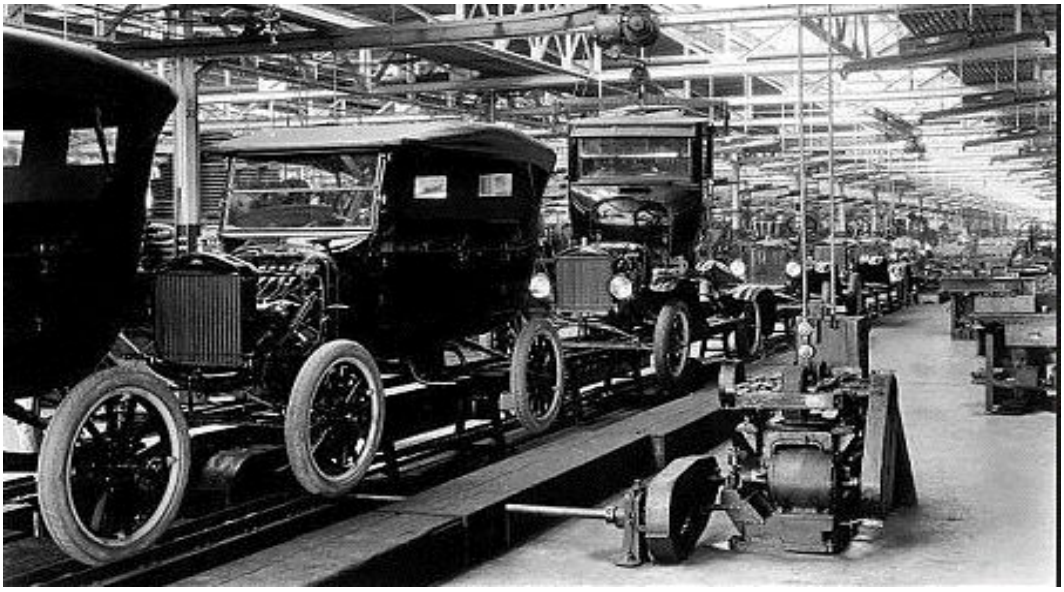
Chapter-1

# Introduction



"We live in a time when automation is ushering in a second industrial revolution" - Adlai Stevenson II

Welcome to the exciting journey of HP Unified Functional Testing (UFT) learning. HP UFT is market leading test automation tool.

In this chapter you will explore the following topics

## CHAPTER - UNDERSTANDING AUTOMATION

1. What is Automation
2. Benefits of Automation
3. How to realize automation – Automation Process
4. Scenarios when automation should be avoided
5. A Brief about HP Unified Functional Testing (UFT)
6. Why UFT is leader in software Automation – UFT Features
7. Trivia about UFT

## 1.0 What is Test Automation?

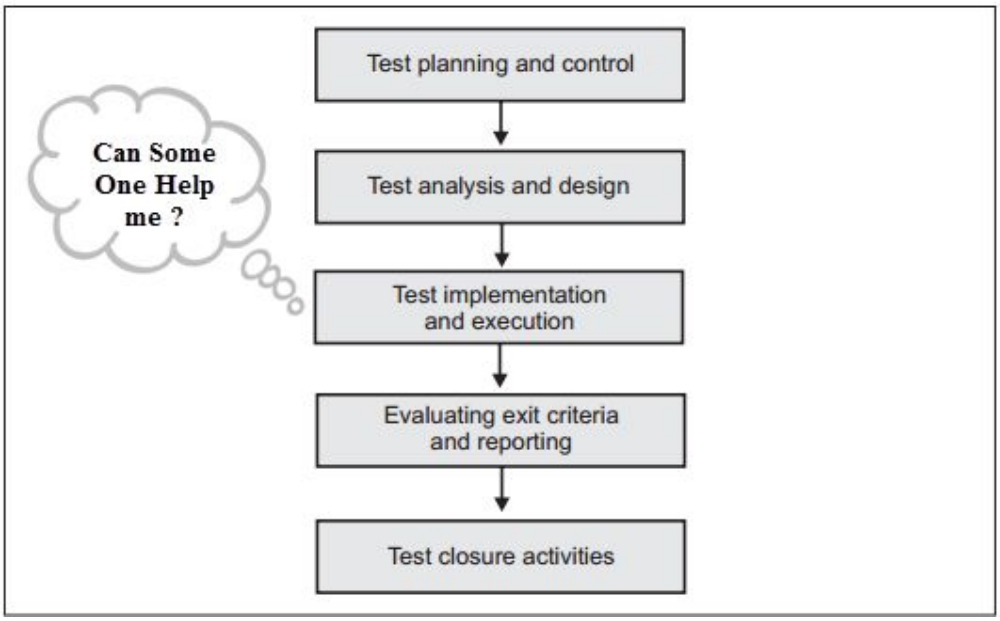
**Automation** is automatically controlled operation of an apparatus, process, or system by mechanical or electronic devices that take the place of human labour.

- Merriam-Webster Dictionary

Humans test the system with various data inputs and take decisions during test process. Test automation is the process of reducing, and when possible, removing human interactions from an existing manual testing process.

