



20 STATE OF TEST 20 AUTOMATION

2020 State of Test Automation Report

2020 is the year of test automation. But a lot of teams aren't quite there yet. Test automation is critical to the DevOps pipeline, but its rate of adoption varies. And many teams still struggle to achieve successful test automation.

To better understand the state of testing today, Perforce and Gatepoint Research partnered up to survey leading digital enterprises across industries.

If your test automation isn't where you want it to be yet, don't worry. We outline key recommendations from the experts to get you to the next level. Just keep reading to find out more.



KEY FINDINGS

Regardless of the past decade's software advancements, test automation is still a huge **pain and obstacle** for DevOps.

The average test automation coverage is less than **50%**. To mature DevOps, organizations must strive toward **75-80%** and beyond.

Organizations lack continuous testing strategy that **involves entire DevOps teams**.

Continuous testing and shifting left heavily depend on the **proper matching of individuals' skillsets**. Unfortunately, this is not properly implemented.

Test automation within the software build cycle and pipeline isn't effective enough, doesn't happen in time, and **suffers from flakiness**.

The Who's Who of Testing

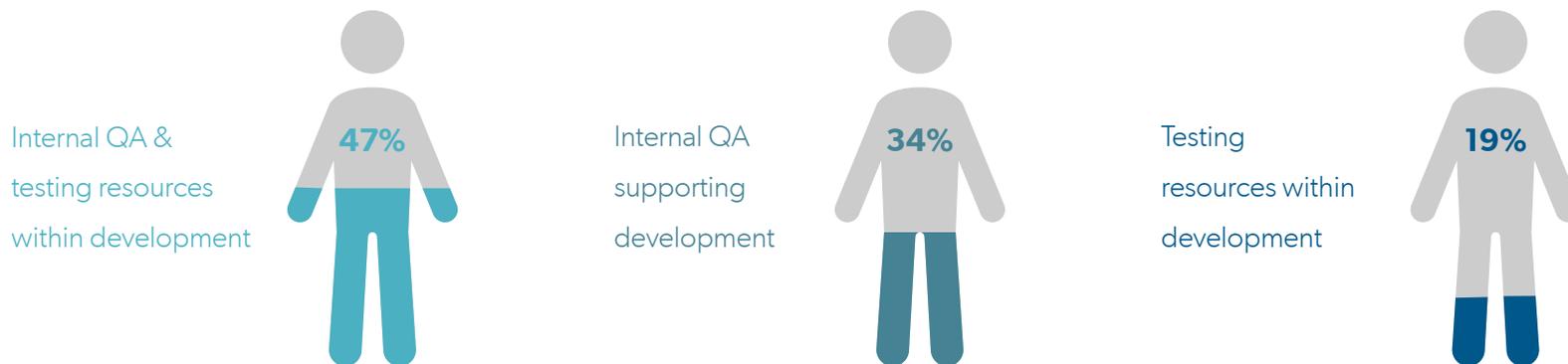
We wanted to know how testing is structured within organizations. For the vast majority of companies, testing is done by a mixture of both QA and dev.

Related Reading: [The QA Tester's Guide to Thriving in an Automated Testing World](#)

The organizations that are more successful at test automation rely on internal QA functions. While these teams face skillset challenges for creating test scripts, they are investing in commercial tools to ensure their success.



Who is responsible for testing in your organization?



WHEN COMPANIES RELY ON QA FOR TESTING...

- They struggle with test instability/flakiness & false negatives.
- They use commercial tools more than development teams.
- They are more automated than fragmented teams.

