

Faculté des technologies de l'information et de la communication

Département informatique appliquée



Travaux pratique

TP5

Réalisé par : Nathan Carlinot RANDRIAMIHAJA

Année scolaire : 2023 - 2024

A- Premiers pas

1- Premiers pas avec Flutter

```
Invité de commandes - flutter x + v
Microsoft Windows [version 10.0.22621.3007]
(c) Microsoft Corporation. Tous droits réservés.

C:\Users\CARLINOT>flutter create google_maps_in_flutter
Creating project google_maps_in_flutter...
Resolving dependencies in google_maps_in_flutter... (2.6s)
Got dependencies in google_maps_in_flutter.
Wrote 129 files.

All done!
You can find general documentation for Flutter at: https://docs.flutter.dev/
Detailed API documentation is available at: https://api.flutter.dev/
If you prefer video documentation, consider: https://www.youtube.com/c/flutterdev

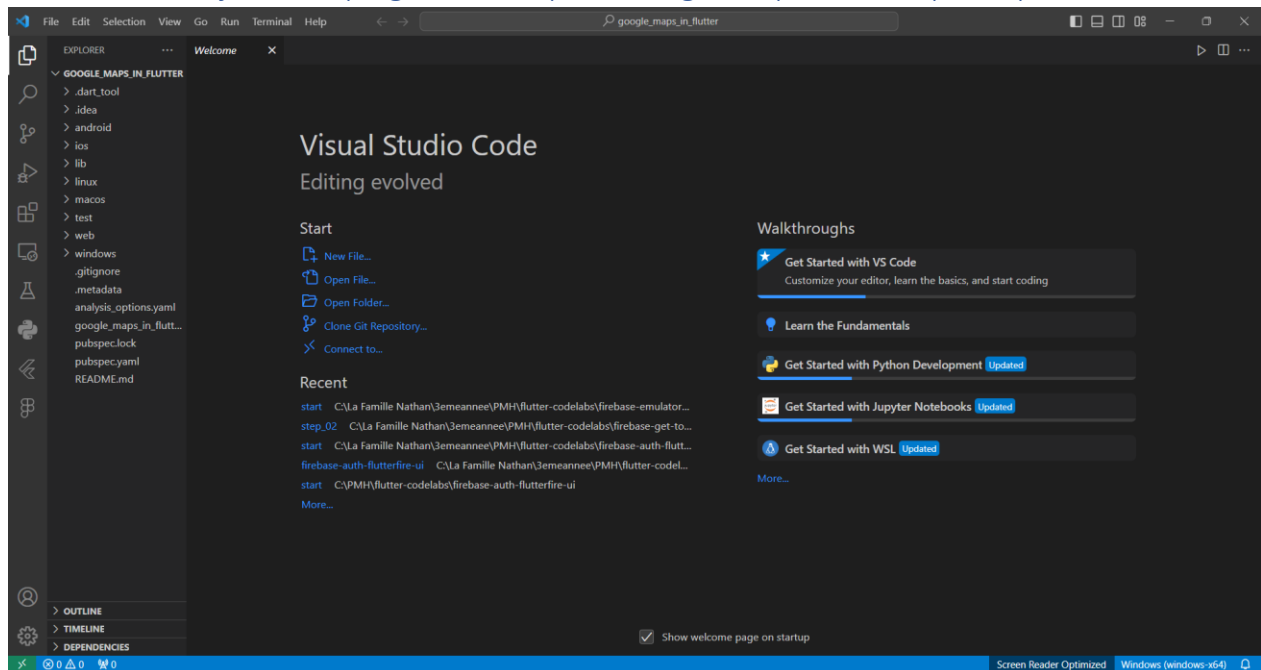
In order to run your application, type:

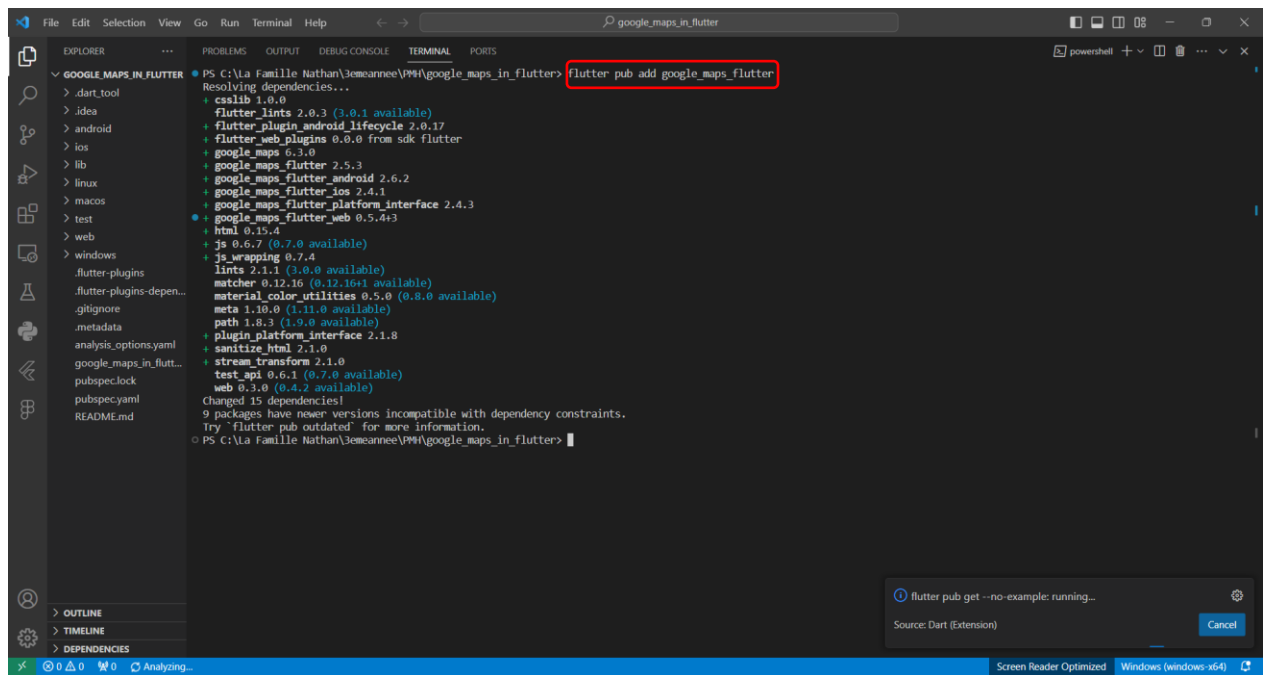
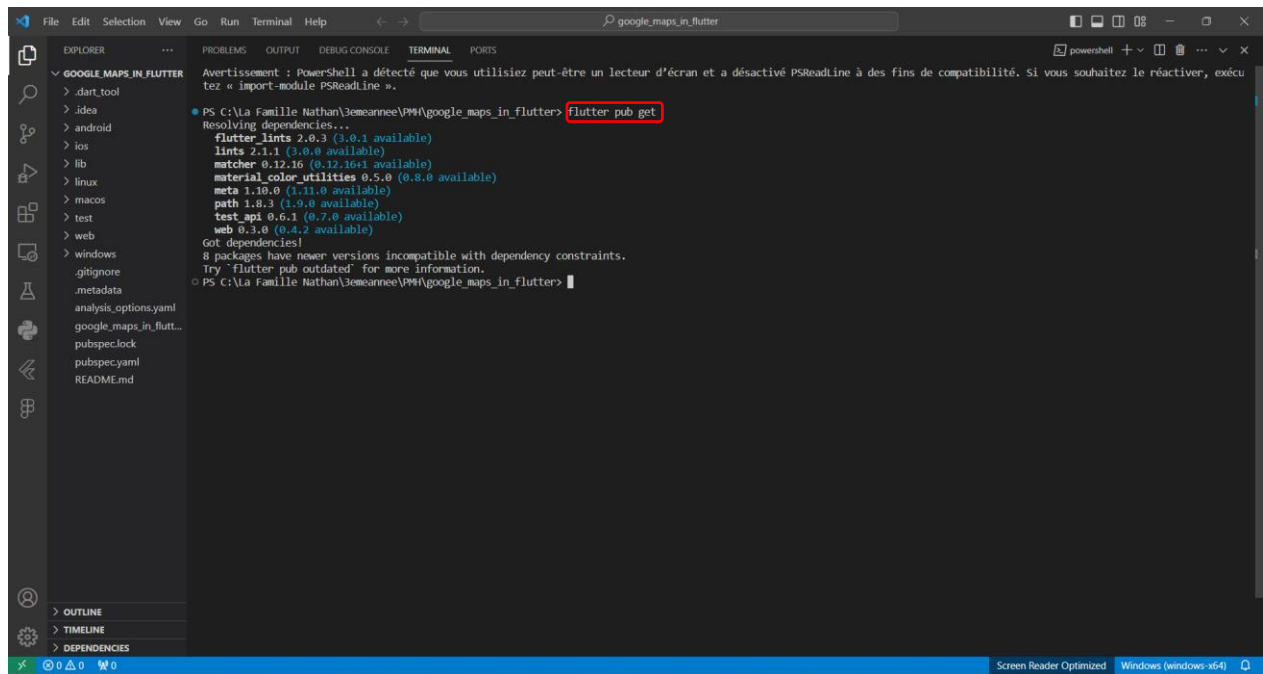
$ cd google_maps_in_flutter
$ flutter run

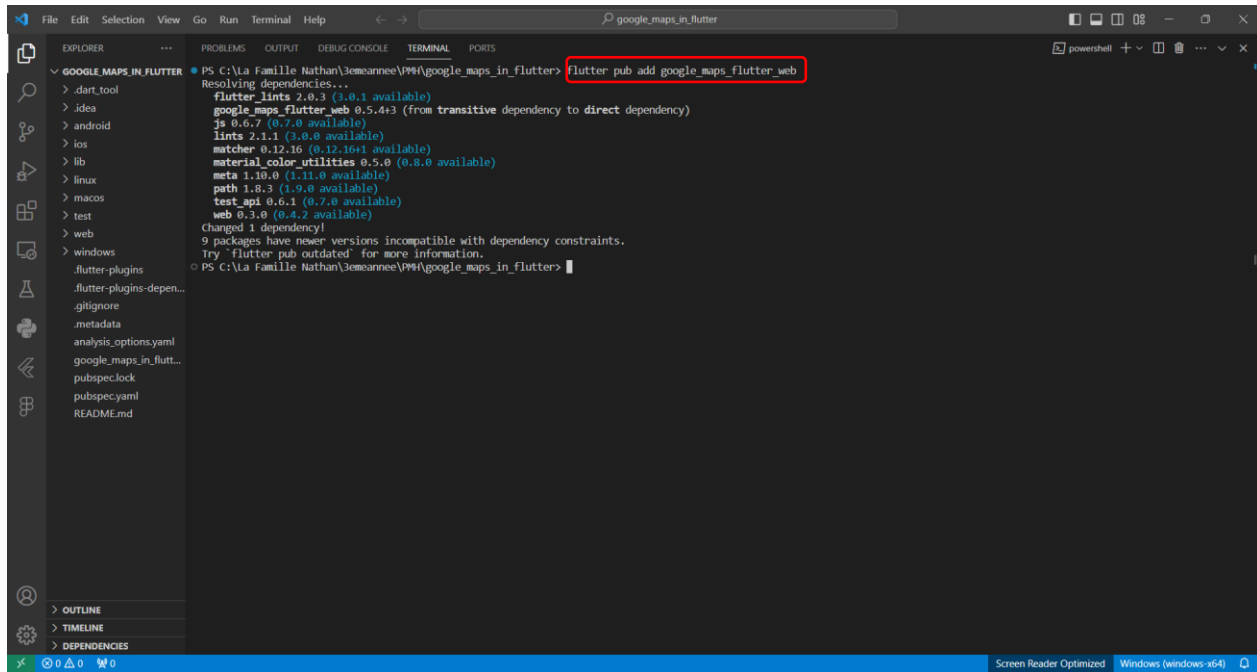
Your application code is in google_maps_in_flutter\lib\main.dart.

C:\Users\CARLINOT>
```

