

Recherche von best practices von UX Design Guidelines für mobile Apps

Dennis Sima

Version 0.1 - 2021-04-06

Inhaltsverzeichnis

1. Abstract	1
2. Warum ist UI und UX wichtig?.....	2
2.1. Was ist UI & UX?	2
2.2. Usability Vs. User Experience.....	3
2.3. UI & UX im Laufe der Geschichte:.....	4
3. Wie entsteht User Experience?	6
3.1. Den Nutzer verstehen	6
3.2. User Research.....	7
3.3. Empathize	7
3.4. Create	7
3.5. Test	8
3.6. Develop	9
4. UI-Patters für mobile Apps	10
4.1. Was ist das?	10
4.2. Vorteile von UI-Patterns	11
4.3. Beliebteste Patterns in der Smartphone-Welt	12
4.4. Dark UI-Design Patterns	13
4.5. Wie sollte man UI-Design Patterns anwenden?.....	16
4.6. Beliebteste UI-Patterns Library	16
5. Weiteres Design.....	19
5.1. Farbwahl.....	19
5.2. Typografie.....	23
6. Accessibility und Internationalisierung	26
6.1. Warum sollte man auf Accessibility Bezug nehmen?	26
6.2. Auf was achten?	26
6.3. Internationalisierung.....	29
Glossar	31
Quellen.....	32

Chapter 1. Abstract

Diese Arbeit bietet die Grundlage für User Interface und User Experience-Design Guidelines für mobile Applikationen. Es werden nicht die Basics des App-Designs erklärt, sondern auch Themen, wie Accessibility, Internationalisierung etc., welche für eine moderne Smartphone-App oftmals entscheidend sein können, um erfolgreich zu werden.

Es ist dennoch zu beachten, dass es sich um keine vollständigen Richtlinien handelt, da diese Schriftfassung als kompakte Recherche zum Thema UI und UX dienen soll, in einem für den Leser möglichst einfachen dünnen Format.

Chapter 2. Warum ist UI und UX wichtig?

2.1. Was ist UI & UX?

Damit der Enduser mit einer Software kommunizieren kann (z.B. Daten bearbeiten, speichern etc.) braucht dieser eine Schnittstelle, um dies einfach zu tätigen. In der modernen Softwareentwicklung ist auch oft die Rede des Frontends, wobei dies als obersten Sammelbegriff für die gesamte User-Applikation verwendet wird. Da der User keine Hintergrundprozesse des Backends im Normalfall mitbekommt, ist das Userinterface, alles, was der User sieht und bedienen darf. Es ist auch beim UI folglich das Screendesign die Rede, da es nur um das Design der Applikation geht.

Um das User Interface von der User Experience zu unterscheiden, kann man logisch differenzieren, dass das UI, jene Komponenten zusammenfasst, welche der User sieht und die UX wie der User die Applikation folglich bedient. So gesehen stehen UI und UX komplementär zueinander und sind in der modernen Frontend-Entwicklung nicht mehr wegzudenken.

Darüber hinaus ist das UX-Design so gesehen essenzieller als das Screendesign, da sich im Regelfall aus der User Experience das Design der App ergibt und nicht umgekehrt.

"Vergleicht man nur ein Handy aus den 1990er Jahren mit einem Smartphone von heute wird deutlich, wie komplex Produkte zum Teil geworden sind. Genau dann wird die User Experience essenziell wichtig für die Produktentwicklung."

— Prenger C. (2019 14. August): matter-of-design, <https://matter-of-design.com/darum-ist-user-experience-so-wichtig> aufgerufen am 06.04.2021

User Interface	User Experience
<ul style="list-style-type: none"> • Visuelles Aussehen der Applikation • Fokussiert sich auf das Produkt • Elemente die der User sieht und bedienen darf 	<ul style="list-style-type: none"> • Identifizierung der Bedürfnisse der Benutzer • Fokus auf den Zweck und die Funktionalität • Fokussiert sich auf die Probleme der User

Table 1. Table UI VS. UX

2.2. Usability Vs. User Experience

Usability ist vom englischen übersetzt die Benutzerfreundlichkeit. Es beschreibt wie sehr eine App effizient und gebrauchstauglich ist.



Als Enduser fällt schlechte Usability mehr auf als Gute.

Der Unterschied zwischen User Experience und Usability ist, dass bei der Usability nur der Fokus auf die effiziente Nutzergestaltung gelegt wird. Wohingegen User Experience das komplette Nutzererlebnis samt Emotionen, Gefühle und Freude beschreibt.

Aus der Praxis würde dies bedeuten, dass man besonderes Augenmerk beispielsweise auf die Menüleiste für die Navigation legen sollte, da dies meist der Startpunkt für den User ist, um sämtliche Features der App zu erkunden.

Die App-Navigation sollte sich hierbei möglichst für den Benutzer intuitiv anfühlen. Wichtig sind zusätzlich ebenso, neben der Menüplatzierung, die Beschriftung der Themenfelder. Deshalb sollte großen Wert auf geläufige Begriffe gelegt werden. So können sich User sehr gut in der App zurechtfinden.

Trotzdem muss stets eine gute Lesbarkeit der Inhalte gegeben sein. In diesem Fall

ist es wichtig, dass nicht nur genügend große Buchstaben verwendet werden, sondern auch, dass die ausgewählte Schriftart unabhängig von der verwendeten Plattform gut lesbar ist. In diesem Fall sind Buchstaben mit einfachem Design von Vorteil.^[1]

2.3. UI & UX im Laufe der Geschichte:

Die historische Entwicklung von dem sogenannten User Interface und der User Experience begann mit der Erfindung des **HTML (Hypertext Markup Language)**, welcher der Informatiker Tim Berner Lee im Jahre 1990 im Laufe seines Projektes im europäischen Kernforschungszentrum CERN betrieb. Damals war das oberste Ziel möglichst einfach Daten und Informationen, sowohl effizient und visuell ansprechend, über Computer zu versenden.^[2] Über HTML konnte man strukturiert Daten und Informationen darstellen.

