

Module :

Développement mobile native :



Chapitre 5:
Géolocalisation et Google Maps sous Android



Pr. Zakia EL UAHHABI

Géolocalisation



- Il y'a deux API qui sont liées au concept de la géolocalisation:
 - Une API qui permet d'afficher des cartes (dans l'espace de noms `com.google.android.maps`) ;
 - Une API qui permet de localiser l'appareil (dans l'espace de noms `android.location`)



Affichage de la carte

- Android nous permet d'intégrer Google Maps dans nos applications en offrant plusieurs fonctionnalités à savoir:
 - Afficher n'importe quel emplacement sur la carte.
 - Afficher l'emplacement actuel de l'appareil
 - Personnaliser la carte en fonction de vos choix
 -



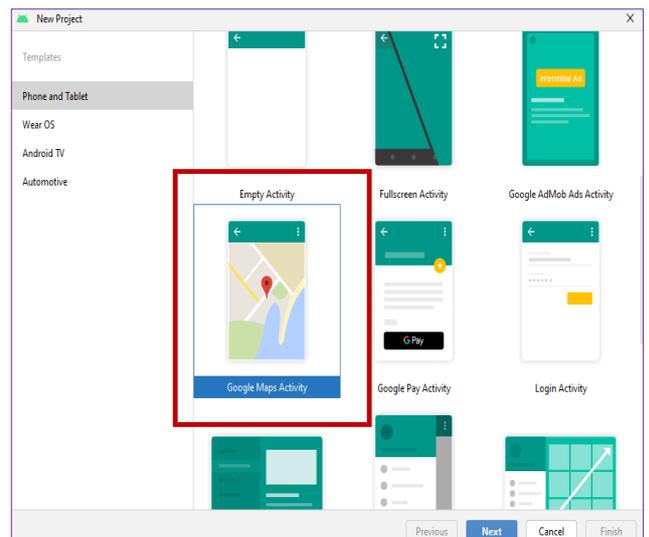
Géolocalisation

Affichage de la carte

- Pour afficher une carte Google Maps dans Android, il faut avoir une clé de l'API. Pour avoir cette clé, il faut suivre les étapes suivantes:

Etape 1:

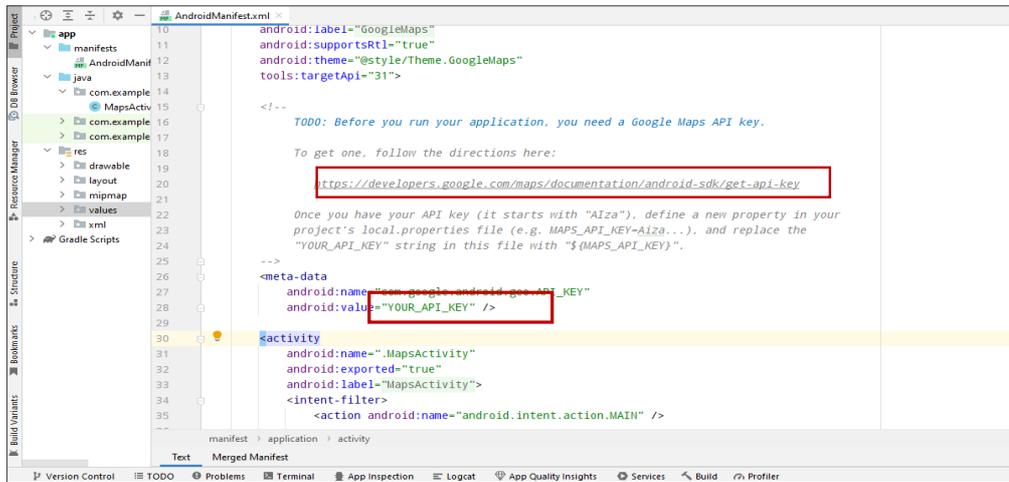
Créer un nouveau projet dans Android studio en choisissant Google Maps Activity



Géolocalisation

Affichage de la carte

- **Etape 2 :** Dans le fichier **Manifest.xml**, vous trouvez le lien de la console dans laquelle vous pouvez générer votre clé. Vous pouvez accéder également à ce lien <https://console.developers.google.com>



