# Accessibility Testing

## **What is accessibility testing?**

Accessibility Testing is a subset of usability testing, and it is performed to ensure that the application being tested is usable by people with disabilities like hearing, color blindness, old age and other disadvantaged groups.

People with disabilities use assistive technology which helps them in operating a software product.

* **Speech Recognition Software -** It will convert the spoken word to text, which serves as input to the computer.
* **Screen reader software** - Used to read out the text that is displayed on the screen
* **Screen Magnification Software**- Used to enlarge the monitor and make reading easy for vision-impaired users.
* **Special keyboard-** made for the users for easy typing who have motor control difficulties

## **Why accessibility testing?**

**Reason 1**: Cater to market for Disabled People.

About 20% of the population has disability issues.

* 1 in 10 people have a sever disability
* 1 in 2 people over 65 have reduced capabilities

Disabilities include blindness, deaf, handicapped, or any disorders in the body.

A software product can cater to this big market, if it's made disabled friendly. Accessibility issues in software can be resolved if Accessibility Testing is made part of normal testing life cycle.

**Reason 2**: Abide by Accessibility Legislations

Government agencies all over the world have come out with legalizations, which requires that IT products to be accessible by disabled people.

Following are the legal acts by various governments -

* United States: Americans with Disabilities Act - 1990
* United Kingdom: Disability Discrimination Act - 1995
* Australia: Disability Discrimination Act - 1992
* Ireland : Disability Act of 2005

Accessibility Testing is important to ensure legal compliance.

**Reason 3**: Avoid Potential Law Suits

In the past,Fortune 500 companies have been sued because their products were not disabled friendly. Here a few prominent cases

* National Federation for the Blind (NFB) vs Amazon (2007)
* Sexton and NFB vs Target (2007)
* NFB Vs AOL settlement (1999)

It's best to create products which support disabled and avoid potential lawsuits.

## **Which disabilities to support?**

Application must support people with disabilities like -

|  |  |
| --- | --- |
| **Type of Disability** | **Disability Description** |
| **Vision Disability** | * Complete Blindness or Color Blindness or Poor Vision * Visual problems like visual strobe and flashing effect problems |
| **Physical Disability** | * Not able to use the mouse or keyboard with one hand. * Poor motor skills like hand movements and muscle slowness |
| **Cognitive disability** | * Learning Difficulties or Poor Memory or not able to understand more complex scenarios |
| **Literacy Disability** | * Reading Problems |
| **Hearing Disability** | * Auditory problems like deafness and hearing impairments * Cannot able to hear or not able to hear clearly |

# How to do accessibility testing?

Accessibility Testing can be performed in 2 ways, and they are:

1. Manual
2. Automated

## **Categories for Testing**

* Keyboard Navigation
* Non-Visual Navigation with a screen reader
* Alternate Visual Access
* Usability
* Non-Audio Access
* STEM Content
* Downloadable Files

## **Web Accessibility Quick Test Checklist**

The following tests can be used to quickly and easily check the accessibility of web pages and PDF documents.

|  |  |
| --- | --- |
| Test | Summary |
| Coding | |
| Valid HTML | Check HTML validation |
| Headings | Use clear headings and labels |
| Text & Colors | |
| Large Fonts | Text can be resized to 200% without loss of content or function |
| High Contrast | Contrast ratio between text and background is at least 7:1 |
| Images & Multimedia | |
| Alternate Text | Provide text alternatives for non-text content |
| Captions | Provide captions for videos with audio |
| [Sign Language (Pre-recorded)](http://www.wuhcag.com/sign-language-pre-recorded/) | Provide sign language translations for videos |
| Links & Forms | |
| Skip Navigation | Provide a ‘Skip to Content’ link |
| Tab Order & Link Text | Logical order & Every link’s purpose is clear from its text |
| Form Labels | Label elements and give instructions |
| Scripts & Objects | |
| Keyboard Operation | Accessible by Keyboard only |
| Downloadable Documents | |
| PDF Documents | Save Document as text |
| PDF Forms | It can be operated using the keyboard. |

## **Testing Tools Needed**

* Keyboard
* Web Accessibility Analyzer tools such as “WAVE toolbar” or Browser extension
* Screen Reader (example: NVDA, Jaws)
* Color Contrast Analyzer

## **Testing**

Automated testing such as using the WAVE Tool only captures about 30-50% of accessibility issues.

