TestRail

FOURTH EDITION

Software Testing and Quality Report

Contents

Introduction

For the fourth consecutive year, TestRail has conducted an in-depth industry survey to analyze the evolving landscape of software testing and quality. This report captures insights from thousands of Quality Assurance (QA) professionals, developers, and engineering leaders worldwide, revealing how testing teams are adapting to modern challenges, leveraging automation, and improving software quality.

Since its inception in 2018, our annual survey has played a pivotal role in our mission to understand and address the evolving needs of QA teams worldwide. By connecting with thousands of QA teams globally, we gain valuable insights into their current practices, challenges, and priorities, allowing us to deliver timely solutions and drive excellence in test management systems.

The survey gathered insights primarily from professionals directly involved in testing processes. Most respondents were hands-on testing professionals—QA/Test Engineers and Analysts. This ensures the findings are grounded in real-world experience and relevant to their peers.

In line with our commitment to providing valuable insights, the fourth edition of the Software Testing and Quality report builds on previous years by introducing new questions on compliance and security, artificial intelligence (AI), the QA job market, and continuous learning.

This report explores the current state of QA, examining survey findings across three main areas:

- QA processes, teams, and benchmarks
- Testing tools and technologies
- QA responsibilities, challenges, and priorities

Keep reading to take a deep dive with us into today's world of quality—and what the future may have in store.

SECTION 01

Foreword

Foreword

It's been just one short year since the last edition of the Software Testing and Quality report, and so much in the world of QA has changed—and yet, so much has remained exactly the same.

Each year, the Software Testing and Quality survey gives us a unique opportunity to connect with thousands of QA professionals worldwide and hear their honest, unfiltered thoughts on the state of QA. The insights gleaned from this survey are illuminating, motivating, and sometimes surprising. It's an honor to hear from the QA community at large, and it's an honor to have your voice included in this snapshot of our careers and our lives.

Amongst the biggest changes in the world since our last survey are artificial intelligence (AI) and machine learning (ML). Nearly every platform and product now has AI-driven features, quickly making it a mainstay in daily life. The world of QA is no different, with AI capabilities popping up across the Software Development Lifecycle (SDLC), promising to help us code, test, and release faster and more efficiently than ever before.

However, our survey found that adoption rates of such AI-driven QA tools are remarkably low. While ChatGPT has found footing in QA, the adoption of other tools remains fragmented, and the overall reported impact of AI on QA processes and efficiency is mixed.

Despite the rush to integrate AI-driven features across every layer of the tech stack and the buzzy headlines heralding the dawn of a new fully-automated and fully-intelligent era, the reality of QA is that most things haven't changed much in the past few years. Teams are still hiring manual testers, still struggling with automating as many tests as they'd like to, and still balancing the demands of faster releases with maintaining high quality.

While we have no doubt that technologies such as AI and ML will make massive impacts on QA and the SDLC at large, it seems that time is not upon us yet. Quality remains a largely people-driven pursuit, with its most valuable resources being the humans behind the machines.

To that end, we added some new questions in this edition to help benchmark QA team composition, the QA job market, and sources of continuous learning—adding a more people-driven layer to the insights we collect year-over-year.

With that, we are honored to present the full findings of the fourth edition Software Testing and Quality Report. We hope you'll finish this report feeling more connected to your fellow humans in QA, and excited for what the next year in quality will bring.

Happy testing,
Amanda Sundara
General Manager, TestRail

SECTION 02

Key Insights

1. Testing Metrics and KPIs

The highlights:

- **Test Pass/Fail Rate (70%)** is the most commonly tracked metric, indicating a strong focus on fundamental software validation.
- **60% of teams track defects reported in production**, emphasizing post-release quality control.
- **43% measure automated test creation and 42% measure automated test execution,** highlighting the growing reliance on automation.

Top challenges:

- Many QA teams lack tracking for advanced quality indicators such as **cost per defect, defect leakage rate, and test automation ROI.**

> **“While test pass rates are useful, true quality measurement requires deeper defect analysis and efficiency metrics.”**

2. Top Testing Priorities

The highlights:

- **35% of teams ranked increasing test coverage as their top priority,** ensuring broader software validation.
- **20% prioritized reducing bugs in production,** aligning with quality-driven development.
- **13% identified automating more tests as their top focus,** aiming to reduce manual testing overhead and improve speed.

Top challenges:

- Automation remains a goal, but many teams struggle with implementation due to **skill gaps, tooling issues, and maintenance overhead.**

> **“We want automation, but executing it effectively remains a challenge due to resource and technical constraints.”**

3. Major QA Challenges

The highlights:

- **33% of teams report end-to-end testing complexity** as their biggest challenge, particularly in integrated systems.
- **32% struggle with developing automated tests,** even though it's a major priority.
- **32% feel disconnected from early-stage development,** leading to delays and misalignment with product teams.

Top challenges:

- QA teams face persistent difficulties in **automation execution, early-stage involvement, and managing complex test environments.**

> **"Testing efficiency is constrained not by budget but by process bottlenecks and lack of integration with development."**

4. Future Initiatives and Industry Trends

The highlights:

- **43% of teams plan to increase test automation,** including AI-driven solutions for self-healing tests and defect prediction.
- **39% are shifting toward earlier QA involvement (Shift-Left Testing)** to catch defects before they reach production.
- **35% are prioritizing better test data and environment stability,** as unreliable environments slow down execution and reduce confidence in results.

Top challenges:

- Teams struggle to keep up with **evolving testing tools, AI adoption, and skill shortages** in automation engineering.

> **"Staying ahead in testing requires continuous learning, but rapid technology shifts make it difficult to adapt."**

Key Insights

5. Speed vs. Quality Dilemma

The highlights:

- **58% of teams report that rapid releases lead to defects slipping into production,** showing that speed often comes at the cost of thorough testing.
- **45% indicate that CI/CD adoption improves with QA team size,** suggesting that larger teams can integrate automation more effectively.
- **Most teams with strong automation and CI/CD integration** report both faster release cycles (86%) and reduced defect leakage (71%), demonstrating the importance of such technologies in keeping up with modern development demands.

Top challenges:

- Balancing **fast deployment expectations with sufficient testing depth** remains an industry-wide challenge.

> **“Our biggest struggle is meeting release deadlines while ensuring we don’t ship defects to production.”**

SECTION 03

QA Processes, Teams, and Benchmarks

Section Introduction

This section explores how QA teams structure their testing processes, manage personnel, and respond to evolving development demands. From planning and staffing to methodologies and tooling, the data reveals both consistency and change—manual testing and in-house QA remain strong, but teams face increasing pressure to automate, integrate, and scale more efficiently.

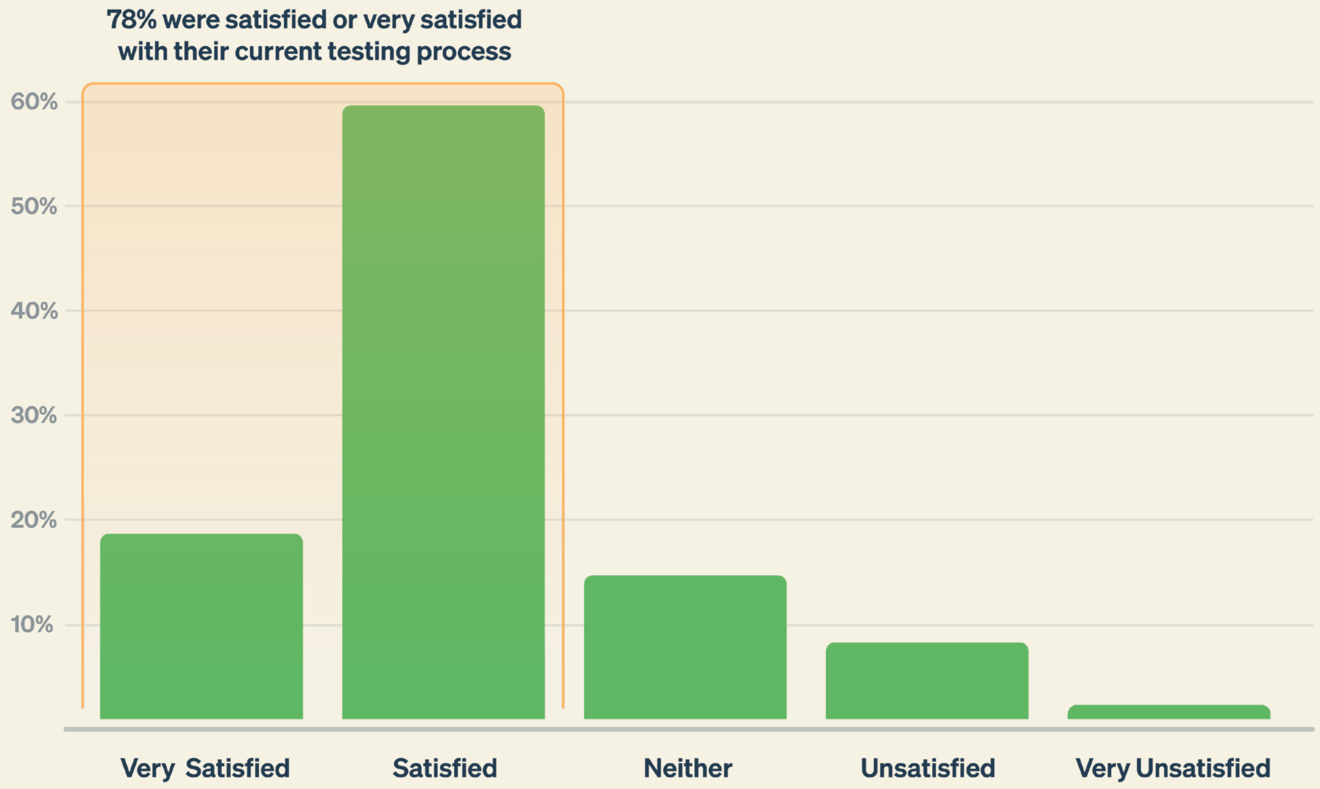
This section also examines how satisfaction with QA processes correlates with factors like early involvement, automation maturity, and team efficiency. For many teams, resource constraints continue to shape their ability to grow and evolve, especially when hiring specialized talent or adapting to new security and compliance requirements.

This section is divided into three main areas:

- **Testing Practices and Processes:** Insights into satisfaction levels, manual testing trends, release frequency, methodology adoption, and correlations between team maturity and testing goals.
- **Compliance, Security, and Standards:** An overview of which security roles are involved in testing, what compliance frameworks teams must follow, and which security tools are used within QA workflows.
- **QA Personnel Composition:** A breakdown of how QA teams are staffed, which roles they are hiring for, the use of external testing partners, and the most common challenges teams face in recruiting and retaining QA talent.

These insights help to illustrate where QA teams are today and what it takes to keep testing practices aligned with shifting priorities, tighter timelines, and increasingly complex software environments.

How satisfied are you with your current testing process? Why?

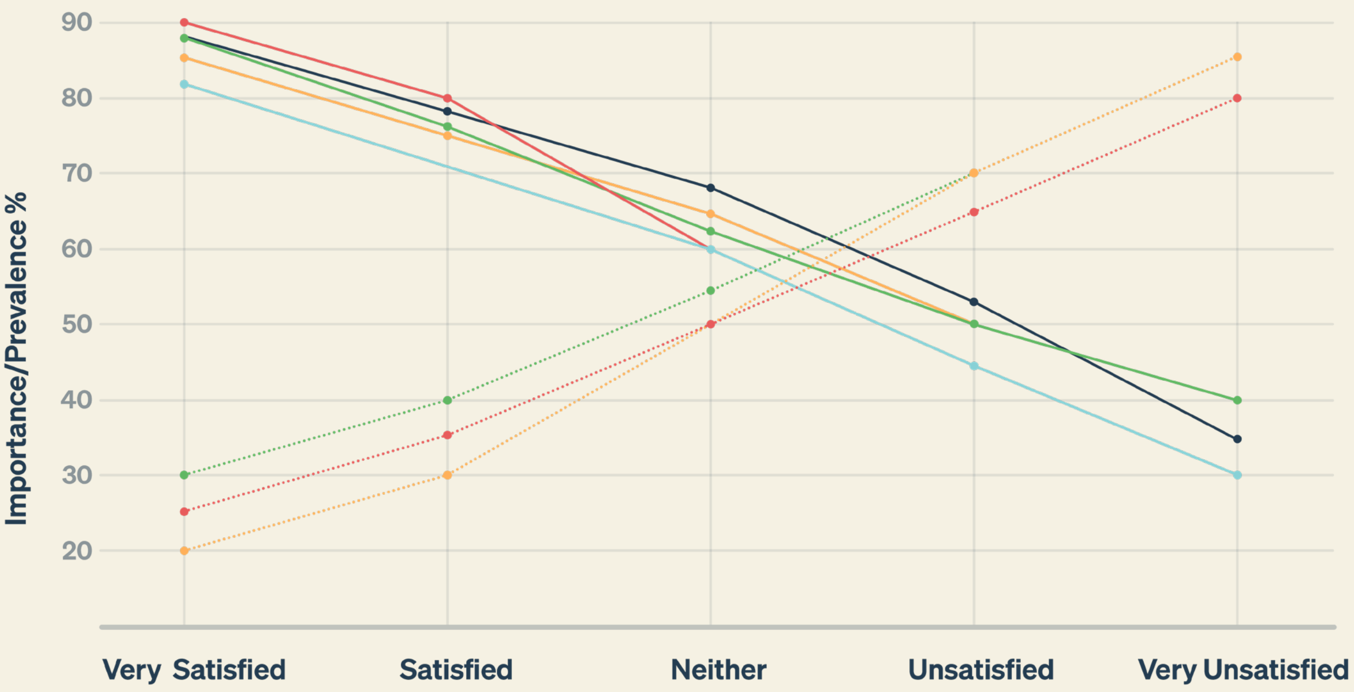


78% of respondents said they're satisfied or very satisfied with their current testing process. That's a strong signal that many teams have solid foundations in place, especially when it comes to test execution and cross-team collaboration.

Still, about 1 in 5 teams said there's room for improvement. For many, that likely means they're still working through challenges with automation, efficiency, or getting involved early enough in the development process.

And while just 7% of teams reported being dissatisfied, it suggests there are some teams out there struggling with outdated tools, siloed workflows, or limited resources—areas where better processes and support could make a big difference.

Correlation Insights: Satisfaction Levels vs. Testing Objectives, Challenges, and Priorities



What drives high satisfaction in QA teams?

The data shows that satisfied teams share a few key priorities in their testing processes.

Here's what stands out among teams who reported being very satisfied or satisfied:

- 90% make **test automation** a top priority
- 85% focus on increasing **test coverage**
- 82% **involve QA earlier** in the lifecycle
- 88% value **efficiency** as a core objective
- 88% emphasize **preventing bugs** before they hit production

> **"Our testing process ensures good coverage, efficient automation, and smooth CI/CD integration."**

Teams that focus on early involvement, automation, and efficiency are more likely to feel good about their testing processes. These aren't just best practices, they're clear drivers of satisfaction.

What causes dissatisfaction in QA teams?

While most teams are satisfied with their testing processes, the survey also sheds light on what holds others back. For the 7% of respondents who reported being dissatisfied, a few clear themes emerged: lack of time, disconnected systems, and underinvestment in QA.

Among dissatisfied teams, here are the common factors identified in the survey:

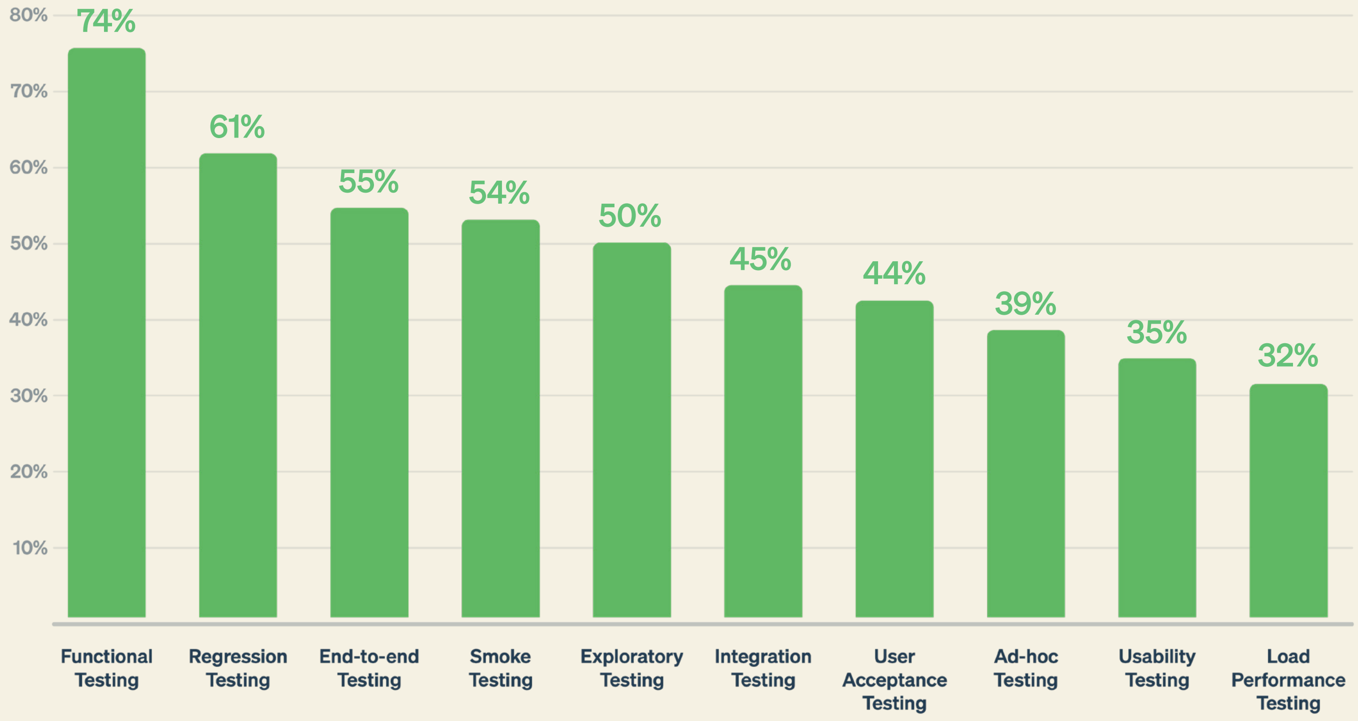
- 85% report ongoing challenges with **CI/CD integration**
- 80% say they **don't have enough time** for proper QA
- 85% cite **budget limitations** as a major blocker
- Only 30% of dissatisfied teams said they prioritize involving QA early in the development process. This suggests that for the remaining 70%, testing is still happening too late—likely after key decisions have been made—which may be contributing to their dissatisfaction.

