

**OST**  
Ostschweizer  
Fachhochschule

**Part 3 & 4**

# **LLab1 / UX**

**HS2023**

# DESIGN & USABILITY

Part 3  
Design &  
USABILITY

In den ersten beiden Phasen des Wissensnuggets haben wir uns mit dem Usability Prozess (Stories und Use Cases) und dem Paper Prototyping befasst. Diese Outcomes dienen nun als Ausgangslage für die folgenden Schritte.

Wir betrachten die wichtigsten Prinzipien des UI Designs, um Anwendungen benutzerfreundlich umzusetzen. Dabei bewegen wir uns mit der Anwendung von Wireframes (Low Fidelity, geringe Genauigkeit) über Mockups bis hin zu Prototypen mit hohem Funktionsumfang (High Fidelity), also Prototypen mit grosser Genauigkeit mit dem Endprodukt.

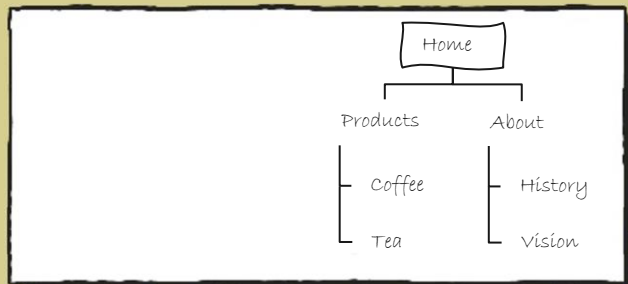
Während der ersten Phasen befassen wir uns mit der Informationsarchitektur sowie dem Papier Prototype unserer Anwendung. Nun möchten wir darauf aufbauen.

## Schritt 1

### Informationsarchitektur (IA) visualisieren:

Informationsarchitekturen werden häufig graphisch als hierarchisches Modell visualisiert. Könnte Ihre Informationsarchitektur aus der vorherigen Phase ebenfalls graphisch aufbereitet werden?

Optional: Skizzieren Sie ein Diagramm mit der IA Ihrer Anwendung ähnlich zur Vorgabe.



## Schritt 1: Die Informationsarchitektur aufbauen

Da wir nun wissen, wie die Inhalte der Anwendung zusammengehören, kann mit dem Entwurf der Benutzerschnittstelle begonnen werden.

Während der Blockwoche setzten Sie sich in der Benutzeranalyse intensiv mit den Eigenschaften der Benutzer und deren Bedürfnisse auseinander. Dabei kristallisierte sich im Verlauf der Definition der User Stories die Wunschplattform (beispielsweise Mobile App, Web-Applikation, usw.) für Ihre Anwendung heraus.

Jede Plattform definiert Gegebenheiten (wie Knöpfe, Gesten, Eingabegeräte, ...), welche man bei der technischen Umsetzung beachten sollte. So sind zum Beispiel die Bedienparadigmen für Mobile Apps nicht immer dieselben wie bei klassischen Windows-Applikationen oder Spielkonsolen: Die Interaktion erfolgt bei mobilen Apps mittels Touch, bei Windows vorwiegend mit Tastatur & Maus und bei Spielkonsolen mittels Game Controller.

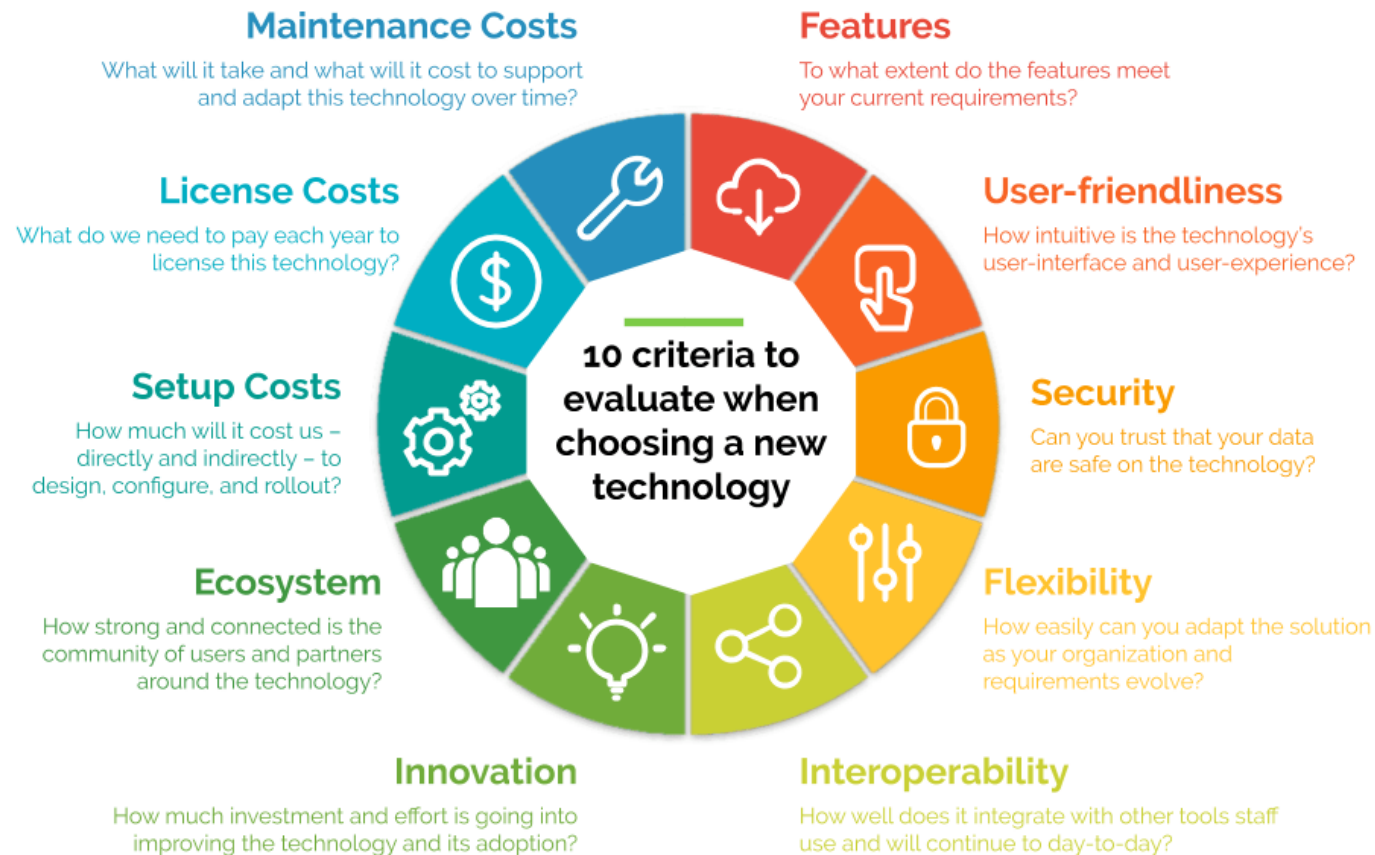
Auf solche plattformspezifischen Eigenheiten muss je fortgeschrittener der Prototyp ist, umso detaillierter eingegangen werden. Allerdings läuft man dadurch auch in Gefahr, kreative und innovative Lösungsansätze im Vornhinein durch «alte Zöpfe» abzuklemmen.

