

# Web platforma za oglašavanje računalne opreme

---

**Gudelj, Kristijan**

**Undergraduate thesis / Završni 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:200:033088>

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

*Download date / Datum preuzimanja:* **2025-02-16**

*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 preddiplomski studij Računarstvo**

**Web platforma za oglašavanje računalne opreme**

**Završni rad**

**Kristijan Gudelj**

**Osijek, 2024.**

**FERIT**FAKULTET ELEKTROTEHNIKE, RAČUNARSTVA  
I INFORMACIJSKIH TEHNOLOGIJA OSIJEK**Obrazac Z1P: Obrazac za ocjenu završnog rada na sveučilišnom prijediplomskom studiju****Ocjena završnog rada na sveučilišnom prijediplomskom studiju**

<b>Ime i prezime pristupnika:</b>	Kristijan Gudelj
<b>Studij, smjer:</b>	Sveučilišni prijediplomski studij Računarstvo
<b>Mat. br. pristupnika, god.</b>	R4646, 28.07.2021.
<b>JMBAG:</b>	0165091142
<b>Mentor:</b>	prof. dr. sc. Krešimir Nenadić
<b>Sumentor:</b>	
<b>Sumentor iz tvrtke:</b>	
<b>Naslov završnog rada:</b>	Web platforma za oglašavanje računalne opreme
<b>Znanstvena grana završnog rada:</b>	<b>Programsko inženjerstvo (zn. polje računarstvo)</b>
<b>Zadatak završnog rada:</b>	Kratko opisati zathjeve prema funkcionalnostima koje bi trebao imati portal za oglašavanje računalne opreme. Dati kratki pregled postojećih rješenja i usporediti ih s izrađenim portalom. Za potrebe web portala potrebno je projektirati i izraditi odgovarajuću bazu podataka te dati opis izrađene baze. U web portal potrebno je ugraditi funkcionalnosti registracije i prijave korisnika. Prijavljeni korisnici trebaju imati mogućnosti postavljanja oglasa kojim oglašavaju prodaju računalne opreme, upravljanje postavljenim oglasima, zadavanje liste željenih proizvoda. mogućnost filtriranja proizvoda prema nekim
<b>Datum prijedloga ocjene završnog rada od strane mentora:</b>	06.09.2024.
<b>Prijedlog ocjene završnog rada od strane mentora:</b>	Izvrstan (5)
<b>Datum potvrde ocjene završnog rada od strane Odbora:</b>	25.09.2024.
<b>Ocjena završnog rada nakon obrane:</b>	Izvrstan (5)
<b>Datum potvrde mentora o predaji konačne verzije završnog rada čime je pristupnik završio sveučilišni prijediplomski 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:**

Kristijan Gudelj

**Studij:**

Sveučilišni prijediplomski studij Računarstvo

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

R4646, 28.07.2021.

**Turnitin podudaranje [%]:**

9

Ovom izjavom izjavljujem da je rad pod nazivom: **Web platforma za oglašavanje računalne opreme**

izrađen pod vodstvom mentora prof. dr. sc. Krešimir Nenadić

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

1. UVOD.....	1
1.1. Zadatak završnog rada.....	1
2. PRIKAZ STANJA U PODRUČJU I POSTOJEĆA SLIČNA RJEŠENJA.....	3
2.1. Pregled postojećih rješenja.....	3
2.1.1. Njuškalo.....	3
2.1.2. Plavi Oglasnik.....	4
2.1.3. Index oglasi.....	5
2.1.4. eBay.....	6
3. MODEL I GRAĐA WEB APLIKACIJE.....	7
3.1. Funkcionalni zahtjevi za web platformu.....	7
3.1.1. Prijava i registracija korisnika.....	7
3.1.2. Objava oglasa na platformi.....	8
3.1.3. Uređivanje i brisanje oglasa.....	8
3.1.4. Pretraživanje i filtriranje oglasa.....	8
3.1.5. Slanje poruka.....	8
3.1.6. Odjava korisnika.....	8
4. OPIS POSTUPKA IZRADE APLIKACIJE.....	9
4.1. Korišteni programski alati u izradi aplikacije.....	9
4.1.1. HTML.....	9
4.1.2. CSS.....	9
4.1.3. JavaScript.....	9
4.1.4. Django.....	10
4.1.5. SQLite.....	10

4.1.6. Visual Studio Code .....	10
4.2. Programsko rješenje u izradi aplikacije .....	11
4.2.1. Programsko rješenje registracije i prijave korisnika .....	11
4.2.2. Programsko rješenje predaje oglasa .....	13
4.2.3. Programsko rješenje slanja poruka .....	14
4.2.4. Programsko rješenje pretraživanja proizvoda .....	16
4.2.5. Programsko rješenje dodavanja i uređivanja proizvoda .....	17
5. PRIKAZ RADA APLIKACIJE .....	19
5.1. Način korištenja web aplikacije .....	19
5.1.1. Početna stranica .....	19
5.1.2. Registracija i prijava korisnika .....	20
5.1.3. Korisničke opcije prijavljenog korisnika .....	22
6. ZAKLJUČAK .....	28
LITERATURA .....	29
SAŽETAK .....	30
ABSTRACT .....	31

# 1. UVOD

Dobra i kvalitetna organizacija upravljanja web platformama za oglašavanje bilo koje vrste opreme složen je proces koji zahtijeva pažljivo planiranje i izvršavanje, te proces olakšava povezivanje kupaca i prodavača, potiče transparentnost i omogućuje brže donošenje odluka o kupovini. Prije interneta i web platformi, oglašavanje se obavljalo putem tiskanih oglasa, telefonskih poziva i osobnih susreta, što je oduzimalo puno vremena i novca. Danas, s razvojem digitalnih tehnologija, oglašavanje je znatno jednostavnije jer se sve potrebne informacije nalaze na jednom mjestu, čime se smanjuju potrebni resursi. Digitalna rješenja omogućuju brže prikupljanje podataka o ponudama, bolju personalizaciju korisničkog iskustva te proširene mogućnosti praćenja i analize tržišnih trendova. Web platforme omogućuju oglašivačima da postavite detaljne oglase prema specifičnim uvjetima, a kupcima da jednostavno pretražuju i filtriraju.

Cilj ovog završnog rada je riješiti izazove vezane uz modernu web platformu koja će sadržavati sve ključne elemente potrebne za uspješno funkcioniranje internetskog oglasnika. Naglas rada je na razvoju web platforme koja omogućuje učinkovito pretraživanje i filtriranje proizvoda, te implementaciju funkcionalnog sustava za razmjenu poruka među korisnicima. Kroz četiri poglavlja, rad detaljno opisuje sve faze razvoja aplikacije, uključujući korištene tehnologije i funkcionalnosti pojedinih elemenata, s naglaskom na praktična rješenja i tehničku izvedbu.

U drugom poglavlju dan je opis postojećih rješenja te usporedba sa platformom završnog rada. Treće poglavlje fokusirano je na opis modela i građe, dok se u četvrtom poglavlju opisuje postupak izrade aplikacije s pojašnjenjima programskog koda, te kratkim opisom korištenih tehnologija. Peto poglavlje vezano je za opis korištenja aplikacije i funkcionalnosti. Naposljetku slijedi zaključak s preporukama za daljnja istraživanja i mogućnosti proširivanja aplikacije dane ovim radom.

## 1.1. Zadatak završnog rada

Zadatak završnog rada kratko je opisati zahtjeve prema funkcionalnostima koje bi trebao imati portal za oglašavanje računalne opreme. Dati kratki pregled postojećih rješenja i usporediti ih s izrađenim portalom. Za potrebe web portala potrebno je projektirati i izraditi odgovarajuću bazu podataka te dati opis izrađene baze. U web portal potrebno je ugraditi funkcionalnosti registracije i prijave korisnika. Prijavljeni korisnici trebaju imati mogućnosti postavljanja oglasa

kojim oglašavaju prodaju računalne opreme, upravljanje postavljenim oglasima, zadavanje liste željenih proizvoda, mogućnost filtriranja proizvoda prema nekim kriterijima (kategorijama). Postupak izrade web portala je potrebno detaljno opisati.



## **2. PRIKAZ STANJA U PODRUČJU I POSTOJEĆA SLIČNA RJEŠENJA**

Ovo poglavlje pruža pregled i usporedbu sličnih aplikacija domaćih i stranih oglasnika, prema kojima je izrađena web platforma. Također, analizirana je važnost ovakvih platformi u suvremenom poslovnom svijetu, s posebnim naglaskom na njihove funkcionalnosti i primijenjene tehnologije.

