

Bazaar – php example code – part 8 – shopping cart of user

Article focus on way of implementation of user shopping cart. Our `bazar_item` table is altered for new field `cart_number`. If cart number is zero, item is not added into a buyer cart. After adding item into a cart, this number is changed to a number of buying user. Showing content of cart relays on display `bazaar_itmes` marked by `users_id` of appropriate user.

Way how to implement shopping cart

Shopping cart will display all items added by appropriate user for further revision and commitment to buy.

In our cart page we must be able to do:

- display all items added to buy by a single user
- enable delete items from cart
- calc total price for items in cart
- delivery adress can be updated by `editinfo.php` page, here is only displayed content
- after definitive submitting, send info to seler by e-mail about succesfull buy

Cart is available only for loged in users, also loged in user can commit buy to a seler/ sellers of selected items.

For further improvement or consideration is way how to inform sellers about succesull buy. Our simle solution send one e-mail for one item. But there is not a small possibility, tahat

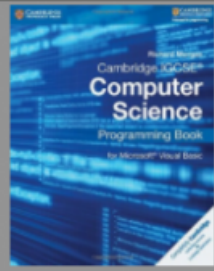
cart can contain more items from one seler. For all of these items is send to seler one buy commitment request.

Visual look of shopping cart

Next pictures show content of our shopping cart and related scripts output.



Cart item of user admin with id - 10 are:

Name	Price	Category	Screenshot1	More info
Computer science - used - good quality	55 €	books/computer		Remove from cart

Total price for items in cart: 55 €

Your delivery adress is:

Please check your contact and delivery info, these information are important for seller of the items for correct contact and delivery!

e-mail:	First name:	Last name:
admin@admin.sk	Firstname example	Lastname
Adress in form - Street Nr.:		
Street 678		
City:		
City 17		
ZIP code in form XXXXX:		
00002		

If any of displayed info need correction, please visit your profile page [here](#).

For confirmation of buy select YES and click on red button below:

☐ Yes
☒ No

I CONFIRM THE PURCHASE WITH THE OBLIGATION TO PAY

Shopping cart frontend

The selected operation cannot be performed. Please select YES for further buy confirmation.

Bazaar - best items for a best prices! Edit profile Logout admin
Manage your page



Cart item of user admin with id - 10 are:

Name	Price	Category	Screenshot1	More info
Computer science - used - good quality	55 €	books/computer		>> Remove from cart

Total price for items in cart: 55 €

Your delivery adress is:

Please check your contact and delivery info, these information are important for seller of the items for correct contact and delivery!

e-mail:	First name:	Last name:
admin@admin.sk	Firstname example	Lastname
Address in form - Street Nr.:		
Street 678		
City:		
City 17		
ZIP code in form XXXXX:		
00002		

If any of displayed info need correction, please visit your profile page [here](#).

For confirmation of buy select YES and click on red button bellow:

☐ Yes
☒ No

I CONFIRM THE PURCHASE WITH THE OBLIGATION TO PAY

Displayed info about not selected comitting YES for BUY

Your seller of Garden scissors - green was successfully contacted via e-mail.
Your seller of Garden scissors - green small was successfully contacted via e-mail.
Your seller of Computer science - used - good quality was successfully contacted via e-mail.



Cart item of user admin with id - 10 are:

Name	Price	Category	Screenshot1	More Info
Garden scissors - green	9.50 €	garden/scissors		» Remove from cart
Garden scissors - green small	7.50 €	garden/scissors		» Remove from cart
Computer science - used - good quality	55 €	books/computer		» Remove from cart

Total price for items in cart: 72 €

Your delivery adress is:

Please check your contact and delivery info, these information are important for seller of the items for correct contact and delivery!

e-mail:	First name:	Last name:
admin@admin.sk	Firstname example	Lastname
Address In form - Street Nr.:		
Street 678		
City:		
City 17		
ZIP code in form XXXXX:		
00002		

If any of displayed info need correction, please visit your profile page [here](#).

For confirmation of buy select YES and click on red button below:

☐ Yes
☒ No

I CONFIRM THE PURCHASE WITH THE OBLIGATION TO PAY

Successfully submitted buy with notification about contacting of sellers for that items

Implementation of shopping cart

Next code shows how shopping cart is implemented in our application.

```
<!--
*****
*** -->
<!-- PHP „self“ code showing content of items added into a
cart -->
<!--
*****
*** -->
<!-- Vrsion: 1.0          Date: 1.11.2020 by CDesigner.eu
-->

<!--
*****
*** -->
<!-- ***** MEMO – base is from index.php – show all
items with cart_number = session(users_id) + calc total summ,
create remove from cart link with removefromcart.php show
address for delivery and button submitt to buy
***** -->
<?php
    require_once(,appvars.php'); // including variables for da
tabase
    session_start(); // start the session – must be added on a
ll pages for session variable accessing
    // solution using SESSIONS with COOKIES for longer (30days
) login persistency

    if(!isset($_SESSION[,users_id'])) { // if session is no mo
re active
        if(isset($_COOKIE[,users_id']) && isset($_COOKIE[,user
name'])) { // but cookie is set then renew session variables a
```

```

long them
    $_SESSION[,users_id'] = $_COOKIE[,users_id'];
    $_SESSION[,username'] = $_COOKIE[,username'];
    $_SESSION[,user_role'] = $_COOKIE[,user_role']; //
added for role
    }
}
// two variables for message and styling of the message with
bootstrap
$msg = "";
$msgClass = "";
// default values of auxiliary variables
$name_of_item = "";
$price_eur = "";
$subcategory_id = "";
$users_id = "";
$item_add_date = "";
$subcategory_id = "";
$published = false;
$screenshot1 = "";
$screenshot2 = "";
$screenshot3 = "";
$item_description = "";
$is_result = false; //before hitting submit button no result
is available

// Control if data was submitted
if(filter_has_var(INPUT_POST, 'submit')) {
    // Data obtained from $_postmessage are assigned to local
    variables
    if($_POST[,confirm'] == ,Yes, ){ // if user selected
YES and hit Buy button on below of the page
        //read all data from $_POST array
        $users_id = htmlspecialchars($_POST[,users_id']);
        /*****
*****
        *    obtain data about buyer
        */
        // read data about buying user with users id from
database
        // make database connection

```

```

        $dbc = mysqli_connect(DB_HOST, DB_USER, DB_PW, DB_
NAME);

        // Check connection
        if($dbc === false){
            die („ERROR: Could not connect to database.
“ . mysqli_connect_error());
        }

        //– only one needed – getting users data for purcha
se e-mail
        $sql = „SELECT * FROM bazaar_user WHERE users_id =
„.“, $users_id'“.“LIMIT 1” ;
        if($output = mysqli_query($dbc, $sql)){
            if(mysqli_num_rows($output) > 0){ // if any r
ecord obtained from SELECT query

                while($row = mysqli_fetch_array($output)){
                    //next rows outputed in while loop
                    $first_name_buyer = $row[,first_na
me'];
                    $lastname_name_buyer = $row[,lastn
ame_name'];
                    $addressss_buyer = $row[,addresss']
;

                    $city_buyer = $row[,city'];
                    $ZIPcode_buyer = $row[,ZIPcode'];
                    $email_buyer = $row[,email'];

                }

                // Free result set
                mysqli_free_result($output);
            } else{
                echo „No info about buyer obtained.“; // i
f no records in table
            }
        } else{
            echo „ERROR: Could not able to execute $sql. “
. mysqli_error($dbc); // if database query problem
        };
    }
}

```



```

/*****
*****
        *    obtain data bought items from this buyer with
users_id defined by current SESSION
        */
        //get info about sold items – we must go through a
ll bought items and send email one by one for all different sel
ers of item (first approach for all item one)
        $sql = „SELECT * FROM bazaar_item WHERE cart_numbe
r = „.“, $users_id'“ ;
        if($output = mysqli_query($dbc, $sql)){
            if(mysqli_num_rows($output) > 0){ // if any r
ecord obtained from SELECT query

                while($row = mysqli_fetch_array($output)){
//next rows outputed in while loop
                    $item_id = $row[,item_id'];
                    $name_of_item = $row[,name_of_item
'];

                    $price_eur = $row[,price_eur'];
                    $users_id_of_seller = $row[,users_
id'];

/*****
*****
                *    sent info to seller item by it
em in the buyer cart
                */
                    // send appropriate e-
mails about buy items by items
                    // validate e-mail
                    if(filter_var($email_buyer, FILTER
_VALIDATE_EMAIL) === false){
                        // E-mail is not valid
                        $msg = ,Wrong e-
mail format of buyer, purchase can not be created. Please cont
act page admin.';

                        $msgClass = ,alert-danger';
                    } else {
                        // E-mail is ok

```

```

$is_result = true;
/* request e-mail of seller */
/*****

*****

* obtain e-mail of
appropriate seller – this is done for all buying items one by
one

*/
$sql2 = „SELECT email FROM baz
aar_user WHERE users_id = „.$users_id_of_seller'“ ;
if($output2 = mysqli_query($db
c, $sql2)){
    if(mysqli_num_rows($output
2) > 0){ // if any record obtained from SELECT query

        while($row = mysqli_fe
tch_array($output2)){ //next rows outputed in while loop
            $email_of_seller =
$row[,email'];

        }

        // Free result set
        mysqli_free_result($ou
tput2);

    } else{
        echo „No email about s
eller can be obtained.“; // if no records in table
    }
} else{
    echo „ERROR: Could not abl
e to execute $sql2. “ . mysqli_error($dbc); // if database que
ry problem
};
/*****

*****

* construct information e-mails
about item buy one by one for all items in cart

*/

```

```

        $toEmail = $email_of_seller; /
/!!! e-mail address to send to
        $subject = ,Item ,.$name_of_it
em.' purchased on Bazaar by ,.$first_name_buyer.' ,.$lastname_
name_buyer;
        $body = ,<h2>Item ,.$name_of_i
tem.' was succesfully purchased by : ,.$first_name_buyer.' ,.$
lastname_name_buyer.'</h2>
        <h4>Delivery adress for th
is purchase is: </h4><p>'.$addresss_buyer.',</p><p> ,.$city_bu
yer.', </p><p>'.$ZIPcode_buyer.'</p>
        <h4>Email</h4><p>E-
mail of buyer is'.$email_buyer.' this e-
mail can be used for further communication.</p>
        <h4>Selling price was:</h4>
><p'.$price_eur.' €.</p>
        ,;
        // Email Headers
        $headers = „MIME-
Version: 1.0” .“\r\n”;
        $headers .=“Content-
Type:text/html;charset=UTF-8” . „\r\n”;
        // Additional Headers
        $headers .= „From: “ . $first_n
ame_buyer. „<„.$email_buyer.”>“ . „\r\n”;

        if(mail($toEmail, $subject, $b
ody, $headers)){
                // Email Sent
                $msg .= ,<p> Your seller o
f ,.$name_of_item.' was successfully contacted via e-
mail.</p>';
                $msgClass = ,alert-
success';
        } else {
                // Failed
                $msg = ,Information about
your buy cannot be delivered to seller via e-
mail. Please contact site admin for further help.';
                $msgClass = ,alert-

```

```

danger';

    }

}

}

// Free result set
mysqli_free_result($output
);

    } else{
        echo „No info about buyer
obtained.“; // if no records in table
    }
} else{
    echo „ERROR: Could not able to
execute $sql. “ . mysqli_error($dbc); // if database query pr
oblem
};

// close database connection
mysqli_close($dbc);

} else {
    echo „<p class=“alert alert-
danger“ > The selected operation cannot be performed. Please s
elect YES for further buy confirmation. </p>“;
}

};

?>
<!-- ***** ->
<!-- HTML code containing Form for submitting ->
<!-- ***** ->

```

```

<!DOCTYPE html>
<html>
<head>
    <title> Bazaar Cart </title>
    <link rel="stylesheet" href="./css/bootstrap.min.css"> <!--
bootstrap mini.css file ->
    <link rel="stylesheet" href="./css/style.css"> <!-- my local
css file ->
    <script src="https://code.jquery.com/jquery-3.1.1.slim.min
.js" integrity="sha384-
A7FZj7v+d/sdmMqp/n0QwliLvUsJfDHW+k90mg/a/EheAdgtzNs3hpfag6Ed95
0n" crossorigin="anonymous"></script>
    <script src="https://cdnjs.cloudflare.com/ajax/libs/tether/1.4.0/js/tether.min.js" integrity="sha384-
DztdAPBWPRXSA/3eYEEUWrWCy7G5KFbe8fFjk5JAIxUYHKkDx6Qin1DkWx51bB
rb" crossorigin="anonymous"></script>

</head>
<body>
    <nav class="navbar navbar-default">
        <div class="container">
            <div class="navbar-header">
                <?php // generate menu if user is logged in or not
                // old solution with cookies if(isset($_COOKIE['username'])) { // logged in user
                if(isset($_SESSION['username'])) { // logged in user
                    echo '<a class="navbar-brand" href="index.php">Bazaar – best items for a best prices!
</a>';
                    echo '<a class="navbar-brand" href="editprofile.php"> Edit profile </a>';
                    echo '<a class="navbar-brand" href="logout.php"> Logout , $_SESSION['username'] . '</a>';
                    if(isset($_SESSION['user_role'])=='admin') { //
                    // if logged user is admin role
                        echo '<a class="navbar-brand" href="admin.php"> Manage your page </a>';
                    };
                    require_once('sell_icon.php'); // graphic menu

```

item for selling your items

```
require_once('cart_icon.php'); // small cart icon in menu
```

```
    } else { // visitor without login
        echo '<a class="navbar-brand" href="login.php"> Log In </a>';
        echo '<a class="navbar-brand" href="signup.php"> Sign Up for better membership! </a>';
    }
```

```
        echo '<a class="navbar-brand" href="index.php">Bazaar – best items for a best prices! </a>';
    }
```

```
?>
```

```
</div>
```

```
</div>
```

```
</nav>
```

```
<div class="container" id="container_1060">
```

```
<?php if($msg != ""): ?>
```

```
    <div class="alert <?php echo $msgClass; ?>"><?php echo $msg; ?></div>
```

```
<?php endif; ?>
```

```
<br>
```

```

```

```
<br>
```

```
<h4> Cart item of user
```

```
<?php echo $_SESSION['username']; // creating title of cart for users
```

```
    echo " with id – {$_SESSION['users_id']} are :";
```

```
?>
```

```
<br>
```

```
</h4>
```

```
<!-- Showing content of the cart of appropriate user with items marked with users_id in filed cart_number ->
```

```

<?php
    /* Attempt MySQL server connection. Assuming you are running MySQL
    server with default setting (user 'root' with no password) */
    $dbc = mysqli_connect(DB_HOST, DB_USER, DB_PW, DB_NAME);

    // Check connection
    if($dbc === false){
        die("ERROR: Could not connect to database – stage of article listing. " . mysqli_connect_error());
    }

    // read all rows (data) from guestbook table in „test“ database
    $_usr_id = $_SESSION['users_id'];
    $sql = „SELECT * FROM bazaar_item WHERE cart_number=“.$_usr_id.“ ORDER BY item_id ASC „; // read items marked in cart_number with appropriate users_id
    /******
    ******/
    /* Output in Table – solution 1 – for debugging data from database */
    /******
    ******/
    // if data properly selected from guestbook database table

    echo „<br>“;

    echo „<br>“; echo „<br>“;
    if($output = mysqli_query($dbc, $sql)){
        if(mysqli_num_rows($output) > 0){ // if any record obtained from SELECT query
            // create table output
            echo „<table>“; //head of table
            echo „<tr>“;
                echo „<th>id</th>“;
                echo „<th>Name</th>“;
                echo „<th>Price</th>“;

```

```

        echo „<th>Category</th>“;
        echo „<th>Screenshot1</th>“;
        echo „<th>More info</th>“;

        echo „</tr>“;
        $cart_total_eur = 0; // initialize variable calculating total price for items in cart
        while($row = mysqli_fetch_array($output)){
            //next rows outputted in while loop
            echo “ <div class=“mailinglist“> “ ;
            echo „<tr>“;
            //echo „<td>“ . $row[,item_id'] .
            „</td>“;
            echo „<td class=“item_name“>“ .
            $row[,name_of_item'] . „</td>“;
            echo „<td class=“price“>“ . $row
            [,price_eur'] . “ € </td>“;
            $cart_total_eur += $row[,price_eur
            '];

        * convert category_id in to category and subcategory */
            $subcategory_id = $row
            [,subcategory_id'];
            $category_idsupl =
            „“ ;
            $subcategory_idsupl =
            „“ ;
            // (*) – conversion of
            category and subcategory into category%id

            // create SELECT query for category and subcategory names from database
            $sql_supl = „SELECT
            category, subcategory FROM bazaar_category WHERE subcategory
            _id = „.$subcategory_id'“ ;
            /*$output_supl = m
            ysqli_query($dbc, $sql_supl);
            $row_supl = mysqli
            _fetch_array($output_supl);
            $category_id =