Im Folgenden widmen wir uns der Evaluation einer möglichen Plattform und Technologie. Eine abschliessende Einschätzung kann hier nicht gegeben werden, da jedes Projekt individuelle Anforderungen stellt.



# Design & Usability

## Plattformauswahl: 10 Kriterien (verasolutions.org)



DESIGN

# PLATTFORM

Part 3  
Design &  
USABILITY

Fragekatalog für die Technologieauswahl:  
Bearbeiten Sie zuerst das Wissensnugget «Übersicht  
digitale Technologien» und beantworten Sie im Anschluss  
die nachfolgenden Fragen.

Schritt 1

Wird die gewählte Technologie unsere aktuellen  
Anforderungen (User Stories und Use Cases) abdecken?

Schritt 2

Wird die gewählte Technologie unsere zukünftigen  
Anforderungen erfüllen?

Schritt 3

Können wir mit der gewählten Technologie unsere Zeit-  
und Kostenvorgaben einhalten?

Schritt 4

Definiert die gewählte Technologie ein Designsystem,  
Styleguides (siehe [UX Style Guide einfach erklärt](#) von Indra  
Burkart) oder andere Regeln bezüglich der User Experience?

Schritt 1: Die funktionalen Anforderungen  
abklären

Schritt 2: Die zukünftigen Anforderungen  
abklären

Schritt 3: Die Kostenvorgaben im Blick  
behalten

Schritt 4: Designvorgaben der  
Technologie evaluieren

Die Evaluation der Technologie stellt die Basis für Ihre weitere Vorgehensweise bezüglich des Designs der Anwendung dar. Viele der namhaften Hersteller definieren für ihre Technologien Styleguides und Best Practices. Anbei seien einige der bekanntesten Styleguides genannt:

- **Apple Human Interface Guidelines (HIG)**  
«The HIG contains guidance and best practices that can help you design a great experience for any Apple platform.»  
<https://developer.apple.com/design/human-interface-guidelines/guidelines/overview/>
- **Microsoft Fluent Design System**  
«Fluent is an open-source, cross-platform design system that gives designers and developers the frameworks they need to create engaging product experiences—accessibility, internationalization, and performance included.»  
<https://www.microsoft.com/design/fluent/>
- **Google Material Design**  
«Material Design is a design system built and supported by Google designers and developers. Material.io includes in-depth UX guidance and UI component implementations for Android, Flutter, and the Web.»  
<https://m3.material.io/get-started>

## Schritt 5

Falls die evaluierte Technologie einen Styleguide beschreibt, lesen Sie die Topics zur Rasterung (Grid) und zu Menus (teils auch Navigation genannt) durch. Lassen Sie die Ideen der Styleguides in die weiteren Arbeitsschritte einfließen. Somit können Eigenheiten der Technologie früh miteinbezogen und Arbeitsaufwand reduziert werden.

Beispiel: Das nebenstehende Bild zeigt eine Rasterung, welche über die OST-Homepage gelegt wurde.

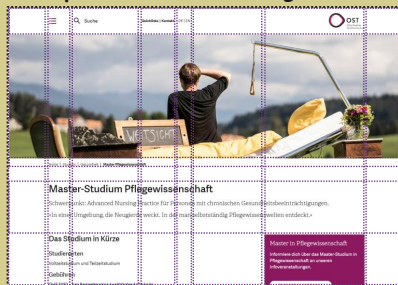
## Schritt 5: Vorhandene Guidelines auf Ideen zur Strukturierung (Grid/Menu/Navigation) prüfen



Anwendungen bestehen aus verschiedensten Interaktionselementen (auch Bedienelemente genannt) und Multimedia-Elementen sowie zahlreichen Teilbereichen wie Navigation, Unternavigation und Menüs. Sollen all diese Elemente auf einer Seite platziert werden, sind Rastersysteme nützlich.