Hier sieht man ein einfaches HTML-Beispiel, wie früher das UI ausgesehen hat:

```
<!DOCTYPE html>
<html lang="de">
  <head>
    <meta charset="UTF-8">
    <meta content="Das sind alle Informationen über HTML" name=
"description">
    <title>HTML lernen: Das Grundgerüst</title>
  </head>
  <body>
    <h1>Hauptüberschrift: <i>Meine erste HTML-Seite</i></h1>
    <p>Dies ist meine <strong>erste</strong> HTML-Seite!</p>
    <h2>Überschrift des zweiten Absatzes</h2>
    <p>Dies ist der zweite Absatz meiner <em>ersten HTML-Seite</em>
.</p>
    <p><b>Anmerkung</b>: Ein typisches Anwendungsbeispiel des &lt;
b&gt;-Tags.</p>
    <ol>
      <li>erster Punkt der geordneten Testliste</li>
      <li>zweiter Punkt der geordneten Testliste</li>
      <li>dritter Punkt der geordneten Testliste</li>
    </ol>
  </body>
</html>
```

Dieses HTML erzeugt folgendes UI:



Figure 1. HTML-Frontend Beispiel

Seit der Erfindung des HTML wurde der optischen Gestaltung des UI für die Enduser mehr Aufmerksamkeit geschenkt und wurde laufend weiterentwickelt. Sowohl die Ansprüche heutiger Apps und Websites haben sich angepasst, wodurch man heutzutage wesentlich mehr braucht als ein HTML-file um den User zufriedenzustellen mit allen Funktionen.

[1] vgl. <https://www.seobility.net/de/wiki/Usability>

[2] vgl. https://de.wikipedia.org/wiki/Tim_Berners-Lee

Chapter 3. Wie entsteht User Experience?

Für User Experience gibt es keine einheitlichen Vorgaben oder konkrete Prozesse, da diese sich individuell an den spezifischen Endnutzer und der eigentlichen Zielgruppe für das Software-Produkt richtet und jene Kunden unterschiedliche „Needs“ und „Pains“ haben:

3.1. Den Nutzer verstehen

Der Enduser steht immer im Mittelpunkt, mit samt seiner Wünsche und Bedürfnisse. Je nach Budget und Zeitdruck kann man dies mittels Umfragen direkt am Markt oder durch Kundenbeobachtungen abschätzen.

Zentrales Element sind hierbei die Personas, welche realitätsgetreu die Kunden abbilden, welche schließlich die Software Applikation nutzen werden.

Im Zuge des UX-Designprozesses werden ebenso auch erste Wireframes oder Klick-Dummys visualisiert.^[3]

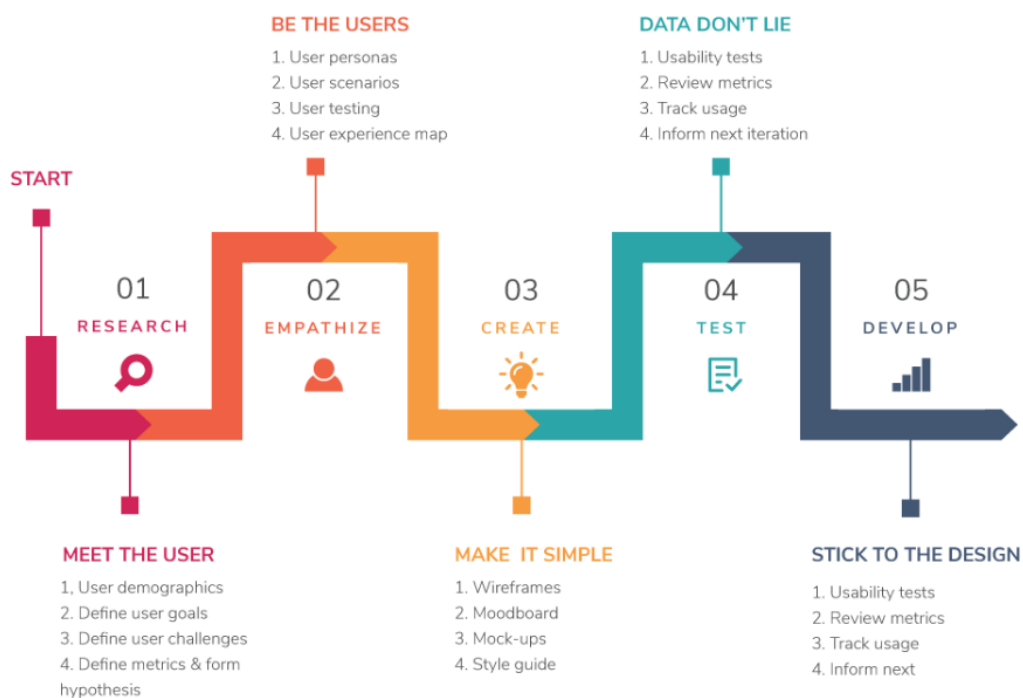


Figure 2. UX-Design Prozess, Quelle: <https://medium.com/nyc-design/ux-ui-design-process-for-beginner-753952bb2241>



Es gibt in der Theorie sehr viele Arten von gängigen UX-Prozessen, aber grundsätzlich meinen alle dasselbe: ***Customer first!***

3.2. User Research

Ublicherweise beginnt der UX-Designprozess mit einer sogenannten User Research. Hier versucht man zu verstehen, was die Kunden aktuell und in Zukunft auch für Features brauchen oder ebenso auch nicht wollen. Diese Recherche kann über mehrere Formen passieren. Zum einen über Marktumfragen oder Kundenbeobachtungen. Sei gesagt, dass Marktumfragen wiederum sehr kostenintensiv sind, aber meist bessere und genauere Ergebnisse liefern.

Wichtig zu verstehen ist genauso, dass man nicht nur die zukünftigen Kunden analysiert, sondern auch die Projektanforderungen, sowie eine Wettbewerbsanalyse.^[4]

3.3. Empathize

In dieser Phase muss man sich in den Kunden hineinversetzen. Gängige Methoden wären unter anderem mehrere Personas zu erstellen, welche die Zielgruppe darstellen. Zusätzlich sollten diese Personas, abhängig ihres persönlichen Backgrounds, verschiedene Szenarios durchleben, wie, wann, wo, warum, diese Persona mit der App interagiert.^[4]

3.4. Create

Mit den bisher gesammelten Daten sämtlicher zukünftiger Nutzer werden im Anschluss erste Design-Entwürfe erstellt. Diese können in Form von Wireframes, Click-Dummies etc. dargestellt werden.^[5]

Um erste Entwürfe zu skizzieren, kann dies mit zahlreichen Online-Tools oder auf Papier erstellt werden.

Hier sieht man einen Beispiel-Wireframe der Smartphone-App:



Figure 3. Wireframe-Design von einer App

3.5. Test

Beim Testen ist es essenziell zu beobachten, wie der Erstentwurf bei den potenziellen User ankommt:^[4]

- Prototyp soll markttauglich genug sein, um genaues Marktfeedback zu bekommen
- Ziel ist zu Erkennen, ob der Design-Prototyp nach Plan funktioniert
- Feedback kritisch analysieren und einbauen

3.6. Develop

In der letzten Phase wird das fertige Produkt/ Prototyp entwickelt, Feedback der Kunden eingearbeitet und grundsätzliche Feinarbeiten an der App erledigt. Danach können eventuelle Beta-Tests angestrebt werden.