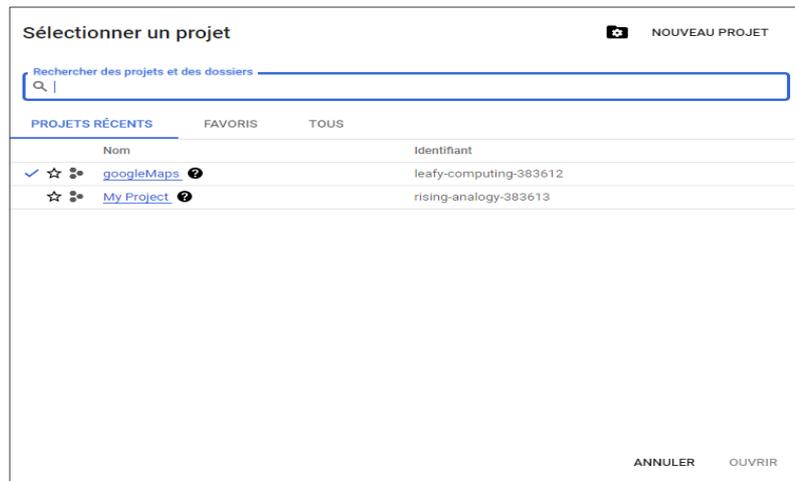
```
10 android:label="@string/google_maps_label"
11 android:supportRtl="true"
12 android:theme="@style/Theme.GoogleMaps"
13 tools:targetApi="31">
14
15 <!--
16     TODO: Before you run your application, you need a Google Maps API key.
17
18     To get one, follow the directions here:
19     https://developers.google.com/maps/documentation/android-sdk/get-api-key
20
21     Once you have your API key (it starts with "AIza"), define a new property in your
22     project's local.properties file (e.g. MAPS_API_KEY=AiZa...), and replace the
23     "YOUR_API_KEY" string in this file with "${MAPS_API_KEY}".
24 -->
25
26 <meta-data
27     android:name="com.google.android.gms.API_KEY"
28     android:value="YOUR_API_KEY" />
29
30 <activity
31     android:name=".MapsActivity"
32     android:exported="true"
33     android:label="@string/app_name">
34     <intent-filter>
35         <action android:name="android.intent.action.MAIN" />
```



Géolocalisation

Affichage de la carte

- **Etape 3 :** Cliquer sur créer un projet



Sélectionner un projet

Rechercher des projets et des dossiers

PROJETS RÉCENTS		FAVORIS	TOUS
Nom	Identifiant		
✓ ☆ googleMaps	leafy-computing-383612		
☆ My Project	rising-analogy-383613		

ANNULER OUVRIER



Géolocalisation

Affichage de la carte

- **Etape 4 :** donner le nom de votre projet et cliquer sur créer



Nom du projet *

googleMaps

ID du projet : leafy-computing-383612. Vous ne pourrez pas le modifier par la suite.

