UiT The Arctic University of Norway

Faculty of Science and Technology
Department of Computer Science

Continuous and automated data collection in migraine research

Extending the data collection capabilities of the Empatica E4

—

Daniel Ursin

INF-3971 Master's Thesis in Health Technology - June 2023

“Migraine certainly involves the brain” – Goadsby et al, 2015

Acknowledgements

This project represents the culmination of more than 7 years of work and life adjustments, achievements that would not have been possible without the help, assistance, and support from many good people.

Firstly, want to thank my main supervisor André Henriksen both for providing invaluable support through these last 5 years and for introducing me to the field of informatics in the first place. This extends to my second supervisor, Erlend Hoftun Farbu, for introducing me to the topic of this thesis. I would also like to thank Gunnar Hartvigsen and the rest of the Health Informatics and Technology group for actively including us students in the research community. And for my fellow students, thank you for listening to my rants, especially Dorthe, Markus, Neethan, and Sigurd.

Finally, a very special thanks goes out to my old friend Jonas Erstad for the numerous coffee breaks and for the interesting (and sometimes heated) discussions.

Daniel Ursin – Tromsø, June 2023

Abstract

Migraine is a recurrent headache disorder that afflicts significant portions of the global population. There is no current cure and migraines are mainly managed through symptomatic medical treatments and manual biofeedback routines. Automated data collection and prediction of migraine attacks through machine learning could be viable approaches for helping migraineurs and for reducing the impact of migraines, both on a societal and an individual level. However, machine learning approaches require access to large amounts of high-quality real-time data for facilitating prompt and reliable prediction under everyday conditions and within useful timeframes. The *Empatica E4* is an unobtrusive wearable sensor device that can satisfy these data collection needs, although not without flaws and shortcomings. Several studies have reported issues with E4 data collection, most regarding participant involvement and the logistical aspects of the collection process. On top of this, the native systems provided by Empatica for storing, retrieving, and utilizing collected data do not properly facilitate real-time data analysis or machine learning approaches.

This project creates a flexible data collection solution based on the E4 for facilitating real-time prediction of migraine attacks. It incorporates features and elements for increasing user involvement and for maximizing the data collection potential of the E4. Additionally, the solution is integrated with the *mSpider* data storage platform, facilitating reliable and flexible data storage and retrieval options.

The prototype system was tested on three potential end-users under everyday conditions over the course of 20 days. After the data collection period, each user attended a semi-structured interview. Testing and interview results show that the data collection capabilities of the prototype system are on-par with other similar systems, it offers stable data collection under everyday conditions, and it can store data in the mSpider system. However, the added features for increasing participant involvement had little discernible effect on the data collection process or the amount of collected data. This was probably caused by the low intensity of the added features or the short duration of the testing period. Additionally, the testing process found that the high technical proficiency requirements and the necessary daily maintenance of the E4 makes it unsuited for continuous migraine treatment purposes, although it is a good tool for migraine research. Future prototype iterations should increase the intensity of the participant involvement features and greatly increase the length of testing periods.

Table of contents

Figures and tables

Acronyms and abbreviations

API	Application Programming Interface
IBI	Inter-Beat intervals
BLE	Bluetooth Low Energy
SDK	Software Development Kit
PPG	Photoplethysmogram (Optical detection of blood volume)
ML	Machine Learning
HTTPS	HyperText Transfer Protocol Secure
JSON	JavaScript Object Notation
UI	User Interface
GUI	Graphical User Interface

1 Introduction

1.1 Background

Problems with migraine

Migraines are episodes of moderate to severe headaches that can significantly reduce the quality of life for individuals. According to statistics, more than 15% of the global population are experiencing migraine pain episodes (attacks) at least once per year [1], and approximately 4.5% of the global population suffers from attacks more than 15 times per 30 days [2]. On top of the burden of individuals, the combined societal costs of migraine in Norway were estimated to 58B NOK in 2020 [3]. Despite high prevalence and substantial societal impact, the direct causes of migraines are not known. Neither is there a cure, and pharmaceutical treatment of primary symptoms like nausea and head pain is the most common medical approach to the illness. Many migraineurs respond well to *triptans*, a family of drugs that reduces pain by directly affecting the physiological and neurological changes in the brain during migraine attacks [4]. Triptans are most effective if they are taken early in the pain period, as they are better at preempting the physiological changes than they are at reverting them. Activation time of triptans is normally less than 60 minutes, indicating that taking them right before the start of a migraine attack will have the best possible effect [4].

Migraine management

Biofeedback routines are important tools for migraineurs. Statistics and the patients' own mapping of their illness have shown correlations between migraine attacks and physiological states, potential trigger factors, and genetics [5]. This makes illness diaries and self-management valuable tools for migraineurs to recognize and avoid situations that increase the likelihood of migraine attacks, helping them to retain fulfilling lifestyles and possibly preempt pain episodes. Current biofeedback cycles for migraine consist of recording data on attacks, daily routines, and possible trigger factors, which are then utilized to make lifestyle adjustments. Both data gathering and interpretation must be done manually in either physical journals or in digital management applications like "Migraine Buddy" by *Healint* [6] or "Migraine Monitor" by *RPM Healthcare* [7]. Regardless of the nature of the journal platform, the general reliance on manual data entry requires constant vigilance and dedication from the migraineur to record and process all important information. This can be taxing in everyday settings, especially for exposed demographics like children, people with cognitive disabilities, or elderly people.

Possible solutions to this may be found in automated data collection through biometric sensors. A data collection trial for migraine research [8] shows that automatic data collection with wearable sensors can be a substitute for manual data collection in child and adolescent populations. And in a systems development project from NTNU [9], daily biometric measurement sessions were used to supplement manual data entry in biofeedback routines, giving the study participants detailed information on their physiological state. According to participant statements, the system had positive effects on their illness management routines and their daily lives.

Correlating biometrics with attacks

Biometric measurements have been shown to correlate with neurological events, like migraine attacks. In a study from 2015, *Pagán et al* [10] shows that migraine attacks can be predicted by utilizing annotated biometric data. Their system was able to forecast migraine attacks ~40 minutes in advance with up to 70% accuracy. Although the results of the study were promising, a complex ambulatory body-sensor network was used for data collection, making this approach unfeasible for continuous monitoring under everyday conditions. Building on this research, in a project from 2018, *Siirtola et al* [11] shows that annotated biometric data captured during sleep by an Empatica (Empatica Inc., MS, USA) E4 wearable device [12] can be correlated with migraine attacks occurring on the following day with more than 95% accuracy for some participants.

Real-time data and prediction

Although showing promising results, both *Pagan et al* [10] and *Siirtola et al* [11] are feasibility studies that apply a two-step process where data collection and correlation experiments are conducted as separate project phases. This retrospective analysis approach is widely used for making initial correlations in datasets, although it may not benefit study participant or patients directly. A natural continuation of this research will be to develop migraine prediction solution that utilize data in real-time, and that could present users with predictions and evaluations on their health within useful timeframes. A system like this would enable migraineurs to react to more quickly to physiological changes and possibly help them to better manage their illness and retain fulfilling lifestyles.

Implementing a system for directly capturing data from the E4 in real-time would enable the use of custom data storage and management, which would be central to facilitate prompt and frequent migraine forecasts. *mSpider* [13] is a secure and flexible data storage solution in development by the Institute for Informatics at The University of Tromsø for storing research data. It provides unified data storage and access options for several brands of biometric data collection devices such as activity trackers, smartwatches, and medical devices. The main premise for *mSpider* is to make data from all supported data collection devices accessible through the same database interface, which is accomplished through specialized data parsing modules for each individual device platform. Currently, *mSpider* has no support for Empatica devices, and integrating support for the E4 with *mSpider* would make data storage and data processing functionality available for a potential migraine prediction system.

Data collection

Access to reliable data is central for facilitating migraine prediction through machine learning (ML). Supervised learning relies on correlations between simultaneous events in different datasets, and in the case of migraine prediction, differences should be found between normal data and special data. Normal data represents the baseline state where the user is not suffering from migraines, while special data covers extraordinary events, like migraine pain episodes or trigger factor encounters. Special data will be used as samples in the ML process, where the number of available samples will determine the prediction success rate of the ML model.

A previous project by the author [14] conducted a simple trial of the data collection capabilities offered by the Empatica E4 data collection infrastructure. The trial found several weak points in the design and implementation of the system, which could lead to data loss and possible user frustration. The most important of these issues was the silent failures where arbitrary errors halted the data collection process without informing the user, which lead to a data collection coverage[1] of ~70%.

[1] The collection coverage of a data collection process is the actual amounts of collected data divided by the maximum possible amounts of data that could have been collected. In terms of collection time, a collection coverage of 100% would indicate that data have been collected at all possible times during the collection period.

Some of the same problems with the Empatica systems are mentioned by *Siirtola et al* [11] where the majority of data loss was caused by either equipment or system faults. *Siirtola et al* [11] initially planned to collect 24h continuous data to be used for short-term prediction. However, data collected from participants during ambulatory hours proved too noisy and incomplete to be useful in the research. This was largely caused by shortcomings in the data collection system. In a related article, *Koskimäki et al* [15] details the data collection experiences of *Siirtola et al* [11], stating that during 753 days of data collection only 358 days of useful data were recorded. This translates to a collection coverage of ~48%. To facilitate reliable predictions, systems must strive to retain the highest possible data collection coverage.

1.2 Related works

1.2.1 mBrain

mBrain [16] is a semi-automatic headache diary system for managing and classifying chronic headache disorders (diagnosis) under everyday conditions. The system was developed through ha project lead Mathias De Brouwer and Nicolas Vandenbussche at Ghent University. Classification of headache disorders is an important step for selecting fitting treatments or therapies and help determine the causes of the individual illnesses. mBrain collects biometric data through an Empatica E4 connected to the mBrain application on the user's smartphone. Algorithms automatically derive physical exercise, sleep periods, and stress level data points based on the raw sensor data. Like other illness diaries, the mBrain application relies on the user for manually recording real-life events that can be tied to their illness, such as headache incidents or medication intake. Detailed information on headache incidents such as duration, intensity and location are particularly important for the system as it is the basis for the classification process. Potential trigger factors can also be recorded through the application interface.

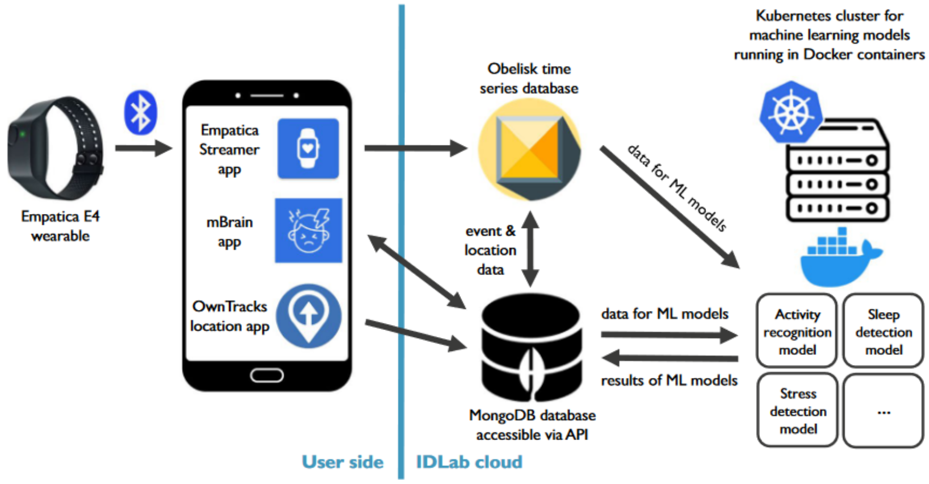


Figure 1: The mBrain system, sourced mBrain [16]

All entered and derived data points are sent to a backend system for storage and evaluation, where machine learning and statistics are employed to find correlations between the derived data points and the manual annotations. The full process is shown in figure 1. Over time, the system should be able to broadly classify the major headache disorders such as tension-type headaches, cluster headaches, and migraines through a knowledge-based approach with foundation in the International Classification of Headache Disorders (ICHD-3) [17]. This is approach was only partially successful, as it is discovered that pain episodes are only infrequently correlated with the expected triggers and symptoms detailed in classifications from ICHD-3. The authors mention that this may be caused by apparent differences between isolated pain attacks, even for individual participants and especially for participants with migraine. This indicates that classifying and detecting headaches will be most effective if performed on a per-participant basis, especially when detecting and identifying individual headache episodes.

The authors state data collection difficulties as an important reason for poor classification results. Over two iterative testing cycles, both lasting ~22 days, the system recorded an average of ~9 and ~13 hours of data respectably, which equates to data collection coverages of 37,5% for the first testing cycle and 54.2% for the second testing cycle. Three main reasons for the low percentages are mentioned by the researchers:

- Connection issues and frequent disconnects between the E4 and the smartphone application.
- Short battery lives on E4 devices (approx. 6 hours).
- Problems with integrating data collection with daily routines.

Another mentioned issue of the system is the high reliance on manual data entry. Accompanied with the mentioned data collection difficulties, these factors may have big impacts on the daily lives of participants/users, further discouraging them from interacting with and collecting data through the system. This would have negative effects on the usefulness of the system, further discouraging use. The authors propose more automation in the data collection process as a probable solution.

1.2.2 Earlier work by author

The project "*Possibilities of automated migraine prediction under every-day conditions: Investigating current technologies and solutions*" [14] performed a trial of the data collection capabilities of the Empatica E4 [12] under everyday conditions. Findings from the project established the E4 as a capable data collection device for migraine research, although not without its flaws and shortcomings. The project focused on the qualitative aspects of the data collection process by using a single device and a single participant over a long period of time.

Table 1: Metadata of data from sessions (Excerpt from Table 7, Ursin [14])

Metadata description	Metadata value
Days of data collection	54.5
Number of sessions	192
Average sessions per day	3.5
Average collection time per session	38336 s = 10.65 h
Data collection coverage	70.98%
Derived data collection rate	801.85 B/s = 2.75 MiB/h

During the data collection trial, the E4 was used in streaming mode in combination with the Empatica cloud solution *E4 connect* (see subsection 2.3.2), meaning that data was continuously streamed to the cloud through Bluetooth to the connected smartphone rather than being stored on the device itself. The trial lasted for ~54.5 days and collected ~55.500 minutes of data from a single participant, resulting in a collection coverage of close to 71% (see table 1). As the device has to be disconnected and powered of for charging, some data collection downtime was to be expected. The particular device used during the data collection period experienced no battery issues and retained the expected battery characteristics of ~24h charge cycles with >2h charging times. Put shortly, charging times only accounted for a maximum of ~8% of the lost collection time, meaning that the remaining 21% of lost collection time was due to various system failures.

Abrupt and irregular disconnects were the most prevalent system failure, which can be seen in the relatively large number of data collection sessions per day (3.5). In the Empatica data collection system, a session is the period from the time an E4 device being connected to the smartphone until the time that the connection is lost. A new session is started every time a connection is established. During the 54.5-day collection period, an average of 3.5 sessions per day were recorded, meaning that the device had to be manually reconnected to the smartphone 2.5 times per day, disregarding charging breaks.

Regardless of the potential data loss, the biggest issue was that most of these disconnects were silent failures, meaning that they happen without the user being informed or alerted, forcing users periodically check whether the system is recording data or not. Over time, this will build mistrust and possibly reduce the user’s willingness to engage with the system.

Another important issue that was uncovered was that the E4 intermittently fails to identify itself and reverts to a default identity. This normally happen after a disconnect and makes the device unrecognizable by the Empatica servers, which renders the device unable to reconnect to any data collection system. Although a manual reset of the device will rectify the issue, this is another silent failure, and the user is never properly informed on what is happening with the device or how the device can be manually reset.

Other issues that were encountered during the project were directly tied the E4 hardware, like the potential for skin rashes or the lack of device UI. These issues will not be discussed here as the scope of this project only include software fixes.

1.3 Project scope and problem statement

The long-term vision for this project is to deploy a highly automated system for managing and forecasting migraine attacks. However, this vision is not possible to achieve with the limited resources and timeframe of this project. Instead, this project will focus on strengthening the data collection capabilities of the Empatica E4 device. The Empatica E4 (see subsection 2.3) is a ubiquitous and non-intrusive sensor device that allows for continuous monitoring of several biometrics under everyday conditions.

Main Research question (RQ):
How can the data collection capabilities for the Empatica E4 be improved with regards to migraine prediction and research?

Even though Empatica offers an application, *E4 realtime,* for real-time data streaming as a part of their data collection infrastructure, captured data can only be accessed retrospectively through the *E4 connect* cloud storage solution. Accessing captured data from the E4 in real-time would greatly improve the possibilities for short-term migraine prediction as described by *Pagán et al* [10]. Additionally, direct data capture would also enable additional data storage options.

Sub-question 1 (SQ1)
How can a system for real-time data capture and processing be implemented?

As mentioned earlier, the data management system mSpider [13] offers a flexible and secure data storage solution. Integrating a real-time data collection system with mSpider would increase data availability and data security.

Sub-question 2 (SQ2)
How can the data collection system be integrated with mSpider?

As described by *Siirtola et al* [11] and *Pagán et al* [10] annotating the captured data is central for facilitating machine learning approaches. The data collection system must include facilities for annotating captured data and tagging important real-life events.

Sub-question 3 (SQ3)
How could real-life event-tagging be implemented in the system?

As mentioned in sections 1.1 and 1.2, multiple studies have pointed out flaws in the data collection capabilities of the E4 and its support infrastructure. Minimizing data loss would increase data availability and increase the potential of migraine prediction through machine learning.

Sub-question 4 (SQ4)
How can data losses during data collection with the E4 be minimized?

1.4 Project Methods

A. Perform a systematized literature review for investigating and mapping issues with the Empatica E4 and the "E4 connect" data collection system.
    - Investigate prevalence and magnitude of data loss.
    - Define causes of data loss.

B. Design and implement a data collection system that enables real-time data manipulation from the E4 and that can store data in mSpider.
    - Create a requirement specification for a prototype system based the findings from the literature review in subsection 6.1.3 and on issues outlined in subsection 1.2.
    - Create a proof-of-concept prototype based on established requirements.

C. Conduct user and field testing of the developed system on potential end-users.
    - Participants use the system to collect data over a fixed period of time.
    - Analyze metadata to get collection coverage.
    - Analyze metadata to get data storage sizes.
    - Interview participants on their use of the system (Appendix C and D)

1.5 Report structure and contents

This report details the process of researching, designing, implementing, testing, and evaluating a prototype system for answering the main research question (RQ) and the sub-questions (SQ). As described it the previous section, this is accomplished through three interconnected methods. However, this report does not list them as separate processes and the list below shows where each step of the individual methods may be found.

A. Review methods can be found is subsection 3.1, while the review results and interpretations can be found in subsection 6.1. The results of the review were used to create the requirement specifications.

B. All methods used during the requirement, design, and implementation phases are located in subsection 3.3 and 3.4. Requirement specifications for the prototype system can be found in section 4. Requirements are used for design and implementations in section 5. A system evaluation is located in subsection 7.1.

C. Methods for all testing endeavors, including interviews, are described in subsection 3.5. Testing results are shown in subsections 6.2, 6.3 and 6.4. Discussions of the results are located in subsection 7.2.

In addition to sections tied to the project methods, technical background on important topics in this project may be found in section 2, while subsection 7.3 contains discussions on topics that are not directly tied to the project methods or processes described above. A section on future work can be found in 8, and conclusions are located in section 9. Finally, appendices A through F can be found after the references. Please note that all appendices are in Norwegian.

2 Technical framework

2.1 Migraines

Migraine is a complex neurophysiological headache disorder which is mainly characterized by episodes of moderate to severe headaches. Pain episodes are described by migraineurs as throbbing and pressure-like, normally tied to a single brain hemisphere, and often accompanied by secondary symptoms like nausea, photophobia and phonophobia [5]. Although the underlying causes of migraines are not known, research indicates that failures or imbalances in the relationships between neurological sensory processing and various homeostatic mechanisms may be to blame [5].

Resilience and thresholds

The homeostatic mechanisms of the body consist of several major and minor bioregulatory systems, all working towards keeping various local and global balances. None of these systems are completely self-contained, which creates a complex, interwoven, and moderately chaotic interplay of mechanisms and activators. Examples of theses mechanisms can be seen in the endocrine and hemodynamic systems of the body. The interplay between the regulatory systems is in constant flux, resulting in high degrees of resilience [18]. This makes them able to rapidly adapts to changes in internal states caused by external or internal antagonists, like stress or food intake. Intermittently, some regulatory mechanisms are unable to keep their balance. This can cause cascading events in other interconnected systems, which can ultimately cross thresholds where the normal homeostatic balance is impossible to retain [18]. Breakdown events like this often result in observable external events like migraine attacks, strokes, or heart attacks.

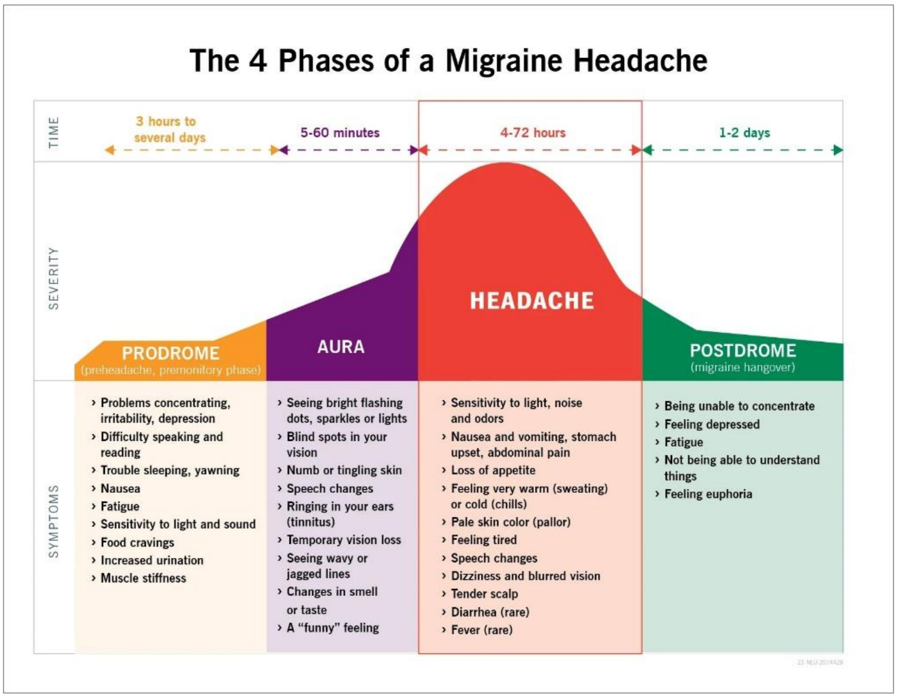

The 4 Phases of a Migraine Headache
TIME
SEVERITY
SYMPTOMS
3 hours to several days
5-60 minutes
4-72 hours
1-2 days
PRODROME
(preheadache, premonitory phase)
AURA
HEADACHE
POSTDROME
(migraine hangover)
› Problems concentrating, irritability, depression
› Difficulty speaking and reading
› Trouble sleeping, yawning
› Nausea
› Fatigue
› Sensitivity to light and sound
› Food cravings
› Increased urination
› Muscle stiffness
› Seeing bright flashing dots, sparkles or lights
› Blind spots in your vision
› Numb or tingling skin
› Speech changes
› Ringing in your ears (tinnitus)
› Temporary vision loss
› Seeing wavy or jagged lines
› Changes in smell or taste
› A "funny" feeling
› Sensitivity to light, noise and odors
› Nausea and vomiting, stomach upset, abdominal pain
› Loss of appetite
› Feeling very warm (sweating) or cold (chills)
› Pale skin color (pallor)
› Feeling tired
› Speech changes
› Dizziness and blurred vision
› Tender scalp
› Diarrhea (rare)
› Fever (rare)
› Being unable to concentrate
› Feeling depressed
› Fatigue
› Not being able to understand things
› Feeling euphoria

Figure 2: Phases of a migraine attack, sourced my.clevelandclinic.org.

Migraine attacks

Migraine attacks are often divided into four main phases: the *prodrome* phase where early symptoms or indicators may be recognized, the *aura* phase where sensory distortions are common, the *headache* phase where the majority of pain is experienced, and the *postdrome* phase of physiological and neurological aftermath. Detailed symptoms of each phase are listed in figure 2. Not all migraineurs experience all the different phases, and each individual migraine attack may involve different phases and symptoms.

In the case of migraine prediction, the *prodrome phase* is most interesting. In this phase, internal bioregulatory systems are unbalanced, leading to physiological and neurological symptoms like fatigue, communication difficulties and concentration problems. Many migraineurs are susceptible to various *trigger factors* [5], which can be highly varied between individual migraineurs. Triggers are external of internal stimuli like noise, stress, or particular smells that act as migraine antagonists. This causes changes in the neurological state of the brain and triggers threshold events that bring on the next step of the migraine attack. These rapid changes in neurological and bioregulatory systems are externally measurable.

Causes of pain

Although the exact causes of migraine pain are not known, neurological and vascular changes in the brain may be responsible. During an attack, the most probable cause of pain may be attributed by rapid dilation of capillaries in the outer parts of the brain [5]. On top of this, parts the brain becomes inflamed, and the activation thresholds of pain receptors are lowered, further increasing the perception of pain.

Management and medication

There is no current medical cure for migraine. Neither are there any effective medical preventive treatments, making pain relief and nausea dampening drugs the current best pharmaceutical response to the illness [19]. As mentioned in subsection 1.1, diaries and biofeedback can be effective tools in migraine prevention, although they require substantial amounts of manual data entry and interpretation to be effective. Because of the inflammatory aspects of migraine, anti-inflammatory medication can often be effective at blunting pain perception during attack. Drugs like *triptans* that reduce capillary dilation are also very effective at removing headache, although they normally have serious side effects and must be taken in moderation [5].

