



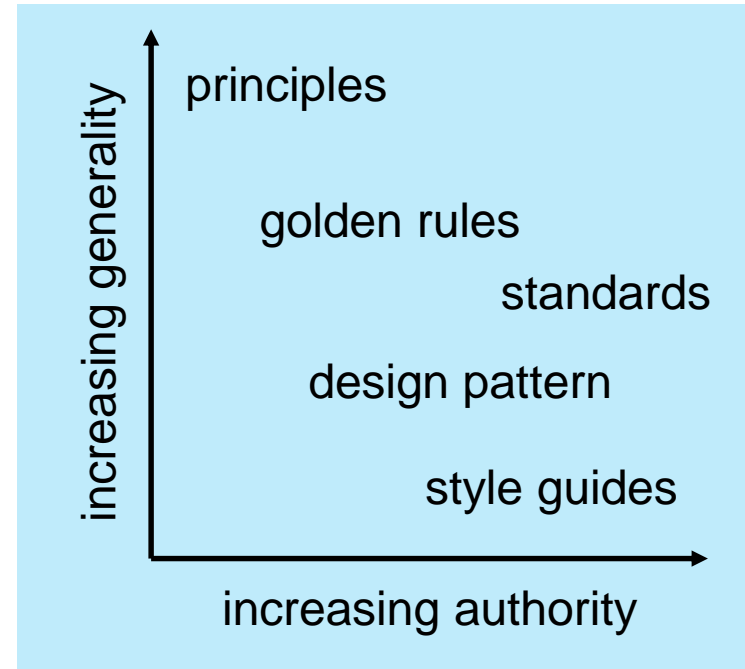
Style Guides and UI Guidelines

Learning Goals

- Know about standards, recommendations, regulations and laws applicable to user interface design
- Understand ...
 - The basic structure of a user interface guideline
 - What is described in a user interface guideline
- Be able to ...
 - find for specific user interface elements, how the should be designed and behave on a given platform (e.g. where to find, how to design a switch in iOS)

Types of Design Rules

- **Principles**
 - abstract design rules
- **Golden rules and heuristics**
 - more concrete than principles
- **Standards**
 - (very) detailed design rules
- **Design patterns**
 - generic solution for a specific problem
- **Style guides**
 - provided for devices, operating systems, widget libraries



- **Authority:** whether or not a rule must be followed or whether it is just suggested
- **Generality:** applied to many design situations or focused on specific application situation.

(more) Guidelines

Hix and Hartson, *Developing User Interfaces*, Wiley, 1993

- User centered design
- Know the user
- Involve the user
- Prevent user errors
- Optimize user operation
- Keep control with the user
- Help the user to get started
- Give a task-based mental model
- Be consistent
- Keep it simple
- Design for memory limitations
- Use recognition rather recall
- Use cognitive directness
- Draw on real world analogies
- Use informative feedback
- Give status indicators
- Use user-centred wording
- Use non-threatening wording
- Use specific constructive advice
- Make the system take the blame
- Do not anthropomorphise
- Use modes cautiously
- Make user action reversible
- Get attention judiciously
- Maintain display inertia
- Organize screen to manage complexity
- Accommodate individual difference

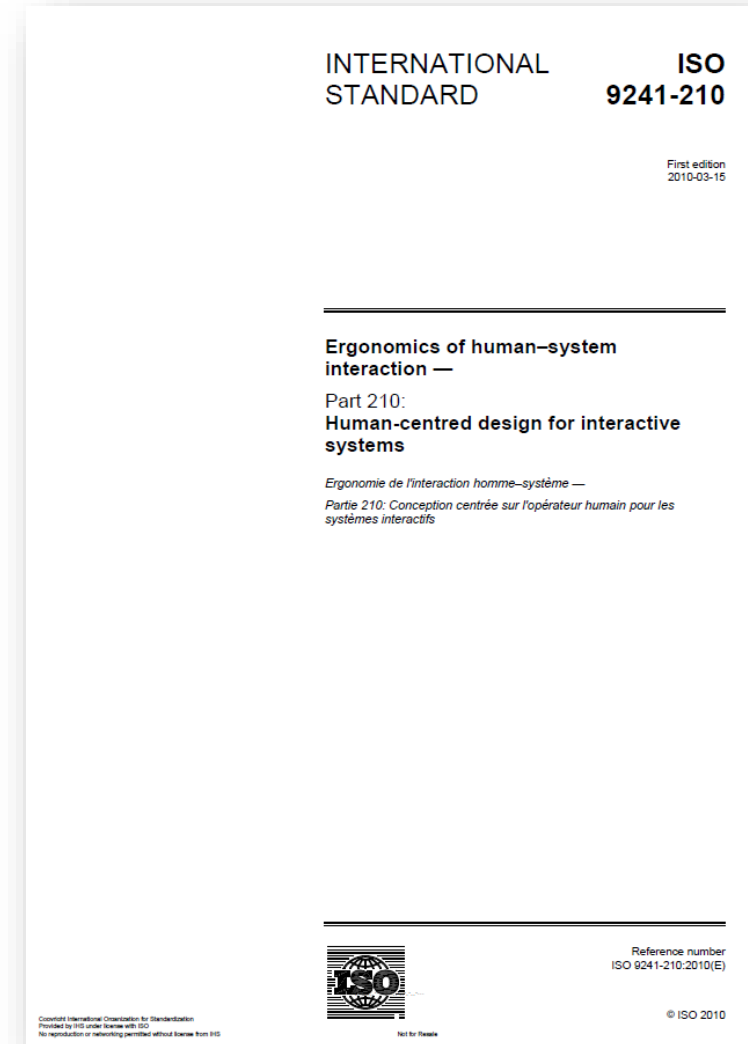
ISO 9241

Ergonomics of human–system interaction

Many parts with specific focus

- Part 1: General introduction
- Part 2: Guidance on task requirements
- Part 3: Visual display requirements
- Part 4: Keyboard requirements
- ...
- Part 110: **Dialogue principles**
- Part 151: Guidance on **World Wide Web** user interfaces
- Part 171: Guidance on software **accessibility**
- Part 210: Human-centered design for interactive systems
- ...
- Part 420: Selection procedures for physical input devices
- Part 910: Framework for tactile and haptic interaction
- Part 920: Guidance on tactile and haptic interactions

ISO 9241 Ergonomics of human–system interaction



ISO 9241

Ergonomics of human–system interaction

Part 210: Human-centered design for interactive systems



Human Centered Design

ISO 9241-210



1

Albrecht Schmidt

ISO 9241 Ergonomics of human–system interaction

ISO 9241

Ergonomics of human–system interaction

Part 110 Dialogue Principles

- Suitability for the task
- Self-descriptiveness
- Controllability
- Conformity with user expectations
- Error tolerance
- Suitability for individualisation
- Suitability for learning

Aufgabenangemessenheit	Selbstbeschreibungsfähigkeit	Steuerbarkeit
Erwartungskonformität	Fehlertoleranz	Individualisierbarkeit
Lernförderlichkeit		

ISO 9241 Ergonomics of human–system interaction

Recommendations, Regulations & Laws

Verordnung über Arbeitsstätten (Arbeitsstättenverordnung - ArbStättV)

