

WEB TRGOVINA ZA KUĆNE LJUBIMCE

Bočina, Dea

Undergraduate thesis / Završni rad

2024

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **University of Split / Sveučilište u Splitu**

Permanent link / Trajna poveznica: <https://um.nsk.hr/um:nbn:hr:228:664603>

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

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



Repository / Repozitorij:

[Repository of University Department of Professional Studies](#)



UNIVERSITY OF SPLIT



SVEUČILIŠTE U SPLITU
SVEUČILIŠNI ODJEL ZA STRUČNE STUDIJE

Prijediplomski stručni studij Računarstvo

DEA BOČINA

ZAVRŠNI RAD

WEB TRGOVINA ZA KUĆNE LJUBIMCE

Split, rujan 2024.

SVEUČILIŠTE U SPLITU
SVEUČILIŠNI ODJEL ZA STRUČNE STUDIJE

Prijediplomski stručni studij Računarstvo

Predmet: Sigurnost računala i podataka

ZAVRŠNI RAD

Kandidat: Dea Bočina

Naslov rada: Web trgovina za kućne ljubimce

Mentor: Lada Sartori, viši predavač

Split, rujan 2024.

Sadržaj

Sažetak	2
Summary	2
1. Uvod.....	3
2. Korištene tehnologije	4
2.1 Java Spring Boot.....	4
2.2 PostgreSQL	5
2.3 HTML, CSS	5
2.4 JavaScript	6
3. Izrada baze podataka.....	7
4. Izrada aplikacije.....	10
4.1 Izrada korisničkog sučelja.....	10
4.1.1 Izrada korisničkog računa i prijava.....	10
4.1.2 Početna stranica	13
4.1.3 Pregled proizvoda	14
4.1.4 Lista želja	17
4.1.5 Košarica	18
4.1.6 Plaćanje narudžbe i pregled narudžbe.....	20
4.1.7 Korisnički profil	24
4.1.8 Profili za ljubimce	25
4.2 Izrada zaposleničkog sučelja	26
4.2.1 Početna stranica	26
4.2.2 Upravljanje proizvodima.....	26
4.2.3 Upravljanje narudžbama	28
4.2.4 Upravljanje popustima	30
4.3 Izrada administratorskog sučelja	32
4.3.1 Početna stranica	32
4.3.2 Upravljanje korisnicima	32
4.3.3 Upravljanje sistemskom konfiguracijom	33
5. Zaključak.....	35
Literatura	36

Sažetak

Ovaj rad detaljno opisuje izradu web shop aplikacije za kućne ljubimce, koja korisnicima omogućava jednostavno pronalaženje i kupovinu proizvoda za pse, mačke i male životinje. Aplikacija podržava tri različite uloge: korisnik, zaposlenik i administrator, pri čemu svaka uloga donosi specifične mogućnosti i privilegije koje su temeljito razrađene kroz rad.

Aplikacija je implementirana koristeći Spring Boot okruženje (engl. *framework*) za poslužiteljski (engl. *backend*) dio, dok je baza podataka realizirana u PostgreSQL-u. Klijentski (engl. *frontend*) dio aplikacije je izrađen koristeći HTML, CSS i JavaScript, što osigurava pregledno i responzivno korisničko sučelje. Ovaj rad opisuje postupak izrade aplikacije, s posebnim naglaskom na tehničke aspekte i funkcionalnosti koje aplikacija nudi.

Ključne riječi: *Java Spring Boot, JavaScript, kućni ljubimci, PostgreSQL, Web trgovina*

Summary

Web Shop for Pets

This paper provides a detailed description of the development of a pet web shop application, designed to offer users an easy way to find and purchase products for dogs, cats and small animals. The application supports three different roles: user, employee and administrator, each offering specific features and privileges that are thoroughly elaborated throughout the paper.

The application is implemented using the Spring Boot framework for the backend, while the database is realized in PostgreSQL. The frontend of the application is developed using HTML, CSS and JavaScript, ensuring a clear and responsive user interface. This paper describes the process of developing the application, with a special emphasis on the technical aspects and functionalities it offers.

Keywords: *Java Spring Boot, JavaScript, pets, PostgreSQL, Web shop*

1. Uvod

Razvoj web shop aplikacija predstavlja ključni dio današnjeg digitalnog tržišta. Iz razloga što pruža korisnicima brz, praktičan i personaliziran način kupovine raznih proizvoda, potražnja za raznovrsnim web shop platformama kontinuirano raste iz dana u dan. Među takvim popularnim platformama nalaze se i web shop aplikacije za kućne ljubimce, koje nude široki spektar proizvoda za razne životinje. Slijedno tome, ovaj rad se fokusira na detaljan proces izrade web shop aplikacije za kućne ljubimce koja će omogućavati jednostavnu navigaciju i kupovinu, uz mnoštvo drugih funkcionalnosti za poboljšanje korisničkog iskustva.

Motivacija za odabir ove teme proizlazi iz ljubavi prema životinjama i želje da se napravi aplikacija koja će korisnicima preko intuitivnog i vizualno atraktivnog korisničkog sučelja omogućiti jednostavno pretraživanje i kupovinu proizvoda za njihove ljubimce.

U radu će se detaljno opisati tehnički aspekti aplikacije, s naglaskom na korištene tehnologije poput Java Spring Boota, PostgreSQL-a, HTML-a, CSS-a i JavaScripta. Također, pružit će se detaljni uvid u arhitekturu baze podataka, koja je zadužena za spremanje i čuvanje podataka nužnih za funkcioniranje aplikacije. Ujedno će biti pružen detaljni opis za korisnička sučelja za različite uloge (korisnik, zaposlenik, administrator) te sve funkcionalnosti vezane za njih.

Završni rad je strukturiran u pet cjelina koje pokrivaju sve ključne aspekte razvoja web shop aplikacije za kućne ljubimce. Prva cjelina obuhvaća uvodni dio koji opisuje motivaciju za izradu aplikacije. Druga cjelina detaljno opisuje korištene tehnologije. Treća cjelina posvećena je opisu baze podataka. Četvrta cjelina opisuje proces izrade aplikacije za različite korisničke uloge, dok se u petoj cjelini opisuje zaključak. Na kraju se navodi literatura korištena pri izradi rada.

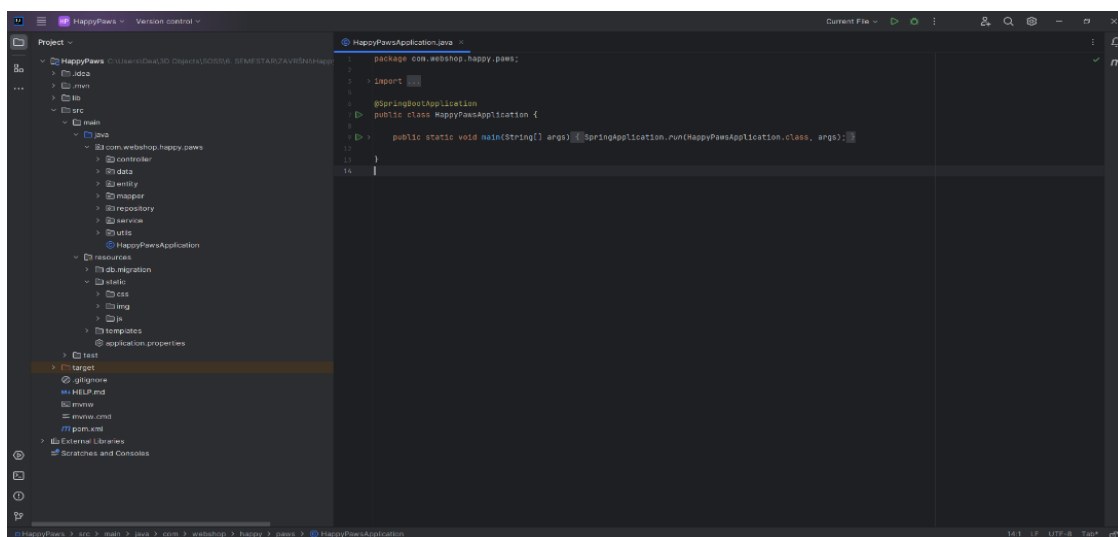
2. Korištene tehnologije

Za izradu funkcionalnog i sigurnog webshopa za kućne ljubimce korištene su tehnologije Java Spring Boot, PostgreSQL, HTML, CSS i JavaScript.

2.1 Java Spring Boot

Java Spring Boot je moćan razvojni okvir za razvoj aplikacija koji se oslanja na Spring Framework. Njegova glavna svrha je pojednostaviti proces razvoja novih aplikacija zasnovanih na Springu (Slika 1). Autokonfiguracija omogućava Spring Bootu da automatski konfigurira aplikaciju na osnovu dodanih biblioteka, čime se smanjuje potreba za ručnim podešavanjem [1].

Ugrađeni poslužitelji (engl. *embedded servers*) kao što su Tomcat, Jetty ili Undertow omogućavaju pokretanje aplikacija kao samostalnih JAR ili WAR datoteka. Spring Boot Starters pruža unaprijed konfigurirane biblioteke za specifične funkcionalnosti, kao što su web razvoj, sigurnost, i baza podataka, što ubrzava razvojni proces [2]. Ujedno se često koristi i Spring Initializr, web alat koji olakšava kreiranje novih Spring Boot projekata s osnovnim konfiguracijama i potrebnim ovisnostima (engl. *dependencies*) [3]. Sve u svemu, Spring Boot je idealan za izradu RESTful web servisa, mikroservisnih arhitektura i drugih složenih aplikacija koje zahtijevaju robusnu, skalabilnu i lako održivu platformu [4].



