
# MEASURING WHAT MATTERS

A Practitioner's Guide to Tracking Accessibility Metrics Using Issue Tracking Data and the W3C Accessibility Maturity Model

**AFixt. Accessibility + Fixed**

afixt.com

# Table of Contents

# 1. Executive Summary

Organizations investing in digital accessibility often struggle to answer a deceptively simple question: Are we getting better? Despite growing regulatory pressure from the ADA, Section 508, the European Accessibility Act, and EN 301 549, most organizations lack the instrumentation to quantify their accessibility posture, track progress over time, or demonstrate return on investment to leadership.

This white paper presents a comprehensive, practitioner-focused framework for tracking accessibility metrics using data that already lives in your issue tracking system. It aligns each metric with the W3C Accessibility Maturity Model (AMM), an independently developed framework for evaluating organizational accessibility maturity and provides concrete implementation guidance using Jira as the reference platform.

Inside, you will find specific custom field configurations, ready-to-use JQL queries, dashboard blueprints, and automation recipes that transform raw issue tracking data into the proof points the W3C AMM requires. Whether you are launching a new accessibility program or scaling an established one, this guide will help you move from anecdote-driven reporting to evidence-based accessibility management.

> *Throughout this paper, you will see references to services and practices that AFixt, provides to organizations building or maturing their accessibility programs. AFixt's approach emphasizes hands-on remediation, defensible audits, strategic consulting, and training - all grounded in the kind of metrics-driven program management described here.*

# 2. The Challenge: Why Accessibility Metrics Matter

Accessibility programs fail for many reasons, but among the most common is the inability to demonstrate measurable progress. When accessibility exists only as a compliance checkbox or a vague aspiration, it is easily deprioritized in favor of feature work with clearer ROI stories. Metrics change this dynamic fundamentally.

## 2.1 The Cost of Inattention

Without metrics, organizations face several compounding problems. Remediation efforts lack direction because teams cannot distinguish between high-impact systemic issues and isolated edge cases. Leadership cannot evaluate whether the program is working, making budget justification an exercise in storytelling rather than evidence. Legal counsel cannot assess risk posture accurately, and procurement teams cannot verify vendor accessibility claims. Perhaps most critically, the people who depend on accessible products - users with disabilities - continue to encounter barriers that could have been identified and addressed months earlier.

## 2.2 What Good Metrics Look Like

Effective accessibility metrics share several characteristics. They are quantifiable, meaning they can be expressed as numbers rather than subjective assessments. They are comparable over time, allowing teams to detect trends and measure the impact of interventions. They are actionable, pointing to specific areas that need attention rather than producing a single pass-fail verdict. And they are aligned with a recognized framework, which provides external credibility and a shared vocabulary for discussing progress.

The W3C Accessibility Maturity Model provides exactly this kind of framework, and issue tracking systems like Jira contain exactly the kind of data needed to populate it.

## 2.3 The Issue Tracker as Accessibility Observatory

Most organizations already track accessibility issues in their bug tracking or project management systems. What they lack is the structure to transform that data into meaningful metrics. A Jira instance with hundreds of accessibility bugs tells you very little on its own. But the same data, properly categorized with custom fields for WCAG success criteria, severity, affected user groups, discovery method, and resolution status, becomes a rich source of program intelligence.

This paper shows you how to build that structure, populate it with the right data, and extract insights that map directly to the W3C AMM’s proof points.

# 3. The W3C Accessibility Maturity Model: A Framework for Measurement

The W3C Accessibility Maturity Model (AMM) provides a structured approach for organizations to evaluate and improve their accessibility practices across seven dimensions. Unlike audit-centric approaches that focus solely on technical conformance, the AMM examines whether accessibility is embedded into organizational culture, processes, and capabilities.

## 3.1 The Four Maturity Levels

The AMM defines four progressive maturity levels, each building upon the previous one:

| Maturity Level | Description | Key Characteristic |
|---|---|---|
| Inactive | Accessibility is not yet part of organizational practice. | No formal processes, no dedicated resources, ad-hoc or absent effort. |
| Launch | The organization has begun formalizing accessibility efforts. | Initial policies exist, some training has occurred, basic tracking is in place. |
| Integrate | Accessibility is embedded into standard workflows and processes. | Accessibility is part of Definition of Done, regular testing occurs, metrics are tracked. |
| Optimize | The organization continuously improves based on data and feedback. | Data-driven decisions, proactive prevention, continuous improvement culture. |

## 3.2 The Seven Dimensions

Each dimension represents a functional area of the organization where accessibility maturity can be assessed and improved:

1. Communications – Are internal and external communications delivered in accessible formats? This includes documents, presentations, social media, video content, and marketing materials.
2. Knowledge and Skills – Does the organization maintain sufficient accessibility expertise? This encompasses training programs, certification tracking, role-based competency, and skill gap analysis.
3. Support – Can the organization support employees and customers who need accessibility accommodations? This covers help desk capabilities, feedback mechanisms, and accommodation processes.

4. ICT Development Lifecycle – Is accessibility integrated into design, development, testing, and release? This is often the most metric-rich dimension, covering defect tracking, testing coverage, release gates, and remediation velocity.
5. Personnel – Are hiring practices inclusive? This includes accessible job applications, diverse recruiting, workplace accommodations, and employee resource groups.
6. Procurement – Does the organization acquire accessible solutions? This covers vendor evaluation criteria, contract language, VPAT/ACR review processes, and procurement policy.
7. Culture – Does leadership prioritize accessibility? This includes executive sponsorship, published accessibility policy, budget allocation, and organizational values.

## 3.3 Proof Points: The Currency of Maturity

Advancement through maturity levels is contingent upon achieving specific outcomes for each dimension. The AMM validates these outcomes through proof points - tangible, objective pieces of evidence that demonstrate an organization's commitment and capability. A proof point might be a published accessibility policy, a training completion record, a dashboard showing defect resolution trends, or a procurement contract with accessibility clauses.

This is where issue tracking data becomes invaluable. Many of the proof points required for the ICT Development Lifecycle, Support, and Knowledge and Skills dimensions can be generated directly from a well-structured Jira instance. The remaining dimensions can be supported through complementary tracking within the same system.


*Figure 6: W3C Accessibility Maturity Model radar chart showing current state vs. target across all seven dimensions*

> *AFixt's strategic consulting practice helps organizations map their current state against the W3C AMM, identify gaps across all seven dimensions, and build roadmaps for advancement. This assessment process forms the foundation for the metrics system described in the following sections.*

# 4. Designing Your Accessibility Metrics System in Jira

A well-designed Jira configuration transforms your issue tracker from a simple bug list into a comprehensive accessibility metrics platform. The goal is to capture enough structured data on each issue to support the metrics described in subsequent sections, without burdening contributors with excessive data entry.

## 4.1 Custom Fields for Accessibility Issues

The following custom fields should be added to your Jira instance to enable accessibility-specific tracking. Each field is mapped to the W3C AMM dimension it supports.

### WCAG Success Criterion

Field type: Select list (single). This field captures which specific WCAG success criterion the issue violates. Populate the list with all WCAG 2.2 Level A and AA success criteria (e.g., “1.1.1 Non-text Content,” “1.3.1 Info and Relationships,” “4.1.2 Name, Role, Value”). This field is essential for conformance gap analysis and for understanding which types of issues recur most frequently.

AMM Dimension: ICT Development Lifecycle

### WCAG Conformance Level

Field type: Select list (single). Options: Level A, Level AA, Level AAA. This allows quick filtering to focus on issues that affect the target conformance level, typically AA for most organizations.

AMM Dimension: ICT Development Lifecycle

### Accessibility Severity

Field type: Select list (single). This field captures the severity of the accessibility barrier from the user’s perspective, distinct from Jira’s built-in Priority field which often reflects business priority.

| Severity | Definition | Example |
|---|---|---|
| Critical | Blocks users from completing essential tasks. No workaround exists. | Submit button cannot be activated via keyboard; screen reader cannot access main navigation. |
| High | Major usability barrier with limited or cumbersome workarounds. | Form fields lack programmatic labels; screen reader users must guess at field purpose. |
| Medium | Reduces usability but does not prevent task completion. | Low-contrast text fails WCAG AA ratios, but content is still discernible. |
| Low | Minor issue with minimal impact on task completion. | Decorative image has empty alt attribute instead of null alt; redundant ARIA role on semantic element. |

AMM Dimension: ICT Development Lifecycle, Support

### Affected User Group

Field type: Checkboxes (multi-select). Options should include: Blind/Screen Reader Users, Low Vision Users, Deaf/Hard of Hearing Users, Motor/Mobility Impaired Users, Cognitive/Learning Disabilities, All Users. This field enables user-impact analysis and helps prioritize issues that affect the broadest population.

AMM Dimension: ICT Development Lifecycle, Support, Personnel

### Discovery Method

Field type: Select list (single). Options: Automated Testing, Manual Expert Audit, Assistive Technology Testing, User-Reported, Design Review, Code Review, CI/CD Pipeline. This field tracks how issues are found, which is critical for evaluating the effectiveness of your testing strategy and for identifying gaps in your quality assurance process.

AMM Dimension: ICT Development Lifecycle, Knowledge and Skills

### Component/Product Area

Field type: Select list or Jira component. Map this to the areas of your product (e.g., Navigation, Forms, Checkout, Dashboard, Media Player). This enables per-component accessibility scoring and helps identify which teams or product areas need the most support.

AMM Dimension: ICT Development Lifecycle

### Remediation Complexity

Field type: Select list (single). Options: Low (quick fix, under 2 hours), Medium (requires focused development, 2–8 hours), High (requires design changes or architectural work, 8+ hours). This supports capacity planning and helps predict remediation timelines.

### Accessibility Testing Status

Field type: Select list (single). Options: Not Tested, Automated Only, Manual Test Complete, Assistive Tech Verified, User Tested. This progressive field helps track testing depth per issue and contributes to testing coverage metrics.

