



UNIVERSITY
OF WOLLONGONG
AUSTRALIA

ISIT

**School of Computing & Information Technology Faculty of Engineering and
Information Sciences**

ISIT307 Web Server Programming

Group Name	
	Hellopanda/T2FG14
Assignment 1	
Name:	UOW ID:
Janeeta Parveen D/O Haja Mydeen	7765290
Punyisa Rachatachotikul	8221601

Table Of Contents

Table Of Contents	2
Overview	3
Brief Description of Each File	3
1. Index.php	3
2. Menu.php	6
3. functions.php	10
4. Quiz.php	13
5. results.php	18
6. Leaderboard.php	22
7. Exit.php	23
User Manual	24

My Name: Janeeta Parveen D/O Haja Mydeen

My Student Number: 7765290

My email address: jpdoe@uowmail.edu.au

Assignment Number: Assignment #1

My Name: Punyisa Rachatachotikul

My Student Number: 8221601

My email address: pr486@uowmail.edu.au

Assignment Number: Assignment #1

Overview

This is a report on our website for the game “Challenge Fun”. We have developed an interactive website, utilizing web forms, manipulating strings and arrays, handling user input, and using text files for storing questions, answers, and player information.

We have developed 7 PHP files which are index.php, menu.php, functions.php, quiz.php, results.php, leaderboard.php, and exit.php. Furthermore, we have used two text files to store the data needed for the website. We have used questions.txt, which contains the quiz's questions, answers, and images, and leaderboard.txt to store the player's nickname and scores.

Following this introduction, we will dive into a brief description of each PHP file used in the site's development and a user manual showcasing the screenshots of all the interfaces.

Brief Description of Each File

1. Index.php

This file contains the functionality of the players entering their chosen nickname and beginning the game.

The primary components and their functionalities are as follows:

- **PHP Code**

```
index.php
1  <?php
2  session_start();
3  if ($_SERVER['REQUEST_METHOD'] == 'POST') {
4      $_SESSION['nickname'] = $_POST['nickname'];
5      $_SESSION['overall_points'] = 0;
6      header('Location: menu.php');
7      exit();
8  }
9 ?>
10
```

- ‘*session_start()*’
 - Initiating a session that allows to store user data throughout the user’s interaction with the game.
- Handling Form Submission
 - ‘*\$_POST['nickname']*’ = retrieves the nickname from the form input
 - ‘*\$_SESSION['nickname']*’ = stores in this nickname session variable
 - Initializes the overall point of the player to 0 and stores it in ‘*\$_SESSION['overall_points']*’
 - Redirects the user to ‘*menu.php*’ and terminates with ‘*exit()*’

● HTML & CSS Code

```
index.php
11  <html>
12  <head>
13  <title>Challenge FUN</title>
14  <style>
15  .error {color: #FF0000;}
16  .header-box {
17      background-color: white;
18      background: rgba(300, 300, 300, 0.9);
19      padding: 20px; /* Spacing inside the box */
20      text-align: center; /* Center-align text */
21      border-radius: 15px; /* Rounded corners for a softer look */
22      box-shadow: 0px 4px 15px rgba(0, 0, 0, 0.1); /* Subtle shadow for depth */
23      margin: 0 auto; /* Centered horizontally */
24      width: 100%; /* Adjusts width to fit content */
25      position: absolute; /* Position it absolutely */
26      top: 0; /* Position it at the top of the page */
27      left: 50%; /* Center horizontally */
28      transform: translateX(-50%); /* Center horizontally relative to the left position */
29  }
30
31  h3 {
32      text-align: center;
33      color: black;
34      font: italic bold 70px Georgia, Serif;
35      text-shadow: -4px 3px 0 #ffe0cc, -14px 7px 0 #ffa366;
36      margin: 0px;
37  }
38
39  body {
40      background: url('/A1-T2F-G14-7765290,8221601(Janeeta,Punyisa)/bg.jpg') no-repeat center center fixed;
41      background-size: cover;
42      text-align: center;
43      font-family: verdana;
44      font-size: 20px;
45      padding: 0;
46      height: 100vh;
47      display: flex;
```

```
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
```

```
    flex-direction: column;
    justify-content: center;
    align-items: center;
}

.form-container {
    background: rgba(255, 255, 255, 0.85);
    padding: 20px;
    border-radius: 10px;
    box-shadow: 0px 0px 10px #D3D3D3;
    text-align: left;
    width: 30%;
}

label {
    font-weight: bold;
    color: #e67300;
    display: block;
    margin-bottom: 20px;
}

input[type="text"] {
    margin-bottom: 5px;
    padding: 5px;
    width: calc(100% - 10px);
    font-size: 20px;
    border: 2px solid #0066cc;
    border-radius: 5px;
}
```

```

80      select {
81          font-size: 20px;
82          padding: 5px;
83          width: 100%;
84          margin-bottom: 20px;
85          border: 2px solid #0066cc;
86          border-radius: 5px;
87          font-weight: bold;
88          color: #0066cc;
89      }
90      .btn-submit {
91          font-size: 16px;
92          font-weight: bold;
93          padding: 10px;
94          color: #ffffff;
95          border: 3px solid #0066cc;
96          cursor: pointer;
97          background-color: #e67300;
98          border-radius: 5px;
99      }
100
101  </style>
102 </head>
103 <body>
104     <div class="header-box">
105         <h3>ChallengeFUN</h3>
106     </div>
107     <form method="post">
108         <label for="nickname">Enter your nickname:</label>
109         <input type="text" id="nickname" name="nickname" required>
110         <input type="submit" class="btn-submit" value="Let's Begin">
111     </form>
112 </body>
113 </html>

```

- Creates web page for the “Challenge Fun” game.
- Styled header and a form for the users to enter their nickname.
 - Used required fields to ensure users submit their nickname before starting the game.

2. Menu.php

This file contains the functionality of a menu page where it displays a welcome message that includes the user’s nickname and the user can navigate to “Movies” or “Sports” quiz, “Leaderboard” and “exit” the game.

