**Final Project Report For PhotoAlbumsApp**

Yukui Ye

[yye@syr.edu](mailto:yye@syr.edu) or [mollysmile.ye@gmail.com](mailto:mollysmile.ye@gmail.com)

**Abstract—I play around images that be dynamically fetched from flickr.com, then create an iphone app to show every images in one flickr user account. Second, I fetch all the geological images in flickrs.com , which has detail information about the photographer and his photos, and then store all the images into local core database by using xcode core data framework. Third, I will transform iphone storyboard into ipad storyboard which will disappear an map rather than table view. I am trying to create two map, one is showing all the photographers geological location on the earth map, then when we touch one pin, it goes to another earth map where will display all the photos that has been token by the photographer, and when we click on the pin, it will show out next page which will display the detail of the image. Finally, I will summarize all the challenges and possibilities for future work.**

**I. Introduction**

In recent years, people always love to post their images on internet, Flickr.com provide an excellent plate to enable people sharing photos around all over world. What if you want to store all the images in any one of the user account on flickr.com ? Downloading images one by one will certainly be a tedious job. I did some research and come up one solution to solve this kinds of problem. As described in abstract, My first goal is to use iphone simulator to dynamically browse all images from one flickr user account in table view framework, and It successfully worked. Then I surprisingly found out that flickr enable people to post images which has geological feature, which means it can shows the images location, then I dig more deeper based on previous project. Each photo record its owner, latitude, longitude and some other information, I create two entity in core database to store all the image data fetched from flickr.com. one is photo, another called photographer. Their relationship is many-to-one, which means one photographer can have many photos. The problem comes to how can I record the geological information so that It can display in one map. There is one method called core location, and you need to use MapKit framework.

A.Xcode

I implement the project by using Xcode. Xcode has many useful framework, such as CoreData.framework, MapKit.framework, CoreGraphic.framework.

B.Iphone/Ipad simulator.

I develop the user interface and interaction by using iphone/Ipad simulator. The simulator is used to capture the image, and communicate with Xcode program, and display the result on the user interface.

**II. System Implementation**

**Part.1. Create an applicant that represents a list of photos fetched from Flickr.com**

Concept: Table view, Scroll view, Image view, Flickr’s API Key.

You need to apply for Flickr’s API Key in order to querying photos. Besides, If you want to get all the photos from one user account, all you need is the Flickr user account name, then you need to use “http://idgettr.com/ “ to get the user ID. More code information, please check FlickrFetch folder. The fetched Flickr results is an NSArray of NSDictionary objects.

Start with Single View Application template, Create a UITableViewController, which will display any list of photos title, and owner’s name as table view cell’s subtitle. When the user chooses a photo from any list, display its image inside a scroll view which will allow user to pan and zoom. I obtain the image URL for a Flickr photo’s image by using FlickrFetcher’s urlForPhoto:formate: () method.