> **"Testing is chaotic, rushed, and lacks proper planning."**

These challenges often stack—underfunded teams are less likely to adopt automation or shift-left testing, and time constraints lead to rushed or incomplete QA. This results in lower confidence, more bugs in production, and strained collaboration across teams.

- High-performing QA teams **learn from past failures** and **refine their processes.**
- Retrospective data suggests that **teams who track defect causes and adjust accordingly** improve faster over time.
- A holistic **culture of quality,** with it being a shared responsibility across the entire SDLC, helps improve quality from start to finish.
- Adopting **agile methodologies,** such as **retrospectives and post-mortems,** is critical to a culture of continuous improvement.

What kinds of testing does your organization currently do manually?

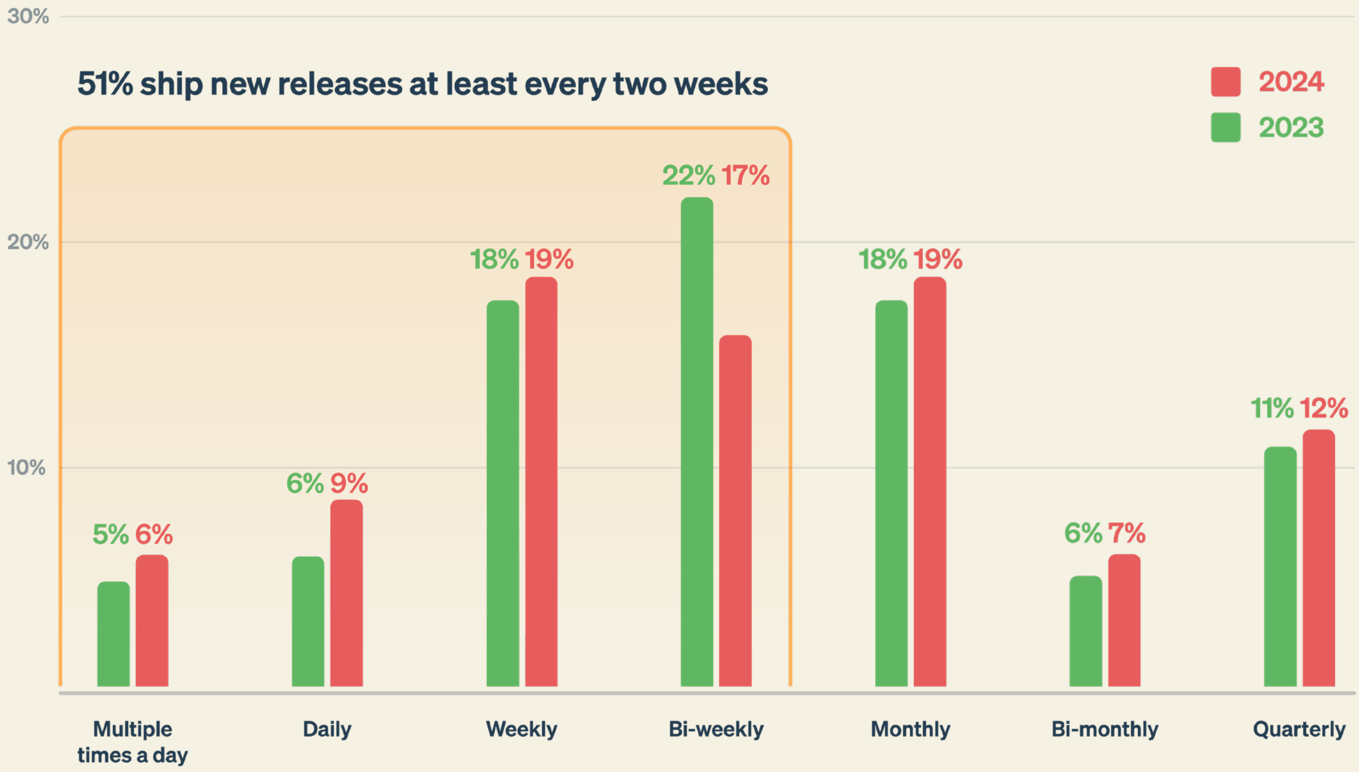


Manual testing remains a core part of QA workflows, especially for areas that require human judgment or are harder to automate.

- **Functional testing (74%)** is the most common manual activity, showing teams still rely on hands-on validation for core features.
- **Regression testing (61%)** is the second-most common type of manual testing, suggesting ongoing challenges with automating repetitive tests.
- **Exploratory testing (50%)** remains largely manual by nature, as it relies on human intuition and investigation.
- **End-to-end (55%) and smoke testing (53%)** are also commonly manual, likely due to system complexity or UI volatility.
- **Integration testing (45%) and user acceptance testing (45%)** often require stakeholder involvement or cross-team coordination, making automation more difficult.

Even in automation-heavy environments, manual testing continues to fill critical gaps—proving it's not going away, but evolving alongside automation.

How often does your organization deploy new code or release new products?



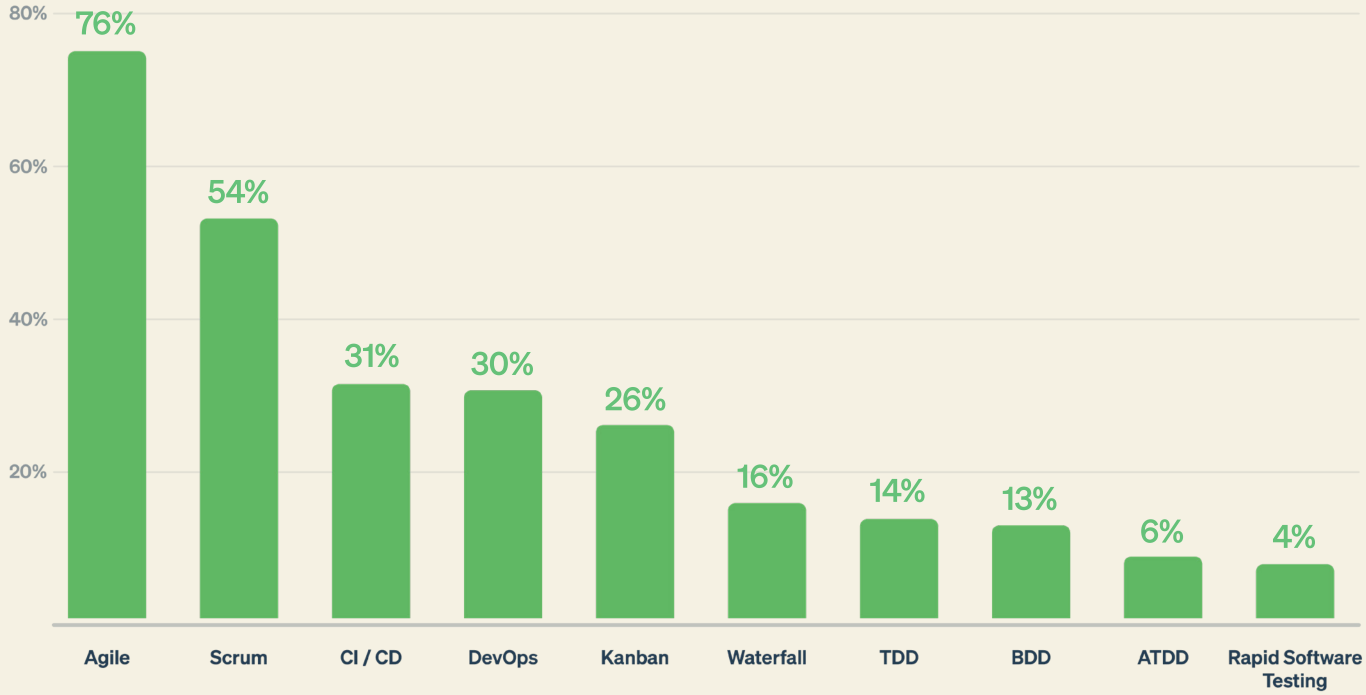
Release frequency continues to vary across teams, reflecting different business models, industry requirements, and levels of DevOps maturity. While some teams push changes multiple times a day, others follow a more structured or slower cadence to prioritize stability.

- Over 55% of respondents follow a **structured release schedule.** Most commonly weekly, bi-weekly, or monthly.
- 15% deploy **daily or multiple times a day,** highlighting the growing adoption of DevOps and continuous delivery practices.
- 19% release on a slower schedule **(bi-monthly or quarterly)**, often due to regulatory constraints or risk-averse cultures.
- 11% follow a **flexible or ad hoc cadence,** likely due to legacy systems or rapidly shifting priorities.

Year-over-year, the data shows a clear shift toward faster, more frequent releases. Daily deployments rose from 6% in 2023 to 9% in 2024, while multiple daily releases also ticked up slightly. At the same time, bi-weekly deployments declined, and fewer teams reported having no set release cadence, signaling a move toward more predictable and iterative delivery.

These trends align with broader industry movements toward DevOps and CI/CD, where small, continuous updates help teams release faster, reduce risk, and stay aligned with business demands.

Does your team use any of the following development methodologies and/or techniques today?



Agile remains the dominant methodology across QA teams, with 76% of respondents reporting its use. **Scrum (54%)** is the most common Agile framework, reinforcing the industry's strong preference for iterative, collaborative development.

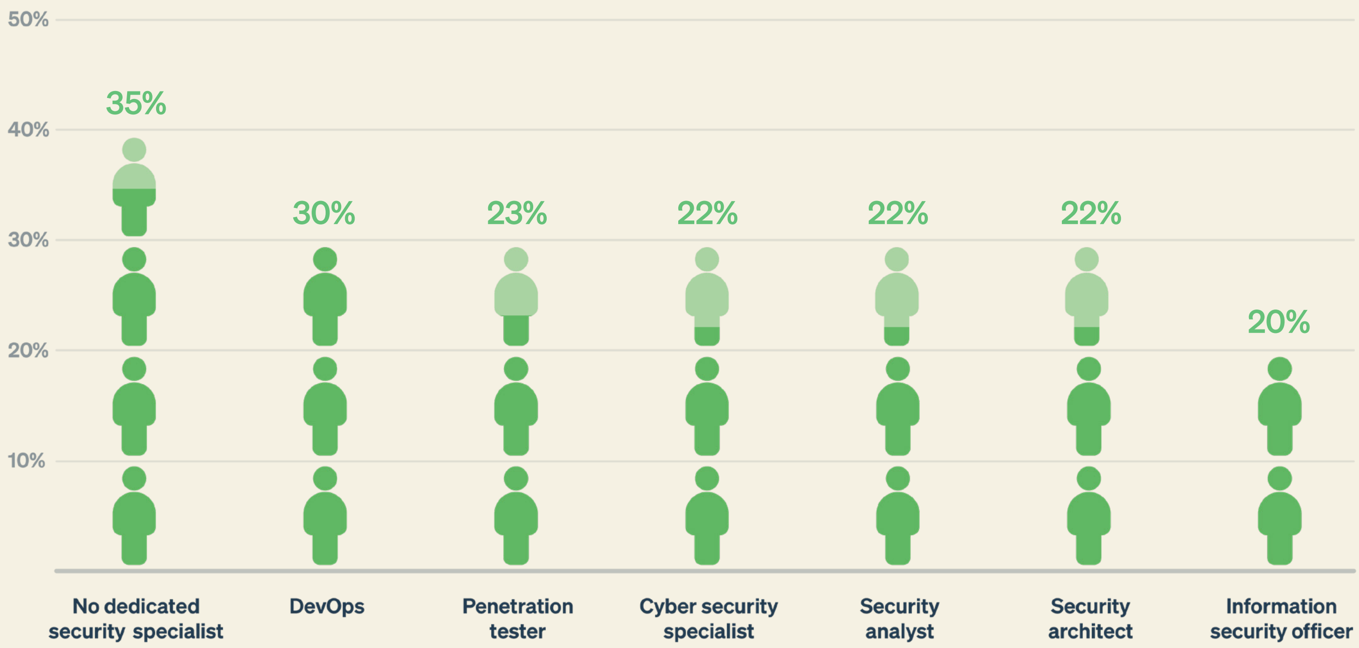
Meanwhile, **DevOps (31%) and CI/CD (30%)** continue to hold sizable shares across QA teams—highlighting an emphasis on automation, speed, and delivery efficiency.

Kanban (26%) also stands out as a complementary approach to Agile, especially for teams favoring continuous flow over time-boxed sprints.

Despite Agile's foothold in QA, 16% of teams reported using **Waterfall**, showing that traditional, sequential models remain relevant for certain industries, risk tolerances, or project types.

The mix of methodologies reflects the diverse needs of QA teams—and underscores the importance of adaptable processes that support both speed and stability.

What roles are involved in your security testing?



Security testing responsibilities vary widely across organizations and many teams still lack dedicated security roles. Over 35% of respondents reported that no security-specific roles are involved in their testing process at all.

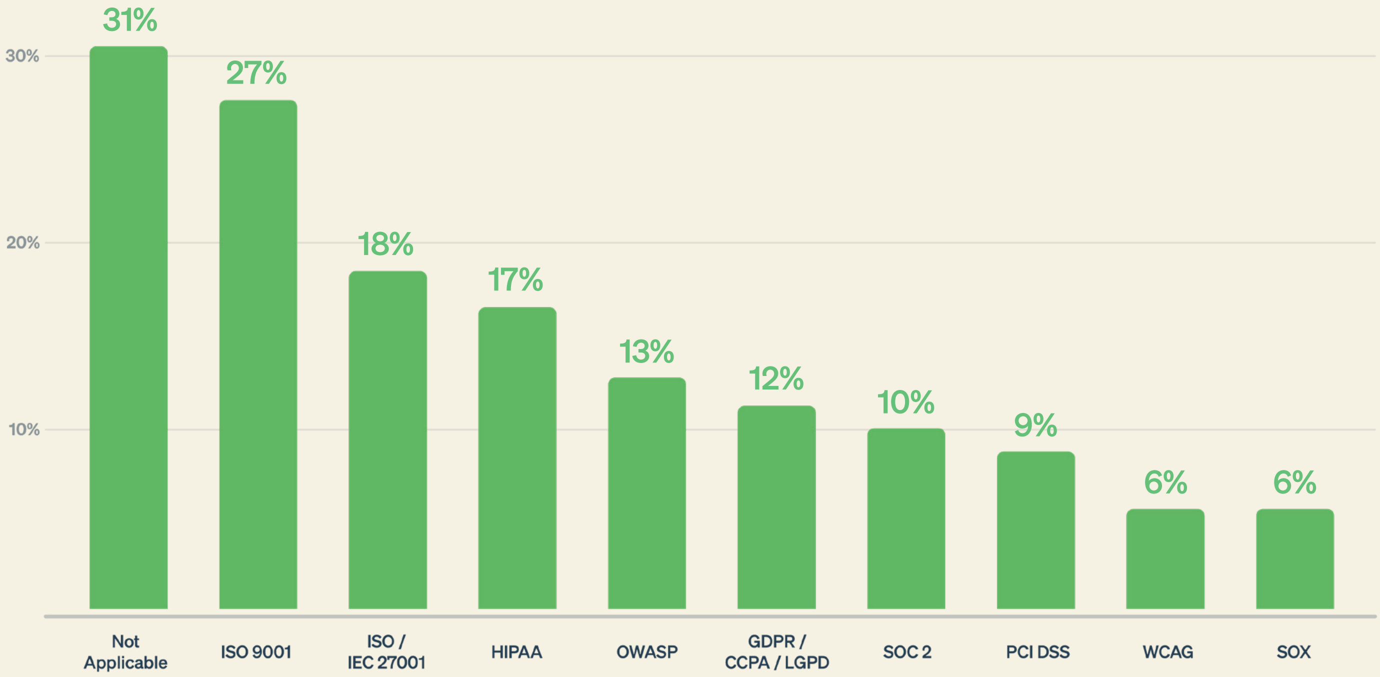
Among teams that do include dedicated security roles:

- **DevOps (30%)** is the most commonly involved function, suggesting security is increasingly being baked into deployment pipelines.
- **Penetration testers (23%), security architects/analysts (22%)**, and **cyber security specialists (22%)** are also frequently involved.
- **Information security officers (20%)** round out the list of common contributors.

In many cases, security testing is handled outside of the QA team, either by third-party vendors, development teams, or compliance teams. Some organizations lean on automated tools, security champions within engineering, or external audits rather than building in-house security expertise.

This data reflects a growing awareness of security's importance but also a fragmentation in how it's managed, especially for teams without the resources to dedicate full-time security personnel.

What kinds of compliance or regulatory standards does your QA team have to abide by?



Compliance requirements vary significantly by industry, but a notable portion of QA teams aren't directly governed by formal standards. 31% of respondents said that compliance standards are not applicable to their QA processes.