2.2 Migraine prediction

As mentioned in subsection 2.1, many of the internal processes of the body are externally measurable. If data on these processes can be recorded and correlated with migraine attacks, the resulting models may be used to forecast migraine events. This was shown to be possible by both *Pagán et al* [10] and *Siirtola et al* [11], although with differing goals and approaches. In their project, *Pagán et al* [10] experimented with short-term prediction using highly detailed data from a body-sensor network, managing to successfully forecast individual migraine attack with high precision within a timeframe of less than an hour. On the other hand, *Siirtola et al* [11] used less detailed data collected from an unobtrusive wearable device (Empatica E4) for successfully estimating the probability of the occurrence of migraine events during the following day. Both these approaches utilize supervised machine learning (ML) and statistical methods for correlating manually collected discrete events in the form of recorded migraine attacks with automatically collected continuous data.

The quality and "resolution" of data determines the accuracy and timeframe of predictions. Because of their high data resolution and quality, *Pagán et al* [10] could utilize a fine-grained approach where the continuously collected data is divided into 5-minute discrete sections. This makes it possible to correlate individual section with defined stages of the migraine attack, facilitating short-term prediction. On the other hand, the sparse and noisy data used by *Siirtola et al* [11] forced them to utilize data segments with full-day resolution, where full days of biometric data are correlated with singular migraine attacks. This made it possible to predict the probability of migraine attacks with a much coarser daily granularity.

Both these approaches are interesting in real-time migraine prediction as they fulfil separate roles in the biofeedback and management process. Coarse-grained predictions can be utilized for daily planning, while the fine-grained predictions can be used as an alarm system.

2.3 Empatica E4

2.3.1 E4 sensor device

The Empatica E4 is a wrist-worn sensor device designed to collect biometric data from the wearer under everyday conditions. It is certified as medical device under the European Union Medical Device Directive (EUMDD) and meets the requirements for ISO 13485:2016 Medical device quality management regulations [20], [21] and it is still one of the best regarded sensor devices for collecting ambulatory data, even as it was first introduced in 2014. The device is inobtrusive and facilitates fully autonomous data collection besides a single button for manual data input.

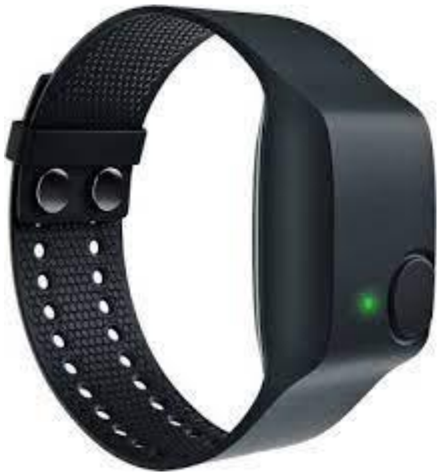


Figure 3: Empatica E4, sourced empatica.com

The device offers two operational modes: data storage mode and data streaming mode. In storage mode, the device stores all collected data to internal memory. The device battery size allows for continuous operation in storage mode for at least 36 hours, and the internal storage may store up to 48 hours of data. The device must be connected by cable to a computer using the *E4 manager* software for uploading data to the Empatica cloud. In streaming mode, all data is sent directly to a connected device through a BLE connection. Software for connecting to the device and receiving data is available for Windows, MacOS, iOS and Android. Data is automatically uploaded to the Empatica cloud storage solution when a data collection session is ended. A new session is started when the E4 is turned on and ends when the device powers of. Due to the active data streaming, the device battery life is typically lower in streaming mode than in storage mode, lasting approximately 24 hours between each <2h charge.

Regardless of the operational mode, the E4 is can produce both raw and derived secondary data in real-time from a variety of sensors:

- **PPG (photoplethysmography)**
  Used for calculating blood volume pulse (BVP). Measures amount of red blood cells by using changes in light reflection. Emits monochromatic light at a specific frequency that is absorbed by oxygenized red blood cells. Light that is not absorbed by the red blood cells is measured and used to estimate the amount of red blood cells in the target area.

The raw data from this sensor can be used to derive secondary measurements like inter-beat intervals (IBI), pulse and heart rate (HR). The PPG sensor has a maximum polling rate of 64Hz.

- **EDA (Electrodermal activity)**
  Measures electrical conductivity of outer skin layer. Skin conductivity can change very quickly based on micro activity in the sweat glands, which is controlled by the sympathetic nervous system. Both biological and neurological states or events can alter the sweat excretion, which makes them potentially detectable through EDA data. The E4s EDA sensor has a polling rate of 32Hz.

- **Thermopile**
  Measures infrared (IR) emission from the body. Can be used to accurately estimate the skin temperature by measuring IR radiation in time segments. Variations in body heat can be used for detecting physical activity, for strengthening hemodynamic measurements, etc. The IR thermophile in this device has a maximum polling rate of 4Hz.

- **Accelerometer**
  A piezoelectric sensor is used to measure device acceleration on three axes. Acceleration measurements can be used to calculate movement, energy expenditure, etc. The accelerometer in the E4 has a maximum polling rate of 32Hz.

- **Manual tags**
  Saves Unix timestamp on button press. Timestamps can be used to mark important information. An example of this can be seen in *Siirtola et al* [11] where manual tagging is used to mark migraine attacks.

2.3.2 E4 connect and E4 realtime

E4 realtime smartphone app

E4 realtime is the name of the real-time data streaming application delivered by Empatica. As can be seen in figure 4, the application features a base UI with simple data visualization tools for each of the E4s sensors (PPG, EDA, skin temperature, acceleration), battery and connection time with the device. On top of this, the app features a more comprehensive and detailed data graphing interface that reacts in real-time to the data streaming from the connected device. This can be a very useful tool for closely monitoring the supported biometric variables under specific circumstances like exercising (see figure 4).

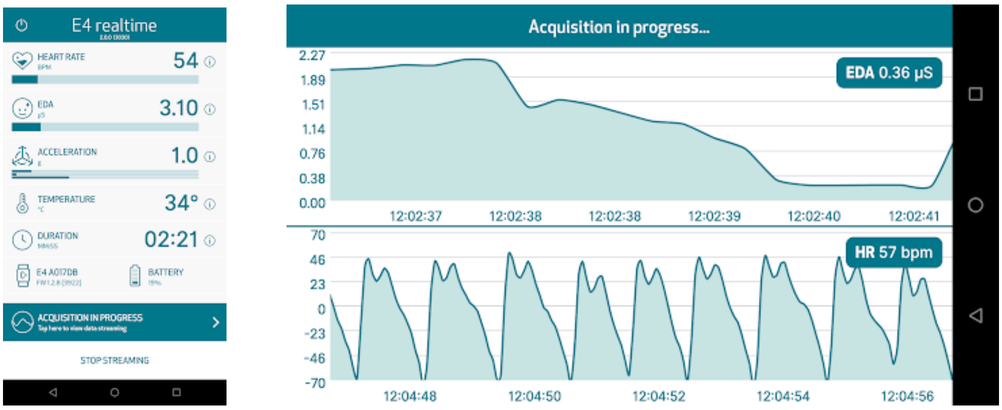


Figure 4: E4 realtime smartphone application.

Although the application is user-friendly and displays useful real-time insight on multiple biometric variables, it only offers the bare minimum for ensuring persistent data collection. Firstly, device disconnects are only silently handled by the application, meaning that users have to detect disconnects by themselves, potentially leading to periods without any data collection. Secondly, the application has no persistent presence in the smartphone operating system, meaning that it can be accidentally killed by the user if operating system (OS) cleanup mechanisms are used. This would also lead to a collection device disconnect and potential data loss. Thirdly, the E4s tag-feature (button press) has no in-app functionality, meaning that tags are sent to cloud storage without any contextual information. This may not be an issue for many of the systems potential use-cases, although for migraine or other neurological research, where many different events like stages in a migraine attack or potential trigger factors may occur, it is importation to be able to clearly differentiate between tagged events.

E4 connect cloud solution

Empatica offers the cloud storage solution *E4 connect* as a part of the E4 data collection system. The cloud solution stores uploaded data and makes it available for download through a web page interface. However, only manual on-click single-session downloads are available, and there is no API for accessing the stored data through scripting. Neither is there an option for downloading all data at the same time, making the process of retrieving data very cumbersome even for small research projects. This is mentioned as a major problem in *Siirtola et al* [11], whose research involved several participants over a long period of time, resulting in large amounts of individual sessions. Solutions to this issue would be to contact Empatica for download support, utilize data-scraping techniques for mining the data directly from the web interface, or to create a separate data storage solution for your project.

2.3.3 Empalink SDK

In addition to their native E4 data collection infrastructure, Empatica offers the *Empalink* software development kits (SDK) for both Android and iOS environments. This subsection will focus on the Android SDK as it was utilized in this project, although both SDKs have similar functionality. The Android SDK consists of a Java library that can be imported into any Android studio [22] project, and creating an instance of the *EmpaDeviceManager* class gives access to management and connection for an E4 device. Devices are automatically authenticated when connected through the *EmpaDeviceManager*. An Empatica account at *E4 connect* is required for this, and the connecting E4 device must be registered with the account to complete authentication.

The instantiation of the *EmpaDeviceManager* class starts a separate controller thread for the E4 infrastructure. Data and status events streamed from the E4 are automatically pushed from the *EmpaDeviceManager* through a class method interface. The receiving class will implement methods for handling incoming data and status events. Data points are pushed individually through the interface for each sensor polling in the connected E4, which equates to >100 individual data inputs per second. Data flow from the *EmpaDeviceManager* can be hampered by slow execution times in the receiving methods and infrastructure.

2.4 mSpider

mSpider is a data collection solution developed by *Henriksen et al* [13] for effectively storing and utilizing biometric data across wearables of multiple types and brands. The system was created for simplifying data collection in large populations and for centralizing the inherent research potential in already existing data collection infrastructure. As can be seen in figure 5, mSpider has support for many of the major health trackers and smartwatches. Although other commercial data aggregation systems like *WeFitter* [23] offer comparable functionality, mSpider is the currently the only platform that has been developed purely for research.

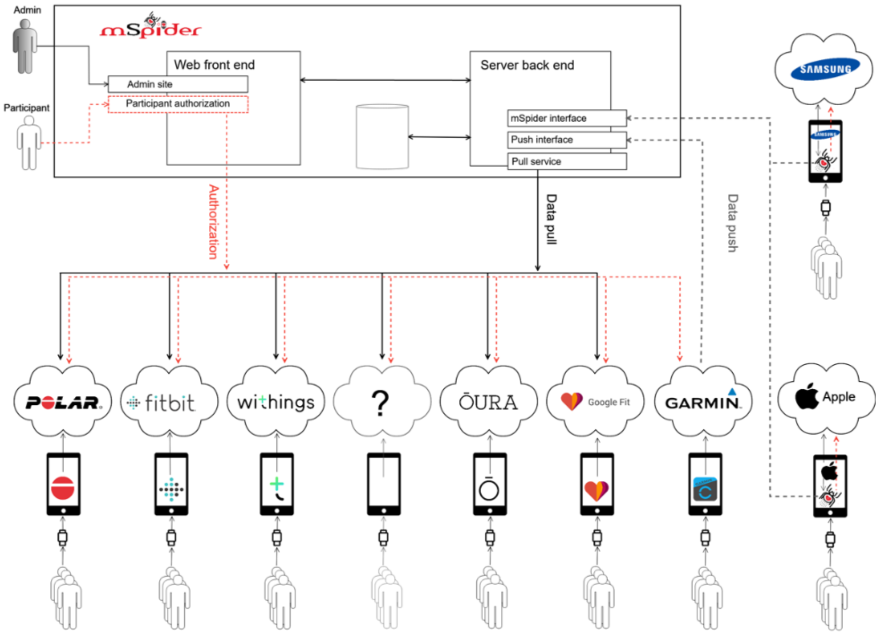


Figure 5: Architecture of mSpider, sourced Henriksen et al [13]

The mSpider system consists of multiple, brand or device specific, data collection modules. Each module is specially designed for actively fetching data from a specific brand or device line. The system relies on OAuth tokens, making it possible the system to utilize the participant's own login credentials for fetching data. Additionally, mSpider incorporates API endpoints for pushing data from devices or brands that have no support for data pulling, making it highly versatile.

mSpider is currently undergoing major redesign to increase its flexibility and security, and to make it production ready. As discussed by *Johannessen et al* [24], performing a through risk-assessment and changing the design would ready the system for use in large and complex data collection projects, like the 8th iteration of The Tromsø Study (Tromsø8) [25], which is scheduled for 2025.

3 Methods

This section details all methods used for completing the separate parts of this project.

3.1 Review of data collection experiences with the Empatica E4

This subsection contains the methodology used to conduct a systematized literature review on the experiences gained in the medical research community from using the E4 and its support system as data collection tools. Results of the review can be found in subsection 6.1.

3.1.1 Review objectives

As mentioned in subsections 1.1 and 1.2, previous projects [11], [14], [16] have experienced issues regarding the data collection capabilities of the E4 and its support systems. This review investigates to what extent the reported shortcomings have been encountered throughout the medical research community and if solutions to these issues can be derived from their findings. The first objective of this review was to extract and quantify the findings and conclusions from relevant research on the use of the E4 as a medical data collection device. The second objective was to find possible solutions for mitigating the effects of the issues. Findings from the review were used as a basis for the requirements specification in section 4.

3.1.2 Inclusion criteria

The following selection criteria were used as a basis during the manual scrutiny of the search results. Eligible articles must:

- be available in full-text.
- be in either in English or Norwegian.
- use the E4 for data collection.
- discuss the data collection process using the E4.
- use more than the accelerometer data from the E4 device.

Due to the objective of this review, only studies that conduct real-life data collection should be included and studies that rely on lab experiments should be discarded. However, studies with lab experiments or experiments under very controlled conditions should still be included if they comment on the data collection process.

3.1.3 Databases and the query process

Although this project and the main premise for this review is of a technical nature, the E4 has mostly been used for data collection in medical research. Because of this, both medical and technical databases were searched:

- ACM digital library
- PubMed
- IEEE Xplore

Because this review was result-orientated, compiling a fitting search query was a non-trivial process. Relevant articles may be found in many different areas of research and the wording of the results may differ wildly based on the technical knowledge of individual research groups, creating a very wide potential search area with a myriad of search terms.

The initial solution to this issue was to create a limited search query that encompassed most of the medical and technical fields that were likely to utilize the E4 for their research. Inclusion and exclusion terms were added until a manageable number of articles were left. This approach was abandoned as it was too complex and produced many false positives and too few usable results.

The second solution took the opposite approach and started with the simple search for the term "Empatica E4". This resulted in large but manageable numbers of results from all the chosen databases and refinement of the search string was not needed. This may be because of the relative recent introduction of the E4 device (2014).

Final search term: "Empatica E4"

3.2 Privacy and security

This subsection evaluates and describes the privacy and security measures implemented for this project.

3.2.1 Project data handling summary

This project aims to create a prototype system for collecting biometric data under everyday conditions. Testing of the prototype system involves the recording, storing, and handling of biometric data from third-party end-users. The collected data are used to evaluate the data collection potential of the prototype system. Additionally, each end-user takes part in a semi-structured interview at the end of the data collection period. Answers from the interviews are used to evaluate the effectiveness of the data collection features of the prototype system.

3.2.2 Responsibilities

Personal data is defined as all information that can be used to identify an individual [26]. This also extends to information that can be used to indirectly identify individuals, such as thorough statistical information on very small populations. *Sensitive information* is a subcategory of personal data defined as information that can potentially be used to harm or disadvantage individuals, such as health data [27]. *Health data* is defined as all information tied to the health of an individual, including all data captured through medical devices. As the Empatica E4 is categorized as a medical device (see subsection 2.3.1) and is utilized for collecting user biometrics, this project must follow the EU General Data Protection Regulation (GDPR) [28].

3.2.3 Implemented measures

Consent form

The rights of the data subject are detailed in GDPR chapter 3 [29]. Especially relevant for this project is the participants right to know what the collected data will be used for and their right to deletion of data. A consent agreement document was created to ensure that the participants are informed of their rights and to give them the tools necessary for exercising said rights. The consent form includes links to information sources and contact informant to the project staff. Additionally, the individual participant's pseudonym is noted on the consent form, making it the only connection between the participant and the collected data, thus enforcing privacy regulations. Signed participant consent forms have been stored in a secure location and have only been accessible by the project staff. An unsigned version of the consent form can be seen in appendix A.

Pseudonymization

As mentioned in the previous paragraph, each participant has been given a pseudonym. All data collected through the project have been stored with connections to the pseudonyms. This results in no personal data being stored in the system.

Software security

- All communications between the fronted and backend parts of the system use secure HTTPS connections.
- The smartphone application relies on the native security measures of the OS.
- Backend database access is abstracted by an API solution.
- The mSpider software is running on hardware on the UIT campus.

3.2.4 Summary of correspondence with authorities

The implemented privacy and security measures are sufficient, and the project has been approved for handling personal data. Full correspondence with authorities can be seen in appendixes D, E and F.

Table 2: Summary on correspondence with authorities

Authority	Status	Result of correspondence
UiT DPO	Approved	Participant consent form (see Appendix A) looks good. Please contact SIKT for user privacy evaluation.
REK	Approved	Project only handles anonymized data and there is no need for extra measures.
SIKT	Approved	The user privacy measures implemented for this project are sufficient. The project can collect and utilize data within the specified measures.

3.3 Requirement specification methods

The requirement specification for the prototype system is based on the main research question (RQ) and sub-questions (SQ) of the project stated in subsection 1.4, the mentioned issues with related systems from subsection 1.2, and the findings and conclusion of the literature review in subsection 6.1.3. The Volere [30] requirement specification template was used in the requirements process. It consists of a well-defined and extensive list on the parts and procedures needed for creating a full requirement specification, although the procedures described in the template have been narrowed down and adjusted to fit the scope and scale of this project.

As the premises and purpose of the planned system have been defined through section 1, the initial step of the requirement specification is defining the systems intended audience and stakeholders. This is then followed by describing the intended usage through descriptions of possible usage scenarios and an overview of how each user or stakeholder is expected to interact with the system. Proper and through definitions are important parts of the planning process as they force designers and planners to view the system from a user perspective, possibly revealing hidden flaws or biases in their vision or initial system design. Usage specifications and lists of requirements can be seen is subsections 4.1.4 and 4.2.

Requirement categories chosen included in this project:

- **Functional requirements** are tied to the intended usage and use-cases of the system. In systems that include user interaction they are often salient and noticeable, and they highlight important aspects and must-haves for ensuring functionality. Examples of functional requirements are UI functions for displaying data or data extraction options for researchers. Requirements in this category were based on RQ and SQs (subsection 1.4), findings from the literature review (subsection 6.1.3), and the flaws presented in the related systems (subsection 1.2).

- **Interface requirements** concern all edges of the system where parts or components interact with each other. UI, network, and database connections can all be categorized as system interface elements. Requirements in this category were based on the know possibilities and limitations of the operating environment [31].

- **Non-functional requirements** cover all functionality that allows the system to function in its environment. This includes housekeeping tasks and performance, all functions that are tied to the operation and infrastructure of the system. Requirements in this category were based on prior knowledge and the Android Guide to app architecture [31].

- **Security and privacy requirements** cover functionality that ensures safe, secure, and trustable environments for collecting and storing potentially sensitive data, including secure communications, storage, and interfaces. Requirements in this category were based on statements from subsection 3.2.

3.4 Software development methods

3.4.1 Design principles

Although the main premise for this project is based on facilitating real-time data handling as well as increasing the data collection potential with the Empatica E4, the development process followed a user-centered approach. Findings from related works and the literature review (subsection 6.1.3) indicate that increased user involvement in the data collection process could be highly important both for improving the data collection capabilities of the E4 (RQ, SQ4) and for facilitating continuous real-time data capture (SQ1). In user-centered development, users are normally employed for testing and giving feedback on the different iterations of the software. This was not feasible for this project as only a single E4 device was available, leaving iterative testing to be performed by the developer.

Although both the earlier mentioned Empatica environment and the mBrain system employ many automated features in their data collection processes, their data-centered designs indirectly force the user to take overall data collection responsibility. For instance, as the E4 is used for data collection in both systems, they are prone to intermittent and relatively frequent device disconnects, which naturally stops the data collection. These failures are allowed to happen silently and without alerting the user, which forces them to frequently check whether the system is collecting data. This can easily cause user fatigue and irritation, resulting in reduced user compliance and data loss.

For increasing data collection potential, the data collection responsibility must be transferred from the user to the collection system, where the system has access to tools for actively requesting help from the user when needed. Under optimal conditions, this will result in a system that can maximize the data collection coverage while minimizing the burden on the user. On the other hand, allowing the system too much autonomy could result in overactivity, effectively negating the positive effects of the automation aspects. Finding the appropriate level of system-initiated user interactions will be very important.

3.4.2 Development methods

The development of the prototype system followed an iterative bottom-up approach, which was mainly a result of the developer's lack of experience with the chosen platforms and necessary frameworks. The iterative development process consisted of six main phases, where components and features were added to the system when they were needed or when their supporting infrastructure was completed. Rudimentary field-testing was performed between each iteration and frequently between the implementation and inclusion of individual components. List of phases with main focus and components specified:

Table 3: Development phases

Phase	Area of focus
1 Initial	Familiarize with development frameworks and tools. Create initial prototype application for communicating between a smartphone and the E4.
2 Infrastructure	Create data collection infrastructure for storing data locally on the system frontend.
3 User interaction	Create a user interface. Implement persistent local storage for user preferences. Create functionality for handling user input.
4 Communications	Create notification functionality to enable the system to alert the user. Set up infrastructure communications with external servers.
5 Backend	Create mSpider module for receiving and handling data. Implement primitive user login system.
6 Field-testing	Test the performance of the system in the field under everyday conditions. Identify and remove bugs.

3.5 Testing methods

3.5.1 Testing phases

Testing of the prototype system was performed in four distinct phases, each with their own goal:

- **Testing during iterative development** conducted by the author with the goal of identifying important bugs and faults while details of the development process are still easy to recall. An additional sub-goal was to learn important development patterns in the chosen tools and frameworks and implement them into the system.

- **Field testing on development completion** conducted by the author with the goals of identifying possible issues with long-term system usage under real-life conditions, as well as to gather knowledge on system operation for the *User information sheet* (see Appendix B).
- **Performance testing on the finalized prototype** with the goal of measuring and quantifying system performance under everyday conditions, which is important for determine whether the system is fit for use.
- **User testing and interviews** with the goals of gauging system usability and to measure the effectiveness of implanted features for increasing data collection rates and coverages.

3.5.2 Testing during development

Iterative testing was performed during system development. Each new feature or component were tested by the author for 12-24 hours in the environments and settings where the full prototype was meant to function. Although this testing process was not exact or systematic, it had positive impacts on the development of the prototype, as major faults and shortcomings could be easily found and scrutinized in high detail. A simple logging system was utilized to identify faults and track down their origins.

3.5.3 Field testing

Field testing was conducted over 2 weeks where the system was used by the author to continuously collect data under everyday conditions. During this period, both minor and major faults in the system were found and rectified. This increased system stability, decreased power usage, and introduced various minor quality-of-life changes. Field testing is the last phase of the development period, and the system was considered completed when this testing phase was over. Field testing was performed by the author using both a *Samsung Galaxy Note 8* smartphone running Android 9 and on a *Samsung Galaxy S21* smartphone running Android 12L. A single Empatica E4 data collection device was used during testing. All devices were fully functional.

3.5.4 Performance testing

Performance testing of the prototype system includes multiple measurements. Results of the performance testing can be seen is subsection 6.2.

- **Impact on smartphone battery life** is measured by comparing the battery usage of a smartphone while the system is fully active against baseline power usage over a fixed period of time. As a baseline, the regular battery usage of the smartphone was measured during 8 hours of sleep over 3 consecutive nights. The same procedure was performed with the prototype application installed, both with and without the E4 connected for data collection. In addition to the data from experiments, the profiling tools in *Android studio* [22] are be used to roughly estimate application's power consumption.
- **Memory usage** can be hard to measure accurately over long time periods, although profiling tools in Android studio can be used to create estimates on average memory requirements. To generate metrics on memory usage, the Android studio profiling tools were connected to the prototype application for ~1 hour while the system was collecting data.
- **Data storage requirements** can be measured by counting the number of bytes collected and stored by the system. Facilities for measuring and calculation data storage requirements

have been included in the prototype system and can be read directly from the application UI. Changes in stored data volumes over 1 hour were measured on 5 different occasions.

- **Data upload sizes** can be measured by counting uploaded bytes. This can be measured directly in the application code, it can be read through the Android OS application settings, and it can be estimated by the system backend. All these approaches have been used to calculate average data upload sizes for the system.

All performance testing was performed by the author on a *Samsung Galaxy Note 8* device running Android 9 and connected to a single Empatica E4 data collection device. Both devices were fully functioning during testing.

3.5.5 User testing

Three testing participants (n=3) were involved in the user testing process of the prototype system. Participants were selected by the author among friends and acquaintances. Each participant utilized the prototype system for approximately 6-7 days (~1 normal week) with the goal of collecting as much data as possible during this period. Each participant has read and signed a participant consent form, which can be seen in appendix A. The participants individually attended a 30-minute instruction session on the use of the system and equipment, and they were given an information sheet (see appendix B) containing operational details on the system and the E4 device. The prototype system was installed on the participant's private smartphone, and all participants used the same E4 collection device.

The number of data packets collected per participant was used to calculate their individual collection coverage percentages. Timestamps on the first and last data packets sent by the participant was used to measure their collection times.

