

Bachelorarbeit

Jakob Rohwer

Konzeption und Evaluation eines Systems zur Nutzung
eines Smart Wearables für die Unterstützung des
Regelrettungsdiensts

Jakob Rohwer

Thema der Arbeit

Konzeption und Evaluation eines Systems zur Nutzung eines Smart Wearables für die Unterstützung des Regelrettungsdienstes

Stichworte

Smart Wearables, Notfallrettung, Datenintegration, Prototyp

Kurzzusammenfassung

Diese Bachelorarbeit befasst sich mit der Konzeption und Evaluation eines Systems zur Nutzung eines Smart Wearables zur Unterstützung des Regelrettungsdienstes. Die zentrale These dieser Arbeit ist, dass die Integration von Daten aus Smart Wearables in die gängigen Handlungsabläufe des Rettungsdienstes die Erstbeurteilung oder Verdachtsdiagnose von Notfallpatienten unterstützen kann. Zur Überprüfung dieser These wird zunächst die Eignung von Smart Wearables für den Rettungsdienst analysiert sowie potenzielle Anwendungsfälle identifiziert. Darauf aufbauend wird untersucht, wie die Smart Wearable-Daten in die Handlungsabläufe des Rettungsdienstes integriert werden können. Ein Konzept wird entwickelt, um die Daten von Smart Wearables in ein Rettungsszenario zu integrieren, und ein Prototyp wird erstellt, um das Konzept in ausgewählten Rettungsszenarien zu testen. Die Bewertung durch Experten dient der Überprüfung der These und der Bewertung des Konzepts. Die Ergebnisse zeigen, dass die Daten von Smart Wearables eine Verdachtsdiagnose unterstützen können, aber keine relevanten Informationen für eine Erstbeurteilung und Therapie beinhalten. Das Konzept zeigt brauchbare Ansätze für die Verarbeitung und Kommunikation der Daten.

Jakob Rohwer

Title of Thesis

Design and evaluation of a system for the use of a smart wearable for the support of the regular rescue service

Keywords

Smart Wearables, Emergency rescue, Data integration, Prototype

Abstract

This bachelor thesis deals with the conception and evaluation of a system for the use of a smart wearable to support the regular rescue service. The central thesis of this work is that the integration of data from smart wearables into the common action procedures of the rescue service can support the initial assessment or suspected diagnosis of emergency patients. To test this thesis, the suitability of smart wearables for emergency services is first analyzed and potential use cases are identified. Based on this, it will be investigated how the smart wearable data can be integrated into the action procedures of the rescue service. A concept will be developed to integrate smart wearables data into a rescue scenario, and a prototype will be created to test the concept in selected rescue scenarios. Expert evaluation is conducted to verify the thesis and evaluate the concept. The results show that data from smart wearables can support a tentative diagnosis but do not contain relevant information for initial assessment and therapy. The concept shows usable approaches for processing and communicating the data.

Inhaltsverzeichnis

Abbildungsverzeichnis	ix
Tabellenverzeichnis	xi
Abkürzungen	xii
Glossar	xiii
1 Einleitung	1
1.1 Konkretisierung der Aufgabenstellung	2
1.2 Aufbau der Arbeit	2
2 Grundlagen	4
2.1 Überblick Smart Wearables	4
2.1.1 Sensoren von Lifestyle Smart Wearables	5
2.1.2 Daten von Lifestyle Wearables	6
2.2 Abläufe im Rettungsdienst	8
2.2.1 ABCDE Schema	9
2.2.2 SAMPLE Schema	10
2.2.3 4Hs & HITS	10
3 Konzeptentwicklung	11
3.1 Anforderungsanalyse	12
3.1.1 Experteninterviews	13
3.1.2 Ergebnisse der Experteninterviews	13
3.1.3 Workshops	15
3.1.4 Ergebnisse der Workshops	16
3.2 Anforderungsdokumentation	23
3.3 Teilfunktionen	25
3.3.1 Lösungsauswahl	27

3.3.2	Ausgewählte Lösungsvarianten	32
4	Realisierung	35
4.1	Auswahl eines Smart Wearable	35
4.2	Kommunikation über Fitbit API	36
4.3	Implementierung eines Programms zur Datenerfassung und -verarbeitung	37
4.4	Entwicklung eines GUI Prototyps	40
5	Evaluierung	45
5.1	Gestaltung und Durchführung der Testszenarien	45
5.2	Auswertung der Testszenarien	47
5.3	Überprüfen der Anforderungen	51
5.4	Diskussion	54
5.4.1	Bewertung des Konzepts	54
5.4.2	Diskussion der These	56
6	Zusammenfassung und Ausblick	58
6.1	Zusammenfassung	58
6.2	Ausblick	59
	Literaturverzeichnis	62
A	Anhang	66
A.1	Aufgabenstellung	67
A.2	Experteninterviews	68
A.2.1	Fragen	68
A.2.2	Interview Protokoll	69
A.3	Workshop Ergebnisse	71
A.3.1	Flipcharts	71
A.3.2	Visualisierung der Daten für ausgewählte Fallbeispiele	75
A.4	Quellcode	76
A.4.1	Quellcode fitbit_data_for_EMS_usage.py	76
A.4.2	Ergänzungen in python-fitbit Bibliothek	90
	Selbstständigkeitserklärung	91

Abbildungsverzeichnis

3.1	Produktgestaltung nach Pahl und Beitz (Ausschnitt) Iterativer Ansatz und Ablauf nach Phasen geordnet. [5]	12
3.2	Verallgemeinerte Abläufe von Fallbeispielen unterschieden in Patient kommunikationsfähig/nicht kommunikationsfähig. Mit Sonderfall Reanimation	18
3.3	Einordnung von Smart Wearable Daten (Rot) in die Abläufe der im Workshop dargestellten Fallbeispiele	22
3.4	Funktionsstruktur des Konzepts nach dem „Input-Output“ Modell	26
3.5	Punktebewertung für die Teilfunktion Datenzugriff erlangen	27
3.6	Punktebewertung für die Teilfunktion Daten sammeln	28
3.7	Punktebewertung für die Teilfunktion Daten abspeichern	28
3.8	Punktebewertung für die Teilfunktion Auffälligkeiten erkennen	28
3.9	Punktebewertung für die Teilfunktion Informationen aufbereiten	29
3.10	Punktebewertung für die Teilfunktion Informationen an Rettungsdienstpersonal kommunizieren	29
3.11	Punktebewertung für die Teilfunktion Informationen an Krankenhaus kommunizieren	29
3.12	Morphologischer Kasten mit Lösungsauswahl	31
3.13	Systemkontext - Lösungsvariante für ein konkretes Smart Wearable zur Integration von Smart Wearable Daten in den Rettungsdienst	33
3.14	Systemkontext - Optimale Lösungsvariante zur Integration von Smart Wearable Daten in den Rettungsdienst	34
4.1	Ausgewähltes Smart Wearable: Fitbit Charge 5 [10]	36
4.2	Datenfluss von Gerät zu Python Applikation über Fitbit API	37
4.3	Input-Output Sicht der im Programm abgebildeten Funktionen aus der Funktionsstruktur 3.4	38
4.4	Klassendiagramm des entwickelten Programms für das Einlesen, Verarbeiten und Darstellen der Daten eines Fitbit Charge 5	39

4.5	Sinngemäßer Ablauf des entwickelten Programms für das Einlesen, Verarbeiten und Darstellen der Daten eines Fitbit Charge 5	40
4.6	Input-Output Sicht der mit der Graphical User Interface (GUI) abgebildeten Funktion aus der Funktionsstruktur 3.4	40
4.7	Hardware des MEDICALPAD [30]	41
4.8	Ablauf der in der GUI simulierten Autorisierung	42
4.9	Grundgerüst der GUI mit Wearbale Daten Reiter	43
4.10	Visualisierte SpO2 Daten eingeordnet in B - Breathing	43
4.11	Alarmfeld zur Übersicht der Abweichungen	43
4.12	Vollständige Ansicht des entwickelten GUI Prototyps	44
5.1	Ablauf der Thematischen Analyse der Interview Antworten	47

Tabellenverzeichnis

2.1	Ausgewählte Smart Wearables der größten Hersteller und jeweils erfasste Daten	7
3.1	Zuordnung Fallbeispiele und Smart Wearable Daten	19
3.2	grundlegende Überlegungen für die Visualisierung der Smart Wearable Daten	23
3.3	Anforderungsliste an ein System zur Integration von Smart Wearable Daten in den Rettungsdienst	24
5.1	TestszENARIO: Sturzereignis ohne Fremdeinwirkung durch Dehydration . . .	46
5.2	TestszENARIO: Hypoglykämie durch falsche Insulindosierung	46
5.3	Kommentare der Experten zu den vier befragten Themenbereichen bezogen auf die TestszENARIEN	48
5.4	Fortsetzung: Kommentare der Experten zu den vier befragten Themenbereichen	49
5.5	Überprüfung der Anforderungen	53

Abkürzungen

ATLS	Advanced Trauma Life Support.
AWMF	Arbeitsgemeinschaft der Wissenschaftlichen Medizinischen Fachgesellschaften.
BIA	bioelektrische Impedanzanalyse.
CPR	kardiopulmonale Reanimation.
DiGA	digitale Gesundheitsanwendungen.
EDA	elektrodermale Aktivität.
EKG	Elektrokardiografie.
ePA	elektronische Patientenakte.
GUI	Graphical User Interface.
PHTLS	Pre Hospital Trauma Life Support.
PPG	Photoplethysmografie.
RTW	Rettungswagen.
SpO2	Blutsauerstoffsättigung.
WEINMANN	Weinmann Emergency Medical Technology GmbH + Co. KG.

Glossar

Akutes Koronarsyndrom	Akute Brustschmerzen infolge einer Blockade von Herzkranzgefäßen (Koronararterien).
Bradykardie	Verlangsamter Herzschlag < 60 SPM.
Exazerbierte COPD	Akute Verschlechterung der chronisch obstruktiven Lungenerkrankung.
Hyperglykämie	Überzuckerung.
Hypertensiver Notfall	Erhöhter Blutdruck mit Zeichen der Endorganschädigung.
Hypoglykämie	Unterzuckerung.
Synkope	Kurzzeitiger Verlust der Bewusstseins.
Tachykardie	Erhöhter Herzschlag > 100 SPM.

1 Einleitung

Diese Bachelorarbeit wird in Zusammenarbeit mit Weinmann Emergency Medical Technology GmbH + Co. KG (WEINMANN) verfasst. WEINMANN entwickelt und vertreibt innovative medizintechnische Geräte für den Notfalleinsatz rund um Beatmung und Defibrillation. Die Firma Weinmann zeichnet sich unter anderem durch einen engen Kundenkontakt, z. B. mit verschiedenen Rettungsdiensten, aus. Im Rahmen der Vorentwicklung von WEINMANN werden vielversprechende Ideen und Themen der Kunden aufgegriffen und untersucht, ob diese in Zukunft für die Firma relevant sein könnten.

In der heutigen Gesellschaft erleben wir aktuell einen Trend zu mehr Gesundheitsbewusstsein. Dazu gehört auch das sogenannte Self-Tracking. Tragbare Computer messen biometrische und Bewegungsdaten des Trägers. Diese werden analysiert und bieten dem Träger die Möglichkeit, seine Gesundheit und Fitness zu überprüfen. Das Ziel: die Gesundheit und Fitness optimieren.

Die Nutzer tragen, meist gespeichert auf ihrem Smartphone, die historischen Daten der Wearables mit sich. Diese Menge an potenziell medizinisch relevanten Daten wird aktuell wenig bis gar nicht medizinisch genutzt. In einem Rettungseinsatz, indem sich möglichst schnell ein Eindruck über einen zu versorgenden Patienten gebildet werden muss, befindet sich also, vorausgesetzt der Patient trägt ein Smart Wearable, im Smartphone des Patienten ein völlig ungenutzter Datensatz, der einen Blick in die pathologische Vergangenheit des Patienten erlaubt und somit für den Einsatz relevante Informationen enthalten könnte. Zudem befinden sich bereits mit dem Wearable, Sensoren am Körper des Patienten, die aktuelle biometrische Daten erfassen, die im Zuge des Notfallmonitoring auch erhoben werden. Bis jetzt gibt es keinen Ansatz diese Daten auszulesen, zu interpretieren und dem Rettungsdienst aufbereitet zu kommunizieren.

In dieser Arbeit wird ein erster Schritt in diese Richtung gegangen und untersucht, ob die, in einem Notfalleinsatz, bisher ungenutzten Daten von Smart Wearables für diesen relevante Informationen enthalten und wie diese in den Rettungsdienst integriert werden könnten.

1.1 Konkretisierung der Aufgabenstellung

Die vorliegende Arbeit beschäftigt sich mit der folgenden These:

„Die Daten von Smart Wearables können, integriert in gängige Handlungsabläufe im Rettungsdienst, die Erstbeurteilung oder Verdachtsdiagnose eines Notfallpatienten unterstützen.“

Für die Untersuchung der These wird zunächst anhand von Experten Interviews und Workshops analysiert, ob Smart Wearables für den Rettungsdienst brauchbare Daten erfassen und bei welchen Fallbeispielen, diese unterstützen könnten. Darauf aufbauend wird untersucht, an welcher Stelle und wie sich die Smart Wearable Daten in gängige Handlungsabläufe des Rettungsdiensts integrieren lassen. Für die Beurteilung der These soll ein Konzept entwickelt werden, welches es ermöglicht die Daten von Smart Wearables in ein Rettungsszenario zu integrieren. Das entwickelte Konzept soll anhand eines Prototyps umgesetzt und in ausgewählten Rettungsszenarien von Experten erprobt werden. Die Beurteilung der Experten wird für die Überprüfung der These und der Bewertung des Konzepts genutzt.

1.2 Aufbau der Arbeit

Zunächst wird in Kapitel 2 (Grundlagen) definiert was unter einem Smart Wearable verstanden wird und ein Überblick gegeben, welche Daten von einem Smart Werabel gemessen und wie diese erfasst werden. Anschließend wird ein Verständnis über den Rettungsdienst und die Abläufe, nach denen im Rettungsdienst gehandelt wird, vermittelt. In Kapitel 3 wird die Konzeptentwicklung für eine System beschreiben welches es ermöglicht Smart Wearable Daten in den Rettungsdienst zu integrieren. Zuerst wird anhand von Experteninterviews und Workshops eine Anforderungsanalyse durchgeführt. Die Ergebnisse werden anschließend in der Anforderungsdokumentation dokumentiert. Das Konzept wird anhand der Anforderungen in Teilfunktionen gegliedert. Für dieser werden Lösungen ausgewählt und zu Lösungsvarianten kombiniert. Die ermittelte Lösungsvariante für das Konzept wird in Kaptitel 4 anhand eines Prototyps realisiert. Dieser wird anschließend in Kapitel 5 dafür genutzt, die aufgestellte These und das Konzept zu evaluieren, in dem der Prototyp in Testszenarien erprobt wird und von Experten bewertet wird. Die Bedeutung der Ergebnisse für die These und das Konzept werden

anschließend diskutiert. Den Abschluss der Arbeit bildet Kapitel 6 mit einer Zusammenfassung und einem Ausblick.

2 Grundlagen

Dieses Kapitel befasst sich mit den für das Verständnis der Arbeit benötigten Grundlagen. Diese sind in den Abschnitt Überblick Smart Wearables und Abläufe im Rettungsdienst gegliedert. Im ersten Abschnitt wird definiert was man unter einem Smart Wearable versteht, welche Sensoren in Smart Wearables das Erfassen von Daten ermöglichen und welche Daten von gängigen Lifestyle Wearables erfasst werden. Der zweite Teil vermittelt ein Verständnis über den Rettungsdienst und die im Rettungsdienst gängigen Abläufe.

2.1 Überblick Smart Wearables

Als Wearable werden elektronische Geräte bezeichnet, die meist als Accessoire am Körper getragen werden oder in Bekleidung integriert sind. Dazu gehören beispielsweise Smartwatches, Armbänder und Ringe [4]. Aktuelle Smart Wearables sind in der Lage, biophysikalische Daten, wie Schrittzahl, Körperposition, Kalorienverbrauch, Blutdruck bis hin zu Herzfrequenz und EKG zu erfassen und zu speichern [29].

Smart Wearables werden, vor allem im Lifestyle und Fitnessbereich, überwiegend in Verbindung mit dem Smartphone und den entsprechenden Apps genutzt, welche die Daten der Wearables speichern, auswerten und für andere Anwendungen nutzbar aufbereitet. Durch die einfache Handhabung ist es jedem möglich seinen Gesundheitszustand zu verfolgen.

Aber nicht nur Lifestyle Geräte werden als Wearables bezeichnet, auch medizinische Geräte zählen zu den Wearables. Beispiele sind Glucosesensoren, Insulinpumpen und Herzschrittmacher, die über einen Computer oder ein Smartphone angesteuert werden können und aufgezeichnete Daten versenden. Auch im Monitoring Bereich werden medizinische Patches eingesetzt, die durch auf die Haut geklebte Sensoren Vitaldaten sammeln.

Grade durch die Entwicklung in der Covid Pandemie hat der Einsatz von Wearables in der klinischen Routine stark zugenommen. Hier bieten Wearables die Möglichkeit eines Remote-Monitoring ohne einen persönlichen Kontakt [16].

Heutzutage besitzt ca. 90% der deutschen Bevölkerung ein Smartphone [26] und jeder Dritte besitzt einen Fitnessstracker [25], welcher meist in Verbindung mit dem Smartphone als fester Bestandteil des Lebens am Körper getragen wird und kontinuierlich Daten des Trägers sammelt. Das bedeutet, dass es im Rahmen des Regelrettungsdienst zusätzliche Daten gibt, die bei der Beurteilung eines Patienten helfen können.

Aufgrund der hohen Verfügbarkeit wird in dieser Arbeit ein Fokus auf Daten gelegt, die von Smart Wearables aus der Kategorie Lifestyle Geräte erhoben werden. Ergänzt werden diese Daten durch den im Smartphone integrierten Notfallpass und Glucosesensoren, die bei Diabeteserkrankungen eine hohe Anwendung finden.

2.1.1 Sensoren von Lifestyle Smart Wearables

Lifestyle Smart Wearables bieten die Möglichkeit eine Vielzahl von Daten über die integrierten Sensoren aufzuzeichnen. Die verbauten Sensoren lassen sich in Aktivitätssensoren und biometrische Sensoren einteilen. [4]. Aktivitätssensoren erfassen die Aktivität des Trägers und quantifizieren diese. Biometrische Sensoren erfassen eine Funktion oder Eigenschaft des Körpers.

Zu den Aktivitätssensoren zählen u. a. folgende Sensoren:

- **Beschleunigungssensor:** Erfasst die Beschleunigung entlang drei Achsen.
- **Gyroskop:** Erfasst die Drehbewegung um drei Achsen.
- **GPS Sensor:** Ermittelt die Position des Sensors. Mit mehreren Positionen und entsprechenden Zeitstempeln, lässt sich die Geschwindigkeit und Bewegungsrichtung des Sensors bestimmen.
- **Barometer:** Erfasst den Luftdruck. Dieser lässt eine Höhenschätzung zu.

Über die Aktivitätssensoren kann die Position des Smart Wearable bestimmt, Höhenänderungen in Form von Stufenanzahl oder Stürzen erkannt und Aktivitätsmuster wie z. B. gehen oder laufen unterschieden werden. Es kann der Kalorienverbrauch geschätzt

und mit Kombination der Daten von den biometrischen Sensoren können Rückschlüsse auf die Schlafqualität gezogen werden.

Zu den biometrischen Sensoren zählen u. a. folgende Sensoren:

- **bioelektrische Impedanzanalyse (BIA) Sensor:** Mit Elektroden, die auf der Haut liegen oder berührt werden müssen, werden Veränderungen der Leitfähigkeit des Körpers erfasst. So kann die Körperzusammensetzung (z. B. Fettmasse und Wasserhaushalt) ermittelt werden. Elektrodermale Aktivität (EDA) ist eine Form der BIA bei der die Stressreaktion des Körpers ermittelt wird.
- **Temperatursensor:** Erfasst die Temperatur auf der Hautoberfläche. Oft wird nur eine Abweichung der Normaltemperatur und keine absolute Temperatur angegeben.
- **Elektrokardiografie (EKG):** Bei einem EKG werden die Herzströme anhand von elektrischen Oberflächensignalen auf der Haut gemessen. Der Pluspol hat bei einem Wearable Kontakt mit dem Handgelenk. Die Elektrode ist so in das Wearable integriert, dass sie mit der anderen Hand berührt werden kann. So ist es möglich ein 1-Kanal-EKG zu messen.
- **Photoplethysmografie (PPG) Sensor:** Ein Großteil der roten Blutkörperchen besteht aus dem Protein Hämoglobin, welches die Eigenschaft besitzt Infrarotlicht zu absorbieren. Ein PPG Sensor sendet Infrarotlicht auf die Haut, misst über eine Fotodiode die Intensität des reflektierten Lichts und stellt diese als Pulswellenkurve da. Diese entspricht der Änderung des Blutvolumens in den Kapillargefäßen [18]. Daraus kann die Herzfrequenz, Blutsauerstoffsättigung (SpO2) und der Blutdruck ermittelt werden.

2.1.2 Daten von Lifestyle Wearables

Da nicht alle auf dem Markt verfügbaren Geräte betrachtet werden können, werden für eine Aufstellung der gängigen Daten, Geräte der drei größten Hersteller im Lifestyle Smart Wearable Bereich Apple, Samsung und Fitbit betrachtet [27]. Diese Hersteller haben zusammen einen Smartwatch Marktanteil von über 60% und werden daher als Referenz angesehen. Für eine Gegenüberstellung der Funktionen in Tabelle 2.1 werden die Apple Watch Series 8 [9], die Galaxy Watch 5 von Samsung [24] und das Fitbit Charge 5 [10] gewählt. Die betrachteten Smart Wearables erfassen sehr ähnliche Daten.

Bei der Sturz & Unfallerkennung, dem Blutdruck und der BIA unterscheiden sich die Smart Wearables.

Voraussetzung einer Verwendung von Smart Wearable Daten ist eine ausreichende Datenqualität. Eine kurze Literaturrecherche zu der Datenqualität der führenden drei Hersteller zeigt, dass es eine umfangreiche Studienlage gibt. Diese legt nahe, dass die Datenqualität für eine Unterstützung der Erstbeurteilung oder Verdachtsdiagnose ausreichen könnte [29] [20] [22]. Eine ausführliche Recherche und Bewertung der Studienlage sind aus Zeitgründen nicht Teil dieser Arbeit.

Tabelle 2.1: Ausgewählte Smart Wearables der größten Hersteller und jeweils erfasste Daten

	Apple Watch 8	Galaxy Watch 5	Fitbit Charge 5
SpO2	x	x	x
EKG	x	x	x
Herzfrequenz	x	x	x
Temperaturerkennung	x	x	x
Schlafphasen	x	x	x
Sturz & Unfallerkennung	x	Sturz	
Blutdruck		x	
BIA		x	EDA
Schrittzähler	x	x	x
Workoutaufzeichnung	x	x	x

2.2 Abläufe im Rettungsdienst

In dieser Arbeit werden die Daten von Smart Wearables mit dem Regelrettungsdienst verknüpft. Daher wird im Folgenden ein Verständnis über den Rettungsdienst und die Abläufe, nach denen im Rettungsdienst gehandelt wird, vermittelt.

