Navigating User-System Gaps: Understanding User-Interactions in User-Centric Context-Aware Systems for Digital Well-being Intervention

Inyeop Kim
KAIST
Daejeon, South Korea
inyeop@kse.kaist.ac.kr

Uichin Lee*
KAIST
Daejeon, South Korea
uclee@kaist.edu

ABSTRACT

In this paper, we investigate the challenges users face with a user-centric context-aware intervention system. Users often face gaps when the system's responses do not align with their goals and intentions. We explore these gaps through a prototype system that enables users to specify context-action intervention rules as they desire. We conducted a lab study to understand how users perceive and cope with gaps while translating their intentions as rules, revealing that users experience context-mapping and context-recognition uncertainties (instant evaluation cycle). We also performed a field study to explore how users perceive gaps and make adaptations of rules when the operation of specified rules in real-world settings (delayed evaluation cycle). This research highlights the dynamic nature of user interaction with context-aware systems and suggests the potential of such systems in supporting digital well-being. It provides insights into user adaptation processes and offers guidance for designing user-centric context-aware applications.

CCS CONCEPTS

• **Human-centered computing** → *Empirical studies in HCI*.

KEYWORDS

User-centric Context-aware Systems, Technical Gap, Context-triggered Actions, Digital Well-being Intervention

ACM Reference Format:
Inyeop Kim and Uichin Lee. 2024. Navigating User-System Gaps: Understanding User-Interactions in User-Centric Context-Aware Systems for Digital Well-being Intervention. In *Proceedings of the CHI Conference on Human Factors in Computing Systems (CHI '24), May 11–16, 2024, Honolulu, HI, USA.* ACM, New York, NY, USA, 15 pages. https://doi.org/10.1145/3613904.3641979

1 INTRODUCTION

The pervasive use of mobile devices has integrated technology into our lives on a personal level, enabling seamless interactions. Through mobile devices, it is possible to monitor users' daily contexts, such as location and activities. Consequently, context-aware mobile services that capture users' diverse situations and deliver services or information accordingly have been proposed in various domains [9, 16, 20, 22, 54]. Moreover, a user-centric approach has been suggested to cater to users' evolving goals based on contexts [4, 7, 14, 19, 66, 67]. In context-aware services embracing the user-centric approach, users are provided with an interface to define rules specifying contextual conditions for triggering system actions known as "context-triggered actions" [63].

According to Norman's interaction model [55], users generally expect their goal intentions (e.g., reducing social media usage) to be satisfied as they interact with a system, by formulating specific implementation intentions for behavior change (when, where, and how) [57] (e.g., limiting social media usage while studying at a library). However, users may perceive gaps when there is a disparity between the system's desired capabilities (e.g., complex activity tracking such as "studying") and what it can actually provide (e.g., simple activity tracking such as "still" or "moving") or when the system behaves differently from what users expected (e.g., activity tracking errors), failing to meet their implementation intentions. These gaps are inherent in the interaction between users and systems. The reason for these gaps lies in the fact that users have diverse and detailed requirements for implementation intentions [3, 27, 33, 53, 58] while the capabilities of the system are constrained by technological infrastructure, often dictating certain interactions between users and systems [15].

Furthermore, users' goals changing over time may widen the gap between user desires and system operations. For example, a user may rely on a digital wellbeing app that provides personalized recommendations for social media usage, but a failure to adapt to changes in the user's routine (e.g., during exam periods) could create a gap in delivering relevant interventions. If the users perceive a large gap, users might have negative experiences [15], resulting in service discontinuation or the adoption of alternatives. Therefore, understanding and addressing this gap is crucial.

User-centric context-aware services typically provide interfaces for users to allow specifying desired context-action rules to define system operations. However, predicting user desires and behaviors is highly challenging [31]. Additionally, catering to the diverse requirements of a heterogeneous set of users from various backgrounds and experiences poses a significant difficulty [27, 33, 53].

When defining context-based trigger action rules, users recognize that their implementation intentions cannot be expressed as they desire, leading them to explore alternative approaches to address the gap using system-provided configurable contexts and actions (i.e., instant evaluation). For example, users have an implementation intention of limiting their phone usage at specific rooms,

*Corresponding author

CHI '24, May 11–16, 2024, Honolulu, HI, USA

ACM ISBN 979-8-4007-0330-0/24/05
https://doi.org/10.1145/3613904.3641979

but the infrastructure only supports building-level location tracking due to localization constraints. Thus, an implementation intention of building-level locations (e.g., at a library) is alternatively used for behavior change. Another aspect of user-centric context-aware systems is their execution of actions based on user-defined contextual conditions [18]. Consequently, evaluating whether the system behaves as expected by the users is *delayed* until rule operation (i.e., delayed evaluation). If system behavior diverges from user expectations at a sensing level (e.g., not accurate) or a goal level (e.g., too tight), users may need to change their implementation intentions which necessitate adaptive rule modifications [70]. Through these interaction cycles with an user-centric rule-based system, users repeatedly encounter and address gaps.

We aim to understand how users perceive and address the gap between their goal/implementation intentions and the behaviors of the user-centric context-aware intervention system. We structured our research questions as follows: (1) How do users perceive and address gaps while specifying their implementation intentions as rules? (i.e., instant evaluation cycle), (2) How do users perceive and address gaps when the execution of specified rules? (i.e., delayed evaluation cycle). We selected *digital wellbeing intervention* as our case study domain to explore gaps in user-centric context-aware approaches given that users exhibit varied preferences for enhancing digital well-being [33, 37], leading to diverse goal and implementation intentions. This provides us with the opportunity to observe how users interact with the system, especially in setting up and managing rules that align with their diverse intentions. Also, we selected college students as our study participants considering the literature that highlights the significance of college students' digital well-being [62, 64, 69].

We conducted a technical review study, investigating user-generated scenarios expressed in natural language for digital well-being intervention from prior work [33]. Subsequently, we outlined the necessary support for context-action rules in digital well-being interventions and identified feasible technical provisions. Building upon this groundwork, we developed FocusAid, a straightforward context-aware mobile intervention system. FocusAid offers an interface for allowing users to set context-action rules for digital well-being interventions.

In our lab-based user study (n=12), we investigated the instant evaluation cycle by observing how users perceive and adaptively specify rules when given certain implementation intentions. This study revealed two uncertainties: context-mapping uncertainty and context-recognition uncertainty. We identified how these uncertainties correspond to the perceived gaps and the strategies participants employed to bridge gaps by formulating alternative rules. We additionally carried out a three-week field study (n=46) to delve into the delayed evaluation cycle. This study examined how users adaptively adjusted context-action rules when the execution of specified rules does not align with their goal/implementation intentions.

Our research offers multiple contributions to the HCI community. By extending the existing interaction model, we have deepened the understanding of user interactions in the context of mobile context-aware services. Our prototype development and case study have illuminated the nature of users' experiences with gaps in user-centric context-aware digital well-being interventions and their strategies for managing these gaps. Furthermore, our study provides practical design considerations that are specifically aimed at enhancing user experiences by effectively addressing these identified gaps.

2 BACKGROUND AND RELATED WORK

2.1 User-centric Context-aware Intervention for Well-being Behaviors

Context-aware interventions utilize users' contextual information to dynamically adapt and provide relevant information or services, aiming to promote users' wellbeing behaviors [12] across diverse domains, including alcoholism [22], smoking cessation [54], and weight management [20]. To effectively deliver context-aware interventions, service providers should determine which information or services to provide, when, and where by using users' contextual information [17].

Context-aware intervention can be classified into either system-centric or user-centric approaches depending on the customizability of intervention. In a system-centric approach, predefined intervention rules are embedded in the system. This approach entails continuous monitoring of user contexts or states, followed by intervention delivery based on pre-established rules [20, 22, 29, 34, 54]. However, its limitations include restricted customization, making it difficult to address the unique goals and requirements of individual users [27, 33, 42, 53]. In contrast, the user-centric approach empowers users with interfaces to customize context-action rules as their implementation intentions. For instance, Lee et al. [42] developed an end-user programmable application that supports the creation of simple rule-based context-aware interventions for improving sleep quality by integrating home automation sensors and prompting devices (e.g., wireless speakers and mobile phones). The user-centric approach, although potentially more intricate to develop than the system-centric approach, offers users the advantage of customizing interventions aligned with their evolving requirements. One of the representative user-centric approaches is trigger-action programming, which allows users to establish rules by connecting many possible events with desired action trigger conditions [44].

2.2 Addressing User Requirements for Technically Feasible Context-aware Intervention

In general, an interactive HCI system is implemented considering the needs of potential users [2]. Nevertheless, a gap emerges between the user requirements necessary to achieve their goals and the system's technical capabilities [3]. This gap can be attributed to the fine-grained and contextual nature of human behaviors [3, 27, 31, 33, 42, 53, 58]. Moreover, a system's dependence on the technological infrastructure upon which the system is built can cause gaps between user requirements and the system's functional scope [15]. As an example, developers have to implement a system interface within the boundaries defined by the technological infrastructure (e.g., Android API), resulting in the gap between user requirements and technical capability. Edwards et al. explained *constrained possibilities* where design choices taken by technological infrastructure may preclude supporting certain desirable user experiences [15]. Edwards et al. also indicated that when the low-level concept of the system is implemented as a user-facing feature, the user may not have a correct mental model because abstracted features become part of the conceptual model of the interface (i.e., *interjected abstractions*) [15]. These situations where gaps exist may shape negative user experience toward the system [15].

To address the gap, multiple strategies can be employed. One approach is a *first-order approximation*, wherein user requirements are partially met by approximating features that align with their objectives with a trade-off or workaround [3]. Another perspective is *seamful design* [8], which handles the technical gap by revealing and exploiting inevitable technical limitations in computing technology rather than hiding them from users. Additionally, explaining the system's configuration and state to users in an understandable way to overcome mismatches between users' mental models and the system's model of operation may help cope with the technical gap (i.e., *supporting intelligibility*) [13, 15, 46]. Further, supplying new/additional infrastructure technologies may be considered [15]. For example, integrating wearable device's sensors can be considered to infer a user's state that is difficult to detect only with sensors mounted on a mobile phone.

