

4. Wireflow und Key Screen

Artikel von Mike Angeles (mike@balsamiq.com)
aus dem englischen sinngemäss übersetzt

Wireflows einer Benutzerführung

Ein Wireflow ist ein hybrides Designdokument, das Wireframing mit User Flows kombiniert. Sie sind im Wesentlichen Wireframes, die Benutzerführung und Systemablauf zeigen.

Durch einfaches Hinzufügen von Pfeilen und Anmerkungen zwischen den Wireframes (Screens) werden die Pfade angezeigt, die ein Benutzer bei der Verwendung einer Anwendung einschlagen kann. Diese kleinen Änderungen an Wireframes kommunizieren die visuellen Änderungen an der Mensch-Maschinen-Schnittstelle und beschreiben gleichzeitig Zustandsänderungen und Sequenzen in der Benutzerinteraktion.

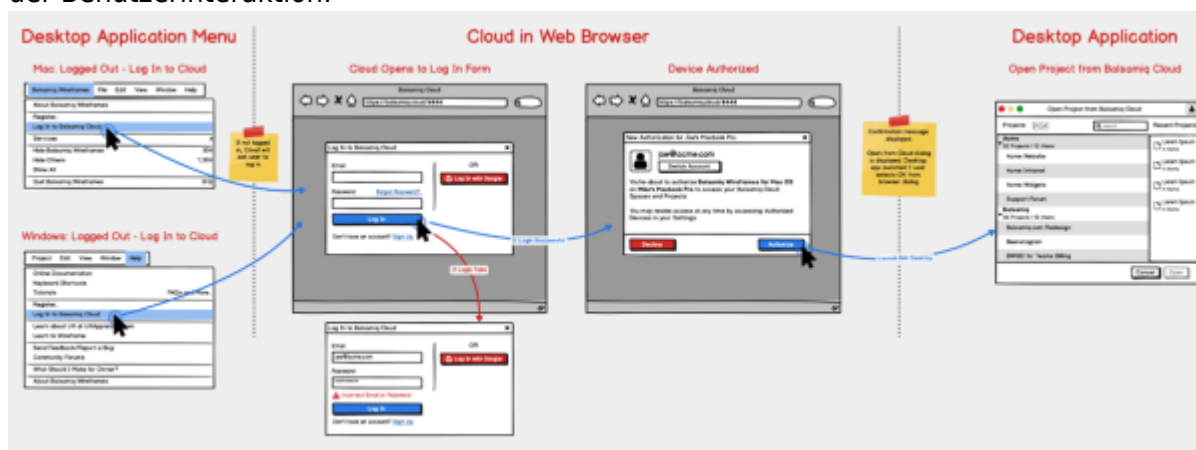


Abb. 6-5: Beispiel eines Wireflows

In diesem Text wird aufgezeigt, wie diese Technik im Designprozess eingesetzt werden kann und warum Wireflows ein gut geeignetes Werkzeug für die gleichzeitige Kommunikation eines Schnittstellen- und Interaktionsdesigns sind.

Grundlegende Elemente von Wireflows

Wir beginnen mit einem Vergleich von Wireframes und User Flows zur Gestaltung von Abläufen und fahren dann mit der Gestaltung eines Wireflows fort.

Wireframes

Wireframes (auch als Key Screens bezeichnet) eignen sich gut für die Darstellung statischer Schnittstellen.

Die Daten und das Layout einer Schnittstelle ändern sich jedoch häufig, wenn Benutzer mit einem Produkt interagieren, so dass ein einzelnes statisches Wireframe die Komplexität dieser Änderungen möglicherweise nicht gut darstellt.

Das folgende Beispiel zeigt eine Seite für ein Produkt auf einer Shopping-Website.