Rettungsdienste stellen eine präklinische Notfallversorgung für alle Patienten sicher, grundsätzlich wird versucht das therapiefreie Intervall (Zeitraum in dem keine spezifische medizinische Behandlung stattfindet) zu verkürzen. Weltweit haben sich zwei unterschiedliche Systeme etabliert, das Paramedic und das Notarztsystem. Im Paramedic System wird das therapiefreie Intervall so kurz wie möglich gehalten, indem der Patient so schnell wie möglich klinisch versorgt wird. Hierfür müssen die präklinischen Maßnahmen einfach durchführbar und zeitsparend sein. Diese Maßnahmen werden von speziell dafür ausgebildeten paramedizinischem Personal (Paramedics) durchgeführt. Umgangssprachlich wird dieses System auch als "Load and Go" bezeichnet. Das Ziel ist es den Patienten nur soweit zu stabilisieren, dass er transportfähig ist und die Therapie im klinischen Umfeld durchzuführen. Diese System findet z. B. in den USA Anwendung. Im Notarztsystem wird die Verkürzung des therapiefreien Intervalls durch eine möglichst spezifische Patientenversorgung vor Ort gewährleistet. Hierfür wird ein Notarzt und die entsprechende Ausrüstung in das System integriert. Wenn der geschilderte Zustand des Patienten den Einsatz eines Notarztes erfordert, kommt dieser parallel zu einem Rettungswagen (RTW) zum Einsatzort. Das Ziel ist es, den Patienten vor Ort zu stabilisieren und unter erster Therapie zu transportieren. [19]

Die SOPs (Standard Operating Procedure) im Rettungsdienst richten sich nach den medizinischen Leitlinien der Arbeitsgemeinschaft der Wissenschaftlichen Medizinischen Fachgesellschaften (AWMF) [3]. Umgesetzt werden diese im deutschen Raum durch das Pre Hospital Trauma Life Support (PHTLS) Konzept. Das PHTLS ist das am weitesten verbreitete Ausbildungskonzept zur präklinischen Versorgung von Patienten. Es basiert auf dem Advanced Trauma Life Support (ATLS) Konzept. Teil des PHTLS Konzepts sind das ABCDE und SAMPLE Schema [1]. Die genaue Ausgestaltung der Konzepte variiert innerhalb der Bundesländer stark, die grundlegende Ausbildung ist aber dieselbe.

2.2.1 ABCDE Schema

Die folgenden Abschnitte zum ABCDE und SAMPLE Schema sind dem Artikel „Leben retten mit System“ [17] entnommen. Das ABCDE Schema wird am Unfallort sukzessive abgearbeitet und unterstützt dabei nach Prioritäten vorzugehen. Es entspricht dem traumatologischen Grundsatz "Treat first what kills first".

Bevor der Patient am Einsatzort versorgt wird, wird der Einsatzort, die Situation und die Sicherheit des Einsatzpersonals erfasst. Erst danach beginnt die Versorgung des Patienten mit dem Ersteindruck. Der Ersteindruck sollte innerhalb von 15 Sekunden entstehen und den Patienten als potenziell kritisch oder unkritisch einordnen. Danach beginnt der erste Untersuchungsgang, indem das ABCDE Schema abgearbeitet wird.

A - Airway: Als Erstes wird sichergestellt, dass die Atemwege des Patienten frei sind und eine Atmung vorhanden ist. Maßnahmen für eine Atemwegssicherung sind z. B. Reklination (Überstreckung) des Kopfs, legen eines Tubus, entfernen von Fremdkörpern oder Immobilisation der Halswirbelsäule.

B - Breathing: Nach Ausschluss eines A-Problems wird die Qualität der Atmung des Patienten überprüft. Hierfür wird u. a. die Atemfrequenz, Atemtiefe, das Atemmuster, die Thoraxbewegung (Brustkorb), Blutsauerstoffsättigung und der Status der Halsvenen überprüft. Ist die Atmung unzureichend wird der Patient assistiert bzw. kontrolliert beatmet.

C - Circulation: Ist ein B-Problem beseitigt, schließt sich eine Untersuchung des Kreislaufs des Patienten an. Es werden der Puls, Blutdruck und die Rekapillarierungszeit geprüft und äußere Blutungen werden gestillt. Bei Kreislaufinsuffizienz wird die entsprechende Ursache durch Gabe von Volumen (körpereigene oder künstliche Stoffe) bis hin zur kardiopulmonalen Reanimation (CPR) bekämpft.

D - Disability/Defizit: Der neurologische Zustand des Patienten wird überprüft, indem er auf Bewusstseinsstörungen, Lähmungen, Krämpfe und Pupillendifferenz untersucht wird. D-Probleme werden vor Ort durch Gabe von Glukose oder Antidot (Gegengift) oder durch Transport in ein entsprechendes Zentrum gelöst.

E - Environment/Exposure: Für die Untersuchung auf weitere Verletzungen und Einblutungen wird der Patient entkleidet. Dabei muss auf den Wärmehalt geachtet werden.

2.2.2 SAMPLE Schema

Ist die Erstuntersuchung abgeschlossen, wird diese im zweiten Untersuchungsgang durch eine Notfalleanamnese ergänzt. Der Fokus liegt hierbei auf der Diagnosefindung. Das SAMPLE Schema geht auf alle wichtigen Fragestellungen für eine Diagnose ein. Ist ein Patient nicht kommunikationsfähig kann das SAMPLE Schema auch für eine Fremddiagnose genutzt werden. SAMPLE steht für:

Symptoms:	Unter welchen Symptomen leidet der Patient?
Allergies:	Unter welchen bekannten Allergien leidet der Patient?
Medication:	Steht der Patient unter Dauermedikation? Welche Medikamente hat der Patient eingenommen?
Past Medical History:	Welche persönliche medizinische Vorgeschichte hat der Patient?
Last Oral Intake:	Wie lange liegt die letzte Nahrungsaufnahme zurück?
Events Prior to Incident:	Gab es vor dem Vorfall vorangegangene , eventuell auslösende Ereignisse?

2.2.3 4Hs & HITS

Liegt bei einem Notfallpatienten ein Atem-Kreislauf-Stillstand vor, wird unverzüglich mit einer CPR begonnen. Ziel der Reanimation ist es den Spontankreislauf wieder herzustellen. Damit dies gelingt, müssen die Ursachen des Atem-Kreislauf-Stillstands erkannt und behandelt werden. Die 4Hs & HITS stehen für potentiell reversible Ursachen und helfen diese strukturiert zu erkennen.

3 Konzeptentwicklung

Für die Untersuchung der in Abschnitt 1.1 aufgestellten These, wird im folgenden Kapitel die Entwicklung des Konzepts für ein System beschreiben, welches es ermöglicht die Daten eines gängigen Smart Wearables in die Handlungsabläufe des Rettungsdienst zu integrieren. Auf die Anforderungsanalyse folgt die Entwicklung von Teilfunktionen und eine anschließende Auswahl aus dem Lösungsraum.

Der Ablauf der Konzeptentwicklung und -umsetzung ist an die systematische Konstruktion nach Pahl und Beitz [5] angelehnt. Diese beschreibt den Produktentwicklungsprozess mit den Phasen *Planen - Konzipieren - Entwerfen - Ausarbeiten*. Die für diese Arbeit relevanten Phasen sind in Abbildung 3.1 dargestellt. Zunächst werden in der Anforderungsanalyse die Anforderungen an das Konzept herausgearbeitet. Anschließend werden anhand der Anforderungen Funktionen ermittelt und Lösungen gesucht. Diese werden systematisch bewertet und in einem Gesamtkonzept kombiniert. Die Funktionen des Konzepts werden in Kapitel 4 im Rahmen eines Prototyps umgesetzt.

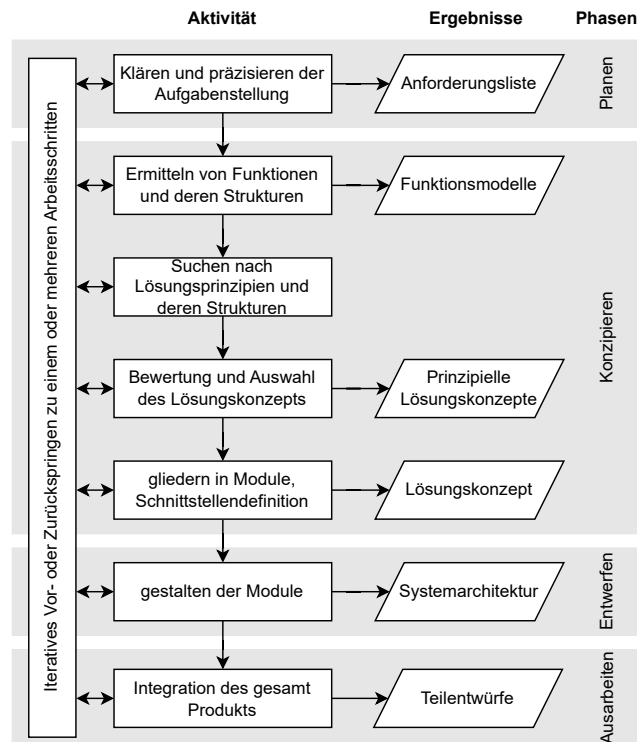


Abbildung 3.1: Produktgestaltung nach Pahl und Beitz (Ausschnitt) Iterativer Ansatz und Ablauf nach Phasen geordnet. [5]

3.1 Anforderungsanalyse

Die Anforderungsanalyse bildet den ersten Schritt der Konzeptentwicklung. Da es sich bei dieser Arbeit um keine klassische Produktentwicklung handelt, die mit einem Lastenheft beginnt, wird zur Gewinnung der Anforderungen die Methode des gezielten Fragens verwendet und in Form von Experteninterviews und Workshops durchgeführt. Hierfür wird als Stakeholder das Rettungsdienstpersonal als spätere Anwender identifiziert und ausgewählt. Auf diese Weise bilden die Anforderungen ein klares Bild, welche spezifischen Kriterien und Bedürfnisse der Rettungsdienst hat. Diese Kriterien sind maßgeblich für die spätere Verifikation und Bewertung des Konzepts [5].

Ziel der Anforderungsanalyse ist eine klare Einschätzung welche Smart Wearable Daten bei welchen Fallbeispielen im Rettungsdienst, an welcher Stelle der Abläufe sinnvoll

genutzt werden können und wie diese in die Abläufe integriert werden können. Diese Erkenntnisse werden im Anschluss in Anforderungen umgesetzt.

3.1.1 Experteninterviews

Für eine gefächerte Einschätzung der Thematik werden Interviews mit mehreren Experten durchgeführt. Das Rettungsdienstpersonal hat als Anwender-Stakeholder direkten Kontakt zu den Patienten, kennt die Abläufe aus erster Hand, ist eventuell schon mit der Thematik von Smart Wearables im Rettungsdienst konfrontiert worden und hat sich mit dieser auseinandergesetzt. Dieses Wissen soll für einen Gesamtüberblick der Thematik genutzt werden. Um das Expertenwissen abzufragen, werden offene Fragen entwickelt, die sich aus der Aufgabenstellung in Abschnitt 1.1 und einer ersten Recherche ableiten. Die Experteninterviews orientieren sich an den entwickelten Fragen, lassen aber Raum, auf die Dynamik des Interviews einzugehen. Es werden insgesamt Experteninterviews mit drei Personen, die auf verschiedenen Rettungswachen tätig sind, geführt. Die Interviewpartner wurden nach gängigen Abläufen und Schemata im Rettungsdienst, häufigen Einsatzgründen, benutzten Geräten, Erfahrungen mit Smart Wearables im Rettungsdienst und Ihrer Einschätzung von Smart Wearable Daten im Rettungsdienst befragt. Der Fragenkatalog und Auszüge aus den Interviews befinden sich im Anhang A.2.

3.1.2 Ergebnisse der Experteninterviews

Die Ergebnisse der Experteninterviews sind nach den Interviewfragen in Anhang A.2 gegliedert.

Abläufe/Schemata für Rettungseinsätze:

Alle Interviewpartner handeln grundsätzlich bei jedem Rettungseinsatz nach dem ABCDE und SAMPLE Schema. Danach wird je nach Fall in feinere Abläufe unterteilt. Im Zuge dieser Schemata wird ein Ersteindruck gebildet, eine Notfallanamnese erhoben und eine erste Verdachtsdiagnose gestellt.

Häufige Einsatzgründe:

Häufige Einsatzgründe sind Sturz, allgemeine Zustandsverschlechterung, Abdomen Beschwerden, akutes Koronarsyndrom und Alkoholvergiftung.

Genutzte Geräte bei diesen Einsatzarten: Bei fast jedem akuten Einsatz wird der Notfallrucksack und ein EKG-Gerät mit zum Patienten genommen. Ein Beatmungsgerät wird nach Ermessen mitgenommen. Das Einsatzinformationsgerät wird teilweise nicht mit zum Patienten genommen und wird erst im Fahrzeug zur Einsatzdokumentation genutzt. Größtenteils wird direkt über ein EKG-Gerät, eine Blutdruckmanschette und ein Fingerpulsoximeter erste Vitalparameter aufgezeichnet. Dieses erste Monitoring findet überwiegend in einem Zeitraum von maximal 5 Minuten statt.

Erfahrungen mit Smart Wearables im Rahmen des Rettungsdiensts:

Die Erfahrungen mit Smart Wearables im Rettungsdienst sind sehr unterschiedlich. Die Interviewpartner sagen aus, dass sich schon länger tätige Rettungsdienstmitarbeiter häufig nur auf selbst erhobene Daten und Erfahrungen verlassen. Bei Personen, die selber Smart Wearables benutzen ist die Akzeptanz gegenüber EKG-Einschätzungen und Glukose-Verläufen sehr hoch. Diese Daten werden mitunter, vorausgesetzt der Patient ist ansprechbar, eingesehen und mit dem Patienten besprochen. Aktuell werden von einigen Patienten Systeme genutzt, die Notfallinformationen enthalten und z. B. über einen QR-Code abgerufen werden können [21].

Einsicht von Smart Wearable Daten mit Einordnung in die Abläufe:

Eine mögliche Einsicht von Wearable Daten wurde von allen Interviewpartnern bejaht und an unterschiedlichen Stellen eingeordnet. Eine mögliche Stelle ist im ABCDE und SAMPLE Schema als Ergänzung der Anamnese und besonders der Fremdanamnese. Als eine andere mögliche Stelle zur Einsicht wurde der Abtransport des Patienten genannt oder sobald der Patient stabil ist. Hier wurden Bedenken bezüglich der Kapazität und des Zeitdrucks geäußert, welche eine frühere Einsicht verhindern. Grundsätzlich sind alle Vitalparameter interessant, die über einen längeren Zeitraum aufgenommen wurden. Diese könnten auch für eine Übergabe an ein Krankenhaus von Relevanz sein. Zusätzlich wurde das Thema ePA und Bedenken bezüglich des Datenschutzes angeschnitten.

Für eine weitere Recherche werden aus den Interviews folgende Themen analysiert:

- ABCDE und SAMPLE Schema
- Einordnung in Erstbeurteilung oder Verdachtsdiagnose - Zeit- und Kapazitätsfrage
- erstmal alle Vitaldaten potenziell relevant
- Fokus auf Vitaldaten von Smart Watches, Fitness Trackern und Glukose Sensoren, da diese oft von Patienten getragen werden

3.1.3 Workshops

Die Erkenntnisse aus den Interviews und die darauf aufbauende Recherche werden als Grundlage für die Entwicklung von Workshops verwendet. Workshops sind eine geeignete Methode, um Ideen zu generieren und die Bedürfnisse von Stakeholdern in den Produktentwicklungsprozess zu integrieren. Für die Durchführung der Workshops werden folgende Leitfragen definiert:

1. Welche Smart Wearable Daten haben einen potenziellen Nutzen für den Rettungsdienst?
2. Bei welchen Fallbeispielen würden die Daten helfen?
3. An welcher Stelle der Fallbeispiele müssen die Daten integriert werden?
4. Wie müssen die Daten dargestellt und aufbereitet werden, um sie nutzbar zu machen?

Um die Leitfragen beantworten zu können, werden die beiden Kreativtechniken Brainstorming und Paper Prototyping, beschrieben in 77 Tools for Design Thinker [14], angewandt. Die Kombination aus Brainstorming und Paper Prototyping ermöglicht es zuerst schnell und effektiv viele Ideen zu generieren und danach ausgewählte Ideen, ohne technische Einschränkungen, direkt in Konzepten zu visualisieren. Es entstehen für jeden Teilnehmer optimale Prototyp-Ansätze, aus denen später die Anforderungen extrahiert werden können.

Im Brainstorming werden zuerst verschiedene Daten, die von Smart Wearables erfasst werden können, in Papierform zur Verfügung gestellt. Diese Daten werden diskutiert und potenziell interessante Daten für den Rettungsdienst herausgearbeitet. Die Menge der Daten darf um eigene Ideen ergänzt werden. Um einen Fokus zu setzen, wird zu Beginn klar angesprochen, dass die Anwendung für eine Erstbeurteilung oder Verdachtsdiagnose betrachtet wird und dass im Zuge dieser Arbeit Smart Wearables betrachtet werden, die potenziell häufiger bei Patienten anzutreffen sind.

Darauf aufbauend werden Fallbeispiele gesammelt, die von den herausgearbeiteten Daten profitieren können. Hierbei wird direkt eine Verbindung zwischen den Fallbeispielen und den Daten gezogen.

Im nächsten Schritt wird das Paper Prototyping eingesetzt. Hierbei entwerfen die Teilnehmer auf einem DIN A4 Blatt eine GUI, die alle relevanten Daten für ein vorher ausgewähltes Fallbeispiel, aufbereitet anzeigt. Zusätzlich wird sich Gedanken darüber gemacht, wann und unter welchen Voraussetzungen es innerhalb des gewählten Fallbeispiels sinnvoll ist, die Daten einzusehen. Die Grundlage, von der die Teilnehmer ausgehen ist eine fiktive, optimale Schnittstelle, die auf Knopfdruck innerhalb von 30 Sekunden alle gewünschten Daten bereitstellen kann. Somit ist ein Fokus ohne Einschränkungen durch technisches Vorwissen möglich.

Es werden zwei Workshops mit insgesamt vier Teilnehmern durchgeführt. Die Teilnehmer setzen sich aus Mitarbeitern der Firma WEINMANN zusammen, die Rettungsdienst Erfahrung und ein entsprechendes Expertenwissen haben.

3.1.4 Ergebnisse der Workshops

Die aus den Workshops erarbeiteten Ergebnisse, werden gemäß der Leitfragen 3.1.3 gegliedert dargestellt. Die Dokumentation der Workshops befinden sich zur Übersicht im Anhang A.3.

Potenzielle Smart Wearable Daten

Laut den Workshopteilnehmern können Smart Wearable Daten ein Monitoring, welches im Zuge des Notfalleinsatzes durchgeführt wird, nicht ersetzen, bieten aber oft die Möglichkeit Zeitverläufe von Daten einzusehen, die vor dem Einsatzstart aufgezeichnet wurden. Die Zeitverläufe der Daten können Hinweise auf eine Verdachtsdiagnose geben und so zur Unterstützung dieser genutzt werden. Folgende Daten werden für den Rettungsdienst als potentiell hilfreich eingeordnet:

1. Blutzucker
2. Blutdruck
3. Beschleunigungskräfte
4. EKG
5. Herzfrequenz

6. Notfallpass

7. SpO₂

Eine kontinuierliche Blutdruckmessung ist z. B. mit dem Aktia Armband [2] verfügbar. Eine punktuelle Blutdruckmessung ist z. B. mit der Galaxy Watch von Samsung [24] verfügbar. Dabei wird im Wesentlichen die Messmethode PPG verwendet, welche sich auch in gängigen Smart Wearables wiederfindet. Es ist also denkbar, dass in Zukunft eine Blutdruckmessung bei einem breiten Spektrum von Smart Wearables verfügbar sein wird.

Der Notfallpass ist in jedem Smartphone integriert und wird vom Patienten selber ausgefüllt. Wird das Smartphone als potenzielle Schnittstelle betrachtet, ist der Notfallpass eine sinnvolle Ergänzung zu den Daten.

Glukosesensoren finden bei Diabetespatienten eine sehr hohe Anwendung. Sie messen kontinuierlich den Blutzuckergehalt des Trägers und senden diese an ein verbundenes Smartphone. Somit können diese Daten ebenfalls eine gute Ergänzung sein.

Fallbeispiele

Aus den Workshops geht hervor, dass die Abläufe der meisten Fallbeispiele ähnlich sind. Der Ablauf ändert sich mit der Eigenschaft Patient kommunikationsfähig oder nicht kommunikationsfähig wie in Abbildung 3.2 dargestellt. Alle Fallbeispiele können mit beiden Ausprägungen auftreten. Bei einem Fallbeispiel, in dem der Patient kommunikationsfähig ist, können die Smart Wearable Daten als Unterstützung der Verdachtsdiagnose dienen. Bei einem Fallbeispiel, in dem der Patient nicht kommunikationsfähig ist, können die Smart Wearable Daten die Fremdanamnese unterstützen.

Alle Fallbeispiele können unter Umständen zu einer Reanimation führen. Deshalb muss diese gesondert behandelt werden. Ein Atem-Kreislauf-Stillstand hat meist potenziell reversible Ursachen. Je früher diese erkannt und behandelt werden, desto größer ist die Wahrscheinlichkeit einer erfolgreichen Reanimation [8]. Die Daten von Smart Wearables können die Ursachenforschung, die durch die 4Hs + HITs beschrieben ist (s. Abs. 2.2.3), unterstützen.

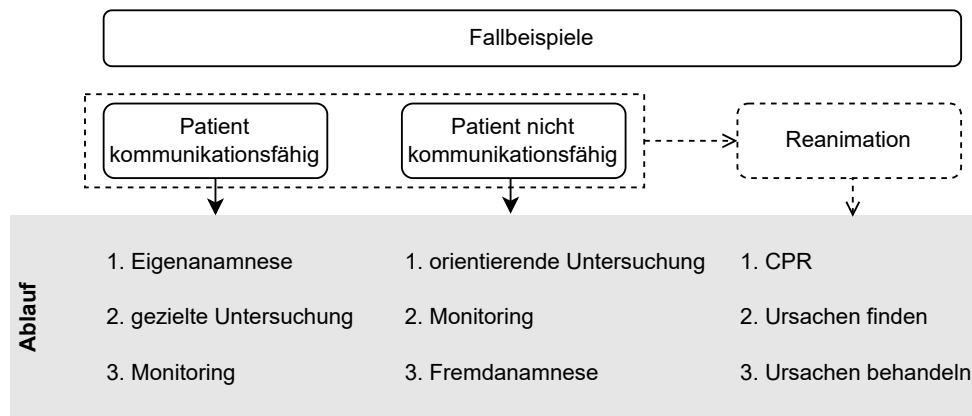


Abbildung 3.2: Verallgemeinerte Abläufe von Fallbeispielen unterschieden in Patient kommunikationsfähig/nicht kommunikationsfähig. Mit Sonderfall Reanimation

Zuordnung der Smart Wearable Daten zu Fallbeispielen

In Tabelle 3.1 werden die von den Workshopteilnehmern als interessant eingestuften Smart Wearable Daten den expliziten Fallbeispielen zugeordnet. Es lässt sich erkennen, dass den Fallbeispielen Reanimation und Sturzereignis ohne Fremdeinwirkung die meisten Daten zugeordnet werden. Diese Fallbeispiele können aus den vorangegangenen Fallbeispielen resultieren. Hier werden die Smart Wearable Daten statt für eine Stützung einer Verdachtsdiagnose, für eine Einschätzung genutzt, woraus der Zustand des Patienten resultiert (vgl. 3.1.4). Anschließend wird erklärt, wofür die Daten in den jeweiligen Fallbeispielen genutzt werden können.

Tabelle 3.1: Zuordnung Fallbeispiele und Smart Wearable Daten

	Notfallpass	Herz- frequenz	EKG	SpO2	Blut- zucker	Blut- druck	Beschleuni- gungskräfte
Hypertensiver Notfall	X					X	
Tachykardie & Bradykardie	X	X	X				
Akutes Koronar- syndrom	X		X				
Hypoglykämie & Hyperglykämie	X				X		
Exazerbierte COPD	X			X			
Synkope	X	X	X			X	
Verkehrsunfall	X						X
Sturzereignis ohne Fremdeinwirkung	X	X	X		X	X	
Reanimation	X	X	X	X		X	

Notfallpass:

Bei jedem Notfalleinsatz ist es sinnvoll sich den Notfallpass anzuschauen. Dieser kann Informationen zur Medikation und Vorerkrankungen sowie Notfallkontakten enthalten. Ist der Patient ansprechbar, ist dieser die erste Quelle für diese Informationen und der Notfallpass kann als Abgleich dienen. Ist der Patient nicht ansprechbar ist der Notfallpass die einfachste Möglichkeit an diese Informationen zu gelangen. Die Voraussetzung für die Nutzung des Notfallpasses ist, dass dieser hinreichend aktuell ist.

