Note this document is OLD .

the new version is at

https://raw.githack.com/w3c/coga/design-doc-transfer/design/index.html

You can also add comments to:
https://docs.google.com/document/d/1HG5I-nkIRkDPXaDfcQ9v7q4SgAkYbAen1rV5cSlZWls/edit?usp=sharing

Working on design patterns

These are the design patterns that we are currently working on
For the full document see https://w3c.github.io/coga/design/
In most cases a link is provided in the section to detailed information that will be useful.

Design patterns review process (first review)

See the Review checklist for full details

- Author selects a reviewer, and adds name in Review column and notifies them
- Reviewer makes suggestions
- Update Review column below to **Questions** if waiting for response from author
- Author and reviewer : accepts changes
- Reviewer marks as **complete** in Review Column

Tracking Table

Guide to the Draft Status column

- **<Name>** means your are working on it in this document
- **Complete** means in this document and awaiting review
- **Unchanged** means not reworked in this document
- **Missing** means has an Editor's Note but not found in this document

Guide to the Review column

- **<Name>** means your are reviewing

- **<Date>** is the date you plan to have completed your review
- **Questions** means reviewer has questions to be answered by author or Task Force
- **Complete** means review is complete review

Pattern with link to section in editors draft	Draft Status	1 st Review	Xfered
Objective 1: Help users understand what things are and how to use them			
Make the purpose of your page clear Only in this doc as an example	Unchanged		
Make each step clear	Rachael Complete	Jennie Complete	
Headings, visual hierarchies and white space “Use White spacing” has been factored out Nothing on headings and hierarchies	steve		
Call out boxes renamed Use Graphical indicators	Abi Complete	Lisa Complete EA No failures mentioned comments added	
Chunk Media	Steve Complete	EA Complete	
Make the purpose of each section clear	Glenda Steve Complete	**Lisa questions**	
Make it clear what are cohttps://w3c.github.io/coga/design/#make-the-purpose-of-each-section-clearntrols and how they should be used	Unchanged		
Each region and its controls can be clearly recognized	Unchanged		
Toolbars and controls are visible or easy to find Renamed Clear structure and relationships	Jennie Completed	Rachael Complete	
Sub-menu items that are clearly associated with the main menu items under which they fall Merge in 1st part of “Use a Clear navigation”	John K Complete	**Jennie Questions**	
Use a design that the user is likely to recognise and understood	Rachael Complete	Steve Complete	
Be internally consistent	Steve, Jennie &	David Complete	

Changed to Use a consistent visual design Merge in “Make a consistent page structure” (	E.A. Complete		
Use symbols that help the user	EA Complete	Steve Complete	
Objective 2: Help the user find what they need			
Make it easy to find the most important things on the page Merge in “Help users find the content they can use” and “Avoid scroll ”	Jamie	Lisa Complete	
Add search	Lisa complete	Steve Complete	
Avoid scroll merge with Make it easy to find the most important things on the page	Merged		
Always let the user go back	Unchanged		
Objective 3: Use clear and understandable content and text			
Use clear words	Unchanged	EA	
Use a simple tense and written language Was Use a simple tense and voicing	Unchanged	EA	
Do not use double negatives or clauses inside clauses	Unchanged	EA	
Use literal language	Unchanged		
Separate each instruction	Unchanged		
Keep text succinct	Unchanged		
Use clear spacing Renamed Use White spacing	Lisa Completed	David Complete	
Use clean typography and punctuation Renamed Use clear and accurate text formatting and punctuation	**Abi Completed**	David Complete	
Provide summary of long documents	Unchanged		
Use clear images Merged with use symbols	Merged		
Provide Alternatives for numbers	Lisa Complete	Steve Complete	
Objective 4: Prevent the user from making mistakes and make it easy to correct mistakes when they do occur.			

Build forms so that people make less mistakes	Unchanged		x
Make it easy to undo errors	Unchanged		x
Use clear labels and instructions Split into 2 Be flexible with form inputs	**Jamie** **Complete**	**John K**	x
Make it easy for the user to undo actions	Jamie Missing		x
Avoid data loss and "time outs"	Unchanged		x
Provide feedback	**Lisa complete**	**?**	**x**
Tell the users about any fees and charges at the beginning of a task	Lisa- complete	Jennie Complete	
The user knows when the content changes	Missing	Rachael Complete	
Keep the user information safe	Lisa complete	Rachael Complete	
Objective 5: Help the user focus and restore context if attention is lost.			
Avoid interruptions	Unchanged		
Avoid too much content on the page completed but question about being part of another SC - bottom of this page (EA)	EA Complete	Steve Complete	
Objective 6: Processes do not rely on memory			
Logging in does not rely on good memory or other cognitive skills	Unchanged		
Let people avoid navigating voice menus	Unchanged		
Do not rely on users memorizing information	Jamie Missing		
Objective 7: Provide help and support			
Provide help for complex information Remove any content that duplicates content in Objective 3 or 7	Steve Complete	**Abi Complete (email sent to Steve)**	

Provide help with directions	Steve Complete	Abi Questions	
Provide help for forms and non-standard controls Ensure doesn't overlap with "Support personalization of symbols and controls)	Steve Complete	**Abi Complete (email sent to Steve)**	
Provide human help I've removed human help as didn't need changing	Unchanged		
Provide reminders	Lisa complete	Complete Jennie: Questions	
Make it easy to find help and feedback channels Was Feedback is usable by everyone) Merge in other "Make it easy to find help" NB merge object level info as well	Glenda Steve Missing	Lisa I think it was merged….	
Objective 8: Adapt and Personalize			
Support API's Renamed Enable APIs and extensions	Lisa complete	Complete Jennie	
Support simplification	Lisa - completed	J kirkwood complete	
Support personalization of symbols and controls Merge 2nd half of "Use a clear Navigation"	Lisa - completed	J kirkwood complete	

Working on design patterns **1**
Design patterns review process (first review) 1
Tracking Table 1
Template 9
Name 10
How it helps 10
More details (optional) 10
Getting started (optional) 10
Examples 10
Technical details 10
Worked example **11**
Make the purpose of your page clear 11
How it helps 11
More details 11
Examples 11
Technical details 11
Patterns 11
Clear Structure and Relationships. (Jennie) 12
How It Helps 12
More Details 12
Examples 13
Technical details 13
Sub-menu items that are clearly associated with the main menu items under which they fall. (John K) 14
How it helps 14
Getting started 14
Examples 14
Technical details 15
Make it easy to find help and feedback channels (Glenda, Steve 16
How it helps 16
More details 16
Examples 16
NB !The following object level information needs to me merged in to the new objective with the above pattern 18
User Testing: 18
User Stories: 18
Provide Alternatives for numbers (Lisa) 19
How it helps 19

More details 19
Use white spacing (lisa) 21
How it helps 21
More details 21
Example 21
Technical details and history 22
Provide reminders (lisa) 23
How it helps 23
More Details 23
Technical details 24
Tell the users about fees and charges at the beginning of a task (lisa) 26
How it helps 26
Examples 26
Technical details 27
Add search (Lisa) 28
How it helps 28
More details 28
Examples 28
Technical details 28
Changing context 29
How it helps 29
More details 29
Examples 30
Technical details 30
Keep the user information safe (lisa) 31
How it helps 31
More details 31
Examples 31
Technical details 32
Objective 8: Support personalization and adaptation 33
8.1User testing 34
8.2 User stories 34
Support simplification (lisa) 35
How it helps 35
More details 35
Getting started 35
Examples 35
Technical details 36
Support personalization of symbols and controls (lisa) 37

More details 37
How it helps 37
Getting started 38
Examples 38
Technical Details 38
Enable APIs and extensions 39
How it helps 39
More details 40
Getting started 40
Examples 41
Technical details 41
Use clear symbols 42
How it helps 42
More details 42
Getting started 42
Examples 42
Technical details 42
Make Each Step Clear 44
How it helps (Rachael) 44
Examples 44
Technical details 44
Make the purpose of each section clear (JohnR Janina Glenda, Steve) 44
More details 45
Examples 46
Technical details 46
Use Graphical indicators to group and highlight information (Abi) 47
How it helps 47
More details 48
Examples: 48
Chunk Media (steve) 49
How it helps 49
More details 49
Examples 49
Technical details 49
Use clear and accurate text formatting and punctuation (Abi) 51
How it helps 51
Use cases: 51
More details 51
Examples 52

Technical details 53
Use symbols that help the user (EA) 54
How it helps 54
More details 54
Getting started 54
Examples 54
Technical details 55
Form input flexibility from Use clear labels and instructions (Jamie, Michael, a bit of Gareth) 56
How it helps 56
More details 56
Getting started (optional) 56
Examples 56
Technical details 56
Instructions from Use clear labels and instructions (Jamie, Michael, a bit of Gareth) 57
How it helps 57
More details 57
Getting started (optional) 57
Examples 57
Technical details 57
Use a consistent visual design (Jennie, EA, Steve) 58
How it helps 58
More details 58
Getting started 58
Examples 58
Technical details 59
Use a design that the user is likely to recognise and understand (Rachael) 59
How it helps 59
More details 59
Getting started 60
Technical details 60
Provide help for complex information and tasks (steve) 61
How it helps 61
More details 61
Examples 61
Technical details 62
Provide help for forms and non-standard controls (steve) 63
How it helps 63
More details 63