Die Raster selbst werden vom Benutzer unbewusst wahrgenommen, da diese häufig nicht direkt sichtbar sind, trotzdem aber das Auge mittels der Anordnung der Interaktions- und Multimedia-Elemente führen. Das Design wirkt somit aufgeräumt und hochwertig. Zugleich ermöglichen Rastersysteme eine Wiedererkennbarkeit des Designs oder Produkts, auch über Gerätegrenzen hinaus.

Als Gestaltungs raster werden zuerst Führungslinien (Hilfslinien) gezeichnet, auf dessen Basis die Interaktions- und Multimedia-Elemente ausgerichtet werden. Im Web-Bereich werden häufig Raster auf Basis von 12, 16 oder 24 Spalten (Columns) erstellt. Der Zwischenabstand zwischen den Spalten wird Gutter genannt.

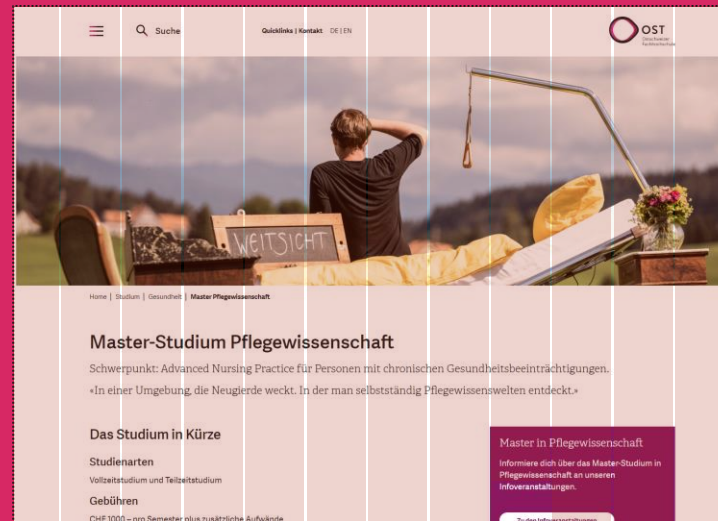


## Schritt 6

**Das Raster:** Schauen Sie sich [das Video von Jesse Showalte](#) an. **Tip:** schalten Sie unter CC die Untertitel ein und wählen Sie unter ⚙ Settings > Subtitles (CC) die Sprache «Deutsch» aus.

Legen Sie im Anschluss ein Raster auf Ihren Paper Prototype. Alternativ können Sie ein komplett neues Grid System (Gestaltungsraster) erstellen, welches zu Ihrer Anwendung und der evaluierten Technologie passt.

## Schritt 6: Gestaltungsraster erstellen



Wireframes werden häufig als «Low Fidelity» Prototypen erstellt und im Verlauf des Projekts zu einem detailgetreueren Prototyp ergänzt. Da wir bereits über einen validierten «Low Fidelity» *Paper Prototype* sowie ein Gestaltungsraster verfügen, können wir unsere Anwendung auf diesen Ideen aufsetzen.

Im folgenden Schritt fügen wir also die Erkenntnisse aus den bereits angefertigten Artefakten *Paper Prototype* (strukturell, teils auch funktional orientierter Prototyp), *Informationsarchitektur* (Navigation, Inhaltsgliederung) sowie der *Rasterung* in einen neuen Prototyp zusammen.

Das daraus resultierende, genauere Wireframe kann bereits eine erste Diskussionsbasis für den Kunden bezüglich der zukünftigen Benutzeroberfläche bieten. Die Kunden und Benutzer können sich also ein erstes Bild der Applikation machen. Entsprechend soll dieses bereits Layout und Navigation zeigen, aber nach wie vor «Prototyp»-Charakter haben.

Wahlweise wird dieses genauere Wireframe mit Papier und Bleistift oder auch digital erstellt. Analoge Hilfsmittel haben zum Vorteil, dass diese sich sehr einfach während der Diskussion mit den Kunden und Benutzern interaktiv weiterentwickeln lassen.

Da wir aber bereits einen *Paper Prototype* vorliegend haben, werden wir im nächsten Schritt direkt ein digitales Tool zur Umsetzung dieses Wireframes verwenden.

Für die Modellierung der Wireframes können Sie ein beliebiges digitales Tool wie [draw.io](#), [Sketch](#), [balsamiq.cloud](#), [figma](#), [miro](#), [InVision](#), [justinmind](#) oder Adobe Suite mit XD/Illustrator/Photoshop bzw. Microsoft Office mit Visio/PowerPoint verwenden. Diese Liste ist nicht abschliessend und soll lediglich eine mögliche Auswahl geben.

Schauen Sie sich [das Video von Indra Burkart](#) an. Sie beschreibt die Wireframes auf «Low Fidelity» Ebene; übertragen Sie die beschriebenen Techniken auf Ihren Anwendungsfall.

## Schritt 6

**Wireframing:** Erstellen Sie ein digitales «Low Fidelity» Wireframe Ihrer Anwendung.

Verwenden Sie die Erkenntnisse der bereits erstellten Artefakten, bzw. digitalisieren Sie nun Ihren *Papier Prototype* im gewählten Tool. Je detailgetreuer das Wireframe aber wird, umso mehr tendiert dieses zu einem «Medium Fidelity» Wireframe.