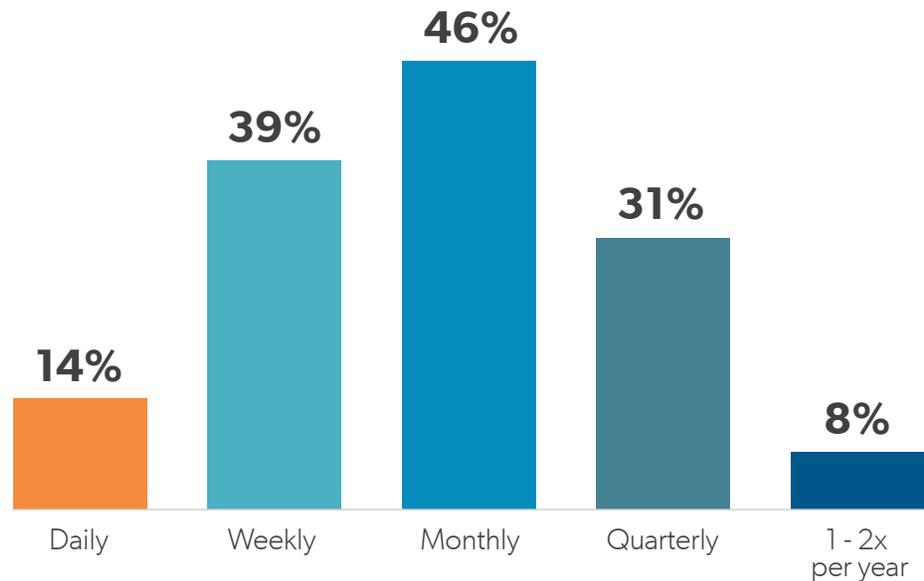
WHEN COMPANIES RELY ON DEVELOPMENT FOR TESTING...

- Open source tools & frameworks are most commonly used.
- 56% are automating less than a quarter of their test cases.
- Very few rely on dev for testing — the **shift left movement** is not taking hold.

Most Teams Release Weekly or Quarterly

Only **14%** of those surveyed release daily.

What is your typical release cycle?



HOW TO RELEASE MORE

The keys to releasing more often depends on the entire software development process from coding, building, integrating, testing, and deploying the code to production. Whenever tasks are broken, or not automated, the **entire process slows down** and delays the release.



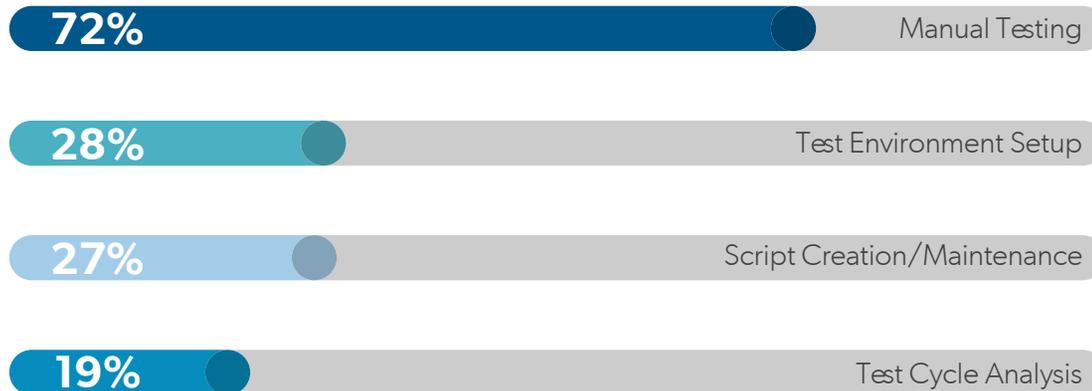
Not just testing should be automated. Everything in the pipeline, before and after, should be automated as well in order to release more frequently.

While in this research we focus on test automation, more than just that needs to be automated. This needs to be a DevOps team priority and not a single team's goal. Making the connection between the different "players" in the pipeline comes through process and technology, and this requires choices based on priorities to be made by management.

Related Reading: [Top 5 Things Slowing Down Your Testing Life Cycle](#)

Manual Testing Is the Leading Cause of Longer Test Cycles

What are the most time-consuming activities within a test cycle?



Through proper strategy and prioritization, time-consuming manual testing can be minimized.

MANUAL TESTING IS AVOIDABLE & SOLVABLE

1. Addressing skillset needs

Better matching of test automation tools needs to be done so business testers and noncoders can feel comfortable enough to contribute test automation to the pipeline. In addition, coders must reduce test flakiness through proper coding guidelines, code reviews, better test automation maintenance and certification, and more.

2. Time to automate

Developers need to deliver software builds as early as possible and not wait until everything is fully done to release builds to testers. This can happen through TDD (test driven development) methods, early access builds that aim to enable test automation foundation building by test teams, and more.