2.3 Modeling User Interactions for User-centric Context-aware Intervention

Norman's interaction model [55] serves as a valuable framework for exploring user-system interaction [1, 43]. This model includes three key concepts: (1) goal, (2) execution, and (3) evaluation [43, 55]. The goal represents what the user aims to achieve through the system. Execution involves navigating the system interface and taking specific actions to accomplish this goal. In the evaluation phase, users assess the system's feedback and how well it facilitates their goals. Pirolli et al. [57] highlighted a distinction between *goal intention*, the aim to achieve a specific behavior change, and *implementation intention*, the action plan to realize that goal. For instance, a user's goal (e.g., reducing social media usage) may lead to a set of implementation intentions (e.g., no usage while studying or limiting its use to no more than 30 minutes daily). In user-centric context-aware intervention systems, users perform execution by expressing their implementation intentions in the form of context-action rules through the interface that the system supports (or an interface model).

Each implementation intention should be expressed by the interface model. However, the expressiveness of the interface model might be limited to fully specify the user's intentions. For example, technically detecting when a user is studying using limited sensors can be challenging, making it difficult for a user to specify a rule like 'no social media usage while studying.' Upon interaction, users may quickly realize through the interface that it is not feasible to specify studying activities as a condition (i.e., instant evaluation). Consequently, users might alternatively set a study-related location (e.g., a library) as the context condition for triggering action (e.g., restricting social media usage). Hutchins et al. [28] suggested *semantic distance* to reflect the relationship between the user intentions and the meaning of expressions in the interface languages. In user-centric context-aware systems, semantic distance concerns how well the rules specified by users align with goal Intention as well as implementation intentions. Users perceive a gap when there exists semantic distance, meaning when the system interface does not support their intentions.

In typical user-centric context-aware systems, the interface language is based on the conversation metaphor [28], as users compose commands via the interface, which then prompts the system to operate context-action rules in the physical world. The interface acts as *an intermediary*, enabling implicit communication between the user and system, both instant and in a delayed manner. Users' evaluation of how well the specified rules operate in accordance with their intentions is delayed until the conditions defined in the rule are met, and the rule actually operates [18]. If users find the rule operations undesirable (i.e., perceiving a gap due to the existence of semantic distance), they may modify the rules [70]. Additionally, users' implementation intentions may change; for example, a university student approaching exams may wish to further limit their social media usage. In such cases, the previously specified rules need to be adapted to suit their current situation. Besides semantic distance, there is *articulatory distance* between the meaning of rules and their physical form due to a lack of articulatory similarity and directness in user-centric context-aware systems (e.g., actual operations of rules in real worlds are abstracted as visually presented context-aware rules in a user interface) [28]. The entire interaction process, including both instant and delayed evaluation cycles, is depicted in Figure 1.

2.4 Context-aware Digital Wellbeing Interventions

In digital well-being interventions, the primary focus is on users' technology use, aiming to mitigate the negative impacts of excessive and frequent usage and to improve productivity, mental health, and social interactions [60]. Numerous studies in digital wellbeing have proposed and evaluated methods and tools for self-controlling the use of mobile devices and specific applications, focusing on alleviating the negative effects on subjective well-being caused by digital overuse [60]. Research investigating tools developed for enhancing digital wellbeing shows that, while a range of intervention strategies (e.g., self-tracking, goal setting) exist, the most commonly employed strategy involves blocking or removing distractions through interaction restraint [50].

The effectiveness of intervention strategies in digital wellbeing is significantly influenced by diverse situational contexts (e.g., phone snubbing [32], distracted learning [35], walk safety risks [41], and problematic social media usage [25]). Consequently, prior studies have been conducted on *context-aware* digital wellbeing interventions, which provide interventions and evaluate effectiveness in contexts where unnecessary digital device usage could be problematic. For example, one study designed and evaluated a location-based service to assist students in regulating their mobile phone usage in classrooms [35]. Another study introduced a proactive context-aware system to tackle smartphone-induced distractions in study-related contexts for college students [36]. There have also been studies on context-aware systems that encourage limiting behavior among co-located users in group activities [39], provide interventions (e.g., feedback provision, removing the newsfeed) in situations where users problematically use Facebook [51], and

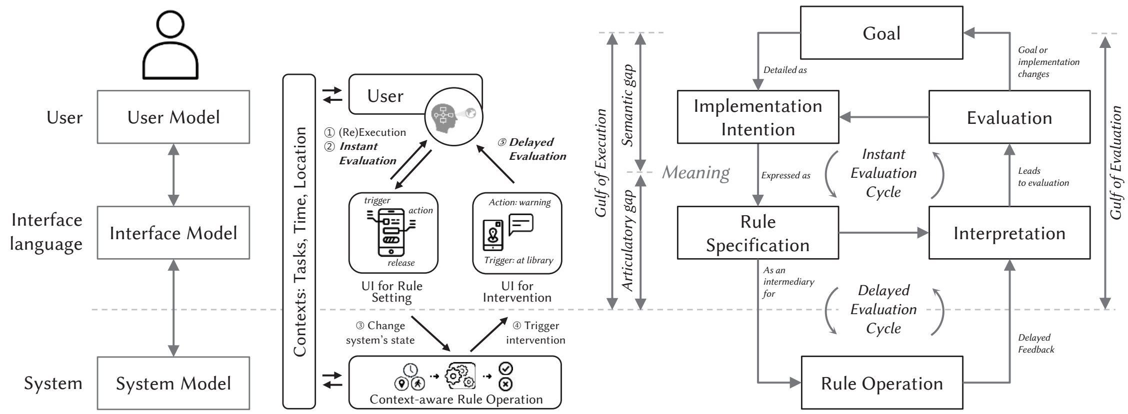


Figure 1: User interaction modeling with user, interface, and system layers for user-centric context-aware intervention. There are two different user interfaces, namely rule setting and intervention delivery, and there are three different models, namely a user's mental model, an interface model related to the interface language, and a system's model for rule operation. A user's goal is a set of implementation intentions, which are specified as rules. In this process, a user's interaction using the UI for rule setting involves an instant evaluation cycle. Context-aware rules are then executed by the system and interventions are delivered to the user via the intervention UI. A user interprets the feedback and evaluates whether current rules and system operation satisfy a user's implementation or goal intentions, which leads to changes in implementation or goal intentions.

detect transition points in the workplace and blocking distracting websites [65]. Also, one study proposed a system that monitors user phone usage behavior and proactively aids in learning users how to manage their smartphone usage through adaptable and continuously variable interventions [61].

It is crucial to consider not only the user's situational context but also the diversity of preferred coping strategies in context-aware digital well-being interventions [33]. This diversity illustrates that individuals have diverse implementation intentions for improving digital well-being. Considering this, we have selected context-aware digital well-being interventions as our case domain to explore gaps in user-centric context-aware approaches. The reason behind this choice is that interventions with a user-centric approach provide an interface enabling users to flexibly define system behavior to suit their implementation intentions. With a wide range of user preferences for enhancing digital well-being [33, 37], a user-centric approach facilitates observation of how individuals utilize the system, particularly in terms of rule setup and management. Furthermore, we aim to understand how users perceive and address the gap between their goal/implementation intentions and the system's behaviors.

There is a lack of prior studies on context-aware digital well-being interventions with fully customizable options for system behaviors [24, 37, 38, 40, 47, 52]. Users could set usage goals, but they lacked the ability to specify interaction restraints upon exceeding these goals or to define context-specific goals (e.g., limiting Facebook use to 30 minutes in the library). Hence, by employing this system, we expect to gain a comprehensive understanding of how users interact with and assess a user-centric context-aware digital intervention system that empowers them to tailor system behaviors for their digital well-being.

3 TECHNICAL REVIEW OF CONTEXT-ACTION RULES FOR DIGITAL WELL-BEING INTERVENTIONS

We conducted a technical review to implement a user-centric context-aware intervention system for digital well-being. Our work builds upon the prior work [33], which explored contexts and coping strategies for phone distraction management. In this work, we performed an in-depth analysis of the dataset used in their study, consisting of user-described context-action rules expressed in natural language for managing phone distractions. In particular, we explored how context-action rules can be technically supported within a mobile platform for digital well-being intervention.

In the prior work [33], participants were asked to articulate how they wished to manage phone distractions as context-action rules (e.g., "If I am in a library, silence all notifications"). A total of 216 rules were collected, revealing diverse requirements for managing phone distractions. The authors derived four components essential for context-aware phone distraction management: (1) trigger condition, (2) filtering condition, (3) action, and (4) action-releasing condition. Trigger conditions denote the contexts where users perceive phone distractions, filtering conditions determine which external interruptions (e.g., notifications) should be managed, actions describe how to limit interactions (e.g., blocking app usage), and action-releasing conditions specify when these actions should be released (e.g., when leaving a library). This work provided a conceptual framework for what should be supported in context-action rules for digital well-being interventions. However, further examination is needed to consider the technical feasibility of system implementation. Thus, we conducted a detailed analysis of the rules employed in the previous study and presented our decisions regarding the implementation of context-action rule-based systems for digital well-being interventions.

3.1 Supporting Trigger Contexts

The prior work [33] demonstrated that users predominantly specified context conditions related to location, activity, time, and social state as scenarios requiring intervention. We also identified that users expressed context conditions with varying levels of granularity (e.g., "Silence all notifications *after bedtime*" vs. "Set to silence all notifications *from 1:00 pm to 6:00 am*"). When users define context conditions with coarse granularity, the system may struggle to determine when to trigger actions effectively. Hence, it is essential for the system to guide users in *explicitly defining conditions that dictate when to initiate or release actions*.

We observed users using different types of context conditions interchangeably to describe the same situations. For instance, some users described sleeping as an activity context (e.g., "when I fall asleep"), while others considered it a time context (e.g., "from 1 pm to 6 am"). This indicates that the system can assist users in specifying situations alternatively by employing *generic types of context conditions* rather than accommodating every situation with unique conditions. For example, if users are in a library (a location context) and stationary (an activity context), we can infer that they are studying in the library.

Users also expressed the desire to control interventions in exceptional situations and requested the ability to *manually execute or override rules* (e.g., "Before going to bed, prevent app use with a warning message. However, it can be released when necessary"). This indicates that users do not wish to be rigidly bound by context rules but seek to have full control over interventions. Therefore, we concluded that the system should allow users to manually control a rule or temporarily release an intervention.