So accessibility testing need both manual & automate testing to find out more issue

### **Testing Keyboard Navigation**

The first step to manual testing is to throw away mouse. Navigate the page, including all menus and interactive elements, using only the keyboard.

**Keyboard Focus:**

Use the TAB, ENTER, SPACE, UP & DOWN ARROW keys to navigate

**TAB Order:**

1. TAB through the page to see if the order is logical.

2.Activate all interactive elements with the ENTER or SPACE keys such as POP UP, Dropdown Menu, Dialog Box, Multimedia

### **Testing Alternate Visual Access**

Many users are colorblind or have low vision. Color choice matters for both groups of users. Contrast impacts those with low vision, while those who are colorblind cannot distinguish between certain colors.

**Choice Of color:**

Use Color Contrast analyzer tools such as “Chrome WCAG 2.0 Color Contrast Analyzer” to indicate sufficient contrast for text on top of image

**Font Size:**

Font size increase when zoom in page

### **Testing Usability**

Many users have a non-apparent disability, such as a traumatic brain injury or a cognitive or learning disability that affects their ability to process information. Clear organization of information is vital for these users.

**Clear and Simple Content:**

* + Fonts are basic, legible, easy to read
  + There is plenty of white space

**Logic in Visual Navigation:**

* + Visual navigation is logical

### **Testing Non-Audio Access:**

This includes audio files such as podcasts as well as videos.

* + All Videos have Caption
  + Caption has sufficient contrast
  + All files have a link to transcript

### **Testing STEM (Science, Technology, Engineering, Mathematics) Content**

* + Screen Reader recognizes and reads the symbols

### **Testing Downloadable Files:**

Documents that are linked for download from a site also need to be accessible.

**PDF Documents:**

Open PDF documents in Adobe Reader and save as "accessible text" (File: Save as Text). Open the text file and confirm that all text is present and in the right order.

**PDF Forms**:

Hover the mouse pointer over each form field and check that an appropriate label appears as a tool tip. Tab to each field and check that it can be operated using the keyboard.

### **Testing Non-Visual Navigation with a Screen Reader**

Using NVDA to Evaluate Web Accessibility

#### Introduction

It is important to evaluate the accessibility of web content with a screen reader, but screen readers can be very complicated programs for the occasional user, so many people avoid them. This doesn't need to be the case. While screen readers are complicated, it is possible to test web content for accessibility without being a "power user."

#### Getting Started

NVDA is open-source software. **Download NVDA** for free (Windows only). Since NVDA is a relatively new project, some of its capabilities are still basic. After download and installation NVDA start running the program by holding Ctrl + Alt + N.

The **NVDA** key is set to the Insert key by default, but it can be changed to the Caps lock key when installing NVDA for the first time. To change NVDA key preference later, press Ctrl + NVDA + K.

* While NVDA has early support for accessing Windows and many Windows applications, we will be focusing on accessing web content only.
* Make sure that NumLock is off.
* Remember that screen reader users typically do not use a mouse. As you become more comfortable with NVDA, try using only the keyboard.
* Most browser shortcut keys will work when using NVDA.
* The page *may not* scroll while reading, so may hear content being read by NVDA that isn't visible on the screen.

#### Reading

There are dozens of keyboard shortcuts that allow to read content by line, sentence, word, character, etc. The following is a list of essential reading shortcuts. With these shortcuts, should be able to navigate through most content.