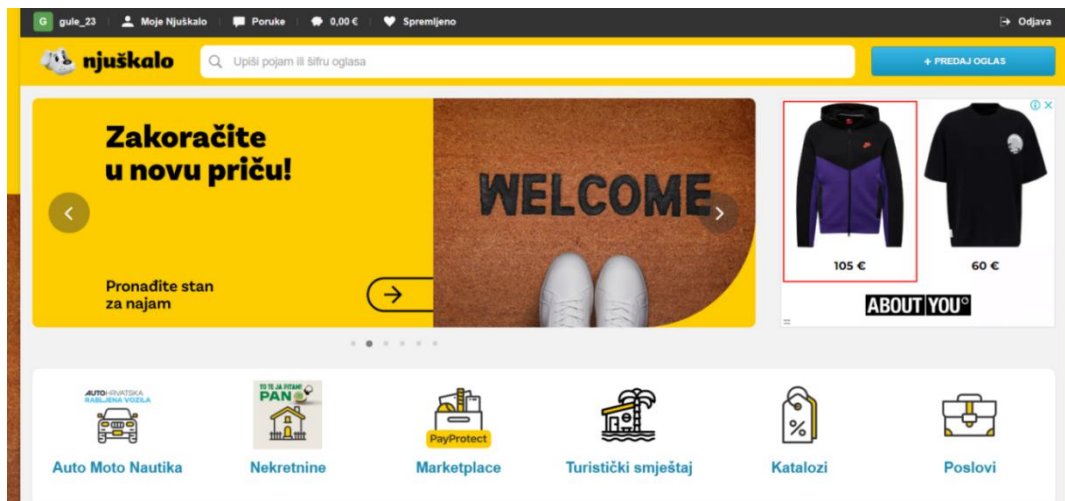
### **2.1. Pregled postojećih rješenja**

#### **2.1.1. Njuškalo**

Njuškalo je prema [1] najveći internetski oglasnik u Hrvatskoj, pokrenut 2007. godine. Broji preko 2,7 milijuna registriranih korisnika s preko 43 milijuna objavljenih oglasa. Na mjesečnoj bazi stranica je posjećena preko 1,5 milijuna puta.

Njuškalo i platforma koja je izrađena u okviru ovog rada imaju jako slične funkcionalnosti. Kao i u završnom radu, oglas se može objaviti, a zainteresirani se mogu javiti osobi koja je objavila oglas. Prilikom predaje oglasa korisnici moraju unijeti podatke poput: naziva, opisa, slike, adrese stanovanja, kontakta... Jedna od ključnih karakteristika platforme je širok raspon kategorija, uključujući nekretnine, vozila, tehniku, modu, dom i vrt, posao i razne druge kategorije. U drugu ruku, platforma u okviru završnog rada specijalizirana je za proizvode koje spadaju u kategoriju računalne opreme. Također, Njuškalo nudi korisnicima slanje poruka unutar platforme što olakšava dogovaranje uvjeta kupnje, odnosno prodaje. Slična funkcionalnost implementirana je u završnom radu.

*Slika 2.1.* prikazuje naslovnu stranicu Njuškala na kojoj su prikazane korisničke mogućnosti prijavljenog korisnika, kao što su mogućnost slanja poruka, predaje oglasa, pregled trenutno stanja računa i sl.



*Slika 2.1. Prikaz naslovne stranice Njuškala prijavljenog korisnika [1]*

### 2.1.2. Plavi Oglasnik

Plavi Oglasnik prema [2], poznat i kao Oglasnik.hr, jedna je od najstarijih i najpoznatijih platformi za oglašavanje u Hrvatskoj. Osnovan je 1989. godine, a započeo je kao agencija za tiskano oglašavanje koja je korisnicima omogućavala objavu oglasa za prodaju, kupnju i najam raznih proizvoda i usluga [2]. Razvojem interneta, platforma se prilagođavala novim trendovima i osnovala web platformu, što je omogućilo objavljivanje i pretraživanje oglasa online.

Plavi Oglasnik nudi razne mogućnosti koje olakšavaju oglašavanje proizvoda i usluga. Platforma pokriva širok raspon kategorija, uključujući nekretnine, vozila, tehnologiju, modu, dom i vrt, posao i razne druge kategorije kao što su sportska oprema, knjige i glazbeni instrumenti, dok se platforma u okviru završnog rada specijalizira za proizvode koji spadaju u računalnu opremu. Plavi Oglasnik nema dobro razvijen sustav za filtriranje s obzirom na širinu spektra proizvoda koje nudi, pa je snalaženje otežano. Platforma u okviru završnog rada nudi filtriranje i sortiranje po cijeni, vrsti proizvoda i slično. Ova značajka omogućuje korisnicima da brže pronađu ono što traže.

*Slika 2.2. prikazuje stranicu za detalje o odabranom korisniku. Po uzoru na nju napravljena je i stranica za detalje o proizvodu na platformi završnog rada.*

### Matrični pisač Epson DFX-9000 Epson EPS-1176



Slika 2.2. Prikaz detalja o proizvodu [2]

### 2.1.3. Index oglasi

Index Oglasi je platforma koja korisnicima nudi mogućnost jednostavne i brze promocije proizvoda i usluga. Platforma je vrlo brzo stekla popularnost zahvaljujući suradnji s jednim od najposjećenijih portala u Hrvatskoj, čime je velikom broju korisnika omogućen pristup oglasima na jednom mjestu. Prema [3], Index.hr je osnovan 2002. godine, a s vremenom je svoju ponudu proširio na široku ponudu usluga, uključujući i oglase.

Platforma nudi nekoliko značajki koje olakšavaju oglašavanje i traženje proizvoda i usluga. Index oglasi podijeljeni su u nekoliko glavnih kategorija. Dostupan je vrlo dobro optimiziran sustav za filtriranje proizvoda, odnosno svaka kategorija sadrži vrlo prilagođen način odabira filtera što olakšava snalaženje. Na slici 2.3. vidljiv je način filtracije za automobile gdje je moguće odabrati vrstu motora, model, tip karoserije i slično. Dodatno, Index oglasi sadrže kategorije za poslove, usluge i razne druge proizvode i usluge koji ne spadaju u navedene kategorije.



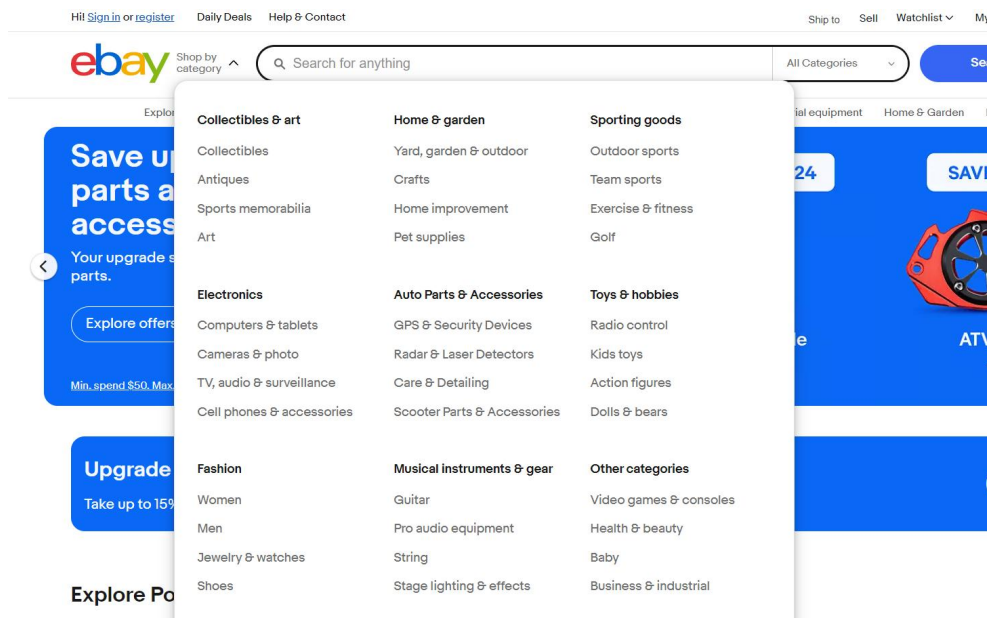
Slika 2.3. Prikaz mogućnosti filtriranja za kategoriju automobila [3]

#### 2.1.4. eBay

eBay jedna je od vodećih online platformi za kupnju i prodaju proizvoda koja je, prema [4], pokrenuta 1995. godine, kada ju je osnovao Pierre Omidyar u San Joseu. Prvotno poznat kao "AuctionWeb", eBay je brzo stekao popularnost kao platforma za internetske aukcije, dozvoljavajući korisnicima da licitiraju na proizvode koje su postavili prodavači. Prema [4], platforma broji oko 130 milijuna aktivnih korisnika sa preko 190 tržišta u svijetu.

Kako se razvijao, eBay je proširio svoje usluge kako bi uključivao mogućnosti za kupnju po fiksnim cijenama, čime je postao jedan od najvećih e-commerce tržišta na svijetu. Platforma eBay omogućuje korisnicima da postavljaju oglase na jednostavan način. Za to je potrebno registrirati se i prijaviti na eBay račun. Nakon prijave, korisnici mogu kreirati oglase za proizvode koje žele prodati. Postavljanje oglasa uključuje unos naslova, detaljnog opisa, cijene, te dodavanje fotografija proizvoda.

