UNIVERSITY OF CALIFORNIA,
IRVINE

# ZotCare: A Novel mHealth Service Provider

DISSERTATION

submitted in partial satisfaction of the requirements
for the degree of

DOCTOR OF PHILOSOPHY

in Computer Science

by

Sina Labbaf

Dissertation Committee:
Distinguished Professor Nikil Dutt, Chair
Professor Amir M. Rahmani
Professor Jessica L. Borelli

2023

# DEDICATION

To my parents Reza & Zohreh, and their determination in a better future.

# TABLE OF CONTENTS

# LIST OF FIGURES

# LIST OF TABLES

# ACKNOWLEDGMENTS

I want to thank the National Science Foundation for supporting my work through the Smart and Connected Communities (S&CC) grant CNS-1831918. Also, the Department of Computer Science at the University of California, Irvine (UCI), for admitting and supporting me academically and financially to present this dissertation.

I also like to extend my gratitude to my advisors, Nikil Dutt, and Amir Rahmani, for their support and guidance. Also, Jessica Borelli, the other dissertation committee member, for providing feedback and participating in my defense.

My appreciation extends to my colleagues at Dutt Research Group, Health Science and Technology Lab, and Institute of Future Health. Also, other collaborators from the Marco Levorato Research Lab, Thrive Lab at the School of Social Ecology at UCI, the School of Nursing at UCI, and the University of Turku, Finland.

Portion of Chapters 1, 2, 3, 5, 6, and 8 of this dissertation is a reprint of the material as it appears in [15], used with permission from Frontiers in Digital Health. The co-authors listed in this publication are Mahyar Abbasian, Iman Azimi, Amir M. Rahmani, and Nikil Dutt.

OpenAI's ChatGPT was used in editing and proofreading this dissertation.

# VITA

## Sina Labbaf

### EDUCATION

**Doctor of Philosophy** in **Computer Science** **2023**
University of California, Irvine *Irvine, California*

**Master of Science** in **Computer Science** **2022**
University of California, Irvine *Irvine, California*

**Bachelor of Science** in **Computer Engineering** **2017**
University of Tehran *Tehran, Iran*

### RESEARCH EXPERIENCE

**Graduate Student Researcher** **2018–2023**
University of California, Irvine *Irvine, California*

### TEACHING EXPERIENCE

**Teaching Assistant** **2017–2018**
University of California, Irvine *Irvine, California*

## REFEREED JOURNAL PUBLICATIONS

**ZotCare: A Flexible, Personalizable, and Affordable mHealth Service Provider** **2023**
Frontiers in Digital Health

**Objective prediction of next-day's affect using multimodal physiological and behavioral data: Algorithm development and validation study** **2023**
JMIR Formative Research

**Sleep Patterns and Affect Dynamics Among College Students During the COVID-19 Pandemic: Intensive Longitudinal Study** **2022**
JMIR Formative Research

**A technology-based pregnancy health and wellness intervention (two happy hearts): case study** **2021**
JMIR Formative Research

**Using multimodal assessments to capture personalized contexts of college student well-being in 2020: Case study** **2021**
JMIR Formative Research

## REFEREED CONFERENCE PUBLICATIONS

**Impact of COVID-19 Pandemic on Sleep Including HRV and Physical Activity as Mediators: A Causal ML Approach** **2023**
BSN

**Loneliness Forecasting Using Multi-modal Wearable and Mobile Sensing in Everyday Settings** **2023**
BSN

**Active Reinforcement Learning for Personalized Stress Monitoring in Everyday Settings** **2023**
CHASE

**Personalized Stress Monitoring using Wearable Sensors in Everyday Settings** **2021**
EMBC

**Data Collection and Labeling of Real-Time IoT-Enabled Bio-Signals in Everyday Settings for Mental Health Improvement** **2021**
GOODIT

**Long-Term IoT-Based Maternal Monitoring: System Design and Evaluation** **2021**
Sensors

## CODE

**ZotCare** `https://github.com/ZotCare`
*The project's code respository*

## SOFTWARE

**ZotCare Android**
*ZotCare Android Application*
https://play.google.com/store/apps/details?id=org.healthscitech.zotcare

**ZotCare Dashboard** `https://panel.healthscitech.org/`
*ZotCare Administrative Dashboard*

# ABSTRACT OF THE DISSERTATION

ZotCare: A Novel mHealth Service Provider

By

Sina Labbaf

Doctor of Philosophy in Computer Science

University of California, Irvine, 2023

Distinguished Professor Nikil Dutt, Chair

The availability of mobile Internet and health devices is enabling new opportunities for health researchers to access an immense amount of digital health data ubiquitously and deliver intervention and treatment online and in real time. However, to take advantage of these opportunities, the researchers need to build systems and applications to collect these data, process them, and deliver actionable items accordingly, which requires building a mobile health (mHealth) system. mHealth solutions are tools designed to assist researchers in creating their desired mHealth systems, but these solutions often have limitations in their setup time, cost, or customization level. This dissertaiton presents ZotCare, a service platform that aims to resolve these limitations by offering ready-to-use mHealth solutions in a shared environment to reduce the cost and provide a service orchestration that can be customized to different types of study by providing tools for personalization and adaptation. ZotCare's contribution also extends to its methods in creating a data pipeline scalable to the number of users, amount of occupied storage, and the load of different types of tasks by considering the opportunities and limitations in mHealth research. This dissertation will provide an introduction and background on mHealth solutions and ZotCare's position. Then, ZotCare's service orchestration is explained to demonstrate its capabilities in creating a customizable environment. This will be followed by ZotCare's data pipeline design challenges to indicate how ZotCare can run mHealth studies at scale. In the end, to showcase ZotCare's

practical contribution, several use cases of ZotCare in mHealth research will be provided, followed by a short discussion on other contributions, artifacts, and future work.

# Chapter 1

# Introduction

The widespread adoption of smartphones, wearable technologies, and other Internet-connected health devices has led to the availability of reliable digital health data streams [10]. These devices and applications have played a significant role in various domains, such as improving lifestyles, achieving fitness goals, monitoring high-risk populations, and enhancing productivity [18]. Many vendors now offer access to the data streams generated by their products, opening up new opportunities for researchers to explore ubiquitous remote monitoring by leveraging different health data streams [29, 9, 7, 33, 6, 25]. For instance, studies such as [1] and [17] have utilized Garmin smartwatches [7] to longitudinally monitor maternal sleep and dementia patients' caregivers, respectively. Furthermore, the rise in mobile Internet connectivity [8] has provided researchers with the ability to promptly interact with participants, facilitating the collection of supplementary information for data modeling or the delivery of interventions within minutes or seconds. By capitalizing on these two opportunities, researchers can not only collect accurate health data streams but also process the information, engage with participants, and implement necessary interventions.

For health researchers, leveraging these opportunities necessitates developing and deploying mobile health (mHealth) applications. These applications perform tasks such as collecting health data streams, processing the data, invoking actions, and receiving feedback. Figure 1.1 outlines a typical mHealth system composed of three critical components: the central cloud server and separate interfaces for researchers/clinicians and participants. The cloud server forms the foundation for data storage, model building, and action invoking aimed at participants and researchers, all while ensuring the preservation of data integrity, security, and participant privacy. The participant interface, another critical component, necessitates real-time interaction capabilities with participants, as well as mechanisms for subjective and objective data collection. Conversely, the researcher dashboard should be furnished with data analysis and monitoring tools essential for executing a mHealth study. Each component operates within a distinct segment of the technology stack and possesses specific functionalities, giving rise to various development and deployment challenges. First, researchers face the complex task of developing a diverse system encompassing various components, ranging from mobile and wearable applications to web servers, requiring diverse programming skills and knowledge. Moreover, after platform development, deploying and maintaining these applications can pose substantial obstacles due to the high frequency, longitudinal nature, and potential scalability of health data streams. These challenges can impede research progress and divert focus from the core experiments.

Several open-source software platforms have been developed to facilitate mobile health (mHealth) studies [11, 27, 2]. These platforms offer a range of tools encompassing servers, mobile applications, and analytics tools, providing researchers with diverse possibilities. Researchers can also reprogram these platforms to suit their specific requirements. While mHealth platforms can reduce the need for extensive development, the deployment burden still rests on the researchers. Additionally, the costs associated with deployment are typically borne by a single organization, making it relatively more expensive for smaller organizations conducting smaller-scale studies.

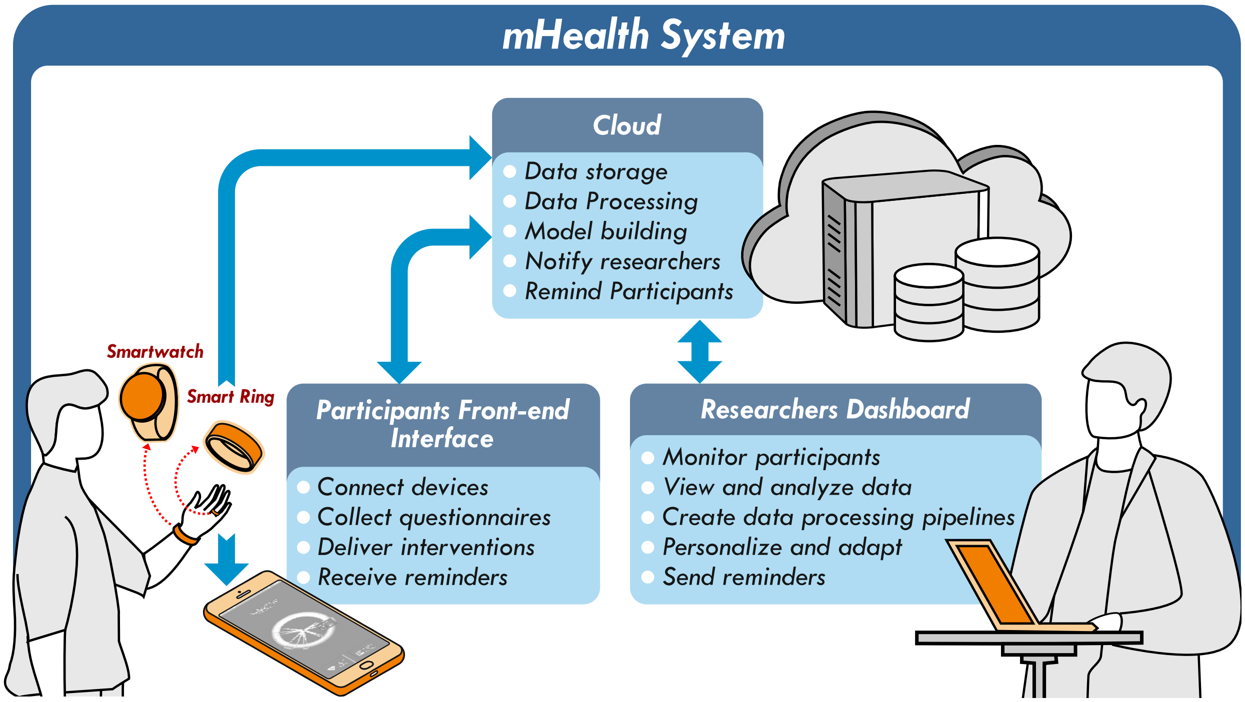



Figure 1.1: An overview of a mHealth system

Alternatively, researchers can utilize online services for conducting their mHealth studies [12, 3]. These services are platform solutions provided and deployed by service providers. They are designed to share resources between different organizations and studies, reducing the time and effort required for developing and deploying a custom mHealth application. By sharing resources, these services effectively cut down costs. Typically, these services offer researchers a dashboard for reconfiguring the services with various options. However, the available configurations may not provide the necessary flexibility required for real-time studies.

Despite the significant advances in existing mHealth solutions, a pressing demand for a comprehensive solution that integrates three essential features into a unified package persists. Firstly, such a solution should offer a ready-to-use setup that eliminates the requirement for computer programming or infrastructure skills, ensuring accessibility for researchers without technical expertise. Secondly, the solution should prioritize affordability by reducing deployment costs through resource sharing and providing reusable components. Lastly, the solution must exhibit flexibility by offering components that can be combined in various ways to accommodate the diverse and evolving demands of modern mHealth studies, such as personalization.

