ECS417U Project Phase 2 Documentation

Name:	ABDIRAHMAN HUSSEIN AHMED
Student Number:	220644231

Assessment Criteria 7
Each time a new entry is added, the most recent post appears on top, followed by the next most recent post using a PHP-driven technique (writing a sorting algorithm) instead of a SQL Query.

PROMPT FOR DEVELOPING A SORTING ALGORITHM

Develop a PHP sorting algorithm that organizes blog entries based on their publication time and date i.e. most recent post appears at the top. Assume that the date is stored in the database in Y/m/d format and time in h:i:sa format. The blog entries are retrieved from the database as an array. If the time and date is invalid, do not sort that specific blog post.

I have provided code that is used to display blog entries once they have been posted where $data['time'] and $data['date'] are used to store the time and date precisely.
File name: 'viewBlog.php'

```
   echo '<div class="post-add">';         echo '<h2 class="title">' .
$data['title'] . '</h2>';      echo '<i class="fa-solid fa-clock"></i><span
class="date">' . $data['date'] . '</span>'; // Display date alongside the
clock icon
   echo'                              <p
class="time">||'.$data['time'].
'</p>';     echo '<p class="post-
meta">By ' . $data['author'] .
'</p>';
   echo '<p class="content">' . $data['content'] . '</p>';      echo
'<hr class="divider">';
   echo '</div>';
```

RESPONSE

```
function sortByDateTime($blogEntries) {
   // Define an empty array to store sorted blog entries
$sortedEntries = array();

   // Loop through each blog
entry     foreach ($blogEntries as
$entry) {
       // Check if the date and time are valid
       $dateTime = $entry['date'] . ' ' .
$entry['time'];       if (strtotime($dateTime)
!== false) {
          // If valid, store the entry with its timestamp as the key
          $timestamp = strtotime($dateTime);
          $sortedEntries[$timestamp] = $entry;
       }
   }

   // Sort the entries based on their timestamps (most recent first)
krsort($sortedEntries);

   // Return the sorted blog
entries     return $sortedEntries;
```

```
}

//     Example     usage:
$blogEntries = array(
    array('title' => 'Title 1', 'date' => '2024/04/10', 'time' => '10:30:00am', 'author' =>
'Author 1', 'content' =>
'Content 1'),     array('title' => 'Title 2', 'date' => '2024/04/09', 'time' => '09:45:00pm',
'author' => 'Author 2', 'content' =>
'Content 2'),
    array('title' => 'Title 3', 'date' => '2024/04/11', 'time' => '08:00:00am', 'author' =>
'Author 3', 'content' => 'Content 3'),
);

$sortedEntries = sortByDateTime($blogEntries);

// Display sorted blog entries foreach ($sortedEntries as $entry) {        echo '<div
class="post-add">';
```

IMPLEMENTING THE SORTING ALGORITHM

We can implement a ‘sortbyDateTime()’ function that takes in an array with blog entries. In this function, looping through the blog entries and targeting the date and time allows validation of the date and time and proceeds to store the entry with its time stamp as the key and store it in an empty array. Ksort() can sort the entries based on their timestamps in descending order. The sorted blog entries can then be returned.

```
function sortByDateTime($blogEntries) {
    // Define an empty array to store sorted blog entries     $sortedEntries = array();

    // Loop through each blog entry    foreach ($blogEntries as $entry)
{
        // Check if the date and time are valid
        $dateTime = $entry['date'] . ' ' . $entry['time'];       if (strtotime($dateTime)
!== false) {
            // If valid, store the entry with its timestamp as the key
            $timestamp = strtotime($dateTime);
            $sortedEntries[$timestamp] = $entry;
        }
    }

    //   Sort   the   entries   based   on   their   timestamps   (most   recent   first)
krsort($sortedEntries);

    // Return the sorted blog entries     return $sortedEntries;
```

We can initialize an empty array and call the function sorted entries to sort the blog entries based on the date and time.

```
$sortedEntries = sortByDateTime($blogEntries);
```

Finally, to display the sorted blog entries, we can store all entries in the sortedEntries array, which sorts the entries by date and time.

```
foreach ($sortedEntries as $entry) {      echo
'<div  class="post-add">';           echo  '<h2
class="title">' . $entry['title'] . '</h2>';
    echo '<i class="fa-solid fa-clock"></i><span class="date">' . $entry['date'] . '</span>';
// Display date alongside the clock icon
    echo'  <p  class="time">||'.$entry['time'].  '</p>';
echo '<p class="post-meta">By ' . $entry['author'] .
'</p>';                  echo    '<p    class="content">'    .
$entry['content']    .    '</p>';                   echo    '<hr
class="divider">';      echo '</div>';
}
```