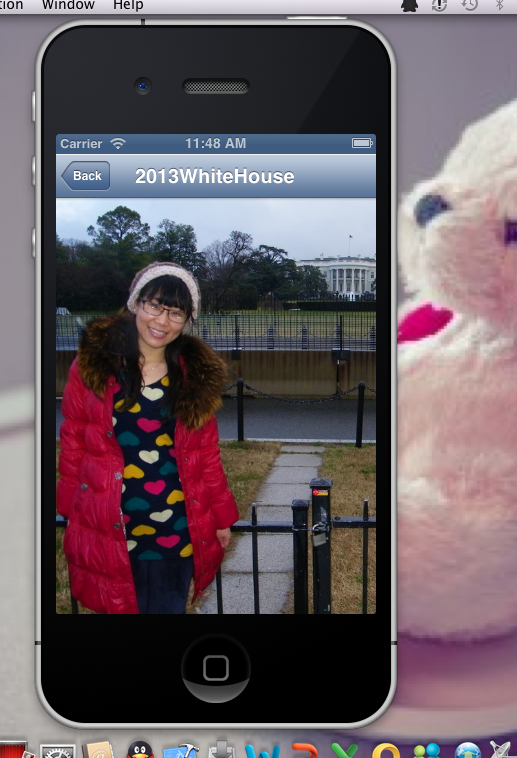
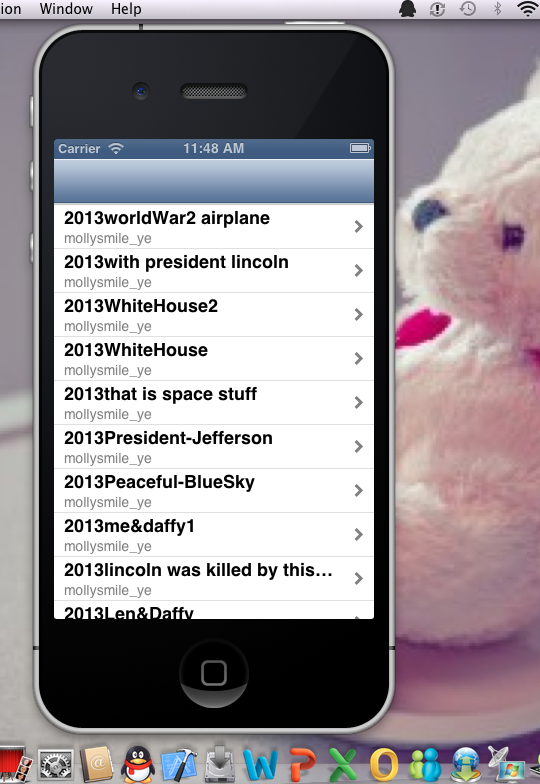
Turning a URL on the internet into a UIImage is easy, Just create an NSData with the contents of that URL( NSData datawithContentsOfURL: the URL), then create a UIImage using that NSData ( UIImage imageWithData: imageData). It also require some calculation involving the UIScrollView’s bounds and the size of the photo. Besides, From one tableView goes into another image View you need prepareForSegue:Sender: which will link you to enter from one view to another view.

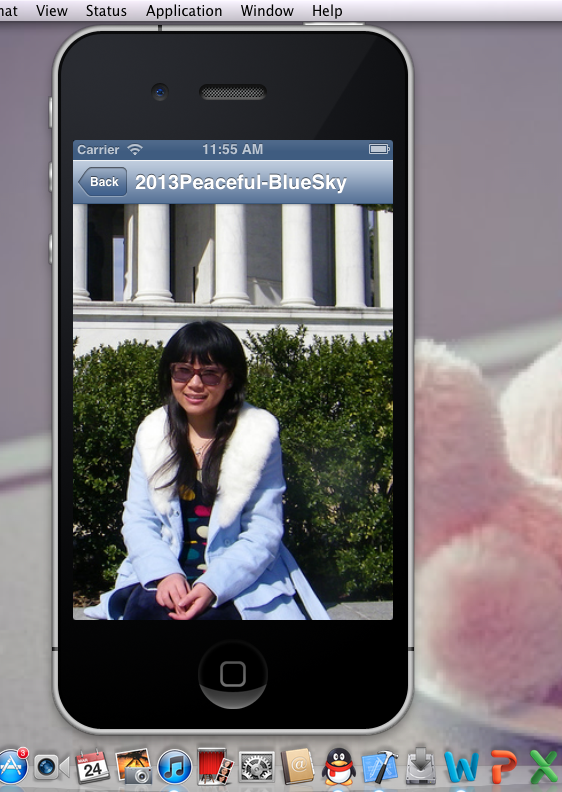
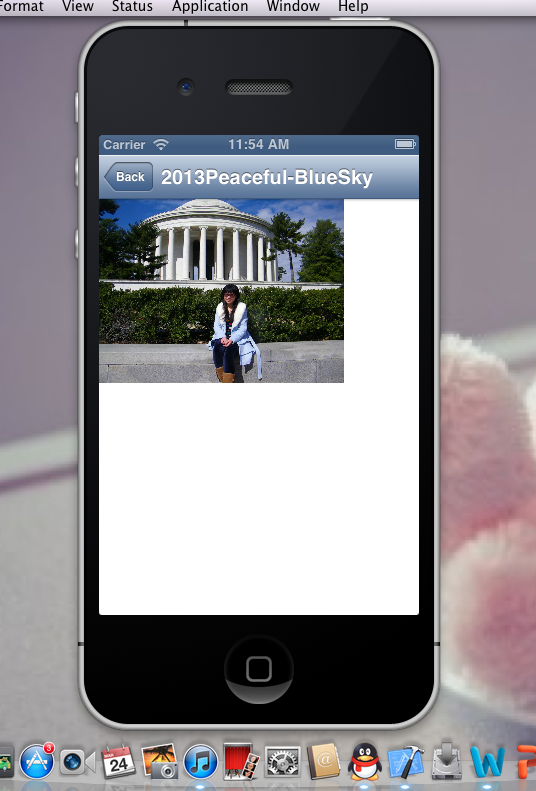
How things work out ?

