



## 1. SYSTÈME D'ESPACEMENT (Spacing)

[UNIVERSAL] Toutes plateformes adoptent une **grille de 4 unités** (4px, 4dp, 4pt) comme incrément de base [1](#) [2](#). Ainsi les valeurs de spacing usuelles sont des multiples de 4 (ex. 4, 8, 12, 16, 24, 32, 48, 64...).

[WEB] Échelle de spacing typique (ex. Tailwind par défaut): 1=4px, 2=8px, 3=12px, 4=16px, 5=20px, 6=24px, 8=32px, 10=40px, 12=48px, 16=64px, 20=80px, 24=96px, etc [2](#). Les **marges de page** sont souvent : ~12px sur mobile, ~32px sur tablette et ~80px sur desktop (avec largeur de contenu max ~1120px) [3](#). Les **padding internes** des composants reprennent cette échelle (ex. un bouton text min. 8px haut/bas et 16px gauche/droite). Les **gaps** entre éléments (listes, grilles) suivent aussi ces multiples de 4 (ex. gap de 8px ou 16px). Espacement vertical entre sections souvent 64-128px sur desktop (scaling de l'échelle de 4).

[ANDROID] Matérial Design 3 impose un pas de **4dp** pour tous les espacements [1](#). Par exemple, les marges et padding se font en 4dp, 8dp, 12dp, 16dp, 24dp, 32dp... Un « spacer » standard entre grandes sections vaut 24dp [4](#). Les **marges de bord d'écran** peuvent être de 16dp en mobile, élargies à 24dp+ sur tablette (pour plus d'ouvertures).

[iOS] HIG/iOS recommande un pas de **8pt** (équivalent ~8px) comme grille de base. Les espacements usuels sont 8, 16, 24, 32, 40, 48pt, etc [5](#). Par défaut, les vues ont 16pt de marge intérieure sur les côtés (layout margins) [6](#). Les composants (boutons, champs, cartes) utilisent des padding internes typiquement 8-16pt selon le contexte. Les targets tactiles sont  $\geq 44 \times 44$ pt (ce qui impose au moins 44pt de spacing minimal) [5](#).

## 2. TYPOGRAPHIE

[WEB] **Police système** : utiliser la police système (`font-family: system-ui, -apple-system, ...`), ce qui sous-entend l'usage de polices natives (Arial/Helvetica sur Windows/Mac, Roboto sur Android) pour de meilleures performances et cohérence. Échelle typographique courante (classes Tailwind) – *text-sm*: 14px, *text-base*: 16px, *text-lg*: 18px, *text-xl*: 20px, *text-2xl*: 24px, *text-3xl*: 30px, *text-4xl*: 36px, *text-5xl*: 48px [7](#) [8](#), *text-6xl*: 60px, *text-7xl*: 72px, *text-8xl*: 96px [9](#). Line-heights associées typiquement 1.5x (par ex. 16px → 24px) [7](#), letter-spacing normal (tracking-normal 0em) ou très léger (tailwind dispose de classes comme *tracking-tight* = -0.025em). Poids de police usuels : Regular (400) pour le texte courant, SemiBold (600) ou Bold (700) pour titres/CTA.

[ANDROID] **Police système** : Roboto (condensed) ou Roboto par défaut (Google Sérit dans M3), parfois remplacée par Google Sans. Matérial Design 3 définit 15 styles typographiques organisés en cinq rôles (Display, Headline, Title, Body, Label) chacune en Large/Medium/Small [10](#). Par exemple : *displayLarge* = 57sp (line-height 64sp), *displayMedium* = 45sp/52sp, *displaySmall* = 36sp/44sp; *headlineLarge* = 32sp/40sp, *headlineMedium* = 28sp/36sp, *headlineSmall* = 24sp/32sp; *titleLarge* = 22sp/28sp (Roboto Medium), *titleMedium* = 16sp/24sp, *titleSmall* = 14sp/20sp; *bodyLarge* = 16sp/24sp, *bodyMedium* = 14sp/20sp, *bodySmall* = 12sp/16sp; *labelLarge* = 14sp/20sp (Medium), *labelMedium* = 12sp/16sp (Medium), *labelSmall* = 11sp/16sp (Medium) [10](#). Les poids typiques sont Regular (400) pour le texte normal, Medium (500) ou SemiBold (600)

pour titres et labels. Letter-spacing : souvent 0sp pour la majorité, quelques styles (par ex. titleMedium) ont +0.15sp <sup>11</sup>.

