

**Kamil Filar,**

***Informatyka rok II***

**18.05.2020r.**

# **E-zielarnia**

***Aplikacje internetowe I***

**PHP, Java Script, HTML, CSS**

## Spis treści:

1. Spis treści .....	1
2. Opis i cel projektu .....	2
3. Opis funkcjonalności .....	3
• Navbar .....	4
• Footer.....	5
• Strona główna .....	6
• Panel logowania .....	7
• System logujący .....	8
• Panel rejestracyjny .....	9
• Podgląd konta .....	10
• Edycja konta .....	11
• Koszyk .....	12
• Zamówienia .....	13
• Protokół SMTP .....	14
• Strefa wiedzy .....	15
4. Wymagania do korzystania z aplikacji .....	16

## Opis i cel projektu:

Celem projektu było stworzenie responsywnej witryny, która będzie sklepem internetowym („e-zielarnia”). W projekcie zostały wykorzystane języki programowania takie jak: PHP, Java Script, HTML5 oraz CSS z wykorzystaniem frameworku – Bootstrap, a także język HTML do obsługi bazy danych. Baza danych składa się z 7 tabel, połączonych ze sobą w odpowiedni sposób przy pomocy relacji.

Projekt składa się z osiemnastu plików wchodzących w skład witryny. Są to: „class.phpmailer.php”, „class.smt.php”, „connect.php”, „contact.php”, „index.php”, „knowledge\_zone.php”, „koszyk.php”, „login\_panel.php”, „login.php”, „logout.php”, „PHPMailerAutoload.php”, „polityka\_prywatnosci.php”, „registartion\_panel2.php”, „regulamin.php”, „user\_account.php”, „user\_edit.php”, „welcome.php”, „zamówienia.php”. Plik z bazą danych ma nazwę: „zielarnia.sql”. Projekt umożliwia tworzenie nowych kont dla klientów, składanie zamówień, wysyłanie automatycznie e-maili o zamówieniu dokonany w serwisie e-zielarnia, edycję konta użytkownika. Fundamentem projektu jest szczegółowa walidacja danych wprowadzanych przez użytkownika.

## Navbar:

```
<nav class="navbar navbar-expand-xl bg-dark navbar-dark justify-content-end fixed-top">
  <a class="navbar-brand logo" href="#"></a>
  <button class="navbar-toggler ml-auto mr-1" type="button" data-toggle="collapse" data-target="#navbarSupportedContent">
    <span class="navbar-toggler-icon"></span>
  </button>
  <div class="collapse navbar-collapse flex-grow-0 ml-auto mr-1 mt-1" id="navbarSupportedContent">
    <ul class="navbar-nav text-right">
      <li class="nav-item mt-1 mr-1">
        <a class="nav-link text-center" href="koszyk.php"><i class="fas fa-shopping-cart fa-lg"></i>
        <?php
          error_reporting(0);
          try {
            echo (count($_SESSION["cart"]));
          } catch (Exception $e) {
            echo "0";
          }
          error_reporting(1);
        ?>
        KOSZYK</span></a>
      </li>
      <li class="nav-item mt-1 mr-1">
        <a class="nav-link text-center" href="index.php">SKLEP</a>
      </li>
      <?php
        if (isset($_SESSION['logged'])) {
          echo ' <li class="nav-item mt-1 mr-1" ">
            <a class="nav-link text-center" href="zamowienia.php">MOJE ZAMÓWIENIA</a>
            </li>';
        }
      ?>
      <li class="nav-item mt-1 mr-1">
        <a class="nav-link text-center" href="knowledge_zone.php">STREFA WIEDZY</a>
      </li>
      <li class="nav-item mt-1 mr-1">
        <a class="nav-link text-center" href="contact.php">KONTAKT</a>
      </li>
      <?php
        if (isset($_SESSION['logged'])) {
          echo ' <li class="nav-item mt-1 mr-1" ">
            <a class="nav-link text-center" href="user_account.php">MOJE KONTO</a>
            </li>';
          echo ' <li class="nav-item mt-1 mr-1" ">
            <a class="nav-link text-center" href="logout.php">WYLOGUJ SIĘ</a>
            </li>';
        } else {
          echo ' <li class="nav-item mt-1 mr-1" ">
            <a class="nav-link text-center" href="login_panel.php">ZALOGUJ SIĘ</a>
            </li>';
        }
      ?>
    </ul>
  </div>
</nav>
```

Navbar służy do intuicyjnego poruszania się w projekcie, jest interaktywny, wraz ze scrolowaniem strony przesuwa się wraz ze stroną. W zależności od tego czy użytkownik jest zalogowany czy nie, przybiera różne formy.



Navbar dla niezalogowanego użytkownika.