2.0 Let's Explore Manual Testing

To confuse you, testing have many definitions from various sources. But overall, testing verifies the product quality by conforming business requirements and product standards. Generally, testers do the following activities




Suppose, we need to verify booking of a rail ticket on website. Booking of one ticket takes 3 minutes with following steps

Navigate to Site | Login details | Search Availability | Customer details | Pay the amount | Logout

Now suppose, we need to check ticket availability for 100 trains and pay by 4 methods (debit, credit, PayPal and net banking). The total time will increase to 3(min) * 100 (trains) * 4(pay) = 1200 minute (i.e. 20 Hour or 2.5 days).

Most of times, manual test execution takes lot of time. UFT can mimic the user action and repeat them for any number of times. UFT scripts can be modify with VBS programing to enhance and meet the user requirements.

## 2.0 Benefit of Automated Testing

Test automation minimizes the manual intervention during test execution. Manual testing may have following pitfall

1. Human errors
2. No reusability of actions. Every action repeats same user efforts.
3. Slow, Human need more think time to provide input and take decisions.
4. Manual Testing is a costly affair and should mix with suitable automation strategy to minimize the testing cost.
5. Manual test become boring to perform monotonous procedures of testing. Automation comprises scope of innovation and can also take care the routine jobs.

We can automate the users actions e.g. click, filling forms etc. but the automation tasks can do various tasks including the following list

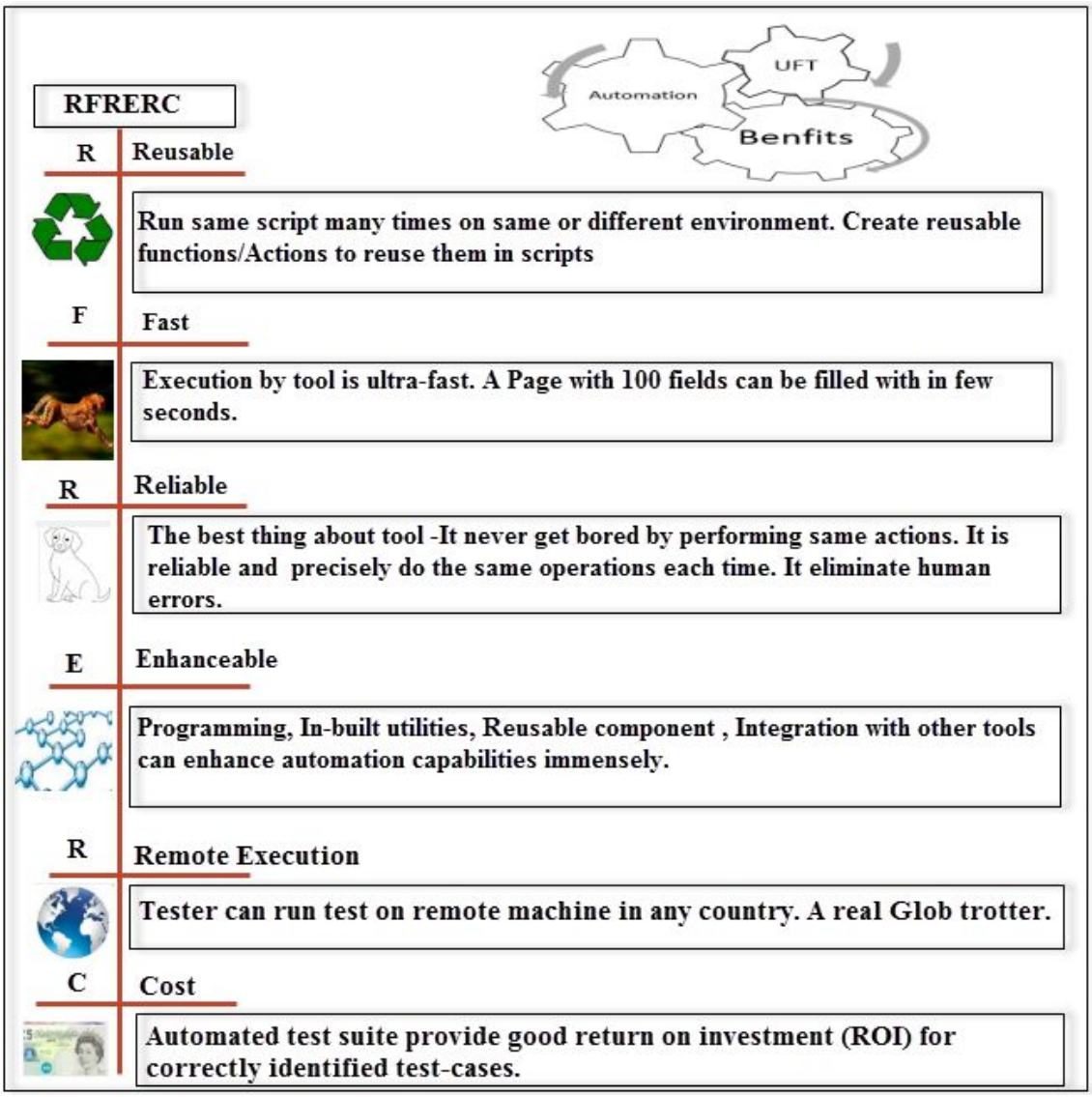
| S.No. | Type of Automation | S.No. | Type of Automation |
|---|---|---|---|
| 1 | Process Automation | 8 | Run-Book-Automation |
| 2 | **GUI Automation (UFT Stands Here)** | 9 | Data Center Automation |
| 3 | API Automation | 10 | Accessibility Test Automation |
| 4 | Database task Automation | 11 | Business Process Automation |
| 5 | Windows task automation | 12 | Performance Test Automation |
| 6 | UNIX task Automation | 13 | Support Automation |
| 7 | Reporting automation | 14 | Automated IT-Service Management |

