THE STATE OF TEST AUTOMATION REPORT

2022

SIMFORM

CONTENTS

Chapter 1

Introduction & Summary

Why we created this report?

In my extensive career as a VP of Engineering spanning over two decades, one common thing I've observed is that organizations across the globe are constantly working to successfully implement agile & DevOps practices to improve their software development cycle. Yet, a thin percentage of these organizations are leveraging test automation to keep with the pressure of continuous testing.

Whether you're prepared or not, it's time to get serious about test automation. We are currently at an inflection point where people are starting to realize that automation testing is the only golden key that can help them cope up with the increasing pressure of continuous testing.

This report discusses some of the most important questions pertaining to the software testing industry. We hope this report helps tech leaders, QA managers, automation testers and their teams to understand the true power of automation testing, so they unlock its true strategic potential.

We discovered some interesting findings

01 In terms of total testing efforts directed towards automation, **19.3%** of respondents automate more than **50%** of the total tests.

02 When it comes to preferences for the test automation tools, some of the highlighted ones are the ones are Selenium **(64.2%)**, Appium **(22%)**, and Cucumber **(18%)**.

03 Although test automation entered the market more than a decade ago, the overall results show that 18% of respondents are yet not planning to implement test automation anytime soon.

04 Reportedly, **46.7%** of the total respondents use open-source test automation tools at their organization.

.. and many more.

Chapter 2

Profile of testers we surveyed

1579 participants from across the globe responded to this survey.

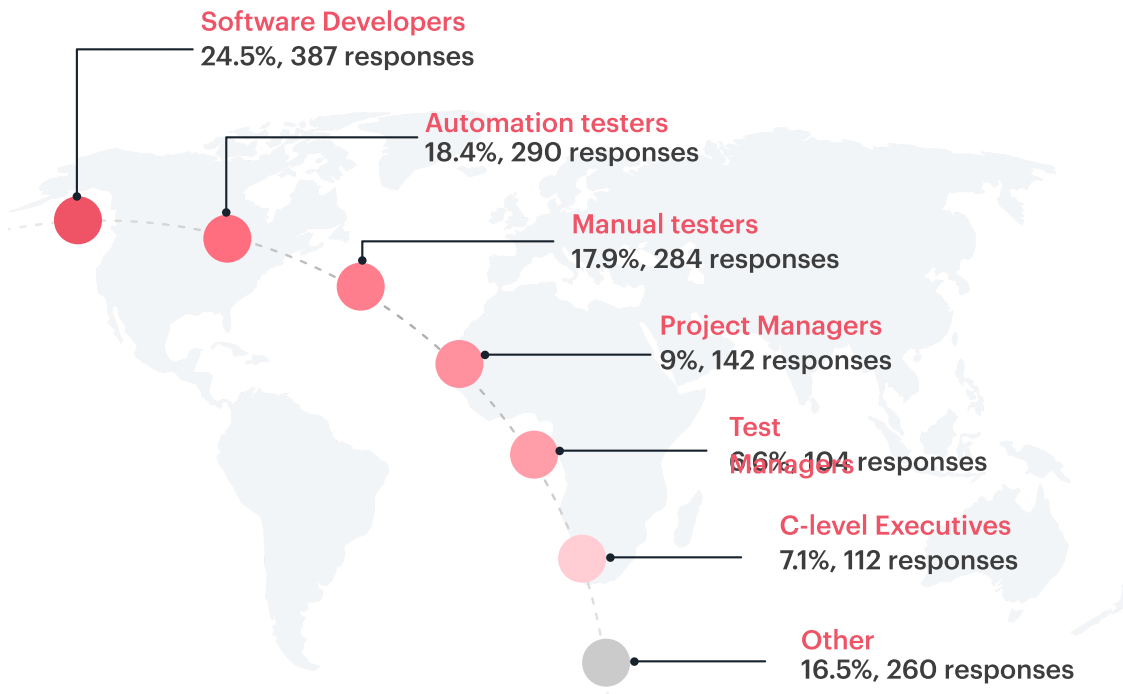
Survey respondents work for a variety of organizations including Google, Facebook, Tinder, TCS, Thrive Global, AWS, and many more.

Let's have a detailed look at the demographics and characteristics of the individual who responded to the survey, their designation, and their experience with test automation.

Current role

Amongst the software testers who completed the survey, we have an interesting combination of different experience levels and types of roles- from C-level executives and test managers to freelancer testers and project consultants.

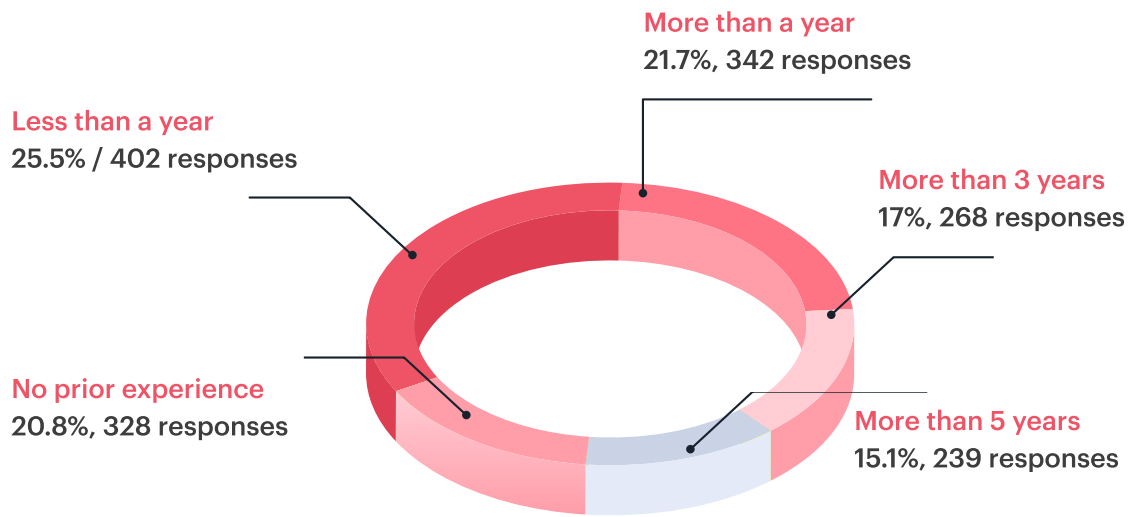
The people who responded to the survey are primarily Software Developers **(24.5%)**, Automation Testers **(18.4%)**, and Manual Testers **(17.9%)**, with significant numbers of Project Managers **(9%)**, Test Managers **(6.6%)** and C-level Executives **(7.1%)**.



Years of experience

Years of test automation experience also varied amongst the survey participants.

25.5% of survey participants have less than 1 year of experience with test automation, **21.7%** have more than 1+ years of experience, **20.8%** have no experience, **17%** have 3+ years of experience and **15.1%** have 5+ years of experience.

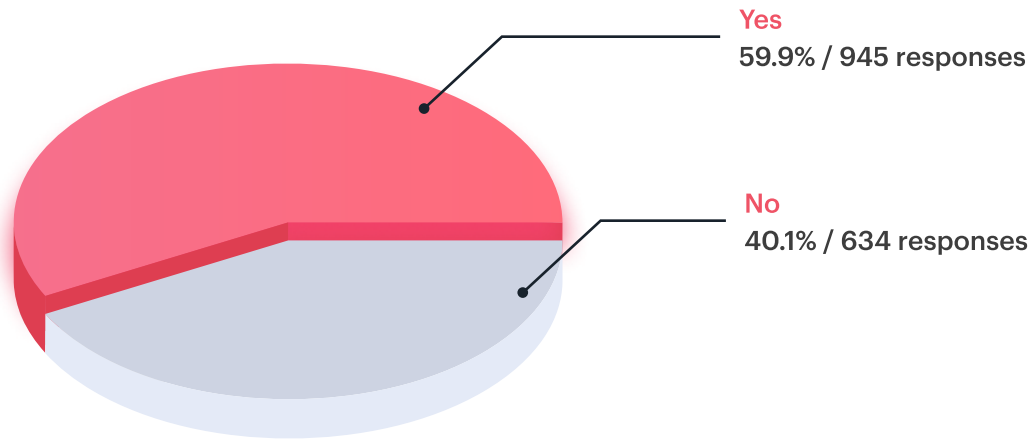


Designated DevOps team

The majority of the respondents (59.9%) admit to having a designated DevOps team in their organization while the remaining of them (40.1%) don't.