Die effektive UI Design-Phase mit Bildern und der echten Farbpalette setzt erst später ein. Man versucht somit die aufwändigeren «High Fidelity» Prototypen auf einem stabilen Grundgerüst aufzubauen. Verrinnerlichen Sie sich zum Thema [Wireframe vs Mockup das Video](#) von Indra Burkart.

Tip: Lassen Sie auch die digitalen Wireframes durch den Benutzer validieren; somit erhalten Sie ein Feedback, ob Ihre Ideen bezüglich der Anordnung der Elemente sowie der Navigierbarkeit funktionieren. Je nach Komplexität der Anwendung kann dieser Schritt durchaus mehrere Iterationen umfassen.

Als digitales Wireframing Tool dürfen Sie - wie bereits beschrieben - ein beliebiges Programm auswählen. An dieser Stelle soll aber noch erwähnt werden, dass es sich lohnt, ein Tool auszuwählen, welches sämtliche Typen von Wireframes unterstützt: Also von «Low Fidelity» bis hin zu komplett gestalteten Prototypen. Ansonsten besteht die Gefahr, dass Sie sich in mehrere Programme einarbeiten müssen.

Falls Sie noch kein digitales Wireframing Tool kennen, schauen Sie sich zuerst den nächsten Schritt an, wo wir [figma](#) genauer betrachten.

## Schritt 6: «Low Fidelity» Wireframe zeichnen

# figma

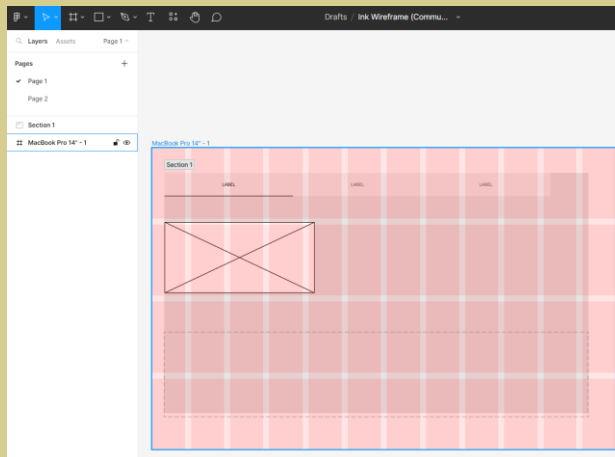
Part 3  
Design &  
USABILITY

In der Kategorie der durchgängigen UI Prototyping Tools hat sich **figma** bewährt. Falls Sie sich für **figma** entscheiden, schauen Sie sich das **figma-Tutorial von Indra Burkart** als Grundlage sowie das **figma Frame Tutorial für das Erstellen von Grids** an.

## Optional

**figma Education:** Starten Sie im Anschluss mit einem neuen Design («**New design file**» auf der Startseite). Für ein einfaches Wireframe sollten die vorhandenen Werkzeuge ausreichen; später werden wir voraussichtlich weitere figma Plugins benötigen.

Als Student sind Sie qualifiziert für eine gratis figma Education-Lizenz, welche sämtliche Pro-Features enthält. Erstellen Sie dazu **einen figma-Account** und verifizieren Ihren Account unter **Figma Education plan** für die freie Lizenz. Geben Sie als Schulname **Eastern Switzerland University of Applied Sciences** und als Webseite **<https://www.ost.ch>** an.



## Optional: figma Education einsetzen

# MOCKUP & VISUAL DESIGN

Sie haben nun eine konkretere Vorstellung, wie Ihre Anwendung aussehen soll, wie die Interaktions- und Multimedia-Elemente platziert werden, sowie über welche Funktionalität Ihre Anwendung verfügen soll.

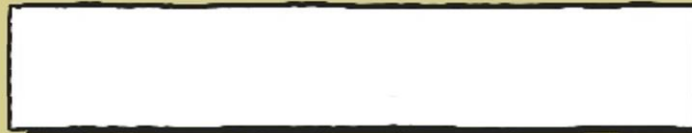
In diesem Schritt wird auf die Farbgebung sowie die visuelle Gestaltung der Benutzeroberfläche eingegangen, sodass schlussendlich ein «High Fidelity» Prototyp resultiert.

Bevor wir aber zur künstlerischen Ausgestaltung der Benutzerschnittstelle übergehen, sollen im Folgenden die 10 Heuristiken nach Nielsen & Molich betrachtet werden. Diese bilden eine gute Basis für die Entwicklung benutzerfreundlicher Anwendungen.

## Schritt 1

Schauen Sie sich [das Video von Indra Burkart zum Thema Heuristiken](#) nach Nielsen und Molich an.

Welche Verbesserungsmöglichkeiten sind erkennbar und wie könnten die Heuristiken helfen, Ihre Anwendung noch intuitiver zu machen? Könnten diese Erkenntnisse bereits im Wireframe umgesetzt werden?



Prinzipiell sollen Ihnen die Heuristiken von Nielsen und Molich eine Checklist in die Hand geben, wie Sie Ihre Benutzerschnittstelle nach den gängigen Usability-Prinzipien validieren können. Sie können die Heuristiken auch später, im weiteren Verlauf des Projekts, wieder zur Hand nehmen und Ihre Benutzeroberfläche damit validieren.