Slika 1 – Spring Boot framework u IntelliJ IDEA

2.2 PostgreSQL

PostgreSQL je napredan sustav za upravljanje relacijskim bazama podataka (engl. *RDBMS – Relational Database Management System*) otvorenog koda. Poznat po svojoj stabilnosti i pouzdanosti, PostgreSQL je idealan za širok spektar aplikacija, od jednostavnih web stranica do složenih poslovnih sistema. PostgreSQL podržava veliki dio SQL standarda, uključujući složene upite, transakcije, podupite i zajedničke tablice [5].

Spring Boot nudi ugrađenu podršku za PostgreSQL putem Spring Data JPA, što omogućava jednostavnu integraciju i rad s bazom podataka. Autokonfiguracija u Spring Bootu automatski prepoznaje PostgreSQL kao bazu podataka i konfigurira potrebne parametre, čime se smanjuje vrijeme potrebno za postavljanje i omogućava brži početak rada [6].

PostgreSQL omogućava lako izvršavanje CRUD (Create, Read, Update, Delete) operacija, što znači da korisnici mogu kreirati, čitati, ažurirati i brisati podatke u bazi. Ovo je posebno korisno u aplikacijama koje zahtijevaju česte promjene podataka, a Spring Boot pruža alate i biblioteke koje ove operacije čine još jednostavnijima i efikasnijima. PostgreSQL podržava kaskadno brisanje i ažuriranje, što omogućava automatsko održavanje integriteta podataka.

2.3 HTML, CSS

HTML (engl. *Hyper Text Markup Language*) je osnovni jezik za izradu web stranica i web aplikacija. Njegova glavna funkcija je strukturiranje sadržaja na web stranici putem raznih elemenata. Elementi HTML-a su osnovni građevni blokovi koji definiraju različite dijelove sadržaja kao što su naslovi, odlomci, liste, linkovi, slike, tabele i forme. Svaki element je opisan pomoću oznaka (engl. *tag*). HTML omogućava kreiranje interaktivnih formi za unos podataka i podržava raznovrsne tipove unosa kao što su tekst, checkboxovi, radio dugmad, padajuće izbornike (engl. *dropdown menus*) i tekstualna polja. Kroz strukturiranje sadržaja, HTML čini osnovu za prikaz informacija na webu [7].

CSS (engl. *Cascading Style Sheets*) je jezik za stiliziranje HTML elemenata, omogućavajući definiranje izgleda web stranica. Koristeći selektore, CSS definira koji će elementi biti stilizirani. CSS Box Model definira strukturu i raspored elemenata putem margina, obruba, paddinga i sadržaja, omogućavajući precizno podešavanje izgleda stranica. CSS također podržava responzivni dizajn, prilagođavajući izgled stranica različitim veličinama ekrana pomoću Flexboxa i Grida. Kroz kombinaciju različitih stilova i tehnika, CSS omogućava kreiranje vizualno atraktivnih web stranica [8].

2.4 JavaScript

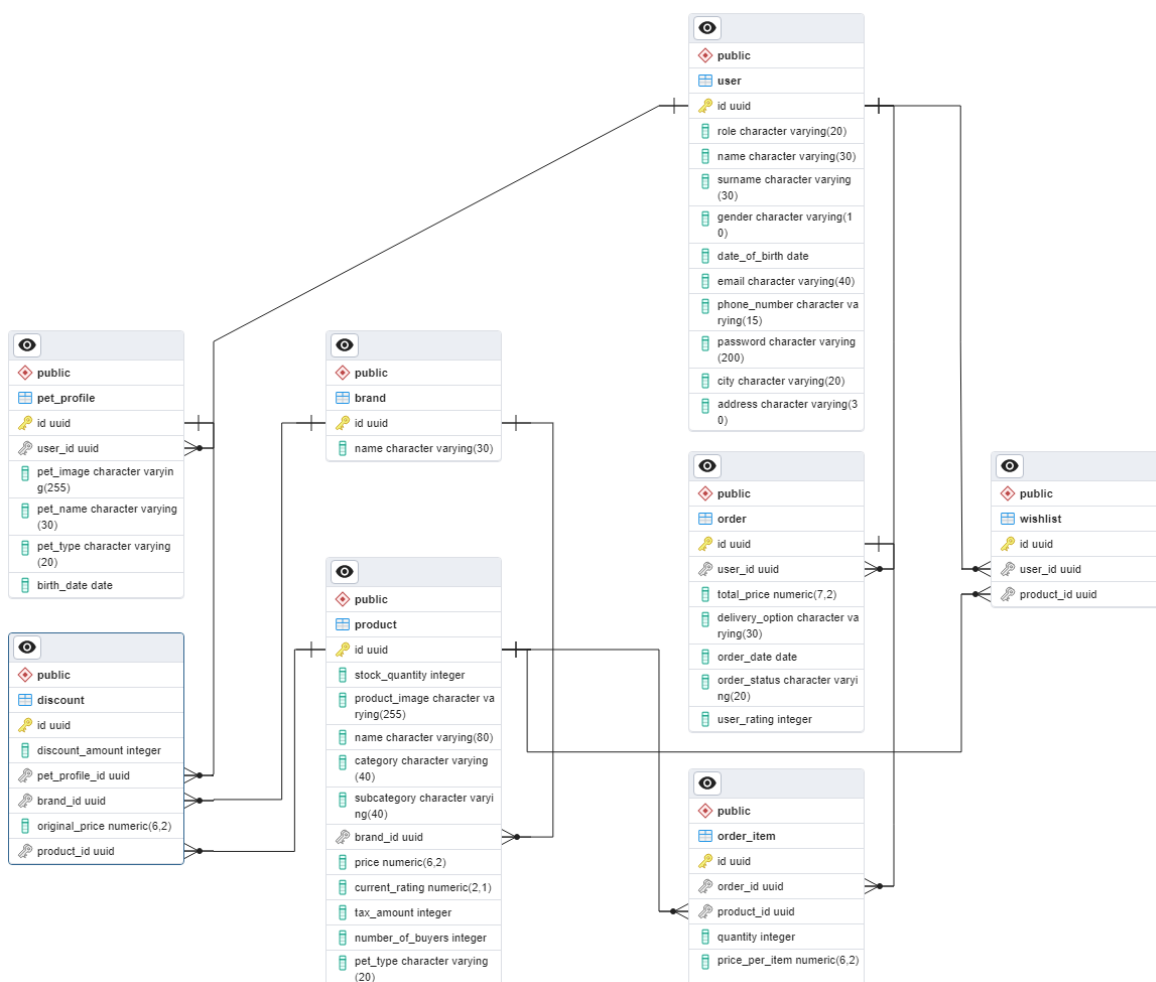
JavaScript je dinamičan programski jezik koji se koristi za dodavanje interaktivnog i dinamičkog sadržaja web stranicama. Može se koristiti na strani klijenta (u web pregledniku) i na strani poslužitelja (putem okruženja kao što je Node.js).

U kontekstu aplikacija razvijenih pomoću Spring Boota, JavaScript igra ključnu ulogu u kreiranju bogatih korisničkih sučelja. Korištenjem JavaScripta, programeri mogu upravljati događajima kao što su klikovi mišem i unos tipkovnice, što omogućava stvaranje interaktivnih korisničkih sučelja. JavaScript također omogućava direktnu manipulaciju DOM (engl. *Document Object Model*) strukture, što omogućava dinamičko ažuriranje sadržaja i strukture web stranica.

Ekosistem JavaScripta obuhvaća brojne biblioteke i okvire kao što su React, Angular i Vue za izgradnju složenih i visoko interaktivnih web aplikacija. JavaScript je neophodan za razvoj modernih web aplikacija sa Spring Bootom, omogućavajući responzivno i interaktivno korisničko iskustvo koje odgovara potrebama korisnika [9].

3. Izrada baze podataka

Osnovna svrha ove baze podataka je upravljanje podacima za web trgovinu specijaliziranu za prodaju proizvoda za kućne ljubimce. Baza podataka omogućava efikasno čuvanje i upravljanje ključnim informacijama, što osigurava sve potrebne funkcionalnosti aplikacije.



Slika 2 – Baza podataka

Naime, baza podataka ove aplikacije se sastoji od sljedećih tablica: user, order, wishlist, order_item, product, discount, brand i pet_profile (Slika 2). Svaka tablica je detaljnije opisana u nastavku.

Tablica `user` čuva informacije o korisnicima aplikacije. Svaki korisnik ima jedinstveni identifikator (UUID), ulogu (`User`, `Staff` ili `Admin`), ime, prezime, spol, datum rođenja, e-mail adresu, broj telefona, šifriranu lozinku, grad i adresu. Grad se bira iz padajućeg izbornika preko korisničkog sučelja kako bi se osigurala konzistentnost podataka. Također, svaki korisnik sa ulogom `User` ima mogućnost izmjene svojih podataka putem osobnog profila, gdje mu se ujedno nudi i opcija trajnog brisanja računa.

Tablica `brand` čuva informacije o brendovima proizvoda. Svaki brend ima jedinstveni identifikator (UUID) i ime. Ime postojećeg brenda se bira iz padajućeg izbornika preko zaposleničkog i administratorskog sučelja prilikom dodavanja i ažuriranja proizvoda, što pomaže u održavanju integriteta podataka. S druge strane, dodavanje, ažuriranje i brisanje brendova je omogućeno samo preko administratorskog sučelja.

