



Tenjin
ONLINE

Changing business landscape: How test automation can help?



Introduction



The app ecosystem is ever evolving, which is subjected to technological advancements, changing customer preferences, and shifting business landscapes. The app market has grown rapidly and accounts to \$206 billion for mobile apps and about \$5 billion for web apps respectively, giving scope to businesses to tap into the immense potential it holds. These huge numbers indicate the growing significance of apps and hints to a future that will be app centric. Businesses are taking advantage of the growing app market and creating innovative apps that connect with users, reach wider audience, enable faster delivery, and eventually improve the ROI.

As options are increasing, people are getting picky about their choices. Apps with even minor glitches can be beyond users' tolerance and may lead them to turn towards competitors' apps. Hence, testing them is an important part of the entire app development process that works towards improving the app quality. A well-tested app will create a lasting impression, engage and retain existing users, and attract new ones.

To understand the importance of testing let us take a use case scenario. TikTok app was introduced in the market in September 2016, and it managed to grab 850 million global downloads on both Google play store and Apple store by 2022, despite being banned in countries like India, Afghanistan, Pakistan, Bangladesh, and Indonesia, it has emerged as the top downloaded app worldwide. What makes it so special to have a whopping number of downloads? The app is simple with platform allowing to create and share videos ranging from few seconds to minutes duration. The reason for its success is its flawlessly working features that is designed to handle a traffic of even 850 million users (which accounts to nearly 10% of the world's population) all at once. It makes it clear that the app has been tested a hundred times and all the glitch that raised during development or upgrades were fixed before making it available for the users. The well tested and resolving approach made the app stand out in the market despite having similar competitors. Hence, it is clear that testing is a critical part in the success of an app.

Different app testing methodologies have been introduced to improve speed, accuracy, and efficiency. This whitepaper discusses the changing business landscape, the challenges in app testing, and the best practices to follow.

Adapting to the changing business landscape

Business landscape is rapidly changing as new business models are being introduced, wider customer demographics are catered, and new technologies are making their presence. Further, competitors moving ahead in this changing landscape increases the risk of not capturing a larger audience as they may be attracted by the new technology implementations by the competitors. Hence, organizations should adapt to the ever-evolving business landscape to cope up with uncertainty that lies ahead and secured a good market positioning.

The demand for faster and frequent release is critical to surpass the competitors and reach the audience soon. In order to do so, organizations are adapting modern agile and DevOps development practices. To resolve quality-related issues across these complex development approach, organizations are embracing automation techniques. A survey by InfoWorld highlights as why test automation is an inevitable requirement - 88% increase in speed were recorded to 50% of the tests that were run, 68% increase in earlier identification of bugs, and 71% increase in test coverage. With these numbers, it is evident that testing increases development speed, saves time, and reduces cost. But this was not

Let's understand how the testing landscape has changed over years, how organizations are embracing the change, and how experts foresee the future.

Manual Regression

Before the era of digitization and automation, software testing was done manually. Testing was an extended part of the whole software development cycle and not a standalone sector. It was performed after the entire development process was complete and required a team of skilled programmers to write the code and carry out frequent regressions.

As manual testing involves a dedicated team to write codes, manually identify errors, and document the issues, it is highly tedious and time-consuming. The complexity of manually performing the testing further increases the chances of human errors. Hence, the whole process of manual testing was increasing the cost and turnaround time, while affecting accuracy and efficiency. This created a new path of introducing automation in testing to address the challenges faced by manual testing and speed up the process.

No matter how much the technological evolution happens, manual testing will still be relevant across all timelines, even in the future. Here are some significant points that will explain why manual testing can never go out of fashion:

- No new-gen technology can ever imitate human intelligence, logical reasoning, and analytical behavior. Scenarios like illogical navigations, testing for user experience, usability issues, etc., cannot be performed to 100% accuracy by a machine.
- Even in the event of automating the process, continuous monitoring is required by human being to have the situation under control.
- Sometimes the application can give unexpected results due to hidden bugs, which can be identified and analyzed only by human intelligence.