When we started development at Simform, deployments were often troublesome. We'd a traditional mindset with devs writing the code and ops deploying it. But over time, we realized that this won't scale, and hence moving towards the DevOps was the only option.

However, what I have observed is a lot of development teams abandon the idea of adopting DevOps after it doesn't work for a quarter or two which is the reason why we observe **40.1%** of people still not part of a DevOps team at their organization. It takes longer than a couple of quarters to wholly reap the benefit of DevOps and people need to be aware of it.



Programming language ecosystem

Analyzing the current testing language environment, Java (50%), JavaScript (49.1%), Python (34.4%), Node.js (29.7%), C# (17.9%), PHP (16.5%), C/C++ (9%), Ruby (5.7%) and Go (5.2%) were the most preferred choices.

One universal choice we observe here is Java and JavaScript being the frontrunners. Firstly, being the simplest language, it is often easy for programmers to get started with test automation. Secondly, it's the capability of WORA (write once, run anywhere) that makes it popular among the developers since it brings the benefits of cross-platform benefits.

On the other hand, the dominance of JavaScript can be credited to the wider adoption of shift-left testing methodologies which usually involve developers in test code development.

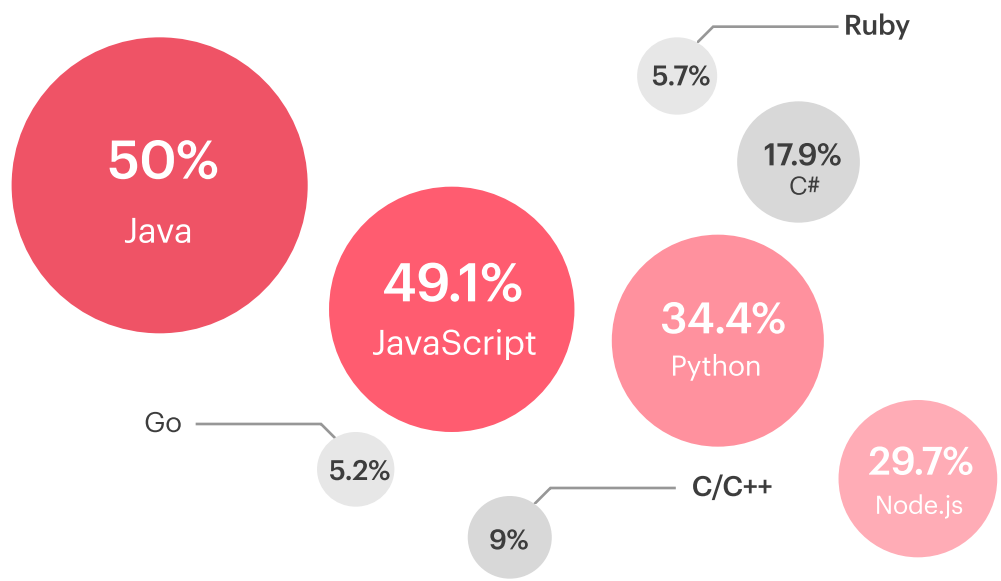
Speaking of the third-most used language Python, its vast amount of tutorials and documentation makes it an attractive option to easily get started with test automation. Moreover, it ships code with IDE as well as text-based live interpreter. This gives an immediate feedback to the testers which aids learning, but at the same time it is fairly slow compared to the compiled languages like C, C++ and Java.

Programming language ecosystem

Another frontrunner is C# which is steadily gaining popularity for test automation because of the inherent power of the language itself and its compatibility with Selenium WebDriver.

Meanwhile, testers are also fond of working with PHP mainly because of its difficulty level in comparison to Python and Java, plus, it supports various frameworks like Laravel Dusk, PHPUnit and Codeception.

Lastly, Ruby which is preferred due to its simplicity and productivity. Also, Selenium is compatible with Ruby and hence Selenium automation testing is possible.



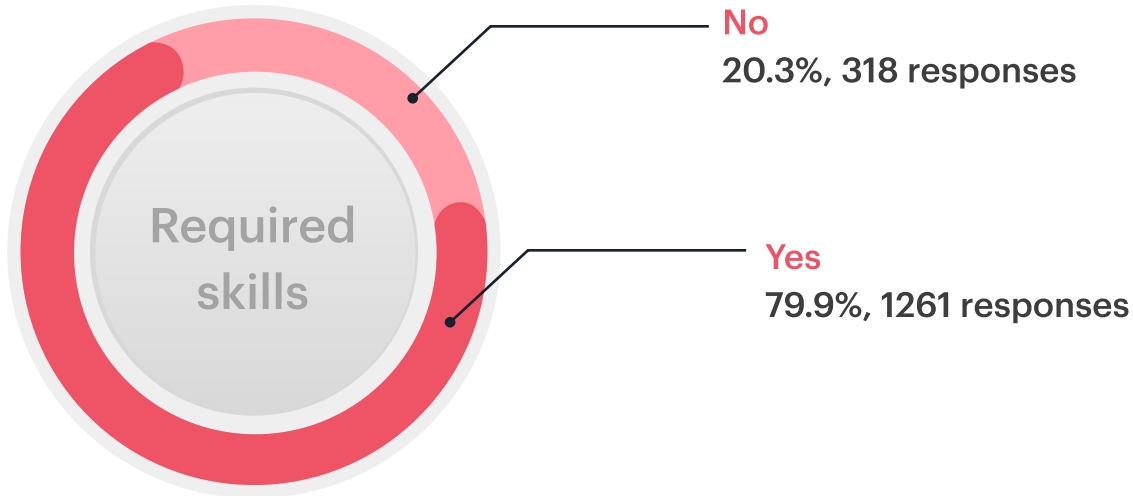
Chapter 3

Skills required for the test automation

Test automation skill requirement

Interestingly, 79.9% of respondents admit to having the availability of required skills and resources while 20.3% didn't.

Having been working closely with the testing teams, these numbers are satisfactory. If you'd have asked me 5 years back, automation testing had a flat curve in terms of availability of the testing talent. However, things changed with the diverse availability of the remote QA services and testing teams in the past couple of years.



