



UX/UI per Mobile

app intuitive e piacevoli da usare

Dalla prima schermata al gesto finale: cosa rende un'app davvero semplice, veloce e amata dagli utenti



Antonello Carlomagno

WAU73 SRL – CEO - Founder



carlomagno.dev



a.carlomagno@wau73.com



linkedin.com/in/antonellocarlomagno



@acarlomagno



19 Novembre 2025
Padova





Software & App

Miglioriamo il tuo modo di lavorare con Software e App personalizzati

Branding & digital marketing

Ti aiutiamo a comunicare meglio chi sei e cosa fai.

Progettazione logo e brand identity, allestimenti eventi e fiere. Realizzazione siti e ottimizzazione SEO.

Consulenza formativa

Ci occupiamo del tuo aggiornamento professionale.

wau73.com





scopri | parti | condividi

Ci muove una passione profonda per il viaggio, un carburante che ci accende e ci tiene vivi. Proviamo l'impulso irrefrenabile di viaggiare, siamo una redazione con lo zaino in spalla

www.ilturista.info



19 Novembre 2025
Padova



AGENDA



- UX, UI e CX - Tre Pilastri
- 5 Leggi UX Fondamentali
- Operazioni Sincrone vs Asincrone
- Pattern Asincroni per UX Eccellente
- Navigazione Confusa vs Chiara
- Gerarchia Visiva Assente vs Chiara
- Form Complessi vs Ottimizzati
- Esempi mockup
- Strumenti x mockup
- Touch Target Inadeguati vs Ottimizzati
- Psicologia del Colore
- Microinterazioni
- Architettura e Navigazione
- Onboarding e FTUX (First Time User Experience)
- 10 Errori UX Comuni
- Checklist Definitiva
- Come lavorare in Delphi ?
- Q&A



UX/UI +CX per Applicazioni Mobili

Dalla Teoria alla Pratica

Un viaggio completo attraverso i principi fondamentali
dell'esperienza utente nelle applicazioni mobili moderne



UX, UI e CX: Tre Pilastri dell'Esperienza Digitale

User Experience (UX)

L'esperienza complessiva che un utente vive quando interagisce con un prodotto digitale. Comprende l'intero percorso dell'utente dalla scoperta dell'app all'utilizzo quotidiano fino al supporto post-vendita. Una buona UX anticipa i bisogni dell'utente, rimuove ostacoli e crea un'esperienza fluida e intuitiva che va oltre l'estetica.

User Interface (UI)

Il punto di contatto visivo e interattivo tra l'utente e l'applicazione. Include tutti gli elementi grafici: pulsanti, icone, tipografia, colori, spaziature e animazioni. Una UI efficace non è solo bella da vedere, ma guida l'utente attraverso l'interfaccia in modo chiaro e coerente, rispettando i pattern mentali consolidati.

Customer Experience (CX)

L'intera relazione tra cliente e brand, includendo tutti i touchpoint: marketing, vendite, utilizzo del prodotto, assistenza clienti e fidelizzazione. Mentre UX si concentra sul prodotto, CX considera l'ecosistema completo dell'interazione brand-cliente attraverso ogni canale e momento di contatto.

Relazione tra i Concetti

La CX contiene la UX, che a sua volta include la UI. Un'app può avere una UI bellissima ma una UX frustrante se la navigazione è confusa. Allo stesso modo, un'ottima UX può essere compromessa da una CX negativa se il servizio clienti è inefficiente.

Esempio Pratico: App di Banking

UI elegante con colori professionali e icone chiare + UX eccellente con trasferimenti in 3 tap e riconoscimento facciale rapido + CX superiore con onboarding semplificato, supporto chat 24/7 e notifiche proattive su transazioni sospette = Esperienza completa di successo.

5 Leggi UX che Ogni Designer Deve Conoscere

1. Soglia di Doherty (Doherty Threshold)

Quando il sistema risponde in meno di 400ms, l'utente percepisce l'interazione come istantanea. Nelle app mobili: ottimizzare tempi di caricamento, utilizzare skeleton screens e feedback immediati. Esempio: Instagram mostra il layout mentre carica le immagini.

2. Legge di Hick (Hick's Law)

Il tempo per decidere aumenta con il numero di opzioni. Più scelte offri, più l'utente è paralizzato. Soluzione: ridurre opzioni visibili, raggruppare funzionalità, menu progressivi. Esempio: Netflix limita le categorie e mostra raccomandazioni personalizzate.

3. Legge di Fitts (Fitts's Law)

Il tempo per raggiungere un target dipende da distanza e dimensione. Pulsanti più grandi e vicini al pollice sono più facili da toccare. Nelle app: posizionare azioni primarie nella "thumb zone", pulsanti CTA di almeno 44x44px, aumentare spazio tappabile.

4. Legge di Jakob (Jakob's Law)

Gli utenti passano la maggior parte del tempo su altre app, quindi si aspettano che la tua funzioni allo stesso modo. Non reinventare pattern consolidati: hamburger menu, swipe per eliminare, pull-to-refresh. Esempio: carrello in alto a destra nelle app e-commerce.

5. Legge di Miller (Miller's Law)

La memoria di lavoro può contenere 7 ± 2 elementi. Organizza le informazioni in gruppi: numeri di telefono in blocchi, menu con massimo 5-7 voci, onboarding in 3-5 step chiari. Il chunking migliora memorizzazione e usabilità.

Approfondimento: Soglia di Doherty

Definizione

La Soglia di Doherty stabilisce che la produttività aumenta drammaticamente quando il computer risponde all'utente in meno di **400 millisecondi**. Sotto questa soglia, il ciclo di feedback uomo-computer diventa così rapido che l'utente entra in uno stato di "flow" dove la tecnologia sembra rispondere istantaneamente ai suoi pensieri. Scoperta da Walter Doherty e Ahrvind Thadani nel 1982 presso IBM, questa legge rimane fondamentale per la UX moderna.

Esempi Pratici Mobile

Instagram Feed Loading

Usa skeleton screens che appaiono istantaneamente (< 100ms) mentre carica le immagini reali, dando percezione di velocità immediata.

Gmail Sync

Mostra nuove email entro 300ms dall'apertura app grazie a background sync e caching intelligente.

Spotify Play Button

Inizia riproduzione in < 200ms usando pre-buffering e transizione immediata a stato "playing".

