

Ponuda i potražnja usluga u građevinarstvu putem informacijsko komunikacijske tehnologije

Valentić, Luka

Undergraduate thesis / Završni rad

2023

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:200:061467>

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

Download date / Datum preuzimanja: **2024-11-30**

Repository / Repozitorij:

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



**SVEUČILIŠTE JOSIPA JURJA STROSSMAYERA U OSIJEKU
FAKULTET ELEKTROTEHNIKE, RAČUNARSTVA I
INFORMACIJSKIH TEHNOLOGIJA**

Stručni studij

**PONUDA I POTRAŽNJA USLUGA U
GRAĐEVNARSTVU PUTEM INFORMACIJSKO
KOMUNIKACIJSKE TEHNOLOGIJE**

Završni rad

Luka Valentić

Osijek 2023.

Sadržaj

1. UVOD	1
1.1. Zadatak završnog rada	1
2. PRIKAZ POSTOJEĆIH RJEŠENJA	2
2.1. Web aplikacija PraviMajstor	2
2.2. Web aplikacija eMajstor	3
2.3. Web aplikacija Workie	4
2.4. Web aplikacija Meštar	4
2.5. Dinamička web aplikacija za razmjenu usluga	5
3. RAZVOJ WEB APLIKACIJE	6
3.1. Tehnologije korištene pri izradi web aplikacije	6
3.2. Rješenje za klijentski dio web aplikacije	6
3.2.1. Naslovna stranica	7
3.2.2. Dio web aplikacije - Novi oglas	10
3.2.3. Dio web aplikacije - O oglasu	12
3.2.4. Responzivnost web stranice	13
3.3. Poslužiteljski (back-end) dio rješenja	15
3.3.1. Izrada tablica u bazi	15
3.3.2. Unos podataka u tablice	16
3.3.3. Kreiranje novog oglasa	20
4. KORIŠTENJE WEB APLIKACIJE	21
4.1. Prikaz naslovne stranice	21
4.2. Prikaz dijela web aplikacije – Novi oglas	23
4.3. Prikaz dijela web aplikacije – O nama	25
4.4. Prikaz dijela web aplikacije – O oglasu	26
5. ZAKLJUČAK	28

LITERATURA.....	29
SAŽETAK.....	31
ABSTRACT.....	32

**FERIT**FAKULTET ELEKTROTEHNIKE, RAČUNARSTVA
I INFORMACIJSKIH TEHNOLOGIJA **OSIJEK****Obrazac Z1S: Obrazac za imenovanje Povjerenstva za završni ispit na preddiplomskom stručnom studiju**

Osijek, 19.09.2023.

Odboru za završne i diplomske ispite

**Imenovanje Povjerenstva za završni ispit
na preddiplomskom stručnom studiju**

Ime i prezime Pristupnika:	Luka Valentić
Studij, smjer:	Stručni prijediplomski studij Računarstvo
Mat. br. Pristupnika, godina	AR 4701, 26.07.2018.
OIB Pristupnika:	41082773849
Mentor:	prof. dr. sc. Dominika Crnjac Milić
Sumentor:	prof. dr. sc. Krešimir Nenadić
Sumentor iz tvrtke:	Ivan Javorović
Predsjednik Povjerenstva:	doc. dr. sc. Tomislav Galba
Član Povjerenstva 1:	prof. dr. sc. Dominika Crnjac-Milić
Član Povjerenstva 2:	izv. prof. dr. sc. Alfonzo Baumgartner
Naslov završnog rada:	Ponuda i potražnja usluga u građevinarstvu putem informacijsko komunikacijske tehnologije
Znanstvena grana završnog rada:	Programsko inženjerstvo (zn. polje računarstvo)
Zadatak završnog rada	Potraga za izvođačem građevinskih radova često biva značajna životna prepreka. S druge strane nedostatak posla u Republici Hrvatskoj u sektoru građevinarstva rezultirao je migracijom zaposlenika u toj struci. Porast cijena građevinskog materijala rezultira s druge strane većim rizikom od financijskih gubitaka uslijed loše obavljenog posla od strane izvođača radova. Kako bi se premostili navedeni problemi putem informacijsko komunikacijske tehnologije moguće je napraviti platformu koja bi
Prijedlog ocjene pismenog dijela ispita (završnog rada):	Izvrstan (5)
Kratko obrazloženje ocjene prema Kriterijima za ocjenjivanje završnih i diplomskih radova:	Primjena znanja stečenih na fakultetu: 2 bod/boda Postignuti rezultati u odnosu na složenost zadatka: 2 bod/boda Jasnoća pismenog izražavanja: 3 bod/boda Razina samostalnosti: 3 razina
Datum prijedloga ocjene od strane mentora:	19.09.2023.

Potvrda mentora o predaji konačne verzije rada:

Mentor elektronički potpisao predaju konačne verzije.

Datum:

**FERIT**FAKULTET ELEKTROTEHNIKE, RAČUNARSTVA
I INFORMACIJSKIH TEHNOLOGIJA OSIJEK**IZJAVA O ORIGINALNOSTI RADA**

Osijek, 28.09.2023.

Ime i prezime studenta:

Luka Valentić

Studij:

Stručni prijediplomski studij Računarstvo

Mat. br. studenta, godina upisa:

AR 4701, 26.07.2018.

Turnitin podudaranje [%]:

9

Ovom izjavom izjavljujem da je rad pod nazivom: **Ponuda i potražnja usluga u građevinarstvu putem informacijsko komunikacijske tehnologije**

izrađen pod vodstvom mentora prof. dr. sc. Dominika Crnjac Milić

i sumentora prof. dr. sc. Krešimir Nenadić

moj vlastiti rad i prema mom najboljem znanju ne sadrži prethodno objavljene ili neobjavljene pisane materijale drugih osoba, osim onih koji su izričito priznati navođenjem literature i drugih izvora informacija.

Izjavljujem da je intelektualni sadržaj navedenog rada proizvod mog vlastitog rada, osim u onom dijelu za koji mi je bila potrebna pomoć mentora, sumentora i drugih osoba, a što je izričito navedeno u radu.

Potpis studenta:

1. UVOD

Nedostatak kvalitetnih i iskusnih izvođača radova može biti prepreka pri izvođenju građevinskih radova. Iako je s digitalizacijom i napretkom tehnologije taj proces pojednostavljen, postojeća web rješenja krajnjem korisniku mogu biti komplicirana. Od korisnika postojeća rješenja traže detaljnije poznavanje rada na računalu, nude prevelik broj informacija i može se reći da su komplicirana.

U ovom radu prikazano je rješenje problema u obliku web aplikacije koja povezuje izvođače radova s naručiteljima radova na jednostavniji i korisniku prihvatljiviji način.

Primarna svrha ove web aplikacije je pomoći ponuditeljima građevinskih usluga, te onima koji ju potražuju.

Web aplikacija naziva Majstors nudi korisnicima jednostavno korištenje. Od korisnika se traži da prilikom objava svojeg oglasa postavi sve potrebne informacije u samo nekoliko jednostavnih koraka. Za objavu oglasa, aplikacija ne traži kreiranje računa ili prijavu već nudi izravno objavljivanje oglasa. Pretraga oglasa je omogućena u četiri koraka kako bi korisnik što jednostavnije došao do majstora ili majstor do posla.

U drugom poglavlju opisan je prikaz postojećih rješenja web aplikacija, odnosno njihove prednosti i nedostatci. U trećem poglavlju opisane su tehnologije korištene za izradu web aplikacije detaljno je opisan način izrade i razvoja web aplikacije.

