

The Quality Engineer's Guide to Integrating Accessibility Testing into DevOps

How Customer-Centric QA Teams Can Deliver
Inclusive Digital Experiences with Test Automation

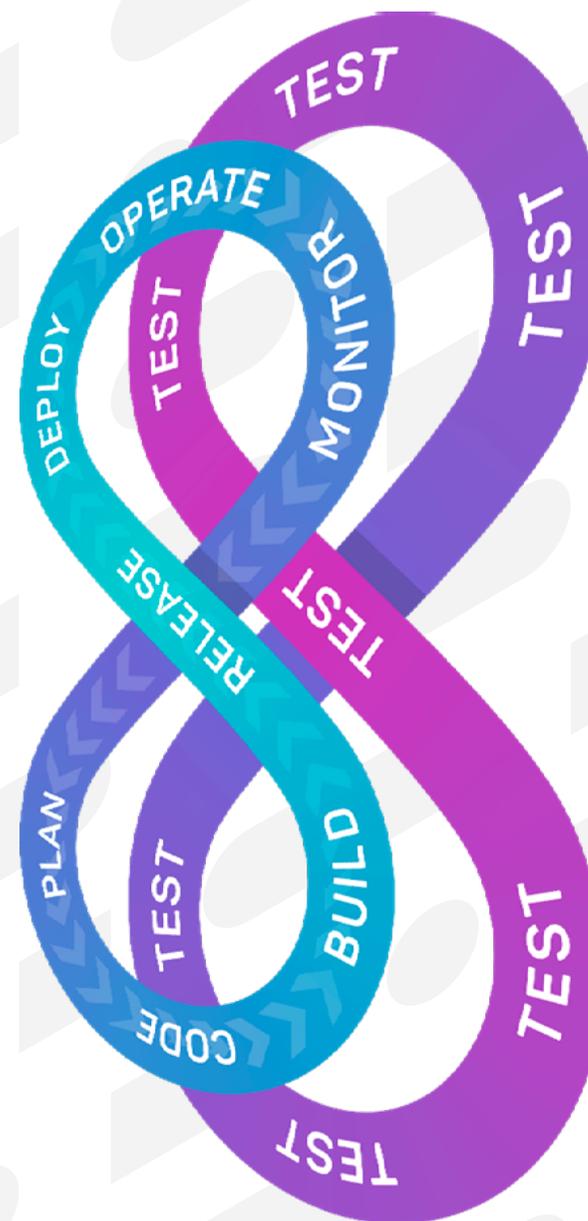


Quality Engineering: Creating Quality Digital Experiences for Everyone

Compounding the issue is that no one owns accessibility in the software development lifecycle. Those most concerned with ADA compliance - legal teams - are often located outside the product and engineering organization entirely. Embedding their expertise into development pipelines without slowing production schedules is nearly impossible.

Enter quality engineering, the practice of embedding testing throughout the development lifecycle to deliver positive user experiences that build customer loyalty and improve customer acquisition. QE helps software testing and QA teams expand the definition of quality to match the full user experience through high-impact, scalable test automation and targeted manual testing. With the right solutions, QA teams can harness their product expertise and embrace their role as customer advocates to integrate automated accessibility checks into their organization's DevOps practice without slowing down release schedules.

This guide will help quality teams understand how automated accessibility checks can be embedded into development pipelines, how to build collaborative accessibility workflows, discover how a real-world quality engineering team made their platform more accessible, and come away with actionable tips for making their own apps and websites better for everyone.



01	Who Owns Accessibility in DevOps?	03
02	Elements of a WCAG-Compliant UI	07
03	Data-Driven Empathy with Accessibility Testing	11
04	Designing Processes and Partnerships for Accessibility Testing	16
05	How SmugMug Embedded Accessibility into Their End-to-End Tests	19



