Faculty of Science and Technology
Department of Computer Science

From Palm to Arm

The system design of a standalone smartwatch application for diabetes management.

Bernt Julian Klevstad

Master's thesis in Computer Science INF-3981, June 2022

Abstract

The number of people diagnosed with diabetes is increasing at an alarming rate. However, strong evidence shows that health information technology has improved medical outcomes, especially within the field of diabetes research.

This thesis investigates how to motivate people with diabetes to perform self-management activities with the help of a smartwatch application. The work is grounded in a literature review, discovering how people manage diabetes with smartwatches today and the lack of existing motivational features on existing solutions. As a result, a system design of a smartwatch application is presented, including a graphical user interface (UI). The system aims to manage and monitor the essential diabetes metrics: nutrition, blood glucose, and physical activity while generating motivation through goal setting. In addition, the presented system is based on a standalone architecture, removing the need to pair a smartphone to the smartwatch and introducing novel features for smartwatch diabetes management. Finally, a proof of concept is implemented using Android studio to solidify the systems requirements.

Furthermore, a descriptive analysis of a survey presents that among people with diabetes, simplicity is the most crucial factor in smartwatch applications. Based on this, the presented UI is evaluated according to the simplicity of other systems and the impact the motivational features have on the system's complexity. Then, the potential of a standalone architecture for diabetes management is discussed. Finally, it is concluded that goal-setting features should be more widely used among smartwatch applications due to their low impact on the application.

The future work of the thesis is to test the system on people with diabetes. Both to evaluate the system useability scale and observe the impact goal-setting has on performing diabetes self-management. Furthermore, in this thesis, it is assumed that there is a communication channel between diabetes devices and the smartwatch. This must be further investigated with the next generation of diabetes devices.

Acknowledgments

First and foremost, a special thanks to my supervisor André Henriksen. Thank you for guiding me throughout this project, answering my questions, and keeping me on the right track. Thanks to Eirik Årsand for the expert insight and Gunnar Hartvigsen for the wise knowledge.

Thanks to my fellow students, especially Elisabeth, and A136, who has always believed in me and stayed with me throughout all of these years.

On a final note, I would like to give a special honor to the student counselor at the faculty, Jan Fuglesteg. Thank you.

Bernt Julian Klevstad, June 2022.

Table of content

List of Tables

List of Code Listings

List of Abbreviations

API – Application Programming Interface

BLE – Bluetooth Low Energy

CGM – Continuous glucose monitor

DIY – Do it yourself

HbA1c – Glycated Hemoglobin

HTTPS – Hypertext Transfer Protocol Secure

MVVM – Model-View-ViewModel

NFC – Near Field Communication

ROAMM – Real-Time Online Assessment and Mobility Monitor

TIR – Time In Range

UI – User interface

WHO – World Health Organization

List of Figures

1 Introduction

1.1 Background

Diabetes is a chronic metabolic disease characterized by increased blood glucose levels [1]. There are two main types of diabetes, type 1 diabetes, where the body produces little to no insulin, and type 2 diabetes, where the body cannot efficiently utilize the produced insulin. Furthermore, insulin is a hormone that breaks down and regulates blood sugar, and for people with type 1 diabetes it is vital to have daily injections of it.

Diabetes is one of the leading causes of death globally [1], and the prevalence is continuously increasing. The number of diagnoses has increased by over 400 million over the past 40 years, affecting 537 million people in 2021 [2].

People living with diabetes must follow a strict self-management regime to reduce the disease's impact on one's health and wellbeing [3]. Such a regime includes intervention and decision-making regarding medication use, blood glucose monitoring, dietary intake, and physical activity. These activities directly or indirectly affect the blood glucose levels, which are monitored by a blood glucose monitor. The monitors are the cornerstone of diabetes management [4] and give people with diabetes indirect feedback on what self-management activities affect their blood glucose. Furthermore, studies show that following all recommended diabetes management activities can take up to two hours each day [5]. Therefore, addressing the motivational aspect of accomplishing the self-management activities is essential.

Technology regarding diabetes treatment has advanced a lot in the past decades. Since the discovery of insulin in 1921 [6], the diabetes industry has introduced insulin syringes, insulin pens, handheld glucose monitors, continuous glucose monitors, insulin pumps, and closed-loop systems. However, it is not only enterprise systems pushing the development of diabetes treatment. The do it yourself (DIY) community has greatly impacted the technology industry [7], such as the “WeAreNotWaiting” [8] movement. This movement started as a hashtag on Twitter, where people expressed their frustration; #WeAreNotWaiting for competitors to cooperate or #WeAreNotWaiting to make it easier to get data off of devices. Over time, this led to a large community that today encompasses large DIY projects such as NightScout [9] and The Open Artificial Pancreas System project [10].

Smartwatch technology is a field with great potential to enhance and ease the daily self-management process for diabetic patients [11]. Smartwatches give a unique user experience with higher availability than smartphones with the same set of sensors. Furthermore, the prevalence of smartwatch owners has dramatically increased over the years. It is forecasted to increase from 593 million sold devices in 2018 to 929 million devices by 2021 [12]. However, despite the amount of sold units, the developed market of smartwatch applications is unsatisfactory [13].

Today, the main advisement for smartwatch development is to configure the application as standalone [14]. When a smartwatch is configured as standalone, it indicates that the application is functional without the need for a paired smartphone. The result of a standalone smartwatch is that it is more available and does not depend on the smartphone.

1.2 Scope and research problem

This project aims to explore the potential of a standalone smartwatch application, where the goal of the application is to motivate and assist people with diabetes through the self-management process. By using a standalone smartwatch application, the relevant information is made more easily available than using a smartphone to access the same information. The main research problem is:

RP: How can a standalone smartwatch application motivate users to perform self-management activities regarding diabetes?

The main research problem can be broken down into multiple research problems to facilitate the main problem. Firstly, understanding what people with diabetes find essential and viable in a smartwatch application is vital. The screen and capabilities of smartwatches are limited, and deciding where the focus in development should reside is essential. The first research problem addresses the needs of the users:

RP1: What elements do persons with diabetes find essential in smartwatch applications?

According to the main research problem, the application must motivate the user. Therefore, the second research problem addresses the challenge related to user motivation, and aims to explore what motivational techniques that can work in a smartwatch environment:

> *RP2: What motivational techniques can be implemented in a standalone smartwatch application to encourage self-management of diabetes?*

One of the features of having a standalone smartwatch application is the direct communication with external services without having a paired smartphone. This architecture enables features where people without smartphones, such as children or the elderly, can generate and make data accessible to others directly. The last research problem addresses the aspects of standalone functionalities within a diabetes application:

> RP3: *How can we best utilize the properties of standalone smartwatches when designing a diabetes application?*

1.3 Assumptions and limitations

This thesis will assume that continuous glucose monitor (CGM) data is available over the Bluetooth Low Energy (BLE) protocol, and available for smartwatches to pair with this connection. With the current technology, this is not possible, however, both rumors and Dexcom presents direct smartwatch communication in the next generation of CGMs.

Moreover, the thesis will address the backend solution for completeness. However, the main focus will be the smartwatch application and its design. Throughout the implementation chapter will some of the presented features be implemented. The main reason that the implementation of the backend service is not addressed is the time restrictions of the thesis. In addition, implementing an external backed service to collect, store and share health data is a demanding task, especially with privacy and security in mind. Furthermore, the assumption of collecting CGM data over BLE limits the implementation due to the inherited nature of not having such hardware yet. However, the implementation will address what and where the HTTPS and BLE communication would reside within the smartwatch application.

The implemented system is tailored for a Samsung Galaxy Watch 4[1]. From a capstone project by Klevstad [15], it was evaluated to use this device on the condition of available programming interfaces, prior experiences with the programming environment, and the uniform operating system Wear OS[2]. Due to this limitation, Apple Inc. products will not be in-depth presented in this thesis.

1.4 Structure of the thesis

Chapter 1 – Introduction – Presents the motivation, background, and the research problems of the thesis.

Chapter 2 – Theoretical framework – Presents the needed theoretical concepts introduced throughout the thesis. The concepts are split into diabetes, motivation, and smartwatch theory.

Chapter 3 – Methods – Outlies the approach and process of this thesis. Presented elements are the literature review, the survey analysis, the research paradigm, the implementation process, and the tools used.

Chapter 4 – Related work – Presents the work done within this field. Both state of the art and literature review is presented here.

Chapter 5 – Requirement specification – Presents the requirements of the presented system.

Chapter 6 – Design – Defines the system and its needed elements. Additionally, the smartwatch's visual design.

Chapter 7 – Implementation – Describes the implementation of the proof of concept.

Chapter 8 – Results – Presets the results from the survey “motivation in mobile Health”.

Chapter 9 – Discussion – In this chapter, the survey is discussed, followed by a brief discussion of state of the art. Furthermore, the presented system is evaluated and put in perspective of the survey and the related work. Then a discussion of the future of the standalone architecture and diabetes management is held. In this chapter, the research questions are answered.

Chapter 10 – Conclusion – Summarizes the work and presents concluding remarks.

---

[1] https://www.samsung.com/no/watches/galaxy-watch/galaxy-watch4-classic-black-lte-sm-r885fzkaeud/

[2] https://developer.android.com/wear

2 Theoretical framework

2.1 Diabetes

Diabetes is a chronic metabolic disease where the body cannot efficiently utilize the produced insulin or produce any insulin. Insulin is a hormone that breaks down and regulates blood sugar [1]. In the case of inefficient usage or low production, the need for preventive intervention depends on the type and severity of the disease. However, inadequate management of the disease can lead to long-term illness or, in worst cases, death [16]. There are several types of diabetes, where type 1 and type 2 diabetes are the most common [1]. The World Health Organization (WHO) has categorized diabetes as one of the leading non-communicable diseases [17].

People with diabetes must oversee their blood glucose. Diabetes can cause lethal damage to the body if it gets too high or too low. Therefore, ensuring that the blood glucose is within ideal levels; for people with diabetes, between 4 and 7 mmol/L [18], is essential for a healthy life with the disease.

2.1.1 Type 1 Diabetes

People diagnosed with type 1 diabetes have reduced or no insulin production [1]. The cause of the low insulin production is due to the immune system attacking the insulin-producing cells in the pancreas. As a result, people with type 1 diabetes are life dependent on daily insulin injections, hence type 1 diabetes is also known as insulin-dependent diabetes. In Norway, diagnosis is concluded over a blood sample [19].

The symptoms of type 1 diabetes are frequent need for urination, thirst, constant hunger, weight loss, vision changes, and fatigue [1].

2.1.2 Type 2 Diabetes

Type 2 diabetes is the most common type of diabetes and concerns 95% of the total cases of diabetes [1]. As a result of poor utilization of the body's naturally made insulin (insulin resistance), the person develops diabetes. This type is different from type 1 diabetes since the body still has insulin production. The cause of developing such inefficiency is primarily due to excess body weight and physical inactivity.

Furthermore, the main symptoms of type 2 diabetes are similar to type 1, however, they are not as apparent. As a result, a diagnosis of type 2 diabetes can transpire long after the disease has set roots [1]. One of the significant concerns of late diagnosis of diabetes is prolonged and frequent occurrences of hyperglycemia (i.e., high blood glucose).

Type 2 diabetes is diagnosed by testing the amount of glycated hemoglobin (HbA1c) in the blood. In Norway, diabetes is diagnosed if the HbA1c is over 48 mmol/mol (6.5%) in two separate tests [20].

2.1.3 Monitoring

Monitoring blood glucose is an essential tool to mitigate the impact diabetes has on one's life [3]. In order to monitor blood glucose, people can use blood glucose readers. A blood glucose reader is a digital device that punctures the skin, collects a drop of blood, and reads the blood's glucose level. These devices are beneficial since blood glucose naturally fluctuates throughout the day, and therefore, can be challenging to predict. Factors such as the time since the previous meal, carbohydrates in that meal, or the type of medication taken can also affect the blood glucose [21]. Hence, blood glucose readers are the cornerstone of diabetes self-management.

It is common to perform multiple recordings to understand if the glucose levels are rising, falling, or stabilizing [21]. Furthermore, discovering hyper- or hypoglycemia is done with a glucose monitor [22]. Another option to get accurate results is to monitor blood glucose after overnight fasting.

Continuous glucose monitor

Glucose monitors are essential in order to achieve glycemic targets and understand the impact of medication and actions [3]. People who need insulin supplements can install a continuous glucose monitor (CGM) with a prescription from the doctor. Continuous glucose monitors are formed as a large band-aid and perform the same function as a standard glucose monitor. In contrast to the standard blood glucose monitor, the CGM is installed on the body and passively collects glucose recordings, reducing active self-management. These devices have proven to help people come closer to their HbA1c goal [23].

Dexcom[3] is one of the leading CGM producers. Their current CGM flagship, Dexcom G6, is a two-part transmitter and sensor. The CGM consists of a transmitter that reads data from the sensor and sends it to the paired smartphone over Bluetooth. However, in the next generation of CGM (Dexcom G7), it will also be possible to send data directly to the Apple smartwatch[4].

FreeStyle Libre 2[5] is another brand of CGM. This CGM is designed around data transmission to the smartphone over NFC. Unfortunately, near-field communication is not as accessible as Bluetooth since the smartphone must be within 4 cm. However, third-party devices can read the data and send it over Bluetooth, such as MiaoMiao[6]. MiaoMiao is a strap the person can use to connect to the FreeStyle and transmit glucose data over Bluetooth. Rumors state that Bluetooth adapters will be integrated into the CGM In the next generation of FreeStyle Libre[7].

2.1.4 Hyperglycemia

Hyperglycemia is the medical term for too high blood glucose [24], specifically if the blood glucose measures over 10 mmol/L two hours after eating [25]. People with type 1 diabetes can develop hyperglycemia by medicating insufficient amounts of insulin, whereas people with type 2 diabetes can develop it by having highly ineffective insulin. Activities that elevate blood glucose can also lead to hyperglycemia, such as an unplanned amount of physical activity, binge eating, or stress compared to the planned medication. Other symptoms of hyperglycemia are high levels of sugar in the urine, frequent urination, and increased thirst.

Over time hyperglycemia can cause severe damage to the nerves and blood vessels, and if not treated, cause a diabetic coma (ketoacidosis) [24]. Diabetic coma occurs when the body has insufficient amounts of blood glucose and instead starts burning fat as a replacement. Burning fat produces ketones as a bi-product, and this bi-product is usually extracted with the urine.

---

[3] https://www.dexcom.com/

[4] https://investors.dexcom.com/news-releases/news-release-details/dexcom-g7-receives-ce-mark-next-generation-continuous-glucose

[5] https://www.freestyle.abbott/no-no/freestyle-libre-system/freestyle-libre-2.html

[6] https://miaomiao.cool/

[7] https://diatribe.org/freestyle-libre-3-cleared-europe-%E2%80%93-smaller-thinner-and-no-more-scanning

However, this is not done effectively due to the low blood glucose and, thus, starts to build up in the blood. Diabetic coma is life-threatening and needs acute medical attention.

2.1.5 Hypoglycemia

Hyperglycemia is the medical term for too low blood glucose [22]. Hyperglycemia typically occurs when glucose levels fall below 3.9 mmol/L. However, due to variations between people, hypoglycemia can be identified as the point where the blood glucose does not naturally stabilize. The most prominent cause of hypoglycemia is either a too large insulin injection or an inadequate diet in relation to insulin medication [16]. For example, hypoglycemia can be caused by not being familiar with a new medication, or following standard medication but not eating a meal with the usual amount of carbohydrates.

Hypoglycemia is common among people with diabetes who use insulin supplements, and for type 1 diabetes patients, it can occur as often as twice a week [22]. There are many symptoms of hypoglycemia, such as feeling shaky, anxiety, sweating, confusion, or hunger.

Hypoglycemia is often treated according to the 15-15 rule [22]. When a person recognizes symptoms or discovers too low blood glucose, they ingest 15 grams of carbohydrates and monitor the blood glucose after 15 minutes. If it is still below, the procedure is repeated until the blood glucose values are over 70 mg/dL. Following the 15-15 rule reduces the chance of overshooting the needed carbohydrates, avoiding hyperglycemia.

2.1.6 Glycated Hemoglobin

Glycated Hemoglobin, also known as HbA1c or A1c, is a measurement that indicates glycated hemoglobin (red blood cells that are attached to blood glucose) levels in the blood [26]. HbA1c is measured in mmol/mol or percent, and reflects the average blood glucose over the past three months and is used to help doctors diagnose, adjust treatment and prescribe medication for diabetes [26]. In order to collect HbA1c data, medical personnel performs a simple blood sample. HbA1c for healthy people lies between 4-6%, whereas 6.5% and above indicates diabetes. However, to determine diabetes, two samples with at least two weeks in between are needed [26].

One challenge with HbA1c is the fact that the results are from an indication of the average value from the three past months [27]. Unfortunately, measurements based on average values tend to hide extremes (hypoglycemia and hyperglycemia). Therefore, in order to understand the complete picture of the disease, daily recordings are valued.

2.1.7 Time In Range

Time in range (TIR) is a measurement that indicates the percentage of time spent in ideal blood glucose levels (between 3.9 and 10 mmol/L, however, these values depend on the individual) [27]. For example, within the scope of a day and with a TIR value of 50%, will the TIR data indicate that the person is within the satisfied range half the time throughout a day. To follow TIR percentage, a CGM is needed because of the need for frequent blood glucose readings. By observing TIR, the person can be cautious of the extreme levels of glucose irregularities.

The main difference between TIR and HbA1c is that HbA1c indicates an average diabetic condition over three months. In comparison, TIR indicates a "real-time" state, including the outliers. For further explanation, **Figure 1** shows a reprinted image from diatribe [28], comparing TIR and HbA1c. The figure has three pie charts displaying glucose measurements over time. All of the charts have the same HbA1c of 7% and average blood glucose of 8.5 mmol/L. However, the three graphs also show varying degrees of TIR, hyperglycemia, and hypoglycemia. Looking at the first graph, we have two instances of hyperglycemia and one with hypoglycemia, which is not ideal. The third graph has no instances of too high or too low blood glucose. If we were to look at the HbA1c of these, we would not see a difference. However, the TIR shows a 40% compared to 100%.

According to the American Diabetes Association, research shows that staying within TIR will lower the likeliness of developing certain diabetes complications [27].

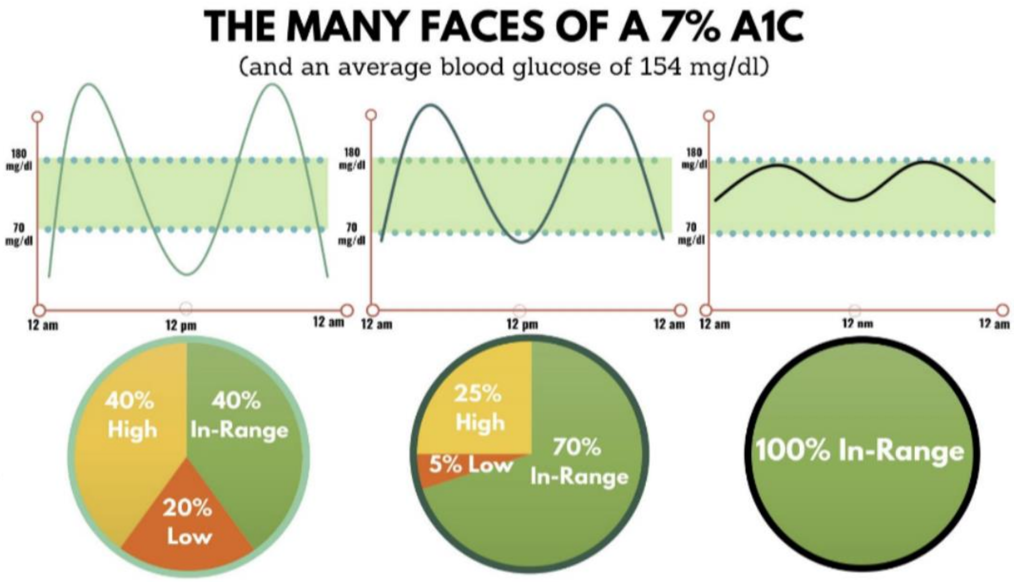

Figure 1: *Three different values of time in range, but with equal average blood glucose.*

2.2 The challenges of self-management

Living with diabetes can be challenging; the disease requires a strict ruleset with daily management activities. The Association of diabetes care & Education Specialists has defined seven key areas of care, and supplied guidelines for each area [29]. The areas are healthy coping, healthy eating, being active, taking medication, monitoring, problem-solving, and reducing risks. Following these guidelines has proven to positively affect glycemic control, reduce complications, and improve quality of life [3]. However, guidelines and education can be worthless without action from the individual diagnosed with diabetes.

Self-management activities are concrete behavioral tasks that a person performs to reduce the risk of aggravating the disease [3]. The extensiveness of such tasks depends on the person's medical condition, including physical state and genetics [3]. For example, when people are at risk of developing type 2 diabetes due to high blood glucose levels, they are diagnosed with prediabetes [30]. Prediabetes is serious and can lead to chronic illness. However, one of the main actions to prevent the development of type 2 diabetes is to reduce body weight [31].

The discovery of prediabetes is rare compared to the cases of type 2 diabetes. Upon the diagnosis of diabetes, people are expected to participate in positive lifestyle behaviors, such as following a dietary plan, performing at least 30 minutes of physical activity every day, taking medications, monitoring blood glucose levels, seeking medical care as needed, and following foot-care guidelines [3]. Daily adherence is expected, although these activities have varying importance and urgency. Unfortunately, these activities are time-consuming, Russel et al. [5] have estimated that experienced people with type 2 diabetes must spend two hours following all of the recommended self-management activities. However, Stafford et al. [32] presented a study showing that people (N=1482) use 48 minutes each day (median) on self-management activities.

