

Projektdokumentation

Name: WalkHome
Team: 2
Gruppe: 23
Autoren: Yannik Huber, Tobias Szuminski
Dozenten: Gerhard Hartmann, Kristian Fischer
Mentoren: Ngoc-Anh Dang, Daniela Reschke
Modul: EIS Projekt Wintersemester 2016/17



Inhaltsverzeichnis

Projektdokumentation	1
Inhaltsverzeichnis	2
1. MCI - Durchführung der ISO 9241-210	4
1.1 Iteration1	4
1.1.1 Verstehen und Festlegen des Nutzungskontexts	4
1.1.1.1 Die Benutzer	4
1.1.1.2 Stakeholder	4
1.1.1.2.1 Stakeholdergruppen	5
1.1.1.2.2 Stakeholderanalyse	6
1.1.1.3 User Profiles	7
1.1.1.4 Personae und reale Personen	8
1.1.1.5 Use Cases:	9
1.1.2 Festlegen der Nutzungsanforderungen	12
1.1.2.1 Funktionale Anforderungen	13
1.1.2.2 Qualitative Anforderungen	15
1.1.2.3 Organisatorische Anforderungen	15
1.1.2.4 Befragung der Personen Anna Coen und Nicole Coen zur Interfacegestaltung	16
1.1.3 Erarbeitung von Gestaltungslösungen zur Erfüllung der Nutzungsanforderungen	19
1.1.4 Gestaltungslösungen aus der Benutzerperspektive evaluieren	22
1.1.4.1 Prototyp	22
1.1.4.2 Think Aloud	22
1.1.4.2.1 Think Aloud mit Regina Huber	22
1.1.4.2.2 Think Aloud mit Anna Coen	24
1.1.4.2.3 Think Aloud mit Anatol Walger	25
1.1.4.2.4 Think Aloud mit Marcel Huber	26
1.1.4.2.5 Zusammenfassung Think Aloud aller Probanden	27
1.2 Iteration 2	27
1.2.1 Festlegen der Nutzungsanforderungen	27
1.2.1.1 Hinzugefügte funktionale Anforderungen	28
1.2.2 Erarbeiten von Gestaltungslösungen zur Erfüllung der Nutzungsanforderungen	28

1.2.3 Evaluieren der Gestaltungslösungen anhand der Anforderungen	29
2. Systemdokumentation - WBA	32
2.1 Client-Server Architektur	33
2.2 Webserver	33
2.3 Ressourcentabelle	33
2.4 Datenstrukturen	34
2.5 Anwendungslogik	35
2.6 Schnittstellen	39
2.7 Probleme	41
3 Quellen	41

1. MCI - Durchführung der ISO 9241-210

1.1 Iteration1

1.1.1 Verstehen und Festlegen des Nutzungskontexts

1.1.1.1 Die Benutzer

Laut Bundeskriminalamt sind die Opfer von Vergewaltigung meist Jugendliche(14-18 Jahre), Heranwachsende(18-21 Jahre) und Erwachsene(unter 60 Jahre). Bezogen auf Körperverletzungsdelikte sind es in 70 % der Fälle Erwachsene zwischen 21 und 60 Jahren. Unsere Anwendung richtet sich also überwiegend an Menschen zwischen 14 und 60 Jahren. Es handelt sich also um eine sehr große Gruppe unserer Bevölkerung. Dabei sind sowohl Männer als auch Frauen betroffen.

Ein wichtiger Faktor ist, dass bei einer Gefahrensituation Adrenalin ausgestoßen wird. Dies kann eventuell das Verhalten der Benutzer in einer Gefahrensituation beeinflussen. Da es sich um eine sehr große Bevölkerungsgruppe handelt, kann man davon ausgehen, dass sowohl erfahrene User als auch unerfahrene User mit dem System interagieren werden.

Auch die Sehkraft der Nutzer muss in Betracht gezogen werden. Denn auch dies kann ein wichtiger Faktor des Designs werden. Besonders in Situationen, in denen man schnell reagieren muss ist es wichtig, die wichtigen Elemente des Designs schnell zu erkennen. Hier muss man davon ausgehen, dass die Person eine eingeschränkte Sehkraft haben kann.

Auch andere körperliche Einschränkung der Nutzer sind Faktoren im Gestaltungsprozess. Wenn ein Nutzer zum Beispiel nur eine Hand hat, ist es wichtig das System auch in Gefahrensituationen schnell mit einer Hand steuern zu können.

1.1.1.2 Stakeholder

Beteiligte Stakeholder an WalkHome.

- Begleitperson(Kinder/Erwachsene)
- Notfallkontakte
- Softwareentwickler
- Designer
- Polizei
- Entführer
- Diebe

- Gewaltverbrecher
- Menschen im Umkreis
- Staat

Die Begleitperson will sicher von einem Ort zum anderen gelangen und dabei von ihren Mitmenschen begleitet werden. Falls etwas auf dem Weg passiert, sollen die Mitmenschen dies mitbekommen.

Der Notfallkontakt hilft der Begleitperson sicher von einem Ort zum anderen zu kommen. Dabei will er informiert werden, falls etwas auf dem Weg passiert, um etwaige Schritte einzuleiten.

Der Softwareentwickler entwickelt das verteilte System zur Lösung des Problems.

Der Designer versucht eine auf den Nutzer angepasste Gestaltlösung zu entwerfen.

Der Mensch im Umkreis ist ebenfalls ein Nutzer der Anwendung. Er ist zufällig in der Nähe eines gerade passierenden Verbrechens, in welches ein User der App involviert ist.

Der Polizist ist für die Sicherheit der Bürger verantwortlich. Ihm könnte das System helfen, wenn er als *Mensch im Umkreis* fungiert.

Der Staat bzw. die aktuelle Regierung. Teile der Regierung könnten Interesse an der Anwendung haben, da ihnen die Sicherheit der Bürger wichtig ist.

Entführer, Diebe, Gewaltverbrecher könnten die App zu ihrem Vorteil missbrauchen.

1.1.1.2.1 Stakeholdergruppen

Nutzer

- Begleitperson
- Notfallkontakte
- Polizei
- Menschen im Umkreis

Entwickler

- Softwareentwickler
- Designer

potenzielle Unterstützer

- Staat
- Firmen

Ursache des Problems

- Gewaltverbrecher
- Diebe
- Entführer

1.1.1.2.2 Stakeholderanalyse

Bezeichnung	Beziehung zum System	Objektbereich	Erfordernis, Erwartung
Benutzer (Begleitperso n & Notfallkontakt)	<p>Anrecht: -</p> <p>Anteil: Sie muss sich im System registrieren. Sie muss ihre Standortdaten für andere Nutzer der App freigeben.</p> <p>Anspruch: Die Anwendung soll zuverlässig sein, denn im Notfall ist man auf die Anwendung angewiesen.</p> <p>Interesse: Die Person will von einem Ort zum anderen gelangen. Wenn etwas auf dem Weg passiert, will sie sich anderen Leute mitteilen können.</p>	Nutzerschnittstelle, ganze Anwendung	<p>Erfordernis: Es erfordert ein gewisses Grundverständnis über die Benutzung eines Smartphones.</p> <p>Erwartung: Die Begleitperson erwartet eine funktionierende Anwendung, die auch in Gefahrensituationen zuverlässig ist.</p>
Entwickler (Software Entwickler & Designer)	<p>Anrecht: -</p> <p>Anteil: Der Entwickler muss Komponenten des Systems programmieren.</p> <p>Anspruch: -</p> <p>Interesse: Er will eine Anwendung entwickeln, die zuverlässig funktioniert und den Erwartungen der Nutzer entspricht.</p>	System	<p>Erfordernis: Er benötigt ausreichende Kenntnis in Programmierung und in dem Erarbeiten einer Gestaltlösung.</p> <p>Erwartung: Eine Anwendung, die bei den Nutzern auf Zuspruch trifft.</p>

Bezeichnung	Beziehung zum System	Objektbereich	Erfordernis, Erwartung
Potentieller Unterstützer (Staat)	Anrecht: - Anteil: Würde versuchen die App zu verbreiten. Anspruch: Eine funktionierende Anwendung, die zuverlässig und sicher ist. Interesse: Sicherheit für die Bürger des Staates.	ganze Anwendung	Erfordernis: Strategie zur Verbreitung der Applikation. Erwartung: Steigerung der Sicherheit einiger Bürger durch die Anwendung.
Ursache des Problems (Gewaltverbrecher, Diebe, Entführer)	Anrecht: - Anteil: - Anspruch: - Interesse: Menschen überfallen, bestehlen, nötigen. Die App für diese Zwecke missbrauchen.	Sicherheit der Anwendung	Erfordernis: Erwartung: Durch mangelnde Sicherheit der Applikation einen eigenen Vorteil ziehen.

1.1.1.3 User Profiles

Begleitperson

Alter: 14-60 Jahre
 Geschlecht: männlich/weiblich
 Beruf: beliebig
 Smartphone-Kenntnisse: Grundkenntnisse
 Beeinträchtigung: keine
 Verhalten: unvorsichtig, nervös
 Verfügbare Technologie: Smartphone
 Motivation: sicher von einem Punkt zum anderen gelangen

Notfallkontakt

Alter: 18-60 Jahre
 Geschlecht: männlich/weiblich
 Beruf: beliebig
 Smartphone-Kenntnisse: Grundkenntnisse
 Beeinträchtigung: eventuelle
 Verhalten: besorgt, nervös
 Verfügbare Technologie: Smartphone/Computer
 Motivation: Freund/Kind soll sicher von einem zum anderen Punkt gelangen

Person im Umfeld

Alter: 14-60 Jahre
Geschlecht: männlich/weiblich
Beruf: beliebig
Smartphone-Kenntnisse: Grundkenntnisse
Beeinträchtigung: eventuelle
Verhalten: aufmerksam, unvorsichtig, mutig
Verfügbare Technologie: Smartphone
Motivation: anderen Menschen helfen

1.1.1.4 Personae und reale Personen

Persona 1 (Begleitperson)

Tim Schulz ist 26 Jahre alt und gelernter Bäcker. Er ist eher schmächtig und macht in seiner Freizeit keinen Sport. Tim ist ein Mensch von rückhaltender Natur. Er ist in einer 3 jährigen Beziehung und lebt mit seiner Freundin in einer gemeinsamen Wohnung. Seine Freunde kann oft nicht weiterschlafen, bis sie weiß dass er gut angekommen ist. Er lebt in Berlin und muss nachts zu Fuß zur Arbeit gehen. Er hat einen guten Draht zu seinem Chef, welcher genau wie er in der Nacht arbeitet. Sein Chef wohnt über der Bäckerei und ist daher immer vor Tim am Arbeitsplatz. Da er kein besonders hohes Einkommen hat, wohnen er und seine Freundin in einer Wohnung in Berlin-Neuköln. Diese Gegend ist in Berlin als ein „Problemviertel“ bekannt. In seiner Gegend kommt es häufig vor, dass jemand nachts auf der Straße überfallen oder zusammengeschlagen wird.

Persona 2(Notfallkontakt)

Fiona Schmitz ist 24 Jahre alt und lebt mit ihrem Freund in einer gemeinsamen Wohnung in Berlin-Neuköln. Sie ist Erzieherin und arbeitet von 8 Uhr morgens bis 13 Uhr Mittags. In ihrer Freizeit treibt Fiona gerne Sport und spielt gerne mit ihrem Smartphone. Mit 4 Jahren verlor sie bei einem Autounfall ihren rechten Unterarm. Mittlerweile hat sich jedoch damit abgefunden und gelernt damit umzugehen. Daher meistert sie mittlerweile Alltagssituationen sehr gekonnt. Sie ist oft besorgt um ihre Freunde und vor allem um ihren Freund, wenn diese/r nachts unterwegs sind. In ihrer Wohngegend, gibt es viele Gewaltdelikte und ein Freund von ihr wurde vor kurzer Zeit nachts überfallen und ausgeraubt. Wenn ihr Freund um 2 Uhr zur Arbeit geht kann sie erst wieder schlafen, wenn sie weiß, dass er sicher angekommen ist. Ihr Freund meldet sich aber oft nicht sofort, da er, wenn er ankommt, mit seinem Chef redet und daraufhin vergisst ihr zu schreiben.

Persona 3(Person im Umfeld)

Thorsten Werner ist 30 Jahre alt und Handwerker. Er wohnt in Berlin und geht oft abends mit Freunden in eine Bar. In seiner Freizeit, übt er Kampfsport aus. Mittlerweile macht er das schon seit 5 Jahren und hat schon sehr viele Kämpfe gewonnen. Wenn er Abends mit seinen Freunden in einer Bar ist, gehen sie manchmal erst nach Mitternacht nach Hause. Sein kleiner Bruder geht auch öfters mit seinen Freunden etwas trinken. Wenn Thorsten kann holt er ihn mit dem Auto ab, damit er sicher nach Hause kommt.