01

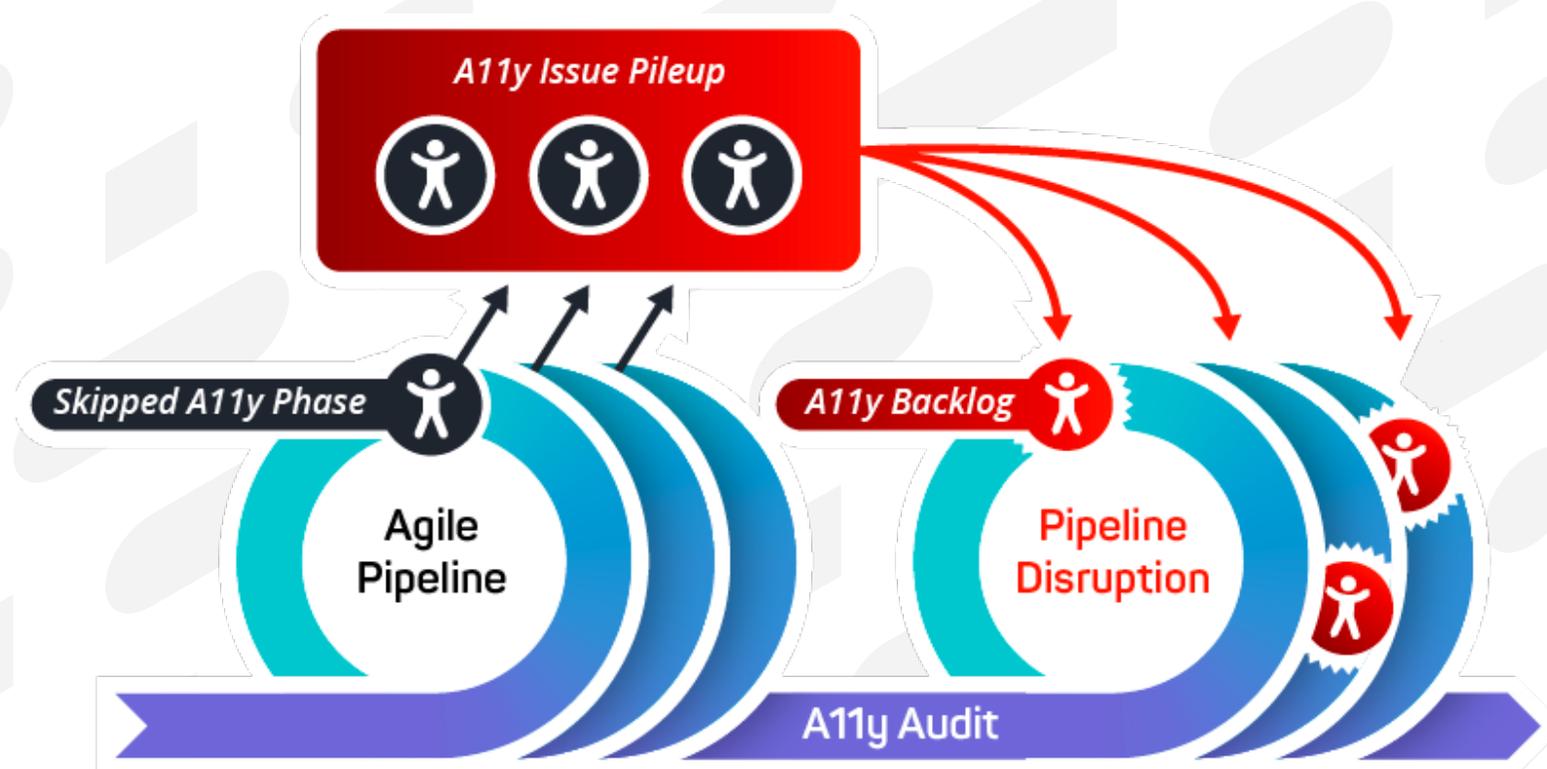
Who Owns Accessibility in DevOps?

Accessibility is one of the most pressing concerns facing companies today, yet it's often unclear who owns or manages accessibility in an organization. Legal teams are concerned about maintaining compliance with regulations like the [Americans with Disabilities Act](#), but they often lack day-to-day contact with software development. On the other hand, development teams are struggling to manage accelerating development cycles and a talent shortage, making it difficult to add more steps to development pipelines. DevOps, with its focus on automating and accelerating code releases, only adds to the confusion by limiting the amount of time available to bring new stakeholders into the development process.

Quality Engineering Allows QA to Expand Testing to Include Accessibility Checks

Accessibility testing is a natural - and ethically imperative - extension of quality engineering. As the glue between development and users, quality engineers are well positioned to take a leadership role in managing accessibility...if they have the right tools and processes in place.