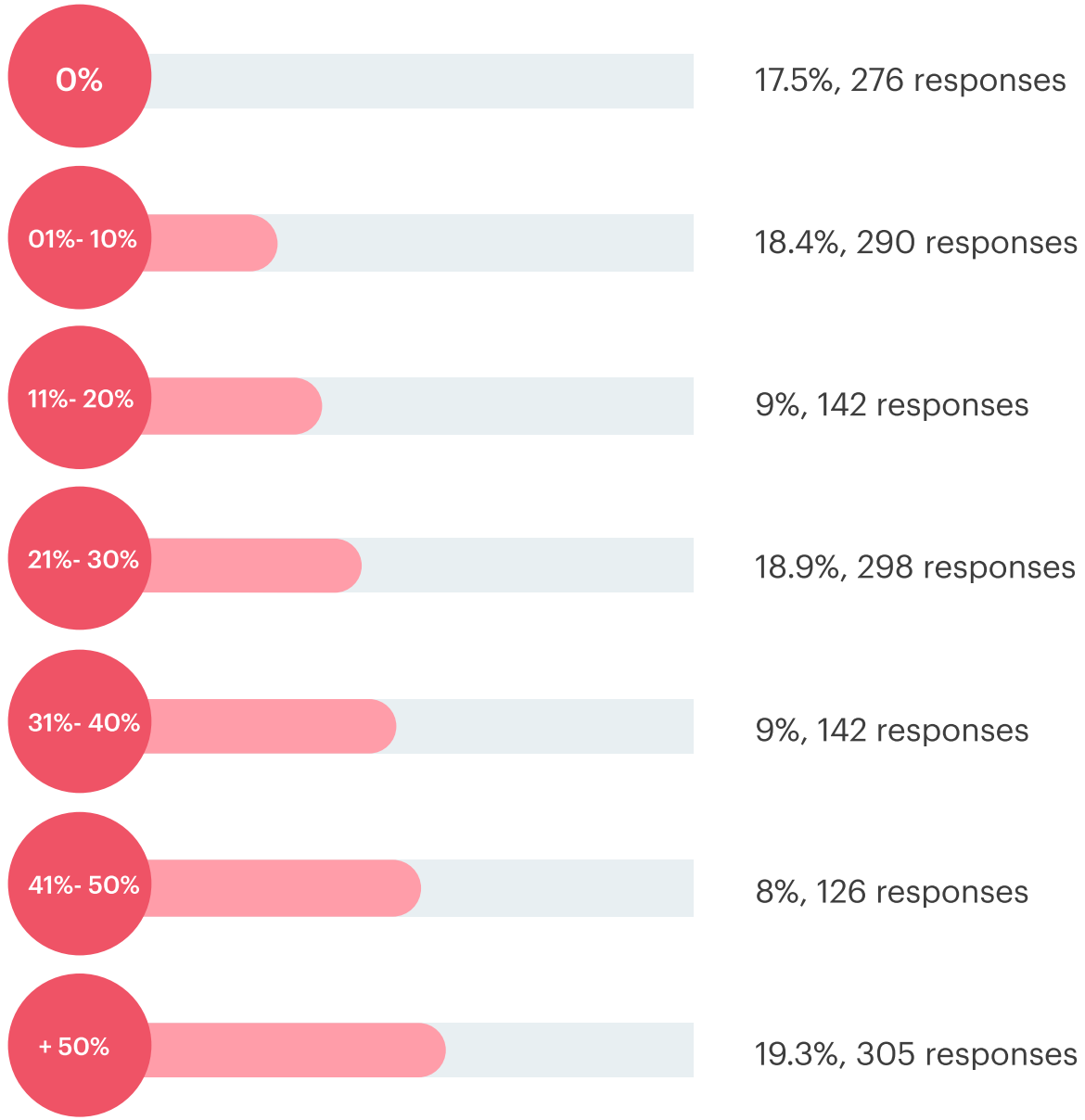
Automated testing efforts

In terms of total testing efforts directed towards automation, **19.3%** of respondents automate more than **50%** of the total tests. On the other hand, **44.9%** of respondents have less than **20%** of their total tests automated.

One noteworthy point that surfaces here is almost **80%** of people admit of having the required skills for the test automation, yet, just **19.3%** of them devote more than **50%** of their total testing efforts.

One major reason behind this is poor experiences and ROI with previous automation testing projects. There's more than just the outer picture when it comes to automation testing and one of the principal ones is choosing the right tool, for the right team, for the right job.

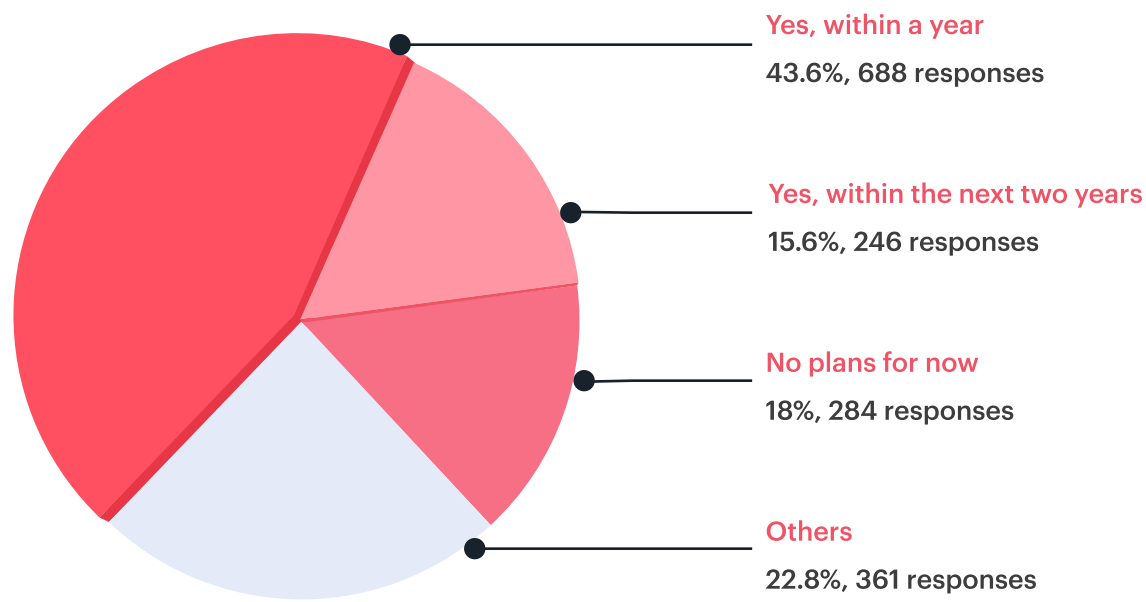
Automated testing efforts



Test automation implementation

Although test automation entered the market more than a decade ago, the overall results show that **18%** of respondents are yet not planning to implement test automation anytime soon.

Meanwhile, **22.8%** of respondents are currently automating their software tests, **43.6%** are planning to do it within a year, and **15.6%** within the upcoming two years.