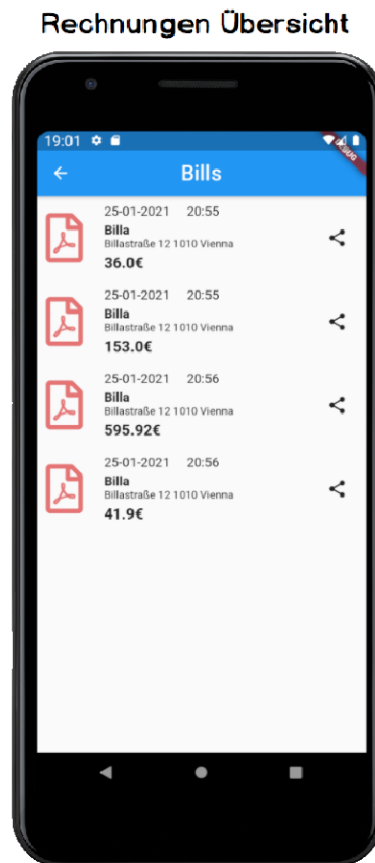


Figure 4. Wireframe-Design zum Prototypen



Wie am Beispielbild zu erkennen ist, ist es wichtig auch von den ersten Entwürfen und Ideen abzuweichen und nicht auf diese zu beharren, wenn die Ergebnisse dafür sprechen.

[3] vgl. <https://academy.technikum-wien.at/ratgeber/was-ist-ux-design/>

[4] vgl. <https://www.uxmatters.com/mt/archives/2020/08/mobile-app-ux-design-process.php>

[5] vgl. <https://www.mobileappdaily.com/mobile-app-design-guidelines>

Chapter 4. UI-Patters für mobile Apps

4.1. Was ist das?

Es ist grundsätzlich zu beachten, nicht nur in der Software Entwicklung, das Rad so wenig wie möglich neu zu erfinden. Es gibt viele bereits fertige Software-Lösungen zum Einbinden etc. Genauso ist es auch beim Screen Design, wie mit den UI-Patterns, wo man mit Erfahrung etwas Zeit und Arbeit sparen kann.

Gewisse UI-Patterns wurden nicht speziell nur für UI-Designer entwickelt und definiert. Sie sollen darüber hinaus auch der Software Architektur und den Programmierern zugutekommen.

Heutzutage sind UI-Patterns vollständig in die Software Entwicklung integriert und nicht mehr wegzudenken. Man findet diese in allen gängigen Apps und sogar Websites. Vor allem auch Endnutzer profitieren von solchen Patterns. Durch den Erkennungseffekt der bedienbaren Elementen wissen Nutzer darüber hinaus, in wenigen Sekunden, wie jene App-Elemente und -Komponenten funktionieren.^[6]



Wichtig ist bei UI-Patterns, dass nicht alle Lösungsentwürfe für alle App-Lösungen geeignet sind. Diese müssen immer stets an den individuellen Use Case angepasst werden.

„Most of us wear t-shirts. My t-shirt and yours may vary in size and fit, but both are recognizable as t-shirts. We can add a nifty little pocket, details on the short sleeves, and print all kinds of stuff on them. However, the structure is pretty much the same regardless of how much we tailor the garment to our liking.“

— MARIA DE LA RIVA. (2020 12. August): Careerfoundry,
<https://careerfoundry.com/en/blog/ui-design/user-interface-patterns/#1-what-are-ui-design-patterns> aufgerufen am 06.04.2021

4.2. Vorteile von UI-Patterns

Nicht zum Verwechseln sind UI-Patterns allerdings mit einem Baukastensystem, mit welchen man diverse Designs und Features zusammenbauen kann und die Arbeit damit getan ist. UI-Patterns sind für UI-Designers wie Baupläne, an denen man sich bei der Frontendgestaltung richten kann. Wie beispielsweise, wenn sich ein Architekt alte Baupläne von anderen Gebäuden ansieht und sich inspirieren lässt.

Zusätzlich sind UI-Patterns eine gemeinsame Sprache für die Kommunikation für UI-Designer. Dadurch werden vor allem auch Missverständnisse reduziert und sorgen für Kontinuität bei der Zusammenarbeit mit mehreren Designern am selben Projekt.

Bei den Endusern der Apps werden Design-Patterns auch zum Vorteil. Da sich viele Apps von der Gestaltung nicht allzu unterscheiden, haben Nutzer bereits eine gewisse Vorstellung auf welcher App-Page welche Elemente zum Vorschein kommen.

Beispielweise erwarten sich User als Mindestanforderungen von einer Login-Seite einer App Zwei Input-Felder (Für E-Mail und Passwort) und einen „Bestätigen“-Button zu einloggen in die App. Andere Elemente wären unschlüssig und führen zu Verwirrungen der Nutzer.^[6]

Hier kann man als exemplarisches Beispiel eine Login-Maske einer Smartphone-App sehen, welche diverse Patterns nachgeht und erfüllt:

(Dadurch weiß der Nutzer ohne Überlegen, was die Intention dieser Page ist)

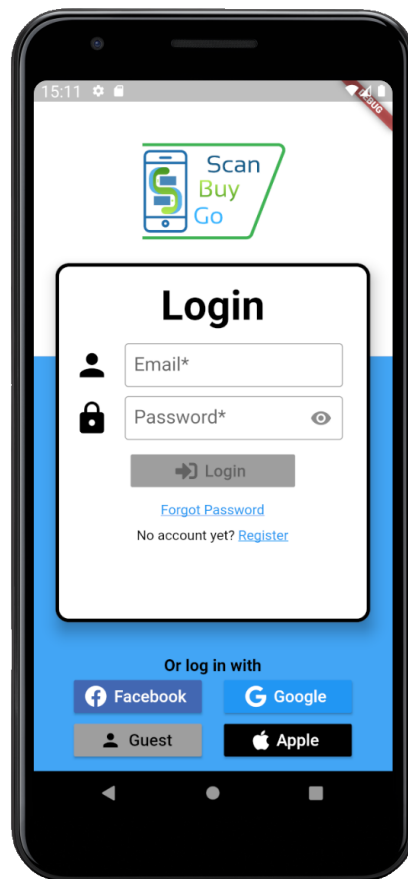


Figure 5. Design Patterns einer Login-Maske von einer App

4.3. Beliebteste Patterns in der Smartphone-Welt

Pauschal kann man nie definieren welche Patterns in eine App gehören. Jedoch gibt es auffallend viele Apps die zumindest folgende aufweisen:^[6]

1. **Teilen:** Ermöglicht, meist als Button, dem User seinen Inhalt auf Social-Media-Plattformen zu teilen
2. **Navigation:** Ermöglicht, meist als Pfeil nach links, dem User zur vorigen Page zu gelangen oder zurück zur Startseite
3. **Input/output:** Ermöglicht dem User Information zu schreiben und abschicken

und darauffolgend Feedback zu seiner Aktion zu bekommen

4. **Content Struktur:** In welcher Art werden Elemente dem User präsentiert?
Wirkt es für den User, als wäre der Inhalt organisiert, intuitiv und zugänglich?

