DataSentry: Building Missing Data Management System for In-the-Wild Mobile Sensor Data Collection through Multi-Year Iterative Design Approach

Yugyeong Jung
School of Computing
KAIST
Daejeon, Republic of Korea
yugyeong.jung@kaist.ac.kr

Hei Yiu Law
School of Computing
KAIST
Daejeon, Republic of Korea
emilyelhy@yahoo.com

Hadong Lee
Seoul National University
Seoul, Seoul, Republic of Korea
leeha@snu.ac.kr

Junmo Lee
School of Computing
KAIST
Daejeon, Republic of Korea
leejunmo84@gmail.com

Bongshin Lee
Yonsei University
Seoul, Republic of Korea
b.lee@yonsei.ac.kr

Uichin Lee*
School of Computing
KAIST
Daejeon, Republic of Korea
uclee@kaist.ac.kr

Abstract

Mobile sensor data collection in people's daily lives is essential for understanding fine-grained human behaviors. However, in-the-wild data collection often results in missing data due to participant and system-related issues. While existing monitoring systems in the mobile sensing field provide an opportunity to detect missing data, they fall short in monitoring data across many participants and sensors and diagnosing the root causes of missing data, accounting for heterogeneous sensing characteristics of mobile sensor data. To address these limitations, we undertook a multi-year iterative design process to develop a system for monitoring missing data in mobile sensor data collection. Our final prototype, DataSentry, enables the detection, diagnosis, and addressing of missing data issues across many participants and sensors, considering both within- and between-person variability. Based on the iterative design process, we share our experiences, lessons learned, and design implications for developing advanced missing data management systems.

CCS Concepts

• **Human-centered computing → Empirical studies in ubiquitous and mobile computing**; *Visualization systems and tools.*

Keywords

mobile data, data collection, visualization

ACM Reference Format:

Yugyeong Jung, Hei Yiu Law, Hadong Lee, Junmo Lee, Bongshin Lee, and Uichin Lee. 2025. DataSentry: Building Missing Data Management System for In-the-Wild Mobile Sensor Data Collection through Multi-Year Iterative Design Approach. In *CHI Conference on Human Factors in Computing Systems (CHI '25), April 26–May 01, 2025, Yokohama, Japan.* ACM, New York, NY, USA, 18 pages. https://doi.org/10.1145/3706598.3713314

*Corresponding author

ACM ISBN 979-8-4007-1394-1/25/04
https://doi.org/10.1145/3706598.3713314

1 Introduction

Research in human–computer interaction (HCI) and ubiquitous computing fields is increasingly leveraging mobile sensor data to gain fine-grained insights into human behaviors, including diagnosing health conditions [42, 57, 58], predicting productivity [36, 41, 56], and analyzing social interactions [10, 24]. The collection of sensor data from mobile devices in everyday life is crucial for conducting mobile sensing studies [13, 27].

As these sensor data are collected in real-world environments, various participant- or system-related issues can result in missing data (e.g., empty periods in the sensor stream) during data collection due to issues related to participants' behaviors or data collection systems [15, 25, 26, 37, 64]. Failing to address these missing data issues can trigger a data cascade, leading to poor performance in downstream tasks such as modeling and analysis [51]. To prevent this situation, it is crucial for domain researchers to detect missing data, diagnose root causes, and address them promptly [64]. Previous studies have proposed diverse monitoring systems alongside mobile sensing frameworks [16, 17, 47] to address the missing data challenges. These systems provide an interface focusing on *detecting* missing data in raw data streams [20, 47] or aggregated metrics [16, 54] to allow researchers to find missing data.

Despite such efforts, challenges remain in detecting, diagnosing, and addressing missing data. Because various sensors adopt event-based sensing based on human behavior (e.g., logging when specific events related to human behavior occur, such as moving or using apps), the data inherently exhibit variability *between* participants. For instance, participants who are active throughout the day log data frequently, whereas those who are less active log data sparsely. Even *within* a participant, periods of frequent data collection exist, followed by those of low activity where data are scarcely logged. Such variability complicates determining whether a missing period is simply owing to the common behavior patterns or the result of data collection issues.

In addition, due to the complex nature of mobile sensor data, prior systems are constrained by two key limitations. First, they provide a limited view to capture missing data originating from various sensors and participants. They have focused on monitoring compliance-related data (e.g., data collected through participant compliance, such as self-report surveys or wearable use) or examining individual sensor streams, making it difficult to obtain a comprehensive understanding of missing data across many sensors and participants. Second, existing systems do not account for the inherent within- and between-person variability in human behavior sensing data, making it challenging to diagnose the causes of missing data. Furthermore, different causes (e.g., frequent battery depletion or server issues) may need different ways of addressing them (e.g., giving instructions to participants or checking server status). However, examining aggregated metrics or individual sensor data does not provide information on why the data is missing or which interventions should be taken.

Considering these limitations, we designed and developed a missing data management system to help researchers detect missing data, diagnose their root causes, and address them. We engaged in a multi-year iterative design process and reported on our experience with how the prototypes were improved. Our design process included three iterations, each involving prototype deployment for in-the-wild data collection, reflecting the practical challenges faced by domain researchers. In the initial stage, we gathered design requirements by interviewing seven researchers. Using these insights, we developed the first prototype, which was deployed in a one-month data collection. Based on the feedback received, we developed a second prototype that was deployed and tested through two data collections. We completed the final prototype, incorporating the feedback, and conducted a user study (N = 26).

Our final prototype, DataSentry, provides an interface enabling researchers to detect, diagnose, and address missing data during mobile sensor data collection. It provides a comprehensive overview to help detect missing data when collecting data from many people and sensors. Based on this, the system offers visualization that allows a fine-grained diagnosis of missing data. It considers both within- and between-participant variabilities, allowing researchers to identify the context and underlying causes for missing data. In addition, it provides a means to communicate the identified issues with participants to address them. To summarize, our main contributions are as follows:

- We identify key challenges in detecting, diagnosing, and addressing missing data from participant- and system-related issues, considering within-/between-participant variability of mobile sensor data. We report experiences and lessons learned from a multi-year iterative design aimed at managing missing data to handle these challenges.
- We present our final system, called DataSentry, which allows researchers to import and monitor mobile sensor data collected from data collection campaigns. The code is available on GitHub.[1]
- We propose design implications that can be utilized for developing advanced missing data management systems.

[1]https://github.com/Kaist-ICLab/DataSentry

2 Related Work

In this section, we review existing literature and tools relevant to issues that lead to missing data in mobile data collection and the systems developed to monitor and manage mobile sensor data.

2.1 Participant- and System-related Issues Causing Missing Data in Mobile Sensor Data Collection

Studies in the field of ubiquitous computing have focused on collecting "human data" using mobile and wearable sensors [12, 62]. Researchers encounter diverse issues affecting data quality while collecting data in the wild. We categorize them into participant- and system-related issues depending on their causes.

Participant-related issues result from the behavior of data collection participants, including compliance with self-reports, powering off smartphones or sensors, and dropping out of the data collection campaign. The primary is self-report compliance, wherein participants forget or ignore self-report requests [29]. Furthermore, they deliberately or unintentionally switch off smartphones or sensors for battery saving or privacy protection [15, 46, 64]. Sometimes, they forget to switch them on later, leaving valuable information missing. Some are disconnected from research teams on certain occasions, which results in complete absence from data collection [26, 64]. This phenomenon, commonly known as dropout, refers to a situation where participants withdraw from the study [22, 43].

System-related issues stem from the data collection system (e.g., networks, servers, and sensing devices). With the loss of wireless connections for smartphones, a problem exists in which data is not sent to the server [15, 18, 26, 52]. Data collection servers can also cause issues: when they experience technical problems, data reception from smartphones may face challenges [45, 64]. Due to technical problems with smartphones or sensors, a possibility of missing data exists [25].

Various studies in the fields of ubiquitous computing and HCI [19, 33, 54, 60, 61] have been conducted to identify and address these issues, with a focus on improving self-report compliance (e.g., whether participants respond to self-report) or wearable compliance (e.g., whether participants wear the devices for the period stipulated by researchers). For example, an interactive visualization system [54] and regression modeling [33] have been used to track and predict compliance. Different approaches have been adopted to enhance compliance, including visualization, gamification, and context-informed scheduling [19, 60, 61]. Although these studies offer opportunities to monitor and resolve missing data, most work focused on participant-related issues (e.g., self-report or wearable compliance). Our work extends existing research by developing a comprehensive visualization system that supports the exploration of sensor data streams, enabling the identification of diverse participant- and system-related issues.

2.2 Sensor Data Collection and Missing Data Monitoring Systems in Mobile Sensing Research

2.2.1 Sensor Data Monitoring Systems: Monitoring of Raw Data vs. Aggregated Data. Diverse sensing frameworks can capture mobile sensor data from people's daily lives [7, 8, 28, 48, 59]. A simple approach to ensure the quality of the collected data is to visualize the data (e.g., Python plotting) [39, 49]. Beyond this, many tools have been developed for data collection monitoring. These tools help alleviate researchers' burden by providing the ability to monitor data quality including missing data. Depending on the type of data being monitored, these tools can be categorized into 1) monitoring of raw data and 2) monitoring of aggregated data.

Several tools provide the ability to monitor raw sensor data, allowing researchers to check for missing data and the status of data collection. The Incense IoT sensing framework [47] offers a monitoring interface by visualizing individual sensor streams, allowing users to select the specific device, sensor type, and time period. The IBM Watson IoT Platform [20] enables the collection of sensor data while allowing real-time monitoring of the connection, server, and live data of IoT devices, supporting the observation of missing data. The Ohmage sensing framework [55] provides a dashboard to monitor the progress of raw sensor data collection.

Another approach involves tools that support monitoring aggregated metrics. The mk-sense sensing framework [16] provides a dashboard for supervising the data count per time window in the form of a heatmap, allowing users to identify periods with missing data. The AndWellness platform [17] calculates metrics such as battery charge levels, the number of smartphone interactions, and survey-response count to analyze participation quality, to determine if there is any missing data. Research by Talkad et al. [54] and the MobisenseXS platform [40] focus on features to monitor user compliance (e.g., whether users respond to surveys or wear wearables). They calculate metrics such as the number of survey responses or wearable usage time to identify periods with missing compliance-related data. The RedCap platform [6] offers a dashboard to track whether participants have completed the tasks required for the study, providing an overview of where the task completion was missing.

2.2.2 Remaining Key Challenges in Sensor Data Monitoring. While existing tools enable missing data monitoring, they have limited support in detecting missing data, and lack capabilities for diagnosing and addressing missing data issues in mobile sensor data collection. Researchers, therefore, face the following key challenges:

Challenge 1. Comprehensive understanding of missing data. Most existing systems focus narrowly on individual sensor streams or aggregated metrics of specific sensor types, requiring users to manually select small subsets of sensors, participants, or time frames to inspect. This makes it challenging to achieve a holistic understanding of missing data over diverse sensors and participants.

