

Informationsdesign

Eine Desktopapplikation auf einem mobilen Endgerät
darstellen

Bachelorarbeit

zur Erlangung des akademischen Grades

Bachelor of Science in Engineering (BSc)

eingereicht am Fachhochschul-Studiengang

Mobile Software Development

FH JOANNEUM (University of Applied Sciences), Kapfenberg

Betreuer:in: Elmar Krainz

(Firmenbetreuer:in: Roman Krbez; APUS Software GmbH)

Eingereicht von: Kathrin Hubmann

September 2024

Eidesstattliche Erklärung

Ich versichere hiermit an Eides statt, dass ich die vorliegende Bachelorarbeit/Masterarbeit selbständig und ohne fremde Hilfe angefertigt und die mit ihr verbundenen Tätigkeiten selbst erbracht habe und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe. Die hochgeladene Version und die allenfalls abgelieferte gedruckte Version sind identisch.

Ich erkläre zudem, dass ich die Arbeit im Sinne der Prinzipien der Richtlinie der FH JOANNEUM zur Sicherung Guter Wissenschaftlicher Praxis und zur Vermeidung von Fehlverhalten in der Wissenschaft erstellt habe. Insbesondere erkläre ich, dass ich Inhalte, die ich aus Werken Dritter oder auch aus eigenen Werken wörtlich oder inhaltlich übernommen habe, geeignet - und den Regeln für gutes wissenschaftliches Arbeiten entsprechend - gekennzeichnet und die Informationsquellen durch detaillierte Quellenangaben deutlich ersichtlich gemacht habe.

Die vorliegende Originalarbeit ist in dieser Form zur Erreichung eines akademischen Grades noch keiner anderen Hochschule im In- oder Ausland vorgelegt worden.

Ich bin mir bewusst, dass eine unwahre eidesstattliche Erklärung rechtliche Folgen haben kann.

(Ort, Datum)

1 Inhaltsverzeichnis

2	Abstract	1
3	Zusammenfassung	2
4	Einleitung	3
5	Ausgangssituation	4
5.1	Ionio	4
5.2	Problemstellung	4
5.3	Wunschplan als Desktopapplikation	5
6	Theoretischer Teil	7
6.1	Kalendarische Prototypen	7
6.1.1	Gemeinsame Basis	7
6.1.2	Erster Entwurf	8
6.1.3	Zweiter Entwurf	9
6.1.4	Komponenten	9
6.1.4.1	Dienst hinzufügen - Dialog	10
6.1.4.2	Datumswähler	10
6.2	Usability Tests	11
6.2.1	Was ist Usability	11
6.2.1.1	Lernbarkeit	13
6.2.1.2	Effizienz	13
6.2.1.3	Merkbarkeit	14
6.2.1.4	Fehleranfälligkeit	14
6.2.1.5	Zufriedenheit	15
6.2.2	Phasen eines Usability Tests	16
6.2.3	Quantitativ vs. Qualitativ	17
6.2.4	Testverfahren 1 – Thinking Aloud	18
6.2.5	Testverfahren 2 – Fragebogen	19

7	Praktischer Teil.....	20
7.1	Umsetzung der Prototypen.....	20
7.1.1	Verwendete Technologien.....	20
7.1.2	Gemeinsame Basis	20
7.1.3	Erster Prototyp	21
7.1.4	Zweiter Prototyp	22
7.1.5	Komponenten.....	23
7.1.5.1	Dienst hinzufügen - Dialog	23
7.1.5.2	Datumswähler	24
7.2	Usability Test.....	24
7.2.1	Fragebogen.....	25
7.2.2	Durchführung	25
7.2.3	Auswertung	26
7.2.3.1	Aufgaben 1 bis 3	26
7.2.3.2	Aufgabe 4.....	28
7.2.3.3	Fragebogen.....	29
8	Resultate	33
8.1	Technische Umsetzung	33
8.2	Usability Test.....	33
9	Fazit.....	36
10	Literaturverzeichnis.....	37
11	Anhang – GitHub Repository	38

2 Abstract

This thesis explores how an existing desktop application can be adapted for mobile devices. This is shown using the example of a specific part of the Ionio software, the so called "Wunschplan" (a company's duty roster which considers the employees' wishes). Two prototype versions are developed and afterwards evaluated through usability testing.

The theoretical section establishes the foundation for the prototypes based on relevant literature and design guidelines. It also contains the theoretical basis for conducting usability tests.

The practical section applies this theoretical basis to the creation of the prototypes and outlines the development and execution of the usability tests.

The results chapter analyzes the outcomes of the usability test and presents insights for future designs. The findings indicate that while the prototypes provide a solid foundation, further refinement is needed to achieve a satisfactory and user-friendly design.

3 Zusammenfassung

Diese Arbeit beschäftigt sich damit, wie eine vorhandene Desktopapplikation auf einem mobilen Endgerät dargestellt werden kann. Das wird anhand eines Teilbereichs der Software Ionio, dem sogenannten Wunschplan, gezeigt. Dafür werden zwei Versionen eines Prototyps erstellt und anschließend mithilfe eines Usability Tests überprüft.

Der theoretische Teil bildet einerseits die Grundlage für zwei Prototypen anhand von Fachliteratur und Design Guidelines, andererseits stellt er die theoretische Basis für den folgenden Usability Test dar.

Im praktischen Teil werden die Prototypen aufbauend auf relevanter Literatur und theoretischen Grundlagen erstellt. Auch die Entwicklung und die Durchführung der Usability Tests sind hier beschrieben.

Im Kapitel Resultate werden die Ergebnisse des Usability Tests analysiert und Schlüsse für zukünftige Designs gezogen. Hier zeigt sich, dass die Prototypen eine gute Basis für weitere Entwicklungen sind. Allerdings wird noch weitere Arbeit nötig sein, bevor sie zufriedenstellend und gut benutzbar sind.

4 Einleitung

In der heutigen Zeit werden mobile Anwendungen immer wichtiger, weil viele Erledigungen des täglichen Lebens mittlerweile hauptsächlich über Smartphones durchgeführt werden. Daher ist es auch üblich, mobile Versionen von Desktopapplikationen anzubieten. In dieser Arbeit wird der Entwicklungsprozess dargestellt, der von einer bestehenden Desktopapplikation zu einer mobilen Version führt.

Konkret werden im Zuge dieser Arbeit zwei Prototypen einer mobilen Applikation, basierend auf einem Teil der Desktopapplikation Ionio, erstellt. Diese werden anschließend einem Usability Test unterzogen. Die vorliegende Arbeit beschreibt diesen Prozess Schritt für Schritt:

Zunächst wird im Kapitel Ausgangssituation beschrieben, was Ionio ist und wieso diese Applikation als Beispiel in dieser Arbeit herangezogen wird. Dann wird noch erklärt, welche für die geplanten Prototypen relevante Funktionen die ursprüngliche Applikation besitzt.

Im darauf folgenden theoretischen Teil der Arbeit werden mögliche Darstellungen der App anhand von Vorgaben in der theoretischen Literatur sowie Grundlagen und Guidelines erörtert. Daraufhin wird die Grundlage für Usability Tests geschaffen, indem erklärt wird, was Usability eigentlich ist und welche Formen von Usability Tests es gibt. Anhand dieses theoretischen Überblicks wird anschließend ausgewählt, welcher Test am besten geeignet ist, in weiterer Folge tatsächlich durchgeführt zu werden.

Im praktischen Teil der Arbeit werden schließlich anhand der theoretischen Grundlagen zwei Prototypen erstellt. Daraufhin werden mit einigen Testpersonen Usability Tests der Prototypen durchgeführt. Eine Darstellung der Resultate der Tests und ihrer Implikationen bindet den Abschluss der Arbeit.

5 Ausgangssituation

5.1 Ionio

Ionio ist eine Dienstplanungs- und Ressourcenmanagement Software, die von Apus entwickelt wurde und hauptsächlich im Gesundheitssektor, aber mittlerweile auch im Blaulichtsektor und der Industrie verwendet wird. Ionio umfasst ein Dienstplanungssystem (Sollplan, Istplan, Wunschplan und Absenzenplan), Arbeitszeiterfassung sowie Projektzeiterfassung und Ressourcenmanagement für Personen, Material und Güter.

Der Sollplan ist der Plan, der nach der Fertigstellung existiert, hier sind alle Wünsche berücksichtigt worden. Der Istplan ist der Plan, der durch ungeplante Krankenstände, Verschiebungen oder getauschten Diensten entsteht. Am ende der Istplan-Phase wird dieser gesperrt, damit keine Änderungen vorgenommen werden können und für die Personalverrechnung herangezogen. Der Absenzenplan zeigt alle geplanten Absenzen, wie Langzeitkrankenstände oder Urlaube, an.

Die vorliegende Arbeit konzentriert sich auf den Wunschplan. Bei dem Wunschplan handelt es sich um den Teil von Ionio, in dem die Mitarbeitende eintragen, welche Dienste und Absenzen, wie beispielsweise Urlaub, sie gerne an welchen Tagen hätten. Diese Wünsche können dann von der Person, die die Dienstpläne erstellt, bestätigt oder abgelehnt werden. Dieser Plan dient dann als Grundlage für den Sollplan, die fertige Version des Dienstplans.

5.2 Problemstellung

Ionio existiert derzeit ausschließlich als Desktopanwendung im Browser, soll aber in naher Zukunft auch als mobile Anwendung zur Verfügung stehen. Eine mobile Anwendung wird benötigt, da es vermehrt als Wunsch der Kunden rückgemeldet wurde, dass diese gerne über ihre Smartphones schnell und einfach ihre Dienste eintragen möchten, auch wenn sie sich gerade nicht in der Nähe eines Computers befinden, wie zum Beispiel Angestellte im Außendienst. Da Ionio aber für den Desktop designt wurde, ist es notwendig, herauszufinden,

wie die einzelnen Komponenten auf einem mobilen Gerät dargestellt werden können. Um dieses Ziel zu erreichen, wird in dieser Arbeit der Wunschplan prototypisch als mobile Applikation dargestellt. Die Prototypen fokussieren dabei darauf, Benutzerfreundlichkeit und Funktion miteinander zu vereinbaren.