Tecniche di Implementazione

Skeleton Screens:

Mostra layout vuoto istantaneamente invece di spinner generici

Ottimizzazione Backend:

API response time < 200ms per operazioni critiche

Caching Intelligente:

Salva dati frequenti localmente per accesso immediato

Lazy Loading:

Carica solo contenuto visibile, differisci il resto

Feedback Immediato:

Mostra stato "loading" entro 100ms dall'azione utente

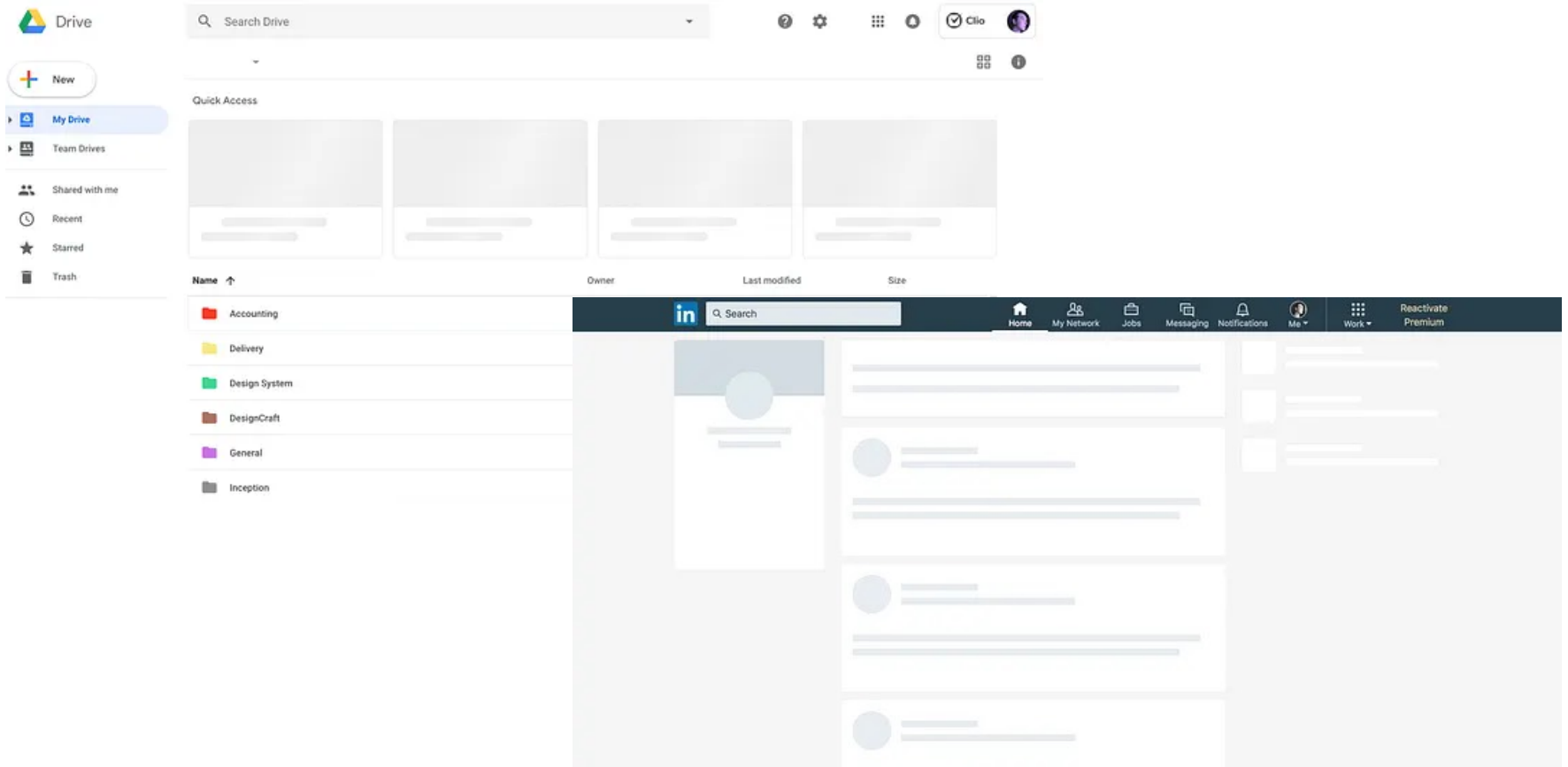
Optimistic UI:

Assume successo e aggiorna UI immediatamente

Rispettare la soglia di 400ms aumenta l'engagement del 25% e riduce il bounce rate del 32%

Approfondimento: Soglia di Doherty

Skeleton screen



Approfondimento: Legge di Fitts

$$T = a + b \log_2(D/W + 1)$$

Tempo necessario per raggiungere un target in funzione di distanza e dimensione

D = Distanza

Distanza dal punto di partenza (dito dell'utente) al centro del target. Più il target è lontano, più tempo serve per raggiungerlo.

W = Dimensione

Larghezza del target lungo l'asse del movimento. Target più grandi sono più facili e veloci da colpire.

Applicazioni Pratiche Mobile

FAB (Floating Action Button)

Grande (56x56dp) e posizionato in basso a destra nella thumb zone per accesso rapido.

Thumb Zone

Azioni frequenti nella zona facilmente raggiungibile dal pollice (bottom 1/3 dello schermo).

Bottom Navigation Bar

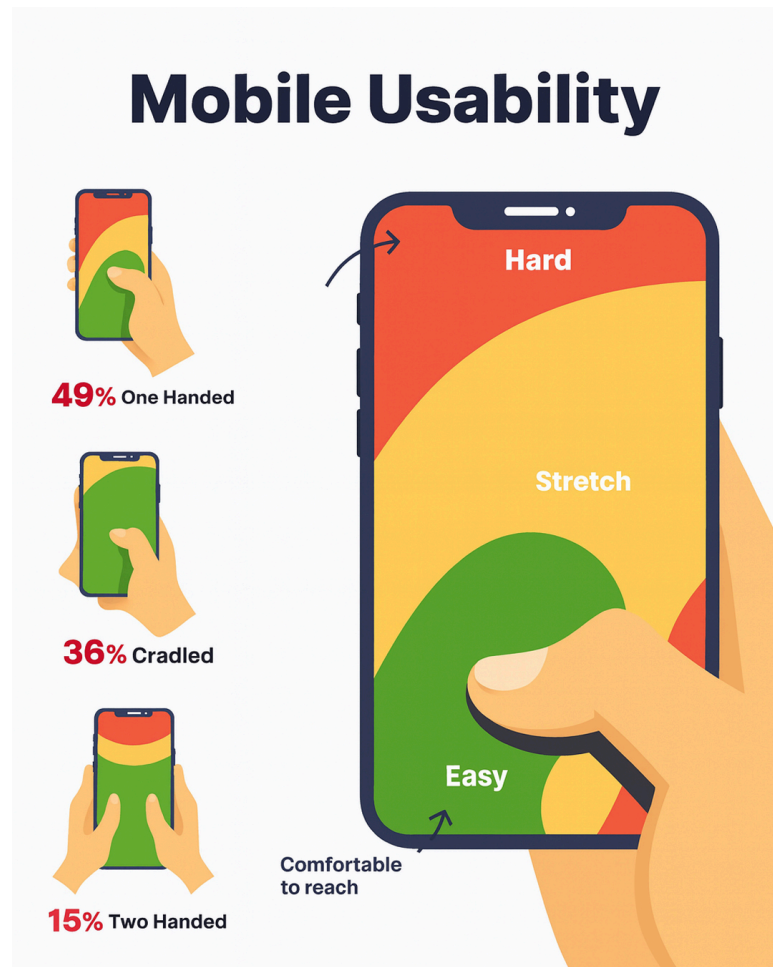
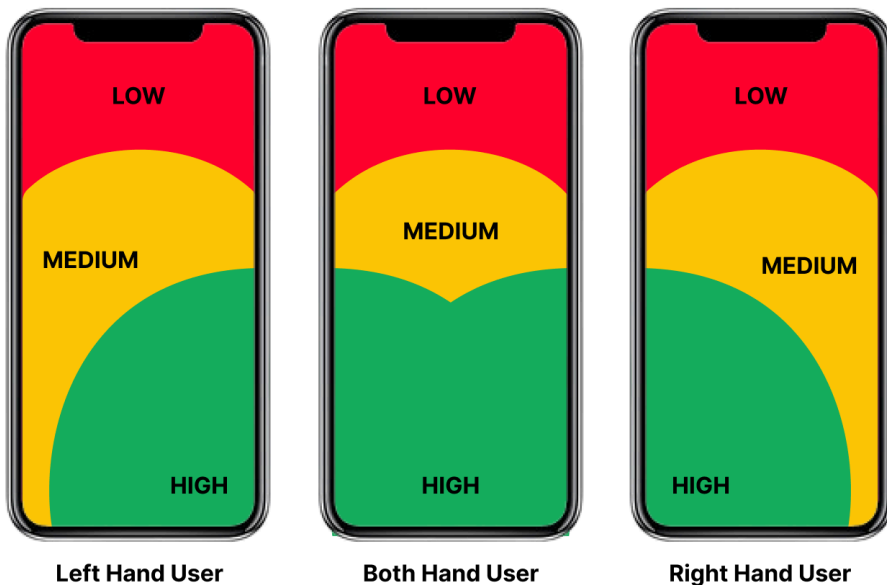
Menu principale in basso con icone grandi (24dp) e padding generoso per tap facile.

