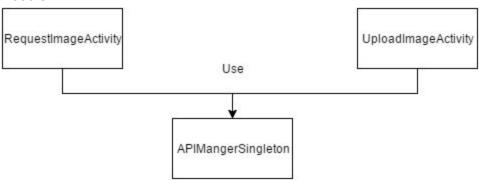
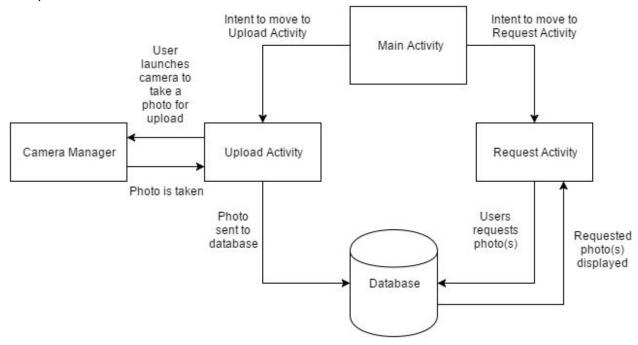
Module:



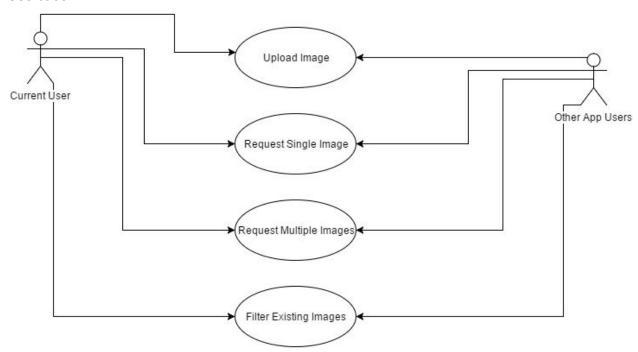
Uses diagram showing the relationship between both requesting and uploading an image and the APIManager.

The APIManager handles all of the requests, so whenever the request or upload activity wishes to send a request to the backend, they have to use the APIManager.

Component-Connector:



Use case:



At any given moment, only one actor will be interacting with a given instance of the app. However, other users may be interacting with the app as well, so we have created an actor to represent the other users of the app.

Class:

MainActivity

- #onCreate(Bundle):void +launchRequestImageActivity(View):void +launchUploadImageActivity(View):void

RequestImageActivity

#onCreate(Bundle):void +requestImage(View):void

UploadImageActivity

#onCreate(Bundle):void
#onActivityResult(int, int, Intent):void
+uploadImage(View):void
+kickOffUpload(String, String):void
+takeImage(View):void
-dispatchTakePictureEvent():void
-createImageFile():File

BitmapAsyncTask

#doInBackground():Bitmap #onPostExecute(Bitmap):void

<<Interface>> APIInterface

- + URL:String
- +getAllImages():Call<JSONArray> +getImage(String):Call<JsonObject> +postImage(JSONObject):Call<JsonObject>

APIManagerSingleton

- -instance:APIManagerSingleton -retrofit:Retrofit
- -api:APIInterface
- -APIMangagerSingleton():APIManagerSingleton +getInstance():APIManagerSingleton +getImages(String, LinearLayout):void

- +postImage():void

ServerData

- name:String <<get/set>> description:String<<get/set>> date: long<<get/set>> data: String<<get/set>>

ServerImageViewFactory

- -parent:LinearLayout
- -imageView:ImageView -txtID:TextView -txtName:TextView
- -txtDesc:TextView -txtDate:TextView
- +ServerImageViewFactory(Context, byte[]):ServerImageViewFactory +addTitle(String):ServerImageViewFactory +addDescription(String):ServerImageViewFactory +addID(int):ServerImageViewFactory

- +addDate(long):ServerImageViewFactory +build():LinearLayout

ImageViewWithListener

-listener:OnSetImageBitmapListener

#setImageBitmap(Bitmap):void

+setOnImageBitmapListener(OnSetImageBitmapListener):void

<</nterface>>
OnSetImageBitmapListener

-updateData:void