Roomer App

Directed Research

Jay Shah(W1114028) Rohit Makhija(W1149747)

Winter 2016

Table of Contents

Introduction

We all, as international students have faced the problem of finding apartments and room mates. Roomer is an app that helps you solve this problem and helps you find roommates and apartments.

It is aimed at mostly the university and college students.

Technologies used:

Parse

Parse is used for the back end. Parse is a MBaaS(Mobile Back End as a Service). All the data related to the users is stored and linked entirely on Parse.

Google Places API

The Google Places Web Service is a service that returns information about a specific "place": an establishment, a geographic location, or prominent point of interest using an HTTP request. Place requests specify locations as latitude/longitude coordinates.

Two basic Place requests are:

a Place Search request

a Place Details request.

Generally, a Place Search request is used to return candidate matches, while a Place Details request returns more specific information about a Place.

We first use a Place Search Request to find the list of the apartments and then use the Place Details Request to find the specific details of a particular apartment like contact number, email, website, hours of operation etc.

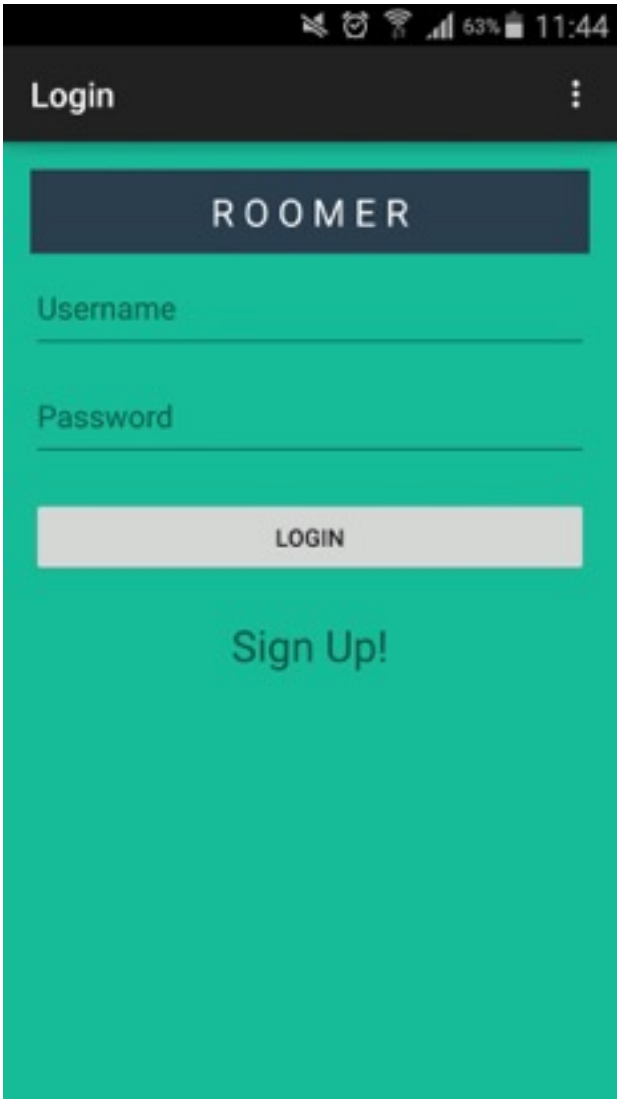
(We have not created an index, but a flow structure to navigate you through the app. It starts below.)

Login Page

(with all Successful Validations)

Database: Parse

A user can view this first page when he enters the application. He can create his login and password through sign up. Once he has the required Login and Sign up