Related Reading: [Manual Testing vs. Automated Testing vs. Continuous Testing](#)

Only 25% Leverage Commercial Labs Today

However, 46% of companies are looking to invest in commercial testing tools in 2020.

Teams using commercial lab vendors....

- Are more likely to release daily.
- Are less likely to have escaped defects.
- Deal with less test automation flakiness.

HOW DO COMMERCIAL TOOLS HELP?

Commercial lab vendors help connect open source test frameworks to mobile and web platforms. Lab vendors facilitate a stable test automation and execution environment. This promotes greater **test coverage** for automation. Teams also encounter less “noise” that is associated with DIY solutions.

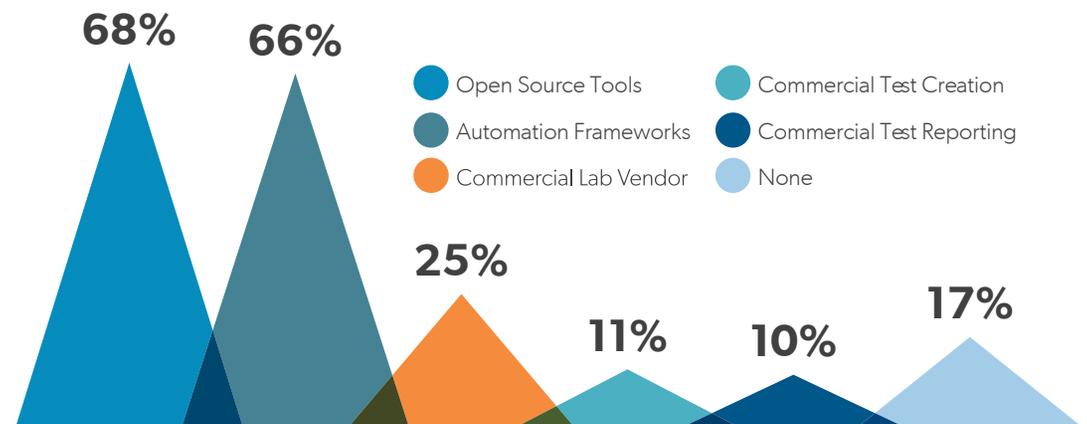
Many commercial tools like Perfecto also provide additional productivity features like **test reporting**, **test creation**, scalability, app debugging, and more.

Related Reading: [How Testing Tools Have Evolved for the Better](#)

Open source tools and automation frameworks are a necessary part of any automation project, but they do have their limitations.



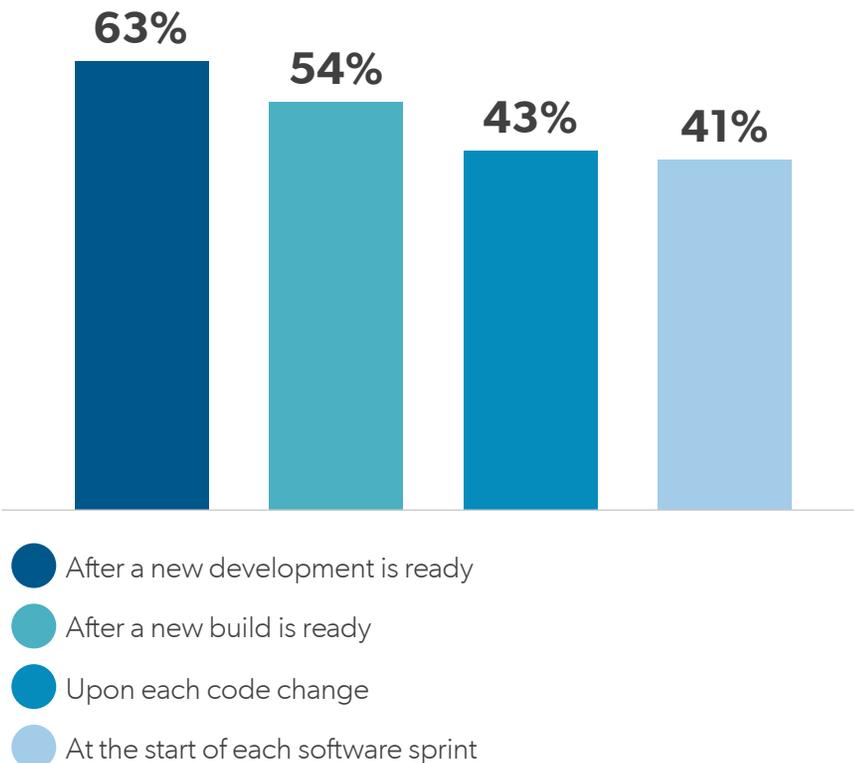
What test automation tools are you leveraging today?



