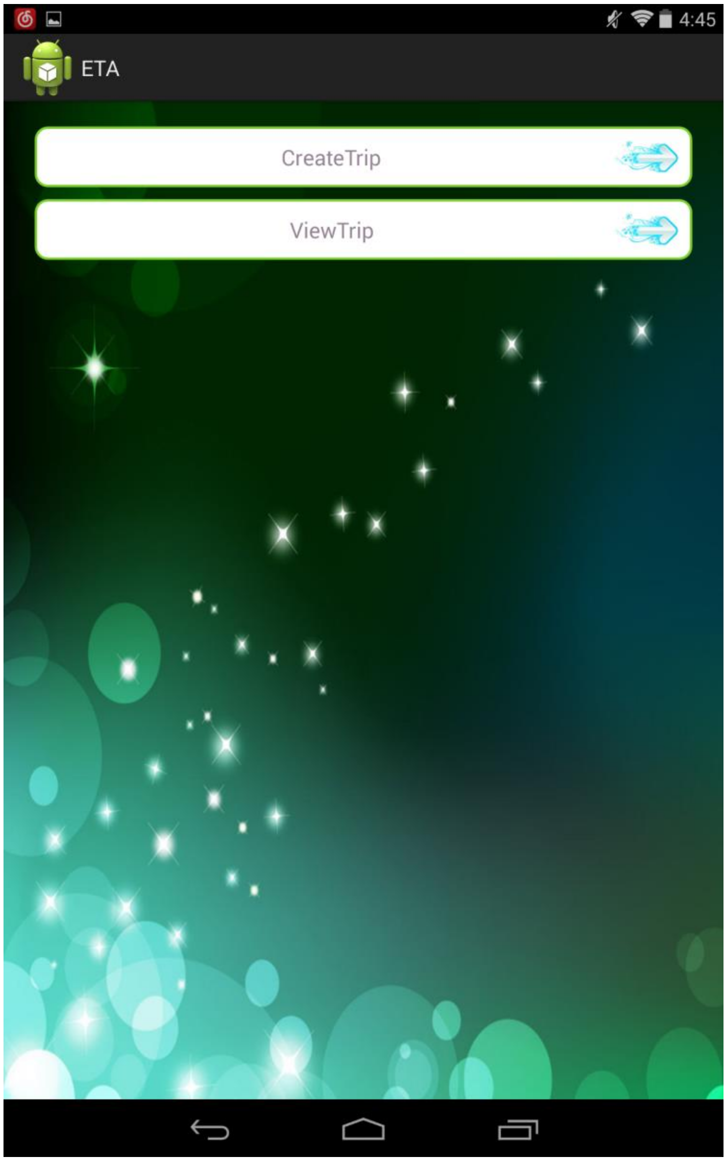
EDT- HOMEWORK 3

THE MAIN ACTIVITY:

