Accessibility tester

Welcome to accessibility testing

Discover how to make React Native components accessible to all users. Through practical examples and guided testing, learn essential techniques for creating inclusive applications.

Start testing

Other modes:

Dark mode

Components

Select a category to test:

Text & headers
Test basic text components, headings, and content hierarchy

Interactive elements
Explore buttons, links, and touch targets

Form components
Test inputs, selections, and form controls

Complex interactions
Try modals, gestures, and advanced patterns

Screen readers

Another important component in accessibility is the usage of screen readers.

TalkBack (Android)
Basic gestures and navigation

VoiceOver (iOS)
Essential commands and features

1st

Enable screen readers

Gestures

Here, different actions will be tested. Learn and test common gesture patterns.

```
1. Basic Taps
   "Tap the box below to continue"
   [Interactive Box]
   Progress: 1/4 gestures

2. Long Press
   "Press and hold the button"
   [Hold Area]
   Timer shows duration
```

```
3. Swipe Actions
   "Swipe the card left or right"
   [Swipeable Card]
   Arrows show direction

4. Custom Gestures
   "Double tap and hold"
   [Practice Area]
   Visual feedback provided
```

Snackbar giving succesful feedback

Accessibility docs

Screen readers
We offer basic support and information for voice readers like VoiceOver and TalkBack. Enable these features in your device

Text scaling and headings
Support dynamic scaling based on user preferences..

Color contrast
Learn how to maintain compliant contrast ratios throughout the app.

Touch targets
Size appropriately the elements for easier interaction.

Contrast checker

Contrast checker

Test the contrast of your components in real-time (Link to WCAG contrasts)

Foreground
Hex Value
#0000FF
Color Picker
Alpha
1
Lightness

Background
Hex Value
#FFFFFF
Color Picker
Lightness

Contrast Ratio
8.59:1

Frameworks

React Native | Flutter

Brief description of framework's accessibility approach

Accessibility Tree
Subhead

Semantic Properties
Subhead

Screen reader support
Subhead

```
[Key Features]
(Expandable cards)
1. Accessibility Tree
   - Detailed explanation
   - Implementation notes
   - Code example

2. Screen Reader Support
   - Platform specifics
   - Configuration
   - Testing tips

3. Semantic Properties
   - Available properties
   - Usage examples
   - Best practices
```

```
←    Framework Comparison

[Feature Grid]
Feature   | React Native | Flutter
--------------------------------
Setup     | ✓ Simple     | ⚠ Complex
Props     | Direct       | Wrapped
Testing   | Native       | Built-in
```

Select time

07 : 00 AM PM

Cancel OK

Screen readers

Name
Value
Email
Value
Password
Value
Gender
Value

Date picker

Label Label Label Label Label Label

Framework chosen

Accet terms

Select date

Enter date

Date
mm/dd/yyyy

Cancel OK

Basic dialog title

A dialog is a type of modal window that appears in front of app content to provide critical information, or prompt for a decision to be made.

Action 2 Action 1