There are some myths associated with UFT automation testing.

| GUI Automation | Myth | Myth-buster |
|---|---|---|
| **Tool is Not Omnipotent** | UFT will replace manual testing completely | UFT can assist to automate as much as possible, Importantly Regression and Sanity test suites but it will not replace manual testing. |
| **No Silver Bullet** | Buy tool, record & run test and be Happy! | UFT automation need proper implementation. |
| **Tool has long Learning curve** | Record and playback is good. Exploring too much is difficult for testers new to technical testing | UFT provides various help material, support. The learning curve is not very long for UFT. |
| **It needs maintenance** | Just create tests and it will run whenever you want, even after years after creating tests. | UFT scripts, like any automation, need maintenance. Automation suite need to maintain along with application changes. |

| | | |
|---|---|---|
| **It is not just buzz-word** | Automation fails so often that we see no real value in it. | Automation without proper thinking is bound to fail. So Automation should be treated as any software development life cycle |

These are the major benefits from software automation

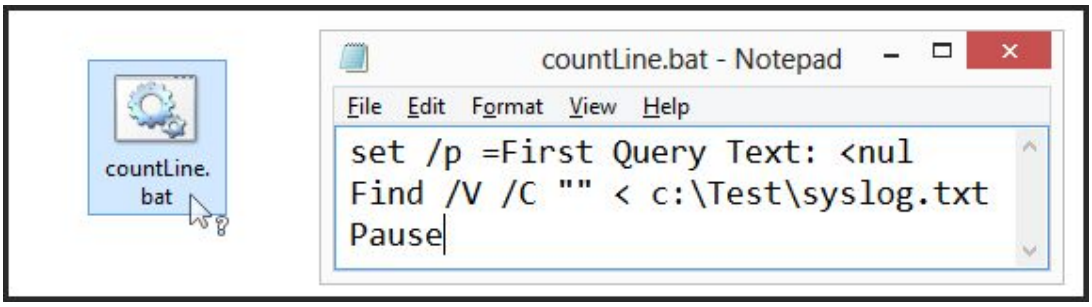



## Exercise 1.0 (of 3 STEPS) – Understand the Automation benefit

This exercise will create a bat command program (not VBScript) to count the lines in logfile.

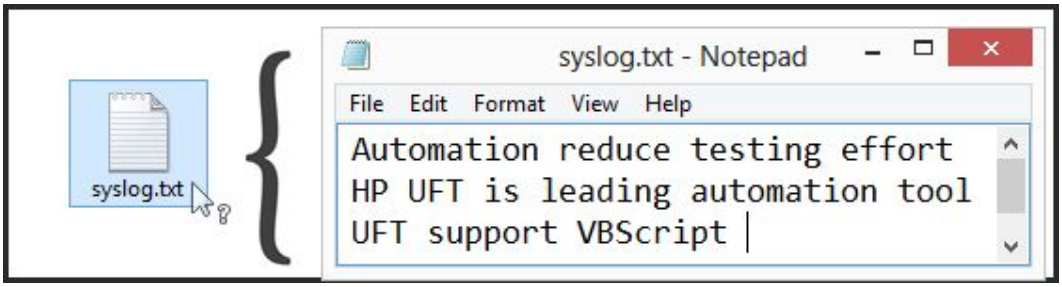
STEP1: Create a program to read file

*Navigate to C: drive | Create folder "Test" if already not exist | Create a file countLine.bat | Select Edit by right click on file | write following file | Save | Close*




STEP2: Create a file that need to read

*Navigate to "C:\Test" | Create a text file and rename to "syslog.txt" | double click on "syslog.txt" | Write following lines |Save | Close*




STEP3: Run program to automatically count the lines in file

*Navigate to "C:\Test" | either double click on "countLine.bat" – or – Right click on "countLine.bat" and run as administrator | verify the command prompt window| Press Enter*

## Concept of Exercise 1.0 – Understand the Automation benefit

In STEP1, we have create a countLine.bat file. This file contains the batch command to read a file and count the lines in them. We can change location of file and run same program on any other file to count the number of lines. Programmer create program in various language and the file extension change based on the programming e.g. .java for Java , .php for PHP, .htm for HTML and .vbs for VBScript programming.

In STEP2, we have created a syslog.txt with three lines in them. In STEP3, we execute program to count the lines. Suppose the log has been generated with millions of line. Will it be appropriate to count the lines? Or log keep generating for every 30 second poll, will it be appropriate to check the lines manually?