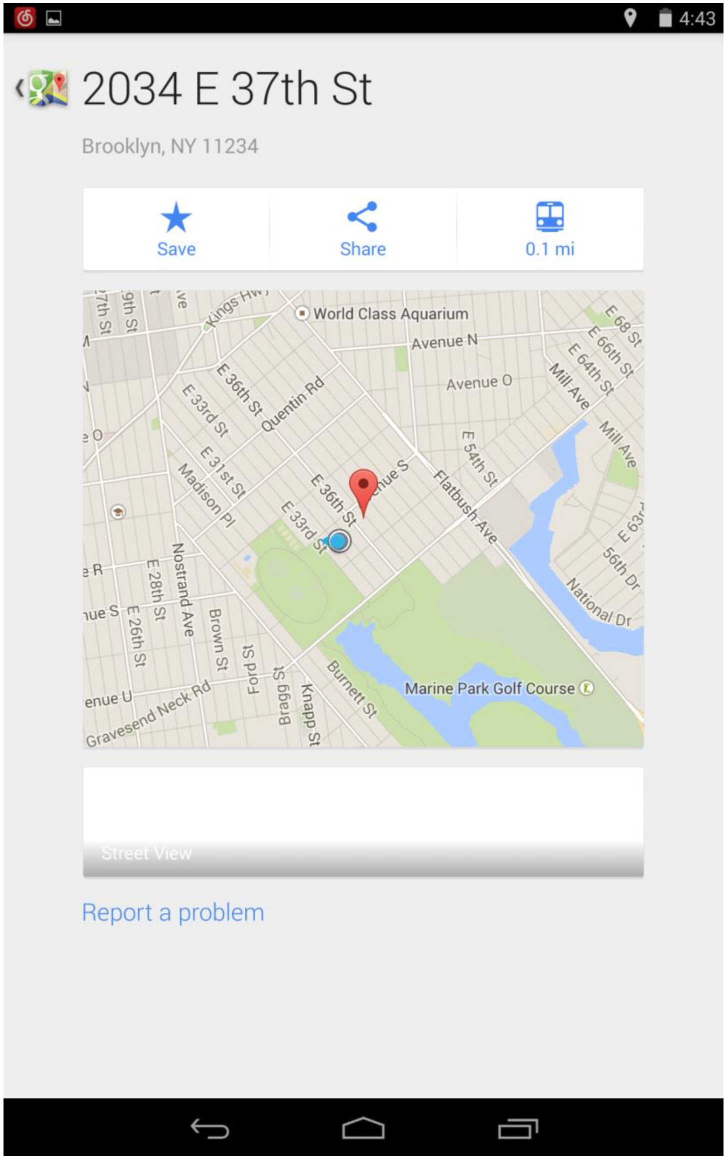


GOOGLE MAP FUNCTION:

It used to get address infromation from Google Map. The address information will be used as the trip location to build a new trip.

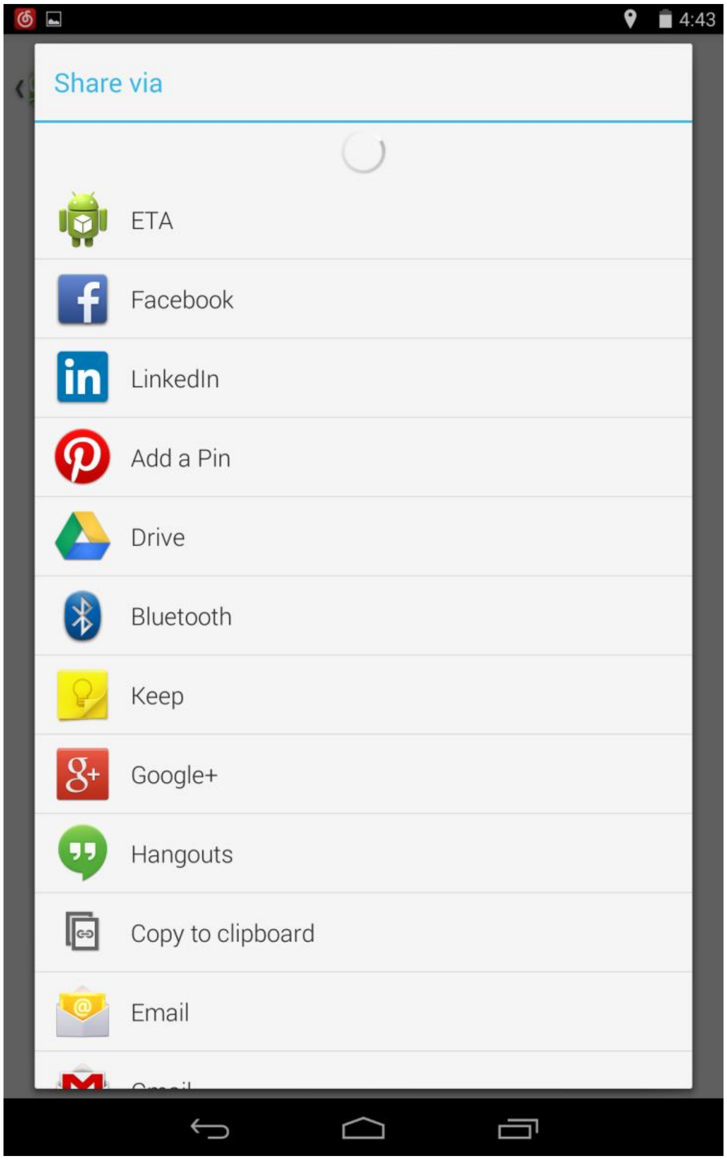
OPERATION:

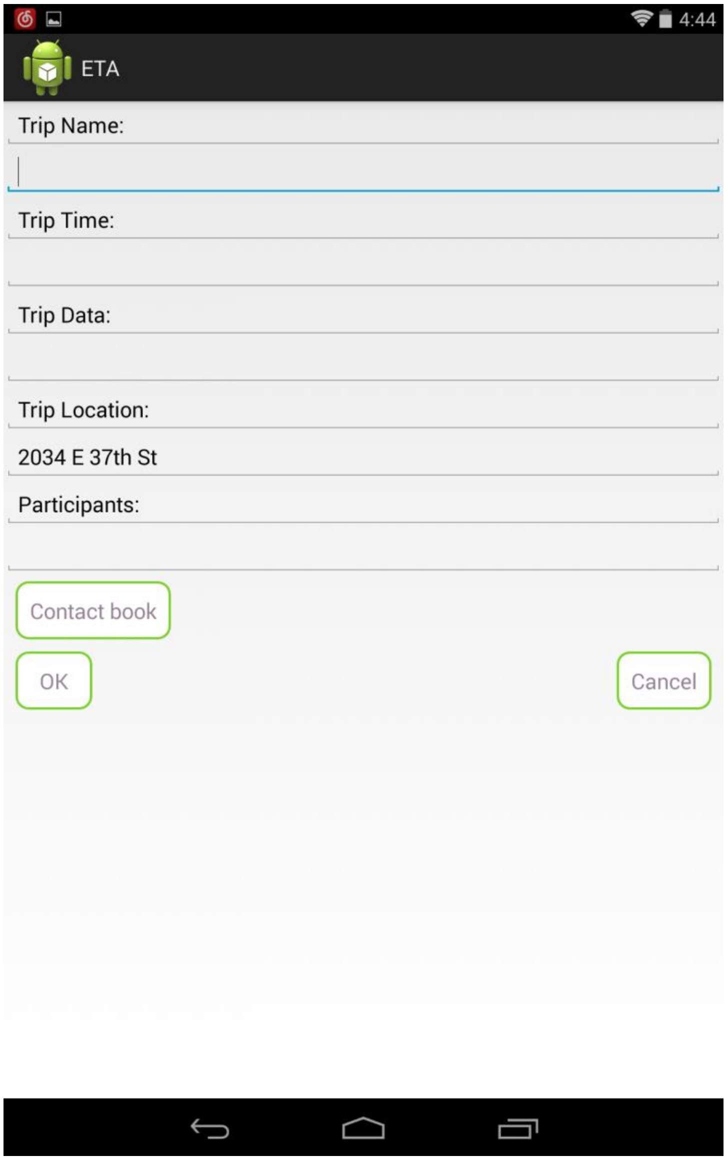
Find place in Google Map -> share the address -> the createtrip activity of ETA will be open -> the address from Google Map will be filled the trip location textbox



2034 E 37th St

Brooklyn, NY 11234


4:43
Share via
ETA
Facebook
LinkedIn
Add a Pin
Drive
Bluetooth
Keep
Google+
Hangouts
Copy to clipboard
Email
Gmail



Trip Name:

Trip Time:

Trip Data:

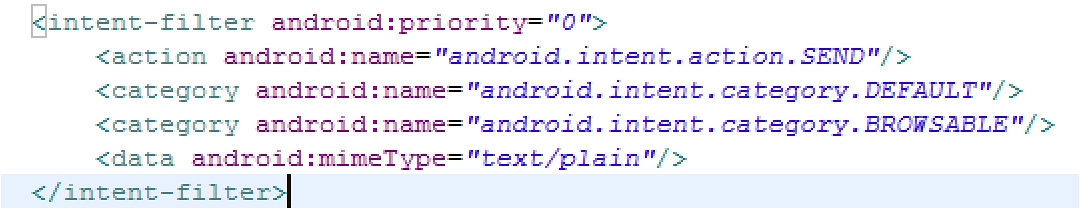
Trip Location:

2034 E 37th St

Participants:

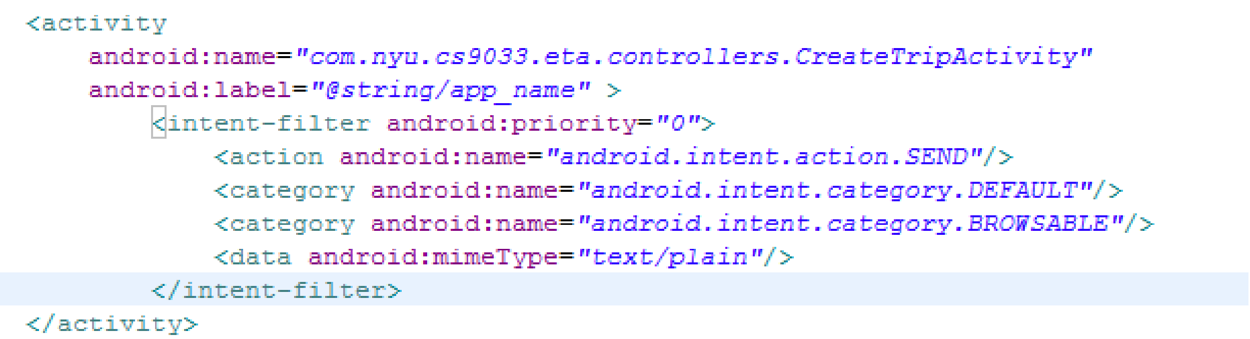
IMPLEMENTATION:

1. I and the intent filter to the "AndroidManifest.xml".



```
<intent-filter android:priority="0">
    <action android:name="android.intent.action.SEND"/>
    <category android:name="android.intent.category.DEFAULT"/>
    <category android:name="android.intent.category.BROWSABLE"/>
    <data android:mimeType="text/plain"/>
</intent-filter>
```