<table>
<tr><td colspan="2"><b>Assessment Criteria 8 - Extra Feature 1</b>
(Note: delete any of the two features below)
• Organise your blog entries into different months. Provide a drop-down menu for the blog viewers to view the entries of different months stored in an entry archive.
• Allow blog viewers to log in and add comments to the entries. You, as the administrator of the blog should be able to delete entries or comments.
• Add a “preview” button in addpost. When this button is clicked, the new entry is previewed, and you can then decide (via a set of navigational links) whether to upload the entry or go back to edit it.</td></tr>
<tr><td>Instruction: use this column to paste the code from the files that have been used to develop this functionality.
<b>Paste the code directly from visual studio code, do not include a screenshot of the code as this document has a Turnitin requirement and this will lead to an automatic mark of zero. See example below.</b></td><td>Instruction: use this column to address the following in your own words:
1. How did you develop this feature? Describe the steps involved.
2. Describe the inner workings of the approach you developed.
Give reference to the code you have written when answering the questions above.
You can use diagrams (e.g. flow charts) or screenshots of your webpage output to illustrate your answer.</td></tr>
</table>

PREVIEW BUTTON

A preview button has been added to the form so that it redirects to the post page when clicked.

When the preview button is clicked, entries into the form are gathered and previewed

File name: ‘addEntry.php’ and ‘addPost.php’

```
    <input class="btn-submit" type="submit" value="Post" name="submit">
    <button type="submit" class="btn-preview" name="preview">Preview</button>
    $title = isset($_POST["title"]) ? $_POST["title"] : '';
    $author = isset($_POST["author"]) ? $_POST["author"] : '';
    $content = isset($_POST["content"]) ? $_POST["content"] : '';
```

The data gathered is then prepared for preview by the script. After storing the data, the user is redirected to the preview page.

File name: ‘addPost.php’

```
    $_SESSION['preview_data'] = array(
        'title' => $title,
        'author' => $author,
        'content' => $content
    );

    header("Location: preview.php");
    exit();
```

On the preview page, the stored data is retrieved from the session and displayed to the user for previewing.The preview page displays the post data in a format that reflects how it will appear once it's posted.

File name: ‘preview.php’

```
</head>
<body>
    <div class="post-add">
        <h2 class="title"><?php echo $previewData['title']; ?></h2>
        <p class="post-meta">By <?php echo $previewData['author']; ?></p>
        <p class="content"><?php echo $previewData['content']; ?></p>
    </div>
    <form method="post" action="addpost.php">
        <button type="button" onclick="window.history.back();">Edit</button>
    </form>
</body>
</html>
```

Assessment Criteria 8 - Extra Feature 2
(Note: delete any of the two features below)

- Organise your blog entries into different months. Provide a drop-down menu for the blog viewers to view the entries of different months stored in an entry archive.

- Allow blog viewers to log in and add comments to the entries. You, as the administrator of the blog should be able to delete entries or comments.

- Add a "preview" button in addpost. When this button is clicked, the new entry is previewed, and you can then decide (via a set of navigational links) whether to upload the entry or go back to edit it.

Instruction: use this column to paste the code from the files that have been used to develop this functionality. **Paste the code directly from visual studio code, do not include a screenshot of the code as this document has a Turnitin requirement and this will lead to an automatic mark of zero. See example below.**	Instruction: use this column to address the following in your own words: 1. How did you develop this feature? Describe the steps involved. 2. Describe the inner workings of the approach you developed. Did you use any Generative AI tools to achieve this? If yes, can you provide the prompts you wrote to achieve this? Give reference to the code you have written when answering the questions above. You can use diagrams (e.g. flow charts) or screenshots of your webpage output to illustrate your answer.

ORGANISING BLOG ENTRIES INTO DIFFERENT MONTHS

CHATGPT:
User:
Add a feature to my blog that allows users to view blog entries by month. Use a drop-down to display the months, and upon the user selecting a month the user should be able to see the entries for the month they have selected. You may assume the blog entries stored in the database are id, name, content, date, and time.

