Izrada mrežne trgovine za prodaju lokalnih proizvoda

Sić, Karla

Undergraduate thesis / Završni rad

2022

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **Polytechnic of Međimurje in Čakovec / Međimursko veleučilište u Čakovcu**

Permanent link / Trajna poveznica: https://urn.nsk.hr/urn:nbn:hr:110:628648

Rights / Prava: In copyright / Zaštićeno autorskim pravom.

Download date / Datum preuzimanja: **2024-05-20**

Repository / Repozitorij:

Polytechnic of Međimurje in Čakovec Repository - Polytechnic of Međimurje Undergraduate and Graduate Theses Repository

MEĐIMURSKO VELEUČILIŠTE U ČAKOVCU
PREDDIPLOMSKI STRUČNI STUDIJ RAČUNARSTVO

Karla Sić

IZRADA MREŽNE TRGOVINE ZA PRODAJU LOKALNIH PROIZVODA

ZAVRŠNI RAD

ČAKOVEC, rujan 2022.

MEĐIMURSKO VELEUČILIŠTE U ČAKOVCU
PREDDIPLOMSKI STRUČNI STUDIJ RAČUNARSTVO

Karla Sić

IZRADA MREŽNE TRGOVINE ZA PRODAJU LOKALNIH PROIZVODA

CREATION OF AN ONLINE STORE FOR THE SALE OF LOCAL PRODUCTS

ZAVRŠNI RAD

Mentor:
dr. sc. Sanja Brekalo, prof. v. š.

ČAKOVEC, rujan 2022.

MEĐIMURSKO VELEUČILIŠTE U
ČAKOVCU ODBOR ZA ZAVRŠNI RAD

Čakovec, 12. svibnja 2022.

ZAVRŠNI ZADATAK br. 2021-RAC-I-1 16

Pristupnik: Karla Sić (0313014023)

Studij: izvanredni preddiplomski stručni studij Računarstvo

Smjer: Inženjerstvo računalnih sustava i mreža

Zadatak: Izrada mrežne trgovine za prodaju lokalnih proizvoda

Opis zadatka:

Izraditi korištenjem standardnih mrežnih tehnologija mrežnu aplikaciju za prodaju koja omogućuje prodaju lokalnih proizvoda. Izrađena aplikacija osim standardnih dijelova mrežne trpel ine posjeduje dio za dodavanje novih prodavača, registriranih korisnika, koji mogu ponuditi vlastite proizvode po kategorijama. Kupci kao registrirani korisnici mogu postavljati narudžbe personaliziranih proizvoda te pratiti njihovu isporuku i realizaciju.

Zadatak uručen pristupniku: 12. svibnja 2022.

Rok za predaju rada: 20. rujna 2022.

Mentor: Predsjednik povjerenstva za završni ispit:

dr. sc. Sanja Brekalo, prof. v. š.

ZAHVALA

Najveću zahvalu prvenstveno dugujem mentorici dr. sc. Sanji Brekalo koja mi je uvijek pomagala davanjem savjeta kod izrade ovog završnog rada, izdvojila dovoljno vremena i strpljenja kod mojih brojnih upita.

Iskreno se zahvaljujem svim profesorima Međimurskog veleučilišta u Čakovcu na prenesenom znanju o upotrebljavanju raznih programa i stjecanju određenog razmišljanja.

Posebnu zahvalu iskazujem cijeloj svojoj obitelji koja me uvijek podržavala i usmjeravala na pravi put.

Na kraju, veliku zahvalu dugujem svojim roditeljima koji su uvijek uz mene, i u sretnim i u teškim trenucima, a bez njih moja postignuća ne bi bila moguća.

Karla Sić

SAŽETAK

Tehnološki napredak, koji iz godine u godinu olakšava pojedine životne potrebe, postao je važan dio trgovine. Trgovina se danas sve više obavlja putem interneta, stoga je i potreba za mrežnim trgovinama sve veća.

Tema ovog završnog rada je planiranje, izrada i oblikovanje prototipa mrežne trgovine na primjeru mrežne trgovine za prodaju lokalnih proizvoda s ciljem digitalizacije i olakšanja prodaje lokalnih proizvoda i povećanja obujma tržišta, ne samo na lokalnoj razini nego i šire. Svrha rada je razvijanje i implementacija mrežne trgovine implementacijom vlastitog CMS (*engl. Content management system*) sustava, odnosno sustava za upravljanje sadržajem.

U radu su objašnjene tehnologije, alati i sustavi korišteni prilikom izrade mrežne aplikacije. Prikazuje se opis tehnologija i programskih jezika kao što su XAMPP, PHPMyAdmin, Visual Studio Code, PHP, HTML, CSS, JavaScript, jQery, Bootsrap i Stripe.

U radu je opisan proces izrade sustava i korištenih tehnologija te prikaz bitnijih mogućnosti sustava, tehnologija, alata i biblioteka. Detaljno je opisana izrada baze. Objašnjeno je administrativno i korisničko sučelje, njihov vizualni aspekt prikazan je slikama. Izdvojene su pojedine slike koda i objašnjena je njegova funkcionalnost. Izrađena mrežna trgovina ima mogućnosti autorizacije korisnika (prijava i registracija), pregleda ponude, olakšani pregled do proizvoda izborom kategorije proizvoda te mogućnost kupnje pomoću košarice (za registrirane korisnike). Sama kupnja omogućena je sustavom Stripe koji je namijenjen za kartične transakcije. Nakon obavljene kupnje kupac ima uvid u pregled kupljenih artikala i status narudžbe. Vlastitom sustavu za upravljanje ima pristup administrator, on dodaje i uređuje kategorije, proizvode i korisnike te ima mogućnost pregleda narudžbi i promjenu statusa dostave.

Mrežna trgovina sastoji se od dijela koji je vidljiv korisnicima i dijela koji im nije vidljiv. Izgled, prezentacijski dio mrežne trgovine postignut je korištenjem CSS-a, HTML-a i JavaScript-a za pojedine funkcionalnosti. Pozadinski se dio razvija pomoću poslužiteljskih (*engl. Server side*) programskih jezika kao što je PHP. Preko njega se

ostvaruje konekcija s bazom MySQL i rad s pohranjenim podacima. Sigurnost od SQL injekcije je postignuta pripremljenim izjavama (*engl. prepared_statement*) kojima se štite podaci koji se upisuju u bazu. Preko API-a ostvarena su komuniciranja servisa s vanjskim servisima, tj. komunikacija košarice i mrežne trgovine. Sustavom Stripe omogućena je kupovina i plaćanje. Obavljenom kupovinom, kupac ima mogućnost pregleda kupnje i statusa dostave.

Ključne riječi: *mrežna trgovina, CMS, JavaScript, PHP, MySQL, API, Stripe.*

SADRŽAJ

1. UVOD 8
2. CILJ 9
3. KORIŠTENE TEHNOLOGIJE, ALATI, SUSTAVI I BIBLIOTEKE 10
   3.1. XAMPP 11
   3.2. PhpMyAdmin 12
   3.3. Visual Studio Code 13
   3.4. PHP 14
   3.5. HTML i CSS 15
   3.6. JavaScript, Bootstrap, AJAX, JQuery i API 16
   3.7. Stripe 17
4. PROCES IZRADE MREŽNE TRGOVINE 18
   4.1. Izrada baze i povezivanje 20
   4.2. Administrativni dio 23
   4.3. Korisnički dio 25
   4.4. Obrada korisničkih podataka i naplata 27
   4.5. Izgled mrežne trgovine 29
5. ZAKLJUČAK 36
6. POPIS LITERATURE 37
7. PRILOZI 38

1. UVOD

Osim ponude informacija, internet ima i mogućnost olakšavanja nekih svakodnevnih obaveza. Sve više je onih koji smatraju da je neophodan za izvor informacija i pomoć kod mnogih obaveza.

Tehnološkim napretkom došlo je i do razvoja društva u pogledu digitalizacije. Danas je internet postao veoma bitan i u području trgovine. Sve veći broj ljudi kupuje preko interneta, stoga dolazi do unapređivanja i nadogradnje sadržaja koji se pružaju na mreži.

Kako bi bio moguć napredak lokalnih proizvođača i njihovih proizvoda na što većem tržištu, sve je veća potreba za digitalizacijom i izradom mrežnih trgovina za prodaju takvih proizvoda. Internet ubrzava i olakšava prodaju, a samim time dostupne su informacije o proizvodima i mogućnost kupnje na širem tržištu od lokalnog.

Ideja ovog rada je napraviti prototip mrežne trgovine za prodaju lokanih proizvoda uz pomoć korištenja sustava, tehnologija, alata i biblioteka za izradu. Njihovim korištenjem postignuta je vizualna i funkcionalna dosljednost mrežne trgovine.

2. CILJ

Cilj završnog rada je stvoriti prototip jednostavne mrežne trgovine, preglednog i jednostavnog sadržaja za lako snalaženje i jednostavnu kupnju s vlastitim sustavom za upravljanje sadržajem, CMS-om (*engl. Content management system*). Razvoj mrežnih stranica, aplikacija ili trgovina odvija se na dvije osnovne razine. Dio programa je vidljiva i funkcionalna prezentacija podataka i u interakciji je s krajnjim korisnikom. Bitno je da je ta prezentacija podataka vizualno dosljedna i jednostavna za korištenje. Pozadinskom dijelu programa, u kojem je sadržana cijela logika, imaju pristup administratori koji rade na njenom ažuriranju, izgradnji baze podataka i manipuliranju podacima.

