Plugins in Flutter

What are Flutter Plugins?

They are used to streamline and speed up the development process and offers pre-built solutions for native features.

Plugins are packages in Flutter that provide access to device and platform specific functionalities. They act as bridge between the flutter code and the underlying platform code.

Plugins allows developers to use platform specific features such as gps, camera access and file storage without writing platform specific code.

Benefits:

1- Simplified Development:
   They save time by allowing reusing code for standard functionalities.
2- Cross-Platform Integration:
   Offers consistent API across both IOS and Android.
3- Community Support:
   Most Plugins are community driven meaning that they are well tested and maintained.

Commonly Used Plugins:

1- http:
   It is for making http requests.
2- shared_preferences:
   for storing simple data persistently.
3- firebase_core:
   to integrate firebase into your app.
4- SQLite:
   For using SQLite in your app.
5- google_maps_flutter:
   To integrate google maps.

Creating Your Own Plugins:

Step no 1:

Create a new Plugin using Flutter CLI

Syntax (*flutter create --template=plugin --platforms=android,ios,web,windows,linux,macos <plugin_name>*)

Step no 2:

Write the Platform specific code by navigating to IOS and Android folders in <plugin_name> directory.

Step no 2:

Implement channels to communicate between dart and native code.

Step no 4:

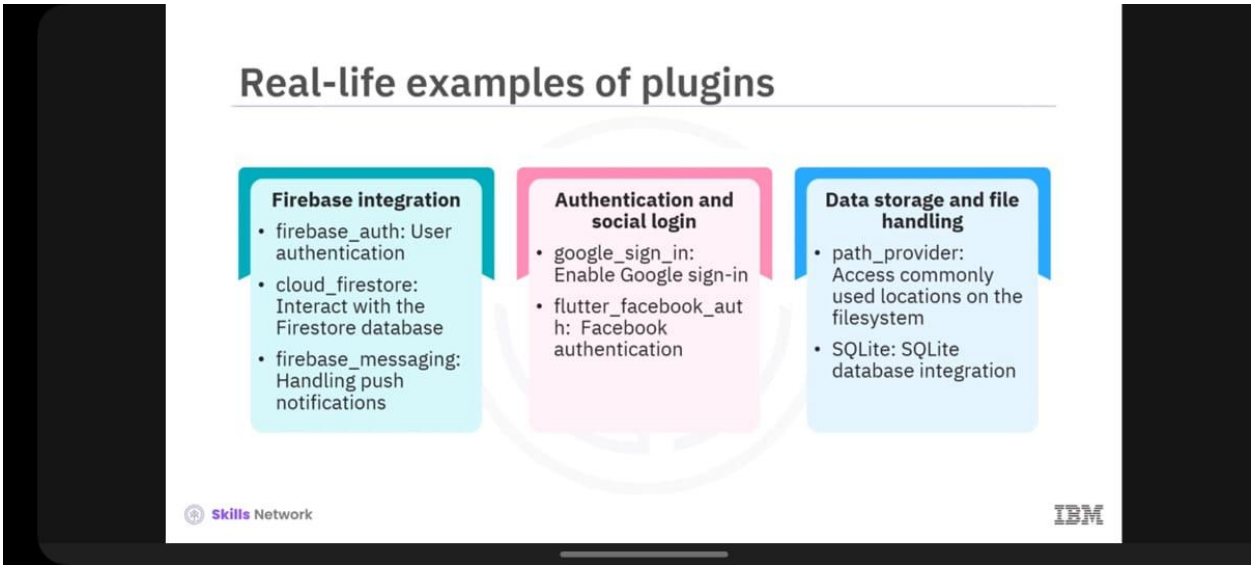
After testing, publish your plugin to pub.dev

Managing Plugin Compatibility:

1- **Managing Version Constraints-**
   Ensures Compatibility with your flutter and dart version.
   We need to specify constraints in pubspec.yaml file
   Example : dependencies: url_launcher >= 6.0.0 < 7.0.0

2- **Resolving Dependencies Conflict-**
   Dependency resolver finds a compatible version automatically.
   Or Resolve conflicts manually by specifying exact versions.

3- **Testing on Multiple Platforms-**
   Test Application on iOS and android devices and emulators.
   Ensure Proper working of your application.


Real-life examples of plugins
Firebase integration
• firebase_auth: User authentication
• cloud_firestore: Interact with the Firestore database
• firebase_messaging: Handling push notifications
Authentication and social login
• google_sign_in: Enable Google sign-in
• flutter_facebook_auth: Facebook authentication
Data storage and file handling
• path_provider: Access commonly used locations on the filesystem
• SQLite: SQLite database integration
Skills Network
IBM