Reale Person 1

Anna Coen ist 20 Jahre alt und Studentin. Sie studiert in Wuppertal Grundschullehramt. In ihrer Freizeit reitet sie, kocht gerne und trifft sich mit Freunden. Sie lebt zu Hause bei ihren Eltern. Das Haus ist ländlich gelegen. Sie wohnt als einzige ihrer Freunde in der Gegend und muss daher, zum Beispiel nach einem Abend in Köln, alleine nach Hause. Wenn Sie irgendwo hin muss fährt sie meist mit dem Auto, da die öffentliche Verkehrsanbindung nicht sehr gut ist. Des Weiteren ist der Bus abends sehr leer, was ihr Angst macht. Anna benutzt täglich sehr oft ihr Smartphone und ist daher eine gewandte Nutzerin. Ihre Eltern sind sehr besorgt um sie, besonders wenn sie abends oder nachts weg ist.

Reale Person 2

Nicole Coen ist 46 Jahre alt und die Mutter von drei Kindern. Sie wohnt mit ihrem Mann und zwei der beiden Kinder in einer ländlichen Gegend. In der Freizeit beschäftigt sich Nicole viel mit Pferden und ist gerne in der Natur. Sie hat leichte Sehschwäche, in Form von einer Weitsichtigkeit. Daher trägt sie, zum Beispiel beim Lesen, oft eine Brille. Ihre Kinder liegen ihr besonders am Herzen, weshalb sie sich sehr um sie sorgt. Sie benutzt regelmäßig ihr Smartphone und ist eine gewandte Userin.

1.1.1.5 Use Cases:

Use Case 1: Weg zum Ziel bestimmen	
Primary Actor	Begleitperson
Scope	Walkhome Android App
Level	Ziel des Benutzers
Stakeholder and Interests	Begleitperson; will den Weg zu einem bestimmten Ziel berechnen lassen.
Preconditions	App ist geöffnet und der Nutzer ist angemeldet. Der Nutzer befindet sich auf dem Startbildschirm. Es besteht eine Internet- & GPS-Verbindung.
Minimal Guarantees	Dem User wird angezeigt, dass ein Fehler aufgetreten ist.
Success Guarantees	Die Route wird angezeigt. Der Erfolg wird visuell bestätigt.
Scenario	<ol style="list-style-type: none">Der Nutzer übermittelt dem System ein Ziel.Das System Sucht einen WegDas System stellt den Weg visuell dar

Use Case 1: Weg zum Ziel bestimmen	
Extensions	<ol style="list-style-type: none"> 1. Die Route konnte nicht gefunden werden <ol style="list-style-type: none"> 1.1 Es wird dem User visuell gezeigt, dass die Route nicht gefunden werden konnte 2. Es besteht keine Internetverbindung/ GPS Signal <ol style="list-style-type: none"> 2.2 Es wird dem Nutzer visuell angezeigt, dass keine Internetverbindung/GPS Signal besteht.

Use Case 2: Kontakt hinzufügen	
Primary Actor	Begleitperson
Scope	Walkhome Android App
Level	Ziel des Benutzers
Stakeholder and Interests	Genereller Nutzer; will einen Kontakt hinzufügen, den er begleiten kann oder der als Notfallkontakt dient.
Preconditions	App ist geöffnet und der Nutzer ist angemeldet. Der Nutzer befindet sich auf dem Kontaktbildschirm. Es besteht eine Internetverbindung.
Minimal Guarantees	Falls ein Fehler auftritt, wird dies dem User angezeigt und darauf hingewiesen das Smartphone mit dem Internet zu verbinden.
Success Guarantees	Der gesuchte Kontakt wird angezeigt. Der gesuchte Kontakt wird hinzugefügt.
Scenario	<ol style="list-style-type: none"> 1. Der Nutzer übermittelt dem System einen Nutzernamen 2. Das System sucht den Nutzernamen 3. Das System übermittelt dem User visuell den gefundenen Kontakt 4. Der Nutzer fügt den Kontakt hinzu 5. Das System ordnet den Kontakt dem User zu.
Extensions	<ol style="list-style-type: none"> 1. Es konnte kein Kontakt mit dem Namen gefunden werden <ol style="list-style-type: none"> 1.1 Dem Nutzer wird visuell gezeigt, dass kein Kontakt mit dem Namen gefunden wurden konnte 2. Es besteht keine Internetverbindung <ol style="list-style-type: none"> 2.1 Dem Nutzer wird visuell angezeigt, dass keine Internetverbindung besteht

Use Case 3: Alarm manuell auslösen	
Primary Actor	Begleitperson
Scope	Walkhome Android App

Use Case 3: Alarm manuell auslösen	
Level	Ziel des Benutzers
Stakeholder and Interests	Begleitperson; will ihren Mitmenschen mitteilen, dass ihr etwas zugestoßen ist Notfallkontakt; will benachrichtigt werden, falls der Begleitperson etwas zustößt
Preconditions	App ist geöffnet und der Nutzer ist angemeldet. Der Nutzer befindet sich auf dem Startbildschirm. Es besteht eine Internet- & GPS-Verbindung.
Minimal Guarantees	Ein lauter Alarmton wird ausgelöst.
Success Guarantees	Der Begleitperson wird bestätigt, dass ein Alarm ausgelöst wurde. Dem Notfallkontakt wird der Alarm des Nutzers in der Anwendung angezeigt.
Scenario	<ol style="list-style-type: none"> 1. Der Nutzer löst den Alarm aus. 2. Das System fragt nach ob der Nutzer den Alarm wirklich auslösen will <ol style="list-style-type: none"> 2.1 Der Nutzer beantwortet mit „Ja“ - Das System sendet einen Alarm an die Notfallkontakte 2.2 Der Nutzer beantwortet mit „Nein“ - Das System zeigt visuell dass der Alarm nicht ausgelöst wurde
Extensions	<ol style="list-style-type: none"> 1. Es besteht keine Internetverbindung. <ol style="list-style-type: none"> 1.1 Dem Notfallkontakt wird eine SMS als Alarm zugeschickt. 2. Es besteht keine Mobilfunkverbindung. <ol style="list-style-type: none"> 2.1 Das Smartphone gibt einen Alarmton wieder.

Use Case 4: Notfall-Alarm erhalten	
Primary Actor	Notfallkontakt
Scope	Walkhome Android App
Level	Ziel des Benutzers

Use Case 4: Notfall-Alarm erhalten	
Stakeholder and Interests	Begleitperson; will ihren Mitmenschen mitteilen, dass ihr etwas zugestoßen ist Notfallkontakt; will der Begleitperson, welcher etwas zugestoßen ist, helfen.
Preconditions	Der Nutzer ist angemeldet. Es besteht eine Internetverbindung.
Minimal Guarantees	Er kann den Kontakt oder den Notruf „110“ über die App anrufen.
Success Guarantees	Die GPS Daten der Begleitperson werden dem Notfallkontakt angezeigt.
Scenario	<ol style="list-style-type: none"> 1. Der Nutzer erhält einen Alarm, vom Smartphone des Begleitkontakte. 2. Der Nutzer erkennt auf der Karte an welchem Ort sich der Begleitkontakt befindet. 3. Der Nutzer ruft die Begleitperson an und fragt, ob alles in Ordnung ist <ol style="list-style-type: none"> 3.1 Die Begleitperson nimmt den Anruf an und teilt mit, dass alles in Ordnung ist. 3.2 Die Begleitperson nimmt den Anruf nicht an - Der Nutzer ruft über die App den Notruf „110“ an und teilt den Beamten den Standort der Begleitperson mit.
Extensions	<ol style="list-style-type: none"> 1. Der aktuelle GPS Standpunkt der Begleitperson wird nicht mehr übermittelt. <ol style="list-style-type: none"> 1.1 Das System zeigt den letzten verfügbaren Standpunkt der Person an. 2. Der Nutzer erhält eine Notfall-SMS der Begleitperson. <ol style="list-style-type: none"> 2.1 Der SMS Empfang wird nicht vom System behandelt.

1.1.2 Festlegen der Nutzungsanforderungen

Der Nutzungskontext ist den oben stehenden Artefakten, der Benutzer- und Aufgabenmodellierung zu entnehmen.

Da die Altersgruppe der User einen sehr großen Bereich abdeckt, ist ein Erfordernis, dass möglichst viele Nutzer in verschiedenen Altersgruppen mit der App interagieren können. Hierbei ist auf die Wahl der Symbole/Icons zu achten, da verschiedene Altersgruppen eventuell verschiedene kognitive Vorstellungen von den darzustellenden Begriffen haben. Außerdem kann die Lesbarkeit und Größe der Layout-Elemente entscheidend sein, da die Sehstärke im Alter abnehmen kann, es aber auch im jungen Alter zu einer Störung/Schwächung der Sehkraft kommen kann.

Weitere körperliche Einschränkungen müssen beachtet werden, sodass zum Beispiel die App auch mit nur einer Hand benutzt werden kann. Die Applikation soll die User auf dem Weg nach Hause begleiten, dieser ist in den meisten Fällen draußen. Ein weiteres Erfordernis ist also, dass das Display tagsüber nicht zu dunkel und nachts nicht zu hell ist. Dies wird von der Zielplattform Android allerdings schon übernommen. Die Helligkeit des Displays wird anhand des eintreffenden Lichtes angepasst.

Die App ist darauf ausgelegt in Gefahrensituationen zu helfen. Auch hier ist es wichtig, die richtigen Designentscheidungen zu treffen. Bei Gefahrensituationen schüttet der menschliche Körper das Stresshormon Adrenalin aus. In der Folge handelt der Mensch oft anders, als er das normalerweise tun würde. Daraus folgt, dass bei der Gestaltung darauf geachtet werden sollte eine Oberfläche zu erarbeiten, bei der alle Eingabemöglichkeiten auch in Gefahrensituationen schnell erkannt werden.

Bei der Aufgabenermittlung hat sich herausgestellt, dass eine Alternative zu GPS und Internet bestehen sollte. Wenn also weder Internet noch GPS zur Verfügung steht, kann zum Beispiel auf das Mobilfunknetz gewechselt werden. Weiterhin hat sich herausgestellt, dass eine visuelle Bestätigung des zuvor passierten Vorgangs in der App dem User angezeigt werden muss.

Auf Grund von langjähriger Erfahrung mit Applikationen jeglicher Art hat sich herausgestellt, dass ein Menüpunkt für die Einstellungen in einer großen Anzahl erfolgreicher und etablierter Apps zu finden ist. Die Präsenz dieses Menüpunktes kann somit den Erwartungen vieler User entsprechen. Das Gleiche gilt für den Menüpunkt „Kontakte“.

In der Domänen-Recherche wurde herausgefunden, dass einige Opfer von Gewaltverbrechen entweder niedergeschlagen wurden oder zu einem anderen Ort verschleppt wurden, an dem sie dann festgehalten wurden. Die Opfer konnten sich oft nicht mehr anderen Mitmenschen mitteilen, wodurch das Verbrechen erst nach der Tat aufgefallen ist. Hieraus ergibt sich die Anforderung, dass die App das Abkommen vom Weg erkennt und erkennt, wenn der Nutzer auf einem Punkt verharrt (zum Beispiel weil er niedergeschlagen wurde).

Beim Eintreffen eines der beiden Ereignisse soll sie den Nutzer abfragen, ob alles in Ordnung ist. Wenn dieser nicht antwortet, soll ein Alarm ausgelöst werden. Darüber hinaus soll es aber noch die Möglichkeit geben einen Alarm manuell auszulösen. Da eine Notfallsituation zu jeder Zeit möglich sein kann, sollte dieser manuelle Alarm schnell und einfach auszulösen sein. Es ist allerdings darauf zu achten, dass man ihn nicht aus Versehen auslösen kann.

Aus dem Konzept abgeänderte Funktionale, Qualitative und Organisatorische Anforderungen

1.1.2.1 Funktionale Anforderungen

- F10 Wenn das Handy eine Verbindung zum Internet hat, muss das System eine Route zwischen zwei Standorten berechnen können.
- F20 Wenn der Nutzer von der Route abkommt, muss das System die Abweichung berechnen können.
- F30 Wenn der Standort des Smartphones sich für eine gewisse Zeit nicht ändert, muss das System diese Zeit berechnen können.