```



```

$row_supl[,category'] ;
                                                                    $subcategory_id =
$row_supl[,subcategory'] ;
                                                                    echo „<td>“ . $cat
egory_id.“/“.$subcategory_id.“</td>“;*/
                                                                    // execute sql and
populate data list with existing category in database
                                                                    if($output_supl =
mysqli_query($dbc, $sql_supl)){
                                                                    if(mysqli_num_
rows($output_supl) > 0){ // if any record obtained from SELEC
T query
                                                                    while($row
_supl = mysqli_fetch_array($output_supl)){ //next rows outpute
d in while loop
                                                                    $categ
ory_idsupl      = $row_supl[,category'] ;
                                                                    $subca
tegory_idsupl = $row_supl[,subcategory'] ;

                                                                    }

                                                                    // Free re
sult set
                                                                    mysqli_fre
e_result($output_supl);
                                                                    } else {
                                                                    echo „Ther
e is no such category-
subcategory in category table. Please correct your error.“; //
if no records in table
                                                                    }
                                                                    } else{
                                                                    echo „ERROR: C
ould not able to execute $sql. “ . mysqli_error($dbc); // if d
atabase query problem
                                                                    }
                                                                    echo „<td>“ . $category_idsupl.“/“

```

```
.$subcategory_idsupl."</td>";
```

```
$image_location = IMAGE_PATH.$
```

```
row[,screenshot1'];
```

```
echo „<td id=\"gray_under_picture\"> <img src=\""$image_location\" alt=\" screenshot of product  
primary \" height=\"250\"> </td>“;
```

```
echo „<td colspan=\"1\"><a id=\"DEL\"  
href=\"removefromcart.php?cart_number='.$row[,cart_number'] . ,&  
&item_id=' . $row[,item_id'] . ,&name_of_item=' . $row[,n  
ame_of_item'] .'\"> >> Remove from cart </a></td></tr>“; //con  
struction of GETable link
```

```
echo „</tr>“;
```

```
echo “ </div> “ ;
```

```
}
```

```
echo „</table>“;
```

```
echo „<br><br>“;
```

```
echo „<p><center><h5>Total price for items  
in cart: <strong> $cart_total_eur </strong>€ </h5></center></  
p>“;
```

```
// Free result set
```

```
mysqli_free_result($output);
```

```
} else{
```

```
echo „There is no item for sell. Please ad  
d one.“; // if no records in table
```

```
}
```

```
} else{
```

```
echo „ERROR: Could not able to execute $sql. “  
. mysqli_error($dbc); // if database query problem
```

```
}
```

```
// Close connection
```

```
mysqli_close($dbc);
```

```
?>
```

```
<!-- Recapitulation of user delivery adress – important be  
acause is sent to seller with e-  
mail about succesfull buy of listened item ->
```

```
<h4> Your delivery adress is: </h4>
```

```
<?php
```

```
// connect to a database
```

```
$dbc = mysqli_connect(DB_HOST, DB_USER, DB_PW, DB_
```

```
NAME);
```

```
        // Check connection
        if($dbc === false){
            die(„ERROR: Could not connect to database –
stage of article listing. “ . mysqli_connect_error());
        }

        // get info about user from database
        $users_id = $_SESSION[,users_id'];
        $sql = „SELECT * FROM bazaar_user WHERE users_id =
„.“, $users_id'“.“LIMIT 1” ;
        if($output = mysqli_query($dbc, $sql)){
            if(mysqli_num_rows($output) > 0){ // if any re
cord obtained from SELECT query

                while($row = mysqli_fetch_array($output)){
//next rows outputed in while loop
                    $first_name = $row[,first_name'];
                    $lastname_name = $row[,lastname_nam
e'];

                    $addressss = $row[,addressss'];
                    $city = $row[,city'];
                    $ZIPcode = $row[,ZIPcode'];
                    $email = $row[,email'];
                    ?>

                    <div id=“frame_green“>

                        <br>
                        <h5> Please check your contact
and delivery info, these information are important for
seller of the item
s for correct contact and delivery! </h5>
                        <br>
                        <table>
                            <tr>

                                <td>
                                <label>e-mail:</label>
                                <input type=“text“ na
me=“nickname“                                class=“form-
```

```

control" value="<?php echo $email; ?>" disabled>
<br>
</td>
<td>
<label>First name:</la
bel>
<input type="text" nam
e="first_name"
class="form-
control" value="<?php echo $first_name; ?>" disabled>
<br>
</td>
<td>
<label>Last name:</lab
el>
<input type="text" nam
e="lastname_name"
class="form-
control" value="<?php echo $lastname_name; ?>" disabled>
<br>
</td>
<tr>
</tr>
<td colspan="3">
<label>Address in form
- Street Nr.:</label>
<input type="text" na
me="addresss"
class="form-
control" value="<?php echo $addresss; ?>" disabled>
<br>
</td>
<tr>
</tr>
<td colspan="3">
<label>City:</label>
<input type="text" nam
e="city"
class="form-
control" value="<?php echo $city; ?>" disabled>
<br>
</td>
<tr>
</tr>
<td colspan="3">

```

```

        <label>ZIP code in for
m XXXXX:</label>

        <input type="text" nam
e="ZIPcode"          class="form-
control" value="<?php echo $ZIPcode; ?>" disabled>
        </td>

        </tr>
    </table>
    <br>

    <h5> If any of display
ed info need correction, please visit your profile page <a href
f="editprofile.php"><u>here</u>. </a></h5>
    </div>

    <?php

    }

    // Free result set
    mysqli_free_result($output);
} else{
    echo „Error while reading data.“; // if no
records in table
}
} else{
    echo „ERROR: Could not able to execute $sql. “
. mysqli_error($dbc); // if database query problem
};
// Close connection
mysqli_close($dbc);
?>
<br>
</h4>
    <form method="post" action="<?php echo $_SERVER[‘PHP_S
ELF’]; ?>">
        <input type="hidden" name="users_id" value="<?php ec
ho $_SESSION[‘users_id’] ?>" />
        <h5> For confirmation of buy select YES and click on
red button bellow:</h5>
        <center><input type="radio" name="confirm" value="Ye

```

```

s" /> Yes    <br>
            <input type="radio" name="confirm" value="No" checked="checked" /> No </center><br><br>
            <center><button type="submit" name="submit" class="btn btn-danger btn-bg"> I confirm the purchase with the obligation to pay </button> </center>

            <br><br>
        </form>

    </div>

    <div class="footer">
        <a class="navbar-brand" href="https://cdesigner.eu"> Visit us on CDesigner.eu </a>
    </div>

</body>
</html>

```

Cart page generate GETable link to script removefromcart.php to enable remove item from cart. This is done by reverting cart_number filed of appropriate item from number equal to buyers users_id to zero. User is befor removing item also informed about that item is now visible for other portal user for buy. And second decision must not be available, because item will add another user to a cart.

removefromcart.php script

Content of our supplementary removal script follows here:

```
<!--
*****
** -->
<!-- PHP „self“ code GET request for remove from cart
-->

<!--
*****
** -->
<!-- Vrsion: 1.0           Date: 2.11.2020 by CDesigner.eu
-->

<!--
*****
** -->
<?php // leading part of page for simple header securing and basic variable setup
    require_once('appvars.php'); // including variables for database
    session_start(); // start the session – must be added on all pages for session variable accessing
    // solution using SESSIONS with COOKIES for longer (30days) login persistency

    if(!isset($_SESSION['users_id'])) { // if session is no more active
        if(isset($_COOKIE['users_id']) && isset($_COOKIE['username'])) { // but cookie is set then renew session variables along them
            $_SESSION['users_id'] = $_COOKIE['users_id'];
            $_SESSION['username'] = $_COOKIE['username'];
            $_SESSION['user_role'] = $_COOKIE['user_role']; // added for role
        }
    }

    // two variables for message and styling of the message with bootstrap
    $msg = "";
    $msgClass = "";
```

```
// default values of auxiliary variables

?>
<!-- ***** ->
<!-- script for removing item from cart ->
<!-- ***** ->
<!-- obtain GET data from cart.php and trough ->
<!-- POST submit remove goods from cart by ->
<!-- seting cart_number filed to 0 - notassigned ->
<!-- to any user ->
<!-- ***** ->
<!DOCTYPE html>
<html>
<head>
    <title> Bazaar remove from cart – remove script </title>
    <link rel="stylesheet" href="./css/bootstrap.min.css"> <!-- bootstrap mini.css file ->
    <link rel="stylesheet" href="./css/style.css"> <!-- my local.css file ->
    <script src="https://code.jquery.com/jquery-3.1.1.slim.min.js" integrity="sha384-A7FZj7v+d/sdmMqp/nOQwliLvUsJfDHW+k90mg/a/EheAdgtzNs3hpfag6Ed950n" crossorigin="anonymous"></script>
    <script src="https://cdnjs.cloudflare.com/ajax/libs/tether/1.4.0/js/tether.min.js" integrity="sha384-DztdAPBWPRXSA/3eYEEUWrWCy7G5KFbe8fFjk5JAIxUYHKkDx6Qin1DkWx51bBrb" crossorigin="anonymous"></script>

</head>
<body>
    <nav class="navbar navbar-default">
        <div class="container">
            <div class="navbar-header">
                <a class="navbar-brand" href="cart.php">Return to your shopping car</a>
                <a class="navbar-brand" href="index.php"> -> return to main shop page</a>
            </div>
        </div>
    </nav>
    <div class="container" id="formcontainer">
```



```

<?php if($msg != ""): ?> <!-- alert showing part -->
    <div class="alert <?php echo $msgClass; ?>"><?php echo
$msg; ?></div>
    <?php endif; ?>

    <br> <!-- logo on the center of the page -->
    <h4>Confirmation of removal item from cart.</h4>
    <br>
    <br> <!-- logo on the center of the page -->
    
    <br>

    <?php // code for GET info about what to remove and subm
it removing approval
        if(isset($_GET[,cart_number']) && isset($_GET[,item_id
']) && isset($_GET[,name_of_item'])) ){
            // take a data from GET link generated by adminscr
ipt
            $cart_number = htmlspecialchars($_GET[,cart_number
']);
            $item_id = htmlspecialchars($_GET[,item_id']);
            $name_of_item = htmlspecialchars($_GET[,name_of_it
em']);

            } else if (isset($_POST[,cart_number']) && isset($_POS
T[,item_id']) && isset($_POST[,name_of_item'])) { //grab score
from POST – different behavior for removal
            $cart_number = htmlspecialchars($_POST[,cart_numbe
r']);
            $item_id = htmlspecialchars($_POST[,item_id']);
            $name_of_item = htmlspecialchars($_POST[,name_of_i
tem']);

            } else { //error info message
                echo ,<p class="alert alert-
danger"> Please specify any cart item for removal. </p>';

```

```

};
if(isset($_POST['submit'])){

    if($_POST['confirm'] == 'Yes' ){ // delete appropriate score post with imagescreenshot
        //delete the screenshotimage from the
        $cart_number = htmlspecialchars($_POST['cart_number']);

        $item_id = htmlspecialchars($_POST['item_id']);
        $name_of_item = htmlspecialchars($_POST['name_of_item']);

        // conect to the database
        $dbc = mysqli_connect(DB_HOST, DB_USER, DB_PW, DATABASE_NAME);

        //Delete score data from the database
        $sql = „UPDATE bazaar_item SET cart_number = ,0‘
WHERE item_id = $item_id LIMIT 1“;
        // execute SQL
        mysqli_query($dbc, $sql);
        // close database connection
        mysqli_close($dbc);
        // confirm executed command
        echo „<p> The item , . $name_of_item . , with id
<strong>‘ . $item_id . ,</strong> was sucesfully removed from
your cart and now is available in listening for sell
for another user. </p>‘;

    } else {

        echo „<p class=“alert alert-danger“ > The selected item cannot be removed. </p>‘;
    }
    } else if (isset($cart_number) && isset($item_id) && isset($name_of_item) ) {
        echo „<h5>Are you sure to remove , . $name_of_item
. , from your cart? Item will be set for sell listening and can be bought by another user.</h5>‘;
        // show short describtion of score for deletion
        echo „<p> <strong> item_id: </strong> , . $item_id
. ,<br> <strong> item name is: </strong>‘ . $name_of_item .

```

```

        ,</p>';

//generating removing confirmation form

        echo ,<form method="POST" action="removefromcart.p
hp">'; //not self but direct this script removecategory.php
- we dont want include any GET data tahat previously send
        echo ,<input type="radio" name="confirm" value="Ye
s" /> Yes ,;
        echo ,<input type="radio" name="confirm" value="No
" checked="checked" /> No <br><br>';

        echo ,<input type="hidden" name="cart_number" valu
e=",.$cart_number.'" />';
        echo ,<input type="hidden" name="item_id" value="
,.$item_id.'" />';
        echo ,<input type="hidden" name="name_of_item" val
ue=",.$name_of_item.'" />';
        echo ,<input type="submit" class="btn btn-
danger" value="submit" name="submit" />';
        echo ,</form>';

};
echo ,<br><br>';
echo ,<p> <a href = „cart.php"> &lt;&lt; Back to your
cart. </a></p>';
?>

</div>

<div class="footer">

                                <a class="navbar-
brand" href="https://cdesigner.eu"> Visit us on CDesigner.eu <
/a>
</div>

</body>
</html>

```

Conclusion and final thoughts

Our cart solution implement simple way how to manage users items for buy. For further improvements is important mark sold items for admin removal or better mark them for removal by automated script after some time. Way how to do this, is mark item after succesfull buy comitt them from users_id to -1 (cart_number filed of that item in bazaar_item table). This mean item was sold and after some time they must be removed from portal.

Full aplcation code for further study can be obtained from github [here](#).

Bazaar – php example code – part 7 – limiting user access and diferent page content display

Article deeper focus on mechanism how to use session variables created for logged in user for diferentiating page display. This mechanism relays on login features described in article part 6. Together make reusable solution for another projects

with need in restricting access and different page content display for different categories of users.

Different content display – how to?

In many cases is important to display content of the page differently for different types of users. Page for anonymous user will not display ability for editing user profiles or only logged in user can sell or buy listed items.

First part of all pages restricting access to them contains initialization session function call and reading info from session. If session is not set up, then look at data stored in cookies. If cookies contain valid information about logged in users, then session information is per page restored by their content. If not, page display content as for anonymous user.

This code looks like this

```
require_once('appvars.php'); // including variables for database
// two variables for message and styling of the message with bootstrap
session_start(); // start the session – must be added on all pages for session variable accessing
// solution using SESSIONS with COOKIES for longer (30days) login persistency

if(!isset($_SESSION['users_id'])) { // if session is no more active
    if(isset($_COOKIE['users_id']) && isset($_COOKIE['username'])) { // but cookie is set then renew session variables along them
```

```

        $_SESSION[,users_id'] = $_COOKIE[,users_id'];
        $_SESSION[,username'] = $_COOKIE[,username'];
        $_SESSION[,user_role'] = $_COOKIE[,user_role']; //
added for role
    }
}

```

Diferent page content display

For diferent page display, we request for existence one fo session variable. If present, user is logged in, if not alternative part of page code must be shown.

```

<!-- ***** _->
<!-- HTML part available after succesfull login as user ->
<!-- ***** _->

```

```

<?php if(isset($_SESSION[,users_id']) ) { //if user is logged w
ith users_id then editprofile form is available?>

```

... part of page code shown if user logged in ...

```

<!-- ***** _->
<!-- HTML part displayed for unlogged user _->
<!-- ***** _->
<?php } else { // else if user is not logged then form will noo
t be diplayed?>

```

```

        <br>
        
        <br>
        <h4>For listening items for sell you must be logged in
<a class="navbar-brand" href="login.php"> here. </a></h4>
        <br>

```

```

<?php } ?>

```

Main page menu variable content display

Main menu is another example of part fo adaptive dsisplay relaying on category of logged in user.

Next code display way how to display their content.

```
<body>
    <nav class="navbar navbar-default">
        <div class="container">
            <div class="navbar-header">
                <?php // generate menu if user is logged in or not
                    // old solution with cookies if(isset($_COOKIE['username'])) { // logged in user
                        if(isset($_SESSION['username'])) { // logged in user
                            echo '<a class="navbar-brand" href="index.php">Bazaar – best items for a best prices!</a>';
                            echo '<a class="navbar-brand" href="editprofile.php"> Edit profile </a>';
                            echo '<a class="navbar-brand" href="logout.php"> Logout , $_SESSION[username] .</a>';
                            if(isset($_SESSION['user_role'])=='admin') {
                                echo '<a class="navbar-brand" href="admin.php"> Manage your page </a>';
                            };
                            require_once('sell_icon.php'); // graphic menu item for selling your items – we focus on this two items in next articles from tihis series
                            require_once('cart_icon.php'); // small cart icon in menu
                        } else { // visitor without login
                            echo '<a class="navbar-
```

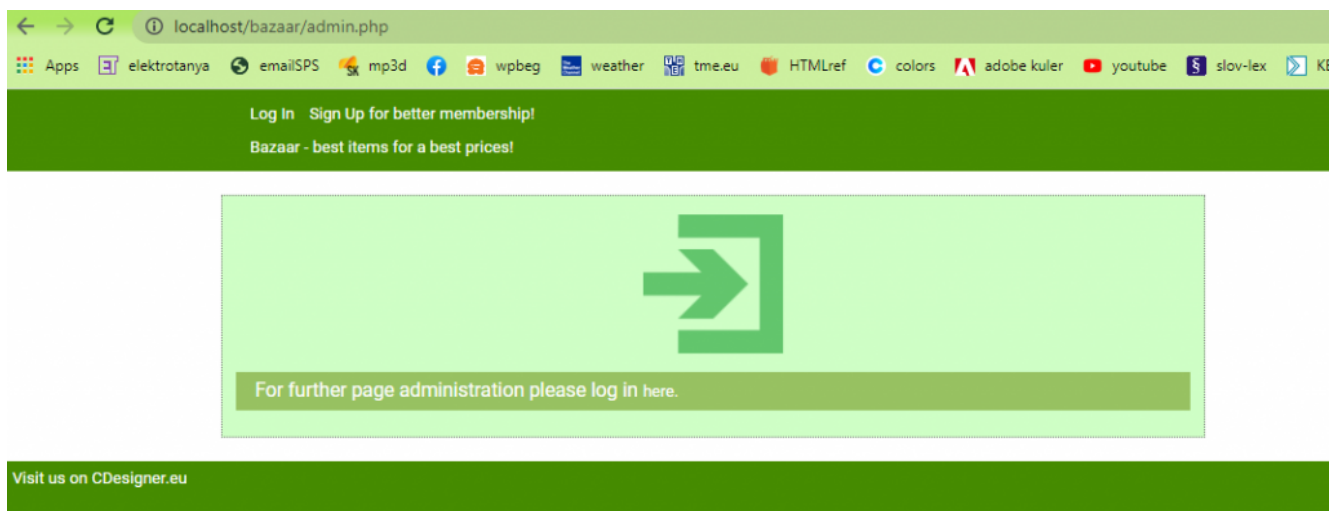
```

brand" href="login.php"> Log In </a>';
                                echo ,<a class="navbar-
brand" href="signup.php"> Sign Up for better membership! </a>'
;

                                echo ,<a class="navbar-
brand" href="index.php">Bazaar – best items for a best prices!
</a>';
                                }
                                ?>
                                </div>
                                </div>
                                </nav>

```

Different display of content shows following pictures on example of admin page.



