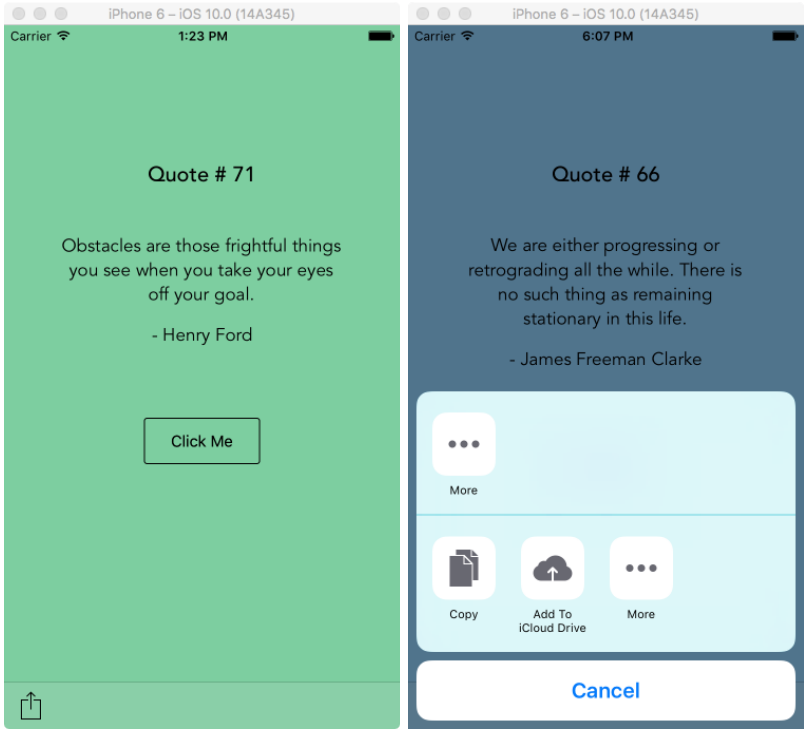
Here's a brief overview of my projects related to Android and IOS:

I currently have 4 published applications on Google Playstore. All IOS apps were built using swift. The applications are arranged starting from my latest work. You can find all the links at the bottom.

Good Quotes(IOS):

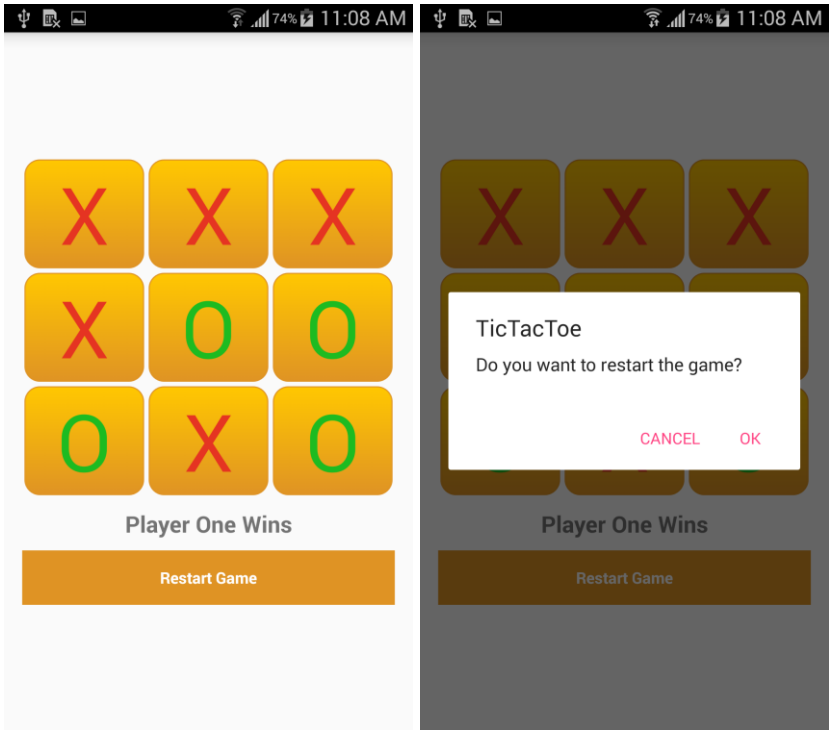
Using forismatic api, developed a simple application that uses Alamofire to make web requests. This application includes:

- Network call using Alamofire
- JSON Parsing
- Use of Persistent data to save the quote numbers.
- Share option, to share text with other applications



Tic Tac Toe(Android):

I developed a simple tic tac toe few months back and I decided to polish it and publish it out. It's a simple two player local game. Nothing fancy. (Future plans : Will probably set up a server to make people on two different phones play against each other).



