

# **HOW AGILITY SHAPES QA**

**A GUIDE TO MANAGING  
A COMPANY'S AGILE  
WEB DEVELOPMENT**

# HOW AGILITY SHAPES QA

A Guide to Managing a Company's Agile Web Development

Within the last two decades the approach to software development has changed drastically. Agile development has gotten much more popular within the software world ever since agile methods were first introduced over 15 years ago.

66% of development teams are now using agile methods in their software development process to create new features and almost three quarters of companies use agile methods in project management. The wide adaptation of these methods doesn't just come without a reason. According to PwC, agile projects are 28% more successful than traditional project using the waterfall method.

But how does this new ubiquitous methodology change the way we approach testing and quality assurance (QA)? What new challenges arise for test managers and how can they overcome these to maintain or even enhance their profession? This white paper will show the impact that agility has on software development and will examine chal-

lenges QA Managers face in an agile development environment. It will give you multiple approaches to how you can successfully integrate quality and security into agile methods.

## TABLE OF CONTENTS

1. Why is there a QA Management?
2. What are the tasks of QA Managers?
3. Why is Security so important?
4. How Agile Development changes QA
5. Test-Driven Development (TDD)
6. Managing Processes
7. Managing People
8. Building a functioning DevSecOps Pipeline
9. Conclusion

## EXECUTIVE SUMMARY

The historic purpose of QA management has been to ensure application stability without flaws. Therefore, the main task of a QA manager has been the inspection of the code itself. An integral part of that inspection is to enhance application security, leading to reduced risk, higher productivity, and higher revenue.

Agile principles, the autonomy of software development teams, and much more frequent, automated releases led to a change in the QA department tasks. Especially the Test-Driven Development practice extends this role change.

The new role of the QA department includes the essential role of managing processes and people, especially the toolchains and included test processes.

A crucial task for QA managers in this role is the establishment of a functioning DevSecOps pipeline to ensure the security of all software projects that are being delivered.

## 1. WHY IS THERE A QA MANAGEMENT?

In a company dealing with several software projects there are many development teams with developers on different levels of expertise working on different tasks. To assure a high quality product for your customers, it is necessary to set the same quality standards for all these teams. Because this can not be dealt with by the teams themselves, a separate entity, the QA department, needs to be established.

People make mistakes. They always have and they always will. Even the brightest minds of us are not perfect so it is always better to have a second view on important projects or features. Additionally, the pressure to produce new features, doesn't allow developers to sufficiently check their code quality after finishing a task. This is one of the reasons why extreme programming (XP) suggests the use of pair programming to have the written code checked at all times. In development teams where XP is not used, the QA department is taking over that function.

Since most software developers only look at their own code, feature or part of a web application, it is beneficial to have someone (or even a team) to look at the bigger picture. A problem in one part can lead to a problem in another or stand against the need of a customer that has just recently come up. A test manager ensures that the entire application is working without any flaws. Reaching a certain size, a company may also need formal testing processes, which is why many organizations create testing teams whose task is also to automate testing to make it as agile as the development process itself. The Puppet State of DevOps Report shows that highly-evolved organizations are 24 times more likely to always automate security policy configurations compared to the least evolved organizations, strengthening the point that high performing teams automate as much as possible.

## 2. WHAT ARE THE TASKS OF A QA MANAGER?

Quite simply: QA Managers test software or manage a department of testers.

However, the job contains much more than scrolling through lines of code all day. QA Managers need to develop concepts for new software tests that fit best for their development team. In the case of a bigger organization they might need to manage a team of QA testers that need to be trained, mentored and supervised. It is beneficial that a QA Manager knows most scripts, tools and techniques that software engineers use. However, this is not necessarily mandatory, if a QA Manager has a network of people that can fix any sort of issue.

In large organizations, the task of testing is not manually manageable, with companies releasing code multiple times a day across hundreds of applications. At this size of an organization, the task of the QA manager shifts more towards integrating QA policies as much as possible in the automated deployment process.

Additionally to the simple testing of code, a QA Manager gives valuable feedback and advice to software developers to improve their secure coding practices and reduce the probability of future problems arising.

### 3. WHY IS SECURITY SO IMPORTANT?

Test managers do not only test web applications for the functionality but also for possible vulnerabilities that can lead to security breaches.