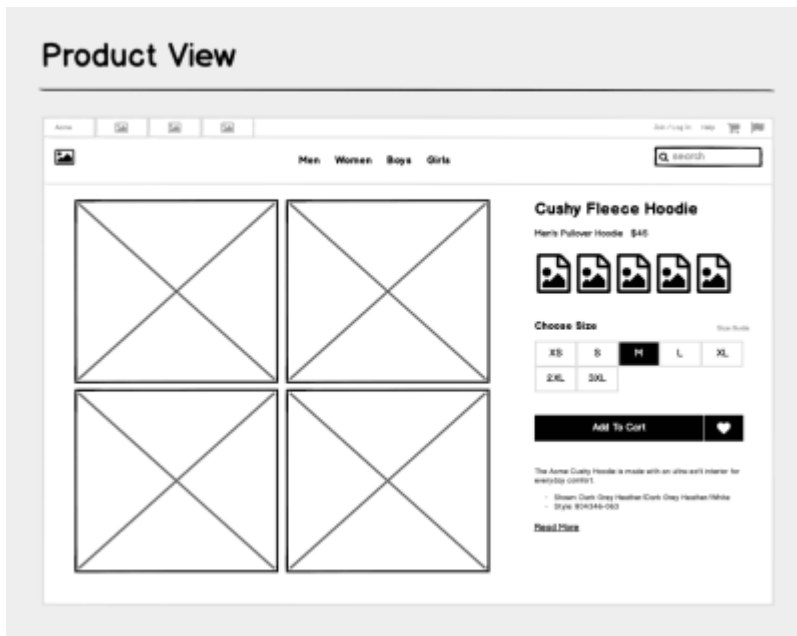


Abb. 6-6: Wireframe einer Anwendung

Dieser Wireframe vermittelt eine gute Vorstellung davon, welche Elemente auf den Produktseiten eines Shops erscheinen und wo sie angeordnet werden sollten. Einige dieser Oberflächenelemente lösen jedoch Aktionen aus. Zum Warenkorb hinzufügen (Add to Cart) ist hier ein offensichtlicher Auslöser. Es muss daher explizit festgelegt werden, was passiert, wenn der Benutzer diese Aktion auslöst. Um dies zu tun, könnte ein Produktdesigner typischerweise jeden geänderten Zustand drahtlos darstellen, wenn der Benutzer die Kaufabwicklung durchläuft.

User Flows

User Flows (Nutzerfluss) können für die Gestaltung von Interaktionen verwendet werden.

Es gibt mehrere Arten von Flussdiagrammen, die bei der Gestaltung von Softwareprodukten nützlich sind - Aufgabenflüsse, Benutzerflüsse und Flussdiagramme. Die Grenzen zwischen den einzelnen Diagrammtypen sind fließend, und der Begriff „Nutzerfluss“ wird häufig synonym mit „Aufgabenfluss“ und „Flussdiagramm“ verwendet. Im Allgemeinen eignet sich diese Diagrammklasse hervorragend zur Darstellung von gerichteten Abläufen und/oder entscheidungsbasierter Logik. Aufgabenabläufe sind nützlich, um zu entwerfen, wie ein Benutzer eine Aufgabe erledigen wird. Sie werden in der Regel als lineare Abfolge von Schritten dargestellt und können auf einer hohen Ebene entworfen werden oder sehr detailliert sein, indem eine Aufgabe in Teilaufgaben unterteilt wird. Sie können das Ergebnis einer Aufgabenanalyse sein, bei der beobachtet wird, wie Benutzer Aufgaben erledigen.

Aufgabenflüsse sind auch bei der Planung der optimalen Pfade für die Aufgabenerledigung nützlich, zumal sie sich leicht in natürlicher Sprache ausdrücken lassen. Man könnte sie auch als visuelle Antworten auf User Stories betrachten, da sie auf ähnliche Weise geschrieben sind.

