

# Unapređenje CMS rješenja

---

**Rajkovača, Ivan**

**Master's thesis / Diplomski rad**

**2024**

*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:200:119416>

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

*Download date / Datum preuzimanja:* **2024-10-20**

*Repository / Repozitorij:*

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



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

**Sveučilišni diplomski studij Računarstvo**

**UNAPREĐENJE CMS RJEŠENJA**

**Diplomski rad**

**Ivan Rajkovača**

**Osijek, 2024.**

**Obrazac D1: Obrazac za ocjenu diplomskog rada na sveučilišnom diplomskom studiju**

**Ocjena diplomskog rada na sveučilišnom diplomskom studiju**

<b>Ime i prezime pristupnika:</b>	Ivan Rajkovača
<b>Studij, smjer:</b>	Sveučilišni diplomski studij Računarstvo
<b>Mat. br. pristupnika, god.</b>	D-1158R, 22.10.2020.
<b>JMBAG:</b>	0165075486
<b>Mentor:</b>	izv. prof. dr. sc. Mirko Köhler
<b>Sumentor:</b>	
<b>Sumentor iz tvrtke:</b>	
<b>Predsjednik Povjerenstva:</b>	prof. dr. sc. Krešimir Nenadić
<b>Član Povjerenstva 1:</b>	izv. prof. dr. sc. Mirko Köhler
<b>Član Povjerenstva 2:</b>	izv. prof. dr. sc. Ivica Lukić
<b>Naslov diplomskog rada:</b>	Unapređenje CMS rješenja
<b>Znanstvena grana diplomskog rada:</b>	<b>Informacijski sustavi (zn. polje računarstvo)</b>
<b>Zadatak diplomskog rada:</b>	U radu je potrebno unaprijediti jednostavni CMS koji će korisniku izgenerirati web stranicu na osnovu unesenog sadržaja. Korisnik samo popunjava obrasce i na osnovu njih se kreira sadržaj. Također korisnik može dodavati nove stranice
<b>Datum ocjene pismenog dijela diplomskog rada od strane mentora:</b>	02.09.2024.
<b>Ocjena pismenog dijela diplomskog rada od strane mentora:</b>	Izvrstan (5)
<b>Datum obrane diplomskog rada:</b>	18.9.2024.
<b>Ocjena usmenog dijela diplomskog rada (obrane):</b>	Izvrstan (5)
<b>Ukupna ocjena diplomskog rada:</b>	Izvrstan (5)
<b>Datum potvrde mentora o predaji konačne verzije diplomskog rada čime je pristupnik završio sveučilišni diplomski studij:</b>	26.09.2024.



**FERIT**

FAKULTET ELEKTROTEHNIKE, RAČUNARSTVA  
I INFORMACIJSKIH TEHNOLOGIJA **OSIJEK**

## IZJAVA O IZVORNOSTI RADA

Osijek, 26.09.2024.

**Ime i prezime Pristupnika:**

Ivan Rajkovača

**Studij:**

Sveučilišni diplomski studij Računarstvo

**Mat. br. Pristupnika, godina upisa:**

D-1158R, 22.10.2020.

**Turnitin podudaranje [%]:**

7

Ovom izjavom izjavljujem da je rad pod nazivom: **Unapređenje CMS rješenja**

izrađen pod vodstvom mentora izv. prof. dr. sc. Mirko Köhler

i sumentora

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 pristupnika:

# SADRŽAJ

<b>1. UVOD .....</b>	<b>1</b>
<b>1.1. Zadatak diplomskog rada .....</b>	<b>2</b>
<b>2. ANALIZA POSTOJEĆIH RJEŠENJA .....</b>	<b>3</b>
<b>2.1. Najpopularniji sustavi za upravljanje sadržajem .....</b>	<b>4</b>
2.1.1. WordPress.....	4
2.1.2. Wix .....	5
2.1.3. Shopify .....	5
2.1.4. Joomla! .....	6
<b>2.2. Trenutna verzija jednostavnog CMS-a.....</b>	<b>6</b>
<b>3. TEHNOLOGIJE KORIŠTENE ZA IZRADU APLIKACIJE .....</b>	<b>8</b>
<b>3.1. Visual Studio Code.....</b>	<b>8</b>
<b>3.2. React.....</b>	<b>9</b>
<b>3.3. Express .....</b>	<b>10</b>
<b>4. PROGRAMSKO RJEŠENJE I NAČIN RADA APLIKACIJE .....</b>	<b>12</b>
<b>4.1. Programsko rješenje.....</b>	<b>12</b>
4.1.1. Programsko rješenje na klijentskoj strani .....	12
4.1.2. Programsko rješenje na serveru .....	15
<b>4.2. Način rada aplikacije .....</b>	<b>17</b>
<b>5. ZAKLJUČAK.....</b>	<b>23</b>
<b>LITERATURA .....</b>	<b>24</b>
<b>SAŽETAK.....</b>	<b>25</b>
<b>ABSTRACT .....</b>	<b>26</b>
<b>ŽIVOTOPIS.....</b>	<b>27</b>

# 1. UVOD

Godine 2024. u Republici Hrvatskoj registrirano je više od 150 tisuća obiteljskih poljoprivrednih gospodarstava (OPG-ova). Svi OPG-ovi imaju zajednički cilj – prodati što više svojih proizvoda i ostvariti maksimalan profit. Kako bi dosegli potencijalne kupce, vlasnici OPG-ova koriste različite moderne metode oglašavanja svojih proizvoda i usluga. Osim oglašavanja na društvenim mrežama poput Instagrama i Facebooka, jedan od najučinkovitijih načina predstavljanja usluga i proizvoda je putem web stranice OPG-a. Iako je web stranica često najbolje rješenje za prezentaciju, mnogim vlasnicima OPG-ova postojeće opcije za izradu web stranica su prekomplikirane i oduzimaju previše vremena. Tipične web stranice OPG-ova sadrže relativno jednostavan sadržaj i zahtijevaju samo nekoliko ključnih sekcija, pa bi njihova izrada trebala biti moguća u svega nekoliko klikova.

Sve navedene činjenice čine osnovu motivacije za izradu ovog diplomskog rada, koji nudi jedinstveno i jednostavno korisničko sučelje za izradu web stranica OPG-a. Aplikacija razvijena u sklopu ovog rada predstavlja nadogradnju postojećih rješenja, čime se proces izrade web stranica dodatno pojednostavljuje, a same stranice postaju modernije i brže. Web stranica koju nudi ova aplikacija jednostavna je prezentacijska platforma koja korisnicima pruža pregled svih usluga i proizvoda jednog obiteljskog gospodarstva.