The primary components and their functionalities are as follows:

- **PHP Code**

```
menu.php
1  <?php
2  session_start();
3  if (!isset($_SESSION['nickname'])) {
4      header('Location: index.php');
5      exit();
6  }
7  ?>
8  <html>
```

- ‘`session_start()`’ = starts the session
- ‘`!isset($_SESSION['nickname'])`’ = if the nickname is not set, it will redirect the user to ‘index.php’

- **HTML & CSS Code**

```
menu.php
8  <html>
9  <head>
10 <title>Challenge FUN - Menu</title>
11 <style>
12 body {
13   font-family: Arial, sans-serif;
14   background-color: #f8f9fa;
15   margin: 0;
16   padding: 0;
17   background-color: #f9e8e1;
18 }
19 .navbar {
20   background-color: rgba(255, 255, 255, 0.9);
21   padding: 20px;
22   box-shadow: 0px 4px 15px rgba(0, 0, 0, 0.1);
23   margin: 0 auto;
24   width: 100%;
25   position: fixed;
26   top: 0;
27   left: 0;
28   display: flex;
29   justify-content: space-between;
30   align-items: center;
31 }
32 .navbar-brand {
33   font-size: 24px;
34   font-weight: bold;
35   color: #333;
36 }
37 .container {
38   background-color: #FFFFFF;
39   padding: 20px;
40   border-radius: 10px;
41   box-shadow: 0px 0px 10px #D3D3D3;
42   max-width: 300px;
43   margin: 100px auto 20px auto; /* Adjusted for fixed navbar */
44 }
```

```
45     align-items: center;
46     text-align: center;
47     margin-top: 10%;
48   }
49   li{
50     font-size: 20px;
51     font-weight: bold;
52     color: #0099ff;
53   }
54   .btn-exit {
55     background-color: #dc3545;
56     color: #ffffff;
57     font-size: 16px;
58     font-weight: bold;
59     border: 3px solid #0066cc;
60     padding: 10px 20px;
61     border-radius: 5px;
62     cursor: pointer;
63     position: absolute;
64     top: 30px;
65     right: 50px;
66   }
67   .btn-submit {
68     font-size: 16px;
69     font-weight: bold;
70     padding: 10px;
71     color: #ffffff;
72     border: 3px solid #0066cc;
73     cursor: pointer;
74     background-color: #e67300;
75     border-radius: 5px;
76     margin-top: 10px;
77   }
```

```

78     margin-left: 10px;
79 }
80 .question-container {
81     margin-bottom: 20px;
82 }
83 </style>
84 </head>
85 <body>
86 <nav class="navbar">
87     <a class="navbar-brand" href="menu.php">Menu</a>
88     <div class="navbar-buttons">
89         <form action="exit.php" method="post" style="display: inline;">
90             <button type="submit" class="btn-exit">Exit</button>
91         </form>
92         <form action="menu.php" method="post" style="display: inline;">
93             <button type="submit" class="btn-submit" style="right: 110px; position: relative;">Menu</button>
94         </form>
95     </div>
96 </nav>
97 <div class="container">
98     <h1>Welcome, <?php echo htmlspecialchars($_SESSION['nickname']); ?></h1>
99     <ul>
100         <li><a href="quiz.php?topic=movies">Movies</a></li>
101         <li><a href="quiz.php?topic=sport">Sport</a></li>
102         <li><a href="leaderboard.php">Leaderboard</a></li>
103     </ul>
104 </div>
105 </body>
106 </html>

```

- The CSS style defines the layout, color, and fonts.
- A navigation bar is included at the top of the page with buttons, “Menu” and “Exit.”
 - The “Menu” button will redirect the user to the menu page (“[menu.php](#)”)
 - The “Exit” button will redirect the user to the index page (“[index.php](#)”)

3. functions.php

This file contains the functionalities of “`getQuestions($topic)`”, “`getLeaderboard()`” and “`saveScore()`” which helps in managing quiz questions and the leaderboard.

- **PHP Code**

```
functions.php
1  <?php
2
3  function getQuestions($topic) {
4      $file = 'questions.txt';
5      $questions = [];
6
7      // Check if the questions file exists
8      if (!file_exists($file)) {
9          throw new Exception("Questions file not found.");
10     }
11
12     $lines = file($file, FILE_IGNORE_NEW_LINES | FILE_SKIP_EMPTY_LINES);
13
14     foreach ($lines as $line) {
15         list($question_topic, $question, $answer, $image) = explode('|', $line);
16         if ($question_topic === $topic) {
17             $questions[] = [
18                 'question' => $question,
19                 'answer' => $answer,
20                 'image' => $image
21             ];
22         }
23     }
24
25     // Ensure at least 4 questions are available for the quiz
26     if (count($questions) < 4) {
27         throw new Exception("Not enough questions available for this topic.");
28     }
29
30     shuffle($questions);
31     return array_slice($questions, 0, 4);
32 }
33
34 function getLeaderboard() {
35     $file = 'leaderboard.txt';
36     $leaderboard = [];
```

```

38     // Check if the file exists
39     if (file_exists($file)) {
40         $lines = file($file, FILE_IGNORE_NEW_LINES | FILE_SKIP_EMPTY_LINES);
41         foreach ($lines as $line) {
42             list($nickname, $score) = explode('|', $line);
43             // Store scores in an associative array
44             $leaderboard[trim($nickname)] = (int)trim($score);
45         }
46     }
47
48     return $leaderboard;
49 }
50
51 function saveScore($nickname, $currentGameScore) {
52     $file = 'leaderboard.txt';
53     $current_scores = getLeaderboard(); // Get the current scores
54
55     // Check if the player already exists in the leaderboard
56     if (isset($current_scores[$nickname])) {
57         // If the player exists, add the current game score to their overall score
58         $current_scores[$nickname] += $currentGameScore; // Update overall score
59     } else {
60         // If the player does not exist, set their score to the current game score
61         $current_scores[$nickname] = $currentGameScore;
62     }
63
64     // Write updated scores back to the file
65     $handle = fopen($file, 'w'); // Open the file for writing (this overwrites existing content)
66     if ($handle) {
67         foreach ($current_scores as $player => $playerScore) {
68             fwrite($handle, $player . '|' . $playerScore . PHP_EOL); // Write each player and their score
69         }
70         fclose($handle); // Close the file
71     } else {
72         throw new Exception("Failed to open leaderboard file for writing."); // Improved error handling
    }

```

- ‘*getQuestions(\$topic)*’
 - Retrieves the quiz question from the file, ‘questions.txt’ based on the chosen topic by the user.
 - It checks if the file exists; if not, it throws an exception.
 - It reads the file line by line, splitting each line into components (topic, question, answer, and image)
 - Filters the questions by the specified topic
 - Shuffles the selected questions and returns a random set of four questions
- ‘*getLeaderboard()*’
 - Reads the ‘leaderboard.txt’ file to retrieve the player nickname and their nickname.
 - Reads the score and stores the information in an associative array, mapping nicknames to their scores.
- ‘*saveScore()*’
 - Saves or updates a player’s score in the ‘leaderboard.txt’ file.
 - Retrieves the current score using ‘*getLeaderboard()*’

- It opens the leaderboard file for writing, overwriting the existing content, and writes the updated scores back into the file.

4. Quiz.php

This file contains the quiz functionalities and their structure.