1.1. Zadatak završnog rada

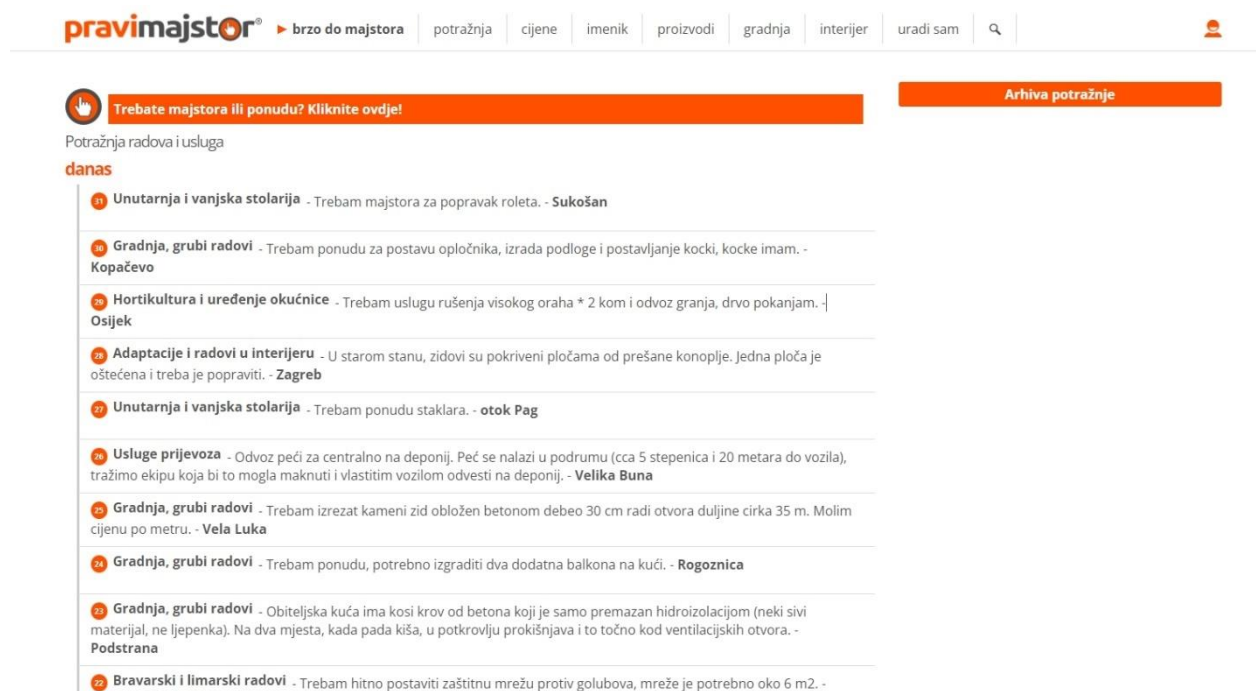
Potruga za izvođačem građevinskih radova često biva značajna životna prepreka. S druge strane nedostatak posla u Republici Hrvatskoj u sektoru građevinarstva rezultirao je migracijom zaposlenika u toj struci. Porast cijena građevinskog materijala rezultira s druge strane većim rizikom od financijskih gubitaka uslijed loše obavljenog posla od strane izvođača radova. Kako bi se premostili navedeni problemi putem informacijsko komunikacijske tehnologije moguće je napraviti platformu koja bi povezivala ponuđače i potraživače usluga u građevinarstvu. Radom treba izraditi web aplikaciju koja omogućava izvršiteljima radova i potraživačima radova objavu svojih oglasa i zahtjeva. Aplikacija treba biti interaktivna, jednostavna za korištenje i prilagodljiva za sve vrste uređaja (responzivni dizajn). Potrebno je dizajnirati i realizirati odgovarajuću bazu podataka koju će aplikacija koristiti za pohranu svih potrebnih podataka.

2. PRIKAZ POSTOJEĆIH RJEŠENJA

U ovom poglavlju dani su primjeri postojećih rješenja za zadanu web aplikaciju, njihove prednosti i mane. Neka od postojećih web aplikacija su: PraviMajstor, eMajstor, Workie, Meštar i Dinamička web aplikacija za razmjenu usluga Xchange Skills.

2.1. Web aplikacija PraviMajstor

PraviMajstor [1] je web aplikacija koja omogućava prikupljanje ponuda za građevinske i obrtničke radove ili usluge, te razne proizvode kao što su: građevinski materijal, boje, lakovi i zaštita, stolarija, vrt, alat i pribor, sirovine i slično.



The screenshot shows the PraviMajstor website interface. At the top, there is a navigation bar with the logo and menu items: 'brzo do majstora', 'potražnja', 'cijene', 'imenik', 'proizvodi', 'gradnja', 'interijer', 'uradi sam', and a search icon. Below the navigation bar, there is a red banner with the text 'Trebate majstora ili ponudu? Kliknite ovdje!' and a button 'Arhiva potražnje'. The main content area is titled 'Potražnja radova i usluga danas' and displays a list of job offers. Each offer is represented by a red circle with a number, followed by the job title and location, and a brief description of the work required.

Number	Job Title	Location	Description
31	Unutarnja i vanjska stolarija	Sukošan	Trebam majstora za popravak roleta.
30	Gradnja, grubi radovi	Kopačevo	Trebam ponudu za postavu oplodnika, izrada podloge i postavljanje kocki, kocke imam.
29	Hortikultura i uređenje okućnice	Osijek	Trebam uslugu rušenja visokog oraħa * 2 kom i odvoz granja, drvo pokanjam.
28	Adaptacije i radovi u interijeru	Zagreb	U starom stanu, zidovi su pokriveni pločama od prešane konoplje. Jedna ploča je oštećena i treba je popraviti.
27	Unutarnja i vanjska stolarija	otok Pag	Trebam ponudu staklara.
26	Usluge prijevoza	Velika Buna	Odvoz peći za centralno na deponij. Peć se nalazi u podrumu (cca 5 stepenica i 20 metara do vozila), tražimo ekipu koja bi to mogla maknuti i vlastitim vozilom odvesti na deponij.
25	Gradnja, grubi radovi	Vela Luka	Trebam izrezat kameni zid obložen betonom deĳeo 30 cm radi otvora duĳline cirka 35 m. Molim cijenu po metru.
24	Gradnja, grubi radovi	Rogoznica	Trebam ponudu, potrebno izgraditi dva dodatna balkona na kući.
23	Gradnja, grubi radovi	Podstrana	Obiteljska kuća ima kosi krov od betona koji je samo premazan hidroizolacijom (neki sivi materijal, ne ljepenka). Na dva mjesta, kada pada kiša, u potkrovlju prokišnjava i to točno kod ventilacijskih otvora.
22	Bravarski i limarski radovi		Trebam hitno postaviti zaštitnu mrežu protiv golubova, mreže je potrebno oko 6 m2.

Slika 2.1. Prikaz potražnje za radova ili uslugama na web aplikaciji pravimastor, izvor: [1]

Na slici 2.1. prikazana je stranica za potražnju na web aplikaciji, koja prikazuje neke radova i usluga koje se mogu pretraživati, bez obzira da li je potreban majstor ili samo ponuda radi usporedbe ranije dobivenih ponuda.

Prednosti ove web aplikacije su što obuhvaća pronalaženje majstora, usluga, građevinskih materijala i proizvoda sve na jednom mjestu.

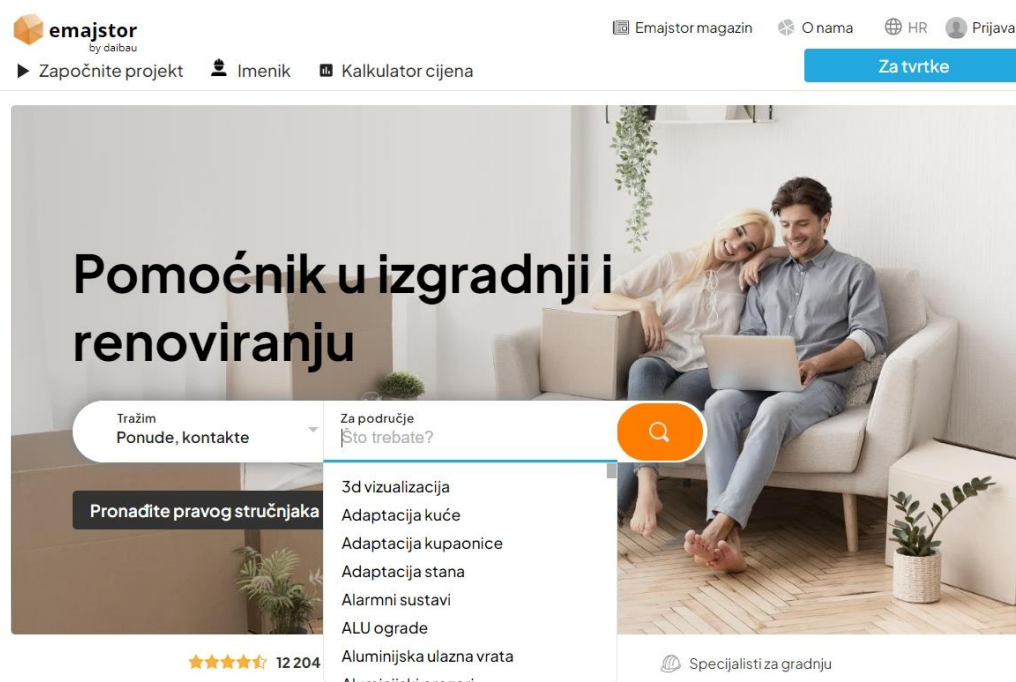
Nedostatak web aplikacije je u tome da se korisnik mora prijaviti u sustav ili ispuniti online obrazac kako bi došao do kontakta s majstorom. Što danas korisnici manje preferiraju jer imaju niz korisničkih imena i lozinki koje moraju čuvati/pamtiti. Stoga bi PraviMajstor trebao slijediti

primjer drugih sličnih alata koji u suradnji s Facebookom omogućuju prijavu na njihovu platformu, uz korištenje korisničkog imena i lozinke koja se koristi za prijavu na Facebook platformu.