U drugom poglavlju rada nalazi se kratak opis sustava za upravljanje sadržajem (engl. *Content Management System*). [1] Treće poglavlje donosi pregled postojećih rješenja, kao i tehnologija korištenih za izradu aplikacije diplomskog rada. U četvrtom poglavlju opisuje se rad aplikacije i samo programsko rješenje, uključujući dijelove programskog koda i slike aplikacije.

## **1.1. Zadatak diplomskog rada**

U radu je potrebno unaprijediti jednostavni CMS koji će korisniku generirati web stranicu na osnovu unesenog sadržaja. Korisnik samo popunjava obrasce i na osnovu njih se kreira sadržaj. Također korisnik može dodavati nove stranice.

## 2. ANALIZA POSTOJEĆIH RJEŠENJA

Sustav za upravljanje sadržajem, poznat kao CMS (engl. *Content Management System*), je softverska platforma koja omogućuje korisnicima jednostavno kreiranje, upravljanje, uređivanje i objavljivanje digitalnog sadržaja bez potrebe za dubokim tehničkim znanjem ili programiranjem. CMS se najčešće koristi za upravljanje web stranicama, blogovima, e-trgovinama i drugim online sadržajima.

CMS funkcionira kao posrednik između korisnika i složenih tehničkih zadataka vezanih uz izradu i održavanje web stranica. U suštini, CMS se sastoji od dva osnovna dijela:

1. Administrativno sučelje: Ovo je dio sustava kojem pristupaju administratori i urednici kako bi kreirali, uređivali i organizirali sadržaj. Sučelje obično uključuje tekstualne editore, upravljanje kategorijama, dodavanje meta podataka, itd.
2. Korisničko sučelje: Ovo je dio koji posjetitelji vide. Sadržaj koji se kreira i uređuje u administrativnom sučelju automatski se prikazuje na korisničkom sučelju u skladu s unaprijed definiranim dizajnom i strukturom.

CMS se koristi za niz različitih svrha, uključujući:

- Web stranice i blogove: Omogućuje jednostavno kreiranje i održavanje web stranica, često s integriranim blogovima, forumima i kontakt formama.
- E-trgovine: CMS platforme omogućuju postavljanje online trgovina s alatima za upravljanje proizvodima, narudžbama, plaćanjima i inventarom.
- Portali i vijesti: Sustavi za upravljanje sadržajem koriste se za vođenje kompleksnih web portala i novinskih web stranica s redovitim ažuriranjima i velikim količinama sadržaja.

CMS posjeduje brojne prednosti u odnosu na druga tehnička rješenja, neke od njih su:

- Jednostavnost: Većina CMS-ova je dizajnirana tako da ih mogu koristiti i korisnici bez programerskog znanja.
- Fleksibilnost: CMS-ovi omogućuju lako dodavanje novih funkcionalnosti putem dodataka ili modula.
- SEO-pristupačnost: Mnogi CMS-ovi nude alate za optimizaciju sadržaja za tražilice, što može pomoći u poboljšanju vidljivosti web stranice.
- Skalabilnost: CMS platforme mogu rasti zajedno s vašim potrebama, omogućujući jednostavno dodavanje novih stranica, proizvoda ili funkcionalnosti.



Ukratko, sustavi za upravljanje sadržajem igraju ključnu ulogu u modernom digitalnom svijetu, omogućujući korisnicima jednostavno i učinkovito upravljanje njihovim sadržajem na internetu.

## **2.1. Najpopularniji sustavi za upravljanje sadržajem**

U modernom dobu internet je preplavljen brojnim alatima i programskim rješenjima s pomoću kojim je moguće generirati web stranicu. Najveće mane svih trenutno dostupnih alata su složenost i cijena. Većina dostupnih alata zahtijeva određenu količinu informatičkog znanja kako bi se izradila web stranica. Također, većina istih alata nije besplatna i često je potrebno izdvojiti određenu svotu novca za izradu čak i najjednostavnije web stranice, kao i za njeno održavanje.

Zbog svih gore navedenih razloga vlasnici OPG-ova često odustaju od samostalne izrade web stranice, pa se neki odlučuju i platiti nekom drugom da im izradi web stranicu, što je u većini slučajeva preskup i nepotreban trošak jer ipak se radi o zaista jednostavnim web stranicama.

Uz sve dostupne alate i tehnologije ipak se izdvajaju četiri najpopularnija alata. Trenutno najpopularniji CMS alati su WordPress, Wix, Shopify i Joomla!. Svi navedeni alati su već godinama prisutni na internetu i svake godine postaju sve moderniji i sofisticiraniji pa ne čudi da se toliko korisnika odlučuje koristiti ih za potrebe izrade web stranica.

U nastavku ovog poglavlja predstavljen je kratki opis najpopularnijih CMS alata kao i njihove prednosti odnosno nedostatci.

### **2.1.1. WordPress**

WordPress je najpopularniji sustav za upravljanje sadržajem (CMS) na svijetu, koji pokreće više od 40 % svih web stranica. [2] Prvobitno je razvijen kao platforma za blogove, no s vremenom se razvio u moćan alat za izradu svih vrsta web stranica, od jednostavnih blogova do složenih korporativnih stranica i e-trgovina.

WordPress je platforma otvorenog koda (engl. *open-source*), što znači da je besplatna za korištenje, a korisnici je mogu prilagoditi i proširiti koristeći razne dodatke (engl. *plugins*) i teme. Instalacija je jednostavna, a upravljanje sadržajem se odvija putem intuitivnog korisničkog sučelja. WordPress podržava razne vrste sadržaja, uključujući blog postove, stranice, galerije i proizvode za e-trgovinu.

Glavne prednosti WordPress platforme su fleksibilnost, optimizacija za tražilice i velika zajednica, ima ogromnu bazu korisnika i programera koji pružaju podršku i razvijaju nove funkcionalnosti.

Glavni nedostatak WordPress-a je njegova kompleksnost, zahtijeva veći količinu informatičkog znanja kako bi se izradila web stranica. Također, radi se o zastarjeloj tehnologiji koja zbog prevelike količine dodataka često rezultira sporim web stranicama i lošim korisničkim iskustvom.

### **2.1.2. Wix**

Wix je platforma za izradu web stranica koja omogućuje korisnicima da jednostavno kreiraju i uređuju web stranice pomoću vrlo jednostavnog sučelja. [3] Namijenjen je korisnicima koji nemaju tehničko znanje, a žele brzo i lako postaviti web stranicu.

