**Robot Framework Summary**

**Overview Robot Framework**

**Its main use is to enable writing acceptance tests with a high level of abstraction for software products.**

The framework requires less technical skill than programming language-based frameworks, and so can be used by team members who have very little programming experience. For example, an agile team product owner could express acceptance tests using the framework, without having to know the details of how the product is implemented.

On the other hand, because of its keyword-based design, those with a high degree of technical skill can write keywords in their language of choice to test low level functions, and even do unit and integration testing if so desired

**Definition robot framework for automation testing**

Robot Framework is an open-source test automation framework, exclusively designed for acceptance testing and acceptance test-driven development.

The environment it uses is built with a number of libraries and tools. If comparing Robot Framework with other test automation tools, then the best benefit with Robot Framework for the users is that there is no need for using any sort of programming language for implementing and running tests.

Also, the Robot Framework itself has a range of libraries to support a wide range of applications. There are even more benefits that prove Robot Framework a good test automation tool for the users. “Are there any openings for testing using this tool?” Many of the teams feel enjoyed with Robot Framework as it simplifies and helps in the automation.

Being an open-source tool with good number of features, the tool has become a great preference to many leading organisations. With its easy syntax feature, human-readable keywords, and ability to extend its capabilities with libraries, etc. the tool is widely preferred across the globe in different contexts and domains. Hence, there are ample opening for testing with this tool.

**When to use robot framework for automation testing**

Because of its flexibility, robot framework can be used to test desktop applications, web applications, mobile applications, and RESTful and SOAP-based services. Because it provides an abstraction layer on top of the physical implementation of the system under test, it is possible to write tests that work cross-platform (ie: the same test case could be used to test both an android and iOS app, or for testing a web app that runs on chrome, Firefox and safari).

Although robot framework is often used to test browser-based applications, it has been used to test databases, mobile devices, calculators, and many other things.

Robot Framework is used extensively as a test tool for acceptance test-driven development (ATDD) in a variety of applications, and increasingly also for Robotic Process Automation.

**Extensibility**

robot framework is that it is highly extensible. Many of the features mentioned above are implemented as libraries. For example, you can plug in a library to use selenium to drive a browser. You can plug in a database library to directly access databases. There are libraries to support testing desktop apps, services, and many more.

**Example**

To illustrate how abstract a robot test case can be, a BDD-style testcase might look -- literally -- like this:

“Given I am on the application login screen

When I enter a valid username and password

And I press the "login button"

Then I should be logged in

And my personal dashboard should be displayed”

Note: this isn't the only way to write a test. Robot is very flexible, allowing you to choose between BDD (Behavior Driven Development) style, a traditional procedural style, or in a data-driven style.

The framework provides the infrastructure for converting those statements to actionable items. Ultimately, they get boiled down to a function call -- either from an external library, or one provided by the development team. Those functions typically are written in python, but can be written in Java, a .NET language, or other languages through an interface, depending on how you actually run the test.

**Reporting**

able to run tests, robot framework provides reports and logs that can be used to visualize the state of the product. Logs provide details of every single keyword and function call, reports show the status of complete suites and test cases, and can provide summary information based on arbitrary tags

**Limitation robot framework for automation testing**

* Robot framework does not support parallel execution
* Hard to customize html report
* Robot framework is hard to maintain
* Some error are difficult to debug
* Robot framework has strict indentation rules

**Benefit robot framework**

* It is open-source framework.
* It follows keyword driven, behaviour driven and data driven approach for writing test script.
* Robot Framework has very easy syntax and uses human readable keywords.
* It capabilities can be extended libraries implemented in Java and python.
* Robot framework is extensively used for RPA, Database, API automation & Web automation.
* This is the only framework which comes with all industry standard framework features with very minimal coding.