

Fakultät für
Informatik und
Mathematik



Entwicklung einer API-basierten Suchfunktion zur Vereinigung mehrerer Datenquellen in der "My BMW" App

Helena Berndt

Bachelorarbeit Informatik

Prüfer:

Prof. Dr. Lars Wischhof, Hochschule München

Firmenlogo

Betreuer:

Daniel Abram, BMW Group

xx.xx.xxxx

Erklärung

Helena Berndt, geb. 01.03.2002 (IF7, WS 2024/2025)

Hiermit erkläre ich, dass ich die Bachelorarbeit selbständig verfasst, noch nicht anderweitig für Prüfungszwecke vorgelegt, keine anderen als die angegebenen Quellen oder Hilfsmittel benutzt sowie wörtliche und sinngemäße Zitate als solche gekennzeichnet habe.

München, xx.xx.xxxx

.....

Unterschrift

Zusammenfassung

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nam et turpis gravida, lacinia ante sit amet, sollicitudin erat. Aliquam efficitur vehicula leo sed condimentum. Phasellus lobortis eros vitae rutrum egestas. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae; Donec at urna imperdiet, vulputate orci eu, sollicitudin leo. Donec nec dui sagittis, malesuada erat eget, vulputate tellus. Nam ullamcorper efficitur iaculis. Mauris eu vehicula nibh. In lectus turpis, tempor at felis a, egestas fermentum massa.

Danksagungen

Interdum et malesuada fames ac ante ipsum primis in faucibus. Aliquam congue fermentum ante, semper porta nisl consectetur ut. Duis ornare sit amet dui ac faucibus. Phasellus ullamcorper leo vitae arcu ultricies cursus. Duis tristique lacus eget metus bibendum, at dapibus ante malesuada. In dictum nulla nec porta varius. Fusce et elit eget sapien fringilla maximus in sit amet dui.

Mauris eget blandit nisi, faucibus imperdiet odio. Suspendisse blandit dolor sed tellus venenatis, venenatis fringilla turpis pretium. Donec pharetra arcu vitae euismod tincidunt. Morbi ut turpis volutpat, ultrices felis non, finibus justo. Proin convallis accumsan sem ac vulputate. Sed rhoncus ipsum eu urna placerat, sed rhoncus erat facilisis. Praesent vitae vestibulum dui. Proin interdum tellus ac velit varius, sed finibus turpis placerat.

Inhaltsverzeichnis

Zusammenfassung	i
Danksagungen	ii
Abbildungsverzeichnis	iii
Tabellenverzeichnis	iv
Abkürzungsverzeichnis	v
1 Einleitung	1
1.1 Hintergrund	1
1.2 Motivation	2
1.3 Ziele dieser Bachelorarbeit	3
1.4 Abgrenzung	3
1.5 Aufbau der Arbeit	4
2 Grundlagen	5
2.1 Suchfunktionen in mobilen Anwendungen	5
2.2 Flutter	6
2.3 Die MyBMW-App	7
2.4 API-Entwicklung	8
3 Vorbereitung und Analyse der Ausgangssituation	10
3.1 Allgemeine Funktionen und Anforderungen an eine Suchfunktion .	10
3.2 Recherche und Analyse vergleichbarer Suchfunktionen	11
3.2.1 Suchfunktionen in Automotive Apps	12

INHALTSVERZEICHNIS

3.2.2	Suchfunktionen in Apps anderer Bereiche	12
3.3	Ist-Analyse der vorhandenen Daten und Datenquellen	14
4	Konzeption	15
4.1	Anforderungsanalyse der Suchfunktion	15
4.1.1	Systematische Erfassung der Anforderungen	16
4.1.2	Konzeptentwurf	18
4.2	Konzeption der API	20
4.2.1	Schnittstellen-Design	20
4.2.2	Architektur-Entwurf	21
5	Implementierung der API und Suchfunktion	23
5.1	Implementierung der API	23
5.1.1	Vertiefte Hintergründe der API	23
5.2	Implementierung des Suchfunktions-Prototyps	23
5.3	Benutzeroberfläche	23
6	Evaluierung	25
6.1	Anforderungsabgleich	25
6.2	Bewertung des Konzepts anhand von Beispielszenarien	25
6.3	Evaluierung der Suchfunktion	25
6.3.1	Evaluierung der Suchergebnisse	25
6.3.2	Nutzerbefragung zur Suchfunktion	25
6.4	Vergleich mit Daten	26
7	Fazit	29
7.1	Zusammenfassung der Arbeit	29
7.2	Ausblick	29
	Anhang 1: Einige Extras	31
	Anhang 2: Noch mehr Extras	32
	Literatur	33

Abbildungsverzeichnis

4.1	BeschriftungToBeDone	18
4.2	BeschriftungToBeDone	19
4.3	BeschriftungToBeDone	21
4.4	BeschriftungToBeDone	22

Tabellenverzeichnis

6.1	Das ist die Tabellenbeschriftung. Suspendisse blandit dolor sed tellus venenatis, venenatis fringilla turpis pretium.	28
-----	--	----

Abkürzungsverzeichnis

API	A pplication P rogramming I nterface
CRUD	C reate R ead U pdate D eleate
JSON	J ava S cript O bject N otation
REST	R Epresentational S tate T ransfer
SDK	S oftware D evelopment K it
URL	U niform R esource L ocator

1 | Einleitung

„Mit der neuen App-Generation gehen wir einen weiteren Schritt in der Gestaltung des digitalen Kundenerlebnisses rund um unsere Fahrzeuge [...]. Mit der My BMW App [...] integrieren wir unsere Fahrzeuge nahtlos in den digitalen Lifestyle unserer Kunden.“ Mit diesen Worten des damaligen Senior Vice President der 'Connected Company' der BMW Group, Peter Henrich, wurde die MyBMW App im Jahr 2020 auf den Markt eingeführt. [1]

1.1 Hintergrund

Die BMW Group ist ein renommiertes Unternehmen der Automobilindustrie, das sich auf die Produktion von Fahrzeugen der Premiumklasse fokussiert. Das Portfolio der BMW Group umfasst die Marken BMW, Mini, Rolls Royce und BMW Motorrad. [2], [3] Die BMW Group ist u.a. mit einer Absatzrate von 2,5 Millionen Autos im Jahr 2023 ein signifikanter Faktor für den Wohlstand Europas [4]. Der Automobilsektor trägt mit einem Anteil von 6,1 % zur gesamten Beschäftigung in der EU bei [5] und generiert etwa 7 % des BIP in Europa [6].

Der allgemeine Trend der Digitalisierung zeichnet sich auch in der Automobilindustrie ab. Das digitale Erlebnis eines Produktes ist für Marken zu einem entscheidenden Unterscheidungsmerkmal geworden [7]. Es wird prognostiziert, dass Geschäftsmodelle, wie etwa Konnektivitätsdienste die Einnahmequellen um etwa 30 % steigern könnten [8], [9]. Im Bezug auf diese Entwicklung haben bereits alle Premiumhersteller auf die Entwicklung reagiert und investieren nun u.a. in die

Entwicklung und Bereitstellung von Apps, die oftmals Remote-Funktionen anbieten und es ermöglichen, auf das Auto und dessen Daten über ein Handy zuzugreifen [10]. Daraus resultieren auch Geschäftsstrategien, die eine detaillierte Analyse der Kundendaten ermöglichen, um personalisierte Lösungen zu entwickeln [9].

1.2 Motivation

Die Entwicklung und der Vertrieb von Apps stellen eine Möglichkeit dar, digitale Innovationen zu fördern, wie es die BMW Group mit der MyBMW App bereits umgesetzt hat. Diese Applikation bietet den Kunden eine universelle Schnittstelle zu ihrem Fahrzeug sowie zu einer Vielzahl weiterer Produkte und Dienstleistungen von BMW [11].

Die Bereitstellung einer mobilen Anwendung knüpft an eine Entwicklung an, in der die Bedeutung von Apps kontinuierlich zunimmt. Weltweit nutzen mehr als drei Viertel der über 10-jährigen Bevölkerung ein Mobiltelefon, in Europa sind es 93 % [12]. Die Anzahl der jährlichen Downloads von Apps hat sich zwischen den Jahren 2018 und 2023 mehr als verdoppelt und steigt weiterhin an [13].