Automation is a way to reduce manual effort. Manual effort require to provide input , take decisions, verify data, provide user action to application, manage certain condition of system to run specific application. Automation can be achieved by any programming language and tools.

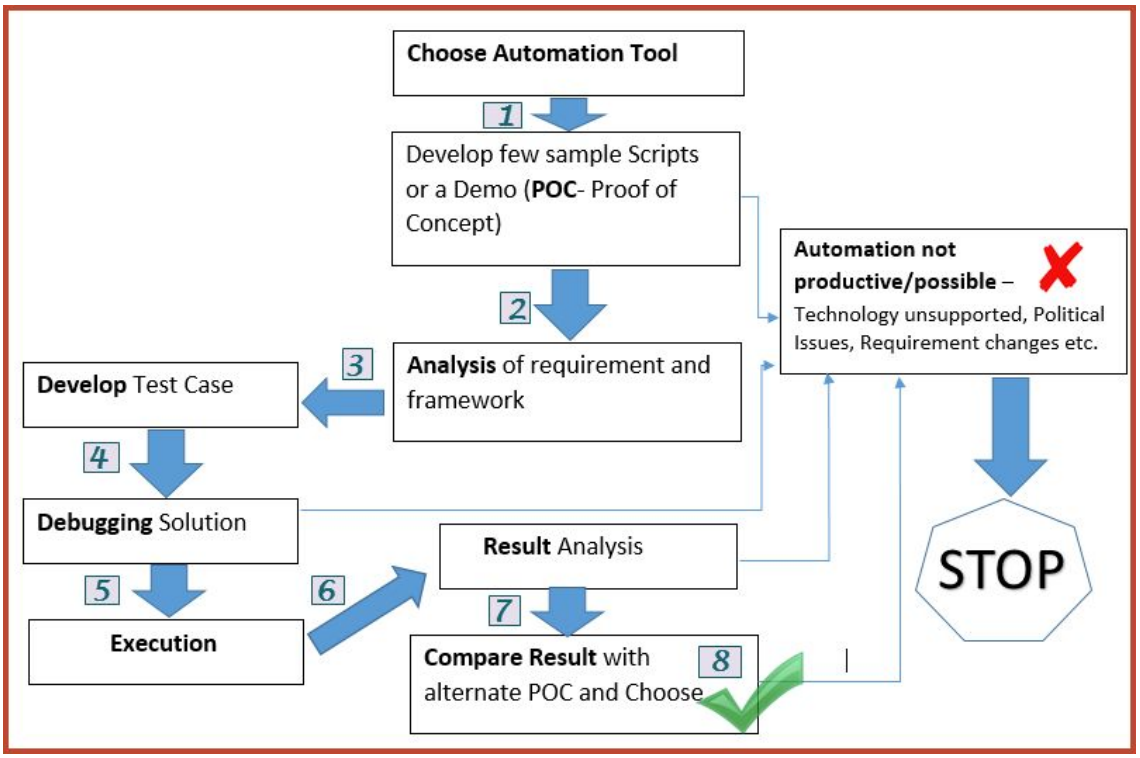
## 3.0 The Automated Testing Process

Let's meet Aadhya and Aric. Aadhya is an aspirant of automation testing and Aric is an **Automation** mentor who guides people to thrive in automation career. Let's see what they are discussing

**Aadhya** **Aric**

***Aadhya****: Hello Aric*
***Aric*** *: Hi Aadhya ☺*
***Aadhya****: I need to start automation. I do not know how to start? Which tool to start with? How I will convince people that it will reap benefit? Can you please guide me?*
***Aric****: Why Not!*

*It is important to understand the various phases involved in the automated testing process in order to develop and effective framework and test cases.*
*Let me explain you. Please have a look on the phases in picture below.*

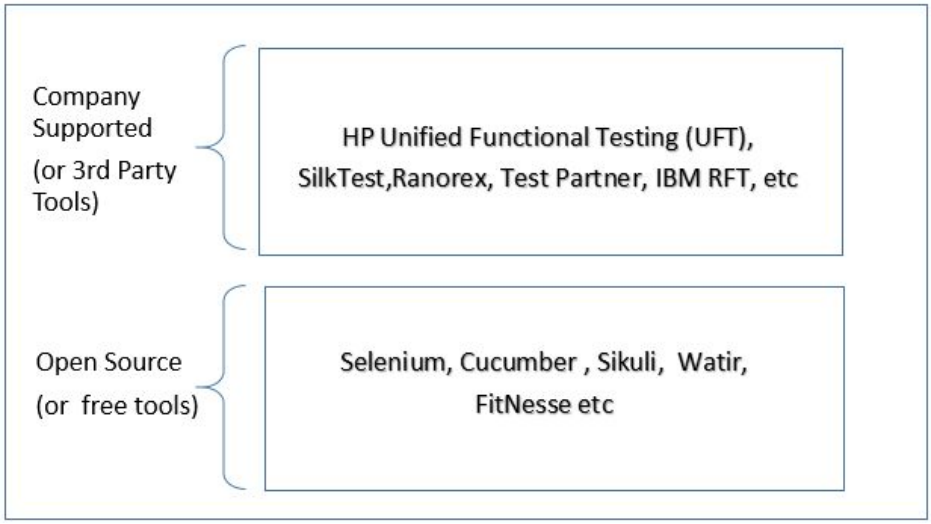


***Aadhya**: oh I see! Can you please tell me more about these phases Aric?*

***Aric**: Why not!*

**Phase 1 - Selecting the "best fit" Automation Tool:** Before starting to automate testing in any application it is important to pick the best fit tool for the majority of your application. The choice should be based on variety of factors such as cost, ease of use, application support capabilities and product support.