Display of admin page of unlogged user



You must be logged with administrator role for admin. Please [logout](#) and then login as admin.

Display of admin page for user with low privileges – relogin as admin requested.



Logged in as admin

Conclusions and further thoughts

We described ways how to display different content of the pages for different categories of users. One mechanism uses the existence of session variables of the logged user for making a decision of which part of the page is visible.

Implementation of these features in the whole project can be visible on our github account [here](#).

Bazaar – php example code – part 6 – user login with SESSIONS and COOKIES

This new year article will focus on using `$_SESSION` and `$_COOKIE` variables in proces on login user into a application. As first part of our next topics focused on user validation and diferentiation app behavior we will take closer look on signing up and login of page user.

Login mechanism in closer look

In our previeous application of mailer, we used for restricting access in to a specific page only HTTP header authentication mechanism. This approach is very simple but not scale well in larger sites with different pages with partialy or complet access restrictions.

New approach relays on login ability created by login.php script. Login script obtain user login credentials, compare it with those stored in database (passwords can not be stored as plain text, but we store only sha1 hashes and make comparism with hashes together for deciding about corect or incorrect login).

After successful login \$_SESSION variables are set. For better persistency there is made combination with \$_COOKIE variable. COOKIES are stored in local users browser and are available only if they are enabled by user, that must be kept in mind.

In all pages are SESSION started in first parts of php code and from COOKIES restore sessions among all pages where login restrictions must be made. In our next article, we take a closer look how limiting or differentiating page looks for different category of users introduced with category of users database field.

For logout, there is available logout script with mechanism for invalidating COOKIES and clearing SESSIONS variables.

Creating new users with sign up script

Our application needs mechanism for subscribing new users for deeper access in pages for submitting items for sell or more specific page available only for admin role of user – adminpage.

New user provides username, e-mail and type 2x password. Username must be unique and not used by another user. This mechanism is implemented in sql query but in our approach not in UNIQUE restriction in database field.

After successful login credentials creation, user can log in with existing login script.

Now we can take closer look at mentioned pages.

Sign Up script

User provide from sign up form username, e-mail and password. If username is unique (no unique e-mail is needed, because we can expect different logins of seler for different account with the same e-mail) user login credentials are added and user with role „user“ is introduced into bazar_user database table.

User role grant ability to buy and sell items, but not provide management ability mark items as eligible for visibility on title page of bazaar.

Listening of our script follows.

```
<!--
*****
*** -->
<!-- PHP „self“ code handling sign up for membership on the bazar app -->
<!--
*****
*** -->
<!-- Vrsion: 1.0          Date: 24.10-24.10.2020 by CDesigner.eu
-->

<!--
*****
*** -->
<?php
    require_once('appvars.php'); // including variables for database

    // two variables for message and styling of the message with bootstrap
```

```

$msg = "";
$msgClass = "";
$u_name = "";
$usr_passwd = "";
/* Attempt MySQL server connection. */
$dbc = mysqli_connect(DB_HOST, DB_USER, DB_PW, DB_NAME);
if(isset($_POST['submit'])) {
    // obtaining submitted data from POST
    $u_name = htmlspecialchars($_POST['u_name']);
    $u_pass_1 = htmlspecialchars($_POST['u_pass_1']);
    $u_pass_2 = htmlspecialchars($_POST['u_pass_2']);
    $email = htmlspecialchars($_POST['email']);
    if(!empty($u_name) && !empty($email) && !empty($u_pass_1)
    && !empty($u_pass_2) && ($u_pass_1 == $u_pass_2)) {
        // make sure that username is available and is not regist
        ered for someone else
        $sql = „SELECT * FROM bazaar_user WHERE username = „.$u
        _name'“ ;
        $data = mysqli_query($dbc, $sql);

        if(mysqli_num_rows($data) == 0) {
            // username is unique and have not been used by any
            previous user
            $usr_passwd_sha1 = sha1($u_pass_2);
            $sql = „INSERT INTO bazaar_user (username, pass_wor
            d, write_date, email, nickname)
            VALUES (,$u_name', , $usr_passwd_sha1' , now
            ()), , $email', '$u_name')“; // by default nickname and username
            are the same, next user can change
            if(mysqli_query($dbc, $sql)){
                $msg = , Your new account has been created success
                fully.
                You are now ready to <a href="login.php">log in</a
                >;
                $msgClass = ,alert-success';
            } else{
                echo „ERROR: Could not able to execute $sql. “
                . mysqli_error($dbc); // if database query problem
            }
            //success confirmation for registered user

```

```

        // Free result set
mysqli_free_result($data);
        // Close connection

        //exit(); if used blank page will be displayed without any other redirecting
    } else { // an account already exists for this username
, so display an error message

        $msg = , An account for submitted username already exists. Please use different username ...';
        $msgClass = ,alert-danger';
    }
} else {

        $msg = , You must enter all of the required data, including contact e-mail address.';
        $msgClass = ,alert-danger';
    }

}

    // Close connection
mysqli_close($dbc);
?>
<!-- ***** ->
<!-- HTML code containing Form for submitting ->
<!-- ***** ->
<!DOCTYPE html>
<html>
<head>
    <title> Bazaar signup page </title>
    <link rel="stylesheet" href="./css/bootstrap.min.css"> <!-- bootstrap mini.css file ->
    <link rel="stylesheet" href="./css/style.css"> <!-- my local.css file ->
    <script src="https://code.jquery.com/jquery-3.1.1.slim.min.js" integrity="sha384-A7FZj7v+d/sdmMqp/nOQwliLvUsJfDHW+k90mg/a/EheAdgtzNs3hpfag6Ed950n" crossorigin="anonymous"></script>
    <script src="https://cdnjs.cloudflare.com/ajax/libs/te

```

```
ther/1.4.0/js/tether.min.js" integrity="sha384-DztdAPBWPRXSA/3eYEEUWrWCy7G5KFbe8fFjk5JAIxUYHKkDx6Qin1DkWx51bB  
rb" crossorigin="anonymous"></script>
```

```
</head>
```

```
<body>
```

```
  <nav class="navbar navbar-default">
```

```
    <div class="container">
```

```
      <div class="navbar-header">
```

```
        <a class="navbar-brand" href="index.php">Bazaar – Signup for submitting/ buying  
        your items</a>
```

```
      </div>
```

```
    </div>
```

```
  </nav>
```

```
  <div class="container" id="formcontainer">
```

```
    <?php if($msg != ""): ?>
```

```
      <br>
```

```
      <div class="alert <?php echo $msgClass; ?>"><?php echo  
$msg; ?></div>
```

```
    <?php endif; ?>
```

```
    <br>
```

```
    
```

```
    <br>
```

```
    <form method="post" action="<?php echo $_SERVER[,PHP_  
SELF']; ?>">
```

```
      <div id="login">
```

```
        <legend> Please register for Bazaar membership
```

```
    </legend>
```

```
      <label>Username:</label>
```

```
      <input type="text" onfocus="this.value='<?  
php echo isset($_POST[,u_name']) ? $u_name : "; ?>' " name="u_n  
ame" class="form-  
control" value="<?php echo isset($_POST[,u_name']) ? $u_name :  
,Login name'; ?>">
```

```

        <label>e-mail:</label>
        <input type="text" onfocus="this.value='<?
php echo isset($_POST[,email']) ? $email : "; ?>' " name="email
"
                class="form-
control" value="<?php echo isset($_POST[,email']) ? $email : ,
@'; ?>">

        <label>Password:</label>
        <input type="password" onfocus="this.value
='<?php echo isset($_POST[,u_pass_1']) ? " : "; ?>' " name="u_p
ass_1"
                class="form-
control" value="<?php echo isset($_POST[,u_pass_1']) ? " : ";
?>">

        <label>Password:</label>
        <input type="password" onfocus="this.value
='<?php echo isset($_POST[,u_pass_2']) ? " : "; ?>' " name="u_p
ass_2"
                class="form-
control" value="<?php echo isset($_POST[,u_pass_2']) ? " : ";
?>">

        </div>
        <input id="loginsubmitt" type="submit" name="submit
" class="btn btn-info" value="Sign In">
        <br>
</form>

</div>

<div class="footer">
                <a
                        class="navbar-
brand" href="https://cdesigner.eu"> Visit us on CDesigner.eu <
/a>
        </div>

</body>
</html>

```

Front look at signup page follows.

Your new account has been created successfully. You are now ready to [log in](#)



Please register for Bazaar membership

Username:

e-mail:

Password:

Password:

Login page

Code for loginpage take place only if user is not allready loged in. In this option is not set `isset($_SESSION[,users_id'])` . After verfication of login credentials SESSION variables and COOKIES are set for loged in user. Also loed in user is redirected on index.php main page as it contains next code.

```
<!--
```

```
*****
```

```
*** -->
```

```

<!-- PHP „self“ code handling login into the bazaar app
        ->

<!--
*****
*** ->
<!-- Vrsion: 1.0          Date: 11.10-24.10.2020 by CDesigner.eu
        ->

<!--
*****
*** ->
<?php
    require_once(,appvars.php'); // including variables for datab
ase
    session_start(); // start the session

    // two variables for message and styling of the mesage with b
ootstrap
    $msg = “;
    $msgClass = “;
    $usr_username = “;
    $usr_passwd = “;
//get info that user is logged in, if not try it looking at coo
kies
//if(!isset($_COOKIE[,s’])) { old solution with cookies
    if(!isset($_SESSION[,users_id’])) { //new with session varia
bles
        if(isset($_POST[,submit’])) {
            /* Attempt MySQL server connection. */
            $dbc = mysqli_connect(DB_HOST, DB_USER, DB_PW, DB
_NAME);

                // accessing user entered login data
            $usr_username = htmlspecialchars($_POST[,u_name’]
);
            $usr_passwd = htmlspecialchars($_POST[,u_pass’]);

            if(!empty($usr_username) && !empty($usr_passwd))
{
                // try lookup user database
                $usr_passwd_SHA = sha1($usr_passwd);
                $sql = „SELECT users_id, username, user_role FRO

```

```

M bazaar_user WHERE username = „.“, $usr_username'“. “ AND pass
_word = „.“, $usr_passwd_SHA'“ ;
    // debug output echo $usr_username;
    // echo $usr_passwd;
    //echo $usr_passwd_SHA;
    $data = mysqli_query($dbc, $sql);

    if(mysqli_num_rows($data) == 1) {
        // login is ok, set user ID and username co
        okies and redirect to the homepage
        $row = mysqli_fetch_array($data);
        //setcookie(,users_id', $row[,users_id']); o
        ld solution with cookies
        //setcookie(,username', $row[,username']);
        $_SESSION[,users_id'] = $row[,users_id']; //
        sloution with sessions
        $_SESSION[,username'] = $row[,username'];
        $_SESSION[,user_role'] = $row[,user_role'];
        // added user_role session variable
        // new cookies for login persistency that ex
        pires after 30 days without logout combination SESSION with CO
        OKIES is available
        setcookie(,users_id', $row[,users_id'], time
        ()+(60+60*24*30));
        setcookie(,username', $row[,username'], time
        ()+(60+60*24*30));
        setcookie(,user_role', $row[,user_role'], ti
        me()+(60+60*24*30)); // cookie for user_role of logged in user
        added
        $home_url = ,http://'. $_SERVER[,HTTP_HOST']
        . dirname($_SERVER[,PHP_SELF']) . ,/index.php';
        header(,Location:'. $home_url);
        // Free result set
        mysqli_free_result($data);
        // Close connection
        mysqli_close($dbc);
    } else {
        // urname/ password are incorrect – error m
        eesage is displayed
        $msg = „Incorrect username or password. Logi
        n denied! „;
    }
}

```

```

        $msgClass = ,alert-danger';

    }

    } else {
        // username/ password were not entered – display error message
        $msg = „Sorry, you must enter username and password to log in. „;
        $msgClass = ,alert-danger';

    }

}
?>
<!-- ***** ->
<!-- HTML code containing Form for submitting ->
<!-- ***** ->
<!DOCTYPE html>
<html>
<head>
    <title> Bazaar login page </title>
    <link rel="stylesheet" href="./css/bootstrap.min.css"> <!-- bootstrap mini.css file ->
    <link rel="stylesheet" href="./css/style.css"> <!-- my local.css file ->
    <script src="https://code.jquery.com/jquery-3.1.1.slim.min.js" integrity="sha384-A7FZj7v+d/sdmMqp/n0QwliLvUsJfDHW+k90mg/a/EheAdgtzNs3hpfag6Ed950n" crossorigin="anonymous"></script>
    <script src="https://cdnjs.cloudflare.com/ajax/libs/tether/1.4.0/js/tether.min.js" integrity="sha384-DztdAPBWPRXSA/3eYEEUWrWCy7G5KFbe8fFjk5JAIxUYHKkDx6Qin1DkWx51bBrb" crossorigin="anonymous"></script>

</head>
<body>
    <nav class="navbar navbar-default">
        <div class="container">
            <div class="navbar-header">
                <a class="navbar -

```

```

brand" href="index.php">Bazaar – Login page</a>
    </div>
</div>
</nav>
<div class="container" id="formcontainer">
<?php if($msg != ""): ?>
    <br>
    <div class="alert <?php echo $msgClass; ?>"><?php echo
$msg; ?></div>
    <?php endif; ?>

    <?php
        //if(empty($_COOKIE[,users_id'])) { solution with
cookies
        if(empty($_SESSION[,users_id'])) { // solution w
ith sessions
            // only show for if session with name users_id
does not exist
            //echo , <br> ;;
            //echo ,<p class="alert alert-
danger">' . $msg . ,</p>';
            ?>

            <br>
            
            <br>
            <form method="post" action="<?php echo $_SERVER[,PHP_
SELF']; ?>">
                <div id="login">
                    <legend> Log In </legend>
                    <label>Username:</label>
                        <input type="text" onfocus="this.value='<?
php echo isset($_POST[,u_name']) ? " : "; ?>' " name="u_name" c
lass="form-
control" value="<?php echo isset($_POST[,u_name']) ? ,Please r
eenter' : ,Login name'; ?>">
                            <label>Password:</label>
                                <input type="password" onfocus="this.value
='<?php echo isset($_POST[,u_pass']) ? " : "; ?>' " name="u_pas
s"
                                class="form-

```

```

control" value="<?php echo isset($_POST[,u_pass']) ? ,Please r
eenter' : ,Login name'; ?>">
    </div>
    <input id="loginsubmitt" type="submit" name="submit
" class="btn btn-info" value="Log In">
    <br>
</form>
<?php } else {
    // successfull login
    // cookie solution echo ,<p class="alert ale
rt-
success"> You are logged in as , . $_COOKIE[,username']. ,</p>'
;
        echo ,<br>';
        echo ,<p class="alert alert-
success"> You are logged in as <em>' . $_SESSION[,username']. ,
</em></p>'; // session solution
        echo ,<p class="alert alert-
success"> If you will logout or login with anither credentials
, please first <a href="logout.php">logout!. </a></p>';
    }
    ?>
</div>

<div class="footer">
    <a class="navbar-
brand" href="https://cdesigner.eu"> Visit us on CDesigner.eu <
/a>
</div>

</body>
</html>

```

Next pictures display output on page after difrenet state of login process.



Log In

Username:

Password:

LOG IN

Log In page of Bazaar app

Incorrect username or password. Login denied!



Log In

Username:

Password:

LOG IN

Error messaging after unsuccesfull login

Logout script

Last and shortest code page is logout page. After hitting link for logout, user sessions and cookies are immediately destroyed and user is logged out. Without logout with cookies enabled, user is logged in in browser for 30 days. If browser disables cookies, user is logged in only if sessions are active up to closing browser windows.

