The Automated Accessibility Coverage Report

Why we need to change how we view accessibility testing coverage.

deque

Contents

Introduction

The core charter of digital accessibility testing professionals around the world is to help ensure that digital assets are accessible to all, including people with disabilities. There are two common success metrics these pros use:

1. Testing and resolution progress against Web Content Accessibility Guidelines (WCAG) Success Criteria.
2. Testing and resolution progress against the raw volume of issues, typically in the order of severity or impact.

Developed by the W3C, [WCAG Success Criteria](#) exist to provide guidance to us all, helping define what accessible conditions should look like. These guidelines fuel laws like Section 508, the Americans with Disabilities Act (ADA), Accessibility for Ontarians with Disabilities Act (AODA), the EU Web Accessibility Directive and European Accessibility Act (EAA). As such, many accessibility professionals use WCAG Success Criteria as a success metric, as they track toward legal compliance. Using a combination of automated and manual testing approaches, they report progress as Success Criteria are met/partially met and compliance coverage increases.

Based on 20 years of industry experience and thousands of client engagements, we believe that to truly make an immediate and sustainable long-term impact on your state of accessibility, the best method of measurement is total issues addressed in order of severity or impact. This does not negate the need for compliance tracking in any way, but it better enables organizations to move the needle, and build a user-experience focused culture.

In order to "do the most good" without disrupting existing processes, a combination of automated and manual testing procedures has become standard practice (excluding those who unknowingly purchase overlay tools). However, the amount of testing that can cover the issues found by WCAG is debated within the accessibility community. To help remove a stigma attached to automated testing, *it is our intention to disprove the widely accepted belief that automated accessibility testing only provides 20-30% of accessibility testing coverage.*[1][2][3]

This statistic is founded on an inaccurate definition that accessibility coverage is calculated by how many individual WCAG success criteria can be tested by automation. As a result, organizations new to digital accessibility are discouraged by the perceived value of automated testing, driving many of them to overlay tools or unsustainable manual efforts.

In this report, we'll analyze and present how real audit data reveals a higher accessibility coverage for automated testing.

[1] https://webaim.org/projects/million/
[2] https://accessibility.blog.gov.uk/2017/02/24/what-we-found-when-we-tested-tools-on-the-worlds-least-accessible-webpage/
[3] https://engineering.linkedin.com/blog/2020/automated-accessibility-testing

Accessibility Audit Data Sample

We compiled anonymized audit data from a large number of companies across various industries and geographies, spanning 13,000+ pages/page states, and nearly 300,000 issues. In an effort to provide an accurate representation of this audit data, this study concentrated on first-time audits, i.e. if a page/page state was tested multiple times during the study period, that page/page state was only counted once, and only issues from its first accessibility audit were included. This removes any unintended biases introduced by varying remediation priorities and schedules.
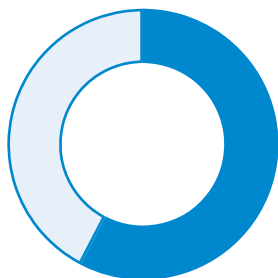
57.38% of Total Issues were detected using automated tests

The automated testing in this data set was done using the popular open-source axe-core rules library. It is important to note that axe-core puts great emphasis on not reporting "false-positives" or erroneous issues that may in fact not be issues at all. This study focused on HTML pages only and spans across various conformance standards like WCAG 2.0/2.1 Level A and AA.

If you'd like to look more into how we mapped coverage for both our automated and Intelligent Guided Testing Tools, you may dive into more detail in the Appendix of this paper.

In the report below, we will discuss what is accessibility testing coverage, how much digital accessibility can be covered by automation alone, how much coverage, and the impact of testing accuracy.

What is Accessibility Testing Coverage?

The State of the Market Today

How much coverage is provided by automated accessibility testing tools available today? Depending on who you are talking to, the answer to this question usually varies anywhere between 20 and 30 percent (again, assuming you throw out silly overlay claims). ***Many people in the industry today define coverage as the percentage of individual WCAG Success Criteria that can be tested using automated accessibility tools.*** The remaining coverage required to achieve compliance is achieved with manual testing.