[iOS] **Police système** : San Francisco (SF Pro) by default. Apple définit des styles d'interface prédéfinis. Par défaut au niveau standard (Dynamic Type « Default »), on a : *Large Title* = 34pt, *Title1* = 28pt, *Title2* = 22pt, *Title3* = 20pt; *Headline* = 17pt, *Body* = 17pt; *Callout* = 16pt, *Subhead* = 15pt; *Footnote* = 13pt, *Caption1* = 12pt, *Caption2* = 11pt <sup>12</sup> <sup>13</sup>. Les line-heights associés sont par exemple 41pt pour 34pt (LargeTitle), 34pt pour 28pt (Title1), 32pt pour 22pt, 25pt pour 20pt, 22pt pour 17pt, etc. (en pratique ~1.2-1.5× la taille de police). Letter-spacing : proche de 0 (légèrement compressé sur les textes UI). Poids standards : Regular pour le corps de texte, Semibold/Bold pour Headline/Titles (éviter UltraLight/Thin pour l'accessibilité <sup>14</sup>).

## 3. PALETTE COULEURS

[UNIVERSAL] Les interfaces doivent respecter **les contrastes WCAG** : ratio  $\geq 4.5:1$  (niveau AA) pour texte normal,  $\geq 3:1$  pour texte large <sup>14</sup>. Les couleurs d'état communes incluent: *Error* (rouge, ex. #B00020 en Material), *Warning* (orange), *Success* (vert #4CAF50), *Info* (bleu #2196F3). Les états désactivés utilisent ~0.38 d'opacité (Material) ou 50% (web) sur la couleur de base. Les overlays (backdrop) sont typiquement noir avec opacité ~30-60%.

[WEB] Le web n'impose pas de palette standard, mais utilise souvent les classes utilitaires (Tailwind, Bootstrap) pour *primary/secondary/surface*. Par exemple, Tailwind propose des couleurs primaires (bleu-500, vert-500...) ajustables <sup>2</sup>. En design système, on définit typiquement : primary (ex. #3B82F6), secondary (ex. #F59E0B), background (blanc ou gris très clair), surface (blanc), etc. Les lois WCAG 2.2 s'appliquent (minimum 4.5:1) pour textes standards; le mode sombre inverse souvent arrière-plans et textes (texte clair sur fond sombre).

[ANDROID] Material 3 utilise un système de tokens de couleur dynamiques : *primary*, *onPrimary*, *primaryContainer*, *secondary*, *error*, *background*, *surface*, etc., avec correspondance de contrastes élevées <sup>10</sup> (contraste minimum 4.5:1 pour « on »). Exemple de couleurs par défaut Material 3 light : Primary = #6750A4, Secondary = #625B71, Error = #B3261E, Success = #4CAF50, etc. (les valeurs exactes varient selon le thème). Les composants désactivés utilisent 38% d'opacité sur la couleur de fond (Material). En mode sombre, on inverse généralement les tonalités : les couleurs primaires s'éclaircissent et surfaces foncent (ex. primary clair vs foncé).

[iOS] Apple HIG définit des **couleurs système** : *label* (texte principal, adaptatif noir/blanc), *secondaryLabel*, *systemBackground*, *secondarySystemBackground*, etc., qui s'adaptent clair/sombre. On choisit *AccentColor* (couleur principale) pour liens/éléments d'action. Les couleurs d'état courantes : Error = systemRed, Success = systemGreen, Warning = systemOrange (ex. #FF9500) ; ces *system colors* garantissent contraste auto. En mode sombre, les couleurs système sont inversées (p. ex. *label* devient blanc, *systemBackground* devient noir).