Herzfrequenz:

Die Herzfrequenz kann auf eine Tachy-/Bradykardie (zu schneller / zu langsamer Herzschlag) hinweisen, bei der Ursachenfindung bei Synkopen (kurzzeitiger Bewusstseinsverlust) oder Sturzereignissen ohne Fremdeinwirkung genutzt werden. Anhand des Verlaufs der Herzfrequenz kann überprüft werden, wie lange die Abweichungen vom Normalwert bestehen und wann diese zuerst aufgetreten sind.

EKG:

Das EKG kann auf Tachy-/Bradykardie und Synkopen hinweisen und zur Einschätzung des akuten Koronarsyndrom (akute Brustschmerzen) genutzt werden. Vor dem Rettungseinsatz durchgeführte EKG lassen mithilfe des Zeitstempels eine Einschätzung zu, wie lange die Symptome bestehen.

Blutsauerstoffsättigung:

Die Blutsauerstoffsättigung kann zur Unterstützung einer Verdachtsdiagnose bei einer exazerbierten COPD (akute Verschlechterung der chronisch obstruktiven Lungenerkrankung) genutzt werden. Hierfür kann laut Rettungsdienstexperten auch die Nachtsättigungen der letzten Tage für eine grobe Einschätzung ausreichen. Eine höhere zeitliche Auflösung ist wünschenswert, wenn diese über die Smart Wearables verfügbar ist.

Blutzucker:

Die Blutzucker-Verläufe können bei Hyper-/Hypoglykämie (Über-/Unterzuckerung) dazu beitragen eine Einschätzung darüber zu bekommen, welche Blutzuckerwerte für den Patienten normal sind und wie lange ein Befund vorliegt. Dies kann unter anderem die Transportentscheidung beeinflussen.

Beschleunigungskräfte:

Die bei einem Unfall oder Sturz auftretenden Beschleunigungskräfte können bei der Einschätzung der Unfall- und Patientensituation unterstützen sowie als Indikator für die Alarmierung des Schockraumteams dienen. Die Einschätzung für die Alarmierung des Schockraumteams bezieht sich nach Polytrauma/Schwerverletzten-Behandlung Leitlinie [6] unter anderem mit einem Wert > 30 km/h auf eine Geschwindigkeitsänderung. Eine Darstellung der Beschleunigungskräfte in G ist für die Einschätzung des Patienten ebenfalls denkbar.

Blutdruck:

Die Blutdruckmessung kann bei Hypertensiven Notfällen (kritischer Anstieg des Blutdrucks), sowie bei kurzzeitigem Bewusstseinsverlust (Synkope) eine zeitliche Einschätzung geben und somit auch die Transportentscheidung bei leichten Fällen beeinflussen.

Einordnung der Daten in Abläufe von ausgewählte Fallbeispielen

In Grafik 3.3 sind die Abläufe der von den Workshopteilnehmern beschriebenen Fallbeispielen dargestellt. Die Abläufe sind zwecks einer groben Übersicht stark vereinfacht

und erheben kein Anspruch auf Vollständigkeit. Die Stellen, an denen eine Integration von Smart Wearable Daten denkbar ist, sind in Rot markiert. Aus den Abläufen lässt sich erkennen, dass ein möglicher Zugriff auf die Daten erst nach einer Anamnese, Untersuchung und einem Monitoring stattfindet. Erst danach lässt sich der Zugriff auf die Daten unterteilen in:

1. Einsicht der Daten für die Festigung einer Verdachtsdiagnose
2. Einsicht der Daten für die Findung von Hinweisen und Ansätzen für eine Verdachtsdiagnose

Für beide Fälle gilt, dass eine Einsicht erst erfolgen kann, wenn der Patient stabilisiert ist oder Kapazität besteht die Daten parallel zur Behandlung einzusehen.

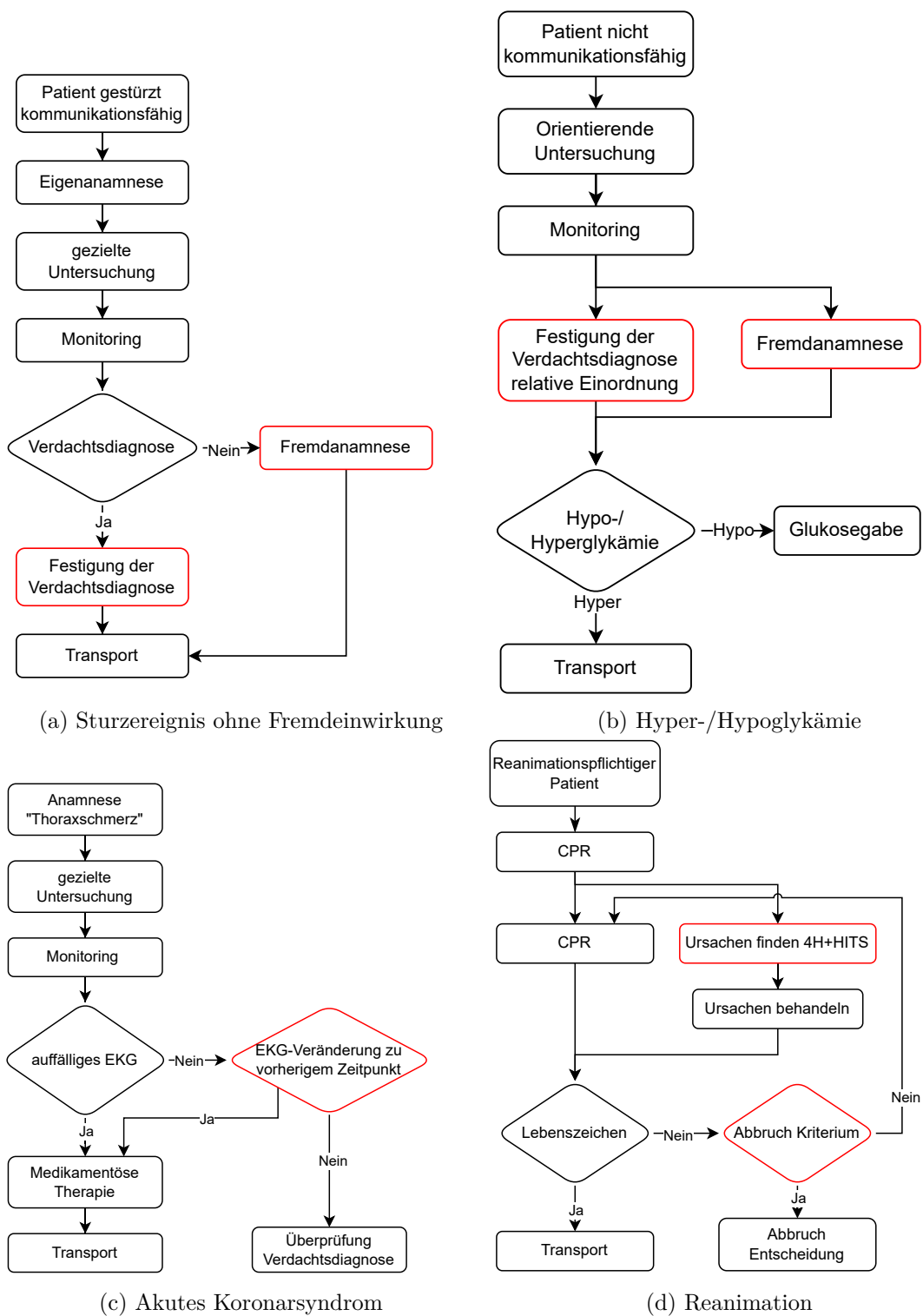


Abbildung 3.3: Einordnung von Smart Wearable Daten (Rot) in die Abläufe der im Workshop dargestellten Fallbeispiele

Darstellung/Aufbereitung:

Für die Darstellung und Aufbereitung der Smart Wearable Daten werden die skizzierten Visualisierungen der Workshopteilnehmer als Grundlage herangezogen. Anhand der Visualisierungen der Teilnehmer werden die Anforderungen an die Visualisierung der Daten entwickelt und in der Anforderungsliste 3.3 dokumentiert. Die folgenden grundlegenden Überlegungen sind in den Visualisierungen der Teilnehmer im Anhang A.3.2 numerisch markiert:

Tabelle 3.2: grundlegende Überlegungen für die Visualisierung der Smart Wearable Daten

1	Große Darstellung des Notfallpasses als zentrales Objekt.
2	Einordnung der Daten in Kategorien oder Schemata
3	Integration in ein bestehendes Design eines Einsatzdokumentationsgeräts.
4	Alarmierung von Abweichungen und auffälligen Werten.
5	Kombination von Zeitverläufen und statischen Kennzahlen.
6	Optionale Integration in Einsatzdokumentation.

Ein Workshopteilnehmer hat eine Visualisierung als Dashboard abgelehnt und eine direkte Integration in die Einsatzdokumentation vorgeschlagen, diese Überlegung fließt in die Visualisierung der Teilnehmer in Tabelle 3.2 unter Punkt 6 mit ein.

3.2 Anforderungsdokumentation

Aus den Ergebnissen der Experteninterviews und Workshops werden Anforderungen an das Konzept erarbeitet und in der Dokumentationsform der natürlichen Sprache festgehalten (s. Tabelle 3.3). Diese hat den Vorteil, dass jeder Stakeholder die Anforderungen versteht ohne die Voraussetzung eine bestimmte Notation zu beherrschen. Um eine Mehrdeutigkeit der Anforderungen zu vermeiden wurden die Anforderungen mithilfe einer Satzschablone formuliert [23]. Die Anforderungen wurden dabei logisch in die drei Teile *Anforderungen an die Daten* - *Anforderungen an die Visualisierung* - *Anforderungen an die Schnittstelle* unterteilt.

Tabelle 3.3: Anforderungsliste an ein System zur Integration von Smart Wearable Daten in den Rettungsdienst

Anforderungen an die Daten (generell):

A1	Die von einem Smart Wearable erfassten Daten müssen aktuell sein und ein Zeitlabel haben.
A2	Die Zeitverläufe der Daten müssen eine zeitliche Auflösung < 30 min haben.
A3	Die Blutsauerstoffsättigung darf eine zeitliche Auflösung von max. 1 d haben.
A4	Brauchbare Daten sollen nach dem Einsatz an ein Krankenhaus weitergegeben werden können.

Anforderungen an die Visualisierung der Daten:

A5	Der Anzeige-Bildschirm soll mindestens 10"Bildschirmdiagonale haben.
A6	Das System soll mehrere Daten gleichzeitig anzeigen.
A7	Das System soll die Daten nach Kategorien oder Schemata sortiert anzeigen.
A8	Das System soll auf Abweichungen von Normwerten der jeweiligen Daten hinweisen.
A9	Der Benutzer des Systems soll die Möglichkeit haben, Datenverläufe zu vergrößern.
A10	Das System soll die Zeitverläufe der Daten um statische Kennzahlen ergänzen.

Anforderungen an die Schnittstelle:

A11	Das System soll nach einer Startbedingung automatisch die verfügbaren Daten von allen Smart Wearables des Patienten auslesen.
A12	Das System soll die Daten mehrerer Smart Wearables unabhängig vom Hersteller auslesen können.
A13	Das System muss die Berechtigung besitzen, um auf die Daten der Smart Wearables des Patienten zugreifen zu dürfen.
A14	Das System soll nur Daten des gewünschten Patienten anzeigen.
A15	Der Prozess vom Auslesen bis zum Anzeigen der Daten soll max. 30s dauern.
A16	Die Datenübertragung muss kabellos stattfinden.

3.3 Teilfunktionen

Um die Komplexität des Konzepts zu reduzieren, wird das Konzept in Teilfunktionen abstrahiert und als Funktionsstruktur dargestellt. Hierfür eignet sich besonders die ablauforientierte Funktionsstruktur nach dem „Input-Output“ Modell, da innerhalb des Konzepts der Durchlauf eines Gutes, in diesem Fall die Daten der Smart Wearables, eine Rolle spielt. [5]

Zunächst wird anhand der Aufgabenstellung (Abschnitt 1.1) die Hauptfunktion „Integration von Smart Wearable Daten in den Rettungsdienst“ abstrahiert. Die Hauptfunktion wird anschließend in Abbildung 3.4 in zwei Abstraktionsebenen in Teilfunktionen unterteilt.

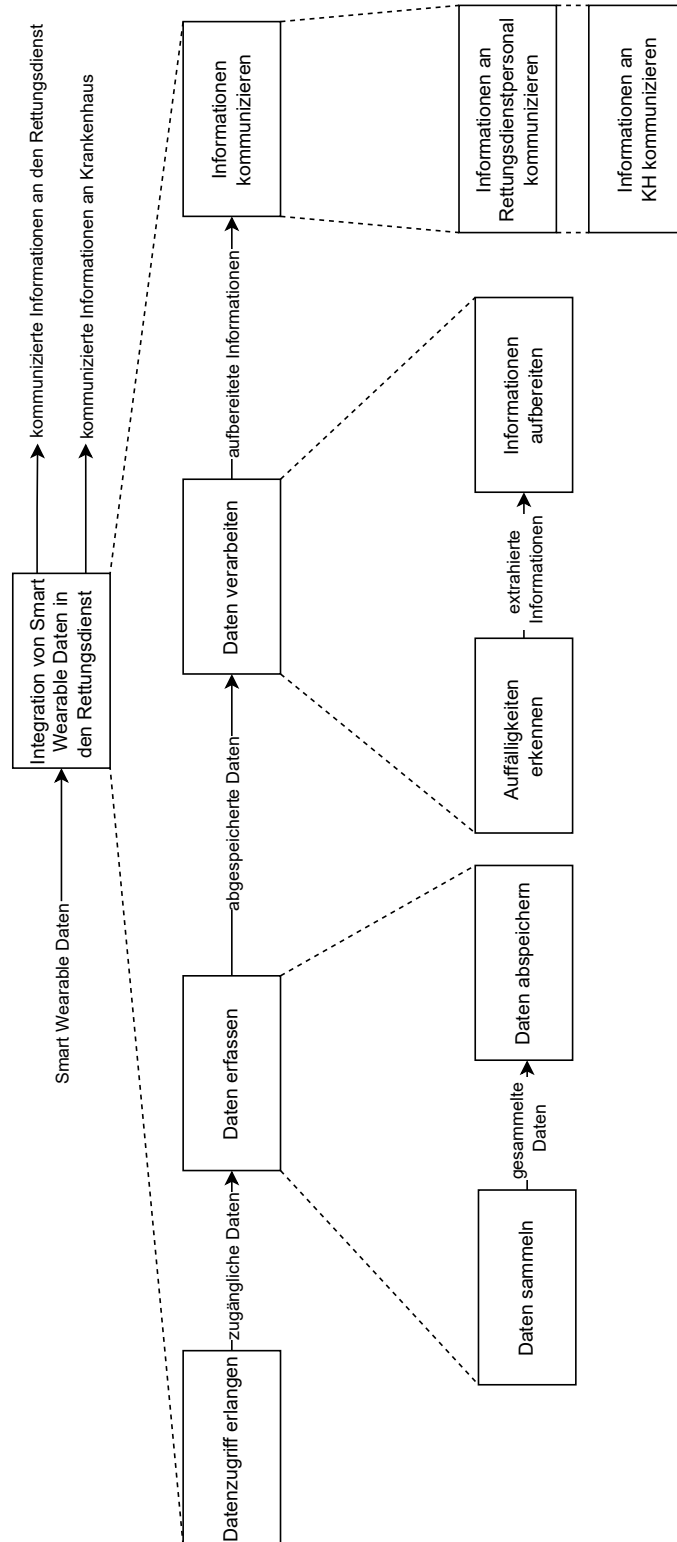


Abbildung 3.4: Funktionsstruktur des Konzepts nach dem „Input-Output“ Modell

3.3.1 Lösungsauswahl

Die Lösungsauswahl wird mit einem Morphologischen Kasten 3.12 unterstützt. In diesem werden Lösungen für die Teilfunktionen gesammelt und anschließend in Lösungsvarianten kombiniert. Zusätzlich zu einer Lösungsvariante, die ein Konzept mit einem konkreten Smart Wearable für die Überprüfung der These 1.1 beschreibt, wird ein optimaler Lösungsansatz für die Hauptfunktion „Integration von Smart Wearable Daten in den Rettungsdienst“ generiert, der einen allgemeinen Einsatz des Konzepts im Rettungsdienst beschreibt.

Die Lösungsvarianten der Teilfunktionen werden mithilfe einer Punktebewertung bewertet und so eine Rangfolge für die Lösungsalternativen ermittelt. Hierfür gilt folgendes Punktesystem: Die Gewichtung (G) der Kriterien wird prozentual nach ihrer Wichtigkeit aufgeteilt. Die Kriterien werden mit 0 bis 5 Punkten gewertet (W). Die Gewichtung (G) wird mit der Wertung (W) multipliziert. Je höher die Punktzahl, desto besser ist ein Kriterium erfüllt. Die Punktebewertung wird für alle Teilfunktionen aus dem Morphologischen Kasten 3.12 durchgeführt und ist in den Abbildungen 3.5 bis 3.11 dargestellt.

Kriterium	Gewichtung		Zugriff über Systemschwachstelle		Patienten ID und Passwort		Rettungsdiensttoken		Heilberufsausweis	
	G	W	G·W	W	G·W	W	G·W	W	G·W	
Benutzerfreundlichkeit	0,1	1	0,1	4	0,4	5	0,5	4	0,4	
hohe Datensicherheit	0,3	1	0,3	4	1,2	5	1,5	5	1,5	
Universell für alle Smart Wearables	0,3	1	0,3	3	0,9	5	1,5	5	1,5	
geringe Patienteninteraktion	0,3	3	0,9	1	0,3	5	1,5	5	1,5	
Gesamt:	1	-	1,6	-	2,8	-	5	-	4,9	

Abbildung 3.5: Punktebewertung für die Teilfunktion Datenzugriff erlangen

Kriterium	Gewichtung		Direkter Zugriff auf Smart Wearables		Zugriff über Smartphone		Zugriff über Hersteller Server (API)		Zugriff über ePA	
	G	W	G*W	W	G*W	W	G*W	W	G*W	
geringer Entwicklungsaufwand	0,2	2	0,4	2	0,4	5	1	3	0,6	
Benutzerfreundlichkeit	0,2	1	0,2	4	0,8	5	1	4	0,8	
Universell für alle Smart Wearables	0,3	1	0,3	5	1,5	1	0,3	5	1,5	
geringer zeitlicher Verzug der Daten	0,3	5	1,5	3	0,9	3	0,9	2	0,6	
Gesamt:	1	-	2,4	-	3,6	-	3,2	-	3,5	

Abbildung 3.6: Punktebewertung für die Teilfunktion Daten sammeln

Kriterium	Gewichtung		lokal auf Gerät		Online in Datenbank		Externes Medium	
	G	W	G*W	W	G*W	W	G*W	
Datensicherheit	0,4	4	1,6	5	2	2	0,8	
hohe Konnektivität	0,2	1	0,2	5	1	3	0,6	
Benutzerfreundlichkeit	0,2	5	1	5	1	3	0,6	
Datenschutz	0,2	5	1	4	0,8	3	0,6	
Gesamt:	1	-	3,8	-	4,8	-	2,6	

Abbildung 3.7: Punktebewertung für die Teilfunktion Daten abspeichern

Kriterium	Gewichtung		Maschine Learning		Festgelegte Regeln und Parameter	
	G	W	G*W	W	G*W	
geringer Entwicklungsaufwand	0,1	1	0,1	5	0,5	
Benutzerfreundlichkeit	0,3	3	0,9	5	1,5	
hohe Detektionsrate	0,3	5	1,5	3	0,9	
komplexität der erfassbaren Abweichungen	0,3	5	1,5	2	0,6	
Gesamt:	1	-	4	-	3,5	

Abbildung 3.8: Punktebewertung für die Teilfunktion Auffälligkeiten erkennen

Kriterium	Gewichtung		Textuelle Aufbereitung		Visuelle Aufbereitung		Auditive Aufbereitung	
	G	W	G*W	W	G*W	W	G*W	
hoher Informationsgehalt	0,4	2	0,8	5	2	2	0,8	
gute Erkennbarkeit von Auffälligkeiten	0,3	5	1,5	5	1,5	3	0,9	
hohes Informationsverständnis	0,3	5	1,5	4	1,2	5	1,5	
Gesamt:	1	-	3,8	-	4,7	-	3,2	

Abbildung 3.9: Punktebewertung für die Teilfunktion Informationen aufbereiten

Kriterium	Gewichtung		Dashboard auf Einsatzdokumentationsgerät		Auditiv aufbereitet über Kopfhörer		Darstellung auf Smartphone		Darstellung der Daten über Smartglasses	
	G	W	G*W	W	G*W	W	G*W	W	G*W	
Benutzerfreundlichkeit	0,3	5	1,5	3	0,9	5	1,5	2	0,6	
hohe Integration in bestehende Systeme	0,3	5	1,5	3	0,9	5	1,5	1	0,3	
Übersichtliche Datenvermittlung	0,4	5	2	2	0,8	3	1,2	4	1,6	
Gesamt:	1	-	5	-	2,6	-	4,2	-	2,5	

Abbildung 3.10: Punktebewertung für die Teilfunktion Informationen an Rettungsdienstpersonal kommunizieren

Kriterium	Gewichtung		senden per Email		Ausdruck übergeben		Drahtlose Schnittstelle		Abruf über Datenbank	
	G	W	G*W	W	G*W	W	G*W	W	G*W	
Benutzerfreundlichkeit	0,3	3	0,9	2	0,6	3	0,9	4	1,2	
hohe Integration in bestehende Systeme	0,4	5	2	5	2	5	2	5	2	
hohe Konnektivität	0,3	2	0,6	1	0,3	3	0,9	5	1,5	
Gesamt:	1	-	3,5	-	2,9	-	3,8	-	4,7	

Abbildung 3.11: Punktebewertung für die Teilfunktion Informationen an Krankenhaus kommunizieren

Auf Basis der Punktebewertungen werden im Morphologischen Kasten 3.12 zwei Lösungsvarianten ausgewählt.

Um eine Lösungsvariante für den Rettungsdienst anwendbar zu machen, müssen die Funktionen *Datenzugriff erlangen* und *Daten sammeln* so umgesetzt werden, dass ein System eine universelle Berechtigung hat auf Smart Wearable Daten des Patienten zuzugreifen und die Daten unabhängig vom Hersteller über eine universelle Schnittstelle abrufen kann. Zusätzlich darf eine Beteiligung des Notfallpatienten für den Zugriff nicht erforderlich sein. Im Rahmen dieser Arbeit können entsprechende Teillösungen zur Erfüllung dieser Ausschlusskriterien nicht umgesetzt werden. Der Vollständigkeit halber, werden diese im Rahmen eines optimalen Lösungsansatzes beschreiben. Der optimale Lösungsansatz ist in Rot markiert. Für die Diskussion der These wird im Rahmen dieser Arbeit eine Lösungsvariante mit Teillösungen ausgewählt, die beispielhaft für ein konkretes Smart Wearable gültig und umsetzbar ist. In dieser Lösungsvariante sind Teillösungen für die Funktionen *Datenzugriff erlangen* und *Daten sammeln* enthalten, die die genannten Ausschlusskriterien nicht erfüllen, um trotzdem Datenzugriff zu erlangen. Eine Lösung zur Erfüllung der Ausschlusskriterien kann dann im Rahmen der Evaluierung in Kapitel 5 für die Auswertung der These simuliert werden. Diese Lösungsvariante ist in grün markiert und wird in Kapitel 4 umgesetzt. Eine Beschreibung beider Lösungsvarianten erfolgt in Abschnitt 3.3.2.

