



RAFT - REUSABLE AUTOMATION FRAMEWORK FOR TESTING

RAFT – Re Usable Automation Framework for Testing

In this digital era, there is good opportunity to expedite deliver of applications to market with improved quality. Test automation plays a major role in accomplishing the above said objectives. As more projects are moving towards agile, test automation becomes indispensable to reduce the testing hours without impacting the quality. Test automation is also crucial for product development as its requires frequent repetitive testing, Novature tech realizes the need to have a strong framework to deliver test automation solutions.

With the expertise on test automation, Novature Tech developed home grown test automation framework. We call it as RAFT. Key elements of the framework are

- Re-usability
- Flexibility
- Robustness
- Maintainability
- Reliability
- Portability

RAFT

RAFT is Re-usable Automation Framework which automates and enhances your testing process by decreasing the cost and man-hours for testing your applications. Our RAFT is a ready to use automation framework which enables you to test the Web, Web services and Mobile applications

- ✓ The key dimensions of our RAFT as clearly depicted in its logo are
- ✓ Highly Re Usable .
- ✓ Ready to use and easy to customize.
- ✓ Easy to maintain thereby reduces the maintenance cost.

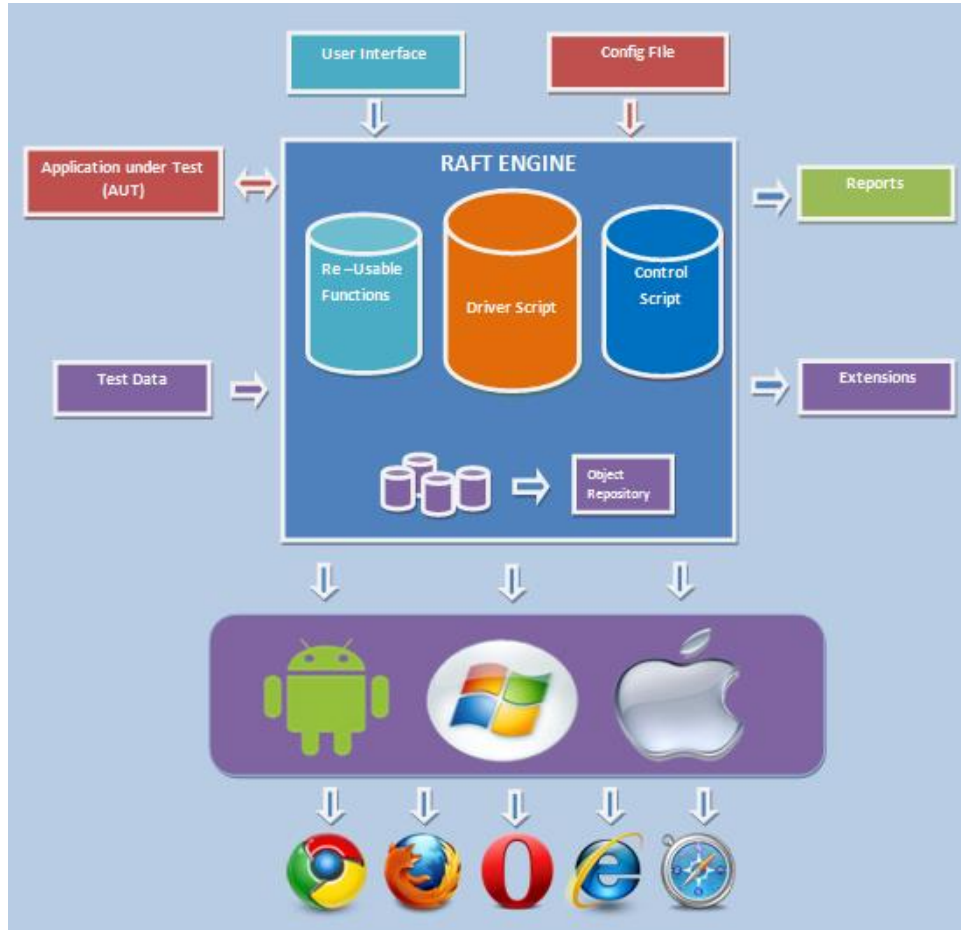
Key Benefits:

- ✓ Ready to use that reduces the time to Develop Automation Scripts
- ✓ Enables easy migration of Scripts from Licensed tool (Such as QTP, RFT, etc) to Open Source
- ✓ Reduce greatly the Cost of Future Testing with improved Quality
- ✓ Delivers higher ROI through automation
- ✓ Higher productivity
- ✓ Lower maintenance cost

RAFT Specifications:

- ✓ Supports Data Driven Testing (DDT) and reusable functions
- ✓ Supports Behavior Data Driven Testing (BDD)
- ✓ Advanced libraries / frameworks is built on open source tools
- ✓ Allows to run failed test cases
- ✓ Allows selective test script execution
- ✓ Test script schedulers.
- ✓ Ease of integration with open source tools like Jenkins, Maven and Cucumber
- ✓ Contains flexible test suite / test case configuration, command line execution and out-of-the-box test case execution support
- ✓ Robust and extensible framework to support test automation for different types of Applications
- ✓ Enhance your methodology for Future Testing with improved Quality
- ✓ Enables users to perform functional, acceptance and compatibility testing for most applications
- ✓ Demands less maintenance of test scripts
- ✓ Increases test automation coverage
- ✓ Offers user defined exception logging along with custom exceptions
- ✓ Provides detailed reporting with screenshots, passed, failed, skipped, and error logs
- ✓ Auto Email reports
- ✓ Customizable reports.
- ✓ Mobile compatible reports

RAFT Architecture



RAFT Components

1. Application Under Test

Our RAFT supports all types of applications such as Web, Mobile and Web Services. it also supports different browsers such as Chrome, IE, Mozilla, Safari and Opera.

2. RAFT User Interface

Our Re- Usable Automation Framework provides a structured user friendly UI which enables even a non-technical person to execute the automation and capture the results. It also acts as a wrapper to the engine to configure the Test Data and Sequence of execution. User Interface is the new feature being added in the latest version of RAFT. It can be either customized or newly created based on our Client's requirements. The different components of User Interface and the purpose of those components are detailed below.

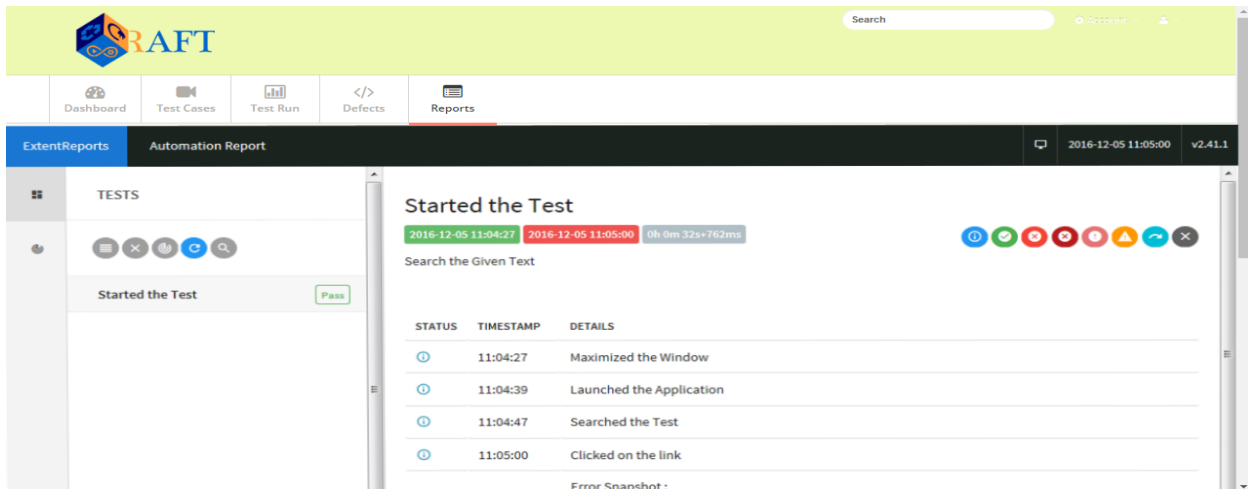
Dashboard – This displays the snapshot of all recent activities such as Test Cases, Test Results and Test Reports

Test Run – This module or tab is use to plan the Test Run and Capture Test Results. Test Automation Scenarios can be pulled from the Test Case Tab. It also enables selective execution of certain Test Automation scenarios.

