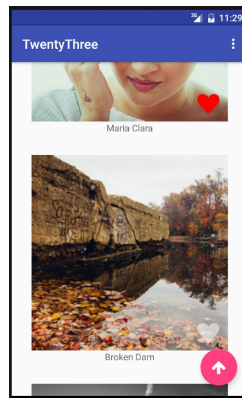


Technical Breakdown

Functionality:

This app displays a list of interesting photos using the Flickr API. It keeps track of favorite photos when the user taps or un-taps the heart button. A favorite photo will display a red heart. The list of photos can be refreshed using the refresh menu button.



Technical Details:

This app is a single Activity Android application. It has an ActionBar with a menu. The menu only has one item - "Refresh". The Refresh button clears the cache and downloads the current set of interesting images.

The photos are displayed in a ListView using an PhotoAdapter. It uses a Flickr static function for converting each photo data into a url. The urls are passed to the Picasso library for downloading the images. Other Picasso functionality used are resizing and center-cropping.

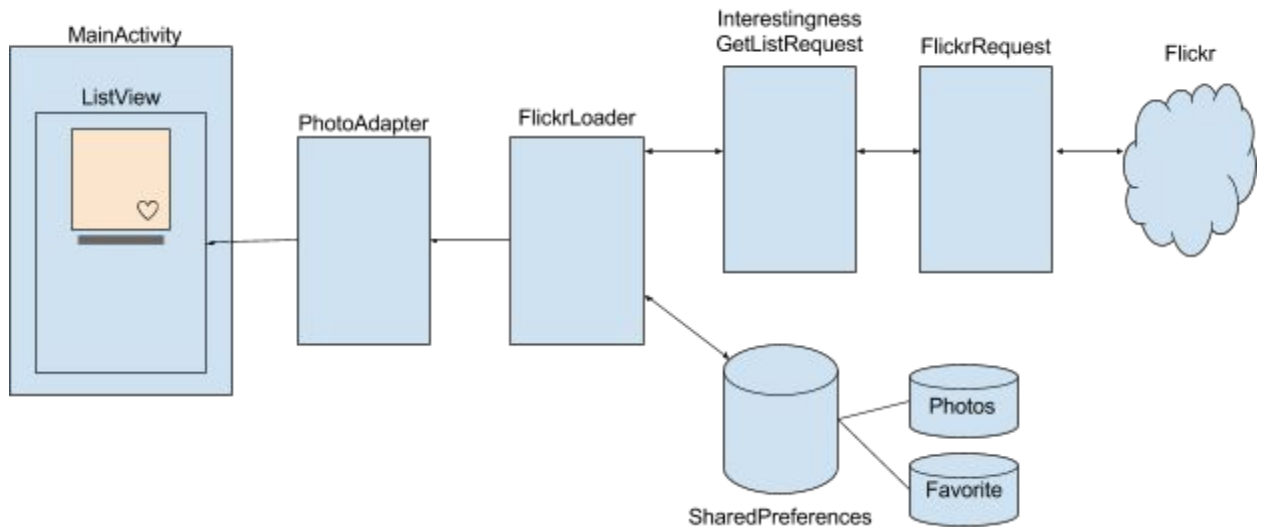
The FlickrLoader supplies the PhotoAdapter with data. It serves data from cache or fetches the data using the FlickrRequest object. The API fetched data is stored into cache. When fetching data using the FlickrRequest object, data is fetched in batches of 100. The Loader will resume loading data on app startup if the cached data is not complete.

An Interestingness FlickrRequest object is used for fetching the list of photos. It implements the parse function of the FlickrRequest to convert the JSON string into a list of Photo objects. It also has optional API parameters that can be set. The actual fetching of data is done at the base class FlickrRequest.

The FlickrRequest is a base class for any API requests being made to Flickr. It's only function is to make the requests with the supplied parameters.

The Utils class is a set static functions for performing DB like operations. The two data sets it is maintaining are Favorites and Photos.

The Flickr class contain the models for Photo and Photos. It also has a couple static functions for converting Photo objects into urls.



Libraries:

I've decided to use the Picasso library for downloading images. It automatically handles caching, ImageView assignment and handles cases where Activities become unavailable.

Next steps:

- Consideration when there is no network.
- Fetch images based on screen size.
- Use a database instead of SharedPreferences for cache.
- Cache the urls to limit use of the url conversion function.