Sadržaj na eBay-u je raspoređen u različitim kategorijama proizvoda, kao što je vidljivo na slici 2.4., kako bi olakšao pretraživanje. Kategorije uključuju elektroniku, modu, automobile, nekretnine i mnoge druge. U svakoj kategoriji, oglasi su detaljno razrađeni s informacijama i slikama proizvoda, što pomaže kupcima da pronađu ono što im je potrebno.



Slika 2.4. Prikaz kategorija na naslovnoj stranici eBay-a [4]

### **3. MODEL I GRAĐA WEB APLIKACIJE**

U ovome poglavlju opisani su model i građa web platforme kroz funkcionalne zahtjeve. Planiranje projekta prvi je korak koji se mora poduzeti. Dio ovog planiranja uključuje identifikaciju karakteristika projekta i njihov utjecaj na proces razvoja [13].

#### **3.1. Funkcionalni zahtjevi za web platformu**

Web platforma na početnoj stranici unutar navigacijske trake prikazuje korisnicima poveznice za dva obrasca. Prvi obrazac omogućuje registraciju korisnicima koji nemaju već postojeći račun. Prilikom registracije unosi se korisničko ime, prezime, grad, e-mail adresa i ostali podaci, nakon čega se vrši autentifikacija i pohrana korisničkih informacija u bazu podataka. S druge strane, drugi obrazac je namijenjen korisnicima koji već imaju izrađen račun. Korisnik imaju mogućnost prijave unosom korisničkog imena i lozinke. Prijavljenom korisniku otvara se mogućnost objavljivanja oglasa na platformi. Prilikom prijave, na zaglavlju stranice, stvara se novi obrazac koja služi za objavu oglasa. Korisnik pomoću tog obrasca unosi podatke o proizvodu kao što su naziv proizvoda, kategorija, cijena, fotografije itd. Korisnik ima pristup podacima o svom računu gdje ima mogućnost izmjene privatnih podataka i pristup listi svih svojih objavljenih proizvoda. Kako bi se osigurala privatnost i sigurnost korisnika, aplikacija omogućuje jednostavnu odjavu s korisničkog računa, čime korisnici izlaze iz platforme i osiguravaju svoje podatke.

Platforma sadržava oglase raznih vrsta proizvoda iz kategorija računalne opreme kao što su računala, monitori, miševi, tipkovnice i slično. Korisnik ima mogućnost filtriranja i sortiranja proizvoda. Filtriranje i sortiranje ključne su funkcionalnosti, koje omogućavaju korisnicima efikasan pronalazak proizvoda koji ih zanimaju. Korisnici mogu odabrati određene kategorije proizvoda koje žele prikazati. Na primjer, ako korisnik želi vidjeti samo proizvode u kategoriji Računala, može označiti ovu opciju kako bi prikazao samo relevantne proizvode.

Također, bitna stavka platforme je mogućnost međusobne komunikacije između korisnika unutar same platforme. Registrirani i prijavljeni korisnici mogu razmjenjivati poruke s drugim korisnicima. Štoviše, imaju pristup dijelu platforme koji sadržava sve poruke sa ostalim korisnicima.

##### **3.1.1. Prijava i registracija korisnika**

Korisnici bez registracije i prijave mogu koristiti aplikaciju za pregled dostupnih oglasa, ali ne mogu komunicirati s drugim korisnicima niti objavljivati vlastite oglase. Pristup oba

obrasca se nalazi na naslovnoj stranici. Korisnik se mora registrirati da bi mogao koristiti obrazac za prijavu. Registracijom se podaci o korisničkom računu spremaju u bazu podataka, a obrazac za prijavu zahtjeva unos korisničkog imena te dohvaća podatke o korisniku iz baze podataka.

### **3.1.2. Objava oglasa na platformi**

Nakon prijave, korisniku se na navigacijskoj traci stvara pristup novom obrascu za objavu oglasa. Klikom na gumb za predaju oglasa, korisnik se preusmjerava na novu stranicu s obrascem pomoću kojeg unosi podatke o proizvodu kao što su cijena, kategorija, opis, slika i slično.

### **3.1.3. Uređivanje i brisanje oglasa**

Korisnik ima mogućnost uvida svih objavljenih oglasa na vlastitoj profilnoj stranici. Pritiskom na gumb za uređivanje oglasa otvara se obrazac pomoću kojeg je moguće urediti postojeće podatke o proizvodu. Također, omogućena je mogućnost brisanja oglasa ukoliko korisnik to odluči.

### **3.1.4. Pretraživanje i filtriranje oglasa**

Korisnik, koji ne mora biti prijavljen, može pretraživati i filtrirati proizvode što pomaže pri brzem i lakšem pronalaženju željenih proizvoda. Filtriranje je omogućeno po kategoriji i rasponu cijena. Također, dostupna je tražilica koja funkcionira na način da pretražuje proizvode po riječi ili dijelu riječi iz naziva proizvoda.

### **3.1.5. Slanje poruka**

Prijavljeni korisnici imaju mogućnost komunikacije unutar platforme. Ona se odvija pomoću sustava za slanje poruka koji olakšava dogovaranja oko detalja kupnje, odnosno prodaje proizvoda. Svaki korisnik ima poseban pretinac za slanje poruka unutar kojeg se nalaze spremljeni razgovori sa svim korisnicima.

### **3.1.6. Odjava korisnika**

Svi korisnici imaju mogućnost odjave sa svog korisničkog računa tako da jednostavno pritisnu gumb za odjavu na navigacijskoj traci. Klikom na ovu opciju, korisnik više nema mogućnost objave oglasa ni komuniciranja sa drugim korisnicima, ali su također njegovi podaci sigurni od neovlaštenog korištenja.

## 4. OPIS POSTUPKA IZRADE APLIKACIJE

U ovom poglavlju opisan je način izrade aplikacije zajedno sa tehnologijama koje su korištene u izradi. Prikazani i objašnjeni su dijelovi koda koji obavljaju specifične zadatke na platformi.

### 4.1. Korišteni programski alati u izradi aplikacije

#### 4.1.1. HTML

Prema [5], HTML (*HyperText Markup Language*) je standardni jezik koji se koristi kao baza u strukturiranju i kreiranju internetskih stranica. HTML je započeo ranih 1990-ih kao kratak dokument koji je opisivao nekoliko elemenata koji se koriste za izradu web stranica [11]. Mnogi od tih elemenata bili su namijenjeni za sadržaj poput naslova, paragrafa, popisa i poveznica na druge stranice. Kako se jezik razvijao s uvođenjem novih elemenata i prilagođavanjem pravila, broj verzija HTML-a se povećavao, a najpoznatija verzija je HTML5. Njegova glavna funkcija je strukturiranje sadržaja na stranici kroz upotrebu oznaka. HTML ima mnogo primjena, posebno u sljedećim područjima: web razvoj, navigacija internetom i web dokumentacija.

#### 4.1.2. CSS

CSS (*Cascading Style Sheets*) je, prema [6], standardni jezik koji je osnova za stiliziranje i dizajniranje web stranica. Prva verzija, CSS1 specifikacija razvijena je 1996. godine od strane W3C-a, dok je CSS2, koji je sadržavao mnoge nove funkcionalnosti, objavljen 1998. godine [12]. Glavna mu je funkcija kontrola izgleda i rasporeda sadržaja na web stranici kroz upotrebu različitih stilova kao što su boja, font, margine... CSS ima mnogo primjena u web razvoju, vizualnom dizajnu i kreiranju responzivnog izgleda stranica. U aplikaciji je korišten za uređivanje izgleda stranica pisanih u HTML kodu.

#### 4.1.3. JavaScript

JavaScript je skriptni jezik koji se izvodi na strani klijenta, a u web programiranju se prvenstveno koristi za validaciju podataka u obrascima, izradu HTML dokumenata s podrškom za Ajax te za stvaranje dinamičnih HTML dokumenata [17]. Omogućava automatsko osvježavanje sadržaja različitih stranica, npr. prilikom objave sadržaja na društvenim mrežama, do prikazivanja animacija i interaktivnih mapa [16]. JavaScript omogućuje funkcionalnosti koje značajno poboljšavaju korisničko iskustvo. Kao jezik skriptiranja na strani klijenta, JavaScript je jedna od ključnih tehnologija na webu.

#### 4.1.4. Django

Django je Python web *framework* visoke razine koji se koristi za izradu različitih vrsta web aplikacija. Sadrži vrlo visoku razinu podrške od razvojnih programera što uvelike olakšava njegovo korištenje. *Backend framework* se temelji na programskom jeziku Python. Prema [10], Python je svestran programski jezik poznat po svojoj jednostavnosti i lakoći čitanja, što ga čini prikladnim za početnike i iskusne programere. Koristi se u različitim domenama, poput web razvoja, analize podataka, automatizacije, umjetne inteligencije i znanosti o podacima.