1. Allgemeine Anforderungen
2. Maßnahmen zum Schutz vor besonderen Gefahren
3. Arbeitsbedingungen
4. Sanitär-, Pausen- und Bereitschaftsräume, Kantinen, Erste-Hilfe-Räume und Unterkünfte
5. Ergänzende Anforderungen und Maßnahmen für besondere Arbeitsstätten und Arbeitsplätze
- 6. Maßnahmen zur Gestaltung von Bildschirmarbeitsplätzen**
 1. Allgemeine Anforderungen an Bildschirmarbeitsplätze
 2. Allgemeine Anforderungen an Bildschirme und Bildschirmgeräte
 3. Anforderungen an Bildschirmgeräte und Arbeitsmittel für die ortsgebundene Verwendung an Arbeitsplätzen
 4. Anforderungen an tragbare Bildschirmgeräte für die ortsveränderliche Verwendung an Arbeitsplätzen
 5. Anforderungen an die Benutzerfreundlichkeit von Bildschirmarbeitsplätzen

https://www.gesetze-im-internet.de/arbst_ttv_2004/anhang.html

Recommendations, Regulations & Laws

Verordnung über Arbeitsstätten (Arbeitsstättenverordnung - ArbStättV)

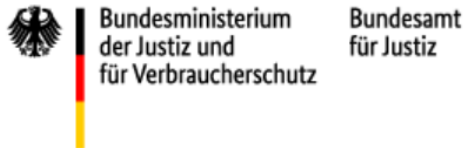
6.5 Anforderungen an die **Benutzerfreundlichkeit** von Bildschirmarbeitsplätzen

- (1) Beim Betreiben der Bildschirmarbeitsplätze hat der Arbeitgeber dafür zu sorgen, dass der Arbeitsplatz den Arbeitsaufgaben **angemessen** gestaltet ist. Er hat insbesondere **geeignete Softwaresysteme** bereitzustellen.
- (2) Die Bildschirmgeräte und die Software müssen **entsprechend den Kenntnissen und Erfahrungen der Beschäftigten** im Hinblick auf die jeweilige **Arbeitsaufgabe angepasst** werden können.
- (3) Das Softwaresystem muss den Beschäftigten **Angaben über die jeweiligen Dialogabläufe** machen.
- (4) Die Bildschirmgeräte und die Software müssen es den Beschäftigten ermöglichen, die **Dialogabläufe zu beeinflussen**. Sie müssen eventuelle Fehler bei der Handhabung beschreiben und eine **Fehlerbeseitigung** mit begrenztem Arbeitsaufwand erlauben.
- (5) Eine Kontrolle der Arbeit hinsichtlich der qualitativen oder quantitativen Ergebnisse darf ohne Wissen der Beschäftigten nicht durchgeführt werden.

https://www.gesetze-im-internet.de/arbsta_tv_2004/anhang.html

Recommendations, Regulations & Laws

Verordnung zur Schaffung barrierefreier Informationstechnik nach dem Behindertengleichstellungsgesetz



Startseite
Gesetze / Verordnungen
Aktualitätendienst
Titelsuche
Volltextsuche
Translations
Hinweise
Impressum
Tastenkombinationen

Verordnung zur Schaffung barrierefreier Informationstechnik nach dem Behindertengleichstellungsgesetz

zur Gesamtausgabe der Norm im Format: [HTML](#) [PDF](#) [XML](#) [EPUB](#)

- [Eingangsformel](#)
- [§ 1 Ziele](#)
- [§ 2 Anwendungsbereich](#)
- [§ 2a Begriffsdefinitionen](#)
- [§ 3 Anzuwendende Standards](#)
- [§ 4 Erläuterungen in Deutscher Gebärdensprache und Leichter Sprache](#)
- [§ 5 Ausschuss für barrierefreie Informationstechnik](#)
- [§ 6 Beratung und Unterstützung durch die Bundesfachstelle für Barrierefreiheit und die Informationstechnik-Dienstleister des Bundes](#)
- [§ 7 Erklärung zur Barrierefreiheit](#)
- [§ 8 Überwachungsverfahren](#)
- [§ 9 Berichterstattung](#)
- [§ 10 Folgenabschätzung](#)
- [Anlage 1 \(weggefallen\)](#)
- [Anlage 2 \(zu § 3 Absatz 2\)](#)

https://www.gesetze-im-internet.de/bitv_2_0/index.html

Recommendations, Regulations & Laws

Verordnung zur Schaffung barrierefreier Informationstechnik nach dem Behindertengleichstellungsgesetz

§ 1 Ziele

- (1) Die Barrierefreie-Informationstechnik-Verordnung dient dem Ziel, eine umfassend und grundsätzlich uneingeschränkt **barrierefreie Gestaltung moderner Informations- und Kommunikationstechnik** zu ermöglichen und zu gewährleisten.
- (2) Informationen und Dienstleistungen öffentlicher Stellen, die elektronisch zur Verfügung gestellt werden, sowie elektronisch unterstützte Verwaltungsabläufe mit und innerhalb der Verwaltung, einschließlich der Verfahren zur elektronischen Aktenführung und zur elektronischen Vorgangsbearbeitung, sind **für Menschen mit Behinderungen zugänglich und nutzbar zu gestalten**

https://www.gesetze-im-internet.de/bitv_2_0/index.html

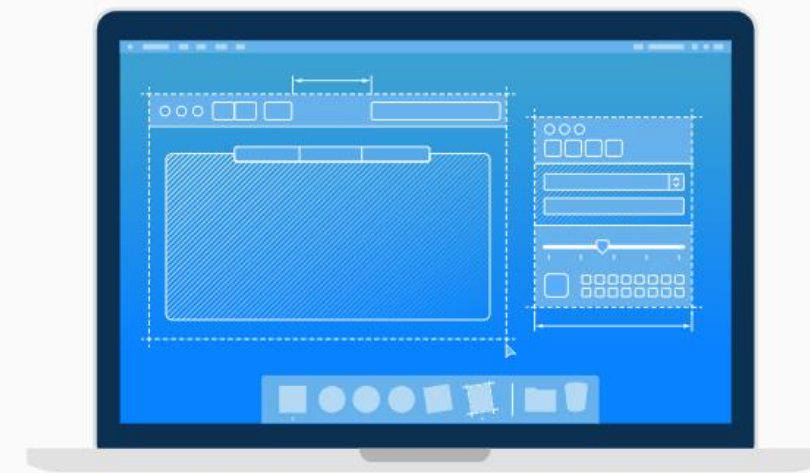
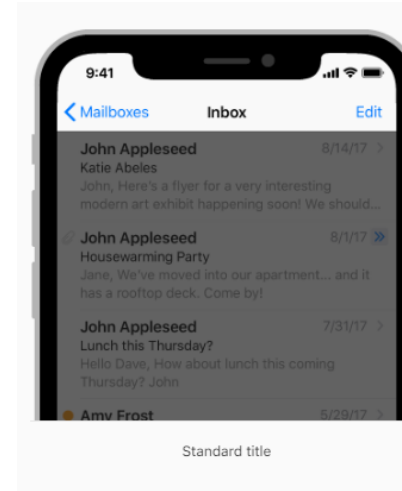
Web Content Accessibility Guidelines (WCAG) 2.1

WCAG 2.1 at a Glance, Summary, www.w3.org/TR/WCAG21

- “Perceivable
 - Provide **text alternatives** for non-text content,
 - Provide captions and other **alternatives for multimedia**.
 - Create content that can be **presented in different ways**, including by assistive technologies, without losing meaning.
 - Make it easier for users to **see and hear** content.
- Operable
 - Make all functionality available from a **keyboard**.
 - Give users **enough time** to read and use content.
 - **Do not use content that causes seizures or physical reactions**.
 - Help users **navigate** and find content.
 - Make it easier to use **inputs other** than keyboard.
- Understandable
 - Make text **readable and understandable**.
 - Make content appear and operate in **predictable ways**.
 - Help users **avoid and correct mistakes**.
- Robust
 - Maximize **compatibility** with current and future user tools.”

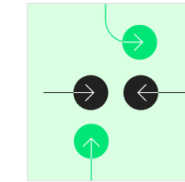
User Interface Guidelines

- Define how the user interface
 - Looks like
 - Is operated
 - Reacts
 - Feels
- Usually most is “encoded” in the libraries, frameworks and UI builder
- Need to learn, how to find it and how to read it



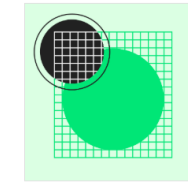
Principles

Gestures help users perform tasks rapidly and intuitively using touch.



