# Tranquil Services Website

**NAME:** Paul Stuart

**ID:** 000389223

### TEST CASES:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | ***Test case*** | ***Pre-conditions*** | ***Test steps*** | ***Test data*** | ***Expected Result*** | ***Actual result*** | ***Pass/Fail*** |
|  | What is being tested | What needs to be in place before running the test | The steps involved in running the test for reliable repeatability | Any data that needs to be used for conducting the test | What is the expected outcome | What was the actual outcome | Y/N |
| 1 | Retrieval of Data from DB and render the DOM – Staff Profiles | Tables set up, PHP/SQL connection set up. Back end PHP script with correct SQL statement and variables to store the data for echoing onto the DOM.  Ensure CSS (bootstrap) and DIVS are set up correctly with Dummy image and text so it displays as required. | Test the DB on phpMyAdmin first so it is working correctly. By directly inputting the SQL query and observing the retrieval.  Test the SQL connection so the user, p/w and db are connected to  Then script the PHP, make sure the result is saved to a variable, then test the correct retrieval names in the array with [ ‘column name’].  Set up the DOM so the PHP echo correctly in right places displaying data retrieved. | Dummy String text and a blob file or jpeg file. | The SQL statement should display the data using a SELECT \* FROM TEMPLOYEES  Also it should display the retrieved data on the DOM for rendering. | Once syntax and DB CMS issues were sorted. The expected outcome was produced with the script working and the data being displayed on the website and adhering to the CSS style instructions. | Y |
| 2 | POST to DB from HTML Form | DB GET and POST need to be already working and pre-tested, so we know the connection is solid. | Set up the tag names on the form fields in the html page.  Ensure that a correct SQL statement is being used by running it through the DB CMS first.  Have all the form set up and accepting data. Check that the submit button is linked correctly and the PHP script is in the same HTML section.  The form needs to have correct attributes so it is communicating with the PHP script.  Check through all the POST scripting for the PHP and ensure the names and attributes are correctly aligned. | Enter some dummy text and upload any jpeg to see if it works | The database on php my admin should refresh and show a new row that has been posted from the html form via the php script | As expected, new row added to the employee table | Y |
| 3 | Ensure CSS div wrapping occurs with each new employee entry | Need to configure bootstrap 4 styling with row and columns, so each new div that gets added will either be side by side. Or if it is uneven numbers will wrap to the next line. | Ensure that the divs are all inside a container using bootstrap tag=”row”, this allows for wrapping. Set up colours, padding, hr, etc. Test with DIVS just using the html page, without retrieval.  Once it is working, then set up the variables and echo statements in php, so when the **for** loop runs, it produces the DIV styling to remain consistent and looks the same as when the divs were rendered using just html.  With each retrieval when the script runs it will scan the DB and pull out any newly added employees. Displaying them automatically by creating new DIVS with styling.  Test this firstly by just updating the DB manually in php myAdmin before using the post method from a form HTML page. | Dummy String text and a blob file or jpeg | The new staff member should automatically be added and the divs should wrap to the next line if uneven numbers.  Also, the styling should all look consistent using the pre-defined bootstrap settings. | Works fine, some tweaking needed for the DIVS margins etc. But the process works fine. | Y |
| 4 | Show/Hide Articles | Have dummy text and image set up, set all bootstrap styling up first, ensure button is configured with ID to trigger functions in jQuery | Ensure that the js file is linked correctly in the HTML page.  Make sure the correct syntax and ID for selectors have been set up. Test that the button listener is set up.  Ensure also the ID and the class for the DIV are different names, then run the jquery hide(toggle) method. Refer to the W3 schools and Jquery site. | Dummy String text and a jpeg | Should be able to toggle the div on and off with the button. Also there is a smooth jquery animation transition. The heading remains but the image and text toggles. | Works correctly | Y |
| 5 | Logo Animation start | Need the DIV and canvas stage set up. Using Konva js library linked in the HTML page | Ensure the shape objects are made with dimensions and position required. Create separate objects for the text.  Using tweens and animation classes to position the objects where needed.  Set up timeout functions and stop/start functions for the animation of the shapes on stage. Animation is only temporary, runs once on page load then fades out to allow for animated nav/slide show for the services section | Konva library CDN link set up Test objects are rendering on the HTML page within the section required | The animation loads from blank screen and rings fly in off the page to display the full logo. | Works OK, however, cannot get the rings to start fully transparent off the page | Y |
| 6 | Show/Hide DIVS for services on main page | Need the DIVs set up with IDs, need to have document on load in the JS file, linked correctly to the page. Jquery and bootstrap linked to index page. Set up images and dimensions.  Have text from Tranquil team for the services | Create one div that displays and aligns as required with styling. Must look as needed first.  Set up all the buttons the IDs are linked to show/hide Jquery functions.  Test one of the buttons, get it working correctly first. Ensure the DIV shows/hides with on click event.  Get 2 buttons working with the show/hide, ensure it behaves as required. All other DIVS should be hidden whilst one is showing.  Once 2 buttons are working, set up the rest by transferring the code and changing the selector IDs. | Have all the data from Tranquil Services initial consult ready as this will be what is displayed on the DIVs are the various functions are called | There should be a clean and smooth transition between the DIVs showing/hiding. There should be no limits on how much you can show hide. It should all be produced with each click. The close button should close off all the DIVs | Works correctly | Y |

### RESULT NOTES:

Any notes or comments regarding the testing results, fixes and future changes/improvements should be made here.

* A lot of issues with php my Admin, and with the blob files.
* The database was temperamental in accepting some jpegs but not others. This was very frustrating and the DB still is not 100% in accepting JPEGS.
* Some configuring was needed to adjust the timeouts and the file limits on the DB CMS. This was stressful and time consuming, taking a lot of time away from the site testing and development
* The databases also had to be completely recreated towards the end of testing because of SQL and XAMMP needing updates. There were a lot of new error messages that came up, even when testing directly in the database CMS, and even after already successfully testing the post/get php scripts. Again, this took up a lot of time and delayed the testing of the site and functionality.
* There were also some configurations that needed changing with direct SQL inputs into the DB CMS to update the system itself and some of the default settings.