

CO7006 Web Systems – Assignment

Data Driven Website & Critique

Link to the website.....	p.2
Website structure.....	p.2
All code from the project.....	p.3 – 36
Critique and reference list.....	p.37 – 42

Link of the website

<http://2114907.linux.studentwebserver.co.uk/>

Website structure

	Webpage	Type	Introduction	Targeted user
Front-end	index.php	Data-driven content	Home page providing a search function to find recipes by ingredients. Data are retrieved from TheMealDB API.	Public
	inspiration.php	Data-driven content	Display four random recipes. Data are retrieved from TheMealDB API.	
	blog.php	Data-driven content	Display cooking tips. Data are retrieved from MySQL.	
	contact.php	Information display	Display contact information.	
	form.php	Insert data to database	A HTML form to let public submit their recipes. Inputted data will be stored in MySQL.	
Back-end	admin.php	Authentication system	Website content editor can login to the system to view the submitted recipes. Please use the following credentials to login: Username: editor Password: editor	Content Editor
	userSubmit.php	Authentication system	Display the submitted recipes which can only be accessed if logged in.	
	adminReg.php	Authentication system	For IT department staff to create Content Editor accounts.	IT department

All code from the project

index.php

```
<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <link rel="icon" href="image/icon.png">
    <style>
        @import
url('https://fonts.googleapis.com/css2?family=Akshar:wght@400;600&display=swap');
    </style>
    <style>
        @import
url('https://fonts.googleapis.com/css2?family=Lobster+Two&display=swap');
    </style>
    <link rel="stylesheet" type="text/css" href="style.css" />
    <title>Taste the World | Recipes of World Cuisines</title>
</head>

<body>
    <?php include('header.php') ?>

    <section id="intro">
        <h1 id="heroTitle" class="title">Search Recipes by Ingredient</h1>
        <div id="searchBox">
            <input type="text" class="search" placeholder="Enter an ingredient"
id="search-input">
            <button type="submit" id="searchButton">Search
            </button>
        </div>

        <div id="main-banner">
        </div>
    </section>
    <section id="searchSection" class="container">
        <h2 class="title" id="searchTitle">Search Results:</h2>
        <div id="searchResult">
            <!-- meal item start -->
            <!-- <div class="mealItem">
                <div class="sMealImg">
                    

                </div>
                <div class="sMealName">
```

```

<h3>Beef Steak</h3>
    <h2>American dish</h2>
    <a href="#" class="sMealButton">Get Instructions</a>
</div>
</div> -->
<!-- meal item end -->

</div>

<div class="sMealDetails container">
    <button type="button" id="sCloseDetail">X</button>

    <div class="sMealDetailsContent" id="sMealDetailsContent">
        <!-- <div class="sMealDerailsImg">
        </div>
        <h3 class="sMealDetailsName">Chicken</h3>
        <p class="sMealDetailsCat">Mexican food</p>
        <p class="sMealDetailsIns">Lorem ipsum, dolor sit amet consectetur
adipisicing elit. Tenetur ut,
        culpa,
        pariatum distinctio iste labore neque ad minus deserunt sed
quasi cum harum eius porro aperiam
        dolore,
        dolor delectus nihil!</p>

        <a href="#" class="sMealDetailsVideo">Watch the video</a> -->
    </div>

</div>
<p style="color: white; float: right; bottom: 20px; position:
relative;">(Banse, 2017)</p>
</section>

<footer>
    <div id="footerContent" class="container">
        <p>
            All recipe data is retrieved from <a
href="https://www.themealdb.com/" target="_blank" style="color:
aquamarine;">TheMealDB</a>.
        </p>
        <ul>
            <li style="font-size: 0.7em;">Banse, L. (2017). cooked dish on gray
bowl photo. Unsplash. Available at:
                https://unsplash.com/photos/-YHSwy6uqvk (Accessed: 4 May
2022).</li>
            <li style="font-size: 0.7em;">TheMealDB (2016). TheMealDB API.
Available at:

```

<https://www.themealdb.com/api.php> (Accessed: 10 May 2022).

```
        </ul>
    </div>
</footer>

<script src="script.js"></script>
</body>

</html>

<u>inspiration.php</u>
<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <link rel="icon" href="image/icon.png">
    <style>
        @import
url('https://fonts.googleapis.com/css2?family=Akshar:wght@400;600&display=swap');
    </style>
    <style>
        @import
url('https://fonts.googleapis.com/css2?family=Lobster+Two&display=swap');
    </style>
    <link rel="stylesheet" type="text/css" href="style.css" />
    <title>Taste the World | Recipes of World Cuisines</title>
</head>

<body>
    <?php include('header.php') ?>

    <section id="intro">
        <h1 id="heroTitle-s" class="title">Inspiration for your Meals</h1>
        <p id="heroTitle-intro">Don't know what to cook today? See the following
random recipes for meal ideas!</p>
        <div id="main-banner-s">
            <p style="color: white; float: right; top: 170px; position:
relative;">(Banse, 2017)</p>
        </div>

    </section>

    <section id="inspiration" class="container">

        <div id="mealInspiration">
            <div id="receipe1"></div>
```

```

        <div id="receipe2"></div>
        <div id="receipe3"></div>
        <div id="receipe4"></div>

    </div>

</section>
<footer>
    <div id="footerContent" class="container">
        <p>
            All recipe data is retrieved from <a
href="https://www.themealdb.com/" target="_blank" style="color:
aquamarine;">TheMealDB</a>.
        </p>
        <ul>
            <li style="font-size: 0.7em;">Banse, L. (2017). cooked dish on gray
bowl photo. Unsplash. Avaliable at:
                https://unsplash.com/photos/-YHSwy6uqvk (Accessed: 4 May
2022).</li>
            <li style="font-size: 0.7em;">TheMealDB (2016). TheMealDB API.
Avaliable at:
                https://www.themealdb.com/api.php (Accessed: 10 May 2022).</li>

        </ul>

    </div>
</footer>
<script src="recommendation.js"></script>

</body>

</html>

```

blog.php

```

<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <link rel="icon" href="image/icon.png">
    <style>
        @import
url('https://fonts.googleapis.com/css2?family=Akshar:wght@400;600&display=swap');
    </style>
    <style>
        @import
url('https://fonts.googleapis.com/css2?family=Lobster+Two&display=swap');
    </style>
    <link rel="stylesheet" type="text/css" href="style.css" />

