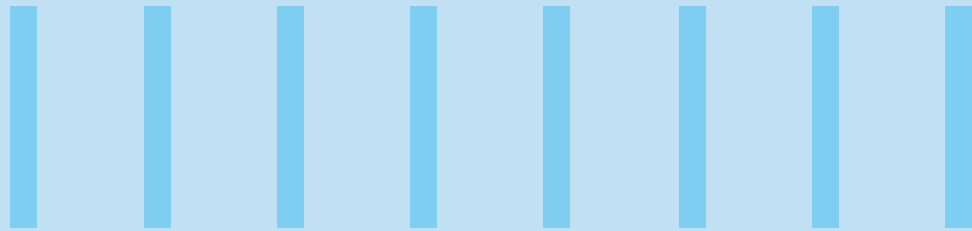




TEST AUTOMATION FRAMEWORK – HOW TO CHOOSE THE RIGHT ONE FOR DIGITAL TRANSFORMATION?



Introduction

Digitalization and the disruption caused by the adoption of digital technologies are rapidly changing the world. Speed matters a lot in all IT operations, and this calls for a paradigm shift in quality assurance (QA). Quality at high speed is the key focus in digital assurance, and organizations want to deliver quality products much faster than ever before. This is making QA teams to bank on test automation. From the initial automation of regression tests, the industry is moving towards progressive automation and day one automation. At the same time, extreme automation and zero touch automation are the buzz words in the QA world these days.

Various advancements have evolved in the area of automation testing. However, it is critical that organizations choose the right automation framework, which is considered a critical factor for its success. In this document, we will explore the different types of automation frameworks, and how to choose the right framework which will help in achieving the digital assurance goals of the organizations.

What is a test automation framework and what are its different types?

A test automation framework is a combination of guidelines, coding standards, concepts, practices, processes, project hierarchies, reporting mechanism, test data, to support automation testing. A tester follows these guidelines while automating applications to take advantage of various productive results.

Keyword-driven framework

In the keyword-driven framework, testers create various keywords and associate different actions or functions with each of these keywords. Function library contains the logic to read the keywords and call and perform the associated actions. Generally, test scenarios are written in excel sheets. The driver script reads the scenario and performs test execution. This is used in situations where the testers who create test scripts

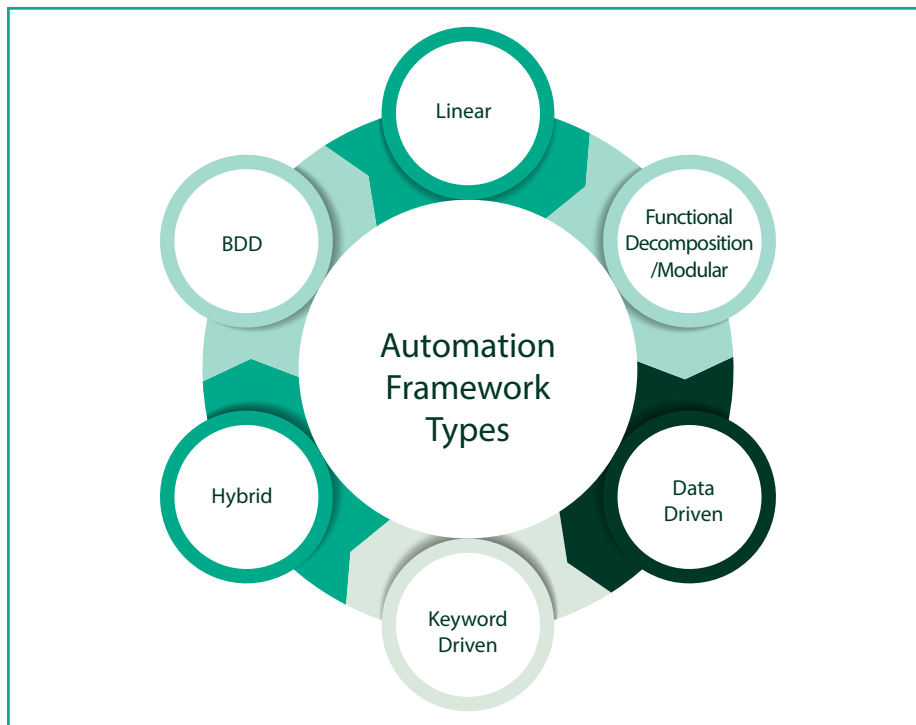
the strengths of the different frameworks and mitigates their weaknesses. It is highly robust, flexible, and more maintainable. However, this requires strong technical expertise to design and maintain.

Behavior-driven development framework

Behavior-driven development (BDD) framework automates validations in an easily readable and understandable format to business analysts, developers, testers, etc. Such frameworks do not necessarily require the user to be acquainted with any programming language. There are different tools available for BDD like Cucumber, JBehave, and more which work along with other test automation tools. This framework is more suitable for applications using agile methodology and where user stories and early automation are required. It focuses on the behavior of the system rather than the implementation aspect of the system. The traceability between requirements and scripts is maintained throughout, and test scripts are easy to understand for the business users.

Pillars of the right framework for the digital era

Automation can improve quality and lead to higher testing efficiency. Hence, it is important to plan it well and make the right choice of tools and frameworks. When test automation uses the right framework based on the context, it yields great benefits. Hence, it is worth understanding the key requirements of the framework, before choosing the right one.