There are various GUI Automation tools available [Ref 1.0].




Tool evaluation is out of scope of this book and we will focus on HP UFT in remaining part of this book.

**Phase 2 - Proof of Concept (POC):** People have more confidence when they can verify the sample solution. This activity involves creating a few sample scripts that validate a few important business workflows in one or two of your most important applications. This helps identify any major issues that might be encountered during future test case development. A proof of concept should also be used to select the best automation test tool for your applications.

**Phase 3- Design the Test Architecture**

**3A - Requirements Analysis:** This activity involves analysing the requirements of an application, studying the existing manual test cases and then defining the scope of the test automation project.

**3B - Project Estimates:** Once the scope of automation is defined, estimates can then be formulated based on various factors like the number of test cases to be automated, their level of complexity, what re-usable components need to be developed, staffing requirements etc.

**3C - Framework Design:** This activity involves creating shared object repository(s), any reusable components, writing a best practice guideline document and completing any supplementary activities to prepare a base of supporting components that will be utilized to develop automation test scripts.

**Phase 4 - Test Script (Cases) development:** Test cases are created by calling the reusable components and adding appropriate validations to each workflow specific script.

**Phase 5 - Debugging:** Completed test cases are debugged to make sure they work as designed. Make sure you force the code through all error handling paths during phase.

**Phase 6 - Execution:** In this phase test cases are finally put to work through regression testing and validating the application under test.

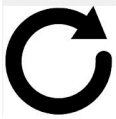
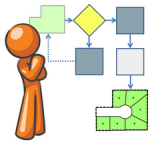
**Phase 7 - Result Analysis:** Process followed by the results created by each executed test after execution.

**Phase 8 – Choose and Maintenance:** This phase involves updating scripts to fix any code related to issues found during execution. These issues may include UI or mark-up changes, changes in flow or functionality or any other inevitable changes made to the application during new builds or releases. A well designed framework and set of tests ensures that maintenance costs are kept to a minimum

***Aadhya***: That is very insightful. Does all the phases are equally important?
***Aric***: To realize the robust automation test suite, these all steps are equally important.
***Aadhya***: But I do not understand small bits of these phases. Can you please help me out?
***Aric***: I hope that completing this book will resolve all these questions, so hold the breath for now ☺ .

## When Should Test Automation Be Used?

Aadhya: So, Should we automate everything?
Aric: No, Definitely not. You need to analyse various factors before starting any automation otherwise you may end up updating your CV.
Aadhya: oh! Please tell me those factor quickly Aric or send me good CV template.
Aric: ☺ following are the important factors:

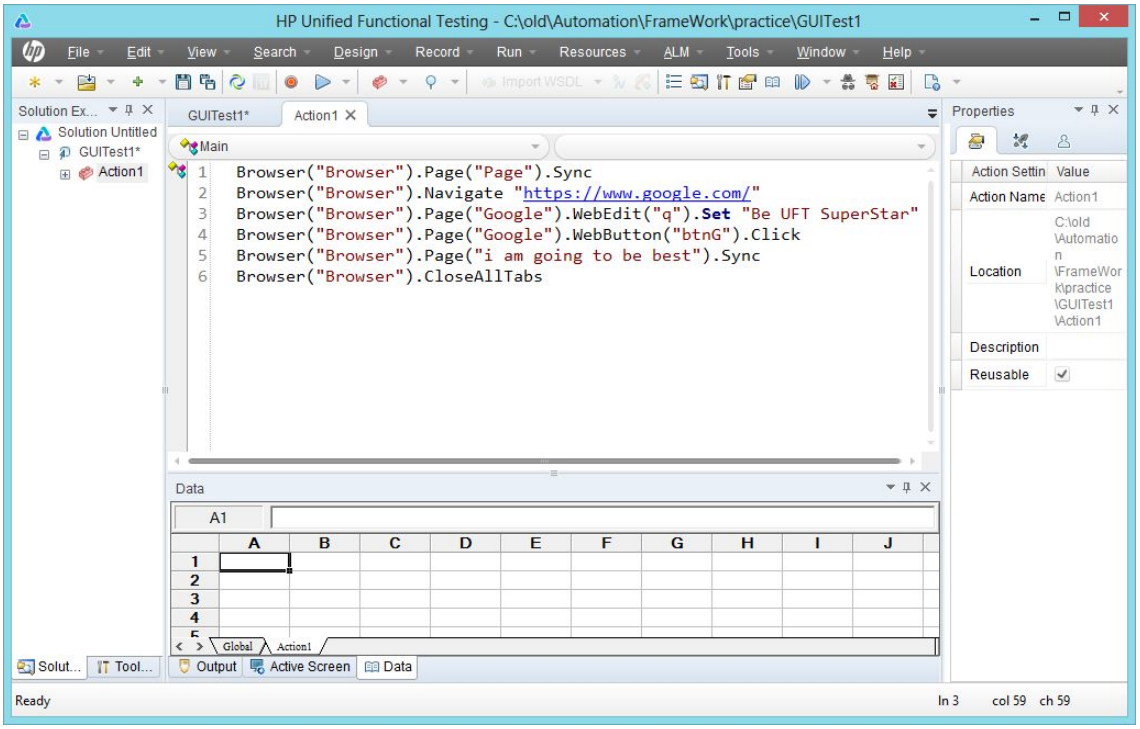
| DECSION FACTORS | Manual test a good candidate for automaton | When Should Test Automation Be Avoided? |
|---|---|---|
|  **Repetitive** | The test must be repeated often | Ad hoc testing where a subject matter expert randomly prowls through a variety of combinatorial workflows. **OR** One time testing or testing repeated only a few times. |
|  **workflow** | The test's workflow and its | Changes are very frequent |