Challenge 2. Diagnosing and addressing missing data issues. Various causes for missing data could exist; they could stem from participant-related issues (e.g., non-responsiveness to self-reports, or switching off smartphones or sensors) and system-related issues (e.g., server failures, network disruptions, or sensor malfunctions). However, existing systems focus primarily on detecting missing data without offering capabilities to diagnose its root causes. Furthermore, they do not support addressing missing data issues, such as contacting participants with preventive instructions.

To tackle these challenges, we introduce DataSentry, a missing data management system for mobile sensor data collection. It supports the exploration of missing data at both the overview and fine-grained levels, providing a visualization that helps diagnose the causes of missing data and features to address them.

3 Formative Study

In this section, we present a formative study focusing on insights gained from interviews with researchers experienced in mobile sensor data collection.

3.1 Method

To identify the design requirements for a missing data management system, we conducted semi-structured online interviews with seven researchers (R1 - R7) from six research groups experienced in mobile data collection and analysis (Table 1 in Appendix). We targeted research groups in mobile sensing working on independent projects. The interviews covered three areas: 1) monitoring setup (goals and tools), 2) monitoring tasks, and 3) challenges. With participants' consent, the interviews were audio-recorded, and each participant received a $30 USD coffee coupon as compensation.

We applied thematic analysis [2] to the transcribed interview data, following six stages: familiarizing with data, generating codes, searching for themes, reviewing themes, defining and naming themes, and producing report. We adopted an inductive approach, starting from raw interview data. Two of the authors repeatedly reviewed interview data, assigned thematic codes, and merged similar ones. The codes were then categorized into major themes, which were prioritized by frequency until reaching consensus. This analysis resulted in three key design requirements.

3.2 Results

Researchers collected diverse sensor data from smartphones, such as physical activity, call logs, and app usage logs; specifically, six collected self-reports using mobile ESM by periodically sending requests to participants. A few researchers collaborated with teammates who assisted in managing data collection (R2 and R3). However, most of them worked alone owing to the difficulty of recruiting assistants, which fatigued them during data collection.

Design Requirement 1: Overviewing missing data across many people and sensors. Researchers wanted to identify missing data at a glance across many sensor streams. Their goal was to ensure that data was collected without missing data related to participant- or system-related issues. They periodically downloaded data from database, counted the number of rows per sensor and participant, and plotted using Python. They thought if the count were small, there would be missing data. Six researchers were interested in low self-report compliance (e.g., how many times participants responded to self-reports). They counted the number of daily surveys and found fewer participants than the predefined number. Three researchers were interested in missing periods of smartphone log data, such as location or app usage. They counted the number of rows within a time window (day or hour); if the count was low, they considered missing period inside the window. All researchers acknowledged that reviewing data collection status from many people and sensors requires considerable effort. Because they worked alone, they did not have enough time to monitor all sensor streams. R4 noted, "*There were so many sensors, activity transitions, phone calls, and screen logs that I couldn't thoroughly monitor the data. I ended up just doing the bare minimum.*" Consequently, unmonitored data led to missing data. R7 proposed a comprehensive view to check the entire data collection status at a glance: "*At the very least, it would be helpful to display whether data from each sensor and each person was collected.*"

Design Requirement 2: Identifying missing data in event-based sensing. Researchers desired to determine which data collections contained long missing periods for event-based sensing data. After researchers had found participants in a few rows, they wanted to determine whether this information implied missing data. If the daily number of rows or sensing frequency were predefined, it was easy to figure out participants with missing data. They calculated the number of rows within a time window based on predefined frequency and figured out participants who were lower than it. However, most data items did not have a predefined frequency; they are event-based sensing based on human behaviors. Therefore, researchers encountered difficulty in identifying the number of rows to be collected over some time (e.g., hours or days). Even if the number of rows was small, it was difficult to determine whether the participant had missing data due to data collection issues or a low degree of behavior. R2, who designed a GPS collector to be triggered only when a participant started to move, mentioned, "*If there's only a small number of rows, it might seem like there's an issue with the sensor collector. However, it could be because the user didn't move.*" To cope with the challenge, R5 was comparing the number of rows across participants and selecting participants with a relatively low number of rows. He thought the average number of rows could be a reference representing participants who performed a common activity and regarded participants below the average as possibly having missing data.

Design Requirement 3. Diagnosing missing data causes. Researchers needed to diagnose detailed causes of missing data by observing several data streams. R7 wanted to find participants who intentionally shut down smartphones and wanted to observe multiple data streams together: "*If I can check multiple data items at a glance, then I can determine why the data was not collected. If all data streams were not collected simultaneously, the participants might turn off their smartphones.*" R5 wanted to understand the reasons for prolonged missing periods in self-reports. To gain insights into missing self-reports, he observed various data items together: "*When the self-report was missing, I opened up the raw data files and cross-checked other frequently sensing data during that period. If they exist, this helped me realize that the missing data was not due to smartphone breakdown but because the participant did not respond to the survey.*" Additionally, one of the reasons for missing data was related to the OS version. After completing data collection, R5 discovered that a group of people using Android 10 had a low number of GPS data. He reflected, "*Twenty people didn't have GPS data! I want to add a feature that groups multiple users with low data counts and checks whether they're using the same version.*"

4 Iterative Design of Missing Data Management System in Mobile Sensor Data Collection

This section details the iterative design of a missing data management system for mobile data collection, incorporating feedback from the formative study and deployments, and concludes with a user study.

4.1 Overview of Iterative Design Process

Starting from the researchers' experiences and challenges in formative study, we conducted a multi-year iterative design (2021 to 2023) to develop a missing data management system. Inspired by previous studies on multi-year system design and improvement [4, 14, 44], we iteratively designed, developed, and evaluated the system over a few years. We collaborated with target users to continuously identify and address their needs. We deployed our system in in-the-wild data collection, incorporated features informed by real-world system usage, and iteratively resolved issues to enhance system capabilities. Figure 1 provides a summary of the iterative design process and data collection campaigns to evaluate each prototype.

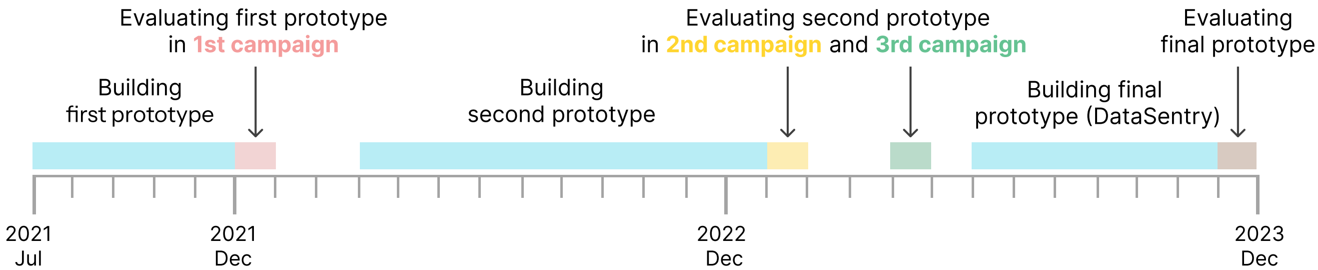
To design the initial prototype, we invited two graduate students with extensive knowledge of system design and mobile sensor data, incorporating their suggestions to address the challenges. This prototype was deployed in a one-month data collection, where 17 types of sensor data and emotion-related self-report ESM were collected from 116 participants (43 women, 73 men; age: $M = 23.5$, $SD = 3.5$). Two graduate student researchers were invited to use the prototype. Based on their experiences, we collected the researcher's feedback and improved a second prototype, which was deployed at two in-the-wild data collection campaigns; the second campaign collected 17 types of mobile sensor data with self-report ESM collected via a smart speaker from 20 participants (7 women, 13 men; age: $M = 24.8$, $SD = 2.7$), while the third campaign collected five types of mobile sensor data with self-report ESM from 24 participants (9 women, 15 men; age: $M = 21.3$, $SD = 2.1$). Three researchers (two for the second campaign and one for the third campaign) were recruited for missing data monitoring using the prototype. Three of the authors synthesized the researchers' feedback and implemented the final prototype. Finally, we conducted an in-lab user study with 26 mobile sensing researchers to assess whether the prototype could be useful for researchers. Details of the researchers in the three data collection campaigns and user study are described in Tables 2 and 3 in Appendix.

Following the Institutional Review Board (IRB) recommendations, all processes were informed by the researchers, and consent was obtained. We describe how the design was refined, highlighting key feedback from in-the-wild data collection campaigns.

4.2 First Design Iteration

4.2.1 Prototype design. To design and develop the first prototype, we did a rapid prototyping with Tableau [53], a commercial software that supports the easy visualization of tabular data. We detail how we reflected three design requirements from the formative study.

Overview of missing data across people and sensors. Reflecting on the *Design Requirement 1*, the prototype provides a view letting users identify missing data across multiple participants and sensors



First campaign	Second campaign	Third campaign
Data Collection	**Data Collection**	**Data Collection**
17 kinds of mobile sensor data 5 kinds of self-report ESM	19 kinds of mobile sensor data 4 kinds of self-report ESM	5 kinds of mobile sensor data 2 kinds of self-report ESM
Participants	**Participants**	**Participants**
116 participants (43 women; age: M = 23.5, STD = 3.5)	20 participants (7 women; age: M = 24.8, STD = 2.7)	24 participants (9 women; age: M = 21.3, STD = 2.1)
Researchers	**Researchers**	**Researchers**
2 researchers (1 women; age: M = 28.5, STD = 2.2)	2 researchers (2 women; age: M = 26.0, STD = 1.4)	1 researcher (1 man; age = 33)

Figure 1: The overview of our multi-year iterative development process: (upper diagram) Timeline of our iterative design process and (table below) Details of the three data collection campaigns conducted to evaluate and iteratively refine the prototypes.

(Figure 2A). It calculates the daily item count for each sensor and participant and visualizes the information as stacked bars (with the x-axis being data counts, the y-axis being participant IDs, and colors being sensor data types). By observing the stacked bars, users can identify participants with low data counts, which might indicate the presence of missing data. For example, in Figure 2A, users may suspect that Participant 10 has missing data, as their total item count is lower than others'. To check individual sensor data, users can click on the sensor name in the Data Type legend to filter a single bar chart.

Data-driven guidelines using statistical quality control mechanism. Along with the overview of data collection (Figure 2A), users can get guidelines to determine which item count might indicate missing data in event-based sensing (Figure 2B), as the data collection varies depending on human behaviors. Drawing inspiration from the *Design Requirement 2*, we enabled the identification of participants with missing data by comparing the data counts across participants to find those with relatively lower data counts. We adopted the concept of control charts [38] commonly used in the manufacturing domain, defining outlier metric as values outside $[\mu - k\sigma,\ \mu + k\sigma]$, where k is a constant, and μ and σ are the average and standard deviation. After selecting a specific data type from a dropdown, users can adjust the value of k using a checkbox to decide which outlier range (outside the gray box) to apply. Participants falling below this range are considered to have significant missing data. Based on this, users can filter participants who fall outside the normal range using the range slider in Figure 2A.

