

3

Why Software Testing Is Key to DevOps

4

Test Everywhere: A Journey into DevOps and Continuous Testing

7

Shifting Your Testing: When to Switch Gears

11

Continuous Testing, Shifting Left, and Test Automation: Getting It Straight

12

Demystifying DevOps: A Day in the Life of a DevOps Tester

13

Why You Need Continuous Testing in DevOps

17

Continuous Delivery Is Not a Pipeline

19

5 Key Factors to Achieve Agile Testing in DevOps

23

Key Enablers for Continuous Testing

25

Insight from the Industry

26

Additional Resources

In order to fully realize the potential of DevOps and a team's ability to deliver higher-quality software faster and more securely, testing must be done continuously throughout the software development lifecycle. But continuous testing isn't just about automation and tools; there are cultural and process shifts that must happen, and everyone must be on the same page. This eGuide collects methodologies, processes, and tips to help your organization venture into continuous testing in DevOps.

In this Testing in DevOps eGuide

Why Software Testing Is Key to DevOps

One of the major reasons organizations adopt DevOps practices is to accelerate delivery of software to production. However, many fail to include quality components in their practices. Continuous deployment without quality is just delivering continuous bugs. Here's why software testing is an essential part of DevOps.

Test Everywhere: A Journey into DevOps and Continuous Testing

A move to DevOps creates an opportunity to shift the testing process to the left. But what if you went further? DevOps supports continuous testing, so you can advocate for a constant focus on quality, with testing permeating the entire software development process. Here's how you can actually have a faster testing process when the software is tested throughout the lifecycle, by developers, testers, and automation alike.

Shifting Your Testing: When to Switch Gears

Shifting your testing either left or right can meet different needs and improve different aspects. How do you know whether to make a change? Let your test cycles be your guide. Just like when driving a car with a manual transmission, if the engine starts to whine or you're afraid you're about to stall out, switching gears may be just what you need.

Continuous Testing, Shifting Left, and Test Automation: Getting It Straight

Continuous testing can help you achieve the optimal balance between speed and risk and deliver high-quality products faster. But what exactly does continuous testing entail? Is it just shifting testing left in a DevOps environment? And where does automation fit in? Here's a breakdown of all these testing concepts.

Demystifying DevOps: A Day in the Life of a DevOps Tester

The idea of working as a test specialist on a team using DevOps can be intimidating. There are at least two technology stacks, containerization and continuous integration, that you need to be familiar with. But few people need to be able to start from scratch. Here's what a normal day of testing in DevOps looks like.

Why You Need Continuous Testing in DevOps

DevOps is more than adopting the right set of tools; it's a cultural shift that incorporates testing at each stage of the agile project lifecycle. Continuous testing is key to unlocking this culture change because it weaves testing activities into every part of the software design, development, and deployment processes, which helps everyone involved communicate more, collaborate better, and innovate faster.

Continuous Delivery Is Not a Pipeline

Pretty much everything you hear about DevOps mentions "the pipeline." Continuous delivery is not really about the pipeline, however. Continuous delivery is about two things: testing strategy and branching strategy. The pipeline is important; it is an integral part of DevOps. However, the central element is the practice of testing continually using automated tests.

5 Key Factors to Achieve Agile Testing in DevOps

Part of the path to DevOps requires adoption of agile methodologies. What does it mean for testing when you switch from the traditional waterfall model, with a few long release cycles per year, to the agile model, with changes occurring every two weeks? Here are five key factors to achieve the agile software testing necessary in DevOps.

Key Enablers for Continuous Testing

Continuous testing means testing before, during, and after each software change is made. Testers have long advocated for this, but DevOps has made it more popular by pushing for rapid feedback and shifting testing left in the lifecycle. Here are three practices your company should embrace to enable continuous testing.

Insight from the Industry

Additional Resources

Why Software Testing Is Key to DevOps

By Alan Crouch

One of the major reasons organizations adopt DevOps practices is to accelerate delivery of software to production. This includes deploying more frequently and reducing lead time.

However, many organizations fail to include quality components in their practices. This leads to organizations delivering code faster, but unfortunately, that code is just poor. Continuous deployment without quality is just delivering continuous bugs to your customers.

