



Automate *Or Not* To Automate

Introduction

It's not possible to work in software quality assurance and not have heard about automation. Some people swear by it and recommend automating everything possible; others feel that leaving the human, manual element is more effective, even if it is less efficient.

Automation vendors are going to sing the praises of automation, for certain, but just how can automation help you, if at all? Should you automate, or not? Is it worth the time, money, and effort to find, buy, utilize, and maintain automation tools?

Unfortunately, the answer isn't simple. There are many ways that automation can help, and ways that it cannot. Some projects will do better with automation, and for others it could be a waste of time. Some types of testing must be automated, while for others it's far more difficult. How do you know if it's right for your project or organization?

Determining if Automation Can Benefit You

Before deciding to automate, you should look at your current project status to determine if automation could help you at this point. Consider the following questions:

- Do you have a lot of frequent regression cycles to run?
- Do you have a stable GUI but frequent code changes?
- Are you using a waterfall approach where changes are delivered all at once?

If the answer to any of those questions is yes, then automation should be strongly considered. You possibly have both the need and a good foundation for automation. Regression tests are the most ideal candidates for automation since they are generally stable tests and re-run frequently. If your regression test cycle consumes more time than you think it should and it's being performed frequently, then automation may help reduce the testing cycles.

Automation also works best for applications that have a fairly stable user interface. If your user interface changes frequently, then most of the automation tests will need to change frequently as well. This is why it's recommended that the GUI is stable before embarking on automation. If there are future plans to change the GUI, then it might be beneficial to hold off on the automation until that's complete.

Determining if Your Team is Ready

You may have a need for automation, but what about your team? A successful implementation requires a significant resources, time, and skills.

- Do you have a solid, comprehensive, and easy-to-understand manual test repository?
- Do you have the budget for a tool purchase?
- Do you have staff with the skillset to utilize automation tools (free or otherwise)?
- Do you have the budget for staff training or new hires?
- Do you have the time to dedicate to getting automation off the ground?
- Do you have the time to dedicate to automation test maintenance?



Finding the right tool for the job takes time, and the tool identified may be expensive. There are free tools out there, but they may or may not be suitable for your project. Having the budget ready to handle the possible purchase is ideal.

Existing staff skills, training, or new hires are also a significant consideration. All of these contribute to the time necessary to automate, the budget required, and the amount of effort involved.

Common Problems

The problem many QA Organizations encounter is one of assuming automation will be the answer to all of their problems without realizing the amount of effort that must go into a successful automation endeavor. Teams must be able to effectively use resources and time, select a tool appropriate for their project(s), and have the skillset available to use the tool correctly and efficiently.

Incorrect Tool Selection

Sometimes budget drives what tool is purchased and the one selected may not be the right one for the project. The right tool is critical to the success of automation. Taking cost out of the equation, the tool selected must be compatible with the test environment, development languages, and the testers that will be using it.



Thorough research should be performed on available tools and hours of demos may be necessary if you are selecting the tool yourself. The alternative is to bring in automation consultants to evaluate your development and test environment, style, and staff skill levels. Outside experts can provide a team with years of experience to counsel them on the right tool for their particular project. While there is certainly an expense involved in that route, the benefits are even greater. By utilizing expert advice you can drastically cut down on the research and demos needed to evaluate tools yourself as well as the risk of trial and error failures. In QA Mentor's experience, this first step can lead the way to success or failure.

Inadequate Staff Skills

Another common issue with failed automation endeavors falls into the domain of staff skills and by extension, staff training. Once a tool has been purchased, the team needs someone with appropriate skills to use the tool. Learning automation takes time and training, and far too often businesses lack one and forgo the other. This can create a heavy burden on the tester or testers tasked with implementing automation.

For automation to be successful, either a member of the team needs to have existing automation skills and experience, or one or more team members need to be trained. Training can be done by the publisher of the selected tool, external workshops, or automation consultants brought in-house for a short period of time. Regardless of which way is chosen, training is necessary for successful automation.

Aiming for 100% Automation

Some teams aim too high and expect that they can automate every single aspect of testing. This is not the case, and believing it is only sets teams up for disappointment. Automation implementation is a process that takes time, and that requires manual tests be done in conjunction.

Some areas of the application under test may not be suitable for automation, or may require manual reviews of data, such as reports. Also, new functionality must be tested manually and made stable before automation can be introduced. Generally speaking, regression tests are most suitable for automation.

Keep your expectations in line with automation capabilities to reduce stress and disappointment. Aim to automate regression tests first, then look to what else might be a candidate for automation. But, don't expect that manual testing is in the past no matter what your automation goal is.



Poor Test Design

In some ways, this does relate to the skills of the testers on the team. Test design is really an art that requires experience and patience. While some testers can both design and engineer automation tests, doing so could be a bit daunting.

Poor test design can make tests ineffective and difficult to maintain. A good suggestion would be to divide the automation tasks up. Have one tester (or group of testers) work on test design, while other work on engineering the tests. This helps to divide up the work effort and allow testers to focus on one specific aspect – and in the process get very good at it.

Neglecting Maintenance

Another misconception about automation is that once the tests are written, they don't need to be touched any longer. In reality, automation test maintenance is an ongoing process. Changes to the system require that tests be reviewed, updated, or even eliminated from the repository. Failing to do this will cause failed test runs and expand the time of the testing cycle.

Final Notes

Automation decisions should not be taken lightly. It's not suitable for all projects, especially those in their early, unstable stages. But for the projects and teams that are ready, it can be extremely useful and helpful to find defects faster and reduce testing cycle times. For it to be successful, an investment of time, money, and staff is required. As long as you go into it with your eyes open and with proper expectations, deciding to automate will be a major boon to your QA department.

QA Mentor's recommendation for teams lacking in-house automation expertise is to bring in automation consultants to help find the right tool, provide appropriate training, and assist in the automation process in general.

About the Author

The author, Ruslan Desyatnikov, is the CEO & President of QA Mentor. After working as the Global Head of Testing at Citi, Ruslan created QA Mentor to fill the gap he has witnessed in QA service providers during his near 20 years in QA. With Ruslan's guidance, unique services and methodologies were developed at QA Mentor to aid clients in their QA difficulties while still offering a high ROI. Ruslan offers monthly seminars aimed at imparting his extensive testing knowledge that can be applied to start-ups as well as large companies.

About QA Mentor

QA Mentor is independent software testing company and top provider of 37 different Quality Assurance testing types, with 24 different services – several of which are not found at any other company. With a global presence in 8 different countries, a staff of 100, and a pool of over 1500 crowdsourced testers, QA Mentor uses an effective and unique combination of onshore and offshore resources to keep costs low for our clients. QA Mentor aims to bring quality to the forefront of software development and become the central figure in Quality Assurance with an active QA community full of engaging discussions and resources.



QA Mentor, Inc. United States

**1441 Broadway, 3rd Floor,
New York, NY 10018.**

Toll Free: 1-800-622-2602

Manhattan Office: 1-212-960-3812

Brooklyn Office: 1-718-606-7086

New Jersey Office: 1-201-918-4616

Email: support@qamentor.com

Web: www.qamentor.com

All content & information presented in the White Paper is the exclusive property of QA Mentor, Inc. The content & information contained here is correct at the time of publishing. No material from white paper may be copied, modified, reproduced, republished, uploaded, transmitted, posted or distributed in any form without prior written permission from QA Mentor. Unauthorized use of the content & information appearing here may violate copyright, trademark and other applicable laws, and could result in legal actions.

Copyright © 2014 QA Mentor, Inc.