Context conditions can be classified as either *state* or *event*, each having distinct implications [26]. A state condition persists as long as a specific situation lasts, whereas an event condition can only be evaluated as true or false at a particular moment. Consequently, state and event conditions require different treatment when developing context-action rules. The previous work [33] highlighted that some users exclusively used event conditions in rule descriptions, introducing uncertainty [6, 26] since it is unclear when the system should release actions. This error, referred to as "missing reversal," can occur when users believe that actions will terminate automatically [26]. Managing this uncertainty is critical, as actions should be released when users no longer require intervention for digital well-being (e.g., unblocking specific apps). Furthermore, we noted that users often used state and event conditions interchangeably to describe equivalent concepts. This suggests that users can alternatively specify desired conditions using state conditions instead of event conditions. By guiding users to employ state conditions rather than event conditions, the system can potentially reduce uncertainty, as state conditions clarify when an intervention should begin and end. To address uncertainty issues, Huang et al. suggested the concept of *disallowing confusing options* [26]. In line with this approach, we allowed users to use only state conditions when specifying conditions.

3.2 Supporting Actions

The previous work [33] identified that users primarily articulated three types of actions for managing distractions: (1) restricting phone or app usage, (2) handling external interruptions (e.g., notifications), and (3) altering a phone's state (e.g., ringer mode). However, the detailed needs for each action differ across users: some users required simple functions (e.g., blocking specific apps or displaying a warning message), whereas others described functions that involved complex operations (e.g., allowing temporary usage after solving a complex arithmetic problem or entering a lengthy sequence of numbers). An increase in the complexity of system functions can hinder users' understanding of system functionality and result in reduced usability [19]. While offering these basic actions is considered important, implementing every requested function poses significant technical challenges. Consequently, we opted to implement representative and straightforward functions based on users' requirements, as shown later. We found some users prefer a more restrictive approach to limit app usage, while others opt for a less restrictive approach.

4 IMPLEMENTING CONTEXT-AWARE SYSTEMS FOR DIGITAL WELL-BEING INTERVENTION

We implemented FocusAid that allows users to set context-action rules for digital wellbeing. FocusAid also provides a rule-review function, which clarifies when rules are activated and deactivated, addressing issues of missing reversal errors and uncertainties regarding the operation of rules. FocusAid also offers a rule management feature that allows users to maintain control in exceptional situations and align interventions with their specific situational requirements (i.e., implementation intention).

4.1 Setting Context Conditions

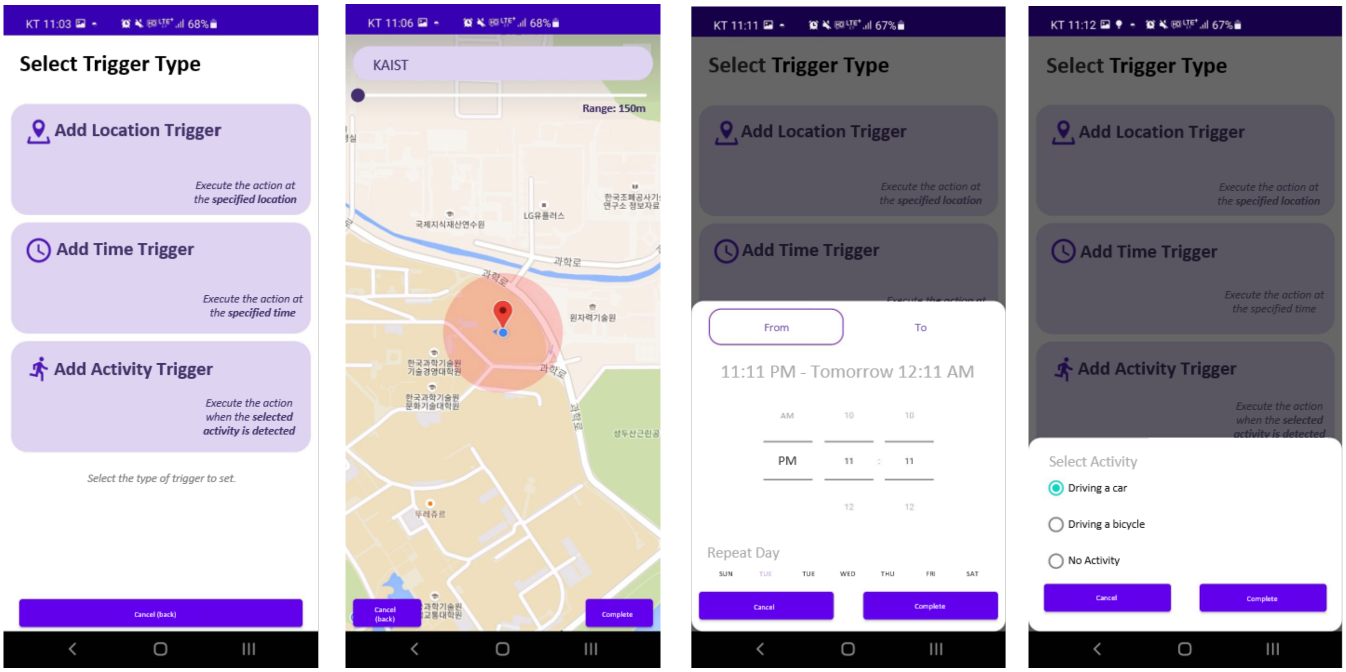
Based on our technical analysis, FocusAid offers three generic context conditions: (1) Location, (2) Time, and (3) Activity.

For clarity in triggering and *releasing conditions*, FocusAid restricts users to defining only state conditions, thereby eliminating any confusing options (i.e., disallowing confusing options).

Location Condition Setting: Users can define location conditions by utilizing a search bar to find specific places (Figure 2b). They can also set a radius around a selected location, with options ranging from 150m to 2,000m. Given potential GPS indoor tracking inaccuracies, the minimum threshold was set at 150m. This approach was informed by the user-defined automation application 'Bixby Routine' provided by Samsung.

Time Condition Setting: Users can define time conditions specifying when actions are triggered and released, including start and end times (i.e., state condition). Furthermore, users can configure recurring time conditions by selecting specific days of the week (Figure 2c).

Activity Condition Setting: FocusAid supports three primary activity conditions based on our technical review: (1) Driving, (2) Cycling, and (3) Still. The inclusion of the "Still" activity condition accommodates situations where users are expected to be in a stationary state, such as during activities requiring concentration, like studying [36].



(a) Selecting conditions **(b) Setting location** **(c) Setting time** **(d) Setting activity**

Figure 2: (a) Selecting conditions: Choose location, time, and activity context as conditions, (b) Setting location: Search for and set the range for the location where the action will be executed, (c) Setting time: Set the start and end times for when the action will occur and (d) Setting activity: Specify the activity during which the action will be executed.

4.2 Setting Actions

After establishing context conditions, users can define actions. FocusAid offers four primary action types, reflecting those frequently mentioned in user scenarios [33]: (1) Limiting app use, (2) Hiding notifications, (3) Activating Do Not Disturb (DND) mode, and (4) Changing ringer mode.

Limiting App Use: Users can select and designate apps for usage limitations. For each chosen app, users can set a usage goal (e.g., 10 minutes) and determine how app usage should be restricted when it exceeds the defined goal. Based on the technical review, we found some users prefer a more restrictive approach to limit app usage, while others opt for a less restrictive approach [33]. FocusAid provides both preferences by providing two straightforward options for restraining methods: (1) terminating an app and (2) displaying a warning message. If the "terminate an app" option is selected, FocusAid will freeze the phone screen, closing the app when the user selects the "OK" button. Alternatively, if the "display a warning message" option is chosen, FocusAid will show a dialog stating, "Allowed usage time for [App name] has exceeded." This approach allows users to continue using the app by selecting the "OK" button.

Hiding Notifications: Users can hide notifications by specifying apps or keywords as filtering conditions. For example, when a list of apps is designated as the filtering condition, FocusAid will conceal notifications from those apps, making them unnoticeable to the user. Alternatively, if keywords are provided, FocusAid will hide notifications where the sender or message contains the specified keywords. The Android platform only supports the notification-hiding option, and blocking any notifications (or automatic responding) is technically infeasible.

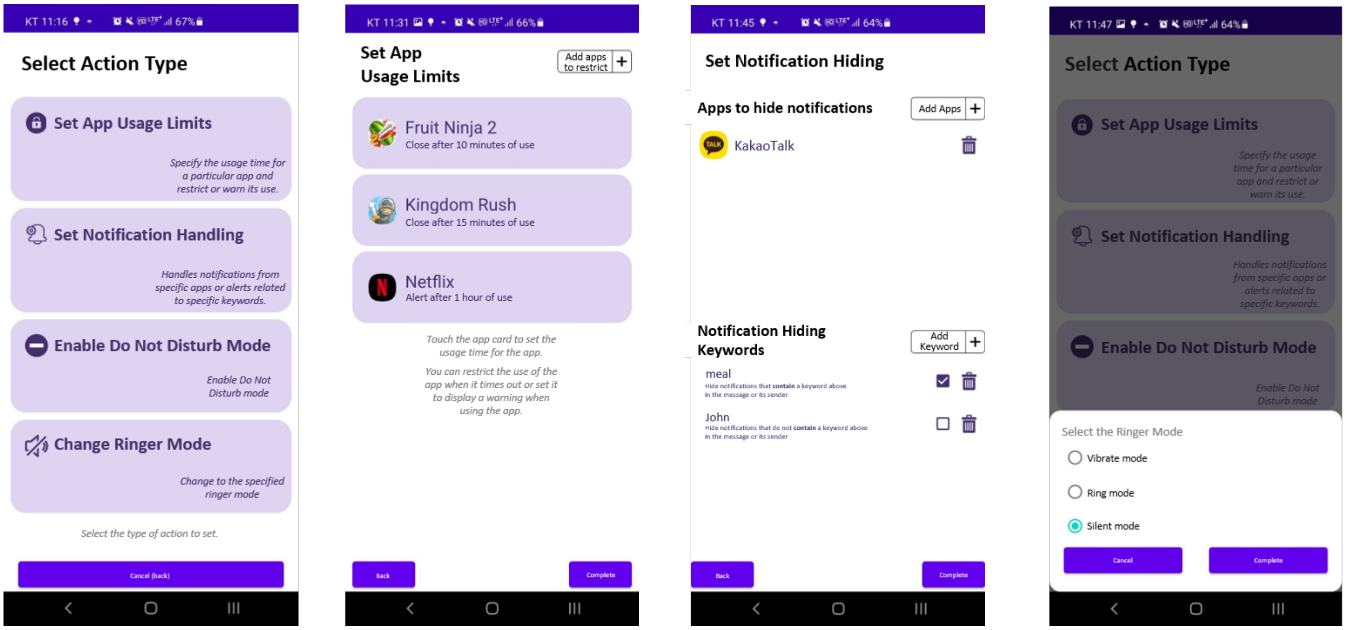
Activating DND mode and Changing Ringer Mode: FocusAid leverages two options for altering a phone's state, namely "Do Not Disturb (DND) mode" and "changing ringer mode," as key actions to manage distractions.