4.4. Dark UI-Design Patterns

Direkt vergleichbar mit den „normalen“ UI-Design Patterns sind die sogenannten „Dark UI-Design Pattern“. Wie der Name bereits ausdrückt stecken für den Enduser meist nicht kundenfreundlich moralische Design Entwürfe zur Verfügung, welche aber dennoch dem Plattformbetreiber zugutekommen würden. Man versucht den Usern in der App (gibt es in allen Frontend-Software-Lösungen) zu einer speziellen Aktion zu drängen, der dieser eventuell ohne dieses Design-Pattern nicht machen würde. Natürlich versuchen sowohl UX- und UI-Designer gezielt mit der Psychologie der zukünftigen Kunden zu spielen, um diese zu überlisten, ohne dass diese es bemerken.

Trotzdem sind Dark Patterns heutzutage in der Frontend-Gestaltung sehr üblich und auf vielen Apps und auch Websites zu finden. Trotzdem sind sie allgemein unbeliebt und gefährden bei Missbrauch die Beziehung eines Unternehmens zu seinen Kunden.

4.4.1. Beliebteste „Dark Patterns“

Auffallend ist, dass Dark Patterns überwiegend von Shopping-Applikationen und bei der Anmeldung der User angewendet werden.^[7]

Verwirrende Fragen & Sprache:

Hier versucht man den User oftmals durch mehrere Verneinungen zu verwirren und mit mehreren vermeidlich ähnlichen Fragen zu täuschen.

Please enter your details to reserve your item(s)

Title :

First name * :

Last name * :

Email * :

Phone number * :

☐ Please do not send me details of products and offers from Currys.co.uk

☐ Please send me details of products and offers from third party organisations recommended by Currys.co.uk

Reserve items

Figure 6. Beispiel: Verwirrende Sprache und Fragen bei der Registrierung, Quelle: <https://www.darkpatterns.org/types-of-dark-pattern/trick-questions>

Versteckte Akzeptierung

Auf manchen Verkaufs-Apps kann es sein, dass man mit einem automatisch aktivierten Optionsfeld ein zusätzliches Produkt hinzugefügt bekommt. Am deutlich häufigsten ist es allerdings, dass sich der User unabsichtlich für einen Newsletter anmeldet.

Bestellübersicht [Warenkorb bearbeiten](#)

1. Rechnungsadresse angeben 2. Zahlart 3. Bestellung bestätigen

Ich bin bereits foodspring Kunde

E-Mail-Adresse

Passwort

Anmelden

[Passwort vergessen?](#)

Ich bin Neukunde

Ich möchte als Gast fortfahren

Bezahlmethode auswählen

☐ PayPal

☐ Kreditkarte

☐ Sofortüberweisung

☐ eps Online-Überweisung

☐ Vorkasse

☐ Lastschrift

☐ Rechnung

LIEFERUNG:
Circa: Mo. 19.04 - Mi. 21.04

ARTIKEL	SUMME
1x Protein Bar Extra Chocolate 12er Paket mix	24,99 €
Versand	4,90 €
Gutschein vorhanden?	
Gesamtsumme inkl. MwSt.	29,89 €

Bestellung abschließen

Mit deiner Bestellung erklärst du dich mit unseren [AGB](#), [Datenschutzbestimmungen](#) und der [Widerrufsbelehrung](#) einverstanden.

☒ Für den Newsletter anmelden
Abmeldung jederzeit möglich

Figure 7. Beispiel: Versteckte Akzeptierung des Newsletter, Quelle: <https://www.foodspring.at>

Premium Mitgliedschaft

Das Design und das Nutzererlebnis machen es möglichst einfach etwas zu kaufen, aber im Nachhinein sehr schwer es wieder abzubestellen. Dadurch muss man oft Umwege durch die App nehmen.

Ablenkung

Das Design wird entsprechend gestaltet, sodass der User unterbewusst seine Aufmerksamkeit auf etwas Bestimmtes lenkt und etwas anderes wichtiges vernachlässigt oder sogar übersieht. Wenn, beispielsweise etwas groß angezeigt wird und gut farblich hervorgehoben wird, dann übersehen viele die restlichen Elemente.

Versteckte Kosten

Dem User werden erst kurz vor dem Check-out Prozess alle Kosten angezeigt und nicht vorher.

Schuld einreden

Um dem User ein schlechteres Gefühl zu beschern, wenn dieser zum Beispiel ein Werbeangebot ablehnen will, wird diesem mit gezielter Sprache eingeredet, dass er selber Schuld ist, dieses Angebot nicht anzunehmen.

Zahlreiche Beispiele sind unter <https://confirmshaming.tumblr.com> zu finden.

4.5. Wie sollte man UI-Design Patterns anwenden?

Zunächst ist wichtig zu verstehen, dass man nicht einfach mit einzelnen UI-Design Patterns eine grafische Oberfläche erstellen sollte. Zuerst muss man diese noch speziell für den eigenen Use Case anpassen.

Beschreibe das Problem, welches aktuell bearbeitet wird. Dann wäre ein Blick in eine UI-Design Pattern Library empfehlenswert, wie andere Designer ähnliche Lösungen entwickelt haben. Danach stellt sich die Frage, was man daraus lernen kann in Bezug zu dem eigenen Problem und erstellt im Anschluss seine eigene Lösung.^[6]

4.6. Beliebteste UI-Patterns Library

In dem gesamten World Wide Web gibt sehr viele UI-Design Libraries. Zu Beginn sollte man sich nicht nur auf eine fokussieren und in der weiteren Folge wird man seine Lieblings-Library finden.

Bekanntere UI-Design Libraries sind folgende:

1. <https://mobbin.design/patterns>
2. <https://pttrns.com/>
3. <https://uigarage.net>
4. <https://material.io/>
5. <https://www.mobile-patterns.com/>



Für Cross Plattform oder Progressive Web-Apps gibt es wenig bis keine Unterschiede zum Design für iOS oder Android. Bei einer Native-App sieht dies ein wenig anders aus. Man sollte hierbei achten, ob das jeweilige Design des Patterns zu dieser App-Plattform passt.