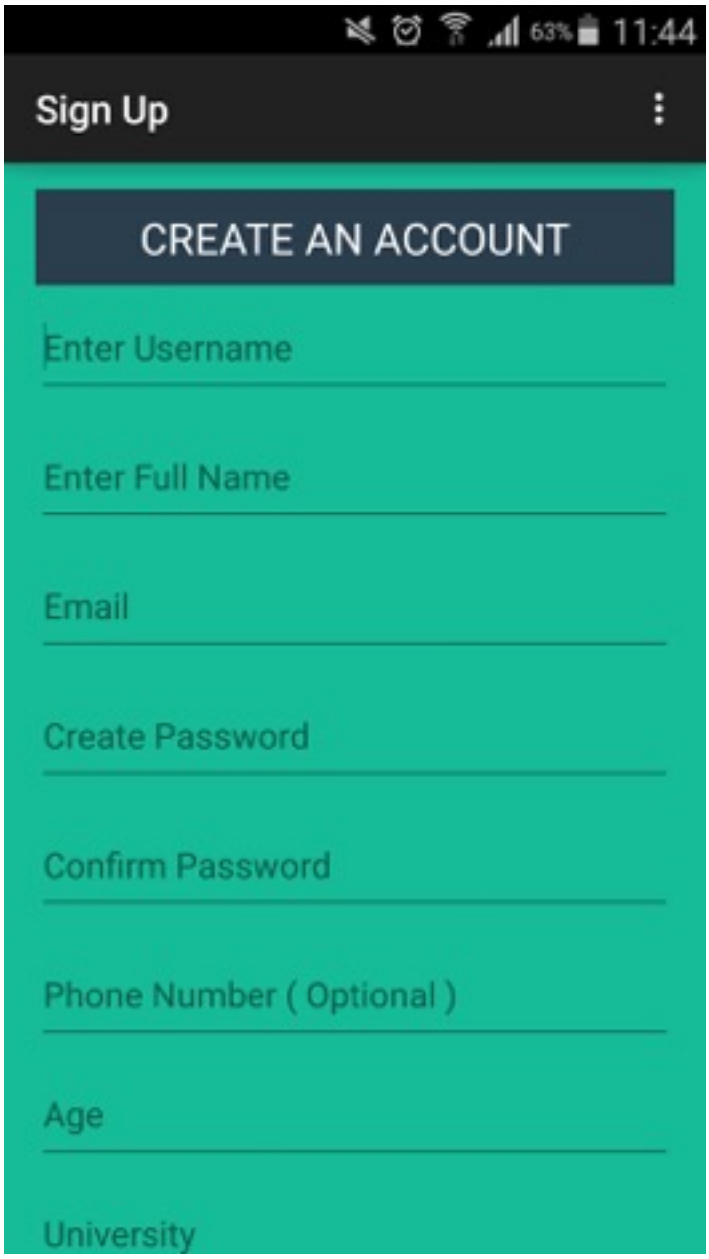


he can then Login to the app and see the next page. All the required validations are done and we have made sure that only a correct user can login.

Sign Up

(with user live photo)

This is our Sign up page where the user can create his account by entering all the



required detail, confirming his password. He also has to submit his image to the

application where a live camera will be opened in through the app and the user can click and submit an image to it. We have kept to mandatory for a user to click an

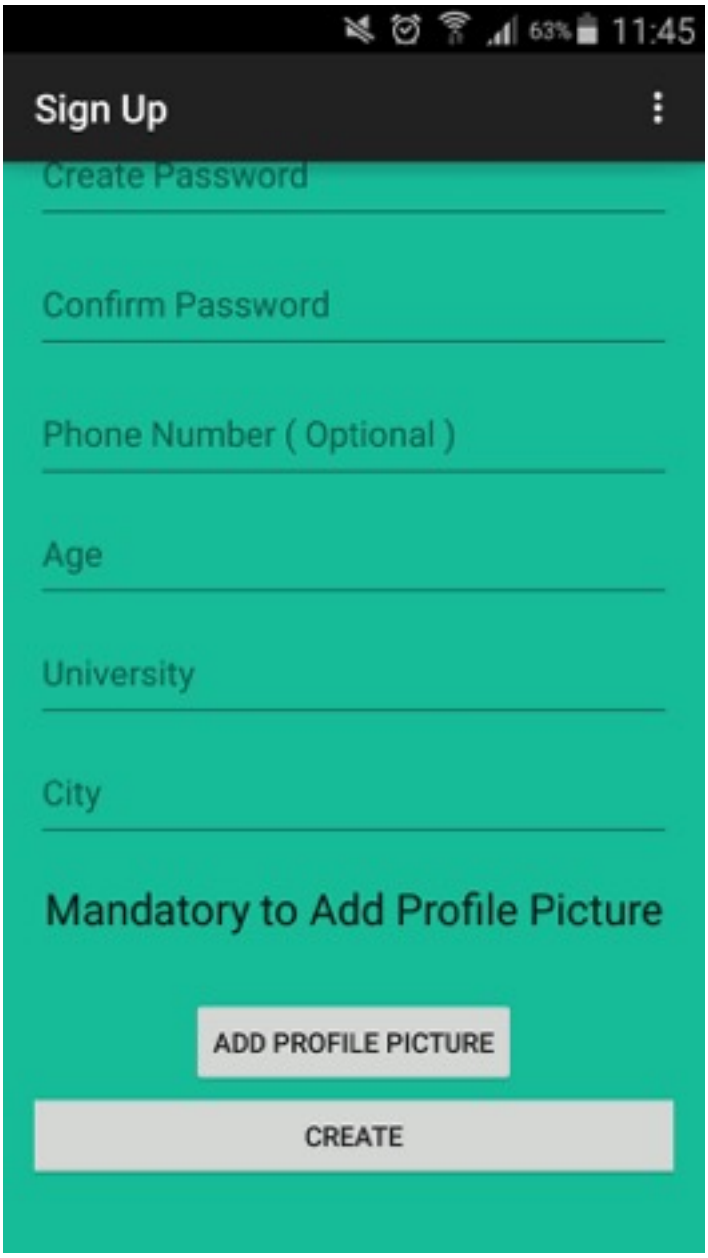
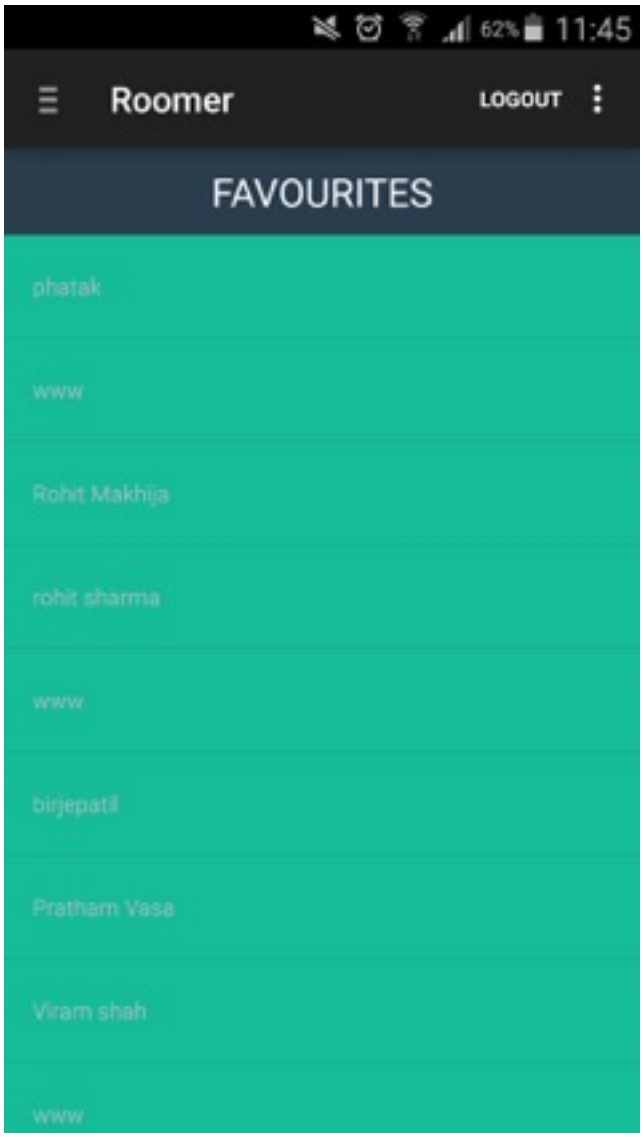


