#### https://www.halvorsen.blog



## Create User page and Login page in PHP and MySQL

Hans-Petter Halvorsen

#### Contents

- Introduction
  - Overview of existing PHP and MySQL CRUD Web Application
- Security and Passwords
- <u>Create Database and User Page</u>
  - <u>Create Database</u>
  - Create User Page
- <u>Create Login Page</u>
- Check if User is Logged in
- Improvements and Additional Resources
  - 2-Factor Authentication
  - Authorization
- <u>Summary</u>

#### https://www.halvorsen.blog

#### Create User page and Login page in PHP

# Introduction

#### Hans-Petter Halvorsen



## Introduction

- This tutorial will demonstrate how to create basic login functionality in PHP.
- We show the principles on a Books/Library Application created earlier.
- The focus is to learn how to implement basic login functionality, so user experience, layout, robustness, code structure, code quality, etc. can be significantly improved compared to the simplified examples given here.

### **Books Web Application**

New Book

Please enter book information:

This is the existing PHP Application made. The focus was to create CRUD functionality, i.e., Create, Read, Update and Delete data in a MySQL database. The next step would be to add some authentication functionality.

Title:
Author: Topic:
Save
Update Book Please enter book information:
Title: Web Apps
Author: Elvis Presly
Topic: Programming

## **Books PHP Web Application**

The Books PHP CRUD Web Application consists of the following files:

- config.php
- books.php
- book\_create.php
- book\_create\_db.php
- book\_update
- book\_update\_db.php
- book\_delete.php

**CRUD** – **C**reate, **R**ead, **U**pdate and **D**elete Data in the MySQL Database



#### **New Library Application**



## **Create User and Login**

We will update the Books WebApp with Login functionality. Here is a simple step by step procedure to do that:

- 1. Create a new Database Table with Username and Password information.
- 2. Crete a "NewUser" page that inserts UserName and Password information into the Database.
  - Here you can use the built-in "password\_hash" PHP function.
- **3.** Create a "Login" page where the user needs to enter UserName and Password.
  - Check the entered UserName and Password with the information stored in the Database. Here you can use the built-in "password\_verify" PHP function.
  - If Password is correct, create a Session variable that says the user is logged in.
- 4. In all other PHP files, perform a check in the beginning whether the user is logged in or not (check if the session variable is true)

#### Tools

- **PHP** a server scripting language for making dynamic web pages, typically communicating with a Database.
- **MySQL** a widely used relational database management system (RDBMS). MySQL is free and open-source.
- We will host our PHP files on an existing **Web Server** that supports PHP and MySQL. You can also create your own or use an existing hosting provider.
- We will use Visual Studio Code (you can use another IDE if you prefer).
- We will transfer the local files to the Web Server using **FTP** (File Transfer Protocol). We will use **WinSCP** (you can use another FTP tool if you prefer).
- **phpMyAdmin** a free and open-source administration tool for MySQL (and MariaDB).

#### https://www.halvorsen.blog

Create User page and Login page in PHP

# Security and Passwords



**Table of Contents** 

## **Security and Passwords**

Here are some security and password related topics that we should know about.

- Authentication vs Authorization.
- Encryption and Decrypting.
- Hashing.
- Salting.
- 2 Factor Authentication.
- Etc.

## Authentication vs Authorization

- Authentication is a process to authenticate a user, that is, to verify that someone is who they say they are.
- Authorization is about determining a user's level of access and then granting access based on that level.

We will focus on Authentication in this tutorial.

#### Authentication

Simplified Example of Authentication:



Simplified Example of Authorization :

#### User



Username:

Johnny

Password:

Password123

"Read" Access Level: If User = "User X", he gets access to read data only.

"Super User" Access Level: If User = "User Y", he gets access to creating new documents.

"Administrator" Access Level: If User = "User Z", he is an "Administrator" and gets access to everything.

Database

Here you see a basic example how to implement Authorization in the Database:



A specific User can be part of only one Access Level in this example Access Levels Examples:

- Student
- Teacher
- Administrator

Here you specify the Features the different Access Levels have access to Features Examples:

Database

- Search
- Show Grades
- Delete User

# **Encryption and Decrypting**

- Encryption is to transform information to make it unreadable by persons that should not have access to the information.
- You need a Key in to transform (decrypt) the information back to plain text.
- To encrypt data, you use an algorithm. Many different encryption algorithms do exist.

Plain Text



**Encrypted Text** 

Plain Text

#### When should encryption be used?

- Encryption is a two-way function.
- You encrypt information with the intention of decrypting it later.
- Examples when to use encryption:
  - Protecting Files and Information on your Computer
  - Protecting your Cloud data
  - Transmitting Data between 2 Computers
  - Etc.
- Encryption is reversible if you have the key.
- It can be used for password protection, but a better approach is to use something called "Hashing".

## Hashing

- Password hashing is a way to store passwords by transforming them into a fixed-length string of characters, which is not reversible.
- Hashing is a one-way function, while Encryption is a two-way function.
- Encryption is meant to protect data in transit, while Hashing is meant to verify that a file or piece of data hasn't been altered—that it is authentic. In other words, it serves as a check-sum.
- Every hash value is unique.

## Hashing – 2 Steps



#### Possible to hack the Hashed Password?

#### Password Table for System X

UserName	HashedPassword		
Mike	4420d1918bbcf7	-	-
Bob	73fb51a0c9be7d		/
Peter	4420d1918bbcf7		

Password	HashedPassword
tesla	4420d1918bbcf7 -
friendship	73fb51a0c9be7d
bicycle	7420e1618abcf6

Rainbow table

If a Hacker gets access to this Database, he can see that Mike and Peter have the same password.

But he does not know the actual password.

