

# Android

Camera & Map

# Open Camera

```
Intent intent = new Intent(Intent.ACTION_PICK);
```

```
intent.setType("image/*");
```

```
startActivityForResult(
```

```
    Intent.createChooser(intent, "Choose App"), 1
```

```
);
```

# Result

@Override

```
public void onActivityResult(int requestCode, int resultCode,  
Intent data) {
```

```
    super.onActivityResult(requestCode, resultCode, data);
```

```
}
```

```
<!--Photo-->  
<ImageView  
    android:scaleType="centerInside"  
    android:id="@+id/imvPhoto" ←  
    android:layout_gravity="center"  
    android:src="@drawable/photo"  
    android:layout_width="wrap_content"  
    android:layout_height="200dp" />
```

# ImageView

# Display

```
try {  
  
    Uri uri = data.getData();  
  
    Bitmap bitmap = BitmapFactory.decodeStream(  
  
        getActivity()  
  
            .getContentResolver()  
  
            .openInputStream(uri)  
  
        );  
  
    Bitmap bitmap1 = Bitmap.createScaledBitmap(bitmap, 800, 600, true);  
  
    getView().findViewById(R.id.invPhoto).setImageBitmap(bitmap1);  
  
} catch (Exception e) {  
  
    e.printStackTrace();  
  
}
```

```

@Override
public void onActivityResult(int requestCode, int resultCode, Intent data) {
    super.onActivityResult(requestCode, resultCode, data);
    if (resultCode == getActivity().RESULT_OK) {
        uri = data.getData();
        aBoolean = false;
        try {
            Bitmap bitmap = BitmapFactory.decodeStream(
                getActivity()
                    .getContentResolver()
                    .openInputStream(uri)
            );

            Bitmap bitmap1 = Bitmap.createScaledBitmap(bitmap, 800, 600, true);
            imageView.setImageBitmap(bitmap1);

        } catch (Exception e) {
            e.printStackTrace();
        }

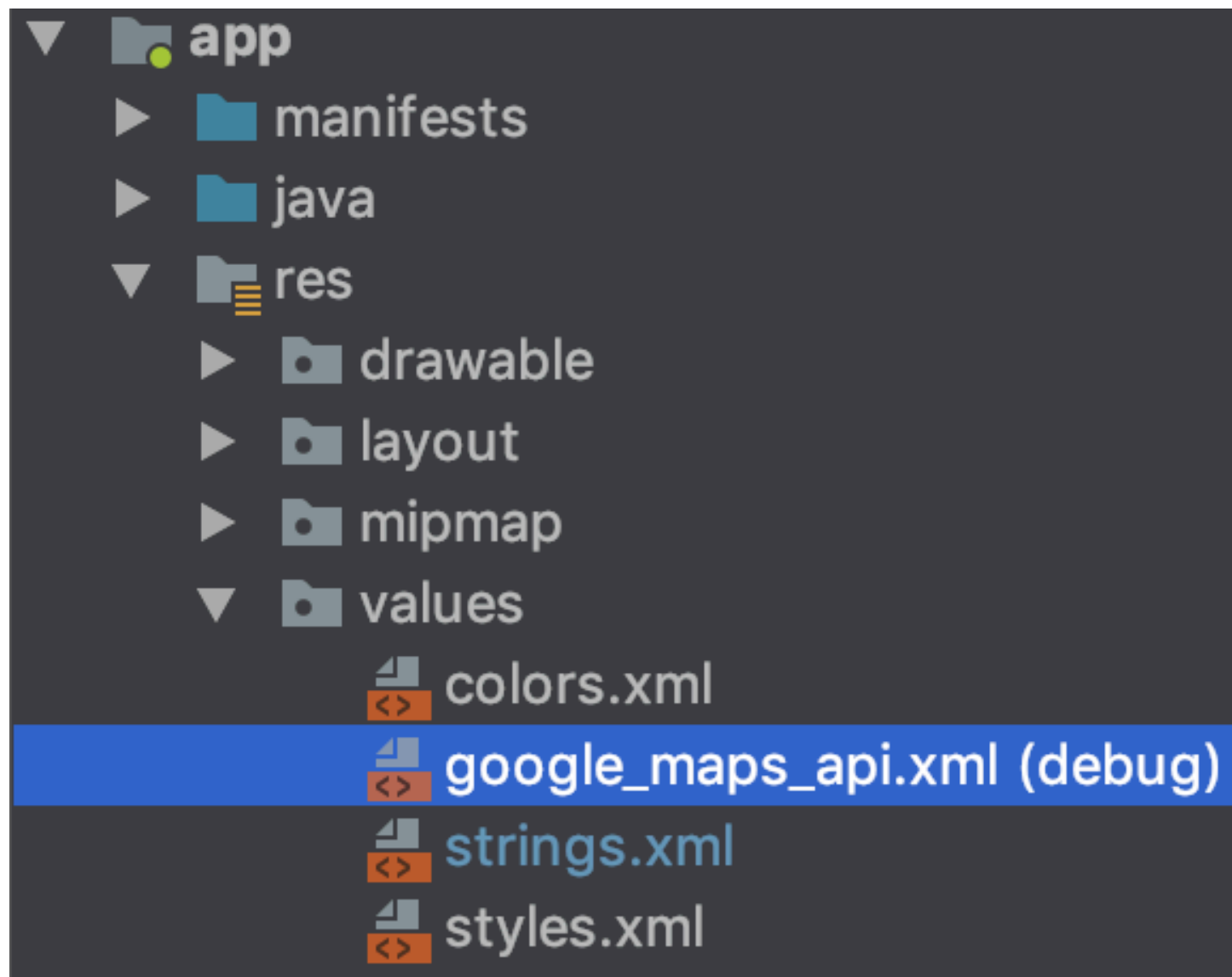
    } else {
        Toast.makeText(getActivity(), "Please Choose Image", Toast.LENGTH_SHORT).show();
    }
}

private void photoController() {
    imageView = getView().findViewById(R.id.imvPhoto);
    imageView.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) {
            Intent intent = new Intent(Intent.ACTION_PICK);
            intent.setType("image/*");
            startActivityForResult(Intent.createChooser(intent, "Choose App"), 1);
        }
    });
}