User Information App(Android):

Developed a simple app which does the following:

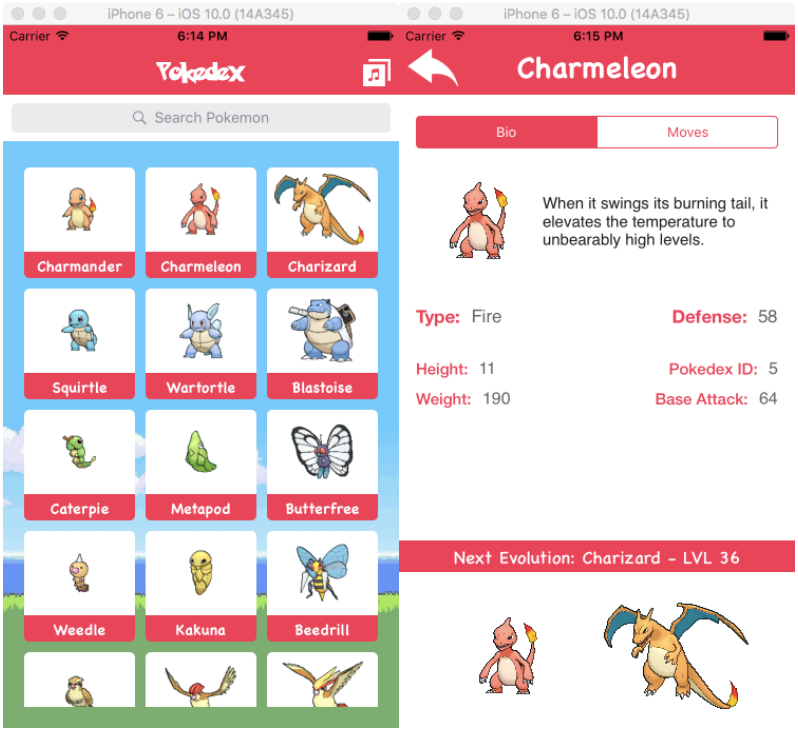
- Saves user information in to the database,
- Views information of all the users,
- Delete the information of the user and
- Update the information of the user.

I created my very first server using **NodeJS** . I used Heroku to host my NodeJs Application. The database I used in the app is a NoSQL database (MongoDB) using mlab. This is my beginner attempt on both nodejs and mongoDB. The main focus of this application is the **MVP** design pattern, **NodeJs** server and **MongoDB** database.

Pokedex (IOS):

Developed when learning with devslopes, gained a lot of knowledge working on collection Views.

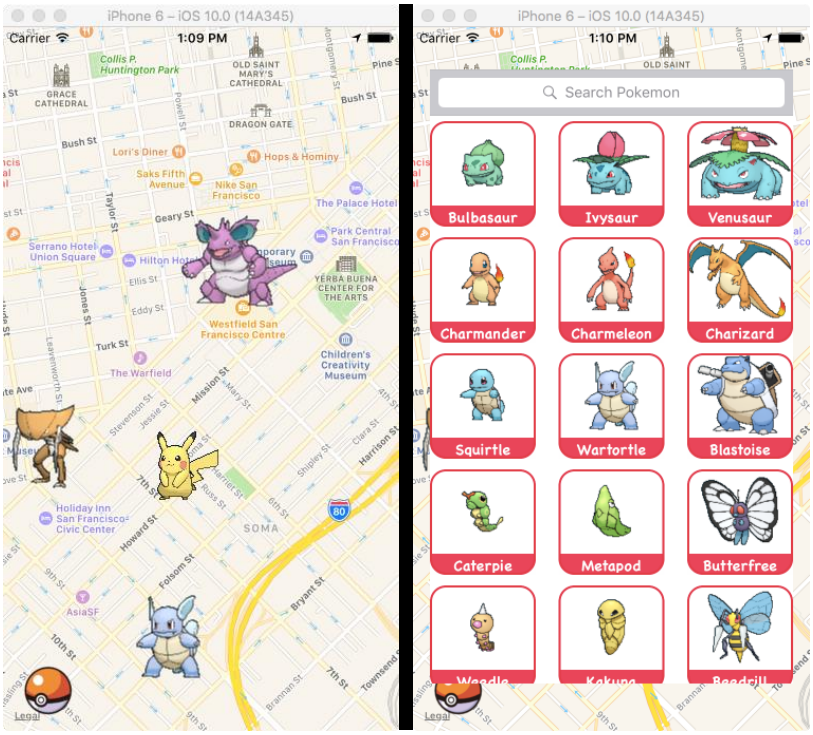
- CollectionView
- AlamoFire to PokeApi v1
- AudioPlayer to play music



PokeFind(IOS):

This application is a mixed one, learned through devslopes but integrated pokemon selection that I learned from the previous application:

- Used Mapkit
- Used Firebase and GeoFire to Store the pokeom Sightings
- Used a ContainerView to display an overlapped View to select pokemon
- Played with annotationView to annotate different points
- Used LocationManager to fetchLocations

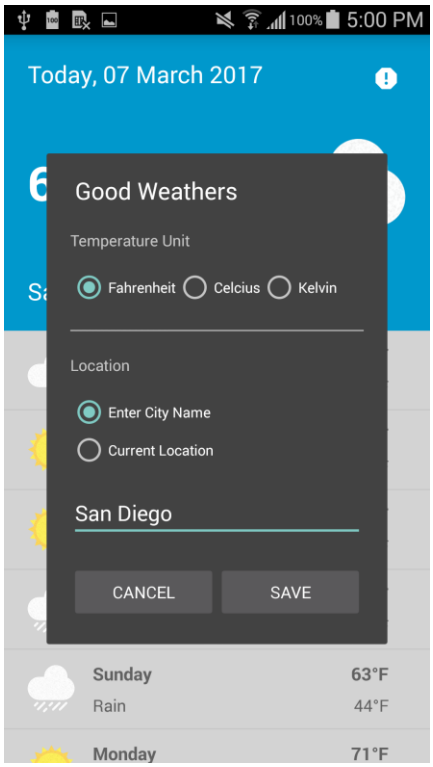


Good Weathers (Android and IOS):

Developed a good looking weather app which allows user to get weather information using openweathermap API for its current location as well as he can enter a location manually. I've submitted the app on the play store. It should be up any time now.