Der Wettbewerb auf dem App-Markt ist sehr intensiv. Einige wenige Apps dominieren den Großteil des Marktes, was sich daran zeigt, dass Nutzer bis zu 95 % ihrer Zeit mit ihren persönlichen Top-10-Apps verbringen [14]. Insgesamt sind Apps zum Hauptkonsumenten von internetbasierten Diensten geworden [15].

Die zuvor dargestellten Punkte verdeutlichen die Notwendigkeit, dass insbesondere Unternehmen der Automobilindustrie die Entwicklung von Apps vorantreiben müssen, um ihre Kunden zu überzeugen. Dabei ist auch die kontinuierliche Weiterentwicklung der Apps mit der Einführung neuer Funktionen von entscheidender Bedeutung, um das Kundenerlebnis langfristig zu optimieren und die Kundenzufriedenheit zu erhöhen. Die Implementierung einer Suchfunktion ist eine Methode zur Optimierung einer mobilen Anwendung. Sie ermöglicht es Kunden, Inhalte innerhalb der App effizienter zu finden, was zu einer Verbesserung des Nutzungserlebnisses und der Benutzerzufriedenheit führt.

1.3 Ziele dieser Bachelorarbeit

- Forschungsfrage Worum geht es in der wissenschaftlichen Arbeit? Was will ich aufzeigen? Warum ist dies wichtig? (Relevanz) Was will ich damit erreichen? (Ziel) - Formulierung mit Thema, Forschungsfrage, Ziel oder Berechtigung der Forschungsfrage (Ich untersuche den Zusammenhang zwischen Studienzufriedenheit und Prüfungsleistungen bei angehenden Absolventen [Thema]. Damit will ich herausfinden, wie sich die Zufriedenheit auf die Leistungen von Studierenden in dem letzten Studiensemester bei verschiedenen Prüfungsformen auswirkt [Forschungsfrage]. Daraus sollen konkrete Maßnahmen für Studierende und Absolventen abgeleitet werden, um die Prüfungsleistung zu optimieren [Ziel oder Berechtigung der Forschungsfrage]. Leitfragen für Forschungsfragen: für Gestaltung -> Welche Massnahmen können ergriffen werden, um ein Ziel zu erreichen?

Erstellung prototypischer Suchfunktion, die exemplarisch mehrere Quellen vereint und so durchsuchbar macht. Diese Ansätze bieten eine Grundlage für eine App-umfassende Suchfunktion.

1.4 Abgrenzung

Nur 3 Quellen, kein Fokus auf UI

Bei dem Begriff Suchfunktionen wird in dieser Arbeit davon ausgegangen, dass die Inhalte einer App durchsucht werden. Also beispielsweise die Texte, die in einer Seite vorkommen. Es soll nicht um die Suche durch dynamische Inhalte wie zum Beispiel bei YouTube gehen.

1.5 Aufbau der Arbeit

Nach einem einführenden ersten Kapitel ... Kapitel 2: Grundlagen des Projektumfeldes Kapitel 3: Vorbereitung und Analyse vergleichbarer existierender Produkte Kapitel 4: Konzeption der Suchfunktion und API Kapitel 5: Implementierung der API, Suchfunktion und Benutzeroberfläche Kapitel 6: Anforderungsabgleich, Bewertung und Evaluierung des Konzeptes der Suchfunktion Kapitel 7: Zusammenfassung der Arbeit und Ausblick

2 | Grundlagen

2.1 Suchfunktionen in mobilen Anwendungen

In der heutigen Zeit hat die Relevanz von Suchfunktionen in mobilen Anwendungen signifikant zugenommen, da eine Vielzahl von Nutzern regelmäßig mit ihren mobilen Geräten nach Produkten, Dienstleistungen und Informationen sucht. Aufgrund der umfangreichen Datenmengen, die in Apps bereitgestellt werden, erweist sich eine effektive Suchfunktion als unerlässlich, da das eigenständige Durchsuchen dieser Informationen häufig als zu zeitintensiv und ineffizient wahrgenommen wird. Unternehmen müssen daher sicherstellen, dass ihre mobilen Anwendungen ihren Kunden ein gutes Such- und Entdeckungserlebnis bieten, um die Erwartungen der Zielgruppe zu erfüllen [16].

Die Suche in mobilen Anwendungen unterscheidet sich von den Suchfunktionen in Desktop-Anwendungen. Im Jahr 2012 wurde festgestellt, dass Nutzer weniger Suchanfragen pro Sitzung stellen, wenn sie ein Mobiltelefon verwenden, als wenn sie einen Desktop-PC verwenden [17]. Dies deutet darauf hin, dass Nutzer die mobile Suche als größere Hürde empfinden. Trotzdem besteht ein starker Konsens darüber, dass mobile Anwendungen die gleichen Usability-Anforderungen erfüllen sollten wie Desktop-Anwendungen [18].

Bei der mobilen Suche muss die Balance gefunden werden, dem Nutzer die relevanten, gesuchten Inhalte zu liefern - ihn aber nicht zu überfordern, was dazu führen kann, dass die Suche verfeinert und wiederholt werden muss. Es ist sinnvoll, die Benutzerfreundlichkeit durch Funktionen wie Filter, Rechtschreibfehler-

toleranz, Vorschläge und frühere Suchanfragen zu verbessern. [16]

2.2 Flutter

Die Entwicklung plattformübergreifender Anwendungen ist für Unternehmen von entscheidender Bedeutung, um die breite Masse an Kunden zu erreichen. Dies ist jedoch mit einer gewissen Komplexität verbunden, da die Plattformen iOS und Android, auf denen die Anwendungen ausgeführt werden, unterschiedliche Funktionalitäten und Anforderungen aufweisen. Aktuell werden weltweit etwa 70 % der Mobiltelefone mit dem Betriebssystem Android und 29 % mit iOS betrieben [19]. Entwickler benötigen in der Regel unterschiedliche Fertigkeiten und müssen Apps aufgrund der unterschiedlichen Plattformen mehrfach erstellen [20].

Mit Flutter steht Entwicklern ein plattformübergreifendes Open-Source-Framework zur Verfügung, das die Erstellung hochperformanter mobiler Anwendungen aus einer einzigen Codebasis für die beiden Plattformen iOS und Android ermöglicht [21]. Die mobile SDK wurde 2015 durch Google angekündigt [22] und der erste Release wurde Ende 2018 publiziert [23].

Flutter-Apps werden in der Programmiersprache *Dart* verfasst, die ursprünglich die Funktion von JavaScript übernehmen sollte und daher eine der Programmiersprache Java ähnliche Syntax aufweist. Flutter erleichtert die Entwicklung durch Funktionen und zeitsparende Tools. So können Entwickler die 'just-in-time' Kompilierung verwenden, bei der der Computercode während der Programmausführung zur Laufzeit kompiliert wird. Darüber hinaus ermöglicht die 'Hot-Reload' Funktion den Entwicklern, Benutzeroberflächen zu gestalten oder Features hinzuzufügen, ohne dass diese Änderungen lange neu geladen werden müssen. Dabei werden die aktualisierten Quelldaten in die laufende Dart Virtual Machine eingefügt, die die betroffenen Klassen aktualisiert und den Widget-Tree automatisch neu baut. [20]

BMW hatte in der nativen Entwicklung früherer Anwendungen Schwierigkeiten, da die Entwicklungsprozesse als zu aufwendig erachtet wurden. Daher wurde bei

der Konzeption einer neuen Anwendung von Beginn an die Entscheidung getroffen, Flutter als Entwicklungsframework zu verwenden. Diese Wahl ermöglicht es, die Vorteile der SDK zu nutzen und eine plattformübergreifende Entwicklung zu realisieren. [24]

2.3 Die MyBMW-App

Im Jahr 2020 hat die BMW Group die MyBMW- bzw. Mini-App veröffentlicht, die den Kunden einen digitalen Zugang zu ihrem Fahrzeug ermöglicht. Die Entwicklung der App basiert auf dem Feedback und den Erkenntnissen aus dem Nutzerverhalten der Vorgängergenerationen, der BMW i Remote App und der BMW Connected App. [1] Derzeit nutzen mehr als 13 Millionen Nutzer die App. Sie wird etwa fünf Mal im Jahr durch Updates aktualisiert [3].