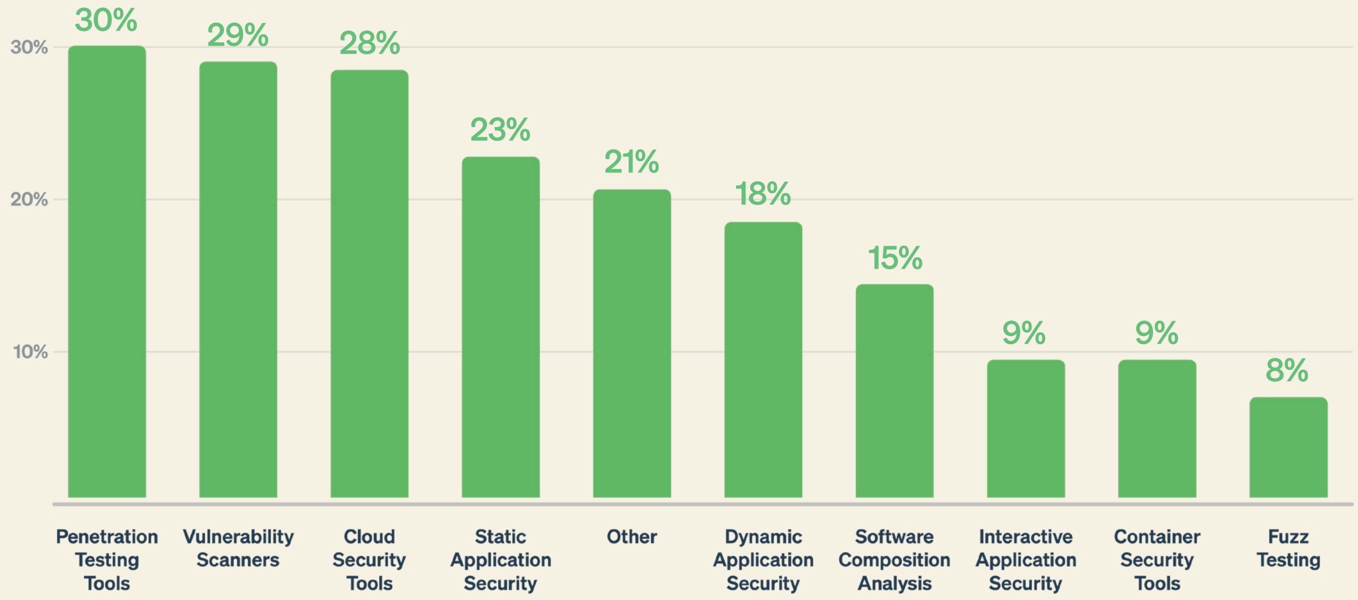
Among those who do follow compliance or regulatory standards:

- **ISO 9001 (27%)** is the most commonly cited, highlighting the importance of quality management frameworks.
- **ISO/IEC 27001 (18%)** and **HIPAA (17%)** follow closely, reflecting a strong focus on information security and healthcare-specific compliance.
- **OWASP (13%), GDPR/CCPA/LGPD (12%)**, and **SOC 2 (10%)** also appear frequently, pointing to growing awareness of data privacy and application security.

Fewer than 10% of teams follow standards like **PCI DSS, SOX, FedRAMP, or FISMA**, suggesting that strict financial and government-related compliance remains more niche within QA teams. 7% selected "Other," often citing internal company policies, regional mandates, or industry-specific standards tailored to their sector.

These responses reflect the diversity of QA's role across regulated and non-regulated environments and underscore the need for flexible, adaptable testing processes that can scale with compliance demands.

What security tooling are you using in the QA process?



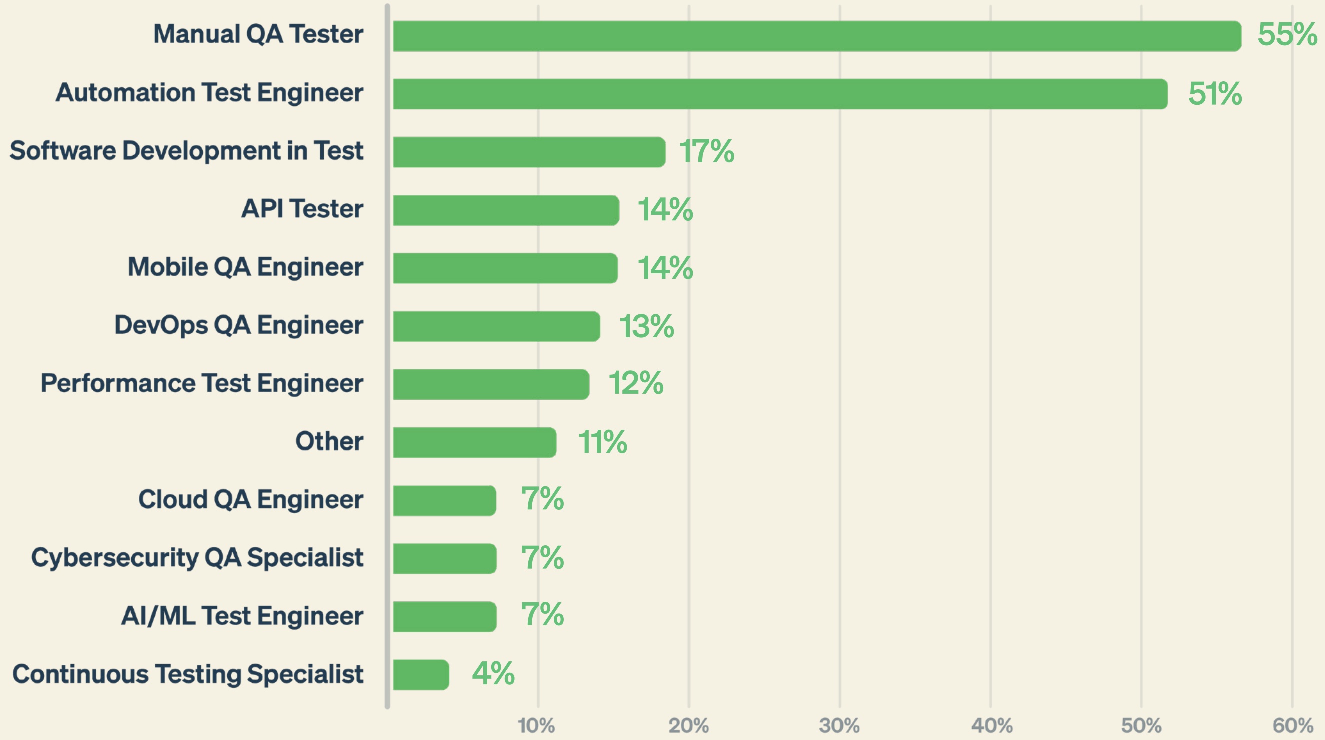
While some teams are integrating security tools directly into their testing workflows, others rely on separate departments or third-party providers.

- 30% of teams use **penetration testing tools**, making them the most commonly adopted security tool in QA.
- 29% use **vulnerability scanners**, focusing on identifying weaknesses before deployment.
- 28% have adopted **cloud security tools**, reflecting the widespread shift to cloud infrastructure.
- Static application security testing (SAST) is used by 23%, showing growing interest in identifying code-level vulnerabilities early.
- **Dynamic application security testing (DAST)** is used by 18%, indicating some investment in runtime security testing.

Notably, over 30% of respondents either don't use security testing tools or rely entirely on separate teams or external vendors for security efforts. Additionally, 21% selected "Other," with many clarifying that QA doesn't directly own or manage security.

These responses suggest that while security tooling is gaining traction in QA, it's still not a universal practice, especially in organizations where security responsibilities are handled outside of the QA function.

Which QA roles did you hire for in 2024?



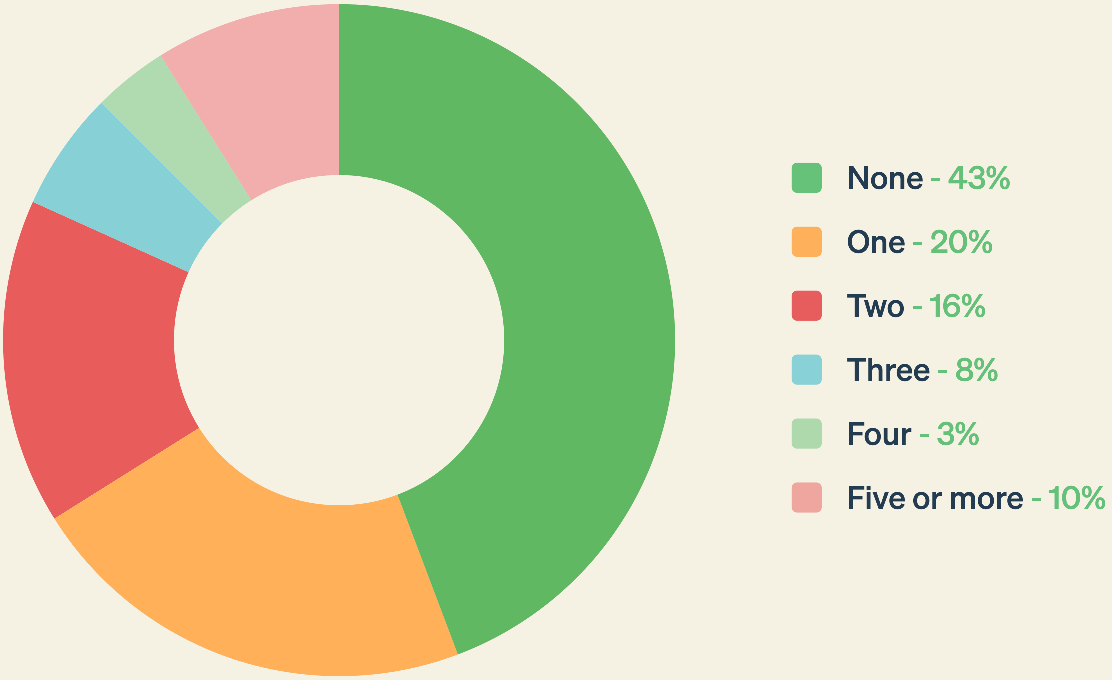
Hiring trends in QA continue to center around core testing roles, with a strong emphasis on both manual and automation skills.

- **Manual QA testers** were the most commonly hired role in 2024, selected by 55% of respondents.
- **Automation test engineers** followed closely at 51%, reinforcing the demand for scalable, automated testing practices.
- **Software Development Engineers in Test (SDETs)** (17%) were also hired by some teams, particularly those with more advanced or mature automation frameworks.

Emerging roles like **cybersecurity QA specialists (7%)** and **AI/ML test engineers (6%)** are gaining some traction, though they remain relatively niche.

Nearly 11% of organizations reported no QA hiring activity in 2024, citing hiring freezes or even reductions in QA headcount. This suggests that while essential roles remain in demand, economic pressures and shifting team priorities are influencing hiring decisions across the board.

How many external organizations do you partner with to help with testing your product(s)?

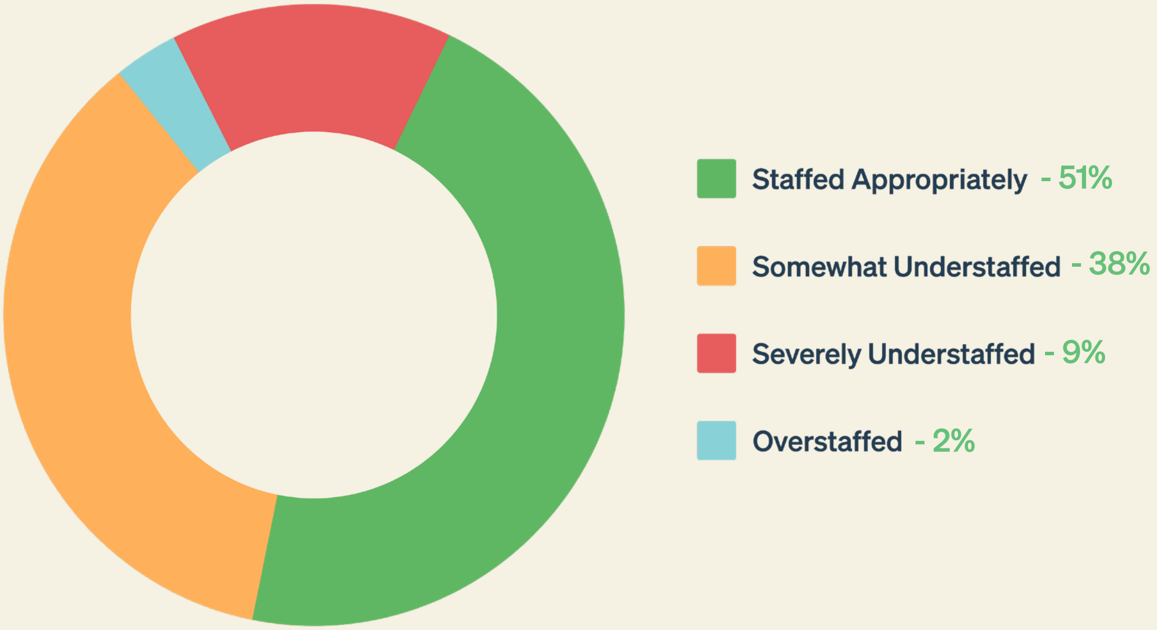


QA resourcing strategies vary, but a majority of teams partner with at least one external resource to accomplish their goals.

- 43% of respondents said they **don't partner with any external organizations**, indicating that many teams rely entirely on in-house QA resources.
- 20% work with **one external partner**, often for specialized testing like performance, compliance, or security.
- 16% partner with **two organizations**, reflecting a more blended model of internal and external support.
- 21% of teams partner with **three or more** external partners, respectively, suggesting deeper outsourcing models often tied to **complex or highly regulated environments.**

These findings show that while many QA teams operate independently, the majority (57%) of QA teams rely on third-party support to scale or specialize their testing efforts.

Please describe the staffing level of your QA team to accomplish your tasks and goals.



Staffing remains a key challenge for nearly half of QA teams.

- Only 51% of respondents said their teams are **appropriately staffed** to meet current testing needs.
- 47% of respondents reported that their team is either **somewhat or severely understaffed**, with 38% of those respondents citing **moderate resource gaps** and 9% reporting **severe understaffing.**

These numbers reflect ongoing constraints in QA resourcing. While some teams are adequately resourced, nearly half continue to face pressure to do more with less, impacting everything from test coverage to release velocity.

Understaffed teams report:

- Higher defect leakage
- Slower release cycles
- Lower automation adoption

A balanced mix of manual testers, automation engineers, and performance/security testers with well-defined roles and a focus on continuous skill development creates a stronger QA team.

What are your challenges related to finding new QA team members?

Hiring QA talent continues to be a complex challenge, shaped by technical skill gaps, a competitive market, and internal organizational constraints.

1. Skill and Experience Gaps

- Many candidates lack the technical depth required for modern QA roles, especially in automation, CI/CD, and AI-driven testing.
- Domain expertise and non-functional testing skills (like performance, security, or accessibility testing) are also hard to find.
- A lack of coding knowledge further limits candidates' ability to contribute to test automation.

> **"Finding candidates with the right mix of automation expertise, domain knowledge, and AI-driven testing experience is challenging."**

2. Competitive Market Conditions

- High demand for skilled QA professionals has made recruitment highly competitive.
- Budget limitations often prevent companies from offering competitive salaries, and many experienced testers move on quickly, either to bigger companies or roles outside QA.
- Hiring for on-site positions can also restrict access to top talent.

> **"Retention is a major issue. Many QA professionals use testing as a stepping stone to development or AI roles."**

3. Organizational and Process Challenges

- Internal barriers like slow hiring cycles, long onboarding processes, or a lack of internal QA investment can stall growth.
- In some companies, QA is undervalued or deprioritized, with leadership expecting developers to shoulder more testing responsibilities.
- Verifying "real-world" QA skills beyond resumes and interviews can be challenging, leading to hires who struggle in their roles despite appearing qualified during the hiring process.

> “Assessing QA skills in an interview setting is difficult—many candidates seem qualified on paper but struggle with real-world testing scenarios.”

These challenges highlight a disconnect between the evolving demands of modern QA and the realities of today’s hiring market, placing additional strain on teams that are already stretched thin.

Section Summary

This section provides a detailed view of how QA teams are structured, how they work, and the shifting pressures they face. The data reflects a landscape that balances established practices with ongoing transformation.

Manual testing remains a cornerstone of QA workflows, particularly for functional, exploratory, and regression testing. At the same time, more teams are investing in automation to improve speed and efficiency.

While most teams operate on a **structured release cadence** (weekly, bi-weekly, or monthly), year-over-year data shows a clear trend toward faster deployments. Daily releases grew from 6% to over 9%, potentially signaling increased adoption of DevOps and CI/CD practices, especially among mature and agile teams.

Satisfaction with QA processes strongly correlates with **early involvement, automation maturity, and overall efficiency**. Teams that emphasize these areas tend to report better outcomes and fewer production issues. Conversely, teams facing challenges with CI/CD integration or limited resources experience more friction.

Security and compliance approaches differ widely. Many QA teams **do not have dedicated security roles or tools** and often depend on DevOps, external vendors, or other departments. However, tools like **penetration testing, SAST, and vulnerability scanners** are becoming more common.

Staffing continues to be a pain point. Most QA teams rely on **in-house resources**, with manual testers and automation engineers being the most frequently hired. **But hiring freezes, skill gaps, and retention issues** remain a challenge, particularly as demand grows for expertise in automation, AI, and performance testing.

Overall, the QA landscape is evolving. Teams are moving faster, aiming for broader test coverage, and increasing collaboration across functions. As a result, **smarter processes and better-integrated workflows** are more important than ever.

SECTION 04

Testing Tools and Technologies

Section Introduction

As technology evolves, so does the SDLC. QA teams are constantly facing pressure to deliver higher-quality software faster—a challenge that is impossible to meet with manual testing alone.

Testing processes and practices such as test automation, CI/CD, and now AI and ML can help QA teams speed up their testing cycles and meet such demands. However, this technology isn't a magic solution—its effectiveness depends on how well it is integrated and utilized.

This section of the report provides an overview of the testing tools and technologies that teams are currently using to streamline their QA processes, and how effective they feel those tools have been.

