This list is for Client Side Scripting

Code Utilised -

Contact form – Cloudways <https://www.cloudways.com/blog/custom-php-mysql-contact-form/> validated, has all content and meets requirements (contact name and email address, artist album to help better find the record)

Used because it comprehensively and cleanly presents the necessary form layout and JS actions – the main configuration required was converting the example form inputs into those necessary for my database

Image gallery – W3 Schools <https://www.w3schools.com/howto/howto_js_slideshow_gallery.asp>

Used because it gets the job done neatly and simply – just had to swap out the example images and text

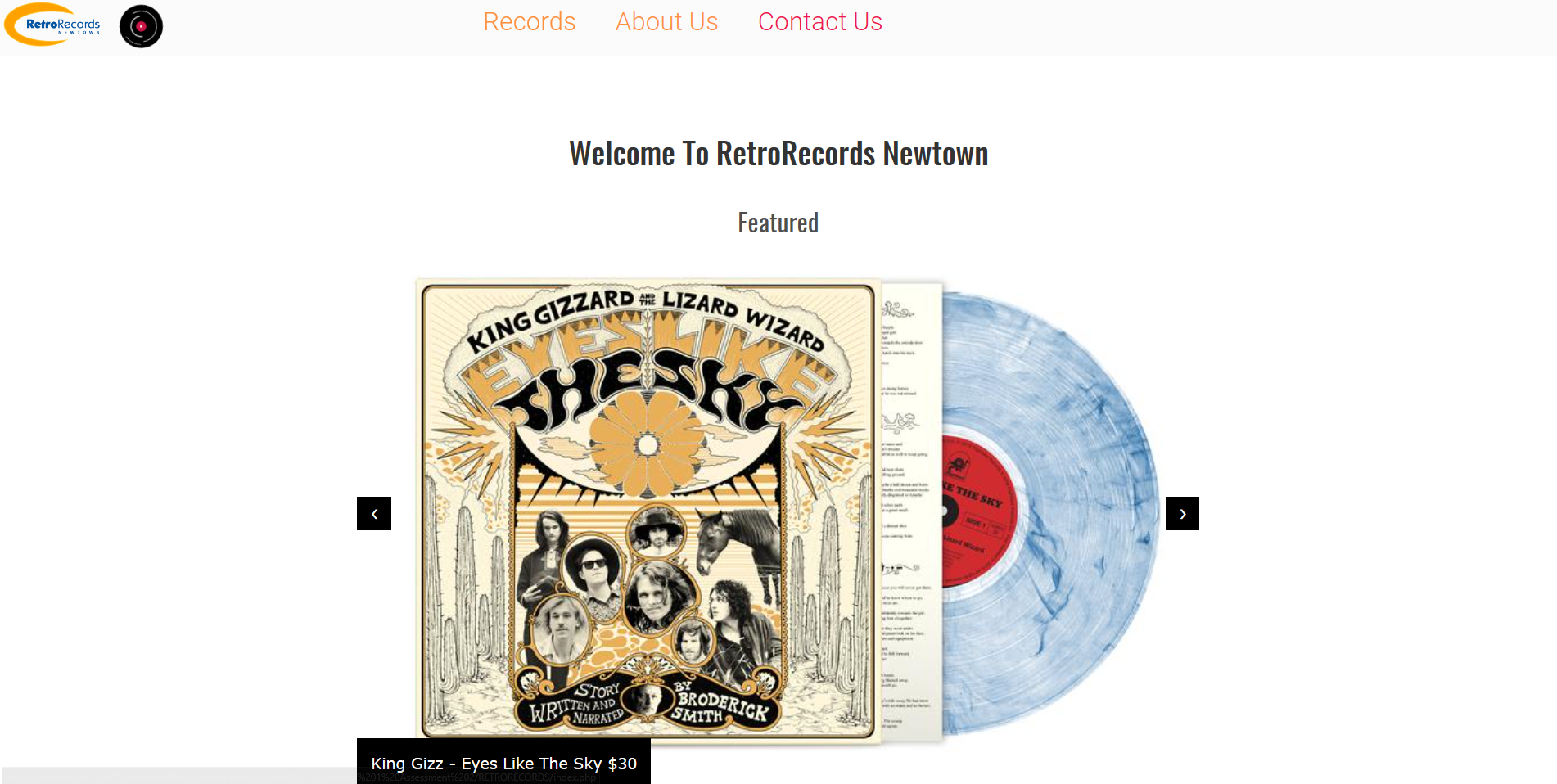
Collapsible FAQ – W3 Schools <https://www.w3schools.com/howto/howto_js_collapsible.asp> animated JavaScript

