



Fact sheet Accessibility Web

E-Accessibility Solutions for Everyone

Table of contents

Text content

Non-text content

Colors and contrasts

Keyboard navigation

General navigation

Layout

ARIA attributes

Toolbox



Orange guidelines

<https://a11y-guidelines.orange.com/en/>

Text content

Define a title for each page

Each page should have a title introducing its content.

```
<title> Page title - website name </title>
```

Using headings

When headings are clear and descriptive, users can find the information they seek more easily.

Use heading tags **<h1>** to **<h6>**.

```
<h1> Home - Orange </h1>
```

```
<h2> The news </h2>
```

Indicate the main language of the page

```
<html lang="en">
```

Indicate language changes

```
Discover Orange <span lang="fr">café</span>
```

Respect HTML semantics

- Elements must have start and end tags.
- Links are indeed links and refer to a page.
- Buttons are indeed buttons, they trigger an action, a validation or allow a choice (**<button>** or **<input type="button">**).
- If necessary, use HTML lists (****, **** et ****). No empty list or containing only one item.

Data tables

- Do not use tables for layout purposes.
- Use tags **<table>**, **<tr>**, **<th>** et **<td>**.
- For complex tables, use the **scope** attribute (**row/col**) and the **headers** and **id** attributes.
- Add a summary using **<caption>** tag.
- Validate the HTML code of the pages using the W3C validator : <https://validator.w3.org/>

Associate text labels with form controls

Using **label** tag, **for** and **id** attributes.

```
<label for="email">E-mail address</label>
```

```
<input id="email" type="text">
```

Using **title** and **aria-label** attributes:

```
<input type="text" title="Search Orange.com" aria-label="Search Orange.com">
```

Non-text content

Provide text alternative for images

For informative images:

```

```

For decorative images:

```

```

Provide transcription for audio/video tracks

Provide a text transcript within the page or through a link to a text file.

Colors and contrasts

Check color contrast

- 4.5/1 min. for standard size text.
- 3/1 min. for large text (150% normal text or 120% if bold).

You can use « Colour Contrast Analyser » app:

<https://www.tpgi.com/color-contrast-checker/>

Don't rely on color

Color is not used as the only visual means of conveying information.



Keyboard navigation

Allow keyboard navigation

Check that all the important actions performed with the mouse can also be done with the keyboard, focus is visible (CSS **outline**) and focusable components receive focus in an order that preserves meaning and operability.

- TAB/Shift+TAB: move the focus (forward/backward)
- Enter: activate the focused element (link, button...)
- Space bar: check/uncheck a checkbox.
- Directional arrows : to modify the selection of the radio buttons, to move in a drop-down list, or to scroll through the scrollbar when they are present).

General navigation

Provide descriptive links

Links and buttons must be understandable out of context. In order to make them more explicit, they can be completed with a `title` attribute.

Visually hidden

Elements can be vocalized by screen readers without being displayed. Use the `aria-label`, or use an accessible CSS class for hiding (for example: "`visually-hidden`" class provided by the Bootstrap or Boosted frameworks).

```
<a href="..">send
  <span class="visually-hidden"> the invoice</span>
</a>
```

You can also hide the content displayed on the screen for screen readers, it will no longer be vocalized. To do this, use the `aria-hidden="true"` attribute.

Warn the user when opening a new window

Notify the user of any new window opening by mentioning it in the text of the link or using `aria-label` and `title` attributes:

```
<a href=".." title="See terms of use (new window)"
  aria-label=" See terms of use (new window)">See
  terms of use</a>
```

Provide skip links

Set up skip links allowing access to the main areas of the page.

Allow control of animations

Any moving, updating, flashing content must be able to be stopped, hidden or paused.

Layout

Use relative sizes (Responsive Design)

Give preference to relative units (em, rem, %...) and make the arrangement of elements fluid. So that the user can zoom in and resize text up to 200% with no loss of information.

ARIA attributs

Aria attributes are used to improve the vocalization for screen reader users. They have no effect for other users.

Global attributes

aria-label, aria-labelledby

Adds a label to the element :

```
<button aria-label="search">...</button>
<h2 id="search">Search the catalog</h2>...
<button aria-labelledby="search"></button>
```

aria-describedby

Adds a description to the element (vocalized after the label):

```
<label for="num">serial number</label>
<input id="num" aria-describedby="desc" type="text">
<span id="desc">1 letter followed by 4 digits </span>
```

aria-hidden

Hides the element and its children from screen readers:

```
<button>
  <span aria-hidden="true" class="icon">U</span>
  Search
</button>
```

aria-haspopup

Indicates the presence of a drop-down menu:

```
<button aria-haspopup="true">Options menu</button>
```

aria-expanded

Indicates if the element is expanded (menu, section, etc.):

```
<button aria-expanded="true">Menu</button>...
```

aria-live

Tells the screen reader that this element should be spoken automatically when it is updated:

```
<span aria-live="polite">Progress 50%</span>
```

disabled (HTML5)

Indicates that the element is disabled:

```
<button disabled>Save</button>
```

Forms

required (HTML5) : indicates that the element is required

```
<input type="text" id="email" required>
```

aria-invalid : indicates an input error

```
<input type="text" id="email" aria-invalid="true">
```

Roles

dialog : indicates the presence of a dialog

```
<div role="dialog" aria-labelledby="title">
  <div id="title">...</div>[...]
</div>
```

alert and alertdialog : tells the screen reader to announce an important message.

```
<p role="alert">You must accept the TOS</p>
```

Toolbox

Screen readers

To test your websites:

- On Windows : **arrow**
- On MacOS and IOS: **VoiceOver**
- On Android: **TalkBack**

Axe Dev Tools

Extensions available for Chrome and Firefox to perform automatic accessibility tests.

<http://bit.ly/aXe-Firefox> and <http://bit.ly/aXe-Chrome>

Orange Boosted with Bootstrap

Orange Boosted (HTML, CSS and Javascript framework), based on Bootstrap. Make responsive web sites, accessible and compliant with the Orange brand.

<http://boosted.orange.com>

La Va11ydette

To help you check that your site meets accessibility criteria : <https://la-va11ydette.orange.com/?lang=en>

Orange Digital Accessibility

Guidelines, tools and good practices on accessibility, offered by EASE.

<https://a11y-guidelines.orange.com/en/>