image of himself for security reasons. We are saving this image as a bitmap string in our parse database.

Landing page

(with list view)

This is out first page when the correct user logs into the app. It'll be empty initially but once you add some contacts to your favourites. Like some people you might have browsed and might be interested in checking their profiles as a potential room

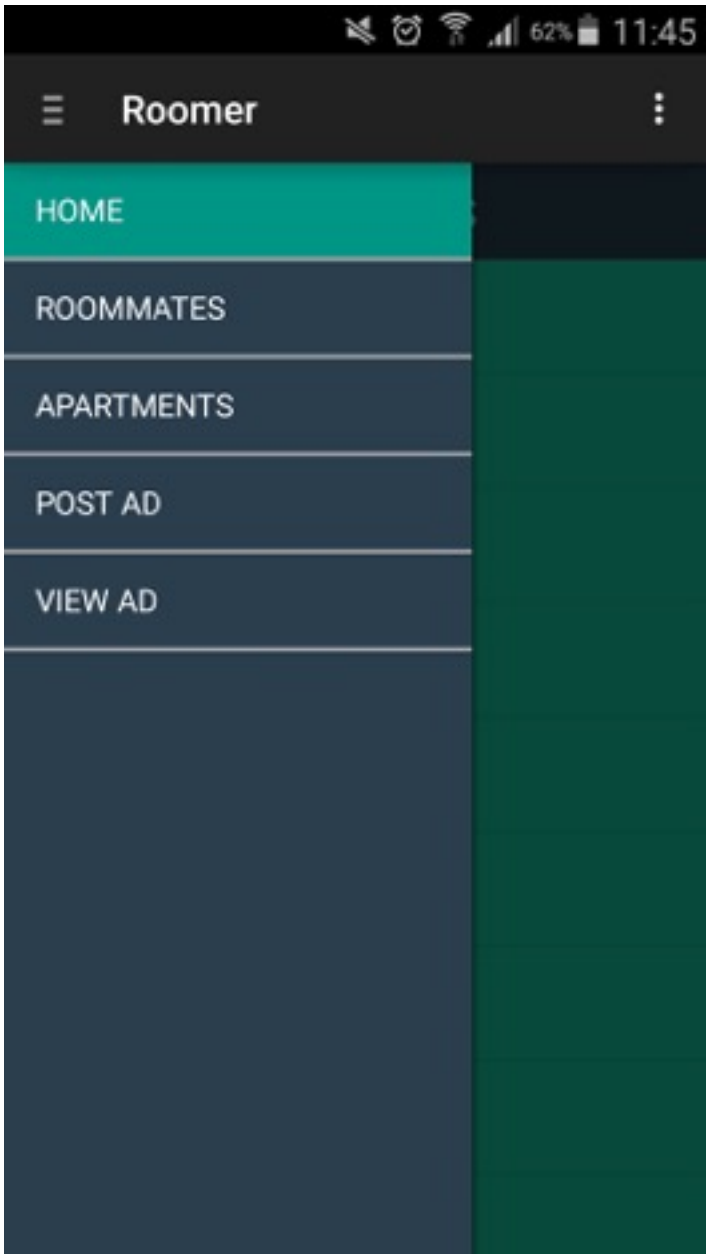


mate. This is a basic list view we created and displayed the individual names along with a logout button. and a navigation drawer icon.