5.3 Wunschplan als Desktopapplikation

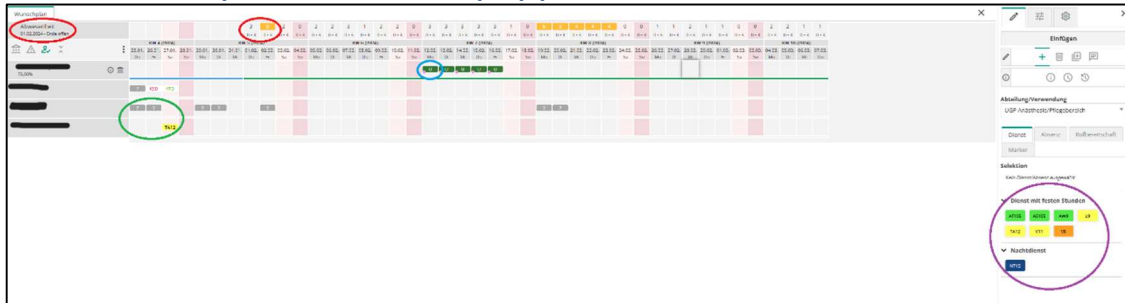


Abbildung 1: Ionio Desktopapplikation
(eigene Darstellung)

Abbildung 1 zeigt einen Screenshot der Desktopapplikation des Wunschplans in Ionio. Da dieser Screenshot in der realen Applikation entstanden ist, sind die Namen, der derzeit angezeigten Personen aus Datenschutzgründen geschwärzt.

Die Applikation hat als Hauptelement eine Kalenderansicht. In diesem Kalender sind die Dienste eingetragen. Wenn man die Applikation öffnet, sieht man in der obersten Zeile immer die eigenen Dienste. Da die oberste Zeile immer die eigene ist, wird den Benutzern die Planungsperiode direkt darunter angezeigt. Im Beispiel in Abbildung 1 kann man sehen, dass die ersten Tage noch in der Planungsphase, dem Sollplan, sind. Diese sind daher blau dargestellt. Die restlichen Tage befinden sich in der grünen Planungsphase, dem Wunschplan.

Unter der eigenen Zeile kann man sich als Nutzer der Anwendung auch die Dienste von anderen Personen der Abteilung anzeigen lassen, allerdings sind nicht alle Einträge aus Gründen des Datenschutzes ersichtlich, dies betrifft hauptsächlich Absenzen. Da Absenzen auch Krankenstände umfassen und diese als Gesundheitsdaten zu den besonders schützenswerten personenbezogenen Daten zählen, dürfen andere Mitarbeiter nicht erkennen können um welche Art von Absenz es sich handelt [vgl. (DI Dr. Lang, et al., 2019)].

Im grünen Kreis in der Abbildung 1 sind eben solche eingetragene Dienste zu sehen. Dienste, die man als Benutzer nicht sehen darf, sind als graue Blöcke mit Fragezeichen als Beschriftung dargestellt. Meist sind das Absenzen, die andere Mitarbeitende aus Datenschutzgründen nicht sehen dürfen (siehe oben).

Im blauen Kreis in der Abbildung wurde vom Benutzer oder der Benutzerin ein Urlaub eingetragen (zu erkennen am U als Beschriftung und der von Ionio festgelegten grünen Farbe für Urlaube), und an der linken unteren Ecke des Urlaub-Blocks befindet sich ein kleiner Punkt in lila. Dieser Punkt zeigt an, dass der Wunschtermin noch nicht bestätigt worden ist.

Rechts unten im Bild, eingekreist in lila, sieht man die Dienste, die ausgewählt werden können. Die Farben und Abkürzungen auf den Blöcken werden von der Organisation verwaltet und sehen daher von Kunde zu Kunde unterschiedlich aus.

Oben links, im roten Kreis, sieht man den ausgewählten Filter für die Besetzungsvorgabe. Die Besetzungsvorgabe definiert wie viele Dienste einer Art an einem Tag mindestens und höchstens erlaubt sind. Diese Angabe sind oben in der Mitte, im zweiten roten Kreis, ersichtlich. In der Zeile steht oben die Anzahl der eingetragenen Dienste und unterhalb die Spanne der erlaubten Anzahl. Sollte die tatsächliche Anzahl über oder unter der erlaubten Grenze liegen, wird die obere Zahl orange hinterlegt. In diesem Beispiel werden Absenzen angezeigt. An einem Tag dürfen nicht mehr als 3 Absenzen eingetragen werden (0 – 3). Im roten Kreis sieht man, dass dies, allerdings an diesem Tag überschritten wurde, daher ist die darüber angezeigte fünf orange hinterlegt.

Nach der Darstellung der Ausgangssituation dieser Arbeit, werden im nächsten Teil, die theoretischen Grundlagen für einen Prototypen und einen anschließenden Usability Test behandelt.

6 Theoretischer Teil

Der theoretische Teil dieser Arbeit beschäftigt sich mit einer möglichen Darstellung des Wunschplans in Form einer kalendarischen Monatsansicht sowie mit Methoden für Usability Tests. Die im ersten Abschnitt besprochenen theoretischen Erkenntnisse fließen anschließend in die Umsetzung der beiden Prototypen ein. Die Erkenntnisse des zweiten Abschnitts bilden die Grundlage für die Durchführung der Usability-Tests.

6.1 Kalendarische Prototypen

Um den Wunschplan auf einem mobilen Gerät darzustellen, ist es notwendig, sich zuerst mit Kalenderdarstellungen, im speziellen mit den Monatsansichten, zu beschäftigen. Zwei der prototypischen Entwürfe werden sich an schon bestehenden Kalenderapps orientieren, einmal an Apple und einmal an Google. Diese zwei Darstellungsmöglichkeiten sind vielen Endbenutzern schon bekannt und erleichtern somit die Bedienung der beiden Prototypen.

Da die Einbindung iOS- Betriebssystems den möglichen Umfang dieser Arbeit überschreitet, werden sich die Prototypen auf das Android Betriebssystem konzentrieren.

6.1.1 Gemeinsame Basis

Die Basis für die App sind ein Header und eine Navigationsleiste. Der Header besteht aus der Statusbar und der sogenannten Top app bar.

In der Top app bar befindet sich der Titel, links davon ein Navigationsicon und am rechten Bildschirmrand interaktive Icons [vgl. (Google, Material Design, p. Top app bars)]. Im Prototyp wird das Navigationsitem ein Burgermenü sein und die interaktiven Icons werden zu den Filtern und den Einstellungen führen.

Die Statusbar ist ein integraler Bestandteil jeder App und zeigt wichtige Hardwarefunktionen, wie zum Beispiel den Akkustand und die Netzwerkverbindung, an. Die Statusbar kann modifiziert werden, wobei man aber nur die Anzeige und nicht die Funktion ändern sollte, zum Beispiel in dem man den Hintergrund farblich an das Design anpasst [vgl. (Hoover/Berkman, 2011, S. 17 - 19)].

6.1.2 Erster Entwurf

Der erste Entwurf wird sich an der Google Kalenderapp orientieren, deren Monatsansicht tabellarisch aufgebaut ist. Am oberen Bildschirmrand wird der Monat angezeigt und direkt darunter sieht man die Initialen der Wochentage. Darunter folgen die einzelnen Daten als Tabelle in hochkant rechteckigen Feldern. Darin befinden sich, sofern ein Termin eingetragen wurde, kleinere farbige Rechtecke, die den Termin darstellen. In diesen Terminboxen steht auch der Anfang der Terminbezeichnung. [siehe Abbildung 2]

Für den ersten Entwurf eines Prototyps für die mobile Version von Ionio, werden große Teile dieses Designs übernommen, jedoch wird der Header, der derzeit das Monat und ein Menü enthält, anders aufgebaut werden, da an die Applikation andere Ansprüche gestellt werden als an einen einfachen Kalender. Im Header werden sich daher der Name der Applikation („Wunschplan“) sowie ein Zugang zum Menü, zu den Einstellungen der App und zu den Filtern des Wunschplans befinden. Da aber der Monat auch erkenntlich sein muss, wird dieser darunter angezeigt werden,.

Eine weitere Änderung des ursprünglichen Google-Designs wird sein, dass der Anwender erkennen muss, in welcher Planungsphase sich die angezeigte Periode befindet. Dafür wird entweder eine farbige Linie unter der Datumszahl zu sehen sein, oder die Felder werden farblich hinterlegt werden. Welche dieser Optionen sich durchsetzt, wird sich im Verlauf der Erstellung zeigen.

Alle Termine auf einen Blick sehen

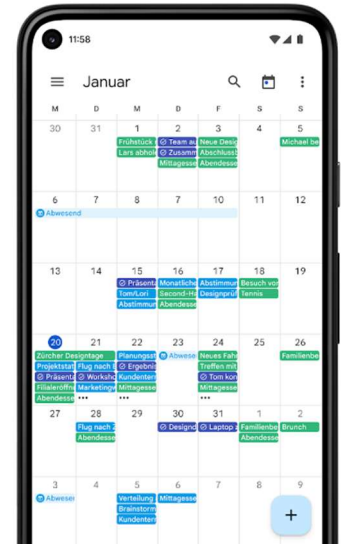


Abbildung 2: Google Kalender – Monatsansicht
(<https://play.google.com/store/apps/details?id=com.google.android.calendar&hl=>

6.1.3 Zweiter Entwurf

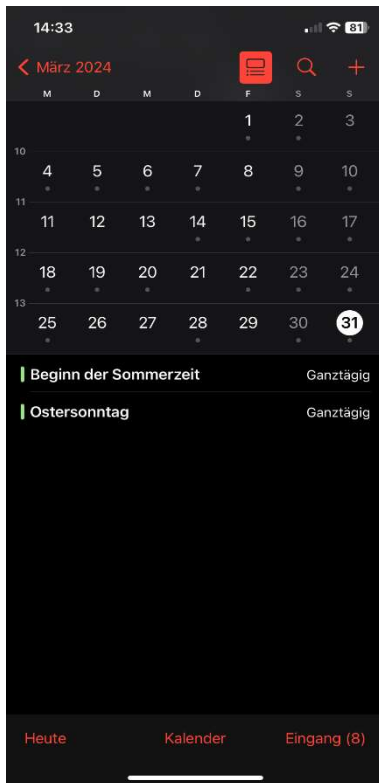


Abbildung 3: Apple Kalender – Monatsansicht (eigene Darstellung)

Der zweite Entwurf wird sich an der Kalenderapp von Apple orientieren. Der Apple Kalender hat im Gegensatz zum Google Kalender keine Monatsansicht, in der der User erkennen kann, welche Termine oder in diesem Fall Dienste an den jeweiligen Tagen eingetragen sind. Die Monatsansicht ist ein Kalender, der nur aus den Datumszahlen und dem Monat besteht. Wenn ein Termin eingetragen ist, erscheint unter der entsprechenden Zahl ein Punkt. Man kann aber mithilfe der Icons am oberen Rand einstellen, dass die Termine des ausgewählten Tages unter dem Kalender als scrollbare Liste erscheinen wie in Abbildung 3 dargestellt.

