

myFlickr: An Alternative Flickr Mobile App

Dexter Legaspi (dlegaspi@bu.edu)
BU MET MSSD

- Overview
- Related Work
 - Essential
 - Desirable
 - Optional
- Requirement Analysis and Testing
- Design and Implementation
 - Basic Architecture
 - UI Design and Implementation
 - Third-Party APIs
 - Data Design and Implementation
- Project Structure
- Timeline
- Future Work (Optional)
- Lessons Learned
 - Iteration 0
- References

Overview

I have been an avid photographer since I was in high school when my mother purchased my first Pentax SLR. The advent of digital technology lending itself to the advancement of photography has further helped on improving my craft since I also work in the computer field. There are a million applications (desktop and mobile alike) for photography and I have benefitted from most of them, and I would like to pay it forward with a contribution of my own for anyone who wants to improve in their craft.

Flickr is one of the first established and popular photography websites, and have been instrumental with my learning finding images of other people around the world. Flickr is a photography website, but it is more geared towards the social aspects of it: followers, groups and sharing of images. These are important, but it is equally important to look at one's own catalog without the noise, and see how he/she has improved. This is the focus the app: hence the name “myFlickr”: to focus on **my** images and have a fresh perspective on them. It would have the usual images viewer but the browser would be more optimized for quick browsing with the metadata (which the official app doesn't do today); there will be minimal features geared towards the social/sharing and more tools to filter and inspect your own images—e.g., there will be search and filtering based on the metadata of the images

Related Work

Essential

The essential features of the application will revolve around being able to authenticate and retrieve the images of the authenticated user. The authentication requires interfacing with a relatively antiquated OAuth 1.0a authentication scheme of the Flickr API. In addition, the core functionality, of course, is to be able to show all the images of the user, either in individual mode or in a grid. When viewing the individual image, the user should be able to optionally see the overlay of the image metadata.

Desirable

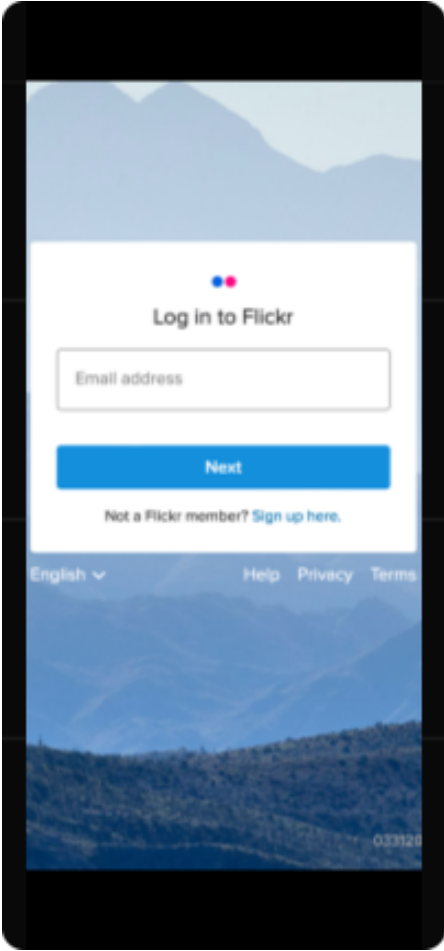
With regards to the image viewing and presentation, it would be desirable to have as much options as possible: it would be nice to have a split view of the images where one can have a film strip view of the images on the bottom half and then the selected image on the top and also be able to optionally group the images based on the metadata.

Optional

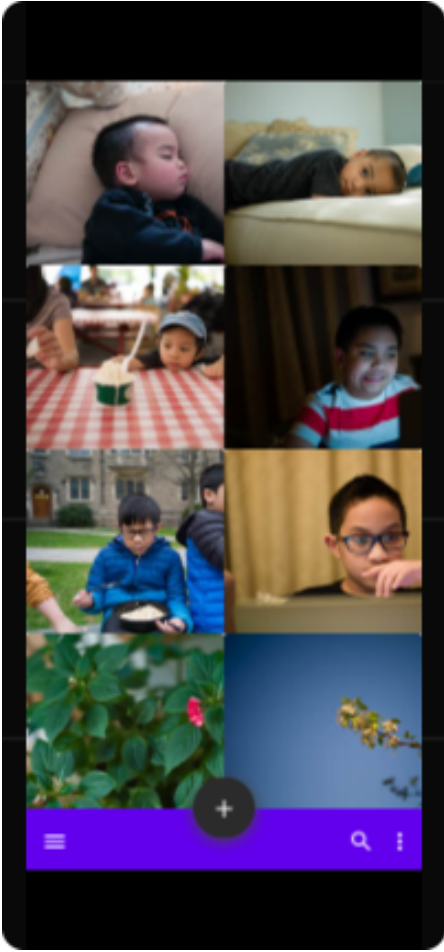
It would be nice to have some push notifications for any likes or increased views but it will be very low on the totem pole of features to be implemented and will only be there time permitting.