## Footer:



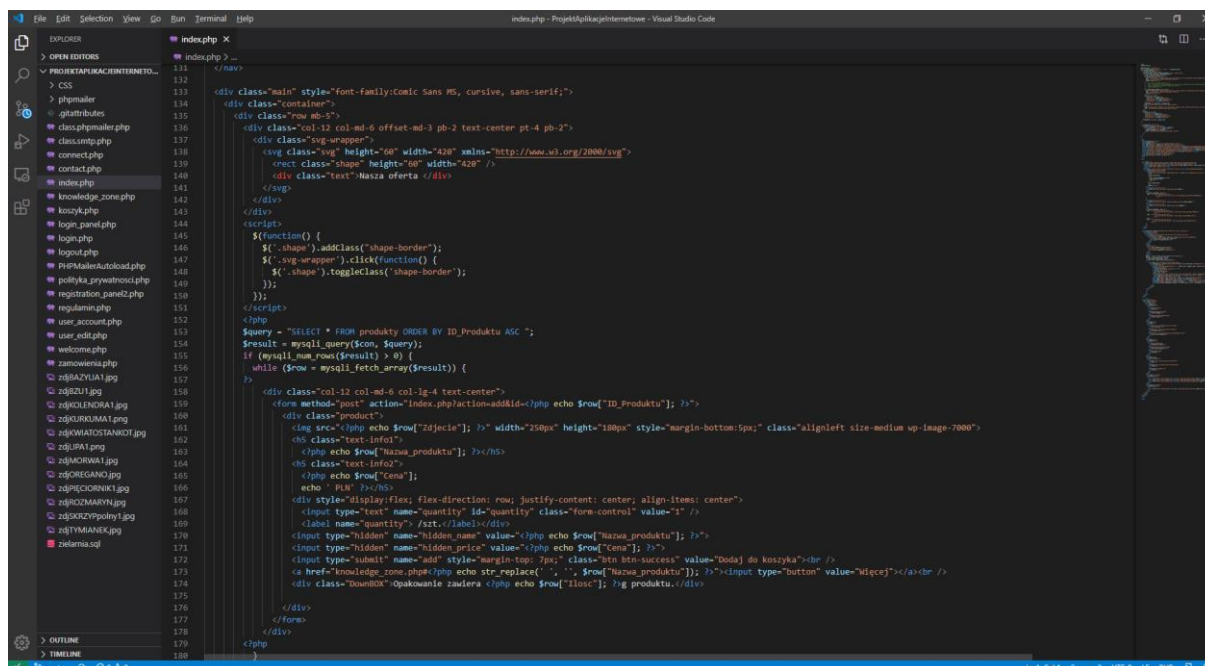
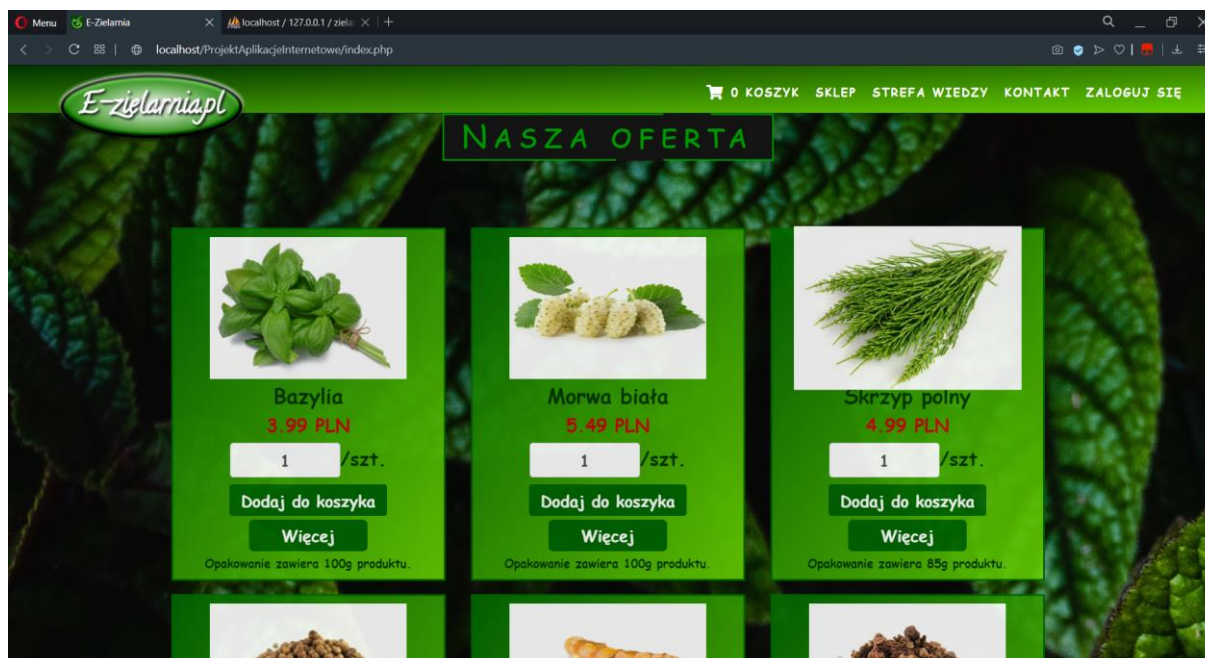
## Wygląd footera

```
<footer class="footer">
  <div class="container">
    <div class="row">
      <div class="col">
        <h1>Nawigacja</h1>
        <ul>
          <a href="index.php">
            <li>Sklep</li>
          </a>
          <a href="knowledge_zone.php">
            <li>Strefa wiedzy</li>
          </a>
          <a href="registration_panel2.php">
            <li>Stwórz konto</li>
          </a>
        </ul>
      </div>
      <div class="col">
        <h1>BLOGI</h1>
        <ul>
          <a href="https://zioladookola.pl/pl_PL/blog/">
            <li>Zioła dookoła</li>
          </a>
          <a href="https://www.pracowniaziol.pl/blog/">
            <li>Pracownia ziół</li>
          </a>
          <a href="http://ziołaojcagrzegorza.pl/przykladowa-strona/">
            <li>Zioła Ojca Grzegorza</li>
          </a>
        </ul>
      </div>
      <div class="col">
        <h1>Wsparcie</h1>
        <ul>
          <a href="regulamin.php">
            <li>Regulamin serwisu</li>
          </a>
          <a href="polityka_prywatnosci.php">
            <li>Polityka prywatności</li>
          </a>
          <a href="contact.php">
            <li>Kontakt</li>
          </a>
        </ul>
      </div>
    </div>
  </div>
</footer>
```

Footer pomaga w szybkim dostępie do różnych podstron oferowanych przez serwis e-zielarnia. Posiada odnośniki do mediów społecznościowych oraz do wyróżnień przyznanej stronie.

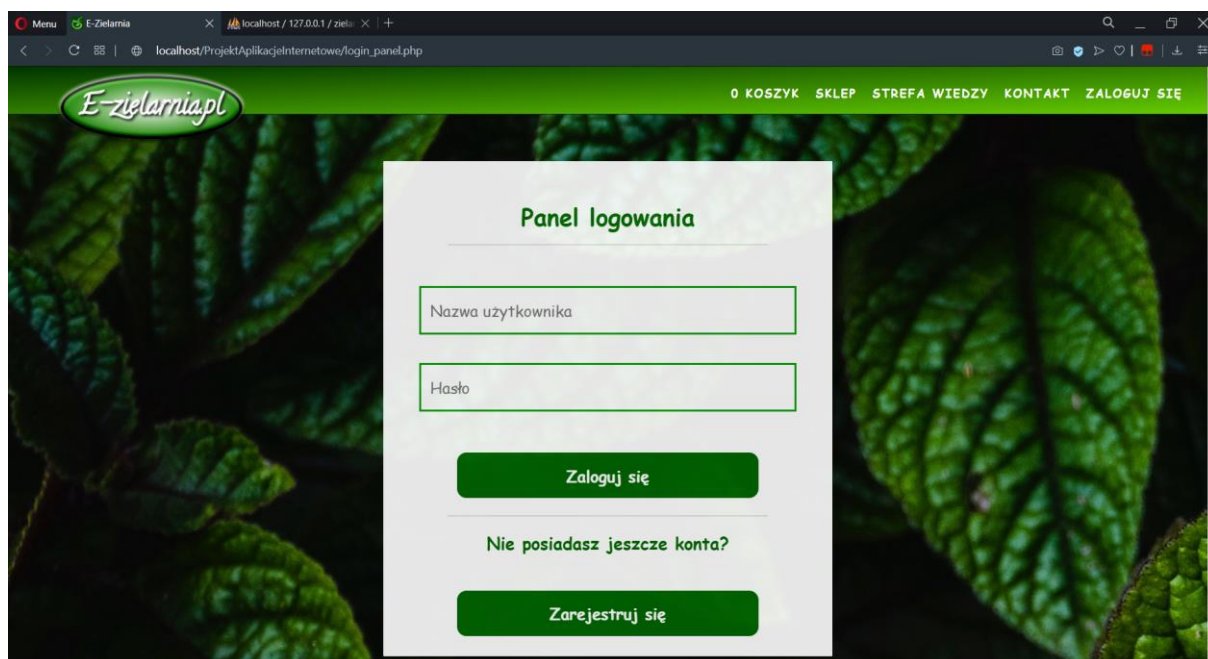
## Strona główna:

Strona główna zawiera się w pliku: „index.php”. Umożliwia dodawanie produktów do koszyka, a także przekierowuje do odpowiedniego opisu danego zioła, które znajduje się w pliku: „knowledge\_zone.php”.



Kod dzięki któremu automatycznie wypisywane są odpowiednie bloki z produktami, które pobierane są z bazy danych: zielarnia.sql.

