Android Homework-1 Report

NYU – Poly CS9033 Mobile Application Programming

Professor: Jeffrey Bickford <jbickford@nyu.edu>

Controller, Models and Views were the main focus of this homework which were implemented in the following way:

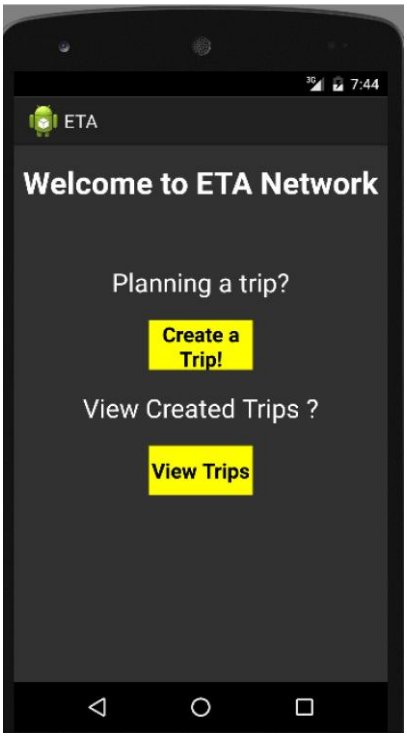
Views: The three views implemented are as follows:

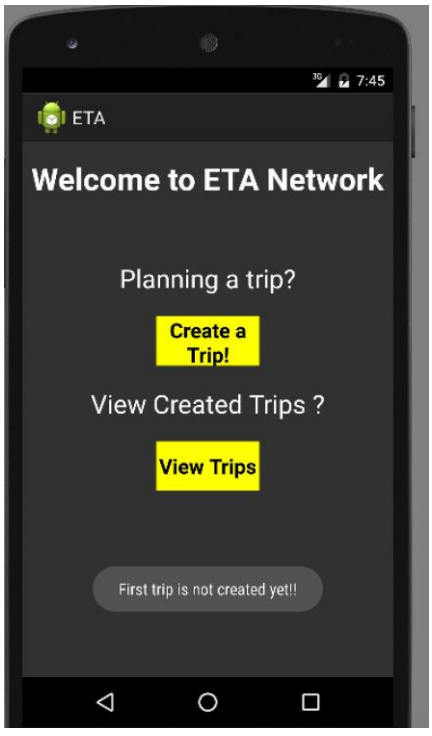
1. **activity_main**: Main activity of the app. Whenever the app is used via the launcher this activity gets displayed. This is the default page of the app where the finish () step in the cycles always finishes.

In screenshot below we can see there are three defaults texts and two relating buttons which would lead to the other two activities. The texts help us in selecting the options if using as a novice.

Create Trips Button: Navigates through the us to the activity_create_trip and helps creating a new trip by filling up the various details.

View Trip Button: Navigates us to the activity_view_trip and the user is able to see the latest trip details that has been saved by the user.





The mechanism of the activity is that user can create as many trips as he wants to when he clicks on the create trips , and the user is not able to see the latest trip if there are not trips created by then. User has to create a trip in order to view one.

2.**activity_create_trip**: User lands up on this activity whenever he clicks on the Create Trip button. We can fill in the details on the trip with various fields like :

Name of the trip:

Meeting Location:

Trip Destination:

Trip Time:

Trip Date:

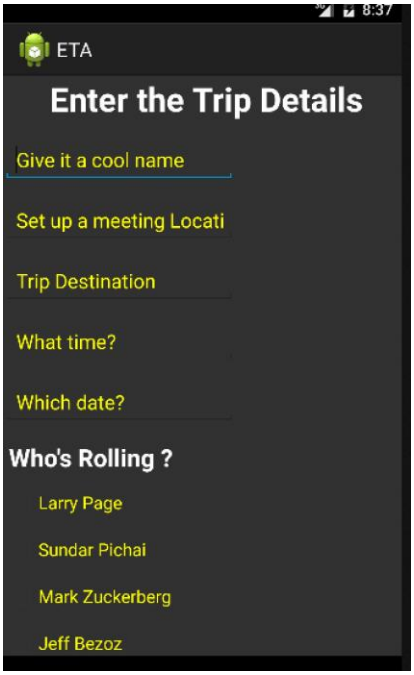
Attendees:

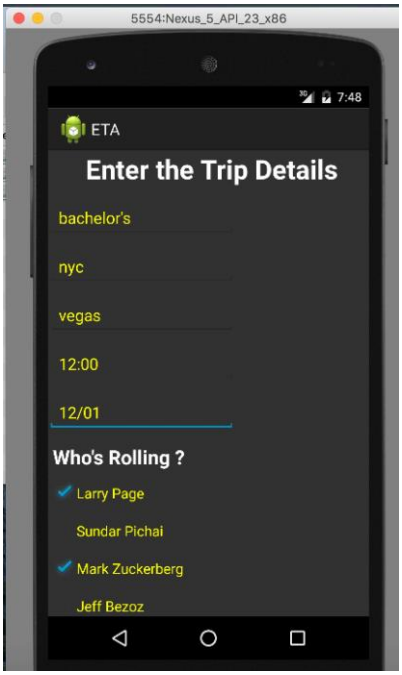
User is allowed to check mark certain number of friends with the help of check boxes he wants to invite for the trip from a list of the friends in this network and only those friends will be considered for the trip.

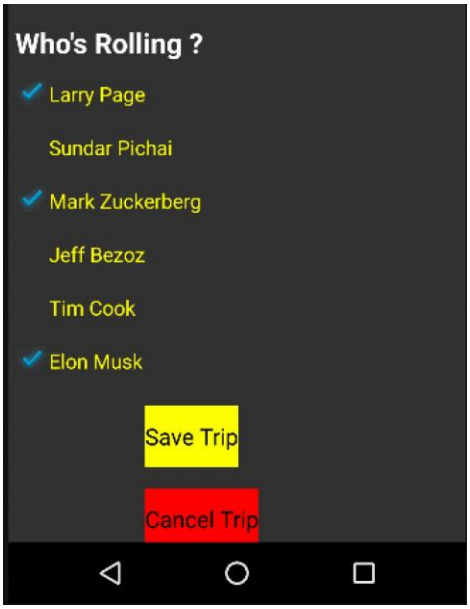
All the other fields have been given a hint name to help the user fill in the details for each field. Also the scroll view is enabled to view the every fields and the list of friends and the buttons at the end.

Save Trip Button: It saves the details of the trip and redirects back to the activity_main view and when the user clicks on the View Trip Button one is able to see the details entered in that.

Cancel Trip Button: It cancel the trips and does not save the details of the trip and redirects back to the activity_main view while the same can also happen with the back press from the phone.



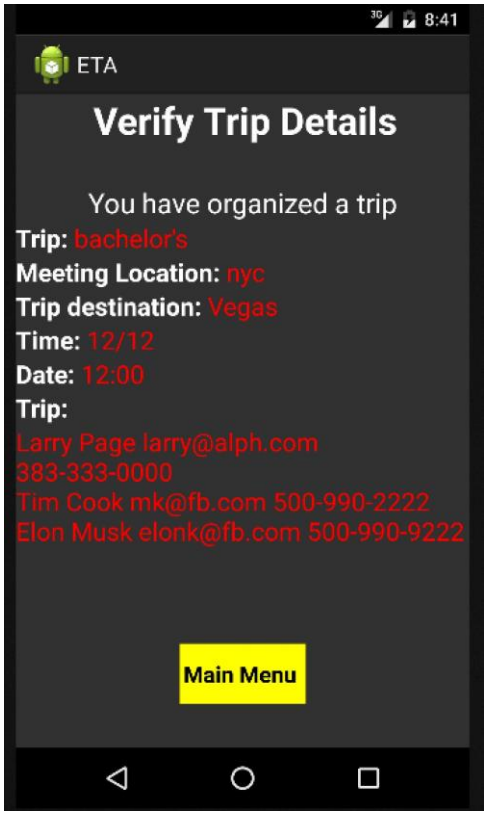




3. **activity_view_trip:** The user is able to view the details of the trip which he had created last. The different fields are displayed with the labels indicating what values signify what meaning.