```

```

<title>Taste the World | Recipes of World Cuisines</title>
</head>

<body>
    <?php include('header.php') ?>

    <section id="intro">
        <h1 id="heroTitle-s" class="title">Cooking Tips</h1>
        <div id="main-banner-s">
            <p style="color: white; float: right; top: 170px; position:
relative;">(Banse, 2017)</p>
        </div>

    </section>

    <section id="blog">
        <?php
            $conn = mysqli_connect('localhost', 'c2114907_admin', 'K3@onq%;}A.',
'c2114907_meals');
            if ($conn->connect_error) {
                echo ("DB connection failed ");
            }
            $sql = "SELECT title, content FROM blog";
            $result = $conn->query($sql);

            while ($row = $result->fetch_assoc()) {

                echo "<div class='post'><h2 class='blog-title'>" . $row["title"] .
"</h2><p class='blog-content'>" . $row["content"] . "</p></div>";
            }
            $conn->close();
        ?>

        <!--
        <div class="post">
            <h2 class="blog-title">Never use damp oven mitts.</h2>
            <p class="blog-content">Unless you'd like a fresh layer of skin, that
is.</p>
        </div>

        <div class="post">
            <h2 class="blog-title">Never use damp oven mitts.</h2>
            <p class="blog-content">Unless you'd like a fresh layer of skin, that
is.</p>
        </div> -->
    </section>

    <footer>
        <div id="footerContent" class="container">

```

```

        <ul>
            <li style="font-size: 0.7em;">Banse, L. (2017). cooked dish on gray
            bowl photo. Unsplash. Available at:
                https://unsplash.com/photos/-YHSwy6uqvk (Accessed: 4 May
                2022).</li>
            <li style="font-size: 0.7em;">Delish (2020). 15 Cooking Tips
            That'll Change Every Home Cook's Life. Available at:
            https://www.delish.com/kitchen-tools/kitchen-secrets/g25585978/best-cooking-tips/
            (Accessed: 15 May 2022).</li>
        </ul>
    </div>
</footer>

```

```

    <script src="script.js"></script>
</body>

</html>

```

contact.php

```

<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <link rel="icon" href="image/icon.png">
    <style>
        @import
url('https://fonts.googleapis.com/css2?family=Akshar:wght@400;600&display=swap');
    </style>
    <style>
        @import
url('https://fonts.googleapis.com/css2?family=Lobster+Two&display=swap');
    </style>
    <link rel="stylesheet" type="text/css" href="style.css" />
    <title>Taste the World | Recipes of World Cuisines</title>
</head>

<body>
    <?php include('header.php') ?>

    <section id="intro">
        <h1 id="heroTitle-s" class="title">Contact Us</h1>
        <div id="main-banner-s">
            <p style="color: white; float: right; top: 170px; position:
            relative;">(Banse, 2017)</p>
        </div>
    </section>

```



```

<section id="contact" class="container">
  <p id="web-intro">This is a website created for the assignment of C07006 -
Web Systems. It shares the recipes of
    meals from all
    around the world. The data of recipes is retrieved
    from a data source API built by <a href="https://www.themealdb.com/"
target="_blank">TheMealDB</a>.</p>
  <div>
    <table id="contact-info" class="container">
      <tr>
        <iframe
src="https://www.google.com/maps/embed?pb=!1m16!1m12!1m3!1d2907.006475370735!2d-
2.902031915251768!3d53.19969912358626!2m3!1f0!2f0!3f0!3m2!1i1024!2i768!4f13.1!2m1!1
suniversity%20of%20chester!5e0!3m2!1sen!2suk!4v1652628400517!5m2!1sen!2suk"
width="1000" height="300" style="border:0;" allowfullscreen="" loading="lazy"
referrerpolicy="no-referrer-when-downgrade"></iframe>
      </tr>
      <tr class="address-info">
        <th class="address-info">
          Address
        </th>
        <th class="address-info">
          University of Chester<br>
          Parkgate Road<br>
          Chester
          CH1 4BJ
        </th>
      </tr>
      <tr>
        <th class="email-info">
          Email
        </th>
        <th class="email-info">
          j94263@chester.ac.uk
        </th>
      </tr>
      <tr>
        <th class="phone-info">
          Phone
        </th>
        <th class="phone-info">
          +44 (0)1234567890
        </th>
      </tr>
    </table>
  </div>
</section>

<footer>
  <div id="footerContent" class="container">

```

```

        <ul>
            <li style="font-size: 0.7em;">Banse, L. (2017). cooked dish on gray
bowl photo. Unsplash. Available at:
                https://unsplash.com/photos/-YHSwy6uqvk (Accessed: 4 May
2022).</li>
        </ul>
    </div>
</footer>

    <script src="script.js"></script>
</body>

</html>

```

form.php

```

<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <link rel="icon" href="image/icon.png">
    <style>
        @import
url('https://fonts.googleapis.com/css2?family=Akshar:wght@400;600&display=swap');
    </style>
    <style>
        @import
url('https://fonts.googleapis.com/css2?family=Lobster+Two&display=swap');
    </style>
    <link rel="stylesheet" type="text/css" href="style.css" />
    <title>Taste the World | Recipes of World Cuisines</title>
</head>

<body>
    <?php include('header.php') ?>

    <section id="submitSection" class="container">
        <h1 class="title">Submit Your Recipe</h1>
        <form action="insert.php" method="POST" id="form">
            <div><input type="text" class="generalForm" id="submitMealName"
name="submitMealName" placeholder="Meal Name" required></div>

            <div><textarea class="generalForm" id="submitMealInstruct"
name="submitMealInstruct" placeholder="How to make the meal?" rows="10"
required></textarea></div>
            <div>
                <select class="generalForm" id="submitMealArea"
name="submitMealArea" required>

```

```

        <option value="" selected disabled hidden>Original
Area</option>
        <option value="American">American</option>
        <option value="British">British</option>
        <option value="Canadian">Canadian</option>
        <option value="Chinese">Chinese</option>
        <option value="Croatian">Croatian</option>
        <option value="Dutch">Dutch</option>
        <option value="Egyptian">Egyptian</option>
        <option value="French">French</option>
        <option value="Greek">Greek</option>
        <option value="Indian">Indian</option>
        <option value="Irish">Irish</option>
        <option value="Italian">Italian</option>
        <option value="Jamaican">Jamaican</option>
        <option value="Japanese">Japanese</option>
        <option value="Kenyan">Kenyan</option>
        <option value="Malaysian">Malaysian</option>
        <option value="Mexican">Mexican</option>
        <option value="Moroccan">Moroccan</option>
        <option value="Polish">Polish</option>
        <option value="Portuguese">Portuguese</option>
        <option value="Russian">Russian</option>
        <option value="Spanish">Spanish</option>
        <option value="Thai">Thai</option>
        <option value="Tunisian">Tunisian</option>
        <option value="Turkish">Turkish</option>
        <option value="Vietnamese">Vietnamese</option>
        <option value="Unknown">Unknown</option>
    </select>
</div>
    <div><input type="text" class="generalForm" id="submitMealVideo"
name="submitMealVideo" placeholder="YouTube Link for Cooking Tutorial Video
(Optional)"></div>
    <div><input id="submitButton" class="submitReceipe" type="submit"
value="Submit"></div>

</form>
</section>

<footer>
    <div id="footerContent" class="container">
        <p>
            Website built for C07006 – Web Systems
        </p>
    </div>
</footer>

<script src="script.js"></script>

