All prerequisite packages to start Appium in flutter

System Requirement: node.js installed supported at least v16 or higher.

Install WebDriverIO

What is Appium Flutter Finder?

Appium Flutter Finder is a utility package designed to enable Appium to interact with UI elements in Flutter applications during automation testing. It works alongside the Appium Flutter Driver, which is specifically built to automate Flutter apps. The Appium Flutter Finder helps Appium identify and interact with Flutter’s unique UI components, which are different from those in native Android or iOS apps. Essentially, the Appium Flutter Finder acts as a helper tool to the Flutter Driver, making it easier to locate and manipulate Flutter-specific UI elements during test automation.

Command to install Appium Flutter Finder: (npm install appium-flutter-finder –save).

Features of Appium Flutter Finder:

Flutter-Specific Element Locators

1. **Value Key**: Allows you to locate Flutter widgets by their valueKey (a unique key assigned to a widget).
2. Example: FlutterBy.valueKey('widget\_key')
3. **Text**: Allows you to find a widget by its displayed text content.
4. Example: FlutterBy.text('Button Text')
5. **Type**: Locate widgets by their type (e.g., buttons, text fields).
6. Example: FlutterBy.type('Button')
7. **Tooltip**: Find widgets by their tooltip text.
8. Example: FlutterBy.tooltip('tooltip\_text')
9. **Custom Finder**: Supports using a custom widget locator based on various attributes.

**Support for Flutter Widget Hierarchy**

* The **Flutter Finder** enables finding UI elements based on the **widget hierarchy**, allowing you to navigate through parent-child relationships in the app's UI structure.
* You can search for elements by their position in the widget tree, enabling you to interact with elements in complex Flutter UIs.

**Integration with Appium Flutter Driver**

* The **Flutter Finder** works seamlessly with the **Appium Flutter Driver**, allowing Appium to interact with Flutter-specific elements through the Flutter Driver’s API.
* It leverages the Flutter Driver’s underlying capabilities while adding functionality for Appium’s more extensive cross-platform automation framework.

**Cross-Platform Support**

* Like Appium, the **Appium Flutter Finder** supports both **iOS** and **Android** Flutter applications, making it possible to automate tests across multiple platforms using a consistent API.

**Support for Multiple Locator Strategies**

* **Flutter Finder** provides multiple strategies to locate elements, including valueKey, text, and type, as well as more advanced approaches like FlutterBy.widget() for more granular control.

**Efficient Widget Interaction**

* It provides optimized methods for **efficient interaction** with widgets in Flutter apps, enabling Appium to perform actions like clicking, typing, and scrolling efficiently on Flutter UI elements.

**Helps with Flutter's Custom Rendering**

* Since Flutter apps use a custom rendering system, **Appium Flutter Finder** helps Appium locate UI elements that would be challenging to target using traditional methods (e.g., by ID, class name) in native apps.

**Full Access to Flutter Widgets**

* It allows testers to interact with all types of **Flutter widgets**, including custom widgets, ensuring comprehensive test coverage for your Flutter app’s user interface.

**Customizable Locator Strategies**

* Developers can create custom locator strategies by using specific attributes or widget properties, allowing more flexibility and control in locating complex Flutter widgets.

**Supports Integration with Other Appium Plugins**

* As part of the Appium ecosystem, **Flutter Finder** can be integrated with other Appium plugins for additional features like visual validation, screenshot comparison, and more.

Webdriverio command: npm install webdriverio –save

Install Specific Version: npm install [webdriverio@^9.7.1](mailto:webdriverio@%5e9.7.1)

"webdriverio": "^9.7.2"

"start": "node test.js" onece your have add this run this command (npm start)

"ios": "export APPIUM\_OS=ios && npm start",

"android": "export APPIUM\_OS=android && npm start"

After you have successfully added the above file run the command:

1). Npm run ios

2). Npm run android

appium --log-level debug (another method to run appium)

Set environment variable

set android\_and\_ios=android

verify the set environment veriable

command: echo $env:APPIUM\_OS