Abstract

A web-based tool for managing research outputs by CAIR, a South African research centre was designed, developed, tested, documented and presented to our client Tommie Meyer. The aim of this new system was to introduce new features, improvements and quality-of-life changes not present in the current state of the CAIR website.This project was developed using Javascript, HTML, CSS, PHP, a MYSQL database and incorporated several frameworks and libraries to build a user-friendly client-server based website to allow users to securely request and retrieve data from a centralised database as well as make requests to modify the content stored within it. The system had to allow users to perform CRUD operations corresponding to research outputs; search up entries based on various criteria; and produce reports based on filtering parameters input by the user, among other specifications. The final product was a fully functional system, meeting the requirements laid out by our client.

1.1 Introduction

The purpose of this project was to create a web-based tool for managing the research outputs produced by the members of the Centre for Artificial Intelligence Research (CAIR) - a virtual South African Research Centre, hosted at the Council for Scientific and Industrial Research (CSIR). The centre has nodes at several universities in the country. The goal of this organization is to bring together the peer reviewed research work of scientists, researchers, and students across the country in order to contribute to the advancement of AI research in South Africa. CAIR stores and maintains these research papers on their website (cair.org.za) and they are available to the general public to read and download for free.

The aim of this new system was to introduce new features, improvements and quality-of-life changes not present in the current state of the CAIR website. The additions needed to allow users more control over the information they access and/or contribute to the organization. The system was required to enable users to:

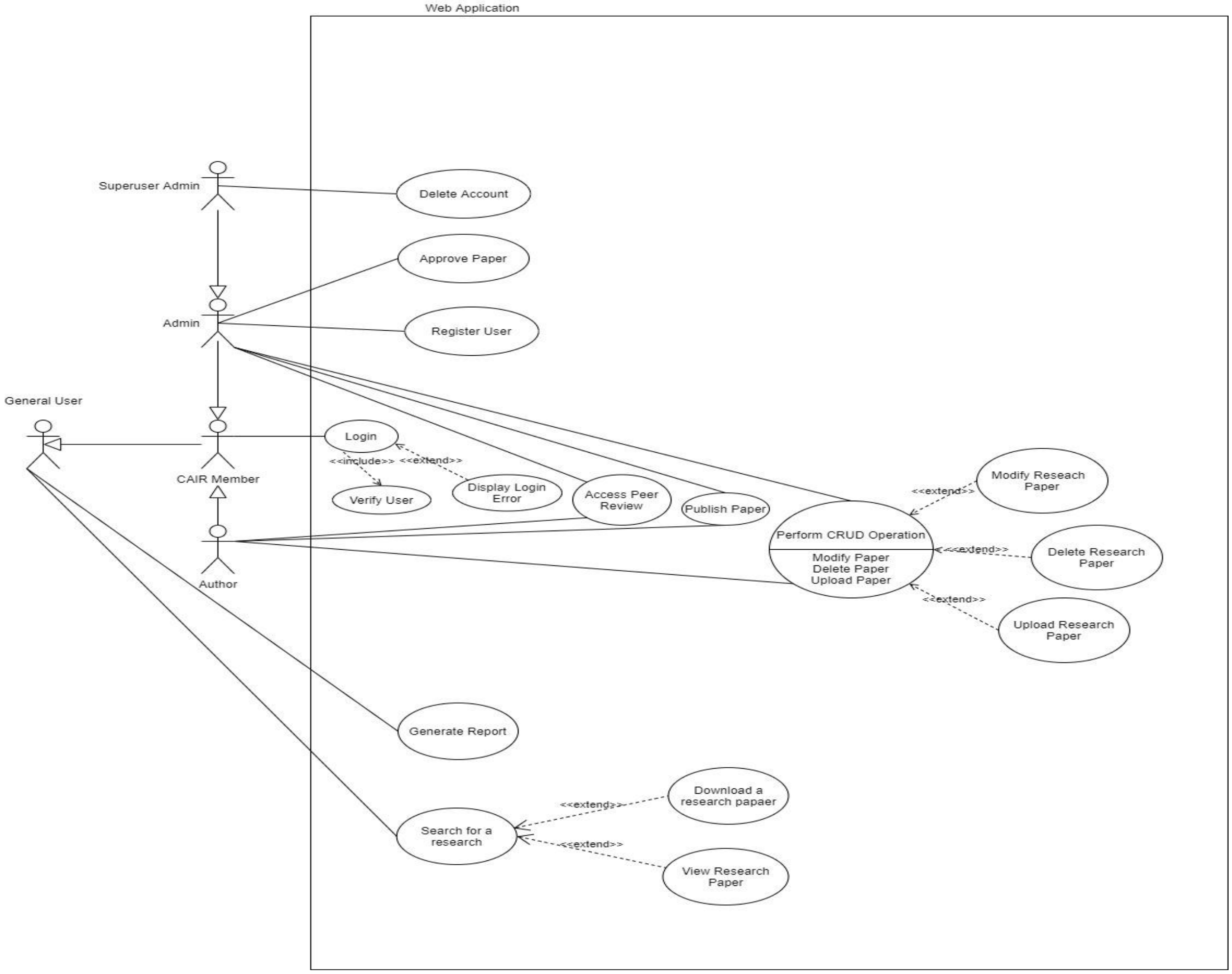
- add, modify, and delete entries corresponding to research outputs
- search entries based on various criteria
- produce various reports based on the information stored

This functionality had to be provided with appropriate levels of security built in, alongside several other specifications described by the client.

This project was developed using a hybrid agile approach, with the final product being built upon the first prototype we designed in the early stages of development after gathering the requirements from our client during the first three meetings with them. The prototype we designed was intended to be evolutionary and so as we developed each iteration of the software every week or two, more functionality was added to it. The approach taken was not completely agile however, as we decided to follow the waterfall structure for testing, of which we did at the end of the development of the final product. As we were building a website, it was rather easy to designate work/functionality amongst different group members and then consolidate the work done at the end of every week or two, once we had the foundation of the entire system set up.