2- Ajouter le plug-in Flutter pour Google Maps en tant que dépendance

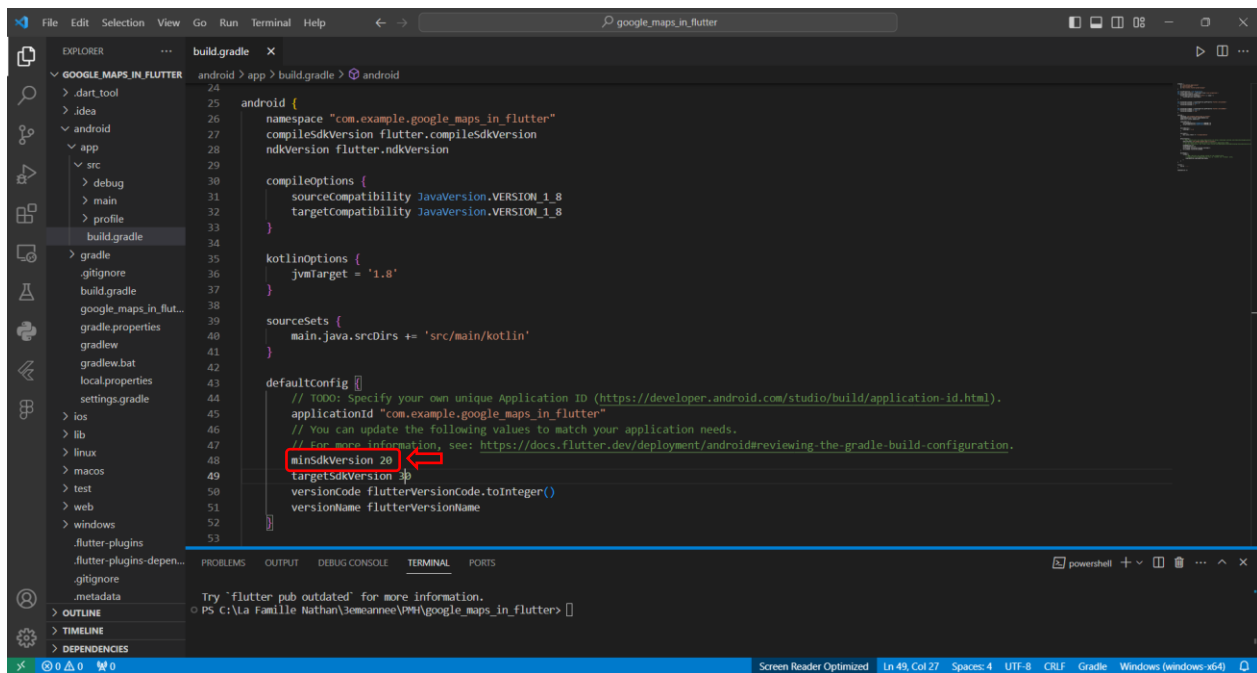






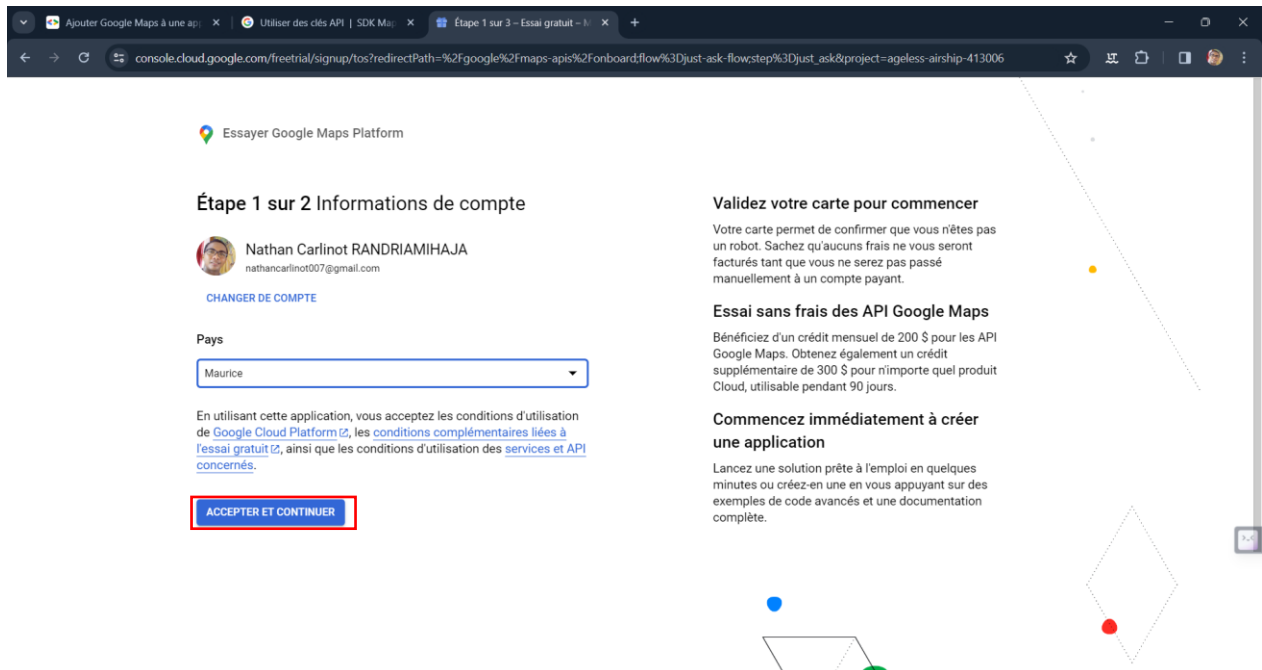
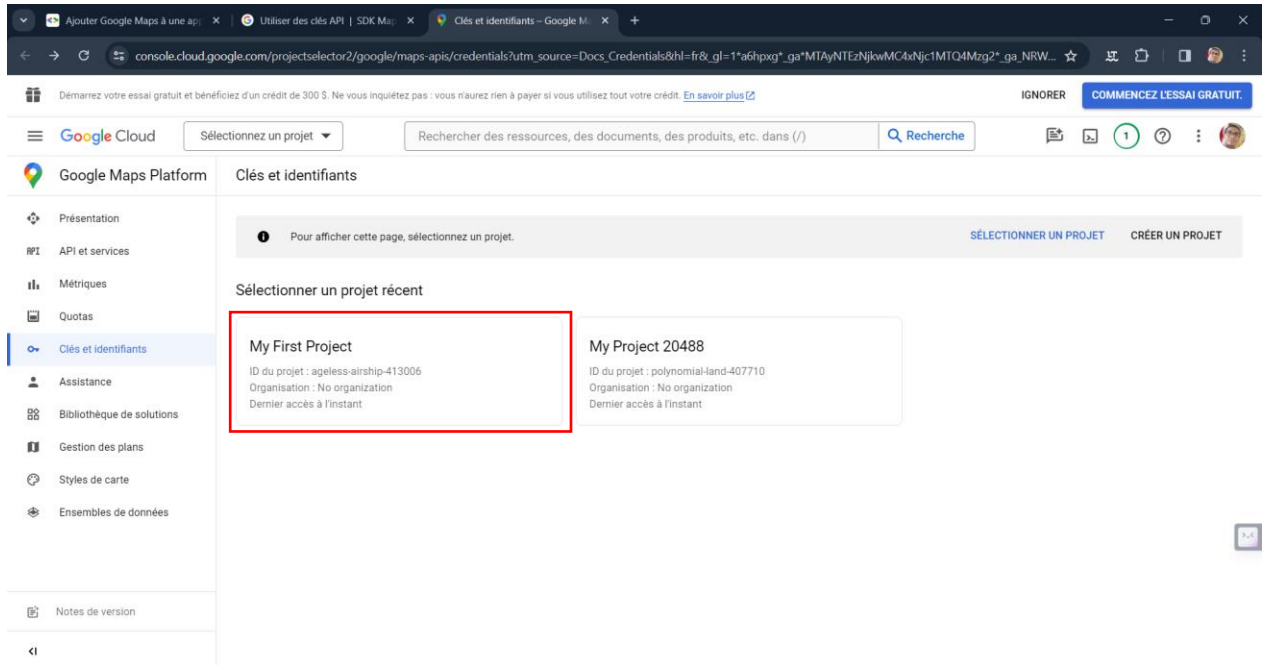
3- Configurer Android minSDK

