

Usando el caché para cargar imágenes más rápido



[anncode](#)

27 de Mayo de 2019

Sobre lo que llevamos construido en nuestra aplicación notarás que las imágenes de las listas tardan en mostrarse.

Utilizaremos una librería que nos permitirá utilizar el cache para optimizar la carga, es muy fácil de utilizar y puedes encontrar su documentación en el [siguiente enlace](#).

Comenzaremos integrándola como recurso en el archivo `pubspec.yaml` en la sección de dependencias:

```
dependencies:  
  cached_network_image: ^0.8.0
```

En nuestro proyecto estamos reutilizando el `Widget CardImageWithFabIcon` para mostrar una imagen que viene de internet y una que proviene de la carpeta temporal del dispositivo cuando tomamos la foto.

Por esta razón es importante agregar un *flag* que nos permita distinguir si la imagen que mostraremos en el Card viene de internet o es un Asset, además lo pondremos como parámetro en el método constructor.

```
class CardImageWithFabIcon extends StatelessWidget {  
  
  final double height;  
  final double width;  
  double left;  
  final String pathImage;  
  final VoidCallback onPressedFabIcon;  
  final IconData iconData;  
  bool internet = true;  
  
  CardImageWithFabIcon({  
    Key key,  
    @required this.pathImage,  
    @required this.width,  
    @required this.height,  
    @required this.onPressedFabIcon,
```

```

    @required this.iconData,
    this.internet,
    this.left

  });
.
.
.

```

Posteriormente asignaremos la validación en la propiedad image de esta estructura:

```

final card = Container(
  decoration: BoxDecoration(
    image: DecorationImage(
      image:
    ),
  ),
);

```

Anteriormente usábamos el Widget NetworkImage() para mostrarlo pero al no tener la validación cuando capturábamos la foto fallaba, así que pondremos lo siguiente:

```

image:
internet?CachedNetworkImageProvider(pathImage):AssetImage(pathImage)

```

No olvides importar la librería: import

```
'package:cached_network_image/cached_network_image.dart';
```

Aplicalo también para la lista de fotos de perfil en el archivo profile_place.dart

[Mira el proyecto completo aquí](#)

Notarás inmediatamente cómo cambia la velocidad en la carga de las imágenes

[Reportar un problema](#)

StreamController, sink, add y StreamBuilder