

## Bài thực hành số 8 – Bản đồ

**Chú ý chung với toàn bộ các buổi lab:**

- Toàn bộ bài tập trong các buổi lab, bắt buộc sinh viên phải viết mã bằng tay, sử dụng một trong các chương trình soạn thảo mã sau:
  - o Eclipse

### Mục tiêu

**Bài 1: Hiển thị google map và thêm overlay item vào map (thời gian thực hiện: 60 phút)**

#### **1.1. Cài đặt Google Map Library**

Tạo mới ứng dụng Android sử dụng Google Map Android API v2 đòi hỏi một số bước.

##### **1.1.1. Cài đặt Android SDK**

##### **1.1.2. Download và cài đặt Google Play Service SDK**

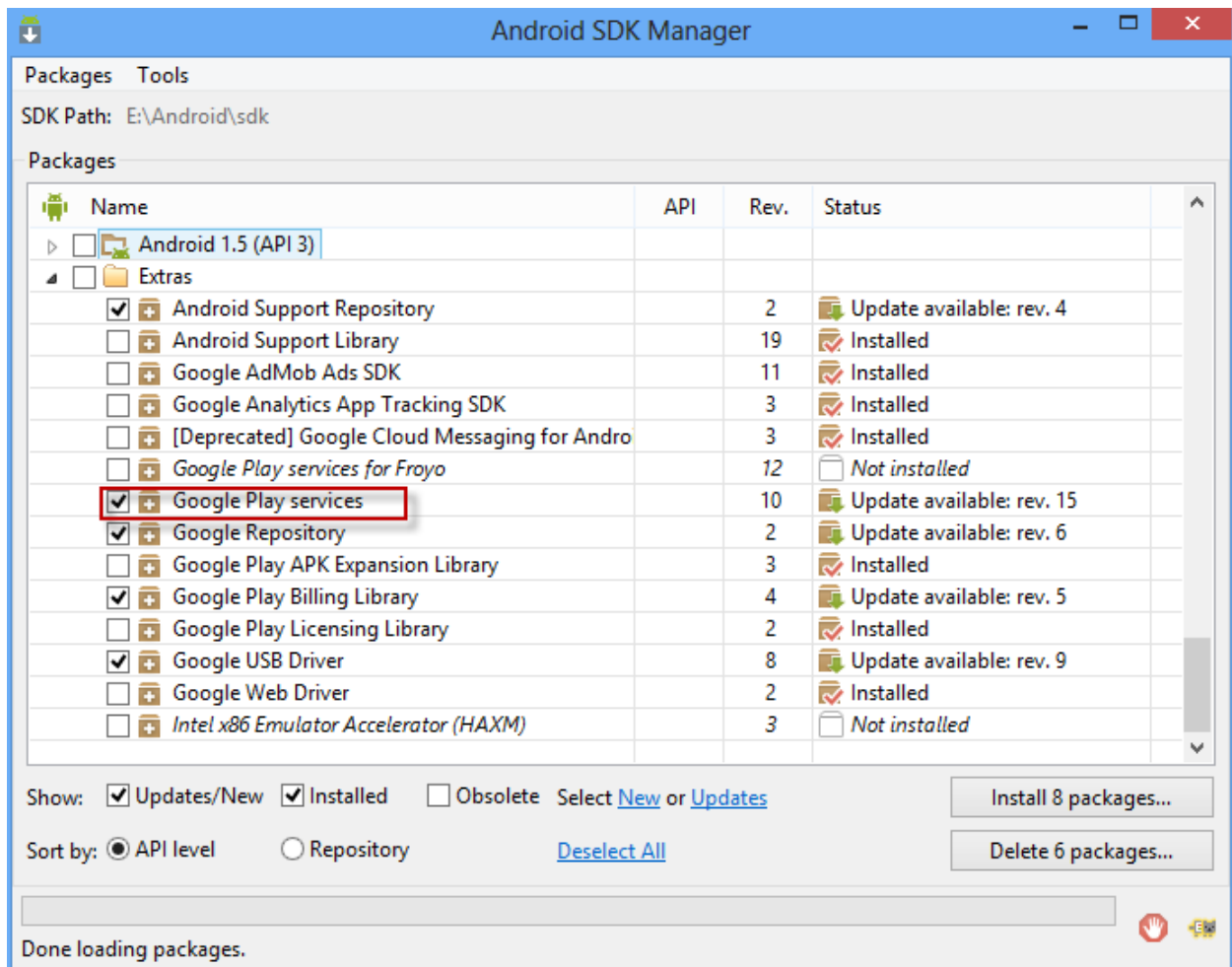
Google Play Service SDK có sẵn Google Map API.