Automating process will also need manual intervention for initial process setup, monitoring, and report analysis. Automation testing can never be 100% independent of human intervention.

Code-based automation

Code-based automation refers to the testing processes that are carried out by writing codes. It requires highly skilled individuals with programming knowledge. Here, the coded automation works with union of machine and tester, where the tester is involved in test creation and the machine takes care of execution. As testing is executed automatically, the occurrence of errors is reduced in code-based testing. It is more efficient than manual testing and involves lesser time and cost comparatively.

Previously, when organizations were operating completely on manual approach, they found repetitive scenarios where bugs weren't identified efficiently, causing huge business risks. Hence, code-based testing was introduced in which codes were created by the tester and automating the execution; the whole process is referred to as code-based automation and has tremendously reduced human errors, time, and cost which were referred to as major challenges while performing manual testing. It should also be noted that code-based automation further improved accuracy and efficiency of the process.

Codeless test automation

As the development processes are moving from outdated models to more efficient agile models, how can testing be any behind? Codeless test automation is the right solution for the modern developmental processes. It efficiently tests the applications across the complex DevOps pipeline and identifies errors quickly and efficiently

Codeless testing or codeless test automation as the name refers is the process of creating and executing automated tests without writing codes. As the need for writing codes is eliminated, even a non-technical person without the knowledge of programming can carry out testing efficiently. As the whole process is automated, it considerably reduces the time, cost, effort, and human errors while improving the accuracy and consistency like never before.

Codeless test automation is the advanced version of manual and code-based automation and offers much more accurate results than its other counterparts. It is implemented to achieve quality at speed to help organizations release quality applications in the market quickly, thereby helping them achieve their business objectives, build brand, and improve ROI.

Reasons to choose codeless test automation



AI and ML integrated testing

AI (artificial intelligence) and ML (machine learning) - the two technological jargons I bet anyone is unaware of. AI and ML are the future technology which have already made their presence in various industries to automate their processes with a human touch. These advanced technologies are slowly making their presence in the software testing industry too, for improved intelligence and speed.

AI-integrated testing mimics human behavior and carries out testing with logical reasoning and predictive analysis. This model not just checks the actual results with the expected result but tests the app from user perspective and predicts any risks that may occur over a period of time.

Offers High
Efficiency

AI and ML integrated testing processes will soon revolutionize the app industry to delivery top-quality apps in reduced time. This is the future of app testing and will change the landscape of the software testing for better.

Automation testing requirements are increasing

Companies are constantly updating their systems and processes to keep up with the changing trends and offer updated services/ solutions to their customers. The same is true for app testing too, where testing should be updated and aligned with the development process to provide solution that will match with the customer preferences.

Developmental processes are no longer based on older models. Today is the time of advanced developmental processes like agile methodologies and DevOps, which demand advanced and agile testing process to support development and yield expected and exponential results. Traditional testing models will do no justice to the agile development process, hence, the need for advanced testing process is required. Further, regressions are high, and the frequency of testing should be increased which could be time-consuming and prone to errors when performed manually.

Codeless test automation is the need of the time, it supports continuous testing and performs across CI/ CD (continuous integration/ continuous delivery) pipeline. CI/ CD is the core differentiator of DevOps app development process, to ensure that the quality is uncompromised during this process it is important to implement continuous testing. Continuous testing cannot be performed by manual or code-based automation due to its complexity and highly rigid nature; due to major human intervention in both the processes, the process will be prone to human errors causing deviation from the expected result. Hence, codeless test automation is the most feasible solution to achieve quick and efficient results.

Going codeless is essential to sustain in this fast-paced world, keep up with the trend, and move ahead of the competitors. Codeless approach skips the tedious and time-consuming coding or scripting stage, making the test creation and execution part simple and easy for the tester. All that the tester needs to do is set up the process, monitor the flow, and analyze the report, rest of the testing activities will be taken care of by the machine.

As agile processes are slowly being adapted by organizations, the app release and upgrades come in iteration rather than a one-time release. With this changing approach of development process, incorporating testing as a one-time practice at the end of the SDLC (software development lifecycle) will not serve any purpose. Testing should be involved right from the beginning of the development cycle, which can only be justified by codeless automation.