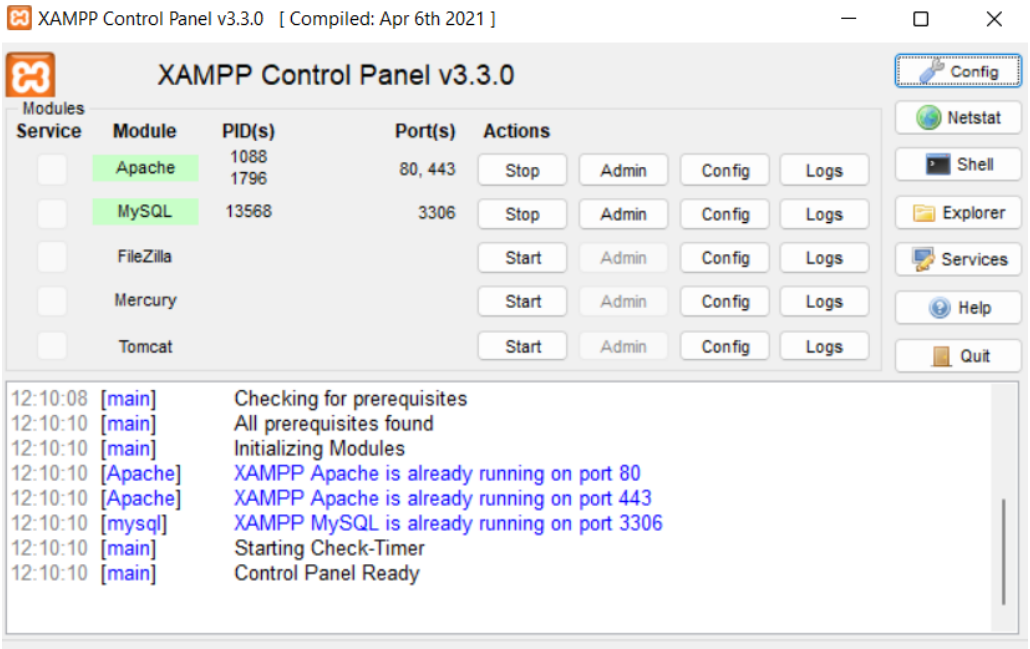
Mrežna trgovina sadrži mogućnosti prijave i registracije korisnika, pregleda ponude, olakšava pregled proizvoda po kategorijama te kupnju pomoću košarice (samo autoriziranim korisnicima preko sustava Stripe). Ostali korisnici mogu pregledavati proizvode, bez mogućnosti kupnje istih. Administrator ima pristup vlastitom sustavu za upravljanje sadržajem, on ima mogućnost dodavanja, uređivanja i brisanja kategorija, proizvoda i korisnika, a narudžbe može pregledavati i mijenjati im status dostave.

3. KORIŠTENE TEHNOLOGIJE, ALATI, SUSTAVI I BIBLIOTEKE

Kako bi bilo moguće napraviti mrežnu trgovinu, upotrebljavaju se mrežne tehnologije, alati, biblioteke i sustavi. U nastavku će biti opisani korišteni programski jezici, alati, sustavi i biblioteke, njihova namjena, mogućnosti i funkcionalnosti te prikaz njihovih sučelja. Korišteni su: XAMPP, PHPMyAdmin, Visual Studio Code, PHP, HTML, CSS, JavaScript, jQuery, Bootrstrap, AJAX i Stripe. Kombinacija korištenog pomaže da mrežna trgovina funkcionira, omogućava komunikaciju s bazom podataka i uređivanjem podataka te naručivanjem i plaćanjem proizvoda.

3.1. XAMPP

XAMPP (*Cross- Platform (X), Apache (A), Maria DB (M), PHP (P), Perl (P)*) je razvojno okruženje koje integrira PHP. U osnovi je Apache, besplatna distribucija koja služi kao internet poslužitelj, a unutar koje su dodani alati poput Maria DB, PHP i Perl. Otvorenog je koda koji je razvila tvrtka Apache Friends, funkcionira na više različitih platformi (*engl. Cross-platform*), a sastoji se od Apache HTTP poslužitelja, Maria DB baze podataka i tumača za skripte napisane u PHP i Perl programskim jezicima. Koristi se za testiranje i razvoj mrežnih trgovina, stranica i aplikacija prije nego se stranice postave (*engl. Upload*) na udaljeni mrežni poslužitelj. Instalacija je jednostavna, kao i korištenje.[1] Slika 1. prikazuje jednostavno sučelje koje olakšava upravljanje razvojnim okruženjem.

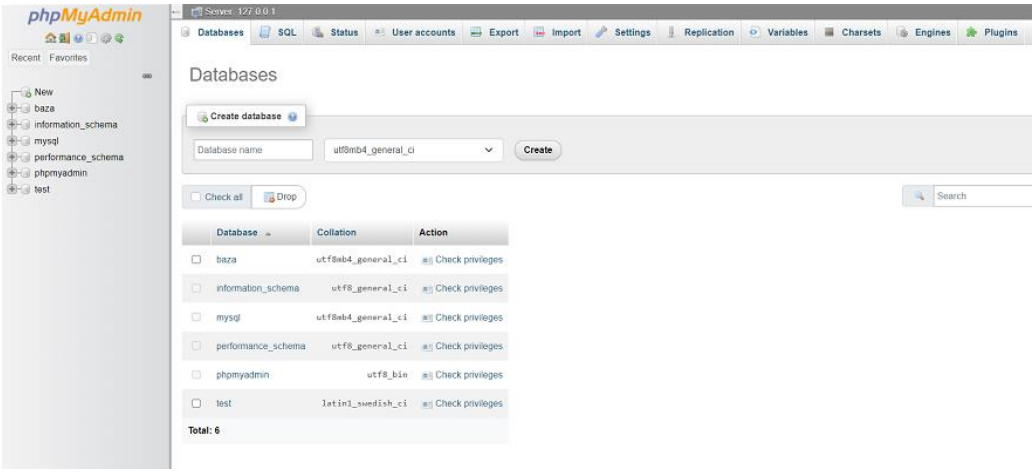


Slika 1. Programsko sučelje XAMPP alata

IZVOR: autor

3.2. PhpMyAdmin

PhpMyAdmin besplatan je alat pisan u PHP-u, a služi za upravljanje i administraciju My SQL baze podataka. Na lokalnom poslužitelju pristupa mu se upisivanjem adrese http://localhost/phpmyadmin. Uz pomoć njega mogu se izvršavati SQL operacije putem korisničkog sučelja izravno u pregledniku. PhpMyAdmin je GUI, grafičko korisničko sučelje (*engl. Graphical user interface*). Koristi se za upravljanje podacima unutar baze podataka, poljima, tablicama, indeksima, korisnicima, dozvolama pristupa, izvršavaju se naredbe na bazu i dr. PhpMyAdmin sučelje mrežne stranice prikazano je na slici 2. [2]

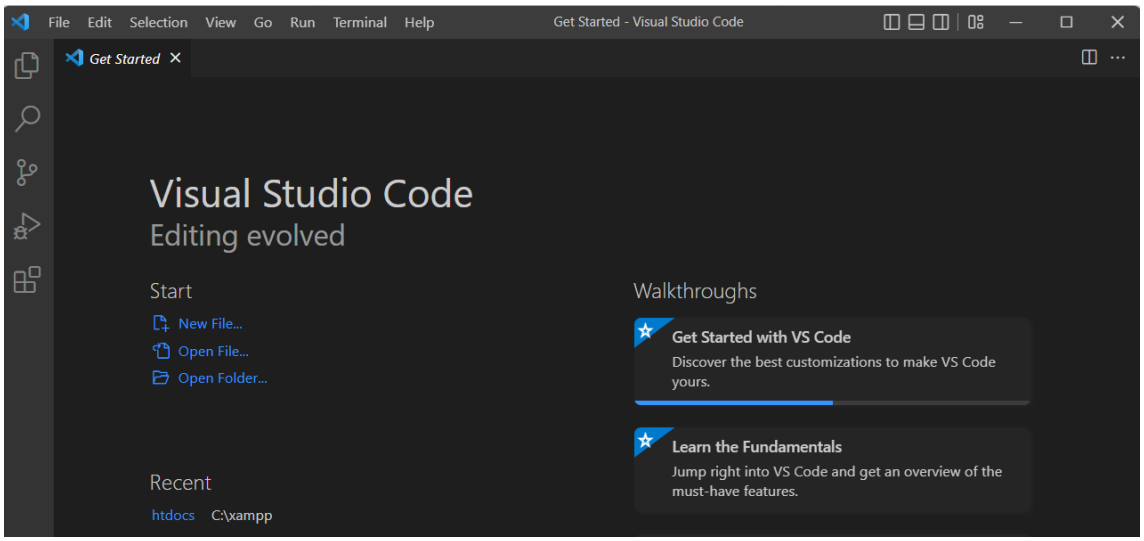


Slika 2. PHPMyAdmin sučelje

IZVOR: autor

3.3. Visual Studio Code

Visual Studio Code, je uređivač (*engl. Editor*) izvornog koda. Besplatan je alat koji može olakšati izradu aplikacije.[3]Pomaže kod pisanja koda, uklanjanja grešaka i otkriva ako je dio koda ostavljen nepotpun. Obično podržava sve programske jezike. Ako se želi koristiti programski jezik koji nije podržan, može se preuzeti proširenje. Za korištenje je dovoljno skinuti dodatak sa službene stranice, instalirati ga i nakon instalacije pokrenuti te početi s radom.[3] Na slici 3. je prikazano jednostavno sučelje programa VS Code.