Django nudi širok spektar alata i funkcionalnosti, uključujući snažan sustav za upravljanje bazama podataka, napredne sigurnosne mjere, te fleksibilnu arhitekturu za izradu skalabilnih web aplikacija [15]. Sadrži detaljnu dokumentaciju koja ga čini idealnim izborom za razvoj projekata svih veličina, od jednostavnih osobnih stranica do složenih korporativnih sustava.

#### 4.1.5. SQLite

SQLite je, prema [14], lagani sustav za upravljanje relacijskim bazama podataka koji ne zahtijeva zaseban poslužitelj, već se integrira direktno u aplikaciju koja ga koristi. Pohranjuje sve podatke u jednu *cross-platform* datoteku, što značajno pojednostavljuje administraciju i omogućuje lako premještanje ili sigurnosno kopiranje baze podataka. SQLite se automatski konfigurira kao zadana baza podataka u Django projektima, omogućujući brz početak rada [8]. Iako je vrlo brz i učinkovit za manje aplikacije, nije namijenjen za velike projekte s visokim opterećenjem ili kompleksnim upitima.

#### 4.1.6. Visual Studio Code

Visual Studio Code (VS Code), prema [9], je lagano, ali moćno razvojno okruženje koje podržava veliki broj programskih jezika. Razvijen od strane Microsofta, VS Code postao je popularan izbor među programerima zbog svoje fleksibilnosti, brzine i brojnih proširenja koja omogućuju prilagodbu radnog okruženja prema specifičnim potrebama svakog korisnika. Kao *open source* alat, Visual Studio Code je besplatan i dostupan za različite operative sustave poput Windowsa, macOS-a i Linuxa [9]. Njegova proširivost je jedna od ključnih prednosti, omogućujući korisnicima da instaliraju dodatke za podršku specifičnim jezicima, alatima i *frameworkovima*. Zbog ovih karakteristika, VS Code se često koristi u razvoju aplikacija svih vrsta, od jednostavnih web stranica do kompleksnih softverskih sustava. Uz to, Visual Studio Code nudi jednostavno sučelje s moćnim alatima za upravljanje projektima, što ga čini idealnim izborom za programere svih razina iskustva.



## 4.2. Programsko rješenje u izradi aplikacije

### 4.2.1. Programsko rješenje registracije i prijave korisnika

Podaci za korisnički profil definirani su unutar modela klase *Profile* čiji je programski kod prikazuje *slici 4.1*. Prema [15], Django ima ugrađenu snažnu podršku za upravljanje korisničkim računima, uključujući autentifikaciju, autorizaciju i integraciju s admin sučeljem. Ovo omogućava jednostavno proširenje osnovnog *User* modela dodatnim podacima specifičnim za aplikaciju, poput imena, prezimena, adrese i drugih informacija, a sve to se lako administrira putem Django admin sučelja.

Korisnik koji još nema kreiran račun ispunjava obrazac za registraciju, čiji je kod prikazan na *slici 4.2.*, te, pritiskom na gumb za registraciju, šalje HTTP zahtjev koji se obrađuje unutar funkcije za registraciju, *register*. Ova funkcija provjerava jesu li uneseni podaci ispravni, uključujući provjeru postoje li već korisničko ime ili e-mail adresa u bazi podataka. Ako se utvrdi da su uneseni podaci već iskorišteni, korisnik će biti obaviješten putem odgovarajuće *toast* poruke. Kada su svi podaci ispravno uneseni, oni se spremaju u bazu podataka, a korisnički račun postaje aktivan i spreman za prijavu i korištenje.

```
class Profile(models.Model):
    user = models.OneToOneField(User, null=True, on_delete=models.CASCADE)
    city = models.CharField(max_length=50)
    address = models.CharField(max_length=100)
    name = models.CharField(max_length=50, null=True)
    surname = models.CharField(max_length=50, null=True)
    profile_picture = models.ImageField(upload_to='profile_pictures/', blank=True, null=True)
    created_at = models.DateTimeField(auto_now_add=True)

    def __str__(self):
        return str(self.user)
```

Slika 4.1. Prikaz programskog koda za model korisničkog računa

```

def register(request):
    if request.method == 'POST':
        username = request.POST['username']
        name = request.POST['name']
        surname = request.POST['surname']
        email = request.POST['email']
        password = request.POST['password']
        password2 = request.POST['password2']

        if password == password2:
            if User.objects.filter(email=email).exists():
                messages.info(request, 'Email adresa je već iskorištena!')
                return redirect('register')
            elif User.objects.filter(username=username).exists():
                messages.info(request, 'Korisničko ime je već iskorišteno!')
                return redirect('register')
            else:
                user = User.objects.create_user(username=username, email=email, password=password)
                user.save()

        else:
            messages.info(request, 'Lozinka nije ista!')
            return redirect('register')

        form = ProfileForm(request.POST, request.FILES)
        if form.is_valid():
            profile = form.save(commit=False)
            profile.user = user
            profile.name = name
            profile.surname = surname
            profile.city = request.POST['city']
            profile.address = request.POST['address']
            profile.save()
            return redirect('login')
        else:
            messages.error(request, 'Molim Vas ispravite pogreške.')
            return redirect('register')
    else:
        form = ProfileForm()
        return render(request, 'registration/register.html', {'form': form})

```

**Slika 4.2.** Prikaz dijela koda za registraciju korisnika

Korisnik koji već ima kreiran račun, ispunjava obrazac za prijavu te, pritiskom na gumb za prijavu, šalje HTTP zahtjev koji se obrađuje unutar funkcije za prijavu. Funkcija *login*, prikazana na *slici 4.3.*, provjerava unesene podatke, uključujući korisničko ime i lozinku, kako bi se korisnik autentificirao. Ako su uneseni podaci točni i odgovaraju postojećem korisničkom računu, korisnik se prijavljuje i preusmjerava na početnu stranicu aplikacije. U slučaju netočnih podataka, korisnik će biti obaviješten putem odgovarajuće poruke o pogrešci. Kada su podaci točni, korisnički račun postaje aktivan za pristup zaštićenim dijelovima aplikacije.

```

def login(request):
    if request.method == 'POST':
        username = request.POST['username']
        password = request.POST['password']

        user = auth.authenticate(username=username,password=password)

        if user is not None:
            auth.login(request,user)
            return redirect(reverse('home'))
        else:
            messages.info(request,"Netočna lozinka!")
            return redirect('login')
    else:
        return render(request,'registration/login.html')

```

Slika 4.3. Prikaz koda za registraciju korisnika

#### 4.2.2. Programsko rješenje predaje oglasa

Predaja oglasa predstavlja ključnu funkcionalnost aplikacije, omogućujući korisnicima da unesu i objave informacije o proizvodima ili uslugama koje žele ponuditi. Programsko rješenje uključuje implementaciju obrasca za unos podataka o oglasu, validaciju unesenih informacija te pohranu oglasa u bazu podataka. Dodavanje proizvoda se vrši pomoću funkcije *add\_product*, prikazana na slici 4.4., koja prima HTTP zahtjev za predaju oglasa te zatim provjera ispravnost i sprema podatke u bazu podataka. Model, koji je prikazan na slici 4.5., sastoji se od više polja koja se popunjavaju pomoću posebno definiranog obrasca za unos, *ProductForm*. Ovaj proces je osmišljen tako da svi oglasi sadrže potrebne informacije i zadovoljavaju postavljene kriterije kvalitete.

```

@login_required
def add_product(request):
    if request.method == 'POST':
        form = ProductForm(request.POST, request.FILES)
        if form.is_valid():
            product = form.save(commit=False)
            product.seller = request.user
            product.save()
            return redirect('products_list')
    else:
        form = ProductForm()
    return render(request, 'advertisement/add_product.html', {'form': form})

```

Slika 4.4. Prikaz koda funkcije za dodavanje proizvoda

```

class Product(models.Model):
    CATEGORY_CHOICES = [
        ('Mobiteli', 'Mobiteli'),
        ('Zvučnici', 'Zvučnici'),
        ('Slušalice', 'Slušalice'),
        ('Monitori', 'Monitori'),
        ('Računala', 'Računala'),
        ('Tableti', 'Tableti'),
        ('Tipkovnice', 'Tipkovnice'),
        ('Kamere', 'Kamere'),
        ('Ostala oprema', 'Ostala oprema'),
    ]

    seller = models.ForeignKey(User, on_delete=models.CASCADE)
    title = models.CharField(max_length=128)
    description = models.TextField(max_length=1000, blank=True, null=True)
    price = models.DecimalField(max_digits=10, decimal_places=2)
    category = models.CharField(max_length=128, choices=CATEGORY_CHOICES, blank=True, null=True)
    quantity = models.IntegerField(default=1)
    date_posted = models.DateTimeField(default=timezone.now)
    image = models.ImageField(upload_to='images', blank=True, null=True, default='images/default_product_image.jpg')
    status = models.CharField(max_length=128, default="Dostupno")

    def __str__(self):
        return f"{self.id}: {self.title}"

```

Slika 4.5. Prikaz koda modela proizvoda

### 4.2.3. Programsko rješenje slanja poruka

Slanje poruka važna komponenta je platforme jer omogućuje bržu i bolju komunikaciju između korisnika. Implementirana je kao posebna aplikacija unutar glavnog projekta.