2.2.1 Motivation

Diabetes requires daily attention, both in order to maintain the current condition of the disease and for future investment. However, understanding the effects and complications of diabetes is not enough to motivate people living with the disease. Knowing the risks of diabetes only plays a part in enhancing people's motivation.

To understand what motivates people with diabetes, one must understand what builds motivation in general. Motivation is a psychosocial construct and can be interpreted as *"... the degree of wanting, desiring, or intending to carry out actions in relation to specific goals"* [33]. In other words, motivation is the inner drive that helps people achieve their goals. This inner drive is affected by intrinsic or extrinsic factors [34].

Intrinsic motivation is the type of motivation where performing the activity gives a personal gain. For example, walking can relieve stress, and running might boost a person's confidence. In contrast, extrinsic motivation is when the activity gives a reward outside the personal level, for example, receiving a reward (e.g., money from the parents) for emptying the dishwasher or knowing friends have an expectation of me.

Adapting to a self-care regime often requires behavioral change due to the natural cause and treatment of the disease, for example, adhering to the daily activity minutes. Invoking behavioral change can undoubtedly be challenging. However, there are some methods to provoke motivation.

Self-determination theory

Self-determination theory is a common theory that focuses on the importance of autonomy and intrinsic motivation for producing a healthy adjustment [35]. Further, it states that behavioral change (adopting healthy behaviors or changing unhealthy ones) is more likely to happen if the psychological needs for *autonomy, competence, and relatedness* are satisfied [36]. Autonomy defines the need to behave willingly and be in control of their goals.

For example, the feeling of being autonomous due to taking medications on a personal importance promise and not because the doctor told them to take them. Competence elaborates on the fulfillment gained by knowing what it takes to complete and learn a skill. And lastly, relatedness is the need for social support. In addition, self-determination theory states that negative outcomes related to motivation can relate to people who only are driven by extrinsic forces and rewards [35].

Goal-setting theory

Goal-setting is one of the most evidence-based activities to promote behavioral change [37]. Indeed, National Standards for Diabetes Self-Management Education and Support have acknowledged SMART (specific, measurable, achievable, reasonable, and timely goals) goals as an evidence-based technique. Moreover, the concept of goals is mainly separated into goal specification and difficulty. Goal specification defines the scope of the goal, narrowing variation in behavior. For example, a goal without a specific target (e.g., "Lose as much weight as possible") will only rely on the self-effort, allowing a widespread execution of the goal. Whereas goals with high specifications (e.g., "Measure blood glucose four times each day") define an expected behavior outcome. Goal difficulty is measured in the probability of completing the goal and is related to performance in a positive linear relation [37]. A high-challenge goal will require more effort and result in a more significant behavioral change. There are many other factors affecting the process of goal setting. **Figure 2** presents a reprint from Miller et al. [37] that shows the relation between self-efficacy, the elements of goal setting, and outcome. We see that self-efficacy has an effect on the goal specification and the strategy of the goal. Then, the goal results in a behavior that gives a self reaction, which gives the motivation to continue the cycle.

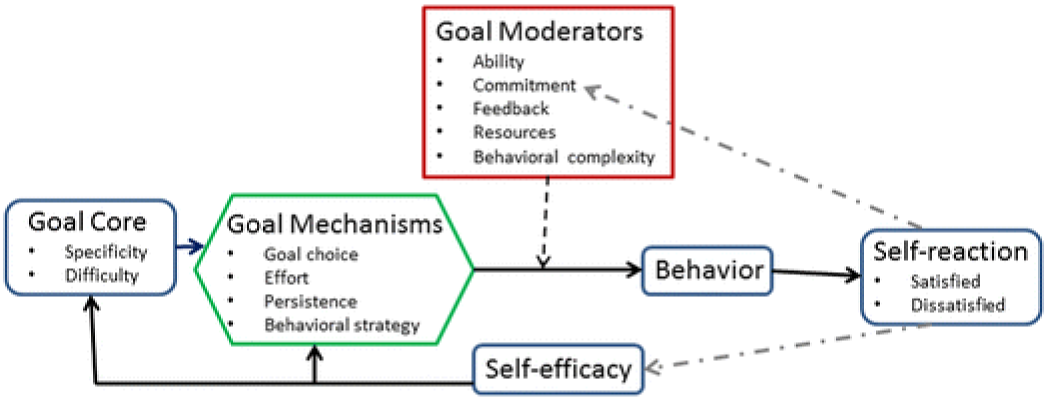

Figure 2: *The graph of goal theory. Dashed lines indicate feedback channels.*

2.3 Smartwatches

A smartwatch is a small wearable computer with the facade of a digital watch. Furthermore, modern smartwatches usually have a touch screen and are paired with the smartphone [38]. Unlike activity trackers, where the primary function is to collect activity data, smartwatches provide more general-purpose and complex functionality. Between smartwatches, smartphones, tablets, and laptops, smartwatches have the highest market growth [39].

Most commonly, smartwatches are equipped with an optical heart rate sensor, accelerometer, gyroscope, magnetometer, and GPS [38]. However, the higher-end models support 4G communication, direct Wi-Fi communication, and NFC payment.

2.3.1 Standalone

Most smartwatches are connected to a smartphone via Bluetooth, which allows for the processing and storing of data and external network communication to be offloaded to the smartphone. This has significant benefits as it reduces power usage and the hardware constraints on the smartwatch. However, as technology has advanced and cross-platform compatibility (Android phone and Apple watch, vice versa) has become more appreciated, it is now advised to implement smartwatch applications as standalone solutions [14]. If the application depends on a companion application on the smartphone, the application can be implemented as semi-standalone. Semi-standalone applications should serve basic functionalities without connecting to a paired smartphone/companion application. Lastly, annotating an application as standalone does not exclude external communication with a backend service available over the internet.

Standalone applications tend to have more prolonged user interactions. Hence, standalone applications should implement functionality usually handled by the smartphone, such as user preferences and login screens.

2.3.2 Layout

Interacting with a smartwatch application can be done through the following components: watchface, complications, notifications, application, and, on some devices, tiles.

Watchface

The watchface is the immediate "landing page" of the watch, this is the component that displays the time. This screen is often mimicked to look like a standard watch. In addition, the user can glance into limited application metrics on this screen, such as today's total steps, watch battery, and registered glasses of water consumed.

Complications

The metrics on the watchface are called complications. Complications are data points generated in an application and often displayed with an associated icon. This component gives the user quick and glanceable information, such as step data with a shoe icon. Furthermore, a watchface often supports a configurable amount of complications. Tapping on a complication often leads to the data source, for example, the shoe icon may lead to the fitness application of the watch

Notifications

Notifications are the elements of the applications that give notice of information. These are often used to deliver messages, reminders, or alarms. Furthermore, these notifications can be interacted with, such as quick responses or dismiss.

Tiles

Tiles are similar to watchfaces. However, tiles are placed next to the watchface, i.e., can be seen if the user swipes horizontally on the screen when unlocking the watch. Tiles benefit from the freedom of not showing the time, resulting in a canvas the developers can use for dedicated shortcuts or features accessible from the home screen.

Overlay

The overlay is a synonym for the application and is the application's core. The overlay binds together the watchface, notifications, tiles, and complications. Furthermore, the overlay can be opened from the app drawer on the watch or through the shortcuts the other components deliver. Within the overlay, the user configures the application and can access the main functionalities.

3 Methods

3.1 Literature review

Prior to this thesis, a capstone project [15] was done within the field of smartwatches. In that project, a literature review was performed. However, that review was unsatisfactory and narrow. Therefore, a less restrictive review is presented in this thesis.

The motivation behind this literature review was to collect information regarding the state of standalone smartwatch applications. The review was done using the following databases: PubMed[8], ACM[9], and IeeeXplore [10]. These databases were chosen due to the diversity of content they provide. Hence, PubMed provides literature dedicated to biotechnology, whereas ACM and IeeeXplore provide literature regarding computer science. This thesis is grounded in the three keywords: (1) diabetes, (2) smartwatch, and (3) standalone, and these keywords were, consequently, used in the search.

The search query in ACM and IeeeXplore was:
"diabetes" AND ("smartwatch OR "smart watch" OR "wearable") AND "standalone".

Due to the initial unsatisfactory results in Pubmed (only a single paper), it was decided to use a different search query compared to the one used in ACM and IeeeXplore, removing the keyword wearable and standalone. The search query was less specific and formulated as follows:
"diabetes" AND ("smartwatch OR "smart watch").

The process of eliminating results from the review was done according to PRISMA (Preferred Reporting Items for Systematic Reviews and Meta-Analyses) guidelines. **Figure 3** presents the PRISMA diagram. The first phase consisted of removing duplicate papers. Furthermore, papers were screened based on the title, abstract, and conclusion. The most common reason papers were rejected in this phase was because the papers were neither related to

[8] https://pubmed.ncbi.nlm.nih.gov/

[9] https://dl.acm.org/

[10] https://ieeexplore.ieee.org/Xplore/home.jsp

smartwatches nor diabetes. The second screening phase, eligibility, where a single paper excluded due to only being an abstract. The final result was ten papers.

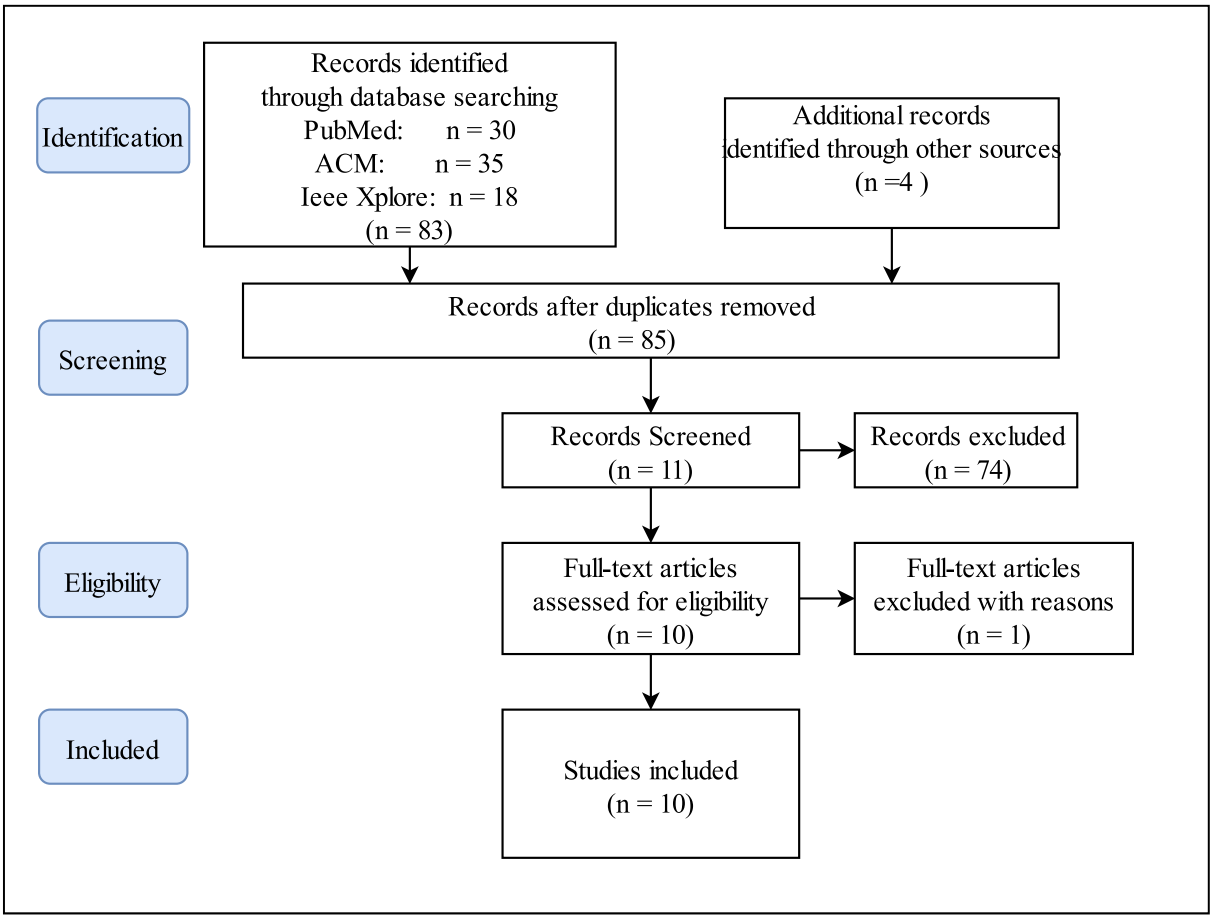

Figure 3: *The Prisma diagram.*

The review process is split into collecting diabetes management and smartwatch development information. In addition to papers and digital readings, smartwatch applications regarding diabetes and health management were queried in the Google Play Store[11]. This data collection was done between January and April of 2022.

3.2 Survey

Prior to this thesis, a survey regarding "Motivation in mobile health" was performed by Woldaregay et al. [40]. Although the author of this thesis did not conduct the survey, he has

[11] https://play.google.com/store/apps

gained access to the raw data. This survey will be descriptively analyzed and presented in the results.

Participants were primarily members of Norwegian and English diabetes groups and a Swiss cohort of healthy people. The data was collected between 11/2018 and 8/2019.

The survey was performed as a questionnaire had 38 questions and was split into the following seven categories: (1) background; (2) wearables and sensors; (3) mobile applications; (4) logging; (5) data sharing and data integration; (6) social media and entertainment factors; and (7) optional. However, not all of the questions and categories from the survey will be addressed here, as that will be outside the scope of this thesis. Instead, questions (**Table 1**) focused on wearables and features regarding health management were selected.

Furthermore, the raw data of the survey were collected from 814 participants. However, the results presented in this thesis will be a subset, specifically, those who answered: “Yes, Diabetes 1/2" to the question “Do you have a chronic disease?”. Accordingly, the presented results are from the N=71 participants who have diabetes. The results are presented in Section 8.1.

Table 1. *Ten selected survey questions from motivation in mobile health*

Nr.	Description	Options (abbreviated)
6	Which of these technologies for health tracking do you regularly use? (Multiple choice)	a) Sensors integrated in smartphone. b) Mobile health apps. c) Health specific measurement. d) Do not use sensors or wearables. e) Other.
7	How important are these features for you when choosing a wearable device? (Single, 1-4, + I don’t know)	a) Sensor accuracy and range b) Access to several types of data c) Notifications from mobile phone d) Quality of associated mobile app e) Known or specific brand/price f) Ergonomic and design g) Other features
8	Which features would motivate you most to use a wearable sensor longer?	a) Relevant personalized feedback b) Access to aggregated/summarized data based on the population

		c) Ease of use/non-disruptive d) Social integration (e.g. Facebook) e) Don’t know f) Other
9	How important are these specific health related features for you when choosing a wearable device? (Single, 1-4, + I don’t know)	a) Physical activity tracking (e.g. step counting) b) Manage disease (e.g. blood glucose) c) Predicting and preventing symptom and health deterioration d) Alerts when risky behavior/when approaching limits
10	How important are these features when choosing a mobile health app? (Single, 1-4, + I don’t know)	a) Simplicity/usability b) Functionality/features c) Price d) Trust/security/privacy e) Personalization (tailored features)
11	How do you decide if a mobile app is trustworthy? (Multiple choice)	a) Security, accuracy, and usefulness, etc. b) Product specification/provider reputation c) Personal experience or other people’s experiences d) Certificate by independent organization/expert judgment e) App rating in AppStore/GooglePlay f) Other
14	If you are required to do manual logging (registration) in a health mobile app, how important are these criteria for you?	a) Easiness and simplicity b) Time required c) Frequency of logging
15	How willing would you be to manually log or register the following types of data? (Single, 1-4, + I don’t know)	a) Medication intake b) Dietary intake c) Signs of infection/risks of infection d) Physiological indicators e) Daily mood/how you feel
28	Do you have a chronic disease? (Single)	a) Yes, Diabetes (Type 1) b) Yes, Diabetes (Type 2/Other) c) Yes, Sickle-cell disease (SS, S-Beta) d) Yes, Sickle-cell disease (SC/other) e) Yes, other chronic disease f) No g) Do not want to answer

29	Are you a caretaker of someone with a chronic disease? (single)	a) Yes, Diabetes (Type 1) b) Yes, Diabetes (Type 2/Other) c) Yes, Sickle-cell disease (SS, S-Beta) d) Yes, Sickle-cell disease (SC/other) e) Yes, other chronic disease f) No g) Do not want to answer

3.3 Research Paradigm

Denning et al. [41] have proposed a four-step guideline for approaching engineering within computer science. The guideline comes in three versions depending on the research problem: theory, abstraction, and design. Respectively, theory is rooted within mathematics, abstraction is rooted in the experimental scientific method, and design is rooted in engineering. The approach of this thesis is based on the latter, design.

The four-step approach for design is formulated:

1. State the requirements.
2. State the specifications.
3. Design and implement the system.
4. Test the system.

Furthermore, Denning et al. [41] state that these four steps should be reiterated as the system progresses. Hence, progress in software engineering is achieved when the system satisfies the specific requirement. A concluding remark is that they state that the approach paradigms are intended as a guideline and not an instruction.

The process of stating the requirements have been collected by analyzing the current field of smartwatches and the related work. Then, the requirements are specified in a requirement specification. Furthermore, based on these requirements, a design and implementation process started. Throughout the design process, feedback was gained from co-students and supervisors.

3.4 Volere Requirement Specification

Volere requirement specification Template [42] is a structural approach to defining requirements in software engineering. This specification is used in this thesis to map out the functionalities used in the design. Furthermore, the functionalities described in the specification are derived from the literature review, the survey, and the author's knowledge of software engineering. The requirement specification is a part of the research paradigm: stating the requirements.

3.5 Implementation

The implementation process of the work in this project started with a sketching program called Figma. Figma[12] is a program used by interaction designers for a fast and thorough documentation of web components.

The sketching was based on results from the review process and was mainly split into two parts; content and design. The content was based on what other management systems do and how people manage diabetes. Design techniques, such as pathing and styling, were based on Google's design guidelines and their public Figma template[13].

The implementation process consisted of continuous development of the application while adjusting according to feedback from fellow students. In addition, the library used for implementing the front-end of the smartwatch application is in the developer preview (alpha). Therefore, there were occasions when the author had to ask the developers of *Compose for Wear OS*[14] (UI library) for consultations due to incomplete documentation and frameworks to work with.

3.6 Tools

Developing the smartwatch application was done using Android Studio[15]. Android Studio is an integrated development environment made by Google (Google Inc., CA, US), which developers use to develop, maintain, run and compile code. Today Android Studio is mainly used to develop smartphone applications for Android phones. However, it can also build smartwatch applications.

[12] https://www.figma.com/

[13] https://developer.android.com/training/wearables/design/download

[14] https://developer.android.com/training/wearables/compose

[15] https://developer.android.com/studio

Kotlin was used as the programming language for the application[16]. Kotlin is a modern general-use programming language that is based on Java. Today Kotlin is the primary programming language for developing applications for the Android platform.

The application was tested on a Samsung Galaxy Watch 4 throughout the implementation. The application is built for this watch's operating system, Wear Os 3. Additionally, the software development kit was configured for version 31.

The presented icons in the design and implementation are retrieved under license from Freepik via Flaticon [43].

[16] https://kotlinlang.org/

4 Related work

4.1 Introduction

This chapter describes the current state of smartwatches regarding diabetes and health management. Firstly, relevant smartwatch applications are presented. Hence, exploring the available applications gives insight into the current field of smartwatch that is not documented in scientific databases. Then, the results from the literature review are presented. The presented ten papers are the outcome of the literature review.

4.2 State of the art

4.2.1 Diabetes:M

Diabetes:M [44] is the leading smartphone application for diabetes management in the Android store (March 2022), with a smartwatch integration. The smartphone application highlights the following: logging carbohydrates, glucose readings and medicine, bolus advisor, food database, detailed graph, remainders, importing data from external sources, and exporting collected data. Furthermore, all input methods that depend on a unit are configurable, e.g., medicine type or nutrition type (proteins, calories, carbohydrates, or fats). Some functionalities are locked behind a paywall subscription, such as Bluetooth integration with CGM's and having additional profiles on the platform.

The smartwatch application is dependent on the smartphone and is inoperative without installing the application on the smartphone first. Furthermore, the application connects to the smartphone application on launch over Bluetooth. The watchface is the smartwatch's core, displaying metrics synchronized from the phone. The watchface includes shortcuts to the logging features of food, medicine, and glucose readings. Lastly, the user can swap out the watchface with a TIR graph/watchface.

4.2.2 Diabetes Diary

Diabetes Diary [45] is a smartphone application and research project developed by the Norwegian center for e-health science. The goal of the application is to log enough data so that the user can gain more insight into the condition and receive more accurate treatment. The application supports manual input of international units, glucose recordings, food data (calories/carbohydrates or grams), minutes of physical activity, weight, and medicines.

Additionally, the user can set goals, such as performing more physical activity, weight target, or controlling carbohydrate intake. These goals progress as the user logs data.

Diabetes Diary has implemented a smartwatch integration for the smartphone application. While doing so, they extended the existing platform with a physical module collecting CGM data and transmitting it to the smartphone. The authors highlight the following functionalities: displaying glucose data, time since last injected insulin, carbohydrate logging, collection and manual logging of step data, and heartrate monitor. The smartwatch application depends on a Bluetooth connection between the smartwatch and smartphone. While Bluetooth is connected, the smartwatch synchronizes data to the smartphone and vice versa. **Figure 4** is a reprint from Pektas et al. [46], depicted as the logging view within the watchface.