Full code follows:

```
<!--
*****
*** -->
<!-- PHP „self“ code handling logout procedure into the bazaar
app -->
<!--
*****
*** -->
<!-- Vrsion: 1.0          Date: 24.10-24.10.2020 by CDesigner.eu
-->
<!--
*****
*** -->
<?php
    require_once('appvars.php'); // including variables for database
    // part for SESSION solution for login persistence and its ending
    //even when logging out you have to first start the session in order to access the session variables
    session_start();
    if(isset($_SESSION['users_id'])) {
```



```

$_SESSION = array(); // deleting session vars

};
// if session cookie exists, then delete it
if(isset($_COOKIE[session_name()])) {
    setcookie(session_name(), "", time() - 3600);

};
// Destroy session
session_destroy();
// logout user by deleting cookie – for COOKIES persistence solution

/* if(isset($_COOKIE[,users_id'])) {
    setcookie('user_id', "", time() - 3600);
    setcookie('username', "", time() - 3600);
    echo „deleted cookies“;

}; */
// for our final solution SESSIONS+ longer login persistence with COOKIES must be also cookies deleted
setcookie('users_id', $row['users_id'], time()-3600);
setcookie('username', $row['username'], time()-3600);
setcookie('user_role', $row['user_role'], time()-3600); // added deletion of user_role cookie – after altering table for user_role
// redirect to homepage in logout state
$home_url = 'http://'. $_SERVER['HTTP_HOST'] . dirname($_SERVER['PHP_SELF']) . '/index.php';
header('Location: ' . $home_url);
?>

```

Conclusion and final thoughts

In this article we take focus on way how to grant user limited access in to a page resources. This solution uses sessions

with supporting role by cookies.

In our next article we will focus how to use session log in information for differentiating page display for different roles of page users (anonymous = unlogged, user-s and admin-s).

Full code in as is state can be obtained from github [here](#).

Bazaar – php example code – part 5 – admin page

Article focus on way how to create site admin page. In our first approach we focus only on content of the page, limiting access only for admin will be described in further articles.

Expectation for site admin page

Admin page is accessible only for specific user category. This category is defined by user_role as admin. Restricting access and different display of this page for different category of users will be explained in further articles.

With admin page we can publish/ unpublish or remove items for sell. Second functionality created up to time of writing of this article is category management.

Management operation is maintained by links leading to operational scripts as is `removecategory.php` or `removeitem.php`.

Removeitem php script is much more interesting because display three options with conformation of operation. After succesfull submitting one of three posible sql is executed along content inserted in \$oprtion variable.

Visual look of admin page

Visual look of admin page is as follows:



I. Manage list of items for sell

[CREATE NEW ITEM PAGE >](#)

Item_id	name	published?	date	price	category id	photo 1	photo 2	photo 3	Manage
11	Landmark photo	ok-Published	2020-10-17 17:40:33	49 €	7				Manage entry: **Publish/UnPub/Remove
10	Head first series	X-waiting	2020-10-17 17:39:38	49 €	16				Manage entry: **Publish/UnPub/Remove
9	Head first series	X-waiting	2020-10-17 16:34:10	49 €	24				Manage entry: **Publish/UnPub/Remove
8	Books for sale	ok-Published	2020-10-17 16:27:12	150 €	24				Manage entry: **Publish/UnPub/Remove
7	Books for sale	X-waiting	2020-10-17 16:24:56	150 €	24				Manage entry: **Publish/UnPub/Remove
5	Books for sale	X-waiting	2020-10-17 15:54:24	150 €	24				Manage entry: **Publish/UnPub/Remove
4	Books	X-waiting	2020-10-17 15:37:23	10 €	7				Manage entry: **Publish/UnPub/Remove

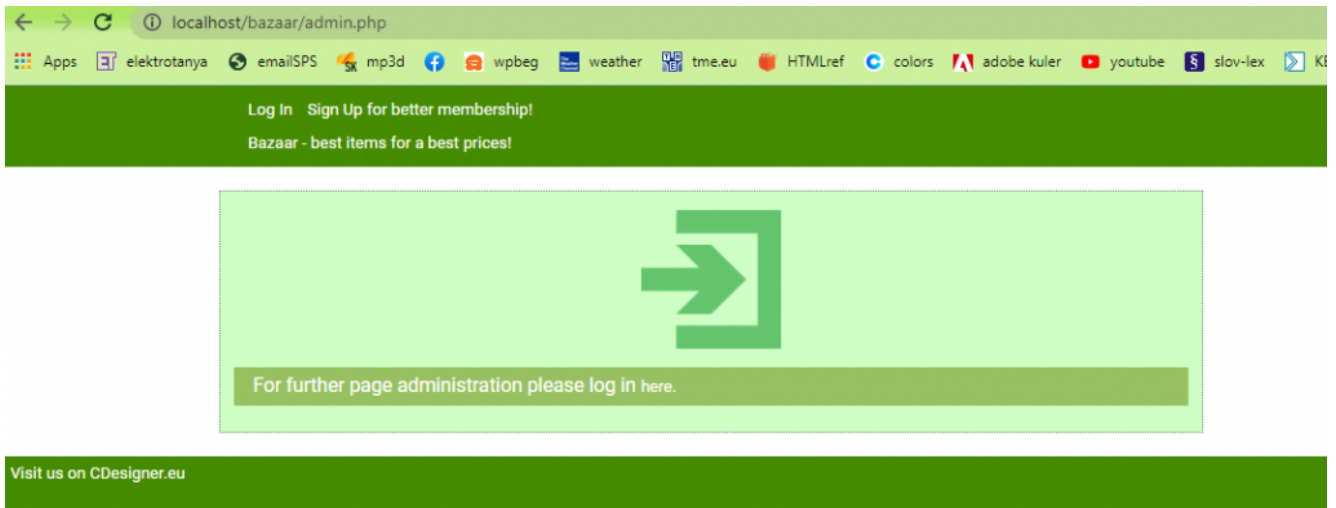
II. Manage List of active categories and subcategories on store

[CREATE NEW CATEGORY/SUBCATEGORY >](#)

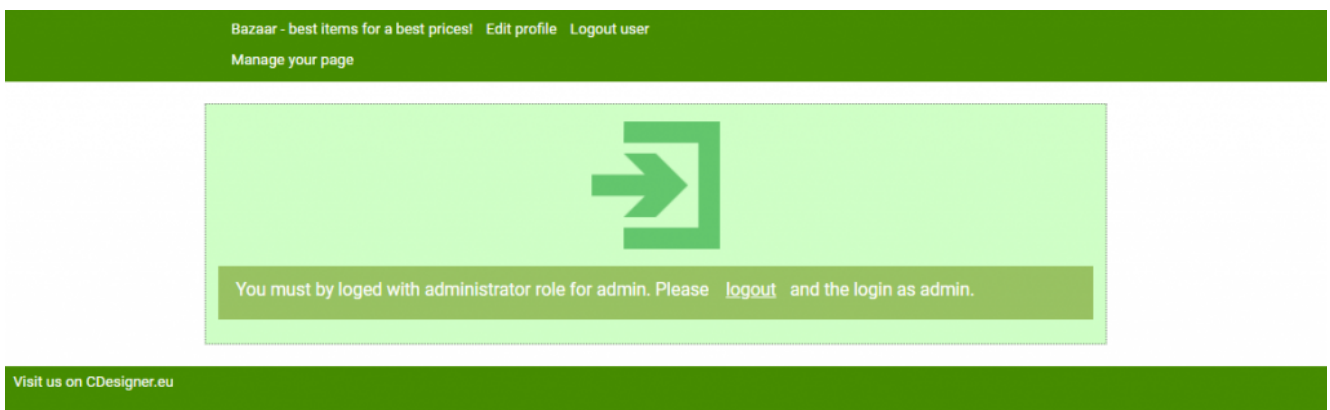
subcategory_id	category	subcategory	delete category
8	autos	van	Manage entry: ** Remove
24	books	computer	Manage entry: ** Remove
11	electronic	DVD players	Manage entry: ** Remove
6	electronic	videos	Manage entry: ** Remove
7	garden	sissors	Manage entry: ** Remove
16	networking	routers	Manage entry: ** Remove

Admin page – first implemented version

After implementing login functionality, there will be two other output of this page, one for anonymous user and one for common user role.



admin page for logged out user



admin page for common user will request relogin with elevated privilege

Implementation of desired functionality

Generating table for item and for category is most common, we will discuss these parts in previews articles. But we wil take a closer look at code of removeitem.php, with new option datalist functionality for selection among three functionalities: publishing, unpublishing and removing item by site admin.

All selection lead to different sql script, first two manipulate with contnt of field published on bazaar_item page.

If this value is zero item is not displayed in listings for users in index page. After setting on 1, items are displayed.

Removing item is LIMITED on only one matching item for further security. Also one yes selection during submitting these operations are required.

Full code of removeitem.php script will follow:

```
<!--
*****
*** -->
<!-- PHP „self“ code GET request for remove andable or disable
product item -->
<!--
*****
*** -->
<!-- Vrsion: 1.0          Date: 18.10.2020 by CDesigner.eu
-->

<!--
*****
** -->
<?php // leading part of page for simple header securing and basic variable setup
    require_once('appvars.php'); // including variables for database
    session_start(); // start the session – must be added on all pages for session variable accessing
    // solution using SESSIONS with COOKIES for longer (30days) login persistency

    if(!isset($_SESSION['users_id'])) { // if session is no more active
        if(isset($_COOKIE['users_id']) && isset($_COOKIE['username'])) { // but cookie is set then renew session variables along them
            $_SESSION['users_id'] = $_COOKIE['users_id'];
            $_SESSION['username'] = $_COOKIE['username'];
        }
    }
}
```

```

}

// two variables for message and styling of the message with
bootstrap
$msg = "";
$msgClass = "";
// default values of auxiliary variables

?>
<!-- ***** ->
<!-- script for appropriate scode removal ->
<!-- ***** ->
<!-- obtain GET data from admin.php and trough ->
<!-- POST submit remove data from database ->
<!-- ***** ->
<!DOCTYPE html>
<html>
<head>
    <title> Bazaar score – publish/unpublish/remove item script
</title>
    <link rel="stylesheet" href="./css/bootstrap.min.css"> <!-- b
ootstrap mini.css file ->
    <link rel="stylesheet" href="./css/style.css"> <!-- my local.
css file ->
    <script src="https://code.jquery.com/jquery-3.1.1.slim.min
.js" integrity="sha384-
A7FZj7v+d/sdmMqp/n0QwliLvUsJfDHW+k90mg/a/EheAdgtzNs3hpfag6Ed95
0n" crossorigin="anonymous"></script>
    <script src="https://cdnjs.cloudflare.com/ajax/libs/te
ther/1.4.0/js/tether.min.js" integrity="sha384-
DztdAPBWPRXSA/3eYEEUWrWCy7G5KFbe8fFjk5JAIxUYHKkDx6Qin1DkWx51bB
rb" crossorigin="anonymous"></script>

</head>
<body>
    <nav class="navbar navbar-default">
        <div class="container">
            <div class="navbar-header">
                <a class="navbar-
brand" href="admin.php"> -> Bazaar admin page</a>
                <a class="navbar-

```

```

brand" href="index.php"> -> return to main shop page</a>
    </div>
</div>
</nav>
<div class="container" id="formcontainer">

    <?php if($msg != ""): ?> <!-- alert showing part ->
        <div class="alert <?php echo $msgClass; ?>"><?php echo
$msg; ?></div>
        <?php endif; ?>

    <br> <!-- logo on the center of the page ->
    <h4>Please select what you will do.</h4>
    <br>
    <br> <!-- logo on the center of the page ->
        
    <br>

    <?php // code for GET info about what to remove and submit
removing approval
    /* structure of generated link on admin.php page for further
reference
        echo ,<td colspan="1"><a id="DEL" href="removeitem.php?
item_id='.$row[,item_id'] . ,&name_od_item='
            . $row[,name_of_item'] . ,&price_
eur='. $row[,price_eur'] .
                ,&published='. $row[,published']
. ,&snapshot1='. $row[,snapshot1'] .
                    ,&snapshot2='. $row[,snapshot
2'] . ,&snapshot3='. $row[,snapshot3'] . ,"> >Publish
/UnPub./Remove </a></td></tr>';
    */

    if(isset($_GET[,item_id']) && isset($_GET[,name_of_item'])
&& isset($_GET[,price_eur']) && isset($_GET[,published'])
&& isset($_GET[,snapshot1'])){
        // take a data from GET link generated by adminscript
        $item_id = htmlspecialchars($_GET[,item_id']);

```



```

        $name_of_item = htmlspecialchars($_GET[,name_of_it
em']);
        $price_eur = htmlspecialchars($_GET[,price_eur']);
        $published = htmlspecialchars($_GET[,published']);
        $screenshot1 = htmlspecialchars($_GET[,screenshot1
']);
        $screenshot2 = htmlspecialchars($_GET[,screenshot2
']);
        $screenshot3 = htmlspecialchars($_GET[,screenshot3
']);

        } else if (isset($_POST[,item_id']) && isset($_POST[,n
ame_of_item']) && isset($_POST[,price_eur']) && isset($_POST[,
published']) && isset($_POST[,screenshot1'])) { //grab score f
rom POST – different behavior for removal

        $item_id = htmlspecialchars($_POST[,item_id']);
        $name_of_item = htmlspecialchars($_POST[,name_of_i
tem']);
        $price_eur = htmlspecialchars($_POST[,price_eur'])
;
        $published = htmlspecialchars($_POST[,published'])
;
        $screenshot1 = htmlspecialchars($_POST[,screenshot
1']);
        $screenshot2 = htmlspecialchars($_POST[,screenshot
2']);
        $screenshot3 = htmlspecialchars($_POST[,screenshot
3']);

        } else { //error info message
                echo ,<p class="alert alert-
danger"> Please specify any category for removal. </p>';
        };
        if(isset($_POST[,submit'])) {

                if($_POST[,confirm'] == ,Yes' ){ // delete appropr
iate score post with imagescreenshot
                //read all data from $_POST array
                $item_id = htmlspecialchars($_POST[,item_id']);
                $name_of_item = htmlspecialchars($_POST[,name_of

```

```

_item']);
    $price_eur = htmlspecialchars($_POST[,price_eur'
]);
    $published = htmlspecialchars($_POST[,published'
]);
    $screenshot1 = htmlspecialchars($_POST[,screensh
ot1']);
    $screenshot2 = htmlspecialchars($_POST[,screensh
ot2']);
    $screenshot3 = htmlspecialchars($_POST[,screensh
ot3']);
    $operation = htmlspecialchars($_POST[,operation'
]);

    // conect to the database
    $dbc = mysqli_connect(DB_HOST, DB_USER, DB_PW, D
B_NAME);

    //create sql query along selected operation
    switch ($operation) {
        case „publish„:
            $sql = „UPDATE bazaar_item SET published =
,1' WHERE item_id = $item_id LIMIT 1„;
            // execute SQL
            mysqli_query($dbc, $sql);
            // confirm executed command
            echo „<p> The item <strong>' . $name_of_it
em . ,</strong> with id <strong>' . $item_id . ,</strong> was
succesfully published. </p>“;
            break;
        case „unpublish„:
            $sql = „UPDATE bazaar_item SET published =
,0' WHERE item_id = $item_id LIMIT 1„;
            // execute SQL
            mysqli_query($dbc, $sql);
            // confirm executed command
            echo „<p> The item <strong>' . $name_of_it
em . ,</strong> with id <strong>' . $item_id . ,</strong> was
succesfully unpublished. </p>“;
            break;
        case „delete„:
            $sql = „DELETE FROM bazaar_item WHERE item

```

```

_id = $item_id LIMIT 1";
        // execute SQL
        mysqli_query($dbc, $sql);
        // confirm executed command
        echo ,<p> The item <strong>' . $name_of_it
em . ,</strong> with id <strong>' . $item_id . ,</strong> was
succesfully deleted from listening on bazaar. </p>';
        break;
    }

    // close database connection
    mysqli_close($dbc);

} else {
    echo ,<p class="alert alert-
danger" > The selected operation cannot be performed. </p>';
}
} else if (isset($item_id) && isset($price_eur) && iss
et($name_of_item) && isset($published) && isset($screenshot1)
) {
    echo ,<h5>Are you sure perform selected operation
with bazaar item? </h5>';
    // show short describtion of score for deletion
    $image_location = IMAGE_PATH.$screenshot1;
    echo ,<p> <strong> Item_id: </strong> , . $item_id
. ,<br> <strong> Name: </strong>' . $name_of_item .

    ,</p>';
    echo " <img src=\""$image_location\" alt=\" score image
\" height=\"150\"> „;

    //generating removing/ publishing/ unpublishing
confirmation form

    echo ,<form method="POST" action="removeitem.php">
,; //not self but direct this script removecategory.php – we
dont want include any GET data tahat previously send

```

```

        echo ,<h4> Please select your operation </h4>';
        echo ,<input list="operation" name="operation" placeholder="select" >';
        echo ,<datalist id="operation">';
        echo ,<option value="publish">';
        echo ,<option value="unpublish">';
        echo ,<option value="delete">';
        echo ,</datalist>';

        echo ,<br><br>';

        echo ,<input type="radio" name="confirm" value="Yes" /> Yes ,;
        echo ,<input type="radio" name="confirm" value="No" checked="checked" /> No <br><br>';

        echo ,<input type="hidden" name="item_id" value=",$item_id.'" />';
        echo ,<input type="hidden" name="price_eur" value=",$price_eur.'" />';
        echo ,<input type="hidden" name="name_of_item" value=",$name_of_item.'" />';
        echo ,<input type="hidden" name="published" value=",$published.'" />';
        echo ,<input type="hidden" name="screenshot1" value=",$screenshot1.'" />';
        echo ,<input type="hidden" name="screenshot2" value=",$screenshot2.'" />';
        echo ,<input type="hidden" name="screenshot3" value=",$screenshot3.'" />';
        echo ,<input type="submit" class="btn btn-danger" value="submit" name="submit" />';
        echo ,</form>;

    };
    echo ,<br><br>';
    echo ,<p> <a href = „admin.php"> &lt;&lt; Back to admin page. </a></p>';
?>

```

```
</div>
```

```
    <div class="footer">
        <a class="navbar-
brand" href="https://cdesigner.eu"> Visit us on CDesigner.eu <
/a>
    </div>
```

```
</body>
</html>
```

Visual look of confirmation part with operation selection is displayed here:

→ Bazaar admin page → return to main shop page

Please select what you will do.



