WaterMarky 2.0

Login-Mechanismus ist sicher implementiert, Passwörter sind gehashed und gesalzen gespeichert.

sign_in.php

Clientseitige Validierung:

username

```
<input type="text" name="username" class="form-control"
       id="username"
       value=""
       placeholder="Upper- and lower-case, min 6 characters."
       maxlength="30" required="true"
       pattern="(?=.*[a-z])(?=.*[A-Z])[a-zA-Z]{6,}"
       title="Upper- and lower-case, min 6 characters.">
```

password

```
<input type="password" name="password" class="form-control"
       id="password"
       placeholder="Upper- and lower-case letters, numbers,
specialcharacters, min. 8 characters"
pattern="(?=^.{8,}$)((?=.*\d+)(?=.*\W+))(?![.\n])(?=.*[A-Z])(?=.*[
a-z]).*$"
       title="minimum one Upper-, one lower-case letter, one
number and one specialcharacter, minimum 8 characters long."
       required="true">
```

POST Request

```
if ($_SERVER["REQUEST_METHOD"] == "POST" && empty($error))
$username = trim($_POST['username']);
$password = trim($_POST['password']);
```

Serverseitige Validierung

username

```
if(!preg_match("/(?=.*[a-z])(?=.*[A-Z])[a-zA-Z]{6,}/", $username) ||
strlen($username) > 30)
    $error .= "The username does not meet the required format.<br />";
```

password

```
if(!preg_match("/(?=^.{8,}$)((?=.*\d)|(?=.*\W+))(?![.\n])(?=.*[A-Z])(?=
.*[a-z]).*$/", $password))
    $error .= "The password does not meet the required format.<br />";
```

```
$username = htmlspecialchars(trim($_POST['username']));
$password = htmlspecialchars(trim($_POST['password']));
```

Passwort wird gesalted

```
$saltedPw = "iLiKeMy".$password."ButIlIkeCaKeMuChMoRe";
```

Prepared statement

```
$query = "SELECT * FROM users WHERE username = ?";
$stmt = $mysqli->prepare($query);
$stmt->bind_param('s', $username);
$stmt->execute();
```

```
while($user = $result->fetch_assoc())
    {
```

Passwort überprüfen

```
if(password_verify($saltedPw, $user['password']) && $user['username']
=== $username)
```

alte Session invalidieren und neue Session ID vergeben

```
session_regenerate_id();
```

User einloggen und Rechte zuweisen

```
$_SESSION = array();
$_SESSION['username'] = htmlspecialchars(trim($username));
$_SESSION['loggedin'] = true;

$role = 'User';
if($user['role_id'] == 2)
   $role = 'Magick User';

$_SESSION['role_id'] = $role;
```

User weiterleiten

```
header('Location: WaterMarky.php');
```

Das Problem der Session-Fixation wurde in der Lösung berücksichtigt und abgesichert.

Invalidierung der Session zum geeigneten Zeitpunkt. Allfällige Übernahme von Daten aus der alten Session.
(Wie wurde das realisiert?)

Beim Log in (sign_in.php) wird `session_regenerate_id();` ausgeführt und beim Log out, (sign_out.php) `session_destroy();`.

Session Timeout in php.ini:
session.gc_maxlifetime=1200

on all sites

```
if (isset($_SESSION['LAST_ACTIVITY']) && (time() -
$_SESSION['LAST_ACTIVITY']) > 1200)
{
   if(isset($_SESSION['username']))
      error_log("SESSION TIMEOUT: LAST_ACTIVITY:
".$_SESSION['LAST_ACTIVITY']." User: ".$_SESSION['username']);

    // last request was more than 20 minutes ago
    session_unset();     // unset $_SESSION variable for the run-time
    session_destroy();   // destroy session data in storage
    header('Location: sign_in.php');
```

```
}
else
   $_SESSION['LAST_ACTIVITY'] = time();


if (!isset($_SESSION['CREATED']))
    $_SESSION['CREATED'] = time();
else if(isset($_SESSION['CREATED']) && (time() - $_SESSION['CREATED'])
> 1200)
{
    if(isset($_SESSION['username']))
        error_log("SESSION REGENERATE ID: CREATED:
".$_SESSION['CREATED']." User: ".$_SESSION['username']);

    // session started more than 20 minutes ago
    session_regenerate_id(true);    // change session ID for the
current session and invalidate old session ID
    $_SESSION['CREATED'] = time();  // update creation time
}
```

Injections werden durch entsprechende Gegenmassnahmen verhindert.