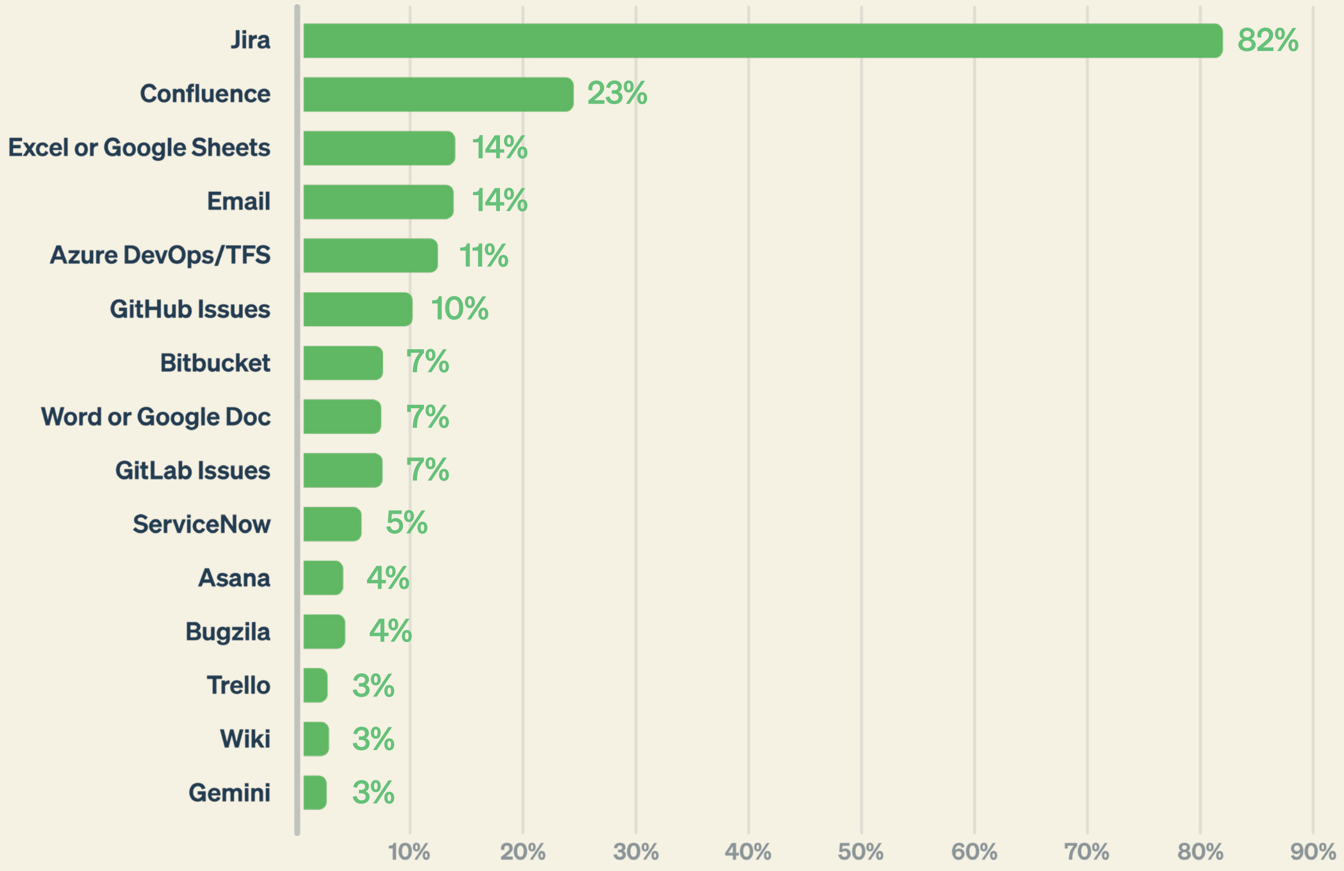
This section is divided into three key areas:

Tools, Technologies, and Integrations:
An exploration into defect trackers, automation frameworks, and CI/CD tools that survey respondents are currently using—and how integrated they feel their QA tech stack is.

Test Automation:
A review of the types and volume of automated tests that respondents are running, as well as their goals and challenges for test automation in 2025.

AI in QA:
A snapshot of current AI adoption, including how AI is being used, where it's making an impact, and what barriers still exist.

What tool does your team use to track defects/bugs?

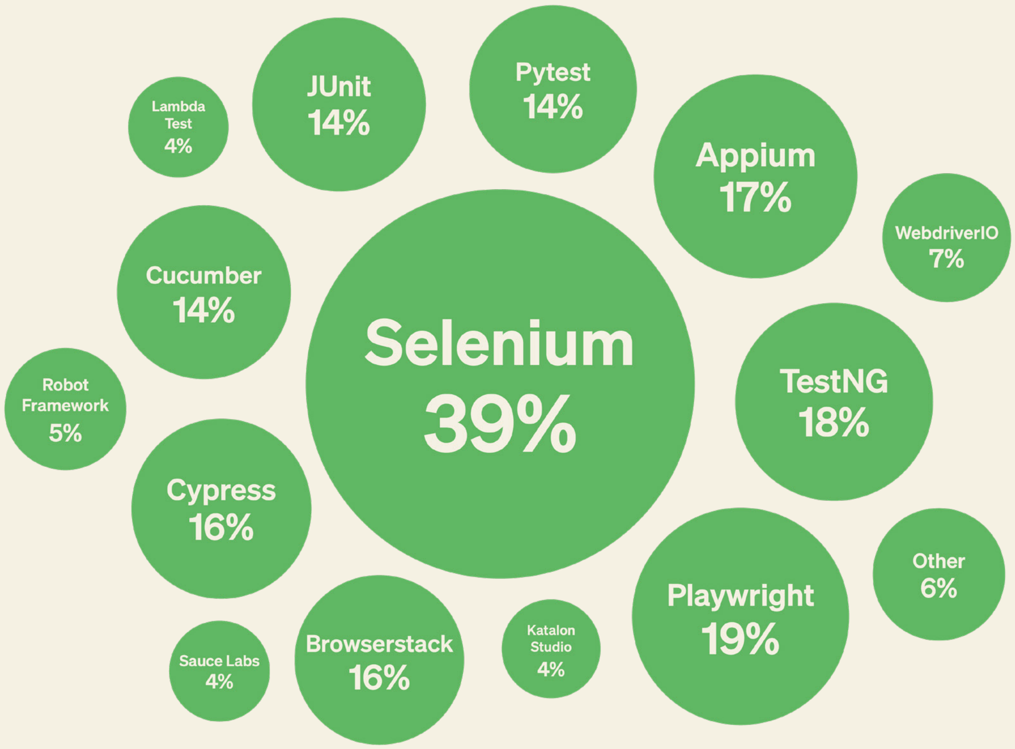


The vast majority of surveyed teams (82%) use Jira for defect and bug tracking, making it the clear industry standard. This dominance suggests that Jira's integration with Agile, DevOps, and CI/CD pipelines makes it a preferred choice for structured and automated workflows.

Other widely used tools include Confluence (23%), Azure DevOps/TFS (11%), GitHub Issues (10%), GitLab Issues (7%), and Bitbucket (7%).

- **71%** of teams with strong traceability **(integrations enabling them to link requirements → tests → defects → fixes)** report fewer escaped defects than those without such integrations.

What test automation tools, suites, or frameworks do you use?

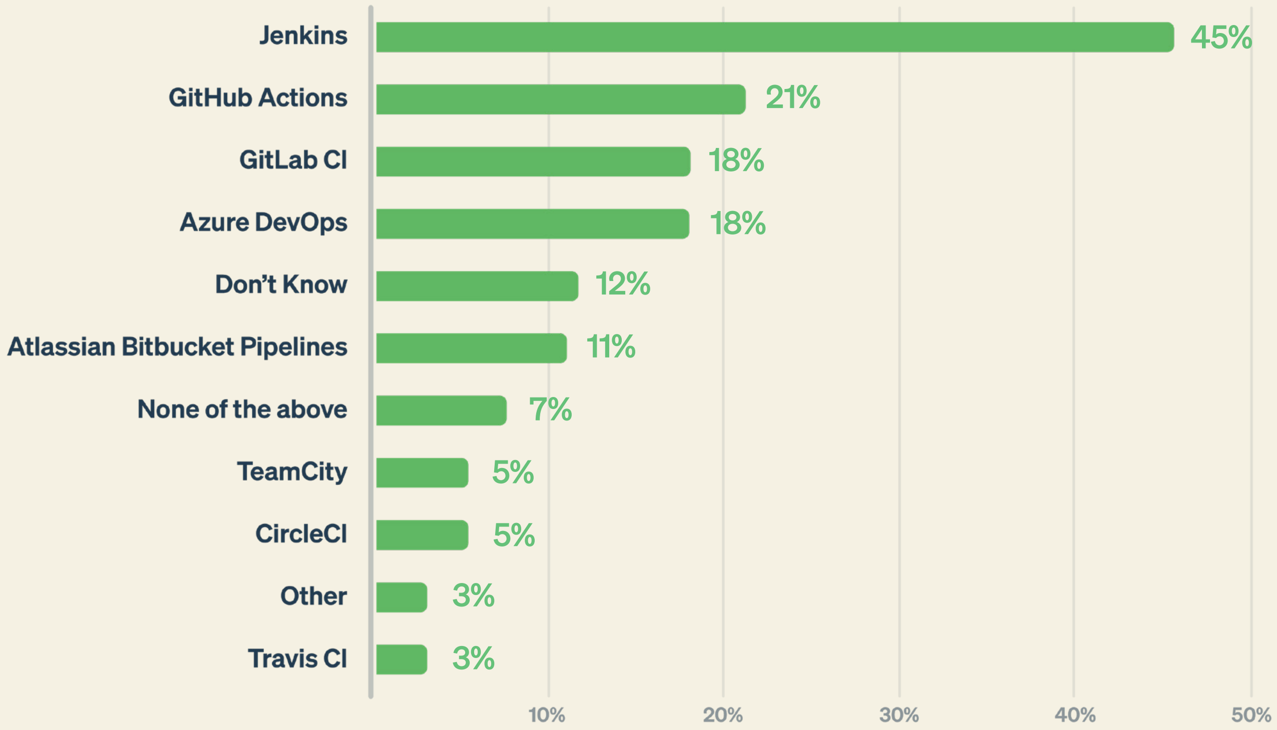


The survey data highlights a diverse landscape of test automation tools, although QA professionals continue to prioritize open-source frameworks. Selenium remains the most widely used automation tool, chosen by 39% of respondents. Playwright (19%) follows as the second top choice, particularly for modern web testing, then TestNG (18%) for Java-based testing.

- 67% of teams with **fully automated test execution pipelines** (via Jenkins, GitLab, Azure DevOps, etc.) report that **test execution time is no longer a bottleneck.**
- **Teams without automation report a 3x slower execution cycle** on average.

> **“Before automation, we relied heavily on manual testing, which slowed us down. Now, 80% of our tests are automated, and our release velocity has improved dramatically.”**

Does your team use a continuous integration/continuous deployment (CI/CD) tool in your development process?

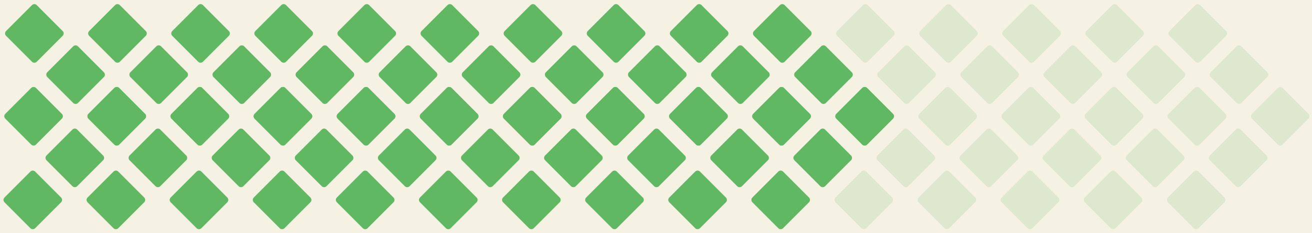


There's a **strong adoption of CI/CD tools** within software development and testing processes, with some clear and consistent leaders in the industry. Jenkins maintains its position as the most popular tool among respondents for the past five years, and its adoption rate continues to grow, with 45% still selecting it as their preferred CI/CD tool. **GitHub Actions** and **GitLab CI** are still the second and third most popular tools, respectively.

- **86% of teams with high test automation and CI/CD integration report faster release cycles** compared to only 42% of teams with low automation.
- **71% of teams with high test automation and CI/CD integration see reduced defect leakage,** compared to just 35% of low-automation teams.
- **58% of teams using automated test execution tied to CI/CD report a measurable return on investment (ROI)** in automation within 6 months.

> **"Our CI/CD pipeline reduced our deployment time by half. We catch regressions much earlier, and our defect leakage dropped significantly."**

How integrated are the different tools you use for QA testing?

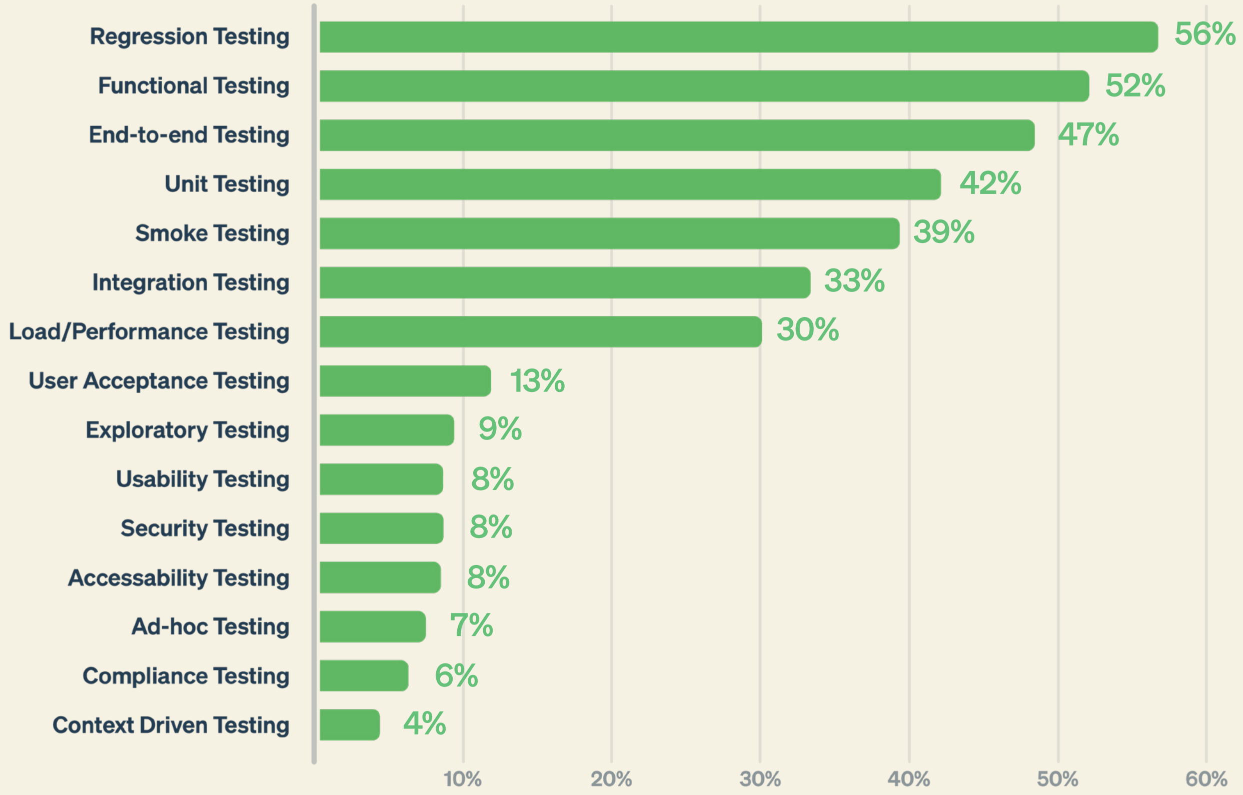


Survey respondents rated their tool integration

62 out of 100

The QA tech stack is becoming increasingly integrated, but there's still room for improvement. On average, respondents rated their tool integration 62 out of 100, suggesting a moderate level of integration. While many teams have some level of integration between their QA tools, most haven't yet achieved full automation or seamless end-to-end integration.

What kinds of testing does your team run with test automation?



Survey responses show that most teams focus their automation efforts on test types that deliver high impact and are easy to repeat.

Regression testing is the most commonly automated category, with 56% of teams using automation to ensure that core functionality remains stable across releases.

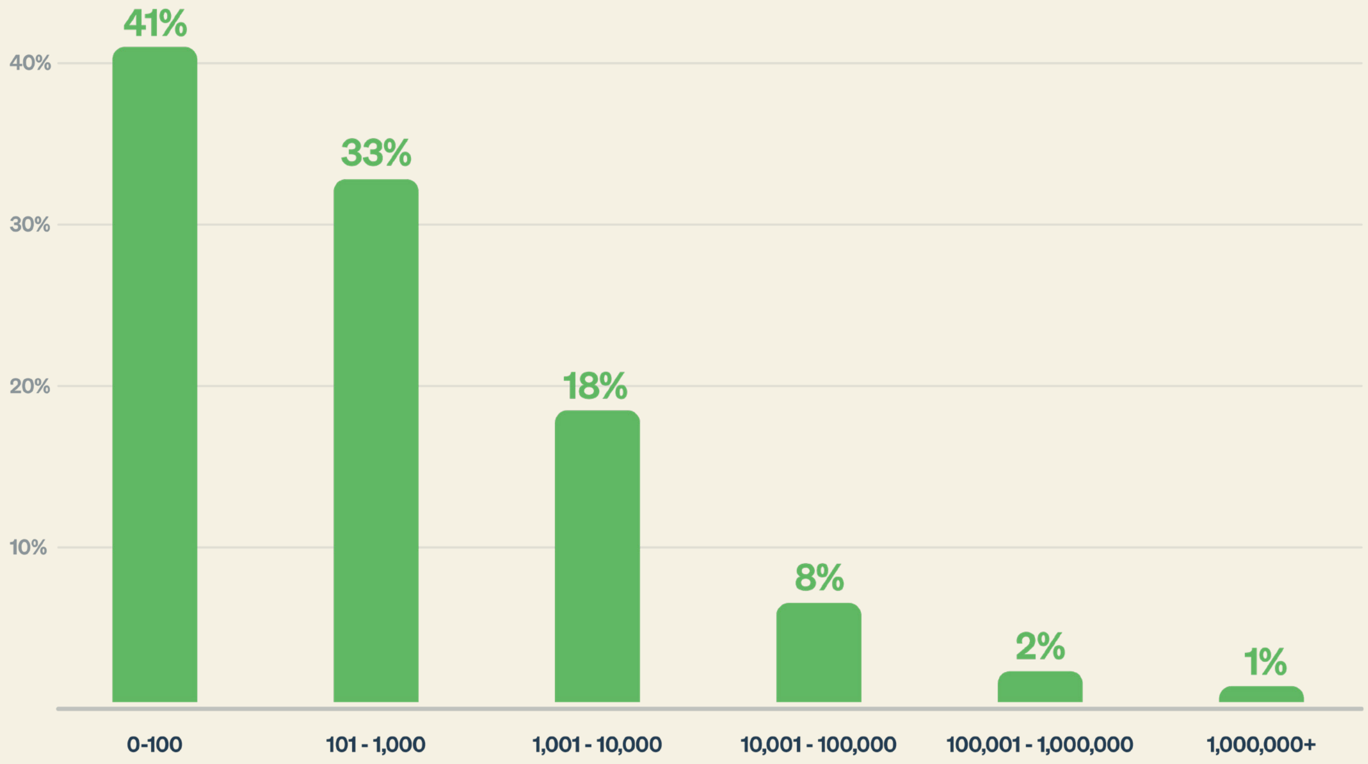
Functional testing follows closely at 52%, as teams work to validate that key features behave as expected with each new build.