```

```

</body>

</html>

admin.php
<?php session_start(); ?>
<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <link rel="icon" href="image/icon.png">
    <style>
        @import
url('https://fonts.googleapis.com/css2?family=Akshar:wght@400;600&display=swap');
    </style>
    <style>
        @import
url('https://fonts.googleapis.com/css2?family=Lobster+Two&display=swap');
    </style>
    <link rel="stylesheet" type="text/css" href="style.css" />
    <title>Admin</title>
</head>

<body>
    <header>
        <div id="top-header" class="container">
            <a href="index.php">
            </a>
        </div>
    </header>
    <section class="container">
        <h1>Backend Login</h1>
        <?php require_once 'msg.php'; ?>
        <form id="admin-login" action="login.php" method="POST">
            <div>
                <label for="username">Username: </label>
                <input type="text" name="username" id="username">
            </div>
            <div>
                <label for="password">Password: </label>
                <input type="password" name="password" id="password">

            </div>

            <div><button type="submit" name="login-btn" id="login-btn">Log
in</button></div>

```

```

        </form>
    </section>
</body>

</html>

adminReg.php
<?php session_start(); ?>
<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <link rel="icon" href="image/icon.png">
    <style>
        @import
url('https://fonts.googleapis.com/css2?family=Akshar:wght@400;600&display=swap');
    </style>
    <style>
        @import
url('https://fonts.googleapis.com/css2?family=Lobster+Two&display=swap');
    </style>
    <link rel="stylesheet" type="text/css" href="style.css" />
    <title>Admin</title>
</head>

<body>
    <header>
        <div id="top-header" class="container">
            <a href="index.php">
            </a>
            <a href="admin.php" class="button">Backend Login</a>

        </div>
    </header>
    <section class="container">
        <h1>Backend User Creation</h1>
        <?php require_once 'msg.php'; ?>
        <form id="admin-reg" action="signup.php" method="POST">
            <div>
                <label for="username">Username: </label>
                <input type="text" name="username" id="username">
            </div>
            <div>
                <label for="password">Password: </label>
                <input type="password" name="password" id="password">

            </div>
        </form>
    </section>
</body>
</html>

```

```

        <div>
            <label for="password">Enter the Password again: </label>
            <input type="password" name="passwordRepeat" id="passwordRepeat">

        </div>
        <div><button type="submit" name="signup-btn" id="signup-btn">Sign
up</button></div>
    </form>
</section>
</body>

</html>

```

userSubmit.php

```

<?php
session_start();

if (isset($_SESSION["login"]) !== TRUE) {
    header("location: admin.php");
    exit;
}
?>

<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <link rel="icon" href="image/icon.png">
    <style>
        @import
url('https://fonts.googleapis.com/css2?family=Akshar:wght@400;600&display=swap');
    </style>
    <style>
        @import
url('https://fonts.googleapis.com/css2?family=Lobster+Two&display=swap');
    </style>
    <link rel="stylesheet" type="text/css" href="style.css" />
    <title>Admin</title>
</head>

<body>
    <header>
        <div id="top-header" class="container">
            <a href="index.php">
            </a>
            <a href="logout.php" class="submitReceipe">Logout</a>
        </div>

```

```

</header>
<section class="container">
  <h1>Recipes submitted</h1>

  <table id="submittedData">
    <tr>
      <th>ID</th>
      <th>Meal Name</th>
      <th>Meal Instruction</th>
      <th>Meal Area</th>
      <th>Meal Video</th>
    </tr>

    <?php
      $conn = mysqli_connect('localhost', 'c2114907_admin', 'K3@onq%;}A.',
'c2114907_meals');
      if ($conn->connect_error) {
        echo ("DB connection failed ");
      }
      $sql = "SELECT id, submitMealName, submitMealInstruct, submitMealArea,
submitMealVideo FROM submit";
      $result = $conn->query($sql);

      while ($row = $result->fetch_assoc()) {

        echo "<tr><td>" . $row["id"] . "</td><td>" .
$row["submitMealName"] . "</td><td>" . $row["submitMealInstruct"] . "</td><td>" .
$row["submitMealArea"] . "</td><td><a target='_blank' href='" .
$row["submitMealVideo"] . "'>" . $row["submitMealVideo"] . "</a></td>";
      }
      $conn->close();
    ?>

    <!-- <tr>
      <td>1</td>
      <td>Egg custard tartad</td>
      <td>Lorem ipsum dolor sit amet, consectetur adipiscing elit. Cras
felis dui, ultricies sed dignissim ut,
tempor a dolor. Cras blandit ultricies purus a vestibulum. Sed
justo elit, auctor ac tortor nec,
scelerisque vehicula justo. Maecenas et nisl vel enim auctor
imperdiet. Proin in ante ex. </th>
      <td>British</td>
      <td>youtube.com</td>
    </tr> -->

  </table>

```

```

    </section>
</body>

```

```

</html>

```

header.php

```

<header>
    <div id="top-header" class="container">
        <a href="index.php">
        </a>
        <a href="form.php" class="submitReceipe">Submit Your Recipe</a>
    </div>
    <nav>
        <ul class="conrainer">
            <li><a href="index.php">Home</a></li>
            <li><a href="inspiration.php">Inspiration</a></li>
            <li><a href="blog.php">Cooking Tips</a></li>
            <li><a href="contact.php">Contact us</a></li>
        </ul>
    </nav>
</header>

```

login.php

```

<?php
session_start();
if (empty($_POST['username']) || empty($_POST['password'])) {
    $_SESSION['messages'][] = 'Please fill in all fields';
    header("location: admin.php");
    exit();
}

$conn = mysqli_connect('localhost', 'c2114907_admin', 'K3@onq%;}A.',
    'c2114907_meals');
if ($conn->connect_error) {
    echo ("DB connection failed");
}

if ($stmt = $conn->prepare('SELECT password FROM admin WHERE username =?')) {
    $stmt->bind_param('s', $_POST['username']);
    $stmt->execute();
    $stmt->store_result();

    if ($stmt->num_rows > 0) {
        $stmt->bind_result($password);
        $stmt->fetch();
    }
}

```



```

        if (password_verify($_POST['password'], $password)) {
            session_regenerate_id();
            header("location: userSubmit.php");
            $_SESSION['login'] = TRUE;
            // echo 'welcome!';
        }
    } else {
        $_SESSION['messages'][] = 'Incorrect username or password';
        header("location: admin.php");
        exit();
    }

    $stmt->close();
}