Chapter 4

Test Automation & types of tests

Test automation preferences

When it comes to automating the tests, the most favored ones are

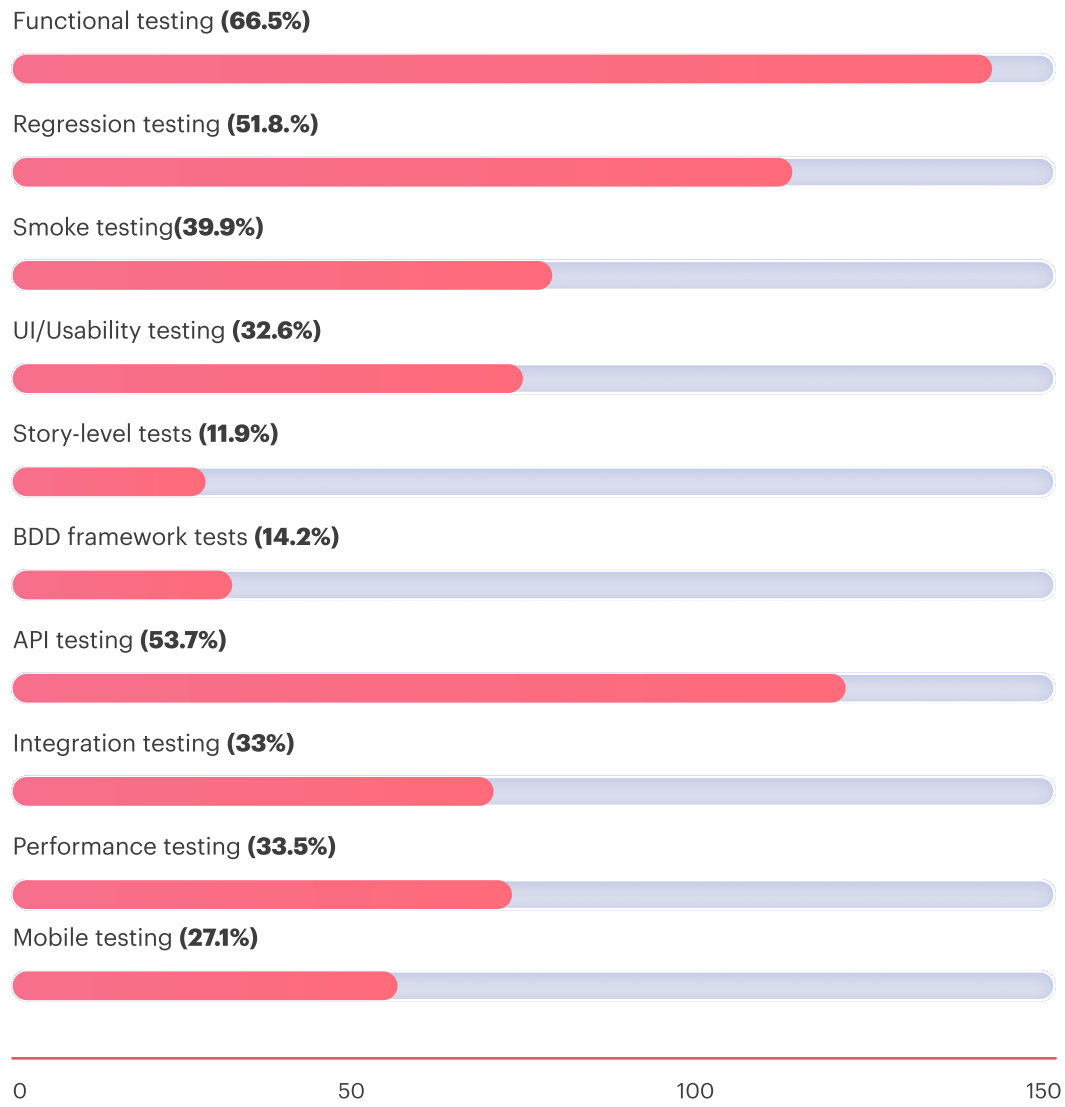
- Functional testing **(66.5%)**
- API testing **(54.2%)**
- Regression testing **(50.5%)**
- Smoke testing **(38.2)**

The common thing between these tests is they are repetitive, involve a lot of data entry, and have clear pass/fail results, which makes them more likely to be automated.

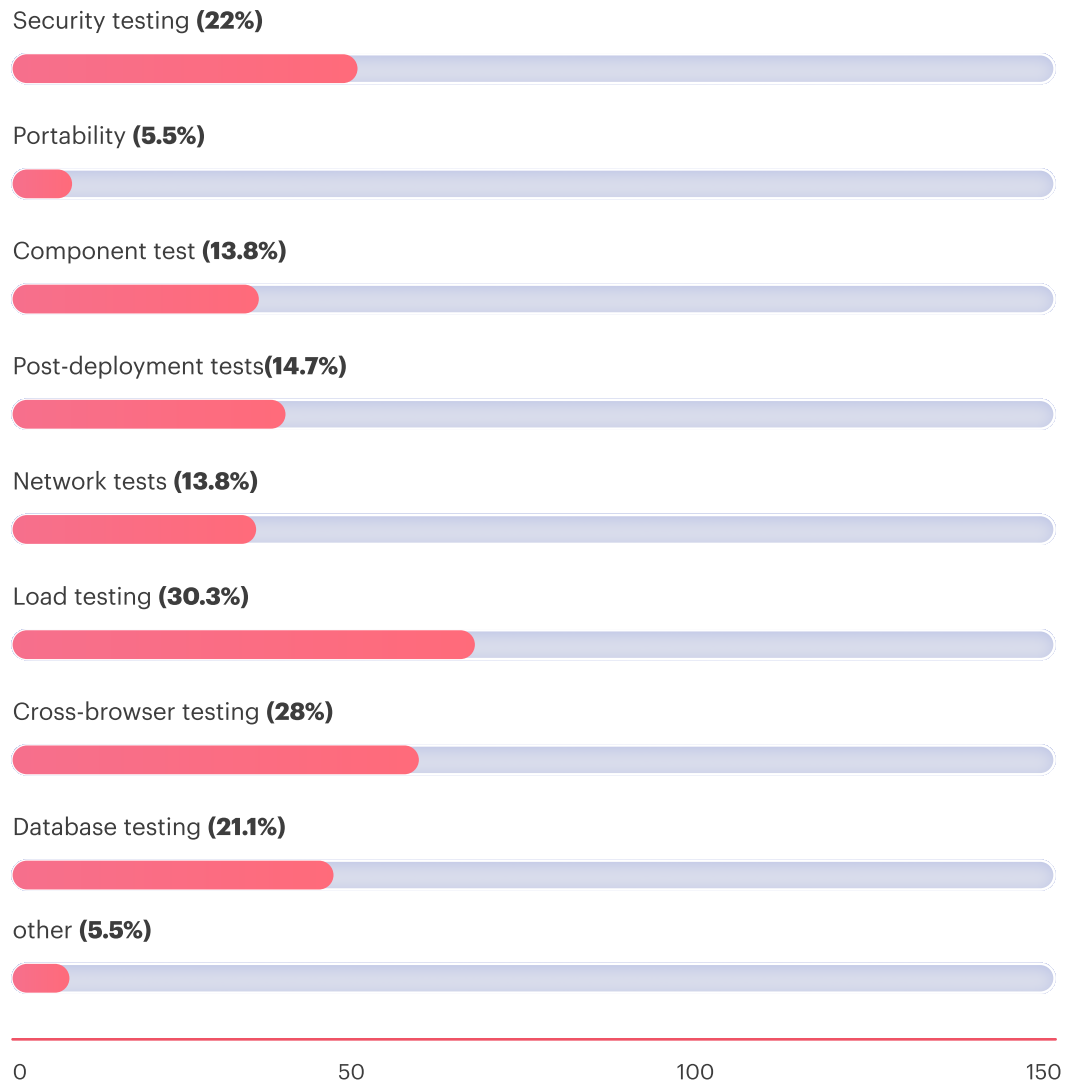
While the least automated tests are

- Probability testing **(5.7%)**
- Story-level tests **(12.3%)**
- BDD framework tests **(13.7%)**