- F40 Das System muss der Begleitperson die Möglichkeit geben einen Notfall-Alarm auszuwählen.
- F50 Wenn ein Alarm ausgelöst wurde und keine Verbindung zu Internet oder Mobilfunk besteht, muss das System einen lauten Ton ausgeben.
- F60 Wenn ein Alarm ausgelöst wurde und keine Verbindung zum Internet besteht, muss das System eine Alarm-SMS versenden können.
- F70 Wenn ein Alarm ausgelöst wurde und eine Internetverbindung besteht, muss das System einen Alarm an den Server schicken.
- F80 Das System muss die Internet-, GPS- und Mobilfunkverbindung überprüfen können.
- F90 Das System muss dem Nutzer die Möglichkeit geben, andere Menüpunkte auszuwählen.
- F100 Wenn der Nutzer einen Menüpunkt auswählt, muss das System ein anderes Layout rendern können.
- F110 Das System muss dem Nutzer die Möglichkeit geben einen gesuchten Nutzer einzugeben.
- F120 Das System soll dem Nutzer die Möglichkeit geben seine Nutzerdaten einzugeben.
- F130 Wenn eine Internetverbindung besteht, muss das System Nutzerdaten an den Server senden können.
- F140 Wenn eine Internetverbindung besteht, muss das System dazu fähig sein Nutzer von der Datenbank abzurufen.
- F150 Das System soll den GPS-Standpunkt des Smartphones an den Server senden können.
- F160 Das System soll die GPS-Standpunkte von bestimmten anderen Geräten vom Server abrufen können.
- F170 Falls ein Alarm eines Kontaktes im Server eintrifft, soll das System den Alarm automatisch zugesendet bekommen.
- F180 Das System muss dem Notfallkontakt die Möglichkeit geben den Notruf auszuwählen.
- F190 Wenn der Notruf ausgewählt wurde und eine Verbindung zum Mobilfunknetz besteht, muss das System eine Mobilfunkverbindung zum Notruf herstellen können.
- F200 Das System muss dem Notfallkontakt die Möglichkeit geben einen Anruf der Begleitperson auszuwählen.

- F210 Wenn ein Anruf der Begleitperson ausgewählt wurde und eine Verbindung zum Mobilfunknetz besteht, muss das System eine Mobilfunkverbindung zur Begleitperson herstellen können.
- F220 Das System soll dem Nutzer die Möglichkeit geben Einstellungen auszuwählen.
- F230 Wenn der Nutzer bestimmte Einstellungen auswählt, soll das System bestimmte Eigenschaften ändern können.
- F240 Das System soll die Postleitzahl des aktuellen Standorts vom Server anfragen können.
- F250 Wenn ein Nutzer einen Alarm auslöst, soll das System eine Alarm-Nachricht an alle Nutzer, die sich im Gebiet mit der gleichen Postleitzahl befinden, senden können.
- F260 Das System muss dem Nutzer die Möglichkeit geben einen Angekommen-Knopf auszuwählen.
- F270 Wenn der Nutzer den „Angekommen-Knopf“ ausgewählt hat, muss das System eine Meldung dafür an den Server senden.
- F280 Wenn eine „Angekommen-Meldung“ von einem Kontakt den Server erreicht hat, muss das System automatisch eine „Angekommen-Nachricht“ vom Server zugeschickt bekommen.

1.1.2.2 Qualitative Anforderungen

- Q10 Zuverlässigkeit - Das System sollte sehr zuverlässig sein, da die Sicherheit der User im Vordergrund steht und diese durch die App verbessert werden kann, wenn sie im Notfall auch funktioniert.
- Q20 Geschwindigkeit - Das System sollte möglichst schnell den Alarm an die Notfallkontakte weiterleiten, da es im Notfall auf Sekunden ankommen kann.
- Q30 Übersichtlichkeit - Das System sollte übersichtlich sein, um den Nutzern auch im Notfall eine möglichst schnelle Benutzung zu ermöglichen.
- Q40 Selbstbeschreibungsfähigkeit - Da die User der App eine große Altersgruppe umfasst, sollte für erfahrene User und unerfahrene User das Verständnis der App möglichst leicht fallen.

1.1.2.3 Organisatorische Anforderungen

- O10 Hohe Nutzerzahl - Um das Feature, Menschen in der Umgebung zu informieren effektiv zu machen, benötigt man eine große Userzahl. Nur wenn viele Menschen die App haben ist die Wahrscheinlichkeit hoch, dass ein „Helfer“ in der Nähe ist.

1.1.2.4 Befragung der Personen Anna Coen und Nicole Coen zur Interfacegestaltung

Beim menschzentrierten Gestaltungsprozess ist es wichtig, den potentiellen User in den Prozess mit einzubeziehen. Daher haben wir uns Vorschläge, von Anna Coen und Nicole Coen für die Interfacegestaltung geholt. Beide passen ins User Profile.

Den Testpersonen wurde eine Vorlage mit einem Bereich zum Zeichnen, welcher in der Größe eines Smartphone-Bildschirms ist, gegeben. Dazu wurden verfügbare Elemente angegeben, die auf dem Bildschirm angezeigt werden sollten. Diese wurden aus den Nutzungsanforderungen abgeleitet.

Die Elemente sind: Karte, Alarm, Einstellungen, Wegziel, Angekommen, Kontakte. Nun wurde den Probanden genau erklärt, welchen Zweck die App erfüllt und worum es sich bei den Elementen handelt.

So ist :

- das Element Karte zum Anzeigen der Karte und der Strecke vorgesehen
- das Element Alarm zum Auslösen eines Alarms
- das Element Einstellungen, um zu den Einstellungen zu gelangen
- das Element Wegziel, um ein gewünschtes Ziel einzugeben
- das Element Angekommen, um zu signalisieren, dass man sicher angekommen ist
- das Element Kontakte, um auf den Kontaktbereich zu kommen, in welchem man diese verwaltet.

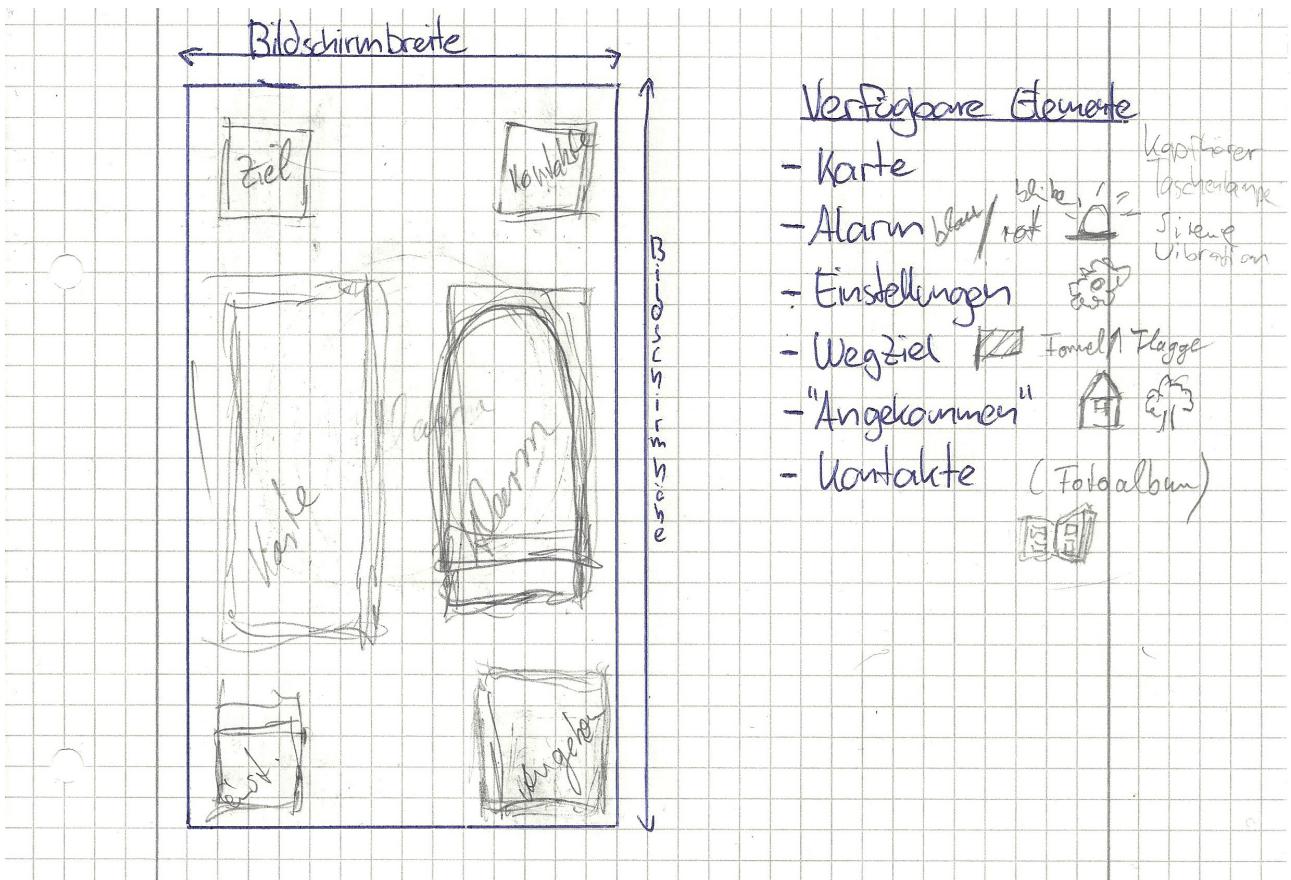
Darauf hin sollten die Probanden ein Interface gestalten. Bei diesem sollten sie die verschiedenen Optionen so anordnen, dass es für sie im Allgemeinen und bezogen auf mögliche Gefahrensituationen am meisten Sinn ergibt. Darüber hinaus sollten sie noch zu den einzelnen Elementen geeignete Symbole finden, die sie mit dem Zweck des Elements assoziieren.

Vorschlag Nicole Coen

Der Erste Vorschlag kommt von Nicole Coen. Sie fand es besonders wichtig, dass der Alarmknopf möglichst groß auf dem Bildschirm zu sehen ist. Die Karte sollte groß daneben sein, da diese für sie auch sehr wichtig ist. Das Ziel hat sie oben links angeordnet, da man es nur einmal braucht und nicht schnell ändern muss. Die Begründung war, dass man den oberen Bereich schlechter mit dem Daumen erreichen kann. Das gleiche gilt für die Kontakte. Die Elemente „Einstellungen“ und „Angekommen“ sind unten angeordnet, da diese laut Nicole wichtiger sind und daher leichter zu erreichen sein sollen. Der Alarmknopf ist aus demselben Grund rechts angeordnet. Hierbei ist zu erwähnen, dass Nicole ihr Smartphone mit der rechten Hand benutzt.

Ein weiterer Vorschlag von Nicole ist es, den Alarmknopf wie eine Sirene aussehen zu lassen. Hierdurch könnte man die Funktion gut im Voraus erahnen. Wenn man ihn drückt soll er blau Rot leuchten, die Taschenlampe soll blinken, das Handy soll vibrieren und ein lauter Ton soll zu hören sein. Hierdurch solle man zum einen merken, dass der Alarm ausgelöst wurde, zum anderen soll es Menschen im Umfeld aufmerksam machen und den Täter verunsichern.

Ein weiterer Vorschlag entwickelte sich im Gespräch mit Nicole. Sie sagte, dass sie von einem Gerät gelesen hat, welches man Kindern in die Tasche stecken kann. Wenn diese an einer mit dem Gerät verbundenen Reißleine ziehen, wird ein lauter Alarm ausgelöst. Sie fragte, ob man so etwas mit einem Smartphone verbinden könnte.



gez. Nicole Coen

Da die App allein stehen soll und kein weiteres Gerät oder Gadget von Nöten sein soll, erklärte der Gesprächsleiter Yannik Huber, dass man als Alternative den Aux-Eingang, welcher mit den Kopfhörern verbunden ist, als Reisleine nehmen könnte.

