# Iteration 2 – Test log

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Description of test | Test data (if required) | Expected outcome | Actual outcome | Comments & intended actions |
| Stylesheet linking to HTML |  | The CSS should link to the HTML and apply styles. | The CSS doesn’t link, and no changes are present. | I will double check the HREF to the CSS and make sure its pointing to the correct place. |
| Stylesheet linking to HTML |  | The CSS should link to the HTML and apply styles. | Expected outcome |  |
| Custom font using CSS |  | The font should change for all the text on the page. | There are no font changes, even though CSS is correctly linked. | I was incorrectly applying the font to the text on the screen. Re-testing. |
| Custom font using CSS |  | The font should change for all the text on the page. | Expected outcome |  |
| Navbar appearance |  | The navbar should be evenly laid out like the design. | The navbar isn’t correctly lined up with its elements, nor is it in one straight line. Padding is also incorrect. | I will alter the padding to make it more fitting, use the flex display type to keep everything in one line, and use the align-items property to keep everything vertically centred. |
| Navbar appearance |  | The navbar should be evenly laid out like the design. | Everything is now inline and looks right, but the text is too small and not of the right colour. | I will make the text bigger and colour it to the same as the design. |
| Navbar appearance |  | The navbar should be evenly laid out like the design. | Expected outcome | Everything now looks great, but the links don’t work. We will conduct separate tests for that. |
| Navbar functionality |  | The navbar should have links that are clickable and redirect. | The links aren’t links and don’t take the user anywhere. | I will replace the elements with “A” elements, which you can attach “HREF” properties onto to link them to other areas of the application. |
| Navbar functionality |  | The navbar should have links that are clickable and redirect. | Expected outcome. |  |
| Button styles |  | The buttons should be customised with a green background and white text. | The buttons are the default browser buttons and aren’t styled properly. | I will use CSS to style them to how the design has them. This includes changing the border-radius, colours, and size. |
| Button styles |  | The buttons should be customised with a green background and white text. | The buttons now look great but are unresponsive and show no interactivity. | I will add slight animations and transitions to make them come to life. I will use the CSS Hover and Active events to slightly enlarge/reduce the size of the button. |
| Button styles |  | The buttons should be customised with a green background and white text. | Expected outcome |  |
| Error card |  | When an error occurs during a process, an error should show with a nicely formatted card. | Nothing happens, and a message is printed to the console. | I will instead pass a variable through the render-template function provided by Flask and use it on the HTML page. |
| Error card |  | When an error occurs during a process, an error should show with a nicely formatted card. | Now an error message shows, but is not nicely formatted or on a card. | I will use CSS to draw up a card behind the text and nicely align it. |
| Error card |  | When an error occurs during a process, an error should show with a nicely formatted card. | The card shows, but the alignment is not correct. Also, the font-size isn’t suitable, and the card seems very bland. | I will change the card colour to something more fitting, increase the font size, and add an icon to give it a bit more detail. |
| Error card |  | When an error occurs during a process, an error should show with a nicely formatted card. | Everything looks great, but alignment is still slightly off. | I will use flex’s justify and align properties to properly align the content. |
| Error card |  | When an error occurs during a process, an error should show with a nicely formatted card. | Expected outcome |  |
| Login form |  | The login form should look almost exactly like what was designed in Task 1 using CSS. | The form looks nothing like that currently. The inputs are completely un-styled, the header is wrong, and the layout is wrong. | I will firstly sort out the layout by using the flex display type and the associated properties it allows us to use. |
| Login form |  | The login form should look almost exactly like what was designed in Task 1 using CSS. | The layout is now correct, but the colours are wrong, the spacing is slightly off, and the inputs are still completely un-styled. | I will change the colours to make them more appropriate, and style the inputs as they are in desperate need. |
| Login form |  | The login form should look almost exactly like what was designed in Task 1 using CSS. | The layout and colours are correct, but there are some very small issues with spacing and the font-sizes aren’t quite right. | I will fix the issues with the spacing by using a combination of gaps, margins and paddings, and then alter the font-size. |
| Login form |  | The login form should look almost exactly like what was designed in Task 1 using CSS. | Expected outcome. |  |
| JavaScript form disabler |  | The JavaScript should disable the form’s inputs and buttons when a submission is being processed. | The form doesn’t disable anything. | I will need to use JavaScript to create some code that will do this for me. |
| JavaScript form disabler |  | The JavaScript should disable the form’s inputs and buttons when a submission is being processed. | The form doesn’t disable anything. | The script isn’t loading as I forgot to link it using the script tags in the HTML. Let’s try now. |
| JavaScript form disabler |  | The JavaScript should disable the form’s inputs and buttons when a submission is being processed. | The form doesn’t disable anything. | The script loads, and I can see that because I have used debugs in the console. However, the form element is returning as “undefined” for some reason. |
| JavaScript form disabler |  | The JavaScript should disable the form’s inputs and buttons when a submission is being processed. | The form doesn’t disable anything. | I am going to try and wait for the page to load first and then access elements like the form. |
| JavaScript form disabler |  | The JavaScript should disable the form’s inputs and buttons when a submission is being processed. | The form is found, but an error occurs when accessing the children’s children. | I am going to properly check the node name types and make sure I am not typo-ing. |
| JavaScript form disabler |  | The JavaScript should disable the form’s inputs and buttons when a submission is being processed. | The form is being disabled but it isn’t very clear to the user as there’s no unique styles. | I am going to add some unique styles using CSS. They will affect both the inputs and the buttons. |