Kao što *slika 4.6.* prikazuje, model se sastoji od primatelja, pošiljatelja, vremena slanja poruke te samog teksta poruke. Korisnik putem obrasca za slanje poruke unosi tekst poruke, dok se podaci o pošiljatelju, primatelju i vremenu slanja automatski dodaju na temelju trenutnog prijavljenog korisnika, razgovora s drugim korisnikom i vremenu slanja. Ovaj model omogućuje precizno praćenje komunikacije između korisnika, čuvajući povijest razgovora i olakšavajući pretraživanje i pregled prethodnih poruka.

```

class Message(models.Model):
    m_sender = models.ForeignKey(User, on_delete=models.CASCADE, related_name="m_sender")
    m_receiver = models.ForeignKey(User, on_delete=models.CASCADE, related_name="m_receiver")
    sent_at = models.DateTimeField(default=timezone.now)
    text = models.CharField(max_length=5000)

    def __str__(self):
        return f"{self.m_sender} -> {self.m_receiver}"

    @classmethod
    def get_all_conversations(cls, user):
        messages = cls.objects.filter(models.Q(m_sender=user) | models.Q(m_receiver=user))
        users_in_conversations = User.objects.filter(
            id__in=messages.values_list('m_sender', flat=True).union(messages.values_list('m_receiver', flat=True))
        ).distinct()
        users_in_conversations = users_in_conversations.exclude(id=user.id)

        return users_in_conversations

    @classmethod
    def get_all_messages(cls, user, other_user):
        messages = cls.objects.filter(
            (models.Q(m_sender=user) & models.Q(m_receiver=other_user)) |
            (models.Q(m_sender=other_user) & models.Q(m_receiver=user))
        ).order_by('-sent_at')

        return messages

```

**Slika 4.6.** Prikaz koda modela poruke

*Slika 4.7.* prikazuje funkciju koja omogućuje dohvaćanje svih poruka i slanje poruka između korisnika. Funkcija zahtijeva da korisnik bude prijavljen i koristi njegov ID za pristup razgovorima. Nakon što korisnik uđe na stranicu, funkcija prvo dohvaća sve razgovore u kojima je prijavljeni korisnik sudjelovao, koristeći metodu *get\_all\_conversations*. Zatim se provjerava specifični korisnički ID u zahtjevu, ako on postoji dohvaća se korisnik iz baze podataka i prikazuje razgovor s tim korisnikom. U suprotnom, prikazuje se razgovor s prvim korisnikom s popisa. Nakon identifikacije drugog korisnika, funkcija dohvaća sve poruke između trenutnog korisnika i odabranog korisnika pomoću metode *get\_all\_messages*. Poruke se prikazuju kronološkim redosljedom kako bi korisnik mogao pratiti razgovor. Kada korisnik unese i pošalje novu poruku putem obrasca, funkcija provjerava valjanost obrasca i, ako je ispravan, pohranjuje poruku u bazu podataka.



```

@login_required
def messages_from_conversations(request, user_id=None):
    user = request.user
    my_conversations = Message.get_all_conversations(user)

    if not my_conversations:
        other_user = None
        all_messages = None
    else:
        if user_id:
            other_user = get_object_or_404(User, pk=user_id)
        else:
            other_user = my_conversations.first().user
        if other_user:
            all_messages = Message.get_all_messages(user, other_user)
        else:
            all_messages = None

    if request.method == 'POST':
        form = MessageForm(request.POST)
        if form.is_valid():
            message = form.save(commit=False)
            message.m_sender = request.user
            message.m_receiver = other_user
            message.save()
        else:
            form = MessageForm()

    context = {
        'my_conversations': my_conversations,
        'other_user': other_user,
        'all_messages': all_messages,
        'form': form,
        'no_conversations': not my_conversations
    }

    return render(request, 'messenger/messenger.html', context)

```

Slika 4.7. Prikaz funkcije za dohvaćanje i slanje poruka

#### 4.2.4. Programsko rješenje pretraživanja proizvoda

Pregledno i efikasno pretraživanje proizvoda jedna je od ključnih stavki svakog oglasnika. Unutar aplikacije ono je implementirano pomoću tražilice i filtriranja proizvoda.

Kod u HTML-u, prikazan na slici 4.8., implementira funkcionalnost filtriranja. Na lijevoj strani nalazi se mogućnost filtriranja po dva kriterija: cijena i kategorija. Kategorije se odabiru pomoću kvačica, klikom na *checkboxes*, dok se kod određivanja cijene odabire minimalna i maksimalna cijena proizvoda. Moguće je kombinirati ova dva kriterija. Obrazac koristi metodu *get* za slanje zahtjeva, čime se omogućuje da rezultati filtriranja budu prikazani na istoj stranici bez potrebe za dodatnim učitavanjima.

Slika 4.9. prikazuje funkciju *filtered\_products* koja obrađuje ove filtere. Funkcija dohvaća odabrane kategorije i cjenovni raspon iz GET zahtjeva i filtrira proizvode na temelju tih kriterija. Ona koristi Django ORM za pretraživanje baze podataka i vraća samo proizvode koji su dostupni unutar zadane cjenovne granice. Rezultati filtriranja se zatim prosljeđuju predlošku

kako bi se prikazali korisniku, omogućujući mu da vidi proizvode koji odgovaraju njegovim kriterijima pretraživanja.

```
<main class="main-container">
  <aside class="left-aside">
    <h3>Filtriraj svoju pretragu:</h3>
    <form method="get" action=".">
      <div class="category-search">
        <p><strong>Kategorije:</strong></p>
        {% for category in categories %}
          <div class="form-check">
            <input class="form-check-input filter-checkbox" type="checkbox" name="category" value="{{ category }}" id="category-{{ category }}"
              {% if category in selected_categories %}checked{% endif %}>
            <label class="form-check-label" for="category-{{ category }}">
              {{ category }}
            </label>
          </div>
        {% endfor %}
      </div>

      <div class="by-price">
        <p><strong>Cijena:</strong></p>
        <div class="price-range">
          <label for="min-price">Minimalna cijena:</label>
          <input type="number" id="min-price" name="min_price" min="0" value="{{ min_price }}" placeholder="0">

          <label for="max-price">Maksimalna cijena:</label>
          <input type="number" id="max-price" name="max_price" min="0" value="{{ max_price }}" placeholder="10000">
        </div>
        <button type="submit">Filtriraj</button>
      </div>
    </form>
```

Slika 4.8. Prikaz HTML koda za filtriranje proizvoda

```
def filtered_products(request):
    selected_categories = request.GET.getlist('category', Product.CATEGORY_CHOICES)

    min_price = request.GET.get('min_price', 0)
    max_price = request.GET.get('max_price', 10000)

    products = Product.objects.filter(
        status="Available",
        category__in=selected_categories,
        price__gte=min_price,
        price__lte=max_price
    )

    categories = [choice[0] for choice in Product.CATEGORY_CHOICES]

    context = {
        'products': products,
        'categories': categories,
        'selected_categories': selected_categories,
        'min_price': min_price,
        'max_price': max_price,
    }

    return render(request, 'advertisement/products_list.html', context)
```

Slika 4.9. Prikaz koda funkcije za filtriranje proizvoda

#### 4.2.5. Programsko rješenje dodavanja i uređivanja proizvoda

Dodavanje proizvoda stavka je kojoj pristup imaju samo prijavljeni korisnici. Korisnik ispunjava obrazac unošenjem svih detalja o proizvodu. Kada korisnik pošalje obrazac putem POST zahtjeva, funkcija `add_product` obrađuje te podatke. Ako je obrazac ispravan, podaci o proizvodu spremaju se u bazu podataka. U ovom procesu, `ProductForm` prikuplja sve unesene

informacije i, uz pomoć `request.FILES`, omogućuje dodavanje slika proizvoda. Nakon što korisnik uspješno unese podatke i obrazac se pošalje, funkcija dodjeljuje korisnika kao prodavača proizvoda. Tako se povezuje proizvod s korisnikom koji ga je dodao. Nakon što se proizvod pohrani, korisnik je preusmjeren na stranicu s popisom proizvoda. Funkcija za dodavanje proizvoda je prikazana na *slici 4.10*.

```
@login_required
def add_product(request):
    if request.method == 'POST':
        form = ProductForm(request.POST, request.FILES)
        if form.is_valid():
            product = form.save(commit=False)
            product.seller = request.user
            product.save()
            return redirect('products_list')
    else:
        form = ProductForm()
    return render(request, 'advertisement/add_product.html', {'form': form})
```