4.6.1. Apple Human Interface Guidelines

Apples eigener App Store legt im Gegensatz zu dem, von Mitbewerber Google angebotenen, Google Play Store sehr großen Wert auf Standards.

Zusammengefasst, sind diese in gesamt 5 Punkten unterteilt: ^[8]

1. Safty
2. Performance
3. Business
4. Design
5. Legal

Sollte eine App nicht die Mindeststandards der UI/UX-Guidelines entsprechen wird sie auf dem Weg in den App Store abgelehnt. Deswegen ist es absolut essenziell für jeden UI/UX-Designer sich mit Apples Vorgaben auseinandergesetzt zu haben.

Zu finden sind diese Richtlinien von Apple unter: <https://developer.apple.com/design/human-interface-guidelines/>

4.6.2. Android Material Design Guidelines

Ebenfalls wie Apple besitzt auch Google für ihre Plattform Empfehlungen und Hilfeleistungen für Entwickler und Designer. Im Gegensatz zu Apple kann ein Entwickler somit, aufgrund der optionalen Richtlinien von Android, Apps in den Google Play Store hochladen, welche nicht unbedingt qualitativ sind.

Zu finden sind diese Empfehlungen/Hilfestellungen von Google unter: <https://material.io>



Material.io enthält sowohl Elemente und Komponenten für iOS, Android, Web und dem Frontend-Framework Flutter

[6] vgl. <https://careerfoundry.com/en/blog/ui-design/user-interface-patterns/#1-what-are-ui-design-patterns>

[7] vgl. <https://www.darkpatterns.org/types-of-dark-pattern>

[8] vgl. <https://developer.apple.com/app-store/review/guidelines/>

Chapter 5. Weiteres Design

Für eine gelungene App spielen hierbei noch mehrere Faktoren eine Rolle. Es ist vielmehr ein Zusammenspiel von einerseits der **Farbwahl**, welche man keineswegs, von sowohl ihren Nutzen und Effektivität, unterschätzen darf und andererseits der **Typografie**.

5.1. Farbwahl

Eine grundlegende Entscheidung für jede mobile App wird jene sein: Eine geeignete Farbe zu suchen. Deswegen ist es von großer Bedeutung eine Farbpalette mit Bedacht auszuwählen. Zusätzlich spielen Farben ebenso auch in der Psychologie des Menschen eine Rolle, mit Emotionen und Gefühlen, welche unterbewusst ausgelöst werden.

Deswegen verhilft die sogenannte Color Theory diesen Prozess der Farbwahl zu beschleunigen. Dieser umfasst zudem das Color Wheel.^[9]

5.1.1. Das Color Wheel

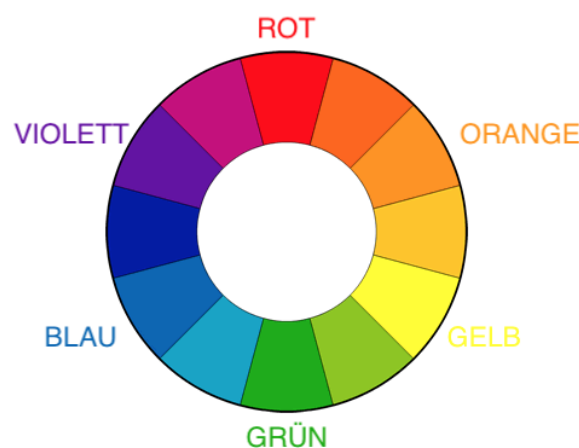


Figure 8. Das Color Wheel, Quelle: https://www.tigercolor.com/color-lab/color-theory/color-theory-intro.htm#Primary_colors

Der Aufgabenbereich des Color wheel, auch Farbkreis genannt, besteht darin Beziehungen zwischen Primär-, Sekundär-, und Tertiärfarbe aufzuzeigen. Einen

ersten Entwurf lieferte im Jahre 1666 Sir Isaac Newton.

Dabei umfassen diese die Primärfarben Blau, Rot und Gelb. Aus diesen Grundfarben ergeben sich wiederum die Sekundärfarben Grün, Orange und Violett. Die zusätzlichen Tertiärfarben sind im Anschluss grün-gelb, gelb-orange, orange-rot, rot-violett, violett-blau und blau-grün.^[9]

Warme und kühle Farben

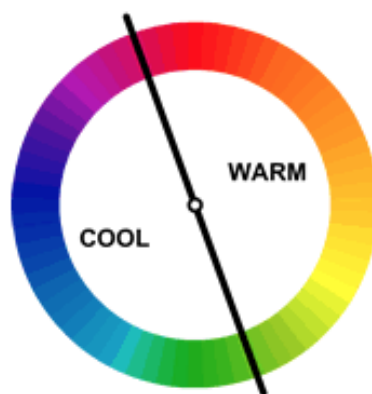


Figure 9. Warme oder kalte Farben im Color Wheel, Quelle: https://www.tigercolor.com/color-lab/color-theory/color-theory-intro.htm#Primary_colors

Grundsätzlich kann man Farben in zwei große Überkategorien fassen. Da die Augen warmes oder kaltes Licht speziell wahrnehmen, gilt dies ebenso bei Farben. Es werden beispielsweise die Farben von Rot über Orange bis Gelb als warme Farbe bezeichnet, da der Mensch diese mit warmen Gegebenheiten assoziiert, wie das Feuer (Rot-Orange) und die Sonne (Gelb). Dem Gegenübergestellt sind die kühlen Farben. Mit Blau wird beispielsweise kühles Wasser verbunden.^[9]

Farbharmonie

Um passende Farben für das User Interface auszuwählen ist es wichtig eine gewisse Harmonie zwischen den Farben aufzubauen.

Hierfür gibt es mehrere Möglichkeiten Farben zu bestimmen:

Monochromatische Farbvariante

Die monochromatische Farbvariante besteht aus mehreren Farbtönen aus derselben Primärfarbe. Diese entstehen, wenn man diese Farbe heller oder dunkler aussehen lässt.^[9]

Komplementäre Farbvariante

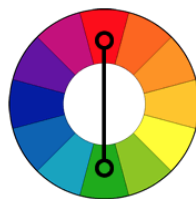


Figure 10. Eine komplementäre Farbmischung, Quelle: https://www.tigercolor.com/color-lab/color-theory/color-theory-intro.htm#Primary_colors

Komplementärfarben sind in den meisten Fällen schwierig anwendbar, funktionieren dennoch als Farben, damit ein Element auffällig aussieht.^[9]



Komplementärfarben sind schlecht für Text geeignet.