Why is coverage important?

Today's agile development practices rely on automation to achieve maximum throughput for the product development teams. Digital accessibility is sometimes looked at as a non-functional requirement and is often deprioritized to meet business' critical 'functional' requirements. Development and QA managers need to budget and plan for resources ahead of time. The need to forecast how much work can be handled by automation, and how many manual resources will be needed to meet the product deliverables, timelines and budget.

It is often with this intent that the question of coverage is asked. The higher the number of issues that can be caught and addressed in earlier stages of product development, the lower the overall cost. Moreover, automated tools with high 'coverage' reduce the reliance on specialized skills and make it possible to 'mainstream' the development of an accessible product.

Accessibility Coverage: WCAG Criteria vs. Individual Issues

Looking at the percentage of WCAG success criteria is certainly one way to think about the 'coverage.' In our analysis we found automated issues for 16 out of the 50 Success Criteria under WCAG 2.1 Level AA. This supports the 20-30% automated coverage claims that many experts claim today. However, our analysis indicates that this definition does not accurately reflect the number of issues found in testing real web pages as they exist in the wild. In practice, some types of issues occur much more frequently than others, and these can result in a much higher percentage of total accessibility issues that can be discovered using automated tools.

In our studies, we looked at over 2,000 audits that were conducted using Deque's automated testing tools and manual testing methodology. *In the majority of the audits, we discovered that the number of issues found using automated tests formed a higher percentage of issues as compared to manual issues.*

We believe that the number of issues is a much better indicator of the level of effort required to address accessibility issues. We find that the volume of issues impacts the effort to address issues much more than the type of issue in most instances. For example, consider a web page with 10 missing field label associations. While it is one WCAG criteria, a developer (in most cases) has to address these issues one issue at a time. Therefore, the effort required to address the 10 missing field label associations, while may not be 10X the effort to fix one, is certainly much higher than the effort required to fix one missing field-label association.

> **In the majority of the audits, we discovered that the number of issues found using automated tests formed a higher percentage of issues as compared to manual issues.**

Some key findings from our analysis:

- On average across all the audits included in the sample data, we found that 57.38% of total issues were identified using Deque's automated tests.

- The top 5 issues categories (WCAG Success Criteria) accounted for over 78% of the total issues discovered, and a majority of these issues were discovered using automated testing.

- The top seven WCAG success criteria with the highest proportion of automated issues were (refer Table 2 in Appendix):

    - 3.1.1 Language of Page
    - 4.1.1 Parsing
    - 1.4.3 Contrast (Minimum)
    - 2.4.1 Bypass Blocks
    - 1.1.1 Non-Text Content
    - 4.1.2 Name, Role, Value
    - 1.3.1 Info and Relationships

It is worth noting that in the data we analyzed, the above seven categories accounted for over 80% of total issues recorded, with 1.4.3 Contrast (Minimum) accounting for about 30%.

> We believe that the number of issues is a much better indicator of the level of effort required to address accessibility issues. We find that the volume of issues impacts the effort to address issues much more than the type of issue in most instances.