Response:
To add the requested feature, you can use the following code:

```
<!-- In the HTML file -->
<form method="get" action="">
   <label for="month">Select Month:</label>
   <select name="month" id="month" onchange="this.form.submit()">
      <option value="">All Months</option>
      <?php
      // Query to retrieve all distinct months with blog entries
      $months_query = "SELECT DISTINCT DATE_FORMAT(date, '%Y-%m') AS month
FROM blog_entries ORDER BY date DESC";
      $months_result = mysqli_query($conn, $months_query);

      // Displaying all the months in a dropdown list        while ($month =
mysqli_fetch_assoc($months_result)) {            $display_month = date('F Y',
strtotime($month['month']));            echo '<option
value="'.$month['month'].'">'.$display_month.'</option>';

      ?>
   </select>  </form>
```

USING THE AI-GENERATED CODE TO CREATE THE SORTING ALGORITHM

If a user chooses all months we do not need to sort. If the user decides to pick a specific month, then we need to display the entries for that month using the data retrieved from the database. The months were stored in a PHP array and each month was associated with a key, so that we can access data related to those months from the database.

```
<option value="all">All Months</option>
        <?php
        $months = array(
            '01' => 'January',
            '02' => 'February',
            '03' => 'March',
            '04' => 'April',
            '05' => 'May',                '06' => 'June',
            '07' => 'July',
            '08' => 'August',
            '09' => 'September',
            '10' => 'October',
            '11' => 'November',
            '12' => 'December'
        );
```

We generate a list of months from the array and display them as months to allow the user pick a specific month they would want to sort. The values picked by the user can then be used for further processing.

```
  foreach ($months as $monthNum => $monthName) {             echo '<option value="' .
$monthNum . '">' . $monthName .
'</option>';
        }
```

When the user, selects a specific month from the dropdown menu, we check if the month is not equal to all. If the user selects all in this case, then no where clause is added to the query. If the user selects a specific month, then we can use the select to get the entries related to that specific month.

```
if(isset($_GET['month']) && $_GET['month'] != 'all') {
    $month = $_GET['month'];
    $sqlSelect .= " WHERE MONTH(date) = '$month'";
}
```

Assessment Criteria 14 and 15 - Identification of semantic HTML5 elements and importance of semantic markup

What semantic HTML5 elements have been used within your portfolio site?
Why is semantic markup important?
You will need to provide examples related to your project to illustrate your answer. (max. 250 words)

Semantic elements were used to enhance readability and provide structure across all pages. In addition to that, using semantic elements helped specify styles for elements repeated across all pages.

- Semantic elements used included:
- Header – Used <header> to define nav links. Used in all pages.
- Nav bar – Used <nav> within the header to encapsulate navigation links
- Article – Content of the web pages including sections were wrapped in <article> tags.
- Section – Used <section> for grouping related content.
- A-side – Used the <aside> for content that was main content related but separate.
- Field-set – Used <fieldset> to group related form controls together. Used in pages with forms.
- ul – Used <ul> to represent lists without applying order.
- Footer –A <footer> tag was included to represent the footer at the bottom of each page.

Semantic elements play a crucial role in improving web accessibility by creating a clear structure and meaning for content.

Screen readers and other assistive technologies rely on these tags to accurately interpret information. Search engines can better parse and index web content when it is marked up semantically, leading to improved SEO rankings.

Structured content is key to enhancing visibility on search engines. Developers benefit from using semantic HTML as it makes code more readable and standardized. Clear element names help browsers consistently identify and interpret the content.

Web developers can ensure HTML is more interoperable across the internet by incorporating semantic elements. This reduces confusion and promotes consistent rendering on different platforms.