Analoge Farbvariante



Figure 11. Eine analoge Farbmischung, Quelle: https://www.tigercolor.com/color-lab/color-theory/color-theory-intro.htm#Primary_colors

Da diese Farben sehr nah nebeneinander liegen, fungieren diese einheitlich. Diese Farbkombination lässt sich vor allem in der Natur vorfinden und ist dem menschlichen Auge sehr angenehm. Üblicherweise verwendet man eine

dominantere Farbe, eine Zweite zur Unterstützung und zusätzlich eine Dritte, als Akzentfarbe.^[9]

Triadische Farbvariante



Figure 12. Eine triadische Farbmischung, Quelle: https://www.tigercolor.com/color-lab/color-theory/color-theory-intro.htm#Primary_colors

Aufgrund dieser Farbkombination wird eine lebendigere Stimmung erzeugt, mit der zusätzlich, im Gegensatz zur analogen Variante, mehr Kontrast zum Vorschein kommt. Üblicherweise wird eine dominantere Farbe ausgewählt und die restlichen nur zur Unterstützung verwendet.^[9]

5.1.2. Welche Farben für welche App?

Für die Gestaltung des Frontends darf man die Wichtigkeit und Wirkung dieser niemals unterschätzen. Mit einer adäquaten Farbgestaltung des UI kann man den User positiv emotional und unterbewusst beeinflussen. Beispielsweise strahlt die Farbe Blau Ruhe, Objektivität, Neutralität und Klarheit. Dieses verstärkt das Vertrauen der User und gibt diesen ein sicheres Gefühl als bei anderen Farben.



Figure 13. Blau wäre eine Option für eine vertrauenswürdige Payment-App

5.1.3. Corporate Identity

Die sogenannte Corporate Identity ist das Selbstbild jedes Unternehmens. Es umfasst alle Strategien, welche zu einer besseren, sowohl Firmeninternen und -externen, Ausstrahlung verhelfen.^[10]

Eine Corporate Identity wird in weitere Untergebiete unterteilt:

- Corporate Design (CD)
- Corporate Behaviour (CB)
- Corporate Culture
- Corporate Communication (CC)
- Corporate Philosophy (CP)

Corporate Design in der App-Entwicklung

Das CD richtet sich spezifisch auf den Wiedererkennungseffekt eines Unternehmens, welches die entsprechenden Firmenwerte mit sich tragen soll.

Grundsätzlich ist davon auszugehen, dass die Farbwahl mit denen des Firmenlogos möglichst übereinstimmen sollte, wenn dieses in der App verwendet werden soll, da dadurch ein größerer Wiedererkennungseffekt von der App ausgestrahlt wird. In Summe umfasst das Corporate Design sämtliche Bereiche um die Farbauswahl, Logodesign und Schriftdesign.

5.2. Typografie

In der weiten Welt der App-Entwicklung werden ungefähr 95% des gesamten Contents über Text dargestellt. Das heißt, dass der User egal, ob bewusst oder unterbewusst mit dem Text kommunizieren wird. Wenn ein Logo beispielsweise einen Schriftzug beherbergt, fällt dieser immer dem menschlichen Auge zuerst auf. Dies unterstreicht die Wichtigkeit von Typografie.^[11]



Bei iOS und Android-Apps sind der Default-Schriftzug Roboto beziehungsweise Helvetica Neue.^[12]

5.2.1. Lesbarkeit

Obwohl Smartphones mit der Zeit tendenziell größer werden, ist es auf den kleinen Bildschirm nicht immer leicht viel Text unterzubringen. Besonders für Apps, die großen Mengen an Text dem User aufbereiten, ist die Lesbarkeit sehr von Bedeutung. Optimal wäre es vor allem auch im mobilen App Bereich lesbare Schrift, welche auch bei direkter Sonneneinstrahlung gut lesbar ist.

5.2.2. Professionalität

Die Schriftart sollte stets mit dem Image der App und der eigentlichen Zielgruppe übereinstimmen. Wenn beispielsweise eine Banking-App keine seriöse Schrift wählt, verunsichert dies viele Kunden.^[11]

5.2.3. Schriftgröße

Die Guidelines von Apple geben Aufschluss darüber, dass bei iOS-Apps die Schriftgröße nur zwischen 11 und 17p betragen darf. Für Android-Apps ist diese Richtlinie ebenfalls legitim. Zu beachten gilt weiters, dass die ungefähre Zeichenanzahl pro Zeile im Hochformat bei 35 bis 50 liegt. Dadurch entsteht die Herausforderung Sätze nicht kompliziert zu formulieren und dennoch genug Informationen an den User zu bringen.^[11]

5.2.4. Kontrast

Ebenso wichtig ist auch der Spielraum zwischen Buchstaben und dem Hintergrund. Den besten Kontrast liefert dementsprechend schwarz auf weiß. Allerdings soll hier auch Bezug genommen werden zum Logo, Corporate Design etc.^[11]

[9] vgl. <https://www.tigercolor.com/color-lab/color-theory/color-theory-intro.htm>

[10] vgl. <https://www.ionos.at/startupguide/unternehmensfuehrung/corporate-identity/>

[11] vgl. <https://de.yeePLY.com/blog/app-design-die-bedeutung-der-typografie/>

[12] vgl. <https://mobilbranche.de/2014/08/typografie-beim-app-design>

Chapter 6. Accessibility und Internationalisierung

Heutzutage bestehen Apps im Gegensatz zu früher, aus mehr als nur funktionalen Features. Mehr Design, mehr Experience, mehr Usability sind die Folge, um User zufriedenzustellen und sind nicht mehr wegzudenken. Jedoch wird sehr schnell die Bedeutung von Accessibility vergessen, also eine Hilfestellung für beeinträchtigte Menschen.