Teilfunktion	Lösungsraum					
	Zugriff über Systemschwachstelle	Patienten ID und Passwort	Rechtungsdiensttoken	Heilberufsausweis	Zugriff über Smartwatches	Zugriff über Smartphone
Datenzugriff erlangen						
Daten sammeln	direkter Zugriff auf Smart Wearables	Zugriff über Hersteller Server (API)			Zugriff über Smartphone	
Daten abspeichern	lokal auf Gerät	externes Medium			Online in Datenbank	
Auffälligkeiten erkennen	festgelegte Regeln und Parameter		Maschine Learning			
Informationen aufbereiten	Textuelle Aufbereitung	Visuelle Aufbereitung			Auditive Aufbereitung	
Informationen an Rettungsdienstpersonal kommunizieren	Dashboard auf Einsatzdokumentationsgerät	Auditiv aufbereitet über Kopfhörer			Darstellung auf Smartphone	Darstellung der Daten über Smart Glasses
Informationen an KH kommunizieren	senden per Email	Ausdruck übergeben			drahtlose Schnittstelle	Abruf über Datenbank

Abbildung 3.12: Morphologischer Kasten mit Lösungsauswahl

3.3.2 Ausgewählte Lösungsvarianten

Im Folgenden werden die beiden, im Morphologischen Kasten 3.12 ausgewählten Lösungsvarianten für die Integration von Smart Wearable Daten in den Rettungsdienst vorgestellt.

Lösungsvariante Grün - Konzept für konkretes Smart Wearable

Die im morphologischen Kasten 3.12 in Grün markierte Lösungsvariante ist für ein konkretes Smart Wearable gültig. Diese Lösungsvariante wird in Kapitel 4 für eine beispielhafte Umsetzung genutzt. Der Systemkontext ist in Abbildung 3.13 dargestellt. Die von einem Smart Wearable erfassten Patientendaten werden mit dem Smartphone des Patienten synchronisiert. Das Smartphone synchronisiert die Daten mit dem Server des Wearable Herstellers. Auf dem System können die User Daten des Patienten hinterlegt werden, die für einen Zugriff auf die Patientendaten, die auf dem Server des Herstellers gespeichert sind, nötig sind. Eine im Rettungsdienst tätige Person kann den Auslesevorgang der Daten starten. Das System authentisiert sich darauf hin am Server des Herstellers und kann über eine API die Patientendaten auslesen. Die Patientendaten werden auf dem System abgespeichert. Auffälligkeiten der Daten werden über festgelegte Regeln und Parameter erkannt. Die erkannten Abweichungen werden mit den Datenverläufen visuell aufbereitet und auf einem Einsatzdokumentationsgerät in Form eines Dashboards, in dem alle Datenverläufe und Abweichungen gleichzeitig eingesehen werden können, dargestellt. Ein externer Zugriff auf die Wearable Daten ist über die Integration in das Einsatzprotokoll, welches über einen Server zugänglich ist, möglich.

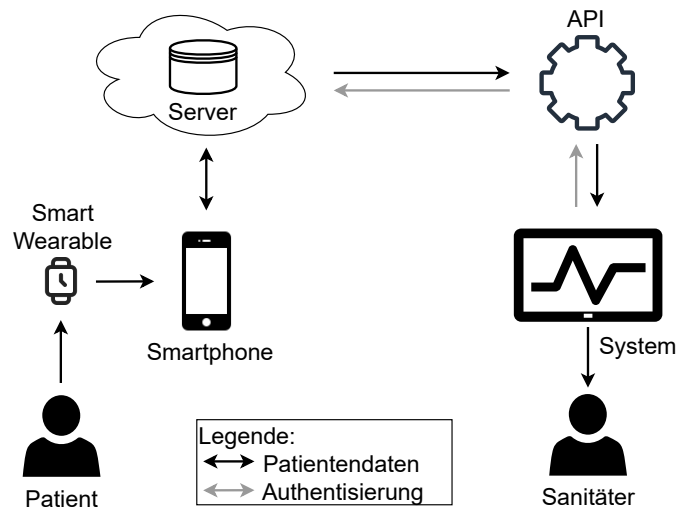


Abbildung 3.13: Systemkontext - Lösungsvariante für ein konkretes Smart Wearable zur Integration von Smart Wearable Daten in den Rettungsdienst

Lösungsvariante Rot - Konzept mit optimalem Lösungsansatz

Die im Morphologischen Kasten 3.12 in Rot markierte Lösungsvariante beschreibt einen optimalen Lösungsansatz für die Integration von Smart Wearable Daten in den Rettungsdienst. Der Systemkontext ist in Abbildung 3.14 dargestellt. Die Lösungsvariante sieht eine Integration in die elektronische Patientenakte (ePA) vor. In der aktuellen Ausbaustufe der ePA ist die Integration von Daten aus digitalen Gesundheitsanwendungen (DiGA) vorgesehen [12]. Zusätzlich ist über die ePA ein digitaler Notfalldatensatz verfügbar, der in einer Notfallsituation ohne Beteiligung des Patienten ausgelesen werden kann. Dieser ist aktuell medizinischem Personal mit einem Heilberufsausweis vorbehalten und protokolliert die Zugriffe mit dem angegebenen Auslesegrund [13].

Die von Smart Wearables erfassten Patientendaten werden mit dem Smartphone des Patienten synchronisiert. Die jeweiligen Hersteller Applikationen der Wearables sind als DiGA anerkannt. So können die Wearable Daten in die ePA integriert werden. Notfallrelevante Wearable Daten werden in den digitalen Notfalldatensatz aufgenommen. Auf den digitalen Notfalldatensatz kann über die auf dem Smartphone hinterlegte ePA zugegriffen werden, indem der Notfalldatensatz mit einem Auslesegrund angefordert wird. In einer Notfallsituation ist für den Zugriff auf den Notfalldatensatz keine Beteiligung des Notfallpatienten erforderlich. Im Rettungsdienst tätige Personen besitzen einen Rettungsdiensttoken, der sie ausschließlich für einen Zugriff auf den Notfalldatensatz au-

torisiert. Die Authentisierung erfolgt über NFC, eine anschließende Datenübertragung erfolgt über ein Protokoll, welches eine höhere Übertragungsrate ermöglicht. Auf dem Einsatzdokumentationsgerät selber erfolgt eine Auswertung der Patientendaten, gestützt durch maschinelles Lernen. So können mehrdimensionale Zusammenhänge der Wearable Daten mit bestimmten pathologischen Events erkannt werden. Diese Informationen werden mit den visuell aufbereiteten Daten auf dem Einsatzdokumentationsgerät in Form eines Dashboards dargestellt. Ein externer Zugriff auf die Daten von weiterem medizinischem Personal ist über eine Integration in das Einsatzprotokoll oder einen Zugriff auf die digitalen Notfalldaten über die ePA möglich.

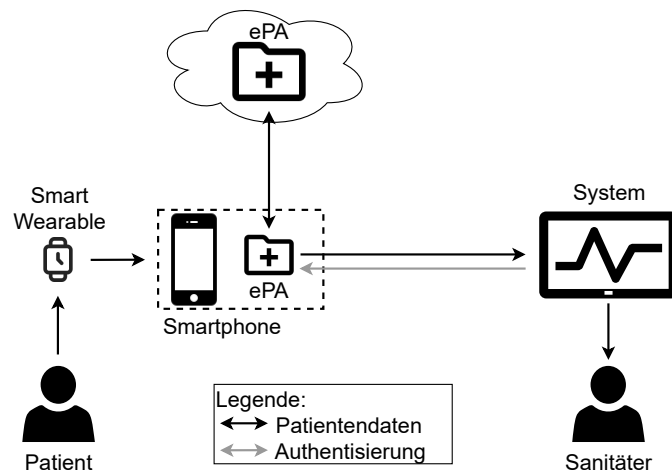


Abbildung 3.14: Systemkontext - Optimale Lösungsvariante zur Integration von Smart Wearable Daten in den Rettungsdienst

4 Realisierung

Um eine Untersuchung der Eingangsthese aus Kapitel 1.1 zu ermöglichen, wird in diesem Kapitel die Umsetzung des in Kapitel 3 entwickelten Konzepts zur Integration eines Smart Wearables in ein konkretes Rettungsszenario beschrieben.

Zuerst wird ein Smart Wearable ausgewählt, dessen Daten genutzt werden sollen. Anschließend wird ein Programm entwickelt welches die Daten des Smart Wearables erfasst und verarbeitet. Für die Kommunikation der aufbereiteten Informationen an den Rettungsdienst wird ein GUI Prototyp entwickelt. Anschließend an die Konzeptrealisierung wird der entwickelte GUI Prototyp mit den ausgelesenen Daten in Abschnitt 5.1 in Test-szenarios erprobt und die Ergebnisse für die Beurteilung der These und des Konzepts genutzt.

4.1 Auswahl eines Smart Wearable

Für die Umsetzung wird ein konkretes Smart Wearable ausgewählt, dessen Daten ausgelesen, aufbereitet und kommuniziert werden sollen. Für die Auswahl werden folgende Kriterien berücksichtigt.

- Deckt einen Großteil der gängigen Daten ab (vgl. Tabelle 2.1).
- Repräsentativ für den aktuellen Markt von Lifestyle Smart Wearables.
- Externer Zugriff auf die Daten ist möglich / im Rahmen der Arbeit umsetzbar.

Aufgrund der Kriterien wurde von Fitbit, ein in Abbildung 4.1 dargestelltes Charge 5 Fitness Armband ausgewählt. Das Charge 5 deckt einen Großteil der in Tabelle 2.1 dargestellten gängigen Daten ab. Ein mittelklassiges Gerät wie das Charge 5 repräsentiert einen breiteren Schnitt der auf dem Markt verfügbaren Geräte als ein High-End Gerät.

Fitbit bietet zudem mit der Fitbit Web API eine Möglichkeit, auf die erfassten Daten zuzugreifen und diese extern zu verarbeiten.



Abbildung 4.1: Ausgewähltes Smart Wearable: Fitbit Charge 5 [10]

4.2 Kommunikation über Fitbit API

Fitbit bietet mit der Fitbit Web API eine eigene API an, die die Möglichkeit bietet mit externen Apps auf die Daten der verbundenen Fitbit Geräte zuzugreifen. Mit der API werden die Teilfunktionen *Datenzugriff erlangen* und *Daten sammeln* umgesetzt. Für verschiedene Sprachen empfiehlt Fitbit Bibliotheken und Beispielcode die von der Fitbit-Community entwickelt wurden. In dieser Arbeit wird die python-fitbit Bibliothek von *orcasgit* [15] genutzt und angepasst.

Um die Fitbit API zu nutzen wird unter <https://dev.fitbit.com/> eine Applikation registriert, welche mit einem Fitbit Account verknüpft ist. Eine registrierte Applikation erhält eine Client ID und ein Client Secret. Mit diesen Daten wird über das OAuth 2.0 Authorization Framework ein Access Token gewährt, der für HTTP Requests an die Fitbit API benötigt wird. Die Fitbit API gibt die angefragten Daten in einem JSON File zurück. Abbildung 4.2 zeigt beispielhaft den Datenfluss zwischen einem Fitbit Gerät und einer Applikation.

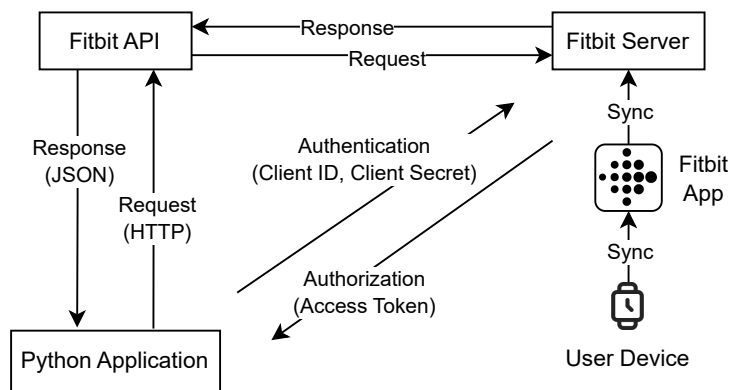


Abbildung 4.2: Datenfluss von Gerät zu Python Applikation über Fitbit API

Anpassungen an python-fitbit Bibliothek

Die python-fitbit Bibliothek wird angepasst um zusätzliche Parameter einzulesen, die von der Fitbit Web API bereitgestellt werden, aber in der Bibliothek noch nicht implementiert wurden. Alle Ressourcen die von der Fitbit Web API bereitgestellt werden sind unter <https://dev.fitbit.com/build/reference/web-api/> zu finden. Es werden die Ressourcen *oxygen_saturation* und *electrocardiogram* und entsprechende Funktionen für das Abrufen der SpO2 und EKG Daten in der Bibliothek ergänzt. Die Änderungen befinden sich im Anhang A.4.2.

4.3 Implementierung eines Programms zur Datenerfassung und -verarbeitung

Die Teilfunktionen *Datenzugriff erlangen*, *Daten erfassen* und *Daten verarbeiten* werden anhand eines Python Programms umgesetzt. Abbildung 4.3 beschreibt die in der Funktion abgebildeten Funktionen und die Systemgrenzen des Programms mit In- und Output.

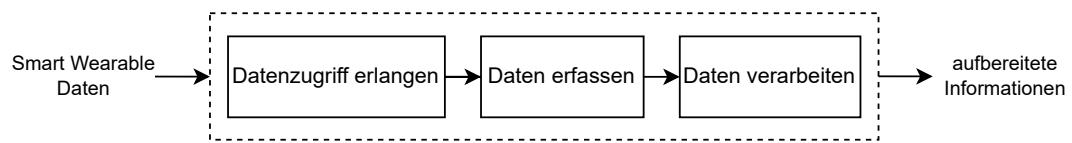


Abbildung 4.3: Input-Output Sicht der im Programm abgebildeten Funktionen aus der Funktionsstruktur 3.4

Die Teilfunktion *Datenzugriff erlangen* wird über hinterlegte User Daten (Client ID und Client Secret) realisiert. Somit ist das Programm nur für das Charge 5 mit dem entsprechenden Fitbit Account gültig. Das Programm liest über die Fitbit API Parameter des ausgewählten Fitbit Charge 5 aus, analysiert die Daten auf Abweichungen von Normwerten und visualisiert anschließend die Daten mit den Abweichungen.

Für die Untersuchung der These werden die Daten des Charge 5 mit den Daten eines Glukose Messsystems [11] und eines Notfallpass ergänzt. Die Daten des Glukose Messsystems werden, statt direkt von dem Programm selber abgerufen zu werden, als .csv exportierte Datei von dem Programm eingelesen.

Der Notfallpass wird von der Klasse *Profile* simuliert. Die Fitbit Profil Daten sind nicht explizit dafür gedacht, um relevante Notfallinformationen zu hinterlegen, kann jedoch im Sinne dieser Arbeit dafür genutzt werden.

Das Programm besteht aus den im Klassendiagramm 4.4 abgebildeten Klassen, die jeweils für den Zugriff, die Verarbeitung und die Visualisierung von spezifischen Sensordaten zuständig sind. Die Klasse *Authorize* ist für das Autorisieren mit den hinterlegten User Daten zuständig. Die Klasse *Time* erzeugt die Zeitstempel, die für die API Requests nötig sind. Die Klassen für die spezifischen Sensordaten haben jeweils eine Methode für das Abrufen und Speichern der Daten, für das Erkennen von Abweichungen und für die Visualisierung der Daten.

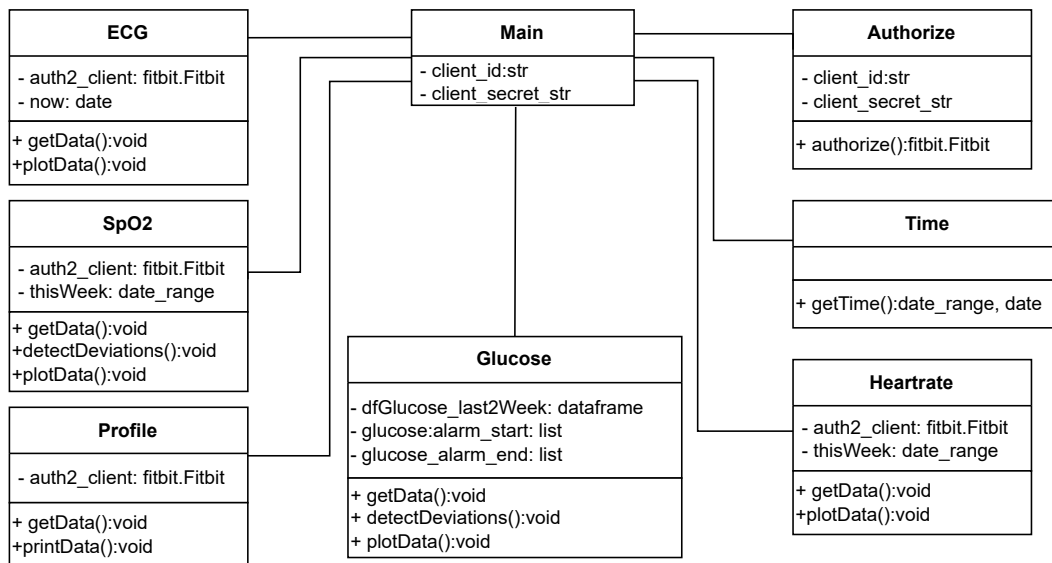


Abbildung 4.4: Klassendiagramm des entwickelten Programms für das Einlesen, Verarbeiten und Darstellen der Daten eines Fitbit Charge 5

Nach Starten des Skripts findet gemäß dem Ablauf 4.5 eine Autorisierung des Users und ein Erzeugen der Zeitstempel statt. Anschließend werden die Klassen instanziiert und die Methoden für das Abrufen, Speichern, Erkennen von Abweichungen und die Visualisierung der Daten aufgerufen.

Die von den Visualisierungsfunktionen erzeugten Plots der Daten werden für die Darstellung in der GUI genutzt. Der Quellcode des Programms befindet sich im Anhang A.4.1.

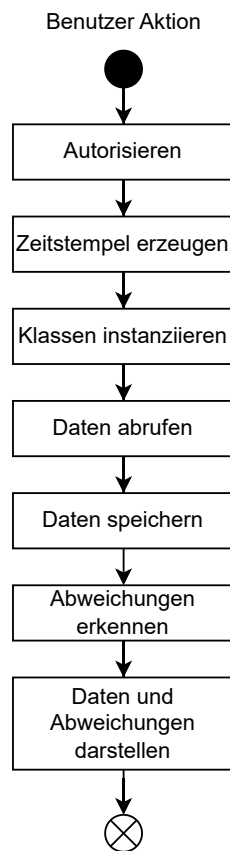


Abbildung 4.5: Sinngemäßer Ablauf des entwickelten Programms für das Einlesen, Verarbeiten und Darstellen der Daten eines Fitbit Charge 5

4.4 Entwicklung eines GUI Prototyps

Die Teilfunktion *Informationen an Rettungsdienstpersonal kommunizieren* wird anhand einer GUI realisiert. Die In- und Output Sicht und die in der GUI umgesetzte Teilfunktion ist in Abbildung 4.6 dargestellt.

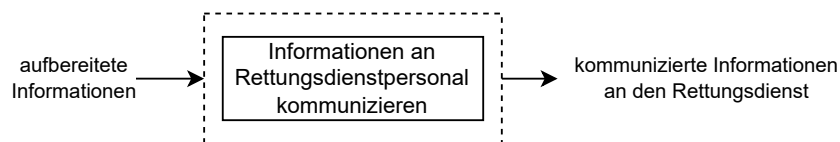


Abbildung 4.6: Input-Output Sicht der mit der GUI abgebildeten Funktion aus der Funktionsstruktur 3.4

Als Grundlage der Entwicklung der GUI dienen die grundlegenden Überlegungen an die Visualisierung der Smart Wearable Daten 3.2 aus den Workshops 3.1.3 und die daraus entwickelten Anforderungen an die Visualisierung der Daten 3.3. Mithilfe des Designtools Figma wird ein GUI Prototyp erstellt. Die GUI orientiert sich an dem Design des MEDICALPADs, einem Einsatzdokumentationsgerät von WEINMANN [31]. Dieses basiert auf einem in Abbildung 4.7 dargestellten Tablet mit Windows Betriebssystem. Ein solches steht für die Arbeit zur Verfügung und wird für die Darstellung der GUI genutzt. Die visualisierten Smart Wearable Daten sind im Prototyp hinterlegt und werden unter dem extra Reiter Wearable Daten angezeigt.

Um eine Erprobung in einem Testszenario zu ermöglichen, wird eine universelle Berechtigung und Schnittstelle des Systems zu den Wearable Daten anhand der GUI simuliert, indem diese den User durch einen in Abbildung 4.8 dargestellten, simulierten Autorisierungsvorgang führt. Um diesen zu starten, wird auf einen Button „Smart Wearable Daten von Patienten einlesen“ gedrückt. Anschließend wird dazu aufgefordert das Gerät in die Nähe des Patienten Smartphones zu bringen. Dieser Vorgang soll eine Autorisierung am Smartphone über NFC und eine Verbindung mit einer anderen Übertragungstechnologie, die für größere Datenmengen geeignet ist, simulieren. Auf eine simulierte Verarbeitungszeit von 30 Sekunden folgt die Darstellung der hinterlegten visualisierten Daten.



Abbildung 4.7: Hardware des MEDICALPAD [30]

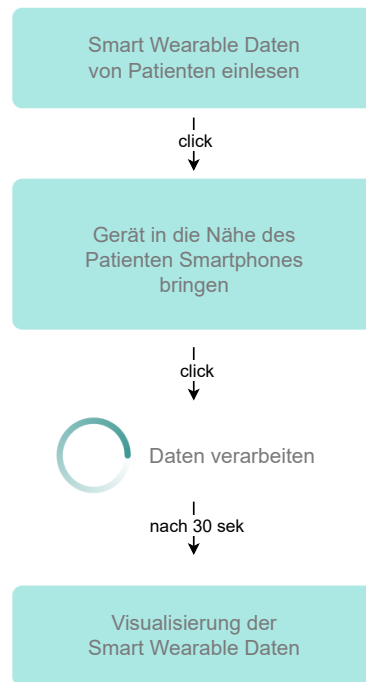


Abbildung 4.8: Ablauf der in der GUI simulierten Autorisierung

Die Darstellung der visualisierten Daten lässt sich grob in drei Bereiche aufteilen. Diese werden nachfolgend beispielhaft gezeigt. Eine vollständige Ansicht der GUI ist in Abbildung 4.12 dargestellt.

1. **Grundgerüst der GUI:** Für die Integration in ein bestehendes System wird das Grundgerüst der GUI des MEDICALPADs übernommen und um den Reiter Wearable Daten erweitert. Die Kopf- und Navigationsleiste mit Wearable Daten Reiter ist in Abbildung 4.9 dargestellt.
2. **Visualisierte Wearable Daten:** Die visualisierten Wearable Daten, mit markierten Abweichungen, werden geordnet nach ABCDE Schema dargestellt. Es gibt für jeden Datensatz eine anwählbare Option für eine Übernahme in das Protokoll. Für detaillierter verfügbare Daten ist eine Zoom Funktion verfügbar, die von einer Wochenansicht in eine Tagesansicht und von einer Tagesansicht in eine Stundenansicht führt. Beispielhaft sind in Abbildung 4.10 die SpO2 Daten dargestellt.

3. **Alarmierung von Abweichungen:** Alle erkannten Abweichungen werden gesammelt in dem in Abbildung 4.11 dargestellten Alarmfeld beschrieben. Zusätzlich werden abweichende Datenpunkte in den Visualisierungen farblich markiert.