Building a service that is capable of providing these features by design also comes with several technical challenges in the implementation. These features need to meet certain criteria to meet the requirements of a healthcare service. First, this solution needs to be scalable. Scalability for mHealth solutions comes with additional challenges compared to traditional systems since mHealth services operate on real-time, high-frequency data. This increases the burst load on the system while putting a deadline on the end-to-end latency. Besides that, the size of the data is big and can cause disruptions in database systems as more data is stored in the system. Second, these solutions need to execute the logic provided by their users (researchers). This can raise many potential problems. The problems are in security, reliability, and especially resource sharing. This solution needs to implement measures to provide an acceptable quality of service regardless of the load of the processes and their type in the system.

### Dissertation organization

The organization of the rest of this document will be as follows. First, Chapter 2 presents the background on mHealth solutions. In this chapter, different categories of mHealth solutions will be introduced and examples will be provided for each category. These solutions will be analyzed based on different metrics to highlight the gap that exists between these solutions and the solutions that are demanded by researchers. At the end of the chapter, I will introduce ZotCare and explain how ZotCare can contribute to closing that gap.

Chapter 3 will discuss ZotCare's service orchestration. This service orchestration explains how ZotCare's services can work in tandem to provide the environment for researchers that is capable of providing the necessary features for their studies. This chapter will introduce different services and components in ZotCare and explain how end users such as researchers and participants can interact with the system.

Chapter 4 will take a deeper look at building a data pipeline for ZotCare. Configuring and building data pipelines pose major challenges for building a mHealth solution. In this chapter, I will discuss how ZotCare can handle many of the technical challenges faced in building these data pipelines.

Chapter 5 showcases past and current studies that have utilized ZotCare as their mHealth solution. In this chapter, I introduce each study and then discuss how ZotCare was used in the study. Along with that, I will discuss the challenges that these studies faced and how ZotCare could contribute to addressing them.

Chapter 6 touches on some of the other critical aspects faced in building Zotcare, including privacy and security contributions, standardization, and design.

Chapter 7 lists all the artifacts that were produced from or by ZotCare. This list includes services and software, codes, datasets, and related publications.

The thesis concludes in chapter 8 with a discussion of future work and limitations of ZotCare.

# Chapter 2

# Background and Related Work

The adoption of mHealth solutions within healthcare applications has witnessed a significant surge, fueled by the shared objective of improving healthcare delivery and outcomes, as highlighted by [19]. These solutions encompass a wide array of features that greatly facilitate the implementation of mHealth studies. One key aspect is the ability of mHealth solutions to seamlessly integrate wearable devices and diverse data sources, thereby enabling real-time health monitoring. These solutions can also provide data visualization and analytic methods, promote interoperability, and support interventions while ensuring the privacy and security of users.

In the implementation of mHealth solutions, it is crucial to consider and explore three key aspects. The first aspect pertains to the *setup time* required to initiate and configure mHealth studies using the chosen solution. This setup time encompasses various phases, including *system design*, *development*, and *deployment*, each demanding a significant amount of time and effort. These stages involve designing the system architecture, developing the necessary functionalities, and deploying the infrastructure to support the intended mHealth studies.

The second aspect to be considered is the associated *costs* involved in the development and deployment of the mHealth solution. Development costs encompass the investment of human resources and time required for designing and developing the system infrastructure. This includes the efforts of software engineers, data scientists, and other relevant professionals. In addition, ongoing modifications and enhancements may require additional development efforts. Deployment costs encompass the procurement of necessary processing resources, such as servers or cloud infrastructure, as well as ongoing maintenance and operational expenses.

The third aspect revolves around the *customization* capabilities offered by the mHealth solution. Customization can be viewed across three distinct levels: *development*, *configurability*, and *programmability*. At the development level, customization refers to the ability to tailor the solution to meet specific research requirements and objectives. This may involve creating new functionalities or modifying existing ones. Configurability, on the other hand, allows users to adapt the solution's settings and parameters to align with the unique needs of their mHealth studies. Programmability refers to the capability of leveraging programming interfaces or APIs to integrate the solution with other systems or to extend its functionalities.

At the *development* level of customization, researchers are advised to allocate additional efforts to introduce new functionalities and tailor existing mHealth solutions to their specific needs. This level of customization entails direct involvement with the underlying codebase of the solution, thereby necessitating a high level of programming expertise. Researchers must possess the technical skills required to modify the existing code, introduce new functionalities, or make changes to the underlying algorithms. Moving to the *configurability* level of customization, researchers can customize the solution by reconfiguring the available features within the provided framework. This level of customization does not demand extensive technical expertise and programming skills. Instead, researchers can make adjustments to the system's settings, parameters, or options offered by the solution. While configurability provides a certain degree of customization, it may be limited to predefined configurations and settings, constraining researchers from making substantial modifications beyond the available options. Finally, at the *programmability* level of customization, researchers can leverage the solution's programming interfaces or APIs to customize its behavior based on specific situations and conditions. In contrast to development-level customization, programmability-level customization offers researchers the ability to incorporate their own functionalities into the system with minimal effort, without requiring extensive technical expertise. In the following, we will provide an overview of existing mHealth solutions, highlight their limitations, and subsequently present the advantages of our solution, ZotCare, in addressing and bridging these gaps.

## 2.1 Related mHealth Solutions

The existing landscape of mHealth solutions can be broadly categorized into two primary classifications: *platforms* and *services*. *Platforms* encompass comprehensive frameworks that integrate various components of mHealth solutions through the utilization of one or multiple open-source software. One such platform is Radar-base by [27], which focuses on remote monitoring and data collection. It facilitates the integration of data from multiple sensors and devices, enabling comprehensive monitoring capabilities. Another notable open-source platform is mCerebrum by [11], which provides tools for real-time monitoring, data processing, and personalized health interventions based on mobile sensor data. These platforms offer a wide range of features, including data integration, real-time monitoring, analytics, and decision support tools. The Bridge Platform [2] by Sage Bionetworks is another noteworthy example, providing an open-source software framework for digital health research studies. It allows researchers to develop mobile apps, securely collect participant data, and foster participant engagement while emphasizing privacy and data sharing.

However, deploying and utilizing these platforms for mHealth studies require substantial effort, as setting up the necessary software can extend the setup time of studies. Technical challenges may arise, particularly for researchers lacking expertise in Internet infrastructure. Moreover, these platforms are typically designed to operate within a single organization or study, making the deployment costs exclusive to that particular organization. Consequently, this exclusivity can disproportionately affect smaller-scale studies, potentially rendering the deployment financially burdensome. Another significant challenge associated with these platforms is the limited availability of customization methods. While the open-source nature of these platforms provides some level of customizability at the development level, implementing additional features and functionalities typically necessitates the involvement of technically skilled developers. This dependency on technical expertise may hinder researchers' ability to efficiently add or modify elements within the platform to suit their specific requirements.

Conversely, *services* encompass pre-built solutions that are tailored to specific healthcare needs. These solutions are designed to address particular aspects of healthcare and offer a more focused approach. For instance, ilumivu [12] provides a closed-source service that facilitates remote patient monitoring and data collection through user-friendly mobile applications. This service emphasizes patient engagement and includes features for symptom tracking, medication adherence, and communication with clinicians. Ethica [3], another closed-source service, places emphasis on privacy-preserving data collection and analysis. It ensures compliance with privacy regulations while enabling remote monitoring and research data collection.

These services offer ready-to-use features and intuitive interfaces, enabling researchers to swiftly adopt and utilize these mHealth solutions without requiring extensive technical expertise. By providing a streamlined and straightforward setup process, these services enable researchers to initiate their studies promptly, leveraging the available features and minimizing setup time. In terms of costs, services typically entail lower expenses compared to platforms. This is primarily due to the fact that the deployment, maintenance, and resource management burdens are assumed by the service providers. Consequently, these costs are distributed among different studies that utilize the shared resources, making it more cost-effective for researchers. However, services generally offer limited customizability options, particularly with regard to advanced functionalities. Customization opportunities mainly revolve around configuring existing features to align with researchers' needs. Researchers may encounter limitations when attempting to tailor these services to their specific workflows or integrate additional features beyond those provided by the service.

The choice between platforms and services depends on various factors, including specific requirements, available resources, researchers' technical expertise, and study objectives. Different solutions offer distinct trade-offs in terms of setup time, costs, and customization capabilities. Services generally offer shorter setup times and lower costs compared to platforms. The pre-built nature of services allows for swift deployment and immediate utilization of the provided features. However, researchers may face limitations in customizing these services to align precisely with their experimental needs. The available configurations may be restricted to the options provided by the service, potentially constraining researchers in their experimentation. On the other hand, platforms provide a more comprehensive range of customization options. This level of customization, however, typically necessitates expertise in modifying the underlying codebase. Furthermore, there is a distinction in the burden and costs associated with deployment and maintenance between platforms and services. With platforms, the responsibility of deployment and maintenance lies with the researchers, entailing additional efforts and costs. In contrast, services assume these burdens on behalf of the researchers, sharing the costs across different organizations utilizing the service.

Figure 2.1 provides a comprehensive overview of the key steps involved in conducting a mHealth study and illustrates how platforms and services can aid researchers in each stage of the process. Notably, the figure highlights the advantage of services in facilitating the deployment phase, specifically in building the mHealth system infrastructure. Conversely, services may have limitations in terms of personalization and adaptability, which are addressed by platforms during the system development stage. Table 2.1 presents a summary of the distinctions between state-of-the-art mHealth solutions, focusing on three key aspects: customization, cost, and setup time.

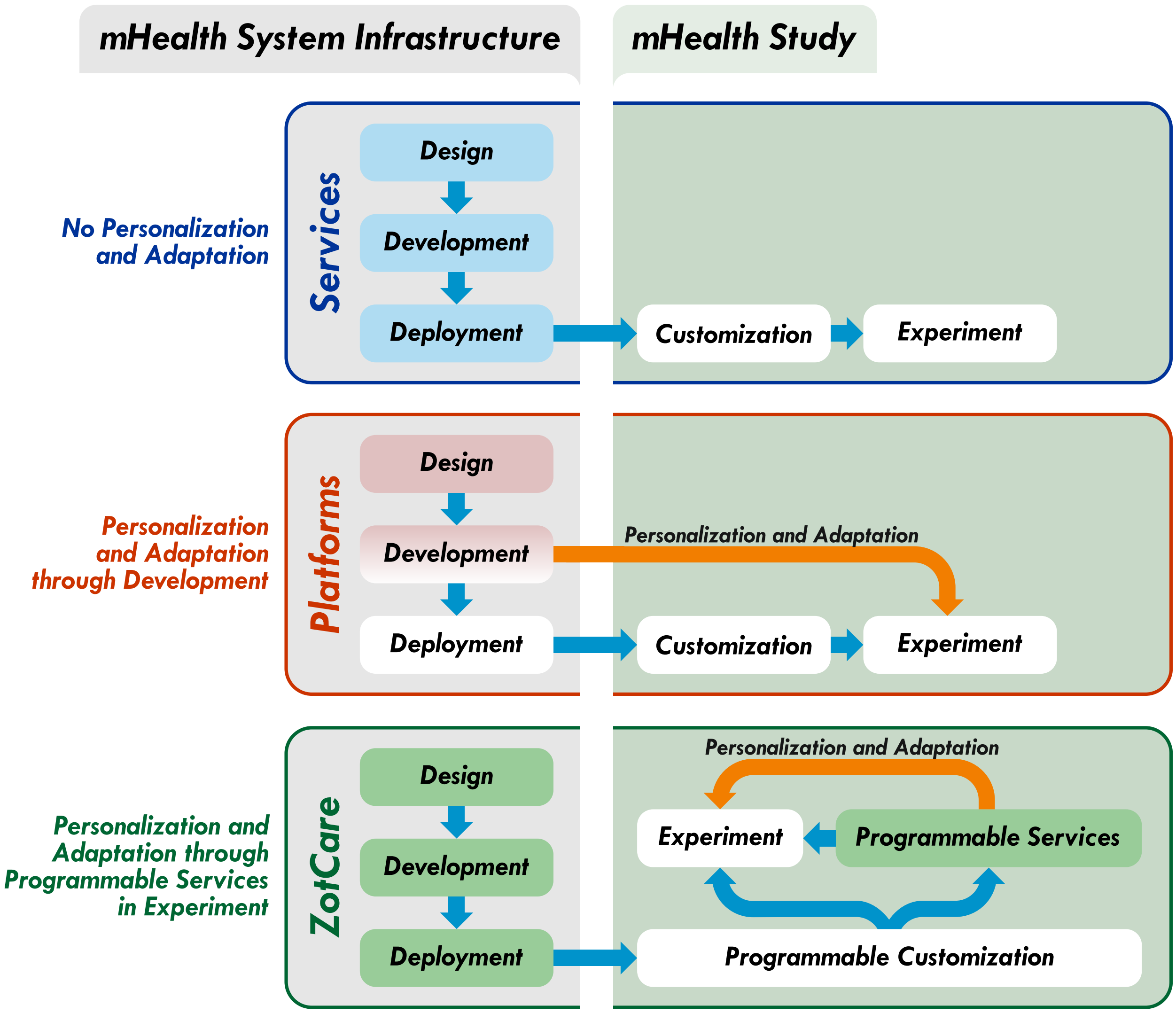



Figure 2.1: mHealth solutions comparison

Our primary objective is to introduce ZotCare, a comprehensive programmable service orchestration, that combines the advantages of both platforms and services while remaining within the services category. ZotCare is specifically designed to operate within a shared environment, accommodating multiple organizations, studies, and researchers. This shared environment facilitates reduced setup time and costs compared to traditional mHealth platforms.