Diese Ansicht wird für den zweiten Entwurf als Vorlage verwendet, da auch in der Desktopapplikation erkennbar ist, welche Dienste bereits eingetragen sind. Die Punkte werden aber nicht in der bestehenden Form übernommen, sondern etwas abgewandelt, um Über- oder Unterschreitungen der Besetzungsvorgabe zu markieren, wenn dies der Fall ist erscheint ein oranger Punkt unter dem Datum.

Um eine bestimmte Planungsperiode darzustellen, werden entweder die Zahlen farblich hinterlegt oder eine farbige Linie wird unter den Zahlen erscheinen. Welche der Optionen sich durchsetzt, wird sich im Verlauf der Erstellung zeigen. Für den Fall, dass die Variante mit der Linie zum Einsatz kommt, wird der orange Punkt über dem Datum anstatt darunter angezeigt werden.

6.1.4 Komponenten

Die Prototypen orientieren sich sowohl an Designs von Apple als auch von Google. Hier gibt es Überschneidungen, da mehrere Komponenten in beiden Versionen gebraucht werden. Um die Übersichtlichkeit zu verbessern, werden

diese Komponenten einheitlich gestaltet. Für das Design wurden die Guidelines von Google Material Design herangezogen.

6.1.4.1 Dienst hinzufügen - Dialog

Um einen neuen Dienst hinzufügen zu können braucht es einen Dialog, in dem der Nutzer oder die Nutzerin den Dienst auswählen kann und festlegen kann an welchem Tag er hinzugefügt werden soll. Da es aber auch Absenzen, wie Urlaub, als Auswahlmöglichkeit gibt, muss es eine Möglichkeit geben eine Datumsspanne auszuwählen. Um das zu erreichen, wird es eine Checkbox geben, die beim Aktivieren ein zweites Datumsfeld erscheinen lässt.

6.1.4.2 Datumswähler

Eine Komponente, die in beiden Entwürfen zum Einsatz kommen wird, ist ein Datumswähler, der benötigt wird, um das Datum für einen neuen Dienst auszuwählen. Wie ein Datumswähler aufgebaut ist, beschreibt die Website Material Design, ein open-source Design System von Google [vgl. (Google, Material Design, p. Date pickers)]. Material Design beschreibt drei gängige Varianten von Datumswählern, die in unterschiedlichen Situationen verwendet werden. Es gibt den „Docked date picker“, der an das Eingabefeld gedockt ist, den „Modal date input“, bei dem man das Datum selbst eingeben muss, und den „Modal date picker“, der als Erweiterung für den „Modal date input“ dient, wenn man das Datum nicht selbst eingeben möchte. Der „Docked date picker“ ist am besten geeignet, um Daten in naher sowie ferner Zukunft und Vergangenheit auszuwählen, er ist jedoch an das Eingabefeld gedockt und öffnet sich, sobald man in dieses klickt, weshalb er sich gut für Desktopanwendungen eignet. Bei der zweiten Variante, dem „Modal date input“ gibt es ein Eingabefeld für das Datum. Diese Variante ist allerdings so klein, dass sie sich auch für mobile Anwendungen eignet. Dieser „Modal date input“ kann mit dem „Modal date picker“ erweitert werden, indem man ihn mit Hilfe eines Icons aufrufen kann. Diese Art des Datumswählers kann entweder bildschirmfüllend oder als kleiner Dialog geöffnet werden. Bei der bildschirmfüllenden Variante scrollt man horizontal zwischen den Monaten, im

Dialog vertikal. Das Jahr kann mithilfe eines Dropdown Menüs ausgewählt werden. Daher eignet sich der „Modal date picker“ gut für zeitnahe Ereignisse. Der Wunschplan wird den „Modal date picker“ benutzen, da er sich für mobile Geräte eignet und die Dienste im Wunschplan nie lange Zeit Voraus einzutragen sind. Den „Modal date input“ werde ich nicht so verwenden wie er von material design beschrieben wird, da ich in meinem Dialog für das Hinzufügen eines Dienstes mehr als nur eine Datumsangabe benötige. Übernommen wird allerdings, dass der Datumswähler wahlweise per Icon aufgerufen werden kann oder das Datum eingegeben werden kann.

6.2 Usability Tests

Um herauszufinden, ob die Prototypen benutzerfreundlich sind, wird im praktischen Teil der Arbeit ein Usability Test durchgeführt. In diesem Kapitel wird die theoretische Grundlage dafür geschaffen. Zuerst wird erklärt, was Usability ist und wofür es verwendet wird. Danach werden die Phasen eines Usability Tests und der Unterschied zwischen quantitativen und qualitativen Usability Tests erklärt, um festzustellen, welche Art sich für diese Arbeit am besten eignet. Zuletzt werden zwei verschiedene Testverfahren beschrieben, um zu ermitteln welches eingesetzt werden soll.

6.2.1 Was ist Usability

Usability ist aus dem ursprünglichen Konzept der Userfreundlichkeit entstanden. Da dieser Begriff allerdings nicht passend war, da Maschinen nicht freundlich, sondern dem Nutzer nicht im Weg sein sollen, etablierten sich in Fachkreisen für Userinterfaces andere Begriffe wie HCI (human-computer interaction), UCD (user-centered design) oder MMI (man-machine interface).

Nielsen definiert Usability als Teil der Systemakzeptanz. Systemakzeptanz beschreibt, ob ein System allen Anforderungen der User sowie aller anderen Interessensgruppen, zum Beispiel Manager, entspricht. Diese Akzeptanz gliedert sich in soziale und praktische Akzeptanz, wobei die Nutzbarkeit als Teil der praktischen Akzeptanz sich wiederum in Utility und Usability aufteilt. Utility beschreibt, ob die Anwendung die gewünschten Funktionen bereitstellt und den tatsächlichen Bedürfnissen entspricht, während Usability sich darauf bezieht, wie benutzerfreundlich und effektiv diese Funktionen genutzt werden können. [vgl. (Nielsen, 1993, pp. 23 - 25)]

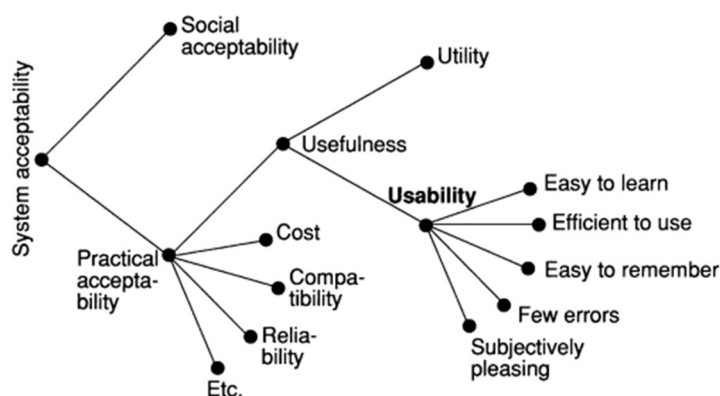


Abbildung 4: Attribute der Systemakzeptanz (Nielsen 1993, S. 25)

Usability hat, wie in Abbildung 4 ersichtlich, fünf Attribute (Lernbarkeit, Effizienz, Merkbarkeit, wenig Fehler und Zufriedenheit), welche Usability besser messbar machen sollen. Für die Messung von Usability gibt es zwei Möglichkeiten: entweder bekommen Testuser vorher spezifizierte Aufgaben, die sie im System erfüllen müssen, oder reale Nutzer werden beim Erfüllen alltäglicher Aufgaben beobachtet. Egal welche der beiden Varianten gewählt wird, die Aufgaben sollten relativ für jedes Attribut extra bewertet werden [vgl. (Nielsen, 1993, pp. 26 - 27)].

Im folgenden Abschnitt werden die Attribute von Usability im Detail besprochen.

6.2.1.1 Lernbarkeit

Die Lernbarkeit ist ein grundlegendes Merkmal in der Usability, da die meisten Systeme leicht erlernbar sein müssen. Besonders wichtig ist die erste Nutzungserfahrung, bei der die Nutzer das System kennenlernen. Systeme mit hoher Erlernbarkeit ermöglichen es, schnell ein angemessenes Kompetenzniveau zu erreichen.

Ausnahmen bilden sogenannte "Walk-up-and-use"-Systeme, wie zum Beispiel Infoterminals in Museen, die sofort nutzbar sein müssen. Zudem profitieren Nutzer von bereits bekannten Systemen bei Upgrades von einer verkürzten Lernkurve.

Lernbarkeit wird oft in Tests mit unerfahrenen Nutzern gemessen. Allerdings ist es besser, wenn sowohl Anfänger als auch erfahrene Nutzer berücksichtigt werden. Die meisten Nutzer erlernen nicht das gesamte System, bevor sie es nutzen, sondern nur die grundlegenden Funktionen. Deshalb sollte man nicht nur die Zeit messen, die es braucht, um das gesamte System zu verstehen sondern auch die Zeit, die nötig ist, um die notwendigen Grundlagen zu erlernen.

[vgl. (Nielsen, 1993, pp. 27 - 30)]

6.2.1.2 Effizienz

Die Effizienz beschreibt das Leistungsniveau von erfahrenen Nutzern, wenn die Lernkurve abgeflacht ist. Es kann Jahre dauern, bis Nutzer Expertenniveau erreichen, besonders bei komplexen Systemen. Viele Nutzer stagnieren jedoch, sobald sie ein „ausreichendes“ Wissen erworben haben, obwohl das Erlernen fortgeschrittener Funktionen ihnen langfristig Zeit sparen könnte.