```

# Google Map Api Step

- Create google\_maps\_api.xml



# google\_maps\_api.xml

```
<resources>
```

```
<string name="google_maps_key"  
templateMergeStrategy="preserve"  
translatable="false">YOUR_KEY_HERE</string>
```

```
</resources>
```



Home - PNPSOLUTION - Google X

https://console.cloud.google.com/home/dashboard?project=pnpsolution-33c31

Google Cloud Platform PNPSOLUTION

DASHBOARD ACTIVITY CUSTOMIZE

### Project info

Project name  
PNPSOLUTION

Project ID  
pnpsolution-33c31

Project number  
891896287527

→ Go to project settings

### Resources

Cloud Storage  
2 buckets

Cloud Functions  
1 function

### Trace

No trace data from the past 7 days

→ Get started with Stackdriver Trace

### Getting Started

### API APIs

Requests (requests/sec)

Request timed out.

api/request\_count:consumed\_api:REDUCE\_SUM(pnpsolution-33c31): 0.017

→ Go to APIs overview

### Google Cloud Platform status

All services normal

→ Go to Cloud status dashboard

### Billing

Estimated charges  
For the billing period Nov 1 - 4, 2018

USD \$0.00

→ View detailed charges

### Error Reporting

No sign of any errors. Have you set up Error Reporting?