Test automation preferences

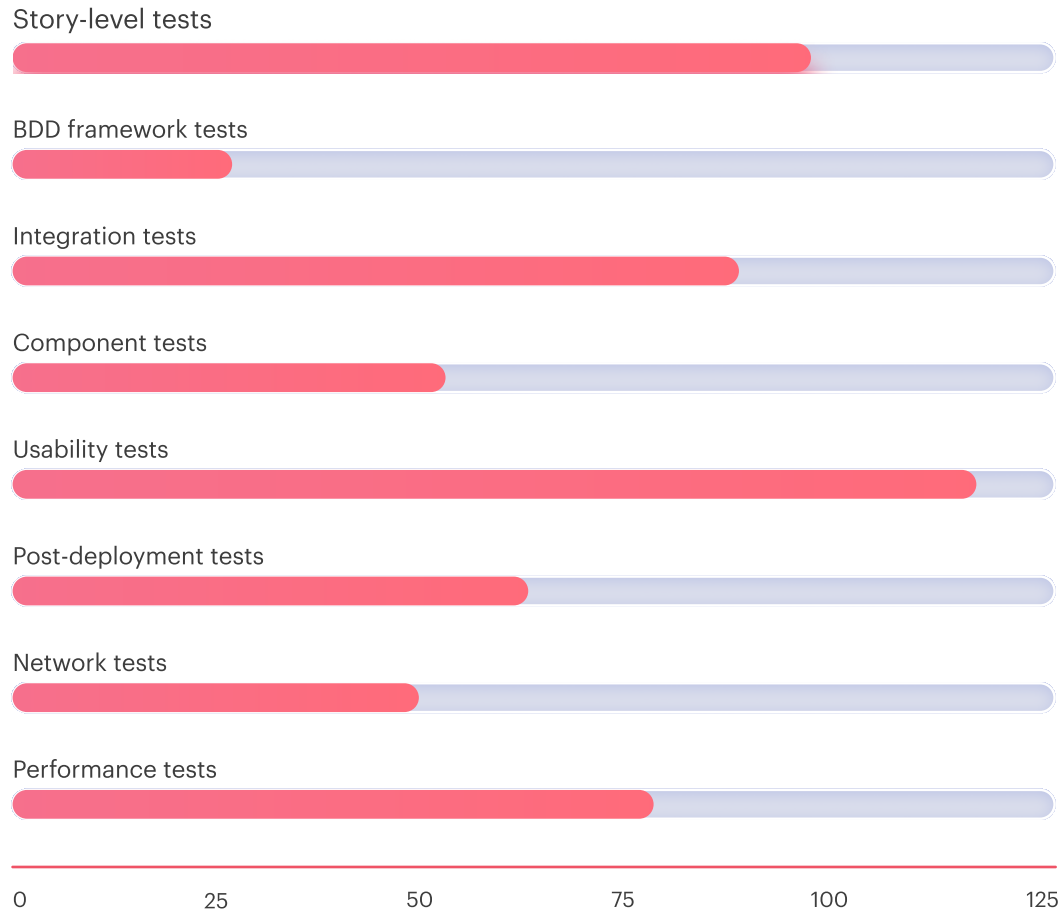


Test automation preferences



Manual testing preferences

On the contrary, there are still a few tests which are performed manually. Among them, the user acceptance test **(56.9%)**, Usability test **(54.5%)**, Story-level tests **(43.6%)**, and Integration test **(41.7%)**.



Test automation bottlenecks

When it comes to barriers to test automation, there are various reasons tests are yet not successful.

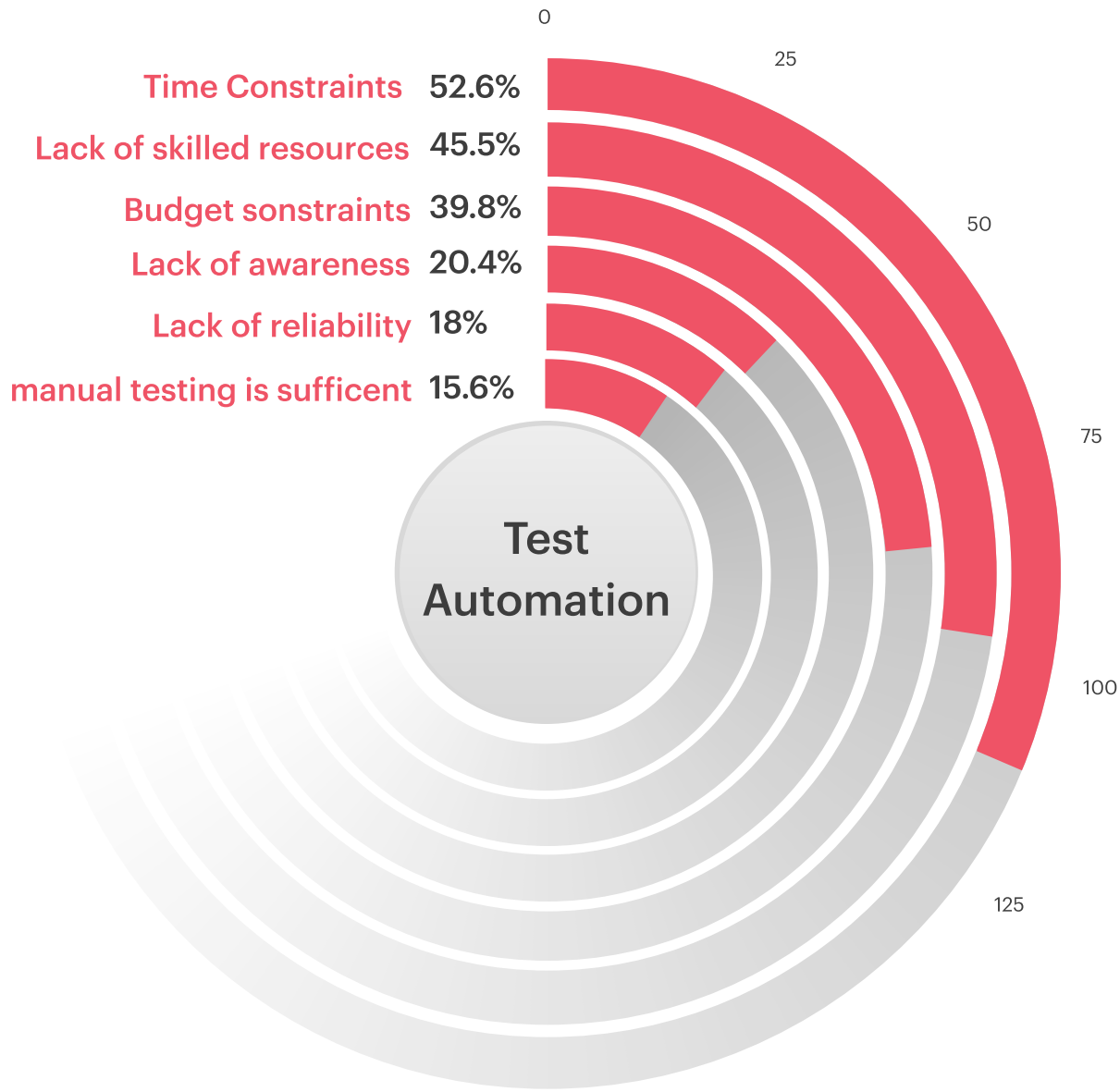
Few of the major ones are

Time constraints **(52.6%)**
Lack of skilled resources **(45.5%)**
Budget constraints **(39.8%)**

As discussed earlier, people might be coming around with the remote teams and trying to make their ends meet for the talent requirement but it still seems the supply is inadequate. Lack of expertise in the current team is another common challenge that a lot of organizations face. While this cannot be solved completely without gaining significant efforts, DevOps can certainly help in educating the current workforce and get started with the transition.

For example, they can make use of the testing frameworks which can utilize the programming languages with which the testing team is already familiar. A good example of this is Selenium which basically automates browsers and has bindings for a number of languages.