Die Effizienz wird durch Tests mit erfahrenen Nutzern gemessen, wobei „Erfahrung“ entweder durch die Dauer der Nutzung (z. B. ein Jahr) oder durch die Anzahl der Nutzungsstunden definiert wird. Eine weitere Möglichkeit ist die Messung der Leistungskonstanz: Wenn sich die Leistung über einen gewissen Zeitraum nicht mehr verbessert, gilt der Nutzer als erfahren. Effizienz wird typischerweise anhand der Zeit gemessen, die Nutzer für typische Aufgaben

benötigen.

[vgl. (Nielsen, 1993, pp. 30 - 31)]

6.2.1.3 Merkbarkeit

Merkbarkeit ist für gelegentliche Nutzer entscheidend, die ein System in unregelmäßigen Abständen verwenden. Diese Nutzer müssen sich nicht komplett neu einarbeiten, sondern lediglich ihr Wissen auffrischen. Typische Beispiele sind Programme für seltene Aufgaben, wie die Erstellung eines Quartalsberichts oder andere Tools für besondere Zwecke. Auch Nutzer, die nach einer längeren Pause zu einem System zurückkehren, profitieren von einer leicht erinnerbaren Benutzeroberfläche.

Eine hohe Erinnerbarkeit wird oft durch gute Erlernbarkeit unterstützt. Es gibt zwei Hauptmethoden zur Messung, nämlich erstens Praxistests mit Nutzern, die das System nach einer Pause verwenden, wobei die Zeit gemessen wird, die sie für typische Aufgaben benötigen; und zweitens Gedächtnistests, bei denen Nutzer nach einer Nutzungssession erklären müssen, wie bestimmte Befehle funktionieren, oder diese benennen beziehungsweise zeichnen sollen.

Moderne Benutzeroberflächen setzen häufig darauf, Informationen sichtbar zu halten, sodass sich Nutzer nicht alles merken müssen. Solche Systeme erleichtern die Nutzung, auch wenn sich die Nutzer nicht aktiv an Befehle oder Menüs erinnern können.

[vgl. (Nielsen, 1993, pp. 31 - 32)]

6.2.1.4 Fehleranfälligkeit

In einem nutzerzentrierten Systemdesign ist es essenziell, die Fehleranfälligkeit so gering wie möglich zu halten. Fehler werden dabei als Aktionen definiert, die das gewünschte Ziel nicht erreichen. Die Fehlerrate eines Systems lässt sich durch die Anzahl der Fehler während der Ausführung spezifischer Aufgaben ermitteln.

Man muss allerdings zwischen zwei Fehlerarten unterscheiden. Einerseits gibt es Fehler, die vom Nutzer direkt behoben werden; diese verlangsamen den

Nutzer zwar, werden aber in diesem Punkt nicht berücksichtigt, dafür aber im Punkt Effizienz. Andererseits gibt es kritische Fehler, die so bezeichnet werden, weil sie entweder nicht entdeckt werden und somit zu verfälschten Arbeitsergebnissen führen oder weil sie die Arbeit des Nutzers zerstören und eine Wiederherstellung erschweren.

[vgl. (Nielsen, 1993, pp. 32 - 33)]

6.2.1.5 Zufriedenheit

Das subjektive Empfinden der Nutzer, wie angenehm oder zufriedenstellend ein System zu nutzen ist, ist eine wichtige Usability-Eigenschaft, insbesondere bei freiwillig genutzten Systemen wie Spiele oder kreative Software. In solchen Fällen steht der Unterhaltungswert oft im Vordergrund. Am häufigsten wird die Zufriedenheit der Nutzer durch kurze Fragebögen ermittelt, die nach der Nutzung ausgefüllt werden. Typischerweise nutzen diese Fragebögen Skalen wie die Likert-Skala (zum Beispiel Bewertung von „stimme überhaupt nicht zu“ bis „stimme voll zu“). Eine andere Methode ist die semantische Differenzierung, bei der Nutzer zwischen gegensätzlichen Begriffen wählen (z. B. „leicht zu lernen“ vs. „schwer zu lernen“).

Es gibt allerdings einige Faktoren, die bei einer Befragung berücksichtigt werden müssen. Die Bewertung der subjektiven Zufriedenheit wird oft von einzelnen, besonders schwierigen Nutzungserfahrungen beeinflusst, da diese den Nutzern stärker im Gedächtnis bleiben. Zudem kann eine unpersönliche Atmosphäre bei der Interaktion mit dem System zu schlechteren Bewertungen führen. Außerdem neigen Nutzer dazu, in Testsituationen höflicher zu sein und das System positiver zu bewerten. Um ehrlichere Ergebnisse zu erzielen, können daher Fragen mit entgegengesetzten Formulierungen eingesetzt werden, die sowohl positive als auch negative Aspekte abfragen.

Objektive Messungen wie Herzfrequenz oder EEG sind zwar möglich, aber oft unpraktisch, da sie sehr aufwendig sind. Am besten zeigt eine freiwillige Nutzung, wie zufrieden die Nutzer mit dem System sind.

[vgl. (Nielsen, 1993, pp. 33 - 37)]

6.2.2 Phasen eines Usability Tests

Usability Tests durchlaufen üblicherweise vier Phasen: Vorbereitung, Einleitung, der eigentliche Test und Nachbereitung.

Bei der Vorbereitung sollte der Versuchsleiter sicherstellen, dass die Umgebung, der Computer oder das Smartphone, sowie alle anderen benötigten Materialien bereit sind. Das System auf dem der Test durchgeführt wird sollte in seinen originalen Zustand zurückgesetzt werden, falls vorherige Tests daran etwas verändert haben. Das alles sollte erledigt werden bevor die Testperson zum Test eintrifft. Zusätzlich sollten auch alle störenden Benachrichtigungen vorsorglich abgeschaltet werden.

Sobald die Testperson eingetroffen ist, beginnt die Einleitung. Hier erklärt der Versuchsleiter oder die Versuchsleiterin kurz, weshalb der Test durchgeführt wird, erläutert das System, auf dem der Test durchgeführt wird, falls es der Testperson unbekannt ist, und beschreibt zuletzt die Testmethode. Falls es noch Fragen seitens der Testperson gibt, können sie auch in dieser Phase an den Versuchsleiter oder die Versuchsleiterin gestellt werden.

Wenn die Testperson bereit ist und alle Fragen beantwortet wurden, kann der Test starten. Während der Durchführung sollte der Versuchsleiter oder die Versuchsleiterin nicht mit der Testperson interagieren. Der Versuchsleiter oder die Versuchsleiterin darf auch nicht bei Schwierigkeiten weiterhelfen, da dies sonst den Test verfälschen würde. Eine Ausnahme aber ist, wenn die Testperson feststeckt und nicht mehr weiter kommt, oder wenn ein Fehler auftritt, der bei vorherigen Tests auch aufgetreten ist und zu bekannten Folgefehlern führt. Wenn ein Test auf Zeit ausgeführt wird, darf der Versuchsleiter oder die Versuchsleiterin aber unter keinen Umständen eingreifen.

Bei der Nachbereitung kann ein Zufriedenheitsfragebogen ausgefüllt werden. Wenn das der Fall ist, sollte dieser aber direkt nach dem Test und vor der Besprechung über das System geschehen, um eine Voreingenommenheit zu vermeiden. Nach dem Fragebogen kann der Versuchsleiter oder die Versuchsleiterin noch nach Feedback und Verbesserungsvorschlägen fragen. Selbst wenn diese nicht direkt umgesetzt werden, können sie hilfreich für

spätere Überarbeitungen sein. Der Versuchsleiter oder die Versuchsleiterin kann die Nachbereitung auch nutzen, um der Testperson Unklarheiten im Test zu erklären. Wenn dieses Gespräch abgeschlossen ist, sollte der Versuchsleiter oder die Versuchsleiterin sicherstellen, dass alle Aufzeichnungen und Notizen mit der Person gekennzeichnet sind.

[vgl. (Nielsen, 1993, pp. 187 - 191)]

6.2.3 Quantitativ vs. Qualitativ

Alle Usability Tests folgen dem Prinzip, dass Testpersonen Aufgaben im System erledigen. Die Daten die daraus gezogen werden können, sind allerdings unterschiedlich.

Quantitative Daten bestehen hauptsächlich aus Messwerten, zum Beispiel wie lange eine Aufgabe gedauert hat. Da es aber relativ schwierig ist diese Zahlen alleine zu interpretieren, beschreiben quantitative Tests selten die Usability des Systems, sondern vergleichen die gewonnenen Daten mit festgelegten Standards oder den Daten von Konkurrenten. Die Daten können zwar zeigen, in welchen Teilen des Systems User Schwierigkeiten haben, allerdings erfährt man nicht, was genau das Problem war oder wie man es lösen und die Usability verbessern könnte. Quantitative Daten haben aber auch ihre Vorteile: die Messwerte sind, im Gegensatz zur persönlichen Empfindung, objektiv, da sie durch viele Personen entstanden sind und somit auch repräsentativer für die ganze Zielgruppe sind.

Qualitative Daten hingegen entstehen meist durch das Beobachten von wenigen Testnutzern. Versuchsleiter erfahren so, welche Probleme Nutzer haben, und können anschließend auch nachfragen und um Feedback bitten. Anhand der eigenen Erfahrung und weiterer Testergebnisse wird das Feedback dann abgewogen und, wenn notwendig, eingebaut.

Trotz der Unterschiede sind beide Testtypen für die Usability von Systemen sehr wichtig, auch wenn qualitative Tests häufiger genutzt werden. [vgl. (Budiu, 2017)]

Im Zuge dieser Arbeit wird ein qualitativer Usability Test durchgeführt. Qualitative Daten und das Feedback der Testuser sind wichtig, um Designfehler zu erkennen den Prototypen in weiterer Folge dementsprechend anzupassen und zu erweitern. Da in dieser Arbeit aber zwei Prototypen miteinander verglichen werden, wird ein kleiner Teil des Tests auch quantitative Daten sammeln, um herauszufinden, welche Aspekte der jeweiligen Prototypen besser gelungen sind. Quantitative Daten alleine würden in diesem Fall allerdings nicht hilfreich sein, da es weder bekannte Daten von Konkurrenzprodukten noch festgelegte Standards gibt.

Die kommenden Abschnitte befassen sich mit zwei verschiedenen Testverfahren.

