This document summarizes the contributions made within this portfolio.

**User Registration/verification**

This code amongst some other snippets is responsible for creating the page where users can register for and sign into their accounts. The page consists of a simple form created in HTML; when the user enters their details the results are checked with JavaScript on the front end to ensure they’re using a real email address and save some potentially wasted power on the back end. An email is then sent to the users email address containing an md5 hashed word that they must retrieve and enter back into the website to verify they’re a human. This changes the active column in the users Table to 1 and allows the users to get onto the main page of the website.

**View & Download Files**

This code is responsible for retrieving data from the MySQL database with PHP and displaying the data in an easily comprehendible way to the user. There have been many changes to the way the data has been presented due to the demands of the client, but I’ve finally come to an agreement with them on the design choices. It was required that there be relatively large preview data alongside the information that goes with the media, so I used anchors on the data that is matched with the media to seamlessly show the media and data simultaneously by clicking their respective links.

**Uploads, Verification, and automatic sorting**

One of the requirements given by the clients asked that the media that’s uploaded be automatically stored in their correct folders e.g. (music, movies, etc.). To achieve this I examined the file types of the media as it is being processed by PHP. Using a large collection of MIME media types (“application/pdf”, “audio/mp3”, etc.) the files are examined and compared to this large sample of data types to be sorted into their correct *directories*. As would be expected the users are given notifications about how the downloads have been processed to let them know whether they’ve used incompatible file types or that they succeeded to go through.

**Updated GUI**

While remaining similar to the template we had been previously using that was set up by Michael a new one was required in order to present the data to the users. Again using a HTML5UP template I then modified the code to contain all the links required and organized the presentations into the classes the template provided. Other than inserting new icons and data into the tables not a great deal was changed.

**Delete Files**

Since the media vault is planned to have a data limit per user it was required that users had the ability to clear up space. Giving users the ability to delete files from the server is potentially a risky manoeuvre to allow so in order to help prevent any attacks I set up a base search variable to limit potential attacks to the lowest directory possible. Thorough checks are also put in place with the SQL strings by escaping all the characters and a few mandatory WHERE statements.

With a lot of these server side processes there is much room to improve. Lots of calls to pages dedicated to nothing more than entering data to the database are made frequently causing the user to have the browser redirected twice in a case that could potentially be zero. If I can get familiar with AJAX soon the project would benefit greatly from dynamic pages instead of all the forced requests it’s going through at the moment.