

Web trgovina prilagođena životnim potrebama korisnika

Antunović, Filip

Undergraduate thesis / Završni rad

2017

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **Josip Juraj Strossmayer University of Osijek, Faculty of Electrical Engineering, Computer Science and Information Technology Osijek / Sveučilište Josipa Jurja Strossmayera u Osijeku, Fakultet elektrotehnike, računarstva i informacijskih tehnologija Osijek**

Permanent link / Trajna poveznica: <https://um.nsk.hr/um:nbn:hr:200:351389>

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

Download date / Datum preuzimanja: **2024-11-29**

Repository / Repozitorij:

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



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

Sveučilišni preddiplomski studij računarstva

**WEB TRGOVINA PRILAGOĐENA ŽIVOTNIM
POTREBAMA KORISNIKA**

Završni rad

Filip Antunović

Osijek, 2017.

SADRŽAJ

1. UVOD	1
1.1 Zadatak završnog rada	1
2. PRIJEDLOG VLASTITOG RJEŠENJA WEB TRGOVINE	2
3. OPIS KORIŠTENIH TEHNOLOGIJA	4
3.1 Razvojna okolina JetBrains PhpStorm	4
3.2 Web poslužitelj XAMPP	4
4. KORISNIČKO SUČELJE	7
4.1 HTML	7
4.2 CSS	7
4.3 Front-end projekta	8
5. POSLUŽITELJSKA STRANA PROJEKTA	17
5.1 PHP	17
5.2 Programski alat phpMyAdmin	22
5.3 Baze podataka	23
6. NAČIN KORIŠTENJA I TESTIRANJE APLIKACIJE	28
6.1 Način korištenja aplikacije	28
6.2 Testiranje aplikacije	28
7. ZAKLJUČAK	32
LITERATURA	33
SAŽETAK	34
ABSTRACT	35
ŽIVOTOPIS	36
Prilozi (na CD-u)	37

1. UVOD

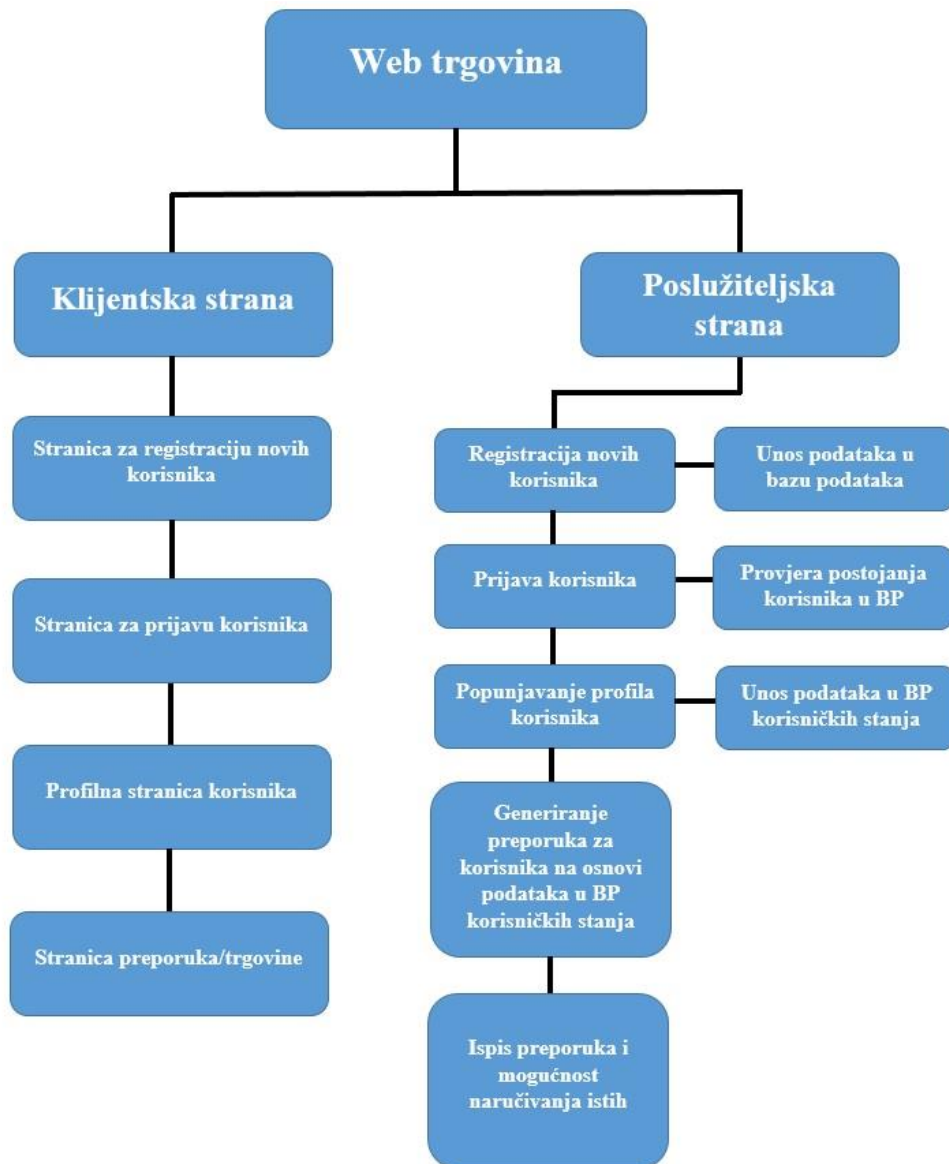
Zdravstveno stanja i životne navike značajno određuju potrebe korisnika, odnosno kupaca prehrambenih i suplimenarnih proizvoda. U ovom završnom radu je potrebno na temelju unesenih osobnih podataka korisnika, podataka o trenutnom zdravstvenom stanje, izraditi model preporuka za kupnju prehrambenih proizvoda, dodataka prehrani i drugih proizvoda koji utječu na kvalitetu života korisnika, odnosno kupca. Predloženi model treba ugraditi u web okolinu s bazom podataka zasnovanu na aktualnim web tehnologijama. Ta okolina treba omogućiti sučelje za registriranje, prijavu, korisnika, unos potrebnih zdravstvenih podataka u profil korisnika, mehanizam za odlučivanje s generiranjem preporuka, te web prodaju sugeriranih proizvoda prilagođenih zdravstvenom stanju korisnika. Poglavlje 2 opisuje postojeća rješenja u web trgovini i prijedlog vlastitog rješenja web trgovine. Također prikazuje blok dijagram idejnog rješenja. U poglavlju 3 prikazane su tehnologije korištene za izradu programske podrške s klijentske i poslužiteljske strane. Poglavlje 4 daje detaljno objašnjenje programskih jezika i koda korištenih za klijentsku stranu projekta. Poglavlje 5 opisuje programske jezike i programski kod poslužiteljske strane programskog rješenja. U poglavlju 6 prikazan je postupak testiranja i analize projekta.

1.1 Zadatak završnog rada

Zadatak ovog završnog rada je izraditi web okolinu s bazom podataka zasnovanu na aktualnim web tehnologijama. U radu je potrebno opisati sve faze izrade web okoline, počevši od izgleda web okoline, registracije korisnika, prilagođavanja njegovog profila, ispunjavanja potrebnih podataka, obrade podataka, izrade mehanizma za odlučivanje s generiranjem preporuka, web prodaje prilagođene životnim navikama korisnika.

2. PRIJEDLOG VLASTITOG RJEŠENJA WEB TRGOVINE

Web trgovina ili e-trgovina (eng. e-commerce) je web stranica koja omogućuje svojim internetskim posjetiteljima, korisnicima, naručivanje proizvoda putem Interneta. Kupovina preko Interneta može biti najbrži i najjeftiniji oblik kupovine određenih proizvoda koje kupac potražuje [1]. Prednost web trgovine je jednostavnost uporabe te povećanje profita zbog smanjenja troškova u odnosu na tradicionalnu trgovinu. Neke od najpoznatijih i profitabilnijih web trgovinu su Amazon.com, Ebay.com, Alibaba.com koje nude veliki raspon proizvoda. Web trgovine predodređene prehrani poput Fitocracy.com, MyFitnessPal.com, PEERtrainer posjeduju određene pristupe rješavanja problema poput kreiranja prehrane specifične za pojedinog korisnika, ponuda kupnje tih preporučenih proizvoda, mogućnost povezivanja s ostalim korisnicima, neka od tih rješenja biti će zastupljena u ovom završnom radu. Neke od postojećih okolina za programska rješenja u web trgovini su: PrestaShop, Magento, OpenCart, Spree commerce, Zen Cart, simpleCart. Slika 2.1 prikazuje blok dijagram idejnog rješenja klijentske i poslužiteljske strane web trgovine projekta.