Test automation bottlenecks

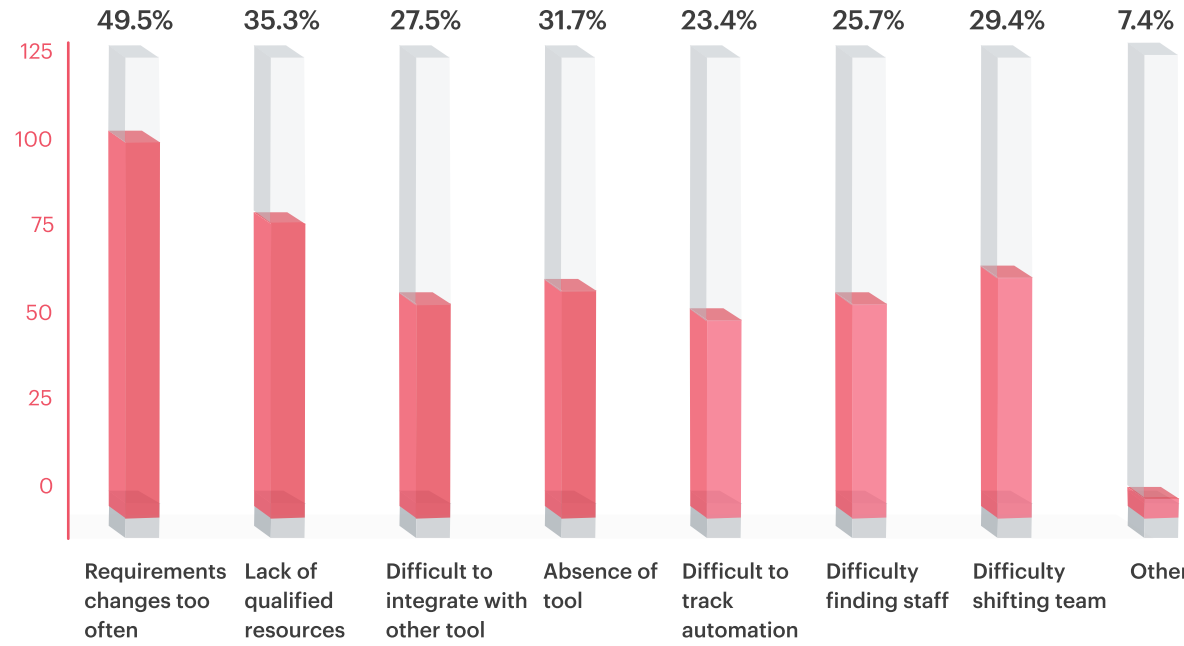


Test automation challenges

All the testers have multiple challenges when it comes to implementing test automation. Some of the most highlighted ones are

- Frequent change in the requirements
- Lack of qualified resources

Almost **50%** of the respondents admit to having frequent changes in the testing requirements as a problem in implementing test automation. The most common aspect of this is lack of predictable test environments and test data for the team to use.



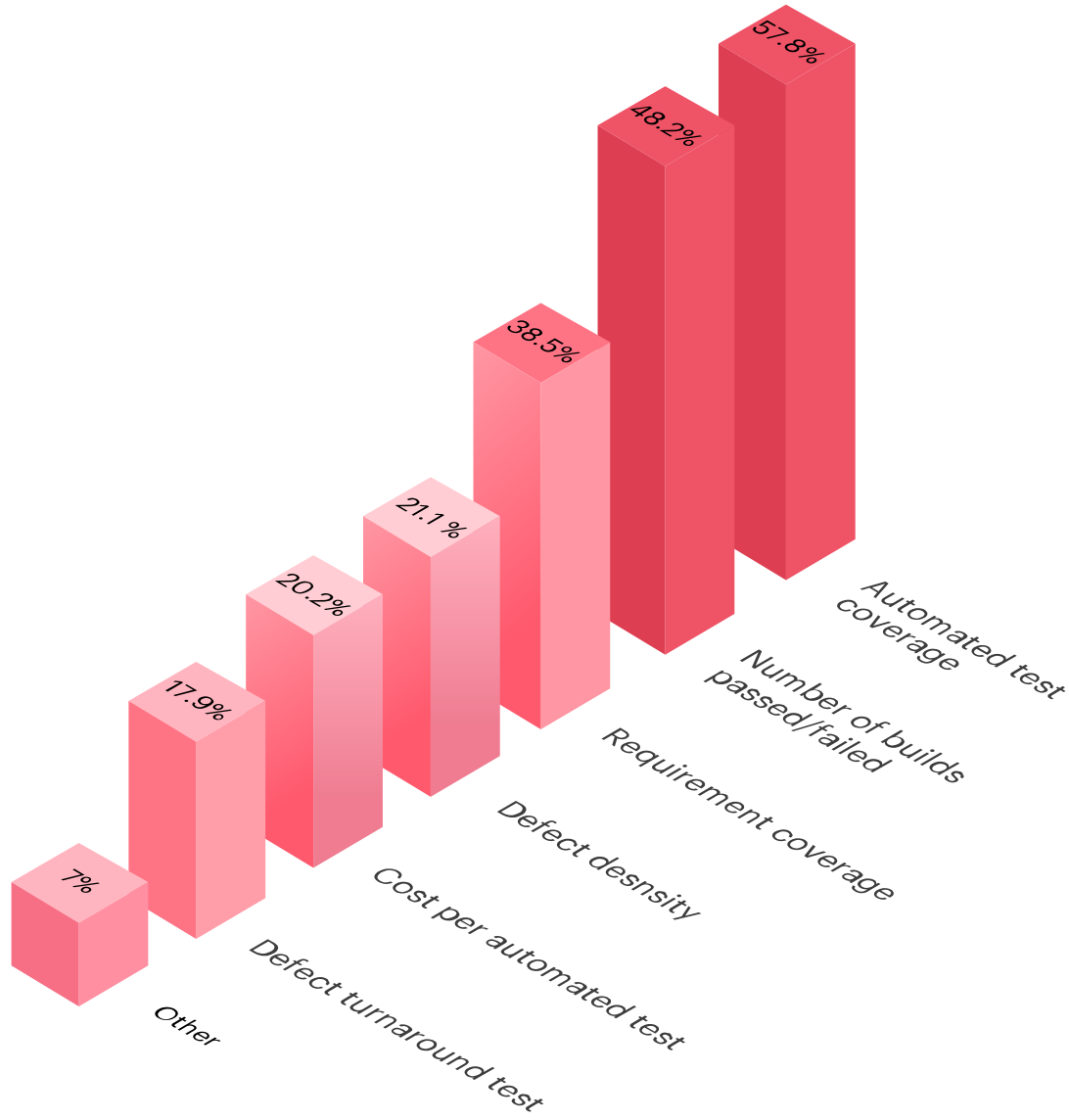
Test automation success KPIs

KPIs for the success of automated testing varied to a greater extent. However, the key KPIs were

- Automated test coverage **(57.1%)**
- No. of builds passed/failed **(47.6%)**
- Requirement coverage **(38.2%)**

Since efficiency of test automation is debatable, KPIs plays an integral part in measuring and improving your team's automated testing process and tracking its status. Moreover, having common APIs across different teams will help everyone be on the same page. There are a number of KPIs that you can take into account, however, one should select which aligns with the business objectives.

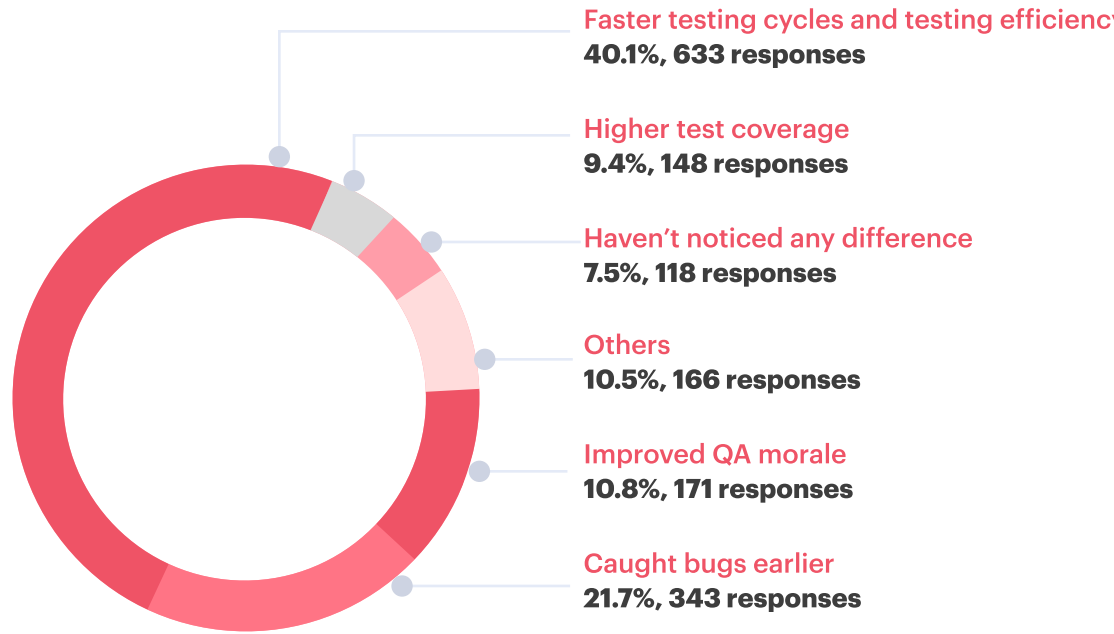
Test automation success KPIs



Test automation benefits

The survey defined some clear advantages that team benefitted from after adopting test automation,

