

---

# **DHS Section 508 Compliance Test Process for Android Mobile Applications**



September 29, 2017

---

## CONTENTS

<b>Contents</b> .....	<b>2</b>
<b>About this document</b> .....	<b>4</b>
DHS Test Process for Android Mobile Applications.....	4
How this document is structured.....	4
<b>Section 1: Introduction and Rationale for Tests</b> .....	<b>5</b>
Applying the Section 508 Standards to the Android Mobile Applications Test Process .....	5
The Rationale for each test .....	5
<b>Section 2: Test Environment</b> .....	<b>10</b>
Devices:.....	10
Additional Equipment:.....	10
Test Tools:.....	10
Select the correct test process for testing .....	15
<b>Section 3: DHS Section 508 Compliance Tests for Android Mobile Applications</b> .....	<b>16</b>
Structure of this section .....	16
1. Keyboard and Focus .....	17
2. Screen Reader Test.....	25
3. Video, Audio, and Multimedia .....	32
4. Color and contrast.....	38
5. Flashing .....	40
6. Time Outs .....	41
7. Built-in Accessibility Features .....	43
8. Accessible Alternative .....	47
<b>Elements Table</b> .....	<b>49</b>
<b>Android Keystrokes And Gestures</b> .....	<b>54</b>
How to take a Screenshot .....	54
Android Keystrokes without TalkBack enabled .....	54
Android Keystrokes with TalkBack enabled .....	55
TalkBack Gestures .....	56
Magnification Gestures.....	58
<b>Document Content Change Log</b> .....	<b>59</b>



## ABOUT THIS DOCUMENT

### DHS TEST PROCESS FOR ANDROID MOBILE APPLICATIONS

The Department of Homeland Security (DHS) Office of Accessible Systems and Technology (OAST) has a mission to provide strategic direction, technical support, and training to ensure agency employees and customers with disabilities have equal access to information and data.

Section 508 of the Rehabilitation Act of 1973, as amended (29 U.S.C. 794d), requires all federal departments and agencies to ensure that their electronic and information technology (EIT) is accessible to people with disabilities. This DHS Section 508 Compliance Test Process for Android Mobile Applications has been produced in support of this mission.

### HOW THIS DOCUMENT IS STRUCTURED

**Section 1** describes the **rationale** for testing mobile software apps (including those with embedded web views) using the process defined in this document. The rationale behind each individual test in this document is also provided.

**Section 2** provides details of the **test environment for this process** (device, additional equipment, test tools), including device recommendations, installation and settings guidance.

**Section 3** is the **Android test process**. Each test includes step-by-step instructions on how and what to test, as well as instructions on which standards to mark as compliant or not compliant. Please be sure you are following the test process for the correct App Operating System (OS).

## SECTION 1: INTRODUCTION AND RATIONALE FOR TESTS

### APPLYING THE SECTION 508 STANDARDS TO THE ANDROID MOBILE APPLICATIONS TEST PROCESS

The Section 508 technical standards include one section aimed at browser-based information (Web), and another section aimed at native applications and operating systems (Software). The standards do not specifically call out mobile apps or mobile web content. DHS has determined that mobile applications require a testing process distinct from the desktop/laptop application test process due in part to the lack of tools to evaluate accessibility on mobile platforms. Mobile platforms also have greater limitations on accessibility applications programming interfaces (APIs) and assistive technologies that are available for a given platform.

The Section 508 Standards also contain a section on Functional Performance Criteria (FPC). The FPC require that "at least one mode of operation and information retrieval that does not require user [vision, hearing, speech or fine motor skills] shall be provided, or support for assistive technology used" by people who are blind, low vision, deaf, hard of hearing, speech disabled, or have limited physical capabilities.

There are two broad categories of mobile content: native mobile apps that are designed to run on a mobile device and mobile web content that is designed to be viewed in a mobile browser. For testing purposes, mobile apps including those with embedded web views – sometimes referred to as “hybrid” apps – should use this mobile testing process. Hybrid apps are covered under this test process primarily because it may not be possible to gain access to the web content in these web views via methods used for testing non-mobile web and software content.

Web content designed for mobile consumption should continue to use the Software and Web application test process used for desktop/laptop content, with a modified viewport size.

The DHS mobile testing process begins with evaluating elements that are most commonly found in mobile applications and therefore most likely to have applicable accessibility requirements. There are 8 main steps in the test process.

### THE RATIONALE FOR EACH TEST

Each step of the test process includes only the information that will need to be referenced frequently, namely the directions on how to test, and how to interpret the test results. To simplify the organization of this document, the rationale for each individual test has been separated out, and is presented below.

#### 1. KEYBOARD AND FOCUS TEST

---

##### KEYBOARD

Interactive elements of interfaces include any elements that a user is expected to use, modify, or edit. Examples include navigation controls (links, buttons etc.), and editable content (selectable text, data input etc.).

It must also be possible for users to determine what the interactive elements are, and how to use them. This requires that the visual label / instructions are programmatically associated with controls; otherwise non-visual users will not be able to tell which label relates to which control / form-element.

The keyboard-only test determines whether it is possible to control the interface without the visual and/or physical capabilities necessary to use touch gestures without assistive technology or a pointing device.

Wherever users are expected to interact with components, it must be possible for users to access those components or perform those functions using only the keyboard because (i) using a pointing device or touch gestures without assistive technology is not possible when the user has no sight, and (ii) using a pointing device or touch gesture is not possible when the user does not have the physical capability / dexterity to effectively control a pointing device or use touch gestures without assistive technology.

Keyboard access is defined as use with a physical keyboard that is attached to the mobile device, either separately through a protocol such as Bluetooth, or integrated.

---

### FOCUS

Ideally, interfaces use standard keyboard commands (TAB, Space Bar, Enter, Escape, etc.), making their use easy and intuitive. On occasion, an interface may be designed to expand on the basic set of standard keyboard commands, and/or remap standard keys. In both of these cases, users must learn the non-standard keys. In order to be aware of non-standard key commands, users must be notified of their existence and correct use through the interface, application help, and/or documentation.

When controlling the interface with the keyboard only, if there is no visual differentiation between the current focused item and the rest of the interface / content, then it is not possible to tell where in the interface you are. Therefore, a visual indication of focus is necessary.

A logical order and groupings of interface components is normally a given in the design of software applications and embedded web content. Groupings and order are usually visually apparent. Logical arrangements are used to aid visual appeal and improve usability. However, when the focus/TAB order does not follow the logical order, users can become confused, make errors, and may not understand the contextual meaning of components. This is especially true for people who have no vision, or who have low vision, and are relying on AT.

Some components in embedded web content and software screens are intentionally hidden to reduce visual clutter. Other components only appear as part of a procedure such as an error notification. Content with such interface components may be revealed in an inaccessible manner by requiring user vision and/or requiring the use of a mouse. Keyboard and touch screen users need to be able to access the information and controls that are revealed, and users without vision, or with low vision, need to know that new content has appeared.

## 2. SCREEN READER TEST

Assistive technology utilizes accessibility properties of elements and provides them to users through various modes to provide access to the application. Accessibility properties are provided through the accessibility APIs available in mobile operating systems, HTML code, and ARIA properties or other protocols. Without these labels and accessibility properties, the assistive technology may not provide correct information to the user.

In order to correctly and accurately complete a form, it is necessary to follow instructions, directions and cues, as well as enter information in the correct places. A given form component may be the subject of instructions that are not positioned next to the component (e.g., at the top of a form, the instruction is "If you are the home owner, complete parts a, b, and f"). In such cases, form designers will use visual layout and flow to direct the user. However, users without vision, or with low vision, may not have access to the visual cues, and hence will be unable to easily find the related instructions for the current form component. For this reason, it is necessary to programmatically associate all relevant instructions, directions and cues with their respective components/controls.

All interface elements including images, charts, and tables that provide meaningful information must be conveyed correctly to assistive technology through their accessibility properties (name, role, state, and value) in a consistent manner for users without vision or with low vision. For the purpose of testing, the name, role, and state expected outcomes are listed in the Elements Table. The tester is testing to see if *sufficient* characteristics of *all* elements are announced.

To aid navigation with screen reading AT software, users can bring up a list of navigation controls on their screens. Users can read through content and decide which of the links in the content they wish to follow (i.e., they do not have to navigate back to the link itself).

Links must have a unique and descriptive name. Say each item for sale has a 'click here' link next to it, and the user calls up the list of controls. The list will have multiple 'click here' links that are not distinguishable. Another common problem occurs when the links only contain URLs, and the purpose of each link may not be apparent. It is therefore required to use meaningful and unique names for links and user controls, to aid navigation using assistive technology for users without vision or with low vision.

For screen reader users, screen orientation is important in order to ensure proper gestures are used. Some apps will allow rotation as the device moves; others are static. Orientation must be announced by the mobile screen reader if the app supports device rotation.

### 3. VIDEO, AUDIO, AND MULTIMEDIA

This section addresses audio files, animations, video files, and multimedia. Screen reader software cannot interpret images, animation, video, or multimedia. Screen readers will, however, read text that has been associated with interface elements. The interpretation (meaning) of an interface element must therefore be conveyed textually in the interface programming for assistive technology for users without vision or with low vision.

Animation includes sequences of overlaid images, dynamic changes of state such as a moving speed dial, a chart illustrating dynamic flow changes from one state to another, etc. Video-only files include animations, screen, video captures etc. The visual information provided through animation and video-only must be provided through alternative means for assistive technology for users without vision or with low vision.

Audio-only files include speeches, sound-bites, ambient (background) sounds, etc. Equivalent text descriptions must be provided for users with no hearing or who are hard of hearing.

Synchronized media is a presentation consisting of time-synchronized video and audio. Synchronized media includes public information films, Webcasts, press conferences, and online training presentations.

Some users will not be able to hear the content. Therefore there needs to be another mode to provide the audio information, such as captions (text showing what is being said, and other relevant sounds). Captions need to be available, but do not necessarily need to be turned on by default. For example, users who need captions can switch them on with a control (usually a 'CC' button for Closed Captions). If there is no means of switching modes, then the default mode must be accessible (i.e., Open Captions).

Because captions must be time-synchronized, separate transcripts will not meet this requirement on their own.

Some users will not be able to see the content. Therefore there needs to be another mode to provide descriptions of the visual information. In synchronized media, this usually means additional narration inserted during breaks in the dialog, describing visual events and cues.

Audio descriptions need to be available, but are not required to be turned on by default. For example, users who need descriptions can switch them on with a control. If there is no means of switching modes, then the audio descriptions must be enabled by default.

The alternative presentation of information must allow understanding of the relevant information. For example, descriptions might include the looks on people's faces, people handing items to each other, or who has entered the room.

### 4. COLOR AND CONTRAST

The use of color to convey meaningful information must be provided through alternative means for users who cannot distinguish colors. Insufficient contrast may make it difficult for some users to see and use the content.

Color dependence is using color as the sole method to convey information. For example, a single unlabeled indicator that is green for 'on', orange for 'standby', and red for 'off' is color dependent.

When color is the only means to convey information, people who are color blind, and people who cannot see, do not have access to the same information that others have. The status or function that is being conveyed by color also needs to be available in a textual format that can be viewed by all, and can be read by screen reader software.

This requirement does not mean that color cannot be used; it means that color cannot be the *only* means of conveying the information.

The visual difference between the background behind text, and the text itself, may be perceivable by a given designer. However, beyond color choice which is under control of the designer, many factors beyond the designer's control affect peoples' ability to discern between colors/shades, including age (contrast sensitivity reduces with age), screen brightness, ambient light, color blindness and some types of low vision. The use of color/shade choices that do not contrast well with each other may be deliberate (i.e., artistic preference), or may be the result of programmatic features (e.g., a button's text is black on white, but the text turns yellow in a certain mode, and the background remains white).

In general, the higher the level of contrast used, the more people will be able to see and use the content.

If the color contrast cannot be tested, this will be noted as a failure.

### 5. FLASHING

The term 'flashing' encompasses interface elements that blink, flicker repetitively, or elements that scroll (e.g., marquee text).

An element that flickers or blinks in the visual field can cause adverse reactions/seizures in people who have photosensitive epilepsy. Section 508 does not permit flashing in the frequency range between 2Hz to 55Hz (from twice per second to 55 times per second). This frequency has been updated in WCAG 2.0, which is followed by DHS, to not permit flickering above 3 hertz (flickering above 55hz is not visually apparent).