● PHP Code

```

quiz.php
1  <?php
2  session_start();
3  require 'functions.php';
4
5  if (!isset($_SESSION['nickname'])) {
6      header('Location: index.php');
7      exit();
8  }
9
10 // Initialize overall points if not set
11 if (!isset($_SESSION['overall_points'])) {
12     $_SESSION['overall_points'] = 0;
13 }
14
15 // Check if quiz is already initialized in the session
16 if (!isset($_SESSION['current_quiz'])) {
17     $topic = $_GET['topic'];
18     $questions = getQuestions($topic);
19     $_SESSION['current_quiz'] = $questions; // Store the current quiz in session
20     $_SESSION['correct'] = 0;
21     $_SESSION['incorrect'] = 0;
22 } else {
23     $questions = $_SESSION['current_quiz']; // Retrieve questions from session
24 }
25
26 // Handle form submission
27 if ($_SERVER['REQUEST_METHOD'] == 'POST') {
28     $answers = $_POST['answers'];
29     foreach ($answers as $index => $answer) {
30         // Check the answer against the questions in the session
31         if (trim(strtolower($answer)) == trim(strtolower($questions[$index]['answer']))) {
32             $_SESSION['correct']++;
33         } else {
34             $_SESSION['incorrect']++;
35         }
36     }

```

```

38  // Calculate the current game score
39  $currentGameScore = ($_SESSION['correct'] * 3) - ($_SESSION['incorrect'] * 1);
40
41  // Save the score only if the quiz is completed
42  $nickname = $_SESSION['nickname'];
43  saveScore($nickname, $currentGameScore); // Save current game score to leaderboard
44
45  // Update overall points in session
46  $_SESSION['overall_points'] += $currentGameScore; // Update overall points
47
48  // Clear current quiz session data for next quiz
49  unset($_SESSION['current_quiz']);
50
51  header('Location: result.php'); // Redirect to results page
52  exit();
53 }
54 ?>

```

- Starts a session and includes 'functions.php' for utility functions.
- Initializing the quiz
 - If the current quiz is not initialized in the session, it retrieves the quiz questions passed via the URL (`$_GET['topic']`) and stores them in the session.
- Form submission
 - When the form is submitted, it processes the answers provided by the player.
 - Compares the submitted answers with the correct answers stored in the session, updating the counts for correct and incorrect answers.
 - The current game score is calculated based on the correct and incorrect answers and saved to the leaderboard through 'saveScore' function.
- The user will be redirected to the results page ('`result.php`') when the overall points are updated in the session.

- **HTML & CSS**

```
quiz.php
56  <html>
57  <head>
58      <title>Challenge FUN - Quiz</title>
59      <style>
60          body {
61              font-family: Arial, sans-serif;
62              background-color: #f8f9fa;
63              margin: 0;
64              padding: 0;
65          }
66          .navbar {
67              background-color: rgba(255, 255, 255, 0.9); /* Corrected rgba value */
68              padding: 20px;
69              box-shadow: 0px 4px 15px rgba(0, 0, 0, 0.1);
70              margin: 0 auto;
71              width: 100%;
72              position: fixed;
73              top: 0;
74              left: 0;
75              display: flex;
76              justify-content: space-between;
77              align-items: center;
78          }
79          .navbar-brand {
80              font-size: 24px;
81              font-weight: bold;
82              color: #333;
83          }
84          .container {
85              background-color: #e6f7ff;
86              padding: 20px;
87              border-radius: 10px;
88              box-shadow: 0px 0px 10px #D3D3D3;
89              max-width: 790px;
90              margin: 100px auto 20px auto; /* Adjusted for fixed navbar */
91              align-items: center;
92          }

```

```
95  .card {
96      background-color: #fff;
97      border: 1px solid #ddd;
98      border-radius: 5px;
99      box-shadow: 0 2px 4px rgba(0,0,0,0.1);
100     margin-top: 20px;
101     margin-bottom: 20px;
102     width: auto;
103     display: flex;
104     flex-direction: column;
105
106 }
107 .card-body {
108     padding: 15px;
109 }
110 .btn-exit {
111     background-color: #dc3545;
112     color: #ffffff;
113     font-size: 16px;
114     font-weight: bold;
115     border: 3px solid #0066cc;
116     padding: 10px 20px;
117     border-radius: 5px;
118     cursor: pointer;
119     position: absolute;
120     top: 30px;
121     right: 50px;
122
123 }
124 .btn-submit {
125     font-size: 16px;
126     font-weight: bold;
127     padding: 10px;
128     color: #ffffff;
129     border: 3px solid #0066cc;
```

```

130      cursor: pointer;
131      background-color: #e67300;
132      border-radius: 5px;
133      margin-top: 10px;
134      margin-left: 10px;
135  }
136  </style>
137 </head>
138 <body>
139   <nav class="navbar">
140     <a class="navbar-brand" href="menu.php">Quiz</a>
141     <div class="navbar-buttons">
142       <form action="exit.php" method="post" style="display: inline;">
143         <button type="submit" class="btn-exit">Exit</button>
144       </form>
145       <form action="menu.php" method="post" style="display: inline;">
146         <button type="submit" class="btn-submit" style="right: 110px; position: relative;">Menu</button>
147       </form>
148     </div>
149   </nav>
150   <form method="post">
151     <div class="container">
152       <h1><?php echo ucfirst($topic); ?> Quiz</h1>
153       <?php foreach ($questions as $index => $question): ?>
154         <div class="card">
155           <div class="card-body">
156             <?php if ($topic == 'movies'): ?>
157               <p><?php echo htmlspecialchars($question['question']); ?></p>
158               <label><input type="radio" name="answers[<?php echo $index; ?>]" value="true" required> True</label>
159               <label><input type="radio" name="answers[<?php echo $index; ?>]" value="false" required> False</label>
160             <?php else: ?>
161               
162               <p><?php echo htmlspecialchars($question['question']); ?></p>
163               <input type="text" name="answers[<?php echo $index; ?>]">
164             <?php endif; ?>
165           </div>
166         </div>
167       <?php endforeach; ?>
168       <button class="btn-submit" type="submit">Submit Answers</button>
169     </div>
170   </form>
171 </body>
172 </html>

```

- A navigation bar is included at the top of the page with buttons, “Menu” and “Exit.”
 - The “Menu” button will redirect the user to the menu page (“[menu.php](#)”)
 - The “Exit” button will redirect the user to the index page (“[index.php](#)”)
- Quiz content
 - For the “Movies” topic, it presents a true/false question with radio buttons and requires to be answered.
 - For the “Sports” topic, it displays an image related to the questions and allows the user to input their answer in a text field.
- Form submission buttons
 - For movies, only allows the button to submit the form when all the questions are answered.

- For sports, allows users to submit their answers even if the textfield is empty.

5. results.php

This file consists of the functionalities of calculating and displaying the scores.