Abbildung 4.9: Grundgerüst der GUI mit Wearable Daten Reiter

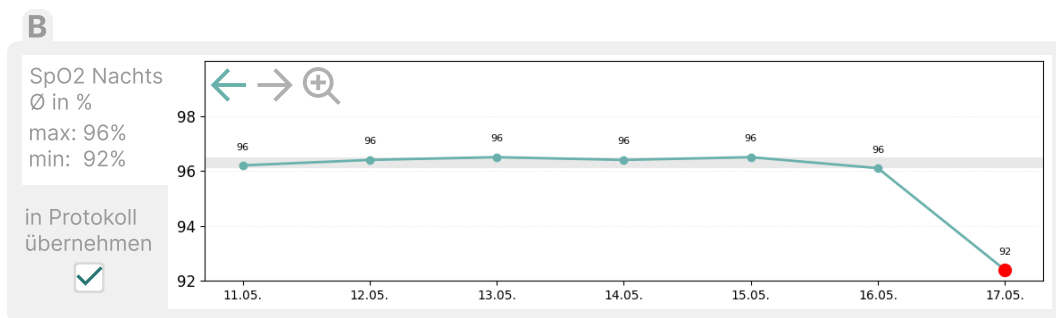


Abbildung 4.10: Visualisierte SpO2 Daten eingeordnet in B - Breathing



Abbildung 4.11: Alarmfeld zur Übersicht der Abweichungen

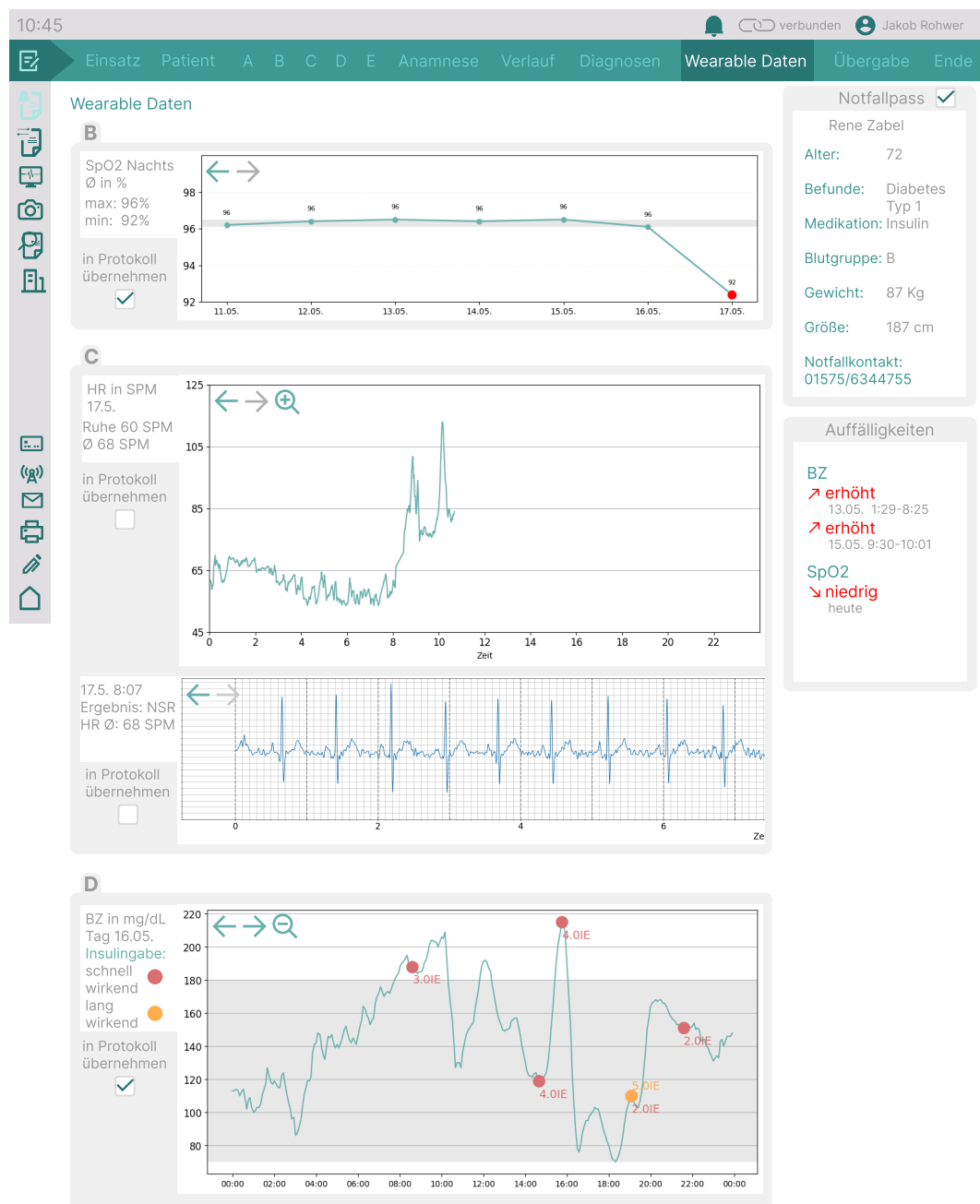


Abbildung 4.12: Vollständige Ansicht des entwickelten GUI Prototyps

5 Evaluierung

Für die Evaluierung der aufgestellten These und des entwickelten Konzepts wird in diesem Kapitel die Gestaltung und Durchführung von zwei Testszenarien beschrieben. In den Testszenarien wird der entwickelte GUI Prototyp verwendet. Anschließend erfolgt die Auswertung der Testszenarien und eine Überprüfung der Anforderungen 3.3. Auf dieser Basis wird abschließend das Konzept und die These diskutiert.

5.1 Gestaltung und Durchführung der Testszenarien

Basierend auf den in den Workshops 3.1.3 erarbeiteten Fallbeispielen 3.3 werden 2 Szenarien ausgestaltet, die von insgesamt vier Experten durchlaufen werden. Für die Szenarien werden die vom System erfassten Daten passend zu den Diagnosen des Patienten manipuliert und mit der entwickelten GUI dargestellt. Tabelle 5.1 und 5.2 beschreiben über den Patienten die Ausgangssituation des Szenarios. Den teilnehmenden Experten (als Sanitäter 1 und 2 bezeichnet) werden Aufgaben zugewiesen. Zusätzlich wird vorgegeben wann der Prototyp innerhalb des Szenarios eingesetzt werden soll.

Die Szenarien werden insgesamt zweimal mit jeweils zwei Experten durchgeführt. Die Experten werden vor der Durchführung der Szenarien mit dem Prototyp vertraut gemacht. Der erste Durchgang wird ohne medizinische Geräte durchgeführt, im zweiten Durchgang werden diese dann verwendet um die Durchführung für die Teilnehmer zu erleichtern. Ein studentischer Praktikant simuliert den Patienten.

Im Szenario Sturzereignis ohne Fremdeinwirkung wird der Prototyp, von dem Sanitäter der für die Wundversorgung und das Monitoring zuständig ist, an der in den Workshops 3.1.3 identifizierten Stelle 3.3a eingesetzt. Im Szenario Hypoglykämie wird der Prototyp von dem Sanitäter, der für die Anamnese und Untersuchung zuständig ist, so früh wie möglich eingesetzt.

Tabelle 5.1: Testszenario: Sturzereignis ohne Fremdeinwirkung durch Dehydration

Patient	Alter	72
	Vorerkrankungen	Diabetes Typ 1
	Zustand	kleine Platzwunde am Kopf, ansprechbar, leicht verwirrt, schlechte Erinnerung an Zeitraum vor dem Sturz
	Hergang	Dehydration > 24 h Schwarz vor Augen - gestürzt Nach Sturz Wasser getrunken und hingesezt - Verbesserung des Zustands
	Manipulierte Werte	Herzfrequenz - steigt stark kurz vor Sturz SpO2 - sinkt leicht in letzter Nacht
Sanitäter 1	Spricht Patient an, zuständig für Anamnese und Untersuchung	
Sanitäter 2	zuständig für Wundversorgung und Monitoring, danach Einsatz des Prototyps	

Tabelle 5.2: Testszenario: Hypoglykämie durch falsche Insulindosierung

Patient	Alter	32
	Vorerkrankungen	Diabetes Typ 1
	Zustand	Benommen, gibt langsame unklare Antworten, Schwindelig, sieht verschwommen, schwitzt
	Hergang	vor 3/4 Stunde Insulin gespritzt wegen Cola (mit Cola Zero verwechselt)
	Manipulierte Werte	Herzfrequenz steigt mit Zustandsverschlechterung Blutzucker - fällt nach Insulin Gabe in kritischen Bereich ab
Sanitäter 1	Spricht Patient an, benutzt so früh wie möglich den Prototyp, zuständig für Anamnese und Untersuchung	
Sanitäter 2	Zuständig für Monitoring	

5.2 Auswertung der Testszenarien

Anschließend an die Durchführung der Szenarien wird ein Feedback Interview geführt, bei dem die Experten zu den Themenbereichen *Einordnung in den Handlungsablauf*, *dargestellte Daten*, *Aufbereitung und Kommunikation der Daten* und *Kriterien für den Einsatz* befragt werden. Die Gespräche werden aufgezeichnet und anschließend auf wiederkehrende Aussagen untersucht. Die Prävalenz der Aussagen wird gemäß der Methodik einer thematischen Analyse [7] bewertet. Das Vorgehen wird in Abbildung 5.1 gezeigt. Die Kommentare der Experten zu den vier Themenbereichen sind mit Angabe der Prävalenz in Tabelle 5.3 aufgelistet und werden anschließend bewertet. Pro Kommentar sind dabei, begrenzt durch die Anzahl der Teilnehmer, maximal 4 Nennungen möglich.

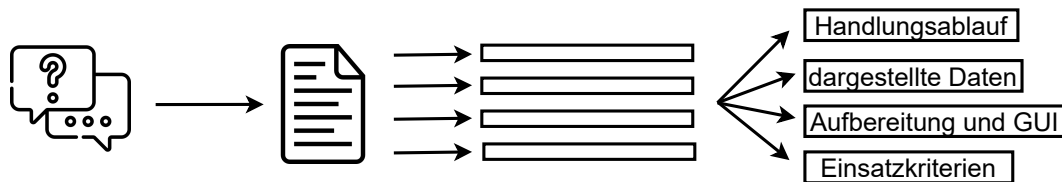


Abbildung 5.1: Ablauf der Thematischen Analyse der Interview Antworten

Tabelle 5.3: Kommentare der Experten zu den vier befragten Themenbereichen bezogen auf die Testszenarien

Einordnung in den Handlungsablauf

- Ich finde den Zeitpunkt, an dem das System in dem Szenario eingesetzt wurde, angemessen. **(4)**
- Ich finde, dass das System keine kritischen Informationen oder Hinweise für eine Therapie vor dem Einsatzzeitpunkt des Geräts beinhaltet. **(4)**
- Ich finde, dass das System bei jedem Rettungseinsatz mit Transport spätestens dort genutzt werden kann. **(2)**
- Ich finde das System kann dann genutzt werden, wenn sich Leerlauf ergibt. **(1)**

dargestellte Daten

- Ich finde, dass die Daten zur Festigung der Verdachtsdiagnose beigetragen haben. **(4)**
- Ich finde, dass durch die nicht auffälligen Daten pathologische Ereignisse in der Vergangenheit ausgeschlossen werden können. **(4)**
- Ich finde, dass ein gepflegter Notfallpass im Akutfall die wichtigsten Informationen beinhaltet. **(3)**
- Ich finde, dass die Insulingaben bei Diabetes-Patienten der wichtigste Parameter sind. **(2)**
- Ich finde die 1-Kanal EKG-Messungen können höchstens zur Einschätzung eines zu langsamen oder schnellen Puls genutzt werden. **(1)**

Tabelle 5.4: Fortsetzung: Kommentare der Experten zu den vier befragten Themenbereichen

Aufbereitung und Kommunikation der Daten

- Ich finde, dass eine Einsatzdokumentation die richtige Wahl für die Darstellung der Daten ist. **(4)**
- Ich finde, dass die simulierte Ladezeit von 30 Sekunden angemessen ist. **(3)**
- Ich finde, dass die Kategorisierung der Daten nach dem ABCDE-Schema übersichtlich ist. **(2)**
- Ich finde, dass Daten mit erkannten Abweichungen als erstes dargestellt werden sollten. **(2)**
- Ich finde, dass prägnante Werte oder eine Einschätzung des Systems gefehlt haben. **(2)**
- Ich finde, dass eine einheitliche Zeitskala für einen zeitlichen Abgleich der Daten gefehlt hat. **(2)**
- Ich finde, dass das System intuitiv bedient werden konnte. **(2)**
- Ich finde, dass das System dazu verleiten kann, sich zu wenig mit dem Patienten zu beschäftigen. **(2)**

Kriterien für den Einsatz

- Ich finde, dass das System sinnvoll ist, wenn klar ist, dass der Patient ein Wearable trägt. **(2)**
- Ich finde, dass das System sinnvoll ist, um mein Wissen über den Patienten zu erweitern. **(2)**
- Ich finde, dass das System sinnvoll ist, wenn aus den normalen diagnostischen Maßnahmen keine Verdachtsdiagnose hervorgeht. **(1)**

Einordnung in den Handlungsablauf:

Die Experten äußerten sich positiv zur Einordnung des Systems in den Handlungsablauf. Sie empfanden den Zeitpunkt des Einsatzes als angemessen, wobei auffiel dass der Prototyp, auch bei Anweisung diesen so früh wie möglich zu nutzen, in beiden Durchläufen erst genutzt wurde, nachdem eine akute Therapie und Stabilisierung durchgeführt wurde. Alle Experten waren der Meinung, dass das System keine kritischen Informationen

oder Hinweise für eine Therapie enthält. Als Zeitpunkt zum Einsatz des Systems wurden Leerlaufzeiten bzw. der Transport als spätester Zeitpunkt genannt. Die Aussagen bestätigen, dass das Konzept in den Workflow des Rettungsdienstes integriert werden kann. Da das System für die Therapie keine kritischen Informationen oder Hinweise beinhaltet sollte es erst nach der Durchführung einer Therapie eingesetzt werden.

Bewertung der dargestellten Daten:

Die Experten bewerteten die Daten positiv hinsichtlich der Festigung der Verdachtsdiagnose und der Möglichkeit, pathologische Ereignisse in der Vergangenheit auszuschließen. Es sind also nicht nur die auffälligen Daten, sondern auch die nicht auffälligen Daten wertvoll. Ein ausgefüllter Notfallpass wurde von einem Großteil der Experten als sehr wichtig empfunden, da dieser z. B. Informationen zur Medikation enthalten kann und unter Umständen eine Suche nach Hinweisen im Umfeld des Patienten ersetzt. Es wurde angemerkt, dass die Insulingaben bei Diabetes-Patienten als wichtigster Parameter angesehen werden sollten. Dabei wurde Bedenken geäußert, das Insulingaben, wenn die selbst eingetragen werden, fehlerhaft sein können. Außerdem wurden Bedenken gegenüber einer 1-Kanal EKG-Messung geäußert. Die Aussagen bekräftigen, dass die dargestellten Daten einen positiven Einfluss auf die Verdachtsdiagnose haben. Es geht außerdem hervor, dass noch genauer betrachtet werden sollte welche Informationen aus den Daten besonders hilfreich sind. Zusätzlich sollte eine Quelle der Daten für eine Einordnung der Qualität kommuniziert werden.

Aufbereitung und Kommunikation der Daten:

Alle Experten bewerteten die Wahl eines Einsatzdokumentationsgerätes zur Darstellung der Daten als positiv. Die simulierte Ladezeit wurde als angemessen empfunden. Die Kategorisierung der Daten nach dem ABCDE-Schema wurde von der Hälfte der Experten als übersichtlich empfunden, während die andere Hälfte zusätzlich eine Priorisierung nach Abweichungen befürwortet hätte. Für eine bessere Einschätzung der Daten wurde eine einheitliche Zeitskala der verschiedenen Daten und zusätzliche statistische prägnante Werte oder Einschätzungen des Systems angesprochen. Von zwei Experten wurden Bedenken geäußert, dass das System dazu verleiten kann zu wenig mit dem Patienten zu kommunizieren, da eine Beurteilung der Daten anhand von Zeitverläufen zu viel Aufmerksamkeit erfordern kann. Die Experten bewerteten die Aufbereitung der Daten grundsätzlich als positiv. Um eine schnelle Einschätzung der Daten zu verbessern, sollten prägnante Werte und erkannte Abweichungen noch mehr in den Fokus gestellt werden.

Kriterien für den Einsatz:

Als generelle Voraussetzung für einen Einsatz des Systems wurde genannt, dass ein Patient ein Smart Wearable tragen muss. Zusätzlich wurde als Kriterium genannt, dass das System sinnvoll ist, wenn das Wissen über den Patienten erweitert werden möchte oder die herkömmlichen diagnostischen Maßnahmen keine Verdachtsdiagnose ermöglichen. Diese Aussagen deuten an, dass das System grundsätzlich sinnvoll ist, wenn ein auslesbares Wearable vorhanden ist und es für die behandelnde Person hilfreich ist mehr über die pathologische Vergangenheit des Patienten zu erfahren.

5.3 Überprüfen der Anforderungen

In Tabelle 5.5 werden die Anforderungen daraufhin überprüft, ob diese durch das in Kapitel 4 umgesetzte Konzept und den, in den Testszenarios eingesetzten Prototyp, erfüllt werden. Eine Erfüllung der Anforderung wird unterschieden in Anforderung umgesetzt und Anforderung simuliert. Als umgesetzte Anforderungen werden solche bezeichnet, die im Rahmen der Konzeptrealisierung anhand eines konkreten Smart Wearables umgesetzt wurden. Als simulierte Anforderungen werden solche bezeichnet, die nicht anhand eines konkreten Smart Wearable umgesetzt werden konnten, aber für ein schlüssiges Gesamtkonzept, welches in den Testszenarien erprobt wird, notwendig sind. Diese wurden mithilfe des Prototyps simuliert.

Die Überprüfung der definierten Anforderungen an die Daten und an die Visualisierung der Daten zeigt, dass ein Großteil erfolgreich umgesetzt wurde. Anforderung A4 (Datenweitergabe an ein Krankenhaus) wurde über eine anwählbare Schaltfläche „in Einsatzprotokoll übernehmen“ simuliert. Eine Implementierung dieser Funktion ist für die Fragestellung der Arbeit nicht zwingend erforderlich.

In Bezug auf die Schnittstelle wurde ein Großteil der Anforderungen über den Prototyp simuliert. Das liegt vor allem daran, dass die Anforderungen an die Schnittstelle eine universelle Lösung für die Funktionen *Datenzugriff erlangen* und *Daten sammeln* beschreiben. Da dieser Teil des Konzepts im Programm über das Hinterlegen von User Daten und ein Zugriff auf die Daten über eine API gelöst wurde, ist ein Großteil der Anforderungen an die Schnittstelle im Rahmen dieser Arbeit nur bedingt erfüllt. Für die Testszenarien wird mit dem Prototyp eine optimale Funktion simuliert.

Die Überprüfung der Anforderungen zeigt somit, dass die Realisierung des Konzepts anhand eines konkreten Smart Wearables die Anforderungen bedingt erfüllt. Eine universelle Lösung für die Funktionen *Datenzugriff erlangen* und *Daten sammeln* konnte nicht umgesetzt werden und wird durch eine Lösung ersetzt, die nur für das ausgewählte Smart Wearable gültig ist. Der für die Testszenarios entwickelte Prototyp genügt den definierten Anforderungen. Der Prototyp simuliert alle Anforderungen die im Rahmen dieser Arbeit nicht umgesetzt werden konnten um einen Eindruck eines funktionierenden Konzepts zu vermitteln. Dieser ist notwendig, um über die Testszenarios die These und eine Integration des Konzepts in den Regelrettungsdienst zu bewerten.

Tabelle 5.5: Überprüfung der Anforderungen

Anforderung	Erfüllt	Kommentar
Anforderungen an die Daten:		
A1	Umgesetzt	Zeitliche Einordnung über die x-Achse und Legende
A2	Umgesetzt	Zeitliche Auflösung Glukose: 5 min, Herzfrequenz: 1 min
A3	Umgesetzt	zeitliche Auflösung 1 d
A4	Simuliert	optionale Übernahme in das Einsatz Protokoll
Anforderungen an die Visualisierung der Daten:		
A5	Umgesetzt	Bildschirmdiagonale des MEDICALPADs $\geq 10,1''$
A6	Umgesetzt	Darstellung mehrerer Daten untereinander in einem Fenster
A7	Umgesetzt	Einordnung der Daten in das ABCDE Schema
A8	Umgesetzt	Abweichungen von Normwerten in Alarmfeld und farbige Markierung in Zeitverläufen
A9	Umgesetzt	Zoom über Schaltfläche möglich
A10	Umgesetzt	Statische Kennzahlen werden neben den Zeitverläufen angezeigt
Anforderungen an die Schnittstelle:		
A11	Simuliert	Der Prototyp führt nach einer Startbedingung durch einen simulierten Autorisierungs- und Auslesevorgang, die Daten sind im Prototyp hinterlegt
A12	Simuliert	Nur Daten eines Fitbit mit hinterlegtem Account können ausgelesen werden, andere Daten sind wurden exportiert und hinterlegt
A13	Simuliert	Der Prototyp besitzt nur die Berechtigung auf die Daten des hinterlegten Fitbit Accounts zuzugreifen.
A14	Simuliert	Der Prototyp simuliert einen Autorisierungsvorgang über annähern an ein Smartphone (NFC)
A15	Simuliert	Das System simuliert eine Autorisierungs- und Verarbeitungszeit von 30 sek.
A16	Umgesetzt	Auslesen der Daten erfolgt mit API über Fitbit Server

5.4 Diskussion

In diesem Abschnitt erfolgt die abschließende Bewertung des Konzepts und die Diskussion der These.

5.4.1 Bewertung des Konzepts

Das in Abschnitt 3.3.2 dargestellte Konzept zur Integration von Daten eines konkreten Smart Wearable in den Rettungsdienst wurde in Kapitel 3 umgesetzt. Im Rahmen der Testszenarios konnte beispielhaft für das gewählte Fitbit Charge 5 gezeigt werden wie eine Integration von Wearable Daten in den Rettungsdienst aussehen kann.

Eine Umsetzung der Anforderungen an die Schnittstelle wurde aus Zeitgründen und aus Gründen der Zielsetzung der Arbeit ausgeschlossen. Die identifizierten Ausschlusskriterien für die Funktionen *Datenzugriff erlangen* und *Daten sammeln* konnten nicht erfüllt werden, da es bislang keine Möglichkeit gibt, dass eine dritte Person in einer Notfallsituation eine Berechtigung für einen Zugriff auf Smart Wearable Daten erhält. Diese ist die Grundvoraussetzung für eine Umsetzung eines Konzepts zur Integration von Wearable Daten in den Rettungsdienst. Die Funktion *Datenzugriff erlangen* wurde über das Hinterlegen von User Daten gelöst. Diese müssten bei einem Rettungseinsatz abgefragt werden, dies verstößt gegen das Ausschlusskriterium, dass ein Notfallpatient für den Zugriff nicht beteiligt sein darf. Die Funktion *Daten sammeln* wurde über ein Zugriff über die Fitbit API gelöst. Hierbei wird eine vorherige Autorisierung für jedes konkrete Gerät benötigt und ist daher nicht allgemeingültig. Die Problematik der Zugriffsmöglichkeit kann im Rahmen der Arbeit nicht weiter aufgelöst werden und wird abschließend in Abschnitt 6.2 diskutiert.

Das Konzept erfüllt 9 von 10 der Anforderungen an die Daten und die Visualisierung der Daten. Aus den Testszenarien geht hervor, dass ermittelte Auffälligkeiten noch mehr in den Fokus gestellt werden sollten. Im Rahmen der Arbeit wurden Schwellenwerte für die Alarmierung von Abweichungen festgelegt. Diese Methode könnte bei einer ausreichenden Datenlage durch maschinelles Lernen erweitert werden. Somit könnten komplexe Zusammenhänge der Daten erfasst und bestimmten Diagnose zugeordnet werden. Zusätzlich würde sich durch eine Einordnung jedes Datenverlaufs in auffällig oder nicht auffällig die Zeit reduzieren die eine Person in die Analyse der Daten investieren muss. An dieser Stelle muss genau identifiziert werden ob das System zur Visualisierung aller

Daten genutzt genutzt werden soll, damit sich behandelnde Personen einen umfassenden Eindruck über die pathologische Vergangenheit des Patienten machen können, oder ob das System möglichst schnell die wichtigsten pathologischen Events aus den Datensätzen kommunizieren sollen.