6.2.4 Testverfahren 1 – Thinking Aloud

Eine bekannte und häufig genutzte Methode für qualitative Usability Tests ist der sogenannte Thinking Aloud Test [vgl. (Budi, 2017)]. Ein solcher Test wird durchgeführt, in dem die Testuser, während sie die Aufgabe erledigen, ihre Gedanken laut aussprechen. Es gibt drei wichtige Faktoren, die man bei einem Thinking Aloud Test beachten sollte. Erstes sollten die Testuser repräsentativ für alle Usergruppen sein, zweitens sollten die Aufgaben, die sie ausführen, auch repräsentativ für das System sein und drittens sollte der Versuchsleiter oder die Versuchsleiterin leise sein und das Reden den Testusern überlassen, Ausgenommen davon sind nur, kleine Anstöße, um die Testuser zum Weiterreden zu bewegen.

Thinking Aloud Tests haben einige Vorteile. Sie sind billig, da man keine spezielle Ausrüstung braucht, sondern sich einfach neben die Testperson setzt, zuhört und sich Notizen macht. Diese Methode ist auch sehr flexibel, und kann in jeder Phase der Entwicklung eingesetzt werden. Somit ist sie auch sehr gut für agile Projekte geeignet. Außerdem kann sie mit jedem System durchgeführt werden. Auch für Anfänger als Versuchsleiter oder Versuchsleiterin ist ein Thinking Aloud Test gut geeignet: die Basis für diese Art von Test ist sehr leicht und schnell gelernt und es ist schwer, etwas falsch zu machen, solange man den Testpersonen keine Aussagen in den Mund legt.

Es gibt aber auch einige Nachteile. Diese Art von Test ist für die Testpersonen sehr unnatürlich, da sie normalerweise nicht jeden Schritt verbalisieren müssen. Manchmal muss der Versuchsleiter oder die Versuchsleiterin etwas für die Testuser klarstellen oder sie auffordern über etwas zu sprechen. In diesen Fällen können die User beeinflusst werden und ändern vielleicht dadurch ihr Verhalten, was zu einer Verfälschung der Ergebnisse führen kann. Thinking Aloud Tests sind außerdem nicht immer ausreichend, in manchen Fällen muss eine andere Methode hinzugezogen werden, um ein einheitliches Gesamtbild zu bekommen.

[vgl. (Nielsen, 2012)]

6.2.5 Testverfahren 2 – Fragebogen

Fragebögen sind in vielen Forschungsbereichen und auch im Usability Bereich ein beliebtes Mittel, um an Daten zu gelangen. Die Daten die ein Fragebogen liefert sind meist quantitativ, können aber gut mit qualitativen Methoden kombiniert werden. Ein Fragebogen ist ein recht billiges Mittel für die Erhebung von quantitative Daten und sehr schnell erstellt, sofern der Ersteller weiß, wie ein guter Fragebogen auszusehen hat. Eine häufiger Kritikpunkt in der Usability Branche ist, dass viele, die diese Methode einsetzen, nicht wissen wie und wo man sie richtig anwendet und so irreführende oder fehlerhafte Daten kreieren.

[vgl. (Brown, 2024)]

7 Praktischer Teil

Im praktischen Teil der Arbeit werden die Erkenntnisse aus dem theoretischen Teil genutzt, um die Prototypen und den folgenden Usability Test umzusetzen.

7.1 Umsetzung der Prototypen

In diesem Abschnitt wird die Umsetzung der zwei Prototypen bezüglich der verwendeten Technologien beschrieben und ihre gemeinsame Basis erklärt.

7.1.1 Verwendete Technologien

Für die Erstellung der Prototypen wird Proto.io eingesetzt. Dabei handelt es sich um eine Webseite, die zur Erstellung von Prototypen und Flows dient, die sich auch als Preview auf verschiedenen emulierten Geräten mit unterschiedlichen Bildschirmgrößen anzeigen lassen. Das gesamte Projekt kann als HTML, PDF oder als ZIP-Datei mit PNGs exportiert werden.

Dieses Feature mache ich mir zunutze, um das Projekt auf GitHub abzulegen. So kann ich auf ältere Versionen zurückgreifen und verfüge zugleich über eine zeitliche Dokumentation des Projekts.

Ein weiteres Kriterium bei der Auswahl dieser Technologie war, dass mir Proto.io schon durch vergangene Projekte bekannt war, was den Vorteil bringt, dass ich direkt mit den Prototypen starten kann.

7.1.2 Gemeinsame Basis

Die Basis der App ist so aufgebaut, wie in Kapitel 6.1.1 beschrieben. Sowohl die Statusbar als auch die Tab app bar, sind farblich an das von Ionio vorgegebene Farbschema angepasst. Zusätzlich befindet sich in der Tab app bar ein Switch Icon, um zwischen den Entwürfen wechseln zu können.



Abbildung 5: Statusbar und Tab app bar (eigene Darstellung)

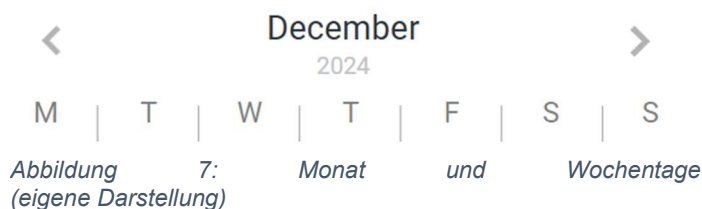
Die Navigationsleiste ist auch farblich passend hinterlegt, aber mit einer leichten Durchsichtigkeit, damit sie einerseits nicht so auffällt und von der App ablenkt, andererseits aber trotzdem farblich ins Schema passt.



Abbildung 6: Navigationsleiste (eigene Darstellung)

7.1.3 Erster Prototyp

Der erste Entwurf basiert, wie in Kapitel 6.1.2 beschrieben, auf der Monatsansicht des Google Kalenders. Unter der Tab app bar befindet sich der Monat, das Jahr und Pfeile, um zwischen den Monaten zu wechseln. Da der Wunschplan, nur für Planung in der nahen Zukunft genutzt wird, gibt es hier keine Möglichkeit, das Jahr zu wechseln.



Darunter beginnt der eigentliche Kalender. Die Tage sind als Tabelle dargestellt, in der jeder Tag durch ein hochkant Rechteck mit der Datumszahl dargestellt wird. Das Datum ist farblich hinterlegt, um den Planstatus anzuzeigen; dabei steht, grün für den Wunschplan. Die orange Schattierung stellt eine Übertretung der Besetzungsvorgabe dar. Die Dienste, die sich die Nutzer und Nutzerinnen wünschen, sind als kleine Rechtecke in den Tagen eingetragen. Sie zeigen die Person und die ersten Buchstaben des gewünschten Dienstes. Der Dienst ist zusätzlich farblich hinterlegt, um die Dienste auch unterscheiden zu können, falls die Dienstbezeichnung oder der Name der Person zu lang sind.

29	30	31	1	2	3	4
C - Früh...	C - Urlaub			B - ?		
B - Nach...						
5	6	7	8	9	10	11
	D - Nach...					
12	13	14	15	16	17	18
			Ich - Frü...			
19	20	21	22	23	24	25
	Ich - Nach...	Ich - Urlaub				
26	27	28	29	30	1	2
3	4	5	6	7	8	9
						+

Abbildung 8: Kalender - Entwurf 1 (eigene Darstellung)

Der Floating Action Button ruft eine übergreifende Komponente auf, die im Kapitel 6.2.1 beschrieben wird.

7.1.4 Zweiter Prototyp

Der zweite Entwurf orientiert sich an der Monatsansicht des Apple Kalenders wie in Kapitel 6.1.3 beschrieben. Auch in diesem Entwurf befindet sich direkt unter der Tab app bar der Monat, das Jahr und die dazugehörigen Steuerelemente (Abbildung 7), da aber der Kalender selbst anders dargestellt ist, sind die Wochentage nicht gleich designt wie in der Abbildung 7.

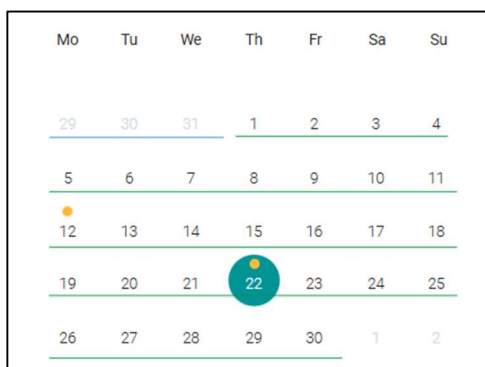


Abbildung 9: Kalender - Entwurf 2 (eigene Darstellung)

der orange Punkt über der Zahl. Das Datum der Dienste, die in der Liste unterhalb angezeigt werden, ist mit einem dem Ionio Farbschema entsprechendem Kreis hinterlegt.

Die Liste darunter zeigt einerseits die Dienste und andererseits die Besetzungsvorgabe an. Falls diese über oder unterschritten ist, ist sie orange hinterlegt. Die Dienste selber sind wie im Entwurf 1 aufgebaut: der Name der Person, die sich den Dienst gewünscht hat, und anschließend die Dienstbezeichnung. Auch hier sind die Dienste farblich hinterlegt, um auf den ersten Blick sehen zu können, welche Dienste schon gewünscht sind, ohne die Bezeichnungen lesen zu müssen.

Die Anzeige für die Grenzen der

Die Daten sind wie beim originalen Kalender von Apple angeordnet. Die Punkte, die unter den Zahlen erscheinen, wurden aber, wie in Kapitel 6.1.3 beschrieben anders verwendet. Um den Planstatus darzustellen, wurde eine farbliche Linie unterhalb der Zahlen gewählt. Aus diesem Grund befindet sich

der orange Punkt über der Zahl. Das Datum der Dienste, die in der Liste unterhalb angezeigt werden, ist mit einem dem Ionio Farbschema

entsprechendem Kreis hinterlegt.