→ Learn how to set up Error Reporting

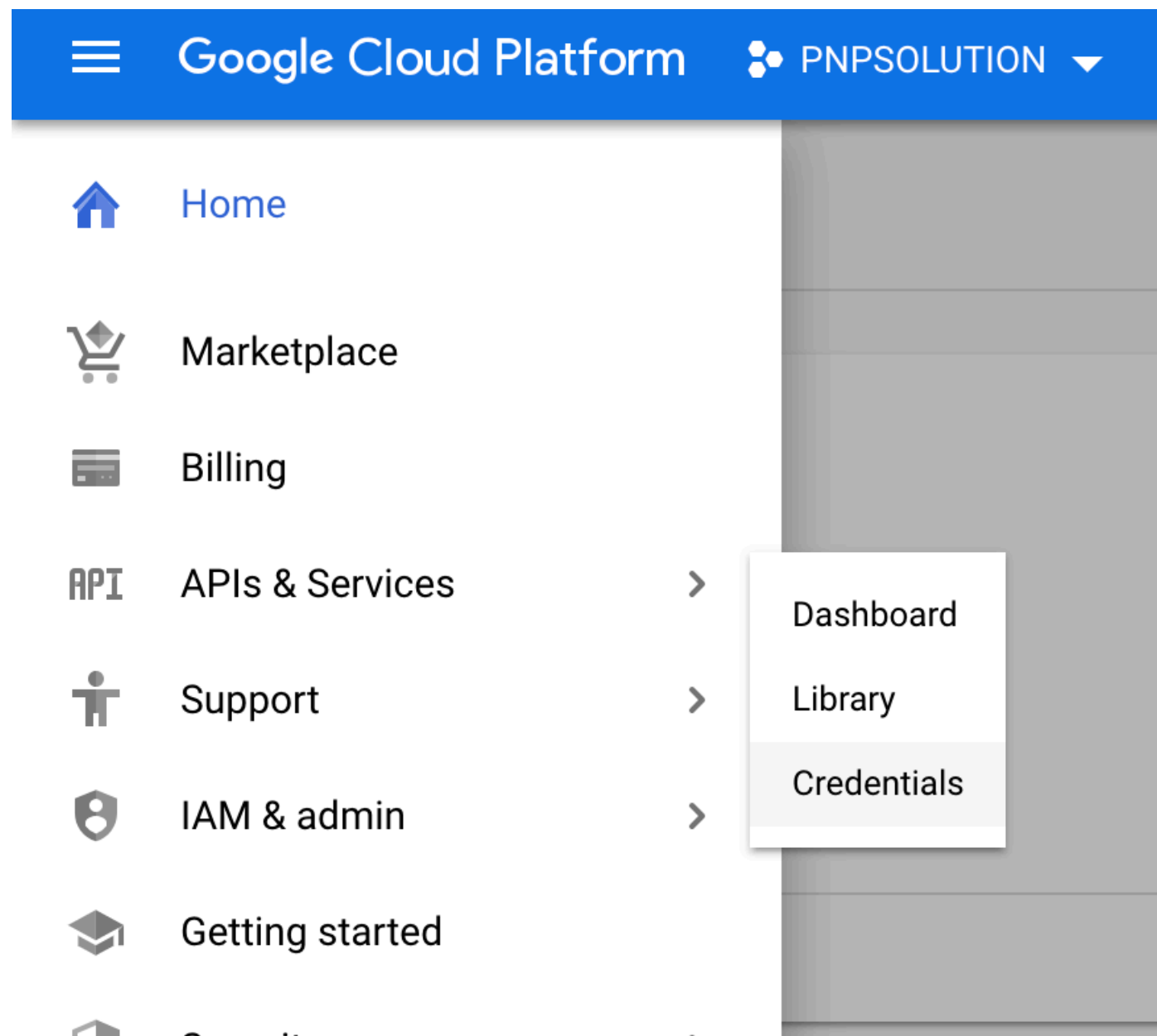
### News

Serverless from the ground up: Connecting Cloud Functions with a database (Part 3)  
3 days ago

Cutting costs with Google Kubernetes Engine: using the cluster autoscaler and Preemptible VMs  
3 days ago

# Generate map api key

Goto : <https://console.cloud.google.com>



# เลือก APIs & Services

แล้วเลือก Credentials

# Credentials

Credentials

OAuth consent screen

Domain verification

Create credentials ▼

Delete

## API key

Identifies your project using a simple API key to check quota and access

## OAuth client ID

Requests user consent so your app can access the user's data

## Service account key

Enables server-to-server, app-level authentication using robot accounts

ation for details.

trictions

ne


# หน้า Credentials


กดปุ่ม Create credentials / API Key

## API key created

Use this key in your application by passing it with the `key=API_KEY` parameter.

Your API key

[Redacted API key] 

 Restrict your key to prevent unauthorized use in production.

[CLOSE](#)

[RESTRICT KEY](#)

# จะได้ api key

ให้นำ key ที่ได้ไปใส่ใน file google\_maps\_api.xml

# Add dependencies

```
implementation 'com.google.android.gms:play-services-maps:15.0.1'
```

# Map Fragment

```
<fragment xmlns:android="http://schemas.android.com/apk/res/android"
```

```
    xmlns:map="http://schemas.android.com/apk/res-auto"
```

```
    xmlns:tools="http://schemas.android.com/tools"
```

```
    android:id="@+id/map"
```

```
    android:name="com.google.android.gms.maps.SupportMapFragment"
```

```
    android:layout_width="match_parent"
```

```
    android:layout_height="match_parent"
```

```
    tools:context=".MapsActivity" />
```

```

public class MapsActivity extends FragmentActivity implements OnMapReadyCallback {

    private GoogleMap mMap;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_maps);
        // Obtain the SupportMapFragment and get notified when the map is ready to be used
        SupportMapFragment mapFragment = (SupportMapFragment) getSupportFragmentManager()
            .findFragmentById(R.id.map);
        mapFragment.getMapAsync(onMapReadyCallback: this);
    }

    @Override
    public void onMapReady(GoogleMap googleMap) {
        mMap = googleMap;

        // Add a marker in Sydney and move the camera
        LatLng sydney = new LatLng(v: -34, v1: 151);
        mMap.addMarker(new MarkerOptions().position(sydney).title("Marker in Sydney"));
        mMap.moveCamera(CameraUpdateFactory.newLatLng(sydney));
    }
}

```

# OnMapReadyCallback

เอาไว้สำหรับในกรณีที่ต้องการสร้าง marker  
 เราจะสามารถสร้างได้ก็ต่อเมื่อ map อยู่ใน state ready แล้ว