Tablica `product` čuva informacije o proizvodima. Svaki proizvod ima jedinstveni identifikator (UUID), količinu na zalih, sliku, naziv, vrstu kućnog ljubimca za kojeg je namijenjen, kategoriju, potkategoriju, identifikator brenda (FK), cijenu, trenutnu ocjenu, iznos poreza i broj kupaca koji su kupili proizvod. Vrsta kućnog ljubimca se odabire iz padajućeg izbornika preko korisničkog sučelja, a ponuđene opcije su pas, mačka i male životinje. Cijene su izražene u eurima (€), a ocjene proizvoda su izražene u decimalnom obliku, pružajući preciznije informacije o kvaliteti proizvoda. Pomoću atributa `stock-quantity` može se pratiti dostupnost proizvoda za kupnju. Prilikom svake nove kupnje ova se vrijednost smanjuje, a samo zaposlenik i administrator imaju mogućnost ažuriranja vrijednosti preko korisničkog sučelja. Radi sigurnosti, uvedena je funkcionalnost koja spriječava da korisnik može kupiti veću količinu nekog proizvoda nego što je dostupno. Ukoliko je proizvod nedostupan, opcija dodavanja proizvoda u košaricu je onemogućena. `Category` i `subcategory` omogućavaju korisnicima detaljniju pretragu i filtriranje proizvoda kako bi bili u mogućnosti brže doći do ciljanih rezultata. Također, `brand_id` služi kao strani ključ prema entitetu `brand` gdje se nalazi izlist svih pohranjenih brendova. Trenutna ocjena proizvoda je spremljena u decimalnom obliku i temelji se na prosjeku svih korisničkih recenzija za taj proizvod. Ako trenutačno nema pohranjenih recenzija za taj proizvod, bit će označeno da proizvod još nije ocijenjen.

Tablica `order` čuva informacije o narudžbama korisnika. Svaka narudžba ima jedinstveni identifikator (UUID), identifikator korisnika koji je izvršio narudžbu (FK), ukupnu cijenu, opciju dostave, datum narudžbe, status narudžbe i ocjenu korisnika. Opcija dostave se bira neposredno prije izrade narudžbe pomoću padajućeg izbornika, a ocjene proizvoda tek mogu biti unešene kada status narudžbe bude „dostavljeno“ (engl. *delivered*). To osigurava da su ocjene relevantne i da se temelje na isporučenim narudžbama. Moguća ocjena proizvoda se kreće od 1 do 5.

Tablica `order_item` čuva informacije o stavkama unutar narudžbi. Svaka stavka ima jedinstveni identifikator (UUID), identifikator narudžbe kojoj pripada (FK), identifikator proizvoda (FK), količinu, te cijenu po stavci. Cijena po stavci je izražena u eurima. Glavna zadaća ove tablice je praćenje detalja svake pojedinačne stavke u narudžbi.

Tablica `pet_profile` čuva informacije o profilima kućnih ljubimaca korisnika. Svaki profil ima jedinstveni identifikator (UUID), identifikator korisnika kojem pripada (FK), sliku, ime, vrstu kućnog ljubimca i datum rođenja. Aplikacija prepoznaje rođendane kućnih ljubimaca te im čestita rođendan, pritom sugerirajući korisniku koji je povezan sa tim profilom da im kupi nešto kao poklon.

Tablica `wishlist` čuva informacije o proizvodima s liste želja korisnika. Svaka lista želja ima jedinstveni identifikator (UUID), identifikator korisnika kojem pripada (FK) i identifikator proizvoda (FK). Korisnici mogu automatski dodavati proizvode iz liste želja u košaricu, što olakšava proces kupovine.

Tablica `discount` čuva informacije o popustima. Svaki popust ima jedinstveni identifikator (UUID), iznos popusta, identifikator profila kućnog ljubimca (FK), identifikator brenda (FK), originalnu cijenu i identifikator proizvoda (FK). Upravljanje popusta je moguće preko zaposleničkog i administratorskog sučelja.

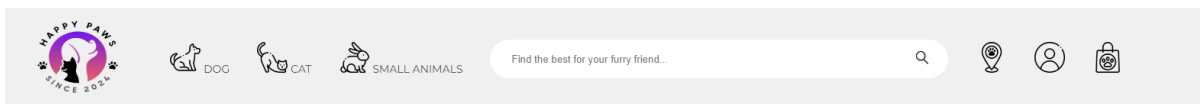
4. Izrada aplikacije

4.1 Izrada korisničkog sučelja

Neregistrirani korisnik može pregledavati proizvode na web shop aplikaciji, pretraživati ih putem pretraživača, filtrirati ih po kategoriji, podkategoriji i brendu, sortirati po cijeni te ih dodavati u košaricu. Međutim, za pristup dodatnim funkcionalnostima, poput plaćanja, pregleda povijesti narudžbi i ostavljanja recenzija, potrebna je registracija korisnika.

4.1.1 Izrada korisničkog računa i prijava

Kada neregistrirani korisnik pristupi stranici, navigacijska traka mu nudi nekoliko poveznica, koje uključuju logotip, opcije za odabir proizvoda za pse, mačke i male životinje, traku za pretraživanje (engl. *search bar*), pregled lokacije trgovine, opciju za prijavu (engl. *login*) te pregled košarice (engl. *shopping bag*) (Slika 3).



Slika 3 – Navigacijska traka neprijavljenog korisnika

Klikom na logotip, korisnik će se vratiti na početnu stranicu za pregled proizvoda. Međutim, opcije gdje se nalaze slike specifičnih životinja vode direktno na stranicu sa proizvodima namijenjenih toj vrsti životinje. Traka za pretraživanje omogućuje korisnicima jednostavno pronalaženje željenih proizvoda na brz i jednostavan način. Pregled lokacije pruža informaciju o točnoj lokaciji trgovine putem ugrađene Google Maps karte.

Naime, opcija za prijavu podrazumijeva prijavu na postojeći račun, dok neregistrirani korisnici imaju mogućnost odabrati opciju registracije i izraditi novi račun. Preko korisničkog sučelja mogu se kreirati samo računi s ulogom `USER`, dok se zaposlenički i administratorski računi kreiraju direktno u bazi podataka. Pregled košarice omogućuje

korisnicima da vide sve proizvode koje su dodali, uključujući njihove količine te ukupnu cijenu.

Registration Form

Already have an account? [Login here.](#)

Name *	Surname *
<input type="text"/>	<input type="text"/>
Gender *	E-mail *
<input type="text" value="Select Gender"/>	<input type="text"/>
Phone Number	Date of Birth *
<input type="text"/>	<input type="text" value="dd/mm/yyyy"/>
City *	Address *
<input type="text" value="Select City"/>	<input type="text"/>
Password *	Confirm Password *
<input type="text"/>	<input type="text"/>

REGISTER

* Required field

Slika 4 – Forma za registraciju

Što se tiče registracije (engl. *register*), korisnik ispunjava formu sa svojim podacima, pritom pazeći na obavezna polja označena crvenom zvijezdicom (Slika 4). Nakon što korisnik popuni formu i klikne na gumb za registraciju, podaci se šalju na poslužitelj gdje metoda `register` obrađuje POST zahtjev za registraciju.

Prvo se provjerava podudaranje unesenih lozinki. Ako lozinke nisu jednake, baca se iznimka i proces se zaustavlja. Potom se vrši provjera je li uneseni e-mail već u upotrebi. Ako je registracija uspješna, korisniku se postavlja uloga `User`, sprema se u sesiju kao prijavljeni korisnik i preusmjerava na početnu stranicu. U slučaju greške tijekom registracije, poruka o grešci dodaje se u model zajedno s podacima koje je korisnik unio, kako bi mogli prikazati u formi registracije (Ispis 1).

Ovaj pristup omogućuje korisnicima jednostavnu registraciju putem web sučelja, uz osiguranje da su svi potrebni podaci uneseni i da se lozinke podudaraju.

```

@PostMapping("/register")
public String register(@ModelAttribute UserData request,
@RequestParam String confirmPassword, Model model, HttpSession
session) {
    try {
        if (!request.getPassword().equals(confirmPassword)) {
            throw new RuntimeException("Passwords do not match.");
        }

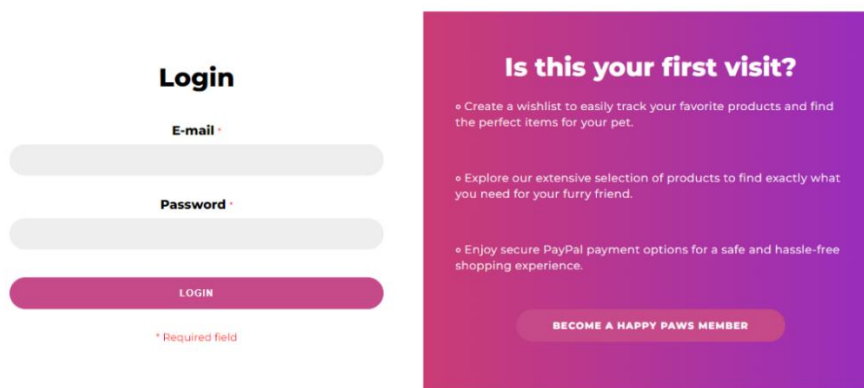
        Optional<User> existingUserOptional =
userRepository.findByEmail(request.getEmail());
        if (existingUserOptional.isPresent()) {
            throw new RuntimeException("E-mail already exists.");
        }

        request.setRole("User");
        User registeredUser = userService.register(request);
        if (registeredUser == null) {
            return "error";
        }
        session.setAttribute("loggedInUser", registeredUser);
        return "redirect:/";
    } catch (RuntimeException e) {
        model.addAttribute("error", e.getMessage());
        model.addAttribute("registerRequest", request);
        return "register";
    }
}