2.2. Web aplikacija eMajstor

Web aplikacija eMajstor [2] povezuje izvođače radova s kupcima. Princip rada eMajstor je isti kao PraviMajstor. Korisnik ispunjava obrazac pomoću kojeg daje informacije o poslu za koji treba majstora, koliki je opseg posla, te kada počinje izvedba radova.

Prednosti web aplikacije eMajstor za razliku od PraviMajstor su jednostavnija upotreba, te brzina odziva na korisnikove akcije. Na web aplikaciji eMajstor postoje već unaprijed prikazane okvirne cijene po području posla, dok stranica PraviMajstor za cijene traži popunjavanje obrasca kako bi korisnik dobio informaciju o cijenama. Navedena razlika uvelike olakšava korištenje stranice eMajstor i čini stranicu je pristupačnijom korisniku koji želi brzo doći do rezultata.



Slika 2.2. Prikaz naslovne stranice web aplikacije, izvor: [2]

Slika 2.2. prikazuje naslovnu stranicu web aplikacije eMajstor, ono što se može uočiti na slici je pametno izrađena tražilica koja omogućuje lakše snalaženje korisnika. Tražilica se grana na majstore ili potencijalne korisnike usluge ovisno o vrsti područja rada ili usluge koju majstori nude, odnosno krajnji kupci potražuju.

2.3. Web aplikacija Workie

Za razliku od PraviMajstor i eMajstor, Workie [3] je puno jednostavnija web aplikacija za ponudu i potražnju građevinskih radova i radova općenito. Sastoji se od jednostavne tražilice po kategoriji posla i po županiji. Sučelje aplikacije se razlikuje po tome što osoba koja objavljuje ili traži oglas može biti majstor ili osoba koja traži majstora. Osobe koje traže majstora imaju opciju kreiranja računa putem kojeg je moguće ostaviti recenzije i kontaktirati majstore. Aplikacija nudi anonimnost tako da izvođači radova ne mogu vidjeti tko i kada provjerava njihove usluge, dok kontakt može biti ostvaren samo od strane korisnika koji potražuje uslugu majstora.

Naslovna stranica web aplikacije Workie prikazana je na slici 2.3. Aplikacija u startu usmjerava korisnike prema potrazi za majstorom, odnosno uslugom.

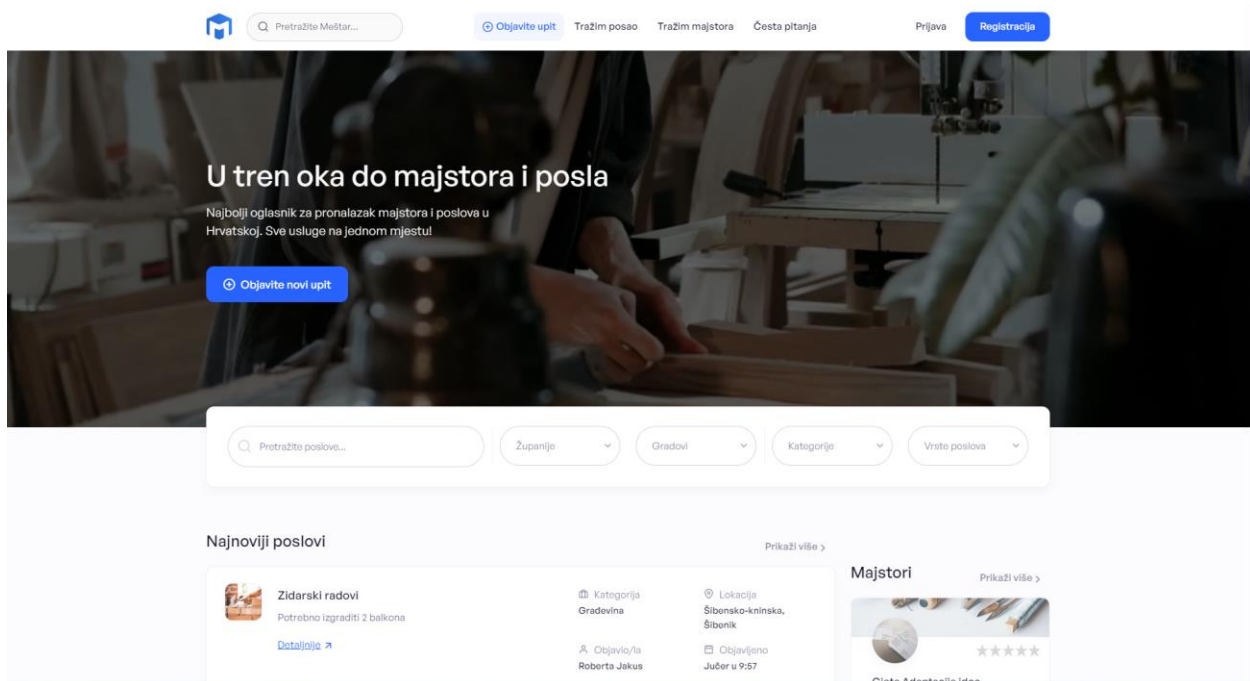


Slika 2.3. Prikaz naslovne stranice web aplikacije Workie, izvor: [3]

2.4. Web aplikacija Meštar

Dok su web aplikacije PraviMajstor i eMajstor slične funkcionalnosti, web aplikacija Meštar [4] i Workie također imaju povezanih funkcionalnosti. Kao i Workie, Meštar omogućava i korisnicima potražuju izvođača radova i majstorima objavljivanje oglasa ponude svojih usluga. Web aplikacija Meštar nudi opcije Tražim posao i Tražim majstora, dok kod PraviMajstor i eMajstor takva opcija nije uključena. Dok Workie ne koristi kreiranje računa za objavu oglasa, web aplikacija Meštar zahtjeva registraciju koja je obavezna ukoliko majstor ili korisnik žele objaviti oglas. Registracija na web aplikaciji Meštar moguća je i putem Facebook-a ili Google-a, dok aplikacije PraviMajstor

i eMajstor ne nude takvu opciju. Prikaz samih oglasa jednostavan je i nudi podjelu oglasa po kategorijama, županijama, gradovima i vrstom posla unutar određene kategorije. Na slici 2.4 nalazi se prikaz naslovne stranice web aplikacije Meštar.



Slika 2.4. Prikaz naslovne stranice web aplikacije Meštar, izvor: [4]

2.5. Dinamička web aplikacija za razmjenu usluga

Osim aplikacija koje omogućuju ponudu i potražnju usluga također postoje i aplikacije koje funkcioniraju na principu razmjene usluga za usluge. Primjer jedne od takvih web aplikacija je Xchange Skills. Stoga se u literaturi navodi da je upravo ova web aplikacija za razmjenu usluga [5, str.3] „aplikacija s bazom podataka u kojoj su pohranjeni podaci o građanima kojima je potrebna određena vještina i koji su spremni na razmjenu vještina na principu razmjene usluga“.

3. RAZVOJ WEB APLIKACIJE

U ovom poglavlju objašnjeno je programsko rješenje web aplikacije i tehnologije korištene pri izradi web aplikacije.

3.1. Tehnologije korištene pri izradi web aplikacije

Kada se govori o izradi web aplikacija, programer se treba odlučiti za skriptni jezik kojim će se služiti. Do sada je razvijen veći broj jezika koji imaju različite mogućnosti. Potrebno je znati što i kako aplikacija treba obavljati, odnosno koja je njena svrha i namjena, kako bi se na pravi način pristupilo odabiru. Odabir skriptnog jezika ovisi o programeru, te znanjima i vještinama koje posjeduje. Neki od najpoznatijih skriptnih jezika su JavaScript, Python, Ruby, Django, jezik za postavljanje sadržaja HTML (engl. Hyper Text Markup Language), jezik za uređivanje prikaza i stila CSS (engl. Cascading Style Sheets), PHP (engl. PHP: Hypertext Preprocessor) i drugi.

Osim skriptnih jezika, u web programiranju je važno poznavati i mjesta izvršavanja koda koji su: klijentski uređaji (eng. front-end) i poslužitelj (eng. back-end) [6]. Tehnologije na klijentskoj strani koriste se za interakciju korisnika i klijentskog dijela web aplikacije. Odnosi se na korisničko sučelje i korisničko iskustvo. Poslužiteljski dio web aplikacije obavlja funkcionalnosti koje korisnik direktno ne vidi. Neke od tih funkcionalnosti su pristup bazi podataka, pohranjivanje podataka, neki aspekti sigurnosti. Tehnologije na klijentskoj strani su: HTML, CSS i JavaScript, a neki od jezika za poslužiteljskoj strani su: PHP, Python, C++, Java te JavaScript koji je implementiran unutar Node.js platforme. U ovom radu naglasak će biti na HTML-u, CSS-u, JavaScriptu, Bootstrapu i Laravelu.