```

signup.php

```

<?php
session_start();

$username = $_POST['username'];
$password = $_POST['password'];
$passwordRepeat = $_POST['passwordRepeat'];

if (empty($username) || empty($password) || empty($passwordRepeat)) {
    $_SESSION['messages'][] = 'Please fill in all fields';
    header("location: adminReg.php");
    exit;
}

if ($password !== $passwordRepeat) {
    $_SESSION['messages'][] = 'Two passwords do not match';
    header("location: adminReg.php");
    exit;
}

$conn = mysqli_connect('localhost', 'c2114907_admin', 'K3@onq%;}A.',
    'c2114907_meals');
if ($conn->connect_error) {
    echo ("DB connection failed ");
}

if ($stmt = $conn->prepare('SELECT password FROM admin WHERE username=?')) {
    $stmt->bind_param('s', $username);
    $stmt->execute();
    $stmt->store_result();
    if ($stmt->num_rows > 0) {
        $_SESSION['messages'][] = 'user existed';
        header("location: adminReg.php");
    }
}

```

```

        exit;
    } else {
        if ($stmt = $conn->prepare('INSERT INTO admin (username, password) VALUES
(?, ?)')) {
            $hashedPassword = password_hash($_POST['password'], PASSWORD_DEFAULT);
            $stmt->bind_param('ss', $username, $hashedPassword);
            $stmt->execute();
            $_SESSION['messages'][] = 'User has been created';
            header("location: adminReg.php");
        } else {
            header("location: adminReg.php?e=wrong");
        }
    }
    $stmt->close();
}

```

insert.php

```

<?php
$submitMealName = $_POST['submitMealName'];
$submitMealInstruct = $_POST['submitMealInstruct'];
$submitMealArea = $_POST['submitMealArea'];
$submitMealVideo = $_POST['submitMealVideo'];

$conn = mysqli_connect('localhost', 'c2114907_admin', 'K3@onq%;}A.',
'c2114907_meals');
if ($conn->connect_error) {
    echo ("DB connection failed ");
} else {

    $stmt = $conn->prepare("insert into submit(submitMealName, submitMealInstruct,
submitMealArea, submitMealVideo) values(?, ?, ?, ?)");
    $stmt->bind_param("ssss", $submitMealName, $submitMealInstruct,
$submitMealArea, $submitMealVideo);
    $execval = $stmt->execute();
    echo $execval;
    echo "
    <!DOCTYPE html>
<html lang='en'>

<head>
    <meta charset='UTF-8'>
    <meta http-equiv='X-UA-Compatible' content='IE=edge'>
    <meta name='viewport' content='width=device-width, initial-scale=1.0'>
    <link rel='icon' href='image/icon.png'>
    <style>
        @import
url('https://fonts.googleapis.com/css2?family=Akshar:wght@400;600&display=swap');
    </style>

```

```

<style>
    @import
url('https://fonts.googleapis.com/css2?family=Lobster+Two&display=swap');
</style>
<link rel='stylesheet' type='text/css' href='style.css' />
<title>Taste the World | Recipes of World Cuisines</title>
</head>

<body>
<header>
<div id='top-header' class='container'>
    <a href='index.php'><img id='logo' src='image/logo.png' alt='Taste the World is
a website sharing recipes of the world dishes'>
    </a>
    <a href='form.php' class='submitReceipe'>Submit Your Recipe</a>
</div>
<nav>
    <ul class='conrainer'>
        <li><a href='index.php'>Home</a></li>
        <li><a href='inspiration.php'>Inspiration</a></li>
        <li><a href='blog.php'>Cooking Tips</a></li>
        <li><a href='contact.php'>Contact us</a></li>
    </ul>
</nav>
</header>

    <section id='submitted' class='container'>
        <div id='thankYou'>
            <h1>Thank you for submitting your recipe!</h1>
            <p>Our editors update recipes frequently. Check back to see if your
recipe is published!</p>
        </div>
    </section>

<footer>
    <div id='footerContent' class='container'>
        <p>
            Website built for C07006 – Web Systems
        </p>
    </div>
</footer>

    <script src='script.js'></script>
</body>

</html>

";
$stmt->close();

```

```

        $conn->close();
    }

```

logout.php

```

<?php
session_start();
$_SESSION = array();
session_destroy();
header("location: index.php");
exit;

```

msg.php

```

<?php
session_start();
if (empty($_SESSION['messages'])) {
    return;
}
$msg = $_SESSION['messages'];
unset($_SESSION['messages']);
foreach ($msg as $m) : ?>
    <p style="color: #c30000">
        <?php echo $m; ?>
    </p>
<?php endforeach; ?>

```

style.css

```

* {
    margin: 0;
    padding: 0;
    box-sizing: border-box;
}

```

```

img {
    display: block;
    max-width: 100%;
}

```

```

body{
    font-family: 'Akshar', sans-serif;
    color: #3b3b3b;
    min-height: 100vh;
    position: relative;
}

```

```

.title{
    font-family: 'Lobster Two', cursive;
    font-size: 2rem;
}

```

```

    text-align: center;
    margin: 0.5em auto;
}

#top-header{
    padding: 20px;
    display: flex;
    justify-content: space-between;
}

#logo{
    width: 300px;
}

.submitReceipe, .button{
    position: relative;
    text-decoration: none;
    margin-top: 1em;
    margin-bottom: 0.5em;
    width: 150px;
    background: #Ffb1a0;
    display: inline-block;
    padding: 0.5em;
    border-radius: 1em;
    border-bottom: #ff9b85 5px solid;
    color: #262626;
}

.submitReceipe:hover {
    border-bottom: #ff9b85 5px solid;
    top: 1px;
    background-color: #fffc5e;
    transition: all .2s ease-out;
}

.container {
    max-width: 1000px;
    margin: auto;
    text-align: center;
}

nav{
    background-color: #Ffb1a0;
    height: 50px;
}

```

```

nav a {
  color: #3b3b3b;
  letter-spacing: 1px;
  text-decoration: none;
  background: #Ffb1a0;
  padding: 16px 5px;
  display: inline-block;
  width: 100%;
  text-align: center;
}

nav a:hover {
  color: #070063;
  background: #fffc5e;
  font-weight: 600;
}


nav ul {
  background-color: #Ffb1a0;
  list-style: none;
  margin: 0;
  padding: 0;
  display: flex;
}

nav li {
  flex: 5    ;
}

/* End of Header & Nav */

#main-banner{
  background: url(image/banner.jpg) center no-repeat;
  background-size: cover;
  width: 100%;
  height: calc(100vh - 204px);
}


#mealInspiration img {
  max-width: 100%;
  margin: auto;
  border-radius: 10% 10% 0 0;
  border-bottom: #Ffb1a0 10px solid;
}