Die User Story «Der Benutzer möchte eine schnelle Möglichkeit, einzelne Kleidungsstücke zu kaufen» könnte durch einen Aufgabenablauf wie den folgenden dargestellt werden:

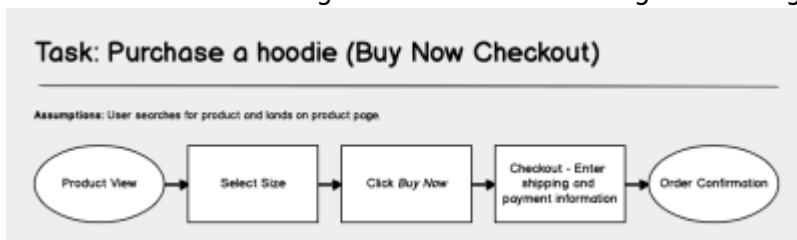


Abb. 6-7: Aufgabenfluss für Kaufprozess

Aufgabenflüsse eignen sich zwar für die Vorhersage, wie der Benutzer eine Aufgabe erledigt, aber

Abweichungen von diesen idealen Pfaden werden in der Regel nicht untersucht.

Flussdiagramme sind gut geeignet, um den Weg des Benutzers bei der Interaktion mit dem System (den Benutzer-Taskflow), die Entscheidungen, die er auf dieser Reise trifft, und die reaktiven Backend-Prozesse, die er möglicherweise auslöst, darzustellen.

Die Diagramme enthalten Start- und Endpunkte, Entscheidungspunkte, den gerichteten Fluss des Systems und die Reaktion des Systems auf die Eingaben des Benutzers, einschliesslich der Art und Weise, wie Daten verarbeitet und transportiert werden.

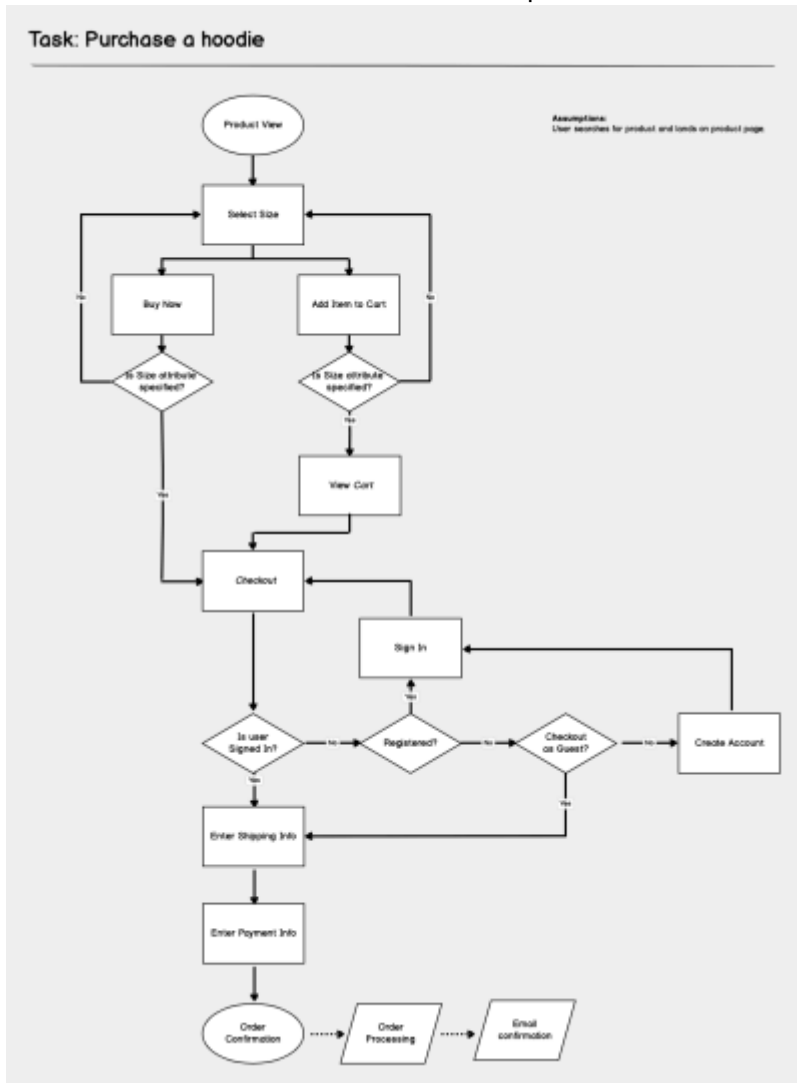


Abb. 6-8: Flussdiagramm zu Kaufprozess

Sie sind zwar gut geeignet, die Komplexität innerhalb des Produkts darzustellen, zeigen aber nicht das Design der Schnittstelle an verschiedenen Stellen. An dieser Stelle helfen Wireflows, die Lücke zwischen Wireframes und Flussdiagrammen zu schliessen.