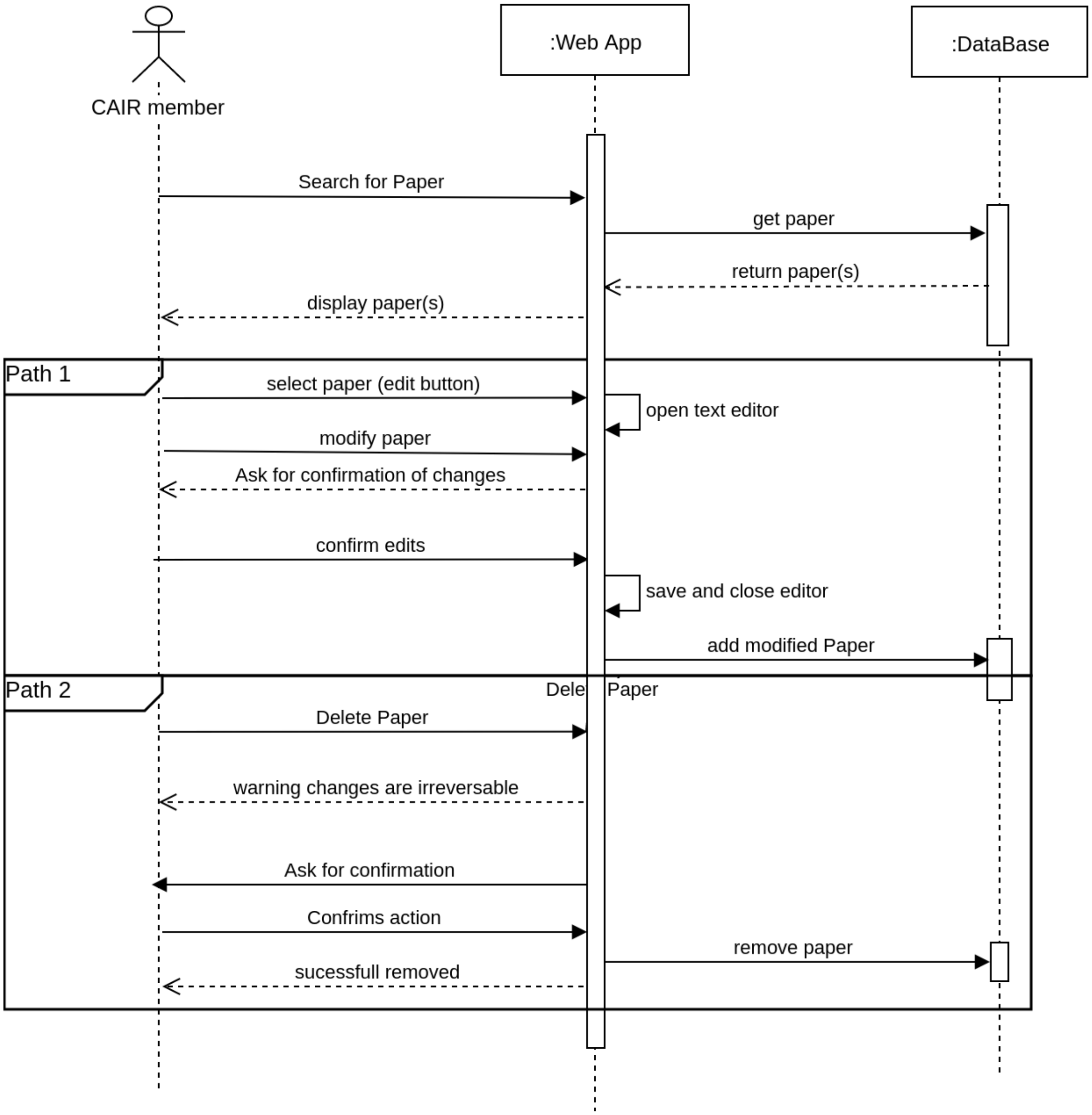
1.2 Requirements Captured

Figure 1: A Use Case diagram of the Cair Web Applications showing the use case of the entire System with different users:



Performing CRUD Operations: Admin. All the users that have valid credentials can perform these operations. To perform CRUD operations, a user needs to be logged in, so they have to be a CAIR member. Different kinds of users should be able to perform specific operations to certain files depending on the type of user they are (i.e admin, general user, CAIR member).In order to modify or delete a research related document, a user must locate the item (through the use of a search bar or scrolling component), which should be a part of a list of documents the user has the authority to modify or delete. Once they've located it, they can then choose to modify or delete the document. If they choose to delete the file, they are warned and prompted to confirm the deletion of the file. Once confirmed the file is deleted and no longer accessible. If cancelled, nothing happens. If the user chooses to modify the file instead, they can choose to upload a missing file, re-upload an updated file, or add missing information tags to the file, like authors, dates, etc. When uploading or re-uploading a file, the uploaded document is stored temporarily to first be verified to be the correct type of file before being permanently stored on the database. If the file is valid, the system should notify the user that the upload was successful, and the file will be moved from the temporary storage to permanently on the database. If the file is invalid, the file is removed from the temporary storage and the user is notified that the upload was unsuccessful, with a possible reason as to why the validation failed.