3.5.6 User interviews

After their data collection period, each participant attended individual, semi-structured interviews where they were questioned on their experiences with using the system for data collection. Notes were taken and no audio was recorded during the interviews. The interview guide for the semi-structured interviews can be seen in appendix C.

4 Requirement specification

This section details the requirements specification process for the prototype system based on the methods described in subsection 3.3.

4.1 Intended audience and usage

4.1.1 Audience and limitations

The system in its prototype state will be mainly intended for medical research. Although the long-term vision for the presented concept is to facilitate automated migraine prediction for migraineurs, the timeframe and scope of the project do not allow for fully exploring all aspects of the complete system. This limits the potential usages of the prototype system to a large degree, leaving researchers and their research participants as the current systems main audience.

4.1.2 Intended system deployment

Both the prototype and the long-term vision variants of the system are meant to be deployed for continuous data collection (24h) under everyday conditions over long periods of time (years). Users or participants are meant have access to a Empatica E4 device and they are meant to install the data collection system on their private smartphone.

4.1.3 Stakeholders

Regardless of the stage of development, the system has three distinct stakeholder groups or actors. These are shown as points 1, 2 and 3 in figure 6. Users have full data input access, and they also have the ability to extract small amounts of data such as minor statistics. Researchers or physicians have full data extraction access and limited access to data editing and insertion. Administrators have access to user registration and other user account features, although they have no access do either data input or data extraction.

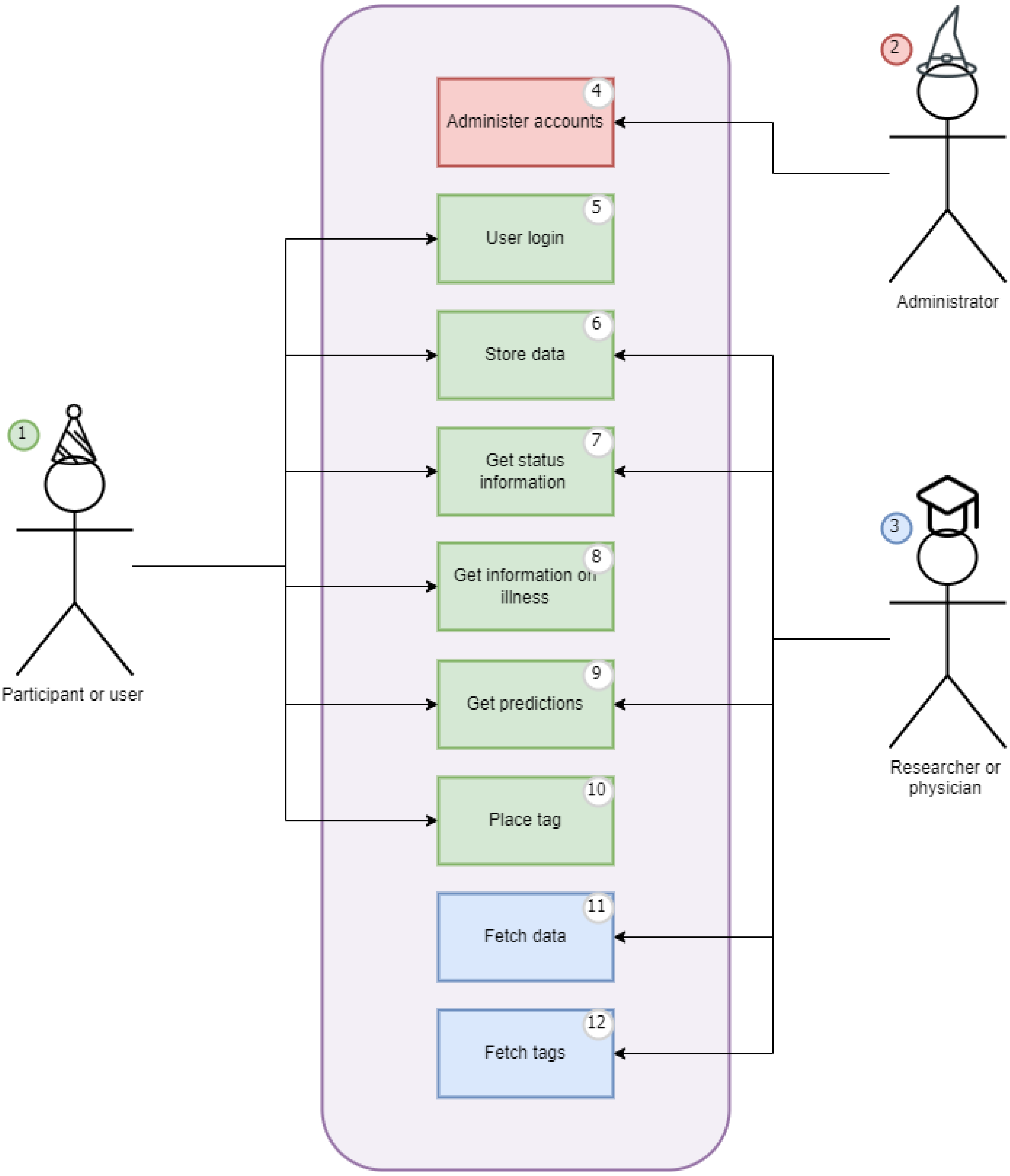


Figure 6: System use-case diagram

4.1.4 Use-cases

A graphical representation of the intended use-cases can be seen in figure 6, which shows the intended features of a fully implemented system, although the prototype has some feature limitations which are noted in the description below. Different actors are marked with numbers in range 1-3, while system features are marked with numbers in range 4-12. Arrows between actors and features represents intended use-cases.

1. Participant or user. Main feature target and arguably the actor that benefits most from a complete system. Users will access the system both actively and passively. Possible user actions are:
    - User login (5). Active usage. This is a security and logistics measure for authenticating users and connects them to their data through an account system (4). This is done anonymously for research, although non-anonymous options could be made available for medical contexts.
    - Store data (6). Passive usage. The main goal of the system is to successfully collect data from the user. This normally happens through a user-worn E4 sensor device and data is automatically stored in the system. A description of major system components can be found in subsection 5.2.
    - Get status information (7). Active and passive usage. The users can view statistical and medical status information on demand in the system UI. Should be available in real-time.
    - Get information on illness (8). Active usage. The user can access information on their illness or on the significance of individual biometric variables relating to their illness.
    - Get predictions (9). Active and passive usage. The system will automatically warn the user of imminent migraine attacks. Users may also view their current estimated migraine status on demand.
    - Place tags (10). Active usage. The user can tag important events with timestamps. In the case of migraine, events can be encounters with known triggers or the start of migraine attacks. This is a very important data collection feature as it provides ground truth or annotations for the automatically collected biometric data.

2. Administrators of the system (2). Responsible for the functioning of the system and for access supervision. As the prototype data collection system is integrated with mSpider [13] for data storage, the responsibilities of the administrator role have not yet been completely finalized, and all use-cases tied to this actor are marked with dashed lines. Current administrator actions are:
    - Administer accounts (4). Active usage. The admin is responsible for maintaining the security and privacy of actors by providing proper access control to the system. This includes account creation for new users and establishing appropriate access levels for each stakeholder group.

3. Researcher or physician (3). Active usage. Responsible for handling and manipulating collected data. These actors have not been prioritized during the development of the system and is only included here to demonstrate the long-term system vision. As all the backend features for the system are managed through mSpider, none of the features meant for these actors have been implemented by this project. As mentioned in subsection 2.4, the mSpider system currently under major revision and will incorporate standardized native features for

data extraction and manual data insertion in the future. Although, for the sake of completeness, features intended for the researcher or physician actors will be detailed.

- Store data (6). Active usage. Researchers may remove faulty data or add metainformation to the automatically collected user data. Researchers are also responsible maintaining the mostly automatic prediction system.
- Get status information (6). Active usage. Researchers and physicians can access status information on a user. This information should be stored and should not be available in real-time to ensure patient privacy.
- Get predictions (9). Active usage. Physicians can access status information on a user. This could be an important tool for monitoring user medical progression, especially for physicians. This information should be stored and should not be available in real-time to ensure patient privacy.
- Fetch data (11). Active usage. Researchers can fetch data from the system for statistical purposes. As the prototype system is integrated with mSpider, this can be performed without violating user privacy.
- Fetch tags (12). Active usage. Researcher can fetch tag data inserted by users. This is a very important feature of a complete migraine prediction system, as the tag data will be the ground truth for the ML process.

4.2 Lists of requirements

This subsection will list the requirements for a real-time data collection system for migraine data. The requirements are all marked with an ID number that will be referenced in the design section to indicate which requirements are fulfilled by each component. The ID number will also include category information. Requirement are presented with rationale and completion criteria, and the requirements are considered met if their completion criteria are fulfilled. Each requirement is also denoted with an importance rating: high, medium, and low. Requirements marked "high" are most important and will be completed first. System overviews can be seen in subsections 5.1 and 5.5 for clarification on components and communications.

As mentioned earlier, requirements are based on findings from the literature review in subsection 6.1.3, research questions RQ and SQs in subsection 1.4, privacy and security in subsection 3.2, findings from mBrain in subsection 1.2.1, and findings from earlier work by the author in subsection 1.2.2. Affiliations of the individual requirements are noted.

4.2.1 Functional requirements

As mentioned in subsection 3.3, functional requirements are tied to the intended usage and use-cases of the system. In systems that include user interaction they are often salient and noticeable, and they highlight important aspects and must-haves for ensuring functionality. Examples of functional requirements are UI functions for displaying data or data extraction options for researchers.

ID	Import.	Requirement	Rationale	Completion criteria	Source
F-1	High	The application must acquire permissions for communicating with Bluetooth devices.	Access to Bluetooth functionality is essential connecting the data collection device to the system.	The application has facilities for automatically requesting Bluetooth functionality from the user and can react properly to an access refusal.	SQ1
F-2	Low	The UI must show user heath information and collected data.	The UI should be useful to the user, at least to a minor degree.	User health information and collected biometrics are shown in the UI.	Finding B, literature review.
F-3	High	The application must acquire permission for displaying notifications.	Notifications are important for actively informing the user on important system events.	The application has access to the notification infrastructure of the OS.	SQ4
F-4	High	The UI must have facilities for connecting the E4 to the system.	The system cannot collect data without the main data collection device.	The system can connect to the E4 and receive data.	SQ1
F-5	Low	The UI must display the connection status of the E4 device.	The user should not be required to actively check the connection to the E4 device, this information should be available at a glance.	The UI has options for displaying connectivity status.	SQ1
F-6	High	The UI must notify the user on changes in the connection status of the E4 device.	Again, the user should not be required to actively check the connection to the E4 device. The system should have the responsibility of data collection and request help from the user when needed.	The system has active UI elements that have access to high-level channels for notifying the user on connection states.	Issues of mBrain. Finding A, literature review.

F-7	High	The system must detect faults in the E4 device and inform the user.	An intermittent fault in the E4 firmware makes it intermittently unable to identify itself towards the Empatica servers. This is a soft failure and must be automatically detected and rectified.	The system can detect a fault in the connected device and guide the user on how the faults can be rectified.	Findings from earlier work. Finding C, literature review.
F-8	Low	The UI must display information on the general status of the E4 device, e.g., battery levels.	Battery charge status and other device metrics must be easily accessible by the user as the device has no physical display.	The UI has components for displaying status information on the E4.	Finding C, literature review.
F-9	High	The system must store tag data on user request.	Tags are very important for the functioning of the system as tag information is used to correlate real-life events with collected data.	Tags created by the user are shown in the UI.	SQ3
F-10	High	The UI must provide facilities for annotating tags.	Tag annotation is very important for describing real-life events for correlation with collected data.	The UI has an interface for annotating tags.	SQ3
F-11	High	The system must store all sensor data from sent form the E4.	Collecting and storing data from the E4 is the main premise for this project.	Data collected through the smartphone application can be retrieved from the backend database on request.	SQ1, SQ4
F-12	Low	The UI must give access to information on the user’s illness.	The application should be a source for information for the user.	The UI contains elements for displaying information on the various biometric variables, illness information, and management tips.	Finding B, literature review.

4.2.2 Interface requirements

As mentioned in subsection 3.3, interface requirements concern all edges of the system where parts or components interact with each other. UI, network, and database connections can all be categorized as system interface elements.

ID	Import.	Requirement	Rationale	Completion criteria	Source
I-1	High	The Empatica E4 must communicate with the application.	Communications between the E4 and the application is central for data collection.	The application can connect to the E4.	SQ1
I-2	High	The application must verify the E4 device against the Empatica servers.	The E4 needs to be authenticated towards servers at Empatica before the device can be connected for data collection.	The system automatically authenticates the device.	Empatica security and business model.
I-3	High	The application must receive data from the Empatica E4.	Data collection is one of the amin premises for this project.	Data can be streamed from the E4 to the application.	SQ1, SQ4
I-4	High	The application must receive user tags from the E4 device.	Tags are used to annotate the collected data and are essential for facilitating the ML process.	Tags sent from the E4 are collected and stored by the application.	SQ1, SQ4
I-5	High	The application must send collected data to the mSpider system.	Smartphones have limited data storage capacity. Data must be transferred to properly facilitate continuous data collection.	Data is transferred to the backend as soon as possible.	SQ4
I-6	High	The application must send tag data to the mSpider system.		Tags are transferred to the backend as soon as possible.	SQ4

4.2.3 Non-functional requirements

As mentioned in subsection 3.3, non-functional requirements cover all functionality that allows the system to function in its environment. This includes housekeeping tasks, security, and performance, all functions that are tied to the operation and infrastructure of the system.

ID	Import.	Requirement	Rationale	Completion criteria	Source
NF-1	High	The application must run on different models of smartphones.	The user should install the system on their own smartphone. This will help to make the system accessible.	The system supports a range of smartphone operating systems.	Findings from earlier work. Findings from mBrain.
NF-2	High	The UI must be interactable.	The user interface should be interesting to the user and promote user interaction.	The UI has interactable features besides login, logout, and device connection features.	Finding B, literature review.
NF-3	Medium	The system must be simple to use.	The basic functions of the system must be as simple as possible to accommodate all user types.	Connecting and disconnecting the E4 device should be very easy. Annotating tags should be very easy.	Finding B, literature review.
NF-4	Medium	The application must store persistent information until intended removal.	User settings, application state, and data collection states must be stored between user sessions.	The application must reset when the user logs out.	Findings from mBrain. SQ3
NF-5	Medium	The system should facilitate time-limited persistent login.	The user should not have to manually log in every time the application is started.	The system implements a login session solution for time-limited persistent login.	Security
NF-6	Low	The application must delete all persistent information when requested.	From a privacy perspective, users should be able to remove all information that ties them to the collected data.	The application must reset when the user logs out.	GDPR and security.
NF-7	Medium	The system should not have unreasonable power usage.	The system should not be an inconvenience for the user.	The system should increase the charging frequency of the user's smartphone by more than 50%.	SQ2 Findings from mBrain.
NF-8	Medium	The system should only use Wi-Fi networks for transmitting data.	The system should not inconvenience the user. The system should not cause the user monetary issues.	The system can detect when a Wi-Fi network is available.	-

NF-9	High	The system must store data at the highest possible resolution.	High data resolution will give the best possible environment for facilitating prediction.	Data captured by the E4 device is not compressed in any way before storage.	Finding from mBrain. Result D, literature review.
NF-10	Medium	The system UI should be prompt and responsive.	Users who regularly interact with modern information systems are accustomed to high responsive UIs.	User interaction with the UI should not take more than 1 second to take effect.	Common sense. Result B, literature review.
NF-11	High	The system must store data locally if no suitable internet connection is available.	Unavailable networks should not be a cause for data loss.	The device stores data on the device when a suitable network connection is not unavailable.	Result A, literature review. Findings from mBrain. SQ2
NF-12	High	The system must guarantee data atomicity between frontend and backend.	Connection issues or other network errors should be responsible for data loss.	No data is deleted from the frontend until a successful storage response is received from the backend.	SQ2
NF-13	High	The system must continuously collect data.	Continuous data collection is important to ensure high data collection coverage.	The system collects data from the participant as long as the E4 is connected.	Findings from earlier work. Result A, literature review. SQ1
NF-14	High	The Empatica service must automatically start after smartphone reboot.	The user must be reminded to continue the data collection after a reboot.	The system detects the boot-complete signal from the OS and can react to it.	SQ1, SQ2

4.2.4 Security and privacy requirements

As mentioned in subsection 3.3, security and privacy requirements cover functionality that ensures safe, secure, and trustable environments for collecting and storing potentially sensitive data, including secure communications, storage, and interfaces.

ID	Import.	Requirement	Rationale	Completion criteria	Source
S-1	High	The frontend system must facilitate user login.	Important for connecting the user alias with the collected data.	The system has login infrastructure that can take user input and authenticate the user towards the backend server.	GDPR and security.
S-2	High	The backend system must implement user accounts.	The backend system must have methods for connecting incoming data to individual users.	Users have individual accounts on the backend server. All data received from a user is stored for the same user.	GDPR and security.
S-3	High	The UI must provide user logout options.	The user must be able to log out of the system at any time.	The system has UI elements that can remove all user information.	GDPR and security.
S-4	High	The system must enable manual disconnection of the E4 device.	The user should have the option of manually disconnecting the E4 device through the UI at any time.	The UI has interactable components that can disconnect a connected device.	GDPR and security.
S-5	High	The backend system must store data anonymously.	Participant privacy is important, and data must be available on a per-participant basis.	Data stored on the backend server are only tied to participant pseudonyms.	GDPR and security.
S-6	High	Communications between the frontend and backend of the system must be secure.	Participant data must not be leaked to or tapered with by malicious third parties.	All data traffic between the backend and the frontend utilizes encrypted communication channels.	GDPR and security.

5 Design and implementation

The following subsections outline the design and implementation of the prototype system. Sections 5.1 through 5.3 show a design and component overview, subsection 5.4 shows the GUI layout and design, while subsection 5.5 contains a detailed component-wise description of the internal workings of the prototype system.

5.1 System overview

Figure 7 shows a graphical overview of the major components of the system. Arrows between components indicate communication lines. Dotted lines show central functionality that was not implemented for this prototype project but that is important for the long-term vision of the system. The system revolves around the user and the prototype application installed on the user's interface device (smartphone). The internal data processing of the prototype application is described in detail in subsection 5.3.

Major system components:

- Empatica E4 data collection device. Worn by the user. Actively pushes biometric data to the UI application when connected.
- UI application running on a smartphone. Main system component. Responsible for most of the important functionality in the system. Utilizes a persistent foreground thread for communications and data processing. Responsible for interfacing with the user.
- mSpider database. Main data storage component. Stores all collected data. Implements user account management. Will in future system iterations be responsible for calculating migraine attack predictions and for performing other statistical workloads.

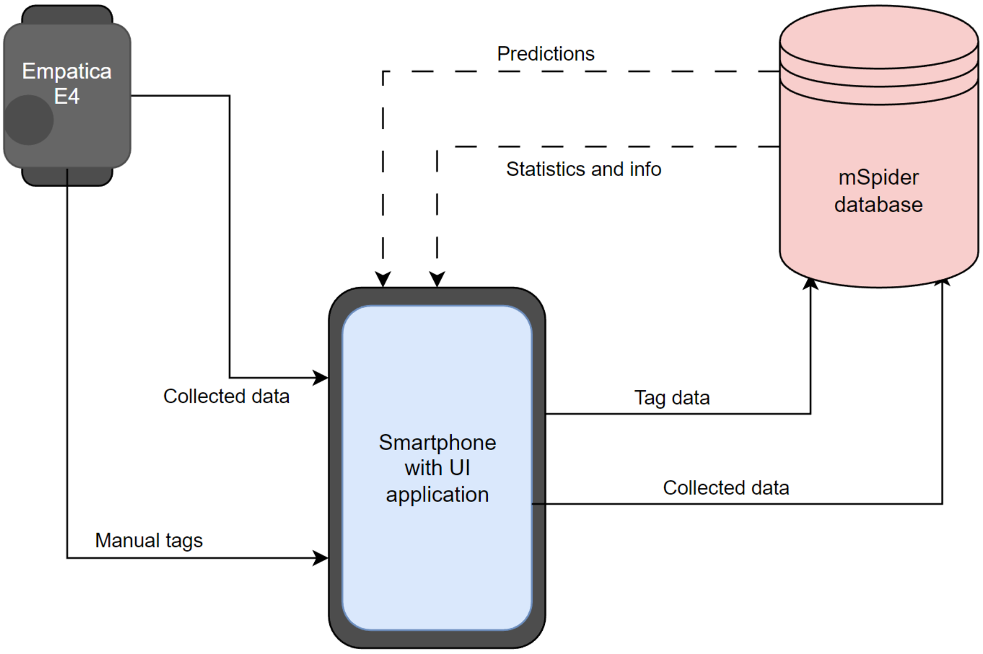


Figure 7: External system overview with major components

The data collection process starts with the Empatica E4 device. When the E4 is connected to the UI application data will automatically be collected and stored on the device. Manual tags are stored when this functionality is instigated by the user. The UI application will process and organize the collected data before it is sent to the mSpider database backend on fixed intervals. Data is stored on the backend server and utilized to estimate the current likelihood of future migraine attacks. In future system iterations, migraine forecasts will be automatically sent back to the Ui application where they will be presented to the user. The UI application can actively request statistics and additional information from the mSpider backend at any time. Communications between the E4 and the UI application are facilitated through a BLE connection, while communications between the UI application frontend and the mSpider backend are facilitated through an internet connection using HTTPs.

5.2 Application components

The internal component layout of the prototype application broadly follows the Android Guide to app architecture [31], where the functions of application are organized into layers based on shared functionality aspects. This allows for functionality to be accessible between layers by utilizing abstractions and interfaces. An overview of components and interconnectivity can be seen is figure 8. The application relies on multiple threads to ensure that important functionality always is available when needed. This is especially important for the Empalink interface.

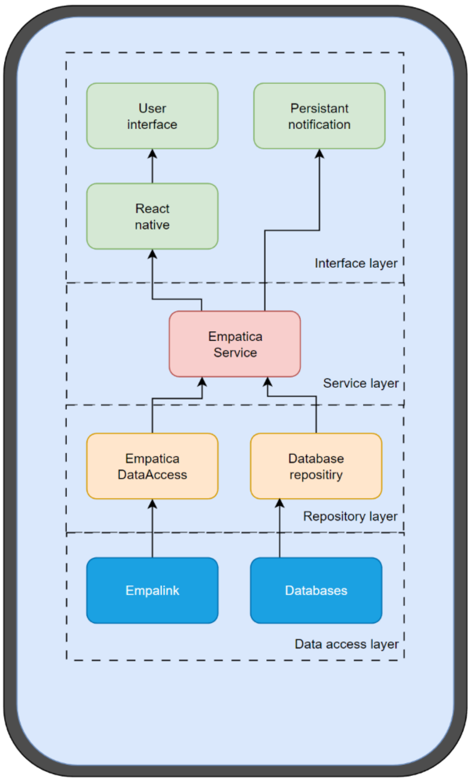


Figure 8: System layers and major components

5.2.1 Data access layer

As the name suggests, the data access layer handles access to data sources and storage. It consists of three main components:

- An instance of the Empalink service for facilitation communications with the E4 device.
- The main database for storing raw data captured from the E4.
- The state database for persistent storage of user settings, the application state, and structured data.

5.2.2 Repository layer

The repository layer works as an interface for the data access layer, and all communications with the data sources are controlled through the repository layer.

- The database repository is responsible for organizing and abstracting all database traffic. This allows for better handling of potential database issues, such as duplicate keys.

- The Empatica data-access component creates an interface with the Empalink service and handles all actions regarding the E4 device. This includes robust handling of device disconnects.

5.2.3 Service layer

The service layer is inhabited by the highly important Empatica service, which is arguably the main component of the application. The Empatica service is running as a unique and persistent foreground OS service, granting it continuous access to system resources, even when the application interface is minimized or closed. Most system components and subcomponents are instantiated through the Empatica service, which forces unique instantiations and gives all other components access to each other's functionality through the central service. The Empatica service is responsible for actively updating the UI and the persistent notification. It is also responsible for instigating the data aggregation thread and data sending thread on specific time rotations (see subsection 5.5.4), as well as actively pushing data to components in the interface layer.

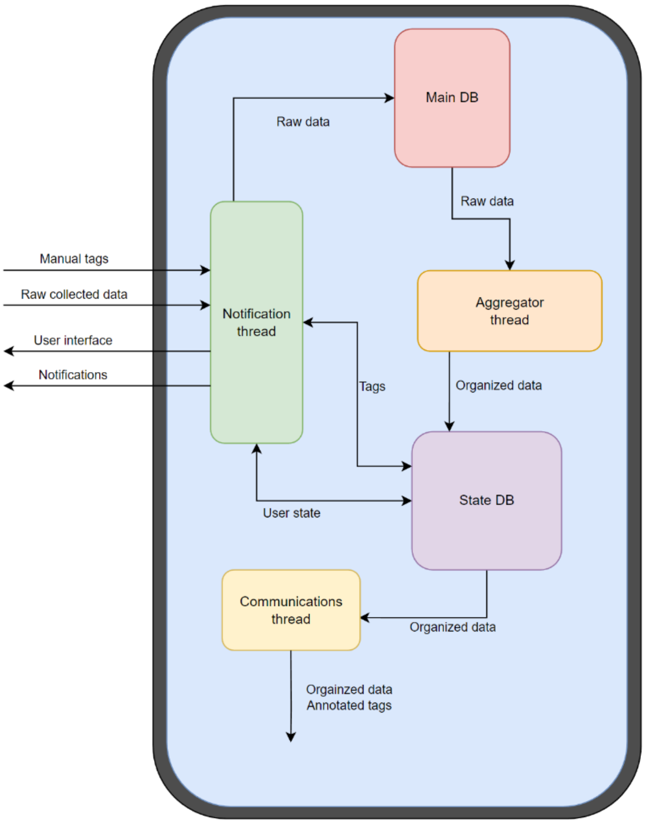


Figure 9: System dataflow diagram

5.2.4 Interface layer

The interface layer is responsible for all user contact. This is accomplished through a UI based on React-native [32] and through the OS notification system.