Sl. 2.1 Blok dijagram idejnog rješenja

Izrada projekta podijeljena je u dvije etape. Prva etapa se sastoji od dizajna i pisanja programskog koda za klijentsku stranu projekta, dok se druga etapa sastoji od dodavanja interaktivnosti i povezivanja s bazom podataka te generiranja preporuka proizvoda.

3. OPIS KORIŠTENIH TEHNOLOGIJA

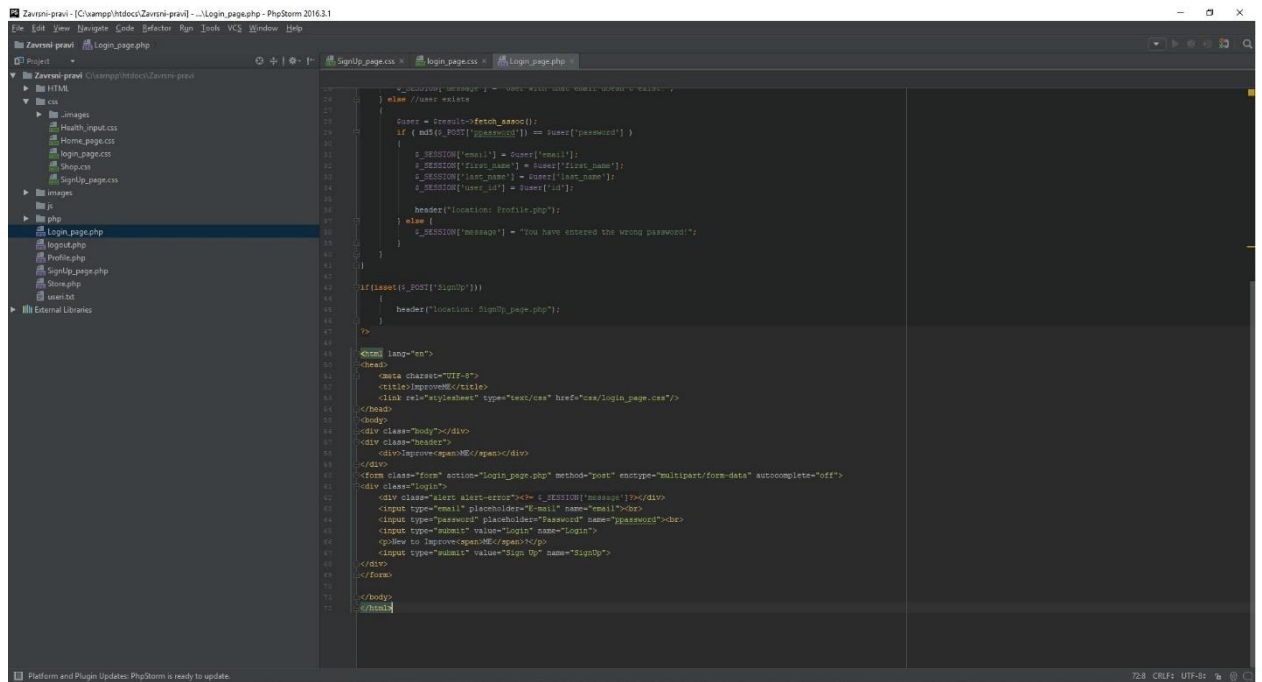
Za izradu završnog rada potrebno je koristiti JetBrains PhpStorm [2] za izradu front-end i back-end dijela rada, dok je XAMPP [3] potreban za izradu lokalnog web poslužitelja.

3.1 Razvojna okolina JetBrains PhpStorm

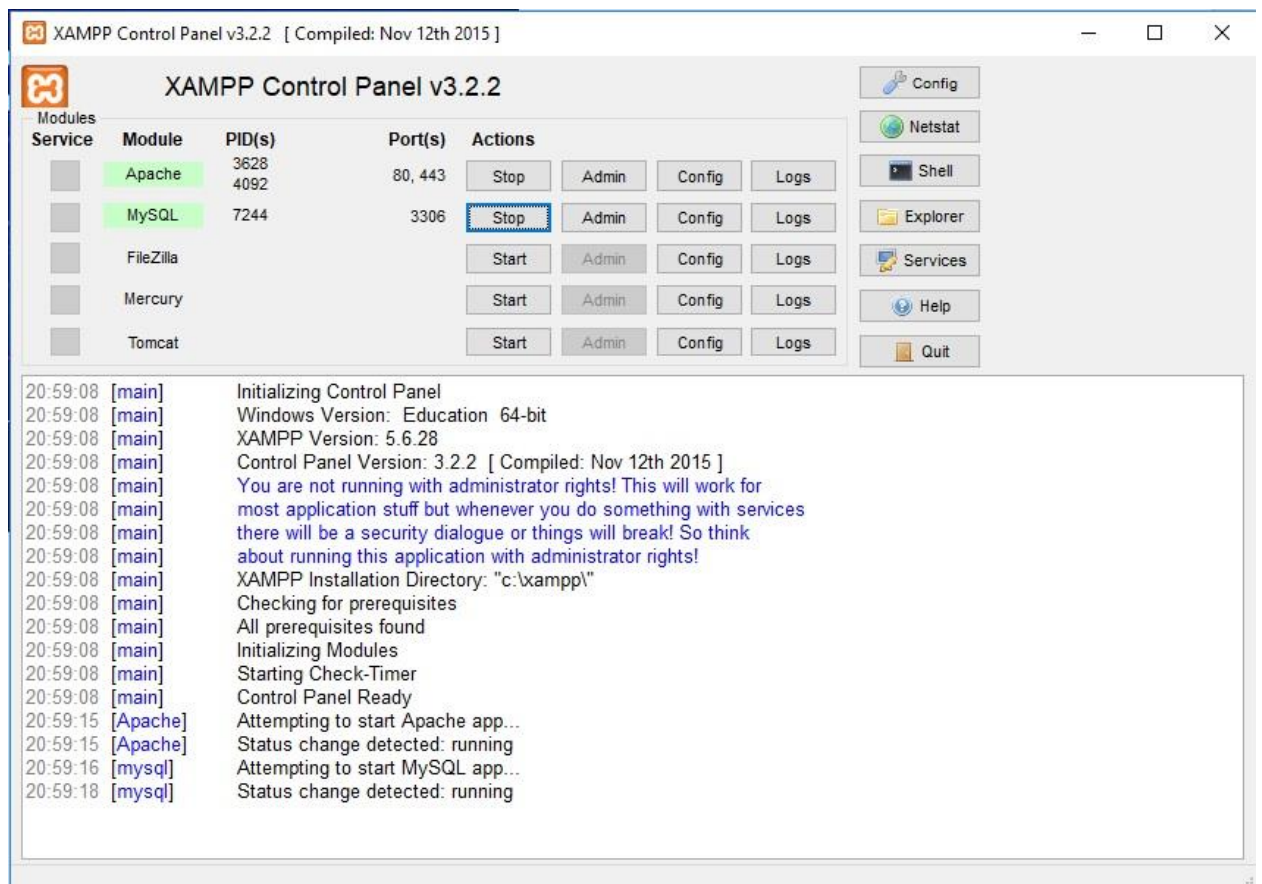
JetBrains PhpStorm je višeplatformsko okruženje za razvoj web okolina. Kao takvo podržava najnoviju inačicu PHP-a, HTML-a, CSS-a i SQL-a koje su potrebne za realizaciju projekta. PhpStorm podržava pametni PHP uređivač koda koji naglašava sintaksu koda, formatiranje i trenutno provjeravanje pogrešaka. Dozvoljava jednostavnu i laganu navigaciju programskog koda i pretragu programskog koda. PhpStorm također podržava pronalaženje pogrešaka, trenutno testiranje programskog koda direktno iz direktorija, datoteke ili klase. Za dodatnu prilagodbu korisnicima je omogućeno stvaranje vlastitih profila kojima se pojednostavljuje korištenje razvojne okoline PhpStorm. PhpStorm podržava HTML i CSS editor koji pruža mogućnost prikazivanja trenutnih promjena web stranice u pregledniku bez potrebe za osvježavanjem web stranice. PhpStorm pruža alate i programsku podršku za rad s bazama podataka i SQL-om u projektima. Omogućeno je povezivanje s bazama podataka, unos, dohvaćanje i izmjena podataka unutar baza podataka, SQL kod može biti dodan u druge programske jezike kako bi vršio potrebne radnje nad bazom podataka. PhpStorm posjeduje karakteristiku lokalne povijesti kojom se osiguravaju izvorne datoteke od slučajnih gubitaka ili izmjena, iako su one prouzročene od strane drugih aplikacija ili procesa. U bilo kojem trenutku moguće je provjeriti lokalnu povijest i vratiti se na prijašnju verziju projekta [4]. Na slici 3.1 može se vidjeti kako se sa lijeve strane nalazi glavna datoteka koja će sadržavati sve ostale datoteke potrebne za realizaciju projekta. U sredini se nalazi prostor za pisanje potrebnoga koda dok se iznad njega nalazi izbornik pomoću kojega se jednostavno prelazi iz jednoga dokumenta u drugi.