Für die Entwicklung der App wurde Flutter verwendet. Auf dieser Grundlage ist es möglich, die Applikation sowohl für Android und iOS als auch für die verschiedenen Skins und Regionen mit der gleichen Code-Basis zu entwickeln und bauen. Die Skins repräsentieren die Marken BMW, BMW M, Mini und Toyota. Darüber hinaus existieren spezifische Versionen für vorgegebene Regionen, wie Nordamerika oder Südkorea. Daraus resultieren ca. 30 Apps, die den Kunden in den Apps Stores angeboten werden. Das Team von ca. 250 Entwicklern ist eines der größten Flutter-Entwicklungsteams überhaupt. [24]

Die Applikation stellt dem Kunden eine universelle Schnittstelle zum Fahrzeug bereit. Mittels dieser Schnittstelle können Remote-Funktionen ausgeführt werden, womit der Fahrzeug- oder Ladestatus, die Reichweite sowie die Türen und Fenster des Autos aus der Ferne über das Mobiltelefon überprüft werden können. Die Funktionsweise ist auf Fahrzeuge ab dem Baujahr 2014 optimiert und kann sich abhängig von der Fahrzeugausstattung sowie länderspezifischen Vorgaben unterscheiden. [11]

Nach erfolgter Anmeldung mit der BMW-ID ermöglicht die App den Zugriff auf das Fahrzeug, BMW-Services und -Store. Die App ist in mehrere Unterseiten,

so genannte Tabs, unterteilt. Im Fahrzeug-Tab erhält der Kunde einen Überblick über den aktuellen Zustand seines Autos, also Fahrzeugstatus und Remote-Funktionen. Im Karten-Tab können Ziele zur Navigation gesucht und ausgewählt werden. Der BMW-Services- und Store-Tab bietet den Kunden direkten Zugriff auf Updates und Finanzdienstleistungen und ermöglicht den Kontakt zu Service Partnern. Im Profil-Tab können persönliche Einstellungen vorgenommen werden. [25]

Ein besonderes Feature ist das Remote Software Upgrade (RSU), mit dem Updates für die Fahrzeugsoftware direkt 'over-the-air' ins Fahrzeug oder zunächst in die MyBMW-App und dann auf das Auto geladen werden. [26]

todo: hier Bilder?

2.4 API-Entwicklung

APIs sind entscheidend für die nahtlose Kommunikation zwischen Softwarekomponenten und -diensten. Sie sind essenziell für die Verbindung verschiedener Systeme und ermöglichen die Nutzung von Diensten, Daten und Funktionalitäten Dritter. [27]

Mit APIs kann auf unterschiedliche Komponenten zugegriffen werden, darunter Hardwarekomponenten, Datenbanken, einzelne Programmfunktionen oder auch Oberflächen. [28]

Der Begriff API ist vielseitig interpretierbar und kann sich auf eine rein technische Schnittstellenbeschreibung, ein Kommunikationsmittel für Software oder auch um ein digitales Produkt beziehen. In allen Fällen sind APIs kein separates Softwaresystem, sondern lediglich eine Kommunikationsschnittstelle, die der Interaktion mit der Software dient. [29]

Darüber hinaus wird zwischen funktions-, datei-, objekt- und protokollorientierten APIs differenziert. [28]

Die Eigenschaften von APIs lassen sich wie folgt zusammenfassen: Modularität,

das heißt die Zusammensetzung eines Services aus mehreren separaten Services, Interoperabilität, also die Existenz von Standards in der Kommunikation zwischen Diensten, und die Kapselung, bei der die Programmlogik und die Datenbasis des Services verborgen bleiben. [29]

Die Modularisierung von Software führt zur Trennung einzelner Programmteile, die spezifische Funktionen erfüllen, und zur Trennung vom Rest der Applikation. Die Kommunikation erfolgt über eine genau definierte Schnittstelle, was zu einer sauberen Gesamtstruktur innerhalb des Projektes führt. Dies kann besonders komplexe Programme vereinfachen. Die einzelnen Programmteile sind leichter wartbar und damit weniger fehleranfällig. [28]

Die Verwendung von APIs hat sich als signifikant effizienter und beschleunigender Faktor in der Entwicklung von Anwendungen und Software erwiesen. Die gemeinsame Nutzung von Daten ermöglicht die Freigabe spezifischer Informationen, während systeminterne Details verborgen bleiben, was zu einer Optimierung der Sicherheit und Vertraulichkeit beiträgt. [30]

Die Implementierung einer API ist ein wesentlicher Faktor für den Erfolg mobiler Applikationen. Instabilität und Fehleranfälligkeit der API können den Erfolg der Software erheblich beeinträchtigen. Eine gute API ist ein wesentlicher Faktor für den Erfolg einer App. Eine Auswertung von Google Play-Bewertungen hat ergeben, dass erfolgreiche Apps weniger fehleranfällige APIs verwenden [31].

3 | Vorbereitung und Analyse der Ausgangssituation

Im Vorfeld der eigentlichen Konzeption der Suchfunktion erfolgt zunächst eine umfassende Analyse der Ausgangssituation. Zunächst werden allgemeine Funktionen und Anforderungen zusammengefasst, die üblicherweise an eine moderne Suchfunktion gestellt werden. Im Anschluss daran werden vergleichbare Suchfunktionen in Automotive-Apps sowie in anderen Anwendungsbereichen analysiert und mit den Anforderungen abgeglichen. Abschließend wird der Ist-Zustand der vorhandenen Daten und Datenquellen betrachtet.

3.1 Allgemeine Funktionen und Anforderungen an eine Suchfunktion

Bei der Entwicklung von Suchfunktionen in mobilen Anwendungen sollten bewährte Verfahren, auch bekannt als 'Best Practices', in die Konzeption einfließen. Oftmals verfügen Suchfunktionen über einen Übergangsbildschirm, der vergangene Suchanfragen, trendige Suchbegriffe oder vordefinierte Kategorien anbietet. Eine Filtermöglichkeit ist besonders sinnvoll, wenn relevante Daten vorhanden sind. Bei der Eingabe von Suchbegriffen wird häufig eine Autovervollständigung angeboten, die Vorschläge für mögliche Suchanfragen liefert. Dabei werden häufig auch Schreibfehler toleriert, um die Benutzerfreundlichkeit zu erhöhen. In den Suchergebnissen können die eingegebenen Suchwörter durch eine Markierung

3.2. RECHERCHE UND ANALYSE VERGLEICHBARER SUCHFUNKTIONEN

hervorgehoben werden, was die Übersichtlichkeit verbessert. Die Integration von Künstlicher Intelligenz kann in bestimmten Anwendungsfällen von Suchfunktionen von Vorteil sein, beispielsweise zur Schaffung eines personalisierten Einkaufserlebnisses. Letztlich ist es entscheidend, dass die Erwartungen der Nutzer an die Geschwindigkeit der Suchfunktion erfüllt werden, um eine positive Benutzererfahrung zu gewährleisten.[16]

Um die Suchfunktion zu starten, kann die Suchleiste entweder durch das Anklicken eines Suchsymbols aufgerufen oder direkt angezeigt werden. Wenn die Suche in einer Anwendung im Vordergrund steht, ist es empfehlenswert, die Suchleiste sofort anzuzeigen. Ist die Suche hingegen optional, kann ein kontextbezogenes Suchsymbol verwendet werden. [32]

Ein weiterer Trend, der sich bei den Suchfunktionen abzeichnet, sind Sprachfunktionen. So wurden 2016 20 % der Google-Suchanfragen über die Sprachfunktion getätigt [33]. Laut einer Studie im Jahr 2018 tätigen die Handynutzer mit 27 % die meisten Suchanfragen mit Sprachfunktion bzw. Sprachbefehlen, gefolgt von Tablet- und PC-Nutzern. Es wurde festgestellt, dass jüngere Nutzer öfter Sprachsuche oder Sprachbefehl-Tools verwendet haben. [34]

3.2 Recherche und Analyse vergleichbarer Suchfunktionen

Im Folgenden werden zunächst Apps von Automobilherstellern, die als direkte Wettbewerber der MyBMW-App angesehen werden können, näher betrachtet. Dabei wird ein besonderer Fokus auf eventuell vorhandene Suchfunktionen gelegt. Anschließend erfolgt eine allgemeine Analyse von Suchfunktionen in Apps, die einen Überblick über typische Konzepte und Funktionen liefert.