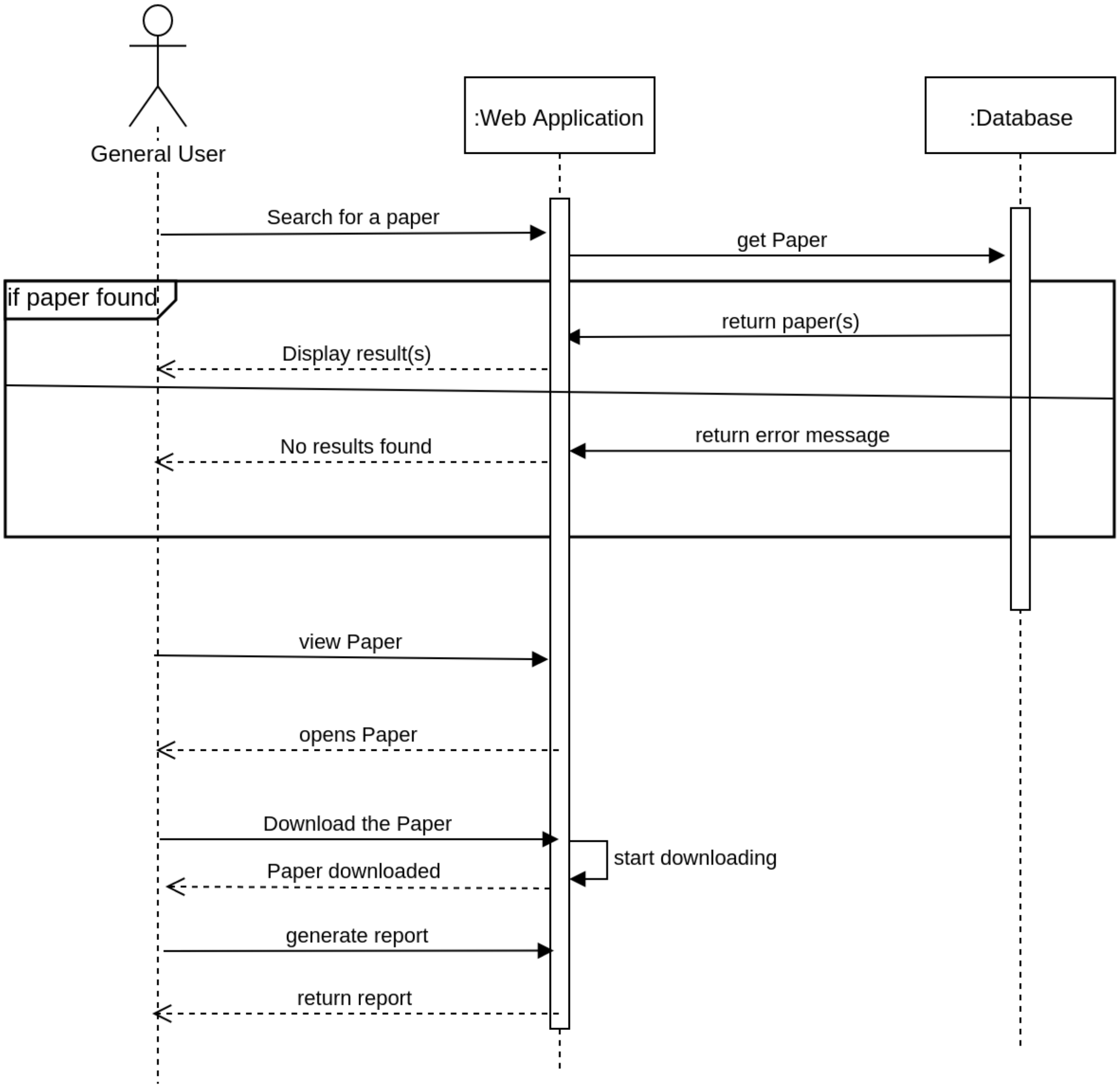
Figure 2: A Sequence diagram depicting the interaction between a CAIR Member, Cair Web Applications and the database



Searching For A Paper: This function can be performed by anyone, even people outside the CAIR organization. Research must be published and Research data (I.e. research paper, abstract) must be available, that is the only condition. The User visits the CAIR website. The website's home page is displayed. The User views researches displayed on the home page or the User filters for specific research using the date it was published, the author's or co-author's name , research group, name(s) of university/universities involved in the research. The website filters and displays the research data (research paper, abstract). The user can choose to view the research paper or abstract, or to download the research paper. The User generates a report using results from the search. If the research publication is not on the database the website informs the user that the research they searched for is not available.