Best Practices per Ottimizzazione

- ✓ Touch target minimo 44x44px (Apple) o 48x48dp (Android) per garantire tap accurato
- ✓ Posiziona azioni frequenti vicino alla thumb zone (bottom-center dello schermo)
- ✓ Aumenta dimensione target per azioni critiche (CTA primari, pulsanti di conferma)
- ✓ Spaziatura minima 8px tra elementi tappable per evitare tap accidentali

Esempio: Pulsante 48x48px a 300px di distanza è 3x più veloce da raggiungere rispetto a 24x24px alla stessa distanza

Approfondimento: Legge di Fitts



Approfondimento: Legge di Hick

$$RT = a + b \log_2(n)$$

Il tempo di reazione aumenta logaritmicamente con il numero di scelte disponibili

Ogni scelta aggiuntiva aumenta il carico cognitivo dell'utente. Ridurre le opzioni migliora la velocità di decisione e l'esperienza complessiva.

✗ Menu Complesso (15 voci)

Home • Profilo • Impostazioni • Notifiche • Messaggi • Amici • Gruppi • Eventi • Foto • Video • Marketplace • Gaming • Notizie • Pagine • Altro...

Tempo decisione: ~4.5 secondi

Sovraccarico cognitivo elevato, utente confuso

✓ Menu Semplificato (5 categorie)

 Home •  Profilo •  Social •  Media •  Altro

Tempo decisione: ~1.8 secondi

Navigazione intuitiva, decisioni rapide

Tecniche di Riduzione Scelte

- **Chunking:** Raggruppa opzioni correlate in categorie (max 5-7 per categoria)
- **Progressive Disclosure:** Mostra solo opzioni essenziali, nascondi funzioni avanzate in sottomenu
- **Categorizzazione:** Usa icone e colori per differenziare visivamente le categorie
- **Default Intelligenti:** Pre-seleziona l'opzione più comune per ridurre decisioni

Paradosso della Scelta: troppi opzioni paralizzano l'utente e riducono la soddisfazione del 40%

Approfondimento: Legge di Jakob

Secondo la Legge di Jakob, siamo stupidi :)

Gli utenti preferiscono che i siti e le app funzionino nello stesso modo in cui funzionano quelli che già conoscono.

In pratica, quando l'interfaccia segue pattern familiari, l'utente non deve imparare nulla di nuovo: si muove più velocemente, commette meno errori e percepisce l'app come “naturale” e ben progettata.

Per questo nella UX/UI è fondamentale:

- usare convenzioni note (icone, menu, layout);
- mantenere coerenza tra schermate;
- evitare sorprese inutili.

La familiarità riduce il carico cognitivo e rende l'esperienza immediata e piacevole.

Approfondimento: Legge di Miller

La Legge di Miller afferma che la memoria umana a breve termine può gestire in media fino a 7 ± 2 elementi. In UX/UI questo significa che l'interfaccia deve ridurre il carico cognitivo, evitando di mostrare troppe informazioni tutte insieme.

Per applicarla bene:

- raggruppa gli elementi in blocchi logici (**chunking**);
- limita le opzioni nei menu e nelle liste;
- semplifica i percorsi e riduci le decisioni richieste all'utente.

Esempi pratici di **chunking** in UX/UI:

- Numeri di telefono: 349 123 4567 (è più facile da leggere che 3491234567)
- Menu suddivisi in sezioni: Impostazioni → Profilo / Notifiche / Privacy / Pagamenti
- Form lunghi spezzati in step: Step 1 → Dati personali • Step 2 → Indirizzo • Step 3 → Pagamento
- Card visive che raggruppano info correlate, invece di una lunga lista testuale.

Operazioni Sincrone vs Asincrone: Non Bloccare l'Utente

Le operazioni **sincrone** bloccano l'interfaccia fino al completamento. Le operazioni **asincrone** permettono all'utente di continuare a interagire mentre l'operazione procede in background.

✗ Operazione Sincrona (Bloccante)

- Utente preme "Upload Foto"
- App inizia upload
- UI si blocca, spinner infinito
- Utente non può fare nulla
- Upload completa (10-30 secondi)
- UI si sblocca finalmente

UI BLOCCATA PER 10-30 SECONDI

Esempio Negativo

App di social media che blocca completamente l'interfaccia durante l'upload di una foto. L'utente non può scorrere il feed, non può aprire altre sezioni, non può fare nulla. Se la connessione è lenta, l'app sembra crashata. Risultato: frustrazione e abbandono.

✓ Operazione Asincrona (Non Bloccante)

- Utente preme "Upload Foto"
- App avvia upload in background
- UI rimane responsive immediatamente
- Utente continua a navigare l'app
- Progress bar discreta mostra avanzamento
- Notifica al completamento

UI SEMPRE RESPONSIVE

Esempio Positivo

Instagram permette di continuare a scorrere il feed mentre la foto viene caricata in background. Una barra di progresso discreta in alto mostra l'avanzamento. L'utente può persino chiudere l'app e l'upload continua. Risultato: esperienza fluida e senza interruzioni.

Collegamento con Soglia di Doherty: operazioni asincrone mantengono l'app responsive sotto 400ms, creando percezione di velocità istantanea

Pattern Asincroni per UX Eccellente



Background Sync

Sincronizza dati in background senza bloccare UI



Optimistic UI

Aggiorna UI immediatamente assumendo successo



Progress Indicator

Mostra avanzamento operazioni lunghe



Queue Management

Gestisci operazioni multiple in coda



Offline-First

Funziona offline, sincronizza quando possibile

Esempi Reali da App di Successo

Gmail

Invia email in background. Puoi chiudere l'app immediatamente dopo aver premuto "Invia". Se offline, mette in coda e invia quando torna connessione. Mostra "Invio in corso..." discretamente.