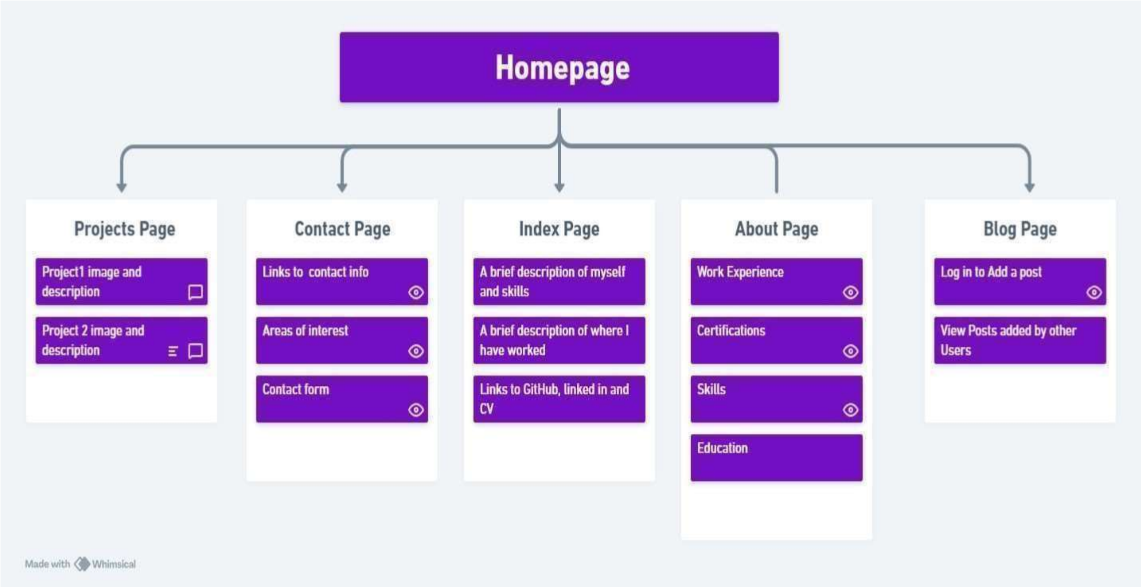
Assessment Criteria 16 - Content organisation
Explain your portfolio site's structure and navigation.
Justify the arrangement of the content and how it enhances user experience.
You should also include a site navigation diagram.

JUSTIFICATION FOR ARRANGEMENT

The website focuses on creating a user-centric flow. The linear model ensures visitors can follow a logical path while navigating the website. Hierarchical models may overwhelm users with too many choices providing a complicated design. The home page, projects, contact, and about me are essential and are prominently featured in a linear flow making it easy for users to get in touch, the arrangement reduces clutter on other pages.

In addition, I have included contact information in the footer (email) to give users other ways of accessing me regardless of which page they are on. This approach enhances accessibility and ensures that users have multiple avenues to reach out, contributing to a more user-friendly design.

PORTFOLIO SITE MAP


Homepage
Projects Page
Project1 image and description
Project 2 image and description
Contact Page
Links to contact info
Areas of interest
Contact form
Index Page
A brief description of myself and skills
A brief description of where I have worked
Links to GitHub, linked in and CV
About Page
Work Experience
Certifications
Skills
Education
Blog Page
Log in to Add a post
View Posts added by other Users
Made with Whimsical

Assessment Criteria 17 and 18 – Design concept and rationale

Describe the website design, including the theme and colour scheme.
What CSS style rules did you write to achieve this?
Did you use any Generative AI tools to achieve this? If yes, can you provide the prompts you wrote to achieve this?
Finally, provide a rationale behind the chosen design and how it reflects your personal brand.

DESIGN CONCEPT

To enhance usability and aesthetics, the contact form was styled with buttons, labels, and input fields. More emphasis was put on styling content that pointed out my areas of interest.

```
.animated-text {    margin-left: 20px;    border-right: 2px solid;    white-space: nowrap;    overflow: hidden;
animation: typing 4s steps(40, end) infinite, blink-caret .75s step-end
infinite, fadeInUp 0.8s ease forwards;
}
```

The sole purpose of 'blink caret' was to mimic the blinking cursor, and typing to imitate typing. This design was used for the text "Harnessing opportunities in tech to drive growth". We intend to catch the user's attention ensuring that they are aware of particular interests before filling in the contact form.

On the projects page, I presented the tech stack I am familiar with using a slideshow animation rather than listing the tech stack, which will take up a huge section of the page, providing a more organized approach. We used keyframes to achieve this.

```

.logos-slide { white-space: nowrap; animation:   slide 5s linear infinite;
}

.logos  {   overflow: hidden;  padding:
60px 0;  background-color:
white;
}
@keyframes slide { from { transform: translateX(0);
} to { transform: translateX(-100%);
}
}
```

The animation name is set to slide and given a running time of 5 seconds. The animation starts from the initial position (0% translation along the x-axis) and smoothly transitions to the left within the specified duration of 5 seconds.