Are you sure perform selected operation with bazaar item?

Item_id: 31

Name: Computer science - used - good quality



Please select your operation

☐ Yes ☒ No

[<< Back to admin page.](#)

Visit us on CDesigner.eu

removeitem.php script confirmation dialog with option

Conclusion and further thoughts

Our admin page contains baseline for page administration, in next times we can add some parts for admin or moderation if needed.

Full code of bazaar app can be obtained from github here.

Bazaar – php example code – part 4 – Selling item script

Our special Christmass article focus on listening item for sell on bazaar page. Data are obtained from registered user, inserted into a database and because default published filed in bazaar_item table is false, waiting for admin approval. In final version, this page is available only for registered user.

Expectation from sellingitem script

In final applicaton, selling page is available only for common registered user. User must be able do these things:

- decribe name and price for selling item
- add up to three different photos
- select category with subcategory
- provide any further describtion for selling item
- item is not listed for sell untip approval of admin of the page

After sucessful post for sell, short describtion about adding document to a database is generated in pottom part of form.

After finalizing other main parts of app responsible for user

management, we will add column for marking selling by users_id for difrerentiating of the seller. When someone decide to buy any their item, seller must be notified about buy and must fullfil item delivery.

Visual look of the page

Sell item page consist from sell form and list of published item for sell. In the middle part, ist optionally after succesfull post displayed short info about user action.



* Please provide name of selling item:

* Please provide price for item in €:

* Select main category-subcategory for proper item listing on bazar pages:

If no proper category-subcategory exist, please contact admin of the pages for creation them for you.

In this part you can select upto 3 pictures of the product. First picture is required!

* Please select location of your score screenshot from drive - max 5MB!

Screenshot1 - required:

[Browse](#)

Screenshot2 - optional:

[Browse](#)

Screenshot3 - optional:

[Browse](#)

* Item description:

[ADD ITEM FOR SELL](#)[RESET FORM](#)[RESET FORM](#)

Chart of benchmark results

id	Name	Price	Category	Screenshot1
31	Computer science - used - good quality	55	24	
30	Analyse for e-commerce	350	29	
29	Garden scissors - green small	7.50	7	

Implementation of functionality

Main part of page consist form form part. In this part is insered script for generating available category list obtained

from bazaar_category table.

Separate par enable selecting up to three files for transfer. First screenshot of product is required and can not be omitted. Other two are optional as you can see. Separate Javascript code is responsible for listening file path after selecting them into a dialog field.

Next output show content of the form:

```
<form    enctype=„multipart/form-data“    method=“post“
action=“<?php echo $_SERVER[„PHP_SELF“]; ?>“> // because data
file transfer enctype must be defined in form tag!!!
    <input type=“hidden“ name=“MAX_FILE_SIZE“ value=“5242880
">
        <div class=“form-group“>
            <label>* Please provide name of selling item:</l
abel>
                <input type=“text“ onfocus=“this.value='<?php ec
ho isset($_POST[„name_of_item“]) ? $name_of_item : “; ?>'" nam
e=“name_of_item“                class=“form-
control“ value=“<?php echo isset($_POST[„name_of_item“]) ? $na
me_of_item : „Name for product“; ?>“>

                <label>* Please provide price for item in €:</la
bel>
                    <input type=“text“ onfocus=“this.value='<?php ec
ho isset($_POST[„price_eur“]) ? $price_eur : “; ?>'" name=“pri
ce_eur“                class=“form-
control“ value=“<?php echo isset($_POST[„price_eur“]) ? $price
_eur : „Price in €“; ?>“>

                    <!-- slection of category and subcategory -->
                        <label>* Select main category-
subcategory for proper item listing on bazar pages:</label>
                            <input list=“category_subcategory“ name=“categor
y_subcategory“ >
                                <datalist id=“category_subcategory“> <!-- must
```

be converted in subcategory_id in script – marked with (*) →
<?php // here read data from mysql bazaar_
category and display existing category where subcategory will be
nested

```
$dbc = mysqli_connect(DB_HOST, DB_USER
, DB_PW, DB_NAME);

// Check connection
if($dbc === false){
    die(„ERROR: Could not connect
to database. “ . mysqli_connect_error());
};

// create SELECT query for category
y names from database
$sql = „SELECT DISTINCT category,
subcategory FROM bazaar_category ORDER BY category ASC, subcat
egory ASC„;

// execute sql and populate data l
ist with existing category in database
if($output = mysqli_query($dbc, $s
ql)){
    if(mysqli_num_rows($output) >
0){ // if any record obtained from SELECT query

        while($row = mysqli_fetch_
array($output)){ //next rows outputed in while loop

            echo „<option valu
e=“ . $row[,category'] .“-„.$row[,subcategory'] . „>„;

        }

        // Free result set
        mysqli_free_result($output
);

    } else{
```



```

        <input type="file" name="screenshot2" class="custom-
file-input" id="screenshot2" lang="en" >
        <label class="custom-file-label2 custom-file-
label" for="customFile">Screenshot2 – optional:</label>

```

```

    </div>
    <div class="custom-file">
        <input type="file" name="screenshot3" class="custom-
file-input" id="screenshot3" lang="en" >
        <label class="custom-file-label3 custom-file-
label" for="customFile">Screenshot3 – optional:</label>

```

```

    </div>
    <script type="application/javascript"> // javasc
ript handling chaging filename of selected file
    $(document).ready(function(){
        $(„#screenshot1“).change(function(){
            //alert(„A file 1 has been selected.“);
            var thefile1 = document.getElementById(„sc
reenshot1“);

            var fileName1 = thefile1.value;
            //var fileName1 = „A file 1 has been selec
ted.“;

            $(„.custom-file-label1“).html(fileName1);

        });
        $(„#screenshot2“).change(function(){
            //alert(„A file 2 has been selected.“);
            var thefile2 = document.getElementById(„sc
reenshot2“);

            var fileName2 = thefile2.value;
            //var fileName2 = „A file 2 has been selec
ted.“;

            $(„.custom-file-label2“).html(fileName2);

        });
        $(„#screenshot3“).change(function(){

```

```

        //alert(„A file 3 has been selected.“);
        var thefile3 = document.getElementById(,screen
shot3');

        var fileName3 = thefile3.value;
        //var fileName3 = „A file 3 has been selected.“;

        $(, .custom-file-label3').html(fileName3);
    });
});

```

</script>


```

<div class="form-group">
    <label>* Item description:</label> <!-- textarea for
input large text -->
    <textarea id="item_description" onfocus="this.value='<?php echo isset($_POST[,item_description : ,Please provide description of selling item ...'; ?>'" name="item_description" class="form-control" rows="3" cols="50"><?php echo isset($_POST[,item_description : ,Description of item for sell goes here ...'; ?></textarea>
</div>

```

```

<!-- div class="form-group">
    <label>Your message for Guestbook:</label--> <!-- textarea for input large text -->
    <!-- textarea id="postmessage" name="postmessage" class="form-control" rows="6" cols="50"><?php echo isset($_POST[,postmessage']) ? $postmessage : ,Your text goes here ...'; ?></textarea>
</div-->

```

```

<button type="submit" name="submit" class="btn btn-

```

```

warning"> Add item for sell </button>
    <button type="submit" name="reset" class="btn btn-
info"> Reset form </button>

    <!-- remove comment after implementation
    <button type="submit" name="delete" class="btn btn-
danger"> Delete recently posted score </button>
    ->
    <button type="submit" name="reset" class="btn btn-
info"> Reset form </button>
    <br><br>

    <?php    //part displaying info after succesfull adde
d subscriber into a mailinglist
        if ($is_result ) {

            echo „<br> <br>“;
            echo " <table class=\"table table-
success\"> „;

            echo " <tr>
                <td><h5> <em> Item for selll: <
/em> $name_of_item for $price_eur € </h5> <h5> has been succe
sfully added to selling list. Item will be visible
                on bazaar page after admin appr
oval. </h5> „;

                $image_location = IMAGE_PATH.$s
creenshot1;

                echo " <img src=\"\$image_location\" al
t=\" score image \" height=\"150\"> „;

            echo "        <td>    </tr> „;
            echo " </table> „;

            //echo " <input type="text" id="result_fie
ld" name="result_field" value="$result" > <br>" ;
        } ;
        ?>
        <br>

</form>

```

Part responsible for obtaining data after hitting submit button consist from:

```
if(filter_has_var(INPUT_POST, ,submit')) {  
    // Data obtained from $_postmessage are assigned to local variables  
    $name_of_item = htmlspecialchars($_POST[,name_of_item'  
]);  
    $price_eur = htmlspecialchars($_POST[,price_eur']);  
    $users_id = htmlspecialchars($_POST[,users_id']);  
    //echo ,users_id'; echo $users_id;  
  
    $category_subcategory = htmlspecialchars($_POST[,category_subcategory']); // must be converted to subcategory_id (*)  
    // separate category and subcategory with strtok()  
function  
    $words = explode(-, , $category_subcategory);  
    $category=$words[0];  
    //echo $category;  
    //echo ,<br>';  
    $subcategory=$words[1];  
    //echo $subcategory;  
  
    $screenshot1 = htmlspecialchars($_FILES[,screenshot1'  
[,name']]);  
    $screenshot2 = htmlspecialchars($_FILES[,screenshot2'  
[,name']]);  
    $screenshot3 = htmlspecialchars($_FILES[,screenshot3'  
[,name']]);  
    $item_description = htmlspecialchars($_POST[,item_description']]);  
    // (*) – conversion of category and subcategory into category%id  
    $dbc = mysqli_connect(DB_HOST, DB_USER, DB  
_PW, DB_NAME);  
    // Check connection  
    if($dbc === false){
```



```

        die(„ERROR: Could not connect to datab
ase. “ . mysqli_connect_error());
    };

    // create SELECT query for category names
from database
    $sql = „SELECT subcategory_id FROM bazaar_
category WHERE category = „.$category'“. “ AND subcategory =
    „.$subcategory'“ ;
    // execute sql and populate data list with
    existing category in database
    if($output = mysqli_query($dbc, $sql)){
        if(mysqli_num_rows($output) > 0){ //
if any record obtained from SELECT query
            while($row = mysqli_fetch_array($o
utput)){ //next rows outputed in while loop

                $subcategory_id = $row[,subcat
egory_id'] ;

            }

            // Free result set
            mysqli_free_result($output);
        } else {
            echo „There is no souch category-
subcategory in category table. Please correct your error.“; //
            if no records in table
            }
        } else{
            echo „ERROR: Could not able to execute
$sql. “ . mysqli_error($dbc); // if database query problem
        }
        // Close connection
        mysqli_close($dbc);

        // Controll if all required fields was written
        if(!empty($name_of_item) && !empty($price_eur) && !emp

```

```

ty($subcategory_id) && !empty($screenshot1)) { // these item i
dentifiers are mandatory and can not be empty
    // If check passed – all needed fields are written
    // Check if E-mail is valid

        // move image to /images final folder from tem
porary download location
        $target1 = IMAGE_PATH . $screenshot1;
        $target2 = IMAGE_PATH . $screenshot2;
        $target3 = IMAGE_PATH . $screenshot3;
        // !!! Add entry to the database and redraw al
l score in chart list descending from highest score
        // insert into databse
        if (move_uploaded_file($_FILES[,screenshot1'][,tmp_name'], $target1)) {
            move_uploaded_file($_FILES[,screenshot2'][,tmp_name'], $target2);
            move_uploaded_file($_FILES[,screenshot3'][,tmp_name'], $target3);
            // make database connection
            $dbc = mysqli_connect(DB_HOST, DB_
USER, DB_PW, DB_NAME);
            // Check connection
            if($dbc === false){
                die(„ERROR: Could not conn
ect to database. “ . mysqli_connect_error());
            }

            // INSERT new entry

            $sql = „INSERT INTO bazaar_item (n
ame_of_item, price_eur, subcategory_id, users_id, item_add_dat
e, screenshot1, screenshot2, screenshot3, item_description)
            VALUES (,$name_of_item', $price_eu
r , ,,$subcategory_id' , ,,$users_id' , now(), ,,$screenshot1', ,
,$screenshot2', ,,$screenshot3', ,,$item_description' )“;
            //show added item true
            $is_result = true;
            if(mysqli_query($dbc, $sql)){

```

```

                                $msg = ,New item ,.$name_of_it
em. , for ,. $price_eur. , € succesfully added to sell item –
waiting for admin approvall.';
                                $msgClass = ,alert-success';
                                } else {

                                $msg = „ERROR: Could not able
to execute $sql. “ . mysqli_error($dbc);
                                $msgClass = ,alert-danger';
                                }
                                // end connection
                                mysqli_close($dbc);

                                } else {
                                    // Failed –if not all fields are
fullfiled
                                $msg = ,Please fill in all * marke
d contactform fields';
                                $msgClass = ,alert-
danger'; // bootstrap format for allert message with red color
                                };
                                };
};

```

Conclusion and other thoughts

Our page must be updated with code displaying its content and menus differently for logged in user or anonymous users. Next improvement is expanding/ altering code to mark added item by users_id from active session for distinguishing between submitters.

Full code of application in as is current state can be obtained for further study from [here](#).

Bazaar – php example code – part 3 – Manage category script

Article focus on way how we populate category/ subcategory for selling items. Management page contains listing of successfully added files with removal link as option. Our project will lead from category, through item, simple login functionality into a editprofile option with more robust user login.

Discuss managecategory.php functionality

Our managecategory.php script must enable these functionalities:

- create subcategory of item with existing main category
- if we creating subcategory with non existing main category, we must be able to add them along
- after all successfully added category, list of existing category/ subcategory must be displayed with link for removal appropriate category row
- after preparing parts for login verification, these scripts can be viable only for user with admin role

assigned by page admin

Visual content of this page along with removecategory.php script follow:

Bazaar - management of selling category

New categorybookssucesfully added into a bazaar_category table.

* Set name for new subcategory:

computer

* Select main category for nesting created subcategory:

CREATE NEW SUBCATEGORY

RESET

List of active categories and subcategories

| subcategory_id | category | subcategory | delete category |
|----------------|------------|-------------|-------------------------|
| 8 | autos | van | Manage entry: >> Remove |
| 24 | books | computer | Manage entry: >> Remove |
| 11 | electronic | DVD players | Manage entry: >> Remove |
| 6 | electronic | videos | Manage entry: >> Remove |
| 7 | garden | scissors | Manage entry: >> Remove |
| 16 | networking | routers | Manage entry: >> Remove |

Visit us on CDesigner.eu

Visual look of managecategory.php script

Confirmation of deletion selected category removal.



Are you sure to delete the next category item from bazaar?

subcategory_id: 8
Category: autos
Subcategory: van

☐ Yes ☒ No

<< Back to category management page.

Category removal script confirmation dialog

Implementation of expected functionality

Our page consist from parts for obtain user category and subcategory names. Error messaging, reporting of successfull category addition into bazaar_category table are next parts of our script.

