. Ima već napravljene dizajne za gumbe, tablice, forme, navigacijske trake i sl. Osim Bootstrap *frameworka*, korišten je i `style.css` dokument koji sadrži naredbe za promjenu boje pozadine, veličine teksta, itd. Kako bi se koristio Bootstrap *framework*, korišteni su „`class`“ atributi svakog HTML elementa. Radi na način da se „`class`“ atributu elementa pridoda određeno ime koje je ključna riječ unutar Bootstrapa te se onda tim imenom određuje kako će taj element izgledati. Od Bootstrap elemenata korišteni su samo gumbi kojima se onda mijenjala veličina, pozicija unutar web aplikacije te veličina teksta unutar gumba pomoću dodatne riječi u „`class`“ atributu (npr. `class="btn btn-warning dodaj"` → „`btn btn-warning`“ pripada Bootstrap *frameworku* i oblikuje gumb da bude žute boje i zaobljenih rubova, a „`dodaj`“ pripada „`style.css`“ dokumentu i oblikuje gumb prema naredbama koje se nalaze unutar dokumenta). Naredbama unutar „`style.css`“ dokumenta mijenjala se boja pozadine cjelokupne web aplikacije u zelenu te su se korigirali atributi pojedinih elemenata poput širine `div` elemenata, širine gumbova, margine gumbova, pozicije gumbova unutar stranice i dr.

#### CSS kod web aplikacije (`style.css`):

```
body {
    background-color: #5cb85c;
}
div {
    display: block;
}
.naslovna {
    color: white;
    font-size: 90px;
}
.welcome {
    color: white;
    font-weight: bold;
}
.mentor {
    color: white;
}
.prijava {
    width: 150px;
    height: 60px;
    font-size: 30px;
    text-align: center;
```

```

}
.dodaj {
    width: 200px;
    height: 40px;
    position: absolute;
    top: 30;
    left: 50;
}
.logout {
    width: 200px;
    height: 40px;
    position: absolute;
    top: 30;
    right: 50;
}
.stud_review {
    width: 300px;
    margin-left: 30px;
    margin-top: 3px;
    margin-bottom: 3px;
}
.glavni_div {
    border-bottom: 1px solid black;
    height: 80px; class
}
.dodaj_div {
    width: 33%;
    display: inline-block;
}
.top_div {
    width: 34%;
    display: inline-block;
}
.logout_div {
    width: 33%;
    display: inline-block;
}
.ment_review {
    margin-left: 30px;
    margin-top: 3px;
    margin-bottom: 3px;
    width: 300px;
}

```

## 4. ZAKLJUČAK

Kako bi izrada web aplikacije vođenja dnevnika rada bila moguća, trebalo se upoznati s PHP programskim jezikom, AngularJS skriptnim jezikom, HTML opisnim jezikom te WAMP aplikacijom koja u svom sastavu ima phpMyAdmin program za upravljanje bazom podataka.

Baza podataka se sastoji od tri tablice: „login\_stud“, „login\_ment“ te „izvjestaj“. Tablica „login\_stud“ sadrži podatke o studentima, ID studenta, korisničko ime, lozinku, puno ime i prezime te korisničko ime mentora. Tablica „login\_ment“ sadrži ID mentora, korisničko ime, lozinku te puno ime i prezime mentora, a tablica „izvjestaj“ sadrži ID izvještaja, korisničko ime studenta koji je predao izvještaj, datum izvještaja, tvrtku u kojoj se praksa odraduje, opis zadatka, rješenje zadatka, komentar mentora na izvještaj te ocjenu mentora.

Web aplikacija sastoji se od početne stranice, stranice za studente te stranice za mentore. Na početnoj stranici nalazi se gumb kojim se otvara modal s obrascem za prijavu u Dnevnik rada. Ako se student prijavi u Dnevnik rada, prikaže mu se stranica za studente na kojoj se nalazi gumb za dodavanje izvještaja, gumbi za pregled već dodanih izvještaja, gumb za odjavu iz Dnevnika rada te ime mentora i prosječna ocjena. Klikom na gumb za dodavanje izvještaja otvara se modal s obrascem za dodavanje datuma, tvrtke, opisa zadatka te rješenja zadatka koji se zatim sprema i stvara novi unos u tablici „izvjestaj“. Klikom na gumb za pregled dodanog izvještaja otvara se modal s detaljima o izvještaju (datum, tvrtka, opis zadatka, rješenje zadatka, komentar mentora, ocjena mentora). Klikom na gumb za odjavljivanje, studenta se preusmjerava na početnu stranicu. Ako se mentor prijavi u Dnevnik rada, prikaže mu se stranica za mentore s gumbima za pregled izvještaja od studenata kojima je mentor, gumb za odjavljivanje iz Dnevnika rada te broj studenata kojima je on mentor. Klikom na gumb za pregled izvještaja otvara se modal s detaljima o tom izvještaju i s obrascem za dodavanje komentara i ocjene na taj izvještaj koji se zatim spremaju u tablicu „izvjestaj“ za onaj izvještaj kojeg se želi ocijeniti. Klikom na gumb za odjavljivanje, mentora se preusmjerava na početnu stranicu.