## Panel logowania:



```
<div class="main" style="font-family:Comic Sans MS, cursive, sans-serif;">
<div class="container">
<div class="row mb-5">
<div class="col-12 col-md-6 offset-md-3 text-center pt-4 pb-2 background">
<form action="login.php" method="post">
<div class="row">
<div class="col-12 py-4">
<div class="textE1">Panel logowania</div>
<div class="links"></div>
</div>
<div class="col-12 py-2"><input type="text" placeholder="Nazwa użytkownika" name="login" /></div>
<div class="col-12 py-2"><input type="password" placeholder="Hasło" name="password" /></div>
<div class="col-12 py-2"><?php
    if (isset($_SESSION['error1'])) {
        echo $_SESSION['error1'];
        unset($_SESSION['error1']);
    }
    ?></div>
<div class="col-12 py-2"><input type="submit" value="Zaloguj się" /></div>
<div class="col-12 py-1">
<div class="links"></div>
</div>
<div class="col-12 py-1">
<div class="gutgut">Nie posiadasz jeszcze konta?</div><br />
</div>
<div class="col-12 py-2"><a href="registration_panel2.php"><input type="button" class="button_active" onclick="" value="Zarejestruj się" /></a></div>
</div>
</form>
</div>
</div>
</div>
</div>
```

Panel logowania umożliwia zalogowanie się do bazy danych przez użytkownika, który posiada konto. W momencie, gdy go nie ma, z poziomu panelu logowania płynnie może przejść do panelu rejestracyjnego. W kodzie pobierane są dwie wartości dla loginu i hasła po czym są wysyłane do pliku: „login.php” gdzie odbywa się sprawdzenie poprawności danych podawanych przy logowaniu.

## System logujący:

System logujący, który sprawdza poprawność danych podawanych przez użytkownika w momencie logowania znajduje się w pliku: „login.php”.

```
if ((strlen($_POST['login']) == 0) && (strlen($_POST['password']) == 0)) {
    $_SESSION['error1'] = '<span div class="error">Nie podano danych!</span>';
    header('Location: login_panel.php');
} else {
    require_once "connect.php";
    $connection = @new mysqli($host, $db_user, $db_password, $db_name);
    if ($connection->connect_errno != 0) {
        echo "Error: " . $connection->connect_errno;
    } else {

        $Userlogin = $_POST['login'];
        $UserPassword = $_POST['password'];

        $UserLogin = htmlentities($Userlogin, ENT_QUOTES, "UTF-8");

        if ($result = @$connection->query(
            sprintf(
                "SELECT * FROM konto_logowanie AS K, konto AS KO, adres AS A WHERE K.ID_Konta=KO.ID_Konta
                AND KO.ID_Adres = A.ID_Adres AND K.UserLogin='%s'",
                mysqli_real_escape_string($connection, $UserLogin)
            )
        )) {
            $Number_of_users = $result->num_rows;
            if ($Number_of_users > 0) {
                $row = $result->fetch_assoc();
                if (password_verify($UserPassword, $row['Haslo'])) {

                    $_SESSION['logged'] = true;

                    $_SESSION['ID_user'] = $row['ID_Konta'];
                    $_SESSION['ID_ad'] = $row['ID_Adres'];
                    $_SESSION['H'] = $row['Haslo'];
                    $_SESSION['UserLogin'] = $row['UserLogin'];
                    $_SESSION['Imie'] = $row['Imie'];
                    $_SESSION['Nazwisko'] = $row['Nazwisko'];
                    $_SESSION['Numer_Telefonu'] = $row['Numer_Telefonu'];
                    $_SESSION['Miejscowosc'] = $row['Miejscowosc'];
                    $_SESSION['Ulica'] = $row['Ulica'];
                    $_SESSION['Nr_Lokalu'] = $row['Nr_Lokalu'];
                    $_SESSION['Kod_Pocztowy'] = $row['Kod_Pocztowy'];
                    $_SESSION['Nr_Domu'] = $row['Nr_Domu'];
                    $_SESSION['Mail'] = $row['Mail'];

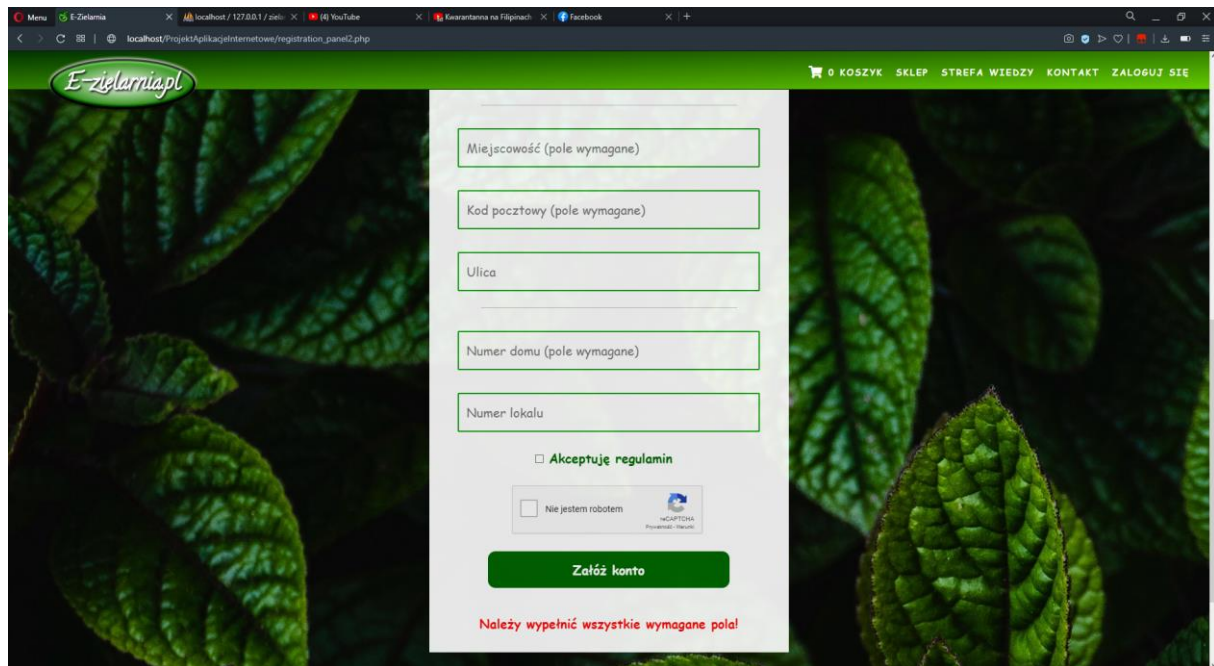
                    unset($_SESSION['error1']);
                    $result->free_result();
                    header('Location: index.php');
                }
            }
        }
    }
}
```

W tym pliku zostaje nawiązane połączenie z bazą danych. W momencie kiedy wszystkie dane podane przez użytkownika są poprawne do sesji trafiają jego dane – dzięki, którym strona wie na jakich danych mają wykonywać się pozostałe operacje. W momencie błędnych danych zadziała ta część kodu:

```
} else {
    $_SESSION['error1'] = '<span div class="error">Podano błędne dane!</span>';
    header('Location: login_panel.php');
```



## Panel rejestracyjny:

The image shows a web browser window displaying a registration form for 'E-zielarnia.pl'. The form is centered on a green background with a leaf pattern. It includes input fields for 'Miejscowość (pole wymagane)', 'Kod pocztowy (pole wymagane)', 'Ulica', 'Numer domu (pole wymagane)', and 'Numer lokalu'. Below these fields are checkboxes for 'Akceptuję regulamin' and 'Nie jestem robotem' (with a reCAPTCHA logo). A green 'Załącz konto' button is at the bottom. A red error message at the bottom reads 'Należy wypełnić wszystkie wymagane pola!'. The browser's address bar shows 'localhost/ProjektAplikacjiInternetowej/registration\_panel2.php'.

