FLUTTER BOOTCAMP

STATE MANAGEMENT

Workshop 4

Speaker:
Rashid Wassan

Table Of Contents

01 Welcome Message

02 About Me

03 What is state?

04 Declarative Approach

05 State Types & Management

06 Provider Intro & SM

07 Repository Pattern

08 To-Do List App

09 Wrap Up

Main kaun Hoon?

Rashid Wassan, a third year Software Engineering undergrad at Mehran University, Jamshoro.

I am a:

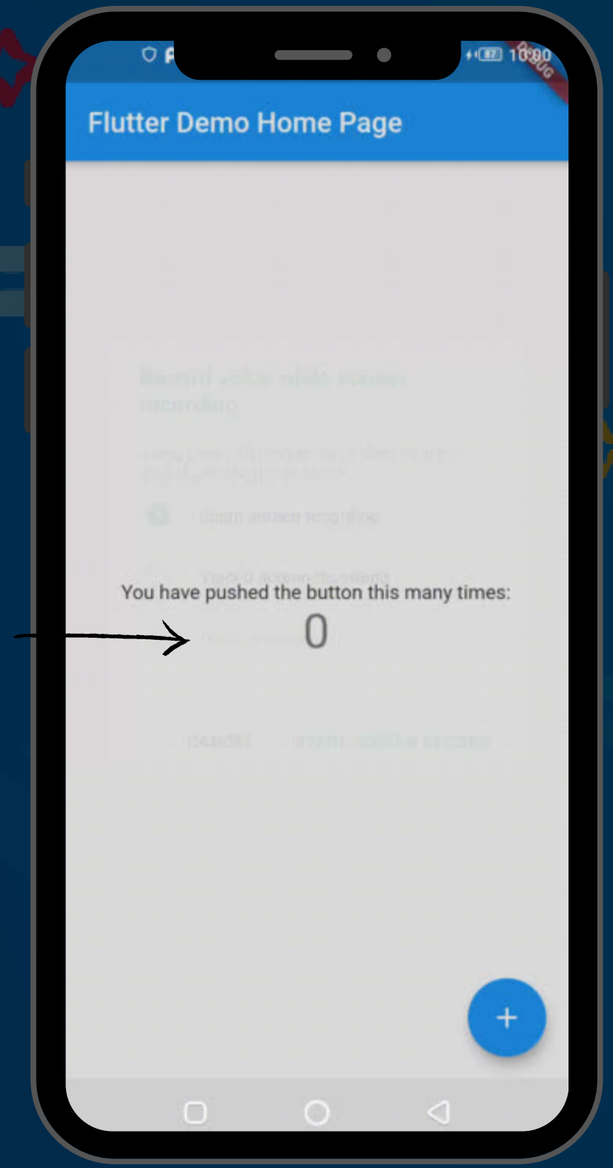
- Flutter Developer at **IsaaTech** (2yr exp)
- Google Developer Student Clubs Lead
- Microsoft Learn Student Ambassador
- Member @ **Strapi.io** & Appwrite.io
- Tech Enthusiast & Evangelist
- Occasional **Freelancer**
- Above all that, a proud **Pakistani**

Reach Me Out At:

rashidwassan.tech

What is State?

In the software development world, the state is the information you required to rebuild your UI at any moment.