„Whilst many development teams see accessibility as a ‘nice-to-have’, it is crucial in today’s mobile world.“

— Caspar Geerlings (2019, 21. Oktober): How to create an accessible app (and why you should); Quelle: <https://medium.com/oberonamsterdam/how-to-create-an-accessible-app-and-why-you-should-5493f41f8bdb>

6.1. Warum sollte man auf Accessibility Bezug nehmen?

Ein großes Missverständnis liegt auf Accessibility. Viele UI- und UX-Designer, sowie Entwickler nehmen dies als optionales Feature wahr. Jedoch wird geschätzt, dass ungefähr 15%, also 1.000.000.000 Menschen davon betroffen sind.^[13] Zusätzlich möchte der User eventuell auch die App während dem Autofahren oder beim Joggen verwenden. Das bedeutet, dass es auf jeden Fall sinnvoll und in der heutigen Welt essenziell geworden ist, seine App accessible zu designen und zu entwickeln.^[14]

6.2. Auf was achten?

Um für möglichst viele User verfügbar zu sein, sollte man ebenso auf die verschiedenen Aspekte der Accessibility eingehen. Was sind die Bedürfnisse, die daraus resultieren? Was brauchen die User?^[14]

6.2.1. Sehschwäche

Da man hauptsächlich visuell mit dem Smartphone kommuniziert, ist dies ein großes Hindernis für zahlreiche Menschen. Hinzu kommen mehrere Ausprägungen, wie Farbenblindheit und Blindheit über mehrere Stufen bis zum kompletten Sehverlust. Abhilfe schafft hier, wenn man nicht explizit auf Farben setzt und das UI ebenfalls auch anders gestaltet, z.B. größeren Zoom der Texte auf 200% erlauben etc. Zusätzlich würde es der Accessibility guttun einen Screen Reader oder ähnlich einzubinden.^[14]

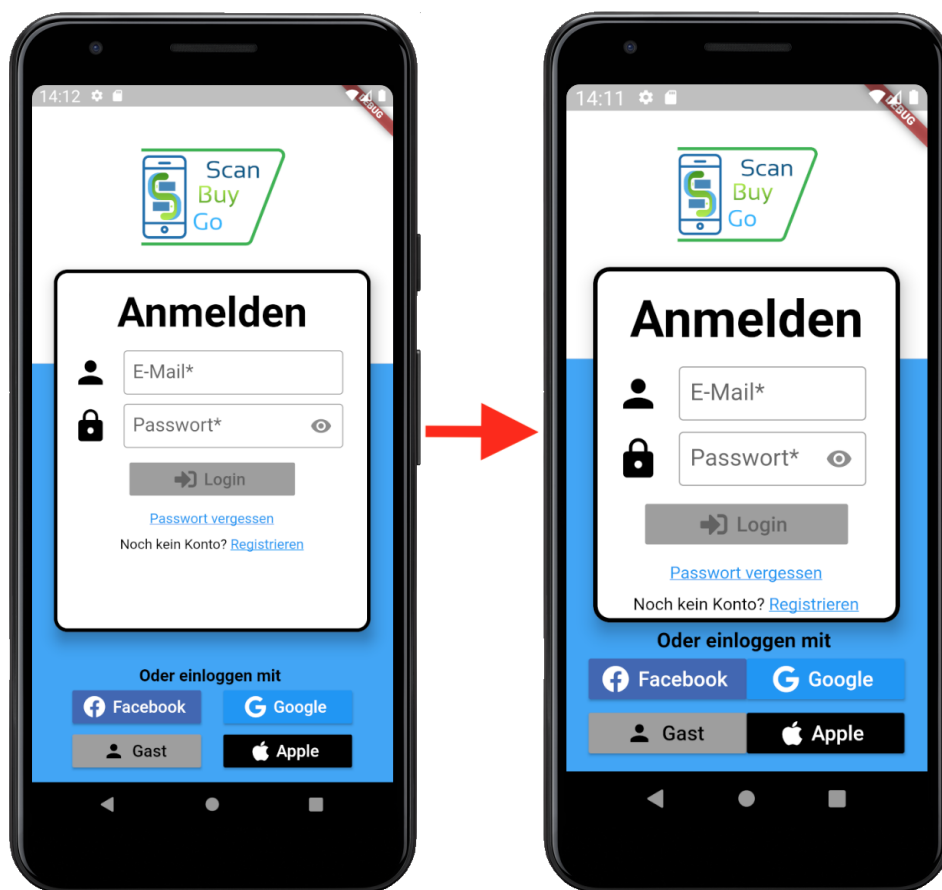


Figure 14. Beispiel für größere Schrift/ Zoom

6.2.2. Hörschwäche

Diese Beeinträchtigung ist normalerweise bei vielen Apps kein allzu großes Problem. Bei Videos oder Sound-Files könnten, als Option zusätzlich mittels Transkription angezeigt werden.^[14]

6.2.3. Lernschwäche

Unter diesem Übergriff werden sämtliche Konzentrations-, Lese- und Schreibschwächen gemeint. Diese kann man beispielsweise mit kurzen Sätzen, mehr Icons oder über die Spracheingabe, statt dem Schreiben in der Smartphone-App, optimieren.^[14]

6.2.4. Was liefern Apple and Google?

Da sowohl die Betriebssysteme iOS und Android zahlreiche Accessibility-Features in ihr Ecosystem integriert haben, sind diese Funktionen ebenso für Apps verwendbar.

Gängige iOS-Features sind:^[15]

- Untertitel und Audiobeschreibungen
- Voice-over
- Übersetzung
- Geführter Zugang
- Anpassung anzeigen

Gängige Android-Features sind:^[15]

- Audio und Bildschirmtext
- Anzeigeoptionen
- Interaktionskontrollen
- Screenreader

6.3. Internationalisierung

Um global zu denken und eine mobile App erfolgreich über Vertriebskanäle, wie gängige App-Stores zu vertreiben ist es äußerst ratsam ebenfalls diese in möglichst viele Sprachen zu übersetzen. In der IT-Welt ist auch häufig von i18n die Rede. Dies hat den folgenden Hintergrund, da genau 18 Buchstaben zwischen dem "I" und dem "n" von Internationalization liegen.

Aus der Praxis empfiehlt sich zumindest eine App, als absolute Mindestanforderung in Englisch bereitzustellen. Bestenfalls sogar in die meistgesprochene Sprache der Welt. Diese sind beispielsweise: Englisch, Deutsch, Französisch, Spanisch, Portugiesisch, Italienisch, Japanisch und Russisch. Sei gesagt, dass dies vorwiegend von westlichen Ländern gesprochen wird, da in anderen bevölkerungsstarken Ländern, wie China und Indien mehrere Sprachen gesprochen werden und man diese nicht leicht definieren kann.