- The persistent notification is used for actively communicate the system data collection state with the user. The notification is tied to the context of the Empatica service, which makes it always available. Pressing the notification will activate the UI through the Empatica service.
- The React-native component is responsible for drawing the UI and managing user input. It passively receives data and change events from the Empatica service.
- The UI component is responsible for all user interaction (see subsection 5.4). This includes displaying data, managing user settings and preferences, and user login. It also provides facilities for annotating manual tags.

5.3 Data management

The application dataflow describes the process of transporting data from the collection device to the backend storage. Several components and threads are involved for storing, processing, restructuring, and transporting data. Although multiple biometric variables are captured, only two main datatypes are recognized by the system: automatically collected data and manually collected data. Auto-collected represents all data captured from the data collection device in real-time and stored directly in the main database. Manual data represents user tags, which are manually instigated by the user through the device interface. Both data types need processing before they can be transported to the backend server for permanent storage. As can be seen in figure 9, the data handling processes for both datatypes involve multiple steps.

Tag dataflow summary:

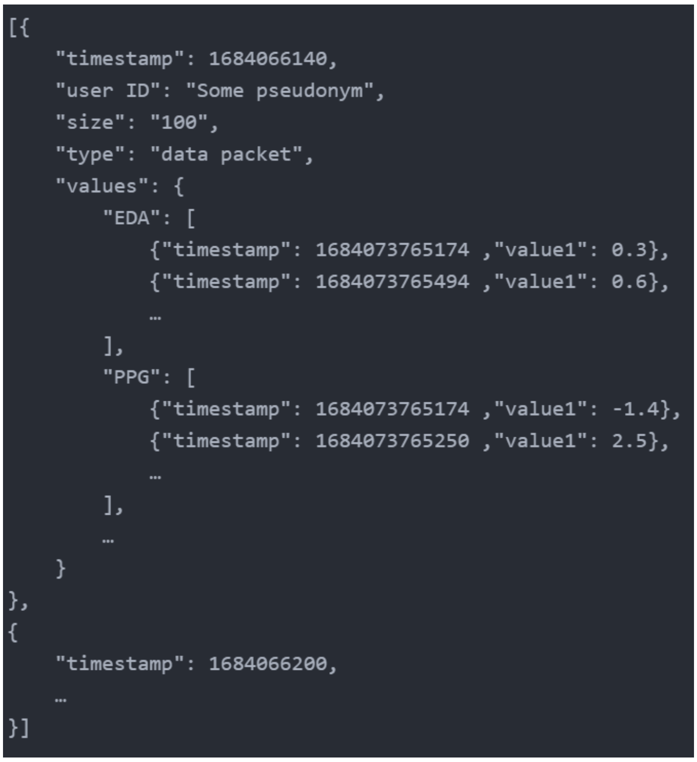
1. Tag manually instigated by the user.
2. Non-annotated tag is stored in state database.
3. All non-annotated tags in the state DB are actively pushed to the UI by the Empatica service, which runs on the notification thread.
4. The UI presents the user with annotation options.
5. The user annotates the tag and approves it.
6. The annotated tag is stored in the state DB.
7. The notification thread instigates a communications thread on fixed intervals.
8. The communication thread searches the state DB for annotated tags.
9. All annotated tags are sent to the mSpider backend for permanent storage.
10. Tags are deleted from the state DB when the communications thread receives positive confirmation from the backend server.

Dataflow of auto-collected data:

1. Biometric data event automatically instigated by the E4.
2. The notification thread stores the incoming data directly in the main database. All datapoints are stored with a millisecond timestamp and a value.
3. The notification thread instigates an aggregator thread on fixed intervals.
4. The aggregator thread fetches data from the main database and restructures them into data packets.
5. The aggregator thread stores the data packets in the state database.
6. All data points that have been aggregated into packets are deleted from the main DB.
7. The notification thread instigates a communications thread on fixed intervals.
8. The communication thread searches the state DB for data packets.
9. Data packets are sent individually to the mSpider backend for permanent storage.
10. Data packets are deleted from the state DB when the communications thread receives positive confirmation from the backend server.

Data packets

Data packets are JSON (JavaScript object notation) objects used in the system for organizing auto-collected data. Examples of data packets can be seen in figure 10. Raw data is received from the E4 data collection device in the form of a floating-point sensor reading value and a millisecond resolution Unix timestamp as a double-sized integer. This data format is not well-suited for being directly communicated to the backend server as the E4 produces ~100 individual readings per second. Additionally, large numbers of individual datapoints would require a data streaming approach with persistent connections, which would be non-trivial to manage and that could result in data loss. Instead, the system uses minute-packages where all data collected within the same minute are compile together, regardless of the number of datapoints. This created reasonably sized packets that are better suited for internet communications. The JSON data format is a good fit for this use-case, as it allows for additional metadata in the data packages. On top of this, this formatting allows for directly inserting the data packages into the mSpider database without restructuring.



```
[{
    "timestamp": 1684066140,
    "user ID": "Some pseudonym",
    "size": "100",
    "type": "data packet",
    "values": {
        "EDA": [
            {"timestamp": 1684073765174 ,"value1": 0.3},
            {"timestamp": 1684073765494 ,"value1": 0.6},
            …
        ],
        "PPG": [
            {"timestamp": 1684073765174 ,"value1": -1.4},
            {"timestamp": 1684073765250 ,"value1": 2.5},
            …
        ],
        …
    }
},
{
    "timestamp": 1684066200,
    …
}]
```

Figure 10: Example of a data packet in JSON format

5.4 GUI layout and components

As found in related works and the literature review, user involvement and interaction can be vital for strengthening data collection process. The main design goal of the prototype UI is to actively involve users in the data collection process when needed while at the same time keeping the required interactions to a minimum, thus moving the main responsibility for data collection from the user to the system. On top of this, the UI should be interesting and contain information that could be useful to the user.

5.4.1 Main interface

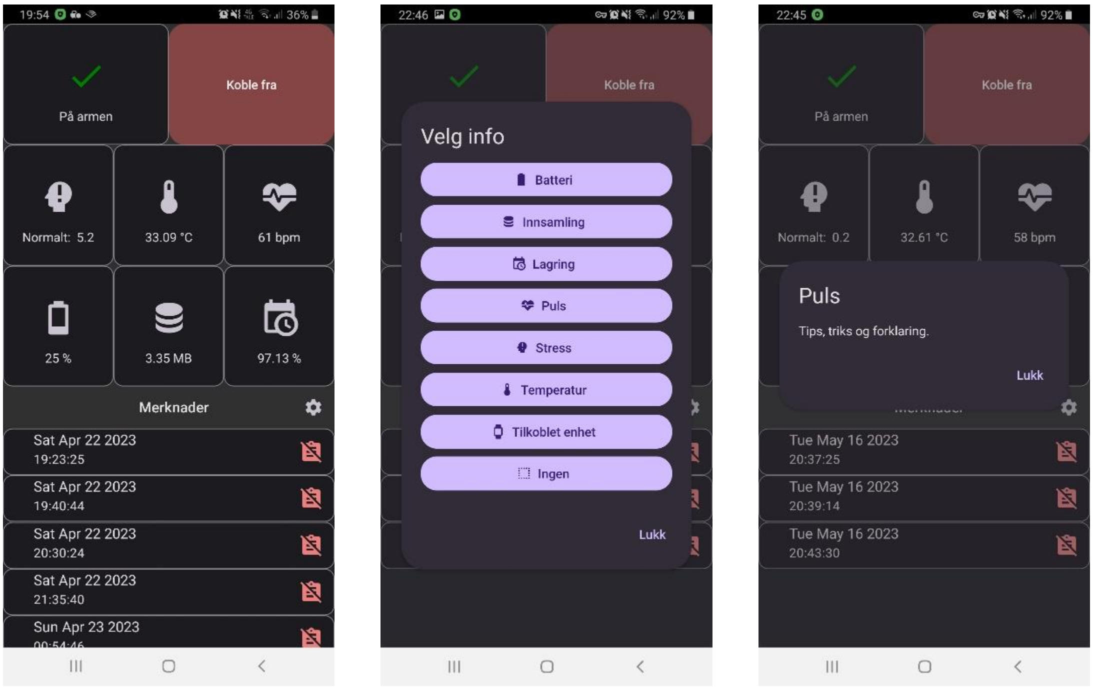


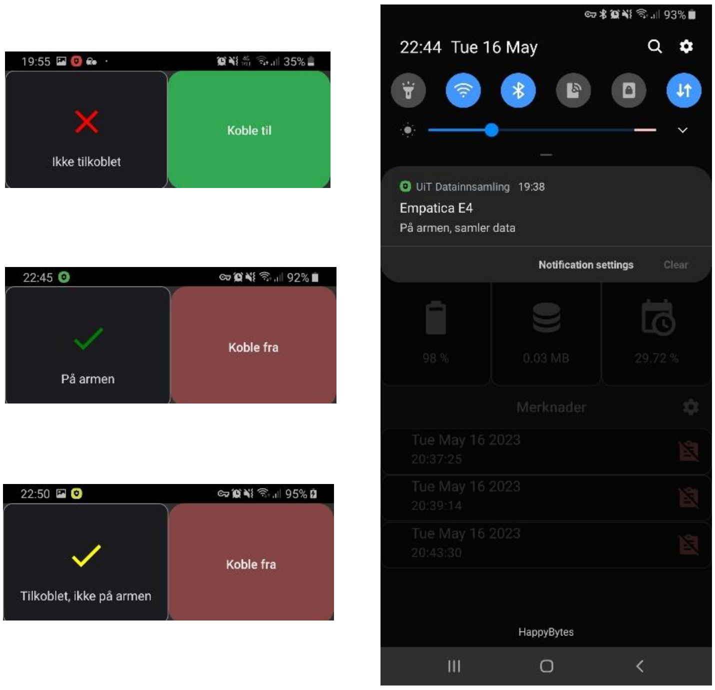
Figure 11: Main user interface (left), customization screen (middle), extra information (right)

The applications main interface (figure 11, left) is designed to be informative and compact. Most features and information are easily accessible through a single UI page, which should make the UI simple to understand and use. As can be seen in the left image of figure 11, the upper portion of the UI contains device connection status and buttons, which will change based on the connection status of the Empatica E4 device. This is covered in more detail in subsection 5.4.2.

The middle portion of the UI shows the information grid that displays live information on the data collection process and on the user's health. Each panel of the grid can be touched to show extended facts and explanations on the panel's information (figure 11, right). Additionally, the grid can be configured to display information based on user choice and the panel selection menu can be seen in the center image of figure 11. This modular design makes it easy to add more interface options if new functionality and metrics become available (see subsection 7.4.7, Empatica EmbracePlus).

The lower portion of the main UI is inhabited by the manual tag interface. All non-annotated tags created manually by the users are shown here. This is covered in more detail in subsection 5.4.3. This portion of the UI helps to fulfil requirements NF-3, F-2, NF-2, F-4, F-5, F-6, F-7, F-8, F-10, F-12, S-1, and S-4.

5.4.2 Connection states and notifications



UiT Datainnsamling 19:38
Empatica E4
På armen, samler data

Tue May 16 2023
20:37:25

Tue May 16 2023
20:39:14

Tue May 16 2023
20:43:30

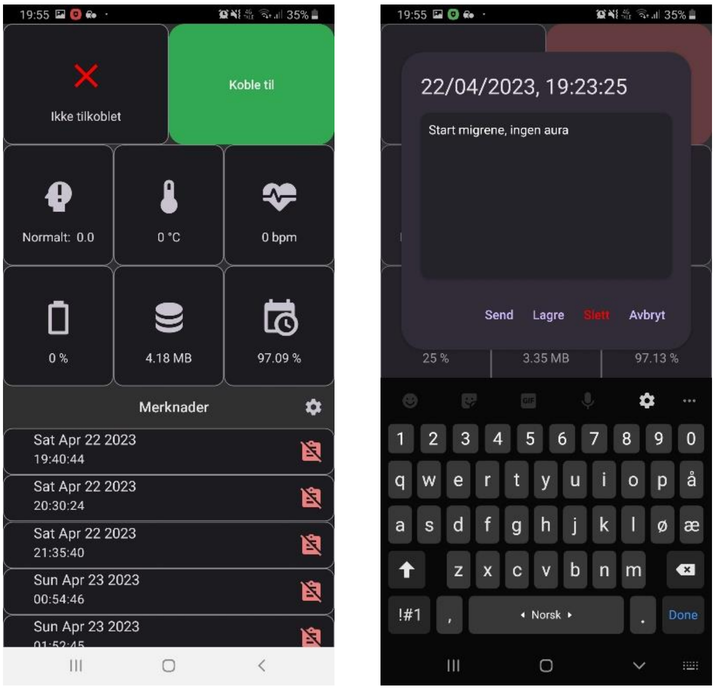
Figure 12: Connection states (left) and notification (right).

Most of the goals of this project are reliant on an effective and continuous connection between the E4 data collection device and the prototype application. There are three possible connection states (see figure 12, left) that are recognized by the system, where state 3 is the most desirable:

1. Device is not connected. No data collection.
2. Device connected, but not worn by the user. No data collection, although very simple to rectify by the user.
3. Device connected and worn by user. Device collects data.

Although it can detect each state, the system needs help from the user to change the states and restart the data collection process if states 2 or 1 are detected. To facilitate this, the system employs an in-application status display and a persistent system notification. Both follow the traffic light system, which is shown in the left images of figure 12, where colors red, green, and yellow are used to indicate device connection states. The same colors are used in the persistent notification that can be seen on the right image of figure 12. In the case of a E4 disconnect, the system detects the disconnect and refreshes the notification, which causes the OS to alert the user through noise or vibrations. Additionally, depending on settings and version of the Android OS, the persistent notification may also be visible on the lock-screen of the device. This allows the user to view the connection state between the E4 and the application at a glance while performing other tasks on their smartphone. This portion of the UI helps to fulfil requirements NF-3, F-4, F-5, F-6, F-7, and S-4.

5.4.3 Tags and annotations



Sat Apr 22 2023 19:40:44
Sat Apr 22 2023 20:30:24
Sat Apr 22 2023 21:35:40
Sun Apr 23 2023 00:54:46
Sun Apr 23 2023 01:52:45

22/04/2023, 19:23:25

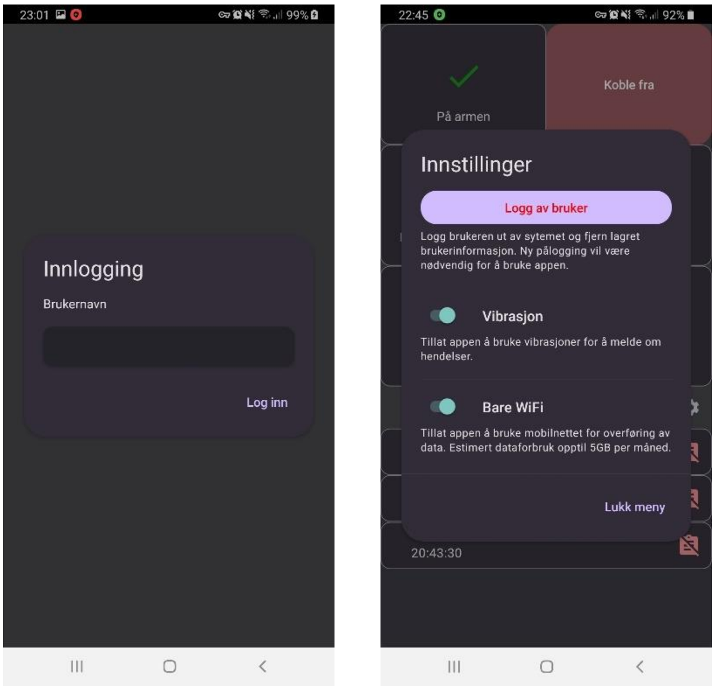
Start migrene, ingen aura

Figure 13: List of tags in main interface (bottom left) and tag annotation interface (right)

As mentioned earlier, the single input button on Empatica E4 can be clicked by users to create timestamped tags. All non-annotated tags in the system will be displayed in the lower portion of the main interface, as can be seen in the left image of figure 13. Tags are very important for migraine prediction as the represent the connection between the collected data and the real-life events experienced by the user. Raw tags from the E4 are only represented by a timestamp and they must be manually annotated by the user before they can be used in the statistics and ML processes. The right image of figure 13 shows the tag annotation page of the GUI where free-text information may be added to timestamped tags. The user must approve all tags before they are sent to the backend server. Tags may also be deleted or partially annotated for later completion.

This portion of the UI helps to fulfil requirements NF-3, NF-2, F-9, and F-10.

5.4.4 System login and settings



Logg brukeren ut av sytemet og fjern lagret brukerinformasjon. Ny pålogging vil være nødvendig for å bruke appen.

Vibrasjon

Tillat appen å bruke vibrasjoner for å melde om hendelser.

Bare WiFi

Tillat appen å bruke mobilnettet for overføring av data. Estimert dataforbruk opptil 5GB per måned.

Figure 14: Login screen (left) and setting (right)

The prototype system utilizes a simple login mechanism for authenticating individual users. The login screen can be seen in the left image of figure 14. The full authentication process is described in subsection 5.5.6. Each user of the system issued a unique username that is used for identification and that is entered at login, and all collected data will be marked with the registered username before it is sent to the backend server.

The UI settings screen, which can be seen on the right side of figure 14, presents the user with options on how the application will handle certain situations. In this prototype, only a small number of options are available, including a logout option for the current user. The logout button will remove the user login credentials and reset all settings and metrics stored in the application. User data stored on the device will be uploaded to the backend server before it is deleted. Application network access policy may be toggled in the settings, potentially giving the application permission to utilize the cellular networks for data upload. This option is set to “Only Wi-Fi” by default as utilizing cellular networks may be expensive for the user. Vibration setting toggles the usage of vibrations when the application sends notifications. This portion of the UI helps to fulfil requirements NF-3, NF-2, S-1, S-3, S-4, NF-8.

5.5 Implementation details and components

This subsection details the implementation process of the prototype frontend application and its backend data storage solution in mSpider. Each component, subsystem, and feature of the system, including their functions and methods, will be described. The requirements fulfilled by each component will be listed at the end of each description.

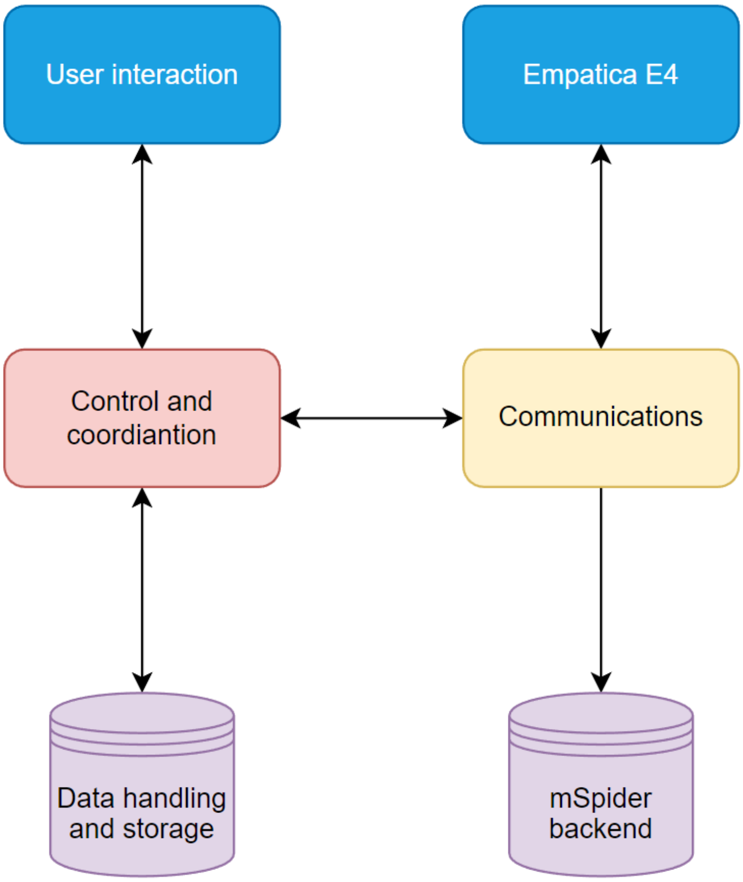


Figure 15: Diagram of major system components

5.5.1 Limitations

Because of cross-platform options of the Empatica SDK, the prototype system was initially intended to be available for both Android and iOS systems. This was the main reason for using React-Native as the application's UI engine. However, complications with iOS software architecture and Apple's approach to background application execution made a dual platform prototype impossible within the timeframe of the project. This limits the prototype to exclusively supporting Android devices.

5.5.2 User interaction

Empatica module

The Empatica module is a React-native native Android module created for this project and functions as the system's bridge between React-Native and the Android-specific parts of the program. It uses events and call-backs for communicating between the compiled Java code and the UI, and operates within the React context (UI thread) of the application (see figure 16). Information to be displayed by the UI is actively pushed through this module and communicated to UI by named events. The Empatica module can be considered as a part of the UI and implements functionality for fulfilling requirements of the UI components listed earlier.

React-Native UI

The React-native UI component is based on tutorials and instructions from the react native website. At this stage in the development, the UI included only a device connection button and information on E4 device connection status. Apart from UI elements, this component implements an Android native module named *EmpaticaPackage* to facilitate compilation of the native Android code and to enable events from the compiled code to be detected by the UI. This component implements functionality for fulfilling requirements F-4, F-5, and S-4.

Tags and annotation

Connected E4 devices will send timestamp tags to the application when the input button on the hardware is pressed. Tags from the E4 consist only consist of a single Unix timestamp, which is stored in the *state database* as a *TagItem* object. The system detects the arrival of a new tag and automatically initiates an update of the list of tags in the UI, making the new tag available for annotation by the user. Tags can be annotated via the tag annotation screen in the UI. Annotated and approved tags are automatically sent to the backend server and deleted from the local database. The tag functionality helps to fulfil requirements I-4, I-6, F-9, and F-10.

5.5.3 Communications

Empatica data access

The Empatica data access component is based on sample code from the Empatica SDK package and utilizes the Empalink library for implementing full communications between the Empatica E4 and the application, including infrastructure for Bluetooth connections and data transfer. It is mainly responsible for connecting to E4 devices, receiving data from the connected device, and monitoring device connectivity. For connecting to an E4 device, the system must authenticate the E4 model towards the Empatica servers. This is automatically performed by the SDK, although the connecting E4 device must be registered beforehand through an Empatica development account, and the accounts API key must be included in the authentication request.

Collected data is automatically pushed from the connected device through an interface in the SDK code. For receiving the data stream, the Empatica data access component implements a series of class methods that are callable by the SDK through the interface. When one of the class methods is called, it receives data from the E4 as input parameters, which can then be archived or used internally by the component class.

Device connectivity and connection statues are managed by the Empatica SDK through the same abovementioned interface. When the connection status of the connected device changes, specialized methods are called in the component class with the new device state as an input parameter. The component can then react to this information and instigate status-change events. The Empatica data access component implements functionality for fulfilling requirements I-1. I-2, I-3, F-4, F-5, NF-4, and S-4.

Permissions

In the Android OS environment, applications and services need permissions from either the user or the OS for accessing various device services and functionalities. The prototype application needs permissions for multiple functionalities, which may vary based on Android OS version. The UI through React-Native is responsible for acquiring functionality that can only be accessed through user permission. Bluetooth is necessary for communications between the E4 and the application.

The Android OS classifies Bluetooth access as potentially dangerous, and the user must give permission for this functionality to be available for the application. Various Android versions can require different permissions for Bluetooth access. For instance, on Android 11 and below, Bluetooth access can be gained through permission for *ACCESS_FINE_LOCATION*, while Android 12 and above require permissions for the specific usage for Bluetooth, such as *BLUETOOTH_ADMIN* or *BLUETOOTH_SCAN*. In addition to Bluetooth, the application needs access to *POST_NOTIFICATION* for posting and handling persistent notifications, and it needs to have the ability to listen for the *BOOT_COMPLETED* event to facilitate automatic service startup on OS boot completed. Both these functionalities are non-critical and can be declared to be used rather than asked to be used. The permission handling implements functionality for fulfilling requirements NF-1, F-3, F-1, NF-14, and I-1.

Network communicator

The network communicator component is responsible for all communications between the frontend application and the backend server. When instigated by the controller thread, the communicator pushed all approved data in the state database to the server. All communications are performed asynchronously by worker threads, keeping the application responsive and available to the while the sending process is conducted. Each thread checks the current network policy, as mentioned in section 5.4.4, before sending data. Data atomicity is retained by deleting data only if a positive *201* HTTP status message [33] is returned from the server, which indicates backend storage success. Data is sent as individual packages or tags, making the sending process very flexible and robust. As described earlier, JSON web tokens are used as session identification for the user. When the communicator send data to the server, the user's JWT accompanies the data and is used to identify the user. The communication component helps to fulfil requirements I-5, I-6, NF-8, NF-10, NF-11, NF-12, and S-6.