```

```

    }

#intro{
    display: flex;
    justify-content:center;
    position: relative;
}

#heroTitle{
    position: absolute;
    color: white;
    font-size: 3.5em;
    text-shadow: 3px 3px 4px #262626;
    margin-top: 200px;
}

#searchBox{
    position: absolute;
    display: flex;
    width: 60%;
    margin-top: 280px;
    height: 50px;
}

#searchBox input{
    width: 100%;
    border-radius: 10px 0px 0px 10px;
    border: white 2px solid;
    border-left: white 20px solid;
}

#searchBox input:focus{
    outline: 1px solid #Ffb1a0;
}

#searchButton{
    border: none;
    padding-left: 25px;
    padding-right: 25px;
    border-radius: 0 10px 10px 0;
    background: linear-gradient(to right, #fffc5e 50%,#Ffb1a0 50%);
    background-size: 200% 100%;
    background-position: right bottom;
    transition: all .2s ease-out;
}

#searchButton:hover{
    background-position: left bottom;
}

```

```

        color: #070063;
        border-left: 3px solid #Ffb1a0;
    }

    /* Start of footer */
    footer{
        height: 50px;
        width: 100%;
        background: #262626;

    }

    #footerContent{
        color: white;
        text-align: left;
        font-size: 12px;
        padding: 6px 0;
    }

    /* Start of CSS for index.html */
    #searchResult{
        text-align: center;
        display: flex;
        justify-content: center;
        flex-wrap: wrap;

    }

    #searchTitle{
        display: none;
        margin: auto;
        line-height: 300%;

    }

    .mealItem{
        max-width: 30%;
        text-align: center;
        padding: 0 0 20px 0;
        margin-bottom: 3em;
        background-color: #ffd3ca;
        border-radius: 3em;
        margin: 1rem 1.5%;
    }

    .sMealImg img{
        max-width: 100%;
        margin: auto;
    }

```



```

    border-radius: 10% 10% 0 0;
    border-bottom: #Ffb1a0 10px solid;
    margin-bottom: 1em;
}
.sMealButton{
    position: relative;
    text-decoration: none;
    margin-top: 1em;
    margin-bottom: 0.5em;
    width: 50%;
    background: white;
    display: inline-block;
    padding: 0.5em;
    border-radius: 10em;
    border-bottom: #Ffb1a0 5px solid;
    color: #070063;
}

.sMealButton:hover {
    border-bottom: #ff9b85 5px solid;
    top: 1px;
    background-color: #fffc5e;
    transition: all .2s ease-out;
}

.sMealName>h3{
    font-size: 1.5em;
}

.sMealName>h2{
    font-size: 1em;
    font-weight: normal;
}

.sMealDetails{
    margin: auto;
    position: fixed;
    top: 50%;
    width: 80%;
    height: 90%;
    overflow-y: visible;
    left: 50%;
    transform: translate(-50%, -50%);
    padding: 1.5rem 3rem;
    border-radius: 2rem;
    background-color: #fffc5e;
    box-shadow: #ff9b85 3px 3px;
    display: none;
}

.sMealDerailsImg>img{

```

```

        max-width: 20%;
        margin: auto;
        border-radius: 50%;
        border-bottom: #Ffb1a0 5px solid;
        margin-bottom: 0.5em;
    }

    .sMealDetailsName{
        font-size: 1.5em;
    }

    .sMealDetailsCat{
        font-weight: 600;
        color: #535353;
    }

    .sMealDetailsIns{
        line-height: 150%;
        margin-top: 1rem;
    }

    #sCloseDetail{
        background-color: white;
        border: #ff9b85 1px solid;
        padding: 0.2rem;
        border-radius: 0.3rem;
        color: #262626;
        position: absolute;
        right: 3em;
        cursor: pointer;
    }

    #sCloseDetail:hover{
        background-color: #Ffb1a0;
    }

    .sMealDetailsVideo{
        position: relative;
        text-decoration: none;
        margin-top: 1em;
        margin-bottom: 0.5em;
        width: 20%;
        background: white;
        display: inline-block;
        padding: 0.5em;
        border-radius: 10em;
        border-bottom: #Ffb1a0 5px solid;
        color: #070063;
    }

```

```

.sMealDetailsVideo:hover {
  border-bottom: #ff9b85 5px solid;
  top: 1px;
  background-color: #ffd3ca;
  transition: all .2s ease-out;
}

/* start of CSS for inspiration.html */

#insBanner{
  background: url(image/banner.jpg) center no-repeat;
  background-size: cover;
  width: 100%;
  height: 180px;
}

#mealInspiration{
  padding-top: 20px;
}

#mealInspiration h3{
  font-size: 1.5em;
  padding: 8px 0;
}

#mealInspiration {
  display: flex;
  flex-wrap: wrap;
  justify-content: space-between;
}

#mealInspiration>div {
  max-width: 45%;
  text-align: center;
  padding: 0 10 20px 0;
  margin-bottom: 3em;
  background-color: #ffd3ca;
  border-radius: 5em
}

#mealInspiration>div>p{
  margin: 0 25px;
}

#mealInspiration>div>a {
  position: relative;
  text-decoration: none;
  margin-top: 1em;
  margin-bottom: 1em;
}

```

```

        width: 30%;
        background: white;
        display: inline-block;
        padding: 0.5em;
        border-radius: 10em;
        border-bottom: #Ffb1a0 5px solid;
        color: #070063;
    }

    #mealInspiration>div>a:hover {
        border-bottom: #ff9b85 5px solid;
        top: 1px;
        background-color: #fffc5e;
        transition: all .2s ease-out;
    }

    #mealInspiration>div>p {
        line-height: 150%;
    }

    .show{
        display: block;
    }

    /* start of CSS for blog.php */

    #main-banner-s{
        background: url(image/banner.jpg) center no-repeat;
        background-size: cover;
        width: 100%;
        height: 200px;
    }

    #heroTitle-s{
        position: absolute;
        color: white;
        font-size: 4.3em;
        text-shadow: 3px 3px 4px #262626;
        margin-top: 50px;
    }

    #heroTitle-intro{
        position: absolute;
        color: white;
        font-size: 1.2em;
        text-shadow: 3px 3px 4px #262626;
        margin-top: 145px;
    }

```

```

#blog{
    max-width: 700px;
    margin: auto;
    text-align: center;
}

.post {
    border-radius: 10px;
    box-shadow: 1px 1px 10px #535353;
    margin: 40px 0;
}

.blog-title{
    background-color: #fffc5e;
    border-bottom: #Ffb1a0 4px solid;
    font-size: 1.7em;
    border-radius: 10px 10px 0 0;
    padding: 15px 5px;
}

.blog-content{
    padding: 30px 10px;
    font-size: 1.3em;
}

/* start of CSS for contact.html */

#contact{
    padding: 40px 0;
}

#web-intro{
    font-size:1.3em;
    padding-bottom: 30px;
}

#contact th{
    padding: 10px 50px;
}

.address-info, .email-info{
    border-bottom: #7a7a7a 1px solid;
}

#contact-info{