- **PHP Code**

```
result.php
1  <?php
2  session_start();
3  require 'functions.php'; // Include your functions file to access leaderboard
4
5  if (!isset($_SESSION['nickname'])) {
6      header('Location: index.php');
7      exit();
8  }
9
10 // Get the player's nickname
11 $nickname = $_SESSION['nickname'];
12
13 // Load overall points from the leaderboard
14 $leaderboard = getLeaderboard(); // Ensure this function reads from leaderboard.txt
15 $overall_points = isset($leaderboard[$nickname]) ? $leaderboard[$nickname] : 0; // Fetch player's score
16
17 $correct = $_SESSION['correct'];
18 $incorrect = $_SESSION['incorrect'];
19 $currentGameScore = ($correct * 3) - ($incorrect * 1); // Calculate current game score
20 ?>
21
```

- Starts a session and includes '*functions.php*' for utility functions.
- Player information
 - Retrieves the player's nickname from the session.
 - '*getLeaderboard()*' function is called to load the leaderboard data, reads from the 'leaderboard.txt' file.
 - Overall points for the player are fetched, defaulting to 0 if not found.

- Score calculation
 - The number of correct and incorrect answers is retrieved from the session.
 - `$currentGameScore` is calculated using the formula: ‘`(correct * 3) - (incorrect * 1)`’

- **HTML & CSS Code**

```

22  <html>
23  <head>
24  |   <title>Challenge FUN - Result</title>
25  </head>
26  <style>
27  |   /* Your existing styles */
28  |   .navbar {
29  |       background-color: white;
30  |       background: rgba(300, 300, 300, 0.9);
31  |       padding: 20px;
32  |       box-shadow: 0px 4px 15px rgba(0, 0, 0, 0.1);
33  |       margin: 0 auto;
34  |       width: 100%;
35  |       position: absolute;
36  |       top: 0;
37  |       left: 50%;
38  |       transform: translateX(-50%);
39  |       display: flex;
40  |       justify-content: space-between;
41  |       align-items: center;
42  |   }
43  |   .navbar-brand {
44  |       font-size: 24px;
45  |       font-weight: bold;
46  |       color: #333;
47  |   }
48  |   .container {
49  |       background-color: #FFFFFF;
50  |       padding: 20px;
51  |       border-radius: 10px;
52  |       box-shadow: 0px 0px 10px #D3D3D3;
53  |       max-width: 600px;
54  |       margin: 200px auto 20px auto;
55  |       text-align: center;
56  |   }
57  |   th, td {
58  |       border: 4px solid #96D4D4; /* Outer border layer with #96D4D4 */

```

```
59         padding: 15px;
60         text-align: center;
61         position: relative;
62         background-color: #fff;
63         box-shadow: 0 0 0 2px #0066cc;
64     }
65     th {
66         font-size: 16px;
67         font-weight: bold;
68         color: #0066cc;
69     }
70     .btn-exit {
71         background-color: #dc3545;
72         color: #ffff;
73         font-size: 16px;
74         font-weight: bold;
75         border: 3px solid #0066cc;
76         padding: 10px 20px;
77         border-radius: 5px;
78         cursor: pointer;
79         position: absolute;
80         top: 40px;
81         right: 90px;
82     }
83     .btn-submit {
84         font-size: 16px;
85         font-weight: bold;
86         padding: 10px;
87         color: #ffff;
88         border: 3px solid #0066cc;
89         cursor: pointer;
90         background-color: #e67300;
91         border-radius: 5px;
92         margin-top: 20px;
93     }
```

```

94     .button-container {
95         display: flex; /* Enable flexbox */
96         justify-content: center; /* Center items horizontally */
97         margin-top: 20px; /* Margin to add space around the container */
98     }
99     .btn-newQ {
100         font-size: 16px;
101         font-weight: bold;
102         padding: 10px;
103         color: white;
104         border: 3px solid #0066cc;
105         border-radius: 5px;
106         cursor: pointer;
107         background-color: #96D4D4;
108     }
109 </style>
110 <body>
111     <nav class="navbar">
112         <a class="navbar-brand" href="menu.php">Quiz Result</a>
113         <div class="navbar-buttons">
114             <form action="exit.php" method="post" style="display: inline;">
115                 <button type="submit" class="btn-exit">Exit</button>
116             </form>
117             <form action="menu.php" method="post" style="display: inline;">
118                 <button type="submit" class="btn-submit">Menu</button>
119             </form>
120         </div>
121     </nav>
122     <div class="container">
123         <table>
124             <tr>
125                 <th>Player</th>
126                 <th>Correct Answers</th>
127                 <th>Incorrect Answers</th>
128                 <th>Points from this quiz</th>
129                 <th>Overall Points</th>
130             </tr>
131             <tr>
132                 <td><?php echo htmlspecialchars($nickname); ?></td>
133                 <td><?php echo $correct; ?></td>
134                 <td><?php echo $incorrect; ?></td>
135                 <td><?php echo $currentGameScore; ?></td>
136                 <td><?php echo $overall_points; ?></td>
137             </tr>
138         </table>
139     </div>
140     <div class="button-container">
141         <form action="menu.php" method="post">
142             <button type="submit" class="btn-newQ">Back to Menu</button>
143         </form>
144     </div>
145 </body>
146 </html>

```

- A navigation bar is included at the top of the page with buttons, “Menu” and “Exit.”
 - The “Menu” button will redirect the user to the menu page (“*menu.php*”)
 - The “Exit” button will redirect the user to the index page (“*index.php*”)
 - Results Display
 - Player’s nickname
 - Number of correct answers
 - Number of incorrect answers
 - Points scored in the current quiz.
 - Overall points accumulated from the previous quizzes.
6. Leaderboard.php

This file contains the functionality of displaying the scoreboard of all the players and sorting the score with the player’s nickname and by highest score.

- **PHP Code**

```

leaderboard.php
1  <?php
2  session_start();
3  require 'functions.php';
4
5  // Redirect to the index page if the nickname is not set in the session
6  if (!isset($_SESSION['nickname'])) {
7      header('Location: index.php');
8      exit();
9  }
10
11 // Get the leaderboard data
12 $leaderboardData = getLeaderboard();
13
14 // Convert the associative array into an indexed array for easier sorting
15 $leaderboard = [];
16 foreach ($leaderboardData as $nickname => $score) {
17     $leaderboard[] = [
18         'nickname' => $nickname,
19         'score' => $score
20     ];
21 }
22
23 // Search functionality
24 $searchNickname = isset($_POST['search_nickname']) ? trim($_POST['search_nickname']) : '';
25 if ($searchNickname) {
26     $leaderboard = array_filter($leaderboard, function($player) use ($searchNickname) {
27         return stripos($player['nickname'], $searchNickname) !== false; // Case-insensitive search
28     });
29 }
30
31 // Determine the order by parameter
32 $order_by = isset($_GET['order_by']) ? $_GET['order_by'] : 'score';
33 if ($order_by === 'nickname') {
34     // Sort by nickname
35     usort($leaderboard, function($a, $b) {
36         return strcmp($a['nickname'], $b['nickname']);
37     });
38 } else {
39     // Sort by score (default)
40     usort($leaderboard, function($a, $b) {
41         return $b['score'] - $a['score'];
42     });
43 }
44 ?>