MODIFIER

Zone *

Aucune organisation

PARCOURIR

Organisation ou dossier parent

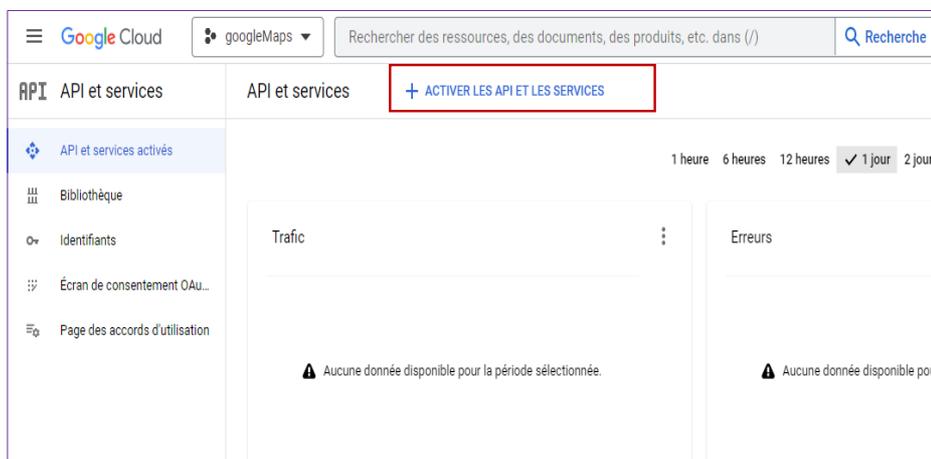
CRÉER ANNULER



Géolocalisation

Affichage de la carte

- **Etape 5 :** sélectionner l'option Activer les API a et les services



Google Cloud googleMaps Rechercher des ressources, des documents, des produits, etc. dans (/) Recherche

API et services API et services + ACTIVER LES API ET LES SERVICES

API et services activés 1 heure 6 heures 12 heures 1 jour 2 jours

Bibliothèque

Identifiants

Écran de consentement OAu...

Page des accords d'utilisation

Trafic Erreurs

Aucune donnée disponible pour la période sélectionnée.

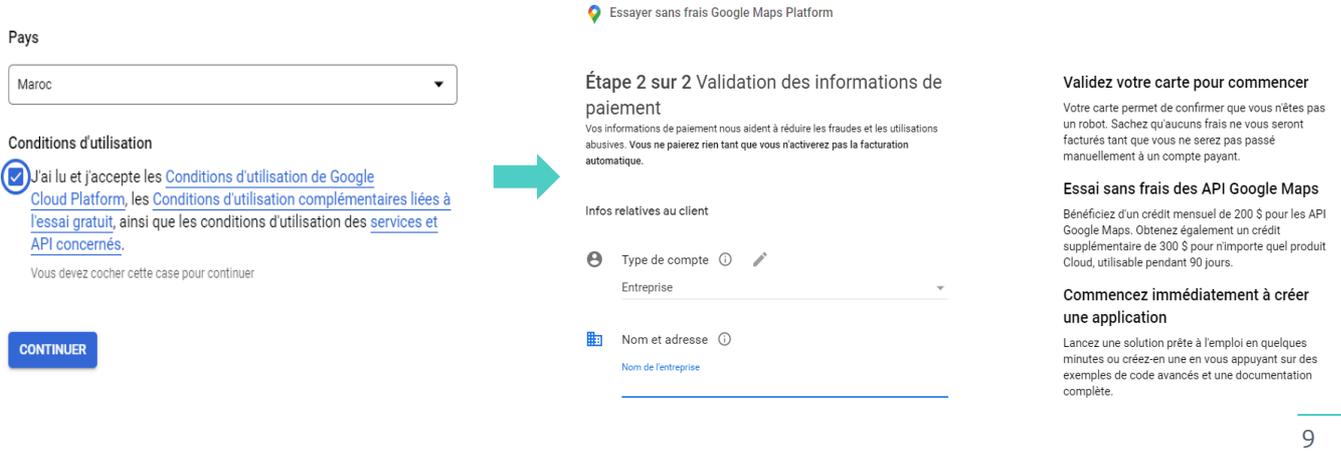
Aucune donnée disponible pour



Géolocalisation

Affichage de la carte

- **Etape 6 :** Cliquer Copier la clé que vous avez obtenu et vous la coller dans le fichier *Manifest.xml*



Pays
Maroc

Conditions d'utilisation
 J'ai lu et j'accepte les [Conditions d'utilisation de Google Cloud Platform](#), les [Conditions d'utilisation complémentaires liées à l'essai gratuit](#), ainsi que les conditions d'utilisation des [services et API concernés](#).
Vous devez cocher cette case pour continuer

CONTINUER

Essayer sans frais Google Maps Platform

Étape 2 sur 2 Validation des informations de paiement
Vos informations de paiement nous aident à réduire les fraudes et les utilisations abusives. Vous ne paierez rien tant que vous n'activerez pas la facturation automatique.

Infos relatives au client

Type de compte ⓘ ✎
Entreprise

Nom et adresse ⓘ
Nom de l'entreprise

Validez votre carte pour commencer
Votre carte permet de confirmer que vous n'êtes pas un robot. Sachez qu'aucuns frais ne vous seront facturés tant que vous ne serez pas passé manuellement à un compte payant.