Scrolling ('marquee') text should be avoided where possible, because the scrolling effect causes a form of flickering that can be in the prohibited frequency range. Even though this may be imperceptible for many viewers, it can have the same flickering effect for some. Scrolling text should also be tested for Screen reader access.

Due to the current lack of source code and testing tool limitations, all flashing will fail.

## 6. TIME OUTS

Messages and/or instructions to the user requesting their response within a given time are typically associated with sites that require a secure login. This includes both server time outs and client side security time outs.

People who use AT such as screen reader software or voice input software may require more time than other users to assimilate the information and execute the controls on a Web page or software application. Because AT users may need more time, applications that have a time out must provide (a) prior notification/warning that a time out is about to occur, and (b) a means for the user to request more time.

## 7. BUILT-IN ACCESSIBILITY FEATURES

It is possible to write software that controls various aspects of the OS. This may inadvertently cause an OS accessibility feature to deactivate. For example, TalkBack is a screen reader built in to Android which allows those who are blind or low vision to access the operating system and applications through the use of speech output and braille. TalkBack contains its own set of gestures that are used to interact with elements on the screen. Developers should ensure that they do not disable or override the gestures that are used by TalkBack users, as this will impact their ability to interact with the mobile application.

Similarly, applications must allow assistive technology to be run concurrently with the app. For example, if a low vision user enables the Android Magnification assistive technology feature to get a closer look at a control within an app, the app should not terminate when magnification is activated.

The accessibility features of Android (the platform for which the tests are written) contain the following user-configurable accessibility features that should not be disabled or disrupted by the software application:

- Android: TalkBack, Magnification, Closed Captioning, Large Font Size, Large Display Size, microphone text input

## 8. ACCESSIBLE ALTERNATIVE

An 'Alternate' is an accessible version containing the same information as the primary application. Alternates will usually contain text in place of the inaccessible content from the primary application. For example, a complex organizational chart may be written in prose. The text must be equivalent, and it must be kept up-to-date.

An 'Alternate' should only be provided for accessibility when the primary application cannot be made accessible. The accessible version must contain the same information as the primary application.

The information should be 'equivalent', but by definition this is not going to be 'exactly the same'. The main points, themes, concepts etc. that the authors are trying to get across in the primary content should also come across in the alternate. For example, if a complex chart on the primary page shows a year with a small increases in earnings in Q2 and a large decrease in Q2, and the text discusses why these trends seem to be occurring, the Alternate should convey the trends, and the high and low data points of interest. An Alternate that just provides all the data points in linear form, with no highlighting of the trends under consideration, would not be considered equivalent.

Alternative versions for accessibility are only permitted when the primary application cannot be made compliant. Common examples where alternative versions are usually permitted include maps and directions, and very complex diagrams and charts.

## SECTION 2: TEST ENVIRONMENT

### DEVICES:

Mobile application testing must be performed on mobile devices such as phones or tablets. The DHS Android Mobile Test Process's test environment typically utilizes the current Android Operating Systems, which as of writing this is Android 7.x **Note:** When new OS updates are released, there could be bugs that impact AT use; an older OS may be recommended for testing purposes until an update fixing AT issues is released.

The latest version of this platform may not be available for all Android devices. Note: the assistive technology on the Android is not updated with the platform. For this reason, it is recommended that the latest version of the Talkback screen reader, which as of this writing is version 5.2.1, is used for Android Screen Reader testing.

Some Android products contain additional user interfaces that sit on top of the Android platform. Our recommendation is that testing devices do not contain these additional user interfaces and testers should use the standard Android platform provided on Google's product line.

A general note on mobile web browsers: The Chrome web view is used in applications with embedded web views on Android. While the Chrome browser in Android is used for embedded web views, Mozilla Firefox provides additional accessibility features. When possible, it is suggested using Mozilla Firefox for more accurate test results.

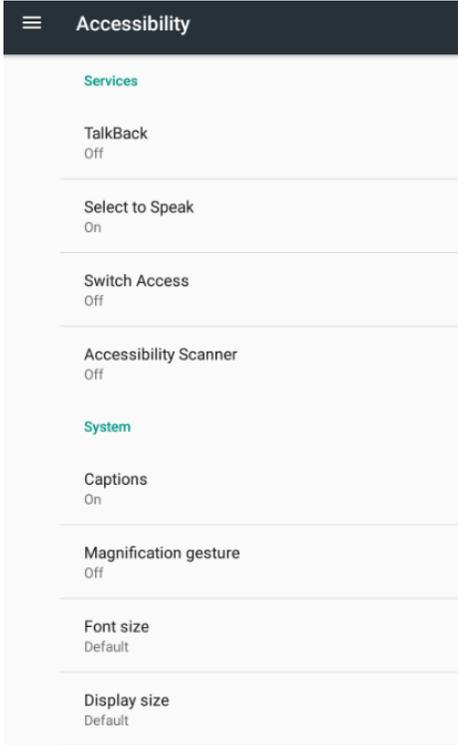
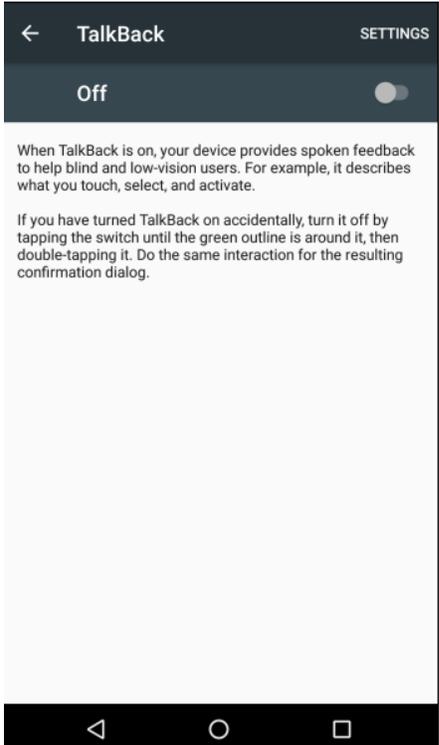
### ADDITIONAL EQUIPMENT:

- Desktop/laptop to record the test results and for use with the Colour Contrast Analyzer
- A Bluetooth keyboard
- Sound, Wi-Fi, and Bluetooth capabilities must be available on the mobile device
- Mobile device must have the OS version listed above and tools described in the following section installed

### TEST TOOLS:

The tools used in the DHS Mobile Android Test Process have been chosen based on the availability of accessibility inspection tools on mobile platforms. This test process uses platform specific assistive technology to inspect accessibility properties in an evidenced based manner. Use of these tools does not require the tester to view source code or have knowledge of programming languages.

TALKBACK

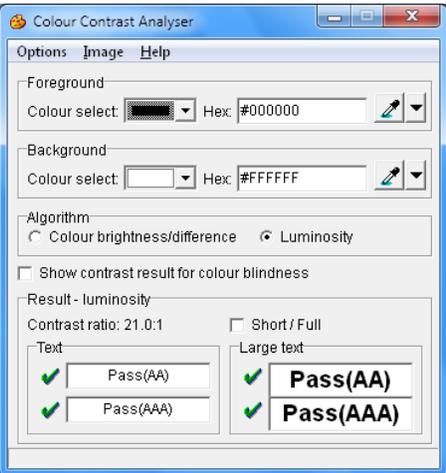
<p><b>Official Name</b></p>	<p><b>TalkBack</b></p>
<p><b>Also Known As</b></p>	<p>n/a</p>
<p><b>URL of Product</b></p>	<p>https://play.google.com/store/apps/details?id=com.google.android.marvin.talkback&amp;hl=en</p>
<p><b>Purpose for Section 508 Testing</b></p>	<p>Reveal the name, role, value, and state of a user interface control.</p>
<p><b>Developed By</b></p>	<p>Google</p>
<p><b>Current Version</b></p>	<p>Version 5.2.1</p>
<p><b>Installation Advice</b></p>	<p>To enable TalkBack Choose Settings &gt; Accessibility &gt; TalkBack</p>
<p><b>Settings</b></p>	<div style="display: flex; justify-content: space-around;">   </div> <p>To enable TalkBack, choose Settings &gt; Accessibility &gt; TalkBack: On</p> <p>Settings &gt; Accessibility &gt; TalkBack &gt; Settings</p> <ul style="list-style-type: none"> <li>• <b>Explore by touch:</b> On</li> <li>• <b>TalkBack suspend and resume shortcut:</b> On.</li> </ul>

	<p><u>Under Verbosity:</u></p> <ul style="list-style-type: none"> <li>• <b>Use pitch settings:</b> On</li> <li>• <b>Speak usage hints:</b> On</li> <li>• <b>Speak list and grid information:</b> On</li> <li>• <b>Speak element type:</b> On</li> <li>• <b>Item description order:</b> state, name, type</li> </ul> <p><u>Under text to Speech Settings:</u></p> <ul style="list-style-type: none"> <li>• <b>Preferred engine:</b> “Google Text-to-speech Engine”</li> <li>• <b>Speech rate:</b> set to a comfortable setting</li> </ul> <p><u>Under Developer Settings:</u></p> <ul style="list-style-type: none"> <li>• <b>Display Speech Output:</b> On This will display the text that is spoken by TalkBack at the bottom center of the screen.</li> </ul> <p>Settings &gt; Accessibility</p> <ul style="list-style-type: none"> <li>• <b>Select to Speak:</b> On</li> <li>• <b>Captions:</b> On <ul style="list-style-type: none"> <li>○ Choose a Caption <b>style</b> that is easy for you to view</li> </ul> </li> <li>• <b>Font Size:</b> set to largest setting (slide all the way to the right)</li> <li>• <b>Display Size:</b> set to largest setting (slide all the way to the right)</li> <li>• <b>Auto-Rotate screen:</b> On</li> </ul> <p>Settings&gt; Languages and Input</p> <ul style="list-style-type: none"> <li>• <b>Virtual keyboard</b> &gt; Gboard and Google voice typing</li> </ul> <p>Settings &gt; Bluetooth</p> <ul style="list-style-type: none"> <li>• <b>Bluetooth:</b> On</li> <li>• Pair your Bluetooth keyboard: <ul style="list-style-type: none"> <li>○ Choose <b>Search for devices</b></li> <li>○ When the device is found select the device from the list</li> <li>○ Enter the number display on the Android display into the Bluetooth keyboard and then press “Enter”</li> <li>○ The device should now be paired</li> </ul> </li> </ul> <p>From any screen: To temporarily pause TalkBack long press the volume Up and Down Arrow keys</p>
--	---

TalkBack is a screen reader. While this tool is assistive technology used by people who are blind or visually impaired for access, it can also provide a consistent method of identifying programmatic accessibility information and can be a very useful tool. In order to provide access to touchscreen devices, mobile screen readers provide an alternative set of gestures. For example, touching an item no longer activates the item when the screen reader is running. Instead it focuses the item under the finger and announces it to the user. The user must double tap to activate the item – thus, a double tap sends a normal single tap gesture. There are a number of gestures that you will need to learn to use the assistive technology. These gestures and keystrokes for using the screen reader with an external keyboard are provided; please review these [commands](#) and become familiar with how TalkBack works before performing testing.

TalkBack provides a mode that displays text spoken by TalkBack. When this mode is enabled the text is displayed at the bottom center of the mobile device screen. Testers who have difficulty understanding synthetic speech or testers who are deaf or hard of hearing can use this option. This also will provide a textual representation of speech output for reporting purposes.

