**Univerzita Karlova**
**1. lékařská fakulta**

Studijní program: Doktorský
Studijní obor: Biomedicínská informatika

UNIVERZITA KARLOVA
1. lékařská fakulta

**Ing. Miroslav Mužný**

# Elektronický systém pro podporu provádění klinických studií s možností zpracování dat pomocí umělé inteligence

# Electronic clinical study management system with artificial intelligence-based data processing capabilities

Disertační práce

Školitel: Ing. Jan Mužík, PhD.
Konzultant: Prof. Eirik Årsand

Praha, 2020

**Prohlášení:**

Prohlašuji, že jsem závěrečnou práci zpracoval/a samostatně a že jsem řádně uvedl/a a citoval/a všechny použité prameny a literaturu. Současně prohlašuji, že práce nebyla využita k získání jiného nebo stejného titulu.

Souhlasím s trvalým uložením elektronické verze mé práce v databázi systému meziuniverzitního projektu Theses.cz za účelem soustavné kontroly podobnosti kvalifikačních prací.

V Praze,

Ing. Miroslav Mužný

Identifikační záznam:

MUŽNÝ, Miroslav. *Elektronický systém pro podporu provádění klinických studií s možností zpracování dat pomocí umělé inteligence. [Electronic clinical study management system with artificial intelligence-based data processing capabilities]*. Praha, 2020. Univerzita Karlova, 1. lékařská fakulta. Školitel Ing. Jan Mužík, Ph.D.

# Preface

The research described in this document was conducted between September 2012 and August 2020 under the supervision of Ing. Jan Mužík, Ph.D. and Prof. Eirik Årsand.

Spin-off Companies and Research Results Commercialization Centre, The First Faculty of Medicine, Charles University in Prague, and the Norwegian Centre for E-health Research, Tromsø, Norway, were involved in close cooperation the whole period of my Ph.D. study.

To the best of my knowledge, the research described in this work is original, except for parts where the references and acknowledgements are stated.

Miroslav Mužný
December 2020

## Statement of Contribution

Hereby I declare my own contribution to the research published via co-authored publications, that lay the foundation for this thesis.

| **Publication:** M. Muzny, A. Henriksen, A. Giordanengo, J. Muzik, A. Grøttland, H. Blixgård, G. Hartvigsen and E. Årsand, "Wearable sensors with possibilities for data exchange: Analyzing status and needs of different actors in mobile health monitoring systems," *International journal of medical informatics,* vol. 133, 2020. | | | |
|---|---|---|---|
| | Contribution | | |
| | minor | major | predominant |
| conceptual design of the research idea | | | X |
| development of the research plan | | X | |
| practical realization | | | X |
| data acquisition | | | X |
| data analyses/interpretation | | | X |
| writing of the manuscript | | | X |

| **Publication:** E. Årsand, M. Muzny, M. Bradway, J. Muzik and G. Hartvigsen, "Performance of the first combined smartwatch and smartphone diabetes diary application study," *Journal of diabetes science and technology,* vol. 9, no. 3, pp. 556-563, 2015. | | | |
|---|---|---|---|
| | Contribution | | |
| | minor | major | predominant |
| conceptual design of the research idea | | | X |
| development of the research plan | | X | |
| practical realization | | | X |
| data acquisition | | | X |
| data analyses/interpretation | | X | |
| writing of the manuscript | | X | |

| **Publication:** E. Årsand, M. Bradway, H. Blixgård, M. Muzny, A. Giordanengo, A. Grøttland and G. Hartvigsen, "Experience from using a dynamic study management service for an mHealth diabetes type 2 RCT," *DIABETES TECHNOLOGY & THERAPEUTICS,* vol. 20, p. 73, 2018. | | | |
|---|---|---|---|
| | Contribution | | |
| | minor | major | predominant |
| conceptual design of the research idea | | X | |
| development of the research plan | | | X |
| practical realization | | | X |
| data acquisition | | | X |
| data analyses/interpretation | | X | |
| writing of the manuscript | | X | |

| **Publication:** A. Henriksen, M. H. Mikalsen, A. Z. Woldaregay, M. Muzny, G. Hartvigsen, L. A. Hopstock and S. Grimsgaard, "Using Fitness Trackers and Smartwatches to Measure Physical Activity in Research: Analysis of Consumer Wrist-Worn Wearables.," *Journal of medical Internet research,* vol. 20, no. 3, 2018. | | | |
|---|---|---|---|
| | Contribution | | |
| | minor | major | predominant |
| conceptual design of the research idea | X | | |
| development of the research plan | X | | |
| practical realization | X | | |
| data acquisition | | X | |
| data analyses/interpretation | | X | |
| writing of the manuscript | X | | |

**Abstract**

An increasing amount of data are collected through wearable devices during ambulatory, and long-term monitoring of biological signals, adoption of persuasive technology and dynamics of clinical trials information sharing – all of that changes the possible clinical intervention. Moreover, more and more smartphone apps are hitting the market as they become a tool in daily life for many people around the globe. All of these applications are generating a tremendous amount of data, that is difficult to process using traditional methods, and asks for engagement of advanced methods of data processing.

For recruiting patients, this calls for a shift from traditional methods of engaging patients to modern communication platforms such as social media, that are providing easy access to up-to-date information on an everyday basis. These factors make the clinical study progression demanding, in terms of unified participant management and processing of connected digital resources.

Some clinical trials put a strong accent on remote sensing data and patient engagement through their smartphones. To facilitate this, a direct participant messaging, where the researchers give support, guidance and troubleshooting on a personal level using already adopted communication channels, needs to be implemented. Since the process of such support is very time-consuming and often difficult to assure quality, it calls for revisiting methods for performing a randomized control trial.

A system named *Hubro* for electronic management of clinical studies to address many of the identified challenges has been created. The system supports following processes throughout the study: recruitment, randomization, follow up via messaging, automatic usage data collection and patient self-reported data collection through electronic questionnaires – all accessible from one single user interface. Data collected through tailored questionnaires is what is essentially defined as electronic patient-reported outcome per today, which is currently on the rise in a health research setting.

The modular design of the presented system assures the possibility of functionality enhancement in various levels of the data processing chain. The system has loosely coupled

architecture and utilizes pluggable computation modules, with support of artificial intelligence. Therefore, the system can be extended with additional sources data, but also with data post-processing capabilities, which is useful when discovering various insights or detecting specific patterns.

So far, two studies have used this system – with 50 and 8 participants. The Hubro electronic study management system has significantly improved the feasibility of these interventions through streamlined workflow, electronic messaging and data collection support. The Hubro system has saved a considerable amount of time to researchers when managing the study. The usability of the administration interface of the system on various devices such as smartphones and tablets also allowed a quick turnaround when reacting on requests of participants. Although, researchers have reported a steep learning curve for third-party tools integrated with Hubro that are used for usage analytics and questionnaire distribution. Also, as primary candidates for updates in future revisions, they identified an expansion of existing communication capabilities by a secure communication channel directing from participants to researchers; implementation of tools assisting data extraction and formatting; and introduction of reminders and recruitment planning functions.

Keywords: mHealth, clinical study, automatic data collection, mobile application, electronic questionnaire, wearable device

**Abstrakt**

Objem dat, který je generován nositelnými zařízeními v průběhu ambulatorního i dlouhodobého snímání biologických signálů, adopce pervazivních technologií a dynamika předávání informací v rámci klinických studií – to vše mění způsoby, kterým mohou prováděny klinické studie. Více a více aplikací, které přicházejí na trh se stávají pomůckou v denním životě lidí po celém světě. Všechny tyto aplikace produkují obrovské množství dat, jež je obtížné zpracovat tradičními metodami, a vyvstává tak nutnost využití pokročilých metod.

Je také možné sledovat odvrat od tradičních metod náboru pacientů, k moderním komunikačním platformám jako sociální sítě, které usnadňují přístup k aktuálním informacím. Tyto faktory činí postup v klinické studii náročným s ohledem na management účastníků studie a zpracování informací ze zdrojů dat.

Některé klinické studie kladou velký důraz na sběr dat ze senzorů a zapojení pacientů do studie prostřednictvím jejich mobilních telefonů. Pro usnadnění tohoto přístupu, je nutné využít přímou komunikací s pacientem, kdy administrátoři studie poskytují podporu a pomáhají řešit problémy, které se mohou v průběhu studie vyskytnout, a to za pomocí moderních komunikačních platforem a elektronických zpráv vedených přímo s účastníkem studie. Celý tento postup je nicméně časově náročný, a je tedy nutné přehodnotit způsob provádění randomizované kontrolované studie.

Byl vytvořen elektronický systém *Hubro* pro podporu provádění klinických studií, který adresuje tyto nově vzniklé požadavky. Tento nově vyvinutý systém podporuje: nábor účastníků, randomizaci, zasílání zpráv, automatický sběr dat o používání aplikace a uživatelských skrze online dotazníkový systém – tato funkcionalita je integrována v rámci jednoho uživatelského rozhraní. Data, shromážděna z elektronických dotazníků, jsou v zásadě obdobou elektronických formulářů výsledků hlášených pacientem, jako způsobu sběru dat, který je v současné době na vzestupu v oblasti výzkumu eHealth.

Modulární design zajišťuje možnost rozšiřitelnosti funkčnosti na různých úrovních řetězce zpracování dat. Architektura systém je postavena na volně vázaných komponentech, a používá zásuvné moduly s podporou umělé inteligence. Systém tak může být rozšířen o dodatečné

zdroje dat, ale také o možnost post-processingu dat, který napomáhá při zpracování úloh typu detekce vzorů.

Systém Hubro byl prozatím využit v rámci dvou studií – s 50-ti a 8-mi účastníky. Tento systém výrazně usnadnil provádění těchto klinických studií díky zjednodušenému modelu administrace studie, systému pro zasílání zpráv a podpoře pro sběr dat. Používání systému Hubro zároveň ušetřilo množství času při administraci studie, díky použitelnosti administračního rozhraní na různých typech zařízení (smartphone, tablet, PC), které umožnilo rychlejší reakce na požadavky. Výzkumníci, kteří tento systém používali, nicméně zaznamenali poměrně strmou křivku učení při používání nástrojů třetích stran integrovaných do systému Hubro, které jsou používány pro sběr uživatelských dat o používání aplikace a distribuci dotazníků. Zároveň byly výzkumníky navrhnuty následující vylepšení – dvoucestná komunikace integrovaná v systému Hubro, implementace nástrojů usnadňujících následnou extrakci dat a formátování a implementace funkcí pro plánování náboru účastníků studie.

Klíčová slova: mHealth, klinická studie, automatický sběr dat, mobilní aplikace, elektronický dotazník, nositelné zařízení

# Table of Contents

# Introduction

The conducted research has been described and interpreted in detail in separate publications. They are referred to in a text for further reference as:

- **Publication 1.** - M. Muzny, A. Henriksen, A. Giordanengo, J. Muzik, A. Grøttland, H. Blixgård, G. Hartvigsen and E. Årsand, "Wearable sensors with possibilities for data exchange: Analyzing status and needs of different actors in mobile health monitoring systems," *International journal of medical informatics,* vol. 133, 2020. IF 3.025. [1]
- **Publication 2.** - E. Årsand, M. Muzny, M. Bradway, J. Muzik and G. Hartvigsen, "Performance of the first combined smartwatch and smartphone diabetes diary application study," *Journal of diabetes science and technology,* vol. 9, no. 3, pp. 556-563, 2015. [2]
- **Publication 3.** - E. Årsand, M. Bradway, H. Blixgård, M. Muzny, A. Giordanengo, A. Grøttland and G. Hartvigsen, "Experience from using a dynamic study management service for an mHealth diabetes type 2 RCT," *DIABETES TECHNOLOGY & THERAPEUTICS,* vol. 20, p. 73, 2018. [3]
- **Publication 4.** - A. Henriksen, M. H. Mikalsen, A. Z. Woldaregay, M. Muzny, G. Hartvigsen, L. A. Hopstock and S. Grimsgaard, "Using Fitness Trackers and Smartwatches to Measure Physical Activity in Research: Analysis of Consumer Wrist-Worn Wearables.," *Journal of medical Internet research,* vol. 20, no. 3, 2018. IF 5.03. [4]

## Introduction to Modern Clinical Trials Design

The diversity of various mobile health (mHealth) -enabled clinical trials puts a strong accent, especially on the possibility of individual handling of participants data in different stages of a study. In ordinary cases, this need is covered by direct participant messaging, which supports the guidance and troubleshooting on a personal level using already adopted communication channels. The situation calls for a revisit of methods of performing randomized controlled trial (RCT) [5].

Individuals are also becoming more empowered to take responsibility for their health. Besides using various health applications, they have adopted a wide portfolio of wearable self-management devices as well [6]. Such a setting, where individuals use multiple, interconnected devices, however increases demands for assessment and processing during the clinical intervention. These demands have to be addressed to ensure that proper data-processing and communication support tools are available for researchers that run a study and also for patients that participate in a study.

The patient-reported outcomes can be gathered in multiple ways. The primary ones are through the traditional paper-based diaries, interactive voice record (IVR) systems and electronic patient-reported outcome (ePRO) devices [7]. IVR systems have been used intensively in clinical studies as a cost-saving method of patient screening. In a basic principle, the patient calls into an IVR system, or an outbound call is placed to the patient, who interacts with the system using a tone-dial or voice commands. The disadvantages include hard to understand instructions or an information overload [8]. ePROs are collected through online questionnaires, that can be conveniently processed on different personal devices (tablets, smartphones, laptops). Therefore, ePRO has the potential to eliminate drawbacks of IVR systems by a built-in input validation. Reminders can be delivered through various channels such as email or in-app notifications [9].

Besides ePROs, multiple other observables are forming the input when performing a mobile health intervention. Examples of these observables are mobile application usage data, self-registered data and health data collected from connected wearable devices with possibilities for data exchange. While usage data and self-registered data collection and processing is a deterministic task, to better understand wearable device's data exchange, it is first necessary to establish a framework that describes current data exchange possibilities due to heterogeneous capabilities of various wearable devices.

We have created a system named *Hubro* for electronic management of clinical studies to address identified challenges. The system supports following processes throughout the study: recruitment, randomization, follow up via messaging, automatic data collection and patient self-reported data collection through electronic questionnaires. In particular, we reviewed current health data exchange capabilities of wearable devices, and we describe how different actors in mobile health monitoring systems relate to these wearable sensor systems.

## Wearable Sensors with Possibilities for Health Data Exchange (Publication 1.)

Sensors and wearables have become mutually emerging categories. Meaning of both terms is often interchangeable in various resources. In this work, we understand a sensor to be an electronic component used for measuring of specific physiological health parameter. By a wearable we distinguish a consumer-oriented device integrating one or multiple sensors. First experiments with wearable devices are dated back to 80‘s. A hearing aid or calculator watch can be considered early examples of wearable technology as we know it today. During the time the wearable technology evolved and became part of ubiquitous computing, where a computing capability is integrated into most of the things of daily life. Monitoring of physiological health parameters via wearable sensors is closely connected to a movement of continuous logging of various aspects of persons’ daily life into a quantifiable form, which is known as quantified-self.

The important position of wearable devices in daily life is underlined by the fact that they can act as a source of inadvertently collected data throughout the day because of their pervasiveness. They often come equipped with multiple health sensors with a possibility of continuous body parameter and vital signs measurement, and they often wirelessly communicate with user’s smartphone, showing users graphically interpreted aggregated data on a smartphone’s screen. As wearable devices became a widely adopted piece of technology, it also makes them an essential piece of the technology stack performing and carrying out a patient intervention in a real-life setting.

Although wearable may provide rich intra-day feedback to the consumer, processing of the data collected by various sensors integrated into wearable devices is often black-boxed in a way, that the end-user only sees the final synthesized measurements. Therefore, the validity of the data has to be tested without knowledge of internally working algorithms [10, 11]. The demand for devices with validated/tested measurement methods is steered by an expanding portfolio of associated health data analytics services. Besides, many associated companion applications provide advanced statistics, suggestions, and prognosis, which can be used by many users as a recommendation for future medical-related actions. Therefore, these applications should be classified as medical devices [12].

To set a foundation for documenting data exchange possibilities, we have identified a framework [1], that describes data flow between different data collection systems. The basic block of the framework is the wearable device as a source of data. The framework also contains systems such as Health platforms (Google Fit [13], Apple Health [14]), third party fitness services, device's cloud service, middleware and Electronic Health Record (EHR). Device's data transfer capabilities are, however, often influenced by made-by-design constraints. Manufacturers tend to implement their own proprietary Application Programming Interfaces (API) for transferring the data. In a world of mobile applications backed with cloud support, this allows device producers to control the flow of data from the initial point of transmission, leaving out a possibility for implementation of e-health solutions through standardized protocols.