```

```

        font-size:1.3em;
    }

    #contact-info{
        border-spacing: 0;
    }

    /* start of CSS for form.html */

    #submitSection{
        display: block;
        height: calc(100vh - 204px);
        padding-bottom: 10px;
    }

    .generalForm{
        width: 50%;
        margin-bottom: 30px;
        padding: 10px;
        border: #ff9b85 1px solid;
        border-radius: 10px;
    }

    #submitMealName::placeholder{
        color: rgb(55, 55, 55);
        font-family: 'Akshar', sans-serif ;
        font-size: 1.15em;
    }

    #submitMealInstruct::placeholder{
        color: rgb(55, 55, 55);
        font-family: 'Akshar', sans-serif ;
        font-size: 1.15em;
    }

    #submitMealVideo::placeholder{
        color: rgb(55, 55, 55);
        font-family: 'Akshar', sans-serif ;
        font-size: 1.15em;
    }

    #submitButton{
        border-top: none;
        border-right: none;
        border-left: none;

```

```

}

#submitted{
    height: calc(100vh - 204px);
}
#thankYou{
    position: relative;
    top: 20vh;
    font-size: 1.4rem;
}

/* start of CSS for admin.php & adminReg.php */

#admin-login>div, #admin-reg>div {
    margin: 20px 0;
}

#admin-login input, #admin-reg input{
    padding: 5px 10px;
}

#login-btn, #signup-btn {
    background-color: #Ffb1a0;
    padding: 3px 5px;
    border-radius: 5px;
    border: #ff9b85 1px solid;
    cursor: pointer;
}

/* start of CSS for userSubmit.html */

#submittedData{
    border-collapse: collapse;
    margin-bottom: 30px;
}

#submittedData, #submittedData tr,#submittedData th,#submittedData td{
    border: #535353 solid 1px;
}

#submittedData td, #submittedData th{
    padding: 10px;
}

#countryList{
    display: flex;
    flex-wrap: wrap;
    justify-content: center;
}

```

```

.country{
  margin: 30px;
  display: block;
}

.country p{
  font-size: 3.4em;
  line-height: 0.6em;
}

.country h2{
  font-size: 1.7em;
}

```

recommendation.js

```

fetch(`https://www.themealdb.com/api/json/v2/9973533/randomselection.php`)
  .then((response) => {
    if (response.ok) {
      return response.json();
    }
  })
  .then(data => {
    console.log(data);
    randomRecommend(data);
  })

```

```

function randomRecommend(data) {

  const meal1 = data.meals[0];
  const meal2 = data.meals[1];
  const meal3 = data.meals[2];
  const meal4 = data.meals[3];

  const mealDiv1 = document.getElementById("receipe1");
  const mealDiv2 = document.getElementById("receipe2");
  const mealDiv3 = document.getElementById("receipe3");
  const mealDiv4 = document.getElementById("receipe4");

  const mealImg1 = document.createElement("img");
  mealImg1.src = meal1.strMealThumb;
  mealDiv1.appendChild(mealImg1);

  const mealName1 = meal1.strMeal;
  const mealTitle1 = document.createElement("h3");
  mealTitle1.innerHTML = mealName1;
  mealDiv1.appendChild(mealTitle1);
}

```



```

var mealHow1 = meal1.strInstructions;
const mealIns1 = document.createElement("p");
mealIns1.innerHTML = mealHow1;
mealDiv1.appendChild(mealIns1);

const mealVideo1 = meal1.strYoutube;
const mealWatrch1 = document.createElement("a");
mealWatrch1.href = mealVideo1;
mealWatrch1.target = "_black";
mealWatrch1.innerHTML = "Watch the Video";
mealDiv1.appendChild(mealWatrch1);

const mealImg2 = document.createElement("img");
mealImg2.src = meal2.strMealThumb;
mealDiv2.appendChild(mealImg2);

const mealName2 = meal2.strMeal;
const mealTitle2 = document.createElement("h3");
mealTitle2.innerHTML = mealName2;
mealDiv2.appendChild(mealTitle2);

var mealHow2 = meal2.strInstructions;
const mealIns2 = document.createElement("p");
mealIns2.innerHTML = mealHow2;
mealDiv2.appendChild(mealIns2);

const mealVideo2 = meal2.strYoutube;
const mealWatrch2 = document.createElement("a");
mealWatrch2.href = mealVideo2;
mealWatrch2.target = "_black";
mealWatrch2.innerHTML = "Watch the Video";
mealDiv2.appendChild(mealWatrch2);

const mealImg3 = document.createElement("img");
mealImg3.src = meal3.strMealThumb;
mealDiv3.appendChild(mealImg3);

const mealName3 = meal3.strMeal;
const mealTitle3 = document.createElement("h3");
mealTitle3.innerHTML = mealName3;
mealDiv3.appendChild(mealTitle3);

var mealHow3 = meal3.strInstructions;
const mealIns3 = document.createElement("p");
mealIns3.innerHTML = mealHow3;
mealDiv3.appendChild(mealIns3);

```

```

const mealVideo3 = meal3.strYoutube;
const mealWatrch3 = document.createElement("a");
mealWatrch3.href = mealVideo3;
mealWatrch3.target = "_black";
mealWatrch3.innerHTML = "Watch the Video";
mealDiv3.appendChild(mealWatrch3);

const mealImg4 = document.createElement("img");
mealImg4.src = meal4.strMealThumb;
mealDiv4.appendChild(mealImg4);

const mealName4 = meal4.strMeal;
const mealTitle4 = document.createElement("h3");
mealTitle4.innerHTML = mealName4;
mealDiv4.appendChild(mealTitle4);

var mealHow4 = meal4.strInstructions;
const mealIns4 = document.createElement("p");
mealIns4.innerHTML = mealHow4;
mealDiv4.appendChild(mealIns4);

const mealVideo4 = meal4.strYoutube;
const mealWatrch4 = document.createElement("a");
mealWatrch4.href = mealVideo4;
mealWatrch4.target = "_black";
mealWatrch4.innerHTML = "Watch the Video";
mealDiv4.appendChild(mealWatrch4);

}

```

script.js

```

const searchButton = document.getElementById("searchButton");
const result = document.getElementById("searchResult");
const mealDetailsPop = document.querySelector(".sMealDetailsContent");
const mealDetailsPopFull = document.querySelector(".sMealDetails");
const closeMealDetails = document.getElementById("sCloseDetail");

searchButton.addEventListener('click', getMeal);
result.addEventListener('click', getDetails);
closeMealDetails.addEventListener('click', () => {
  mealDetailsPopFull.style.display = "none";
});
document.addEventListener('click', function (e) {
  //code adapted from DevDreamz, n.d.
  var parent;
  if (!(parent = e.target.closest('#sMealDetailsContent')))