3.2. RECHERCHE UND ANALYSE VERGLEICHBARER SUCHFUNKTIONEN

3.2.1 Suchfunktionen in Automotive Apps

Eine Analyse der von den Automarken Mercedes, Audi, Tesla, Volkswagen und Volvo angebotenen Apps zeigt, dass sie mit der MyBMW App vergleichbar sind. Die Untersuchung der Funktionalitäten der Wettbewerber-Apps ergibt, dass die Grundfunktionen ähnlich sind. Dazu gehören Remote-Funktionen, mit denen z.B. das aktuelle Klima eingesehen und gesteuert werden kann, sowie die Möglichkeit, Fahrzeuge über das Handy zu entriegeln und den Fahrzeugstatus abzurufen. Zusätzlich haben Kunden die Möglichkeit, den aktuellen Standort des Autos zu überprüfen und Routen zu planen. Die Analyse der im App Store sichtbaren Funktionen ergab jedoch, dass keine der untersuchten Apps über eine Suchfunktion verfügt, die eine Suche nach Stichwörtern ermöglicht. [35],[36],[37],[38],[39]

todo: diese quellen sind eigene quellen -> in den anhang?

3.2.2 Suchfunktionen in Apps anderer Bereiche

ToDo: hier klären, wie diese "eigenen" Aufnahmen zitiert werden *ToDo: hier Bilder einfügen!*

Um einen Überblick über typische Suchfunktionen in mobilen Anwendungen zu erlangen, werden im Folgenden die Suchfunktionen von Apps aus verschiedenen Anwendungsbereichen analysiert. Ziel dieser Analyse ist die Entwicklung eines Verständnis für die Konzepte und Funktionen von In-App-Suchfunktionen.

Die Suchfunktion der 'Einstellungen'-App von Apple für iOS ermöglicht die Suche nach spezifischen Begriffen, die in das am Seitenanfang befindliche Suchfeld eingegeben werden. Hierbei durchsucht die App sämtliche Inhalte der Einstellungen. In der Version der Softwarestufe iOS 17 werden während der Eingabe Suchergebnisse angezeigt, ohne dass eine explizite Bestätigung der Eingabe durch den Nutzer erforderlich ist. Die Suche generiert eine Liste mit treffenden Ergebnissen, wobei der Titel des zugehörigen Feldes innerhalb der App, ein zugehöriges Symbol und teilweise der Pfad angegeben werden. Durch Anklicken eines Ergebnisses

3.2. RECHERCHE UND ANALYSE VERGLEICHBARER SUCHFUNKTIONEN

wird die entsprechende Seite geöffnet bzw. der gesamte Pfad, wobei das zugehörige Feld kurzzeitig grau hervorgehoben wird. Navigiert man zurück, gelangt man, wenn im Pfad vorhanden, auf eine Oberseite, bevor man wieder auf das Suchfenster zurückkommt. [40]

Ab iOS 18 werden vor der Eingabe in das Suchfeld Such-Vorschläge durch große Symbole und Text angezeigt und darunter der Verlauf der aufgerufenen Suchergebnisse. [41]

Die 'Einstellungen'-App von Android kann ebenfalls nach Begriffen durchsucht werden. Nach Anklicken des Such-Icons wird die Suchseite geöffnet, in der vergangene Suchen und allgemeine Suchvorschläge aufgelistet sind. Den Vorschlägen geht ein Rauten-Symbol vor, wodurch nach Themen und nicht nach konkreten Begriffen gesucht werden kann, wie etwa "beliebt". Während der Eingabe in das Suchfeld werden dann anstelle dessen die Suchergebnisse für die aktuelle Eingabe angezeigt. Diese sind in beliebte und restliche Ergebnisse aufgeteilt, die jeweils in ihre zugehörigen Kategorien gruppiert sind. Durch Anklicken eines Ergebnisses wird die entsprechende Seite innerhalb der 'Einstellungen'-App geöffnet. Durch Betätigen der 'Zurück'-Taste des Handys, gelangt man in die Suchfunktion. Durch Betätigung des 'Zurück'-Button in der Leiste oben, geht man dem der Seite zugehörigen Pfad entlang zurück. [42]

Als weiteres Beispiel bietet die App 'Instagram' für ihren Einstellungen-Bereich eine Suchfunktion an, mit der die Inhalte dieses Bereiches durchsucht werden können. Bei der Eingabe von Suchbegriffen werden die Suchergebnisse in verschiedenen Gruppen untereinander bereits während des Tippens, ohne explizite Bestätigung, angezeigt. Diese Gruppierung entspricht der Kategorisierung, die in den Einstellungen verwendet wird. Beim Anklicken eines Ergebnisses wird direkt die entsprechende Unterseite geöffnet. Es sei jedoch angemerkt, dass nicht alle in den Einstellungen dargestellten Inhalte durch die Suche auffindbar sind. [43]

Die untersuchten Apps setzen viele der zuvor erhobenen Anforderungen um. Häufig gibt es einen Übergangsbildschirm, der Vorschläge und vergangene Suchanfragen anzeigt. Während des Tippens werden jedoch oft nur die Ergebnisse für den aktuellen Stand des Suchbegriffs angezeigt, ohne dass Vorschläge für eine au-

tomatische Vervollständigung angeboten werden. Zudem wurden die Suchergebnisse auf den Seiten nur teilweise explizit hervorgehoben. Es besteht auch keine Konsistenz darin, ob die Suchfunktion über eine Suchleiste oder ein Suchsymbol aufgerufen wird. Insgesamt zeigen sich sowohl Übereinstimmungen als auch Unterschiede in den Implementierungen der Suchfunktionen.

3.3 Ist-Analyse der vorhandenen Daten und Datenquellen

Um eine umfassende Suche durch verschiedene Bereiche der MyBMW-App zu ermöglichen, werden die Daten aus verschiedenen Quellen angebunden.

Die Texte, die in der MyBMW-App verwendet werden, beispielsweise in Buttons, Menüpunkten oder anderen Anzeigekomponenten, sind in sogenannten 'String Files' gespeichert. Diese Dateien ermöglichen es, die Anwendung in verschiedenen Sprachen darzustellen, indem die Texte in der jeweiligen Sprache aus der entsprechenden Datei geladen werden. Jeder Text ist mit einer eindeutigen ID versehen, die es ermöglicht, die Texte den verschiedenen Features innerhalb der App zuzuordnen. Jedes Feature hat dabei einen zugehörigen Pfad, der dann auf die entsprechende Seite des Features führt. Im Rahmen der Implementierung der Suchfunktion sollen exemplarisch zwei Dienste der MyBMW-App an die Suchfunktion angebunden werden. Diese Daten werden im Weiteren als '*String-File*-Daten' bezeichnet.

Zusätzlich wird eine dynamische Quelle integriert, die über die App bereitgestellt und über den 'Context' geladen werden kann. Diese Quelle umfasst die Artikel, die auf der Explore-Seite der MyBMW-App dargestellt werden, und bietet somit aktuelle Inhalte für die Suche. Diese Daten werden als *Kontext-Daten* bezeichnet.

Darüber hinaus werden lokale Daten in Form einer JSON-Datei in die Suche integriert. In dieser prototypischen Umsetzung enthält die JSON-Datei Datensätze zu 'My Highlights', die besondere Änderungen bei RSU für die Kunden hervorheben. Diese Daten werden als *Lokale-Daten* referenziert.

4 | Konzeption

todo: Forschungsfrage konkretisieren hier

4.1 Anforderungsanalyse der Suchfunktion

todo: sollen hier auch nicht-umgesetzte sachen stehen? wenn ja hier nochmal mehr strukturieren und priorisieren - in must-haves und nice-to-haves?

Im 'Requirements Engineering', also der Anforderungsanalyse, werden die Bedürfnisse der Benutzer, Auftraggeber und anderen Interessensgruppen analysiert, um eine geeignete Lösung für ein Produkt zu entwickeln. Dabei wird eine Übereinkunft über die Funktionen des Systems, die Systemgrenzen und die Benutzerschnittstellen getroffen. Diese Informationen helfen auch den Entwickler, die Anforderungen besser zu verstehen. Dabei wird zwischen funktionalen und nicht-funktionalen Anforderungen unterschieden. Die funktionalen Anforderungen beziehen sich auf Aspekte, die die Funktionalität des Produkts erklären. Nicht-funktionale Anforderungen beziehen sich auf die erforderlichen Qualitätsaspekte und Rahmenbedingungen, d.h. Elemente, die für das Nutzererlebnis eine wichtige Rolle spielen. In der Regel werden die Inhalte u.a. durch Interviews mit Stakeholdern oder Nutzerbefragungen ermittelt. [44]

Da es sich bei der Entwicklung der Suchfunktion um eine prototypische Umsetzung handelt, wurde hierfür keine Nutzerbefragung zur Ermittlung der Anforderungen durchgeführt. Vermutliche Nutzerbedürfnisse wurden viel mehr antizipiert.

