Accessibility

Accessibility for the web refers to designing your websites and applications so that individuals with disabilities are able to use them effectively. Disabilities can come in many forms including visual, physical, auditory, etc. It is important to design your applications to be accessible, both to reach as wide an audience as possible and for potential legal reasons. Making your application accessible improves the user experience for individuals without disabilities as well.

> *For international products it is important to research what level of compliance is required by law — accessibility requirements vary from one country to another.*

The best practice guidelines used internationally for making websites accessible are the Web Content Accessibility Guidelines (WCAG) 2.0, released by the World Wide Web Consortium (W3C) in 2008.

Four Principles of Accessibility

There are four principles that lay the foundation necessary for anyone to access and use web content — Perceivable, Operable, Understandable, and Robust.

Perceivable

Information and user interface components must be presented to users in ways that they can perceive. For example, a button should look like a button and not be hidden behind an image or other hard-to-distinguish elements.

[Design System Image]

Operable

Operable means that users must be able to perform the actions required to interact with interface elements. This means that all functionality should be available via the keyboard (including navigation), temporary content should be displayed long enough to be read by individuals with visual impairments, etc.

Understandable

Users must be able to consume information and interact with the application easily. This includes using appropriate language (avoiding abbreviations, writing at an appropriate reading level, etc.), making sure that interface elements behave in consistent and predictable ways, providing helpful error messages and instructions, etc.

For example, accompanying a button with a text label makes it more understandable if the meaning of the icon on its own is unclear.

[Design System Image]

Robust

Web content must be robust enough that it can be interpreted reliably by a wide variety of users, including those who rely on assistive technologies (e.g., screen readers, text-to-speech software, alternative pointing devices, etc.). As assistive technologies advance, your applications will need to account for these advances.

Color

[Design System Image]

Brightlayer UI colors and components have been tested to meet minimum accessibility requirements. By using these default styles, you should be able to reduce the effort required for your product to pass accessibility tests.

If your project requires the use of additional styles or colors, you should adhere to the following guidelines:

- Use Brightlayer UI colors in the 500+ range for white backgrounds.
- Avoid using yellow, red, and orange for text (except for single-line error messages on forms)
- Do not rely on colors for instructions (e.g., do not say "Click the red button to close")

Most Brightlayer UI text elements use the Black 500 color, with some headlines using Blue 500. Button labels should use White 50 for dark background and Black 500 on light backgrounds. You can check the contrast ratio to determine which option provides greater contrast.

[MaterialDesignDescription Component - Interactive React component]

} />

[MaterialDesignDescription Component - Interactive React component]

} />

[MaterialDesignDescription Component - Interactive React component]

} />

Checking Contrast

Low contrast makes it difficult for all users to view your product, especially for individuals with visual impairments. You can test your contrast ratios using free online tools and Figma plugins.

Text

In addition to making sure that the colors used for text are accessible, you should also ensure that text can be made larger without affecting the content or function of the page. Users with visual impairments may need to increase the font size of applications in order to read more clearly.

Most browsers offer a feature for you to change the default font size (larger or smaller). You should test your application across multiple sizes to make sure that everything still looks right and functions correctly.

Most Brightlayer UI components are designed to accommodate bigger font sizes when required.

Images

Any images in your application should be accompanied by a caption or other descriptive text, unless they are purely for decoration. Decorative images should show a brief description on hover. Images should always include an "alt" property so that they are accessible to users utilizing a screen-reader or other assistive technologies.

- Images should include descriptive text/alt text in the markup/code.
- Complex images should have more detailed descriptions near the image
- Videos must provide visual access to the audio information through in-sync captioning

---

Testing Tools For Web

The tools used in the Trusted Tester Test Process (and ICT Testing Baseline) have been chosen based on several factors including ease of use, ease of teaching, and accuracy of results. They are also free to install and use.

- Accessible Name & Description Inspector (ANDI)
- Color Contrast Analyzer for Mac and Windows (CCA)