**What is soapUI?**

* soapUI is a tool which can be used for both functional and non-functional testing.
* The first release of soapUI (v1.0) was in October 2005. Ole Lensmer, while working in a project related to SOA, felt the need for a testing tool to support agile development. Therefore, he started to develop soapUI in his spare time. Eventually, the project was open sourced and the community grew. Ever since, a number of versions have been released with various new features and enhancements
* The originator of soapUI, Ole Lensmer was managing the project releases through a company called Eviware for the past few years. In July 2011, Eviware was acquired by SmartBear Software (http://smartbear.com/) and now soapUI is part of SmartBear Software.
* soapUI is also distributed as a non-free commercial version known as soapUI Pro, which basically provides users with custom utilities and enhanced production level testing capabilities

**Capabilities of soapUI**

soapUI is use to provide users with a simple and user-friendly utility which can be used to create and run functional as well as non-functional tests through a single test environment.

**Some of the important features of soapUI:**

• **Complete coverage of functional aspects of web services and web applications**: soapUI supports most of the standards used in web applications, such as message transmission through HTTP, HTTPS transport

as well as JMS. **It also supports testing SOAP and RESTful web services**

• **Service mocking**: Using soapUI mock services, you can simulate the web services before they are actually implemented. This gives you the ability to test the web service consumer applications without waiting until the web service providers are implemented.

• **Scripting**: Either using Groovy or JavaScript, soapUI allows you to do various pre- or post-processing test configurations such as dynamic mock responses, initialize or cleanup tests, dynamic mock operation dispatching, and so on.

• **Functional testing** : soapUI lets you do functional verifications against web services, web applications, and JDBC data sources. You can validate responses of your tests using various in-built and custom assertions. It also allows you to add conditional test steps to control the test execution flow.

• **Performance testing**: With just a few clicks, you can generate performance and load tests quickly using soapUI.

• **Test automation**: soapUI can be integrated into automated test frameworks such as JUnit, and the tests can also be launched through Apache Maven. It can also be integrated into continuous integration tools such as Hudson or Bamboo.