AMM Dimension: ICT Development Lifecycle

> *AFixt's audit methodology produces issue reports that map directly to these custom fields. When AFixt performs an accessibility audit, the resulting issues are delivered in a format that can be imported directly into a pre-configured Jira instance, saving teams weeks of manual data entry and ensuring consistent field population from day one.*

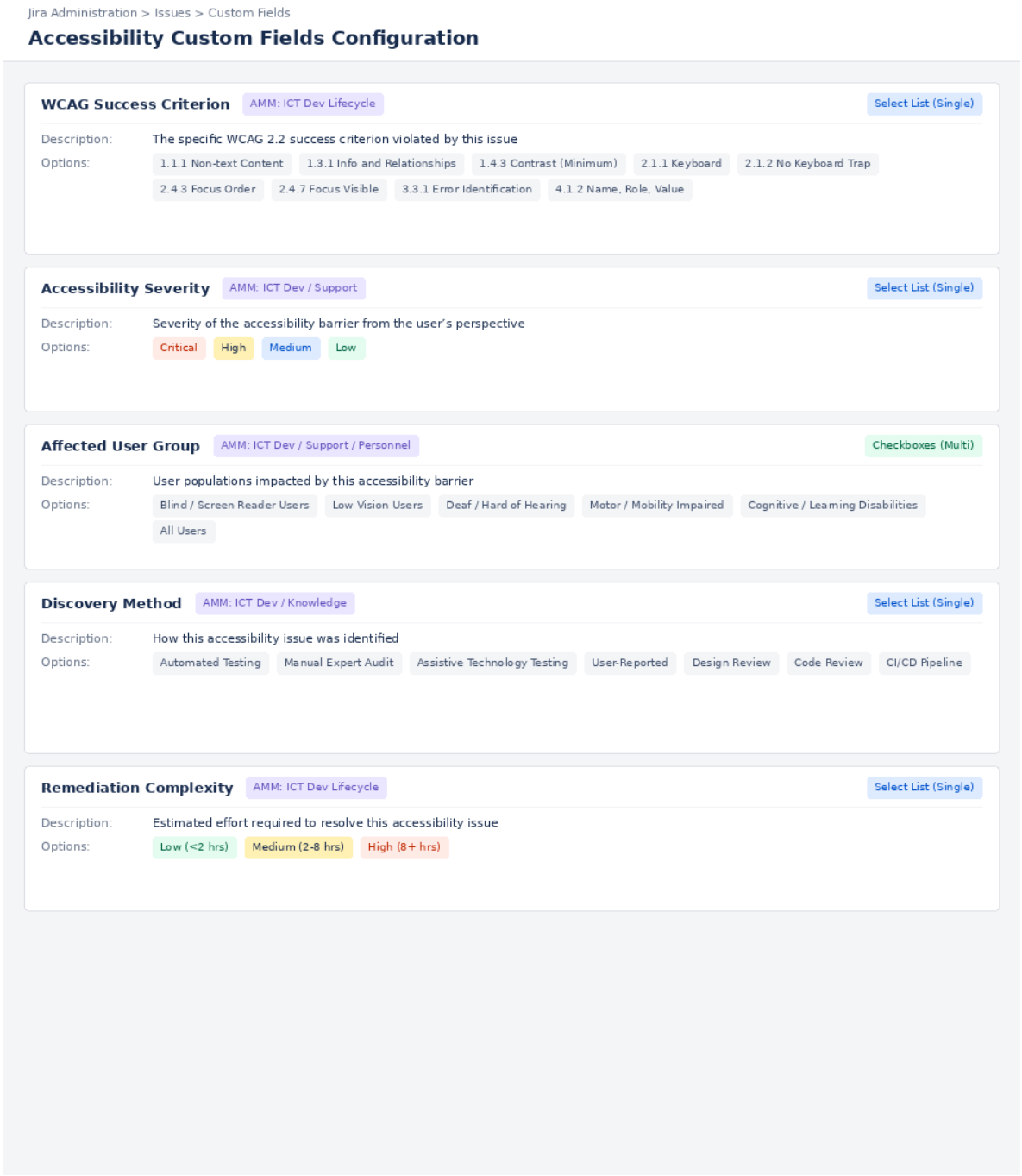


*Figure 1: Accessibility custom fields configuration in Jira, with AMM dimension mapping*

## 4.2 Issue Types and Workflows

### Recommended Issue Types

Most organizations benefit from creating a dedicated Accessibility Defect issue type. This keeps accessibility issues visible without complicating the standard Bug workflow. The Accessibility Defect type should include all the custom fields described above, plus standard fields like Description, Steps to Reproduce, Expected vs. Actual Behavior, Screenshots, and Affected URL or Screen.

Some organizations prefer to use their existing Bug issue type and distinguish accessibility issues through a label or component. Either approach works, but the dedicated issue type provides cleaner filtering and dashboard construction.

### Workflow Configuration

An effective accessibility workflow captures the full lifecycle of a defect from identification through verified resolution. A recommended workflow follows this progression:

| Status | Description | Who Acts |
|---|---|---|
| Open | Issue has been logged and is awaiting triage. | Triage lead / Accessibility SME |
| Triaged | Issue has been assessed for severity and assigned to a team. | Accessibility lead |
| In Progress | Developer or designer is actively working on a fix. | Assigned developer |
| In Review | Fix is in code review or design review. | Peer reviewer |
| Ready for Retest | Fix has been deployed to a test environment. | QA / Accessibility tester |
| Verified | Fix has been confirmed through manual and/or assistive technology testing. | Accessibility tester |
| Closed | Issue is resolved and verified. No further action needed. | Triage lead |
| Deferred | Issue has been acknowledged but deprioritized with documented justification. | Product manager |

Tracking transitions between these statuses enables metrics like mean time to resolution, bottleneck identification (e.g., issues stuck in “Ready for Retest” indicate insufficient testing capacity), and workflow efficiency analysis.

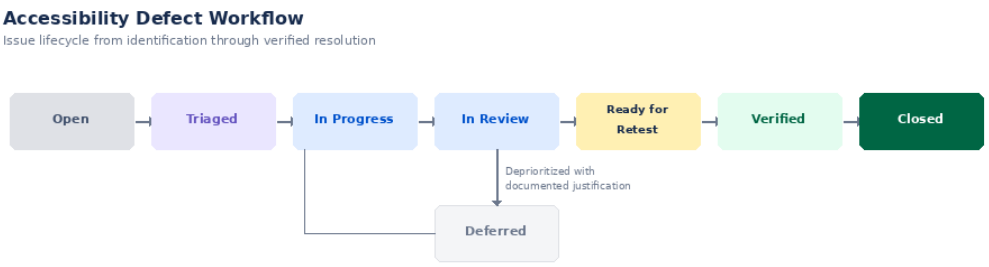


*Figure 2: Accessibility defect workflow showing the progression from Open through Verified/Closed, with a Deferred branch*

## 4.3 Labels and Components Strategy

In addition to custom fields, use Jira labels and components strategically to enable cross-cutting analysis:

- Labels: Use labels like “accessibility,” “a11y-audit-q1-2025,” “screenreader,” “keyboard,” “color-contrast,” “focus-management,” and “aria” to enable ad hoc filtering and cross-issue pattern detection.
- Components: Map Jira components to your product architecture. This allows you to answer questions like “Which component has the most open Critical accessibility defects?” and “Which team’s components have the fastest remediation velocity?”
- Epics or Initiatives: Group accessibility work under dedicated epics like “Q1 Accessibility Remediation Sprint” or “Keyboard Navigation Overhaul.” This provides a portfolio-level view of accessibility investment.
- Fix Versions: Tag resolved issues with the release version in which they were fixed. This enables “accessibility changelog” reporting and contributes to the Communications dimension of the AMM.

# 5. Metrics That Map to Maturity: Dimension by Dimension

This section maps specific, measurable metrics to each of the seven W3C AMM dimensions. For each dimension, you will find the metric definition, what it measures, how to capture it in Jira, and a JQL query example for extracting the data.

## 5.1 ICT Development Lifecycle Metrics

This is the most data-rich dimension, encompassing the design, development, testing, and release of digital products. Issue tracking data provides direct evidence for most proof points in this dimension.

### Metric: Open Defect Count by Severity

What it measures: The current accessibility defect backlog, segmented by severity level. A declining trend in Critical and High severity issues indicates program effectiveness.

JQL:

```
project = "MYPROJECT" AND issuetype = "Accessibility Defect" AND status NOT IN (Closed, Verified) ORDER BY "Accessibility Severity" ASC
```

### Metric: Defect Discovery Rate

What it measures: The number of new accessibility defects discovered per sprint or per month. An initial spike followed by a gradual decline is the healthy pattern - it means your testing is thorough (high discovery) and your prevention practices are working (declining new defects).

JQL (defects created in the last 30 days):

```
project = "MYPROJECT" AND issuetype = "Accessibility Defect" AND created >= -30d
```


*Figure 13: Defect discovery rate trend showing the healthy pattern - initial audit spike followed by gradual decline as prevention practices mature*

### Metric: Defect Resolution Rate

What it measures: The percentage of accessibility defects resolved within a defined time period. Compare this to the discovery rate to determine whether the backlog is growing or shrinking.

JQL (defects resolved in the last 30 days):

```
project = "MYPROJECT" AND issuetype = "Accessibility Defect" AND status
CHANGED TO Closed DURING (-30d, now())
```

### Metric: Mean Time to Resolution (MTTR)

What it measures: The average number of days between issue creation and issue resolution. Segment by severity to identify whether Critical issues are being resolved faster than lower-severity ones (they should be). MTTR is a powerful leading indicator of program maturity.