Wireflows

Wireflows vereinen die Vorteile von Wireframes und Flussdiagrammen. Der Begriff Wireflow wurde von der Nielsen Norman Group geprägt, nachdem sie beobachtet hatte, wie sich die Praxis in diesem Bereich entwickelte.

Wireflows zeigen visuell, wie sich Teile der Benutzeroberfläche verändern, wenn der Benutzer mit der Anwendung interagiert. Zusätzlich können Anmerkungen hinzugefügt werden, um Faktoren wie die Übertragung von Daten innerhalb des Systems anzuzeigen.

Wireflows können einfach als eine Erweiterung der Wireframing-Praxis betrachtet werden, um Benutzerflussinformationen direkt in das Schnittstellendesign einzufügen. Das einfachste Beispiel zeigt lediglich eine Abfolge von Ereignissen bei der Ausführung einer Aufgabe.

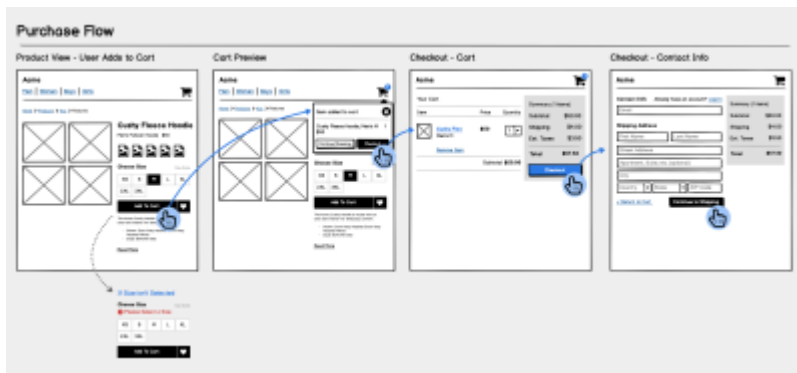


Abb. 6-9: Veranschaulichung des Ablaufs anhand eines Wireflows

Es können auch komplexere Beispiele für Wireflows erstellt werden, um zu zeigen, wie eine Vielzahl von Aktionen innerhalb einer Schnittstelle Änderungen oder Systemreaktionen in einer dynamischen Schnittstelle auslösen können.

Die komplexesten Beispiele können Flussdiagrammen ähneln und zeigen die Art von Entscheidungsverzweigungen, Backend-Prozessen und Datenbewegungen, wie sie in detaillierten Flussdiagrammen erwartet werden.

Wie man einen Wireflow erstellt

Produktdesigner verwenden schon seit langem hybride Methoden für das Wireframing. Es ist üblich, ganze Schlüsselbildschirme (Key Screens) zu reproduzieren und Zustandsänderungen zu zeigen. Häufig wird dabei eine Einzelansicht (Key Screen) um Teileansichten ergänzt, um das dynamische Verhalten zu zeigen.

Die Erstellung von Wireflows kann in 3 einfache Schritte, durchgeführt werden:

1. Beschreibender Text des Ablaufs oder User Flow
2. Definieren von Key Screens
3. Screens verbinden

1. Beschreibender Text

Oftmals wird mit einem Szenario oder einer Problemstellung begonnen. Dann kann eine Lösung in Worte gefasst und ein einfacher Aufgabenablauf festgehalten werden.

2. Definieren von Key Screens

Als Nächstes werden die wichtigsten Bildschirme (Key Screens) der Schnittstelle erstellt. Eine gute Möglichkeit, dies zu tun, ist es, in Form von Start- und Endpunkten zu denken. Im Beispiel der Aufgabe „Kauf eines Hoodies“ ist der Startpunkt die Produktseite und der Endpunkt die Bestätigungsseite der WEB-Anwendung.