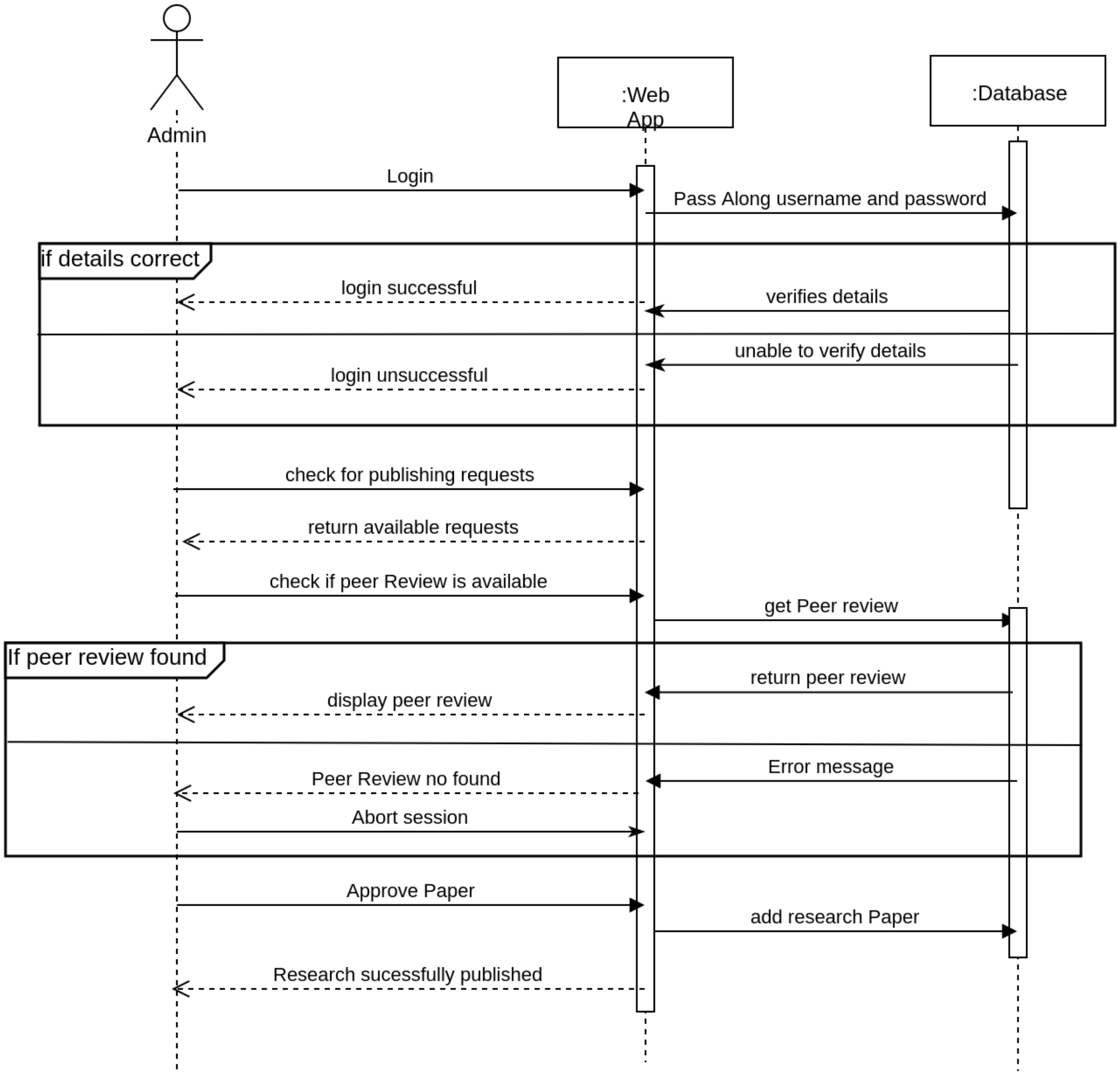
The following sequence diagram show the functionality described above:

Figure 3: A Sequence diagram depicting the interaction between a General User, Cair Web Applications and the database



Approving and publishing. This action is only performed by the administrators (super admin, university admin. The authors submit/upload the research Paper on the Website and request for publication as the condition. They need to include peer review on the request form either by email or by uploading a pdf document. The admin provided they have valid credentials will be responsible for publishing the research paper given all requirements are met. The admin will login on the system. The system verifies credentials and grants access to the admin. The admin checks for publishing requests on the Web Application. The System displays available requests. The admin checks if a peer review is available for the submitted paper. The system will retrieve and return peer review. The admin will approve the paper. The system will publish the paper and send approval notification to the authors. Alternatively, when the user tries to login, the system denies access due to invalid credentials. Sometimes peer review is not found in the database, and the system sends messages to authors requesting a Peer review.

Figure 4: A Sequence diagram depicting the interaction between a Admin, Cair Web Applications and the database



The system has multiple users with different privileges; this is to protect user data from users that have fewer privileges in the system. Loading pages should not take more than two seconds when moving from one page to another. When the user searches for a research publication it must be returned within 3 seconds.

1.3 Design Overview

Figure 5: A Class diagram depicting the interaction between classes of the entire system