*COLOR CONTRAST ANALYSER*

<b>Official name</b>	<b>Colour Contrast Analyzer</b>
<b>Also Known As</b>	Contrast Analyzer
<b>URL / product page.</b>	<a href="http://www.paciellogroup.com/resources/contrastAnalyser">http://www.paciellogroup.com/resources/contrastAnalyser</a> ; <a href="http://www.wat-c.org">http://www.wat-c.org</a> (Included with the Web Accessibility Toolbar (WAT))
<b>Purpose for 508 testing</b>	Allows the user to use an eye dropper to check the contrast of background and foreground text of an image taken from a mobile device. This application runs on the Windows platform.
<b>Developed by / Owned by</b>	The Paciello Group, WAT-C.org
<b>Current version at the time of writing</b>	2.4, English
<b>Installation advice</b>	<p>Download application. Run executable from PC. From the Web Accessibility Toolbar choose Colour &gt; Colour Contrast Analyzer.</p> <p>If using the standalone version (not within WAT), Admin rights are not required to execute this tool of version 2.2 (version 2.4 may require admin rights).</p>
<b>Enabling the Colour Contrast Analyzer</b>	Run the executable. From the main screen's <b>Algorithm</b> group, select the <b>Luminosity</b> radio button. Use the eye dropper tool for the foreground to obtain the foreground color and then eye dropper tool from the background to obtain the background color. Verify the results in the <b>Result – Luminosity</b> section.
<b>Screenshot of the Accessibility Inspector</b>	

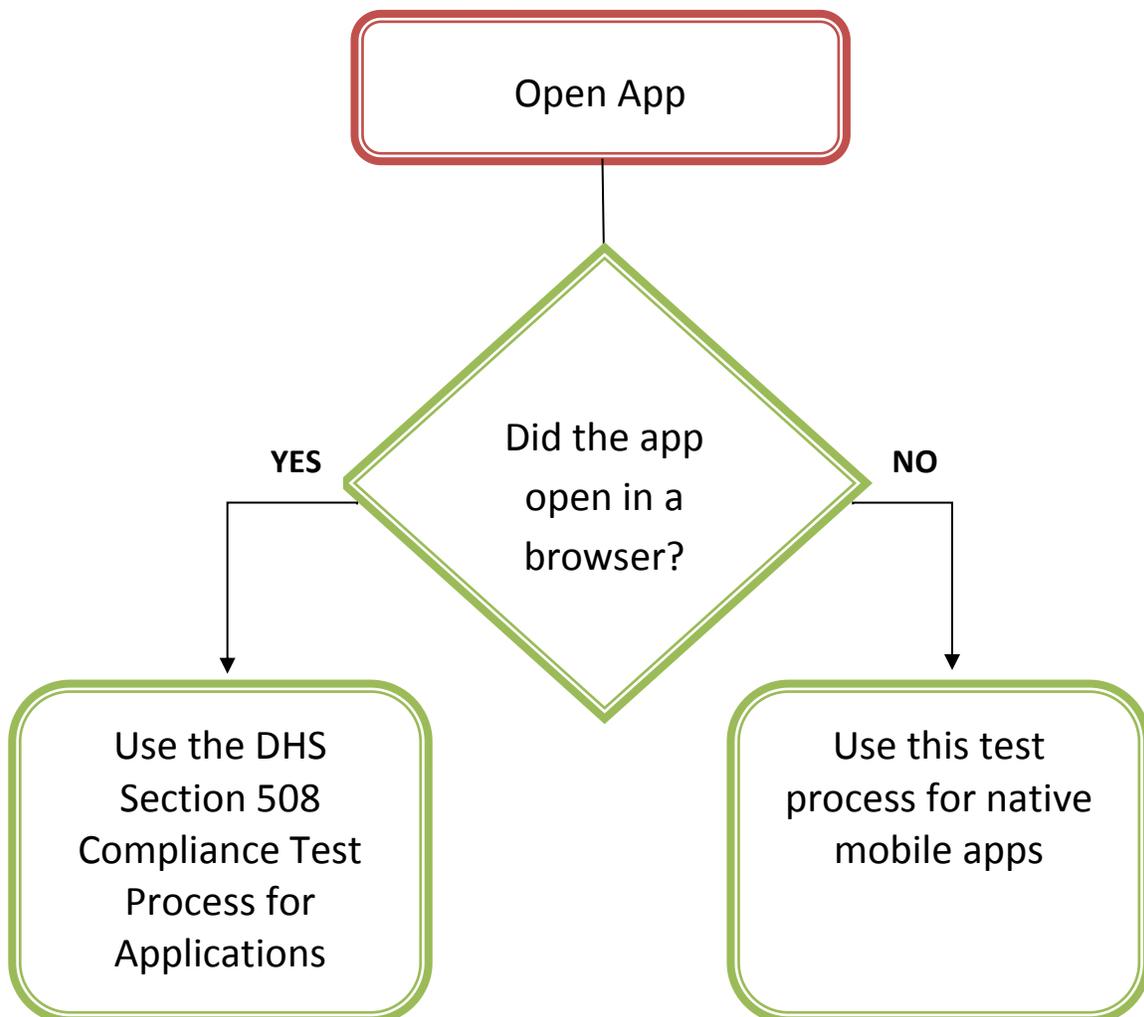
SELECT THE CORRECT TEST PROCESS FOR TESTING

It is important to identify whether the application you are testing is software or Web so that you know which test process to use and what test outcomes are expected.

- Mobile software apps are those delivered to the user via native operating system-based processes (e.g. the app is accessed by downloading through an App Store) – this includes web content embedded in a native app. These apps should be tested with the Mobile Test process.
- Mobile web interface apps are delivered to the user via the mobile Web browser. If the app opens in a web browser then use the non-mobile DHS Section 508 Compliance Test Process for Applications.

**Note: If an application has an Android OS version and iOS version of the app, the app must be tested on both platforms. Please use the iOS Mobile Application test process for all iOS apps.**

Through the testing process, **a test result can change from Compliant to Not Compliant, but do not change a result from Not Compliant to Compliant** or your results will become inaccurate.



**SECTION 3:  
DHS SECTION 508 COMPLIANCE TESTS FOR ANDROID MOBILE APPLICATIONS**

**STRUCTURE OF THIS SECTION**

Each step of the **Android** test process includes only the information that will need to be referenced frequently, namely:

- the directions on **how to test**, and
- how to interpret the **test results**, and
- the **applicable standard(s)** and the **related technical 508 standards**

To simplify the organization of this document:

- the **rationale** for each individual test has been separated out (see page 5).

## 1. KEYBOARD AND FOCUS

### 1.1 KEYBOARD ACCESS

Keyboard access must be available for all interactive interface components (links, buttons, etc), editable content (text input fields, select options, etc.), and panes that can be accessed or controlled with touch gestures. Where non-standard keyboard commands are employed, users must be notified of alternate keyboard commands through the interface, application help, and/or documentation.

For a user to know where the keyboard focus is on the screen there must be a visible indication of the currently focused component. If hidden content is revealed on the screen, the keyboard focus must move to the new content. When the sequence of interface components has meaning or requires an order of operation, the focus/TAB order must logically flow with the application/content.

Keyboard access is defined as use with a physical keyboard that is attached to the mobile device, either separately (wireless keyboard) or integrated (built-in keyboard). On-screen keyboards *cannot* be used for this test. Keyboard tests with Android must be run *without* TalkBack enabled as TalkBack will change what controls are placed in the focus order.

#### 1: KEYBOARD TASKS- HOW TO TEST

##### 1. Find all visible and hidden interactive interface components that are accessible by touch.

Interactive elements may be distinguished from non-interactive elements by using the app without a screen reader and touching each interface control to determine its level of interactivity. Only elements that can be accessed or controlled with a mouse or touch gesture must be keyboard accessible (this also includes any actions that are triggered by a single finger Tap and Hold gesture). Hidden and disabled form fields do not require keyboard access. Control options triggered by physical manipulation of the device (such as a shake, tilt, or swipe to show menu) must also be keyboard accessible.

##### 2. Ensure TalkBack is disabled.

TalkBack will change what controls are placed in the focus order and should not be enabled.

##### 3. Enable Bluetooth keyboard.

##### 4. Using only the Bluetooth keyboard, navigate through the entire screen to each interactive interface component, reveal hidden content, and activate all interactive components.

For a list of keyboard shortcuts for the Android platform, refer to [TalkBack Gestures and Keystrokes](#).

Use the Tab key to move forward through all interactive controls. Use the Shift + Tab key to move backwards through all interactive controls. Use of the arrow keys may be necessary to move between certain controls such as toolbar buttons in a toolbar. Use Enter to activate a control. Use Esc to go back.

##### 5. Note if the keyboard gets “trapped”, appears to loop or becomes stuck within an area of the application as a failure. Use touch gestures only in this situation to bypass the trap and continue testing.

##### 6. Investigate any instances where the standard platform keyboard methods do not provide access to interactive elements.

Check if alternative keyboard access instructions are provided (in the application's Help section or on the page). Instructions may include extending standard keyboard command operations (e.g., getting out of a keyboard "trap"), and/or alternate keys. Verify that the alternative keyboard commands work.

Some applications may offer multiple ways to activate a function. Only one method of keyboard access is required. Special access keys (such as Alt + O or Command + P) and redundant menu options are not required for compliance if platform shortcuts work.

**7. Complete form fields.**

Enter text, select text in input fields, select (arrow and tab) different options from drop down lists, select and unselect checkboxes and radio buttons. Try to select the second or third option from a drop down list by keyboard only. Where multiple selections in a drop down list are possible, select multiple combinations. Where single finger Tap and Hold gestures provide interactive element access, ensure keyboard access is also available (to include standard copy/paste and cut in form fields).

**8. If a single finger Tap and Hold gesture reveals interactive content (such as copy/paste), ensure access is also provided through keyboard access.**

An example of the single finger Tap and Hold gesture is a copy/cut/paste/select all. This will be applicable for most text input fields, but could also be used to perform actions such as a save or bookmark, for example.

---

**2: KEYBOARD FOCUS TASKS- HOW TO TEST**

**1. Using the keyboard, put focus at the top of the screen and Tab through each screen, following the application's visual focus (typically a dotted rectangle, highlight, or a vertical bar in a text field).**

For a list of keyboard shortcuts for the Android platform, refer to [TalkBack Gestures and Keystrokes](#).

It is not a requirement that only interactive elements receive focus. For example, form instructions in text may receive focus by design. This is not a failure.

In forms, the focus may automatically shift to the next form field once valid text is entered (e.g. after three numbers are entered for an area code, the focus immediately shifts to the field for the next three numbers of a phone number.) The focus must be visible after shifting. The user must also be able to return to the prior field to check their entry.

Navigate to a screen, then place focus at the top of the screen.

Press Tab and/or Arrows to navigate forward. Use Shift + Tab to move backwards through the focus order.

**2. Note any instances where the focus order is illogical.**

Review the entire screen starting from the top of the screen by navigating through all of the elements using the keyboard to determine whether the order in which the different elements receive focus makes sense. If the focus order can cause confusion or errors for a user it may be necessary to consult with the application developer to confirm whether this focus order is intentional.

**3. Note any instances of a loss of visual focus while Tabbing through the page or if the visual focus appears on the wrong element as a failure.**

Loss of focus should not occur while manually shifting focus through the page (using the Tab or Arrow keys). The focus must be visible when it is placed on a pane, such as a scrolling license agreement that must be scrolled to review.

When a function that moves the focus is executed (such as from the skip link itself, or when hidden content is revealed), it may be necessary to press Tab so focus becomes visible again. This is not considered a failure.

**4. When hidden content is revealed, check that the visual focus was moved to the new content.**

Move the focus to the control that reveals hidden content, activate the control with the keyboard, and determine whether focus is on the revealed content.

If a modal dialog box (requires user interaction to return to the application) is revealed, the visual focus must remain within the modal dialog box until the box is closed. If a dialog box behaves like a modal dialog box, test it as a modal dialog box.

If focus does not move to the revealed content, an accurate description of the content change must be provided. This description may be revealed in many ways, such as on screen text, tooltips, or a description on the element that reveals the hidden content. It may be necessary to advance the focus once to find the focus.

Use Tab/Arrows.