From a high-level point of view, the current status in the area of wearable technologies and their data exchange capabilities makes integration with various services a non-trivial task, only handled on an individual basis without a prospect for systematical approach. This situation motivates patients to implement their own solutions with functionality tailored for self-management of the specific disease, or condition. This approach is called Do-It-Yourself (DIY) and represents a community-based development driven by a lack of official solutions serving the desired purpose [15]. Tutorials, guides, source codes and other resources related to a specific project are shared via publicly accessible channels, often through social networks, with an engaged community collectively improving these solutions. Probably a most well-known example of such a DIY solution is NightScout – an open-source project assisting people with diabetes getting real-time access to a continuous glucose monitor (CGM) data [16]. The NightScout project consists of both hardware and software updates to commercial CGM devices, giving timely access to the CGM data on a computer screen, smartwatch or other devices. DIY projects, such as NightScout, have a nature of a non-commercial, open-source project and therefore, they are being neither reviewed nor approved by any regulatory authority. Concerns are given to the process of code review, safety monitoring and provision of support, which usually constitutes mainly of web pages with solutions to common problems and support discussion forums [17]. Ultimately, there is a pressure coming from the DIY community back to service providers to implement presented functions into their commercial solutions.

### Summary of Identified Challenges in the Area of Mobile Health Interventions

Based on the findings, we have identified and addressed the following challenges:

- the current status around wearable devices signalizes lack of systematic health data exchange capabilities. Wearable devices generate a vast amount of data from their integrated sensors and are an essential input of mobile interventions. However, the support for exchange of this data is limited
- commercially-available wearable devices do not fit into the framework of mobile electronic health (eHealth) self-management tools only as a passive source of health data. However, in some cases, they can complement the functionality of a mobile device and improve patients comfort when dealing with a chronic disease such as diabetes. To prove this, we have designed, developed and tested the usability of the first combined smartwatch-smartphone diabetes diary application
- there is a need for updated evaluation approaches for eHealth and mHealth interventions, that would provide not only survey-based instruments but also integrate a data collection framework, that would support researchers addressing the challenges above. To address this issue, we have created and tested a research tool for the management of mHealth and eHealth studies

## Methods

We have designed and implemented a system for electronic study management, Hubro, and also newly developed accompanying smartwatch application. We have demonstrated its usability with an existing smartphone application *Diabetes Diary* [18]. Diabetes Diary is a self-management tool for patients with diabetes, developed by the Norwegian Centre for E-health Research (Tromsø, Norway) in cooperation with Spin-off Companies and Research Results Commercialization Centre, The First Faculty of Medicine, Charles University in Prague (Prague, Czech Republic). Its primary function set consists of glucose, carbohydrates, insulin, medication and physical activity self-tracking; visualization of the data; decision support through insulin dose recommendation based on similar situations and integration with the RunKeeper fitness platform [19]. The Diabetes Diary is currently available worldwide.

Although the application supports both the iOS and Android platform, the functionality slightly differs.

The Diabetes Diary application is a research-based tool, developed with user involvement and participation during the process. Due to the nature of the development process, the application has several flavours/variants, used across different studies with baseline functions that are present at all of them. Also, some of the functions are hidden for regular users and can be unlocked by inserting a special code. Such functions may still be in testing. Diabetes Diary integrates with the RunKeeper platform – a third-party fitness service, that allows us to connect with multiple fitness tracking devices. RunKeeper is connected via its REST API, and the Diabetes Diary pulls physical activity records into its database. This way, the Diabetes Diary users have a freedom of choice among the physical activity tracking devices that are supported by the RunKeeper platform, or they can use the GPS-equipped smartphone with the RunKeeper application to track these activities.

To demonstrate the usability of smart wearable devices with physical activity tracking capabilities in a self-management of chronic diseases, we have integrated the Diabetes Diary smartphone application with a smartwatch, resulting in the development of the first combined smartwatch/smartphone diabetes application [2]. The novel concept of combined smartwatch/smartphone diabetes application enabled users to utilize the potential of the wearable device by making new registrations and having them synchronized with the smartphone; getting timely reminders for blood glucose check; having an estimated step count transferred to the smartphone and segmented into individual physical activities. The application was developed for nine months using agile development methods with three diabetes patients continuously testing the application. Agile development methods allow an incremental model, which is based on a tight collaboration within the team and stakeholders and using dynamic interactions, it provides continuous delivery contrary to traditional methods of development, such as the Waterfall [20]. In a research setting, it is desirable to adhere to dynamically changing requirements flexibly, and therefore we chose Agile. The performance evaluation of the combined smartwatch/smartphone diabetes application was assessed using a questionnaire, where users reported increased usability and provided specific suggestions for improvements. Indeed, the lack of data collection infrastructure and nonexistent usage analytics platform served as additional indicators for a need for an update of research evaluation methods.

We have shown that for consumer-based physical activity trackers and smartwatches, the number of new devices coming to the market every year is high and increasing [4]. With analysis interoperability, we have documented data-exchange capabilities of 362 wearable devices in terms of data transfer protocols, communication interfaces, integration with PC/smartphone, access to user data, developer-access to the device, and market status. Based on this analysis, we created a framework, which we further use to describe relations between different components in terms of communication capabilities. This framework allowed us to better understand issues with interventions using mobile and wearable devices, from the standpoint of different actors in mobile health monitoring systems: EHR providers, software developers, and patient users.

Piwek et al. identified three reasons that limit the extensive growth and further use of mobile devices for research [21]:

- programming barriers
- consenting issues
- concerns regarding privacy and data security

The new Hubro platform was created based on needs to perform studies dynamically and fully electronically. With a massive use of social networks, email and electronic communication in general, the online recruitment process can be much more time-effectively than during an ordinary RCT [22]. We can reach potential participants through electronic channels and provide them with a link to an electronic resource (website) where they can get more information about the study and show their interest to participate if they find the study relevant. From this point, the study manager can guide the participants through the whole study process by directly sending them personalized messages. Participants can be randomized into different groups by a built-in block randomization routine or manually by the study manager. The study manager is also able to provide participants with electronic questionnaires in different phases of the study. Hubro research platform also provides tools for a data collection from the software which study participant uses during the study, i.e. the subject of intervention (e.g. smartphone application). The potential of Hubro research is in its' modularity, expandability and re-usability for different intervention types.

Figure 1. illustrates all steps of study:

1. Electronic recruitment of participants through e-mail, social networks, web and other media
2. Informed Consent forms, distribution and reception of answers
3. Randomization via a built-in and adjustable algorithm
4. Questionnaires: creation, individual delivery and collection
5. Remote distribution of intervention, e.g. through mobile apps, web-URLs or user-logins
6. Follow-up of participants during and after the trial, i.e. reminders, user-support
7. Data gathering, automatically and continuously, including health parameters and usage logs from intervention tools, e.g. sensor data, app data or website input
8. Data analysis, i.e. regular and continual analysis, and data-grouping for later advanced analysis
9. Study closure, patient follow-up, reporting and delivery of expected results and insights

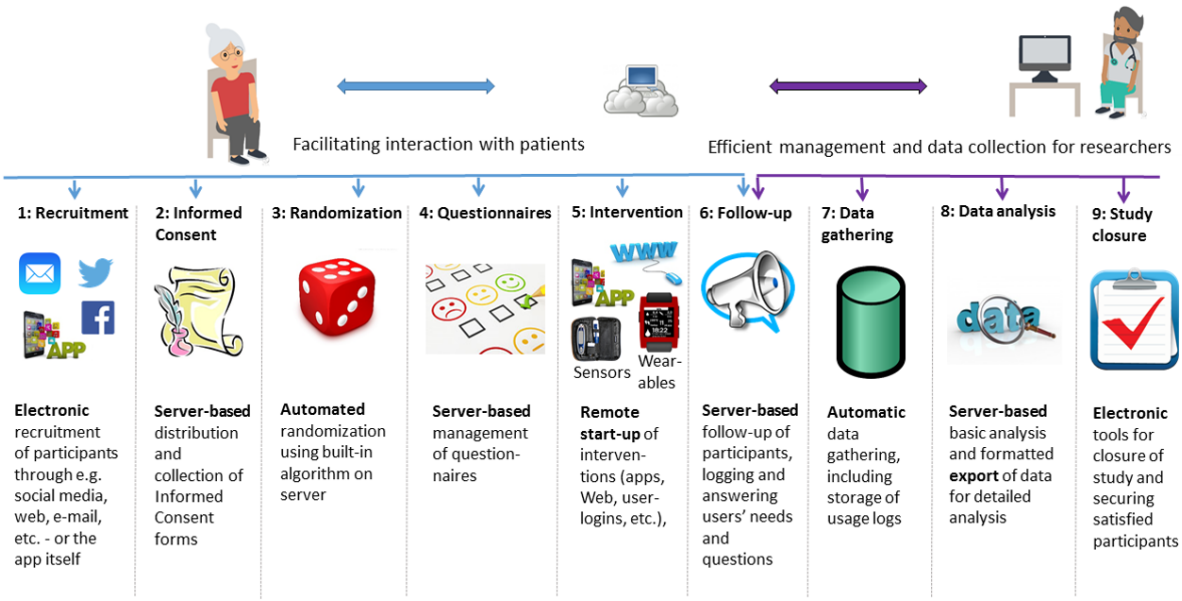



*Figure 1. Hubro - electronic study-management system [3].*

Hubro addresses identified issues in the following way. It reduces the programming barriers by isolating some of the system functionality from its tight integration into the smartphone, resulting in the usability of the Hubro system, i.e. by using an only partial set of functions when performing the study. The integrated survey system is eliminating the need for by providing a generic web-based surveys/consent forms, usable on PCs, smartphone, tablets and other devices that enable interactive input from study participants. Privacy and data security measures include

decoupling identity database from the depersonalized data, where the only link is the password-protected translation function.

While the Android version of Diabetes Diary is used as a research-vehicle, i.e. as a platform, that integrates functionality that is currently being evaluated, the iOS-version provides only the baseline functions. Therefore, the Hubro system has been integrated with an Android version of the Diabetes Diary application in the following way:

- In-app messaging – a user can view rich-formatted messages sent from the Hubro system directly in the app; the user is notified for newly received messages by push notifications
- Automatic and manual upload of the database – a user can manually trigger the upload of the database of self-registered data from the Diabetes Diary to the Hubro server. When a user enrolls to an active study, the automatic upload of self-logged data at regular intervals can be enabled on behalf of a researcher
- Usage logging – all actions performed in the Diabetes Diary application are logged in a real-time and accessible for analysis by a researcher

Hubro was developed on an iterative basis, by continuously deploying and testing loosely coupled services (usage logging, self-registered data upload and messaging) with active users of Diabetes Diary. This strategy facilitated user's adoption of the user interface (UI) and functionality changes, e.g. by using in-app messages to announce future workshops, or by allowing users to turn on anonymous usage data collection.

# Results

The Results section contains outcomes resulting from the design part of the conducted research and outcomes that have been obtained during evaluation and testing phases.

The aim of a design phase was to:

- identify health data-exchange capabilities of wearable sensors
- demonstrate the application of wearable computing in personal health care on a specific use case of a novel combined smartwatch-smartphone application for diabetes self-management
- design and implement a clinical trial support tool, with support for data-collection from wearable devices and use of artificial intelligence for data processing

Within the evaluation and testing phase, we aimed to:

- evaluate the usability of the combined smartwatch-smartphone application on a small sample of T1 diabetes patients using questionnaires
- perform two studies using the Hubro tool, and evaluate the usability of the system using questionnaires that have been responded by the study manager

## Identifying Health Data Exchange Capabilities of Wearable Devices (Publication 1., Publication 4.)

We have performed an analysis of wrist-worn fitness wearables to assess their usage in research projects [4], and based on findings, we have described criteria to consider when selecting a wearable device brand, or a specific wearable device for a research project (Figure 2.). These criteria were identified based on an analysis of 423 unique devices from 132 different brands. As a source of data, we used six databases: The Queen's University's Wearable Device Inventory [23], The Vandrico Wearables database [24], GsmArena [25], Wearables.com [26], SpecBucket [27], and PrisGuide [28].

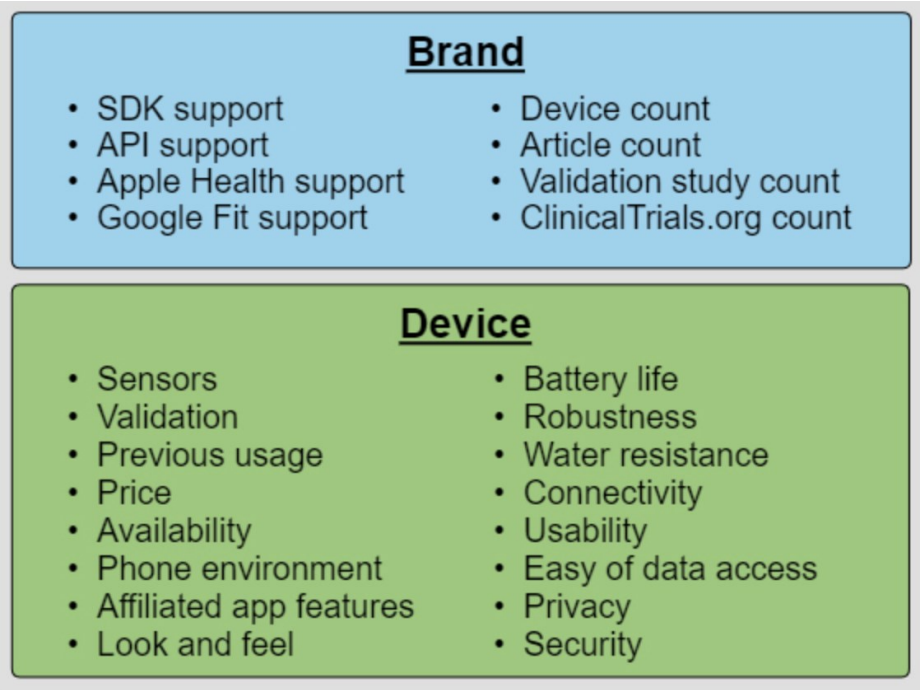



*Figure 2. Criteria to consider when choosing wearable device brand or device. [4].*

Based on our findings, the main criteria when selecting target wearable device brand for a research project can be generally divided into two categories: interoperability and use of the brand in any previous research. As for the device itself, the main choice criteria are software capabilities (e.g. app features, usability), hardware attributes (such as look and feel or battery life) and other characteristics related to specific project needs and economic limitations such as device's price, its previous usage in other projects and validation (i.e. device certification). Generally, the selection process of a suitable device for a research study is simplified due to a considerable high wearable device count.

To describe current health data-exchange capabilities of wearable devices in detail, we have carried out an additional review of 362 wearable devices [1]. As a primary source of data, we have used Vandrico database [24], which is a structured database of wearable devices. From the total count of reviewed devices, 310 were included from the Vandrico database, and the rest of the devices were identified based on information from other sources (i.e. Google search, manufacturer's website linked back from Vandrico).