3.2. Rješenje za klijentski dio web aplikacije

Tehnologije na klijentskoj strani odgovorne su za dizajn web aplikacije kao i za interakciju korisnika s web aplikacijom u obliku prikaza podataka korisniku ili u obliku prikaza obrazaca u kojima korisnik može upisivati podatke. Osim navedenih funkcionalnosti na klijentskom dijelu web aplikacije može se obavljati i validacija podataka te interpretacija akcija koje korisnik obavlja sa sučeljem web aplikacije.

Za izradu klijentsko dijela web aplikacije u ovom radu korišteni su HTML, CSS, JavaScript jezici i Bootstrap koji su zajedno omogućili prikaz i funkcionalnost web stranice u postojećem obliku. HTML služi za opisivanje hiper tekstualnih datoteka. Pomoću HTML-a definiraju se dijelovi kao

što su: glava i tijelo (eng. Head, body) [7] [8]. Web preglednik čita te parametre i interpretira ih korisnicima web stranice.

Klijentski dio web aplikacije podijeljen je u nekoliko logičkih organizacijskih cjelina. Navedena web aplikacija se sastoji od sljedećih dijelova: *Naslovna stranica*, *Izradi oglas*, *O nama*.

3.2.1. Naslovna stranica

Naslovna stranica sastoji se od sljedećih stavki: navigacijske trake, tražilice oglasa i prostora za ispis pretraženih oglasa. Na slici 3.1. prikazan je dio koda kojim je izrađena navigacijska traka koja se sastoji od logotipa aplikacije, izbornika i kontrole *Napravi oglas*. Navigacijska traka uređena je korištenjem bootstrap klasa [9]: *navbar*, *navbar-expand-lg*, *navbar-light*, *bg-light*, *container*. Pomoću tih klasa određeno je ponašanje navigacijske trake, boja teksta i boja pozadine.

```
20 <nav class="navbar navbar-expand-lg navbar-light bg-light container">
21 <a class="navbar-brand" href="{route('home')}}"> Naslovnica </a>
22 {
23 <button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarSupportedContent" aria-controls="navbarSupportedContent"
24     aria-expanded="false" aria-label="Toggle navigation">
25     <span class="navbar-toggler-icon"></span>
26 </button>
27
28 <div class="collapse navbar-collapse" id="navbarSupportedContent">
29     <ul class="navbar-nav mr-auto">
30         <li class="nav-item active">
31             <a class="nav-link" href="{route('create')}}"> Novi oglas </a>
32         </li>
33     </ul>
34     <ul class="navbar-nav mr-auto">
35         <li class="nav-item active">
36             <a class="nav-link" href="{route('onama')}}"> O nama </a>
37         </li>
38     </ul>
39 </div>
</nav>
```

Slika 3.1. Prikaz koda kojim je izrađena navigacijska traka

Na slici 3.2. nalazi se kod kojim je izrađen logotip. Kako je prikazano u kodu, logotip se sastoji od PNG (engl. *Portable Network Graphics*) slike, teksta i SVG (engl. *Scalable Vector Graphics*) vektorske grafike. PNG slika „majstora“ je preuzeta s besplatne stranice za slike [10], korišteni font za tekst logotipa je Nunito, Sans Serif, te ikona kućice je SVG vektorska grafika pronađena na besplatnim bootstrap ikonama [11].

```

<a class="navbar-brand" href="{{route('home')}}">
  
  <span class="text-xl font-bold tracking-wide" style="position: relative;">
    M
    <svg xmlns="http://www.w3.org/2000/svg" width="16" height="16" fill="currentColor" class="bi
    bi-house-door-fill" viewBox="0 0 16 16" style="margin-bottom: 5px;"><path d="M6.5 14.5v-3.505c0-.245.25-.495.
    5-.495h2c.25 0 .5.25.5.5v3.5a.5.5 0 0 0 .5.5h4a.5.5 0 0 0 .5-.5v-7a.5.5 0 0 0-.146-.354L13 5.793V2.5a.5.5 0 0
    0-.5-.5h-1a.5.5 0 0 0-.5.5v1.293L8.354 1.146a.5.5 0 0 0-.708 0l-.6 6A.5.5 0 0 0 1.5 7.5v7a.5.5 0 0 0 .5.5h4a.5.
    5 0 0 0 .5-.5z"/></path></svg>
    jstors
  </span>
</a>

```

Slika 3.2. HTML kod logotipa

Slika 3.3. prikazuje izgled logotipa aplikacije, navedeni čini vizualni identitet ne samo stranice već i potencijalne tvrtke. Stoga je važno logotip izraditi što kvalitetnije i privlačnije kako bi postao znak prepoznatljivosti same web aplikacije.



Slika 3.3. Prikaz logotipa weba aplikacije Majstors

Za postizanje responzivnosti navigacijske trake, korištene su bootstrap klase: *collapse* i *navbar-collapse*, kako bi se navigacijska traka u prikazu na mobilnim uređajima promijenila. Pomoću HTML elementa *button* koji sadrži klasu *navbar-toggler* moguće je kreiranje kontrole za aktiviranje padajućeg izbornika u mobilnom prikazu web aplikacije. Na slici 3.4. prikazan je kod za kreiranje kontrole padajućeg izbornika sa elementom *button*.


```

<button class="navbar-toggler" type="button" data-bs-toggle="collapse"
data-bs-target="#navbarSupportedContent" aria-controls="navbarSupportedContent" aria-expanded="false"
aria-label="Toggle navigation">
<span class="navbar-toggler-icon"></span>
</button>

```

Slika 3.4. Kod za kreiranje kontrole padajućeg izbornika

U tijelu aplikacije nalazi se tražilica oglasa s tri padajuća izbornika za odabir naredbe s vrijednostima: *Što tražiš*, *Odaberi županiju* i *Odaberi kategoriju/zanimanje*.

Svaka ponuđena opcija vezana je uz vrijednost (engl. *value*) koju poslužitelj preuzima nakon što korisnik pritisne naredbu *Pretraži*. Na slici 3.5. vidi se prikaz HTML koda kojim je izrađena navedena tražilica. Popis županija i kategorija povezan je s prethodno ispunjenim tablicama *zupanije* i *kategorije* u bazi podataka. Padajući izbornici kreirani su pomoću HTML elementa *select* [12]. Korisnik treba ispuniti sva polja kako bi mogao odabrati naredbu *Pošalji*. Uz pomoć atributa *required*, postignuto je da korisnik u svakom od izbornika treba odabrati jednu od ponuđenih vrijednosti.

```

<select name="vrsta" required>
  <option selected hidden disabled value="">Što tražiš?</option>
  <option value="trazim"> Tražim majstora </option>
  <option value="nudim"> Nudim majstorske poslove </option>
</select>
<select name="zupanija" required>
  <option selected hidden disabled value="">Odaberi županiju</option>
  @foreach($zupanije as $zupanija)
    <option value="{{ $zupanija->id }}"> {{ $zupanija->naziv }} </option>
  @endforeach
</select>
<select name="kategorija" required>
  <option selected hidden disabled value="">Odaberi kategoriju/zanimanje</option>
  @foreach($kategorije as $kategorija)
    <option value="{{ $kategorija->id }}"> {{ $kategorija->naziv }} </option>
  @endforeach
</select>

```

Slika 3.5. Prikaz HTML koda tražilice i funkcionalnosti ispisa kategorija i županija

Na slici 3.6. prikazana je funkcionalnost ispisa oglasa nakon što korisnik odabere naredbu *Pretraži*. Funkcija prolazi kroz petlju *foreach* koja preuzima podatke iz tablice *Oglasi*. Kako bi funkcija mogla ispisati točno tražene oglase, predane su joj vrijednosti koje je korisnik odabrao: *kategorija*,

županija, te vrši li se ispis oglasa ponude, odnosno potražnje poslova uz vrijednost *trazim* ili *nudim*.

```
<div id="oglas1">
  @if(isset($oglas1))
  <div id="popis">
    <h2 {{ $podnaslov }}</h2>
  </div>
  <div class="row justify-content-center align-items-center">
    @foreach($oglas1 as $oglas)
      <div class="card">
        <div class="card-body">
          <h5 class="card-title">{{ $oglas->naslov }}</h5>
          <h6 class="card-subtitle mb-2 text-muted">{{ $oglas->kategorija->naziv }}</h6>
          <p class="card-text">{{ $oglas->opis }}</p>
          <a href="{{ route('pregled', ['oglas_id' => $oglas->id, 'vrsta' => $vrsta]) }}" class="btn btn-info">Prikaži
            više</a>
        </div>
      </div>
    @endforeach
  </div>
@endif
</div>
```

Slika 3.6. Prikaz ispisa oglasa

3.2.2. Dio web aplikacije - Novi oglas

Web aplikacija Majstors sastoji se i od stranice koja služi za izradu novog oglasa. Stranica *Novi oglas* dijeli se na dva dijela. Prvi dio je već navedena i prethodno objašnjena navigacijska traka, dok je drugi dio tijelo stranice. Tijelo stranice sastoji se od obrasca za izradu novog oglasa. Kod obrasca za izradu oglasa prikazan je na slici 3.7. Izrada novog oglasa se sastoji od sljedećih polja: *Naslov oglasa*, *Tražiš ili nudiš posao*, *Županija*, *Adresa*, *Kategorija*, *Email*, *Kontakt broj*, *Opis* i *Oglas vrijedi do*, te kontrole *Predaj oglas*.