Spotify

Scarica playlist offline in background. Puoi continuare ad ascoltare musica, esplorare, creare playlist mentre il download procede. Progress bar discreta mostra avanzamento.

Twitter

Posta tweet anche offline usando Optimistic UI. Il tweet appare immediatamente nel feed. Se invio fallisce, mostra badge "Retry" discreto. Utente non viene mai bloccato.

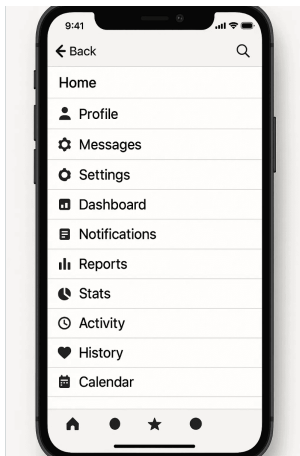
Best Practices per Operazioni Asincrone

- ✓ Mostra sempre feedback visivo immediato (< 100ms)
- ✓ Usa progress indicator per operazioni > 2 secondi
- ✓ Permetti cancellazione operazioni in corso
- ✓ Persisti stato per sopravvivere a crash/chiusura app
- ✓ Gestisci errori gracefully con retry automatico
- ✓ Notifica completamente se utente ha cambiato schermata

Operazioni asincrone ben implementate riducono l'abbandono del 45% e aumentano la percezione di velocità del 60%

Errori Comuni: Navigazione Confusa vs Chiara

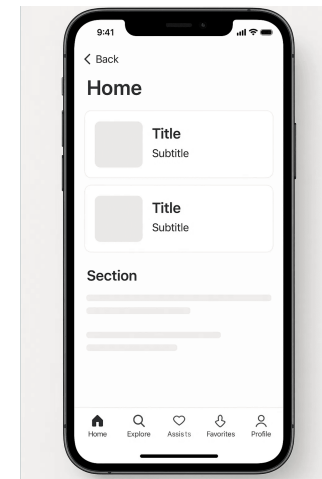
✗ Navigazione Confusa



- • Menu con 10+ voci tutte allo stesso livello
- • Icone senza etichette o ambigue
- • Pulsante "Indietro" inconsistente
- • Navigazione nascosta senza indicatori

✓ Navigazione Chiara

- • Menu principale con 4-5 categorie chiare
- • Icone universali con etichette descrittive
- • Navigazione bottom-bar per azioni frequenti
- • Breadcrumb o indicatori di posizione chiari



Errori Comuni: Gerarchia Visiva Assente vs Chiara

✗ Gerarchia Visiva Assente

- Tutti i testi della stessa dimensione senza distinzione tra titoli e corpo
- Colori uniformi senza contrasto che non guidano l'attenzione
- Elementi importanti sepolti nel contenuto senza evidenza visiva
- Mancanza di spazi bianchi (whitespace) che crea confusione visiva
- Impossibile distinguere azioni primarie da secondarie

✓ Gerarchia Visiva Chiara

- Titoli grandi (24-28pt), sottotitoli medi (18-20pt), corpo testo (14-16pt)
- Colore primario per CTA, colori neutri per contenuto secondario
- Elementi critici posizionati in posizioni prominenti (top, center)
- Generoso uso di padding e margin per respirabilità visiva
- Contrasto elevato tra elementi interattivi e contenuto statico

Principio Chiave: Pattern di Lettura F e Z

L'occhio umano segue naturalmente pattern a "F" o "Z" durante la lettura su schermo. Il pattern F è tipico per contenuti testuali (inizia in alto a sinistra, scansiona orizzontalmente, poi scende verticalmente a sinistra). Il pattern Z è comune per layout con meno testo (zigzag da sinistra a destra). Posiziona le informazioni più importanti seguendo questi percorsi naturali per massimizzare l'attenzione dell'utente. **Esempio reale:** Airbnb usa una gerarchia perfetta con immagini grandi e immersive in alto, prezzo in evidenza con font bold e colore contrastante, dettagli secondari (recensioni, posizione) in grigio chiaro, e CTA "Prenota" sempre visibile con colore vibrante e alta prominenza visiva.

Form Complessi vs Form Ottimizzati

✗ Form Sbagliati

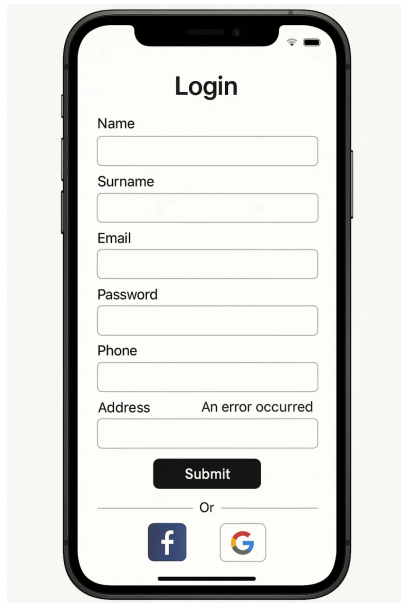
- Tutti i campi in una singola schermata lunga
- Nessuna validazione in tempo reale
- Messaggi di errore generici ("Errore nel campo")
- Tastiera numerica non attivata per campi numerici
- Nessun indicatore di progresso

✓ Form Corretti

- Form multi-step con progress indicator
- Validazione inline con feedback immediato
- Messaggi di errore specifici e costruttivi
- Tastiera contestuale per tipo di input
- Autofill abilitato, pulsante disabilitato fino a completamento

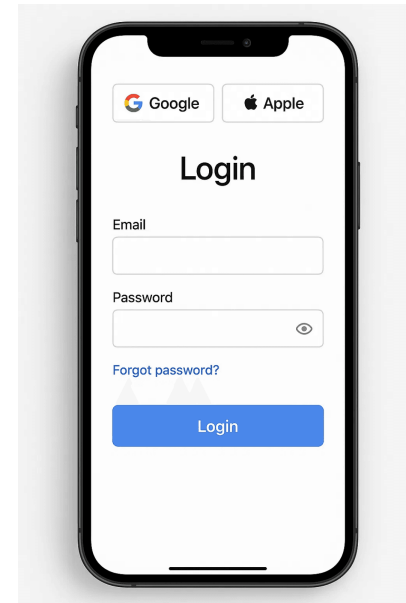
Mockup: Login Screen - Sbagliato vs Corretto

✗ Login Sbagliato



- Troppi campi non necessari (nome, cognome, telefono, indirizzo)
- Nessun toggle per mostrare/nascondere password
- Messaggio di errore generico e poco utile
- Social login nascosti in fondo alla pagina
- Pulsante submit piccolo e poco visibile

