



State Management

Con Flutter

El problema con el state

- Desarrollar en Flutter es sencillo, pero también es muy sencillo escribir muy mal código
- En Flutter normalmente escribimos la lógica para modificar el state junto con la interfaz de usuario en un mismo archivo, esto puede causar que tu código se vuelva muy complicado de entender y sobre todo de mantener, es por eso que tenemos lo que se conoce como soluciones de State Management

```
class _HomeState extends State<Home> {
  var pageIndex = 0;
  final GlobalKey<ScaffoldState> _scaffoldKey = new GlobalKey<ScaffoldState>();
  var pages = [HomePage(), PlaylistPage(), SettingsPage()];
  var downloadedItems = 0;
  var totalItems = 0;
  var progress = 0;
  var progressTotal = 0;

  var _isDialogShowing = false;
  var connection = false;
  final failedToConnectSnackBar =
    SnackBar(content: Text("Couldn't connect to server!"));
  final undefinedIpSnackBar =
    SnackBar(content: Text("Server IP is not defined!"));

  @override
  void initState() {
    super.initState();

    var storage = Storage();
    playing = false;
    storage.checkIfInternalFileExists("settings.txt").then((found) {
      if (found) {
        var data = storage.readSettings().then((value) {
          var decodedData = jsonDecode(value);
          setState(() {
            id = decodedData["id"];
            ip = decodedData["ip"];
            name = decodedData["name"];
            lastPlayed = decodedData["lastPlayed"];
          });
          if (ip == null) {
            Scaffold.of(context).showSnackBar(undefinedIpSnackBar);
          } else {
            connection.connect();
          }
          print("Last played $lastPlayed");
          if (lastPlayed != null) {
            Home.changeScreen(lastPlayed);
          }
          print(ip);
        });
      } else {
        var uuid = Uuid();
        id = uuid.v1();
        Map data = {"id": id, "ip": ip, "name": name, "lastPlayed": lastPlayed};
        storage.createSettings(jsonEncode(data));
      }
    });
    storage.requestPermission();

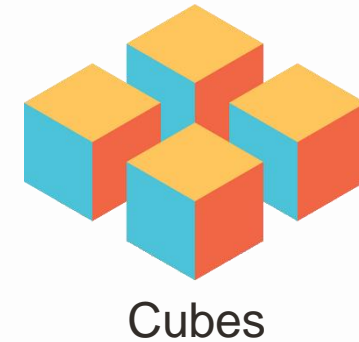
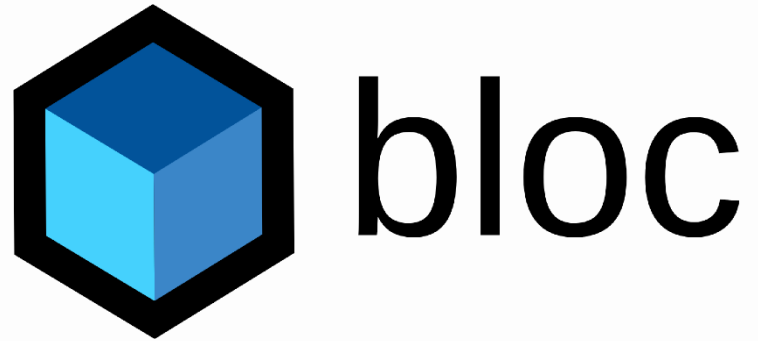