ZotCare offers extensive customization options across various levels, including development, configurability, and programmability. These customization capabilities allow for seamless implementation of new features and functionalities tailored to specific research needs. Notably, ZotCare excels at the programmability level, providing researchers with a diverse set of tools to achieve the personalization and adaptation required in modern mHealth studies. Figure 2.1 illustrates how researchers can leverage ZotCare's programmable services to attain personalized and adaptive features within their experiments, eliminating the need for additional development efforts. At the development level, researchers can utilize the open-source version of ZotCare, similar to existing platforms, enabling independent deployment and utilization. Table 2.1 summarizes the distinctions between ZotCare and other commonly used platforms and services in the field of mHealth studies. In the subsequent section, we will delve into a detailed discussion of ZotCare's capabilities.

Table 2.1: mHealth solutions summary

| | | Customization level | | | | |
|---|---|---|---|---|---|---|
| | | Development | Configurability | Programmability | Cost | Setup time |
| **Platforms** | mCerebrum | yes | no | no | exclusive | high |
| | Radar-base | yes | no | no | exclusive | high |
| | Bridge | yes | no | no | exclusive | high |
| **Services** | ilumivu | no | yes | no | shared | low |
| | Ethica | no | yes | no | shared | low |
| | **ZotCare** | yes | yes | yes | **shared** | **low** |

# Chapter 3

# ZotCare: Service Orchestration

ZotCare constitutes a Health Cybernetics platform, specifically designed to operate as a closed-loop real-time monitoring-intervention system. Its purpose is to cater to the requirements of researchers, clinicians, and community health workers engaged in conducting studies or delivering digital health services. This comprehensive platform enables ubiquitous monitoring of individuals, encompassing both general populations and those at heightened risk, while also providing mHealth interventions. Additionally, it offers a direct avenue for end-users to engage in self-management.

ZotCare encompasses fundamental components essential for conducting mHealth studies across the entire health technology stack. Notably, it provides services that streamline data collection through the utilization of intelligent devices, such as wearables and portable devices. Furthermore, it enables bidirectional interactions between study participants and researchers through gateway devices, including smartphones. Augmenting its capabilities, ZotCare's cloud services provide data analysis and visualization, facilitate the construction and execution of real-time predictive models, and initiate actions necessary to enable just-in-time adaptive interventions (JITAI).

The primary aim of ZotCare is to enable the expeditious and convenient advancement of mHealth solutions catering to users possessing varying degrees of programming and engineering expertise, irrespective of their level of technological literacy. Consequently, through utilizing ZotCare services, researchers can efficiently diminish the time and expenses associated with the implementation and deployment of monitoring systems, enabling them to focus their endeavors on study design, conceptualization, and participant engagement.

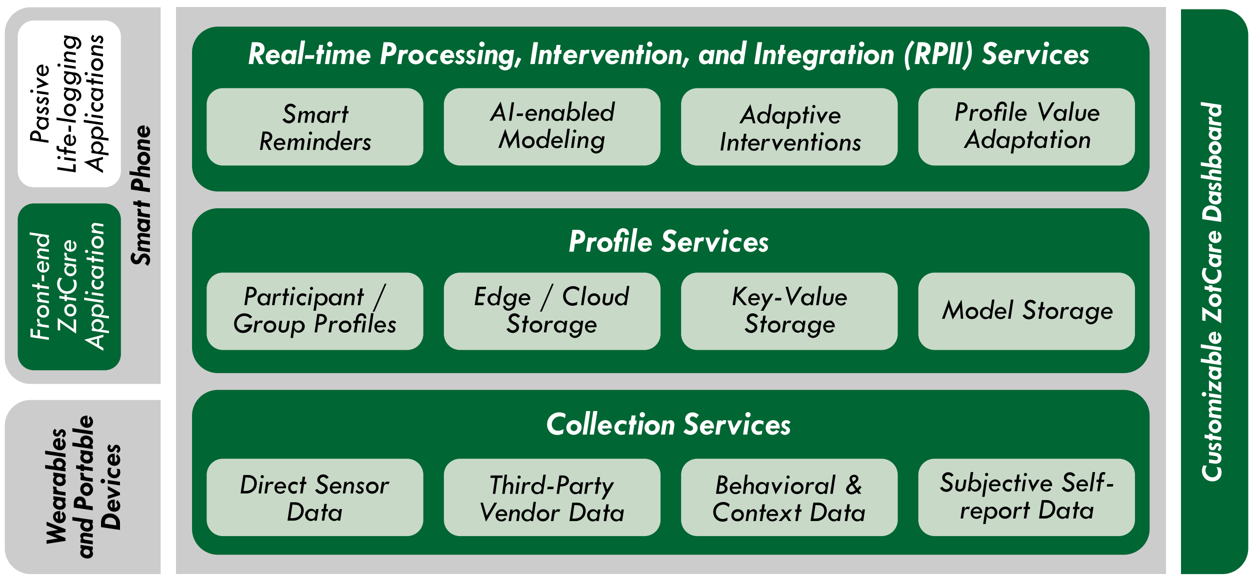



Figure 3.1: ZotCare services overview

Figure 3.1 illustrates a comprehensive overview of ZotCare services and interfaces. The *Data Collection Services* facilitate the ingestion of data from diverse devices, applications, and services. Once collected, the data undergoes processing and is stored as a continuous stream within ZotCare. The *Profile Services* assume responsibility for the storage and processing of data in the form of key-value pairs. This storage mechanism enables the creation of profiles for participants and groups, serving as a repository for personalized study-related data and models, as further elucidated subsequently. Through the *Real-time Processing, Intervention, and Integration (RPII) Services*, researchers possess the capability to incorporate adaptive, intelligent, and real-time components into their studies. These components are capable of triggering various actions based on the data obtained from the *Profile* and *Collection* services. In conjunction with these services, ZotCare provides two interfaces: a customizable dashboard and a user-facing mobile application. The customizable ZotCare dashboard serves as a web application, offering researchers an interface for accessing and modifying ZotCare services pertinent to their respective studies. Researchers can employ the dashboard to manage collected data, recruit participants, and customize it for clinical purposes if desired. The ZotCare application, on the other hand, functions as a user-facing mobile application for participants. It allows them to interact with ZotCare services, enabling functionalities such as receiving reminders, engaging in ecological momentary assessments (EMAs), and benefiting from adaptive mobile health interventions. Moreover, ZotCare facilitates the integration of contextual and behavioral monitoring applications, commonly referred to as lifelogging applications. The subsequent sections delve into further details regarding ZotCare services and provide insights into how researchers can effectively leverage these services to construct their closed-loop mHealth solutions.

## 3.1 Collection Services

The *Collection Services* assume the responsibility of acquiring and integrating participants' data within ZotCare. Given the multifaceted nature of mHealth studies, various types of data are typically employed. Objective physiological, behavioral, and contextual data, alongside subjective self-reported data, constitute the principal data types utilized in the context of mHealth studies. Furthermore, third-party vendors and applications offer diverse methodologies for data collection, encompassing direct sensor readings as well as indirect data acquisition through their server-side APIs. To accommodate these disparate data types and collection methods, ZotCare incorporates a range of features that enable the acquisition of data through diverse channels, subsequently presenting them to researchers in a cohesive and standardized format.

The Collection Service possesses the capability to gather physiological data from prominent fitness and well-being devices. These devices encompass wearable options, such as smartwatches and rings, and portable devices, like smart blood pressure monitors and scales. These devices are capable of providing physiological data in processed formats, and in some cases, as raw data. The raw data typically comprises inertial measurements (accelerometer and gyroscope), photoplethysmography (PPG), electrocardiogram (ECG), air pressure, luminosity sensor data, and other sensor readings, contingent upon the specific type and model of the device. On the other hand, processed data generally entails higher-level derived physiological metrics such as heart rate, heart rate variability, sleep quality, steps, exercise data, weight, and other relevant parameters. These metrics are derived from the raw sensor readings by the respective vendors.

To facilitate the collection of such data, ZotCare has been integrated with various healthcare device vendors. Presently, ZotCare offers support for Samsung, Garmin, Empatica, and Fitbit smartwatches, as well as Oura rings for smart wearables. Additionally, ZotCare can integrate with Withings smart scales and blood pressure monitors. It is important to note that the list of supported devices is continually expanding, as indicated in Table 3.1. For certain devices that provide a software development kit (SDK) and open access to their operating system/firmware (e.g., Samsung Active watches running Tizen OS), ZotCare offers a native smartwatch application. This application enables direct access to the raw signals from these devices and transmits them to the ZotCare back-end. Researchers also have the flexibility to incorporate new devices through direct connections or by utilizing third-party services, utilizing standard open authentication (OAuth) methods.

Furthermore, to augment data collection capabilities within ZotCare, we have seamlessly integrated the AWARE smartphone-based logging framework [4] to enable the passive collection of behavioral and contextual data. Through AWARE, researchers can leverage participants' smartphones to gather data from various sensors, including location, accelerometer, battery status, light intensity, temperature, and more. AWARE also allows the extraction of contextual information from participants' daily lives, such as screen lock/unlock events, application usage patterns, step count, and even communication activities such as notifications, text messages, and phone calls.

To encompass the collection of self-reported subjective data within ZotCare, we have incorporated an *Interaction* sub-service into the system. This feature empowers researchers to design and deploy dynamic questionnaires, indicators, and interactive tasks using the Interaction's functionality. The ZotCare front-end application effectively handles these Interactions, capturing participants' responses along with detailed metadata for comprehensive analytics. Moreover, the Interactions feature serves as a versatile tool for various purposes, including EMAs, information delivery, assessments, recommendations, and interventions. Researchers have the flexibility to update questions, EMAs, and other interactive components on-the-fly using the ZotCare dashboard, granting them dynamic control over the study's data collection processes.

Table 3.1: ZotCare objective data collection

| **Device** | **Data type** | **Integration type** | **Dev stage** |
|---|---|---|---|
| Samsung Tizen watches | raw/processed | direct | supported |
| WearOS-enabled watches | raw/processed | direct | under-dev |
| Empatica E4 wristband | raw/processed | direct/third-party | under-dev |
| Garmin | processed | third-party | supported |
| Whitings devices (BP, Scale, etc.) | processed | third-party | supported |
| Fitbit | processed | third-party | under-dev |
| Oura | processed | third-party | supported |
| AWARE | raw/processed | direct | supported |

## 3.2 Profile Services

The Profile Services within ZotCare assume the responsibility of storing specific information pertaining to groups or individual participants. Researchers can program these profiles to establish key-value storage for data management purposes. In the case of participant profiles, the programmed key-value storage consists of a predetermined set of keys established by the researchers for all participants. However, individual values can be stored per key for each participant, allowing for personalized data storage. For group profiles, a single value is associated with each key, which can be replicated across different groups. This replication enables the creation of distinct groups, such as control and intervention groups, or allows for customization of shared resources, such as the ZotCare Frontend application. Each key within the profiles can be configured with a variety of features. Researchers have the flexibility to choose whether the values associated with these keys should be stored on participants' edge devices or in the cloud. Additionally, researchers can determine whether these values should be visible to the participants, depending on the study's specific requirements and privacy considerations.

The Profile Services play a crucial role in enabling researchers to personalize and adapt their studies over time, particularly in advanced studies that require participant engagement, personalized interactions, or the utilization of statistical or AI models. However, studies that primarily focus on monitoring and passive data collection may not extensively utilize this service. Within participants' profiles, researchers can store a range of important dates and times, such as join date, delivery date, significant personal events, and preferred notification times. Additionally, characteristics such as height, weight, and fitness level can be recorded. Serializable entities, such as personal AI models or statistical models, as well as files like images, audio recordings, or voice recordings, can also be stored within participants' profiles.