Kao i kod tražilice oglasa, popis županija i kategorija povezan je s prethodno ispunjenim tablicama *zupanije* i *kategorije* koje se nalaze u bazi podataka. Korisniku koji primjeni opciju *Nudim posao* biti će prikazano novo polje *Adresa*. Unos adrese služi kako bi korisnik koji traži majstora mogao doći do majstorske radnje uživo. Polje adresa skriveno je od korisnika pomoću CSS naredbe za kontrolu izgleda `style="display: none"`.

```

<div id="forma">
  <h2 id="title"> Novi oglas </h2>
  <form id="create" method="POST" action="{{route('store')}}">
    @csrf
    <div class="form-group">
      <label for="naslov"> Naslov oglasa </label>
      <input class="form-control" type="text" name="naslov">
    </div>
    <div class="form-group">
      <label for="vrsta"> Tražiš ili nudiš posao ? </label>
      <select class="form-control" name="vrsta" id="vrsta">
        <option value="trazim"> Tražim posao </option>
        <option value="nudim"> Nudim posao </option>
      </select>
    </div>
    <div class="form-group">
      <label for="zupanija"> Županija </label>
      <select class="form-control" name="zupanija">
        <option selected hidden disabled></option>
        @foreach($zupanije as $zupanija)
          <option value="{{ $zupanija->id }}"> {{ $zupanija->naziv }}
        </option>
        @endforeach
      </select>
    </div>
    <div class="form-group" id="adresa" style="display: none">
      <label for="adresa"> Adresa </label>
      <input class="form-control" type="text" name="adresa">
    </div>
    <div class="form-group">
      <label for="kategorija"> Kategorija </label>
      <select class="form-control" name="kategorija"
      id="select_kategorija">
        <option selected hidden disabled></option>
        @foreach($kategorije as $kategorija)
          <option value="{{ $kategorija->id }}"> {{ $kategorija->naziv }}
        </option>
        @endforeach
        <option value="0" id="ostalo" style="display: none;"> Ostalo </
        option>
      </select>
    </div>
    <div class="form-group" id="nova_kategorija" style="display: none">
      <label for="nova_kategorija"> Navedi kategoriju </label>
      <input class="form-control" type="text" name="nova_kategorija">
    </div>
    <div class="form-group">
      <label for="email"> Email </label>
      <input class="form-control" type="email" name="email">
    </div>
    <div class="form-group">
      <label for="kontakt"> Kontakt broj </label>
      <input class="form-control" type="text" name="kontakt">
    </div>
    <div class="form-group">
      <label for="opis"> Opis </label>
      <textarea class="form-control" name="opis"
      placeholder="Opis oglasa"></textarea>
    </div>
    <div class="form-group">
      <label for="vrijedi_do"> Oglas vrijedi do </label>
      <input class="form-control" type="date"
      name="vrijedi_do">
    </div>
    <button type="submit" id="submit-btn" class="btn btn-info">
      Predaj oglas </button>
  </form>

```

Slika 3.7. Prikaz obrazaca za izradu novih oglasa

Polje *Navedi kategoriju* dostupno je samo kada korisnik u padajućem izborniku kategorija odabere *ostalo*. Polja *Navedi kategoriju* i polje *Adresa* na stranici se prikazuju pomoću JavaScript-a koji mijenja njihove CSS vrijednosti atributa *style*. Prikaz JavaScript funkcije *addEventListener* [13] i promjena atributa *style* prikazana je na slici 3.8.

```
<script>
  const kategorije = document.getElementById('select_kategorija');
  const vrsta = document.getElementById('vrsta');
  kategorije.addEventListener('change', () => {
    let nova_kategorija = document.getElementById('nova_kategorija');
    if (kategorije.value == 0){
      nova_kategorija.style.display = 'block';
    }else{
      nova_kategorija.style.display = 'none';
    }
  });

  vrsta.addEventListener('change', () => {
    let adresa = document.getElementById('adresa');
    let ostalo = document.getElementById('ostalo');
    if (vrsta.value == 'nudim'){
      adresa.style.display = 'block';
      ostalo.style.display = 'block';
    }else{
      adresa.style.display = 'none';
      ostalo.style.display = 'block';
    }
  });
</script>
```

Slika 3.8. JavaScript skripta za dodavanje polja *Adresa* i *Navedi kategoriju*

3.2.3. Dio web aplikacije - O oglasu

Ovaj dio web aplikacije se aktivira kada korisnik, koji je na naslovnoj stranici pretražio oglase, odabere kontrolu *Prikaži više* uz određeni oglas. Prilikom pritiska na kontrolu, otvara se stranica o tome oglasu, gdje korisnik može saznati sve detaljne informacije vezane uz odabrani oglas. Prikaz koda za ispis podataka o oglasu nalazi se na slici 3.9.

```

<div class="container">
  <div id="forma">
    <h2 id="title"> {{{oglas->naslov}}} </h2>

    <div class="form-group">
      <label for="zupanija"> Županija </label>
      <input class="form-control" type="text" value="{{oglas->zupanija->naziv}}" readonly>
    </div>

    @if($oglas->adresa)
      <div class="form-group">
        <label for="adresa"> Adresa </label>
        <input class="form-control" type="text" value="{{oglas->adresa}}" readonly>
      </div>
    @endif

    <div class="form-group">
      <label for="email"> Email </label>
      <input class="form-control" type="email" value="{{oglas->email}}" readonly>
    </div>

    <div class="form-group">
      <label for="kontakt"> Kontakt broj </label>
      <input class="form-control" type="text" value="{{oglas->kontakt}}" readonly>
    </div>

    <div class="form-group">
      <label for="opis"> Opis </label>
      <textarea class="form-control" readonly>{{{oglas->opis}}}</textarea>
    </div>

    <div class="form-group">
      <label for="vrijedi_do"> Oglas vrijedi do </label>
      <input class="form-control" type="email" value="{{oglas->vrijedi_do}}" readonly>
    </div>
  </div>
</div>

```

Slika 3.9. Kod za ispis podataka o oglasu

3.2.4. Responzivnost web stranice

Responzivnost web aplikacije je postignuta korištenjem prethodno definiranih klasa iz bootstrap biblioteke. Kao što je prikazano na slici 3.10., korištenjem *@media* [14] naredbe unutar CSS dokumenta, pomoću CSS izraza moguće je ograničiti da djeluju samo kada je uvjet ispunjen.

```

@media only screen and (max-width: 975px) {
  select {
    width: 90%;
    word-wrap: normal;
    text-align: center;
    height: 30px;
    margin-top: 10px;
    border-radius: 5px;
  }
}
@media only screen and (max-width: 768px) {
  .navbar-brand { display: none; }
  .btn .btn-info{
    margin-top: 0.5rem;
  }
}
@media only screen and (min-width:768px ){
  .naslov-mob{
    display: none !important;
  }
}

```

Slika 3.10. Prikaz @media naredba i zadanih

U kodu postoje 3 uvjeta:

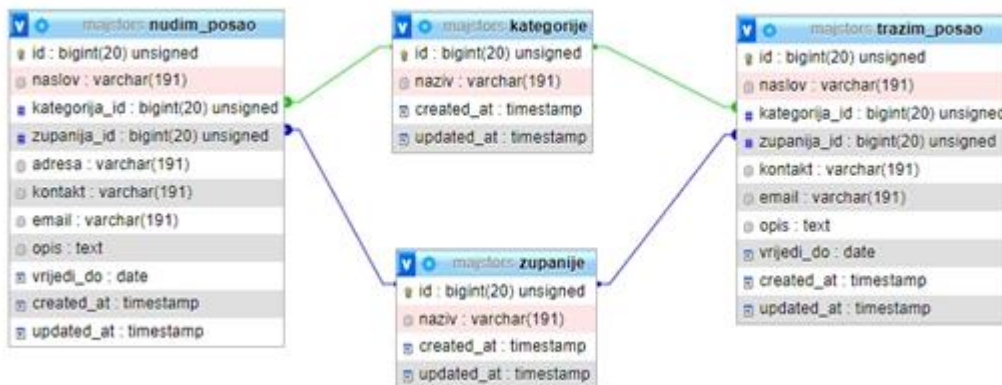
1. kada je ekran maksimalne širine 975px (piksela),
2. kada je širina ekrana manja od 768px i
3. kada je širina ekrana veća od 768px.