How to capture: Use Jira's built-in "Time in Status" reports or a marketplace plugin like Time to SLA. JQL cannot calculate averages directly, but you can export data for analysis:

```
project = "MYPROJECT" AND issuetype = "Accessibility Defect" AND status =
Closed AND resolved >= -90d ORDER BY created ASC
```

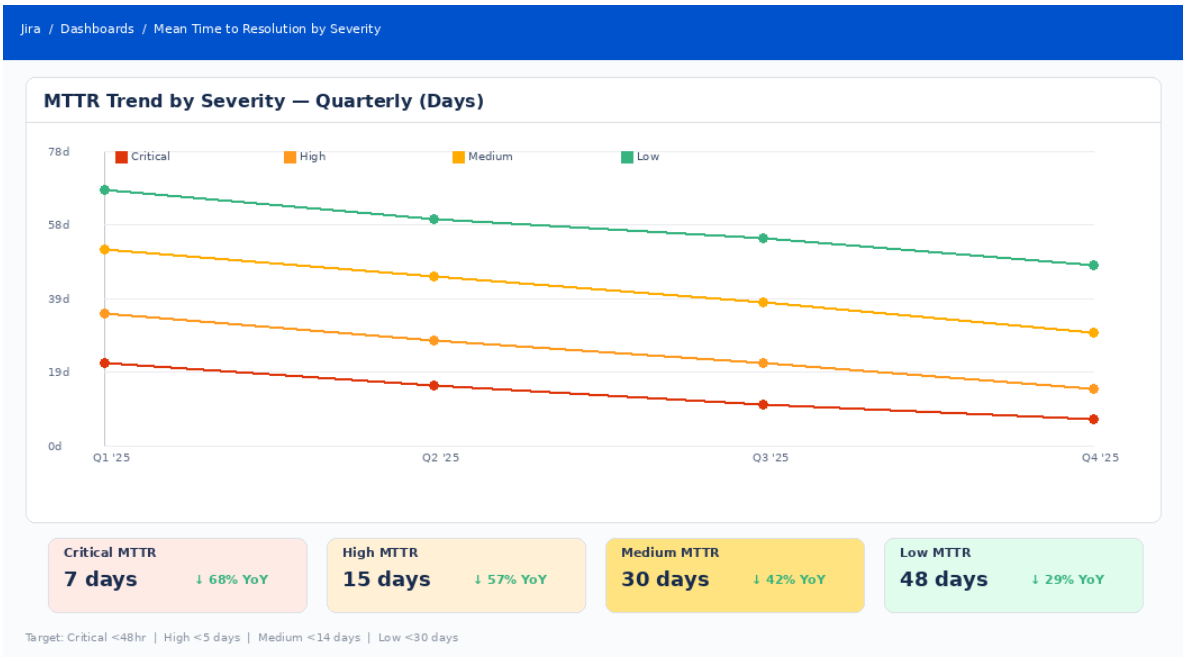


*Figure 12: Mean Time to Resolution trend by severity level, showing quarterly improvement across all categories*

### Metric: Defect Density by Component

What it measures: The number of accessibility defects per product component. Components with disproportionately high defect counts may indicate teams that need additional training, components that are architecturally problematic, or areas of the product that receive insufficient testing.

JQL (Critical and High defects by component):

```
project = "MYPROJECT" AND issuetype = "Accessibility Defect" AND
"Accessibility Severity" IN (Critical, High) AND component IS NOT EMPTY
ORDER BY component ASC
```

### Metric: WCAG Success Criterion Failure Distribution

What it measures: Which WCAG success criteria are violated most frequently. This reveals systemic patterns. If 40% of your defects violate SC 1.3.1 (Info and Relationships), that tells you your teams need targeted training on semantic HTML and ARIA. If SC 4.1.2 (Name, Role, Value) dominates, your component library likely needs attention.

JQL (group by WCAG SC):

```
project = "MYPROJECT" AND issuetype = "Accessibility Defect" AND "WCAG
Success Criterion" IS NOT EMPTY ORDER BY "WCAG Success Criterion" ASC
```

### Metric: Testing Coverage by Method

What it measures: How issues are being discovered. If nearly all issues come from automated testing, your manual and assistive technology testing coverage is likely insufficient. Conversely, if most issues come from user reports, your proactive testing program needs strengthening.

JQL (issues discovered by automated testing only):

```
project = "MYPROJECT" AND issuetype = "Accessibility Defect" AND "Discovery Method" = "Automated Testing"
```

### Metric: Accessibility Release Gate Pass Rate

What it measures: The percentage of releases that pass the accessibility release gate on the first attempt. A low pass rate indicates that accessibility is not being addressed early enough in the development cycle.

How to capture: Create a Jira issue type or label for “Release Gate Review” and track pass/fail outcomes. Query:

```
project = "MYPROJECT" AND labels = "a11y-release-gate" AND resolution = "Pass" AND fixVersion >= "2025.1"
```

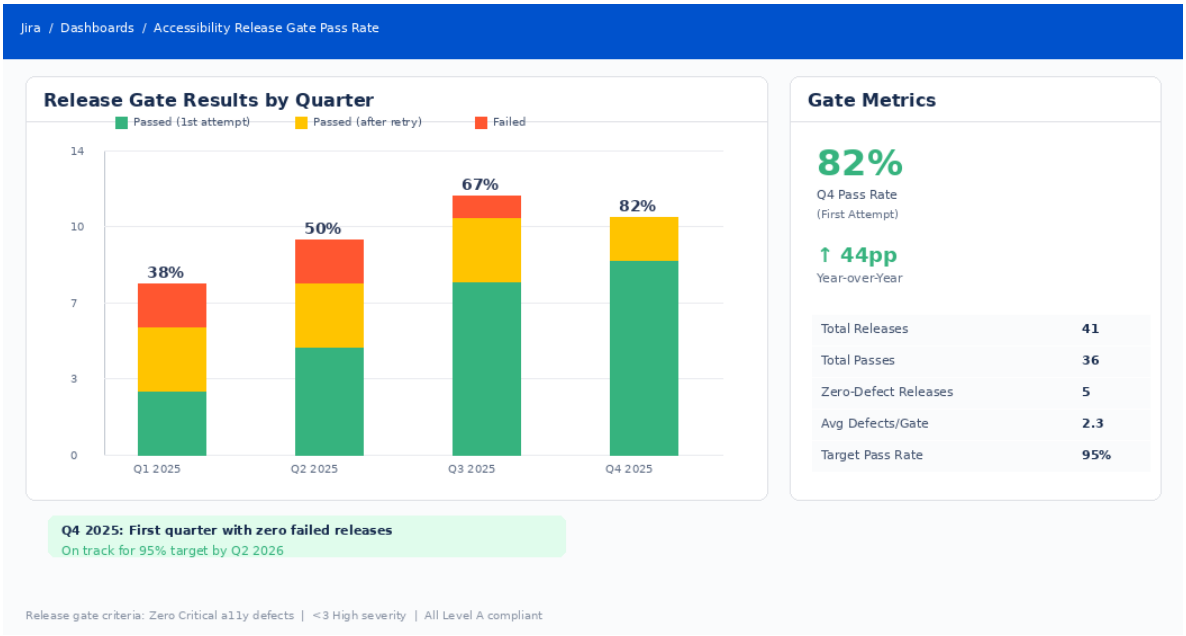


*Figure 14: Accessibility release gate pass rate by quarter, showing first-attempt passes, retries, and failures with trend toward 95% target*

### Metric: Regression Rate

What it measures: The percentage of previously resolved accessibility defects that reopen. A high regression rate indicates insufficient testing, inadequate fixes, or systemic architectural issues.

JQL (reopened accessibility defects):

```
project = "MYPROJECT" AND issuetype = "Accessibility Defect" AND status
CHANGED FROM Closed TO Open
```

> *AFixt's accessibility audits provide a complete initial defect inventory, categorized by WCAG success criterion, severity, affected user group, and recommended fix. This baseline dataset populates the ICT Development Lifecycle metrics from day one, giving organizations an immediate snapshot of their conformance posture and a foundation for tracking improvement.*

## 5.2 Culture and Leadership Metrics

Culture metrics assess whether accessibility is supported from the top of the organization. While much of this dimension is qualitative, several proof points can be tracked through Jira or complementary tools.

### Metric: Executive Accessibility Actions Logged

What it measures: Tangible actions taken by executive sponsors, such as reviewing accessibility reports, approving accessibility budgets, or communicating accessibility priorities. Create a Jira issue type or task category for “Accessibility Leadership Action” and log these as they occur.

JQL:

```
project = "A11Y-PROGRAM" AND issuetype = "Task" AND labels = "executive-action" AND created >= startOfYear()
```

### Metric: Accessibility Budget Allocation

What it measures: The percentage of overall IT budget dedicated to accessibility activities (tooling, training, remediation, testing, consulting). Track this as a custom field on a quarterly “Accessibility Program Review” issue.

### Metric: Policy Publication and Review Cadence

What it measures: Whether the organization has a published accessibility policy and how frequently it is reviewed. Track the policy review cycle as a recurring Jira task with defined due dates. An overdue policy review task is a visible signal that the Culture dimension needs attention.

## 5.3 Knowledge and Skills Metrics

This dimension measures the organization’s investment in accessibility expertise and training.

### Metric: Training Completion Rate by Role

What it measures: The percentage of employees who have completed accessibility training, segmented by role (developer, designer, QA, product manager, content author). Create Jira tasks for each training cohort and track completion.

JQL:

```
project = "A11Y-PROGRAM" AND issuetype = "Training" AND labels = "a11y-training" AND status = "Completed" AND "Role" = "Developer"
```

### Metric: Certification Count

What it measures: The number of team members who hold recognized accessibility certifications (e.g., IAAP CPACC, WAS, CPWA). Track certifications as Jira issues with expiration dates to ensure renewal tracking.

### Metric: Knowledge Assessment Scores

What it measures: Pre- and post-training assessment scores that demonstrate actual skill improvement. Track these as custom fields on training-related issues.