Navigation drawer

(shown on all activities)

This is a simple navigation drawer with 5 options. viz home, roommates, apartments, post ad, view ad. This will be maintained throughout the app. User can easily navigate through out the

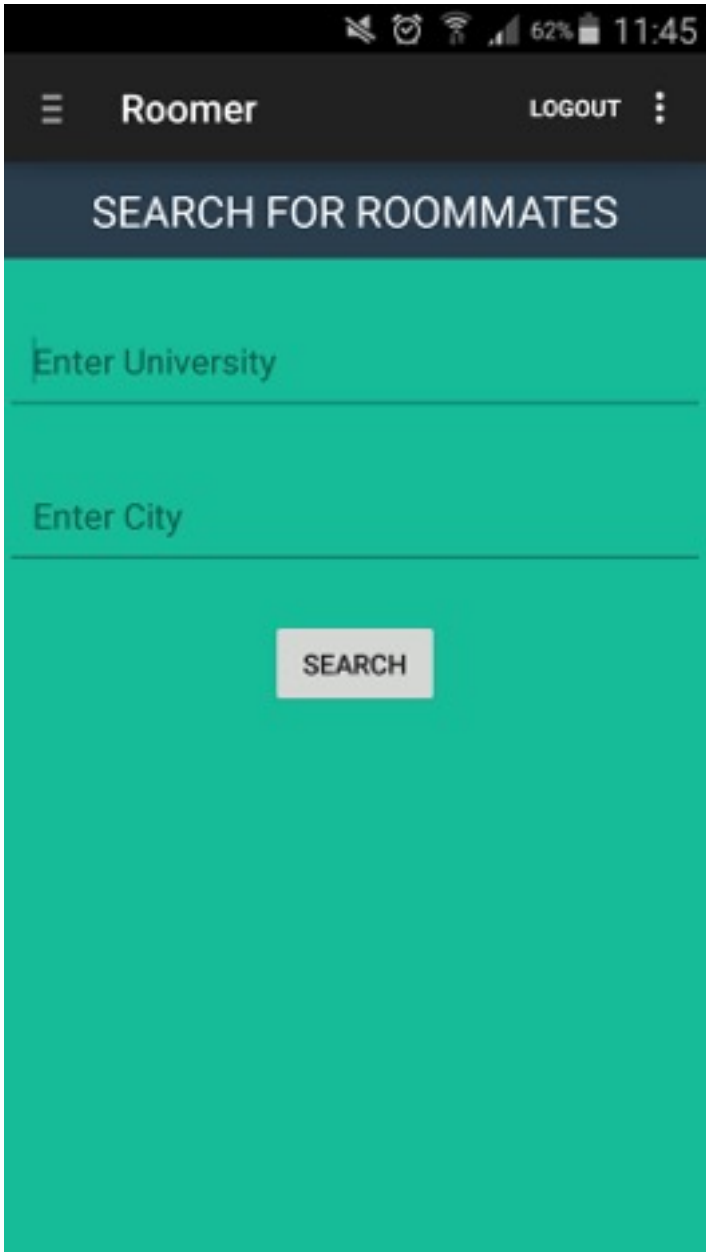


app because of this feature. We had various options of also maintaining a quick header or footer with these options but we choose this options because of user experience easiness.

Search Roommates

(API integration)

Once you click on the first item of the navigation drawer, you will be directed to this activity. Here you can enter any university's name and city and click the search