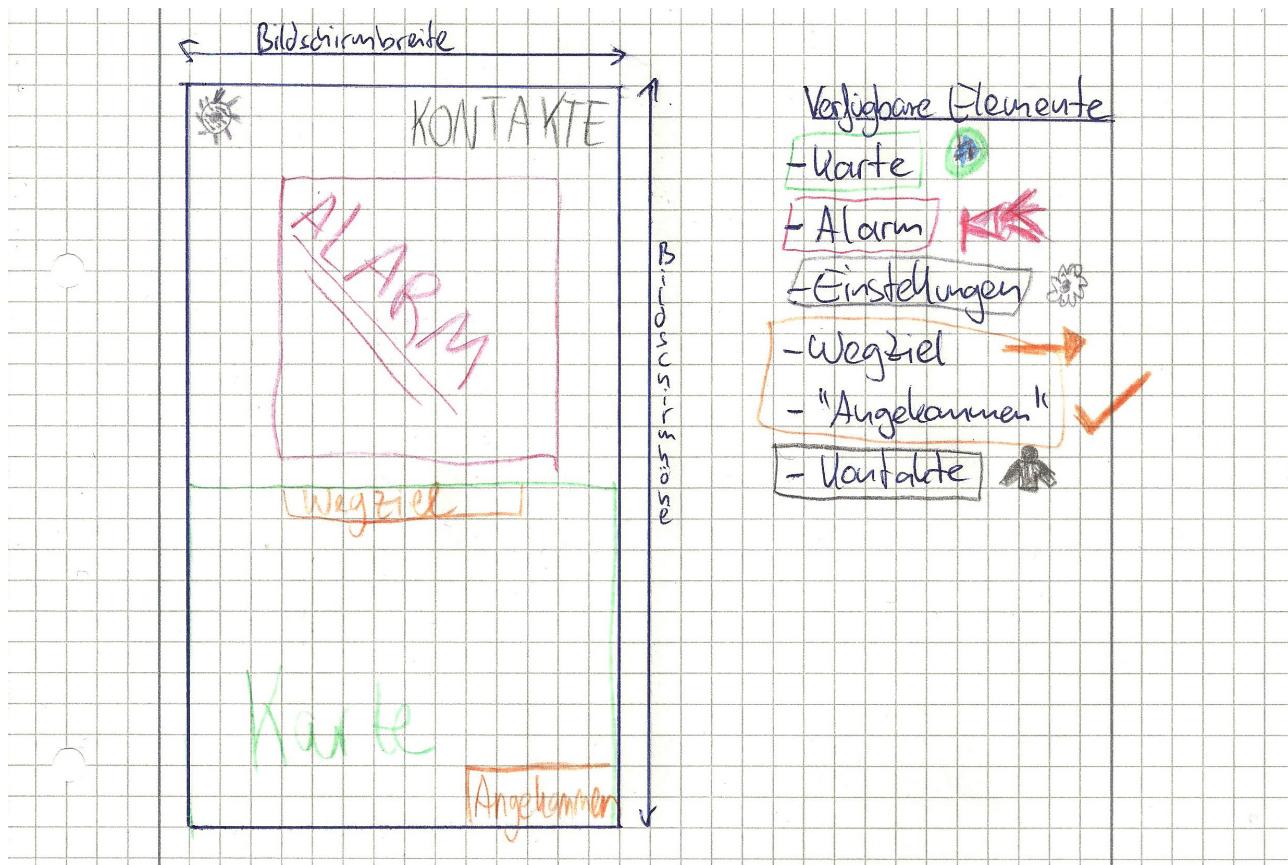
Diese Idee fand Nicole sehr gut und sagte, dass sie ein solches Feature als sehr nützlich erachtet. Denn in Gefahrensituationen sei es einfacher an den Kopfhörern zu ziehen, als einen Knopf zu drücken.

Die Symbole für die Elemente, waren:

- Eine Sirene für den Alarm
- Ein Zahnrad für die Einstellungen
- Eine Flagge für das Ziel
- Ein Haus oder Baum für Angekommen
- Ein Fotoalbum für Kontakte

Dies waren die ersten Symbole, die Nicole mit den Begriffen assoziiert hat.

Vorschlag Anna Coen



gez. Anna Coen

Der zweite Vorschlag kommt von Anna Coen. Auch für Anna war es wichtig, dass der Alarmknopf möglichst groß auf dem Bildschirm zu sehen ist. Sie hat diesen daher in der oberen Mitte des Bildschirms angeordnet. Die Kontakte und die Einstellungen waren ihrer Meinung nach nicht so wichtig, daher hat sie diese in den oberen beiden Ecken des Bildschirms gezeichnet. Die Karte sollte den unteren Bereich des Bildschirms einnehmen, da man dort laut Anna leicht und schnell hinsehen kann. Die Eingabe für das Wegziel sollte in der Mitte liegen. Und der „Angekommen“-Button unten rechts auf dem Bildschirm. So könne man den „Angekommen“-Button schnell erreichen.

Die Symbole für die Elemente waren folgende:

- Eine Erdkugel für die Karte
- Ein Megaphon/Horn für den Alarm
- Ein Zahnräder für die Einstellungen
- Ein Pfeil nach rechts für das Wegziel
- Ein Haken für „Angekommen“
- Ein neutraler Oberkörper für Kontakte

Auch hier waren es die ersten Assoziationen, die Anna bei den Begriffen hatte.

1.1.3 Erarbeitung von Gestaltungslösungen zur Erfüllung der Nutzungsanforderungen

Die erstellte Gestaltungslösung basiert auf den Nutzungsanforderung, sowie den Grundsätzen der menschzentrierten Gestaltung, als auch auf den vorgeschlagenen Interface-Designs von den Probanden Anna Coen und Nicole Coen.

Gestaltungslösung 1 - Homescreen



Es wurde die Gestaltungslösung für den Homescreen entworfen. Dieser ist der wichtigste Teil der Applikation, denn er ist die Aktivität in der die Begleitperson die meiste Zeit verbringt.

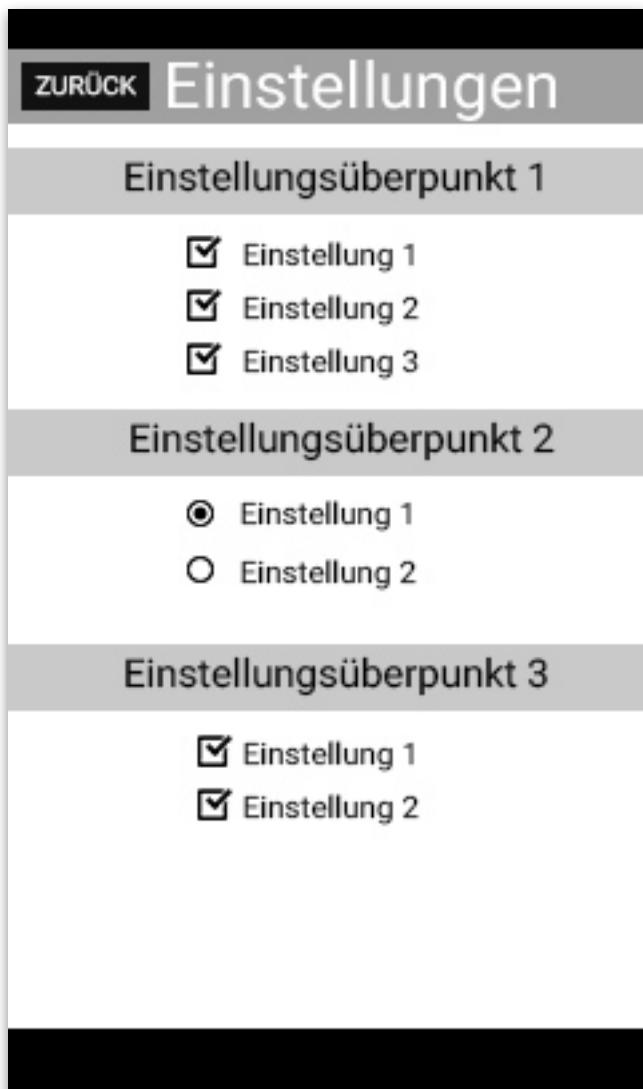
Dieser Teil der App muss besonders gut auf Gefahrensituationen angepasst sein, was die besondere Schwierigkeit der Gestaltung unseres Systems ist.

Der Alarmbutton ist aufgrund der Vorschläge von Anna Coen und Nicole Coen sehr groß. Auch die Karte nimmt einen großen Platz auf dem Interface ein. Die Flächen für Einstellungen, Angekommen und Kontakte sind verhältnismäßig klein, da sie laut der User nicht von so großer Wichtigkeit sind, wie der Alarm. Alarm, Einstellungen, Angekommen und Kontakte sind leicht mit dem Daumen zu erreichen. Dabei sind Alarm und Angekommen sowohl für Rechts- als auch für Linkshänder besonders gut zu erreichen. Die Zieleingabe ist direkt unter der Karte platziert, um eine direkte visuelle Verbindung zu dieser herzustellen. Im Hinblick auf die Selbstbeschreibungsfähigkeit dienen Symbole zur besseren Erkennung der einzelnen Komponenten des Interface.

Dabei wurde auf die Vorschläge der beiden Probanden eingegangen. Das Ziel ist eine Flagge und der Alarm eine Sirene, wie es Nicole Coen vorgeschlagen hat. Das Zeichen für „angekommen“ ist wie von Anna Coen vorgeschlagen ein Häckchen. Das Zahnrad für Einstellungen haben beide vorgeschlagen. Für die Kontakte wurde eine Mischung der Vorschläge gestaltet. So ist das Symbol ein Telefonbuch mit dem Oberkörper einer Person darauf. Des weiteren wurde von beiden Probanden vorgeschlagen, dass der Alarmknopf rot sein soll. Dies wurde hier auch so umgesetzt. Bezogen auf die Fehlertoleranz soll dem User bei Fehlern sofort signalisiert werden, was er falsch gemacht hat. Dies geschieht in Form eines „Prompts“ der sich öffnet, falls etwas nicht korrekt eingegeben wurde oder ein

anderer Fehler aufgetreten ist. Ein weiterer wichtiger Aspekt war die Übersichtlichkeit des Designs. Es sind lediglich sieben Elemente auf dem Homescreen vorhanden. Darunter sind 6 Eingabemöglichkeiten für den User. Dies zielt darauf ab, dass das Arbeitsgedächtnis des Users alle Eingabemöglichkeiten behalten kann.

Gestaltlösung 1 - Einstellungen



Gestaltlösung 1 - Kontakte



Das Layout der Einstellungen ist ziemlich einfach gehalten. Es gibt einige Einstellmöglichkeiten, die „ein“ oder „aus“ gestellt werden können. Diese sind untereinander angeordnet. Hier wurde das Design an die Erfahrungen des Designers mit Apps, welche auf dem Markt erfolgreich sind, angepasst. Im oberen Teil des Layouts gibt es eine Überschrift die besagt, in welchem Bereich der App sich der User befindet. Daneben ist ein Button angeordnet, der zurück zum vorigen Layout führt. Diese beiden Elemente helfen dem Nutzer den Überblick bei der Benutzung der Applikation zu behalten. Sie sind auch im Layout der Kontakte vorhanden. Das Layout, auf dem die Kontakte angezeigt werden ist zweigeteilt. Auf der oberen Hälfte kann der User nach anderen Usern der App suchen. Die Lupe ist ein häufig benutztes Symbol für die Suchfunktion und wird daher auch hier benutzt. Die gefundenen Kontakte werden unter der Texteingabe angezeigt, was auch häufig in anderen Apps vorkommt. Wenn die App keinen Kontakt mit

dem gesuchten Nutzernamen findet, wird das dem User visuell angezeigt. Das gleiche gilt für das Symbol, welches für das Hinzufügen des anderen Users verwendet wurde. Im unteren Teil des Layouts werden die vorhandenen Kontakte des Nutzers angezeigt. Des weiteren gibt es eine Information darüber, ob der Kontakt unterwegs ist, angekommen ist oder nicht unterwegs ist. Dies erlaubt einen schnellen Überblick über den Status der Kontakte. Wenn ein Kontakt unterwegs ist oder angekommen ist, kann man auf das Element des Kontaktes drücken und kommt zum speziellen Layout des Kontaktes.

Gestaltlösung 1 - Kontakt



Hier wird einem der aktuelle Standpunkt des Kontaktes auf der Karte angezeigt, sofern er noch unterwegs ist. Des weiteren wird angezeigt ob er einen Alarm ausgelöst hat, angekommen ist oder noch unterwegs ist. Bei jedem der drei Ereignisse wird die zugehörige Uhrzeit angegeben. Im unteren Bereich gibt es zwei weitere Designelemente. Diese führen zu einem Anruf bei der Polizei oder zu einem Anruf zu dem Kontakt. Der Nutzer kann durch drücken auf die Elemente auf die beiden Funktionen zugreifen.

Angekommen um:
20:00 Uhr

Unterwegs seit:
19:00 Uhr

Im Hinblick auf die Fehlertoleranz gibt es nicht übermäßig viele Eingabemöglichkeiten. Wenn der Nutzer bei einem der beiden Textfelder eine falsche Eingabe macht, wird ihm das visuell mitgeteilt. Um mit dem Rest der Eingabemöglichkeiten zu interagieren, bedarf es nur des Berühren des Touchscreens. Das nicht beabsichtigende Drücken eines Elementes könnte allerdings zu Problemen führen. Genau wie das Verwechseln der Buttons. Daher ist der Alarm-Button sehr groß gehalten, damit man ihn auch in Gefahrensituationen schnell findet und drücken kann. Wenn der User den Alarm drückt wird dies noch einmal überprüft und der Nutzer wird im Dialog gefragt, ob er den Alarm auslösen will. Hierdurch können Fehler durch das unabsichtliche Drücken des Alarmknopfes minimiert werden.

Die App ist im Bereich der Einstellungen individualisierbar. Hier soll man verschiedene Optionen auswählen können. Zum Beispiel, ob die App die Abweichung der Route messen soll oder ob auf das Feature verzichtet wird.

1.1.4 Gestaltungslösungen aus der Benutzerperspektive evaluieren

1.1.4.1 Prototyp

Um den Probanden das finale Produkt so nah wie möglich zu bringen, wurde ein Prototyp der App erstellt. Dieser Prototyp hatte schon die Funktion der Routensuche und konnte die Abweichung von der Route berechnen. Des weiteren waren die Layouts implementiert, wodurch man sich durch die App navigieren konnte. Die restlichen Funktionen waren noch nicht gegeben. Die meisten werden jedoch im finalen Produkt für die Nutzer nicht sichtbar sein, da sie im Hintergrund laufen.

