This file will be a document explaining the differences in design decisions between the Use Case Document created during the initial analysis of the project requirements and the actual implementation of a solution. While certain aspects of the projected solution have been retained, others have been modified or entirely transformed to better suit the needs of the assignment.

1. User File Management

User File Management has not been implemented into an interface as of right now. Currently, files are stored and manipulated on the file system via SQL (a more efficient method of manipulation for photographs over 1MB in size), but manipulation via the website has not been managed. Instead, file manipulation is only possible through uploads via the website, and through manually changing file names and locations with SSH.

1. Image Assessment

Image assessment has begun using one of the example python image assessment libraries provided. This has given us a good jumping off point, both for a reference point when it comes to a more complex image assessment algorithm later on, and allowing us to test and get a feel for how this implementation will operate within a server. Currently, we project a change with when image assessment will occur - we plan to have images be assessed as they are uploaded, instead of having to separately initiate uploading and assessment.

1. Photo Browser

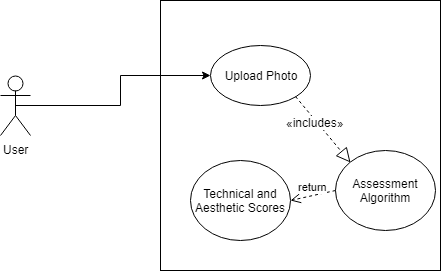
The photo browser, like File Management, has not been implemented - however, this will be a much easier task than full File Management. Once File Assessment has been worked on further, we will utilize the website to display uploaded images and their associated rankings.

1. Train Model

Currently, the Train Model portion of the project is the already-implemented library, expounded upon in Image Assessment. As a result, training has not occurred yet. When training is implemented, we will not have one singular ‘trained’ model, but will train multiple models based on the images classification - Football, Crowd, etc. As a result, image assessment will occur in two layers - classification, to associate an image with its best-suited model, and actual assessment, to provide a rating based on images in the same category.

Current Implementation:

Our current implementation is utilizing Amazon AWS for the image assessment and storage library. Currently, we have implemented a basic website that allows for users to upload images. In addition, we have set up an image assessment library and seeded it with dummy data to test its capabilities. We have also implemented a basic database schema, and have stored our files using the Filestream functionality SQL has.



In the future, we have a collection of tasks we will accomplish to improve our implementation. First, we will utilize the website to display the images that have been uploaded, as well as the ranking they have been given, completing basic implementation of the planned photo browser features. Second, we will implement basic login/logout functionality, both to restrict uploading and downloading of files and to associate files and their ratings with users. Third, we will allow users to delete the files they have uploaded, completing basic implementation of user file management. As that is occurring, we will also be developing the image assessment algorithm. For one, we will implement training, and utilize the already-provided photographs to differentiate ‘good’ photographs from ‘bad’ photographs, as well as a 1.0-10.0 scale the image assessment algorithm will utilize to grade images. Once the algorithm has been properly trained, we will run it on uploaded images as they are uploaded, then display the ranking of the image to their associated user, allowing for instant feedback.

Overall, while there is much to be done, we have implemented the basics of the project, as required by Ed McCain - *“The basic expectation is to be able to take a large group of images and process them using software in order to rank them in terms of quality, both technical and aesthetic.”* As a result, our efforts going forward are less on core requirements and more on smoother and better implementation of the project.