Inputvalidierung

Client

sign_in.php

```
<input type="text" name="username" class="form-control"
       id="username"
       value=""
       placeholder="Upper- and lower-case, min 6 characters."
       maxlength="30" required="true"
       pattern="(?=.*[a-z])(?=.*[A-Z])[a-zA-Z]{6,}"
       title="Upper- and lower-case, min 6 characters.">
```

```
<input type="password" name="password" class="form-control"
       id="password"
       placeholder="Upper- and lower-case letters, numbers,
specialcharacters, min. 8 characters"
```

```
pattern="(?=^.{8,}$)((?=.*\d+)(?=.*\W+))(?![.\n])(?=.*[A-Z])(?=.*[
a-z]).*$"
       title="minimum one Upper-, one lower-case letter, one
number and one specialcharacter, minimum 8 characters long."
       required="true">
```

sign_up.php

```
<input type="text" name="firstname" class="form-control" id="firstname"
                  value="<?php echo $firstname ?>"
                  placeholder="Enter you're firstname."
                  required="true">
```

```
<input type="text" name="lastname" class="form-control" id="lastname"
                  value="<?php echo $lastname ?>"
                  placeholder="Enter you're lastname"
                  maxlength="30"
                  required="true">
```

```
<input type="email" name="email" class="form-control" id="email"
                  value="<?php echo $email ?>"
                  placeholder="Enter you're mailadress."
                  maxlength="100"
                  required="true">
```

```
<input type="text" name="username" class="form-control" id="username"
                  value="<?php echo $username ?>"
                  placeholder="Upper- and lower-case letters, min 6
characters."
                  maxlength="30" required="true"
                  pattern="(?=.*[a-z])(?=.*[A-Z])[a-zA-Z]{6,}"
                  title="Upper- and lower-case letters, min 6
characters.">
```

```
<input type="password" name="password" class="form-control"
                  id="password"
                  placeholder="Upper- and lower-case letters, numbers,
specialcharacters, min. 8 characters"
pattern="(?=^.{8,}$)((?=.*\d+)(?=.*\W+))(?![.\n])(?=.*[A-Z])(?=.*[a-z])
.*$"
                  title="minimum one Upper-, one lower-case letter, one
number and one specialcharacter, minimum 8 characters long."
```

```
                    required="true">
```

Server

sign_in.php

```
if(!preg_match("/(?=.*[a-z])(?=.*[A-Z])[a-zA-Z]{6,}/", $username) ||
strlen($username) > 30)
    $error .= "The username does not meet the required format.<br />";
```

```
if(!preg_match("/(?=^.{8,}$)((?=.*\d)|(?=.*\W+))(?![.\n])(?=.*[A-Z])(?=
.*[a-z]).*$/", $password))
    $error .= "The password does not meet the required format.<br />";
```

```
$username = htmlspecialchars(trim($_POST['username']));
$password = htmlspecialchars(trim($_POST['password']));
```

sign_up.php

```
$firstname = htmlspecialchars(trim($_POST['firstname']));
```

```
$lastname = htmlspecialchars(trim($_POST['lastname']));
```

```
$email = htmlspecialchars(trim($_POST['email']));
 if (filter_var($email, FILTER_VALIDATE_EMAIL) === false)
```

```
if(!preg_match("/(?=.*[a-z])(?=.*[A-Z])[a-zA-Z]{6,}/", $username))
```

```
if(!preg_match("/(?=^.{8,}$)((?=.*\d+)(?=.*\W+))(?![.\n])(?=.*[A-Z])(?=
.*[a-z]).*$/", $password))
```

```
$username = htmlspecialchars(trim($_POST['username']));
$password = htmlspecialchars(trim($_POST['password']));
```

Prepared Statement

sign_in.php

```
$query = "SELECT * FROM users WHERE username = ?";
$stmt = $mysqli->prepare($query);
$stmt->bind_param('s', $username);
$stmt->execute();
```