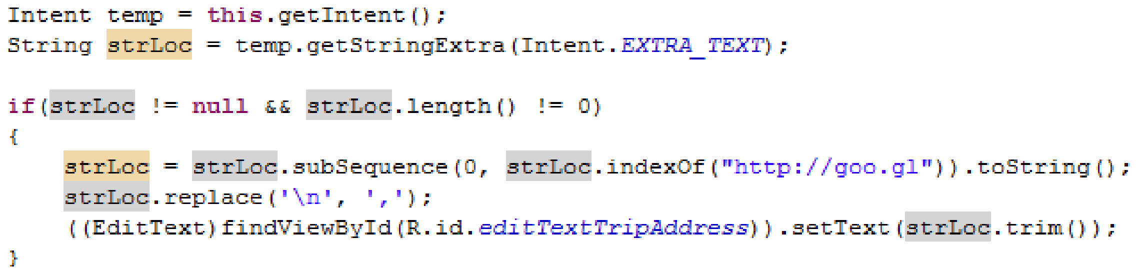
This intent filter should be add to the creativeTripActivity part. It's code like this:



```xml
<activity
    android:name="com.nyu.cs9033.eta.controllers.CreateTripActivity"
    android:label="@string/app_name" >
        <intent-filter android:priority="0">
            <action android:name="android.intent.action.SEND"/>
            <category android:name="android.intent.category.DEFAULT"/>
            <category android:name="android.intent.category.BROWSABLE"/>
            <data android:mimeType="text/plain"/>
        </intent-filter>
</activity>
```

2. Deal the information in the “createTripActivity” class.

   I add some code to the OnCreate() function to deal the information from google map. The following code will extract information from the intent and then set the location editText in the User Interface.



```java
Intent temp = this.getIntent();
String strLoc = temp.getStringExtra(Intent.EXTRA_TEXT);

if(strLoc != null && strLoc.length() != 0)
{
    strLoc = strLoc.subSequence(0, strLoc.indexOf("http://goo.gl")).toString();
    strLoc.replace('\n', ',');
    ((EditText)findViewById(R.id.editTextTripAddress)).setText(strLoc.trim());
}
```

CONTACT FUNCTION:

We can use this function to get people name from contact book. Then the people name will be used as the participants.

OPERATION:

Start the createtrip Activity -> click “contact” button, the contact book will be shown -> pick one person in the contact book -> the person name will be add to the participant and the contact book activity will be destroyed -> we need more participants, then click “contact” button again.

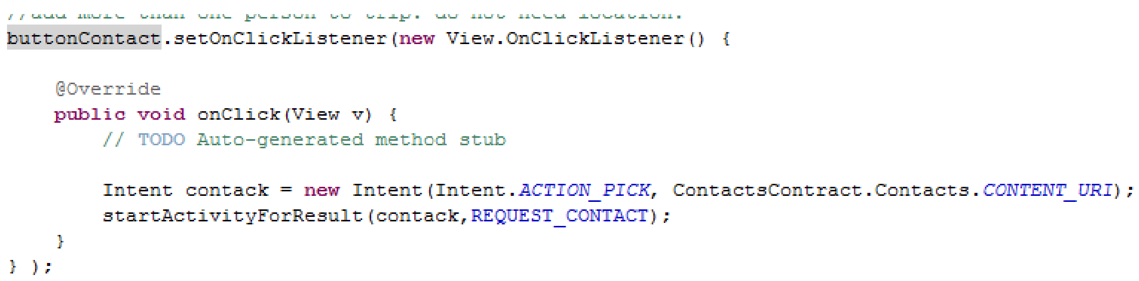
IMPLEMENTATION:

VIEW PART:

I add one button to the createTrip Activity. Add the xml code in create_trip.xml file

CONTROL PART:

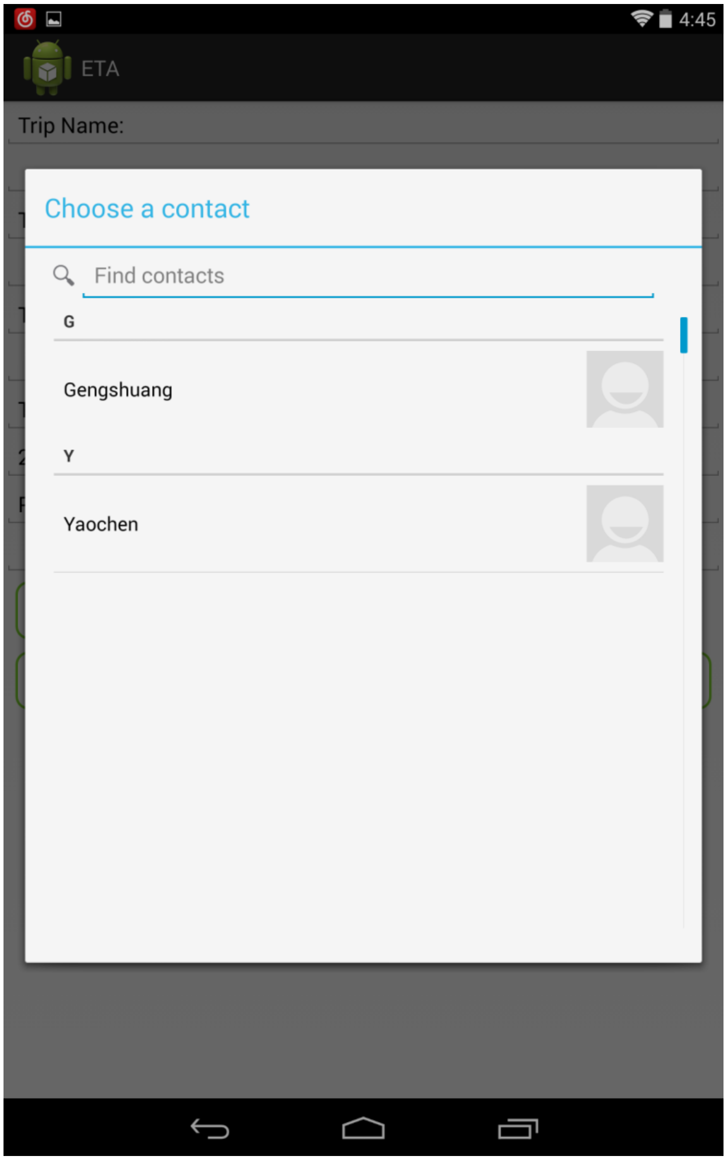
Add the click event of button in OnCreate() function of the CreateTripActivity class. In this part, I create an intent to get the contact information from contact book and then start the contact book activity.



```
//add more than one person to trip. do not need location.
buttonContact.setOnClickListener(new View.OnClickListener() {

    @Override
    public void onClick(View v) {
        // TODO Auto-generated method stub

        Intent contack = new Intent(Intent.ACTION_PICK, ContactsContract.Contacts.CONTENT_URI);
        startActivityForResult(contack,REQUEST_CONTACT);
    }
} );
```