## Schritt 1: Heuristiken von Nielsen Molich anwenden

# Usability Guidelines

Umsetzungsideen: **10 Usability Heuristiken** (Nielsen & Molich, 1995)

- (1) Sichtbarkeit des Systemstatus
- (2) Übereinstimmung von System und Wirklichkeit
- (3) Nutzerkontrolle und Freiheit
- (4) Beständigkeit und Standards
- (5) Fehlervermeidung
- (6) Wiedererkennung statt Erinnerung
- (7) Flexibilität und Effizienz
- (8) Ästhetisches und minimalistisches Design
- (9) Hilfestellung beim Erkennen, Bewerten und Beheben von Fehlern
- (10) Hilfe und Dokumentation

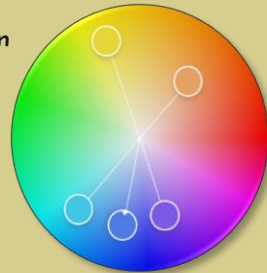
# FARBEN

Farben sind etwas emotionales, so sind die Meinungen zur Lieblingsfarbe weit gestreut. Trotzdem gibt es einige Kriterien, welche uns helfen die richtige Farbgestaltung für unsere Anwendung zu wählen.

Die Farbauswahl kann einerseits das Benutzererlebnis besser gestalten, andererseits muss darauf geachtet werden, die Farbgestaltung aufs Wesentliche zu beschränken, um die Übersicht zu gewährleisten. So wählt man für eine gute Farbpalette einer Applikation eine Farbe mit zwei ergänzenden Farben für Abstufungen und potentiell eine Komplementärfarbe.

Die Komplementärfarbe sollte aber nur sehr spärlich für hervorzuhebende Elemente verwendet werden.

Beispielsweise wurde im Farbkreis rechts als Grundfarbe **blau** mit den Abstufungsfarben **violett** und **türkis**, sowie eine Komplementärfarbe (gelb oder orange) gewählt.



Hier dient als Hilfsmittel das **Adobe Color Wheel** in der **Double Split Complementary** Option.

Um den Kontrast zu gewährleisten, sollten die eingesetzten Farbtöne nicht zu nahe aneinander liegen; auch in Bezug zur Hintergrundfarbe. Ebenfalls gilt es zu beachten, dass viele Menschen die roten und grünen Farbtöne schwer unterscheiden können. Sie können dem entgegenwirken, indem den Farbtönen mehr Blau beigemischt wird.

Beachten Sie für die Farbenwahl Ihrer Anwendung ebenfalls die Zielgruppe. So sind die Farben je nach Kulturkreis, Geschlecht, Alter, Mode, usw. unterschiedlich zu wählen. Denn Farbe kann Ihre Anwendung nicht nur schöner, sondern auch unbenutzbar machen und sogar Aversionen bei den Benutzern hervorrufen. Heute versucht man daher durch minimalistische Farbwahl punktuell Elemente einer Anwendung hervorzuheben.

MOCKUP

# COLOR THEMES

Part 4  
Design &  
USABILITY

Schritt 2

Viele Styleguides definieren bereits mögliche Farbpaletten (Themes) mit Kontrastfarben und ebenfalls Dark Mode Themes (Farbpaletten basierend auf einer dunkeln Hintergrundfarbe).

Erkunden Sie sich als Erstes: Definiert Ihre ausgewählte Plattform in dem Styleguide bereits Farbpaletten, welche Sie verwenden könnten?

Wählen Sie ein für Ihre Benutzergruppe eine passende Farbpalette aus. Verwenden Sie das [Adobe Color Wheel](#), falls Sie die Farbpalette selbst gestalten möchten.

Notieren Sie die RGB-Codes der gewählten Farben, Sie werden diese später für Ihren Prototypen wieder benötigen.



Tip: Falls Sie sich nicht sicher sind, wie Sie die Farben auswählen sollen, können Sie auch auf die [Farbgestaltungsmöglichkeiten von Google's Material Design](#) zurückgreifen.

Halten Sie sich bei Ihrer Farbwahl an die Regel **Ästhetisches und minimalistisches Design** nach Nielsen & Molich.

## Schritt 2: Farbpalette auswählen

Als nächstes befassen wir uns mit der Gestaltung der Benutzeroberfläche in Bezug auf die Schriftart und Textformatierung. Dabei geht es um ausreichend Kontrast zum Hintergrund, der Schriftgröße sowie der der gewählten Schriftart.

Für die optimale Dekodierbarkeit sollte der Text auf einem kaum gemusterten, möglichst einfarbigen Hintergrund dargestellt werden. Je höher der Kontrast, umso lesbarer wirkt die Schrift (vgl. vorheriges Kapitel).

Die Schriftgröße hängt stark von den Lichtverhältnissen sowie vom Leseabstand ab. So müssen Texte auf einer Anzeigetafel im öffentlichen Verkehr wesentlich grösser dargestellt werden als auf Mobile Devices.

Weiterhin gelten **Frakturschriften**, **Schreibschriften** oder **Schriften mit übermässigen Verzerrungen** als schwer lesbar und sollten für längere Absätze (Fliesstext) vermieden werden. Werden bei Fliesstext solche Schriften verwendet, ist davon auszugehen, dass die Benutzer die Lektüre potentiell früher abbrechen. Für Blickfänge (Key Visuals) in Überschriften, Banners, usw. kann aber auch auf unterschiedliche oder sogar auffällige **Schriftarten** zurückgegriffen werden.

Ebenfalls gibt es auf die Frage «*Was ist eine gute Leseschrift?*» keine eindeutige Antwort; es gibt viele Faktoren zu beachten. Je nach Schrifttyp wirken beispielsweise Texte unbewusst vertrauenswürdiger; schlecht Abschnitten gemäss einer Untersuchung der New York Times comicartige Schriften, sowie Georgia oder Helvetica.