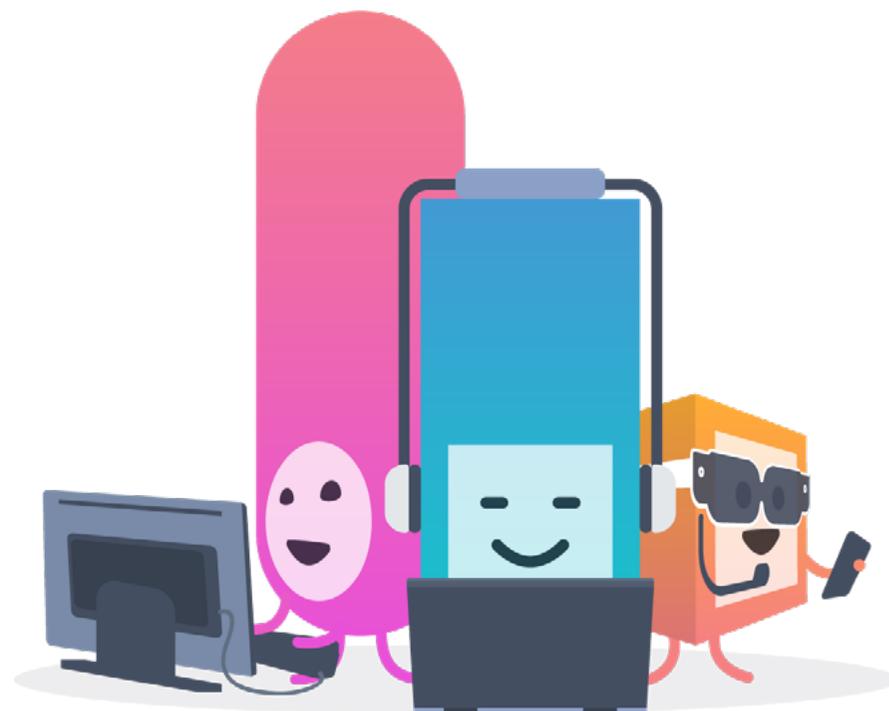
Once adopted, quality engineering ultimately enables a [culture of quality](#), where everyone shares responsibility for software testing and product quality. When testing is a shared responsibility, QA teams have more time to expand their software testing strategy without slowing down development.



Using a unified test automation platform like mabl, software testers can embed automated accessibility checks into development pipelines, typically as part of deployment stage end-to-end tests. This means every release can be checked against the Web Content Accessibility Guidelines (WCAG) - versions 2.0 and 2.1 for accessibility management practices that match the pace of DevOps.

The resulting data from these frequent accessibility checks make it easier for quality engineering teams to collaborate with their organization's compliance departments for cohesive accessibility management. Rather than wait for a quarterly or annual accessibility audit, the entire company can use accessibility testing data to proactively make their application or website more accessible, even as it evolves with daily code releases.

When quality engineers are able to automate unit, regression, and end-to-end tests, they're able to devote more time to expanding their testing strategy in ways that better reflect the customer experience, including users with different access needs. Automated accessibility checks empower software testing teams to embed inclusive quality practices into their development pipelines, ensuring that everyone has a positive experience with their product.





02

Elements of a WCAG- Compliant UI

Once you're ready to start integrating accessibility testing into your development pipeline, it's important to understand the most widely accepted accessibility standards and common issues that hinder compliance. Accessibility requirements and guidelines vary widely around the world, but the [Web Content Accessibility Guidelines \(WCAG\) - versions 2.0 and 2.1](#) are an international standard that meets or exceeds requirements in a [large number of countries](#).

The standards have 12-13 guidelines, each with three levels of success criteria: A, AA, and AAA. These criteria [determine](#) how well a website or application conforms to the **WCAG guidelines**:

- **Level A**, the minimum level of conformance, means that an app or website meets all A success criteria.
- **Level AA** indicates that every AA and A requirement is met.
- **Level AAA**, the highest level of adherence, demonstrates that all success criteria (A, AA, and AAA) are met.

It's worth noting that AAA conformance is not recommended as a general policy since it's not possible to meet it for some web content. Many organizations may find themselves between levels, especially as they start their accessibility improvements. To help quality engineering teams get started, let's look at the most common issues uncovered by automated accessibility tests and what can be done to resolve them quickly in a DevOps pipeline. Note that while these issues don't represent the full range of accessibility improvements that may be needed, they're among the most common obstacles to WCAG compliance.

Labeling Form Elements

One of the most common - and most serious - web accessibility issues is the lack of labels on form elements, particularly those that are necessary to interact with the web page, such as buttons. Clear labels that can be easily read by screen readers are essential for those living with vision loss, a population estimated to be [2.2 billion worldwide](#). Considering that the majority of people with vision impairment and blindness are over the age of 50, that already massive number will continue to rise as the global population ages. If a website fails to correctly label buttons necessary to complete a transaction, that translates to a growing number of possible missed customers.

