

axe DevTools Mobile Release Notes

November 18, 2021

Component Versions

- Attest-iOS v2.10.0
- axeDevToolsXCUI v1.0.0
- axe-devtools-android v 1.1.0
- axe DevTools Mobile desktop app v0.24.0

New Features

axe DevTools Mobile for iOS

SwiftUI Support

This release of axe DevTools Mobile includes the ability to test the accessibility of iOS applications built with SwiftUI. There is an additional framework available called axeDevToolsXCUI. This framework works fundamentally different from the existing framework. It does not use the Floating Action Button (FAB) and can only be used via API in UI testing such as XCUITest and Appium. See the [getting started guide](#) for instructions on how to use the new framework.

Rules included in axeDevToolsXCUI framework:

- Color Contrast
- ActiveControlName
- CollidingControls
- CollidingViews
- Touch Target Size
- Conflicting Traits
- Focusable Text
- ImageView Name
- In ScrollView
- Label At Front
- Label In Name

Note:

The following UIKit rules can no longer be violated within SwiftUI and as such are not supported in the SwiftUI framework:

- A11yElementFocusBox
- Nested Active Controls
- Nested Elements Name
- Supports Dynamic Type

This new framework will also work with apps built with UIKit, however we will continue to maintain and improve the existing framework (Attest-iOS).

Ignore Rules

This release includes the ability to ignore a specific rule entirely, ignore rules by class name, and ignore rules by accessibility identifier. While we feel that all rules are important, these customizations will let you ignore views that have yet to be fixed and focus more on unknown or new issues. For details on how to use this new API, see the following documentation:

- [UIKit framework \(Attest-iOS\)](#)
- [SwiftUI framework \(axeDevToolsXCUI\)](#)

axe DevTools Mobile for Android Library

New Rules

This release contains 3 new rules for testing the accessibility of Android applications:

- HiddenActiveViewFocus: Views that users can interact with must not be covered by other views.
- HiddenInformativeViewFocus: Informative views must not be covered by other views.
- InactiveAccessibleView: Views that have a name and value assigned, should not be marked unimportant for accessibility.

For more information on these new rules, see the [remediation help documentation](#).

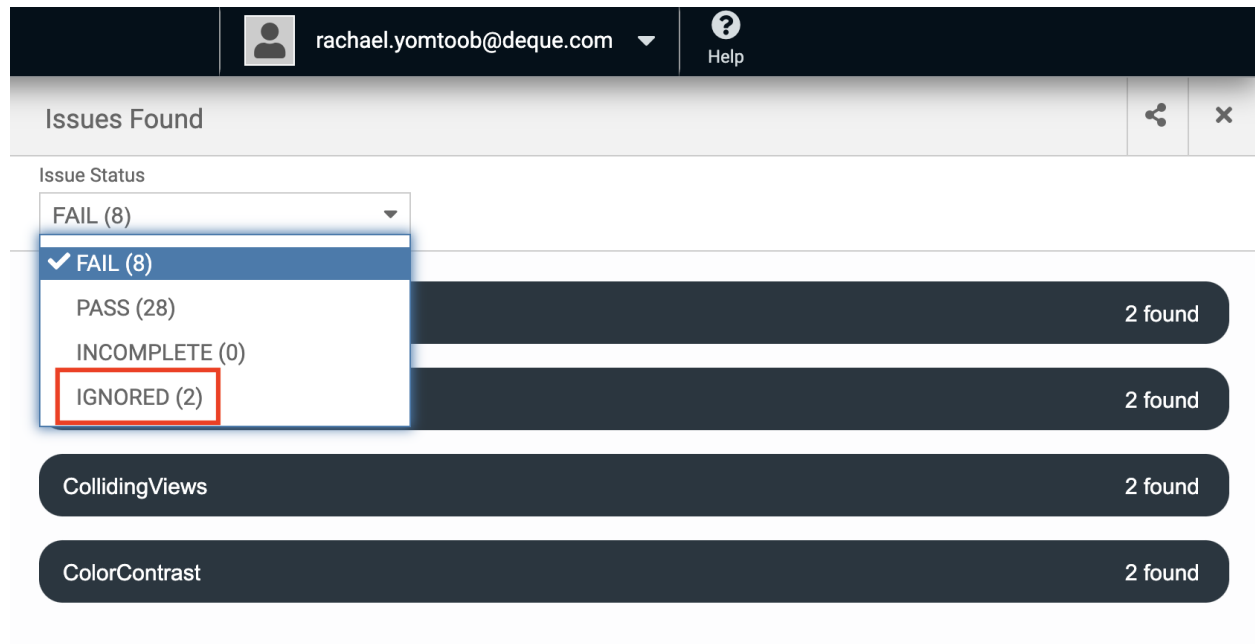
Ignore Rules

This release includes the ability to ignore rules by view id, ignore rules by class name, and ignore all rules by class name or view id. While we feel that all rules are important, these customizations will let you ignore views that have yet to be fixed and focus more on unknown or new issues. For details on how to use this new API, see the [axe DevTools for Android documentation](#).