Android:

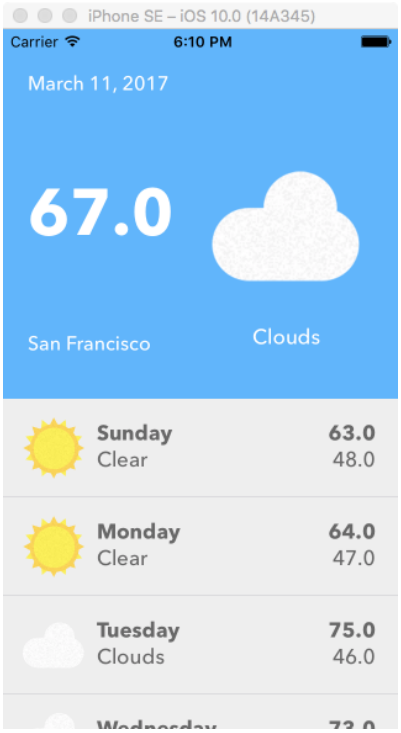
- Used Google location API to get location data.
- The app also allows the user to change the temperature units.
- Used Sharedpreferences to save the information (Temperature unit and location name)
- Used Retrofit, GSON, Play Services and few other libraries.
- MVP design pattern



IOS:

I learned to develop this app for IOS with Devslopes. Features include

- Using LocationManager to fetch current Location
- Using Alamofire to make API Calls
- JSON Parsing
- Images were stored locally



Worked as an Android Developer at App Partner:

I've worked on plenty of different applications while working there. I've developed an entire application on my own and I've worked on adding features and Bug fixes to existing applications as well.
Following are the name of the apps I've worked on:

- BlackCarFund
- JusMove
- Genlines
- ConAir
- B&H

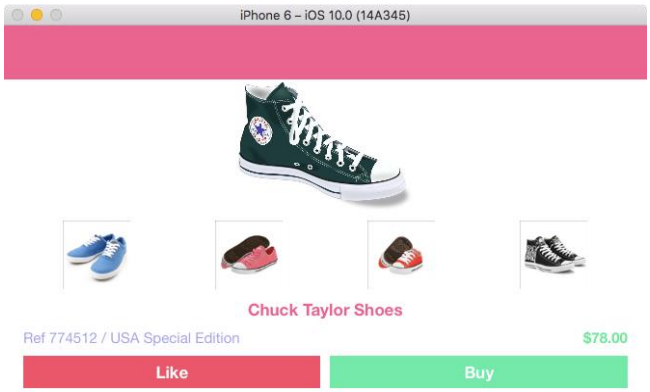
Over my time at the company I've got myself quite familiar with

- Pivotal Tracker
- Git (SmartGit/Command line)
- Slack
- Zeplin

Working on different projects gave me a great deal of exposure. Used MVP / MVP Interactor pattern in all of the applications. Following proper company coding guidelines eventually helped me write efficient and clean code.

Stack View (IOS):

A simple implementation of how to use Stack Views in IOS. Such an amazing feature that allows you to support different screen sizes.

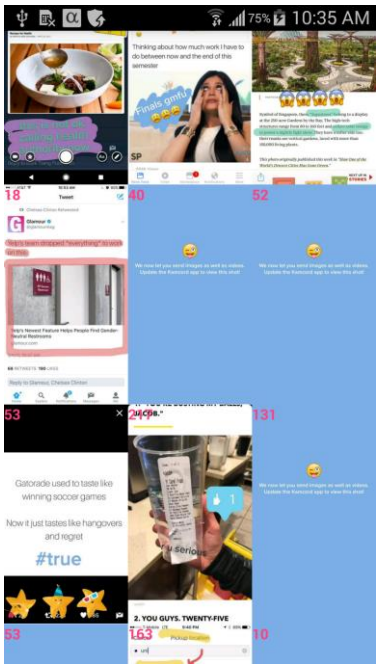


Kamcord Coding Challenge (Android):

Challenge included to build only one app, but I made two:

App 1 (Original challenge)

- Used TextureView and ImageView in a gridlayout with recycler view.
- Allowing user to click on any thumbnail to play video associated with it.
- Made asynchronous calls to get the JSON.
- Implemented a custom on click on recyclerview.
- Support to set 5 items per row for tablets.