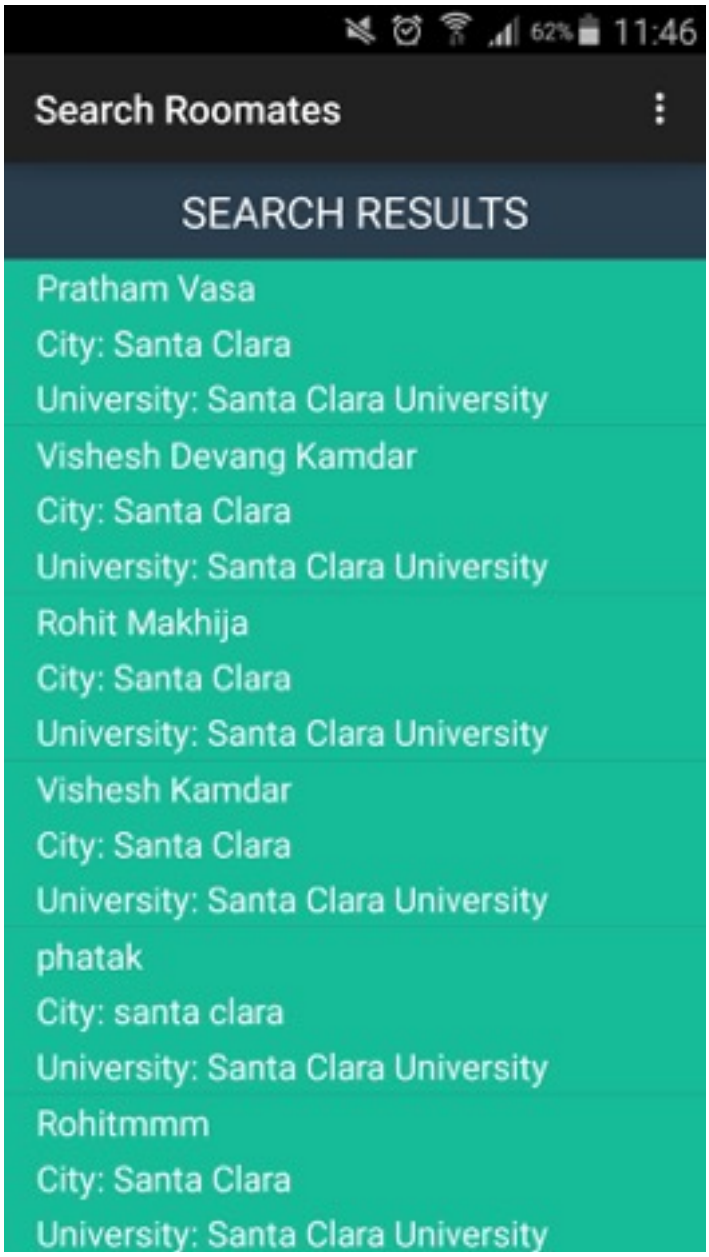


button to make a API call (Googles Places API) and see a list of students with the same need of roommates.

Search Results

(with custom list view)

This activity will be basically be a custom list view page displaying all the names , cities and universities of different students. We get all these details by implement

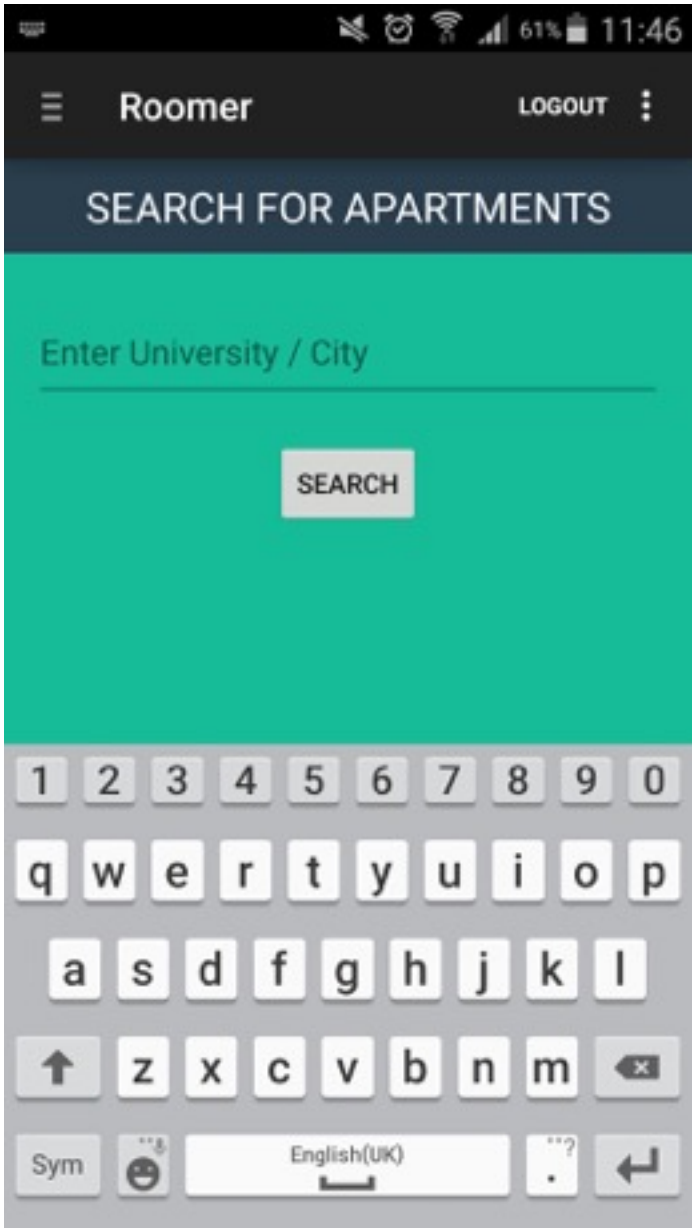


the Google API and storing and fetching and displaying it correctly through list adapters. We can also click on them and make a call or send and email to them.