Ziel ist es, Bildschirme zu identifizieren, bei denen die Schnittstelle ihren Zustand ändert, z. B. wenn Daten während eines Prozesses wie dem Auschecken aus einem Geschäft dynamisch aktualisiert werden. Es könnte ein Formular verwendet werden, das nach und nach mehr Eingaben erfordert, z.B. Kontaktinformationen, Versandinformationen und Rechnungsinformationen in einer Ansicht. Jedes dieser Formulare müsste entworfen werden.

3. Bildschirme verbinden

Als Nächstes wird die Verbindung der wichtigsten Bildschirme (Key Screens) skizziert. Dazu werden Pfeile zwischen die Kästchen gezeichnet, um den Raum zwischen den Key Screens zu definieren und den Benutzer durch seine Aufgabe voranzubringen. Es können Entscheidungspunkte eingefügt werden, um zu zeigen, was in verschiedenen Fällen passiert, wenn der Benutzer eine Option wählt, und was passiert, wenn er eine andere wählt.

Beispiel 14: App für Katzenfotos

Nehmen wir an, wir entwerfen eine App zur gemeinsamen Nutzung von Katzenfotos und möchten die Anmeldung und das erste Benutzererlebnis gestalten. Dazu kann folgender Flow skizziert werden:

1. Aufgabe die durchzuführen sind
 - Anmeldung: Der Benutzer meldet sich an und legt einen Benutzernamen, eine E-Mail und ein Passwort fest.
 - Erster Beitrag: Die erste Aufgabe des Nutzers ist es, sein erstes Foto direkt nach der Anmeldebestätigung zu machen.
2. Key Screens
 - Unregistrierte Startseite
 - Anmeldungsbildschirme
 - Anmeldebestätigung
 - Anmeldungsbildschirme
 - Erster Beitrag
 - Feed (Bestätigung)
3. Wireflow