Slika 3. Sučelje Visual Studio Code

IZVOR: autor

3.4. PHP

PHP (*engl. Hypertext Preprocessor*) je često korišteni jezik skriptiranja opće namjene otvorenog koda koji je pogodan za razvoj mrežnih aplikacija i stvaranje dinamičkih mrežnih stranica. PHP: *Hypertext Preprocessor* je tzv. rekurzivni akronim kod kojeg se sama skraćenica pojavljuje u značenju. Većina PHP-ove sintakse slična je drugim jezicima kao što su C++, Java i Perl. Popularan je izbor za stvaranje dinamičnih mrežnih stranica zbog svoje integracije sa SQL-om. Jednostavnog je održavanja i korištenja.[2]

Omogućuje razvojnom programeru da razvije programe koristeći tehnike proceduralnog i objektno orijentiranog programiranja. Za PHP datoteke nužno je da imaju ekstenziju .php kako bi se mogle proslijediti PHP procesoru prilikom izvršavanja. Unutar PHP datoteke moguće je postavljati samo PHP kod ili on može biti postavljen u kombinaciji s HTML, CSS i/ili JavaScript kodom. Bitno je da se PHP obilježi oznakom kako bi se odvojio od ostatka sadržaja u dokumentu. [2]

PHP skripte mogu biti od jednostavnih jednorednih naredbi do složenih programa. PHP kod se obrađuje prije učitavanja HTML-a, stoga korisnici ne mogu vidjeti PHP kod unutar mrežne stranice. Oni mogu vidjeti samo HTML izlaz nakon izvršenja PHP skripte.

3.5. HTML i CSS

HTML i CSS su dva odvojena jezika, ali budući da ovise jedan o drugom, često ih se uparuje zajedno. HTML opisuje strukturu stranice, dok CSS opisuje kako će stranica izgledati. Slika 4. prikazuje dio CSS koda. HTML (skraćeno od *engl. Hypertext Markup Language*) je skriptni jezik koji semantički opisuje sadržaj dokumenata. CSS (skraćeno od *engl. Cascading Style Sheets*) je jezik koji opisuje stilove dokumenata, tj. njezin vizualni izgled. HTML5 i CSS3 su posljednja izdanja navedenih jezika koja su uvela velike novosti u izradu mrežnog sadržaja.[4]

Slika 4. Izgled CSS koda

IZVOR: autor

3.6. JavaScript, Bootstrap, AJAX, JQuery i API

JavaScript je postao jedan od glavnih jezika skriptiranja pri izradi mrežnih stranica i aplikacija. Postoje mnogi razlozi za to, uključujući njegov rad s objektima i mogućnost brze izrade aplikacija. Kako bi se ispravili nedostaci ugrađeni u sam jezik, programeri su kreirali mnoge biblioteke koje pokušavaju riješiti različite probleme.[4] Neke od najčešće korištenih biblioteka su Vanilla JavaScript i jQuery. JQuery je najpoznatija biblioteka, čije su mogućnosti danas implementirane u novije verzije JavaScripta.[4] Međutim, sve više se koriste programski okviri i biblioteke kao što su React, Vue i Angular (od kojih svaka ima svoj način pisanja i strukturu koja odgovara određenoj veličini i vrsti projekta te se koriste u izradi dinamičkih mrežnih sjedišta).

Bootstrap je najpopularniji CSS programski okvir za razvoj responzivnih mrežnih stranica koje su usmjerene na mobilne uređaje. Bootstrap 5 je najnovija verzija Bootstrap-a. Bootstrap 5 podržava glavne preglednike osim Internet Explorera 11 i starijih. Kao programski okvir, Bootstrap uključuje osnove za responzivan dizajn. Programeri trebaju samo umetnuti kod u predefinirani sistem mreže (*engl. grid system*). Bootstrap je građen na temelju HTML-a, CSS-a i JavaScripta. Mrežni programeri, koji koriste Bootsrap, mogu brže graditi mrežne stranice bez potrebe programiranja nekih osnovnih naredbi i funkcionalnosti. [7]

AJAX je sinkroni JavaScriptu i XML-u. AJAX nije programski jezik. On samo koristi kombinaciju XMLHttpRequest objekta JavaScript i HTML DOM za prikaz ili korištenje podataka. AJAX aplikacije mogu koristiti XML za prijenos podataka. Jednako je uobičajen prijenos podataka kao običan tekst ili JSON tekst. AJAX omogućuje asinkrono ažuriranje mrežnih stranica razmjenom podataka s mrežnim poslužiteljem preko pozadinske komunikacije. To omogućuje ažuriranje dijelova mrežne stranice bez ponovnog učitavanja cijele stranice.

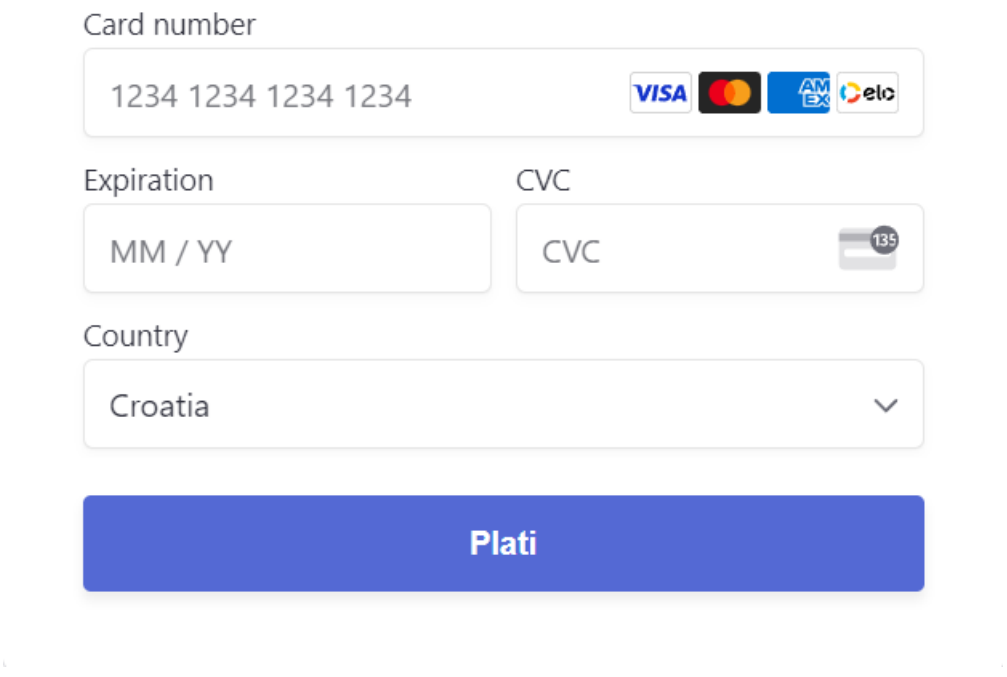
Svrha jQueryja je olakšati korištenje JavaScripta na mrežnoj stranici. JQuery uzima uobičajene zadatke koji zahtijevaju mnogo redaka JavaScript koda i omata ih u metode koje se mogu pozvati jednim redom koda. Biblioteka jQuery sadrži sljedeće značajke: HTML/DOM manipulacija, CSS manipulacija, HTML metode događaja i efekti i animacije, AJAX i dr. JQuery je podržan u svim preglednicima. Glavna zadaća JQueryja

je da pomoću gotovih funkcija olakšava korištenje JavaScripta kod pozivanja API-a, to jest PHP skripti koje asinkrono izvršavaju zadaće pomoću AJAX-a. API je standardizirani način komuniciranja servisa s vanjskim servisima.[8]

3.7. Stripe

Stripe je platforma za mrežno plaćanje i obradu podataka s bankovnh kartica za tvrtke, tj. mrežne trgovine. Stripe omogućuje sigurnu i učinkovitu obradu sredstava putem kreditne kartice ili banke i prenosi ta sredstva na račun prodavača. Slika 5. prikazuje izgled Stripe kartice za plaćanje koju se popunjava prilikom transakcija.