3.2 Web poslužitelj XAMPP

XAMPP je besplatan i višeplatformski web poslužitelj otvorenog koda razvijen od strane Apache Friends koji se sastoji od Apache HTTP poslužitelja, MariaDB bazepodataka i interpreter-a za skripte pisane u PHP i Perl programskom jeziku. XAMPP omogućuje jednostavno kreiranje lokalnog web poslužitelja za testiranje i razvoj web okolina [3].



Sl. 3.1 sučelje PhpStorm



Sl. 3.2 XAMPP sučelje

Za potrebe projekta korištene su mogućnosti lokalnog poslužitelja za Apache server i MySQL bazu podataka. Ostale mogućnosti koje XAMPP pruža su prema [3]:

1. Konfiguraciju poslužitelja
2. Ljuskicu za korištenje naredbi
3. Automatsko pokretanje poslužitelja
4. Upravljanje bazama podataka preko aplikacije phpMyAdmin
5. Konfiguriranje glavnih datoteka za Apache, PHP ili MySQL

4. KORISNIČKO SUČELJE

Prema [5], korisničko sučelje (eng. *front-end*) je način interakcije čovjeka s računalom kroz manipulaciju grafičkim elementima i dodacima uz pomoć tekstovnih poruka i obavijesti, klijentska strana programske podrške je ono što korisnik prvo ugleda prilikom posjete web stranici. Razvoj *front-end-a* predstavlja proces razvijanja i kodiranja nekog određenog usvojenog dizajna web stranice ili okoline te kao takav mora biti funkcionalan. *Front-end* se sastoji od više sastavnih dijelova poput:

- 1.HTML-a,
- 2.CSS-a,
- 3.JavaScript-a,
- 4.jQuery-a.

Za razvoj klijentske strane projekta korišteni su HTML i CSS.

4.1 HTML

HTML (eng. HyperText Markup Language) je jezik koji služi za prezentaciju sadržaja web stranice. Prva inačica HTML-a objavljena je 1993. godine, dok će se u razvoju projekta koristiti HTML5. Strukturu HTML dokumenta na najosnovnijoj razini se dijeli na:

1. Head
2. Body

Unutar njih mogu biti dodane razne vrste dokumenata poput .mp3, .jpg, .png, .mpg. Pomoću HTML-a moguće je oblikovati i dokumente i tekst koji je prikazan na web stranici. Tekst može biti prikazan bilo kojim fontom, obojan bilo kojom bojom, pozicioniran na specificiranom području web stranice, upisivan unutar dugmadi koju stvorimo isto pomoću HTML-a.

4.2 CSS

CSS (eng. Cascading Style Sheets) je stilski obrazac koji služi za oblikovanje HTML dokumenata [8]. Pomoću CSS-a je moguće odvojiti strukturu i oblik dokumenta. CSS se dijeli na tri vrste stilskih obrazaca:

1. Ugrađeni

2. Linijski
3. Vezani

Kod ugrađenog CSS-a stilska obilježja se ugrađuju unutar „<style>“ tag-a nekog HTML dokumenta. Stil pridružen takvom tag-u primjenjuje se na sve tag-ove iste vrste unutar dokumenta. Linijski CSS ugrađuje se kroz cijeli HTML dokument za razliku od ugrađenog CSS-a te se svakom pojedinom HTML tag-u pridružuje vlastito stilsko obilježje. Vezani CSS se pohranjuje u zasebnoj datoteci te se ta datoteka može povezati s bilo kojim HTML dokumentom. Za izradu projekta korišten je vezani CSS zbog jednostavnog povezivanja s više HTML dokumenata i mogućnosti korištenja jednog CSS dokumenta za potrebe više HTML dokumenata koji imaju elemente koji se pojavljuju na više HTML dokumenata.

4.3 Front-end projekta

Korisničko sučelje ili *front-end* projekta započinje web stranicom za prijavu postojećih korisnika koja se sastoji od pozadinske slike, forme za unos potrebnih podataka za provjeru postojanosti korisničkog računa, dugmadi za prijavu postojećih korisnika i registraciju novih korisnika. Izgled korisničkog sučelja će biti prikazan slici 4.1, a programski kod kojim je opisan sadržaj web stranice će biti prikazan na slici 4.2.



Sl. 4.1 Izgled korisničkog sučelja

```

<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>ImproveME</title>
  <link rel="stylesheet" type="text/css" href="css/login_page.css"/>
</head>
<body>
<div class="body"></div>
<div class="header">
  <div>Improve<span>ME</span></div>
</div>
<form class="form" action="Login_page.php" method="post" enctype="multipart/form-data" autocomplete="off">
<div class="login">
  <div class="alert alert-error"><?=$_SESSION['message']?></div>
  <input type="email" placeholder="E-mail" name="email"><br>
  <input type="password" placeholder="Password" name="ppassword"><br>
  <input type="submit" value="Login" name="Login">
  <p>New to Improve<span>ME</span>?</p>
  <input type="submit" value="Sign Up" name="SignUp">
</div>
</form>
</body>
</html>

```

Sl. 4.2 HTML programski kod stranice za prijavu korisnika

Prilikom registracije novih korisnika oni su preusmjereni s web stranice za prijavu postojećih korisnika na web stranicu za registraciju novih korisnika koja se sastoji od pozadinske slike i forme za unos potrebnih podataka kako bi se stvorio novi korisnički račun u projektu. Podaci potrebni za registraciju novog korisnika su ime korisnika, prezime korisnika, adresa e-pošte, zaporka te ponovni unos zaporke radi provjere da je dobra zaporka unesena. Izgled korisničkog sučelja će biti prikazan na slici 4.3, a programski kod kojim je opisan sadržaj web stranice će biti prikazan na slici 4.4.



Sl. 4.3 Stranica za registraciju novih korisnika

```

<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>ImproveME</title>
  <link rel="stylesheet" type="text/css" href="css/SignUp_page.css"/>
</head>
<body>
<div class="body"></div>
<div class="header">
  <div>Improve<span>ME</span></div>
</div>
<form class="form" action="SignUp_page.php" method="post" enctype="multipart/form-data" autocomplete="off">
<div class="sign_up">
  <div class="alert alert-error"><?= $_SESSION['message']?></div>
  <input type="text" placeholder="First Name" name="first_Name" required><br>
  <input type="text" placeholder="Last Name" name="last_Name" required><br>
  <input type="email" placeholder="E-mail" name="email" required><br>
  <input type="password" placeholder="New Password" name="Password" required><br>
  <input type="password" placeholder="Re-enter Password" name="REPassword" required><br>
  <input type="submit" value="Sign Up" name="SignUp">
</div>
</form>
</body>
</html>

```

Sl. 4.4 HTML kod stranice za registraciju

Nakon prijave postojećeg korisnika ili registracije novog na web stranicu, korisnik je preusmjeren na web stranicu svoga profila u kojemu je obavezan unijeti sve potrebne podatke za uspješno generiranje preporuka prilagođenih njegovom zdravstvenom stanju.

Podaci koje je korisnik obavezan unijeti su:

1. Spol
2. Visinu
3. Težinu
4. Opseg struka
5. Opseg trupa
6. Krvni tlak
7. Povijest bolesti
8. Alergije

Izgled forme za popunjavanje i profila korisnika je prikazan na slici 4.6, a programski kod kojim je opisan sadržaj forme i profilne web stranice je prikazan na slici 4.5.

```

<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>ImproveME</title>
  <link rel="stylesheet" type="text/css" href="css/Health_input.css"/>
</head>
<body>
<div class="header">
  <a href="Profile.php" class="profile">Profile</a>
  <a href="Store.php" class="store">Store</a>
  <a href="logout.php" class="logout">Logout</a>
</div>
<div class="body_center">
  <form class="form" action="Profile.php" method="post" enctype="multipart/form-data" autocomplete="off">
    <div class="health_form">
      <input type="radio" name="gender" value="Female"> Female <input type="radio" name="gender" value="Male"> Male
      <input type="text" placeholder="Height [cm]" name="Height"><br>
      <input type="text" placeholder="Weight [kg]" name="Weight"><br>
      <input type="text" placeholder="Waist circumference [cm]" name="Waist_crc"><br>
      <input type="text" placeholder="Vertical trunk circumference [cm]" name="Ver_crc"><br>
      <input type="text" placeholder="Blood pressure" name="BP"><br>
      <p>History of illness: </p>
      <input type="checkbox" name="Coronary" value="1">Coronary artery disease<br>
      <input type="checkbox" name="Diabetes" value="2">Type 2 diabetes<br>
      <input type="checkbox" name="Hypertension" value="3">Hypertension<br>
      <input type="checkbox" name="Stroke" value="4">Stroke<br>
      <p>Allergies: </p>
      <input type="checkbox" name="Peanut" value="5">Peanut<br>
      <input type="checkbox" name="Pollen" value="6">Pollen<br>
      <input type="checkbox" name="Shellfish" value="7">Shellfish<br>
      <input type="checkbox" name="Stone_fruits" value="8">Stone fruits<br>
      <input type="checkbox" name="Soy" value="9">Soy<br>
      <input type="checkbox" name="Wheat" value="10">Wheat<br>
      <input type="checkbox" name="Milk" value="11">Milk<br>
      <input type="checkbox" name="Eggs" value="12">Eggs<br>
      <input type="submit" name="Calculate" value="Calculate My Diet">
    </div>
  </form>
</div>
</body>
</html>