Die Gestaltung des Prototypen war gemäß der vorher erarbeiteten Gestaltungslösung. Die Probanden hatten, zusätzlich zu dem Prototypen, die Gestaltungslösung aller Layouts in bildlicher Form auf einem Laptop vorliegen.

1.1.4.2 Think Aloud

1.1.4.2.1 Think Aloud mit Regina Huber

Über Regina Huber

Regina Huber ist 53 Jahre alt und Mutter von drei Kindern. Sie arbeitet in einer Kindertagesstätte. Sie wohnt mit ihrem Mann und einem ihrer drei Kinder in einer ländlichen Gegend.

Sie benutzt ihr Smartphone mehrmals täglich, ist allerdings keine erfahrene Nutzerin.

Versuchsaufbau

Der Proband sitzt an einem Tisch, auf dem ein Laptop und ein Smartphone liegen. Auf dem Laptop sind die erarbeiteten Gestaltungslösungen der App WalkHome zu sehen. Auf dem Smartphone läuft ein Prototyp dieser App. Der Versuch wird von einer Videokamera aufgenommen. Der Versuchsleiter Yannik Huber sitzt neben dem Probanden und gibt ihm Anweisungen.

Zusammenfassung des Think Alouds

Zunächst erklärte der Versuchsleiter dem Probanden, welche Funktion die App im fertigen Zustand erfüllen soll. Dann bat er Frau Huber, alle Gedanken frei auszusprechen.

Zunächst sollte sie die ersten Gedanken zur App loswerden. Frau Huber erklärte, welche Funktionen sie bei den dargestellten Elementen erwarten würde. So erkannte sie sofort,

dass die Karte den Standpunkt anzeigt, das Texteingabe Feld und der „Route suchen“ Button für die Wegbestimmung zuständig sind und der Alarm-Button zum Auslösen eines Alarms führt. Des weiteren erkannte sie, dass der Kontakt-Button zu den zu Kontakten führt, die benachrichtigt werden sollen. Auch die Funktion der anderen beiden Button „Einstellungen“ und „Angekommen“ erkannte sie sofort.

Als nächstes sollte der Proband die erste Aufgabe lösen. Die Aufgabe war es, zu den Kontakten zu navigieren und zu erklären, was die einzelnen Elemente im Kontakt-Layout zu bedeuten haben.

Das Navigieren zum Layout stellte kein Problem für Frau Huber dar. Daraufhin erklärte sie, dass sie glaubt, im Texteingabe-Feld einen Kontakt suchen zu können und diesen dann zu den vorhandenen Kontakten zuordnen kann. Sie verstand also direkt, welche Funktionen die Elemente in dem Layout haben. Des weiteren sagte sie, dass man dies sehr leicht verstehen kann, wenn man die App benutzt.

Danach zeigte der Versuchsleiter Frau Huber das Layout eines vorhandenen Kontaktes, für den man auch ein Notfallkontakt sein kann. Hier fragte er sie, ob die Farbgebung für „angekommen“, „unterwegs“ und „Alarm ausgelöst“ gut oder schlecht gewählt wurde.

Sie fand die Farbgebung sehr passend und hätte die Farben auch so ausgewählt.

Des weiteren fand sie das Layout sehr einfach zu verstehen und damit passend.

Die nächste Aufgabe war es, zurück auf den Startbildschirm zu navigieren und dort eine gewünschte Zieladresse einzugeben. Das Navigieren funktionierte ohne Probleme und auch die Eingabe der Adresse und das Suchen der Route funktionierte einwandfrei. Beim Eingeben der Adresse überlegte sie kurz, ob sie die Postleitzahl eingeben muss, machte es aber dann nicht und bekam trotzdem die richtige Route angezeigt. Was sie ein wenig störte war, dass man nicht die komplette Route angezeigt bekommen hat, bevor die Navigation startete. Das ist ein Punkt, der bei der Entwicklung definitiv berücksichtigt werden sollte.

Dann wurde Frau Huber gefragt, wie sie das Design des Startbildschirms im Hinblick auf mögliche Gefahrensituationen bewertet. Sie findet, dass der „Angekommen“-Button zu nah am „Alarm“-Button ist und würde die Position ändern. Sie schlägt vor, den Alarm-Button nach ganz unten zu verschieben, da man dort am schnellsten hingelangt. Die drei unteren Buttons für „Angekommen“, „Einstellungen“ und „Kontakte“ würde sie über der Karte anordnen.

Zuletzt wurde Frau Huber gefragt, wie sie die gewählten Symbole für die einzelnen Elemente findet. Sie fand alle Symbole sehr passend und würde nichts ändern.



Versuchsaufbau Proband 1 - Regina Huber

1.1.4.2.2 Think Aloud mit Anna Coen

Anna Coen war schon im Gestaltungsprozess involviert. Sie kennt also die Funktion der App und kennt die Einzelheiten. Der Versuchsaufbau war der gleiche wie beim ersten Proband.

Zusammenfassung des Think Alouds

Anna Coen konnte sich zunächst einen Überblick vom Prototypen verschaffen. Sie probierte sofort die „Routensuche“ aus und navigierte über die Karte. Was ihr sofort auffiel war, dass der Alarm-Button zu nah am Angekommen-Button ist. Bei einer Notfallsituation könnte man zu leicht versehentlich auf den falschen kommen. Der Versuchsleiter fragte den Proband ob es besser wäre, wenn der Button ganz unten angeordnet wäre und die drei anderen Buttons über der Karte, oder ob Frau Coen einen noch besseren Vorschlag hätte. Sie fand den genannten Vorschlag sehr gut.

Danach bekam sie die Aufgabe zu dem Kontakt-Layout zu navigieren und zu sagen, was sie von der Gestaltungslösung halte. Sie fand die Gestaltungslösung sehr gut, da sie „sehr einfach gestaltet ist“. Sie hat gefragt, ob alle Kontakte bei vorhandene Kontakte angezeigt werden oder nur bestimmte. Der Versuchsleiter sagte ihr, dass es die hinzugefügten Kontakte sind. Frau Coen wusste sofort, dass man die Kontakte im Textfeld oben suchen kann.

Danach wurde sie zu dem Layout, bei dem man einen bestimmten Kontakt verfolgen kann, befragt. Sie fand das Layout an sich sehr gut und übersichtlich, allerdings gefiel ihr auch

hier nicht, dass der „Notruf 110“-Button zu nah am „Kontakt anrufen“-Button ist. Sie würde den Notrufbutton ganz unten platzieren, den Status (angekommen, unterwegs & Alarm ausgelöst) darüber platzieren und über dem Status den „Kontakt anrufen“-Button.

Die Farbwahl der verschiedenen Statusmeldungen fand sie sehr gut.



Versuchsaufbau Proband 2 - Anna Coen

1.1.4.2.3 Think Aloud mit Anatol Walger

Über Anatol Walger

Anatol ist ein 22 Jahre alter Student der Medieninformatik und ist im Umgang mit Medien sehr begabt. Der Gebrauch von technischen Hilfsmitteln wie sein Smartphone sind aus seinem Alltag nicht mehr wegzudenken. Anatol geht öfter alleine von der Bahn nach Hause. Er findet die Idee von WalkHome sehr gut.

Zusammenfassung des Think Alouds

Zunächst sollte sich Anatol die App angucken und seine ersten Eindrücke schildern. Er fand das Design sehr übersichtlich und gelungen. Der Gesprächsleiter Yannik Huber stellte ihm die Frage, ob er an der Anordnung der Elemente etwas ändern würde. Darauf antwortete er, dass er eventuell die unteren drei Button weiter oben anordnen würde. Dies begründete er damit, dass diese nicht so wichtig sind wie zum Beispiel der Button für den Alarm. Außerdem findet er den „Angekommen-Button“ wichtiger als die Buttons Kontakte und Einstellungen. Daher würde er den Angekommen-Button größer machen.

Als nächstes bekam Anatol die Aufgabe ein Ziel einzugeben, zu dem er navigiert werden will. Er gab „Kölner Dom“ ein, anstatt ein übliches Adressformat. Die App zeigte ihm darauf hin die Route zum Kölner Dom, was er sehr gut fand.

Danach bekam er die Aufgabe zu den Kontakten zu navigieren und dort seine ersten Eindrücke zu äußern. Er fand das Design sehr aufgeräumt und übersichtlich. Seine erste Intention war, dass man im oberen Bereich neue Kontakte suchen kann und unten die vorhandenen Kontakte angezeigt werden. Er erkannte allerdings nicht sofort, dass man die Kontakte „scrollen“ kann. Das würde er sich im finalen Design besser erkenntlich wünschen. Die Icons fand er sehr gut und erkannte die Bedeutung aller sofort.

Auch das Design für das Verfolgen eines bestimmten Kontaktes fand er sehr gut, besonders die Karte mit der aktuellen GPS Position der Begleitperson. Was er allerdings ändern würde, ist die Größe der Schrift des Status. Diese ist genauso groß wie die Buttons zum Anrufen des Notrufs oder des Kontaktes. Daher würde er die Schriftgröße verkleinern. Den „Kontakt anrufen“-Button würde er farbig gestalten, um ihn etwas hervorzuheben.

1.1.4.2.4 Think Aloud mit Marcel Huber

Über Marcel Huber

Marcel Huber ist 56 Jahre alt und von Beruf Personalleiter. In seiner Freizeit ist er gerne sportlich aktiv und geht oft Joggen. Da er spät von der Arbeit kommt geht er sehr oft im Dunkeln joggen. Er benutzt regelmäßig während der Arbeit sein Smartphone, wodurch er ein erfahrender Nutzer ist. Der Versuchsaufbau ist der gleiche wie bei Regina Huber.

Zusammenfassung des Think Alouds

Zunächst machte sich Marcel mit der App vertraut. Er fand das Design sehr übersichtlich und fand sich sofort zurecht. Er probierte direkt aus ein Ziel einzugeben und die Route zu suchen. Obwohl er die Adresse sehr umständlich und ohne Formatierung eingegeben hat, funktionierte die Suchanfrage. Dies fand der Proband sehr gut. Die Aufteilung der Buttons fand er auch gut und würde nichts daran ändern.

Nun bekam Marcel die Aufgabe, auf das Layout der Kontakte zu wechseln. Dies klappte ohne Probleme. Als nächstes sollte er beschreiben, was er dort sieht und welche Funktion er hinter den Elementen vermutet. Er erklärte richtigerweise, dass es sich bei den vorhandene Kontakten um die schon hinzugefügten Kontakte handeln und dass es sich bei der Texteingabe im oberen Bereich des Layouts um eine Kontakt Suche handeln würde. Die Aufteilung fand er gut und würde hier nichts verändern.

Das Layout der einzelnen Kontakte, bei dem der aktuelle Status des Kontaktes angezeigt wird (zum Beispiel dass dieser gerade unterwegs ist oder einen Alarm ausgelöst hat) fand er auch gelungen. Er machte den Vorschlag, den Notruf-Button über den Statustext zu verschieben, da dieser vom Kontaktanrufen-Button getrennt sein soll. Die Begründung war, dass man sich sonst leicht vertippen könnte.

1.1.4.2.5 Zusammenfassung Think Aloud aller Probanden

Was die Probanden am meisten gestört hat, war die Aufteilung der Elemente im Startbildschirm. So fanden sie den Alarm-Button zu nah am Angekommen-Button platziert. Ein Vorschlag eines Probanden war, den Alarmknopf nach unten zu verlegen und die drei unteren Buttons über die Karte zu verschieben. Die Idee hatten auch andere Testpersonen. Des weiteren gab es den Vorschlag, dass die Route angezeigt werden sollte, bevor die Navigation startet. Die Symbole und gewählten Farben fanden die Probanden sehr gut gewählt und sehr leicht zu verstehen.

Auch insgesamt fanden sie das Design sehr leicht zu verstehen und sehr übersichtlich. Des weiteren kam heraus, dass man nicht sofort erkennen kann, dass man die vorhandenen Kontakte scrollen kann. Dies liegt aber daran, dass bei dem Prototypen schon 10 Kontakte vorhanden waren. Wenn man nun einzelne Kontakte hinzufügt, werden diese nacheinander in den vorhandenen Kontakten angeordnet. Dadurch wird diese Verwirrung nicht auftreten. Ein weiterer Punkt, bei dem sich die meisten Probanden einig waren, war die Aufteilung der Elemente im „Kontakt verfolgen“ Layout.

Hier fanden einige, dass der Button für den Notruf und der Button für den Anruf des Kontakts zu eng aneinander liegen. Außerdem wurde geäußert, dass der „Status“ des Kontakts („angekommen“, „unterwegs“, „Alarm ausgelöst“) auf dem „Kontakt verfolgen“ Layout etwas kleiner ausfallen sollte. Der Grund ist, dass die beiden „Anruf“-Buttons wichtiger sind.