If the Hacker also has access to a socalled "**Rainbow table**" (which is essentially a pre-computed database of hashes), he may also be able to find the Password (as seen here). If you have a complicated password and that you are not reusing your password several places, it is less likely that your password is in such a Rainbow table

## Salting

- Salting is a technique typically used for Password Hashing.
- It is a unique value that can be added to the end of the password to create a different hash value.
- This means two identical passwords don't end up with the same hashed value.
- The additional value is referred to as a "salt".
- This is done to make it even more secure.
- Typically, the Hashing Algorithm uses a Random salt.
  - This prevents an attacker from seeing whether users have the same password.

## Hashing with Salt

```
password = "Password123"
salt = "Tesla"
```

passwordHashed = HashPassword(password, salt);

Typically, Salting is built into the Hashing Algorithm, and it is changed every time

```
password = "Password123"
ph1 = HashPassword(password);
ph2 = HashPassword(password);
```



This means if 2 different Users use the same Password, the Hashed Password will be different!

## Hashing with Salt

Assume Mike and Peter use the same Password



If a Hacker gets access to this Database, he cannot see that Mike and Peter have the same password. Because a random Salt has made these 2 Hashed Passwords different!

#### **Create User and Login**



#### **Password Functions**

\$password = 'Password123';

\$hashed\_password = password\_hash(\$password, PASSWORD\_DEFAULT);

https://www.php.net/manual/en/function.password-hash.php

password\_hash() has built-in salting

2

You can also choose between different types of hashing algorithms.

\$verified = password verify(\$password, \$hashed password);

#### https://www.php.net/manual/en/function.password-verify.php

Here, <code>\$password is entered by the user in the Login page and <code>\$hashed\_password</code> is the hashed password database stored in the database</code>

## **2-Factor Authentication**

- To make your authentication even better you can also consider implementing so-called 2-Factor Authentication.
- This can be done in different ways.
- The easiest way is to use email.
- You can, e.g., use a built-in email() function.
- The procedure is then to generate and send a verification code on email from your web application during the login process. Then the the user needs to enter the verification code received on email during the login process.

```
$verificationcode = strtoupper(uniqid());
```

mail(\$to, \$subject, \$message, \$headers);

#### https://www.halvorsen.blog

Create User page and Login page in PHP

# Create Database and User Page

Hans-Petter Halvorsen

**Table of Contents** 

### **Books Web Application**

New Book

Please enter book information:

This is the existing PHP Application made. The focus was to create CRUD functionality, i.e., Create, Read, Update and Delete data in a MySQL database. The next step would be to add some authentication functionality.

Title:
Author: Topic:
Save
Update Book Please enter book information:
Title: Web Apps
Author: Elvis Presly
Topic: Programming

#### **New Library Application**



#### https://www.halvorsen.blog

#### Create User page and Login page in PHP

# **Create Database**

#### Hans-Petter Halvorsen



## phpMyAdmin



phpMyAdmin is used to administrate your MySQL Database. Here you can create tables, run SQL queries, etc. phpMyAdmin is basically just a web application written in PHP. We will use phpMyAdmin to create a Database Tables and insert some data into that tables.

#### Library and Books

BookId int PRIMARY KEY AUTO_INCREMENT, Title varchar(100) NOT NULL, Author varchar(100) NOT NULL, Topic varchar(100) NOT NULL );	CREATE TABLE BOOK (						
Title varchar(100) NOT NULL, Author varchar(100) NOT NULL, Topic varchar(100) NOT NULL );	BookId int PRIMARY KEY AUTO	D_INCREMENT,					
Author varchar(100) NOT NULL, Topic varchar(100) NOT NULL );	Title varchar(100) NOT NULL,						
Topic varchar(100) NOT NULL );	Author varchar(100) NOT NUI	L,					
);	Topic varchar(100) NOT NULL						
	);						

```
insert into BOOK (Title, Author, Topic)
values ('Web Apps', 'Elvis Presly', 'Programming');
insert into BOOK (Title, Author, Topic)
values ('IoT and Cloud', 'John Wayne', 'IoT');
insert into BOOK (Title, Author, Topic)
values ('C#', 'Rune Hansen', 'Programming');
```

#### **User Information**

**CREATE TABLE PERSON (** 

);

PersonId int PRIMARY KEY AUTO\_INCREMENT, FullName varchar(100) NOT NULL, EMail varchar(100) NOT NULL UNIQUE, Pwd varchar(100) NULL

insert into PERSON (FullName, EMail, Pwd)
values ('xxxxx', 'xxxxx', 'xxxxx');

#### https://www.halvorsen.blog

#### Create User page and Login page in PHP

# Create User Page

#### Hans-Petter Halvorsen



#### **User Administration**

			S Library App X S Create Person X + X
			← → C to web01.usn.no/~hansh /person_create.php @ ☆ ☆ 😩 Error :
<ul> <li>S Library App</li> </ul>	× S User Administration ×	+ – –	
← → ♂ 🖙 web01.usn.no/	~hansha/persons.php	* 😩	Please enter the personal information:
			Full Name:
Persons Here you find a list of available pe	ersons registered in the system:		EMail: Password:
# Name	EMail	Action	C Library App     X O Update Personal Information X +
1 Hans-Petter Halvorsen	hans.p.halvorsen@usn.no	Update Delete	Save ← → C <sup>c</sup> web01.usn.no/~hansh //person_update.php?personid=1 ∞ ☆ <sup>2</sup> E
2 Elvis Presley	elvis.presely@usa.com	Update Delete	Update Personal Information Please update person information:
3 John Wayne	john.wayne@hollywod.com	Update Delete	Full Name: Hans-Petter Halvorsen
New Person			EMail: hans.p.halvorsen@usn.no
			ressword.
			Save

#### **Persons/Users**

~	3	Library A	рр	×	S User Administration	×	+		_		×
÷	$\rightarrow$	G	0-0	web01.usn.no/~har	nsha/persons.php			☆		Er	rror :