About 47% of teams automate **end-to-end testing** to ensure that complete workflows and integrated systems function properly. **Unit testing** is also widely used, with 42% of teams automating it to catch issues early in development.

Other types of testing, like **smoke tests** (40%) and performance testing (30%), are automated by a notable number of teams. Smoke tests are often used to quickly validate builds, while performance testing tends to rely on specialized tools and infrastructure.

That said, more **complex or** exploratory testing, like usability, accessibility, security, and compliance, remains mostly manual. These areas often require human insight and more refined contextual understanding.

On average, how many automated tests does your organization run each day?



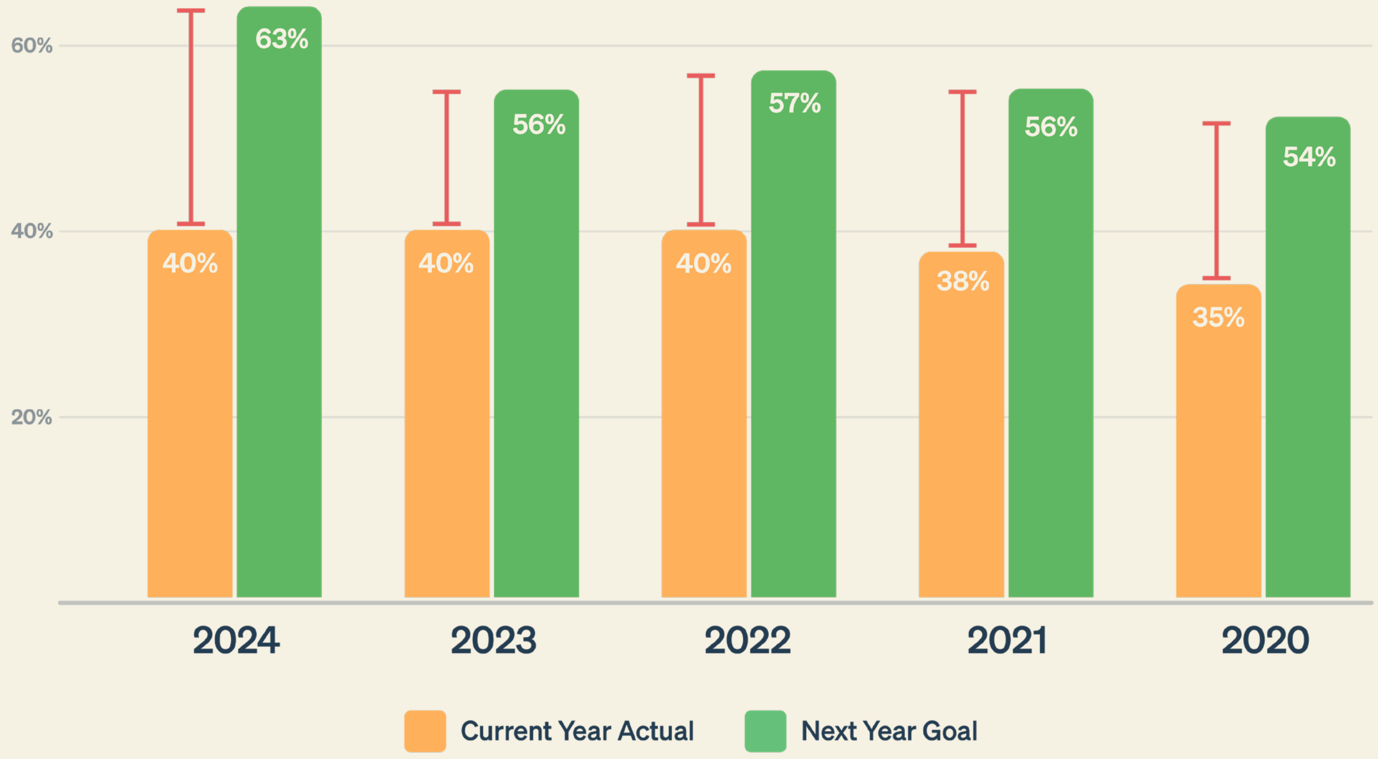
The number of automated tests run each day varies widely across organizations, reflecting different levels of automation maturity and infrastructure.

Most teams are still in the early to mid stages of scaling. About **41% run fewer than 100 tests per day**, while **another 33% run between 101 and 1,000**. This suggests that many QA teams are building up their automation capabilities but haven't yet reached full scale.

About **18% of teams run between 1,001 and 10,000 tests per day**, indicating more mature automation pipelines. Just over **8% run more than 10,000 tests daily**, including a small subset of **3% that execute more than 100,000 tests per day**. These are typically large enterprises with advanced CI/CD systems and high-volume testing needs.

While only a few organizations operate at this level of scale, it highlights what's possible with the right investment in infrastructure, tooling, and test design. For most teams, reaching this level remains an aspirational goal as they continue to expand automation across the SDLC.

What percentage of your tests do you expect to be automated vs. manual in the next year?



Teams expect automation to make up a larger share of their testing in the coming year. On average, respondents anticipate that **63% of their tests will be automated.**

This points to a continued push toward automation maturity, but adoption still varies. Some organizations continue to rely on manual testing, often due to legacy systems, complex user flows, or strict compliance requirements that make automation more challenging.

When looking at year-over-year data, the trend is clear: teams are moving steadily toward greater automation to improve speed, efficiency, and scalability.

What are your pain points and challenges in test automation?

Teams are making progress with automation, but it's not without challenges. Here are the top pain points in test automation:

45%

Frequent test breakages

UI changes and frequent application updates often cause automated tests to fail, leading to high maintenance costs and reduced reliability.

32%

Test data management

Creating and maintaining test data is time-consuming, and many teams lack streamlined processes to support fast, consistent automation.

30%

Lack of skilled personnel

A shortage of experienced automation engineers makes it harder for teams to build and maintain reliable test suites.

29%

Tool selection challenges

With so many tools on the market, finding one that fits your tech stack and application complexity is a common struggle.

26%

Flaky or unreliable tests

Inconsistent test results slow teams down and make it harder to trust automation as part of the CI/CD pipeline.

26%

Inadequate test coverage

Many teams say they still can't automate enough of their testing, leaving gaps in their quality assurance strategy.

21%

Slow test execution

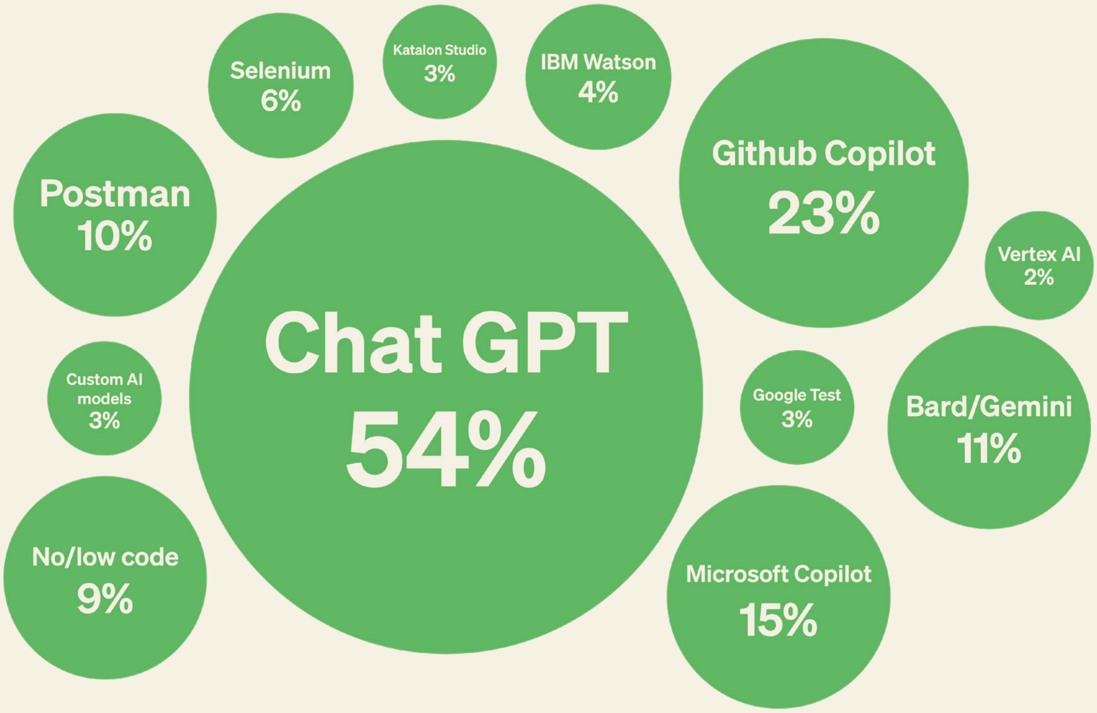
Long test runs can create bottlenecks in release workflows, making it harder to maintain fast feedback cycles.

19%

Limited CI/CD integration

Some organizations are still working on better connecting their automation with DevOps tools and workflows.

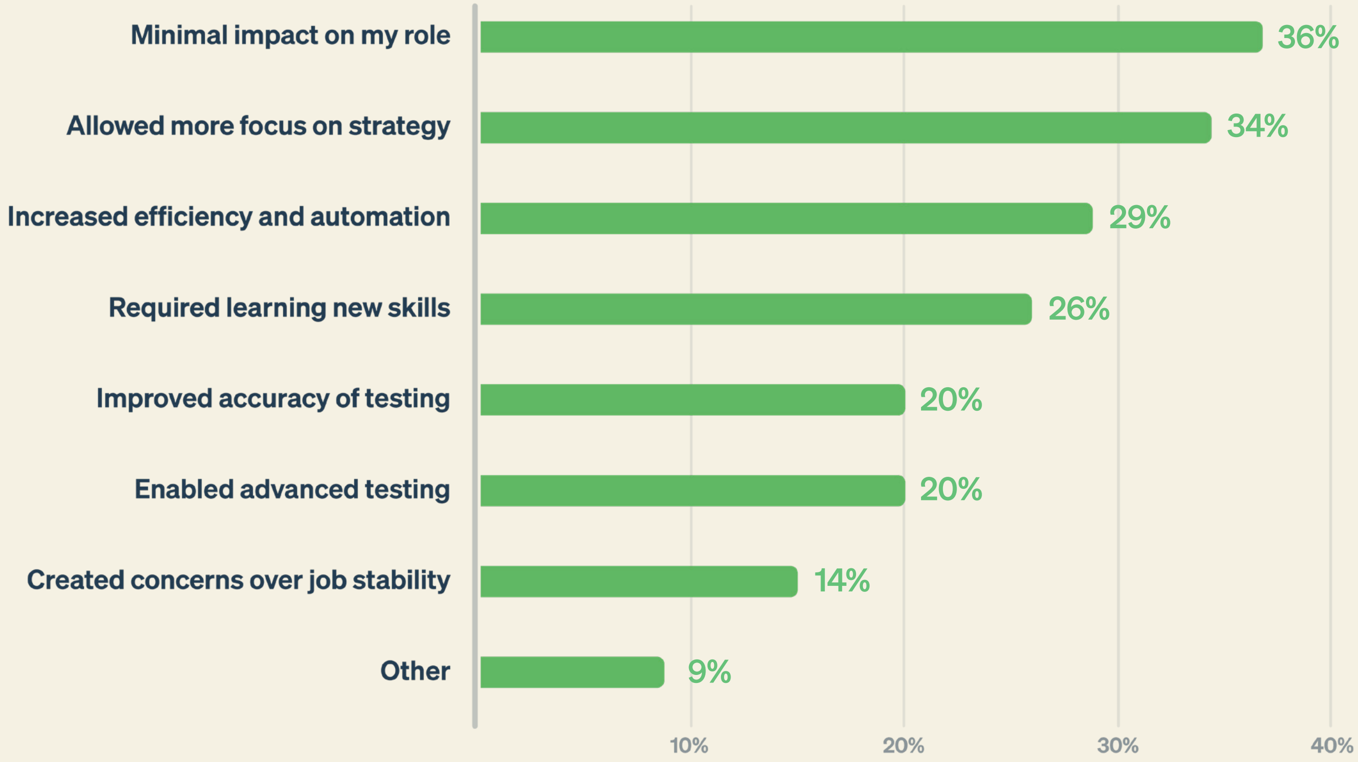
Do you currently leverage AI tools in your existing QA processes? If yes, which?



AI tools are becoming increasingly integrated into QA workflows, with most respondents leveraging **ChatGPT (54%)** and **GitHub Copilot (23%)** for support, especially when it comes to test generation, debugging, and automation assistance. However, adoption across **specialized AI-driven test automation and QA tools** remains relatively **low and fragmented**, suggesting more industry education and tool maturity are needed.

AI-powered defect tracking and management is also underutilized, leaving opportunities for AI-driven predictive analytics and auto-triage to grow in importance.

How has AI impacted your role in QA?



AI adoption in QA has delivered mixed results. While it's helping teams work more efficiently by automating repetitive tasks, it hasn't dramatically changed most QA roles. The most significant change is a shift toward more strategic tasks, with AI automating repetitive work.

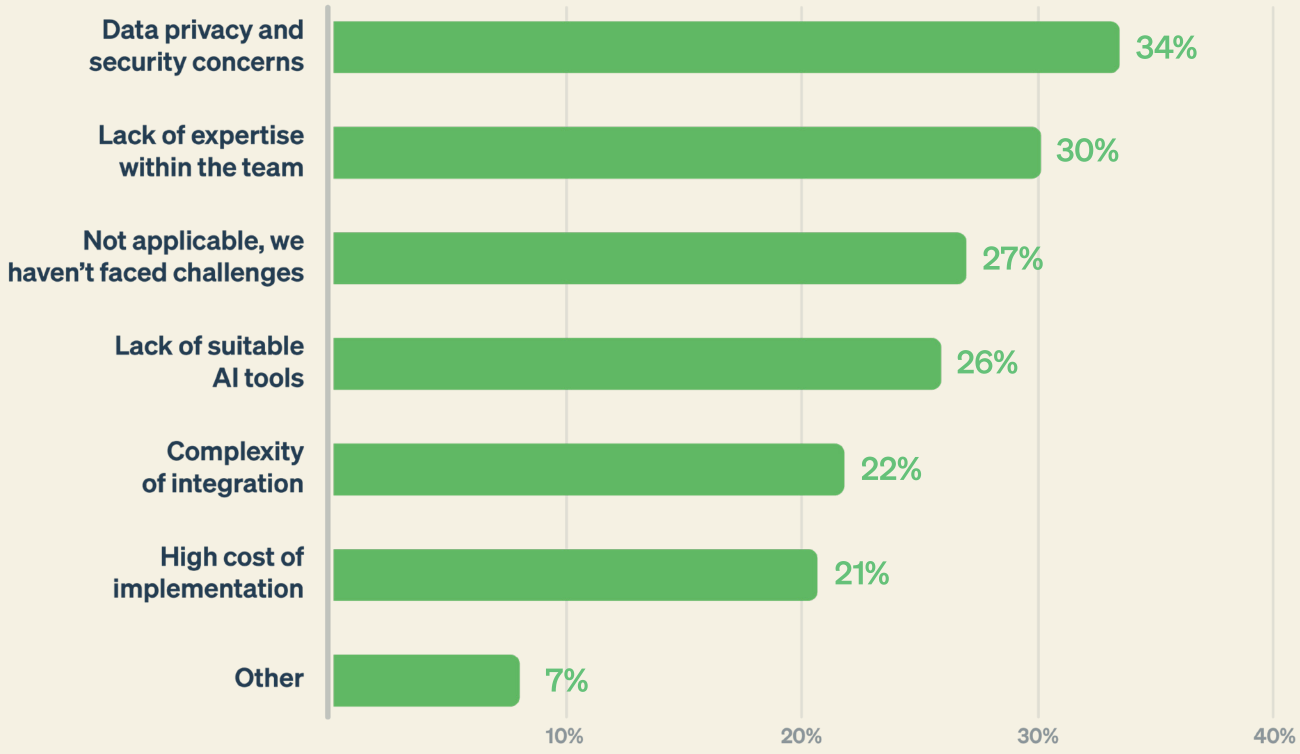
1. Efficiency Gains and Strategic Focus:
34% of respondents say AI allowed them to focus on strategic and complex tasks rather than repetitive test execution, and 29% reported increased efficiency and automation in testing.

2. Minimal Impact and Resistance to AI Adoption:
For 36% of survey participants, AI has had minimal impact on their role. Meanwhile, many in the "Other" category also noted that their organizations have not yet implemented AI or do not allow its use, indicating limited practical use cases.

3. Skills and Job Security Concerns:
Although AI has become increasingly popular, there's still a need for training and adaptation, as 26% of respondents have said AI required them to upskill or learn new tools. The fears about AI replacing human testers also remain present, with 14% reporting concerns over job stability.

4. Advanced AI Use Cases (Still Emerging):
20% of survey participants say AI has enabled advanced testing techniques like predictive analytics and anomaly detection, though adoption remains limited.

What challenges have you faced integrating AI into your QA testing process?



As QA teams explore AI-driven solutions, they face various challenges in integration. The most frequently cited obstacles include **data privacy and security** concerns (34%), followed by a **lack of expertise** within the team (30%). Additionally, teams struggle with finding suitable AI tools (26%), the **complexity of integration** (22%), and the **high cost of implementation** (21%).

