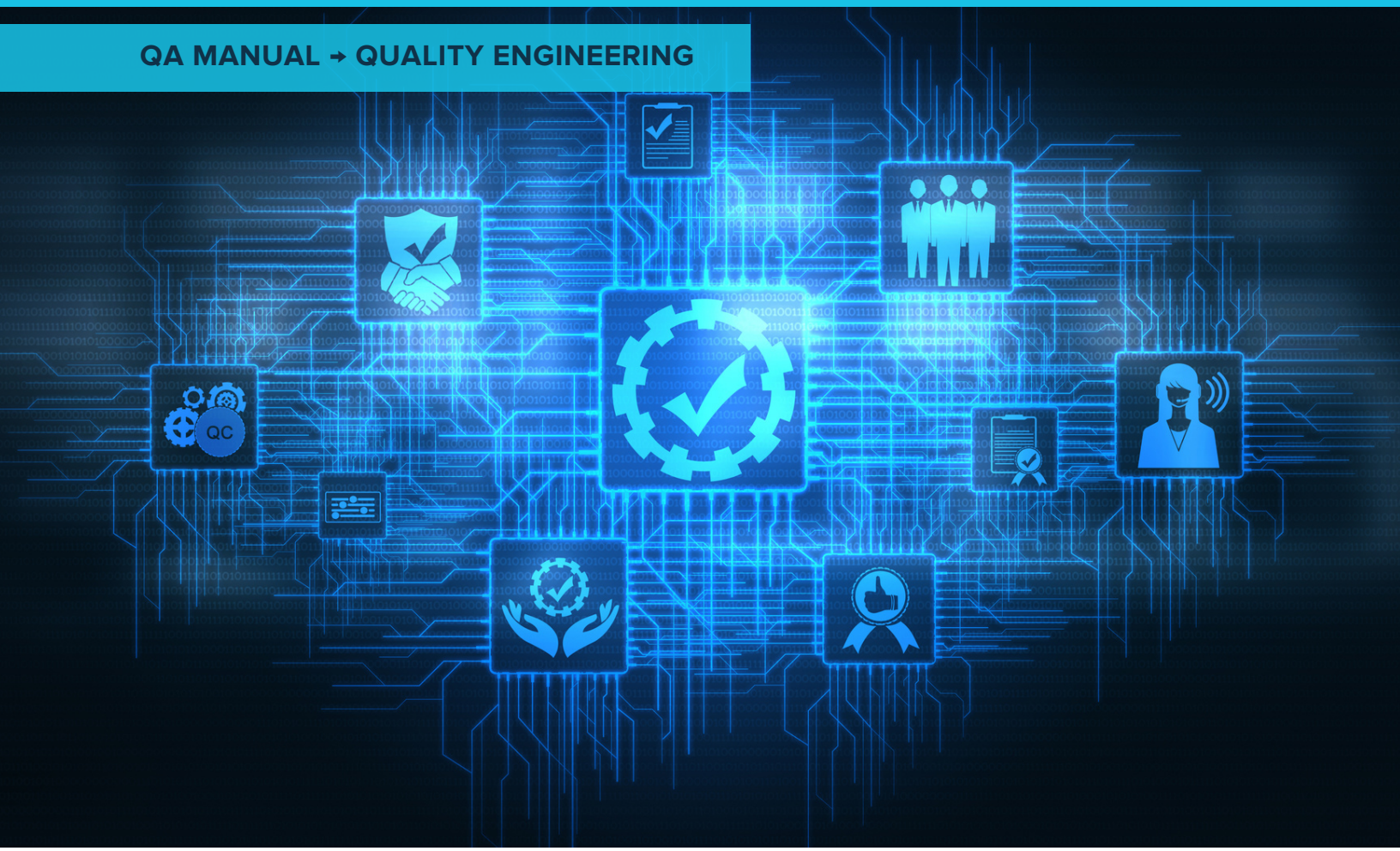
JUNE 2023

GROWTH ACCELERATION PARTNERS

TESTING TIMES: HOW THE BEST QAs GET RESULTS IN ADVERSE CONDITIONS

A QA Manual Team Remains Vital for Your Application Rollout — And GAP Can Help

QA MANUAL → QUALITY ENGINEERING



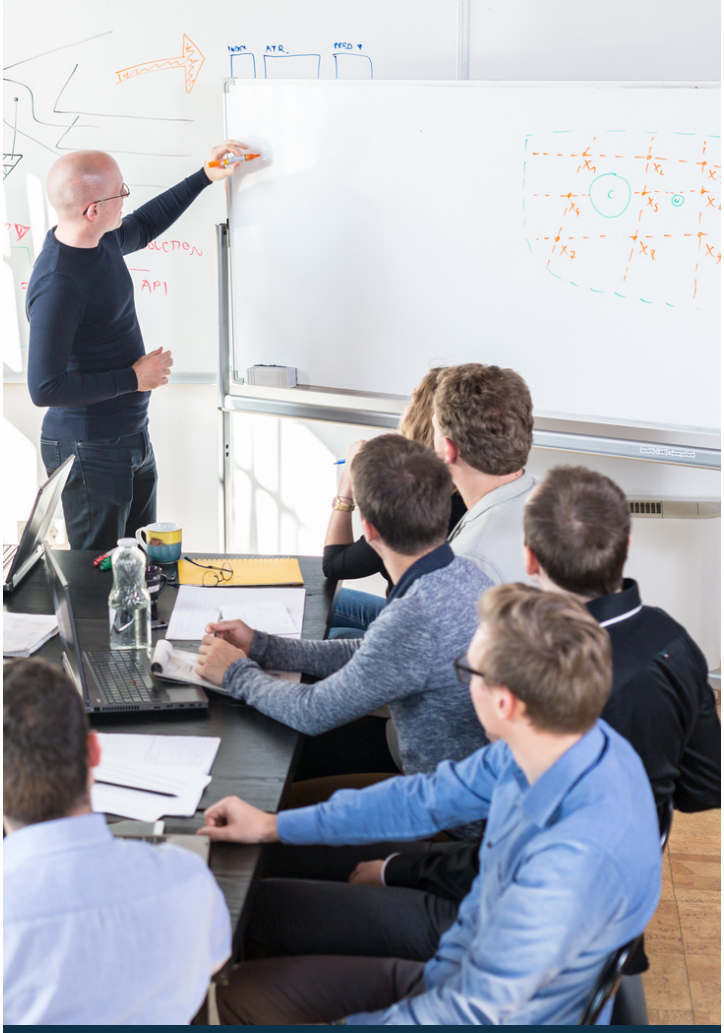
Engineers know quality assurance (QA) is crucial to ensure software meets planned functionality, and will perform as expected (and minimizing the number of bugs) under a variety of circumstances. Yet for the benefits of automated QA testing around time-to-market, scalability and freeing resources, automation alone is not enough.

WeAreGAP.com

If you have repetitive , high-frequency, time-consuming tests in a stable environment, then automation makes sense. But do not fall into the trap of thinking automation experts will solve all problems on the QA side. To give one example, automation is not going to work if you are in the middle of development on a feature and conditions change frequently. Manual testing can not only provide a more rigorous service, but it can save resources – namely time and money – as well; and having a trusted partner who understands this can mean the difference between project success and failure. Equally, do not make the mistake of thinking the QA manual team's job is simply to get a piece of software, make a bunch of clicks, and tick a few boxes. The fundamental guidelines that any qualified QA team will practice come from the industry de facto International Software Testing Qualification Board (ISTQB). When it comes to designing tests, for example, the ISTQB advocates a three-step process:

- Test conditions
- Test cases, such as inputs and expected outcomes
- Test procedures — the actual sequence of actions or steps to execute the test

Take functional testing as an example. Functional testing is defined by the ISTQB as a type of testing that checks the functionality of a component or whole system, compared with non-functional testing, which concerns itself with usability, performance and security, among others.



The QA manual team will define the test, plan the test — selecting particular scenarios, and creating test cases for them — and then execute those cases in an ideal scenario. The problem — and this is what separates a good QA strategy from the rest — is that many projects do not live in an ideal world.

There are various challenges organizations face when conducting such projects. The two primary issues are around budget and time. Four common types of software development environments seen across different projects are development, test, staging, and production; and alongside this, there are multiple testing types, from alpha, to beta, to security, to performance.

GROWTH ACCELERATION PARTNERS