    Connection().stream.listen((data) {
      print(data);

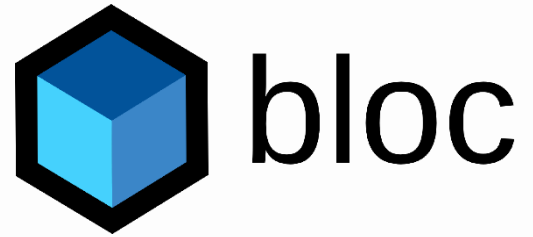
      switch (data) {
        case 0:
          setState(() {
            connection = false;
          });
          break;
        case 1:
          setState(() {
            connection = true;
          });
          break;
        case 3:
          _scaffoldKey.currentState.showSnackBar(failedToConnectSnackBar);
          setState(() {
            connection = false;
          });
          break;
      }
    });

    DownloadImages().stream.listen((data) {
      switch (data[0]) {
        case 0:
          Navigator.of(context).popUntil((route) => route.isFirst);
          Navigator.push(context,
            MaterialPageRoute(builder: (context) => ShowPlaylist(data[1])));
          break;
        case 1:
          if (!_isDialogShowing) _showDialog();

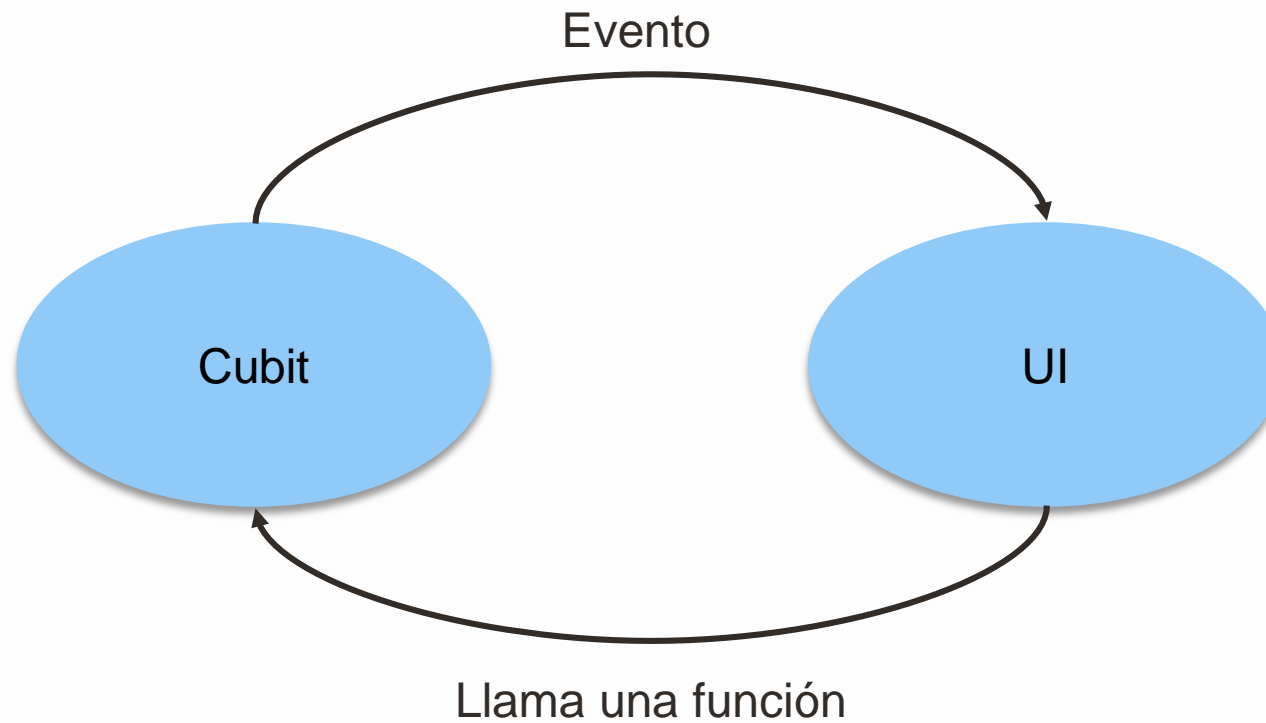
          setState(() {
            progress = data[1];
            progressTotal = data[2];
            totalItems = data[3];
            downloadedItems = data[4];
          });
          break;
      }
    });
  }
}
```

Soluciones de State Management





La librería de bloc nos ayuda a manejar el estado de nuestra aplicación
separando nuestra código de la lógica del código de nuestra interfaz





REST API

¿Qué es un REST API



- Una API de transferencia de estado representacional o REST, es una interfaz para obtener datos o generar operaciones a través de un protocolo HTTP.
- Genera respuestas en formato JSON
- Se pueden integrar en cualquier aplicación





JSON

Un JSON o JavaScript Object Notation es el formato de texto utilizado para la transferencia de datos, está basado en la forma en que escribes objetos en el lenguaje de JavaScript



```
{
  "mal_id": 5680,
  "url": "https://myanimelist.net/anime/5680/K-On",
  "image_url": "https://cdn.myanimelist.net/",
  "title": "K-On!",
  "airing": false,
  "type": "TV",
  "episodes": 13,
  "score": 7.84,
  "start_date": "2009-04-03T00:00:00+00:00",
  "end_date": "2009-06-26T00:00:00+00:00",
  "members": 794312,
  "rated": "PG-13"
}
```