4.3 Rule Reviewing & Management Support

Rule Reviewing: Recognizing that users often overlook the necessity of specifying action release conditions due to an assumption that actions will automatically end (i.e., missing reversal), FocusAid addresses this issue by providing users with an overview of rule operations. This includes details on the conditions under which actions will be triggered, the specific actions to be activated, and the circumstances under which actions will be released (Figure 4a).

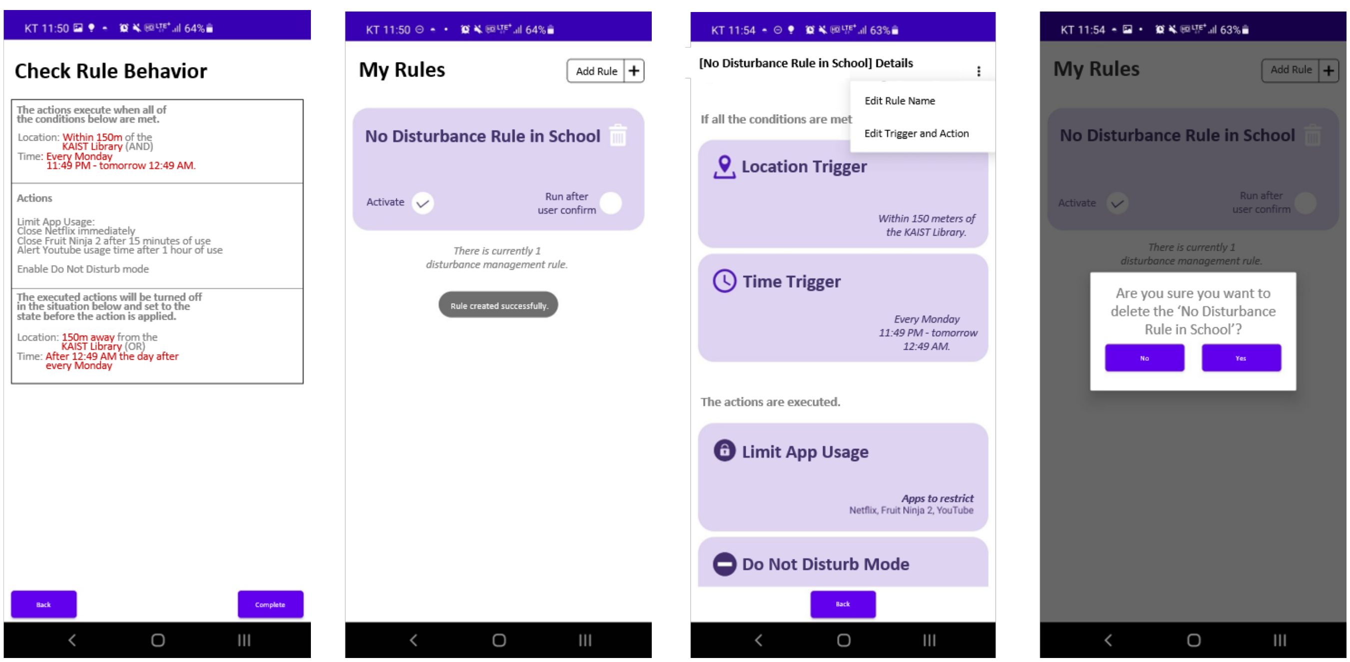
Activating and Deactivating Rules: FocusAid offers users manual rule control, allowing them to manually activate or deactivate rules (Figure 4b). Users can deactivate a rule by unchecking the "Activate" checkbox (i.e., the left checkbox). In this state, the rule will not trigger actions even when its conditions are met.

Dialog-based Automated Rule Execution: FocusAid provides a semi-automated reconfiguration option that grants users greater control over interventions. This feature enables action triggering only with user approval after rule conditions are met (Figure 4b).



(a) Selecting actions (b) Setting apps to limit (c) Hiding notifications (d) Ringer mode change

Figure 3: (a) Selecting actions: Choose limiting app usage, notification handling, enabling Do Not Disturb (DND) mode, and changing ringer mode as actions, (b) Setting apps to limit: Set which apps to limit and the duration of the limitation, (c) Hiding notifications: Configure notification filtering for specific apps and keywords and (d) Changing ringer mode: Define how the ringer mode will be changed.



(a) Rule reviewing (b) Main screen (c) Detailed info. of a rule (d) Deleting a rule

Figure 4: (a) Rule reviewing: Explain when rules will be executed and released, and what actions will be triggered, (b) Main screen: Provide options for managing rules, (c) Detailed information of a rule: Offer comprehensive information about rule operations and (d) Deleting a rule: Remove a rule from the system.

Users can activate this option by checking the "Run after user-confirm" checkbox (i.e., the right checkbox). When enabled, FocusAid will send a notification to the status bar upon satisfying rule conditions. Users can then trigger the rule's actions by selecting the notification.

5 LAB STUDY: EXPLORING INSTANT EVALUATION CYCLE

We conducted a lab-based user study to investigate the instant evaluation cycle, i.e., How do users perceive and address gaps while specifying their implementation intentions as rules.

5.1 Lab Study Method

Overview: We recruited 12 participants (5 females, mean age = 26.3, SD = 4.0) from a university community bulletin board. The study consisted of one-on-one sessions with each participant. Initially, we provided an instruction on how to use FocusAid. Participants then engaged in an exercise session where they specify three simple rules for digital well-being interventions using FocusAid. Following this, we conducted the main session, presenting 21 rules derived from prior work [33] representing various scenarios for distraction management. Participants were asked to program these rules. Finally, we conducted exit interviews to investigate user experiences during rule programming. Participants received approximately 17.5 USD as compensation.

Rule Selection for Evaluation: For the main session, we employed 216 rules from previous research, expressed in natural language scenarios [33]. These rules were categorized into seven representative scenarios: (1) sleep, (2) primary work (e.g., study, research, work), (3) class, (4) in public space, (5) in a library, (6) driving or cycling, and (7) all day. We randomly selected three rules for each scenario, resulting in a total of 21 rules for the main session. Refer to Table 1 for a summary of these rules.

Main Session: The 21 rules were presented to participants one by one in a randomized order to minimize order-related bias. Given the absence of a quantitative metric for assessing users' perceived gap, we postulated that participants would experience a substantial gap when the workload for rule programming was high. Therefore, we employed the NASA Task Load Index, a well-established, subjective, multidimensional workload assessment tool [23], to measure perceived workload. Participants rated their perceived workload across four dimensions: (1) mental demand, (2) effort, (3) performance, and (4) frustration. Additionally, if participants encountered difficulty in directly programming a rule and tried alternative approaches, we asked them to explain why they programmed the rule in that way (i.e., think aloud) [45]. This helped us understand how users perceived the gap and how they addressed it. Through this lab study, we observed a total of programmed 252 rules (21 rules from each of the 12 participants).

Post-Interviews: After completing all programming tasks, we conducted semi-structured interviews with each participant. We began by summarizing their workload scores. We then focused on rules with relatively higher workload scores, hypothesizing that these rules were perceived as more challenging due to a greater perceived gap. Participants were asked to explain why these rules were challenging to represent, with questions such as, "Why did you find it difficult to program this rule?" and "What aspects of programming the rule did you find challenging?"

Data Analysis Methods: Interviews were audio-recorded, transcribed, and segmented into sentences for thematic analysis [21]. The analysis involved an open coding process to extract meaningful expressions, with labeled codes categorized into themes. An iterative approach was used for further analysis, employing affinity diagramming. This iterative process continued until a consensus was reached on the finalized themes [21].

5.2 Lab Study Results

Overall, participants achieved high accuracy and reported relatively low task loads when programming the suggested rules. Participants correctly programmed 17.5 out of 21 rules on average (sd = 1.83) and rated the task load for rule programming as 32.8 (sd = 24.4). However, they made errors in a large fraction of the rules and they perceived a high level of task load (see column #Correct and Task load in Table 1). These errors mainly come from three cases: Firstly, some participants misinterpreted the description of rule to specify, leading to incorrect rule programming. Secondly, in instances involving multiple conditions, there was a misunderstanding of the decision logic; while the rule required all conditions to be met for action triggering (logical AND operation), some participants erroneously believed that satisfying any single condition would suffice (logical OR operation). Lastly, errors were made in setting parameter values for conditions or actions.

Our investigation revealed uncertainties stemming from users' irregular daily routines and users' incorrect mental models about how the system detects context conditions. We found two uncertainties: (1) context-mapping uncertainty and (2) context-recognition uncertainty, affecting users' perception of the gap and their coping strategies.

5.2.1 *Context-Mapping Uncertainty.*

In user-centric context-aware systems, many context-action rules are routinely triggered by specific user contexts. For instance, a rule like "Silence all notifications during class time" (Rule 17) activates whenever a user enters a class. Users should recall relevant routines that necessitate intervention and map these routines to a set of predefined context conditions. However, the irregularity of daily routines introduces *uncertainty* during this mapping process. In essence, users configure context conditions to represent situations requiring intervention, but these situations may also occur in other unanticipated contexts.

This uncertainty arises because the system cannot fully support users in representing their dynamic contexts as context-action rules. Importantly, participants' perceptions of the gap varied according to the extent of their context-mapping uncertainty. For instance, participants with irregular routines, such as variable sleep schedules, struggled to map context conditions to their routines due to uncertainty about whether interventions would always be required within contexts defined by the conditions. As one participant expressed, "*I do not go to bed at a fixed time, so it was hard to predict when I would fall asleep, so I didn't know how to set it up*" (P9).