5.5.4 Control and coordination

Empatica service

The *EmpaticaService* is the central component of the prototype system. It is responsible for most of the "under-the-hood" features and servers as a hub for instantiating other subsystems, overseeing control flow, and for instigating timed events. The component consists of a persistent Android OS service tied to a permanent non-dismissible system notification. It executes in the Android OS foreground which grants it continuous and immediate access to system resources such as processing time and interrupts. This is not only important for making the application responsive and flexible, but also central for facilitating uninterrupted data streaming from a connected E4 device. The *EmpaticaService* will always run as long as

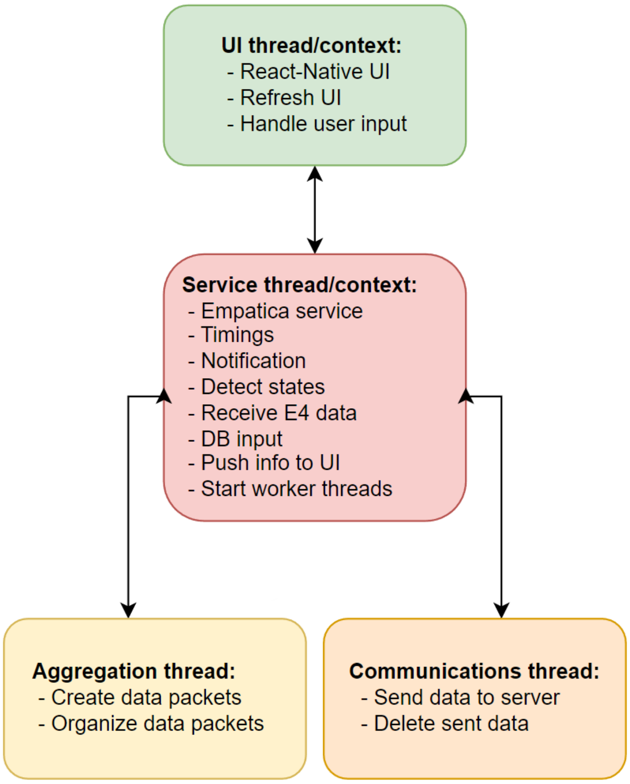


Figure 16: System contexts and threads

the application is installed on the Android system, disregarding situations where the user kills the application processes through OS interfaces.

The *EmpaticaService* component executes on a separate context (see figure 16) to the UI, which enables the user to dismiss the GUI part of the application without halting the data collection process. Additionally, this enforces data collection continuity and separates user interaction from the data collection process, which could be important, as slow and irregular data acquisition infrastructure may bottleneck the data collection frequency of the E4.

Timed events are used in the system for performing tasks on specific time intervals. All timings are controlled by the *EmpaticaService,* which periodically initiates data aggregation processes, data upload to the backend, and status updates in the GUI at pre-defined time intervals. The long-lasting workloads of data aggregation and data upload are handled by separate worker thread, while the GUI updates are performed directly by the service. The *EmpaticaService* component helps to fulfil requirements F-4, F-5, F-7, F-11, I-1, I-3, I-5, I-6, NF-7, NF-9, NF-10, NF-13.

Notification system

The notification system is the active link between the user and the prototype system. As described earlier, the notification system detects the E4 connection state in the system and seeks user assistance to alter these states. The notification system uses the infrastructure of the Android OS for facilitating active communications with the user and utilizes a traffic light system to passively inform on system states. Although notifications are closely connected to the *EmpaticaService* and exists in the same context, it is technically a part of the UI and touching the persistent notification will start the user interface part of the application. The notification system helps to fulfil requirements F-1, NF-2, F-5, F-6, F-7, F-8, NF-10.

5.5.5 Data handling

Databases

The prototype system has two databases: the *main database* for unorganized storage of data directly from the E4, and the *state database* for storing organized data, current system state, and user preferences. Both databases are tied to the service context of the application, and they are stored in the Android OS filesystem, which makes them persistent through smartphone reboots, potential application crashes, and application shutdowns. Utilizing separate data bases for state and raw data storage minimizes the risk of bottlenecking the data acquisition process from the E4. All datapoints sent from the E4 arrive as floating-point numbers accompanied by a millisecond-resolution Unix timestamp based on timings from the E4 device. Individual datapoints are directly stored in the *main database* on arrival, although annotated with a datatype descriptor. The main database contains only a single table with raw data. On the other hand, the *state database* the contains multiple tables and is frequently accessed by several components of the system. System status data, status data from the E4, persistent login information, as well as settings and configurations made through the UI by the user are all stored in or communicated through this database, making it very important for both persistent storage and internal communications. Additionally, the *state database* is responsible for storing aggregated data packets as well as storing and providing access to user tags for the tag annotation system. The database system helps to fulfil requirements NF-4, NF-5, NF-6, NF-9, NF-10, NF-13, F-2, F-8, F-9, and F-11.

Data aggregation system

The data aggregation system creates organized data packets from unorganized data captured through the Empatica E4 device. The aggregation process is automatically initiated by the *EmpaticaService* on fixed intervals and runs asynchronously. As described in section 5.3.1, data packets are used in the system to reduce complexity in the handling, storage, and retrieval of data. A data packet covers a full minute of collected data, where all datapoints collected with the same minute are stored in a JSON object for simple retrieval (see figure 10). Each datatype collected by the E4 are in the JSON object represented by a list of timestamps and variables. This approach is created to be flexible, as future Empatica devices may support additional biometric variables. The aggregation system helps to fulfil requirements NF-9 and NF-12.

5.5.6 Backend

mSpider API

As described earlier, the different modules of mSpider are primarily designed for actively pulling data from device-specific or provider-specific online data storage solutions at fixed intervals. As the prototype data collection system has no intermediate data storage solution, flexible API functionality was added to mSpider to accommodate passive data acquisition. The API developed for the prototype data collection system is rudimentary and only has support for user login, user authentication, and data upload. The backend API helps to fulfil requirements I-5, I-6, S-6, and NF-12.

User accounts and authentication

User or participant accounts are uniformly handled across all provider modules of the mSpider system. Login is facilitated through an authentication endpoint where incoming login credentials are exchanged with a session token (JWT), which can be user later for authentication when uploading data. Session tokens have time limits of 7 days and tokens may be refreshed at any time during this period. The prototype application refreshes the session tokens every time data uploading is initiated. The backend user account system helps to fulfil requirements S-2, S-5, and NF-5.

Data handling

Most of the data handling is performed by the frontend systems. When data arrives through the backend API, it is correctly formatted and can be directly inserted into the database. The JSON objects used to send data are annotated with metainformation which describes the contents and type if the incoming data. Data packets and tags are sent as identical objects, marked with datatype for distinction, and they are stored in the same table in the database.

6 Results

6.1 Literature review results

6.1.1 Review execution

The review follows the PRISMA (Preferred Reporting Items for Systematic Reviews and Meta-Analyses) [34] review guidelines.

This systematized literature review was performed using the PRISMA reviewing method, although duplicate screening was performed at the end of the review process due to formatting errors. Final searches and record retrievals were performed between 30. April and 02. May 2023. The review process is visualized in figure 17.

A total of 444 records were identified through initial searches in the mentioned databases and an additional 13 records were from other sources. This makes for a total of 457 records for initial screening.

The initial screening process removed records based on title, abstract and access. 70 records were automatically removed because of restricted access to free text articles. 11 records were removed because they were based on ongoing or proposed projects and did not show results. 18 records were removed because they referred published books or full conference proceedings. Records were then removed based on detailed scrutiny of experiment nature, data collection methods, and data collection discussions. 74 records were removed because they did not use the Empatica E4 directly for data collection, which included studies that used already existing datasets, studies that used other devices, and studies that did not perform data collection.

In the main screening procedure, 284 records were removed because they did not properly discuss their data collection processes. This included data collection under very controlled conditions (lab experiments), data collection over very short (<1h) time periods, and data collection with only a single biometric variable (e.g. only ACC). This screening processes resulted in 16 eligible articles. A summary of the screening process can be seen in figure 17. Each of the final articles were scrutinized and data on their usage of the E4 and their experiences with data collection were extracted. This was not an exact process as the individual articles have very different approaches to describing and discussing their data collection processes.

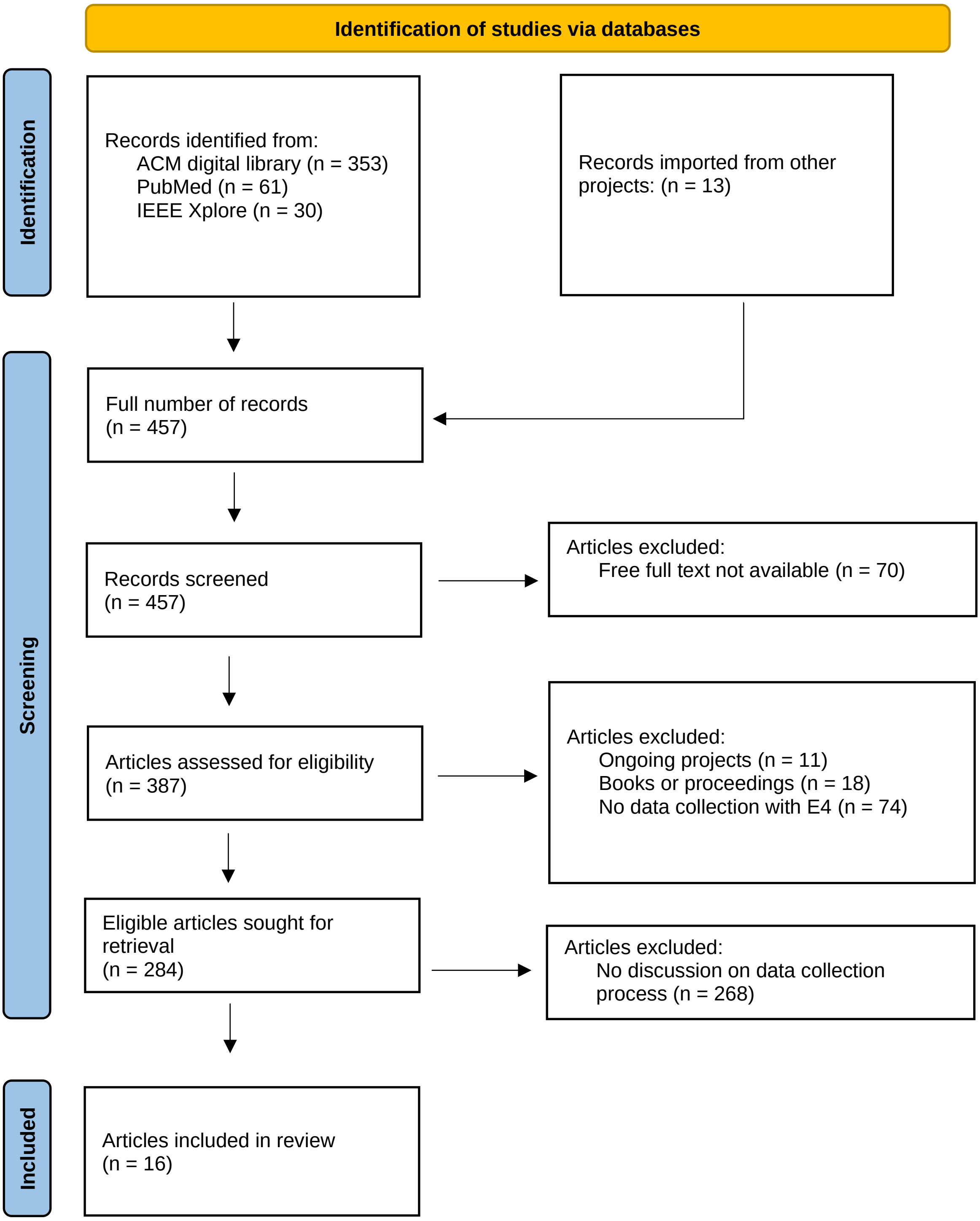

Identification of studies via databases
Identification
Screening
Included
Records identified from:
ACM digital library (n = 353)
PubMed (n = 61)
IEEE Xplore (n = 30)
Records imported from other projects: (n = 13)
Full number of records (n = 457)
Records screened (n = 457)
Articles excluded:
Free full text not available (n = 70)
Articles assessed for eligibility (n = 387)
Articles excluded:
Ongoing projects (n = 11)
Books or proceedings (n = 18)
No data collection with E4 (n = 74)
Eligible articles sought for retrieval (n = 284)
Articles excluded:
No discussion on data collection process (n = 268)
Articles included in review (n = 16)

Figure 17: PRISMA diagram of review process

6.1.2 Table of extracted data

Table 4: Results of literature review

#	Source	Methods, abridged	Relevant findings	E4 mode
[35]	*Ahmadi et al. 2022*, "Quantifying Occupational Stress in Intensive Care Unit Nurses: An Applied Naturalistic Study of Correlations Among Stress, Heart Rate, Electrodermal Activity, and Skin Temperature"	Correlates biometric variables with stress in intensive care nurses. Collected eye training data and EDA (E4) over ~12h shifts.	E4 shows reduced EDA data collection performance in "wet" environments. Claims data loss due to low device reliability.	Data storage
[36]	*Ajayi et al. 2021*, "Mobile Health-Collected Biophysical Markers in Children with Serious Illness-Related Pain"	Evaluates feasibility of using wearables for monitoring vital signs in child population (7-20y) with serious illnesses. E4 used to collect data over 2x24h for each patient.	HRV data not reliable enough to be used in study due to high activity artefacts. User tag input (button press) seldom correlates with the tagged event.	Data storage
[37]	*Amores et al. 2023*, "Olfactory Wearables for Mobile Targeted Memory Reactivation"	Uses olfactory wearable to increase learning and memory recall. Uses E4 to continuously monitor participant biometrics over multiple individual 24h sessions.	Reports connection issues with the E4. Biometric data lost for 12 of 30 participants during experiment sessions, data lost for 19 of 30 participants during sleep.	Streaming
[38]	*Barrios et al. 2019*, "Evaluating the accuracy of heart rate sensors based on photoplethysmography for in-the-wild analysis"	Evaluation of multiple wearables for data collection (IBI, HRV) during various activities and intensities. Data collected in experiments under laboratory conditions.	Shows that the E4 is significantly worse at collecting data during activity than other medically approved devices. Shows significant falloff in IBI data collection rate (>90%) between resting and walking.	Data storage

[39]	*Gao et al. 2022,* “Individual and Group-wise Classroom Seating Experience: Effects on Student Engagement in Different Courses”	Investigates if classroom seating patterns correlate with perceived and measured student engagement. E4 used to measure EDA.	EDA data quality highly reliant on factors like skin hydration, air humidity, and sweat buildup. The E4s EDA data collection accuracy is limited compared to lab equipment.	Data storage
[40]	*Gao et al. 2020,* “n-Gage: Predicting in-class Emotional, Behavioural and Cognitive Engagement in the Wild”	Measures engagement and emotional state of adolescent students during class. E4 biometric data correlated with participant surveys. Real-life conditions.	Reports >32% data loss due to device and participant errors. Biggest sources of errors were badly fitted devices and devices being accidentally turned off.	Data storage
[41]	*Gashi et al. 2020,* “Detection of Artifacts in Ambulatory Electrodermal Activity Data”	Analysis of ambulatory everyday data collection in healthy adult population of 13 participants. E4 is used to continuously collect all available data during waking hours for 36 days.	Reports a total of 2260 hours of collected data for 13 participants over 36 unique days. Provides data collection “heat-map”.	Data storage

[42]	*Kleiman et al. 2019,* "Using Wearable Physiological Monitors With Suicidal Adolescent Inpatients: Feasibility and Acceptability Study"	Evaluates the feasibility of continuous psychological monitoring in using biometric data. E4 is used to collect 24h biometric data form 50 suicidal adolescent patients. Button presses used for annotation. Two E4 units per patient, no charging downtime. Technical support staff available.	Very high ratio of collection days (days with non-zero collection, 95%). Average of 18.3 collection hours per day. Only 1 out of 216 button presses was accidental.	Data storage
[43]	*Larradet et al. 2019,* "Appraisal theory-based mobile app for physiological data collection and labelling in the wild"	Novel application for collecting and labeling physiological data in everyday settings. Uses ML model trained on previously collected data to detect emotional states in real-time.	Reports poor quality in data collected under ambulatory conditions using the E4, which results in poor recognition success rate of emotional states. Uses active notification system to ensure continuous data collection during app trial.	Streaming
[44]	*Nasseri et al. 2020,* "Signal quality and patient experience with wearable devices for epilepsy management"	Evaluates several devices (including the E4) in everyday data collection for use in epilepsy research.	Reports that more than 71% of all field-collected E4 data was of acceptable quality.	Data storage

[45]	*Ravindran et al. 2022,* “Evaluating the Empatica E4 Derived Heart Rate and Heart Rate Variability Measures in Older Men and Women”	Evaluates the E4 as a data collection device for collecting biometric data during sleep in an older participant population.	Concludes that the E4 is a suitable device for collecting data during sleep. Finds that 15% of IBI data was missing, and that could not be correlated with movement.	Data storage
[46]	*Sabeti et al. 2019,* “Learning Using Concave and Convex Kernels: Applications in Predicting Quality of Sleep and Level of Fatigue in Fibromyalgia”	Correlates pain episodes with poor sleep quality in an adult participant population. Uses the E4 for collecting sleep data from patients under everyday conditions.	Missing 21 out of 140 possible nights of data. On top of this, ~20% of HR/IBI data is lost due to noise.	Data storage
[47]	*Schmidt M. et al. 2019,* “Assessing Objective Indicators of Users' Cognitive Load During Proactive In-Car Dialogs”	Tries to measure changes in cognitive load based on biometric variables. Experiments on impact of using voice assistants (Siri, Alexa, etc.) while driving in a simulator. E4 used to capture biometrics.	Finds that E4 EDA measurements are overall too noisy to be used in these experiments.	Streaming
[48]	*Schmidt P. et al. 2019,* “Multi-target affect detection in the wild: an exploratory study”	Tries to correlate emotional states with biometric data. E4 is used for data capture during waking hours, questionaries are used as ground truth.	Collects >1400h of data from 11 patients over ~16 days. Data collected in under everyday conditions are intrinsically noisy. Manual labeling of is unreliable.	Data storage

[49]	*Van Voorhees et al. 2022,* “Ambulatory Heart Rate Variability Monitoring: Comparisons Between the Empatica E4 Wristband and Holter Electrocardiogram”	Compares the Holter heart monitoring device against the E4 for monitoring HRV data on 13 adult participants over 24h under everyday conditions.	The E4 is not well suited for HRV measurements in everyday conditions. E4 data lost for 4 participants, partly because of device charging issues. Experiences high data loss in HRV values. Concludes that the E4 is probably best suited for long-term data collection.	Data storage
[50]	*Vila et al. 2021,* “Real-Time Quality Index to Control Data Loss in Real-Life Cardiac Monitoring Applications”	Data collected during waking hours under everyday conditions from 3 adult participants over 7 days.	PPG data noise closely tied to ACC magnitude. Collected 124h of data over 11 total recording days.	Data storage

6.1.3 Evaluation of review results

As mentioned in subsection 3.1.4, most of the scrutinized articles include no discussion on data collection experiences (n=275), although their data collection methods and data collection results are normally highly detailed. The main cause for this in many of the excluded articles is that lost or missed data is naturally less interesting or useful than the data that was actually collected, and there is no incentive to include metadata or statistics on e.g., data collection coverage. Another probable cause is simply that no problems were encountered, which is arguably the case for studies that perform laboratory experiments where the collection intervals are short, the processes are tightly monitored, and technical staff is on-hand for fixing issues. However, a small number of articles (n=16) do include brief descriptions of their data collection experiences. Table 4 lists and details the articles that have either explicitly or implicitly commented on data collection.

Data quality issues

Many of the selected articles [35]–[38], [44]–[46], [49] report periods of low biometric data quality when using the E4. Most of the issues [36]–[38], [44], [46] are regarding IBI, HR or HRV measurements, all secondary values that are automatically derived by the E4 based on the raw PPG sensor data. A well-known issue with PPG sensor hardware is that it reacts badly to movement while the sensor is active, which can lead to noisy and erratic data, which again leads to patchy derived data coverage. This issue is stated by Empatica in the E4 datasheet [51], and it is also shown in [38], where the E4s ability to derive IBI values from PPG data falls by 90% under mild activities. However, correlations between ACC data and PPG data are mentioned by [50], hinting at the possibility of normalizing noisy PPG data against the ACC. On the other hand, findings from [45], [46] that show data loss from unmoving, sleeping participants, which could indicate that some of the data loss and noise in PPG and derived data may stem from other causes than movement and may be hard to normalize.

Several articles report a broader spectrum of reduced data quality, spanning all of the biometric variables recorded by the E4 [37]–[39], [43], [44], [47], [49]. Articles [37], [43], [47] indicated that most of the noise and data quality loss can be tied to device movement. This could be tied directly to compromises in the design of the E4, as [38], [39] conclude that the E4 is less accurate and more prone to noise than more specialized medical equipment. On the other hand, [44] finds that more than 70% of their collected data from a 10-day collection period were of acceptable quality. This ties in with [49], who experienced high noise levels and significant data loss in a short-term experiment and concludes that the E4 is probably best suited for data collection over long time periods, where sheer data volume could mitigate data loss.

All these findings could indicate that increasing the amount of collected data could be a viable solution to the data quality issues.

Data collection issues tied to hardware or software

Articles [35], [39], [40], [49] report problems with the E4 hardware or its accompanying software solutions. The construction quality of the E4 hardware is criticized by [35] and is deemed unsuited for wet and hectic workplaces. This is also mentioned by [39], where the EDA measurements of the E4 become highly unreliable in wet or sweaty conditions. Article [35] also reports that they experienced complete device failures during their data collection involving nurses at a hospital.

Data collection issues tied to participants

Data loss is high for studies that mainly rely on participants for data collection without giving immediate feedback. This is especially true in studies where the E4 is used in data storage mode, and where data collection happens under everyday conditions and may collide with the participants daily life. This can be seen in studies [41], [48], [50] where participants have been tasked with continuously wearing the devices and recording data during waking hours. If waking hours can be estimated to 16 of 24 hours, allowing for 8 hours of sleep, the mentioned studies have data collection coverages of 31%, 50% and 25% respectably. Although it can be very hard to speculate on the causes of the low collection coverage shown in [41], [48], [50], some of the data loss may stem from low participant engagement, as it is likely that the data collection could have been deprioritized for other daily activities. This notion is strengthened by the findings from [42] that showed a collection coverage of more than 76% for participants in controlled conditions and with access to support personnel. On top of this, findings from *Ursin* [14], which also was conducted with access to support personnel, showed a data collection coverage of approximately 70% under everyday conditions. Articles [40], [49] report data quality reduction and data loss tied to badly fitted devices and mentions that participants need to be thoroughly instructed on how the device is to be used.

Data collection issues tied to the E4 in streaming mode were reported in [37], which relies on a purpose-built smartphone application for data collection. Participants were tasked with collecting biometric data during sleep under everyday conditions, which resulted in a data coverage of only 36% due to lost connections. Although connection errors may be tied to malfunctions in the application, it may also be caused by lack of participant motivation for rectifying system errors and continuing the data collection. This is seconded by the findings of [43], which purpose-built application included systems for increasing participant involvement, like notifying the participant about disconnects or malfunctions. No significant data loss is reported by [43].

All these findings could indicate that “black-box” and pure data collection studies can have negative effects the quantity of data collected for medical research, and that participant involvement and motivation could be important for strengthening the data collection process.

6.1.4 Review result summary

The findings of this review pinpoint some important issues and failures with the E4 that have been encountered by the research community.

A. Increase data volume. High data volume may mitigate data noise and loss.
B. Actively involve the user in the data collection process. This can increase data volume.
C. Thoroughly instruct the user on how to use the data collection system and give them access to technical assistance when needed. This should also include information on the known limitations of the data collection system.
D. Data pre-processing can mitigate some data noise. Data should be stored as raw as possible to allow for this.

6.2 Performance testing results

This section shows a metadata analysis of the performance of the prototype system. All data collected by the author under optimal conditions. Performance metrics of the E4 device have not been measured.

Table 5: System battery usage

Average 8h battery usage, baseline	10%
Average 8h battery usage, application installed, no data collection	11% ± 1%
Average 8h battery usage, application installed, active data collection	58% ± 1%

Table 5 shows average battery usage based no 8h of operating time, each based on 3 samples. Table shows reductions in battery percentages during measurement period. Lower is better. Smartphone fully charged before each measurement.

Table 6: System memory usage

Average memory usage over 1h in from Android studio profiler	39 MB
Average memory usage over 3h reported by Android OS	38 MB

Table 6 shows average memory usage as measured by the Android studio profiler and by the Android OS on the utilized smartphone.

Table 7: System data storage and upload sizes

Average data storage increases per hour over 5 consecutive hours	25.5 MB ± 0.5 MB
Average data upload per hour as measured by frontend application	25.6 MB ± 0.5 MB
Average data storage increases per hour measured by mSpider backend	25.5 MB ± 0.5 MB
Average size of data packet (1 minute)	425 KB

Table 7 shows average data storage and upload amounts during a 5-hour period. Data collected under optimum conditions while resting.

6.3 User testing results

This section shows a metadata analysis of the data collection of the prototype system under everyday conditions. All data have been collected through the prototype system by research participant and retrieved form the backend database.

Table 8: Total participant collection coverages

#	Collection time	Total available time	Collection coverage
Participant 1	3 636 min	10 392 min	34.99 %
Participant 2	5 923 min	10 055 min	58.91 %
Participant 3	6 802 min	8 659 min	78.55 %
Averages			57.48 %
Totals	16 361 min	29 106 min	

Table 8 shows the data collection minutes and coverages of each participant. Collection time is the number of unique data packets collected per participant, while total available time shows total minutes between first and last data packet (data packets contain 1 full minute of data).