Identified devices were described using 13 main attributes that are listed in Table 1. We have further tracked six descriptive attributes, that identify device's manufacturer, description, source URL, keywords, system variety and source of information.

| Attribute | Attribute description |
|---|---|
| **Type of wearable system** | Device classification |
| **Communication interfaces** | Set of integrated communication interfaces for transmitting data |
| **Data protocol** | Indicates whether the device uses proprietary or standardized/open data protocols |
| **Smartphone/PC integration** | Types of systems the devices can be connected to |
| **Direct integration with health platforms** | Indicates whether the device supports direct import of data to Google Fit or Apple Health |
| **3rd party integration with health platforms** | The device can be connected to one or more health platforms via a third-party provider |
| **Connection to Health Care System/Middleware** | The device supports import of data to a health care system/middleware |
| **Health data types** | Enumeration of types of physiological data extracted from integrated sensors |
| **Integrated sensors** | Enumeration of sensors integrated within the device |
| **Medical device** | Indicates a certification or approval by corresponding agencies/authorities |
| **User data access** | Collected data are accessible either by directly inquiring the device or via a cloud solution |
| **Developer access** | Indicates support for the development of custom applications running on the device |
| **Device availability** | Indicates the production status of the device |

*Table 1. Collected attributes for wearables [1].*

Based on the collected data, we have concluded the following:

- only a few producers are using standardized transmission protocols/APIs to enable third parties to transfer a data from a wearable device
- Bluetooth compatibility issues may hinder the integration of wearable devices with a full spectrum of smartphones

- Google Fit integrates directly with slightly more devices compared to Apple Health. However, 87% of the reviewed devices do not connect with any health platform
- connection to Health Care Systems/Middleware is sparsely supported
- majority of identified health data types were formed by accelerometer-derived data (physical activity, sleep data)
- only a few wearables covered in the review integrate advanced sensors of physiological health parameters such as ECG, EEG or blood pressure
- the wearable devices consumer market comprises of a small number of certified devices
- there is a need to use a standardized data exchange format to improve interoperability
- only a few devices provide developer tools that allow creating applications running directly on the device

## Combined Smartwatch/Smartphone Application for Diabetes Self-Management (Publication 2.)

The smartwatch is a relatively new player in the area of wearable technology. By the very first steps in the area of smartwatch development can be considered products made by Seiko & Timex in the 1980s, which were programmable watches able to communicate with a computer. By another attempt in this area can be considered Linux-running watch, which was introduced by IBM in cooperation Citizen Watch co in the year 2000. However, the most significant expansion of the smartwatch happened on the CES exhibition in 2014, where a lot of major consumer electronics manufacturers introduced their smartwatch product and showed their willingness to adopt this technology.

Definition of today's smartwatch is vague. In general terms, by smartwatch is considered a programmable, display-equipped, wrist-worn device with a wireless interface for communication with a smartphone. These days, Bluetooth and Wi-Fi are the most frequently utilized technologies for this purpose. Smartwatch connected with a smartphone provides a convenient and time-effective way how to process smartphone-provided information, which requires a short time of user interaction – typically notifications/reminders and messages. With the growing complexity of the smartphones operating systems, the smartwatches simplicity, its

easy reachability on a wrist and notification via its vibration can make up strong arguments for different types of users to use such a device in daily life. The smartwatch can also integrate a variety of sensors, which can be interfaced via provided development toolkit.
To the best of our knowledge, very little of research on utilizing smartwatches in diabetes self-management has been conducted so far. Several concepts of wrist-worn devices measuring vital parameters like blood glucose level and blood pressure have been reported on the internet. Only a few devices were however approved by regulating agencies and introduced to the market.

We have proposed a novel concept of a diabetes companion application running on a smartwatch. This application aim is to facilitate the use of digital diabetes diary on the smartphone and add several new features. We mainly focus on the integration of physical activity tracking capabilities, and we investigate how these capabilities affect the sugar level balance of diabetes patients. Our solution delegated commonly used functions of a smartphone-based diabetes diary to the smartwatch and added on-smartphone processing of accelerometer data measuring physical activity.

As a candidate for the implementation, we selected Pebble smartwatch [29]. Pebble supports a desirable set of functionalities that we have identified during the review of wearable devices. Mainly it is the possibility to develop own applications running on the device, access to the raw sensor data and compatibility with both leading smartphone operating systems – Android and iOS. Testers in the evaluation phase have appreciated the device's features like a simple user interface, long battery life and well-readable monochromatic display.

The basic set of our smartwatch application's functions included insertion of new carbohydrates/insulin/glucose recordings with a decimal precision (Figure 3. – A-C), presentation of prior registrations (Figure 3. – D), recording of physical activities (Figure 3. – E), reminders for glucose measurement and steps counting (Figure 3. – F). Pre-processed physical activity data was continuously transferred to the smartphone, where it was further transformed into physical activity segments and presented with the rest of the data to the user.

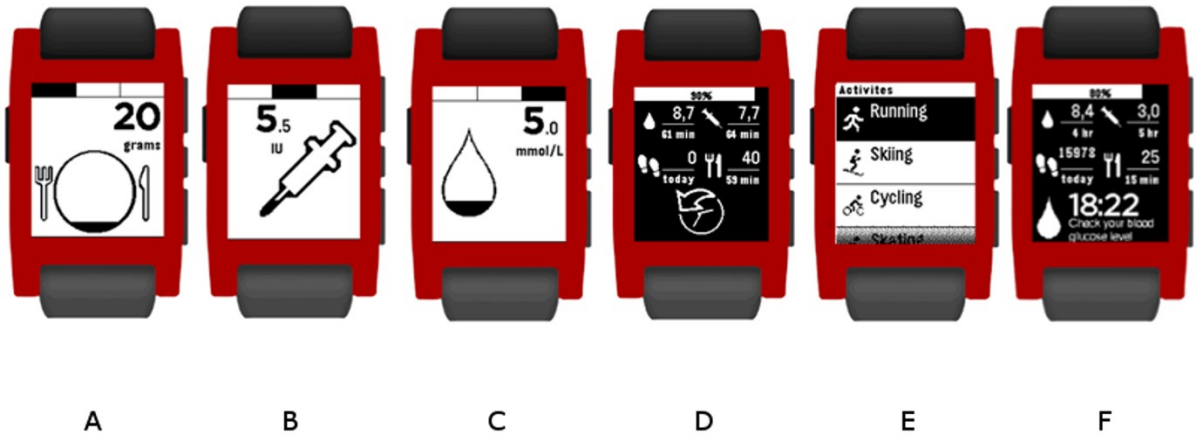



*Figure 3. Diabetes Diary application on a Pebble smartwatch.*

The smartwatch application was configurable via smartphone to enable:

- customization of reminders timeouts
- sensitivity adjustment for the integrated step counting algorithm
- customization of screen order, when making a new registration

Several extra proof-of-concept functions have been integrated and tested alongside the version that has released for the public audience. These functions include:

- speech-based insertion of new registrations
- registration of insulin type
- similar situation matching
- displaying a near real-time CGM value

## Evaluation of Combined Smartwatch/Smartphone Application for Diabetes Self-Management

The testing population consisted of 4 males and 2 females with type 1 diabetes. Test participants received a pre-test questionnaire, where we asked for their expectations regarding provided functionalities. Most of the identified desirable functions, such as reminders, physical activity tracking and two-way communication with the smartphone diary were already implemented during the 9-month development period and were ready to be tested.

We have used the post-test questionnaire to cover topics related to the daily usage of the application. Users have reported positive usability of the system; however, some of them have troubles understanding the advanced functions such as looking at the second latest values through the smartwatch application.

Following features received positive user feedback:

**Stand-alone use of the application**

To a certain extent, it is possible to use the Diabetes Diary for the Pebble smartwatch exclusively without a smartphone. This use might be an option for older people, who may find the user interface of the smartphone too complicated. In this case, the functionality of the smartwatch application is limited to show the latest and second latest registrations, track physical activity and provide alarms for an upcoming blood glucose measurement.

**Possibility to use the application offline without having it connected to a phone**

It is possible to use the smartwatch application for a certain amount of time offline, without having it connected to a phone. This feature may be useful when the smartphone runs out of battery. All registrations and physical activity data recorded in this offline period are consequently transferred to the phone after the smartwatch reconnection.
The smartwatch app will remember approximately 200 days of data.

**Speed of making a new registration**

We have compared the speed of making new registrations using the smartwatch and using the smartphone. The test has shown the average time required to make a new registration using a smartwatch was 13 seconds, on the other hand making a new registration using a smartphone took 36 seconds, which is considerably longer than in a case of the smartwatch.

The smartwatch application has been released for a public audience and gained positive feedback from its users. However, since the production of the Pebble smartwatch has been discontinued, the platform infrastructure has been unmaintained since then. The experience, however, showed the potential and improved usability of combined smartwatch/smartphone self-management application.

## Design and Implementation of an Electronic Clinical Study Management System (Publication 3.)

### State of the Art - New Evaluation Methods for eHealth and mHealth Services

With the ever-evolving landscape of mobile and wearable technology, it is important to empower clinical trials investigators by delivering them sophisticated tools to increase efficiency of mHealth clinical trials. While the traditional RCTs are well described, newly emerging methods of mHealth evaluation are still being developed and their efficacy is being evaluated. In particular, for mHealth interventions targeting chronic diseases, the hypothesis that better monitoring with mobile technology will lead to better management, better outcomes and reduced disease burden has yet to be adequately tested [30].

Apple ResearchKit [31] is a Software Development Kit (SDK) that allows researchers to perform medical interventions through mobile applications. The ResearchKit integrates the onboarding process by including eligibility determination, presentation of informed consent and endorsement of access to user data. One of its disadvantages is, that utilization of ResearchKit, as an iOS-compatible software implicitly selects certain economic strata, and thus it is introducing coverage bias. Another drawback is that it reduces the flexibility of creation of new surveys/questionnaires since the integration of ResearchKit requires knowledge of programming, i.e. involvement of iOS developers in addition to the researcher's effort. Also, the framework does not include any backend data storage or data management solution, leaving the handling of data on a researcher.

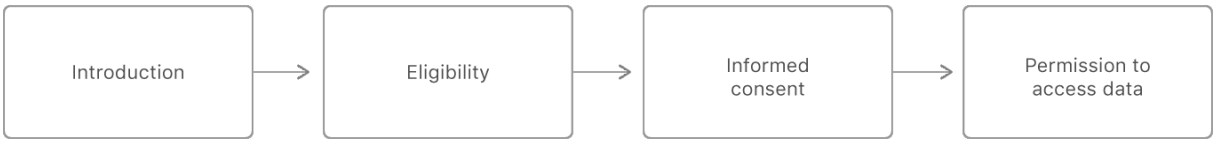



*Figure 4. Onboarding process as defined by a ResearchKit [31].*

ResearchStack [32] is an SDK aiming to help to build research applications for Android. One of its primary goals is to make it easy to port existing applications built using Apple's ResearchKit to Android. ResearchStack allows to facilitate the onboarding flow (i.e. to present and process the informed consent and administer sign up). The SDK also supports on-disk data encryption. While the system is extensible, its design also introduces several limitations, which are analogous to the limitations of ResearchKit.

ResearchDroid [33] is another project aiming to port the functionality of ResearchKit to the Android platform. It primarily addresses the survey forms automation and information building process. Similarly to the ResearchKit, the ResearchDroid has to be integrated into an existing application, and therefore its utilization requires development skills.

One of the options, how to eliminate identified drawbacks (i.e. limitations in terms of missing data collection platform), is to use the multi-mode approach. This approach essentially means that the researcher combines the selected framework with an online data collection solution. The multi-mode approach has the potential to increase the willingness of potential study participants to participate, through a more convenient exchange of their health data. The multi-mode approach also mitigates the downsides of using individual research library by:

- providing an easy-to-use interface, which allows researchers to create and manage user-friendly, responsive questionnaires
- targeting other platforms besides smartphones and therefore enabling researchers to reach out to a broader audience
- utilizing a secure data storage

Alongside ResearchKit, Apple has introduced another framework, CareKit [34]. CareKit is a toolkit for building iOS that help users follow described care plans. CareKit consists of 3 basic

screens, i.e. Care Contents, Insights and Connect. Care Contents presents tasks, like medication, which has to be followed by a patient daily. Insights screen lets a user understand the treatment progress, and graphically presents the data. Lastly, the Connect screen allows the patient to communicate with physicians, nurses, family and vice-versa.

Although there is an undeniable overlap between functionality provided by ResearchKit and CareKit, Apple states that the ResearchKit is aimed for research whereas CareKit does not have this limitation. It is important to note, that ResearchKit, ResearchDroid, CareKit and ResearchStack are libraries or development kits, which integration requires knowledge of programming and essentially, presence of smartphone application skeleton.

We have identified only a few systems that integrate complex workflows corresponding to those, that are offered by using previously described multi-mode approaches.
AppBakery [35] is a commercial research tool and app generator developed by TrialX company. The concept is based on an on-demand generation of iOS and Android application scaffold with an integrated AppBakery development kit. Multiple customization options are provided at the time of app generation. These include tools to collect patient-reported/generated health data, psychosocial behaviour, fitness, nutrition and sleep. The AppBakery is also addressing treatment arm assignment through randomization, distribution of questionnaires, reminders and informed consents. The collected data is retained at the AppBakery's servers for 30 days.

Another identified system with a complex toolset for study management is DHARMA [36]. The follow-up of study participants is possible through text messages. Authors reach General Data Protection Regulation (GDPR) and Health Insurance Portability and Accountability Act (HIPAA) compliance by encrypting the database with patient's data, and they are protecting the data access by using firewalls, secure socket layers connection between system components and enforcing a use of Virtual Private Network (VPN) connection to administer the system. It is not clear whether the system is provided as-a-service, or whether it can be installed on-demand and managed by the organization's IT teams.

The last reviewed project, which takes a similar approach to that of AppBakery, is MoRe [37]. The project is targeting researchers with a zero or limited knowledge of programming. However, while the AppBakery lets the researcher generate a skeleton of a new application per

a study, MoRe works with a single application distributed to all participants regardless of study they join. Also, the questionnaire, consent and eligibility modules are available alongside a toolset for data collection. The project has been released in a Beta stage in 2017; current development is unknown.

Overall, we have identified lack of services/platforms, that would enable the researcher to create and manage studies with support for patient consent easily, and offer randomization, messaging and data processing with a possibility to run on-premise appliance (e.g. research organization's in-house servers).

### Identified Use Cases of Hubro System

To visualize the functional requirements of the Hubro system, we have identified main use cases and created a use case diagram (Figure 5.) that includes study participant and researcher actor. Essentially, the use case diagram is based and further elaborates on Figure 1., which illustrates all steps of study when managed with the Hubro system.

Internal use case dependencies are not included in the use case diagram since currently, the flow of individual actions is not set by a decision support module, but solely depends on a design of the study, that is not captured in any component of the system. For example, a user is not able to check the progression and has to rely on information passed from the study manager. We have anticipated that this fact may introduce a feeling of uncertainty or confusion among some of the users. To eliminate possible users' disorientation and confusion when using the system throughout the study, we have created a user manual, that describes an expected flow throughout a particular study for which the user has signed up. The user manual has been distributed via email to users once they have signed up for the study.

Identified use cases have been further used to determine optimal software architecture. Based on our analysis, we have decided to implement the entire solution using loosely coupled components. Also, due to expected complexity of some components (e.g. survey creation and management, usage tracking), we have decided to use properly tested open-source projects, that integrate well due to existence of an API interface.

