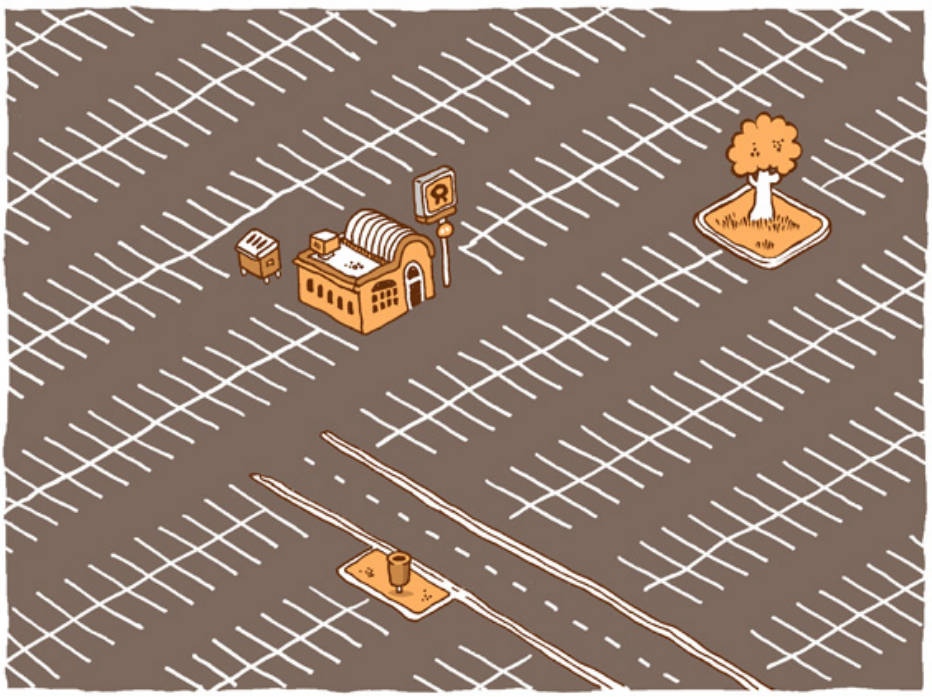
Find Vehicle (by Jasmine Song, Dec.2015)

Help users find where they park their vehicles.



Functionality

5 Activities + Database

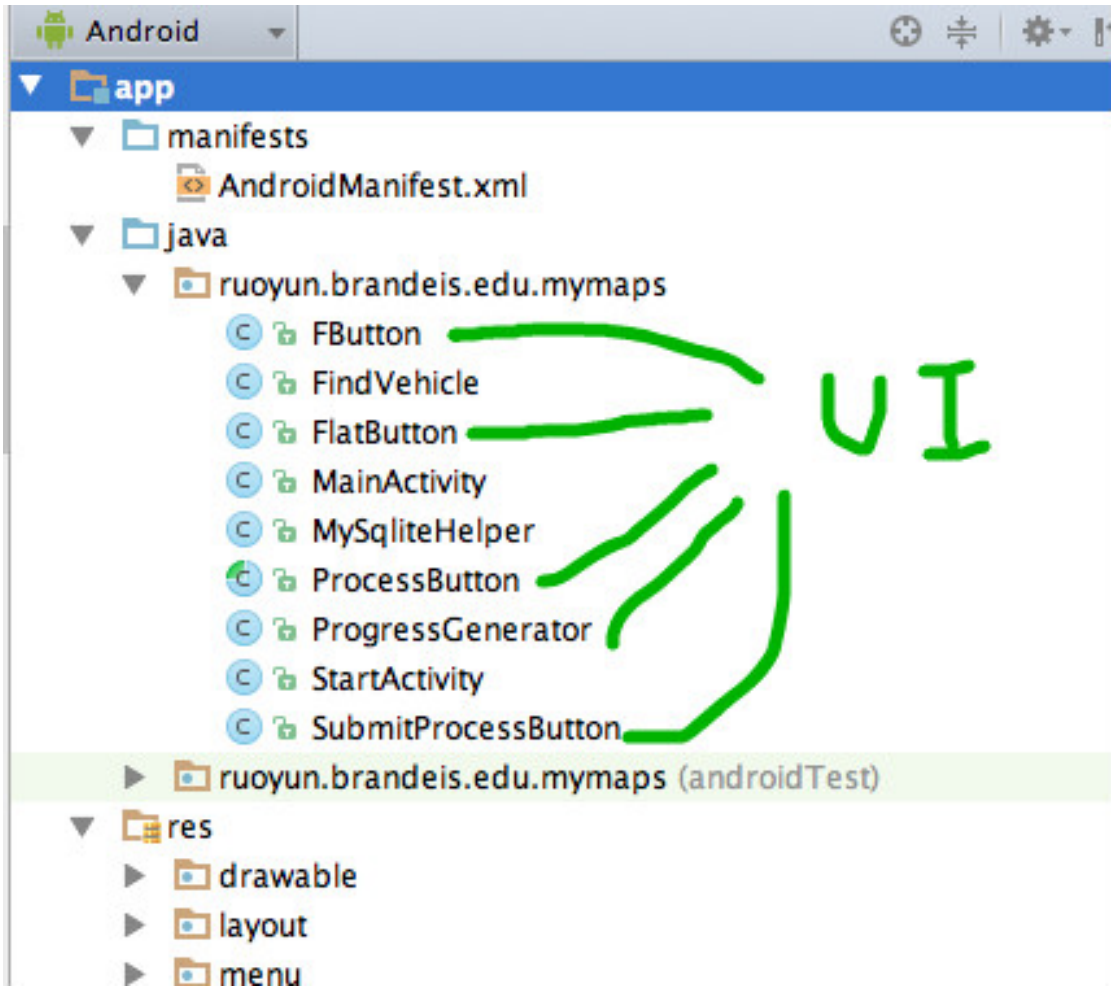
Start Activity ---- for users to choose indoor or outdoor usage situation