Figure 15. Eine App in mehrere Sprachen ist mittlerweile schon Pflicht

6.3.1. Lokalisierung

Lokalisierung, auch l10n genannt, ist die Anpassung von nicht textlichen Elementen, wie die Formatierung von Datum, Uhrzeit, Adressen, Icons und Währungen etc. Dies bewahrheitet sich in der praktischen Entwicklung, als sehr mühsam, aber erhöht und verbessert den Komfort und die User Experience der Nutzer.^[16]

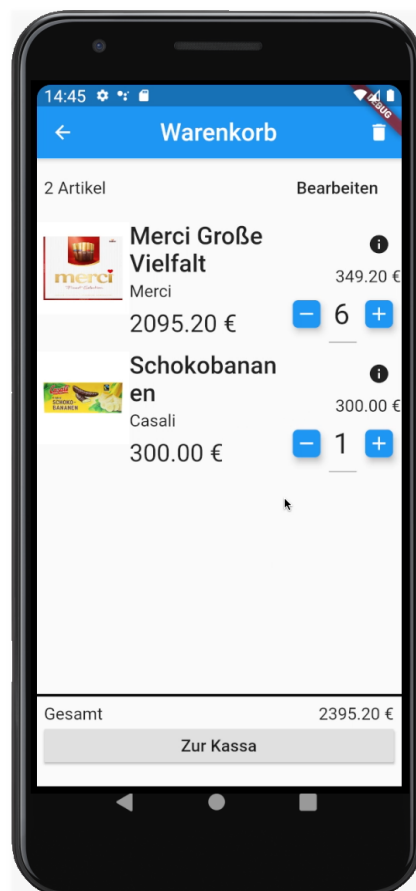


Figure 16. Beim Warenkorb sind beispielsweise das Datum und die Währung wichtig

[13] vgl. <https://mobilbranche.de/2014/08/typografie-beim-app-design>

[14] vgl. <https://medium.com/oberonamsterdam/how-to-create-an-accessible-app-and-why-you-should-5493f41f8bdb>

[15] vgl. <https://plusqa.com/2020/03/26/the-best-accessibility-apps-in-2020/>

[16] vgl. <https://www.infopulse.com/de/blog/internationalisierung-der-mobilen-apps-wege-und-methoden-zur-steigerung-der-einnahmen-um-26/>

Glossar

Backend

Schnittstelle zwischen der User-Applikation und einer Datenbank.

HTML

Ist die Grundstruktur, mit der Websites aufgebaut sind und Text dargestellt werden.

Wireframes

Ein visueller Design-Entwurf, wie Elemente in einer App dargestellt werden.

Click-Dummys

Ist eine reine Frontend-Applikation zum Testen, ohne Backend-Zugriff und komplexere Funktionen.

Cross-Plattform Apps

Ein Framework, welches dem Frontend ermöglicht auf iOS und Android lauffähig zu sein.

PWA

Progressive Web Apps sind Websites, welche zusätzlich als mobile Apps optimiert sind.

Quellen

Prenger C. (2019 14. August): matter-of-design, <https://matter-of-design.com/darum-ist-user-experience-so-wichtig> aufgerufen am 06.04.2021

Seobility: Usability, vgl. <https://www.seobility.net/de/wiki/Usability> aufgerufen am 07.04.2021

Wikipedia: Tim Berners-Lee, vgl. https://de.wikipedia.org/wiki/Tim_Berners-Lee aufgerufen am 07.04.2021

Technikum-wien: WAS IST UX-DESIGN?, vgl. <https://academy.technikum-wien.at/ratgeber/was-ist-ux-design> aufgerufen am 07.04.2021

Medium: UX/UI Design Process (Bild), <https://medium.com/nyc-design/ux-ui-design-process-for-beginner-753952bb2241> aufgerufen am 07.04.2021

Uxmatters: Mobile App UX Design Process, vgl. <https://www.uxmatters.com/mt/archives/2020/08/mobile-app-ux-design-process.php> aufgerufen am 07.04.2021

obileappdaily: Mobile App Design Guidelines to Build an App Users Love, vgl. <https://www.mobileappdaily.com/mobile-app-design-guidelines> aufgerufen am 07.04.2021

Careerfoundry: What Are User Interface (UI) Design Patterns? Everything You Need To Know, vgl. <https://careerfoundry.com/en/blog/ui-design/user-interface-patterns/#1-what-are-ui-design-patterns> aufgerufen am 07.04.2021

Darkpatterns: TYPES OF DARK PATTERN, vgl. <https://www.darkpatterns.org/types-of-dark-pattern> aufgerufen am 07.04.2021

Apple Inc.: App Store Review Guidelines, vgl. <https://developer.apple.com/app-store/review/guidelines> aufgerufen am 07.04.2021

Tigercolor: Basic color schemes - Introduction to Color Theory, vgl.

<https://www.tigercolor.com/color-lab/color-theory/color-theory-intro.htm>

aufgerufen am 07.04.2021

Tigercolor: Basic color schemes - Introduction to Color Theory (Bilder),

[https://www.tigercolor.com/color-lab/color-theory/color-theory-intro.htm#](https://www.tigercolor.com/color-lab/color-theory/color-theory-intro.htm#Primary_colors)

[Primary_colors](#) aufgerufen am 07.04.2021

Ionos: Corporate Identity, vgl. [https://www.ionos.at/startupguide/](https://www.ionos.at/startupguide/unternehmensfuehrung/corporate-identity)

[unternehmensfuehrung/corporate-identity](#) aufgerufen am 07.04.2021

Yeeply: App Design: Die Bedeutung der Typografie, vgl. [https://de.yeeply.com/blog/](https://de.yeeply.com/blog/app-design-die-bedeutung-der-typografie)

[app-design-die-bedeutung-der-typografie](#) aufgerufen am 07.04.2021

Mobilbranche: 3 typografische Fauxpas im App-Design., vgl.

<https://mobilbranche.de/2014/08/typografie-beim-app-design> aufgerufen am

07.04.2021

Medium: How to create an accessible app (and why you should), vgl.

[https://medium.com/oberonamsterdam/how-to-create-an-accessible-app-and-why-](https://medium.com/oberonamsterdam/how-to-create-an-accessible-app-and-why-you-should-5493f41f8bdb)

[you-should-5493f41f8bdb](#) aufgerufen am 07.04.2021

Plusqa: The Best Accessibility Apps in 2020, vgl. [https://plusqa.com/2020/03/26/the-](https://plusqa.com/2020/03/26/the-best-accessibility-apps-in-2020)

[best-accessibility-apps-in-2020](#) aufgerufen am 07.04.2021

Infopulse: Internationalisierung der mobilen Apps: Wege und Methoden zur

Steigerung der Einnahmen um 26%, vgl. [https://www.infopulse.com/de/blog/](https://www.infopulse.com/de/blog/internationalisierung-der-mobilen-apps-wege-und-methoden-zur-steigerung-der-einnahmen-um-26)

[internationalisierung-der-mobilen-apps-wege-und-methoden-zur-steigerung-der-](#)

[einnahmen-um-26](#) aufgerufen am 07.04.2021