Group profiles, on the other hand, contain information that is shared among the members of a specific group. This may include timing information for different stages of the study or shared AI models. Furthermore, group profiles can include customization data specific to each study, such as differentiating between intervention and control groups or specifying menu items. The information stored within profiles serves multiple purposes within the

*Real-time Processing, Intervention, and Integration Services*. It allows for the adaptation of study procedures based on individual participant characteristics. Researchers can also leverage profile information within Interactions to customize and personalize the individual experiences of participants. Furthermore, profiles can be used to locally store personal identifiers such as names, addresses, and photos, instead of saving them on servers. This enables further customization of the participant's experience while preserving their privacy.

Overall, the Profile service provides researchers with a versatile tool for personalization, adaptation, and customization, enhancing the effectiveness and participant-centric nature of their studies.

## 3.3 Real-time Processing, Intervention, and Integration (RPII) Services

ZotCare offers researchers a comprehensive suite of *Real-time Processing, Intervention, and Integration (RPII) Services*, which equip them with the capability to transform data into knowledge, incorporate intelligence into their studies, and effectively close the loop within their solutions.

Through the RPII Services, researchers gain the ability to process data derived from the Profile and Collection services, enabling them to extract meaningful insights and execute subsequent actions based on the processed data. These services can be leveraged at various stages of the data processing pipeline, encompassing tasks such as data pre-processing, AI model development, collection of smart labels and EMAs, scheduling adaptive interventions, and sending intelligent reminders.

By utilizing the RPII Services, researchers are empowered with complete control over the flow of data within their studies. This enables them to dynamically analyze and respond to data in real-time, facilitating the integration of intelligence into their research and ultimately closing the loop within the solution they have developed.

Within ZotCare, each study is capable of containing multiple Real-time Processing, Intervention, and Integration (RPII) instances, which play a pivotal role in enabling dynamic and intelligent functionality. Each RPII instance consists of three essential components: ***Triggers***, ***Conditions***, and ***Actions***. *Triggers* serve as indicators that determine when an RPII unit is to be executed. These triggers can be categorized as either data-driven, responding to incoming new data, or chronological, based on fixed times or frequencies. *Conditions*, on the other hand, evaluate the data to determine if any adaptations or actions need to be performed. Based on the specified conditions and the available data, the RPII instance can make informed decisions regarding the subsequent actions. *Actions* within an RPII instance are programmable functions that can trigger internal modifications within the ZotCare environment or invoke external functionalities. Researchers have the flexibility to program RPII instances with various internal functions within ZotCare, including data fetching, participant grouping and filtering, data processing, AI model building, and writing to data streams or profile values. Furthermore, ZotCare supports external actions such as sending emails, push notifications to the ZotCare mobile application, and accessing external resources. To provide an overview of these features, a comprehensive summary is presented in Table 3.2, which outlines the various logic features supported by ZotCare.

Moreover, ZotCare offers seamless integration options for external systems with its RPII services. Researchers are provided with dedicated endpoints to access ZotCare from their own machines and servers, facilitating the integration of external resources into the ZotCare environment. To streamline the process of utilizing ZotCare externally, an SDK is available, designed to simplify the interaction with ZotCare and offer additional features. The SDK enables researchers to fetch, cache, and process data, as well as invoke actions within

Table 3.2: Zotcare RPII services features

| **Component** | **Type** | **Options** | **Dev stage** |
|---|---|---|---|
| Triggers | Data | incoming data | supported |
| | Chronological | cron expressions | supported |
| Conditions | Fetch | data streams & profiles | supported |
| | Filter | data streams & profiles | supported |
| | If / Else | - | supported |
| | Inferring AI models | - | under-dev |
| Actions | Send Email | templates & plain | supported |
| | Send Push Notification | ZotCare & Firebase [5] & OneSignal [24] | supported |
| | Write Profile | - | supported |
| | Training AI models | - | under-dev |

ZotCare, all without the need to handle complex authentications or intricate API calls. By leveraging these integration capabilities, researchers can utilize their own resources to replace or supplement ZotCare's RPII components, enhancing the flexibility and adaptability of the system to suit their specific requirements.

## 3.4 The Customizable ZotCare Dashboard

The ZotCare dashboard serves as a customizable interface that facilitates interaction between users (such as researchers and clinicians) and ZotCare services. Researchers can create different study groups through the dashboard. Each group can be configured to utilize the ZotCare services for the purpose of that specific research or product.

The ZotCare dashboard incorporates a dedicated section for user management. Within this section, researchers can recruit new participants for their studies. This can be achieved through the utilization of random IDs for direct recruitment or by utilizing sign-up links for anonymous recruitment. Additionally, the dashboard enables researchers to edit user information and profile values as needed. Furthermore, the ZotCare dashboard offers a comprehensive suite of data analysis capabilities, ensuring that researchers have the necessary tools to derive valuable insights from their research data. Researchers can leverage the provided tools to visualize data in its original format or apply sophisticated aggregation and filtering techniques to create visually informative charts and graphs. Moreover, the dashboard empowers researchers to employ their domain knowledge and expertise by facilitating direct annotation of data within the platform. These annotations are seamlessly stored as new data streams within ZotCare, contributing to a rich and comprehensive dataset for further analysis.

In addition to user management and data analysis functionalities, the ZotCare dashboard provides researchers with effective tools for managing their services within the platform. Through an intuitive interface, researchers can easily activate, modify, or review the configurations of their services. While certain services, such as collection services, entail straightforward setup steps, others, such as programmable services like RPII, profile, and interactions services, necessitate more advanced configurations. To streamline this process, the dashboard offers interactive editors that facilitate researchers in editing, debugging, and testing these programmable services, ensuring a seamless and efficient management experience.

ZotCare also incorporates a fine-grained access control mechanism that allows users to have specific permissions within individual studies. This feature enables researchers to involve different collaborators in their study, assigning them distinct roles based on their access scope. These roles can range from recruiters or data analysts to study managers or clinicians. Each collaborator is granted access only to the relevant parts of the dashboard that align with their assigned role. This stringent access control is crucial for safeguarding the privacy and integrity of the study, ensuring that each collaborator can only view and utilize the components that pertain to their specific responsibilities.

## 3.5 ZotCare Mobile Application

The ZotCare mobile application serves as an interface for facilitating ZotCare services to participants. This mobile app functions as a front-end interface, enabling various services, including mHealth interventions, multimedia interactions, and interactive profiles through its components. Additionally, the ZotCare app acts as an assistant to participants, aiding them in device setup and facilitating communication between participants and researchers/clinicians. The primary purpose of the app is to provide participants with interactive "interactions." These interactions encompass a range of components, such as multiple-choice, numerical, time, data, and text input, as well as sliders, among others. These components are well-suited for various purposes, such as EMAs, questionnaires, and data labeling, which are commonly employed in mHealth studies. Furthermore, interactions are equipped with multimedia features, including videos, images, audio, and audio-video recorders. Extensive research has demonstrated the effectiveness of these multimedia tools for both assessment and mHealth interventions, as evidenced by studies presented in Chapter 5.

Moreover, interactions can incorporate customized components that researchers can create and incorporate, allowing for further customization and enhancement of their studies. Previous studies using ZotCare have showcased the utilization of such components for interventions, such as interactive breathing exercises, mindfulness-oriented image galleries, relational savoring exercises, and educational materials. Additionally, these components have been employed in assessments, such as cognitive games (e.g., finger tapping, word pair memory tests, rule-switching games, etc.). In addition to the visible components, interactions can include condition statements, representation configurations, variables, and metadata. These features provide researchers with a broader set of tools for personalization and customization.

Furthermore, participants have the ability to grant authorization to ZotCare, via the application, to access their health data from third-party services, such as Oura and Garmin. This integration allows for seamless retrieval of pertinent health information. Additionally, the app offers comprehensive instructions and troubleshooting steps for devices and applications that establish a direct connection with ZotCare, including Samsung and AWARE. Furthermore, participants can access certain features of the Collection and Profile services through the ZotCare app. These services provide participants with valuable functionalities and data management capabilities. The ZotCare app serves as a means for researchers and participants to maintain a continuous connection. This connection is facilitated through various means, including reminders, notifications, and messages. Researchers can choose to automate these communications through the RPII services or manually trigger them using the dashboard.

A general version of the ZotCare app is readily available for installation and use on Android and iOS smartphones. However, the app's flexibility allows for customization to accommodate different research studies. Researchers possess the capability to modify the app's colors, logos, menus, and other visual aspects to align with the specific requirements of their study. Moreover, they can also modify the app's components to create tailored "Interactions" with additional functionalities. By leveraging the Profiles feature, researchers can further personalize the app's appearance to suit individual studies or specific participant requirements.

# Chapter 4

# ZotCare: Technical Challenges and Implementation of a mHealth Service Data Pipeline

Besides the features that ZotCare provides directly to the researchers, the system must meet specific technical criteria to be useful for mHealth studies. The end-to-end latency of the processing pipeline needs to be reliable and scalable. This pipeline starts by receiving the data through a web endpoint, storing the data, running the RPII instance associated with the data, and invoking actions based on the result. Since ZotCare's collection service can process high-frequency raw sensor data, processing and storing the data can introduce new challenges to the scale of the system.

The first challenge is how the system would **scale compared to the number of participants**. As the number of participants increases, the time it takes to store the data or process the RPII instance would either stay the same or increase as well. This causes the upload requests from the participants to queue in the web server. In other applications, the increase in queue time might not be a big issue, however, for limited resources in wearable devices, this time can increase the device's awake time and energy consumption, and ultimately cause the requests to fail or drop out.

The second challenge is how the system would **respond as the amount of data in the storage increases**. High-frequency raw data can take up a high volume of storage space and since the data is constantly generated by wearable devices, it grows faster than conventional data. Since it is essential in mHealth applications to find these data efficiently, there is a need to index the raw data. However, a big index tree causes both read and write requests to slow down. And if the index tree grows bigger than the memory limit of the database it can cause additional failure and slow down to read and write processes. This will slow down both the writing operations of the data collection service and reading operations of the RPII service and ultimately slow down the end-to-end latency of the system.

The third challenge in the ZotCare data processing pipeline is the systems **quality of service against different RPII process tasks**. Some RPII tasks might need access to many data points such as downloading data or retraining a machine learning model. However, some other tasks can be quick but have a short deadline such as inference tasks. However, in a system that can process all these tasks homogeneously, this can cause the longer tasks with lower priority to block the resources for shorter tasks that have higher priorities.

## 4.1 Method

To introduce ZotCare's data's data pipeline architecture, This section will start with a conventional API server design architecture. At each step of the process, we will explain the problems with the architecture and how it can be improved considering the characteristics of the mHealth data and application.

### 4.1.1 Conventional Blocking Approach

The conventional approach to designing a server architecture is to design a typical web server to receive the data, store the information, run the necessary processes, and return the result to the participant. The request for storing the data will occupy a connection in the API server for the duration of storage and processing of the information. The use of this approach is common in general use web-servers and is supported by most web-server platforms.

The benefit of this approach is that the requests are processed immediately and the end-to-end response time of the data processing pipeline is short. However, this approach queues the requests at the API level and keeps the connection with the uploading device alive. As the storage time can potentially be long and the RPII instance process can take a random time, these queues can be long and accumulate over time. Besides the mHealth devices that upload the data are usually performed on battery and the wait time for requests can consume the limited energy.

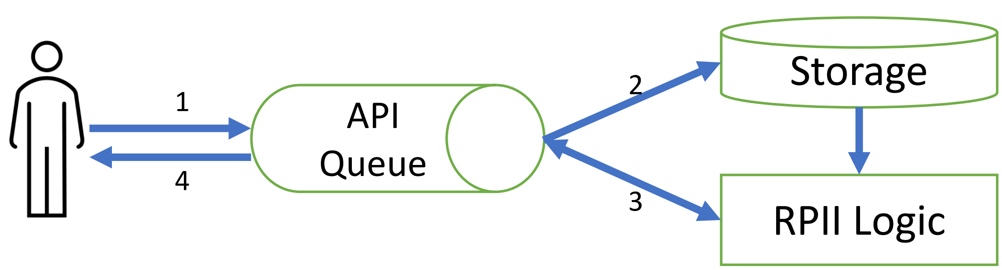



Figure 4.1: Conventional data pipeline

### 4.1.2 Double queue approach

To improve the conventional approach, we can consider limitations and opportunities that exist in mHealth systems and change the system's design based on them.