Also the list of the friends that are attending the event are also listed along with the email addresses and the contact numbers for the same thus the user is able to contact them easily relating to his trips.

Main Menu Button : This button redirects to the activity_main when the user is finished viewing the details for the trip and thus it completes the entire flow of the app.



Models:

There are two models created as follows

Person: The details of the friends are extracted via this class and presently name, email and phone numbers of the classes are included in the information provided through them. This class implements parceable interface so that the objects can be serialized.

Trip: This model is created for storing the details of the trip to be created. The classvariables used are as follows: name, location, date, time, meetingLocation, friends1. This class implements the interface Parceable so that the object can be serialized.

Controllers: Following are the three controllers used:

MainActivity.java: In this controller we pass the control of the activity_main to the

activity_create_trip or activity_view_trip depending on the button pressed. When create trip button is pressed we use the startActivity method as we are expecting some value from the activity_create_trip.
While navigating to the activity_view_trip if we find that Trip is null then we pop a Toast saying that No trip is created. If Trip is not null we navigate to activity_view_trip.

CreateTripActivity.java: In this controller we create a Trip object from the values entered in the EditTexts on the view. If any value is missing then we pop a Toast that 'No field can be left blank'. If all fields are entered then we create the Trip object and pass it to the activity_main.

ViewTripActivity.java: In this controller we extract the Trip details from the trip object and display it on the desired fields.

Work continued for Homework-3

Classes used:

TripDatabaseHelper.java: Class for interacting with the internal database of the application. This class is added in a separate helper folder within the root directory for the activity classes and it extends the SQliteOpenHelper and is used to store all the information regarding the trips in the database supporting all the fields that were created for the model objects.

TripHistoryActivty.java: Class which is used to integrate the lists of the all the trips been saved and it displays all the trips saved so far retrieving it from the database. Each trip is clickable and by clicking on it, user navigates through the view_trip_activity where one can see the details of the fields created, the search location selected from the API and the trip members selected from the contacts app.

The upgrade uses the contact book in the phone to add the trip members invited for the trip and we can add more than more person for the same. Button "Add Members" was added to take the user to the Contacts application in the phone and we can then select's the multiple members one by one for the trip.

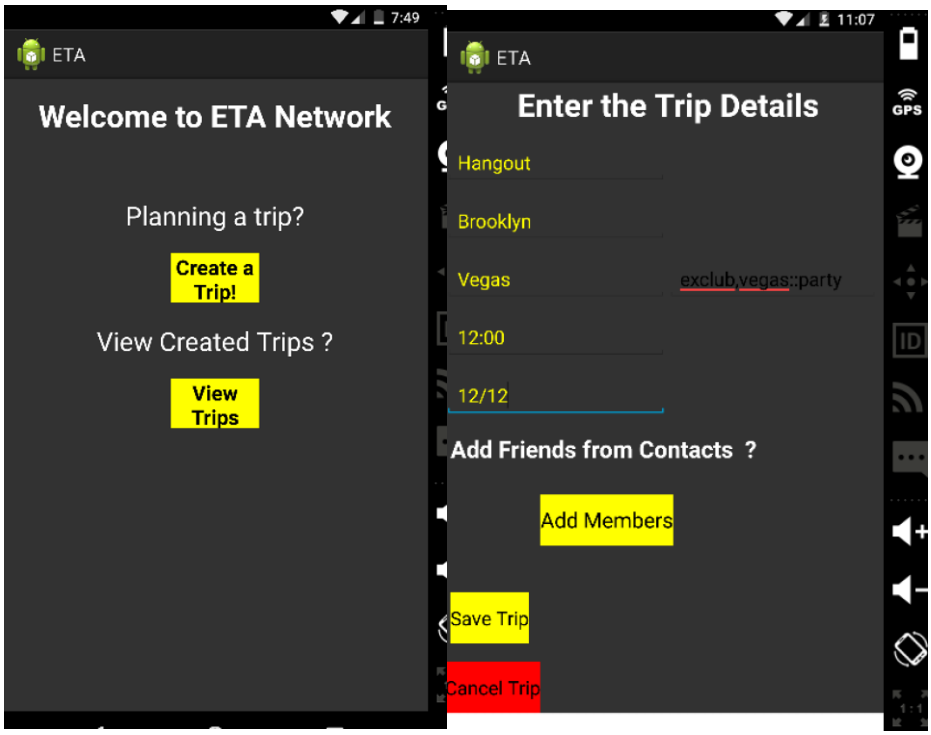
HW3API is installed in the emulator and is used to given results based on the searches made by the user to get the destination for the trip. Respective changes were made in the UI like adding a text area to enter in the location in the following format:

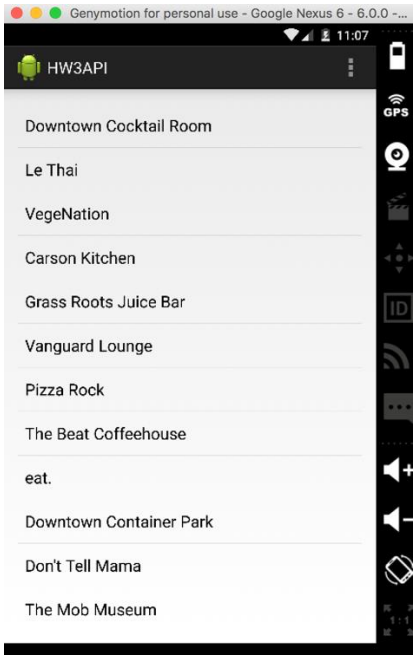
Vegas,club::party

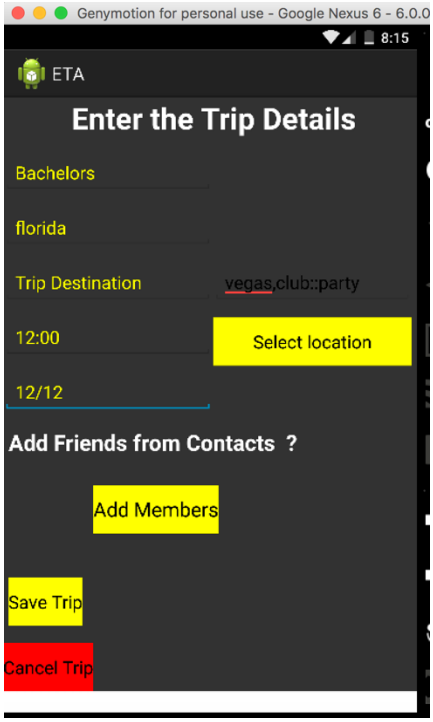
After entering the text in the text area user can click the “Search location “ Button and in the taken to the API app and user can then select the search option display in the list.

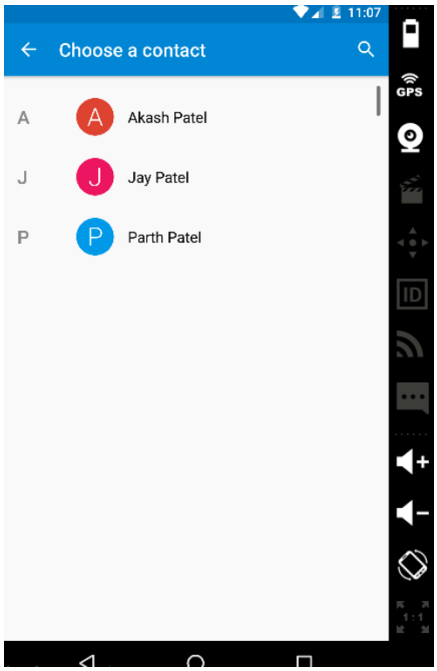
Two others changes were added in the UI- Trip History button in the view trip activity when clicked navigates back to the trip history activity and Main Menu button on the Trip History Activity whick navigates back to the Main activity.