**1.1 RESULTS**

Failure condition	Does Not Apply? [DNA]	Not Compliant [NC]	Compliant [C]
<p>A. An interactive element or function cannot be accessed or activated by keyboard.</p> <p>[31(b), 31(f)]</p>	<p>Test ID always applies. [DNA] is not an acceptable result.</p>	<p>[NC] if an interactive element cannot be accessed by keyboard.</p>	<p>[C] if all interactive elements can be accessed by keyboard.</p>
<p>B. A keyboard “trap” is found.</p> <p>[31(b), 31(f)]</p>	<p>Test ID always applies. [DNA] is not an acceptable result.</p>	<p>[NC] if a keyboard “trap” is found.</p>	<p>[C] if a keyboard “trap” is not found.</p>
<p>C. Non-standard or alternative keyboard commands are required for access but are not documented.</p> <p>[31(b), 31(f)]</p>	<p>[DNA] if all interactive elements can be accessed with the standard keyboard commands. (E.g. tab, arrow keys, enter, space, etc.)</p>	<p>[NC] if non-standard keyboard commands are needed to access an interactive element and the commands are not documented.</p>	<p>[C] if non-standard keyboard commands are needed and they are documented.</p>

Failure condition	Does Not Apply? [DNA]	Not Compliant [NC]	Compliant [C]
<p>D. Information/interactive elements revealed by single finger Tap and Hold gesture are not available to keyboard-only users.</p> <p>[31(b), 31(f)]</p>	<p><b>[DNA]</b> if no elements are revealed by a single finger Tap and Hold gesture. <b>Note:</b> This will <u>apply to most text input fields.</u></p>	<p><b>[NC]</b> if information/interactive elements revealed by single finger Tap and Hold gesture are not available to keyboard-only users.</p>	<p><b>[C]</b> if information/interactive elements revealed by single finger Tap and Hold gesture are available to keyboard-only users.</p>
<p>E. At any time, there is no visual indication of the current focus (loss of focus).</p> <p>[31(b), 31(f)]</p>	<p>Test ID always applies. <b>[DNA]</b> is not an acceptable result.</p>	<p><b>[NC]</b> if a loss of visual focus occurs.</p>	<p><b>[C]</b> if the visual focus can be determined at all times.</p>
<p>F. The visual focus appears on the wrong element.</p> <p>[31(b), 31(f)]</p>	<p>Test ID always applies. <b>[DNA]</b> is not an acceptable result.</p>	<p><b>[NC]</b> if focus appears on a different element than the one that has programmatic focus.</p>	<p><b>[C]</b> if focus always appears on the element it is programmatically on.</p>
<p>G. The visual focus does not remain within a modal dialog box until closed.</p> <p>[31(b), 31(f)]</p>	<p><b>[DNA]</b> if there are no modal dialog boxes.</p>	<p><b>[NC]</b> if focus leaves the modal dialog box while the box is open.</p>	<p><b>[C]</b> if focus remains in the modal dialog box until the box is closed.</p>
<p>H. The visual focus does not move to revealed content and no description of the content change is provided.</p> <p>[31(b), 31(f)]</p>	<p><b>[DNA]</b> if there is no revealed content.</p>	<p><b>[NC]</b> if the focus does not move to the revealed content <u>AND</u> a description of the content change is not provided.</p>	<p><b>[C]</b> if focus moves to revealed content <u>OR</u> a description of the content change is provided.</p>
<p>I. The keyboard reading order is not logical.</p> <p>[31 (b), 31(f)]</p>	<p>Test ID always applies. <b>[DNA]</b> is not an acceptable result.</p>	<p><b>[NC]</b> if the focus order is not logical.</p>	<p><b>[C]</b> if the focus order is logical.</p>

**Enter the test results into the DHS reporting tool.**

[APPLICABLE 508 STANDARDS 1194.XX](#)

---

**31(b) Use with Low Vision**

At least one mode of operation and information retrieval that does not require visual acuity greater than 20/70 shall be provided in audio and enlarged print output working together or independently, or support for assistive technology used by people who are visually impaired shall be provided.

**31(f) Use with physical limitations**

At least one mode of operation and information retrieval that does not require fine motor control or simultaneous actions and that is operable with limited reach

## 1.2 GESTURE FOCUS TEST

For a TalkBack user to know where the focus is on the screen there must be a visible indication of the currently focused component while TalkBack is enabled. If hidden content is revealed on the screen, the focus must move to the new content. When the sequence of interface components has meaning or requires an order of operation, the TalkBack gesture order must logically flow with the application/content.

---

### HOW TO TEST

#### 1. Enable TalkBack.

Enable via Settings > Accessibility > TalkBack or Resume (from suspend shortcut) via long press Volume Up and Down keys

#### 2. Using gestures, put focus at the top of the screen and gesture through each screen, following the application's visual focus (typically a dotted rectangle, highlight, or a vertical bar in a text field).

For a list of gestures for the Android platform, refer to [TalkBack Gestures and Keystrokes](#).

#### 3. Note any instances where the focus order is illogical.

Review the entire screen starting from the top of the screen by navigating through all of the elements using gestures to determine whether the order in which the different elements receive focus makes sense. If the focus order can cause confusion or errors for a user it may be necessary to consult with the application developer to confirm whether this focus order is intentional.

If the focus becomes “trapped”, appears to loop or becomes stuck within an area of the application, use the touch screen in this situation to touch and bypass the trap and continue testing (gesture traps are tested in test process 2.1)

#### 4. Note any instances of a loss of visual focus while gesturing through the page or if the visual focus appears on the wrong element as a failure.

Loss of focus should not occur while manually shifting focus through the page (using the swipe gesture). The focus must be visible when it is placed on a pane, such as a scrolling license agreement that must be scrolled to review.

When a function that moves the focus is executed (such as from the skip link itself, or when hidden content is revealed), it may be necessary to perform a forwards/backwards gesture so focus becomes visible again. This is not considered a failure.

#### 5. When hidden content is revealed, check that the visual focus was moved to the new content.

Move the focus to the control that reveals hidden content, activate the control with the appropriate gesture, and determine whether focus is on the revealed content.

If a modal dialog box (requires user interaction to return to the application) is revealed, the visual focus must remain within the modal dialog box until the box is closed. If a dialog box behaves like a modal dialog box, test it as a modal dialog box.

If focus does not move to the revealed content, an accurate description of the content change must be provided. This description may be revealed in many ways: on screen text, tooltips, a description on the element that reveals the hidden content. It may be necessary to advance the focus once to find the focus.

1.2 RESULTS

Failure condition	Does Not Apply? [DNA]	Not Compliant [NC]	Compliant [C]
<p>A. At any time, there is no visual indication of the current focus (loss of focus).</p> <p>[31(a), 31(b)]</p>	<p>Test ID always applies. [DNA] is not an acceptable result.</p>	<p>[NC] if a loss of visual focus occurs.</p>	<p>[C] if the visual focus can be determined at all times.</p>
<p>B. The visual focus appears on the wrong element.</p> <p>[31(a), 31(b)]</p>	<p>Test ID always applies. [DNA] is not an acceptable result.</p>	<p>[NC] if focus appears on a different element than the one that has programmatic focus.</p>	<p>[C] if focus always appears on the element it is programmatically on.</p>
<p>C. The visual focus does not remain within a modal dialog box until closed.</p> <p>[31(a), 31(b)]</p>	<p>[DNA] if there are no modal dialog boxes.</p>	<p>[NC] if focus leaves the modal dialog box while the box is open.</p>	<p>[C] if focus remains in the modal dialog box until the box is closed.</p>
<p>D. The visual focus does not move to revealed content and no description of the content change is provided.</p> <p>[31(a), 31(b)]</p>	<p>[DNA] if there is no revealed content.</p>	<p>[NC] if the focus does not move to the revealed content <u>AND</u> a description of the content change is not provided.</p>	<p>[C] if focus moves to revealed content <u>OR</u> a description of the content change is provided.</p>
<p>E. The gesture reading order is not logical.</p> <p>[31(a), 31(b)]</p>	<p>Test ID always applies. [DNA] is not an acceptable result.</p>	<p>[NC] if the gesture reading order is not logical.</p>	<p>[C] if the gesture reading order is logical.</p>

**Enter the test results into the DHS reporting tool.**

**APPLICABLE 508 STANDARDS 1194.XX**

**31(a) Use without Vision**

At least one mode of operation and information retrieval that does not require user vision shall be provided, or support for assistive technology used by people who are blind or visually impaired shall be provided.

**31(b) Use with Low Vision**

At least one mode of operation and information retrieval that does not require visual acuity greater than 20/70 shall be provided in audio and enlarged print output working together or independently, or support for assistive technology used by people who are visually impaired shall be provided.

## 2. SCREEN READER TEST

Descriptive information must be provided for all elements to enable assistive technologies to pass useful information on to its user. The descriptive label must include identification, status and purpose of the element.

All software elements must have useful and complete descriptions.

### 2.1 ELEMENTS TEST

#### HOW TO TEST

##### 1. Enable TalkBack.

Enable TalkBack via Settings > Accessibility > TalkBack or Resume (from suspend shortcut) via long press volume up and down keys.

##### 2. Enable “Display Speech Output”.

Enable “Display Speech Output” via Settings > Accessibility > TalkBack > Settings > Developer Settings

##### 3. Using gestures only, navigate (from top to bottom, left to right) each screen and find all elements.

Interact with each element to include filling out forms, selecting buttons, reading text, etc. For form fields, also find all associated instructions and cues. In addition, locate all non-text interface elements include images, text rendered as an image, tables, graphs, charts, audio files, animations and video files.

For a list of gestures for the Android platform, refer to [TalkBack Gestures and Keystrokes](#).

##### 4. Determine if sufficient speech output is provided for every element on each screen by listening to screen reader output and comparing it to “What To List For” outcomes in the [Elements table](#).

Note: For any elements that are not accessible with TalkBack, fail under 2.1 k. Failures 2.1 A-J only apply to elements that have TalkBack output.

##### i. Non-interactive Images

1. If it is a decorative image, nothing should be announced.
2. Repeat images must have consistent meaning.
3. Images that contain meaningful text should have that same text read aloud.
4. Interactive image elements should be tested under *User Controls, Links and Interactive Elements (iv)*.

##### ii. Decorative Non-interactive Elements

1. These could be symbols (such as a separator line), images, or other decorative elements that developers have added to the app.
2. If the element is interactive AND meaningful, it should be tested under *User Controls, Links and Interactive Elements (iv)*.

##### iii. Form fields and Hints

1. Complete form fields. If a single finger Tap and Hold gesture opens menu items such as “copy”, “paste”, and “cut” (as found in most text input fields), make sure all menu items can also be activated by gesture.
2. Hidden form fields should not be announced.
3. Disabled form fields are not required to be announced, but if they are, must note they are disabled.

4. Read-only form fields must have sufficient element data read aloud.
5. Form fields must have a programmatic label.
  - a. Placeholder text for form field identification does not meet conformance. Because users can enable/disable the hint property, the hint property does not meet conformance for required control labels or status. A hint can be identified as it will be read after a short pause in TalkBack speech output.
  - b. Be sure to test items like “search” fields with and without text in the input field to ensure the label is not just placeholder text.
- iv. User Controls (Menus, Buttons, Toolbars, etc.), Links, and other Interactive Elements
  1. Activate controls, other Interactive Elements and links. Verify they function properly using gestures.
  2. Each control and other Interactive Element must reflect its name and function, to include alerting screen reader users of its name and that it is interactive (this could be achieved in a number of ways, to include programmatically setting it as a “button”, or adding label text to alert the user of the interactive control).
    - a. If a control announces its type twice (such as “Start button button”), this would be considered a failure. This may indicate from a coding perspective, that the control type has been included in the accessible name; however, control types are handled via a different trait and to avoid redundancy, do not need to be included in the accessible name.
  3. If there are multiple User Controls or other Interactive Elements with the same visual label, and context is necessary to understand purpose, the speech output must include unique identifiers that describe each control’s purpose.
    - a. In instances where context is necessary to understand the purpose of a user control other interactive element (such as multiple “Submit” buttons meaning specific submissions based on location), these should be included in the name of the element (and read aloud by the screen reader).
  4. All links should have a name that matches the onscreen text and alerts the user to the functionality of a “link”.
    - a. In instances where context is necessary to understand the purpose of a link, context should be included in the name of the link (and read aloud by the screen reader).
- v. Text
  1. Ensure all text can be accessed while using a screen reader and is correctly read aloud. Note any instances of garbled or indiscernible text as a failure.
- vi. Revealed content
  1. Ensure all revealed content provides sufficient user information for interaction.
- vii. Lists
  1. Ensure the list is read aloud to include list structure and status indication (if applicable).
- viii. Headings
  1. Visual headings should also be programmatically labeled. Android recently updated their API to allow coding of headings but heading and heading level may be noncompliant if the app has not been re-coded accordingly.
- ix. Tables
  1. Table headers should be announced. Android recently updated their API to allow coding of headers but headers may be noncompliant if the app has not been re-coded accordingly.

**5. Note any elements with multiple statuses which *provides meaningful information for user interaction*.**

If a menu opens/closes and a user MUST have the status announced in order to understand how to interact with the screen, that menu should provide its status.

For items that can change status, but do not provide meaningful information via status, it is not necessary for them to announce current status (this would be for items such as a menu that is opened/closed).

- 6. Take a screenshot of Talkback “Display Speech Output” for each element failure.
- 7. Investigate any instances where elements cannot be accessed or cannot be accessed by standard platform gesture methods.

This can include a gesture “trap” or use of non-standard or alternative gesture commands that are required for access but are not documented. Note if the tester gets “trapped”, appears to loop, or becomes stuck within an area of the application as a failure. Use touch gestures in this situation to bypass the trap and continue testing.

If a single finger Tap and Hold gesture opens menu items such as “copy”, “paste”, and “cut”, make sure *all* menu items are also able to be activated by gesture. Control options triggered by physical manipulation of the device (such as a shake, tilt, or swipe to show menu) must also be accessible when the screen reader is enabled.