Die Integration in ein bestehendes Einsatzdokumentationsgerät konnte über die GUI simuliert werden. Die Bedienung wurde als intuitiv empfunden. Durch die gleichzeitige Darstellung von mehreren Zeitverläufen auf dem Einsatzdokumentationsgerät, lässt sich nicht vermeiden, dass nicht alle Zeitverläufe auf einen Blick sichtbar sind. Mit Blick in die Zukunft sind zusätzliche Daten wie z. B. Blutdruck und Beschleunigungskräfte denkbar, die das Problem verschärfen. Der Fokus sollte hier darauf liegen eine zeitliche Vergleichbarkeit zwischen den Datenverläufen zu ermöglichen. Hierfür sollten darauf geachtet werden, dass alle Zeitverläufe eine identische Zeitachse haben. Die Zoom Funktion sollte alle Datenverläufe gleichzeitig anpassen um eine Vergleichbarkeit beizubehalten. Bei entsprechenden Auffälligkeiten sollten Datenverläufe in der Visualisierung nach oben rutschen, damit Zusammenhänge der auffälligen Verläufe besser erkennbar sind.

Die Daten eines Lifestyle Smart Wearables bieten mit der Herzfrequenz, der SpO2 und den EKG-Messungen, Anhaltspunkte für eine Festigung der Verdachtsdiagnose. Begeisterung bei den Experten haben der Notfallpass und die Glukosdaten hervorgerufen. Der Notfallpass beinhaltet mit Informationen über Allergien und Vorerkrankungen als einziger potenziell relevante Informationen für den Zeitpunkt der Therapie. Dieser wurde über die Fitbit Profil Daten simuliert. Im Normalfall ist dieser über das Smartphone oder als digitaler Notfallpass über die ePA verfügbar. Bei Diabetes Erkrankungen wurden die Glukosdaten und die damit einhergehenden Glukosegaben als besonders relevant empfunden. Es bietet sich an, die Daten eines Lifestyle Wearables um weitere Daten zu ergänzen. Ein Notfallpass ist potenziell für jeden Notfallpatienten verfügbar. Bei Notfallpatienten mit speziellen Erkrankungen wie z. B. Diabetes können spezifische Daten verfügbar sein, die in diesen Fällen von Relevanz sind.

Zusammenfassend lässt sich sagen, dass das umgesetzte Konzept zur Integration eines konkreten Smart Wearables, aufgrund der positiven Bewertung, für einen Einsatz in einem Rettungsszenario geeignet ist. Durch eine Simulation der Schnittstelle konnte gezeigt werden, dass das Konzept für die Verarbeitung und Darstellung der Daten eine gute Grundlage bietet. Zusätzlich wurden Möglichkeiten für die Verbesserung der Verarbeitung und Darstellung der Daten identifiziert. Ein Lösungsansatz für die nicht um-

gesetzten Ausschlusskriterien für die Schnittstelle, auf Basis der ePA wird in Abschnitt 3.3.2 beschrieben und in Abschnitt 6.2 diskutiert.

5.4.2 Diskussion der These

In der Konkretisierung der Aufgabenstellung (Abs. 1.1) wurde die These aufgestellt, dass die Daten von Smart Wearables, integriert in gängige Handlungsabläufe im Rettungsdienst, die Erstbeurteilung oder Verdachtsdiagnose eines Notfallpatienten unterstützen können. Die These wird in diesem Abschnitt überprüft und diskutiert.

Die im Rahmen der Anforderungsanalyse 3.1 durchgeführten Workshops haben gezeigt, dass Smart Wearables Daten erheben, die für eine Erstbeurteilung und Verdachtsdiagnose hilfreich sein könnten. Anhand des in den Testszenarien erprobten GUI Prototyp konnte gezeigt werden, dass die Daten aufbereitet, auf einem Einsatzdokumentationsgerät, in ein Rettungsszenario integriert werden können. Hierbei konnte gezeigt werden, dass die dargestellten Daten keine Informationen oder Hinweise für eine Therapie beinhaltet haben. Die Therapie beruht immer auf einer Erstbeurteilung des Patienten. Dementsprechend haben die Daten für eine Erstbeurteilung keine Relevanz. Eine Ausnahme bildet der Notfallpass, wenn dieser Informationen zu Allergien oder Vorerkrankungen enthält.

Zusätzlich konnte gezeigt werden, dass ein Einsatzdokumentationssystem, welches Smart Wearable Daten ausliest und visualisiert, erst nach einer Stabilisation und eines Monitoring des Patienten in den Handlungsablauf eines Rettungsszenarios integriert werden kann. Hier können die Daten dafür genutzt werden, die Ursachen zu identifizieren, aus denen der Zustand des Patienten resultiert. Existiert bereits ein Verdacht, können die Daten nach dazu passenden, pathologischen Events untersucht werden. Auch das nicht auftreten von pathologischen Events, die nicht zu der Verdachtsdiagnose passen ist von Relevanz. Somit können die dargestellten Daten eine Verdachtsdiagnose unterstützen, in dem die Aussagen des Patienten belegt oder pathologische Ereignisse in der Vergangenheit durch die Daten ausgeschlossen werden können.

Zusammenfassend lässt sich sagen, dass die Daten eines Lifestyle Wearables, integriert in den Handlungsablauf, die Verdachtsdiagnose unterstützen können. Erweitert man die Daten um medizinische Wearables, die bei konkreten Erkrankungen eingesetzt werden, können diese ebenfalls zur Unterstützung der Verdachtsdiagnose beitragen. Wenn die Daten um den Notfallpass erweitert werden und dieser Informationen zu Allergien oder

Vorerkrankungen beinhaltet, sind die Daten bereits für eine Erstbeurteilung und Therapie von Bedeutung.

6 Zusammenfassung und Ausblick

Nach der Bewertung des Konzepts und der positiven Beantwortung der These wird in diesem abschließenden Kapitel die Arbeit zusammengefasst und ein Ausblick gegeben.

6.1 Zusammenfassung

Aktuell sind die von Smart Wearables erfassten Patientendaten nicht einheitlich für den Rettungsdienst nutzbar. Notfallpatienten besitzen auf ihren Smartphones Datensätze, die potenziellen Nutzen für die Einschätzung des Patienten in einer Notfallsituation haben.

In dieser Arbeit wird ein Konzept entwickelt und evaluiert, welches es ermöglicht, die Daten von Smart Wearables in den Rettungsdienst zu integrieren. Das entwickelte Konzept wird dazu genutzt die aufgestellte These zu untersuchen, ob die Daten von Smart Wearables, wenn diese in gängige Handlungsabläufe des Rettungsdiensts integriert werden, die Erstbeurteilung oder Verdachtsdiagnose eines Notfallpatienten unterstützen.

Anhand von Experteninterviews und Workshops wird im Rahmen einer Anforderungsanalyse zunächst untersucht, welche Smart Wearable Daten für den Rettungsdienst brauchbar sind, an welcher Stelle diese in die gängigen Handlungsabläufe integriert werden können und wie eine Integration aussehen könnte. Die analysierten Informationen werden in den Anforderungen an das Konzept festgehalten. Anhand der Anforderungen wird das Konzept in Teilfunktionen gegliedert. Für diese werden Lösungen ausgewählt und zu Lösungsvarianten kombiniert. Aufgrund der im Rahmen dieser Arbeit als nicht umsetzbar identifizierten Funktion einer Autorisierung des Rettungsdiensts für einen Datenzugriff, ohne Beteiligung des Patienten, wird eine allgemeingültige optimale Lösungsvariante beschrieben, die einen Lösungsansatz für die Problematik beinhaltet. Zusätzlich wird eine Lösungsvariante ausgewählt, die für ein konkretes Smart Wearable gültig ist.

Damit das Konzept umgesetzt werden kann, beinhaltet dieses eine nicht optimale Autorisierungsfunktion. Für das für ein konkretes Smart Wearable gültige Konzept wird anschließend ein Smart Wearable ausgewählt. Dieses wird dazu genutzt, beispielhaft die Daten anhand der realisierten Teilfunktionen des Konzepts einzulesen, aufzubereiten und zu visualisieren. Für die Erprobung des Konzepts und die Diskussion der These wird ein GUI Prototyp entwickelt, der die visualisierten Daten des Smart Wearable für den Rettungsdienst greifbar macht. Der Prototyp wird von Experten in ausgewählten Rettungsszenarien erprobt und bewertet. Eine Analyse der Bewertung zeigt, dass die dargestellten Daten eine Verdachtsdiagnose unterstützen können, jedoch für eine Erstbeurteilung und eine daraus resultierende Therapie keine relevanten Informationen enthalten. Die positive Bewertung zeigt ein Interesse der Experten an einem solchen System. Für die Verarbeitung und Kommunikation der Daten konnten brauchbare Ansätze gezeigt und Verbesserungsvorschläge identifiziert werden. Eine allgemeine Zugriffsberechtigung auf die Daten, wurde als Herausforderung identifiziert. Im Rahmen dieser Arbeit wurde diese Funktion simuliert und Lösungsansätze aufgezeigt.

6.2 Ausblick

In dieser Arbeit konnte gezeigt werden, dass Smart Wearables Daten erheben, die die Verdachtsdiagnose eines Notfallpatienten in einem Rettungsszenario unterstützen. Zusätzlich konnte anhand eines konkreten Smart Wearable beispielhaft gezeigt werden, wie eine Aufbereitung und Integration von Wearable Daten für den Rettungsdienst aussehen kann.

Im Folgenden werden Aspekte aufgegriffen die im Laufe der Bearbeitung der Arbeit aufgekommen und offengeblieben sind.

Ein wichtiger Aspekt, der in dieser Arbeit nicht explizit behandelt wurde, ist die Datenqualität der Smart Wearables. Mit dem Ziel, Daten von verschiedenen Smart Wearables nutzen zu können, ist man mit unterschiedlichen Datenqualitäten konfrontiert. Für einige Daten, wie z. B. die EKG-Messungen der führenden Hersteller, gibt es bereits medizinische Zertifizierungen. Die Studienlage zu der Datenqualität von Smart Wearables der führenden Hersteller ist umfangreich. Für die medizinische Nutzung von den Wearable Daten aus unterschiedlichen Quellen und mit unterschiedlicher Datenqualität (auch nicht medizinisch zertifizierte Daten), braucht es eine einheitliche Regelung die

genau beschreibt in welchem Maße die Daten genutzt werden können und dürfen. Eine Nutzung zur Unterstützung der Verdachtsdiagnose, die durch den Einsatz gängiger Diagnoseverfahren bestätigt wird, ist definitiv denkbar.

Ein Aspekt, der in dieser Arbeit als Herausforderung identifiziert wurde, ist die Notwendigkeit einer Autorisierung für einen Datenzugriff ohne die Beteiligung des Patienten und eines universellen Datenzugriffs für Geräte verschiedener Hersteller. Mit Anbietern, z. B. TERRA [28], die APIs entwickeln, die einen Zugriff auf unterschiedlichste Smart Wearables ermöglichen gibt es einen Ansatz für einen universellen Datenzugriff. Eine Möglichkeit einer Autorisierung des Rettungsdiensts für einen Datenzugriff gibt es bis jetzt nicht. Einen Ansatz hierfür bietet der digitale Notfalldatensatz der ePA, auf den mit einem Heilberufsausweis ohne Beteiligung des Patienten zugegriffen werden kann. Eine Umsetzung einer ähnliche Zugriffsberechtigung für den Rettungsdienst, bildet die Voraussetzung für die Nutzung von Smart Wearable Daten in einem Rettungsszenario.

Ein Einsatz von Smart Wearable Daten als Live-Monitoring für den Rettungsdienst, ist mit dem heutigen Stand der Technik nicht denkbar. Abhängig davon an welcher Stelle die Smart Wearable Daten abgegriffen werden, ergibt sich durch die Synchronisierungsintervalle zwischen Wearable und Smartphone und Smartphone und Server (s. Abb. 4.2) eine nicht reproduzierbare Latenzzeit, von mindestens einer Minute. Diese wurde im Zuge der Arbeit nicht weiter untersucht. Um Smart Wearables für ein Live-Monitoring in Betracht zu ziehen, fehlen noch folgende kontinuierliche Messungen: Blutdruck, EKG, SpO₂, Glukose. Werden diese Werte in Zukunft abgedeckt und medizinisch zertifiziert, ist unter der Voraussetzung einer Verbindung mit geringer Latenzzeit eine Live-Monitoring denkbar.

Die umgesetzte Methode zur Erkennung von Abweichungen basiert auf simplen Schwellenwerten. Eine Erweiterung dieses Ansatzes mittels maschinellen Lernens würde es ermöglichen, komplexe Zusammenhänge der Datenverläufe zu erkennen. Hierbei ist nicht nur eine Analyse der relevanten Notfalldaten denkbar, sondern auch deren Kombination mit anderen Daten, um Hinweise auf konkrete Diagnosen zu erlangen. Ein einfaches Beispiel hierfür wäre die Feststellung einer erhöhten Herzfrequenz bei geringer körperlicher Aktivität des Patienten. Angesichts der weit verbreiteten Nutzung von Smart Wearables und der Fülle an gesammelten Daten ist es realistisch, ein System zu trainieren, wenn entsprechende Datensätze mit den aufgetretenen Notfalldiagnosen gelabelt werden.

Der Ausblick auf zukünftige Entwicklungen im Bereich der Smart Wearables, Datenauswertung durch künstliche Intelligenz und der ePA zeigt eine vielversprechende Perspek-

tive für die Nutzung von Smart Wearable Daten im Rettungsdienst. Es sind jedoch noch einige Herausforderungen zu bewältigen, darunter die Gewährleistung der Datenqualität, die Entwicklung eines universellen Datenzugriffs und die Umsetzung einer Autorisierung für den Rettungsdienst.

Literaturverzeichnis

- [1] : PHTLS®: *Prehospital Trauma Life Support*. 2022. – URL <https://www.dgu-online.de/bildung/fortbildung/phtlsr-prehospital-trauma-life-support>. – Zugriffsdatum: 13.06.2023
- [2] AKTIIA: *Aktiia - Langzeit-Blutdruckmessung für zu Hause*. 28.02.2023. – URL <https://aktiia.com/de/>. – Zugriffsdatum: 21.04.2023
- [3] AWMF e.V.: *Leitlinien der AWMF für Ärzt*innen und Gesundheitspersonal / Arbeitsgemeinschaft der Wissenschaftlichen Medizinischen Fachgesellschaften e. V.* 13.06.2023. – URL <https://www.awmf.org/leitlinien>. – Zugriffsdatum: 13.06.2023
- [4] BAYOUMY, Karim ; GABER, Mohammed ; ELSHAFEEY, Abdallah ; MHAIMEED, Omar ; DINEEN, Elizabeth H. ; MARVEL, Françoise A. ; MARTIN, Seth S. ; MUSE, Evan D. ; TURAKHIA, Mintu P. ; TARAKJI, Khaldoun G. ; ELSHAZLY, Mohamed B.: Smart wearable devices in cardiovascular care: where we are and how to move forward. In: *Nature reviews. Cardiology* 18 (2021), Nr. 8, S. 581–599
- [5] BENDER, Beate ; GERICKE, Kilian: *Pahl/Beitz Konstruktionslehre*. Berlin, Heidelberg : Springer Berlin Heidelberg, 2021. – ISBN 978-3-662-57302-0
- [6] BIELER, Dan ; GOOSSEN, Käthe: 187-0231_S3_Polytrauma-Schwerverletzten-Behandlung_2023-02. . – URL https://register.awmf.org/assets/guidelines/187-0231_S3_Polytrauma-Schwerverletzten-Behandlung_2023-02.pdf. – Zugriffsdatum: 21.04.2023
- [7] BRAUN, Virginia ; CLARKE, Victoria: Thematic analysis. In: COOPER, Harris (Hrsg.) ; CAMIC, Paul M. (Hrsg.) ; LONG, Debra L. (Hrsg.) ; PANTER, A. T. (Hrsg.) ; RINDSKOPF, David (Hrsg.) ; SHER, Kenneth J. (Hrsg.): *APA handbook of research methods in psychology, Vol 2: Research designs: Quantitative, qualitative, neuropsychological, and biological*. Washington : American Psychological Association, 2012, S. 57–71. – ISBN 1-4338-1005-0

- [8] BRENNER, Sebastian: Erkennen und Behandeln reversibler Ursachen. In: *Pädiatrie* 30 (2018), Nr. S1, S. 28–30. – ISSN 1867-2132
- [9] DEUTSCHLAND: *Apple Watch Series 8*. 09.05.2023. – URL <https://www.apple.com/de/apple-watch-series-8/>. – Zugriffsdatum: 09.05.2023
- [10] FITBIT: *Fortschrittlicher Gesundheits- und Fitness-Tracker | Fitbit Charge 5*; 08.05.2023. – URL <https://www.fitbit.com/global/de/products/trackers/charge5?sku=421BKBK>. – Zugriffsdatum: 09.05.2023
- [11] FREESTYLE LIBRE: *FreeStyle Libre | Das Glukosemesssystem der nächsten Generation*. 19.05.2023. – URL <https://www.freestylelibre.de/>. – Zugriffsdatum: 19.05.2023
- [12] GEMATIK: *E-Patientenakte | gematik*. 25.05.2023. – URL <https://www.gematik.de/anwendungen/e-patientenakte>. – Zugriffsdatum: 25.05.2023
- [13] GEMATIK: *Notfalldaten | gematik*. 25.05.2023. – URL <https://www.gematik.de/anwendungen/notfalldaten>. – Zugriffsdatum: 25.05.2023
- [14] GERSTBACH, Ingrid: *77 Tools für Design Thinker: Insidertipps aus der Design-Thinking-Praxis*. 2. Auflage. Offenbach : GABAL, 2018 (Whitebooks). – ISBN 3869368055
- [15] GITHUB: *GitHub - orcasgit/python-fitbit: Fitbit API Python Client Implementation*. 19.05.2023. – URL <https://github.com/orcasgit/python-fitbit>. – Zugriffsdatum: 19.05.2023
- [16] HAN, Janet K. ; AL-KHATIB, Sana M. ; ALBERT, Christine M.: Changes in the digital health landscape in cardiac electrophysiology: A pre-and peri-pandemic COVID-19 era survey. In: *Cardiovascular digital health journal* 2 (2021), Nr. 1, S. 55–62
- [17] HOHENEGGER, Mario: Leben retten mit System. In: *Heilberufe* 67 (2015), Nr. 9, S. 24–26. – ISSN 0017-9604
- [18] KIM, Nam H. ; KO, Jum S.: Introduction of Wearable Device in Cardiovascular Field for Monitoring Arrhythmia. In: *Chonnam Medical Journal* 57 (2021), Nr. 1, S. 1. – ISSN 2233-7385
- [19] KOPPENBERG, J. ; BRIGGS, S. M. ; WEDEL, S. K. ; CONN, A. K.: Das amerikanische Notfallwesen - "emergency medical service" und "emergency room". In: *Notfall & Rettungsmedizin* 5 (2002), Nr. 8, S. 598–605. – ISSN 1434-6222

-
- [20] NISSEN, Michael ; SLIM, Syrine ; JÄGER, Katharina ; FLAUCHER, Madeleine ; HUEBNER, Hanna ; DANZBERGER, Nina ; FASCHING, Peter A. ; BECKMANN, Matthias W. ; GRADL, Stefan ; ESKOFIER, Bjoern M.: Heart Rate Measurement Accuracy of Fitbit Charge 4 and Samsung Galaxy Watch Active2: Device Evaluation Study. In: *JMIR formative research* 6 (2022), Nr. 3, S. e33635
- [21] NOTFALL-ID: *Notfallarmband SOS Armband ID für Erwachsene, Kinder und Senioren*. 16.01.2022. – URL <https://notfall-id.de/sos-id-notfallarmband/>. – Zugriffsdatum: 04.06.2023
- [22] PEREZ, Marco V. ; MAHAFFEY, Kenneth W. ; HEDLIN, Haley ; RUMSFELD, John S. ; GARCIA, Ariadna ; FERRIS, Todd ; BALASUBRAMANIAN, Vidhya ; RUSSO, Andrea M. ; RAJMANE, Amol ; CHEUNG, Lauren ; HUNG, Grace ; LEE, Justin ; KOWEY, Peter ; TALATI, Nisha ; NAG, Divya ; GUMMIDIPUNDI, Santosh E. ; BEATTY, Alexis ; HILLS, Mellanie T. ; DESAI, Sumbul ; GRANGER, Christopher B. ; DESAI, Manisha ; TURAKHIA, Mintu P.: Large-Scale Assessment of a Smartwatch to Identify Atrial Fibrillation. In: *The New England journal of medicine* 381 (2019), Nr. 20, S. 1909–1917
- [23] POHL, Klaus ; RUPP, Chris: *Basiswissen Requirements Engineering: Aus- und Weiterbildung nach IREB-Standard zum Certified Professional for Requirements Engineering Foundation Level*. 5., überarbeitete und aktualisierte Auflage. Heidelberg : dpunkt.verlag, 2021. – ISBN 978-3-86490-814-9
- [24] SAMSUNG DE: *Galaxy Watch5 & Watch5 Pro Bespoke Kaufen | Preis & Angebote | Samsung Deutschland*. 2023. – URL <https://www.samsung.com/de/watches/galaxy-watch5/buy/>. – Zugriffsdatum: 09.05.2023
- [25] STATISTA: *Anteil der Nutzer von Fitnesstrackern in Deutschland 2019 | Statista*. 08.05.2023. – URL <https://de.statista.com/statistik/daten/studie/1047599/umfrage/anteil-der-fitnesstracker-nutzer-in-deutschland/>. – Zugriffsdatum: 08.05.2023
- [26] STATISTA: *Anzahl der Smartphone-Nutzer in Deutschland | Statista*. 08.05.2023. – URL <https://de.statista.com/statistik/daten/studie/198959/umfrage/anzahl-der-smartphonennutzer-in-deutschland-seit-2010/>. – Zugriffsdatum: 08.05.2023

- [27] STATISTA: *Smartwatch-Marken in Deutschland 2022 | Statista*. 09.05.2023. – URL <https://de.statista.com/prognosen/999765/deutschland-beliebteste-smartwatch-marken>. – Zugriffsdatum: 09.05.2023
- [28] TERRA: *Terra - Fitness and Health API to connect all wearables to your app*. 27.05.2023. – URL <https://tryterra.co/>. – Zugriffsdatum: 27.05.2023
- [29] VELTMANN, Christian ; EHRLICH, Joachim R. ; GASSNER, Ulrich M. ; MEDER, Benjamin ; MÖCKEL, Martin ; RADKE, Peter ; SCHOLZ, Eberhard ; SCHNEIDER, Hendrik ; STELLBRINK, Christoph ; DUNCKER, David: Wearable-basierte Detektion von Arrhythmien. In: *Der Kardiologe* 15 (2021), Nr. 4, S. 341–353. – ISSN 1864-9726
- [30] WEINMANN: *WEINMANN Emergency übernimmt Tech2go und stärkt damit seine Kompetenz im Bereich Digital Services*. 04.06.2023. – URL <https://www.weinmann-emergency.com/de/news/news-presse/weinmann-emergency-uebernimmt-tech2go-und-staerkt-damit-seine-kompetenz-im-bereich-digital-services/>. – Zugriffsdatum: 04.06.2023
- [31] WEINMANN: *MEDICALPAD | WEINMANN*. 16.05.2023. – URL <https://www.weinmann-emergency.com/de/produkte/datenmanagement/medicalpad/>. – Zugriffsdatum: 16.05.2023

A Anhang

A.1 Aufgabenstellung



Bachelor- Arbeit

Hamburg, 19. Dezember 2022

Titel:	Konzeption und Evaluation eines Systems zur Nutzung eines Smart Wearables für die Unterstützung des Regelrettungsdiensts.
Dauer:	3 Monate
Kurzbeschreibung:	<p>In der außerklinischen Rettung geht es darum einem zumeist kritisch kranken Patienten schnell zu diagnostizieren, zu stabilisieren und dann in ein Krankenhaus zur weiteren Versorgung zu transportieren. Vor allem die Diagnose stützt sich auf Berichte des Patienten und weiteren Personen, sowie den medizinischen Messdaten, welche mittels der mitgebrachten Medizintechnik erhoben werden.</p> <p>Heutzutage haben jedoch viele Personen Geräte der persönlichen Gesundheitsfürsorge. Dies geht von Uhren, welche ein EKG aufzeichnen, zu Pulsoxymetern, Temperaturmessungen, Schlafdiagnose Systemen zu Blutdruckmessungen, ohne das diese Liste einen Anspruch auf Vollständigkeit hat.</p> <p>Diese teils für die Diagnose potenziell wertvollen Daten werden bisher kaum oder nicht systematisch beachtet.</p> <p>Ziel dieser Arbeit ist es zunächst eine Übersicht zu erarbeiten welche solcher Systeme heutzutage üblich sind und über welche Schnittstellen diese verfügen, um diese Daten systematisch abfragen zu können.</p> <p>Darauf aufbauend und durch Analysen der aktuell gültigen SOPs im Rettungsdienst und Expertenbefragungen gestützt soll eine Lösung erarbeitet werden mit welcher die Erfassung und Visualisierung/ Analyse solcher Daten im Rettungsdienst ermöglicht wird. Insbesondere soll die Nutzung eines Einsatzdokumentationssystems hierfür betrachtet werden.</p> <p>Das erarbeitete Szenario/ Vorgehen ist mittels eines Demonstrators, welcher ein persönliches Messsystem umfasst, zu überprüfen.</p> <p>Die gewonnen Erkenntnisse sind in einer ausführlichen schriftlichen Arbeit zu dokumentieren.</p>
Anforderungen:	Studium der Elektrotechnik, Infotronik, Medizintechnik, Mediziningeniurswesen, Informatik oder Mechatronik mit medizinischen Grundkenntnissen

Weinmann Emergency Medical Technology GmbH + Co. KG
Frohbösestr. 12
22525 Hamburg

Ansprechpartner:

Christian Neuhaus



Steffen Schmitt



A.2 Experteninterviews

A.2.1 Fragen

1. Wie lange bist du im Rettungsdienst tätig und welche Funktion besetzt du?
2. Nach welchen Abläufen/Schemata richten sich Rettungseinsätze?
3. Wie viele und welche Personen sind an einem Einsatz beteiligt?
4. Welche Einsätze treten besonders häufig auf?
5. Welche Geräte werden bei diesen Einsatzarten genutzt?
6. Hast du im Rahmen des Rettungsdiensts Erfahrungen mit Smart Wearables gemacht?
7. Wie gehst du mit Patienten Daten von Smart Wearables um?
8. Würdest du Smart Wearable Daten einsehen und wenn ja wann?