Wix nudi veliki broj predložaka koje korisnici mogu prilagoditi prema svojim potrebama. Web stranica se izrađuje putem vizualnog sučelja gdje se elementi povlače i ispuštaju na željeno mjesto. Wix također uključuje hosting, što znači da se korisnici ne trebaju brinuti o tehničkim detaljima vezanim uz postavljanje web stranice.

Glavna prednost je jednostavnost korištenja, također Wix nudi sve u jednom, hosting, domena sigurnost sve to je obuhvaćeno i nije korisnikova briga. Neki od nedostataka su nedostatak prilagodljivosti i proširenja, a za sve dodatne opcije potrebno je pretplatiti se na neki od skupljiv planova korištenja.

### **2.1.3. Shopify**

Shopify je vodeća platforma za e-trgovinu koja omogućuje korisnicima izradu online trgovina bez potrebe za naprednim tehničkim znanjem. [4] Popularan je među malim i srednjim poduzetnicima koji žele brzo pokrenuti i upravljati vlastitom online trgovinom.

Shopify omogućuje korisnicima postavljanje online trgovine kroz nekoliko jednostavnih koraka. Platforma nudi razne alate za upravljanje proizvodima, narudžbama, plaćanjima i dostavom. Također, korisnici mogu birati između različitih predložaka za dizajn trgovine, a dodatne funkcionalnosti mogu se dodati putem aplikacija iz Shopify trgovine.

Prednosti su jednostavnost korištenja, integracija plaćanja i stalno dostupna korisnička podrška kao i baza za učenje. Neki od nedostataka uključuju potencijalno visoke troškove korištenja, ograničenu prilagodljivost kao i transakcijske naknade.

#### **2.1.4. Joomla!**

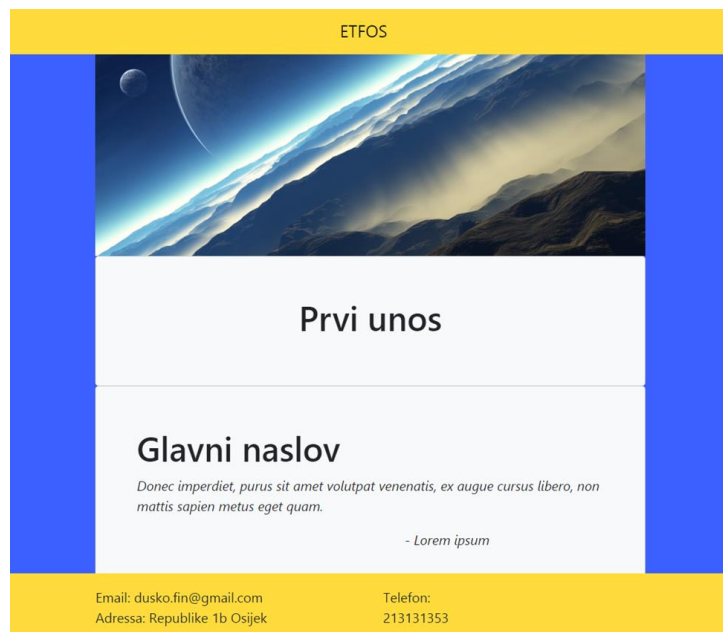
Joomla! je CMS koji nudi odličnu ravnotežu između fleksibilnosti i jednostavnosti korištenja. [5] Namijenjen je korisnicima koji žele više mogućnosti prilagodbe, ali ne žele previše tehničkih komplikacija. Joomla! je popularna među korisnicima koji žele izraditi složenije web stranice s više funkcionalnosti.

Joomla! omogućuje izradu i upravljanje web stranicama putem administracijskog sučelja koje je složenije od onog kod WordPressa, ali još uvijek dovoljno intuitivno za korisnike s osnovnim tehničkim znanjem. Joomla! podržava više vrsta sadržaja, uključujući članke, forume, korisničke profile i slično. Kao i WordPress, Joomla! omogućuje proširenje funkcionalnosti putem dodataka i tema.

Prednosti uključuju fleksibilnost, podršku za više jezika te prilagodljivost. Nedostatci su kompleksnost korištenja i manja zajednica u kojoj je ponekad teško naći rješenje za neki problem ili resurse za učenje.

## **2.2. Trenutna verzija jednostavnog CMS-a**

Kao jedno od postojećih rješenja važno je navesti rješenje koje je potrebno unaprijediti u ovom diplomskom radu. Na slici 2.1. je vidljiva web stranica generirana korištenjem postojeće aplikacije. Za izradu postojećeg rješenja uvelike je korišten programski jezik PHP i njegov mrežni okvir Laravel. PHP programski jezik se smatra relativno zastarjelom tehnologijom, a tako dobivene web stranice ne odgovaraju modernim standardima u pogledu performansi i korisničkog iskustva. Postojeće rješenje je funkcionalno i zadovoljava potrebe vlasnika OPG-a, ali ga je potrebno unaprijediti korištenjem novijih tehnologija i web stranicama modernijeg izgleda.



Sl. 2.1. CMS usmjeren na sadržaj

### 3. TEHNOLOGIJE KORIŠTENE ZA IZRADU APLIKACIJE

U ovom poglavlju opisane su tehnologije korištene prilikom izrade diplomskog rada. Za svaku tehnologiju pružen je kratki opis te su navedene njihove prednosti i mane.

#### 3.1. Visual Studio Code

Visual Studio Code, poznatiji kao VS Code je besplatno razvojno okruženje koje je razvijeno od strane Microsofta. [6] Dizajniran je za rad na više platformi, pa se može koristiti na Windows, macOS i Linux operativnim sustavima. Iako je lagan, VS Code je vrlo moćan i nudi brojne funkcionalnosti koje programeri trebaju za efikasno pisanje, uređivanje i debugiranje koda. Izgled sučelja može se vidjeti na slici 3.1.