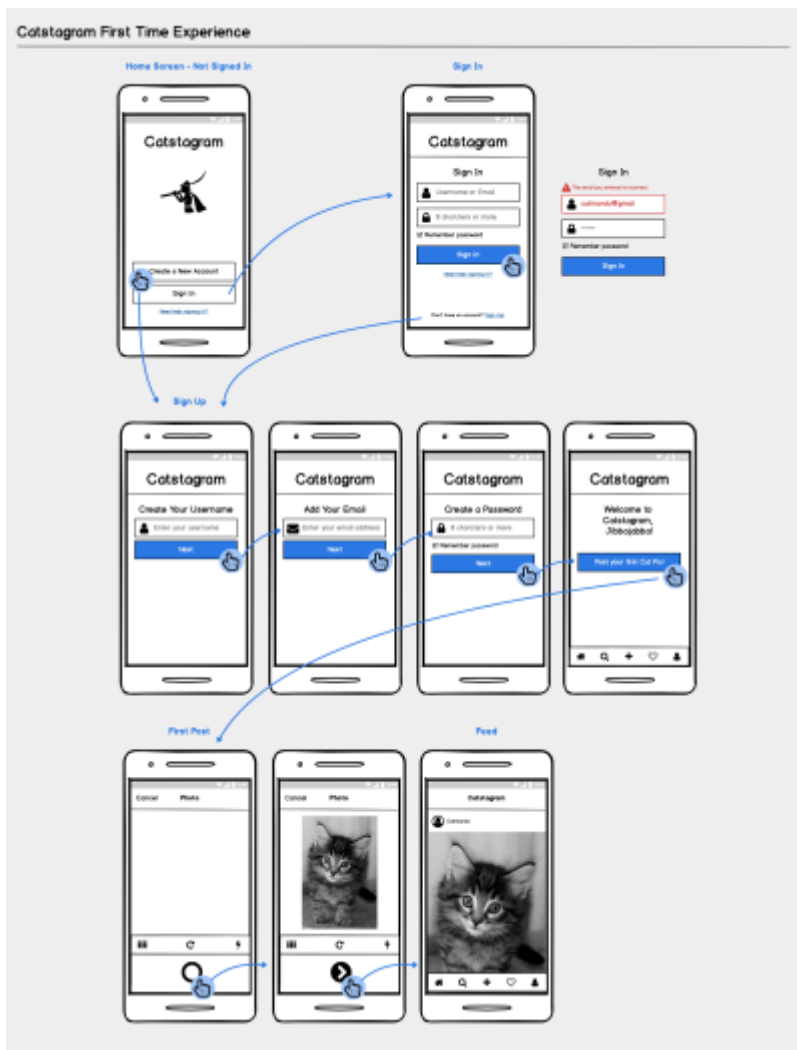


Abb. 6-10: Illustration mit Wireframes zur Veranschaulichung des Ablaufs

Schlussfolgerung

User Flows helfen dabei, optimale Pfade für die Erledigung von Benutzeraufgaben zu definieren.⁽¹⁾ und es können auch potenzielle Abweichungen von Best-Case-Szenarien vorhergesagt werden. Weiter kann aufgezeigt werden, wie der Benutzer wieder auf den richtigen Weg kommt (oder einen anderen beabsichtigten Pfad einschlägt).

Eine umfassende Untersuchung des Ablaufs stellt sicher, dass die Anwendung bereit ist, mit gut formulierten Meldungen zu helfen⁽²⁾, im Gegensatz zu vagen Fehlertexten. Außerdem können nützliche Optionen angeboten werden, um weiterzukommen⁽³⁾, wenn ein Hindernis vorliegt, anstatt dass der Benutzer sich festgefahren fühlt und herausfinden muss, was er als Nächstes tun soll. Wie wir im obigen Vergleich gesehen haben, bieten uns Wireflows eine Reihe von Vorteilen gegenüber der Verwendung jeder einzelnen dieser Entwurfstechniken. Der Hauptgrund für die Verwendung von Wireflows besteht darin, dass Sie als Produktdesigner die Möglichkeit haben, die Schnittstelle und den Ablauf auf die effektivste Art und Weise mit einem einzigen Design-Artefakt visuell zu kommunizieren.

- (1) Vergleiche ISO 9241: Aufgabenangemessenheit
(2) Vergleiche ISO 9241: Fehlerrobustheit (-toleranz)
(3) Vergleiche ISO 9241: Steuerbarkeit

Werkzeuge

Das Erstellen von Key Screens und Wireflows kann am einfachsten mit Papier und Bleistift realisiert werden. Um einen ersten Eindruck eines Konzepts zu erhalten, genügt dies vollauf.

Lernende/r	Datum	Lektion/Lernperson	Status	Bemerkung	Grund	Volumen
Name: [Name] Vorname: [Vorname] Klassen: [Klassen] Offene Absenden: [Offene Absenden] [Abmelden]	01.02.9999	Gesellschaft Mulp	[X]	[]	[]	01.02.99 Mulp 01.02.99 Verle
		Deutsche Mulp	[X]	[]	[]	01.02.99 Mulp 01.02.99 Verle
		Informatische Prakt	[X]	[]	[]	
		Englisch Bufe	[X]	[]	[]	
Zeitraum Semester: [Semester] von: [von] bis: [bis] Überprüfen [Mailversand] [Kurs] [Offene Absenden]	01.02.9999	Mathematik	[X]	[]	[]	
		Gesellschaft Mulp	[X]	[]	[]	
		Deutsche Mulp	[X]	[]	[]	

Status:
 ✓ anwesend ✓ entbunden ✓ dispensiert - Schulp
 x abwesend e. erkrankt x nicht dispensiert K. Schulp krank

Abb. 6-11: Handskizze eines Key Screen

Wer sich mit skizzieren schwer tut, kann auch Anwendungen wie Visio, Balsamiq usw. nutzen.