Prvi omogućava da se tražilica za ispis oglasa prilikom smanjivanja širine ekrana ispod 975px proširi na 90% širine nadređenog elementa. Drugi sakriva logotip aplikacije i pomiče kontrolu s pripadajućom klasom kada je širina ekrana manja od 768px. Treći omogućava da je logotip prikazan na ekranima manje veličine na sredini, dok se na širini ekrana većoj od 768px on sakrije.

3.3. Poslužiteljski (back-end) dio rješenja

Poslužiteljski dio web aplikacije nikada nije vidljiv korisnicima. Poslužiteljski dio se može sastojati od jednog ili više dijelova (skripti) Najpoznatiji jezik za razvoj poslužiteljskog dijela je PHP [13]. On je u potpunosti besplatan, te se izvršava na poslužitelju. Za izradu web aplikacije u ovom radu korišten je PHP-ov aplikacijski okvir Laravel[14].

U ovom poglavlju detaljnije su objašnjene tehnologije korištene za izradu poslužiteljskog dijela. Govori se o načinu izrade baze podataka, samoj bazi podataka i kako se upravlja podatcima u bazi. Slika 3.11. prikazuje ER (engl. *Entity-Relationship*) dijagram tablica iz baze podataka. Tablice koje se nalaze u bazi su: *trazim_posao*, *nudim_posao*, *kategorije* i *županije*. Tablice *nudim_posao* i *trazim_posao* povezane su s tablicama *kategorije* i *županije*, tako da su *kategorija_id* i *zupanija_id* povezani pomoću stranog ključa (engl. *Foreign key*) [15]. Kada se koriste podaci za kategorije i županije unutar tablica *trazim_posao* i *nudim_posao*, one mogu biti korištene jedino iz prethodno definiranih tablica *kategorije* i *županije* uz odgovarajući identifikator.



Slika 3.11. Prikaz ER dijagram baze podataka

3.3.1. Izrada tablica u bazi

Za izradu tablica u bazi koriste se klase *CreateTrazimPosaoTable*, *CreateNudimPosaoTable*, *CreateKategorijeTable* i *CreateZupanijeTable*. Na slici 3.12. se nalazi prikaz koda klase *CreateTrazimPosaoTable*.

```

class CreateTrazimPosaoTable extends Migration
{
    /** ...
    public function up()
    {
        Schema::create('trazim_posao', function (Blueprint $table) {
            $table->id();
            $table->string('naslov');
            $table->unsignedBigInteger('kategorija_id');
            $table->unsignedBigInteger('zupanija_id');
            $table->string('kontakt')->nullable();
            $table->string('email')->nullable();
            $table->text('opis');
            $table->date('vrijedi_do')->nullable();
            $table->timestamps();

            $table->foreign('kategorija_id')->references('id')->on('kategorije');
            $table->foreign('zupanija_id')->references('id')->on('zupanije');
        });
    }

    /** ...
    public function down()
    {
        Schema::dropIfExists('trazim_posao');
    }
}

```

Slika 3.12. Prikaz klase *CreateTrazimPosaoTable*

3.3.2. Unos podataka u tablice

Kao što je već spomenuto, za potrebe testiranja aplikacije pomoću funkcija *NudimPosaoSeeder* i *TražimPosaoSeeder*, popunjeni su podatci u tablice *trazim_posao* i *nudim_posao*.

NudimPosaoSeeder i *TražimPosaoSeeder* su Laravel klase koje omogućavaju zapis već definiranih podataka u bazu podataka. [16]

Na slikama 3.13. i 3.14. nalazi se kod kojim se popunjava baza generiranim podacima za oglase.


```

namespace Database\Seeders;

use App\Models\NudimPosao;
use Illuminate\Database\Seeder;

class NudimPosaoSeeder extends Seeder
{
    /**
     * Run the database seeds.
     *
     * @return void
     */
    public function run()
    {
        for($i = 1; $i < 1500; $i++){
            $kategorijaId = rand(1,10);
            $zupanijaId = rand(1,10);
            $model = new NudimPosao();
            $model->naslov = 'Nudim posao - '.$i;
            $model->kategorija_id = $kategorijaId;
            $model->zupanija_id = $zupanijaId;
            $model->adresa = 'Adresa majstora - '.$i;
            $model->kontakt = 'Kontakt - '.$i;
            $model->email = 'Email - '.$i;
            $model->opis = 'Opis posla- '.$i;
            $model->vrijedi_do = date("Y-m-d", strtotime("+1 week"));
            $model->save();
        }
    }
}

```

Slika 3.13. Prikaz popunjavanje tablice nudim posao


```

namespace Database\Seeders;

use App\Models\TrazimPosao;
use Illuminate\Database\Seeder;

class TrazimPosaoSeeder extends Seeder
{
    /**
     * Run the database seeds.
     *
     * @return void
     */
    public function run()
    {
        for($i = 1; $i < 1500; $i++){
            $kategorijaId = rand(1,10);
            $zupanijaId = rand(1,10);
            $model = new TrazimPosao();
            $model->naslov = 'Tražim majstora - '.$i;
            $model->kategorija_id = $kategorijaId;
            $model->zupanija_id = $zupanijaId;
            $model->kontakt = 'Kontakt - '.$i;
            $model->email = 'Email - '.$i;
            $model->opis = 'Opis posla- '.$i;
            $model->vrijedi_do = date("Y-m-d", strtotime("+1 week"));
            $model->save();
        }
    }
}

```

Slika 3.14. Prikaz popunjavanje tablice tražim posao

TrazimPosaoSeeder i *NudimPosaoSeeder* su Laravel funkcije koje omogućavaju unos testnih podataka u tablicu u bazi kroz program. U ovim funkcijama postoji *for* petlja koja ide kroz 1500 brojeva i unosi ih u kategorije i županije pod identifikatorom kojeg generira funkcija *rand* (1,10). S obzirom na to da postoji 21 županija i 21 kategorija, popunjeno je samo prvih 10 županija i kategorija kako bi svaka tablica unutar baze podataka imala veći broj oglasa za testiranje aplikacije. Funkcije *TrazimPosaoSeeder* i *NudimPosaoSeeder* pozivaju se naredbom: *php:artisan db:seed*.

Osim *TrazimPosaoSeeder* i *NudimPosaoSeeder* za aplikaciju se koriste klase *ZupanijeSeeder* i *KategorijeSeeder*, koje popunjavaju tablicu županija i kategorija poslova. Podatci za tablice *kategorije* i *zupanije* uneseni su kroz funkciju *foreach* koja prolazi kroz zadano polje podataka i unosi ih u tablicu. Na slici 3.15. vidljiv je kod koji obavlja unos podataka u tablicu *kategorije*.

```
class KategorijeSeeder extends Seeder
{
    /**
     * Run the database seeds.
     *
     * @return void
     */
    public function run()
    {
        $array = [
            'Automehaničar',
            'Bravar',
            'Elektroinstalater',
            'Fasader',
            'Keramičar',
            'Krovopokrivač',
            'Montaža namještaja',
            'Odvoz glomaznih stvari',
            'Parketar',
            'Plinoinstalater',
            'Rent A-car',
            'Servis bijele tehnika',
            'Servis mobitela',
            'Servis računala',
            'Servis strojeva i alata',
            'Soboslikar',
            'Staklar',
            'Stolar',
            'Suha gradnja',
            'Tokar',
            'Vodoinstalater'
        ];

        foreach ($array as $kategorija){
            $model = new Kategorije();
            $model->naziv = $kategorija;
            $model->save();
        }
    }
}
```