Figure 4: *Diabetes diary's view for logging blood glucose.*

4.2.3 LifeSum

LifeSum is a digital diary that focuses on nutrition intake and physical activity[17]. The smartphone application leverages a well-designed food logger that aims to achieve the user's preferred initial goal; increase, decrease or keep a stable weight. The application's landing page summarizes how many calories the user has left for the day, based on the sum of the weight goal, logged nutrition, and today's physical activity. Nutrition is logged with manual input or scanning the product's QR-code. Physical activity is logged manually or synchronized from a third-party service, such as Samsung Health or Google Fit. The

[17] https://lifesum.com/

application also has a set of minor functionalities, such as following a diet and suggestions to become healthier.

LifeSum is also available on Android smartwatches. The smartwatch application leverages an overlay accessing authentication, two logging functionalities (calories and hydration), and a panel showing minutes of physical activity. The authentication process follows one of the following flows; signing in with Google or typing a one-time passcode retrieved from the LifeSum webpage. Signing in with Google gives a plug-and-play experience, whereas one-time passcode requires a second device/internet browser. When authenticated, one can access the rest of the functionalities. The logging collects and displays the number of glasses of water consumed and calories. The number of calories is only estimated based on meal type (breakfast, lunch, dinner, and snack) and size (small, medium, and large). The data in the application is synchronized to and from the LifeSum profile over Wi-Fi. **Figure 5** depicts screenshots from LifeSum's input methods.

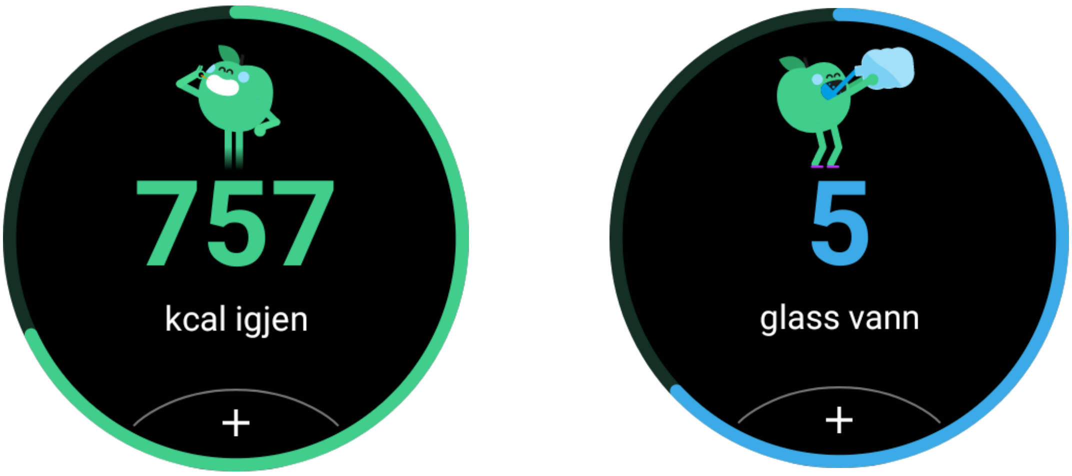

Figure 5: *The input views of LifeSum.*

4.2.4 My Fitness Pal

My Fitness Pal[18] is a digital diary that focuses on nutrition intake and physical activity. The smartphone application leverages a simplistic user interface with the same core functionalities as LifeSum, such as logging meals through QR-code or manual input, logging physical

[18] https://www.myfitnesspal.com/

activity, and goals. The main difference between My Fitness Pal and LifeSum is the social media aspect. In the "My Fitness Pal" platform, the user can add friends and interact with each other's progress.

My Fitness Pal is also available on Android smartwatches. The smartwatch application leverages the same overlay functionalities as LifeSum; authentication, logging calories and consumed glasses of water. However, My Fitness Palis also has a summary screen, which can operate as a tile, giving the user easy and quick access to the summary. Lastly, the authentication flow is dependent on a paired smartphone, making it impossible to authenticate without the smartphone being connected to the smartwatch.

4.3 Literature review

Seiderer et al. [47] implemented and tested a system for nutrition logging. The system integrated multiple systems (smart scale, smartphone, and smartwatch), allowing users to log on to the device they found the most convenient. They found that the participants greatly accepted logging nutrients on the smartwatch. Interestingly, the participants found the smartwatch application convenient when they were in a hurry, as well as convenient due to the discreet nature of the nutrition logging.

Årsand et al. [45] implemented a smartphone and smartwatch platform called Diabetes Diary. The paper presented Diabetes Diary as the first smartwatch application for diabetes management on the Pebble watch (as of 2014, at the time, My diabetes existed on Android Wear). The application supports logging insulin units, carbohydrates, and blood glucose. They suggested that wrist-worn computers may ease the self-management process of diabetes.

Timurtas et al. [48] studied if tailored exercise programs can help people with type 2 diabetes reduce HbA1. The aim was to compare an automatic system (mobile and smartwatch application) with supervised exercise training. The study had some participants use a mobile application and some a smartwatch. From a randomized selection, they found that smartwatches with sensors and physical activity programs show evidence of helping to reduce HbA1c.

Osborn et al. [49] also looked at the positive impact a smartphone and smartwatch stack could have on the HbA1c. They used the One Drop platform implemented for Apple devices and saw that there was an independent relation between calorie-tracking and HbA1c improvement. The paper did not state anything regarding the increased use of smartwatches or details about smartwatch architecture (standalone configuration). The smartwatch application included data logging (undefined what data), viewing statistics, monitoring goal progress, and propagating reminders from the connected smartphone.

Hosseini et al. [50] proposed an algorithm to prevent sudden death among instances of hypoglycemic coma. They used the smartwatch's accelerometer, heart rate sensor, and humidity sensor to evaluate instances of diabetic coma. If the algorithm proposed a seizure, and the user did not dismiss the presented alarm, the watch called the emergency number. However, the suggested system has not been tested.

Karnati et al. [51] implemented a standalone smartwatch application called Real-Time Online Assessment and Mobility Monitor (ROAMM). ROAMM collects patient-generated health data based on a researcher's preferred questions (which are prompted on the watch) and sensor data. The paper presented the logical architecture of the application and its communication with its cloud architecture.

Pektas et al. [46] implemented a modern watchface for Årsand et al. Diabetes Diary. As with the original application, it supported the logging of nutrition, blood glucose, insulin, and carbohydrates. In the new version, the user could monitor heart rate. However, the application did not include documentation of standalone functionality, except for two-way synchronization (over Bluetooth) when connected to the smartphone. Nevertheless, the users who tested the system found the colorful interface motivating, useful, and beneficial to use.

Preuveneers et al. [52] presented a standalone smartwatch application that leveraged cloud services and homomorphic encryption for caregivers to analyze blood glucose, insulin values, and other parameters in a privacy-friendly manner. This algorithm allegedly allowed processing in the cloud without leaking non-encrypted data to the cloud, while healthcare workers could access the processed data. They concluded that with the current resource limitations of the existing (2016) wearable devices and the complexity of their solution, their algorithm would take 5 minutes.

In 2021, Chen et al. [13] published a comparative study regarding smartwatches and smartphones. They collected 223 smartphone- and smartwatch (145 standalone and 78 companion) application pairs implemented for Android and looked at resource sharing, code similarities, and user interaction. Of interest, they state that standalone smartwatch applications are more than a smartphone application replica. Standalone smartwatch applications have a different user story compared with the same functionality on the smartphone or companion application. Hence, standalone applications tend to require more code and permissions when compared to companion applications.

Reeder et al. [11] performed a systematic review of smartwatch studies that focused on health in everyday living. The study was performed on 17 papers from the time period between 2011 and 2016, and most of the studies were patient-related. Among others, the study concluded that many smartwatch applications have power limitations and data quality challenges. Furthermore, they stated that smartwatch applications must have a higher focus on representing data as meaningful information and follow a user-centric design process. They ended the discussion by criticizing the challenges regarding data validation. Comparing the integrity of sensor data, such as sensor data, battery power, and usability, that does not follow a standardized format (e.g., STARE-HI or mini STARE-HI).

5 Requirement specification

5.1 Introduction

This chapter will describe the requirements of the system. The requirements will be systematically described through the Volere Requirements Specification Template [42].

5.2 Functional requirements

A functional requirement is a description of a system's service or system as a whole, commonly expressed as a functionality [42]. These functionalities can vary in dimensions, however, they can often be interpreted through an input, a behavior, and an output. For instance, “Application A should serve bus tickets.” or “The timer should increment over time.”.

Furthermore, understanding the behavior of a specific functionality helps the designing process of the composition of multiple functionalities. A structural way to define such orchestration is through Voleres’s requirement specification template. This template defines a structural way to define functionalities and their relation, all through constraints. These constraints depend on the state of the project, and the following are chosen:

- **Requirement number (Nr.) –** The identifier to a specific functionality.
- **Description** – Definition of the behavior of the functionality.
- **Rationale** – Statement on why the functionality is needed.
- **Fit criterion –** How to measure or benchmark the integrity of the functionality.
- **Dependencies -** What functionalities does this depend on, defined by requirement number.
- **MoSCoW –** Prioritization, MoSCoW is short for must have, should have, could have, and wont have. MoSCoW is not part of the original requirement specification. However, due to limitations, there is a need to scope out a priority of functionalities.

Table 2: *Volere Requirement Specification.*

Nr.	Description	Rationale	Fit Criterion	Dependencies	MoSCoW
1	Display glucose metrics generated from a CGM.	Having access to glucose data is essential for people with diabetes.	Data from the CGM must arrive at the smartwatch in a timely fashion		Should have
2	The application should persist data between application launches	Fundamental requirement of the entire platform. The data is relying on being persisted.	Data that is stored in the application is persisted between application lifetimes.		Must have
3	The application should not rely on a smartphone.	This feature enables decoupling between the smartphone and smartwatch, enabling more availability.	The features of the application should work without Bluetooth enabled.		Must have
4	The application should display data in a graph.	A graph is a compact way to display many data points, such as TIR.		1/6/7/10/11/12	Should have
5	The application should alarm the user if the glucose	Smartwatches can notify the user with haptic feedback and	The alarm is delivered to the user within the	1	Should have

	levels are too low /high	sound. These are effective ways to alarm the user	alarm constraints of the watch (exact / inexact)		
6	The application reminds the user to take medication or monitor glucose at set time.	Medication is vital. Having a lightweight nonintrusive alarm as a reminder may benefit the users.	The alarm is delivered to the user within the alarm constraints of the watch (exact / inexact)		Should have
7	The application should support manual logging of glucose data	The user may use manual blood glucose devices. These recordings should have an entry point in the application.	Logged data persisted between runs.		Must have
8	Logged carbohydrates should be annotated with timestamp: breakfast/dinner	Predefined timestamps reduce the time spent in the application.	The data in the database is with a timestamp		Should have
9	Entry of specific user-defined goals	According to the questionnaire, users appreciate personification. If the user has a	Input method for specific goals		Should have

		specific goal they want to manage, the application should support it.			
10	The application should track physical activity	It is recommended to perform 30 minutes of physical activity each day. Monitoring this can activate the user.	Real-life activity is reflected on the watch		Must have
11	The application should collect steps	Monitoring step count can motivate the user to walk more.		10	Should have
12	The application should have an adjustable step-goal.	People have a different starting point. A set goal can be too challenging for an individual.	Adjusted goal in reflected in the application.	11	Should have
13	The application should persist data on an external server	If the user changes device, there must be a way to sync the data.	The same data is reflected on both ends		Should have
14	The application should simulate authentication	If data should be persisted on an external server, the user must be authenticated.		14	Will not have

15	The application should have an authenticated state and an unauthenticated state	The application should be functional even if the user is unauthenticated.		15	Will not have
16	The application should have an administration mode (parental control).	People who cant manage the application should have an option.	Data is reflected on the separate devices	15	Should have
17	The application should display a TIR Graph.	TIR is a measurement over time. Graphs are suitable for such samples.	Logged blood glucose appears in the graph	1,10	Should have
18	The application should have adjustable font-size	Reduced eyesight is a common complication of prolonged diabetes illness.	The font scales when the value is changed	10	Will not have
19	The application should display a diabetes-management summary of the day. Activity levels, TIR, steps, daily goals.	To reduce time spent in the application, having one view of aggregated data of current progress is beneficial.	The aggregated data is correct, in terms of data points	10	Should have
20	The application should have a "watch face"	When smartwatches are opened, the watch	The opening screen is		Will not have

		face is the first view for the user.	made by the application		
21	The application should have a "tile"	Tiles give the user fast access to application logic and its state.	The screen next to the watchface is made by the application		Should have
22	The application should have a "complication"	Complications are data-points shown on the watchface. Such as battery-life, step count and weather	Data on the watchface is defined by the metrics in the application	20	Should have

5.3 Non-functional requirements

A non-functional requirement is properties that the functional requirements must have, such as performance guarantees or security usability.

Developing a system for smartwatches requires more attention to detail than for smartphones. Planning the interface components, such as navigation, placement of view and input/output, is more complex due to a smaller screen.

Most systems that require a user interface (UI) aim to navigate the user within the least amount of clicks, which is also true for smartwatches. However, interactions with smartwatches are measured in the number of seconds. One reason is that it is a "race" between the smartwatch and the smartphone. Users tend to use the smartphone if a task/activity can be performed faster on a smartphone. Multiple factors, such as arm fatigue over prolonged smartwatch interactions or compliance with a larger screen, affect this behavior.

For example, many smartwatch applications have a companion application on the smartphone, even if the smartwatch application is oriented around a standalone architecture, such as Spotify. Business logic (e.g., Skip to the next song on Spotify) resides on both devices and impacts the interface on both devices. When a user interacts with any of the applications, changes propagate to the other devices.

Any manual logging the user must do should be done within three taps.

Furthermore, in health applications is, security and privacy important. The smartwatch application must ask the user to access sensor data, such that in the background, logic is apparent to the user.

Any in background logging must ask for permission to access sensor data or file system access.

Furthermore, most of the data stored in the application are characterized as sensitive. Therefore, any movement of the data should be regulated with security measurements.

The data stored within the application should only be accessible to the application and no other party

And,

Data exposed to the internet must be encrypted within the standard of transport layer security.

6 Design

6.1 Approach

The system design proposed in this chapter is derived from the survey presented in Section 8.1, the requirement specification, how people manage diabetes, and the review process.

Pektas et al. [46] implemented an Android smartwatch interface for the Diabetes diary [45]. This application supported carbohydrate and insulin logging, heart rate monitoring, and reading step data from the watch. Furthermore, the application provided CGM data from the smartphone.

Another product of interest is Diabetes:M [44]. Diabetes:M is the most downloaded diabetes management application on the Android platform (May, 2022). Like the system of Pektas et al. [46], the user can log nutrition, blood glucose values, and insulin units. Furthermore, the application relies on a smartphone connection (providing CGM data) and does not collect physical activity data.

From these two prominent diabetes management applications, we can see that nutrition logging, blood glucose logging, and physical activity are areas of interest. These features will be included in the system. However, none of the presented applications addresses motivational aspects of disease management.

As stated in the theoretical background, goal setting is one of the most evidence-based activities to promote behavioral change. The only aspect of goal-setting in Diabets:M and Pektas et al. application is the step-goal in functionality for the Diabetes Diary interface.

The presented system introduces an “in the background” goal-setting feature. The idea is that the user can become more engaged with self-monitoring while keeping the complexity of the application low. The goal-setting features are based on setting goals related to the monitoring categories (nutrition, physical activity, and monitoring). For example, a calorie goal - many people with type 2 diabetes are advised to reduce weight. Therefore, having a relationship between the number of calories one consumes and a limit on how many calories one should consume throughout a day may help reduce excess eating. Additionally, a separate in-the-foreground goal-setting category is added. This category allows the user to monitor intrinsic goals or metrics that are outside of the standard logging systems. For instance, log or have as a goal “monitor fasting blood glucose five times each week.”.

6.1.1 Benefits of Standalone Smartwatch Application

Diabetes technology today is highly coupled with the manufacturer. The challenge with this is that the product is coupled with the device. For instance, if a user would like to use Dexcom’s smartwatch application, the connected smartphone must be of the same operating system (Android[19] or Apple[20]). However, if these smartwatch applications were oriented around a standalone architecture, users could mix and match operating systems across different smartphone and smartwatch devices.

6.2 Overview

To understand the overall architecture of the proposed system, one must understand its components. **Figure 6** shows the four components often used to show glucose data on a smartwatch. These devices are a smartphone, a CGM, a smartwatch, and a cloud service. The cloud service is not needed. However, cloud services are often used for persisting data or sharing it between devices.

[19] https://www.dexcom.com/wear-os-google

[20] https://developer.android.com/training/wearables/wear-v-mobile

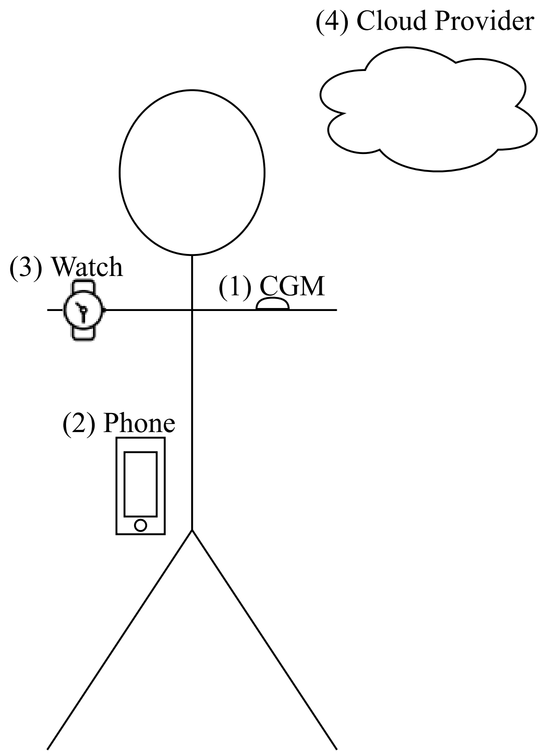

Figure 6: *The four components of a CGM stack.*

In today’s smartwatch applications, the following must happen to have glucose data arrive at the smartwatch: the CGM (1) sensor reads the glucose data of the person, the smartphone (2) pulls the data from the sensor over NFC (or Bluetooth), and lastly, the data is transmitted to the smartwatch over Bluetooth (3). The entire smartwatch application will reside on stale data if the smartphone is unavailable.

The flow between the smartwatch and smartphone can be omitted. Instead, the data can be sent to the cloud (4) by the smartphone and then collected directly from the smartwatch. However, this architecture still depends on the smartphone to collect the data from the CGM.

As described, the next generation of CGM’s will likely support the Bluetooth Low Energy (BLE) protocol. On this assumption, data may be sent directly to the smartwatch. Hence, new higher-end smartwatches can already communicate over the BLE protocol. For example, the Samsung Galaxy Watch 4 can transmit music directly to wireless earplugs over Bluetooth without the need for a smartphone. On this premise, the smartwatch can likewise collect data from an arbitrary BLE device.

6.3 System architecture

6.3.1 Overview

For completeness, the backend will be mentioned in the overview, however, it will not be described in detail. The system architecture consists of a smartwatch, a database (backend), and a CGM. First, the smartwatch is responsible for collecting and displaying data. The data collection is, for example, manual data through user input, such as nutrition, or sensor data, such as physical activity or CGM data. Then, depending on how the user utilizes the system, the data will be sent to the backend. Lastly, CGM data will be sent to the smartwatch and displayed to the user.

The backend’s impact on the system depends on how the user chooses to use the platform. For example, if the user wants to use the system offline, the smartwatch application will accommodate this, and the external service will be unused. However, the external service handles user sessions, data processing, and data persistence. These features are needed such that the user can sync and share data. Furthermore, storing data on the external service will allow other users to subscribe (caretaker or parental control) to one's data and present them on their version of the application. Lastly, an external database is needed to back up the data (e.g., in case of the user changes device).

Figure 7 presents the architecture between the caretaker, the person with diabetes, and the backend.

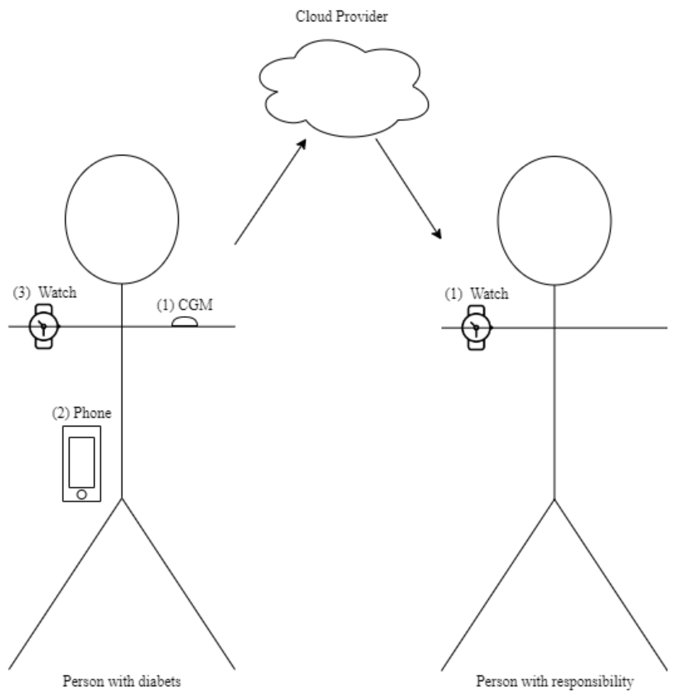

Figure 7: *The overview of components to provide CGM data to a third party.*

The smartwatch will have an application that manages the internal database, CGM communication, external communication, and user input.