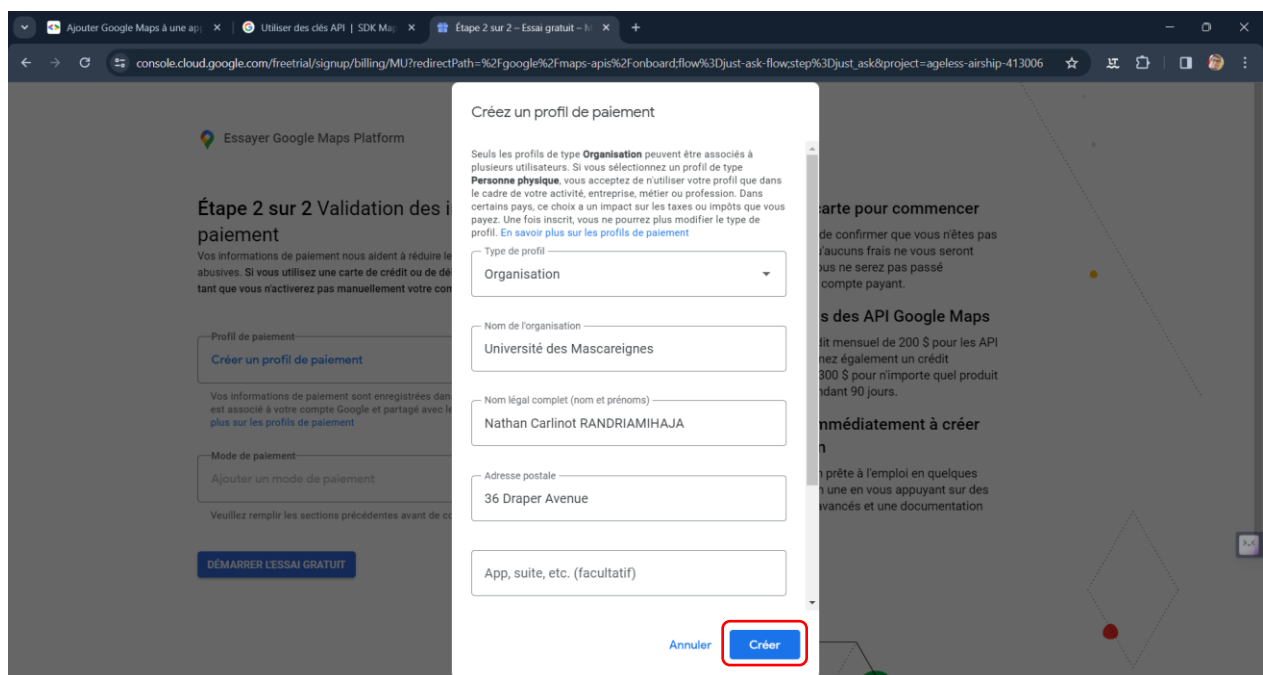
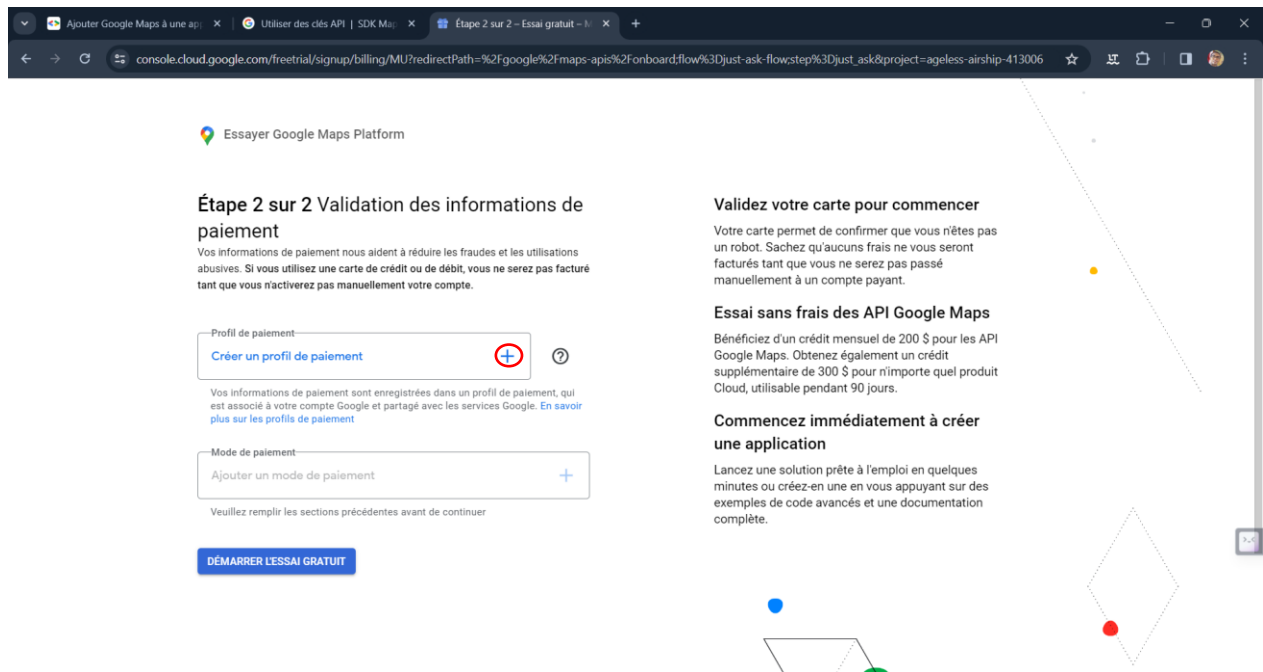
Pour utiliser le SDK Google Maps sur Android, vous devez définir minSDK sur 20. Modifiez le fichier android/app/build.gradle comme suit.