Table 1: WCAG Success Criteria with the Most Issues

#	Success Criteria #	Success Criteria Name	Total issues	Manual issues	Auto issues	Manual %	Auto %	% of ALL Issues by SC	Cumulative % of Issues
1	1.4.3	Contrast (Minimum)	88,714	14,981	73,733	16.89%	83.11%	30.08%	30.08%
2	4.1.2	Name, Role, Value	48,287	22,011	26,276	45.58%	54.42%	16.37%	46.45%
3	1.3.1	Info and Relationships	36,382	19,950	16,432	54.83%	45.17%	12.33%	58.78%
4	4.1.1	Parsing	34,488	3,351	31137	9.72%	90.28%	11.69%	70.47%
5	1.1.1	Non-text Content	23,701	7,687	16,014	32.43%	67.57%	8.04%	78.51%
6	2.4.3	Focus Order	9,553	9,553	0	100.00%	0.00%	3.24%	81.75%
7	2.1.1	Keyboard	9,412	9,178	234	97.51%	2.49%	3.19%	84.94%
8	2.4.7	Focus Visible	7,312	7,312	0	100.00%	0.00%	2.48%	87.42%
9	1.4.11	Non-text Contrast	4,539	4,539	0	100.00%	0.00%	1.54%	88.96%
10	1.4.1	Use of Color	3,713	3,261	452	87.83%	12.17%	1.26%	90.22%
11	1.3.2	Meaningful Sequence	3,313	3,313	0	100.00%	0.00%	1.12%	91.34%
12	3.3.2	Labels or Instructions	2,537	2,019	518	79.58%	20.42%	0.86%	92.20%
13	2.4.1	Bypass Blocks	2,533	532	2,001	21.00%	79.00%	0.86%	93.06%
14	2.4.2	Page Titled	2,211	1,962	249	88.74%	11.26%	0.75%	93.81%
15	3.1.1	Language of Page	2,173	178	1,995	8.19%	91.81%	0.74%	94.54%
	#.#.#	Rest of WCAG 2.1 A/AA SC	16,090	15,889	201	98.75%	1.25%	5.46%	100.00%
		Totals	294,958	125,716	169,242	42.62%	57.38%		

How Much of Digital Accessibility Can Really Be Automated?

Automated accessibility testing is when a rules engine, such as axe-core, scans, or analyzes a web page for accessibility issues. These rules engines are built to test against accessibility standards, such as WCAG, which have predefined criteria for whether or not something is accessible. Automated testing tools can either be browser extensions, like axe DevTools, or they can be rules-engines built into automated test environments.

As previously mentioned, we analyzed 13,000+ pages/page states, and nearly 300,000 issues and found that **57.38% of issues from first-time audit customers could be found from automated testing.** Each data set will have a unique coverage percentage based on the number of issues that occur. We are confident in the accuracy of the coverage percentage from this data set, as it's from a large sample size and from a wide variety of first-time customers.

The Impact of Testing Accuracy

Not All Accessibility Tools Are Created Equal

The accuracy of accessibility tools depends on the collaboration of developers and the accessibility experts who create them.

When Deque reports issues using our axe-core powered tools, we exclude false positives. This means that any issues we cannot state are in fact issues with 100% certainty are not reported as such. False positives can waste time, erode trust and derail progress. Additionally, if a flagged item needs manual verification, or is a best practice, it is not included in the reported issues. This exclusion, while it reduces the total number, is important to ensure that we do not inflate the coverage percentage. This also helps us stay true to the initially stated intent of coverage to provide estimate, planning, and forecasting capabilities.

Repeat issues

Modern web pages very often include templates (like header, footer, navigation, etc.) repeated across multiple pages. Any accessibility issues present on these templates can most likely be fixed once and bring benefits to all the pages where they are included. Therefore, we account for issues on these common templates only once for our analysis.

For example, if a header had 8 issues that were repeated across 10 pages, instead of counting these as 80 issues our analysis includes only 8 issues. While this may not be an accurate representation of user experience on these 10 pages, it aligns more closely to effort required to fix the issues on the header. Counting all 80 issues will actually lead to an increase in the overall percentage of issues discovered.

In Summary

Accessibility coverage should not be generically defined by the number of WCAG Success Criteria that are covered, but by the volume of issues that can be covered in real-life examples. Our large sample size that covers a wide range of first-time audits provides us an accurate estimation of how much issue coverage to expect from automated and semi-automated accessibility tools.

This new coverage percentage of 57.38% for automated testing will give dev teams and accessibility experts a more accurate depiction of the value they'll receive from using automated tools.

If paied with an appropriate semi-automated testing approach, like the Intelligent Guided Tests offered in axe DevTools, this coverage can be increased even further.

