

Creating a Navigation Menu Based on User Roles

In the previous tutorial, you created a role-based access control system using admin, teacher and student accounts. In this tutorial, you will build a navigation menu that changes depending on the role of the logged-in user.

This allows different users to see different menu options.

Current Situation

At the moment, every user sees the same page after logging in.

Example:

```
Welcome, admin
```

```
Role: admin
```

```
Logout
```

We can improve this by displaying different navigation links based on the user's role.

Create a Navigation Section

Open:

```
members.php
```

Add the following code underneath the welcome message:

```
<h2>Navigation</h2>

<ul>

    <li>
        <a href="members.php">
            Home
        </a>
    </li>

</ul>
```

The page should now display a simple menu.

Members Area

Welcome, admin

Role: admin

Navigation

- [Home](#)

Add an Admin Link

Add the following code inside the navigation list:

```
<?php if ($_SESSION["role"] == "admin") { ?>

<li>
    <a href="admin.php">
        Admin Area
    </a>
</li>
```

```
<?php } ?>
```

This link will only appear for administrators.

Create a Teacher Page

Create a new file called:

```
teacher.php
```

Add the following code:

```
<?php

session_start();

if (!isset($_SESSION["user_id"])) {

    header("Location: login.php");
    exit();

}

if ($_SESSION["role"] != "teacher") {

    die("Access denied.");

}

?>

<!DOCTYPE html>
<html>
<head>
    <title>Teacher Area</title>
</head>
<body>
```

```
<h1>Teacher Area</h1>

<p>Welcome,
<?php echo $_SESSION["username"]; ?>
</p>

</body>
</html>
```

Save the file.

Add a Teacher Link

Add the following code to your navigation menu:

```
<?php if ($_SESSION["role"] == "teacher") { ?>

<li>
  <a href="teacher.php">
    Teacher Area
  </a>
</li>

<?php } ?>
```

Only teachers will see this link.

Create a Student Page

Create a new file called:

```
student.php
```

Add:

```
<?php

session_start();

if (!isset($_SESSION["user_id"])) {

    header("Location: login.php");
    exit();

}

if ($_SESSION["role"] != "student") {

    die("Access denied.");

}

?>

<!DOCTYPE html>
<html>
<head>
    <title>Student Area</title>
</head>
<body>

<h1>Student Area</h1>

<p>Welcome,
<?php echo $_SESSION["username"]; ?>
</p>

</body>
</html>
```

Save the file.

Add a Student Link

Add:

```
<?php if ($_SESSION["role"] == "student") { ?>

<li>
    <a href="student.php">
        Student Area
    </a>
</li>

<?php } ?>
```

Only students will see this link.

Add a Logout Link

Add:

```
<li>
    <a href="logout.php">
        Logout
    </a>
</li>
```

This link should be visible to all logged-in users.

Complete Navigation Menu

Your completed navigation menu should look like:

```
<h2>Navigation</h2>

<ul>
```

```
<li>
    <a href="members.php">
        Home
    </a>
</li>
```

```
<?php if ($_SESSION["role"] == "admin") { ?>
```

```
<li>
    <a href="admin.php">
        Admin Area
    </a>
</li>
```

```
<?php } ?>
```

```
<?php if ($_SESSION["role"] == "teacher") { ?>
```

```
<li>
    <a href="teacher.php">
        Teacher Area
    </a>
</li>
```

```
<?php } ?>
```

```
<?php if ($_SESSION["role"] == "student") { ?>
```

```
<li>
    <a href="student.php">
        Student Area
    </a>
</li>
```

```
<?php } ?>
```

```
<li>
    <a href="logout.php">
        Logout
```

```
</a>
</li>

</ul>
```

Test as an Administrator

Log in as:

admin

You should see:

Home
Admin Area
Logout

Test as a Teacher

Log in as:

teacher

You should see:

Home
Teacher Area
Logout

Test as a Student

Log in as:

student

You should see:

```
Home
Student Area
Logout
```

Prevent Direct Access

The navigation menu improves the user experience, but it does not secure the pages.

The following checks should still exist in:

```
admin.php
teacher.php
student.php
```

Example:

```
if ($_SESSION["role"] != "admin") {

    die("Access denied.");

}
```

This prevents users from manually typing the page URL into their browser.

Next Steps

You now have:

- User registration
 - Password hashing
 - Login system
 - Sessions
 - Protected pages
 - Logout functionality
 - Role-based access control
 - Dynamic navigation menus
-

Revision #1

Created 2026-06-08 09:09:42 UTC by Mr Napper

Updated 2026-06-08 09:15:40 UTC by Mr Napper