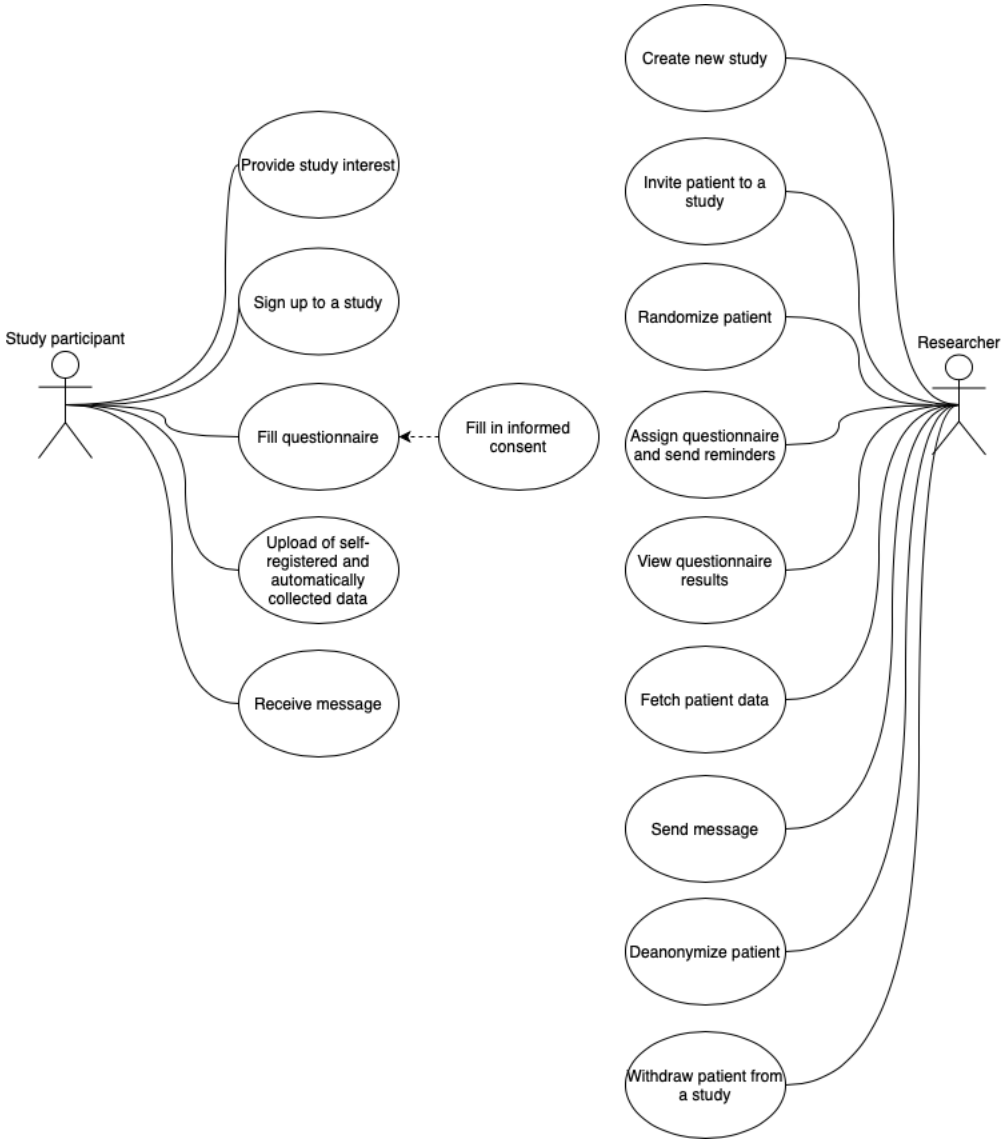



*Figure 5. Use case diagram of the Hubro system.*

### System Provisioning

Complex frameworks with data collection support such as AppBakery are available through the Software as a service (SaaS) distribution model. SaaS's popularity is mainly based on low usage costs, continues updates on behalf of the software vendor and scalability of the platform. The distribution model, however, also suffers from several disadvantages, that mainly involve difficulties with the platform integration in regulation and compliance matters [38]. On-premise software provisioning can potentially mitigate these downsides by offloading whole infrastructure management to the company's IT teams. On-premise requirements, such as in-house hardware, integration capabilities could be synergic with demands for implementation in highly regulated areas, such as eHealth.

The Hubro system consists of multiple loosely coupled applications and server components, making Hubro a candidate for provisioning through the SaaS model. However, considering the nature of collected data, rolling the Hubro out as a service to end-users might be troublesome, due to compliance matters. Also, a naïve, on-premise distribution might be complicated because of many onboarded components resulting in a significant workload on IT system operators. The way for seamless delivery and operation could be a use of self-sealed container-based architecture orchestration through the management system like Kubernetes [39], bringing in the best of both worlds – easily manageable and configurable system running on hardware located within the premises of the institution, or within a cloud computing environment.

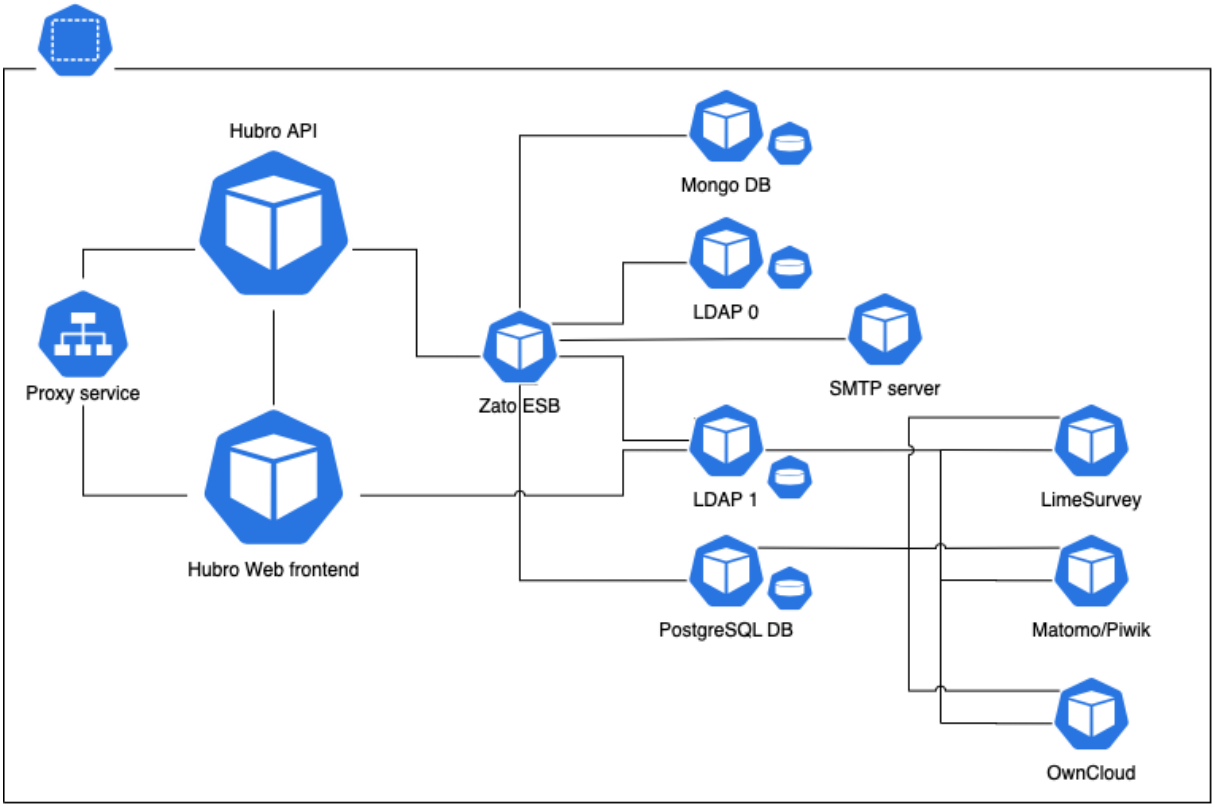



*Figure 6. Kubernetes cluster diagram visualizing individual Docker containers that are sealing off services of Hubro.*

Figure 6. shows the architecture of a Kubernetes cluster with docker containers, that are sealing off various services of Hubro. Special considerations had to be given to the persistent storage, as containers are immutable, meaning the data is lost, once the container reaches the end of its lifetime. Containers that utilize persistent storage are Mongo DB, LDAP servers (LDAP0 and LDAP1) and PostgreSQL DB [40].

## Participant Recruitment, Enrollment and Staging Throughout the Study

Compared with traditional recruitment strategies such as print media, social media advertising is favourable in terms of its reach (especially with hard-to-reach populations), cost-effectiveness, and usability. Therefore, recruitment for research through social media is increasing and likely to continue to grow [41]. The specific targets within a social media can be interest-groups, such as DIY groups, which are gathering people sharing the same motivation-incentive.

An initial step when performing a study is the determination of interest for participation. The generic web-based interest form is providing compressed information about the enrollment process, and it is provided for each project in the Hubro evidence. The link to interest form can be distributed via various channels. Social networks, a media with a significant outreach, can be utilized to share the link among various targeted patient groups. The interest link can also be sent out via email to existing distribution lists. Another option implemented in the Hubro system is to express an interest on behalf of a particular person. In practice, the study manager can insert one or more email addresses through the Hubro user interface, which would make these persons implicitly interested in the project.

Having the required initial amount of people who are interested in participating enrolled in the system, the study manager can start distributing informed consent forms. Traditionally, informed consent has been signed and collected in personal contact. The consent form has to contain information about the research organization, the purpose of the study, what data are being collected and how they are going to be used. The consent form should also contain information about how to leave the study.

Hubro is presenting the digital consent form as a regular questionnaire with a single checkbox, used to express the consent, and a button for the form submission. Once the form is submitted, the system lets the user download the finalized form in a pdf format. Distribution of consent forms is done via email, with an option to send additional reminders. Indication in the system clearly shows whether, or when the consent form was submitted (Figure 9).
Once the person interested in participating in the project gives his informed consent, the study manager can perform the randomization for that specific participant. Treatment group

assignment immediately propagates to the Hubro web user interface, and the study manager can proceed with further actions such as questionnaire distribution.


*Figure 7. A form allows the project manager to invite users to the study, that includes a shareable URL pointing to an interest form for a self-sign-up.*

### User Interface Design

One of the primary goals was to avoid an exhausting time dedication to study management as the number of participants increases. Therefore, we designed the user interface in a way, that the researcher can discover actions which may require his attention with a glance. We introduced an informational column for each participant, that summarizes the following information:

- date and time of the last check for new messages via REST API including client info
- indication, whether there are any unfilled questionnaires
- date and time when the user was enrolled in the study
- date and time when the data was uploaded to the Hubro server

Having this information available, the study manager can quickly check, for example, whether the participant has upgraded to the newest available version of the software, which is part of the study. Other observations may include, whether the participant uses mainly the 4G connection or Wi-Fi, and thus, whether it provides all usage logging data. Combined with a date-time of the last data upload, this information could be used to trigger a force upload of the data to the server.

To simplify the identification of participants, that require attention, we introduced a flag symbol, which is shown for those who have not filled in all distributed questionnaires yet. The study manager can, however, extend the set of rules which trigger flags' appearance as required per treatment group or project. This way, the study manager can to include more participant-related modalities affecting their status.

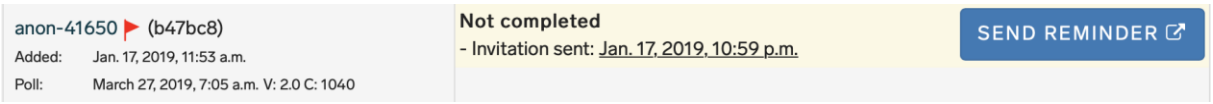



*Figure 8. A participant with an unfilled questionnaire indicated by a flag icon.*

## Randomization

When planning a randomized clinical trial, careful consideration must be given to how participants are selected for various arms/treatment groups of a study.

Selection bias may occur when participants are not assigned to treatment groups with equal probability. A technique which is commonly used to prevent a selection bias is called allocation concealment. Traditionally, this has been implemented by randomly shuffling and then sending participants sealed envelopes with treatment group identifiers. Block randomization is a commonly used allocation concealment technique in clinical trial design to reduce bias and to achieve balance in the allocation of participants to treatment groups, especially when the sample size is small [42]. We have implemented block randomization as a default randomization method in the Hubro system.

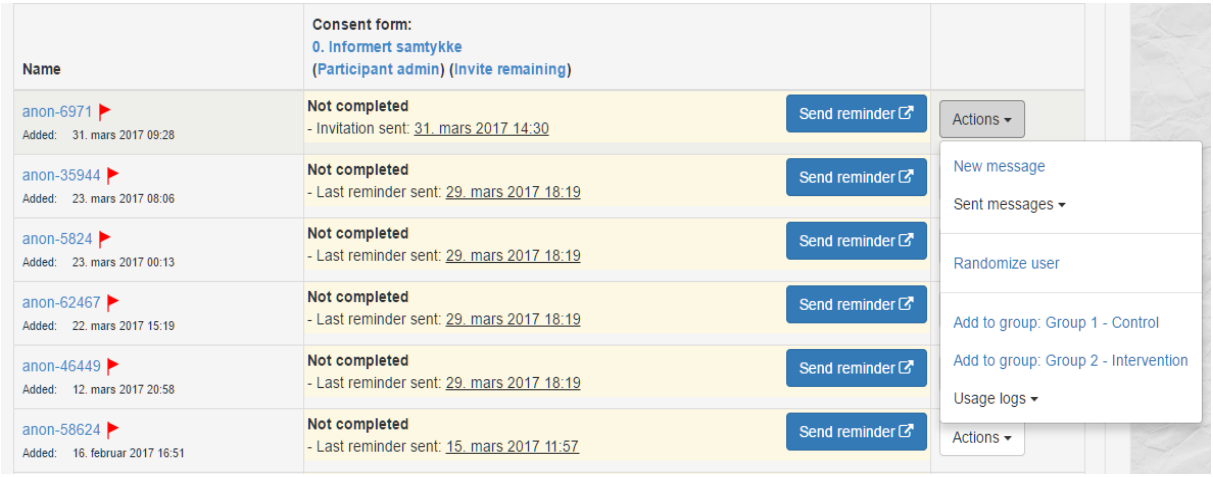



*Figure 9. Context menu with an option to randomize participant.*

Block randomization works with a single, pre-generated block at the time, from where the values identifying a particular treatment group are sequentially pulled and are used to determine the assignment in the randomization process. To achieve uniform distribution of treatment groups with a single block, the researcher must select a block size that is a multiple of the number of treatment groups. Mainly, the number of treatment groups has to stay constant once the randomization process starts. Block size, however, can change with every generated block – in this case, we speak about random block size. Also, small and large block sizes have different properties – small block sizes assure that sizes of treatment groups are very similar at all times of study progression, although the blinding could be reduced by increased predictability of some assignments to treatments groups.

The architecture design of the Hubro system allows to integrate multiple randomization algorithms and use them variously per studies. Alternative methods may include simple or stratified randomization.

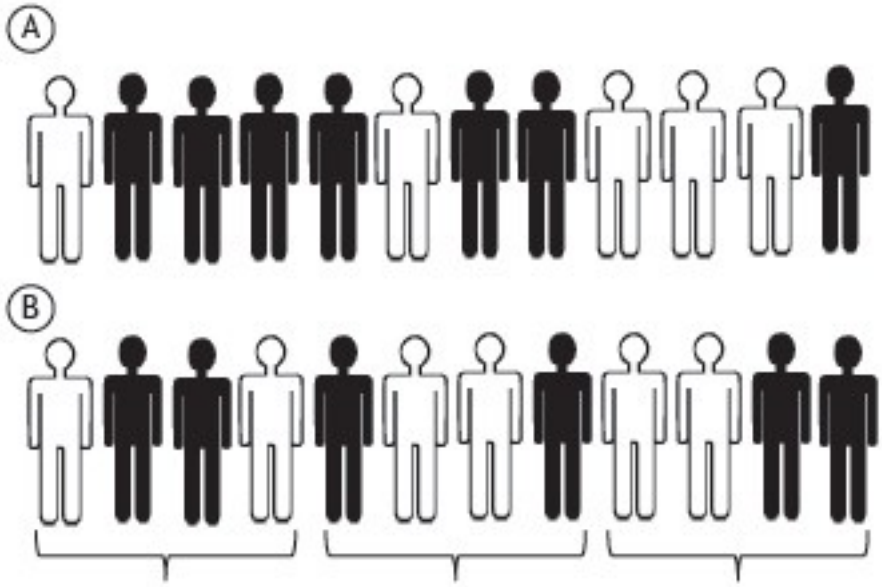



*Figure 10. Example A) Simple randomization process, that resulted in 7 patients being assigned to the intervention treatment group, and 5 into the control treatment group. B) Block randomization using a block size of 4, that resulted in an equal number of participants assigned to each treatment group [43].*

### Communication with Study Participants

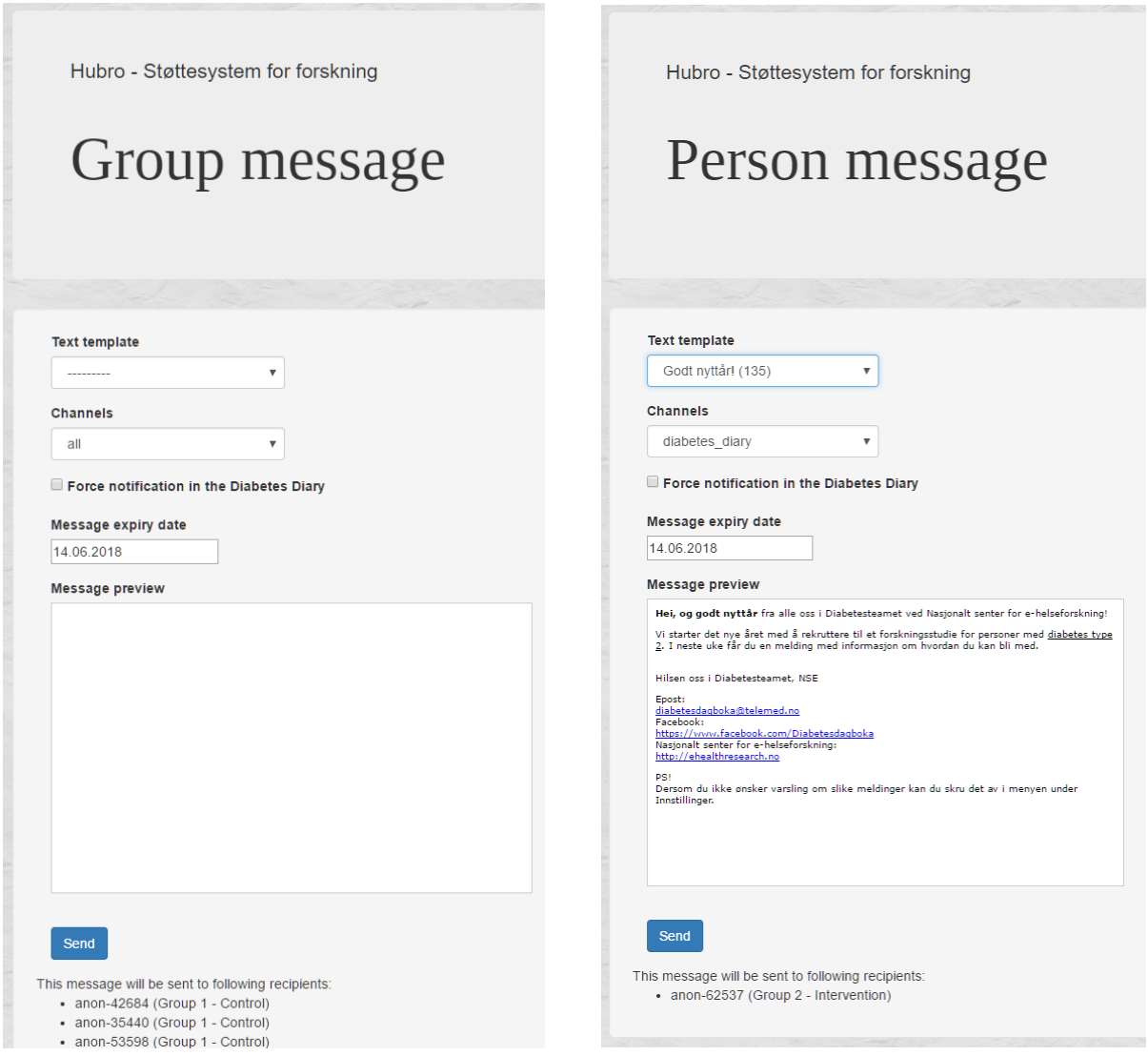
Communication with study participants has to be in place during the study period for various reasons. Obvious use cases include support during the study period, sending out reminders, but also handling of situations when the study manager needs to reach for a piece of specific information.