As we all continue to make the web a better, more inclusive place, it is important to consider the role automation can have in helping us move the needle. By reconsidering how big an impact it can really make by accurately communicating the coverage it offers, you'll help remove any doubt from newcomers, helping put them on a path toward sustainable digital accessibility.

*Axe and Intelligent Guided Test are trademarks of Deque Systems, Inc.

Appendix

Automated Accessibility Data

Table 2: Issue Counts by Success Criteria, summarized by Automated, Manual, and Total Issues

	Success Criteria	Automated Issues	Manual Issues	Total Issues
1	1.1.1 Non-Text Content	16,014	7,687	23,701
2	1.2.1 Audio-only and Video-only (Prerecorded)	-	140	140
3	1.2.2 Captions (Prerecorded)	-	212	212
4	1.2.3 Audio Description or Media Alternative (Prerecorded)	-	120	120
5	1.2.4 Captions (Live)	-	7	7
6	1.2.5 Audio Description (Prerecorded)	-	98	98
7	1.3.1 Info and Relationships	16,432	19,950	36,382
8	1.3.2 Meaningful Sequence	-	3,313	3,313
9	1.3.3 Sensory Characteristics	-	570	570
10	1.3.4 Orientation	-	44	44
11	1.3.5 Identify Input Purpose	132	730	862
12	1.4.1 Use of Color	452	3,261	3,713
13	1.4.4 Resize Text*	1,668	2,099	3,767
14	1.4.2 Audio Control	-	3	3
15	1.4.3 Contrast (Minimum)	73,733	14,981	88,714
16	1.4.5 Images of Text	10	1,778	1,778
17	1.4.10 Reflow	-	1,181	1,181
18	1.4.11 Non-text Contrast	-	4,539	4,539
19	1.4.12 Text Spacing	15	657	672
20	1.4.13 Content on Hover or Focus	-	685	685
21	2.1.1 Keyboard	234	9,178	9,412
22	2.1.2 No Keyboard Trap	-	377	377
23	2.1.4 Character Key Shortcuts	-	3	3
24	2.2.1 Timing Adjustable	22	381	403
25	2.2.2 Pause, Stop, Hide	-	560	560
26	2.3.1 Three Flashes or Below Threshold	-	3	3
27	2.4.1 Bypass Blocks	2,001	532	2,533

Table 2 Continued

	Success Criteria	Automated Issues	Manual Issues	Total Issues
28	2.4.2 Page Titled	249	1,962	2,211
29	2.4.3 Focus Order	-	9,553	9,553
30	2.4.4 Link Purpose (In Context)	-	1,376	1,376
31	2.4.5 Multiple Ways	-	181	181
32	2.4.6 Headings and Labels	-	1,228	1,228
33	2.4.7 Focus Visible	-	7,312	7,312
34	2.5.1 Pointer Gestures	-	7	7
35	2.5.2 Pointer Cancellation	-	-	-
36	2.5.3 Label in Name	32	495	527
37	2.5.2 Pointer Cancellation	-	-	-
38	3.1.1 Language of Page	1,995	178	2,173
39	3.1.2 Language of Parts	-	317	317
40	3.2.1 On Focus	-	167	167
41	3.2.2 On Input	-	281	281
42	3.2.3 Consistent Navigation	-	17	17
43	3.2.4 Consistent Identification	-	10	10
44	3.3.1 Error Identification	-	668	668
45	3.3.2 Labels or Instructions	518	2,019	2,537
46	3.3.3 Error Suggestion	-	142	142
47	3.3.4 Error Prevention (Legal, Financial, Data)	-	15	15
48	4.1.1 Parsing	31,137	3,351	34,488
49	4.1.2 Name, Role, Value	26,276	22,011	48,287
50	4.1.3 Status Message	-	1,337	1,337

*Axe-core contained a rule for automatic checking of criteria '1.4.4 Resize Text'. This rule was lowered from a failure to a best practice in version 3.5. Therefore, as part of this analysis, these 1,668 failure issues automatically reported have been dropped and not included in the summary.