Examples 63
Technical details 64
Provide help with directions (steve) 65
How it helps 65
More details 65
Examples 65
Technical details 65
Avoid too much content on the page (E.A.) 67
How it helps 67
More details 67
Examples 67
Technical details 67
Provide feedback 68
More details 68
Examples 69
Technical details 69

Items for editorial review during HTML transfer 70
Leave for next iteration? 70

ISSUES found during HTML transfer 70

Template

Look below or at https://raw.githack.com/w3c/coga/master/design/index.html#make-the-purpose-of-your-page-clear for a worked example

Name

Add a short text on what to do

How it helps

This section should have

1. An explanation of the problem / user need that the pattern solves
2. an everyday understandable example of how it helps or the problems

More details (optional)

Add annoying details hear, such as exceptions (if necessary)

Getting started (optional)

Add content here if necessary such as what you might recommend as a proposed Success Criteria at the WCAG single A (measurable/testable/easy to achieve)

Examples

(These should make it easier to understand)
At this point we can have one example. (You can add an editor's note if you feel more examples would add clarity for different cases, but do not have time to do it now.)
Success example:
(find from bbc gel)
Failure example:

Technical details

"The following are proposals for WCAG. They experiment with more testable language"
Link to github issue and /or earlier version

Other links and resources (optional)
Links to research and resources

Worked example

Make the purpose of your page clear

Use a clear title or heading that summarizes the purpose of a page, or other clear signposts that have been tested by users with cognitive disabilities.

How it helps

This helps many people, including those with poor memory, attention, or anyone who is easily distracted. This includes people with age-appropriate forgetfulness, or Attention-Deficit/Hyperactivity Disorder.

For example, someone with mild dementia is using online shopping. They get distracted and then when they look at the screen again they have forgotten what they were doing. Put a clear heading at the top of each page that shows clearly **what the page is about and what they are doing**.

More details

Heading need to clarify the purpose of this specific page

Examples

Success example:

Headings tell me exactly where I am

Failure example:

Heading that doesn't clarify the step in a form

Technical details

More technical details and testable wording proposals are available at:

coga github: clear purpose

WCAG issue 55

Pull request

Patterns

Clear Structure and Relationships. (Jennie)

Structure, content and controls must be easy to identify and understand.

The user should not experience difficulty identifying which control to use with specific parts of the content, or how to use the controls. The user should be provided visual cues and text that communicate the structure and layout of the page.

How It Helps

This helps many people, including those with cognitive disabilities that impact problem solving skills, those that get overwhelmed when presented with a lot of text, and those with difficulties with more complicated tasks. This can include some individuals with early stages of dementia, people who have had a concussion or a stroke, people with intellectual disabilities, and others. Those impacted may not complete tasks, miss key information, and not return to pages that are complicated to use and understand.

By creating clear layouts, with easy to use controls, a greater number of individuals can focus on the task instead of spending time figuring out how to use the controls and information. They easily find key information, and are more likely to return to the page.
One example: do not have two scroll bars close together. Some users may find it difficult to determine which one to use with a particular section of content. Instead, use clear visual layout and placement of the scroll bars, break the content into two separate pages, or consider removing unnecessary information from the page.

More Details

Some people with disabilities need a clear layout to help them know how to use the information. When structure and relationships are unclear, end users may need to experiment with different layouts and structures until they work out how to use them. However, people with cognitive disabilities may not be able to do so, and may not be able to use the content or application.

Example 1: A website has 2 scroll bars, each operates a different section. When users try the wrong scroll bar, they do not get the effect they desire. Many users will look again at the content; try and work out what they are supposed to do; and discover the correct scroll bar. However, many people with cognitive disabilities will not be able to work out what they did incorrectly. Others will feel cognitive overload, and will give up before they try. They may

assume the application is broken, or that it is just too complicated for them. For all of these users, the application will not be usable.

Example 2: Chunks of content run into each other with a "flat design". Whereas some users can work out which chunks belong together, many users with cognitive disabilities will find it challenging or impossible. Thus, all the benefits of chunking content are lost.

Examples

Success examples include:
Using Summary Content - Placing an outline of content at the top of the page, and/or key information and elements “above the fold” will provide greater access. For example, in an email application, the send button should be near the top so it is easy to find and use.

Separating Content - White space, call outs, and other methods should be used in addition to headings to help define sections of content. This organizes the information on the page so it is easier to determine layout and find specific information.

Separating Interactive Elements - Place interactive elements like scroll bars and buttons close to the content they can impact. Also, keep interactive elements further from content to which they do not apply. This makes it easier to identify which elements will impact each section of content.

Failure examples include:
Users may not be easily able to find features that do not display in their initial view of the page (items “under the fold”). For example, in an email application, the send button being located under the email’s body text.

Dense text, with little white space, no call outs, and a lack of visually differentiated headings to define sections.

Pages with scroll bars close together that impact different content areas.

Technical details

On coga github: clear-structure-and-relationships.html

wcag issue 26 and wcag issue 39

Sub-menu items that are clearly associated with the main menu items under which they fall. (John K)

Splitting into 2 patterns:

A: Can the user identify that there are sub-menu items and how to get to them? (Jennie)

B: Can the user understand why that sub-menu item is paired with that menu item? This would help them predict where to find them. (John K)

How it helps

Confusion can occur when visual hierarchy of information is not immediately apparent to the user. Distinction between levels in content hierarchy may be difficult to understand or perceive due to minimal type size or type weight differences or color differences that are not easily perceived or understood. Furthermore hierarchy solely dependent on small unique design elements [s.a. rotating chevrons unique similar sized icons] may create confusion.

For example a drop down accordion menu of additional submenu items may not be viewable without understanding it needs to be clicked (or 'rolled over') as indicated by a small unique design element.

More details

Small design elements that indicate submenu items will be presented that aren't always readily apparent or meaningful to the user and not universally adopted.
Examples include a chevron (triangle). In different designs It can be either left facing right facing, up facing or down facing depending on state and the unique design standard affecting interpretation of state.

Confusion can occur when a right facing chevron can indicate that more information will be presented on the current page when clicked or it may mean that information is currently being presented, or it may indicate that it takes the user to a new page. Consistent and general best practices should be used to make it understandable to the user.
Furthermore a series of these nested elements on a long page can create visual hierarchy confusion if dependent solely on interpretation of design elements.

Getting started

Create a visually clear hierarchy of in-line information in either a revealed or hidden state. Clearly indicate when text is hidden or when it can be hidden or revealed.

Examples

An example of consistent “+” sign to show that additional information will be shown when pressed..

Success example: +

Failure example: <

If there is no clear indication that more information will or will not be revealed when clicked on. Inconsistent styles or left right arrow directions can confuse user to differentiate between revealed state and hidden state.

Technical details

On coga github: clear-structure-and-relationships.html

wcag issue 26

A: Sub-menu items: easy to know they are there, how to get to them

Users should be able to easily identify that there are sub-menu items and how to get to them.

How It Helps

When opening a web page for the first time, the sub-menus are typically collapsed. And, their design may make it difficult to even know that they are there. Some users with cognitive disabilities may not guess that they are present, or after seeing one expand by accident, may not generalize that this structure may be present for other items in a menu. Making it easy to notice that there are sub-menu items ensures the user can use this part of your site. An example is a menu without any visual indication that there are sub-menu items.

Also, there are times where how to open the sub-menu item may not be easy for some with cognitive disabilities. If the control to expand a menu item relies on a particular gesture or way of rolling over the area with a mouse, for example, the end user may not figure out how to expand the sub-menu and may abandon the task. An example would be a menu that expands only after moving the mouse over a particular side of the menu text.

Getting Started

1. All menu items which have a sub-menu have a clear visual indicator that they are present. This can include (but is not limited to): an arrow, a plus sign, or a triangle.

2. All menu items with sub-menu items, when accessed with a mouse (note: because keyboard operation is already part of a SC) expand using standard mouse movements and clicks. If a particular area of the menu requires the interaction, or a separate control (e.g. a triangle requires the click) this is visually easy to identify.

Success examples

- The presence of sub-menu items is easy to find because there are triangles next to their menu items.

Failure example

- No visual indication of sub-menu items is next to the menu item. The only way to discover the presence of the sub-menu item if using a mouse is to move the mouse over the location of the sub-menu item.
- Expanding to view the sub-menu items requires interaction by mouse with a specific area of the menu item, and this area is not visually distinguished.

Make it easy to find help and feedback channels (Glenda, Steve

Make it easy for the user to ask for help or report issues. This includes:

- Easy to Use: Feedback information and forms are simple and clear. (User testing with different user groups is highly recommended.)
- Easy to Find: Available from any place where the user may get stuck

The option to provide feedback should never require the user to manage complex menu systems such as Integrated Voice Menus (IVR) with many different options.