Many managers still think that functionality is more important than security, but neglect the positive aspects that web application security has on the company, such as:

#### HIGHER PRODUCTIVITY

Securing a company's web development is not only a burden, but can actually enhance productivity. If a company has a well working testing department or uses a web application security solution, software developers can concentrate on creating new features that successfully attract and retain customers.

#### HIGHER REVENUE

With the many data breaches happening nowadays, customers are more and more looking for a web application that is secured. Companies with a high level of web application security can leverage that security to make it a sustainable competitive advantage over their competition, especially when dealing with very sensitive data.

Enhances collaboration between teams that don't usually work together	54%
Increases the level of software quality in organisations	52%
Results in increased customer satisfaction	49%
Shortens time to market	43%
Reduces cost of development	42%

Source: Techbeacon

### 4. HOW AGILE DEVELOPMENT CHANGES QA

The high percentage of companies using agile methods implies that there is a need for everyone to adapt to the new methodology since it affects every part of an organization.

But how is QA management and testing influenced by the new way software development teams work? What challenges arise for QA Managers and what new opportunities surface?

Before getting into the changes that come with agility, let's have a look at what agile actually means:

The individual principles of agile software development were formally brought to us under the term agile in 2001, when the Agile Alliance published the agile manifesto consisting of 12 principles to be followed.

Agile development is a method where development teams incrementally create software in shorter cycles to reach a finished product in a shorter period of time instead of creating a product that is finished, tested and released only in the end (as done in the traditional waterfall method).

During these development cycles, development teams are given much more autonomy and individual authority in general is decreased since every team-member also acts as a leader.

The main advantages of agile development include a shorter time to market, a higher collaboration within the development team and a product that is more customer centric than in any other method.



## HOW DOES THIS CHANGE THE WAY QA WORKS?

The higher autonomy of developers also gives them the obligation to test their own code on any flaws it may have. Because software developers are now empowered to create and test every part of their code, there is no real testing department needed anymore. This additional iteration would take up too much time to have a place in agile development.

These changes and the general reduction of hierarchy might scare QA Managers that there may not be a spot for them in the organization. However, QA is no longer only about testing software at the end of a process. It's about testing it after every iteration hand-in-hand with the short development cycles. The next chapter expands this idea and introduces the testdriven development approach, in which tests are already written before the code.

The shorter development cycles that come with agile methodology also have an impact on processes and tools used by the development team and test managers. How these are being changed will be shown in chapter 6.

Additionally, agile development has a higher need for collaboration and communication. How this changes the role of QA Managers will be examined in Chapter 7.

## 5. TEST-DRIVEN DEVELOPMENT (TDD)

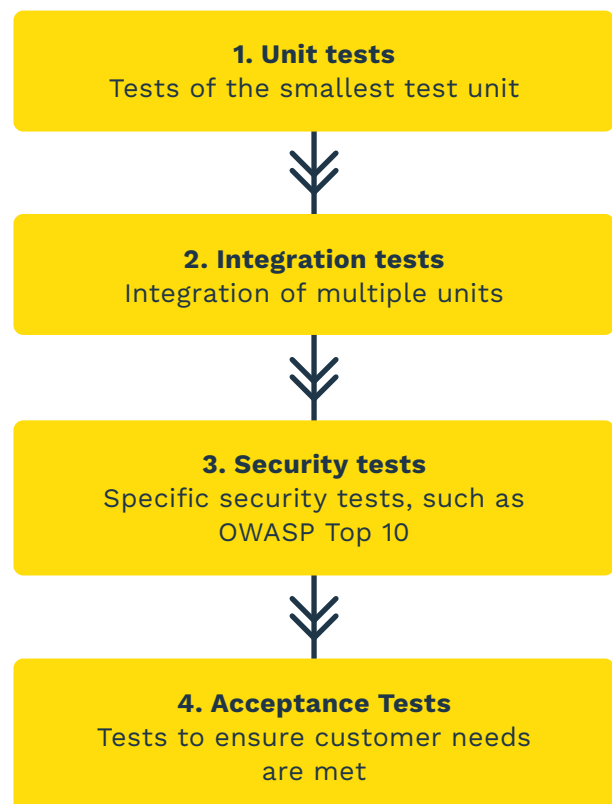
In parallel to the development of agile principles, extreme programming was first mentioned in 1999 by Kent Beck. XP is a methodology that delivers small, frequent changes to improve code quality and focus on customer requirements.