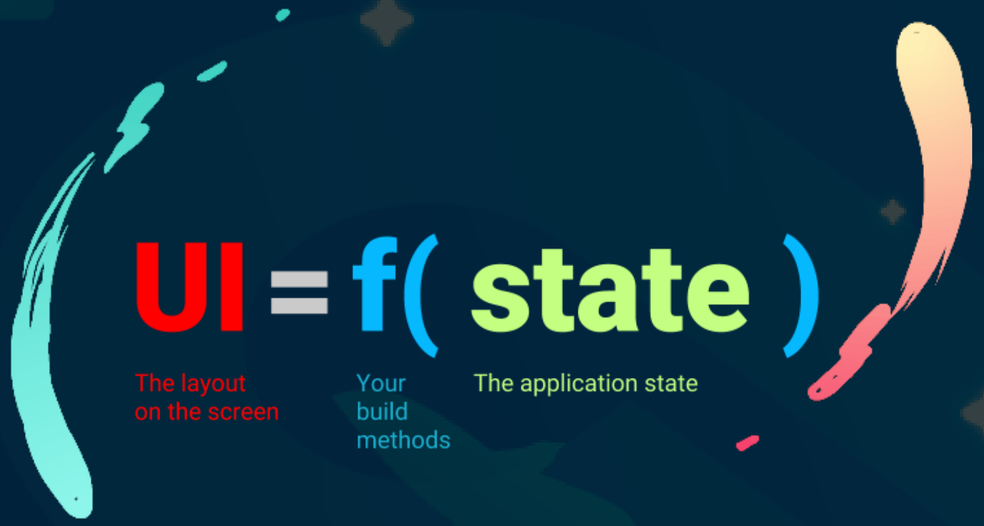
Why State Management?

Does it make any difference??

State management is very important in application development. It centralizes all the states of various UI controls to handle data flow across the application. A proper state management can save a lot of computational overhead by avoiding rebuilding of unecessary widgets.

Thora Declaratively Socho!!!

When the state of your app changes (for example, you press that increment counter button), you change the state, and that triggers a redraw of the user interface. There is no imperative changing of the UI itself (like widget.setText)—you change the state, and the UI rebuilds from scratch.



Flutter is declarative. This means that Flutter builds its user interface to reflect the current state of your app:

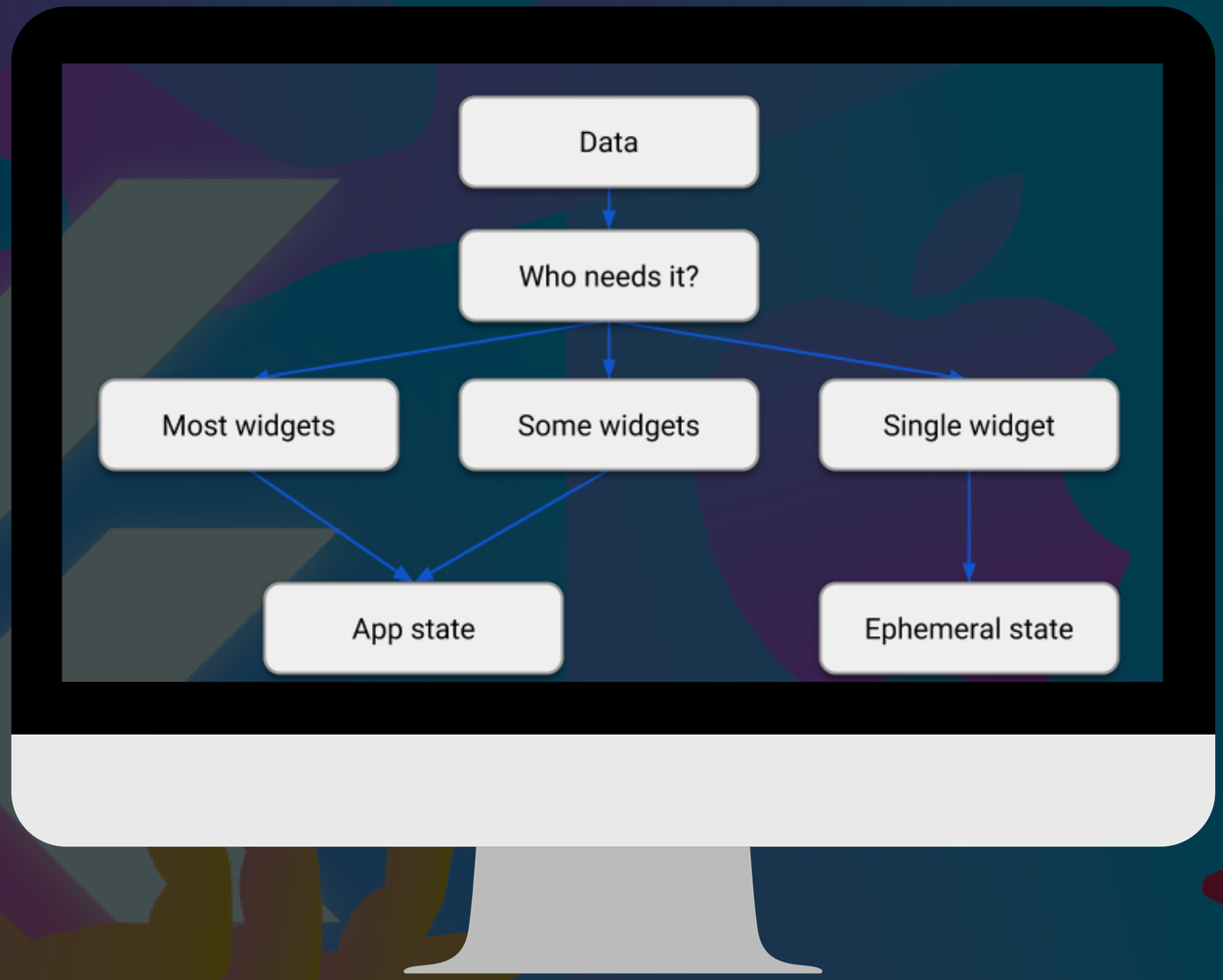
State - Types

1 Ephemeral State

Ephemeral state (sometimes called UI state or local state) is the state you can neatly contain in a single widget.
Example: on/off state of a switch.

2 App State

State that is not ephemeral, that you want to share across many parts of your app, and that you want to keep between user sessions, is what we call application state (sometimes also called shared state).
Example: user login info.



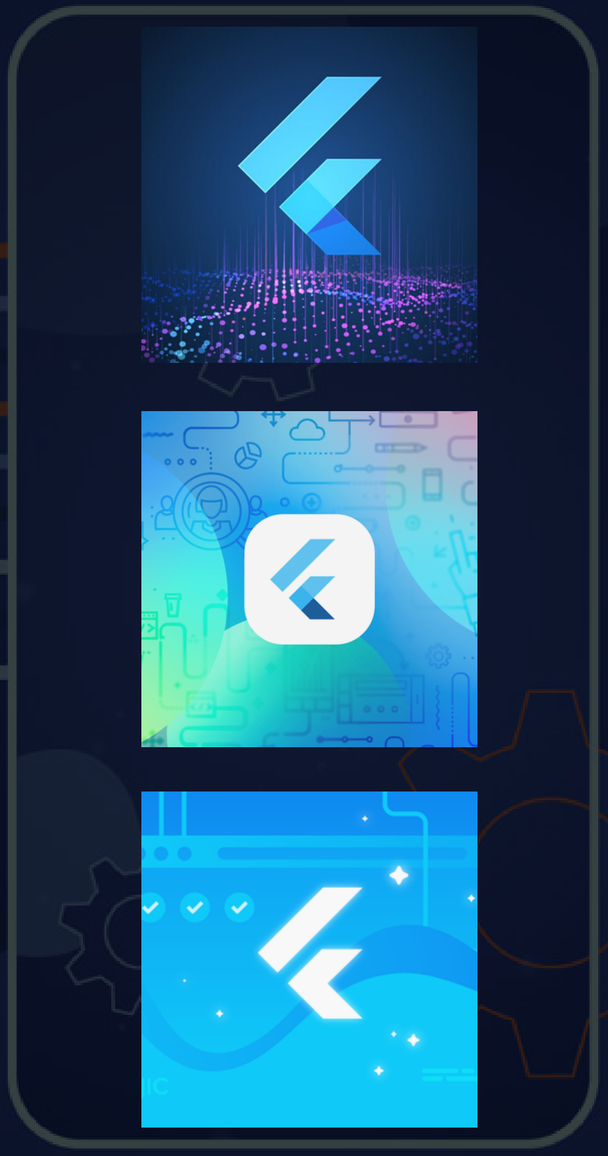
Managing the State

Consider an e-commerce app, you added a product in cart and you want your orders screen (cart screen) or any screen where changes need to me made to update it's state without calling setState() in that screen's widget hierarchy.