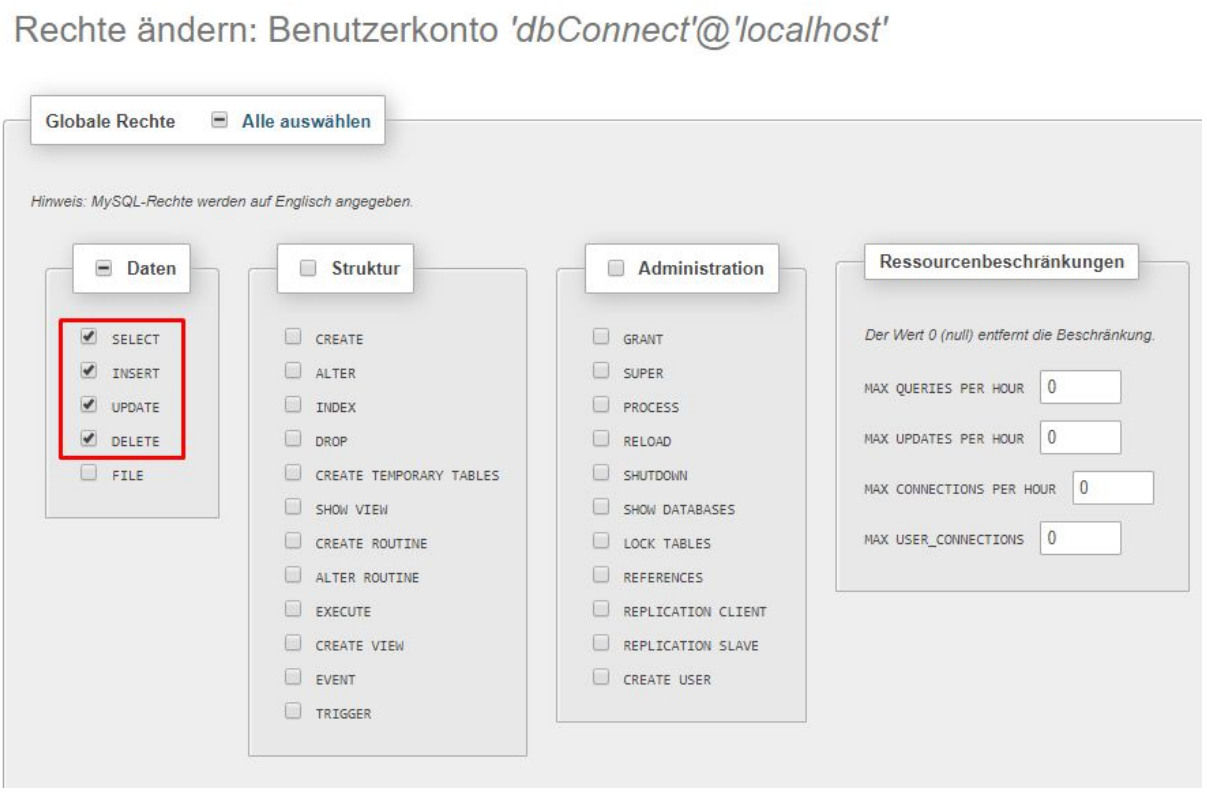
sign_up.php

```
//check if username already exists
$query = "SELECT * FROM users WHERE username = ?";

$stmt = $mysqli->prepare($query);
$stmt->bind_param('s', $username);
$stmt->execute();

$result = $stmt->get_result();

while($user = $result->fetch_assoc())
 {
  if($user['username'] === $username)
     $error = 'Username '.$username.' is already taken!';
 }
```

```
$query = "INSERT INTO users (role_id, email, firstname, lastname,
password,username)
VALUES (?,?,?,?,?,?); ";

$stmt = $mysqli->prepare($query);

if($stmt == false)
   $error = 'Something went wrong!';

if(empty($error))
  {
   $stmt->bind_param('isssss', $role_id, $email,
$firstname,$lastname,$password,$username);
   $stmt->execute();

   $result = $stmt->get_result();
   $stmt->close();
```

DB-Benutzer

Der DB-Benutzer “dbConnect” welcher verwendet wird hat nur die Rechte auf: SELECT, INSERT, UPDATE und DELETE.