| | | | |
|---|---|---|---|
| | | validation evolve and change slowly over time. | |
|  | **Validations** | The test validates a business process or workflow, rather than look and feel, colour, table layout, etc. | Testing where look and feel, colour, table layout, etc. are validated. |
|  | **Repetitive n-Number of same Steps** | The test is very repetitive and/or has a lot of steps, and it is important that those steps be performed<br>exactly the same each time, where manual tester fatigue must be avoided. | Testing which requires covering multiple functional areas such that the test travels through a small amount<br>of virtually all of the product's functionality. |
|  | **Regulatory Compliance** | The test produces results for a regulatory body that demands that those results be electronically recorded<br>and archived as formal evidence of compliance. | Regulatory body prohibit to do any electronic data recording. |
|  | **Record and Playback** | The test's pass/fail results are reasonably easy to determine and capture with the selected automation tool | Testing where pass/fail validation requires evaluating information from several different and unrelated systems and/or applications. |
|  | **Data Intensive Operation** | The test needs to drive a significant amount of data to the application | Low data intensive operation with more focus on Look and Feel |

**4.0 Love at First Sight - HP Unified Functional Testing (UFT)**

HP UFT is a Functional Test Automation tool. It is upgraded version of HP QuickTestPro (QTP v11.0). Out of the box, it supports a GUI* Record and Playback methodology where automation engineers can record and capture events with the application under test and then replay those GUI actions during the testing phase. UFT also provides API automation facilities along with GUI automation in single IDE.

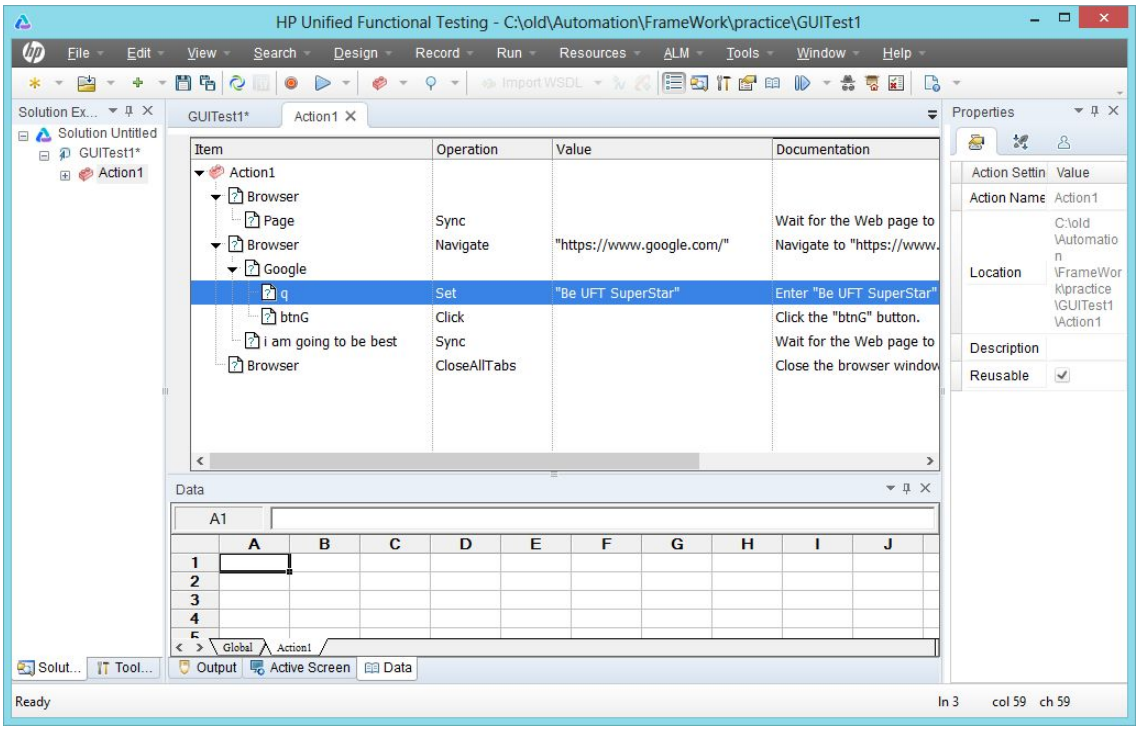
By default UFT recording can be view in Action1 tab.