When accessibility checks are added to automated end-to-end tests in mabl, quality engineering teams will find labeling issues grouped together in their [accessibility dashboard](#). As more tests are run over time, trends in labeling issues will help quality, development, and compliance teams understand how the user experience is improved for those using screen readers.

LABEL YOUR FORM FIELDS

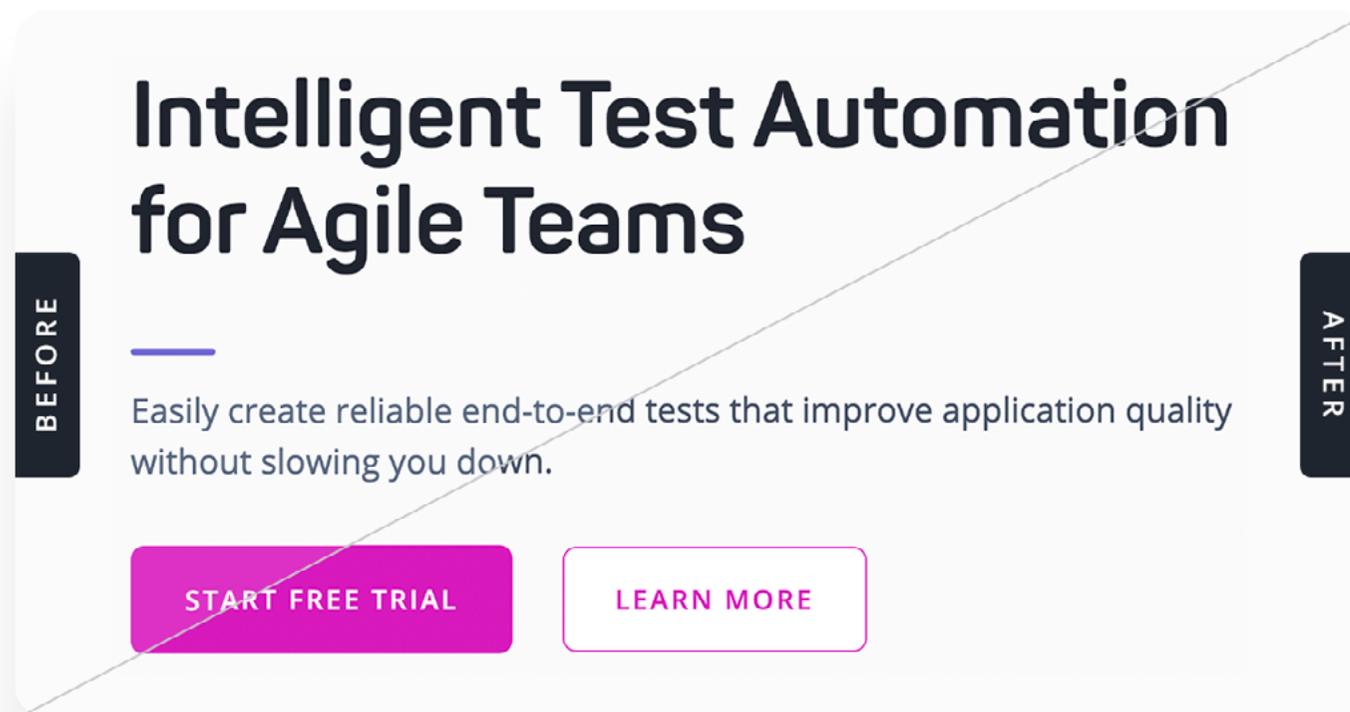
First name *	Last name *
<input type="text"/>	<input type="text"/>
Business Email *	Company *
<input type="text"/>	<input type="text"/>
Country/Region *	
<input type="text" value="Please Select"/>	
<input type="submit" value="Submit"/>	

Improved Color Contrast and Starker Text

For the millions of people with vision issues that don't require a screen reader, improving the contrast of text, buttons, and other visual elements can make their experience with an app or website significantly easier. Considered an AA level criteria by the WCAG, adopting crisp, high-contrast color schemes is an excellent example of how accessibility guidelines help improve the online world for everyone, regardless of access needs.

As mabl considered how to make our website more accessible, we made subtle adjustments to our own website color palette to make text easier to read and images easier to understand.

