Phamily Fotos Project Report

# Motivation

This project has been triggered by the notion that by 2020 (if not before) the general public might become disenchanted with the rights Social Media will have over their own private material.

## Project Scope

By referring to private material, this project will deal with images in the form of private family photographs (and videos as a natural evolution) and will enable people to upload images for other family members to view, as well as enabling these other family members to upload and share their photos too.

## Area of Contribution

This application will exercise no rights over the images that are posted and shared by families with each other and will allow all users to retain their rights to these images.

# State of the Art Review

## Privacy Concerns

## Security Concerns

## Review of current Social Media & attitudes to rights over content

## Current trends/opinions expressed online re privacy

## Potential/challenges raised by the use of DRM (Digital Rights Management)

## More?? Survey of users opinions of this applications aims vs current similar applications

# User Interface Design

## Design Concept

The main motivation behind the design of the user interfaces was to at first make the experience welcoming, friendly and easy to use. And by virtue of the application’s responsive design using HTML5, CSS3, and JavaScript, the application will be accessible on all devices with an internet connection, ranging from Desktops, to Laptops and down to Tablets and Smart Phones

Figure 1. Welcome / Log in Screen



As can be seen in the Welcome screen on the left, it’s important to give the user a sense of confidence in the applications ability to live up to its promise of preserving their privacy on one hand, whilst enabling them to share their images with the close friends and family of their own choosing. This will be achieved by having a well-structured log in and join process, where people can only log in with their own username and password, but equally, when it comes to sharing their content with other people, this will be strictly controlled and will only be achieved by mutual consent of the users concerned.

Figure Mobile Welcome Screen

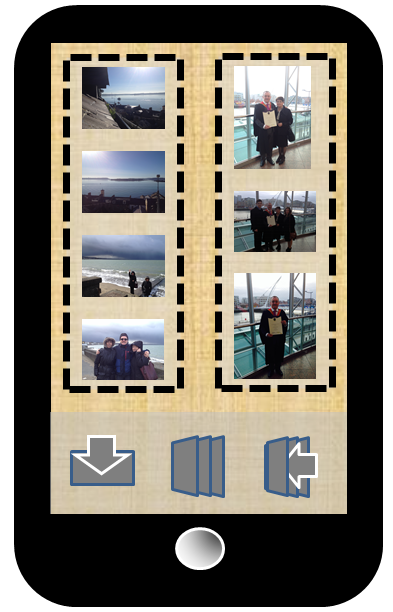


Figure Mobile Gallery Screen

Once logged into the application, the user will be presented with either a gallery of all their own photos or a collection of the various albums they’ve created from their photos. A side or bottom menu will give the user the options to upload new photos, to create a new album and then to add photos to this new album.



Figure Mobile Phamily Photos Screen



Figure Browse Albums Screen. Plus Side Menus

Having created one or more albums, the user will be able to scroll through thumbnails of the images and then if they select the album for viewing, they’ll be able to see a larger representation of these photos which they can flick through.

Figure View Phamily Fotos Screen



The scrolling of the thumbnail images, as well as the layout of the albums in a Photo Strip and the subsequent display of and flicking through the **“Phamily Fotos”** will all be achieved using HTML5 and CSS3 effects in conjunction with JavaScript and Ajax. The layout of the various screens will depend on the size of screen available, this will be controlled in CSS the @mediaquery feature which establishes the size of the screen available before displaying the webpage and applies the CSS rules appropriate to the size found. Where possible images and icons will be designed and drawn in Inkscape using SVG (Scalable Vector Graphics) which is the W3Schools recommended technology as an Open Source alternative to Flash.

# Architecture

## Application Architecture

The chosen application architecture will be the standard MVC (Model View Controller) using HTML5, CSS3 with a combination of JavaScript and Ajax at the Front End (the View). The back end or Model will consist of a MySQL database, finally PHP will be used to communicate between the Front and Back Ends acting as the Controller.

Client Side



Server Side

This, to me, is the most obvious approach to take, as, to facilitate sharing, there will be a basic requirement for storage of the images centrally in a secure place. Following on from this, some kind of Graphical User Interface will be required for the users to interact with. Having dealt with these two essential requirements, the application will need some means by which the front and back ends can communicate with each other. By the same token, keeping the storage and the controller functionality on the server side, being accessed by web service API’s, means the client side can be kept light. This strategy will be ideal for users accessing the application on mobile devices where data storage and processing power is at a premium.

## Security

## Toolkits and Frameworks

### Front-end

The HTML5, CSS3, JavaScript and PHP were all coded using Notepad++. The individual web pages were tested on the local machine on all the major HTML5 compatible browsers (Chrome, Firefox, Safari and IE) and extensive use was made of the development tools available in each browser to enable the CSS to be refined to take account of each Browsers particular quirks and qualities.

### Back-end

The Server-Side PHP scripting was also written using Notepad++ and the MySQL database was developed using the Open Source phpMyAdmin. Extensive use was made of PHP Include files. These Include files encapsulate most of the repetitive functionality of the Server Side scripting such as database connectivity and MySQL queries and enable other files/web pages to use and re-use this code, making for more efficient back-end functionality.

## Data Transfer Strategies

Data transfer will be over either a wireless or wired connection, and as such will be over an http connection, however for security and privacy purposes this will have to be an https, offering greater security to the users.

Given that the vast majority of uploads to the site will be images, with only a little text for captions or album titles, it will be important for users to have a fast, reliable data connection. It might be worth considering more than one Data Transfer Strategy depending on what kind of connection is available to the user, with a final fall back offering local storage (in a situation where the quality of connection is insufficient) until such time as the user has access to a better connection.