Organizations have to constantly make changes to meet the modern standards, the processes and techniques used a few years ago may not be relevant today. This is true in the case of app testing too, traditional testing methods like manual testing and code-based testing are not relevant to today's customers preferences, codeless is the solution to meet the speed and quality requirements.

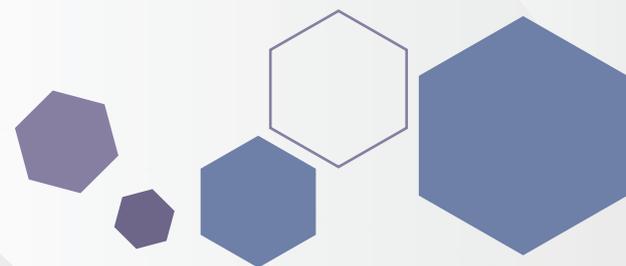
Automation helps in testing

With the growth of DevOps, testing processes has been redefined to include planning, execution, monitoring, and releasing. Here, testing (continuous testing) is introduced in the early stages of development to test every unit as it is being developed, to ensure that the quality is not compromised. The QA process, however, involves all the testing activities in addition to user expectation based on user's feedback. The introduction of user inputs makes the process user-centric and helps organizations to develop products to cater to what users are expecting.

As testing is an integral part of the development process, it must be carried out with precision to get the expected results. Further, testing should be performed quickly to match the speed of fast-paced development process. Under these circumstances, the load on manual testing can be reduced with the help of automation.

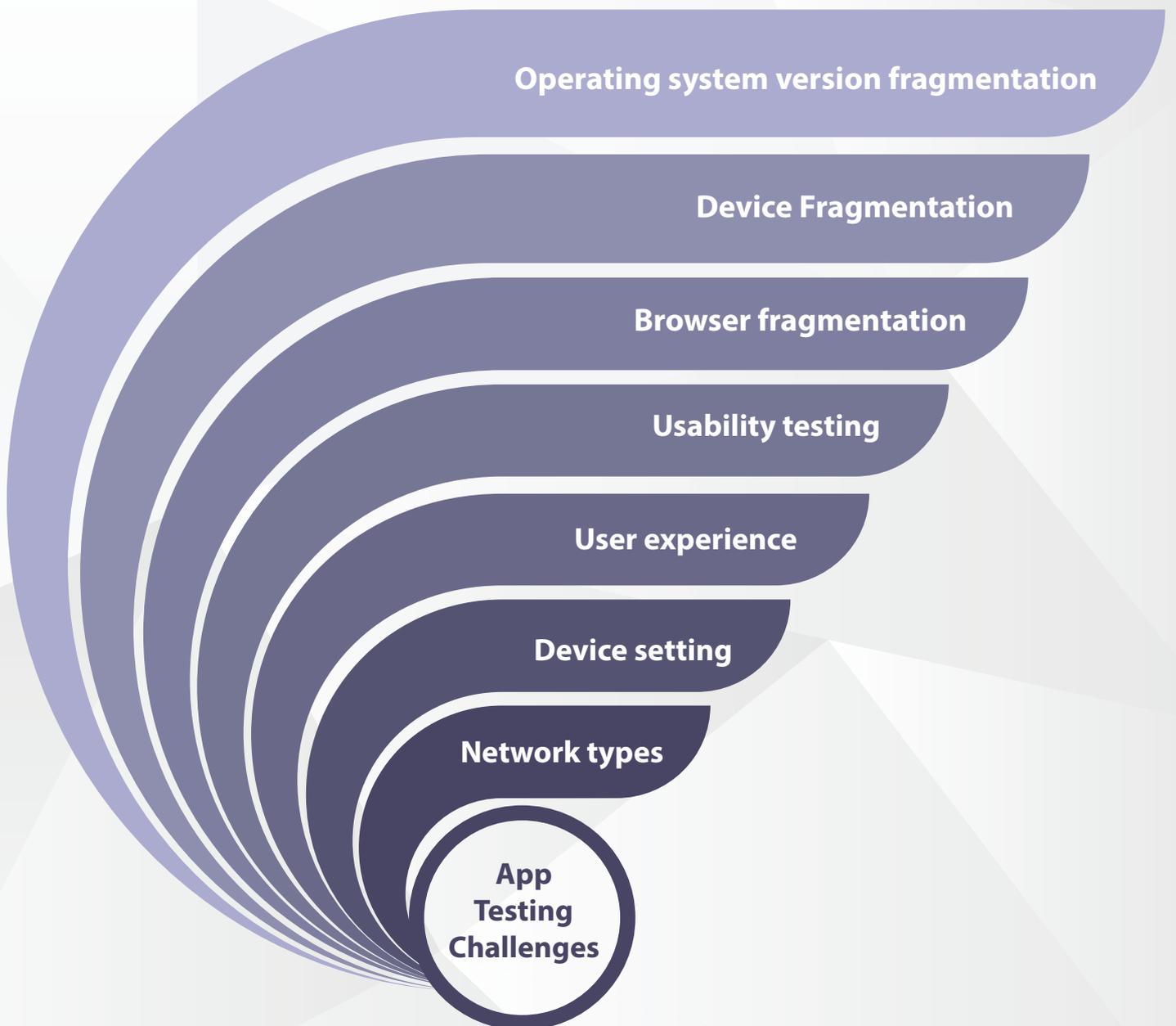
Automation helps in testing the app under rare conditions, which will otherwise be difficult to carry out manually. For example, mask defects may not cause any failure in the application, and to identify them manually may be a daunting task. Automation helps to ease the process and identify even the trickiest error in lesser time.