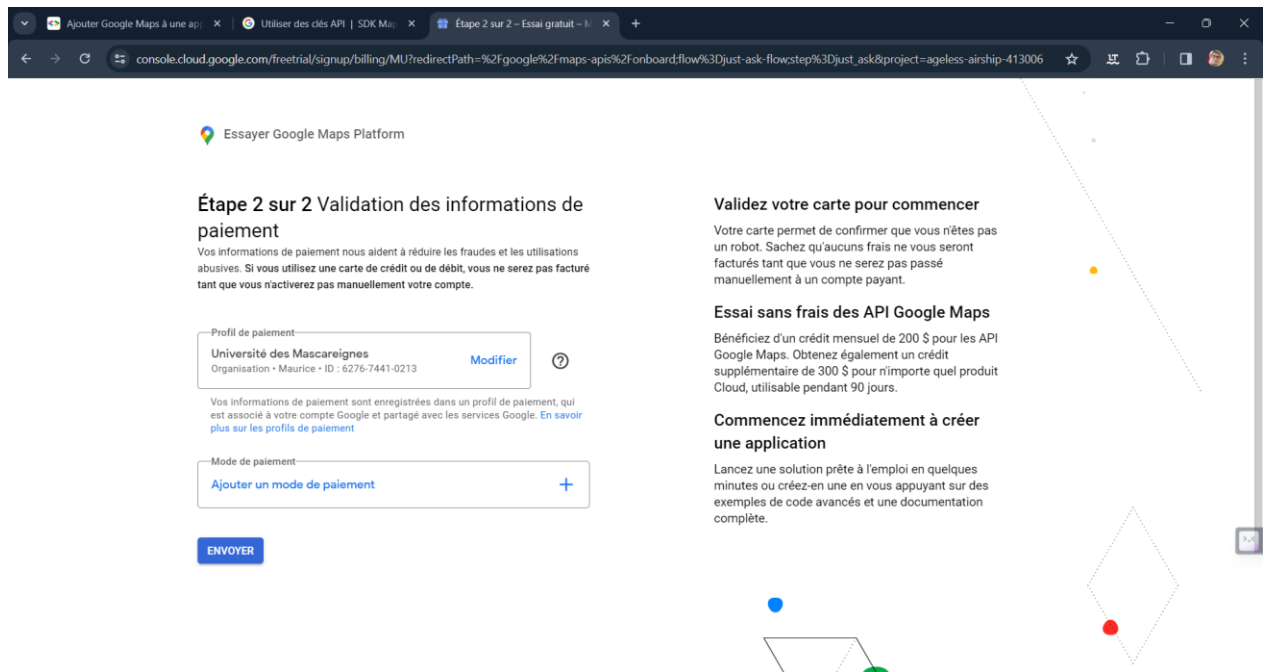


B- Ajouter Google Maps à l'application

1- Ajouter une clé API pour une application Android

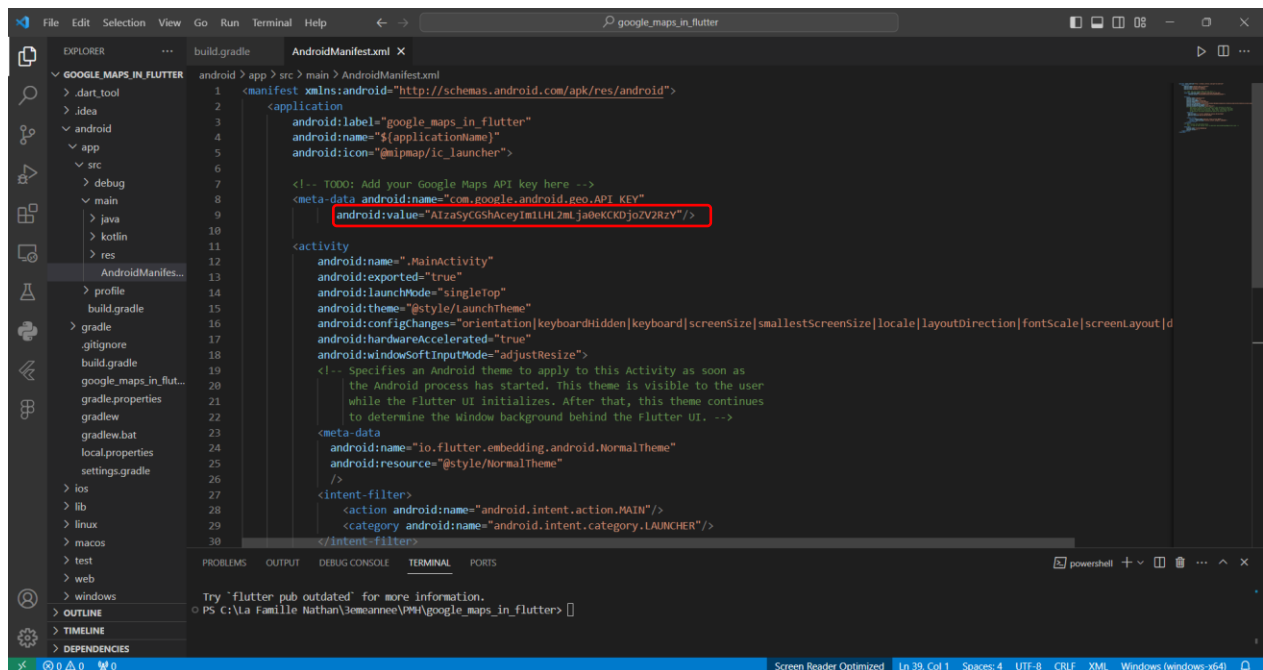




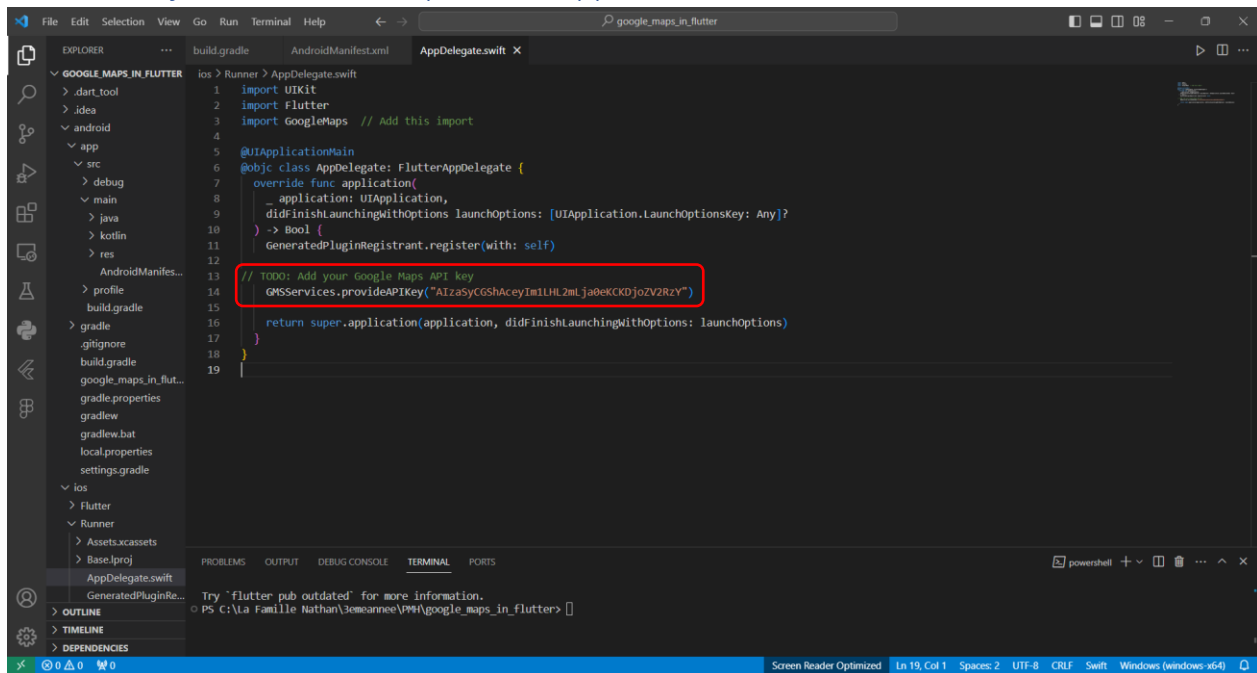


Pour ajouter une clé API à l'application Android, modifiez le fichier AndroidManifest.xml dans android/app/src/main. Ajoutez une seule entrée meta-data contenant la clé API (créée à l'étape précédente) dans le nœud application.