**Slika 4.10.** Prikaz koda funkcije za dodavanje proizvoda

Funkcija za uređivanje proizvoda, prikazana na *slici 4.11.*, omogućuje prijavljenim korisnicima da izmjene podatke o proizvodima koje su prethodno dodali. Funkcija prvo dohvaća specifični proizvod koristeći njegov ID i provjerava je li korisnik koji pokušava izvršiti izmjene stvarni prodavač tog proizvoda. Ako korisnik nije autor proizvoda, bit će prikazana stranica s pogreškom. Kada korisnik pošalje obrazac putem POST zahtjeva, funkcija `edit_product` obrađuje unesene podatke koristeći `ProductForm`. Ako su podaci u obrascu ispravni, oni se ažuriraju u bazi podataka, a korisnik je preusmjeren na stranicu s detaljima o uređenom proizvodu.

```
@login_required
def edit_product(request, product_id):
    product = get_object_or_404(Product, id=product_id, seller=request.user)

    if request.method == "POST":
        form = ProductForm(request.POST, request.FILES, instance=product)
        if form.is_valid():
            form.save()
            return redirect('product_detail', product_id=product.id)
    else:
        form = ProductForm(instance=product)

    return render(request, 'advertisement/edit_product.html', {'form': form, 'product': product})
```

**Slika 4.11.** Prikaz koda funkcije za uređivanje proizvoda



## 5. PRIKAZ RADA APLIKACIJE

U ovom poglavlju detaljno je opisan rad aplikacije kroz ključne komponente koje ona nudi. Prikazane su glavne funkcionalnosti, uključujući registraciju, prijavu, objavu oglasa i komunikaciju s drugim korisnicima na platformi. Izgled i funkcionalnosti aplikacije temelje se na sličnim aplikacijama koje rješavaju slične probleme, a koje su već razmotrene u prethodnom poglavlju: *Prikaz stanja u području i postojeća slična rješenja*. Aplikacija je testirana kako bi se provjerilo radi li ispravno. Primjeri na kojima je testirana uključuju uspješnu registraciju i prijavu korisnika, kreiranje i objavu oglasa, te slanje poruka između korisnika. Sve navedene funkcionalnosti pokazale su se ispravnim tijekom testiranja.

### 5.1. Način korištenja web aplikacije

Aplikacija uključuje oglase različitih korisnika iz područja računalne opreme. Iako neprijavljeni korisnici mogu pregledavati oglase, registracija i prijava otvaraju im pristup dodatnim funkcionalnostima. Svaki korisnik mora se autentificirati unosom korisničkog imena i lozinke putem obrasca za prijavu. Korisnici koji još nemaju račun mogu jednostavno kreirati svoj profil pomoću registracijskog obrasca. Nakon uspješne prijave, korisnicima postaju dostupne sve mogućnosti koje aplikacija nudi. Navigacijska traka pruža brz pristup ključnim dijelovima aplikacije, uključujući pregled i upravljanje vlastitim oglasima, postavljanje novih oglasa, te uređivanje profila. Korisnici mogu koristiti opciju za filtriranje i pretraživanje oglasa, kako bi lakše pronašli proizvode koji ih zanimaju. Stranica s detaljima o proizvodu nudi korisnicima sve potrebne informacije o odabranom proizvodu, poput opisa, cijene, slike i mogućnosti kontaktiranja prodavača. Aplikacija također omogućava korisnicima slanje privatnih poruka, čime se olakšava komunikacija i pregovaranje oko uvjeta kupovine. Korisnici na kraju korištenja mogu vrlo jednostavno završiti svoju sesiju klikom na tipku za odjavu, te se na taj način osigurava zaštita njihovih podataka.

#### 5.1.1. Početna stranica

Početna stranica aplikacije dizajnirana je s ciljem da korisnicima odmah pruži ključne informacije i jednostavan pristup svim važnim funkcionalnostima. Na vrhu stranice nalazi se navigacijska traka koja uključuje *linkove* za prijavu i registraciju, omogućavajući novim korisnicima jednostavan početak autentifikacije, dok postojeći korisnici mogu brzo pristupiti svojim računima. Navigacijska traka također sadrži *hover* meni, koji pruža brze *linkove* na stranice s traženim proizvodima, olakšavajući korisnicima pretragu i pregledavanje ponude.

Središnji dio stranice ističe promotivne poruke koje motiviraju korisnike na objavu oglasa, ističući prednosti i jednostavnost korištenja platforme. Ove poruke su osmišljene tako da privuku pažnju i potaknu korisnike na interakciju, bilo da se radi o postavljanju novog oglasa ili pretraživanju već objavljenih proizvoda. Na dnu stranice nalazi se *footer*, koji sadrži osnovne informacije o platformi, uključujući kontakt informacije, pravne napomene i linkove na stranice s uvjetima korištenja i politikom privatnosti. *Footer* je dizajniran tako da korisnicima pruža sve važne informacije i resurse na dohvat ruke, čime se dodatno poboljšava korisničko iskustvo.

### **5.1.2. Registracija i prijava korisnika**

Poput svake aplikacije namijenjene za interakciju s korisnicima, aplikacija sadrži obrasce za prijavu i registraciju. Ukoliko korisnik još nema račun, a želi koristiti pogodnosti koje su dostupne samo registriranim korisnicima, može se lako registrirati putem jednostavnog registracijskog obrasca. Nakon uspješne registracije, korisnik dobiva pristup svim funkcionalnostima aplikacije, uključujući mogućnost objave oglasa i komunikaciju s drugim korisnicima. Prijava na aplikaciju omogućava brz i siguran pristup korisničkom računu, čime se omogućuje personalizirano iskustvo korištenja.

*Slika 5.1.* prikazuje obrazac koji se otvara prilikom registracije korisnika. Za uspješnu registraciju, od korisnika se zahtijeva unos korisničkog imena, e-mail adrese, lozinke, imena, prezimena, grada te adrese stanovanja. Nakon registracije stvara se korisnički račun koji se sprema u bazu podataka te se korisnik preusmjerava na stranicu za prijavu.

The image shows a registration form on a dark background with green accents. At the top, the text 'REGISTRIRAJ SE' is displayed in green, flanked by two small green plant icons. Below this, the form consists of several input fields and checkboxes. The 'Korisničko ime:' field contains the text 'kiki'. The 'Email:' field is empty. The 'Lozinka:' field contains a series of dots. The 'Ponovite lozinku:' field is empty. The 'Ime:' field is empty. The 'Prezime:' field is empty. The 'Grad:' field is a dropdown menu with 'Zagreb' selected. The 'Adresa stanovanja:' field is empty. At the bottom, there are two checkboxes: the first is checked and labeled 'Pročitao sam [Politiku privatnosti](#) \*', and the second is checked and labeled 'Potvrđujem da imam više od 18 godina te prihvaćam [Uviete i pravila korištenja](#) \*'. A large green button with the text 'REGISTRIRAJ SE' is positioned at the bottom of the form.

**REGISTRIRAJ SE**

Korisničko ime:  
kiki

Email:

Lozinka:  
.....

Ponovite lozinku:

Ime:

Prezime:

Grad:  
Zagreb

Adresa stanovanja:

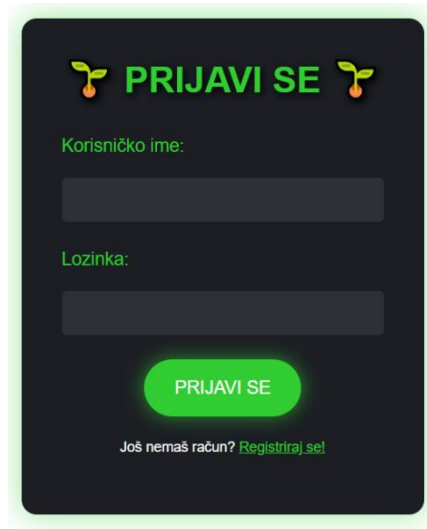
Pročitao sam [Politiku privatnosti](#) \*

Potvrđujem da imam više od 18 godina te prihvaćam [Uviete i pravila korištenja](#) \*

**REGISTRIRAJ SE**

Slika 5.1. Obrazac za registraciju korisnika

*Slika 5.2.* prikazuje obrazac za prijavu korisnika. Korisniku je nakon registracije potrebno unijeti samo prethodno kreirano korisničko ime i lozinku, te ako je prijava uspješna server ga ponovno preusmjerava na početnu stranicu. Na početnoj stranici korisniku se u navigacijskoj traci otvaraju nove opcije poput poveznice na stranicu svog profila i obrasca za predaju oglasa.