The name stems from taking known software development practices and taking them to the extreme. For example, the principle of frequent code reviews is taken to the extreme by introducing pair programming. This practice has one person coding and one person reviewing code at the same time.

Another outcome of taking practices to the extreme is the test-driven development approach. Instead of introducing tests only after the code has been written, XP turns the principle on its head by promoting to write tests before the code itself. This forces developers to think in-depth about the customer requirement and what test would prove this requirement to be done. It also forces developers to not over-engineer features that might be accomplished with less work.

## THIS PRACTICE SUPPORTS THE MISSION OF THE QA MANAGER ON TWO LEVELS:

First, the developers write tests specific to customer requirements. As long as quality is important to the customer - which it always should be - the developers also write tests according to these quality requirements. Therefore, some parts of the QA testing tasks are performed by the developers themselves.



Second, this framework changes the state of mind of developers from writing code first to writing code that satisfies tests. This heightened quality awareness makes it easier for the QA department to be heard. Additionally, developers welcome the support from the QA department to share best practices for testing frameworks.

In the end, it is also important to understand the hierarchy of test cases and separate between the different levels of testing (see Figure 1).

Since we've now showed how software development and QA is changing through the wide adoption of agile principles, it is time to look at how the tasks of a QA manager can be adapted to ensure a high standard of quality for all development projects.

## 6. MANAGING PROCESSES

Since the software is tested with every release, the testing at the end of the development process becomes less important. Therefore, test managers find themselves in new tasks within the development team: supporting the test of every release. The high competency of a QA manager can be used as a guide for all processes and tools that are used to assure the sufficient quality of the software product in the end.

Since development cycles are becoming way shorter, quality testing moves further forward and may even start before the actual development. This is why it is important for QA Managers to make sure that every developer tests their code sufficiently before releasing it. Code testing is a time consuming task and may often be seen as unnecessary since developers believe in their own abilities to develop high quality software.

An efficient way around this task is found in the automation of (security) testing. If every iteration of the software product is being automatically tested after a deployment to the test system, test managers can make sure that a new cycle is being started on a sufficient level of quality and security.

Using the Test Driven Development approach can move testing even further forward to ensure that there is a test for every component that is being created. End-to-end solutions like Testifi.io can be remarkably beneficial to coordinate the new process of testing. These solutions help organizations create a SecDevOps Pipeline, where testing and security is as important as functionality (see Figure 2).

Additionally a test manager's role can shift more towards coordinating these processes and being a project leader for the implementation of an automated vulnerability testing tool. Since they know best, what kind of vulnerabilities are most common and what are the most difficult to re-mediate, a QA Manager knows, what such a tool would have to cover to ensure all critical vulnerabilities are found. In a managing role, the QA manager could lead the development teams testing efforts and decide on who remediates certain vulnerabilities. In the best case, the QA manager may serve as a Key User to help every other user with issues regarding the use and understanding of the application.

Depending on the size of the organisation and the development team, such a vulnerability assessment tool can be rather expensive. That is why an additional role for QA managers can be found in the management of funds for software testing. According to the World Quality Report the QA-budget should be a quarter of the development department's overall budget. Justifying such funds in front of C-level executives can be challenging, which is why you should carefully calculate the ROI of your web application security tool.



## 7. MANAGING PEOPLE

The change to agile is also a change towards more collaboration and communication within the development team. Less documentation and authority may seem daunting for test managers at first glance but can be seen as opportunities for new tasks. In an agile development environment the role of a test manager shifts from directing towards mentoring the software developers assigned to a certain project.

As mentioned earlier, most software engineers are highly educated individuals that don't necessarily see any flaws in their code so it is good to have a qualified test manager as an anchor point for all software developers. That manager does not only help with issues but is also someone to look at the overall impact of a quality or security flaw. Since developers primarily only see their own code, a test manager may see the impact one vulnerability can have on the entire application.

In order to reach a sufficient level of security, a test manager should also be responsible for the creation and training of the development team. By choosing qualified developers and training them on secure coding practices, QA Managers can mitigate the risk of vulnerabilities or other flaws arising in a later stage. Through this training QA Managers can empower developers to write flawless code and therefore move their testing forward.