Participants adopted alternative strategies to cope with the perceived gap resulting from context-mapping uncertainty. Some explored *alternative context conditions*; for example, participants who needed to set a specific time for studying (Rule 17in Table 1) opted

Table 1: The list of 21 rules used for the lab study. Average task load and the number of correctly answered participants (percentage).

No.	Rule description	Task load (avg.)	#Correct (%)
1	YouTube is blocked while riding a bike	11.15	12 (100.0%)
2	Silence all notifications while driving	13.44	9 (75.0%)
3	KakaoTalk notifications are not displayed when in the library	16.25	11 (91.7%)
4	Do not receive notifications when riding a bike	18.85	11 (91.7%)
5	When in the library, silence all notifications and warn if using YouTube for more than 30 minutes	20.83	12 (100.0%)
6	At home, the phone should not make any sound after dawn	24.38	5 (41.7%)
8	Display a warning message when using Facebook for more than 30 minutes	26.15	11 (91.7%)
7	Vibrate all notifications during class	26.15	12 (100.0%)
9	When in the movie theater, turn off the sound in all apps	28.75	4 (33.3%)
10	Silence all notifications during class time	29.69	8 (66.7%)
11	Limit application usage to 2 hours per day	30.31	10 (83.3%)
12	Hide the professor's notifications	33.02	9 (75.0%)
13	Block all app notifications when sleeping	36.04	10 (83.3%)
14	KakaoTalk and Instagram cannot be used while in the classroom	40.83	12 (100.0%)
15	When lying down in bed to sleep, Facebook is available up to 10 minutes, after which it is blocked	40.94	12 (100.0%)
16	All notifications are turned off when sitting in the lab and reading a paper	44.27	11 (91.7%)
17	Block apps for fun like games or YouTube when beginning studying	46.77	12 (100.0%)
18	Block non-work related app notifications during working hours	47.29	11 (91.7%)
19	Silence all notifications in quiet public places where silence is required	47.92	9 (75.0%)
20	When in the library, all apps are disabled, and all KakaoTalk notifications received from all people are hidden except for specific people	49.90	10 (83.3%)
21	Silence notifications and phone calls when riding a bus or subway	54.90	9 (75.0%)

to use location conditions instead of time conditions. One participant who specified a location condition for studying explained, "*I sometimes study at dawn, during the day, and at night. It seemed so diverse, so I thought the time was meaningless*" (P1). Others attempted to identify *the bare minimum* needed to represent their desired situations alternately. These participants narrowed down conditions until they guaranteed the desired situation, even if they recognized that the system might not detect the complete situation. For example, a participant said, "*At least, I fall asleep between 3 am and 8:30 am, so I set the rule to hide all notifications if there is no smartphone movement during that time*" (P10). These observations indicate how participants adapt their rule representations when struggling to predict irregular routines and map them using the provided context condition set. In contrast, participants with more regular routines found it easier to map their routines to the given context conditions without difficulty. As one participant stated, "*I usually sleep only at home, so I set the location as 'at home,' and my typical sleep time is quite consistent, so I set it accordingly*" (P2).

5.2.2 Context-Recognition Uncertainty. Context-recognition uncertainty is closely tied to users' mental models of how the system identifies context conditions based on contextual information. Our research revealed that users who experienced more substantial context recognition uncertainty also perceived a more pronounced gap. Furthermore, the level of perceived context recognition uncertainty varied depending on users' familiarity with context-aware technology. For example, Rule 21 required participants to set conditions related to using public transportation, such as buses or subways ("Silence notifications and phone calls when riding a bus or subway"). Some participants expected that specifying driving activity as a condition would trigger similar contexts. One participant explained, "*When driving, the speed is fast, and there will be acceleration, so I thought the system would sense taking a bus or subway similarly to driving a car*" (P7). However, participants who were uncertain about how the system detected context conditions faced a more significant technical gap. One participant commented, "*I wondered if this could recognize the case of taking a bus or subway, but I used the driving activity anyway. But, in fact, I am not sure.*" (P6).

Additionally, context recognition uncertainty could be increased by a lack of clear descriptions for context conditions. For instance, we labeled the "still" activity condition as "no movement" in the user interface, as it triggers when the user is in a stationary state without movement. While most participants employed the "still" activity to specify stationary activities like sleep or study, some were uncertain about how the system interpreted the "still" activity. P9 questioned, "*Does 'no movement' mean that when 'I hold my phone without moving' or 'I leave it alone?'*"

In response to perceived gaps stemming from context recognition uncertainty, participants tried to represent their desired situations alternatively. However, some participants ended up specifying invalid conditions. For example, Rule 21 required participants to set driving activity as the condition, but some selected a location condition (e.g., a subway station), which was invalid because the action would be released when the user was outside the specified place. A few participants even abandoned rule programming, complaining about the difficulty of the task.

6 FIELD STUDY: EXPLORING DELAYED EVALUATION CYCLE

We conducted a field study to investigate the delayed evaluation cycle, i.e., How do users perceive and address gaps when the execution of specified rules.

6.1 Field Study Method

We recruited 46 participants (21 females, mean age = 25.5, SD = 5.8) from two large universities in Korea through online bulletin boards. During an online orientation, participants were instructed to install FocusAid. To establish personalized goals, we utilized a commercial app (i.e., ActionDash) to track each participant's phone usage over the week before the field study, setting this as their baseline. These personalized goals targeted a 15% reduction in app usage duration and screen unlock frequency, tailored to each individual's baseline data as suggested by a prior study [37]. For example, if a participant's baseline data showed an average daily screen unlock frequency of 100 times, their goal would be to reduce it to 85 times per day. Likewise, if another participant had an average daily app usage duration of 3 hours, their target would be to bring it down to 2 hours and 33 minutes.

To encourage participant engagement, we offered rewards at the end of the field study, with a maximum compensation of approximately 12 USD for achieving screen time and on/off frequency goals (15% reduction), which is a similar level of compensation considered in a recent study on digital wellbeing service evaluation [56].

After three weeks of the field study, all participants completed a post-survey. We evaluated the usability of FocusAid using the USE questionnaire, which assessed usefulness, satisfaction, and ease of use dimensions [49]. Additionally, we asked participants to evaluate FocusAid's effectiveness in managing distractions (e.g., 'FocusAid helped me avoid excessive app use') using a 5-point Likert scale. Subsequent to the survey, semi-structured interviews were conducted with fifteen participants, each lasting about 20 minutes. These interviews included questions like 'Why did you create certain rules?' and 'In what situations did you find it necessary to modify these rules?' to gain deeper insights into their rule management strategies and experiences. Participants received a compensation of 50 USD for their participation in the study, with an additional 8 USD provided to interview participants. Taking into account additional rewards (up to 12 USD), the maximum monetary compensation a participant could receive amounted to about 70 USD.

6.2 Field Study Results

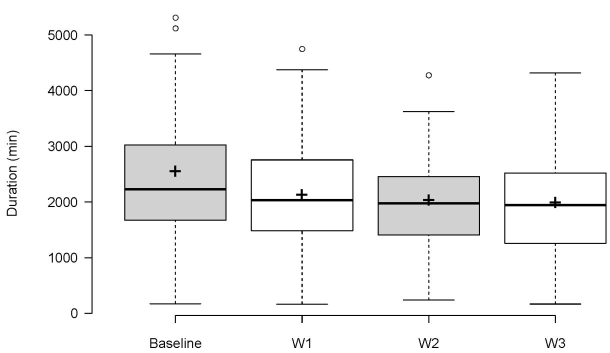
In this section, we present the results of the field study. We begin by reporting the overall statistics of the field study. Subsequently, we detail how participants adaptively managed context-action rules (i.e., modification or deletion) as they interacted with the intervention systems over time. Finally, we report on how users coped with technical problems, such as late intervention triggers.

6.2.1 Overall Statistics of the Field Study. *Statistics of Created Rules*: Participants generated a total of 134 rules (average = 2.91, SD = 1.31). These rules pertained to their primary work (e.g., study, work) (66 rules), sleep (26 rules), rest (19 rules), classes (11 rules), and driving (5 rules). Of the 134 rules, 98 specified time conditions, 59 described location conditions, and 27 described activity conditions. Among the rules that specified activity as the condition, six were related to 'driving activity,' while 21 were associated with 'still activity.' We found that 'Still activity' was often combined with other conditions (i.e., time, location) and primarily used to define situations where participants needed to concentrate, such as during classes, studying, and work. Furthermore, 100 out of the 134 rules specified actions for limiting app usage. Participants predominantly opted to restrict app usage by designating specific apps rather than applying limitations to all apps (86 rules vs. 14 rules, respectively). On average, participants set a time allowance of 46.74 minutes per app (SD = 42.23). Additionally, 62 out of the 134 rules specified actions for hiding notifications. Among these, nine rules hid notifications from all apps, while 25 targeted specific apps for hiding notification. Five rules hid notifications based on keywords. Finally, 28 rules detailed configurations of smartphone settings (e.g., changing the ringer mode and activating the Do Not Disturb (DND) mode).

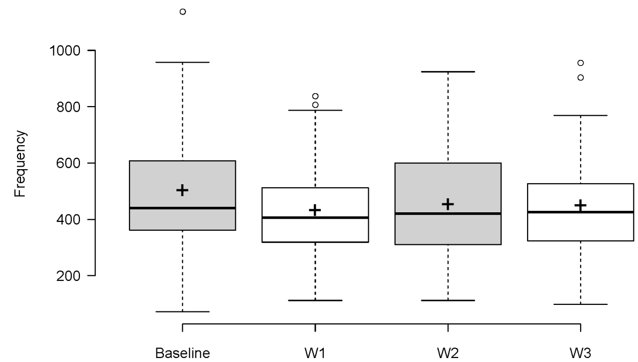
Usability and Efficacy of FocusAid: The overall usability rating was given as 4.00 (SD = 0.74), with sub-ratings as follows: usefulness (average = 3.88, SD = 0.79), ease of use (average = 3.91, SD = 0.85), ease of learning (average = 4.42, SD = 0.54), and satisfaction (average = 3.97, SD = 0.67). Participants rated FocusAid's efficiency as 4.13 (SD = 0.69) in response to the question "FocusAid helped me avoid excessive app use." We analyzed whether participants managed to reduce their app usage time and the frequency of unlocking their screens while using FocusAid. Figure 5 presents participants' average total app usage time and screen unlock frequency during each week, with the baseline period for comparison being the week before they began using FocusAid. Paired t-test is conducted to assess the differences in app usage time between the baseline and the first week ($p = 0.06$), revealing no statistically significant difference. However, statistically significant differences were observed between the baseline and the second and third weeks ($p = 0.019$ and $p = 0.013$, respectively). Concerning the frequency of screen unlocks, no significant differences were found between the baseline and subsequent weeks ($p > 0.05$). These findings suggest that while the number of screen-unlocks triggered by notifications may have decreased to some extent through notification hiding, it could not be significantly reduced, as FocusAid could not prevent users from habitually unlocking the screen without external triggers (e.g., notifications). Encouragingly, 78.3% of participants responded they would continue to use FocusAid.