Nicht alle an die Suchfunktion gestellten Anforderungen werden aufgrund des begrenzten Zeitrahmens der Durchführung der Bachelorarbeit umgesetzt.

4.1.1 Systematische Erfassung der Anforderungen

Die funktionalen Anforderungen an die zu entwickelnde Suchfunktion sind wie folgt definiert:

- Die Suchfunktion soll der bestehenden 'Explore'-Seite hinzugefügt werden
- Nach dem Anklicken eines Suchsymbols soll der Benutzer einen einzelnen Suchbegriff in ein vorgegebenes Suchfeld eingeben können
- Vor der Eingabe einer Suchanfrage in die Suchleiste sollen dem Nutzer allgemeine Vorschläge und frühere Suchanfragen angezeigt werden
- Während des Tippens in die Suchleiste sollen Suchvorschläge gegeben werden, die das aktuelle Suchwort ergänzen - diese Anforderung wurde im Rahmen der Bachelorarbeit nicht umgesetzt
- Bei der Bestätigung eines Begriffs werden die zur Suche verfügbaren Quellen über die Schnittstelle durchsucht und anschließend jene Ergebnisse angezeigt, die den Suchbegriff beinhalten
- Mithilfe eines Filters können die zu durchsuchenden Quellen bestimmt werden - diese Anforderung wurde im Rahmen der Bachelorarbeit nicht umgesetzt
- Kleine Rechtschreibfehler oder nicht exakte Übereinstimmungen zwischen Suchbegriff und durchsuchten Daten sollen toleriert werden
- Die Resultate der Suche werden in reduzierter Form untereinander angezeigt
- Zu jedem Ergebnis werden ein Titel und Ausschnitte eines längeren Textes dargestellt. Durch Anklicken wird die dazugehörige Seite geöffnet, während beim Verlassen der Seite wieder die Ergebnisse der Suchfunktion angezeigt werden

Die nicht-funktionalen Anforderungen umfassen:

- Die Suchfunktion soll eine gute Performance aufweisen, d.h. die Suchergebnisse sollen schnell angezeigt werden
- Die Suchfunktion soll benutzerfreundlich sein. Bei der prototypischen Umsetzung der Suchfunktion im Rahmen dieser Bachelorarbeit steht das UX-Design jedoch nicht im Vordergrund

Neben der Analyse der Anforderungen an die Suchfunktion, werden auch die technischen Rahmenbedingungen analysiert. So sollen die technischen Möglichkeiten und Grenzen untersucht, geeignete Frameworks gefunden und Schnittstellen geprüft werden. Die Suchfunktion soll in die bestehende App-Umgebung der MyBMW-App integriert werden. Da die MyBMW-App mit Flutter entwickelt wurde, basiert die grundlegende technische Infrastruktur der Suchfunktion auf der Programmiersprache Dart.

Flutter stellt u.a. die Klasse `SearchDelegate` zur Verfügung, die eine Suchseite erzeugt, die u.a. Vorschläge und Ergebnisse anzeigt und im Code Zugriff auf das aktuelle Suchwort, also `query` ermöglicht [45]. Diese Klasse vereint die genannten Anforderungen. Die Klasse `SearchBar` stellt zum Vergleich nur das Eingabefeld für die Suchbegriffe zur Verfügung [46].

Die Datenquellen für die Suchfunktion sind die bereits erwähnten *'String Files'*, *Kontext-* und *Lokale-Daten*.

Um letztere Daten aus der lokalen Datei, also der JSON-Datei, lesen zu können, müssen diese zunächst in ein für Dart lesbares Format gebracht werden. Dies kann mit dem Dart-Package `json_serializable` realisiert werden. Damit können Daten zwischen dem JSON-Format und der gewünschten und durch den Code definierten Struktur konvertiert werden. [47]

In allen Quellen werden für die Suchfunktion unterschiedliche Strings nach Übereinstimmungen durchsucht. Um kleine Abweichungen zwischen Such- und Vergleichswort zu tolerieren, kann das Dart Package *'Fuzzy'* verwendet werden. Mit diesem wird ein für einen String-Vergleich ein sogenannter Fuzzy-Score erzeugt, der angibt, wie groß die Übereinstimmung ist. [48]

4.1.2 Konzeptentwurf

Auf Basis der systematischen Erfassung der Anforderungen an die Suchfunktion wird nun das Konzept für die Suchfunktion konkretisiert. Die Konzeption und der Aufbau der API, an die die Suchanfragen gestellt wird und die die verschiedenen Quellen zusammenführt, werden in Kapitel 4.2 behandelt. Daher wird an dieser Stelle nicht näher darauf eingegangen. Grob betrachtet schickt die Suchfunktion ein Suchbegriff an die API und bekommt von dieser dann die Ergebnisse für die Anfrage zurück. (Siehe Abbildung 4.1)

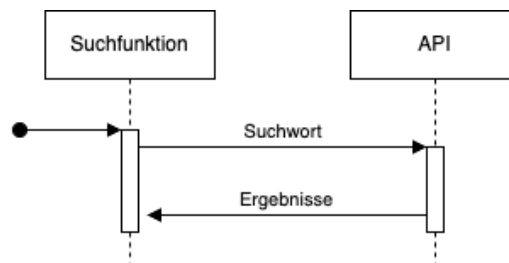


Abbildung 4.1: BeschriftungToBeDone

Der Ablauf der Suchfunktion aus Benutzersicht soll wie folgt ablaufen: Durch Anklicken des Lupensymbols auf der 'Explore'-Seite, öffnet sich die Seite der Suchfunktion. Über die Leiste am oberen Rand dieser Seite können folgende Funktionen ausgeführt werden: Es besteht die Möglichkeit zurück zur vorherigen Seite zu navigieren, einen Suchbegriff einzugeben oder den bisher eingegebenen zu löschen. Unterhalb der Leiste befinden sich Suchvorschläge, die von der Funktionsseite gegeben werden und die zuletzt durchgeführten Suchanfragen. Dabei muss entschieden werden, ob diese Funktionen durch Symbole dargestellt oder mit Text angeschrieben werden. Auch muss entschieden werden, ob die Suchvorschläge dynamisch oder statisch sind.

Eine Suchanfrage wird durchgeführt, indem ein Suchbegriff in das Suchfeld eingegeben und dann mit der Eingabetaste bestätigt wird. Während des Tippens wird die Seite mit Suchverlauf und den Vorschlägen weiter angezeigt. Der Suchbegriff wird an die API geschickt, die diese Anfrage bearbeitet. Anschließend werden die durch die API gefundenen Ergebnisse nach Relevanz sortiert aufgelistet. Diese

können je nach Quelle und damit Inhalt unterschiedlich aufgebaut sein.

Bei den Ergebnissen aus den '*String-Files*' müssen einige Punkte bedacht werden. So muss geklärt werden, ob nur die Datei für die aktuelle Sprache, in der die App gerade ist, oder alle vorhandenen Sprachen, und damit alle Dateien durchsucht werden. So könnten Nutzer beispielsweise mit einer deutschen App-Einstellung nach einem englischen Begriff suchen und dafür ein Ergebnis erwarten. Andererseits könnte besonders durch die Toleranz von Rechtschreibfehlern Begriffe aus anderen Sprachen, die ähnlich sind, als Treffer angesehen werden können. Zudem muss geklärt werden, wie die Ergebnisse aus den '*String-Files*' dargestellt werden. Da alle Ergebnisse eines Dienstes auf die gleiche Seite verweisen, ist festzusetzen, ob Ergebnisse pro Dienst gruppiert oder einzeln angezeigt werden.

Wird ein Ergebnis angeklickt, öffnet sich die entsprechende Seite der App. Navigiert man zurück, gelangt man wieder auf die Ergebnisseite. (Siehe Abbildung 4.2) Dieser Aufbau ähnelt der Struktur, die von der Flutter-Klasse `SearchDelegate` vorgegeben wird, in der Platz für Suchvorschläge und -ergebnisse vorgesehen ist [45].