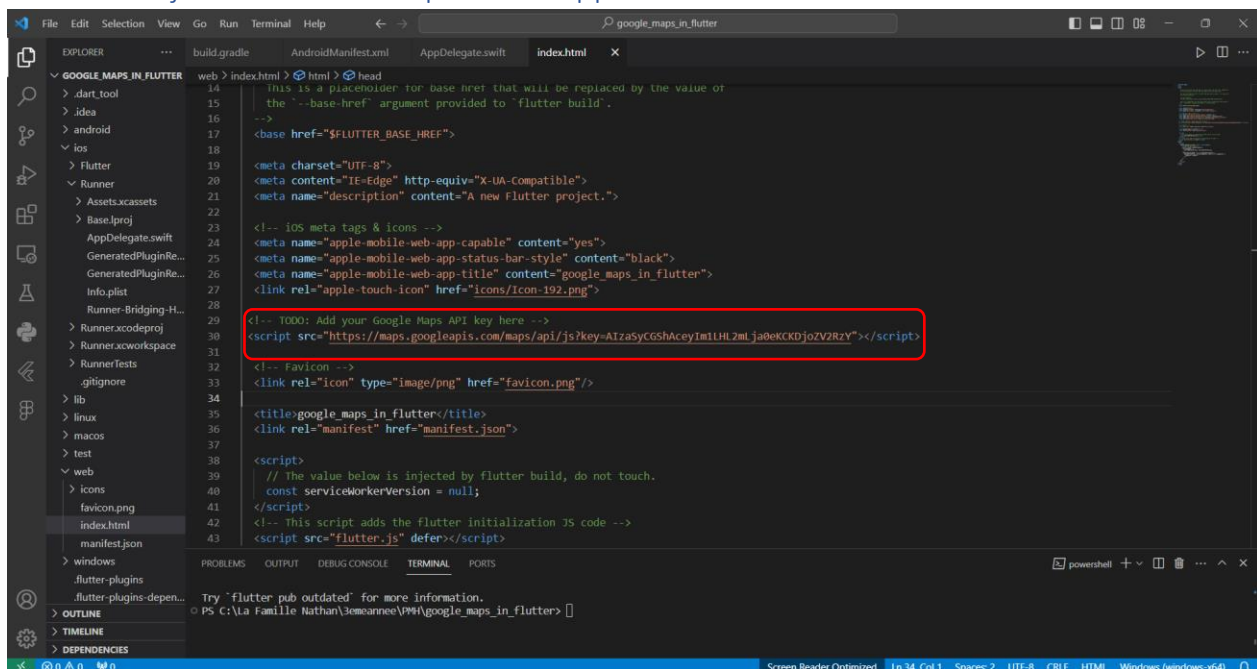
NB : J'ai pris une clé API sur YouTube mais ce n'est pas le mien car il demande ma carte bancaire mais à la fin du résultat, il marche toujours.



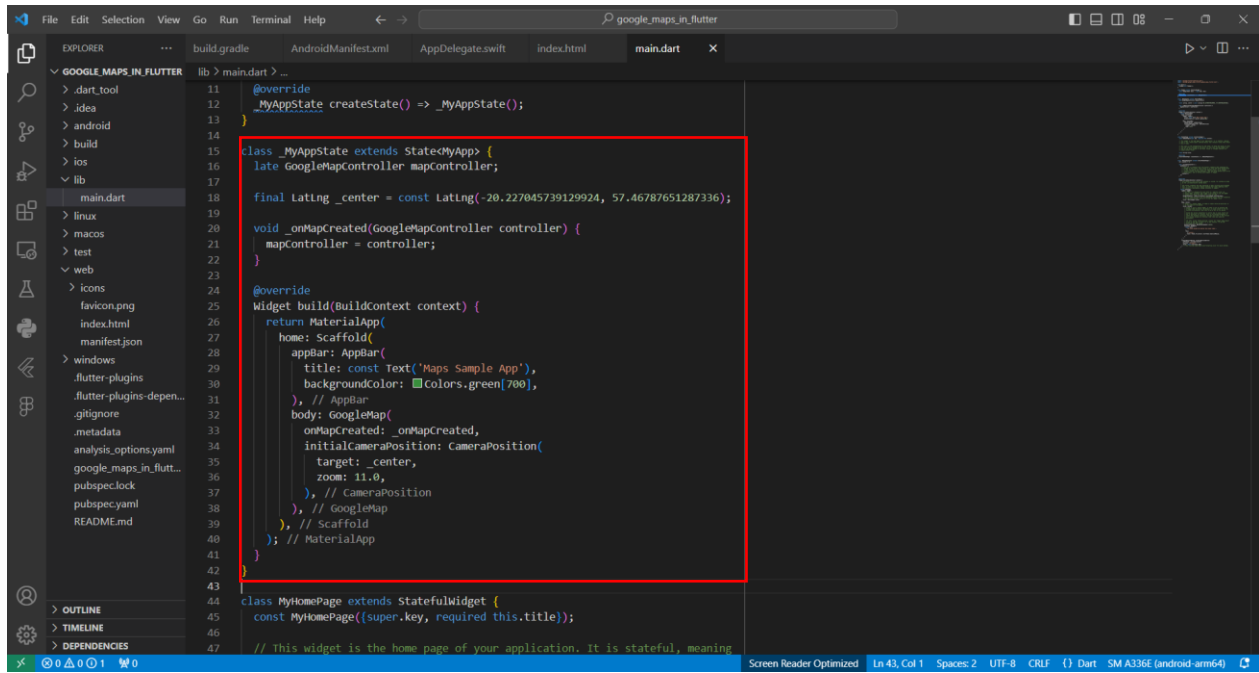
2- Ajouter une clé API pour une application ios



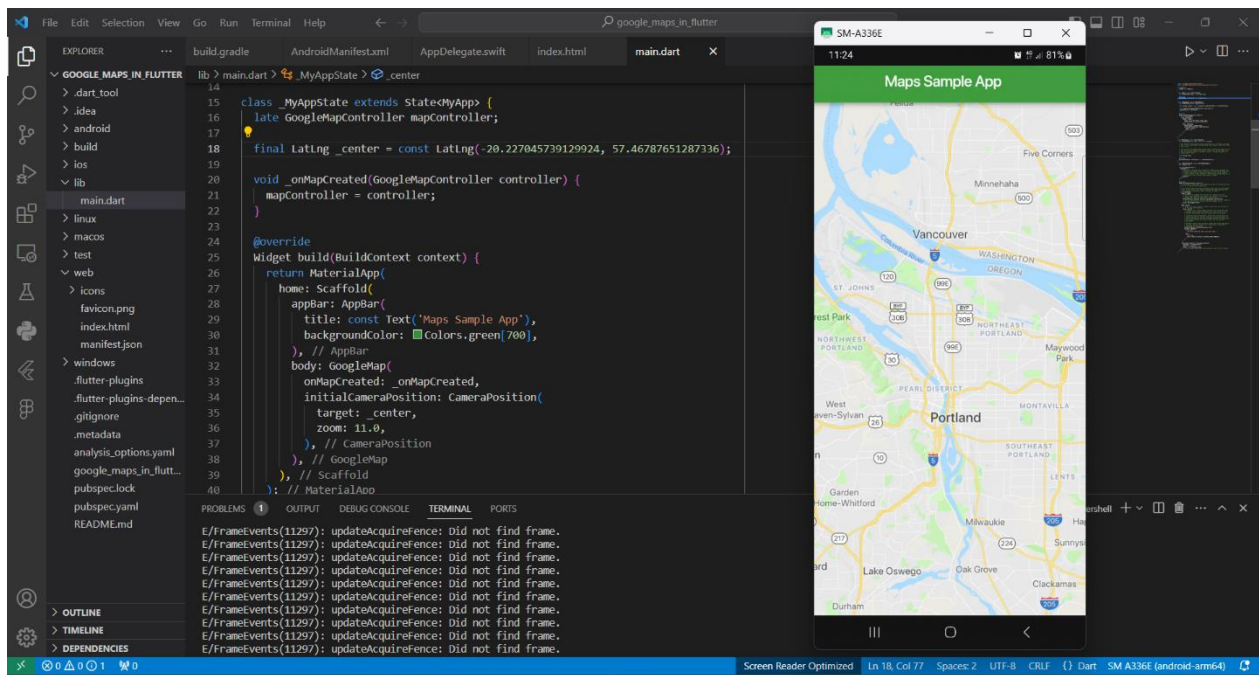
3- Ajouter une clé API pour une application Web