6.3.2 Communication

The communication paths are twofold. Firstly, there is a communication flow between the smartwatch and the CGM, and then there is a communication flow between the system and the external service. The CGM and the smartwatch communicate over BLE. The flow depends on the CGM, however, data is regularly pushed to the watch. Furthermore, the smartwatch communicates directly with the external service over the internet interface (WiFi or LTE).

6.3.3 Security and Privacy

The communication between the smartwatch application and the external service is over HTTPS. Furthermore, to authenticate, the users will the Oauth2 protocol be used.

Local storage on the device is within a secure database, only allowing the application to access the database.

6.4 Smartwatch Application

6.4.1 Introduction

This section presents the smartwatch application design. The application on the smartwatch is split into four categories; nutrition, physical activity, goal setting, and glucose monitoring. Splitting the application's content into categories makes it possible to arrange the content logically, resulting in a faster navigation process.

6.4.2 Navigation

The navigation within the application is presented in **Figure 8**. When users enter the application, they are presented with the overlay. The user may navigate to five views from the overlay, four of which are the presented self-management categories. All of the views (except settings) have a representative logging view. When the user has logged the data, the application routes the user back to the logging category. The settings view configures the metrics, font size, and authentication. The categories also contain an immediate view of the current values, such as glucose values and calorie intake. All of the views will be presented in the following sections.

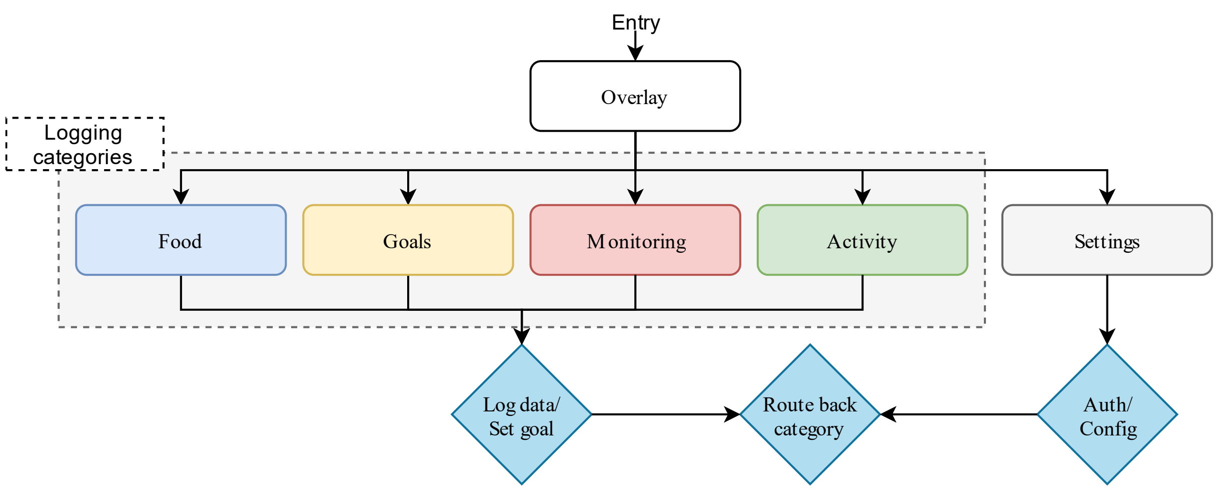

Figure 8: *Navigation tree of the application.*

6.4.3 Information at a Glance

The smartwatch application delivers data within a *second scale* glance. A glance is an interaction with the device without touching the device (flick the wrist, and the lock opens) [53], hence the name. In comparison, glanceability among activity trackers is more important than on smartwatches. Hence, activity trackers provide specific functionality (monitoring activity compared to general-purpose smartwatches), and therefore it is easier to pack the product within a glance. Furthermore, allowing a second-scale glanceability (not immediate glanceability) is also reflected in how we use a smartwatch. It is estimated that 43% [54] of the time spent with a smartwatch is less than 6 seconds, compared to 70% [55] on activity trackers.

The suggested system has a second-scale glanceability of data. Smartwatch applications can access and view data quickly through a tile.

6.4.4 Interaction between the categories

Since nutrition, physical activity, and blood glucose values are intertwined, it is natural for them to impact each other. Therefore, the user is notified to log blood glucose two hours after logging dietary intake (calories or carbohydrates) in the proposed system. However, the notification will not be sent if the user has logged blood glucose within two hours.

6.4.5 Notifications

The proposed system will push notifications to the user regarding critical blood glucose values. Some of the goals that the user may track can generate alerts or reminders. For example, to log fasting blood glucose. Logging fasting blood glucose must be between eight and twelve hours after eating. The notification can then be sent statically (each morning, overnight) or if the user has not logged nutrition since a set time.

6.4.6 Landing page for the overlay

When the user opens the smartwatch application, the main overlay is opened. The main overlay presents the root of the navigation tree, splitting the path of each category. As seen in **Figure 9**, the user is presented with five buttons. Four of the buttons navigate to the corresponding management category, while the fifth navigates to the settings panel.

The overlay introduces the application's color scheme and its icons, blue for food, yellow for goals, red for monitoring, and green for physical activity.

Figure 9: *Landing page of the smartwatch application.*

6.4.7 Food view

Logging nutrition on the smartwatch has multiple benefits. First, the user can become more deliberate about eating habits: having the number of today's calories consumed displayed on the watch at all times. Second, the user can be reminded to check blood glucose after a meal, given that the user logs the nutrition consecutively after eating.

The nutrition view follows a blue color theme and presents the user with the current number of logged calories and carbohydrates, each with the option to add a new entry. The view also has the option of changing the daily calorie goal.

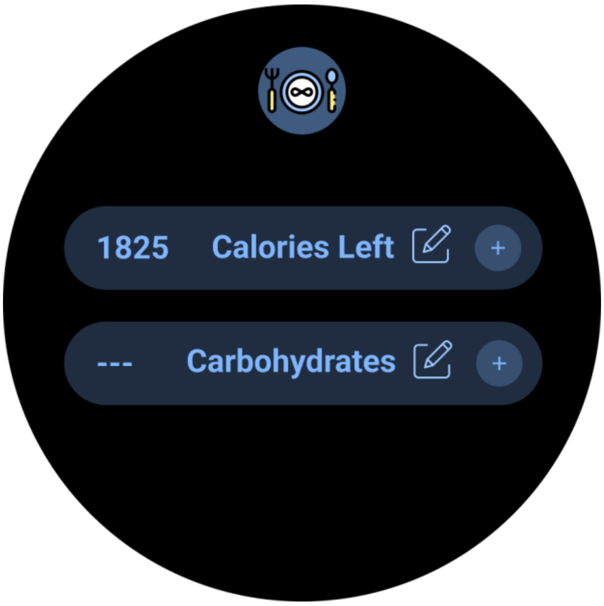

Figure 10: The nutrition view.

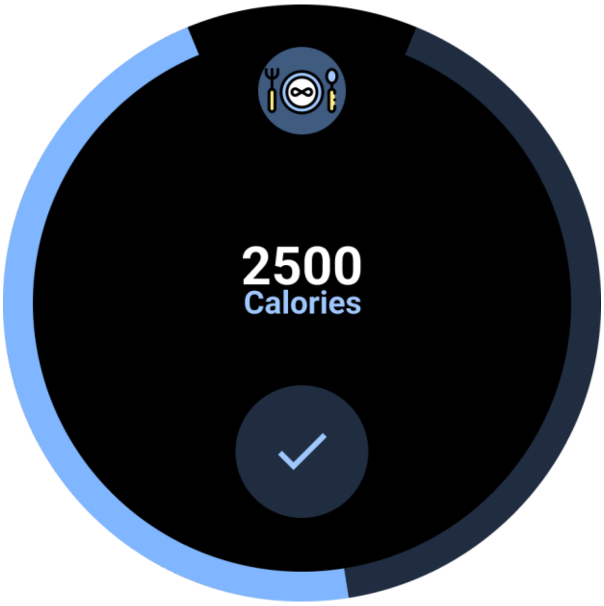

Figure 11: The logging nutrition view.

Changing the goal will lead the user to the view presented in **Figure 11**. The user can rotate the bezel to enter a specific value and submit it by tapping the checkmark. Here we see that the user may set a target number of calories to eat each day.

To log nutrition (calories or carbohydrates), the user taps on the presented plus icons in **Figure 10**. Logging nutrition consists of two steps: (1) enter the amount and (2) enter the time.

The plus icon will lead the user to the first step needed for logging nutrition. In the first step (see **Figure 12**), the user has four options: log a small, medium, large meal, or specify the amount. If the user wants to specify the amount, a similar view as the nutrition goal is presented.

The time selection view is presented when the user has selected the food size (see **Figure 13**). Time logging is presented with a Picker object (time scrolling component). Finally, when the user has logged all the metrics, the user is sent back to the root nutrition view.

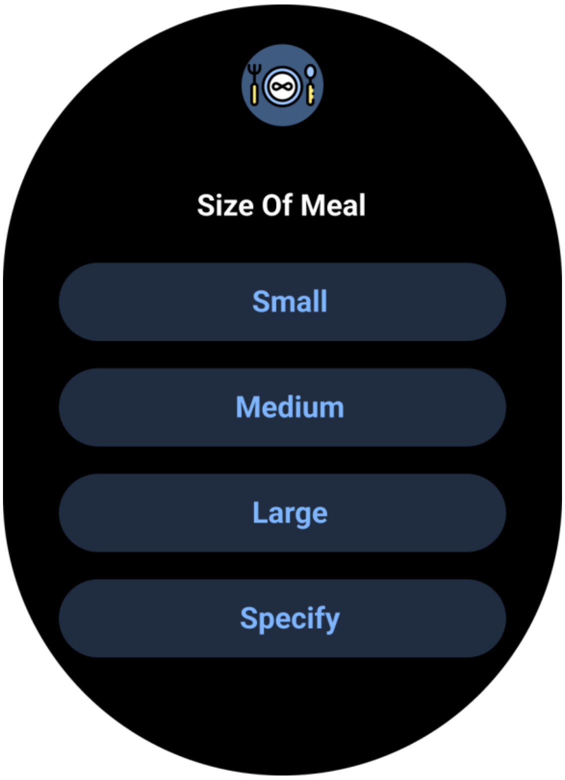

Figure 12: *The new meal entry view.*

Figure 13: *The view for annotating the timestamp in the view.*

6.4.8 Goal setting view

The goal-setting view (see **Figure 14**) shows the user's progress on the current goals. The goal view follows a yellow color scheme presented on the landing view. Furthermore, the goal-setting view aims to provide the user with goals related to diabetes management and other activities that the user wants to manage. In this view, the user can see the progress presented with concrete values and a progress bar. At the bottom of the view, the user may track new goals by tapping on the plus icon. If the user taps on an already tracked goal, the user is presented with the view in **Figure 15**. This screen displays the goal in fullscreen, and if the user rotates the bezel on the watch, the goal progress will increase.

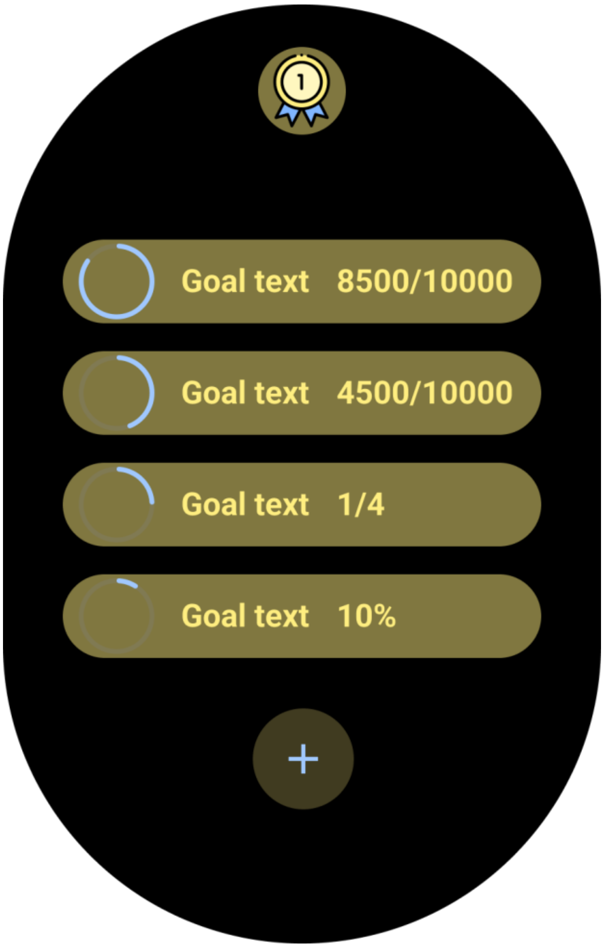

Figure 14: The goals view.

Figure 15: *The view for increasing the progress of a goal.*

If the user proceeds to track a new goal, the view in **Figure 16** is presented. In this view, predefined goals are presented in a Picker object. The user may scroll through the goals and continue the process of tracking a new goal by tapping the arrow pointing to the right.

The next step in tracking a new goal is presented in **Figure 17**. Here the user selects the target of the goal. Tapping on the checkmark completes tracking a new goal and presents the user with the initial goal view.

An important factor of these goals is that the page must have corresponding target values. For example, the goal "drink ten glasses of water" must have the corresponding target, " 10 glasses" and not "10000 steps".

Figure 16: The view for selecting goals.

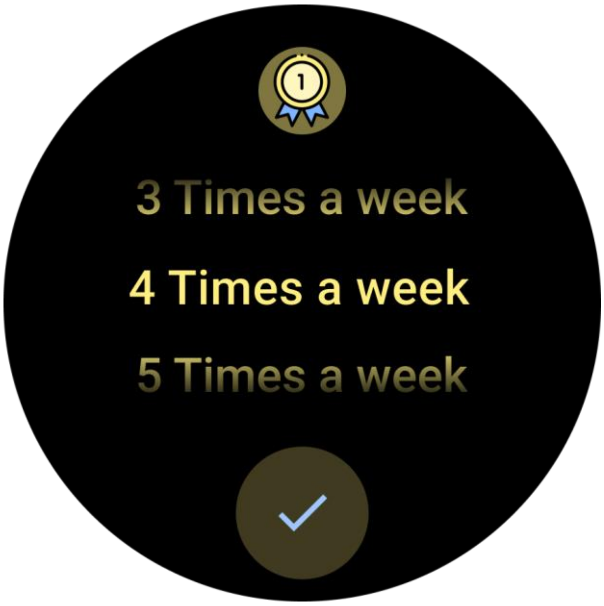

Figure 17: The view for selecting the goal target.

6.4.9 Monitoring view

The monitoring screen (see **Figure 18**) follows a pink color theme. The user is presented with the last recorded blood glucose value, when the last value was measured, and the blood glucose tendency. The values presented on this view are automatically collected from the CGM over Bluetooth and the logged values. Additionally, the user is presented with two buttons, one for navigating to the history view and one for logging a new blood glucose entry.

Figure 18: Entry point for the monitoring view

Figure 19: The log blood glucose view

The plus icon (bottom right in **Figure 18**) in the monitoring view navigates to the view for logging a glucose recording. When in the logging view (see **Figure 19**), the value on the screen changes as the user rotates the bezel on the watch. Logging a blood glucose value will send the user back to the initial monitoring view.

The menu icon (bottom left in **Figure 18**) navigates the user to the history screen. The history view has two formats, one for presenting each sample and one for showing the TIR representation.

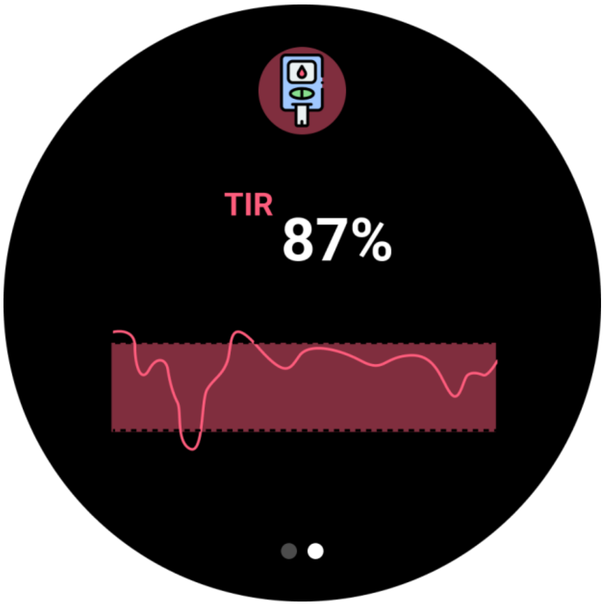

***Figure 21:** The TIR graph.*

***Figure 20:** The Blood glucose history view.*

The history view displays the time and value of the past blood glucose recordings (see **Figure 20**). Additionally, the user may remove or edit values from the history view.

If the user swipes to the left from the history view, the user is presented with a TIR graph (see **Figure 21**). This graph is another representation of the values presented in the history view. The page indicator at the bottom indicates the current view.

6.4.10 Activity view

The minimum goal of the physical activity logic of the smartwatch application is to make the user perform 30 minutes of physical activity each day [56]. In order to do so, the application must track physical activity and make the user aware of the current activity levels and provide motivation for physical activity. Most smartwatches have access to the integrated accelerometer sensor and the heartbeat sensor. Together, these can be used to estimate the user's daily activity minutes.

The physical activity view presents (see **Figure 22**) a green color theme, the steps performed today, and the activity minutes today. Additionally, by pressing the button next to the steps, the user may change the daily step goal (see **Figure 23**). The step goal can be adjusted by turning the bezel on the watch, and submitted by pressing the checkmark, which brings the user back to the initial physical activity view.

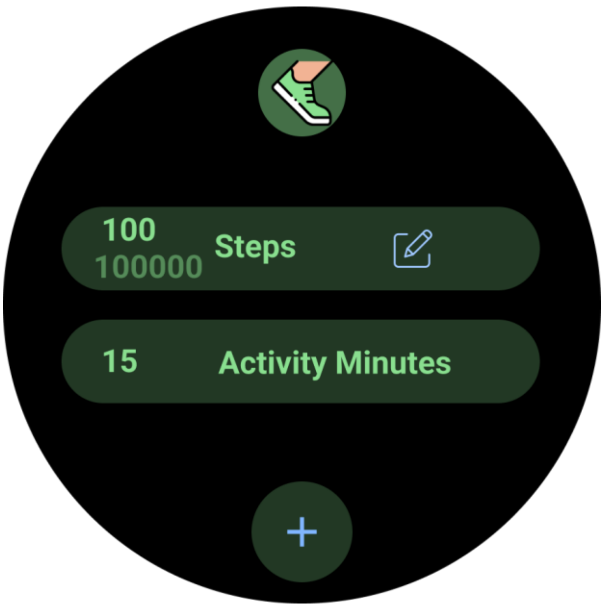

Figure 22: *The physical activity view.*

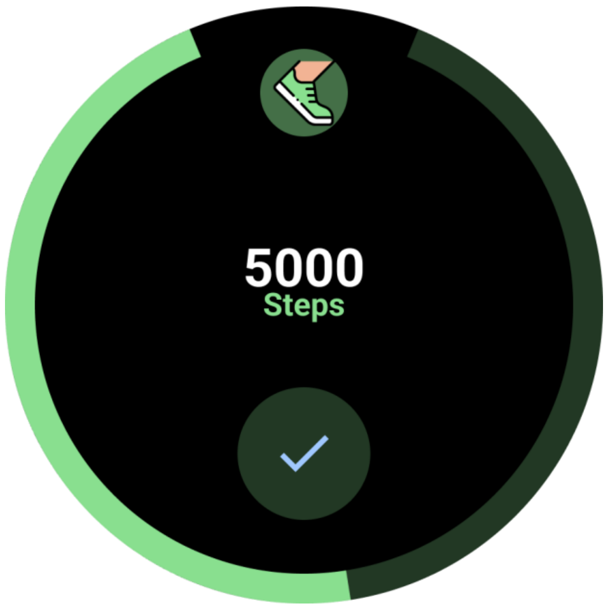

Figure 23: *The step goal view.*

6.4.11 Settings View

The settings view is presented in **Figure 24**. From here, the user can perform the authentication process or navigate to the metrics view. The metrics view (see **Figure 25**) allows the user to change the units presented in the application (mmol/L to mg/dL) and the font size.

The user can also configure the parental/caretaker control on the settings view. When users sign in, they are presented with an ID (see **Figure 26**). This ID can be used to subscribe to other users' data. The connection can be closed by pressing the ID of the connected user. Users who are invited to be overseen must agree to the subscription. Users who are subscribed to can detach users by pressing the button checkbox in **Figure 27**.

The signup flow must be done outside of the application.