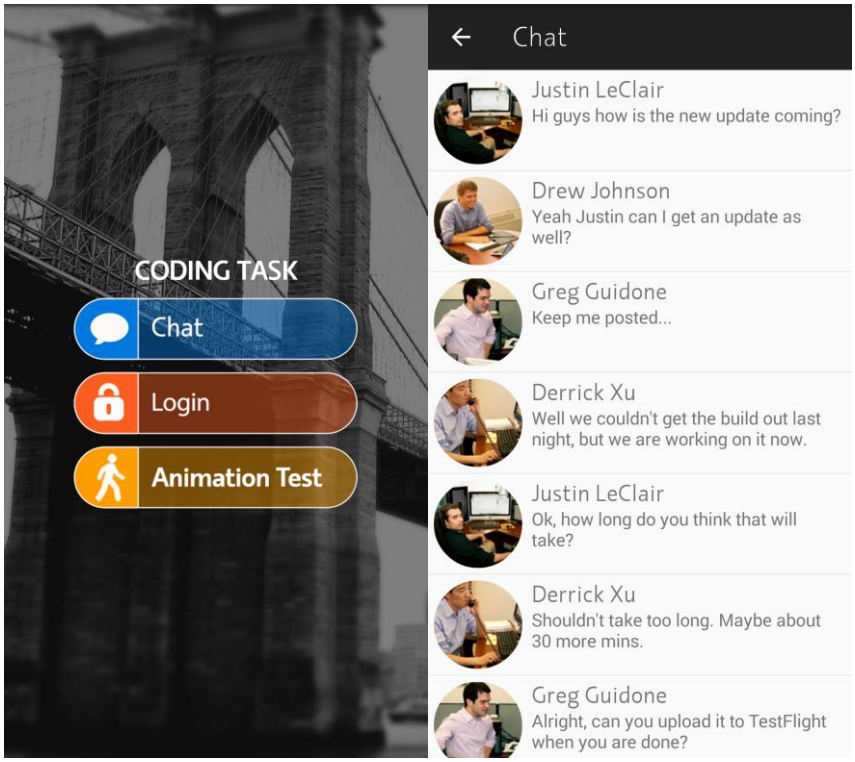
App 2

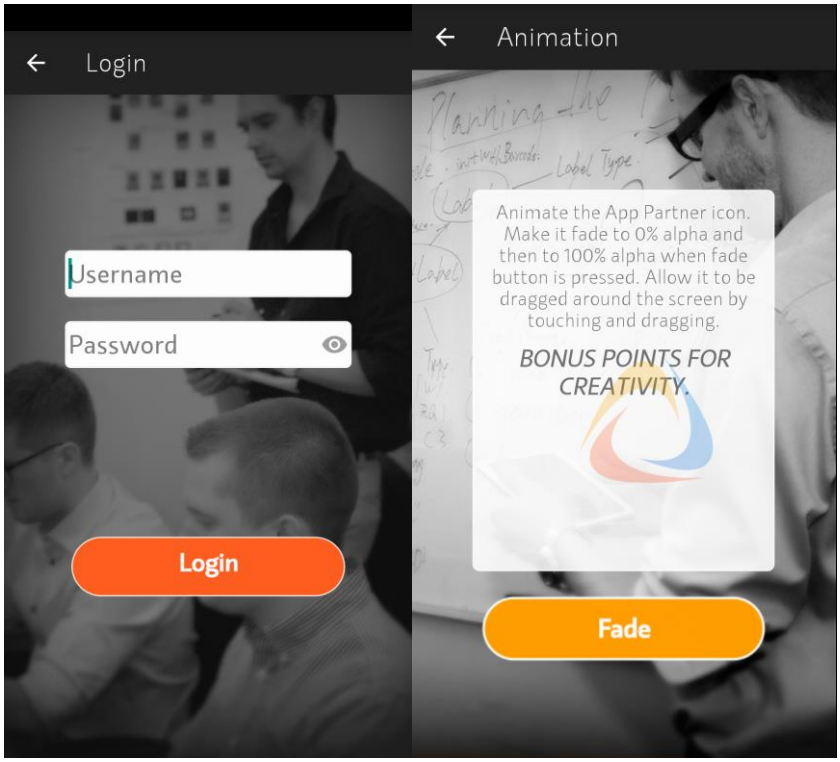
- This was a prototype of the original application of Kamcord.
- Used VideoView to play the videos.
- A Recyclerview with each time covering the whole video.

App Partner Coding Challenge(Android):

Built an application with following features:

- Using glide populated a listview.
- Made a post request for login using volley.
- Used fade in and fade out animation and on an imageview and allow user to drag drop that image.