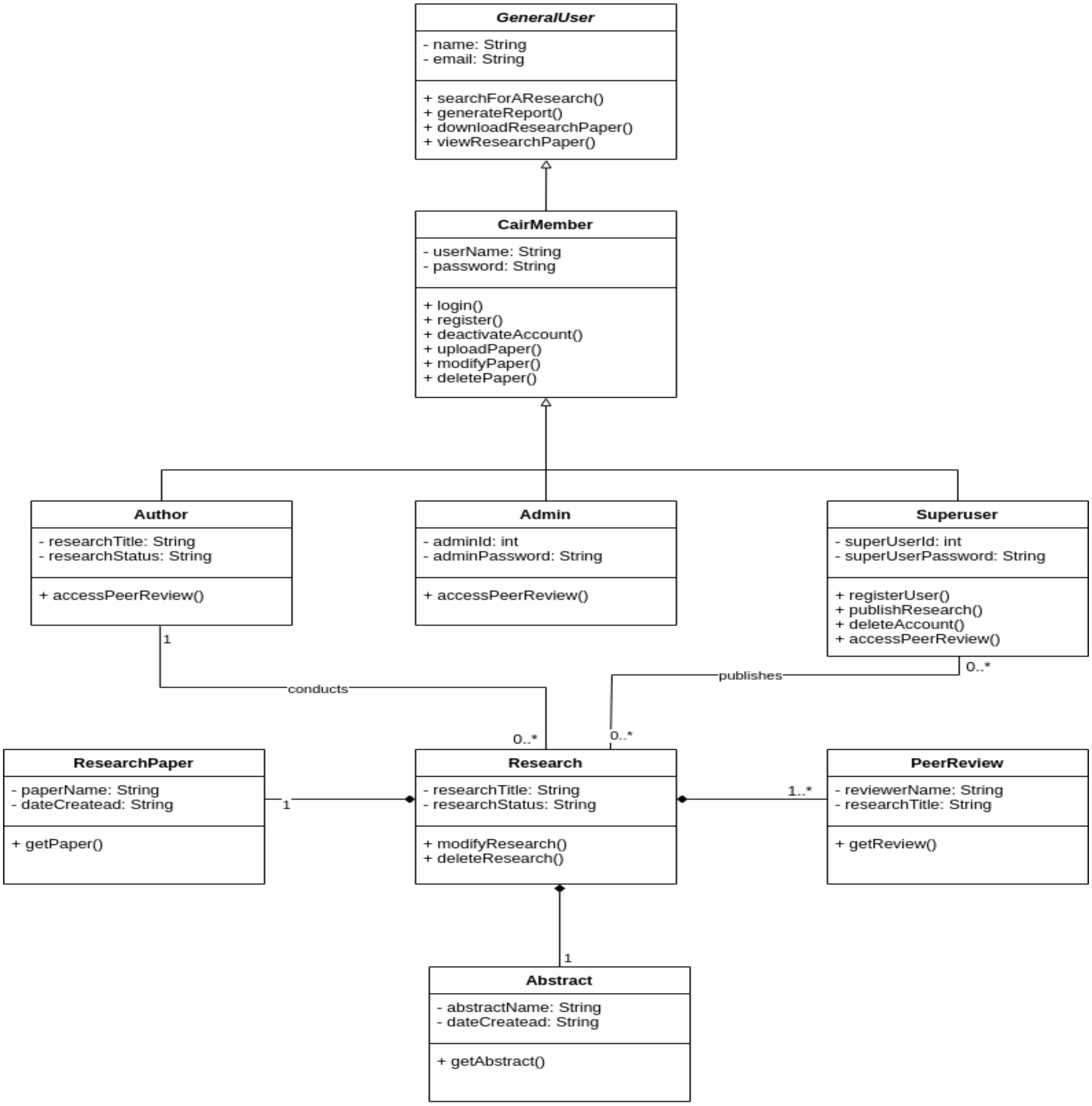
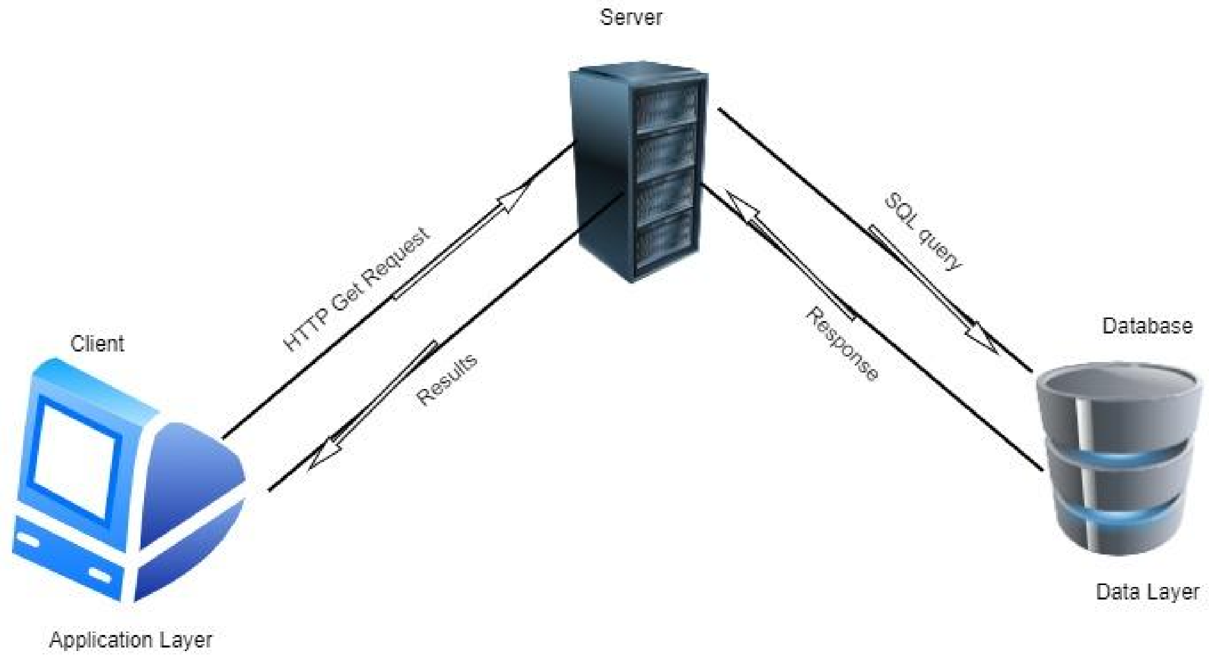


Figure 6: Architecture Design, Client-server diagram



The Centre of Artificial Intelligence Research Web tool is a three- layer system architecture, consisting of commonly used client-server discrete layers. The functionality of the website is based on the integration of different commonly used software tools e.g. a server and a database.

The client layer (which is just the actual website) is the presentation level of the system providing the users with the user interface in order to visually interact with the system, to access and query services that are offered by the website (these services are described in the requirements section). This layer is what the end-users use to communicate with other systems such as the database by issuing a command that performs some function by clicking a button (the complexity of such action is hidden from the users) , the results for executing such a command will be displayed on this layer. This is the backbone of the entire website as it is the layer that end-users can only see and interact with.

The Application Layer contains different types of software that are essential for data manipulation and data retrieval. This is where the connection between the client (web browser) and the MySQL database is established, and this essentially allows data flow between the client and such a database (in the Data layer).

Two different servers were used, the apache and sql server. The apache server is a HTTP server responsible for hosting the entire web tool and storing necessary files essential for the functionality of the tool. It is also responsible for handling requests and redirecting responses between the applications and the client.

And Lastly the Database is used for storing all the information, as mentioned a connection needs to be established in order to access this information and perform crud operations. And this encapsulates the Data Layer.

Website Design

The entire website was designed in such a way that is simple, easy to navigate and to provide more but easy to understand information. This is made possible by the following:

Navigation bar which contains all the links to essential pages, such as the page for searching research publications, the option to login in the system, Members of the organization and contact details should someone want to know more about the services offered by the website. The pages are linked together and this navigation bar is available on almost all of the pages so that the user can easily move around and can go back to the home page at any time they want.