6.2.2 Users' Rule Management. During the field study, participants experienced and evaluated the appropriateness of the intervention when a context-action rule was triggered based on conditions they had set (i.e., delayed evaluation). They then adaptively modified the rules by adjusting the context or actions.

Rule Modification: Initially, participants planned their daily routines by configuring rules, but occasionally, they faced challenges in adhering to these plans. For instance, P3 mentioned, "*I initially planned to wake up around eight o'clock. So, I configured YouTube to be unavailable until 9 am. However, I woke up later than expected and adjusted the time condition to 10 am.*" Participants also modified rules as their overall circumstances changed. For example, in the second week of the field study, coinciding with a midterm exam (note that



(a) Changes in app usage



(b) Changes in screen unlocks counts

Figure 5: Changes in participants' total usage of smartphone apps and the screen unlocks count

our participants were undergraduate or graduate students), some participants created restrictive rules to minimize unproductive app use. After the midterm exam, these participants relaxed these rules. P8 explained, "*During the midterm test period, I restricted many apps, such as social media, games, and webtoons, which I typically use. However, after the test, I allowed access to social media apps.*" We also observed participants modifying rules to find appropriate restraining actions. These modifications involved increasing or decreasing the level of restriction. Some participants found that actions were more or less coercive than they initially thought and adjusted them accordingly. One participant stated, "*Initially, I allowed only 15 minutes for a quick check of KakaoTalk, but I increased it to 30 minutes because I found that 15 minutes was insufficient. As for Chrome, I initially set a generous usage time, but I reduced it slightly after realizing it was more than enough.*" (P13). In terms of managing external interruptions, some participants found that they still received notifications despite the filtering conditions they had initially set. These participants adjusted the filtering conditions to hide more notifications. One participant mentioned, "*Initially, I configured it to prevent notifications only from certain apps, but I still received notifications from other apps. Therefore, I modified the rule to include additional apps for hiding notifications.*" (P2).

Rule Deletion: Rule deletions were rare throughout the study (only two participants deleted rules). One participant deleted a rule because it became inconvenient. He explained, "*I had a rule that prohibited the use of KakaoTalk while driving, but during one trip, I stopped the car briefly to send a message, but the rule prevented me. After encountering this inconvenience several times, I realized there might be situations where I needed to use KakaoTalk while driving, so I deleted that rule.*" (P4). Another participant did not delete the rules but expressed, "*In my opinion, if a rule interferes with my daily life, I would consider deleting it.*" (P15). Participants stated that they would only delete a rule if their circumstances changed. One participant commented, "*If I get a job and my needs change, I might delete some rules.*" (P7).

User Responses to Occasional Technical Issues: There were occasional instances where FocusAid did not perform as expected. However, participants refrained from modifying the rules in these cases. For instance, participants reported that FocusAid sometimes recognized a context with a slight delay, which they attributed to technological limitations. One participant noted, "*Shortly after the car started moving, it appeared that FocusAid didn't recognize that I was driving. I thought this might be due to technical limitations.*" (P4). Interestingly, some participants opted to comply with the intervention even when the rules occasionally failed to work as intended. One participant remarked, "*Instead of thinking, 'Let's continue using the app,' I turned it off because I needed to focus on studying.*" (P8). Participants explained that FocusAid heightened their awareness of desired behaviors, such as limiting phone usage, by consistently providing interventions in designated contexts. This, in turn, helped them adhere to the intervention, even when technical issues occasionally prevented interventions from triggering as expected.

7 DISCUSSION

7.1 Considering Technical Feasibility for User-Centric Context-Aware Interventions

In order to assess the technical feasibility of implementing a user-centric context-aware intervention system for digital well-being on the mobile platform, we conducted a comprehensive technical review study based on an existing dataset [33]. Our investigation revealed several challenges in meeting the diverse requirements of users when we implement them in mobile platforms, such as Android. To bridge these technical gaps, we employed an approach that provided approximated features catering to users' needs (i.e., first-order approximation) [3]. For instance, we abstracted heterogeneous context conditions required by users [33] into three generic conditions: location, time, and activity. This simplification allowed users to specify their desired trigger conditions using these abstracted conditions in a generic and alternative manner. Such abstract types of conditions can be useful for specifying rules when users' intentions are not articulated. In the future, various methods can be utilized to elicit intentions, like engaging in natural language dialogue with a conversational agent [10]. Furthermore, we implemented system features while taking into account various technical constraints (i.e., constrained possibilities) [15]. For example, we provided the "still" condition to address stationary activities like sleep and study, which are challenging to detect using mobile phone sensors alone [36].

Technically supporting users' mental models should also be carefully considered when implementing a rule-based context-aware intervention system. Huang et al. presented that users' misunderstanding of types of conditions (states vs. events) can cause mental model errors and suggested making it clear when the system triggers and ends a certain action [26]. Therefore, we only allowed state conditions since state conditions clearly describe when to trigger and release intervention (i.e., disallowing confusing options [26]). In addition, we offered the rule-reviewing function to support the system's intelligibility [5, 46]. We provided an explicit description to improve users' understanding of how rules operate, including when to trigger and release interventions. We expected this feature to be helpful for users since many users tend to consider only trigger conditions for intervention [6, 26]. However, some participants commented that a plain text-based description (i.e., summarization of rule operation) was not intuitive to the users. The rule-reviewing function can be further improved by providing a user-friendly or intuitive description so that users can be more engaged (e.g., visualization of system operation [59]). In general, trigger-action programming for context-aware systems is based on the interface-as-conversation paradigm, lacking *articulatory similarity* between a user's real-world intention specified in the rules and the actual physical form of the rules in the mobile app [28]. In practice, debugging context-aware rules at the interface level is challenging due to a lack of real-world operations, as shown in a delayed evaluation cycle. A recent work aimed to bridge this gap by leveraging a 3D simulator in a smart home environment [71], which can potentially increase *articulatory directness* with the interface-as-model-world paradigm [28] that enables direct interaction in a simulated world. Another factor that affects a user's mental model is that the details of real-world operations are not exposed to the users (e.g., GPS errors or activity recognition accuracies). This issue originates from the fact that the interface language of FocusAid has limited expressiveness, which can be improved by exposing such operational details to the users. This approach can be explained by using the concept of "seamful design," which transparently reveals system mechanisms and technical limitations to users instead of concealing them to address technical gaps and enhance users' mental models [8]

7.2 Addressing Context-Mapping and Context-Recognition Uncertainties

According to Activity Theory [30], human behavior is shaped through interactions with tools and the environment. This theory emphasizes that such interactions are significantly influenced by various factors, including social, individual, and cultural contexts. These diverse elements collectively contribute to shaping individual behavior and experience in different scenarios. Our study found that the regularity of users' daily routines significantly affects *context-mapping uncertainty*, impacting both behavior (e.g., rule specification) and experience (e.g., perceiving gaps). Therefore, when designing user-centric context-aware interventions for digital wellbeing on a mobile platform, it's essential to consider users' routine regularity, given the constant use and seamless interaction with mobile devices, such as smartphones.

To address irregular routines in context-rule-based digital wellbeing interventions, incorporating additional contextual information is beneficial. For example, while smartphones may not always accurately infer user activities, especially when not carried by users, wearable devices can continuously track user activities. By using a combination of sensors, activities like sleep or exercise can be inferred in real-time and set as conditions in context-action rules. This could significantly reduce the effort required by users to specify rules for their irregular routines. Furthermore, utilizing calendar information [48] for events like exams or social appointments allows users to adapt rules proactively before the events occur, easing the burden of users in specifying their dynamic daily routine as the rules.

The technology acceptance model (TAM) suggests that a low perceived ease of use, due to limited understanding, requires more user effort and leads to negative attitudes towards the system [11]. Our study showed that a lack of knowledge about the workings of context-aware systems can lead to incorrect user mental models, causing *context-recognition uncertainty*. We also observed a link between users' perception of gaps and their levels of context-recognition uncertainty.

Addressing the context-recognition uncertainty through better technical support and clear user guidance is crucial. Service providers should prioritize achieving a high level of system intelligibility that aligns with users' mental models. A user-centric context-aware system should be presented in an intelligible manner, enabling users to easily grasp how the context rules function. This is in line with existing research emphasizing the significance of enhancing system intelligibility to bridge the gap between users' mental models and the system's conceptual model [13, 15, 46]. Notably, contemporary applications operating in complex contextual environments often fall short of explaining the specifics of user context utilization. For example, mobile apps offering recommendation services based on user contexts typically request consent for context tracking (e.g., location, activity, phone usage) but do not explain its operational mechanics.

7.3 Understanding User Interaction with User-Centric Context-Aware Systems

This study thoroughly explores the interaction process between user-centric context-aware systems and users through the lab study (instant evaluation cycles) and the field study (delayed evaluation cycles). These two studies provide insights into different aspects of the user interaction process as presented in Figure 1.

Firstly, the lab study revealed that users recognize gaps in the process of translating their implementation intentions into rules due to context-mapping and context-recognition uncertainties. Notably, this study identified coping strategies that users adopt when they perceive these gaps (i.e., instant evaluation cycle). This provides crucial insights into how users specify their implementation intentions as rules despite technological limitations. The field study observed how users perceive and address gaps that arise when rules are actually executed (i.e., delayed evaluation cycle), identifying the reasons for these gaps and how users modify rules.