Interestingly, **27% of respondents reported no significant challenges**, indicating that AI adoption is smoother for some teams.

What do you see as the biggest opportunity for AI in QA in the next 5 years?

AI continues to shape the QA landscape, with professionals expecting significant opportunities. The top anticipated advancements include self-healing test scripts, AI-powered test case generation, and predictive defect analysis. Many respondents also highlighted improved efficiency in regression testing, intelligent test prioritization, and automated exploratory testing as areas where AI could have a profound impact.

1. AI-Driven Test Automation and Self-Healing Test Scripts

- AI will **enhance test automation** by automatically generating and maintaining test cases, reducing manual effort and ensuring continuous test execution.
- **Self-healing scripts** will adapt to UI and functionality changes, eliminating the need for frequent manual updates.

> **“Automated test generation and execution will be the biggest breakthrough, saving time and ensuring higher test coverage.”**

2. Predictive Defect Analysis and Risk-Based Testing

- AI will **analyze historical test data and predict high-risk areas** in applications, enabling teams to focus on preventing defects rather than just finding them.
- Machine learning models will **prioritize test cases** based on defect history and code changes, making testing more efficient and targeted.

> **“Predictive analytics will help us focus testing on critical areas, reducing time spent on unnecessary test cases.”**

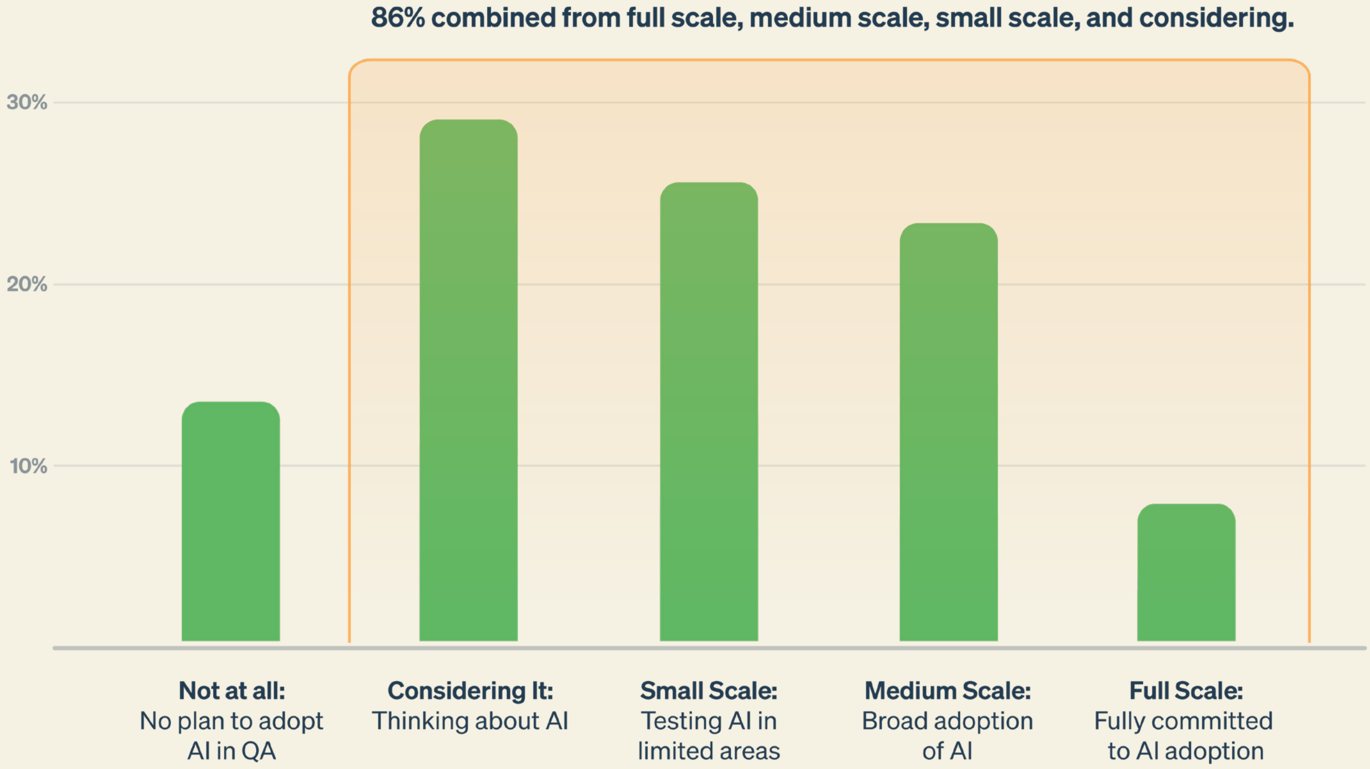
3. AI-Assisted Test Case and Script Generation

- AI will **generate manual and automated test cases** from requirements, user stories, or system logs, making test creation faster and more scalable.
- AI-powered tools will **convert manual test cases into automated scripts**, speeding up regression testing and improving coverage.

> **“The ability for AI to create comprehensive test cases based on user behavior will be a game-changer.”**

Although there’s excitement around the notion of AI in QA, concerns remain about data privacy, security, and job displacement, alongside challenges in ensuring AI-generated test cases are reliable and contextually accurate.

How would you rate your plans to increase the usage of AI capabilities in QA over the next year?



A vast majority of respondents (86%) indicate at least **some level of consideration or active implementation of AI in QA**, while only 14% have no plans to adopt it.

The largest segment (29%) is still in the **consideration phase**, suggesting that AI in QA is at a tipping point where many organizations are exploring its potential but haven't yet taken concrete steps. A smaller but significant 9% are **fully committed to AI**, signaling a strong shift toward automation and predictive QA methodologies.

Section Summary

This section highlights the tools and technologies QA teams are adopting and how they rate their impact on testing processes. Despite the ubiquity of artificial intelligence, **AI-powered tools still haven't gained much of a foothold in testing**, with many teams still struggling to fully integrate more established processes such as test automation and CI/CD.

This year's survey didn't find many shake-ups in the landscape of testing tools, with the vast majority of teams still using Jira for defect tracking and preferring open-source automation tools such as Selenium and Playwright.

There continues to be **strong adoption of CI/CD tools**, with Jenkins leading the pack as the industry leader ahead of GitHub Actions and GitLab CI. However, despite such strong adoption levels, survey respondents **rated their overall tech stack integration level at 62%**—suggesting that there is still ground to cover in this journey.

Automation continues to be a key part of the modern QA workflow, with respondents commonly reporting that they automate regression, functional, end-to-end, unit, smoke, and performance testing. At the same time, the majority of teams still run fewer than 1,000 automated tests per day, indicating that most QA teams are still in the **early-to-mid stages of scaling their automation programs.**

Those same teams are **aggressively pushing to scale automation** in the coming year, however, with respondents anticipating that an average of 63% of their tests will be automated in 2025. The top pain points preventing QA teams from reaching this goal are frequent test breakage, test data management, a lack of skilled personnel, and problems with finding the right automation tools.

And finally, the topic on everyone's mind: **artificial intelligence**. Despite seeming like AI is now integrated into every app, appliance, tool, and platform you encounter, its adoption among QA remains fragmented. While integration of AI tools into QA workflows is increasing, ChatGPT and GitHub Copilot are driving the majority of that movement. AI-driven test automation and QA tools are available, but their **adoption is low**, suggesting that more industry education and tool maturity are needed.

Regarding efficiency gains, teams report mixed results from integrating AI into their QA processes. While many respondents are reporting success in AI automating more repetitive tasks, many others still report minimal impact or an inability to utilize AI at their organization altogether. Regardless, respondents remain optimistic, expressing **hopes that advances in AI will transform test automation, defect analysis, and test case and script generation in the coming years.**

SECTION 05

Challenges, Priorities, and KPIs

Section Introduction

As QA teams adapt to increasingly fast-paced development cycles, their biggest hurdles and strategic goals continue to evolve. This section explores the key challenges teams are facing today, the metrics they use to measure success, and the priorities shaping their testing strategies for the future.

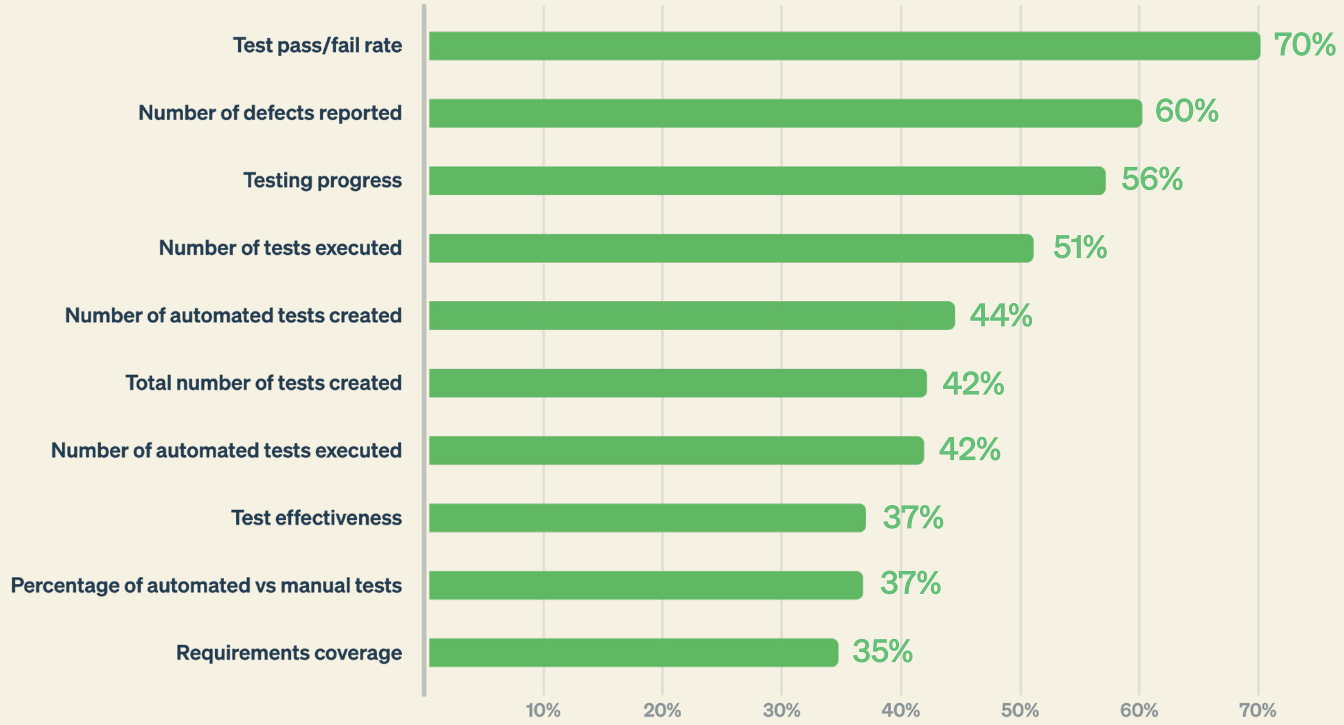
The data highlights a growing focus on automation, test coverage, and efficiency, paired with persistent pain points around test environment complexity, late-stage QA involvement, and resource constraints. At the same time, there's a noticeable shift toward aligning QA objectives with broader DevOps and Agile goals.

This section covers three key themes:

- **Testing Objectives, Challenges, and Priorities:**
  A breakdown of what QA teams are aiming to improve, the most common challenges they face today, and how those challenges align with broader testing goals.
- **Metrics and KPIs:**
  A look at the metrics teams currently track, which ones they wish they could measure, and what that says about how QA success is defined across the industry.
- **Future Planning:**
  Insights into what QA teams are prioritizing over the next 1 to 3 years, including automation, shift-left testing, performance, security, and better collaboration across teams.

From the KPIs teams track to the initiatives they're planning for the next few years, one thing is clear: modern QA teams are under pressure to deliver more value, faster, and they're responding with smarter tooling, earlier collaboration, and data-driven decisions.

Which metrics or KPIs does your team track and report on?

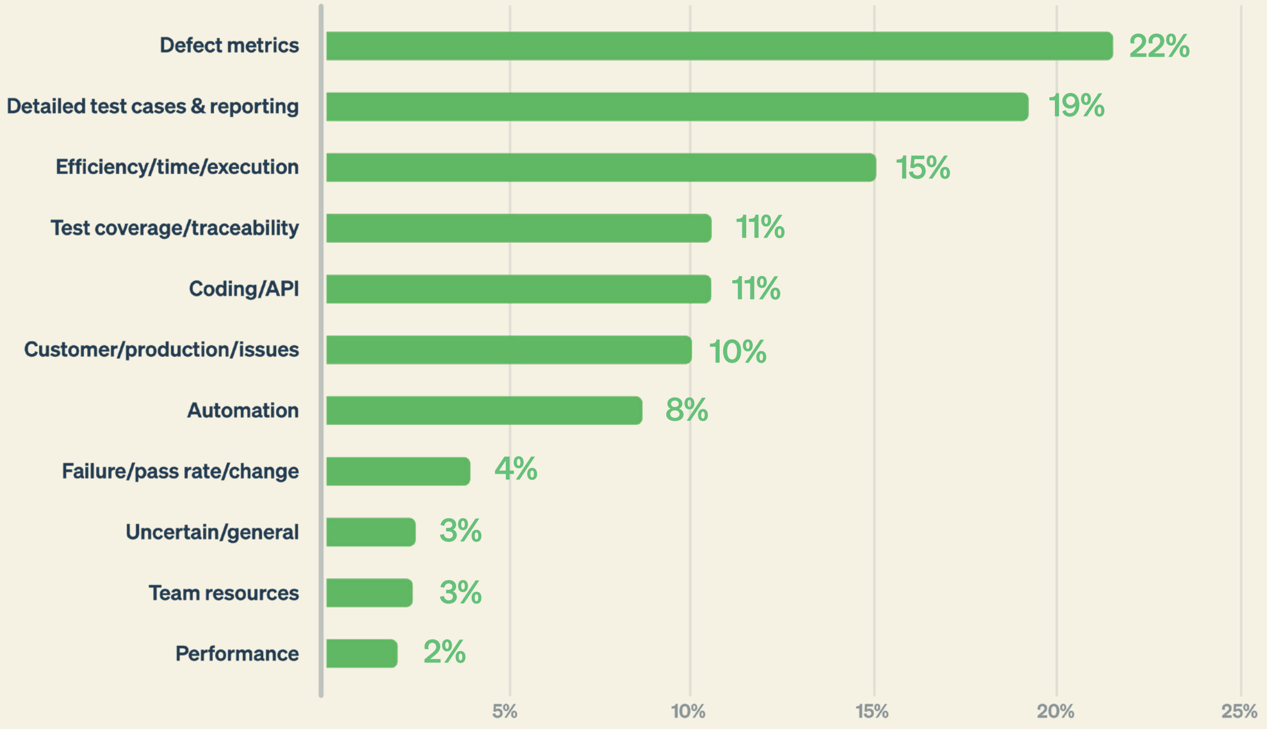


The data reveals that QA teams rely on a core set of metrics to measure testing effectiveness and software quality. These KPIs reflect a balance between operational visibility and outcome-driven tracking.

- **Test pass/fail rate (70%)** is the most widely tracked metric, serving as a foundational indicator of software stability and quality during test execution.
- **The number of defects reported in production (60%)** highlights the importance teams place on minimizing post-release issues and improving release reliability.
- **Testing progress (56%)** is also commonly monitored, allowing teams to track the status of tests (e.g., passed, failed, blocked) throughout a test cycle.
- **The total number of tests executed (51%)** offers a lens into testing activity and helps teams evaluate coverage over time.
- **Automated tests created (43%)** and **executed (42%)** show a strong interest in measuring the scale and impact of test automation efforts.

These findings indicate that while traditional QA metrics remain central to quality reporting, many teams are expanding their focus to include automation-specific metrics, showing its growing role in modern testing strategies.

Which metrics would you like to report on that you currently can’t track?



While most teams report tracking core metrics like pass/fail rates and production defects, many still feel they lack visibility into deeper or more strategic insights. When asked what they wished they could measure, responses covered a broad spectrum, but several recurring themes emerged.

The most desired untracked metrics fell into the following categories:

22%

Defect Metrics

Teams want a clearer understanding of defect trends, especially around cost, recurrence, and resolution time.

> **“We need to know the root cause of defects and the time it takes to fix them.”**

19%

Detailed Test Case Reporting

Many respondents called for better documentation and reporting of executed tests across sprints and modules.

> **“We do the testing, but we lack a proper way to document all the executed test cases in real time.”**

15%

Efficiency, Time, and Execution

Metrics related to test cycle time, regression suite speed, and overall time-to-completion were frequently cited.