Scenarios that involve testing multiple devices at once, may be prone to errors when tested manually due its complexities. Automation helps in reducing the errors, increases speed, and delivers efficient results.



App Testing Challenges

Testing app is not a breeze, it involves ample planning, technology selection, work allocation, execution, and report analysis. In this process, there are a number of challenges that company's face in order to yield efficient results. Addressing these challenges and finding a way to resolve them will allow to improve app quality; in case organizations fail to address these challenges, the quality will be hit causing reputational and monetary losses.



They key challenges that company's face while testing the app are:

Device fragmentation: There are a diverse range of devices available in the market with different screen sizes, screen resolutions, operating systems, network types etc. In order to ensure that the app works perfectly on smartphone, desktop, laptop, tablet, smartwatch, and other digital devices, it is important to optimize its performance across all the devices. However, the issue of fragmentation has always been a challenge and made the testing process tedious.

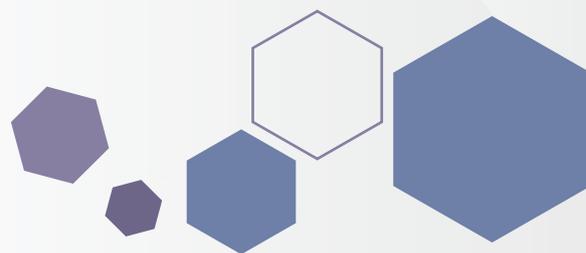
Network types: Internet connection is evolving, which was earlier available as 2G and 3G connection is now available as 4G and 5G. In addition to mobile data, the trend of Wi-Fi is also increasing. However, not everyone uses the same internet connection type, this makes app testing with different network types very difficult.

Browser fragmentation: There are different web browsers available in the market like Chrome, Firefox, Microsoft Edge, etc., To test the web app across all the browsers is difficult due to high fragmentation. Hence, browser fragmentation is one of the major challenges faced by companies today while testing web apps.

Usability testing: App usability is the factor that ensures if the design is perfect, engages users, navigates seamlessly, and connects with the user. As apps are tested by technical team and by a few users at the piolet phase, the audience reaction remains mystery until release and used by the real-world users. Usability testing is another challenge that companies face and need to be addressed.