UFT also provide Keyword view. The Keyword View displays the script in the form of keywords (each application object** is an icon) arranged in a tree-like format that is targeted at Subject Matter Experts (SMEs)/Business Analyst (BAs) with little or no programming background.

When tester start recording in UFT, It interacts with application. HP provides the add-ins for various technologies so that UFT can understand objects and record the actions on them. When user replay, UFT replay (run-time) same steps as in recording (or known as design-time). If UFT finds any change during run-time from the design-time then it report the discrepancy. Is not it testing, matching expected outcome with actual outcome?

*Navigate to UFT | View | Keyword View*




***ConfuSense** – Graphical User Interface (GUI, sometimes pronounced 'gooey') is frontend visual representation of any application. The most common elements of GUI are WIMP ("window, icon, menu, and pointer").

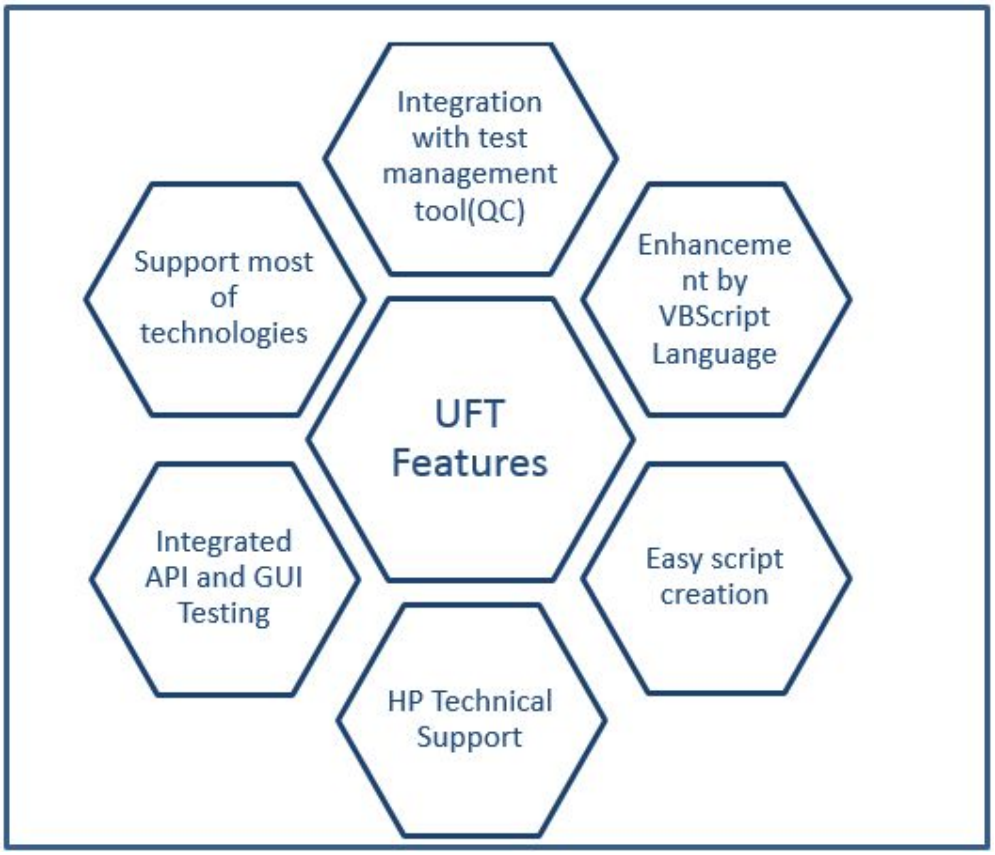
****ConfuSense** – Word "Object" have various meaning in different situations. UFT recognize the GUI buttons, menu, links, label, Browser, windows etc. and consider as Objects. So whatever you see on application screen is summation of GUI objects.

**5.0 UFT Features:**

Popularity of UFT is the outcome of various favourable factors. UFT provide support for various technologies, In-build capabilities and the integration with other HP products. UFT support easy-to-develop VBScript scripting language to enhance the test scripts. UFT can run test on remote machine.

UFT can integrate with ALM (formerly known as Quality Centre QC), a very popular and market leading HP test management tool. HP provides add-ins for all leading technologies e.g. SAP, Siebel, Java, .Net, Silverlight and so on. Mr. All-rounder, UFT, is perhaps the only tool to cover these many technologies (Please accept my apologies to assume UFT as male. Though gender of UFT has never disclosed by HP and I am taking my chance).