```

```

//end of adapted code
{
    mealDetailsPopFull.style.display = "none";
}

}))

function getMeal() {
    const searchTitle = document.getElementById("searchTitle");
    searchTitle.style.display = "unset";

    let searchInput = document.getElementById('search-input').value.trim();

    fetch(`https://www.themealdb.com/api/json/v1/1/filter.php?i=${searchInput}`)
        .then(response => response.json())
        .then(mealData => {
            let Addedhtml = "";
            if (mealData.meals) {
                searchTitle.innerHTML = "Search Results:";
                mealData.meals.forEach(meal => {
                    Addedhtml += `
                        <div class = "mealItem" data-id = "${meal.idMeal}">
                            <div class = "sMealImg">
                                <img src = "${meal.strMealThumb}" alt = "Image of
the meal">
                            </div>
                            <div class = "sMealName">
                                <h3>${meal.strMeal}</h3>
                                <a href = "#" class = "sMealButton">Get
Instructions</a>
                            </div>
                        </div>
                    `;
                });
            } else {
                searchTitle.innerHTML = "No recipe found";
            }
            result.innerHTML = Addedhtml;
        });
}

function getDetails(e) {
    e.preventDefault();
    if (e.target.classList.contains("sMealButton")) {
        let item = e.target.parentElement.parentElement;
    }
}

```

```

fetch(`https://www.themealdb.com/api/json/v1/1/lookup.php?i=${item.dataset.id}`)
    .then(response => response.json())
    .then(data => mealDetailsDisplay(data.meals));
}
}

function mealDetailsDisplay(meal) {
    meal = meal[0];
    let addedHtml =
        `

36


```

Critique

Evaluate LAMP stack and ASP.NET framework

LAMP stack is a popular open-source stack for delivering web applications (IBM Cloud Education, 2019). It is consisted of Linux, Apache, MySQL and PHP which respectively work as the operative system, web server, database and scripting language.

LAMP stack work well in creating and maintaining a data-driven and dynamic website. In a website running on LAMP, Apache is installed on Linux for processing requests and transmitting data through HTTP (Simic, 2022). While PHP on the one hand can be worked with HTML, on the other hand, it can interact with MySQL to get or insert data for a dynamic website.

One of the benefits of using LAMP stack is that it has a well-developed ecosystem. As the stack which has been used for more than a decade, it has been widely applied by developers. For instance, the well-known content management system WordPress also uses LAMP (MATT WATSON, 2020). The popularity and time-proven usage of LAMP grows a supportive community which allows developers to find the solutions for the issues and bugs of their projects.

In addition, the four components of LAMP are all free and open-source software (FOSS) (Javatpoint, n.d.). It makes building web applications more accessible because anyone can download the tools for free, keeping the LAMP community active. The open-source nature of the LAMP makes the source codes of the four tools available for configuration and customisation, which enhance the flexibility of developing dynamic web applications.

However, there are still disadvantages of using LAMP stack which will be discussed in the following. First, MySQL is not efficient when dealing with large sized data. If the data size is large, MySQL will slow down when trying to run more complexed queries or even fail to run it (Makble, n.d.). This drawback makes developers who need to build a dynamic website loading with large data move to other databases such as MongoDB.

Besides, Apache was viewed as having a lower performance when compared to other web servers such Nginx, especially when there is a hight traffic on the website (Jankov, 2022). In fact, there is a trend that developers choose to other web sever due to its performance issue. According to a web server survey conducted in 2022 which presents the market share of web servers used by global websites, Apache was surpassed by Nginx in 2019 and the gap has been growing since then (Netcraft, 2022). It might indicate that Apache will be phasing out in the future.

Another technology stack this article will discuss is ASP.NET which is an open-source framework developed by Microsoft for building different type of applications. ASP.NET can run on different operative systems including Windows, Linux, macOS, and Docker (Microsoft, n.d.). It is a platform loaded with tools which allow developers to create applications such as dynamic websites. Programming languages including

C#, F# and Visual Basic can be used on ASP.NET which also includes a variety of libraries offering common functionalities (Umbraco, n.d.).

ASP.NET is powerful in building dynamic website in the sense that it can act as a back-end code which combines with HTML to generate dynamic contents. It has a web-page templating syntax called Razor which allows developers to write server-side code with C# that can generate HTML markup (FitzMacken, 2020).

Furthermore, ASP.NET has various tools that can make building a dynamic website easier. One of the useful tools is for the purpose of authentication and authorization. Authentication means to correctly identify the user and authorization means to define the allowed actions for a particular user. The built-in authentication and authorization tool including membership system help developer handle login and manage content access more easily (Gascon, 2022).

Also, ASP.NET is known for requiring less code due to its web development model particularly when developers create a large-scaled website (Widjaya, 2017). For example, it has different validation controls to ensure user submit proper data, thus developers can save programming time used for creating code of inputted data validation.

Despite the advantages of using ASP.NET, it still has some drawbacks. ASP.NET is owned by Microsoft and not community-driven (GraffersID, 2020). This can have the merit of active support from Microsoft. Nevertheless, it also leads to the vendor lock-in problem, making developers difficult to transfer the developed application to another platform. Also, Microsoft's decision on the ASP.NET such as support policy and new release may influence the web applications which have already been created.

Compared to other open-source stacks, ASP.NET may be more costly. Developing a website needs a text editor or an integrated development environment (IDE), whilst dynamic content requires a database as the content source. Though developers can use other free software for replacement, ASP.NET works better with SQL Server database and Visual Studio IDE which are also developed by Microsoft. Those tools require licence fees for full functionalities and developing commercial projects.

Evaluate the various approaches to securing user credentials

Password storage: Hash and salt

A straightforward method to store user credentials is creating a database table which contains usernames mapped with passwords. The way of storing password in the database determines the security strength. Instead of stored in plain text, the password should be hashed first before sending to the database. It is a one-way function to transfer a password into a set of scrambled value.

However, passwords which are only hashed are not perfectly secure. If attackers obtain a list of hashed passwords from the database, they can carry out a brute-force attack in which attacks create random hashed passwords to match the existed

hashed passwords on the database (Gibbs, 2016). Also, if the user passwords are the same, the hashed passwords will also be the same, making them easier to be cracked. Therefore, before going through the hashing process, the passwords also need to be added with a unique salt (a random data) to enhance the password security. In this way, even the passwords of several users are the same, the final hashed and salted passwords will be different.

SQL injection

SQL injection is a website attacking technique to let attackers input code to a malicious SQL statement (W3Schools, n.d.-a). It enables running SQL queries to interact with the database. If the SQL injection is conducted in user login page, it may disclose user credentials in the user table. There are several ways to prevent SQL injection which will be discussed as follows.