Operating system version fragmentation: Web and mobile apps constantly release new version of the operating system to offer seamless user experience. It's important that the devices or browsers are upgraded to the latest version to improve app performance. The upgrade should not be affecting the app's performance in any way; hence, it should be tested, however, the wide range of OS versions available causes fragmentation and is one of the major challenges faced by organizations.

User experience: Until an app is offering an excellent user experience, it can hardly succeed. Creating an app that fulfills users' preferences is not an easy task, it involves checking of features, functionalities, performance, security, and all other functional and non-functional aspects that can perform well and connect with the users. The whole process of improving user experience involves a subset of many tasks that must be performed with attention to details to avoid any discrepancy in the bigger picture.



Device setting: Some of the settings in a device are specific to that particular device and testing for device-specific settings is important to ensure the app is working fine even with specific settings. This is one of the challenges faced by companies while testing the app.

These are some of the prominent challenges in app testing which should be addressed in order to improve the quality. When these issues are left untreated, they can cause serious consequences causing

Web and mobile testing best practices

To yield the best outcome from test automation practices, one should incorporate best practices to avoid any confusion, streamline the process, and get best results. Here are some of the best practices that one should follow:

Defining Test Coverage



Testing and development should go hand-in-hand



Shifting to codeless test automation



Identifying the appropriate test data



Choosing the right tool



Define Test Coverage

In order to find more errors in an advanced manner, test coverage should be increased as higher test coverage often give nearly 100% accuracy. Hence, determining test coverage is an important consideration to make; increasing coverage will improve the test results.

Testing and development must go hand-in-hand

Testing should be introduced right in the beginning till the end of the development cycle. The errors are identified as the app is being developed and resolved before moving on to the next sprint. This process improves the quality and saves time and cost.

Shifting to codeless test automation

Shifting to codeless testing is the need of the time to align with modern practices. Codeless test automation eases the entire testing process and reduces time, cost, and effort. It doesn't require coding knowledge, hence, freeing the testers from creating new code-based test cases and focusing on implementing new features or fixing bugs.

Identify the appropriate test data

It doesn't matter how exceptional the test scripts are if data provided is insufficient. Hence, rich data should be provided to test the app efficiently and uncover distinct details that might otherwise arise due lack of error indication when fed with insufficient data.

Choosing the right tool

There are hundreds of tools available in the market, not all of them will suit your individual requirement, choosing the right tool is essential to get the desired outcome. Decide on what kind of testing to carry out, functional, non-functional testing, or both, whether code-based or codeless testing approach to be implemented, what is the budget, etc. Based on the requirements, the right tool should be chosen.



Conclusion

Various testing processes have been tried and tested over decades to improve the quality of the developed applications. The methods have evolved with the evolution of development processes, technological advancements, changing user preferences, time constraints, changing business landscapes, cost, and so on. To cope up with all these changes yet offer quality outcomes, organizations are using advanced testing processes like codeless test automation. The need for automation, changing business landscape, associated changes, and best practices are mentioned in this whitepaper.

No matter which testing technique an organization is practicing, the aim is to get efficient outcomes. Because when issues are not identified effectively and left untreated, they may cause profound consequences in functioning of the final product. Hence, to create an app with greater user experience, reach wider audience, and improve business revenue, efficient testing is the key.

About Tenjin Online

Tenjin Online is a powerful SaaS-based, codeless test automation platform for web, Android and iOS apps. It has the capability to conduct functional testing, UI testing, project tracking, and defect management with minimal or no human intervention. Try out this revolutionary tool to improve the quality of the app which will eventually improve business returns. Further, pre-integrated JIRA and Device Farm makes the test tracking and management more easy. JIRA makes the defect tracking and management easy and efficient, while Device Farms like BrowserStack and SauceLabs provide access to a wide range of devices on Cloud.

Disclaimer

The whitepaper attempts to provide detailed insights of the subject matter. The details included herein can be subjected to changes at any point with the changing market trends, without any further updates. Information provided are based on market research, latest trends, historical information, and industry statistics. The whitepaper is intended for knowledge sharing and reproduction of its content in any form is forbidden.

For any queries, feel free to write to us - hello@tenjinonline.com