Figure 24: *Settings View.*

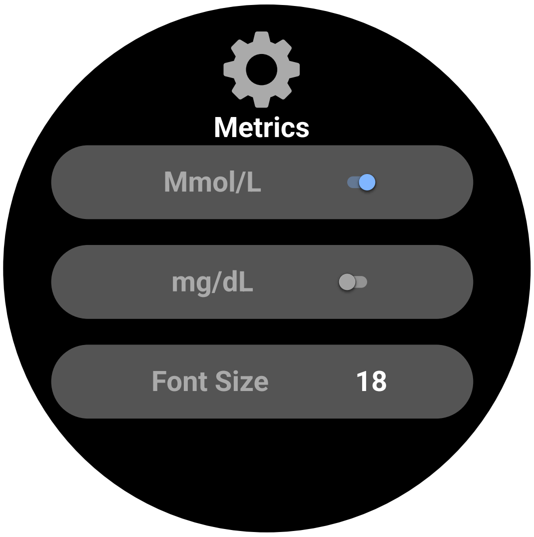

Figure 25: *Metrics View.*

Figure 26: *The connect view when authenticated.*

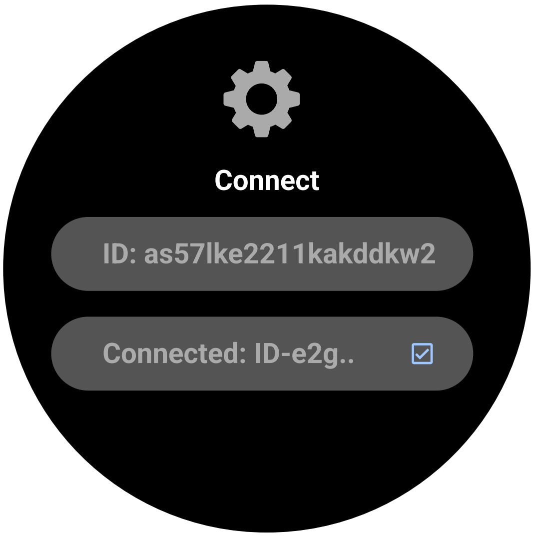

Figure 27: *Settings view when someone is subscribed to the data.*

7 Implementation

7.1 Overview

This chapter will present the implementation of the smartwatch application. The implementation includes most of the application's architecture, the physical activity, and the goal view. Furthermore, a glanceable tile is implemented. The implemented requirements from Chapter 5 will be marked with parenthesis and the requirement number.

7.2 Smartwatch Application

The system has been implemented on a Galaxy watch 4 running Wear Os 3.

Firstly, one key element differentiates an Android smartwatch application from an Android smartphone application. In the application's " description" the device is set to watch. When this attribute is set, the application will be compiled and run within the constraints of an Android smartwatch application. Furthermore, the standalone feature is set accordingly to ensure further that the application is interpreted as standalone (nr. 3). How the presented attributes (smartwatch and standalone) are set can be seen in **Code Listing 1**.

Code Listing 1: *The birth of a smartwatch application.*

```
<application>
...
<uses-feature android:name="android.hardware.type.watch" />
...
  <meta-data
    android:name="com.google.android.wearable.standalone"
    android:value="true" />
...
</application>
```

A sidenote for annotating the application as standalone is the application installer on Android watches. In order to have applications appear in the application store on wearables, the application must be annotated as standalone.

7.2.1 Architecture

Rendering data structures to the screen is costly; data must be retrieved, processed, and indexed. Therefore, planning the underlying architecture before the implementation phase is essential to achieve high cohesion and low coupling between components. The underlying architecture can be expressed through software design patterns. The concept of a software design pattern is providing a general solution to a common problem. These common problems can be small, such as, at the component level (e.g., composite pattern: solves challenges regarding multiple inheritances), or large, at the system scale (e.g., model-view-controller, addresses component communication and the relation between UI and data).

Smartwatch applications must be fast. Otherwise, the barrier of using the application on the wrist becomes more prominent, and the user may instead perform the action on the smartphone. Therefore, it is essential to only process what the user needs. For example, if the desired view *only* displays step count, irrelevant data should not be retrieved (from disk or external resource), such as heart-beat data. An architectural design pattern that accommodates data and its representation in a view is the Model-View-ViewModel (MVVM), among others. However, Android recognizes the Model-View-ViewModel architecture as the best practice, and therefore, is chosen for this application.

Figure 28 depicts the communication flow between three main components of the application's MVVM architecture. Rightmost in the figure, we have the view layer (known as the user-interface layer), then the ViewModel, and lastly, the repository (Model layer). Lastly, the arrows indicated linking between data types or references.

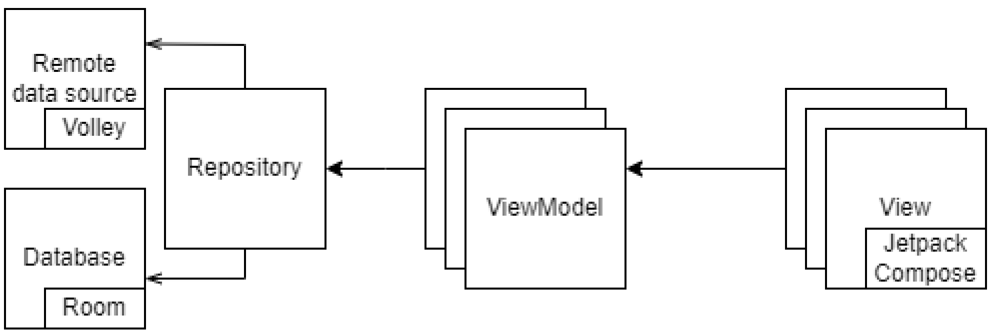

Figure 28: *The system architecture of the application.*

The view layer displays data and reflects data changes to the user. Such data changes can be external API calls or local logic. Furthermore, the view layer primarily consists of component markup, orchestrating the visible components. For instance, the views presented in the design chapter are realized in the view layer.

Then, we have the ViewModel component. The ViewModel ensures the linking between the data and the views, and retrieves data of interest from the Model. For instance, the steps shown in the physical activity view are stored within the ViewModel. It also keeps intermediate data changes in memory, for example, when the user rotates the bezel on the watch. The rotary input is stored in memory until the user exits the view, instead of writing to disk on every rotary event.

Lastly, we have the repository (the Model in MVVM, depicted to the left in **Figure 28**). This component centralizes and persists the “raw” data and manages API calls (HTTP or Bluetooth in the future). Hence, it is in this component where the business logic resides. The database engine is SQLite, which is provided through the Room framework.

In summary, MVVM separates the UI from the business logic.

The simplified code example in **Code Listing 2** describes the initialization flow of the goal-setting view. First, the repository is initialized. The repository has low-level functionalities such as fetching data from HTTP and accessing the local database, in this example, retrieving the goals from disk. Then this reference is passed to the View, where the View initializes the ViewModel. Next, the ViewModel interacts with the repository and makes data available for the View. Lastly, the View displays the data. Looking at the body of the View, we can see that there is low coupling between the data and the screen implementation.

__Code Listing 2:__ Flow of the MVVM in the system.

```
//Top Level
override fun onCreate() {

  //initialize the Repo, this is shared between all views

  val repository = (application as GoalsApp).goalRepo

  composable(route.Goals) {Goals(repository)}

  ...

}

//View
@Composable
fun Goals(
     viewModel = GoalsViewModel(repository)//initialize the ViewModel

) {
    val goals = viewModel.goals
    Text("These are your goals: $goals")
}


//ViewModel
class GoalsViewModel(private val repository: GoalRepo) : ViewModel() {


    var goals = repo.getGoals()
    //Goal logic
    //e.g. repo.updateGoals(goals or, convert metric units to imperial units
}
```

7.2.2 Components

The application uses Jetpack's Compose library for rendering components. Compose is a declarative programming interface, in contrast to imperative. Imperative programming is a programming paradigm that uses statements to change a program's state. When working with UI's and views, the component must be updated to reflect the state, this can be `button.setText(String)`. Unfortunately, this paradigm is prone to errors and inconsistency since the state must be set "manually". Declarative, however, expresses the logic of a computation without describing its control flow[21]. Jetpack Compose will automatically re-render (declarative) components that depend on a state. Developing UI's on

[21] https://codeburst.io/declarative-vs-imperative-programming-a8a7c93d9ad2

the wear platform has been done through the imperative paradigm until the alpha developer preview was released in late October 2021[22].

Composable

A composable is a component in the UI library Jetpack Compose. For example, the component *Button* in the familiar markup Language HTML, is a *Button* composable in Compose.

The design follows a color theme (pink, green, yellow, and blue), and accordingly, four color themes were *implemented*. The benefit of implementing a color scheme is the ease of use. Pages, views in Compose, can be wrapped in a theme, theming all subcomponents accordingly. Additionally, a set of generic Composables was implemented. The generic Composables implemented were Chips, Lists, and Pickers. Chips are the navigation buttons presented in the design (see **Figure 29**). These chips are ordered within Lists. The reason Lists are mentionable is because they provide an API for shrinking items at the top and bottom of the screen. This gives the user the feel of scrolling and more screen space for the developer. The Pickers are used for selecting timestamps and tracking manual goals.

Navigation

The frontend library also handles the main navigation logic in the application. The user can navigate between views by tapping on components and navigate back by swiping or pressing the back button.

Views

The implemented categories from the design are represented as separate views. Views consist of a bundle of Composables, resulting in a complete page. The reason for splitting the content into different views (and not changing each component individually) will be apparent in the presentation of the application's architecture.

Figure 29 presents a screenshot from one of the implemented views, based on the design presented. Tapping on any of the chips will lead to the representative view. The user may scroll the screen or the bezel on the watch to navigate the view.

---

[22] https://developer.android.com/jetpack/androidx/releases/wear-compose#1.0.0-alpha09

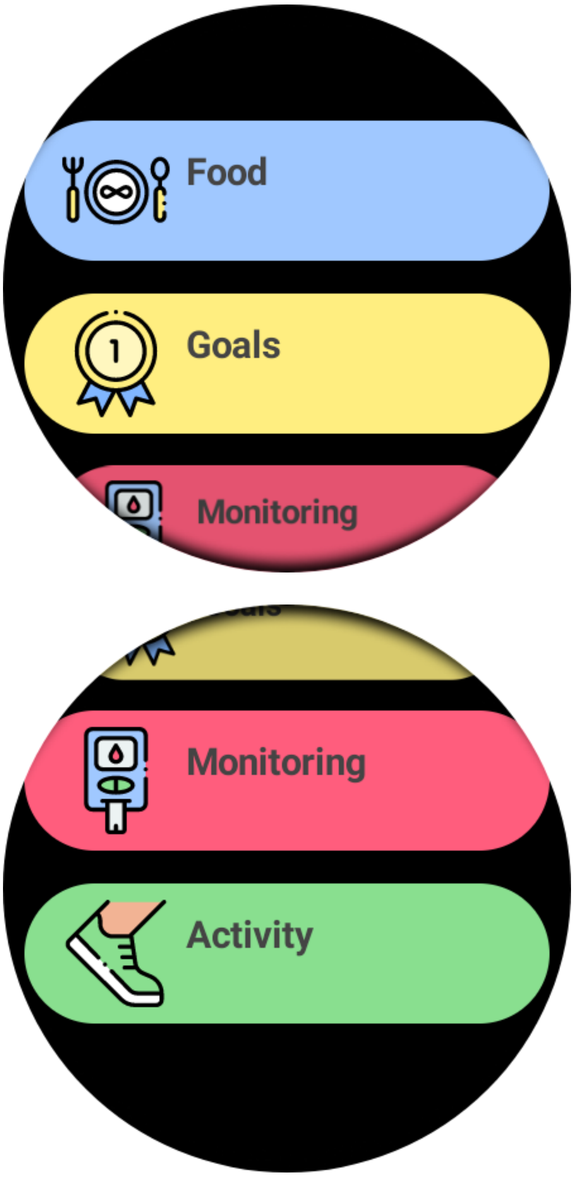

Figure 29: *The screenshot of the Implemented overlay view.*

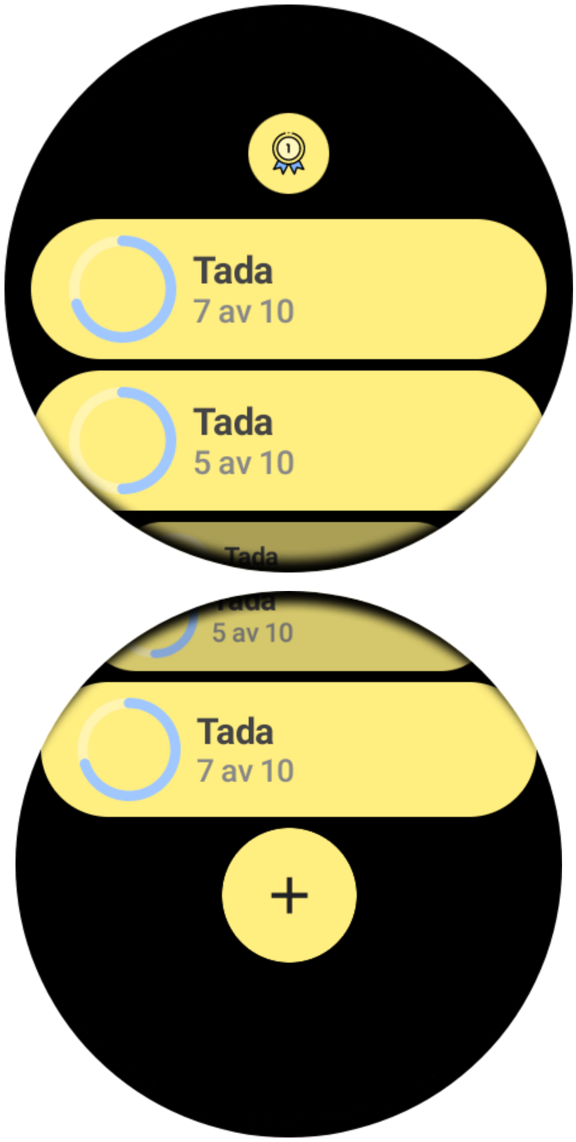

Figure 30*: The implemented goal view*

Goals

The goal screen is the overview of ongoing goals and their progress. The ViewModel collects ongoing goals from the local database on the watch and sends them to the view when they are retrieved. Tapping on a goal will lead to the goal screen, and tapping on the plus button will lead to the new goal screen.

Goal

The goal view is the view where the user logs the goal progress. The progress indicator increase or decreases as the user turns the bezel on the watch. The progress state is immediately stored in memory (in the ViewModel) and written to disk when the user exits the screen. All goals are manually tracked.

New Goal

The new goal screen allows a user to track new goals (nr. 9). The goals the user can choose from are fetched from the repository, which in turn the repository collects goals from an internal state. However, the current implementation can be augmented to be fetched from an external backend. When the goal is selected, the goal is written to the database.

Activity

When entering the activity menu for the first time, the user is prompted to agree to “read physical activity” - permissions. This is because the activity uses the Samsung Health Library to display steps, and the user must confirm that the user allows the watch to track activity. The user's preferences in this dialog are preserved, removing the need to request it each time the user opens the activity view.

The activity screen displays physical activity metrics and their progress towards the user-defined goal (nr 11, 12). The activity data is retrieved from the Samsung Health Library, and the goal target is collected from the disk.

7.2.3 Tile

In addition to the screens described in the previous section, a tile was implemented (nr. 21). The tile lives outside the overlay application and is accessible from the watch's home screen (next to the watchface). The implemented tile can be seen in **Figure 31**. Here we can see progress indicators around each category. Programmatically, these indicators collect data from a repository. However, this data is mocked and not linked to the repository connected to the main application. The progress indicator around the nutrition icon indicates the percentage of maximum calories consumed. Likewise, the physical activity indicator displays the progress towards the step count goal. In the upper right corner, we see the TIR percentage. Lastly, the goal progress indicator displays the aggregated goal completion. For example, having two goals at 75% and 25% will show 50% progress on the goal indicator.

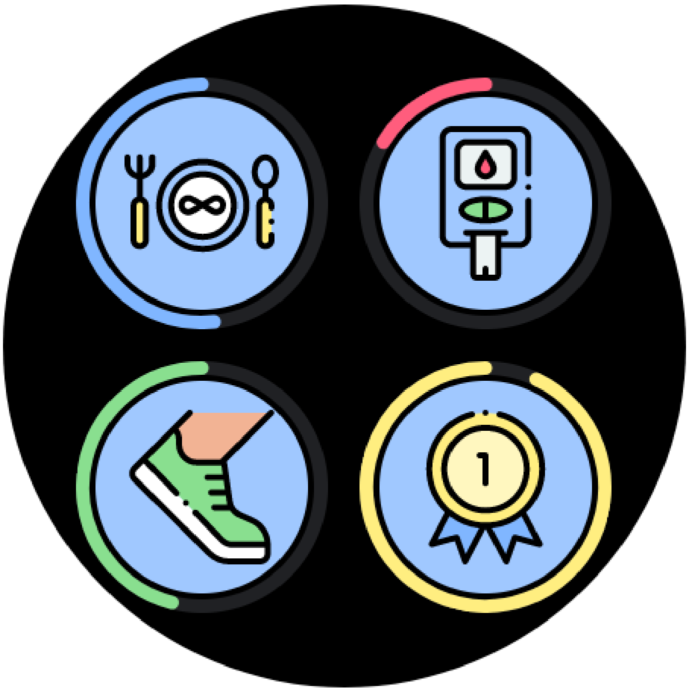

Figure 31: The smartwatch tile.

When the user taps on any icon on the Tile, an *Intent* (programmatic request to open a specific application) is sent from the tile service to the main application. The Intent is parsed in the main application, and according to the Intent's destination, the requested screen is opened.

The code snippet depicted in **Code Listing 3** presents the relation between the Tile and the application. First, the Tile has a method that opens the main application. The subsequent lines evaluate if the application was "started" from the tile service or from the app drawer. If it were opened from the tile, a destination is passed from an Intent. On the other hand, if the application were opened from the application drawer, there would not be a passed Intent, and the start screen is set to the home menu.

Code Listing 3: *Code snippet of the navigation logic within the application.*

```
//===============================

//Tile service

.setOnClick(
    .setClassName("diabetes.app.overlay.MainActivity")
      .addKeyToExtraMapping("fromTile")
      .setValue("Goals")

    ...

//===============================

//Overlay application

override fun onCreate() {

  var tileEntry: String? = intent.getStringExtra("fromTile") //Tile
event

  if (tileEntry == null) {

    tileEntry = route.Home //No intent, open home screen

  }

  Navigation(navController = swipe, startDestination = tileEntry) {

    composable(route.Home) {Home()}

    composable(route.Food) {Food()}

    composable(route.Goals) {Goals()}

    composable(route.Activity) {Activity()}

    composable(route.Monitor) {Monitor()}

}
```

The tile service has a repository containing the data displayed. The data in the tile service should be aggregated data, and this data should be located on a disk. Long processing/remote API calls are not advised due to the unpredictable latency of computation. Hence, the tile should be available as soon as the user looks at the watch.

7.2.4 Database

The local database on the smartwatch is an SQLite database provided by the Room library. SQLite is a relational database following a set schema. The database is used for persisting Goals between sessions (nr .2). Furthermore, the goal schema consists of four fields; ID, description, progress, and goal (target).

7.2.5 Design choices

Step Sensor provider

Collecting steps can be done in various ways. For example, when Årsand et al. [45] implemented the first combined smartwatch and smartphone platform for diabetes management, they had to implement their own step counter. This counter performed some algorithms based on the data read from the accelerometer. However, this is not a feasible solution for this project. Firstly, implementing such an algorithm is time-consuming.

In the first iteration of the smartwatch application, the MeasureClient in Android's health library was used. This API is an abstraction layer on top of the sensors. Instead of reading raw data (data streams of changes in accelerometer position), one can read processed data (steps deducted from accelerometer data, heartbeat data, etc.). This library solved challenges related to retrieving activity data. However, architectural challenges, such as continuously reading steps while the application is closed and storing them, were not solved.

Due to these limitations, further development led to the use of Android's new alpha release of the Health Platform API[23] . The Health Platform API is a standard interface for reading, writing, and sharing health data across health applications. This library is powerful because developers can integrate towards one interface instead of tailoring synchronization logic to each application.

The smartwatch application uses this library to collect activity data. However, this leads to the smartwatch being dependent on other platforms publishing data to the Health Platform. In the case of the Galaxy Watch 4 (the driver of the application implementation), Samsung Health is publishing its processed step data (among others) to the Health Platform.

The benefits of collecting the step data from the Health Platform processed by Samsung Health is that the processing is already done, and the programmer does not need to deal with storage, as data is served upon request. Another benefit is that the sensor data is only read once from the sensor, instead of once by Samsung Health and once by the smartwatch

[23] https://developer.android.com/training/wearables/health-services/health-platform

application (e.g. by the first introduced MeasureClient). The drawback, however, is that the application depends on an external source, in this case, Samsung Health.

7.2.6 Future work

The implementation described in this chapter is not complete compared to the platform described in the design. In a further development of the application, prioritizing local dependencies first would be beneficial (completing the monitoring and nutrition screens). Then, apply business logic such as parental control and Bluetooth communication to a potential sensor.

The current implementation contains the essential parts of the software system, e.g., the initialization and use of the database, system architecture (relation between screens and logic), and network calls. These are all of the must-have features from the requirement specification. Moreover, with the current implementation, enhancements and further work can be applied with ease. For example, the repository can be expanded to support Bluetooth.

One experience that impressed during the development phase was the health platform API. Hypothetically the health platform could provide glucose data to a given application, given that another provider/application/platform submits data to said platform. If this is the case in the future, the management of Bluetooth will be unnecessary. For example, if the enterprise applications submit data to this platform, developers could implement health applications with a low effort.

8 Results

8.1 Survey

The survey performed ahead of this project is the basis of functionalities and design introduced in this project. In this section, selected questions from the questionnaire are presented descriptively, in conjunction with histograms. All of the questions were optional and varied between multiple-choice, single-choice, and ratings 1-4 (one is not important at all, and four is very important). Furthermore, the number of participants where N = 71.