How it helps

Providing an easy way for users to give feedback will help people be able to share problems, ask for help, make suggestions and positive comments. If users cannot give feedback easily, problems will continue to exist without the site owner being aware of the problems. Ideas for improvements and positive feedback will also be missed.

More details

Make sure the feedback option is:

- simple to use

- available in all stages of the process
- has a process in place to respond helpfully to any feedback submitted
- does not make the user provide unnecessary information
- does not rely on complex menu systems

Providing multiple methods for gathering feedback is recommended. For example, on a website, consider providing all 4 options for feedback including live chat, a phone number, a web form and a feedback email address.

Note that chat bots may not be appropriate for this particular type of feedback other than to start the feedback process. These can be extremely frustrating if you cannot easily get to the area you are trying to reach.

Examples

A banking website had a major accessibility problem. This problem blocked some customers from paying their bills online. One of these customers found a feedback form on the page where they got stuck. The customer was able to report the problem. A help desk employee reached out to the customer and helped them complete their bill payment successfully. That help desk employee also reported the accessibility problem to the software team. The software team corrected the problem in the next software release. The new design was easier for all users and resulted in more customers successfully paying their bills on time.

- **Web Chat or Web Call** - An option to provide feedback using live chat or a video call. Note: The live chat or video call feature must be fully accessible. Web chat should not be a distraction and easy to close. Check usability with user testing.
- **Phone** - A feedback phone number, ideally with a feature to automatically call via Voice over IP. Make sure there are no complex voice menus.
- **Web Form** - A simple site contact form with *no more than 3 required fields*
- **Email** - An email link using the 'mailto' protocol with prefilled "to" and "subject" fields. Note will not work on all platforms or all mail clients.
- **Interactive Voice Response (IVR)** - Provide an automatic option at the end of a IVR to give feedback by pressing a specific digit on the phone.

NB !The following objective level information needs to me merged in to the new objective with the above pattern

User Testing:

Make sure your user testing has a wide range of different cognitive disabilities represented.

Test for the following:

- *Are enough user groups represented*
- For example a typical project may wish to include: People living with early stage dementia, age appropriate forgetfulness , intellectual disabilities, different specific learning disabilities and communication disorders .
- Identify the different ways a user may “Report Issues and Problems”
- Can the user find a way to submit their feedback without asking for help?
- Can the user submit their feedback at each stage of the process including from the home page and any place they may get stuck?
- Does the user make errors trying to submit their feedback?
- Does the user find it easy to submit their feedback?
- Does the user’s mood deteriorate when submitting feedback? (A sign of frustration)
- Ask the users if they would find this easy to do if under stress or tired.
- Ask the user where they might have trouble if they were under stress.
- Does the user understand the feedback process and are they able to complete the task? Use concrete ways to check that the user understands. For example: Is the user able to identify if/when they will receive a response back? The method a response back may come (e.g. email, phone). Where the feedback goes/what happens to the feedback?
    - make sure it is simple to use , and does not require a lot of information that will prevent !people from givingiving feedback
    - Confirm it is available at different stages in the process and is one click away
    - Confirm that when feedback is given and a process is in place for acting on it!

User Stories:

Reporting issues is usable by everyone. This leads to the following user stores:

- As a user with a cognitive disability I want to be able to give feedback easily.
- As a user with a cognitive disability I want to be able to give feedback as soon as I get stuck from any part in the process.
- As a user with a cognitive disability I want to be able to give feedback in any form that other people can. When I try and give feedback and can not manage I feel excluded and think the organization does not care about me.
    -

Provide Alternatives for numbers (Lisa)

Provide alternatives for numbers and numerical concepts

How it helps

Not all people can understand numbers and numerical concepts.

For example, some people have dyscalculia, a learning disability specifically-related to mathematics. People with dyscalculia have significant problems with numbers and mathematical concepts, but still have a normal or above-normal IQ.

For example, a user with dyscalculia may have difficulty processing temperature data when presented only in a numeric format. However if non-numeric alternatives are provided (cold, warm, hot etc.) then they are able to understand the content.

Numeracy issues can occur due to a range of disabilities, the most severe being the inability to read or understand numbers. Other people have challenges with any calculations such as relative sizes or times. When reading measurement an individual with cognitive impairment may understand the concept of 90cms as a length but find it hard to cope with the fact that 0.9m and 900mm are the same length.

For example, a train schedule has a long list of relative times that they train leaves for different zones on the hour. The user can not calculate when the next train leaves from their location.

More details

Where an understanding of mathematics is not a primary requirement for using this content use one of the following:

- Reinforce numbers with non-numerical concepts, e.g., Very Cold, Cold, Cool, Mild, Warm, Hot, Very Hot
- Using personalization semantics to supply a non-numerical concepts (when mature) See https://github.com/w3c/personalization-semantics/wiki and aui-numberfree

It should be noted that different users may find math easier to understand than long text.

Where some math skills are essential for the content:

- Move towards digital math that can be extended (not numbers in images)
- Enable highlighting of sections as they are being discussed
- Link sections of numbers to extra help that can be read together
- Enable replacing math sections with words or summaries for users who prefer this.

Content that may need extra support include:

1. understanding what the representation of a number may mean as a concept of:
2. size
3. quantity
4. distance
5. time
6. date
7. temperature
8. positive/negative
9. calculation
10. sequencing
11. memory
12. cultural differences
13. alternative representation

Technical details and history

See wcag issue 32

Use white spacing (lisa)

Put white space around objects and text, including boxes, paragraph headings, and content, so that each section is clearly separated.

How it helps

White space (also called negative space or the background color) reduces clutter and provides definition to content. This gives the viewer a clear overview of a web page. It is used by designers to enhance text and the position of objects on a page.

Use of white space aids navigation through a page and helps people read it. It can help the user find important elements on a page. For those with cognitive impairments, it has been shown to ease reading difficulties and improves understanding of content.

Also, make sure the user can also adjust the amount of white space around objects and text via a web extension or user setting. This supports the ability to identify important elements in the content of a web page.

More details

Use clear spacing between letters, words , sentences lines, paragraphs and blocks of text.

Allow for the ability to easily adjust white space around objects and text, including boxes, paragraph headings, and content, to a degree that suits the user and does not disrupt the overall integrity of a web page.

Note that “ white space” is a term that means the background color. It does not always need to be always white!

Example

Technical details and history

Use clear spacing between letters, words, sentences, lines, paragraphs and blocks of text. Text is not fully justified and On coga github: visual-presentation.html, wcag issue 51 and PR number 113.

Provide reminders (lisa)

Make it easy for the user to set a reminder for date and time sensitive events. Use standard API's when possible.

Reminders must be set only at the user's request and the user must be able to personalize the reminder method.

How it helps

People with cognitive and learning difficulties often have challenges managing events and time. In fact, being unable to correctly manage events and time without support is a diagnostic criteria for some groups of disabilities. This results in missing meetings, not submitting a request by a certain date or a form within a specified time period.

Using calendar APIs (or task manager) that allow the user to automatically add events and deadlines to their own calendar can help in many cases, for example:

- When the user copies information into a calendar they often copy the day or time incorrectly.
- The user is challenged processing and retaining time based information.
- The user is challenged in sequencing time bound events.
- The user's skills decrease when tired to such an extent that they have to stop a task. They may wish to reschedule the task.

For example, a user with a learning disability set a doctors appointment online. Often they copy the detailing incorrectly onto their calendar. However, the website gives them an option to add the appointment to the calendar and sets a reminder an hour before. The user now comes to the correct place at the correct time.

The benefit to users with cognitive accessibility needs is that they can independently manage appointments, deadlines and schedules. The ability to set reminders can reduce the cognitive load associated when processing time bound tasks. Time dependent activities may be monitored and tracked by the user to ensure that they are completed in a timely manner.

Always give the option to set a reminder at the end of the task so that the user does not get interrupted.

It is essential not to add unwanted reminders as this makes the users calendar too full. This can even prevent them from being able to use their calendar at all. The user is the best person to know how many reminders, and which type, will best meet their needs.

More Details

Where a standard mechanism exists for the platform or technologies, it must be used.

See

https://developer.mozilla.org/en/docs/Web/API/notification

https://www.html5rocks.com/en/tutorials/notifications/quick/

https://developers.google.com/google-apps/calendar/concepts/reminders

https://docs.api.ai/docs/reminders-and-notifications

Date and time sensitive events are any event that has to be completed by a certain time. The time constraints on such an event may be defined by a calendar date and time or by the total elapsed time.

Variables that could be considered on 'when to supply a reminder' include:

- Time - at a logical time
- Location - prompted when at an appropriate location
- Context - on computer vs. mobile, on specific site, etc

This design pattern addresses two broad classes of issues associated with this type of information:

- If the user perceives the activity to be too complex the user may decide to abandon the activity and therefore be excluded from the information and/or services derived from the completion of the activity.
- If the activity relies on a number of distinct events being carried out sequentially over an extended period of time or if a single event must be completed by a specified date and time then the likelihood of errors being made during the activity increases, particularly for users with cognitive accessibility needs. Activities are often missed because the date and time is confused.