Texte in Helvetica und **Verdana** wirken eher modern, Fliesstext in Times New Roman und Georgia eher seriös und gediegen. Grundsätzlich sollte diejenige Schriftart gewählt werden, welche sich die Benutzer gewöhnt sind. So wird auf Android Systemen häufig die Schriftart Roboto (in verschiedenen Varianten) eingesetzt.

## Moderne Schriften:

Calibri  
Helvetica  
Open Sans  
Roboto  
Segoe UI  
Verdana

## Seriöse Schriften:

Baskerville  
Bodoni  
Cambria  
Georgia  
Lucida Bright  
Times New Roman

Eine angenehme Schrittgrösse für Fliesstexte ist 16px (Pixel), kleinere Schriftgrößen gelten als veraltet. In modernen Browsern ist die Standard-Schriftgrösse beispielsweise auf 16px festgelegt.

Wichtig ist aber zu beachten, dass 16px je nach Gerät eine andere Grösse darstellt. So hat beispielsweise ein 21.5" (=55cm Diagonale) Monitor eine Auflösung von 1920x1080px (=2073600 Pixels). Ein Smart Phone Screen enthält bei einer Diagonalen von 6.1" (=15.5cm) bis zu 1080x2340px (=2527200 Pixels). Entsprechend viel kleiner werden auf dem Smart Phone 16px dargestellt.

Es muss also bei der Wahl der Schriftgrösse auf die Schriftart, den Abstand des Auges zum Screen und auf die Pixeldichte geachtet werden. Falls keine Best Practices von der Technologie vorliegen, sind effektiv Usability Tests vonnöten – bei welcher Grösse schenkt der Benutzer den dargebotenen Texten ausreichend Beachtung?

Hervorhebungen in Texten wie ALLES IN GROSSBUCHSTABEN oder Unterstreichung sind zu vermeiden. Grossbuchstaben wirken sehr unleserlich; unterstrichene oder farblich hervorgehobene Wörter in Anwendungen assoziieren viele Anwender mit einem Link. Besser funktionieren Hervorhebungen in Texten mit fettgeschriebenen oder kursiven Textstellen.

## Schritt 3

Definieren Sie die Schriftart und die Schriftgrößen für Ihre Applikation. Gehen Sie von der Standard-Textgrösse von 16px aus; die Überschriften sollten etwas grösser/markanter sein als der Fliesstext und können sich auch farblich vom Fliesstext unterscheiden.

Beachten Sie wiederum die Styleguides Ihrer Technologie: Gibt es Vorgaben in der Schriftgrösse sowie bei der farblichen Gestaltung von Überschriften?

## Schritt 3: Schriftstil evaluieren

Wir möchten nun mit unserem Design konkreter werden und die Schriften und Farben im Wireframe umsetzen.

**Schritt 4**

Kopieren Sie Ihr bestehendes Wireframe (in [figma das Frame kopieren](#)) und ergänzen Sie die Farben, Schrifttypen und Schriftgrößen; wenden Sie diese auf Schaltflächen, Überschriften, Links usw. an. Ergänzen ebenfalls Textstellen, welche bereits echten Inhalt zeigen.

Optional: In figma können Sie die Farben und Schriften organisieren. Lassen Sie sich vom UI Design von Indra Burkart im [Tutorial Styles - Farben, Text und Effekt](#) inspirieren: Welche Elemente wurden wie hervorgehoben?



Tip: Spielen Sie die neuen Wireframes mit den Benutzern durch; wie reagiert die Kundschaft auf die neuen Farben und veränderten Schriftgrößen? Werden die Textstellen erfasst und ist der Lesefluss sichergestellt?

## Schritt 4: Schriften und Farben ins Wireframe integrieren

MOCKUP

# TECHNOLOGIE

Part 4

Design &  
USABILITY

Zu guter Letzt möchten wir nun den Prototypen der Benutzerschnittstelle in einem hohen Reifegrad fertigstellen. Dieser Prototyp ist sehr stark an die Technologie gebunden, da diese meist unterschiedliche Mensch-Maschine-Interaktionsmöglichkeiten bieten.

Schritt 5

Arbeiten Sie den Styleguide für Ihre Technologie und Plattform (Zielsystem) durch. Durchforsten Sie diese nach den vordefinierten Interaktionselementen (auch Komponenten, engl. Components genannt).

Falls Sie keinen Styleguide finden, durchforsten Sie die Dokumentation der Wunschtechnologie nach Best Practice bezüglich der Benutzerinteraktion. Welche Möglichkeiten zur Interaktion (beispielsweise Buttons / Labels / ...) mit der Technologie werden vom Hersteller/Standard bereits angeboten? Wo finden diese Interaktionselemente in Ihrem Wireframe Anwendung?

DESIGN &  
USABILITY

## Schritt 5: Interaktionsmöglichkeiten der Technologie sichten

# BILDER & ICONS

Im Folgenden befassen wir uns mit einigen Oberflächen-Elementen, welche bei allen Technologien vorkommen und was es dabei zu beachten gilt.

So können Bilder stark zur guten Benutzbarkeit von Anwendungen beitragen. Ein Bild kann eine Zeichnung, ein Foto, ein Diagramm oder eine grafische Darstellung von Daten sein. Ein Icon (Piktogramm) ist eine stark vereinfachte Darstellung eines Objekts und wird häufig für bedienbare Flächen (Buttons / Links) eingesetzt.

