

Flutter Packages & Plugins

Key Points	Notes
	<p>1. What is Packages and Plugins?</p> <ul style="list-style-type: none">a. Flutter supports shared packages and plugins developed using Dart and Flutter. They are contributed by other developers to ease Flutter development. In other words, you do not have to develop everything from scratch.b. Flutter packages support many use case from styling, sensors interaction, networking, custom navigation and many more. The number of contributed packages and plugins keep increasing every day.c. There is a distinct difference between packages and plugins:<ul style="list-style-type: none">i. Packages – Libraries are created purely using Dart and they are similar to Java Packages. They have better performance in Flutter.ii. Plugins – An interface to interactive with native code such as Kotlin (Android) and Swift (iOS). They are convenience to use but require a bridge to interact with Flutter. Thus, performance might reduce on the interaction. <p>2. Using Packages</p> <ul style="list-style-type: none">a. Most of the contributed packages are published to pub.dev web page. A single web page for all your packages and plugins that you can use with your Flutter development.b. When searching for packages and plugins to use for your Flutter, search the one in the Flutter Favourites page or the one with Flutter Favourite label. Packages and Plugins with Flutter Favourite label are certified reliable and having highest level of quality.
Summary	

Flutter Packages & Plugins

Key Points	Notes
	<p>c. If you decided to use one of the listed packages or plugins, you can refer to the instructions on:</p> <ol style="list-style-type: none">Readme – Description of the package and pluginChangelog – Updates made to the package and plugin.Example – Code example on how to implement or use the package or pluginInstalling – Steps needed to install the package and plugin in your Flutter project.Version – Latest version and improvements made to the package and plugin.Scores – Rating for the package and plugin based on pub points (Google Compliance Criteria). <p>3. Adding Package/Plugin Dependency to An App</p> <p>a. Once you have decided to use a package listed in the pub.dev, you need to add package/plugin dependency in your pubspec.yaml file of your project. This dependency information is available under Installing section of the package page. Example:</p> <pre>dependencies: flutter: sdk: flutter url_launcher: ^5.4.0</pre> <p>b. Once the dependency information already added and saved in pubspec.yaml file, you can start download the package or plugin into your Flutter project.</p> <ol style="list-style-type: none">Flutter CLI<ol style="list-style-type: none">flutter pub getMS VS Code<ol style="list-style-type: none">Click at pubspec.yaml file and click at download icon on the top right menu.Android Studio<ol style="list-style-type: none">Click at pubspec.yaml file and click at Pub get menu on the top right menu.
	<p>Summary</p>

Flutter Packages & Plugins

Key Points	Notes
	<p>c. Now you can import the package or plugin to be used in your <code>main.dart</code> file.</p> <pre>import 'package:flutter/material.dart'; import 'package:url_launcher/url_launcher.dart';</pre> <p>d. Stop and restart you app to make sure the package or plugin is running properly.</p> <p>e. Refer to the complete example https://flutter.dev/docs/development/packages-and-plugins/using-packages#url-example</p> <p>4. References</p> <p>a. Windmill, E. (2020). Flutter in Action (1st Ed.). USA: Manning Publications.</p> <p>b. Flutter Official Documentation. Retrieved on 1 December 2020 from https://flutter.dev/docs</p> <p>c. Flutter Official Documentation: Using Packages. Retrieved on 29 December 2020 from https://flutter.dev/docs/development/packages-and-plugins/using-packages</p>
Summary	