Example

A health care site allows you to set a local medical appointment. Once the appointment is set the user is given the option to add it to their calendar (automatically) with a reminder three hours before. They are also given the option to add or edit the reminder.

Technical details

The following are proposals for WCAG.

On coga github: reminders.html

wcag issue 34

Tell the users about fees and charges at the beginning of a task (lisa)

- Tell the user about all charges at the start of a transaction.
- Tell them the typical value for each type of charge.

Any conditions and terms should be available at the start of the transaction in easy language.

How it helps

Users with cognitive disabilities who have trouble with memory, attention to detail or reading comprehension may not be aware of charges unless they are explicitly noted at the start of a transaction task. Terms and conditions can be under a link but charges must be clearly displayed and available in plain language.

Clearly identifying charges at the start of a sale benefits all users. Those with cognitive disabilities will particularly benefit because some groups are less likely to have inferred or guessed the charges would be included. They may not know to look in other locations in the user flow or in another location, for example on the homepage, or on a rates page.

People with impaired Executive Function or memory need to have all the consequences presented in an orderly form to be able to make an informed decision. When charges are not clear, the consent of the transaction is unclear.

It also can take much longer for users with disabilities to go through the process of making a purchase. If a person has spent hours making an online purchase, it is much more difficult and upsetting to find out that they cannot afford it. They will often blame themselves for not understanding the price and may experience a loss of confidence. They may stop trusting themselves for day-to-day activities.

For example a person with Executive Function challenges may be trying to order a plane ticket, and not realize that there are extra fees not quoted in the original price, such as taxes, international fees, baggage fees, etc. They may spend hours booking a holiday only to find that they can’t afford it. Or, they end up purchasing something they cannot afford. And even if they have completed this process in the past, they are not able to bring their experience into future purchases to anticipate the final price. The result is the user loses confidence in their ability to independently purchase a holiday online, may have incurred a debt they are unable to pay, may not attempt again, or only with the help of a hired professional (e.g. travel agent or assistant).

Examples

Pass

- There are no surprise charges or conditions.
- Users are aware of all charges (including hidden fees) and can make an informed decision when they decide to purchase an item and put it in a shopping cart.

- For items that shipping charges vary, the range of shipping charges and the issues that change the rate would be listed, along with a link to where more details can be found. For example, weight and speed of shipping my impact your shipping fees which can be between $4 and $400 depending on location.

Failures

- Final transaction includes unknown charges that result in higher-than-expected total charges.
- Final transactions include conditions of purchase that are not clear to users from the beginning of the task.
- Transactions contain charges or conditions that the user did not know about until they have invested a lot of effort into the sale.
- Final transaction is completed and users are surprised by the total they receive.

Technical details

The following are proposals for WCAG. They experiment with more testable language.

On coga github: identify-charges.html

wcag issue 37

Add search (Lisa)

Search: Provide a search capability.

How it helps

Having a search capability allows users to find the content they need even if they cannot manage to find content via the site menus or other browsing mechanisms. A user can learn how to use search and reuse that skill on many sites.

User studies have shown the aging population tend to rely on search to locate information rather than navigating menus or surfing.

Menu systems, and most site navigation require the user to understand the categories of the pages that the content provider has created for general use. Users with impaired executive function may be unable to identify the correct categories

In some cases, users know the correct category via memory, rather than logic. For example, most users remember that the print function is often found under the file menu. Users with impaired memory may not be able to find these menu items based on recall.

Navigating a site and going to many pages to look for content, is also difficult for people with impaired short-term memory or for people who are easily distracted, as they may lose focus and forget what they are looking for.

Search is most useful when it corrects misspellings, finds appropriate or related content or provides suggested auto-corrected versions of the search terms that the user can choose from.

More details

This is less important on small sites if every page is a maximum of two clicks away from the main page

Examples

A site has a search

Technical details

On coga github: search.html

wcag issue 44

Changing context

Avoid changes of content or context that may confuse the user.

Make it easy for the user to switch off automated changes of context, functionality, settings, navigation such as routes, and orientation.

Provide an easily available mechanism to go to previous context, functionality, settings, route and orientation

How it helps

Content, settings or functionality which change unexpectedly, without user initiation, can result in significant barriers for users with cognitive disabilities. Unexpected changes in any of these areas can result in loss of focus, anxiety, or confusion in understanding or using a user interface component (such as menus, buttons and design components). Examples include but are not limited to:

- Automatic launching of new windows or pop-ups
- Submission of forms through mechanisms other than a button that is clearly labeled using simple language to submit the form
- Rerouting automatically by a GPS
- Changing the direction of a map in a GPS

For example, a user may not have a sense of direction or know their left and right. This makes following GPS instructions difficult. To avoid making errors, they may study the route before using a GPS so that they know approximately what they are doing and can augment the directions of the GPS with their own knowledge. If the GPS automatically reroutes them because of a small traffic delay, these users may become lost and disoriented and no longer be able to use the application.

Avoiding changes in context and allowing users to turn off automatic changes give users with cognitive disabilities more control over how Websites and applications behave.

More details

Exception: The changes are part of an activity where it is essential (e.g. a game)

route: Directions and flow such as a GPS route

orientation: perspective or view such as map direction

easily available (or easily available mode or setting): one or more of the following is true:

- It can be set one time with as a wide a scope as possible (such as using the standards of the OS, From ISO 9241-112 or GPII when available);
- It has the option to save or to change the setting, were available interoperably, but also for the scope of the set of web pages;
- It is reachable from each screen where it may be needed, and the path and the control conforms to all of this document.
- It can be set one time with as wide a scope as possible (such as using the standards of the OS, ETSI or GPII when available); and
- It has the option to save or to change the setting, were available interoperably, but also for the scope of the set of Web pages; and
- It is reachable from each screen where it may be needed, and the path and the control conforms to all of the document.

Examples

Success examples:
In a GPS application the user can

- Turn off rerouting automatically in one click
- The user can go back to old route in one click
- The user can set rerouting defaults such as each new turn saves at least four minutes

Submission of forms are clearly labeled using simple language to submit the form and can be undone in one click

Technical details

Wcag issues

Keep the user information safe (lisa)

Keep the user safe. This includes:

- Checking how safety and security techniques work with a wide range of customized profile including aging, learning and cognitive disabilities.
- Using known techniques to keep sensitive user information safe.
- Ensuring all users understand known risk.

How it helps

It is vital that users stay safe on the internet. Information which suggests a user has Dementia or an intellectual disability allows predatory companies and individuals to target that user for scams or other risky activity. For example, a predatory company could send requests for money, saying “you haven’t made your donation” despite the user having made one. Avoid storing information that implies this or, if you do store it, provide strong security for that information.

Users with weak executive functioning are less likely to identify risks correctly so clearly identifying potential risks helps the user stay safe. Add helpful tips for staying safe while using your content and provide help in case of problems.

To help identify risks, we suggest holding research and focus groups with people with cognitive and learning disabilities and to work with people with disabilities to solve potential and existing problems.

For example, many people who can not copy and paste passwords or use two-step authorization codes ask a caregiver to help them. As caregivers are often just temporary employees, this leaves the user completely exposed. Making passwords longer or requiring users change them regularly increases these unsafe practices and actually makes the application less secure for many people. This type of design error is common because people with cognitive and learning disabilities are left out of the user research and analysis.

More details

???

Examples

Success examples:

- ,Providing alternative login options that have been tested with people who have learning or cognitive disabilities that are approved security techniques, in your jurisdiction, for sensitive data
- Working with a wide range of people with learning and cognitive disabilities
- Using industry best practices for storing and securing user information

- Using consent forms in plain language that have been tested with people with learning and cognitive disabilities to ensure they understand the risks.

Technical details

There are some useful resources and easryreader.eu

On coga github: user-information.html

wcag issue 25

Objective 8: Support personalization and adaptation

Personalization can enable us to really meet the individual user's needs. Some users need extra support which we can support with minimal effort from the user via personalization. This can include:

- Familiar symbols and graphics that the user knows
- Tool tips;
- Language they understand such as literal language or common terms;
- Less features and options to the optimum number for the individual user;
- Keyboard short cuts that are familiar to the user;
- Widgets for help.

Personalization works by having user preferences and a series of open source adaptive widgets that can supply the user with the extra support they need with minimum work for the author. It also enables to keep their original design in tack, whilst enabling a personalized experiences for different users. We need personalization because:

- Sometimes user needs conflict;
- Learning new designs, patterns and widgets can be confusing - we want to allow users use widgets they already know;
- Extra support can be annoying to people who do not need it;
- Making content predictable is necessary for accessibility for some people but can often be considered boring design for other users;
- Ability to change levels of complexity and simplify the content is important as people skills improve or decrease over time or context.

In another use-case we would like to see interoperable symbol set codes for non-verbal users. Products for people who are non-vocal often use symbols to help users communicate. These symbols are in fact peoples language. Unfortunately, many of these symbols are both subject to copy write and are not interoperable. That means end-users can only use one device, and can-not use apps or AT from a different company. An open set of references for symbol codes for these symbol sets however, could be interoperable. That means the end user could use an open source symbol set

or buy the symbols and use them across different devices or applications. Symbols could still be proprietary but they would also be interoperable.