Use two fingers press and hold to “long press” and open additional menus

- 8. Note if the gesturing gets “trapped”, appears to loop or becomes stuck within an area of the application as a failure. Bypass the trap and continue testing.

2.1 RESULTS

	Failure condition	Does Not Apply? [DNA]	Not Compliant [NC]	Compliant [C]
Images	A. A meaningful image does not have an equivalent description (purpose and function). [31(a)]	[DNA] if there are no meaningful images.	[NC] if a meaningful image does not have an equivalent description.	[C] if all meaningful images have an equivalent description.
	B. A decorative element is announced by the screen reader. [31(a)]	[DNA] if there are no decorative images.	[NC] if a decorative image is announced by the screen reader.	[C] if all decorative images are not announced by the screen reader.
	C. An image has inconsistent meaning. [31(a), 31(b)]	[DNA] if there are no images.	[NC] if an image has inconsistent meaning throughout the application.	[C] if all images have consistent meaning throughout the application.

	<b>Failure condition</b>	<b>Does Not Apply? [DNA]</b>	<b>Not Compliant [NC]</b>	<b>Compliant [C]</b>
	<p>D. An image that contains text does not have the same text announced by the screen reader.</p> <p>[31(a)]</p>	<p><b>[DNA]</b> if there are no images with text.</p>	<p><b>[NC]</b> if an image that contains text does not have the same text announced by the screen reader.</p>	<p><b>[C]</b> if all images that contain text have all the text announced by the screen reader.</p>
	<p>E. Any element that has multiple statuses that is meaningful for interaction does not indicate its current status.</p> <p>[31(a), 31(b)]</p>	<p><b>[DNA]</b> if no elements have multiple statuses.</p>	<p><b>[NC]</b> if an element with multiple statuses does not indicate its current status.</p>	<p><b>[C]</b> if all elements with multiple statuses indicate their current status.</p>
<b>Form Fields</b>	<p>F. The screen reader output for form fields (text field, checkbox, radio button, etc.) does not match its visual label <u>or</u> does not include complete instructions and cues.</p> <p>[31(a)]</p>	<p><b>[DNA]</b> if there are no form fields.</p>	<p><b>[NC]</b> if the screen reader output does not match the form field's visual label <u>or</u> does not include complete instructions and cues</p>	<p><b>[C]</b> if the screen reader output matches all the form field's visual label <u>or</u> includes complete instructions and cues</p>
<b>User Controls, Links and Other Interactive Elements</b>	<p>G. A user control, link, or other interactive element does not have a descriptive and unique name and does not identify purpose (to include alerting the user of the <i>interactive</i> function of the control).</p> <p>[31(a)]</p>	<p><b>[DNA]</b> if there are no user controls, links, and/or other interactive elements.</p>	<p><b>[NC]</b> if a user control, link, and/or other interactive element does not have a descriptive and unique name and does not identify its purpose</p>	<p><b>[C]</b> if all user controls, links, and/or other interactive element have a descriptive and unique name and identifies their purpose.</p>

	Failure condition	Does Not Apply? [DNA]	Not Compliant [NC]	Compliant [C]
<b>Headings</b>	H. Visually apparent headings are not programmatically identified. [31(a)]	[DNA] if there are no visually apparent headings on the page.	[NC] if a visually apparent heading is not programmatically identified.	[C] if all visually apparent headings are programmatically identified.
	I. Programmatically identified heading levels do not match the visual outline. [31(a)]	[DNA] if there are no visually apparent headings on the page. [DNA] if there is only 1 visually apparent heading on the page.	[NC] if programmatic levels for visually apparent headings do not match the visual structure.	[C] if programmatic levels on all visually apparent headings match the visual structure.
<b>Catch All</b>	J. Every element does not provide sufficient information for user operation (otherwise not listed). [31(a)]	[DNA] if all failures were captured under Failure Conditions A-I.	[NC] if all elements do not provide sufficient information for use operation	[C] if all elements provide sufficient information for user operation.
<b>Gestures</b>	K. An element or function cannot be accessed or activated by gesture. [31(a)]	Test ID always applies. [DNA] is not an acceptable result.	[NC] if an interactive element cannot be accessed by gesture	[C] if all interactive elements can be accessed by gesture.
	L. A gesture "trap" is found. [31(a)]	Test ID always applies. [DNA] is not an acceptable result.	[NC] if a gesture "trap" is found.	[C] if a gesture "trap" is not found.

Failure condition	Does Not Apply? [DNA]	Not Compliant [NC]	Compliant [C]
<p>M. Non-standard or alternative gesture commands are required for access but are not documented.</p> <p>[31(a)]</p>	<p>[DNA] if all interactive elements can be accessed with the standard gesture commands.</p>	<p>[NC] if non-standard gesture commands are needed to access an interactive element and the commands are not documented.</p>	<p>[C] if non-standard gesture commands are needed and they are documented.</p>
<p>N. Information/interactive elements revealed by single finger Tap and Hold gesture are not available to screen reader users.</p> <p>[31(a)]</p>	<p>[DNA] if no elements are revealed by a single finger Tap and Hold gesture.</p>	<p>[NC] if information/interactive elements revealed by single finger Tap and Hold gesture are not available to screen reader users.</p>	<p>[C] if information/interactive elements revealed by single finger Tap and Hold gesture are available to screen reader users.</p>

*Enter the test results into the DHS reporting tool.*

**APPLICABLE 508 STANDARDS 1194.XX**

**31(a) Use without Vision**

At least one mode of operation and information retrieval that does not require user vision shall be provided, or support for assistive technology used by people who are blind or visually impaired shall be provided.

**31(b) Use with Low Vision**

At least one mode of operation and information retrieval that does not require visual acuity greater than 20/70 shall be provided in audio and enlarged print output working together or independently, or support for assistive technology used by people who are visually impaired shall be provided.

## 2.2 LAYOUT TEST

### HOW TO TEST

1. **With device Screen Reader and Auto-rotate enabled, change the screen from portrait to landscape by rotating the device. Continue rotating the device 90 degrees to portrait then landscape, back to the initial screen position. Verify the screen reader announced the proper orientation of the screen.**

Enable TalkBack via Settings > Accessibility > TalkBack or Resume (from suspend shortcut) via long press volume up and down keys.

Enable Auto-rotate screen via Settings>Accessibility> Enable Auto-rotate screen

Layout changes such as changes from landscape to portrait must be announced *if they occur*. If the app supports this layout change, the screen reader should announce this change.

This requirement is *NOT* that the app *must* support orientation change.

### 2.2 RESULTS

Failure condition	Does Not Apply? [DNA]	Not Compliant [NC]	Compliant [C]
<p>A. Layout changes such as changes from portrait to landscape or a forced application direction is not indicated or announced by screen reader with sufficient information for user to orient device.</p> <p>[31(a)]</p>	<p><b>[DNA]</b> if app does not allow layout changes.</p>	<p><b>[NC]</b> if there is not sufficient indication of layout change such as a change from landscape to portrait.</p>	<p><b>[C]</b> if there is sufficient indication of layout change such as a change from landscape to portrait.</p>

**Enter the test results into the DHS reporting tool.**

### APPLICABLE 508 STANDARDS 1194.XX

#### 31(a) Use without Vision

At least one mode of operation and information retrieval that does not require user vision shall be provided, or support for assistive technology used by people who are blind or visually impaired shall be provided.

### 3. VIDEO, AUDIO, AND MULTIMEDIA

This section addresses audio, video, animation and multimedia.

---

#### 3.1 VIDEO-ONLY AND ANIMATION

Videos (without audio) and animation includes sequences of overlaid images, dynamic changes of state such as a moving speed dial, a chart illustrating dynamic flow changes from one state to another, etc. This requirement applies to visual/video information that is not accompanied by meaningful sounds.

Note: This test is different from the test for Multimedia (combined audio AND video). Files that have both video and meaningful audio are tested in Test Process 3.3 Multimedia. Do not test Multimedia here.

---

#### HOW TO TEST

- 1. Find interface components that play video-only or animated content.**
- 2. Check that video-only content that starts automatically has a playback control (pause/stop).**

Note: the control button(s) must also be tested under the Keyboard and Focus and Screen Reader tests.

- 3. Check that any animation that (1) starts automatically (2) lasts more than 5 seconds and (3) is presented in parallel with other content has a mechanism for the user to pause, stop, or hide it or to control the frequency of the update unless the auto-updating is part of an activity where it is essential.**

An animation that occurs as part of a preload phase or similar situation can be considered essential if interaction cannot occur during that phase for all users and if not indicating progress could confuse users or cause them to think that content was frozen or broken.

- 4. Check that the information conveyed through animation or video is also available in accessible screen text or audio.**

Animation or video must not be the only way to convey information.

- 5. The alternative equivalent description must be an accurate and complete representation of the video/animated content.**

All relevant visual information must be included (e.g.: driver hands license to officer). Equivalent (unsynchronized) audio descriptions for video files are acceptable but not required.

3.1 RESULTS

Failure condition	Does Not Apply? [DNA]	Not Compliant [NC]	Compliant [C]
<p>A. Video-only content that starts automatically does not have a playback control.</p> <p>[31(a)]</p>	<p>[DNA] if there are no video files. [DNA] if there are no videos that start automatically.</p>	<p>[NC] if video-only content that starts automatically does not have a playback control.</p>	<p>[C] if all video-only that starts automatically has a playback control.</p>
<p>B. Animation that starts automatically, lasts more than 5 seconds and is presented in parallel with other content does not have a playback control or allow control of frequency for update.</p> <p>[31(a)]</p>	<p>[DNA] if there is no animation. [DNA] if there is no animation that starts automatically, lasts more than 5 seconds, or is presented in parallel with other content.</p>	<p>[NC] if animation that starts automatically, lasts more than 5 seconds and is presented in parallel with other content does not have a playback control or allow control of frequency for update.</p>	<p>[C] if all animation that starts automatically, lasts at least 5 seconds and is presented in parallel with other content has a playback control or allows control of update frequency.</p>
<p>C. Video-only or animated content does not have an equivalent text or audio description.</p> <p>[31(a)]</p>	<p>[DNA] if there are no videos or animations.</p>	<p>[NC] if video-only or animated content does not have an equivalent text or audio description.</p>	<p>[C] if all video-only and animated content has an equivalent text or audio description.</p>

*Enter the test results into the DHS reporting tool.*

APPLICABLE 508 STANDARDS 1194.XX

**31(a) Use without Vision**

At least one mode of operation and information retrieval that does not require user vision shall be provided, or support for assistive technology used by people who are blind or visually impaired shall be provided.

### 3.2 AUDIO-ONLY

Audio-only files include speeches, soundbites, ambient (background) sounds, etc.

Note: This test is different from the test for Multimedia (combined audio AND video). Files that have both video and meaningful audio are tested in Test Process 3.3 Multimedia. Do not test Multimedia here.

#### HOW TO TEST

**1. Find all interface components that play audio-only content.**

Short sounds such as confirmation beeps and error notifications are not included in this requirement.

**2. If the audio starts automatically and lasts for more than 3 seconds, check that the audio-only content has a playback control (pause/stop) or a mechanism to control audio volume independently from the overall system volume level.**

Note: the control button(s) must also be tested under the Keyboard and Focus and Screen Reader tests.

**3. Find the associated transcript or description.**

**4. Play audio and check that the information conveyed through audio is also available in accessible screen text/transcript.**

Audio must not be the only way to convey information.

The alternative equivalent description/transcript must be an accurate and complete representation of the audio-only content. All speech and relevant audio cues must be included (e.g.: doorbell rings, car honks, etc.)

#### 3.2 RESULTS

Failure condition	Does Not Apply? [DNA]	Not Compliant [NC]	Compliant [C]
<p>A. Audio content that plays automatically and lasts for more than 3 seconds does not have a playback control or a mechanism to independently control audio volume.</p> <p>[31(c)]</p>	<p><b>[DNA]</b> if there are no audio only files. <b>[DNA]</b> if there are no audio only files that start automatically and last for more than 3 seconds.</p>	<p><b>[NC]</b> if audio content that plays automatically and lasts for more than 3 seconds does not have a playback control nor a mechanism to independently control audio volume.</p>	<p><b>[C]</b> if all audio content that plays automatically and lasts for more than 3 seconds has a playback control or a mechanism to independently control audio volume.</p>
<p>B. An audio file does not have an equivalent text description/transcript.</p> <p>[31(c)]</p>	<p><b>[DNA]</b> if there are no audio only files.</p>	<p><b>[NC]</b> if an audio-only file does not have an equivalent text description/transcript.</p>	<p><b>[C]</b> if all audio-only files have an equivalent text description/transcript.</p>

*Enter the test results into the DHS reporting tool.*

[APPLICABLE 508 STANDARDS 1194.XX](#)

---

**31(c) Use without Hearing**

At least one mode of operation and information retrieval that does not require user hearing shall be provided, or support for assistive technology used by people who are deaf or hard of hearing shall be provided.