- Faster testing cycles and testing efficiency
- Reduction in the bug reporting cycle
- Improved QA morale
- Higher test coverage



Chapter 5

Tools, preferences and selection criteria

Test automation tool preferences

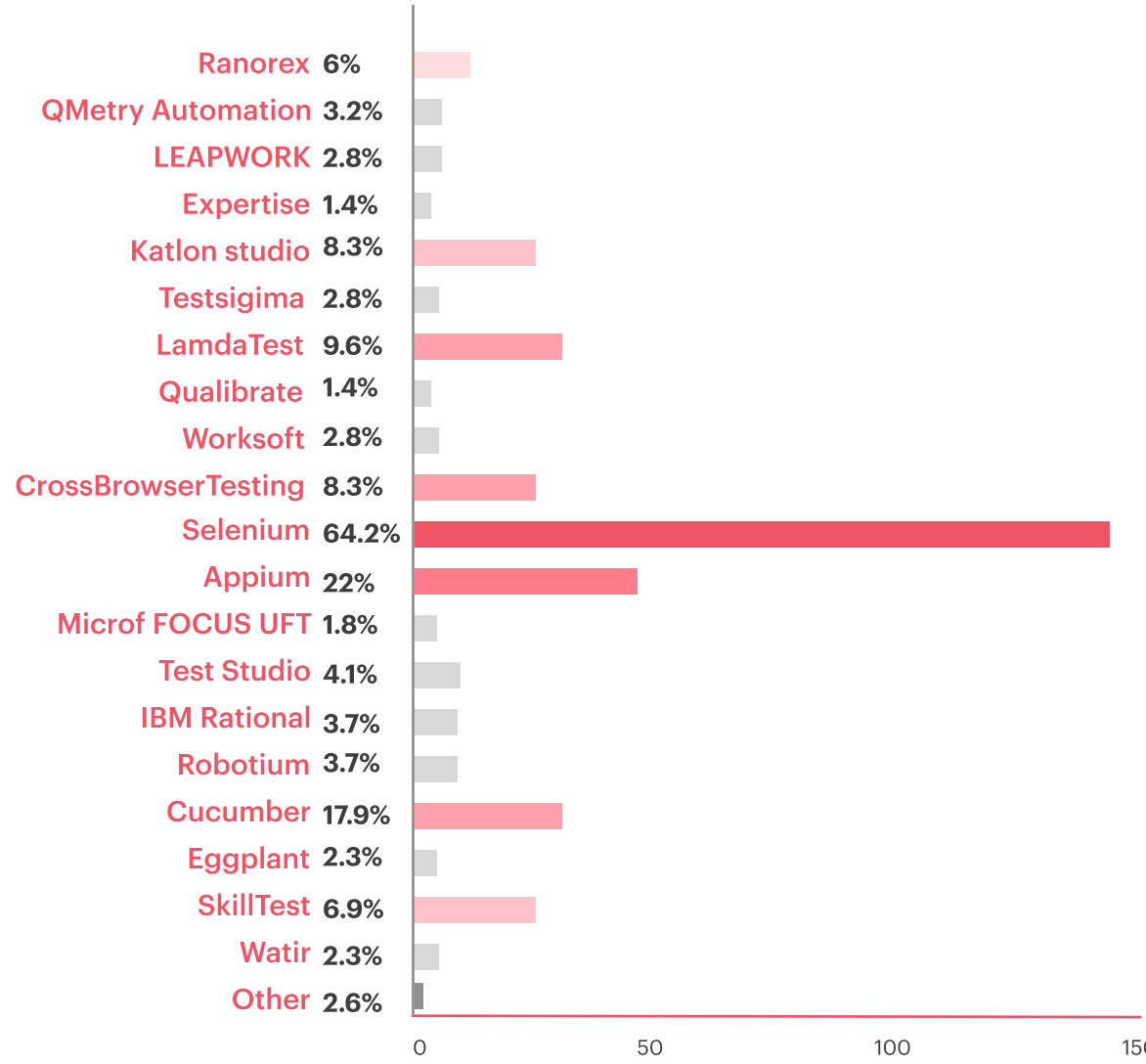
When it comes to preferences for the test automation tools, some of the highlighted ones are the ones are Selenium (64.2%), Appium (22%), and Cucumber (18%).

In fact, a lot of these tools have been in the market for a long time and we've been using them extensively at Simform. Though one word of advice I'd like to share is that selection of tools is a subjective thing because it varies on the individual's requirements.

Though few points to look for in each of them are :

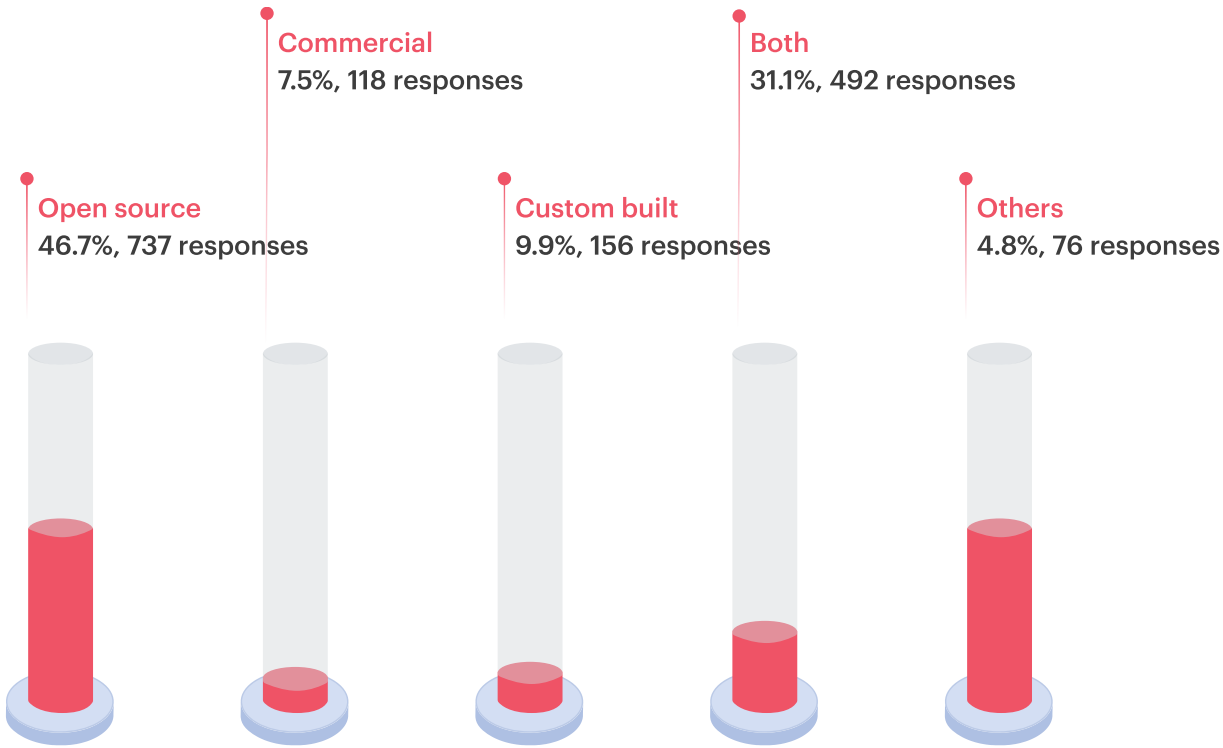
- Features & functionalities
- Licensing & costs
- Skills required & support
- Overall community

Test automation tool preferences



Test automation tool types

Reportedly, **46.7%** of the total respondents use open-source test automation tools at their organization, **7.5%** uses commercial tools, **31.1%** use both open-source and commercial tools while **9.9%** uses custom-built tools for their test automation.