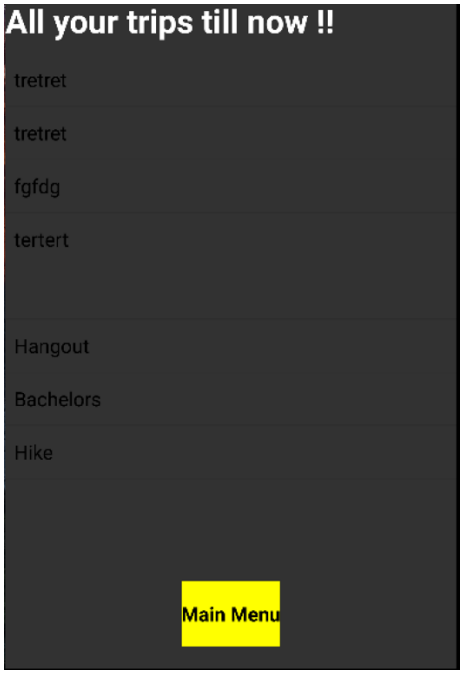
User Interface

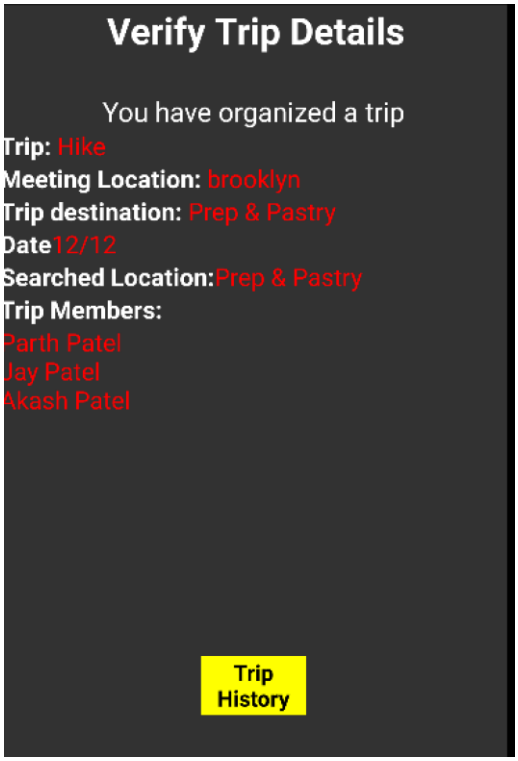



Genymotion for personal use - Google Nexus 6 - 6.0.0 -...
11:07
HW3API
Downtown Cocktail Room
Le Thai
VegeNation
Carson Kitchen
Grass Roots Juice Bar
Vanguard Lounge
Pizza Rock
The Beat Coffeehouse
eat.
Downtown Container Park
Don't Tell Mama
The Mob Museum
GPS


Genymotion for personal use - Google Nexus 6 - 6.0.0
8:15
ETA
Enter the Trip Details
Bachelors
florida
Trip Destination
vegas,club::party
12:00
Select location
12/12
Add Friends from Contacts ?
Add Members
Save Trip
Cancel Trip


11:07
Choose a contact
A
Akash Patel
J
Jay Patel
P
Parth Patel
GPS
ID
1:1


All your trips till now !!
tretret
tretret
fgfdg
tertert
Hangout
Bachelors
Hike
Main Menu



Models implement Parcelable interface. Android Parcelable implementation allows objects to read and write from Parcels, which can contain flattened data inside message containers.

If we want to convert a Java object into Parcelable, then the best way to do so is by implementing the Parcelable interface and overriding the writeToParcel() methods in its own class. The first step is to override the writeToParcel() method and write all object members into parcel objects.

The second is to create a static Parcelable.Creator object to de-serialize the Java object.

Controllers use getParcelableExtra method to retrieve objects of class that implement Parcelable interface.