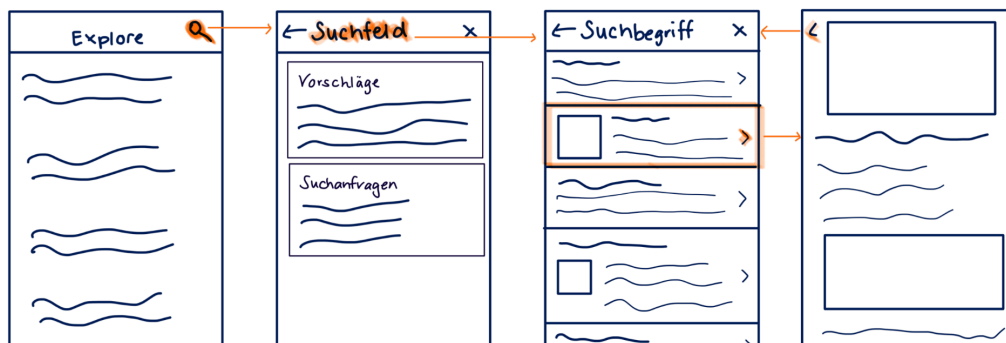


Abbildung 4.2: Beschriftung ToBeDone

Der Suchbegriff wird durch Ausführen der Suche an die API übergeben, die nach der Verarbeitung die passenden Ergebnisse aus den verschiedenen Quellen zurückgibt. Diese werden dann untereinander aufgelistet, wobei die verschiedenen Ergebnisse deutlich voneinander unterschieden werden. Die eigentliche Suchfunktion liegt also in der API, die im folgenden Kapitel konzipiert wird.

4.2 Konzeption der API

Nachdem im vorherigen Kapitel das Bedienkonzept erarbeitet worden ist, soll nun die bisher nicht näher betrachtete API konzipiert werden. Die hier zu entwickelnde API wird als Schnittstelle in das existierende App-Umfeld hinzugefügt.

4.2.1 Schnittstellen-Design

Der Zugriff auf die API erfolgt über einen lokalen Aufruf innerhalb der Suchfunktion. Bei Bestätigung der Eingabe in das Suchfeld wird der Suchbegriff an die Schnittstelle übergeben. Als Rückgabewert werden dann die Ergebnisse der Suche erwartet.

Um die Daten auslesen zu können, muss die API auf verschiedene Datenquellen zugreifen. Für die prototypische Umsetzung sind die *'String-Files'*-, *Kontext-* und *Lokale-Daten*. Die detaillierte Betrachtung des Zugriffs auf diese Daten erfolgt in Kapitel 5.1. Um diese zu vereinheitlichen werden alle Daten in Klassen der gleichen Oberklasse geladen. Diese Oberklasse wird als `Searchable` bezeichnet - die Klasse kann also von der API durchsucht werden. Dabei soll von jedem `Searchable`-Element ein *Fuzzy-Score* ermittelt werden, der aussagt ob der Suchbegriff in dem Element vorkommt und damit zu den Ergebnissen hinzugefügt wird. Diese Ergebnisse werden als `Findable` geladen. Dabei wird der ermittelte *Fuzzy-Score* gespeichert. Jedes `Findable` kann außerdem in einer reduzierten Ansicht als Ergebnis angezeigt werden. (Siehe Abbildung 4.3)

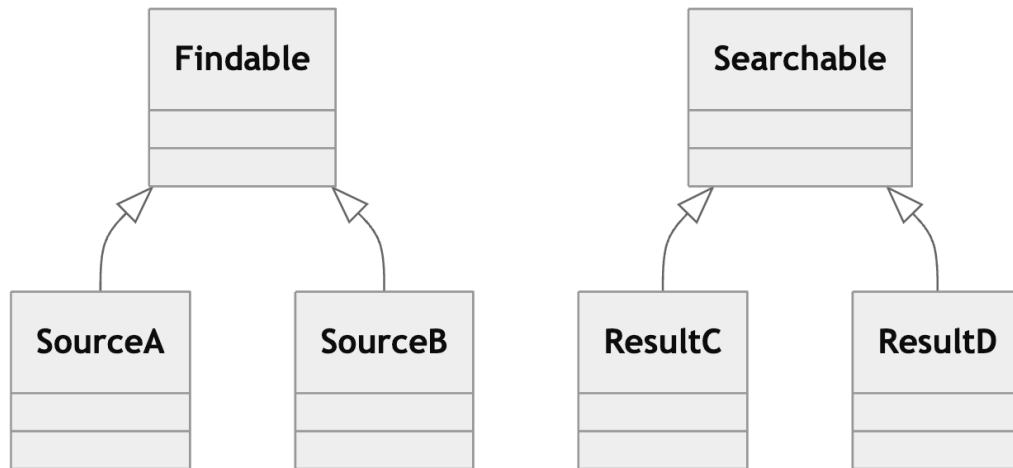


Abbildung 4.3: BeschriftungToBeDone

4.2.2 Architektur-Entwurf

Nachdem im vorherigen Kapitel die Endpunkte der zu entwickelnden API aufgezeigt wurden, wird nun konzipiert, wie die Ergebnisse ermittelt werden.

Für jede Quelle, die an die Suchfunktion angebunden werden soll, wird ein Provider implementiert. Diese erweitern eine Oberklasse `SearchProvider`, die eine Methode hat mit der die jeweilige Quelle nach einem Suchwort durchsucht werden kann. In diesen Providern wird zunächst auf die Daten der Quelle zugegriffen und diese dann nach dem Suchwort durchsucht. Wie das für die jeweiligen Quellen aussieht, wird in Kapitel 5.1 näher betrachtet. (Siehe Abbildung 4.4)

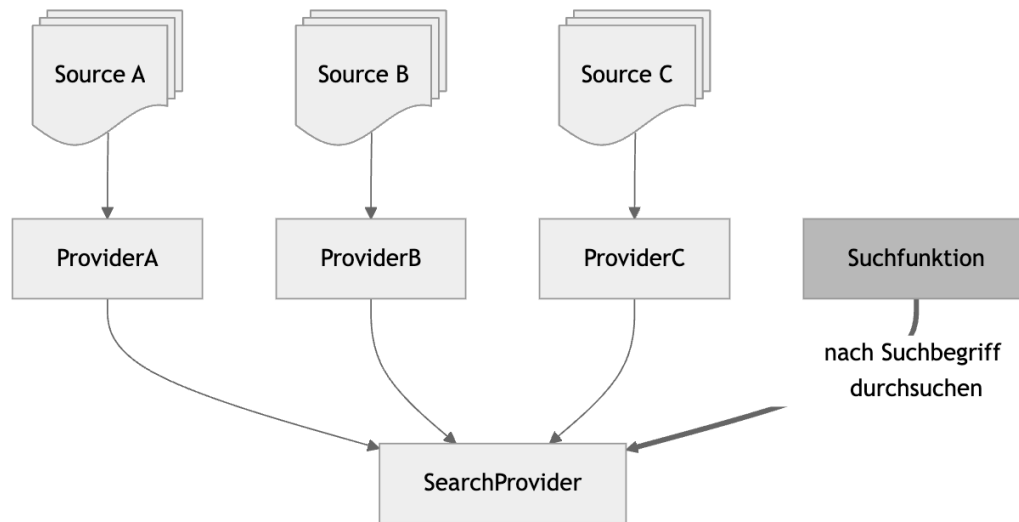


Abbildung 4.4: BeschriftungToBeDone

Mit diesem Aufbau können in Zukunft weitere Quellen auf eine einheitliche und simple Art an die Suchfunktion hinzugefügt werden. Damit ist diese Konzipierung skalierbar.

5 | Implementierung der API und Suchfunktion

5.1 Implementierung der API

5.1.1 Vertiefte Hintergründe der API

5.2 Implementierung des Suchfunktions-Prototyps

5.3 Benutzeroberfläche

Mehr in Quelle [49] ab Seite 122 ...