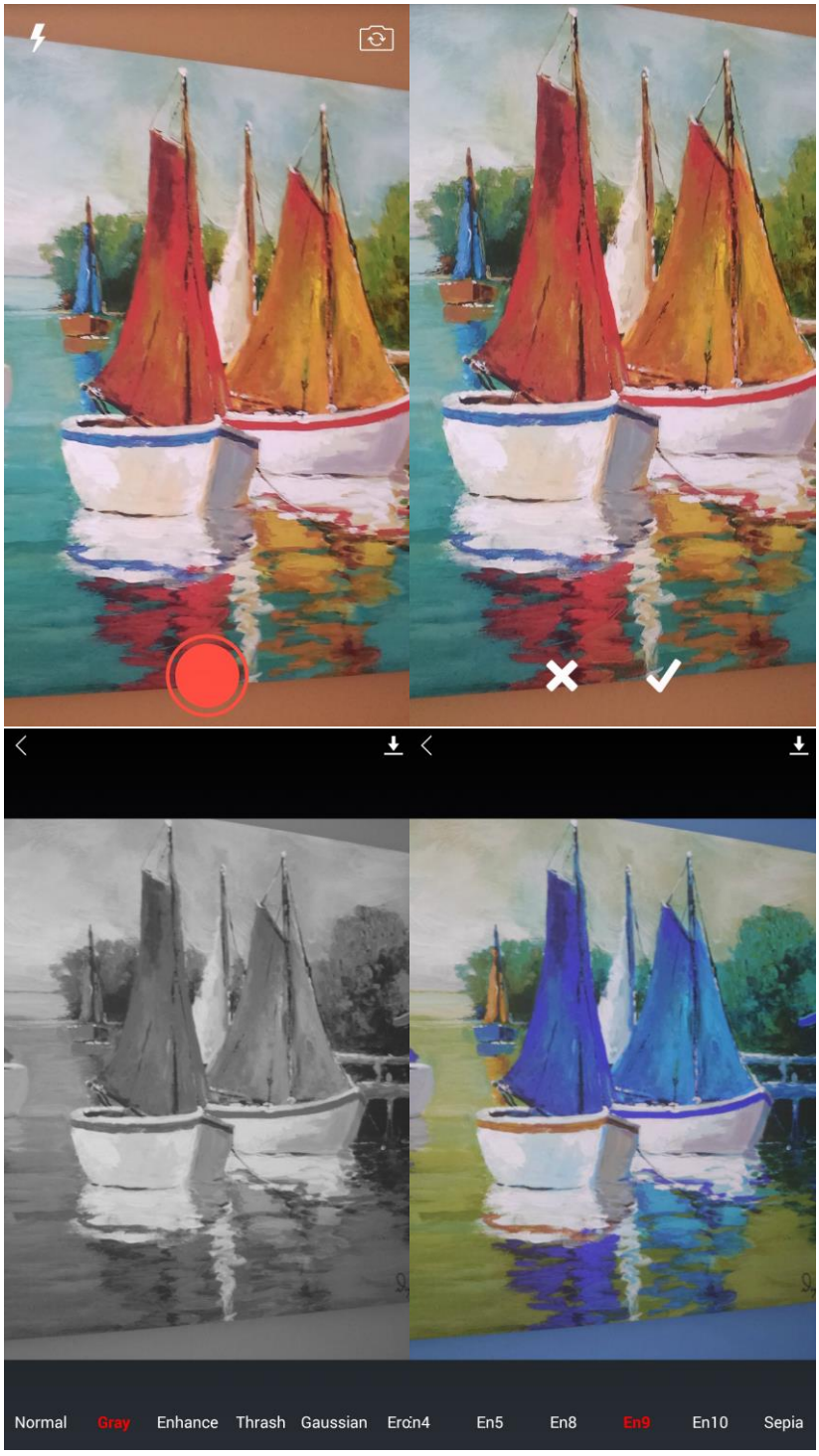

CODING TASK
Chat
Login
Animation Test
Chat
Justin LeClair
Hi guys how is the new update coming?
Drew Johnson
Yeah Justin can I get an update as well?
Greg Guidone
Keep me posted...
Derrick Xu
Well we couldn't get the build out last night, but we are working on it now.
Justin LeClair
Ok, how long do you think that will take?
Derrick Xu
Shouldn't take too long. Maybe about 30 more mins.
Greg Guidone
Alright, can you upload it to TestFlight when you are done?


Login
Username
Password
Login
Animation
Animate the App Partner icon. Make it fade to 0% alpha and then to 100% alpha when fade button is pressed. Allow it to be dragged around the screen by touching and dragging.
BONUS POINTS FOR CREATIVITY.
Fade

Cam Filters (Android):

Working on a simple camera application. The User can capture photo and apply different image filters.

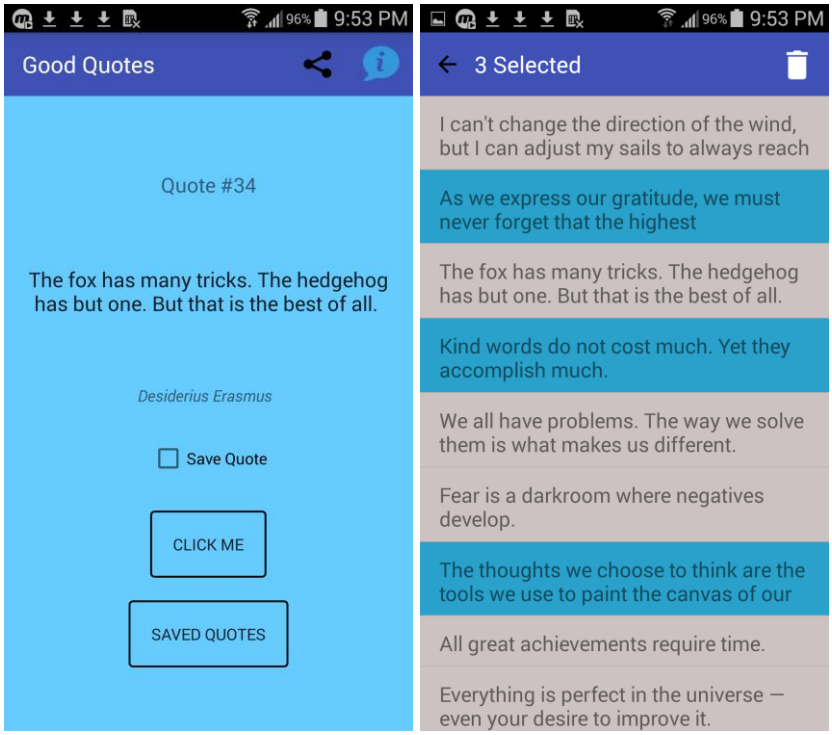
- Used Camera API to capture photo.
- Used Opencv Android library for Image processing.



Good Quotes (Android) :

I used Forismatic API which provides a random motivational quote when you pass a key to it. Following are its features:

- AsyncTask to make Asynchronous calls.
- SQLite Database to store quotes that user's like.
- Shared Preferences to save the count (Key) which gets passed to the API(Web) to get a quote.
- Contextual Action bar (to select multiple items from list view and delete them).
- Allows user to share Text and Image of the quote with other applications.