```

Sl. 4.5 HTML programski kod forme za popunjavanje korisničkog profila

The image shows a mobile application interface for a diet calculator. At the top right, there are two tabs: "Profile" and "Store". Below the tabs, there are two radio buttons for "Female" and "Male". The form contains five input fields: "Height [cm]", "Weight [kg]", "Waist circumference [cm]", "Vertical trunk circumference [cm]", and "Blood pressure". Below these fields, there are two sections: "History of illness:" and "Allergies:". The "History of illness:" section has four checkboxes for "Coronary artery disease", "Type 2 diabetes", "Hypertension", and "Stroke". The "Allergies:" section has seven checkboxes for "Peanut", "Pollen", "Shellfish", "Stone fruits", "Soy", "Wheat", "Milk", and "Eggs". At the bottom of the form, there is a button labeled "Calculate My Diet".

Sl. 4.6 Forma za popunjavanje

Na sljedećim slikama će biti prikazan CSS programski kod korišten na vezani način te će biti korišten od strane više web stranica. Na početku CSS programskog koda prvo je definirana „html“ klasa unutar koje se nalaze definicije pozadinske slike i prilagodba na različite web preglednike poput Opera-e i Mozilla Firefox-a. Nakon toga definirana je klasa „header“ koja određuje pozadinsku boju, prozirnost i poziciju zaglavlja.

```

html
{
    background: url("../images/matcha.jpg") no-repeat center center fixed;
    -webkit-background-size: cover;
    -moz-background-size: cover;
    -o-background-size: cover;
    background-size: cover;
}

.header
{
    background-color: black;
    -moz-opacity: 0.8;
    opacity: 0.8;
    color: white;
    position: fixed;
    top: 0;
    left: 0;
    width: 100%;
    height: 50px;
    padding: 0;
}

.body_center
{
    background-color: black;
    opacity: 0.8;
    -moz-opacity: 0.8;
    /*color: white;*/
    color: #a18d6c;
    position: relative;
    top: 42px;
    right: -20%;
    width: 60%;
    height: auto;
    padding: 0;
    margin: 0;
}

/*tu pocinje dodatak za MIS*/
.login p
{
    color: white;
    font-family: sans-serif;
    font-size: 180px;
    font-weight: 500;
}

```

Sl. 4.7 Prvi dio CSS programskog koda

Na slici 4.8 je prikazana klasa „login“ koja opisuje poziciju, visinu, širinu, boju, font, veličinu fonta. Nakon klase „login“ opisana je klasa „logout“ koja također opisuje iste karakteristike.

```
.login p span
{
    color: #aafaf2 !important;
}

.login
{
    position: absolute;
    top: calc(30% - 75px);
    left: calc(54% - 50px);
    height: 150px;
    width: 350px;
    padding: 10px;
    z-index: 2;
}

.login div
{
    float: left;
    color: #fff;
    font-family: sans-serif;
    font-size: 60px;
    font-weight: 300;
}

.login div span
{
    color: #aafaf2 !important;
}

.logout
{
    width: 100px;
    height: 50px;
    background: transparent;
    border: 1px transparent;
    border-radius: 2px;
    cursor: pointer;
    color: #a18d6c;
    font-family: sans-serif;
    font-size: 18px;
    font-weight: bold;
    padding: 1px;
    position: relative;
    right: -57%;
    top: 25%;
}
```

Sl. 4.8 Drugi dio CSS programskog koda

Na slikama 4.9 i 4.10 prikazan je CSS programski kod koji opisuje poziciju, visinu, širinu, granice, boju, font, veličinu fonta, pozadinu za klase „store“, „profile“, „health_form“.

```
.store
{
    width: 100px;
    height: 50px;
    background: transparent;
    border: 1px transparent;
    border-radius: 2px;
    cursor: pointer;
    color: #a18d6c;
    font-family: sans-serif;
    font-size: 18px;
    font-weight: bold;
    padding: 1px;
    position: relative;
    right: -54%;
    top: 25%;
}

.profile
{
    width: 100px;
    height: 50px;
    background: transparent;
    border: 1px transparent;
    border-radius: 2px;
    cursor: pointer;
    color: #a18d6c;
    font-family: sans-serif;
    font-size: 18px;
    font-weight: bold;
    padding: 1px;
    position: relative;
    right: -51%;
    top: 25%;
}

.health_form input[type=text]
{
    width: 250px;
    height: 30px;
    background: transparent;
    border: 2px solid rgba(252, 255, 246, 0.6);
    border-radius: 2px;
    color: #fff;
    font-family: sans-serif;
    font-size: 16px;
    font-weight: 400;
    padding: 4px;
    margin-top: 10px;
}
```

Sl. 4.9 Treći dio CSS programskog koda

```

.health_form input[type=text]:focus
{
    outline: none;
    border: 3px solid rgba(255,255,255,0.9);
}

.health_form input[type=submit]
{
    width: 260px;
    height: 35px;
    background: #fff;
    border: 1px solid #fff;
    cursor: pointer;
    border-radius: 2px;
    color: #a18d6c;
    font-family: sans-serif;
    font-size: 16px;
    font-weight: 400;
    padding: 6px;
    margin-top: 10px;
}

.health_form input[type=submit]:hover
{
    opacity: 0.8;
}

.health_form input[type=submit]:active
{
    opacity: 0.6;
}

.health_form input[type=submit]:focus
{
    outline: none;
}

::-webkit-input-placeholder
{
    color: white;
}

input:-moz-placeholder
{
    color: white;
}

.health_form
{
    position: relative;
    right: -39%;
}

```

Sl. 4.10 Četvrti dio CSS programskog koda

5. POSLUŽITELJSKA STRANA PROJEKTA

Prema [6], poslužiteljska strana (eng. *back-end*) podrazumjeva rad na arhitekturi i logici sustava, u ovom slučaju web sustava. *Back-end* u web okruženju je dio koji korisnik ne vidi, za razliku od klijentske strane (*front-end*), a nužan je za pravodobno funkcioniranje same web stranice. Za realizaciju projekta na poslužiteljskoj strani se koriste PHP i MySQL.

5.1 PHP

PHP (eng. *PHP: Hypertext Preprocessor*) je programski/skriptni jezik orijentiran po C i Perl sintaksi [7]. PHP je prvenstveno namijenjen programiranju dinamičnih web stranica, te se ističe širokom podrškom raznih internet protokola i baza podataka kao i raspoloživim programskim knjižnicama. „PHP/FI 2.0“ je službeno po prvi put izdan 1997. godine te je bio korišten na jako malom broju web stranica toga vremena, no to se kasnije promijenilo i PHP je postao vodeći programski/skriptni jezik za izradu web stranica. Prema [7], PHP je jednostavan i istovremeno moćan skriptni jezik otvorenog izvornog koda za izradu dinamičkog web sadržaja. Milijuni web stranica na kojima se koristi PHP dokaz su njegove jednostavnosti i popularnosti. Koriste ga programeri koji cijene fleksibilnost i brzinu te web dizajneri kojima odgovaraju njegove mogućnosti i lakoća upotrebe. PHP se može koristiti na tri osnovna načina:

1. Skriptiranje sa strane poslužitelja
2. Naredbeno skriptiranje
3. GUI aplikacije sa strane klijenta

PHP podržava sve popularne sustave baza podataka, uključujući MySQL, PostgreSQL, Oracle, Sybase, SQLite i baze podataka koje su kompatibilne s ODBC-om. Prilikom izrade završnog rada korišten je PHP 7. Na slici 5.1 prikazan je PHP programski kod web stranice za prijavu korisnika.

```

<?php
session_start();
$_SESSION['message']='';

$servername = "localhost";
$username = "root";
$password = "mypass123";
$databasename = "accounts";
// Create connection
$conn = new mysqli($servername, $username, $password,$databasename);
// Check connection
if ($conn->connect_error) {
    die("Connection failed: " . $conn->connect_error);
}
echo "Connected successfully";

if(isset($_POST['Login']))
{
    $email = $conn->escape_string($_POST['email']);
    $result = $conn->query("SELECT * FROM users WHERE email='$email'");

    if ($result->num_rows == 0) //user doesn't exist
    {
        $_SESSION['message'] = "User with that email doesn't exist!";
    } else //user exists
    {
        $user = $result->fetch_assoc();
        if ( md5($_POST['password']) == $user['password'] )
        {
            $_SESSION['email'] = $user['email'];
            $_SESSION['first_name'] = $user['first_name'];
            $_SESSION['last_name'] = $user['last_name'];
            $_SESSION['user_id'] = $user['id'];

            header("location: Profile.php");
        } else {
            $_SESSION['message'] = "You have entered the wrong password!";
        }
    }
}

if(isset($_POST['SignUp']))
{
    header("location: SignUp_page.php");
}
?>

```

Sl. 5.1 PHP programski kod korišten za web stranicu prijave korisnika

Na početku programskog koda započinje se sesija linijom programskog koda „session start()“, nakon čega dolazi otvaranje konekcije s bazom podataka i provjera da li je konekcija uspostavljena. Ako je uspostavljena konekcija s bazom podataka izvršava se provjera da li postoji unesena adresa e-pošte u bazi podataka. Kada je utvrđeno da postoji korisnik s zadanom email adresom i ako je zaporka potvrđena, zapamtit će se određeni podaci o korisniku za vrijeme sesije te će korisnik biti proslijeđen na web stranicu profila korisnika. Ako je novi korisnik u pitanju, on će klikom na dugme „Sign Up“ biti proslijeđen na web stranicu za kreiranje novog korisničkog profila. Slika 5.2 prikazuje PHP programski kod web stranice za registraciju novih korisnika.

```

<?php
session_start();
$_SESSION['message']='';

$servername = "localhost";
$username = "root";
$password = "mypass123";
$databasename = "accounts";
// Create connection
$conn = new mysqli($servername, $username, $password,$databasename);
// Check connection
if ($conn->connect_error) {
    die("Connection failed: " . $conn->connect_error);
}
echo "Connected successfully";

if($_SERVER['REQUEST_METHOD'] == 'POST')
{
    if ($_POST['Password'] == $_POST['REPassword'])
    {
        $first_name = $conn->real_escape_string($_POST['first_Name']);
        $last_name = $conn->real_escape_string($_POST['last_Name']);
        $email = $conn->real_escape_string($_POST['email']);
        $password = md5($_POST['Password']);

        $_SESSION['email'] = $email;

        $sql = "INSERT INTO users (first_name, last_name, email, password)"
            . "VALUES ('$first_name', '$last_name', '$email', '$password)";

        if ($conn->query($sql) === true) {
            $_SESSION['message'] = "Registration successful!"
                . "Added $first_name to the database!";
            header("location: Login_page.php");
        }
    }
    else
    {
        $_SESSION['message'] = "Two passwords do not match!";
    }
}
?>

```

Sl. 5.2 PHP programski kod web stranice za registraciju novih korisnika

Na početku programskog koda započinje se sesija te odmah nakon nje konekcija na bazu podataka unutar koje će se spremati podaci novog korisnika. Nakon provjere jednakosti unesenih zaporki uneseni podaci poput imena korisnika i njegovog prezimena pamte se u razne varijable koje će biti unesene u bazu podataka pomoću SQL upita prikazanog na slici.

Na slici 5.3 prikazan je PHP programski kod web stranice za popunjavanje profila korisnika.

```
<?php
session_start();
$_SESSION['message']='';
$user_id=$_SESSION['user_id'];

$servername = "localhost";
$username = "root";
$password = "mypass123";
$databasename = "accounts";
// Create connection
$conn = new mysqli($servername, $username, $password,$databasename);

if ( isset($_POST['Coronary']) && ($_POST['Coronary'] == "1" ) )
{
    if(isset($_POST['Calculate']))
    {
        $conn->query("INSERT INTO userstanja( user_id, stanje_id) VALUES('$user_id','1')");
        header("location: Store.php");
    }
}

if ( isset($_POST['Diabetes']) && ($_POST['Diabetes'] == "2" ) )
{
    if(isset($_POST['Calculate']))
    {
        $conn->query("INSERT INTO userstanja( user_id, stanje_id) VALUES('$user_id','2')");
        header("location: Store.php");
    }
}

if ( isset($_POST['Hypertension']) && ($_POST['Hypertension'] == "3" ) )
{
    if(isset($_POST['Calculate']))
    {
        $conn->query("INSERT INTO userstanja( user_id, stanje_id) VALUES('$user_id','3')");
        header("location: Store.php");
    }
}

if ( isset($_POST['Stroke']) && ($_POST['Stroke'] == "4" ) )
{
    if(isset($_POST['Calculate']))
    {
        $conn->query("INSERT INTO userstanja( user_id, stanje_id) VALUES('$user_id','4')");
        header("location: Store.php");
    }
}
}
```

Sl. 5.3 PHP programski kod za popunjavanje profila korisnika

Nakon započete sesije i uspostavljene konekcije na bazu podataka unutar koje će se spremati podaci popunjenog profila korisnika, dolazi do provjere zdravstvenih stanja koja

je korisnik unio na svome profilu te njihov unos u bazu podataka. Slika 5.4 prikazuje PHP programski kod web stranice za preporuku proizvoda korisniku.

```
<?php
session_start();
$_SESSION['message']='';
$user_id=$_SESSION['user_id'];

$servername = "localhost";
$username = "root";
$password = "mypass123";
$databasename = "accounts";

$conn = new mysqli($servername, $username, $password,$databasename);

$result = $conn->query("SELECT DISTINCT namirnice.namirnica FROM userstanja
INNER JOIN stanja ON stanja.id=userstanja.stanje_id
INNER JOIN odlucivanje ON odlucivanje.stanje=stanja.id
INNER JOIN namirnice ON namirnice.id=odlucivanje.namirnica
WHERE userstanja.user_id='{$user_id}'");

while($row=$result->fetch_assoc())
{
    $new_array[]=$row;
}

if(isset($_POST['Logout']))
{
    header("location: Login_page.php");
}

?>
```

Sl. 5.4 PHP programski kod web stranice za preporuku proizvoda korisniku

Na početku programskog koda započeta je sesija i uspostavljena konekcija s bazom podataka kako bi se pomoću SQL upita moglo odlučiti koje proizvode treba preporučiti korisniku prema njegovim unesenim podacima u bazu podataka. Na slici 5.5 prikazan je PHP programski kod za odjavu korisnika.

```
1 <?php
2 session_start();
3 unset($_SESSION['email']);
4 unset($_SESSION['first_name']);
5 unset($_SESSION['last_name']);
6 session_destroy();
7 session_write_close();
8
9 header('Location: Login_page.php');
10 d👤;
11 ?>
```

Sl. 5.5 PHP programski kod za odjavu korisnika

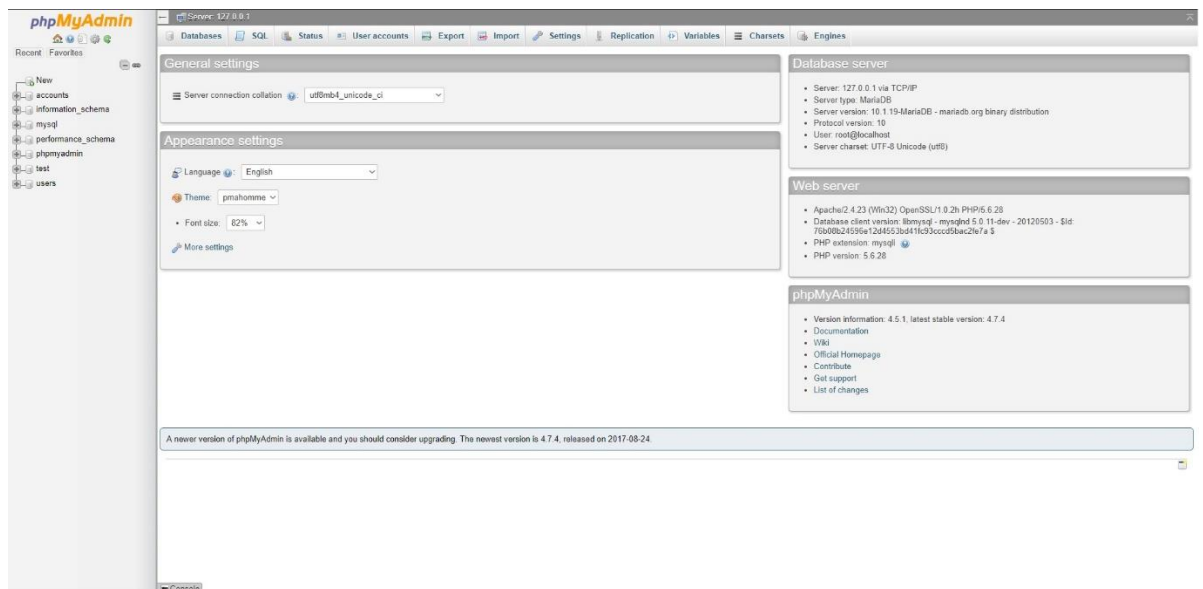
Nakon započinjanja sesije potrebno je obrisati sve ranije zapamćene podatke te nakon toga uništiti sesiju i vratiti korisnika na web stranicu za prijavu korisnika.