Quelle [44]: User Experience Definition: "User Experience (UX): Hier steht das Gesamterlebnis der Benutzer bei der Verwendung von Produkten, Systemen und Diensten im Fokus. Nebst den funktionalen Aspekten werden dabei vermehrt auch emotionale und ästhetische Faktoren berücksichtigt. So liegt neben geschäftlichen Anwendungen ein Schwerpunkt des Gebietes auf Lösungen und Produkten im Consumer-Bereich, also etwa auf E-Services, Smartphone Apps und digitalen Geräten, aber auch für Spiele und Anwendungen im Unterhaltungsbereich spielen die genannten Faktoren eine entscheidende Rolle für den Produkterfolg. Aufgrund der umfassenderen Betrachtungsweise hat der Begriff UX sich in viel-

fältiger Weise durchgesetzt und löst immer mehr auch die Bezeichnung Usability als Qualitätsbegriff ab." + Mehr auf Seite 12 + Seite 14 Tabelle

Quelle [16]:

6 | Evaluierung

6.1 Anforderungsabgleich

6.2 Bewertung des Konzepts anhand von Beispielszenarien

6.3 Evaluierung der Suchfunktion

6.3.1 Evaluierung der Suchergebnisse

6.3.2 Nutzerbefragung zur Suchfunktion

Infos zur Nutzerbefragung: [44]: - Benutzerbefragungen mit Fragebögen sind wichtige Methode, um Antworten von einer größeren Anzahl Personen zu erhalten (sic) - Fragebögen werden zur Analyse von Benutzern und Kontext, und zur Beurteilung eines Systems (Evaluation) eingesetzt - quantitative Studie, weil Antworten mit Fragebogen zählbar sind - Für Erkenntnisse in Usability und UX oke, aber eig immer hinterfragen, ob Usability-Tests oder Experten-Reviews nicht besser wäre - wichtig: Vergleichbarkeit: Ergebnisse können verdichtet, statistisch ausgewertet und miteinander verglichen werden - Gut für Beurteilung von verschiedenen Systemen -

Mehr Infos: [50] Seite 125 - 130

6.4 Vergleich mit Daten

Aufnahme mit “Referenz?”

kursiv: * auf beiden Seiten des Textes -> *kursiv* fett: ** -> **fett** kursiv und fett: *** -> ***fett und kursiv***

Aenean nec dapibus in mL/min⁻¹. Mathematical formula can be inserted using Latex:

$$(1) f(x) = ax^3 + bx^2 + cx + d$$

Die Tabelle 6.1 zeigt uns wie man eine Tabelle hinzufügt.

- erstes Element der Liste
- zweites Element der Liste
- drittes Element der Liste

1. erstes Element
2. zweites Element
3. drittes Element

Syntaxhervorhebung in Codeblöcken erreicht man mit drei “” Zeichen vor und nach dem Codeblock.

```
mood = 'happy'
if mood == 'happy':
    print("I am a happy robot")
```

Die Tabelle 6.1 zeigt uns wie man eine Tabelle hinzufügt. Integer tincidunt sed nisl eget pellentesque. Mauris eleifend, nisl non lobortis fringilla, sapien eros aliquet orci, vitae pretium massa neque eu turpis. Pellentesque tincidunt aliquet volutpat. Ut ornare dui id ex sodales laoreet.

Tabelle 6.1: Das ist die Tabellenbeschriftung. Suspendisse blandit dolor sed tellus venenatis, venenatis fringilla turpis pretium.

Spalte 1	Spalte 2	Spalte 3
Zeile 1	0.1	0.2
Zeile 2	0.3	0.3
Zeile 3	0.4	0.4
Zeile 4	0.5	0.6

7 | Fazit

7.1 Zusammenfassung der Arbeit

Zusammenfassend

7.2 Ausblick

Es gibt mehrere mögliche Richtungen, um diese Arbeit zu erweitern.

Mehr Funktionen: - Suchwort im Text markieren - Filter: Quellen ausschließen / wählen - Suche mit kontextuellen Daten: - [51] -> Such data can provide more meaningful search results by augmenting searches with real-world information related to users' profiles and behavioral patterns. Information about users such as location, how they interact with the mobile device, or what's occurring in the surrounding physical world is called contextual data. Such information is increasingly accessible via sensors embedded in smartphones leading to many new commercial usage scenarios — such as location-based services — and academic research on the topic. ... - Context Awareness: The term context, or context awareness, usually refers to a general class of systems that can sense a continuously changing physical environment and provide relevant services to users on this basis - Im mobilen Context: ■ signal coverage strength; ■ active connectivity methods; ■ environmental information, such as light; ■ current location; ■ activity time and time frame; ■ internal device information, such as power; and ■ the device's

motion and angle.

Anhang 1: Einige Extras

Füge Anhang 1 hier hinzu.

Anhang 2: Noch mehr Extras

Füge Anhang 2 hier hinzu.

Literatur

- [1] C. Koenig, "Your World. My BMW. Die neue App-Generation für BMW Kunden. Jetzt verfügbar in 30 europäischen Märkten, China und Korea." Accessed: Jan. 02, 2025. [Online]. Available: <https://www.press.bmwgroup.com/deutschland/article/detail/T0321612DE/your-world-my-bmw-die-neue-app-generation-fuer-bmw-kunden-jetzt-verfuegbar-in-30-europaeischen-maerkten-china-und-korea?language=de>
- [2] "BMW Geschichte." Accessed: Oct. 02, 2024. [Online]. Available: <https://www.bmwgroup.com/de/unternehmen/historie.html>
- [3] M. Tholund, "Umfangreiche Updates: My BMW App gibt Entscheidungshilfe für Umstieg vom Verbrenner auf E-Fahrzeug." Accessed: Sep. 30, 2024. [Online]. Available: <https://www.press.bmwgroup.com/deutschland/article/detail/T0444273DE/umfangreiche-updates:-my-bmw-app-gibt-entscheidungshilfe-fuer-umstieg-vom-verbrenner-auf-e-fahrzeug?language=de>
- [4] B. Group, "Jahresbericht 2024," BMW, 21.3.24. Accessed: Sep. 18, 2024. [Online]. Available: https://www.bmwgroup.com/content/dam/grpw/websites/bmwgroup_com/ir/downloads/de/2024/bericht/BMW-Group-Bericht-2023-de.pdf
- [5] "Automotive industry - European Commission." Accessed: Oct. 02, 2024. [Online]. Available: https://single-market-economy.ec.europa.eu/sectors/automotive-industry_en
- [6] P. M. de Miguel, C. De-Pablos-Heredero, J. L. Montes, and A. García, "Impact of Dynamic Capabilities on Customer Satisfaction through Digital Transformation in the Automotive Sector," *Sustainability*, vol. 14, no. 8, p. 4772, Jan. 2022, doi: 10.3390/su14084772.
- [7] strategy&, "Digital Auto Report 2023 How fast will the mobility ecosystem really transform?" Accessed: Oct. 02, 2024. [Online]. Available: <https://www.strategyand.pwc.com/de/en/industries/automotive/digital-auto-report2023/vol2/strategyand-digital-auto-report-2023-vol2.pdf>