Search Apartments

(Google Places)

On clicking the second item in the navigation drawer, you will be redirected to this page. Here we have implemented the googles's places api to make a search query

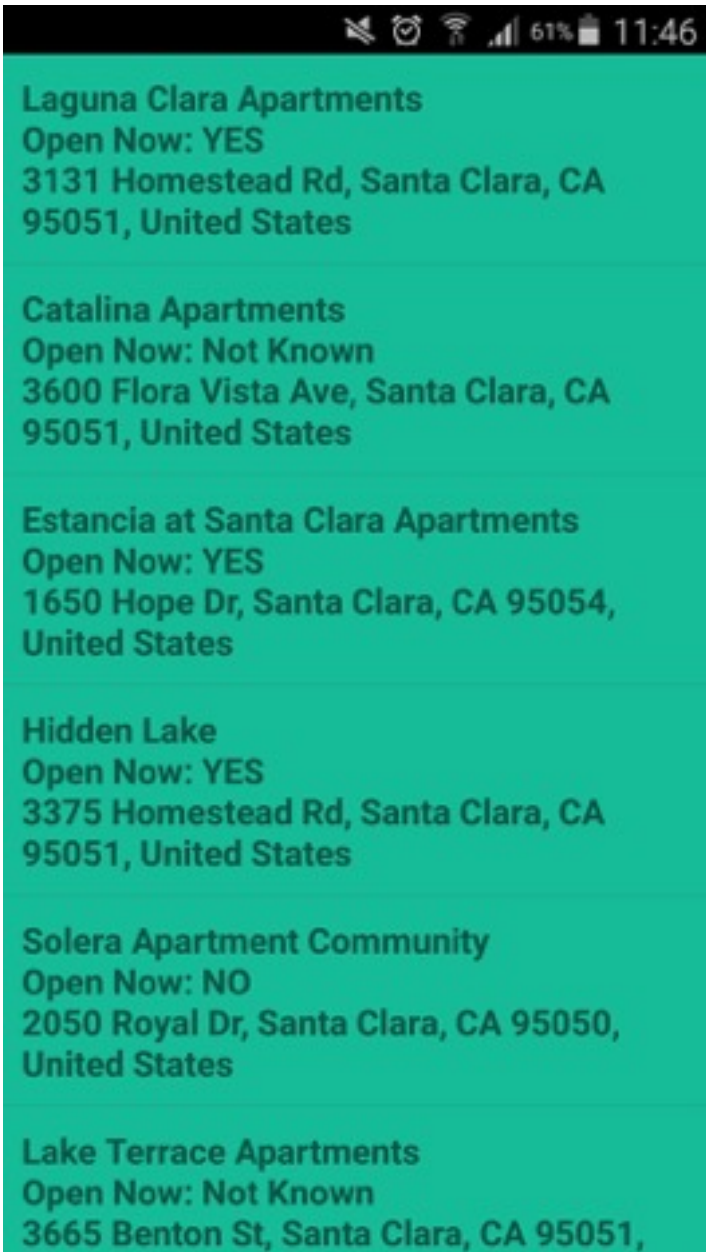


around that university and give you a list of all the apartments with loads of information about it.

Aparment's List

(Custom List Adapters)

We have created a list view to show all the apartments around us. Each List item will have the apartment name, its status of being open or now for enquiry and its