Table 3: Percentage of issues by WCAG Success Criteria, sorted by decreasing % Automated in Category

Success Criteria	% Automated in Category	% Automated of Total	% of Total Issues
3.1.1 Language of Page	91.81%	0.68%	0.74%
4.1.1 Parsing	90.28%	10.56%	11.69%
1.4.3 Contrast (Minimum)	83.11%	25.00%	30.08%
2.4.1 Bypass Blocks	79.00%	0.68%	0.86%
1.1.1 Non-Text Content	67.57%	5.43%	8.04%

Table 3 Continued

Success Criteria	% Automated in Category	% Automated of Total	% of Total Issues
4.1.2 Name, Role, Value	54.42%	8.91%	16.37%
1.3.1 Info and Relationships	45.17%	5.57%	12.33%
3.3.2 Labels or Instructions	20.42%	0.18%	0.86%
1.3.5 Identify Input Purpose	15.31%	0.04%	0.29%
1.4.1 Use of Color	12.17%	0.15%	1.26%
2.4.2 Page Titled	11.26%	0.08%	0.75%
2.5.3 Label in Name	6.07%	0.01%	0.18%
2.2.1 Timing Adjustable	5.46%	0.01%	0.14%
2.1.1 Keyboard	2.49%	0.08%	3.19%
1.4.12 Text Spacing	2.23%	0.01%	0.23%
1.2.1 Audio-only and Video-only (Prerecorded)	0.00%	0.00%	0.05%
1.2.2 Captions (Prerecorded)	0.00%	0.00%	0.07%
1.2.3 Audio Description or Media Alternative (Prerecorded)	0.00%	0.00%	0.04%
1.2.4 Captions (Live)	0.00%	0.00%	0.00%
1.2.5 Audio Description (Prerecorded)	0.00%	0.00%	0.03%
1.3.2 Meaningful Sequence	0.00%	0.00%	1.12%
1.3.3 Sensory Characteristics	0.00%	0.00%	0.19%
1.3.4 Orientation	0.00%	0.00%	0.01%
1.4.2 Audio Control	0.00%	0.00%	0.00%
1.4.4 Resize Text*	0.00%	0.00%	0.71%
1.4.5 Images of Text	0.00%	0.00%	0.60%
1.4.10 Reflow	0.00%	0.00%	0.40%
1.4.11 Non-text Contrast	0.00%	0.00%	1.54%
1.4.13 Content on Hover or Focus	0.00%	0.00%	0.23%
2.1.2 No Keyboard Trap	0.00%	0.00%	0.13%
2.1.4 Character Key Shortcuts	0.00%	0.00%	0.00%
2.2.2 Pause, Stop, Hide	0.00%	0.00%	0.19%
2.3.1 Three Flashes or Below Threshold	0.00%	0.00%	0.00%
2.4.3 Focus Order	0.00%	0.00%	3.24%
2.4.4 Link Purpose (In Context)	0.00%	0.00%	0.47%
2.4.6 Headings and Labels	0.00%	0.00%	0.42%
2.4.7 Focus Visible	0.00%	0.00%	2.48%
2.5.1 Pointer Gestures	0.00%	0.00%	0.00%
2.5.1 Pointer Cancellation	0.00%	0.00%	0.00%

Table 3 Continued

Success Criteria	% Automated in Category	% Automated of Total	% of Total Issues
2.5.4 Motion Actuation	0.00%	0.00%	0.00%
3.1.2 Language of Parts	0.00%	0.00%	0.11%
3.2.1 On Focus	0.00%	0.00%	0.06%
3.2.2 On Input	0.00%	0.00%	0.10%
3.2.3 Consistent Navigation	0.00%	0.00%	0.01%
3.2.4 Consistent Identification	0.00%	0.00%	0.00%
3.3.1 Error Identification	0.00%	0.00%	0.23%
3.3.3 Error Suggestion	0.00%	0.00%	0.05%
3.3.4 Error Prevention (Legal, Financial, Data)	0.00%	0.00%	0.01%
4.1.3 Status Message	0.00%	0.00%	0.45%