### 3.3 MULTIMEDIA

Multimedia is a presentation consisting of time-synchronized media (primarily video and audio).

#### HOW TO TEST

**1. Turn Captions on through the OS Accessibility properties.**

Settings > Accessibility> Captions On

Set your caption options to something unique like yellow on blue text, so it is obvious that the caption options are adopted when testing.

**2. Find interface components that play synchronized media when activated.**

This includes streaming media, and streaming live events. Include only files that have meaningful visual information that is synchronized with meaningful audio information. Examples include webcasts, press conferences, instructional videos, and online training presentations.

**3. If the multimedia starts automatically, check that the multimedia content has a playback control (pause/stop).**

Note: the control button(s) must also be tested under the Keyboard and Focus and Screen Reader tests.

**4. Enable captions in the multimedia player. Play the multimedia.**

Check captions: All relevant audible information (dialog and sounds) from the multimedia should be complete, accurately displayed and synchronized with the multimedia audio.

**5. Check that the application adopted the OS Accessibility Caption settings, that the application did not crash, and that all functionality is still available.**

**6. Enable the audio description feature in the multimedia player. Play the multimedia.**

Check audio descriptions: All relevant visual information from the multimedia should be complete, verbally described accurately and synchronized with the multimedia video.

**7. Synchronized alternatives are required. Transcripts and separate lists of visual events do not meet this requirement.**

#### 3.3. RESULTS

Failure condition	Does Not Apply? [DNA]	Not Compliant [NC]	Compliant [C]
<p>A. Multimedia content that starts automatically does not have a playback control.</p> <p>[31(a), 31(c)]</p>	<p><b>[DNA]</b> if there are no multimedia files. <b>[DNA]</b> if there are no multimedia files that start automatically.</p>	<p><b>[NC]</b> if multimedia content that starts automatically does not have a playback control.</p>	<p><b>[C]</b> if all multimedia content that starts automatically has a playback control.</p>

Failure condition	Does Not Apply? [DNA]	Not Compliant [NC]	Compliant [C]
B. Synchronized captions are not provided for multimedia. [31 (c)]	[DNA] if there are no multimedia files.	[NC] if a multimedia file does not provide synchronized captions.	[C] if all multimedia files provide synchronized captions.
C. The provided captions for multimedia are not equivalent. [31 (c)]	[DNA] if there are no multimedia files. [DNA] if captions are not provided.	[NC] if the provided captions for a multimedia file are not equivalent.	[C] if the provided captions for all multimedia files are equivalent.
D. Synchronized audio descriptions are not provided for multimedia. [31 (a)]	[DNA] if there are no multimedia files.	[NC] if a multimedia file does not provide synchronized audio descriptions.	[C] if all multimedia files provide synchronized audio descriptions
E. The provided audio descriptions are not equivalent. [31 (a)]	[DNA] if there are no multimedia files. [DNA] if audio descriptions are not provided.	[NC] if the provided audio descriptions for a multimedia file are not equivalent.	[C] if the provided audio descriptions for all multimedia files are equivalent.
F. The application did not adopt all of the Android Caption settings. [31 (c)]	[DNA] if there are no multimedia files. [DNA] if there are no captions.	[NC] if the application disrupts caption settings. <b>Note:</b> This will also fail 7.F.	[C] if Android caption settings are not disrupted by the application.

**Enter the test results into the DHS reporting tool.**

**APPLICABLE 508 STANDARDS 1194.XX**

**31(a) Use without Vision**

At least one mode of operation and information retrieval that does not require user vision shall be provided, or support for assistive technology used by people who are blind or visually impaired shall be provided.

**31(c) Use without Hearing**

At least one mode of operation and information retrieval that does not require user hearing shall be provided, or support for assistive technology used by people who are deaf or hard of hearing shall be provided.

## 4. COLOR AND CONTRAST

### 4.1 COLOR DEPENDENCE

Color dependence is using color as the sole means to convey information. For example, a single icon that is green for ‘on’, orange for ‘standby’, and red for ‘off’ is color dependent.

Color must not be the only means of conveying information, indicating an action, prompting a response, or indicating status. Information conveyed through color must also be provided in text on the screen.

#### HOW TO TEST

1. If there is information that is being provided by color, check to see if the same information is represented textually on the screen or through other visual (non-color) differentiation (shape, position, size).
2. Check that the redundant textual information is visible when the meaningful color is displayed (i.e., the user should not have to do anything to reveal the text).

#### 4.1 RESULTS

Failure condition	Does Not Apply? [DNA]	Not Compliant [NC]	Compliant [C]
A. Information is provided only by color. [31(a)]	[DNA] if color is not used to provide information.	[NC] if color is the only method used to provide information.	[C] if color is used but is not the only method to provide information.

*Enter the test results into the DHS reporting tool.*

#### APPLICABLE 508 STANDARDS 1194.XX

##### 31(a) Use without Vision

At least one mode of operation and information retrieval that does not require user vision shall be provided, or support for assistive technology used by people who are blind or visually impaired shall be provided.

## 4.2 CONTRAST

There must be contrasting colors/shades at a ratio of at least 4.5:1 for discerning between background and foreground content.

### HOW TO TEST

1. **Visually examine meaningful text and images of meaningful text displayed on the screen for areas that may have low background to foreground contrast.**

Incidental text, as defined below, is exempt from this requirement:

- Part of logos or brand names
- Part of an inactive user interface component
- Pure decoration
- Not visible to anyone
- Part of a picture that contains significant other visual content

All appearances of meaningful text should be evaluated, including changes due to status.

2. **Take a screenshot and transfer it to a computer.**
3. **Use the Colour Contrast Analyser picker tool to select foreground and background colors from the screen.**

The contrast ratio must be **at least 4.5:1**.

### 4.2 RESULTS

Failure condition	Does Not Apply? [DNA]	Not Compliant [NC]	Compliant [C]
<p>A. The contrast ratio is less than 4.5:1 for content background and foreground colors.</p> <p>[31(b)]</p>	[DNA] if there is no text.	[NC] if the contrast ratio is less than 4.5:1 when comparing the background and foreground colors. (If NC, always document the contrast ratio.)	[C] if the contrast ratio is 4.5:1 or greater when comparing all background and foreground colors.

**Enter the test results into the DHS reporting tool.**

### APPLICABLE 508 STANDARDS 1194.XX

#### 31(b) Use with Low Vision

At least one mode of operation and information retrieval that does not require visual acuity greater than 20/70 shall be provided in audio and enlarged print output working together or independently, or support for assistive technology used by people who are visually impaired shall be provided.

## 5. FLASHING

The term ‘flickering’ encompasses interface elements that flash, blink, flicker repetitively, or elements that scroll (e.g., marquee text).

Current Section 508 requirements do not permit elements to flash or flicker at frequencies between 2Hz and 55Hz. To incorporate the expected update to this requirement, this flashing test adopts the WCAG 2.0 frequency restriction to below 3Hz.

### HOW TO TEST

**1. Visually check for any flashing or blinking interface elements or scrolling text.**

Currently there is no way to accurately test the rate of flashing or flickering on mobile devices and thus use of flashing or flickering elements will be found not compliant.

Scrolling text should also be tested under Test 2.1: Screen Reader test.

### 5. RESULTS

Failure condition	Does Not Apply? [DNA]	Not Compliant [NC]	Compliant [C]
A.If the page contains flashing, flickering, or scrolling. [21(k)]	<b>[DNA]</b> if there are no elements that are flashing, flickering, scrolling, etc.	<b>[NC]</b> if flashing, flickering, or scrolling occurs.	<b>[C]</b> Does not apply.

*Enter the test results into the DHS reporting tool.*

### RELATED TECHNICAL 508 STANDARDS 1194.XX

#### 21(k) Blinking Objects

Software shall not use flashing or blinking text, objects, or other elements having a flash or blink frequency greater than 2 Hz and lower than 55 Hz.

6. TIMEOUTS

Messages and/or instructions to the user requesting their response within a given time are typically associated with apps that require a secure login. This includes both server time-outs and client side security time-outs.

Note: The accessibility of the alert/pop up and option to request more time is tested in the Keyboard and Screen Reader tests (Tests 1 and 2) and should not be tested here.

HOW TO TEST

1. **Inspect the app to determine if there is a timeout function from the application’s documentation, or by leaving the session inactive for a period of time.**
2. **If a timeout is about to occur, an alert must be posted for at least 20 seconds and the user must have the option to request more time.**

6. RESULTS

Failure condition	Does Not Apply? [DNA]	Not Compliant [NC]	Compliant [C]
A. The application timed out without notification. [31(a), 31(b), 31(c), 31 (d), 31(f)]	[DNA] if application does not time out.	[NC] if application times out without notification.	[C] if application provides notification before timing out.
B. The application’s time out notification is displayed for less than 20 seconds. [31(f)]	[DNA] if application does not time out. [DNA] if no time out notification is provided.	[NC] if application’s time out notification is displayed for less than 20 seconds before timing out.	[C] if application’s time out notification is displayed for at least 20 seconds before timing out.
C. The application timed out without an option to request more time. [31(a), 31(b), 31(c), 31 (d), 31(f)]	[DNA] if application does not time out.	[NC] if application timed out without providing user an option to request more time.	[C] if application provides user an option to request more time before timing out.

**Enter test results into the DHS reporting tool.**

APPLICABLE 508 STANDARDS 1194.XX

**31(a) Use without Vision**

At least one mode of operation and information retrieval that does not require user vision shall be provided, or support for assistive technology used by people who are blind or visually impaired shall be provided.

**31(b) Use with Low Vision**

At least one mode of operation and information retrieval that does not require visual acuity greater than 20/70 shall be provided in audio and enlarged print output working together or independently, or support for assistive technology used by people who are visually impaired shall be provided.

**31(c) Use without Hearing**

At least one mode of operation and information retrieval that does not require user hearing shall be provided, or support for assistive technology used by people who are deaf or hard of hearing shall be provided.

**31(d) Use with Hard of Hearing**

Where audio information is important for the use of a product, at least one mode of operation and information retrieval shall be provided in an enhanced auditory fashion, or support for assistive hearing devices shall be provided.

**31(f) Use with physical limitations**

At least one mode of operation and information retrieval that does not require fine motor control or simultaneous actions and that is operable with limited reach.

## 7. BUILT-IN ACCESSIBILITY FEATURES

Operating System (OS) accessibility features are applications that are separate from the software being tested. The features are usually found and activated in the accessibility section of the Settings menu on the device.

OS user-configurable accessibility appearance settings, functions, and assistive technologies must not be disrupted or disabled by the software application.

Note: The navigation instructions to access these features may differ depending on OS updates. If you are testing with an Android device that has an additional User Interface, Accessibility Features could vary.

### HOW TO TEST

#### 1. Enable Accessibility Settings:

Settings > Accessibility:

- *Font Size*
- *Display Size*
- *Magnification*
- *Select to Speak*
- *Voice Input (not part of Accessibility menu, but part of onscreen keyboard)*

#### 2. Set Font Size to the largest setting.

Settings > Accessibility > Font Size: move slider to the far right (largest size)

**Restart the app. Test to see if the app adopted large font size wherever text is present within the app and did not disrupt user's ability to read or navigate the screen.**

Close the app. Disable large Font size.

Settings > Accessibility > Font Size: move to "default" size

#### 3. Set Display Size to the largest setting.

Settings > Accessibility > Display Size: move slider to the far right (largest size)

**Restart the app. Test to see if the app adopted large display size completely within the app and did not disrupt user's ability to read or navigate the screen.**

Close the app. Disable large Display Size.

Settings > Accessibility > Display Size: move to "default" size

#### 4. Check if the application offers at least 4 color options to change foreground and background colors.

#### 5. Enable Magnification feature.

Settings > Accessibility > Magnification

- If magnification appears on but the screen does not magnify, tap one finger three times.
- To adjust the zoom level, pinch in or out to increase the magnification or move down to decrease the magnification.
- Try to make the magnification level approximately 200% of the normal size. That means the normal screen would take up approximately 4 magnified screens.
- To move around the screen drag two fingers around the screen.