FindOut – The Sentiment Analysis App (Android):

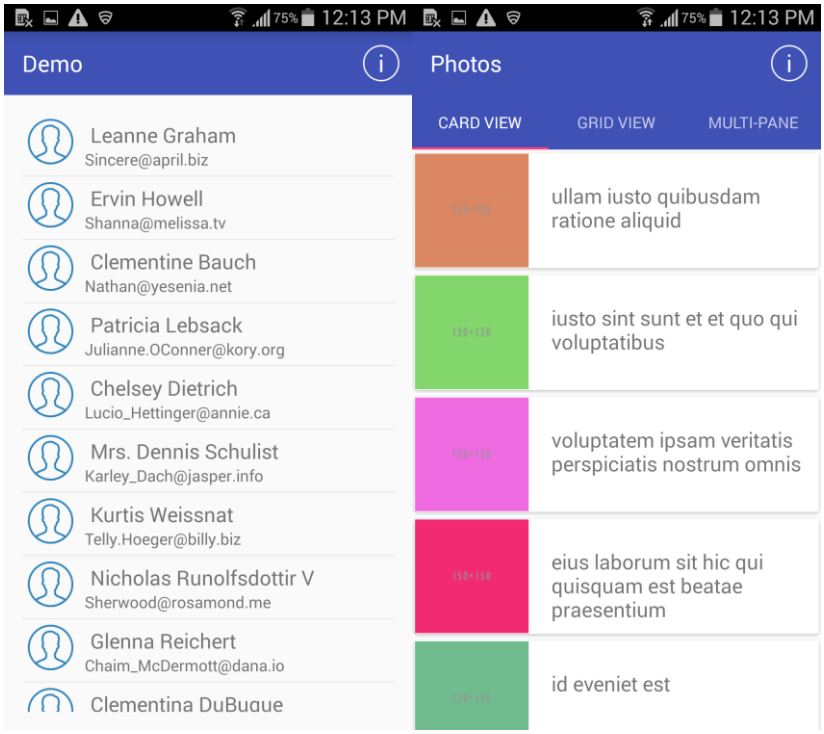
I along with four other teammates developed a Sentiment Analysis Android application. The application would read people's tweets regarding each presidential candidates. We would run our analysis on those tweets and would try to find out whether the tweet is positive, negative or a neutral(using NLTK library on back end). We would then render the result from the back end and show the data in terms of graphs (pie chart for positive, negative and neutral tweets and line chart to show his positive trend over a period).

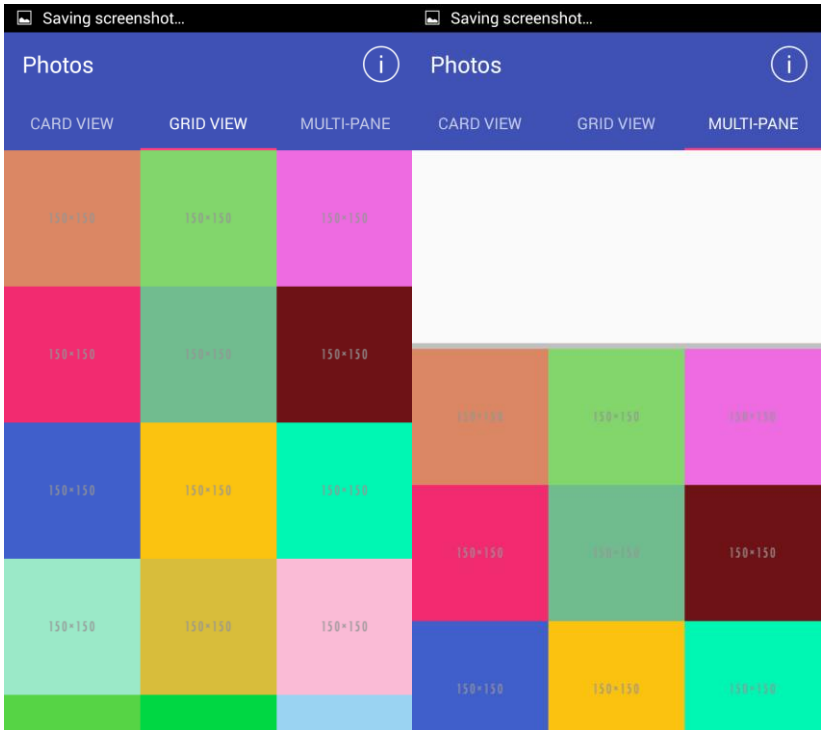
Demo Application I (Android):

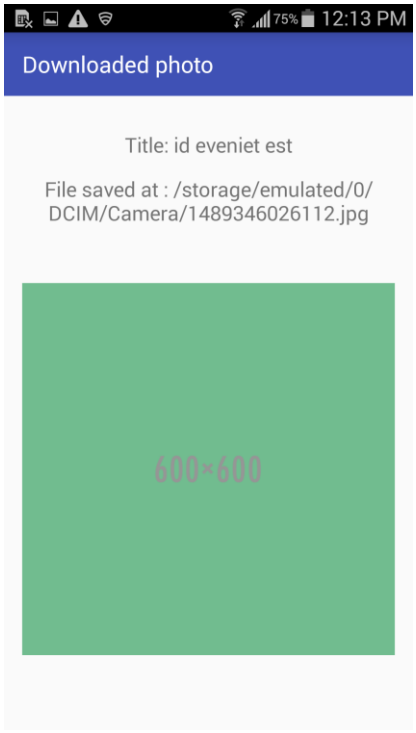
I used JSONPlaceholder to get fake json data. The application shows list of users, their albums, their photos and allows them to download those photos.

