CSC-20021 Technical Report

By 21010398

Link to Live Website:

Admin Username: Computer

Admin Password: Science

Number of words: 1180

|  |  |
| --- | --- |
| **Tasks** | **Completed** |
| 1: Homepage & UI/UX | Yes |
| 2a: CV | Yes |
| 2b.i: Practical 4 – Task 2 | Yes |
| 2b.ii: Practical 4 – Task 3 | Yes |
| 2c: Login System | Yes |
| 3a: Submit.php | Yes |
| 3b: List.php | Yes |
| Used AJAX in List.php | No |
| 3c: Display.php | No |

In this coursework, you will see many of the skills that were taught throughout the course demonstrated to the best of the student’s ability. The coursework aimed to design a website that was intuitive for new users to navigate, and easily accessible to those who may be visually impaired.

To increase usability for new and returning users, one of the major things implemented was the navigation bar. It features links to every different page on the website, as well as two sets of drop-down links. The first drop-down menu is for the CV, where each item on the menu will take you to a different section of the CV via the use of HTML “<a>” links and “<div>” ids. This ensures that users can jump to a specific section if they so need, rather than having to scroll through the entire page. The other drop-down menu is used for the publication section to have different links for adding a new entry versus viewing all the entries that were already in the database. These links could have been separated on the navbar; however, it is more intuitive to have them in the same section so that the user knows they are linked in some way.

To improve accessibility, a few different things were implemented which included a high-contrast colour scheme and varying font sizes for maximum readability. A high-contrast colour scheme was important to help those who may be colourblind to still see the different parts of the page such as the text boxes or the navigation bar. If the contrast were to be minimal, then they would find it much more difficult to see any difference in the layout of the page and, most importantly, may find it harder to read the text on the page itself. The large text sizes are there to help those who may be partially sighted and find it harder to read small-point fonts.

This project was completed with the use of Visual Studio Code and a mixture of JavaScript, HTML, CSS, and PHP programming. The different elements were all attached via the use of HTML’s “<script>” and “<form>” functions or interaction with the navigation bar to push and pull information from databases and the user. This helped to make the website run as smoothly as possible with very few full reloads.

To make sure that the website had some form of security, a login system was implemented along with an admin account. The purpose of this was to make sure that the database would not be littered with fake reports or have real reports being maliciously deleted or edited without the knowledge or permission of an administrator. To make this possible, a final field in the user database was added to make sure the system could identify administrators versus regular users. If a user tries to access the part of the website where they can add entries to the database, but is not logged in as a registered administrator, they will be taken to the list page instead.

A picture containing diagram

Description automatically generated

Figure 1 – State Diagram

Figure 1 shows a state diagram of the website, including all the different pages the user should be able to access via different links or through prior pages. This helps to show how intuitive the web page is for users to navigate as well as how simple it would be to extend or rearrange the pre-existing website. The CV node is split into three different sub-sections of the page that the user can click to visit on the drop-down menu for ease of access. The weather forecast node is split into two different criteria that the user must choose for the graph to be displayed. It is not required to choose a colour as the website will default to pink if one is not chosen; however, this is not the same for the graph type, which needs to be picked for anything to be displayed. Once the full chart is displayed, users can go back and change the chart type or colour without having to reload the page, this is done to make the website experience as quick and easy as possible. The publication system is split into two different pages that the user can access. If they wish to look at the list of journals, they can do so without having to be logged in but if they wish to add a new entry, the website will force them to log in and make sure they’re an admin before allowing access.

If there were an opportunity to redo this project, it would be beneficial to include a couple of things that were missed in this attempt such as the AJAX implementation and the “Display.php” file functionality. These things were missed from the original attempt as time was tight and other things were needing to be fixed or implemented for optimal functionality – such as admin verification for access to “Submit.php”. Although AJAX would have been helpful to have to keep “List.php” up to date in real-time, it was still found that, as adding a new entry and seeing the list were separate pages, the refresh would update the list without the use of AJAX. To compensate for the lack of display functionality, the list table does display all the information in the database for each entry.

To improve this project, there could be the addition of images to help improve comprehension and increase accessibility for those who may not be able to read text. Images could also be used on the list page to represent the different schools they were published under. On top of that, the ability to either delete or edit existing entries in the database would be useful in case the links to those articles change or are no longer useful. This would prevent confusion on the user’s behalf and them having to go and find the article themselves, which would undermine the entire purpose of the website. Another simple way to improve the website would be to add the functionality to add new users (either admin or not) and to increase the functionality of being logged in, possibly by making the list only available to those who are registered. A logo or some form of branding would also be another easy way to improve the look of the website and add some individuality to it, even if it would just be the Keele University branding. All these improvements would also add the opportunity to introduce different coding principles and attempt new code that has not already been used in this project.

References:

W3C (2018). Web Content Accessibility Guidelines (WCAG) 2.1. [online] W3.org. Available at: https://www.w3.org/TR/WCAG21/.

Baker, K. (2021). The Ultimate Guide to Web Accessibility. [online] blog.hubspot.com. Available at: https://blog.hubspot.com/website/web-accessibility.