Where Testing Fits Into the Cycle

Too many are waiting to test until after a new dev build is ready. Organizations need to test throughout the process rather than at the end.

Where does testing fit into your software development lifecycle?



To mature DevOps, most checkpoints must be checked automatically. The more human intervention there is, the slower the process will be. Automating such checkpoints does not mean these gates are being put at risk. Rather, it means teams found the right ways to automate the flow and approve it.

An example of a best practice can be something like this. Instead of test engineers waiting for a new build to start their test automation, developers can grant an early access build that has the right "tag" attached to it to allow test automation activities to begin.

Related Reading: [SDLC Tools For Testing: The Right One For Each Phase](#)



Mature DevOps requires automated checkpoints. Human intervention slows processes down.

Test Coverage Is a Top Challenge for Teams

Test flakiness is a close second, followed by test automaton creation.

OVERCOME TEST COVERAGE DIFFICULTIES

Test automation coverage includes two variables — the test scenarios themselves, and the platforms that these tests run on. Teams struggle with both for different reasons.

The low percentage of automated test scenarios is due to a mismatch between test automation tools and the persona's skillset.

The poor platform coverage of web and mobile permutations is attributed to two main reasons:

- Teams choosing DIY labs rather than commercial cloud-based solutions.
- Risk-based vs. data-driven coverage decisions.

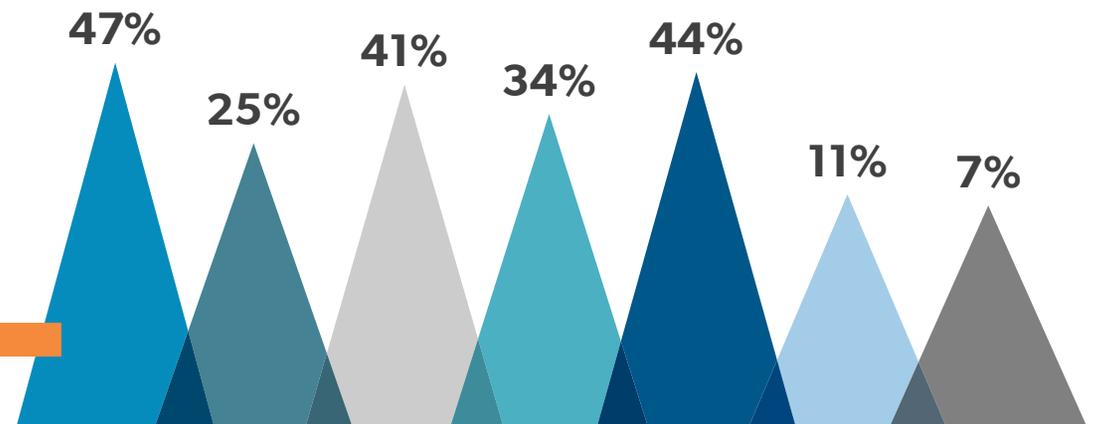
To overcome both challenges, teams must have a solid test coverage strategy in place that is based on analytics, tool matching, and other data-driven factors.

Related Reading: [5 Things to Consider When We Talk About Test Coverage](#)

What are your biggest testing challenges today?

- Achieving complete test automation coverage
- Lab platform setup & maintenance
- Lack of scripting/coding skills
- Analyze results quickly
- Test instability/false negative
- Testing not a priority
- Other

Cloud-based solutions and data-driven decisions can help teams boost test coverage.