##### **1.1.2.1. Khởi tạo SDK Manager**

Vào Eclipse (cài sẵn ADT), chọn Windows > Android SDK Manager

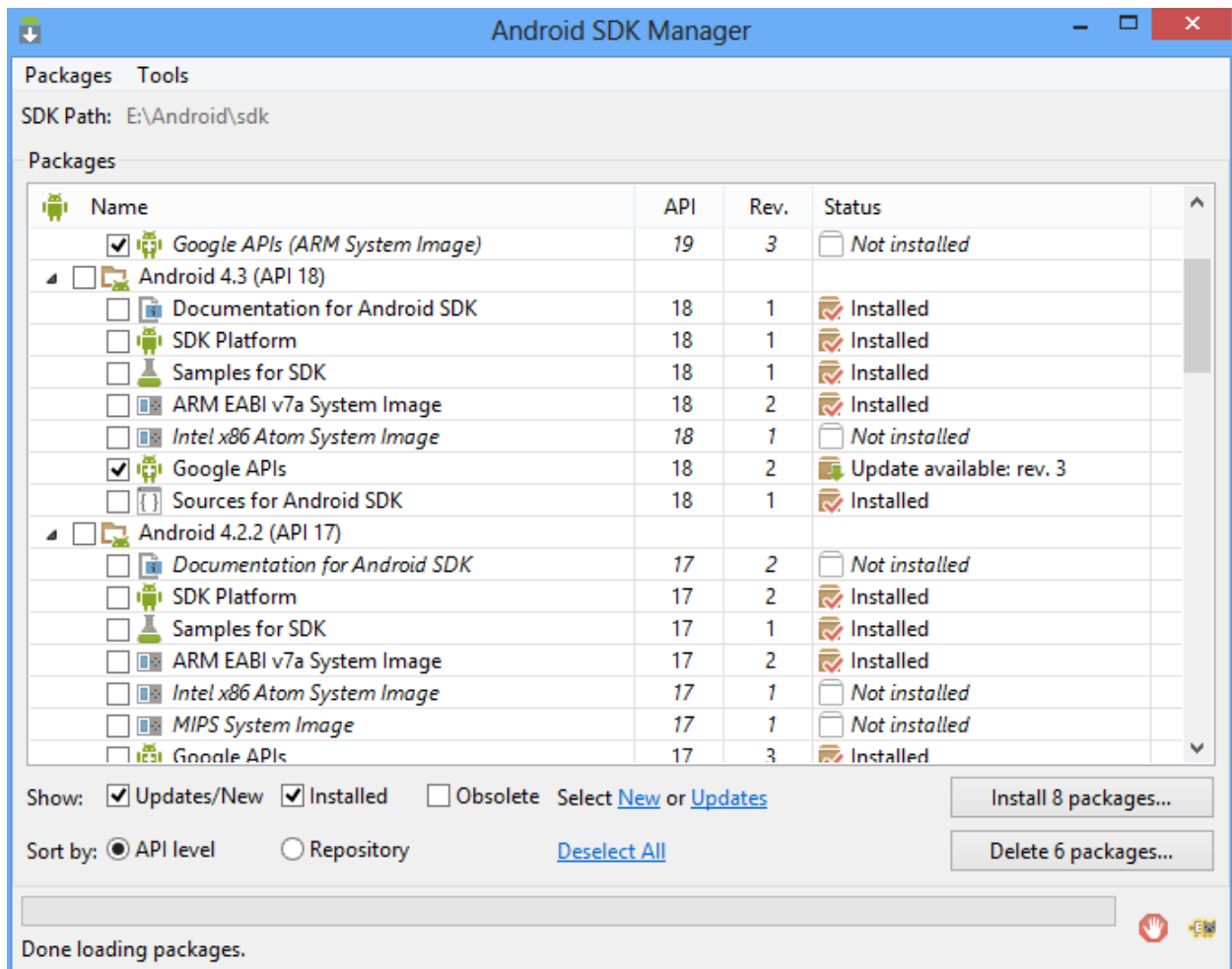
##### **1.1.2.2. Cài đặt Google Play Service SDK**