- Used ListView, GridView, RecyclerView and CardView.
- Used Fragments with View Pager and Tab Layout.

- Used Picasso to load the images and cache them. (Picasso provides cache mechanism).
- On Long press you can preview the image.
- Allowed user to download the image. (Applied run time permission for Android M).

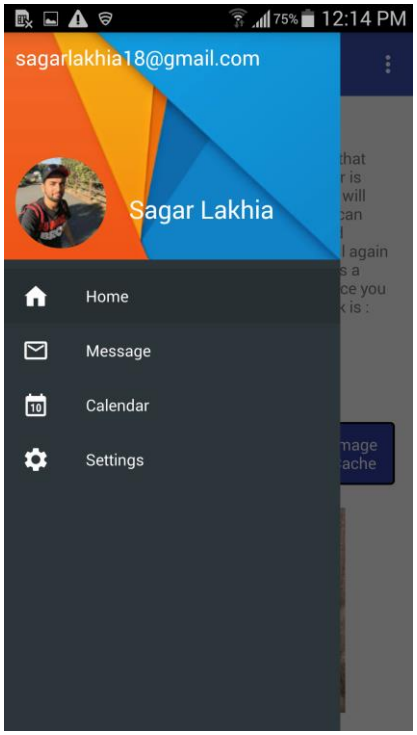


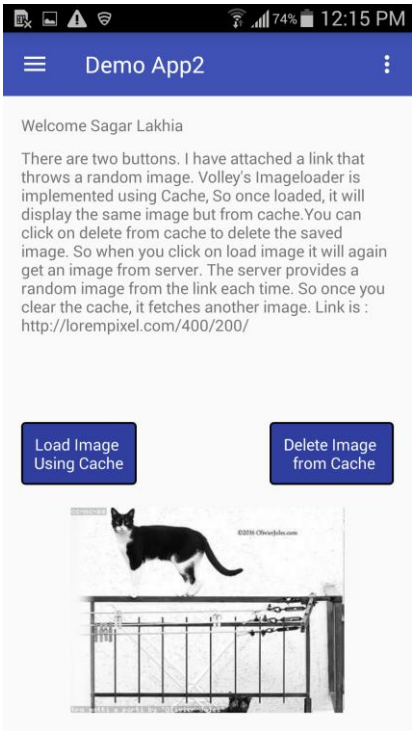




Demo Application II (Android):

- User authentication by providing a login interface via facebook, gmail and custom email.
- Push notification via Firebase console.
- A Navigation Drawer with circularImageView API.
- Used Glide API to load image.

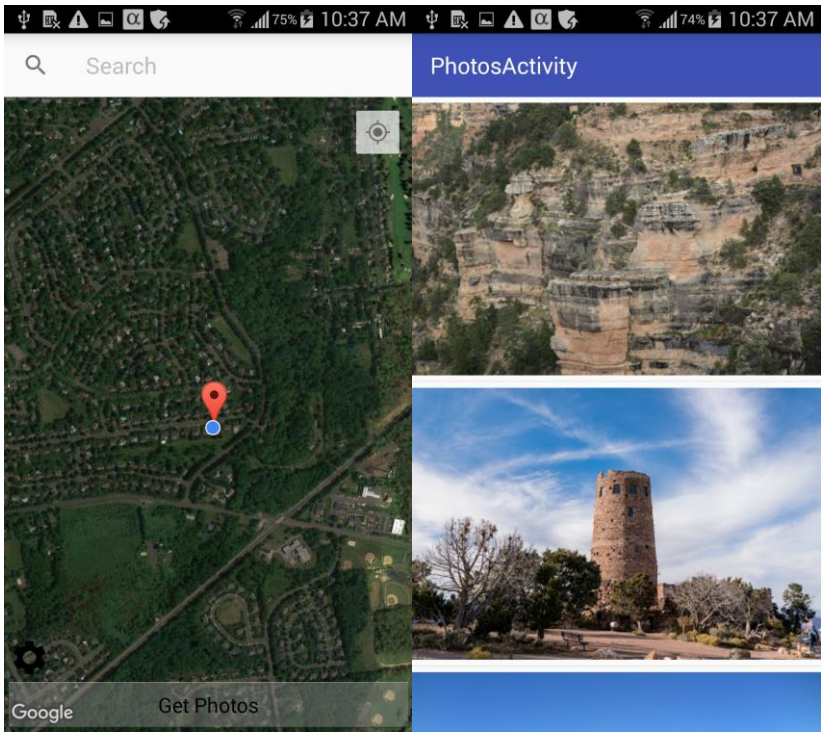




Google Maps and Places Demo App (Android):

It allows user to select a location (Suggestions provided by google places based on the input) and get photos in a Listview.

- Allows user to change Map Type to Hybrid, Normal, Satellite, etc.
- Allows user to enter location (Autocomplete with places API)
- Allows user to get photos (if any) from the places API.
- Used LruCache to cache those images.
- Used Android volley to make network calls.



