



NG Maps

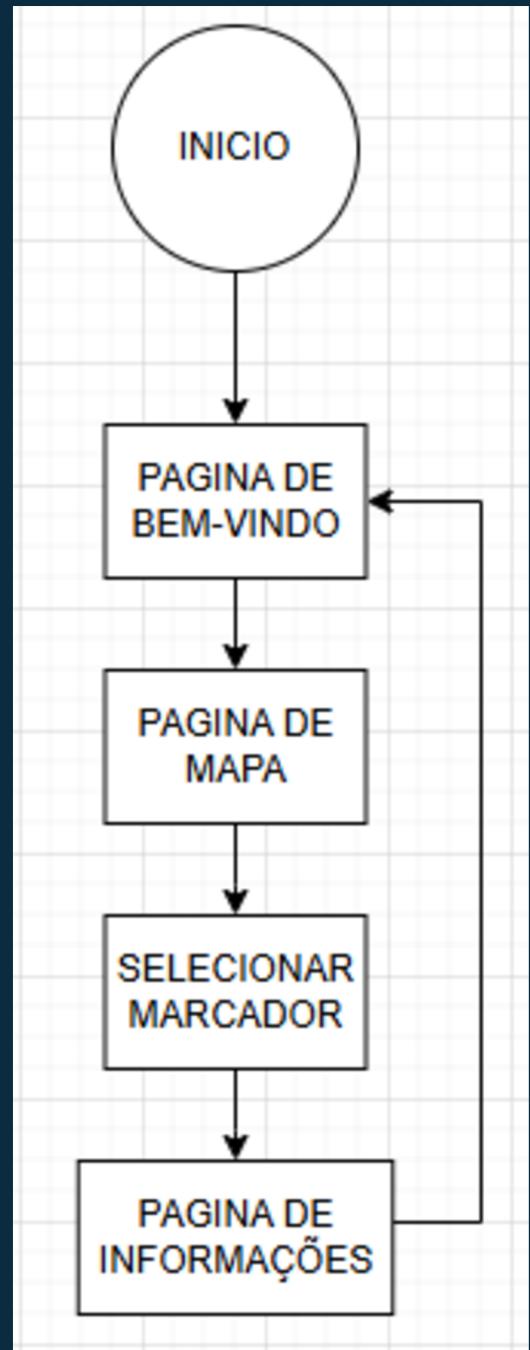
Adriel Ribeiro | Andre Duarte | Felipe Matheus | Nycolas Felix

Por que o ONG Maps foi criado?



O ONG Maps foi criado com a intenção de suprir a necessidade de pessoas que desejam contribuir com causas sociais ou pessoas que procuram ajuda especializada e não sabem onde encontrar. Apesar da crescente conscientização sobre a importância de contribuir com causas sociais, a falta de visibilidade impede as ONGs de alcançarem seu público alvo, surgindo assim, o ONG Maps para resolver esse problema. Sem uma ferramenta de centralizada, encontrar ONGs que atuem em áreas específicas perto da sua localização se torna um obstáculo, fazendo as pessoas recorrerem a pesquisas que podem levar a resultados desatualizados, isso limita o alcance tanto das ONGs a chegarem ao seu público alvo quanto das pessoas que estão procurando por esses serviços.

Funcionamento



Ao iniciar o aplicativo, uma tela de bem-vindo se abre e ao clicar no botão para prosseguir, o aplicativo é direcionado para a tela do mapa, onde aparecerão marcadores da localização das ONGs. Clicando em algum marcador, uma tela de informações da ONG se abrirá, mostrando informações como: nome, horário de funcionamento, fotos, uma breve descrição dos objetivos da ONG, etc.

implementando API



Para que esse aplicativo fosse possível, utilizamos uma API (Interface de Programação de Aplicação) do Google Maps.

Passo a passo para a implementação da API:

Código:

1- Defina o [minSdkVersionem android/app/build.gradle](#):

```
android {  
    defaultConfig {  
        minSdkVersion 20  
    }  
}
```

Isso significa que o aplicativo só estará disponível para usuários que executam o Android SDK 20 ou superior.