Challenges Vary Depending on Automation Maturity

TOP CHALLENGES FOR LOW-MATURITY TEAMS

Teams automating **fewer than 10%** of their test cases reported struggling with:

- Escaped defects to production.
- Delayed software releases.
- Efficient test automation creation (scripting/coding).

Related Reading: [Test Automation Strategy For Beginners](#)

TOP CHALLENGES FOR HIGH-MATURITY TEAMS

On the other end of the spectrum, teams automating **over 50%** of their test cases reported struggling with:

- Achieving complete test automation scenario coverage.
- Web/mobile lab platform set up and maintenance.

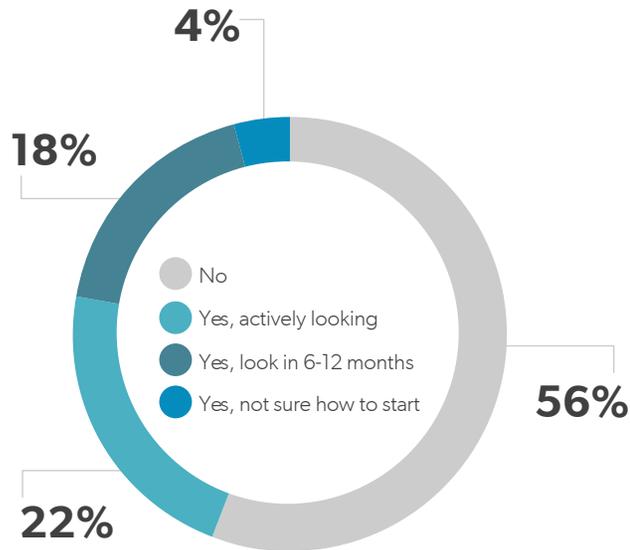
Related Reading: [How to Scale Your Test Automation Practices](#)

Automation Maturity Scale



It's the Year of Test Automation

Are you planning to invest in test automation solutions within the next year?



Scoping out test automation tools?

Download [The Buyer's Guide to Web & Mobile Test Automation Tools](#).

OF THOSE LOOKING FOR TEST AUTOMATION SOLUTIONS, THEIR TOP PRIORITIES ARE:

- Improving automated testing at scale.
- Integrating testing into their CI/CD pipeline.
- Reducing the amount of time regression testing takes.
- Better match their test automation framework to their individuals skillsets.

OF THOSE LOOKING FOR TEST AUTOMATION SOLUTIONS, THEIR TOP CHALLENGES ARE:

- Achieving complete test automation scenario coverage.
- Test instability/false negatives/flakiness.
- Lack of test automation creation (scripting/coding) skills.

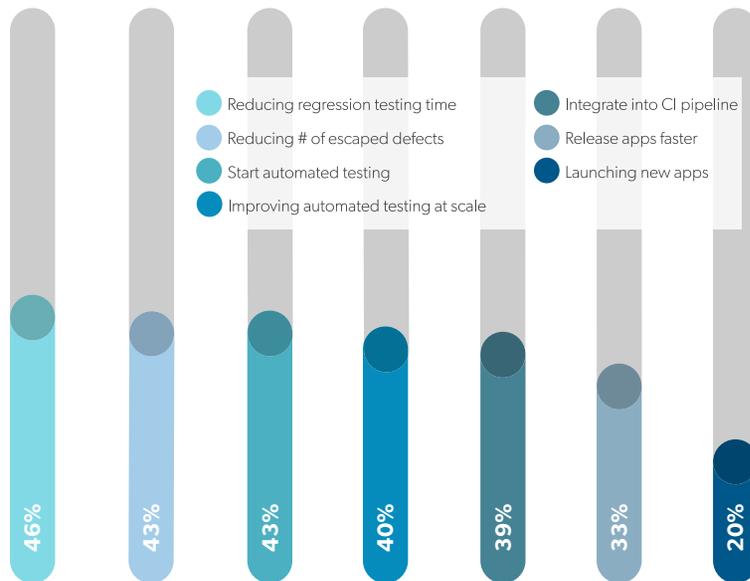


44% of those surveyed are looking to invest in automation solutions within the next year.

The Priorities of DevOps Teams

Top priorities? All of them. Everything is a priority. 6 of the 7 choices were ranked within a 13% range from most-to-least important.