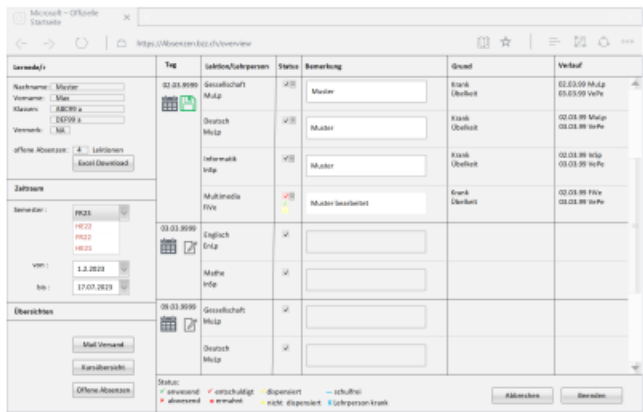


Abb. 6-12: Key Screen mit Visio erstellt

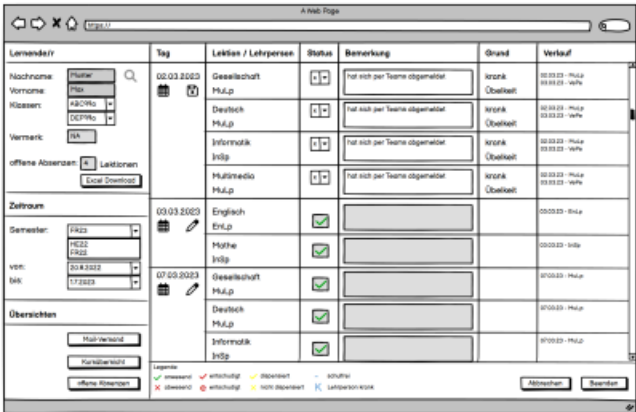


Abb. 6-13: Key Screen mit Balsamiq erstellt

Selbst mit einer Anwendung wie z.B. Excel lassen sich Key Screens entwerfen. Man muss dabei gut Überlegen, wie die Zeilenhöhen und Spaltenbreiten zu wählen sind. Elemente wie Pulldown, Scrollbar oder gewisse Icons können per copy-paste aus z.B. Visio eingefügt werden.

Lernender	Tag	Lektion/Lehrperson	Status	Bemerkung	Grund	Verlauf
Max Muster	02.03.9999	Gesellschaft MuLp		per Teams abgemeldet	krank Überkeit	02.03.99 - MuLp
Klassen: ABC22e, XYZ22e		Deutsch MuLp			krank Überkeit	02.03.99 - MuLp
offene Absenzen: 5 Excel Export		Informatik InSp		Arztzeugnis nachgereicht (Prüfung)	krank Überkeit	02.03.99 - MuLp 05.03.99 - InSp
Zeitraum	03.03.9999	Englisch LpEn				03.03.99 - LpEn
PR23 30.01.2023 17.07.2023		Mathematik InSp				02.03.99 - InSp
Übersichten	09.03.9999	Gesellschaft MuLp				09.03.99 - MuLp
Mit Versand Kurse offene Absenzen		Deutsch MuLp				09.03.99 - MuLp
Status: anwesend entschuldig dispensiert schulfrei abwesend erkrankt nicht dispensiert Lehrpers. Krank						
						Abbrechen Beenden

Abb. 6-14: Key Screen mit Excel erstellt

Eine günstige Alternative für Smartphone Screens bietet auch das Miro-Board an.



Abb. 6-15: Library des Miro-Boards mit Wireframes für Smartphones

Wireflow

Ein Wireflow zeigt den Ablauf, den eine Anwendung bei unterschiedlichen Aktionen nehmen kann. Um einen Wireflow zu skizzieren, ist es sinnvoll, wenn die jeweiligen Key Screens in einer minmierten Form präsentiert werden. Alle Elemente die nicht oder nur wenig von Bedeutung für den Ablauf der Anwendung sind, können als Linie dargestellt werden. Dadurch ergibt sich ein kompakteres Bild.