## LITERATURA

- [1] <https://hr.wikipedia.org/wiki/HTML> (pristupljeno 13.6.2016.)
- [2] <https://en.wikipedia.org/wiki/AngularJS> (pristupljeno 13.6.2016.)
- [3] <https://hr.wikipedia.org/wiki/PHP> (pristupljeno 13.6.2016.)
- [4] <https://angularjs.org/> (pristupljeno 28.3.2016.)
- [5] <http://www.wampserver.com/en/> (pristupljeno 28.3.2016.)
- [6] <http://getbootstrap.com/> (pristupljeno 28.3.2016.)
- [7] A. Gutmans, S. Saether Bakken, D. Rethans, PHP 5 Power Programming, Prentice Hall, Hagerstown, listopad 2014.

## SAŽETAK

Tema završnog rada izrada je web aplikacije za vođenje dnevnika rada. Web aplikacija je rađena programskim jezikom PHP, skriptnim jezikom AngularJS, opisnim jezikom HTML te uz pomoć aplikacije WAMP koja služi za stvaranje baze podataka. Baza podataka sastoji se od tri tablice: „login\_stud“ koja sadrži podatke o studentima (ID, korisničko ime, lozinka, ime, korisničko ime mentora), „login\_ment“ koja sadrži podatke o mentorima (ID, korisničko ime, lozinka, ime) te „izvjestaj“ koja sadrži podatke o izvještajima (ID, korisničko ime studenta, datum, tvrtka, zadatak, rješenje, komentar, ocjena). Web aplikacija sastoji se od početne stranice na kojoj se nalazi gumb za prijavu u Dnevnik rada, stranice za studente na kojoj se nalaze gumb za dodavanje izvještaja, gumbi za pregled dodanih izvještaja, gumb za odjavu, ime mentora te prosječna ocjena, te stranice za mentore na kojoj se nalaze gumbi za pregled izvještaja od studenata kojima su mentori, gumb za odjavu te broj studenata kojima je taj korisnik mentor.

Ključne riječi: angularjs, php, dnevnik rada, html, css

## **ABSTRACT**

AngularJS application for keeping a work log

The topic of this final work is the making of a web application for keeping a work log. The web application is made with the help of PHP programming language, AngularJS scripting language, HTML markup language and WAMP application which is used for the creation of databases. The database consists of three tables: „login\_stud“ which contains information about the students (ID, username, password, name, mentor username), „login\_ment“ which contains information about the mentors (ID, username, password, name) and „izvjestaj“ which contains information about the reports (ID, student username, date, company, assignment, solution, comment, grade). The web application is comprised of a homepage on which there is a button for logging into the Work log, page for students which has a button for adding reports, buttons for reviewing already added reports, a button for logging out, name of the mentor and the average grade, and a page for mentors which consists of buttons for reviewing reports from students which have the logged in user as a mentor, a button for logging out and the number of students which have the logged in user as a mentor.

Keywords: angularjs, php, dnevnik rada, html, css

## ŽIVOTOPIS

Ivan Benke rođen je 6. travnja 1994. godine u Osijeku, Hrvatska. Živi na adresi Franje Račkog 15, Đakovo. Pohađa Osnovnu Školu „Ivan Goran Kovačić“ u Đakovu do 2009., te iste godine upisuje Gimnaziju „Antun Gustav Matoš“, opći smjer u Đakovu. Prva dva razreda srednje škole završava s vrlo dobrim, a druga dva s odličnim uspjehom. Po završetku srednje škole, 2013. godine, upisuje tadašnji Elektrotehnički fakultet Osijek, danas Fakultet elektrotehnike, računarstva i informacijskih znanosti Osijek, smjer Računarstvo.

Izvršno se služi engleskim jezikom, te ima osnovnu razinu znanja za rad s Microsoft Office alatima. Razina znanja za rad s programskim jezicima C, C++ i C# je također osnovna, dok ima srednju razinu znanja za rad s programskim jezicima PHP i AngularJS te opisnim jezicima HTML i CSS.

Dobitnik je stipendije za deficitarna zanimanja Nacionalne zaklade za potporu učeničkom i studentskom standardu za 2015./2016. akademsku godinu. Od radnog iskustva, treba spomenuti da je radio kao radnik u skladištu za METEOR d.d. u Đakovu te kao agent u teleprodajnom centru za Hrvatski Telekom u Osijeku.

Ivan Benke

---

## **PRILOZI**

CD:

PHP, HTML, AngularJS i CSS dokumenti koji sadrže cjelokupni rad (index.php, login.html, login.php, dashboard\_stud.php, dashboard\_stud\_modal.php, dashboard\_stud\_modal.php, dashboard\_stud\_review.php, dashboard\_ment.php, dashboard\_ment\_modal.php, dashboard\_ment\_review.php, app.js, app\_upload.js, app\_review.js)

Seminar u .doc i .pdf formatu