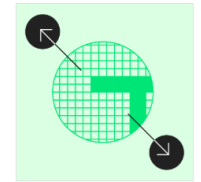
Alternative interaction

Gestures use touch as another way of performing a task.



Easy to use

Users can perform gestures in imprecise ways.



Tactile control

Gestures allow direct changes to UI elements using touch, such as precisely zooming into a map.



Calendar view

September 2014						
Sun	Mon	Tue	Wed	Thu	Fri	Sat
31	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	1	2	3	4

Human Interface Guidelines

Overview Resources Videos What's New

▼ macOS

Themes

Visual Index

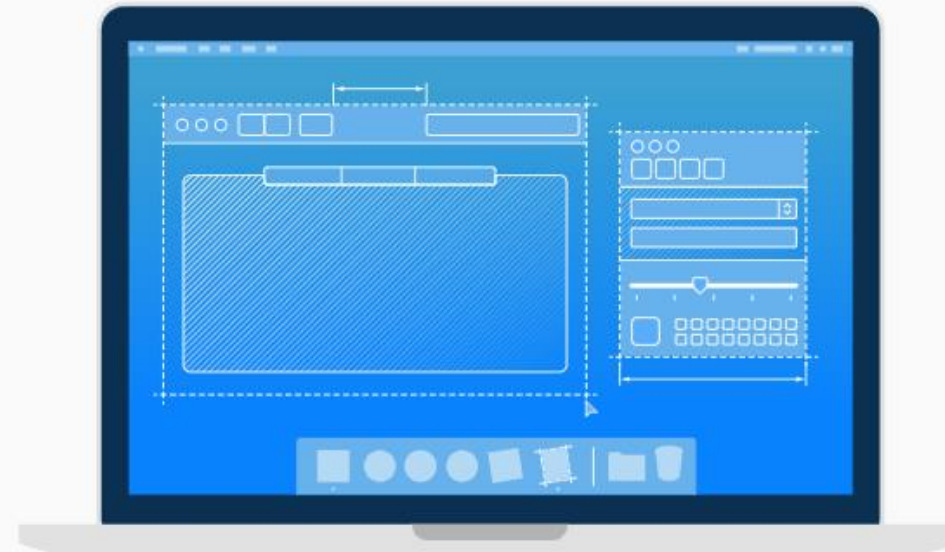
- > App Architecture
- > User Interaction
- > System Capabilities
- > Visual Design
- > Icons and Images
- > Windows and Views
- > Menus
- > Buttons
- > Fields and Labels
- > Selectors
- > Indicators
- > Touch Bar
- > Extensions

iOS

tvOS

watchOS

- > Technologies



macOS Design Themes

Four primary themes differentiate macOS apps from iOS, tvOS, and watchOS apps. Keep these themes in mind as you imagine your app's identity.

<https://developer.apple.com/design/human-interface-guidelines/macos/>

- > macOS
 - > App Architecture
 - > User Interaction
 - > System Capabilities
 - > Visual Design
 - > Icons and Images
 - > Windows and Views
 - > Menus
- ▼ **Buttons**
 - Checkboxes
 - Disclosure Controls
 - Gradient Buttons
 - Help Buttons
 - Image Buttons
 - Pop-Up Buttons
 - Pull-Down Buttons
 - Push Buttons
 - Radio Buttons**
 - Scope Buttons
 - Switches
 - Bevel Buttons*
 - Round Buttons*
- > Fields and Labels
- > Selectors
- > Indicators
- > Touch Bar
- > Extensions

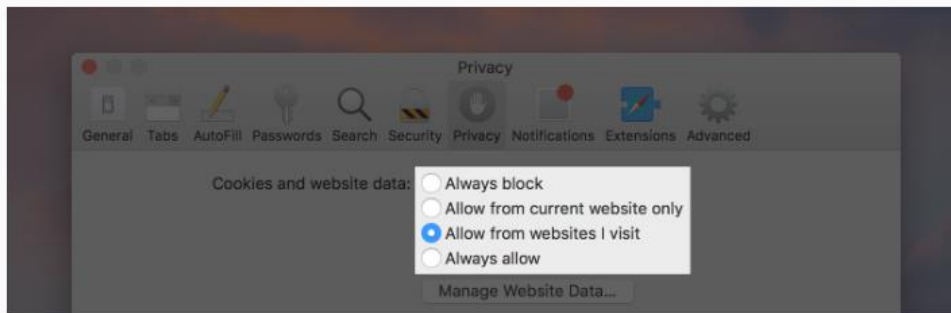
iOS

tvOS

Radio Buttons

A radio button is a small, circular button followed by a title. Typically presented in groups of two to five, radio buttons provide the user a set of related but mutually exclusive choices. A radio button's state is either on (a filled circle) or off (an empty circle).

A radio button can also permit a mixed state (a circle containing a dash) that's partially on and partially off. However, it's better to use checkboxes when your app requires a mixed state.



Give radio buttons meaningful titles. Each radio button's title should clearly describe the effect of choosing it. Generally, use sentence style capitalization without ending punctuation.

Prefer a standard button instead of a radio button to initiate an action. Radio buttons present options to the user. A radio button that initiates an action is confusing and nonintuitive.

Use radio buttons in a view, not a window frame. Radio buttons aren't intended for use within portions of window frames, such as in toolbars and status bars.

Consider using a label to introduce a group of radio buttons. Describe the set of options and align the label's baseline with the baseline of the first radio button's title.

<https://developer.apple.com/design/human-interface-guidelines/macos/>

- > macOS
- > App Architecture
- > User Interaction
- > System Capabilities
- > Visual Design
- > Icons and Images
- > Windows and Views

Menus

Menu Anatomy

Contextual Menus

Dock Menus

Menu Bar Menus

- > Buttons
- > Fields and Labels
- > Selectors
- > Indicators
- > Touch Bar
- > Extensions

iOS

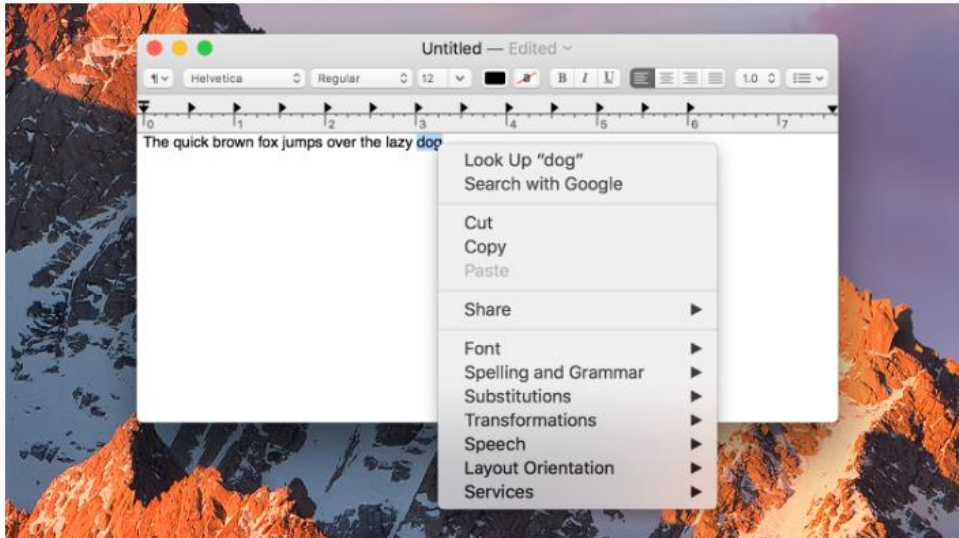
tvOS

watchOS

- > Technologies

Contextual Menus

A contextual menu, or *shortcut menu*, gives people access to frequently used commands related to the current context. A contextual menu is revealed by Control-clicking a view or selected element in an app. For example, Control-clicking selected text in TextEdit displays a contextual menu containing text-specific menu items for initiating actions like changing the font and checking spelling.