Cuộn xuống cuối trang danh sách gói, chọn phần **Extra**, chọn **Google Play Service** và tiến hành cài đặt



### 1.1.2.3. Cài đặt version tương thích của Google API platform

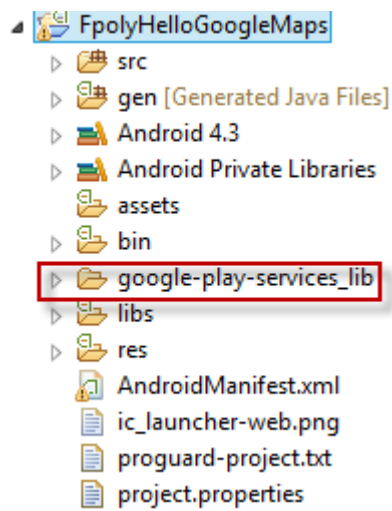
Nếu bạn muốn kiểm tra ứng dụng trên emulator, mở rộng thư mục Android 4.2.2 (API 17) hoặc version mới hơn, chọn Google APIs và tiến hành cài đặt. Sau đó, bạn tạo mới AVD với Google API vừa cấu hình



#### 1.1.2.4. Tạo bản copy của Google Play services library project

Copy library project tại địa chỉ `<android-sdk>/extras/google/google_play_services/libproject/google-play-services_lib/` tới vị trí đặt project

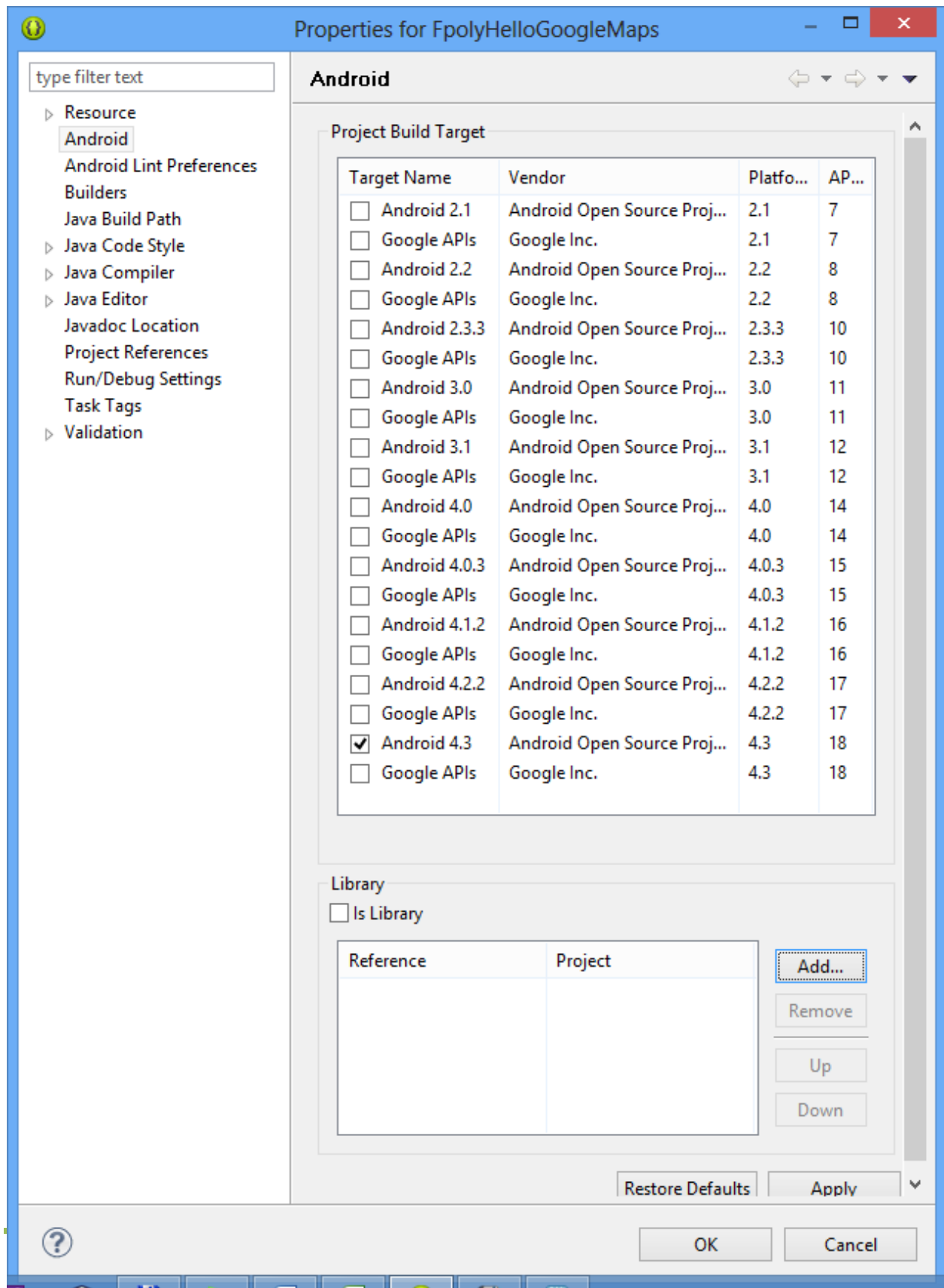
Click **File > Import**, chọn **Android > Existing Android Code into Workspace**, và chọn tới library project và tiến hành import. Sau khi import thành công, bạn sẽ nhìn thấy **google-play-services\_lib** trong cấu trúc của project