**6. Restart the application.**

**7. Check that the application adopted the magnification appearance, the application did not crash, and all functionality is still available.**

**8. Close the application and disable Magnification.**

Settings > Accessibility > Magnification: Off

**9. Enable Select to Speak**

Settings > Accessibility > Select to Speak: On

**Restart the app. Select the entire screen and ensure that all text is read aloud.**

Note: Characters like “bullet” or “slash” will not be read nor will attributes like “heading” or “button”. This is not a failure. Check only that the text output matches.

- Tap the Select to Speak icon then drag your finger across the entire screen to highlight the content.
- To stop spoken text, tap anywhere on the screen.

**10. Find all non-character disappearing text input fields (such as text fields, subject/body of an email, etc.).**

Settings> Languages and Input > Virtual keyboard: ensure you are using the default Gboard and Google voice typing

**Note: Do not include text input fields here that do not display the characters, such as password fields. These are excluded from this test.**

Put focus in the non-character disappearing text input field, then select the “microphone” input on your onscreen keyboard and dictate in to the text field. Verify that the application allows you to utilize the microphone for text input.

7. RESULTS

Failure condition	Does Not Apply? [DNA]	Not Compliant [NC]	Compliant [C]
<p>A. The app did not completely adopt largest Font size.</p> <p>[31(b)]</p>	<p><b>[DNA]</b> if the app has no text.</p>	<p><b>[NC]</b> if all text is not enlarged to setting specifications.</p>	<p><b>[C]</b> if all text is enlarged to setting specifications.</p>
<p>B. The app did not completely adopt largest Display size.</p> <p>[31(b)]</p>	<p><b>[DNA]</b> is not an acceptable result.</p>	<p><b>[NC]</b> if all the display is not enlarged to setting specifications.</p>	<p><b>[C]</b> if all the display is enlarged to setting specifications.</p>
<p>C. The application does not offer at least 4 color options.</p> <p>[31(b)]</p>	<p><b>[DNA]</b> is not an acceptable result.</p>	<p><b>[NC]</b> if the application does not offer at least 4 color options.</p>	<p><b>[C]</b> if application offers at least 4 color options.</p>
<p>D. The application did not enlarge or became illegible when enlarged with magnification.</p> <p>[31(b)]</p>	<p><b>[DNA]</b> is not an acceptable result.</p>	<p><b>[NC]</b> if the application did not enlarge or became illegible when enlarged.</p>	<p><b>[C]</b> if the application enlarged and remains legible when enlarged.</p>
<p>E. Select to Speak did not read all the text.</p> <p>[31(b)]</p>	<p><b>[DNA]</b> if there is no text.</p>	<p><b>[NC]</b> if Select to Speak did not read all the screen text.</p>	<p><b>[C]</b> if Select to Speak read all the screen text.</p>
<p>F. The application did not allow user to utilize the microphone for text input.</p> <p>[31(f)]</p>	<p><b>[DNA]</b> if the app has no text input fields. <b>[DNA]</b> if the app <b>only</b> has character-disappearing form fields.</p>	<p><b>[NC]</b> if the application did not allow user to utilize the microphone for text input.</p>	<p><b>[C]</b> if the application allows user to utilize the microphone for text input.</p>

Failure condition	Does Not Apply? [DNA]	Not Compliant [NC]	Compliant [C]
<p>G. The application disrupted any of the Android Accessibility options (TalkBack, Magnification, Largest Font size, Largest Display size, Select to Speak Caption settings).</p> <p>[31(a), 31(b), 31(c), 31(d), 31(f)]</p>	<p>[DNA] is not an acceptable result.</p>	<p>[NC] if the application disrupts any of the Android Accessibility options.</p>	<p>[C] if all Android Accessibility options are not disrupted by the application.</p>

*Enter test results into the DHS reporting tool.*

APPLICABLE 508 STANDARDS 1194.XX

**31(a) Use without Vision**

At least one mode of operation and information retrieval that does not require user vision shall be provided, or support for assistive technology used by people who are blind or visually impaired shall be provided.

**31(b) Use with Low Vision**

At least one mode of operation and information retrieval that does not require visual acuity greater than 20/70 shall be provided in audio and enlarged print output working together or independently, or support for assistive technology used by people who are visually impaired shall be provided.

**31(c) Use without Hearing**

At least one mode of operation and information retrieval that does not require user hearing shall be provided, or support for assistive technology used by people who are deaf or hard of hearing shall be provided.

**31(d) Use with Hard of Hearing**

Where audio information is important for the use of a product, at least one mode of operation and information retrieval shall be provided in an enhanced auditory fashion, or support for assistive hearing devices shall be provided.

**31(f) Use with physical limitations**

At least one mode of operation and information retrieval that does not require fine motor control or simultaneous actions and that is operable with limited reach.

**8. ACCESSIBLE ALTERNATIVE**

Alternative versions for accessibility are permitted only when the primary application cannot be made compliant due to technology limitations and business requirements. The alternative version must provide the same information as the primary page.

**HOW TO TEST**

- 1. Determine if any alternative versions of the mobile app are provided.**
- 2. Check that the alternative contains equivalent information as the mobile app.**

Compare the content of the mobile app and the alternative, noting any information differences. Ensure that dynamic content are also dynamically updated on the accessible version.

- 3. Perform all applicable tests for the accessible version and mark the appropriate test results.**

**8. RESULTS**

Failure condition	Does Not Apply? [DNA]	Not Compliant [NC]	Compliant [C]
<p>A. An alternative does not contain equivalent information as the primary application.</p> <p>[31(a), 31(b), 31(c), 31(d), 31(e), 31(f)]</p>	<p>[DNA] if there is only one version of the application.</p>	<p>[NC] if an alternative does not contain equivalent information as the primary.</p>	<p>[C] if all alternatives contain equivalent information as the primary.</p>
<p>B. The primary application can be made compliant.</p> <p>[31(a), 31(b), 31(c), 31(d), 31(e), 31(f)]</p>	<p>[DNA] if there is only one version of the application.</p>	<p>[NC] if the primary can be made compliant.</p>	<p>[C] if the primary cannot be made compliant.</p>

*Enter test results into the DHS reporting tool.*

**APPLICABLE 508 STANDARDS 1194.XX**

**31(a) Use without Vision**

At least one mode of operation and information retrieval that does not require user vision shall be provided, or support for assistive technology used by people who are blind or visually impaired shall be provided.

**31(b) Use with Low Vision**

At least one mode of operation and information retrieval that does not require visual acuity greater than 20/70 shall be provided in audio and enlarged print output working together or independently, or support for assistive technology used by people who are visually impaired shall be provided.

**31(c) Use without Hearing**

At least one mode of operation and information retrieval that does not require user hearing shall be provided, or support for assistive technology used by people who are deaf or hard of hearing shall be provided.

**31(d) Use with Hard of Hearing**

Where audio information is important for the use of a product, at least one mode of operation and information retrieval shall be provided in an enhanced auditory fashion, or support for assistive hearing devices shall be provided.

**31(e) Use without Speech**

At least one mode of operation and information retrieval that does not require user speech shall be provided, or support for assistive technology used by people with disabilities shall be provided.

**31(f) Use with Physical Limitations**

At least one mode of operation and information retrieval that does not require fine motor control or simultaneous actions and that is operable with limited reach.

ELEMENTS TABLE

Element Type	What to Listen For	Android Specific Output
<b>Activity Indicator/Progress Bar</b>	<ul style="list-style-type: none"> <li>• Unique and descriptive label to match visual information</li> <li>• Notation of whether or not activity is occurring</li> </ul>	<ul style="list-style-type: none"> <li>• Unique and descriptive label to match visual information</li> <li>• Notation of whether or not activity is occurring</li> </ul>
<b>Alert/Pop Up</b>	<ul style="list-style-type: none"> <li>• Unique and descriptive speech output to match visual information</li> <li>• Includes the word “Alert” or “Pop Up”</li> <li>• If a time out: an alert must be posted for at least 20 seconds and the user must have the option to request more time read aloud.</li> </ul>	<ul style="list-style-type: none"> <li>• “Alert”</li> <li>• Unique and descriptive speech output to match visual information</li> <li>• If a time out: an alert must be posted for at least 20 seconds and the user must have the option to request more time</li> </ul>
<b>Animation</b>	<ul style="list-style-type: none"> <li>• Unique and descriptive speech output to match visual information to include note that it is a graphic or animation</li> </ul>	<ul style="list-style-type: none"> <li>• Unique and descriptive speech output to match visual information</li> <li>• Notes that it is a graphic</li> </ul>
<b>App Name (associated with App icon)</b>	<ul style="list-style-type: none"> <li>• Unique and descriptive title to match visual information (to describe app)</li> </ul>	<ul style="list-style-type: none"> <li>• Unique and descriptive title to match visual information (to describe app)</li> </ul>
<b>Button (to include play/pause/stop, stepper, sort)</b>	<ul style="list-style-type: none"> <li>• Unique and descriptive label that matches text (supplemental information might also be read to describe the button)</li> <li>• Note that is it a “Button”</li> <li>• Note its value/state (to include sufficient information for buttons that change from play to pause)</li> <li>• If the element has a variable status, the current status must be announced.</li> </ul>	<ul style="list-style-type: none"> <li>• Unique and descriptive label that matches text (supplemental information might also be read to describe the button)</li> <li>• “Button”</li> <li>• “Disabled” if button is disabled</li> <li>• Depending on type of button, say a filter button, will also say “Alert”, “Filter By”</li> </ul>
<b>Container</b>	<ul style="list-style-type: none"> <li>• Unique and descriptive name that matches visual information within container should be read</li> </ul>	<ul style="list-style-type: none"> <li>• Unique and descriptive name that matches visual information within container should be read, not container itself</li> </ul>
<b>Custom Interface Control (e.g. Calendar, Clock)</b>	<ul style="list-style-type: none"> <li>• Unique and descriptive name that matches visual information to include value/state</li> </ul>	<ul style="list-style-type: none"> <li>• Unique and descriptive title to match visual information</li> </ul>
<b>Drawer/Slide Out Menu (e.g. Navigation)</b>	<ul style="list-style-type: none"> <li>• Unique and descriptive name that matches visual information</li> </ul>	<ul style="list-style-type: none"> <li>• Unique and descriptive title to match visual information</li> <li>• “Open navigation drawer” will be noted when initially landing on element</li> </ul>

Element Type	What to Listen For	Android Specific Output
<b>Folder</b>	<ul style="list-style-type: none"> <li>• Unique and descriptive name that matches visual information</li> </ul>	<ul style="list-style-type: none"> <li>• Descriptive label, layout of folder</li> </ul>
<b>Form Field: Check box</b>	<ul style="list-style-type: none"> <li>• Unique and descriptive label that matches visual information and instructions</li> <li>• “Checkbox” should be announced)</li> <li>• State should be indicated</li> <li>• Hidden form fields should not be announced</li> <li>• Disabled form fields do not require a form name to be announced</li> <li>• Read-only form fields cannot be edited by the user, but sufficient element data must be read aloud</li> </ul>	<ul style="list-style-type: none"> <li>• Unique and descriptive label that matches visual information and instructions</li> <li>• “Checkbox” and will always say “Checked” or “Not Checked” to indicate state</li> <li>• “Disabled” will be announced if not enabled</li> </ul>
<b>Form Field: Radio button</b>	<ul style="list-style-type: none"> <li>• Unique and descriptive label that matches visual information and instructions</li> <li>• “Radio button” should be announced)</li> <li>• State should be indicated</li> <li>• Hidden form fields should not be announced</li> <li>• Disabled form fields do not require a form name to be announced</li> <li>• Read-only form fields cannot be edited by the user, but sufficient element data must be read aloud</li> </ul>	<ul style="list-style-type: none"> <li>• Unique and descriptive label to match visual information (can include widget, etc. to provide supplemental information)</li> <li>• Will always say “Radio Button” and will always say “Checked” or “Not Checked” to indicate state</li> <li>• “Disabled” will be announced if not enabled</li> <li>• If there is a radio group: unique and descriptive name that matches visual information (label) for the group – announced prior in focus order from radio buttons</li> </ul>
<b>Form Field: Text Field ( blank, with placeholder or inserted text)</b>	<ul style="list-style-type: none"> <li>• Unique and descriptive label that matches visual information and instructions</li> <li>• State should be indicated</li> <li>• If the element has a variable status, the current status must be announced.</li> <li>• Hidden form fields should not be announced</li> <li>• Disabled form fields do not require a form name to be announced</li> <li>• Read-only form fields cannot be edited by the user, but sufficient element data must be read aloud</li> </ul>	<ul style="list-style-type: none"> <li>• “Edit Box”</li> <li>• Unique and descriptive label that matches visual information and instructions or placeholder text</li> <li>• For editing, will announce “is editing”, and any placeholder text or text that user has typed in along with an initial note on where cursor is located within text box</li> <li>• “Disabled” will be announced if not enabled</li> <li>• Character limit must be announced if available</li> </ul>
<b>Grid Layout/View (Android only)</b>	<ul style="list-style-type: none"> <li>• Unique and descriptive name that matches visual information</li> <li>• “Grid view”</li> <li>• Cell contents</li> </ul>	<ul style="list-style-type: none"> <li>• Unique and descriptive name that matches visual information</li> <li>• “Grid view”, or “Grid showing X items”</li> <li>• Cell contents</li> </ul>