Always follow menu design best practices. In general, all menus and menu items should be consistently arranged and titled. See [Menu Anatomy](#).

Include only the most commonly used commands that are appropriate in the current context. For example, in the contextual menu for selected text, it makes sense to include editing commands but it doesn't make sense to include a Save or Print command.

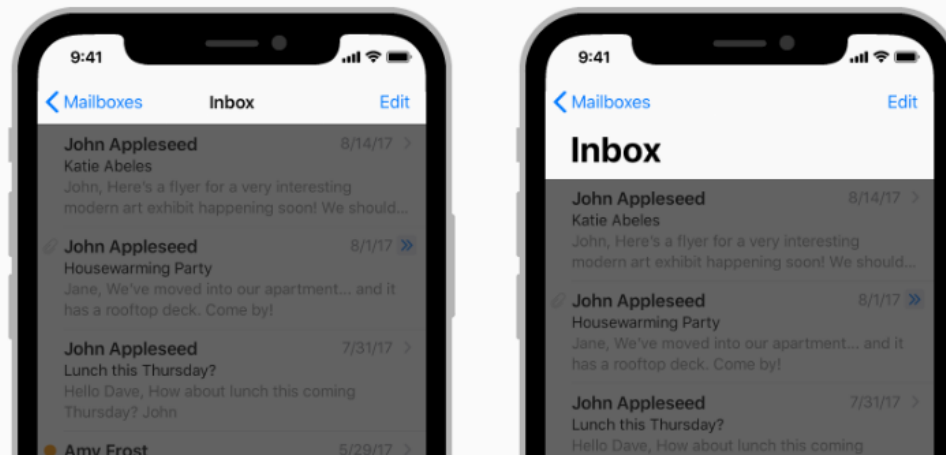
Limit the hierarchical depth of contextual menus to one or two levels. Submenus in contextual menus can be difficult to navigate without accidentally dismissing the contextual

<https://developer.apple.com/design/human-interface-guidelines/macos/>

- > iOS
 - > App Architecture
 - > User Interaction
 - > System Capabilities
 - > Visual Design
 - > Icons and Images
 - ▼ Bars
 - Navigation Bars**
 - Search Bars
 - Status Bars
 - Tab Bars
 - Toolbars
 - > Views
 - > Controls
 - > Extensions
-
- macOS
 - tvOS
 - watchOS
 - > Technologies

Navigation Bar Titles

Consider showing the title of the current view in the navigation bar. In most cases, a title helps people understand what they're looking at. However, if titling a navigation bar seems redundant, you can leave the title empty. For example, Notes doesn't title the current note because the first line of content supplies all the context needed.



Standard title

Large title

Use a large title when you want to provide extra emphasis on context. Large titles should never compete with content, but in some apps, the big, bold text of a large title can help orient people as they browse and search. In a tabbed layout, for example, large titles can help clarify the active tab and indicate when people have scrolled to the top. Phone uses this approach, while Music uses large titles to differentiate content areas like albums, artists, playlists, and radio. In iOS 13 and later, a large title navigation bar doesn't include a background material or shadow by default. Also, a large title transitions to a standard title as people begin scrolling the content. For developer guidance, see [prefersLargeTitles](#).

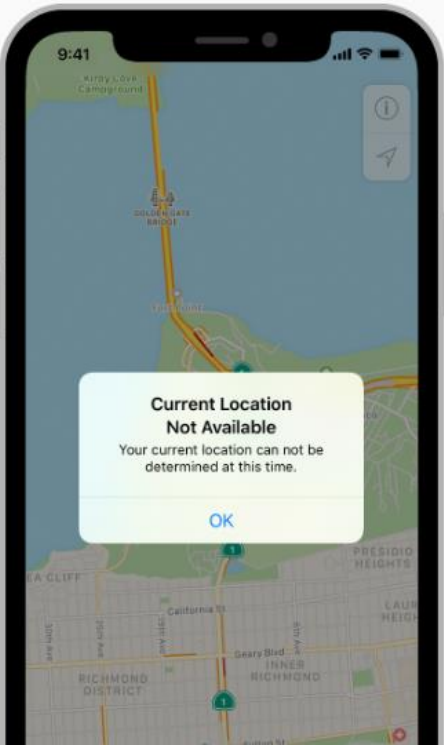
<https://developer.apple.com/design/human-interface-guidelines/ios/views/>

- > iOS
- > App Architecture
- > User Interaction
- > System Capabilities
- > Visual Design
- > Icons and Images
- > Bars
- ▼ Views
 - Action Sheets
 - Activity Views
 - Alerts**
 - Collections
 - Image Views
 - Pages
 - Popovers
 - Scroll Views
 - Split Views
 - Tables
 - Text Views
 - Web Views
- > Controls
- > Extensions

macOS
tvOS

Alerts

Alerts convey important information related to the state of your app or the device, and often request feedback. An alert consists of a title, an optional message, one or more buttons, and optional text fields for gathering input. Aside from these configurable elements, the visual appearance of an alert is static and can't be customized.



<https://developer.apple.com/design/human-interface-guidelines/ios/views/>

- > iOS

> App Architecture

> User Interaction

> System Capabilities

> Visual Design

> Icons and Images

> Bars

< Views

Action Sheets

Activity Views

Alerts

Collections

Image Views

Pages

Popovers

Scroll Views

Split Views

Tables

Text Views

Web Views

> Controls

> Extensions

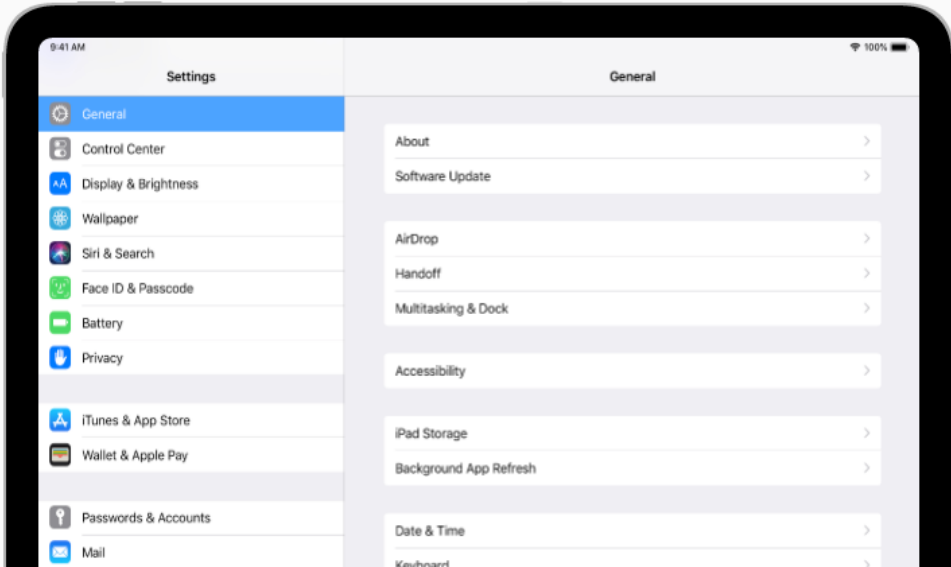
macOS

tvOS

watchOS

Split Views

A split view manages the presentation of two side-by-side panes of content, with persistent content in the primary pane and related information in the secondary pane. Each pane can contain any variety of elements, including navigation bars, toolbars, tab bars, tables, collections, images, maps, and custom views. Split views are often used with filterable content; a list of filter categories appears in the primary pane, and the filtered results for the selected category are shown in the secondary pane. If your app requires it, the primary pane can overlay the secondary pane and can be hidden offscreen when not in use. This is particularly useful when the device is in portrait orientation, as it allows more room for viewing content in the secondary pane. For related guidance, see [Auto Layout](#).