## 4. COMPOSANTS UI

[Note : les dimensions exactes varient selon la bibliothèque, voici les valeurs usuelles recommandées.]

[WEB] **Boutons** – Hauteur minimale ~32px (voire 40px pour le style Material), largeur minimale 64px. *Border-radius* typique : 4px (Matérial) ou 8px (plus arrondi). États : default (fond et texte), *hover* (fond foncé/ blanc), *active* (léger décroché), *focus* (anneau bleu 2px), *disabled* (opacité 50%). Touch target min 48×48px (recommandé). Élévation ombre optionnelle (shadow 0 1px 2px rgba).

**Champs de saisie** – Hauteur ~40px, bordure 1px #ccc, *padding* interne ~8px, border-radius ~4px. États : *focus* (bordure bleue, ombre interne), erreur (bordure rouge), *disabled* (fond gris), etc.

**Cartes** – Padding interne 16px, radius ~8px, ombre (box-shadow) légère (ex. 0 1px 3px rgba(0,0,0,0.1)). Min width/height : variable.

**Listes/Items** – Gap vertical 8–16px. Items minimum 44px tall (pour accessibilité tactile). Bords arrondis ~4px.

**Navigation** – Barre d'app (header) hauteur ~56px mobile, ~64px desktop. *Tabs* hauteur 48px, *bottom nav* 56dp tall, *sidebar* largeur 240px desktop. **Nav links** large (min 48×48px).

**Modals/Dialogs** – Taille min 300px×200px, max 90% écran. *Padding* interne ~24px. *Border-radius* 8px. Fond d'écran modal semi-transparent (~rgba(0,0,0,0.5)).

**Chips/Tags** – Hauteur 24px (avec padding 12px), radius 12px (ou fully round). États : default (fond gris clair), selected (fond principal), *disabled* (opacity).

**Toggles/Checkboxes/Switches** – Touch target 44px. Switches dimension ~50×30px (toggle circle ~24px). Checkbox ~20px.

[ANDROID] (Material 3 specs) :

- **Bouton (Button, FAB)** : Hauteur min 40dp (text button) ou 56dp (fab). Padding horizontal ~16dp. Border-radius = 12dp (vos/Material defaults) ou 20dp pour FAB. États : Pressed (elevation 2dp), Focused (elevation 4dp), Disabled (contenu 38% opacité) <sup>10</sup>. Touch cible = 48×48dp minimum.
- **Champs** : TextField hauteur ~56dp, padding ~16dp, border-radius 4dp, focus = outline bleu de 2dp.
- **Cards** : Élévation 1dp, padding interne 16dp, radius 12dp.
- **Listes** : Items min 48dp, dividers 1px.
- **Navigation** : AppBar hauteur 56dp (mobile) / 64dp (tablet). BottomNav Bar 56dp. NavigationRail ~72dp largeur.
- **Modals/Sheets** : arrondis ~16dp haut, up to full width on mobile; backgrounds overlays scrim noir scrim alpha ~0.32.
- **Chips** : 32dp height, 8dp horizontal padding, radius 16dp. Colors variant primary/outline.
- **Toggles** : Switch width 52dp, height 32dp, track radius 16dp, thumb 20dp. Checkbox 18dp with 2dp stroke.

[iOS] (UIKit/HIG defaults) :

- **Bouton** : Min cible 44×44pt. UIKit buttons auto-padding ~8pt vertical, 16pt horizontal. *Corner radius* ~8pt (capsule style use 21pt for 42pt height). Hover/press (iPadOS) = alpha 0.3, focus = blue glow. Disabled = 50% alpha.
- **Text Field** : Height ≈ 30–40pt, *cornerRadius* ~8pt, border 1pt. Focus = blue border, disabled = grayed.
- **Card** : radius 8–12pt, shadow (opacity ~0.1, radius 4pt).