Sustav Stripe uključuje platformu za obradu plaćanja, kao i pristupnik za plaćanje karticom, a oba su potrebna u svakoj uspješnoj mrežnoj transakciji, što ga čini učinkovitim i jednostavnim sustavom za mrežno plaćanje. [9]



Slika 5. Izgled Stripe kartice za plaćanje

IZVOR: autor

4. PROCES IZRADE MREŽNE TRGOVINE

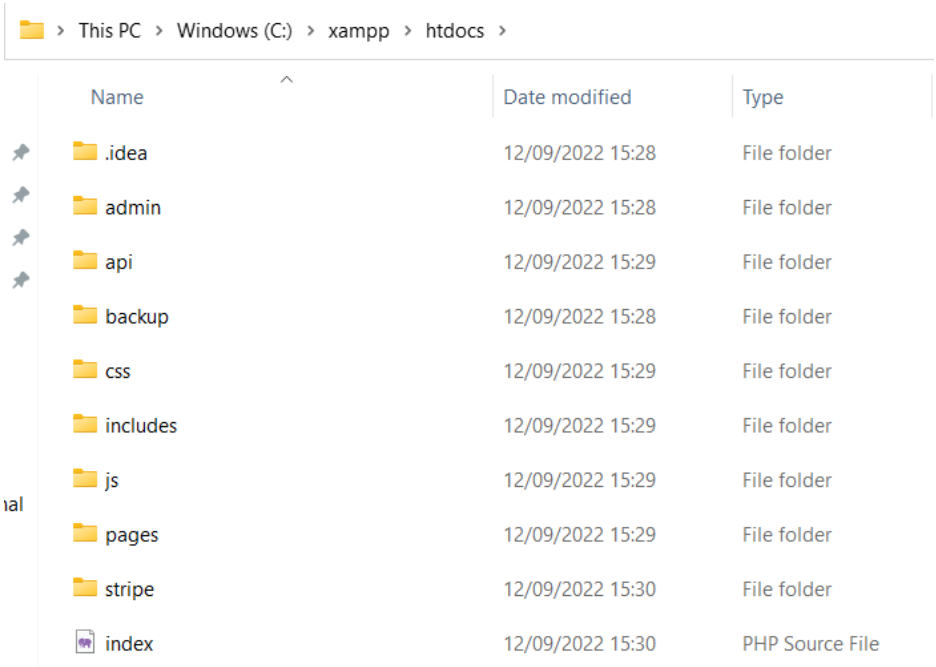
Razvoj mrežnih stranica, aplikacija ili trgovina odvija se na dvije osnovne razine: prednja strana ili prezentacijski dio razvoja i razvoj pozadinskog, tj. funkcionalnog dijela.

Glavne tehnologije za izradu prezentacije podataka su HTML, CSS i JavaScript. CSS se koristi za pozicioniranje elemenata, margina, veličine fonta, boje i drugih vizualnih karakteristika mrežne stranice. Preko HTML-a izrađena je struktura mrežne stranice. Za sve interaktivne elemente kao što su klizači, padajući izbornici i razni efekti koristi se JavaScript. On je jezik na klijentskoj strani, što znači da se izvršava u pregledniku. Postoje još i dodatne razne biblioteke (*engl. Library*) i programski okviri (*engl. Framework*). Oni čine kodiranje lakšim i bržim. Najčešće se koriste jQuery, LESS i Bootstrap. JQuery je JavaScript biblioteka, LESS je CSS biblioteka i Bootstrap je programski okvir prezentacijske strane razvoja.

Pozadinski dio programa mrežne stranice možemo nazvati funkcionalnim te je to dio kojemu imaju pristup administratori i vlasnici koji rade na njegovom ažuriranju. Operacije pozadinskog razvoja su kompleksnije od operacija prezentacije sadržaja. Pozadinski se dio izvršava na serveru te se izrađuje pomoću poslužiteljskih (*engl. Server side*) programskih jezika kao što je PHP. Preko PHP-a ostvaruje se konekcija s bazom MySQL podataka, koja omogućuje pohranu podataka sustava.

Prilikom izrade mrežne aplikacije, osim samog programiranja, potrebno je postaviti i razvojno okruženje. Prilikom izrade mrežne trgovine korišten je XAMPP server preko kojeg se putem PHPMyAdmin-a izrađuje baza podataka. XAMPP je lokalni poslužitelj mrežnih stranica. Instalacija XAMPP-a omogućuje i PGP parser koji PHP programski jezik prevodi u HTML prezentaciju krajnjem korisniku. Pri pisanju koda korišten je Visual Studio Code.

Kako bi mrežna stranica bila dostupna u pregledniku, potrebno je sve datoteke mrežnog sjedišta prebaciti u *htdocs* direktorij koji je zapravo glavno polazište mrežnog poslužitelja. Slika 6. prikazuje direktorij u kojemu se nalazi struktura mrežne stranice.



Slika 6. Htdocs direktorij

IZVOR: autor

Izgled stranice izgrađen je CSS-om i HTML-om, korištenjem Bootstrap-a. Bootstrap je biblioteka predefiniranih klasa koje možemo koristiti bez ponovnog pisanja CSS stilova. Brža je i lakša za upotrebu te koristi standarde koji se primjenjuju u suvremenoj izradi mrežnih stranica. Bootstrap omogućuje responzivne stranice, stranice koje su prilagođene za tablete, stolna računala te mobilne uređaje.

4.1. Izrada baze i povezivanje

Prije izrade aplikacije potrebno je isplanirati bazu podataka, odrediti relacije u bazi te odrediti tipove podataka. Baza izrađene mrežne trgovine podijeljena je na sljedeće stavke: kategorije, korisnik, narudžba i proizvod, kao što je prikazano na slici 7.

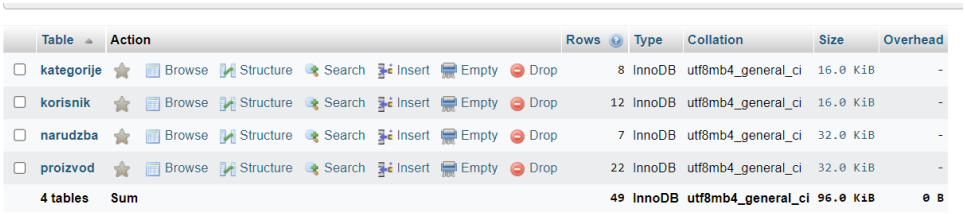


Table	Action	Rows	Type	Collation	Size	Overhead
kategorije	Browse Structure Search Insert Empty Drop	8	InnoDB	utf8mb4_general_ci	16.0 KiB	-
korisnik	Browse Structure Search Insert Empty Drop	12	InnoDB	utf8mb4_general_ci	16.0 KiB	-
narudzba	Browse Structure Search Insert Empty Drop	7	InnoDB	utf8mb4_general_ci	32.0 KiB	-
proizvod	Browse Structure Search Insert Empty Drop	22	InnoDB	utf8mb4_general_ci	32.0 KiB	-
4 tables	Sum	49	InnoDB	utf8mb4_general_ci	96.0 KiB	0 B

Slika 7. Prikaz osnovnih segmenata baze

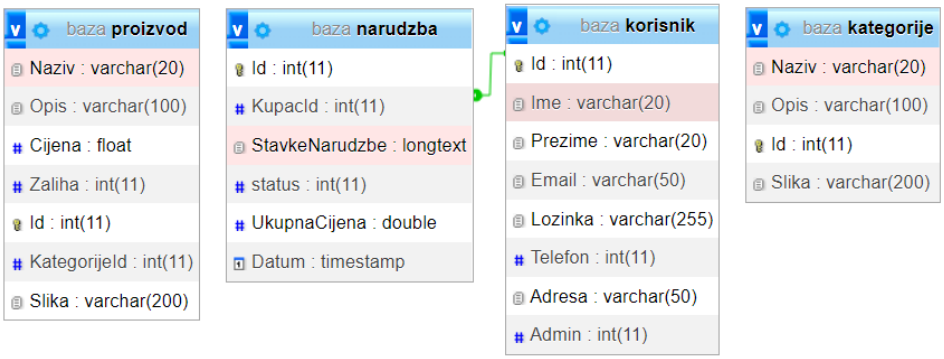
IZVOR: autor

Kategorije se sastoje od polja id-a, naziva, opisa i slike. Id je primarni ključ. Bitno je za njega da je uvijek tipa INTEGER. On je numerički cijeli broj. Potrebno je označiti polje AUTO_INCREMENT, što znači da se svaki sljedeći zapis automatski povećava za prethodno zadani (*engl. Default*) broj 1. Naziv, opis i slika su polja tipa VARCHAR, što je kratica za varijabilni znak. VARCHAR je vrsta podataka koji se koriste za spremanje znakovnih nizova promjenjive dužine. Korisnik se sastoji od id-a, imena, prezimena, email-a, lozinke, telefona, adrese i ADMIN polja koje označava je li korisnik administrator ili kupac.

Narudžba se sastoji od id-a, id-a kupca, stavaka narudžbi, statusa kupovine, ukupne cijene i datuma. Id i id kupca su polja INTEGER, gdje polje id kupca predstavlja relaciju(*engl. Foreign key*) koja se povezuje s tablicom korisnika iz koje se podaci kasnije koriste za prikaz informacija o narudžbi. Stavke narudžbe su tipa LONGTEXT. Upotrebljava se zbog toga jer se u njega pohranjuje JSON struktura podataka u kojoj se nalaze stavke narudžbe. Ovim načinom može se pohraniti više stavaka narudžbi i informacija o njima koje se nalaze u nizu (*engl. Array*). Status kupovine koji je početnog statusa 0, što znači da je plaćeno jer na stranici je omogućena samo kupnja sustavom Stripe koji se koristi samo za kartična plaćanja. Ukupna cijena DOUBLE se koristi za brojeve s pomičnim zarezom (izračun ukupne cijene izvršava se prolaskom kroz cijene niza stavaka narudžbe

spremljenih u JSON formatu). Datum tipa TIMESTAMP pohranjuje vrijednost godine, mjeseca, dana, sata, minute.