```

- The session is started to maintain the user's state.
- Leaderboard Scores retrieval:
 - `'getLeaderboard ()'` function fetches the current leaderboard data from the leaderboard text file, returning it as an associate array.
- Data formatting

- Leaderboard data is converted into an indexed array.
- Search functionality & sorting logic.
 - Search term = it filters the leaderboard to only include the player's nickname
 - Sorting = allows sorting the leaderboard by nickname or score based on '*order_by*'

- **HTML & CSS Code**

```

leaderboard.php
45  <html>
46  <head>
47      <title>Challenge FUN - Leaderboard</title>
48      <style>
49          .navbar {
50              background-color: white;
51              background: rgba(300, 300, 300, 0.9);
52              padding: 20px;
53              box-shadow: 0px 4px 15px rgba(0, 0, 0, 0.1);
54              margin: 0 auto;
55              width: 100%;
56              position: absolute;
57              top: 0;
58              left: 50%;
59              transform: translateX(-50%);
60              display: flex;
61              justify-content: space-between;
62              align-items: center;
63          }
64          .navbar-brand {
65              font-size: 24px;
66              font-weight: bold;
67              color: #333;
68          }
69          .container {
70              background-color: #FFFFFF;
71              padding: 20px;
72              border-radius: 10px;
73              box-shadow: 0px 0px 10px #D3D3D3;
74              max-width: 600px;
75              margin: 20px auto 20px auto;
76              text-align: center;
77          }
78          th, td {
79              border: 4px solid #96D4D4;
80              padding: 15px;
81              text-align: center;

```

```
82     position: relative;
83     background-color: #fff;
84     box-shadow: 0 0 0 2px #0066cc;
85   }
86   th {
87     font-size: 16px;
88     font-weight: bold;
89     color: #0066cc;
90   }
91   .btn-exit {
92     background-color: #dc3545;
93     color: #ffffff;
94     font-size: 16px;
95     font-weight: bold;
96     border: 3px solid #0066cc;
97     padding: 10px 20px;
98     border-radius: 5px;
99     cursor: pointer;
100    position: absolute;
101    top: 40px;
102    right: 90px;
103  }
104  .btn-submit {
105    font-size: 16px;
106    font-weight: bold;
107    padding: 10px;
108    color: #ffffff;
109    border: 3px solid #0066cc;
110    cursor: pointer;
111    background-color: #e67300;
112    border-radius: 5px;
113    margin-top: 20px;
114  }
115  table.center {
```

```
116     margin-left: auto;
117     margin-right: auto;
118 }
119 .btn-search {
120     font-size: 16px;
121     font-weight: bold;
122     padding: 10px;
123     background-color: #007bff;
124     color: white;
125     border: none;
126     border-radius: 5px;
127     cursor: pointer;
128     margin-top: 30px;
129     margin-bottom: 25px;
130 }
131 input[type="text"] {
132     margin-top: 30px;
133     padding: 6px;
134     width: 30%;
135     font-size: 20px;
136     border: 2px solid #0066cc;
137     border-radius: 5px;
138 }
139 }
140 table.center {
141     margin-left: auto;
142     margin-right: auto;
143 }
144 .sort-indicator {
145     font-size: 12px;
146     color: #999;
147 }
148 </style>
149 </head>
```

```

leaderboard.php
150  <body>
151  <nav class="navbar">
152  <a class="navbar-brand" href="menu.php">Leaderboard</a>
153  <div class="navbar-buttons">
154  <form action="exit.php" method="post" style="display: inline;">
155  <button type="submit" class="btn-exit">Exit</button>
156  </form>
157  <form action="menu.php" method="post" style="display: inline;">
158  <button type="submit" class="btn-submit">Menu</button>
159  </form>
160  </div>
161 </nav>
162
163 <div class="container">
164 <h1>Leaderboard</h1>
165 <form method="POST">
166 <input type="text" name="search_nickname" placeholder="Search by nickname" value="<?php echo htmlspecialchars($searchNickname); ?>">
167 <button type="submit" class="btn-search">Search</button>
168 </form>
169 <table class="center">
170 <tr>
171 <th>
172 <a href="?order_by=nickname">Nickname</a>
173 <?php if ($order_by === 'nickname') echo '<span class="sort-indicator">↑</span>'; ?>
174 </th>
175 <th>
176 <a href="?order_by=score">Score</a>
177 <?php if ($order_by === 'score') echo '<span class="sort-indicator">↑</span>'; ?>
178 </th>
179 </tr>
180 <?php foreach ($leaderboard as $player): ?>
181 <tr>
182 <td><?php echo htmlspecialchars($player['nickname']); ?></td>
183 <td><?php echo htmlspecialchars($player['score']); ?></td>
184 </tr>
185 <?php endforeach; ?>
186 </table>
187 <form action="?order_by=score" method="post" style="display: inline;">
188 <button type="submit" class="btn-submit">Sort by Highest Score</button>
189 </form>
190 <form action="menu.php" method="post" style="display: inline;">
191 <button type="submit" class="btn-submit">Back to Menu</button>
192 </form>
193 </div>
194 </body>
195 </html>

```

- A navigation bar is included at the top of the page with buttons, “Menu” and “Exit.”
 - The “Menu” button will redirect the user to the menu page (“*menu.php*”)
 - The “Exit” button will redirect the user to the index page (“*index.php*”)
- Leaderboard display
 - Shows each player’s nickname and score.
 - A search form for users to look for specific players by nickname.
- Action button
 - Sorts the leaderboard by highest score.

7. Exit.php

This file consists of the functionalities of the exit session.

- **PHP Code**

```
exit.php
1  <?php
2  session_start();
3  require 'functions.php';
4
5  // Check if the user is logged in
6  if (!isset($_SESSION['nickname'])) [
7      header('location: index.php');
8      exit();
9  ]
10
11 $nickname = $_SESSION['nickname'];
12
13 // Get the overall points from the leaderboard
14 $leaderboard = getLeaderboard();
15 $overall_points = isset($leaderboard[$nickname]) ? $leaderboard[$nickname] : 0;
16
17 // Reset session variables
18 unset($_SESSION['overall_points']); // Reset overall points
19 unset($_SESSION['current_quiz']); // Reset current quiz data
20 unset($_SESSION['correct']); // Reset correct answer count
21 unset($_SESSION['incorrect']); // Reset incorrect answer count
22 unset($_SESSION['nickname']); // Optionally unset the nickname to log out the user
23
24 session_destroy(); // Destroy the session
25 ?>
```

- Start the session: calling ‘`session_start()`’ to initialize a session or resume the current one.
- Checks if the ‘nickname’ session variable is set using ‘`isset()`’. If it is not set, the player will be redirected to ‘`index.php`’ and will exit.
- if the session is set, it retrieves the values for ‘nickname’,
- Calls function `getLeaderboard()` from `function.php` to get data from leaderboard.
- Set the value of overall points by nickname or default to 0.
- Unset the session: Reset session variables values which are overall points, current quiz point, correct quiz, incorrect quiz, and nickname.
- Destroy the session: The session is destroyed completely.