There are many types of test automation frameworks available in the market, and the most popular ones are listed here.

Each one of these frameworks has their individual characteristics and features.

Let us now examine some of the popular frameworks and understand their pros, cons, and usability recommendations:

have less programming expertise, whereas framework creation is done by automation experts

Hybrid framework

The hybrid automation framework is created by combining distinct features of two or more frameworks. This enhances

Some key aspects of automation framework to look for during the digital assurance journey are provided below:

Extreme automation

Digital transformation programs, big data, cloud, and mobility are changing the way testing is being done. Leaders in testing are moving towards extreme automation to achieve a faster time to market. Extreme automation is the key, and automating every part of the testing process instead of just regression is crucial now. A framework which is more scalable and facilitates lifecycle automation as well as broader test coverage is needed for digital assurance programs.

Technology and tool agnostic approach

The landscape of tools in QA is becoming wider day by day. There are too many tools and frameworks, which poses a lot of integration challenges. Hence, it is imperative to choose a framework which is technology and tool agnostic and supports various tools and technologies. The framework needs to address enterprise-level automation strategy and goals instead of catering to just a single project goals.

Script less capabilities

Automate the automation, and look out for scriptless automation avenues. Most software testers and business users find it challenging to learn programming languages such as Java, Visual Basic, etc. well enough to write the scripts that the test automation demands. There are frameworks and accelerators available with user-friendly graphical user interfaces (GUI) which help to create automation scripts in a much easier way than having to know and write code in any specific programming language. Choosing a framework which helps to create a test script from the recorded script or based on the input from a spreadsheet will help in accelerating automation and reduce dependency on skilled resources.

True shift left attitude

Digitalization and frequent releases call for day one automation. Gone are the days when the automation team would wait until the application is built and start automation activities thereafter. The need of the hour is to shift extreme left and start automation during the requirement gathering phase of the systems development life cycle (SDLC)

itself. Automation framework with exhaustive reusable library and support for BDD will help both business users and QA teams to start automation activities early in the life cycle.

Omnichannel, mobility, and cloud features

Organizations today are focusing more on digital assurance, but it is important to test real user behaviors and to test on multiple devices such as various mobiles, tablets, platforms, and such. Hence, the chosen test automation framework needs to facilitate testing on multiple devices to ensure a uniform experience across devices. If the framework supports the reuse of the script used for online or desktop testing for mobile testing as well with minimal rework, it will help in saving much effort. When addressing the multifaceted needs of mobile testing, conducting comprehensive testing across hundreds of different devices, brands, models, and different operating system combinations is tedious. A framework that facilitates integration with cloud infrastructure will be an added advantage.



Zero touch automation

As DevOps is slowly taking over the IT landscape, it is vital to reduce the distance between development and deployment. Test scripts need to be executed in an unattended manner without requiring much manual intervention. Remote execution, parallel execution, zero touch execution, and execution from continuous integration tools like Jenkins and Hudson, when supported by automation framework, will help a lot in managing multiple sprints and shorter cycles better.

Seamless integration

With a plethora of tools being used in application development and testing landscape, it is important that the automation tool and the framework chosen facilitate integration with various tools. Hence, it is imperative that the chosen automation framework and tool integrate with test management tools, defect tracking tools, build tools, analytics

tools, and continuous integration tools in the landscape.

User-friendly reporting

Agile and Dev Ops has brought the business, development and QA teams to work together. The ability to run a high volume of tests is of little use if the results of the tests are not easy to understand by various stakeholders involved. The framework has to facilitate automatic generation of reports of the test execution and show the results in an easy-to-read format. Though most of the market tools give few reporting options, they are not self-explanatory and adequate. Hence, the framework with good reporting capabilities such as HTML reports, live execution dashboard, screen shots in case of failures, and video reporting of the execution options will be very helpful. Automation framework facilitating detailed test result reporting reduces the overall effort to a greater extent.

Conclusion

No one size fits all. This perfectly holds true when it comes to framework selection. Since every project is unique, the challenges, duration, and tools choices may vary. Organizations seeking agility in their business processes need to onboard robust test automation solutions that ensure superior software quality. Successful test automation frameworks for digital assurance are the ones which support extreme automation, omnichannel testing, zero touch execution of test scripts, and have some or all of the key aspects detailed above. We recommend that organizations select an automation framework that can lead to smarter automation, better overall results, productivity benefits, and cost efficiencies in the highly dynamic digital landscape.

About the author

Indumathi Devi, a project manager with Infosys, has 13+ years of experience in software testing. She has effectively executed a multitude of automation projects and designed and developed automation frameworks. Using her strong working knowledge of multiple test automation tools, including open source and commercial ones, Indu has worked with numerous clients in implementing robust test automation solutions.

For more information, contact askus@infosys.com

Infosys[®]
Navigate your next

© 2018 Infosys Limited, Bengaluru, India. All Rights Reserved. Infosys believes the information in this document is accurate as of its publication date; such information is subject to change without notice. Infosys acknowledges the proprietary rights of other companies to the trademarks, product names and such other intellectual property rights mentioned in this document. Except as expressly permitted, neither this documentation nor any part of it may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, electronic, mechanical, printing, photocopying, recording or otherwise, without the prior permission of Infosys Limited and/ or any named intellectual property rights holders under this document.