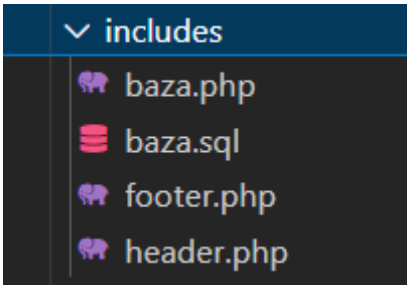
Proizvod se sastoji od id-a, naziva, opisa, cijene, zaliha, id-a kategorije i slike. Id, zaliha i id kategorije su tipa INTEGER. Poveznicom id kategorije na tablicu kategorije dobivaju se informacije o kategoriji za kasniji prikaz na stranici. Polje cijena je tipa FLOAT za pohranu i prikaz decimalnog broja. Naziv i opis proizvoda su tipa VARCHAR. Da bi bio moguć prikaz slike, dovoljno je spremanje njene putanje kao podatak. Ovakav način optimalniji je jer uzima manje mjesta za pohranu od primjerice *Base64* formata slike. Na slici 8. prikazana je struktura cijele baze.



Slika 8. Struktura baze

IZVOR: autor

Nakon što je baza izgrađena, bitno je njeno povezivanje. U direktorij *includes* (slika 9.), koji se nalazi unutar direktorija *admin*, bitno je spremiti datoteku baza.php koja se koristi u svakoj datoteci. Korištenjem *includes* direktorija unutar mrežne trgovine olakšano je pisanje te se izbjegava ponavljanje istog koda. Primjer je korištenje spajanja na bazu, kod kojeg je pri svakom spajanju isti kôd i zaglavlje te podnožje stranice.



Slika 9. Includes direktorij, prikazan u VS Codu

IZVOR: autor

Nakon što je baza povezana, mogu se izvršavati zahtjevi prema bazi podataka, kao što su: uzimanje (*engl. Select*), dodavanje (*engl. Insert*), brisanje (*engl. Delete*) i ažuriranje(*engl. Update*) podataka. Navedene akcije zajedničkim imenom se nazivaju "CRUD" - *Create, Read, Update, Delete.* Zaštitom od MySQL napada, koristeći se ubacivanjem pripremljene vrijednosti (*engl. prepared_statement*) koje se provjeravaju prije izvršavanja SQL naredbe, postiže se osiguranje baze od neželjenih akcija.

4.2. Administrativni dio

Administrativni dio mrežne trgovine, odnosno "CMS", odvojen je pomoću zasebne sesije koja se definira kod prijave na administrativne stranice, dok se kod prijave na korisničke stranice mrežne trgovine koristi sesija drugog naziva. Direktorij, gdje administrator može upravljati narudžbama, proizvodima, kategorijama i kupcima, nalazi se izvan direktorija mrežne trgovine, tako da kupac nikad neće moći pristupiti administraciji bez autorizacije. U administrativnom dijelu, prilikom autorizacije, provjeravamo je li korisnik *admin* i ima li ovlasti mijenjati podatke.

Kod mrežnih trgovina administracija je veoma bitna. Ovaj dio upravlja cjelokupnom strukturom stranice, upravlja proizvodima i njihovim cijenama, ima mogućnost praćenja narudžbi te njihovo obrađivanje i isporuku.

Admin pages direktorij logika sadrži:

1. Pocetna.php - početna strana administratora.
2. Prijava.php - prijavljivanje administratora. Prikaz prijave, provjere podataka i zaštita. slika 10.

```php
<?php
if (isset($_POST['email']) && isset($_POST['lozinka'])) {
    $email = $_POST['email'];
    $lozinka = $_POST['lozinka'];

    // priprema varijabli (zastita od mysql injectiona, osiguramo da nam string nema navodnike)
    $email = mysqli_real_escape_string($db, $email);
    $lozinka = mysqli_real_escape_string($db, $lozinka);
    $lozinka = md5($lozinka);
    $provjeraPrijave = mysqli_query($db,"select * FROM korisnik WHERE Email = '{$email}' AND Lozinka = '{$lozinka}' AND Admin = 1");

    if ($provjeraPrijave) {

        $korisnik = mysqli_fetch_assoc($provjeraPrijave);
        $_SESSION['korisnik'] = $korisnik['Id'];
        Header("Location: /admin/index.php?page=pocetna");
    }
}
```

Slika 10. Prikaz koda za prijavu administratora

IZVOR: autor

3. Odjava.php - odjava administratora.
4. Kategorija.php - prikazuje listu kategorija.
5. Kategorija_novi.php - dodavanje nove kategorije.

6. Kategorija_uredi.php - uređivanje kategorije.
7. Kategorija_obrisi.php - brisanje kategorije.
8. Korisnik.php - prikazuje listu korisnika.
9. Korisnik_novi.php - dodavanje novog korisnika. Spremanje podataka u bazu i zaštita pomoću *prepare statemanta*. (Slika 11)

```php
$sql = "INSERT INTO korisnik (Ime, Prezime, Email, Lozinka, Telefon, Adresa, Admin) VALUES (?,?,?,?,?,?,?)";
$stmt = mysqli_prepare($db, $sql);
if($stmt && !$greska){

    // Set parameters
    $ime = $_POST['Ime'];
    $prezime = $_POST['Prezime'];
    $email=$_POST['Email'];
    $lozinka = $_POST['Lozinka'];
    $telefon = $_POST['Telefon'];
    $adresa = $_POST['Adresa'];
    $admin=$_POST['Admin'];
    // Bind variables to the prepared statement as parameters
    mysqli_stmt_bind_param($stmt, 'ssssisi', $ime, $prezime, $email, $lozinka, $telefon, $adresa, $admin);


    // Attempt to execute the prepared statement
    if(mysqli_stmt_execute($stmt)){
        $poruka= "Uspješno dodani korisnik.";
    } else{
        $greska= mysqli_error($db);
    }
```

Slika 11. Dio koda za dodavanje novog korisnika i zaštita podataka

IZVOR: autor

10. Korisnik_uredi.php - uređivanje postojećeg korisnika.
11. Korisnik_obrisi.php - brisanje korisnika.
12. Proizvod.php - prikazuje listu proizvoda.
13. Proizvod.novi.php - dodavanje novog proizvoda.
14. Proizvod_uredi.php - uređivanje proizvoda.
15. Proizvod_obrisi.php - brisanje proizvoda.
16. Narudzba.php – prikazuje listu narudžbi.
17. Narudzba_uredi.php - pregled postojećih narudžbi.
18. Narudzba_obrisi.php - brisanje narudžbi. Dio koda prikazan na slici 12.

```php
<?php
$id='';
if($_GET['id']){
    $id=$_GET['id'];
}
//uzima iz baze preko kategorije id
mysqli_query($db,"DELETE FROM narudzba WHERE Id={$id}");

?>
<script>
    location.href="index.php?page=narudzba";
</script>
```

Slika 12. Brisanje narudžbe

IZVOR: autor

4.3. Korisnički dio

Pages direktorij služi nam za odvajanje logike na svakoj stranici. Njegove sastavnice su:

1. Kosarica.php je prikaz proizvoda unutar košarice. Nakon dodavanja proizvoda u košaricu, korisnik ima mogućnost pregleda sadržaja košarice te njihovo dodavanje ili brisanje.
2. Odjava.php - odjavljivane korisnika.
3. Placanje.php - stranica za plaćanje. Korištenje STRIPE-a; nakon dohvata iznosa iz podataka o narudžbi pokreće se transakcijski dio i prikaz forme za unos podataka o kartici za plaćanje.
4. Pocetna.php - početna strana trgovine.
5. Prijava.php - prijavljivanje korisnika. Unosom točne email adrese i lozinke pokreće se sesija kojom se kasnije može izvršiti kupovina.
6. Proizvod.php - stranica odabranog proizvoda. Dohvaćanje podataka spremljenih u bazu podataka.
7. Proizvodi.php- prikaz liste proizvoda koristeći parametar „kategorija“ kao definirani filter proizvoda za olakšano snalaženje.

8. Registracija.php - registriranje korisnika unosom podataka i kreiranje korisničkog računa.
9. Zavsetak_narudzbe.php - nakon izvršenja transakcije, ovdje je sažetak narudžbe koji će biti promjenjiv ovisno o statusu transakcije ili statusu koji administrator promjeni.