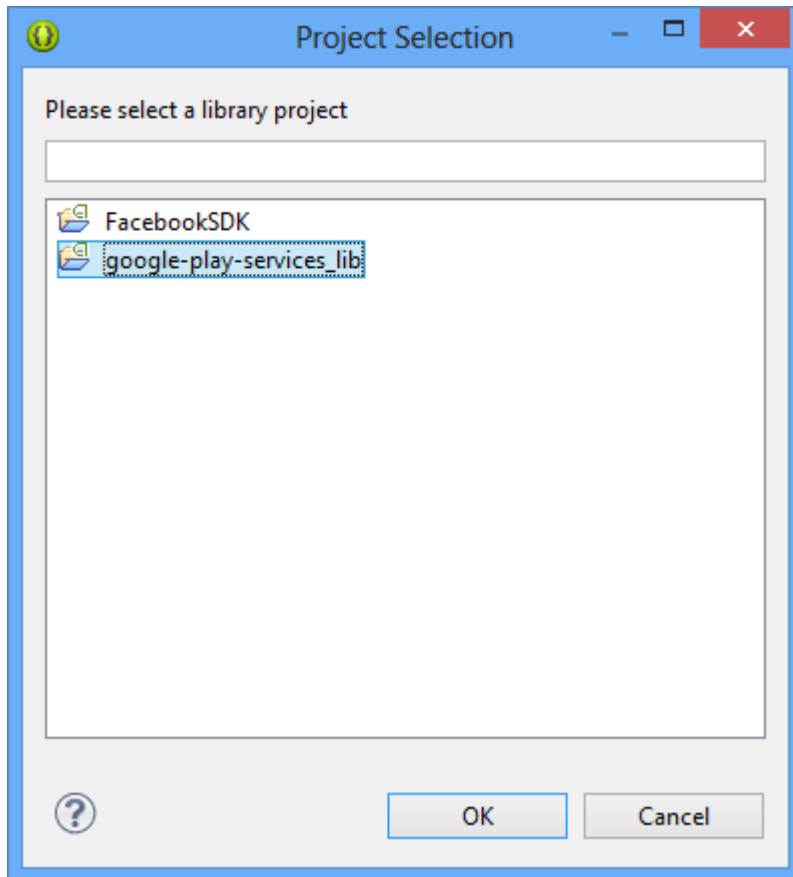


### 1.1.3. Cấu hình project sử dụng Google Play Service

Để ứng dụng của bạn có thể sử dụng được Google Play Service API, bạn phải tham chiếu tới library project bạn tạo ở bước trên