> **“Need clearer time-to-complete data: how long from test assignment to final result.”**

Other common categories included:

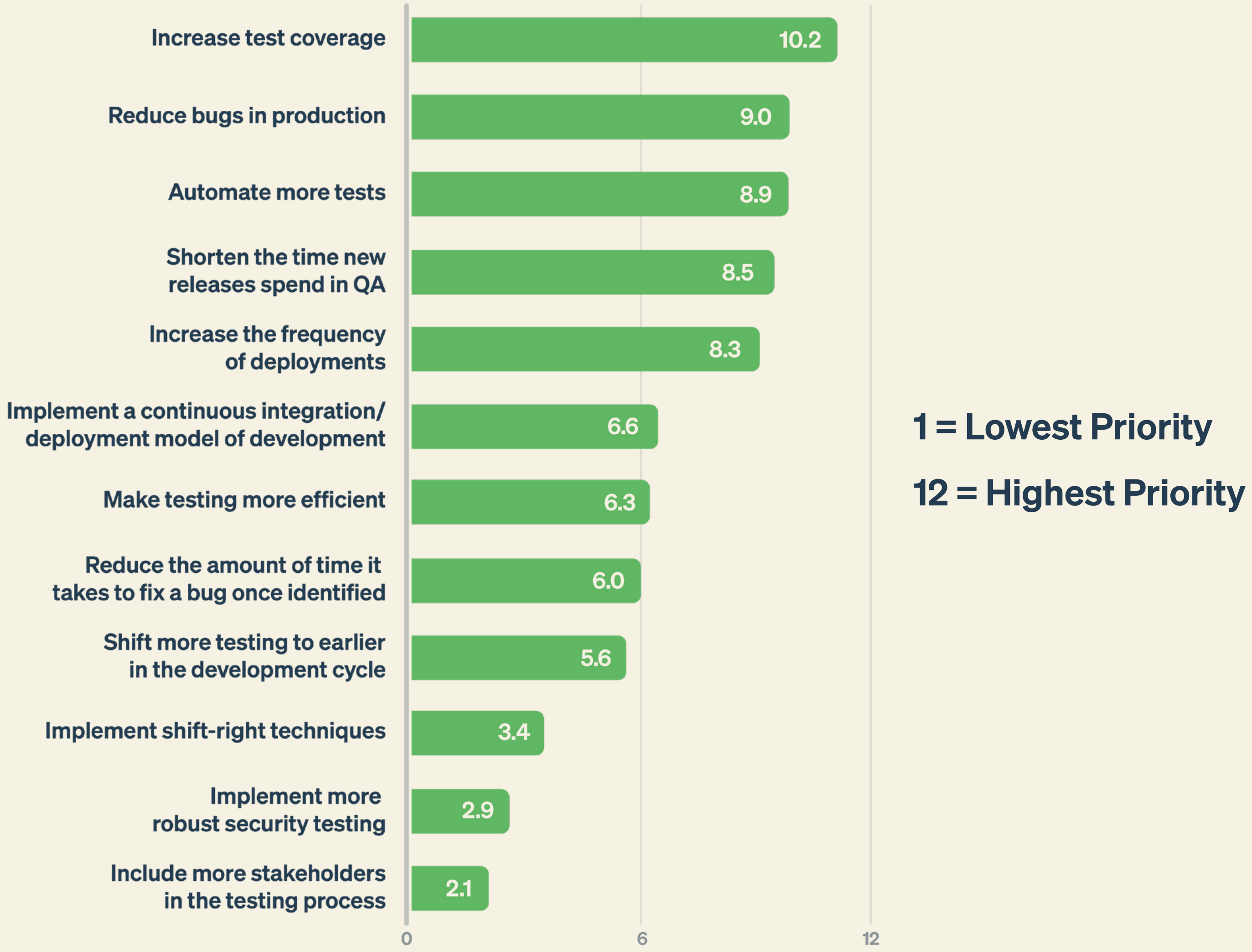
- Test Coverage and Traceability (10%)
- Coding/API Insights (10%)
- Customer or Production-Level Issues (10%)

Some teams also admitted they’re still figuring out what matters most to track:

> **“We don’t currently have a KPI framework, so we don’t know what we’re missing.”**

These responses show that many teams are ready to move beyond basic reporting. They want deeper, more meaningful metrics that connect their QA efforts to real outcomes like product quality, team efficiency, and customer impact.

What are your team's top objectives around testing and QA right now? (Ranked from 1 to 12)



QA teams are focusing on what matters most for product quality and delivery: **better coverage, fewer bugs in production, and more automation**. Speed and deployment frequency are also gaining momentum, reflecting increased alignment with Agile and DevOps methodologies. Meanwhile, strategies like shift-left and shift-right testing are still emerging priorities, not yet fully mainstream.

10.22

1. Increase test coverage (Score: 10.22)

This is the most critical objective for QA teams, signaling a strong push to validate more of the product across features and user flows. It aligns closely with other top-tracked KPIs like test pass/fail rate and total tests executed.

9.05

2. Reduce bugs in production (Score: 9.05)

A clear quality-first priority, this objective emphasizes catching issues before release. It's supported by the fact that 60% of teams actively track defects in production as a core KPI.

8.91

3. Automate more tests (Score: 8.91)

Automation remains a high priority for teams looking to reduce manual effort and scale testing. Nearly half of the respondents already track automated test creation and execution as key metrics.

8.51

4. Shorten the time new releases spend in QA (Score: 8.51)

Teams are working to streamline the QA phase to better support rapid release cycles, while still maintaining test coverage and quality.

8.29

5. Increase the frequency of deployments (Score: 8.29)

This objective reflects growing adoption of CI/CD workflows and continuous testing strategies aimed at faster, more reliable releases.

While traditional goals like team efficiency, collaboration, and traceability remain important, these results show that coverage, automation, and defect prevention are driving QA priorities in 2025.

Correlation Insights: QA Team Size vs. Testing Objectives

<20 = smaller teams

Smaller Teams

Smaller QA teams emphasize efficiency, deployment frequency, and reducing QA cycle times to keep up with fast releases.

+500 = larger teams

Larger Teams

Larger QA teams prioritize comprehensive test coverage, automation, and defect prevention over pure speed.

The survey data reveals a clear pattern: **team size plays a significant role in shaping QA priorities.**

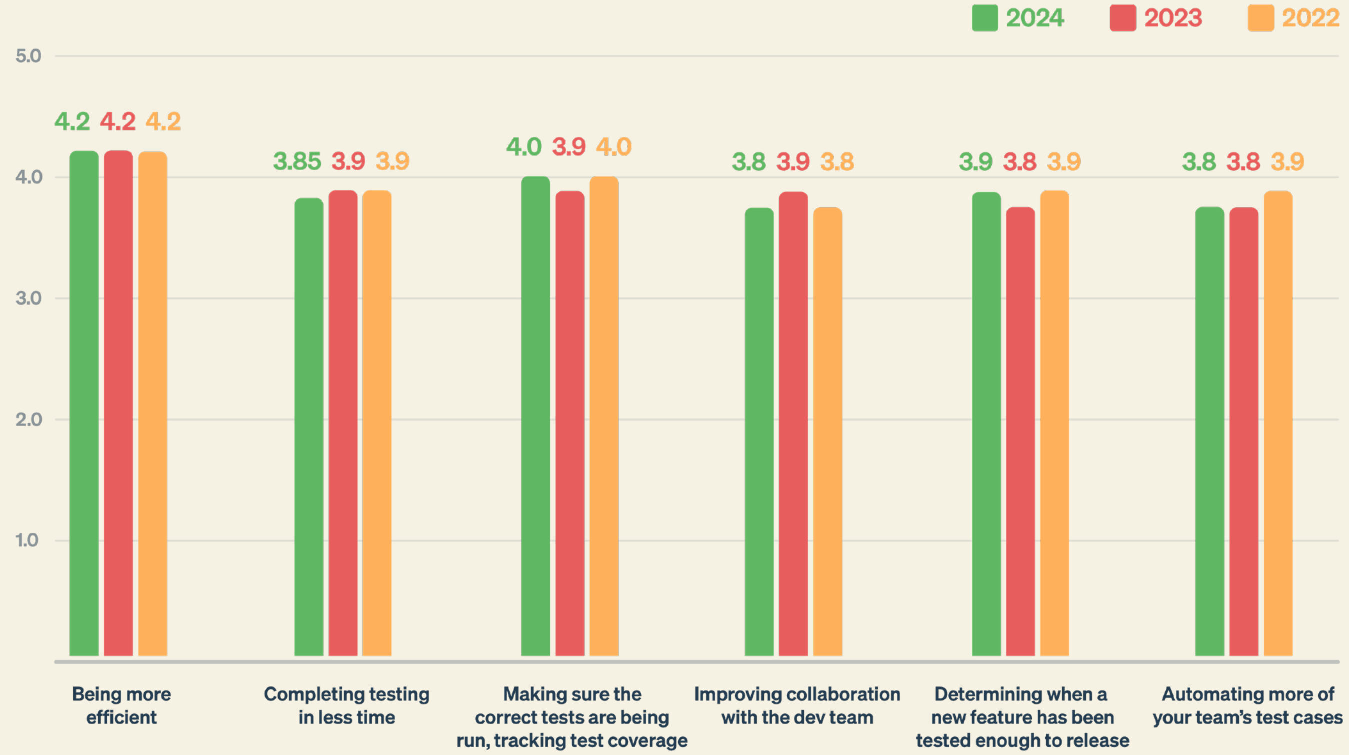
Larger QA teams are more likely to focus on **comprehensive test coverage, increasing automation, and preventing bugs before release.** These teams often have the resources to invest in deeper validation and long-term quality improvements rather than just meeting short-term delivery goals.

In contrast, smaller QA teams tend to prioritize **efficiency, deployment frequency, and reducing QA cycle times**—reflecting the need to move fast and do more with less. For these teams, speed is essential to keeping pace with development demands.

CI/CD adoption also shows a positive correlation with team size. Larger teams are more likely to adopt and scale CI/CD pipelines, whereas smaller teams may still be catching up due to resource or tooling limitations.

Across the board, strategies like shift-left and shift-right testing are gaining attention but haven’t yet become top priorities, regardless of team size. These forward-looking practices may grow in importance as teams mature and look to integrate QA more fully into the development lifecycle.

On a scale of 1 to 5, rate how important each of the following priorities is in your QA processes. (1 means not important and 5 means highest priority)



When asked to rate the importance of various QA priorities on a 1–5 scale, teams consistently emphasized efficiency, coverage, and collaboration over peripheral concerns like outsourcing or stakeholder involvement.

The top-rated priorities (weighted averages between 3.97 and 4.2) reflect a practical, quality-focused mindset:

1. Being more efficient (Average Score: 4.2)

Efficiency is the highest-rated priority across the board. QA teams are clearly focused on streamlining their workflows, reducing bottlenecks, and doing more with limited time and resources. This aligns with other top objectives like automation and shortening QA cycles.

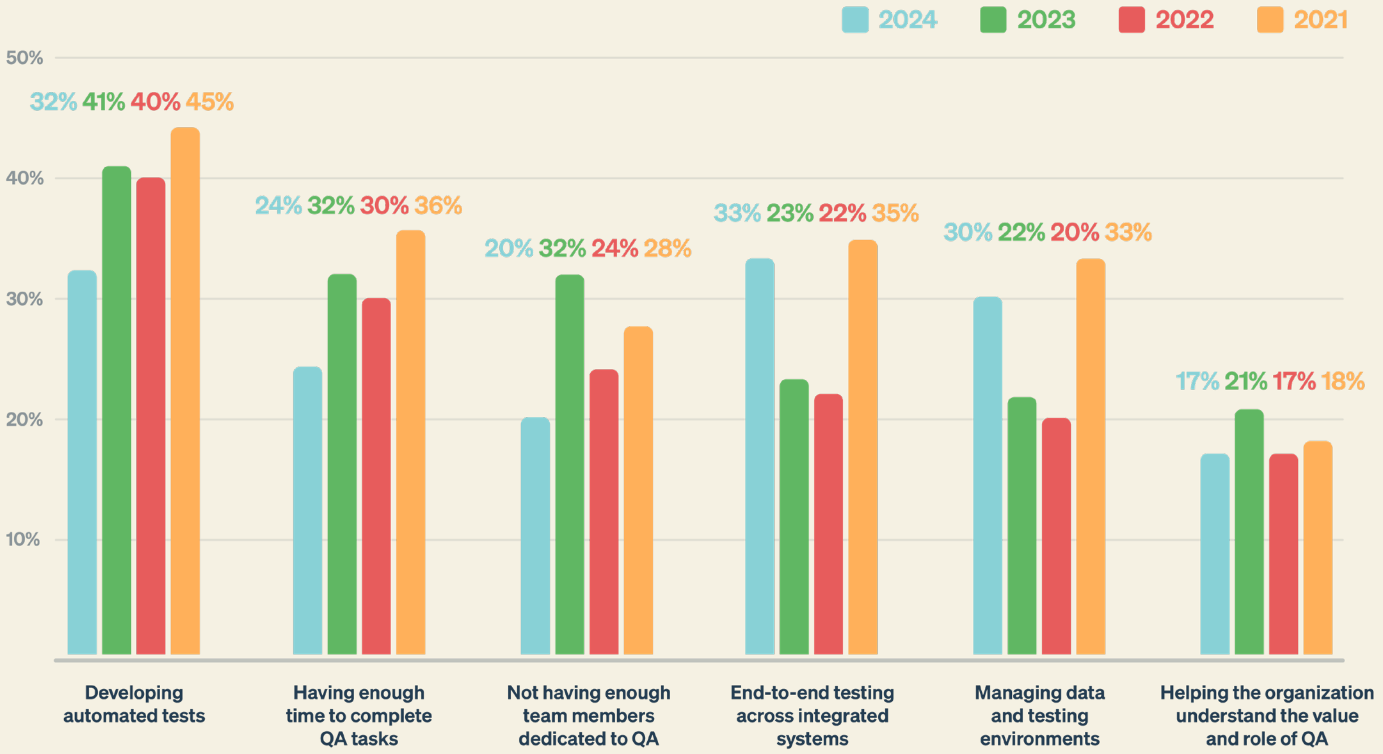
2. Making sure the correct tests are being run (Average Score: 3.97)

Ensuring proper test coverage remains a core concern, especially as teams balance manual and automated testing. This directly supports the top-ranking objective of increasing test coverage reported elsewhere in the survey.

While test automation is still considered important, respondents placed more emphasis on executing the right tests than simply creating more of them. Similarly, while Agile and DevOps adoption is a key part of many QA strategies, it's not rated as a top priority, likely because many teams already operate within those models.

Lower-ranked priorities include outsourcing QA tasks and involving external stakeholders in testing. This suggests that most teams prefer to maintain ownership and direct control over testing responsibilities, rather than distributing them across external teams.

What are your team's top 3–5 biggest challenges around testing and QA right now?



QA teams are still facing familiar hurdles, but this year's data reveals a clear shift in the nature of those challenges. Compared to 2023, when teams struggled most with automation and basic resourcing, 2024 has brought deeper and more complex issues to the forefront.

The most commonly reported challenges now center around testing integrated systems, managing complex environments, and being brought into development too late. Budget limitations are still a concern, but efficiency and process complexity have become more pressing.

Here are the top three challenges reported in 2024:

1. End-to-end testing across integrated systems (33% in 2024, up from 23% in 2023)

This is now the most frequently cited challenge and has grown significantly year over year. As teams mature in their practices, they are encountering the intricacies of testing distributed systems, APIs, and third-party integrations. Maintaining stable test environments and ensuring coverage across interconnected components is becoming increasingly difficult.

2. Developing automated tests (32% in 2024, down from 41% in 2023)

While automation remains a top priority, this challenge has become slightly less prominent, suggesting that some teams have made progress. However, many still cite tooling limitations, fragile test suites, and skill gaps as key obstacles. For most, automation is both a goal and a challenge.

3. Earlier involvement in development (32% in 2024, up from 29% in 2023)

QA teams continue to advocate for being brought into the development process earlier. This steady increase reflects a growing push toward shift-left practices that allow teams to catch issues sooner, influence requirements, and ensure testability from the start.

The most significant trend in this year's data is the pivot from resource shortages to systems complexity. As QA teams grow in maturity, they are no longer just asking for time and tools. Instead, they are navigating the challenges of scaling quality practices across modern, integrated environments.

Correlation Analysis: QA Challenges vs. Testing Objectives

End-to-end testing complexity	Increase test coverage
Test automation difficulties	Automate more tests
Early involvement in development	Shift-left testing
Managing test data and environments	Make testing efficient

The correlation heatmap from this year's survey reveals strong links between the challenges QA teams face and the objectives they prioritize. In many cases, the pain points are directly driving the goals teams are working toward.

1. End-to-end testing complexity ↔ Increase test coverage (Correlation: 0.91)

Teams struggling with end-to-end testing are also the ones most focused on expanding test coverage. This suggests that complex systems, integrations, and third-party dependencies are pushing teams to widen their testing efforts to ensure nothing gets missed.

2. Test automation difficulties ↔ Automate more tests (Correlation: 0.89)

As expected, teams that struggle with automation are also prioritizing it. The challenge isn't a lack of interest—it's execution. This reinforces that while automation is a shared goal, tooling, skill gaps, and scalability remain common roadblocks.

3. Early involvement in development ↔ Shift-left testing (Correlation: 0.85)

QA teams that feel excluded from early development stages are the same ones pushing for shift-left practices. This correlation highlights a growing desire to move quality upstream and become part of the planning and design process.

4. Managing test data and environments ↔ Make testing efficient (Correlation: 0.83)

Environment setup and test data challenges are tightly linked to efficiency goals. Teams recognize that without stable, well-managed environments, it's nearly impossible to streamline testing processes.

What key initiatives or challenges does your team plan to focus on over the next 1 to 3 years?

QA professionals shared a clear picture of where their teams are headed: toward faster, smarter testing that's more closely integrated with development. While the challenges are real, the ambition is just as strong.

Here are the top initiatives respondents plan to focus on:

Expanding test automation

As the most commonly mentioned initiative, teams are looking to boost automation coverage, reduce manual effort, and integrate automation more tightly into CI/CD workflows. Many are also starting to explore AI-powered testing tools.

Shifting left

There's growing momentum around involving QA earlier in the development process. By joining during the planning and design phases, teams hope to prevent issues before they start and improve collaboration.

Improving test coverage and quality metrics

Teams are working to reduce bugs in production and build more comprehensive, traceable test suites that improve release confidence.

Accelerating testing cycles

As Agile and DevOps practices become more common, QA teams are under pressure to move faster. Streamlining testing without compromising quality is a key goal.

Strengthening cross-functional collaboration

Teams are breaking down silos between QA, developers, and product stakeholders to create a shared sense of responsibility for quality.

Improving test data and environment management

As applications grow in complexity, many teams are investing in more stable environments and reusable, realistic test data.

Investing in non-functional testing

Performance, security, and compliance testing are gaining traction. These areas are becoming core priorities rather than nice-to-haves.

And here are the biggest challenges respondents are grappling with:

Balancing speed and quality

Faster release cycles often come at the cost of thorough testing. Teams are struggling to keep quality high while delivering quickly.

Keeping up with change

The testing landscape is evolving rapidly. New tools, frameworks, and AI-driven capabilities are hard to keep up with, especially for smaller or under-resourced teams.

Lack of skilled resources

Hiring and upskilling remain major pain points. Many teams report difficulty finding QA professionals with automation experience and technical know-how.

Scaling test automation

Automation is a goal for most teams, but scaling it is tough. Flaky tests, high maintenance costs, and fast-moving codebases create real challenges.