https://developer.apple.com/design/human-interface-guidelines/ios/views/

watchOS

Themes

Apps

Interface Essentials

App Architecture

User Interaction

System Capabilities

Visual Design

Icons and Images

Interface Elements

iOS

macOS

tvOS

Technologies



watchOS Design Themes

As you design your watchOS app, understand the foundations on which Apple Watch itself was designed:

- **Lightweight interactions.** Apple Watch was designed for quick interactions that make the most of the display and its position on the user’s wrist. Information is quick and easy to access and dismiss. The best apps support fast interactions and focus on the content that users care about the most.
- **Holistic design.** Apple Watch was designed to blur the boundaries between device and software. For example, Force Touch and the Digital Crown let users interact seamlessly

<https://developer.apple.com/design/human-interface-guidelines/watchos/overview/themes/>

- > watchOS
- > App Architecture
- > User Interaction
- > System Capabilities
- > Visual Design
- > Icons and Images
- ▼ **Interface Elements**

Alerts and Action Sheets

Buttons

Dates and Timers

Groups

Images

Labels

Menus

Movies

Pickers

Sliders

Switches

Tables

iOS

macOS

tvOS

- > Technologies

Pickers

Pickers display lists of items that are navigable using the Digital Crown. They are meant to be a precise and engaging way to manage selections. Pickers present their items in one of three styles.



▶ Play

List. Displays text and images in a scrolling list. This style displays the selected item and the previous and next items if those items are available.



▶ Play

Stack. Displays images in a card stack style interface. As the user scrolls, images are animated into position with the selected image on top. This style is best for photo browser interfaces.



▶ Play

Sequence. Displays one image from a sequence of images. As the user turns the Digital Crown, the picker displays the previous or next image in the sequence without

<https://developer.apple.com/design/human-interface-guidelines/watchos/overview/themes/>

- > watchOS
- > App Architecture
- > User Interaction
- > System Capabilities
- > Visual Design
- > Icons and Images
- ▼ **Interface Elements**

Alerts and Action Sheets

- Buttons
- Dates and Timers
- Groups
- Images
- Labels
- Menus
- Movies
- Pickers
- Sliders
- Switches
- Tables

- iOS
- macOS
- tvOS
- > **Technologies**

Alerts and Action Sheets

Alerts and action sheets are full-screen system interfaces that you use to convey information and request feedback. Alerts let you display errors or other important information related to the state of your app and its activities. Action sheets let you prompt the user to choose from one of several possible options. Alerts and action sheets are modal interfaces, and you can present them from any of your app’s screens. Alerts and action sheets come in three different styles, and each has a specific use.




Alerts communicate errors or unusual conditions. An alert displays a title, an optional message, and a button to dismiss the sheet. Use the title and message to communicate precisely


Side-by-side alerts communicate errors or unusual conditions where you need to offer a choice between two options. A side-by-side alert displays a title, an optional message,

Action sheets ask the user to select from a set of possible options. An action sheet displays a title, an optional message, and one or more buttons from which to select. One button is

<https://developer.apple.com/design/human-interface-guidelines/watchos/overview/themes/>

 MATERIAL DESIGN

DesignComponentsDevelopResources



Material System

Introduction

Material studies

Material Foundation

Foundation overview

Environment

Layout

Navigation

Color

Typography

Sound

Iconography

Shape


Motion

Interaction

Communication

Design

Create intuitive and beautiful products with Material Design




FOUNDATION
Material dark theme
Learn how to design a dark theme version of your Material UI

POPULAR

[Material Theming](#)

[Iconography](#)

[Text fields](#)



GUIDELINES
Material guidelines
Use sound to communicate information in ways that augment the user experience



Material System

Introduction

Material studies

Material Foundation

Foundation overview

Environment

Layout

Navigation

Color

Typography

Sound

About sound

Applying sound to UI

Sound attributes

Sound choreography

Sound resources

Iconography

Shape

Applying sound to UI

Sound can give expression to interactions and reinforce specific functionality.

CONTENTS

Sound use cases

Hero sounds

Notifications

System sounds

Ambient sounds

Sound use cases

Sound can provide feedback or add decoration to a user experience when applied to strategic moments.

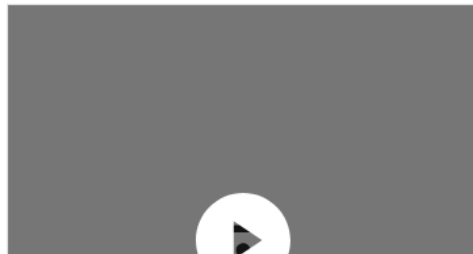
Sound as feedback



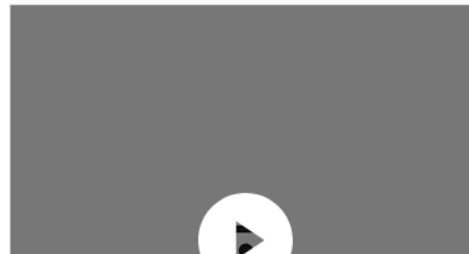
Sound can be applied to interactions by linking an action, or a state change, to an audio cue. These sounds are called **earcons**, and they can represent information, actions, or events. Their sound can reinforce both the meaning of an interaction and a product's aesthetic, emotion, and personality.

To suit an earcon to a particular context, it can be designed using either inspiration from real-world situations or invented specifically to express an abstract concept. Familiar sounds that are based on experiences in the real world are referred to as **skeuomorphic**.

Skeuomorphic sounds



Abstract sounds

<https://material.io/design>



Gestures

Gestures let users interact with screen elements using touch.

CONTENTS

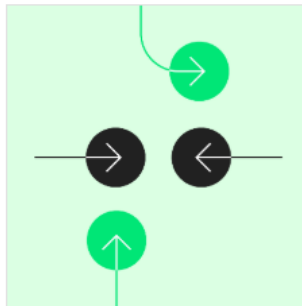
Principles

Properties

Types of gestures

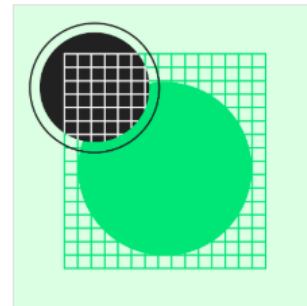
Principles

Gestures help users perform tasks rapidly and intuitively using touch.



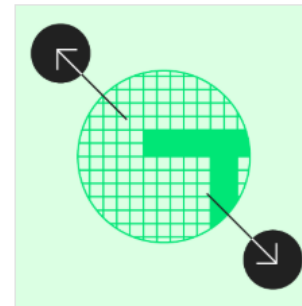
Alternative interaction

Gestures use touch as another way of performing a task.



Easy to use

Users can perform gestures in imprecise ways.



Tactile control

Gestures allow direct changes to UI elements using touch, such as precisely zooming into a map.