I use my Flickr account : mollysmile\_ye as an example to show the result.

And It enable zooming in and zooming out on one photo.

First , It gets a list of photos from my user account, then display it in tableView as shown in first picture below, Second, when click one cell, it links to image view which display the detail of that image.

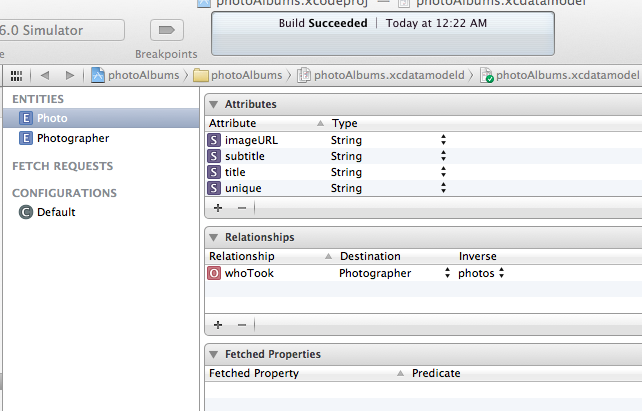




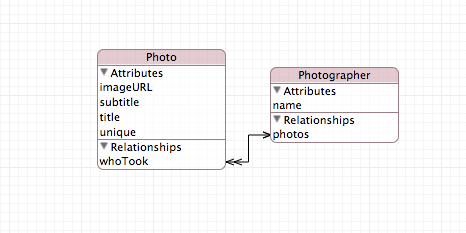
**Part.2. Store Data From Internet to Local CoreDataBase**.

Concept: CoreData, NSManagedObjectContext, UIManagedDocument. NSFetchRequest.

First create a visual mapping(using Xcode tool) between database and objects. Create and query for objects using object-oriented API, Access the “columns in the database table” using @property on these objects.

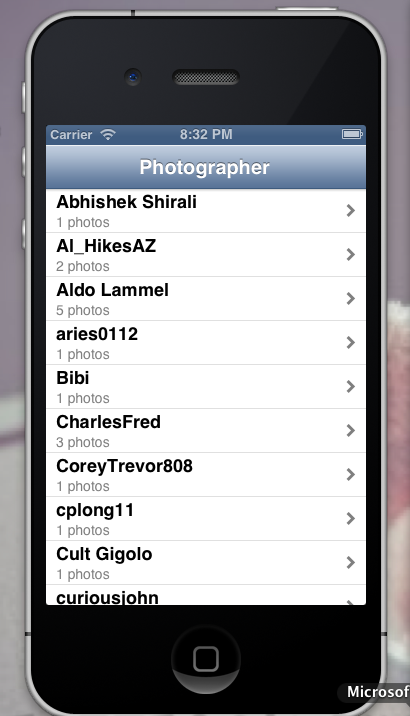


And set up their relationship



After you setup everything, you need to create a UIManagedDocument and ask for NSManagedObjectContext , which is the hub around which all core data activity turns. After you store the data in core database, you need to create category on NSManagedObject to get the data from database.

