Initial Plan

Location Based Photo Sharing Application

Student name: Christopher Paterson

Student number: C1229143

Supervisor: Dr. Matthew J W Morgan

Moderator: Dr. Kirill Sidorov

Module: CM3203 – One Semester Individual Project

Project Description

The project is an iOS application that allows user to share photos linked to their current location. They will be able to tag, rate and comment on these as well as view previously shared photos at that location. I will include a web application for administration purposes such as removing inappropriate photos or comments. I will make use of Flickr's API to supplement the photos as there will not be many taken from the app initially.

As a side effect of collecting information such as tags, ratings, locations and comments, it would then be possible to create recommendations for near places of interest which would be a feature I would look to implement if I had enough time remaining.

Example: Imagine you're at Cardiff castle for a reenactment. You take photos to share with others who will be in the area. You could leave a comment to explain it or say how much you enjoyed the day. Similarly, you could see photos from past events, such as a ghost tour at the castle and rate/comment on them.

Ethics

I intend to conduct user testing at various points in my project. For this I will be adhering to 'Ethics For Conducting A Usability Test' <http://facweb.cs.depaul.edu/cmiller/eval/ethics.html> . I also need to ensure that users information is kept sensibly and securely as I will be handling passwords and possibly email addresses.

Project Aims and Objectives

Mobile

- Ability to share, comment and tag users own photos.
- Ability to comment and rate other users photos.
- Ability to view photos taken near user's current location.
- Intuitive interface and usage for users.
- Ability to save photo to upload later for times with no network connectivity.
- Ability to manage shared photos while not at location.

Web

- Admin ability to remove inappropriate photos and comments.

General

- Include photos from Flickr to expand the photos base.

Extension features

- Recommendations for near places of interest.
- Tour - Users are able to add a collection of photos to create a tour. When followed, users will be guided from photo to photo on the map.

Work Plan

I intend to develop under the agile methodology, working in 2 week sprints. At the end of each sprint I will have a meeting with my supervisor where I will show and discuss my progress. I will also schedule smaller mid-sprint meetings if needed to allow for earlier feedback which may influence the rest of the sprint. I will be testing both code and the user experience throughout development.

Sprint 1

- **Research similar applications/services - 1 day:** By looking at similar applications, I may find some information that I can use in my project. I feel this will give me a solid starting point.
- **Set up project, environment and source control - ½ day:** I have done all of this before and I am familiar with Git so it should not take too long.
- **Get API keys and learn to use the APIs - ½ day:** As I will be using Parse and, later on, Flickr, I will need to register my application and acquire API keys. I have used Parse in the past on Android so I will need a little time to look at the iOS usage. Flickr is new to me so I will need to spend some time looking at how to use their API.
- **Refresh knowledge on Swift and iOS development - 4 days:** It has been a number of months since I have developed for an Apple platform so I will need to refresh my knowledge.
- **Design potential user flows - 1 day:** Using what I have found from similar applications/services I will begin my initial design of the user interface and experience.
- **Implement user accounts - 3 days:** I have not yet decided if I want to add the ability for users to log in via their social media accounts such as twitter or if I will have accounts just for this application.

Sprint 2

- **User is able to take a photo, write a description and share it - 2 days:** I have made a test application that allows the user to take a photo and then stores the photo and the location in a Parse database. Adding the ability to comment on this should not be difficult.
- **View photos taken around the user's location - 3 days:** The easiest way to do this would be to fetch photos within a radius when a user refreshes. Another way would be to send off pings every 30 or so seconds. The second may create a smoother user experience but it may use up a very large amount of data. Due to this I intend to use the first method.
- **Comments on other people's photos - 5 days:** I have not yet decided if comments should be linear or nested, i.e. can users comment on comments or just photos? On top of this, I do not really know how I will achieve commenting on photos from Flickr. My current thought is to store the photo ID along with the comment.

Milestone 1 reached: Minimum viable product.

Sprint 3

- **Tags for own photos - 4 days:** I have not yet decided if I want user defined tags or pre-defined tags. If I go with user defined then it may be harder to use these for suggestion reasons but if I use predefined, it's likely I will not cover all possible tags.
- **Ratings for other people's photos - 2 days:** At this point I intend it to be similar to a Twitter 'heart', and not a 1 - 5 style scale.
- **Make use of Flickr to allow for a larger photo base - 4 days:** Most of the time allocated here will be allowing features such as commenting and rating on the Flickr data set.

Sprint 4

- Web implementation:
    - **Allow admins to remove photos and comments and potentially ban users - 7 days:** I will be using Laravel 4.X to create this as I have the most experience with it. It will likely take me a day to get up to speed as I have not used Laravel recently and the rest of the time will go to implementation.
- Mobile:
    - **Report feature - 1 day:** I imagine this to be an option on a long press menu so I do not think it will take a long time to implement.
    - **Own photo management - 2 days:** The hardest part of this will be cascading the deletion of associated comments when an photo is deleted. However, since they will not take up too much space, I could leave the comments in the database for now.

Milestone 2 reached: Basic features completed.

Sprint 5

- **Buffer:** It's possible that I may have underestimated how long some tasks may take so I will set this time aside to accommodate for these.
- **Extension features:** If I have completed all previous tasks then I will use this time to work on implementing extension features such as recommendations.

Sprint 6

- **Report review/refinement - 8 days.**
- **Project cleanup - 2 days:** I will be refactoring throughout but I may wish to have a cleanup once everything has been implemented and no new features will be added.