Choose a contact

Find contacts

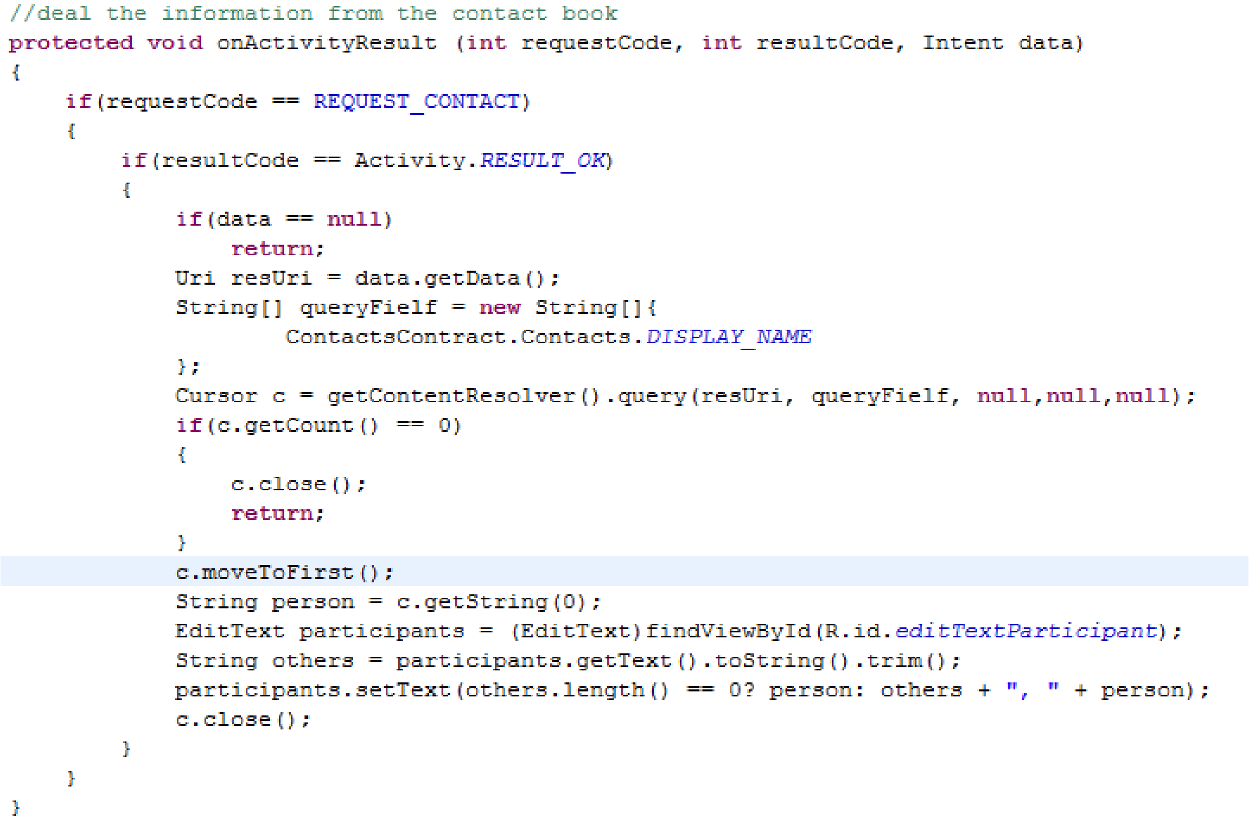
G

Gengshuang

Y

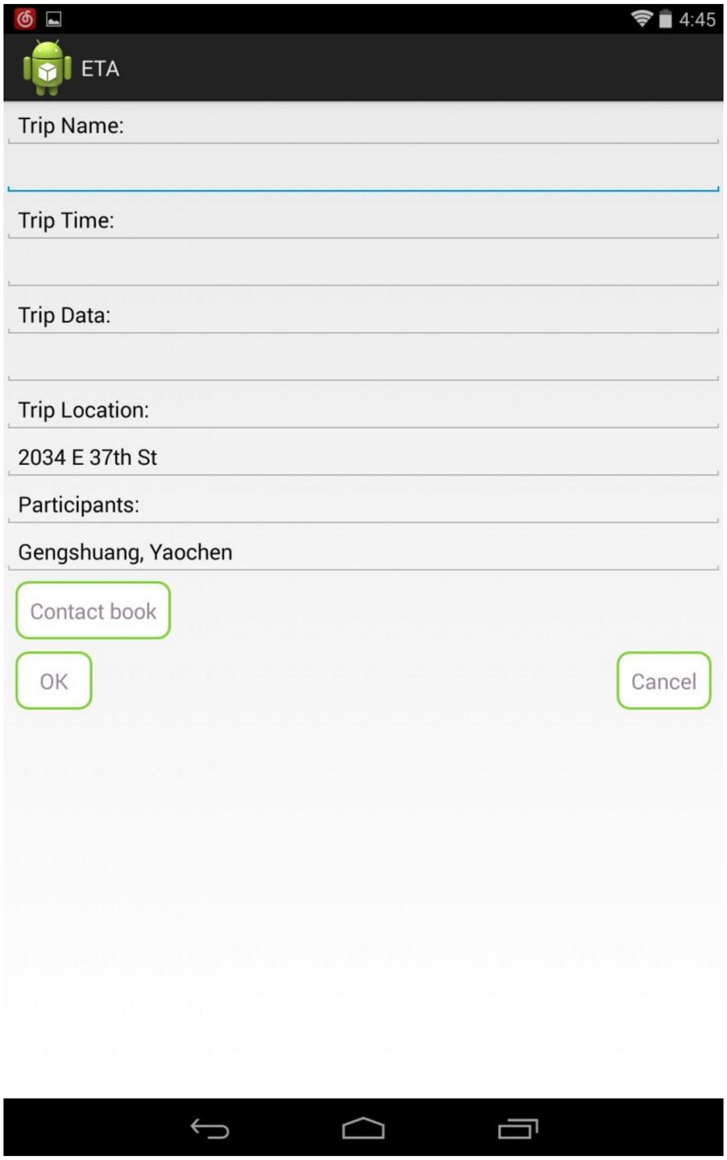
Yaochen

I add onActivityResult() function to deal with the information, which is send back from contact book



```
//deal the information from the contact book
protected void onActivityResult (int requestCode, int resultCode, Intent data)
{
    if(requestCode == REQUEST_CONTACT)
    {
        if(resultCode == Activity.RESULT_OK)
        {
            if(data == null)
                return;
            Uri resUri = data.getData();
            String[] queryFielf = new String[]{
                    ContactsContract.Contacts.DISPLAY_NAME
            };
            Cursor c = getContentResolver().query(resUri, queryFielf, null,null,null);
            if(c.getCount() == 0)
            {
                c.close();
                return;
            }
            c.moveToFirst();
            String person = c.getString(0);
            EditText participants = (EditText)findViewById(R.id.editTextParticipant);
            String others = participants.getText().toString().trim();
            participants.setText(others.length() == 0? person: others + ", " + person);
            c.close();
        }
    }
}
```

Then I set the participants editText component with the contact information from the contact book.