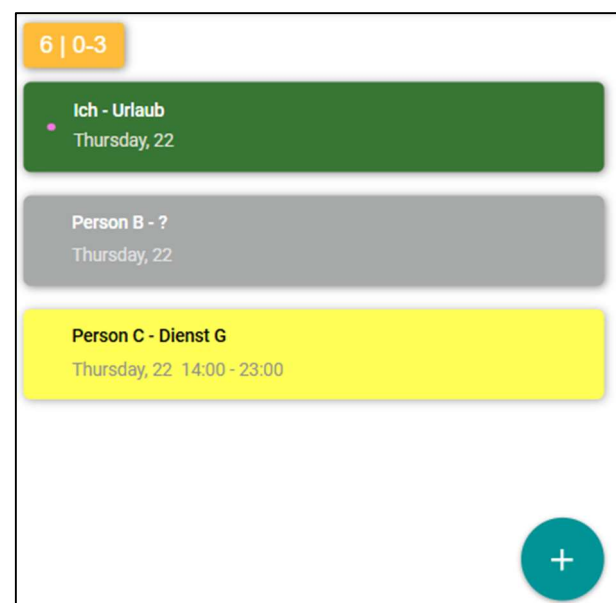


Abbildung 10: Liste der Dienste des ausgewählten Datums

gefilterten Dienste erscheint, passend zum Punkt im Kalender, orange, wenn die Grenzen über- oder unterschritten wurden.

7.1.5 Komponenten

In diesem Abschnitt werden zwei Komponenten beschrieben, die in beiden Prototypen genutzt werden.

7.1.5.1 Dienst hinzufügen - Dialog

The dialog box is titled 'Dienst hinzufügen'. It contains a 'Dienst:' label followed by a dropdown menu showing 'Dienst'. Below this is a 'Datum:' label followed by a text input field containing 'TT.MM.JJJJ' and a calendar icon. At the bottom left is a checkbox labeled 'Mehrtägig' which is unchecked. At the bottom right is a button labeled 'Hinzufügen'.

Abbildung 12: Dialog – eintägig
(eigene Darstellung)

Der Dialog, um einen Dienst hinzuzufügen wird durch den Floating Action Button aufgerufen. Der Dialog verfügt über folgende Funktionalitäten: ein Dropdownmenü, um den Dienst auszuwählen; ein Feld, um entweder das Datum selbst einzutragen oder mit Hilfe des Kalender-Icons den Datumswähler aufzurufen; eine Checkbox, um einen Dienst

für mehrere Tage einzutragen; und einen Button, um den Dienst hinzuzufügen. Sobald der Dialog geöffnet ist, zeigt der Floating Action Button ein X anstelle eines Pluszeichens, damit man den Dialog schließen können, ohne einen Dienst hinzuzufügen.

The dialog box is titled 'Dienst hinzufügen'. It contains a 'Dienst:' label followed by a dropdown menu showing 'Dienst'. Below this is a 'Datum:' label followed by two stacked text input fields. The top field is labeled 'von:' and contains 'TT.MM.JJJJ'. The bottom field is labeled 'bis:' and contains 'TT.MM.JJJJ'. Each field has a calendar icon to its right. At the bottom left is a checkbox labeled 'Mehrtägig' which is checked. At the bottom right is a button labeled 'Hinzufügen'.

Abbildung 11: Dialog – mehrtägig
(eigene Darstellung)

7.1.5.2 Datumswähler

Wenn das Icon im Dialog verwendet wird, erscheint der Datumswähler in der Variante des „Modal date pickers“. Diese Variante wurde gewählt, da sie sich, wie in Kapitel 6.2.2 beschrieben, gut für mobile Anwendungen eignet. Am oberen Rand des Fensters, steht „Datum auswählen“, damit erkenntlich ist



Abbildung 13: Datumswähler (eigene Darstellung)

wofür dieses Fenster genutzt werden soll. Darunter stehen der Wochentag und das Datum allerdings ohne Jahr. Unter einer abgrenzenden Linie befindet sich ein Dropdownmenü, um das Jahr zu ändern und daneben Pfeile, um zwischen den Monaten zu wechseln. Der Nutzer oder die Nutzerin kann allerdings auch durch Swipen des Kalenders die Monate wechseln. Am rechten unteren Rand des Fensters befinden sich die Buttons „Abbrechen“ und „Ok“.

7.2 Usability Test

Um die erstellten Prototypen zu testen, wurde ein Usability Test durchgeführt. Die Theorie hinter dem Test ist in Kapitel 6.3 beschrieben. Um bei den Tests ein breites Spektrum von möglichen Userkenntnissen abdecken zu können, wurden Testpersonen aus verschiedenen Bereichen hergezogen: Testperson A ist teilweise an der Entwicklung der Desktop Applikation beteiligt; Testperson B ist als Customer Support gut mit der Applikation vertraut und steht eng mit den Kunden in Verbindung, außerdem hilft sie als Technical Support dem Entwicklungsteam beim Deploy der Software; Testperson C ist der Product Owner des Entwicklungsteams und kennt sich somit mit der Applikation aus, im Gegensatz zu den Testpersonen A und B allerdings nicht im technischen Bereich. Testperson D ist eine unbeteiligte Person, die die Applikation nicht kennt, aber einen technischen Hintergrund hat.

Das Testverfahren, das für den Test ausgewählt wurde, ist eine Kombination aus beiden in Kapitel 6.3 beschriebenen Testverfahren. Zuerst wurde mit den Testpersonen ein Thinking Aloud Test durchgeführt, bei dem sie vorgegebene

Aufgaben in beiden Prototypen erledigen mussten. Da dies aber nur die funktionale Seite der Prototypen abdeckt, füllten die Teilnehmenden anschließend einen Fragebogen aus, der sich auf das Design der Prototypen und den Vergleich zwischen diesen konzentrierte.

7.2.1 Fragebogen

Der Fragebogen, den die Testpersonen bekommen haben, ist in drei Teile geteilt. Der erste Teil stellt Fragen zum ersten Prototyp, der zweite die gleichen Fragen zum zweiten Prototyp und der dritte Teil vergleicht beide Prototypen direkt miteinander.

In den ersten beiden Teilen wird anhand einer Skala von eins bis fünf (1 = gar nicht erkennbar, 5 = sehr gut erkennbar) die Erkennbarkeit der Planungsperiode, der Besetzungsvorgabe und der Dienste sowie der dazugehörigen Personen abgefragt. Außerdem wird gefragt, wie die Bedienbarkeit des Kalenders auf einer Skala von eins bis fünf (hier 1 = gar nicht intuitiv, 5 = sehr intuitiv) einzuordnen ist.

Im dritten Teil wählen die Testperson aus, in welchem der beiden Prototypen ihrer Meinung nach die Designs der vorher abgefragten Punkte besser umgesetzt wurden. Zusätzlich gibt es am Schluss noch die Möglichkeit, Anmerkungen zum Design und zu den Funktionalitäten abzugeben.

7.2.2 Durchführung

Die Durchführung fand einzeln mit den Testpersonen statt. Die Vorbereitung der Tests erfolgte auf zwei Arten, da ein Test remote stattfinden musste. Bei den Tests, die vor Ort ausgeführt wurden, bekamen die Testpersonen die Prototypen vorgelegt und konnten so nach der Einleitung direkt starten. Bei dem Remotetest musste sich die Testperson zuerst die Proto.io App herunterladen um auf die erstellten Prototypen zugreifen zu können.

In der Einleitung wurde den Personen erklärt was ein Thinking Aloud Test ist, und es wurde der Ablauf des Tests erläutert. Sie sollten für jede Aufgabe, die sie bekommen, eine Bildschirmaufnahme starten und dazusagen, welche

Testperson sie sind, um die Videos später zuordnen zu können. Dann sollten sie die Aufgabe ausführen. Testperson D, die einzige Person, die das Produkt gar nicht kennt, hat zusätzlich eine kurze Erklärung bekommen, was die Aufgabe der Desktopapplikation und somit auch die Aufgabe der Prototypen ist, sowie den Hinweis erhalten, dass Planungsperioden farblich gekennzeichnet werden. Alle Testpersonen wurden am Schluss noch informiert, dass nach dem Thinking Aloud Test noch ein Fragebogen auszufüllen sei.

Die Tests wurden so ausgeführt, wie in der Einleitung beschrieben. Die Aufgaben, die auszuführen waren, lauteten:

1. Lege einen eintägigen Dienst an
2. Lege einen mehrtägigen Dienst an
3. Öffne das Menü
4. Beschreibe den Prototypen

Die ersten drei Aufgaben sollten nur im ersten Prototyp ausgeführt werden, da die Funktionen in beiden Prototypen gleich funktionieren. Der vierte Punkt musste hingegen in beiden Prototypen ausgeführt werden.

Nachdem alle Videos aufgenommen wurden, durften die Testpersonen in der Nachbereitung noch Feedback geben und, im Falle von Unklarheiten während des Tests, nachfragen. Daraufhin wurde ihnen der Fragebogen erklärt. Diesen durften sie allerdings alleine ausfüllen.

7.2.3 Auswertung

In diesem Abschnitt werden die Videos des Thinking Aloud Tests und der Fragebogen ausgewertet. Zuerst werden die Videos der Aufgaben 1 bis 3 (siehe Kapitel 7.2.2) besprochen. Anschließend werden die Videos der letzten Aufgabe ausgewertet. Zuletzt werden die Antworten aus dem Fragebogen analysiert.

7.2.3.1 Aufgaben 1 bis 3

Aufgabe 1 (Arbeitsanweisung: füge einen eintägigen Dienst ein) wurde von allen Testpersonen grundsätzlich souverän gelöst. Allerdings gab es auch ein paar kleine Schwierigkeiten. Die größte Schwierigkeit, die drei der vier Testpersonen

hatten, war, dass man einen neuen Dienst nicht hinzufügen konnte, in dem man auf einen Tag im Kalender tippt, sondern ausschließlich über den Floating Action Button. Ein weiteres Hindernis war, dass das Auswählen des Datums im Datumswähler das Datum nicht im dafür vorgesehen Feld eingetragen hat. Testperson A merkte an, dass sie es intuitiver finden würde, wenn das Datumsfeld über dem Dienst-Dropdownmenü wäre, da man einen Dienst eher in dieser Reihenfolge eintragen würde (d.h. man würde immer zuerst das Datum auswählen).