Secondly, the design and method choices of each study played a crucial role in understanding the interaction between users and the system. The lab study provided a setting for observing users' immediate reactions and adaptations for rule specification in a controlled environment. In contrast, the field study offered users the opportunity to experience and address gaps through delayed evaluation processes in a natural environment. This allowed for a deeper understanding of long-term user adaptation and system effectiveness in the real world.

Lastly, a notable point from the field study is that despite technical issues leading to rules operating contrary to users' implementation intentions (e.g., failed to recognize contexts), there was a tendency for users to comply with and continue using FocusAid. A significant proportion of respondents (78.3%) expressed their desire to continue using FocusAid, and many positively evaluated the system's efficiency. This response can be explained through the 'perceived usefulness' concept of the technology acceptance model [68]. Users likely felt that FocusAid generally helped limit smartphone usage and manage distractions. This perception of usefulness led users to believe that FocusAid was worth using despite technical issues. However, there is a need to observe its use over a longer period of time to better understand a user's long-term adaptation behaviors and their mental model.

Note that the current system design for FocusAid is largely based on restrictive methods that have been widely used in existing work [50]. A recent study showed their limitations for long-term effectiveness, suggesting alternative approaches for educating self-regulatory behaviors in the long [61]. User-centric context-aware systems can be extended to deliver diverse intervention methods. Further long-term studies on user-centric context-aware systems for digital wellbeing should be conducted to observe whether users continue to find them effective when provided with interfaces for customized context-action rules.

7.4 Limitations

This work comes with several limitations, primarily concerning its generalizability. To begin with, our user studies primarily targeted college students, a pivotal demographic that often requires technical aids for self-regulating smartphone usage due to the observed negative correlations between problematic phone usage, academic performance, and mental health conditions [62, 64, 69]. Consequently, further investigations are warranted to explore the diverse user needs within other segments of the population concerning digital well-being, such as youth and knowledge workers. Secondly, our examination of technical feasibility was restricted to the Android platform. Conducting a parallel study on the iOS platform could offer valuable insights into whether analogous user adaptations emerge in alternative mobile ecosystems. Lastly, this work did not investigate how the personalized goals provided to participants in our field study might have influenced the outcomes. A deeper understanding of the impact of these personalized goals on user behavior and well-being would be beneficial.

8 CONCLUSION

The primary aim of this study was to explore the nature of interactions between users and a user-centric context-aware intervention system, specifically focusing on understanding how users perceive and address the gap between their goal/implementation intentions and the system's behavior. To this end, we developed FocusAid, a straightforward mobile intervention system that enables users to set context-action rules for digital well-being. In our lab study, we examined the instant evaluation cycle by observing how users adaptively create rules based on specific implementation intentions. This study uncovered two main uncertainties: context-mapping and context-recognition. We elucidated how these uncertainties relate to the perceived gaps and the strategies participants used to bridge these gaps through alternative rule formulations. Furthermore, our field study delved into the delayed evaluation cycle, revealing how users adaptively adjust context-action rules when the execution of these rules diverges from their goal/implementation intentions. Our study significantly advances the understanding of user interactions within user-centric, context-aware systems, extending existing interaction models. It also identifies key challenges in the realm of digital well-being, providing critical insights for future research and development in user-centric context-aware systems.

ACKNOWLEDGMENTS

This research was supported by the Bio & Medical Technology Development Program and the Basic Science Research Program of the National Research Foundation (NRF), funded by the Korean government (MSIT) (No. 2021M3A9E4080780, 2022R1A2C2011536).

REFERENCES

[1] Gregory D Abowd and Russell Beale. 1991. Users, systems and interfaces: A unifying framework for interaction. In *HCI*, Vol. 91. 73–87.
[2] Chadia Abras, Diane Maloney-Krichmar, Jenny Preece, et al. 2004. User-centered design. *Bainbridge, W. Encyclopedia of Human-Computer Interaction. Thousand Oaks: Sage Publications* 37, 4 (2004), 445–456.
[3] Mark S Ackerman. 2000. The intellectual challenge of CSCW: the gap between social requirements and technical feasibility. *Human–Computer Interaction* 15, 2-3 (2000), 179–203.
[4] Raffaele Ariano, Marco Manca, Fabio Paternò, and Carmen Santoro. 2023. Smartphone-based augmented reality for end-user creation of home automations. *Behaviour & Information Technology* 42, 1 (2023), 124–140.
[5] Victoria Bellotti and Keith Edwards. 2001. Intelligibility and accountability: human considerations in context-aware systems. *Human–Computer Interaction* 16, 2-4 (2001), 193–212.
[6] Will Brackenbury, Abhimanyu Deora, Jillian Ritchey, Jason Vallee, Weijia He, Guan Wang, Michael L Littman, and Blase Ur. 2019. How users interpret bugs in trigger-action programming. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*. 1–12.
[7] Danilo Caivano, Daniela Fogli, Rosa Lanzilotti, Antonio Piccinno, and Fabio Cassano. 2018. Supporting end users to control their smart home: design implications from a literature review and an empirical investigation. *Journal of Systems and Software* 144 (2018), 295–313.
[8] Matthew Chalmers, Ian MacColl, and Marek Bell. 2003. Seamful design: Showing the seams in wearable computing. In *2003 IEE Eurowearable*. IET, 11–16.
[9] Fabio Clarizia, Saverio Lemma, Marco Lombardi, and Francesco Pascale. 2017. A mobile context-aware information system to support tourism events. In *Green, Pervasive, and Cloud Computing: 12th International Conference, GPC 2017, Cetara, Italy, May 11-14, 2017, Proceedings 12*. Springer, 553–566.
[10] Fulvio Corno, Luigi De Russis, and Alberto Monge Roffarello. 2021. From users' intentions to if-then rules in the internet of things. *ACM Transactions on Information Systems (TOIS)* 39, 4 (2021), 1–33.
[11] Fred D Davis. 1989. Perceived usefulness, perceived ease of use, and user acceptance of information technology. *MIS quarterly* (1989), 319–340.
[12] Anind K Dey, Raffay Hamid, Chris Beckmann, Ian Li, and Daniel Hsu. 2004. a CAPpella: programming by demonstration of context-aware applications. In *Proceedings of the SIGCHI conference on Human factors in computing systems*. 33–40.
[13] Anind K Dey and Alan Newberger. 2009. Support for context-aware intelligibility and control. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. 859–868.
[14] Anind K Dey, Timothy Sohn, Sara Streng, and Justin Kodama. 2006. iCAP: Interactive prototyping of context-aware applications. In *International Conference*