Most of the functionality was encapsulated with a button; meaning that a user can perform critical functions by just clicking a button and the complexity of such a function is hidden from the user.

Getting user input by the use of a keyboard is minimised, the idea is that the user after logging in should be able to perform all the operation with just a use of a mouse. This was made possible by using dropdowns that provide users with all other options for a particular field. For instance a user can search a research publication without having to type anything, but by just using the filter option provided that they know some partial information about the research publication they are looking for.

The elements are groups according to their similarity in terms of functionality. For instance when a user searches for a research publication they are more likely to read the abstract associated with that paper and subsequently download that paper. As a result the option to read the abstract and download the paper are grouped together. The second example would be the crud operations (add, delete and modify research publications), these are also grouped together because they are of the same set of functionality.

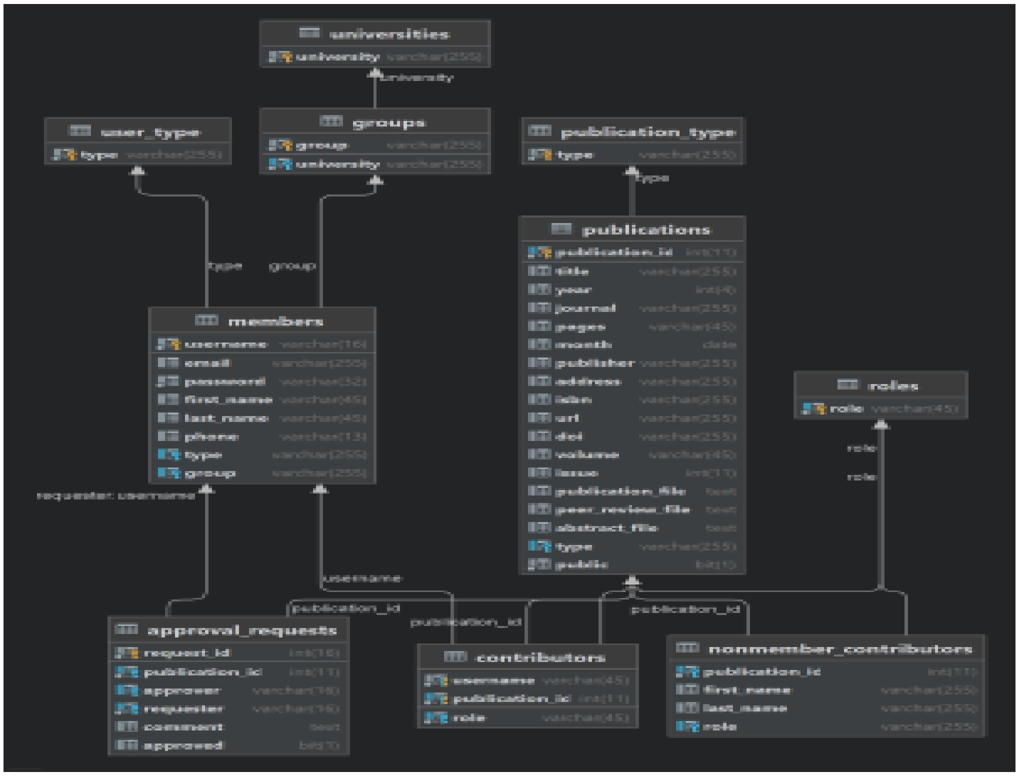
The colours and fonts used are user friendly and up to acceptable standards.

SOFTWARE

Database: MySQL

This is the tool that we used to design and manage our database schema. We chose it due to its simplicity and because it is user friendly. This simplicity is made possible by the GUI provided by the MYSQL workbench. SQL makes performing complex queries faster and simple. And it can meet the demands of growing user data as users continue to add more research papers and more information in the database.

Figure 7: An ERD diagram representing the database Schema of the entire system:



Implementation

Data Structures

Session variables

In order to achieve persistence of data over multiple PHP pages, like the login state of a user, their username, or anything needed to be stored for later use - we have to make use of a PHP feature called sessions. Session variables are stored on the server side, as opposed to cookies which are stored on the client's local system. They provide unique session information for each user that logs in, and also maintain a level of security higher than that of cookies, as this information is not stored locally. The global variable $_SESSION in PHP is an associative array containing session variables available in all scopes throughout a script which we used store and retrieve a user's username, the research group they belonged to, the type of user they were, and several other use cases in which they were necessary.

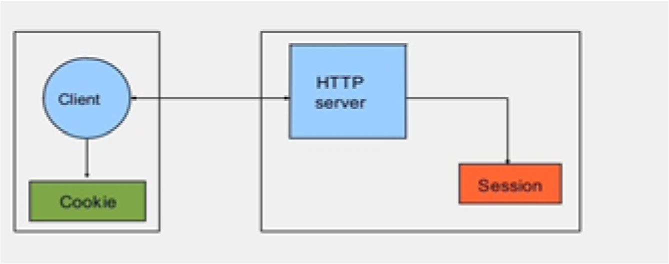


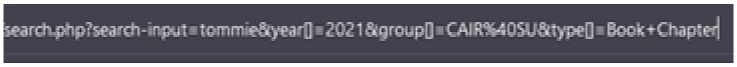
Figure 8- Difference between cookies and sessions

From: https://www.slideshare.net/KapilGupta64/cookies-and-sessions-66638096

GET & POST global variables

The browser client can send information to the web server through two ways, namely the GET method and the POST method. The GET method sends the encoded user information appended to the page request. The page and the encoded information are separated by the ? character (tutorialspoint, 2021). The POST method transfers information via HTTP headers. The information is encoded as described in the case of the GET method and put into a header called QUERY_STRING (tutorialspoint, 2021).