Na zaprezentowanym wyżej zdjęciu znajduje się część panelu rejestracyjnego, który pozwala na utworzenie nowego konta użytkownikowi po pomyślnym przejściu walidacji danych podawanych w odpowiednich polach. Do zabezpieczeń została wykorzystana również captcha.

```
<!-- Recaptcha -->
<div class="g-recaptcha" data-sitekey="6Lc8IuwUAAAAAObb41R83IoY9GHd1eW4TUZQy2NU"></div>
<?php
if (isset($_SESSION['error_bot'])) {
    echo '<div class="error">' . $_SESSION['error_bot'] . '</div>';
    unset($_SESSION['error_bot']);
}
?>
```

```
//Bot verified

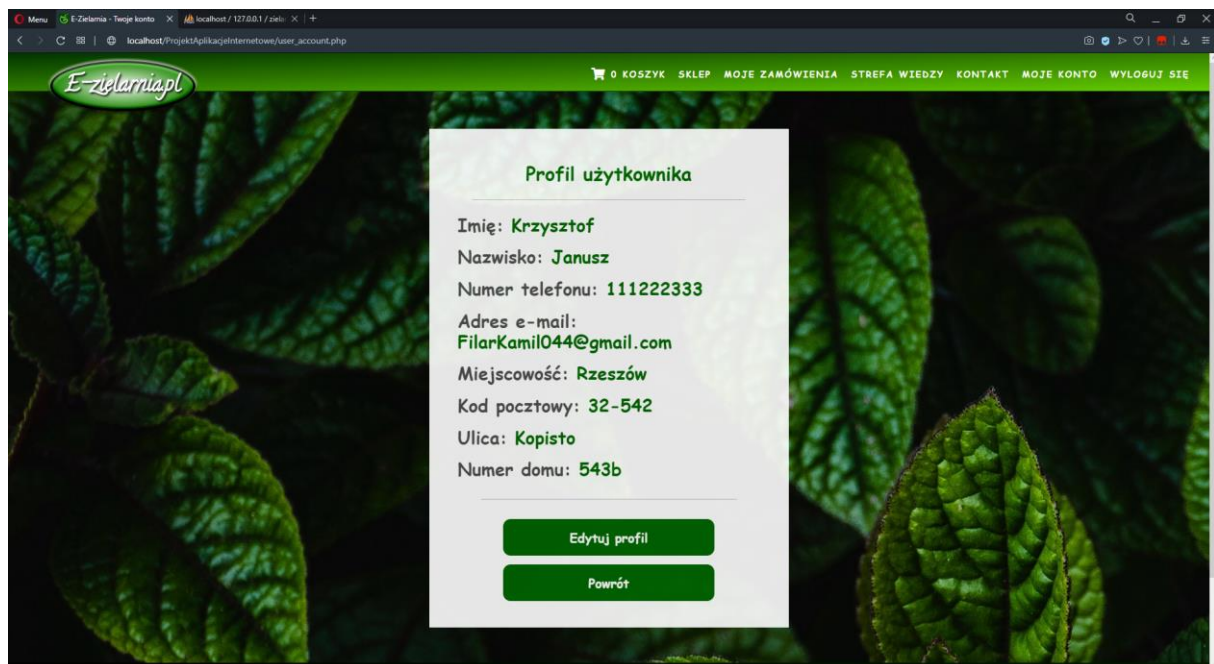
$s = "6Lc8IuwUAAAAAObb41R83IoY9GHd1eW4TUZQy2NU";

$check = file_get_contents('https://www.google.com/recaptcha/api/siteverify?secret=' . $s . '&response=' . $_POST['g-recaptcha-response']);

$answer = json_decode($check);
if ($answer->success == false) {
    $all_ok = false;
    $_SESSION['error_bot'] = "Potwierdź, że nie jesteś botem!";
}
```

Przykład kodu w którym jest zaprezentowana walidacja danych na podstawie captcha. W momencie, kiedy użytkownik nie przejdzie jej pomyślnie zostanie zwrócona odpowiednia wiadomość: „Potwierdź, że nie jesteś botem!”.

## Podgląd konta:



Powyżej znajduje się podgląd na profil użytkownika. Dzięki przekazywaniu danych przez obiekty znajdujące się w sesji witryna rozpoznaje jakie dane powinny zostać wyświetlone. Poniżej zaprezentowany jest kod, który to wykonuje:

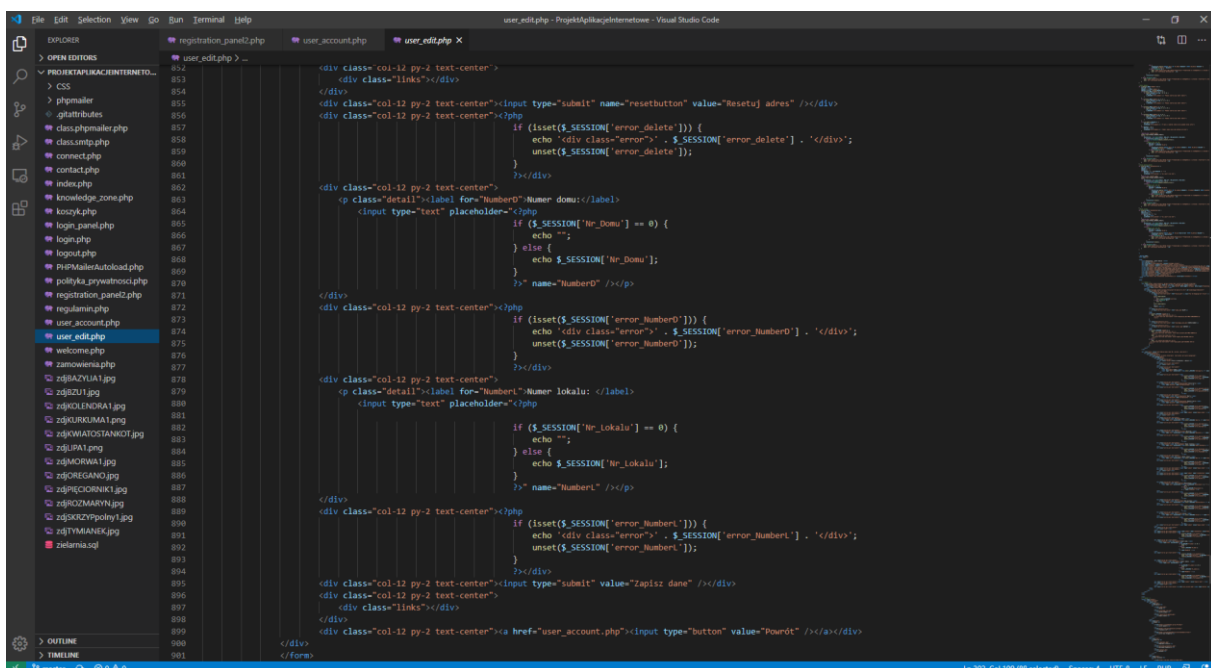
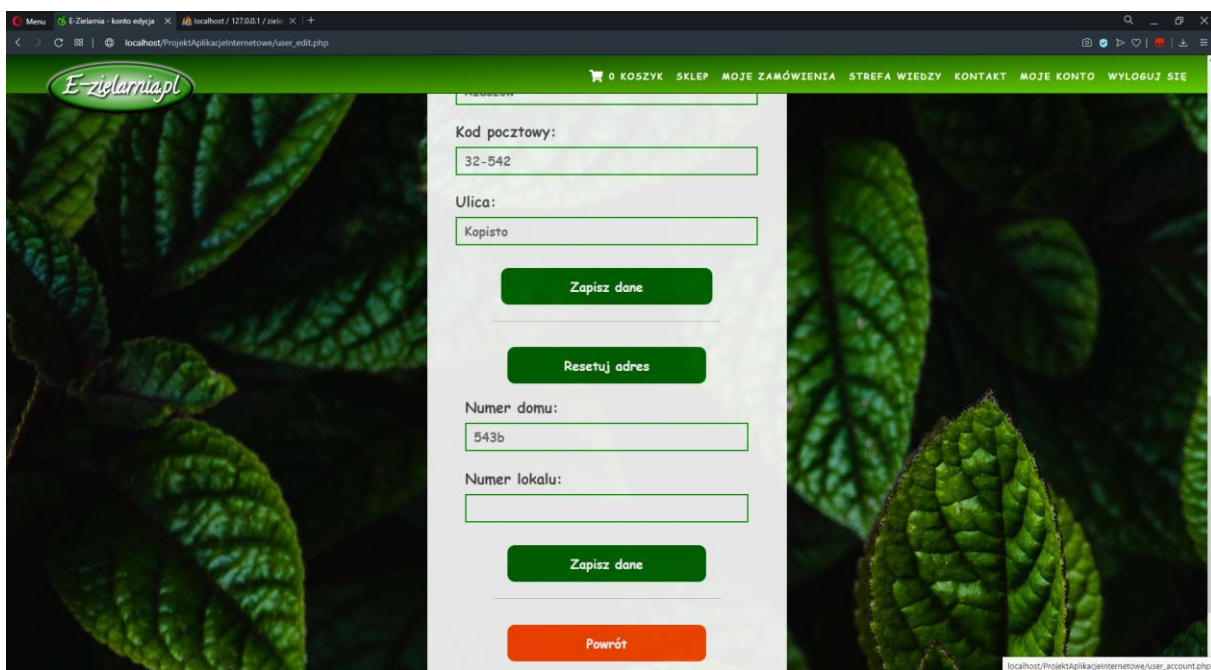
```
<?php
if ($_SESSION['Ulica'] == "") {

    echo '<div class="col-12 py-2"><h3>
    <p class="detail">Ulica: <span class="name">---</span></p></h3></div>';
} else {
    echo '<div class="col-12 py-2"><h3>
    <p class="detail">Ulica: <span class="name">' . $_SESSION['Ulica'] . "</span><p></h3></div>";
}
}
if ($_SESSION['Nr_Domu'] == "") {

    echo '<div class="col-12 py-2"><h3>
    <p class="detail">Numer domu: <span class="name">---</span></p></h3></div>';
    echo '<div class="col-12 py-2"><h3>
    <p class="detail">Numer lokalu: <span class="name">---</span></p></h3></div>';
} else {
    echo '<div class="col-12 py-2"><h3>
    <p class="detail">Numer domu: <span class="name">' . $_SESSION['Nr_Domu'] . "</span><p></h3></div>";
}
}
if ($_SESSION['Nr_Lokalu'] != "") {
    echo '<div class="col-12 py-2"><h3>
    <p class="detail">Numer lokalu: <span class="name">' . $_SESSION['Nr_Lokalu'] . "</span></p></h3></div>";
}
}
?>
```

## Edycja konta:

Plik z edycją konta ma ok. 1000 linii kodowych ponieważ tutaj jest potężny system walidujący dane przy zachowaniu intuicyjnego menu. W każdym momencie użytkownik może zresetować jego dane bez konieczności ponownego uzupełniania ich.

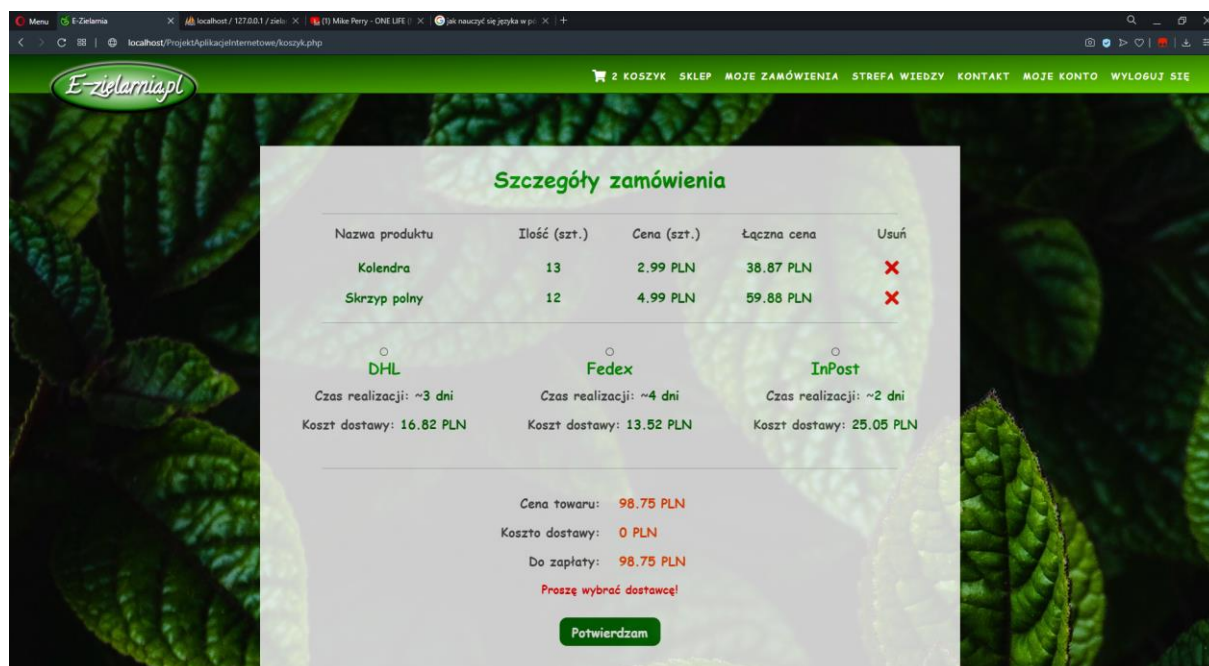


```
853 <div class="col-12 py-2 text-center">
854 <div class="links"></div>
855 </div>
856 <div class="col-12 py-2 text-center"><input type="submit" name="resetbutton" value="Resetuj adres" /></div>
857 <div class="col-12 py-2 text-center"></div>
858
859 if (isset($_SESSION['error_delete'])) {
860     echo <div class="error"> . $_SESSION['error_delete'] . </div>;
861     unset($_SESSION['error_delete']);
862 }
863 </div>
864
865 <div class="col-12 py-2 text-center">
866 <p class="detail"><label for="NumerD" Numer domu: </label>
867 <input type="text" placeholder="</php
868
869 if ($_SESSION['Nr_Domu'] == 0) {
870     echo "";
871 } else {
872     echo $_SESSION['Nr_Domu'];
873 }
874 >> name="NumerD" /></p>
875 </div>
876
877 <div class="col-12 py-2 text-center"></div>
878
879 if (isset($_SESSION['error_NumerD'])) {
880     echo <div class="error"> . $_SESSION['error_NumerD'] . </div>;
881     unset($_SESSION['error_NumerD']);
882 }
883 </div>
884
885 <div class="col-12 py-2 text-center">
886 <p class="detail"><label for="NumerL" Numer lokalu: </label>
887 <input type="text" placeholder="</php
888
889 if ($_SESSION['Nr_Lokalu'] == 0) {
890     echo "";
891 } else {
892     echo $_SESSION['Nr_Lokalu'];
893 }
894 >> name="NumerL" /></p>
895 </div>
896
897 <div class="col-12 py-2 text-center"></div>
898
899 if (isset($_SESSION['error_NumerL'])) {
900     echo <div class="error"> . $_SESSION['error_NumerL'] . </div>;
901     unset($_SESSION['error_NumerL']);
902 }
903 </div>
904
905 <div class="col-12 py-2 text-center"><input type="submit" value="Zapisz dane" /></div>
906 <div class="col-12 py-2 text-center">
907 <div class="links"></div>
908 </div>
909 <div class="col-12 py-2 text-center"><a href="user_account.php"><input type="button" value="Powrót" /></a></div>
910 </div>
911 </form>
```