Main Activity （indoor&outdoor）---- for users to add a marker on the point of their parking location, either user current location or any searched location

FindVehicle Activity （indoor&outdoor）---- Map camera will focus on the spot exactly between user and parking location. It will guide users walk all the way to their parking location by dynamically tracking user's movement. Users will see on map they are getting closer and closer to their vehicles.

MySqliteHelper ---- Keep record of the latitudes and longitudes of every spot user places a car, delete the older one when the newer one is placed. If accidentally exiting the application or application going background, next time user starts the app, the location of the vehicle will still be there.

User Interface



Code Worth Mentioned

Snippet I.

Create Google API client

```
if (initMap()) {
    //mMap.setMyLocationEnabled(true); consumes battery power
    mLocationClient = new GoogleApiClient.Builder(this)
            .addApi(LocationServices.API)
            .addConnectionCallbacks(this)
            .addOnConnectionFailedListener(this)
            .build();      //create client object

    mLocationClient.connect();

} else {
    Toast.makeText(this, "Map is not connected!", Toast.LENGTH_LONG).show();
}
```

Get Current Location

```
public void showCurrentLocation(MenuItem item) {
    Location currentLocation = LocationServices.FusedLocationApi
            .getLastLocation(mLocationClient);
    if (currentLocation == null) {
        Toast.makeText(this, "Couldn't connect", Toast.LENGTH_SHORT).show();
    } else {
        LatLng latLng = new LatLng(currentLocation.getLatitude(), currentLocation.getLongitude());
        CameraUpdate update = CameraUpdateFactory.newLatLngZoom(latLng, 18);
        mMap.animateCamera(update);

        addFlashMarker(latLng);

    }
}
```

Snippet 2

```
private void addFlashMarker(LatLng latLng){

    MarkerOptions options = new MarkerOptions()

            .position(latLng)
            .draggable(true)
            .icon(BitmapDescriptorFactory.fromResource(R.mipmap.ic_flash_client));
    if (client_marker != null) {
        client_marker.remove();
    }
    client_marker = mMap.addMarker(options);

    Timer markertimer = new Timer();
    markertimer.scheduleAtFixedRate(() → {
            runOnUiThread(() → { blinkMarker(); });
    }, 0, 500);
}

//make the marker blink
private boolean blinkMarker(){

    if(client_marker.isVisible()){
        client_marker.setVisible(false);
    }
    else if(!client_marker.isVisible()){
        client_marker.setVisible(true);
    }
    return client_marker.isVisible();
}
```