8.1User testing

Make sure your user testing has all the different cognitive disabilities represented. (Make sure people don't just ask these as questions, but ask something that demonstrates it.)

Test for the following:

- Test with templates from personalization groups (when available) and common personalization scenarios used by people with cognitive disabilities, such as less content, adding and changing symbols and simplified text
- When content is simplified, test that wanted content is not lost, and that critical path still work
- Test that forms autofill works

8.2 User stories

(take from personlization group?)

This leads to the following user stories:

- I am familiar with the UI, and I know how to work it and what will happen when I work it.
- Content delivered in an easy-to-understand mode.
- Simple content with few options and consistent text .
- I find the content easy to understand.

Support simplification (lisa)

Support simplification of your content.

Often this includes:

- Allowing the user to remove or hide features that most users do not use or that are not essential.
- Allowing the user to get less test or more simple text

How it helps

A user who has difficulty reading or using Web content can be easily overwhelmed with too much information on a web page. They need to simplify the page to just the critical information that they need and not spend all their energy reading and understanding other content and features. This is also true for users who are easily distracted.

For example, an email program has lots of features and formatting options when drafting an email. This makes it t to complex for a lot of people. With personalization the user can have a simple options with only send and cancel options. There is a “to” and subject line but no cc or bcc options. In this setting there is a clear heading (write an email) and they have icons that the user understands .

More details

Note that:

- Typically a simple application has 3 to 6 functions.
- Make sure it is easy to get back to the full featured version

You can meet this design pattern by:

- Use aui-simplification on regions and controls
- Use other attributes in personalization semantics
- Add a simplification toolbar
- Provide an alternative version

Getting started

Add aui-simplification=”critical” on content that is in any critical user testing paths

Examples

(These should make it easier to understand)
Success example:

Simplified “reading” view is available and easy to close.

Failure example:

A busy email program with lots of regions with different controls bars and features such as tagging, group tagging, start a new thread etc. There is no easy way to simplify the page.

Technical details

The following are proposals for WCAG. They experiment with more testable language

Support personalization of symbols and controls (lisa)

Add semantics on control links and symbols that allows extra support that is appropriate for each user.

For example:

- Html 5 autocomplete on common fields
- Adding a toolbar that adds personalized images
- Use attributes in personalization semantics

EDITOR'S NOTE

Editor's note: Personalization technology is still young. At the of publication HTML autocorrect on fields was the best supported.

More details

- Test your page with user profiles with different icons to check that the personalization is working.
- You can also provide a link where necessary to supporting user agent or have a personalization options tool bar on your page.

How it helps

Personalization changes the interface to meet the needs of the user.

Having familiar terms and symbols is key to many users being able to use the web. However, what is familiar for one user may be unfamiliar to another requiring them to learn new symbols. Adding semantics allow symbols and support to be added by an extension or browser that is familiar to the individual user.

A stronger example is people using Alternative and Augmentative Communication (AAC) systems. AAC systems designed for people who are non-verbal often use symbols with or without text.

These users usually only learn one symbol set. They cannot easily communicate with other symbol users in a written format or may struggle to understand different symbols used in different applications. Some symbols are subject to copyright and cannot be shared across applications.

If users' symbols are mapped to the same concepts, then user agents can load the symbols

that are understandable by the user and they user can access the web and other applications.

Other support include autocomplete and extension that help the user fill out forms and understand the content. Many users with memory or executive functions impairments could not fill in forms without asking someone to help copy over information or check their work. Autocomplete, many more users to manage forms by themselves.

Getting started

Use Html 5 autocomplete on all common fields

Add a toolbar that adds personalized images or

Add the semantics that can work with a toolbar for personalized images

Examples

Success examples

Example taken from adaptable ui personalisation.

aui-action="compose"

A page working with autocorrect

Examples of working pages can be found Implementations of Semantics, also see Getting started with personalization semantics

Technical Details

On coga github: support-personalization.html

wcag issue 6

Enable APIs and extensions

Allow supporting APIs and extensions to work with your content.

How it helps

People with cognitive disabilities are often using add-ons or extensions as assistive technology. This includes:

- Reading of the long form of acronyms
- Support for text-to-speech with synchronized highlighting of the phrase being read
- Content simplification
- Creating mind maps out of the heading structure
- Support for retaining content that has already been entered
- Password management
- Spell checking
- Changing the symbols or the interface

If these functions are not supported, the author should provide all support all the functions of the add-ons in use as assistive technology.

For example , a user with traumatic brain injury has executive function and memory impairments impacting their ability to remember details such as:

- The Internet of Things (IoT) interface
- Their user name and password
- What an acronym stands for
- A phone number
- The meaning of uncommon words

Supporting the use of an add on that simplifies content and adds help (such as the long form of acronyms, and a popup dictionary) enables them to understand most content.

Supporting password management tools enables him to successfully login and avoid being locked out of secure sites.

Storing non-sensitive information and auto complete helps them fill out a form. This suggests common information, like a person's phone number or address. It also helps them avoid making mistakes. It eliminates the need for accurately recalling this information from memory or having to copy and paste it, which is a task that often prevent them from successfully using a form.

When overwhelmed by textual content, they have an extension that inserts symbols that they

are familiar with that helps them find the content they need.

However, sometimes a web site stops their extensions and API's from working. The result is that the use can not use this web site.

The Internet of Things (IoT) and the Web of things.

Another example is a person that has a language related disability (e.g. Dyslexia) which may cause them to read at a slower rate. They often miss objects and information that they did not know they needed to read. While high-literacy readers scan text, low-literacy users may read the text "word-for-word." This means they only know what is written when they have intentionally read out. This is similar to having a narrow field of view and they can be unaware of other items not directly in the flow of text that they are reading.They may also miss things on the page that help them understand what to do. For example, there may be a side bar or call out box which helps make decisions about which link to go to that matches their particular needs. These readers may select the earlier option they encounter, because they may not have noticed (or prioritized) reading that side bar content. They also might miss information that is essential for successfully completing an interaction.

Too many options may add to the complexity of interacting with IoT devices. Additional options should be easy to ignore and not require a lot of reading to understand that they are additional, as well as how to skip them.

Sometimes IoT interfaces may confuse the user, such as a default "reading" on a meter being set to "2" and not "1." The user would then need to reset it to "1."

It is important in any proposed solution to make operational tasks, such as interacting with the IoT, as transparent as possible so that users can focus their attention on the functional aspects, such as relating to content..

More details

This includes:

- Support of standards for the Internet of Things that allow for simplified and personalized interfaces such as the ISO/IEC 24752 "Universal Remote Console Framework" standard, URC
- Standardized techniques to support interoperable symbol sets that are used when available.

Exceptions:

- When there is a security or safety requirement, these API's may be disabled for the

relevant field
- If it breaks the main function of the site, such as evaluation and testing applications

Getting started

Content can be used with APIs and extensions that support those with cognitive disabilities.

Testing verified through the use of some of the APIs appropriate for the content. For example:

- Testing with spell checker and password storage apps or extensions
- Test with an extension that add to the left click
- Test with a toolbar that enables simplification or personalization and is designed for people with cognitive disabilities

Examples

Success example:
Browser extension and personalization tool bars work. User is able to apply their settings from a personalization toolbar to improve the useability for them of the page.

Failure example:
Password storage applications do not work.
Distraction removing extension does not work
Spell checker extension does not add options to the right click menu, or does not underline mistakes made by the user
The correct symbols can not be added by a simplification toolbar

Technical details

The following are proposals for WCAG.

On coga github: standardized-apis.html

wcag issue 46

Use clear symbols

Symbols with a clear, unambiguous meaning are added at the beginning of short sentences and phrases to aid understanding. However, as some people have difficulty remembering symbols, use text with the symbol.

- Use clear symbols that can easily be seen and expanded
- Use images understood by different users
- In left-to-right languages, place the image to the left of the text

We are also drafting semantics that will add symbols that are easy to use by the individual user.

How it helps

This section should have an every day understandable example of how it helps or the problems

More details

???

Getting started

Add content here if necessary such as what you would put at sincle A

Examples

(These should make it easier to understand)
Success example:

Failure example:

Technical details

The following are proposals for WCAG. They experiment with more testable language

Make Each Step Clear

In a multistep process, clearly indicate the steps completed, the current step and the steps pending. Ensure the user can quickly determine the current location and progress within a sequence.

How it helps (Rachael)

Clearly indicating the current location and progress helps a user who loses focus or gets distractracted reorient to their current activity without reading a great deal of content or restarting. Providing information about the steps that need to be completed ensures that a user who may find a process difficult to complete can determine if they can successfully finish. Examples include:

- Someone with dementia loses focus when purchasing airline tickets and can not remember if they have selected the departing flight yet. They look at breadcrumbs which clearly indicate the departing flight has been selected and they are currently working on the return flight.
- Someone with an attention disability gets distracted when ordering groceries and needs to pick up where they left off. They look at the list of steps and see that they are currently entering a delivery address and the next step is entering payment information. They quickly know they need to have access to that information.
- Someone with a learning disability is not sure if they will be able to complete the sign up process for a service they need or if it has more steps than they can manage. As they get started, they notice the steps listed on the side with an indication of their status. Based on this, they are able to gauge whether they can complete the task

Examples

Success examples:

- Using breadcrumbs to indicate the current step in the process as well as past and future steps.
- Providing a visible list of all steps in a process and indicate the steps completed and current step.