Messages in the Hubro system can be distributed through various channels. So far, the Hubro system supports distribution of information via an ordinary email, or/and through the integrated REST API, from where the messages can be pulled by individual applications connected to the platform. For the messages delivered via REST API, we have not implemented an option for a direct response, and therefore the messages are only one-way function. Lack of option for direct response is however supplemented by a possibility to include formatted text enriched by HTML tags. This way, the study manager, can include images or website links containing additional information or interactive elements such as forms.

Most of the communication occurs via a template-based message system. This approach allows the study administrator to target many participants using messages with personalized content. Placeholders can represent a specific project name, user ID and other information tied to a specific combination of project and participant.

Messages can target individuals, members of a specific treatment group, whole project, or they can be posted for a general audience. The latter option makes messaging a powerful tool for in-app recruitment since messages reach the target audience with the potential of receiving a positive response to a call for research study enrollment.

Information contained in various messages can be of value only for a certain time and may be misleading when presented when its content is not relevant anymore. Therefore, we defined a retention period for which is the message valid. This parameter emerged from the need to invalidate particular messages after a predefined time. Such messages could be updates about other relevant studies or opportunities, which become irrelevant with time. The retention period is included in the message descriptor, which is described later in this document.

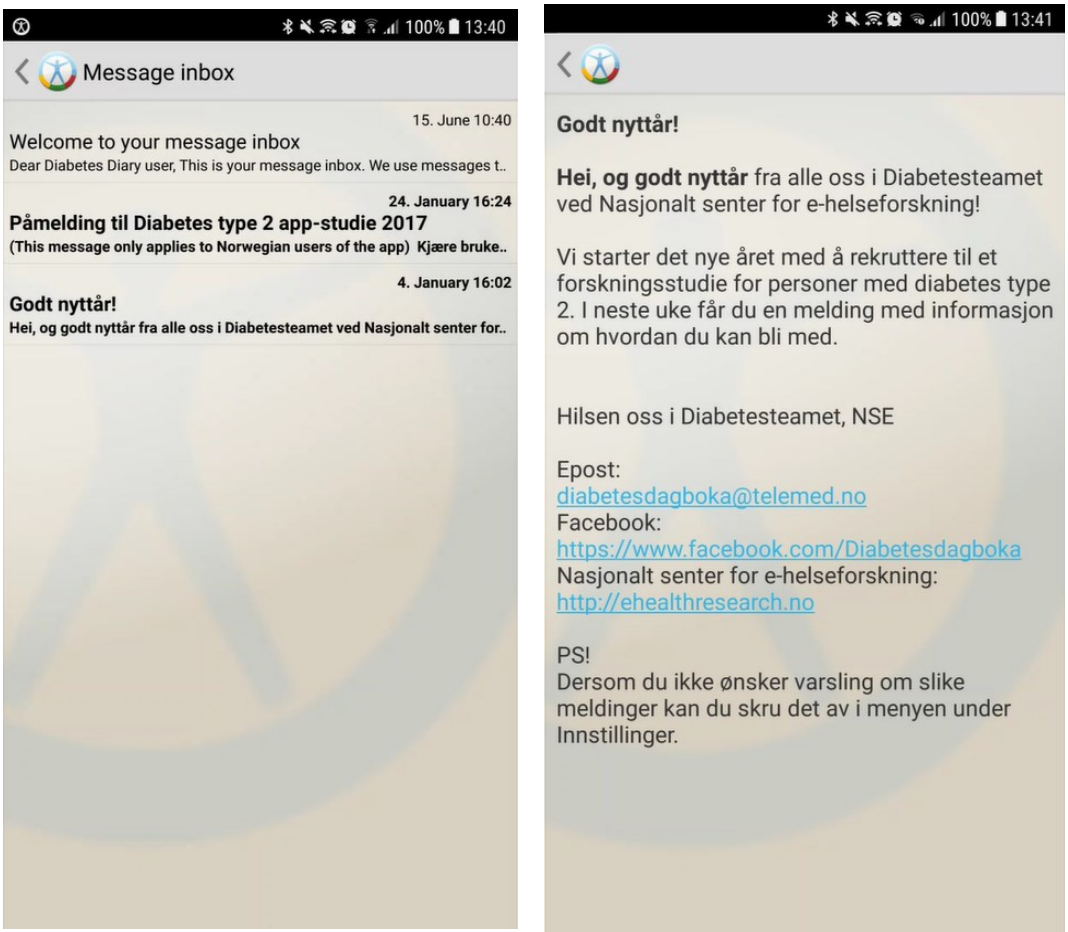



A B

*Figure 11. A messaging interface for sending a group message (A) and a person message (B). The interface allows a selection of message distribution channels, setting a message's expiry date and enforcing a notification, when the message is sent to a Diabetes Diary application. When sending a group message (A), the system enumerates all recipients of the message.*

Figure 12. shows a specific implementation of a messaging client. The client is implemented in the form of a message inbox within the Diabetes Diary smartphone application. The application checks for new messages in regular intervals (every 1 minute) and if there are any, the client interprets their preview in a notification bar, which is then presented to the user.

The messaging system presented here is essentially a so-called "push notifications". The API and tools are usually an out-of-the-box feature of the OS. In our implementation, we provide a simple, fully integrated way of reaching out to users with granular targeting options. The outreach is not only limited to mobile devices/applications but also covers other devices with internet connectivity, which is an important aspect given the upraise of Internet of Things (IoT). We are also keeping the data within the Hubro data cluster, without sharing it with any of the third parties.




A B

*Figure 12. A message inbox (A) and message content view (B) presented in the Diabetes Diary application.*

## Technical Description of Hubro

From a high-level perspective, the Hubro consists of multiple servers, each serving a specific purpose. The responsibility has been divided in the following way:

- Questionnaire platform
- Usage logs collection platform
- Identity directories and authorization server
- Enterprise service bus
    - Identity translation service
    - Randomization service
    - Messaging service
    - Data upload service
- Web frontend for project administration
- File archive area
- REST API
- SMTP server

## Personal Health Information (PHI) Management

As discussed, the trend is shifting towards collecting, storing and exchanging a significant amount of health data originating from various sources. The situation caused a significant awareness among patients regarding the secondary use of health data and its protection.

One of the ways how to protect the PHI is to anonymize the data. What the anonymization essentially means is stripping of the data from any personal information, that can identify the data with the patient. Anonymization is an irreversible process, which allows the transfer of data between organizations, eliminating the risk of unintended disclosure.

Data depersonalization, also known as pseudonymization, on the other side, is the instrument for replacement of personally identifiable information by placeholders. One of the options is to comply with GDPR by keeping the data in a state identifiable with an individual.

Since the system consists of various loosely-coupled components, the PHI in Hubro is scattered across multiple databases owned by the following services:

- Questionnaire service (LimeSurvey)
- Data collection service (Custom implemented)
- Usage log service (Matomo/Piwik)
- LDAP1/LDAP0 directory services (OpenLDAP)

One of the approaches to implementing data de-personalization instead of its' permanent anonymization is through the link table. Evidence of the table could be either on a paper or electronic. In this case, we talk about Identification database and Research database. The Identification database may be owned by a researcher or a 3rd party company.

Since multiple independent services own the PHI in Hubro, it would be problematic to implement a link table operating over all of these services. The approach we chose instead, was to create an alternate identity linked with the real identity, specific study and treatment group. Therefore, implementation of a linkage table in Hubro is based on two directory services (LDAP servers) and a translation service between them.

The identity associated with the stored data is the anonymous one, created by the translation service and managed the LDAP0 directory. Sometimes, the use of full email address is not possible, either for technical reasons or in cases when its use would degrade the patient's or manager's user experience with the system (e.g. when requesting its manual input). Then it is the first five characters that are used to identify records in these databases. Real identities, where email addresses are used as a user principal name, are stored in the LDAP1 directory.

While PHI anonymization would undoubtedly provide the most secure way to implement privacy measures, there are several situations, where its permanent effect could lead to an inability to enforce compliant data retention throughout the study period. Permanent data anonymization could cause issues in the following scenarios:

- The participant is allowed to withdraw from the study. Based on the study protocol, the withdrawal may involve removing the PHI from the database, and therefore it is essential to keep the PHI linked to the identity.
- The PHI record has to be checked on behalf of participant request, e.g. to check current progress through the study.

### Identity Management and Authentication Backend

Directory services present a way to organize network resources or items and manage their attributes. The directory services are optimized for reads, implicitly implementing multiple indexes supporting the read operations in favour of writes, as opposed to relational databases. Therefore, they are extensively used for authentication, authorization and as a provisioner of yellow/white pages. Due to multiple competing implementations of directory services, the X.500 standard has been developed to unify the way of operating with directory services from different vendors. Since the X.500 protocol has been too complicated, the Lightweight Directory Access Protocol (LDAP) has been introduced with a reduced number of functions that are available in the full X.500. The Hubro system uses two LDAP servers, denoted as LDAP0 and LDAP1, are used in Hubro to keep information about participants and project information.

The identity management of Hubro distinguishes between two roles – study manager and participant. No other granular user role differentiation is implemented, and therefore all users with a role of study manager can access all projects on a single running instance of Hubro. This fact has to be considered when using Hubro for management of multiple studies. These roles are tied to LDAP1 identities and are attributed through the LDAP1 group membership. The authentication credentials used for logging into various components refer to LDAP authentication attributes.

The specific LDAP implementation used in Hubro is provided by OpenLDAP [44]. Each LDAP directory has a default schema, which can be extended to fulfil needs of the system in which the LDAP is integrated. Out-of-the-box installation of OpenLDAP supports Hubro-integrated components without a heavy schema modification.

The LDAP0 identity tree structure is designed in a way, that the root level comprises of multiple project organizational units. Each project organizational unit has a set of groups that represent actual intervention groups of participants in a study. Participants remain in a project organizational unit, in a case, that the participant is not assigned the group yet.

A counterpart of LDAP0, the LDAP1, is a flat database of participant identities using an email as an exclusive, primary identifier. Essentially, every identity in LDAP1 can be linked to one or more identities in LDAP0, since every person can participate in multiple projects.
A translation service provides the translation between identities in an anonymized directory (LDAP0) and identities located in an identity directory (LDAP1).

## Enterprise Service Bus (Zato)

An Enterprise Service Bus (ESB) [45] is a middleware tool used to distribute work among connected components of an application. ESBs are designed to provide a uniform means of moving work. Different services are subscribing and consume messages that are published on a central bus and do the work based on an incoming message flow.

For the use in Hubro, we have utilized a Python implementation of ESB provided by Zato [46]. Individual services are implemented in a Python and can integrate the functionality of additional Python libraries. Besides standard features, Zato also implements several additional features, which are useful for the management of the system component in a production environment. One of such functions is 'hot-reload', which makes possible to update the definition of various services on-the-fly, without a need to restart the whole system. Use of the ‘hot-reload’ has significantly improved the prototyping of individual services.

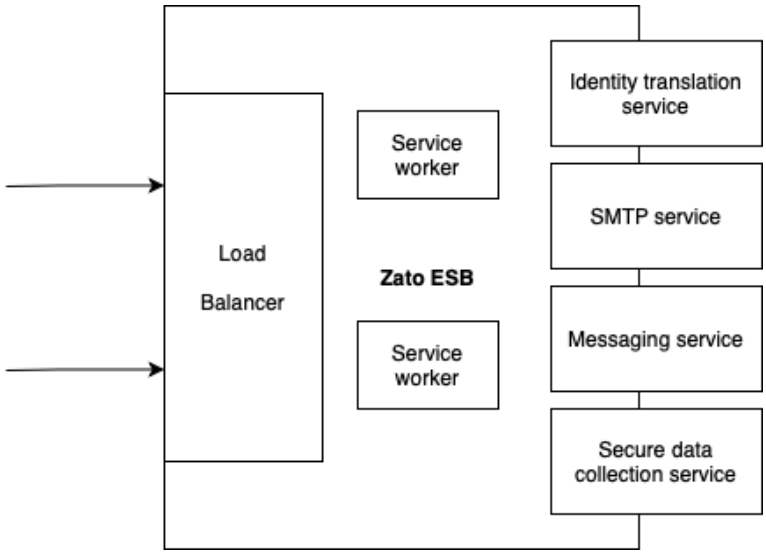



*Figure 13. Overview of Zato services utilized within Hubro system.*

## Translation service

Translation service is responsible for translation between realms in LDAP1 and their corresponding anonymous identities located in LDAP0 through the hashing algorithm. Input variables of the hashing algorithm are LDAP1 entry common name identifier (participant's email), project name and a secret (Figure 14.). The common name identifiers of the LDAP0

entries are structured in a form hash@domain.tld, that is configurable based on an organization's domain settings.

Achieving backwards linking of the entries in LDAP0 to entries in LDAP1 (deanonymization) requires a sequential hashing of LDAP1 entries and their lookup in LDAP0. The use of the de-anonymization process has to be confirmed by inserting a password, even if the study administrator is currently logged in the Hubro system.

Translation service is called from multiple places of Hubro study workflow. These places include an SMTP server at the point of receiver's address translation when sending email to an LDAP0 identity, assignment of an LDAP0 identity when enrolling participant for a new project and deanonymization of LDAP0 identities on study administrator's request.