on Pervasive Computing. Springer, 254–271.
[15] W Keith Edwards, Mark W Newman, and Erika Shehan Poole. 2010. The infrastructure problem in HCI. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. 423–432.
[16] Massimo Esposito, Aniello Minutolo, Rosario Megna, Manolo Forastiere, Mario Magliulo, and Giuseppe De Pietro. 2018. A smart mobile, self-configuring, context-aware architecture for personal health monitoring. *Engineering Applications of Artificial Intelligence* 67 (2018), 136–156.
[17] Gerhard Fischer. 2012. Context-aware systems: the'right'information, at the'right'time, in the'right'place, in the'right'way, to the'right'person. In *Proceedings of the international working conference on advanced visual interfaces*. 287–294.
[18] Simone Gallo and Fabio Paterno. 2022. A Conversational Agent for Creating Flexible Daily Automation. In *Proceedings of the 2022 International Conference on Advanced Visual Interfaces*. 1–8.
[19] Giuseppe Ghiani, Marco Manca, Fabio Paternò, and Carmen Santoro. 2017. Personalization of context-dependent applications through trigger-action rules. *ACM Transactions on Computer-Human Interaction (TOCHI)* 24, 2 (2017), 1–33.
[20] Stephanie P Goldstein, Brittney C Evans, Daniel Flack, Adrienne Juarascio, Stephanie Manasse, Fengqing Zhang, and Evan M Forman. 2017. Return of the JITAI: applying a just-in-time adaptive intervention framework to the development of m-health solutions for addictive behaviors. *International journal of behavioral medicine* 24, 5 (2017), 673–682.
[21] Greg Guest, Kathleen M MacQueen, and Emily E Namey. 2011. *Applied thematic analysis*. sage publications.
[22] David H Gustafson, Fiona M McTavish, Ming-Yuan Chih, Amy K Atwood, Roberta A Johnson, Michael G Boyle, Michael S Levy, Hilary Driscoll, Steven M Chisholm, Lisa Dillenburg, et al. 2014. A smartphone application to support recovery from alcoholism: a randomized clinical trial. *JAMA psychiatry* 71, 5 (2014), 566–572.
[23] Sandra G Hart and Lowell E Staveland. 1988. Development of NASA-TLX (Task Load Index): Results of empirical and theoretical research. In *Advances in psychology*. Vol. 52. Elsevier, 139–183.
[24] Alexis Hiniker, Sungsoo Hong, Tadayoshi Kohno, and Julie A Kientz. 2016. MyTime: designing and evaluating an intervention for smartphone non-use. In *Proceedings of the 2016 CHI conference on human factors in computing systems*. 4746–4757.
[25] Yubo Hou, Dan Xiong, Tonglin Jiang, Lily Song, and Qi Wang. 2019. Social media addiction: Its impact, mediation, and intervention. *Cyberpsychology: Journal of psychosocial research on cyberspace* 13, 1 (2019).
[26] Justin Huang and Maya Cakmak. 2015. Supporting mental model accuracy in trigger-action programming. In *Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing*. 215–225.
[27] Kuo-Ying Huang. 2009. Challenges in human-computer interaction design for mobile devices. In *Proceedings of the World Congress on Engineering and Computer Science*, Vol. 1. San Francisco, USA, 1–6.
[28] Edwin L Hutchins, James D Hollan, and Donald A Norman. 1985. Direct manipulation interfaces. *Human–computer interaction* 1, 4 (1985), 311–338.
[29] Kazi Sinthia Kabir, Stacey A Kenfield, Erin L Van Blarigan, June M Chan, and Jason Wiese. 2022. Ask the Users: A Case Study of Leveraging User-Centered Design for Designing Just-in-Time Adaptive Interventions (JITAIs). *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 6, 2 (2022), 1–21.
[30] Victor Kaptelinin and Bonnie Nardi. 2012. *Activity theory in HCI: Fundamentals and reflections*. Morgan & Claypool Publishers.
[31] Halimahtun M Khalid. 2006. Embracing diversity in user needs for affective design. *Applied ergonomics* 37, 4 (2006), 409–418.
[32] Donghee Kim, Kyungsik Han, Jeong Seop Sim, and Youngtae Noh. 2018. Smombie Guardian: We watch for potential obstacles while you are walking and conducting smartphone activities. *PLoS one* 13, 6 (2018), e0197050.
[33] Inyeop Kim, Hwarang Goh, Nematjon Narziev, Youngtae Noh, and Uichin Lee. 2020. Understanding User Contexts and Coping Strategies for Context-aware Phone Distraction Management System Design. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 4, 4 (2020), 1–33.
[34] Inyeop Kim, Gyuwon Jung, Hayoung Jung, Minsam Ko, and Uichin Lee. 2017. Let's focus: location-based intervention tool to mitigate phone use in college classrooms. In *Proceedings of the 2017 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proceedings of the 2017 ACM International Symposium on Wearable Computers*. 101–104.
[35] Inyeop Kim, Gyuwon Jung, Hayoung Jung, Minsam Ko, and Uichin Lee. 2017. Let's FOCUS: Mitigating Mobile Phone Use in College Classrooms. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 1, 3 (2017), 1–29.
[36] Inyeop Kim, Uichin Lee, and Narae Cha. 2018. Exploring Context-aware Proactive Blocking for Distraction Management. In *Proceedings of the 2018 ACM International Joint Conference and 2018 International Symposium on Pervasive and Ubiquitous Computing and Wearable Computers*. 1202–1207.
[37] Jaejeung Kim, Hayoung Jung, Minsam Ko, and Uichin Lee. 2019. GoalKeeper: Exploring Interaction Lockout Mechanisms for Regulating Smartphone Use. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 3, 1 (2019), 1–29.
[38] Minsam Ko, Seungwoo Choi, Subin Yang, Joonwon Lee, and Uichin Lee. 2015. FamiLync: facilitating participatory parental mediation of adolescents' smartphone use. In *Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing*. 867–878.
[39] Minsam Ko, Seungwoo Choi, Koji Yatani, and Uichin Lee. 2016. Lock n'LoL: group-based limiting assistance app to mitigate smartphone distractions in group activities. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*. 998–1010.
[40] Minsam Ko, Subin Yang, Joonwon Lee, Christian Heizmann, Jinyoung Jeong, Uichin Lee, Daehee Shin, Koji Yatani, Junehwa Song, and Kyong-Mee Chung. 2015. NUGU: a group-based intervention app for improving self-regulation of limiting smartphone use. In *Proceedings of the 18th ACM conference on computer supported cooperative work & social computing*. 1235–1245.
[41] Grégoire S Larue, Christopher N Watling, Alexander A Black, Joanne M Wood, and Mahrokh Khakzar. 2020. Pedestrians distracted by their smartphone: Are in-ground flashing lights catching their attention? A laboratory study. *Accident Analysis & Prevention* 134 (2020), 105346.
[42] Jisoo Lee, Erin Walker, Winslow Burleson, Matthew Kay, Matthew Buman, and Eric B Hekler. 2017. Self-experimentation for behavior change: Design and formative evaluation of two approaches. In *Proceedings of the 2017 CHI conference on human factors in computing systems*. 6837–6849.
[43] Uichin Lee and Jaejeung Kim. 2019. Interaction Restraint Framework for Digital Wellbeing. In *ACM CHI Workshop: Designing for Digital Wellbeing*. ACM.
[44] 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.
[45] Clayton Lewis. 1982. *Using the" thinking-aloud" method in cognitive interface design*. IBM TJ Watson Research Center Yorktown Heights, NY.
[46] Brian Y Lim, Anind K Dey, and Daniel Avrahami. 2009. Why and why not explanations improve the intelligibility of context-aware intelligent systems. In *Proceedings of the SIGCHI conference on human factors in computing systems*. 2119–2128.
[47] Markus Löchtefeld, Matthias Böhmer, and Lyubomir Ganev. 2013. AppDetox: helping users with mobile app addiction. In *Proceedings of the 12th international conference on mobile and ubiquitous multimedia*. 1–2.
[48] Tom Lovett, Eamonn O'Neill, James Irwin, and David Pollington. 2010. The Calendar as a Sensor: Analysis and Improvement Using Data Fusion with Social Networks and Location. In *Proceedings of the 12th ACM International Conference on Ubiquitous Computing* (Copenhagen, Denmark) *(UbiComp '10)*. Association for Computing Machinery, New York, NY, USA, 3–12.
[49] Arnold M. Lund. 2001. Measuring Usability with the USE Questionnaire. *STC Usability SIG Newsletter* 8, 2 (2001), 3–6.
[50] Ulrik Lyngs, Kai Lukoff, Petr Slovak, Reuben Binns, Adam Slack, Michael Inzlicht, Max Van Kleek, and Nigel Shadbolt. 2019. Self-Control in Cyberspace: Applying Dual Systems Theory to a Review of Digital Self-Control Tools. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*. 1–18.
[51] Ulrik Lyngs, Kai Lukoff, Petr Slovak, William Seymour, Helena Webb, Marina Jirotka, Max Van Kleek, and Nigel Shadbolt. 2020. 'I Just Want to Hack Myself to Not Get Distracted': Evaluating Design Interventions for Self-Control on Facebook. *arXiv preprint arXiv:2001.04180* (2020).
[52] Alberto Monge Roffarello and Luigi De Russis. 2019. The race towards digital wellbeing: Issues and opportunities. In *Proceedings of the 2019 CHI conference on human factors in computing systems*. 1–14.
[53] Brad Myers. 1994. Challenges of HCI design and implementation. *interactions* 1, 1 (1994), 73–83.
[54] Felix Naughton, Sarah Hopewell, Neal Lathia, Rik Schalbroeck, Chloë Brown, Cecilia Mascolo, Andy McEwen, Stephen Sutton, et al. 2016. A context-sensing mobile phone app (Q sense) for smoking cessation: a mixed-methods study. *JMIR mHealth and uHealth* 4, 3 (2016), e5787.
[55] Donald A Norman. 1988. *The psychology of everyday things*. Basic books.
[56] Joonyoung Park, Hyunsoo Lee, Sangkeun Park, Kyong-Mee Chung, and Uichin Lee. 2021. GoldenTime: Exploring System-Driven Timeboxing and Micro-Financial Incentives for Self-Regulated Phone Use. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*. 1–17.
[57] Peter Pirolli, Shiwali Mohan, Anusha Venkatakrishnan, Les Nelson, Michael Silva, and Aaron Springer. 2017. Implementation intention and reminder effects on behavior change in a mobile health system: a predictive cognitive model. *Journal of medical Internet research* 19, 11 (2017), e397.
[58] Tie Qiu, Ning Chen, Keqiu Li, Mohammed Atiquzzaman, and Wenbing Zhao. 2018. How can heterogeneous internet of things build our future: A survey. *IEEE Communications Surveys & Tutorials* 20, 3 (2018), 2011–2027.
[59] Fahimeh Raja, Kirstie Hawkey, Steven Hsu, Kai-Le Clement Wang, and Konstantin Beznosov. 2011. A brick wall, a locked door, and a bandit: a physical security metaphor for firewall warnings. In *Proceedings of the Seventh Symposium on*

Usable Privacy and Security. 1–20.
[60] Alberto Monge Roffarello and Luigi De Russis. 2023. Achieving digital wellbeing through digital self-control tools: A systematic review and meta-analysis. *ACM Transactions on Computer-Human Interaction* 30, 4 (2023), 1–66.
[61] Alberto Monge Roffarello and Luigi De Russis. 2023. Hey StepByStep! Can you teach me how to use my phone better? *International Journal of Human-Computer Studies* (2023), 103195.
[62] Maya Samaha and Nazir S Hawi. 2016. Relationships among smartphone addiction, stress, academic performance, and satisfaction with life. *Computers in human behavior* 57 (2016), 321–325.
[63] Bill Schilit, Norman Adams, and Roy Want. 1994. Context-aware computing applications. In *1994 first workshop on mobile computing systems and applications*. IEEE, 85–90.
[64] Sei Yon Sohn, Philippa Rees, Bethany Wildridge, Nicola J Kalk, and Ben Carter. 2019. Prevalence of problematic smartphone usage and associated mental health outcomes amongst children and young people: a systematic review, meta-analysis and GRADE of the evidence. *BMC psychiatry* 19, 1 (2019), 1–10.
[65] Vincent W-S Tseng, Matthew L Lee, Laurent Denoue, and Daniel Avrahami. 2019. Overcoming distractions during transitions from break to work using a conversational website-blocking system. In *Proceedings of the 2019 CHI conference on human factors in computing systems*. 1–13.
[66] Blase Ur, Elyse McManus, Melwyn Pak Yong Ho, and Michael L Littman. 2014. Practical trigger-action programming in the smart home. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. 803–812.
[67] Blase Ur, Melwyn Pak Yong Ho, Stephen Brawner, Jiyun Lee, Sarah Mennicken, Noah Picard, Diane Schulze, and Michael L Littman. 2016. Trigger-action programming in the wild: An analysis of 200,000 ifttt recipes. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*. 3227–3231.
[68] Viswanath Venkatesh and Hillol Bala. 2008. Technology acceptance model 3 and a research agenda on interventions. *Decision sciences* 39, 2 (2008), 273–315.
[69] Jin-Liang Wang, Hai-Zhen Wang, James Gaskin, and Li-Hui Wang. 2015. The role of stress and motivation in problematic smartphone use among college students. *Computers in Human Behavior* 53 (2015), 181–188.
[70] Jong-bum Woo and Youn-kyung Lim. 2015. User experience in do-it-yourself-style smart homes. In *Proceedings of the 2015 ACM international joint conference on pervasive and ubiquitous computing*. 779–790.
[71] Lefan Zhang, Cyrus Zhou, Michael L. Littman, Blase Ur, and Shan Lu. 2023. Helping Users Debug Trigger-Action Programs. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 6, 4, Article 196 (jan 2023), 32 pages.