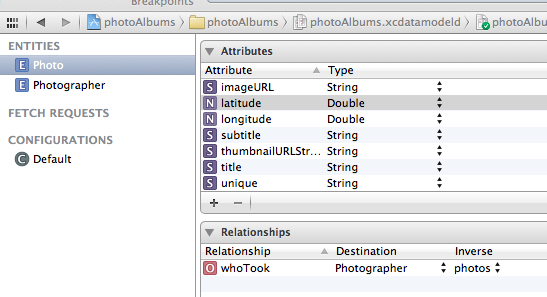
Based on previous step, I fetch the latest geological photos in Flickr.com rather than photos in only one user account. As we can see the title of table view is the photographer’s name and the cell’s subtitle list the number of photos under this photographer, when you click on one photographer it will show out the photos in the second table view. And these images have been stored in local core database.



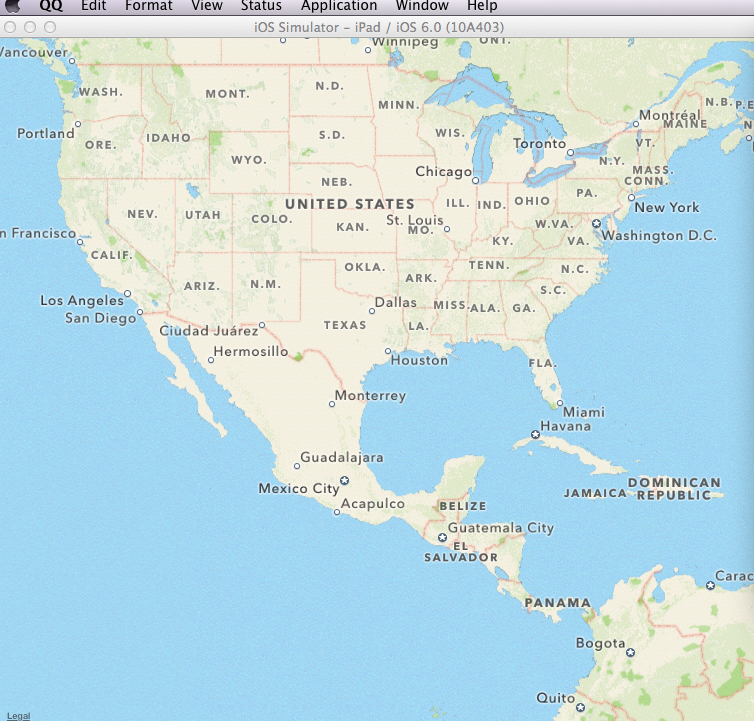
**Part.3 Display Flickr photos on a map**

Concept: MapKit, Core Location, MKMapView, Annotation, MKAnnotationView

First, Based on step2, Change the plate from iphone to universal, then create new ipad storyboard. I will implement the map application on ipad rather than on iphone simulator. Then I add three more attributes of photo entity in the core database, which are latitude, longitude and thumbnailURLString. Then regenerate the NSManagedObject. This will enable you to store the location of the photo, when fetching from the FlickrFetch method.