- **HTML & CSS Code**

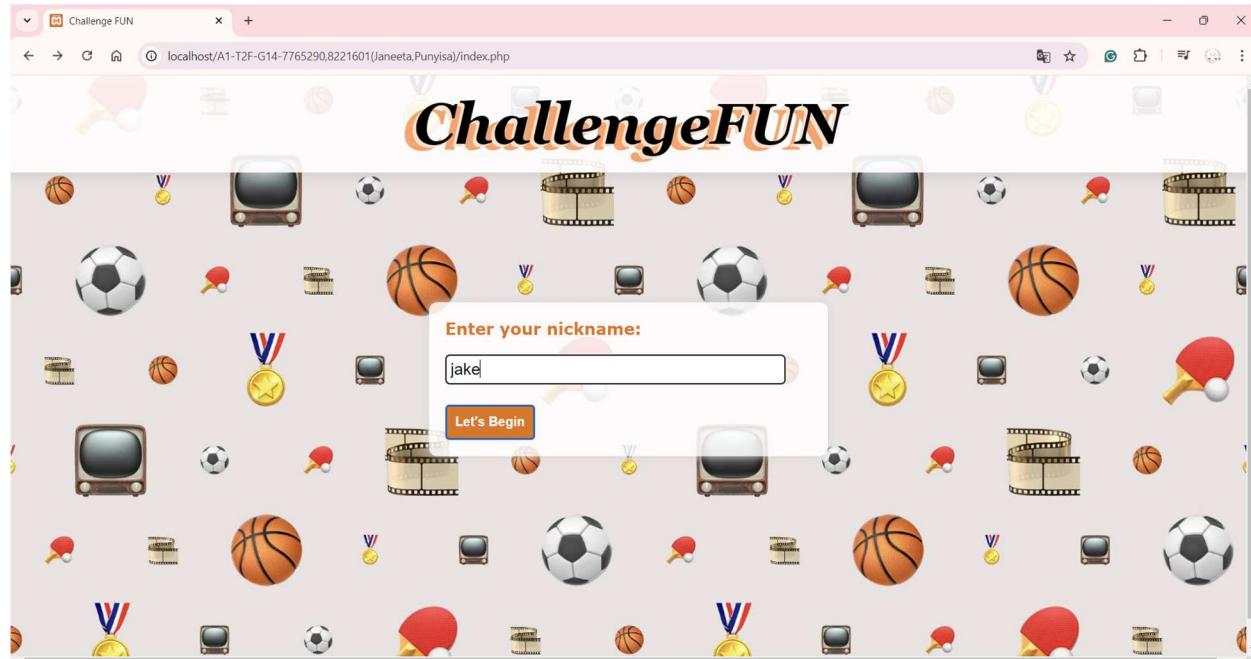
```

exit.php
26
27  <!DOCTYPE html>
28  <html lang="en">
29  <head>
30  |   <meta charset="UTF-8">
31  |   <title>Challenge FUN - Exit</title>
32  </head>
33  <style>
34  body {
35  |   font-family: Arial, sans-serif;
36  |   background-color: #f8f9fa;
37  |   margin-top: 50px;
38  |   text-align: center;
39  }
40
41  .btn-newQ {
42  |   font-size: 16px;
43  |   font-weight: bold;
44  |   padding: 10px;
45  |   color: white;
46  |   border: 3px solid #0066cc;
47  |   border-radius: 5px;
48  |   cursor: pointer;
49  |   background-color: #960404;
50  }
51  </style>
52  </head>
53  <body>
54  |   <h1>Goodbye, <?php echo htmlspecialchars($nickname); ?></h1>
55  |   <p>Your overall points: <?php echo $overall_points; ?></p>
56  |   <form action="menu.php" method="post">
57  |       <button type="submit" class="btn-newQ">Start New Game</button>
58  |   </form>
59  </body>
60  </html>
61

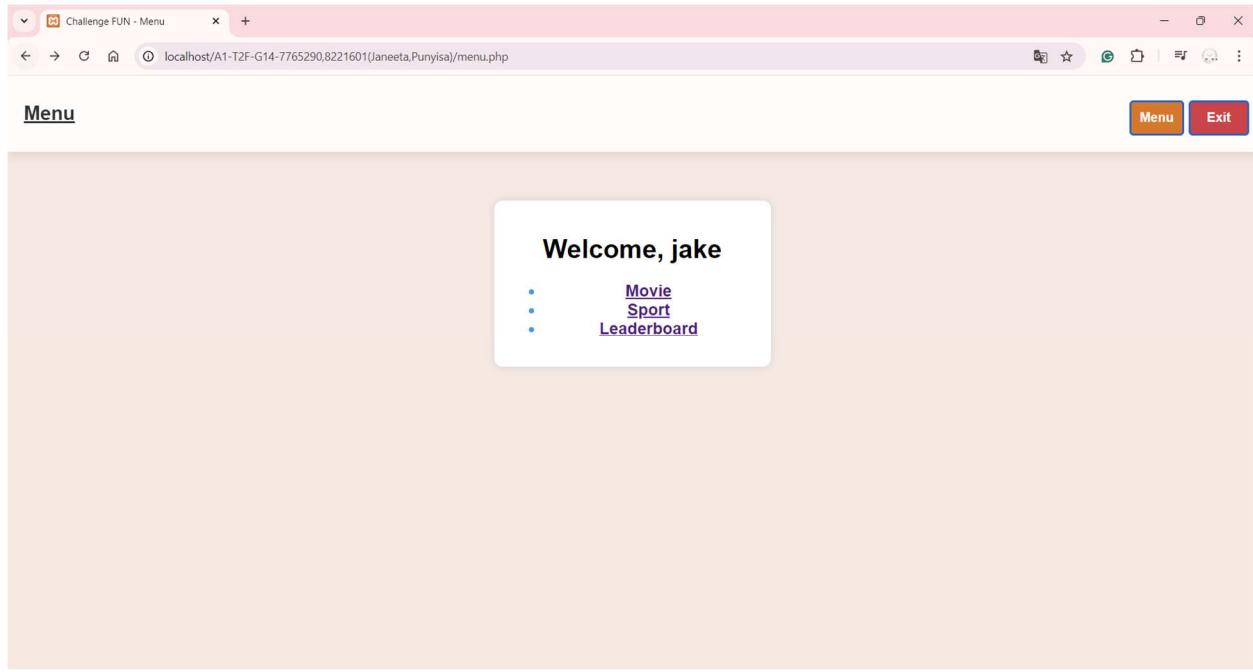
```

User Manual

Step1: Enter username as nickname.