Test Case – Contains the list of Test Automation Suites available for execution

Reports – It has collection of all reports with details of Test Metrics such as the number of Passed, Executed, Failed, etc for different Test Runs. Screenshots can also be retrieved from reports for both passed and failed Test Scenarios.

The snapshot of report is shown below



- ∞ Dashboard
- ∞ Test Run
- ∞ Reports

- ∞ Test cases
- ∞ Defects

It can be modifiable as per your requirement.

3. RAFT Engine

This is the Heart Our RAFT framework which consists of entire logic. It is created in such a way to handle the three types of applications such windows , IOS and Android.

A better automation approach should satisfy the following factors.

- ✓ Reusability
- ✓ Flexibility
- ✓ Robust
- ✓ Maintainability
- ✓ Reliability
- ✓ Portability

We, at Novature designed our RAFT which fulfill the above factors and it contains the below components

Driver scripts These are the main scripts that invoke different controller scripts for running a particular test depending on the test run configuration.

Controller Scripts These scripts comprise of various utility and component function calls and logics to complete a test.

Re Usable Function Library This contains two types of library as follows

Utility Function Library This includes a set of generic functions that can be shared across various scripts and functions. For example, functions for DB Connections, Reporting and Logging etc

Component Function Library This includes set of functions that are specific to functionalities of the application under test. Controller scripts make use of these functions to execute various test scenarios related to the application being tested. For example, functions for login application by tester role.

Test data Test data pool contains the entire set of application specific data that might be required by the component functions to do one or more iterations of test run.

Object Property Manager (OPM) OPM provides the input for dynamic object creation. Properties and values of all the necessary objects are stored in OPM and is used for dynamic object creation.