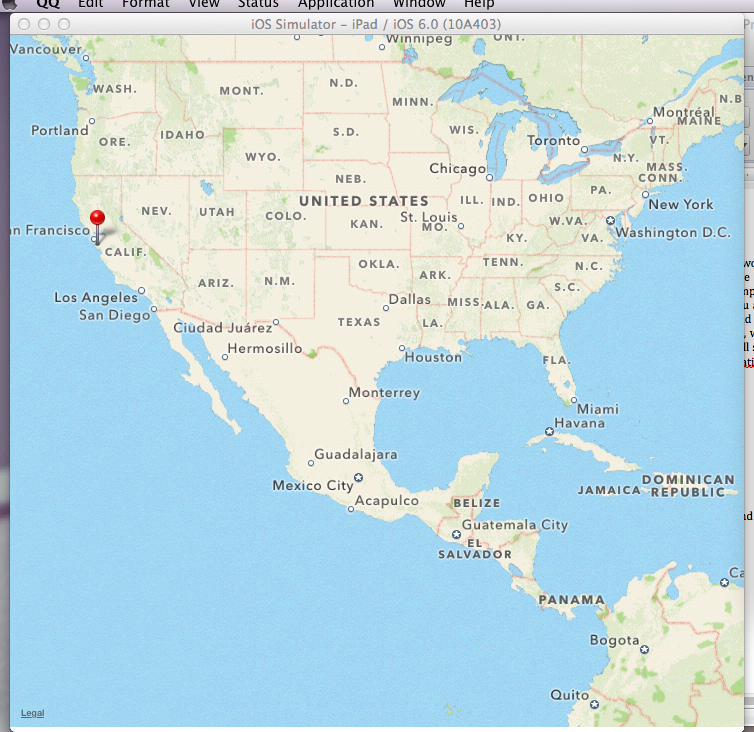


Second, load in two map View, First one will display photographer, it should show in the format of the red pin on the map. The pin is the so-called annotation. Each annotation is simply a coordinate, a title and a subtitle, and it can have a callout, which enable you add left and right accessory view, In the program, the left side of the callout should be a UIImageView, right side is a UIButtonTypeDetailDisclosure. (the blue arrow), when you click on the left accessory view which in here mean the blue arrow, it will segue to second view. After we recreate the database, we need to create MKAnnotation category, which will connect data from local database onto pin of the map.

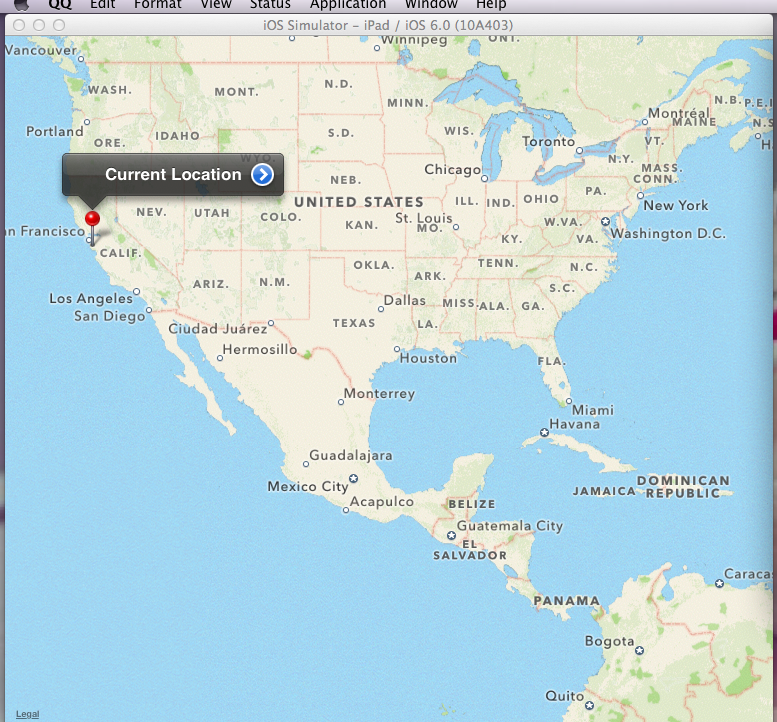


Problem I met: The map app is not working properly. The pin which represents one photographer, It did not show out on the map. I found the problem, It is because when I add new attributes latitude, longitude and thumbnailURLString, and regenerate it again, the data fail to be fetched out from Flickr.com. That means, I donot have any photo in the database. Since each pin represents each photographer, then photographer link to photos, I failed to get the data, thus, no pin show out on the map.