- **List/Cell** : Cell height  $\geq 44\text{pt}$ . Separator thickness 1px.
- **Navigation Bar/Tab Bar** : Height  $\sim 44\text{pt}$  (bar) + status bar 20pt, Tab Bar 49pt.
- **Modal** : Default present = full screen or sheet with rounded top corners (16pt). Background dim=black 50%.
- **Chip/Tag** (Badge) : font Caption2 (11pt) + padding (4pt). Capsule radius = half height.
- **Switch** : Height 31pt, width 51pt (plus system insets). Checkbox (iOS uses switches).

## 5. GRILLE ET LAYOUT

[WEB] Utiliser grilles flexibles responsives (CSS Grid/Flexbox). Grille 12 colonnes est courante (Bootstrap, Tailwind) <sup>15</sup>. **Breakpoints** usuels (Tailwind v4) : sm~480px, md~768px, lg~1024px, xl~1280px, 2xl~1536px. **Colonnes** : 12 colonnes en desktop, 6 en tablette, 4 en mobile. **Gouttière** : entre 16px et 24px horizontal. **Marges extérieures** : container responsive avec padding  $\sim 16\text{px}$  small, 32px medium, 80px large (cf. section 1). **Safe areas** : sur mobile, éviter notch/indicateur d'accueil en utilisant `env(safe-area-inset-*)`.

[ANDROID] Matériel : Layout adaptatif avec trois classes de taille (Compact, Medium, Expanded) <sup>16</sup>. Largeurs de fenêtres : compact <600dp, medium 600–840dp, expanded >840dp. **Canonical layouts** (colonne simple ou multi-panels). Gutter standard 16dp (mobile) à 24dp (tablet). Marges adaptatives : e.g. 16dp minidesktop, 32dp large. Respecter `safe-area` (barres système), p. ex. statut 24dp tall.

[iOS] HIG : Appliquer *Safe Area* automatiquement (top notch, bottom home indicator). **Colonnes** : variable (SplitView sur iPad, 1-2 colonnes). Apple donne peu de contraintes chiffrées, mais suggère par ex. 8pt-16pt gutters. **Gutters** horizontaux 16–20pt. **Marge extérieures** : contenu principal souvent  $\sim 20\text{pt}$  des bords (cf. Layout Margins par défaut <sup>6</sup>). Points de rupture entre adaptatifs (Split view vs fullscreen) dépendent d'UIWindow width classes (Compact/Regular).

## 6. MOTION ET ANIMATION

[UNIVERSAL] **Durées standard** : rapide (~100–150ms), moyen (200–300ms), long (>400ms) <sup>1</sup>. **Courbes d'accélération** : ease-in, ease-out, ease-in-out (par ex. CSS `cubic-bezier(0.4, 0, 0.2, 1)` dans Material). Transitions usuelles : 200ms pour hover, 300ms pour navigation entre écrans. Micro-interactions (bouton press, toggle)  $\sim 100\text{--}150\text{ms}$  de feedback.

[ANDROID] Matériel Motion recommande : rapide 75–150ms, normal 200–300ms, lente >500ms. Easing *accélération/décélération naturelle* (Standards : decelerate at end). **Transitions** : composants Material (Buttons, Cards) ont ombre et couleur qui s'animent  $\sim 100\text{--}200\text{ms}$ . Feedback haptique est couplé.

[iOS] HIG Motion indique : *short*  $\approx 200\text{ms}$ , *medium*  $\approx 400\text{ms}$ , *long*  $\approx 700\text{ms}$ . Courbes *ease-in-out* (UIKit default `UIViewControllerAnimatedCurveEaseInOut`). Animations de présentation de vue  $\sim 300\text{--}500\text{ms}$ . Micro-interactions (switch toggle)  $\sim 100\text{ms}$ . Préférer `UIViewControllerAnimatedProperty` pour spring animations (effet de rebond doux).

## 7. ICONOGRAPHIE

[WEB] **Taille standard** : 24px ou 32px pour icônes inline, 16px pour petites. Stroke width ~2px (ou  $0.125em$ ). Assurer 44px de zone cliquable. Style : contour (outline) ou rempli selon contexte. Ex. Heroicons recommande 24x24px, stroke 2px.