#### Persons

Here you find a list of available persons registered in the system:

#	Name	EMail	Action
1	Hans-Petter Halvorsen	hans.p.halvorsen@usn.no	Update Delete
2	Elvis Presley	elvis.presely@usa.com	Update Delete
3	John Wayne	john.wayne@hollywod.com	Update Delete
N	ew Person		
-			

```
K File Edit Selection View ···
                                                                                     88~
                                                                                             🖙 persons.php 🗙
                                                                                                                   ....
                 neDrive > Courses > Webutvikling > Tutorials > PHP > Development > Create User and Login > 🦛 persons.php
Q
       11
            <head>
             <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/css/bootstrap.min.css" rel="stylesheet">
       16
             <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/js/bootstrap.bundle.min.js"></script>
e٩
            </head>
       18
       19
            <script>
a>
       20
       21
            function deletePerson(personid)
88
       22
       23
             let message = "Do you really want to delete person #'" + personid + "'?";
       24
       25
              if (confirm(message) == true)
       26
       27
               window.location.href = "person_delete.php?personid=" + personid;
       28
       29
       30
       31
            </script>
       32
       33
            <body>
            <div class="container-fluid pt-5">
       34
       35
       36
            <h1>Persons</h1>
       37
       38
            Here you find a list of available persons registered in the system:
       39
       40
            <div class="table-responsive">
       41
            42
              <thead>
       43
               >
       44
                 #
       45
                 Name
       46
                 EMail
       47
                 Action
       48
               49
              </thead>
       50
       51
              52
              <?php
              // Get Data from Database
       54
              $sql = "SELECT PersonId, FullName, EMail FROM PERSON";
       55
             $result = mysqli_query($conn, $sql);
       56
       57
              if (mysqli_num_rows($result) > 0) {
       58
               // output data of each row
       59
               while($row = mysqli fetch assoc($result)) {
       60
                 echo "";
       61
                 echo "" . $row["PersonId"] . "";
       62
                 echo "" . $row["FullName"] . "";
                 echo "" . $row["EMail"] . "";
       63
       64
                 echo " <a href='person_update.php?personid=" . $row["PersonId"] . "' class='btn btn-info'>Update</a>";
       65
                 echo " <a href='javascript:deletePerson(" . $row["PersonId"] . ")' class='btn btn-danger'>Delete</a> 
       66
                 echo "";
       67
       68
              } else {
       69
               echo "0 results";
       70
       71
             2>
       72
       73
             74
            75
            </div>
       76
       77
           <a href="person create.php" class="btn btn-success">New Person</a>
53
       78
✓ ⑦ Restricted Mode ⊗ 0 △ 0 № 0
                                                                                       Ln 1, Col 1 Spaces: 2 UTF-8 CRLF PHP Q
```

#### **New Person**

•	٢	Library A	pp	×	S Create Person	×	+			-		×
÷	$\rightarrow$	С	0	web01.usn.no/~har	nsha/person_create.php			œ	☆		Error	:

#### **New Person**

Please enter the personal information:

Full Name:

EMail:

Password:

Sau

~	ne con a	selection view of	With Search Search	
rC	💏 person	_create.php ×		□ …
-ر	C: > Users	> hansp > OneDriv	> Courses > Webutvikling > Tutorials > PHP > Development > Create User and Login > 🌸 person_create.php	
ρ	1	php</th <th></th> <th></th>		
	2	require_or	ce 'config.php';	
Je J	3	require_or	ce 'functions.php';	
	4	session_st	art();	
ė^	5	checkUser(	);	
00	6	?>		
ш	7			
	8	< ! DOCTYPE	html>	
	9	<html></html>		
	10			
	11	<head></head>		
	12	<title>0</title>	reate Person	
	13	<meta ch<="" td=""/> <td>arset="utf-8"&gt;</td> <td></td>	arset="utf-8">	
	14	<meta na<="" td=""/> <td><pre>me="viewport" content="width=device-width, initial-scale=1"&gt;</pre></td> <td></td>	<pre>me="viewport" content="width=device-width, initial-scale=1"&gt;</pre>	
	15	<link hr<="" td=""/> <td>ef="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/css/bootstrap.mi</td> <td><pre>n.css" rel="stylesheet"&gt;</pre></td>	ef="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/css/bootstrap.mi	<pre>n.css" rel="stylesheet"&gt;</pre>
	16	<script< td=""><td><pre>src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/js/bootstrap.bu</pre></td><td>ndle.min.js"&gt;</td></script<>	<pre>src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/js/bootstrap.bu</pre>	ndle.min.js">
	17			
	18			
	19	<body></body>		
	20	<div class<="" td=""><td>="container-fluid pt-5"&gt;</td><td></td></div>	="container-fluid pt-5">	
	21			
	22	<h1>New Pe</h1>	rson	
	23	Please	enter the personal information:	
	24			
	25	<div class<="" td=""><td>=form-group"&gt;</td><td></td></div>	=form-group">	
	26	<form acti<="" td=""><td>on="person_create_db.php" method="POST"&gt;</td><td></td></form>	on="person_create_db.php" method="POST">	
	27	<label< td=""><td>.for="fullname" class="form-label"&gt;Full Name:</td><td></td></label<>	.for="fullname" class="form-label">Full Name:	
	28	<input< td=""><td>: type="text" id="fullname" name="fullname" class="form-control" autofo</td><td>cus required&gt;</td></input<>	: type="text" id="fullname" name="fullname" class="form-control" autofo	cus required>
	29			
	30	<label< td=""><td>.for="email" class="form-label"&gt;EMail:</td><td></td></label<>	.for="email" class="form-label">EMail:	
	31	<input< td=""><td><pre>: type="email" id="email" name="email" class="form-control" required&gt;</pre></td><td></td></input<>	<pre>: type="email" id="email" name="email" class="form-control" required&gt;</pre>	
	32			
	33	<label< td=""><td>. for="password" class="form-label"&gt;Password:</td><td></td></label<>	. for="password" class="form-label">Password:	
	34	<input< td=""><td>: type="password" id="password" name="password" class="form-control" re</td><td>quired&gt;</td></input<>	: type="password" id="password" name="password" class="form-control" re	quired>
	35			
	36			
	37	<input< td=""><td>: type="submit" value="Save" class="btn btn-success"&gt;</td><td></td></input<>	: type="submit" value="Save" class="btn btn-success">	
	38			
	39			
	40			
Ø	41			
U.	42			
203	43			
× 1	Restricted h			In 1 Col 1 Spaces: 2 LITE-8 CRLE PHP