Which of the following are your top priorities in the next 6 to 12 months?



HOW TO ACHIEVE THESE TOP PRIORITIES

Here's how you can tackle these top priorities in test automation.

Reducing the amount of time **regression testing** takes

Regression testing is the heaviest and most repetitive testing activity within a DevOps lifecycle, making it ideal for optimization. Teams

ought to constantly validate the scope of their regression suite, automate as much as possible, and eliminate flakiness and noise. Parallelization of testing can expedite the regression cycle and produce faster feedback.

Improving **automated testing at scale**

Automation at scale needs to be well thought out. Test automation must include high-value test scenarios from functional, non-functional, and unit testing. Once they are scoped, stable, and consistent, they need to fit into the iteration timelines. Such tests can scale either through burst mode, parallel execution, and/or dividing the suite throughout the pipeline phases so there is constant feedback from all tests as early as possible in the cycle.

Integrating testing into the **CI/CD pipeline**

Tests that enter the CI/CD must be fully certified in a sense of these being the "right" ones from a business flow perspective and coverage fit. They also need to be stable and with a history of detecting defects.

Moving from manual to **automated testing**

Manual tests can be automated by business testers, test automation engineers, and developers. Management needs to enable these practitioners to automate and properly maintain the tests in a way that fits their skillset as well as their business objectives. Developers can conduct unit testing. Test automation engineers can focus on structured functional and non-functional test creation. And business testers can automate the remaining tests scenarios as well as high-value exploratory tests.

Most See Mobile Testing as Critical

This comes as no surprise. In fact, mobile should be rated as even more important. More adults own smartphones than desktops or laptops. And mobile outranks desktop and tablets for market share.

The increased importance of mobile and mobile web apps leads to more complex test automation. Mobile apps have become more advanced and mobile phone releases/functionality are constantly-moving targets.

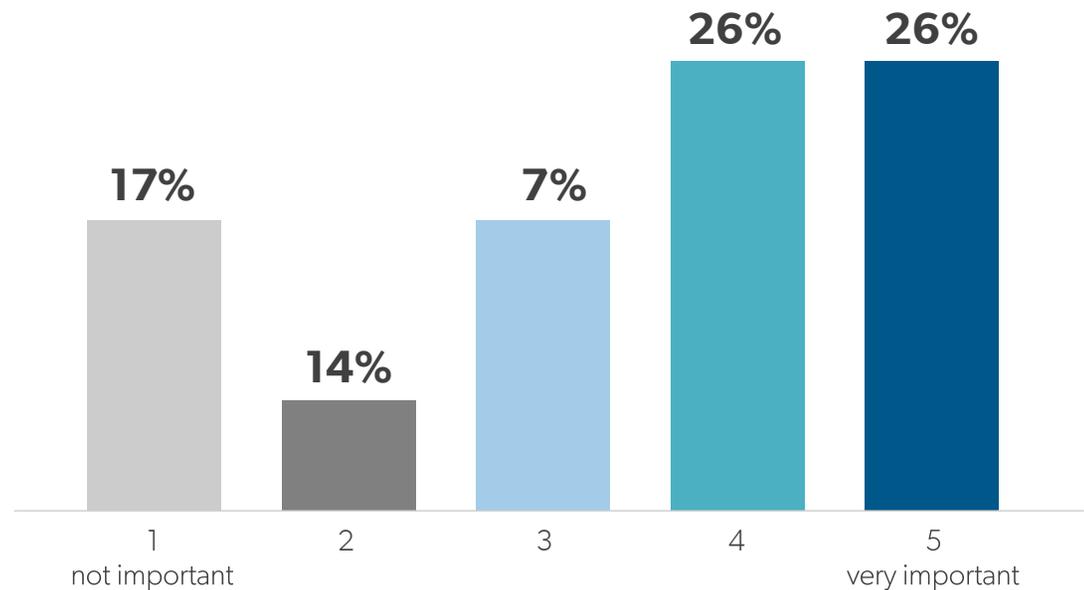
Related Reading: [Top 5 Automation Testing Challenges \(and How to Overcome Them\)](#)



Over half of the companies that we surveyed said that mobile testing is extremely important.