Form obtaining data from user follow next:

```
<form method="post" action="<?php echo $_SERVER[,PHP_SELF'];  
?>,">  
    <div class="form-group">  
        <label>* Set name for new subcategory:</label>  
        <input type="text" onfocus="this.value='<?php ec  
ho isset($_POST[,subcategory']) ? $subcategory : "; ?>' " name=  
"subcategory" class="form-  
control" value="<?php echo isset($_POST[,subcategory']) ? $sub  
category : ,Please provide name of new subcategory'; ?>">
```

```

        <br>
        <label>* Select main category for nesting create
d subcategory:</label>
        <input list="category" name="category" >
        <datalist id="category">
            <?php // here read data from mysql bazaar_
category and display existing category whre subcategory will b
e nested

            $dbc = mysqli_connect(DB_HOST, DB_USER
, DB_PW, DB_NAME);

            // Check connection
            if($dbc === false){
                die(„ERROR: Could not connect
to database. “ . mysqli_connect_error());
            };

            // create SELECT query for categor
y names from database
            $sql = „SELECT DISTINCT category F
ROM bazaar_category ORDER BY category ASC„;

            // execute sql and populate data l
ist with existing category in database
            if($output = mysqli_query($dbc, $s
ql)){
                if(mysqli_num_rows($output) >
0){ // if any record obtained from SELECT query

                    while($row = mysqli_fetch_
array($output)){ //next rows outputed in while loop

                        echo „<option valu
e=" . $row[„category“] . „>„;

                    }

```

```

// Free result set
mysqli_free_result($output
);

} else{
    echo „There is no category
in category table. Please wirite one.“; // if no records in t
able
    }
} else{
    echo „ERROR: Could not able to
execute $sql. “ . mysqli_error($dbc); // if database query pr
oblem
    }

// Close connection
mysqli_close($dbc);
?>

```

```

</datalist>
<br>

```

```

<button type="submit" name="subcategorysubmit" c
lass="btn btn-warning"> Create new subcategory </button>
    <input type="reset" class="btn btn-
info" value="Reset">
</div>
<hr>
</form>

```

```

<br><br>

```

```

<?php //part displaying info after succesfull
added category into a mailinglist
    if ($is_result ) {

```



```

        echo „<br> <br>“;
        echo “ <table class=“table table-
success\“> „;

        echo “ <tr>
            <td><h5> <em> Category: </em> $
category with subcategory $subcategory </h5> <h5> has been suc
cesfully added to category list </h5> „;

        echo “        <td>    </tr> „;
        echo “ </table> „;

        //echo “ <input type=“text“ id=“result_fie
ld“ name=“result_field“ value=“$result“ > <br>“ ;
    } ;
    ?>
    <br>

</form>
<?php

```

Listening of all items in bazaar_category table is generated by this content of script

```

$dbc = mysqli_connect(DB_HOST, DB_USER, DB_PW, DB_NAME);
// Check connection
if($dbc === false){
    die(„ERROR: Could not connect to database – stage of artic
le listing. “ . mysqli_connect_error());
}

// read all rows (data) from guestbook table in „test“ databas
e
$sql = „SELECT * FROM bazaar_category ORDER BY category ASC, s

```

```

ubcategory ASC,,;
/*****
*****/
/*      Output in Table – listening all category in
bazaar_category table      */
/*****
*****/
// if data properly selected from guestbook database table
echo „<h4>List of active categories and subcategories</h4>“;
echo „<br>“;
echo „    ,    <button    class=“btn    btn-secondary    btn-
lg “ onclick=“location.href=\\‘admin.php\\‘“ type=“button“>    adm
in page -> </button>‘;
echo „<br>“; echo „<br>“;
    if($output = mysqli_query($dbc, $sql)){
        if(mysqli_num_rows($output) > 0){ // if any record ob
tained from SELECT query
            // create table output
            echo „<table>“; //head of table
                echo „<tr>“;
                    echo „<th>subcategory_id</th>“;
                    echo „<th>category</th>“;
                    echo „<th>subcategory</th>“;
                    echo „<th></th>“;
                    echo „<th>delete category</th>“;

                    echo „</tr>“;
                while($row = mysqli_fetch_array($output)){ //next
rows outputed in while loop
                    echo “ <div class=\\“mailinglist\\“> “ ;
                    echo „<tr>“;
                        echo „<td>“ . $row[,subcategory_id'] . „</
td>“;

                        echo „<td>“ . $row[,category'] . „</td>“;
                        echo „<td>“ . $row[,subcategory'] . „</td>
“;

                        // removal line with removing link line

```

```

        echo „<td colspan=\“1\“> Manage entry: <
/td>“; // description on first line
        echo „<td colspan=\“1\“><a id=\“DEL\“ href=
f=\“removecategory.php?subcategory_id=\“. $row[,subcategory_id']
. ,&amp;category=
. $row[,category'] . ,&amp;subcategory=
y=\“. $row[,subcategory'] . '\“> >> Remove </a></td></tr>“; //co
nstruction of GETable link
        // for removecategory.php input

        echo „</tr>“;
        echo “ </div> “ ;
    }
    echo „</table>“;
    // Free result set
    mysqli_free_result($output);
} else{
    echo „There is no benchmark result in chart. Pleas
e wirite one.“; // if no records in table
}
} else{
    echo „ERROR: Could not able to execute $sql. “ . mysql
i_error($dbc); // if database query problem
}
// Close connection
mysqli_close($dbc);
?>

```

removecategory.php script

Remove category script obtain data from GET associative array ofered from url snippet added to name of invoked page. After reading data, cerate removal confirmation form.

After selecting yes and submitting form, appropriate category/subcategory row is removed with appropriate sql query.

Content of the page is in following part:

```
<!--
*****
* ->
<!-- PHP „self“ code GET request for remove and POST delete data
a ->
<!--
*****
* ->
<!-- Vrsion: 1.0          Date: 17.10.2020 by CDesigner.eu
->
<!--
*****
* ->
```

```
<?php // leading part of page for simple header securing and basic variable setup
```

```
    require_once('appvars.php'); // including variables for database
```

```
    session_start(); // start the session – must be added on all pages for session variable accessing – further description will be available in appropriate article in a row of articles focused on bazaar app
```

```
    // solution using SESSIONS with COOKIES for longer (30days) login persistency
```

```
    if(!isset($_SESSION['users_id'])) { // if session is no more active
        if(isset($_COOKIE['users_id']) && isset($_COOKIE['username'])) { // but cookie is set then renew session variables along them
            $_SESSION['users_id'] = $_COOKIE['users_id'];
            $_SESSION['username'] = $_COOKIE['username'];
        }
    }
```

```
// two variables for message and styling of the message with bootstrap
```

```
$msg = "";  
$msgClass = "";
```

```
// default values of auxiliary variables
```

```
?>
```

```
<!-- ***** -->  
<!-- script for appropriate scode removal -->  
<!-- ***** -->  
<!-- obtain GET data from admin.php and trough -->  
<!-- POST submit remove data from database -->  
<!-- ***** -->  
<!DOCTYPE html>  
<html>  
<head>  
    <title> Bazaar score – remove script </title>  
    <link rel="stylesheet" href="./css/bootstrap.min.css"> <!-- bootstrap mini.css file -->  
    <link rel="stylesheet" href="./css/style.css"> <!-- my local.css file -->  
    <script src="https://code.jquery.com/jquery-3.1.1.slim.min.js" integrity="sha384-A7FZj7v+d/sdmMqp/nOQwliLvUsJfDHW+k90mg/a/EheAdgtzNs3hpfag6Ed950n" crossorigin="anonymous"></script>  
    <script src="https://cdnjs.cloudflare.com/ajax/libs/tether/1.4.0/js/tether.min.js" integrity="sha384-DztdAPBWPRXSA/3eYEEUWrWCy7G5KFbe8fFjk5JAIxUYHKkDx6Qin1DkWx51bBrb" crossorigin="anonymous"></script>  
  
</head>  
<body>  
    <nav class="navbar navbar-default">  
        <div class="container">  
            <div class="navbar-header">  
                <a class="navbar-brand" href="managecategory.php">Bazaar category manager – par
```

t for Bazaar category management

[-> return to main shop page](index.php)

</div>

</div>

</nav>

<div class="container" id="formcontainer">

<?php if(\$msg != ""): ?> <!-- alert showing part ->

<div class="alert <?php echo \$msgClass; ?>"><?php echo \$msg; ?></div>

<?php endif; ?>

 <!-- logo on the center of the page ->

<h4>Confirmation of deletion selected category removal.</h4>

 <!-- logo on the center of the page ->

<?php // code for GET info about what to remove and submit removing approval

if(isset(\$_GET[subcategory_id,]) && isset(\$_GET[category,])) {

// take a data from GET link generated by adminscript

\$subcategory_id = htmlspecialchars(\$_GET[subcategory_id']);

\$category = htmlspecialchars(\$_GET[category']);

\$subcategory = htmlspecialchars(\$_GET[subcategory']);

```

        } else if (isset($_POST[,subcategory_id']) && isset($_
POST[,category']) && isset($_POST[,subcategory'])) { //grab sc
ore from POST – different behavior for removal
        $subcategory_id = htmlspecialchars($_POST[,subcate
gory_id']);
        $category = htmlspecialchars($_POST[,category']);
        $subcategory = htmlspecialchars($_POST[,subcategor
y']);

    } else { //error info message
        echo ,<p class="alert alert-
danger"> Please specify any category for removal. </p>';

    };

    if(isset($_POST[,submit,])){

        if($_POST[,confirm,] == ,Yes, ){ // delete appropri
ate score post with imagescreenshot
        //delete the screenshotimage from the
        $subcategory_id = htmlspecialchars($_POST[,subca
tegory_id']);
        $category = htmlspecialchars($_POST[,category ,
]);
        $subcategory = htmlspecialchars($_POST[,subcateg
ory']);

        // conect to the database
        $dbc = mysqli_connect(DB_HOST, DB_USER, DB_PW, D
B_NAME);

        //Delete score data from the database
        $sql = „DELETE FROM bazaar_category WHERE subcat
egory_id = $subcategory_id LIMIT 1”;

```

```

// execute SQL
mysqli_query($dbc, $sql);

// close database connection
mysqli_close($dbc);

// confirm executed command
echo ,<p> The category <strong>' . $category . ,
</strong> with id <strong>' . $subcategory_id . ,</strong> was
succesfully removed. </p>';

    } else {
        echo ,<p class="alert alert-
danger" > The selected category was not removed. </p>';
    }
} else if (isset($subcategory_id) && isset($category)
) {
    echo ,<h5>Are you sure to delete the next category
item from bazaar? </h5>';
    // show short describtion of score for deletion
    echo ,<p> <strong> subcategory_id: </strong> , . $
subcategory_id . ,<br> <strong> Category: </strong>' . $categ
ory .
        ,<br> <strong> Subcategory: </strong>' . $sub
category .
        ,</p>';

    //generating removing confirmation form

    echo ,<form method="POST" action="removecategory.p
hp">,; //not self but direct this script removecategory.php
– we dont want include any GET data tahat previously send
    echo ,<input type="radio" name="confirm" value="Ye
s" /> Yes ,;
    echo ,<input type="radio" name="confirm" value="No
" checked="checked" /> No <br><br>';

```



```

        echo ,<input type="hidden" name="subcategory_id" value=",$subcategory_id.'" />';
        echo ,<input type="hidden" name="category" value=",$category.'" />';
        echo ,<input type="hidden" name="subcategory" value=",$subcategory.'" />';
        echo ,<input type="submit" class="btn btn-danger" value="submit" name="submit" />;
        echo ,</form>;

    };
    echo ,<br><br>';
    echo ,<p> <a href = „managecategory.php“> &lt;&lt; Back to category management page. </a></p>';

?>

</div>

<div class="footer">
                                <a class="navbar-brand" href="https://cdesigner.eu"> Visit us on CDesigner.eu </a>
</div>

</body>
</html>

```

Conclusion and other thoughts

Our code snippets create baseline for required functionality for category management. From final code was omitted parts related to user login and logged users persisency. Full bazaar

app code can be obtained from github [here](#).

Bazaar – php example code – part 2 – structure of database tables

Article focus on database structure design for Bazaar app. Process of normalization and atomisation of user data is important for minimisation of redundant data in database tables.

Normalisation of the database tables

For our first approach to establishing database structure also known as describing database schema we need explain some terms.

Schema – or database schema is description of data (tables and columns) in database, along with related objects and connections among them (relations).

For interconnection of separate table are used relations created with pair primary keys and foreign keys.

For further readings about database relationship, please visit

page.

Normalization – means designing database with reduction of duplicate data as possible with separation data into tables with relations between them.

These are the steps leading to normalised database structure:

- Question yourself: What is the main thing you want your table to be about?
- List all information you need to know about one thing in appropriate table.
- Break down all information about that thing into pieces for further organizing the table.

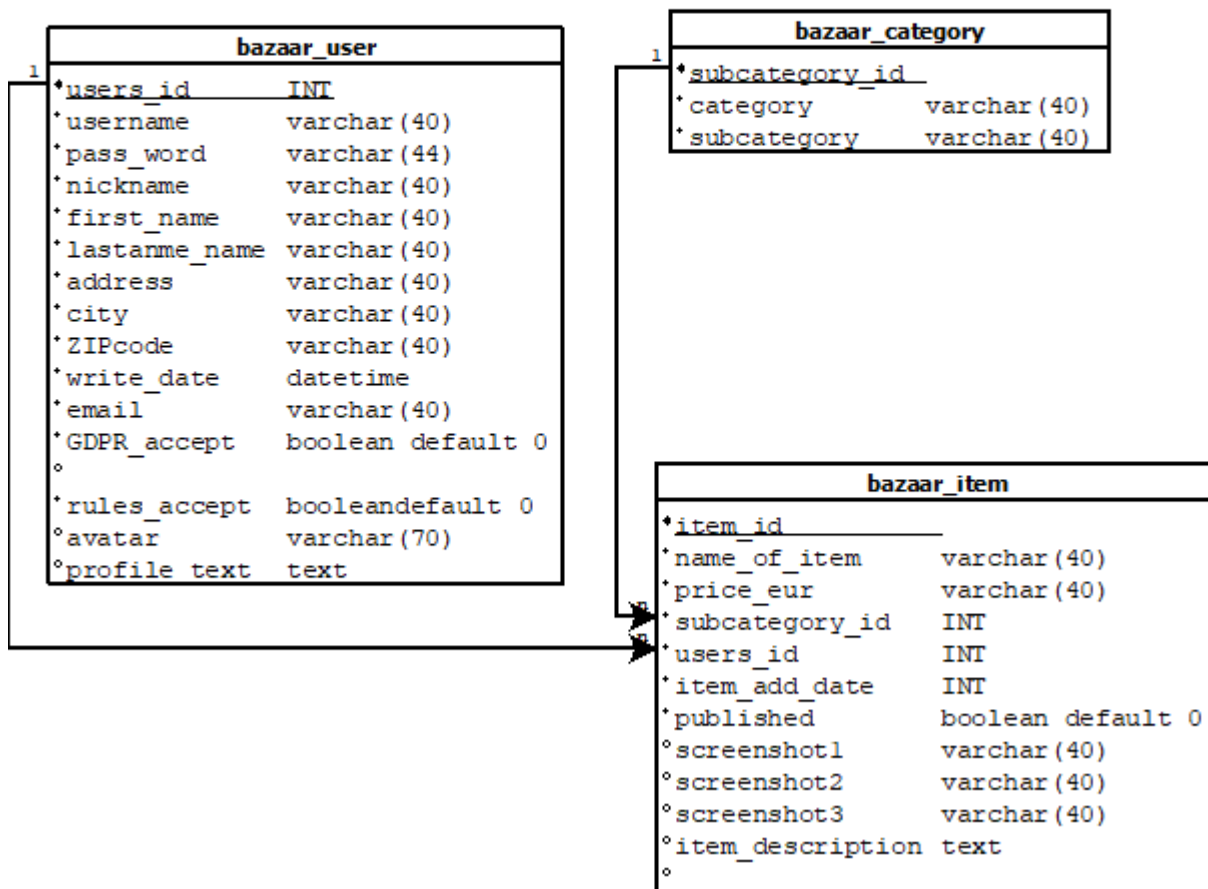
Basic concept focus on term of the atomic data. Atomic data are data that has been broken down into the smallest form needed for a database.

For further readin about data normalization please visit wikipedia page [here](#).

Schema of bazaar table

Our design is break down into a four table, bazaar_user, bazaar_itme and bazaar_category. Relations between them is shown on text picture.

Bazaar app database schema



Bazaar app database schema drawing created in Dia editor

For simplifieng database creation process we prepared our first createdatabase.php script as follows:

```

<!--
*****
* ->
<!-- PHP code for automation of preparation databasetable for
bazaar app ->
<!--
*****
** ->
  
```

<!-- Vrsion: 1.0 Date: 10.10.2020 by CDesigner.eu

→

<!--

** →

<?php // script for accessing database and first table structure establishment

require_once('appvars.php'); // including variables for database

/* Attempt MySQL server connection. Assuming you are running MySQL

server with (user 'admin' with password test*555) */

\$dbc = mysqli_connect(DB_HOST, DB_USER, DB_PW, DB_NAME);

// Check connection

if(\$dbc === false){

 die('ERROR: Could not connect to database. ' . mysqli_connect_error());

}

// Attempt create table query execution

\$sql1 = „CREATE TABLE bazaar_user (

 users_id INT NOT NULL PRIMARY KEY AUTO_INCREMENT,

 username VARCHAR(40) NOT NULL,

 pass_word VARCHAR(44) NOT NULL,

 nickname VARCHAR(40) NOT NULL UNIQUE, /* not two identical nicknames allowed*/

 first_name VARCHAR(40) NOT NULL,

 lastname_name VARCHAR(40) NOT NULL,

 addresss VARCHAR(40) NOT NULL,

 city VARCHAR(40) NOT NULL,

 ZIPcode VARCHAR(40) NOT NULL,

 write_date DATETIME NOT NULL,

 email VARCHAR(70) NOT NULL , /* not UNIQUE emails because one user can submit different benchmark results */

 /* message_text TEXT */ /* optionally add boolean fields for subscription */

 GDPR_accept BOOLEAN NOT NULL default 0, /* BOOLEAN value if user accepted GDPR */

 rules_accept BOOLEAN NOT NULL default 0, /* BOOLEAN value

```

if user accepted portal rules */
    avatar VARCHAR(70),                                /* link to image
*/
    profile_text TEXT                                    /* submit text fro
m publisher */

);
$sql2 = „CREATE TABLE bazaar_item (
    item_id INT NOT NULL PRIMARY KEY AUTO_INCREMENT,
    name_of_item VARCHAR(40) NOT NULL,
    price_eur VARCHAR(40) NOT NULL,
    subcategory_id INT NOT NULL,
    users_id INT NOT NULL,
    item_add_date DATETIME NOT NULL,
    published BOOLEAN NOT NULL default 0,
    screenshot1 VARCHAR(70),                            /* link to
image of item 1 */
    screenshot2 VARCHAR(70),                            /* link to
image of item 2 */
    screenshot3 VARCHAR(70),                            /* link to
image of item 3 */
    item_description TEXT,                               /* item desc
ription */
    CONSTRAINT FK_subcategory_id FOREIGN KEY (subcategory_id)
REFERENCES bazaar_category(subcategory_id), /* foreign key N s
ite of 1 to N relation */
    CONSTRAINT FK_users_id FOREIGN KEY (users_id) REFERENCES
bazaar_user(users_id) /* foreign key N site of 1 to N relation
*/
);
$sql3 = „CREATE TABLE bazaar_category (
    subcategory_id INT NOT NULL PRIMARY KEY AUTO_INCREMENT,
    category VARCHAR(40) NOT NULL,
    subcategory VARCHAR(40) NOT NULL

);
echo „<h2>Processing database tables for bazaar app.</h2>“;
echo
*****
,;
echo „<br>“;