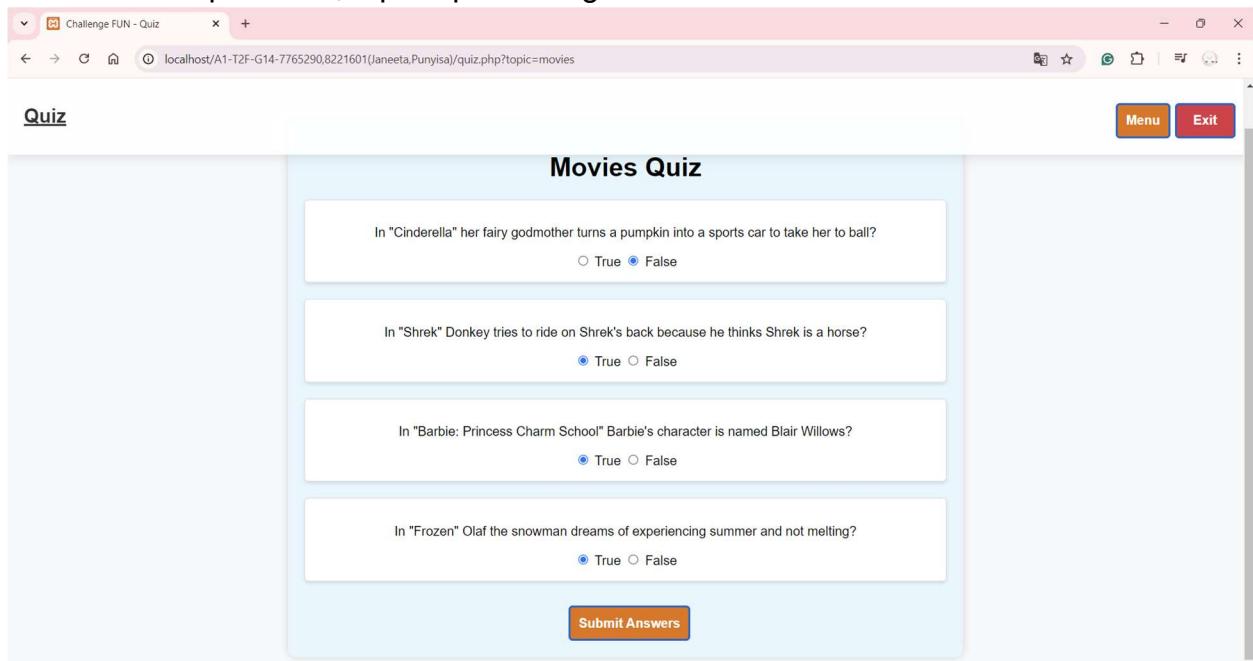


Step2: Menu page will be displayed after pressing the “Let’s Begin” button.



Step3: If the user clicks on Movie, the random 4 movie quizzes will show.

- The user is required to answer the questions.
- For Movies: If the user clicks on the “Submit Answers” button without answering all the questions, a prompt message will be shown.



Or

Step 3: If the user clicks on Sport, the random 4 sports quizzes will show.

Challenge FUN - Quiz

localhost/A1-T2F-G14-7765290.8221601(Janeeta.Punyisa)/quiz.php?topic=sport

Quiz

Sport Quiz



What is the team name?



What is the team name?

Challenge FUN - Quiz

localhost/A1-T2F-G14-7765290.8221601(Janeeta.Punyisa)/quiz.php?topic=sport

Quiz



What is the team name?



What is the team name?

Submit Answers

Step4: After the submitted answer, the result of that quiz will be shown.



Challenge FUN - Result

localhost/A1-T2F-G14-7765290.8221601(Janeeta.Punyisa)/result.php

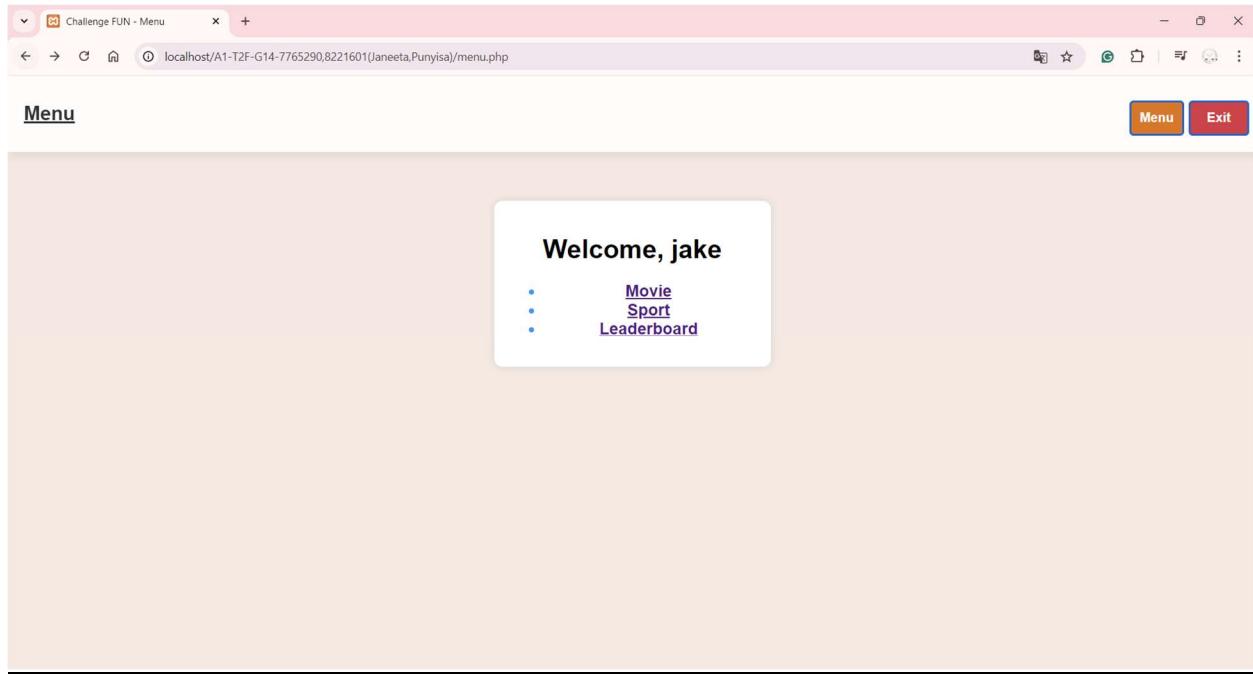
Quiz Result

Exit Menu

Player	Correct Answers	Incorrect Answers	Points from this quiz	Overall Points
jake	4	0	12	12

[Back to Menu](#)

Step 5: If a user clicks on the “Back to Menu” or “Menu” button, it will go back to the menu page and the user can unlimitedly play a quiz on a different topic or the same topic.