Chọn **Properties** của ứng dụng, chọn **Android**, sau đó click nút **Add** để thêm thư viện **google-play-services\_lib**





Chú ý: bạn nên tham chiếu đến thư mục **google-play-services\_lib** trên thư mục project của bạn, tuyệt đối không tham chiếu đến thư mục ở trên Android SDK

Sau đó thêm tag sau vào file AndroidManifest.xml như là phần tử con của <application>

```
<meta-data android:name="com.google.android.gms.version"
    android:value="@integer/google_play_services_version" />
```

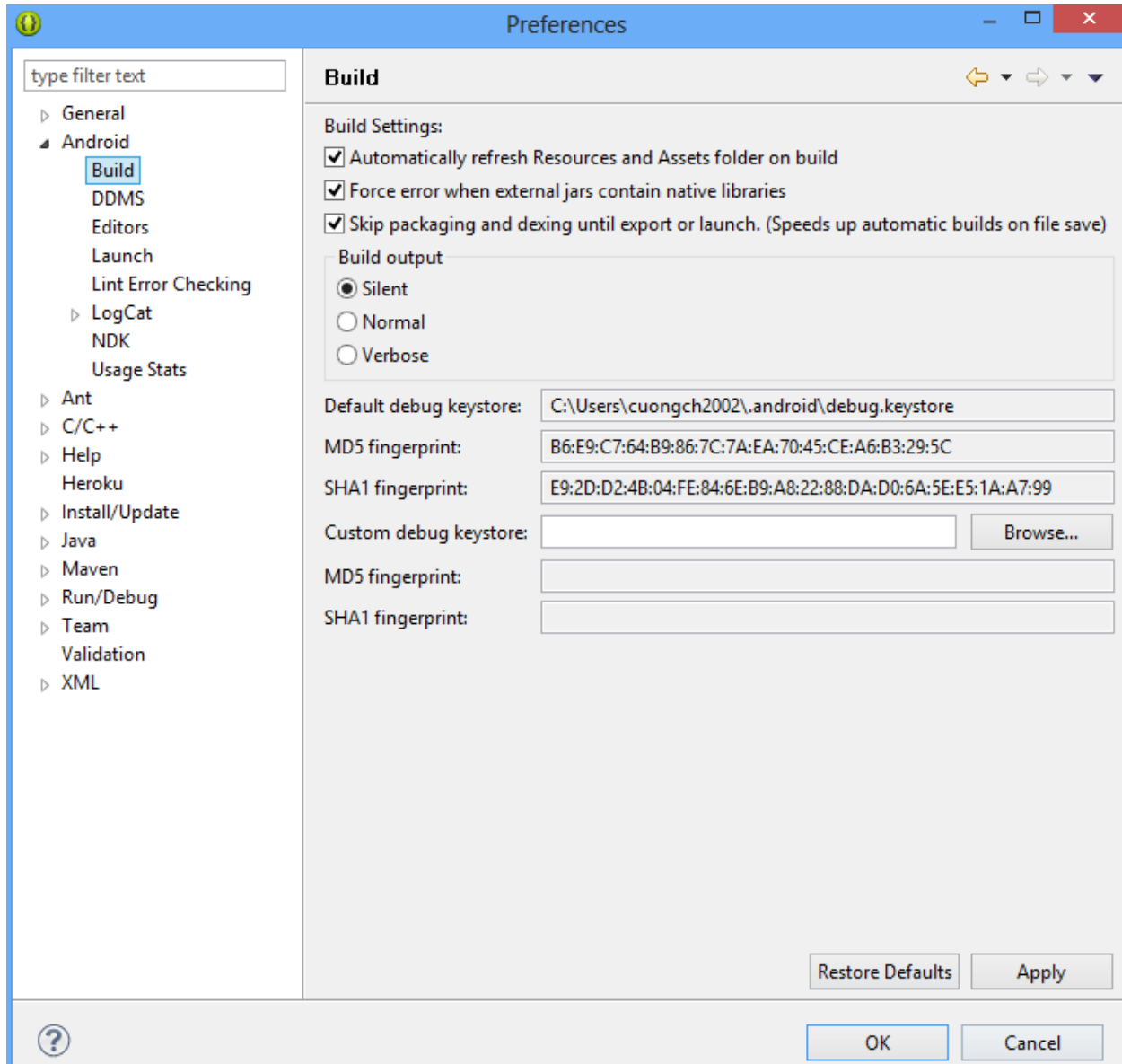
Sử dụng Google Map, bạn cần tạo khóa Google Maps API hợp lệ. Khóa là miễn phí, bạn có thể sử dụng với ứng dụng của bạn dùng để gọi Maps API, và hỗ trợ không hạn chế người dùng