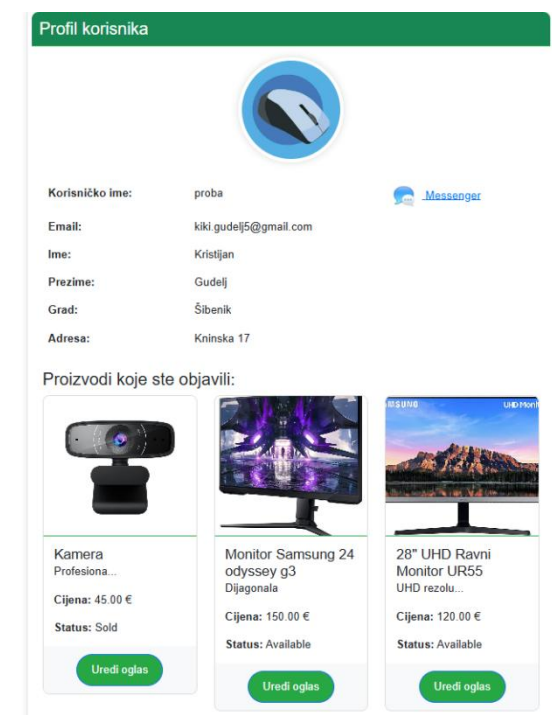
The image shows a dark-themed login form. At the top, it says 'PRIJAVI SE' in green capital letters, flanked by two small green plant icons. Below this, there are two input fields: the first is labeled 'Korisničko ime:' and the second is labeled 'Lozinka:'. Both fields are currently empty. Below the input fields is a prominent green button with the text 'PRIJAVI SE' in white. At the bottom of the form, there is a link that says 'Još nemaš račun? [Registrij se!](#)'.

**Slika 5.2.** *Obrazac za prijavu korisnika*

### **5.1.3. Korisničke opcije prijavljenog korisnika**

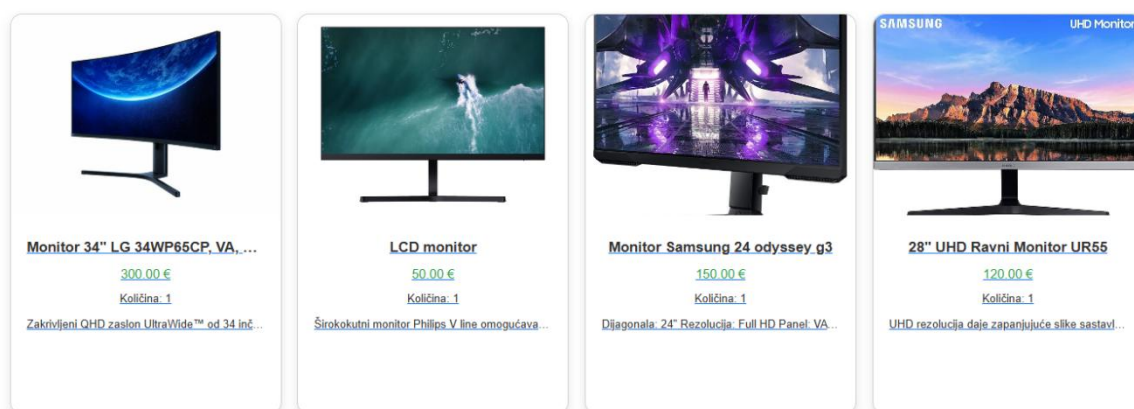
Ukoliko se korisnik uspješno prijavio, može predati svoj prvi oglas klikom na gumb za predaju oglasa koji mu se otvara u desnom kutu navigacijske trake. Klikom na gumb otvara se obrazac za predaju oglasa. Od korisnika se traži da unese naziv proizvoda, opis, cijenu u eurima, sliku, količinu, status dostupnosti i kategoriju. Ako je obrazac ispravno ispunjen, proizvod se sprema u bazu podataka i prikazuje na platformi, a korisnik preusmjerava na stranicu sa svojim profilom gdje su mu izlistani svi oglasi koje je objavio.

*Slika 5.3.* prikazuje primjer profilne stranice prijavljenog korisnika. Stranica sadrži osnovne podatke o korisniku unesene tokom registracije. Profilna stranica intuitivno je dizajnirana kako bi korisnicima omogućila lako upravljanje svojim podacima i oglasima. Također, ispod podataka o korisniku nalaze se svi proizvodi objavljeni od strane prijavljenog korisnika. Proizvodi su pregledno prikazani pa omogućuju brz pristup i uređivanje, čime korisnik zadržava potpunu kontrolu nad svojim objavama na platformi. Korisnik ima mogućnost izmjene podataka ili proizvoda klikom na gumb uređivanje oglasa, koji ga vodi na stranicu sa obrascem za izmjenu podataka. Isto tako, korisnik ima mogućnost odabira i izmjene profilne fotografije.



**Slika 5.3.** Profilna stranica prijavljenog korisnika

Slika 5.4. prikazuje kartice proizvoda na stranici, koje nastaju nakon objave oglasa. Kartice su responzivne, tj. klikom na karticu korisnik dobija detaljniji pregled proizvoda, što olakšava donošenje odluke o kupovini ili kontaktiranju prodavača.



**Slika 5.4.** Prikaz dijela izlistanih proizvoda na stranici

*Slika 5.5.* prikazuje opciju filtriranja koja korisnicima omogućuje lakše snalaženje i pronalaženje željenih proizvoda. Ova funkcionalnost omogućava korisnicima da precizno odaberu kriterije poput raspona cijene i vrste proizvoda, čime se značajno ubrzava proces pretraživanja i pronalaženja proizvoda koji odgovaraju njihovim potrebama.

**Filtriraj svoju pretragu:**

**Kategorije:**

- Mobiteli
- Zvučnici
- Slušalice
- Monitori
- Računala
- Tableti
- Tipkovnice
- Kamere
- Ostala oprema

**Cijena:**

Minimalna cijena:

Maksimalna cijena:

**Filtriraj**

**Slika 5.5.** Prikaz opcija za filtriranje proizvoda

*Slika 5.6.* prikazuje stranicu detalja o proizvodu koja se otvara kada korisnik stisne na responzivnu karticu prikazanu na *slici 5.4.* Prikazuje se cjelokupan pregled proizvoda, koji uključuje ključne informacije kao što su: naziv, opis, cijena, i slike proizvoda. Korisnik može lako pregledati sve ove informacije i, ako je zainteresiran, kontaktirati prodavača direktno putem ponuđenih opcija. Dizajn stranice je takav da omogućava intuitivno i ugodno iskustvo pregledavanja.

## Monitor Samsung 24 odyssey g3



proba  
Šibenik

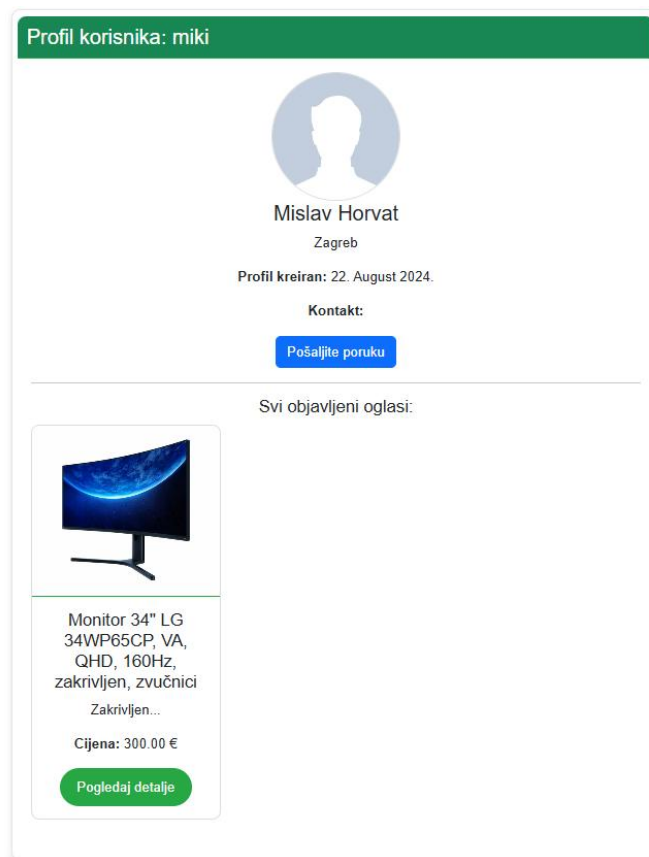
**Cijena:**  
150.00 € /  
1125.00 kn

Mogućnost besplatne dostave na  
Vašu adresu!