I really liked the animation on this one – positioning and stripping out the text for retro records own was all that was required

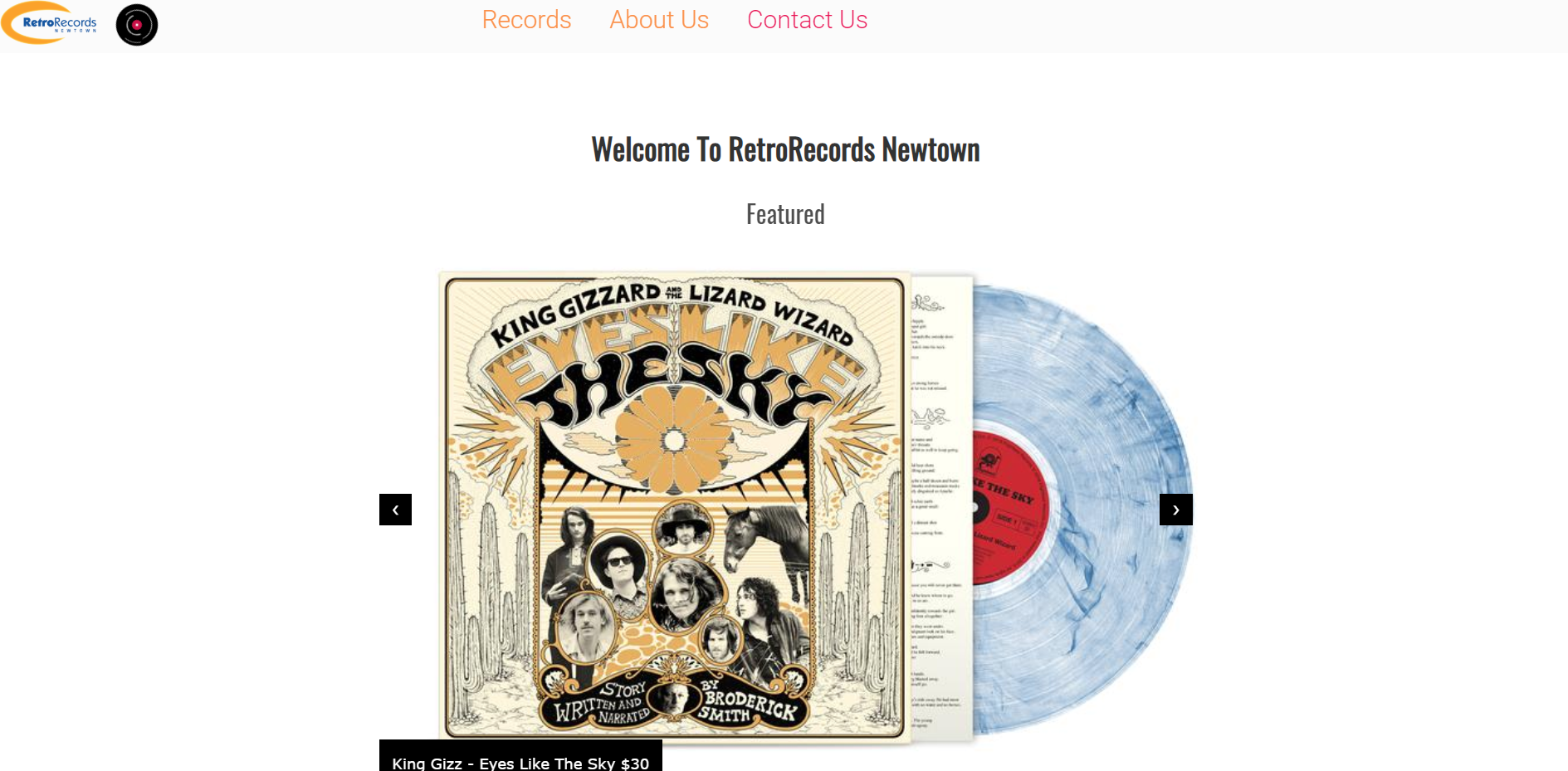
JavaScript confirm window – self-built, I found it highly challenging to build an alert window that prevented the data from being submitted regardless of what buttons were pressed and spent a fair amount of time researching what was required to implement with what I already had