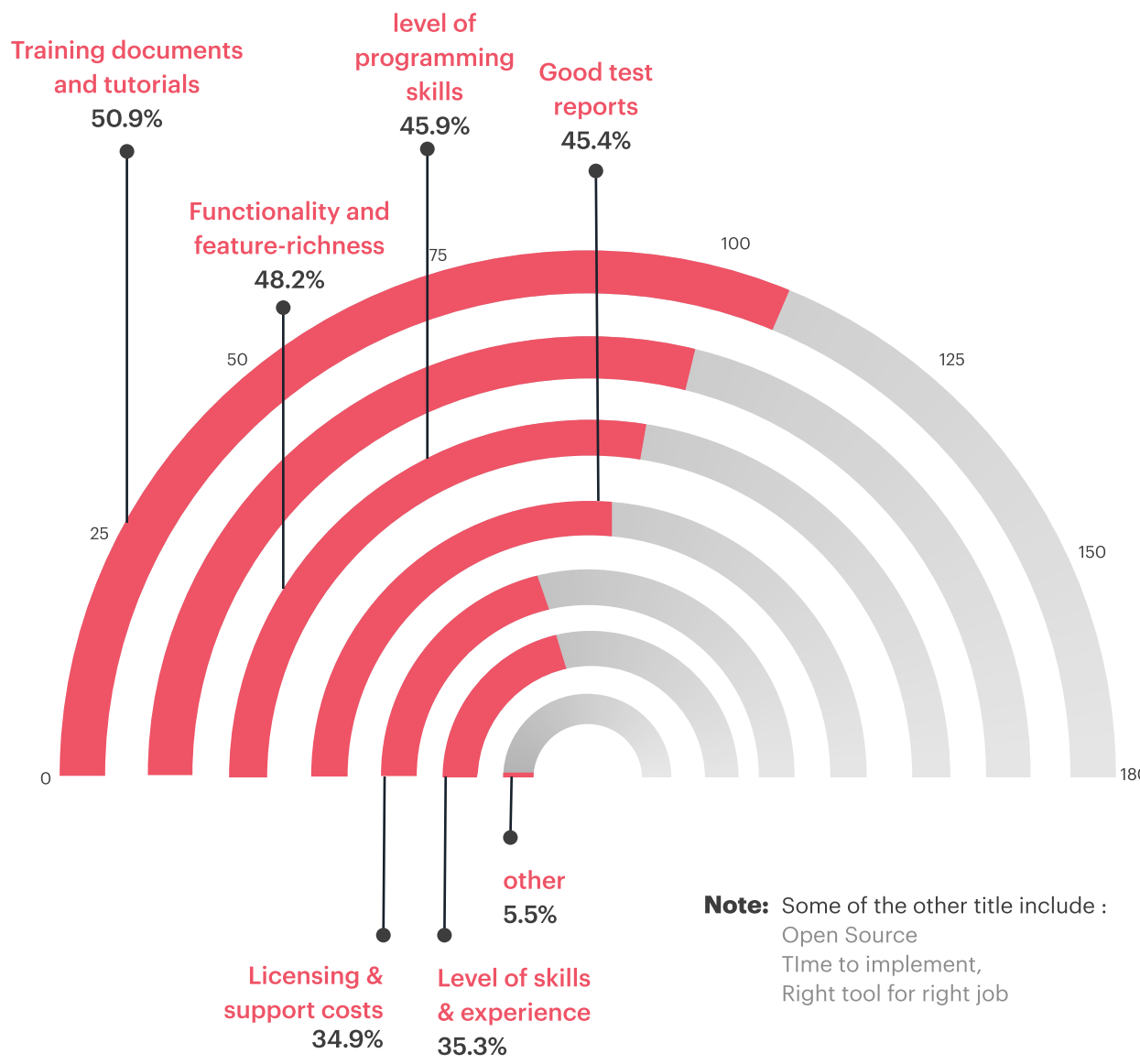


Tool selection criteria

The selection of the tool is largely dependent on the AUT, however, there are few general consideration that were highlighted during our survey,

- More than half of the respondents **(50.9%)** stress on having proper training documents and tutorials for the tools
- Functionality and feature-richness is also one of the top-most concern **(48.2%)** for tool selection criteria
- While 45.4% of the respondents count on the tool's ability to generate good test reports
- A similar number of respondents **(45.9%)** cited the level of programming skills required to operate the tool as a key decision-maker

Tool selection criteria

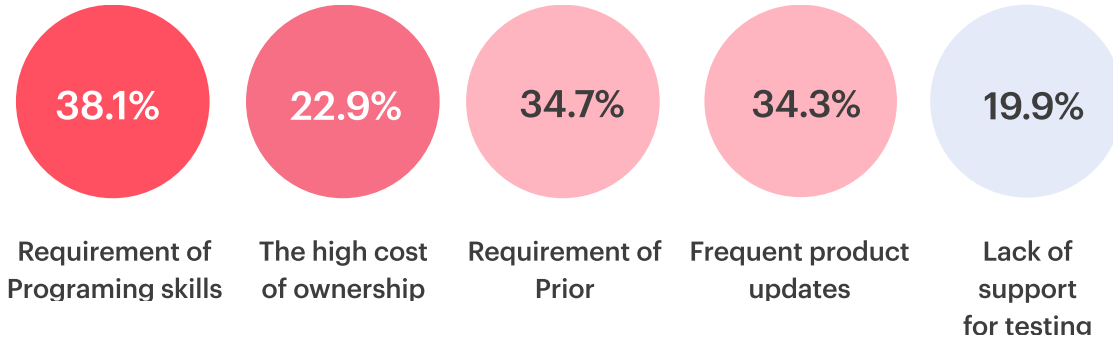


Note: Some of the other title include :
Open Source
TIme to implement,
Right tool for right job

Difficulties with the existing tools

Despite having a diverse range of tools to choose from, each of these comes along with its own shortcomings. Following are the primary problems that surfaced during this survey,

- Firstly, **38.4%** of respondents struggle to meet the programming language requirement that tool requires
- Secondly, **34.4%** of testers share two common problems equally, the requirement of prior experience with the tool and lack of testing non-functional requirements



SIMFORM

Concluding remarks...

This is not a closed report,
It's an open conversation.

With this report, we hope to initiate a conversation and empower you to have a meaningful conversation around test automation backed with data and insights.

If you have any questions on the topic, we'd be happy to hear from you. Send me an email **hiren@simformlabs.com** or connect with me on **Twitter** or **LinkedIn.**

Hiren Dhaduk
VP of Technology
hiren@simformlabs.com

We are Simform!

With over 10+ years of experience under our belt, we are more than ready to supercharge your project with extraordinary code. 10 years ago, Simform was one person. Today, we're Over **600+** people strong and growing.

Simform is a custom software development powerhouse. Let's get in touch to discuss your next project!

Contact Us

- Custom Software Development Services
- Software Testing Services
- Cloud Application Development Services
- Software Product Development Services
- Serverless Application Development
- Application Development Services
- DevOps Consulting
- Infrastructure Management and Monitoring

SIMFORM