Außerdem wurde vorgeschlagen, den „Kontakt anrufen“-Button farbig zu gestalten um ihn hervorzuheben. Die Suche nach der Route fanden die Probanden alle sehr gut. Besonders, dass man keine formatierte Adresse eingeben muss, sondern auch zum Beispiel „Kölner Dom“ eingeben kann, kam positiv an. Die Aufteilung der Elemente kam, bis auf die bereits genannten Problemen, sehr gut an. Insgesamt kann man also sagen, dass die potentiellen Benutzer die Gestaltungslösung sehr gut aufgenommen haben. Besonders die Übersichtlichkeit wurde hervorgehoben und es gab zu keiner Zeit Probleme bei der Benutzung des Prototypen. Ein Paar Schwachstellen sind jedoch aufgekommen, die es in der zweiten Iteration zu verbessern gilt.

1.2 Iteration 2

In der zweiten Iteration sind sowohl die Ergebnisse der Evaluation, als auch die im Laufe des Gestaltungsprozesses entstandenen Vorschläge für Verbesserungen zu beachten. Die Gestaltungslösung hatte ein paar Schwächen aber auch viele Stärken. Diese sind aus der Zusammenfassung des Think Alouds zu entnehmen.

1.2.1 Festlegen der Nutzungsanforderungen

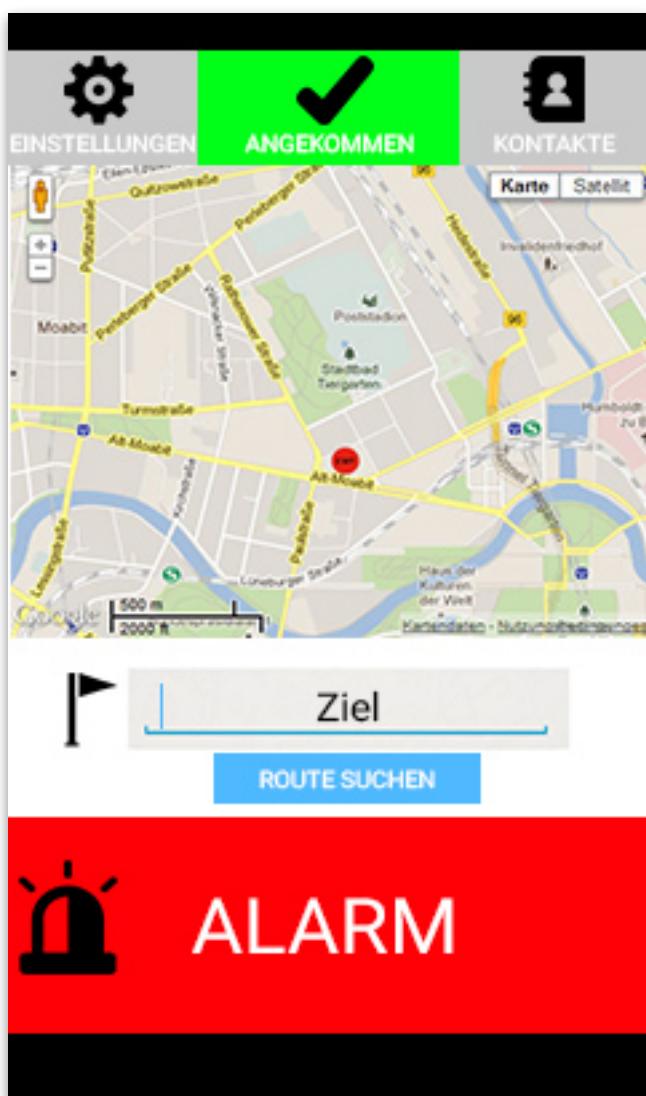
Die Befragung von Nicole Coen zur Gestaltung des Interface ergab einige gute Ansätze. Zum einen wurde vorgeschlagen, dass das Smartphone im Fall eines Notfalls vibrieren soll. Hierdurch soll der Nutzer bemerken, dass ein Alarm ausgelöst wurde. Des weiteren soll in diesem Fall die Taschenlampe blinke und ein lauter Ton abgespielt werden, wodurch Leute in der Umgebung aufmerksam werden könnten. Die funktionale Anforderung F50 zielte darauf ab, einen lauten Ton abzuspielen. Das Leuchten der Taschenlampe und das Vibrieren des Handys sind allerdings eine gute Erweiterung.

Zum anderen kam heraus, dass eine Art Reißeleine zum Smartphone helfen könnte einen Alarm auszulösen, wenn man es nicht mehr schafft auf das Display des Geräts zu drücken. Der Aux-Eingang eines Smartphones ist eine gute Schnittstelle, an der man dieses Feature umsetzen kann.

1.2.1.1 Hinzugefügte funktionale Anforderungen

- F290 Wenn ein Alarm ausgelöst wurde und keine Verbindung zu Internet oder Mobilfunk besteht, soll das Smartphone vibrieren.
- F300 Wenn ein Alarm ausgelöst wurde und keine Verbindung zu Internet oder Mobilfunk besteht, soll das System die Taschenlampe in einem bestimmten Rhythmus aktivieren/deaktivieren.
- F310 Wenn der Kopfhörer in den AUX-Eingang ein oder aus gesteckt wird, soll das System das erkennen.

1.2.2 Erarbeiten von Gestaltungslösungen zur Erfüllung der Nutzungsanforderungen



Die Gestaltungslösung muss anhand der in der Evaluation herausgefundenen Aspekte angepasst werden.

Gestaltungslösung 2 - Homescreen

So wurde nun das Design des „HomeScreens“ angepasst. Die Buttons „Kontakte“, „Einstellungen“ und „Angekommen“ wurde über die „Karte“ verschoben. Der AlarmButton wurde nach unten verschoben und die „Karte“ und „Ziel-Suche“ ist nun in der Mitte. Des weiteren wurde der „Angekommen-Button“ vergrößert. Ansonsten hat sich im Design nichts verändert, da es hierfür keine Notwendigkeit gab.

Gestaltlösung 2 - Kontakt



Bei dem Kontakt Layout, wurde die Anordnung der Elemente geändert. Der „Kontakt anrufen“-Button ist nun über dem „Status“ angeordnet. Dieser ist nun farbig gestaltet, wodurch er sich aus dem Rest des Layouts hervorhebt. Die Schriftgröße des Status ist nun kleiner. Dadurch sind die beiden „Anruf-Button“, welche die wichtigsten Komponenten des Layouts sind, im Vordergrund.

1.2.3 Evaluieren der Gestaltungslösungen anhand der Anforderungen

In der ersten Iteration konnten durch die Think Alouds qualitative Evaluationsergebnisse ermittelt werden. Hierdurch konnte die Gestaltungslösung gezielt verbessert werden. Nun ist es wichtig, auch quantitative Evaluationsergebnisse zu ermitteln. Um dies zu erreichen, wird das Attrakdiff eingesetzt. Die Probanden sollten sich zuerst die Gestaltungslösung ansehen und danach das Attrakdiff ausfüllen. Hierbei werden gegensätzliche Adjektive gegenüber gestellt und der Proband soll entscheiden, ob eher das eine oder das andere Adjektiv zur Gestaltungslösung passt. Dabei gibt es sieben Abstufungen zwischen den beiden Adjektiven, die man auswählen kann. Man kann sich also auch für keins von beiden entscheiden.

Bei der Umfrage haben 20 Personen teilgenommen. Darunter waren 35% im Alter von unter 20 Jahren, 45% im Alter zwischen 20 und 40 Jahren und 20% im Alter von 40-60

Jahren. Es haben also etwas mehr Jugendliche und junge Erwachsene teilgenommen. Da die Zielgruppe in einem Alter von 14-60 Jahren liegt, passen die Teilnehmer, bezogen auf das Alter, zum Profil der Nutzermodellierung. Bezogen auf das Geschlecht, waren 55% der Teilnehmer männlich und dem entsprechend 45% weiblich. Die Aufteilung ist sehr ausgeglichen mit einem leichten Überwiegen der männlichen Teilnehmer. Auch hier passt die Aufteilung sehr gut, da die App nicht speziell für eines der beiden Geschlechter entwickelt wird. Ein Anteil von 40% der Befragten haben einen Beruf angegeben. Die angegebenen Berufe waren Schüler, Student, Personalleiter, Erzieherin und Hotelkaufmann.

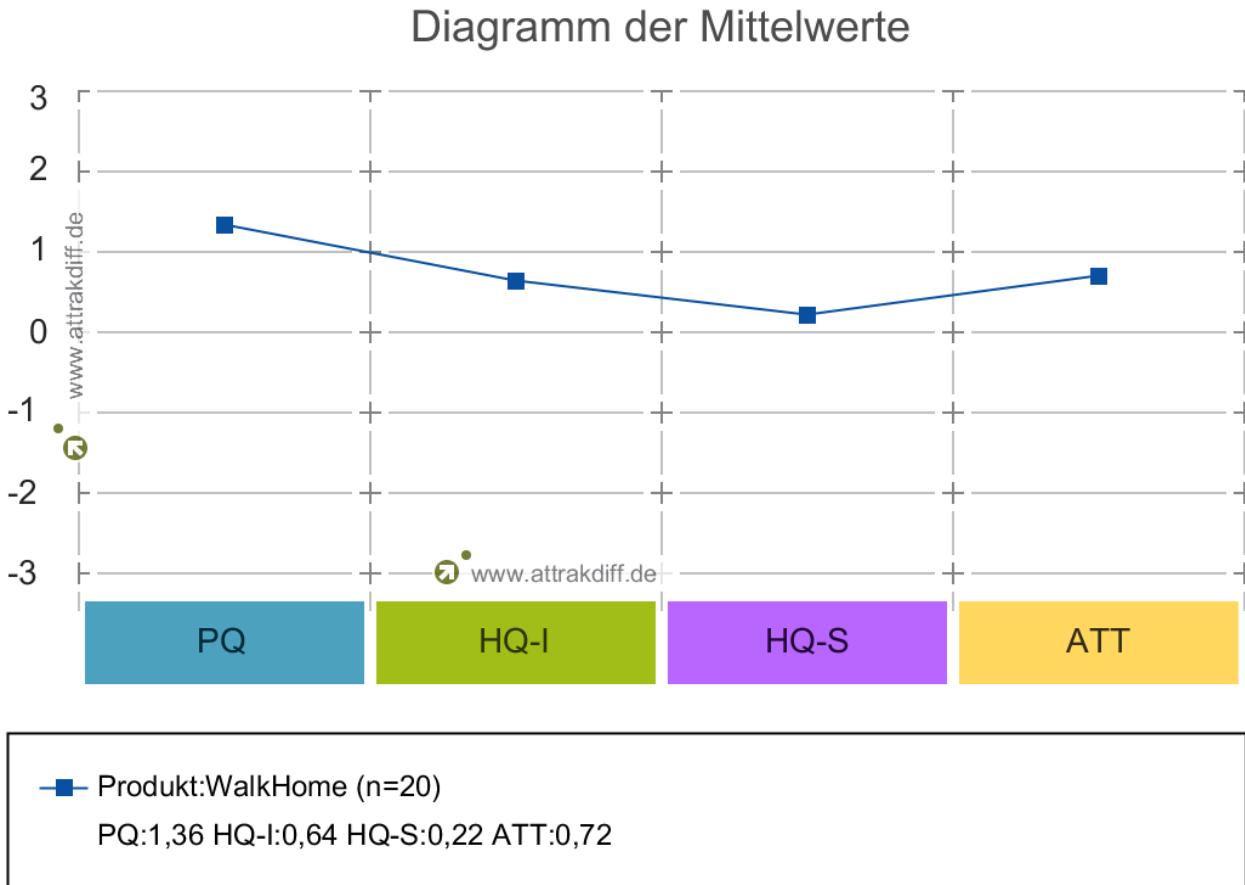
Die Auswertung ergab, dass das Design handlungsorientiert ist. Es hat eine hohe pragmatische Qualität, was auch der Ziel des Gestaltungsprozesses war. Die Anwendung sollte vor allem pragmatisch sein, denn besonders in Gefahrensituationen ist ein unkompliziertes Design wichtig. Die hedonische Qualität ist über dem Mittelpunkt angeordnet und daher positiv zu sehen. Die hedonische Qualität besteht aus Stimulation und Identität. Ein sehr hoher Wert wäre hier wünschenswert, ist aber nicht das Hauptziel.

Portfolio-Darstellung



- Produkt:WalkHome (n=20)
 - PQ:1,36 Konfidenz:0,32
 - HQ:0,43 Konfidenz:0,21

Beim Diagramm der Mittelwerte kann man noch einmal die pragmatische Qualität und die hedonische Qualität in Zahlen sehen. Außerdem wird die Attraktivität dargestellt. Hier sieht man, dass die Stimulation den geringsten Wert hat, die Pragmatik den höchsten Wert und Identität dazwischen liegt. Die Attraktivität liegt im positiven Bereich, was als ein Erfolg zu werten ist.