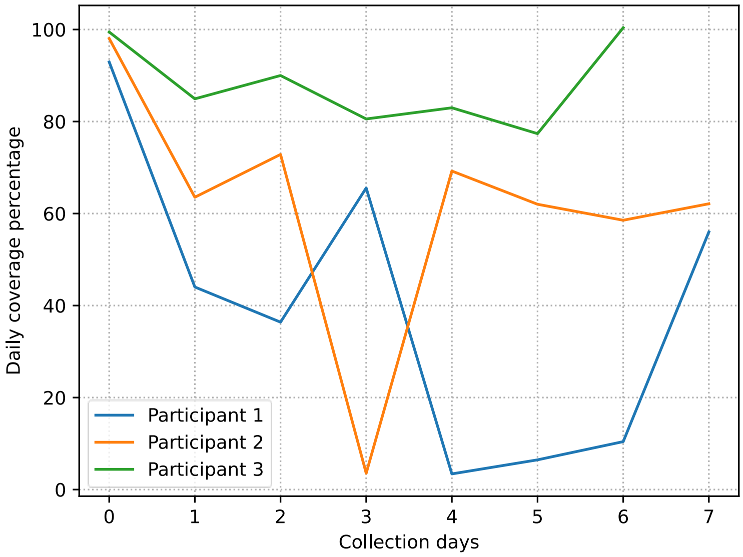


Figure 18: Daily collection coverages per participant

Table 9: Daily participant collection coverages

#	Day 1	Day 2	Day 3	Day 4	Day 5	Day 6	Day 7	Day 8
Participant 1	93.1 %	44.1 %	36.4 %	65.6 %	3.4 %	6.5 %	10.4 %	56.2 %
Participant 2	98.1 %	63.7 %	72.8 %	3.5 %	69.2 %	62.0 %	58.5 %	62.1 %
Participant 3	99.5 %	84.9 %	90.0 %	80.6 %	82.9 %	77.4 %	100 %	-

Table 9 shows collection coverage on individual collection days for each participant. Percentages on first and last days may be misleading as they only represent partial days. Figure 18 visualizes the information from table 9. Figure 19 shows the total number of packages collected per daily hour. This indicates what hours of the day suffered the greatest data loss.

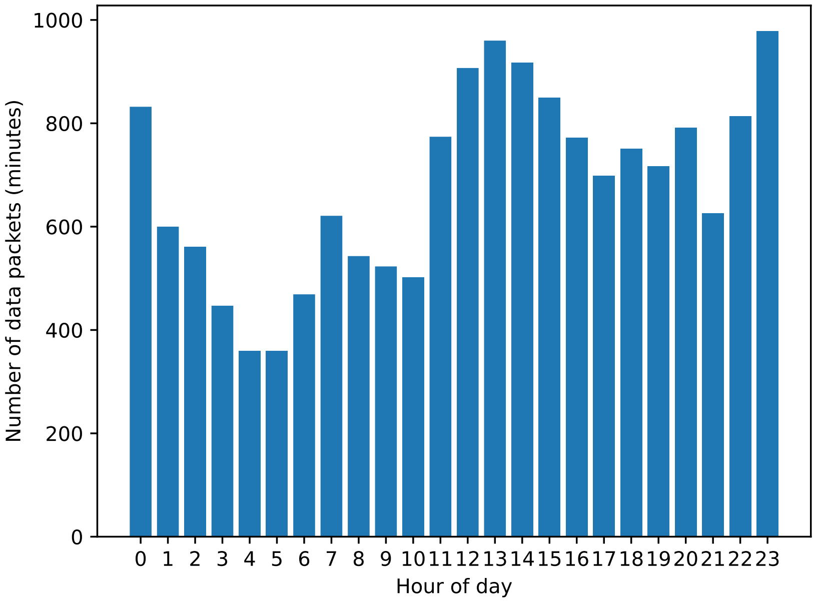


Figure 19: Total collected packets per daily hour for all participants.

6.4 Interview results

This section contains the results of the semi-structured interviews held with each study participant after their test period. The guide used in these interviews can be found in Appendix C.

Table 10: User interview results

#	Participant 1	Participant 2	Participant 3
Q 1.1	The system performs as expected, although with some issues. The E4 is ugly and a bit bulky, but this is no real issue. Hard to merge system into daily routines, especially charging cycles.	Hard to merge system into daily routines. Should have used fixed charging time each day. The system is generally OK to use, and the E4 feels like any other smartwatch. Misses time display on device.	System works well. Frequent disconnects, although quick and easy to reconnect. Charging routines were tiresome.

Q 1.2	No noted negative functionality. No apparent increase in battery usage.	The system uses quite a lot of power. Estimates that the system increased phone power usage by 50%. Application fails when in airplane mode.	No negative functionality. No apparent increase in power usage.
Q 1.3	The system is surprisingly stable. It is easy to detect when the device is connected or not.	No, not really.	System works well and is stable. No need to check whether the system is currently collecting data.
Q 1.4	Miniscule amounts of time spent in system menus. Mainly used to connect device. Would use system full-time.	Not much time spent with system in general. System was not too demanding, could be more demanding. Challenging to keep units connected at all times. Forgets to reconnect devices on disconnects. Would use system full-time.	Miniscule amounts of time spent reconnecting device, despite frequent reconnect. The charging cycle of the E4 is hard to merge with daily routines. Would use system fulltime, although not with the E4.
Q 1.5	Notifications from the app were frequently missed. Should have in-app options for increasing notification intensity. The application does not handle Bluetooth service shutdown. It continues to show the device as connected when it is not. The system should show estimated remaining battery time and it should have active charging reminders.	Should have more aggressive reminders of disconnects. System should inform user when device needs charging. Could the system estimate charging times for the device?	Would like more information in the app. The application crashed once during data collection. Would like to try some other data collection device than the E4.
Q 2.1	Did only use the UI for connecting the device. UI does not contain interesting or relevant information.	Did only use the UI for connecting the device.	Did use the UI daily for checking biometrics.

Q 2.2	The current UI is OK, although it is somewhat cluttered and should be reworked before going into production. Would like to see more diagnostic and statistical information, and more information on tags and annotation. Tag annotation interface can be confusing. The interface should have had a logo or a title.	The current UI is workable. Manual data entry could be added. Could have multiple menus instead of a single interface page. Would like to have access to more information through UI. Could the increased focus on the illness be bad?	Likes the UI. Would like the UI to show graphs and time windows of biometric readings. Would also like to see old tags and annotations. Maybe in a calendar interface?
Q 2.3	Much of the information in the UI was not usable without context. Would like to see explanations on all variables.	The information was not really useful in the current version of the app.	Did like the available information. Information should be more accurate.
Q 2.4	The UI should provide more information on what the collected data is used for. This could be scientific projects, diagnostics, medical purposes, etc.	The UI should show as little information as possible. Information should be available on demand. Integration with fitness apps? Should include more biofeedback and progression systems.	Would like to see tips on possible biofeedback approaches.

6.5 Fulfilled software requirements

Table 11 lists all predefined software requirements of the prototype system. All requirements were fulfilled during development.

Table 11: Fulfilled software requirements.

ID	Fulfilled by
F-1	Permissions
F-2	Databases, Main interface
F-3	Permissions
F-4	React-Native UI, Empatica data access, EmpaticaService, Main interface
F-5	React-Native UI, Empatica data access, Notification systems, Main interface
F-6	Notification systems, Main interface
F-7	EmpaticaService, Notification systems, Main interface
F-8	Notification systems, Databases, Main interface
F-9	Annotation systems, Databases
F-10	Tags and annotations, Main interface
F-11	EmpaticaService, Databases
F-12	Main interface
NF-1	Permissions
NF-2	Notification systems, Main interface, Annotation systems
NF-3	Notification systems, Main interface, Annotation systems
NF-4	Empatica data access, Databases
NF-5	Databases, mSpider user accounts
NF-6	Databases,
NF-7	EmpaticaService
NF-8	Network communicator
NF-9	EmpaticaService, Databases, Aggregator, Annotation systems
NF-10	Network communicator, EmpaticaService, Notification system, Databases, Annotations
NF-11	Network communicator
NF-12	Network communicator, Aggregator, mSpider API
NF-13	EmpaticaService, Databases
NF-14	Permissions
I-1	Empatica data access, Permissions, EmpaticaService
I-2	Empatica data access
I-3	Empatica data access, EmpaticaService
I-4	Annotation systems, Empatica data access
I-5	Network communicator, EmpaticaService, mSpider API
I-6	Annotation systems, Network communicator, EmpaticaService, mSpider API
S-1	Main interface, UI settings
S-2	mSpider user accounts
S-3	UI settings
S-4	React-Native UI, Empatica data access, Main interface, Notification systems, UI settings
S-5	mSpider user accounts
S-6	Network communicator, mSpider API

7 Discussions

7.1 System evaluation

7.1.1 Performance

Impact on battery usage

System performance metrics are listed in subsection 6.2. As can be seen in table 5, the battery usage of the prototype system can have a substantial impact on smartphone battery life. During optimal collection conditions (during sleep, with internet connection), the battery charge percentage of the Samsung Galaxy Note testing device was reduces by an average of 58% over a collection period of 8 hours. The baseline battery usage for the same device, under comparable conditions, without the prototype system installed was only 10%, showing that the prototype caused approximately 48% battery loss over 8 hours. This is an increase of more than 500% over baseline, at least in this data collection setting, which indicates that the data handling processes of the application may be too complex. Regrettably, none of the related research projects [16], [37], [43], [47] that utilized the E4 streaming mode have reported on their systems' battery usages.

On the other hand, it is important to mention that data collection during sleep where the general usage of the smartphone is low could be unrepresentative for data collection periods where the smartphone is in more frequent use. Modern operating systems, including most Android versions, have built-in systems for reducing power consumption during periods of low interaction where the OS greatly reduces resource availability for running applications and processes. As described in subsection 5.5.4, the prototype system utilizes a persistent foreground service for data collection, which could be hindering the Android OS from executing its power saving routines. This notion is strengthened by the low difference in smartphone battery loss when the prototype system is running but not collecting data, which could indicate that the increased power usage caused by data collection will be less noticeable during periods of regular smartphone use.

Nevertheless, the prototype system shows very high levels of battery usage, and future system iterations should incorporate changes in data handling approaches for reducing power consumption.

Impact on device memory

As can be seen in table 6 in subsection 6.2, the average memory usage of the prototype system is stable. Both sources report the average memory usage to be 38 MB on average during runtime, regardless of data collection state. This could indicate that the prototype application has no memory leaks, although this should be verified through more thorough testing over longer time periods.

Data storage and upload

Data storage and upload metrics can be found in table 7 in subsection 6.2. These metrics only include automatically collected data through the E4 device and not any data size increases from manual user tags. The impact on manual tag data input on storage and upload sizes is hard to measure in at this stage of prototype development as the average number of tags per potential user may be impossible to predict.

As can be seen in table 7, the average data storage rate is equal in frontend and backend storage, although upload sizes are slightly larger, which probably stems from formatting and added metadata. Data storage increase found to be an average of more than 25 MB per hour, which is a more than 9-fold increase over the data collection rate found by *Ursin* [14] through testing of the Empatica data collection environment (see table 1 in subsection 1.2.2). This massive data storage increase is probably caused by two reasons: Firstly, the choice of using JSON as data the main storage format, which stores all data in a dictionary data structure represented as strings of characters. This makes the data easily accessible, although it can have large impact on data storages sizes. Secondly, the increased resolution of data in the prototype system. The Empatica data collection environment stores all data on a per-second basis. This means that all datapoints collected within the same second are marked with the same normal-resolution Unix timestamp, which could make it non-trivial to extract the exact timing of individual measurements. On the other hand, the prototype system uses millisecond-resolution Unix timestamps when storing data, where each single datapoint is annotated by an individual timestamp. This not only increases data storage and upload size, but it could also increase the usefulness of the collected data, as mentioned in subsection 7.4.4.

Although the potential economic burden on individual participants caused by large uploads is minimized (requirement NF-8), the prototype system still has very high data collection rates and could cause problems for both participants and researchers in the future. As the maximum data collection and upload rate of the system could potentially reach more than 18 GB per month per participant, the data storage requirements could be very expensive or insurmountable for long-term research projects with many participants that all use the prototype system for data collection.

7.1.2 Known limitations of the prototype

User login not properly implemented

The protype system only implements a rudimentary login system where predefined usernames are utilized for identifying individual participants or users in the system. In future iterations of the prototype a more advanced login system should be implemented that utilizes proper authentication methods such as two-factor authentication for user login.

Data fetching from server not implemented

No API endpoints for fetching data form the backend server were implemented for this project. This feature was not planned for this stage of prototyping and was not prioritized. On the other hand, certain features and information panels in the GUI, like the collection coverage panel, show information that should be based on backend data storage metadata. Future prototype iterations should implement endpoints for fetching metadata and metrics from the mSpider backend. Additionally, the backend system has no interfaces that allow third-party extraction of collected data for research purposes. This makes data extraction only available for system administrators through direct database interfaces.

Not using properly using events for internal communication

Inter-thread events are used in the frontend application for communication between the GUI and the persistent service. In the current prototype, data from the state database is actively pushed from the persistent service to the GUI once per second. This is not a favorable approach, as it forces the GUI to only update on received data, sometimes reducing the responsiveness of the GUI. By connecting the GUI and the state database directly though Android live data approaches, the system would automatically update the GUI when new data was available in the database. This was not implemented in the current prototype because of time constraints.

Duplicate data uploads

The persistent foreground service of the frontend application uses short-time worker threads for uploading data to the backend server. The upload process is instigated on a fixed time interval where a new upload thread is spawned for each interval iteration, and upload threads are automatically terminated if there is no more data be uploaded. However, if there are many data packets to upload and the upload speed is slow, not all data will be uploaded before the next upload iteration, which causes an additional upload thread to be spawned. Communications between upload threads have not been implemented, causing all upload threads to perform the exact same work at the same time. This issue cannot do any harm to the data collection, as the ACID properties of both the frontend and the backend databases will keep data from getting overwritten or lost. On the other hand, large amounts of ready-to-upload data will cause several upload threads to be spawned over time, causing heavily increased network traffic, and potentially tying up system resources. Future prototype iterations should implement techniques for detecting if upload an upload thread is active before a new thread is spawned. Alternatively, implementing synchronization infrastructure between upload threads could be a good solution for increasing data upload speeds.

Frontend data processing

As described in subsection 5.3.1, the frontend application is responsible for compiling and organizing data into data packets. This makes data handling in the mSpider backend simple, as the data packets can be directly inserted into the database without having to consider data conflicts or overwrites. However, additional processing time is needed in the frontend to facilitate this solution, potentially leading to increased power usage as discussed in subsection 7.1.1. Moving more of the data processing to the backend of the system could potentially reduce frontend processing times, although the data handling systems in the frontend application would have to be rebuilt to accommodate the changes. Future prototyping projects on this system should look into possible optional approaches.

7.2 Discussion on user testing and interviews

This section discusses the results of the user testing and user interviews. All tables and figures mentioned can be found in section 6.3, which shows metadata on data collected during user testing.

7.2.1 Collection coverages

General coverage

Subsection 1.2 and 6.1 explored the data collection coverages of similar data collection systems, showing coverages between 25% ([50]) and 70% ([14], [42]), with only a few systems and studies reaching more than 50% coverage.

As can be seen by the averages in table 8, the collection coverage of the prototype system is 57.48%, which is on-par with or slightly above findings from other studies [41], [48], [50] and [16]. This could indicate that the features added to the prototype for increasing participant involvement were largely ineffective. On the other hand, table 8 show that the collection coverages between individual participants are highly varying. Additionally, table 9 and figure 18 show that the data collection coverage for each participant can differ heavily on daily basis. These observations could indicate different levels of participant motivation, or to what degree individual participants are able to mesh the requirements of the data collection process with their daily routines. Evidence of this last point can be found in the results from the user interviews where all participants mention having problems with aspects of the E4 charging cycle, which caused them to miss large portions of their available collection time.

Hourly coverages

Figure 19 shows the combined hourly data collection for all participants throughout the user testing period. During the user testing participants 1 and 2 had approximately 7 days of data collection while participant 3 had 6 days. The estimated maximum number of data packets collected per individual hour of the day throughout the data collection is approximately equal to 1200. As can be seen in figure 19, data collection was at its lowest during the night, which is probably caused by unattended disconnects during sleep or the device not being properly connected before going to bed. This last issue is discussed in subsection 7.2.5. Figure 19 also shows high variations in hourly data collection. This can be caused by participants having problems with integrating the data collection and E4 charging cycle in their daily routines. High levels of integration would show more even data collection, at least during waking hours. Longer testing periods could have given the participants time to get used to the data collection routines and would probably have produced better data.

7.2.2 Notification intensity

As detailed in subsection 5.4.2, the notification infrastructure is used in the prototype system for actively informing users of participant about disconnects between the E4 data collection device and the prototype application. During interviews, participants 1 and 2 remarked that they frequently failed to notice notifications or that they successfully noticed the notification but simply forgot to re-establish the connection. This could indicate that the notification scheme used by the prototype system was too mild and should be made more insistent. On the other hand, participant 3 mentioned no issues with notifications, which could explain the differences between the daily coverages shown in figure 18. A probable cause for the different notification behaviours may be that the prototype application was running on the users' private smartphones during user testing, and differences user settings and OS versions may produce divergent notification behaviours between individual devices.

7.2.3 The Empatica E4 versus daily routines

As discussed in the previous paragraph, participants experienced problems with merging the required data collection routines with their daily lives, where charging and maintaining the E4 was the most frequently mentioned issue. For instance, one participant noted that having very young children was not particularly conducive to the data collection process. Issues regarding the physical and logistical aspects of using the E4 under ever-day conditions were also discussed by several other studies [16], [35], [37], [39], [40], [50], which strengthens the notion of the E4 not being well suited for data collection in environments completely controlled by patients or users. This is further discussed in subsection 7.3.6.

Extending the data collection period could help participants or users to adjust to the maintenance requirements of the E4, and testing period of this project was probably not long enough to develop proper data collection and charging routines.

7.2.4 UI elements

During the user interviews, participants 1 and 3 wanted access to additional and more detailed information through the user interface, and features like diagnostic information and a calendar interface for statistics were mentioned. On the other hand, participant 2 wished for a less informative interface to minimize the focus on illnesses. This indicates that the modular interface of the prototype can be a viable solution, although with many more customization and intensity options.

7.2.5 General issues during testing

Empatica device fault

During the start of the user testing period, the E4 device stopped working. It remained unresponsive and could not be used for data collection, which delayed user testing by several days. After contacting Empatica support, the problem was eventually solved by letting the E4 battery completely drain, remove all data from the device, and installing new firmware through the Empatica manager desktop application. This fault was probably caused by the internal device storage being completely full. Although the issue can be simple to rectify, it can be very hard to diagnose, especially for regular users.

Protype Bluetooth connections

The Bluetooth connection between the E4 device and the frontend application caused issues both during field testing and user testing of the system. The E4 would spuriously disconnect for the frontend application when other Bluetooth devices were connected to the smartphone. The reason for this issue is not known, although it could be caused by Bluetooth bandwidth overload or possibly electromagnetic interferences between devices.

Issues with the notification features of the prototype system were encountered during user testing. When Bluetooth is turned off via OS setting or when device flight mode is engaged, the E4 device is naturally disconnected from the application and switches itself off. However, the new connection state is not detected by the notification systems of the application, and the E4 is still shown to be connected by the application GUI. This is a major flaw in the prototype, as it fails to perform one of its most central tasks of keeping the user informed on the system's data collection state. The cause of this fault is unknown, although it may be rectified by allowing the application to react to OS events concerning Bluetooth connectivity.

7.3 Additional discussions

7.3.1 Issues with project execution

During the development period of the prototype system, some drawbacks and shortcomings of the project became apparent. Firstly, having access to only a single E4 device during prototype development was probably detrimental to proper feature implementation as iterative user involvement and user testing was not possible. This could be partially to blame for the software bugs discovered during user testing at the end of the project. Secondly, the short time span of the project lead to very brief user data collection trials, which caused findings from the trials to be largely vague and inconclusive. This problem was mentioned by *Siirtola et al* [11] who concludes that data collection periods for migraine studies should be at least 6 months long to ensure proper data foundations for research.

7.3.2 Assumed need for system

A major assumption for this project is that real-time migraine prediction systems are wanted in the migraine community. The logical argumentation for this is presented through subsection 1.1, although no migraineurs were directly consulted for this project. This should probably be investigated in the future as these kinds of systems could lead to possible psychological side-effects like anxiety or automation bias. The anxiety issue in particular is an important area to investigate as many migraineurs already suffers from increased anxiety and depression caused by their illness [5].

7.3.3 Expanded system usage

Even though the main premise for this project was to strengthen data collection for migraine research, the resulting software is not limited to detecting migraine. The current prototype system is a tool for capturing real-time data from the Empatica E4, for enabling data storage options, and for collecting ground truth for data correlations. Most of the features of the system (including the UI) could probably be easily repurposed for use in other fields of research. In a paper from 2018, *Scheffer et al* [18] describes the concept of resilience in living biological systems, where resilience is described as the systems innate capability to adjust its own internal processes to keep the system in a functioning state. As briefly described in subsection 2.1, thresholds are “tipping points” in the system where the internal processes are unable to keep a certain state and the system is forced to violently cascade into a different state. In the paper, migraine attacks or epileptic seizures are used as examples of such thresholds in the system of the human body, and by monitoring the stabilizing internal processes of the system these threshold events should be predictable. As migraine events have been shown to be predictable by both *Pagán et al* [10] and *Siirtola et al* [11], other threshold events could also be predictable through measurements and data correlation. This indicates that the prototype system developed through this project could be utilized in the detecting of many types of thresholds in the human body.

7.3.4 Data collection resolution

In subsection 1.2.2 the Empatica solution is noted to have a time resolution of 1 second when storing data. All data captured within any given second is marked with the full second timestamp, regardless of where in the second it was captured. As the order of the measurements is kept, there is some order to this approach. However, the sensor polling rate on the E4 operates on “best effort”, meaning that it constantly strives to keep the predefined polling frequency of each sensor [51]. Sometimes the polling frequency is altered because of various reasons and the device is not able to keep the desired data delivery schedule. This results in unreliable derived timestamps for collected data, especially for acceleration and PPG sensor values, which have desired polling rates of 32hz and 64hz respectively. This is normally not an issue as the resolution of the collected data is sufficient for most usages. However, Empatica’s policy on delivering useful IBI values is based on an only-deliver-when-certain policy, which results in approximately 30% of IBI data missing from datasets collected under everyday conditions. This is an issue with all PPG sensor hardware, and the only-deliver-when-certain policy is probably a good approach to this problem from Empatica.

However, the protype system created through this project stores collected data with full Unix timestamps using millisecond resolution. As the PPG data loss is a product of acceleration and movement during data collection, there is a possibility for reducing the PPG data loss by time-correlating PPG data and accelerometer data. This is especially interesting for migraine research as prediction and machine learning relies on differences in data patterns for detecting migraine events (see subsection 2.2).

7.3.5 Discrete tag annotation

As described in subsection 5.4.3, the UI of the prototype system allows users to tag events and to describe these events using free-text input. This approach relies heavily on the ability of users to properly describe their experiences and symptoms, which could lead annotations that are hard to quantify and that could be less useful for the prediction process. By introducing discrete choices instead of free text input, this problem could be mitigated to some degree, at least within distinct user groups. This was not prioritized for this project, as defining the correct discrete choices could be hard. The *mBrain* [16] system utilized discrete choices in their classification approach based on data from *International Classification of Headache Disorders* [17], which is the standard authority on symptoms of headache disorders. This resulted in many misclassifications, as the data from ICHD fails to incorporate symptom differences between individual patients. Although the findings from *mBrain* [16] could indicate that free-text data input could be the right approach, *Koskimäki et al* [15] points out that discrete choices would help the patient to better identify and quantify their own experiences. Both these arguments are important to consider if discrete annotation choices are added to future iterations of the prototype.

7.3.6 Empatica E4 for data collection

Although the Empatica E4 is regarded by some as the best available device for ambulatory data collection, findings from the testing phase of this project and other mentioned projects have pointed out major drawbacks and deficiencies in both the device’s design and features. Firstly, logistical obstacles like the device’s short battery life and long changing times can make it hard for users to insert the required data collection routines into their daily schedules, as was remarked during participant interviews. Secondly, and arguably the most noticeable and most frequently issue encountered may be the tendency of the device to arbitrarily break Bluetooth connections without informing the user, leading to data loss and possible user irritation, as was mentioned in subsection 7.2. This problem is made worse by the devices inability to automatically reconnect or to continue data collection while disconnected. Although systems can be developed that implement workarounds for all of these issues, like the prototype developed for this project, all data collection systems that include the E4 are still largely centered on the data collection process rather than on the user or patient. This could lead to user fatigue and possibly render these systems much less useful for the user than they potentially could be. Lastly, as discovered through the implementation part of this project, the device intermittently suffers from runtime issues that requires significant technical knowledge to diagnose and rectify, as was briefly discussed in subsection 7.2.5. In combination, all of these major and minor failings of the Empatica E4 could make it largely unfit for use in medical situations, like continuous migraine prediction under real-life conditions. This is seconded by the findings of the literature review. Research projects like [42] that are conducted on fixed locations where participants have ample access to support personnel are achieving much higher data coverage percentages compared to projects like [16], [37] where participants collect data during daily activities.

7.3.7 Empatica EmbracePlus

Figure 20: Empatica EmbracePlus, sourced empatica.com.

The Empatica EmbracePlus [52] is a successor device for the E4 and is set to be launched in 2023. It offers many improvements, including extended sensor availability, adjustable sensor polling rates, and the ability to continue data collection when Bluetooth connections are broken. Although the current iteration of the prototype system is configured to handle the biometric variables collected by the Empatica E4, the system could easily be configured to incorporate extra variables. Both the data handling approaches in the frontend application and the data storage on the backend server allows for adding extra biometric variables, and more data display panels could be added to the configurable status grid of the UI. Additionally, some of the features of the prototype system, like the notification infrastructure, could be made superfluous or obsolete by moving from the E4 to the EmbracePlus. This would have to be evaluated when the EmbracePlus released and reviewed.