[ANDROID] **Material Icons** (« Material Symbols ») : icônes recommandées en 24dp (pour toolbar/action bar) ou 48dp (touch targets). Stroke poids 2dp pour outlined. *Optical sizing* : Material SF Symbols adaptent taille variable en iOS; sur Android utiliser l'icône vectorielle fournie. Touch target min. 48x48dp (icone 24dp centrée). Styles : Filled (standard), Outlined (stroke 2dp), Rounded, Sharp.

[iOS] **SF Symbols** : taille typique 20pt (bar button), 34pt (tab bar), 44pt (touch targets). Les symboles SF adaptent leur épaisseur automatiquement. Touch target 44x44pt minimum (zone dans l'UI). Utiliser variant *regular/medium* correspondante à text style. SF Symbols sont vectoriels, s'adaptent dynamique.

## 8. ACCESSIBILITÉ

[UNIVERSAL] Touchez et cibles : min. **44x44px/dp** <sup>5</sup>. Contraste : WCAG AA  $\geq 4.5:1$  normal, AAA  $\geq 7:1$ . Focus indicateurs : visible (outline, ombre) pour clavier/talkback. VoiceOver/TalkBack : éléments avec labels explicites (`aria-label`, `accessibilityLabel`). Support de *reduced motion* : si utilisateur désactive animations (pref OS), proposer des transitions simplifiées (fade simple).

[WEB] Interactions au clavier (outline 2px noir/bleu sur focus). `alt` sur images, `aria-*` roles sur composants.

[ANDROID] Touch targets  $\geq 48\times 48dp$ , space  $\geq 8dp$  entre cibles <sup>5</sup>. Charactères textuels = rem (scalable). Ensure `contentDescription` for icons. States (focus, pressed) accessibles via Material states.

[iOS] Touch targets  $\geq 44\times 44pt$  <sup>5</sup>. Labels VoiceOver pour chaque contrôle (`accessibilityLabel`). Dynamic Type support (utiliser Text Styles). Focus ring par défaut bleu sous VoiceOver (Smart Invert possible). Respect Safe Area.

**Sources** : Documentation officielle des plateformes et bibliothèques mentionnées <sup>2</sup> <sup>1</sup> <sup>3</sup> <sup>7</sup> <sup>10</sup> <sup>12</sup>  
<sup>13</sup> <sup>14</sup> . (Date de collecte : Février 2026)

---

<sup>1</sup> <sup>4</sup> Spacing | Understanding layout | Made with Supernova

<https://greenhousedl.supernova-docs.io/latest/foundations/layout/understanding-layout/spacing-izOCyQY6>

<sup>2</sup> Tailwind Padding Explained: How to Perfect Your Spacing | Tailkits

<https://tailkits.com/blog/tailwind-padding-guide/>

<sup>3</sup> <sup>15</sup> Responsive Web Design, Part 1: A Simple Layout Formula | by brianonchain | Medium

<https://medium.com/@brianonchain/rwd-a-simple-responsive-web-design-formula-0ef091c29b18>

5 gui design - Standard margins in iOS for info display - User Experience Stack Exchange  
<https://ux.stackexchange.com/questions/76871/standard-margins-in-ios-for-info-display>

6 Notes from the margins  
<https://martiancraft.com/blog/2019/02/notes-from-the-margins/>

7 8 9 font-size - Typography - Tailwind CSS  
<https://tailwindcss.com/docs/font-size>

10 11 Material Design 3 in Compose | Jetpack Compose | Android Developers  
<https://developer.android.com/develop/ui/compose/designsystems/material3>

12 iOS Font Sizes | Preferred font sizes on iOS devices  
<https://www.iosfontsizes.com/>

13 iOS default font sizes - also available on <https://www.iosfontsizes.com> · GitHub  
<https://gist.github.com/zacwest/916d31da5d03405809c4>

14 What are Apple's guidelines on typography for iOS apps?  
<https://median.co/blog/apples-ui-dos-and-donts-typography>

16 Design an Adaptive Layout with Material Design | Android Developers  
<https://developer.android.com/codelabs/adaptive-material-guidance>