Filter by title

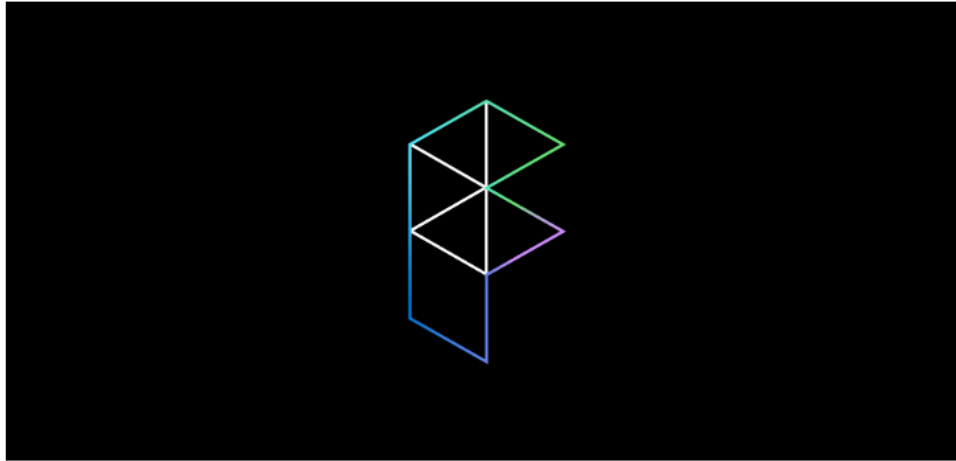
- > What's new
- > Get started
- > Design and UI
 - Overview
 - > Design basics
 - > Layout
 - > Controls
 - > Style
 - > Motion
 - > Shell
 - > Input and interactions
 - > Devices
 - > Usability
 - Design toolkits
- > Develop apps
- > Develop games
- > Publish
- API reference

Design and code Windows apps

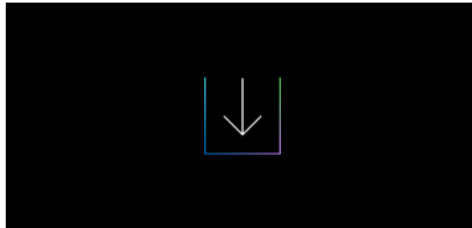
05/06/2019 • 2 minutes to read •

Is this page helpful?

Yes No



Fluent Design guidelines and UI code examples for creating app experiences on Windows 10.



Design downloads

Get design toolkits and samples.

Design basics

An introduction to app design and the Fluent

Download PDF

<https://docs.microsoft.com/en-us/windows/uwp/design/>

Filter by title

- > What's new
- > Get started
- > Design and UI
 - Overview
 - > Design basics
 - > Layout
 - > Controls
 - Overview
 - Intro to controls and events
 - Intro to commanding
 - Index of controls by function
 - > Basic input
 - > Collections
 - > Dialogs and flyouts
 - Forms
 - > Media, graphics, and shapes
 - > Menus and toolbars
 - > Navigation
 - > People
 - > Pickers
 - Color picker
 - Date and time controls**
 - Calendar date picker
 - Calendar view

Download PDF

Calendar, date, and time controls

05/19/2017 • 5 minutes to read •

Date and time controls give you standard, localized ways to let a user view and set date and time values in your app. This article provides design guidelines and helps you pick the right control.

Important APIs: [CalendarView class](#), [CalendarDatePicker class](#), [DatePicker class](#), [TimePicker class](#)

XAML Controls Gallery



If you have the **XAML Controls Gallery** app installed, click here to [open the app and see these controls in action](#).

- [Get the XAML Controls Gallery app \(Microsoft Store\)](#)
- [Get the source code \(GitHub\)](#)

Which date or time control should you use?

There are four date and time controls to choose from; the control you use depends on your scenario. Use this info to pick the right control to use in your app.

Calendar
view

September 2014						
Sun	Mon	Tue	Wed	Thu	Fri	Sat
31	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	1	2	3	4

Use to pick a single date or a range of dates from an always visible calendar.

Is this page helpful?

Yes No

In this article

[Which date or time control should you use?](#)

[Create a date or time control](#)

[Get the sample code](#)

[Related topics](#)

Filter by title

- > What's new
- > Get started
- > Design and UI
 - Overview
 - > Design basics
 - > Layout
- > Controls
 - Overview
 - Intro to controls and events
 - Intro to commanding
 - Index of controls by function
- > Basic input
 - Buttons
 - Check box
 - Combo boxes and list boxes
 - Hyperlinks
 - Radio button
 - Rating control
 - Slider**
 - Toggle
- > Collections
- > Dialogs and flyouts
 - Forms
- > Media, graphics, and shapes
- > Menus and toolbars
- > Navigation
- > People

Sliders

05/19/2017 • 8 minutes to read • 

A slider is a control that lets the user select from a range of values by moving a thumb control along a track.



Get the Windows UI Library



Windows UI Library 2.2 or later includes a new template for this control that uses rounded corners. For more info, see [Corner radius](#). WinUI is a NuGet package that contains new controls and UI features for UWP apps. For more info, including installation instructions, see [Windows UI Library](#).

Platform APIs: [Slider class](#), [Value property](#), [ValueChanged event](#)

Is this the right control?

Use a slider when you want your users to be able to set defined, contiguous values (such as volume or brightness) or a range of discrete values (such as screen resolution settings).

A slider is a good choice when you know that users think of the value as a relative quantity, not a numeric value. For example, users think about setting their audio volume to low or medium—not about setting the value to 2 or 5.

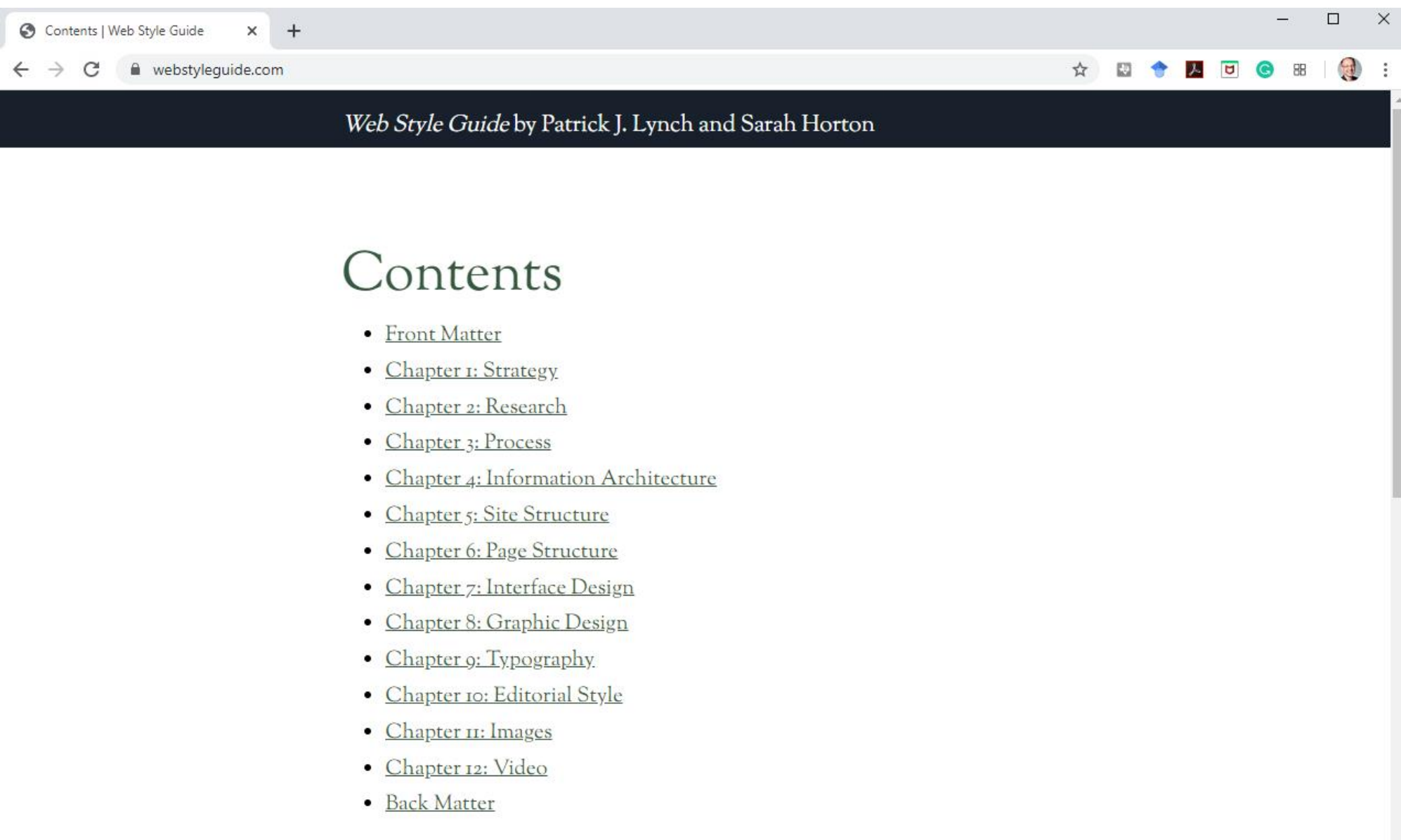
Don't use a slider for binary settings. Use a [toggle switch](#) instead.