```

Ispis 1 – Funkcija za registriranje na poslužitelju

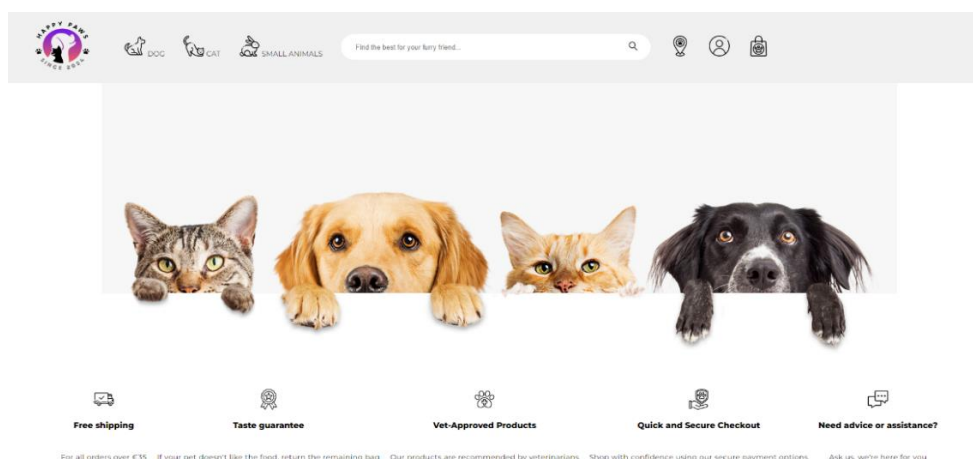
S druge strane, prilikom prijave se od korisnika traži unos e-maila i lozinke (Slika 5). Prvo se provjerava lozinka kako bi se utvrdilo odgovara li šifriranoj verziji pohranjene u bazi podataka za dani e-mail. Ako je autentikacija uspješna, korisnika se sprema u sesiju pod ključem `loggedInUser`. Potom ga se, ovisno o ulozi korisnika, preusmjerava na odgovarajuću početnu stranicu. Ukoliko je autentikacija neuspješna, poruka o grešci se sprema u model kako bi se mogla ispisati na samoj formi za prijavu.



Slika 5 – Forma za prijavu

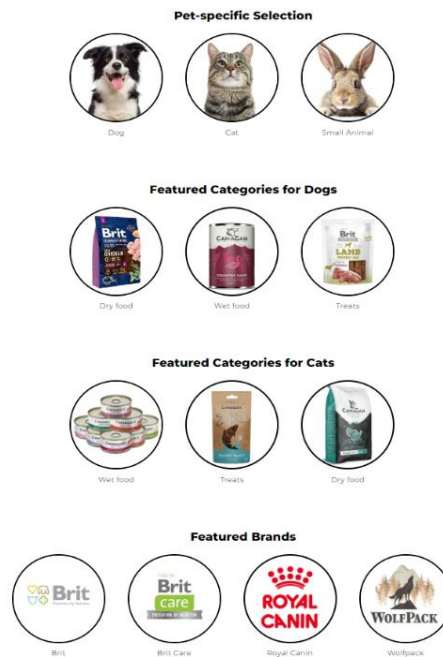
4.1.2 Početna stranica

Na početnoj stranici (engl. *home page*) nalazi se kratki opis svih značajnih karakteristika ove web trgovine. Navigacijska traka omogućuje korisnicima detaljnije istraživanje asortimana proizvoda vezanih za određenu životinju, dok traka za pretraživanje omogućava brzo pronalaženje željenih artikala prema nazivu proizvoda. Također, na raspolaganju su opcije za pregled lokacije trgovine, prijavu na korisnički račun te uvid u sadržaj košarice (Slika 6).



Slika 6 – Početna stranica aplikacije

Ispod ovog opisa trgovine nalazi se brzi izbornik za jednostavni i efikasni pregled traženih proizvoda. Zajedno s opcijom odabira proizvoda za određenog kućnog ljubimca, nudi se opcija odabira proizvoda po određenim kategorijama te pregled proizvoda po određenom brendu (Slika 7).

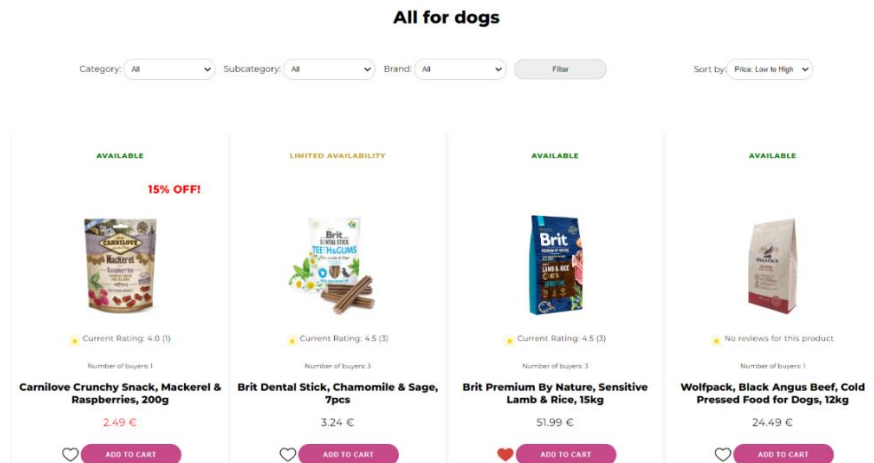


Slika 7 – Pregled brzog izbornika

4.1.3 Pregled proizvoda

Korisnicima se nudi širok spektar proizvoda prilagođenih njihovim ljubimcima. Radi lakšeg pronalaska ciljanog proizvoda, nudi se mogućnost korištenja filtera za odabir proizvoda po kategoriji, podkategoriji i brendu, uz mogućnost sortiranja po cijeni (Slika 8).

Informacije o proizvodima uključuju i stanje zaliha (engl. *stock quantity*), što utječe na dostupnost proizvoda za kupovinu. Prilikom svake kupnje se automatski umanjuje za odabranu količinu kupljenih proizvoda, a boja statusa zaliha mijenja se dinamično: zeleno za više od 10 dostupnih jedinica, žuto za 10 ili manje te crveno ako je proizvod nedostupan. U slučaju nedostupnosti određenog proizvoda, gumb za dodavanje u košaricu postaje onemogućen (engl. *disabled*).



Slika 8 – Pregled proizvoda za pse

Svaki proizvod je opremljen slikom koja ga vizualno predstavlja te recenzijama korisnika prikazanim u decimalnom obliku sa ukupnim brojem ostavljenih recenzija. Ako proizvod nema recenzija, naznačeno je da još nije ocijenjen. Naime, ako je primijenjen popust na određene proizvode, iznos popusta bit će prikazan iznad slike proizvoda s desne strane, dok će nova cijena već uključivati popust.

Također, korisnicima su ispod naziva proizvoda na raspolaganju opcije za dodavanje proizvoda u listu želja (engl. *wishlist*) i u košaricu. Važno je napomenuti da neregistrirani korisnici mogu dodavati proizvode u košaricu, dok je za korištenje liste želja i obavljanje kupovina potrebna registracija.

Filteri su napravljeni tako da se opcije poput hrane ili poslastica u kategoriji uključuju direktno u HTML-u pomoću oznake `option value` u padajućem izborniku. Međutim, prilikom odabira određene kategorije, pomoću JavaScripta se samo pojavljuju relevantne opcije za podkategoriju (Ispis 2). S druge strane, lista svih brendova je dohvaćena sa poslužiteljske strane i prikazana u padajućem izborniku.

```

document.addEventListener('DOMContentLoaded', function() {
    const categoryDropdown =
document.getElementById('category');
    const subcategoryDropdown =
document.getElementById('subcategory');
    categoryDropdown.addEventListener('change', function() {
        const selectedCategory = categoryDropdown.value;
        subcategoryDropdown.innerHTML = '';

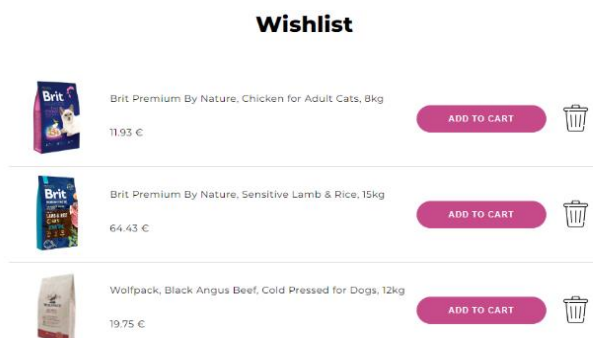
        if (selectedCategory === 'Food') {
            const foodSubcategories = ['All', 'Dry food', 'Wet
food'];
            foodSubcategories.forEach(function(subcategory) {
                const option =
document.createElement('option');
                option.value = subcategory;
                option.text = subcategory;
                subcategoryDropdown.appendChild(option);
            });
        } else if (selectedCategory === 'Treats') {
            const treatsSubcategories = ['All', 'Regular
treats', 'Dental treats'];
            treatsSubcategories.forEach(function(subcategory)
{
                const option =
document.createElement('option');
                option.value = subcategory;
                option.text = subcategory;
                subcategoryDropdown.appendChild(option);
            });
        }
    });
});
});

```

Ispis 2 – JavaScript funkcija za filtriranje po kategoriji i podkategoriji

4.1.4 Lista želja

Svaki korisnik ima mogućnost spremiti svoje omiljene proizvode u listu želja. Ova funkcionalnost omogućava korisnicima da jednostavno prate proizvode koji su im se svidjeli, s opcijom da ih mogu brzo dodati u svoju košaricu ako su dostupni (Slika 9).