5.2 Programski alat phpMyAdmin

Programski alat PhpMyAdmin je besplatni programski alat napisan u PHP-u, čija je namjena rukovanje administracije MySQL preko web-a. Neke od glavnih karakteristike PhpMyAdmin-a su prema [3]:

1. Web sučelje
2. Upravljanje bazom podataka MySQL i MariaDB
3. Uvoz podataka iz CSV i SQL
4. Izvoz podataka u raznim formatima(CSV, SQL, XML, PDF, ...)
5. Prikaz baza podataka u PDF-u

Slika 5.6 prikazuje korisničko sučelje phpMyAdmin-a.



Sl. 5.6 Korisničko sučelje phpMyAdmin-a

Na slici s lijeve strane nalazi se popis svih baza podataka koje je korisnik stvorio do sada. Unutar svake baze podataka može se nalaziti više tablica koje korisnik samostalno kreira. Na vrhu sučelja nalaze se opcije za upravljanje bazama podataka poput SQL sučelja za pisanje SQL programskog koda, opcija za unos ili ispis već postojećih baza podataka te podešavanje postavki phpMyAdmina. U sredini sučelja se nalaze mogućnosti odabira jezika kojime će se korisnik služiti za vrijeme korištenja phpMyAdmina.

5.3 Baze podataka

Baza podataka se može definirati kao zbirka međusobno povezanih podataka pri čemu vrijedi da se ti podaci dijele između različitih programa, aplikacija i korisnika uz uvjet da postoji jedinstveni i kontrolirani pristup unosu, brisanju, izmjeni i dohvaćanju podataka.

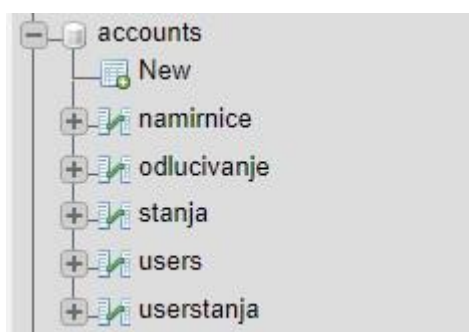
Baze podataka kao zajedničke osobine, karakteristike, imaju:

1. Apstraktni model podataka
2. Visoku razinu pristupa ili upitnih jezika
3. Upravljanje transakcijama
4. Kontrola pristupa i vlasništvo nad podacima
5. Validaciju podataka i provjeru konzistentnosti
6. Oporavak podataka nakon ispada sustava.

Baze podataka mogu biti organizirane u više modela:

1. Hijerarhijski
2. Mrežni
3. Relacijski
4. Relacijske baze podataka s objektno orijentiranim proširenjem
5. Objektno orijentirane baze.

Pri izradi završnog rada korišten je relacijski model baze podataka. Na slici 5.7 prikazana je baza podataka korištena za realizaciju projekta.



Sl. 5.7 Baza podataka korištena za realizaciju projekta

Unutar baze podataka naziva „accounts“ nalaze se baze „namirnice“, „odlucivanje“, „stanja“, „users“, „userstanja“. Na slici 5.8 prikazana je baza „stanja“ koja predstavlja sva moguća zdravstvena stanja koja korisnik može imati.

					id	stanje		
<input type="checkbox"/>		Edit		Copy		Delete	1	Coronary artery disease
<input type="checkbox"/>		Edit		Copy		Delete	2	Type 2 diabetes
<input type="checkbox"/>		Edit		Copy		Delete	3	Hypertension
<input type="checkbox"/>		Edit		Copy		Delete	4	Stroke
<input type="checkbox"/>		Edit		Copy		Delete	5	Peanut
<input type="checkbox"/>		Edit		Copy		Delete	6	Pollen
<input type="checkbox"/>		Edit		Copy		Delete	7	Shellfish
<input type="checkbox"/>		Edit		Copy		Delete	8	Stone fruits
<input type="checkbox"/>		Edit		Copy		Delete	9	Soy
<input type="checkbox"/>		Edit		Copy		Delete	10	Wheat
<input type="checkbox"/>		Edit		Copy		Delete	11	Milk
<input type="checkbox"/>		Edit		Copy		Delete	12	Eggs

Sl. 5.8 Baza podataka „stanja“

Slika prikazuje raspodjelu primarnih ključeva koji se nalaze unutar tablice pod imenom „id“ i odgovarajućih zdravstvenih stanja koja se nalaze u stupcu pod imenom „stanje“. Na slikama 5.9 i 5.10 prikazana je podjela proizvoda po zdravstvenom stanju. Na slikama 5.9 i 5.10 nalaze se primjeri dijela tablice podjele proizvoda po zdravstvenom stanju. Ova relacijska tablica sastoji se od primarnog ključa „id“, stranog ključa „stanje“ i stranog ključa „namirnica“. Svakom stanju pridodjeljeni su proizvodi koje se preporučuju za to stanje te se pomoću ove tablice odlučuje što će biti preporučeno pojedinom korisniku. Na slici 5.11 prikazan je tablica registriranih korisnika.

Options				id	namirnica			
<input type="checkbox"/>		Edit		Copy		Delete	1	Beef
<input type="checkbox"/>		Edit		Copy		Delete	2	Chicken
<input type="checkbox"/>		Edit		Copy		Delete	3	Rice
<input type="checkbox"/>		Edit		Copy		Delete	4	Pasta
<input type="checkbox"/>		Edit		Copy		Delete	5	Protein
<input type="checkbox"/>		Edit		Copy		Delete	6	Salmon
<input type="checkbox"/>		Edit		Copy		Delete	7	Nuts
<input type="checkbox"/>		Edit		Copy		Delete	8	Green tea
<input type="checkbox"/>		Edit		Copy		Delete	9	Asparagus
<input type="checkbox"/>		Edit		Copy		Delete	10	Beans
<input type="checkbox"/>		Edit		Copy		Delete	11	Broccoli
<input type="checkbox"/>		Edit		Copy		Delete	12	Blueberry
<input type="checkbox"/>		Edit		Copy		Delete	13	Bread
<input type="checkbox"/>		Edit		Copy		Delete	14	Cauliflower
<input type="checkbox"/>		Edit		Copy		Delete	15	Apple
<input type="checkbox"/>		Edit		Copy		Delete	16	Carrot
<input type="checkbox"/>		Edit		Copy		Delete	17	Lemons
<input type="checkbox"/>		Edit		Copy		Delete	18	Oranges
<input type="checkbox"/>		Edit		Copy		Delete	19	Pork
<input type="checkbox"/>		Edit		Copy		Delete	20	Tuna
<input type="checkbox"/>		Edit		Copy		Delete	21	Redberry
<input type="checkbox"/>		Edit		Copy		Delete	22	Omega 3 pills
<input type="checkbox"/>		Edit		Copy		Delete	23	Peanut butter
<input type="checkbox"/>		Edit		Copy		Delete	24	Calcium
<input type="checkbox"/>		Edit		Copy		Delete	25	Magnesium