1. The *RPII actions are often asynchronous.*

RPII actions often fall into categories such as push notifications, email reminders, collecting more data, etc. These actions need to go through other Internet services and be delivered asynchronously. Besides that, the result of these actions depends on the human response time and availability which makes a couple of seconds of delay negligible.

2. Upload requests are initiated by *low-power wearables and gateway* devices.

The devices that send the upload requests are often initiated wearables and gateway devices such as smartphones, smartwatches, smart home devices, etc. These devices are low-power and efficient and usually have a short timeout on web requests. This limits the response time for the conventional method.

Using an additional queue after the API queue can internalize the wait time of the requests. The first queue of API accepts the data, stores them temporarily, closes the connection, and creates a new task in a broker queue. There are two workers on the broker queue: a storage worker and an RPII worker. The storage worker gets the data from the temporary storage stores the data in an indexed database, and creates a new task in the broker queue for the RPII worker to process the data. The RPII worker can read the data from the storage, process the data, and invoke the actions asynchronously to the participant or researchers. Since the actions are asynchronous, they can be initiated even after the request connection is closed.

In this approach, the queue for storing and processing the data happens internally in the server and can be done with more resources and more efficiently compared to low-power wearable and edge devices. This causes the wearable devices to get off the API queue as soon as possible, save power, and prevent the wearable requests from timing out. However, this approach comes with the cost of losing access to synchronous actions, and needing temporary storage that can delay the end-to-end latency of the system. Even though this approach helps with the scale of the system to the load of the participants, it does not affect the challenges of the storage scale.

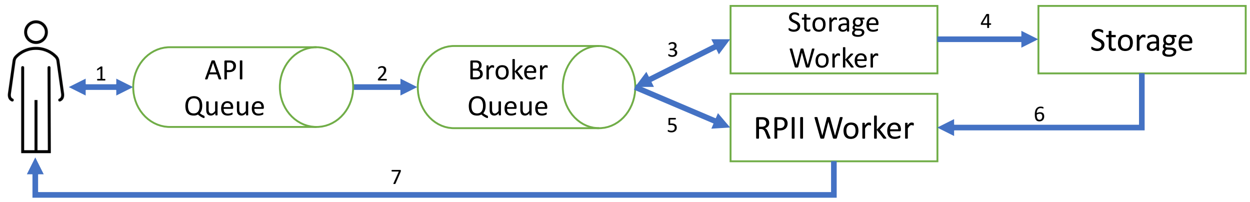



Figure 4.2: Double queue pipeline

### 4.1.3 Double queue with watcher approach

The nature of mHealth studies creates distinct characteristics in storing and accessing data streams that can be used to improve the double queue approach further.

1. The chance of receiving a data decreases with its age

Even though the mHealth data can come out of order, the chances of receiving older data decrease with the age of that data. This is because older data often gets priority in synchronization, and it becomes less probable for old data not to synchronize while new data is uploaded constantly.

2. Most of the processing focuses the recent data

The RPII instances often perform one of the following tasks: 1- updating a trained model with recent data and labels, 2- running the model on current data to make an inference 3- retraining the entire model with a new hypothesis. These tasks create a pattern of accessing the data since both the update and inference tasks need access to the recent data, and access to the older data only happens as a sequential stream. Besides the RPII access, researchers often want to see the recent data with more detail in the dashboard to create a hypothesis or assess participants' states and only use a summary of the older data to control the involvement of the participants.

This access pattern can create a different data service that keeps the old and recent data in separate storage systems. The new data can be stored in highly-indexed storage optimized for high access granularity and supports out-of-order insertions called a *hot storage*. When the data reaches specific criteria, it can be transferred to an ordered, low-indexed storage system called a *cold storage*. In this approach, the storage worker stores the data in the hot storage the same way as in the double queue approach. However, reading the data has to go through a gateway service that can choose between hot storage and cold storage based on the age of the data. There is also a need for a watcher service in the system to transfer the old data from the hot storage to the cold storage as they reach a specific limit.

The hot storage, cold storage, and storage watcher systems need to be configured according to each data stream and application. For instance, consider the case where the hot storage is indexing the data based on minutes, and the cold storage is indexing the data based on days. This will cause the index tree in the cold storage to be 1440 times smaller than the hot storage, while the read time from cold-storage will only increase a constant amount. Choosing the right policy for storage watcher can also affect the scale of the system. for instance, using only age as the policy can cause high-usage summary data to be retired while they are accessed with high frequency, and they do not occupy a big portion of the index tree might cause the system to slow down. However, a fusion of age, demand, and index size can create an optimum policy for the storage watcher.

This architecture scales well with the number of participants while it has the benefit of not slowing down as the amount of stored data goes up. However, accessing the older data from cold storage can be slower, but as discussed, accessing the old data happens less frequently than the recent data.

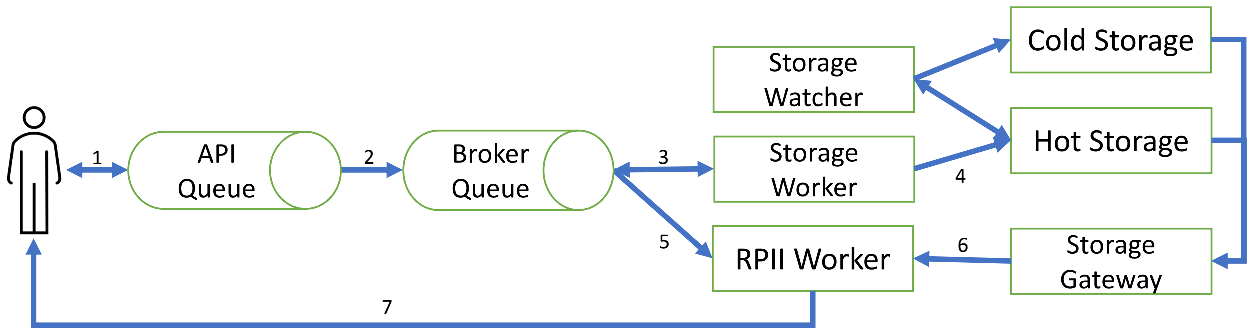



Figure 4.3: Douple queue with storage watcher pipeline

### 4.1.4 Double Queue with watcher and worker manager

As discussed in section 4.1.3, the processes in the RPII worker often follow a similar access pattern.

1. Processes that access the recent data and have a short deadline: inference, updating models

2. Processes that access all the data and have a longer deadline: model retrain, download data

Since the RPII instances are programmed by the users, the system doesn't have information on which category of process is running. However, the data access pattern of each of these processes and their run time can determine that.

This information can be used to create a worker manager wrapper around the workers. The work manager can monitor the access of the worker to the data and its run time. In the cases that the worker is taking more than the limit on time or requesting the amount of data bigger than the limit, the manager can halt the worker after processing some of the data and put it in an inactive state. The manager then can resume the work later to allow the shorter processes to go through.

The limits on time, the limit on the amount of data, and the inactive time between executions can be set based on experiments or can be set dynamically by monitoring the load in the broker queue and the total amount of requested data. Finding the right policies for the worker manager can change the performance of the system and the maximum end-to-end latency of the data pipeline.

This approach can enhance the performance of the system on shorter processes. However, it can cause the longer processes to take longer. Another potential downside of the system is that a bad policy for the worker manager can create starvation for the longer processes and cause them to accumulate and take more resources. One of the notable challenges of this approach is that it limits the RPII worker's actions. Some of the data and processes in the RPII worker (i.e., multi-task processing) might not be serializable, or pausing and restoring the state of the worker might have a big overhead.

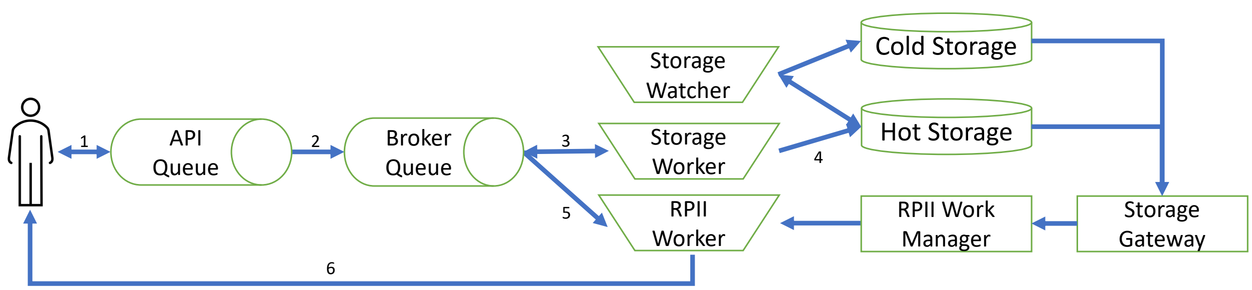



Figure 4.4: Double queue with watcher and work manager method

## 4.2 Experiments and Results

ZotCare has been implemented using a Python Flask web server. Flask is a quick and easy Python framework for server development. Since it is a Python framework it makes it easier to integrate with the commonly used machine learning and data science libraries. The hot storage used in ZotCare is using MongoDB. MongoDB makes it easy to create complex indices for the data and have different data with different formats merged in a single tree. This makes MongoDB a good option for ZotCare since the studies can have a dynamic access pattern and different shapes and formats. RabbitMQ and Celery were utilized to implement the broker queue and workers. Both of these tools are the most commonly used task distribution and broker services in Python and are scalable and distributable as the system is facing more load.

The experiments were done on an emulated server environment using docker-compose. Docker-compose is a software tool that is capable of creating a cluster-like environment within a host and creating different instances of the components in that environment. It provides features for creating networks and volumes to connect these components together. Docker-compose also allows its users to create limitations on processing and memory specifications for each of these components which makes it a good candidate for our experiments. The total processing power of each approach was balanced to the same total to eliminate the effect of adding more processing units for workers and brokers in different approaches.

### 4.2.1 Scalability with the number of participants

This experiment compares the conventional approach and the double queue approach in handling the number of participants who are actively uploading data. This experiment can show how effective the double queue approach is in providing a quality of service for the end-to-end latency of the system and its ability to lower the API wait time. In this experiment, we assumed that the participants would respawn over a period of three minutes with unified distribution. First, each participant will log in to the system, and then upload 2400 data points every three minutes. We changed the number of participants, reverted the database, and ran each of the experiments for a total time of 15 minutes. It was assumed that the RPII worker task takes about 2 seconds in total. This number is based on research by Tazarv et al. [31]. Figure 4.5 shows the end-to-end latency of the conventional method vs the latency of the double stack method split between the API processing time and the rest of the processing time consumed by the storage and RPII workers. As shown in Figure 4.5 the API response time for the double stack method is almost negligible against the end-to-end latency. As observed, the conventional method is faster than the double stack method by less than 5% for 500 and 1000 users but the double stack method surpasses the conventional method by more than 75% at 2500 participants and the gap will only grow after that.

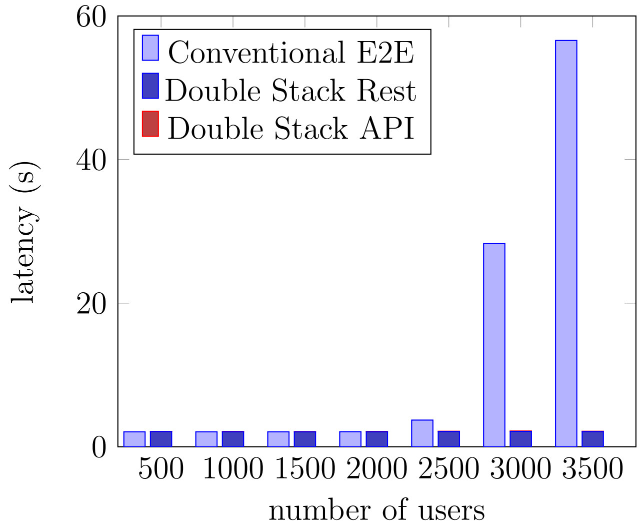



Figure 4.5: System latency scale with the number of participants

# Chapter 5

# ZotCare: Use Cases

ZotCare has been utilized as a service within diverse mHealth research studies. The functionalities of ZotCare were devised to meet the requirements of these studies and were adapted accordingly based on their specific utilization. The initial studies availed themselves of preliminary versions of ZotCare, encompassing provisions for multi-modal data collection. Subsequently, ZotCare broadened its spectrum of services and characteristics to address the requirements for customization and governance. In the following, we will begin by providing an overview of select studies that used ZotCare services for purposes encompassing data collection, data modeling, and intervention. Subsequently, we will delineate the challenges encountered and describe the integration of ZotCare into these aforementioned studies.