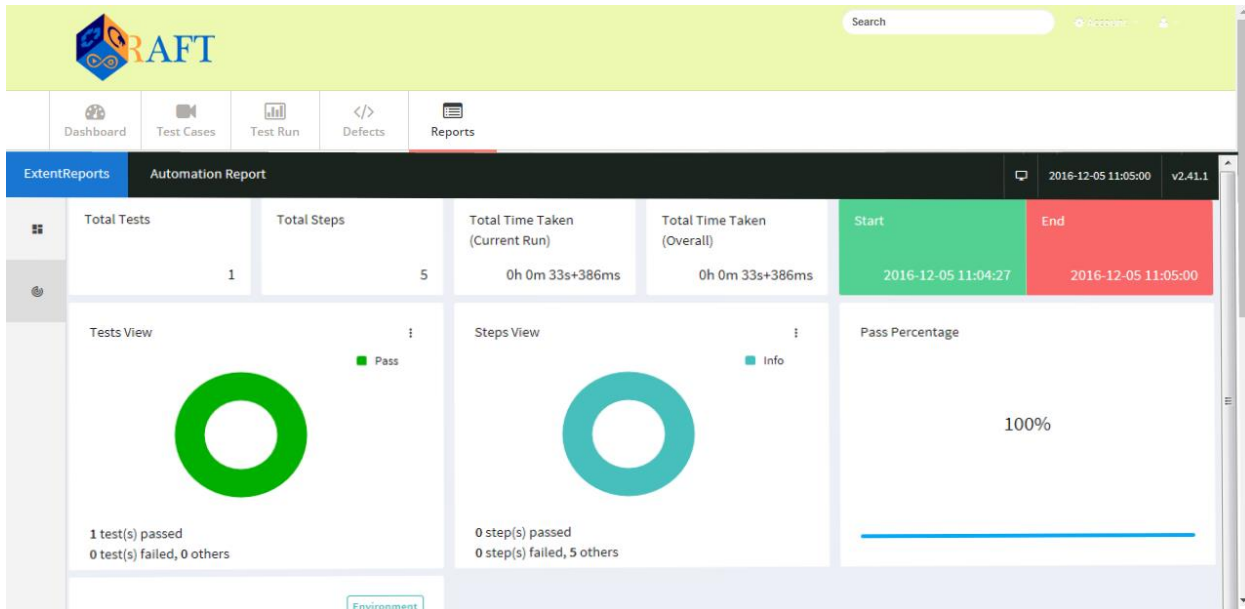
4. Test Configuration

The Framework has the test configuration which enables you to run multiple test cases/ scripts at a time and it used to skip particular number of test cases.

This Ready to use automation framework allows you to run the test case in different browsers and machines.

5. Reports

Our RAFT framework has customized test reports which allow you to find the number of passed and failed test runs and also you can view the cause of the failure.



One of the important key features of our framework is auto sending of the report to mail. So it will be easy for your clients to understand the test execution.

6. Extensions

RAFT enables integration with Jenkins, Maven and Cucumber. RAFT can be configured depending upon Client's requirements. So, RAFT supports Build and Integration, Continuous integration and BDD.

RAFT Implementation approach

Our implementation Approach consists of five phases. The objective of each phase, entry criteria, key activities performed and the deliverables are mentioned below. The Customization of Framework and Demo can be completed in 2 to 3 weeks depending upon the complexity of the applications and requirements. We would be able to give a Clear estimation of the entire Automation upon our completion of Demo.

Phase	Entry Criteria	Activities	Deliverables
Automation Requirement Study and Analysis	The primary objective of this phase is to assess the nature of the application, client requirements, client expectations and establishing the scope of automation.	<ul style="list-style-type: none"> • Identification of Test Cases for Automation • Analysis of Requirements • Conduct Feasibility Study • Ideal Candidates for Automation 	<ul style="list-style-type: none"> • Automation Traceability Matrix • Scope and Estimation • Customize RAFT

Automation Test Planning	The test cases identified for automation should be base lined	<ul style="list-style-type: none"> • Conduct Proof of Concept • Preparation of Automation Plan • Automate Couple of Scenarios for Demo 	<ul style="list-style-type: none"> • Demo of Automation Suite and framework • Automation Plan
Automation Test Design	Automation Plan reviewed and Approved	<ul style="list-style-type: none"> • Customize Framework • Enhance Scripting Standards – Branding, Naming Convention, Parameterization, 	<ul style="list-style-type: none"> • Customized Framework
Test Script Development	Customized Framework and Test Cases to be Automated	<ul style="list-style-type: none"> • Creating Reusable Business Functions and Enhance Common Functions • Converting Requirements into Scripts • Dry Run 	<ul style="list-style-type: none"> • Test Automation Suites • Dry Run Report
Test Execution and Reporting	Test Automation Suites	<ul style="list-style-type: none"> • Test Execution and Reporting • Preparation of Instruction Document 	<ul style="list-style-type: none"> • Test Execution Report • Instruction Document

Novature Tech

Novature Tech is an Independent Software Testing and Quality Assurance Services Company. Novature stands for “Innovation for Future”. As the name implies, Novature Tech strives to implement blend of proven and innovative approaches in the field of Software Testing to deliver best quality, highly efficient and cost effective solutions. It is a private limited company head quartered in Chennai, India with its partners spread across the globe. It provides holistic solutions for Software Testing. It ensures high quality and speed delivery to market. It’s key service offerings are Functional Testing, Regression Testing, Test Automation, Performance Testing, Test Process Consulting, Data Analytics – ETL Testing, Agile Testing, Non-Functional Testing, Security Testing, Security Audit, Mobile Testing and IOT Testing.

For more enquires:

E Mail us: sales@novaturetech.com

USA: + 1 732 -930-6846

INDIA: +91- 44-64600299