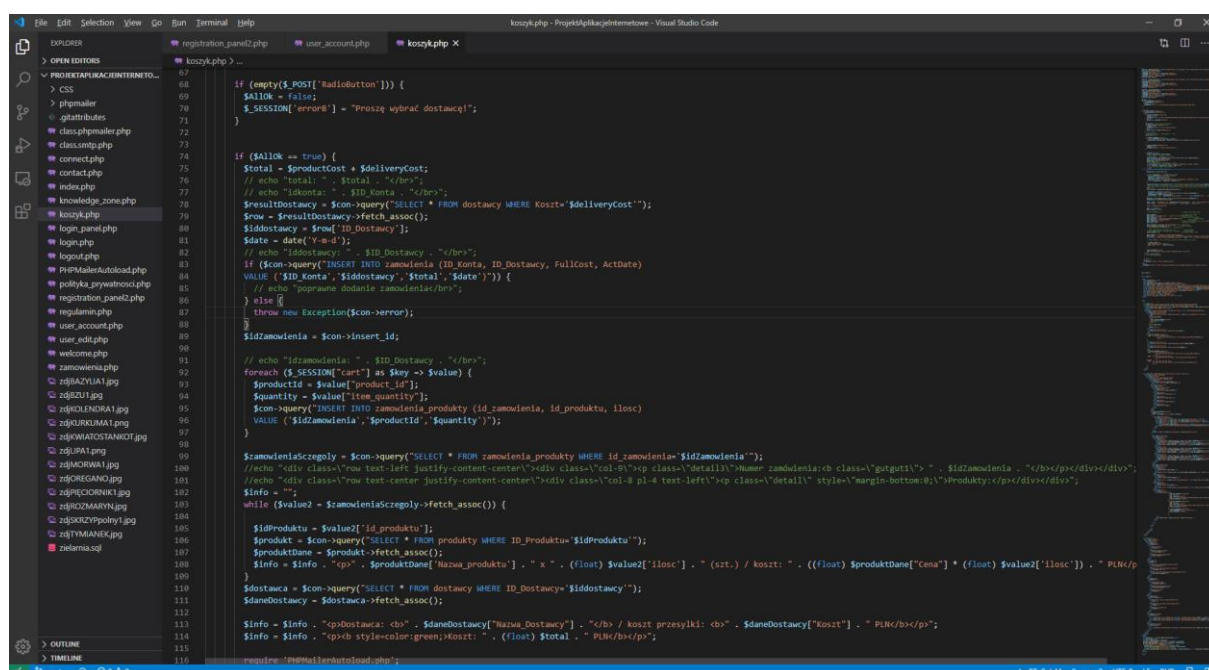
Fragment kodu reprezentujący część zamieszczoną na powyższym zdjęciu



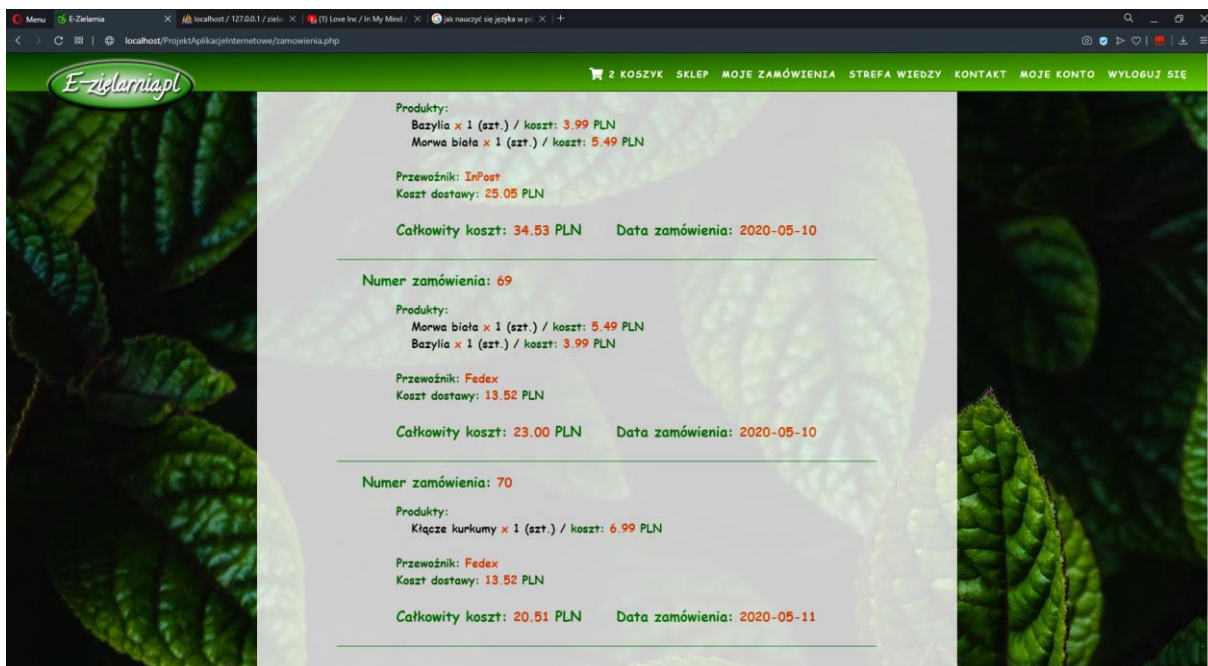
## Koszyk:



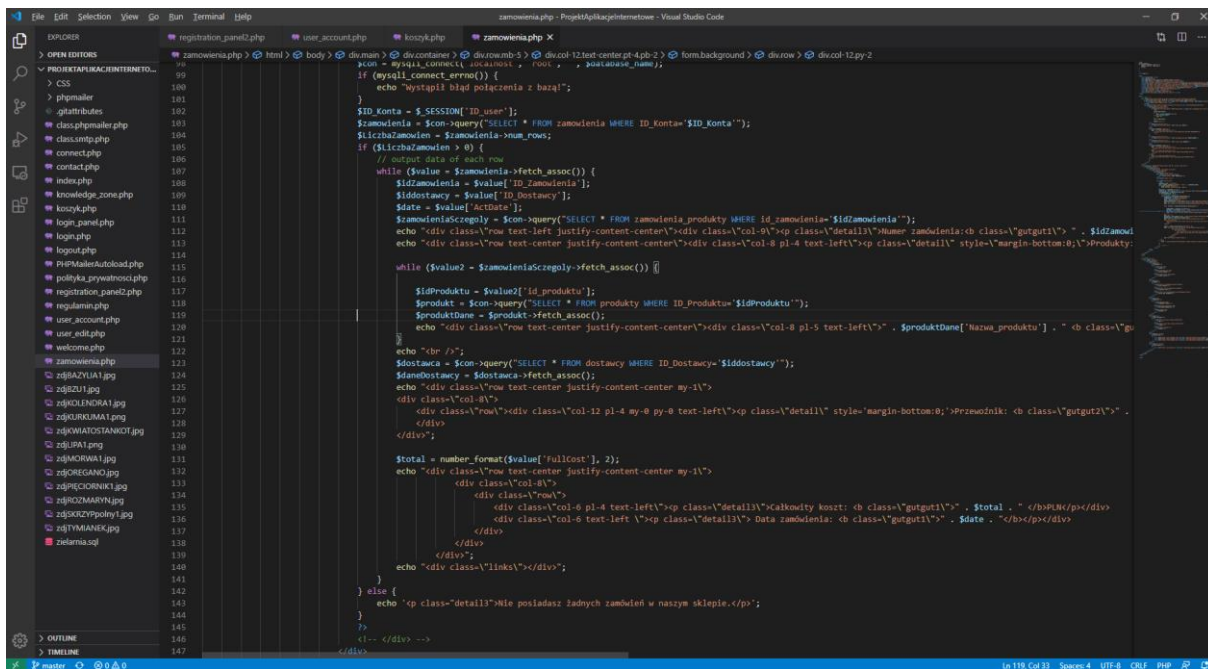
Koszy jest to miejsce, gdzie użytkownik ma podgląd na swoje zamówienie. Jest to miejsce w którym powinien wybrać dostawcę. Może jednocześnie dokonywać modyfikacji produktów w nim zawartych. Po kliknięciu przycisku „Potwierdzam” zostanie przekierowany do listy swoich zamówień, a także zostanie powiadomiony automatycznie e-mailem o złożonym zamówieniu wraz ze szczegółami. Fragment kodu:



## Zamówienia:

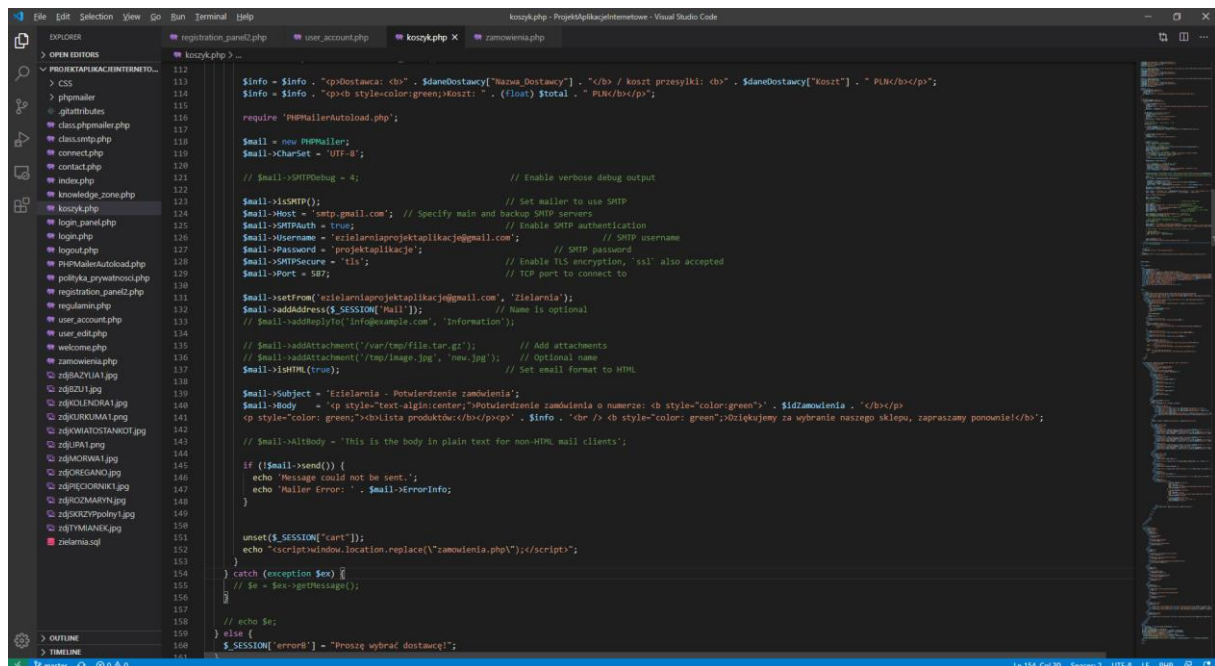


Jest to miejsce, gdzie są wypisane wszystkie zamówienia użytkownika, aby mógł sprawdzić swoją historię co kiedyś kupował.



## Protokół SMTP:

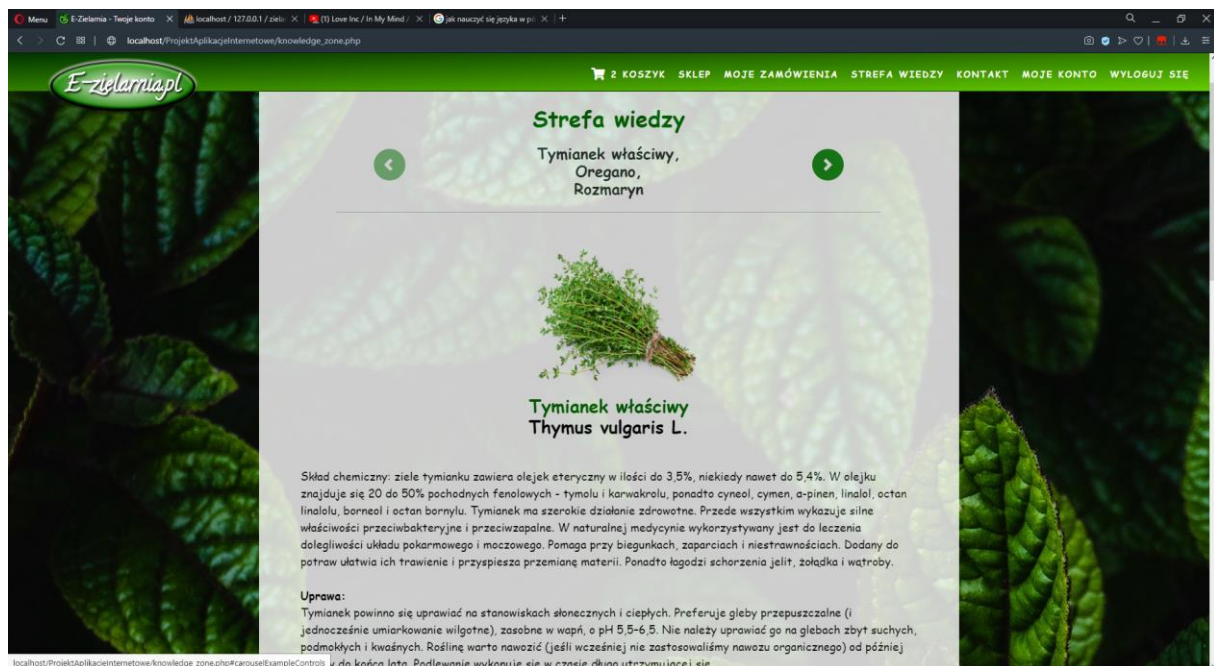
Jest to protokół, który służy do wysłania automatycznie wiadomości e-mail do użytkowników o złożonym zamówieniu. Składa się z trzech plików, są to: „class.phpmailer.php, class.smtp.php, PHPMailerAutoload.php”. Wywołany jest w pliku: „koszyk.php” w momencie potwierdzenia zamówienia. Na załączonym obrazku widać treść wiadomości, którą jest przesyłana do użytkownika w momencie złożenia zamówienia.