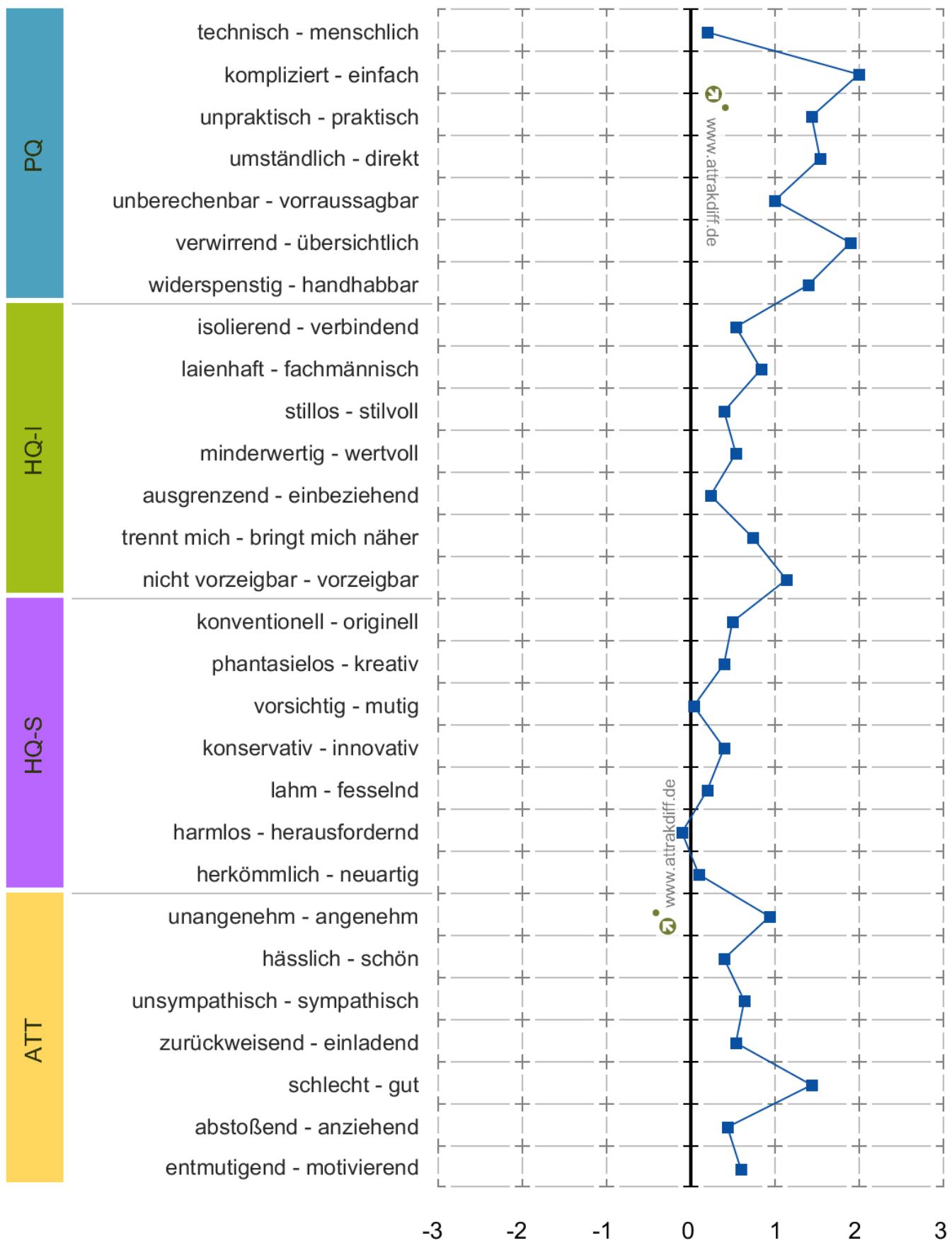


Im Profil der Wortpaare sieht man, welche Adjektive die potentiellen Nutzer gewählt haben. Hier kann man erkennen, dass besonders die Adjektive „einfach“, „praktisch“, „übersichtlich“, „vorzeigbar“ und „gut“ gewählt wurden. Diese Adjektive sind sehr wichtig für unser Projekt, da Nutzer auf Übersichtlichkeit und Einfachheit angewiesen sind. Auch „handhabbar“, „direkt“, „angenehm“ und „voraussagbar“ wurde oft von den Nutzern gewählt. Dies sind weitere Punkte die für die Gestaltungslösung sprechen. Adjektiv-Paarungen die eher neutral gesehen wurde, waren „technisch-menschlich“, „vorsichtig-mutig“, „harmlos-herausfordernd“ und „herkömmlich-neuartig“.

Auch wenn es wünschenswert gewesen wäre, wenn diese Adjektive positiver ausgefallen wären, sind es nicht die primären Ziele unserer Gestaltungslösung.

Zusammenfassend spiegeln die Evaluationsergebnisse die erhofften Erwartungen des Designtools an die Gestaltungslösung wider. Besonders der Fakt, dass die Nutzer das Design übersichtlich und einfach finden, ist eine Bestätigung der Gestaltungslösung. Auch wenn die Stimulation des Designs hätte höher ausfallen können, ist das Design besonders in Hinsicht auf das Einsatzgebiet gut gelungen.

Profil der Wortpaare



■ Produkt:WalkHome (n=20)

2. Systemdokumentation - WBA

Die Systemdokumentation ergänzt das Konzept welches in den Anlagen zu finden ist und geht auf die für das System wichtigen Punkte in der Entwicklung ein. Da sich im Entwicklungsprozess einige Bestandteile geändert haben werden auch Änderungen hier dokumentiert und begründet.

2.1 Client-Server Architektur

Client-Server Architektur wird verwendet um einen Teil der Anwendungslogik auszulagern. Funktionen wie das Alarm verteilen sind vorteilhaft wenn sie von einem zentralen Punkt versendet werden als wenn Sie von einem Client der vermutlich eine schlechte Internetverbindung hat verteilt werden müssen. Außerdem findet die Überprüfung von Standorten für die Funktion „Leute in der Umgebung finden“ statt. Andere Architekturen wie eine Client zu Client Verbindung wurden deshalb abgewägt. Der Webserver stellt nur Dienste für die Anwendung WalkHome zur Verfügung und soll erst mal nicht für dritte verfügbar sein.

2.2 Webserver

Da es sich bei der Anwendung um ein verteiltes System handelt läuft ein Teil des Systems auf einem Server. Um das Testen der Anwendung etwas leichter zu machen und um das Testen mit reellen Bedingungen zu gewährleisten und das testen mit echter Hardware zu ermöglichen wird der Server auf einem V-Server aufgesetzt. Dieser ist unter www.walkhome.de oder alternativ unter 5.199.129.74:81 erreichbar. Auf eine authentifizieren wird aus einfachheitshalber verzichtet, da alle Daten nur automatisch über die AndroidClientID abgerufen werden.

2.3 Ressourcentabelle

Der Note.JS Server stellt einige HTTP Ressourcen zur Verfügung, welche in der Folgenden Tabelle aufgelistet sind. Welche Datentypen der Server entgegen nimmt wird im Punkt Datenstrukturen ausführlich beschreiben.

Ressource	HTTP Verb	Beschreibung	Req	Res
/user	POST	Legt eine neuen USER an und gibt	application/json	application/json
/user/:ID	GET	Gibt eine Repräsentation der USER Ressource zurück	text/plain	application/json
/user/:ID/kontakt	POST	Legt einen neuen Kontakt zu einem bestimmten USER an	application/json	application/json
/user/:ID/kontakte	GET	Gibt eine Repräsentation aller Kontakte einer USER Ressource zurück	text/plain	application/json
/user/:ID/kontakt/:ID	GET	Gibt eine Repräsentation eines Kontaktes einer USER Ressource zurück	text/plain	application/json
/user/:ID/kontakt/:ID	PUT	Ändert die Kontaktdata eines Kontaktes eines bestimmten USER	application/json	application/json
/user/:ID/kontakt/:ID	DELETE	Löscht einen Kontakt eines bestimmten USER	text/plain	text/plain
/user/:ID/alarm	POST	Legt einen neuen Alarm zu einem bestimmten USER an.	application/json	application/json
/user/:ID/alarme	GET	Gibt eine Repräsentation aller Alarne einer USER Ressource zurück	text/plain	application/json
/user/:ID/alarm/:ID	GET	Gibt eine Repräsentation eines Alarmes einer USER Ressource zurück	text/plain	application/json
/user/:ID/alarm/:ID	PUT	Ändert die Alarndata eines Alarmes eines bestimmten USER	application/json	application/json
/user/:ID/alarm/:ID	DELETE	Löscht einen Alarm eines bestimmten USER	text/plain	text/plain

2.4 Datenstrukturen

Datenstrukturen werden festgelegt um mögliche falsch eingaben ins System und somit Fehlerquellen zu vermeiden. Datenstrukturen können zum überprüfen im Code verwendet werden und bevor mit beispielsweise empfangen Daten gearbeitet wird, werden diese auf Richtigkeit und Vollständigkeit überprüft.

JSON-Daten:

Die User bezogenen Daten bestehen aus den folgenden Datensätzen:

- **AndroidClientID** ist die ID die das Smartphone des Users besitzt. Diese ist eindeutig auf jedem Smartphone und wird dazu verwendet User von WalkHome eindeutig zu identifizieren.
- **Name** ist der Name der Realen Person die sich im System anmeldet
- **Adresse** ist die Adresse des Anwender. Sie wird gespeichert um den weg nach Hause schnell einzugeben.
- **Status** wird benutzt um zu speichern ob der Nutzer unterwegs, ob er gut zuhause angekommen ist oder ob er einen Alarm ausgelöst hat.

Die Kontakt bezogenen Daten bestehen aus den folgenden Datensätzen:

- **AndroidClientID** ist die ID die das Smartphone des Users besitzt. Diese ist eindeutig auf jedem Smartphone und wird dazu verwendet Kontakte von WalkHome eindeutig zu identifizieren.

- **FCMID** speichert die ID über die Firebase Cloud Messaging braucht um mit diesem Client zu kommunizieren
- **Name** ist der Name wie des Kontaktes
- **Nummer** ist die Telefon Nummer des Kontaktes diese kann wird im Notfall verwendet im den Kontakt zusätzlich zu benachrichtigen.
- **Status** hier wird festgelegt ob es ein normaler Alarmkontakt ist oder ob der Kontakt gesondert benachrichtigt wird.

Die Alarm bezogenen Daten bestehen aus den folgenden Datensätzen:

- **AlarmID** ist eine fortlaufende Nummer die Alarne eindeutig unterscheidet.
- **Datum** gibt den genauen Tag und die Uhrzeit an wann der Alarm gemeldet wurde.
- **Standort** ist der Standort wo der Alarm ausgelöst wurde.
- **Status** gibt an ob der Alarm Aktiv oder Inaktiv ist.

Die vorher aufgelisteten Daten werden wie folgt im JSON Format vom Client zum Server versendet.

```
User {
    "AndroidClientID" : "number",
    "Name" : "string",
    "Adresse" : "string"
    "Status" : "number"
}
```

```
Kontakt {
    "AndroidClientID" : "number",
    "FCMID" : "string",
    "Name" : "string",
    "Nummer" : "string"
    "Status" : "number"
}
```

```
Alarm {
    "AlarmID" : "number",
    "Datum" : "date",
    "Standort" : "number"
    "Status" : "number"
}
```

2.5 Anwendungslogik

Die im für das System ausschlaggebenden Funktionen wurden aus den Funktionalen Anforderungen des MCI Teil entnommen. Die für das System wichtigsten Funktionen werden hier ausführlich beschreiben und mit Pseudocode und Screenshot beschreiben. Zudem sind Kommentare im Code zu finden die Codeteile so gut wie möglich lesbar machen sollen.

2.5.1 Android

Abweichungen erkennen

Um zu berechnen, ob der Nutzer von der vorher festgelegten Route abkommt, muss ein Algorithmus programmiert werden. Die Routenbestimmung wird von den Google-Servern übernommen. Bei einer Anfrage an den Server mit dem Startpunkt und Ziel, bekommt man einige Informationen in der Form von JSON. Ein Teil davon ist ein Array, welches auf der Route liegende Koordinaten beinhaltet. Diese Koordinaten bilden eine gute Ausgangslage. Eine Idee, die Abweichung des Smartphones von der Route zu erkennen, ist die Entfernung des aktuellen Standpunktes zur am nächsten liegenden Koordinate der Route zu berechnen. Diese Methode wird bei jeder Standortänderung seitens des Smartphones aufgerufen. Dabei wird das Array, welches die Koordinaten der Route beinhaltet, durchlaufen und es wird berechnet, ob die Entfernung vom aktuellen Standort zu dieser Koordinate mehr als 50 Meter beträgt. Bei der ersten Koordinate, die mehr als 50 Meter entfernt ist, wird die Nachfrage an den Nutzer ausgelöst. In dieser Nachfrage wird der Nutzer gefragt, ob alles in Ordnung ist. Wenn die Schleife das ganze Array durchläuft und keine Koordinate mehr als 50 Meter vom aktuellen Standort entfernt ist, läuft die App normal weiter.