It must be said that a good QA team by all means does not need all possible environments to work in; though one current GAP project has sandbox, QA, beta and production available – an ideal scenario. But if, for example, you only had development and production available to you, then this could become risky. If you have only two environments, and one is where developers are pushing changes multiple times a day, then this is not stable for performing QA, and issues going into production can result.

For a comparison, let's take a staging environment, also known as beta. This is designed to mirror production as closely as possible to ensure your software behaves as it should in such an environment. But it is usually recommended to have a test environment as well as a staging environment, with the former to make sure every component does its job properly.This is where an Agile methodology comes in best — adjusting to conditions and doing the best with what you have.

Fostering the correct mindset is another challenge for organizations. Some organizations may not fully appreciate the value of QA, and have their developers testing what they are building, which is evidently not good practice. Elsewhere, if you are a startup with multiple plates to juggle, then it can be difficult to get the processes in place to create an Agile environment. The client may use or prefer an Agile methodology, but they are not yet at the right point of expertise.

This is where a partner such as GAP can come in, to understand the project, the limitations — be they time- or money-based — and go end-to-end, building and implementing a manual QA strategy.

Forward-thinking organizations will realize getting this type of testing right can be a long road, and finding the right partner will ensure a lengthy, fruitful relationship. One client started working with GAP around four years ago, and they fell into the trap of identifying the need for manual testing once the automation testing service had begun. The plan was devised to start with manual and then postpone automation for a later stage, when features were more stable. But the product then grows significantly to have lots of features.

Eventually, the correct balance was struck between manual and automated. Now, the client is releasing once a week; all features are properly tested, while the regression is run by the automation team.

Utilizing a partner such as GAP can also provide value-added services, such as providing metrics information, identification of improvement areas, and analysis on the quantity of successful releases, issues related to production, and new features released to production among others.

Getting the right strategy to suit you, getting the balance right between manual and automation, and putting a full end-to-end process in place is not easy, whether you're a startup or an established organization. As a QA services company, GAP will put your applications through their paces with the right processes in place to ensure they're customer-ready.

To find out more, please visit WeAreGAP.com