XSS-Angriffe werden durch entsprechende Gegenmassnahmen verhindert

CSP

sign_in.php/sign_up.php/sign_out.php/user_info.php/Watermarky.php/PreWaterMarky.php

```
<meta http-equiv="Content-Security-Policy" content="default-src 'self';
script-src 'self'; script-src-elem *; img-src *; style-src
'unsafe-inline'; style-src-elem *">
<meta http-equiv="X-Content-Security-Policy" content="default-src
'self'; script-src 'self'; script-src-elem *; img-src *; style-src
'unsafe-inline'; style-src-elem *">
<meta http-equiv="X-WebKit-CSP" content="default-src 'self'; script-src
'self'; script-src-elem *;  img-src *; style-src 'unsafe-inline';
style-src-elem *">
```

Output Escaping

user_info.php

```
<input type="text" class="form-control" id="firstname" disabled=true
name="change_firstname"
value="<?php echo htmlspecialchars(trim($sfirstname))?>"
aria-label="Kontaktinformation"
aria-describedby="button-addon2" maxlength="30" required="true">
```

```
<input type="text" class="form-control"
id="lastname" disabled=true name="change_lastname"
value="<?php echo htmlspecialchars(trim($slastname));?>"
aria-label="Kontaktinformation" aria-describedby="button-addon2"
maxlength="30"
required="true">
```

```
<input type="text" class="form-control" id="email" disabled=true
name="change_mail"
value="<?php echo htmlspecialchars(trim($semail))?>"
aria-label="Kontaktinformation" maxlength="100"
required="true" aria-describedby="button-addon2">
```

user_info.php/WaterMarky.php

Anzeige des Usernamen

```
<a class="nav-link align-middle text-success" href="WaterMarky.php"
>'.htmlspecialchars(trim($_SESSION['username'])).' <span
class="sr-only">(current)</span></a>
```

```
<a class="nav-link align-middle text-success" href="user_info.php"
>'.htmlspecialchars(trim($_SESSION['username'])).' <span
class="sr-only">(current)</span></a>
```

Inputvalidierung

Client

sign_in.php

```
<input type="text" name="username" class="form-control"
       id="username"
       value=""
       placeholder="Upper- and lower-case, min 6 characters."
       maxlength="30" required="true"
       pattern="(?=.*[a-z])(?=.*[A-Z])[a-zA-Z]{6,}"
       title="Upper- and lower-case, min 6 characters.">
```

```
<input type="password" name="password" class="form-control"
       id="password"
       placeholder="Upper- and lower-case letters, numbers,
specialcharacters, min. 8 characters"
pattern="(?=^.{8,}$)((?=.*\d+)(?=.*\W+))(?![.\n])(?=.*[A-Z])(?=.*[
a-z]).*$"
       title="minimum one Upper-, one lower-case letter, one
number and one specialcharacter, minimum 8 characters long."
       required="true">
```

sign_up.php

```
<input type="text" name="firstname" class="form-control" id="firstname"
                  value="<?php echo $firstname ?>"
                  placeholder="Enter you're firstname."
                  required="true">
```

```
<input type="text" name="lastname" class="form-control" id="lastname"
                  value="<?php echo $lastname ?>"
                  placeholder="Enter you're lastname"
                  maxlength="30"
                  required="true">
```

```
<input type="email" name="email" class="form-control" id="email"
                  value="<?php echo $email ?>"
                  placeholder="Enter you're mailadress."
                  maxlength="100"
                  required="true">
```

```
<input type="text" name="username" class="form-control" id="username"
                  value="<?php echo $username ?>"
                  placeholder="Upper- and lower-case letters, min 6
characters."
                  maxlength="30" required="true"
                  pattern="(?=.*[a-z])(?=.*[A-Z])[a-zA-Z]{6,}"
                  title="Upper- and lower-case letters, min 6
characters.">
```

```
<input type="password" name="password" class="form-control"
                  id="password"
                  placeholder="Upper- and lower-case letters, numbers,
specialcharacters, min. 8 characters"
pattern="(?=^.{8,}$)((?=.*\d+)(?=.*\W+))(?![.\n])(?=.*[A-Z])(?=.*[a-z])
.*$"
                  title="minimum one Upper-, one lower-case letter, one
number and one specialcharacter, minimum 8 characters long."
                  required="true">
```