4- Ajouter une carte à l'écran

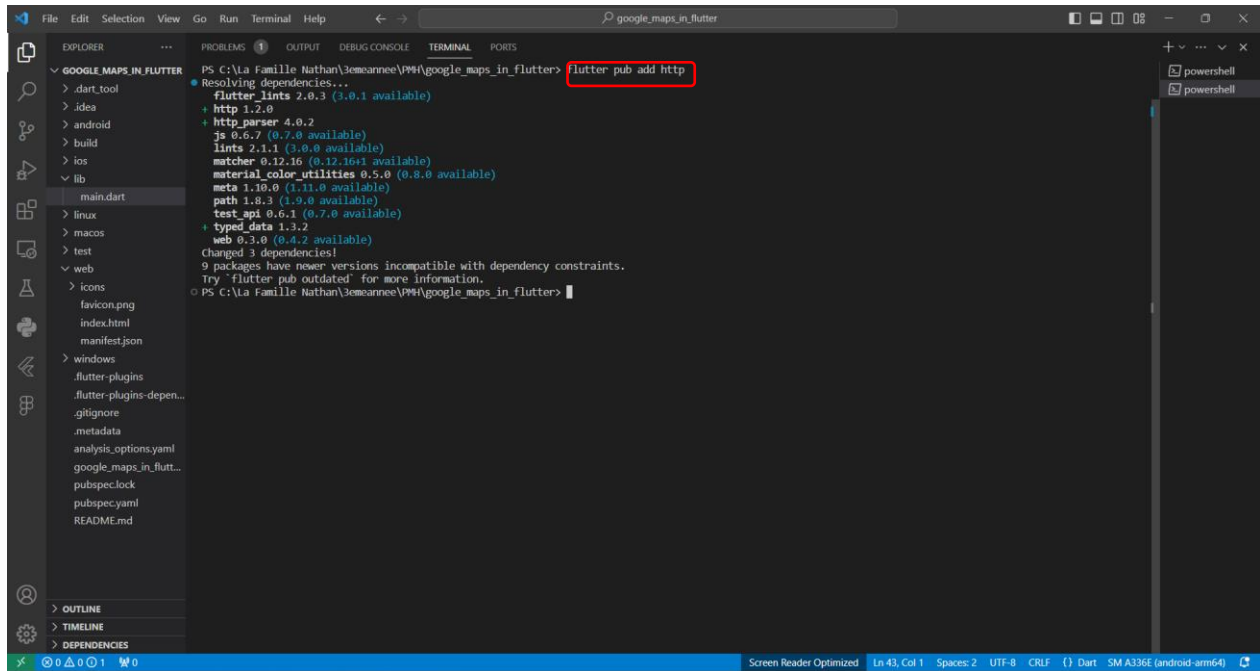


5- Exécution



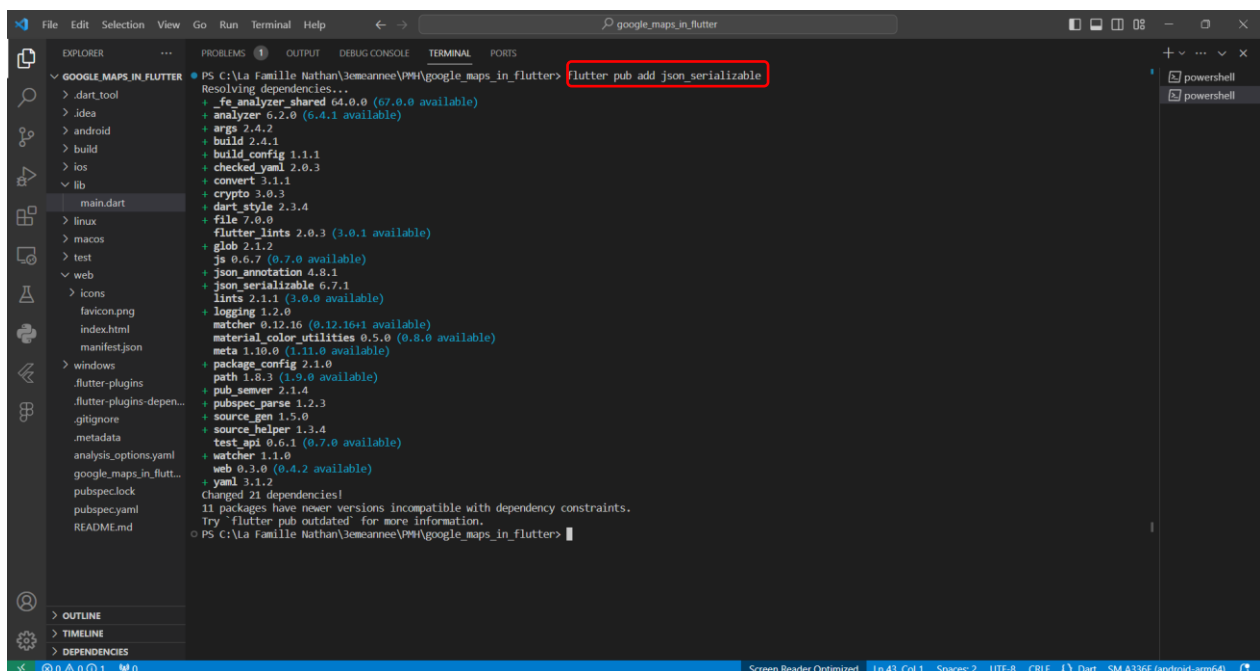
C- Positionner Google sur la carte

- Ajoutez trois dépendances Flutter au projet comme suit. Tout d'abord, ajoutez-le package http pour effectuer facilement des requêtes HTTP.



```
PS C:\La Famille Nathan\3emeannee\PM\google_maps_in_flutter> flutter pub add http
Resolving dependencies...
+ flutter_lints 2.0.3 (3.0.1 available)
+ http 1.2.0
+ http_parser 4.0.2
+ js 0.6.7 (0.7.0 available)
+ matcher 0.12.16 (0.12.16+1 available)
+ material_color_utilities 0.5.0 (0.8.0 available)
+ meta 1.10.0 (1.11.0 available)
+ path 1.8.3 (1.9.0 available)
+ test_api 0.6.1 (0.7.0 available)
+ typed_data 1.3.2
+ web 0.3.0 (0.4.2 available)
Changed 3 dependencies!
9 packages have newer versions incompatible with dependency constraints.
Try 'flutter pub outdated' for more information.
PS C:\La Famille Nathan\3emeannee\PM\google_maps_in_flutter>
```

- Ensuite, ajoutez json_serializable pour déclarer la structure d'objets représentant des documents JSON.



```
PS C:\La Famille Nathan\3emeannee\PM\google_maps_in_flutter> flutter pub add json_serializable
Resolving dependencies...
+ _fe_analyzer_shared 64.0.0 (67.0.0 available)
+ analyzer 6.2.0 (6.4.1 available)
+ args 2.4.2
+ build 2.4.1
+ build_config 1.1.1
+ checked_yaml 2.0.3
+ convert 3.1.1
+ crypto 3.0.3
+ dart_style 2.3.4
+ file 7.0.0
+ flutter_lints 2.0.3 (3.0.1 available)
+ glob 2.1.2
+ js 0.6.7 (0.7.0 available)
+ json_annotation 4.8.1
+ json_serializable 6.7.1
+ lints 2.1.1 (3.0.0 available)
+ logging 1.2.0
+ matcher 0.12.16 (0.12.16+1 available)
+ material_color_utilities 0.5.0 (0.8.0 available)
+ meta 1.10.0 (1.11.0 available)
+ package_config 2.1.0
+ path 1.8.3 (1.9.0 available)
+ pub_semver 2.1.4
+ pubspec_parse 1.2.3
+ source_gen 1.5.0
+ source_helper 1.3.4
+ test_api 0.6.1 (0.7.0 available)
+ watcher 1.1.0
+ web 0.3.0 (0.4.2 available)
+ yaml 3.1.2
Changed 21 dependencies!
11 packages have newer versions incompatible with dependency constraints.
Try 'flutter pub outdated' for more information.
PS C:\La Famille Nathan\3emeannee\PM\google_maps_in_flutter>
```