## Save in DB

```
9 $email = $_POST["email"];
//Get Post Data
                                                                         12
$fullname = $ POST["fullname"];
                                                                         13 // Insert Data
$email = $ POST["email"];
                                                                         16
$password = $ POST["password"];
                                                                         18
                                                                         19
                                                                         20
                                                                            else {
$password_hashed = password_hash($password, PASSWORD_DEFAULT);
                                                                         21
                                                                         22
                                                                         23
                                                                           // Close Connection
                                                                         24
// Insert Data
                                                                            mysqli close($conn);
                                                                         26
                                                                            - ?>
                                                                         27
$sql = "INSERT INTO PERSON (FullName, EMail, Pwd)
                                                                            <!DOCTYPE html>
                                                                         29
                                                                            <html>
VALUES ('$fullname', '$email', '$password hashed')";
                                                                         30
                                                                         31
                                                                            <head>
                                                                         32
                                                                             <title>Save Book</title>
                                                                         33
                                                                             <meta charset="utf-8">
                                                                         34
if (mysqli_query($conn, $sql)) {
                                                                         35
                                                                         36
                                                                         37
                                                                            </head>
  $message = "New person created successfully.<br>";
                                                                         38
                                                                         39
                                                                            <body>
                                                                         41
                                                                             <h1>Save Person</h1>
                                                                         42
else {
                                                                         43
                                                                         44
  $message = "Error: " . $sql . "<br>>" . mysqli error($conn);
                                                                         45
                                                                         46
                                                                         47
  $password hashed = password hash($password, PASSWORD DEFAULT);
```

X File Edit Selection View 88 - $\Box \times$ □ … m person\_create.php m person\_create\_db.php × C: > Users > hansp > utorials > PHP > Development > Create User and Login > 🏘 person create db.php 1 k?php 2 require once 'config.php'; 3 require once 'functions.php'; 4 session\_start(); 5 checkUser(); //Get Post Data 8 \$fullname = \$ POST["fullname"]; 10 \$password = \$\_POST["password"]; 11 \$password hashed = password hash(\$password, PASSWORD DEFAULT); 14 \$sql = "INSERT INTO PERSON (FullName, EMail, Pwd) VALUES ('\$fullname', '\$email', '\$password hashed')"; 17 if (mysqli query(\$conn, \$sql)) { \$message = "New person created successfully.<br>"; \$message = "Error: " . \$sql . "<br>>" . mysqli error(\$conn); <meta name="viewport" content="width=device-width, initial-scale=1"> <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/css/bootstrap.min.css" rel="stylesheet"> <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/js/bootstrap.bundle.min.js"></script> <div class="container-fluid pt-5"> <?php echo \$message ?><br> <a href="persons.php" class="btn btn-success">Go back to Persons</a>

## Update Person

•	0	Library A	pp	×	S Update Personal Information	×	+			_		×
÷	$\rightarrow$	C	010	web01.usn.no/~har	nsha/person_update.php?person	id=1		©"	☆		Error	:

#### **Update Personal Information**

Please update person information:

Full Name:

Hans-Petter Halvorsen

EMail:

hans.p.halvorsen@usn.no

Password:

••••••

Save

88~ person\_update.php × 🥪 Courses > Webutvikling > Tutorials > PHP > Development > Create User and Login > 🤫 person update.php //Get GET Data \$personid = \$ GET["personid"]; 8 9 10 // Get spesific Book Data from Database 11 \$sql = "SELECT FullName, EMail, Pwd FROM PERSON WHERE PersonId = \$personid"; 12 \$result = mysqli\_query(\$conn, \$sql); ₽<sup>0</sup> 13 14 if (mysali num rows(\$result) > 0) { \$row = mysqli\_fetch\_assoc(\$result); 15 \$fullname = \$row["FullName"]; 16 17 \$email = \$row["EMail"]; \$pwd = \$row["Pwd"]; 18 19 else { echo "0 results"; 21 23 ?> 24 <!DOCTYPE html> 26 <html> 27 28 <head> 29 <title>Update Personal Information</title> 30 <meta charset="utf-8"> <meta name="viewport" content="width=device-width, initial-scale=1"> 31 <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/css/bootstrap.min.css" rel="stylesheet"> <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/js/bootstrap.bundle.min.js"></script> 34 </head> 35 36 <body> <div class="container-fluid pt-5"> 38 <h1>Update Personal Information</h1> 40 Please update person information: 41 42 <div class=form-group"> <form action="person\_update\_db.php?personid=<?php echo \$personid ?>" method="POST"> 43 <label for="fullname" class="form-label">Full Name:</label> 44 45 <input type="text" id="fullname" name="fullname" class="form-control" value="<?php echo \$fullname ?>"> 46 <label for="email" class="form-label">EMail:</label> 47 48 <input type="email" id="email" name="email" class="form-control" value="<?php echo \$email ?>"> 49 <label for="password" class="form-label">Password:</label> 50 51 <input type="password" id="password" name="password" class="form-control" value=""> 52 53 <br> 54 <input type="submit" value="Save" class="btn btn-success"> 55 </form> 56 </div> 57 58 <?php