**Opis proizvoda:** Dijagonala: 24" Rezolucija: Full HD Panel: VA Osvježavanje slike: 144Hz

**Slika 5.6.** Prikaz detalja o proizvodu

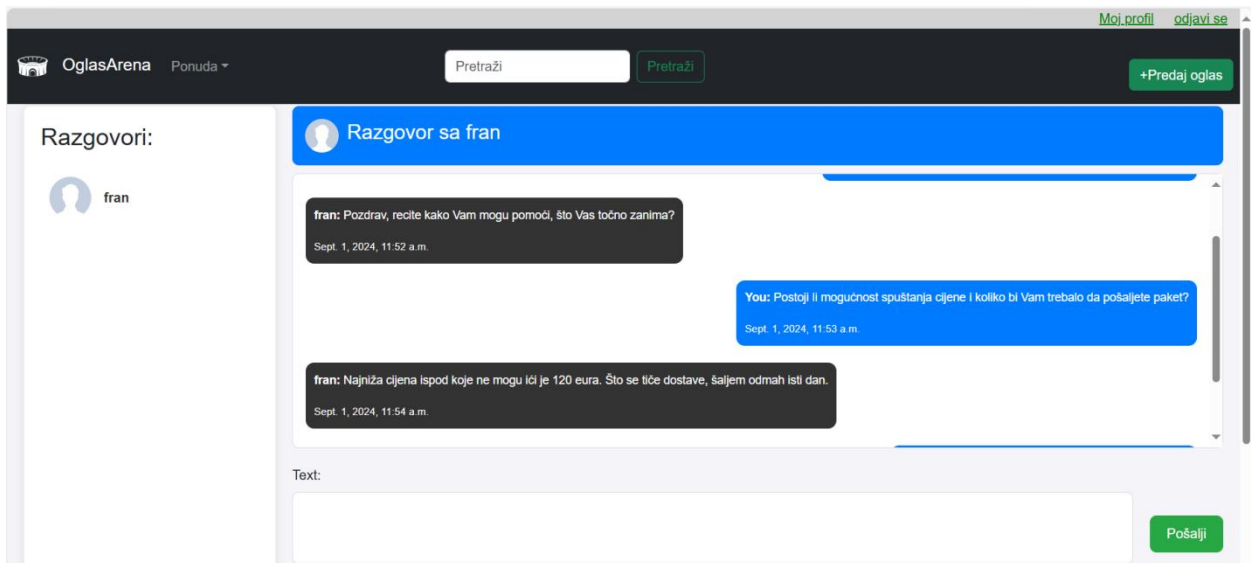
Slika 5.7. prikazuje profil korisnika koji se otvara klikom na ikonicu s profilnom slikom i korisničkim imenom na stranici s detaljima o proizvodu. Stranica je dizajnirana na jednostavan i pregledan način, prikazujući osnovne informacije o prodavaču, kao što su ime i kontakt podaci. Osim toga, na stranici su vidljivi svi oglasi koje je korisnik objavio, omogućujući pregled cijele ponude na jednom mjestu. Ova funkcionalnost olakšava potencijalnim kupcima procjenu vjerodostojnosti i kvalitete prodavača. Također, na stranici je dostupan gumb za slanje poruka, čijim klikom, korisnika direktno preusmjerava na stranicu za slanje poruka.



**Slika 5.7.** *Prikaz detalja o prodavaču*

*Slika 5.8.* prikazuje dio aplikacije namijenjen za razmjenu poruka između korisnika. Ova funkcionalnost omogućava korisnicima da uspostave direktnu komunikaciju s prodavačima ili kupcima unutar same aplikacije, pa se samim time olakšava dogovaranje oko detalja kupovine, postavljanje dodatnih pitanja o proizvodima ili pregovaranje o uvjetima prodaje. Dizajn stranice za razmjenu poruka je vrlo jednostavan, omogućavajući lako slanje i primanje poruka. Stranica je dizajnirana prema općeprihvaćenim standardima i praksama za slanje poruka, pružajući korisnicima poznato iskustvo komunikacije. Korisnici mogu pregledavati cijelu povijest razgovora, što doprinosi transparentnosti i lakšem praćenju dogovora. Ovaj sustav za razmjenu poruka dodatno poboljšava korisničko iskustvo, omogućavajući brzu i efikasnu komunikaciju unutar same aplikacije, bez potrebe za korištenjem vanjskih platformi.





**Slika 5.8.** Prikaz dijela aplikacije za razmjenu poruka

## 6. ZAKLJUČAK

Svrha završnog rada bila je izraditi web platformu koja korisnicima omogućuje jednostavno pretraživanje, filtriranje i oglašavanje računalne opreme. Platforma omogućava korisnicima pregled dostupnih proizvoda, jednostavno filtriranje ponude prema kategoriji i cijeni te razmjenu poruka između kupca i prodavača. Glavna prednost aplikacije je to što ujedinjuje sve informacije o računalnoj opremi na jednom mjestu, čime se olakšava proces pretrage i kupovine. Korisničko sučelje aplikacije jednostavno je za korištenje, a izrađeno je uporabom tehnologija kao što su HTML, CSS i JavaScript, dok je *backend* razvijen uz pomoć Django *frameworka*. Za pohranu podataka o proizvodima i korisnicima korištena je baza podataka SQLite. Aplikacija se može dalje razvijati dodavanjem funkcionalnosti kao što su recenzije i ocjene proizvoda, što bi omogućilo korisnicima da donesu bolje odluke prilikom kupovine. Dodatno, moguća je implementacija naprednog sustava za praćenje narudžbi, čime bi se korisnicima pružilo još bolje i potpunije iskustvo pri korištenju platforme. U zaključku, programsko rješenje postiglo je zadane ciljeve jer omogućuje jednostavnu i učinkovitu pretragu te oglašavanje računalne opreme. Provedeno testiranje aplikacije pokazalo je da su svi implementirani moduli funkcionalni, a buduće nadogradnje mogu dodatno poboljšati korisničko iskustvo.

## LITERATURA

- [1] Njuškalo, <https://www.njuskalo.hr/>, [datum posjete 1.9.2024.]
- [2] Plavi oglasnik, <https://www.oglasnik.hr/>, [datum posjete 1.9.2024.]
- [3] Index Oglasi, <https://www.index.hr/oglasni/>, [datum posjete 1.9.2024.]
- [4] Ebay, <https://www.ebay.com/>, [datum posjete 1.9.2024.]
- [5] HTML, <https://www.w3schools.com/html/>, [datum posjete 28.8.2024.]
- [6] CSS, <https://www.w3schools.com/css/>, [datum posjete 28.8.2024.]
- [7] Javascript, <https://www.w3schools.com/js/>, [datum posjete 28.8.2024.]
- [8] Django, <https://www.djangoproject.com/>, [datum posjete 28.8.2024.]
- [9] Visual Studio Code, <https://code.visualstudio.com/>, [datum posjete 28.8.2024.]
- [10] Barry, Paul. Head First Python. 2nd Edition, O'Reilly Media, 2016.
- [11] Elizabeth Castro, Bruce Hyslop, HTML and CSS, Eighth Edition, Peachpit Press, 2014.
- [12] Sebesta, Robert W. Programming the World Wide Web. 7<sup>th</sup> ed., Pearson Education Limited, 2014.
- [13] Van Villet, Hans, Software Engineering: Principles and Practice, Wiley, 2007.
- [14] Kreibich, Jay A., Using SQLite. First Edition, O' Reilly Media, 2010.
- [15] Vincent, William S., Django for Beginners: Build websites with Python and Django, Independently Published, 2018.
- [16] Abidakun, Taye. JavaScript Simplified. Independently Published, 2023.
- [17] Flanagan, David. JavaScript: The Definitive Guide. 6th Edition, O'Reilly Media, 2011.

## SAŽETAK

U ovom radu izrađena je i opisana web platforma za oglašavanje računalne opreme. Rad u početku pruža uvid u slična rješenja, a zatim su objašnjeni model i građa platforme. U idućem poglavlju objašnjen je kod ključnih funkcionalnosti platforme i navede korištene tehnologije, a to su: HTML, CSS, Javascripts, SQLite i Django. Za kraj je dan opis klijentskog dijela aplikacije. Platforma omogućuje korisnicima objavu oglasa na vrlo jednostavan i intuitivan način. Korisniku je potrebna registracija, odnosno prijava da bi dobio mogućnost objave oglasa. Isto tako, prijavljeni korisnik ima mogućnost komunikacije putem platforme s drugim korisnicima što omogućuje lakše trgovanje proizvodima. Platforma implementira mogućnost filtriranja i pretraživanja radi lakšeg pronalaska željenih proizvoda.

**Ključne riječi:** autentifikacija korisnika, Django, web platforma, oglasnik, računalna oprema

## **ABSTRACT**

**Title:** Web platform for advertising computer equipment

In this paper, a web platform for advertising computer equipment is developed and described. The paper initially provides an overview of similar solutions, followed by an explanation of the model and structure of the platform. In the next chapter, the code for the key functionalities of the platform is explained, and the technologies used are listed, including HTML, CSS, JavaScript, SQLite, and Django. Finally, the client-side of the application is described. The platform allows users to post advertisements in a very simple and intuitive way. Registration or login is required for users to be able to post ads. Similarly, a logged-in user has the ability to communicate with other users through the platform, which facilitates easier trading of products. The platform implements filtering and searching capabilities to make it easier to find desired products.

**Keywords:** computer equipment, Django, marketplace, web platform, user authentication