Da die Koordinaten einige Meter auseinander liegen, ist diese Berechnung nicht sehr genau. Denn man könnte theoretisch auf der Route gehen aber 50 Meter von der letzten und nächsten Koordinate entfernt sein. Deswegen wird das Koordinaten-Array angepasst. Ein Algorithmus durchläuft das Array und fügt weitere Koordinatenpunkte zwischen den vorhandenen Koordinaten hinzu. Hierbei berechnet eine Methode einen Koordinatenpunkt zwischen den aufeinander folgenden Koordinatenpunkten und speichert diese im Array. Dieses Verfahren kann so oft wie nötig wiederholt werden, um der Route noch mehr Koordinatenpunkte hinzuzufügen. Hier ist allerdings zu beachten, dass das die Arraygröße bei jeder Wiederholung des Verfahrens exponentiell steigt. Zum einen dauert dadurch das Erweitern des Arrays um einiges länger. Zum anderen muss dieses Array bei jeder Standortänderung, wegen der Berechnung der Abweichung, durchlaufen werden. Dadurch könnte die Laufzeit der Methoden zu stark steigen, was zu Fehlern führen kann.

Pseudo Code - Hinzufügen von Koordinaten:

```
//neues Array für aktualisierte Koordinaten erstellen  
Array ArrayKoordinatenNeu;  
  
//Array, welches Koordinaten der Route enthält, durchlaufen  
for(ArrayKoordinatenAlt.Länge){  
  
    //aktuell durchlaufene Koordinate dem Neuen Array hinzufügen  
    ArrayKoordinatenNeu.hinzufügen(ArrayKoordinatenAlt(aktuell));  
    //Koordinate in der Mitte von der aktuellen Koordinate und der Nachfolgenden  
    //berechnen  
    KoordinateNeu = (ArrayKoordinatenAlt(aktuell) + ArrayKoordinatenAlt(aktuell+1))/2;  
    //errechnete Koordinate dem neuen Array hinzufügen  
    ArrayKoordinatenNeu.hinzufügen(KoordinateNeu);  
}
```

Pseudo Code - Abweichung der Route berechnen

```
//Counter für den Anzahl der Durchläufe erstellen  
Zahl counter;  
  
//Bei jeder GPS-Standortaktualisierung wird diese Methode aufgerufen  
Methode beiStandortaktualisierung{  
    //das Array, welches alle Koordinaten enthält, wird durchlaufen  
    for(ArrayKoordinatenNeu.Länge){  
        //wenn der Abstand der Koordinate zum aktuellen Standort kleiner als 50  
        Meter ist dann..  
        if(abstand(ArrayKoordinatenNeu(aktuell), aktuellerStandort) < 50){  
            // der User befindet sich auf der Route -> Counter wird auf null gesetzt  
            counter = 0;  
            break  
        }  
    }else{  
        //Counter wird um 1 erhöht  
        counter++;  
        //Wenn Counter der Länge des KoordinatenArrays entspricht, wurden  
        //alle Koordinaten durchlaufen und keine befindet sich auf der Route  
        if(counter == ArrayKoordinatenNeu.Länge){  
            Dialogfeld(„Ist alles In Ordnung“){  
                if(nein) Alarm versenden;  
                if(ja) counter = 0;  
            }  
        }  
    }  
}
```

„Stehen“/„Liegen bleiben“ erkennen

Die App muss erkennen können, ob sich der GPS-Standort des Smartphones für einen gewissen Zeitraum nicht verändert hat. Um dies zu erreichen, wird bei jeder Standortänderung seitens des Smartphones geprüft, ob sich die neue Position um weniger als x Meter geändert hat. Wenn dies mehrfach hintereinander passiert, wird der Nutzer gefragt, ob alles in Ordnung ist. Hierzu wird ein Counter verwendet, der bei jedem mal um „1“ erhöht wird. Falls die Position um mehr als x Meter geändert wurde, wird der Counter wieder auf „0“ gesetzt. Wenn der Nutzer nun absichtlich stehen geblieben ist, kann er die Frage, ob alles in Ordnung ist, mit „ja“ beantworten. In diesem Falle wird der Counter auf x+1 gesetzt. Dadurch wird er nicht alle paar Sekunden erneut gefragt ob er stehengeblieben ist. Wenn der Nutzer dann weiter geht, verändert sich die Position und der Counter wird dadurch wieder auf „0“ gesetzt. Hierdurch ist die Funktion wieder aktiv.

Pseudo Code - „Stehen“/„Liegen bleiben“ erkennen

Zahl counter;

```
//wird bei jeder GPS-Standortaktualisierung aufgerufen
Methode beiStandortaktualisierung{
    if(positionsÄnderung == false){
        counter++;
        if(counter == 7){

            Dialogfeld(„Ist alles in Ordnung“){
                if(nein) Alarm versenden;
                if(ja) counter = 8;
            }
        }
    }else{
        counter = 0;
    }
}
```

Alarm versenden

Zum melden eines Alarms vom Android Client an den Server wird ein HTTP post abgesetzt. Dieser Post wird vom Server entgegengenommen und Automatisch an die Alarmkontakte weiter geleitet. Um einen HTTP post abzusetzen wird das OKHttp Framework benutzt welches es schnell und effizient macht einen Post vom Gerät abzusetzen. Der Post enthält einen JSON String mit den Alarm Informationen welche ausführlich im Punkt Datenstrukturen erläutert werden.

Reisleine (Kopfhörer)

Für den Fall das keine Möglichkeit besteht das Smartphone aus der Tasche zu holen und einen Alarm anzusetzen, kann die in der App die Funktion eingestellt werden das der Kopfhörer Anschluss wie eine Reisleine überwacht wird. Die Funktion überprüft ob ein Kopfhörer über den 3,5 mm Klinke Buchse angeschlossen ist. Sobald der Anwender nur den Kopfhörer Anschluss rauszieht wird nun ein Alarm versendet. Die Funktion ist dann hilfreich wenn der Anwender auf der Route ist und nicht stehen bleibt aber denn noch schnell einen Alarm absetzen möchte.

2.5.2 Node Server

Alarm verteilen

Alarne die von einem Client an der Server geschickt werden werden automatisch verarbeitet. Sobald ein Alarm über eine HTTP Post bei Server eintrifft wird dieser gespeichert. Es der Alarm wird unter der User Ressource untergeordnet, somit kann der Server die Kontakte vom ansprechenden User suchen und den Alarm an diese

Weiterleiten. Die Alarm Verteilung vom Server an die Alarm Clients wird über den Firebase Cloud Messaging Dienst realisiert. Die Voraussetzung dafür ist, dass die Client-Benutzer der App sind und somit auch dem Server bekannt sind und mit dem Cloud Dienst registriert sind. Wenn dies der Fall ist, wird der Alarm an alle bekannten Alarmkontakte des Users via Push Benachrichtigung aufs Handy geschickt.

2.6 Schnittstellen

2.6.1 Push Dienste

```
var fcm = new FCM(serverKey);

var message = {
  to: FCMClientID,
  collapse_key: 'Alarm' + AndroClientID,

  notification: {
    title: 'ALARM bei Kontakt' + AndroClientID
  },

  data: {
    info: 'Alarm!',
    name: 'Name'
    location: 'GPSData'
  }
};
```

Um möglichst schnell und mit wenig Datenverkehr Alarmkontakte zu informieren, wenn ein Alarm ausgelöst wird, werden sogenannte Push Benachrichtigungen gesendet. Beim Push-Verfahren wird die Verbindung zum Server stetig gehalten, um eintreffende Alarms direkt zu übermitteln. Wenn die Anwendung nicht über dieses Technik mit dem Server verbunden wäre, könnte sie über das Polling-Verfahren den Server in regelmäßigen Abständen anfragen, ob ein Alarm vorhanden ist. Das Polling-Verfahren hat den Nachteil, dass wenn ein Alarm unmittelbar nach dem Intervall eintrifft, dieser erst erkannt wird, wenn der Server das nächste Mal angefragt wird.

2.6.2 Kartenmaterial

Die Hauptfunktion des Systems ist es, Abweichungen auf Routen auf dem Weg nach Hause zu erkennen. Für diese Funktion muss eine Route festgelegt werden und diese zusammen mit den aktuellen GPS-Daten abgeglichen werden. Um die Route festzulegen und um diese auch zu veranschaulichen, wird Kartenmaterial benötigt.

2.6.3 Serverlogik

Für das Bereitstellen der Daten und für die Serverlogik wird ein Node.js-Server verwendet. Node.js ist eine auf JavaScript basierte Open-Source-Plattform, welche eine große Reihe an Modulen bereitstellt.

Verwendete Module:

- Express ist ein Node.JS Framework. Dieses Framework ermöglicht es einfacher mit und schneller einen Webserver aufzusetzen. Es wird im System verwendet um die HTTP Routen festzulegen und um das Routing zu unterschiedlichen Ressourcen zu ermöglichen.
- Body-Parser dieses Modul ermöglicht es einfach JSON Strings zu interpretieren und in einem Objekt wie ein Array anzusprechen. Somit wird das Modul verwendet um die eingehende JSON Datenstruktur für die Weiterarbeit in Javascript zu übersetzen.
- Redis wird benötigt um den Zugriff zur Redis Datenbank aufzubauen. Das Modul ist die Schnittstelle zwischen Node.JS und RedisDB.
- FCM / Firebase Cloud Messaging ist ein von Google bereitgestellter Push Dienst. Das Modul ist die Node Schnittstelle zum Google Dienst.

2.6.4 Datenhaltung

Bestimmte Daten wie Userdaten, Kontaktinfos oder Alarminfos müssen dauerhaft gespeichert werden und auf Abruf verfügbar sein. Zum speichern dieser Daten wird eine Redis Datenbank verwendet. Redis ist eine Key-Value Datenbank und gehört zu den NoSQL Datenbanken. Diese Datenbank eignet sich gut für schnelles aufrufen von Werten die nicht in komplexen Datenstrukturen verpackt sind.

Alle Daten die in Redis gespeichert werden werden dem User untergeordnet um die Daten eindeutig zu zuweisen. User werden eindeutig über die ID welche wie in den Datenstrukturen festgelegt die AndroidClientID ist.

get user:id //bestimmte User Daten ausgeben

Ein Alarm ist eine unter Resource eines Users und wird zu einem bestimmten Benutzer zugewiesen. Jeder Alarm hat eine eigene ID welche eine fortlaufende Nummer ist. Jeder User kann unbegrenzt viele Alarne auslösen.

get sser:id:alarm:id //einen bestimmten Alarm anzeigen eines Users ausgeben

Kontakte werden ähnlich wie der Alarm dem User untergeordnet. Kontakte werden durch ihre AndroidClientID eindeutig identifiziert. Jeder User kann unbegrenzt viele Kontakte haben.

get user:id:kontakt:id // einen bestimmten Kontakt eines Users ausgeben

2.7 Probleme

- Verwendung des Android Emulator von Android Studio: Beim testen des Prototypen gab es Schwierigkeiten mit den Google Play Diensten und der Version des Android Studio Emulators. Außerdem dauert das Starten des Emulators sehr lange. Die Lösung des Problems war zum einen die Verwendung von „multidex“ und zum anderen der Wechsel auf den Genymotion Emulator. Außerdem funktionierten die Tests auf einem Android Gerät später auch.
- Kommunikation aus dem Android Emulator mit dem Lokalen Node Server: Es gab Schwierigkeiten aus dem Android Emulator auf den Lokalen Server zu verbinden. Deshalb wurde der Node Server auf einen V-Server ausgelagert wie im Punkt Webserver beschreiben.

3 Quellen

3.1 MCI-Teil:

DIN EN ISO 9241-210, stand 2011

DIN EN ISO 9341-110, stand 2011

Use Cases:

Writing Effektive Use Cases, Alistair Cockburn, 2001

Die Benutzer:

<https://de.statista.com/statistik/daten/studie/152713/umfrage/verteilung-der-opfer-von-koerperverletzung-nach-altersgruppen/> 14.11 - 15 Uhr

<https://de.statista.com/statistik/daten/studie/152661/umfrage/verteilung-der-opfer-von-sexualdelikten-nach-altersgruppen/> 14.11 - 15 Uhr

1.2 WBA-Teil:

- <http://square.github.io/okhttp/>
Zuletzt gesichtet am 03.12.2016

- <https://developer.android.com/studio/build/multidex.html>
Zuletzt gesichtet am 03.12.2016

- <http://stacktips.com/tutorials/android/get-device-id-example-in-android>
Zuletzt gesichtet am 03.12.2016

- <https://firebase.google.com/docs/cloud-messaging/android/client>

Zuletzt gesichtet am 03.12.2016

- <https://inthecheesefactory.com/blog/how-to-install-google-services-on-genymotion/en>

Zuletzt gesichtet am 03.12.2016