⑦ Restricted Mode ⊗ 0 △ 0 № 0

□ …

Ln 1, Col 1 Spaces: 2 UTF-8 CRLF PHP Q

## Update DB

	10	//dec Post Data		
	11	<pre>\$fullname = \$_POST["fullname"];</pre>		
	12	<pre>\$email = \$_POST["email"];</pre>		
< ?nhn	13	<pre>\$password = \$_POST["password"];</pre>		
· · P · P	14	<pre>\$password_hashed = password_hash(\$password, PASSWORD_DEFAULT);</pre>		
	15			
//Get GET Data	16	// Update Person Data		
	1/	<pre>\$sq1 = OPDAIE PERSON SET FUIINAME= \$TUIINAME, EMAIL= \$email, Pwd= \$</pre>		
<pre>\$personia = \$_GET["personia"];</pre>	10	if (mysali aueny(\$conn \$sal)) {		
	20	<pre>\$message = "Personal information undated successfully.</pre>		
//Get Bost Data	21	}		
	22	else {		
<pre>\$tuliname = \$_POST["tuliname"];</pre>	23	<pre>\$message = "Error: " . \$sql . " &gt;" . mysqli_error(\$conn);</pre>		
<pre>\$email = \$ POST["email"]:</pre>	24	}		
	25			
<pre>\$password = \$_POST["password"];</pre>	26	// Close Connection		
password hashed = password hash( $password$ , PASSWORD DEFAULT);	27	<pre>mysqli_close(\$conn);</pre>		
	28	b		
	29			
// Update Person Data	31	<pre></pre>		
<pre>\$sal = "UPDATE PERSON SET FullName='\$fullname'.</pre>	32	(item)		
	33	<head></head>		
EMail='\$email', Pwd='\$password_hashed' WHERE Personid =	34	<title>Personal Information</title>		
<pre>\$personid";</pre>	35	<meta charset="utf-8"/>		
if (mysali query(\$conn \$sal)) {	36	<pre><meta content="width=device-width, initial-scale=1" name="viewport"/></pre>		
	37	<pre><link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/css/bc&lt;/pre&gt;&lt;/th&gt;&lt;/tr&gt;&lt;tr&gt;&lt;th&gt;&lt;pre&gt;\$message = " information="" personal="" successfully.<br="" updated=""/>";</pre>	38	<pre><script container-fluid="" nt-5"="" src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/js/bc&lt;/pre&gt;&lt;/th&gt;&lt;/tr&gt;&lt;tr&gt;&lt;th&gt;3&lt;/th&gt;&lt;th&gt;39&lt;/th&gt;&lt;th&gt;&lt;/head&gt;&lt;/th&gt;&lt;/tr&gt;&lt;tr&gt;&lt;th&gt;&lt;/th&gt;&lt;th&gt;40&lt;/th&gt;&lt;th&gt;al and the&lt;/th&gt;&lt;/tr&gt;&lt;tr&gt;&lt;th&gt;else {&lt;/th&gt;&lt;th&gt;41&lt;/th&gt;&lt;th&gt;&lt;pre&gt;&lt;dux class="></script></pre>

```
8.~
                                                                                                                  🗙 File Edit Selection View
                                                                                                                                      ···· 🗉
ŋ
     person_update.php
                        mail: # person_update_db.php *
      C: > Users > hansp > One
                                 Webstwiking >= dtorials > PHP > Development > Create User and Login > 🌸 person_update_db.php
 Q
        1 <?php
        2 require once 'config.php';
z
        3 require_once 'functions.php';
        4 session_start();
        5 checkUser();
à
        6
        7 //Get GET Data
8
        8 $personid = $_GET["personid"];
        9
       10 //Get Post Data
                                                                                    password_hashed' WHERE PersonId = $personid";
                                                                                    otstrap.min.css" rel="stylesheet">
                                                                                    otstrap.bundle.min.js"></script>
                                                                                                            Ln 1, Col 1 Spaces: 2 UTF-8 CRLF PHP Q
```

#### https://www.halvorsen.blog

#### Create User page and Login page in PHP

# Create Login Page

#### Hans-Petter Halvorsen



## Library App

S Library App × +	×	
← → C 😋 web01.usn.no/~hansha/library.php	🛠 😩 Error :	
Library This is the Library App. Here you can see avalible books, create new books, etc.	Login	
	✓ S Login × +	- 🗆 ×
+ \\   //+	← → ♂ ℃ veb01.usn.no/~hansha/login.php	🖙 🖈 💄 Error 🗄
	Login Please enter your personal information:	
You need to Login into the system to get access to the functionality.	EMail:	
	hans.p.halvorsen@usn.no Password:	
	••••••	

<b>X</b> F	ile Edit Selection	n View Go Run Terminal Help ← → <u> </u>
Ð	🏶 library.php 🗙	
0	C: > Temp > PHP	
~	1	<pre><!--DUCIYPE ntml--></pre>
Po	2	<html></html>
æ	3	
₿	4	<head></head>
	5	<title>Library App</title>
	6	<meta charset="utf-8"/>
	7	<meta content="width=device-width, initial-scale=1" name="viewport"/>
	8	<pre><link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/css/bootstrap.min.css" rel="stylesheet"/></pre>
	9	<pre><script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/js/bootstrap.bundle.min.js"></script></pre>
	10	
	11	
	12	<body></body>
	13	<pre><div class="container-fluid pt-5"></div></pre>
	14	
	15	<h1>Librarv</h1>
	16	
	17	$\langle p \rangle$ This is the Library App. Here you can see avalible books, create new books, etc. $\langle p \rangle$
	18	
	19	<pre></pre>
	20	<a class="btn btn-danger" href="login.php">Login</a>
	21	
	22	
	23	<pre><img alt="Books" height="300" src="images/books.png" width="600"/> </pre>
	24	
	25	xp>You need to Login into the system to get access to the functionality.
	26	
	27	
0	28	
S	20	<pre></pre>
5	29	