How important are mobile native and mobile web apps in your overall testing strategy?

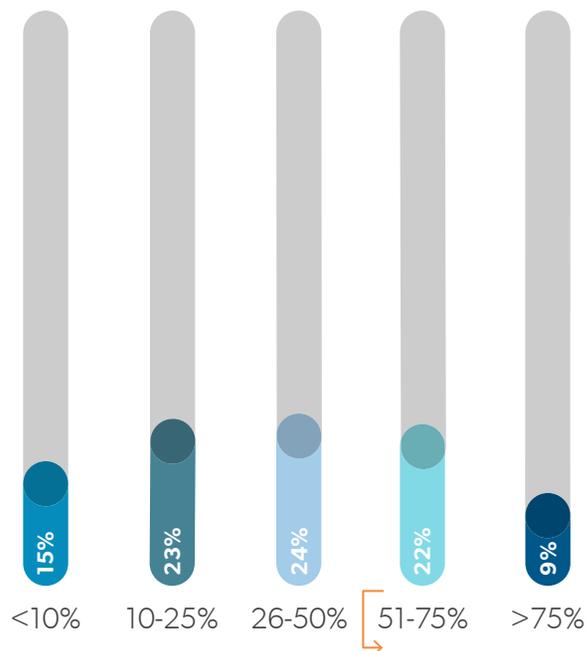


2020 Test Automation Rates

Automation rates are low, but many are looking to invest in solutions.

Looking into a new decade, teams must better strategize and reshape their testing infrastructure. We are experiencing a transformation in the testing landscape with better tools, codeless and [AI in software testing](#), and advanced reporting, all of which are being built to address report findings.

What percentage of your web and mobile test cases are automated today?



TEST AUTOMATION: GOOD, BETTER, CONTINUOUS

Good

Automating 50-60% of your test cases is considered good automation. As you are just starting with test automation creation, your focus is on stabilizing a test environment and overcoming challenges and test coverage blocks. This stage features a daily cycle with up to 1000 executions a day.

Better

Better automation is the next stage. As you mature your path, you start dealing with scalability, test visibility, and test noise. In this stage, you boost test coverage and receive meaningful daily feedback. To get here, teams need advanced test orchestration tools, environment control, and scaled reporting. Teams at this stage can execute 10,000 test cases a day on 50 platforms.