Bilder sind vor allem dann hilfreich, wenn sie dem Benutzer Inhalte vermitteln: Beispielsweise ein Produktbild in einem Web-Shop, oder eine schematische Darstellung in einer Bedienungsanleitung.

Negative Beispiele sind generische, rein zur Ästhetik platzierte Fotos: Diese sind leicht auswechselbar und haben häufig keinen direkten Bezug zum Inhalt. Benutzer haben gelernt, solche Bilder zu ignorieren. Entsprechend haben diese keinen Mehrwert und können sogar die Usability durch lange Lade- und Wartezeiten negativ beeinflussen.

Dem gegenüber stehen die klaren Vorteile von Bildern: Sie schaffen es, den Nutzer emotional anzusprechen (sogenannte *Hero Shots*) und die Aufmerksamkeit zu lenken. Zudem können Bilder Texte auflockern, inhaltlich ergänzen und helfen so Missverständnissen vorzubeugen.

Gute Bilder sind inhaltsbezogen zur Seite, scharf und kontrastreich, nicht mit Informationen überladen und zeigen ein zentrales Motiv.



Icons sind aus den Navigationskonzepten nicht mehr wegzudenken. Die Bedeutung der Icons ist häufig angelernt, d.h. das Verhalten des Programms wird erst durch eine erstmalige Betätigung sichtbar (beispielsweise Menu Button ☰). Trotzdem lassen sich solche Icons in verschiedenste Applikationskontexte zielführend transferieren.

Gerade hier lauern auch die Gefahren: Werden Icons missverständlich oder gar falsch eingesetzt (beispielsweise ☰ schliesst die App), ist der Einsatz des Icons problematisch. Der Benutzer hat eine Erwartung an das ☰ Icon, welche somit nicht erfüllt wird und in einer potentiellen Fehlbedienung resultiert.

Verinnerlichen Sie sich das Kurzvideo von Indra Burkart zum Thema [Icon Design Basics für gute UX](#).

Viele der Design-Sprachen definieren ein Iconset. Prüfen Sie Ihren Styleguide, ob dieser bereits Icons vordefiniert. Beispielsweise können die [Material Design Icons von Google](#) für Android (oder Web/iOS) direkt online abgerufen und in die eigene Applikation übertragen werden. Bitte beachten Sie jeweils den Icon Namen, welcher den Anwendungsfall beschreibt.

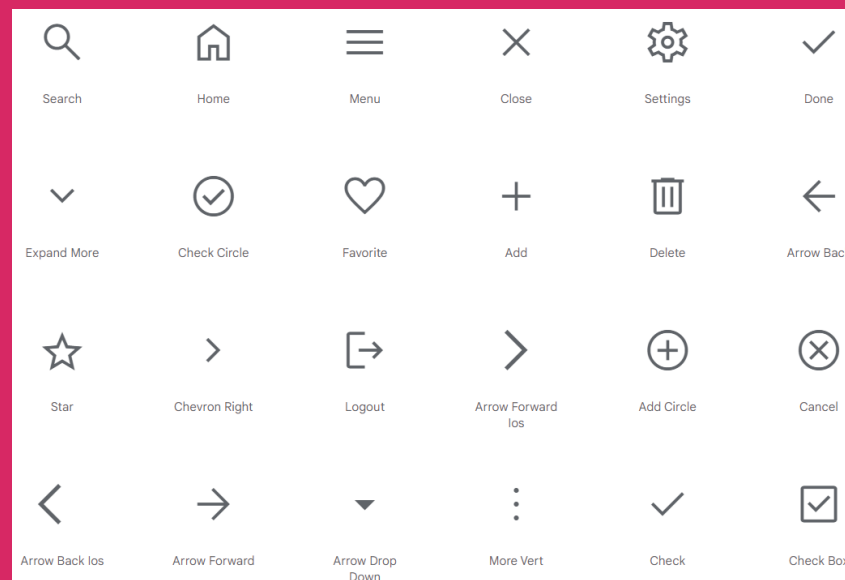
In Ihrem figma-Projekt können Sie beispielsweise mit den [Material Design Icons](#) arbeiten (siehe Button rechts oben, [Open in figma](#)) oder auch das [Font Awesome Plugin](#) installieren (siehe Button rechts oben, [Try it out](#)), welches sehr viele Icons enthält. Hinweis: Nicht alle [Font Awesome Icons sind gratis](#).

Schritt 6

Überarbeiten Sie Ihr Wireframe, indem Sie die zu Ihrer Technologie passenden Icons auswählen und die Platzhalter der Bilder durch effektive Fotos/Abbildungen ersetzen.

## Schritt 6: Bilder in Wireframe einbinden und Icons auswählen

### Auszug aus den Material Design Icons:



# LINKS & BUTTONS

Weitere, immer wieder gesehene Benutzerinteraktions-Elemente sind Buttons (Schaltflächen / Knöpfe) und Links.

Buttons sehen markanter aus, sind häufig farblich hinterlegt und sollen einen Angebotscharakter haben; beispielsweise **bezahlen**, **löschen**, **speichern** usw. Links und Buttons können zusätzlich mit Icons verdeutlicht werden.

Häufig werden auch Abstufungen zwischen den Buttons gemacht, also einen Button in der **Hauptfarbe** und einen Button in der **Sekundärfarbe**. Diese Hierarchie erleichtert es dem Benutzer, die wichtigen primären Aktionen zuerst wahrzunehmen.