Wearables and sensors

The wearables and sensors category addresses topics around technology, features, motivation, and their relation to wearables.

The first question in the "Wearables and sensors"- category investigates what technologies the participants use. In the question "Which of these technologies for health tracking do you regularly use?", multiple answers could be selected.

Figure 32 presents the total percentage of each technology the participants answered that they use. We can see that over 80% of the participants use health-specific measurements, and 50% use physical activity trackers.

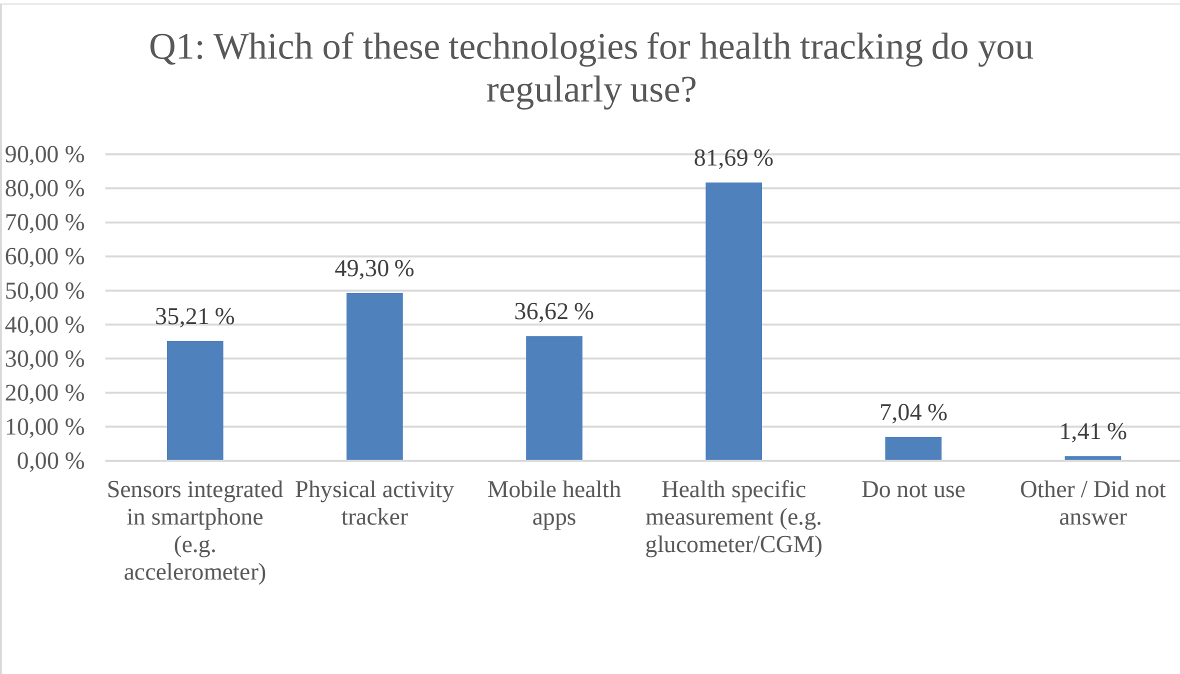

Figure 32: *Diversity in technology among the participants.*

The second question is formulated "How important are these features when choosing a wearable device?". The features were rated from 1 to 4, where 1 is not important at all, and 4 is very important. In **Figure 33,** we see that the most important attributes are sensor accuracy/range values and useability/quality. In contrast, known or specific brand/price are the least important attributes.

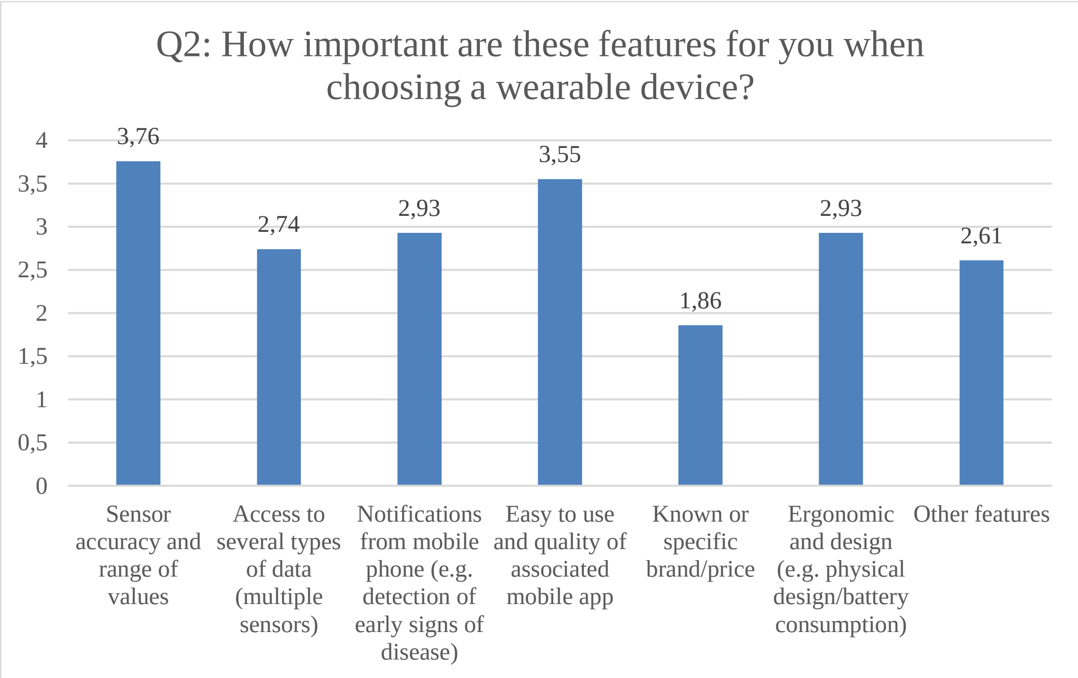

Figure 33: *Important features for wearables.*

Moreover, the next question focused on motivation. The question asked, "Which features would motivate you most to use a wearable sensor longer?" and was a single-choice question.

Figure 34 shows the average answer for each feature. This question has the highest variation in responses among the selected questions. Personalized feedback and ease of use/non-disruptiveness scored the highest, with 50.7% and 32.4%, respectively. The group-based feature "social media" had no responses.

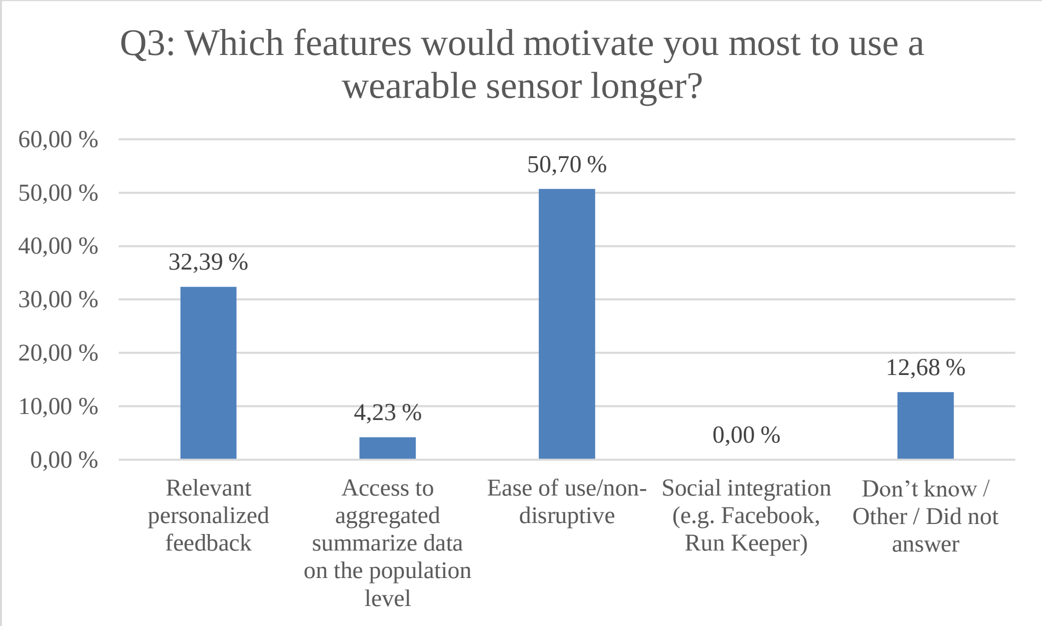

Figure 34: What feature motivates the participants to use a wearable sensor longer.

The last question in the wearables and sensors category focuses on health. The question is stated as "How important are these specific health-related features for you when choosing a wearable device?". All features could be rated from 1 to 4, where one is not important at all, and 4 is very important. **Figure 35** displays the average importance of each feature based on the participants' answers. The figure shows that disease management has slightly higher popularity than health deterioration and alerts.

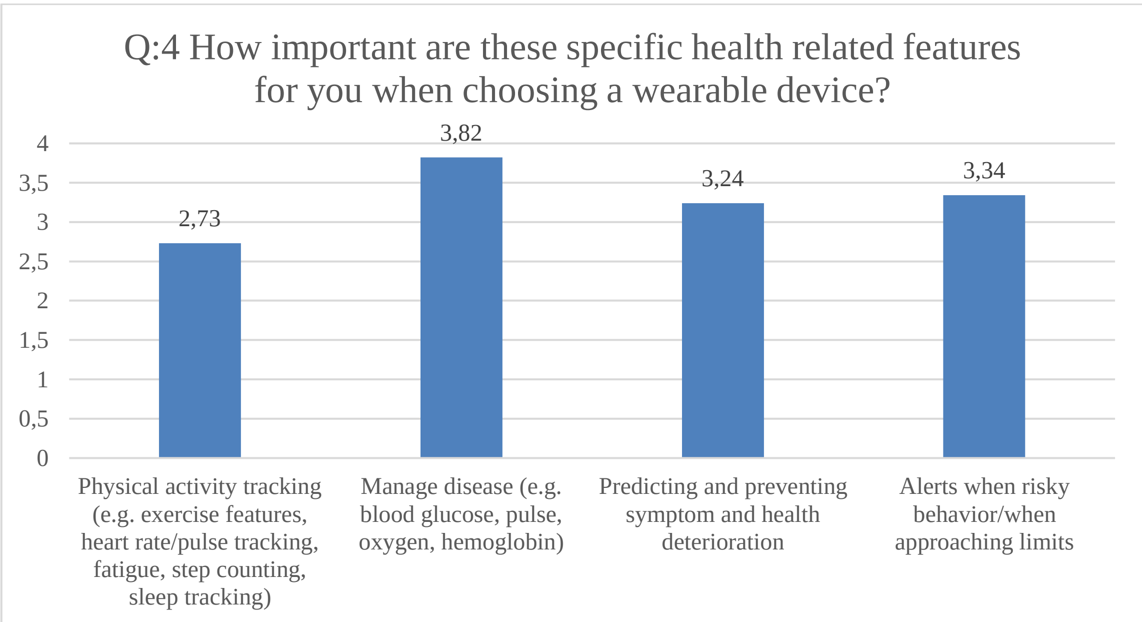

***Figure 35**: The participants' view of the health-related features of wearable devices.*

Mobile Applications

The next category in the survey was called mobile applications. Its focus was on mapping out the use of mobile applications among the participants.

The first question was, “How important are these features when choosing a mobile health app?”. Again, all features were rated from 1 to 4, where 1 is not important at all, and 4 is very important. **Figure 36** displays the average value of all the responses to each feature, respectively. For the average participant, all features are of almost equal importance, however, simplicity and trust have a 0,5 higher score than the second most, at 3.54.

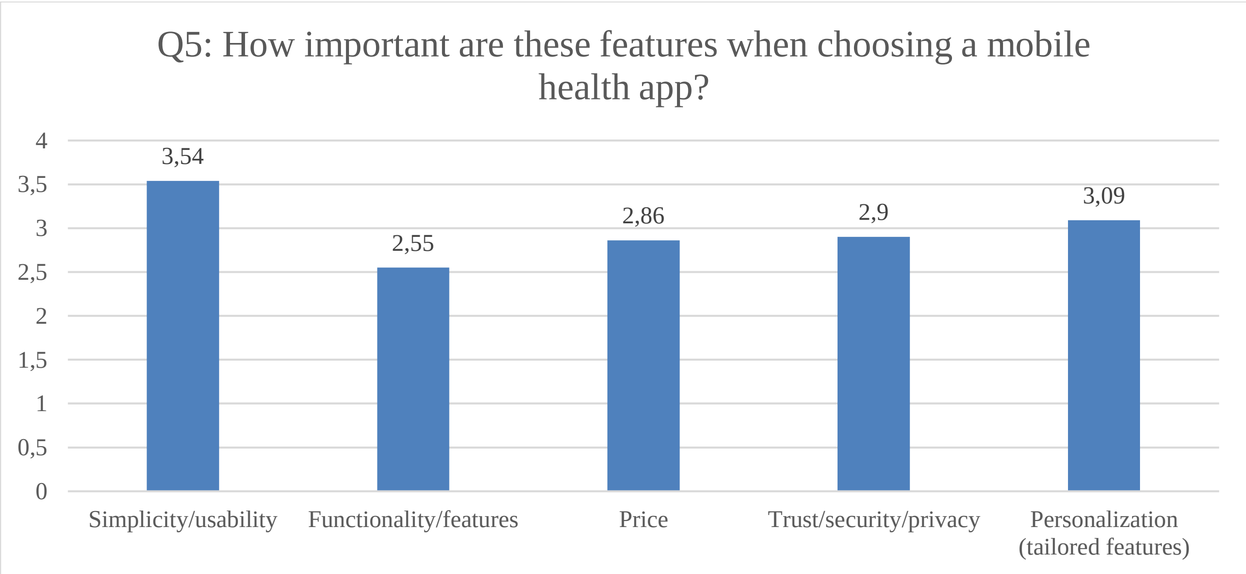

Figure 36: *The participants’ view of mobile health features importance.*

The second question asked, “How do you decide if a mobile application is trustworthy?”. Again, this is a multiple-choice question. **Figure 37** shows that the most crucial factors of trustworthiness among the participants are personal or others’ experience, followed by product specification/provider reputation.

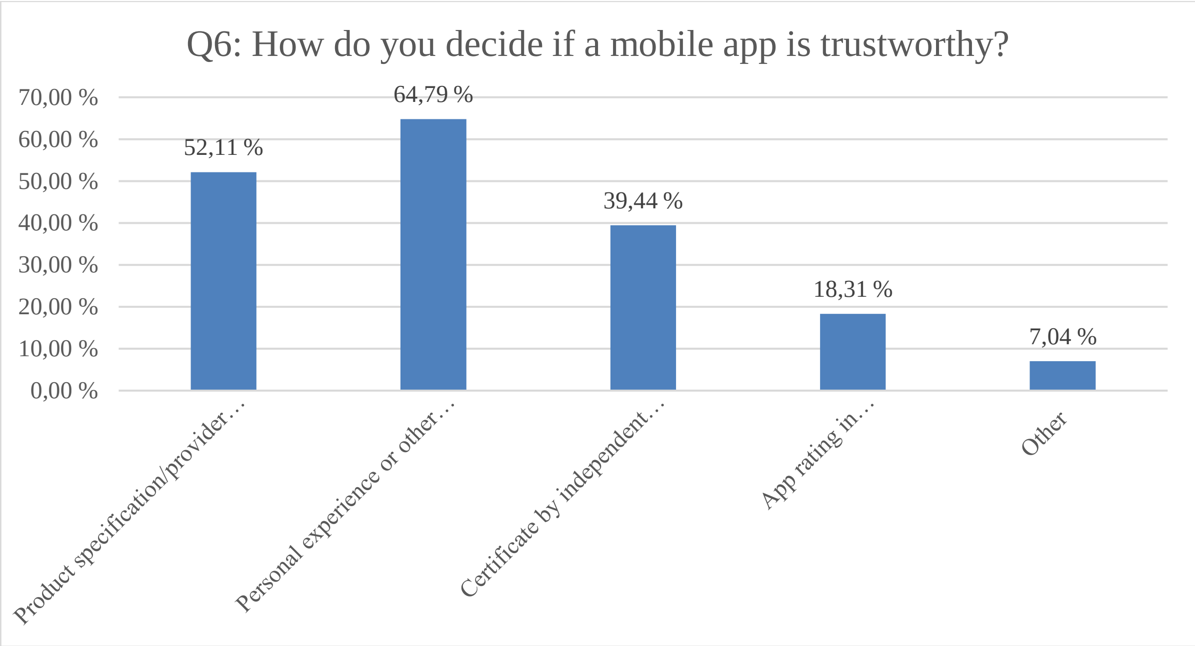

Figure 37: *How the participants decide if an app is trustworthy.*

Logging

The first question in the logging category was, “If you are required to do manual logging (registration) in a mobile health app, how important are these criteria for you?”. The options could be rated from 0 (I don’t know) to 4. **Figure 38** shows a diagram of the average score of each option. The figure shows that the “Easiness and simplicity” option has the highest average score of 3.72.

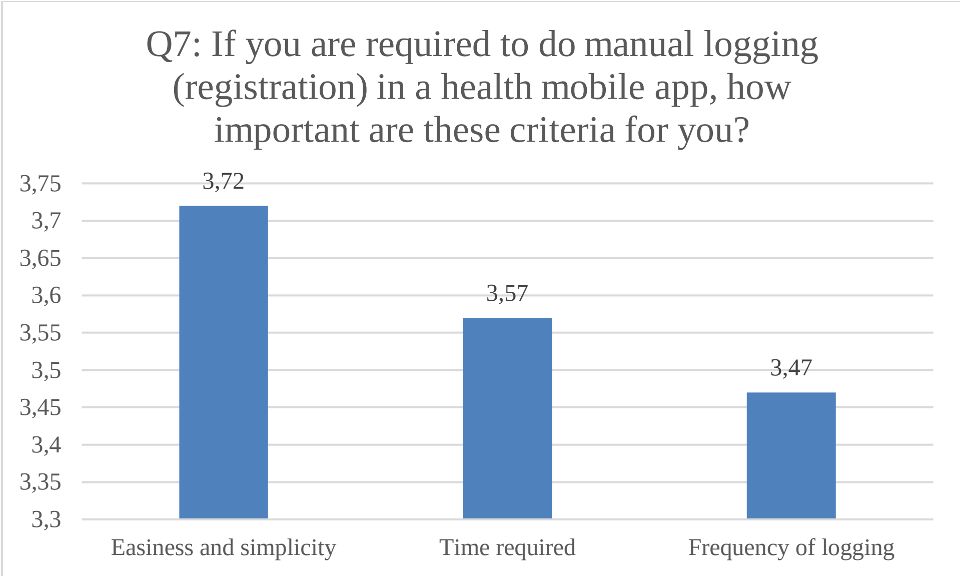

Figure 38: *Important features of manual logging.*

The next question was, "How willing would you be to manually log or register the following types of data?". Again, this question could be rated from 0 (I don't know) to 4. **Figure 39** shows that the participants are most willing to manually log medication and physical metrics, scoring 2.63 and 2.69, respectively. The remaining options score over half a point less than medication and physical metrics.

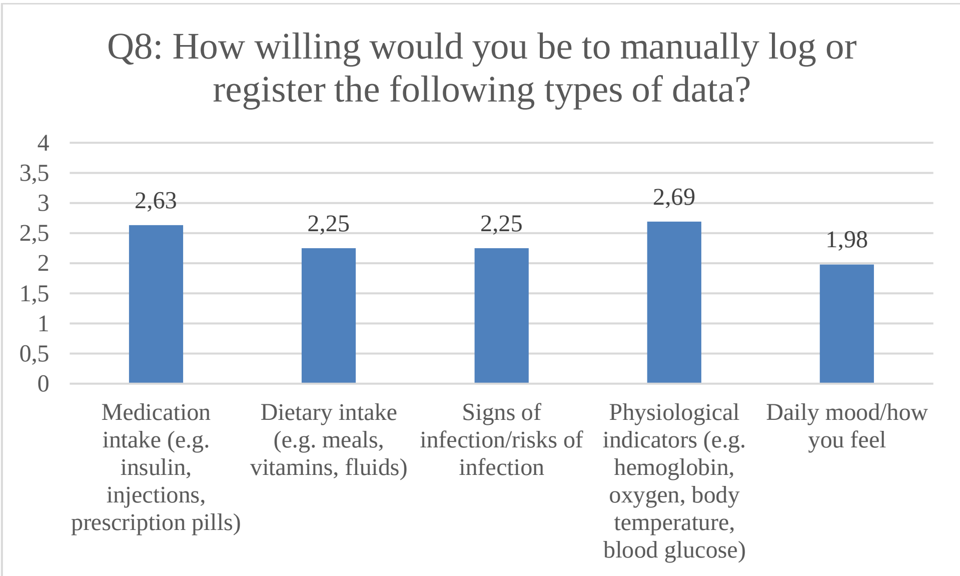

Figure 39: *Willingness of manual logging.*

9 Discussion

This chapter will discuss the findings of the thesis. Section 9.1 discusses the survey: motivation in mHealth and presents its contributions to smartwatch development. Then, the evaluation of the system in the context of the survey as well as the strengths and limitations of the system is presented. Then, an evaluation of the potential of standalone smartwatches for diabetes management.

9.1 The Survey motivation in mobile health

From the survey, we can see that the most important attributes of a disease management system are ease of use and simplicity. Specifically, Q3, Q5, and Q7 value simplicity and ease of use the most. Whereas Q2 value simplicity second, succeeding sensor accuracy. Unfortunately, none of the questions have unfolded what the participants find simplistic or “ease of use”. For instance, does simplicity refer to a simple design (UI, navigation, universal design, logical composition, input methods) or a simple system (number of functionalities, system integration methods, sync. options)?