Testing on multiple browsers / image gallery:

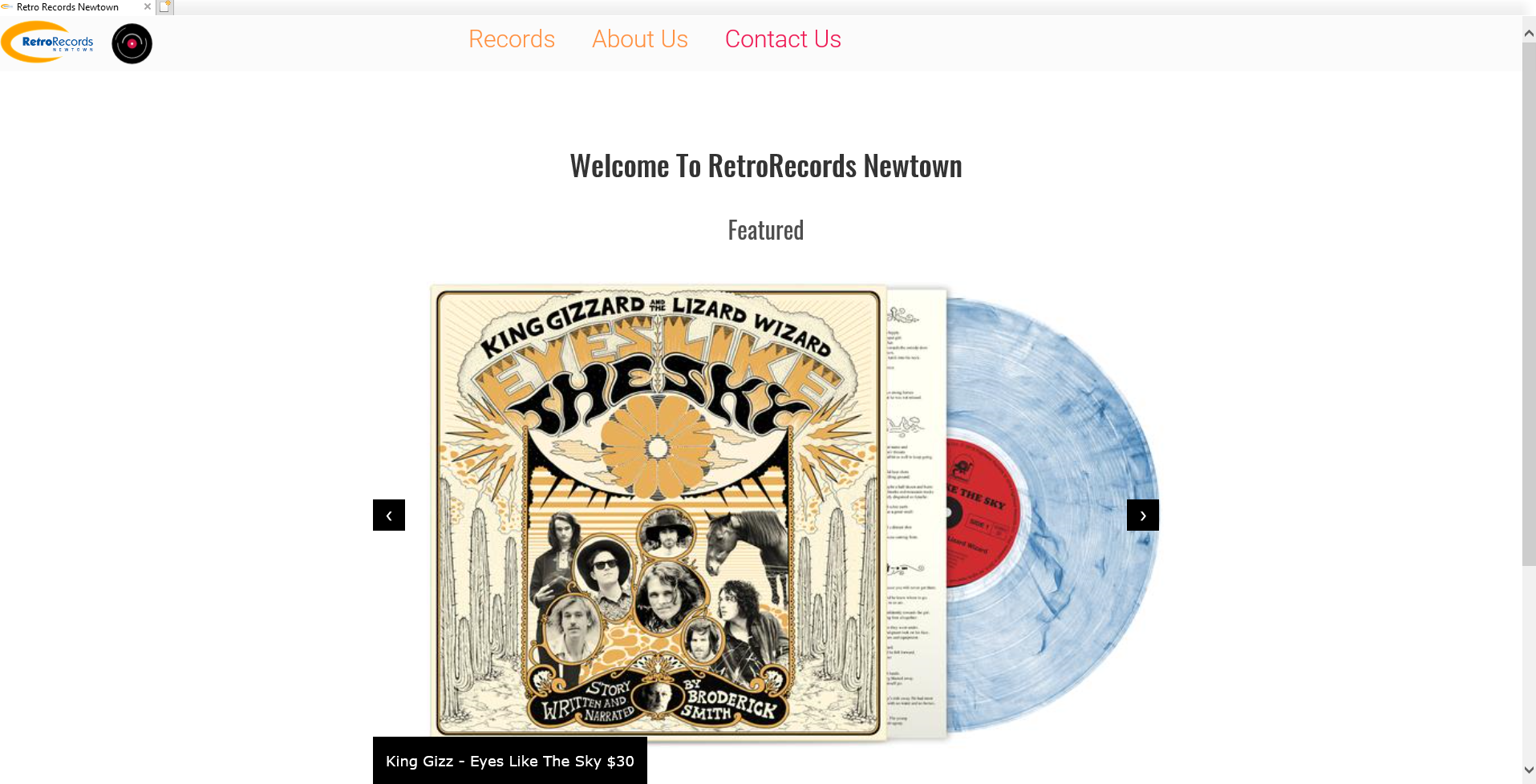
Edge – confirmed



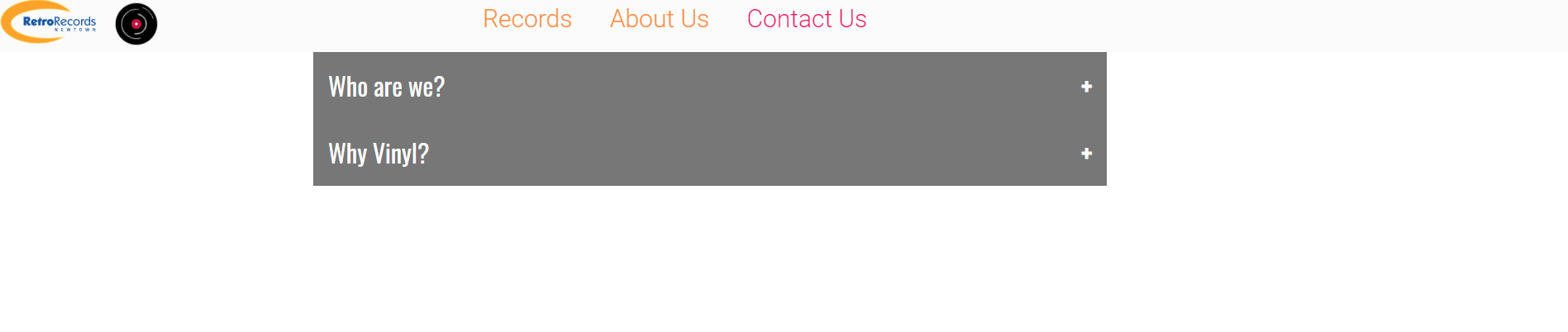
Chrome – confirmed