✓ 🖲 Restricted Mode 🛛 🛇 🛆 🛈

Ln 20, Col 45 Spaces: 2 UTF-8 CRLF PHP 🔀 💭

<b>(</b> F	ile Edit Sel	lection View Go Run $\cdots$ $\leftarrow$ $ o$ (	⊘ Search		<b>8</b> ~			-	$\Box$ $\times$			
ß	🖙 login.php	) ×							□ …			
	C: > Users >	hansp > OneDrive > Courses > Webutvikling > Tutorials	s > PHP > Development > Create User and Login > 🌸 login.php								o nh	
ρ	1	html										
Ŷ٥	2	<html></html>								LUSI		
٣	3									0		
	4	<head></head>										
	5	<title>Login</title>										
Б	6	<meta charset="utf-8"/>										
	7	<meta co<="" name="viewport" td=""/> <td><pre>ntent="width=device-width, initial-scale=1</pre></td> <td>."&gt;</td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td>	<pre>ntent="width=device-width, initial-scale=1</pre>	.">								
	8	<link <="" href="https://cdn.&lt;/td&gt;&lt;td&gt;jsdelivr.net/npm/bootstrap@5.3.3/dist/css/&lt;/td&gt;&lt;td&gt;boot&lt;/td&gt;&lt;td&gt;strap.mir&lt;/td&gt;&lt;td&gt;n.css" td=""/> <td>rel="style</td> <td>eshee</td> <td>t"&gt;</td> <td></td> <td></td> <td></td>	rel="style	eshee	t">							
	9	<script src="&lt;u&gt;https://cdn&lt;/u&gt;&lt;/td&gt;&lt;td&gt;.jsdelivr.net/npm/bootstrap@5.3.3/dist/js/&lt;/td&gt;&lt;td&gt;boot&lt;/td&gt;&lt;td&gt;strap.bur&lt;/td&gt;&lt;td&gt;ndle.m&lt;/td&gt;&lt;td&gt;in.js"></script>										

## **Check Login**

#### password\_verify(\$password, \$pwd\_db)

```
//Get Post Data
$email = $_POST["email"];
$password = $_POST["password"];
```

```
// Get Password from Database
$sql = "SELECT Pwd FROM PERSON WHERE EMail = '$email'";
$result = mysqli_query($conn, $sql);
```

```
if (mysqli_num_rows($result) > 0) {
    $row = mysqli_fetch_assoc($result);
    $pwd_db = $row["Pwd"];
} else {
```

```
echo "0 results";
```

```
}
```

```
if (password_verify($password, $pwd_db)) {
    $_SESSION["LoggedIn"] = TRUE;
    $message = "You have been successfully logged in ..";
}
else {
```

```
$_SESSION["LoggedIn"] = FALSE;
$message = "Something went wrong! Make sure to check ...";
```

```
File Edit Selection View \cdots \leftarrow \rightarrow
                                                                                                    Π …
  M login.php
                 m login_db.php ×
  C: > Users > hansp > OneDrive > Courses > Webutvikling > Tutorials > PHP > Development > Create User and Login > 🦛 login_db.php
         <?php
         require_once 'config.php';
          session_start();
         //Get Post Data
         $email = $_POST["email"];
         $password = $ POST["password"];
    10
         // Get Password from Database
    11
         $sq1 = "SELECT Pwd FROM PERSON WHERE EMail = '$email'";
    12
         $result = mysqli query($conn, $sql);
    13
    14
         if (mysqli num rows($result) > 0) {
           $row = mysqli_fetch_assoc($result);
    15
           $pwd db = $row["Pwd"];
    16
    17
           else
    18
           echo "0 results":
    19
    20
    21
         if (password_verify($password, $pwd db)) {
           $ SESSION["LoggedIn"] = TRUE;
    22
    23
           $message = "You have been successfully logged in to the system";
    24
    25
         else {
           $ SESSION["LoggedIn"] = FALSE;
    26
    27
           $message = "Something went wrong! Make sure to check that you have written your EMail and Password correctly.";
    28
    29
    30
         // Close Connection
         mysqli close($conn);
    32
         25
    33
    34
         <!DOCTYPE html>
    35
         <html>
    36
    37
          cheads
    38
           <title>Login</title>
    39
           <meta charset="utf-8">
    40
           <meta name="viewport" content="width=device-width, initial-scale=1">
    41
           <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/css/bootstrap.min.css" rel="stylesheet">
    42
           <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/js/bootstrap.bundle.min.js"></script>
    43
         </head>
    11
          <body>
          <div class="container-fluid pt-5">
    47
    48
           <h1>Login</h1>
    49
    50
           <?php echo $message ?><br><br>
    51
    52
           <a href="library.php" class="btn btn-success">Go back to Library page</a>
    53
    54
         </div>
    55
         </body>
    56
         </html>
```

```
R Destricted Marks (0.0 A.0. Mto)
```

## Using Session variables

- We need to store information whether the User is logged in or not when the user enters the different pages.
- We can use Session variables to share that information between multiple web pages.

```
session_start();
..
$_SESSION["LoggedIn"] = TRUE;
```

#### https://www.halvorsen.blog

Create User page and Login page in PHP

# Check if User is Logged in

Hans-Petter Halvorsen

Table of Contents

## Check if User is Logged in

- We don't want the user to get access to different web pages if he has not logged in.
- In all other PHP files, perform a check in the beginning whether the user is logged in or not (check if the session variable is true).

```
<?php
require_once 'config.php';
session_start();

$loggedin = FALSE;
if (isset($_SESSION["LoggedIn"]))
    $loggedin = $_SESSION["LoggedIn"];
?>
```

## Check if User is Logged in



The Library page will have different appearance/functionality depending on the user is logged in or not.