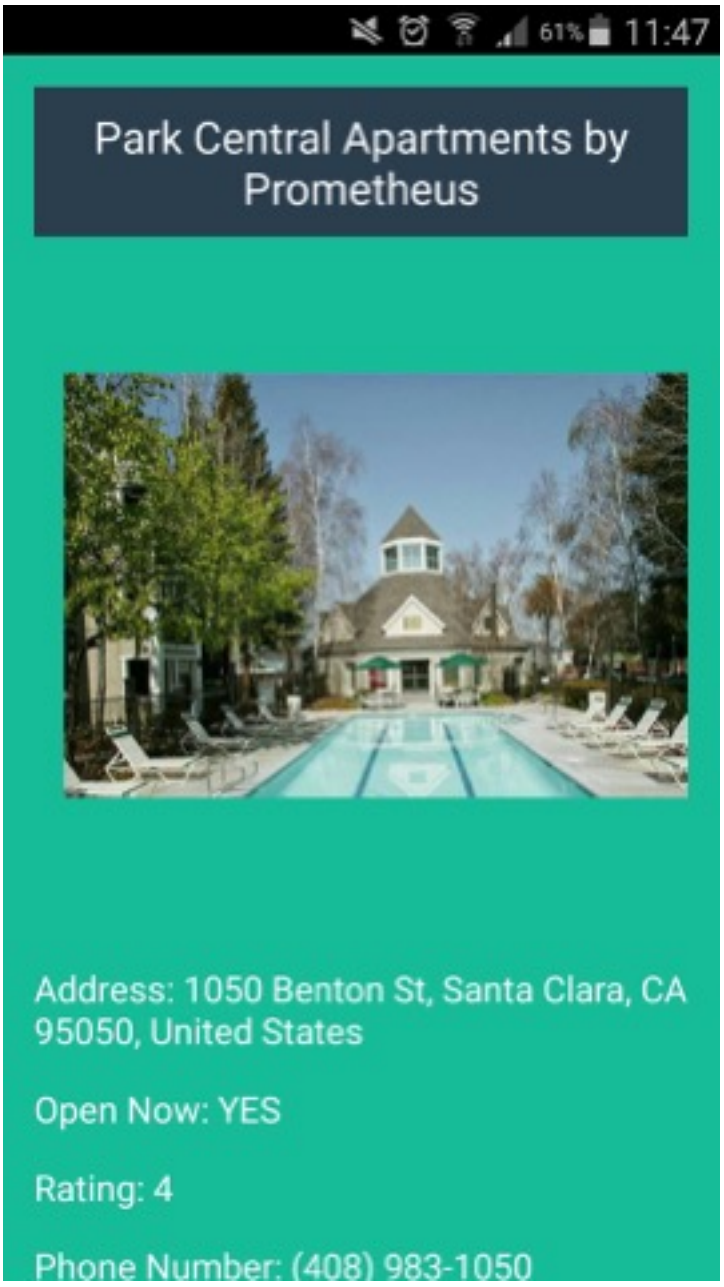


address. This is the ideal list anyone can get and find apartments around it to make a booking for any apartment.

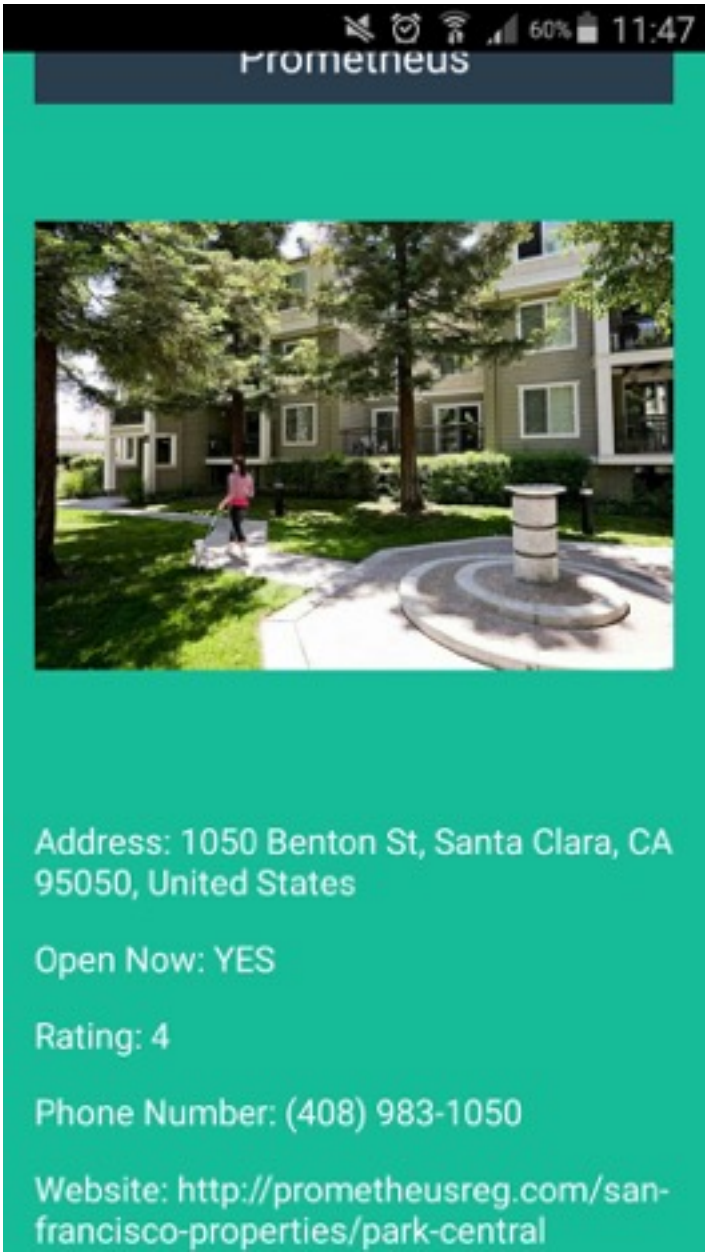
Individual Apartment

(display details)

Once you click on any of the apartment you would like to see, you can see various details about it viz. Name of the apartment,address of the apartment, Image of that apartment. We also have a galley showcase view to show if that apartment has multiple images for deeper user experience and research about that apartment.