-
- [8] P. Gao, H.-W. Kaas, D. Mohr, and D. Wee, "Der Umsatz der Autoindustrie kann bis 2030 auf 6,7 Billionen Dollar 2030 steigen - neue Mobilitätsangebote und Konnektivitätsdienste als Motor." Accessed: Oct. 02, 2024. [Online]. Available: <https://www.mckinsey.com/industries/automotive-and-assembly/our-insights/disruptive-trends-that-will-transform-the-auto-industry/de-DE>
- [9] G. Deryabina and N. Trubnikova, "The Impact of Digital Transformation in Automotive Industry on Changing Industry Business Model," in *IV International Scientific and Practical Conference*, in DEFIN-2021. New York, NY, USA: Association for Computing Machinery, Feb. 2022, pp. 1–7. doi: 10.1145/3487757.3490886.
- [10] D. P. F. Möller and R. E. Haas, "Mobile Apps for the Connected Car," in *Guide to Automotive Connectivity and Cybersecurity: Trends, Technologies, Innovations and Applications*, D. P. F. Möller and R. E. Haas, Eds., Cham: Springer International Publishing, 2019, pp. 379–438. doi: 10.1007/978-3-319-73512-2_7.
- [11] BMW Group, "Die Highlights der my BMW App im Überblick." Accessed: Oct. 02, 2024. [Online]. Available: <https://www.bmw.de/de/my-bmw-app/my-bmw-app.html>
- [12] "Measuring Digital Development - Facts and Figures 2023." Accessed: Sep. 18, 2024. [Online]. Available: https://www.itu.int/hub/publication/d-ind-ict_mdd-2023-1/
- [13] Y. Jeanrenaud, "Globale Ausgaben für mobile Apps steigen um zehn Prozent." Accessed: Oct. 02, 2024. [Online]. Available: <https://www.pocketpc.ch/magazin/news/vermishtes/globale-ausgaben-fuer-mobile-apps-steigen-um-zehn-prozent-92172/>
- [14] B. Martin, "The Global Mobile Report." Sep. 12, 2017. Accessed: Oct. 09, 2024. [Online]. Available: <file:///Users/Q502119/Downloads/The-Global-Mobile-Report-FINAL-ENG-WL3.pdf>
- [15] Y. Ma, X. Liu, M. Yu, Y. Liu, Q. Mei, and F. Feng, "Mash Droid: An Approach to Mobile-Oriented Dynamic Services Discovery and Composition by In-App Search," in *2015 IEEE International Conference on Web Services*, Jun. 2015, pp. 725–730. doi: 10.1109/ICWS.2015.102.
- [16] C. Dee, "Best practices for in-app mobile search UX design." Accessed: Oct. 09, 2024. [Online]. Available: <https://algolia.com/blog/ux/mobile-search-ux-best-practices/>
- [17] D. Komaki, T. Hara, and S. Nishio, "How Does Mobile Context Affect People's Web Search Behavior?: A Diary Study of Mobile Information Needs and Search Behaviors," in *2012 IEEE 26th International Conference on Advanced Information Networking and Applications*, Mar. 2012, pp. 245–252. doi: 10.1109/AINA.2012.134.
- [18] G. Getto, J. T. Labriola, and S. Flanagan, "The State of Mobile UX: Best Practices From Industry and Academia," in *2020 IEEE International Professional Communication Conference (ProComm)*, Jul. 2020, pp. 115–122. doi: 10.1109/ProComm48883.2020.00024.
- [19] "Smartphone Usage by Operating System 2025." Accessed: Feb. 04, 2025. [Online]. Available: <https://worldstats.com/smartphone-usage-by-operating-system-2025/>
-

-
- [20] A. Tashildar, N. Shah, R. Gala, T. Giri, and P. Chavhan, "Application Development using Flutter," *IRJMETs*, Aug. 2020.
 - [21] Flutter, "Flutter - Build apps for any screen." Accessed: Jan. 03, 2025. [Online]. Available: [//flutter.dev/](https://flutter.dev/)
 - [22] M. L. Napoli, *Beginning Flutter: A Hands On Guide to App Development*. John Wiley & Sons, 2019.
 - [23] "Flutter SDK archive." Accessed: Feb. 06, 2025. [Online]. Available: <https://docs.flutter.dev/release/archive>
 - [24] M. Tasior and C. Schmid, *Developers in Cars*. Accessed: Sep. 19, 2024. [Online]. Available: <https://open.spotify.com/episode/4SplAB7lua99Rn1d2WIMPO>
 - [25] BMW, Dir., *How-To: Erste Schritte mit der My BMW App*, (Sep. 19, 2024). Accessed: Oct. 07, 2024. [Online Video]. Available: https://www.youtube.com/watch?v=aErk_6cXfJ8
 - [26] T. Julich, "Ein Update für die Freude am Fahren. Neues Remote Software Upgrade für rund 2 Millionen BMW-Fahrzeuge weltweit." Accessed: Sep. 30, 2024. [Online]. Available: <https://www.press.bmwgroup.com/deutschland/article/detail/T0348312DE/ein-update-fuer-die-freude-am-fahren-neues-remote-software-upgrade-fuer-rund-2-millionen-bmw-fahrzeuge-weltweit?language=de>
 - [27] S. Selvaraj, *Mastering REST APIs: Boosting Your Web Development Journey with Advanced API Techniques*. Berkeley, CA: Apress, 2024. doi: 10.1007/979-8-8688-0309-3.
 - [28] S. Luber, "API-Entwicklung: Grundlagen, Eigenschaften und Best Practices." Accessed: Feb. 06, 2025. [Online]. Available: https://www.dev-insider.de/was-ist-eine-api-a-f47dbf39be4084d392d20f8a9145235d/?utm_source=chatgpt.com
 - [29] R. Frank, S. Strugholtz, and F. Meise, *Bausteine der digitalen Transformation: Wie APIs Unternehmen den Weg in die Programmable Economy ebnen*. Wiesbaden: Springer Fachmedien, 2021. doi: 10.1007/978-3-658-35107-6.
 - [30] IBM, "What Is an API (Application Programming Interface)?" Accessed: Nov. 18, 2024. [Online]. Available: <https://www.ibm.com/topics/api>
 - [31] M. Linares-Vásquez, G. Bavota, C. Bernal-Cárdenas, M. Di Penta, R. Oliveto, and D. Poshyvanyk, "API change and fault proneness: A threat to the success of Android apps," in *Proceedings of the 2013 9th Joint Meeting on Foundations of Software Engineering*, in ESEC/FSE 2013. New York, NY, USA: Association for Computing Machinery, Aug. 2013, pp. 477–487. doi: 10.1145/2491411.2491428.
 - [32] J. Holst, "E-Commerce Search Field Design and Its Implications – Baymard." Accessed: Feb. 04, 2025. [Online]. Available: <https://baymard.com/blog/search-field-design>
 - [33] "Google App Voice Searches." Accessed: Feb. 07, 2025. [Online]. Available: <https://www.thinkwithgoogle.com/consumer-insights/consumer-trends/google-app-voice-search/>
-

-
- [34] "Voice Search," global web index, 2018. Accessed: Feb. 07, 2025. [Online]. Available: <https://www.gwi.com/hubfs/Downloads/Voice-Search-report.pdf>
 - [35] "App Store Mercedes-Benz App." Accessed: Oct. 10, 2024. [Online]. Available: <https://apps.apple.com/de/app/mercedes-benz/id1487652920>
 - [36] "App Store myAudi App." Accessed: Oct. 10, 2024. [Online]. Available: <https://apps.apple.com/de/app/myaudi/id440464115>
 - [37] "App Store Tesla App." Accessed: Oct. 10, 2024. [Online]. Available: <https://apps.apple.com/de/app/tesla/id582007913>
 - [38] "App Store Volkswagen App." Accessed: Oct. 10, 2024. [Online]. Available: <https://apps.apple.com/de/app/volkswagen/id1517566572>
 - [39] "App Store Volvo Cars App." Accessed: Oct. 10, 2024. [Online]. Available: <https://apps.apple.com/de/app/volvo-cars/id439635293>
 - [40] *Einstellungen App iPhone*, (Oct. 10, 2024).
 - [41] *Einstellungen App iPhone 2*, (16.1.25).
 - [42] *Einstellungen App Android*.
 - [43] *Instagram Einstellungen*, (16.1.25).
 - [44] M. Richter and M. D. Flückiger, *Usability und UX kompakt*. in IT kompakt. Berlin, Heidelberg: Springer, 2016. doi: 10.1007/978-3-662-49828-6.
 - [45] "SearchDelegate class - material library - Dart API." Accessed: Nov. 20, 2024. [Online]. Available: <https://api.flutter.dev/flutter/material/SearchDelegate-class.html>
 - [46] Flutter, "SearchBar class - material library - Dart API." Accessed: Oct. 09, 2024. [Online]. Available: <https://api.flutter.dev/flutter/material/SearchBar-class.html>
 - [47] "Json_serializable | Dart package." Accessed: Nov. 07, 2024. [Online]. Available: https://pub.dev/packages/json_serializable
 - [48] "Fuzzy | Dart package." Accessed: Nov. 20, 2024. [Online]. Available: <https://pub.dev/packages/fuzzy>
 - [49] R. Nunkesser, *App-Entwicklung für Mobile und Desktop: Software Engineering mit .NET MAUI und Comet für iOS, Android, Windows und macOS*. Berlin, Heidelberg: Springer, 2023. doi: 10.1007/978-3-662-67476-5.
 - [50] A. Butz, A. Krüger, and S. T. Völkel, *Mensch-Maschine-Interaktion*. De Gruyter Oldenbourg, 2022. doi: 10.1515/9783110753325.
 - [51] E. Yndurain, D. Bernhardt, and C. Campo, "Augmenting Mobile Search Engines to Leverage Context Awareness," *IEEE Internet Computing*, vol. 16, no. 2, pp. 17–25, Mar. 2012, doi: 10.1109/MIC.2012.17.
-