Prevent SQL injection: Real escape string

A string escaping mechanism can stop some of the SQL injection attacks. For example, in PHP, function “mysql_real_escape_string()” can escape special character in a string (W3Schools, n.d.-b). When the function is applied on the user inputted value, if attackers want to conduct the SQL injection by inputting “ ' OR 1=1 -- ”, the value inserted into the SQL statement will become “ '/' OR 1=1 -- ” which successfully stop the SQL injection attack. But, it is worth notice that it is not a perfect way for preventing SQL injection, because attackers can input value that with no quote.

Prevent SQL injection: Parameter binding

One of the common and effective methods to prevent SQL injection is parameter binding. It can separate the value from the expression string by creating parameterised queries. The data will be passed into the statement after the SQL code is defined in the first place (OWASP Cheat Sheet Series, n.d.), thus preventing SQL injection attacks.

Prevent SQL injection: Database permission management

Another way to prevent SQL injection is to limited database permissions and segregate users based on the required functionalities on the webpage. If the webpage only needs to read the database, on the webpage the database should be connected with a user who is only granted with the read permission.

Validation of password complexity

The lengthier and more complexed a password is, the more difficult it can be cracked. When developers build a user registration form, they should add a validation function to the password field to ensure inputted passwords have a proper level of complexity. Conditions can include minimum password length, contain of special characters, etc. Besides, common passwords and a password which is identical to its correlative username should also be denied via password validation function. Furthermore, the inputted password can also be compared with database collecting passwords leaked by data breaches, such as [Have I Been Pwned](#). However, it is also important for developers to make a balance when setting conditions for password complexity. A cyber security expert suggested that requiring

an overly complex password will in turn lead to security problems, such as improperly reusing passwords and writing down passwords, etc (Steinberg, 2017).

Login failed error messages

A minor approach that is easy to overlook for preventing hackers from stealing user credentials is the error message displayed on the login page, when users fail to log in. For instance, instead of showing “Incorrect password” which implies that the username is correct, the error message should show “Incorrect username / password” in order not to give malicious attacker information.

Regular vulnerability analysis on system

If the system storing user credentials has vulnerabilities, attackers may be exploit them and access to user data. For example, one of the security weaknesses may be the lack of secure configuration of MySQL. By default the password of the MySQL root user is empty (Web Hosting Geeks, n.d.), which make it easy for attackers to access the database. Vulnerabilities like it on the system need to be identified to protect user data. It might be costly to seek cybersecurity consultant to conduct a penetration testing or vulnerability scanning. Nevertheless, there are weakness scanning tools freely available online such as OpenVAS and Nessus. Those tools can spot out the security vulnerabilities of the system and suggest countermeasures. When the system security is improved, the user information can be stored in a safer way.

References

- DevDreamz (n.d.). Close this popup by clicking away from it. Available at: <https://devdreamz.com/question/139808-close-this-popup-by-clicking-away-from-it> (Accessed: 1 May 2022).
- FitzMacken, T. (2020). Introduction to ASP.NET Web Programming Using the Razor Syntax (C#). Available at: <https://docs.microsoft.com/en-us/aspnet/web-pages/overview/getting-started/introducing-razor-syntax-c> (Accessed: 23 May 2022).
- IBM Cloud Education (2019). LAMP Stack. Available at: <https://www.ibm.com/uk-en/cloud/learn/lamp-stack-explained> (Accessed: 23 May 2022).
- Gascon, E. (2022). Adding Authentication and Authorisation to ASP.NET Core Web Applications. Available at: <https://endjin.com/blog/2022/03/adding-authentication-and-authorisation-to-aspnet-core-web-applications> (Accessed: 23 May 2022).
- Gibbs, S. (2016). Passwords and hacking: the jargon of hashing, salting and SHA-2 explained. the Guardian. Available at: <https://www.theguardian.com/technology/2016/dec/15/passwords-hacking-hashing-salting-sha-2> (Accessed: 24 May 2022).
- GraffersID (2020). Here are Advantages and Disadvantages of Using .NET. Available at: <https://graffersid.com/advantages-and-disadvantages-of-using-net/> (Accessed: 24 May 2022)
- Jankov, T. (2022). Nginx vs Apache: Web Server Showdown. Kinsta. Available at: <https://kinsta.com/blog/nginx-vs-apache/> (Accessed: 23 May 2022).
- Javatpoint (n.d.). What is LAMP? JavaPoint. Available at: <https://www.javatpoint.com/what-is-lamp> (Accessed: 23 May 2022).
- Makble (n.d.). The advantages and disadvantages of MySQL. Available at: <http://makble.com/the-advantages-and-disadvantages-of-mysql> (Accessed: 23 May 2022).
- MATT WATSON (2020). What is LAMP Stack?. Available at: <https://stackify.com/what-is-lamp-stack/> (Accessed: 23 May 2022).
- Microsoft (n.d.). What is ASP.NET?. Available at: <https://dotnet.microsoft.com/en-us/learn/aspnet/what-is-aspnet> (Accessed: 23 May 2022).
- Netcraft (2022). April 2022 Web Server Survey. Available at: <https://news.netcraft.com/archives/2022/04/27/april-2022-web-server-survey.html> (Accessed: 23 May 2022).

OWASP Cheat Sheet Series (n.d.). SQL Injection Prevention Cheat Sheet. Available at: https://cheatsheetseries.owasp.org/cheatsheets/SQL_Injection_Prevention_Cheat_Sheet.html (Accessed: 24 May 2022).

Pedamkar, P. (n.d.). ASP.Net Validation Controls. EduCBA. Available at: <https://www.educba.com/asp-dot-net-validation-controls/> (Accessed: 24 May 2022).

Steinberg, J. (2017). Why You Should Ignore Everything You Have Been Told About Passwords. Available at: <https://www.inc.com/joseph-steinberg/why-you-should-ignore-everything-you-have-been-told.html> (Accessed: 24 May 2022).

Simic, S. (2022). What is LAMP Stack?. PhoenixNAP. Available at: <https://phoenixnap.com/kb/what-is-a-lamp-stack> (Accessed: 23 May 2022).

Umbraco (n.d.). What is ASP.NET?. Available at: <https://umbraco.com/knowledge-base/aspnet/> (Accessed: 23 May 2022).

W3Schools (n.d.-a). SQL Injection. Available at: https://www.w3schools.com/sql/sql_injection.asp (Accessed: 24 May 2022).

W3Schools (n.d.-b). PHP mysqli real_escape_string() Function. Available at: https://www.w3schools.com/php/func_mysqli_real_escape_string.asp (Accessed: 24 May 2022).

Web Hosting Geeks (n.d.). How to Secure MySQL Database Server. Available at: <https://webhostinggeeks.com/howto/how-to-secure-mysql-database-server/> (Accessed: 24 May 2022).

Widjaya, I. (2017). The 10 Biggest Advantages of ASP.NET. Noobpreneur.com. Available at: <https://www.noobpreneur.com/2017/02/28/the-10-biggest-advantages-of-asp-net/> (Accessed: 24 May 2022).