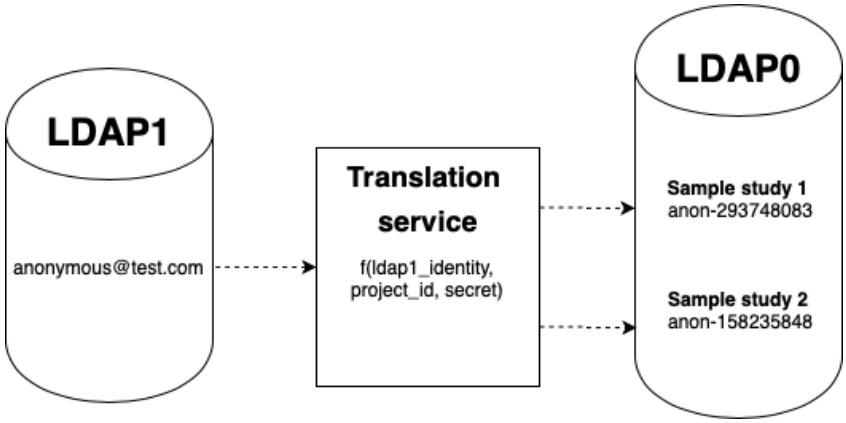



*Figure 14. Translation service function interacting with an LDAP1 database of participant identities to create a linked anonymous identity in LDAP0 directory.*

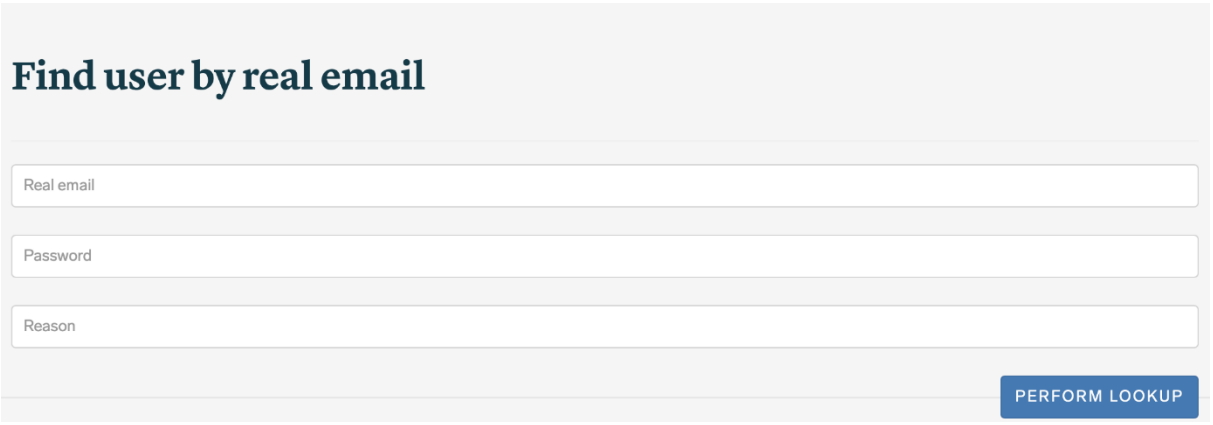



*Figure 15. A lookup form for searching user's anonymous identity based on a real email address.*

### SMTP service

Hubro bundles its own SMTP server, which is located at the forefront of the organization's primary SMTP server. It proxies all outgoing email communication from Hubro and translates the recipient's LDAP0-originating email addresses via translation service to real email addresses resolvable by a public DNS service. If the email addresses are not subject to translation, they pass through to the organization's primary SMTP. No information about translation processing status that would make any project-specific information identifiable with a real email address of the participant is stored in the server's log files.

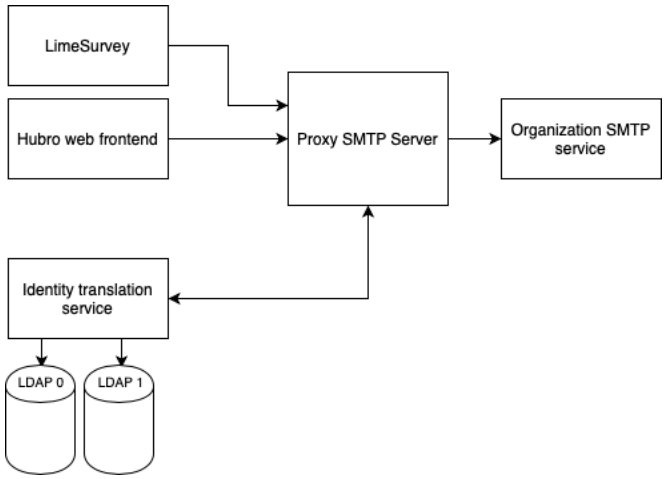



*Figure 16. Email translation and sending process as initiated from LimeSurvey and Hubro web frontend.*

### Messaging Service

As messages exposed via REST API are in principal JavaScript Object Notation (JSON) objects, we have selected a key-value database as a place for its storage. For our implementation, we have used a MongoDB [47], which represents a well-established NoSQL database.

While messages delivered via email have its informational value in a text, links and images, messages delivered via REST API can carry additional information in terms of actions related

to the communication with the Hubro system. For this reason, we have introduced the system parameter. System parameter contains information, which is not interpreted by the user, but is instead meant to be processed by a specific implementation of the client and used to perform specific actions, that cannot be planned or triggered automatically. System parameters we support in the Diabetes Diary smartphone application include a forced upload of data to the server and status control of functions when a user is participating in a study, i.e. a study manager needs to enable/disable usage tracking and automatic uploads of data to the server.

### Secure Data Collection Service

The smartphone application market is saturated with many health applications. A common observation is that many of these applications utilize cloud-based storage to synchronize the data between multiple devices. While this feature introduces a certain level of convenience, it may also raise privacy concerns among the application users.

Secure data collection service is a tool for secure, on-demand transfer of data from client to researcher. The receiving endpoint does not define any mandatory data scheme and thus leaves all data post-processing for a researcher. On the other hand, the service remains generic enough to cover most use cases of transferring various types of data from a wide range of devices (e.g. smartphones, smartwatches, wearables), when the research application is acting as a data collection hub. The data collection service can also be used as an independent component by users outside of the research study when needed, i.e. enrollment in the study is not a prerequisite for sending the data into the system.

The motivation for implementing additional data encryption layer securing the data transmission in addition to the encryption provided by traditional TLS/SSL layer was based on multiple considerations. This way, implementation of the Hubro application layer receives already encrypted data, that might be favourable when introducing different privilege levels for integrated modules. Also, utilizing this approach, the upload process is more resilient to TLS/SSL vulnerabilities, such as Man-in-the-Middle attacks. Besides, the amended design makes it possible to securely share the encrypted data via alternative transmission channels such as email, if needed.

The secure data upload process consists of multiple stages. First of all, the client checks whether it has been assigned the unique ID based on any previous actions. If not, it first queries the service for a unique ID and gets one assigned. Unique IDs are randomly generated strings of 5 characters, which are used to identify the client. The length is shortened intentionally to make its communication between the study manager and a user easier, which would happen in situations when the data need to be shared ad-hoc. After the unique ID is secured, the process continues with setting up the shared symmetric key.

The initial key-exchange resulting in a shared symmetric key is based on the Diffie-Hellman algorithm [48]. Whitfield Diffie and Martin Hellman first published the scheme in 1976 as a method of securely exchanging cryptographic keys over a public channel. The key exchange results in a 256-bit shared key, which is exclusive for each client.

Diffie-Hellman key exchange method is widely supported by many cryptographic libraries available for various programming languages, and it is easily implementable on an ad-hoc basis on the client-side if needed. Similarly to the unique id, the symmetric key is established once, at the very first use of the secure data upload service on behalf of the client. All ongoing secure data upload services then use this key.

With an established symmetric encryption key, the Hubro server is ready to receive the encrypted data. The encryption is performed using the symmetric cypher AES operating with the pre-established 256-bit encryption key. In the client implemented in the Diabetes Diary, the application pushes the database of self-registered data. While the data itself does not reveal a

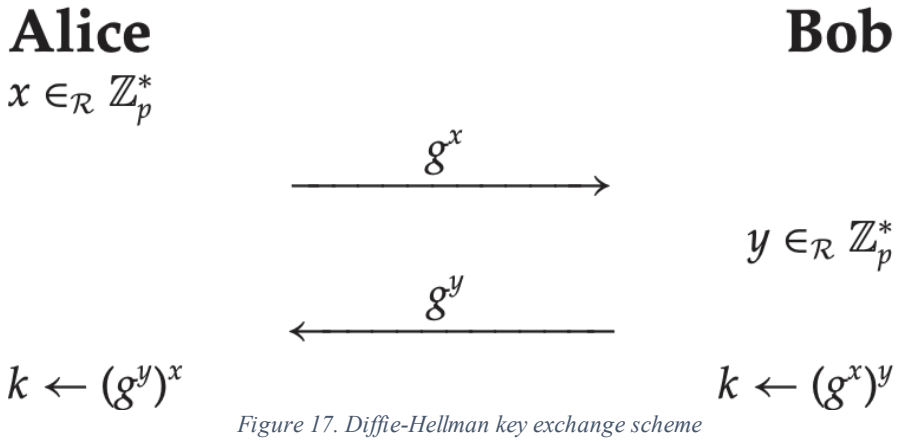



*Figure 17. Diffie-Hellman key exchange scheme*

specific identity, attached comments might contain personal information, and therefore the file has to be encrypted.

Once the data are encrypted, the REST endpoint of the Hubro API receives them as a multipart attachment of the HTTP request. The data needs to be split when the reverse proxy server sets the upper limit of the file size. Upon receiving the file content, the encrypted data is stored in a database as a blob record type.

When the study manager requests to fetch the user's data from the database, he can do so through the web browser interface. Since the database may contain multiple records, as of now, only the latest one is selected by default. This approach expects that full updates are provided. If incremental updates are used, the mechanism would essentially deliver only the latest differentiation. Once the data is picked up from the database and decrypted on-the-fly, it is served back through the response to the study manager. The process is completed by disposing of the utilized memory space, which was used for decryption.

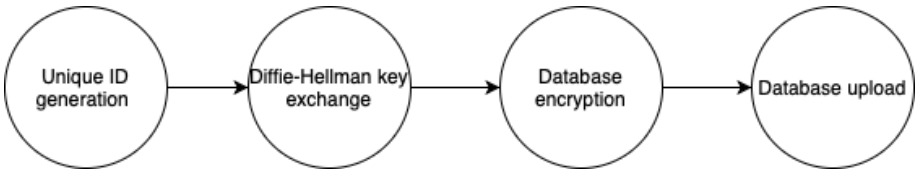



*Figure 18. Self-registered data upload – the process starts with a unique ID generation (A), followed by a key-exchange (B), data encryption (C) and is finished by a data upload.*

## Questionnaire Delivery and Processing Platform

The questionnaire service is provided through the open-source product LimeSurvey [49]. LimeSurvey is an on-line statistical survey web application, which allows to create, distribute and manage web-based surveys. The layout and design of the surveys can be modified using a template system, that makes surveys reusable in multiple different interventions.

LimeSurvey also integrates with LDAP server, using it as an authentication backend. In the Hubro setup, the LimeSurvey's authorization backend is hooked up with LDAP1 server and

therefore survey administrators can log in with the same credentials that they use for logging into other components of Hubro.

Settings of the LimeSurvey had to be adjusted in order to conform with the rest of the Hubro system email distribution settings. Since LimeSurvey itself produces email reminders that are distributed to study participants, the LimeSurvey service has to be hooked up to the translation SMTP server.

### Usage Logs Collection Platform

Besides the secure user-data collection service, we have identified application usage logging as another modality, which needs to be captured to understand users' interaction with the mobile application fully. Usage logging complements questionnaire-based interventions on a more granular level of interaction tracking. These logs help us discover more profound insights into the use of the application.

For this purpose, we have identified and integrated Matomo (formerly Piwik) [50], which is an open-source product, primarily designed to support web analytics. Alongside the cloud-hosted setup, the Matomo is also providing a self-hosted setup. Self-hosting is strongly desirable for use in Hubro, since no data are stored within an external cloud service. As of June 2018, the Matomo has been used by more than 1,455,000 websites around the world.

Even though the Matomo is mainly targeting web application usage tracking, it is also possible, to a certain extent, to use Matomo for usage tracking of smartphone applications. An open-source community, which has developed around the product, maintains ready-to-use client libraries for most major mobile and web-development platforms, which makes its integration on a client-side very straightforward.

The atomic usage unit in Matomo is called a 'visit'. Visit gathers sequences of usage logs where a cut-off threshold determining visit's end is a users' inactivity for 30 minutes. This threshold is configurable on server-side, and the change applies globally with a lasting effect for all users tracked in the system. Usage activity gathered in visit chunks is called 'action'. Every action has a set of properties that describes how the user interacted with the application.

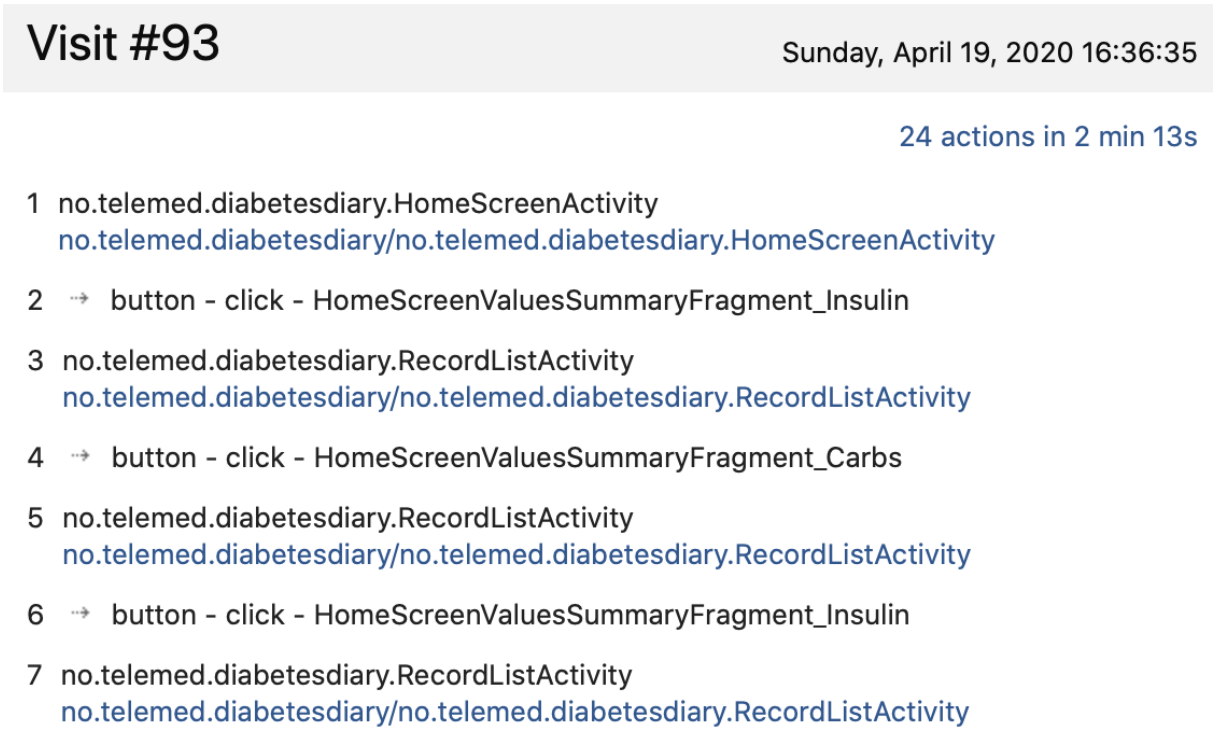



*Figure 19. Sample of collected usage logs as viewed from the Matomo web user interface.*

Identification of users is provided on behalf of Matomo. The identification mechanism logic is based on a decision tree. Its input variables are clients first- and third-party cookies, browser-based fingerprints or optional specific user identifier. Thanks to this granular identity control, it is possible to combine a usage tracking for random and returning visitors participating in a specific study. That means Matomo can also generate a general insight into the application usage outside of the study scope.

Matomo provides rich usage data visualization features implemented directly in its web user interface. However, it also offers options to export the raw usage data into a JSON or XML format. These raw data might be used for further processing to determine possible correlation with other variables collected throughout the study.

### Web Frontend

One of the aims was to provide an easy-to-use administration interface, eventually usable also from a smartphone, on-the-go if needed. Based on these requirements, we have chosen a web application as a user interface for the Hubro.

The user interface of the web application is responsive, i.e. it scales nicely on small device screen sizes. This way, the study manager can interact with the system dynamically through the smartphone or tablet at every place with internet access.

The web application is written in a Django framework [51], which is a high-level Python-based web framework. Django, at its core, can be observed as a Model-View-Controller (MVC) framework. We have selected it due to its simplicity and its pluggable architecture, which would allow us to integrate with more third-party components. Also, the rest of the Hubro components have been written in Python, and therefore we have not introduced a mix-up of programming languages by utilizing Django.

At the front of all services that need to be accessible from an external network, we placed a reverse proxy server. This proxy server is responsible for the routing of requests coming from an external network to individual components of the system, i.e. to the web frontend, Zato service bus and the REST API.