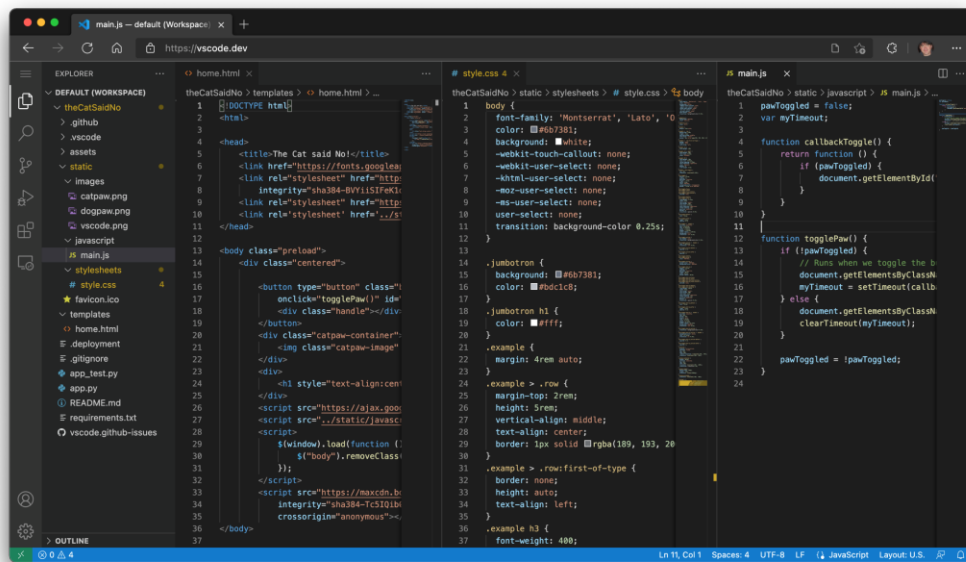
Visual Studio Code je tekstualni editor optimiziran za pisanje i uređivanje koda. Radi s velikim brojem programskih jezika, uključujući JavaScript, Python, C++, Java, Go i mnoge druge. Osnovne funkcionalnosti VS Code-a uključuju automatsko dovršavanje koda, sintaksno bojanje, napredne mogućnosti pretraživanja i zamjene, te integrirano terminalsko sučelje.

VS Code je stekao veliku popularnost među programerima iz nekoliko ključnih razloga:

- Prilagodljivost: VS Code nudi širok spektar proširenja koja omogućuju korisnicima da prilagode uređivač svojim specifičnim potrebama i radnim tokovima.
- Brzina i učinkovitost: Iako je bogat značajkama, VS Code je vrlo brz i optimiziran, što omogućuje glatko i brzo uređivanje i pokretanje koda.
- Integracija s Git-om: Izravna integracija s Git-om i drugim sustavima za kontrolu verzija omogućuje jednostavno praćenje promjena, rad na granama (branches) i upravljanje repozitorijima izravno iz uređivača.
- Cross-platform podrška: Mogućnost rada na različitim operativnim sustavima čini ga dostupnim širokom krugu korisnika, bez obzira na platformu koju koriste.
- Cijena: Kao besplatan alat s otvorenim kodom, VS Code je dostupan svim programerima, a zajednica doprinosi stalnim poboljšanjima i novim značajkama.

- Aktivna zajednica i podrška: VS Code ima veliku i aktivnu zajednicu koja redovito razvija nova proširenja, piše dokumentaciju i nudi podršku.

Ovi faktori čine Visual Studio Code jednim od najčešće korištenih alata među programerima širom svijeta, bez obzira na to rade li na malim osobnim projektima ili velikim industrijskim aplikacijama.



Sl. 3.1. Visual Studio Code

## 3.2. React

React proširenje JavaScript programskog jezika razvijeno od strane Facebooka, koje služi za izradu korisničkih sučelja, posebno onih koja zahtijevaju visoku interaktivnost i brzinu. Prvotno je lansiran 2013. godine, a od tada je postao jedan od najpopularnijih alata za razvoj modernih web aplikacija. [7]

React se koristi prvenstveno za razvoj složenih, dinamičnih web aplikacija koje zahtijevaju brzinu i interaktivnost. S obzirom na to da omogućuje jednostavno stvaranje komponenti koje se mogu ponovno koristiti, React je idealan za projekte u kojima se trebaju često ažurirati ili mijenjati korisnički elementi, poput društvenih mreža, e-commerce platformi ili aplikacija za upravljanje podacima.

React funkcionira kroz koncept komponenti. Komponente su male, neovisne i ponovno iskoristive jedinice koda koje opisujemo kao funkcije ili klase. Svaka komponenta može upravljati vlastitim

stanjem (engl. *state*) i prikazivati određeni dio korisničkog sučelja. React koristi tzv. "virtualni DOM" (engl. *Document Object Model*), što omogućuje brzo ažuriranje samo onih dijelova korisničkog sučelja koji su promijenjeni, bez potrebe za ponovnim prikazivanjem cijele stranice. Ovo značajno poboljšava performanse aplikacija.

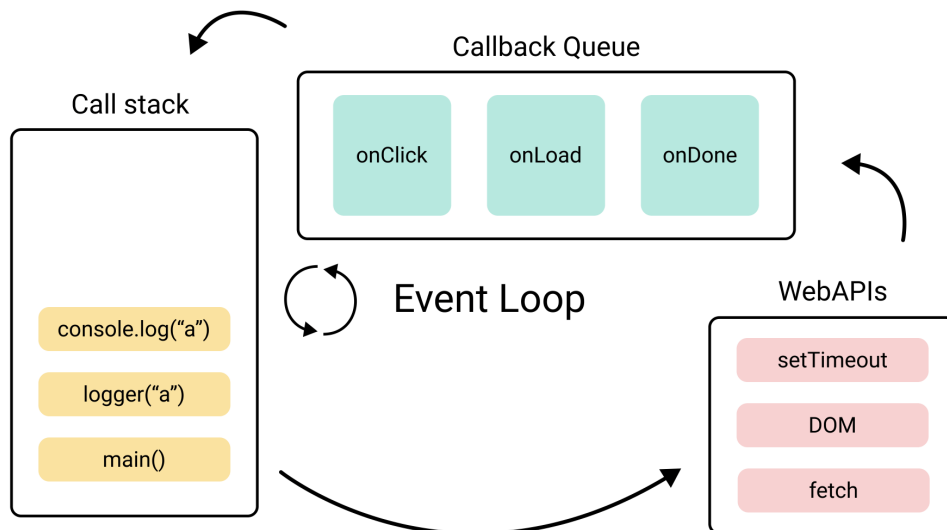
Zbog svoje jednostavnosti i modernog načina rada React je odabran kao glavna tehnologija za izradu korisničkog sučelja ovog rada. Također, gotova proširenja za rad s formama su uvelike olakšali izradu web aplikacije i poboljšali sigurnost iste.

### **3.3. Express**

Express.js je proširenje NodeJS-a, dizajnirano za izgradnju robusnih web aplikacija i API-ja. [8] Express postoji od 2010. godine i brzo je postao jedan od najpopularnijih NodeJS biblioteka, zahvaljujući svojoj jednostavnosti. NodeJS omogućuje izvršavanje JavaScript programskog koda na serveru, što omogućuje programerima koji su upoznati s JavaScript jezikom da vrlo jednostavno pišu i backend kod. Express.js se koristi za izgradnju web aplikacija i API-ja, omogućujući programerima da lako postavljaju rute, upravljaju zahtjevima i odgovorima.