Is this page helpful?

 Yes  No

In this article

[Is this the right control?](#)[Examples](#)[Create a slider](#)[Recommendations](#)[Additional usage guidance](#)[Get the sample code](#)[Related topics](#)<https://docs.microsoft.com/en-us/windows/uwp/design/>



<https://webstyleguide.com/>

Apple Human Interface Guidelines (2005)

Icon Genres and Families

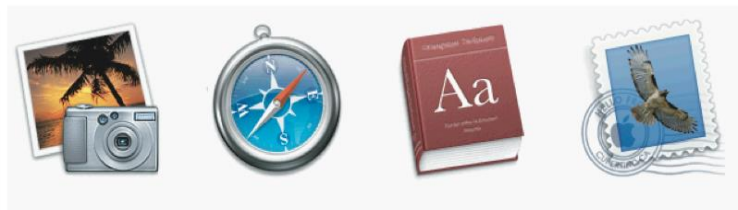
Icon genres help communicate what you can do with an application before you open it. Applications are classified by role—user applications, software utilities, and so on—and each category, or genre, has its own icon style. This differentiation is very important for helping users easily distinguish between types of icons in the Dock.

Figure 10-1 Application icons of different genres—user applications and utilities—shown as they might appear in the Dock



For example, the icons for user applications are colorful and inviting, while utilities have a more serious appearance. Figure 10-2 shows user application icons in the top row and utility icons in the bottom row. These genres are further described in "User Application Icons" (page 125) and "Utility Icons" (page 126).

User application icons in top row



utility icons in bottom row



The diagram shows a standard Mac OS X Trash dialog box. It has a title bar at the top. Below the title bar is a message text area containing the text: "Are you sure you want to erase the items in the Trash permanently using Secure Empty Trash?" and a sub-message: "If you choose Secure Empty Trash, you cannot recover the files." To the left of the message is a square icon with a face, representing the Trash. At the bottom right are two buttons: "Cancel" and "OK".

Labels with leader lines pointing to the dialog box components:

- Message text
- No title
- Informative text
- Application icon
- Cancel button
- Action button

Appointment Preferences

General Calendar Notification Labeling

Daily

Show: 12 hours at a time

Start at: 9:00 AM End at: 6:30 PM

Font: Lucida Grande Size: 13

☒ Split overlapped items

Weekly

Start on: Monday Days per week: 04

Font: Lucida Grande Size: 13

Show: ☒ Holidays ☒ Event times

Scrolling List Specifications

12-point font

19 points baseline to baseline



1 pixel



alligator
American eagle
bicycles
birds - seagulls
chairs
iMac
Pacific Ocean
poppies
seals
Sierra mountains
strawberries
surfers

A diagram of a Safari dialog box with the following dimensions and layout:

- Icon:** A compass icon on the left, with a width of 24 and a height of 64 x 64.
- Title Bar:** A light gray bar at the top with a height of 16.
- Text Area:** Contains the text "Are you sure you want to empty the cache storing the contents of web pages?" and "Safari saves the contents of web pages you open in a cache so that it's faster to visit them again." The text area has a height of 8.
- Buttons:** "Cancel" and "Empty" buttons at the bottom right. The "Empty" button is highlighted in blue. The button area has a height of 10.
- Footer:** A light gray bar at the bottom with a height of 20.
- Spacing:** A gap of 12 between the icon and the text area, and a gap of 24 between the text area and the buttons.

Figure 14-14 Radio button spacing

7 Kind:  Radio off
 Radio on

6  Radio off
 Radio on

7 Kind: ☐ Radio off
☒ Radio on

6 ☐ Radio off
☒ Radio on

5
6 Kind: Radio off
Radio on

5 Radio off
Radio on

UI Guidelines

4.25 Standard shortcut keys

If your application uses any of the standard functions listed in the following tables, use the recommended standard keyboard shortcut for that function.

4.25.1 Standard application shortcuts

Function	Shortcut	Description
Help	F1	Show the help content pages for the current application
Quit	Ctrl+Q	Quit the application

4.25.2 Standard content shortcuts


Function	Shortcut	Description
New	Ctrl+N	Create a new document
Open	Ctrl+O	Open a document
Save	Ctrl+S	Save the current document
Print	Ctrl+P	Print the current document
Close	Ctrl+W	Close the current document

4.25.3 Standard edit shortcuts

Function	Shortcut	Description
Undo	Ctrl+Z	Undo the last operation
Redo	Shift+Ctrl+Z	Redo the last operation

If your application requires both Edit > Find and Edit > Search menu items, use Shift+Ctrl+F as the shortcut for Search.

Incorrect spacing and alignment:



Picture

Name

Angela Avery

Password

.....

Confirm Password

.....


Account Type

Administrator

ON

Permanent Login

Correct spacing and alignment:



Angela Avery

Administrator

Password

.....

Confirm

.....

Permanent Login

ON

18

12

18

18


6

18

6

18

18



Angela Avery

Administrator

Password

.....

Confirm

.....

Permanent Login

ON




<https://developer.gnome.org/hig/stable/visual-layout.html.en>

Touch input

Touch screens are also an increasingly common part of modern computer hardware, and applications created with GTK+ are likely to be used with hardware that incorporates a touch screen. To make the most of this hardware, and to conform to users’ expectations, it is therefore important to consider touch input as a part of application design.

Application touch conventions

Using touch input consistently with other applications will allow users to easily learn how to use your application with a touch screen. The following conventions are recommended, where relevant.

Action	Description	Result
Tap		
	Tap on an item.	Primary action. Item opens — photo is shown full size, application launches, song starts playing.
Press and hold		
	Press and hold for a second or two.	Secondary action. Select the item and list actions that can be performed.
Drag		
	Slide finger touching the surface.	Scrolls area on screen.
Pinch or stretch		

<https://developer.gnome.org/hig/stable/pointer-and-touch-input.html.en>

Edge drag



Slide finger starting from a screen edge.

Top-left edge opens the application menu.
Top-right edge opens the system status menu.
Left edge opens the Activities Overview with the application view visible.

Three finger pinch



Bring three or more fingers closer together while touching the surface.

Öffnet die aktivitäten-Übersicht.

Four finger drag



Drag up or down with four fingers touching the surface.

Wechselt die Arbeitsfläche.

Three finger hold and tap



Hold three fingers on the surface while tapping with the fourth.

Wechselt die Anwendung.

User Interface Checklist

[GNOME Accessibility Developers Guide](#) / [Testing](#)

This section summarizes the guidelines given in [User Interface Guidelines for Supporting Accessibility](#). You should refer to that section of the guide for more detailed information on any of the checklist items given here.