## 5.1 Personal Mental Health Navigation Project

The Mental Health Navigation (MHN) project develops a proactive, personalized approach to monitor, estimate, and guide individuals toward their desirable mental health state [26]. MHN monitors a multimodal stream of objective and subjective information to build inference models to determine participants' mental states, context, and lifestyle. Using the constructed personal model and the current state, a navigator system can steer the participants using interventions at each step. The MHN project comprised two studies, the *Affect* study and the *Loneliness* study. The *Affect* study focused on investigating the connection between college students' psychophysiological signals and sleep on their mood [13]. Due to the onset of the COVID-19 pandemic during the midst of the *Affect* study, a revision was made to expand the study's objectives to encompass the impact of COVID-19 and subsequent lockdown measures on the lives and emotional well-being of college students [22, 16]. The subsequent phase of the MHN project referred to as the *Loneliness* study, was primarily dedicated to the real-time evaluation of the mental well-being of college students, along with the provision of just-in-time adaptive interventions for those individuals requiring support [34]. Moreover, the *Loneliness* study encompassed the collection of life-logging and contextual data, enabling the inference of participants' virtual (through smartphones) and physical communication levels. By integrating the acquired life-logging and contextual data with the pre-existing models established in the *Affect* study, the accuracy of the loneliness assessment models was significantly enhanced. Consequently, the Loneliness study successfully developed adaptive interventions, leveraging these refined models, with the ultimate aim of mitigating the adverse mental state experienced by the participants.

Both the *Affect* and *Loneliness* studies employed the utilization of ZotCare as a means to gather bio-signal and EMA data and administer mHealth interventions. The *Affect* study specifically utilized the Samsung smartwatch and Oura ring to continuously capture physiological signals, monitor sleep quality, and track levels of physical activity. In addition to these devices, the *Loneliness* study incorporated the use of AWARE to gather life-logging data. Furthermore, a customized ZotCare mobile application, namely mSavorUs (Figure 5.2 A), was developed and employed for the participants. This tailored application not only retained all the features of the original ZotCare application but also provided supplementary custom features for relational savoring exercises and interventions. The interventions within the mSavorUs application were personalized by incorporating participants' names and photos sourced from their memories, utilizing locally stored profiles. To implement the interventions in the *Loneliness* study, the RPII services were utilized. These services were employed both directly for delivering conditional notifications and via an API form to establish a connection with a machine learning agent maintained within a separate cluster. The volume of data collected in each of these studies exceeded 200 gigabytes, with a total of 4.5K and 14.9K labels collected for the *Affect* and *Loneliness* studies, respectively.

Table 5.1: MHN project studies summary

| **Study** | **#** | **Duration** | **Collection Service** | **Profile Service** | **RPII Service** |
|---|---|---|---|---|---|
| MHN Affect [35] | 20 | 3-12 months | • Samsung watch<br>• Oura ring<br>• ZotCare app | - | - |
| MHN Loneliness [34] | 20 | 1 year | • Samsung watch<br>• Oura ring<br>• ZotCare app<br>• AWARE | Local (name and memory images) | • Build mood prediction models<br>• Trigger interventions based on mood<br>• Send conditional reminders |

The initial deployment of ZotCare for the MHN *Affect* study presented various challenges that underscored the necessity for additional features or services to address them effectively. Given the study's incorporation of multiple data collection dimensions, including the Samsung smartwatch, Oura ring, AWARE framework, and questionnaires, study coordinators faced difficulties in monitoring the status of all dimensions and promptly informing participants about any potential issues and interruptions in data collections. Interruptions occasionally occurred in the ZotCare collection services running on participants' devices due to factors such as high battery consumption, device inactivity, or inadvertent shutdowns by participants. To overcome this challenge, ZotCare implemented a summary report generator within the data collection service. These daily summaries could be utilized by researchers through ZotCare dashboards to assess the status of participant data and facilitate timely follow-up when necessary. Additionally, interactive troubleshooting components were integrated into ZotCare Interactions, accessible via the ZotCare mobile application. These troubleshooting components systematically analyzed the participants' collected data and provided step-by-step instructions for resolving any data communication issues that may have arisen. Through the implementation of these additional features and services, ZotCare effectively tackled the challenges encountered during the MHN *Affect* study. This ensured efficient data monitoring, facilitated effective troubleshooting, and ultimately enhanced participant engagement and study outcomes. Moreover, during the course of the *Loneliness* study, the construction of the real-time inference model necessitated computational resources that surpassed the capacity of the ZotCare backend services available at that time. To address this challenge, the researchers leveraged the RPII capability of ZotCare, enabling integration with external resources. The team successfully employed the RPII service to execute an inference model within a cluster, which was triggered by ZotCare web-hooks. The obtained data was subsequently retrieved through the ZotCare SDK, processed, and intervention scheduling was performed using ZotCare API calls. This strategic approach effectively resolved the issue of resource limitations and facilitated the seamless integration of just-in-time adaptive interventions by harnessing the capabilities of external resources.

## 5.2 The UNITE Project

Smart, Connected, and Coordinated Maternal Care for Underserved Communities (UNITE) is a research project funded by the United States National Science Foundation, with the primary objective of developing innovative technologies to enhance the physical and emotional well-being of underserved pregnant women and their newborns. The UNITE initiative endeavors to revolutionize conventional maternal care practices, which have traditionally been delivered within homes or clinics, by integrating an AI-supported remote monitoring system. The project comprises three distinct phases, each with its specific focus and objectives. The initial phase, known as the “Feasibility” phase, concentrated on assessing the viability of remote maternal health monitoring. This involved investigating the level of engagement exhibited by pregnant women with the technology, taking into account their individual health conditions [14]. The second phase of UNITE encompassed a series of small-scale randomized controlled trials (MicroRCTs), which sought to examine various aspects of maternal well-being, such as stress management or physical training, and assess their impact on pregnancy outcomes [32]. Currently, the project is in its third phase, an ”AI-assisted” study aimed at exploring the efficacy of incorporating AI assistants and nurses into the care loop for mothers. Within this phase, an AI-enabled exercise recommendation system has been deployed alongside the recommendations provided by healthcare providers. This human-in-the-loop mHealth approach has resulted in the development of a personalized, step-by-step recommender system that adapts to the specific pregnancy conditions and physical measures of each individual mother.

The Feasibility phase of the UNITE project primarily focused on the collection of data and subsequent analysis during the post-study phase. Given the vulnerable nature of the study group consisting of pregnant mothers, it was imperative to implement a triage system to swiftly identify and report any potentially risky situations. To fulfill this requirement, the UNITE initiative incorporated ZotCare’s RPII services. By integrating data-driven triggers within the RPII unit, the behavior of participants could be assessed, enabling immediate alerts to be sent to researchers when necessary.

In the second phase of UNITE, known as the MicroRCT phase, the team explored an alternative approach to collecting labels. Instead of adhering to fixed times and frequencies, the possibility of employing smart labels was investigated, aiming to maximize the information obtained while minimizing participant interruptions. By utilizing statistical and active ma-

Table 5.2: UNITE project studies summary

| **Study** | **#** | **Duration** | **Collection Services** | **Profile Services** | **RPII Services** |
|---|---|---|---|---|---|
| UNITE Feasibility Study [14] | 25 | 6-10 months | • Samsung watch<br>• Oura ring<br>• ZotCare app | - | • Triage system for early risk assessment |
| UNITE Stress Detection [32] | 14 | 30 days | • Samsung watch<br>• ZotCare app | - | • Using stat models to trigger EMAs |
| UNITE Stress Detection [31] | 18 | 2 months | • Samsung watch<br>• ZotCare app<br>• AWARE | - | • Using reinforcement learning to trigger EMAs |
| UNITE Exercise Recommender System [20] | 20 | 4 months | • Samsung watch<br>• Oura ring<br>• ZotCare app | • Create a physical profile (height, weight)<br>• Indicators (exercise, set, repetition, duration, intensity)<br>• AI suggestions<br>• Store models | • Triage system for risky behaviors<br>• Exercise recommender engine |

chine learning models within the RPII services, the UNITE team could send notifications to participants, prompting them to provide labels at more opportune times, resulting in improved accuracy and heightened participant engagement [31].

During the AI-assisted recommender system phase, the main challenge revolved around establishing a cohesive loop involving the participants, the AI recommender engine, and health providers. To address this challenge, the Profile and RPII services were leveraged, providing the necessary flexibility. Three sets of profiles were employed: one for participants to input their physical measurements, another for health providers to input their assessments, and a final one for the AI recommender engine to store its recommendations. The RPII service played a crucial role in executing the recommender engine by utilizing the physical measurements and health providers' assessments from the Profile services, as well as participants' bio-signals, progress, and feedback from the Collection services. Figure 5.1 illustrates the adoption of the services within this solution. The process begins with Step (1), where participants input their physical measurements through the UNITE mobile app (Figure 5.2 B). This information is then used by the nurses to recommend the participants' initial exercise regimen. In Step (2), the recommender engine utilized this information to train its models, infer the next set of exercise regimen recommendations, store the recommendations within the participant's profile, and notify the health providers of the ongoing process. The health providers, in Step (3), evaluated the final exercise regimen for each participant based on the suggestions provided by the recommender engine. Steps (2) and (3) continue in a continuous loop throughout the duration of the study. This system effectively assists health providers in processing a significant amount of information and providing frequent assessments.