× .	ne con	Selection	7-30	arch		ы Ш <del>П</del>		0	
ſ	🕈 librar	y.php ×							ш
5	0.21160	are ) hance	OneDrive > Courses > Webutvikling > Tutorials > PHP >	Development > Create User and Lo	ain > 🐄 I	ibrary.php			
0	1	php</th <th></th> <th></th> <th></th> <th></th> <th></th> <th></th> <th></th>							
$\mathcal{P}$	2	require	once 'config.php';						
0.	3	session	start();						
P	4								
	5	\$logged	= FALSE;						
	6	if (iss	(\$ SESSION["LoggedIn"]))						
ŭ	7	\$logg	lin = \$ SESSION["LoggedIn"];						
	8								
Ъ	9	?>							
	10								
	11	DOCTY</th <th>html&gt;</th> <th></th> <th></th> <th></th> <th></th> <th></th> <th></th>	html>						
	12	<html></html>							
	13								
	14	<head></head>							
	15	<title< th=""><th>Library App</th><th></th><th></th><th></th><th></th><th></th><th></th></title<>	Library App						
	16	<meta< th=""><th>harset="utf-8"&gt;</th><th></th><th></th><th></th><th></th><th></th><th></th></meta<>	harset="utf-8">						
	17	<meta< th=""><th>name="viewport" content="width=device-wi</th><th>ith, initial-scale=1"&gt;</th><th></th><th></th><th></th><th></th><th></th></meta<>	name="viewport" content="width=device-wi	ith, initial-scale=1">					
	18	<link< th=""><th><pre>iref="https://cdn.jsdelivr.net/npm/boots</pre></th><th>crap@5.3.3/dist/css/boot</th><th>strap.m1</th><th>n.css" re</th><th>el="styl</th><th>esheet</th><th>t"&gt;</th></link<>	<pre>iref="https://cdn.jsdelivr.net/npm/boots</pre>	crap@5.3.3/dist/css/boot	strap.m1	n.css" re	el="styl	esheet	t">
	19	<scr1< th=""><th><pre>src="nttps://cdn.jsdelivr.net/npm/boot</pre></th><th>strap@5.3.3/dist/js/boot</th><th>strap.bu</th><th>ndie.min.</th><th><u>.]s</u>"&gt;<th>cript</th><th>&gt;</th></th></scr1<>	<pre>src="nttps://cdn.jsdelivr.net/npm/boot</pre>	strap@5.3.3/dist/js/boot	strap.bu	ndie.min.	<u>.]s</u> "> <th>cript</th> <th>&gt;</th>	cript	>
	20								
	21	(conint)							
	22	vser ipe							
	24	<th></th> <th></th> <th></th> <th></th> <th></th> <th></th> <th></th>							
	25	17 Dei 10							
	26	<body></body>							
	27	<div cl<="" th=""><th>s="container-fluid pt-5"&gt;</th><th></th><th></th><th></th><th></th><th></th><th></th></div>	s="container-fluid pt-5">						
	28								
	29	<h1>Lib</h1>	ary						
	30								
	31	This	s the Library App. Here you can see ava	lible books, create new	books, e	tc.			
	32								
	33	<p alig<="" th=""><th>"right"&gt;</th><th></th><th></th><th></th><th></th><th></th><th></th></p>	"right">						
	34	php i</th <th>(\$loggedin==TRUE)</th> <th></th> <th></th> <th></th> <th></th> <th></th> <th></th>	(\$loggedin==TRUE)						
	35	{							
	20	echo	a nret= logout.pnp class= btn btn-dang	<pre>&gt;Logout ;</pre>					
	38	∫ else							
	39	echo "«	href='login.php' class='htn btn-danger'	login":					
	40	2>	······						
	41								
	42								
	43	<img sr<="" th=""/> <th>"images/books.png" alt="Books" width="6</th> <th>0" height="300"&gt;</th> <th></th> <th></th> <th></th> <th></th> <th></th>	"images/books.png" alt="Books" width="6	0" height="300">					
	44								
	45	php i</th <th>(\$loggedin==TRUE)</th> <th></th> <th></th> <th></th> <th></th> <th></th> <th></th>	(\$loggedin==TRUE)						
	46	{							
	47	echo	a href='books.php' class='btn btn-succe	ss'>List of Books";					
	48	1							
	49	else				11-21			
	50	ecno	pyrou need to Login into the system to	ger access to the function	marity.				
	52	12							
	53	Onhn							
8	54	// Close	Connection						
	55	mysali	.ose(\$conn):						
503	56	?>							
ŝ	57								
× (	Restricted	d Mode 🛛 🛞	▲ 0 🐄 0		Ln 1, Col 1	Spaces: 2	UTF-8	CRLF PI	HP 🗘

## Library page Code cont.



```
<?php if ($loggedin==TRUE)</pre>
```

```
echo "<a href='logout.php' class='btn btn-danger'>Logout</a>";
else
echo "<a href='login.php' class='btn btn-danger'>Login</a>";
22
```

<img src="images/books.png" alt="Books" width="600" height="300"><br>

```
<?php if ($loggedin==TRUE)</pre>
```

echo "<a href='books.php' class='btn btn-success'>List of Books</a>";

#### else

echo "You need to Login into the system to get access to the functionality.";

• •

## **Books** page

The same check needs to be done in the different Books pages, etc.

```
🔀 File Edit Selection View Go Run …
                                        \leftarrow \rightarrow
     💏 books.php
                  💏 functions.php
     C: > Users > hansp > OneDrive > Courses > Webutvikling > Tutorials > PHP > Development > Create User and
ρ
             <?php
         1
             require once 'config.php';
z
             require_once 'functions.php';
         3
             session start():
à
         4
         5
             checkUser();
品
         6
             20
         7
             <!DOCTYPE <pre>html>
         8
             <html>
         9
        10
        11
             <head>
        12
               <title>Books App</title>
        13
               <meta charset="utf-8">
               <meta name="viewport" content="width=device-</pre>
        14
               <link href="https://cdn.jsdelivr.net/npm/boo</pre>
        15
                <script src="https://cdn.jsdelivr.net/npm/bo</pre>
        16
        17
             </head>
```