> *AFixt's Accessibility Bootcamp provides comprehensive, role-based training for developers, designers, QA testers, and content authors. Training completion data integrates directly into your Knowledge and Skills metrics, and AFixt's training methodology aligns with IAAP certification objectives.*

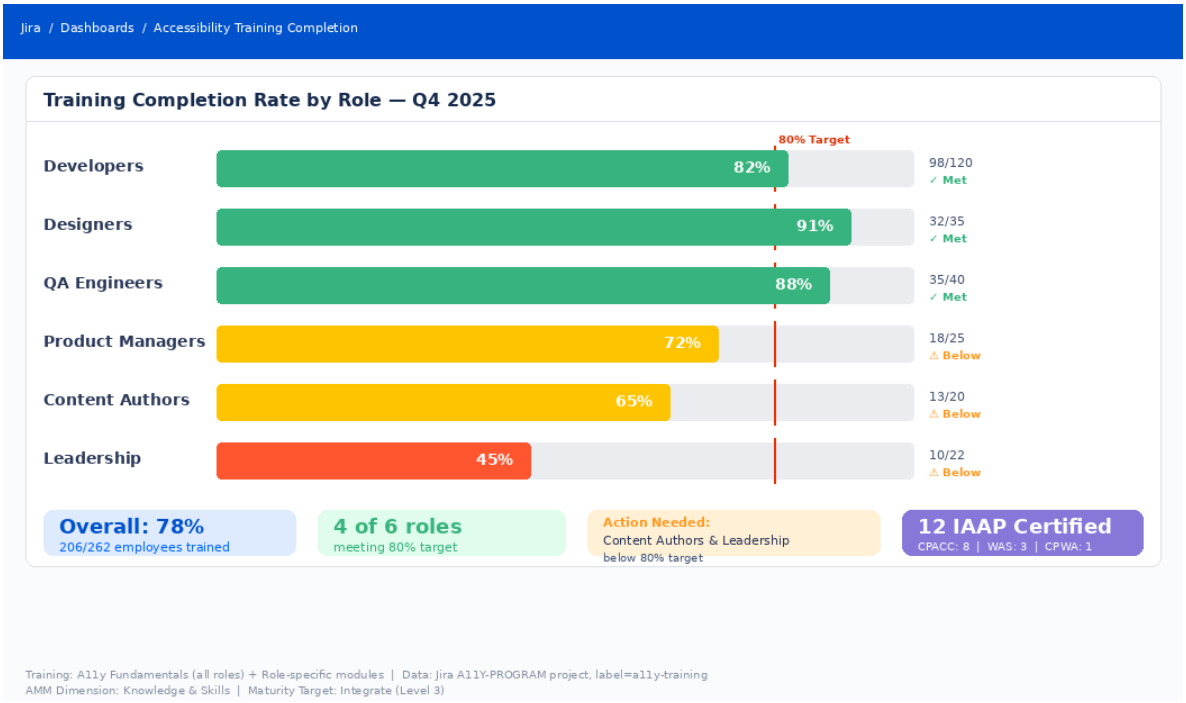


*Figure 15: Training completion rate by role with 80% target threshold, IAAP certification counts, and action items for underperforming groups*

## 5.4 Support Metrics

Support metrics evaluate how well the organization helps employees and customers who encounter accessibility barriers.

### Metric: Accessibility Support Ticket Volume and Resolution Time

What it measures: The number of accessibility-related support tickets received, average response time, and average resolution time. Use labels or a custom field to identify accessibility-related support tickets.

JQL (open accessibility support tickets):

```
project = "SUPPORT" AND labels = "accessibility" AND status NOT IN (Closed, Resolved) ORDER BY created ASC
```

### Metric: User Satisfaction Score for Accessibility Support

What it measures: Satisfaction ratings from users who submitted accessibility-related feedback or support requests. Track this through Jira Service Management's built-in satisfaction surveys or a custom field.

Target: 85% or higher satisfaction rate. The W3C AMM requires evidence that support processes are effective and that user feedback is incorporated into product improvement.

### Metric: Feedback Implementation Rate

What it measures: The percentage of accessibility feedback items that result in product changes. This demonstrates that the organization is not merely collecting feedback but acting on it.

JQL (feedback items that resulted in resolved defects):

```
project = "MYPROJECT" AND issuetype = "Accessibility Defect" AND "Discovery Method" = "User-Reported" AND status = Closed
```

> *AFixt's Accessibility Help Desk service provides organizations with on-demand accessibility expertise to supplement their internal support capabilities. Help desk interactions generate structured data that feeds directly into the Support dimension metrics, including response time, resolution time, and issue categorization.*

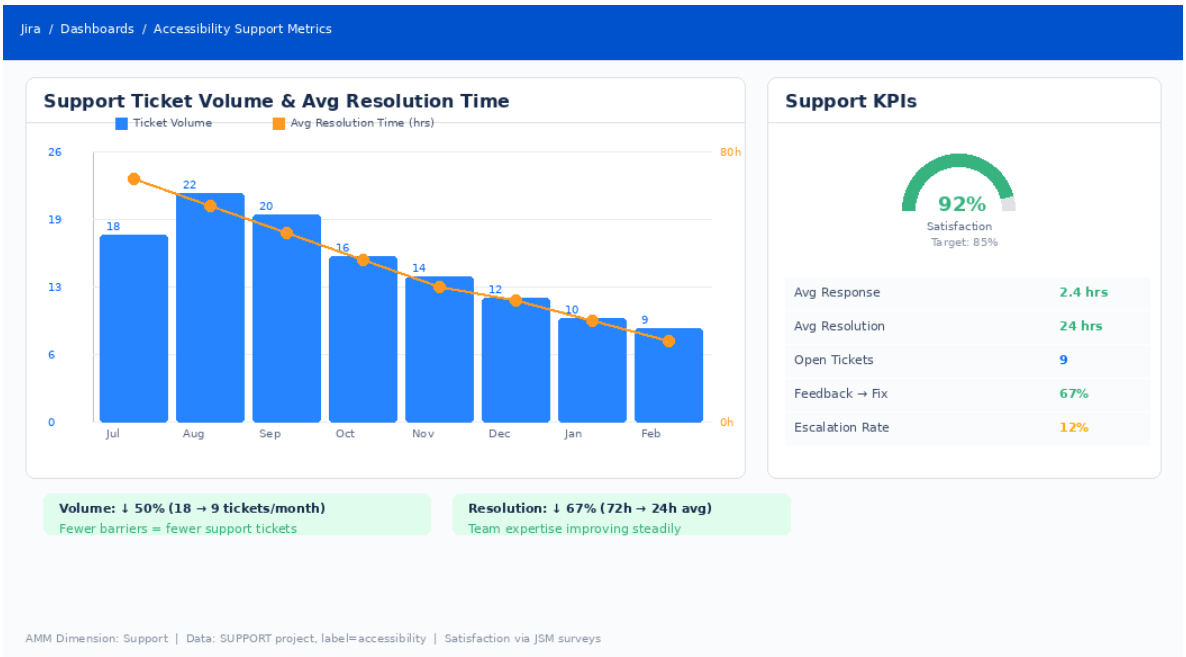


*Figure 16: Accessibility support metrics showing ticket volume decline, resolution time improvement, 92% satisfaction score, and feedback implementation rate*

## 5.5 Procurement Metrics

Procurement metrics ensure that the organization evaluates accessibility when acquiring third-party products and services.

### Metric: Vendor Accessibility Evaluation Rate

What it measures: The percentage of vendor/product evaluations that include an accessibility assessment. Create a Jira project or issue type for procurement evaluations and include a custom field for “Accessibility Evaluation Complete: Yes/No.”

JQL:

```
project = "PROCUREMENT" AND issuetype = "Vendor Evaluation" AND
"Accessibility Evaluation" = "Yes" AND created >= startOfYear()
```

### Metric: Third-Party Defect Tracking

What it measures: The number and severity of accessibility defects attributable to third-party components (e.g., embedded widgets, SaaS platforms, component libraries). Use a label or custom field to tag issues originating from third-party products.

JQL:

```
project = "MYPROJECT" AND issuetype = "Accessibility Defect" AND labels =
"third-party" AND status NOT IN (Closed, Verified)
```

### Metric: Contract Accessibility Clause Inclusion Rate

What it measures: The percentage of new vendor contracts that include accessibility requirements, conformance warranties, and remediation obligations. Track this as a custom field on procurement-related issues.

## 5.6 Communications Metrics

Communications metrics track whether the organization produces accessible content across all channels.

### Metric: Document Accessibility Compliance Rate

What it measures: The percentage of published documents (PDFs, Word documents, presentations) that meet accessibility standards. Log document accessibility reviews as Jira tasks with pass/fail outcomes.

JQL:

```
project = "A11Y-PROGRAM" AND labels = "doc-accessibility" AND resolution =
"Pass" AND created >= startOfYear()
```

### Metric: Multimedia Accessibility Rate

What it measures: The percentage of published video and audio content that includes captions, transcripts, and audio descriptions. Track multimedia accessibility as Jira tasks linked to content publication workflows.

## 5.7 Personnel Metrics

Personnel metrics evaluate the organization's hiring practices and workplace inclusion for people with disabilities.

### Metric: Job Application Platform Accessibility Score

What it measures: The results of periodic accessibility audits of the organization's job application system. Track audit findings as Jira issues linked to the Personnel dimension.

### Metric: Accommodation Request Fulfillment Rate

What it measures: The percentage of workplace accommodation requests that are fulfilled within a defined timeframe. This demonstrates that the organization's HR processes support employees with disabilities.

# 6. JQL Recipes: Extracting Accessibility Data from Jira

This section provides a curated library of JQL queries organized by use case. These queries assume the custom field configuration described in Section 4. Adapt field names and project keys to match your Jira instance.