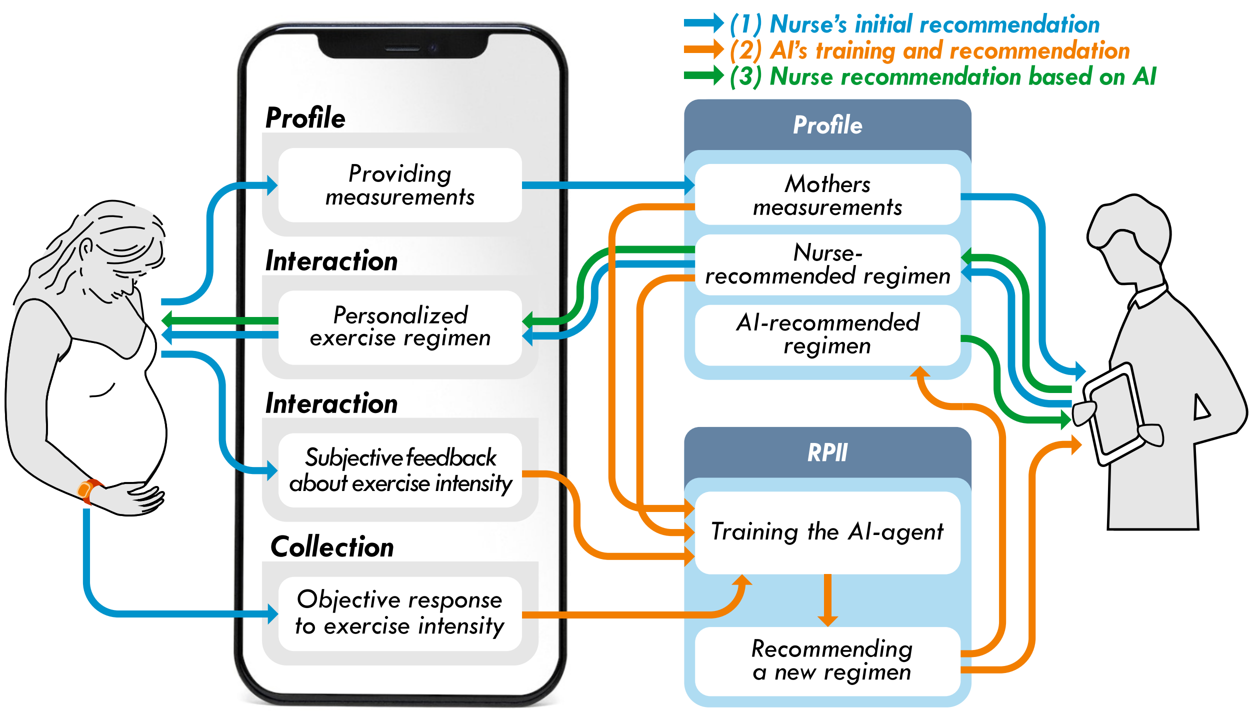



Figure 5.1: UNITE AI-assistant integration in ZotCare

## 5.3 Other Projects

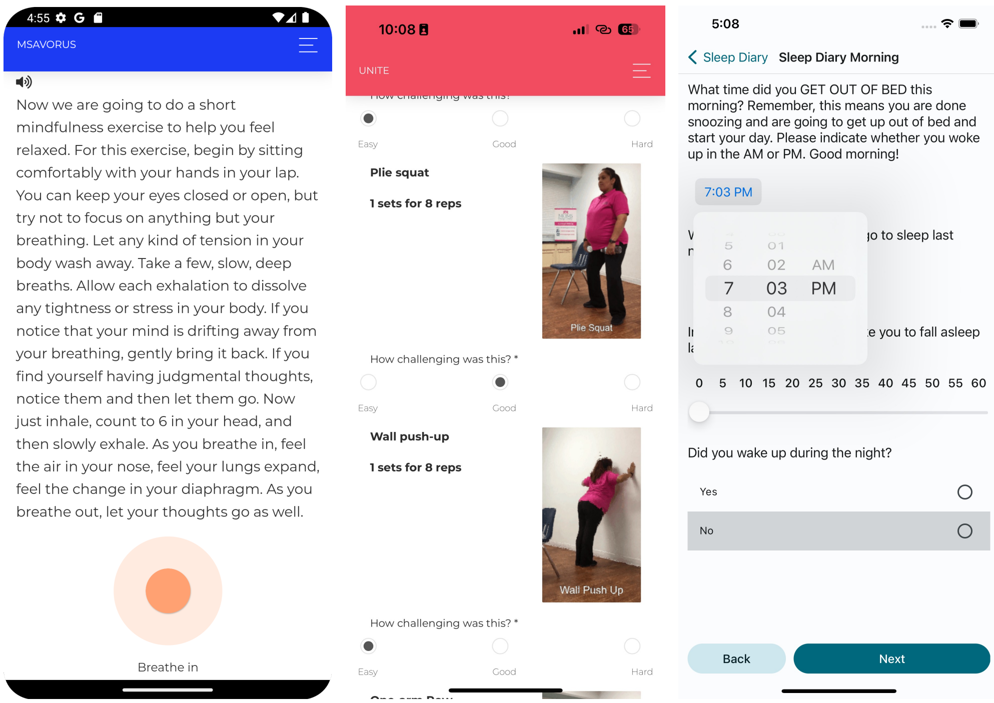
As the services provided by ZotCare continued to develop, they began to be utilized by researchers and universities outside of the original project, encompassing a diverse array of research studies. These studies varied in terms of their contextual settings, languages used, time zones, and specific requirements. The expanding scope of applications enabled ZotCare to adapt and enhance its functionalities, thereby providing researchers with a wider range of features to facilitate their studies effectively. Some example studies are listed in Table 5.3.

Table 5.3: Examples of other studies that utilized ZotCare

| **Study** | **Description** | **#** | **Duration** | **Collection Service** | **Profile Service** | **RPII Service** |
|---|---|---|---|---|---|---|
| Sleep & Menstration [30] | Subjective study on the affect of menstruation on sleep | 20 | 2 months | ZotCare app (customized with cognitive tests) | - | - |
| PREVENT [23] | Daily well-being of pregnant women around COVID-19 pandemic | 38 | 30 days | • Samsung watch<br>• ZotCare app (in Finnish) | - | - |
| D-CCC [28] | D-CCC was proposed to assist community organiztions to monitor elderly | 5 | 2 months | • ZotCare app<br>• Oura ring<br>• Withings Blood Pressure Monitor<br>• Withings Scale | - | • Interventions to promote physical and mental well-being |
| Sleep & Brain study | Impact of sleep quality on memory and cognitive performance | TBD | TBD | • ZotCare app (customized with cognitive tests) | • Custom notification times | • Notification scheduler |
| SERVE OC (ongoing) | Serve OC is focused on preventing high blood pressure and improving health among families | 1000 | 3 Years | • Withings Blood Pressure Monitor | • Family label<br>• Pregnant label<br>• Research Groups | • Missing data detection<br>• Abnormally detection |

Two research projects, namely "Sleep & Menstruation" [30] and "Sleep & Brain," were conducted to investigate subjective sleep assessment and cognitive abilities through the use of questionnaires and cognitive tasks. The first project examined the impact of menstruation on sleep patterns, while the second project explored the relationship between sleep and cognitive abilities. Both projects incorporated traditional questionnaires to gather information on various aspects of sleep, including duration, quality, and mood. Standard question forms components such as multiple-choice, text input, sliders, and time pickers were utilized to ensure comprehensive data collection. Figure 5.2 (C) shows a snapshot of such a questionnaire in the mobile application. Furthermore, interactive cognitive tasks, designed to resemble games, were employed to assess participants' cognitive skills. The flexibility of ZotCare's capabilities in creating customized interaction components proved invaluable for researchers. This allowed them to focus on designing the tasks themselves without the need for developing a separate interactive mobile application. By leveraging ZotCare's functionalities, researchers could streamline the data collection process and efficiently collect data. Given that some questionnaires needed to be administered before bedtime or upon waking up, ZotCare's profile and logic services were utilized to personalize the timing and availability of the questionnaires for each individual participant. This customization ensured that participants received questionnaires at the appropriate times based on their specific sleep schedules and preferences.

In addition to its utilization in the aforementioned studies conducted in the United States, ZotCare has also been employed in research studies conducted in other countries and across different languages. One notable example is the "PREVENT" study conducted in Finland. This study specifically focused on maternal care and aimed to assess the daily well-being of pregnant women during the challenging circumstances imposed by the COVID-19 pandemic [23]. The "PREVENT" study leveraged ZotCare's localization features to adapt the platform to the Finnish language and timezone. This ensured that the study participants in Finland could access and interact with ZotCare's services seamlessly in their native language and




(A) (B) (C)

Figure 5.2: (A) mSavorUs, (B) UNITE, and (C) HowRU app used in MHN, UNITE, and Brain & Sleep projects

within the context of their local time zone.

The Digital Health for the Future of Community-Centered Care (D-CCC) [28] research project aims to explore the integration of technology and community health workers in order to enhance healthcare delivery for underserved communities. Specifically focusing on the elderly population, the project seeks to develop a symbiotic relationship between humans and technology, enabling the design of new technologies that can assist community health workers in providing more effective support. ZotCare played a crucial role in alleviating the burden on participants who were unfamiliar with using advanced technologies. By providing user-friendly interfaces and intuitive interactions, ZotCare facilitated the seamless integration of technology into the participants' lives. Moreover, ZotCare enabled the continuous monitoring of participants' device usage patterns and their overall health status. This feature helped researchers and community health workers gain valuable insights into the participants' well-being and promptly address any emerging issues. The ZotCare dashboard proved to be an invaluable tool in monitoring vital signs trends among participants, such as heart rate and blood pressure. Additionally, the ZotCare application allowed for the subjective capture of daily symptoms, including pain and fatigue, as well as adverse events such as falls. The customization of the ZotCare dashboard specifically catered to the needs of health providers involved in this study. It enabled them to receive intelligent alerts in the event of abnormalities in vital signs or adverse events, and also provided visualizations of the collected data, empowering them to make informed decisions regarding participant care.

The SERVE OC study is a community-focused research initiative spanning three years on a population of 1,000 individuals. With a culturally-tailored approach, the research is facilitated by dedicated community health workers (CHWs). These CHWs play a pivotal role in engaging and guiding families in setting goals, refining skills, and managing risks to fortify their resilience against hypertension. With this approach, this research is seamlessly woven into participants' everyday lifestyles and well-being. SERVE OC uses ZotCare to longitudinally monitor each individual's blood pressure using Whitings Blood Pressure Monitor. It also employs RPII services to process these blood pressure data and detect subtle irregularities for each individual, sending customized notifications to their respective CHWs.

# Chapter 6

# Discussion

The position of the project in terms of its features for researchers has been discussed in Chapter 3 and its technical contributions in building a data pipeline has been brought up in 4. This chapter will be dedicated to exploring some other aspects of the project that are worth mentioning.

## 6.1 Security and Privacy

In order to uphold the integrity of ZotCare's services, it is imperative to prioritize the security and privacy aspects of the platform. Robust security measures have been implemented to safeguard data and communication channels against potential threats posed by unauthorized individuals attempting to manipulate, delete, or disrupt data storage and transmission processes. Privacy considerations within ZotCare are designed to empower participants by granting them control over their personal data. This includes ensuring the protection of identifiable information, thereby safeguarding the privacy of participants. Security and privacy present unique challenges within service-based environments compared to platforms, as multiple organizations share the same resources. Consequently, it becomes necessary to implement measures to safeguard information from both internal and external sources.

Collected data from participants may consist of both objective sensitive data, such as location information and passwords, as well as subjective data that, based on responses provided in Interactions and Profiles, may reveal sensitive information. Similar security and privacy risks exist across other services as well. For instance, programmable services, including RPII services, are susceptible to malware injection. It is worth noting that researchers' mistakes, such as data overwriting, large or repeated queries, and infinite loops, can also introduce malware vulnerabilities.

To address these security challenges, ZotCare has implemented a gateway service that regulates authentication, authorization, and scope through standard encryption methods. This entails a two-step process in which the gateway first verifies the identity of the requester and subsequently checks if the requester has the necessary access permissions to the requested resource. Privacy concerns extend beyond the scope of data collection and storage and begin with participant recruitment. In cases where studies possess knowledge of their participants, researchers can manage deidentification processes on their end, enrolling participants in ZotCare using anonymous IDs. However, for studies that allow individual participant sign-ups, ZotCare can deidentify data associated with participant emails, enabling participants to utilize their emails for password retrieval and receiving notifications. Nevertheless, researchers only have access to anonymous IDs. ZotCare does not currently support deidentification of collected data at this stage. It is important to note that both researchers and users have the option to disable the collection of sensitive data across all ZotCare services, providing an additional layer of privacy control.

## 6.2 Standardization

ZotCare collaborates with various third-party vendors, each offering APIs for data collection and action invocation. These APIs come with unique specifications related to permissions, consent, authentication, and data transfer protocols. While some APIs use standardized protocols like OAuth for permission and authentication or SMTP for email, they often vary in methods and formats, even for similar types of data. For example, accelerometer data from different devices can be returned in distinct metrics and formats.

There has been progress in API protocol standardization, easing the process of integrating diverse vendors into ZotCare. Email protocols and OAuth are good examples where standardization simplifies integration for both ZotCare and researchers. Yet, challenges remain, particularly in achieving universal standardization for all types of data and actions.

Two primary solutions exist to address these challenges. The first involves partnering with intermediary third-party services that offer a unified API by harmonizing protocols from various vendors. This approach, however, might compromise some features available through direct integration. The second is the pursuit of global standardization, requiring all vendors to adhere to stringent protocol and data format standards. While this could streamline integration, it would be resource-intensive and could limit the introduction of new metrics and data types.

For instance, ZotCare's integration with OneSignal [24], a third-party service for sending push notifications to Android and iOS devices, simplifies the notification process but restricts certain platform-specific features. On the global standardization front, organizations like Open mHealth [21] aim to create reusable and standardized mobile health studies. However, many vendors in the ZotCare ecosystem have yet to adopt these standards.

## 6.3 Intuitive Design

An intuitive system design enables users to fully engage with a platform's capabilities without requiring excessive time spent on learning. The interface is self-explanatory, allowing users to intuitively understand how to interact with it, enhancing the user experience and operational speed. For ZotCare, which aims to facilitate research and expedite production, particularly for researchers without a technical background, intuitive design is essential. It broadens the system's accessibility and lowers the learning curve.

However, challenges abound in implementing intuitive design within ZotCare. Primarily, the system's programmable services, often rich in meta-variables, configurations, and conditions, can become convoluted. Debugging adds another layer of complexity, as programming and execution often occur in separate environments. For example, interactions may be programmed in the researcher's dashboard but executed on ZotCare's mobile application, complicating debugging errors and tracing them back to their source.

Addressing these user experience (UX) challenges requires a multi-faceted approach, encompassing UI enhancements and underlying infrastructure changes. On the UI front, a more structured, hierarchical interface can guide the user through the study design process more logically than a flat listing interface. For instance, the dashboard could be designed to follow the workflow of setting up a new study, presenting users with the relevant options at each stage. Additional UI improvements might include graphical helpers for text-based services and visual connections between interacting components. This could simplify navigation because unique identifier keys link various services in ZotCare.