7.4 Research questions

SQ1: How can a system for real-time data capture and processing be implemented?

The prototype system builds on the findings from *Siirtola et al* [11] and *Pagán et al* [10] by giving continuous access to both raw data and data collection metrics in real-time, captured from the Empatica E4. Although this approach is more computationally expensive, it offers clear advantages to the retro-active data access framework created by Empatica, which was discussed and criticized by *Ursin* [14] and *Siirtola et al* [11]. For instance, raw real-time data acquired through the prototype system can be utilized to make continuous migraine forecasts and other types of momentary medical evaluations, which would have been challenging to implement within the native Empatica infrastructure.

SQ2: How can the data collection system be integrated with mSpider?

The prototype system stores all collected data on the *mSpider* [13] data storage solution and can be configures to support other storage options. This increases the availability of the collected data and lays the groundwork for flexible data access through a future API interface. The data access interface of *E4 connect* was mentioned as a weak-point in the Empatica data collection infrastructure by both *Koskimäki et al* [15] and *Ursin* [14].

SQ3: How could real-life events tagging implemented in the system?

Although this feature was not properly tested, the prototype system includes facilities for tagging and annotating data. The design, implementation, and UI elements for this functionality are explained in subsections 5.3, 5.4.3 and 5.5.3. This functionality is central for correlating the automatically collected data with migraine attacks.

SQ4: How can data losses during data collection with the E4 be minimized?

The results of the literature review shown in section 6.1.4 indicate that increased user involvement may be crucial for decreasing data loss under everyday conditions. This is seconded by experiences made described by the *mBrain* [16] project where increased system usability and encouragement by the project team lead to significant increases in the amounts of collected data between testing sessions. The prototype system developed in this project incorporates both active and passive features for increasing user involvement, with the main goal of transferring responsibility for the data collection from the user to the system. Active user involvement is implemented through a notification interface that automatically inform the user about E4 disconnects, while the passive involvement system utilizes a traffic-light approach for continuously showing the E4 connection status on the smartphone UI. These UI elements can improve upon the user interaction approaches in the *mBrain* [16] system and the native Empatica infrastructure application *E4 realtime* discussed in *Ursin* [14], which both allow the E4 to silently disconnect without informing the user, thus increasing data loss.

On the other hand, the project has partially failed to improve the data collection capabilities of the E4, as was indicated in subsection 6.3 and discussed in subsection 7.2. The implemented measures have not improved data collection compared to findings from related research and similar systems. This is probably caused by the low intensity of the active features of the system, system bugs, and the reliance on the E4 as a collection device.

RQ: How can the data collection capabilities for the Empatica E4 be improved with regards to migraine prediction and research?

As shown through sections 4 and 5, this project has produced a prototype system for increasing the data collection capabilities of the Empatica E4 device. Although the system failed to improve the data collection coverage of the E4, other important have improved the data collection capabilities and data accessibility of the E4. The prototype system has increased the data collection capabilities of the E4 by providing access to real-time data, enabling data storage options through mSpider, decreased data loss with active UI elements, and provided annotated tagging of real-life events. This creates solid foundations for completing the long-term vision to deploy a highly automated system for managing and forecasting migraine attacks.

8 Future work

8.1 Next prototype iteration

The next iteration of the prototype system should incorporate fixes to faults and bugs that have been discovered through testing. Most importantly, the bugs in the notification system found through user-testing should be diagnosed. Secondly, the notification system should be made more aggressive, with hourly reminders if the E4 is not connected to the application. This would greatly strengthen the data collection potential of the applications. Thirdly, all GUI information elements should be populated with useful facts and actual information. Finally, better intra-thread communications should be implemented to facilitate live-data approaches and to reduce the reliance on pushing status data.

8.2 Possible future projects

Implement prediction

Although this project has been focused on creating a data collection solution and strengthening its data collection capabilities, the long-term goal of creating a real-time migraine prediction system still remains to be fulfilled. The next step in this process could be to improve the data collection capabilities of the prototype and conduct more expansive data collection trials. Data from the trials would then be used to train participant-specific ML models for prediction, which could be included in the prototype application.

iOS implementation

As mentioned in subsection 5.5.1, the current prototype system is only available on the Android OS platform. This possibly excludes a lot of potential users, as many people are using iOS devices. Implementing an iOS version of the data collection system could be important for making the solution available for as many people as possible. However, this could be a non-trivial task, as there are major differences in the development approaches supported by each platform. On top of this, the *Empalink* library for the E4 is not OS-agnostic and must be tailor-made for each platform.

Empatica EmbracePlus

Initially, this project was meant to include testing and discussions on the new Empatica EmbracePlus, briefly described in subsection 7.3.7. These plans did not come to fruition as the ordered EmbracePlus did not arrive within the timeframe of the project due to logistical problems. Future projects should expand the prototype system to support the EmbracePlus, as it improves on many of the problems encountered with the E4.

Discrete annotation options

Future projects should explore the possibilities of adding discrete annotation options to manual user tags. This would make the tags much more precise and create better foundations for automated ML solutions. However, this could be a non-trivial task as the annotation options must represent all the different states and events that users would want to tag, while at the same time not swamping the user in different annotation options. Close cooperation with migraineurs and medical researchers would probably be needed accomplishing this.

Data collection comparisons

Subsection 1.2.2 shows testing and evaluation of the data management systems created by Empatica to facilitate data collection through the E4 [14], and the listed metrics are used as base arguments for creating the prototype system presented throughout this report. However, the results presented in subsection 1.2.2 can be hard to compare to the results from this project (subsections 7.1 and 7.2) as the data was not collected under the same circumstances. When the initial bugs and problems of the current prototype system have been rectified, a proper data collection trial between the two systems should be conducted. This will show whether the features that have been implemented for the prototype system are effective in increasing data collecting coverage and reducing user involvement in the data collection process.

Move computation to the backend

As described in subsection 5.5.5, the fronted application of the prototype system is responsible for compiling the collected data into data packets. This is done to simplify the transportation of data to the backend as well as making data management easy for the backend database. However, as was briefly discussed in subsection 7.1.3, this approach could be to blame for the high power usage of the application. Moving the data compilation process to the backend system would probably decrease the power usage of the application drastically. However, this would call for a complete redesign of the data storage and data transportation systems.

Improve UI

Although the UI of the prototype application is workable, many important and useful elements could be added in the future. Most of the information cards in the UI are empty and should be filled with useful information. Many changes and improvements were mentioned during user testing. For instance, access to historical data and data visualization tools was requested. Additionally, the tag annotation interface should be completely redesigned to be more intuitive and user-friendly.

9 Conclusion

This report has described the process of implementing and designing a real-time data collection system prototype for capturing, handling, and storing data from the Empatica E4 in real time. The prototype includes functionality for increasing data collection coverage, and it has been integrated with the data storage solution mSpider. Although the application is not completely bug-free, it incorporates active features for assisting users in the data collection process and it has been shown to at least provide data collection capabilities on par with other similar systems. The prototype successfully lays the groundwork a full migraine prediction system.

As discussed in subsection 7.3.1, the scope of this project was probably too large for the limited project timeframe. Adding more time to the project could have produced more cohesive and workable results, especially with regards to the metadata results of the user testing, although limiting the scope would have been both more effective and realistic. Scope and research questions should have been more thoroughly defined at the project beginning, as this could have produced more cohesive end results. Additionally, both the protype development and the user testing portions of the project were hampered by only having access to a single Empatica E4 device. Access to more devices would have greatly increased the user testing potential and would have enabled iterative user testing during prototype development. This could have produced a more advanced prototype system.

Learnings and contributions

As discussed in section 7.1.2, the prototype system developed through this project has increased the usability and data collection flexibility of the Empatica E4 through real-time access to data from the E4, choice of storage, and active features for participant involvement. Although the prototype system is meant to facilitate migraine prediction and research, it can be configured for supporting other health-related research as well.

On top of this, through this project the Empatica E4 has been proven to be a competent data collection device under the right conditions. However, as discussed in subsection 7.3.6, the E4 is not suited as a long-term medical device and should mainly be used for research purposes.

References

[1] T. J. Steiner and L. J. Stovner, 'Global epidemiology of migraine and its implications for public health and health policy', *Nat Rev Neurol*, vol. 19, no. 2, Art. no. 2, Feb. 2023, doi: 10.1038/s41582-022-00763-1.

[2] L. J. Stovner, K. Hagen, M. Linde, and T. J. Steiner, 'The global prevalence of headache: an update, with analysis of the influences of methodological factors on prevalence estimates', *The Journal of Headache and Pain*, vol. 23, no. 1, p. 34, Apr. 2022, doi: 10.1186/s10194-022-01402-2.

[3] 'Migrene i et samfunnsperspektiv | Oslo Economics', *» Migrene i et samfunnsperspektiv*. https://osloeconomics.no/publication/migrene-i-et-samfunnsperspektiv/ (accessed Dec. 13, 2022).

[4] P. Goadsby *et al.*, 'Early vs. Non-Early Intervention in Acute Migraine — "Act When Mild (AwM)". A Double-Blind, Placebo-Controlled Trial of Almotriptan', *Cephalalgia*, vol. 28, no. 4, pp. 383–391, Apr. 2008, doi: 10.1111/j.1468-2982.2008.01546.x.

[5] P. J. Goadsby, P. R. Holland, M. Martins-Oliveira, J. Hoffmann, C. Schankin, and S. Akerman, 'Pathophysiology of Migraine: A Disorder of Sensory Processing', *Physiol Rev*, vol. 97, no. 2, pp. 553–622, Apr. 2017, doi: 10.1152/physrev.00034.2015.

[6] 'Track Your Migraine Symptoms And Triggers - Migraine Buddy - Track your migraine attacks and triggers to control migraine. Download Migraine Buddy now!' https://migrainebuddy.com/ (accessed May 18, 2023).

[7] 'Migraine Monitor | Migraine symptom tracker mobile app', *Migraine Monitor*. https://migrainemonitor.com/ (accessed May 18, 2023).

[8] M. A. Connelly and M. E. Boorigie, 'Feasibility of using "SMARTER" methodology for monitoring precipitating conditions of pediatric migraine episodes', *Headache*, vol. 61, no. 3, pp. 500–510, Mar. 2021, doi: 10.1111/head.14028.

[9] S. H. Ingvaldsen *et al.*, 'A Biofeedback App for Migraine: Development and Usability Study', *JMIR Form Res*, vol. 5, no. 7, p. e23229, Jul. 2021, doi: 10.2196/23229.

[10] J. Pagán *et al.*, 'Robust and Accurate Modeling Approaches for Migraine Per-Patient Prediction from Ambulatory Data', *Sensors (Basel)*, vol. 15, no. 7, pp. 15419–15442, Jun. 2015, doi: 10.3390/s150715419.

[11] P. Siirtola, H. Koskimäki, H. Mönttinen, and J. Röning, 'Using Sleep Time Data from Wearable Sensors for Early Detection of Migraine Attacks', *Sensors (Basel)*, vol. 18, no. 5, p. E1374, Apr. 2018, doi: 10.3390/s18051374.

[12] 'E4 wristband | Real-time physiological signals | Wearable PPG, EDA, Temperature, Motion sensors', *Empatica*. https://www.empatica.com/research/e4 (accessed Dec. 07, 2022).

[13] A. Henriksen, E. Johannessen, G. Hartvigsen, S. Grimsgaard, and L. A. Hopstock, 'Consumer-Based Activity Trackers as a Tool for Physical Activity Monitoring in Epidemiological Studies During the COVID-19 Pandemic: Development and Usability Study', *JMIR Public Health Surveill*, vol. 7, no. 4, p. e23806, Apr. 2021, doi: 10.2196/23806.

[14] D. Ursin, ‘Possibilities of automated migraine prediction under every-day conditions: Investigating current technologies and solutions’. Unpublished work, Dec. 18, 2022.

[15] H. Koskimäki, H. Mönttinen, P. Siirtola, H.-L. Huttunen, R. Halonen, and J. Röning, ‘Early detection of migraine attacks based on wearable sensors: experiences of data collection using Empatica E4’, in *Proceedings of the 2017 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proceedings of the 2017 ACM International Symposium on Wearable Computers*, Maui Hawaii: ACM, Sep. 2017, pp. 506–511. doi: 10.1145/3123024.3124434.

[16] M. De Brouwer *et al.*, ‘mBrain: towards the continuous follow-up and headache classification of primary headache disorder patients’, *BMC Med Inform Decis Mak*, vol. 22, p. 87, Mar. 2022, doi: 10.1186/s12911-022-01813-w.

[17] H. Gobel, ‘The International Classification of Headache Disorders’, *ICHD-3*. https://ichd-3.org/ (accessed May 21, 2023).

[18] M. Scheffer *et al.*, ‘Quantifying resilience of humans and other animals’, *Proceedings of the National Academy of Sciences*, vol. 115, no. 47, pp. 11883–11890, Nov. 2018, doi: 10.1073/pnas.1810630115.

[19] ‘Migraine Headaches: Causes, Treatment & Symptoms’, *Cleveland Clinic*. https://my.clevelandclinic.org/health/diseases/5005-migraine-headaches (accessed Dec. 07, 2022).

[20] ‘ISO 13485:2016’, *ISO*. https://www.iso.org/standard/59752.html (accessed Dec. 07, 2022).

[21] ‘Legal’, *Empatica*. https://www.empatica.com/legal (accessed Dec. 07, 2022).

[22] ‘Meet Android Studio’, *Android Developers*. https://developer.android.com/studio/intro (accessed May 22, 2023).

[23] ‘WeFitter | Health & Fitness Gamification API’. https://www.wefitter.com/en-us/ (accessed May 24, 2023).

[24] E. Johannessen, A. Henriksen, E. Årsand, A. Horsch, J. Johansson, and G. Hartvigsen, ‘Health Research Requires Efficient Platforms for Data Collection from Personal Devices’, *Stud Health Technol Inform*, vol. 302, pp. 841–845, May 2023, doi: 10.3233/SHTI230286.

[25] ‘Tromsøundersøkelsen | UiT’. https://uit.no/research/tromsoundersokelsen?p_document_id=705235&Baseurl=%2Fresearch%2F (accessed May 31, 2023).

[26] ‘What is personal data?’ https://commission.europa.eu/law/law-topic/data-protection/reform/what-personal-data_en (accessed May 22, 2023).

[27] ‘What personal data is considered sensitive?’ https://commission.europa.eu/law/law-topic/data-protection/reform/rules-business-and-organisations/legal-grounds-processing-data/sensitive-data/what-personal-data-considered-sensitive_en (accessed May 15, 2023).

[28] ‘Art. 4 GDPR – Definitions’, *General Data Protection Regulation (GDPR)*. https://gdpr-info.eu/art-4-gdpr/ (accessed Dec. 07, 2022).

[29] 'Chapter 3 – Rights of the data subject', *General Data Protection Regulation (GDPR)*. https://gdpr-info.eu/chapter-3/ (accessed May 22, 2023).

[30] 'Volere Requirements Specification Template', *Volere Requirements*. https://www.volere.org/templates/volere-requirements-specification-template/ (accessed May 10, 2023).

[31] 'Guide to app architecture', *Android Developers*. https://developer.android.com/topic/architecture (accessed May 18, 2023).

[32] 'React Native · Learn once, write anywhere'. https://reactnative.dev/ (accessed May 22, 2023).

[33] R. T. Fielding, M. Nottingham, and J. Reschke, 'HTTP Semantics', Internet Engineering Task Force, Request for Comments RFC 9110, Jun. 2022. doi: 10.17487/RFC9110.

[34] M. J. Page *et al.*, 'The PRISMA 2020 statement: an updated guideline for reporting systematic reviews', *BMJ*, p. n71, Mar. 2021, doi: 10.1136/bmj.n71.

[35] N. Ahmadi *et al.*, 'Quantifying Occupational Stress in Intensive Care Unit Nurses: An Applied Naturalistic Study of Correlations Among Stress, Heart Rate, Electrodermal Activity, and Skin Temperature', *Hum Factors*, vol. 64, no. 1, pp. 159–172, Feb. 2022, doi: 10.1177/00187208211040889.

[36] T. A. Ajayi, L. Salongo, Y. Zang, N. Wineinger, and S. Steinhubl, 'Mobile Health-Collected Biophysical Markers in Children with Serious Illness-Related Pain', *Journal of Palliative Medicine*, vol. 24, no. 4, p. 580, Apr. 2021, doi: 10.1089/jpm.2020.0234.

[37] J. Amores Fernandez, N. Mehra, B. Rasch, and P. Maes, 'Olfactory Wearables for Mobile Targeted Memory Reactivation', in *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*, in CHI '23. New York, NY, USA: Association for Computing Machinery, Apr. 2023, pp. 1–20. doi: 10.1145/3544548.3580892.

[38] L. Barrios, P. Oldrati, S. Santini, and A. Lutterotti, 'Evaluating the accuracy of heart rate sensors based on photoplethysmography for in-the-wild analysis', in *Proceedings of the 13th EAI International Conference on Pervasive Computing Technologies for Healthcare*, in PervasiveHealth'19. New York, NY, USA: Association for Computing Machinery, May 2019, pp. 251–261. doi: 10.1145/3329189.3329215.

[39] N. Gao, M. S. Rahaman, W. Shao, K. Ji, and F. D. Salim, 'Individual and Group-wise Classroom Seating Experience: Effects on Student Engagement in Different Courses', *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.*, vol. 6, no. 3, p. 115:1-115:23, Sep. 2022, doi: 10.1145/3550335.

[40] N. Gao, W. Shao, M. S. Rahaman, and F. D. Salim, 'n-Gage: Predicting in-class Emotional, Behavioural and Cognitive Engagement in the Wild', *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.*, vol. 4, no. 3, p. 79:1-79:26, Sep. 2020, doi: 10.1145/3411813.

[41] S. Gashi *et al.*, 'Detection of Artifacts in Ambulatory Electrodermal Activity Data', *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.*, vol. 4, no. 2, p. 44:1-44:31, Jun. 2020, doi: 10.1145/3397316.

[42] E. Kleiman, A. J. Millner, V. W. Joyce, C. C. Nash, R. J. Buonopane, and M. K. Nock, 'Using Wearable Physiological Monitors With Suicidal Adolescent Inpatients: Feasibility and

Acceptability Study', *JMIR Mhealth Uhealth*, vol. 0, no. 0, p. e0, Sep. 2019, doi: 10.2196/13725.

[43] F. Larradet, R. Niewiadomski, G. Barresi, and L. S. Mattos, 'Appraisal theory-based mobile app for physiological data collection and labelling in the wild', in *Adjunct Proceedings of the 2019 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proceedings of the 2019 ACM International Symposium on Wearable Computers*, in UbiComp/ISWC '19 Adjunct. New York, NY, USA: Association for Computing Machinery, Sep. 2019, pp. 752–756. doi: 10.1145/3341162.3345595.

[44] M. Nasseri *et al.*, 'Signal quality and patient experience with wearable devices for epilepsy management', *Epilepsia*, vol. 61 Suppl 1, pp. S25–S35, Nov. 2020, doi: 10.1111/epi.16527.

[45] K. K. G. Ravindran, C. della Monica, G. Atzori, D. Lambert, V. Revell, and D.-J. Dijk, 'Evaluating the Empatica E4 Derived Heart Rate and Heart Rate Variability Measures in Older Men and Women', in *2022 44th Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC)*, Jul. 2022, pp. 3370–3373. doi: 10.1109/EMBC48229.2022.9871559.

[46] E. Sabeti *et al.*, 'Learning Using Concave and Convex Kernels: Applications in Predicting Quality of Sleep and Level of Fatigue in Fibromyalgia', *Entropy*, vol. 21, no. 5, Art. no. 5, May 2019, doi: 10.3390/e21050442.

[47] M. Schmidt, D. Helbig, O. Bhandare, D. Stier, W. Minker, and S. Werner, 'Assessing Objective Indicators of Users' Cognitive Load During Proactive In-Car Dialogs', in *Adjunct Publication of the 27th Conference on User Modeling, Adaptation and Personalization*, in UMAP'19 Adjunct. New York, NY, USA: Association for Computing Machinery, Jun. 2019, pp. 87–91. doi: 10.1145/3314183.3324985.

[48] P. Schmidt, R. Dürichen, A. Reiss, K. Van Laerhoven, and T. Plötz, 'Multi-target affect detection in the wild: an exploratory study', in *Proceedings of the 2019 ACM International Symposium on Wearable Computers*, in ISWC '19. New York, NY, USA: Association for Computing Machinery, Sep. 2019, pp. 211–219. doi: 10.1145/3341163.3347741.

[49] E. E. Van Voorhees *et al.*, 'Ambulatory Heart Rate Variability Monitoring: Comparisons Between the Empatica E4 Wristband and Holter Electrocardiogram', *Psychosom Med*, vol. 84, no. 2, pp. 210–214, Mar. 2022, doi: 10.1097/PSY.0000000000001010.

[50] G. Vila, C. Godin, S. Charbonnier, and A. Campagne, 'Real-Time Quality Index to Control Data Loss in Real-Life Cardiac Monitoring Applications', *Sensors*, vol. 21, no. 16, Art. no. 16, Jan. 2021, doi: 10.3390/s21165357.

[51] 'E4 data - IBI expected signal', *Empatica Support*, Jan. 24, 2020. https://support.empatica.com/hc/en-us/articles/360030058011-E4-data-IBI-expected-signal (accessed Apr. 22, 2023).

[52] 'EmbracePlus | The world's most advanced smartwatch for continuous health monitoring', *Empatica*. https://www.empatica.com/embraceplus (accessed May 28, 2023).

Appendix A: Study participant consent form

Vil du delta i forskningsprosjektet

«A system for reliable biometric data collection in semi-automated migraine prediction»

Dette er et spørsmål til deg om å delta i et forskningsprosjekt hvor formålet er å utforske, utvikle, og teste et system for å samle inn data til forskning på migrene og migrene-prediksjon. I dette skrivet gir vi deg informasjon om målene for prosjektet og hva en eventuell deltakelse vil innebære for deg. Prosjektet er en masteroppgave ved Universitetet i Tromsø, UiT.

Formål

Formålet med prosjektet er å utvikle et stabilt og pålitelig system for innsamling av biometrisk data til bruk i migreneforskning, med særlig fokus på prediksjon av migreneanfall. Systemet skal være lettforståelig og nyttig for brukeren samtidig som det skal sørge for pålitelig innsamling av biometri. Systemet vil bestå av en håndledds-båren sensorenhet (Empatica E4 eller EmbracePlus) for datainnsamling, en mobil-applikasjon for brukerinteraksjon, og en sky-basert datalagringsløsning utviklet ved Institutt for Informatikk ved UiT (mSpider).

Hvem er ansvarlig for forskningsprosjektet?

Fakultetet for naturvitenskap og teknologi, institutt for informatikk ved UiT er ansvarlig for prosjektet.

Hvorfor får du spørsmål om å delta?

Du får spørsmål om å delta grunnet din kjennskap til hvordan det er å leve med migrene.

Hva innebærer det for deg å delta?

Om du velger å delta i prosjektet innebærer det at du gjennomfører en brukstest av systemet (sensor og mobilapp) over en forhåndsavtalt tidsperiode (3-7 dager). Du vil få opplæring i bruk av systemet, samt utlevert alt nødvendig utstyr. Systemet vil automatisk og kontinuerlig samle og lagre biodata fra deg gjennom testperioden. Data som samles er: Puls (blodvolumpuls), hudkonduktivitet (GSR), akselerometerdata (bevegelse) og hudtemperatur. Innsamlet data vil ikke bil direkte analysert, men metadata (data om dataen) vil bli brukt i prosjektet. All innsamlet data vil bli slettet ved prosjektslutt.

Etter gjennomført testperiode ønsker vi at du deltar i et semistrukturert intervju med varighet 30-60 minutter. Intervjuet vil inneholde spørsmål om dine erfaringer ved bruk av systemet og dine vurderinger av systemets design og brukervennlighet. Det tas notater og under intervjuet. Om lydopptak av intervjuet er nødvendig vil dette bli avtalt med deg på forhånd.

Det er frivillig å delta

Det er frivillig å delta i prosjektet. Om du velger å delta, kan du når som helst trekke samtykket tilbake uten å oppgi noen grunn. Alle dine personopplysninger og all innsamlet data vil da bli slettet. Det vil ikke ha noen negative konsekvenser for deg hvis du ikke vil delta eller senere velger å trekke deg.

Ditt personvern – hvordan vi oppbevarer og bruker dine opplysninger

Vi vil bare bruke opplysningene om deg til formålene vi har fortalt om i dette skrivet. Vi behandler opplysningene konfidensielt og i samsvar med personvernregelverket. Prosjektgruppen (student og veiledere) vil være de eneste med tilgang til data, notater og eventuelle lydopptak. Endelig rapport vil leveres inn som en oppgave til Universitetet. Ditt navn vil ikke nevnes i sluttrapporten og vil bli byttet ut med en kode, for eksempel «sluttbruker 1»

Hva skjer med personopplysningene dine når forskningsprosjektet avsluttes?

Prosjektet vil etter planen avsluttes når oppgaven blir levert, 1. juni 2023. Prosjektet kan forlenges om oppgaven skulle bli underkjent. Etter prosjektslutt vil all innsamlet biodata slettes. Eventuelle lydopptak fra intervju vil også bli slettet.

Hva gir oss rett til å behandle personopplysninger om deg?

Vi behandler opplysninger om deg basert på ditt samtykke. På oppdrag fra Institutt for Informatikk ved UiT har Personverntjenester (Sikt) vurdert at behandlingen av personopplysninger i dette prosjektet er i samsvar med personvernregelverket.