Slika 9 – Pregled liste želja

Kada korisnik pošalje zahtjev, metoda provjerava je li korisnik prijavljen. Ako je korisnik prijavljen, metoda koristi `wishlistService` za dodavanje proizvoda u listu želja na temelju ID-a korisnika i proizvoda. Ako korisnik nije prijavljen, preusmjerava ga na stranicu za prijavu. Nakon što se proizvod doda u listu želja, korisnika se vraća na stranicu na kojoj je bio koristeći referer, HTTP zaglavlje koje sadrži URL prethodne stranice s koje je korisnik došao (Ispis 3).

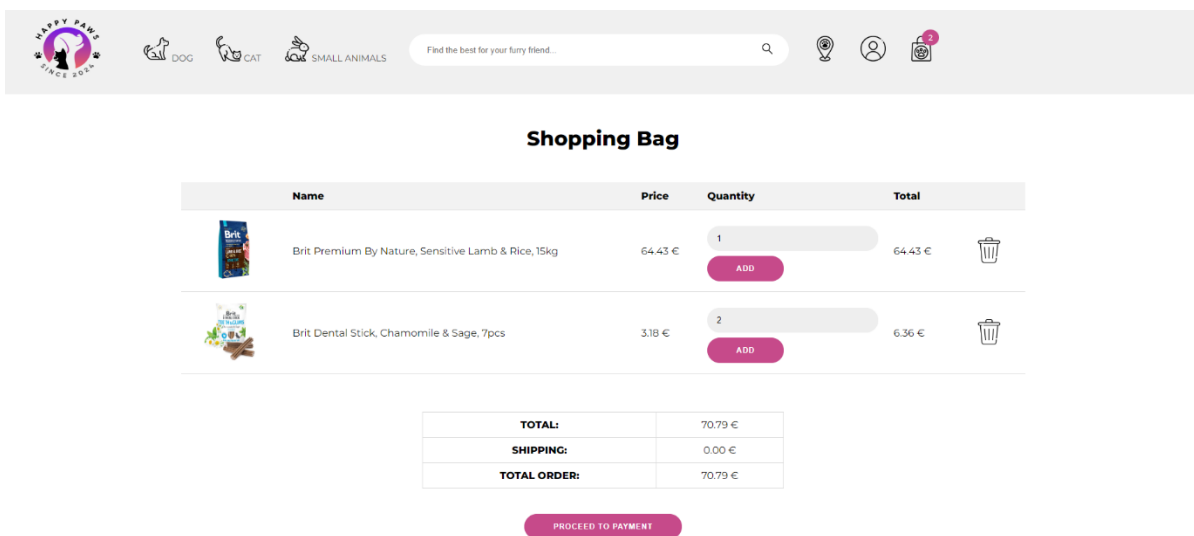

```
@PostMapping("/add-to-wishlist")
public String addToWishlist(@RequestParam UUID productId,
    HttpSession session, HttpServletRequest request) {
    User user = (User) session.getAttribute("loggedInUser");
    if (user != null) {
        wishlistService.addToWishlist(user.getId(), productId);
    } else {
        return "redirect:/login";
    }
    String referer = request.getHeader("Referer");
    if (referer != null && !referer.isEmpty()) {
        return "redirect:" + referer;
    } else {
        return "redirect:/";
    }
}
```

Ispis 3 – Funkcija za dodavanje proizvoda u listu želja

4.1.5 Košarica

U košarici se nalaze svi proizvodi koje je korisnik dodao za kupnju (Slika 10). Tu korisnik može odabrati željenu količinu svakog proizvoda, pri čemu je minimalna količina 1, a maksimalna 100. Međutim, ako je proizvod dostupan u manjoj količini od 100 prema atributu `stock_quantity`, ta vrijednost postaje maksimalna količina koja se može odabrati.

The screenshot shows a shopping bag interface for 'Happy Paws'. At the top, there is a navigation bar with the logo 'HAPPY PAWS SINCE 2014', icons for 'DOG', 'CAT', and 'SMALL ANIMALS', a search bar with the text 'Find the best for your furry friend...', and icons for location, user profile, and a shopping bag with a '2' notification. Below the navigation bar, the title 'Shopping Bag' is centered. The main content is a table with columns 'Name', 'Price', 'Quantity', and 'Total'. It lists two items: 'Brit Premium By Nature, Sensitive Lamb & Rice, 15kg' with a price of 64.43 € and a quantity of 1, and 'Brit Dental Stick, Chamomile & Sage, 7pcs' with a price of 3.18 € and a quantity of 2. Each item has an 'ADD' button and a trash icon. Below the table, a summary table shows 'TOTAL: 70.79 €', 'SHIPPING: 0.00 €', and 'TOTAL ORDER: 70.79 €'. At the bottom, there is a 'PROCEED TO PAYMENT' button.

Name	Price	Quantity	Total
 Brit Premium By Nature, Sensitive Lamb & Rice, 15kg	64.43 €	1	64.43 €
 Brit Dental Stick, Chamomile & Sage, 7pcs	3.18 €	2	6.36 €

TOTAL:	70.79 €
SHIPPING:	0.00 €
TOTAL ORDER:	70.79 €

PROCEED TO PAYMENT

Slika 10 – Pregled košarice

Ukupna cijena svih proizvoda već uključuje iznos poreza i automatski se izračunava s obzirom na količinu. Računa se i iznos dostave koja je besplatna za narudžbe iznad 35 €, a u suprotnom se naplaćuje 4 €. Ukupna cijena koju korisnik plaća predstavlja zbroj cijene svih proizvoda i cijene dostave (Ispis 4).

Korisnik također ima mogućnost brisanja proizvoda iz košarice ukoliko se predomisli. Naime, proizvodi u košarici pohranjeni su u sesiji, što znači da će svi proizvodi biti izgubljeni kada sesija bude prekinuta.

```

document.addEventListener('DOMContentLoaded', function () {
    const quantityInputs = document.querySelectorAll('.quantity-
input');
    quantityInputs.forEach(function (input, index) {
        input.addEventListener('change', function () {
            updateTotal();
        });
    });
    function updateTotal() {
        let total = 0;
        document.querySelectorAll('.shopping-bag-products tbody
tr').forEach(function (row) {
            const price = parseFloat(row.querySelector('td:nth-
child(3)').textContent.replace('€', ''));
            const quantity =
parseInt(row.querySelector('.quantity-input').value);
            const rowTotal = price * quantity;
            total += rowTotal;
            row.querySelector('td.products-total').textContent =
rowTotal.toFixed(2) + ' €';
        });

        const totalElements =
document.querySelectorAll('.shopping-bag-total tbody tr td:first-
child');
        totalElements.forEach(function (element) {
            element.textContent = total.toFixed(2) + ' €';
        });

        const shippingCost = document.getElementById('shipping-
cost');
        const copiedTotal = total.toFixed(2);
        if (copiedTotal >= 35) {
            shippingCost.textContent = '0.00 €';
        } else {
            shippingCost.textContent = '4.00 €';
        }
        const totalOrder = document.getElementById('total-order');
        const totalWithShipping = total + (copiedTotal >= 35 ? 0 :
4);
        totalOrder.textContent = totalWithShipping.toFixed(2) + '
€';
        const copiedTotalElement =
document.getElementById('copied-total');
        copiedTotalElement.textContent = copiedTotal + ' €';
    }

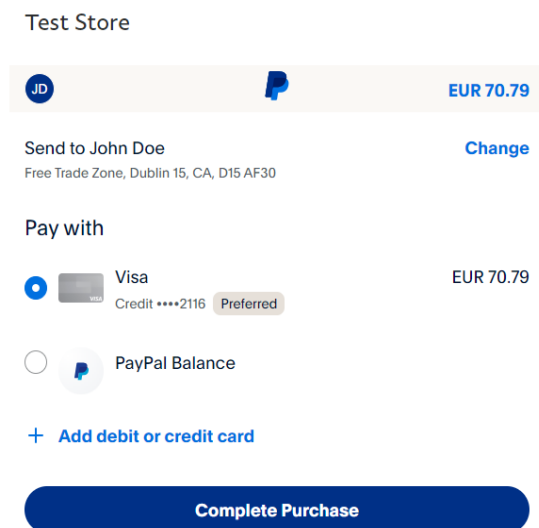
    updateTotal();
});

```

Ispis 4 – JavaScript funkcija za izračunavanje cijena u košarici

4.1.6 Plaćanje narudžbe i pregled narudžbe

Nakon što korisnik potvrdi sadržaj košarice i provjeri osobne podatke, dolazi do plaćanja narudžbe. Kada korisnik pritisne gumb sa slikom PayPal logotipa, odvodi ga se na eksternu PayPal stranicu gdje će se ulogirati, odabrati željeni način plaćanja te potvrditi podatke za plaćanje (Slika 11).