Express.js radi na način da pruža sloj apstrakcije iznad osnovnih Node.js značajki. Omogućuje programerima jednostavno definiranje ruta i specificiranje što bi aplikacija trebala učiniti kada korisnik pristupi tim URL-ovima. To uključuje rukovanje HTTP metodama (GET, POST, PUT, DELETE) i odgovaranje na zahtjeve s odgovarajućim podacima, HTML stranicama ili porukama o grešci.

Express.js je izgrađen na Node.js-u, koji koristi mehanizam petlje događaja (engl. *event loop*). Event loop je središnji dio Node.js-a koji omogućuje asinkrono rukovanje operacijama. Umjesto da čeka na završetak dugotrajnih operacija (poput pristupa bazi podataka ili HTTP zahtjeva), Node.js može nastaviti obavljati druge zadatke. Kada se dugotrajna operacija završi, odgovarajući povratna funkcija se stavlja u red za obradu i izvršava kada dođe na red. Način rada petlje prikazan je na slici 3.2.



Sl. 3.2. NodeJS petlja



## 4. PROGRAMSKO RJEŠENJE I NAČIN RADA APLIKACIJE

U ovom poglavlju opisan je način rada aplikacije, odnosno koraci koje korisnik mora proći kako bi generirao web stranicu za svoje obiteljsko poljoprivredno gospodarstvo. Također, priloženo je programsko rješenje za korisničko sučelje kao i za serverski dio aplikacije.

### 4.1. Programsko rješenje

#### 4.1.1. Programsko rješenje na klijentskoj strani

Korisničko sučelje aplikacije izrađeno je pomoću JavaScript biblioteke React. React je proširenje JavaScript jezika koje omogućuje pisanje reaktivnih komponenti koje zajedno tvore sadržaj aplikacije. Na taj način se proces pisanja koda znatno ubrzava, komponente se nižu jedna na drugu po potrebi i mogu se ponovo iskoristiti. Kako bi aplikacija dobila moderan izgled korištena je biblioteka MaterialUI. MaterialUI je dizajn sustav koji se sastoji od klasičnih web komponenti koje su moderno stilizirane. Time se dodatno ubrzava proces razvoja aplikacije, nije potrebno trošiti vrijeme na uređivanje komponenti i nije potreban dizajn za svaku komponentu. Primjer korištenih komponenti se može vidjeti na slici 4.1. Također, za validaciju forme korištena je biblioteka react-hook-form koja se sastoji od brojnih funkcija koje nam pomažu pri validaciji. Stoga, pisanje koda se sastoji od nizanja već gotovih komponenti i njihovog povezivanja sa procesom za validaciju podataka koje je korisnik upisao. Najvažnija funkcija u aplikacije je funkcija za podnošenje (engl. *submit*) forme. U toj funkciji potrebno je prikupiti sve podatke iz forme te ih prebaciti u oblik pogodan za slanje na server. Na slici 4.2. se nalazi prvi dio submit funkcije. S obzirom da je potrebno slati slike na server, koristi se FormData klasa koja olakšava kreiranje objekta u obliku ključ, vrijednost gdje ključ predstavlja polje u formi a vrijednost ono što je uneseno u to polje. Takav objekt je pogodan za slanje na server, a slike se tada pošalju u pravilnom obliku kao binarni elementi. Drugi dio submit funkcije predstavlja proces slanja podataka na server pomoću fetch funkcije JavaScript programskog jezika. HTTP standardom definirana je POST metoda kojom se podaci šalju na server. Ukoliko je slanje uspješno, na klijentskoj strani se kreira poveznica za preuzimanje generiranog HTML dokumenta. Proces slanja može se vidjeti na slici 4.4. Time je proces završen na klijentskoj strani, a generirani HTML dokument se preuzima na računalo korisnika.

```

<Box
  display="flex"
  justifyContent="center"
  alignItems="center"
  gap={1}>
  <label htmlFor="images">
    <img
      src={fileAttachmentIcon}
      style={{ width: "24px", height: "24px", cursor: "pointer" }}
    />
  </label>
  <Typography variant="subtitle2">
    Fotografije vašeg obrta (maksimalno 3)
  </Typography>

  <input
    type="file"
    id="images"
    accept="image/png, image/gif, image/jpeg"
    multiple
    disabled={isLoading}
    {...register("images", {
      validate: (value) => {
        if (value.length === 0) {
          return "Fotografija proizvoda je obavezna";
        }
      },
    })}
    onChange={onImagesChange}
    style={{
      display: "none",
    }}
  />
  <Typography variant="subtitle2" color="error">
    {errors.images && <span>{errors.images.message}</span>}
  </Typography>
</Box>

```

Sl. 4.1. Komponente sučelja

```

const onSubmit: SubmitHandler<IFormInput> = async (data) => {
  const formData = new FormData();

  formData.append("title", data.title);
  formData.append("description", data.description);
  formData.append("product1Title", data.product1Title);
  formData.append("product1Description", data.product1Description);
  formData.append("product2Title", data.product2Title);
  formData.append("product2Description", data.product2Description);
  formData.append("product3Title", data.product3Title);
  formData.append("product3Description", data.product3Description);
  formData.append("phoneNumber", data.phoneNumber);
  formData.append("email", data.email);
  formData.append("facebookUrl", data.facebookUrl);
  formData.append("instagramUrl", data.instagramUrl);
  formData.append("location", data.location);
  formData.append("theme", data.theme);
  formData.append("street", data.street);
  formData.append("zipCode", data.zipCode);

  if (data.images && data.images.length > 0) {
    data.images.forEach((image, index) => {
      formData.append(`images${index}`, image);
    });
  }

  if (data.logo) {
    formData.append("logo", data.logo);
  }

  if (data.product1Image) {
    formData.append("product1Image", data.product1Image);
  }
  if (data.product2Image) {
    formData.append("product2Image", data.product2Image);
  }
  if (data.product3Image) {
    formData.append("product3Image", data.product3Image);
  }
}

```

Sl. 4.2. Popunjavanje FormData objekta

```

try {
  setIsLoading(true);
  const response = await fetch(
    "http://localhost:8000/api/generate-website",
    {
      method: "POST",
      body: formData,
    }
  );

  if (!response.ok) {
    throw new Error("Network response was not ok");
  }

  const html = await response.text();
  const blob = new Blob([html], { type: "text/html" });
  const url = URL.createObjectURL(blob);

  const a = document.createElement("a");
  a.href = url;
  a.download = "index.html";
  document.body.appendChild(a);
  a.click();

  document.body.removeChild(a);
  URL.revokeObjectURL(url);
} catch (error) {
  console.error("Error:", error);
} finally {
  setIsLoading(false);
}