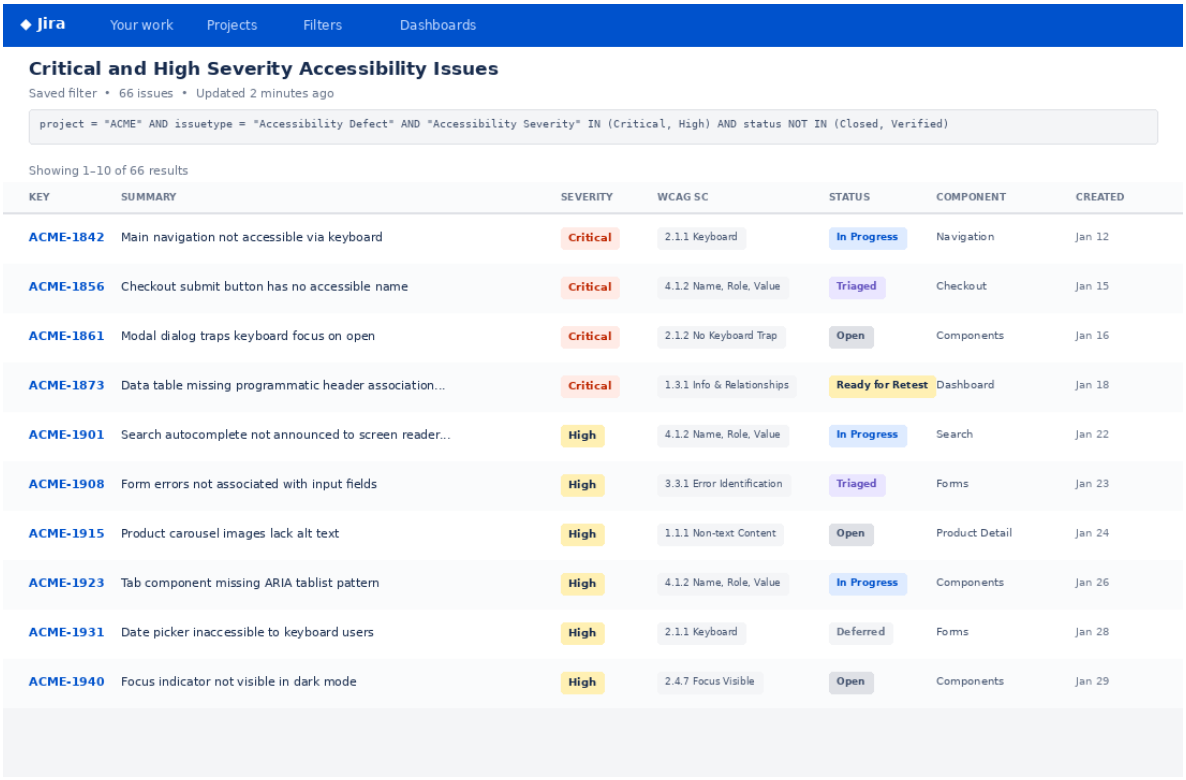


*Figure 3: Jira filter results view showing Critical and High severity accessibility defects with WCAG criteria, status, and component data*

## 6.1 Executive Dashboard Queries

### Current Accessibility Backlog Summary

```
project = "MYPROJECT" AND issuetype = "Accessibility Defect" AND status NOT IN (Closed, Verified, Deferred) ORDER BY "Accessibility Severity" ASC, created ASC
```

This query returns every open accessibility defect that has not been resolved or intentionally deferred, sorted so the most severe issues appear first. It provides the single most important number for executive reporting: the total active backlog. Use it as a Jira filter behind a “Filter Results” gadget on your executive dashboard, or feed it into a pie chart gadget grouped by the Accessibility Severity field to show the severity distribution at a glance. A shrinking total over successive months is the clearest indicator of program momentum.

### Critical and High Severity Issues Requiring Immediate Attention

```
project = "MYPROJECT" AND issuetype = "Accessibility Defect" AND
"Accessibility Severity" IN (Critical, High) AND status NOT IN (Closed,
Verified) ORDER BY "Accessibility Severity" ASC, updated DESC
```

This query narrows the backlog to only the issues that represent the greatest risk - defects that block users from completing tasks or that impose major usability barriers. Ordering by updated date (descending) surfaces issues that have had recent activity, which helps executives see which high-priority items are actively being worked versus which may be stalled. If this list is long and not declining, it signals that the remediation program needs more resources or better prioritization discipline.

### Accessibility Issues Resolved This Quarter

```
project = "MYPROJECT" AND issuetype = "Accessibility Defect" AND status IN
(Closed, Verified) AND resolved >= startOfQuarter()
```

This query counts the defects resolved within the current quarter, using Jira's built-in startOfQuarter() function. It provides the resolution velocity that executives need to evaluate whether the team's output matches expectations. Compare this number quarter-over-quarter to demonstrate acceleration. Pairing this query with the backlog summary shows whether the program is resolving issues faster than new ones are being discovered - the fundamental question for any accessibility program.

### Overdue Accessibility Issues

```
project = "MYPROJECT" AND issuetype = "Accessibility Defect" AND duedate <
now() AND status NOT IN (Closed, Verified, Deferred) ORDER BY duedate ASC
```

This query identifies accessibility defects whose due dates have passed without resolution. It requires that your workflow includes setting due dates, either manually during triage or automatically via Jira automation rules based on severity (e.g., Critical issues get a 5-business-day due date). Overdue issues sorted by due date ascending puts the longest-overdue items first. This query is particularly useful for compliance reporting, as it provides objective evidence of whether the organization is meeting its own SLA commitments for accessibility remediation.

## 6.2 Development Team Queries

### My Team's Open Accessibility Issues

```
project = "MYPROJECT" AND issuetype = "Accessibility Defect" AND component
IN ("Navigation", "Checkout") AND status NOT IN (Closed, Verified) ORDER BY
"Accessibility Severity" ASC
```

This query filters accessibility defects to specific Jira components, making it a team-level view of what needs attention. Replace the component values with those that correspond to your team's ownership areas. Each team should create a saved filter with their own components and pin it to their team dashboard. This decentralized approach makes accessibility work visible within the teams

that own it, rather than keeping it siloed within a central accessibility function. Sorting by severity ensures the team works on the most impactful issues first.

### Issues Ready for Accessibility Retest

```
project = "MYPROJECT" AND issuetype = "Accessibility Defect" AND status = "Ready for Retest" ORDER BY "Accessibility Severity" ASC, updated ASC
```

This query creates a work queue for accessibility testers by surfacing all issues where a developer has completed a fix and moved the issue to the “Ready for Retest” status. Ordering by updated date ascending ensures a first-in, first-out testing queue, so fixes are not waiting unnecessarily long for verification. If this queue grows consistently, it indicates a testing capacity bottleneck - the team can fix issues faster than they can verify them, which means additional testing resources are needed.

### Keyboard-Related Accessibility Issues

```
project = "MYPROJECT" AND issuetype = "Accessibility Defect" AND (labels = "keyboard" OR "WCAG Success Criterion" = "2.1.1 Keyboard" OR "WCAG Success Criterion" = "2.1.2 No Keyboard Trap") ORDER BY "Accessibility Severity" ASC
```

This query demonstrates how to combine label-based and custom-field-based filtering to catch issues across different categorization approaches. Keyboard accessibility issues are among the most impactful defects because they block entire categories of users (motor-impaired users, screen reader users, power users who prefer keyboard navigation). The OR logic ensures you capture issues regardless of whether they were tagged via the WCAG Success Criterion custom field or a free-form label. Use this pattern to create similar queries for other common categories such as color contrast, focus management, or ARIA implementation.

### Issues Affecting Screen Reader Users

```
project = "MYPROJECT" AND issuetype = "Accessibility Defect" AND "Affected User Group" = "Blind/Screen Reader Users" AND status NOT IN (Closed, Verified) ORDER BY "Accessibility Severity" ASC
```

This query filters by the Affected User Group custom field to surface all open issues that impact blind and screen reader users specifically. This user-centric view is valuable during sprint planning because it allows teams to batch related work - fixing multiple screen reader issues in the same sprint is often more efficient than fixing one keyboard issue, one contrast issue, and one screen reader issue, because the developer stays within the same domain of ARIA attributes and semantic markup. Create equivalent saved filters for each user group to enable targeted remediation sprints.

## 6.3 Program Management Queries

### Defect Trend: Created vs. Resolved (Last 6 Months)

Use two separate queries and chart them together on a dashboard:

Created:

```
project = "MYPROJECT" AND issuetype = "Accessibility Defect" AND created >=
-180d
```

Resolved:

```
project = "MYPROJECT" AND issuetype = "Accessibility Defect" AND resolved
>= -180d
```

These paired queries produce the most important trend chart in your entire metrics program. In Jira, create two saved filters (one for each query) and add both to a “Created vs. Resolved Chart” gadget on your dashboard, configured to display as a line chart with monthly intervals. The intersection point where the resolved line crosses above the created line marks the moment your program shifted from backlog accumulation to backlog reduction. If the lines never cross, it means you are discovering issues faster than you are resolving them, which requires either increased remediation capacity or process changes to prevent new defects from being introduced.


*Figure 7: Created vs. Resolved trend chart showing the crossover point where backlog reduction begins*

### WCAG Success Criteria with Most Violations

```
project = "MYPROJECT" AND issuetype = "Accessibility Defect" AND "WCAG
Success Criterion" IS NOT EMPTY ORDER BY "WCAG Success Criterion" ASC
```

Display this as a pie chart or bar chart on your dashboard, grouped by the WCAG Success Criterion field. The resulting visualization reveals your organization’s systemic accessibility patterns. A typical distribution might show that SC 1.3.1 (Info and Relationships), SC 4.1.2 (Name, Role, Value), and SC 1.4.3 (Contrast Minimum) account for 60% or more of all defects. This concentration is actionable: it tells you exactly which topics your training program should prioritize, which component

library patterns need architectural review, and which automated testing rules will catch the highest volume of issues. Review this distribution quarterly to see whether targeted interventions are shifting the pattern.