```

```

echo , |      PHP   code for automation of preparation databaseta
ble for bazaar app      | ,;
echo „<br>“;
echo
*****
**** | ,;
echo „<br>“;
echo , | Vrsion: 1.0      Date: 10.10.2020 by CDesigner.eu
      | ,;

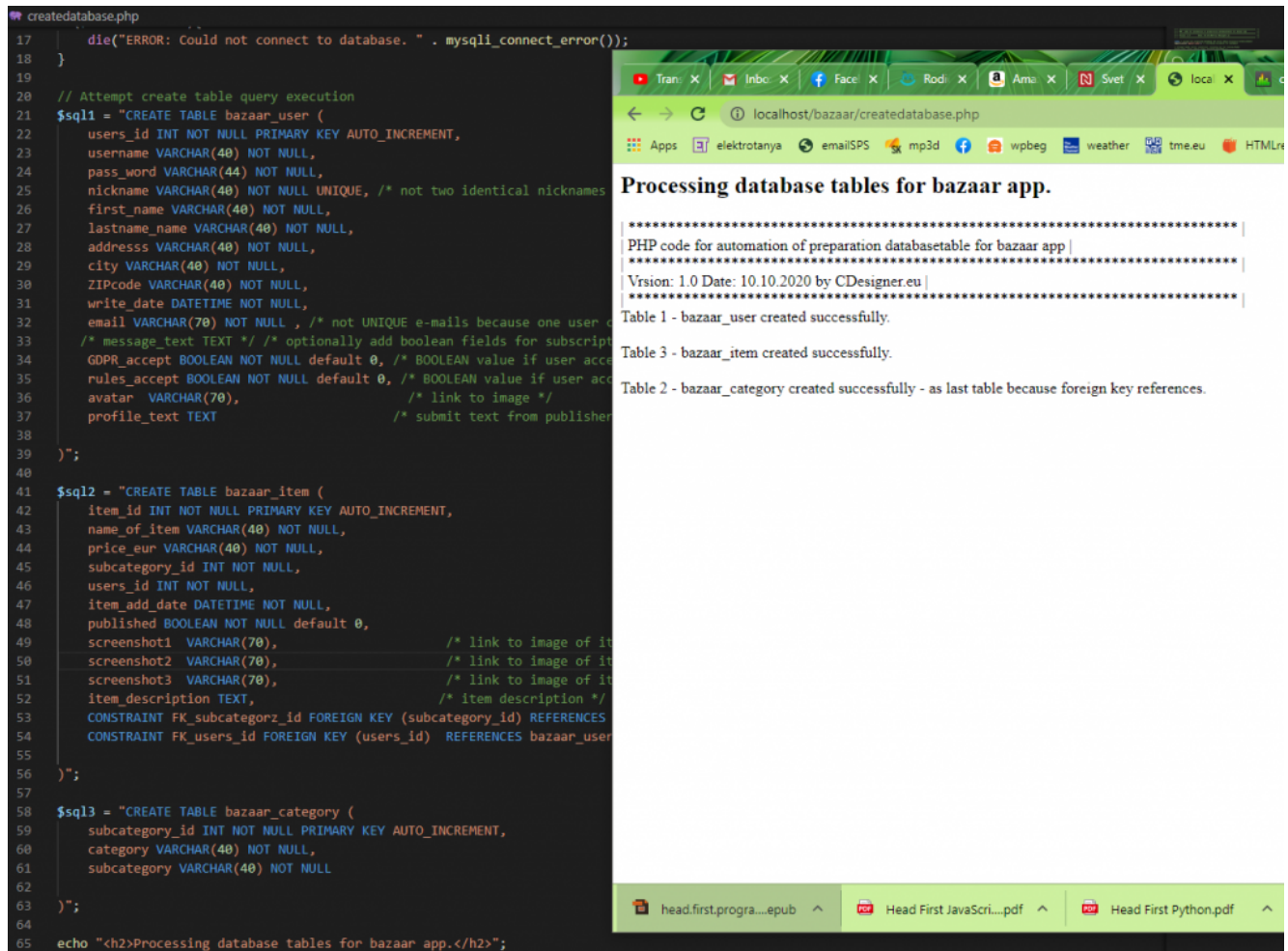
echo „<br>“;
echo
*****
***** | ,;
echo „<br>“;
if(mysqli_query($dbc, $sql1)){
    echo „Table 1 – bazaar_user created successfully.“;
    echo „<br><br>“;
} else{
    echo „ERROR: Could not able to execute $sql. “ . mysqli_er
ror($dbc);
    echo „<br><br>“;
};
if(mysqli_query($dbc, $sql3)){
    echo „Table 3 – bazaar_item created successfully.“;
    echo „<br><br>“;
} else{
    echo „ERROR: Could not able to execute $sql. “ . mysqli_er
ror($dbc);
    echo „<br><br>“;
};
if(mysqli_query($dbc, $sql2)){
    echo „Table 2 – bazaar_category created successfully – as
last table because foreign key references.“;
    echo „<br><br>“;
} else{
    echo „ERROR: Could not able to execute $sql. “ . mysqli_er
ror($dbc);
    echo „<br><br>“;
}

// Close connection

```

```
mysqli_close($dbc);  
?>
```

After succesfull run of this script, we obtained these output in phpmyadmin page:



```
17 die("ERROR: Could not connect to database. " . mysqli_connect_error());  
18 }  
19  
20 // Attempt create table query execution  
21 $sql1 = "CREATE TABLE bazaar_user (  
22     users_id INT NOT NULL PRIMARY KEY AUTO_INCREMENT,  
23     username VARCHAR(40) NOT NULL,  
24     pass_word VARCHAR(44) NOT NULL,  
25     nickname VARCHAR(40) NOT NULL UNIQUE, /* not two identical nicknames  
26     first_name VARCHAR(40) NOT NULL,  
27     lastname_name VARCHAR(40) NOT NULL,  
28     addresss VARCHAR(40) NOT NULL,  
29     city VARCHAR(40) NOT NULL,  
30     ZIPcode VARCHAR(40) NOT NULL,  
31     write_date DATETIME NOT NULL,  
32     email VARCHAR(70) NOT NULL, /* not UNIQUE e-mails because one user c  
33     /* message_text TEXT */ /* optionally add boolean fields for subscript  
34     GDPR_accept BOOLEAN NOT NULL default 0, /* BOOLEAN value if user acc  
35     rules_accept BOOLEAN NOT NULL default 0, /* BOOLEAN value if user acc  
36     avatar VARCHAR(70), /* link to image */  
37     profile_text TEXT /* submit text from publisher  
38 );  
39  
40  
41 $sql2 = "CREATE TABLE bazaar_item (  
42     item_id INT NOT NULL PRIMARY KEY AUTO_INCREMENT,  
43     name_of_item VARCHAR(40) NOT NULL,  
44     price_eur VARCHAR(40) NOT NULL,  
45     subcategory_id INT NOT NULL,  
46     users_id INT NOT NULL,  
47     item_add_date DATETIME NOT NULL,  
48     published BOOLEAN NOT NULL default 0,  
49     screenshot1 VARCHAR(70), /* link to image of it  
50     screenshot2 VARCHAR(70), /* link to image of it  
51     screenshot3 VARCHAR(70), /* link to image of it  
52     item_description TEXT, /* item description */  
53     CONSTRAINT FK_subcategoryz_id FOREIGN KEY (subcategory_id) REFERENCES  
54     CONSTRAINT FK_users_id FOREIGN KEY (users_id) REFERENCES bazaar_user  
55 );  
56  
57  
58 $sql3 = "CREATE TABLE bazaar_category (  
59     subcategory_id INT NOT NULL PRIMARY KEY AUTO_INCREMENT,  
60     category VARCHAR(40) NOT NULL,  
61     subcategory VARCHAR(40) NOT NULL  
62 );  
63  
64  
65 echo "<h2>Processing database tables for bazaar app.</h2>";
```

Processing database tables for bazaar app.

PHP code for automation of preparation databasetable for bazaar app |
Version: 1.0 Date: 10.10.2020 by CDesigner.eu |

Table 1 - bazaar_user created successfully.
Table 3 - bazaar_item created successfully.
Table 2 - bazaar_category created successfully - as last table because foreign key references.

Output of database creational script

The screenshot shows the phpMyAdmin interface with the 'bazaar_item' table selected. The table structure is as follows:

| # | Názov | Typ | Zotriedenie | Atribúty | Nulový | Predvolené | Komentáre | Extra | Akcia |
|----|------------------|-------------|-------------------|----------|--------|------------|-----------|----------------|-----------------------|
| 1 | item_id | int(11) | | | Nie | Žiadny | | AUTO_INCREMENT | Zmeniť Odstrániť Viac |
| 2 | name_of_item | varchar(40) | latin1_swedish_ci | | Nie | Žiadny | | | Zmeniť Odstrániť Viac |
| 3 | price_eur | varchar(40) | latin1_swedish_ci | | Nie | Žiadny | | | Zmeniť Odstrániť Viac |
| 4 | subcategory_id | int(11) | | | Nie | Žiadny | | | Zmeniť Odstrániť Viac |
| 5 | users_id | int(11) | | | Nie | Žiadny | | | Zmeniť Odstrániť Viac |
| 6 | item_add_date | datetime | | | Nie | Žiadny | | | Zmeniť Odstrániť Viac |
| 7 | published | tinyint(1) | | | Nie | 0 | | | Zmeniť Odstrániť Viac |
| 8 | screenshot1 | varchar(70) | latin1_swedish_ci | | Áno | NULL | | | Zmeniť Odstrániť Viac |
| 9 | screenshot2 | varchar(70) | latin1_swedish_ci | | Áno | NULL | | | Zmeniť Odstrániť Viac |
| 10 | screenshot3 | varchar(70) | latin1_swedish_ci | | Áno | NULL | | | Zmeniť Odstrániť Viac |
| 11 | item_description | text | latin1_swedish_ci | | Áno | NULL | | | Zmeniť Odstrániť Viac |

Obtained database tables with appropriate files structures

Conclusion and final thoughts

Established database structure prepared our startingpoint for obtaining data from users. database was created with emphasis on normalisation and atomisation of stred data.

Full sourcecode of our project for further study and free adaptation can be obtained from github [here](#).

Bazaar – php example code –

part 1 – first thoughts about application

Bazaar app enable management of sell offers from variety of registered clients. Main project focus on database design and implementation, login implementation with SESSIONS extended with support of COOKIES, maintaining access to scripts along user roles. Our first part of series will explain basic interaction among pages and show page design expectation.

Enumeration of app expectation

Our Bazaar app will enable manage listenings for sell of used things. Functionality must be limited for registered users and some of management capability can be available only for admin user.

During process of preparation we expanded our first thoughts and adding new features as item cart or user roles.

Requirements for our app:

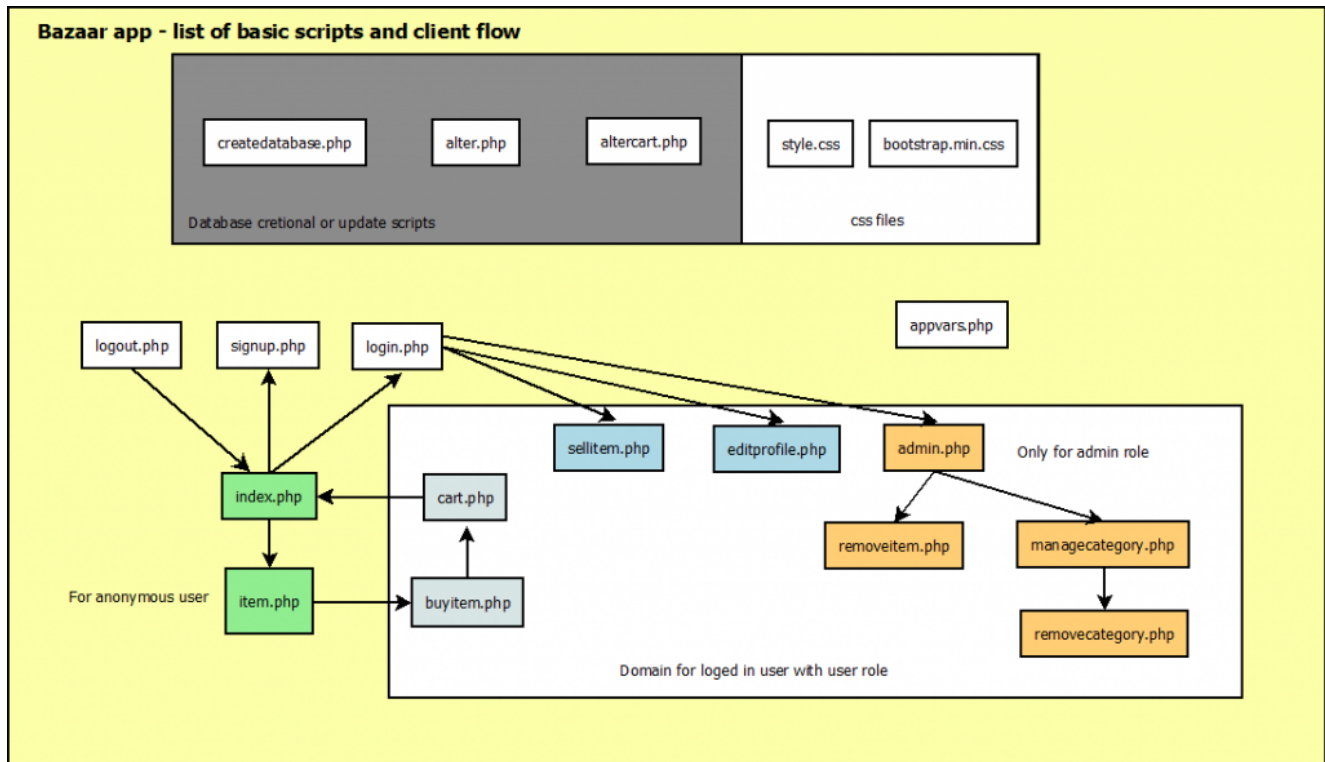
- unregistered user must be able view all sell listenings on index.php page
- list of items for sell must display last 5 products and user must be able select interesting category from drop-down menu
- for all items must be available closer look page with further display and link to buy
- only registered users can add items to a cart – for cart

management we use column `cart_number`. If cart number is 0, item is available for buy. If this number is changed to `users_id` then item is not displayed anymore and is added into a cart of user (there is some difficulty, how to after some time release listening and reset number to 0 after some expiration period)

- application will demonstrate mechanism for sinup user with login functionality
- for user login persistency combination of SESSION with COOKIES mechanism will be used
- on admin page and editprofile page is shown different content for logged in or unlogged user
- user can signin into a common user role with signup form
- only admin has granted full access to admin page
- registered user can add item to sell listening, but item is displayed only after admin approval of content
- admin can manage categories available on bazaar portal

Block diagram of main page scripts

Next picture show names of main page scripts and flow during user work. Some pages/ scripts will be added next in development process for support of new important functionality. This solution depicts traditional, procedural approach for developmnet of PHP applications.



Bazaar app – block diagram of main page scripts

Code scripts can be separated into a three main category: 1 – database creatonal scripts with further database altering scripts, 2 – cascade styles scripts and 3 – main page scripts.

Description some of them:

createdatabase.php – main script for creating three database table for category, item and user data persistency.

alter.php and altercart.php – minor updates and addins into a existing database table. Up to a time of preapartion of this article `user_role` to `bazaar_user` table was added and `cart_number` in `bazaar_item` was added for support of user cart functionality.

index.php – main page with signup, login and logout functionality. Display will differ for unlogged and logged in user. man contentent of the page are two table outputs os

listened items for sell. persistent display show 5 latest listenings. Upper part enable select form dropdown menu category of interest for appropriate user and next show all listed items. Further improvement will be pagination of results.

login.php, **logout.php** and **signup.php** – all these scripts support access management into a content of the page. Unregistered user must sign up with unique e-mail and name. Some of important data can be added into a database next with **editprofile.php** script.

item.php and **buyitem.php** – all items listened on main index page have link leading into a **item.php** page. From GET data is obtained `item_id` that select from database all information provided to listed item. On **item.php** page is link into a **buyitem.php** page. This page is available only for registered user and enable change `cart_number` data from 0 to `users_id` of appropriate user. This way we mark listed item as added into a user cart for further buy fulfillment (this option must be discussed next, at time of writing this article is not in a final stage).

cart.php – (on progress) – show all items marked with appropriate `users_id` of the logged user and show total price for all listened items. This page is a way for fulfillment buy process and send info to site admin/ seller to arrange all important things for correct buy/sell process. (This part needs further improvement of functionality, because to sell listening user must be marked in `bazaar_item` table with their `users_id`).

sellitem.php – enable add info about sold product.

editprofile.php – any further information about registered user is contained in this part of page.

admin.php with **removeitem.php** and **managecategory.php** and **removecategory.php** – access to these pages is enabled only for admin role of the user. This part enables change state of

published boolean filed from 0 to 1 for display item to sell. All page categories can be managed through these gages.

appvars.php – central page with all important constants included on top part of pages in any other scripts.