Visual exploration for missing data diagnosis. For participants below the lower range, users can gain detailed contexts of missing data through the temporal trend of item counts, reflecting *Design Requirement 3*. By clicking a participant ID in the stacked bars (Fig. 2A), users navigate to a more detailed view of the data collection status (Figure 2D). We resampled data at an hourly level and visualized the temporal trends to check missing areas of a single data item (Figure 2D) possibly due to sensor malfunctioning or turning off the sensor, or the missing area of multiple data items (Figure 2E) due to powering off the smartphone or draining out of battery. Additionally, users can check which data collection apps and OS versions are being used by participants who have small data counts (Figure 2C). When multiple participants are selected in Figure 2A, the heatmap shows the distribution of participants by OS and data collection software versions. Selecting participants with low item counts helps to identify the popular versions they use, and provides hints to users about possible version problems in data collection.

4.2.2 Field deployment. We deployed the first prototype in a real-world mobile data collection from December 2021 to January 2022. The primary objective of the deployment was to investigate the experiences and user needs of using the prototype in the wild and to leverage those insights for refinement.

Data collection and participants. The deployment study was conducted with 116 participants during one month. They installed open-source data collection software named ABC logger [23] and received 96 USD for participation.

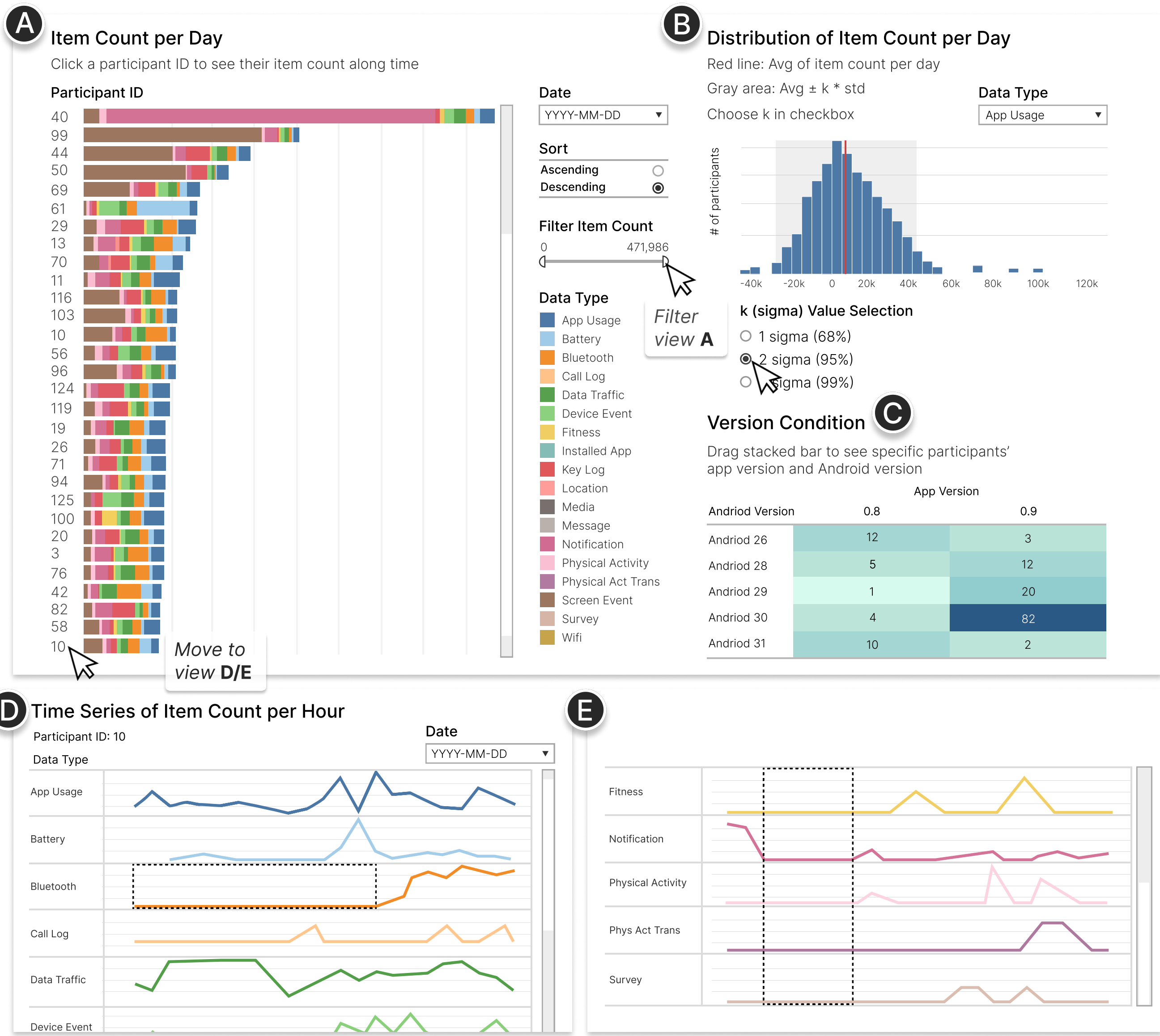


Figure 2: Main views from the first prototype: (A) 'Item Count per Day' view provides the overview of missing data collection across people and sensors, (B) 'Distribution of Item Count per Day' view enables the determination of guidelines for item count metric using statistical quality control mechanism, (C) 'Version Condition' view helps to diagnosis whether the participants with small item counts are using similar version of OS or data collection app, and (D, E) 'Time Series of Item Count per Hour' view visualizes trends of item count per hour to check missing data for a specific sensor (D) or across multiple sensors (E).

Managers and tasks. We recruited two graduate student researchers (R8 and R9). R8 has been working on mobile data collection and analysis and has extensive knowledge in this domain. R9 is not in the mobile sensing domain but has knowledge (such as the types of sensor data collected) of mobile data collection. They were asked to find missing data during data collection and diagnose the causes of missing data. After finding the participants, researchers used a commercial messenger tool to contact them and address the issue (such as asking for ESM surveys or checking whether their smartphones were powered on).

Interview. At the end of data collection, we conducted a 1-hour semi-structured interview with each researcher. The interviews were audio-recorded, transcribed, and analyzed using affinity diagramming to group them into major themes, such as 1) how they used the prototype and their experiences, 2) major design insights, and 3) minor usability issues.

4.2.3 User experience, design insights, and usability issues.

The feedback from researchers helped us understand how the initial prototype supported managing missing data during data collection. Overall, researchers appreciated the use of item count to gain an overview of the data collection and diagnose the contexts of missing data. R8 noted, "*The great thing about these bars is that it's super easy to spot people whose data wasn't collected. If the bar's short, I could tell right away.*" After finding participants with a small bar chart, they moved to the hourly-level time-series view, to observe its temporal trends. R9 shared her experiences of finding one participant with missing data, "*I couldn't believe it - no data was collected over the entire night! It was such a mess that I immediately reached out to him to see if he'd been turning off his phone while sleeping.*"

Design insight 1: Needs for reviewing raw sensor stream data. While the system supported finding participants with missing data, the researchers had insightful suggestions for improving the prototype. Their primary requirement was that they wanted to not only observe aggregated metrics but also dive deeper into raw data from various perspectives to understand where the missing data exist, and what is the context and root cause of the missing. They found that checking the temporal trend of hourly item count could help diagnose the context of missing data to some extent. However, observing the aggregated metrics made it difficult to pinpoint exactly when the data was missing or whether the missing data occurred simultaneously on different sensors. Thus, they wanted to observe raw data to verify exactly when the data was logged.

Design insight 2: Needs for diagnosing the causes of unexpected missing data by observing multiple sensor streams. When a specific participant's data was missing for an extended period, researchers sought to determine whether it was due to typical sensing patterns reflecting the participant's daily behavior, or an unexpected issue in the data collection process (e.g., unexpected participants' behavior or system-related issues). R9 mentioned, "*There was a participant whose app usage data count kept showing up as zero on the line chart. I wanted to know if her data usually showed low counts or if data collection had suddenly stopped, maybe because she turned off the sensor collector.*" R8 added, "*It was interesting to observe multiple sensor data to identify periods where data was missing simultaneously. But even more than that, being able to observe specific sensor data across multiple participants would be useful to catch instances where data wasn't being collected from several people at the same time. Identifying those patterns could help point to issues like server failures where data isn't coming through properly.*" They also emphasized that analyzing multiple sensor streams from different perspectives yields a more comprehensive understanding of the context and underlying causes of missing data.

Usability issues. Researchers provided feedback on the use of stacked bars to display daily item count metrics. In mobile data collection, the wide variation in the scale of count metric across different sensors (e.g., some sensors log more than 10,000 rows while others log just 10–20) made it hard to observe sensor data with smaller scales. They suggested using individual bar charts for each sensor instead, which would make it easier to discriminate between them. Additionally, for data types with specified sampling rates or predefined item counts, they recommended calculating thresholds for these counts within a time window and marking the thresholds on the interface.

Lastly, researchers discussed features that were not particularly useful in the system, such as statistical quality control or version condition exploration. They thought the statistical quality control was still ambiguous to determine participants with missing data since the distribution of daily item counts was usually within a range of $[\mu - 1\sigma,\ \mu + 1\sigma]$. Thus, identifying outlier participants was challenging. Rather than clarifying thresholds of item count using the distribution, they usually sort and skim through whole bars and click participants with relatively small or zero counts. Moreover, since data collection issues due to the app or Android version were rare, the researchers seldom used the version condition heatmap. Consequently, they believed these features could either be removed or assigned a lower priority in the system.

4.3 Second Design Iteration

Reflecting on the feedback from the first design iteration, we improved our design and built the second prototype. The major suggestion from the first iteration was that researchers wanted to observe not only aggregated metrics but also many raw sensor streams (*Design Insight* 1 in the first iteration) to better diagnose the contexts and root causes of missing data (*Design Insight 2* in the first iteration). In addressing the feedback, we considered making a web-based visualization system based on React.js [50] and D3.js [35] instead of Tableau, due to the limitations of the tool that only allows limited types of visualization and features.

4.3.1 Prototype design.

The second prototype offers a comprehensive overview of missing data across many sensors and participants (Figure 3A), which are connected with the fine-grained exploration of raw sensor streams (Figure 3B-D) to diagnose the root causes and contexts of missing data.

Overview of missing data across many people and sensors. As shown in Figure 3A, the prototype visualizes item count metrics (in the 'Count' column) to help users identify participants with low item counts, possibly due to missing data. To address the usability issues found in the first iteration, we replaced the stacked bar chart with individual bar charts and enabled sorting to support the identification of participants with lower item counts. To further support this, the heatmap displays three-hourly item counts in the 'Timeline' column (x-axis as time, color as item count), allowing users to visually detect periods where data collection was sparse. These combined visualizations give users insight into which participants and sensors exhibited lower item counts at specific periods. For data types with predefined item counts (such as a survey) or sensing frequency, the prototype adopts bullet charts [9], which allows for a comparison between target lines and actual measures.