PHP Sesije se koriste za provjeru prijave korisnika za kreiranje narudžbe korisnika. To su globalne varijable na serverskoj strani koje se spremaju prilikom svakog spajanja korisnika te su isto tako jedinstvene na razini svakog preglednika. Slika 13. prikazuje dio koda za provjeru korisnika. U nju se mogu spremati podaci. Ako je sesija nastavljena, može se prenositi na sve stranice ove mrežne trgovine.

```
if ($provjeraPrijave) {

    $korisnik = mysqli_fetch_assoc($provjeraPrijave);
    $_SESSION['korisnik'] = $korisnik['Id'];
    Header("Location: /admin/index.php?page=pocetna");
}
```

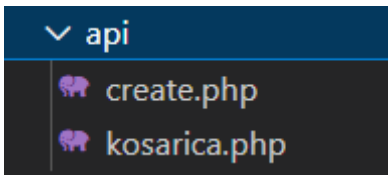
Slika 13. Provjera korisnika dio koda

IZVOR: autor

Funkcionalnost košarice postignuta je korištenjem AJAX-a kao poveznice između JavaScripta i pozadinskih servisa te se tako dobije trenutna promjena i izračun košarice kako bi krajnji korisnik bez ponovnog učitavanja stranice mogao vidjeti promjene. AJAX radi na asinkroni princip obrađivanja podataka koji ne ometa prikaz mrežne stranice. Dio je JavaScripta, što znači da je na strani klijenta, pa zbog toga nije potrebno ponovno učitavanje poslužiteljske strane, odnosno PHP-a.

4.4. Obrada korisničkih podataka i naplata

API je standardizirani skup pravila koja se primjenjuju kod međusobnog povezivanja s drugim vanjskim resursima. Koristi se kod komuniciranja košarice s mrežnom trgovinom. Prikaz direktorija na slici 14.



Slika 14. API, prikaz u VS Codu

IZVOR: autor

Dijelovi API-a:

1. *create.php* je dio za kreiranje transakcije za STRIPE. Za izvršavanje je potrebno poslati podatke o narudžbi prema STRIPE-u i dalje koristeći biblioteke za STRIPE nastavlja se obrada s njihove strane. Slika 15. prikazuje dio koda koji kreira podatke za transakciju.

```php
$jsonStr = file_get_contents('php://input');
$jsonObj = json_decode($jsonStr, true);

// kreiranje plaćanja s iznosom i valutom
$paymentIntent = \Stripe\PaymentIntent::create([
    'amount' => $jsonObj['iznos'],
    'currency' => 'gbp',
    'payment_method' => 'pm_card_visa',
]);
```

Slika 15. Kreiranje podataka za slanje prema Stripe-u

IZVOR: autor

2. kosarica.php - izračun proizvoda unutar košarice i ukupne cijene kod promjena sadržaja košarice.

Povezivanje s API-em implementiran je u *scripts.js*, JavaScript datoteci koja putem AJAX-a manipulira stavkama košarice. Elementi količina, cijena proizvoda i ukupna cijena ažuriraju se nakon dobivene povratne informacije implementiranog API servisa.

Kao način plaćanja, mrežna trgovina koristi STRIPE, koji je jedan od poznatijih kartičnih načina plaćanja. Jednostavna je integracija unutar samog koda. Nakon što kupac unese podatke za plaćanje, STRIPE ih obradi i povratno se vraća odgovor o statusu transakcije. Prema njemu se ažurira status narudžbe koju je kupac kreirao. Korišten je test Stripe za koji postoje testni brojevi koji se upisuju za broj kartice (ovisno o vrsti kartice). Slika 5. prikazuje izgled kartice za plaćanje, a slika 16. prikazuje testne brojeve kartice koji se koriste.

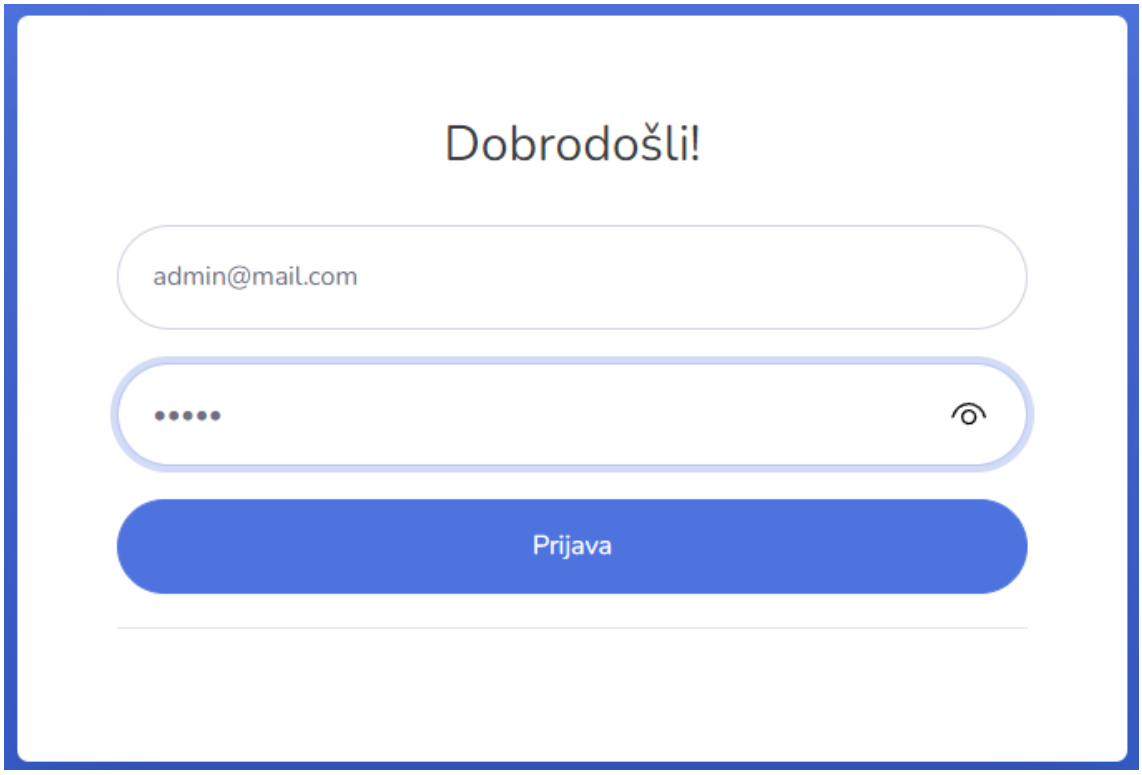
Visa	4242 4242 4242 4242	Any 3 digits	Any future date
Visa (debit)	4000 0566 5566 5556	Any 3 digits	Any future date
Mastercard	5555 5555 5555 4444	Any 3 digits	Any future date
Mastercard (2-series)	2223 0031 2200 3222	Any 3 digits	Any future date
Mastercard (debit)	5200 8282 8282 8210	Any 3 digits	Any future date
Mastercard (prepaid)	5105 1051 0510 5100	Any 3 digits	Any future date
American Express	3782 822463 10005	Any 4 digits	Any future date
American Express	3714 496353 98431	Any 4 digits	Any future date
Discover	6011 1111 1111 1117	Any 3 digits	Any future date

Slika 16. Prikaz testnih brojeva za test Stripe

IZVOR: 8. test Stripe [8]

4.5. Izgled mrežne trgovine

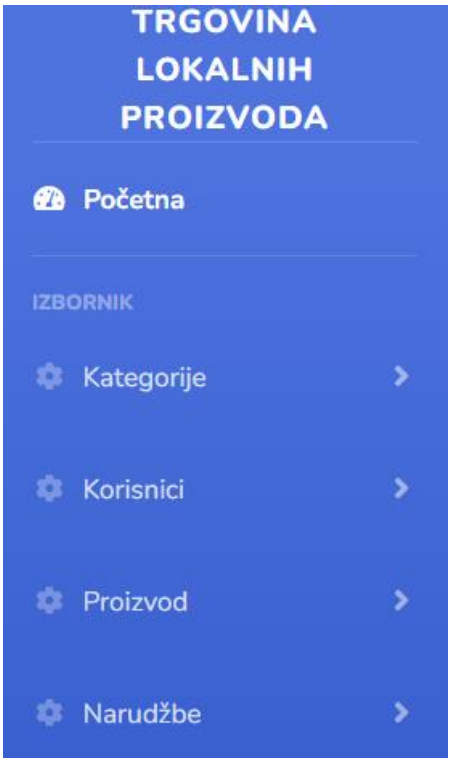
Mrežna aplikacija je sastavljena od sustava za vlastito upravljanje (administrativni dio) i trgovine (korisnički dio). Sustav za upravljanje je dio kojeg nazivamo administrativnim. Pristup administrativnom dijelu ima samo administrator, on ima ovlasti dodavanja, uređivanja, pregleda i brisanja podataka. Stranica za prijavu administratora je prikazana na slici 17., ovaj dio nikako nije vidljiv korisnicima koji nemaju ovlasti administratora.