The information appended to the page requests is obtained through data input and submitted into forms on the site by a user, or through manually appending the URL of a page with the encoded information. PHP provides the $_GET and $_POST associative arrays to access all the sent information using GET and POST methods respectively. These data structures were used to store search query parameters in order to filter out search results for research publications, e.g. a selection of years, research groups and document types. They were also used to process certain types of requests, like uploading, publishing, and approving a research paper, by instantiating the variable in the relevant array and executing PHP commands based on whether the relevant variable had been set . The GET method was used mostly to process search requests, while the POST was used to process requests in which sensitive data was being sent, in order to avoid sensitive information input by the user appearing in the URL.

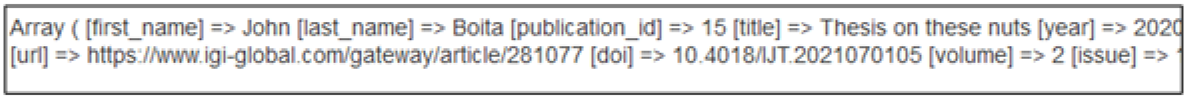


```
search.php?search-input=tommie&year[]=2021&group[]=CAIR%40SU&type[]=Book+Chapter
```

Figure 9 - Example of search query implementing GET method and $_GET variables

SQL Query results

Associative arrays were used to store the returned results of a successful SQL query into the sites database.



```
Array ( [first_name] => John [last_name] => Boita [publication_id] => 15 [title] => Thesis on these nuts [year] => 2020
[url] => https://www.igi-global.com/gateway/article/281077 [doi] => 10.4018/IJT.2021070105 [volume] => 2 [issue] => 1
```

Figure 10 - Example of resultant array from successful SQL query

Technologies, libraries and frameworks used:

jquery-2.1.4 - https://cdnjs.cloudflare.com/ajax/libs/jquery/2.1.3/jquery.min.js

A fast, small, and feature-rich JavaScript library used to make HTML document traversal and manipulation, event handling and animation a lot simpler to do (Js Foundation, 2021).

Bootstrap v5.1.2 - https://cdn.jsdelivr.net/npm/bootstrap@5.1.2/dist/css/bootstrap.min.css

Bootstrap is a potent front-end framework used to create modern websites and web apps. It's open-source and free to use, featuring numerous HTML and CSS templates for UI interface elements such as buttons and forms (Bacinger, 2021). We used this framework for theming a lot of the UI elements on the pages where a user was required to login, like the profile page and page for submitting requests for research approvals. This greatly reduced the time required to manually design and code up the CSS and JavaScript for these elements.

Materialize v1.0.0 - https://cdnjs.cloudflare.com/ajax/libs/materialize/1.0.0/css/materialize.min.css

Materialize is a modern responsive front-end framework based on Material Design. This served the same purpose as Bootstrap and was used for UI elements that Bootstrap was not suitable for.

sweetalert2 - https://sweetalert2.all.min.js

A beautiful, responsive, customizable, accessible (WAI-ARIA) replacement for JavaScript's popup boxes (sweetalert2, 2021). This was used as opposed to repeatedly calling the JavaScript alert() function, as most browsers start to block these alerts after a certain number of them have been called. They also provided better visual feedback for users in order to discern between successful and failed operations, like submitting a request for a research paper that already has a request pending.

XAAMP

An open-source web-solutions kit that provides Apache delivery for a variety of servers and command-line executables, as well as an Apache web server and MYSQL & PHP modules. XAMPP allows a user to execute PHP scripts locally as it creates a web server on their computer to do so. It is a framework that provides a suitable environment for testing and verifying the functionality of projects using PHP and MYSQL, without having to spend money renting out a dedicated server.

HEROKU

Heroku is a cloud platform as a service used to host our MYSQL database at no cost and allowed us to access it from anywhere with a network connection.

JavaScript, HTML, CSS, PHP

JavaScript, HTML and CSS were used to design the front-end (client-side) of the website, whereas PHP was used to execute scripts on the back-end (server side).

User Interface

Methods

The majority of the project focused more on using SQL to query the database for specific data and then displaying those results to the user in a user-friendly way. Almost all the functions we used were thus too simple to warrant discussion in this report. The following are two examples of such functions:

```
function checkIfExists($publication_id,$connection)
```

```
  $sql = "SELECT COUNT(1) FROM approval_requests WHERE (publication_id =
'$publication_id' && approved=1) OR (publication_id = '$publication_id' && approved IS NULL)";

        $result = mysqli_query($connection, $sql);

        $row = $result->fetch_assoc();

        $bin = (int) $row['COUNT(1)'];

        if($bin>0){

        return true;

        }else {return false;}}
```

This method is used to check for any pending requests for approval of a research document. If the approved field of a record in the ‘approval_requests’ table was set to 0, the request had been rejected, if it was set to 1, it had been approved, and if it was NULL it was still pending a decision by an admin. This function returned true if a new request for that particular document could not be submitted and false if it could. The function takes the ‘publication_id’ of a research item and a connection string to the database as arguments.

function array2csv(array &$array)

```
   if (count($array) == 0) {  return null;          }

        ob_start();

        $df = fopen("php://output", 'w');

        fputcsv($df, array_keys(reset($array)));

        foreach ($array as $row) {

        fputcsv($df, $row);     }

        fclose($df);

        return ob_get_clean();

}
```

This method accepts an array and returns a csv file for the user to download.

1.5 Program Validation and Verification

Reliability Testing

The system was tested manually by using incorrect data for instance when trying to login a user that is not registered, for which that the system will verify that the credentials provided are not in the database, thus the user will not be logged in.