Internet explorer – confirmed



Dropdown

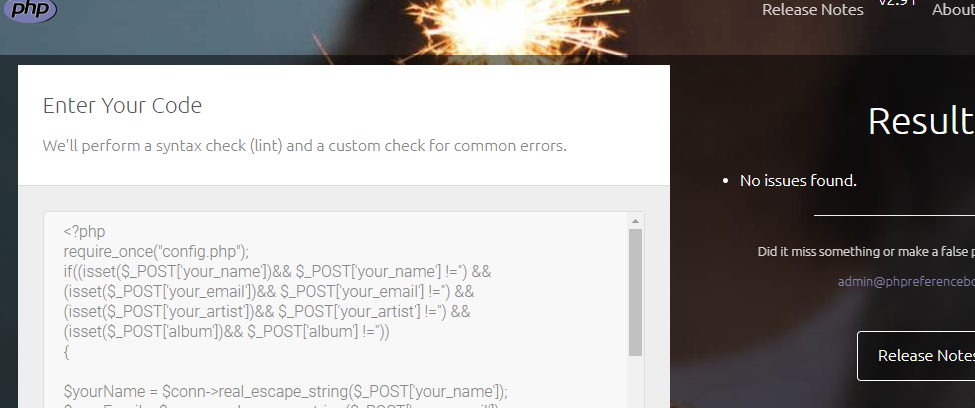


Much of the code had already been validated and carried over from the html and css



