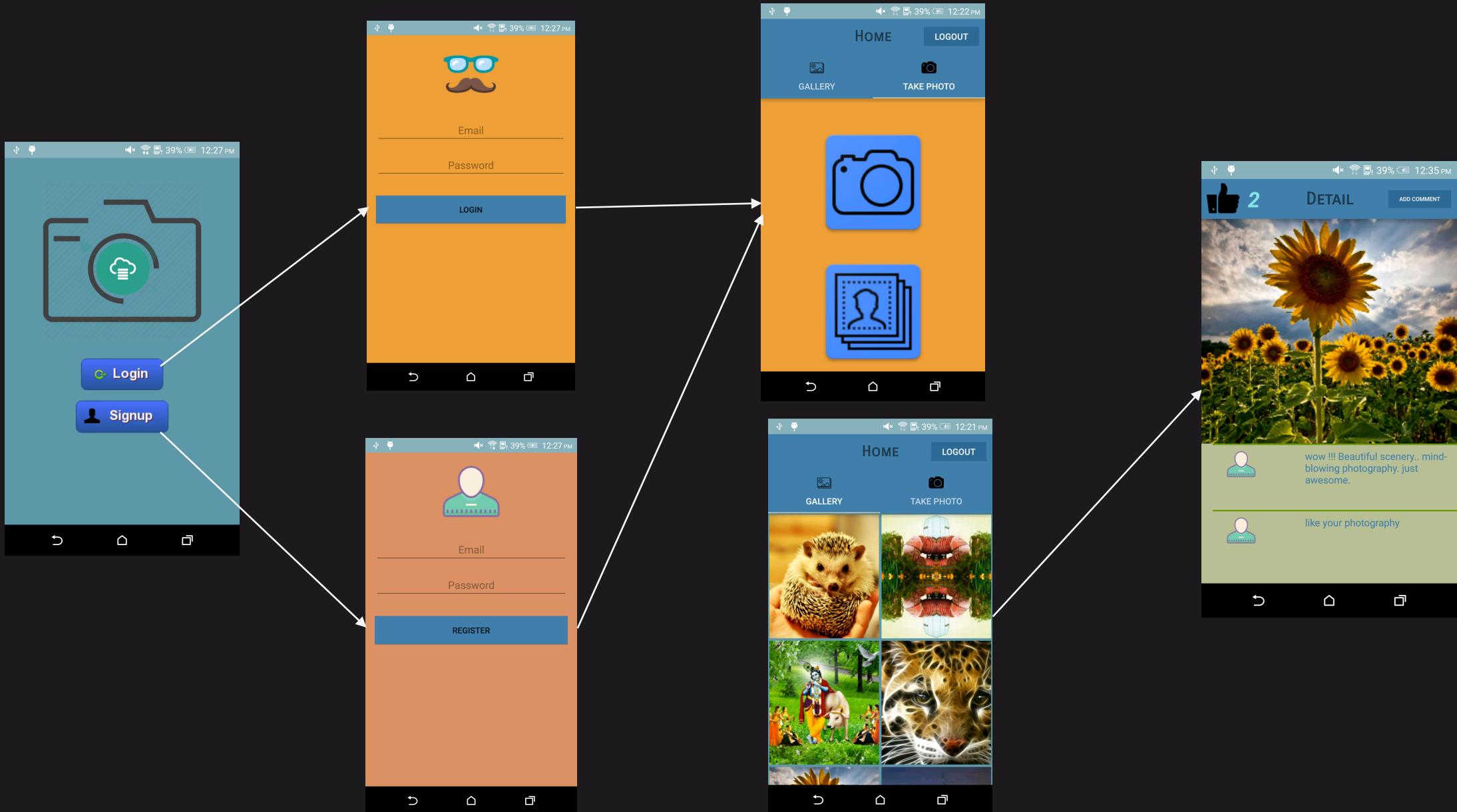


BUILDING OUR 5TH APP

CLOUD CAMERA



CLOUD CAMERA



WHAT WE'LL BUILD

- *Create an app that lets users post photos to 'the cloud' and lets users see any/all photos that have been posted. Users should be able to leave comments and like photos that have been posted. Imagine a simpler version of Instagram—but there is no concept of followers. A user can see all images.*