2- Especifique sua chave de API no manifesto do aplicativo [android/app/src/main/AndroidManifest.xml](#):

```
<manifest ...>  
    <application ...>  
        <meta-data  
            android:name="com.google.android.geo.API_KEY" android:value="YOUR KEY  
HERE"/>
```

Instalação:

Com Flutter:

```
$ flutter pub add google_maps_flutter
```

Isso adicionará uma linha como esta ao **pubspec.yaml** do seu pacote (e executará um implícito flutter pub get):

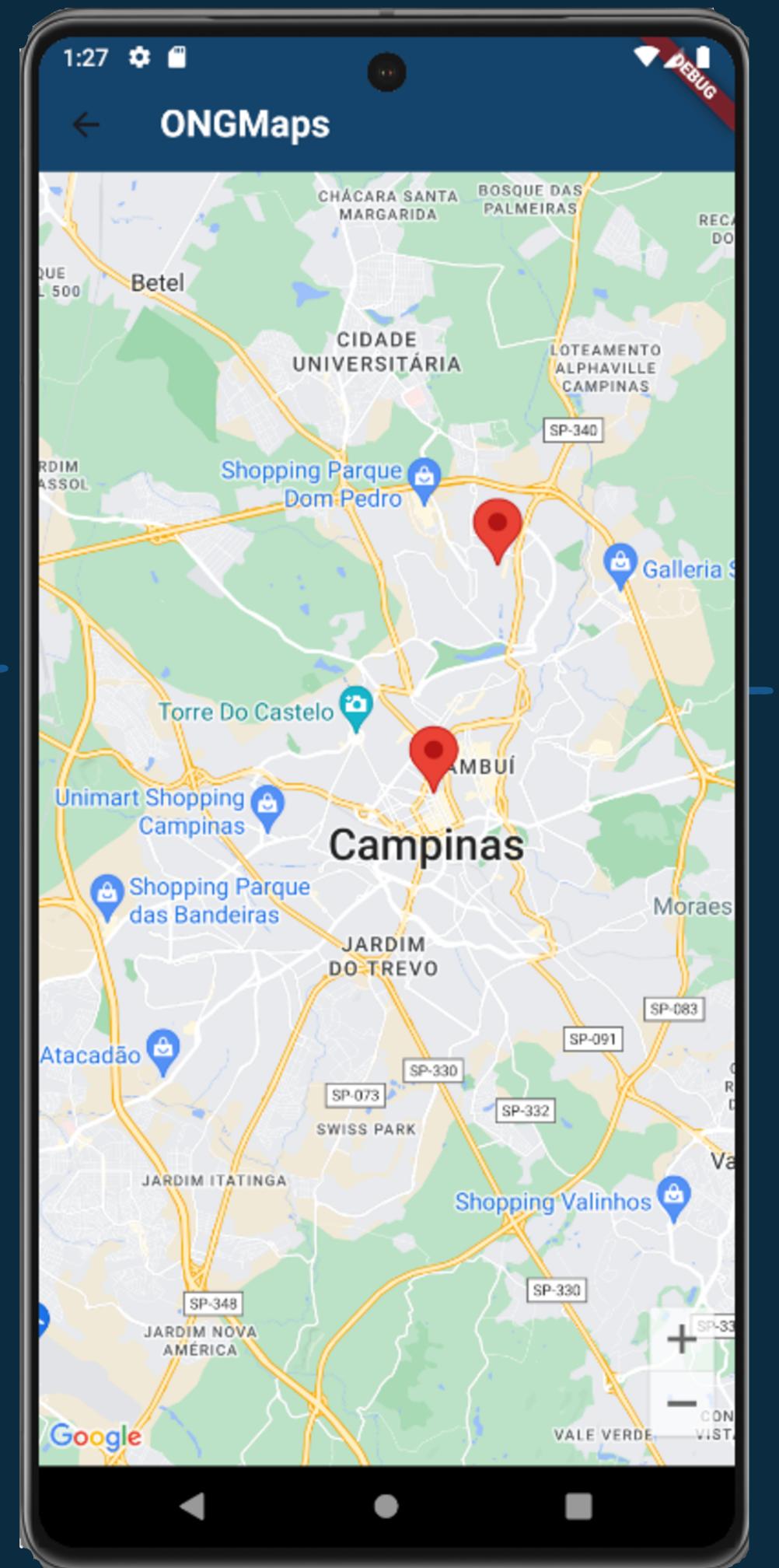
```
dependencies:google_maps_flutter: ^2.10.0
```

Alternativamente, seu editor pode oferecer suporte a flutter pub get. Verifique os documentos do seu editor para saber mais.