Slika 17. Stranica za prijavu administratora

IZVOR: autor

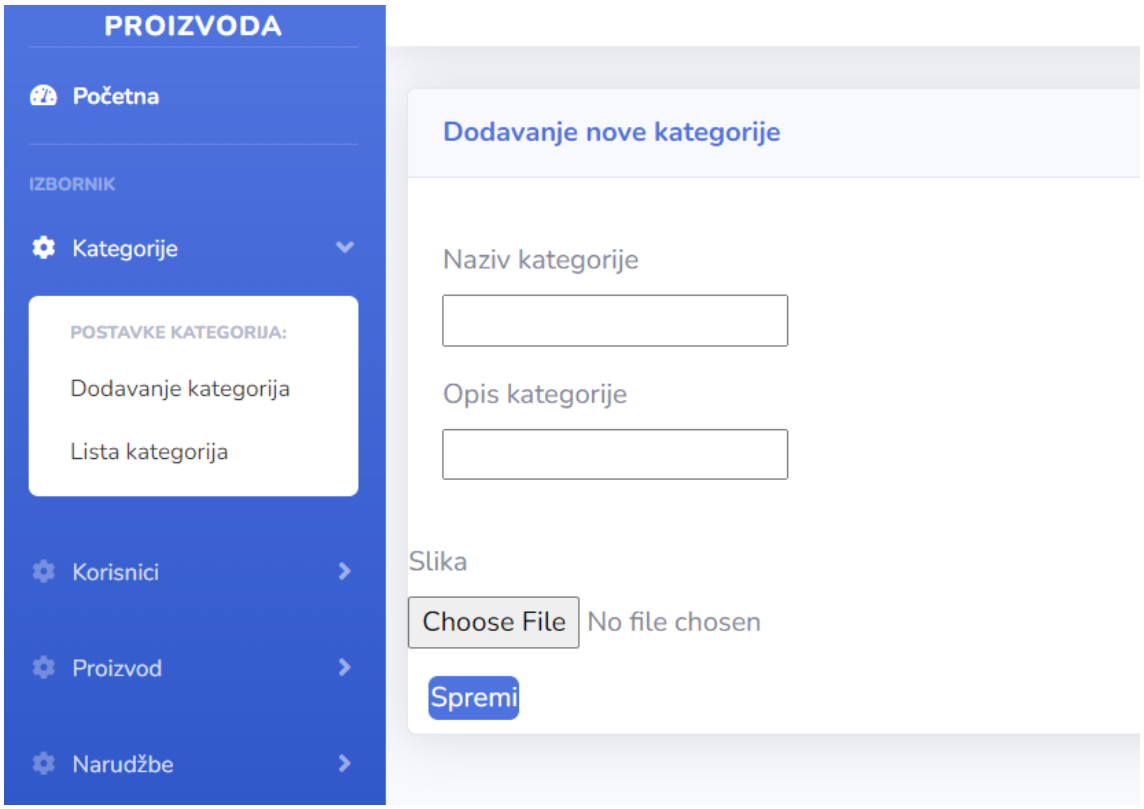
Nakon prijave u administraciju otvara se, kao što je prikazano na slici 18, izbornik koji sadrži kategorije, korisnike, proizvode i narudžbe.



Slika 18. Izbornici koji sadrže liste i prikaz kategorija, korisnika, proizvoda i narudžbi

IZVOR: autor

Pritiskom na kategorije postoji mogućnost dodavanja nove kategorije (prikaz na slici 19.) ili prikaza liste već postojećih kategorija koje su spremljene u bazu. Kod dodavanja kategorije potrebno je ispuniti polja: naziv, opis i slika. Bitno je ispuniti sve podatke prije spremanja jer ako nije sve ispunjeno, ne može se izvršiti spremanje, budući da su sva polja obvezna. Uz dio za dodavanje kategorije, postoji i dio lista kategorija koja prikazuje kategorije koje već postoje u bazi. Uz prikaz postoji i mogućnost uređivanja i brisanja kategorije.



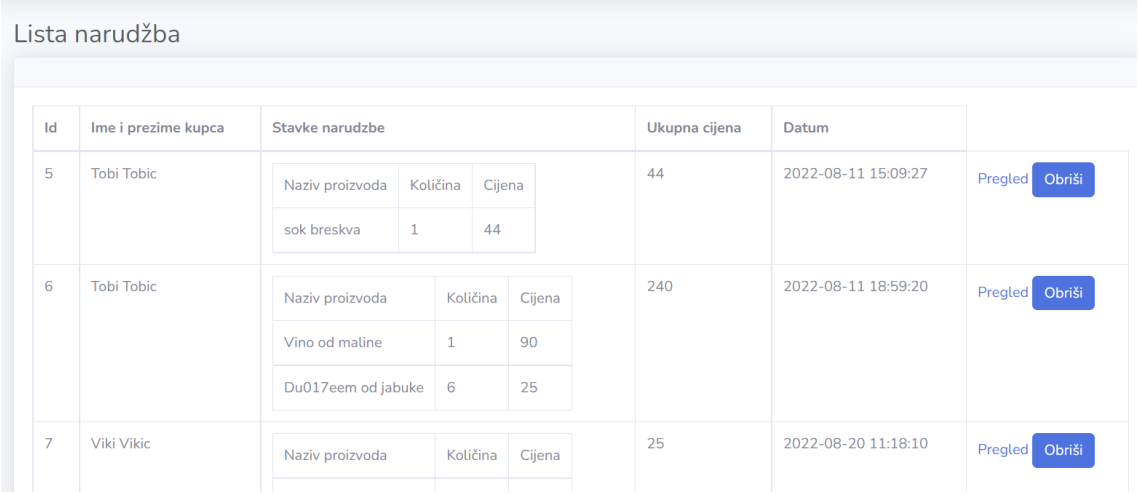
Slika 19. Stranica za dodavanje nove kategorije

IZVOR: autor

Na isti način funkcionira i korisnik. Dodavanje novog korisnika sadrži polja: ime, prezime, email, lozinka, telefon, adresa i polje za odabir je li korisnik koji se dodaje administrator ili nije. Ako su sva polja ispunjena, pritiskom se spremaju podaci. Novi korisnik se pohranjuje u listu korisnika. Lista korisnika prikazuje sve spremljene korisnike, njihovo ime, prezime, email, telefon, adresu te oznaku ako je korisnik administrator, u tom je slučaju označen kao 1, a ako nije administrator, onda je 0. Isto tako postoji mogućnost uređivanja korisnika i njegovo brisanje.

Također proizvod ima dio za dodavanje novih proizvoda i dio za prikaz onih koji su već pohranjeni - nazvan lista proizvoda. Za izradu novog proizvoda potrebno je ispuniti polja: naziv, opis, cijena, zaliha i odabrati kategoriju kojoj pripada te sliku proizvoda koji se dodaje. Sva polja su obvezna, tako da pohrana nije moguća ako sva polja nisu ispunjena. Lista proizvoda prikazuje proizvode, a ako je potrebno moguće je uređivanje i brisanje proizvoda.

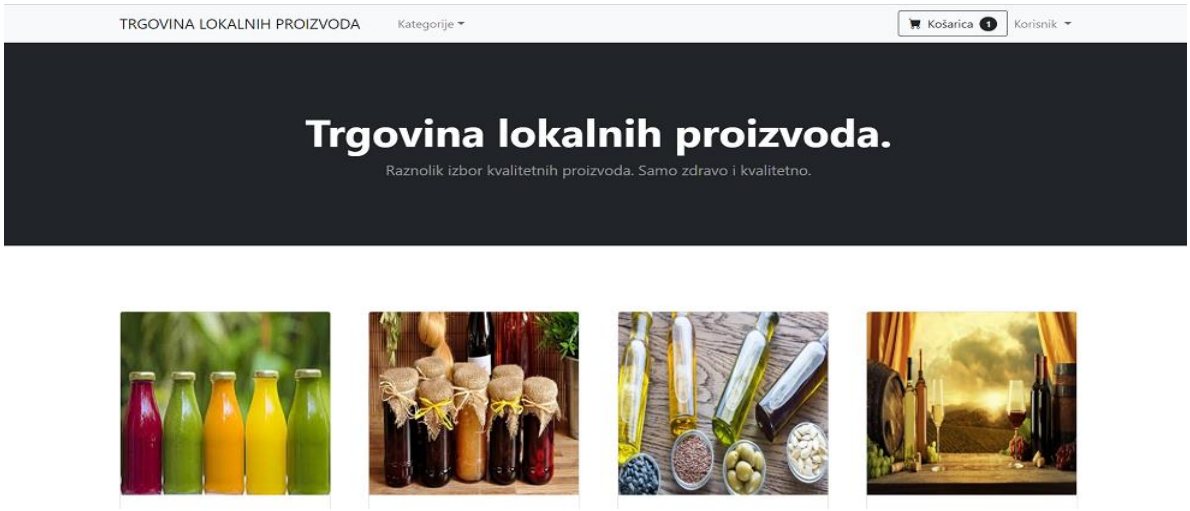
Narudžba sadrži samo dio za prikaz liste narudžbi u kojoj je vidljivo ime i prezime kupca, stavke koje su naručene, ukupna cijena naručenih stavki te datum i vrijeme kad je narudžba napravljena. Moguće je pregledati narudžbe, kao što je prikazano na slici 20., i ako je neophodno, brisati ih.