Bạn có thể nhận khóa khi bạn đăng ký ứng dụng Android trong khi phát triển ứng dụng. Trong khi phát triển ứng dụng với Eclipse, Eclipse sẽ tự động tạo và sử dụng một **debug key**

#### 1.1.4. Tạo SHA-1 cho khóa chữ ký của bạn

Eclipse debug key dùng để đăng ký ứng dụng có thể tìm thấy tại *userhome/.android/debug.keystore*

Hoặc bạn vào Eclipse Preferences, chọn Android->Build. Bạn sẽ nhìn thấy vị trí của debug keystore



Để tạo SHA-1 cho debug keystore bạn sử dụng công cụ **keytool** từ thư mục cài đặt JDK để trỏ tới file **debug.keystore**

```
keytool -list -v -alias androiddebugkey -keystore <path_to_debug_keystore>\debug.keystore\ -storepass android -keypass android
```

Trong đó, **<path\_to\_debug\_keystore>** là đường dẫn đến keystore trên máy bạn


Copy kết quả SHA1 trên màn hình console để dùng trong các bước sau:



### 1.1.5. Đăng ký với Google APIs Console

Bạn vào địa chỉ <https://code.google.com/apis/console/b/0/> để truy cập Google APIs Console. Đăng nhập bằng địa chỉ gmail của bạn

Tạo mới project

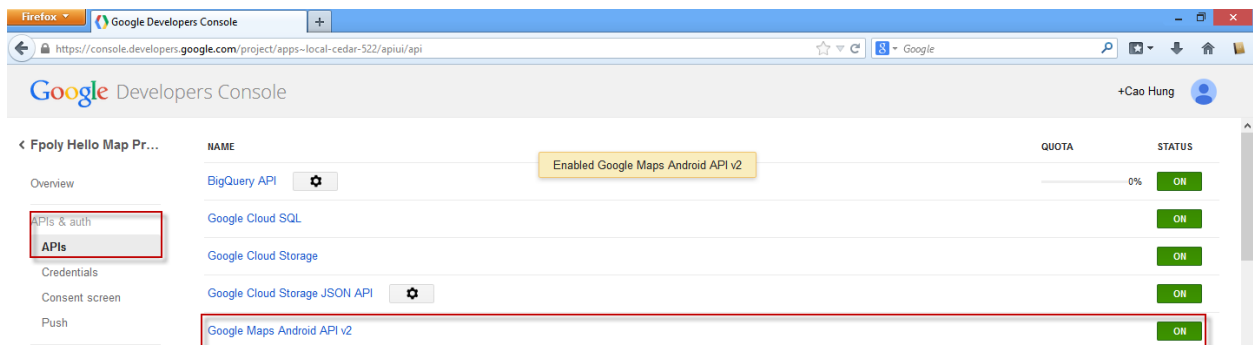
#### New Project

Project name 

Project ID   

Sau đó bạn chờ một lát để google console tạo mới Project, click vào **API & auth**, chọn **APIs**

Kích hoạt Google Maps Android API v2



### 1.1.6. Tạo khóa cho ứng dụng

Bạn sẽ đăng ký ứng dụng của bạn thông qua package trong cửa sổ console cùng với mã SHA-1 của chữ ký số của bạn

Chọn **APIs & auth** > **Credentials**, click **Create new key**

Chọn Android key



## Create a new key

The APIs represented in the Google Developers Console require that requests include a unique project identifier. This enables the Console to tie a request to a specific project in order to monitor traffic, enforce quotas, and handle billing.