A PHP method known as preg_match(), which perform a regular expression match was used to perform the following test.The system was also tested by using data in the incorrect format when adding a new user to the system for instance a name should only contain letters and not any special characters. Should a name contain any special characters, the system will indicate that the name should contain only letters.

The very same approach of testing the system using data in incorrect format was used, however this is the method that was used, filter_var(var_name, FILTER_VALIDATE_EMAIL), in order to validate the format of an email entered when adding a new user.Should an email address be entered on the form in the incorrect format, the system will return a text that the format for the email is incorrect.

The system was also tested manually by trying to submit an empty form, that is with no values entered ,like no name entered. Should a user try to submit an empty form, the system will return a text that values are required.

SECURITY TESTING

The system was tested manually by using data that could be malicious to the system,for instance when a user add details for a new research. A PHP method known as know as mysqli_real_escape_string() is used to ensure that the data that is received from the form is perfectly safe, and will not be harmful by . The function create a legal SQL string that can be used in an SQL statement. For instance the data entered on the form will be entered to the database. An HTLM code entered on the form when adding a new research , the system then returned a mysqli_error.

An Encrypting method and a decrypting method were developed to encrypt the data when it is stored in the database, and it will be decrypted when it displayed and when it is also used. For instance the password of 12345 will be stored as 3hdjsabdhdj53e7fg, that way should the database be compromised , the intruders will only see the encrypted data.However the method was only implemented but never embedded into the entire system.

Table 1: Summary Testing Plan

Process	Technique
1. Class Testing: test methods and state behaviour of classes	Random, Partition and White-Box Tests
2. Integration Testing: test the interaction of sets of classes	Random and Behavioural Testing
3. Validation Testing: test whether user requirements are satisfied	Use-case based black box and Acceptance tests
4. System Testing: test the behaviour of the system	Recovery, security, stress and performance tests

Table 2: Summary of tests carried out.

Data Set and reason for its choice	Test Cases		
	Normal Functioning	Extreme boundary cases	Invalid Data (program should not crash)
Login to the CAIR website	Passed	n/a	System indicate that user is not a member
Downloading a research paper	Passed	n/a	n/a
Uploading document to the website	Passed	Cannot upload large files	Will not be uploaded
Search for specific research paper	Passed	n/a	Display no results
Filter research papers	Passed	Passed, filter by year, group,type of research, etc at the same time	Failed, Display no result
Delete a User	Passed	n/a	n/a
Publish a research	Passed	n/a	Falied, research not published

The results indicate that when Invalid data is used to perform the process of logging into the system, the test case fails, that is the user is not logged in. Invalid data would be the credentials of a user whose not registered with CAIR.Otherwise the test case pass when the user is a registered member.

Users need to upload documents for their researches, when a user attempts to upload a research paper or a peer review the document must have a file extension of pdf, otherwise their documents will not be added, will only be added if they are pdf documents. When a user uploads an abstract it must be a text file, otherwise it will not be uploaded.
The user can publish the research only when all the documents are present, however there must be a peer review for that research, and it must be checked by the superuser or the group admin.
A research paper for a certain research can be downloaded by just clicking the download button.
To search for a research you can type any keywords about a research e.g neural networks, or simply the name of the author, and then press search button then any research associated with neural networks will be displayed.
To filter the researches, you can filter by year, group and many other options available, e.g 2018 , CAIR At SU and then all the researches published during the year 2018, for the group CAIR At SU will be displayed.
The test case of adding a new user into the system can only be done by the Super Admin. The results indicate that it fails when Invalid data is used, this could be when trying to add a new user without specifying which group they belong to or what type of user they are.Should all the required information be specified and correct a new user will be successfully added to the system.
Deleting a user User can only be done by the super user also, when a user is deleted they no longer have access to the system.

1.6 Conclusion

In conclusion, we can say that we were able to understand project requirements, how the tool for managing research outputs should work, and what different kinds of users can do and cannot do, etc. For instance the general users to the system are able to log in, add a new research, modify a research, and even delete a research, etc.Those functionalities mentioned above work very well.

We were also able to design a very well organized database to use for storing, and retrieving information. It stores the user's information for example username, password, etc, and we are able to safely retrieve it when we need to use it.

We were able to design and develop a fully functioning website for CAIR, meeting all the client's requirements.

The user can use the login system to login into the system.The login system was tested by using both invalid and valid data to validate it's correctness.

A manage research feature allows the user to perform any tasks related with to the research, for example adding a research. The feature was tested for every task it can perform, and it was proved that it works perfectly , since all the tests passed.

The Search/Filter feature allows the user to search or filter research based of some criteria. The feature passed all the tests, thus we can conclude that it works accordingly.

The Adding or Deleting a new User feature allows the superuser to add a new user to the system, this feature passed all the tests that were conducted on it, thus we can conclude that it works accordingly

1.7 User Manual

Your system must have a user manual. Append this to your report (make it Appendix A) or bind it separately if it is big. If your system is interactive and has a good user interface with context dependent help then this can be just a cheat sheet. Discuss the level at which your user manual is to be pitched with your client. If

1.8 References

The Ampersand. n.d., PHP - GET & POST Methods. Available from: <https://www.tutorialspoint.com/php/php_get_post.htm>. [October 09, 2021].

Now You Still. n.d., A Simple Bootstrap Tutorial | Toptal. Available from: <https://www.toptal.com/front-end/what-is-bootstrap-a-short-tutorial-on-the-what-why-and-how>. [October 09, 2021].

Normal Code. n.d., PHP: $_SESSION - Manual. Available from: <https://www.php.net/manual/en/reserved.variables.session.php>. [October 09, 2021].

Passing A Parameter. n.d., SweetAlert2 - a beautiful, responsive, customizable and accessible (WAI-ARIA) replacement for JavaScript's popup boxes. Available from: <https://sweetalert2.github.io/>. [October 09, 2021].