For participants identified in the overview, the prototype provides three kinds of interconnected views (Figure 3B-D) allowing users to observe missing data and diagnose its root causes *(Design Insights 1 and 2 in the first iteration).*

Missing data diagnosis via one participant's multiple sensor streams. The prototype visualizes multiple data streams from a single participant, enabling users to check for simultaneous missing data across multiple sensors or missing data from specific sensors

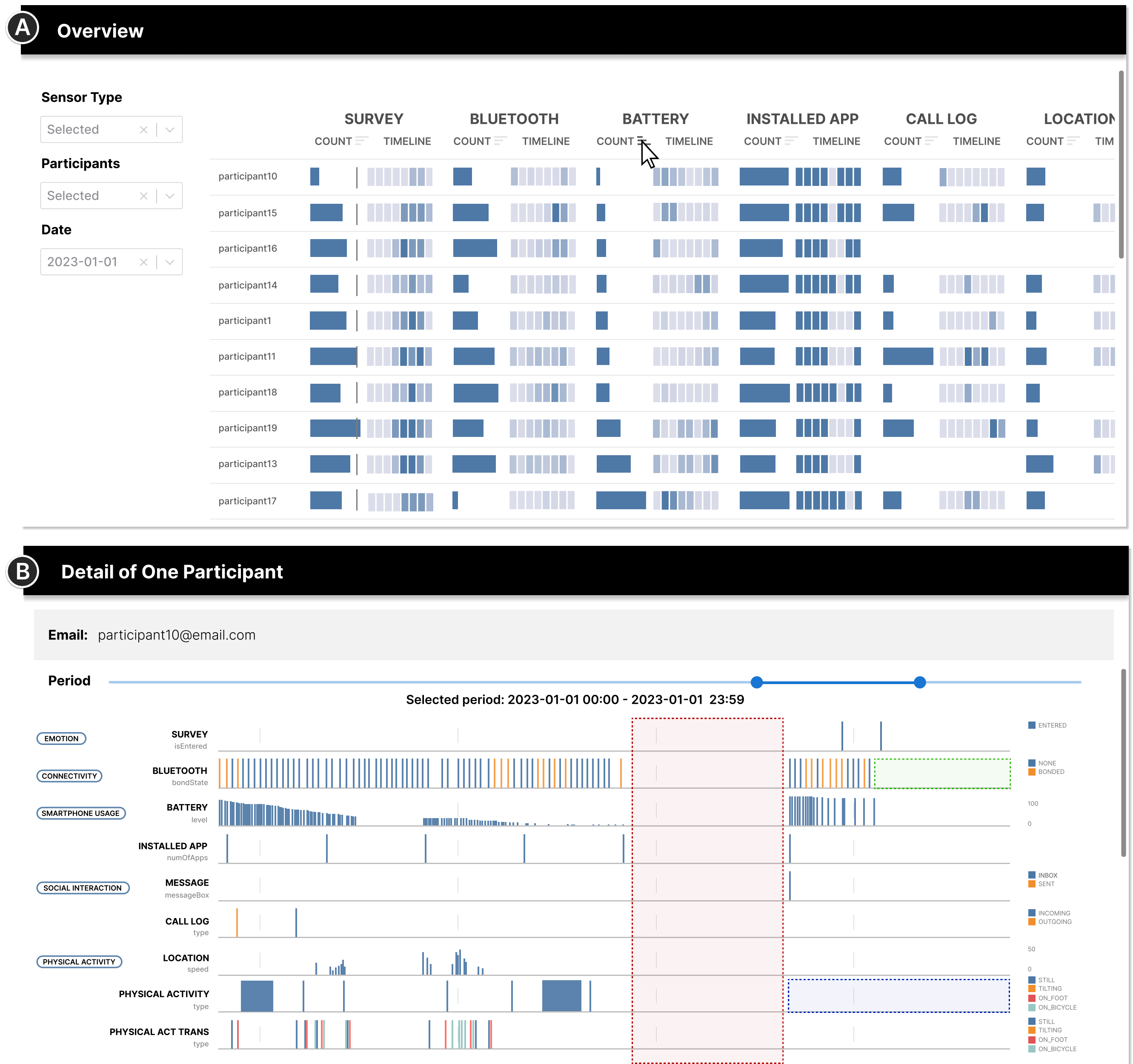


Figure 3: Major views of the second prototype: (A) Overview of item count metric across many people and sensors and (B) Detail of One Participant view.

('Detail of One Participant' view; Figure 3B). For instance, in Figure 3A, users notice that Participant 10 has fewer battery data compared to other participants, a potential indication of the participant experiencing missing data issues. Since battery data is event-based sensing (logging only when specific events such as charging or discharging), a low item count does not necessarily mean missing data. Therefore, users click Participant 10 and move to the 'Detail of One Participant' view to check detailed log data. As marked in the red box in Figure 3B, users diagnose that all sensor streams were missed simultaneously when the battery was discharged. Based on the insights, users can contact the participant to request regular charging to prevent a sudden smartphone shutdown.

Missing data diagnosis using within-participant comparison. The prototype visualizes daily-basis sensor stream data *within* a participant, enabling users to diagnose the contexts of missing data ('Within-Participant Time-Series Comparison' view; Figure 3C). In

Figure 3: Major views of the second prototype: (C) Within-Participant Time-Series Comparison view and (D) Between-Participant Time-Series Comparison view.

the 'Detail of One Participant' view (Figure 3B), noticing the region marked as a blue box, users may wonder why the physical activity data is not collected during the afternoon. Since the data is event-based sensing (logging only when physical activities occur, such as walking or running), it is again challenging to discern whether the missing data is due to the participant's lack of activity or data collection issue (e.g., turning off the sensor or sensor malfunction). To diagnose the detailed contexts, users click on the data type and then select the 'Within-Person Time-Series Comparison' menu item to navigate to Figure 3C. The view shows how the participant's physical activity data has been collected on a daily basis, with the selected period from Figure 3B highlighted in blue text. Users notice that, while the participant's physical activity data was previously collected in the afternoons, the data collection suddenly stopped (marked as a blue box). Given this deviation from the participant's usual sensing patterns, users can contact the participant to check if the sensor was intentionally turned off or if there is a malfunction with the device affecting physical activity sensing.

Missing data diagnosis using between-participant comparison. The prototype visualizes simultaneous sensor stream data *between* participants, enabling users to diagnose the missing data due to server issues ('Between-Participant Time-Series Comparison' view; Figure 3D). In Figure 3B, users may question the period where the participant's Bluetooth data is not collected (marked as a green box). To investigate the detailed contexts, users can click on the data type and select the 'Between-Participants Time-Series Comparison' menu to navigate to Figure 3D. The view shows how Bluetooth sensor data from multiple participants was collected simultaneously, with the selected participant from Figure 3D highlighted in blue text. Users notice that data from many participants is missing at the same time (marked as a green box). This suggests a potential issue with the server receiving data, prompting users to discuss the situation with developers and check the server. Additionally, users notice that the sensing pattern of Participant 17 differs significantly from that of other participants. Then, they can communicate with this participant to check if there are issues with their device's ability to sense Bluetooth data.

4.3.2 Field deployment. We deployed our second prototype to two data collection campaigns. The purpose of the deployment was to assess whether the prototype could be effectively utilized while also identifying insights for further improvement.

Data collection and participants. The second campaign involved smart speaker and mobile data collection from 20 participants for one month, and the third involved stress-related mobile data collection from 24 participants for one month. After receiving an orientation, participants installed data collection apps. In the second campaign, smart speakers (Google Nest Hub) were distributed to participants to collect self-reports of their emotional states in verbal form. This team developed a data collection and context-sensing application running on a smartphone attached to the smart speaker, to trigger it when specific contexts are met (e.g., detection of the presence of a user, or passage of a certain time since the last survey). As compensation for data collection, participants were paid 307 USD. In the third campaign, compensation was based on the number of ESM surveys and their participation in the post-interview, and they received 160 USD on average.

Managers and tasks. Three graduate student researchers (R10 and R11 for the second campaign and R12 for the third campaign), different from those in the first campaign, managed data collection. All had rich experience in mobile data collection and knowledge of mobile sensor data. Before data collection, they were introduced to the system's features and asked to perform detection, diagnosis, and addressing missing data.

Interview. After data collection, a 90-minute semi-structured interview was conducted with each researcher by audio-recording interview contents. We asked what types of data and data collection problems they observed, how the system helped them determine the missing data issues, and what their expectations of the system were. Similar to the first design iteration, the interview contents were transcribed and analyzed via affinity diagramming to analyze major themes, such as their major user experiences of using the system and design insights to improve the prototype.

4.3.3 User experience and design insights. All researchers, who participated in two in-the-wild data collection campaigns, acknowledged that the visualizations in the prototype were helpful, enabling them to detect and diagnose missing data. Notably, the views allowing for fine-grained analysis of multiple sensor streams stood out as particularly useful, as they provided deeper insights into the context and causes of missing data. They shared experiences where these views were practically helpful. R11 recounted how the 'Detail of One Participant' (Figure 3-B) view helped uncover a missing data issue: "*I had no idea there was a missing data before, but this system made it easy to spot. Some participants hadn't had any sensor data for a while, and I found out it was because their Wi-Fi was disconnected, which caused a syncing problem.*" R12 emphasized the value of the within-/between-participant time-series comparison view (Figure 3-C,D) in diagnosing missing data: "*By checking their within-/between-person app usage behaviors, I noticed a sudden missing period that lasted for a long time, on Saturday night! I started thinking maybe the participant turned off the data collector on purpose, so I reached out to them to find out.*"

Design insight 1: Needs for streamlined detection of long-missing periods. In addition to the tool's capability for detecting and diagnosing missing data, researchers suggested improvements for the second prototype. When observing missing data on the 'Detail of One Participant' view, they expressed a desire to know how long these missing periods were, both in comparison to a participant's data and to other participants' data. Before diagnosing the root causes of the missing data, they were initially interested in detecting conspicuously long periods of missing data. To assess the length of these periods, they skimmed through the within/between-participant time-series comparison view (Fig. 3C and Fig. 3D), comparing the length of missing periods against within- or between-participants. If the period was significantly long, they would then observe the raw sensor streams in detail to diagnose the causes of the missing data. However, this process was considered cumbersome; R10 commented, "*To figure out how serious a missing period was, I had to constantly switch between several pages. It wouldn't have been a big deal if I had more time, but doing this for multiple people and sensors made it hard to catch all the missing periods.*" As a result, researchers requested a more streamlined process for detecting long missing periods, with quicker access to information showing whether a specific missing period was unusually long compared to a participant's data or to other participants' data.