Slika 11 – Pregled plaćanja narudžbe preko PayPal računa

PayPal plaćanje je integrirano tako da stvara i vraća mapu u koju sprema parametar `mode`, trenutno podešen na `sandbox`. U ovom slučaju, za testiranje PayPal plaćanja korišteni su privatni (engl. *private*) i poslovni (engl. *business*) sandbox računi. Sandbox računi su simulirani PayPal računi koji omogućuju programerima da testiraju integracije bez stvarnog novca ili financijskih transakcija.

Autentikacija se obavlja pomoću `OAuthTokenCredential` metode, koja koristi ID klijenta i njegovu tajnu lozinku za kreiranje pristupnog tokena. Naime, sama komunikacija s PayPal API-jem primjenjuje se u metodi `APIContext` koja sadrži mapu i token dobivene iz prethodnih metoda (Ispis 5).

```

@Configuration
public class PayPalConfig {

    @Bean
    public Map<String, String> paypalSdkConfig() {
        Map<String, String> configMap = new HashMap<>();
        configMap.put("mode", "sandbox");
        return configMap;
    }

    @Bean
    public OAuthTokenCredential oAuthTokenCredential() {
        String clientId = "AVkAhOYaBciCGogGPIZRorR4rtCT-n0MzkCDIz-
Tmv5K_OkLAPbfMBq4AnGM_YQpxpRu-o8JMe-iUXYzS";
        String clientSecret =
"EL4ix8bCIIEMYGWKYqm46W8h_abUgcMoxyMIRc9vMLL__6SX5doP4Vjvxxg70sEly
A7LsM-hwGGp0swk";
        return new OAuthTokenCredential(clientId, clientSecret,
paypalSdkConfig());
    }

    @Bean
    public APIContext apiContext() throws PayPalRESTException {
        APIContext context = new
APIContext(oAuthTokenCredential().getAccessToken());
        context.setConfigurationMap(paypalSdkConfig());
        return context;
    }
}

```

Ispis 5 – Konfiguracija PayPal plaćanja u Spring Bootu

Nakon što izvrši plaćanje, korisnik će biti preusmjeren na stranicu gdje mu je ispisana potvrda PayPal transakcije. Na toj stranici također može odabrati način dostave (Pickup, GLS Delivery, GLS Pickup) bez dodatnih troškova i finalizirati narudžbu. Nakon toga se stvara potvrda o narudžbi koja sadrži sažetak narudžbe sa svim relevantnim detaljima. Narudžba se zatim može pratiti na korisnikovom osobnom profilu, gdje se nalazi

poveznica za stranicu `My Orders`. Na ovoj stranici korisnik može pregledati sve prethodne narudžbe (Slika 12).

My Orders

Order No.	Total Price	Delivery Option	Order Date	Status	
15355bd8-0894-4444-b627-2cf2be007d44	28.99 €	Pickup	2024-05-28	Delivered	Rate the product: Rated: 5
74a11087-c77c-47b8-ab9e-e5f32221606b	35.91 €	Pickup	2024-05-28	Delivered	Rate the product: Rated: 5
76f1953f-c757-42e1-8447-11299243b955	10.13 €	Pickup	2024-05-28	Delivered	Rate the product: Rated: 4
b9b07d8c-74ba-41a3-b9f7-abf796cbcd82	70.78 €	GLS Delivery	2024-07-09	Processing	
cf3ea9ae-b44c-4e0a-89c4-293dafd392ff	44.28 €	GLS Pickup	2024-06-10	In Transit	

Slika 12 – Pregled svih narudžbi korisnika

Svaka narudžba prikazuje broj narudžbe, ukupnu cijenu, način dostave, vrijeme naručivanja te trenutni status koji je označen različitim bojama radi lakše identifikacije promjena. Prilikom novog naručivanja, status narudžbe bit će `Processing`. Nakon određenog vremena, status će se promijeniti u `In Transit`, što označava da je pošiljka na putu prema odredištu. Konačno, kada pošiljka stigne na svoje odredište, status će biti `Delivered`. Tada će korisnik moći ocijeniti svoje iskustvo s narudžbom i proizvodima ocjenom od 1 do 5.

Ocjenjivaje proizvoda započinje provjerom postoji li već unesena vrijednost za `userRating` atribut u tablici `orders`. Ukoliko ne postoji, odabrana ocjena narudžbe sprema se u bazu podataka. Potom se dohvaća lista svih proizvoda te narudžbe pomoću `orderId` i iterira se kroz svaku zasebnu stavku. U priloženoj funkciji se zbrajaju sve prijašnje ocjene korisnika za taj proizvod i naposljetku dijele s ukupnim brojem ocjena. Na takav način se dolazi do prosječne ocjene proizvoda koja je zaokružena na jednu decimalu. Ujedno se i ažurira atribut `currentRating` proizvoda s novo izračunatom prosječnom ocjenom. (Ispis 6).

```

        List<OrderItem> orderItems =
orderItemRepository.findByOrderId(order.getId());

        for (OrderItem orderItem : orderItems) {
            Product product = orderItem.getProduct();
            if (product != null) {
                List<OrderItem> allOrderItems =
orderItemRepository.findByProductId(product.getId());

                double totalRating = 0.0;
                int numberOfRatings = 0;
                for (OrderItem item : allOrderItems) {
                    Order associatedOrder = item.getOrder();
                    if (associatedOrder.getUserRating() != null) {
                        totalRating +=
associatedOrder.getUserRating();
                        numberOfRatings++;
                    }
                }

                if (numberOfRatings > 0) {
                    double averageRating = totalRating /
numberOfRatings;

product.setCurrentRating(BigDecimal.valueOf(averageRating).setScale(1, RoundingMode.HALF_UP));
                    productRepository.save(product);
                }
            }
        }
        return "redirect:/my-order";
    }
    return "error";
}

```

Ispis 6 – Isječak funkcije za ocjenjivanje proizvoda

4.1.7 Korisnički profil

Korisnički profil pruža korisniku mogućnost pregleda i ažuriranja vlastitih podataka, kao i opciju trajnog brisanja profila (Slika 13). Pregled osobnih podataka obuhvaća informacije poput imena, kontaktnih podataka, mjesta stanovanja, povijesti narudžbi i profila za ljubimce. Korisnik može jednostavno izmijeniti ove podatke ukoliko se nešto promijenilo, poput adrese ili broja telefona. Ažuriranje podataka osigurava da su informacije u korisničkom profilu uvijek točne i relevantne, što je važno za obradu narudžbi.

The screenshot displays the 'My Account' interface. At the top, there are two tabs: 'PET PROFILE' (active) and 'MY ORDERS'. Below the tabs is a greeting 'Hello Dea!' followed by a welcome message: 'Welcome to your profile! Here, you have access to all your relevant data, which you can conveniently view, manage and update as needed. Feel free to explore your pet's profile and check out any recent orders you've made.' The main section is titled 'Update My Profile' and includes a descriptive paragraph: 'This form provides a seamless way for users to view and update their account information within the application. You can easily modify details like email, phone number, address and password.' The form contains several input fields: 'Email' (deboocna@gmail.com), 'Phone Number' (0951234567), 'City' (Split), 'Address' (Zladrina potkana 3), 'New Password' (Enter new password), and 'Confirm New Password' (Confirm new password). A pink 'UPDATE' button is positioned below the form. At the bottom, there is a 'Delete Profile' section with a warning: 'If you wish to permanently delete your profile, please click the button below. Please note that this action cannot be undone, and all your data will be lost.' A pink 'DELETE PROFILE' button is located at the end of this section.

Slika 13 – Pregled osobnog profila korisnika

Trajno brisanje profila omogućuje korisniku da ukloni sve svoje podatke iz sustava (Ispis 7). Ova opcija je korisna u situacijama kada korisnik više ne želi koristiti usluge web trgovine ili želi povući sve svoje osobne podatke iz sustava. Ova funkcionalnost pruža korisnicima kontrolu nad njihovim podacima i privatnošću, osiguravajući da imaju mogućnost brisanja svojih podataka ako to odluče. Uz to se pomoću metode `session.invalidate()` uništava trenutna sesija korisnika, što znači da se svi atributi iz te sesije uklanjaju.

```

@PostMapping("/delete-user-profile")
public String deleteProfile(HttpSession session) {
    User loggedInUser = (User)
    session.getAttribute("loggedInUser");

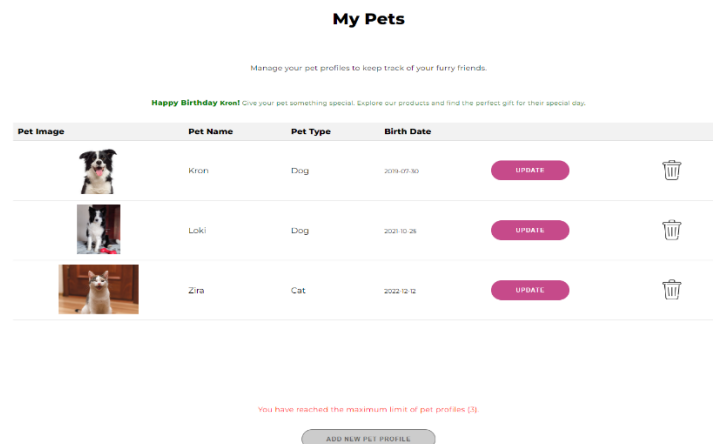
    if (loggedInUser != null) {
        userRepository.delete(loggedInUser);
        session.invalidate();
    }
    return "redirect:/";
}

```

Ispis 7 – Funkcija za trajno brisanje korisničkog računa

4.1.8 Profili za ljubimce

Još jedna od funkcionalnosti koja se nudi registriranim korisnicima su profili za ljubimce (engl. *pet profiles*), koji omogućavaju korisnicima upravljanje informacijama o svojim ljubimcima (Slika 14). Svaki korisnik može istovremeno imati do tri profila ljubimca, gdje za svaki profil korisnik može dodati sliku svog ljubimca, unijeti ime ljubimca, odabrati tip ljubimca (pas, mačka, mala životinja) te njegov datum rođenja. Aplikacija će prepoznati rođendan korisnikovih ljubimaca, uputiti čestitke i potaknuti kupca da im kupi nešto posebno.