UFT provides integrated IDE for API* and GUI testing. UFT supports VBScript. Programmer can develop tests with UFT specific steps and enhance script with VBScript logic. UFT requires licence, it means that it is not free but it also means that HP provides crucial technical support when world seems to cease support for you.




***ConfuSense*** API (Application programming interface) testing escape GUI and directly interact with application. Application expose interface to programmatically interact with them. Basically API test provides a way to test technologies via HTTP, JMS or SOAP etc. requests directly to application. Previously there was a paid "Service Test" add-in required for API testing but current version of UFT comprises it for free.

## 6.0 Gossip Time – History of UFT

Mercury Interactive owned QuickTest Professional (version 5.5 - First release -in 2001) till 2006 and was subsequently acquired by Hewlett Packard (HP) in 2006. HP renamed "QuickTest Professional (QTP)" to "Unified Functional Testing (UFT)" after QTP10 with version UFT11.0. Currently UFT 11.52 (Released in June, 2013) is the latest version and available for one month with evolution licence. User can buy concurrent or seat licence afterwards.
UFT supports various technologies and make it fit candidate for most of the automation tests. HP support available via http://support.openview.hp.com/ portal. HP also provide other tools to assist testing. The name of Application Lifecycle Management (ALM formerly known as Quality Centre QC that also formerly known as TestDirector) can scare you but it is very easy to use and can manage the most of test management activities. Loadrunner is also a very popular performance testing tool.

**Aadhya**: Ok, I understood that UFT is a HP automation tool. We can program exact manual steps in UFT using UFT features and Visual Basic Scripting (VBS). We should analyse our requirements and choose the appropriate automation strategy.  UFT provides editor to write VBS script and Keyword

view to depict the steps with icons. UFT provides two type of licence that can either reside on fix machines (Seat Licence) or connect with pool on server (concurrent Licence).

**Aric**: That's absolutely correct! So, now you need to find someone to have patience for you lecture on "benefits of automation". If anyone try to lecture you on the same then you need to politely enlighten them when to avoid automation. You need to surprise your management by telling the automation evaluation process with full of jargon now.

## Summary

Automation is not to replace the manual testing but to assist and minimize the testing efforts. There are various benefits of test automation and various tools are available. But the best-fit tool and way of automation should be considered after thoughtful automation evaluation process. Neither one automation strategy fit for all application nor all tests can be automated in an application. All stakeholders should give due consideration on "Return on Investment" on chosen test automation tool and strategy.

UFT is distinguished  tool from other automation tools and consider as a leader with various favourable factors e.g. support many technologies, easy-to-use, easy-to-enhance, superior in-build features to assist automation testing etc. There are plenty of help available online but officially HP can be ask to look into particular issue.

## Quiz:

Q1. What is NOT the benefit of functional Automation?

A) It can save money
B) It reduce test execution time
C) Application of any technology can be automate with one tool
D) Test execution can be run on any remote machine

Q2. Which is the most famous functional automation testing tool from HP

A) HP Unified Functional Testing
B) Loadrunner
C) Quality Centre
D) Selenium

Q3. What was the old name of Unified Functional Testing?

A) Unified Test Pro (UTP)
B) Quick Test Pro (QTP)
C) Quick Functional Testing (QFT)
D) Quality Function Centre (QFC)

Q4. Which kind of test UFT can design in UFT Integrated Development Environment (IDE known as UFT editor)

A) API and GUI

B) API and Security Test
C) GUI and Performance Test
D) GUI and Accessibility test

Q5. UFT can work on which operating system

A) UNIX and Linux
B) Linux only
C) Mac only
D) Windows only

Q6. Choose the technology not supported by UFT

A) iPhone , Safari, Opera, UNIX or Mac applications
B) Web, Java, VB or .Net
C) SAP, Oracle, Siebel or PeopleSoft
D) Flex, Web Services, Silverlight or mainframe terminal emulator.

Q7. Which tool is NOT free?

A) Selenium
B) Sikuli
C) Waitr
D) UFT

Q8. To automate any technology by UFT, what component UFT must include in Test

A) Virtual environment for that technology
B) Appropriate Add-In
C) UFT plug-in for specific application
D) UFT driver for specific application

Q9. Which language UFT support for automation development

A) Visual Basic (VB)
B) Visual Basic for Application ( VBA)
C) JavaScript
D) Visual Basic Script (VBS)

Q10. What are the UFT view which does not show code but steps with Item, Operation and Value?

A) Workspace
B) Editor View (or known as Expert View)
C) Macro Editor
D) Keyword View

## Explore:

Q1. List down all the free automation tools and their limitations

Q2. Write a case study where automation should be avoided

Q3. Find out the History of UFT, companies and version information

Q4. Find out that how VBScript is different from VB and VBA

Q5. Search why UFT cannot run on any operating system other than Windows.

## Reference:

1.0 http://en.wikipedia.org/wiki/Test_automation

## Be Technical Star

Problem1: Identify the common programming languages used for various automation tools.

Problem2: Write a pseudo-code (algorithm) to generate odd numbers.

Problem3: Write a pseudo-code to reverse any numeric integer

Problem4: Explore the VBScript math functions

Problem5 Find out VBScript features different from other languages.