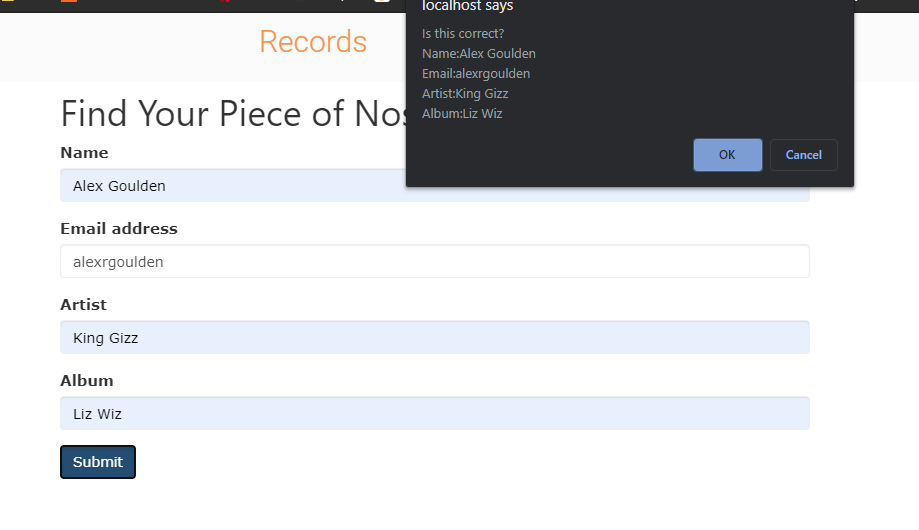
JS valid

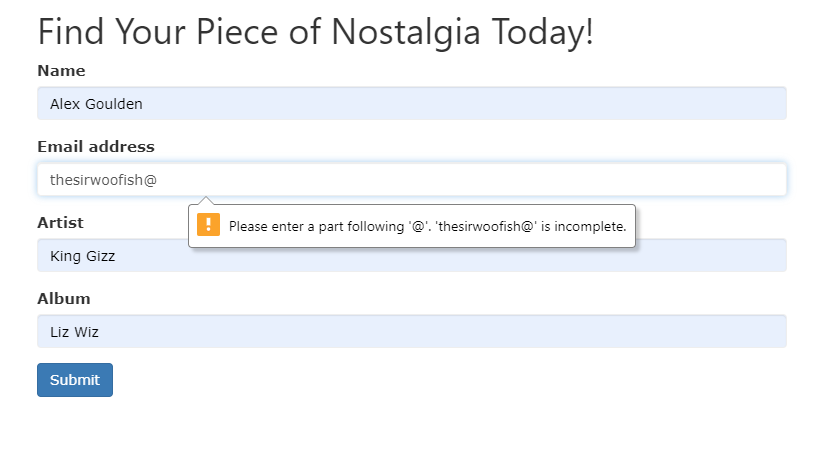
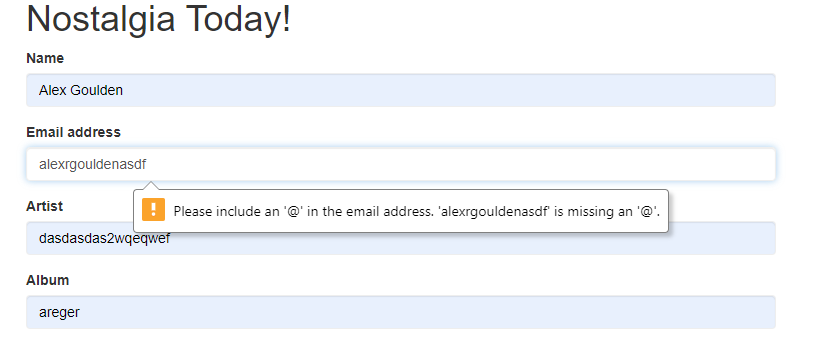
Validation – php