- Enfin, ajoutez build_runner comme dépendance au moment du développement. Elle sera utilisée pour générer du code ultérieurement.

```

PS C:\La Famille Nathan\3emeannee\PM\google_maps_in_flutter> flutter pub add --dev build_runner
Resolving dependencies...
  fe_analyzer_shared 64.0.0 (67.0.0 available)
  analyzer 6.2.0 (6.4.1 available)
  build_daemon 4.0.1
  build_resolvers 2.4.2
  build_runner 2.4.0
  build_runner_core 7.3.0
  built_collection 5.1.1
  built_value 8.9.0
  code_builder 4.10.0
  fixnum 1.1.0
  flutter_lints 2.0.3 (3.0.1 available)
  frontend_server_client 3.2.0
  graphs 2.3.1
  http_multi_server 3.2.1
  io 1.0.4
  js 0.6.7 (0.7.0 available)
  lints 2.1.1 (3.0.0 available)
  matcher 0.12.16 (0.12.16+1 available)
  material_color_utilities 0.5.0 (0.8.0 available)
  meta 1.10.0 (1.11.0 available)
  mime 1.0.5
  path 1.8.3 (1.9.0 available)
  pool 1.5.1
  shelf 1.4.1
  shelf_web_socket 1.0.4
  test_api 0.6.1 (0.7.0 available)
  timing 1.0.1
  web 0.3.0 (0.4.2 available)
  web_socket_channel 2.4.0 (2.4.3 available)
Changed 18 dependencies!
12 packages have newer versions incompatible with dependency constraints.
Try 'flutter pub outdated' for more information.
PS C:\La Famille Nathan\3emeannee\PM\google_maps_in_flutter>

```

1- Analyser le contenu JSON avec la génération de code

Notez que les données JSON renvoyées par le point de terminaison de l'API présentent une structure régulière. Vous pouvez générer le code afin de rassembler ces données dans des objets que vous pouvez utiliser dans le code.

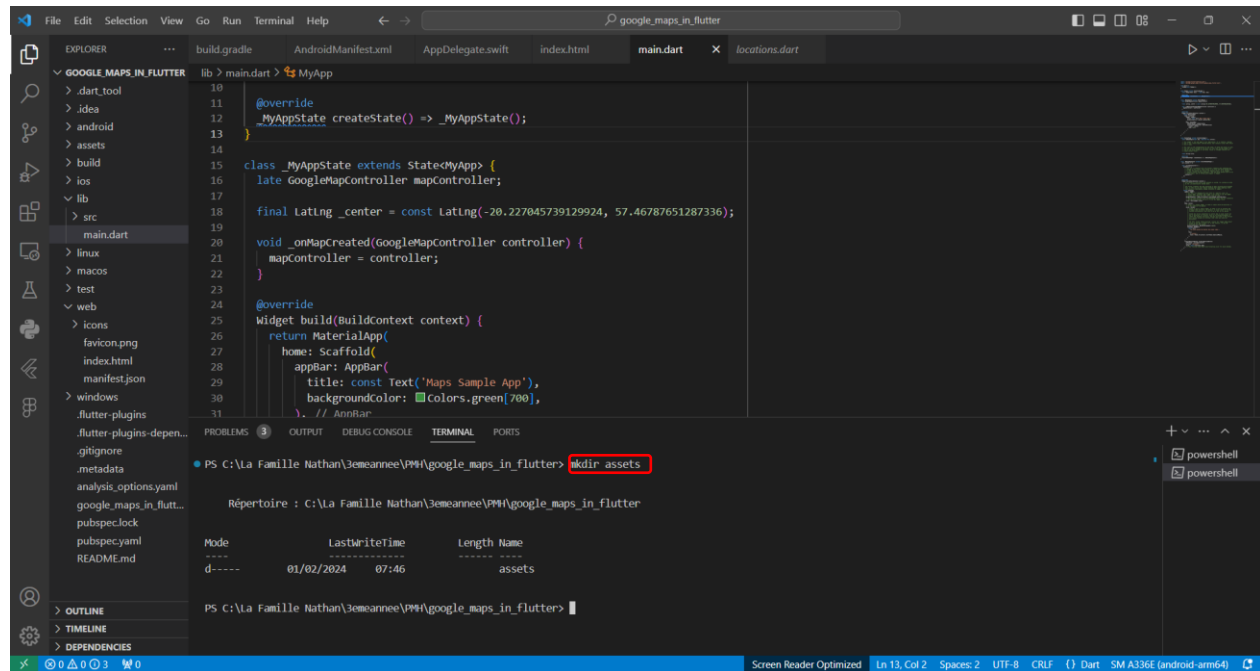
```

PS C:\La Famille Nathan\3emeannee\PM\google_maps_in_flutter> flutter pub run build_runner build --delete-conflicting-outputs
Deprecated. Use 'dart run' instead.
Building package executable... (9.0s)
Built build_runner:build_runner.
[INFO] Generating build script completed, took 393ms
[INFO] Precompiling build script... completed, took 6.8s
[INFO] Building new asset graph completed, took 1.1s
[INFO] Checking for unexpected pre-existing outputs. completed, took 1ms
[INFO] Generating SDK summary completed, took 4.9s
[WARNING] json_serializable on lib/src/locations.dart:
You are missing a required dependency on json_annotation in the "dependencies" section of your pubspec with a lower bound of at least "4.8.1".
[INFO] Running build completed, took 16.9s
[INFO] Caching finalized dependency graph completed, took 68ms
[INFO] Succeeded after 17.0s with 53 outputs (109 actions)
PS C:\La Famille Nathan\3emeannee\PM\google_maps_in_flutter>

```

Le code devrait à nouveau être analysé correctement. Vous devez ensuite ajouter le fichier locations.json utilisé dans la fonction getGoogleOffices. En effet, les données statiques chargées dans cette fonction sont diffusées sans en-têtes CORS. Elles ne seront donc pas chargées dans un navigateur Web. Les applications Flutter sur Android et iOS n'ont pas besoin d'en-têtes CORS, mais l'accès aux données mobiles est souvent difficile.

Accédez à <https://about.google/static/data/locations.json> dans votre navigateur, puis enregistrez le contenu dans le répertoire des éléments. Vous pouvez également utiliser la ligne de commande suivante :