"The rule of thumb is: Do whatever is less awkward."

Flutter SM Options...

- Provider
- InheritedWidget
- SetState
- Redux
- Fish-Redux
- BLoC/Rx
- MobX
- Riverpod
- GetX



Provider

dash-overflow.net

Google recommended. Simple & Beginner Friendly.

Bloc

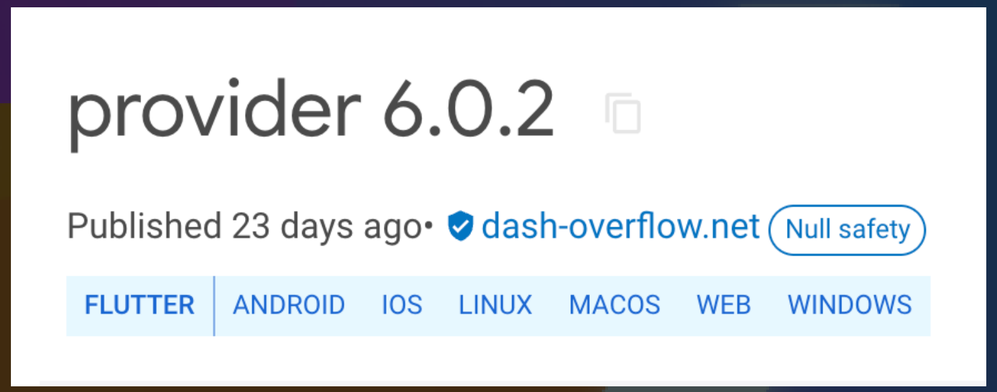
Google

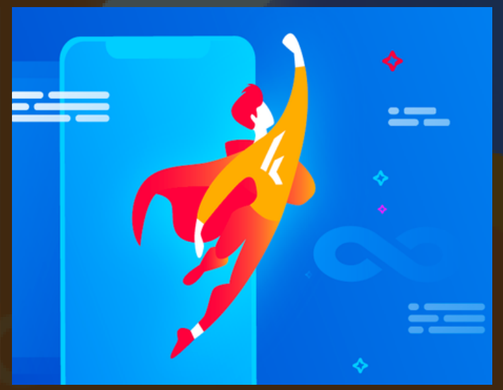
Promotes code reusability.
Business logic & presentation separation,

GetX

getx.site

Extra-light and powerful solution. High-performance state management





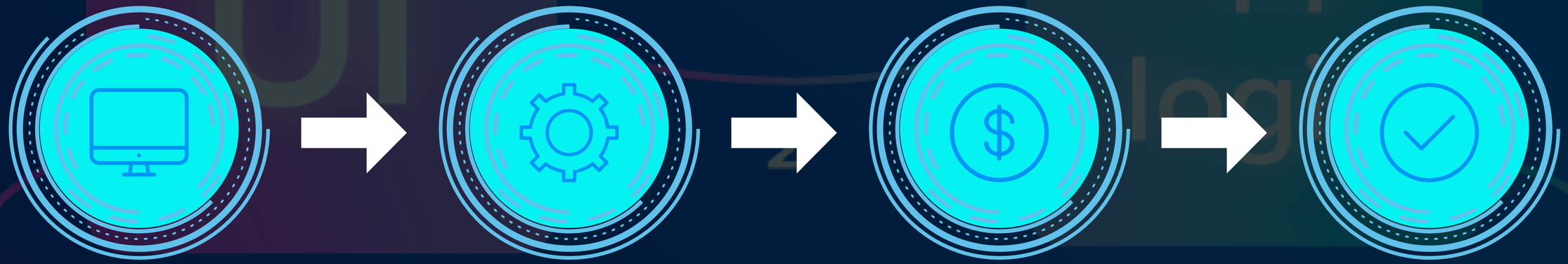
The Provider Package

Provider is the state management solution that Google recommends to beginners.