### Issues Found by Discovery Method

```
project = "MYPROJECT" AND issuetype = "Accessibility Defect" AND "Discovery Method" IS NOT EMPTY ORDER BY "Discovery Method" ASC
```

Chart this as a bar graph grouped by the Discovery Method field to evaluate the relative contribution of each testing method to your defect pipeline. A healthy distribution shows issues coming from multiple sources - automated testing, manual expert audits, assistive technology testing, and CI/CD pipeline checks. If any single method accounts for more than 70% of discoveries, it suggests over-reliance on that method and underinvestment in others. If user-reported issues dominate, it means your proactive testing program is not catching what matters, and end users are bearing the burden of quality assurance. This metric directly supports the W3C AMM's proof points for the ICT Development Lifecycle dimension around testing methodology breadth.

### Accessibility Defects in Deferred Status

```
project = "MYPROJECT" AND issuetype = "Accessibility Defect" AND status = Deferred ORDER BY "Accessibility Severity" ASC, created ASC
```

This query surfaces every defect that has been intentionally set aside rather than resolved. Sorting by severity first, then by creation date, puts the riskiest and oldest deferred items at the top. Program managers should review this list monthly and challenge any Critical or High severity deferrals that have aged beyond 90 days. The Deferred status is a legitimate prioritization tool, but it can also mask an organization's unwillingness to invest in accessibility fixes. Track the size of the deferred backlog over time - a growing list is a leading indicator of accumulating accessibility debt and potential compliance exposure.

## 6.4 Compliance and Audit Queries

### All Open WCAG Level A Violations

```
project = "MYPROJECT" AND issuetype = "Accessibility Defect" AND "WCAG Conformance Level" = "Level A" AND status NOT IN (Closed, Verified) ORDER BY "Accessibility Severity" ASC
```

This query isolates the most fundamental accessibility failures - WCAG Level A criteria represent the baseline requirements without which content is fundamentally inaccessible to one or more groups of users. Any open Level A violation is, by definition, a barrier that prevents access entirely rather than merely degrading the experience. Compliance officers and legal teams need this specific view because conformance claims require meeting all applicable Level A criteria before Level AA can even be assessed. If this list is not empty, the organization cannot credibly claim any level of WCAG conformance for the affected content.

### Issues Resolved Since Last Audit

```
project = "MYPROJECT" AND issuetype = "Accessibility Defect" AND labels =
"a11y-audit-q4-2025" AND status IN (Closed, Verified)
```

This query tracks remediation progress against a specific audit's findings by using the audit-specific label (in this example, "a11y-audit-q4-2025"). Every accessibility audit should tag its resulting issues with a unique label that identifies the audit period. This enables before-and-after comparisons: how many of the Q4 2025 audit findings have been resolved? By comparing this count to the total issues created with that label, you get a percentage-complete metric that is easy to report to stakeholders. This approach also supports trend analysis across multiple audit cycles - if each successive audit finds fewer issues, it provides tangible evidence of maturity progression.

### Third-Party Component Issues by Vendor

```
project = "MYPROJECT" AND issuetype = "Accessibility Defect" AND labels =
"third-party" AND status NOT IN (Closed, Verified) ORDER BY component ASC,
"Accessibility Severity" ASC
```

This query separates accessibility defects that originate from third-party code - embedded widgets, SaaS platforms, component libraries, or vendor-provided integrations - from issues in your own codebase. Ordering by component groups the results by product area, making it clear which third-party dependencies are contributing the most accessibility barriers. This data is essential for procurement decisions: if a vendor's component consistently generates Critical accessibility defects, it provides evidence to renegotiate the contract, demand remediation, or evaluate alternative vendors. It also prevents your team from being held accountable for defects they cannot directly fix, enabling more accurate team-level metrics.

# 7. Building Accessibility Dashboards

Dashboards transform raw Jira data into visual intelligence that different stakeholders can consume and act on. The key is building the right dashboard for the right audience - executives need different views than development teams or accessibility program managers.

## 7.1 Executive Accessibility Dashboard

Audience: C-suite, VP-level leaders, accessibility executive sponsors. Purpose: Provide a high-level view of accessibility posture, risk, and investment impact.

Recommended gadgets:

1. Accessibility Backlog Overview (Pie Chart) – Shows the distribution of open defects by severity (Critical, High, Medium, Low). Uses the Accessibility Severity custom field.
2. Defect Trend: Created vs. Resolved (Two-Dimensional Chart) – Line chart showing defects created and resolved over the past 6–12 months. The lines should converge, with the resolved line exceeding the created line as the program matures.
3. Mean Time to Resolution Trend (Line Chart) – MTTR by quarter, segmented by severity. Demonstrates whether the organization is resolving issues faster over time.
4. Accessibility Release Gate Pass Rate (Bar Chart) – Percentage of releases passing the accessibility gate per quarter. An upward trend demonstrates improving development practices.
5. AMM Maturity Score Summary (Custom HTML or Table) – A heat-map style summary of the organization’s maturity level across all seven dimensions. Update quarterly.


*Figure 4: Executive Accessibility Dashboard showing KPIs, severity distribution, defect trends, release gate pass rates, and AMM maturity summary*

## 7.2 Development Team Dashboard

Audience: Engineering managers, development teams, QA leads. Purpose: Provide actionable detail on current work and immediate priorities.

Recommended gadgets:

1. Open Accessibility Defects by Component (Bar Chart) – Which product areas have the most work remaining?
2. My Team's Accessibility Issues (Filter Results) – Filtered list of open accessibility defects assigned to the team, sorted by severity.
3. Issues Ready for Retest (Filter Results) – A queue of issues awaiting accessibility verification.
4. WCAG Success Criteria Failure Distribution (Pie Chart) – Which WCAG criteria are violated most often? This helps teams identify where to focus training.

5. Regression Alert (Filter Results) – Previously resolved issues that have been reopened.

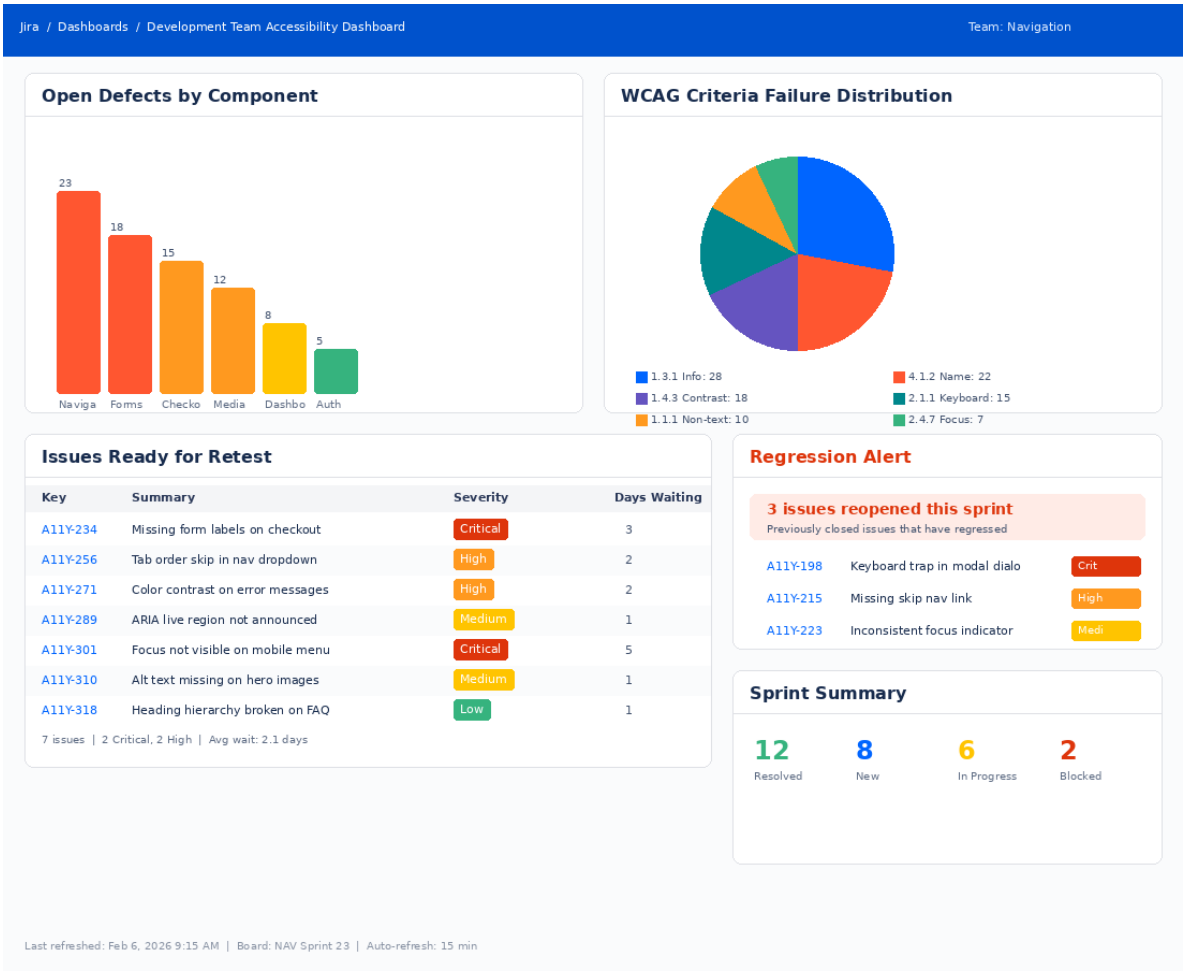


*Figure 8: Development Team Accessibility Dashboard showing component defects, WCAG failure distribution, retest queue, and regression alerts*

## 7.3 Accessibility Program Manager Dashboard

Audience: Accessibility leads, program managers, compliance officers. Purpose: Comprehensive program health monitoring with drill-down capability.

Recommended gadgets:

1. Full Backlog by Severity and Status (Two-Dimensional Chart) – Heat map of defects across severity and workflow status.
2. Discovery Method Distribution (Pie Chart) – Shows the relative contribution of each testing method. Validates testing investment.
3. Affected User Group Analysis (Bar Chart) – Which user populations are most impacted by current defects?
4. Remediation Velocity (Line Chart) – Number of defects closed per sprint, with a trailing average.