Through trainings and workshops, test managers can eventually create a secure culture within the development team or even the entire organization. Within this culture, developers can create and test their code autonomously - just like it is supposed to be. Still, QA managers should know when to intervene, if they see that after a certain iteration the product's quality is in jeopardy.

## 8. BUILDING A FUNCTIONING DEVSECOPS PIPELINE

One of the major benefits of agile developments are the shorter development cycles that lead to a shorter time to market. That time would be sacrificed if testing happened at the end of every cycle. Because of that, testing needs to move further ahead and be done after every iteration. Only through this continuous security and a Test-Driven Development approach can a test manager turn DevOps into DevSecOps and make sure that users are given a secured application at any time.

Since this kind of testing is too time consuming and costly to do it manually, QA managers should implement automated security into their CI/CD processes to create a thorough DevSecOps Pipeline. This Pipeline can be enhanced with a sufficient vulnerability assessment tool, that tests an application or API weekly, monthly or after every deployment to a test system. That way, vulnerabilities are shown before the final version is presented to a customer.

With regards to a vulnerability assessment tool and the promotion of the new approach within the development team QA Managers can be given the role of a project leader for the implementation of all DevSecOps processes. Additionally, QA managers can act as a key user once a solution is in place and a mentor for developers, whenever a vulnerability shows up in their code.

## 9. CONCLUSION

The change to agility is difficult one for all companies, managers and developers. Especially the QA department needs to adapt to new challenges to ensure a sufficient overall quality of all software projects.

In this adaptation, QA managers need to shift their focus more towards managing the processes that developers follow, thus enabling individuals to autonomously create secure software without a special department testing it afterwards.

To effectively collaborate across autonomous software development teams, the QA management also needs to monitor, communicate, and train effectively across a whole organization - all while focusing on and enabling the individual software developer.

Automation of security testing through innovative vulnerability management solutions can help QA testers overcome the challenges arising.

The changed role for QA departments is well worth the effort, because at the end of the steep hill to climb, the benefits outweigh the disadvantages:

- + Every code release is tested automatically
- + Change from a manual security check from a specific point in time to an automated, continuous testing process

- + Higher productivity for the QA department through tool integration
- + Autonomous teams are sensitive to security flaws through test-driven development
- + Toolchains and processes support self-remediation

By creating a new way of testing and with the help of a thorough solution like Testifi.io, can QA Managers focus on guiding developers to ensure that testing is being thought of from the very beginning. With a sufficient tool at hand, software developers are not only secured but also empowered to concentrate on what they do best:

**creating amazing features.**

## ABOUT CRASHTEST SECURITY

Crashtest Security is a Munich-based IT security company.

As an innovator of cyber security solutions for web applications, it develops automated solutions for vulnerability analysis.

Based on artificial intelligence, vulnerabilities are detected, protection against hacker attacks is increased and transparency for companies, users and developers is created.

Visit our website for more:

[WWW.CRASHTEST-SECURITY.COM](http://WWW.CRASHTEST-SECURITY.COM)

### CRASHTEST SECURITY GMBH

Leopoldstraße 21  
80802 Munich  
+49 (0)89 215 41 665

[info@crashtest-security.com](mailto:info@crashtest-security.com)



## SOURCES

1. <https://techbeacon.com/app-dev-testing/survey-agile-new-norm>
2. <https://www.pmi.org/-/media/pmi/documents/public/pdf/learning/thought-leadership/pulse/pulse-of-the-profession-2017.pdf>
3. <https://www.pwc.com/gx/en/actuarial-insurance-services/assets/agile-project-delivery-confidence.pdf>
4. <https://www.agilealliance.org/glossary/xp/>
5. <https://puppet.com/resources/report/2021-state-of-devops-report>
6. <https://databreachcalculator.mybluemix.net>
7. <http://agilemanifesto.org/>
8. <https://techbeacon.com/app-dev-testing/survey-agile-new-norm>
9. Beck, K. (1999). Extreme Programming Explained: Embrace Change. Addison-Wesley. ISBN 978-0-321-27865-4.
10. <https://testifi.io/home-2/>
11. <https://www.microfocus.com/en-us/assets/application-delivery-management/world-quality-report-2021-2022>
12. <https://crashtest-security.com/web-application-security-positive-roi/>