Aufgabe 2 (Arbeitsanweisung: füge einen mehrtägigen Dienst ein) war für die Testpersonen leicht zu lösen. Es gab keine Schwierigkeiten, was auch daran lag, dass der Dialog der gleiche wie in Aufgabe eins ist. Die Testpersonen hatten allerdings ein paar Anmerkungen zum Design sowie zur Funktionalität: Die Checkbox, mit der eine Datumsspanne eingetragen werden kann, sollte laut zwei der Probanden und Probandinnen über der Datumseingabe sein, um dem natürlichen Lesefluss zu folgen. Des weiteren wurde angemerkt, dass, wenn schon ein Datum in das Datumsfeld eingegeben worden ist und die Checkbox erst danach ausgewählt wird, das gewählte Datum in der „von“-Box erscheinen sollte. Eine andere Testperson hat angemerkt, dass es angenehm wäre, wenn man die Datumsspanne direkt in einem Dialog auswählen könnte, in dem man zuerst das Anfangs- und dann direkt das Enddatum auswählt, ohne den Datumswähler dazwischen schließen zu müssen.

Die dritte Aufgabe (Arbeitsanweisung: öffne das Menü) hat keiner Person Probleme bereitet, und es gab auch keinerlei Anmerkungen zu diesem Punkt.

Als allgemeine Anmerkung nannten zwei der Testuser und Testuserinnen, dass die Klickflächen der Buttons zu klein und deshalb schwer zu treffen seien.

7.2.3.2 Aufgabe 4

Die vierte Aufgabe (Arbeitsanweisung: Beschreibe den Prototyp) ist in zwei Teile geteilt, da die Testpersonen jeden Prototypen extra beschreiben sollten.

Bei der Beschreibung des ersten Prototyps (Abbildung 14) konnten von allen Personen die Dienste richtig erkannt werden, eine Person verwechselte allerdings anfangs den Dienst mit der Person, erkannte diesen Fehler aber selbständig während der Aufnahme. Zwei Personen erwähnten auch, dass die Dienste durch ihre farbliche Kennzeichnung gut unterscheidbar sind. Testperson D fand auch, dass die mehrtägigen Dienste gut als solche erkennbar sind, da sie in der grafischen Darstellung verbunden sind.

Drei der vier Personen identifizierten die Planungsphase korrekt. Die vierte Person, Testperson D, kennt das originale Produkt nicht und ging auf diesen Punkt überhaupt nicht ein.

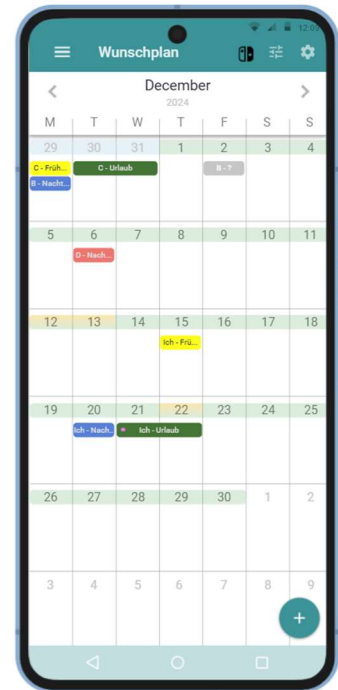


Abbildung 14: Prototyp 1
(eigene Darstellung)

Die Besetzungsvorgabe wurde von keiner Person erkannt. Testperson B mutmaßte, dass es sich um die Markierung für einen Feiertag handeln könnte, da Feiertage auch in der Desktopapplikation markiert werden.

Nur Testperson C ging auf den lila Punkt in einem der Dienstblöcke ein, da sie den Punkt aber als rosa sah, identifizierte sie eine falsche Funktion (Markierung eines Wunsches anstelle eines unbestätigten Wunsches).

Als Anmerkung gab Testperson B noch zu bedenken, dass es intuitiver wäre, wenn der Dialog zum Einfügen eines Dienstes sich auch schließen lässt, wenn man außerhalb des Dialogs auf den Dienstplan tippt. Testperson C wünschte sich, dass man auf einzelne Zellen tippen können sollte, um Details über den betreffenden Tag zu sehen.

Bei der Beschreibung des zweiten Prototyps (Abbildung 15) haben wieder alle außer der Testperson D die Kennzeichnung der Planungsperiode als solche erkannt, Testperson B hat allerdings angemerkt, dass die Kennzeichnung in diesem Prototyp besser erkennbar ist.

Die Besetzungsvorgabe wurde wieder von keiner Person erkannt. Die Punkte über dem Datum, die eine Übertretung der Besetzungsvorgabe darstellen, wurden als Markierungen für Tage mit Dienstseinträgen wahrgenommen. Die gleichfarbige Anzeige darunter wurde nur von Testperson A mit dem Punkt in Verbindung gebracht.

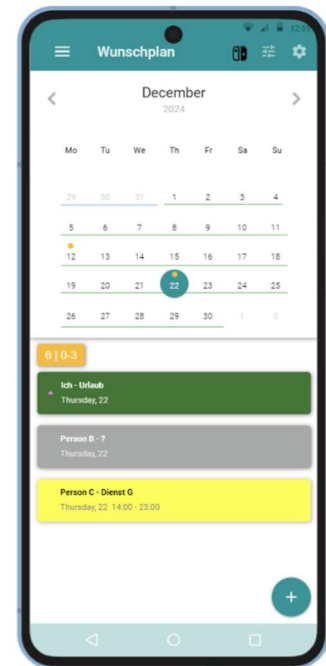


Abbildung 15: Prototyp 2 (eigene Darstellung)

Zwei der vier Testpersonen gingen davon aus, dass der markierte Tag den heutigen Tag markiert.

Der lila Punkt neben dem eigenen eingetragenen Dienst wurde nur von Testperson A korrekt identifiziert. Er wurde zwar auch von Testperson C angesprochen, diese wusste aber nicht, was dieser Punkt darstellt, obwohl sie ihn in der Ansicht des ersten Prototypen noch erkannt hat.

Testperson D merkte an, dass in dieser Ansicht mehrtägige Termine nicht ersichtlich sind. Testperson C meinte hingegen, dass sie das Datum in der Anzeige der Dienste nicht für nötig hält, da der ausgewählte Tag schon ersichtlich ist.

7.2.3.3 Fragebogen

Wie in Kapitel 7.2.1 beschrieben, ist der Fragebogen in drei Abschnitte gegliedert.

Im ersten Abschnitt wurden die Testpersonen zum ersten Prototyp (Abbildung 14) befragt.

Im ersten Punkt war die Erkennbarkeit der Planungsperiode auf einer Skala von eins bis fünf (1 = gar nicht erkennbar, 5 = sehr gut erkennbar) zu bewerten. Drei

Personen wählten vier und eine Person fünf, was bedeutet, dass die Planungsperiode für alle Testpersonen gut als solche erkennbar ist.

Im zweiten Punkt wird die Erkennbarkeit der Besetzungsvorgabe abgefragt. Zwei Personen wählten zwei auf der Skala, eine Person eins und eine Person wählte vier. Daraus kann man ableiten, dass die meisten Testpersonen die Besetzungsvorgabe nicht gut erkennbar finden.

Im dritten Punkt geht es um die Erkennbarkeit der Dienste und der dazugehörigen Personen. Zwei Testpersonen wählten hier drei, eine vier und eine sogar fünf. Daraus lässt sich schließen, dass die Dienste im großen und ganzen erkennbar sind.

Im vierten Punkt wird ermittelt, wie intuitiv die Bedienung des Kalenders ist (1 = gar nicht intuitiv, 5 = sehr intuitiv). Zwei Personen wählten vier, eine drei und eine fünf. Im Durchschnitt kann die Bedienung also als intuitiv mit Verbesserungspotential beschrieben werden.

Im zweiten Abschnitt des Fragebogens wurden die Testpersonen zum zweiten Prototyp (Abbildung 15) befragt.

Der erste Punkt bezieht sich wieder auf die Planungsperiode. Bei diesem Prototyp wählten die meisten Testpersonen eine gute Erkennbarkeit (zwei wählten fünf, eine vier), allerdings wählte eine Person eine schlechte Erkennbarkeit (zwei auf der Skala). Diese Person hat beim ersten Prototyp fünf gewählt, woraus man schließen kann, dass diese Person die Darstellung im ersten Prototyp besser findet.

Der zweite Punkt bezieht sich wieder auf die Besetzungsvorgabe. Hier waren sich die Testpersonen etwas einiger, denn zwei Personen wählten drei und die anderen beiden vier. Das bedeutet, dass in diesem Prototyp die Besetzungsvorgabe relativ gut erkennbar ist.

Der dritte Punkt betrifft erneut die Dienste und die dazugehörigen Personen. Bei diesem Punkt sind sich alle Testpersonen einig, dass dieses Feature sehr gut erkennbar ist (5 auf der Skala).

Im dritten Abschnitt des Fragebogens werden verschiedene Aspekte der Prototypen miteinander verglichen und die Befragten müssen angeben, bei welchem der beiden Prototypen sie diese besser umgesetzt finden.

Der erste Punkt betrifft wieder die Planungsperiode. Hier sind sich die Befragten uneinig, denn 50% stimmten für jeweils einen Prototyp ab. Daraus kann man schließen, dass die Anzeige der Planungsperiode eine individuelle Vorliebe ist.

Der zweite Punkt fragt die Lesbarkeit der einzelnen Prototypen ab. Hier stimmen 75%, also drei der vier Befragten, für den zweiten Prototyp. Dieser Prototyp wird wahrscheinlich ausgewählt worden sein, da er nur einen Tag anzeigt und deshalb als übersichtlicher wahrgenommen wird.

Im dritten Punkt geht es um die Besetzungsvorgabe. Wieder stimmen 75% der Befragten für den zweiten Prototyp. Das könnte daran liegen, dass nicht nur der Tag eingefärbt ist, sondern auch eine Anzeige, die die Besetzungsvorgabe für diesen Tag angibt.

Der vierte Punkt betrifft den Kalender. Bei diesem Punkt sind sich alle Befragten einig, dass der erste Prototyp diesen besser darstellt.

Im letzten Punkt geht es um die Dienste und die dazugehörigen Personen. Auch hier sind sich alle Befragten einig, dass der zweite Prototyp diesen Punkt besser umsetzt.