Importe-o

Agora, no seu código Dart, você pode usar:

```
import 'package:google_maps_flutter/google_maps_flutter.dart';
```



Marcadores



Código:

1- Defina os marcadores:

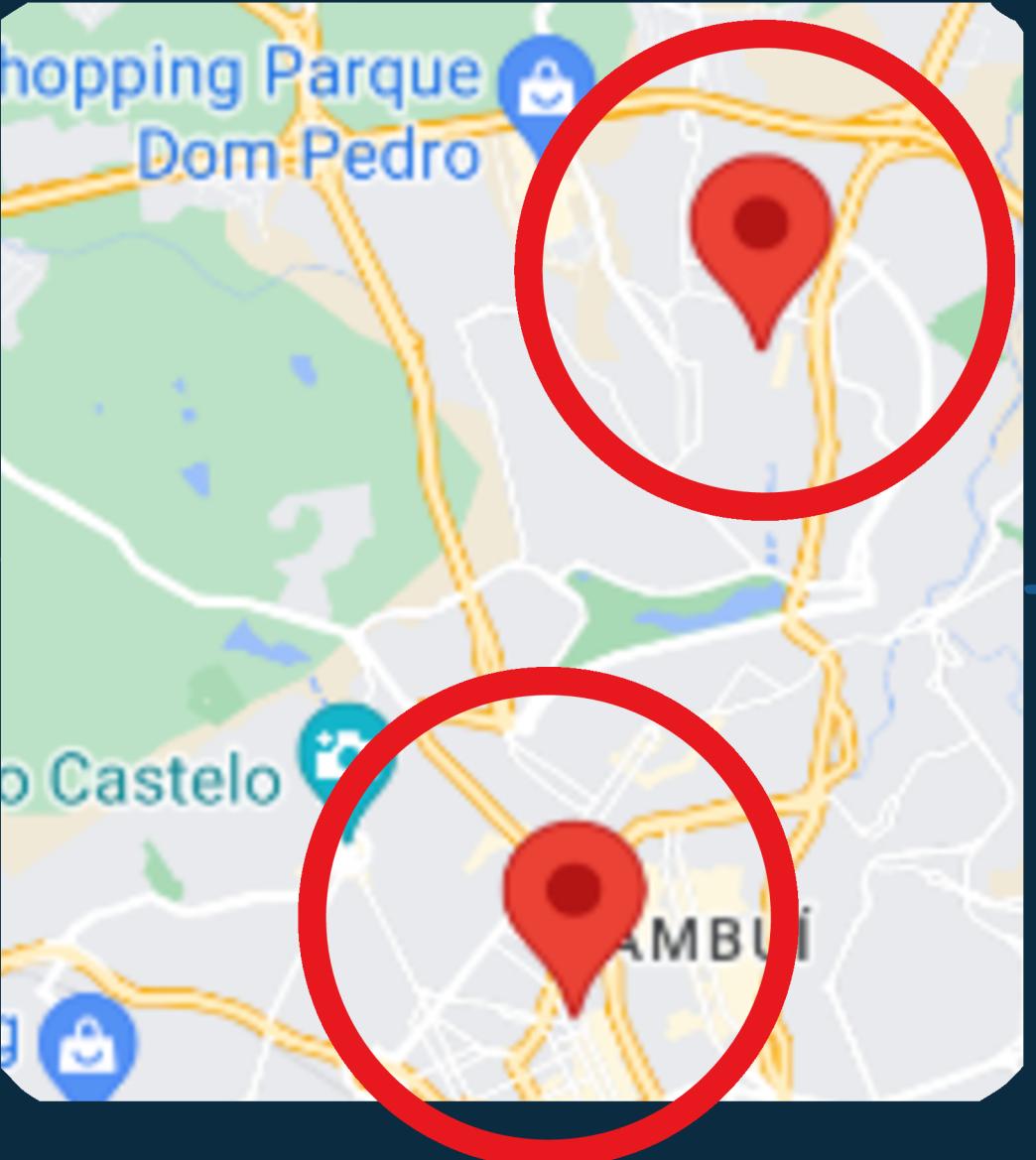
```
Set<Marker> markers = {} // Marcadores.
```