5. Training Completion Tracker (Table) – Shows training completion rates by role and department.
6. Vendor Accessibility Evaluation Status (Filter Results) – Open and completed vendor evaluations.
7. Deferred Issue Risk Monitor (Filter Results) – Deferred issues sorted by severity and age. Alerts when Critical issues remain deferred beyond a threshold.

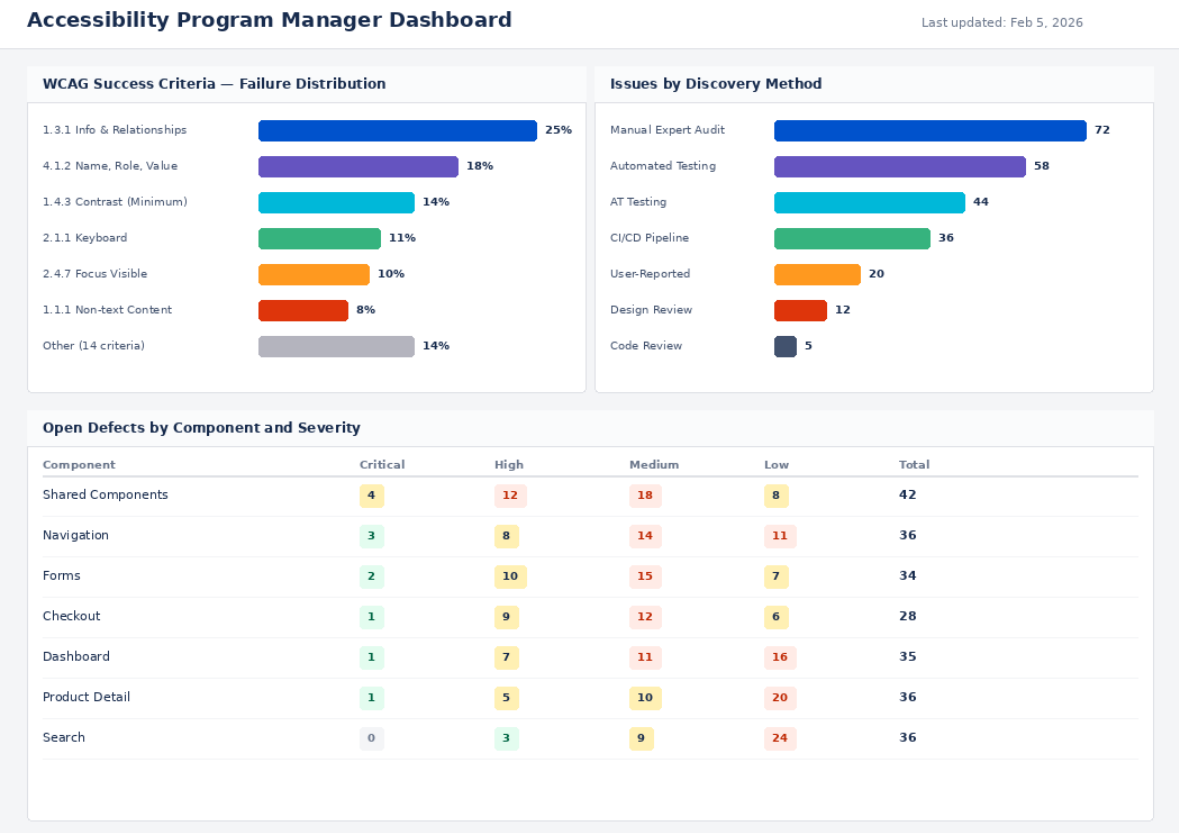


*Figure 5: Program Manager Dashboard showing WCAG failure distribution, discovery method analysis, and component-severity heatmap*

> *AFixt provides dashboard configuration services as part of its strategic consulting engagements. AFixt's team can configure dashboards that align with your specific organizational structure, product portfolio, and maturity goals - ensuring that the right data reaches the right stakeholders in the right format.*

# 8. Automated Metric Collection and Reporting

Manual metric collection does not scale. As your accessibility program matures, you will need automated systems that gather, aggregate, and distribute accessibility data without requiring manual intervention.

## 8.1 Jira Automation Rules

Jira's built-in automation engine can handle many routine accessibility metric collection tasks. Here are several high-value automation recipes:

### Auto-Label Accessibility Issues

Trigger: Issue created with issuetype = "Accessibility Defect." Action: Automatically add the "accessibility" label. This ensures that even if contributors forget to add the label manually, all accessibility defects are consistently tagged for filtering and reporting.

### Severity-Based SLA Assignment

Trigger: "Accessibility Severity" field is set to "Critical." Action: Set due date to 5 business days from creation. Assign to the accessibility lead for immediate triage. Send a Slack or email notification to the accessibility channel. This enforces the severity-based response times defined in your accessibility policy.

### Stale Issue Alert

Trigger: Scheduled check (weekly). Condition: Issue status has not changed in 14 days AND status is not Closed, Verified, or Deferred. Action: Add a comment tagging the assignee and accessibility lead. Move to a "Stale" label. This prevents issues from falling through the cracks and supports the MTTR metric by keeping issues moving through the workflow.

### Monthly Metrics Report

Trigger: Scheduled (first Monday of each month). Action: Execute a set of JQL queries, format results into a summary, and post to a Confluence page or distribute via email. Include total open defects by severity, defects created vs. resolved in the previous month, MTTR for the previous month, and top 5 WCAG success criteria violated.

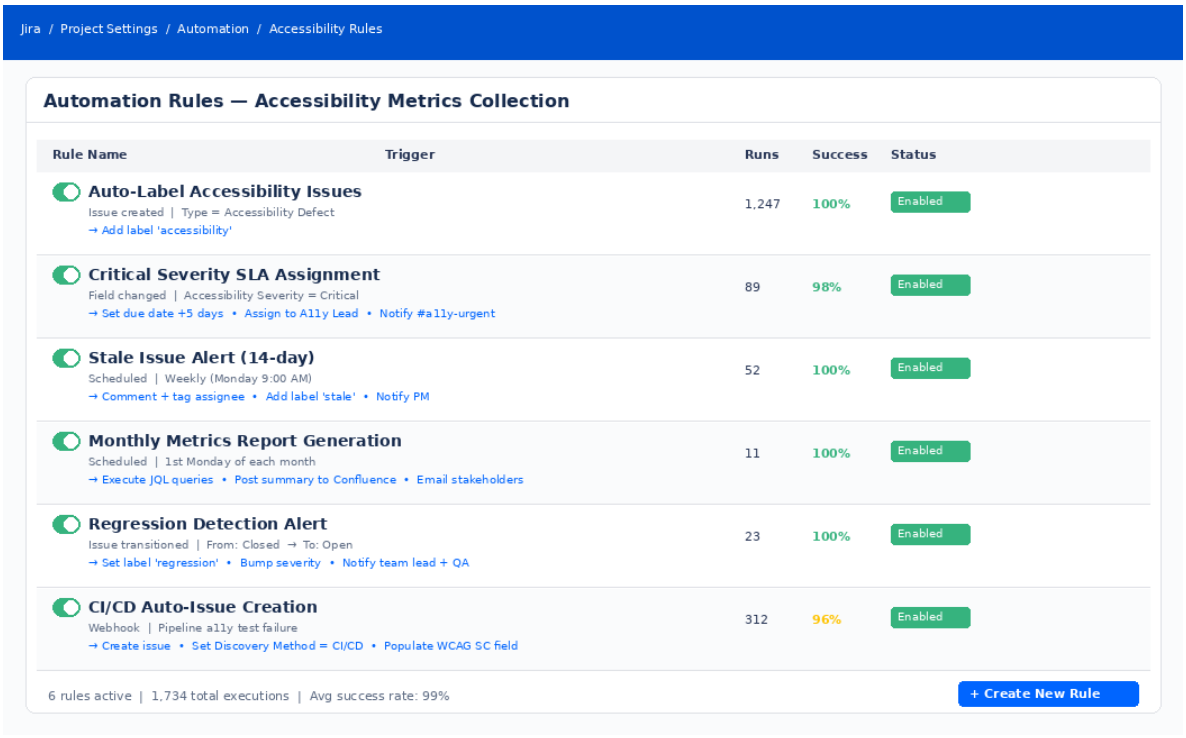


*Figure 9: Jira Automation Rules configured for accessibility metric collection, SLA enforcement, and stakeholder notifications*

## 8.2 CI/CD Pipeline Integration

Integrate automated accessibility testing into your build pipeline to generate issue tracking data automatically. When a CI/CD accessibility test fails, the pipeline can create a Jira issue with pre-populated fields:

- WCAG Success Criterion: Populated from the test rule that failed.
- Discovery Method: Set to “CI/CD Pipeline.”
- Accessibility Severity: Set based on the rule’s severity mapping.
- Component: Set based on the changed files or affected route.
- Labels: Include “automated,” “ci-detected,” and the build number.

This creates a continuous stream of issue data that feeds directly into your dashboard metrics without any manual intervention.