Challenge FUN - Menu

localhost/A1-T2F-G14-7765290.8221601(Janeeta.Punyisa)/menu.php

Menu

Welcome, jake

- [Movie](#)
- [Sport](#)
- [Leaderboard](#)

Menu Exit

Step6: The result will be shown after each quiz is submitted, each score in each round will be continuously counted and store in Overall Points

Quiz Result

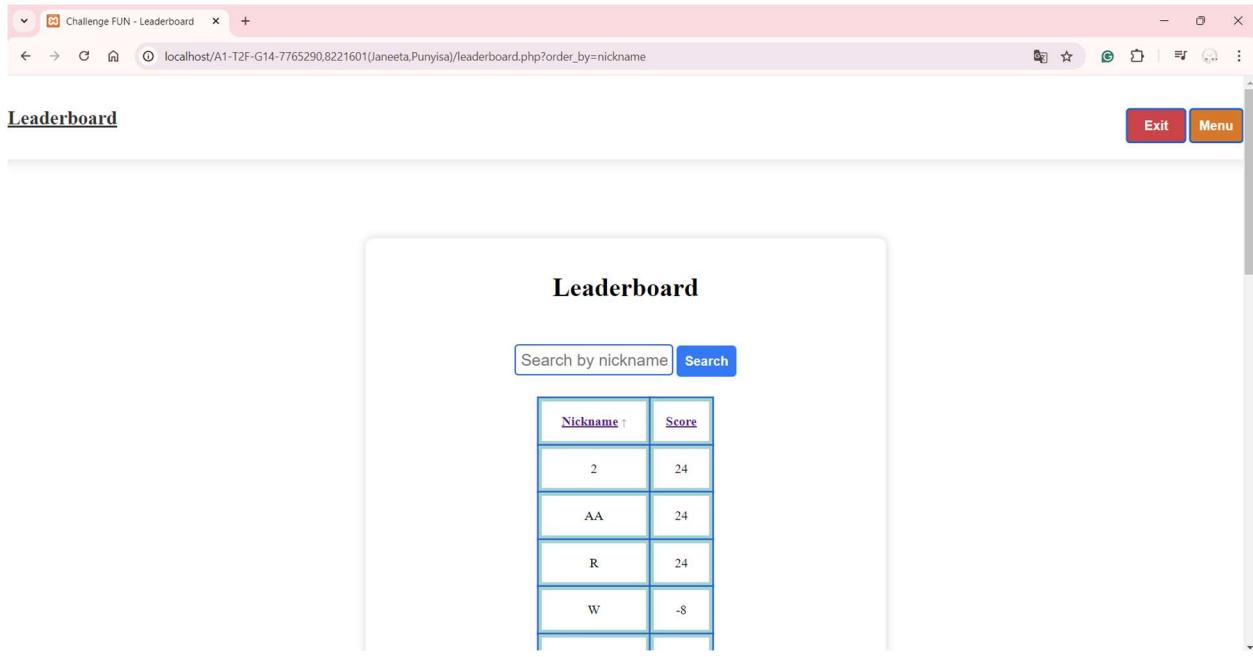
Exit Menu

Player	Correct Answers	Incorrect Answers	Points from this quiz	Overall Points
jake	2	2	4	24

[Back to Menu](#)

Step 7: If the user clicks on the leaderboard on the menu page, all of the user's nicknames and scores recorded will be shown.

- The user can input a nickname in the text field and the results will be sorted by the user's nickname.

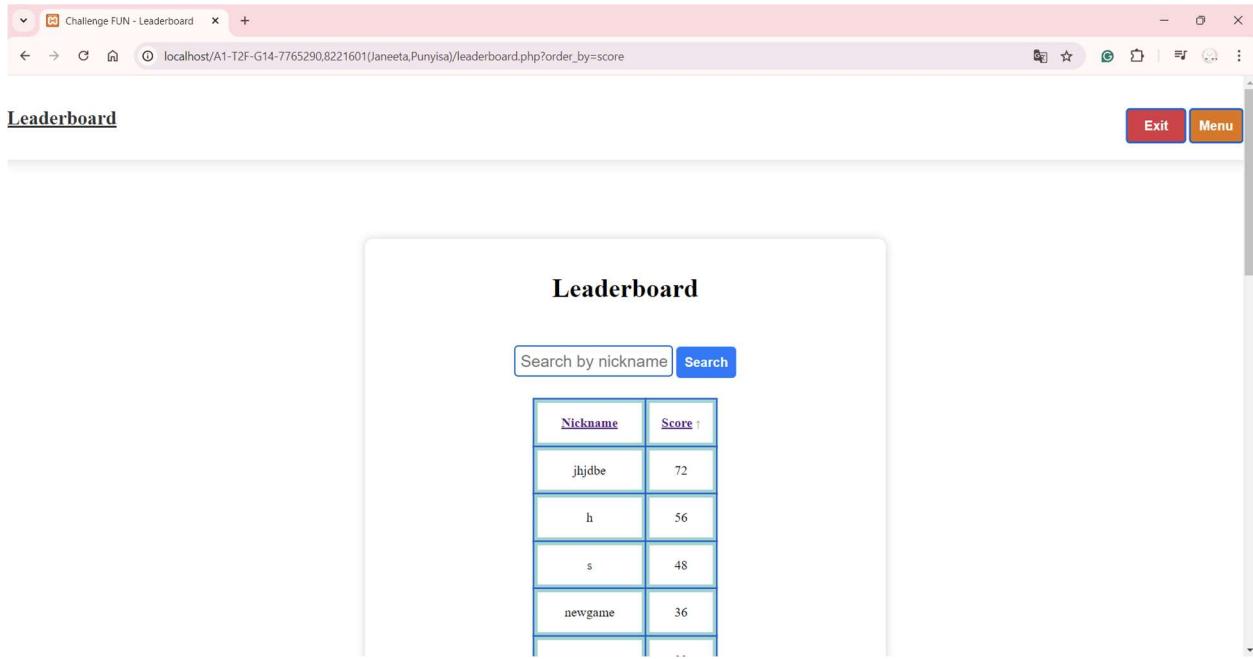


Leaderboard

Search by nickname

Nickname ↑	Score
2	24
AA	24
R	24
W	-8

- If click Score or "Sort by Highest Score" button, the result will be sorted by the user's highest score



Leaderboard

Search by nickname

Nickname	Score ↑
jhjdbe	72
h	56
s	48
newgame	36

a	v
gh	0
ss	0
e	-4
q	-4
p	-4
n	-8
tr	-8
c	-8
W	-8
o	-12

[Sort by Highest Score](#) [Back to Menu](#)

Or

- Users can also search nicknames by input.

[Leaderboard](#) [Exit](#) [Menu](#)

Leaderboard

Nickname	Score ↑
jake	24

[Sort by Highest Score](#) [Back to Menu](#)

Step 8: If the user clicks on the exit button, the nickname and overall points of that user will be shown.