2 - Crie os marcadores

```
vvoid displayInfo() {
  for (int i = 0; i < latlongPoint.length; i++) {
    markers.add(
      Marker(
        markerId: MarkerId(i.toString()),
        icon: BitmapDescriptor.defaultMarker,
        position: latlongPoint[i],
        onTap: () {
          _customInfoWindowController.addInfoWindow!(
            _buildInfoWindow(locationNames[i], locationImages[i], i),
            latlongPoint[i],
          );
        },
      ),
    );
  }
}
setState(() {});}
```

3. Exibição dos Marcadores no Mapa:

```
GoogleMap(
  initialCameraPosition: const CameraPosition(
    target: LatLng(-22.912847283240453, -47.041346547261014),
    zoom: 12,
  ),
  markers: markers, // Aqui os marcadores são passados para o mapa
  onTap: (LatLng position) {
    _customInfoWindowController.hideInfoWindow();
  },
  onCameraMove: (position) {
    _customInfoWindowController.onCameraMove();
  },
  onMapCreated: (GoogleMapController controller) {
    _customInfoWindowController.googleMapController = controller;
  },
),
```



Janelas Customizadas

Código:

1- Definição do Controlador da Janela de Informações:

```
final CustomInfoWindowController _customInfoWindowController =  
CustomInfoWindowController();
```

2-exibição da Janela de Informações:

```
onTap: () {  
    _customInfoWindowController.addInfoWindow!(  
        _buildInfoWindow(locationNames[i], locationImages[i], i),  
        latlongPoint[i],  
    );  
},
```

3 - Codigo semelhante a uma tela com containers botões e etc.

4 - Exibição da Janela de Informações:

```
CustomInfoWindow(  
    controller: _customInfoWindowController,  
    height: 190,  
    width: 250,  
    offset: 35,  
,
```

Instalação:

Com Flutter:

```
$ flutter pub add custom_info_window
```

Isso adicionará uma linha como esta ao **pubspec.yaml** do seu pacote
(e executará um implícito flutter pub get):