Trip Name:

Trip Time:

Trip Data:

Trip Location:

2034 E 37th St

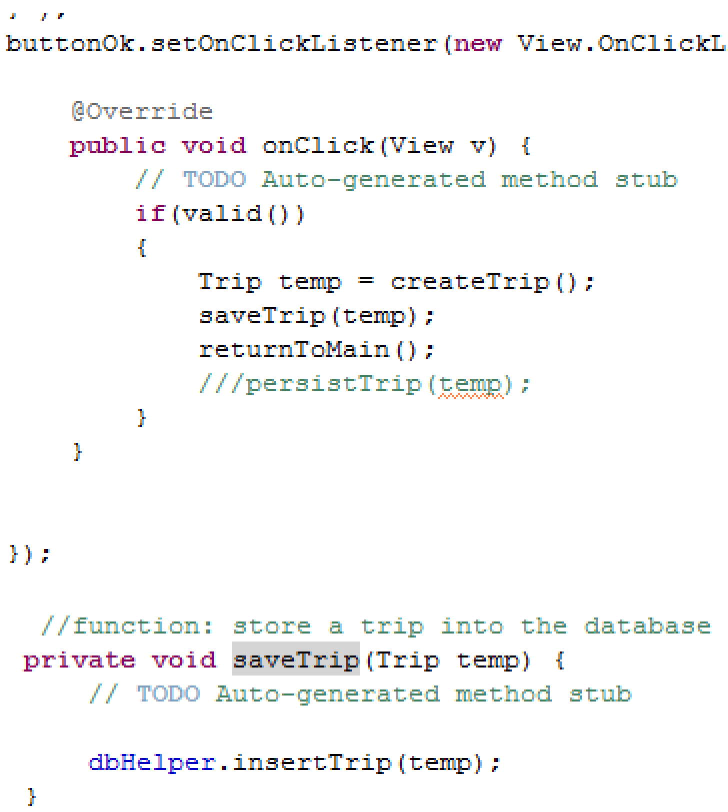
Participants:

Gengshuang, Yaochen

After the user edit the information of the trip and click the “OK” button, I store the information into the SQLite database. To implement this function, I open the database in the Oncreate() function. TripDatabaseHelper is the database module. I will explain in next part.

```
// TODO - fill in here
//open database
dbHelper = new TripDatabaseHelper(this);
```

and then save the data on the click event of button “OK”.



```
buttonOk.setOnClickListener(new View.OnClickListener() {

    @Override
    public void onClick(View v) {
        // TODO Auto-generated method stub
        if(valid())
        {
            Trip temp = createTrip();
            saveTrip(temp);
            returnToMain();
            ///persistTrip(temp);
        }
    }


});

  //function: store a trip into the database
 private void saveTrip(Trip temp) {
     // TODO Auto-generated method stub

     dbHelper.insertTrip(temp);
 }
```

SQLITE MODULE:

I use this module to store the trip information into SQLite database.

I create by module TripDatabaseHelper extends from the ‘SQLiteOpenHelper’

OnCreate(): create trip table and is called by System when the database initialize.

OnUpgrage: create new table and delete the old table

insertTrip(): insert one trip information into the database. When we create one trip on the createTrip activity, we will call this function.

getAllTrip(): It will query the database and return all trip information using a List<Trip> instantce.

There is one database named “trips” and one table named “trip”. There are six columns in the trip table, including id, name, date ,time ,location, participants