Essai sans frais des API Google Maps
Bénéficiez d'un crédit mensuel de 200 \$ pour les API Google Maps. Obtenez également un crédit supplémentaire de 300 \$ pour n'importe quel produit Cloud, utilisable pendant 90 jours.

Commencez immédiatement à créer une application
Lancez une solution prête à l'emploi en quelques minutes ou créez-en une en vous appuyant sur des exemples de code avancés et une documentation complète.

9

Géolocalisation

Affichage de la carte

- **Etape 7 :** Cliquer Copier la clé que vous avez obtenu et vous la coller dans le fichier *Manifest.xml*



	Nom	Date de création ↓	Restrictions	Clé	Utilisation avec tous les services (30 derniers jours) ⓘ
<input checked="" type="checkbox"/>	Clé API 1	14 avr. 2020	Applications Android	AIzaSyAX6Q...8JwPsD5v5w	0



10

Géolocalisation

Affichage de la carte

■ **Etape 8** : Exécution de l'application



11

Géolocalisation

Localisation de l'appareil

- Les services de géolocalisation permettent de déterminer la position courante de l'appareil. Ils incluent le GPS et la technologie Google de localisation cellulaire.
- Il y'a deux API qui sont liées au concept de la géolocalisation:
 - Une API qui permet d'afficher des cartes (dans l'espace de noms `com.google.android.maps`) ;
 - Une API qui permet de localiser l'appareil (dans l'espace de noms `android.location`)



12

Localisation de l'appareil

Pour localiser un appareil mobile, il faut suivre les étapes suivantes:

1. Configurer les services Google Play:

Télécharger et installer le composant des services Google Play via le SDK Manager et ajouter la bibliothèque à votre projet:

```
implementation 'com.google.android.gms:play-services-location:21.0.1'
```



Localisation de l'appareil

2. Spécifier les autorisations des applications:

Les applications qui utilisent les services de localisation doivent demander des autorisations. Android offre les autorisations de localisation suivantes:

- `ACCESS_COARSE_LOCATION`
- `ACCESS_FINE_LOCATION`

Il faut déclarer les permissions appropriées dans le fichier `manifest`.

```
<uses-permission  
android:name="android.permission.ACCESS_FINE_LOCATION"/>
```

```
<uses-permission  
android:name="android.permission.ACCESS_COARSE_LOCATION"/>
```



Localisation de l'appareil

2. Spécifier les autorisations des applications:

L'autorisation que vous choisissez détermine la précision de l'emplacement renvoyé:

- **ACCESS_COARSE_LOCATION**: l'API utilise le Wifi, des données de cellule mobile ou les deux pour déterminer l'emplacement de l'appareil (une localisation par WiFi et antennes relais).
- **ACCESS_FINE_LOCATION** : une géolocalisation par GPS, ainsi que les données Wifi et mobiles.



Le système Android donne un accès de localisation aux applications à condition que les paramètres de sécurité les y autorisent :

Géolocalisation

Localisation de l'appareil: Utilisation des services de géolocalisation

3. Obtenir la liste des fournisseurs de position:

- Les services de géolocalisation sont les différentes technologies utilisées pour spécifier la position courante d'un appareil. Les deux principaux éléments sont les suivants :
 - **Location Manager** : un gestionnaire de position qui permet d'accéder aux services de géolocalisation à partir de la classe [LocationManager](#). Elle permet de:
 - Obtenir une position courante.
 - Suivre des déplacements.
 - Trouver les Location Providers disponibles.



Géolocalisation

Localisation de l'appareil: Utilisation des services de géolocalisation

3. Obtenir la liste des fournisseurs de position:

Les services de géolocalisation sont les différentes technologies utilisées pour spécifier la position courante d'un appareil. Les deux principaux éléments sont les suivants :

Location Providers : (Fournisseurs de position) : Chacun d'eux représente une technologie de localisation de la position de l'appareil. Android se chargera de sélectionner le plus apte à donner la position de l'appareil. Le fournisseur de localisation propose deux types de géolocalisation :

- la première basée sur la technologie GPS, de type `LocationManager.GPS_PROVIDER`. C'est le plus précis des deux, mais c'est également le plus consommateur en terme de batterie.