Dine rettigheter

Så lenge du kan identifiseres i datamaterialet, har du rett til:

- innsyn i hvilke opplysninger vi behandler om deg, og å få utlevert en kopi av opplysningene
- å få rettet opplysninger om deg som er feil eller misvisende
- å få slettet personopplysninger om deg
- å sende klage til Datatilsynet om behandlingen av dine personopplysninger

Om du har spørsmål til studien, ønsker å vite mer om eller benytte deg av dine rettigheter, ta kontakt med:

Institutt for Informatikk ved Universitetet i Tromsø (UiT).

○ *André Henriksen, andre.henriksen@uit.no, +4777645214 (Hovedveileder)* ○ *Daniel Ursin, daniel.ursin@uit.no, +4741696020 (Student)*

Vårt personvernombud: Sølvi Brendeford Anderssen, personvernombud@uit.no

Om du har spørsmål knyttet til Personverntjenester sin vurdering av prosjektet, kan du ta kontakt med Personverntjenester på epost (personverntjenester@sikt.no) eller på telefon: 53 21 15 00.

Appendix B: System information sheet

Brukerinformasjon for appen «UiT Datainnsamling»

Dette er en enkel bruksanvisning for mobilappen UiT Datainnsamling. Her finner du både nyttig og viktig informasjon om hvordan du bruker appen og gjennomfører testingen. Det vil også bli gitt litt ekstra informasjon om kjente feil og hva som må gjøres om feilene skulle oppstå.

Kontakt Daniel Ursin om du har noen spørsmål.

Tlf.: xxxxxxxx Epost: daniel.ursin@uit.no

Informasjon om appen

Appen UiT Datainnsamling er et førsteutkast til en del av et fremtidig støttesystem for personer med migrene. Det fremtidige systemet vil kunne fungere som en slags automatisert migrenedagbok som skal kunne forutsi (prediktere) mulige migreneanfall innenfor en kort tidshorisont (mindre enn 24 timer). Prediksjonen vil baseres på biometrisk data som samles inn via sensorenheten Empatica E4 som du har fått utlevert. Dataene sendes via mobilappen til servere på UiT. Når nok data er samlet inn vil statistiske metoder og maskinlæring brukes til å finne sannsynligheten for nært forestående anfall. Målet er at brukeren av systemet vil kunne ta medisiner eller på annet vis avverge anfallet før det forekommer.

Et viktig prinsipp for systemet er at det skal være så autonomt som mulig og kreve minst mulig av brukeren. Denne versjonen av UiT Datainnsamling er en førsteutgave (prototype) og har derfor noen begrensninger. Prototypen skal vise frem funksjonalitet, samt teste om innsamling av data er gjennomførbart i dagliglivet, og avanserte funksjoner som prediksjon av migreneanfall er derfor ikke til stede.

Empatica E4

Empatica E4 er en datainnsamlingsenhet laget av det amerikanske helseteknologiselskapet Empatica. Enheten har form og fasong som en smartklokke, men den har ingen skjerm og bare én enkel knapp for interaksjon. Flere forskjellige typer data samles automatisk av enheten:

Akselerasjon i tre akser

Blodvolumpuls (Endringer i blodmengde i huden)

Hudtemperatur

Galvanisk hudrespons (Elektrisk overføringspotensial på huden, GSR)

Tidsintervaller mellom hjerteslag (IBI, beregnes fra blodvolumpuls)

Enheten har ca. 24 timer batterilevetid (i streaming-modus) og den full-lades på under 2 timer. Den kan ikke lades mens datainnsamlingen pågår og slår seg automatisk av under lading. Enheten er ikke vanntett, men tåler vann. Små mengder vann fra håndvask, regn o.l. er greit. Datainnsamlingsenheten kan slås på eller av ved å holde inne knappen på enheten i 3 sekunder.

Hvordan fungerer datainnsamlingen?

Appen er koblingen mellom datainnsamlingsenheten (Empatica E4) og universitetets server. Når datainnsamlingsenheten er koblet til appen gjennom telefonen din vil den automatisk sende data til serveren. Systemet vil bare sende data om det har tilgang til internett gjennom et trådløst nettverk (WiFi). Dataoverføring over mobilnettet (3G/4G/5G) er mulig og kan velges i appens innstillinger. Merk at systemet sender mye data og overføring over mobilnettet kan medføre ekstrakostnader for deg som bruker. Innsamlet data vil bli lagret på enheten så lenge et passende nettverk ikke er tilgjengelig.

Hva trenger du å gjøre?

Du trenger ikke gjøre så mye. Systemet klarer seg selv så lenge datainnsamlingsenheten er tilkoblet appen. Koblingen mellom enhetene er basert på Bluetooth og kan av mange forskjellige grunner bli brutt og du må gjenopprette koblingen. Appen vil gi deg beskjed om dette, men systemet er ikke i stand til å gjenopprette koblingen automatisk.

Hvordan starte datainnsamlingen:

- Installer og start opp appen.
- Appen vil be om tillatelse til å bruke enten Bluetooth eller stedstjenester, alt etter hvordan smarttelefon du har. NB! Appen vil ikke fungere uten denne tillatelsen!
- Logg inn i appen ved å legge inn brukernavnet ditt. Dette steget krever internettilgang.
- Koble datainnsamlingsenheten til appen ved å trykke på den grønne knappen («koble til»). Appen må verifisere datainnsamlingsenheten opp mot en server hos Empatica. Dette krever også internettilgang.
- Hold inne knappen på datainnsamlingsenheten i 2-3 sekunder til lyset på enheten begynner å blinke blått.
- Enheten er tilkoblet når lyse slutter å blinke. Den grønne knappen i appen vil bli rød.
- Sett enheten på svak arm (venstre arm om du er høyrehendt).

Steg 1 og 2 er bare nødvendige første gang appen startes eller om den har vært uten internett i mer enn 7 dager. Datainnsamlingen kan når som helst avsluttes ved å enten slå av datainnsamlings-enheten eller ved å avslutte koblingen med hjelp av rød knapp inne i appen. Datainnsamlingsenheten kan slås av ved å holde inne knappen på enheten i 3 sekunder.

Funksjoner og muligheter i appen

Brukergrensesnittet (UI) it appen gir deg løpende informasjon om deler av din helsetilstand. Hver av de seks informasjonsboksene kan trykkes på for å få frem mer detaljert informasjon. Denne delen av UI-et er også konfigurerbar og du kan velge hvilken informasjon du ønsker å følge. Et langt trykk i utkanten av informasjonsboksene vil ta frem konfigurasjonsmenyen. Denne funksjonaliteten er noe begrenset i prototypen. Når appen er installert vil den fortelle deg hva den holder på med gjennom en vedvarende notifikasjon. Denne vil endre farge basert på statusen til systemet:

Rød: Ingen datainnsamling. Systemet venter.

Gul: Datainnsamlingsenheten er koblet til, men den er ikke på armen. Ingen datainnsamling.

Grønn: Datainnsamlingsenheten er på armen. Datainnsamlingen er i gang.

Appen samler som kjent biometrisk data automatisk, men for at dataene skal kunne brukes til forskning må de kobles sammen med hendelser i den virkelige verden. Merknadsfunksjonen i appen er et viktig verktøy som lar deg merke av viktige hendelser i livet ditt som er knyttet til migrene. Slike hendelser kan være starttidspunktet for et migreneanfall eller en stressende enkelthendelse i hverdagen. Merknader kan settes ved å kjapt trykke på knappen på datainnsamlingsenheten. En tom merknad med tid og dato vil dukke opp nederst i appen. Denne kan når som helst fylles inn med detaljer om hendelsen og sendes til datalagringsserveren. Merknader kan slettes og blir ikke sendt før du har gitt klarsignal.

Kjente feil og svakheter

Datainnsamlingsenheten har kort tilkoblingsrekkevidde og vil raskt avslutte tilkoblingen med appen om avstanden mellom enhetene blir for stor. Innsamlingsenheten slår seg automatisk av og må manuelt kobles til på nytt (se del 5). Appen vil gi beskjed om dette og automatisk gå i frakoblet modus.

Appen og systemet er fremdeles under utvikling og kan avslutte seg selv av ukjente grunner. Normalt sett må appen da startes på nytt og innsamlingsenheten må kobles til. All data som er lagret av appen vil fremdeles være tilgjengelig etter en «krasj». Dette inkluderer også innloggingsinformasjon og konfigurasjoner.

Appen har noen ganger vanskeligheter med å sende data til serveren. Dette skyldes vanligvis at serveren ikke er tilgjengelig eller at appen ikke skjønner at den har tilgang til internett. Dette er problemer som normalt sett «går over av seg selv». Ta kontakt med oss om appen har vedvarende problemer med opplasting eller om innsamlet data tar for mye plass på mobiltelefonen.

Appen kan ha problemer med å sende data til serveren gjennom noen VPN-tjenester. Om du ofte bruker VPN-tjenester, sørg for at disse er avslått en periode av dagen (~1 time).

Datainnsamlingsenheten kan av og til «glemme» hva den selv heter. Den kan derfor ikke verifiseres mot serveren hos Empatica og den kan ikke brukes til datainnsamling. Appen oppdager dette, gir deg beskjed og kommer med en løsning. Følg instruksene på skjermen for å løse problemet.

Smarttelefonen og datainnsamlingsenheten kan ha kommunikasjonsproblemer når andre Bluetooth-enheter er tilkoblet. Datainnsamlingsenheten vil da avslutte tilkoblingen.

Appendix C: Interview guide

Intervjuguide

Generell systembruk:

1. Hvordan var din generelle opplevelse av systemet du har testet? Anonymt? På dine premisser?

2. Har systemet noen utpekt negative funksjoner? Beskriv? Er det greit å få så mye detaljert informasjon?

3. Har systemet noen utpekt positive funksjoner? Beskriv? Overraskelser?

4. Hvor mye tid brukte du ufrivillig på systemet om dagen? Var systemet for krevende? Ville du brukt systemet på fulltid?

5. Har systemet noen mangler? Beskriv? Hva kan forbedres? Hvordan kan det forbedres?

Brukergrensesnitt (UI):

1. Har du brukt brukergrensesnittet? Mye? Til hva?

2. Er UI-et forståelig? Kunne gjort noe annerledes? Mer info? Mindre info?

3. Var informasjonen i UI-et nyttig? Hvordan?

4. Det mangler en del informasjon i deler av UI-et. Hvilken informasjon kunne du tenkt deg å finne der? Samme hver gang? Interaktivitet? Eksterne ressurser?

Appendix D: Correspondence with DPO

Tirsdag 21. mars 23 20:55
Fra: daniel.ursin@uit.no
Til: personvernombud@uit.no

Hei,
Jeg er student på Institutt for Informatikk og utvikler et system for innsamling av helsedata i min masteroppgave. I den forbindelse ønsker jeg noen tips om hvordan jeg burde forholde meg til personvern. Jeg planlegger å gjennomføre brukertesting på sluttbrukere, noe som vil innebære innsamling av biodata gjennom en smartklokke (Empatica E4) fra et lite antalltestpersoner (1-3). Siden antallet brukere er så lavt, vil det ikke være nødvendig å lagre noen persondata i forbindelse med datainnsamlingen. All innsamlet data vil bli lagret på campus på UiTs servere og vil ikke mellomlagres i systemer eid av tredjepartsaktører (Google etc.). Formålet medtestingen er å kontrollere at systemet er i stand til å samle inn data under realistiske forhold. Innsamlet data vil ikke bli direkte vurdert eller analysert i løpet av prosjektet da det er innsamlingsprosessen i seg selv som er fokuset. All data vil bli slettet ved prosjektslutt i juni.
Burde jeg melde prosjektet til Sikt? Burde jeg også kontakte REK for en vurdering? Er det noen andre ting jeg burde tenke på?
Prosjektbeskrivelse og et utkast til en samtykkeerklæring er vedlagt.
På forhånd takk!

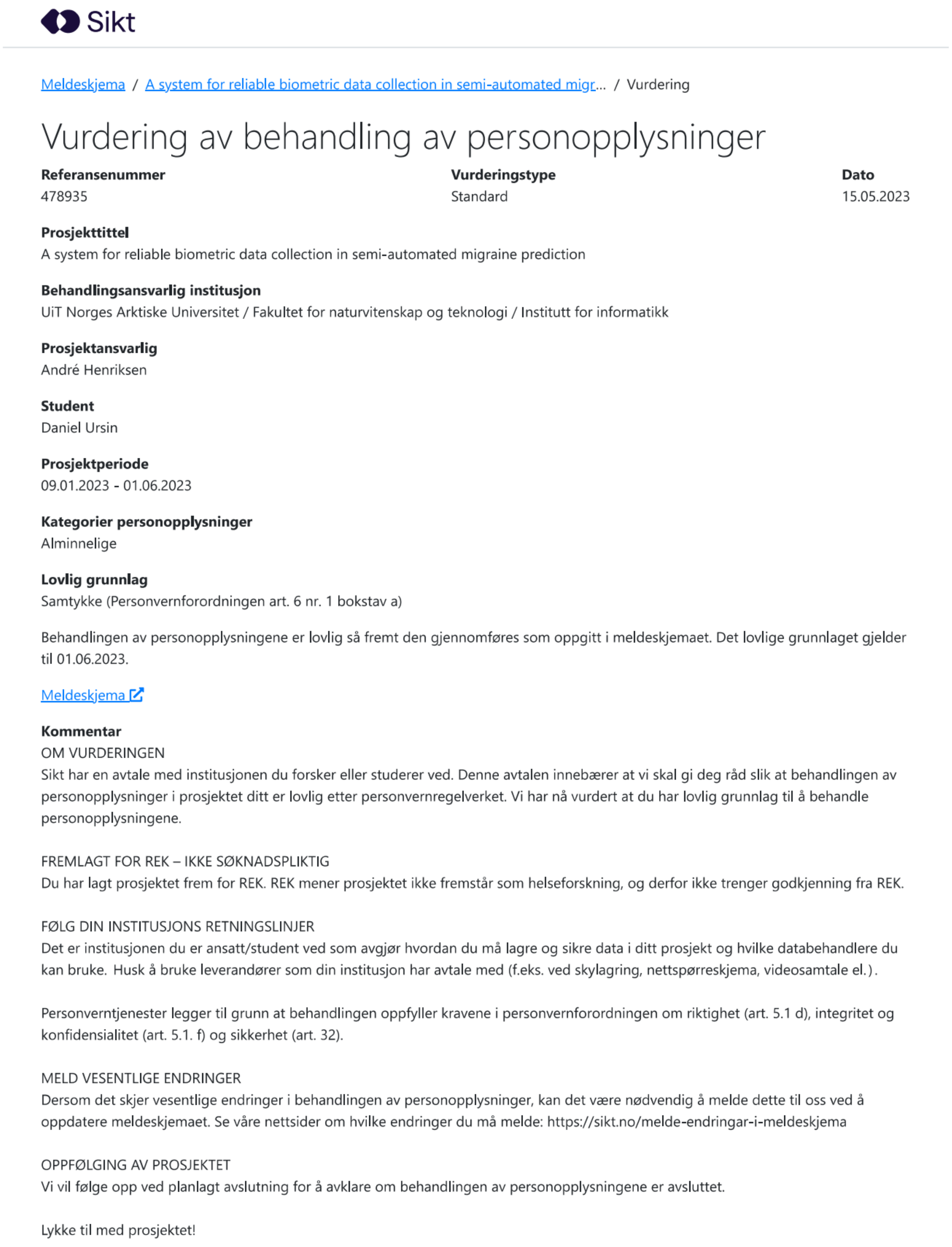
Vennlig hilsen,
Daniel Ursin

Onsdag 12. april 23 13:12
Fra: personvernombud@uit.no
Til: daniel.ursin@uit.no
Hei og beklager svært seint svar.

Jeg anbefaler deg å ta kontakt med Sikt for å sjekke om det bør meldes inn. Dere samler inn personopplysninger som dere ikke skal bruke, så da blir jeg i tvil om meldeplikten.
Samtykkeerklæring og prosjektbeskrivelsen er fin.

Hilsen Sølvi Brendeford Anderssen, fungerende personvernombud

Appendix E: Project evaluation by REK



Sikt

Meldeskjema / A system for reliable biometric data collection in semi-automated migr... / Vurdering

Vurdering av behandling av personopplysninger

Referansenummer	Vurderingstype	Dato
478935	Standard	15.05.2023

Prosjekttittel
A system for reliable biometric data collection in semi-automated migraine prediction

Behandlingsansvarlig institusjon
UiT Norges Arktiske Universitet / Fakultet for naturvitenskap og teknologi / Institutt for informatikk

Prosjektansvarlig
André Henriksen

Student
Daniel Ursin

Prosjektperiode
09.01.2023 - 01.06.2023

Kategorier personopplysninger
Alminnelige

Lovlig grunnlag
Samtykke (Personvernforordningen art. 6 nr. 1 bokstav a)

Behandlingen av personopplysningene er lovlig så fremt den gjennomføres som oppgitt i meldeskjemaet. Det lovlige grunnlaget gjelder til 01.06.2023.

Meldeskjema

Kommentar
OM VURDERINGEN
Sikt har en avtale med institusjonen du forsker eller studerer ved. Denne avtalen innebærer at vi skal gi deg råd slik at behandlingen av personopplysninger i prosjektet ditt er lovlig etter personvernregelverket. Vi har nå vurdert at du har lovlig grunnlag til å behandle personopplysningene.

FREMLAGT FOR REK – IKKE SØKNADSPLIKTIG
Du har lagt prosjektet frem for REK. REK mener prosjektet ikke fremstår som helseforskning, og derfor ikke trenger godkjenning fra REK.

FØLG DIN INSTITUSJONS RETNINGSLINJER
Det er institusjonen du er ansatt/student ved som avgjør hvordan du må lagre og sikre data i ditt prosjekt og hvilke databehandlere du kan bruke. Husk å bruke leverandører som din institusjon har avtale med (f.eks. ved skylagring, nettspørreskjema, videosamtale el.).

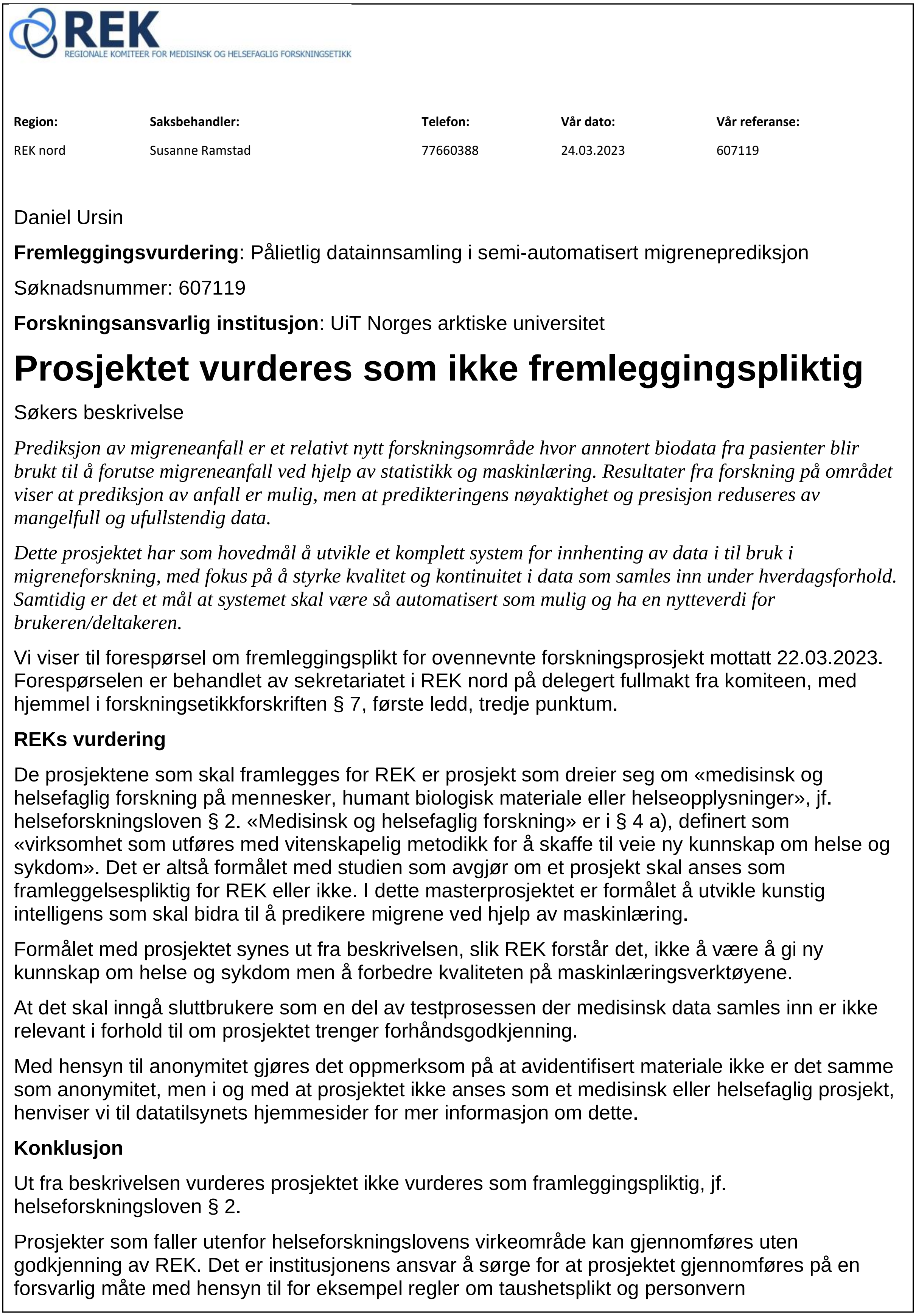
Personverntjenester legger til grunn at behandlingen oppfyller kravene i personvernforordningen om riktighet (art. 5.1 d), integritet og konfidensialitet (art. 5.1. f) og sikkerhet (art. 32).

MELD VESENTLIGE ENDRINGER
Dersom det skjer vesentlige endringer i behandlingen av personopplysninger, kan det være nødvendig å melde dette til oss ved å oppdatere meldeskjemaet. Se våre nettsider om hvilke endringer du må melde: https://sikt.no/melde-endringar-i-meldeskjema

OPPFØLGING AV PROSJEKTET
Vi vil følge opp ved planlagt avslutning for å avklare om behandlingen av personopplysningene er avsluttet.

Lykke til med prosjektet!

Appendix F: Data protection evaluation by SIKT



REK
REGIONALE KOMITEER FOR MEDISINSK OG HELSEFAGLIG FORSKNINGSETIKK

Region:	Saksbehandler:	Telefon:	Vår dato:	Vår referanse:
REK nord	Susanne Ramstad	77660388	24.03.2023	607119

Daniel Ursin

Fremleggingsvurdering: Pålietlig datainnsamling i semi-automatisert migreneprediksjon

Søknadsnummer: 607119

Forskningsansvarlig institusjon: UiT Norges arktiske universitet

Prosjektet vurderes som ikke fremleggingspliktig

Søkers beskrivelse

Prediksjon av migreneanfall er et relativt nytt forskningsområde hvor annotert biodata fra pasienter blir brukt til å forutse migreneanfall ved hjelp av statistikk og maskinlæring. Resultater fra forskning på området viser at prediksjon av anfall er mulig, men at predikteringens nøyaktighet og presisjon reduseres av mangelfull og ufullstendig data.

Dette prosjektet har som hovedmål å utvikle et komplett system for innhenting av data i til bruk i migreneforskning, med fokus på å styrke kvalitet og kontinuitet i data som samles inn under hverdagsforhold. Samtidig er det et mål at systemet skal være så automatisert som mulig og ha en nytteverdi for brukeren/deltakeren.

Vi viser til forespørsel om fremleggingsplikt for ovennevnte forskningsprosjekt mottatt 22.03.2023. Forespørselen er behandlet av sekretariatet i REK nord på delegert fullmakt fra komiteen, med hjemmel i forskningsetikkforskriften § 7, første ledd, tredje punktum.

REKs vurdering

De prosjektene som skal framlegges for REK er prosjekt som dreier seg om «medisinsk og helsefaglig forskning på mennesker, humant biologisk materiale eller helseopplysninger», jf. helseforskningsloven § 2. «Medisinsk og helsefaglig forskning» er i § 4 a), definert som «virksomhet som utføres med vitenskapelig metodikk for å skaffe til veie ny kunnskap om helse og sykdom». Det er altså formålet med studien som avgjør om et prosjekt skal anses som framleggelsespliktig for REK eller ikke. I dette masterprosjektet er formålet å utvikle kunstig intelligens som skal bidra til å predikere migrene ved hjelp av maskinlæring.

Formålet med prosjektet synes ut fra beskrivelsen, slik REK forstår det, ikke å være å gi ny kunnskap om helse og sykdom men å forbedre kvaliteten på maskinlæringsverktøyene.

At det skal inngå sluttbrukere som en del av testprosessen der medisinsk data samles inn er ikke relevant i forhold til om prosjektet trenger forhåndsgodkjenning.

Med hensyn til anonymitet gjøres det oppmerksom på at avidentifisert materiale ikke er det samme som anonymitet, men i og med at prosjektet ikke anses som et medisinsk eller helsefaglig prosjekt, henviser vi til datatilsynets hjemmesider for mer informasjon om dette.

Konklusjon

Ut fra beskrivelsen vurderes prosjektet ikke vurderes som framleggingspliktig, jf. helseforskningsloven § 2.

Prosjekter som faller utenfor helseforskningslovens virkeområde kan gjennomføres uten godkjenning av REK. Det er institusjonens ansvar å sørge for at prosjektet gjennomføres på en forsvarlig måte med hensyn til for eksempel regler om taushetsplikt og personvern

Vi gjør oppmerksom på at ovennevnte ikke er et vedtak men en vurdering og konklusjon som må anses som veiledende jf. forvaltningsloven § 11.

Med vennlig hilsen

May Britt Rossvoll Susanne Ramstad

Komitesekretær REK nord *Seniorrådgiver REK nord*

Kopi til:

UiT Norges arktiske universitet