A.2.2 Interview Protokoll

Interview 1: (Rettungsanitäter 1 Jahr Vollzeit , danach Ehrenamtlich)

- (C)ABCDE
 - Atemweg
 - Atmung
 - Kreislauf
 - Defizit
 - Ganzkörperuntersuchung
- SAMPLE - Anamnese / Verdachtsdiagnose
- 4h/HITS
- Reanimation ACLS

- RTW 1 Rettungsanitäter*in, 1 Rettungsassistent*in
- NEF 1 Notarzt , (+ 1 Fahrer*in)

- Sturz, AZ Verschlechterung, ACS besonders häufig
- fast bei jedem Einsatz EKG, Rucksack
- Einsatzdokumentationsgerät bei jedem Einsatz zur Dokumentation, aber nicht immer mit zum Patienten
- Beatmungsgerät, Absaugpumpe wenn vorher klar benötigt

- Eher wenig Berührung
 - ältere Kollegen lehnen Wearables ab, nur selbsterhobene Daten
 - Datenschutz, gibt es überhaupt Zugriff
 - ePA
- Einsicht der Daten vielleicht in ABCDE ab C, in SAMPLE sinnvoll
 - Ergänzung der Anamnese
 - ergänzt Fremdanamnese, macht diese erst möglich

Abbildung A.2: Protokoll Interview 1

Interview 2: Doppel Interview (Notfallsanitäter 8 Jahre, Rettungssanitäter 2 Jahre)

- Immer ABCDE und SAMPLE
 - Punkte werden direkt beim Eintreffen abgehakt
 - durch ersten Patientenkontakt/Umfeld
- Danach je nach Befund in spezielle Abläufe
 - Schmerzen OPQRST
 - Neurologisch FAST
 - Etc.
- RTW 1 Rettungssanitäter*in, 1 Rettungsassistent*in, NEF 1 Notarzt , 1 Fahrer
- Sturz, Abdomen, AZ Verschlechterung, ACS, Intoxikation C2 (Alkohol am Wochenende) wenn relevant
 - Viele Einsätze wo nichts passiert (immer gleiche Patienten, z.T. nur Krankentransport gewollt oder schnelle Einweisung)
- Rucksack EKG fast immer
- 90% 4 Kanal EKG, Blutdruck Manschette und Sättigungsclip
 - Sehr schnell (max. 3 Minuten)
- Rest nach Bedarf
- Notfall QR Code (im Prinzip wie Notfall Daten vom Smartphone) für ältere Patienten, Risiko Patienten
- Blutzucker Daten von Diabetes Patienten angeschaut über App, Verlauf von mehreren Tagen
- Einige Korrekte Vorhofflimmer Diagnosen von Apple Watch
- Hohe Akzeptanz gegenüber EKG Einschätzung von Apple Watch, und Freestyle Libre Sensoren (Glukose)
- Interessant: alle Daten mit längeren Verläufen, besonders Blutzucker und Blutdruck wenn möglich
- Erst angucken wenn Zeit ist
 - z.B. beim Transport wenn der Patient stabil ist
 - Interessant wenn Patient nicht ansprechbar, also kein Zugriff auf Daten wie z. B: Glukose
 - für Übergabe an KH
- Ersteinschätzung und akute Vitalparameter sehr schnell, keine Zeit und Kapazität zum angucken

Abbildung A.3: Ergebnisse Interview 2

A.3 Workshop Ergebnisse

A.3.1 Flipcharts

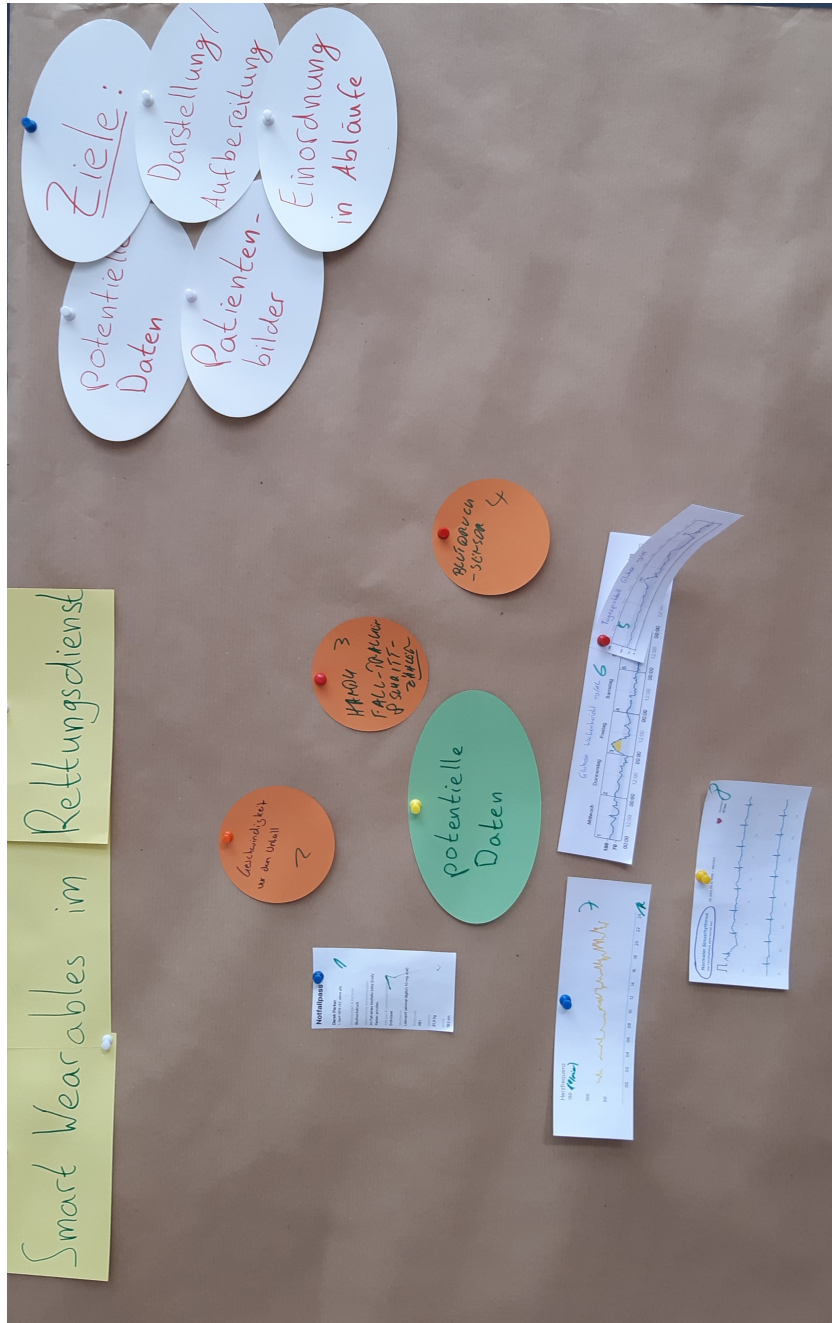


Abbildung A.4: Ergebnisse des ersten Workshops

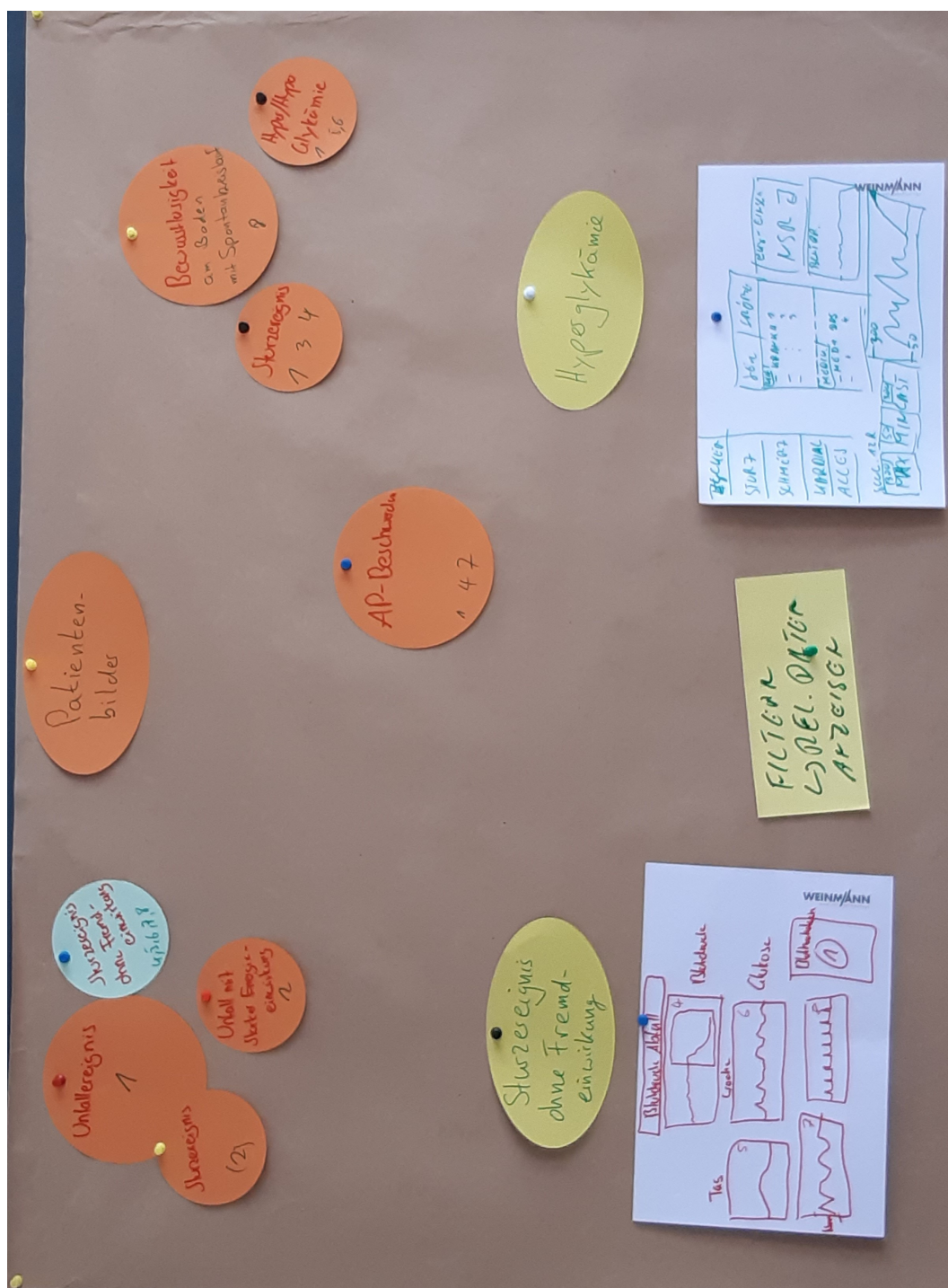


Abbildung A.5: Ergebnisse des ersten Workshops

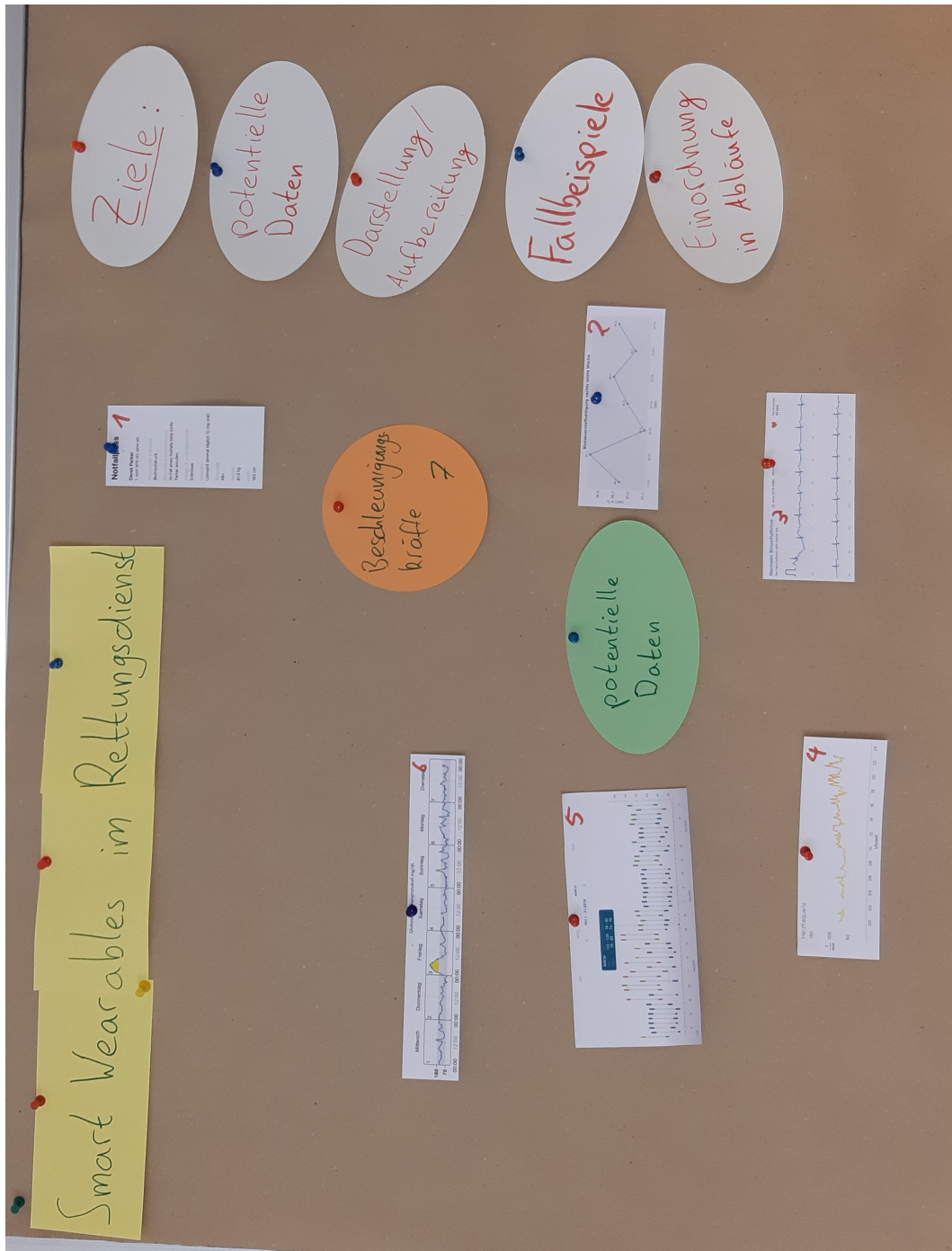


Abbildung A.6: Ergebnisse des zweiten Workshops

A.3.2 Visualisierung der Daten für ausgewählte Fallbeispiele

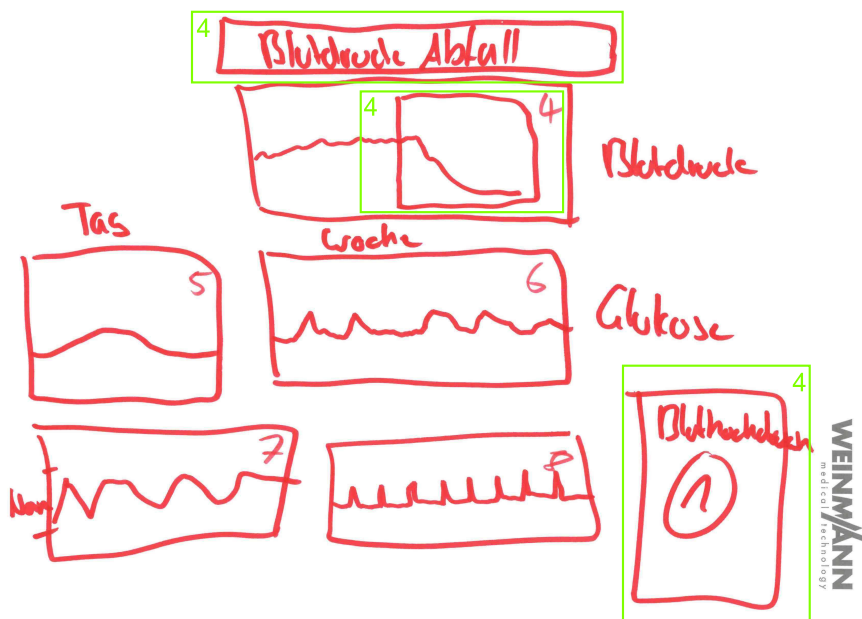


Abbildung A.8: Sturz ohne Fremdeinwirkung Visualisierung



Abbildung A.9: Hyperglykämie Visualisierung

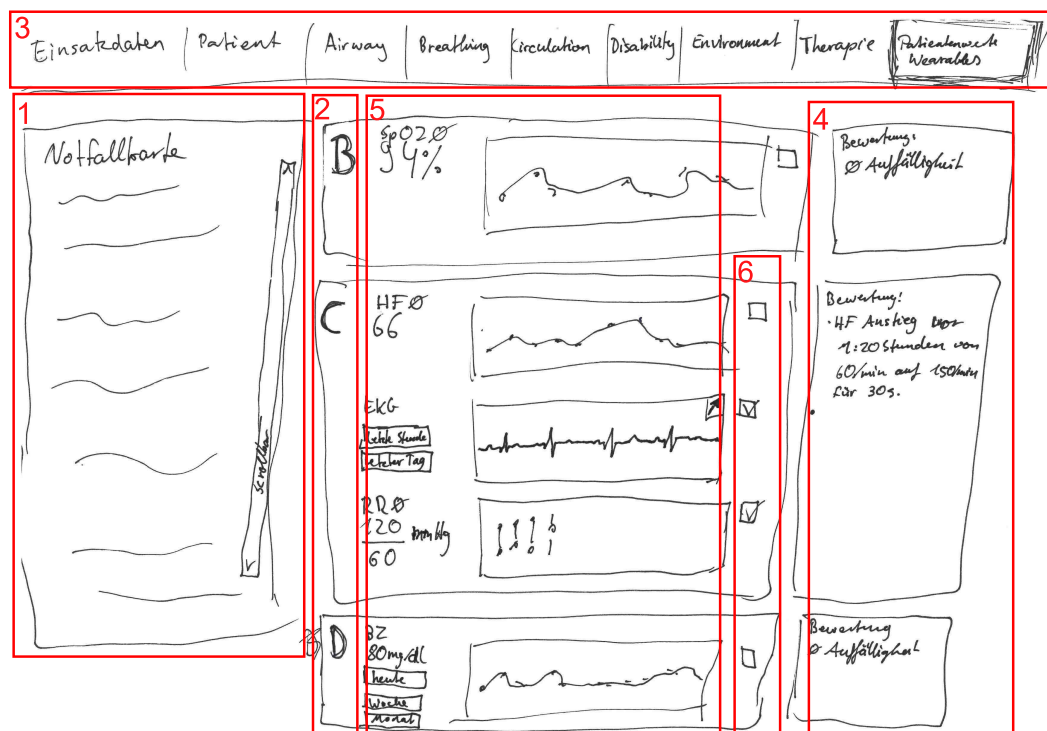


Abbildung A.10: Reanimation Visualisierung

A.4 Quellcode

A.4.1 Quellcode fitbit_data_for_EMS_usage.py

```

import matplotlib.pyplot as plt
2 import fitbit
import numpy as np
4 from numpy import genfromtxt
import os
6 # gather_keys_oauth2.py file needs to be in the same directory.
# also needs to install cherrypy: https://pypi.org/project/CherryPy/
8 # pip install CherryPy
import gather_keys_oauth2 as Oauth2
10 import pandas as pd
import datetime
12 import matplotlib.dates as mdates
import json
14 import os

```

```

16 class Authorize:
17     def __init__(self, client_id, client_secret):
18         self.client_id = client_id
19         self.client_secret = client_secret
20
21     def authorize(self):
22         server = OAuth2.OAuth2Server(self.client_id, self.client_secret)
23         server.browser_authorize()
24         access_token = str(server.fitbit.client.session.token['access_token
25         '])
26         refresh_token = str(server.fitbit.client.session.token['
27         refresh_token'])
28         auth2_client = fitbit.Fitbit(self.client_id, self.client_secret,
29         oauth2=True, access_token=access_token, refresh_token=refresh_token)
30         return auth2_client
31
32 class Time:
33     def __init__(self):
34         pass
35
36     def getTime(self):
37         # get different time formats
38         this_week = pd.date_range(pd.datetime.today().date() - datetime.
39         timedelta(days=6), pd.datetime.today().date())
40         now = datetime.datetime.now().strftime("%Y-%m-%dT%H:%M:%S")
41         return this_week, now
42
43 class Heartrate:
44     def __init__(self, auth2_client, thisWeek):
45         self.auth2_client = auth2_client
46         self.thisWeek = thisWeek
47
48     def getData(self):
49         #date for json filename
50         nowFilename = datetime.datetime.now().strftime("%Y-%m-%d_%H-%M")
51         heartRateDataList = []
52         for oneDate in self.thisWeek:
53             # Convert the date to the right format
54             date = oneDate.date().strftime("%Y-%m-%d")
55
56             # Load the HR data for the current day into a DataFrame
57             oneDayHeart = self.auth2_client.intraday_time_series('
58             activities/heart',

```

```

54                                     base_date=date,
                                       detail_level='1
min')
56     if 'restingHeartRate' in oneDayHeart['activities-heart'][0]['
value']:
        restingHeartRate = oneDayHeart['activities-heart'][0]['
value']['restingHeartRate']
58     else:
        restingHeartRate = []

60
        #prepare Data for dump in json file
62     heartrateData = {'dateTime': oneDayHeart['activities-heart'
][0]['dateTime'], 'restingHeartRate': restingHeartRate, 'dataset':
oneDayHeart['activities-heart-intraday']['dataset']}
        heartRateDataList.append(heartrateData)

64
        # save HR Data in a json file
66     with open(f"heartrate_{nowFilename}.json", "w") as f:
        json.dump(heartRateDataList, f)
68     return

70 def detectDeviations(self):
    return

72
74 def plotData(self):
    dfHeart = pd.read_json('heartrate_2023-05-12_12-27.json')
    # Loop over each day's data and plot it
76     for i, date in enumerate(dfHeart['dateTime']):
        #data for current Day
78         heartData = dfHeart.loc[i, 'dataset']

80         # convert time values to hours
        df = pd.DataFrame(heartData)
82         df['time_in_hours'] = pd.to_timedelta(df['time']).dt.
total_seconds() / 3600

84         # smooth heart rate data
        window_size = 5
86         df['smoothed_HR'] = df['value'].rolling(window_size, center=
True).mean()

88         #cut to %Y-%m-%d
        date_short = date.strftime("%Y-%m-%d")
90