Eine weitere, untergeordnete Hierarchiestufe stellen die **Links** dar. Links sollten primär zur Navigation zwischen den Seiten verwendet werden. Im Fliesstext verlinkte Stellen werden **farblich** hervorgehoben sowie unterstrichen formatiert. Links haben aber das Potential den Lesefluss zu beeinträchtigen. Daher sollte auf exzessive Verlinkungen im Fliesstext verzichtet werden. Hier kann eine Linkliste am Ende des Absatzes Abhilfe schaffen.

Bei Links ist es üblich, diese so zu beschriften, dass der User das Ziel der Navigation kennt. Beispielsweise sind Linkbeschriftungen wie «**mehr**» oder «**hier klicken**» schlecht verständlich und das Ziel unklar. Anstatt «Die Details zur Kamera **finden Sie hier**» könnte der Linkbescrieb «**Kamera DSC-W500 – Datenblatt**» heissen.

Je nach Zielsystem/Plattform und Technologie müssen weitere Aspekte beachtet werden: Soll der Benutzer die Bedienelemente per Touch (Finger) bedienen können, sind grössere Bedienflächen erforderlich. Ist die Bedienung mit einem (Game) Controller vorgesehen, kann diese Interaktionen mit Techniken wie «Auto Aiming» oder auch Skip Links (direktes Navigieren auf wichtigste Inhalte) vereinfacht werden. Es gilt auch hier die gewohnten Bedienungskonzepte der Zielplattform zu beachten und einzuhalten.

## Schritt 7: Wireframe erweitern um Navigiermöglichkeit mit Links und Buttons

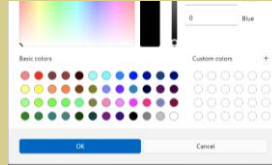
MOCKUP

# PROTOTYPING

Part 4  
Design &  
USABILITY

Auch die Platzierung der Buttons ist je nach Plattform unterschiedlich.

Auf Windows-Systemen ist die Anordnung der Aktions-Buttons wie folgt: **OK** **Abbrechen**  
Die Begründung liegt in der Leserichtung: die wichtigste Aktion wird zuerst Erfasst.



Bei Android- und iOS-Geräten ist die Reihenfolge gedreht: **Abbrechen** **OK**, denn die letzte Option wird vom Auge zuletzt erfasst und kann so direkt angewählt werden.

Schritt 7

Erweitern Sie Ihr Wireframe um passende Links und Buttons. Versuchen Sie sich ebenfalls an die Styleguides Ihrer Plattform zu halten.

Ergänzen Sie ebenfalls die Navigationsmöglichkeit zwischen den Wireframes, falls Sie bereits mehrere Ansichten gezeichnet haben. Aus Ihrem Wireframe wird somit ein klickbarer Prototyp.

Für figma hat Indra Burkart wiederum ein [Tutorial fürs Prototyping](#) zusammengestellt. In diesem erklärt sie, wie sich einzelne Frames mit figma einfach verlinken lassen.

Tip: Gerade die Navigationselemente sind essentiell für eine gute Usability. Entsprechend macht es Sinn, auch diesen neuen Prototypen mit dem überarbeiteten Navigations- und Interaktionskonzept durch den Benutzer validieren zu lassen.

## Schritt 7: Wireframe erweitern um Navigiermöglichkeit mit Links und Buttons

Im letzten Schritt werden Sie die Design-Prinzipien aus der gewählten Technologie in Ihren Prototypen übertragen. Schlussendlich sollten Sie einen «High Fidelity» Prototyp – ein sogenanntes UI Mockup – erhalten.

Nehmen Sie Ihre Notizen aus Schritt 5 zur Hand, bei welchem Sie die Bedienelemente Ihrer Technologie angeschaut hatten.

Gibt es Überschneidungen, wo Sie eigene Elemente gezeichnet haben, diese aber durch Interaktionselemente des Styleguides abgedeckt sind? Könnten diese selbst definierten Elemente einfach ersetzt oder mit den Elementen des Styleguides ergänzt werden?

**Schritt 8**

**UI Mockup:** Überarbeiten Sie das Wireframe im Anschluss mit den Interaktionselementen Ihrer evaluierten Technologie. Beachten Sie beim Überarbeiten auch die 10 Heuristiken nach Nielsen & Molich.

Für häufig angewendete Technologien bietet die figma-Community viele Plugins mit Design Kits an, mit welchen Sie Zugriff auf deren Interaktionselemente erhalten, ohne dass Sie Elemente selbst gestalten müssen. Falls Sie figma verwenden, schauen Sie sich das Tutorial zu [Design Kits als Bibliothek nutzen](#) an.

In diesem Wissensnugget entwickelten Sie auf Basis des Gestaltungsprozess nach Garrett und ISO 9241-210 einen Paper Prototype und bauten diesen zu einem echten UI Mockup aus. Das Design sowie die Interaktionskonzepte validierten Sie in mehreren Iterationen zusammen mit dem Benutzer beispielsweise mittels A/B-Tests. Durch zusätzliches Einhalten der Styleguides und Heuristiken kann eine gute Usability sichergestellt werden.

So werden sich schlussendlich der Kunde, der Benutzer und das Entwicklungsteam einig sein:

**«Ja, so realisieren wir die App!»**

## Schritt 8: Mockup fertigstellen



WELL DONE!

# Weiterführende Wissensnuggets

## Low-Code

- **01.4**  
Einführung in die Low Code basierte Softwareentwicklung
- **02.1.2**  
Low Code Anwendungen mit PowerApps und PowerAutomate

## Pro-Code

- **02.1.1**  
Cloud-basierte Anwendungen
- **01.6**  
3D UX Design für VR/AR