| JavaScript form disabler |  | The JavaScript should disable the form’s inputs and buttons when a submission is being processed. | Expected outcome |  |
| Landing page |  | The landing page should look exactly as designed in Task 1. | The landing page has no content, only the navbar. | I will firstly research how to place text over an image using CSS online. Then, I will test that method out and see if it’s suitable. |
| Landing page |  | The landing page should look exactly as designed in Task 1. | The landing page now has an image with some hero text over the top. It works very nicely. I just need to correctly position the text and add a sub-title. | I will add better title text and subtitle text and reposition it so it fits nicer. I will do this by adding the padding on that has been applied to the other pages. |
| Landing page |  | The landing page should look exactly as designed in Task 1. | The landing page has a nice image with a border-radius effect, and some nice text over the top. It needs some fixing however, as when resizing the page it causes a lot of problems. | I will first set a max height to the image by setting a max height to its container and setting the overflow property to clip. This will stop the image from showing outside of its container’s boundaries. |
| Landing page |  | The landing page should look exactly as designed in Task 1. | The image now no longer resizes outside of its container’s boundaries and sits nicely on the page even when full screen. When resizing the screen too small, the border radius gets cut off though. | I will set a border radius to the image container as well as the image itself. |
| Landing page |  | The landing page should look exactly as designed in Task 1. | Expected outcome |  |
| Navbar |  | The navbar should look exactly as designed in Task 1, as well as show additional links when hovered. | The navbar shows no interactivity and links don’t redirect anywhere. | The first thing I’ll do is add the HTML structure so that I know what content I need and what I don’t. |
| Navbar |  | The navbar should look exactly as designed in Task 1, as well as show additional links when hovered. | The navbar now permanently shows additional options that are meant to only show when hovered. | I will use CSS to hide the additional options unless the link is hovered. In that case, the options will fade in. |
| Navbar |  | The navbar should look exactly as designed in Task 1, as well as show additional links when hovered. | Now the links only show when hovered, but they are slightly misaligned, and the chevrons don’t change even when there is info being displayed. | I will make the alignment adjustments to the links and make the chevrons rotate 180degrees with a nice transition when hovered. |
| Navbar |  | The navbar should look exactly as designed in Task 1, as well as show additional links when hovered. | Expected outcome |  |
| Banner |  | The banner should look exactly as designed in Task 1. | After adding the banner, it is almost perfect first time, but the width of it is too large. | I removed the width property, so flex should naturally take up the available space. |
| Banner |  | The banner should look exactly as designed in Task 1. | The width is good now; however, the font colour is black and not the dark green the rest of the site uses. | I will change the font colour to the correct dark green. |
| Banner |  | The banner should look exactly as designed in Task 1. | Expected outcome |  |
| Landing page content |  | The content should be exactly as designed in Task 1, plus some more that wasn’t included in the design. | There’s no content on the landing page, only the hero image and associated text. | I need to add the cards that advertise the services for the application. |
| Landing page content |  | The content should be exactly as designed in Task 1, plus some more that wasn’t included in the design. | The cards now show but aren’t properly centred on the screen. | I used the wrong alignment property for flex. I used centre instead of justify-between. |
| Landing page content |  | The content should be exactly as designed in Task 1, plus some more that wasn’t included in the design. | The cards are properly placed, but there’s a lot of content missing. Such as the informational advertisement. | I will advertise the informational content on the website in a similar way. I will create a new section and use a green background, however, instead of a white one. This shows separation. |
| Landing page content |  | The content should be exactly as designed in Task 1, plus some more that wasn’t included in the design. | The background is there but there’s still no content. | I need to add the content that is missing. |
| Landing page content |  | The content should be exactly as designed in Task 1, plus some more that wasn’t included in the design. | The content has been added. I think this is enough for now, but I might add more in the future to fill out the page even more. |  |
| Card grid layout |  | The cards should be lined up with not too much spacing. | The cards have too much spacing and aren’t big enough. | I will increase the width and height of the cards using CSS. |
| Card grid layout |  | The cards should be lined up with not too much spacing. | The cards’ size is now correct, but there’s still too much space between them. | I will use the “grid” layout in CSS to place the elements in the correct positions. |
| Card grid layout |  | The cards should be lined up with not too much spacing. | Expected outcome |  |
| Transparent card design |  | Transparent cards should have no proper border and “float”. | They look ok, but the dashed border doesn’t work well. | I will try using a shadow instead. |
| Transparent card design |  | Transparent cards should have no proper border and “float”. | Looks much better, though very static. | I will add a subtle animation when each card is hovered. |
| Transparent card design |  | Transparent cards should have no proper border and “float”. | Expected outcome |  |
| Navbar responsiveness |  | The navbar should have some of its elements removed when the screen is too small. | There is no reduction, and the screen expands past it’s bounds causing scrollbars to appear. | I will use CSS to remove the content that is least necessary when the screen size is too small. |
| Navbar responsiveness |  | The navbar should have some of its elements removed when the screen is too small. | Same thing, perhaps I made a CSS error. | I selected the wrong elements in CSS. I have now corrected it; retest. |
| Navbar responsiveness |  | The navbar should have some of its elements removed when the screen is too small. | Expected outcome |  |
| Banner responsiveness |  | The banner should remove the middle element when the screen is too small. | The banner expands past the screen as it trying to save all the content. | I will use CSS to remove the middle element. |
| Banner responsiveness |  | The banner should remove the middle element when the screen is too small. | Expected outcome |  |
|  |  |  |  |  |