```

```

92     # get min and max time to get dynamic x-axis
    min_time, max_time = np.round(min(df['time_in_hours'])), np.
round(max(df['time_in_hours']))

94     # Create the plot
    fig, ax = plt.subplots(figsize=(10, 5))
96     ax.plot(df['time_in_hours'], df['smoothed_HR'], color='#68B0AB'
, label='HR')
    ax.set_xlabel('Time')
98     ax.set_ylabel('Heart Rate')
    ax.set_title(f"Heart Rate on {date_short}")
100    ax.yaxis.grid(True)
    ax.xaxis.grid(False)
102    plt.xticks(range(25))
    plt.xlim(min_time, max_time)
104    plt.xticks(np.arange(min_time, max_time + 1, 2), fontsize=12)
    plt.ylim(45, 125)
106    plt.yticks(np.arange(45, 126, 20), fontsize=12)
    plt.tight_layout()

108    #print additional information
110    print(f"Zusaetzliche INformationen fuer Herzfrequenz am {date}"
)
    print(f"Herzfrequenz in Ruhe: {dfHeart.loc[i, 'restingHeartRate
']}")
112    print(f"Herzfrequenz Durchschnitt: {df['value'].mean()}")

114    #manipulate date time to use it as a filename
    date_short = date.strftime("%Y-%m-%d")
116    plt.savefig(f"heartrate_{date_short}.png")
    os.system(f"heartrate_{date_short}.png")

118    if date_short == datetime.datetime.now().strftime("%Y-%m-%d"):
#limits the higher resolution plots to the current Day
120    #higher resolution of current date
        for i in range(int(min_time), int(max_time), 2):
122            df_filtered = df.loc[(df['time_in_hours'] >= i) & (df['
time_in_hours'] <= i+2)]
            print('Mean of shown range:', df_filtered['value'].mean
())

124    # Erstellen Sie den Plot mit den gefilterten Daten
    plt.style.use('default')
126    fig, ax = plt.subplots(figsize=(10, 5))

```

```

    ax.plot(df_filtered['time_in_hours'], df_filtered['
smoothed_HR'], color='#68B0AB', label='HR')
128     ax.set_xlabel('Zeit')
    ax.set_title(f"Heart Rate on {date_short} Zoom")
130     fig.tight_layout()
    ax.yaxis.grid(True)
132     ax.xaxis.grid(False)
    plt.yticks(fontsize=12)
134     plt.xticks(range(i, i+2 + 1))
    plt.xlim(i, i+2)
136     plt.xticks(np.arange(i, i+2 +0.1, 1), fontsize=12)
    plt.ylim(45, 125)
138     plt.yticks(np.arange(45, 126, 20), fontsize=12)
    plt.savefig(f"heartrate_{date_short}_zoom{i}.png")
140     os.system(f"heartrate_{date_short}_zoom{i}.png")
    #print additional information
142     print(f"Zusaetzliche INformationen fuer Herzfrequenz am
    {date} von {df_filtered['time_in_hours'].iloc[0]} bis {df_filtered['
time_in_hours'].iloc[-1]}")
    print(f"Herzfrequenz Durchschnitt: {df_filtered['value
'].mean()}")
144     return

146 class Glucose:
    def __init__(self):
148         pass
    def getData(self):
150         #import glucose data
        glucoseData = genfromtxt('glucose_7-3-2023_1_Dehyd_w2.csv',
        delimiter = ',', skip_header=3, usecols=(2,4,5,7,12), converters= {2:
        lambda x: datetime.datetime.strptime(x.decode("utf-8"), '%d-%m-%Y %H:%M'
        )})
152         #convert to DF
        dfGlucose = pd.DataFrame(glucoseData)
154         #rename Columns
        dfGlucose = dfGlucose.rename({'f0': 'time', 'f1': 'value', 'f2': '
value1', 'f3': 'fInsulin', 'f4': 'sInsulin'}, axis='columns')
156         # dfGlucose_lastWeek = dfGlucose[dfGlucose['time'] > (datetime.
        datetime.now().strftime("%Y-%m-%d")-datetime.timedelta(days=13))] #read
        last 2 weeks
        self.dfGlucose_last2Week = dfGlucose[dfGlucose['time'] > (datetime.
        datetime(2023, 5, 5, 0, 0, 0))]
158
        #save glucose Data

```

```

160     glucoseData = {'time': self.dfGlucose_last2Week['time'].dt.strftime
( '%Y-%m-%d %H:%M: ' ).tolist(), 'value': self.dfGlucose_last2Week['value'
].tolist(), 'value1': self.dfGlucose_last2Week['value1'].tolist(), '
fInsulin': self.dfGlucose_last2Week['fInsulin'].tolist(), 'sInsulin':
self.dfGlucose_last2Week['sInsulin'].tolist()}
    with open("glucoseData.json", "w") as f:
162         json.dump(glucoseData, f)
    #dfGlucose_lastWeek.to_json('glucoseLastWeek.json')
164     return

166 def detectDeviations(self):
    threshold = pd.Timedelta(minutes=15)
168     self.glucose_alarm_start = []
    self.glucose_alarm_end = []
170     start_marker_high = 0
    start_marker_low = 0
172     #iterate over rows
    for i, row in self.dfGlucose_last2Week.iterrows():
174         #check if value is not null
        if pd.notnull(row['value']):
176             # check if value is greater than 230 for >30min
            if row['value'] > 230 and start_marker_high == 0:
178                 start_time_high = row['time']
                start_marker_high = 1
180             if start_marker_high == 1 and row['value'] < 231:
                if (row['time'] - start_time_high) > threshold:
182                     self.glucose_alarm_start.append(start_time_high)
                    self.glucose_alarm_end.append(self.
dfGlucose_last2Week.loc[i-1, 'time'])
184                     start_marker_high = 0

                # check if value is smaller than 71 for >30min
                if row['value'] < 71 and start_marker_low == 0:
188                     start_time_low = row['time']
                    start_marker_low = 1
190                 if start_marker_low == 1 and row['value'] > 70:
                    if (row['time'] - start_time_low) > threshold:
192                         self.glucose_alarm_start.append(start_time_low)
                            self.glucose_alarm_end.append(self.
dfGlucose_last2Week.loc[i-1, 'time'])
194                         start_marker_low = 0

                # when deviation intervall not closed, last value is still above
196     200

```

```

198         if start_marker_high == 1:
199             self.glucose_alarm_start.append(start_time_high)
200             self.glucose_alarm_end.append(self.dfGlucose_last2Week.iloc
[-1]['time'])
201
202         # when deviation intervall not closed, last value is still below 50
203         if start_marker_low == 1:
204             self.glucose_alarm_start.append(start_time_low)
205             self.glucose_alarm_end.append(self.dfGlucose_last2Week.iloc
[-1]['time'])
206         return
207
208     def plotData(self):
209         #dfGlucose = pd.read_json('glucoseData.json')
210         current_start_date = datetime.datetime(2023, 5, 5, 10, 45, 0)
211         #plot week data
212         for i in range(0,2,1):
213             current_end_date = current_start_date + datetime.timedelta(days
=6)
214             dfGlucose_current = self.dfGlucose_last2Week[(self.
dfGlucose_last2Week['time'] >= current_start_date) & (self.
dfGlucose_last2Week['time'] < current_end_date)]
215             fig, ax = plt.subplots(figsize=(10,5))
216             ax.plot(dfGlucose_current['time'], dfGlucose_current['value'],
label='Glukose mg/dL', color='#68B0AB')
217             fig.tight_layout()
218             # healthy values from 70-180
219             plt.axhspan(70, 180, facecolor='#D3D3D3', alpha=0.5)
220
221             #plot alarms
222             for i in range(len(self.glucose_alarm_start)):
223                 df_alarm = dfGlucose_current[(dfGlucose_current['time'] >=
self.glucose_alarm_start[i]) & (dfGlucose_current['time'] <= self.
glucose_alarm_end[i])]
224                 ax.plot(df_alarm['time'], df_alarm['value'], color='red')
225
226             # format x-axis
227             days = mdates.DayLocator(interval = 1)
228             dayFmt = mdates.DateFormatter('%d.%m')
229             ax.xaxis.set_major_locator(days)
230             ax.xaxis.set_major_formatter(dayFmt)
231             # format y-axis
232             ax.yaxis.grid(True)

```

```

plt.yticks(fontsize=12)
234 # save and show plot
date_end = current_end_date.strftime("%Y-%m-%d")
236 date_start = current_start_date.strftime("%Y-%m-%d")
plt.savefig(f"glucose_{date_end}_{date_start}.png")
238 os.system(f"glucose_{date_end}_{date_start}.png")
current_start_date = current_end_date

240

#plot day data
242 for i in range(0,5,1):
    current_start_date = current_end_date - datetime.timedelta(days
=1)
244    dfGlucose_current = self.dfGlucose_last2Week[(self.
dfGlucose_last2Week['time'] >= current_start_date) & (self.
dfGlucose_last2Week['time'] < current_end_date)]
    fig, ax = plt.subplots(figsize=(10,5))
246    ax.plot(dfGlucose_current['time'], dfGlucose_current['value'],
label='Glukose mg/dL', color='#68B0AB')
    fig.tight_layout()
248    # healthy values from 70-180
    plt.axhspan(70, 180, facecolor='#D3D3D3', alpha=0.5)

250

#plot alarms
252 for i in range(len(self.glucose_alarm_start)):
    df_alarm = dfGlucose_current[(dfGlucose_current['time'] >=
self.glucose_alarm_start[i]) & (dfGlucose_current['time'] <= self.
glucose_alarm_end[i])]
254    ax.plot(df_alarm['time'], df_alarm['value'], color='red')

256

# format x-axis
hours = mdates.HourLocator(interval = 2)
258 hourFmt = mdates.DateFormatter('%H:%M')
ax.xaxis.set_major_locator(hours)
260 ax.xaxis.set_major_formatter(hourFmt)
ax.xaxis.grid(False) # disable vertical grid
262 # format y-axis
ax.yaxis.grid(True)
264 plt.yticks(fontsize=12)

266

# add markers for fInsulin values
#dfGlucose_current['fInsulin'] = pd.to_numeric(
dfGlucose_current['fInsulin'], errors='coerce')
268 for index, row in dfGlucose_current.iterrows():
    if not pd.isna(row['fInsulin']):

```



```

270         ax.scatter(row['time'], row['value'], marker='o', color
= '#D76E72', s=200, zorder=3)
        ax.text(row['time'], row['value']-14, f"{row['fInsulin
']}IE", color='#D76E72', fontsize=14)
272         if not pd.isna(row['sInsulin']):
            ax.scatter(row['time'], row['value'], marker='o', color
= '#FFAD4C', s=200, zorder=3)
274             ax.text(row['time'], row['value']+7, f"{row['sInsulin
']}IE", color='#FFAD4C', fontsize=14)

276         # save and show plot
        date_end = current_end_date.strftime("%Y-%m-%d")
278         date_start = current_start_date.strftime("%Y-%m-%d")
        plt.savefig(f"glucose_{date_end}--{date_start}.png")
280         os.system(f"glucose_{date_end}--{date_start}.png")
        current_end_date = current_start_date
282     return

284 class ECG:
    def __init__(self, auth2_client, now):
286         self.auth2_client = auth2_client
        self.now = now
288
    def getData(self):
290         #set params for ECG request limit = number of entries returned max.
        10
        params = f'?beforeDate={self.now}&sort=desc&limit=4&offset=0'
292         ecgData = self.auth2_client.get_ecg(opts = params)
        #save ecg data in a json file
294         with open("ecgData.json", "w") as f:
            json.dump(ecgData, f)
296         return

298     def detectDeviations(self):
        return
300
    def plotData(self):
302         with open("ecgData.json", "r") as f:
            ecgData = json.load(f)
304         for i, ecgReading in enumerate(ecgData['ecgReadings']):

306             yAxis = ecgReading['waveformSamples']
            xAxis = np.linspace(0, 15, int(len(yAxis)/2))
308

```

```

310     #arange ticks for grid
311     major_ticks = np.arange(0, 15, 2)
312     minor_ticks = np.arange(0, 15, 0.1)
313     minor_yticks = np.arange(-len(yAxis), len(yAxis), int(len(yAxis)
)/11))
314     major_yticks = np.arange(0, 0, 1)
315     #set figsize
316     fig, axs = plt.subplots(figsize=(50, 6))
317     #plot
318     axs.plot(xAxis, yAxis[int(len(yAxis)/2):])
319     axs.set_xlabel('Zeit in s', fontsize=20)
320     #axs.set_title(ecgData['ecgReadings'][0]['startTime'], fontsize
=16)
321
322     #set ticks for grid
323     axs.set_yticks(minor_yticks, minor = True, major = False)
324     axs.set_yticks(major_yticks, major = False)
325     axs.set_xticks(major_ticks)
326     axs.set_xticks(minor_ticks, minor = True)
327     axs.set_yticklabels([])
328     axs.tick_params(axis='x', labelsize=20)
329     axs.grid(which = 'both')
330
331     # extra lines on each major tick
332     for x in major_ticks:
333         axs.axvline(x=x, color='gray', linestyle='—', linewidth=2)
334     # extra lines between the major ticks (on missing seconds)
335     for x in major_ticks[:-1]:
336         axs.axvline(x=x + 1, color='gray', linestyle='—',
linewidth=2)
337
338     #print additional information
339     print(f"Ergebnis: {ecgReading['resultClassification']}")
340     print(f"Herzfrequenz: {ecgReading['averageHeartRate']}")
341     print(f"Start Time: {ecgReading['startTime']}")
342
343     #manipulate start time to use it as a filename / substitute
invalid characters
344     start_time_str = ecgReading['startTime']
345     start_time_dt = datetime.datetime.fromisoformat(start_time_str
[: -4])
346     filename = start_time_dt.strftime('%Y-%m-%d_%H-%M-%S')
347     plt.savefig(f"{filename}.png")
348     os.system(f"{filename}.png")
349
350     return

```

```

348 class SpO2:
350     def __init__(self, auth2_client, thisWeek):
352         self.auth2_client = auth2_client
353         self.thisWeek = thisWeek
354
355     def getData(self):
356         dateTime = []
357         spo2 = []
358
359         #get spo2 Data of last Week
360         for oneDate in self.thisWeek:
361             #convert date into right time format
362             Date = oneDate.date().strftime("%Y-%m-%d")
363             oneDayData = self.auth2_client.spo2(date = Date) # request
364             if not oneDayData:
365                 print(f"No SpO2 data available for {Date}, skipping.") #
366             if no Data for the specific Day available
367             else: # save Data and spo2 Data
368                 dateTime.append(oneDate.date().strftime("%d.%m."))
369                 spo2.append(oneDayData['value']['avg'])
370
371         spo2Data = {"dateTime": dateTime, "spo2": spo2}
372         with open("spo2.json", "w") as f:
373             json.dump(spo2Data, f)
374         return
375
376     def detectDeviations(self):
377         deviationTime = []
378         deviationValue = []
379         #open spo2 json file
380         with open("spo2.json", "r") as f:
381             spo2Data = json.load(f)
382         # detect values under 93%
383         for i, value in enumerate(spo2Data['spo2']):
384             if value < 93:
385                 deviationTime.append(spo2Data['dateTime'][i])
386                 deviationValue.append(spo2Data['spo2'][i])
387         #add deviations to spo2Data
388         spo2Data['deviationTime'] = deviationTime
389         spo2Data['deviationValue'] = deviationValue
390
391         with open("spo2_thisweek.json", "w") as f:
392             json.dump(spo2Data, f)

```

```

    return
392
def plotData(self):
394     with open("spo2_thisweek.json", "r") as f:
        spo2Data = json.load(f)
396
        if spo2Data['spo2']:
398             # Set the plot style to "ggplot"
                plt.style.use('default')
400             fig, ax = plt.subplots(figsize=(10, 3)) # Adjust figure size

402             plt.plot(spo2Data['dateTime'], spo2Data['spo2'], '-', marker='o',
                markersize=6, linewidth=2, color='#68B0AB')
                if 'deviationTime' in spo2Data:
404                     plt.plot(spo2Data['deviationTime'], spo2Data['deviationValue'], 'o', markersize=10, color='red')

406                     # colorize the norm area
                        min_spo2 = min(spo2Data['spo2'][:6])
408                         max_spo2 = max(spo2Data['spo2'][:6])
                            plt.axhspan(min_spo2, max_spo2, facecolor='#D3D3D3', alpha=0.5)

410
                            plt.ylim(92, 100) # Begrenzung der y-Achse auf 92-100%
412                             plt.grid(axis='y', linestyle='--', linewidth=0.5, alpha=0.2)
                                plt.yticks(np.arange(92, 100, 2), fontsize=12)
414                                 plt.tight_layout()

416                                 #print additional information
                                    print(f"Weche von {spo2Data['dateTime'][0]} bis {spo2Data['dateTime'][-1]}")
418                                     print(f"Min: {min_spo2}")
                                        print(f"max: {max_spo2}")

420
                                        for i in range(len(spo2Data['dateTime'])):
422                                            x = spo2Data['dateTime'][i]
                                                y = spo2Data['spo2'][i]
424                                                    rounded_y = round(y)
                                                        plt.text(x, y+0.5, str(rounded_y), fontsize=8, ha='center',
426                                                            va='bottom')

428
                                                        plt.savefig('spo2LastWeek.png')
                                                            os.system('start spo2LastWeek.png')
else:
430     print('No SpO2 Data available!')

```

```

432         return
433
434 class Profile:
435     def __init__(self, auth2_client):
436         self.auth2_client = auth2_client
437
438     def getData(self):
439         profile = self.auth2_client.get_profile()
440         fullName = profile['user']['fullName']
441         age = profile['user']['age']
442         height = round(profile['user']['height'] * 2.54, 2) # convert
443         height from inches to centimeters and round to 2 decimal
444         weight = round(profile['user']['weight'] / 2.20462, 2) # convert
445         weight from pounds to kilograms and round to 2 decimal
446         allergies = profile['user']['aboutMe'].split('\r\n')[0].split(': ')
447         [1] # extract allergies from the aboutMe string
448         diseases = profile['user']['aboutMe'].split('\r\n')[1].split(': ')
449         [1] # extract diseases from the aboutMe string
450         emergencyContact = profile['user']['aboutMe'].split('\r\n')[2].
451         split(': ')[1] # extract emergency contact from the aboutMe string
452         #save Data
453         profileData = {"Name": fullName, "Alter": age, "Groesse": height, "
454         Gewicht": weight, "Allergien": allergies, "Erkrankungen": diseases, "
455         Notfall Kontakt": emergencyContact}
456         with open("profileData.json", "w") as f:
457             json.dump(profileData, f)
458         return
459
460     def printData(self):
461         with open("profileData.json", "r") as f:
462             profileData = json.load(f)
463
464         # print the information
465         print(f"Name: {profileData['Name']}")
466         print(f"Alter: {profileData['Alter']}")
467         print(f"Groesse: {profileData['Groesse']} cm")
468         print(f"Gewicht: {profileData['Gewicht']} kg")
469         print(f"Allergien: {profileData['Allergien']}")
470         print(f"Erkrankungen: {profileData['Erkrankungen']}")
471         print(f"Notfall Kontakt: {profileData['Notfall Kontakt']}")
472         return
473
474 if __name__ == '__main__':
475     CLIENT_ID = '2397GL'

```

```

468 CLIENT_SECRET = 'a74ce071aeceaddb7a23a44d2421720a'
470 #authorize
    authorize = Authorize(CLIENT_ID, CLIENT_SECRET)
472 auth2_client = authorize.authorize()

474 #get time stamps
    time = Time()
476 thisWeek, now = time.getTime()

478 #instantiate classes
    glucose = Glucose()
480 heartrate = Heartrate(auth2_client, thisWeek)
    ecg = ECG(auth2_client, now)
482 spo2 = SpO2(auth2_client, thisWeek)
    profile = Profile(auth2_client)
484

486 #get and save data
    glucose.getData()
    heartrate.getData()
488 ecg.getData()
    spo2.getData()
490 profile.getData()

492 #detect and save deviations
    glucose.detectDeviations()
494 heartrate.detectDeviations()
    ecg.detectDeviations()
496 spo2.detectDeviations()

498 #plot data
    glucose.plotData()
500 heartrate.plotData()
    ecg.plotData()
502 spo2.plotData()
    profile.printData()

```

Listing A.1: Programm zum Abrufen Verarbeiten und Darstellen von erfassten Daten eines Fitbit Charge 5

A.4.2 Ergänzungen in python-fitbit Bibliothek

```
# changes in api.py
2 # line 115 in authorize_token_url(), specified scopes get unlocked during
  authorization
self.session.scope = scope or [
4     "activity",
  "nutrition",
6     "heartrate",
  "location",
8     "nutrition",
  "profile",
10    "settings",
  "sleep",
12    "social",
  "weight",
14    "oxygen_saturation", # added
  "electrocardiogram", # added
16 ]
# line 199 in class Fitbit, added ressources can be used by the default
  request function
18 RESOURCE_LIST = [
  'body',
20  'activities',
  'foods/log',
22  'foods/log/water',
  'sleep',
24  'heart',
  'bp',
26  'glucose',
  'spo2', # added
28 ]
# line 1041, ecg request url differs from default request url
30 def get_ecg(self, user_id=None, opts=None, **kwargs):
32     base_url = "{0}/{1}/user/{2}/ecg/list.json"
  url = base_url.format(*self._get_common_args(user_id))
34     return self.make_request(url, opts=opts, **kwargs)
```

Listing A.2: Ergänzungen in der python-fitbit Bibliothek zum abrufen der SpO2 und EKG Daten

Erklärung zur selbstständigen Bearbeitung einer Abschlussarbeit

Gemäß der Allgemeinen Prüfungs- und Studienordnung ist zusammen mit der Abschlussarbeit eine schriftliche Erklärung abzugeben, in der der Studierende bestätigt, dass die Abschlussarbeit „— bei einer Gruppenarbeit die entsprechend gekennzeichneten Teile der Arbeit [(§ 18 Abs. 1 APSO-TI-BM bzw. § 21 Abs. 1 APSO-INGI)] — ohne fremde Hilfe selbständig verfasst und nur die angegebenen Quellen und Hilfsmittel benutzt wurden. Wörtlich oder dem Sinn nach aus anderen Werken entnommene Stellen sind unter Angabe der Quellen kenntlich zu machen.“

Quelle: § 16 Abs. 5 APSO-TI-BM bzw. § 15 Abs. 6 APSO-INGI

Dieses Blatt, mit der folgenden Erklärung, ist nach Fertigstellung der Abschlussarbeit durch den Studierenden auszufüllen und jeweils mit Originalunterschrift als letztes Blatt in das Prüfungsexemplar der Abschlussarbeit einzubinden.

Eine unrichtig abgegebene Erklärung kann -auch nachträglich- zur Ungültigkeit des Studienabschlusses führen.

Erklärung zur selbstständigen Bearbeitung der Arbeit

Hiermit versichere ich,

Name: _____

Vorname: _____

dass ich die vorliegende Bachelorarbeit – bzw. bei einer Gruppenarbeit die entsprechend gekennzeichneten Teile der Arbeit – mit dem Thema:

Konzeption und Evaluation eines Systems zur Nutzung eines Smart Wearables für die Unterstützung des Regelrettungsdiensts

ohne fremde Hilfe selbständig verfasst und nur die angegebenen Quellen und Hilfsmittel benutzt habe. Wörtlich oder dem Sinn nach aus anderen Werken entnommene Stellen sind unter Angabe der Quellen kenntlich gemacht.

- die folgende Aussage ist bei Gruppenarbeiten auszufüllen und entfällt bei Einzelarbeiten -

Die Kennzeichnung der von mir erstellten und verantworteten Teile der Bachelorarbeit ist erfolgt durch:

Ort

Datum

Unterschrift im Original