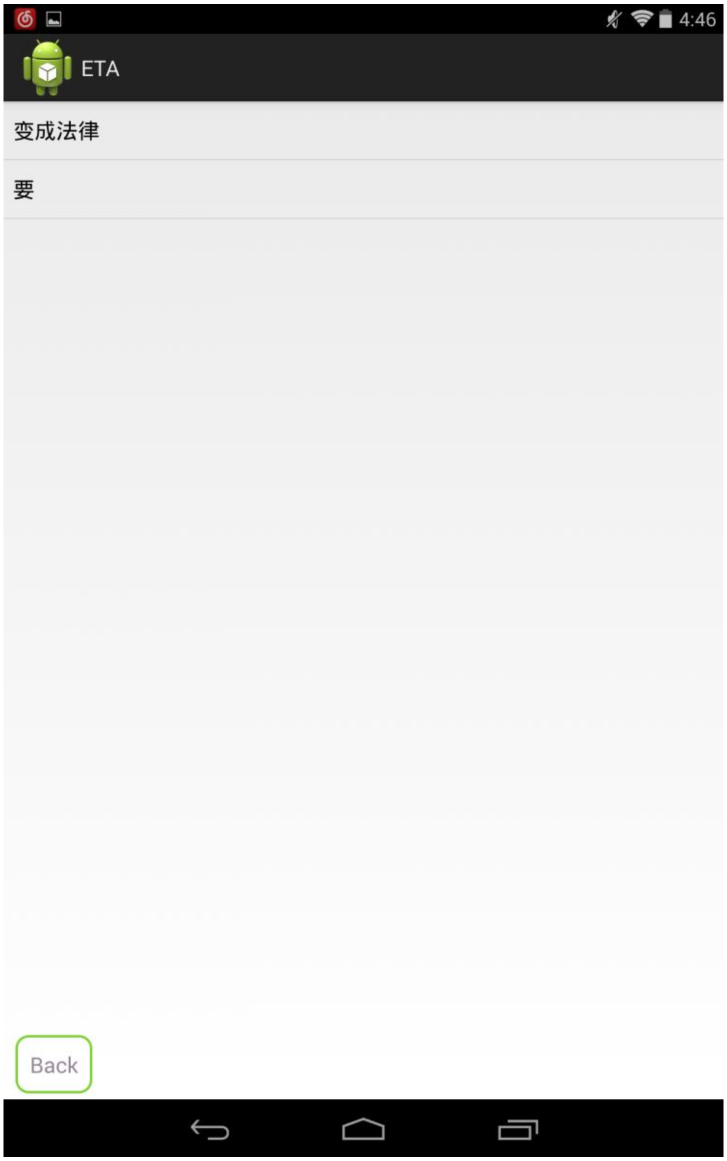
TRIPHISTORYACTIVITY MODULE:

This module is used to show the trip List stored in database.

OnCreate(): initialize List on the TripHistoryActivity with the trip information from database. If there is no trip in the database, show the message dialog to tell user: “There is no trip”

ViewTrip(): Start the ViewTripActivity to show the detail information of the selected trip

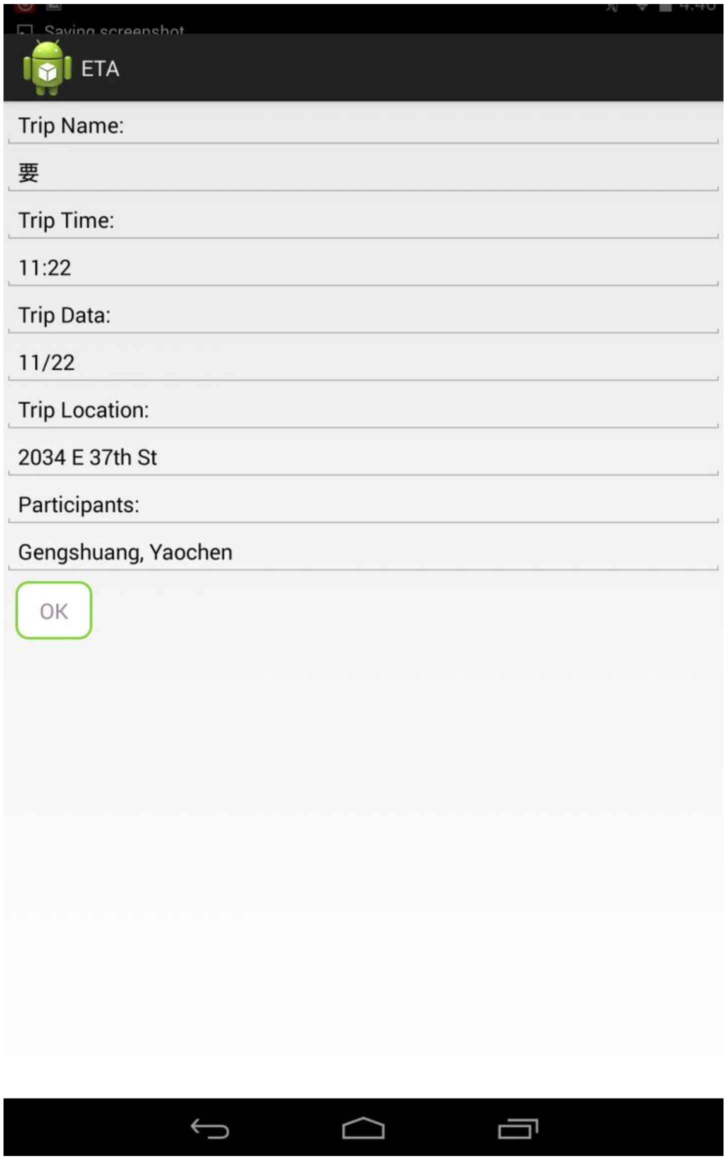
returnToMain(); destroy current activity and back to the main activity.



变成法律

要

If we select one item in the List, we can get the detail information of selected trip.



Trip Name:

要

Trip Time:

11:22

Trip Data:

11/22

Trip Location:

2034 E 37th St

Participants:

Gengshuang, Yaochen