Design insight 2: Needs for lowering the burden of communication. Researchers expected features that would facilitate communication with participants to *address* missing data issues. While our prototype lowered the burden of detecting and diagnosing missing data, researchers still had to communicate with participants

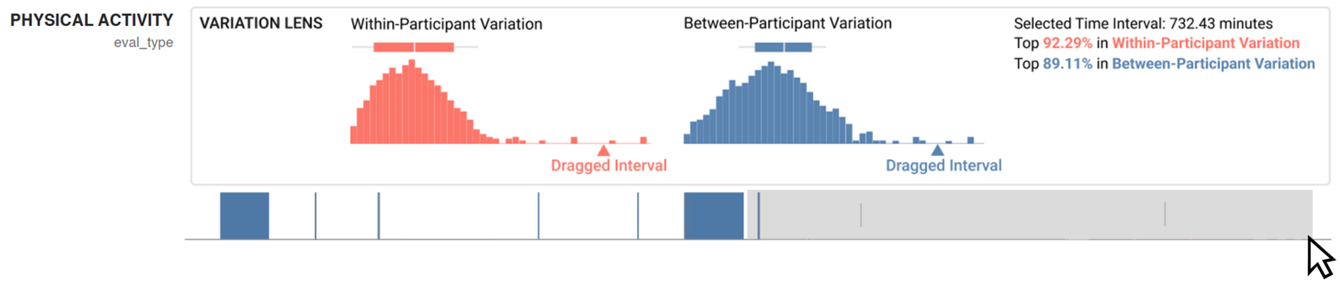


Figure 4: Within- and between-participant variation lens: The within-participant variation shows the time interval distribution of physical activity data within a participant, and the between-participant variation shows the time interval distribution of physical activity data across all participants. The 'dragged interval' is the length of time interval dragged by users. The right side provides information on selected time intervals (length, percentage of within- and between-participant variation).

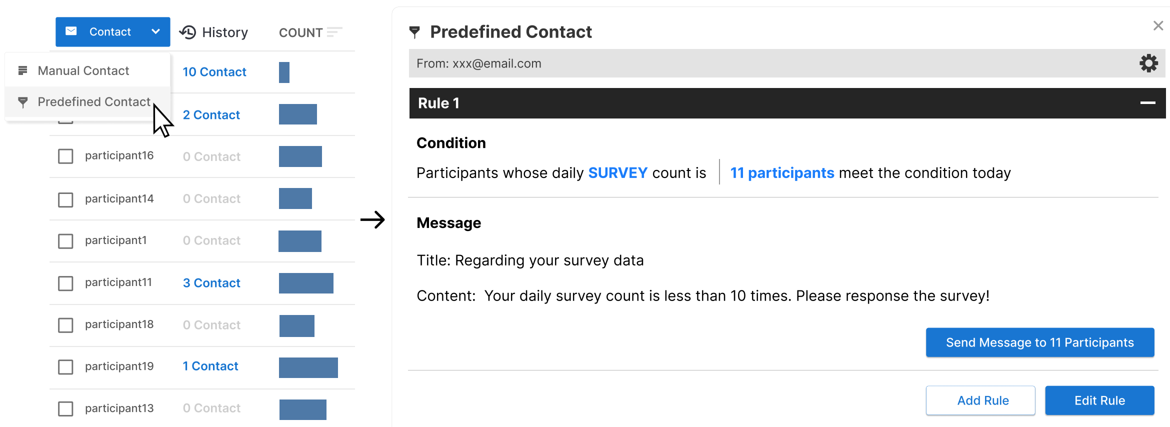


Figure 5: Communication support with predefined contact. Clicking the 'Predefined Contact' menu opens the popup supporting rule-based communication support. Users can add or edit rules using the 'Add Rule' and 'Edit Rule' buttons, respectively.

to address these issues. They often needed to contact participants to correct their improper behaviors (e.g., intentional smartphone shutdowns or missing responses to emotional surveys) or figure out the causes of missing data. However, this process remained a cumbersome task. After identifying participants with missing data, researchers had to find their contact information (phone numbers or email addresses) and input these details into a separate messaging tool to send messages. In this process, they repeatedly sent similar messages to multiple participants, creating a significant burden.

4.4 Final Design

4.4.1 Prototype design. In the second design iteration, researchers wanted to easily detect conspicuously long periods of missing data, compared to a participant's data or multiple participants' data (*Design Insight 1* in the second design iteration). Furthermore, while addressing the missing data issues, they wanted to lower the burden of manual communication with many participants (*Design Insight 2* in the second design iteration). To address the feedback, we introduced two new features on the second prototype and completed our final prototype, **DataSentry**. The following section details the added features to complete the final prototype.

Detection of long-missing periods. To aid researchers find conspicuously long periods of missing data (*Design Insight 1* in the second design iteration), we added a view named "Within/Between-Participant Variation Lens" (Figure 4). This view aids in determining whether a missing data is anomalous compared to a participant's past data or data from other participants. When the user drags the time interval of interest in the raw time series data in the 'Detail of One Participant' view, the system provides a variation lens, visualizing how the selected time interval compares in terms of the distribution of time intervals 1) within the participant's physical activity data (within-participant variation) and 2) of all participants' physical activity data (between-participant variation). Because the selected interval indicates a relatively large in both distributions, the user can infer that the missing period could be an uncommonly long duration compared to the previous sensing patterns and suspects potential data collection issues beyond behavioral characteristics. Through this comparison, users can determine if a specific missing interval deviates from participants' usual sensing patterns, allowing them to investigate raw data in more detail to understand the context and root causes of missing data.

Communication with participants. To facilitate easy contact with participants and prevent recurring missing data issues (*Design Insight 2* in the second design iteration), we introduced a communication support feature (Fig. 5). On the Overview page, the prototype was updated to allow selecting and sending emails to multiple participants simultaneously (left figure in Fig. 5). When users identify participants who require follow-up due to missing data, they can select them and use the 'manual contact' menu to send an email to multiple participants. To further reduce the burden of repetitive communication, we implemented rule-based contact conditions. A popup appears when the user selects the 'predefined contact' menu, where they can set the rule (Fig. 5). For instance, if the rule is set to 'send an email for participants whose daily survey count is less than ten times a day,' 11 participants matching that rule will be filtered. Users can contact them by simply clicking the 'send messages to 11 participants' button. Additionally, the prototype includes a contact history feature, allowing users to track previous communications with each participant. This helps identify participants who frequently require follow-up due to missing data issues. The number of emails sent to each participant is displayed, and users can click on the count to review the content of previous messages via a popup window.

4.4.2 User study. Through deployment studies, we obtained the opinions of researchers in in-the-wild data collection. However, one limitation was the small number of researchers available to provide feedback, as performing multiple data collections is time-consuming and costly. Therefore, to enable researchers from various research groups to evaluate our final system, we opted for a controlled user study to explore the perceived usability of the system and experiences in detecting, diagnosing, and addressing missing data. This study was approved by the IRB and written consent was obtained from researchers.

Participants. We contacted 11 different groups in the mobile sensing domain and recruited 26 researchers (R13–R38, 5 women, 21 men, age: M = 28.48, SD = 4.56), consisting of 5 master's, 10 Ph.D students, and 11 industry professionals in three IT companies. Our goal was to investigate not only how academic research groups but also industry professionals could effectively utilize our system.

Study design. We conducted a within-subject user study to compare the usability and user experience of DataSentry in detecting, diagnosing, and addressing missing data, rather than not using each feature. Participants experienced three conditions: 1) Python plotting, 2) DataSentry Basic version (DataSentry without within-/between-participant time-series comparison and variation lens), and 3) DataSentry Full version (DataSentry with aforementioned features). We chose Python plotting because it was the common practice employed by researchers according to our formative study results. We aimed to assess if and how DataSentry improved the process of detecting, diagnosing, and addressing missing data compared to their prior experiences. We were also interested in examining whether DataSentry's key feature—the within-/between-participant comparison—impacts the usability and experience of the process. To explore this, we included two conditions in our study: a Full version of DataSentry and a Basic version without this feature.

Procedure. We started the study session by introducing our research goal and asked researchers to act as data collection managers. They participated in training sessions to ensure they were familiar with the system. We assigned three conditions in a random order to counterbalance the order effect. Under each condition, they performed three tasks: 1) detecting sensors and participants with missing data, 2) diagnosing causes, and 3) sending messages to participants to prevent further issues. We provided a one-week CSV-formatted dataset collected from the initial data collection campaign, during which diverse issues such as server, turning off smartphones, and non-responsiveness to self-reports were observed. Each researcher had 20 minutes for each condition to perform three tasks using randomly selected one-day dataset. For the Python plotting condition, we used a Google Colab [11] and asked them to write a contact message via a smartphone messenger.

After completing the tasks, we inquired about their utilization of design components of the final prototype and assessed its usability. They responded to the PSSUQ (Post-study system usability questionnaire) [32] and a semi-structured interview to assess usability, system use, and expectations. Two of the authors conducted a thematic analysis to categorize the interview data into major themes, and user experiences of the system's key features. The analysis followed an inductive approach, akin to the qualitative data analysis used in formative studies (see Section 3.1). Participants were compensated with $22 USD.

4.4.3 Study results: Usability and user experiences. For the PSSUQ score, researchers rated the best usability in DataSentry Full version (M = 4.92, SD = 0.76), followed by Basic version (M = 4.38, SD = 0.89) and Python plotting (M = 2.08, SD = 1.03). In the following, we share user experiences with the system's key features.

Overviewing missing data and diligence of participants. Researchers acknowledged that the features provided by the final system are crucial for detecting, diagnosing, and addressing missing data. They believed that such a system would "*reduce the burden of manually plotting and detecting missing data*" (R17), allowing them to "*promptly grasp the overall pattern of missing data*" (R32). One of the interesting aspects was that 13 researchers thought the overview of missing data collection could serve as an indicator of compliance or 'diligence' of participants. R30, who was responsible for mobile data collection and participant management in the IT industry, commented, "*What I liked about the overview page was how easy it was to spot participants who kept their bars filled using the visualization. Keeping track of participant diligence has always been a key part of our data collection efforts, and I think this visualization would be helpful for monitoring diligence and motivating participants to stay on track.*"

Detection and diagnosis of missing data by within-/between-participant variability. Fifteen researchers found the within-/between-participant comparison feature in the DataSentry Full version useful for detecting and diagnosing missing data. R14, who found the within-/between-participant variation lens particularly insightful, commented, "*I think this lens gives a helpful summary of* *missing data that would be hard to spot just by looking at raw time-series. By quickly brushing over the empty periods, I could tell if the missing data was an issue, both within and between participants.*" R37 combined the within-/between-participant time-series comparisons to understand the missing segments: "*There was a long, continuous missing period in a participant's location data. When I compared it to other participants' time-series, and even to the participant's data, the missing segment stood out as unusually long. This made me think that the missing data wasn't due to the participant not moving, but rather because the sensor might have malfunctioned.*" Beyond recognizing the usefulness of these features, researchers also drew inspiration from them and proposed ways to streamline missing data detection and diagnosis. Based on these variabilities, seven researchers suggested a missing data recommendation feature. R25 suggested, "*What I found most interesting in this system was how it identified missing data by detecting and diagnosing deviations from participants' typical sensing patterns. Building on this, it would be efficient if the system could flag such deviations and recommend them to researchers for further investigation.*"

Streamlining communication via rule-based supports. Thirteen researchers appreciated not only the ability to detect and diagnose missing data but also the option to send communications based on the diagnosis. These features were highly valued by student researchers, who often work alone to communicate with participants. Industry professionals, who frequently encounter diverse data collection scenarios, found rule-based communication support particularly useful, and expressed a desire to diversify these rules to easily contact participants. R34 mentioned, "*I appreciated the ability to set rules and contact relevant participants. If I could define conditions for sending communications based on various rules - such as the number of rows, the length of missing periods, or the specific context of missing data - I would want to integrate it into the data collection campaigns our team will conduct.*"

Our final prototype proved valuable for detecting and diagnosing missing data by considering both within- and between-participant variability. Furthermore, it streamlined communication with participants, addressing the challenges and burdens associated with existing missing data management procedures.