```

Sl. 4.3. Poziv na server

```

try {
  setIsLoading(true);
  const response = await fetch(
    "http://localhost:8000/api/generate-website",
    {
      method: "POST",
      body: formData,
    }
  );

  if (!response.ok) {
    throw new Error("Network response was not ok");
  }

  const html = await response.text();
  const blob = new Blob([html], { type: "text/html" });
  const url = URL.createObjectURL(blob);

  const a = document.createElement("a");
  a.href = url;
  a.download = "index.html";
  document.body.appendChild(a);
  a.click();

  document.body.removeChild(a);
  URL.revokeObjectURL(url);
} catch (error) {
  console.error("Error:", error);
} finally {
  setIsLoading(false);
}

```

Sl. 4.4. Poziv na server

#### 4.1.2. Programsko rješenje na serveru

Na serverskoj strani korištena je biblioteka Express, koja pruža gotove funkcije za kreiranje API-ja. Korištenjem Express biblioteke uvelike je smanjena količina koda koja je potrebna za implementaciju servera. Kako bi server mogao čitati podatke koji dolaze iz forme potrebno je koristiti biblioteku multer. [9] Multer omogućuje rad sa datotekama koje se nalaze u formi, odnosno u ovom slučaju sa fotografijama koje je korisnik odabrao za web stranicu OPG-a. Zatim, kako bi kreirali HTML stranicu na serveru potrebno je koristiti neku od biblioteka za kreiranje predložaka (engl. *templates*). Za potrebe ove aplikacije odabrana je biblioteka ejs, koja se smatra najpopularnijim alatom za kreiranje HTML dokumenata na dinamički način, odnosno iz podataka neke forme. [10] Ejs predložak je zapravo HTML kod koji je proširen dinamičkim poljima čija vrijednost ovisi o onome što je korisnik u formi upisao. Na slici 4.5. može se vidjeti način na koji se dobivaju podaci iz forme i umeću u Ejs predložak. Web aplikacija vrši poziv na rutu generate-website i šalje podatke iz forme, a na toj ruti server poziva funkciju za generiranje HTML dokumenta sa istim podacima. S obzirom da slike nije potrebno spremati, aplikacija ne koristi bazu podataka, a slike su prikazane kao binarni niz znakova.

```

app.post("/api/generate-website", uploadFields, (req, res) => {
  res.render(req.body.theme, {
    ...req.body,
    ...(req.files
      ? {
        images0: req.files["images0"]
          ? req.files["images0"][0].buffer.toString("base64")
          : null,
        images1: req.files["images1"]
          ? req.files["images1"][0].buffer.toString("base64")
          : null,
        images2: req.files["images2"]
          ? req.files["images2"][0].buffer.toString("base64")
          : null,
        logo: req.files["logo"]
          ? req.files["logo"][0].buffer.toString("base64")
          : null,
        product1Image: req.files["product1Image"]
          ? req.files["product1Image"][0].buffer.toString("base64")
          : null,
        product2Image: req.files["product2Image"]
          ? req.files["product2Image"][0].buffer.toString("base64")
          : null,
        product3Image: req.files["product3Image"]
          ? req.files["product3Image"][0].buffer.toString("base64")
          : null,
      }
      : {}),
  });
});

```

Sl. 4.5. Kod za generiranje HTML dokumenta na serveru

Na slici 4.6. vidi se primjer popunjavanja Ejs predložka sa podacima iz forme. Dostupne su gotove direktive u koje je potrebno upisati ime polja iz forme, a sadržaj se tada automatski popunjava validnim HTML kodom.

```

<p class="w3-text-blue-grey">
  <!-- Adresa opga -->
  <div>
    <b class="w3-text-blue-grey">Mjesto:</b><span><%=location%></span>
  </div>
  <div>
    <b class="w3-text-blue-grey ">Ulica i kućni broj:</b> <span><%=street%></span>
  </div>
  <div>
    <b class="w3-text-blue-grey ">Poštanski broj:</b><span ><%=zipCode%></span>
  </div>

```

Sl. 4.6. Ejs predložak

S obzirom da svaki predložak ima kontakt formu, neovisno o temi koju je korisnik odabrao, na serveru je potrebno implementirati rukovanje sa kontakt formom. Kako bi kontakt forma radila potrebno je koristiti neku od biblioteka za rad sa email klijentima. Za potrebe ovog rada korišten je nodemailer. Korištenjem nodemailer biblioteke slanje emaila postaje jednostavan poziv na odvojenu rutu. Na slici 4.7. može se vidjeti način na koji se upit iz kontakt forme šalje na email adresu vlasnika OPG-a.

```

app.post("/api/send-email", (req, res) => {
  const mailOptions = {
    from: req.body.contactEmail,
    to: req.body.websiteEmail,
    subject: "OPG-UPIT",
    text: `${req.body.contactMessage}\n\nFrom: ${req.body.contactEmail}`,
    html: `<html><body>${req.body.contactMessage}<br><br><b>From: ${req.body.contactEmail}</b></body></html>`,
  };

  transporter.sendMail(mailOptions, (error, info) => {
    if (error) {
      console.log(error);
      res.status(500).send("Something went wrong.");
    } else {
      console.log("Email sent: " + info.response);
      res.status(200).send("Email sent.");
    }
  });
});

```

Sl. 4.7. Slanje emaila

## 4.2. Način rada aplikacije

S obzirom da je zadatak ovog diplomskog rada poboljšati već postojeće rješenje za izradu web stranice OPG-a, odabran je najjednostavniji način rada aplikacije. Aplikacija radi na način da korisnik popuni formu sa svim potrebnim podacima, te pritiskom gumba preuzme datoteku, odnosno HTML dokument za web stranicu. Forma se sastoji od četiri dijela. U prvom dijelu korisnik popunjava osnovne informacije o svom gospodarstvu što uključuje naziv obrta, kratak opis obrta, logotip obrta i nekoliko fotografija obrta. Sva polja u formi su obavezna i implementirana je validacija koja sprječava korisnika da obavi poziv na server dok nije popunio sve podatke. Polja su obavezna da bi se izbjegli prazni dijelovi web stranice, čime se dobiva moderni izgled i najbolje korisničko iskustvo. Drugi dio forme bavi se proizvodima, gdje je potrebno upisati naziv proizvoda, kratki opis te fotografiju za svaki proizvod. U formu je moguće upisati tri proizvoda. U trećem dijelu forme korisnik upisuje podatke za kontakt i lokaciju svog obrta. Za kontakt su odabrana polja za broj telefona, email te Facebook i Instagram poveznica na profil OPG-a. Ovdje je od posebne važnosti polje za email zbog toga što se na svim temama nalazi kontakt forma u kojoj kupac može poslati upit na email OPG-a. Osim polja za kontakt, treći dio forme uključuje i polja za lokaciju. Lokacija obrta je definirana mjestom, poštanskim brojem te ulicom i kućnim brojem. U zadnjem dijelu forme korisnik aplikacije odabire temu za web stranicu