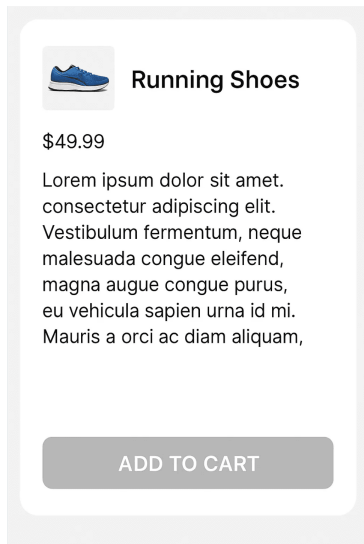
✓ Login Corretto



- Solo campi essenziali: email e password
- Icona occhio per mostrare/nascondere password
- Social login (Google, Apple) prominenti in alto
- Link "Forgot password?" chiaramente visibile
- CTA grande, colorato e facilmente tappable

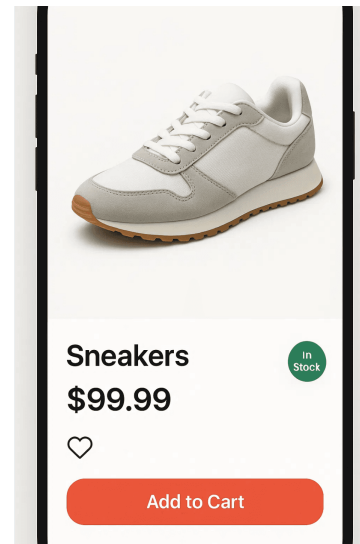
Mockup: Product Card E-commerce - Sbagliato vs Corretto

❌ Product Card Sbagliata



- Immagine prodotto piccola e poco visibile
- Prezzo nascosto in testo piccolo senza evidenza
- Troppo testo descrittivo che distrae
- Pulsante "Add to Cart" grigio e poco invitante
- Nessuna informazione su disponibilità o wishlist

✓ Product Card Corretta

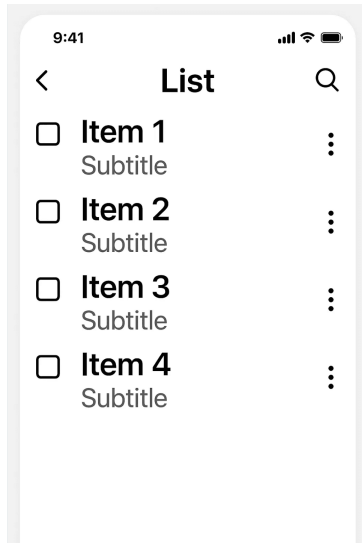


- Immagine hero grande che domina la card
- Prezzo prominente in bold con alta visibilità
- Informazioni essenziali solo (nome, prezzo, rating)
- CTA "Add to Cart" colorato e invitante
- Badge "In Stock" verde e icona wishlist accessibile

Product card ben progettate aumentano il tasso di conversione del 40% rispetto a layout confusi

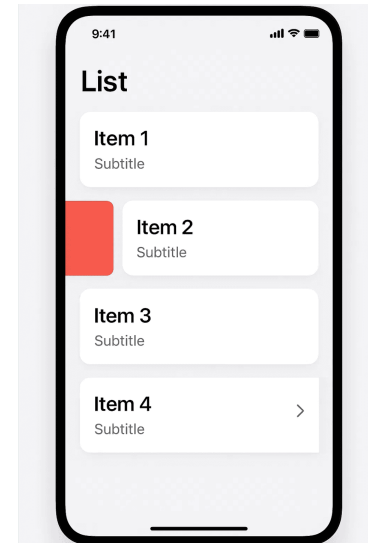
Mockup: Lista/Feed - Sbagliato vs Corretto

✗ Lista Sbagliata



- Elementi ammassati senza spaziatura adeguata
- Nessuna separazione visiva tra gli item
- Touch target troppo piccoli e difficili da toccare
- Azioni nascoste in menu a tre puntini
- Difficile distinguere un item dall'altro

✓ Lista Corretta



- Spaziatura generosa tra card per respirabilità
- Card ben separate con ombre sottili
- Touch target grandi e facilmente tappabili
- Swipe actions visibili con indicatori colorati
- Facile distinguere ogni item con gerarchia chiara

Touch target minimo raccomandato: 44x44px (Apple HIG Human Interface Guideline) per garantire accessibilità e ridurre errori di tap

Mockup: Checkout/Carrello - Sbagliato vs Corretto

✗ Checkout Sbagliato

Shipping Address

Full Name

Address

City

State ZIP Code

Billing Address

Full Name

Address

City

State ZIP Code

Payment

Card Number

Expiration Date CVV

PLACE ORDER

- Form lunghissimo con tutti i campi in una pagina
- Nessun riepilogo ordine visibile durante compilazione
- Costi di spedizione nascosti fino alla fine
- Nessun indicatore di progresso o step
- CTA "Place Order" poco chiaro e non sticky

✓ Checkout Corretto

2 / 3

CHECKOUT

Payment

Card number

MM / YY CVC

Billing address

Enter address

Order Summary

Basic Tee \$32.00

Subtotal \$32.00

Shipping \$5.00

Total \$37.00

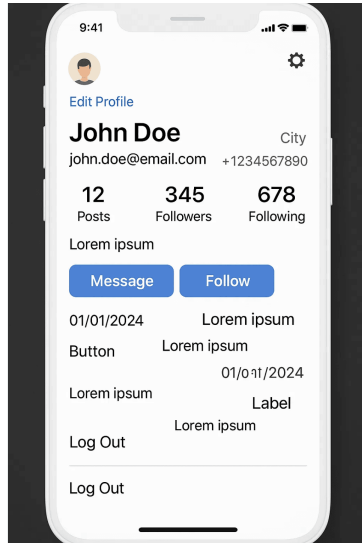
Pay \$37.00

- Multi-step con progress indicator chiaro (2/3)
- Riepilogo ordine sempre visibile con thumbnail prodotto
- Costi trasparenti: subtotal, spedizione, totale
- Step ben organizzati e non sopraffacenti
- CTA "Pay \$37.00" grande, colorato e sticky in basso

Il 69% degli utenti abbandona il carrello a causa di checkout complessi, costi nascosti o form troppo lunghi

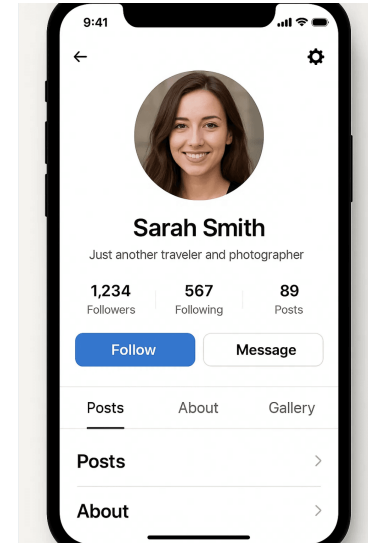
Mockup: Profilo Utente - Sbagliato vs Corretto

✗ Profilo Sbagliato



- Informazioni disorganizzate sparse ovunque
- Avatar piccolo e poco prominente
- Azioni e pulsanti posizionati casualmente
- Nessuna gerarchia visiva o sezioni chiare
- Difficile trovare impostazioni e funzioni importanti