* **Numpad +**: Start reading at the top of the page
* NVDA + ↓: Start reading at the current position
* Ctrl: Stop Reading
* NVDA + ↑ or **Numpad 8**: Current line
* Ctrl + ←/→ or **Numpad 4**/**Numpad 6**: Previous/next word
* ↑ or **Numpad 7**: Prior line
* ↓ or **Numpad 9**: Next line
* ←/→ or **Numpad 1**/**Numpad 3**: Previous/Next character
* F5/Ctrl + F5: Page refresh / hard page refresh. If you get lost, this is how you can start over.
* NVDA + Ctrl + ↑/↓: Rate of speech increase/decrease

#### Navigation

Sighted users visually navigate through web content in a number of ways. They skim for headings, lists, tables, etc. Most of these methods are available to screen reader users if the site is correctly structured and well organized. To navigate forward and backward through links and form elements on the page, use Tab and Shift + Tab. Other shortcuts are listed below.

##### **Quick Keys**

The following Quick Keys will help to navigate common page elements.

* H: Headings
* D: Landmarks
* 1 - 6: Headings level 1-6
* F: Forms
* T: Tables
* B: Buttons
* L: Lists
* I: Items in a list

Shift + Quick Key will allow to navigate through elements in reverse order (works with most Quick Keys). Several other Quick Keys are available in list of NVDA keyboard shortcuts.

##### **Other navigation shortcuts**

* NVDA + F7: Elements List - lists page links, headings, and landmarks
* Ctrl + Home: Top of the page
* Ctrl + End: Bottom of the Page
* Alt + D or F6: Browser address bar

#### Links Descriptive Text

One of the ways a screen reader user navigates a page is by using the TAB key to skip through the links on the page, or by bringing up a list of links. Links should be descriptive so that a user can easily determine where they will go even when taken out of context.

**Checklist:**

* + Links are descriptive, without generic text such as “click here”.
  + The texts of links are intuitive when listed alphabetically.

#### Logical Heading

Screen reader users also navigate content using headings. Headings allow them to jump to a specific section instead of having to listen to the entire page.

**Checklist:**

* + Headings are used logically to provide structure and indicate importance of the content.
  + Pages can be navigated using headings.

#### Skip Navigation Link

A skip navigation link is helpful for users who don’t want to work their way through all the links on the page before getting to the main content. This link does not have to be visible on the page, but it should be listed first in the tab order.

**Checklist:**

* + Skip navigation links are present.

#### Image

Any non-text element needs a text description that screen reading software can read aloud to the listener. It should describe the purpose of the image in context.

**Checklist:**

* + All meaningful images have descriptive alternate text.
  + All navigation buttons have descriptive alternate text.

#### Multimedia

Includes features such as videos, audio files, calendars, Flash content, and photo carousels.

**Checklist:**

* + Video or audio do not start to play automatically, only when triggered by the user.
  + Videos have audio descriptions available for any information that is only conveyed visually.

#### Forms

**Checklist:**

* + Form fields have descriptive labels.
  + Appropriate descriptive labels are present and read by screen reader when tabbing through form fields.
  + Buttons are labeled and read correctly by screen reader.
  + When an error dialog is dismissed, screen reader automatically returns to the empty field.

#### Data Tables

**Checklist:**

* + Data tables have designated header and/or column rows.
  + Tables have captions (short text descriptions).
  + Tables are not nested or filled with spanned or ‘spacer’ cells.

#### Frames

**Checklist:**

* + Each frame has a descriptive title attribute value.
  + When user refresh a page, it stays on the current frame.
  + While listening with screen reader, user can navigate between frames

#### CAPTCHA

If CAPTCHA is used, it must be fully accessible and simple to use.

**Checklist:**

* + CAPTCHA is fully accessible by keyboard.
  + CAPTCHA is fully accessible to screen reading software.
  + Audio CAPTCHA is fully accessible by screen readers, including a pause that allows the screen reader to finish before the audio begins.