- La seconde basée sur le réseau cellulaire et les signaux wifi. Elle est de type



17

Géolocalisation

Localisation de l'appareil

3. Obtenir la liste des fournisseurs de position:

L'accès aux services de géolocalisation est géré par le système à l'aide de la méthode `getSystemService()`. On crée une instance de la classe `LocationManager` pour accéder au service de localisation. **Cela permet aux applications d'obtenir des mises à jour périodiques de la position de l'utilisateur.**



18

Localisation de l'appareil

3. Obtenir la liste des fournisseurs de position:

Il existe plusieurs méthodes pour récupérer les fournisseurs de position disponibles sur le terminal mobile :

- ▣ `getAllProviders()` qui permet de récupérer une liste de tous les noms de fournisseur (même si vous n'avez pas les droits d'accès pour certains) ;

// Le code suivant permet d'obtenir la liste de tous les fournisseurs :

```
LocationManager sysLocalisation = (LocationManager)
getSystemService(Context.LOCATION_SERVICE);
List<String> fournisseurs = sysLocalisation.getAllProviders();
```



Localisation de l'appareil

3. Obtenir la liste des fournisseurs de position:

`getProvider(String name)` qui retourne un `LocationProvider` (fournisseur particulier) à partir de son nom. On spécifie le type de fournisseur à l'aide des constantes prédéfinies `LocationManager.GPS_PROVIDER` ou

```
LocationManager sysLocalisation = (LocationManager)
getSystemService(Context.LOCATION_SERVICE);
String fournisseur = LocationManager.GPS_PROVIDER;
LocationProvider gpsProvider = sysLocalisation.getProvider(fournisseur);
```



Localisation de l'appareil

3. Obtenir la liste des fournisseurs de position:

Après avoir choisi un fournisseur de position, on utilise la méthode `getLastKnownLocation()` issue de la classe `LocationManager`, avec comme paramètre le fournisseur de position choisi.

```
LocationManager sysLocalisation = (LocationManager)
getSystemService(Context.LOCATION_SERVICE);
String fournisseur = LocationManager.GPS_PROVIDER;
Location localisation = sysLocalisation.getLastKnownLocation(fournisseur);
```



Localisation de l'appareil

4. Suivre les déplacements:

La plupart des applications de localisation doivent réagir aux déplacements de l'utilisateur. Dans ce cas, on doit **recevoir des mises à jour de sa position et détecter le mouvement** à l'aide de la méthode `requestLocationUpdates` pour la mise à jour via le fournisseur de position.



Localisation de l'appareil

4.Suivre les déplacements:

La méthode `requestLocationUpdates(String provider, long minTime, float minDistance, LocationListener listener)` possède 4 arguments :

- Le `provider` pour recevoir les mises à jour des coordonnées utilisateurs (GPS / NETWORK ...).
- Le `deuxième paramètre` indique le temps entre deux mises à jour exprimé en millisecondes.
- Le `troisième paramètre` précise la distance correspond au nombre de mètres qui doivent être parcourus avant de recevoir une nouvelle position (en mètre)
- L'instance de `LocationListener`.



Localisation de l'appareil

4.Suivre les déplacements:

Le dernier paramètre propose quatre méthodes qu'on doit les redéfinir :

Méthode	Description
<code>onProviderEnabled</code>	cette méthode est appelée quand une source de localisation est activée. L'argument représente le nom de la source activée.
<code>onProviderDisabled</code>	Elle est appelée quand une source de localisation est désactivée. L'argument est le nom de la source désactivée. on peut par exemple vous désabonner à la mise à jour de localisation via cette source.
<code>onStatusChanged</code>	appelée quand le status d'une source change. Il existe 3 statuts (OUT_OF_SERVICE, TEMPORARILY_UNAVAILABLE, AVAILABLE).
<code>onLocationChanged</code>	appelée quand les coordonnées GPS du téléphone changent. L'argument représente la nouvelle position. Vous pouvez récupérer plusieurs informations comme la latitude, longitude, altitude et précision (en mètre)