Sl. 5.9 Tablica podjele proizvoda po zdravstvenom stanju

← T →				id	stanje	namirnica			
<input type="checkbox"/>		Edit		Copy		Delete	86	4	24
<input type="checkbox"/>		Edit		Copy		Delete	87	4	25
<input type="checkbox"/>		Edit		Copy		Delete	88	5	1
<input type="checkbox"/>		Edit		Copy		Delete	89	5	2
<input type="checkbox"/>		Edit		Copy		Delete	90	5	3
<input type="checkbox"/>		Edit		Copy		Delete	91	5	4
<input type="checkbox"/>		Edit		Copy		Delete	92	5	5
<input type="checkbox"/>		Edit		Copy		Delete	93	5	6
<input type="checkbox"/>		Edit		Copy		Delete	94	5	8
<input type="checkbox"/>		Edit		Copy		Delete	95	5	9
<input type="checkbox"/>		Edit		Copy		Delete	96	5	10
<input type="checkbox"/>		Edit		Copy		Delete	97	5	11
<input type="checkbox"/>		Edit		Copy		Delete	98	5	12
<input type="checkbox"/>		Edit		Copy		Delete	99	5	13
<input type="checkbox"/>		Edit		Copy		Delete	100	5	14
				▼ id	stanje	namirnica			
<input type="checkbox"/>		Edit		Copy		Delete	101	5	15
<input type="checkbox"/>		Edit		Copy		Delete	102	5	16
<input type="checkbox"/>		Edit		Copy		Delete	103	5	17
<input type="checkbox"/>		Edit		Copy		Delete	104	5	18
<input type="checkbox"/>		Edit		Copy		Delete	105	5	19
<input type="checkbox"/>		Edit		Copy		Delete	106	5	20
<input type="checkbox"/>		Edit		Copy		Delete	107	5	21
<input type="checkbox"/>		Edit		Copy		Delete	108	5	22
<input type="checkbox"/>		Edit		Copy		Delete	109	5	24
<input type="checkbox"/>		Edit		Copy		Delete	110	5	25
<input type="checkbox"/>		Edit		Copy		Delete	111	6	1
<input type="checkbox"/>		Edit		Copy		Delete	112	6	2

Sl. 5.10 Tablica podjele proizvoda po zdravstvenom stanju

			id	first_name	last_name	email	password				
<input type="checkbox"/>		Edit		Copy		Delete	1	Filip	Antunovic	topdog@dogg.com	bf95ef447f966d150b7f4e94d4a11f9d
<input type="checkbox"/>		Edit		Copy		Delete	2	John	Doe	johhnyboy@doe.com	1c522a151fa13b14d4c77c04d5324b9e
<input type="checkbox"/>		Edit		Copy		Delete	3	Mario	Komiza	koma@bio.com	f5f38fb329721d69b90c83095642d148
<input type="checkbox"/>		Edit		Copy		Delete	4	Ivana	Trmiz	trma@miz.com	69ca84b880fd1b45b86ffd9005675d74
<input type="checkbox"/>		Edit		Copy		Delete	5	Matea	Simic	simke@got.com	abd7911a7b95ee5a74b70c5dbf9cb0cb
<input type="checkbox"/>		Edit		Copy		Delete	6	Dora	Maril	dora@tamo.com	b08aa16999e264f9f1ffb0321aeb46a8

Sl. 5.11 Tablica registriranih korisnika

Tablica 5.11 se sastoji od stupca primarnog ključa „id“, stupca imena korisnika „first_name“, stupca prezimena korisnika „last_name“, stupca elektronske pošte korisnika „email“ te stupca zaporki korisnika „password“ unutar kojeg se nalaze zapisane zaporke u šifriranom obliku koji je nastao korištenjem md5 kriptografske hash funkcije. Na slici 5.12 prikazana je tablica korisnika i njihovih zdravstvenih stanja.

				id	user_id	stanje_id			
<input type="checkbox"/>		Edit		Copy		Delete	1	1	2
<input type="checkbox"/>		Edit		Copy		Delete	2	2	1
<input type="checkbox"/>		Edit		Copy		Delete	3	2	5
<input type="checkbox"/>		Edit		Copy		Delete	4	2	10
<input type="checkbox"/>		Edit		Copy		Delete	5	1	2

Sl. 5.12 Tablica korisnika i njihovih stanja

Tablica se sastoji od stupca primarnog ključa „id“, stupca stranog ključa „user_id“ te stupca stranog ključa „stanje_id“. U ovu tablicu podaci se spremaju nakon unosa podataka korisnika u njegov profil.

6. NAČIN KORIŠTENJA I TESTIRANJE APLIKACIJE

6.1 Način korištenja aplikacije

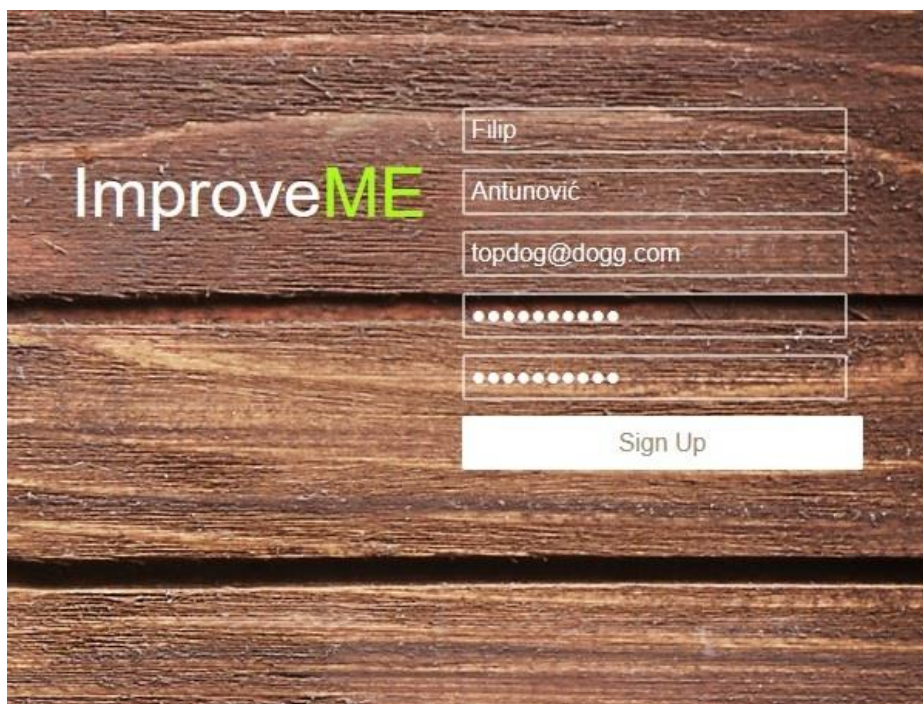
Prvi put kada se korisnik susreće s aplikacijom nalazi se na stranici za prijavu korisnika. Ako je korisnik već registriran, on unosi potrebne podatke za prijavu, a ako korisnik nije registriran treba kliknuti na dugme „Sign Up“ kako bi bio preusmjeren na stranicu za registraciju novih korisnika gdje također unosi potrebne podatke kako bi kreirao svoj korisnički račun. Nakon prijave ili registracije korisnik je preusmjeren na web stranicu svoga profila gdje je obavezan popuniti profil kako bi se mogle generirati preporuke proizvoda određene njegovim zdravstvenim stanjem. Nakon popunjavanja profila i stiskanja „Calculate My Diet“ korisnik je preusmjeren na web stranicu gdje su mu prikazani proizvodi koji se preporučuju njemu te ih on može naručiti ako želi unosom svoje adrese.

6.2 Testiranje aplikacije

Kako bi se uvjerali da web okolina radi ispravno, provest će se testiranje svih dijelova. Na slikama 6.1 i 6.2 prikazana je web stranica za registraciju novih korisnika.



Sl. 6.1 Web stranica za registraciju korisnika



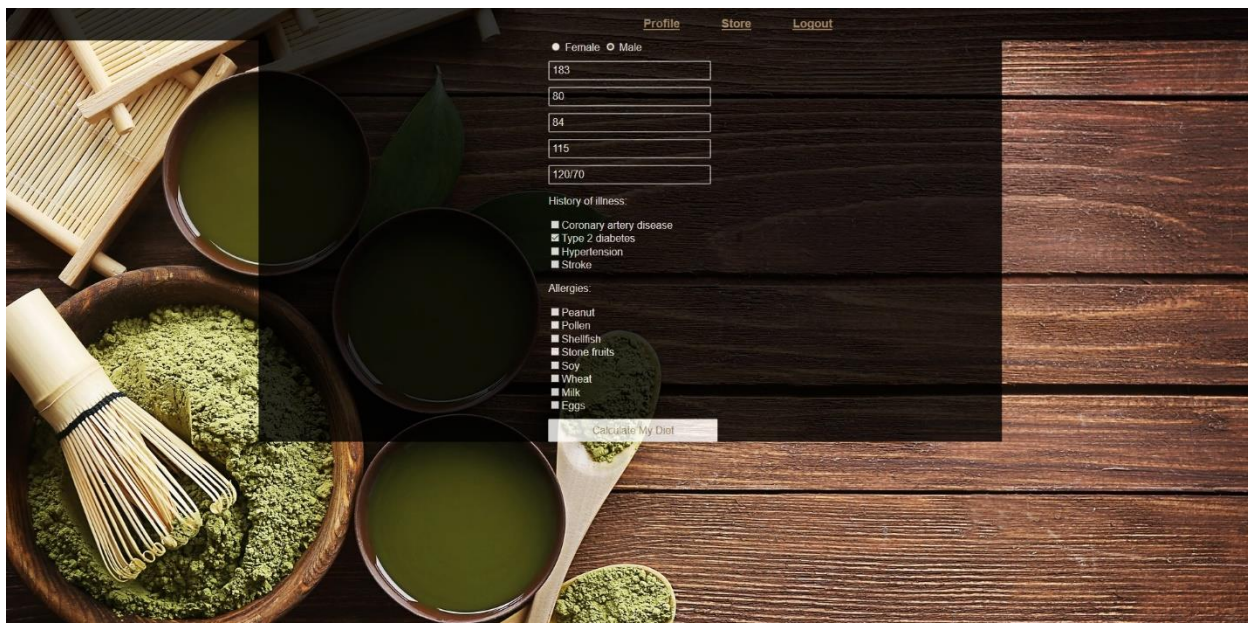
Sl. 6.2 Web stranica za registraciju korisnika