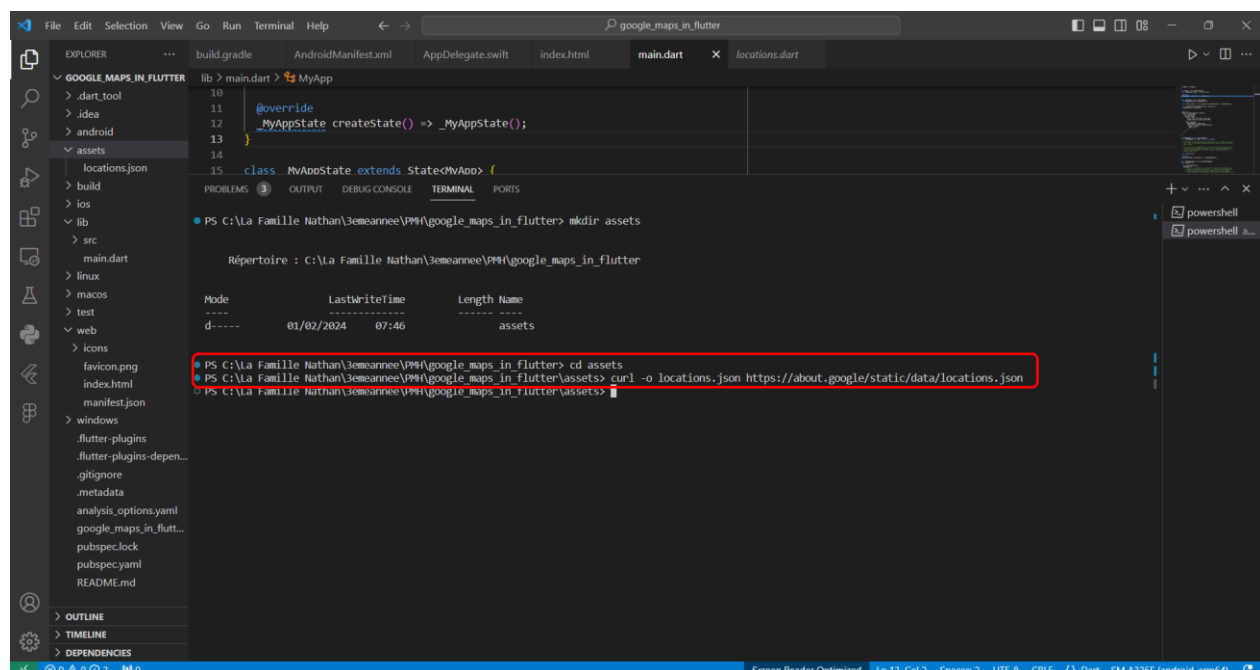
```
File Edit Selection View Go Run Terminal Help
google_maps_in_flutter
main.dart locations.dart

GOOGLE MAPS IN FLUTTER
lib > main.dart > MyApp
10
11 @override
12 _MyAppState createState() => _MyAppState();
13
14
15 class _MyAppState extends State<MyApp> {
16   late GoogleMapController mapController;
17
18   final LatLng _center = const LatLng(-20.227845739129924, 57.46787651287336);
19
20   void _onMapCreated(GoogleMapController controller) {
21     mapController = controller;
22   }
23
24   @override
25   Widget build(BuildContext context) {
26     return MaterialApp(
27       home: Scaffold(
28         appBar: AppBar(
29           title: const Text('Maps Sample App'),
30           backgroundColor: Colors.green[700],
31         ), // End of AppBar
32       ),
33     );
34   }
35 }

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS C:\La Famille Nathan\3emeannee\PM\google_maps_in_flutter> mkdir assets
Répertoire : C:\La Famille Nathan\3emeannee\PM\google_maps_in_flutter

Mode LastWriteTime Length Name
----
d----- 01/02/2024 07:46 assets

PS C:\La Famille Nathan\3emeannee\PM\google_maps_in_flutter>
```



```
File Edit Selection View Go Run Terminal Help
google_maps_in_flutter
main.dart locations.dart

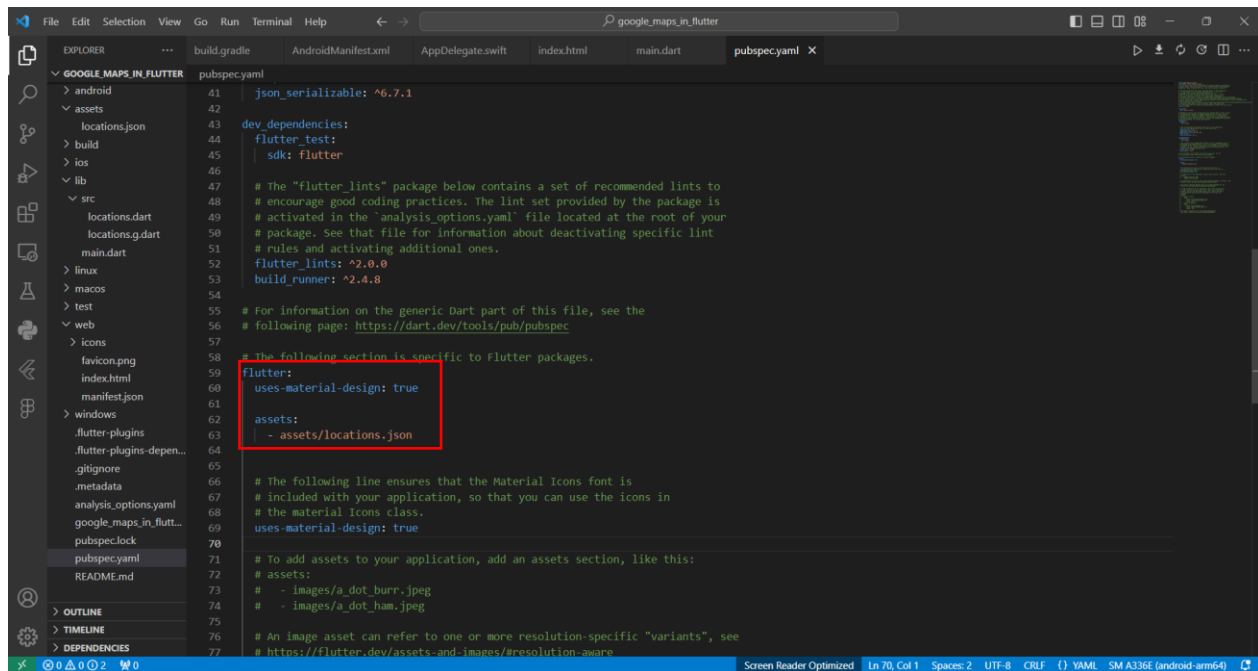
GOOGLE MAPS IN FLUTTER
lib > main.dart > MyApp
10
11 @override
12 _MyAppState createState() => _MyAppState();
13
14
15 class _MyAppState extends State<MyApp> {
16   late GoogleMapController mapController;
17
18   final LatLng _center = const LatLng(-20.227845739129924, 57.46787651287336);
19
20   void _onMapCreated(GoogleMapController controller) {
21     mapController = controller;
22   }
23
24   @override
25   Widget build(BuildContext context) {
26     return MaterialApp(
27       home: Scaffold(
28         appBar: AppBar(
29           title: const Text('Maps Sample App'),
30           backgroundColor: Colors.green[700],
31         ), // End of AppBar
32       ),
33     );
34   }
35 }

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS C:\La Famille Nathan\3emeannee\PM\google_maps_in_flutter> mkdir assets
Répertoire : C:\La Famille Nathan\3emeannee\PM\google_maps_in_flutter

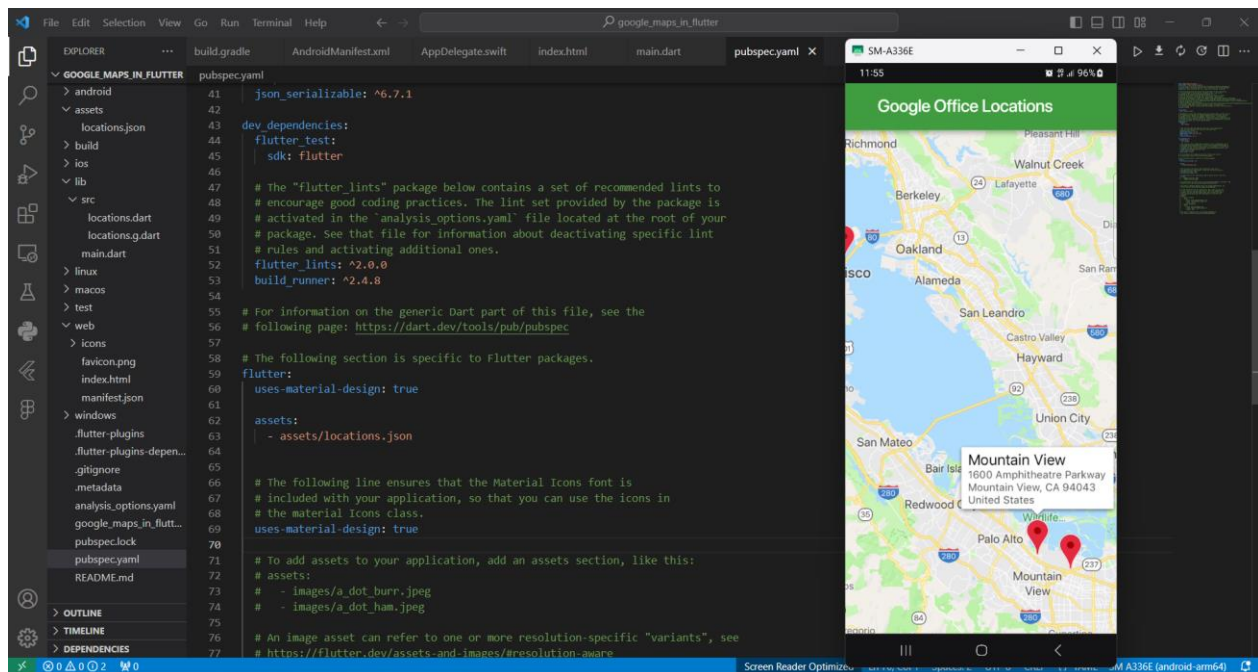
Mode LastWriteTime Length Name
----
d----- 01/02/2024 07:46 assets

PS C:\La Famille Nathan\3emeannee\PM\google_maps_in_flutter> cd assets
PS C:\La Famille Nathan\3emeannee\PM\google_maps_in_flutter\assets> curl -o locations.json https://about.google/static/data/locations.json
PS C:\La Famille Nathan\3emeannee\PM\google_maps_in_flutter\assets> cd ..
```

Une fois que vous avez téléchargé le fichier contenant les éléments, ajoutez-le à la section Flutter du fichier pubspec.yaml.



Modifiez le fichier main.dart pour demander les données cartographiques, puis utilisez les informations renvoyées pour ajouter les adresses à la carte :



Voici une capture d'écran de ce que vous avez réussi à faire : À ce stade, vous pouvez y ajouter de nombreux éléments intéressants

Table de Matière

A- Premiers pas	1
1- Premiers pas avec Flutter	1
2- Ajouter le plug-in Flutter pour Google Maps en tant que dépendance	1
3- Configurer Android minSDK.....	3
B- Ajouter Google Maps à l'application	4
1- Ajouter une clé API pour une application Android.....	4
2- Ajouter une clé API pour une application ios	7
3- Ajouter une clé API pour une application Web	7
4- Ajouter une carte à l'écran	8
5- Exécution	8
C- Positionner Google sur la carte	9
1- Analyser le contenu JSON avec la génération de code.....	10