Popularity
100%

SM with Provider

Provider package works on principles of inherited widget, the process goes like this...



Step 01

A class with Change Notifier

Step 02

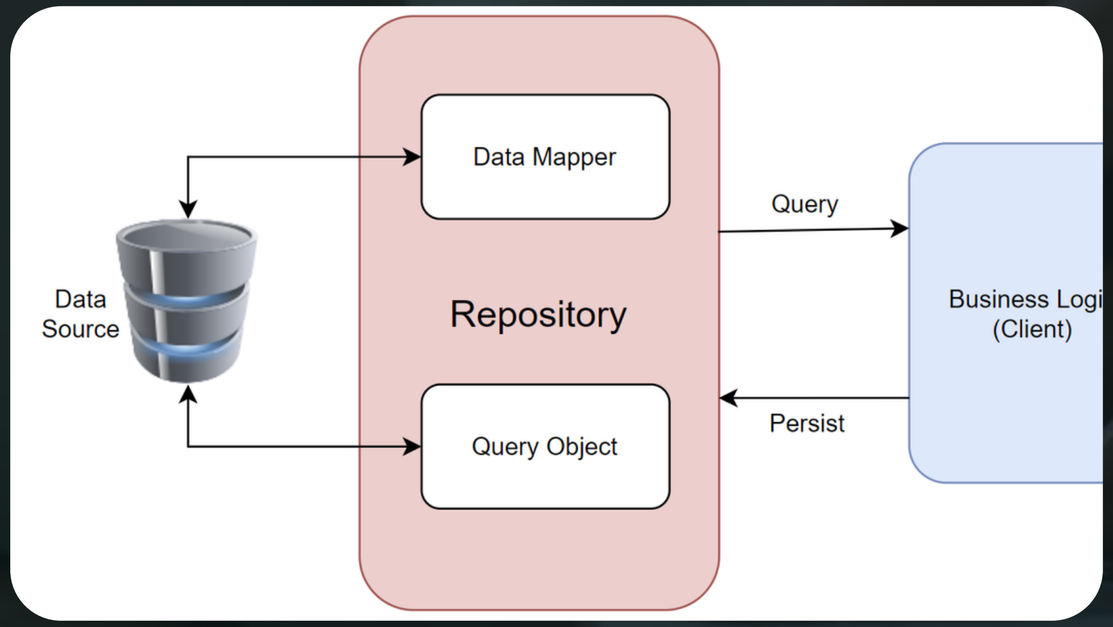
Change Notifier is used to call notifyListeners() method.

Step 03

notifyListeners() updates the widgets which listen to that class.

Step 04

The consumer widget gets rebuilt with new state.



Repository Pattern

A façade before complexity

Repository pattern separates the data access logic and maps it to the business entities in the business logic. Communication between the data access logic and the business logic is done through interfaces.

Some Best Practices

Naming

File names should be in snake_case
Function and field names in camelCase
Class names in PascalCase

Components

Always prefer class components over functional components,
Use **const** keyword most often.

Routing

Use named routes to avoid confusions while adding navigations.

Use of Const

Use of 'const' keyword is essential to avoid unnecessary widget builds.

Some Useful Takeaways

PNGTree

The best place to get free image assets.

01

02

Lottie Anims

Lightweight animations for Flutter projects.

Flaticon

Marketplace to get royalty free icons for your projects.

03

04

VS Code Plugins

Flutter vs code plugins help you in whole development process.

For help, Mentorship & Tech Discussion

Want To Get Connected?

 @rashidwassan

rashidwassan.tech

 @rashidwassan