koja će se generirati pritiskom na gumb „Preuzmi datoteke“. Teme su modernog izgleda i moguće je birati između tri različite teme. Teme se uglavnom razlikuju u nijansama boja i rasporedu sadržaja, dok je sadržaj svih tema isti i uključuje sve podatke unesene u formi. Izgled aplikacije vidljiv je na slikama 4.8. i 4.9.

# OPG CMS

Ispunite formu i preuzmite sve potrebne datoteke za izradu web stranice

## Osnovne informacije

Naziv obrta

OPG Med

Opis obrta

Lorem ipsum odor amet, consectetur adipiscing elit. Quam montes hendrerit congue volutpat finibus magna. Maecenas feugiat mus tempor velit justo. Class tortor donec magnis lacinia orci porta natoque facilisi. Molestie inceptos adipiscing mi arcu dictumst. Aphasellus at parturient nascetur est.

Logo obrta



Fotografije vašeg obrta (maksimalno 3)



## Proizvod 1

Naziv proizvoda

Bagremov med

Opis proizvoda

Suspendisse pulvinar nostra fermentum himenaeos ante amet augue. Condimentum arcu varius habitant diam dapibus enim natoque ultricies. Consequat aliquam mollis lacinia orci, consequat parturient sed ipsum. Aliquam maecenas torquent risus varius integer nunc.

Fotografija proizvoda




## Proizvod 2

Naziv proizvoda

Lipov med

Sl. 4.8. Prvi dio forme







**Proizvod 3**

Naziv proizvoda  
Cvjetni med

Opis proizvoda  
Dignissim purus magna facilisis nisi mi euismod justo. Pulvinar integer duis senectus pharetra varius. Malesuada cras odio adipiscing maecenas; rutrum ultrices blandit.

 Fotografija proizvoda



**Kontakt i lokacija**

Broj telefona  
0957305171

Email  
ivanrajkovaca@hotmail.com


Facebook link  
<https://www.facebook.com/ivanrajkovaca>

Instagram link  
<https://www.instagram.com/irajkovacaa>


Mjesto  
Donja Vrba

Ulica i kućni broj  
Miroslava Naletilića 28

Poštanski broj  
35208



Prva tema



Druga tema



Treća tema

[PREUZMI DATOTEKE](#)

Sl. 4.9. Drugi dio forme

Klikom na gumb „Preuzmi datoteke“ šalje se upit na server sa svim podacima iz forme. Na serveru se podaci iz forme obrađuju i upisuju u HTML dokument ovisno o temi koju je korisnik u formi odabrao. Zatim se generirani HTML dokument preuzima sa servera na računalo korisnika. HTML dokument je potpuno funkcionalan i spreman za postavljanje na server, odnosno na domenu koju je korisnik odabrao za svoju web stranicu. Na slikama 4.10. , 4.11. i 4.12. se mogu vidjeti svi dijelovi web stranice generirane odabirom prve teme. Na slici 4.13. se nalazi prikaz početne stranice druge teme, dok je na slici 4.14. početna stranica treće teme.



## O nama

*Saznajite više o nama*

Lorem ipsum odor amet, consectetur adipiscing elit. Quam montes hendrerit congue volutpat finibus magna. Maecenas feugiat mus tempor velit justo. Class tortor donec magnis lacinia orci porta natoque facilisi. Molestie inceptos adipiscing mi arcu dictumst. Aphasellus at parturient nascetur est.

### Sl. 4.10. Početna stranica prve teme

## Naši proizvodi

### Bagremov med

Suspendisse pulvinar nostra fermentum himenaeos ante amet augue. Condimentum arcu varius habitant diam dapibus enim natoque ultricies. Consequat aliquam mollis lacinia orci, consequat parturient sed ipsum. Aliquam maecenas torquent risus varius integer nunc.



### Lipov med

Placerat dignissim dignissim ligula dapibus ligula malesuada. Vitae dolor ad quisque dolor tortor suscipit sollicitudin parturient conubia. Convallis turpis purus porttitor nisi diam.



### Cvjetni med

Dignissim purus magna facilisis nisi mi euismod justo. Pulvinar integer duis senectus pharetra varius. Malesuada cras odio adipiscing maecenas; rutrum ultrices blandit.



### Sl. 4.11. Proizvodi prva tema

## KONTAKT

Možete nas kontaktirati i telefonom na broj 0957305171 ili putem e-pošte na [ivanrajkovaca@hotmail.com](mailto:ivanrajkovaca@hotmail.com). Također, možete nam poslati poruku ovdje:

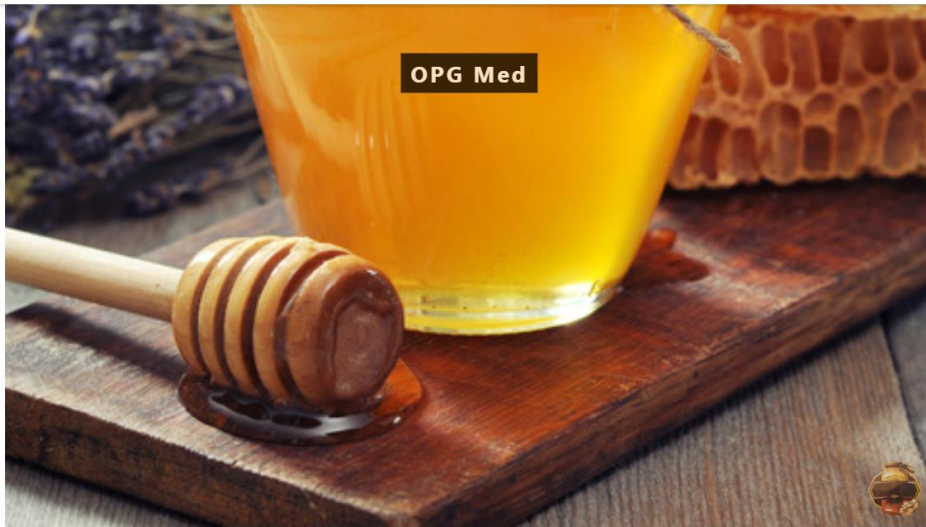
Donja Vrba, Miroslava Naletilića 28, 35208

Phone: 0957305171

Email: [ivanrajkovaca@hotmail.com](mailto:ivanrajkovaca@hotmail.com)

POŠALJI UPIT

### Sl. 4.12. Kontakt forma prva tema



### O nama

Lorem ipsum odor amet, consectetur adipiscing elit. Quam montes hendrerit congue volutpat finibus magna. Maecenas feugiat mus tempor velit justo. Class tortor donec magnis lacinia orci porta natoque facilisi. Molestie inceptos adipiscing mi arcu dictumst. Aphasellus at parturient nascetur est.