✓ Profilo Corretto

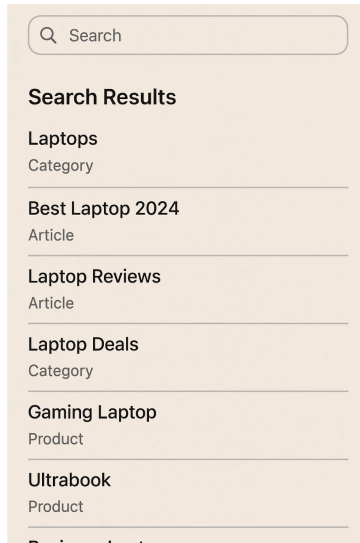


- Avatar grande e prominente al centro in alto
- Informazioni organizzate in sezioni ben definite
- Statistiche (followers, following, posts) in griglia chiara
- Azioni principali (Follow, Message) raggruppate
- Icona impostazioni facilmente accessibile in alto a destra

Architettura dell'informazione: organizzare contenuti in modo logico e gerarchico riduce il carico cognitivo dell'utente

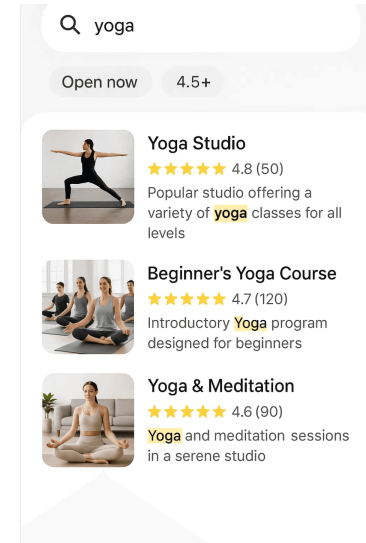
Mockup: Search/Ricerca - Sbagliato vs Corretto

✗ Ricerca Sbagliata



- Barra di ricerca piccola e poco visibile
- Nessun suggerimento o autocomplete durante digitazione
- Filtri completamente nascosti in menu hamburger
- Risultati senza preview o immagini
- Keywords cercate non evidenziate nei risultati

✓ Ricerca Corretta



- Barra di ricerca grande e prominente in alto
- Autocomplete con suggerimenti in tempo reale
- Filtri visibili come chip accessibili
- Risultati ricchi con preview immagini e rating
- Keywords evidenziate in giallo nei risultati

Il 43% degli utenti abbandona un'app se la funzione di ricerca è difficile da usare o produce risultati poco rilevanti

Touch Target: Inadeguati vs Ottimizzati

✗ Touch Target Inadeguati

- Pulsanti piccoli (< 40px) difficili da toccare con precisione
- Elementi interattivi troppo vicini tra loro causando tap accidentali
- Area tappabile limitata all'elemento visivo senza padding esteso
- Azioni distruttive (elimina, cancella) senza conferma o possibilità di annullare

✓ Touch Target Ottimizzati

- Touch target minimo di 44x44px secondo le raccomandazioni Apple HIG
- Spaziatura di almeno 8px tra elementi tappabili per evitare errori
- Area tappabile estesa oltre l'elemento visivo con padding invisibile
- Azioni distruttive sempre accompagnate da conferma e possibilità di undo
- Pulsanti grandi e ben spaziati posizionati nella thumb zone per massima accessibilità

Il 47% degli utenti abbandona un'app dopo un solo errore di tap causato da target troppo piccoli o ravvicinati

Esempio Pratico - Spotify: L'app di Spotify implementa perfettamente i touch target ottimizzati con pulsanti grandi e ben spaziati. Il pulsante "Play" centrale è enorme e facilmente raggiungibile, mentre azioni secondarie come "Aggiungi a playlist" mantengono comunque target generosi di almeno 48x48px. Tutti i controlli principali sono posizionati nella thumb zone inferiore dello schermo, permettendo un uso confortevole con una sola mano.

Psicologia del Colore nelle App Mobili



Rosso

Urgenza, azione, errore. Usato per notifiche importanti, pulsanti di eliminazione e alert.

Esempio: YouTube per brand e pulsante "Iscriviti"



Verde

Successo, conferma, crescita. Ideale per messaggi di completamento e pulsanti di conferma.

Esempio: WhatsApp per sicurezza e comunicazione positiva



Blu

Fiducia, professionalità, calma. Dominante in app finanziarie e social network.

Esempio: Facebook, LinkedIn, PayPal per affidabilità



Arancione

Ottimismo, energia, attenzione. Perfetto per CTA e elementi di highlight.

Esempio: Amazon per "Aggiungi al carrello"

Contrasto e Accessibilità WCAG (Web Content Accessibility Guidelines)

Il rapporto di contrasto minimo secondo le linee guida WCAG è 4.5:1 per testo normale e 3:1 per testo grande. Utilizzare strumenti come Contrast Checker per verificare l'accessibilità dei colori scelti. Un contrasto adeguato garantisce leggibilità per utenti con problemi visivi e in condizioni di luce variabili.

Psicologia del Colore nelle App Mobili



Usate le palette di colori: <https://colorhunt.co/>

Bottoni nelle app Mobili

Android

BUTTON

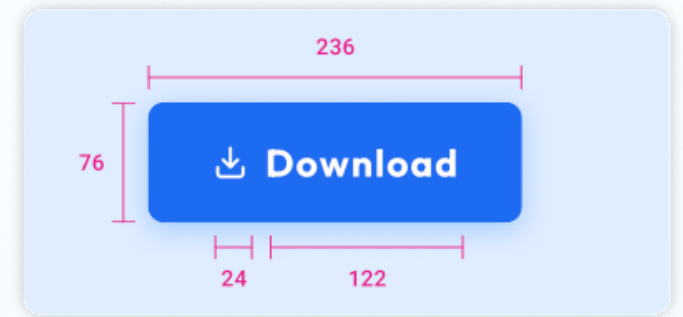
iOS

Button

Flat

Outline

Text



Font nelle applicazioni mobili

Perché i font sono importanti nelle app

Un font leggibile rende i contenuti chiari e immediati, migliorando l'esperienza d'uso. La tipografia crea gerarchia tra titoli, testi e pulsanti, guidando lo sguardo e facilitando la navigazione. Font incoerenti o poco leggibili generano confusione. Una buona scelta tipografica rende l'app più professionale, accessibile e piacevole da usare.



<https://fonts.google.com/>

Icone nelle applicazioni mobili

Perché le icone sono fondamentali nelle app mobili

Le icone guidano l'utente in modo rapido e intuitivo. In uno schermo piccolo, un simbolo chiaro e riconoscibile comunica più velocemente di qualsiasi testo, riducendo lo sforzo mentale e migliorando la navigazione.

Icone ben progettate rendono l'app più semplice da usare, evitano errori, aumentano la comprensione immediata delle azioni e contribuiscono all'identità visiva dell'interfaccia. Al contrario, icone ambigue o incoerenti generano confusione e rallentano l'esperienza.