On the About page, the work experience is displayed in a timeline form. This design is usually used to showcase a list sequences of events or other experiences. To implement the design, we set the section to contain multiple containers each representing an experience. Each container includes headings, paragraphs and lists to provide the necessary information.

```
 <div class="container left">
    <div class="content">
      <h3>2024</h3>
      <h4>Axis Tuition Center, London</h4>
      <p>Web Developer</p>
      <p>November 2023 – January 2024</p>
      <ul>
        <li>Effectively implemented responsive web design principles, ensuring the
```

```
platform was fully optimized for an array of devices and diverse screen sizes. This strategic approach
significantly improved user experience and
accessibility across smartphones, tablets, and desktops.</li>
        <li>Seamlessly integrated a suite of interactive features, including quizzes, forums, and progress
trackers, into the platform. This enhancement fostered a more
engaging and interactive learning environment, leading to noticeable improvements in user engagement
and overall learning outcomes.</li>
      </ul>
    </div>
  </div>
```

To style the elements, I defined a padding and positioning style for the containers. I used the pseudo element ':before' to create a vertical line for connecting containers on the timelinel positioned some containers on the left and others on the right aligning the containers properly on the timeline.

```
.container {
  position: relative;
  padding: 20px;
}
.container.left {
  padding-left: 80px;
}
.container.right {
  padding-right: 80px;
}
.container::before {
  content: '';
  position: absolute;
  top: 50%;
  border-width: 2px;
  border-style: solid;
  border-color: white;}
```

For the Blog Page
• Adding a margin of 1.25rem to the main element, enabling the spacing of the main content area • Setting headings to use the font size of 1.5rem, be centered, and margin 1.5 rem with a font-family: "Poppins" with weight being bold • Titles set to the color orange to stand out as essential and catch the eye. Additional text (p.add) with font-size of 1.5rem, color of blue (#007bff), and right-aligned style.

Button links (#addForm) is styled with padding, background-color (#007bff), white color text, border-radius, and text with no line decoration. When hovering, the background color darkens (#0056b3)

```
#addForm {
  padding: 0.625rem 1.25rem; /* Equivalent to 10px 20px */
  background-color: #007bff;
  color: #fff;
  border: none;
  border-radius: 0.3125rem; /* Equivalent to 5px */
  text-decoration: none;
}

#addForm:hover {
  background-color: #0056b3;
}
```

Blog post content (div.post) with a border, border-radius style, and overflow hidden functions spaced neatly.

Images in posts (post img) set with a fixed width and auto height to maintain image aspect ratio as well as a margin for spacing.

Post reviews (div.post-review) styled with padding along with a flex container in the column layout.

The design of the website is a combination of a modern and minimalist style featuring a clear layout and a touch of a professional look . The color choices are various shades of blue, while the color orange is used to create points of interest in the blog section and highlight importance. This style sets the website to be professional and sophisticated while retaining readability and user-friendly features.

Assessment Criteria 19 - Web technologies/framework

For this project, you have made use of HTML, CSS, JavaScript and PHP. If you were to conduct this project again, which other web technologies/frameworks would you use to develop your portfolio site? Justify your choice. (max. 250 words)

REACT.JS

React.js is a widely used JavaScript library designed to create user interfaces. It is based on a component-based architecture, which simplifies the management of complex user interfaces. Therefore, with the help of React.js, I can develop several parts of my portfolio into reusable components, such as a project preview card, a contact form, and a navbar. Additionally, with the virtual DOM, react.js increases performance by only re-rendering what has changed...

Benefits:

Modularity: the component-based architecture will assist me in making my UI components reusable.

Efficiency: the virtual DOM guarantees efficiency by only updating what needs updating and not re-rendering what does not need a render.

Comprehensibly broad community: react.js has a large and open-source community, which will give me access to several libraries.

SASS SYNTACTICALLY AWESOME STYLE SHEETS FOR STYLING

Justification: Vanilla CSS can be repetitive and challenging when working with a collection of styles, especially while keeping the code base organized and maintainable. Sass is a CSS preprocessor that offered a range of new functionality such as variables, nesting, mixins, and other features that drastically improved the styling side of my portfolio.

Benefits:

Variables: define regular values for the project, such as colors and font sizes that would eliminate repetition.

Nesting: defines a parent-child relationship between select bodies and style them in hierarchies .

Mixins: Creates reusable patterns of styles for elements like media queries that can be implemented across screens.

WEB PACK:

I would use web pack to bundle my front end code since it increases efficiency and reduces unnecessary loads.

EXPRESS.JS:

For the server side functionality, express.js would be used with Node.js for building server-side applications. It can easily connect to databases and implement custom logic.

In conclusion, React.js can be used for the front-end and sass to improve styling and webpack for asset delievery to create a more maintainable portfolio.