The control interface is available upon authorizing against LDAP1 server, i.e. the study manager uses the same credentials as for accessing other components of Hubro. Once the study manager logs in, the Hubro presents all projects currently served by the system.

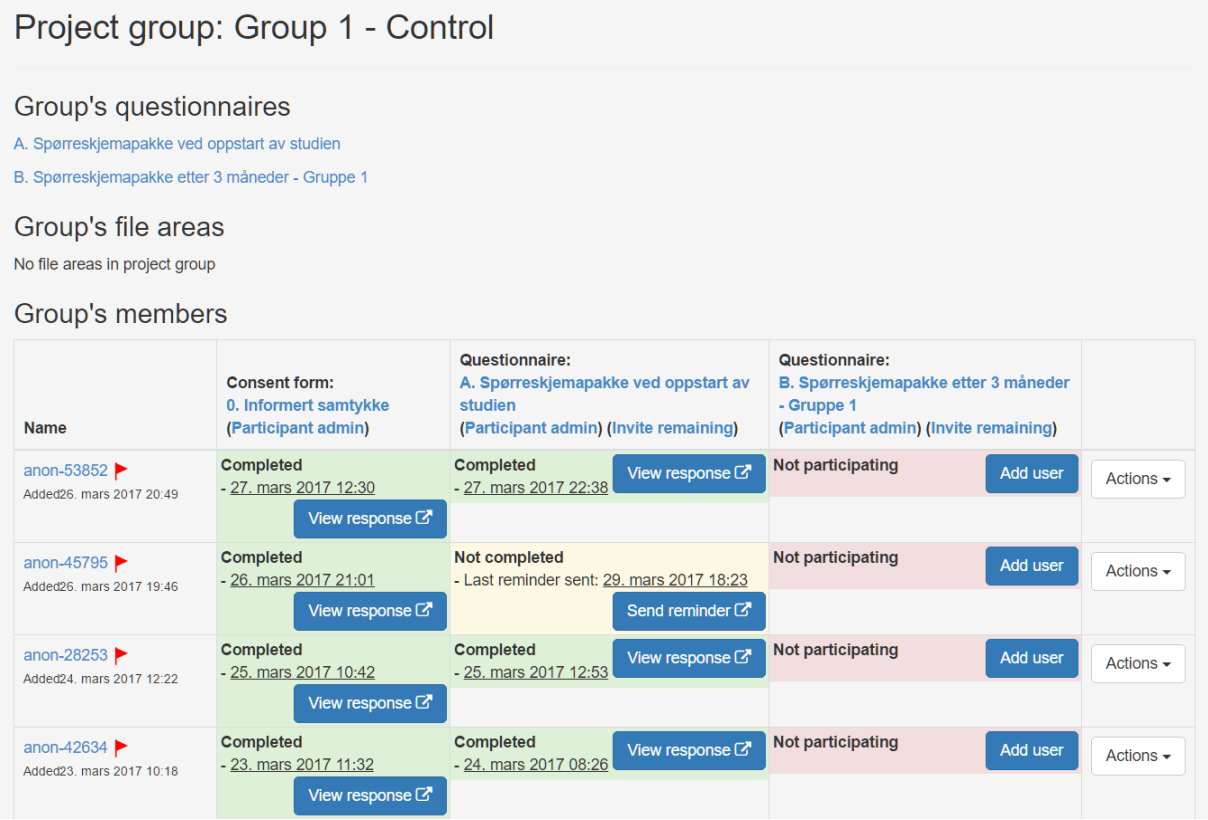



*Figure 20. Treatment group management as viewed by study manager in the Hubro web frontend.*

The web front-end uses an internal database, which is kept in one-way sync with the LDAP0 server and LimeSurvey's internal database. Mutable operations affecting the status of participants (randomization, explicit treatment group assignment or questionnaire assignment) work directly with the authority instance (i.e. with the LDAP server or LimeSurvey API) followed by a trigger of re-synchronization process.

## Decision Support Modules

The Hubro system can generate a vast amount of data depending on its specific application. At the time, the collected data are provided in a different format through individual services' APIs, which may hinder its batch processing in various tools. Therefore, the manual data analysis performed by researchers can be demanding in time and also in resources and needs the assistance of a person providing technical support. Pluggable modules can implement connectors with well-known and widely supported data exchange formats such as HL7 FHIR [52], or OpenEHR [53].

Another use case of utilizing decision support modules would cover smart triggering of alarms for the researcher. These alarms would be based on a decision support algorithm, generated using predefined rules set by a study administrator. These rules would consist of one or multiple conditions related to the status of patient progression throughout the study (survey completion, delivered messages). This approach would eliminate situations, when the study manager has to frequently check for different indicators, to distinguish patients that need additional attention in terms of reminders or technical assistance.

In addition to providing collected set of unprocessed data, the Hubro system can be capable of automatic generation of additional descriptive or predictive information through pattern detection algorithms, machine learning and artificial intelligence. Implementing this feature should not be difficult due to the system's loosely coupled architecture and programming language independence. A use case that demonstrates the use of AI applied has been presented by an EHR vendor Epic, which developed a sepsis prediction module [54]. Another use case, presented by Cerner HealtheDataLab, includes HbA1c prediction for outpatient populations and heart failure prediction [55, 56].

As of now, Hubro supports a generic study workflow, i.e. manual staging of patients through multiple questionnaires combined with a continuous collection of data. While this design can present a good fit for many study protocols, it may be unusable for more complex studies or studies with many participants. As of now, study administrators have to perform many manual check-ups and interventions to keep up timely interaction with study participants in terms of questionnaire distribution and other concerns. Staying in touch is especially important during the recruitment and starting phase of the study, but after some time it may become exhausting, and thus also more vulnerable to human error. A natural expansion of the naïve approach would be an integration of a workflow manager, that would enable better, automatic control of patient staging throughout the whole study. This way, the study administrator would not need to send messages or reminders based on manual assessment of group membership, questionnaire completion or other indicators, as these would present the workflow manager's input parameters. Output actions of workflow manager would consist of questionnaire distribution, message delivery, data processing trigger and others.

Due to a loosely-coupled design of the Hubro system architecture, integration of modules with data post-processing capabilities does not require modification of existing components but

instead makes use of individual services through their APIs. We have developed a proof-of-concept module that analyzes diabetes data collected throughout the study from a specific target patient group and estimates HbA1c over the last several months for each patient.

## Evaluation of the Hubro System

Hubro has been used in two studies with 50 and 8 participants, that were divided into intervention and control treatment group. Both studies aimed to test the new functions of the Diabetes Diary application. They recruited participants through the Diabetes Diary smartphone application and social networks.

The first study has lasted for 11 months, aiming to evaluate new functionality integrated into the Diabetes Diary application. The recruitment has been done online, through social networks such as Facebook and LinkedIn, resulting in 50 patients that participated in the study. Anonymous IDs have been generated for patients in both groups. Automatic self-registered data and usage data collection have been enabled by entering this ID into the Diabetes Diary application. The treatment group of patients has been provided with a new version of the Diabetes Diary via electronic distribution channels alongside with a set-up guide that facilitated the installation and data migration process[1]. Participants have been informed about the progress of the study at regular intervals using the Hubro messaging system utilizing both email and in-app messages. Same communication channels have been used to distribute questionnaires throughout the study.

During the first, pilot study the study manager spent the following amount of time on each task per participant: Informed consent delivery and collection (2-minutes); Randomization (1-minute); Delivery of the Initial questionnaire (1-minute); App distribution (4-minutes); Mid-study questionnaire (1-minute); and Final questionnaire (1-minutes). Minutes spent logging into the system, checking participant status, sending questionnaire reminders and other tasks approximately tripled these times. In total, 30-minutes have been required per user.

[1] Initially, the treatment groups have been using two versions of the application, and it was crucial to make the data migration process from the original version to the new version seamless and robust so that patients do not encounter any problems.

The second study has been conducted for 6 months with 8 patients recruited through their health care providers. The research team sent an interest form link to the patients, that have shown interest in participating. Anonymous IDs have been generated for those who have signed up through the interest form. This ID linked the identity with both LimeSurvey and Piwik/Matomo usage analytics databases. Both emails and in-app messages were used for communication with participants. These were primarily utilized for distribution of links to questionnaires, study progression information and monthly follow-up messages.

During the period of the second study, a researcher had to actively check for responses from health care providers in LimeSurvey. These responses contained their experience with the intervention itself, and have been inserted into the LimeSurvey after every consultation with a patient, including the patient's ID. After six months, at the end of the intervention, the researcher would make sure that the patient participant had gone to the doctor by checking LimeSurvey. Once this has been confirmed, the researcher would send the second questionnaire set to the participant.

### Evaluation Questionnaire

The following questionnaire has been distributed to three respondents, that have been directly working with the Hubro system, throughout the two studies.

1) Can you describe your role/competence when using the Hubro system?
2) Do you find the overall approach effective in comparison with a traditional methods approach in following aspects? Speed, reliability, management of participants, cost-effectivity.
3) What functions are you missing?
4) About which functions do you think should be implemented better/in a more user-friendly way?
5) Have you experienced any problems with the system?
6) Have you used the possibility to manage the study from the smartphone?
7) Do you find the user interface intuitive?
8) Have you got any positive/negative feedback from the study participants?

### Summary of Questionnaire Responses

1) The questionnaire has been filled in by 3 respondents –PI/project manager, researcher and system developer. PI/project manager and researcher have been involved in all phases of two studies, while system developer has provided technical support (service availability assurance and data extraction support).

2) Speed, reliability, cost-effectivity and convenience of participants management were appreciated in the questionnaire answers. A concern has been expressed to a concept of two user identifiers - user ID and anonymous ID, which was hard to understand. This concept was introduced to provide study participants with a way to quickly join the study, i.e. by providing a 5 characters identifier (user ID) that was connected fully qualified GUID-based identifier (anonymous ID) in the database. Both identifiers have to be used when performing basic operations such as, e.g. extracting user data from the database, deanonymizing the user, which introduces an additional level of manual effort. In future, this concept should be revisited and implemented a more user-friendly manner.

3) Based on the feedback from researchers, following new functionalities and system updates would be appreciated: more accessible user-interface; integration of reminders; integration with more advanced recruitment capabilities other than those currently implemented; recruitment scheduling and monitoring; notifications; more secure implementation of the messaging function.

   Previously mentioned connection between user-ID, anon-ID and data in third-party systems (Piwik/Matomo, LimeSurvey) was confusing when performing various operations such as fetching user data or exporting usage logs. Two respondents raised this issue. Another concern was expressed about the utilization of third-party tools - usage analytics platform (Piwik/Matomo) and survey platform (LimeSurvey), specifically questioning their choice due to their complexity and low user-friendliness. These tools are complex, designed to suit a wide range of use cases, and therefore they are coupled with a variety of options and settings, that might be confusing for researchers/study administrators, who only need to use a specific subset of these functions. Although, within the narrowed portfolio of alternatives, these tools stand out among other choices in terms of integration possibilities.

4) Researchers mentioned following optimizations to minimize manual efforts and frequent checkups in following ways - a better-implemented process of operating with patients, i.e. placing them into groups; easier distribution of questionnaires and tools (applications); more streamlined process of questionnaire completion checks, a user interface to interact directly with the LDAP servers (e.g. removing participants).

5) From the researcher's point of view, one of the identified difficulties, was LimeSurvey limited export capability, that only allowed PDF exports, which was not easy to import further in a spreadsheet editor such as MS Excel. Also, it was suggested to implement a two-way messaging function, which would mitigate the necessity to use an ordinary email as a primary communication channel from patients to researchers. Also, the user interface for user's look-up based on their email addresses was perceived as a candidate for improvement, as it required a manual effort in terms of email client utilization as discussed within Question 2. Terms *Texts* and *Messages*, used in the Hubro system, were seen as potentially misleading and confusing for an uneducated researcher using the system. From the patient's perspective, a case was reported when a patient was able to only fill half of the questionnaire, due to accidentally clicking on send button, submitting it in prematurely (post-submission edit function is not implemented). Only a few cases have been reported when the patient got insufficient information or information, that was not providing enough indices on how to proceed further.

6) PI/project manager has used the Hubro functions to recruit and randomize study participants through the smartphone when travelling. The Hubro system and its third-party components are web-based and responsive so that they can be used with mobile devices such as tablets or smartphones.

7) Mixed opinions have been reported on Hubro user interface user-friendliness. An agreement has been given to complexity/steep learning curve for both main third-party components – Piwik/Matomo and LimeSurvey. The complexity of these tools was discussed in question 3.

8) Researchers stated that no negative feedback has been reported about Hubro. The primary investigator (PI) stated there the feedback was overly positive when presenting the system at conferences, workshops and other occasions.

| | |
|---|---|
| **Missing functions** | • Speed<br>• Reliability<br>• Cost-effectivity<br>• Convenience of participants management |
| **Functions that should be implemented in a more user-friendly way** | • Easier user-interface<br>• Integration of reminders integration with more advanced recruitment capabilities other than those currently implemented<br>• Recruitment scheduling and monitoring notifications<br>• More secure implementation of the messaging function |
| **Problems with the Hubro system** | • Limited export capabilities<br>• User's look-up (deanonymization) based on their email addresses requires much manual effort<br>• Few patients reported insufficient information on how to use the system |

*Table 2. Summary of identified missing functions, functions that need improvement and problems with the Hubro system.*

## Discussion and Future Considerations

In this thesis, we have been addressing identified challenges in mobile health interventions in the following ways:

- we have assessed health data-exchange capabilities of wearable devices and identified barriers, that prevent them from being integrated with existing systems
- we have demonstrated wearable devices potential to improve patients comfort when dealing with a chronic disease by the development and evaluation of the first combined smartphone-smartwatch diabetes diary application
- we have addressed a need for updated evaluation approaches for eHealth and mHealth interventions by development and evaluation of the system for electronic management of clinical studies

In this chapter, we further elaborate on specific topics related to the Hubro-system, describe the potential for further development and system's limitations. We also present a future outlook on the development of wearable devices with possibilities for health data exchange, and we discuss how mHealth software developers, EHR providers, and patients relate to these wearable sensor systems.

## Future Improvements and Current Limitations of the Hubro System (Publication 3.)

Functionality-wise, Hubro currently does not integrate tools to implement study-status indication on participant's end, i.e. an indication of progress through the study or option to leave the study without contacting the study manager. These actions have to be handled manually by a study manager on request of a participant. Also, Hubro does not support a handwritten digital signature or any other type of signature for signing the informed consent document. If an institutional review board (IRB) required this, it would have to be implemented accordingly.

It is important to note, that Hubro, currently, neither collects nor generates documents formally required by regulatory authorities or compliance protocols. Essentially, this implies a lack of a Trial Master File (TMF) or its electronic equivalent (eTMF), that has to be maintained alongside the electronic study management system separately in a dedicated repository. Similarly, the Hubro keeps a very limited audit trail of specific actions, that require access to the link table, or the implemented link function respectively (deanonymization). Implementation of this feature is tightly connected with an introduction of further extension of access management with granular access rules.

Hubro has not been extended with granular access management rules that would help to implement role-based access for study administration. This limitation has to be considered when a single instance of Hubro is aimed to be used by multiple researchers to perform multiple studies. Fine-grained role-based access control could spread the perimeter of persons operating with the system to other personnel such as doctors, nurses or assistants. This drawback is partly compensated by a low effort which is needed to spin up an additional, isolated instance of Hubro.

As for now, Hubro also does not integrate eligibility module that might be found in other similar systems. Eligibility module can be seen as a complementary module to the consent, acting as

an instrument to check whether the user meets the inclusion criteria. Therefore, currently, the inclusion criteria have to be placed into an 'interest form' and checked when expressing the interest to join the study by the participant.

Another feature which is commonly integrated into other similar systems is a visualization of collected health data through the graphical user interface (GUI) elements such as plots or charts in the web application. Although this function provides a brief insight into the received data, it is assumed that for the study final evaluation's sake the researcher uses visualization and statistical data processing tools, that overachieve the presentation functionality of web application, that would have to be tailored per trial's needs. Also, the Diabetes Diary smartphone application integrates with a ShareLive system [57]. The ShareLive system could be used by a researcher as a complementary tool to quickly peek into patient's data if necessary.

While some of the similar identified systems (AppBakery, MoRe) makes it possible to plan, create and run the whole study without any programming skills, the full integration with Hubro requires certain development efforts. On the other hand, Hubro allows performing simple questionnaire-based studies with messaging support, without any need for programming skills.