Sl. 4.13. Početna stranica druga tema



### O nama

#### Upoznajte naš obrt

Lorem ipsum odor amet, consectetur adipiscing elit. Quam montes hendrerit congue volutpat finibus magna. Maecenas feugiat mus tempor velit justo. Class tortor donec magnis lacinia orci porta natoque facilisi. Molestie inceptos adipiscing mi arcu dictumst. Aphasellus at parturient nascetur est.

Sl. 4.14. Početna stranica treća tema

## 5. ZAKLJUČAK

Glavni zadatak ovog diplomskog rada bio je pojednostaviti i unaprijediti već postojeće rješenje za kreiranje web stranice obiteljskog poljoprivrednog gospodarstva. Unaprjeđenje je postignuto korištenjem modernijih tehnologija, jednostavnijim korisničkim iskustvom kao i modernijim izgledom generiranje web stranice. S obzirom da se tehnologije za izradu web stranica konstantno mijenjaju, potrebno je pratiti trendove i koristiti najmodernija dostupna rješenja čime se postiže bolja konkurentnost na tržištu kao i bolje korisničko iskustvo. Express i React su među najpopularnijim tehnologijama za izradu web stranica i web aplikacija, stoga su odabrane kao tehnološka rješenja za ovaj rad.

Zadatak rada je ispunjen, odrađeni su svi traženi koraci i postignuto je unaprjeđenje u odnosu na prethodna rješenja za isti problem.

S obzirom da se radi o relativno jednostavnoj aplikaciji, dobiveno rješenje je moguće proširiti na više načina. Formu je moguće proširiti dodatnim poljima koja bi koristila vlasnicima OPG-ova, također može se koristiti baza podataka i sustav za autorizaciju korisnika. Osim toga, moguće je iskoristiti neki od dostupnih alata za integraciju mapi, kojom bi vlasnik OPG-a postavio točnu lokaciju, a korisnik bi na taj način puno jednostavnije došao do točnog mjesta na kojem se OPG nalazi. Također, postoji potreba za automatskim postavljanjem stranice na neki od servera, kao i za automatsko kreiranje domene za OPG, čime bi se dodatno rasteretilo vlasnike obrta.

Ovo rješenje je zamišljeno kao besplatan servis za sve vlasnike OPG-ova koji nemaju potrebno znanje ni vrijeme za izradu web stranice svog obrta. Na tržištu postoje druga rješenja koja imaju svoje prednosti i mane, ali nisu namijenjene isključivo za vlasnike OPG-ova već za komercijalnu upotrebu. Korištenje ove aplikacije zahtijeva minimalno informatičko znanje i minimalan broj koraka koje je potrebno odraditi kao bi se došlo do web stranice obrta. Dobivena web stranica je modernog izgleda i popunjena je osnovnim informacijama koje bi svaka web stranica OPG-a trebala imati. Na web stranici korisnik može dobiti uvid u najbitnije informacije o nekom OPG-u, a za sve dodatne upite dostupna je i kontakt forma. Dobiveno rješenje zadovoljava potrebe većine vlasnika OPG-ova, a za kompleksnije web stranice vlasnicima se preporuča korištenje nekih drugih dostupnih rješenja koja su navedena u uvodnim poglavljima ovog rada.

## LITERATURA

- [1] CMS , [https://en.wikipedia.org/wiki/Content\\_management\\_system](https://en.wikipedia.org/wiki/Content_management_system), pristupio 1.8.2024.
- [2] WordPress , <https://wordpress.com/>, pristupio 4.8.2024.
- [3] Wix , <https://www.wix.com/>, pristupio 4.8.2024.
- [4] Shopify , <https://www.shopify.com/>, pristupio 6.8.2024.
- [5] Joomla! , <https://www.joomla.org/>, pristupio 6.8.2024.
- [6] Visual Studio Code , <https://code.visualstudio.com/>, pristupio 9.8.2024.
- [7] React , <https://react.dev/>, pristupio 10.8.2024.
- [8] Express , <https://expressjs.com/>, pristupio 12.8.2024.
- [9] Multer , <https://www.npmjs.com/package/multer>, pristupio 12.8.2024.
- [10] Ejs , <https://ejs.co/>, pristupio 12.8.2024.

## SAŽETAK

U diplomskom radu dan je opis problema i rješenje jednostavnog CMS-a za izrade web stranice obiteljskog poljoprivrednog gospodarstva. Navedene su prednosti i mane postojećih rješenja. Objasnjene su tehnologije i alati potrebni za izradu aplikacije i dan je detaljan opis programskog rješenja. Na kraju je opisan rad s aplikacijom, a u zaključku je naveden osvrt na rad te problematiku koja se njime rješava.

**Ključne riječi:** CMS, Express, JavaScript, React

## **ABSTRACT**

### **Improvement of CMS solution**

In the thesis, a description of the problem and a solution for a simple CMS for creating a website for a family farm is provided. The advantages and disadvantages of existing solutions are outlined. The technologies and tools required for application development are explained, and a detailed description of the software solution is given. Finally, the operation of the application is described, and the conclusion offers a reflection on the work and the issues it addresses.

**Keywords:** CMS, Express, JavaScript, React

## ŽIVOTOPIS

Ivan Rajkovača je rođen 07. 04. 1999. godine u Slavonskom Brodu. Pohađao je osnovnu školu “Ivan Goran Kovačić“ u Slavonskom Brodu. Fakultetsko obrazovanje započinje 2017. godine na Fakultetu Elektrotehnike, Računarstva i Informatičkih Tehnologija u Osijeku nakon završene opće gimnazije u Slavonskom Brodu. Nakon završenog trogodišnjeg preddiplomskog studija, 2020. godine upisuje diplomski studij računarstva.

---

Ivan Rajkovača