However, in question 7: “If you are required to do manual logging (registration) in a mobile health app, how important are these criteria for you?”, the participants preferred simplicity over logic. This question addresses the physical interaction (touching the screen) with a given system, and in this case, simplicity is more important than the frequency of logging, i.e., if the complexity of the application increased with the benefit of less manual logging, the participants would rather prefer a simplistic application. From this, we can argue that time spent in an application or the need for user input should not be prioritized over system design (e.g., longer user stories). This statement is further reflected in Q5: mobile application functionalities. The attribute the participants find the least interesting are the features of mobile applications and, interestingly, score an entire point lower than simplicity.

Questions 2 and 4 may give further granularity to the participants’ preferences. Hence, question 2 addresses general features to have on a wearable device, whereas, question 4 addresses specific health-related features to have on a wearable device. When we compare the results from these questions, we see that people value sensor accuracy the most (Q2), and physical activity tracking the least (Q4). However, sensor accuracy and physical activity tracking are highly connected, and therefore, it would be expected for these two features to have a higher correlation.

On the other hand, the feature the participants saw as the most important and agreed on the most (compared with other questions) was disease management (Q4). Disease management is the most important health feature to have on a wearable device.

Question 3 is the only question addressing motivation and wearable devices. As presented, simplicity is the feature that motivates the prolonged use of a wearable device. However, personalization is the only option that motivates the participants for extended use when evaluating the other options. Group-based features gained through aggregated data and social media are not in the participants' interest. Furthermore, 13% do not know what motivates the extended use of wearable devices.

Based on the survey, it is hard to precisely determine what motivates users to use wearables for longer. However, from the widespread answers from Q3, plays ease of use, non-disruptiveness, and relevant feedback an essential role. Furthermore, simplicity plays a considerable role in the development of health applications for wearable devices.

In summary, the survey’s primary focus is motivation in mobile health. The responses from the survey have been an essential factor when designing and choosing the features presented in the design. Disease management is the most crucial feature (succeeding physical activity monitoring) the participants sought in a wearable device (**Figure 35**). Furthermore, the first research problem (RP1) can be concluded with simplicity – simplicity is an essential feature people with diabetes find in a wearable application for disease management.

9.2 Review of the state of the art

From the literature review, many papers show promising results in terms of acceptance among wearable devices. However, only four [45, 46, 48, 49] papers document a smartwatch application for disease management, and only two [36, 37] of them document the platform's effect on HbA1c, which were positive.

Firstly, none of the four papers document standalone functionalities among the smartwatch application. To the author's knowledge, there is no scientific documentation of a standalone smartwatch application for diabetes management. However, this is expected due to the relatively immature introduction of standalone architectures on smartwatches and the papers' time of writing. Although, one of the users of Pektas et al. [46] system pointed out a need for NightScout integration resulting in direct communication between the smartwatch and the glucose data stored in the cloud. Inherently, since none of the systems documents standalone architectures, privacy concerns regarding direct smartwatch communication with the internet (and cloud storage) were not documented.

Nevertheless, Preuveneers et al. [52] proposed an algorithm for end-device encryption on wearable devices. Their use case was a given a system that collects health data—however, their solution was unsatisfactory slow.

Among the four papers that documented disease management, only two were relevant in terms of disease management on the watch. Pektas et al. [46] presented a wathcface, where the user may log relevant diabetes metrics, namely carbohydrates, insulin, and blood glucose. Osborn et al. [49] also presented a watchface where users may log and display relevant data. However, specific features such as the type of data or design were not presented. Both applications work functionally as a remote for the companion application on the smartphone.

An interesting observation, based on the described systems, is that the applications are bound to the watchface. As a result, if a user of the application has another watchface due to personal preference, the application would be unavailable. For example, the users of the system of Seiderer et al. [47] appreciated the discrete nature of logging nutrition on the smartwatch. If the watchscreen “flashes” diabetes management, could users see the need to have a natural screen on the watch. The systems presented in the related work do not accommodate access to the application without a watchface. If the user does not want to use the watchface, they do not have an application to use.

9.3 Evaluation of the system

We can see from the design and implementation that there is possible to implement the standalone smartwatch application. However, is such a completely standalone design viable to have on a smartwatch, and will it motivate people to manage diabetes? In this section, the presented system will be evaluated according to the survey and the literature review.

9.3.1 Design

Material theming was used to develop a simplistic familiar design and has provided a design with clear use of colors and continuity. The buttons introduced in the design follow common design familiarities. For example, adding a new goal or adding a new log entry uses the same "add-icon". Furthermore, the colors presented in the system have been evaluated according to the WCAG (Web Content Accessibility Guidelines) [57]. WCAG defines the ruleset for universal design on web platforms, describing international standards for accessibility for people independently of physical capabilities. The lowest color contrast in the system is the yellow chips within the goal listing view (see **Figure 14**). The color ratio between these two colors upholds the minimum criterion of WCAG's AA certificate. The criterion for AA is to provide enough contrast between the background and font so that people with moderately low vision can read it [58]. This is important because people with diabetes can find the application easier to use since the colors of the system provide a clear contrast, as one of the complications of diabetes is reduced eyesight.

9.3.2 Navigation

Another aspect to investigate is navigation. Navigation within the application is almost as important as the system's features. Unintuitive navigation will cause the user to spend more time figuring out where the destination is, rather than using the desired functionality. To further emphasize the importance of navigation, performing a logging action requires the user to scroll the bezel on the watch and tap continue. Navigating to the view that presents the logging feature is as many interactive steps (tapping and scrolling) as the logging itself. The user would spend more time navigating to the feature if the navigation path were slightly increased.

Table 3 presents the navigation paths within the application. A path is defined as the relation between two views and is measured in screen taps. For example, navigating from the main screen to the nutrition screen is equal to one path. The most profound paths the user can traverse in the application are all related to setting a goal target. Logging nutrition has the longest path due to the entry of two separate units: time and amount. However, if the user uses the predefined measures, the user may reduce the path down to 3 taps.

Reducing the logging path has been an essential aspect of the design process. In order to reduce the need for additional taps, a design choice was made not to include timestamp adjustment on the glucose entry screen and instead use the current time. In comparison, the nutrition logging includes timestamps entry as it might be more relevant to do after eating or throughout the day.

Table 3: *The length of navigation within the application.*

Path	Deepest navigation path	Path for logging/updating	Path for setting goals
Nutrition	5	3/4	3
Monitoring	3	2	3
Physical Activity	3	0	3
Goal setting	4 (including scrolling)	2	4
Settings	2	-	-

In the presented table, it is assumed that the user enters from the application's tile. The tile reduces the taps the user needs to manage the application drastically. Opening the application from the app-drawer costs time: first, open the *app drawer* (one swipe), then find the application among the other applications on the watch (multiple swipes), and lastly, open the application (one tap). These steps are omitted with a tile or a watchface that provides a shortcut to the application.

Table 4 presents the nutrition logging process of the systems mentioned in the thesis. The presented values are collected by testing the applications (* have not been tested and are based on documented design). Furthermore, the taps presented do not include finding and opening the application or using the keyboard.

We see that the designated diabetes applications Diabetes Diary and Diabetes:M have the least taps to log nutrition, however, they do not include an adjustable timestamp.

Table 4: *Number of taps needed to log nutrition.*

	Diabetes diary*	Diabetes:M*	**Presented system**	LifeSum	My Fitness Pal	Seiderer et al. smart scale*
Taps	3	3	**4**	4	5	5
Shortcut (- 3 taps)	Yes	Yes	**Yes**	No	No	-
Input Method	Keyboard	Manual (In/decrement 50, 10, 1)	**Manual (Bezel), predefined (s/m/l)**	Predefined (s/m/l)	Keyboard	Keyboard
Time annotation	-	-	**Timestamp**	Breakfast, Lunch, Dinner. Snack	Breakfast, Lunch, Dinner, Snack	-
Nutrition	Carb	Carb, Kcal, proteins, fat	**Kcal, carb**	Food types	Kcal, water	Complete meals

Interestingly, the different input methods are spread. From personal experience, the keyboard methods are the most challenging to use, and the predefined values are the most straightforward (fewer taps). However, predefined values do not give any granularity. Precise values are important when logging carbohydrates in relation to medication due to the relationship between medication and carbohydrates. The bezel, on the other hand, gives the opportunity to do both of which. In addition, the values can increase fast depending on the wheel's acceleration while giving fine granularity. As a result, the input process with the bezel is fast and exact.

Furthermore, there is a wide variety of nutrition types that can be logged. Diabetes:M supports the most finetuned nutrition variants, however, it is one navigation step between each type (navigating a carousel with onscreen arrows). As a result, it will require three additional taps if the user wants to log calories in Diabetes:M. In the proposed system, the user may choose between the nutrition types from the same screen (currently, there are only carbohydrates and calories, but there is enough on-screen space for fat and protein).

In conclusion, the proposed design has a faster logging screen due to its effective use of bezel while keeping the logging degree at fine granularity. The presented system lacks nutrition options and should evaluate if the timestamp option is worth one tap.

9.3.3 Glanceability

In this thesis, a decision was made to implement a tile rather than a watchface. There are systems that already have implemented watchfaces and have not focused on the rest of the capabilities of smartwatches – their entire application resides within a watchface.

Due to the missing watchface, the application has a reduced glanceability compared to competing systems. However, the implemented tile aims to view the state of the management (replacing the watchface). A tile comes with the cost of not glancing into the application's data immediately but within a swipe.

Recall that the implemented tile presents the progress of goals (e.g., the percentage of TIR and the percentage of calories eaten today) in a progress bar. Feedback in the alpha testing (fellow students and supervisors) was that the tile should present numbers, specifically, the latest blood glucose value. The idea of displaying percentages was to make the user more aware of their goals. However, displaying a discrete value instead of either the monitoring icon or the percentage would reduce the need for a tap, increasing the simplicity and glanceability. In a future implementation, numbers should be accommodated.

9.3.4 Motivation

When the “in the background” goal feature was introduced, a decision was made to introduce goal-setting as its own category. There might be other activities the user wants to monitor and manage outside of the specific units within monitoring (mmol/L, kcal, steps); in other words, a more personalized experience. The goal category consists of two types of goals: automatic and manual. Automatic goals can be tracked by using the other features in the application. For example, the goal “monitor fasting glucose each morning of the week” will increment if the user logs blood glucose in the early hours and has not logged nutrition yet. In comparison, manual goals must be incremented within the goals-setting view. Either way, the goal feature aims to encourage and motivate the user to perform more activities.

An appealing design is motivating. This is not a specific feature of the system, however, an aspect that may help others to know what affects the acceptance of the system. For example, Pektas et al. [46] had users that were motivated by the forthcoming colors in the system.

There were other motivational features considered when designing the system. For example, Derek et al. [59] used a social platform (on a mobile device) and a pedometer (step counting device) to reduce sedentary behavior among coworkers. They found that social media can provoke forth motivation. However, the presented questionnaire shows that a social media integration would not encourage users to wear wearable devices longer. We acknowledge that using a wearable device is not equal to performing diabetes activities. However, if the user does not want to wear the device, they cannot perform self-management activities.

Another motivational feature that can be integrated into the smartwatch application is nudging. Nudging is a concept of a choice architecture that aims to alter people's behavior without limiting the options or economic interests of the person targeted by the nudge [60]. For example, a push notification stating, “Remember, taking the stairs over the elevator burns ten calories more than you would with taking the elevator.”. If this nudge is timely relevant, the user might take the stairs over the elevator. Furthermore, these techniques reside not only on notifications; placing the toothpaste in a visible area in the bathroom to remember to brush the teeth can also be interpreted as a nudge. However, in the case of a smartwatch application, such a system would highly rely on push notifications.

Such notifications must rely on a system generating these notifications. If the watch were to generate them, it would need further processing, increasing the complexity of the application. Alternatively, the notifications could be pushed from the external service, however, this would require the user to be authenticated. The main reason this motivational aspect has not been evaluated further is that the participants in the survey value non-disruptiveness. Such a solution would require push notifications and increase the disruptiveness.

Gamification could be implemented on the watch. Gamification is a technique or strategy to make users more engaged in a product using game elements. However, gamification often relies on a goal-setting system. Therefore, testing and implementing goal-setting before implementing the game aspect is logical. The specific gamification features could be unlocking achievements or automatic goal adjustment according to previous performance.

The goal-setting feature focus on intrinsic motivation. For example, the user sets the goals and carries them out themselves. However, in a standalone smartwatch application could introduce extrinsic motivation through supervision. Caretakers, parents, or relatives can follow the progress of a person. This is an already known feature such as Dexcom Follow[24]. However, to the author's knowledge, is this a feature not highlighted on smartwatches.

The following is stated to conclude the second research problem (RP2), "What motivational techniques can be implemented on a diabetes smartwatch application?". The *standard* features (collection of diabetes metrics and physical activity) have documented positive outcomes on HbA1c. Furthermore, based on the survey, smartwatch devices' simplicity and non-disruptiveness are essential features. Therefore, motivational features should not affect how the users interact with this system on a large scale. Goal-setting and caretaker control are two motivational features that have a documented effect on motivation and can mostly rely on the already effective systems and do not increase the system's complexity.

9.4 Standalone

The third research problem addresses how to utilize standalone architecture when designing a diabetes application. Throughout this work, it is found that the standalone architecture of smartwatches for diabetes management is absent. Only one project was found on GitHub (not included in the related work or literature review, as it is a private project), namely

[24] https://www.dexcom.com/dexcom-follow

NightWear[25]. This project connects the smartwatch to the NightScout server, displaying the latest glucose data.

Firstly, this thesis presents a parental control feature. NightWear could arguably be used as a parental control device – have a person with diabetes log into the server on the watch and then give the watch to another person. However, this is not the system design of NigthWear. Features such as multiple users or end-user administration are not addressed. A standalone architecture will be able to reach out to people who do not have or actively use a smartphone, such as young children or the elderly. Secondly, when blood glucose data is transmittable from the CGM and directly to the smartwatch, the need to have a smartphone is reduced. The smartwatch can display, alert and collect the data needed for diabetes management and then send it to the desired storage location. As a result, the availability of blood glucose data is higher than ever – users can integrate blood glucose into systems where the smartphone is not available such as jogging.

However, there are challenges with the standalone architecture that must be considered.

Standalone smartwatch applications cannot rely on actions that reside on the smartphone application. Hence, a standalone application should be able to be downloaded on the smartwatch without any assumptions regarding the smartphone. Actions such as configuration menus or authentication must be adjustable in the smartwatch application. However, non-standalone applications handle this on the companion application on the smartphone.

The goal-setting feature could benefit smartphone management. Firstly, setting a goal target is out of context; real-life actions and actions on the smartwatch are not related. For example, when the user logs nutrition in the smartwatch application, will it be likely to assume that they ate recently. However, when setting goal targets, the user likely manages the application for the first time and configures more than one goal. This will be a more extended task (than logging), and the user must navigate in and out between the categories in order to set each category's goal. If the screen was larger (as on a smartphone), the goals could be seen/set from a single screen, resulting in a faster configuration.

---

[25] https://github.com/rahim/nightwear

Secondly, the input methods on a smartwatch are limited to the screen size. The suggested goal-setting view relies on predefined goals, such as “monitor fasting blood glucose every day”. If the goal-setting view were to allow fully customizable goals, the application must support input methods from the user. Writing text on the smartwatch is not an immense challenge, however, such a feature would not be simplistic and rather time-consuming. A solution for resolving configuration steps is to handle them on the companion application.

In conclusion to the third research problem, “*how to best utilize the properties of standalone smartwatches when designing a diabetes application?”*, the following is stated. Today, smartwatch applications can implement a semi-standalone architecture. Therefore, some data sources do not need to rely on a smartphone. Furthermore, implementing a standalone architecture can increase the availability and userbase of the products. However, completely standalone applications can benefit from the extended one-time connection of smartphones to configure the application.

9.5 Hardware and Security

Since the system has not been tested, it is challenging to determine the specific hardware and security limitations. However, the nature of the system has some aspects that are worth mentioning.

Firstly, the system's impact on battery life must be addressed. A measure to increase the battery life was to have black as the primary background. Furthermore, as presented in the implementation chapter, the sensor data is from a repository. The system does not directly perform the data collection of physical activity. As a result, background activity is reduced, which in turn increases battery life.

A design choice that will affect battery life is notifications. Notifications can be generated from within the smartwatch or delivered by a backend. If the system were to use internal notifications, the system must frequently check for delivery logic, such as evaluating the time since the last log entry. Again, this will be a background task, costing battery life. However, receiving internet packets is done by the smartwatch architecture regardless of application. In turn, this comes with an additional communication overhead. The user must be authenticated, and data must be synced.

The proposed system collects health data. Firstly, the system must address what data is collected and where the data is stored. Furthermore, the user must be presented with a declaration of how to get and delete their data.

Moreover, security measures must be made if the user chooses to use the system with synchronization and the overseeing functionalities. Firstly, the system must comply with the data regulations of the users.

The proposed design has the potential to collect a lot of user data. Together with the person with diabetes and the health personnel, this data can be used to evaluate management patterns, such as correct eating habits, or look at the effects of medication. However, the current design has not addressed data export. Data export could be integrated into the Health Platform (the step provider used in the implementation). Health Platform can sync health data between devices without including a cloud service. Furthermore, the Health Platform supports the integration of nutrition, blood glucose, and physical activity. Therefore, the implemented system could use the Health Platform more extensively for storing diabetes-related metrics and then only use a local database to track goals. This solution would have the user access their data, even if they choose to use the platform in offline mode.

9.6 Future work

Although the system has gotten feedback from fellow students and supervisors (alpha testing), the system has not been tested on people with diabetes. This is an important step that should be done before the system's functionalities or design is adjusted. The user-centric design process is essential when working with health systems, especially when the system's developers do not have the condition themselves. Furthermore, to test the system, the implementation must be complete. Therefore, implementing the remaining views is of high priority.

The proposed system is still in the idea phase. In order to realize the proposed system must some aspects be further investigated. The proposed system assumes communication between the CGM and the smartwatch, however, this is not possible with the current state of the art. When the next generation of CGMs is released, the possibility of obtaining glucose data directly from the CGM must be investigated. A possible solution would be that the manufacturers implement a standalone application. This application pushes the data to their cloud service, from where third-party applications can collect it (e.g., through Dexcom's API).

In the proposed system, it is suggested that the caretaker may connect to another device's data. However, how such architecture would look like is not thoroughly documented. A secure connection to an external cloud server is assumed, however, when and how often such data is synchronized must be evaluated. How such a feature would affect battery life and be reliable in terms of connectivity must be answered in future work.

Furthermore, a future implementation should investigate the need for a companion application. The configuration process in the presented system may be too tedious to do on a smartwatch. Implementing a companion application for one-time configuration might show a higher acceptance of the product. Again, this may be apparent after user-testing.

10 Conclusion

This thesis presents a state-of-the-art of smartwatch used in diabetes management. It is presented that none of the presented work focus on either the standalone capabilities of smartwatches or the motivational aspect of diabetes. Then a system design of a smartwatch system is described focusing on the motivational aspect of diabetes management, including features and an appealing UI. The system design is composed of the results from a survey exploring the needs of people with diabetes and existing solution – where it was concluded that goal-setting and simplicity are the most profound aspects to integrate into a smartwatch application for diabetes management. The proposed system is evaluated based on the analysis of the presented survey and possible motivational features to apply to a smartwatch management system.

In conclusion to the main research problem, "*How can a standalone smartwatch application motivate users to perform self-management activities regarding diabetes?"*, it is concluded that goalsetting in conjunction with common management activities might elevate the motivation of people with diabetes. Furthermore, with the use of a standalone architecture, can the extrinsic motivation be increased by having others directly (not through the smartphone) connect to the smartwatch and monitor the users' data.

Future work presents the continued discovery of directly communicating with the CGM and testing the system on people with diabetes.

Bibliography

[1] WHO. Diabetes: World Health Organisation; [updated 2022. Available from: https://www.who.int/news-room/fact-sheets/detail/diabetes Accessed: 23.03.2022]

[2] IDF. Diabetes Facts & Figures: International Diabetes Federation; [updated 09.12.2021. Available from: https://idf.org/aboutdiabetes/what-is-diabetes/facts-figures.html Accessed: 23.03.2022]

[3] Shrivastava SR, Shrivastava PS, Ramasamy J. Role of self-care in management of diabetes mellitus. Journal of Diabetes & Metabolic Disorders. 2013;12(1):14.

[4] Karter AJ, Ferrara A, Darbinian JA, Ackerson LM, Selby JV. Self-monitoring of blood glucose: language and financial barriers in a managed care population with diabetes. Diabetes Care. 2000;23(4):477-83.

[5] Russell LB, Suh DC, Safford MA. Time requirements for diabetes self-management: too much for many? J Fam Pract. 2005;54(1):52-6.

[6] Clarke SF, Foster JR. A history of blood glucose meters and their role in self-monitoring of diabetes mellitus. Br J Biomed Sci. 2012;69(2):83-93.

[7] Lewis D. History and Perspective on DIY Closed Looping. Journal of Diabetes Science and Technology. 2019;13(4):790-3.

[8] Litchman ML, Lewis D, Kelly LA, Gee PM. Twitter Analysis of #OpenAPS DIY Artificial Pancreas Technology Use Suggests Improved A1C and Quality of Life. Journal of diabetes science and technology. 2019;13(2):164-70.