∢	$\equiv  \leftarrow \rightarrow$	♀ Search
C)	🐄 books.php	🗣 functions.php 🔀 🛄 …
	C: > Users > hansp	> OneDrive > Courses > Webutvikling > Tutorials > PHP > Development > Create User and
P	1 p</th <th>hp</th>	hp
80	2	
2	3 fur	<pre>nction checkInput(\$data) {</pre>
	4 \$	data = trim(\$data);
~	5 \$	data = stripslashes(\$data);
B	6 \$	data = htmlspecialchars(\$data);
	7 r	eturn \$data;
	8 }	
	9	
	10 fur	<pre>nction checkUser() {</pre>
	11 \$	<pre>Sloggedin = FALSE;</pre>
	12 i	f (isset(\$_SESSION["LoggedIn"]))
	13	<pre>\$loggedin = \$_SESSION["LoggedIn"];</pre>
	14	
	15 i	f (\$loggedin!=TRUE)
	16	<pre>header("Location: login.php");</pre>
	17 }	
8	18	
	19 ?>	
202		
× .	TRestricted Mode	0 △ 0 🖗 0 Ln 1, Col 1 Spaces: 2 UTF-8 CRLF PHP 🚨

### Logout page

Here you see the code for the "logout.php" page:



https://www.halvorsen.blog

Create User page and Login page in PHP

# Improvements and Additional Resources

Hans-Petter Halvorsen

**Table of Contents** 

## Library Application with Login



#### Improvements

There are lots of improvements to be made, some examples are:

- Check if "logged in" in all pages.
- 2-Facor Authentication.
- Authorization.
- +++ like GUI/HMI, Code Structure and Quality,

#### https://www.halvorsen.blog

#### Create User page and Login page in PHP

# 2-Factor Authentication

#### Hans-Petter Halvorsen

**Table of Contents** 

## **2-Factor Authentication**

- To make your authentication even better you can also consider implementing so-called 2-Factor Authentication.
- This can be done in different ways.
- The easiest way in PHP is to use email.
- You can use the built-in **email()** function in PHP
- The procedure is then to generate and send a verification code on email from your web application during the login process. Then the user needs to enter the verification code received on email during the login process.

```
$verificationcode = strtoupper(uniqid());
```

mail(\$to, \$subject, \$message, \$headers);

#### **2-Factor Authentication**



#### https://www.halvorsen.blog

#### Create User page and Login page in PHP

# Authorization

#### Hans-Petter Halvorsen



## Authentication vs Authorization

- Authentication is a process to authenticate a user, that is, to verify that someone is who they say they are.
- Authorization is about determining a user's level of access and then granting access based on that level.

So far, we have focused on Authentication in this tutorial. How can we implement Authorization?

In out "Library" application we may want to have different types of users, examples:

- Administrators
  - They can add new Users, edit and delete Users.
- Librarians
  - They can create new Books, edit or delete books.
  - They can see information regarding book rentals, etc.
- Customers
  - They have access to book information, but they cannot create new books, edit or delete books
  - They can rent books, see if books are available or not

=>This can be implemented in different ways, but typically you need to start to add one or more new tables or columns in the database to handle this in your application.

Simplified Example of Authorization :

#### User



Username:

Johnny

Password:

Password123

"Customer" Access Level: If User = "User X", he gets access to read data only.

"Librarian" Access Level: If User = "User Y", he gets access to creating new books, etc.

"Administrator" Access Level: If User = "User Z", he is an "Administrator" and can create, edit and delete users.

Database

Here you see a basic example how to implement Authorization in the Database:



A specific User can be part of only one Access Level in this example Access Levels Examples:

- Customer
- Librarian
- Administrator

Here you specify the Features the different Access Levels have access to Features Examples:

Database

- Search
- Create Books
- Delete User
  - •

٠

#### https://www.halvorsen.blog

#### Create User page and Login page in PHP

# Summary

#### Hans-Petter Halvorsen



## Summary

← → C 😂 web01.usn.no/~hansha/library.php	Erro	r :

#### Library

This is the Library App. Here you can see avalible books, create new books, etc.



You need to Login into the system to get access to the functionality.

ſ	•	0	Library A	hpp	×	Create Person	×	+		-		×	
	←	$\rightarrow$	C	0-0	web01.usn.no/~har	nsha/person_create.php			©≂ ☆		E	irror :	

#### **New Person**

Please enter the personal information:
Full Name:
EMail:

We have created a basic "Library" PHP Web Application. The Application shows a list of Books, but only after the User has logged into the system. We have also created basic User Administration where we can add, delete and modify Users.



#### Name t .. 📒 images book\_create.php book\_create\_db.php book\_delete.php book\_update.php book\_update\_db.php books.php config.php functions.php library.php Iogin.php login\_db.php Iogout.php person\_create.php person\_create\_db.php person\_delete.php person\_update.php person\_update\_db.php persons.php

Password:

## **Resources and References**

- PHP Tutorial w3school: <u>https://www.w3schools.com/php/</u>
- PHP Tutorial TutorialsPoint: <u>https://www.tutorialspoint.com/php/</u>
- PHP Documentation: <u>https://www.php.net/manual/en/</u>
- MySQL Tutorial: <u>https://www.w3schools.com/mysql</u>
- Bootstrap Tutorial: <u>https://www.w3schools.com/bootstrap5/</u>

#### Hans-Petter Halvorsen

**University of South-Eastern Norway** 

www.usn.no

E-mail: <u>hans.p.halvorsen@usn.no</u> Web: <u>https://www.halvorsen.blog</u>