Form Validation –

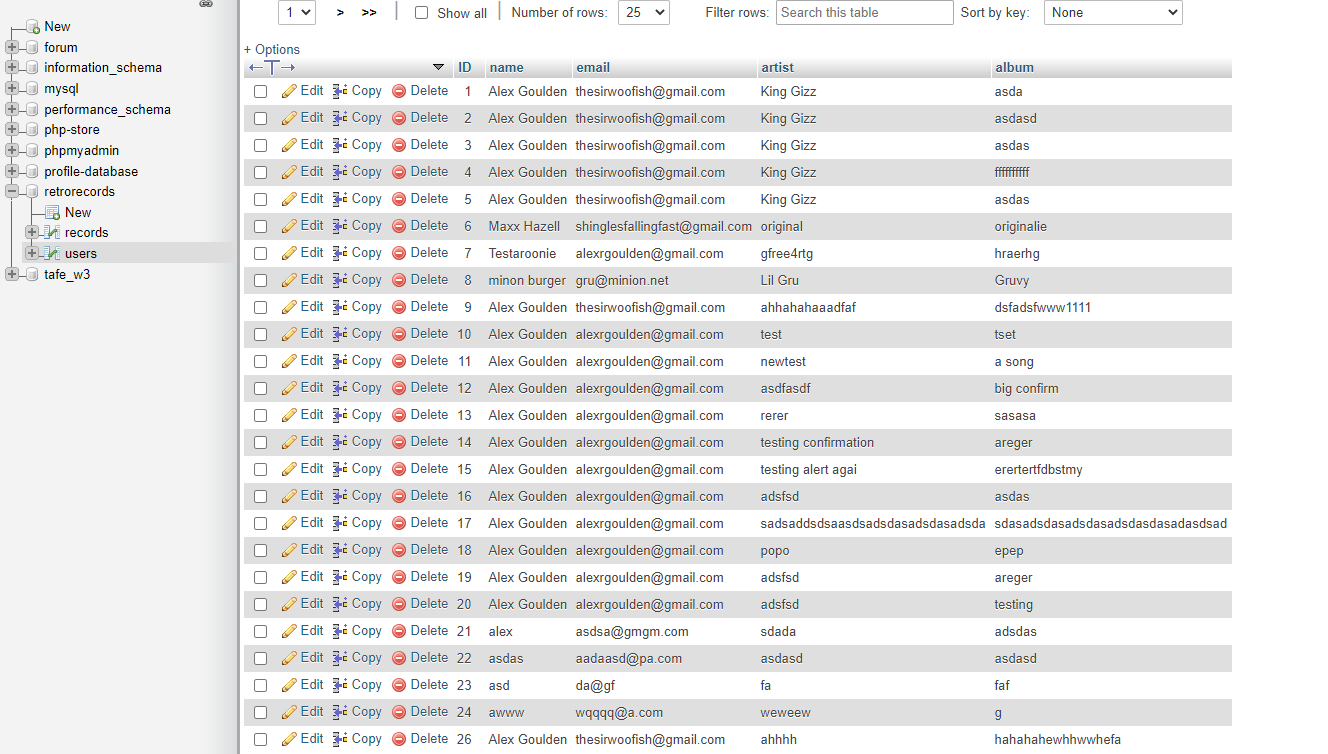
The form checks to see if the user is prepared to send their data to the server – if a user has entered the information incorrectly it is caught and then notified to the user, prevented from submission





Issues –

The primary issue I had to deal with was creating a confirmation window through javascript, as the process of inputting the necessary data to submit and check the if the window was correct led to many, many dummy database queries.

The biggest issue of this was learning that there could only be one declared document.getelementbyid in a function – which meant I needed to create a variable for each. 