Element Type	What to Listen For	Android Specific Output
<b>Heading (text)-if a heading is noted visually, must be noted programmatically</b>	<ul style="list-style-type: none"> <li>• Unique and descriptive label that matches visual information</li> <li>• “Heading” should be read</li> <li>• Supplemental information such as Heading Level number should be set if there are several heading levels.</li> <li>• Headers should be used logically (e.g. not skipping from Heading 3 to Heading 1 to Heading 5 if visual structure does not support that order)</li> <li>• Supplemental information read aloud will not inhibit compliance unless it is confusing for the user</li> </ul>	<ul style="list-style-type: none"> <li>• Unique and descriptive label that matches visual information</li> <li>• “Heading”</li> <li>• Supplemental information such as Aria Landmarks might be read aloud such as “Main”</li> <li>• Android recently updated their API to allow coding of headings but heading and heading level may be noncompliant if the app has not been re-coded accordingly</li> </ul>
<b>Hints</b>	<ul style="list-style-type: none"> <li>• If a hint exists, ensure that sufficient and <i>supplemental</i> information (only) is provided</li> <li>• The hint should not contain necessary information for user interaction</li> </ul>	<ul style="list-style-type: none"> <li>• The hint property provides an accessible name, or label, for input fields</li> <li>• It is only announced for input fields when the field is not populated</li> <li>• It is not recommend using the hint property as the label for a field since it will no longer be announced when the user fills in the field</li> </ul>
<b>Image</b>	<ul style="list-style-type: none"> <li>• Unique and descriptive name that matches visual information</li> <li>• “Image”</li> <li>• If it is a decorative image that has no meaning, nothing should be announced; images must have consistent meaning</li> </ul>	<ul style="list-style-type: none"> <li>• Unique and descriptive name that matches visual information</li> <li>• “Image”</li> </ul>
<b>Link</b>	<ul style="list-style-type: none"> <li>• Unique and descriptive title to match visual information</li> <li>• “Link”</li> <li>• Supplemental information read aloud will not inhibit compliance unless it is confusing for the user. (In Android, screen reader voice changes to note a link)</li> </ul>	<ul style="list-style-type: none"> <li>• Unique and descriptive title to match visual information in robot voice</li> <li>• Android does not differentiate visited vs non-visited link.</li> </ul>
<b>List</b>	<ul style="list-style-type: none"> <li>• Unique and descriptive title to match visual information</li> <li>• Will note number of items in list</li> <li>• If the element has a variable status, the current status must be announced</li> </ul>	<ul style="list-style-type: none"> <li>• Unique and descriptive title to match visual information</li> <li>• Will note if list is a “Dropdown” number of items in list (“List showing X items”) or note “Showing items X of X”</li> <li>• In Chrome: List start, end, and item number will not be read aloud</li> <li>• In Firefox, “First Item” and “Last Item” will be read aloud</li> </ul>

Element Type	What to Listen For	Android Specific Output
<b>Page Control</b>	<ul style="list-style-type: none"> <li>• Current page number out of total number available will be announced or equivalent accessible and unique name to identify page</li> </ul>	<ul style="list-style-type: none"> <li>• Current page number out of total number available will be announced or equivalent accessible and unique name to identify page</li> </ul>
<b>Page Titles</b>	<ul style="list-style-type: none"> <li>• We currently do not test for page titles</li> </ul>	<ul style="list-style-type: none"> <li>• N/A</li> </ul>
<b>Picker Item</b>	<ul style="list-style-type: none"> <li>• Unique and descriptive name that matches visual information</li> </ul>	<ul style="list-style-type: none"> <li>• Unique and descriptive speech output to match visual information</li> <li>• Hint will not be picked up by TalkBack</li> </ul>
<b>Search Bar/Field</b>	<ul style="list-style-type: none"> <li>• Unique and descriptive name that matches visual information</li> </ul>	<ul style="list-style-type: none"> <li>• Unique and descriptive name that matches visual labels and instructions are provided prior to the search button control in the focus order</li> <li>• “Search” for the button</li> <li>• “Edit box” for the text entry field</li> <li>• Will read the contents of the text entry field</li> </ul>
<b>Seek Bar/Control</b>	<ul style="list-style-type: none"> <li>• Unique and descriptive name that matches visual information</li> </ul>	<ul style="list-style-type: none"> <li>• Unique and descriptive label to match visual information</li> <li>• “Seek Control”</li> <li>• Always says the % value</li> </ul>
<b>Skip Link Navigation</b>	<ul style="list-style-type: none"> <li>• Descriptive name to signify skip link navigation after landing on screen</li> <li>• This should read similar to a link.</li> </ul>	<ul style="list-style-type: none"> <li>• “Skip to main content”</li> </ul>
<b>Slider (example: volume, video track position)</b>	<ul style="list-style-type: none"> <li>• Unique and descriptive name that matches visual information</li> </ul>	<ul style="list-style-type: none"> <li>• Unique and descriptive name that matches visual information</li> <li>• “Adjustable”</li> <li>• Current state of the slider</li> <li>• “Disabled” if disabled</li> </ul>
<b>Static Text</b>	<ul style="list-style-type: none"> <li>• Unique and descriptive name that matches visual information</li> </ul>	<ul style="list-style-type: none"> <li>• Unique and descriptive name that matches visual information only (static text will ONLY announce the words and will not announce any function)</li> </ul>
<b>Status Bar Item</b>	<ul style="list-style-type: none"> <li>• Unique and descriptive name that matches visual information</li> </ul>	<ul style="list-style-type: none"> <li>• Unique and descriptive title to match visual information</li> </ul>
<b>Switch/Toggle</b>	<ul style="list-style-type: none"> <li>• Unique and descriptive name that matches visual information</li> </ul>	<ul style="list-style-type: none"> <li>• Unique and descriptive label to match text</li> <li>• “Switch”</li> <li>• “Disabled” if button disabled</li> </ul>

Element Type	What to Listen For	Android Specific Output
<b>Tab</b>	<ul style="list-style-type: none"> <li>• Unique and descriptive name that matches visual information</li> <li>• “Tab”</li> </ul>	<ul style="list-style-type: none"> <li>• Unique and descriptive name that matches visual information</li> <li>• “Tab”</li> </ul>
<b>Table</b>	<ul style="list-style-type: none"> <li>• Unique and descriptive name that matches visual information (to include, “table”, “header” for header column/row)</li> <li>• Read order must be correct and cell association must be clear</li> <li>• For web: Chrome’s GridView might read header and footer, but not column headers. Firefox should be used in this case</li> </ul>	<ul style="list-style-type: none"> <li>• Unique and descriptive name that matches visual information</li> <li>• “GridView” for native, row contents</li> <li>• In Chrome: “header” might not be announced</li> <li>• Android recently updated their API to allow coding of headers but headers may be noncompliant if the app has not been re-coded accordingly</li> </ul>
<b>Text</b>	<ul style="list-style-type: none"> <li>• Unique and descriptive output that matches visual information</li> <li>• Supplemental information read aloud will not inhibit compliance unless it is confusing for the user</li> </ul>	<ul style="list-style-type: none"> <li>• Unique and descriptive output that matches visual information</li> </ul>
<b>Toolbar</b>	<ul style="list-style-type: none"> <li>• Unique descriptive name for toolbar</li> <li>• “Toolbar”</li> <li>• “Dimmed/disabled” if disabled</li> </ul>	<ul style="list-style-type: none"> <li>• Unique descriptive name for toolbar itself may be announced first</li> <li>• “Disabled” if disabled</li> </ul>
<b>Video/Video View</b>	<ul style="list-style-type: none"> <li>• Unique and descriptive name that matches visual information</li> </ul>	<ul style="list-style-type: none"> <li>• “Video” or “Video view”</li> </ul>

## ANDROID KEYSTROKES AND GESTURES

### HOW TO TAKE A SCREENSHOT

Press the Power + Volume Down button **OR** use the camera key on your external keyboard (if your external keyboard has a button).

### ANDROID KEYSTROKES **WITHOUT** TALKBACK ENABLED

Keystroke	Description
Alt + Up Arrow	Moves to the beginning of a text field or top of list box
Alt + Down Arrow	Moves to end of a text field or bottom of list box
Tab	Moves to the next interactive control in the focus order.
Up	Move to the control above the current control
Down	Move to the control below the current control
Left	Move to the control to the left of the current control
Right	Move to the control to the right of the current control
Enter	Activates the current control
Ctrl + X	Cut
Ctrl + C	Copy
Ctrl + V	Paste
Shift + Arrows	Selects text

ANDROID KEYSTROKES **WITH** TALKBACK ENABLED

Keystroke	Description
Alt + Shift + G	Open global context menu
Alt + Shift + L	Open local context menu:
Alt + Shift + Z	Pause or resume TalkBack
Alt + Shift + Right Arrow	Move to next item
Alt + Shift + Left Arrow	Move to previous item
Alt + Control + Enter	Read from top
Alt + Shift + Up Arrow	Move to first item
Alt + Shift + Down Arrow	Move to last item
Alt + Shift + Enter	Activates the current control
Alt + Shift + H	Goes to home screen
Delete	Deletes text
Alt + Shift + Backspace	Back
Tab	Moves to the next interactive control in the focus order.
Shift + Tab	Moves to the previous interactive control in the focus order.

Keystroke	Description
Alt + Shift + Ctrl + Right Arrow	Move to next word
Alt + Shift + Ctrl + Left Arrow	Move to previous word
Alt + Shift+ Right Arrow	Move to next character
Alt + Shift+ Left Arrow	Move to previous character
Shift + Arrows/Ctrl + A	Selects text

#### TALKBACK GESTURES

Gesture	Description
Drag one finger	Explore the screen and hear audible feedback for what is being touched.
Drag two fingers on text input field (when blue arrow is shown)	To move character by character
Two fingers press and hold on screen	Long press to open additional menus such as Copy/Paste/Cut/Select All/Assist
Double-tap anywhere on the screen	Open or activate the item last touched.
Swipe Up or Down using two fingers	Scroll lists and pages.
Swipe Left or Right using two fingers	Change pages and screens.

Gesture	Description
Swipe Right (or Down) using one finger	Move to the next item
Swipe Left (or Up) using one finger	Move to the previous item.
Swipe Down then Up using a single motion	Transition to the next reading level when reading blocks of text, then swipe right to read forward or left to read backward.
Swipe Up then Down using a single motion	Transition to the previous reading level when reading blocks of text, then swipe right to read forward or left to read backward.
Swipe Right then Left using a single motion	Move to the next piece of information.
Swipe Left then Right using a single motion	Move to the previous piece of information.
Swipe Up then Right	Open local context menu
Swipe Up then Left	Home button
Swipe Down then Right	Open global context menu
Swipe Down then Left	Back button
Swipe Right then Down	Open notifications
Swipe Left then Up	Recent apps button
On lock screen, press and hold two fingers until a sound plays	Turn on TalkBack from lock screen

MAGNIFICATION GESTURES

Gesture	Description
Triple tap single finger, hold, and move finger on screen	Temporarily magnify the screen – release to de-magnify
Triple tap single finger	Magnify screen
Pinch in/out	Change magnification level

ADDITIONAL GESTURE AND KEYBOARD COMMAND RESOURCES FOR ANDROID:

*Explore by Touch* Tutorial (within Android Accessibility Settings)

List of all gesture and keyboard commands: <https://support.google.com/accessibility/android#topic=6007234>

DEVELOPER RESOURCES:

<https://developer.android.com/guide/topics/ui/accessibility/index.html>

**DOCUMENT CONTENT CHANGE LOG**

**VERSION 1.0, SEPTEMBER 2017**

Initial version.