Snippet 3

When going to FindVecle achitivity ， camera points at the middle point of User and his car.

```
@Override
public void onConnected(Bundle bundle) {
    //Toast.makeText(this, "Ready to find vehicle!", Toast.LENGTH_LONG).show();
    Location currentLocation = LocationServices.FusedLocationApi
            .getLastLocation(mLocationClient);

    LatLng latLngClient = new LatLng(currentLocation.getLatitude(), currentLocation.getLongitude());
    //LatLng latLngClient = new LatLng(0.0,0.0);
    addClientMarker(latLngClient);
    addVehicleMarker(latLng);

    LatLng cameraLatLng = new LatLng(0.5 * (latLng.latitude + latLngClient.latitude),
            0.5 * (latLng.longitude + latLngClient.longitude));
    Toast.makeText(this, "mLocationClient is connected!", Toast.LENGTH_LONG).show();

    //gotolocation(cameraLatLng,18);
    gotolocation(cameraLatLng, 16);
    Toast.makeText(this, "mLocationClient is connected!" + latLngClient + "----" + cameraLatLng, Toast.LENGTH_LONG).show();
```

Track user movement

```
        Location location = LocationServices.FusedLocationApi.getLastLocation(mLocationClient);
        if (location == null) {
            Toast.makeText(this,"Location null!",Toast.LENGTH_LONG).show();

            //LocationServices.FusedLocationApi.requestLocationUpdates(mLocationClient, mLocationRequest, this);
        } else {
            Toast.makeText(this,"Location not null!",Toast.LENGTH_LONG).show();

            // Register the listener with the Location Manager to receive location updates
            LocationServices.FusedLocationApi.requestLocationUpdates(mLocationClient, mLocationRequest, this);

            handleNewLocation(location);
        }
    }

    private void handleNewLocation(Location location) {
        Toast.makeText(this,"handle New!"+ this.toastID++,Toast.LENGTH_LONG).show();


        double currentLatitude = location.getLatitude();
        double currentLongitude = location.getLongitude();
        LatLng latLng = new LatLng(currentLatitude, currentLongitude);

        addClientMarker(latLng);
    }
```

Snippet 4

Geocoder API can search by location name, zip code, landscape (buildings, etc) and support input error-modification.

```
public void geoLocate(View v) throws IOException {
    hideSoftKeyboard(v);

    Toast.makeText(this, "geolocate!!!", Toast.LENGTH_LONG).show();

    String searchString = tv.getText().toString();

    Geocoder gc = new Geocoder(this);
    List<Address> list = gc.getFromLocationName(searchString, 1);

    if (list.size() > 0) {
        Address add = list.get(0);
        String locality = add.getLocality();
        Toast.makeText(this, "Found:" + locality, Toast.LENGTH_SHORT).show();

        double lat = add.getLatitude();
        double lng = add.getLongitude();
        gotolocation(lat, lng, 15);

        //add vehicle marker automatically
        if (marker != null) {
            marker.remove();
        }
        MarkerOptions options = new MarkerOptions()
                .title(locality)
                .position(new LatLng(lat, lng))
                .draggable(true)
                .icon(BitmapDescriptorFactory.fromResource(R.mipmap.ic_pink_car));
        marker = mMap.addMarker(options);

    }
}
```

Snippet 5

```java
package ruoyun.brandeis.edu.mymaps;

/**
 * Created by reallifejasmine on 11/16/15.
 */

import ...

public class MySqliteHelper extends SQLiteOpenHelper {

    // Database Version
    private static final int DATABASE_VERSION = 1;
    // Database Name
    private static final String DATABASE_NAME = "ParkLocation";

    public MySqliteHelper(Context context) {
        super(context, DATABASE_NAME, null, DATABASE_VERSION);
    }

    @Override
    public void onCreate(SQLiteDatabase db) {
        // SQL statement to create  table
        //modifide by, add indoor level information
        String CREATE_LOCATION_TABLE = "CREATE TABLE IF NOT EXISTS newlocations ( " +
                "id INTEGER PRIMARY KEY AUTOINCREMENT, " +
                "lat TEXT , "+
                "long TEXT , "+
                "level TEXT )";

        // create table
        db.execSQL(CREATE_LOCATION_TABLE);
    }
```