Server key

Browser key

Android key

iOS key

Copy mã SHA-1 và tên package vào textbox, chú ý mã SHA-1 và tên package cách nhau bằng dấu chấm phẩy

## Create an Android key and configure allowed Android applications

**This key can be deployed in your Android application.**

API requests are sent directly to Google from your client Android device. Google verifies that each request originates from an Android application that matches one of the certificate SHA1 fingerprints and package names listed below. You can discover the SHA1 fingerprint of your developer certificate using the following command:

```
keytool -list -v -keystore mystore.keystore
```

[Learn more](#)

**Accept requests from an Android application with one of the certificate fingerprints and package names listed below**

One SHA1 certificate fingerprint and package name (separated by a semicolon) per line. Example:

```
45:B5:E4:6F:36:AD:0A:98:94:B4:02:66:2B:12:17:F2:56:26:A0:E0;com.example
```

```
E9:2D:D2:4B:04:FE:84:6E:B9:A8:22:88:DA:D0:6A:5E:E5:1A:A7:99;com.example.fpolyhellogooglemap
```

Create

Cancel

**Public API access**  
 Use of this key does not require any user action or consent, does not grant access to any account information, and is not used for authorization.  
[Learn more](#)  

CREATE NEW KEY

**Key for Android applications**  

API key	AIzaSyAokmeABKxjQP19K8R3C1pHTJ9nPk6UUDE
Android applications	E9:2D:D2:4B:04:FE:84:6E:B9:A8:22:88:DA:D0:6A:5E:E5:1A:A7:99;com.example.fpolyhellogooglemap
Activation date	Mar 17, 2014 2:44 AM
Activated by	caohungcuong1980@gmail.com (you)

Edit allowed Android applications

Regenerate key

Delete

## Bổ sung permission vào file AndroidManifest

```
<uses-permission android:name="android.permission.INTERNET"/>
<uses-permission android:name="android.permission.ACCESS_NETWORK_STATE"/>
<uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE"/>
<uses-permission android:name="com.google.android.providers.gsf.permission.READ_GSERVICES"/>
<!-- The following two permissions are not required to use
      Google Maps Android API v2, but are recommended. -->
<uses-permission android:name="android.permission.ACCESS_COARSE_LOCATION"/>
<uses-permission android:name="android.permission.ACCESS_FINE_LOCATION"/>
```

## Xác định yêu cầu cho OpenGL ES version 2

Google Maps Android API sử dụng OpenGL ES version 2 để sinh ra map. Bổ sung thêm thẻ sau vào AndroidManifest

```
<uses-feature
    android:glEsVersion="0x00020000"
    android:required="true"/>
```

### 1.1.7. Thêm Map vào activity-main.xml

Vào trong file res/layout/activity-main.xml, thay thế code bằng đoạn code sau:

```
<?xml version="1.0" encoding="utf-8"?>
<fragment xmlns:android="http://schemas.android.com/apk/res/android"
    android:id="@+id/map"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:name="com.google.android.gms.maps.MapFragment"/>
```

### 1.1.8. Thêm code vào file MainActivity.java

```
package com.example.ipolyhellogooglemap;

import android.os.Bundle;
import android.app.Activity;
import android.view.Menu;

public class MainActivity extends Activity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
    }

}
```

Sau đó bạn chạy chương trình trên thiết bị Android và sẽ thấy google map được hiển thị

### Yêu cầu nộp bài

Cuối giờ thực hành, sinh viên tạo thư mục theo tên <Tên đăng nhập SV>\_Lab8, chứa tất cả sản phẩm của những bài lab trên, nén lại thành file zip và upload lên mục nộp bài tương ứng trên LMS.

### Thang điểm đánh giá

Tên bài	Điểm	Cách thức đánh giá
Bài 1	10	