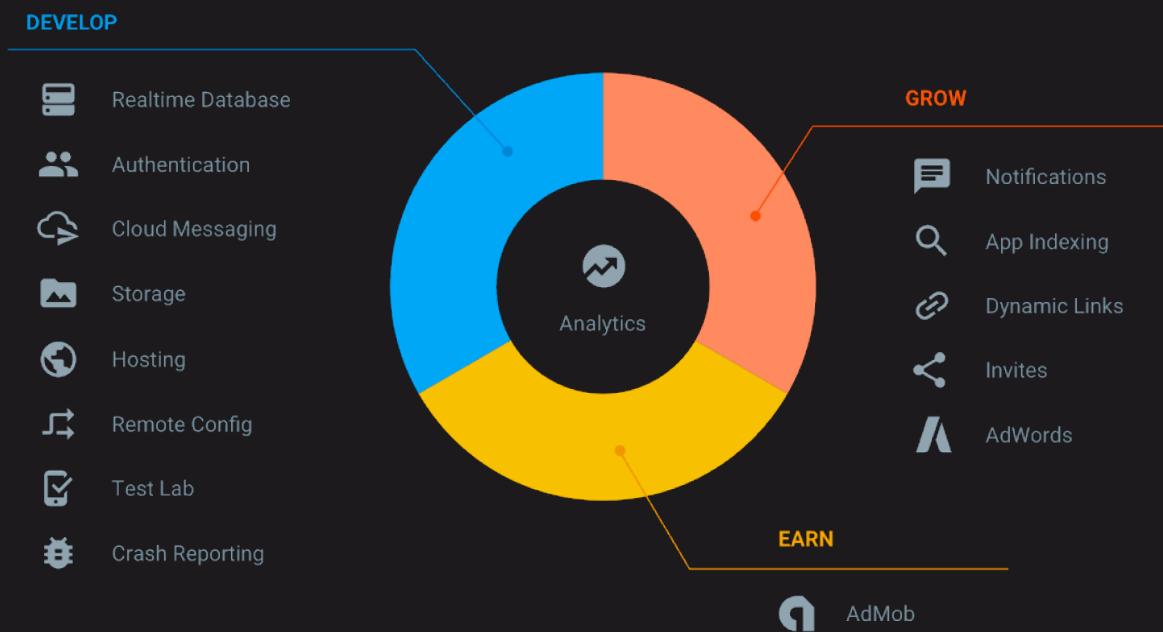
TECHNOLOGIES AND CONCEPTS

- *Planning and designing an app*
- *Web based backend using FireBase DataBase*
- *User Authentication using FireBase Authentication*
- *Store Images to Cloud using FireBase Storage*
- *Android Fragments*
- *Android RecyclerView for showing a grid of thumbnails*
- *Create Multiple Tabs using TabLayout and ViewPager –*
(<https://developer.android.com/reference/android/support/design/widget/TabLayout.html>)
- *Accessing the Camera and Image Gallery on your Android device using Implicit intent.*
- *Glide framework for loading images.* (<https://github.com/bumptech/glide>)

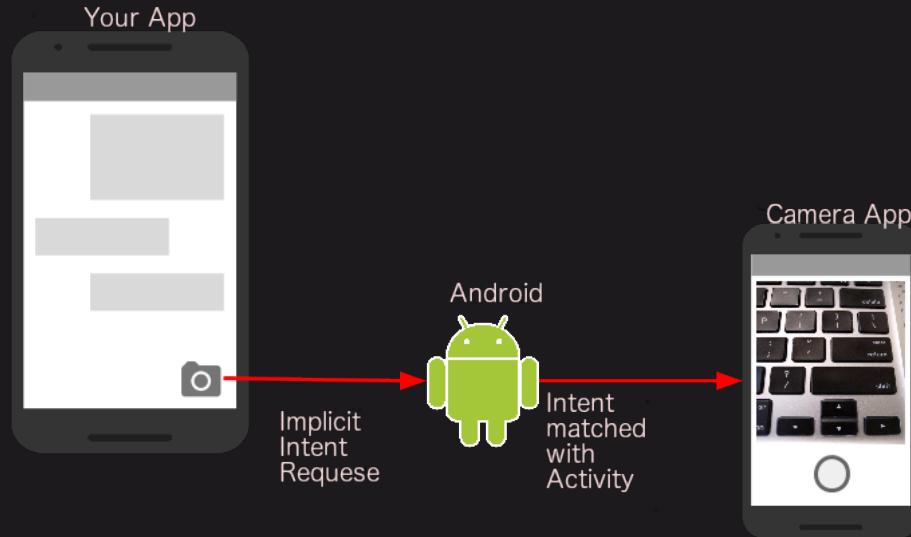


FIREBASE

- *Firebase is a mobile platform that helps you quickly develop high-quality apps, grow your user base. Firebase is made up of complementary features that you can mix-and-match to fit your needs.*
- *Firebase comes with bunch of features essential for every android app starting from authentication to hosting the app.*



IMPLICIT INTENT



- *Implicit intents do not name a specific component, but instead declare a general action to perform, which allows a component from another app to handle it. For example, if you want to show the user a location on a map, you can use an implicit intent to request that another capable app show a specified location on a map.*



ASSIGNMENTS



1. Install the Firebase SDK. Read the getting started documentation about Firebase Storage.
2. Upload a photo to Firebase storage and get a reference to it.
3. Design a data model for your app and how it will be stored as JSON (including a reference to the file)
4. Using the Firebase SDK make a POST with metadata about the image you uploaded earlier (including the reference to it). Metadata is all the information related to the image (but not the actual image) like image name, date created, file type, caption, etc.
5. Get the data from the API and fetch the image. Display these things in a grid. (Use RecyclerView for that)



6. Create API requests to add comments and likes.
7. Create a Login/Registration screen using Firebase authentication. Now without authentication you can't see those images.
8. Logged in user can see a screen will have two tabs:
 - a. One tab of the app will have a RecyclerView to display image thumbnails
 - b. The other tab should offer the user a choice of whether to launch the camera or the image gallery to select an image to upload.
9. Clicking on an image in the first tab should launch a detail Fragment, that shows a larger version of the image, a like button, number of likes, and a list of comments.



COMPLETION CHECKLIST

- *The app should have multiple tabs.*
- *User can take or select an image (using intents).*
- *User should be able to post a photo to Firebase Storage using the Firebase SDK using the Firebase SDK*
- *One particular User Can Like a specific photo Once.*
- *Photos from Firebase should be displayed in a RecyclerView*