Beispiel 15: Wireflow zu Absenzttool

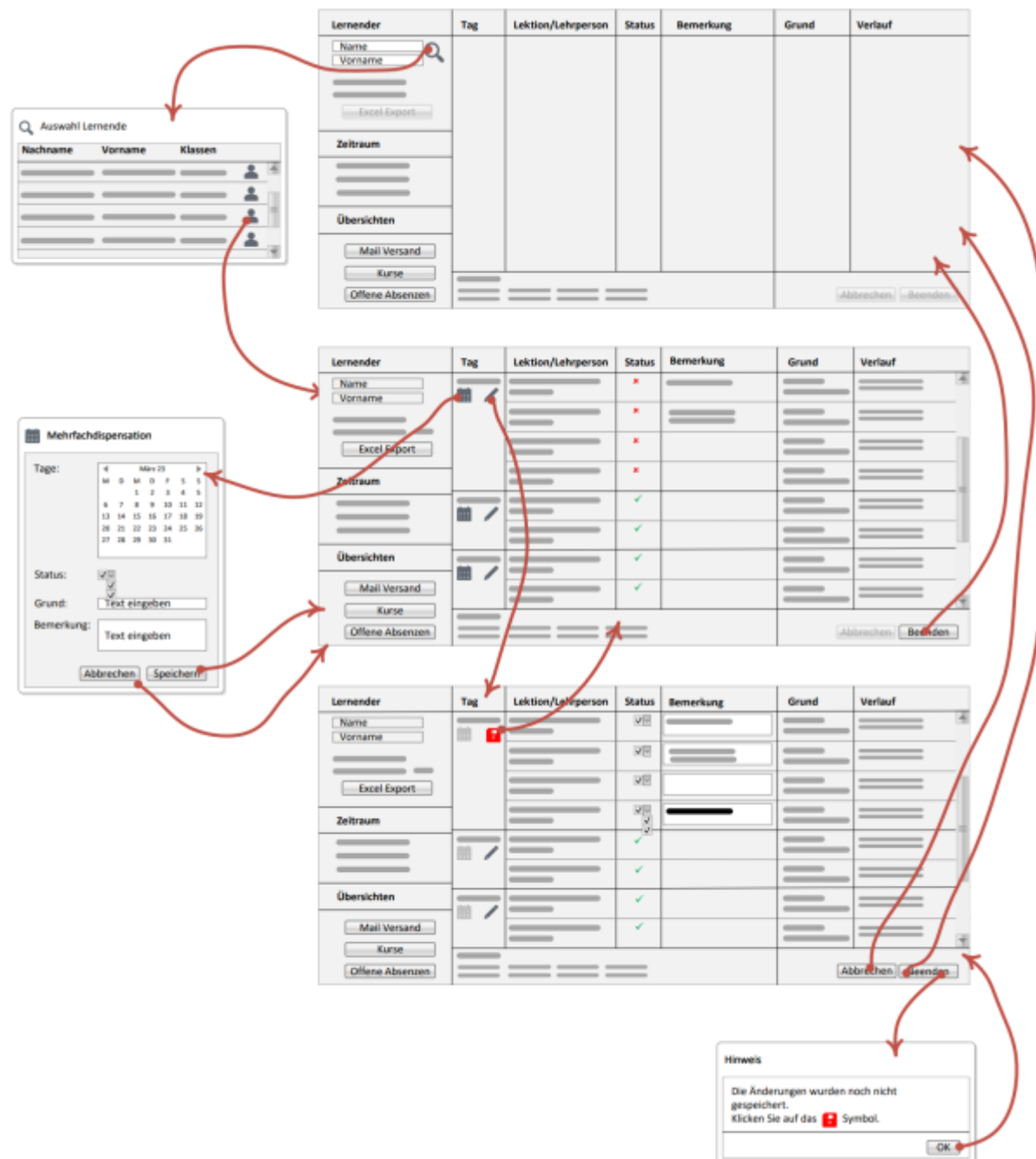


Abb. 6-16: Wireflow einer Anwendung

Hier findet sich auch ein Beispiel für die Lernförderlichkeit einer Anwendung, in dem der Benutzer über die Bedeutung des Disketten-Symbols aufgeklärt wird.

© René Probst

From:
<https://wiki.bzz.ch/> - **BZZ - Modulwiki**

Permanent link:
https://wiki.bzz.ch/modul/m322/learningunits/lu06/theorie/lu6-kapitel_4

Last update: **2025/03/14 17:26**