Zusätzlich zu den beschriebenen Hauptpunkten konnten die Befragten am Ende des Fragebogens auch Anmerkungen zum Design sowie zur Funktionalität der beiden Prototypen hinterlassen. Die Anmerkungen waren keine Pflichtfelder, trotzdem haben bei jeder Frage jeweils drei der vier Personen geantwortet. Eine befragte Person merkte beim Design an, dass im ersten Prototyp die Kalenderansicht gut lesbar ist, aber die Dienste und Personen nicht. Außerdem ist laut dieser Person die Vermischung der Planungsperiode mit der Besetzungsvorgabe nicht gut erkennbar und diese Elemente sollten daher getrennt werden, nachdem es sich auch um verschiedene Funktionen handelt. Auch im zweiten Prototyp fand die Person den Kalender gut lesbar, auch wenn die Tage etwas klein und dadurch schlechter erkennbar als im ersten Prototyp waren. In diesem Prototyp

hingegen findet diese Person die Lesbarkeit der Dienste sehr gut. Als Feedback für beide Prototypen schrieb die Person:

„Die Toolbar mit dem Menü ist optisch ansprechend und setzt sich gut von dem Rest der Benutzeroberfläche ab. Die Steuerung des angezeigten Monats ist sofort als solche erkennbar. Auch das Plus fürs Einfügen der Dienste ist gut sichtbar und kommuniziert die Funktionalität äußerst intuitiv.“

(Originalzitat aus dem Fragebogen „Usability Befragung – Wunschplan“)

In den Anmerkungen gab es auch ein paar Vorschläge für zukünftige Verbesserungen. Eine Person schlug zum Beispiel vor, dass beide Prototypen verknüpft werden könnten. In diesem Fall würde man im Prototyp 1 auf eine Tageszelle tippen und dadurch in die Detailansicht kommen, die der zweite Prototyp darstellt. Ein weiterer Verbesserungsvorschlag ist, im ersten Prototypen die Eingabeintensität zu nutzen. Ein kurzer Druck auf eine Zelle könnte die Tagesansicht öffnen, während ein langer Druck einen neuen Eintrag erzeugt. Das Erzeugen eines Eintrags durch das Tippen auf einen Tag wurde von mehreren Personen als wünschenswert erwähnt.

8 Resultate

In diesem Kapitel wird es um die gewonnenen Erkenntnisse dieser Arbeit gehen. Das umfasst die technische Umsetzung und die Usability Tests.

8.1 Technische Umsetzung

Die technische Umsetzung der Prototypen ist im Großen und Ganzen gut gelungen. Für zukünftige Projekte würde ich allerdings in einigen Punkten einen anderen Ansatz wählen. In dieser Hinsicht ist der wichtigste Punkt die Wahl der Technologie. Proto.io ist ein gutes Tool für kleine und einfache Prototypen, allerdings eignet es sich nicht sehr gut für Designs, die nicht bereits als Vorlage existieren, oder für Funktionalitäten, die weiter gehen als das Hin- und Herwechseln zwischen Ansichten oder das Aufmachen kleiner Komponenten. Man kann auch keine Daten festlegen, die dynamisch verwendet werden. Proto.io sollte daher, meiner Erfahrung nach, hauptsächlich für sogenannte Klick-Dummys verwendet werden, für kompliziertere Prototypen würde ich davon abraten.

8.2 Usability Test

Der Usability Test hat gezeigt, dass die Prototypen über ein gutes Grundgerüst verfügen, allerdings ist dieses stark ausbaufähig.

Im ersten Prototyp sollte die Darstellung der Planungsperiode überdacht werden. Sie wurde zwar während des Tests erkannt, aber manchmal in der falschen Farbe (zum Beispiel nahm eine Testperson die grüne Farbe als blau wahr). Als Feedback in dieser Hinsicht gab es auch, dass die Planungsperiode durch ihre blasse Färbung nicht so gut zu erkennen ist, wie es im zweiten Prototyp der Fall ist.

Ein weiterer Kritikpunkt ist die Darstellung der Besetzungsvorgabe. Durch die leichte orange Färbung dieser Tage im Kalender merkt man zwar, dass etwas markiert wird, was das genau ist, wird allerdings in dieser Darstellung nicht klar. Ein Vorschlag eines Probanden war es, diese beiden Funktionen grundsätzlich

zu trennen. Wie genau die Besetzungsvorgabe dann angezeigt werden soll, wenn dieser Vorschlag umgesetzt wird, kann am besten durch neue Designvorschläge mit anschließenden Usability Befragungen in Erfahrung gebracht werden.

Der größte Kritikpunkt der Testpersonen beziehungsweise der von ihnen am häufigsten begangene Fehler während des Tests ist, dass die Erstellung eines Dienstes nicht durch das Antippen eines Tages funktioniert. Da dies im Test ein so häufiger Fehler war und auch im Fragebogen mehrmals genannt wurde, muss diese Funktionalität in zukünftige Designs unbedingt eingebaut werden. Ob das Erstellen eines Dienstes dann durch ein einfaches Tippen auf den Tag oder durch einen längeren Druck darauf passiert, wird sich bei weiteren Designversuchen zeigen.

Im zweiten Prototypen sollte der Kalender größer angezeigt werden. In den Anmerkungen des Fragebogens gab es die Rückmeldung, dass dieser durch die kleine Schrift schwer zu lesen sei.

Eine weitere Anregung war, dass das aktuelle Datum nicht am Dienst selbst angezeigt zu werden braucht, da dieses bereits durch die Auswahl im Kalender ersichtlich ist. Wo das Anzeigen des Datums aber sinnvoll wäre, ist im Falle von mehrtägigen Diensten. Dann sollten die Daten, von wann bis wann dieser Dienst eingetragen ist, angezeigt werden.

Eine häufige Fehlinterpretation im zweiten Prototypen waren auch die orangen Punkte der Besetzungsvorgabe. Diese wurden als Tage, an denen man Dienste eingetragen hat, interpretiert. Da Tage, an denen Dienste eingetragen sind, gar nicht angezeigt werden, kann man daraus schließen, dass diese Funktionalität gewünscht wird und daher in zukünftigen Designs berücksichtigt werden sollte. Da dieser Punkt in der Zukunft eine andere Funktion haben könnte, sollte darüber nachgedacht werden, wie eine Über- oder Unterschreitung der Besetzungsvorgabe unmissverständlich angezeigt werden kann.

Auch die Komponenten bedürfen einiger kleiner Designänderungen. Ein häufig genannter Punkt war, dass die Checkbox für einen mehrtägigen Dienst über der Datumseingabe stehen sollte. Damit Hand in Hand geht auch der Vorschlag,

dass die Auswahl des Dienstes erst nach der Datumseingabe erfolgen sollte, da dies die intuitivere Reihenfolge ist.

Wenn die Checkbox nun tatsächlich angeklickt wird, aber vorher schon ein Datum in der Eingabe steht, soll dieses Datum automatisch in das „von“-Feld übernommen werden. Außerdem sollte das Datum auch schon vorausgefüllt sein, sobald man über das Antippen eines Tages im Kalender auf diesen Dialog zugreifen kann.

Ein weiterer Vorschlag für das Wählen des Datums war es, dass der Datumswähler es zulassen sollte, im Falle von mehrtägigen Diensten zwei Daten auszuwählen, um die Spanne zwischen den Daten direkt in einem Dialog einzutragen zu können und dadurch weniger tippen zu müssen.

9 Fazit

Um eine Desktopapplikation auf einem mobilen Endgerät darzustellen, muss man auf zahlreiche Faktoren Rücksicht nehmen. Die Zielsetzung dieser Arbeit war zu zeigen, welche Punkte im Übergangsprozess von einer Desktopdarstellung zu einer mobilen Applikation besonders wichtig sind.

Als erster wichtiger Punkt für den Designprozess einer App hat sich die Auswahl der Technologie erwiesen. Mit der Auswahl der richtigen Technologie kann man manche Details einfacher darstellen und die Funktionalitäten so besser umsetzen.

Weiters sind für das Design selbst ein gutes Verständnis von Grundlagen der Usability und profunde Kenntnisse über etablierte sowie aktuelle Prinzipien der Appgestaltung von Vorteil.

Darüber hinaus ist ein Usability Test ein sehr wichtiger Schritt im Designprozess, da man als Designer oder Designerin das System natürlich nicht so gut kennen kann wie jemand, der tagtäglich damit arbeitet oder an der Entwicklung beteiligt ist. Dadurch kennt man als Designer oder Designerin gewisse typische Abläufe nicht und macht diese dadurch unabsichtlich komplizierter. Des weiteren weiß man, wenn man die App selbst designt hat, was die Zielsetzung von bestimmten Abläufen war und wie die einzelnen Funktionen anzuwenden sind. Daher erkennt man offensichtliche Probleme im eigenen Design nicht und ist daher auf den Blick von Außen angewiesen, der von Testpersonen geboten wird.

Wenn man diese Punkte bei der eigenen Umsetzung beachtet, kann man erfolgreich eine mobile Anwendung entwickeln, die der bestehenden Desktopapplikation um nichts nachsteht.

10 Literaturverzeichnis

Brown, M., 2024. *Should You Run a Survey?*. [Online]

Available at: <https://www.nngroup.com/articles/should-you-run-a-survey/>

[Zugriff am 09 Dezember 2024].

Budiu, R., 2017. *Quantitative vs. Qualitative Usability Testing*. [Online]

Available at: <https://www.nngroup.com/articles/quant-vs-qual/>

[Zugriff am 05 Dezember 2024].

DI Dr. Lang, E. C. et al., 2019. *Schutz sensibler Daten*. Wien: BMASGK, Bundesministerium für Arbeit, Soziales, Gesundheit und Konsumentenschutz.

Google, Material Design, kein Datum *Date pickers*. [Online]

Available at: <https://m3.material.io/components/date-pickers/>

[Zugriff am 05 Dezember 2024].

Google, Material Design, kein Datum *Top app bars*. [Online]

Available at: <https://m3.material.io/components/top-app-bar/guidelines>

[Zugriff am 05 Dezember 2024].

Hoober, S. & Berkman, E., 2011. *Designing Mobile Interfaces*. Sebastopol, CA: O'Reilly Media, Inc..

Nielsen, J., 1993. *Usability Engineering*. s.l.:Academic Press, Inc.

Nielsen, J., 2012. *Thinking Aloud: The #1 Usability Tool*. [Online]

Available at: <https://www.nngroup.com/articles/thinking-aloud-the-1-usability-tool/>

[Zugriff am 19 Dezember 2024].

11 Anhang – GitHub Repository

<https://github.com/KathrinGitHub/BachelorArbeit>