Za kreiranje novog korisničkog računa potrebno je unijeti ime korisnika, prezime korisnika, adresu elektronske pošte, zaporku te ponovo unijeti zaporku radi provjere da li je zaporka koju korisnik želi koristiti pravilno zapisana. Na slici 6.3 prikazana je prijava korisnika.

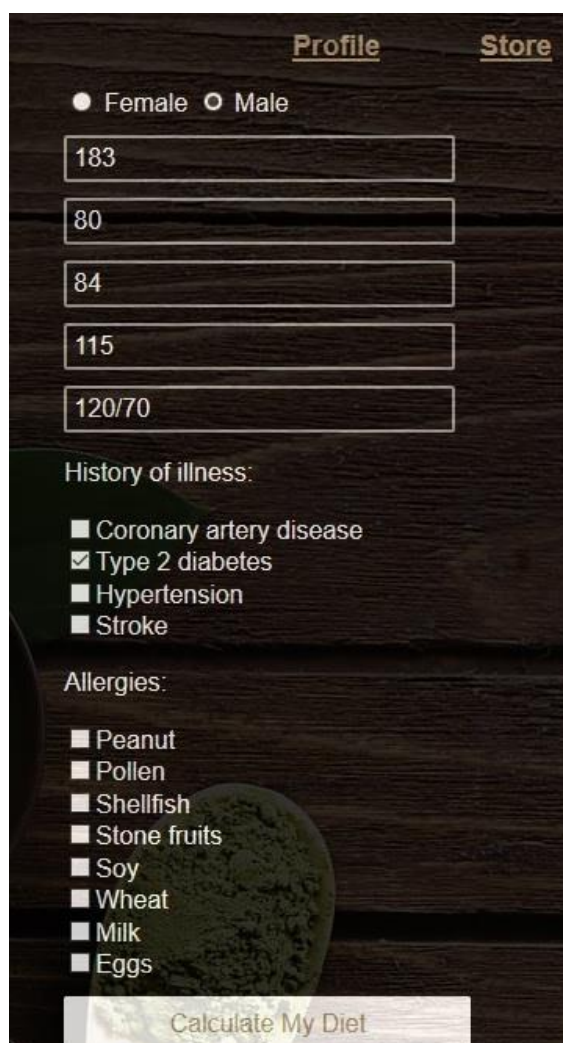


Sl. 6.3 Prijavljivanje registriranog korisnika

Ako je korisnik već registriran, on se prijavljuje na web stranici za prijavu unosom svoje adrese elektronske pošte i zaporke. Na slici 6.4 prikazan je profil kojeg korisnik popunjava, a na slici 6.5 uvećani prikaz istoga.



Sl. 6.4 Profil korisnika



Sl. 6.5 Uvećani prikaz profila

Korisnik treba specificirati svoj spol, tjelesne mjere te povijest bolesti i alergija od kojih je bolovao ili boluje. Na slici 6.6 prikazana je preporuka proizvoda korisniku.



Sl. 6.6 Prikaz preporuka i mogućnost naručivanja

Nakon što korisnik ispuni svoj profil prikazuju mu se preporučeni proizvodi te omogućuje naručivanje proizvoda na kućnu adresu gdje će proizvodi biti naplaćeni po pouzeću. Prema ponuđenim proizvodima koji se mogu naručiti prikazano je točno rješenje u odnosu na unesene podatke na slici 6.5.

7. ZAKLJUČAK

U ovom završnom radu obrađena je tema kreiranja web trgovine prilagođene životnim potrebama korisnika. Kao ciljevi završnog rada bili su zadani samostalno učenje novih tehnologija pomoću kojih treba realizirati projekt. Za praktični dio rada izrađena je web aplikacija pod nazivom „ImproveME“ koja omogućuje registraciju novih korisnika te prijavljivanje postojećih korisnika kako bi ispunili svoj profil zdravstvenim stanjima te dobili preporuku proizvoda vezanu uz njihovo zdravstveno stanje. Korištene programske tehnologije omogućuju razvoj programskog rješenja brzo i zadovoljavajuće zbog svoje velike programske podrške koju pružaju korisniku. Praktični dio rada je moguće nadograditi dodatnim funkcijama poput brojila kalorija dnevnog unosa korisnika te implementacija programa za tjelovježbu koji specificira vježbe i njihovu učestalost tijekom tjedna. Također postoji mogućnost implementacije aspekata društvene mreže kako bi se korisnicima omogućila međusobna interakcija putem poruka i grupa po uzoru na trenutne društvene mreže.

LITERATURA

- [1] „E-trgovina“, Ex13, dostupno na www.hr.wikipedia.org/wiki/E-trgovina, zadnji pristup ostvaren 06.09.2017.
- [2] „JetBrains PhpStorm“, JetBrains, dostupno na www.jetbrains.com/phpstorm/, zadnji pristup ostvaren 06.09.2017.
- [3] „XAMPP“, Apachedriends, dostupno na www.apachefriends.org/index.html, zadnji pristup ostvaren 06.09.2017.
- [4] „JetBrains PhpStorm“, JetBrains, dostupno na www.jetbrains.com/php/storm/features, zadnji pristup ostvaren 06.09.2017.
- [5] „Grafičko korisničko sučelje“, YurikBot, dostupno na www.hr.wikipedia.org/wiki/Grafičko_korisničko_sučelje, zadnji pristup ostvaren 06.09.2017.
- [6] „Backend Developer: A što oni rade po cijele dane?“, Mia Biberović, dostupno na www.netokracija.com, zadnji pristup ostvaren 03.09.2017.
- [7] R., Lerdorf, K., Tatroe, P., MacIntyre, „Programiranje PHP“, Dobar Plan, Zagreb, 2009.
- [8] „CSS“, Zmayo, dostupno na www.hr.wikipedia.org/wiki/CSS, zadnji pristup ostvaren 10.09.2017.

SAŽETAK

Cilj ovog završnog rada je izrada web trgovine prilagođene životnim potrebama korisnika. Za izradu završnog rada korištena je razvojna okolina JetBrains PhpStorm, web poslužitelj XAMPP i programski jezici PHP, MySQL/SQL te HTML i CSS. Kako bi se mogla koristiti web stranica korisnik se prvo mora registrirati te prijaviti, nakon čega je potrebno ispuniti korisnički profil potrebnim podacima kako bi se generirala preporuka specifična svakom korisniku. Testiranjem je pokazano da razvijena aplikacija radi ispravno i brzo, te da se za unos podataka prikazan u poglavlju 6. daje ispravan izbor proizvoda.

Ključne riječi: generiranje preporuka, HTML,MySQL,PHP, web trgovina

ABSTRACT

The aim of this final paper is to create a web store tailored to the needs of the users. For development of the final paper technologies used were JetBrains PhpStorm, the XAMPP web server and the PHP, MySQL / SQL and HTML and CSS programming languages. In order to be able to use a website, the user must sign up and log in, after which user profile needs to be filled out to generate the specific recommendation for each user. Testing has shown that the developed application works correctly and quickly, and that the data entry presented in Chapter 6 provides the right choice of product.

Keywords: HTML, MySQL, PHP, recommendation generator, web store

ŽIVOTOPIS

Filip Antunović rođen 30. siječnja 1995. godine u Osijeku. Obitava u Tenju, predgrađu Osijeka. Završio je Osnovnu školu Tenja te potom upisao I. Gimnaziju u Osijeku. Nakon završene gimnazije upisuje preddiplomski studij računarstva na Fakultetu elektrotehnike, računarstva i informacijskih tehnologija u Osijeku.

Prilozi (na CD-u)

Prilog 1. Dokument i pdf rada

Prilog 2. Programski kod projekta web trgovine za preporuku proizvoda