[9] What is the Nightscout project? : The Nightscout Project; [updated 2022. Available from: http://www.nightscout.info/ Accessed: 23.03.2022]

[10] Lewis D. OPENAPS.ORG #WeAreNotWaiting to reduce the burden of Type 1 diabetes: OpenAPS community; [updated 2022. Available from: https://openaps.org/ Accessed: 23.03.2022]

[11] Reeder B, David A. Health at hand: A systematic review of smart watch uses for health and wellness. J Biomed Inform. 2016;63:269-76.

[12] Mück JE, Ünal B, Butt H, Yetisen AK. Market and Patent Analyses of Wearables in Medicine. Trends in Biotechnology. 2019;37(6):563-6.

[13] Chen X, Chen W, Liu K, Chen C, Li L. A comparative study of smartphone and smartwatch apps. Proceedings of the 36th Annual ACM Symposium on Applied Computing; Virtual Event, Republic of Korea: Association for Computing Machinery; 2021. p. 1484–93.

[14] Independent versus dependent Wear OS apps: Android development documentation; [updated 17.11.2021. Available from:

https://developer.android.com/training/wearables/overlays/independent-vs-dependent Accessed: 10.01.2022]

[15] Klevstad B. How Smartwatches can work as the new personal digital assistant for people with diabetes [Unpublished work]. 2021.

[16] IDF. Hypoglycaemia, Care & Prevention: International Diabetes Federation; [updated 03.02.2020. Available from: https://www.idf.org/our-activities/care-prevention/hypoglycaemia.html Accessed: 23.03.2022]

[17] WHO. Noncommunicable diseases: World Health Organization; [updated 13.04.2021. Available from: https://www.who.int/news-room/fact-sheets/detail/noncommunicable-diseases Accessed: 23.03.2022]

[18] Blood Sugar Level Ranges: The global diabetes community; [updated 1.15.2019. Available from: https://www.diabetes.co.uk/diabetes_care/blood-sugar-level-ranges.html Accessed: 25.02.2022]

[19] Type 1 diabetes: Helse Norge; [updated 22.10.2020. Available from: https://www.helsenorge.no/sykdom/diabetes/diabetes-type-1 Accessed: 25.05.2022]

[20] Type 2 Diabetes: Helse Norge; [updated 22.10.2020. Available from: https://www.helsenorge.no/sykdom/diabetes/diabetes-type-2/#utredning-og-diagnose Accessed: 26.05.2022]

[21] Blodsukker og måling: Diabetes Forbundet; [updated 02.03.2022. Available from: https://www.diabetes.no/diabetes-type-1/behandling/blodsukker/ Accessed: 30.03.2022]

[22] ADA. Hypoglycemia (Low Blood sugar): American Diabetes Association; [updated 2022. Available from: https://www.diabetes.org/healthy-living/medication-treatments/blood-glucose-testing-and-control/hypoglycemia Accessed: 25.04]

[23] Ehrhardt NM, Chellappa M, Walker MS, Fonda SJ, Vigersky RA. The effect of real-time continuous glucose monitoring on glycemic control in patients with type 2 diabetes mellitus. Journal of diabetes science and technology. 2011;5(3):668-75.

[24] ADA. Hyperglycemia (High Blood Glucose): American Diabetes Association; [updated 2022. Available from: https://www.diabetes.org/healthy-living/medication-treatments/blood-glucose-testing-and-control/hyperglycemia Accessed: 25.04.2022]

[25] Mouri MI BM. Hyperglycemia [updated 28.04.2022. Available from: https://www.ncbi.nlm.nih.gov/books/NBK430900/ Accessed: 03.05.2022]

[26] HbA1c, langtidsblodsukker: Norsk Helseinformatikk; [updated 2022. Available from: https://nhi.no/sykdommer/hormoner-og-naring/diabetes-generelt/hba1c/ Accessed: 25.04.2022]

[27] ADA. CGM & Time in Range: American Diabetes Association; [updated 2022. Available from: https://www.diabetes.org/tools-support/devices-technology/cgm-time-in-range Accessed: 25.04.2022]

[28] Time in Range - What is Time in Range? : The diaTribe Foundation; [updated April 8, 2022. Available from: https://diatribe.org/time-range Accessed: 27.04.2022]

[29] The Framework for Optimal Self-Management: Association of Diabetes Care & Education Specialists; [updated 2022. Available from: https://www.diabeteseducator.org/practice/practice-tools/app-resources/the-aade7-self-care-behaviors-the-framework-for-optimal-self-management Accessed: 25.04.2022]

[30] Prediabetes: Helsenorge; [updated 2022. Available from: https://www.helsenorge.no/sykdom/diabetes/prediabetes/ Accessed: 25.04.2022]

[31] Hallberg SJ, Gershuni VM, Hazbun TL, Athinarayanan SJ. Reversing Type 2 Diabetes: A Narrative Review of the Evidence. Nutrients. 2019;11(4):766.

[32] Safford MM, Russell L, Suh D-C, Roman S, Pogach L. How Much Time Do Patients with Diabetes Spend on Self-Care? The Journal of the American Board of Family Practice. 2005;18(4):262-70.

[33] Swanson V, Maltinsky W. Motivational and behaviour change approaches for improving diabetes management. Practical Diabetes. 2019;36(4):121-5.

[34] Sennett P. Understanding intrinsic and extrinsic motivation [updated 29.06.2021. Available from: https://www.rochester.edu/emerging-leaders/understanding-intrinsic-and-extrinsic-motivation/ Accessed: 25.04.2022]

[35] Williams GC, McGregor HA, Zeldman A, Freedman ZR, Deci EL. Testing a Self-Determination Theory Process Model for Promoting Glycemic Control Through Diabetes Self-Management. Health Psychology. 2004;23(1):58-66.

[36] Williams GC, Patrick H, Niemiec CP, Williams LK, Divine G, Lafata JE, et al. Reducing the health risks of diabetes: how self-determination theory may help improve medication adherence and quality of life. The Diabetes educator. 2009;35(3):484-92.

[37] Miller CK, Bauman J. Goal Setting: An Integral Component of Effective Diabetes Care. Current Diabetes Reports. 2014;14(8):509.

[38] Henriksen A, Haugen Mikalsen M, Woldaregay AZ, Muzny M, Hartvigsen G, Hopstock LA, et al. Using Fitness Trackers and Smartwatches to Measure Physical Activity in Research: Analysis of Consumer Wrist-Worn Wearables. J Med Internet Res. 2018;20(3):e110.

[39] Homayounfar M, Malekijoo A, Visuri A, Dobbins C, Peltonen E, Pinsky E, et al. Understanding Smartwatch Battery Utilization in the Wild. Sensors (Basel, Switzerland). 2020;20(13):3784.

[40] Woldaregay AZ, Henriksen A, Issom DZ, Pfuhl G, Sato K, Richard A, et al. User Expectations and Willingness to Share Self-Collected Health Data. Stud Health Technol Inform. 2020;270:894-8.

[41] Denning P, Comer D, Gries D, Mulder M, Tucker A, Turner J, et al. Computing as a discipline. Computer. 1989;22:63-70.

[42] Robertson JR, Suzanne.

. Volere Requirements Specification Template Edition 2012;16.

[43] Freepik. Diabetes Collection: Flaticon; [updated 2022. Available from: https://www.flaticon.com/authors/freepik Accessed: 05.02.2022]

[44] Diabetes:M Wear OS and Apple Watch: Sirma Medical Systems; [updated 2022. Available from: https://www.manula.com/manuals/sirma-medical-systems/diabetes-m-user-guide/mobile/en/topic/android-wear Accessed: 25.03.2022]

[45] Årsand E, Muzny M, Bradway M, Muzik J, Hartvigsen G. Performance of the First Combined Smartwatch and Smartphone Diabetes Diary Application Study. Journal of Diabetes Science and Technology. 2015;9(3):556-63.

[46] Pektas O, Koseoglu M, Muzny M, Hartvigsen G, Arsand E. Design of an Android Wear Smartwatch Application as a Wearable Interface to the Diabetes Diary Application. Academic Platform Journal of Engineering and Science. 2021;9:126-33.

[47] Seiderer A, Flutura S, André E. Development of a mobile multi-device nutrition logger. Proceedings of the 2nd ACM SIGCHI International Workshop on Multisensory Approaches to Human-Food Interaction; Glasgow, UK: Association for Computing Machinery; 2017. p. 5–12.

[48] Timurtas E, Inceer M, Mayo N, Karabacak N, Sertbas Y, Polat MG. Technology-based and supervised exercise interventions for individuals with type 2 diabetes: Randomized controlled trial. Prim Care Diabetes. 2022;16(1):49-56.

[49] Osborn CY, van Ginkel JR, Marrero DG, Rodbard D, Huddleston B, Dachis J. One Drop | Mobile on iPhone and Apple Watch: An Evaluation of HbA1c Improvement Associated With Tracking Self-Care. JMIR Mhealth Uhealth. 2017;5(11):e179.

[50] Hosseini V, Tabasi A. Smartwatch's Application Algorithm to Prevent Sudden Death and Brain Damage in Hypoglycemic Coma. J Diabetes Sci Technol. 2016;10(3):802-3.

[51] Karnati Y, Zapata R, McConnell MJ, Reddy RRK, Regalla V, Thakkar A, et al. ROAMM: A customizable and interactive smartwatch platform for patient-generated health data. 2021 Thirteenth International Conference on Contemporary Computing (IC3-2021); Noida, India: Association for Computing Machinery; 2021. p. 150–8.

[52] Preuveneers D, Joosen W. Privacy-enabled remote health monitoring applications for resource constrained wearable devices. Proceedings of the 31st Annual ACM Symposium on Applied Computing; Pisa, Italy: Association for Computing Machinery; 2016. p. 119–24.

[53] Banovic N, Brant C, Mankoff J, Dey A. ProactiveTasks: the short of mobile device use sessions. Proceedings of the 16th international conference on Human-computer

interaction with mobile devices & services; Toronto, ON, Canada: Association for Computing Machinery; 2014. p. 243–52.

[54] Liu X, Chen T, Qian F, Guo Z, Lin FX, Wang X, et al. Characterizing Smartwatch Usage in the Wild. Proceedings of the 15th Annual International Conference on Mobile Systems, Applications, and Services; Niagara Falls, New York, USA: Association for Computing Machinery; 2017. p. 385–98.

[55] Gouveia R, Karapanos E, Hassenzahl M. How do we engage with activity trackers? a longitudinal study of Habito. Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing; Osaka, Japan: Association for Computing Machinery; 2015. p. 1305–16.

[56] WHO. Physical activity: World Health Organization; [updated 26.11.2020. Available from: https://www.who.int/news-room/fact-sheets/detail/physical-activity Accessed: 20.02.2022]

[57] WWWC. How to Meet WCAG (Quick Reference): World Wide Web Consortium; [updated 2019. Available from: https://www.w3.org/WAI/WCAG21/quickref/ Accessed: 22.05.2022]

[58] WWWC. Contrast (Minimum) Understanding SC 1.4.3: World Wide Web Consortium; 2016 [Available from: https://www.w3.org/TR/UNDERSTANDING-WCAG20/visual-audio-contrast-contrast.html Accessed: 22.05.2022]

[59] Foster D, Linehan C, Kirman B, Lawson S, James G. Motivating physical activity at work: using persuasive social media for competitive step counting. Proceedings of the 14th International Academic MindTrek Conference: Envisioning Future Media Environments; Tampere, Finland: Association for Computing Machinery; 2010. p. 111–6.

[60] Thaler RH, Sunstein CR. Nudge: Improving decisions about health, wealth, and happiness. New Haven, CT, US: Yale University Press; 2008. x, 293-x, p.

Appendix

Motivation in mobile health

Questions included in the questionnaire. In addition, for every option where you can choose "other", there is a follow-up questions to "specify" if they want to.

Background questions

1) Please choose your most important health goal (single)
    a. Avoid sickness/maintain good health
    b. Improve my health
    c. Recover easily from disease/being cured
    d. Other
2) How well do you think you are achieving your health goals? (Single, 1-4 + don't know) 1 = Not well at all, 4 = very well
3) How important are these factors to help you achieve your health goals? (single, 1-4, +don't know)
   1 = Not important at all, 4 = very important

    a. Self-motivation
    b. Presence of advance/relevant technology
    c. Access to quality healthcare
    d. Access to trustworthy information
    e. Healthy behaviors (e.g. diet, physical activity, hydration)
    f. Ability to understand health related information
    g. Better self-knowledge
    h. Social & Life environment (e.g. positive friends, family, Spirituality, pollution, neighborhood, household type)
4) What do you think is your biggest challenge for achieving your health goals? (Single)
   a. Lack of motivation
    b. Limitation in current technology (e.g. medical devices/health apps/drugs)
    c. Lack of knowledge, experiences, or competency/skills
    d. Lack of time
    e. Lack of social support
    f. Lack of access to quality health care
    g. Other
5) Have you ever used a wearable device for collecting activity or other health data? (Single. Yes, No)

Wearables and sensors

6) Which of these technologies for health tracking do you regularly use? (Multiple choice)

a. Sensors integrated in smartphone (e.g. geolocation, accelerometer, stress tracker)
   Physical activity tracker

b. Mobile health apps
c. Health specific measurement (e.g. glucometer/CGM)
d. Do not used any sensors or wearables
e. Other

7) How important are these features for you when choosing a wearable device? (Single, 1-4, + don't know)
   1 = Not important at all, 4 = very important
   a. Sensor accuracy and range of values
   b. Access to several types of data (multiple sensors)
   c. Notifications from mobile phone (e.g. detection of early signs of disease)
   d. Easy to use and quality of associated mobile app
   e. Known or specific brand/price
   f. Ergonomic and design (e.g. physical design/battery consumption)
   g. Other features
8) Which features would motivate you most to use a wearable sensor longer?
   a. Relevant personalized feedback
   b. Access to aggregated summarize data on the population level
   c. Ease of use/non-disruptive
   d. Social integration (e.g. Facebook, Run Keeper)
   e. Don't know
   f. Other
9) How important are these specific health related features for you when choosing a wearable device? (Single, 1-4, + don't know)
   1 = Not important at all, 4 = very important
   a. Physical activity tracking (e.g. exercise features, heart rate/pulse tracking, fatigue, step counting, sleep tracking)
   b. Manage disease (e.g. blood glucose, pulse, oxygen, hemoglobin)
   c. Predicting and preventing symptom and health deterioration
   d. Alerts when risky behavior/when approaching limits

Mobile Apps

10) How important are these features when choosing a mobile health app? (Singe, 1-4, + I don't know)
   1 = Not important at all, 4 = very important

   a. Simplicity/usability
   b. Functionality/features
   c. Price
   d. Trust/security/privacy
   e. Personalization (tailored features)

11) How do you decide if a mobile app is trustworthy? (Multiple choice)
    a. In terms of security, accuracy, and usefulness, etc.
    b. Product specification/provider reputation
    c. Personal experience or other people's experiences
    d. Certificate by independent organization/expert judgement
    e. App rating in AppStore/GooglePlay
    f. Other

Logging

12) Which kind of data logging (registration) do you prefer? (Single, 1-4, + I don't know)
    a. Automatic
    b. Manual entry
    c. Semi-automatic (both is ok)
    d. Don't know
13) What motivates you to collect health data? (Choose 3)
    a. Review and track progress/trend and symptoms (Know more about my health condition)
    b. Share with others /show to my doctor (GP)
    c. Compare/compete with others
    d. Manage health and learn from previous patterns of behaviors
    e. Receive personalized health feedback
    f. Other
14) If you are required to do manual logging (registration) in a health mobile app, how important are these criteria for you? (Single, 1-4, +I don't know)
    1 = Not important at all, 4 = very important

    a. Easiness and simplicity
    b. Time required
    c. Frequency of logging
15) How willing would you be to manually log or register the following types of data? (Single, 1-4, +I don't know)
    1 = Not willing at all, 4 = very willing

    a. Medication intake (e.g. insulin, injections, prescription pills)
    b. Dietary intake (e.g. meals, vitamins, fluids)
    c. Signs of infection/risks of infection
    d. Physiological indicators (e.g. hemoglobin, oxygen, body temperature, blood glucose)
    e. Daily mood/how you feel

Data sharing and data integration

16) Which of these data types would you be willing to share anonymously in a research project?
    (Multiple choice)

a. Medication intake and treatment (e.g. insulin, injections, prescription pills)
b. Lifestyle/dietary intake (e.g. meals, vitamins, fluids, healthy habits, alcohol, cigarettes)
c. Signs of infection
d. Physiological indicators (e.g. hemoglobin, oxygen, body temperature, blood glucose, blood pressure)
e. Daily mood/how you feel
f. Physical activity/exercises data (e.g. weight, steps, respiration, pulse, type, duration, speed, date, time)
g. Geographical location (GPS)
h. Sleep duration
i. Weight
j. Social environment (e.g. household size, transportation habits, risks of infection) k. None of these

17) How concerned are you about these elements when sharing your health data? (Single, 1-4, +I don't know)

1 = Not concerned at all, 4 = very concerned

a. Lack of trust (e.g. confidentiality/privacy/security)
b. Who owns the data
c. Where is the data stored/service availability
d. Transparency of health data usage by 3rd parties

18) How much would these features motivate you to share your health data for research? (Single, 1-4, +I don't know)

1 = Not motivated at all, 4 = very motivated

a. Anonymity
b. Guarantee of appropriate use of the data/trustworthiness
c. Must be useful for myself/family
d. Useful for the public/other patients

19) What do you expect in return by sharing your health data? (Multiple choice) Personalized

Feedback/Notification

a. Integrated view (Data analysis/aggregated results/trends)
b. Comparing status with others
c. Decision support
d. Don't expect anything

Social media and entertainment factors

20) Have you ever accessed an online health group/community on social media? (Yes, No, Don't know)
21) What was your main reason for using social media for health purposes? (Single)

a. Sharing experiences

b. Get useful information/tips and tricks
c. Get knowledge inaccessible other places
d. Ability to share things that others don't understand,
e. Don't know
f. Others

22) What do you consider the main drawbacks with accessing social media health groups/communities for health purposes? (Multiple choice)
   a. Bad quality information (accuracy, reliability/credibility, trustworthiness, bias)
   b. Privacy/information leakage/public exposure/voyeurism
   c. Lack of relevance/coherence
   d. Lack of structure (quality information is hard to find)
   e. Don't know
   f. Others
23) How important are these (social media) features for you when used in a mobile health app? (Single, 1-4, +I don't know)
   1 = Not important at all, 4 = very important

   a. Clean user interface
   b. Anonymous sharing of data
   c. User specified sharing of fitness and health data (not anonymously)
   d. Chatting and discussions
   e. Entertainment (e.g. fun facts)
24) How much would these characteristics motivate you to use social media and entertainment in your self-care? (Single, 1-4, +I don't know)
   1 = Not motivating at all, 4 = very motivating

   a. Sharing information and experiences
   b. Structured information
   c. Trustworthy information
   d. Social support
   e. Compare/compete with others/game like features (Gamification) 25) What is your age?
26) What is your gender? (male, female, other)
27) What is your country of origin? (open)
28) Do you have a chronic disease? (Single)
   a. Yes, Diabetes (Type 1)
   b. Yes, Diabetes (Type 2/Other)
   c. Yes, Sickle-cell disease (SS, S-Beta)
   d. Yes, Sickle-cell disease (SC/other)
   e. Yes, other chronic disease
   f. No
   g. Do not want to answer

29) Are you a care-taker of someone with a chronic disease? (single)
   a. Yes, Diabetes (Type 1)
   b. Yes, Diabetes (Type 2/Other)
   c. Yes, Sickle-cell disease (SS, S-Beta)
   d. Yes, Sickle-cell disease (SC/other)
   e. Yes, other chronic disease
   f. No
   g. Do not want to answer

<u>Optional questions</u>

30) With whom do you prefer to discuss your health issues? (single)
   a. Health providers (GP, specialist, physiotherapist)
   b. Other patients
   c. Family and/or friends
   d. Others
31) Do you own a smart phone (Yes, No, Don't know)
32) There are many mobile health apps available, do you feel that you would be able to choose the correct one for you? (single)
   a. Yes
   b. No, I don't think I have enough knowledge
   c. No, I am not aware of available choices
33) When you want to download a mobile health app, how do you choose which one to download?
   a. Suggestion from family/friends/other patients
   b. Suggestion from health professionals
   c. Supplier advertisement/health magazine/social media
   d. Internet search (e.g. Google, Bing)
   e. Experimentation (trying things yourself, participation in research)
   f. Other
34) Do you have any previous experience with sharing your health data to the cloud? (Yes, No, Don't know)
35) Have you ever used mobile apps that integrates health data with wearables and sensors? If yes, what type of health apps have you used?
   a. Health tracking (e.g. Apple Health, Google Fit, Samsung S health)
   b. Disease management (e.g. Diabetes diary/Drepacare)
   c. Running(training) apps (e.g. Strava, Runtastic)
   d. Have not used any of these
   e. Others
36) Where do you prefer looking for health information? (single)
   a. Healthcare professionals
   b. Family/friends/patients/patient organizations
   c. Online search (e.g. Google, WebMD, Facebook, Doctissimo)
   d. Other

37) How frequent should the following types of feedback you receive be? (Immediate, Hourly, Daily, Weekly, Monthly, Never, Don't know)

a. General information (health status update)
b. Suggestion for changing current behavior
c. Urgent health information/immediate and crucial problem
d. Input request (inputting data)

38) How important are these criteria for you to agree to install an application that collects and shares data from your wearable? (Single, 1-4, +I don't know)
1 = Not important at all, 4 = very important

a. Automatic setup and easy to understand/use
b. Customizable feedback
c. Automatic data collection
d. Non-disturbing
e. Tailored data analysis and functionalities