Failure example:

- Requiring a user complete a form or activity over multiple screens but failing to inform the user of their progress or the steps needed to complete. This increases the chance a user will abandon a task because they are unsure they will be able to complete it.

Technical details

Clear Purpose

Make the purpose of each section clear (JohnR Janina Glenda, Steve)

Make the purpose of each section clear. Often this involves providing clear headings that briefly defines the purpose of each section. Extraneous information that is not directly relevant to the main purpose of a page should be distinctly separated and programmatically determinable.

How it helps

Each page or section of content should be organised and marked so that its purpose is obvious. This might be through the use of headings or labels or even a pyramid style of writing or even non textual markers (with suitable alternatives). A clear organisation of content into pages and sections with obvious purpose allows users to more easily locate relevant sections and to be confident that those they read in detail will match their purposes. There will also be no need to read all the content in case something important is missed. For example, Ads that appear in-line in a section of content are rarely related to the purpose of a section and can be placed in a separate clearly delimited section

Here are some examples of how this will help people:

- A person with memory issues may need a clear heading structure to stay focused.
- Someone with an attention disability gets distracted and then needs to pick up where they left off and headings help
- A slow reader may depend on a heading structure to find important information they need without forcing them to read the whole document.
- Someone with dementia loses focus and then can not remember what they were doing
- Someone with a learning disability is not sure if this application has too many steps and if they will manage. By seeing they are halfway though they can gauge if they can cope with the entire process.

For example, An elementary school publishes a weekly newsletter with interesting stories about activities and im

portant announcements. Important announcements include early school dismissal. If the newsletter has a good heading structure, it will be easier for a parent who is a slow reader to find the important announcements about early school dismissal. Without a good heading structure, the important early dismissal information can be easily missed.

More details

- The heading structure should create an outline of the document that could serve as an abstract of the whole document.
- Heading structure makes the content easier to scan and find more detailed information that a person needs at a moment.
- Want to know more: See articles by Ginny Redish including: AARP Audience-Centered Heuristics: Older Adults
    - "Clearly label content categories; assist recognition and retrieval rather than recall."
    - "If pages are dense with content, is content grouped or otherwise clustered to show what is related?"

Examples

- **Success** example:
    - (find from bbc gel)
    - Show an excellent example with headings in html (use an example from JohnR)
    - Section Abstract - one sentence that describes the purpose of this section.
    - Collapsable sections
- **Failure** example:

Technical details

The following are proposals for WCAG. They experiment with more testable language

Use call out boxes and graphical indicators to group and highlight information (Abi)

[**Notes from Abi**: This replaces Use Call Boxes - I have made this more generic than just call boxes so that it covers other similar UI patterns.

The original text for call boxes included consistent use of graphical clues and symbols but this is covered by other patterns]

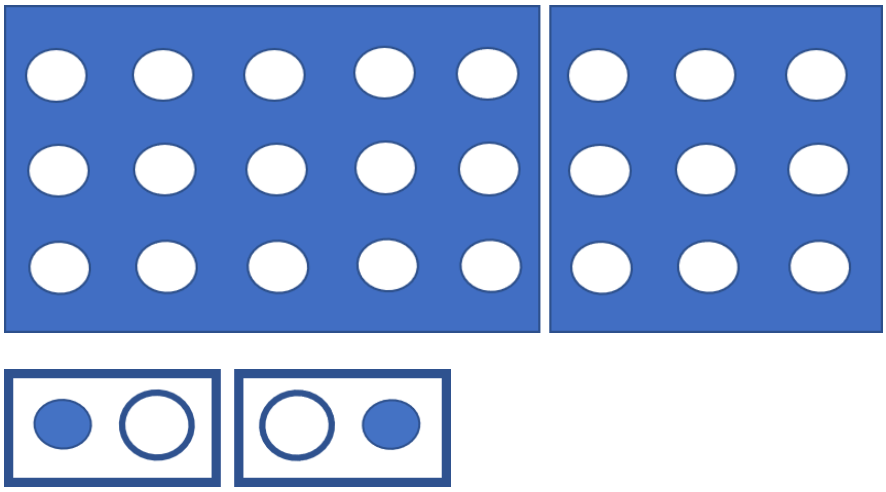
Use graphical indicators to help people identify when items that are associated with each other or have a different purpose to surrounding information.

Examples of common graphical indicators are:

- Group summaries of content with images, such as using a card design
- Flag importing important information, such as using call out boxes
- Indicate different types as information, such as placing quotes in speech bubbles.

How it helps

Visual of grouping of information is based on the psychological principle of Common Regions . It has been found that the grouping of information using a border or colour shading makes it easier for people to identify clusters of information, even if the content of the group is not similar.



Figures: Example of the Common Regions grouping principle.

These graphical indicators allow people to identify structure and information types, without reading text. They also provide distinguishing features between various groups of information. Using the graphical indicators consistently to indicate similar types of items aids with navigating content and reduces the cognitive load.

People who have difficulty with recognising or comprehending written language, and/or poor concentration and memory skills can find it easier to process graphical cues.

People on the autism spectrum are more likely to identify graphical indicators and the use of colour for grouping content than other approaches.

More details

Graphical indicators should also be presented programmatically to enable assistive technologies to interpret the type of indicator being presented. It is also important the graphical indicators do not clutter the interface and are used consistently, as that can add an additional cognitive load for users to process.

Examples:

- **Success** example:
    - Graphical inditaor used to differntiate an example within an article https://www.gov.uk/guidance/accessibility-requirements-for-public-sector-websites-and-apps#making-content-accessible
    - BBC GEL cards design: https://www.bbc.co.uk/gel/guidelines/cards

Related to the following standards:

ISO/FDIS 9241-112 [i.23].

ETSI 203 350 Guidelines for the design of mobile ICT devices and their related applications for people with cognitive disabilities: 12.4.2, 12.4.16, 12.4.16, 12.4.18, 12.4.21

Chunk Media (steve)

Long pieces of media are divided into logical segments with a description of each segment.

How it helps

Providing shorter logical segments allows a person to find and review a specific topic. If that person loses concentration or steps away, clear segmentation allows them to easily find their place in the material and start again. This is especially important for educational style content where review is often necessary.

Chunking media also allows for each segment to be given a unique URL and so easily referenced and shared.

For example:

- Some videos can be naturally organised into chapters or segments
- A podcast can be split into segments rather than a single one hour recording

More details

- **Six minutes or less:** Media should typically be divided into segments that are 6 minutes or less in duration.
- **Navigable:** Navigation to each media segment, and a unique, descriptive label must be provided.
- **Logical order:** Navigation to media segments are presented in a logical order.
- **Exception**: Media that has no logical breaking points, do not need to be subdivided.

Examples

Success example: A 30 minute video is divided into 5 sections, each with a descriptive link to play from that point onwards.

Failure example: A 30 minute video contains no subdivisions or descriptions of sections, forcing the user to play it from the beginning or guess starting locations within the video.

Technical details

coga github: clear purpose.
WCAG issue 55.
Pull request
wcag issue 40

Is this design requirement ready to go in a working draft? Please put your name and vote bellow. If it needs work please explain.
Steve -1 Rachael -1 E.A. -1
Issues for next draft: Must define long and agree if exception is needed. I reviewed this and had a sudden thought! I feel it has come a long way but the issue does not really help those playing video games for education or otherwise. Users are tied to the constraints of the levels or parts of the action - should we be considering this genre or being more specific about how this particular design pattern is to be used?
E.A …I agree short videos are better for learners and we advise around 2 minutes! However, imagine historic speeches - would they need to be provided in their original form and also in a divided form?

Use clear and accurate text formatting and punctuation (Abi)

Use the correct punctuation for text, numbers and symbols to improve readability and comprehension.

How it helps

For some readers, decoding words, numbers and symbols is not an automatic task and can be demanding on working memory and executive functions. If they find content too demanding they are at risk of losing its meaning and the overall context of the message that is being conveyed. Some users may use assistive technology or personalisation tools to reduce the cognitive demands of decoding content. An example would be text to speech that reads aloud content allowing for the use of auditory and visual channels.

Users should not need to be distracted from their task in order to decipher letters, numbers, and words. They should be able to focus on understanding the meaning of the content, especially when they are using assistive technology or personalising content. It is important that the content is created with these adaptations in mind, to ensure that all users gain the experience you want them to achieve. Content creators should ensure that information is provided in a way that can be read accurately by assistive technology. It may have to be personalised to suit certain users, but this can be achieved when all the elements are accessible.

Use cases:

“I listen to content so I need it to be announced and phrased correctly so I can understand it.”

“I listen to content and I hear content read incorrectly or skipped, particularly numbers and symbols.”