Slika 14 – Pregled profila za kućne ljubimce

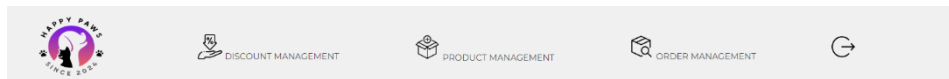
Osim što omogućava dodavanje novih profila ljubimaca, profil za ljubimce također nudi mogućnost ažuriranja postojećih profila. To znači da korisnici mogu mijenjati ili dopunjavati informacije o svojim ljubimcima po potrebi. Također je omogućeno i brisanje profila ljubimca, što korisnicima daje fleksibilnost da uklone profil ljubimca ako žele.

4.2 Izrada zaposleničkog sučelja

Zaposlenici (engl. *staff*) imaju pristup raznim ovlastima koje im omogućuju efikasno upravljanje popustima, proizvodima i narudžbama.

4.2.1 Početna stranica

Na početnoj stranici zaposlenici se susreću s detaljnim pregledom funkcionalnosti koje su im na raspolaganju. To uključuje mogućnost brzog pristupa upravljanju popustima, proizvodima te narudžbama (Slika 15). Ova stranica je centralna točka za sve aktivnosti koje zaposlenici mogu provoditi kako bi osigurali učinkovito vođenje web trgovine.



Slika 15 – Navigacijska traka zaposlenika

4.2.2 Upravljanje proizvodima

Upravljanje proizvodima (engl. *product management*) je ključna funkcionalnost koja omogućuje zaposlenicima detaljno vođenje prodajne ponude (Slika 16). Kroz ovu funkcionalnost, zaposlenici imaju mogućnost ne samo dodavanja novih proizvoda u ponudu (Ispis 8), već i uređivanje postojećih informacija o proizvodima.

To uključuje ažuriranje podataka poput imena, tipa kućnog ljubimca, kategorije, podkategorije, brenda, cijene iznosa poreza, informacija o dostupnosti na skladištu, trenutnih ocjena proizvoda i ukupnog broja kupaca kako bi se osiguralo da kupci uvijek imaju točne i relevantne informacije.

```

@Override
public void addProduct(ProductData productData, Brand brand,
MultipartFile productImage) {
    try {
        String productName =
saveProductImage(productImage);

        Product product = new Product();
        product.setName(productData.getName());
        product.setPetType(productData.getPetType());
        product.setCategory(productData.getCategory());
        product.setSubcategory(productData.getSubcategory());
        product.setBrand(brand);
        product.setPrice(productData.getPrice());
        product.setTaxAmount(productData.getTaxAmount());
        product.setStockQuantity(productData.getStockQuantity());
        product.setTaxAmount(25);
        product.setCurrentRating(BigDecimal.valueOf(0.0));
        product.setNumberOfBuyers(0);
        product.setProductImage(productImageName);

        productRepository.save(product);
    } catch (IOException e) {
        throw new RuntimeException("Error saving product
image", e);
    }
}

```

Ispis 8 – Funkcija za dodavanje proizvoda

Svaki proizvod je kategoriziran prema vrsti životinje (engl. *pet type*), što omogućuje ciljanu pretragu proizvoda za određenu vrstu ljubimca. Osim toga, proizvodi su opisani pomoću kategorija (engl. *category*), podkategorija (engl. *subcategory*) i brendova

(engl. *brand*), što omogućava preciznije i efikasnije filtriranje. Na ovaj način korisnici mogu lako pronaći proizvode koji odgovaraju specifičnim potrebama njihovih ljubimaca.

Product Management

[ADD NEW PRODUCT](#)

ID	Stock Quantity	Image	Name	Pet Type	Category	Subcategory	Brand	Price	Rating	Tax	Buyers	
a46296f1-3ae0-45d7-8a7d-7850eac2a4a5	143		Carnilove Crunchy Snack, Mackerel & Raspberries, 200g	Dog	Treats	Regular treats	Carnilove	6.54 €	0.7	25%	3	UPDATE
cb3a3ce1-d305-4e03-9209-665cbe24ed9d	28		Wolfpack, Black Angus Beef, Cold Pressed for Dogs, 12kg	Dog	Food	Dry food	WolfPack	15.80 €	0.0	25%	1	UPDATE
3dce2429-29e9-4a8e-8767-85a3736d92c7	4		Brit Dental Stick, Chamomile & Sage, 7pcs	Dog	Treats	Dental treats	Brit	2.54 €	1.7	25%	3	UPDATE
2b4-d9b63-9b97-4b6b-6637-e0bc74b4e0a2	29		Brit Care Superfruits, Lamb, Crunchy Snack for Adult Cats, 100g	Cat	Treats	Dry food	Brit Care	8.54 €	2.0	25%	1	UPDATE
1a5e0724-31a5-4b7c-8d14-15a3d69e014	49		Brit Care Dental, Functional Snack, No Grains, 50g	Cat	Treats	Dental treats	Brit Care	7.54 €	0.7	25%	1	UPDATE
3c8f9a27-471cd-42c1b-bf67-236e2b547a63	19		Brit Premium By Nature, Salmon Fillets for Adult Cats, 85g	Cat	Food	Wet food	Brit	3.54 €	0.7	25%	1	UPDATE

Slika 16 – Pregled funkcionalnosti upravljanja proizvodima

Također, zaposlenici mogu brisati proizvode iz ponude kada je to potrebno, čime se omogućuje dinamičko upravljanje inventarom u skladu s internim poslovnim potrebama. Ova funkcionalnost ne samo da olakšava administraciju proizvodima, već i osigurava da prodajna ponuda bude uvijek ažurirana.

4.2.3 Upravljanje narudžbama

Funkcionalnost upravljanja narudžbama (engl. *order management*) omogućava zaposlenicima da prate stanje svake narudžbe, od trenutka zaprimanja do isporuke kupcu (Slika 17).

Order Management

ID	User	Total	Delivery Option	Order Date	Order Status	
15355bd8-0894-4444-b627-2cf2be007d44	Dea Bočina	28.99 €	Pickup	2024-05-28	Delivered	UPDATE
74a11087-c77c-47b8-ab9e-e5f32221606b	Dea Bočina	35.91 €	Pickup	2024-05-28	Delivered	UPDATE
76f1953f-c757-42e1-8447-11299243b955	Dea Bočina	10.13 €	Pickup	2024-05-28	Delivered	UPDATE
b9b07d8c-74ba-41a3-b9f7-abf796bcbdb82	Dea Bočina	70.78 €	GLS Delivery	2024-07-09	Processing	UPDATE
cf3ea9ae-b44c-4e0a-89c4-293dafd392ff	Dea Bočina	44.28 €	GLS Pickup	2024-06-10	In Transit	UPDATE

Slika 17 – Pregled funkcionalnosti upravljanja narudžbama

Zaposlenici imaju pristup detaljnim informacijama o svakoj narudžbi, uključujući broj narudžbe, podatke o kupcu, ukupnu cijenu, odabrani način dostave, datum narudžbe i njen status. Ova funkcionalnost omogućuje zaposlenicima i brisanje narudžbi po potrebi, čime se osigurava organizacija i preglednost u sustavu.

Također, zaposlenici mogu ažurirati status svake narudžbe kako bi pružili kupcima transparentnost i sigurnost tijekom cijelog procesa kupovine (Ispis 9). Kada se odabere status `Delivered`, zaposlenici više ne mogu mijenjati status narudžbe.

```
@PostMapping("/update-order-status/{id}")
public String updateOrderStatus(@PathVariable("id") UUID id,
@RequestParam("orderStatus") String orderStatus, HttpSession
session, Model model) {
    User user = (User)session.getAttribute("loggedInUser");
    model.addAttribute("user", user);

    Order order = orderRepository.findById(id).orElseThrow(() ->
new IllegalArgumentException("Invalid order Id:" + id));
    order.setOrderStatus(orderStatus);
    orderRepository.save(order);
    return "redirect:/order-management";
}
```

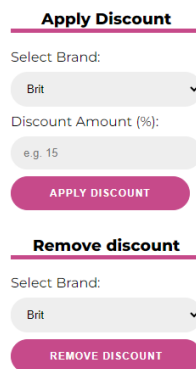
Ispis 9 – Funkcija za ažuriranje statusa narudžbe

4.2.4 Upravljanje popustima

Upravljanje popustima (engl. *discount management*) u web trgovini daje zaposlenicima kontrolu nad definiranjem i upravljanjem popustima (Slika 18).

Može se unijeti iznos popusta i povezati s određenim brendom koristeći selekcijski izbornik. Ovo omogućava zaposlenicima da definiraju specifične popuste koji će se primjenjivati na proizvode tog brenda. Kada se popust unese, odabrani brend automatski će biti povezan s tim popustom, a promjene će se odmah odraziti u cijeloj aplikaciji (Ispis 10).