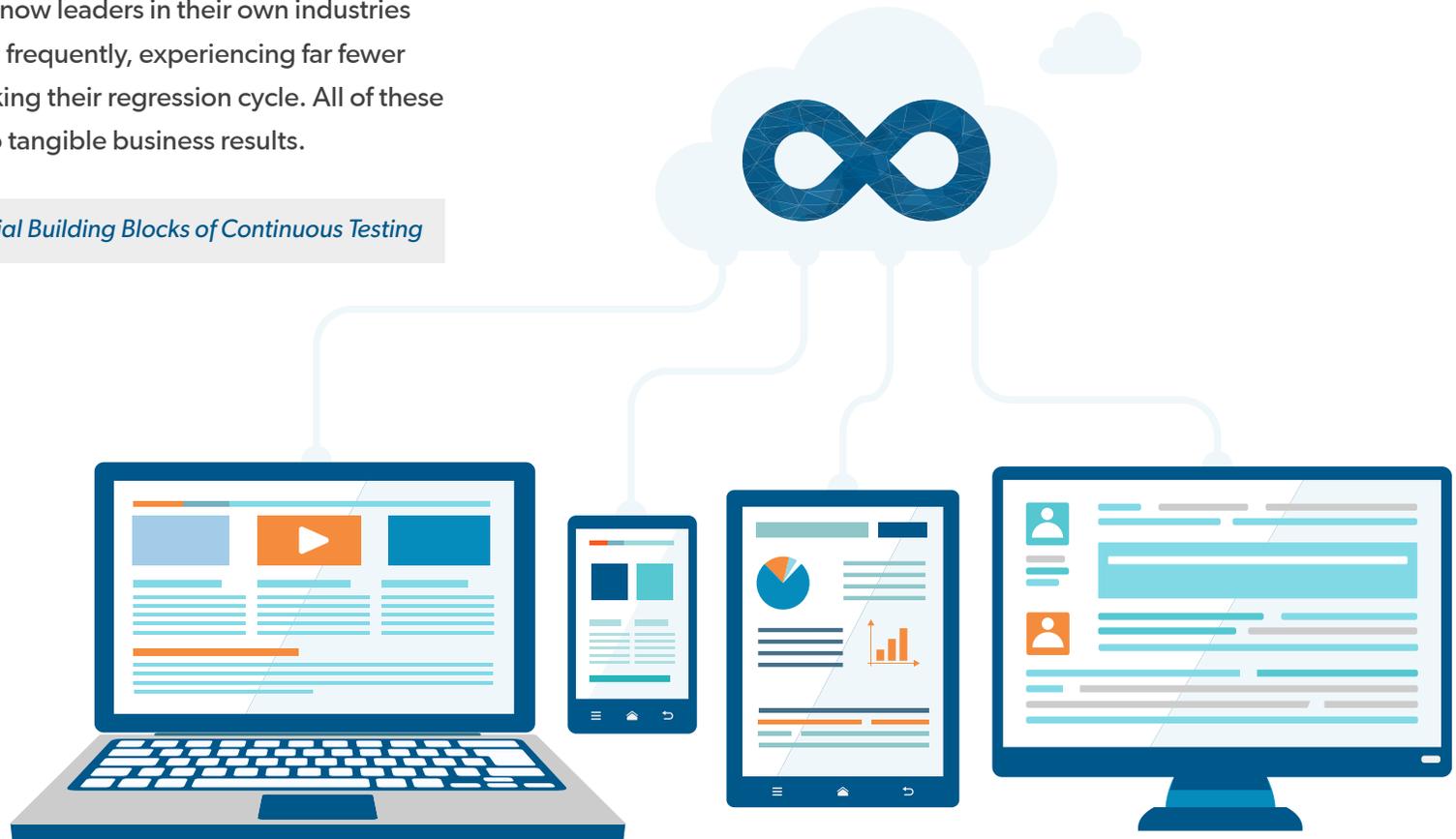
Only 9% of companies automate more than 75% of their web and mobile test cases.

Continuous

The best test automation automates all that is possible — about 95% of tests. This stage requires a very high scale of automation through cloud-based testing. A cloud-based platform connects teams with the devices they need to test and offers them the expansive scalability needed to achieve continuous testing.

When teams reach high automation and success rates, they enter the DevOps-friendly zone of testing. Perfecto's customers who have reached this stage are now leaders in their own industries — releasing faster and more frequently, experiencing far fewer escaped defects, and shrinking their regression cycle. All of these improvements translate into tangible business results.

Related Reading: [The 3 Essential Building Blocks of Continuous Testing](#)



Expert Recommendations

- Testing should not change software development processes, but rather fit into them. To succeed, teams must match frameworks with the team's objectives and skills.
- Quality visibility throughout the DevOps process is key for success. The more blind spots team have, the greater the delay in maturing automated processes will be.
- Build a strategy that allows teams to build a stable test automation foundation from the ground up. Implement a "certification" process for your test automation scripts. Put in place gate keepers and maintenance points to ensure continuous testing stability for your mobile and web apps.
- Validate and calibrate your test automation tool chain and strategy as your apps advance along with the market. You may have an opportunity to adopt newer solutions that can release roadblocks in your journey to DevOps.

Top Reasons Teams Fail in Test Automation

Chief Evangelist, Eran Kinsbruner, breaks down the results of the report and provides actionable insights for organizations looking to improve their test automation. [Watch now.](#)

Related Resources

TEST COVERAGE

- [Mobile & Web Test Coverage Index](#)
- [5 Things to Consider When We Talk About Test Coverage](#)

TEST AUTOMATION

- [Top 11 Challenges in Automated Testing & What to Do About Them](#)
- [5 Reasons Why You Shouldn't Stitch Together Testing in the DevOps Pipeline](#)
- [Top 5 Things Slowing Down Your Testing Life Cycle & Software Release Process](#)

TESTING TOOLS

- [What to Look for in Automation Testing Tools](#)
- [The Buyer's Guide to Web & Mobile Test Automation Tools](#)