Requirement Analysis and Testing


| Title(Essential/Desirable/Optional) | Authentication (Essential) |
|-------------------------------------|--|
| Description | As a user, I will be able to authenticate using my existing Flickr credentials |

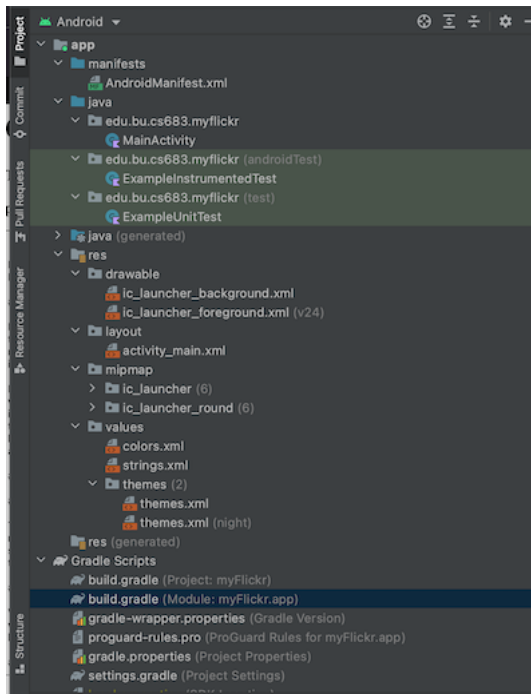
| Title(Essential/Desirable/Optional) | Authentication (Essential) |
|--|---|
| <div>Mockups</div> <div>Acceptance Tests</div> <div>Test Results</div> <div>Status</div> | <div>  </div> <div>User will be able to log in using Flickr credentials and have the opportunity to authorize the app to access his/her account and images</div> <div>TBD</div> <div>Iteration 1: Integrated with Flickr OAuth API</div> |

| Title(Essential/Desirable/Optional) | View Images as Grid (Essential) |
|-------------------------------------|---|
| Description | As a user, I should be able to have an option to view all my images in my account as a grid |

| Title(Essential/Desirable/Optional) | View Images as Grid (Essential) |
|-------------------------------------|---|
| Mockups |  |
| Acceptance Tests | User can display all of his/her images in a grid |
| Test Results | TBD |
| Status | |

| Title(Essential/Desirable/Optional) | View Images as Individual (Essential) |
|-------------------------------------|--|
| Description | As a user, I should be able to have an option to view all my images in my account individually with the option of having the metadata as overlay |

| Title(Essential/Desirable/Optional) | View Images as Individual (Essential) |
|-------------------------------------|--|
| Mockups |  |
| Acceptance Tests | |
| Test Results | |
| Status | |



Timeline

| Iteration | Application Requirements (Essential/Desirable/Optional) | Android Components and Features |
|-----------|---|--|
| 1 | Authentication and Basic Image Browsing | Using Network connectivity and WebView for OAuth Handshake, using Flickr REST APIs |
| 2 | | |
| 3 | | |
| 4 | | |
| 5 | | |

Future Work (Optional)

(This section can describe possible future works. Particularly the requirements you planned but didn't get time to implement, and possible Android components or features to implement them. This section is optional, and you can include this section in the final iteration if you want.)

##Project Demo Links *(For on campus students, we will have project presentations in class. For online students, you are required to submit a video of your project presentation which includes a demo of your app. You can use Kaltura to make the video and then submit it on blackboard. Please check the following link for the details of using Kaltura to make and submit videos on blackboard. You can also use other video tools and upload your video to youtube if you like: https://onlinecampus.bu.edu/bbcswebdav/courses/00cwr_odeelements/metcs/cs_Kaltura.htm)*

Lessons Learned

Iteration 0

- OAuth 1.0a integration is difficult nowadays because most implementations are in 2.x. Unfortunately, Flickr API is using 1.x
- I learned that Since Android 3, apps cannot execute network-related tasks in the UI thread, which prompted me to having to learn about **AsyncTask** (which apparently has been recently deprecated) to geth the OAuth integration to work.

References

- [Flickr API Documentation](#)
- [OAuth 1.0a](#)