Slika 3.15. Prikaz unosa podataka u tablicu kategorija

3.3.3. Kreiranje novog oglasa

Iako se za potrebe testiranja ispunjavanje tablica u bazi koristi funkcija koja samostalno generira podatke, rad ove aplikacije temelji se na tome da korisnici sami mogu stvarati oglase. Funkcionalnost korisničkog stvaranja oglasa prikazana je na slici 3.16. kroz funkciju *store*. Na slici je vidljivo kako se koristi *if else* naredba za provjeru kategorije, kako bi korisnik mogao unijeti novu kategoriju ukoliko ona nije na popisu kategorija. Nova kategorija bit će upisana u tablicu *kategorije* u bazi podataka i ona će ubuduće biti prikazana na popisu kategorija. Duga naredba *if elseif* provjerava je li korisnik odabrao akciju ponude ili traženja posla. Ako korisnik nudi posao njemu se omogućava unos adrese na kojoj se on nalazi, što se kod korisnika koji traži posao ne vidi.

```
public function store(Request $request)
{
    if($request->input('kategorija') == 0){
        $kategorija = Kategorije::create([
            'naziv' => $request->input('nova_kategorija')
        ]);
        $kategorijaId = $kategorija->id;
    }else{
        $kategorijaId = $request->input('kategorija');
    }

    if($request->input('vrsta') == 'trazim'){
        NudimPosao::create([
            'naslov' => $request->input('naslov'),
            'kategorija_id' => $kategorijaId,
            'zupanija_id' => $request->input('zupanija'),
            'kontakt' => $request->input('kontakt'),
            'email' => $request->input('email'),
            'opis' => $request->input('opis'),
            'vrijedi_do' => $request->input('vrijedi_do'),
        ]);
    }elseif($request->input('vrsta') == 'nudim'){
        TrazimPosao::create([
            'naslov' => $request->input('naslov'),
            'kategorija_id' => $kategorijaId,
            'zupanija_id' => $request->input('zupanija'),
            'kontakt' => $request->input('kontakt'),
            'email' => $request->input('email'),
            'opis' => $request->input('opis'),
            'vrijedi_do' => $request->input('vrijedi_do'),
        ]);
    }

    return redirect()->route('home');
}
```

Slika 3.16. Prikaz uvjeta za unos podataka

4. KORIŠTENJE WEB APLIKACIJE

U ovom poglavlju je prikazan način korištenja web aplikacije. Izgled naslovne stranice, način korištenja tražilice oglasa, prikaz pretraženih oglasa, stranice za prikaz podataka o oglasu i prikaz izgleda obrasca za izradu novog oglasa.

4.1. Prikaz naslovne stranice

Naslovna stranica kreirana je na način da korisniku omogući što jednostavnije korištenje i pristup željenim dijelovima i informacijama. Slika 4.1. prikazuje naslovnu stranicu i može se uočiti da se naslovna stranica sastoji od slijedećih stavki: navigacijske trake, tražilice oglasa i od prostora za ispis pretraženih oglasa.

Izbornik u navigacijskoj traci sastoji se od poveznica na dijelove web aplikacije:

- *Naslovnica,*
- *O nama,*
- kontrolom *Napravi oglas.*

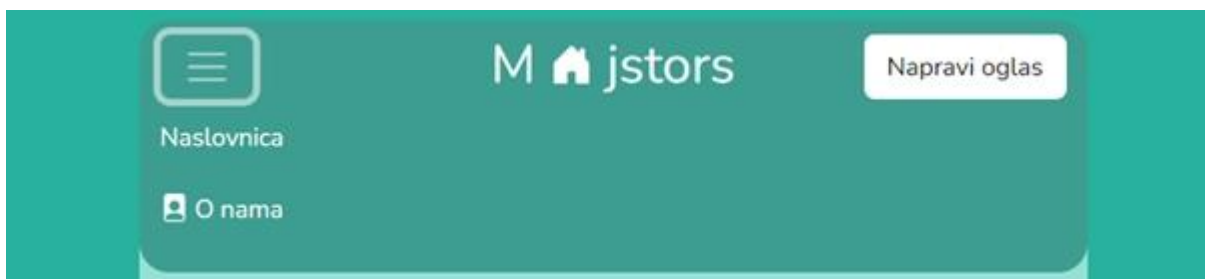


Slika 4.1. Prikaz naslovne stranice Majstors

Slike 4.2. i 4.3. prikazuju izgled navigacijske trake u mobilnom prikazu aplikacije. Navigacijska traka je „collapsable“ odnosno traka se smanji i poveća pritiskom na kontrolu koja je padajući izbornik.



Slika 4.2. Prikaz skupljene navigacijske trake web aplikacije Majstors

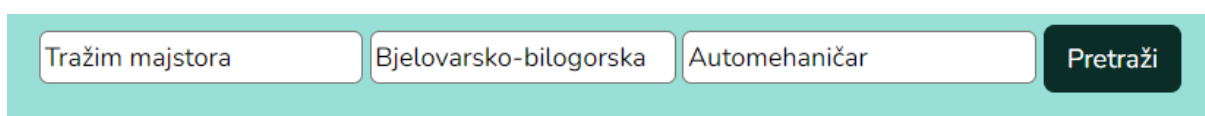


Slika 4.3. Prikaz proširene navigacijske trake web aplikacije Majstors

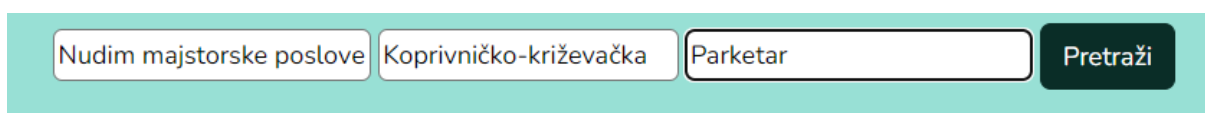
U tijelu aplikacije nalazi se tražilica oglasa s tri padajuća izbornika za odabir naredbe. Prvi izbornik korisniku služi za odabir vrste oglasa koji traži, mogući su slijedeći odabiri: *Tražim majstora* ili *Nudim majstorske poslove*. Drugi izbornik omogućava odabir županije u kojoj korisnik traži ili nudi građevinske radove. Treći izbornik omogućava odabir kategorije/zanimanja za koju je objavljen oglas. Slika 4.4. prikazuje opisani izgled tražilice oglasa, dok je na slikama 4.5. i 4.6. prikazan izgled popunjene tražilice različitim podacima za ispis odgovarajuće grupe oglasa.



Slika 4.4. Prikaz izbornika tražilice oglasa na web aplikaciji



Slika 4.5. Prikaz ispunjenih izbornika od strane korisnika koji potražuje Automehaničara u Bjelovarsko–bilogorskoj županiji



Slika 4.6. Prikaz ispunjenih izbornika od strane majstora koji nudi poslove Parketara u Koprivničko-križevačkoj županiji.

U drugom dijelu tijela aplikacije se ispisuju svi oglasi nakon odabira kontrole *Pretraži*. Oglasi se neće prikazati sve dok se ne odaberu sva tri kriterija i ne pritisne kontrola *Pretraži*. Oglasi se prikazuju kao kartice s osnovnim informacijama: Naslovom oglasa, zanimanje, opis posla i na kraju je kontrola *Prikaži više* koja korisnika šalje na dio aplikacije gdje se mogu vidjeti detaljne informacije pojedinog oglasa. Na slikama 4.7. i 4.8. se vidi ispisa oglasa za odabrane kriterije tražilice popunjene na slikama 4.5. i 4.6.



Slika 4.7. Prikaz ispisa oglasa ponude majstorskih poslova na web aplikaciji



Slika 4.8. Prikaz ispisa oglasa potražnje majstorskih poslova na web aplikaciji

4.2. Prikaz dijela web aplikacije – Novi oglas

Web aplikacija Majstors sastoji se i od dijela koji služi za kreiranje, odnosno izradu novog oglasa. *Novi oglas* dijeli se na dva dijela. Prvi dio je već navedena i prethodno objašnjena navigacijska traka, a drugi dio je tijelo stranice. Tijelo stranice sastoji se od obrasca za izradu novog oglasa. Obrazac za izradu oglasa prikazan je na slici 4.9.

Novi oglas

Naslov oglasa

Tražiš ili nudiš posao ?

Nudim posao

Županija

-

Adresa

Kategorija

-

Email

Kontakt broj

Opis

Opis oglasa

Oglas vrijedi do

dd.mm.gggg.

Predaj oglas

Slika 4.9. Prikaz obrazac za ponudu posla na web aplikaciji Majstors

Obrazac se sastoji od polja za unos: naslova oglasa, opcije nudi li se ili traži posao, županije, kategorije posla, elektroničke pošte za kontakt, telefona za kontakt, opisa, te datumom do kojeg će oglas biti aktivan, dok sa na samom dnu nalazi naredba *Predaj oglas*. Polje *Adresa* prikazuje se ukoliko korisnik koji ispunjava obrazac za kategoriju u polju *Tražiš ili nudiš posao*, primjeni opciju *Nudim posao*, te se navedenom prikazuje novo polje za unos adrese. Na slici 4.10. se vidi navedeno polje unutar obrasca.