If this sounds like your organization, software testing may be the missing component to your DevOps program. Top performing DevOps organizations, like Netflix, Amazon, and Etsy, utilize automated regression, performance, load, and security testing to ensure software quality is built into their DevOps pipelines and ensured by being forced to be run on every build. For Netflix, this allows software to be committed, tested, and fully deployed to production within sixteen minutes!

If your organization isn't as large or doesn't require as rapid deployments as those companies, automated testing in your DevOps pipelines still provides significant benefits. Even a small amount of automated regression testing can ensure basic tests are always run on a build. Typically, organizations start by building an automated smoke test. This smoke test provides a sanity check on a developer's code. This can not only reduce some of the overall manual testing effort by the testing team, but also ensure effort isn't being expended on builds that don't meet minimal quality standards.

More robust automated testing suites can result in even less manual testing, in addition to better-focused exploratory testing efforts



into high-risk areas of the application, including interfaces, misuse cases, and the most important assets to your system. While not every test can (or should) be automated, spend your critical resources wisely by only focusing manual efforts on the things that really require their attention.

Most organizations today use some level of automation when looking at performance, load, and security. Leveraging those existing capabilities into your DevOps pipelines ensures that your deployments are not held back by independent groups late in the software lifecycle and they aren't an afterthought when the application is already in production. This has immediate benefits, such as reducing cyber security impacts on your applications and their data by ensuring critical vulnerabilities are identified earlier in your software development lifecycle and aren't deployed to production in error.

Pulling software testing into your DevOps practices doesn't have to be difficult. You can start by bringing your existing testers and security engineers into your planning sessions. Next, ensure each phase of your pipeline has a quality gate and software quality criteria that should be met in order to move to the next phase of your pipeline. Last, identify gaps between your quality goals and reality, and ensure you prioritize efforts into the activities that provide the greatest return on investment—those that are either run with the highest frequency, reduce the greatest software quality risk, or reduce the biggest bottlenecks.

These simple steps will lead to not just delivering code faster, but delivering better code faster.

3

Why Software Testing Is Key to DevOps

4

Test Everywhere: A Journey into DevOps and Continuous Testing

7

Shifting Your Testing: When to Switch Gears

11

Continuous Testing, Shifting Left, and Test Automation: Getting It Straight

12

Demystifying DevOps: A Day in the Life of a DevOps Tester

13

Why You Need Continuous Testing in DevOps

17

Continuous Delivery Is Not a Pipeline

19

5 Key Factors to Achieve Agile Testing in DevOps

23

Key Enablers for Continuous Testing

25

Insight from the Industry

26

Additional Resources

Insight from the Industry



“In shifting left, teams see huge benefits such as finding bugs much earlier, sometimes even before code is ever written. This results in a higher quality of code and a far less expensive way in achieving such.”

—Angie Jones

[▶ READ THE FULL INTERVIEW](#)



“We’re not going faster, we’re not coding any faster, we’re not typing any faster. We’re learning good ways to slice these features down, so we can deliver small increments.”

—Lisa Crispin

[▶ LISTEN TO THE FULL INTERVIEW](#)

GAIN MORE INSIGHT WITH THESE VIDEO INTERVIEWS



DevOps Misconceptions and Testing Confidence: An Interview with Hans Buwalda



Understanding the Role of QA in DevOps: An Interview with Gene Gotimer



Shift Your Mindset with Your Testing: An Interview with Isabel Evans



Deciding if You’re Technically Ready for DevOps: An Interview with Sunil Sehgal



Continuous Testing and Open Source Integration: An Interview with Alex Martins

3

Why Software Testing Is Key to DevOps

4

Test Everywhere: A Journey into DevOps and Continuous Testing

7

Shifting Your Testing: When to Switch Gears

11

Continuous Testing, Shifting Left, and Test Automation: Getting It Straight

12

Demystifying DevOps: A Day in the Life of a DevOps Tester

13

Why You Need Continuous Testing in DevOps

17

Continuous Delivery Is Not a Pipeline

19

5 Key Factors to Achieve Agile Testing in DevOps

23

Key Enablers for Continuous Testing

25

Insight from the Industry

26

Additional Resources