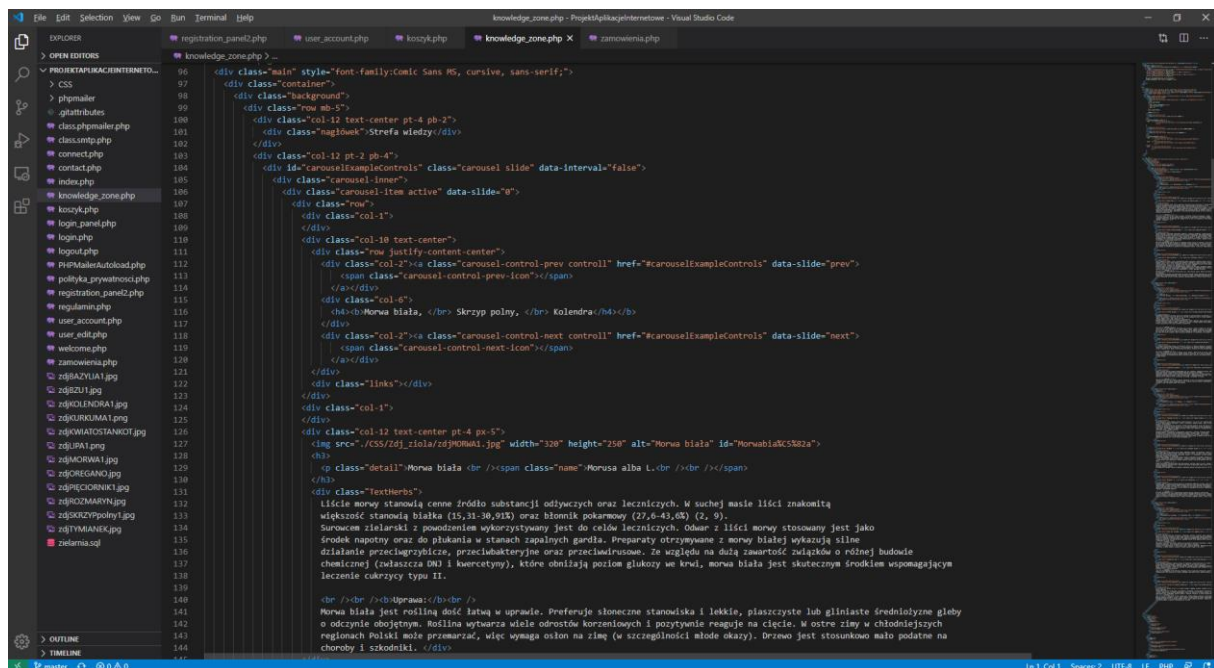
```
112 $info = $info . "<p>dostawca: <b>". $daneDostawcy["nazwa_dostawcy"] . "</b> / koszt przesyłki: <b>". $daneDostawcy["koszt"] . " PLN</b></p>";
113 $info = $info . "<p><b style='color: green;'>Koszt: " . (float) $total . " PLN</b></p>";
114
115
116 require 'PHPMailerAutoload.php';
117
118 $mail = new PHPMailer();
119 $mail->CharSet = 'UTF-8';
120
121 // $mail->SMTPDebug = 4; // Enable verbose debug output
122
123 $mail->isSMTP(); // Set mailer to use SMTP
124 $mail->Host = 'smtp.gmail.com'; // Specify main and backup SMTP servers
125 $mail->SMTPAuth = true; // Enable SMTP authentication
126 $mail->Username = 'zielarniaprojektaplikacje@gmail.com'; // SMTP username
127 $mail->Password = 'projektaplikacje'; // SMTP password
128 $mail->SMTPSecure = 'tls'; // Enable TLS encryption, 'ssl' also accepted
129 $mail->Port = 587; // TCP port to connect to
130
131 $mail->setFrom('zielarniaprojektaplikacje@gmail.com', 'Zielarnia');
132 $mail->addAddress($SESSION['Mail']); // Name is optional
133 // $mail->addReplyTo('info@example.com', 'Information');
134
135 // $mail->addAttachment('user/tmp/file.tar.gz'); // Add attachments
136 // $mail->addAttachment('tmp/image.jpg', 'new.jpg'); // Optional name
137 $mail->isHTML(true); // Set email format to HTML
138
139 $mail->Subject = 'Zielarnia - Potwierdzenie zamówienia';
140 $mail->Body = "<p style='text-align: center;'>Potwierdzenie zamówienia o numerze: <b style='color: green;'>". $idZamowienia . "</b></p>";
141 $mail->Body .= "<p style='color: green;'><b>Lista produktów:</b></p></p>" . $info . "<br /> <b style='color: green;'>Dziękujemy za wybranie naszego sklepu, zapraszamy ponownie</b>";
142
143 // $mail->AltBody = "This is the body in plain text for non-HTML mail clients";
144
145 if (!$mail->send()) {
146     echo "Message could not be sent.";
147     echo "Mailer Error: " . $mail->ErrorInfo;
148 }
149
150 unset($SESSION['cart']);
151 echo "<script>window.location.replace('zamowienia.php');</script>";
152
153 }
154 catch (Exception $ex) {
155     // $e = $ex->getMessage();
156 }
157
158 // echo $e;
159 } else {
160     $SESSION['error0'] = "Proszę wybrać dostawcę!";
161 }
```



## Strefa wiedzy:



Strefa wiedzy jest to miejsce, gdzie użytkownik może dowiedzieć się więcej o produktach oferowanych w sklepie. Został stworzony w oparciu o funkcjonalność bootstrap-carousel.



## Wymagania do korzystania z aplikacji:

Na samym początku należy zaimportować plik z bazą danych (zielarnia.sql) do programu PHPMyAdmin. Następnie należy uruchomić plik index.php używając do tego odpowiedniej ścieżki w zależności, gdzie będzie przechowywany projekt. Kolejnym krokiem będzie poprawne założenie konta w serwisie. Następnie można swobodnie korzystać z aplikacji. Aplikacja bez konta również działa jednak jest nieco bardziej okrojona – nie można wówczas skorzystać z wyżej wymienionych funkcjonalności.