Looking at the top half of the above image, the text is a lighter shade of gray and the "Start Free Trial" button is a slightly lighter magenta. Though the changes are almost indiscernible, they make the entire mabl site a more delightful experience for all users, especially those with vision impairment.



Intelligent Test Automation for Agile Teams

BEFORE

Easily create reliable end-to-end tests that improve application quality without slowing you down.

AFTER

START FREE TRIAL

LEARN MORE



03

Data-Driven Empathy with Accessibility Testing

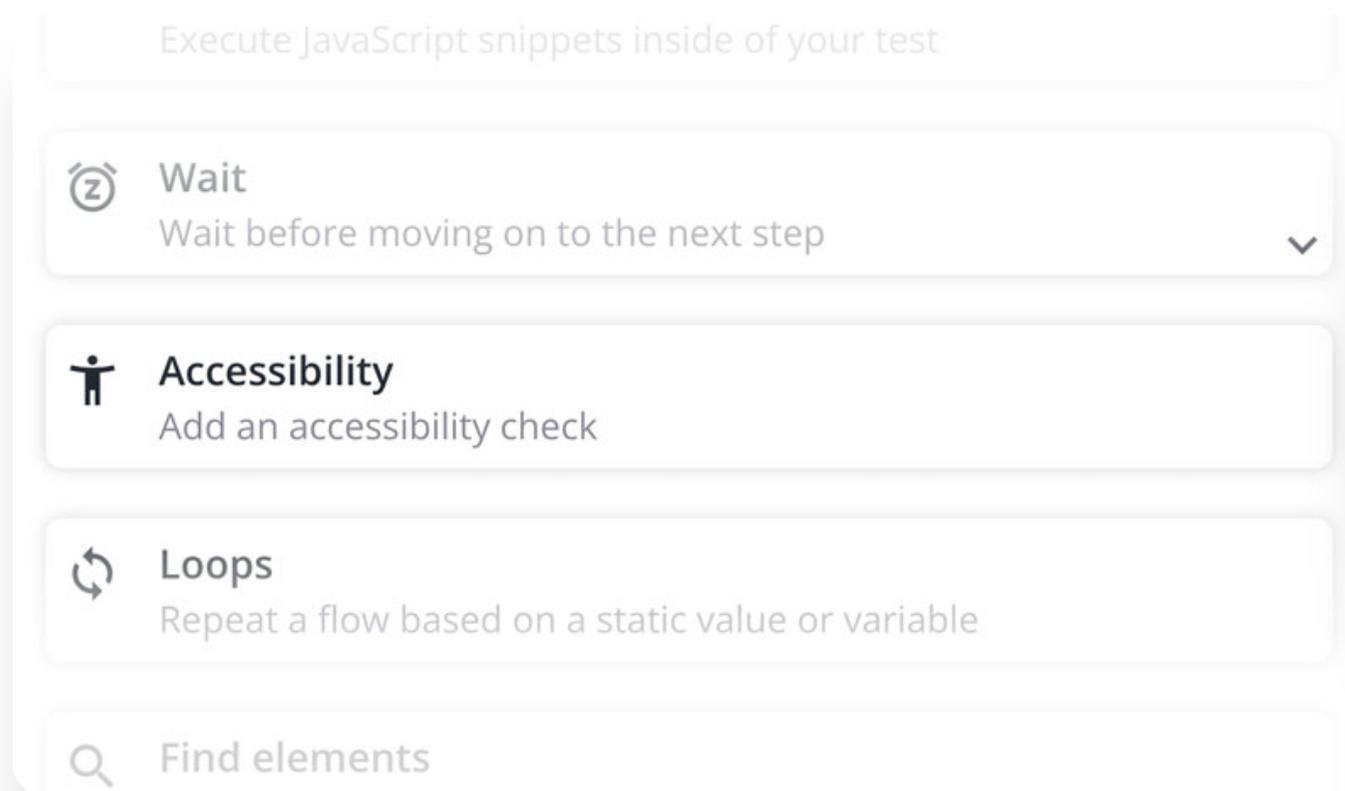
Quality engineering is best positioned to lead the transformation to a more inclusive digital world through expanded software testing and data-driven goals. As useful as accessibility checks are for ensuring that each code release is compliant with the Web Content Accessibility Guidelines (WCAG) - versions 2.0 and 2.1, the foundation for a sustainable accessibility testing strategy lies with the resulting data.