Test data and environment bottlenecks

Poor environment stability and limited access to usable test data are common blockers. These issues slow down testing and reduce confidence in results.

Taken together, these responses show a QA landscape that's growing more sophisticated and ambitious. Teams are preparing for a future that's faster, more collaborative, and more automated—but they need the right support, skills, and infrastructure to get there.

Tool assessment and decision fatigue is a real thing. Lean on independent reports from credible sources like TestRail's TEI by Forrester.

Read Report

EXECUTIVE SUMMARY

Return on investment (ROI) 180%

Benefits PV $4.30M

Benefits (Three-Year)

Test administration productivity gains

Software developer productivity gains

Reduced cost of production bugs

Value of faster time to market for new and improved offerings

FORRESTER

The Total Economic Impact™ Of TestRail

Cost Savings And Business Benefits Enabled By TestRail

A Forrester Total Economic Impact™ Study
Commissioned By TestRail, November 2024

Due Diligence
Interviewed TestRail stakeholders and Forrester analysts to gather data relative to TestRail.

Interviews
Interviewed four representatives at organizations using TestRail to obtain data about costs, benefits, and risks.

Composite Organization
Designed a composite organization based on characteristics of the interviewees' organizations.

Financial Model Framework
Constructed a financial model representative of the interviews using the TEI methodology and risk-adjusted the financial model based on issues and concerns of the interviewees.

Case Study
Employed four fundamental elements of TEI in modeling the investment impact: benefits, costs, flexibility, and risks. Given the increasing sophistication of ROI analyses related to IT investments, Forrester's TEI methodology provides a complete picture of the total economic

Section Summary

QA teams are navigating a dynamic landscape, balancing the need for speed with a growing emphasis on **quality, automation, and early involvement.** This year's data shows a shift from basic resourcing struggles to more sophisticated challenges like end-to-end testing, managing test environments, and integrating QA earlier in the development lifecycle.

The top priorities for teams remain consistent: **increasing test coverage, reducing bugs in production, and automating more tests.** But many also expressed a desire to track deeper, more actionable metrics that go beyond pass/fail rates like defect root causes, test cycle times, and traceability.

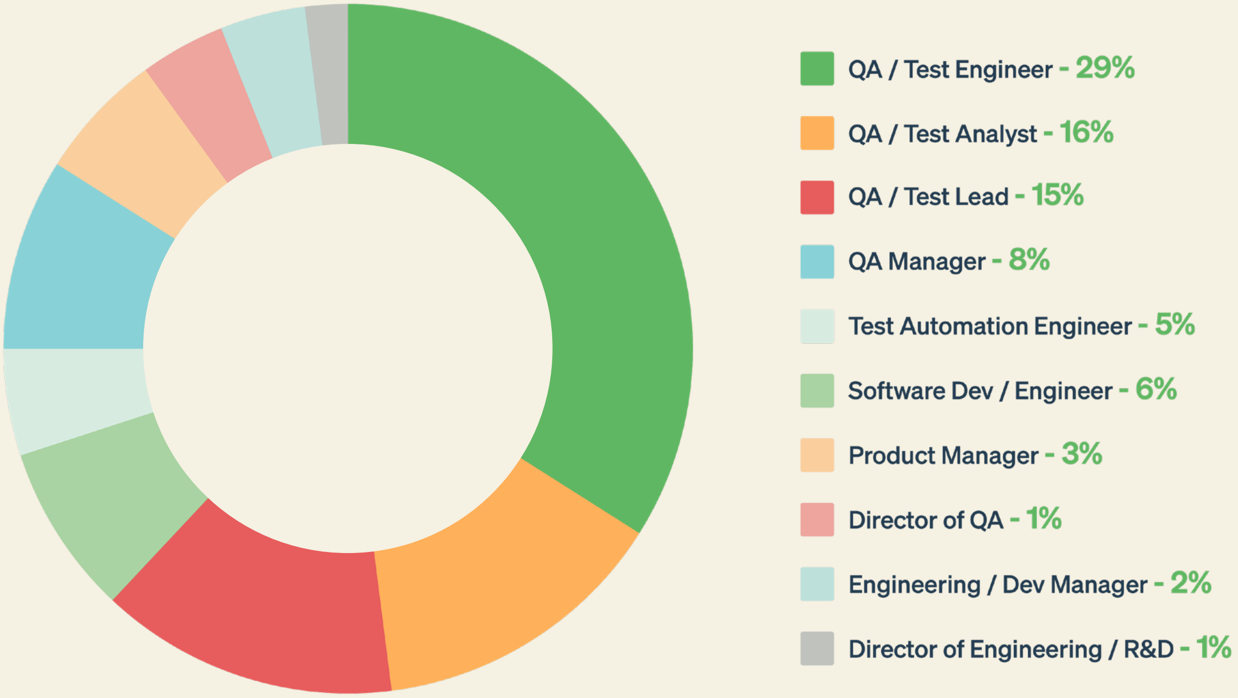
Looking ahead, QA teams are setting ambitious goals: **scaling automation, embedding QA earlier in planning, improving collaboration**, and **investing more in non-functional testing like performance and security.** But they're also realistic about the roadblocks, such as keeping up with tool evolution, finding skilled resources, and maintaining stable environments.

Ultimately, QA teams are aiming to work smarter, not just harder, and build more connected, efficient, and resilient testing processes that can keep pace with modern development.

SECTION 06

Survey Details

Which of the following best describes your job responsibility and title?

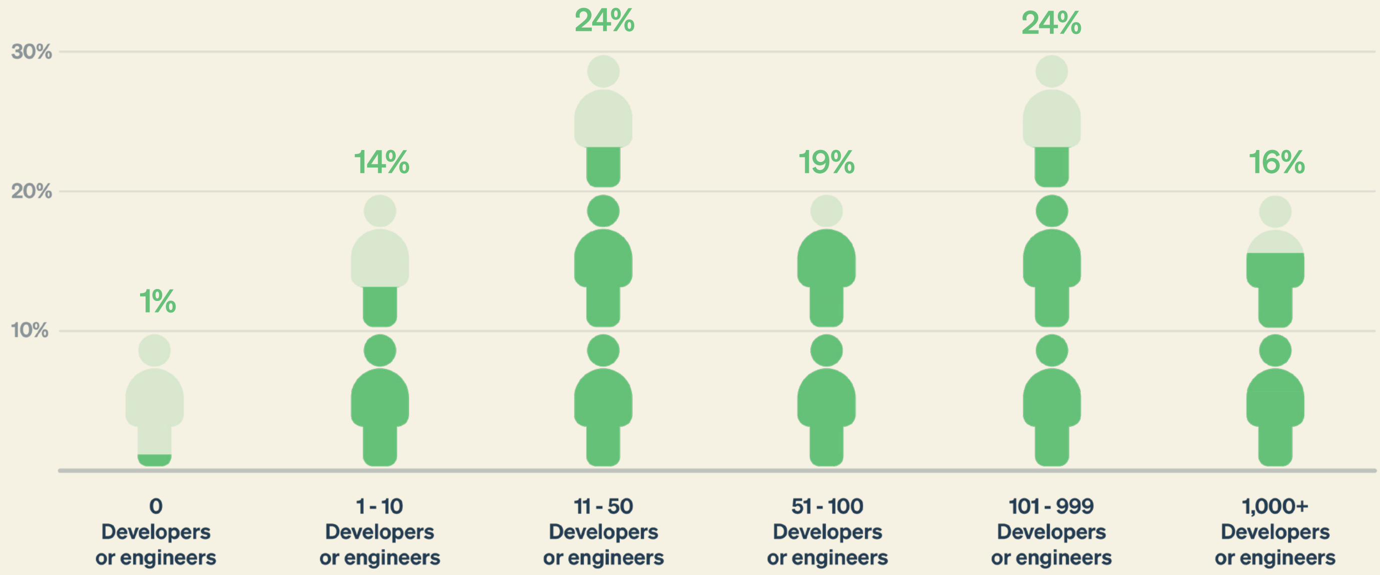


Each year, we extend an invitation to every TestRail user to complete our annual survey. Starting with this year's edition, we also opened up the Software Testing and Quality survey to any interested quality professional, regardless of whether they use TestRail. The most recent survey received a total of **2,751 responses.**

This year's survey responses indicate a strong representation of professionals directly involved in software testing and quality assurance. The majority hold specialized roles in QA analysis, engineering, and leadership.

The largest group of respondents identify as **QA/Test Engineers** (29%), followed by **QA/Test Analysts** (16%) and **QA/Test Leads** (15%). These findings suggest that our feedback primarily comes from actual testers—those creating, executing, and analyzing tests on a daily basis.

How large is your organization's software development and engineering team?

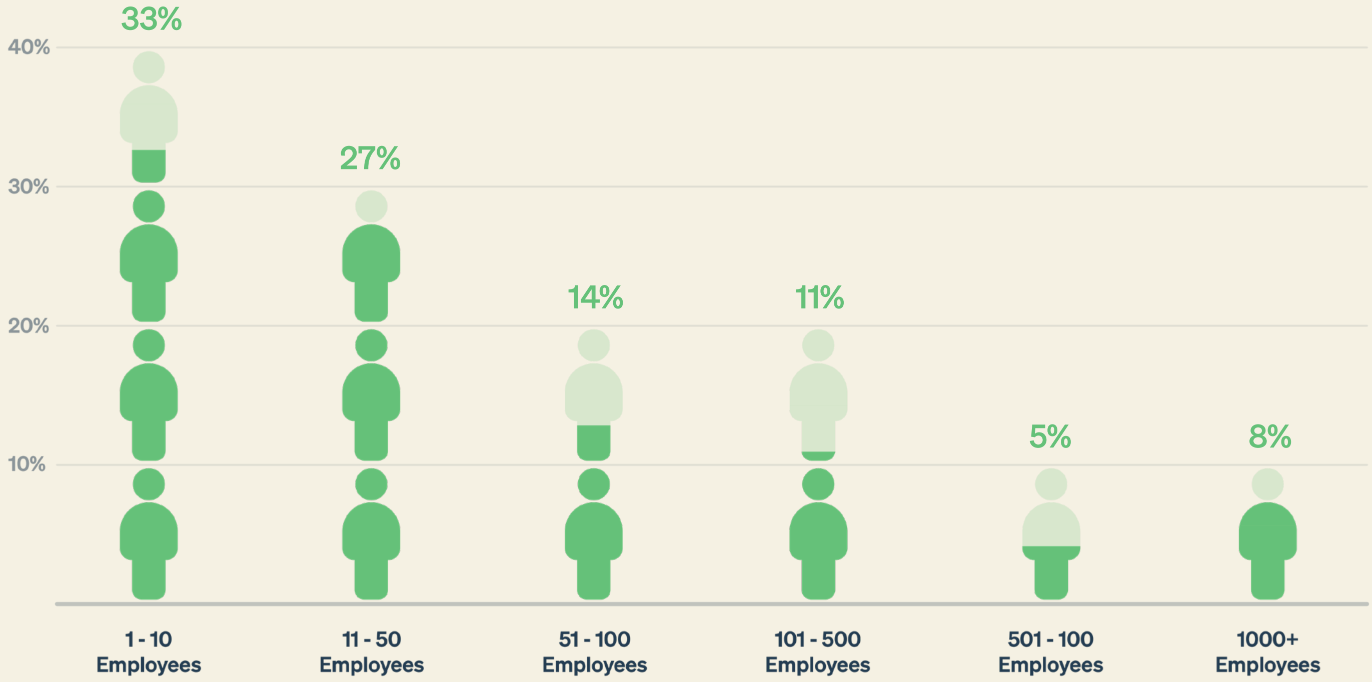


The majority of survey respondents **(66%)** reported operating in mid-to-large-sized engineering environments, defined as more than 11 but less than 1,000 developers. Structured QA processes are more likely to be in place at organizations with mid-to-large-sized development and engineering teams.

This year also saw a growth in respondents from enterprise-level organizations (more than 1,000 developers)—**up nearly 6%** from our last survey, highlighting the presence of QA professionals in large-scale enterprises with extensive development operations.

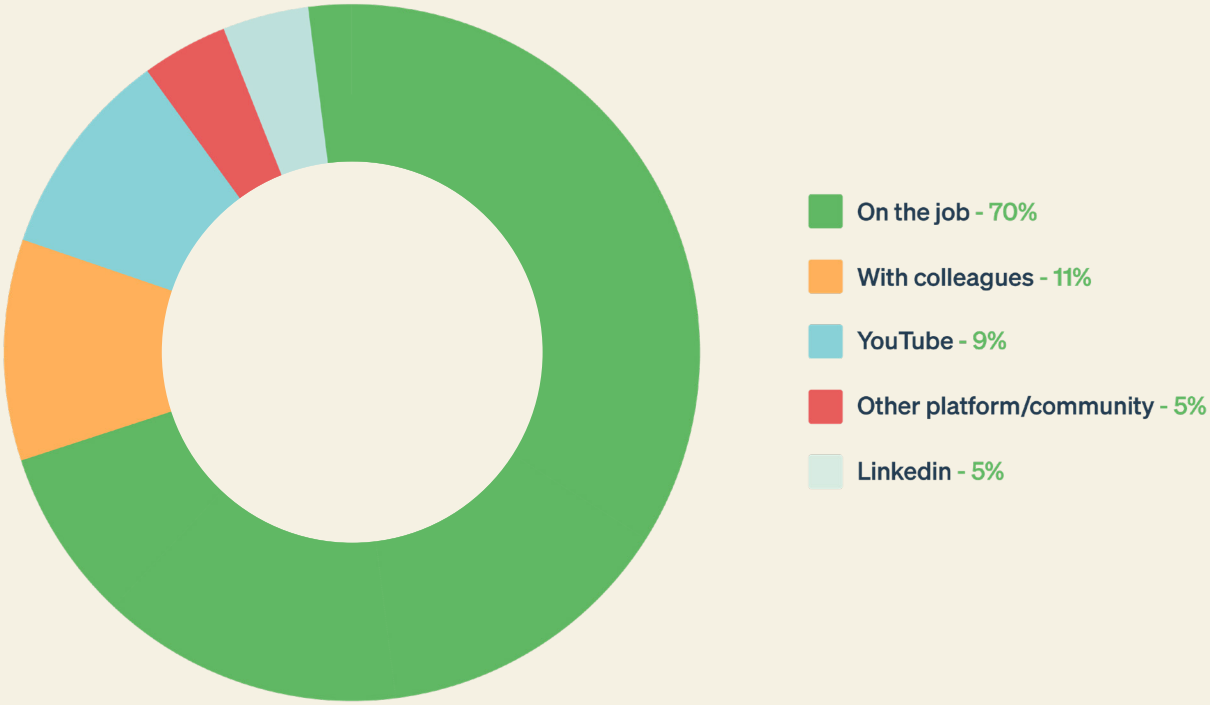
Survey Details

Approximately how many individuals are dedicated to software testing and QA in your organization?



Smaller QA teams are still the most common, with **33%** of respondents working in organizations with 1-10 QA professionals. However, the QA team size represented in our annual survey has been steadily increasing year-over-year, and this year was no different. Our survey results saw a roughly **2% jump** in the 51-100 employee, 101-999 employee, and 1,000+ employee categories, indicating steady growth and investment in QA as companies also grow.

Where do you learn the most about QA-related topics?



The majority of QA professionals learn the most through hands-on experience **(70%)** and discussions with colleagues **(11%)**, making on-the-job learning the primary source of continuous learning. **Online platforms** serve as supplementary resources, with **9%** using YouTube, and **5%** turning to LinkedIn for QA-related topics.

SECTION 07

Conclusion

Conclusion

While year-over-year evolution has been slow and steady, and nothing shook the core of QA in 2024, it still marked a “level up” year for many of us in software testing and quality. Teams are adding more responsibilities and skills to their arsenal to accomplish their tests with greater efficiency, accuracy, and overall confidence as technologies mature and more complex challenges arise.

While AI hasn’t revolutionized everything we know about QA quite yet, it is already proving to be a great driver of efficiency, enabling teams to save precious time on repetitive tasks throughout the testing process. However, the time saved is not “free time”—it is instead valuable hours that are able to be reinvested into exploratory testing, automating more tests, and other tasks that need a human touch. Despite fears that AI may one day replace human testers, it so far appears to be most promising in freeing up time for human testers to do what they do best.

But efficiency gains and time savings through processes like AI and automation mean less and less every day. One trend we can count on holding steady each year is the demand for QA teams to release higher quality software, faster, and release times continue to shorten rapidly, with daily releases increasing. This means that QA teams will have to adapt even faster to keep up in the face of such organizational and customer expectations regarding the quality and speed of releases.

Looking ahead, QA teams are increasingly focused on scaling automation, shifting quality further left, and working smarter under the pressure of often constrained resources and more complex testing. Teams are also largely optimistic about the coming years for AI in QA, hoping that advances will bring major efficiencies to test automation, test case and script generation, and defect analysis.

We hope that this snapshot into the last year of life in QA has been insightful and inspiring, and we want to extend our thanks to every survey respondent who shared their time and insights with us. Who knows what the coming year will bring, but we know one thing for sure—QA teams are resilient, adaptable, creative, and ready for anything 2025 has in store.

TestRail

About TestRail

Gurock Software was founded in 2004 and now has offices in Frankfurt, Dublin, Austin, and Houston. Our flagship test case management solution, TestRail, is used by more than 100,000 members of development and QA teams to build rock-solid software—including companies like Amazon, NASA, Adobe, Sony, PayPal, and Siemens.

TestRail is the only platform that empowers QA teams to build, connect, and optimize all their testing processes. TestRail's Quality OS centralizes manual and automated test management and gives you visibility into your entire quality operation so you can manage your team more flexibly and build repeatable, scalable workflows. And, with a unified platform that integrates with your DevOps pipelines, you can share testing timelines, data, and insights across your whole organization.

TestRail is a leader in the G2 Grid for Test Management and Software Testing, with top ratings year-over-year for best results, most implementable, and overall enterprise leader. For more independently verified research and reviews, visit the TestRail page at G2 or Capterra.

Let TestRail lift your team out of chaos and toward faster, frictionless releases. Experience the TestRail difference with a free trial today—no credit card required. Get started today and start releasing higher-quality software, faster!

Try TestRail

Get a Demo