Slika 20. Pregled narudžbi

IZVOR: autor

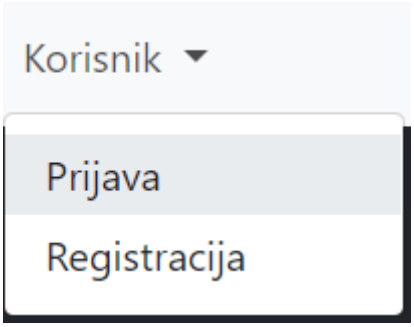
Sama trgovina ili korisnički dio napravljen je jednostavno i pregledno te je podijeljen na korisnika, koji se autorizira, i korisnika, koji samo pregledava mrežnu trgovinu, tj. posjetitelja. Korisnik, koji se autorizira, prolazi dio za registraciju u koju mora upisati svoje ime, prezime, email, lozinku, adresu i telefon. Nakon što napravi registraciju, može se prijaviti emailom i lozinkom. Nadalje, može željene proizvode nesmetano dodavati u košaricu. Ako želi, može potvrditi narudžbu i upisati potrebne podatke u dio za naplatu i potvrdom izvršiti kupovinu. Na slici 21. prikazana je početna strana mrežne trgovine koja je vidljiva posjetiteljima.



Slika 21. Izgled početne stranice mrežne trgovine

IZVOR: autor

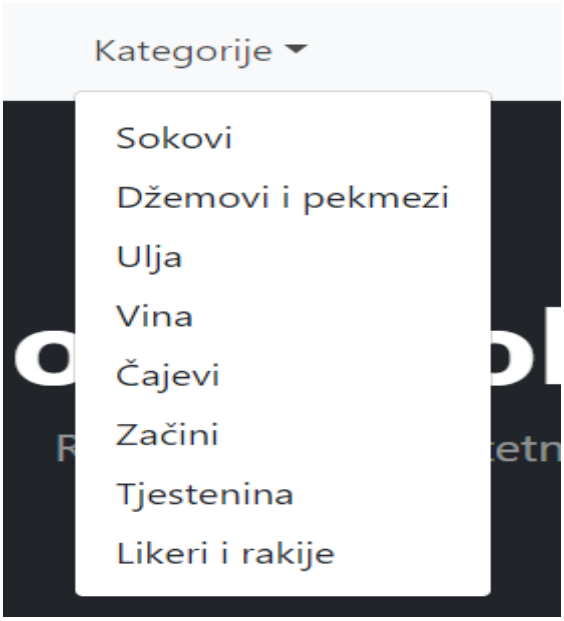
Posjetitelj može pregledavati sve proizvode i kategorije, može željene proizvode dodati u košaricu, ali bez autorizacije, tj. registracije i prijave, ne može završiti narudžbu. Slika 22. prikazuje izbornik za autorizaciju korisnika.



Slika 22. Padajući izbornik za autorizaciju korisnika

IZVOR: autor

Na gornjoj traci nalazi se padajući izbornik za odabir kategorija za lakši i bolji pregled željenih proizvoda, prikazano na slici 23.



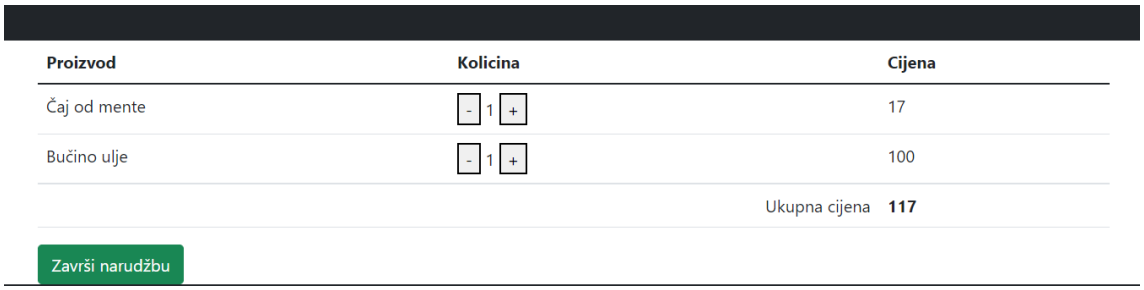
Slika 23. Padajući izbornik za odabir kategorije proizvoda

IZVOR: autor

Odabirom kategorije i željenog proizvoda iz kategorije, moguće je proizvod dodati u košaricu. Nakon dodavanja, ispisuje se obavijest da je proizvod dodan u košaricu (Slika 24.). U košarici se mogu povećavati i smanjivati količine proizvoda dodanih u nju. Pregled košarice prikazan na slici 25. moguć je prijavljenim korisnicima i nakon toga moguće je pritisnuti polje za završavanje narudžbe. U interakciji sa sustavom za kartično plaćanje otvara se forma (slika 5.) u koju se upisuju testni podaci za kartične transakcije(slika 16.). Nakon izvršene transakcije prikazuje se sažetak narudžbe. Vidljiv je kupcima, a administrator ima pravo mijenjati ga. Pet je mogućnosti statusa narudžbe, to su: u obradi, plaćeno, na dostavi, dostavljeno i otkazano.

Slika 24. Proizvod s obavijesti da je dodan u košaricu

IZVOR: autor



Slika 25. Stranica košarice i izračuna

IZVOR: autor

5. ZAKLJUČAK

Izradom mrežne aplikacije koja je prototip mrežne trgovine za prodaju lokalnih proizvoda, nastojala se stvoriti jednostavna mrežna trgovina za lako korištenje. Korisničko sučelje izrađeno je na način da bude što jednostavnije i pristupačnije svakom profilu korisnika, bez obzira na njegovo predznanje rada na računalu. Korišten je minimalistički dizajn kako korisnika ne bi zbunjivale nepotrebne informacije.

Korištenjem programskih jezika, mrežnih tehnologija, alata, sustava i biblioteka, kao što su: XAMPP, PHP, PHPMyAdmin, Visual Studio Code, HTML, CSS, JavaScript, jQuery, Bootrstrap, AJAX i Stripe. Izrađena je mrežna trgovina, građena kao CMS sustav za vlastito upravljanje sadržajem, koji funkcionira kao administrativni dio trgovine. Korisnički dio prikazuje podatke kojima se upravlja u administrativnom dijelu. Korisnik, da bi mogao pristupiti kupnji proizvoda iz trgovine, mora biti autoriziran. Sustavom Stripe omogućen je upis podataka koji su potrebni za kupovinu i sama kupnja. Po izvršenju kartične transakcije, kupac ima uvid u status narudžbe.

Sama izrada mrežne aplikacije zahtjevna je i kompleksna, iziskuje mnogo vremena i truda. U samom početku izrade javila se problematika oko načina na koji će se obavljati kupnja i kojim načinima plaćanja će biti moguća. Stoga je problem riješen implementacijom API-a koji je standardizirani način komunikacije s vanjskim servisi. Preko API-a se uspostavlja komunikacija košarice i mrežne trgovine, a sustavom Stripe omogućene su kartične transakcije. S obzirom na to da se na mrežnoj stranici izvršavaju novčane transakcije, bitna je sigurnost. Budući da se ne koriste gotova programska rješenja za izradu mrežne trgovine, neograničena je mogućnost nadogradnje, povećanje stupnja sigurnosti, mogućnosti dodatnih promjena kao što su dodavanje kupona, popusta, recenzija proizvoda, izražavanje drugih valuta plaćanja i dr.

6. POPIS LITERATURE

Internet:

[1] TUTORIAL.pdf (uns.ac.rs), 06. 9. 2022.

[2] 431BUvod u PHP programiranje (mev.hr), 15. 6. 2022.

[3] Visual Studio Code - Code Editing. Redefined , 10. 4. 2022.

[4] W3Schools Online Web Tutorials , 12. 4. 2022.

[5] HTML Styles CSS (w3schools.com), 7. 9. 2022.

[6] JavaScript.com , 13. 6. 2022.

[7] Bootstrap · Najpopularnija HTML, CSS i JS biblioteka na svijetu. (getbootstrap.com), 12. 4. 2022.

[8] PHP: Pripremljene izjave - Priručnik, 6. 9. 2022.

[9] Testing | Stripe Documentation, 15. 7. 2022.

7. PRILOZI

Slika 1. Programsko sučelje XAMPP alata 11
Slika 2. PHPMyAdmin sučelje 12
Slika 3. Sučelje Visual Studio Code 13
Slika 4. Izgled CSS koda 15
Slika 5. Izgled Stripe kartice za plaćanje 17
Slika 6. Htdocs direktorij 19
Slika 7. Prikaz osnovnih segmenata baze 20
Slika 8. Struktura baze 21
Slika 9. Includes direktorij, prikazan u VS Codu 22
Slika 10. Prikaz koda za prijavu administratora 23
Slika 11. Dio koda za dodavanje novog korisnika i zaštita podataka 24
Slika 12. Brisanje narudžbe 25
Slika 13. Provjera korisnika dio koda 26
Slika 14. API, prikaz u VS Codu 27
Slika 15. Kreiranje podataka za slanje prema Stripe-u 27
Slika 16. Prikaz testnih brojeva za test Stripe 28
Slika 17. Stranica za prijavu administratora 29
Slika 18. Izbornici koji sadrže liste i prikaz kategorija, korisnika, proizvoda i narudžbi 30
Slika 19. Stranica za dodavanje nove kategorije 31
Slika 20. Pregled narudžbi 32
Slika 21. Izgled početne stranice mrežne trgovine 33
Slika 22. Padajući izbornik za autorizaciju korisnika 34
Slika 23. Padajući izbornik za odabir kategorije proizvoda 34
Slika 24. Proizvod s obavijesti da je dodan u košaricu 35
Slika 25. Stranica košarice i izračuna 35