“If text, numbers or symbols are in an unfamiliar layout, I get really confused.”

More details

Use punctuation correctly for the language you are writing in, as it will affect how the stress and intonation (known as prosody) patterns from the text are heard, when converted into speech. For example, in English, commas and semicolons will result in a short pause in the speech, whereas a hyphen – will generally be ignored. Question marks, exclamation marks and speech marks can result in changes in intonation, such as a rise in the pitch of the voice.

Avoid the use of Roman Numerals and unfamiliar symbols in text were possible. These can confuse readers and are likely to be read incorrectly by text to speech tools. If these symbols are necessary then ensure they are marked up correctly, using techniques such as

MathML and abbreviation expansions to provide additional support. Roman Numerals should be presented in upper case if used in isolation as they are likely to be read as individual letters.

Be aware that long numbers may be read as single digits or phrased as a single number. This is a particular problem for phone numbers or zip codes. While it is difficult to control exactly how these numbers are read aloud, content creators can help by:

- Displaying the content of the number and using HTML semantics to ensure users and assistive technologies are aware of the number's purpose. In addition the following recommendations can assist with improving text to speech rendering:

  For phone numbers, use the correct layout for the locality of the phone number and ensure users can select the whole phone number (including area code), so that text to speech voices can recognise the format and phrase it correctly.

  For Zip / Postal codes, including state or address information close to the number so that speech voices can expand known abbreviations (such as state names) and listeners can perceive the context.

- When writing long numbers, consider what separators will be familiar to your readers and how it will be read aloud. In general, English speaking countries will use commas between thousands and a period as the decimal separator whereas German and other European countries do the opposite. For example, 1,245 would represent one thousand two hundred and forty five in English, but one point two four five in German. Text to speech output will assume the separators are being used in the format of the language of its voice. If this does not match the content then listeners can become easily confused. While replacing thousand separators with a space has become a common convention to avoid confusion, it leads to difficulties with text to speech with long numbers being read out in a disjointed fashion. For example, 120 034 943 can be read as one hundred and twenty, zero three four, nine hundred and forty three.

Consider how you write dates, because once again the text to speech will use the format associated with the language of the voice. A date such as 04/03/2019 will be read as "April 3rd 2019" by a US English voice and "4th of March 2019" by a British English voice. Writing out the month in words can avoid confusion.

Examples

Success example:
Article providing style guidance for writing for text to speech users
https://accessibility.blog.gov.uk/2017/02/08/advice-for-creating-content-that-works-well-with-screen-readers/

Failure example:

Technical details

Use symbols that help the user (EA)

Add familiar images and pictographics that offer meaning and context to content to aid understanding.

How it helps

Some people who have language comprehension difficulties may be able to read mechanically but not understand content. Others who have learning and/or reading difficulties may rely on symbols to understand content and navigate areas where they help illustrate an important point.

For example, a person with aphasia, has the intellectual ability to understand concepts, but struggles with language. They may be dependent on the use of pictographics or symbols to browse pages for information.

The elderly population and those with dementia can find cluttered pages with dense text hard to read on a screen. Clear symbols and images that act as signposts to the text content can be very helpful.

More details

- Use personalization semantics (add link)
- Use clear and unambiguous symbols that can easily be seen and are scalable
- Be aware of cultural differences
- Use the direction of the text to inform the placement of images so the eye is drawn from the image to the text.

Getting started

Provide symbols besides key texts, headings, contact us and help
Use common symbol types with care: Make sure they are in the context of the content and appropriate to the locale. For example: a question mark icon for indicating a help section would be shown in different ways across the world.

Nielsen Norman Group: Icon Usability (2014)
Nielsen Norman Group: Usability Testing of Icons (2016)

Examples

Success example:

A set of instructions where the bullet points are symbols relating to the content within the text.

Failure example:
A page with important instructions and no symbols or images to guide the reader or a very cluttered page of symbols that confuses.

Technical details

G103: Providing visual illustrations, pictures, and symbols to help explain ideas, events, and processes https://www.w3.org/TR/WCAG20-TECHS/G103.html

Additional Resources:

- More technical details with regards to Symbols.
- How Symbols are used

On coga github: extra-symbols.html and wcag issue 50PR # 115.

Form input flexibility from Use clear labels and instructions (Jamie, Michael, a bit of Gareth)

Be forgiving of contextual variations in user input such as currency, time zone, and locale.

How it helps

Forgiving form entry processes help prevent user errors up front. This reduces frustration and enhances the user's autonomy and independence because they can avoid asking for help. This helps anybody with memory issues, people new to culture, people on autism spectrum who may miss cultural context.

More details

Add annoying details hear, such as exceptions (if necessary)

Getting started (optional)

For inputs that collect known data types, such as credit card or telephone numbers or geography names, accept common variations on the input format so users don't have to convert or receive avoidable error messages.

Examples

(These should make it easier to understand)
Success example:
User inputting a financial value can specify the currency used, even if different from the site's.
Failure example:
User inputs value thinking it is US dollars, but the web site processes in Euros.

Success example:
User inputs a credit card number with or without spaces, and the information is accepted.
Failure example:
User inputs credit number with spaces only to be told it must include numbers only.

Success example:
User inputs telephone number as written including country code, region code, and number using brackets.
Failure example:
User inputs telephone number but is told it must include numbers only.

Technical details

???

Instructions from Use clear labels and instructions (Jamie, Michael, a bit of Gareth)

Provide clear instructions in addition to form labels up front when appropriate that is clear, concise, and accessible.

How it helps

Clear instructions help prevent user errors up front. This reduces frustration and enhances the user's autonomy and independence because they can avoid asking for help. This helps anybody with memory issues, people new to culture, people on the autism spectrum who may miss cultural context.

More details

Provide instructions at the start of the process, not simply in an error message.

Getting started (optional)

In a system with common errors, tackle the most impactful errors first and add additional guidance.

Examples

(These should make it easier to understand)
Success example:
Provide an image of a passport with the number highlighted to indicate where the number is that the user should enter.
Failure example:
Request a passport number, but the passport may have several numbers and the user is uncertain which is requested.

Success example:
Explicitly say which day of the week is the start (e.g., Sunday or Monday) in calendar control when booking a hotel.
Failure example:
Site does not clarify start of week, and user reads calendar control wrong and books hotel for wrong day.

Technical details

???

Use a consistent visual design (Jennie, EA, Steve)

Use a consistent visual design with a stable look and feel

How it helps

Those with difficulty understanding how to interact with information need to use cues like color, layout and other visual information to help them know where to look, what they should do and how they should complete that task.

- Ensures consistent user experience by providing familiarity and building confidence
- Makes it possible to easily use and interact with content
- Aids the completion of tasks

More details

This includes:

- Design themes, including heading styles, font choices, symbols, colors, visual appearance of controls, buttons and links should be consistent.
- Headings with the same structural level have the same font and visual style.
- Icons, controls and menu items that have same function and role have the same look and style
- State and focus for elements with similar function and roles have the same style used consistently across a site.
- Layout should be consistent, including position of interactive elements and navigational controls.
- Structure of content and style of presenting information should be consistent throughout, such as organization of block text, images and bullet points.

Getting started

Plan the design for your information before adding content. Think about the colours, font choices and areas where text and images will appear.

Examples

Success Examples:
A web page has two submit buttons, both should visually look and function the same way.
All selected radio buttons on the site look the same.
When all links on a page have keyboard focus the focus indicator looks the same.

Failure Examples:
3 pages have a submit button, but each one is located in a different place on the page.
There are 6 heading level 2s on a page. 4 are styled using Times New Roman, and 2 use Helvetica.

Technical details

On coga github: consistent-identification-and-styles.html,

wcag issue 28,

On coga github: consistent-cues.html,

wcag issue 31

wcag pull request 108

Use a design that the user is likely to recognise and understand (Rachael)

Use common design elements, affordances, and patterns that are familiar to most users.

How it helps

Many users, such as those with mild cognitive impairment or dementia, cannot easily learn and remember new design metaphors. Without these skills, it can be much harder or impossible to locate desired items to interact with, and to know what interactions may do. Users can feel lost or overwhelmed.

Because common design elements are repeated often over a long period of use across many sites, users are more likely to recognize the interactions needed.

More details

Common design elements, affordances and patterns include:

- Links that look like links and buttons look and act like buttons
    - For example, underline links with a standard style throughout a page
    - Links general navigate to a new page
    - Buttons general perform an action

- Uses common design patterns, such as are documented in the ARIA authoring best practices or are used in the most popular sites
    - Very common navigation design patterns and common icons.
    - A platform specific user interface design for navigation mechanisms and icons
    - An adaptive user interface design that can be personalized (see above).
- Create a standard Visual Hierarchy - Place elements where the user is expecting them, such as
    - Putting the search in the top right hand corner in a website
    - The link to the home page is in the top left hand corner
    - Link to ‘contact us’ is in the top navigation
    - Link to the site map is in the footer area
    - Submit button is at the bottom right for a form
- User interface (design) from a prior version: Allow users to revert back to a prior version of the application that they are familiar with.