RetroStyle Calculator(Android) :

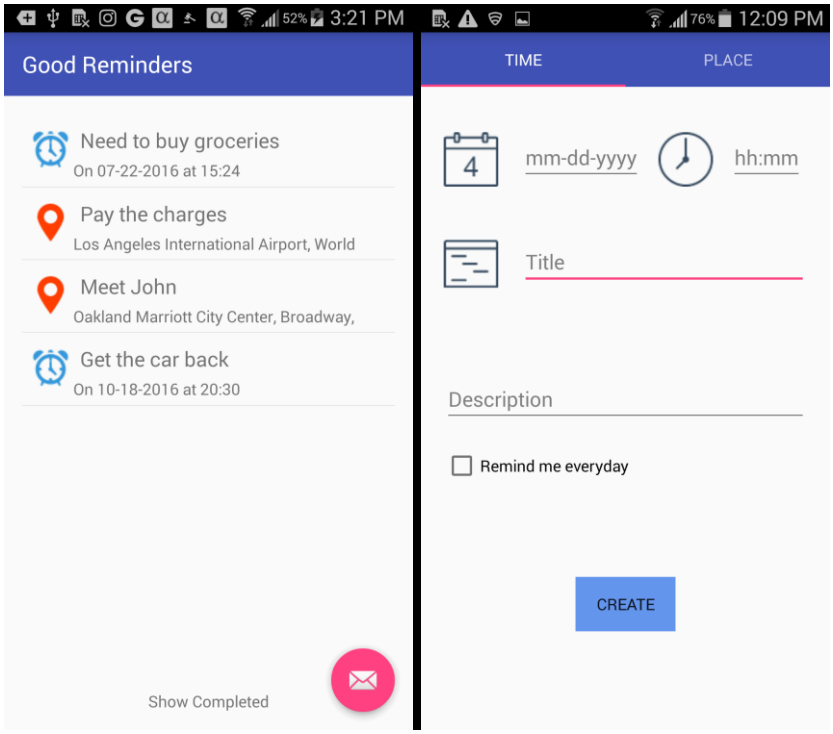
A simple calculator application with a space theme.

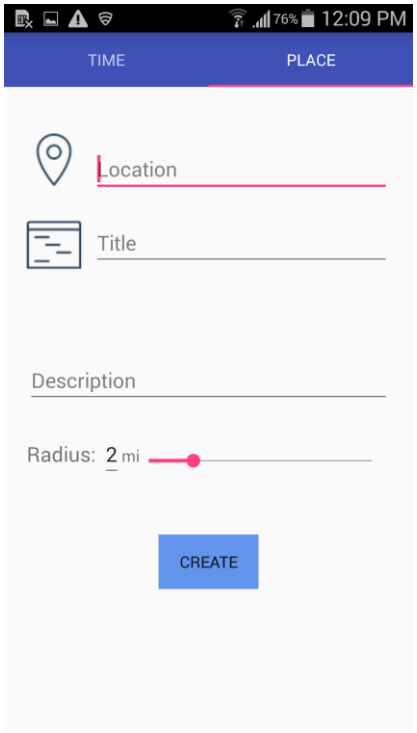
- Used Media Player to play the button sounds.
- Used Shared Preferences to save the user's selection of Sound On/Off.

Good Reminders(Android):

This application which allows user to set reminders based on time and location. Currently only Time based notifications were implemented.

- Used AlarmManager to set reminders.
- Used Ringtone Manager to set ringtone for the alarm.
- Used Pending Intents.
- Yet to implement Location based alarms.





Miscellaneous (Android):

- Have worked on animation with overridePendingTransition, SurfaceView and Canvas.
- StartedServices and BroadcastReceivers.
- Different Sensors like Accelerometer, Ambient Temperature, Step Counter, Step Detector, etc.

Till now I've used the following APIs:

- Glide
- Retrofit
- Butterknife
- Picasso
- aChartEngine
- Camera API
- Google Maps
- Google Places
- Google Directions
- Volley
- CircularImageView and many more

IOS Frameworks used:

- MapKit
- FireBase
- Alamofire
- Core Location
- Webkit

I like to explore a lot. Right now I'm learning about **RxJava** and **Dagger**.

Published applications:

Tic Tac Toe

https://play.google.com/store/apps/details?id=com.blackfruitapps.tictactoe

Good Weathers

https://play.google.com/store/apps/details?id=com.blackfruitapps.goodweather

Good Quotes:

https://play.google.com/store/apps/details?id=com.blackfruit.goodquotes

Retro Calculator:

https://play.google.com/store/apps/details?id=com.blackfruit.retrostylecalculator

---

Applications that are not on Play Store but have videos

FindOut:

https://github.com/sagarlakhia/FindOut

https://www.youtube.com/watch?v=6fo9uE9pKm0

Demo App I :

https://github.com/sagarlakhia/Android_Projects/tree/master/Demo_App1

https://www.youtube.com/watch?v=yF3drgT6di0

Demo App II :

https://github.com/sagarlakhia/Android_Projects/tree/master/Demo_App2

https://www.youtube.com/watch?v=rFVatMz9i5E

Kamcord Challenges :

https://www.youtube.com/watch?v=G247gWkCmRc

https://www.youtube.com/watch?v=AKuqCC5WWOs

AppPartner Challenges :

https://www.youtube.com/watch?v=isC_3gK2xkY

You can check my other work and more projects on my Github at https://github.com/sagarlakhia

Thank you.

Sagar