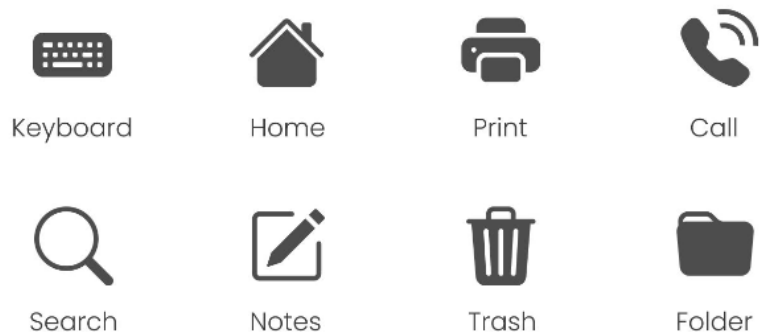
In breve: icone chiare = app più intuitive, veloci e piacevoli.

Formato consigliato: .SVG .PNG

Siti consigliati:

<https://iconscout.com/>

<https://ui8.net/>



Animazioni nelle applicazioni mobili

Animazioni di immagini e icone: perché sono importanti

Le animazioni rendono l'esperienza più fluida e naturale: aiutano l'utente a capire cosa sta succedendo, indicano cambi di stato e guidano l'attenzione. Un'icona che si muove leggermente o un'immagine che si anima al tocco confermano l'azione e rendono l'interfaccia più viva.

Usate con moderazione, le animazioni migliorano la percezione di qualità, rendono l'app più intuitiva e facilitano la comprensione dei passaggi. Se eccessive o troppo lente, però, diventano un ostacolo.

In breve: animazioni semplici e veloci rendono l'interazione più chiara, piacevole e moderna.

Formato consigliato: Lottie files

Sito consigliato: <https://lottiefiles.com/>

Microinterazioni che Fanno la Differenza

Pull-to-Refresh

Animazione fluida che conferma immediatamente l'azione registrata. Fornisce feedback visivo immediato che l'utente ha attivato il refresh, riducendo l'incertezza e migliorando la percezione di reattività dell'app.

Swipe Gestures

Swipe per eliminare con animazione di conferma, swipe per archiviare con indicatore visivo chiaro. Le gesture naturali rendono l'interazione più intuitiva e veloce rispetto ai pulsanti tradizionali.

Transizioni Contestuali

Animazioni che mostrano la relazione spaziale tra schermate: slide da destra per avanzare, da sinistra per tornare indietro. Aiutano l'utente a mantenere l'orientamento mentale nella navigazione.

Skeleton Screens

Invece di mostrare spinner generici, visualizza il layout che si sta caricando. Questo approccio riduce la percezione del tempo di attesa del 30%, rendendo l'esperienza più fluida e professionale.

Haptic Feedback

Vibrazioni sottili quando si completa un'azione importante come invio messaggio, like o completamento task. Il feedback tattile rinforza l'azione visiva creando un'esperienza multisensoriale.

Esempio Eccellente: Headspace <https://play.google.com/store/search?q=headspace&c=apps>

L'app di Headspace implementa magistralmente le microinterazioni utilizzando colori pastello rilassanti, animazioni fluide e delicate che rinforzano il messaggio di calma e meditazione. Ogni interazione è accompagnata da transizioni morbide, feedback visivo sottile e animazioni che non distraggono ma guidano dolcemente l'utente attraverso l'esperienza. Il risultato è un'app che non solo funziona bene, ma comunica i suoi valori attraverso ogni dettaglio dell'interazione.

Strutturare le Schermate per Massimizzare l'Usabilità

Tab Bar (Bottom Navigation)

Ideale per 3-5 sezioni principali di pari importanza con accesso frequente. Icone chiare con etichette, sezione attiva evidenziata, sempre visibile. Posizionata nella thumb zone per massima accessibilità.

Esempio: Instagram (Home, Cerca, Reels, Shop, Profilo)

Hamburger Menu

Utilizzato per molte opzioni secondarie in app content-focused. Problema: riduce il discovery del 50% ("out of sight, out of mind"). Soluzione: combinare con tab bar per funzioni primarie, riservare il menu per impostazioni e funzioni avanzate.

Esempio: Medium (menu laterale + tab bar per lettura)

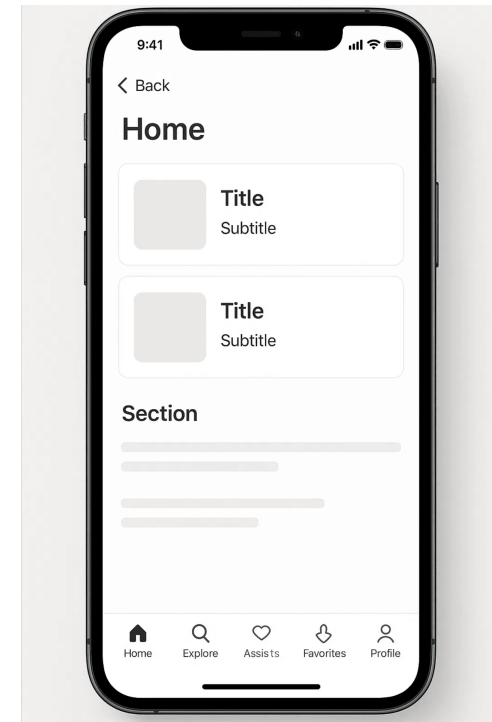
Gesture-Based Navigation

Per app immersive che massimizzano lo spazio contenuto. Richiede tutorial chiaro, hint visivi e fallback con pulsanti. Swipe orizzontale per navigazione tra contenuti, verticale per scroll e discovery.

Esempio: Tinder (swipe left/right), TikTok (swipe verticale)

Thumb Zone & Card Design

Il 70% degli utenti usa lo smartphone con una mano. Posizionare azioni frequenti nella zona verde (bottom center), contenuto secondario in zona gialla (middle/sides), evitare azioni critiche in zona rossa (top corners). Card con padding 16-24px, un'azione primaria per card.



Esempio di Navigazione Ottimale

Best Practice: Bottom tab bar con 5 icone chiare ed etichette, header pulito con back button, contenuto organizzato in card con gerarchia visiva, spazi bianchi generosi, azioni primarie facilmente raggiungibili con il pollice.

I Primi 60 Secondi Determinano il Successo dell'App

Il 25% degli utenti abbandona un'app dopo il primo utilizzo

Il 77% degli utenti usa un'app meno di 3 volte

Un onboarding efficace aumenta la retention del 50%

Tipologie di Onboarding

Progressive Onboarding

Insegna le funzionalità nel contesto d'uso. L'utente impara facendo, non leggendo. Tooltip e coach marks quando necessario. Approccio consigliato per massima efficacia.

Esempio: Duolingo introduce una lezione alla volta

Function-Oriented

3-5 schermate che spiegano value proposition e funzionalità chiave. Visuale accattivante, testo minimo (max 1 frase per slide). Skip sempre disponibile.

Esempio: Calm mostra benefici in 4 schermate eleganti

Account-First

Richiede registrazione immediata. Rischio: aumenta friction, riduce conversioni del 30%. Usare solo se necessario (social, banking). Spiegare i benefici prima.