Showcase of frontend and backend for bazaar app

This part show visual look of appropriate pages in state of time whne this article will be written. Further improvements will be added in next weeks during development and testing process.

[Log In](#) [Sign Up for better membership!](#)

Bazaar - best items for a best prices!



* Select category-subcategory of product for sale:

If no proper category-subcategory exist, please contact admin of the pages for creation them for you.

[SHOW ME INTERESTING THINGS](#)

[RESET FORM](#)

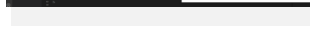
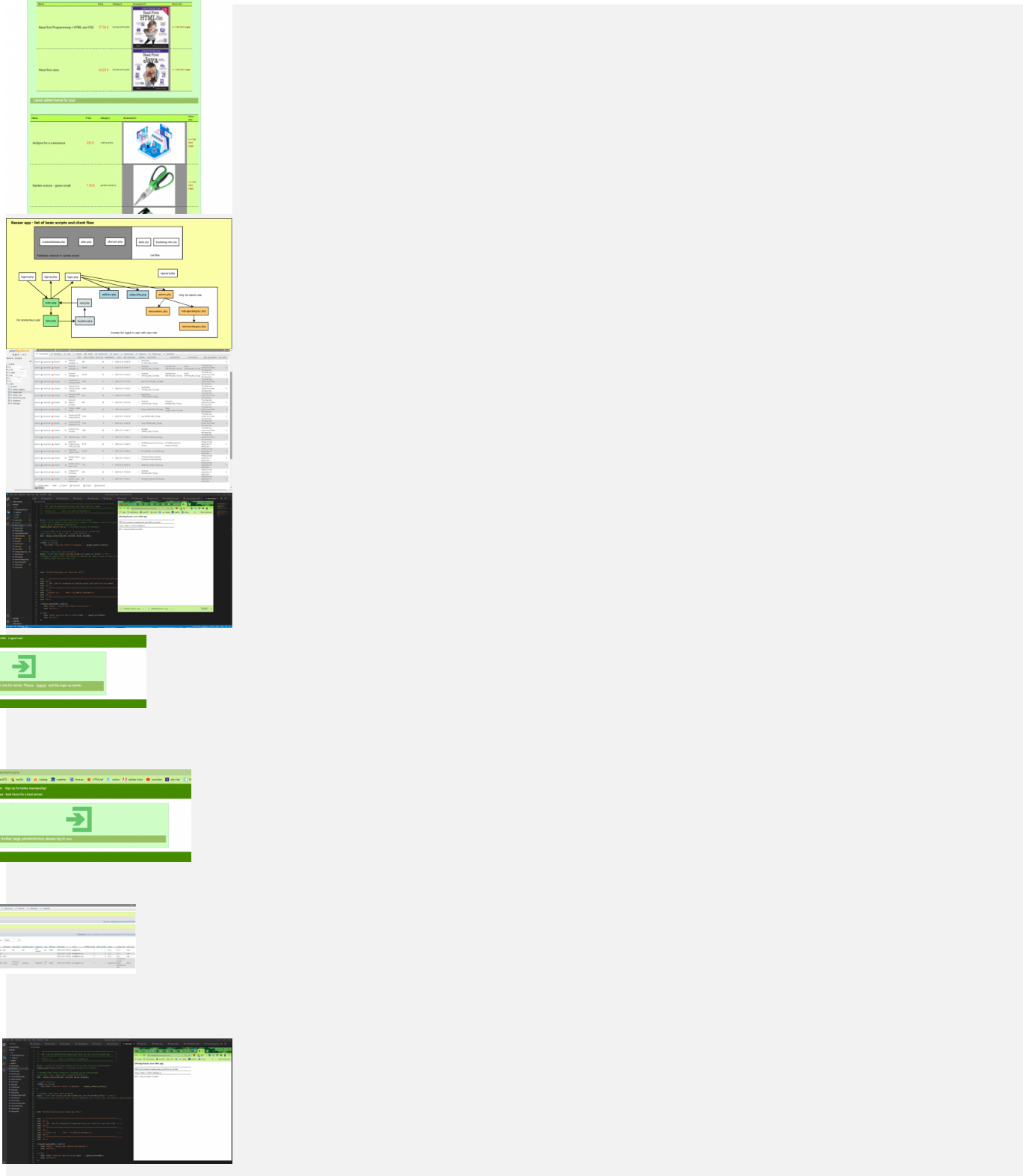
List of items in selected category: books / computer

| Name | Price | Category | Screenshot1 | More Info |
|---------------------------------------|---------|----------------|-------------|---------------------------------------|
| Head first Programming + HTML and CSS | 57.50 € | books/computer | | => Visit item page |
| Head first Java | 43.20 € | books/computer | | => Visit item page |

Latest added items for you!

| Name | Price | Category | Screenshot1 | More Info |
|-------------------------------|--------|-----------------|-------------|---------------------------------------|
| Analyse for e-commerce | 350 € | web/e_shop | | => Visit item page |
| Garden scissors - green small | 7.50 € | garden/scissors | | => Visit item page |

Frontend of the index.php bazaar page





It's time to update about your profile again.

Please select a new profile picture (max. 5MB)

Profile pic: you provide your description for your profile (200)

Bazaar - editing personal profile

Profile updated successfully PASSWORD changed successfully



Your registered with these credentials. They cannot be changed, only way how to obtain new is deleting account asking admin and create new one.

User ID:

User name:

E-mail:

Further user data data:

Nickname:

First name:

Last name:

Address in form - Street No.:

City:

ZIP code in form XXXXX:

localhost:8080/bazaar.php
Unauthorized please Log In



For further profile editing please log in here

Not an on OnDesigner.eu

Home New Order to Chat Group **My Profile** **My Account**

My Profile **My Account**



It's time to update about your profile again.

Please select a new profile picture (max. 5MB)

Profile pic: you provide your description for your profile (200)

Update your data here for your

| ID | First Name | Last Name | Address | City | ZIP |
|----|------------|-----------|------------|---------|-------|
| 10 | admin | admin | Street 678 | City 17 | 00002 |

Bazaar - Login page

Account username is password Login failed



Log In

Username:

Please enter your

Password:

Not an on OnDesigner.eu

Bazaar - Login page



Log In

Username:

Password:

Not an on OnDesigner.eu

Bazaar - Login page

You are logged in as user

If you will login as user with another credentials, please log in again

Bazaar - Sign up for membership buying our items

You are logged in as user

If you will login as user with another credentials, please log in again



Please register for Bazaar membership

Username:

e-mail:

Password:

Password:

Not an on OnDesigner.eu



* Select category-subcategory of product for sale:

If no proper category-subcategory exist, please contact admin of the pages for creation them for you.

SHOW ME INTERESTING THINGS

RESET FORM

List of items in selected category: books / computer

| Name | Price | Category | Screenshot1 | More Info |
|---------------------------------------|---------|----------------|---|---------------------------------------|
| Head first Programming + HTML and CSS | 57.50 € | books/computer |  | => Visit item page |
| Head first Java | 43.20 € | books/computer |  | => Visit item page |

Latest added items for you!

| Name | Price | Category | Screenshot1 | More Info |
|-------------------------------|--------|-----------------|--|---------------------------------------|
| Analyse for e-commerce | 350 € | web/e_shop |  | => Visit item page |
| Garden scissors - green small | 7.50 € | garden/scissors |  | => Visit item page |

Conclusion and source code

In our first article focused on bazaar app page we make basic foundation about main page script and their interactions. During proces of description arrisen some new questions about ways how to extend functionality to fullfil requirements for working portal enabling selling and buying used goods from common users.

Ful code of the projects in as is state can be obtained from github here.

Benchmarkchart – php example code – part 5 – benchmark chart page and further thoughts

Article is focused on way how to alter existing code for showing submitted post as a table. But scores are ordered in descending way and first 3 highest scores are differently styled. Latest part of article list number of ways hot to improve our application.

Numbering posted results in a chart

Before our update to code, we displayed in first table row ID number. For proper ordering from top score to bottom we must order query in descent way by score ID. Then we create auxiliary variable \$position.

```
$position = 1;    // first initialization of position
    while($row = mysqli_fetch_array($output)){
... output omitted
//echo „<td>“ . $row[,'id'] . „</td>“; this line is changed to
incrementing number showing position of the score in chart
```

```

        echo „<td id=\\“$display\\“>“ . $position++
. „</td>“; // increases number of position after displ by one

```

\$position variable is increased by 1 after all pass through while loop.

Different colors for first positions

A bit harder update must be done for marking of first three score position for further styling into a different colors.

| position | score | nickname | date of post | screenshot |
|----------|-------|---------------|---------------------|--|
| 1 | 12636 | maximus636 | 2020-10-03 18:06:31 |  |
| 2 | 11995 | zonerrs555 | 2020-10-03 18:02:47 |  |
| 3 | 11388 | novaProspectX | 2020-10-03 18:01:57 |  |
| 4 | 10750 | fatality536 | 2020-10-03 18:01:15 |  |
| 5 | 6068 | X2U | 2020-10-03 18:00:14 |  |

Our goal for display first scores in a chart

In our code we introduced new variable `$display`. Into `$display` variable is assigned name for `<td>` id first, secondandthree and others. In switch statement along value of `$position` variable these values are added into a `$display` variable.

Next part show how this update is implemented in final php code:

```
$position = 1; // first initialization of position
while($row = mysqli_fetch_array($output)){ //next
rows outputed in while loop
    //echo " <div class=\"mailinglist\"> " ; class
    identification must differ first, two second and other positi
on
        switch ($position) { // along position from fi
rst to bottom displayed rows are marked with different names a
nd recolored in style.css
            case 1: $display='first' ; break;
            case 2:
            case 3: $display='secondandthree' ; break;
            default: $display='others' ;
        };

        echo „<tr>“;
        //echo „<td>“ . $row[,id'] . „</td>“; this
line is changed to incrementing number showing position of th
e score in chart
            echo „<td id=\"$display\">“ . $position++
. „</td>“; // increases number of position after displly by one
            echo „<td id=\"$display\">“ . $row[,score'
] . „</td>“;
            echo „<td id=\"$display\">“ . $row[,nickna
me'] . „</td>“;
            echo „<td id=\"$display\">“ . $row[,write_
```

```

date'] . „</td>“;
                                $image_location = IMAGE_PATH.$row[,screens
hot'];
                                echo „<td id=“gray_under_picture“> <
img src=“$image_location“ alt=“ score image “ height=“95
“> </td>“;
                                echo „</tr>“;
                                echo “ </div> “ ;
                                }
                                echo „</table>“;

```

CSS style.css file was altered this way:

```

/* ***** */
/* Benchamrkchart first position list display      */
/* ***** */
#first {
    background-color: rgb(255, 129, 129);
    text-align: center;
    font-size:18px;
}
#secondandthree {
    background-color: rgb(178, 255, 159);
    text-align: center;
    font-size:18px;
}
#others {
    background-color: rgb(218, 255, 162);
    text-align: center;
}
#gray_under_picture {
    background-color: rgb(88, 88, 88);
    text-align: center;
}
#chart_table_header {

    text-align: center;
}

```

Full code of application can be obtained from github [here](#).



Chart of benchmark results

>>> ADD YOUR SCORE HERE <<<

| position | score | nickname | date of post | screenshot |
|----------|-------|---------------|---------------------|------------|
| 1 | 12636 | maximus636 | 2020-10-03 18:06:31 | |
| 2 | 11995 | zonerrs555 | 2020-10-03 18:02:47 | |
| 3 | 11388 | novaProspectX | 2020-10-03 18:01:57 | |
| 4 | 10750 | fatality536 | 2020-10-03 18:01:15 | |
| 5 | 6068 | X2U | 2020-10-03 18:00:14 | |
| 6 | 5460 | masters | 2020-10-03 17:59:29 | |
| 7 | 3519 | bencherXL | 2020-10-03 17:57:42 | |

Further improvements

As further improvement for benchmarkchart app we encourage you:

- add submit page for admin of the page requests (for score removal)
 - add login mechanism
 - further visual adjustment of frontend
-

Benchmarkchart – php example code – part 4 – limiting access to admin page

Article describe way how to limit access into a admin page using header authentication. Before sending page data, user must by verified by providing username and password. Separate script authorize.php and adding third part into a appvars.php containing username and sha1 has for veriefied password added.

Using headers for authentication

Http headers authentications provide simple way for limiting access for some resources on web. For further reading please wisit

https://en.wikipedia.org/wiki/Basic_access_authentication.

For access to restricted resources user must provide correct username and password. If user enters these correctly, the browser goes ahead and sends page. This dialog between browser and server takes place with headers, which are text messages with specific instructions on what is being requested or delivered.

Further reading about header messages can be obtained from here, 4.10.2020.

Our authentication script looks like this

```
<!--
*****
**** ->
<!-- PHP simple header authorization code
                                ->

<!--
*****
**** ->
<!-- Vrsion: 1.0           Date: 3-4.10.2020 by CDesigner.eu
                                ->

<!--
*****
**** ->
<?php // leading part of page for simple header securing and
basic variable setup
    require_once('appvars.php'); // including variables for da
tabase
    $username = USERNAME;
    $password_sha = PASSWORD_SHA1;
    if (!isset($_SERVER['PHP_AUTH_USER']) ||
        !isset($_SERVER['PHP_AUTH_PW']) ||
        ($_SERVER['PHP_AUTH_USER'] != $username )
    || ( sha1($_SERVER['PHP_AUTH_PW']) != $password_sha) ) {
```

```

        header(,HTTP/1.1 401 Unauthorized');
        header(,WWW-Authenticate: Basic realm="benchmark_admin",);
        exit(<h2>Becnchmarkchart</h2>
Access denied, you must enter a valid username and password to
access this page!
<p> <a href = „index.php„> &lt;&lt; Back to benchmarkresults
homepage page. </a></p>,);
    }

?>

```

For setting original username and password for reference and validation. Our appvar.php definition script was extended of third part as is shown next.

```

<!--
*****
** -><!--      Part III      |      authorization
constants      -> <!--
*****
*** ->
<?php

define(,USERNAME,, ,administrator,);

define(,PASSWORD_SHA1,, ,02cc4d03794b3624b076e48a6d6d18b1f2af8
dc1,); // SHA value for wery weak demonstration password Passw
orD never use
in production environment!!!
    // sha1 has code was generated for example by online app
    http://www.sha1-online.com

?>

```

Adding authorize.php script on secured pages

Our authorize.php script must be added to beginning of all restricted pages. Script must be executed as first code before any HTML content transfer because enables or disables ability to access appropriate web resources.

Next segment of code contains example of including code with require_once(); php function.

```
<!--
*****
**** ->
<!-- PHP „self“ code handling administration of removal
        ->

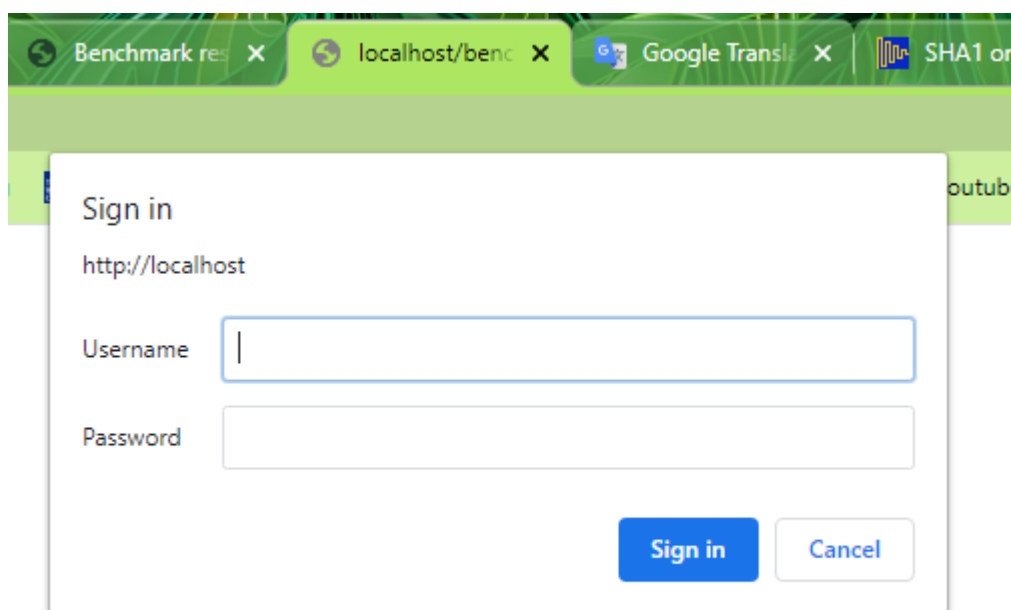
<!--
*****
**** ->
<!-- Vrsion: 1.0           Date: 27-XX.X.2020 by CDesigner.eu
        ->

<!--
*****
**** ->
<?php // leading part of page for simple header securing and
basic variable setup
    require_once('appvars.php'); // including variables for da
tabase
    require_once('authorize.php'); // authorization script for
simple header authorization
    // two variables for message and styling of the mesage wit
h bootstrap
    $msg = “;
    $msgClass = “;
    // default values of auxiliary variables
```

?>

For proper work of admin script is crucial to enable access to code on both admin.php and remove.php in same time. This requirement is fulfilled simply adding same code with same
header(,WWW-Authenticate: **Basic realm="benchmark_admin,,,"**); .

After altering of our pages described way, our browser will prompt for entering username and password as is shown on next picture.



Authentication request for enabling access to limited resources with HTML headers

Full code for further study

Most current version of application code can be obtained from github [here](#).