Building Accessibility into QE Practices

Low-code accessibility testing makes accessibility checks as simple as [adding a step](#) to existing end-to-end tests. Simply incorporating these steps into routine testing is a solid start to fully integrating accessibility into quality engineering, but to fully advocate for all users and amplify the impact of accessibility checks across their company, quality engineers need to build new practices and partnerships.

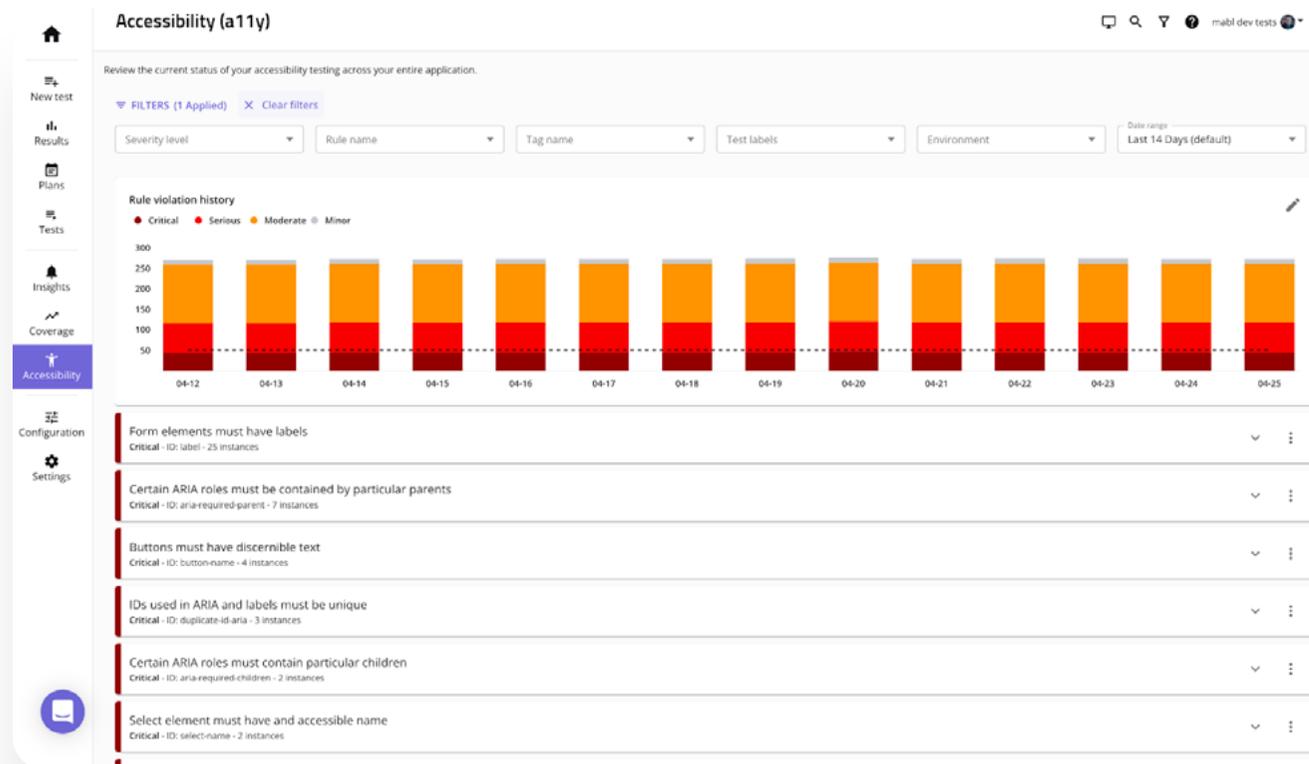
Turn Empathy into Action

One of the [core skills](#) of the quality engineer is empathy. When QE is actively working to understand the perspective of the end user, they're better able to champion the customer perspective in development. Accessibility checks and reporting are critical for turning that empathy into action so all customers have a positive experience with an organization's application or website.



Running automated accessibility checks with mabl results in a dashboard with accessibility issues ranked from critical to minor, making it easy for QA teams with limited familiarity with [accessibility standards](#) to understand where their product can be improved.

Mabl's accessibility reports automatically group similar issues that may be detected across multiple test runs, so quality engineering teams can spend less time chasing down false positives and more time focusing on the issues that matter to customers. This unified view across all accessibility issues drastically simplifies the time and effort needed to identify and resolve accessibility issues, helping quality engineering teams improve the user experience level for all customers without slowing down DevOps pipelines.