Novi oglas

Naslov oglasa

Tražiš ili nudiš posao ?

Nudim posao

Županija

-

Adresa

Kategorija

-

Email

Kontakt broj

Opis

Opis oglasa

Oglas vrijedi do

dd.mm.gggg.

Predaj oglas

Slika 4.10. Prikaz obrazac za ponudu posla na web aplikaciji Majstors

4.3. Prikaz dijela web aplikacije – O nama

Dio web aplikacije *O nama* sastoji se od navigacijske trake i kratkog opisa o web aplikaciji Majstors, tko ju vodi, kada je nastala i koje su osnovne kontakt informacije. Vrlo je bitna jer korisnicima omogućuje brzi dolazak do informacija o administratorima stranice, pri čemu administratori mogu dobiti korisne povratne informacije odnosno reakcije o samom radu stranice, zadovoljstvu/nezadovoljstvu s ponudom/potražnjom ili samim zadovoljstvom izvođenje radova/usluga od strane majstora. Prikaz dijela web aplikacije *O nama* nalazi se na slici 4.11.



Slika 4.11. Prikaz *O nama*

4.4. Prikaz dijela web aplikacije – O oglasu

Ovaj dio web aplikacije se aktivira kada korisnik koji je na naslovnoj stranici pretražio oglase odabere kontrolu *Prikaži više* uz određeni oglas. Prilikom pritiska na kontrolu, otvara se stranica o tome oglasu, gdje korisnik može saznati sve detaljne informacije vezane uz odabrani oglas. Na slikama 4.12. i 4.13. nalazi se prikaz ispisa pojedinosti o oglasima.

Nudim posao - 298

Županija

Koprivničko-križevačka

Adresa

Adresa majstora - 298

Email

Email - 298

Kontakt broj

Kontakt - 298

Opis

Opis posla- 298

Oglas vrijedi do

2022-09-24

Slika 4.12. Prikaz kartice oglasa majstora koji nudi posao

Tražim majstora - 172

Županija

Istarska

Email

Email - 172

Kontakt broj

Kontakt - 172

Opis

Opis posla- 172

Oglas vrijedi do

2023-09-18

Slika 4.13. Prikaz kartice oglasa korisnika koji traži majstora

5. ZAKLJUČAK

Prilikom izrade web aplikacije koja pruža mogućnost ponude i potražnje usluga u građevinarstvu istražene su konkurentne web aplikacije koje postoje na tržištu. Utvrđene se prednosti i nedostaci ovakvih web aplikacija, te se stoga prilikom izrade Majstors web aplikacije vodilo računa da se primjene prednosti, a eliminira što veći broj nedostataka.

Prednosti koje su uključene u samu aplikaciju su jednostavno sučelje s šturo prikazanim sadržajem kako bi web aplikacija bila što jednostavnija za korištenje od strane korisnika. Aplikacija je interaktivna, te je jednostavnija od aplikacija koje imaju slične svrhu, što ju čini bržom i efikasnijom. Aplikacija ima navigacijsku traku s tri poveznice, tražilicu, te prikaz oglasa. Sastoji se od četiri baze podataka od kojih svaka ima svoju svrhu. Za razliku od svoje konkurencije ne traži korisnika da se prijavi ili registrira, već mu daje mogućnost da na svoje oglase postavi vremensko ograničenje.

Ako se pogleda izrađena web aplikacija može se reći da aplikacija posjeduje sve potrebne funkcionalnosti, te stvaranjem veće baze ponuditelja i tražitelja građevinskih usluga može biti podloga za daljnji razvoj aplikacije gdje bi korisnici mogli ocjenjivati pružene usluge, te objavljivati slike izvršenih radova. Stoga se smatra da će budućim razvojem ovakvih web aplikacija biti moguće pronaći kvalitetne izvođače radova. Posljedično tome omogućiti će se ne samo korist za one koji traže usluge ili radove, već i za same izvođače koji će na temelju preporuka graditi svoj imidž odnosno status.

LITERATURA

- [1] Pravimajstor, dostupno na: <https://pravimajstor.hr/> [pristupljeno: 27.06.2022.]
- [2] eMajstor, dostupno na: <https://www.emajstor.hr/> [pristupljeno: 27.06.2022.]
- [3] Workie, dostupno na: <https://www.workie.org/> [pristupljeno: 28.06.2022.]
- [4] Meštar, dostupno na: <https://mestar.hr/> [pristupljeno: 15.08.2023.]
- [5] Ivan Petak "Dinamička web aplikacija za razmjenu usluga.", Fakultet elektrotehnike, računarstva i informacijskih tehnologija Osijek, Osijek, 2017. dostupno na: <https://urn.nsk.hr/urn:nbn:hr:200:349443> [pristupljeno: 25.06.2022.]
- [6] Frontend vs backend, dostupno na: <https://www.geeksforgeeks.org/frontend-vs-backend/> [pristupljeno: 06.09.2023.]
- [7] HTML: HyperText Markup Language, dostupno na: <https://developer.mozilla.org/en-US/docs/Web/HTML> [pristupljeno: 15.09.2023.]
- [8] Jon Duckett "HTML and CSS: Design and Build Websites", <https://wtf.tw/ref/duckett.pdf> [pristupljeno: 15.09.2023.]
- [9] Navbar Bootstrap v5.3. , dostupno na: <https://getbootstrap.com/docs/5.3/components/navbar/> [pristupljeno: 15.09.2023.]
- [10] Worker PNG transparent images, dostupno na: <https://pngtree.com/so/worker> [pristupljeno: 15.09.2023.]
- [11] Bootstrap icons, dostupno na: <https://icons.getbootstrap.com/> [pristupljeno: 15.09.2023.]
- [12] The HTML select element, dostupno na: <https://developer.mozilla.org/en-US/docs/Web/HTML/Element/select> [pristupljeno: 15.09.2023.]
- [13] HTML DOM Document addEventListener(), dostupno na: https://www.w3schools.com/jsref/met_document_addeventlistener.asp [pristupljeno: 15.09.2023.]
- [14] CSS @media Rule, dostupno na: https://www.w3schools.com/cssref/css3_pr_mediaquery.php [pristupljeno: 15.09.2023.]
- [15] What is PHP?, dostupno na: <https://www.php.net/manual/en/intro-what-is.php> [pristupljeno: 15.09.2023.]
- [16] Laravel 10.x, The PHP Framework for Web Artisan, dostupno na: <https://laravel.com/docs/10.x> [pristupljeno: 15.09.2023.]
- [17] SQL FOREIGN KEY Constraint, dostupno na: https://www.w3schools.com/sql/sql_foreignkey.asp [pristupljeno: 15.09.2023.]

[18] Database Seeding, dostupno na:<https://laravel.com/docs/10.x/seeding> [pristupljeno: 15.09.2023.]

SAŽETAK

U ovom radu opisan je tijek izrade web aplikacije za ponudu i potražnju usluga u građevinarstvu. Prikazane su usporedbe sa već postojećim rješenjima i njihovo značenje za izradu u radu prikazanog rješenja. Opisane su tehnologije korištene pri izradi te funkcionalnosti aplikacije. Opisan je prikaz korisničkog sučelja te izgled baze i način upravljanja bazom podataka web aplikacije. Korištene su sljedeće tehnologije: HTML, CSS, JavaScript, Bootstrap, PHP, Laravel, Composer te su korištene sljedeće aplikacije: Visual Studio Code, XAMPP i MySQL sustav za upravljanje bazom podataka.

Ključne riječi: baza podataka, građevinarstvo, HTML, Laravel, web aplikacija

ABSTRACT

Web application for matching construction service providers with demand

This final project outlines the process of developing a web application to facilitate the supply and demand of construction services. It includes a comparison with existing solutions and highlights their significance in shaping the approach presented in this paper. Additionally, the technologies employed for creating the application's functionality are discussed.

The final project also covers the user interface design, database structure, and database management methods. The technologies used in the development include HTML, CSS, JavaScript, Bootstrap, PHP, Laravel, and Composer. The development environment involved Visual Studio Code, XAMPP, and the MySQL database management system.

Key words: construction, database, HTML, Laravel, web application

ŽIVOTOPIS

Luka Valentić rođen je 26. srpnja 1999. u Osijeku. Završio je Osnovnu školu "Augusta Šenoa" u Osijeku nakon čega upisuje "Elektrotehničku i prometnu školu Osijek", smjer Tehničar za računarstvo. Školu završava s odličnim uspjehom i 2018. godine upisuje Fakultet elektrotehnike, računarstva i informacijskih tehnologija Osijek, smjer stručni studij računarstva.