Getting started

When deciding pages, select standard components that look and behave the way users expect. Use standard conventions for layout such as the home link in the upper left corner, navigation at the top, search in the upper right, etc. and create an obvious visual hierarchy in the page.

Technical details

On coga github: familiar-design-a.html
wcag issue 49, pull request

On coga github: familiar-design-aa.html
wcag issue 35

Make it easy to find the most important things on the page" plus merged "Avoid Scroll” (Jamie)

See Jamie’s Gist - https://gist.github.com/JamieKnightBBC/5b3769361b34d60b376fa17bf894f7e4

Provide help for complex information and tasks (steve)

Provide explanatory content for complex information and tasks, especially if it is important for successful completion of a task.

How it helps

The use of complex information, long documents and data in a particular format can present significant barriers to users with cognitive accessibility needs. Users should be able to understand the information and successfully complete described tasks without requiring further external assistance as much as possible. If content is sufficiently complex to cause barrier, it may be inaccessible and then additional supporting help should be provided.

More details

The complexity of information may be inherent in the information itself, for example the language used, the quantity of information, or the subject matter. In this case, it is likely to need careful explanation, organisation and presentation for all users to be able to understand without any mistakes, confusion or need of assistance.

In addition, the presentation mode of information, such as a graph, diagram or table, may obscure its meaning through complexity. Here, a supporting description and guided interpretation will highlight the the key features the user needs to understand.

Help may be provided in various forms, for example

- Text "asides" providing explanation and help for diagrams
- A supporting chart or graph to illuminate text content
- A supplemental table – as long as it is not itself complex
- Popup on hover explanations of keywords - possibly linked to a glossary
- A summary to interpreting data in a chart or graph
- A flow chart of steps in a process

Examples

Success example:

- The explanation of a medical procedure and success rate statistics is amplified through the use of an additional text aside, a diagram and a graph.
- The multi step process for applying for a visa is made made easier to use by adding flow chart of all the steps that is always visible. Each step in the flowchart has links to extra help and the current step is clearly highlighted.

Failure example:

- A graph and data table of sales figures is shown without any explanation of the key features that relate to the content.

Technical details

This Pattern builds on help related WCAG Success Criteria to ensure users are effectively supported in understanding complex content.

On coga github: help.html
wcag issue 32
pull request 118

Provide help for forms and non-standard controls (steve)

Provide help for any complex forms, particularly when there are multiple steps, unusual interactions or non-standard controls.

How it helps

Users often find forms and related tasks to be the most complex experience with web sites and can easily become confused, unsure, or even completely lost. Providing extra help can make the difference between being able to successfully complete a task and giving up. This is especially true if any part of the form is complex or provides non standard interactions.

More details

The standard HTML forms and controls have been carefully specified for maximum usability and accessibility. They are usually understood by users, especially if they are familiar with web interactions. However, users are likely to experience difficulties if the standard form behaviour has been altered or completely new controls are provided. Assuming the new behaviours have been carefully designed and user tested, users may still require help in order to successfully use them.

Examples of forms and controls that are likely to require additional help

- New behaviours like bank sort code fields where there is auto tabbing between the 3 boxes as digits are entered
- Password fields that require certain character types of characters to be entered
- Surveys with complex interactions, for example where buttons only appear depending on previous answers
- Date entry where there could be some ambiguity about the required format.
- Brand new controls like date pickers

Help can be provided via various mechanisms, perhaps with an accessible help button next to a non standard control.

Examples

Success example:

A help button next to date picker provides accessible pop-up help for using the control.

Success example:

A form has multiple steps. Users are informed on their progress through the stages.

Failure example:

A form has complex mechanism for enabling and disabling sections as you scroll or tab between them but no help is provided..

Technical details

On coga github: help.html
000000000000wcag issue 32
pull request 118

Provide help with directions (steve)

Help is provided help with navigational systems and directions.

How it helps

Some people confuse their left with their right and so will may be unable will be unable to use navigation systems. Others may experience difficulty recognising relative and cardinal directions such as North, South, East, and West,

Extra help with these ensures navigational systems and directions can easily be used without constant reminders being needed. Especially if it built in directly to the system or directions.

More details

Providing alternative terms relative and cardinal directions if possible, for example “the drivers side”. This is helped with personalization.

Consider using sing any standard mechanism for the platform or technologies that provide for personalization of relative and cardinal directions and terms.

Examples

Success example:

SImple personalised disambiguation of left and right is always available

Failure example:

Consistent reference to points of the compass including less well known ones (eg N by NE)

Technical details

On coga github: help.html
wcag issue 32
pull request 118

Avoid too much content on the page (E.A.)

Allow users to have content with less than five choices on each screen. This can be a simplified version or have extra choices linked from clear and descriptive titles.

How it helps

Too much text, too many images and too much other content can cause overload, anxiety and distraction. Keeping content down to a small number of important points reduces the clutter, calms the mind and allows for better understanding whilst aiding memory. In particular, it helps those with a short attention span who will leave the page if it appears complex.

More details

Avoiding long paragraphs and non-meaningful imagery ensures those with cognitive impairments can concentrate on the important points being made.

Keeping to a few short bullet points and limiting to one or two images related to the main subject areas of a website or service allows the user to choose whether to explore the site further.

Examples

Success example:

Google has one of the simplest yet easiest sites to use and this has been copied by companies such as trivago with a simple search box and some clear statements followed by some meaningful images as links to other pages. https://www.trivago.co.uk/

Failure example:

A page with too much content, long menus and images set around long paragraphs of dense text does not help anyone but more importantly the message is lost in an overload of information http://www.gatesnfences.com/

Technical details

Labels or Instructions:
Understanding SC 3.3.2
https://www.w3.org/TR/UNDERSTANDING-WCAG20/minimize-error-cues.html
3.3.2 Labels or Instructions: Labels or instructions are provided when content requires user input. (Level A)

“The intent of this Success Criterion is not to clutter the page with unnecessary information but to provide important cues and instructions that will benefit people with disabilities. Too much information or instruction can be just as much of a hindrance as too little. The goal is to make certain that enough information is provided for the user to accomplish the task without undue confusion or navigation.”

Provide feedback (Lisa)

For each step let the user know if its status and if it was successfully completed .

How it helps

Making the result of each user action clear helps people with a variety of cognitive disabilities:

- understand that their actions were processed (e.g., the click did something);
- prevent uncertainty or doubt regarding the outcome; and
- remember what they just did.

For example, a user with age appropriate forgetfulness, may have difficulty remembering how the interface worked. So when they press the send button that may not feel confident that the form was submitted. Feedback, such as a thank you message, will tell them and make them feel confident in the process.

During a multi-step task this feedback (user-action feedback) can also assist people, with attention or short-term cognitive disabilities remember what they are doing . For example, a user with early dementia may get distracted and then forget exactly were they were in the task. This user-action feedback helps re-orientate them. It also helps avoid them leaving a task by reminding them that they are in a process, and where in the process they currently are.

Provide easily-recognizable success or failure feedback with every user action. When possible the feedback should use a consistent and familiar design patterns. For example:

- After a step in a multi-step task is completed, breadcrumbs display a tick or a checkmark next to that step's name; and, if applicable, the title or the name of the next step is readily apparent.
- After a button is clicked, it should look depressed. (Note that if it is a toggle button, the state should also be programmatically determinable).
- After a form is submitted or an email message is sent, feedback communicating what just happened, such as "Your application was submitted, thank you" or "Your email message was sent" is provided.

More details

The success or failure of every user initiated action is clearly indicated to the user by visual, programmatically-determinable, rapid feedback in the primary modalities of the content. Audio feedback is supported.

Examples

Success example:

- Use WAI-ARIA states to provide state feedback for a toggle button with an animation showing the state (such as a button was pushed)
- Use ARIA-pressed with a visual or a checkbox is checked/unchecked,
- Provide a confirmation message when an email message is successfully sent, or a form is successfully submitted.
- Use aui-feedback from the personalization semantics at https://w3c.github.io/personalization-semantics
- Use a progress-indicator element (e.g., breadcrumbs) to communicate completed and current steps in a multi-step process.
- Provide visible and programmatically-determinable information to indicate a new password satisfies security requirements.

Technical details

The following are proposals for WCAG. They experiment with more testable language.

On coga github: feedback.html

wcag issue 54

pull request 109

Items for editorial review during HTML transfer

- Check language and voice - possibly use bits of W3C style guide
- Check “what to do” sentences are good and consistent - some are incorrect and 1 missing
- Check technical details section of each pattern
- Review titles - some are too long. I attempted to fix this with the web version
- Check against W3C style Guidelines

Leave for next iteration?

- Extract repeated mentions of personalization to own section?
- Are Mentioned or ARIA relevant for coga?
- Review across ALL patterns to look for duplications, and possibly perform lumping and splitting
    - Symbol patterns - there is overlap and ‘use clear symbols is not in the table but IS in the text.
    - Complex information - split into information / data / processes (move last one). Is it help?
- Add the user story(s) as 1st item in each pattern. Move initial sentence to a new “what to do” section

ISSUES found during HTML transfer

These need to be discussed on the list on in meetings