5 Discussion

In this section, we discuss the design implications for missing data management in the mobile sensor data collection (5.1 and 5.2) and lessons learned from the iterative design of DataSentry.

5.1 Detection and Diagnosis of Missing Data Considering Within- and Between-participant Variability

We introduced the prototype developed through an iterative design process and its evolution into the final system, DataSentry—a data management system supporting the detection, diagnosis, and addressing of missing data in mobile sensor data collection. By incorporating feedback from system deployments, we aligned the system closely with researchers' practical needs, demonstrating that DataSentry can reduce the burden of missing data management.

A key insight from our work is the critical role of understanding within- and between-participant variability to detect and diagnose participant- or system-related issues reported in prior studies [15, 29, 30]. The overview of missing data and the within-/between-participant variation lens aided in detecting participants with significantly long missing data. Furthermore, the ability to inspect raw sensor streams from various perspectives (e.g., within- or between-participant time-series comparisons) facilitated an understanding of the context and a diagnosis of the root causes behind the issues. This approach distinguishes DataSentry from previous mobile sensing monitoring systems, instead of solely relying on simple metrics [47, 54] or individual sensor streams [16], DataSentry incorporates the inherent within- and between-participant variability in event-based sensing data, enabling the detection and diagnosis of missing data at both overview and fine-grained levels.

To advance this process, integrating the detection and diagnosis feature based on within- and between-participant variability offers a promising design opportunity. Drawing on previous studies on modeling human routine behaviors [1, 63], the system can detect and diagnose missing data by identifying deviations from common behavioral routines. For example, the system can consolidate multiple sensor streams to understand *within-person sensing routines* (e.g., frequent logging of GPS and physical activity data along a participant's commuting path) and *between-person sensing routines* (e.g., sparse GPS and physical activity logging on weekend mornings across a group of participants). Leveraging these spatial-temporal contextual patterns, the system can flag instances of missing data that deviate from common routines (e.g., no GPS logs during a participant's usual weekday commute) and provide potential causes (e.g., sudden sensor malfunctions). This approach could be applied practically to existing mobile data collection practices [34, 66], to offer a deeper understanding of missing data.

5.2 Enhancing the Expressiveness of Missing Data Management Rules

One of the findings regarding DataSentry is that researchers wanted to define diverse rules related to missing data and communicate with participants based on these rules. The rule-based communication feature in DataSentry aligns with the concept of trigger-action programming [31, 67], where 'trigger rules' (e.g., survey count less than ten times a day) are associated with corresponding 'action' (e.g., send emails to participants). While the current system handles simple rules based on data counts, one viable improvement is to enhance the *expressiveness* of rules.

Trigger rules can be set based on raw item counts or simple descriptive statistics (e.g., mean, median). To enable flexible rule setting, it would be useful to incorporate semantically meaningful predicates. For example, GPS data can be analyzed to automatically identify significant places, such as home and work [65]. Users can define a range of values, such as dividing the time of day into work hours (9 AM–5 PM), off-work hours (5–10 PM), and sleep hours (10 PM–6 AM). Considering within-subject variations, researchers can further define a specific period for data aggregation or allow for setting user-specific thresholds (e.g., GPS count is 2 SDs smaller than the mean of the last week).

In addition, similar to the approach discussed by Jiang et al. [21], researchers could establish rules about missing data by combining multiple sensors with AND/OR conditions, enabling actions to be triggered when specific conditions are met. For instance, consider a scenario where a participant's GPS data frequently goes missing despite indications of regular movement from other sensors. An expressive rule could be structured as:

> **IF** ((A participant's GPS data is missing for more than 3 hours between 9 AM to 10 PM) **AND** (A participant's physical activity and accelerometer data are periodically collected))
> **THEN** (Send a message to the participant: "*Your GPS sensor data is missing. Please check if the sensor collector is turned on in your smartphone. If not, please report this issue to our research team.*")

The triggering frequency could be customized based on the *severity* of the missing data issues (e.g., once a day for severe issues vs. once a week for trivial ones). By defining a set of expressive rules, researchers can delegate repetitive and routine tasks to the system, allowing them to focus on more complex missing data issues that require their expertise and decision-making.

5.3 Multi-year, Iterative Design Process through In-the-wild Deployment

Rather than integrating all capabilities at once, the iterative process enabled the progressive expansion of the prototype's capabilities. Through a series of testing and refinements, this approach enabled the gradual fulfillment of challenging requirements [3, 5].

One of the significant advantages of this multi-year, in-the-wild deployment was the ability to uncover unexpected feedback from real-world use. For instance, in the first design iteration, we expected that a statistical quality control mechanism would be effective. However, field deployment revealed its limited practicality, prompting us to shift focus towards a detailed examination of raw sensor data. The iterative deployments also allowed us to test the system across diverse data collection contexts. While the first and third campaigns used a single data collection app, the second campaign involved two data collection apps and a smart speaker to collect self-report ESM. In the campaign, DataSentry played a crucial role in verifying data sync between different devices and apps by letting researchers examine the raw sensor data within the "Detail of One Participant" view.

The in-the-wild deployment and iterative design process was instrumental in uncovering and addressing real-world issues that might have been overlooked in the lab. These observations ensured the system's practical feasibility and usability, aligned with findings from previous HCI studies adopting multi-year iterative design approach [14, 44]. We envision that, although this approach requires substantial efforts, it provides a viable strategy for researchers and designers developing similar data management systems.

5.4 Interoperability and Generalizability

DataSentry supports monitoring of various sensor data and is interoperable for time-series mobile data from many people and sensors collected via various data collection frameworks. In addition to the framework used in our field trials, it supports data collected from other frameworks such as AWARE [8]. The minimum required data schema consists of the user ID, timestamp, and sensor data type. Once the data schema is properly configured, DataSentry automatically identifies the scale of each column, such as qualitative or quantitative, for visualization. Beyond the data types used in the current study, DataSentry can be generalizable to visualize time-series data collected from different devices, such as wearable devices. Researchers can comprehensively monitor multimodal data from many people and many sensors by simply importing data from different devices with a proper data schema definition.

5.5 Limitations and Future Work

One limitation of our study is that the system encountered scalability issues as the number of participants increased on a single screen leading to performance slowdowns, making the system less responsive. To address the issue, we could explore improvements to the current visualization method. For instance, prioritizing the display of participants with severe missing data issues or data types, while presenting less critical data only when the user requests it. Furthermore, the system could be enhanced to allow real-time monitoring to help promptly detect and address missing data issues. Currently, we traded this feature for interoperability by letting researchers periodically run a script that exports data from the database and imports it into DataSentry. Additional development could be easily made to enable real-time monitoring by directly tapping into each researcher's database. Lastly, the controlled user study was considered as the final evaluation due to the logistical challenges of coordinating with sufficient numbers of in-the-wild data collection campaigns, which demand significant time and effort. Our design has been iteratively validated and improved through three rounds of in-the-wild evaluation. Nevertheless, a large-scale, in-the-wild user study will still be beneficial for validating our final system.

6 Conclusion

We conducted a multi-year, iterative design process to build a data management system for missing data issues during mobile sensor data collection. Our approach integrated researchers' practical needs, enabling DataSentry to detect, diagnose, and mitigate missing data while considering within- and between-participant variability. Findings from the final user study suggest that the DataSentry has the potential to lower the burden of missing data management. Lessons learned from our design process provided several implications for advancing missing data management systems in mobile sensor data collection. We envision that DataSentry can be adapted for a wide range of in-the-wild mobile sensor data collection scenarios, ultimately reducing the workload for domain researchers.

Acknowledgments

This research was supported in part by the Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Korean government (MSIT) (2022R1A2C2011536) and the Institute of Information and Communications Technology Planning and Evaluation (IITP) Grant funded by the Korean Government (MSIT), Artificial Intelligence Graduate School Program, Yonsei University, under Grant RS-2020-II201361.

References

[1] Nikola Banovic, Tofi Buzali, Fanny Chevalier, Jennifer Mankoff, and Anind K Dey. 2016. Modeling and understanding human routine behavior. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*. 248–260.

[2] Virginia Braun and Victoria Clarke. 2012. *Thematic analysis.* American Psychological Association.

[3] Bradley Camburn, Vimal Viswanathan, Julie Linsey, David Anderson, Daniel Jensen, Richard Crawford, Kevin Otto, and Kristin Wood. 2017. Design prototyping methods: state of the art in strategies, techniques, and guidelines. *Design Science* 3 (2017), e13.

[4] Daniel Carruth, Julie Baca, Christopher Lewis, and Michael Stephens. 2024. Bridging Understanding: A Multi-year Study of the Effect of Aesthetics, Usability and User Domain Knowledge on Interpreting Scientific Visualization. In *International Conference on Human-Computer Interaction*. Springer, 3–13.

[5] Edward James Christie, Daniel D Jensen, Richard T Buckley, Devin A Menefee, Kyle Kenneth Ziegler, Kristin L Wood, and Richard H Crawford. 2012. Prototyping strategies: literature review and identification of critical variables. In *2012 ASEE Annual Conference & Exposition*. 25–1091.

[6] Hari G Dandapani, Natalie M Davoodi, Lucie C Joerg, Melinda M Li, Daniel H Strauss, Kelly Fan, Talie Massachi, and Elizabeth M Goldberg. 2022. Leveraging mobile-based sensors for clinical research to obtain activity and health measures for disease monitoring, prevention, and treatment. *Frontiers in Digital Health* 4 (2022), 893070.

[7] Robert F Dickerson, Eugenia I Gorlin, and John A Stankovic. 2011. Empath: a continuous remote emotional health monitoring system for depressive illness. In *Proceedings of the 2nd Conference on Wireless Health*. 1–10.

[8] Denzil Ferreira, Vassilis Kostakos, and Anind K Dey. 2015. AWARE: mobile context instrumentation framework. *Frontiers in ICT* 2 (2015), 6.

[9] Stephen Few. 2006. *Information dashboard design: The effective visual communication of data.* O'Reilly Media, Inc.