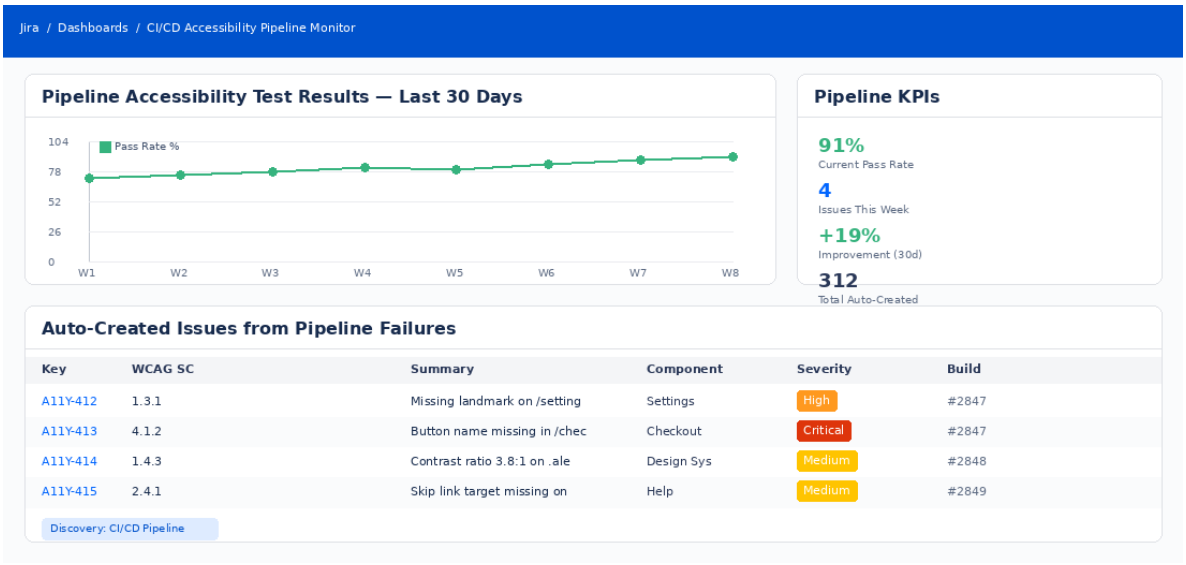


*Figure 10: CI/CD Pipeline Accessibility Monitor showing test pass rates, auto-created issues, and pipeline integration metrics*

## 8.3 Scheduled Reporting and Stakeholder Communication

Different stakeholders need different reporting cadences:

| Audience | Cadence | Content | Delivery Method |
|---|---|---|---|
| Executive Sponsor | Monthly | Backlog trend, MTTR, release gate pass rate, AMM maturity score. | Email summary with dashboard link |
| Development Teams | Per Sprint | Open defects by component, new defects this sprint, retest queue. | Sprint planning deck or Slack summary |
| Program Manager | Weekly | Full program metrics, stale issues, deferred risk, training updates. | Confluence page auto-updated |
| Legal/Compliance | Quarterly | Conformance posture, open WCAG A/AA violations, vendor compliance. | Formal report document |


*Figure 17: Auto-generated monthly accessibility metrics report in Confluence, showing KPIs, severity breakdown, WCAG violations, and AMM maturity status*

# 9. From Metrics to Maturity: Using Data to Drive Program Improvement

Metrics are only valuable if they drive action. This section describes how to use the data you collect to advance through the W3C AMM maturity levels.

## 9.1 Establishing Your Baseline

Before you can measure improvement, you need a clear picture of where you are today. Conduct a comprehensive assessment that includes a full accessibility audit of your digital properties, an inventory of existing accessibility processes and policies, a skills assessment of your teams, a review of procurement practices, and an evaluation of support capabilities.

Map each finding to the corresponding AMM dimension and maturity level. This baseline becomes the reference point against which all future metrics are compared.

> *AFixt's Discovery Project service provides exactly this kind of comprehensive baseline assessment. The engagement produces a detailed maturity map, a prioritized remediation roadmap, and the initial dataset needed to populate your Jira-based metrics system. This foundation ensures that your metrics program starts from an accurate, defensible baseline rather than incomplete assumptions.*

## 9.2 Setting Targets by Maturity Level

Use the baseline to set realistic, incremental targets for each dimension. Organizations at the Inactive level should focus on Launch-level targets: establishing an accessibility policy, completing initial training, setting up issue tracking with accessibility-specific fields, and conducting a first audit.

Organizations at the Launch level should target Integrate-level outcomes: embedding accessibility into the Definition of Done, maintaining a declining defect trend, achieving 80% training completion, and establishing accessibility as a release gate.

Organizations at the Integrate level should pursue Optimize-level goals: achieving sub-48-hour MTTR for Critical defects, demonstrating a regression rate below 5%, achieving 95% release gate pass rate, and publishing an accessibility metrics dashboard accessible to all stakeholders.

## 9.3 Quarterly Maturity Reviews

Schedule quarterly reviews of your AMM progress. In each review:

1. Review metrics trends across all seven dimensions.
2. Identify dimensions where metrics indicate stagnation or regression.
3. Update proof point documentation with new evidence.
4. Adjust targets and resource allocation based on findings.
5. Communicate progress to stakeholders using the reporting cadences defined in Section 8.3.

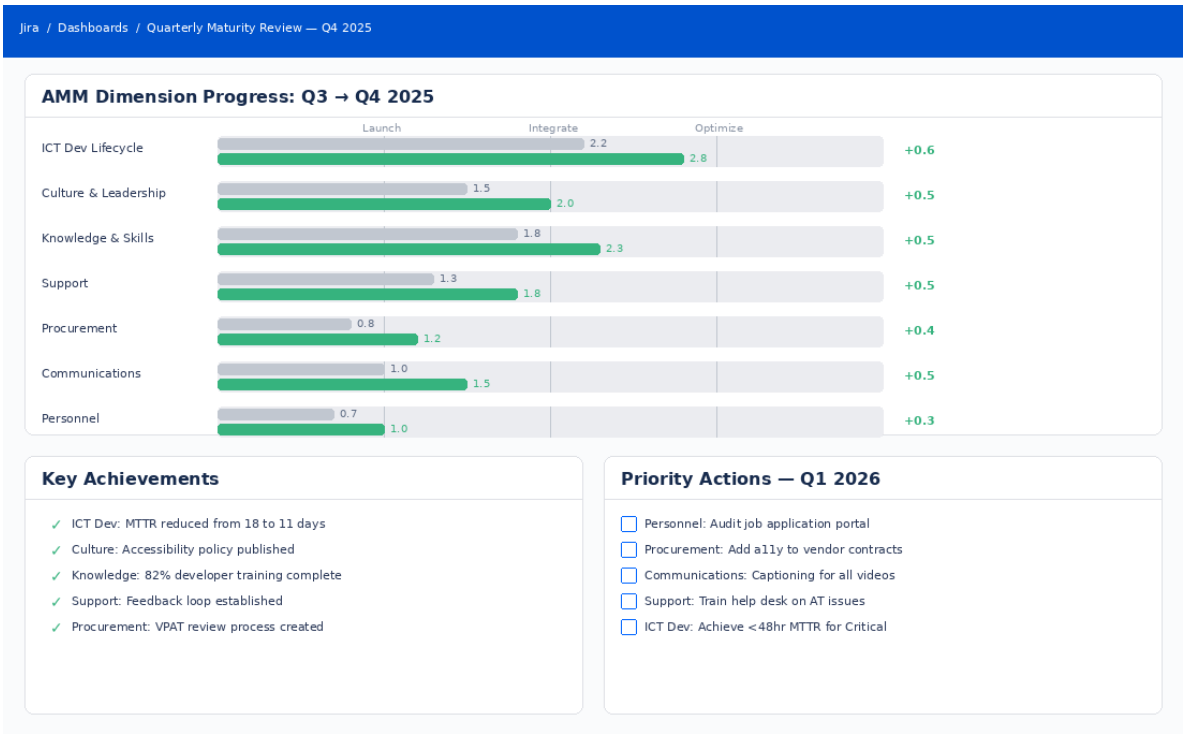


*Figure 11: Quarterly Maturity Review dashboard showing Q3-to-Q4 dimension progress, key achievements, and priority actions*

## 9.4 Common Pitfalls and How to Avoid Them

### Pitfall: Vanity Metrics

Tracking the total number of issues resolved is meaningless without context. An organization that resolves 500 low-severity issues while ignoring 50 Critical ones has not improved its accessibility posture. Always segment metrics by severity and impact.

### Pitfall: Metric Inflation Through Over-Reporting

If your CI/CD pipeline creates a new Jira issue for every automated test failure on every build, you will quickly drown in duplicate issues. Implement deduplication logic that links new findings to existing open issues rather than creating duplicates.

### Pitfall: Deferral as a Resolution Strategy

The Deferred status exists to handle genuine prioritization decisions, not to serve as a permanent parking lot for inconvenient issues. Monitor the Deferred backlog closely and set policies such as mandatory re-review every 90 days and a cap on the number of Critical issues that can be deferred simultaneously.

### Pitfall: Confusing Testing Coverage with Product Accessibility

Metrics that show 100% of pages have been tested do not mean 100% of pages are accessible. Testing coverage metrics must be paired with defect metrics to provide a complete picture. A page

that has been tested and found to have 15 Critical defects is not accessible, regardless of testing coverage statistics.

## 9.5 Scaling Your Metrics Program

As your program matures, extend your metrics into areas that may not have been priorities in earlier stages. Add user research metrics that track inclusion of participants with disabilities. Add procurement compliance metrics that assess vendor accessibility at contract renewal. Add communications metrics that track the accessibility of internal documents and presentations. And add personnel metrics that evaluate the accessibility of your own hiring pipeline.

Each extension maps to a specific AMM dimension and brings your organization closer to the Optimize level across the full model.

# 10. Conclusion

Accessibility maturity is not achieved through a single audit or a one-time remediation effort. It is built through sustained, data-informed investment across every dimension of the organization. The metrics framework presented in this paper provides the instrumentation needed to manage that investment effectively.

By configuring your Jira instance with accessibility-specific custom fields, building dashboards that surface the right data to the right audience, and aligning every metric with the W3C Accessibility Maturity Model's proof points, you create a system that turns raw issue data into strategic intelligence. You move from asking "Are we accessible?" to answering "How are we improving, where are we stalling, and what should we do next?"

The W3C AMM provides the framework. Your issue tracker provides the data. The JQL queries and dashboard blueprints in this paper provide the bridge between the two. What remains is the organizational commitment to act on what the data reveals.

*That commitment is easier to sustain when you have the right partner. AFixt brings over two decades of accessibility expertise to organizations at every stage of maturity. From initial assessments and comprehensive audits to hands-on remediation, role-based training, and ongoing strategic consulting, AFixt's services are designed to complement and accelerate the metrics-driven approach described in this paper.*

**To learn how AFixt can help your organization establish, measure, and advance its accessibility program, visit afixt.com or contact us at karl.groves@afixt.com.**