Discount Management



The image shows a user interface for managing discounts. It is divided into two main sections: 'Apply Discount' and 'Remove discount'. Each section has a 'Select Brand:' label followed by a dropdown menu currently showing 'Brit'. The 'Apply Discount' section also includes a 'Discount Amount (%)' label and a text input field with the placeholder 'e.g. 15'. Below each section is a pink button with white text: 'APPLY DISCOUNT' for the first section and 'REMOVE DISCOUNT' for the second.

Slika 18 – Pregled funkcionalnosti upravljanja popustima

Osim mogućnosti dodavanja novih popusta, zaposlenici također imaju opciju poništavanja postojećih popusta povezanih s određenim brendom. U tom slučaju zaposlenici mogu lako deaktivirati popust putem istog selekcijskog izbornika.

```

@Override
public void applyDiscountToBrand(UUID brandId, int
discountAmount) {
    List<Product> products =
productRepository.findByBrandId(brandId);
    BigDecimal discountFactor =
BigDecimal.ONE.subtract(BigDecimal.valueOf(discountAmount).di
vide(BigDecimal.valueOf(100), 2, RoundingMode.HALF_UP));

    Brand brand =
brandRepository.findById(brandId).orElseThrow(() -> new
RuntimeException("Brand not found"));

    for (Product product : products) {
        Discount discount = new Discount();
        discount.setBrand(brand);
        discount.setDiscountAmount(discountAmount);
        discount.setOriginalPrice(product.getPrice());
        discount.setProduct(product);
        discountRepository.save(discount);

        BigDecimal discountedPrice =
product.getPrice().multiply(discountFactor).setScale(2,
RoundingMode.HALF_UP);
        product.setPrice(discountedPrice);
        productRepository.save(product);
    }
}

```

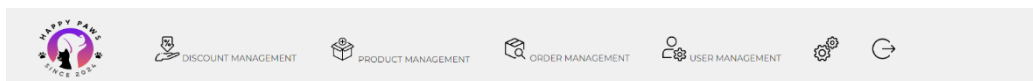
Ispis 10 – Funkcija primjenjivanja popusta na brendove

4.3 Izrada administratorskog sučelja

Administratorsko sučelje pruža sve funkcionalnosti kao i zaposleničko sučelje za upravljanje popustima, proizvodima i narudžbama. Osim toga, administratori imaju dodatne ovlasti za upravljanje korisnicima i sistemskom konfiguracijom.

4.3.1 Početna stranica

Početna stranica administratorskog sučelja pruža centralnu točku za navigaciju kroz aplikaciju, omogućujući administratorima brz i jednostavan pristup svim funkcionalnostima (Slika 19).






Slika 19 – Navigacijska traka administratora

4.3.2 Upravljanje korisnicima

Upravljanje korisnicima (engl. *user management*) u administratorskom sučelju omogućava administratorima detaljan pregled, detaljno uređivanje i sigurno brisanje korisnika s različitim ulogama putem intuitivne i jednostavne forme (Slika 20).

User Management

[ADD NEW USER](#)

ID	Name	Role	Gender	Date of Birth	Email	Phone Number	City	Address		
bf99272a-c135-4496-9a53-f51b0fb95a13	Dea Bočina	User	Female	2000-09-29	deabocina@gmail.com	0955355918	Split	Grabova 9b	UPDATE	
123e4567-e89b-12d3-a456-426614174000	John Doe	Staff	Male	1990-05-15	johndoe@gmail.com	0957233835	Split	Vukovarska 10	UPDATE	
789e4567-e89b-12d3-a456-426614174111	Admin User	Admin	Male	2000-01-01	admin@gmail.com	0957233739	Split	Vukovarska 29	UPDATE	

Slika 20 – Pregled funkcionalnosti upravljanja korisnicima

Svaki korisnik sustava prikazan je s informacijama kao što su ime, uloga, spol, datum rođenja, e-mail, broj mobitela, grad i adresa. Proces dodavanja novih korisnika u sustav ili ažuriranje postojećih podataka odvija se kroz jasno definirane obrasce koji osiguravaju

točnost i dosljednost informacija (Ispis 11). Administratori mogu prilagođavati prava pristupa i uloge korisnika prema potrebama organizacije, osiguravajući da svaki korisnik ima adekvatne ovlasti za obavljanje svojih zadataka. Brisanje korisnika iz sustava provodi se sigurno i brzo.

```
@PostMapping("/add-user")
public String addUser(@ModelAttribute UserData request,
@RequestParam String confirmPassword, Model model, HttpSession
session) {
    try {
        if (!request.getPassword().equals(confirmPassword)) {
            throw new RuntimeException("Passwords do not match");
        }
        User registeredUser = userService.register(request);
        if (registeredUser == null) {
            return "error";
        }
        session.setAttribute("loggedInUser", registeredUser);
        return "redirect:/user-management";
    } catch (RuntimeException e) {
        model.addAttribute("error", e.getMessage());
        model.addAttribute("registerRequest", request);
        return "user-management-add";
    }
}
```

Ispis 11 – Funkcija za dodavanja novih korisnika

4.3.3 Upravljanje sistemskom konfiguracijom

Upravljanje sistemskom konfiguracijom obuhvaća ključne opcije kao što je promjena iznosa poreza (engl. *tax amount*), što je lako izvršiti putem posebne administratorske stranice (Slika 21). Trenutni iznos poreza prikazuje se automatski i može se jednostavno ažurirati prema trenutnim poreznim stopama. Promjenom iznosa poreza automatski se prilagođavaju cijene proizvoda kako bi odražavale nove porezne stope, osiguravajući dosljednost u obračunu cijena.

System Configuration

TAX

Tax Percentage (%):

UPDATE

BRANDS

Add Brand

Brand Name:








SAVE

MANAGE ALL BRANDS

Slika 21 – Pregled systemske konfiguracije

Osim toga, administratori imaju mogućnost dodavanja novih brendova u aplikaciju, uređivanja njihovih podataka radi ispravljanja eventualnih grešaka i brisanja brendova iz sustava prema potrebama (Slika 22). Ova funkcionalnost jamči točnost i pouzdanost podataka unutar aplikacije.

Brand Update & Delete

Brand Name	
Brit	
Brit Care	
Carnilove	
WolfPack	
Canagan	
Royal Canin	
Acana	

Update Brands

Select Brand:

Brand Name:

UPDATE

Slika 22 – Pregled upravljanja brendovima

5. Zaključak

Završni rad predstavlja implementaciju web trgovine specijalizirane za kućne ljubimce. Za efikasno kreiranje ovog funkcionalnog sustava bilo je potrebno koristiti tehnologije poput PostgreSQL, Java Spring Boot, HTML, CSS i JavaScript te time predočiti praktičnu primjenu znanja iz područja web razvoja (engl. *web development*). Glavna svrha aplikacije je omogućiti korisnicima jednostavni pregled, odabir i kupnju proizvoda za njihove ljubimce.

PostgreSQL se koristi za skladištenje i upravljanje podacima nužnim za funkcioniranje aplikacije, poput podataka o korisnicima, proizvodima i narudžbama. Za osnovu poslužiteljskog dijela aplikacije koristi se Java Spring Boot, koji primarno služi za obradu zahtjeva korisnika, upravljanje sesijama i interakciju s bazom podataka. S druge strane, na klijentnom dijelu aplikacije, HTML, CSS i JavaScript omogućuju kreiranje intuitivnog, funkcionalnog i estetski privlačnog korisničkog sučelja.

Za buduća poboljšanja, predlaže se implementacija dodatnih sigurnosnih mjera, optimizacija performansi aplikacije te razvoj mobilne verzije webshopa kako bi se zadovoljile potrebe šireg spektra korisnika. Može se ujedno implementirati detaljnija pretraga i filtriranje, više vrsta popusta te prikazivanje preporučenih proizvoda na temelju povijesti narudžba. Uz to, bilo bi poželjno povezati aplikaciju s društvenim aplikacijama poput Instagrama i Facebooka kako bi se proizvodi efikasnije reklamirali i prodavali.

Sam značaj aplikacije je vidljiv u jednostavnoj administraciji, intuitivnim korisničkim sučeljem te sposobnosti da se prilagodi trenutnim tržišnim trendovima. Sve što je ovdje navedeno čini web trgovinu za kućne ljubimce ključnim alatom za uspješno poslovanje u današnjem digitalnom okruženju.

Literatura

- [1] Spring, „Spring Boot“,
<https://spring.io/projects/spring-boot> (posjećeno 25.06.2024.)
- [2] Baeldung, „Intro to Spring Boot Starters“,
<https://www.baeldung.com/spring-boot-starters> (posjećeno 25.06.2024.)
- [3] Spring, „Spring Initializr“,
<https://start.spring.io/> (posjećeno 26.06.2024.)
- [4] Spring, „Building an Application With Spring Boot“,
<https://spring.io/guides/gs/spring-boot> (posjećeno 26.06.2024.)
- [5] PostgreSQL, „PostgreSQL: About“,
<https://www.postgresql.org/about/> (posjećeno 27.06.2024.)
- [6] Medium, „Spring Boot + Hibernate + PostgreSQL“,
<https://medium.com/javarevisited/spring-boot-hibernate-postgresql-example-4e580ec2020b> (posjećeno 27.06.2024.)
- [7] W3Schools, „HTML Introduction“,
https://www.w3schools.com/html/html_intro.asp (posjećeno 28.06.2024.)
- [8] W3Schools, „CSS Introduction“,
https://www.w3schools.com/css/css_intro.asp (posjećeno 28.06.2024.)
- [9] GeeksForGeeks, „What is Javascript?“,
<https://www.geeksforgeeks.org/javascript/> (posjećeno 29.06.2024.)