Creating traceable logs can help identify and resolve errors more effectively on the infrastructure side. For example, if an error occurs during the rendering of an interaction, ZotCare could log the error at each stage, allowing for targeted troubleshooting. Implementing higher-level plugins can foster a community among ZotCare researchers, enabling them to share solutions. Such a plugin could, for example, send morning notifications based on Oura's sleep data and be activated with a single click utilizing different services in tandem. Finally, modularization can simplify the UI by allowing researchers to compartmentalize various aspects of their projects, making debugging and implementation more manageable. For example, profile keys can be used in different places and for different solutions, but they are all programmed on the same screen. This screen might overflow the researchers with information and make it challenging to debug their solutions.

# Chapter 7

# Artifacts

## 7.1 Access and usage

ZotCare services are available for researchers and health practitioners who would like to run mHealth studies or try a new approach to reaching their clients or patients. Manuals, documentation, and tutorials for ZotCare services are available and will be updated at this address: https://futurehealth.uci.edu/zotcare/. Researchers can fill out the mHealth study form on this webpage to demo the services and use them in their next mHealth study. Besides the services, ZotCare apps for Android and iOS are also available for download.

## 7.2 Code Repositories

Many lines of code and software were built to create ZotCare. These codes are valuable for other researchers who want to use ZotCare in an exclusive environment for their studies, are interested in mHealth research experiments, or are trying to create branches to add features or build new services on top of ZotCare. The source code for ZotCare will be available in

Github under this repository: https://github.com/ZotCare.

## 7.3 Data sets

A considerable amount of data has been generated as a byproduct of using ZotCare in different studies. These data hold great value in future research. These datasets contain multiple data collection modalities, often covering more aspects than the study's primary purpose. This allows researchers from different backgrounds to explore this data through their perspectives. Besides that, there are many methods of processing and experimenting on each dataset, which creates opportunities to get better results even on the explored aspects of a dataset. We plan to curate some of the datasets generated by studies using ZotCare and make these available to the public via the UCI Institure for Future Health.

# Chapter 8

# Future work

The future of ZotCare will predominantly hinge on two principal enhancements: the integration of new features and the introduction of novel services. Additional features can be incorporated into the existing services without altering the current orchestration, for instance, augmenting support for new devices, introducing new types of profiles, or incorporating new actions and triggers into RPII services. Other relevant features, such as chatbots, could serve as beneficial additions to ZotCare's interactions. Furthermore, there is potential for increasing the programmability of AI components within the system. The system could be trained to suggest correlations and models congruent with researchers' experimental objectives, mitigating the need for manual programming. Alterations in service orchestration would pose a more significant challenge, as the services must retain their flexibility, minimalism, and comprehensiveness. Nevertheless, with the burgeoning demand in the mHealth sector, the team will explore opportunities for redesigning the service orchestration.

While programmable services offer a high level of customization, they adhere to stringent syntax and rules, necessitating a learning phase for researchers. Besides that, some functionalities and patterns in mHealth systems often repeat in a similar way between studies, such as smart personalized notifications, standard data processing methods, and standard adaptive interventions. In order to make it simpler for researchers, efforts are underway to incorporate more templates and straightforward, high-level configurations in the form of modules. These modules can be available to researchers to add standard mHealth functionalities to their studies without the need to learn knowledge over ZotCare programmable services or recreate standard functionalities that have been done before.

# Bibliography


[1] I. Azimi, O. Oti, S. Labbaf, H. Niela-Vilén, A. Axelin, N. Dutt, P. Liljeberg, and A. M. Rahmani. Personalized maternal sleep quality assessment: An objective iot-based longitudinal study. *IEEE Access*, 7:93433–93447, 2019.

[2] Bridge. *Bridge Platform*. Sage Bionetworks, 2023. `https://developer.sagebridge.org/index.html` [Accessed at June 10, 2023].

[3] Ethica. *Ethica*. Ethica, 2023. `https://ethicadata.com/` [Accessed at June 2, 2023].

[4] D. Ferreira, V. Kostakos, and A. K. Dey. AWARE: Mobile context instrumentation framework. *Frontiers in ICT*, 2(APR):6, apr 2015.

[5] Firebase. *Firebase*. Google, 2023. `https://firebase.google.com/` [Accessed at June 2, 2023].

[6] Fitbit. *Fitbit Developer Page*. Fitbit, 2023. `https://dev.fitbit.com/` [Accessed at June 2, 2023].

[7] Garmin. *Garmin Developer Page*. Garmin, 2023. `https://developer.garmin.com/` [Accessed at June 2, 2023].

[8] T. J. Gerpott and S. Thomas. Empirical research on mobile Internet usage: A meta-analysis of the literature. *Telecommunications Policy*, 38(3):291–310, apr 2014.

[9] Google. *Wear OS Developer Page*. Google, 2023. `https://developer.android.com/wear` [Accessed at June 2, 2023].

[10] M. Hassanalieragh, A. Page, T. Soyata, G. Sharma, M. Aktas, G. Mateos, B. Kantarci, and S. Andreescu. Health monitoring and management using internet-of-things (iot) sensing with cloud-based processing: Opportunities and challenges. pages 285–292, 2015.

[11] S. M. Hossain, T. Hnat, N. Saleheen, N. J. Nasrin, J. Noor, B. J. Ho, T. Condie, M. Srivastava, and S. Kumar. mCerebrum: A Mobile Sensing Software Platform for Development and Validation of Digital Biomarkers and Interventions. *Proceedings of the ... International Conference on Embedded Networked Sensor Systems. International Conference on Embedded Networked Sensor Systems*, 2017, nov 2017.

[12] ilumivu. *ilumivu*. ilumivu, 2023. https://ilumivu.com/ [Accessed at June 2, 2023].

[13] S. Jafarlou, J. Lai, Z. Mousavi, S. Labbaf, R. Jain, N. Dutt, J. Borelli, and A. Rahmani. Objective prediction of tomorrow's affect using multi-modal physiological data and personal chronicles: A study of monitoring college student well-being in 2020. *arXiv preprint arXiv:2201.11230*, 2022.

[14] T. Jimah, H. Borg, M. A. Mehrabadi, S. Labbaf, and Y. Guo. A Digital Health Approach to Promote Emotional Well-Being in Pregnant Women. *Journal of Obstetric, Gynecologic & Neonatal Nursing*, 50(5):S12–S13, oct 2021.

[15] S. Labbaf, M. Abbasian, I. Azimi, N. Dutt, and A. M. Rahmani. Zotcare: A flexible, personalizable, and affordable mhealth service provider. *Frontiers in Digital Health*, 2023.

[16] J. Lai, A. Rahmani, A. Yunusova, A. P. Rivera, S. Labbaf, S. Hu, N. Dutt, R. Jain, and J. L. Borelli. Using Multimodal Assessments to Capture Personalized Contexts of College Student Well-being in 2020: Case Study. *JMIR Form Res*, 5(5):e26186, may 2021.

[17] J.-A. Lee, S. Labbaf, A. Rahmani, P. Kehoe, and N. Dutt. P4-394: Wearable internet-of-things technology: An immigrant dementia caregivers pilot intervention. *Alzheimer's & Dementia*, 15(7S_Part_28):P1452–P1453, 2019.

[18] L. Lu, J. Zhang, Y. Xie, F. Gao, S. Xu, X. Wu, and Z. Ye. Wearable Health Devices in Health Care: Narrative Systematic Review. *JMIR Mhealth Uhealth*, 8(11):e18907, nov 2020.

[19] McKinsey. *Telehealth: A quarter-trillion-dollar post-COVID-19 reality?* McKinsey, 2021. https://www.mckinsey.com/industries/healthcare/our-insights/telehealth-a-quarter-trillion-dollar-post-covid-19-reality [Accessed at June 14, 2023].

[20] M. A. Mehrabadi. *Holistic Health Monitoring and Personalized Intervention for Well-Being Promotion*. PhD thesis, 2022. Copyright - Database copyright ProQuest LLC; ProQuest does not claim copyright in the individual underlying works; Last updated - 2023-03-08.

[21] O. mHealth. *Open mHealth*. Open mHealth, 2023. https://www.openmhealth.org/ [Accessed at Sept 3, 2023].

[22] Z. A. Mousavi, J. Lai, K. Simon, A. P. Rivera, A. Yunusova, S. Hu, S. Labbaf, S. Jafarlou, N. D. Dutt, R. C. Jain, et al. Sleep patterns and affect dynamics among college students during the covid-19 pandemic: Intensive longitudinal study. *JMIR formative research*, 6(8):e33964, 2022.

[23] H. Niela-Vilén, J. Auxier, E. Ekholm, F. Sarhaddi, M. Asgari Mehrabadi, A. Mahmoudzadeh, I. Azimi, P. Liljeberg, A. M. Rahmani, and A. Axelin. Pregnant women's

daily patterns of well-being before and during the covid-19 pandemic in finland: Longitudinal monitoring through smartwatch technology. *PLOS ONE*, 16(2):1–13, 02 2021.

[24] OneSignal. *OneSignal Notification Service*. OneSignal, 2023. `https://onesignal.com/` [Accessed at June 2, 2023].

[25] Oura. *Oura Ring Developer Page*. Oura, 2023. `https://cloud.ouraring.com/v2/docs` [Accessed at June 2, 2023].

[26] A. M. Rahmani, J. Lai, S. Jafarlou, I. Azimi, A. Yunusova, A. P. Rivera, S. Labbaf, A. Anzanpour, N. Dutt, R. Jain, and J. L. Borelli. Personal mental health navigator: Harnessing the power of data, personal models, and health cybernetics to promote psychological well-being. *Frontiers in Digital Health*, 4, 2022.

[27] Y. Ranjan, Z. Rashid, C. Stewart, P. Conde, M. Begale, D. Verbeeck, S. Boettcher, R. Dobson, and A. Folarin. Radar-base: Open source mobile health platform for collecting, monitoring, and analyzing data using sensors, wearables, and mobile devices. *JMIR Mhealth Uhealth*, 7(8):e11734, Aug 2019.

[28] S. M. Rodrigues, A. Kanduri, A. Nyamathi, N. Dutt, P. Khargonekar, and A. M. Rahmani. Digital Health-Enabled Community-Centered Care: Scalable Model to Empower Future Community Health Workers Using Human-in-the-Loop Artificial Intelligence. *JMIR Formative Research*, 6(4), apr 2022.

[29] Samsung. *Samsung Tizen Device sensors*. Samsung, 2023. `https://docs.tizen.org/application/native/guides/location-sensors/device-sensors/` [Accessed at June 2, 2023].

[30] N. Sattari, M. A. Mehrabadi, S. A. H. Aqajari, J. Zhang, K. Simon, E. Alzueta, T. Dulai, M. de Zambotti, F. Baker, A. Rahmani, and S. Mednick. 079 Sleep Quality Prediction During the Menstrual Cycle based on Daily Sleep Diary Reports. *Sleep*, 44(Supplement_2):A33–A33, may 2021.

[31] A. Tazarv, S. Labbaf, A. Rahmani, N. Dutt, and M. Levorato. Active reinforcement learning for personalized stress monitoring in everyday settings. *arXiv preprint arXiv:2305.00111*, 2023.

[32] A. Tazarv, S. Labbaf, A. M. Rahmani, N. Dutt, and M. Levorato. Data collection and labeling of real-time iot-enabled bio-signals in everyday settings for mental health improvement. In *Proceedings of the Conference on Information Technology for Social Good*, pages 186–191, 2021.

[33] Whitings. *Whitings Developer Page*. Whitings, 2023. `https://developer.withings.com/` [Accessed at June 2, 2023].

[34] Z. Yang, I. Azimi, S. Jafarlou, S. Labbaf, J. Borelli, N. Dutt, and A. M. Rahmani. Loneliness Forecasting Using Multi-modal Wearable and Mobile Sensing in Everyday Settings. *medRxiv*, page 2023.06.08.23291165, jun 2023.

[35] A. Yunusova, J. Lai, A. P. Rivera, S. Hu, S. Labbaf, A. M. Rahmani, N. Dutt, R. C. Jain, and J. L. Borelli. Assessing the Mental Health of Emerging Adults Through a Mental Health App: Protocol for a Prospective Pilot Study. *JMIR Res Protoc*, 10(3):e25775, mar 2021.