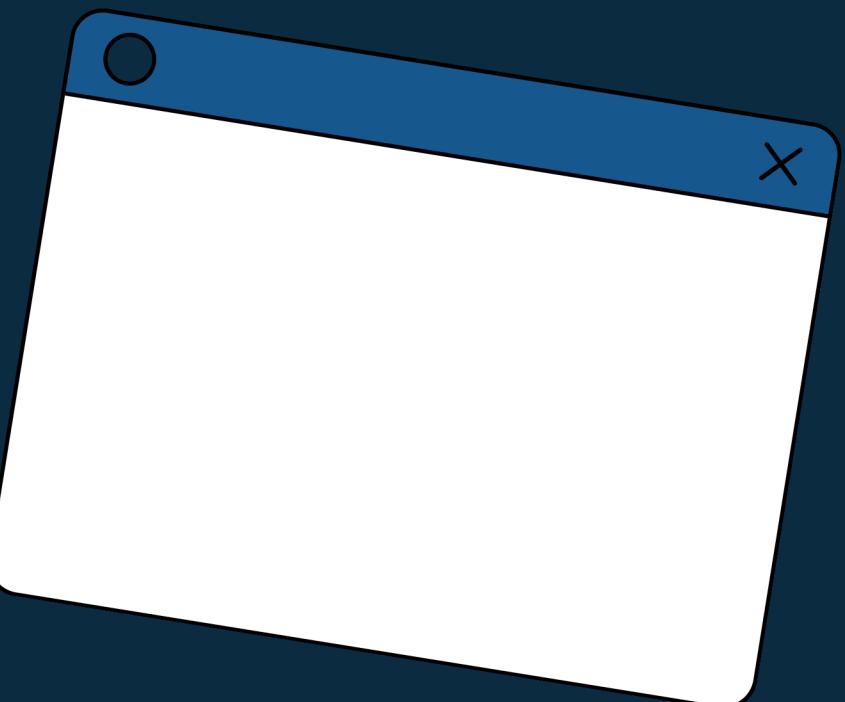
```
custom_info_window: ^1.0.1
```

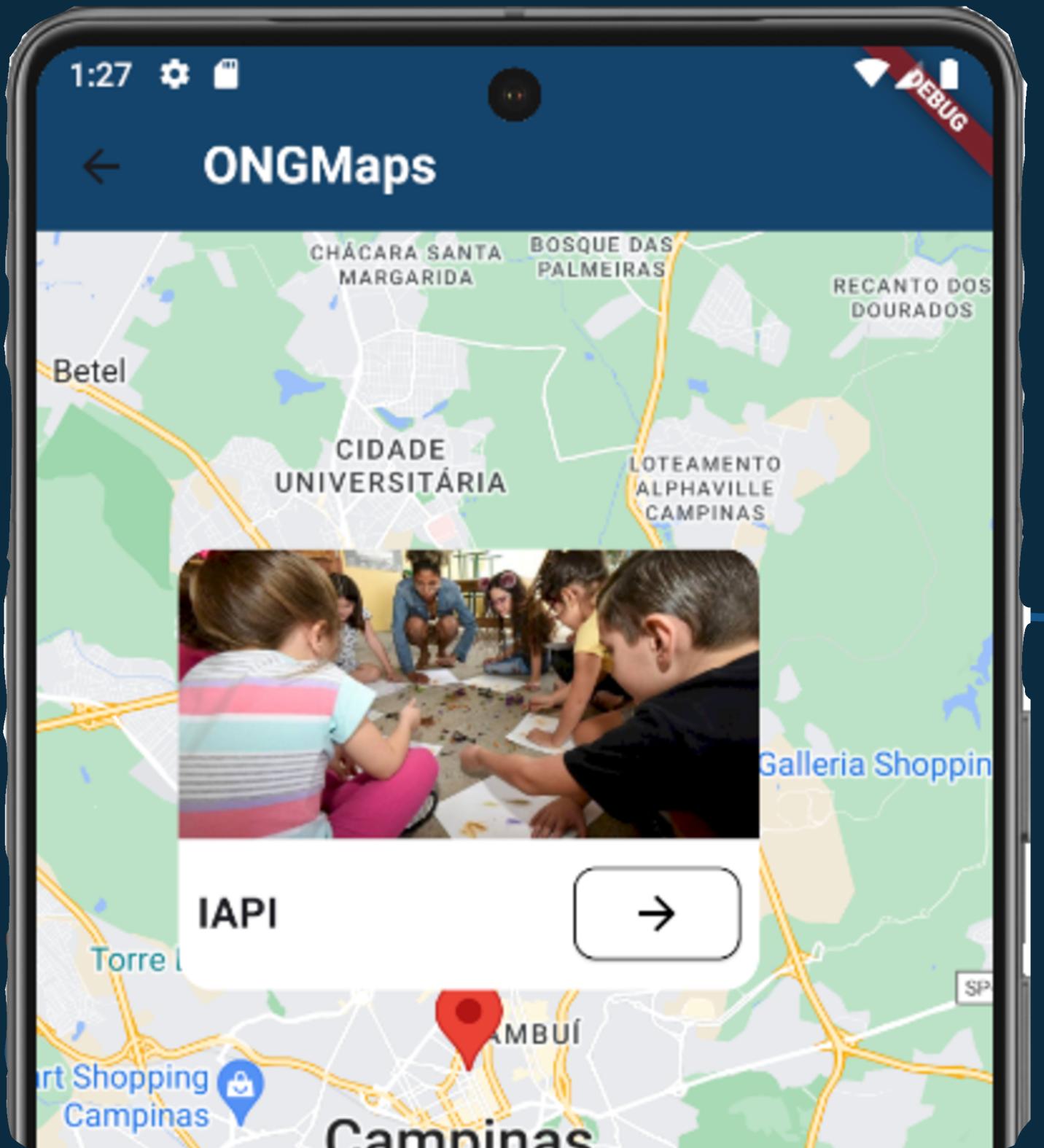
Alternativamente, seu editor pode oferecer suporte a flutter pub get. Verifique
os documentos do seu editor para saber mais.