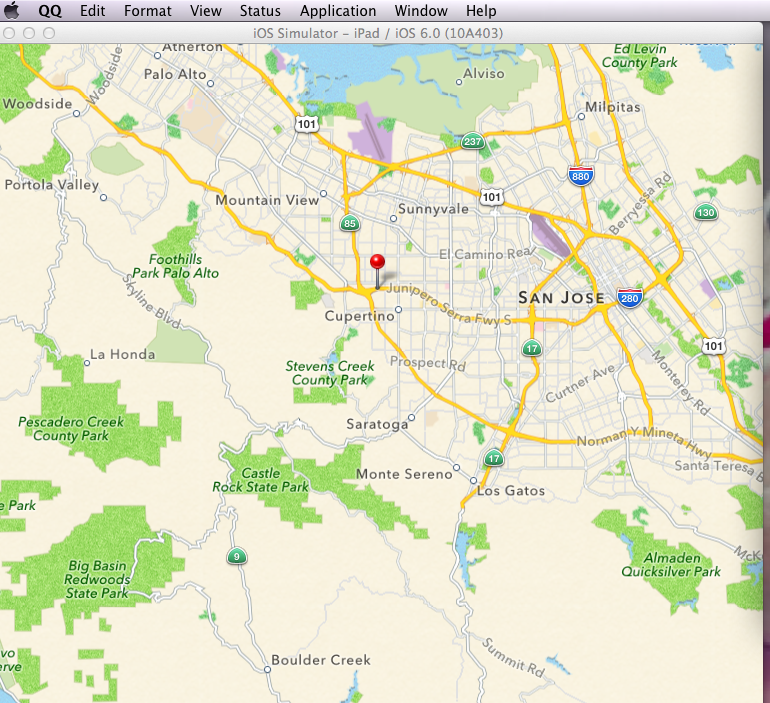
The map should work like below, which I take the current location as pin example.



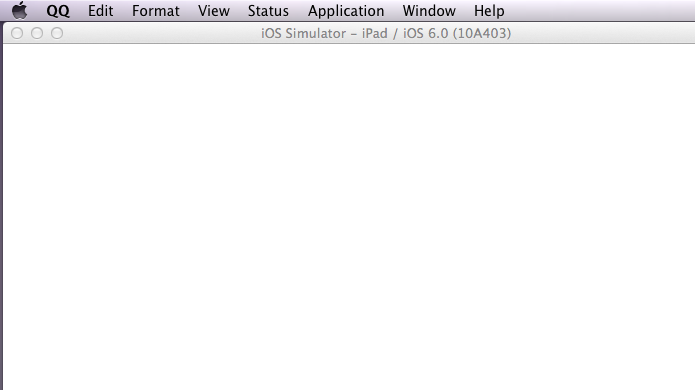
There should be a lot of pin based on the number of photographer, and when I click on the pin it will show the detail information of the photo under one photographer, and when I click the left blue arrow thing, it will lead to the second map to display the location of each photo, where the pin will represents photo, then click on the pin it will lead us to the third view which will display the detail of the image.



click on the blue arrow , it lead to second map, It will atomatically zooming into the specifical location rather than the rough location. Because I wrote the code for relocation which enable automatically zooming in.



then when I click on the pin, it links to the third view which should display the image, but because I did not successfully store the photo in the database, thus it has no photo show out., just plain image view.



**III. Conclusion and Future Work.**

Confusion: In the part 2, I can successfully fetch out the data from Flickr.com and store it into the core database, but I do not know why when I add three new attributes into the photo entity, and regenerate the NSManagedObjectContext, It failed to store the data. That is the main problem I met so far.

I will continue my future work to check it. I played around the ipad and iphone creation one image activity, It is kinds of fun. And I will continue my work to create new app.