Esempio: LinkedIn richiede account per accedere

Best Practices per un Onboarding Efficace

- ✓ Mostra valore prima di chiedere permessi (spiega perché serve la posizione)
- ✓ Personalizza l'esperienza (chiedi preferenze per customizzare contenuti)
- ✓ Offri sempre un'uscita (skip button visibile, possibilità di tornare al tutorial)
- ✓ Permetti di esplorare senza account (guest mode o trial gratuito)
- ✓ Celebra i primi successi (gamification e reward per completamento setup)

Empty States Intelligenti: Quando l'utente non ha ancora contenuto, non mostrare schermate vuote. Usa lo spazio per spiegare come popolare quella sezione, suggerire azioni iniziali, mostrare esempi. Esempio: Pocket mostra articoli suggeriti quando la lista è vuota. Notion inizia con template e introduce funzionalità con tooltip contestuali.

10 Errori UX che Uccidono le App (e Come Risolverli)

1. Ignorare gli Standard della Piattaforma

Seguire Human Interface Guidelines (Apple) e Material Design (Google). Pulsante "Indietro" in alto a sinistra su iOS, con icona freccia su Android.

2. Richiedere Troppi Permessi Subito

Richiedere permessi just-in-time spiegando il beneficio. Le richieste contestuali aumentano l'accettazione dell'80%.

3. Notifiche Invasive e Irrilevanti

Personalizzare notifiche, dare controllo all'utente, limitare frequenza. Esempio: Duolingo manda reminder personalizzati basati sui tuoi obiettivi.

4. Testi Troppo Piccoli o Poco Contrastati

Minimo 16px per corpo testo, contrasto 4.5:1. Il 15% della popolazione ha problemi visivi.

5. Form Lunghi Senza Salvataggio Progressivo

Implementare autosave, draft e possibilità di riprendere. Esempio: Google Forms salva automaticamente le risposte.

6. Feedback Assente o Tardivo

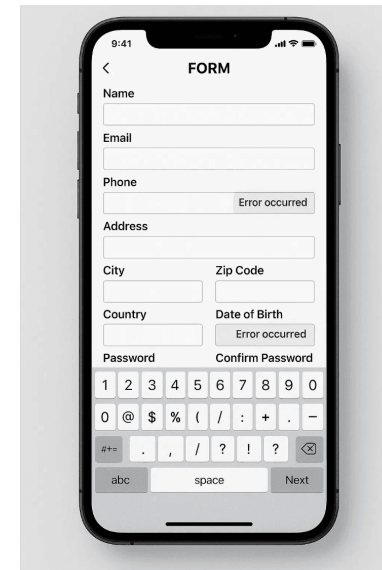
Feedback immediato (< 100ms), loader per operazioni > 1s, conferme per azioni critiche. L'utente deve sempre sapere cosa sta succedendo.

7. Ignorare gli Stati di Errore

Messaggi human-friendly, spiegare cosa è successo, suggerire soluzioni. "Impossibile connettersi. Controlla la connessione e riprova".

8. Navigazione Profonda Senza Uscite

Pulsante "Home" sempre accessibile, shortcut, close button in modal. Non costringere l'utente a tornare indietro 5 volte.



9. Ignorare il Contesto d'Uso Mobile

Ottimizzare per offline, one-hand use, schermi piccoli. Il 70% degli utenti usa lo smartphone con una mano sola.

10. Mancanza di Personalizzazione

Raccomandazioni basate su comportamento, preferenze salvate, contenuti dinamici. Esempio: Spotify crea playlist personalizzate.

La Checklist Definitiva per una UX Mobile Eccellente

Performance e Velocità

- ☐ App si avvia in meno di 3 secondi
- ☐ Transizioni fluide a 60fps
- ☐ Immagini ottimizzate (WebP, lazy loading)
- ☐ Funzionalità core disponibili offline

Contenuto

- ☐ Gerarchia visiva chiara (titoli, sottotitoli, corpo)
- ☐ Microcopy utile e human-friendly
- ☐ Empty states informativi e actionable
- ☐ Messaggi di errore costruttivi con soluzioni

Sicurezza e Privacy

- ☐ Dati sensibili mascherati
- ☐ Logout automatico dopo inattività
- ☐ Permessi richiesti just-in-time
- ☐ Controlli granulari su condivisione dati

Accessibilità (A11y)

- ☐ Contrasto colori conforme WCAG AA (4.5:1)
- ☐ Testi ridimensionabili senza perdita funzionalità
- ☐ Screen reader compatibility con etichette
- ☐ Alternative testuali per contenuti visivi

Navigazione

- ☐ Pattern consistenti in tutta l'app
- ☐ Breadcrumb o indicatori di posizione
- ☐ Back button sempre funzionante
- ☐ Deep linking per condivisione contenuti

Delight e Engagement

- ☐ Microinterazioni piacevoli ma non invasive
- ☐ Animazioni che comunicano relazioni spaziali
- ☐ Haptic feedback per azioni importanti
- ☐ Personalizzazione basata su comportamento

Usabilità

- ☐ Touch target minimo 44x44px
- ☐ Azioni frequenti nella thumb zone
- ☐ Massimo 3 tap per funzionalità principali
- ☐ Feedback immediato per ogni interazione

Form e Input

- ☐ Campi raggruppati logicamente
- ☐ Validazione inline in tempo reale
- ☐ Tastiera contestuale per tipo di input
- ☐ Progress indicator per form multi-step

Testing e Iterazione

- ☐ Test su dispositivi reali (non solo emulatori)
- ☐ Test con utenti di diverse età e competenze
- ☐ A/B testing per decisioni critiche
- ☐ Ciclo continuo di feedback e miglioramento

Metriche di Successo UX

Task Success Rate: % utenti che completano un'azione

Time on Task: Tempo medio per completare un'azione

Error Rate: Frequenza errori utente

System Usability Scale (SUS): Score standardizzato di usabilità

Net Promoter Score (NPS): Probabilità di raccomandazione

Risorse per Approfondire

Nielsen Norman Group (nngroup.com) - Ricerca UX

Laws of UX (lawsofux.com) - Principi interattivi

Material Design (material.io) - Guidelines Android

Human Interface Guidelines (developer.apple.com) - Guidelines iOS

UX Collective (uxdesign.cc) - Case study e best practices

Come lavorare in Delphi ?

- Partendo da un mockup si creano i vari form/frame grafici includendo
 - Icone, animazioni Lottie
 - Font
 - Suoni
 - Immagini
- Decidere il work flow della UI / UX
- Splash screen e onboarding screen
- Utilizzare librerie multithread come quelle di EMBT o Cocinasync
<https://bitbucket.org/sivv/cocinasync>
- Utilizzare i TMessage x aggiornare la UI da qualsiasi form/frame
- Gestire i messaggi di notifica e errori
- Gestione della rete verificando se lo smartphone è connesso a internet
- Notifiche locali e push

Risorse:

<https://www.figma.com/it-it/>

<https://stitch.withgoogle.com/>

<https://iconscout.com/>

<https://lottiefiles.com/>

<https://ui8.net/>

<https://www.pixeltrue.com/free-ui-kits/habit-builder-ui-kit>

<https://www.fiverr.com/>



Q&A



THANK YOU