axe DevTools Mobile Dashboard and Desktop App

Ignore Rules

Results that have been ignored via the new APIs in the iOS frameworks and Android Library will be marked with the status “IGNORED”. These results can still be viewed in the dashboard via the “Issue Status” dropdown on the Issues Panel for a scan.



The screenshot shows the user interface of the axe DevTools mobile dashboard. At the top, there is a dark navigation bar with a user profile icon, the email address 'rachael.yomtoob@deque.com', and a 'Help' button. Below this is a header for 'Issues Found' with share and close icons. The main content area features an 'Issue Status' dropdown menu that is currently open, displaying the following options: 'FAIL (8)' (selected), 'PASS (28)', 'INCOMPLETE (0)', and 'IGNORED (2)'. The 'IGNORED (2)' option is highlighted with a red rectangular box. Below the dropdown, there is a list of issues, each represented by a dark horizontal bar with a white label on the left and a white count on the right. The visible issues are 'CollidingViews' and 'ColorContrast', both with a count of '2 found'.

Bug Fixes

axe DevTools Mobile for iOS

- Fixed false positives in the Supports dynamic type for UIButtons in iOS 15 [CSD-17278]

axe DevTools Mobile for Android Library

- Fixed false negatives in the Color Contrast rule for text field hint text

axe DevTools Mobile Dashboard and Desktop App

- Display “No Title Available” instead of “incomplete data” as the scan name if a screen title can not be automatically detected
- Allow the path to where the desktop app is installed to include spaces [CSD-16852]
- The desktop app for Windows is now code signed, so it is less likely to throw a security warning on installation [CSD-16852]

Known Issues

axe DevTools Mobile Desktop App

- Fails to connect to database if using Java 14+, you must continue to use an earlier version of java for the Desktop app

axe DevTools Mobile for iOS

Attest-iOS (UIKit framework)

- Banner items report requiring InScrollView.
- iOS apps built with SwiftUI have reduced issue detection by the UIKit framework (Attest-iOS). We encourage only using axeDevToolsXCUI to test SwiftUI views.
- The Custom Rule API is not exposed to Objective C.
- On machines running macOS 11.2.3, the Floating Action Button can become stuck while in landscape mode on simulators running versions of iOS earlier than 13 (We have reported this bug to Apple).
- You may experience an issue with the FAB's defaultAttachmentStrategy on iOS 12 simulator or device. If this occurs, please use FABWindowStrategy.attach
- Rules do not run on WebViews or rendered PDFs. We suggest using axe DevTools for HTML in those instances.
- axe DevTools Mobile only runs on native views, hence it is not compatible with hybrid platforms such as Flutter and Cordova.
- The Ignore Rules feature does not work for the Color Contrast rule because this rule is run on the server as opposed to in the framework itself. This will be resolved in a future release.

axeDevToolsXCUI (SwiftUI framework)

- The framework may get stuck scanning a Swift UI Picker on any simulator or device running anything earlier than iOS 15.0.
- A11yElementFocusBox may throw false positives with views partially off-screen.
- The "Visible to User" filter may not work on views behind custom-made modals.
- Landscape orientation is not formally supported at this time. Banner items report requiring InScrollView.
- Rules do not run on WebViews or rendered PDFs. We suggest using axe DevTools for HTML in those instances.
- axe DevTools Mobile only runs on native views, hence it is not compatible with hybrid platforms such as Flutter and Cordova.

- The Ignore Rules feature does not work for the Color Contrast rule because this rule is run on the server as opposed to in the framework itself. This will be resolved in a future release.

axe DevTools Mobile for Android

- Rules will not be run on any Compose views, this will be addressed in a future release.
- Rules do not run on WebViews or rendered PDFs. We suggest using axe DevTools for HTML in those instances.
- axe DevTools Mobile only runs on native views, hence it is not compatible with hybrid platforms such as Flutter and Cordova.
- The Ignore Rules feature does not work for the Color Contrast rule because this rule gets run on the server for all scans that don't have Color Contrast results. This will be resolved in a future release.