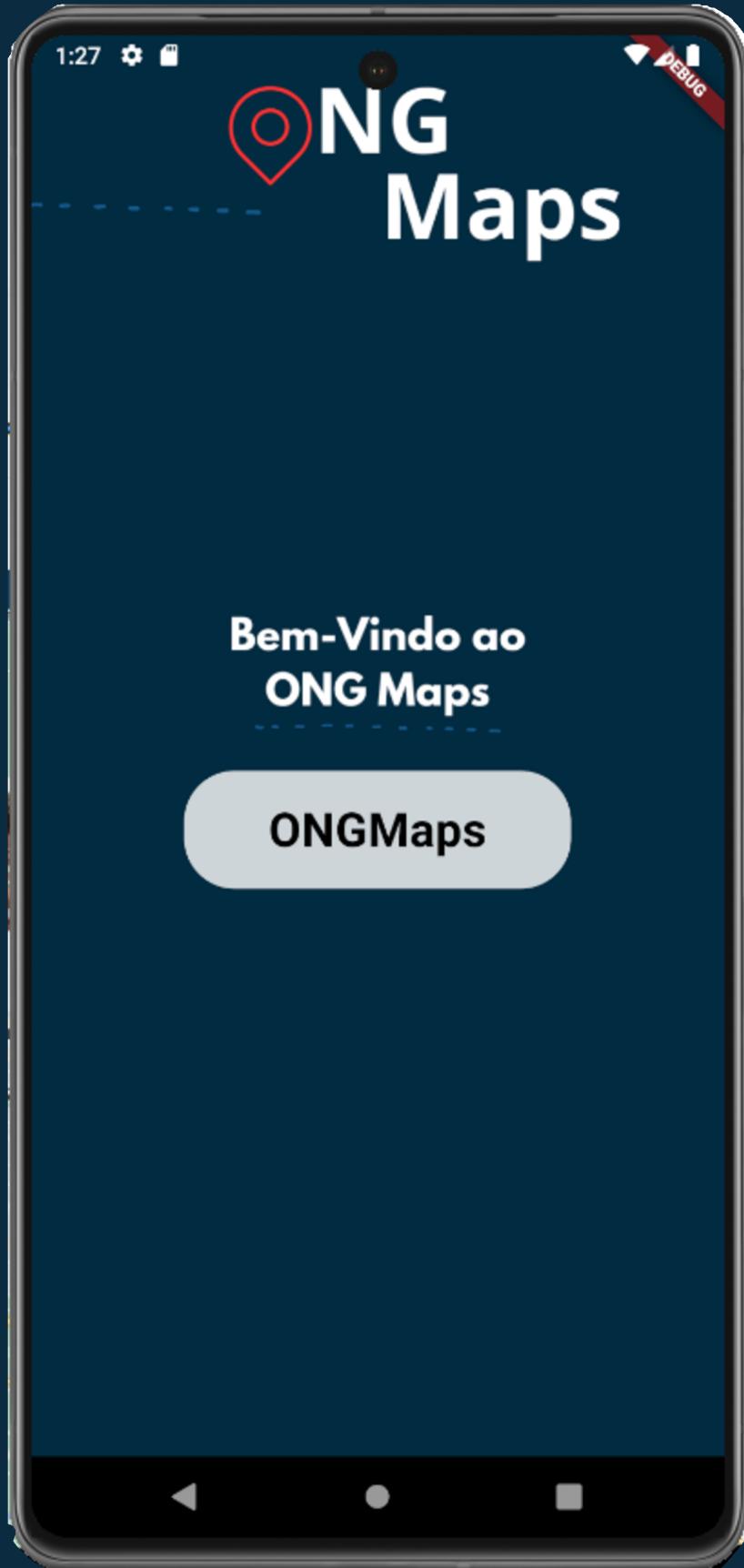
Importe-o

Agora, no seu código Dart, você pode usar:

```
import 'package:custom_info_window/custom_info_window.dart';
```