Tips for Tracking Accessibility of Your App

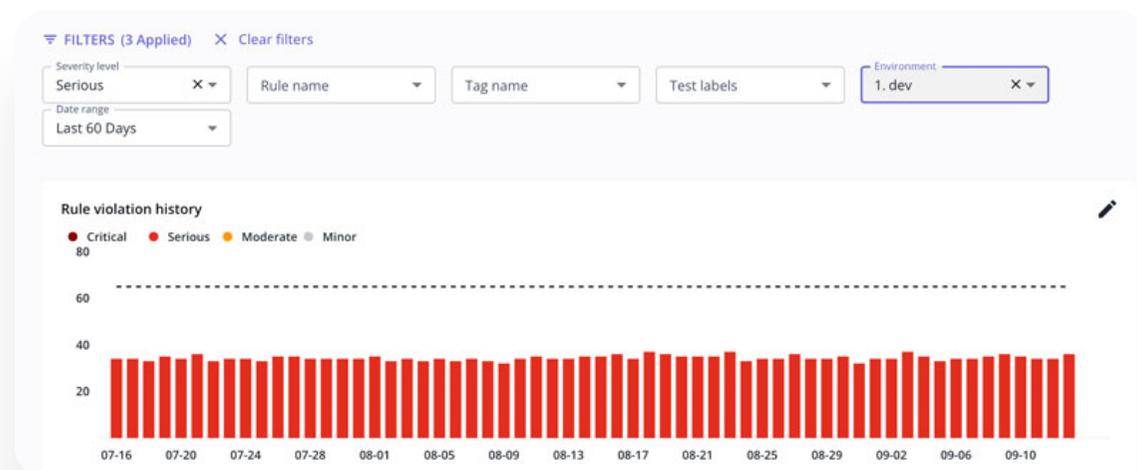
- Agree on which level(s) of WCAG compliance make sense for your app
- Decide which rules won't be addressed in the short term
- Prioritize fixing issues with the highest severity first

Expand Exploratory Testing and Collaborate with Accessibility Audit Teams

A thorough accessibility management strategy uses a combination of embedded accessibility checks, targeted exploratory testing, and regular audits to ensure all customers have a positive experience with an application. Mabl's accessibility reporting features aggregate test data into trend reports, so quality engineers can easily collaborate with legal and compliance teams that manage any existing accessibility audits.

Using accessibility trend data, quality engineers can expand manual exploratory testing to target frequently flagged accessibility issues. If form elements often lack accurate labeling, for example, common tools like screen readers allow software testers to emulate the experience of users with vision impairment. This technique taps into the natural empathy of QA professionals for a better quality strategy.

Conversely, data from automated testing can help inform testers of issues that might otherwise slip by standard exploratory testing, and give quality engineers a common source of knowledge to collaborate with compliance teams.



Manual and automated accessibility checks unlock a complementary testing strategy that covers more of the user experience without slowing down development.

Advocate for All Users and Improve Customer Relationships

Billions of people worldwide live with some form of different access need. Though building inclusive experiences for them is morally necessary and good business sense, many quality engineering teams struggle with creating an accessibility testing strategy with real impact. Unified accessibility issue and trend reporting make it possible to fix issues faster and collaborate with compliance experts for a holistic accessibility solution.



04

Designing Processes and Partnerships for Accessibility Testing

The most [commonly cited](#) DevOps adoption challenges are the lack of good processes and organizational reluctance to change. It's easy to see why process is a major inhibitor to DevOps: in order to release code on a daily basis, each part of the development process must be easily executed every single day. Perhaps most importantly, the handoff between each team - development, QA, and product - needs to be as seamless as possible to ensure code moves easily across each stage of the [DevOps pipeline](#).

Best Practices for Good Processes and Partnerships

Though every software development organization is different, a few best practices are applicable for teams of all sizes and all stages of DevOps maturity:

Use data to create goals and awareness: one of the benefits of running accessibility checks within a DevOps pipeline is the ability to [identify trends and issues](#) based on test results. In unified test automation solutions, this data creates valuable insights that can be shared across the quality engineering, development, and compliance teams for more accurate goal setting. Results that can be shared across the enterprise are essential for getting organizational buy-in to the new accessibility testing process.