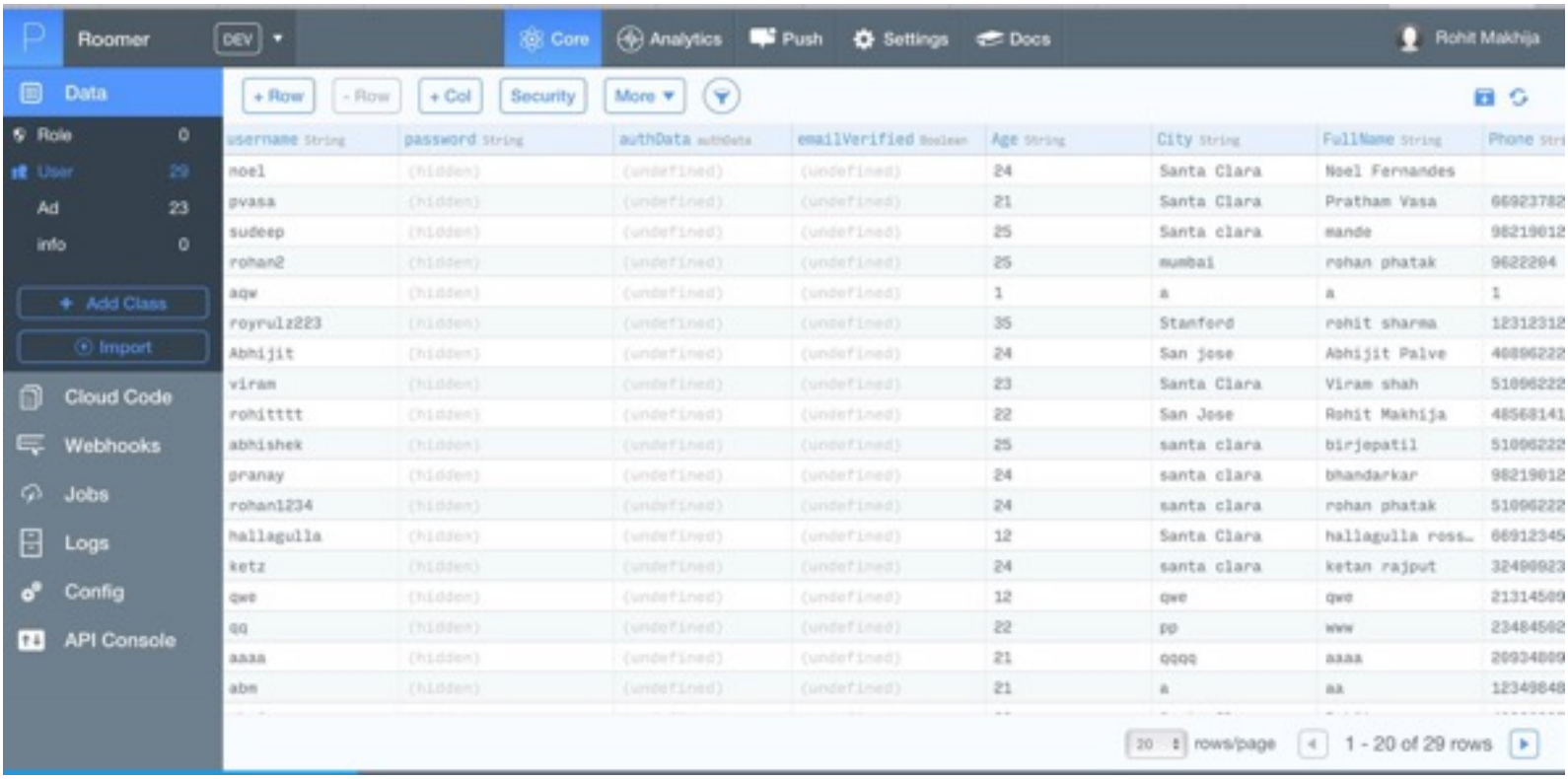
User can also see its rating and status if its open/closed. We have also given a phone number where the user can click and call the apartment.



Server

(Parse)

This is our Server(Parse). Here we have maintained a list of all the tables which we are using to run the app. This is the backend of our system. The sample values and table formation is being explained below.



API

(Google Places for android)

To get the apartments anywhere in the world

We can get data from the same database used by Google Maps and Google+ Local. Places features over 100 million businesses and points of interest that are updated frequently through owner-verified listings and user-moderated contributions.

References

(Major References)

1. https://parse.com/docs

2. http://developer.android.com/guide/index.html

3. http://www.vogella.com/tutorials/Android/article.html

4. http://code.tutsplus.com/articles/google-play-services-using-the-places-api--cms-23715

5. http://karnshah8890.blogspot.com/2013/03/google-places-api-tutorial.html

6. https://www.udacity.com/course/developing-android-apps--ud853

7. https://www.youtube.com/watch?v=m20n_GAsCtM

8. https://www.youtube.com/watch?v=QAbQgLGKd3Y&list=PL6gx4Cwl9DGBsvRxJJOzG4r4k_zLKrnxl

9. https://www.youtube.com/watch?v=mxYainSDHe8&list=PLzV8uWUcseN8x0c3q2hRx9X4vbLYSlipb

10. https://www.youtube.com/watch?v=hzrGAZnMOMQ&list=PLonJJ3BVjZW6hYgvtkaWvwAVvOFB7fkLa

11. https://www.youtube.com/playlist?list=PLB03EA9545DD188C3

12. https://developers.google.com/places/web-service search#PlaceSearchRequests