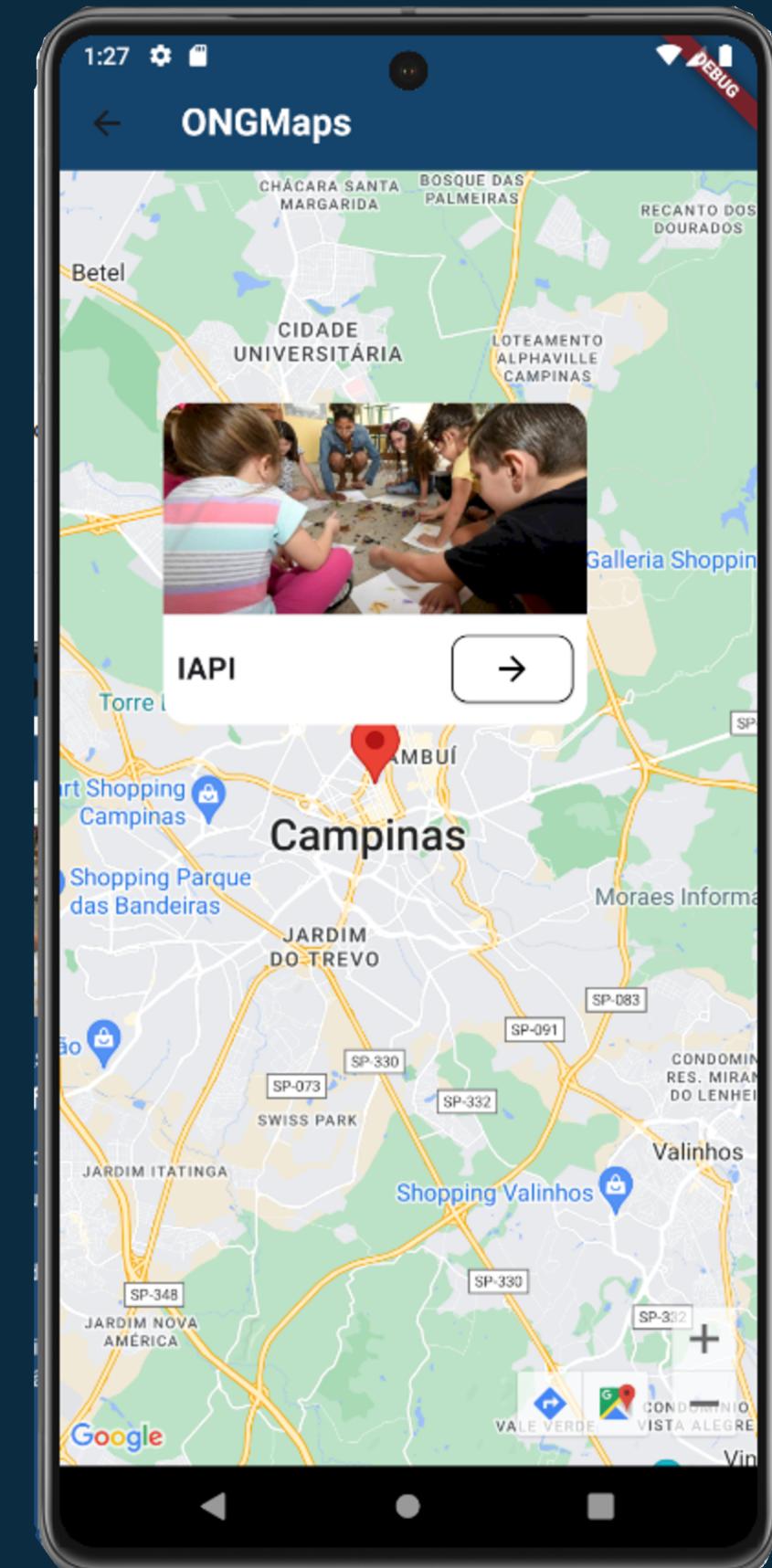


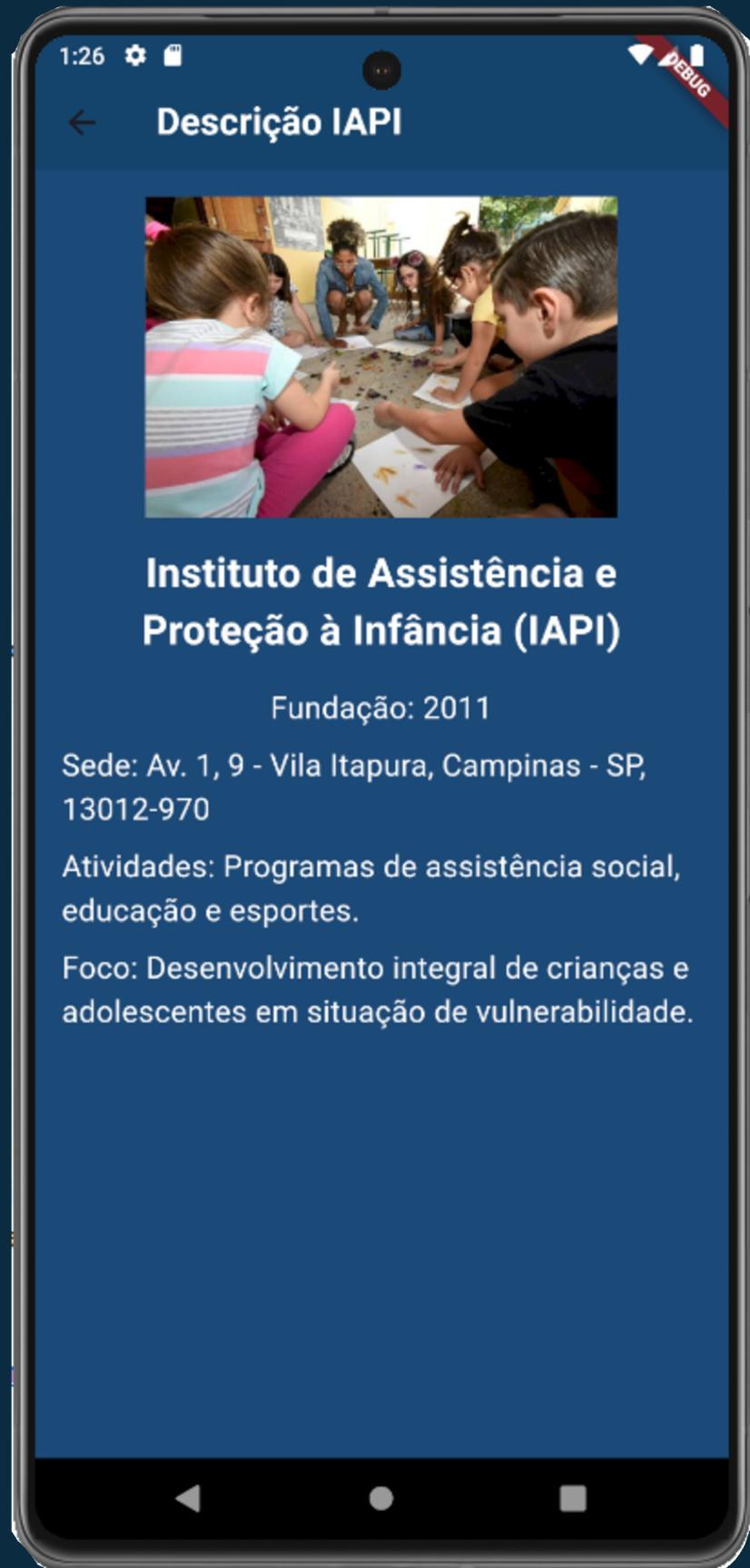


Abrindo o aplicativo, é exibida uma tela de “Bem-vindo”, onde ao clicar no botão ONG Maps, você é direcionado para a tela de mapa



A tela do mapa é bem semelhante a do Google Maps, porém com ONGs destacadas no mapa, que ao selecionar alguma ONG, aparece a sigla do nome da ONG e um botão que direciona para a tela de descrição





Na tela de descrição aparece informações mais detalhadas sobre a ONG como: Nome, ano de fundação, endereço da sede, atividades que a ONG realiza, e os objetivo em foco da ONG.