[10] Daniel Fulford, Jasmine Mote, Rachel Gonzalez, Samuel Abplanalp, Yuting Zhang, Jarrod Luckenbaugh, Jukka-Pekka Onnela, Carlos Busso, and David E Gard. 2021. Smartphone sensing of social interactions in people with and without schizophrenia. *Journal of Psychiatric Research* 137 (2021), 613–620.

[11] Google. 2024. Google Colaboratory. https://colab.research.google.com/ Accessed: 2024-09-08.

[12] Andreas Grammenos, Cecilia Mascolo, and Jon Crowcroft. 2018. You are sensing, but are you biased? a user unaided sensor calibration approach for mobile sensing. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 2, 1 (2018), 1–26.

[13] Gabriella M Harari, Nicholas D Lane, Rui Wang, Benjamin S Crosier, Andrew T Campbell, and Samuel D Gosling. 2016. Using smartphones to collect behavioral data in psychological science: Opportunities, practical considerations, and challenges. *Perspectives on Psychological Science* 11, 6 (2016), 838–854.

[14] Gillian R Hayes and Stephen W Hosaflook. 2013. HygieneHelper: promoting awareness and teaching life skills to youth with autism spectrum disorder. In *Proceedings of the 12th international conference on interaction design and children*. 539–542.

[15] Jennifer Healey, Lama Nachman, Sushmita Subramanian, Junaith Shahabdeen, and Margaret Morris. 2010. Out of the lab and into the fray: Towards modeling emotion in everyday life. In *Pervasive Computing: 8th International Conference, Pervasive 2010, Helsinki, Finland, May 17-20, 2010. Proceedings 8*. Springer, 156–173.

[16] Netzahualcóyotl Hernández, Bert Arnrich, Jesús Favela, Cem Ersoy, Burcu Demiray, and Jesús Fontecha. 2019. A multi-site study on walkability, data sharing and privacy perception using mobile sensing data gathered from the mk-sense platform. *Journal of Ambient Intelligence and Humanized Computing* 10 (2019), 2199–2211.

[17] John Hicks, Nithya Ramanathan, Donnie Kim, Mohamad Monibi, Joshua Selsky, Mark Hansen, and Deborah Estrin. 2010. AndWellness: an open mobile system for activity and experience sampling. In *Wireless Health 2010*. 34–43.

[18] Karen Hovsepian, Mustafa Al'Absi, Emre Ertin, Thomas Kamarck, Motohiro Nakajima, and Santosh Kumar. 2015. cStress: towards a gold standard for continuous stress assessment in the mobile environment. In *Proceedings of the 2015 ACM international joint conference on pervasive and ubiquitous computing*. 493–504.

[19] Gary Hsieh, Ian Li, Anind Dey, Jodi Forlizzi, and Scott E Hudson. 2008. Using visualizations to increase compliance in experience sampling. In *Proceedings of the 10th international conference on Ubiquitous computing*. 164–167.

[20] IBM Cloud Internet of Things. 2024. IBM Cloud Internet of Things. https://internetofthings.ibmcloud.com/. Accessed: 2024-08-26.

[21] Xiaodong Jiang, Nicholas Y Chen, Jason I Hong, Kevin Wang, Leila Takayama, and James A Landay. 2004. Siren: Context-aware computing for firefighting. In *Pervasive Computing: Second International Conference, PERVASIVE 2004, Linz/Vienna, Austria, April 21-23, 2004. Proceedings 2*. Springer, 87–105.

[22] Eunice Jun, Gary Hsieh, and Katharina Reinecke. 2017. Types of motivation affect study selection, attention, and dropouts in online experiments. *Proceedings of the ACM on Human-Computer Interaction* 1, CSCW (2017), 1–15.

[23] Soowon Kang, Woohyeok Choi, Cheul Young Park, Narae Cha, Auk Kim, Ahsan Habib Khandoker, Leontios Hadjileontiadis, Heepyung Kim, Yong Jeong, and Uichin Lee. 2023. K-emophone: A mobile and wearable dataset with in-situ emotion, stress, and attention labels. *Scientific data* 10, 1 (2023), 351.

[24] Kleomenis Katevas, Katrin Hänsel, Richard Clegg, Ilias Leontiadis, Hamed Haddadi, and Laurissa Tokarchuk. 2019. Finding dory in the crowd: Detecting social interactions using multi-modal mobile sensing. In *Proceedings of the 1st Workshop on Machine Learning on Edge in Sensor Systems*. 37–42.

[25] Soban Ahmed Khan, Asma Ahmad Farhan, Labiba Gillani Fahad, and Syed Fahad Tahir. 2020. Personal productivity monitoring through smartphones. *Journal of Ambient Intelligence and Smart Environments* 12, 4 (2020), 327–341.

[26] Inyeop Kim, Rihun Kim, Heepyung Kim, Duyeon Kim, Kyungsik Han, Paul H Lee, Gloria Mark, and Uichin Lee. 2019. Understanding smartphone usage in college classrooms: A long-term measurement study. *Computers & Education* 141 (2019), 103611.

[27] Tim Kuhlmann, Pablo Garaizar, and Ulf-Dietrich Reips. 2021. Smartphone sensor accuracy varies from device to device in mobile research: The case of spatial orientation. *Behavior research methods* 53 (2021), 22–33.

[28] Devender Kumar, Steven Jeuris, Jakob E Bardram, and Nicola Dragoni. 2020. Mobile and wearable sensing frameworks for mHealth studies and applications: A systematic review. *ACM Transactions on Computing for Healthcare* 2, 1 (2020), 1–28.

[29] Fanny Larradet, Radoslaw Niewiadomski, Giacinto Barresi, and Leonardo S Mattos. 2019. Appraisal theory-based mobile app for physiological data collection and labelling in the wild. In *Adjunct Proceedings of the 2019 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proceedings of the 2019 ACM International Symposium on Wearable Computers*. 752–756.

[30] Hyunsoo Lee, Soowon Kang, and Uichin Lee. 2022. Understanding privacy risks and perceived benefits in open dataset collection for mobile affective computing. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 6, 2 (2022), 1–26.

[31] Nicola Leonardi, Marco Manca, Fabio Paternò, and Carmen Santoro. 2019. Trigger-action programming for personalising humanoid robot behaviour. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*. 1–13.

[32] James R. Lewis. 1995. IBM Computer Usability Satisfaction Questionnaires: Psychometric Evaluation and Instructions for Use. In *International Journal of Human-Computer Interaction*, Vol. 7. Taylor & Francis, 57–78. doi:10.1080/10447319509526110

[33] Gonzalo J Martinez, Stephen M Mattingly, Pablo Robles-Granda, Koustuv Saha, Anusha Sirigiri, Jessica Young, Nitesh Chawla, Munmun De Choudhury, Sidney D'Mello, Gloria Mark, et al. 2021. Predicting participant compliance with fitness tracker wearing and ecological momentary assessment protocols in information workers: Observational study. *JMIR mHealth and uHealth* 9, 11 (2021), e22218.

[34] Abhinav Mehrotra, Veljko Pejovic, Jo Vermeulen, Robert Hendley, and Mirco Musolesi. 2016. My phone and me: understanding people's receptivity to mobile notifications. In *Proceedings of the 2016 CHI conference on human factors in computing systems*. 1021–1032.

[35] Mike Bostock. 2023. D3.js – Data-Driven Documents. https://d3js.org/. Accessed: 2024-09-12.

[36] Shayan Mirjafari, Kizito Masaba, Ted Grover, Weichen Wang, Pino Audia, Andrew T Campbell, Nitesh V Chawla, Vedant Das Swain, Munmun De Choudhury, Anind K Dey, et al. 2019. Differentiating higher and lower job performers in the workplace using mobile sensing. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 3, 2 (2019), 1–24.

[37] Xiaoyun Mo, Dianxi Shi, Ruosong Yang, Han Li, ZheHang Tong, and Feng Wang. 2015. A framework of fine-grained mobile sensing data collection and behavior analysis in an energy-configurable way. In *2015 IEEE International Conference on Smart City/SocialCom/SustainCom (SmartCity)*. IEEE, 391–398.

[38] Douglas C Montgomery. 2020. *Introduction to statistical quality control*. John Wiley & Sons.

[39] Barr Moses, Lior Gavish, and Molly Vorwerck. 2022. *Data quality fundamentals*. " O'Reilly Media, Inc.".

[40] movisens. 2024. movisensXS: Experience Sampling, EMA and Ambulatory Assessment. https://www.movisens.com/en/products/movisensxs/. Accessed: 2024-08-26.

[41] Subigya Nepal, Shayan Mirjafari, Gonzalo J Martinez, Pino Audia, Aaron Striegel, and Andrew T Campbell. 2020. Detecting job promotion in information workers using mobile sensing. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 4, 3 (2020), 1–28.

[42] Jukka-Pekka Onnela and Scott L Rauch. 2016. Harnessing smartphone-based digital phenotyping to enhance behavioral and mental health. *Neuropsychopharmacology* 41, 7 (2016), 1691–1696.

[43] Joonyoung Park and Uichin Lee. 2023. Understanding disengagement in just-in-time mobile health interventions. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 7, 2 (2023), 1–27.

[44] Caroline Pitt, Adam Bell, Edgar Onofre, and Katie Davis. 2019. A badge, not a barrier: Designing for-and throughout-digital badge implementation. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*. 1–14.