Harness automation to support repetition: no one wants more work added to their plate, even when that work is as important as improving digital accessibility. This applies doubly to overworked development organizations navigating talent shortages and shortening development cycles. To ensure that accessibility checks are easy to execute (and iterate), quality engineering teams should embrace automated accessibility checks that can be [embedded in DevOps pipelines](#) without generating too much additional work or slowing development.

Create opportunities for shared learning: for most companies, experts on accessibility and compliance issues are on the legal team, not the software development organization. To harness that resource and ensure automated accessibility checks are meeting the necessary regulatory requirements, quality engineering professionals should build processes that support collaboration across departments. Trend reports that aggregate test data into clear, concise reports help break down knowledge gaps between teams, helping everyone set shared goals and ultimately, create better user experiences for everyone.

*To make a meaningful change,
accessibility checks and improvements
need to be grounded in processes that
are repeatable and collaborative.*

Repeatable, Scalable Processes Enable Positive Partnerships

DevOps and quality engineering are collaborative practices. In order to rapidly release new code into production without accessibility issues or other defects, software companies need good processes in place that encourage everyone to contribute their expertise. As the glue between code and customer, quality engineers are well-positioned to lead the development of accessibility testing processes that make inclusive digital experiences possible in a DevOps world.



06

How SmugMug Embedded Accessibility into Their End-to-End Tests

As the pandemic continues, digital platforms have become the de facto choice for many businesses and consumers, pushing a new wave of accessibility requirements from around the world. At the same time, the number of lawsuits concerning accessibility has risen significantly. This increasing legal and financial pressure only underscores the moral necessity of ensuring applications and websites are accessible by default, not when development teams have time to make reactive updates to customer-facing products.

Accessible Experiences are Beautiful Experiences at SmugMug

[SmugMug](#) is the safe, beautiful home for photos, providing passionate photographers at any level of experience a place to store, share, and sell high-quality photos. As the leading platform for photographers worldwide, SmugMug prides itself on building beautiful experiences for all users, regardless of access needs.

For the SmugMug QA team, adopting accessibility checks with mabl aligned with the overall company mission and their own goal of empowering the development team to make bigger and bolder changes without sacrificing speed or confidence. When software testing is comprehensive and efficient, the entire software development organization can innovate faster. Accessibility checks help ensure that each new feature or product benefits the entire SmugMug user community.



Accessibility Testing in mabl Makes Accessibility Efficient, Collaborative

As software becomes more complex and DevOps practices increase release cycles, testing is struggling to balance the need for more advanced quality strategies that cover all functional and non-functional testing, and enable rapid, iterative sprints. On one hand, there's a wide range of automated testing tools that can support this growing range of testing needs. But a complex testing tech stack often results in a disparate testing strategy that allows more bugs to slip into production. A unified approach is key to enabling quality teams to establish and sustain a thorough quality engineering practice.

For Dan Laabs, Senior Test Engineer at SmugMug, accessibility testing needed to be integrated into their larger [software testing strategy](#). The team is small and has limited bandwidth to implement new tools, especially new testing tools that would require recreating existing end-to-end tests.

Organizations like SmugMug are proof that accessibility testing can be adopted as a critical part of a quality engineering strategy that enables bolder, more confident development teams. Their platform is designed to help photographers of all levels celebrate their work: when people with different access needs are excluded, they have less support to pursue their passion. Thanks to accessibility testing with mabl, SmugMug is helping everyone live a creative life.



Rather than creating a new accessibility testing process from scratch, Dan was able to see the value of accessibility testing with mabl in just 10 minutes by reusing existing tests and flows.

Accessibility Made Easy with Low-Code Test Automation

It's important to note that many accessibility issues on the frontend are ultimately connected to backend development practices. In other words, messy code equals inaccessible digital experiences. [Embedding automated accessibility checks](#) into development early and often ensures that quality and accessibility are proactively managed at all stages of the software development lifecycle.

For quality engineers already managing [growing software testing strategies](#), adding accessibility testing may seem like an overwhelming task. But new low-code test automation platforms like mabl are making it easier than ever to embed [accessibility checks](#) into existing end-to-end tests, drastically reducing the time to value and impact. Perhaps even more importantly, these new test automation solutions transform test data into shareable trend reports that simplify the [collaboration process](#) across the organization.



Start Building Inclusive Digital Experiences in Minutes with Low-Code Test Automation

Start embedding automated accessibility checks into your DevOps pipeline with mabl, the low-code test automation platform that makes it easy to build, integrate, and scale testing for more inclusive customer experiences. Build inclusive software in minutes with [mabl's free trial](#).