Server

sign_in.php

```
if(!preg_match("/(?=.*[a-z])(?=.*[A-Z])[a-zA-Z]{6,}/", $username) ||
strlen($username) > 30)
    $error .= "The username does not meet the required format.<br />";
```

```
if(!preg_match("/(?=^.{8,}$)((?=.*\d)|(?=.*\W+))(?![.\n])(?=.*[A-Z])(?=
.*[a-z]).*$/", $password))
    $error .= "The password does not meet the required format.<br />";
```

```
$username = htmlspecialchars(trim($_POST['username']));
$password = htmlspecialchars(trim($_POST['password']));
```

sign_up.php

```
$firstname = htmlspecialchars(trim($_POST['firstname']));
```

```
$lastname = htmlspecialchars(trim($_POST['lastname']));
```

```
$email = htmlspecialchars(trim($_POST['email']));
 if (filter_var($email, FILTER_VALIDATE_EMAIL) === false)
```

```
if(!preg_match("/(?=.*[a-z])(?=.*[A-Z])[a-zA-Z]{6,}/", $username))
```

```
if(!preg_match("/(?=^.{8,}$)((?=.*\d+)(?=.*\W+))(?![.\n])(?=.*[A-Z])(?=
.*[a-z]).*$/", $password))
```

```
$username = htmlspecialchars(trim($_POST['username']));
$password = htmlspecialchars(trim($_POST['password']));
```

Session Timeout in php.ini:
session.gc_maxlifetime=1200

on all sites

```
if (isset($_SESSION['LAST_ACTIVITY']) && (time() -
$_SESSION['LAST_ACTIVITY']) > 1200)
{
   if(isset($_SESSION['username']))
      error_log("SESSION TIMEOUT: LAST_ACTIVITY:
".$_SESSION['LAST_ACTIVITY']." User: ".$_SESSION['username']);

    // last request was more than 20 minutes ago
    session_unset();     // unset $_SESSION variable for the run-time
    session_destroy();   // destroy session data in storage
    header('Location: sign_in.php');
}
else
   $_SESSION['LAST_ACTIVITY'] = time();


if (!isset($_SESSION['CREATED']))
    $_SESSION['CREATED'] = time();
else if(isset($_SESSION['CREATED']) && (time() - $_SESSION['CREATED'])
> 1200)
{
```

```
    if(isset($_SESSION['username']))
        error_log("SESSION REGENERATE ID: CREATED:
".$_SESSION['CREATED']." User: ".$_SESSION['username']);

    // session started more than 20 minutes ago
    session_regenerate_id(true);    // change session ID for the
current session and invalidate old session ID
    $_SESSION['CREATED'] = time();  // update creation time
}
```

Sicherer Upload

upload.php

```
//get uploaded file
        $file = $_FILES['file'];

        //get file extension
        $tmp = explode('.', $file['name']);
        $fileExt = strtolower(end($tmp));

        //check if file extension is allowed
        $allowed = array('jpg', 'png', 'bmp', 'svg');
        if(!in_array($fileExt, $allowed))
        {
            error_log("UPLOAD FAILED: ERROR: filetype is not allowed
User:".$_SESSION['username']);
            return popMsg("Files must be of one of the following types
.jpg/.png/.bmp/.svg You cannot upload files of type ".$fileExt);
        }

        //check if an error occured
        if($file['error'] !== 0)
        {
            error_log("UPLOAD FAILED: ERROR: ".$file['error']."
User:".$_SESSION['username']);
            return popMsg("There was an error uploading your file
error");
        }

        //check if file is bigger than 1GB
```

```
        if($file['size'] > 1000000000)
        {
            error_log("UPLOAD FAILED: ERROR: file is to large
User:".$_SESSION['username']);
            return popMsg("The file is too big! Cannot upload files of
size larger than 1GB");
        }

        //create uniqe filename
        $uniqFileName = explode('.', $file['name'])[0].".".uniqid('',
true).".".$fileExt;

        //upload
        if(move_uploaded_file($file['tmp_name'],
'upload/'.$uniqFileName))
        {
            $_SESSION['newFile'] = true;
            error_log("UPLOAD SUCCESS: FILE: ".$uniqFileName." User:
".$_SESSION['username']);
            return popMsg("Successfully uploaded");
        }

        error_log("UPLOAD FAILED: ERROR: There was an error saving the
file FILE: ".$uniqFileName." User:".$_SESSION['username']);
        return popMsg("There was an error uploading your file!");
```

Quellen Upload: https://www.youtube.com/watch?v=JaRq73y5MJk
https://stackoverflow.com/questions/38509334/full-secure-image-upload-script