When testing an application for accessibility, you should go through each of the items in the list. Note whether the application passes or fails each test, or does not apply to that application.

Table 2–1 **General Principles checklist**

GP	General Principles	Pass/Fail/NA
GP.1	Every action that alters the user's data or application's settings can be undone.	
GP.2	All application settings can be restored to their defaults without the user having to remember what those defaults were.	
GP.3	After installation, the application can be used without the user having to insert a disk or CD at any time.	
GP.4	The most frequently used functions are found at the top level of the menu structure.	

<https://developer.gnome.org/accessibility-devel-guide/stable/gad-checklist.html.en>

Table 2–2 Keyboard navigation checklist

KN	Keyboard Navigation	Pass/Fail/NA
KN.1	Efficient keyboard access is provided to all application features.	
KN.2	All windows have a logical keyboard navigation order.	
KN.3	The correct tab order is used for controls whose enabled state is dependent on checkboxes, radio buttons or toggle buttons.	
KN.4	Keyboard access to application-specific functions does not override existing system accessibility features.	
KN.5	The application provides more than one method to perform keyboard tasks whenever possible.	
KN.6	There are alternative key combinations wherever possible.	
KN.7	There are no awkward reaches for frequently performed keyboard operations.	
KN.8	The application does not use repetitive, simultaneous keypresses.	
KN.9	The application provides keyboard equivalents for all mouse functions.	
KN.10	Any text or object that can be selected with the mouse can also be selected with the keyboard alone.	
KN.11	Any object that can be resized or moved with the mouse can also be resized or moved with the keyboard alone.	
KN.12	The application does not use any general navigation functions to trigger operations.	
KN.13	All keyboard-invoked menus, windows and tooltips appear near the object they relate to.	

<https://developer.gnome.org/accessibility-devel-guide/stable/gad-checklist.html.en>

Table 2-5 **Fonts and Text checklist**

FT	Fonts and Text	Pass/Fail/NA
FT.1	No font styles or sizes are hard-coded.	
FT.2	An option to turn off graphical backdrops behind text is provided.	
FT.3	All labels have names that make sense when taken out of context.	
FT.4	No label names are used more than once in the same window.	
FT.5	Label positioning is consistent throughout the application.	
FT.6	All static text labels that identify other controls end in a colon (:).	
FT.7	Static text labels that identify other controls immediately precede those controls in the tab order.	
FT.8	An alternative to WYSIWYG is provided. For example, the ability to specify different screen and printer fonts in a text editor.	

Table 2-6 **Color and Contrast checklist**

CC	Color and Contrast	Pass/Fail/NA
CC.1	Application colors are not hard-coded, but are drawn either from the current desktop theme or an application setting.	
CC.2	Color is only used as an enhancement, and not as the only means to convey information or actions.	
CC.3	The application supports all available high-contrast themes and settings.	
CC.4	The software is not dependent on any particular high-contrast themes or settings.	

<https://developer.gnome.org/accessibility-devel-guide/stable/gad-checklist.html.en>

Gnome User Interface Checklist

Table 2-3 Mouse Interaction checklist

MI	Mouse Interaction	Pass/Fail/NA
MI.1	No operations depend on input from the right or middle mouse buttons.	
MI.2	All mouse operations can be cancelled before they are complete.	
MI.3	Visual feedback is provided throughout drag and drop operations	
MI.4	The mouse pointer is never warped under application control, or its movement restricted to part of the screen by the application.	

Table 2-4 Graphical Elements checklist

GE	Graphical Elements	Pass/Fail/NA
GE.1	There are no hard-coded graphical attributes such as line, border or shadow thickness.	
GE.2	All multi-color graphical elements can be shown in monochrome only, where possible.	
GE.3	All interactive GUI elements are easily distinguishable from static GUI elements.	
GE.4	An option to hide non-essential graphics is provided.	

<https://developer.gnome.org/accessibility-devel-guide/stable/gad-checklist.html.en>

Gnome User Interface Checklist

Table 2-7 Magnification checklist

MG	Magnification	Pass/Fail/NA
MG.1	The application provides the ability to magnify the work area.	
MG.2	The application provides the option to scale the work area.	
MG.3	The application's functionality is not affected by changing the magnification or scale settings.	

Table 2-8 Audio checklist

AU	Audio	Pass/Fail/NA
AU.1	Sound is not used as the only means of conveying any items of information.	
AU.2	The user can configure the frequency and volume of all sounds and warning beeps.	

Table 2-9 Animation checklist

AN	Animation	Pass/Fail/NA
AN.1	There are no flashing or blinking elements with a frequency greater than 2Hz or lower than 55Hz.	
AN.2	Any flashing or blinking is confined to small areas of the screen.	
AN.3	If animation is used, an option is available to turn it off before it is first shown.	

<https://developer.gnome.org/accessibility-devel-guide/stable/gad-checklist.html.en>

Gnome User Interface Checklist

Table 2-10 Keyboard Focus checklist

KF Keyboard Focus		Pass/Fail/NA
KF.1	When a window is opened, focus starts at the most commonly-used control.	
KF.2	Current input focus position is clearly displayed at all times.	
KF.3	Input focus is shown in exactly one window at all times.	
KF.4	Appropriate audio or visual feedback is provided when the user attempts to navigate past either end of a group of related objects.	
KF.5	The default audio or visual warning signal is played when the user presses an inappropriate key.	
KF.6	There is sufficient audio information for the visual focus that the user can figure out what to do next.	
KF.7	When using assistive technologies, such as a screen reader or braille device, the current program indicates the position and content of the visual focus indicator.	

Table 2-11 Timing checklist

TM Timing		Pass/Fail/NA
TM.1	There are no hard-coded time-outs or time-based features in the application.	
TM.2	The display or hiding of important information is not triggered solely by movement of the mouse pointer.	

<https://developer.gnome.org/accessibility-devel-guide/stable/gad-checklist.html.en>



Did you understand this block?

Can you answer these questions?

- What are the main recommendations in the 4 main recommendations in the Web Content Accessibility Guidelines (WCAG) 2.1
- What is the common structure for a user interface guideline? What are typical parts that are described?
- Find where the following is described described.
 - How do you design buttons on the apple watch?
 - What recommendations are there for icon design for Windows?
 - What are the font recommendations for Material Design?

Reference

- Hix and Hartson, Developing User Interfaces, Wiley, 1993
- ISO 9241 Ergonomics of human–system interaction
- Verordnung über Arbeitsstätten (Arbeitsstättenverordnung - ArbStättV) https://www.gesetze-im-internet.de/arbst_ttv_2004/anhang.html
- Verordnung zur Schaffung barrierefreier Informationstechnik nach dem Behindertengleichstellungsgesetz https://www.gesetze-im-internet.de/bitv_2_0/index.html
- Web Content Accessibility Guidelines (WCAG) 2.1 www.w3.org/TR/WCAG21
- <https://developer.apple.com/design/human-interface-guidelines/macos/>
- <https://developer.apple.com/design/human-interface-guidelines/ios/views/>
- <https://developer.apple.com/design/human-interface-guidelines/watchos/overview/themes/>
- <https://material.io/design>
- <https://docs.microsoft.com/en-us/windows/uwp/design/>
- <https://webstyleguide.com/>
- Apple Human Interface Guidelines (2005), Apple Computer, Inc
- <https://people.gnome.org/~fpeters/hig3.pdf>
- <https://developer.gnome.org/hig/stable/pointer-and-touch-input.html.en>
- <https://developer.gnome.org/accessibility-devel-guide/stable/gad-checklist.html.en>

License

This file is licensed under the Creative Commons Attribution-Share Alike 4.0 (CC BY-SA) license:

<https://creativecommons.org/licenses/by-sa/4.0>

Attribution: Albrecht Schmidt

For more content see: <https://hci-lecture.de>

