Barkha.Java001
CS443
01774410

CS443 CAM-MAPS

Final Project Report

Project Statement

The project aims at creating an app beneficial for people who love to keep memories of their travel. The app can click images and drop them at their locations! You can view a map showing all images, where they were clicked and description of the images. The app rewards the user with achievements; this encourages the user to take more pictures and visit more places. Apple Photos lets the user view his photos on a map as well, but there are no rewards provided.

Application Design

Activity: Below is an overview of how the Activities work.

View Your Achievement

The activity first removes all files at the path DCIM/Camera.

Depending on the number of files at the path it rewards the photographer reward

It also removes the title from the images. It checks how many pictures are titled as “Library”

and rewards the user when there are more than three pictures of a library.

View Map

This Activity lets you view all your images on the map.

It is implemented using onMapReady method.

All image details are retrieved using MediaStore. (Eg. Location, Title)

All markers at the same location are clustered using the classes ClusterRenderer and Clustermarker

Barkha.Java001
CS443
01774410

View Map

All gestures have been enabled for the map. This enables the zoom in and out buttons having + and – Signs.

Pinch gesture has also been enabled for the map

Click Image

Activity Name: SaveImage: This activity lets you capture images.

The following page has the image just captured placed in an imageView.

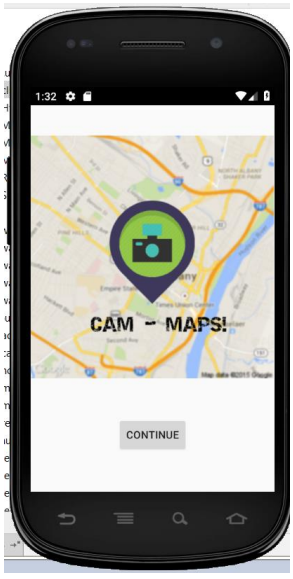
Intents used: MediaStore.ACTION_IMAGE_CAPTURE, ACCESS_COARSE_LOCATION

This page has a dropdown which lets you select the type of the image.

This type is then saved in the image under “Description” and “Title” in MediaStore.

The location where the image was captured is also stored using MediaStore.

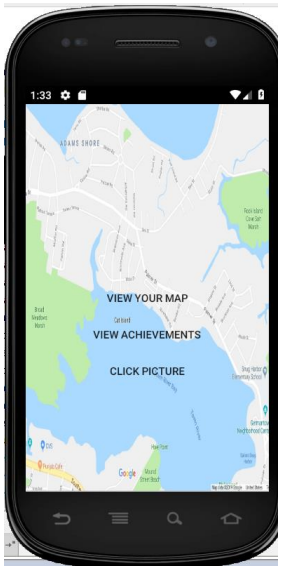
3. Application Implementation and Evaluation



Classes Implemented:

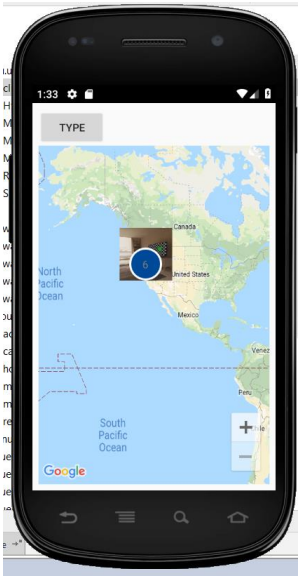
Main Activity: This activity only displays the app logo with a continue button to move forward to the menu.

Barkha.Java001
CS443
01774410



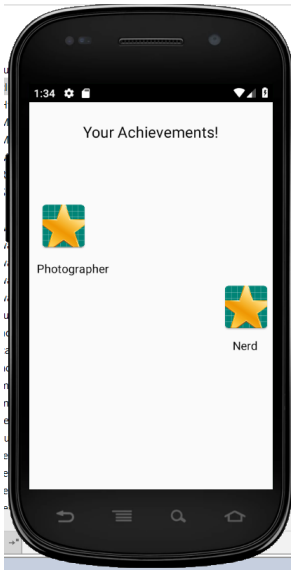
Menu: This Activity Displays a Menu, which navigates you to the other activities.

- View your map - > This navigates you to the Map Activity
- View Achievements -> This navigates you to the rewards activity
- Click Picture->This navigates you to the Save Image activity



Maps: This Activity displays all images taken at their location on map.

- If there are multiple images on one location, a blue bubble displays the number of pictures at that location.
- You can also view the multiple images by clicking on the image. Once clicked it will show you the next image.
- The map can be zoomed in and out using the buttons or the pinch gesture.
- The type button enables the satellite view of the map.
- Clicking on the image will also show you the title of the image.

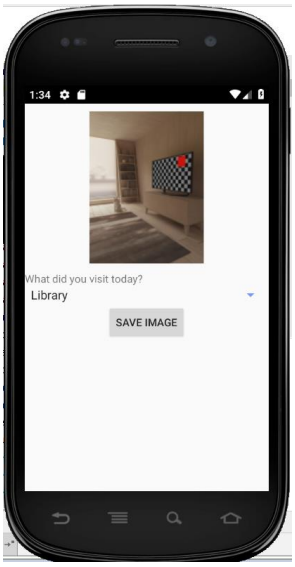


Rewards: This Activity Displays the rewards achieved by the user.

Two rewards are incorporated for now in the app.

- Photographer – If more than two pictures are taken, the user receives a photographer reward.
- Nerd – If more than three libraries are visited, the user receives a nerd reward.

Barkha.Java001
CS443
01774410



Save Image: This Activity Displays the image just captured and lets you add where the picture was taken. It also lets you save the image. Once saved it will navigate back to the Menu Activity.

Clustermarker – This class has been implemented to display the number of markers at a location. Cluster marker will make a cluster of all the markers at a location.

MyClusterItemRenderer – This class has been implemented to display the number of markers at a location. It creates a bubble to display the number.

FragmentActivity – This class has been implemented to allow Fragment Activity

OnMapReadyCallBack – This class has been implemented to make the Google Map visible in the Maps page.

Testing

The app is tested on three Emulators, Pixel XL, Nexus S and Nexus 9.

Issues Faced:

- In Nexus9, even though a pop up was given for allowing permissions for Camera it quit the app abruptly. However, upon restarting the app it worked fine. Have not seen this issue again.
- Gestures Testing cannot be done through the Android Studio by touch. You will need to click Ctrl + left mouse click to test the same.
- On every new emulator, few issues existed. Majorly related to permissions.

Barkha.Java001
CS443
01774410

4. References

References for the project:

https://developers.google.com/maps/documentation/android-sdk/utility/setup

https://developers.google.com/maps/documentation/android-sdk/utility/marker-clustering

https://developer.android.com/guide/topics/media/camera

https://developer.android.com/reference/android/provider/MediaStore

https://developer.android.com/training/location/change-location-settings

Experiences and Thoughts

Building this app has made me realize the importance of frequency of updates required. During my build, I found various documentations on functions I required for my app, which were outdated for future versions of android. One of them being permissions, newer android versions require permissions from the user rather than the manifest.

Few updates, I was trying to achieve but was not able to because of the time constraints are as follows:

1. UI update
2. Goals – Take an input from user on his/her goals. Reward them based on the goals they have set. This would require a database to manage to maintain the goals and check if they were achieved. Also, a toast show message if any goal was achieved.
3. Share – Share your goals and achievements with friends. Sharing buttons for Facebook or Twitter.
4. Save Image – When the user saves images, he can also add notes to his pictures. Things he can remember and revisit if planning the same trip.
5. View Map