[45] Jenny S Radesky, Heidi M Weeks, Rosa Ball, Alexandria Schaller, Samantha Yeo, Joke Durnez, Matthew Tamayo-Rios, Mollie Epstein, Heather Kirkorian, Sarah Coyne, et al. 2020. Young children's use of smartphones and tablets. *Pediatrics* 146, 1 (2020).
[46] Md Mahbubur Rahman, Rummana Bari, Amin Ahsan Ali, Moushumi Sharmin, Andrew Raij, Karen Hovsepian, Syed Monowar Hossain, Emre Ertin, Ashley Kennedy, David H Epstein, et al. 2014. Are we there yet? Feasibility of continuous stress assessment via wireless physiological sensors. In *Proceedings of the 5th ACM Conference on Bioinformatics, Computational Biology, and Health Informatics*. 479–488.
[47] Jesus Ramos-Monteon, Luis A Castro, Luis-Felipe Rodriguez, and Oresti Banos. 2018. Incense iot: A collective sensing system for behavior data in shared spaces. In *Proceedings*, Vol. 2. MDPI, 1206.
[48] Yatharth Ranjan, Zulqarnain Rashid, Callum Stewart, Pauline Conde, Mark Begale, Denny Verbeeck, Sebastian Boettcher, Richard Dobson, Amos Folarin, RADAR-CNS Consortium, et al. 2019. RADAR-base: open source mobile health platform for collecting, monitoring, and analyzing data using sensors, wearables, and mobile devices. *JMIR mHealth and uHealth* 7, 8 (2019), e11734.
[49] RAPIDS Science Team. 2024. Data Quality Visualizations. https://www.rapids.science/1.10/visualizations/data-quality-visualizations/ Accessed: 2024-12-09.
[50] React Contributors. 2023. React: A JavaScript library for building user interfaces. https://reactjs.org/. Accessed: 2024-09-12.
[51] Nithya Sambasivan, Shivani Kapania, Hannah Highfill, Diana Akrong, Praveen Paritosh, and Lora M Aroyo. 2021. "Everyone wants to do the model work, not the data work": Data Cascades in High-Stakes AI. In *proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*. 1–15.
[52] Hillol Sarker, Matthew Tyburski, Md Mahbubur Rahman, Karen Hovsepian, Moushumi Sharmin, David H Epstein, Kenzie L Preston, C Debra Furr-Holden, Adam Milam, Inbal Nahum-Shani, et al. 2016. Finding significant stress episodes in a discontinuous time series of rapidly varying mobile sensor data. In *Proceedings of the 2016 CHI conference on human factors in computing systems*. 4489–4501.
[53] Tableau Software. 2024. *Tableau Software*. Tableau Software, LLC. https://www.tableau.com/ Version 2024..
[54] Poorna Talkad Sukumar, Thomas Breideband, Gonzalo J Martinez, Megan Caruso, Sierra Rose, Cooper Steputis, Sidney D'Mello, Gloria Mark, and Aaron Striegel. 2021. Designing an interactive visualization system for monitoring participant compliance in a large-scale, longitudinal study. In *Extended Abstracts of the 2021 CHI Conference on Human Factors in Computing Systems*. 1–8.
[55] Hongsuda Tangmunarunkit, Cheng-Kang Hsieh, Brent Longstaff, S Nolen, John Jenkins, Cameron Ketcham, Joshua Selsky, Faisal Alquaddoomi, Dony George, Jinha Kang, et al. 2015. Ohmage: A general and extensible end-to-end participatory sensing platform. *ACM Transactions on Intelligent Systems and Technology (TIST)* 6, 3 (2015), 1–21.
[56] Yuri Tani, Shuichi Fukuda, Yuki Matsuda, Sozo Inoue, and Yutaka Arakawa. 2020. Workersense: Mobile sensing platform for collecting physiological, mental, and environmental state of office workers. In *2020 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops)*. IEEE, 1–6.
[57] Jing Xian Teo, Sonia Davila, Chengxi Yang, An An Hii, Chee Jian Pua, Jonathan Yap, Swee Yaw Tan, Anders Sahlén, Calvin Woon-Loong Chin, Bin Tean Teh, et al. 2019. Digital phenotyping by consumer wearables identifies sleep-associated markers of cardiovascular disease risk and biological aging. *Communications Biology* 2, 1 (2019), 361.
[58] John Torous and Adam C Powell. 2015. Current research and trends in the use of smartphone applications for mood disorders. *Internet Interventions* 2, 2 (2015), 169–173.
[59] John Torous, Hannah Wisniewski, Bruce Bird, Elizabeth Carpenter, Gary David, Eduardo Elejalde, Dan Fulford, Synthia Guimond, Ryan Hays, Philip Henson, et al. 2019. Creating a digital health smartphone app and digital phenotyping platform for mental health and diverse healthcare needs: an interdisciplinary and collaborative approach. *Journal of Technology in Behavioral Science* 4 (2019), 73–85.
[60] Niels Van Berkel, Jorge Goncalves, Simo Hosio, and Vassilis Kostakos. 2017. Gamification of mobile experience sampling improves data quality and quantity. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 1, 3 (2017), 1–21.
[61] Niels van Berkel, Jorge Goncalves, Peter Koval, Simo Hosio, Tilman Dingler, Denzil Ferreira, and Vassilis Kostakos. 2019. Context-informed scheduling and analysis: improving accuracy of mobile self-reports. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*. 1–12.
[62] Rui Wang, Weichen Wang, Alex DaSilva, Jeremy F Huckins, William M Kelley, Todd F Heatherton, and Andrew T Campbell. 2018. Tracking depression dynamics in college students using mobile phone and wearable sensing. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 2, 1 (2018), 1–26.
[63] Xuhai Xu, Prerna Chikersal, Afsaneh Doryab, Daniella K Villalba, Janine M Dutcher, Michael J Tumminia, Tim Althoff, Sheldon Cohen, Kasey G Creswell, J David Creswell, et al. 2019. Leveraging routine behavior and contextually-filtered features for depression detection among college students. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 3, 3 (2019), 1–33.
[64] Xuhai Xu, Jennifer Mankoff, and Anind K Dey. 2021. Understanding practices and needs of researchers in human state modeling by passive mobile sensing. *CCF Transactions on Pervasive Computing and Interaction* 3 (2021), 344–366.
[65] Yang Ye, Yu Zheng, Yukun Chen, Jianhua Feng, and Xing Xie. 2009. Mining individual life pattern based on location history. In *2009 tenth international conference on mobile data management: Systems, services and middleware*. IEEE, 1–10.
[66] Fengpeng Yuan, Xianyi Gao, and Janne Lindqvist. 2017. How busy are you? Predicting the interruptibility intensity of mobile users. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*. 5346–5360.
[67] Valerie Zhao, Lefan Zhang, Bo Wang, Michael L Littman, Shan Lu, and Blase Ur. 2021. Understanding trigger-action programs through novel visualizations of program differences. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*. 1–17.

A APPENDIX

ID	Position	Experience of Mobile Sensor Data Collection	Experienced Missing Data	Systems to Check Missing Data
1	PhD (4th-year)	Collecting mobile sensor data (physical activity, call log, location, screen log, etc.) with self-reports (stress, depression) to capture depressive episodes (2 campaigns, 30 participants each)	Missing periods in screen log and physical activity data, simultaneous missing periods of all sensor data, low number of self-reports	Downloading raw data files from Firebase and plotting them using Python script
2	PhD (3rd-year)	Collecting mobile sensor data (WiFi, call log, app usage, screen logs, etc.) with self-reports (stress) from 65 participants to detect stress in daily lives	Simultaneous missing periods of all sensor data, low number of self-reports	Downloading raw data from database and plotting using Python, developing tool to check counts
3	MS (1st-year)	Collecting mobile sensor data (app usage duration and location) with self-reports (stress) from 40 participants to model stress prediction	Missing periods in location, app usage data, entirely missing self-report surveys of specific participants	Downloading raw data from database and plotting them using Python script, reviewing tabular data using Excel
4	PhD (3rd-year)	Collecting mobile sensor data (app usage, physical activity, location, etc.) with self-reports (current activities) autistic individuals (two data collection campaigns, 50 and 20 participants for each)	Simultaneous missing periods of all sensor data, missing periods in physical activity data, low number of self-reports	Downloading raw CSV data from Firebase and plotting them using Python script
5	PhD (4th-year)	Collecting and publishing in-the-wild, longitudinal mobile sensor data (physical activity, call log, WiFi, bluetooth, location, etc.) with self-report (current emotion) from 100 participants	Missing periods in location, WiFi, and physical activity data, simultaneous missing of all sensor data, low number of self-reports	Downloading raw CSV data from database and plotting them using Python script, reviewing tabular data using Excel
6	PhD (2nd-year)	Collecting mobile sensor data (app usage, physical activity, camera) and self-report (emotion, depression) to figure out behavioral and contextual factors affecting mental well-being from 20 participants	Missing periods in app usage data	Downloading raw data from database and plotting them using Python script
7	PhD (3rd-year)	Collecting mobile (location, app usage, call log), wearable (HR, EDA), self-reports (stress, depression) data to model relationships between behavior and mental health status	Missing periods in location data, entirely missing self-report surveys of specific participants	Downloading raw json files from database and plotting them using Python script, reviewing tabular data

Table 1: Demographics of formative study researchers.

ID	Position	Experience of Mobile Sensor Data Collection
8	PhD (4th-year)	(first campaign) Collecting mobile and wearable sensor data to understand emotional changes
9	PhD (1st-year)	(first campaign) Participating in mobile sensor data collection campaign
10	PhD (2nd-year)	(second campaign) Collecting smartphone sensor data and smart speaker self-report ESM to analyze people's mental health in a smart home environment
11	MS (1st-year)	(second campaign) Collecting smartphone sensor data and smart speaker self-report ESM to analyze people's mental health in a smart home environment
12	PhD (4th-year)	(third campaign) Collecting mobile data and self-report to understand the causality between stress and contexts

Table 2: Demographics of deployment study researchers.

ID	Position	Experience of Mobile Sensor Data Collection
13	PhD (4th-year)	Collecting mobile sensor data in a classroom environments
14	PhD (2nd-year)	Collecting mobile sensor data related to physical activity-related data to recognize exercise states
15	MS (1st-year)	Collecting photo and audio data using a smartphone
16	MS (2nd-year)	Collecting emotional and daily life data from patients with ADHD and collecting geolocation data to track users' movement patterns
17	PhD (4th-year)	Collecting location and movement data for personality detection
18	MS (2nd-year)	Collecting smartphone sensor data related to app usage and background information of smartphone states (e.g., which app is running on the background)
19	MS (1st-year)	Collecting app usage data and voice recordings over a period of three weeks
20	PhD (3rd-year)	Collecting mobile sensor data for the daily lives of individuals
21	PhD (3rd-year)	Collecting sensor data from mobile devices to monitor the daily activities of people with autism
22	PhD (2nd-year)	Collecting mobile sensor data for human activity recognition
23	PhD (2nd-year)	Collecting everyday life data through mobile sensing for stress detection
24	PhD (4th-year)	Collecting mobile sensor data for human activity recognition
25	MS (2nd-year)	Collecting mobile sensor data from smartphones and audio recording data from smart speakers
26	PhD (4th-year)	Collecting mobile sensor data for depression detection
27	PhD (2nd-year)	Collecting mobile sensor data related to exercise or physical activity
28	Industry (5th-year)	Collecting health-related data and analyzing usage patterns of smartphone applications
29	Industry (3rd-year)	Collecting and analyzing mobile and wearable sensor data
30	Industry (4th-year)	Collecting diverse mobile sensing data from smartphones and IoT sensor data from smart home environment, monitoring and managing participants' compliance such as self-reports
31	Industry (25th-year)	Collecting health-related sensor data from smartphone and smartwatch, IoT sensor data from smart home environments
32	Industry (6th-year)	Collecting and analyzing smartphone sensor data and physiological signals from Fitbit sensors
33	Industry (5th-year)	Collecting health data and analyzing physical activity patterns from people's daily lives
34	Industry (3rd-year)	Collecting health data from smartphone application, specifically designed to monitor and analyze sleep behavioral patterns
35	Industry (4th-year)	Collecting daily activity data from individuals, alongside location information to monitor and analyze physical activity patterns
36	Industry (6th-year)	Collecting sleep data from smartphone to analyze relationships between mental health and sleep patterns
37	Industry (2nd-year)	Collecting health-related data from Apple Watch and smartphone, utilizing VR accessories to collect and analyze motion sensor data
38	Industry (4th-year)	Collecting gyroscope and accelerometer sensor data from smartphone in exhibition

Table 3: Demographics of user study researchers.