We have identified multiple research systems that are targeting only an individual platform. Given the uneven global market share of two major mobile platforms (Android - 75.82%, iOS - 22.9%) combined with asymmetrical economic strata distribution, researchers might be reluctant to exclude one of the platforms, as a factor that would introduce a coverage bias to the study outcomes. Utilization of two different systems would increase the development workload. Therefore the most optimal option is presented by a framework supporting all major mobile platforms, or by a framework, which is generic enough to be integrable with all of these platforms.

Given that minority of wearable devices with possibilities for health data exchange allow direct integration with mobile apps, it is reasonable to let users aggregate their physical activity data to their smartphone via third-party fitness services, instead of implementing individual API connectors. Thus, the Hubro system targets a wide range of wearable devices, that are sourcing the data into a single aggregator. Since these APIs (if available) can be a subject of updates, using the aggregator instead also partially reduces the development load.

## Wearable Computing in a Personal Health Care (Publication 1., Publication 2., Publication 4.)

We have demonstrated the usability of a wearable device on an example of a combined smartwatch/smartphone diabetes self-management application. Pebble smartwatch that has been used for this demonstration is distinct from the majority of other similar devices currently out on the market. Main differences include the monochromatic display, button controls and long battery life. These first-generation smartwatch attributes have been although highly appreciated by the testing group during the evaluation period. Currently, the application market offers more alternatives, that are implementing functionality similar to our application. These applications are utilizing various new types of input (touch display, digital crown, voice input), novel user interfaces and new sensors - all of these updates call for a further usability reevaluation of a combined smartwatch/smartphone diabetes diary application.

Besides smartwatches, the market is saturated with other types of wearable devices, that are equipped with a variety of health sensors and connected to the smartphone. It becomes more common that patients are using multiple wearable devices, making the data interoperability even more critical parameter. If this is properly implemented, wearable devices could serve as continuous, relatively low-cost monitoring tools of different health parameters such as ECG or $SpO_2$. The importance of interoperability is even more stressed, as a new trend is to use modular accessories, that can extend the functionality by adding additional sensors.

From a developer's viewpoint, the lack of standardized communication APIs limits portfolio of devices, that can be directly integrated with third-party mHealth software without the use of supporting cloud-based data collection service. This lack presents a fundamental motivation for the existence of several research projects, that explore and exploit off-the-record data exchange possibilities of various wearable devices. These projects are often tightly associated with DIY solutions such as Nightscout, The Open Artificial Pancreas System (OpenAPS) [58] and similar.

When integrating a wearable device into an EHR system, we have identified three factors to consider – data reliability, data transmission risks and device certification. The last one, device certification, is aimed to ensure that the standardization, privacy and security issues are considered and adequately implemented. Notably, legal requirements, data protection laws and

regulations differ between regions, and their interpretation may be difficult as wearables represent consumer-grade devices and, at the same time, they collect personal health data.

## Conclusion

The motivation behind the need to move to new evaluation methods is driven by the fact that traditional methods of interventions may provide outdated results, given the fast pace of technological advances. Therefore, it is necessary to react on and evaluate these changes promptly. Inadequate old methods are used on new types of interventions that involve the processing of vast amount of data, dynamic support and troubleshooting of potential participant's technical problems during the study and inclusion of various types of new technology. Commonly implemented clinical trial protocols are too time-consuming to provide results in a reasonable time as they are not systematically addressing identified challenges.

The Hubro system consists of core custom-implemented modules and multiple loosely coupled, independent third-party, open-source components. Since the system does not explicitly define study workflow, we find the system suitable for both small- and large-scale interventions, given that the study protocol would expect engagement of specific subset of components, e.g. performing only survey-based interventions. Also, most of the user-facing technology opened to the outside network, as places of potential security threat, are hardened by continues updates provided by third parties development teams. The system can be installed on-premises using Kubernetes, isolating the infrastructure layer. This isolation enables installation within the organization's IT facility, where the security measures are already in place. However, it can also be easily deployed into a cloud computing environment.

Loosely coupled architecture facilitates the integration of additional data processing modules. In addition to providing collected set of unprocessed data, the Hubro system can be extended with automatic generation of additional descriptive or predictive information through pattern detection algorithms, machine learning and artificial intelligence.

New types of pervasive technology become tools of daily use for many patients. They continuously adopt these technologies and collect health data from various types of sensors. These sensors can help clinicians get an objective view into a patient's lifestyle, given that the

data can be transferred from the device to the clinician or the patient's EHR. However, in the current situation, the data exchange possibilities are limited, making it challenging to implement straightforward health data transfer using a standardized data transfer protocol.

The system we designed proved to improve areas of recruitment, enrollment, engagement, and retention of participants into an RCT. Although there are several other products available out on the market, none of them is integrating such a generic toolset, aiding the process of patient recruitment, randomization, survey and consent distribution, patient follow-up, data processing and trial closure. Inconclusive clinical trials, as a consequence of clinical trial's disrupted integrity, may result in a delay of development of optimal treatment method, or an additional financial burden.

# Bibliography


[1] M. Muzny, A. Henriksen, A. Giordanengo, J. Muzik, A. Grøttland, H. Blixgård, G. Hartvigsen and E. Årsand, "Wearable sensors with possibilities for data exchange: Analyzing status and needs of different actors in mobile health monitoring systems," *International journal of medical informatics,* vol. 133, 2020.

[2] E. Årsand, M. Muzny, M. Bradway, J. Muzik and G. Hartvigsen, "Performance of the first combined smartwatch and smartphone diabetes diary application study," *Journal of diabetes science and technology,* vol. 9, no. 3, pp. 556-563, 2015.

[3] E. Årsand, M. Bradway, H. Blixgård, M. Muzny, A. Giordanengo, A. Grøttland and G. Hartvigsen, "Experience from using a dynamic study management service for an mHealth diabetes type 2 RCT," *DIABETES TECHNOLOGY & THERAPEUTICS,* vol. 20, p. 73, 2018.

[4] A. Henriksen, M. H. Mikalsen, A. Z. Woldaregay, M. Muzny, G. Hartvigsen, L. A. Hopstock and S. Grimsgaard, "Using Fitness Trackers and Smartwatches to Measure Physical Activity in Research: Analysis of Consumer Wrist-Worn Wearables.," *Journal of medical Internet research,* vol. 20, no. 3, 2018.

[5] Q. Pham, D. Wiljer and J. A. Cafazzo, "Beyond the Randomized Controlled Trial: A Review of Alternatives in mHealth Clinical Trial Methods," *JMIR mHealth and uHealth,* vol. 4, no. 3, 2016.

[6] M. Bradway, E. Gabarron, M. Johansen, P. Zanaboni, P. Jardim, R. Joakimsen, L. Pape-Haugaard and E. Årsand, "Methods and Measures Used to Evaluate Patient-Operated Mobile Health Interventions: Scoping Literature Review," *JMIR mHealth and uHealth,* vol. 8, no. 4, 2020.

[7] S. J. Coons, S. Eremenco, J. J. Lundy, P. O'Donohoe, H. O'Gorman and W. Malizia, "Capturing Patient-Reported Outcome (PRO) Data Electronically: The Past, Present, and Promise of ePRO Measurement in Clinical Trials," *The Patient-Patient-Centered Outcomes Research,* vol. 8, no. 4, p. 301–309, 2015.

[8] G. Lieberman and M. R. Naylor, "Interactive voice response technology for symptom monitoring and as an adjunct to the treatment of chronic pain," *Translational behavioral medicine,* vol. 2, no. 1, 2012.

[9] I. Vaghefi and B. Tulu, "The continued use of mobile health apps: insights from a longitudinal study," *JMIR mHealth and uHealth,* vol. 7, no. 8, 2019.

[10] Hans Van Remoortel et al., "Validity of activity monitors in health and chronic disease: a systematic review," *International journal of behavioral nutrition and physical activity,* vol. 9, no. 1, 2012.

[11] K. Price, S. R. Bird, N. Lythgo, I. S. Raj, J. Y. L. Wong and C. Lynch, "Validation of the Fitbit One, Garmin Vivofit and Jawbone UP activity tracker in estimation of energy expenditure during treadmill walking and running," *Journal of Medical Engineering & Technology ,* vol. 41, no. 3, 2016.

[12] W. J. Gordon, A. Landman, H. Zhang and D. W. Bates, "Beyond validation: getting health apps into clinical practice," *NPJ Digital Medicine,* vol. 3, no. 1, 2020.

[13] Google, LLC, "Google Fit," [Online]. Available: https://www.google.com/fit/. [Accessed 2020].

[14] Apple Inc., "iOS - Health - Apple," [Online]. Available: https://www.apple.com/ios/health/. [Accessed 2020].

[15] E. Årsand, M. Bradway, M. Muzny, H. Blixgård, A. Grøttland, A. Giordanengo and G. Hartvigsen, "Warning: the Do-It-Yourself (DIY) wave will drastically change diabetes care!," *International Journal of Integrated Care,* vol. 16, no. 5, 2016.

[16] Nightscout Project, *Nightscout# WeAreNotWaiting,* Nightscout Project McKinney, TX, 2017.

[17] D. Klonoff, "Cybersecurity for connected diabetes devices," *Journal of diabetes science and technology,* vol. 9, no. 5, 2015.

[18] E. Årsand, N. Tatara, G. Østengen and G. Hartvigsen, "Mobile phone-based self-management tools for type 2 diabetes: the few touch application," *Journal of diabetes science and technology,* vol. 4, no. 2, 2010.

[19] ASICS Digital, Inc., "RunKeeper," [Online]. Available: https://runkeeper.com/cms/. [Accessed 2020].

[20] D. Peiris, J. J. Miranda and D. C. Mohr, "Going beyond killer apps: building a better mHealth evidence base," *BMJ Global Health,* vol. 3, no. 1, 2018.

[21] L. Piwek, D. A. Ellis and A. Sally, "Can programming frameworks bring smartphones into the mainstream of psychological science?," *Frontiers in Psychology,* vol. 7, p. 1252, 2016.

[22] I. Darmawan, C. Bakker, T. A. Brockman, C. A. Patten and M. Eder, "The Role of Social Media in Enhancing Clinical Trial Recruitment: Scoping Review," *Journal of medical Internet research,* vol. 22, no. 10, 2020.

[23] D. Richardson SM, *Wearable Device Inventory,* Queen's University, 2017.

[24] Vandrico Solutions Inc, "Wearable Technology Database | Vandrico Inc," [Online]. Available: https://vandrico.com/wearables.html. [Accessed 2020].

[25] GsmArena, "GsmArena," [Online]. Available: http://www.gsmarena.com/results.php3?sFormFactors=8. [Accessed 2020].

[26] WearablesCom, "WearablesCom," [Online]. Available: http://www.wearables.com/devices. [Accessed 2020].

[27] Specbucket, "Smart Watches," [Online]. Available: https://smartwatches.specbucket.com. [Accessed 2020].

[28] Prisguide, "Aktivitetsarmband," [Online]. Available: https://prisguiden.no/kategorier/aktivitetsarmband. [Accessed 2020].

[29] Pebble Technology, "Pebble: E-Paper Watch for iPhone and Android," [Online]. Available: https://www.kickstarter.com/projects/getpebble/pebble-e-paper-watch-for-iphone-and-android. [Accessed 2020].

[30] S. Dick, Y. O'Connor and C. Heavin, "A comparison of mobile health evaluation techniques," in *23rd Americas Conference on Information Systems*, 2017.

[31] ResearchKit - Apple Developer, "ResearchKit," [Online]. Available: https://developer.apple.com/researchkit/. [Accessed 2020].

[32] Small Data Lab & Open mHealth, "ResearchStack," [Online]. Available: http://researchstack.org. [Accessed 2020].

[33] Applied Informatics Inc., "ResearchDroid – Android port of ResearchKit," [Online]. Available: https://appliedinformaticsinc.com/researchdroid/. [Accessed 2020].

[34] CareKit - Apple Developer, "CareKit," [Online]. Available: https://developer.apple.com/carekit/. [Accessed 2020].

[35] TrialX, "Appbakery - Mobilize Research," [Online]. Available: http://trialx.com/appbakery/. [Accessed 2020].

[36] T. Vandenberk, V. Storms, D. Lanssens, H. De Cannière, C. J. P. Smeets, I. M. Thijs, T. Batool, Y. Vanrompay, P. M. Vandervoort and L. Grieten, "A Vendor-Independent Mobile Health Monitoring Platform for Digital Health Studies: Development and Usability Study," *JMIR mHealth and uHealth,* vol. 7, p. e12586, 2019.

[37] M. Zens, B. Grotejohann, A. Tassoni, F. Duttenhoefer, N. Südkamp and P. Niemeyer, "Development of a modular research platform to create medical observational studies for mobile devices," *JMIR Research Protocols,* vol. 5, no. 4, 2017.

[38] Y. Al-Issa, M. A. Ottom and A. Tamrawi, "eHealth cloud security challenges: A survey," *Journal of Healthcare Engineering,* 2019.

[39] K. Hightower, B. Burns and J. Beda, Kubernetes: up and running: dive into the future of infrastructure, O'Reilly Media, Inc., 2017.

[40] The PostgreSQL Global Development Group, "PostgreSQL: The World's Most Advanced Open Source Relational Database," [Online]. Available: https://www.postgresql.org. [Accessed 2020].

[41] L. Gelinas, R. Pierce, S. Winkler, I. G. Cohen, H. F. Lynch and B. E. Bierer, "Using social media as a research recruitment tool: ethical issues and recommendations," *The American Journal of Bioethics,* vol. 17, p. 3–14, 2017.

[42] J. Efird, "Blocked randomization with randomly selected block sizes," *International journal of environmental research and public health,* vol. 8, p. 15–20, 2011.

[43] J. C. Ferreira and C. M. Patino, "Randomization: beyond tossing a coin," *Jornal Brasileiro de Pneumologia,* vol. 42, no. 5, 2016.

[44] OpenLDAP Foundation, "OpenLDAP Software," [Online]. Available: https://www.openldap.org.

[45] D. Chappell, Enterprise service bus, O'Reilly Media, Inc., 2004.

[46] Zato Source, "Zato," [Online]. Available: https://zato.io. [Accessed 2020].

[47] MongoDB, Inc., "The most popular database for modern apps | MongoDB," [Online]. Available: https://www.mongodb.com. [Accessed 2020].

[48] D. Carts, "A review of the Diffie-Hellman algorithm and its use in secure internet protocols," *SANS institute,* 2001.

[49] C. Schmitz and others, "LimeSurvey: An open source survey tool," *LimeSurvey Project Hamburg, Germany. URL http://www. limesurvey. org,* 2012.

[50] Matomo, "Matomo Analytics," [Online]. Available: https://matomo.org. [Accessed 2020].

[51] Django Software Foundation, "Django Framework," [Online]. Available: https://www.djangoproject.com. [Accessed 2020].

[52] D. Bender and K. Sartipi, "HL7 FHIR: An Agile and RESTful approach to healthcare information exchange," in *Proceedings of the 26th IEEE international symposium on computer-based medical systems*, 2013.

[53] K. Dipak, B. Thomas and H. Sam, "The openEHR foundation," *Studies in health technology and informatics,* vol. 115, 2005.

[54] "Epic | ...with the patient at the heart," Epic Systems Corporation, [Online]. Available: https://www.epic.com. [Accessed 2020].

[55] L. Ehwerhemuepha, G. Gasperino, N. Bischoff, S. Taraman, A. Chang and W. Feaster, "HealtheDataLab–a cloud computing solution for data science and advanced analytics in healthcare with application to predicting multi-center pediatric readmissions," *BMC medical informatics and decision making,* vol. 20, 2020.

[56] Cerner Corporation, "HealtheDataLab: Improving population health with a secure, next-generation research tool," [Online]. Available: https://www.cerner.com/gb/en/pages/healthedatalab-improving-population-health-with-a-secure-next-generation-research-tool. [Accessed 2020].

[57] M. Bradway, A. Grøttland, H. K. Blixgård, A. Giordanengo and E. Årsand, "System for enabling clinicians to relate to a mobile health app: Preliminary results of the Norwegian trial in the EU FI-STAR project," *Advanced Technologies & Treatments for Diabetes (ATTD),* 2016.

[58] Dana Lewis & the #OpenAPS community., "OPENAPS.ORG #WeAreNotWaiting to reduce the burden of Type 1 diabetes," [Online]. Available: https://openaps.org. [Accessed 2020].