A Collaborative Health Adherence Optimization System

by

Maybins Douglas Lengwe
B.Sc., Copperbelt University, 2008
M.Sc., Cleveland State University, 2011

A Dissertation Submitted in Partial Fulfillment of the
Requirements for the Degree of

DOCTOR OF PHILOSOPHY

in the Department of Computer Science

University of Victoria

A Collaborative Health Adherence Optimization System

by

Maybins Douglas Lengwe
B.Sc., Copperbelt University, 2008
M.Sc., Cleveland State University, 2011

Supervisory Committee

Dr. Jens Weber, Supervisor
(Computer Science Department, University of Victoria)

Dr. Charles Perin, Co-Supervisor
(Computer Science Department, University of Victoria)

Dr. Morgan Price, Outside Member
(University of British Columbia)

ABSTRACT

Non-adherence to prescribed medication negatively impacts healthcare sector and over 50% of patients fail to adhere to their prescribed medication regimens. The ramifications of non-compliance are extensive, impacting both healthcare systems and patients alike. Within healthcare systems, non-adherence has been identified as a primary factor contributing to approximately half of all hospital admissions related to medication, incurring annual costs in the billions of dollars. For patients, non-adherence manifests in heightened risks or severity of ailments, potential relapses, and, in severe cases, mortality. The underlying causes of non-adherence are diverse, with unintentional omissions, prominently attributed to forgetfulness, accounting for over one-third of all instances of non-adherence. A viable remedy often involves leveraging reminder systems, which have demonstrated relative effectiveness, particularly when complemented with human support.

This dissertation aims to address the issue of non-adherence resulting from challenges in integrating prescriptions into the demanding, active lives of patients. The study delves into ascertaining optimal approaches to support self-management of prescriptions through the utilization of calendars. The research comprises three studies assessing the usability of calendars for effective medication management. These studies encompass; formalizing prescriptions through temporal reasoning frameworks (Study 1); exploring diverse methods of presenting medication information within calendars alongside other events (Study 2); and evaluating the calendar prototype to gauge its efficacy in facilitating medication management (Study 3). Study 1 resulted in the proposition of Simple Temporal Networks (STP) in formalizing prescriptions. Insights from Study 2 informed the proposition of design guidelines encompassing aspects such as (i) employing a familiar design, (ii) facilitating patients' self-reflection on medication adherence, (iii) ensuring medications do not clutter the calendar interface, (iv) empowering users to control the privacy of medication information within the calendar, and (v) enabling the sharing of medication-only calendars. Study 3 validated the usability of the calendar and its efficacy to support medication management, personal reflection, and schedule refinements.

The findings from these studies underscore the potential for calendars to be designed with both expressiveness and efficiency to support medication prescriptions effectively. Additionally, patients utilizing multiple medications expressed receptiveness toward adopting calendars as a means of managing their medication regimens.

Table of Contents

List of Tables

List of Figures

ACKNOWLEDGEMENTS

I would like to express my deepest gratitude and appreciation to my esteemed supervisors, Jens Weber and Charles Perin and my external supervisor Morgan Price, for their invaluable guidance and unwavering support throughout my research journey. Their extensive knowledge, expertise, and patience were instrumental in shaping this dissertation, especially considering my initial lack of research experience.

I would also like to thank Lead Lab and Vixi Lab members for their support during my studies. I learnt a lot from you guys!

Special gratitude to my wife (Memory Kaoma) and kids (George, Mweni, Lubuto, Kasalwe), who have gone through a lot to let me pursue my dream. I am forever indebted to you guys. I will surely repay you for my absence someday!

DEDICATION

To my beloved mother, Emma Changwa Lengwe, whose unwavering support and sacrifices have shaped my life.

Chapter 1

Introduction

1.1 Background

Adherence refers to the extent to which a person's behavior of taking medication, following a diet, or executing lifestyle changes corresponds with agreed recommendations from a health care provider [3]. These agreed-upon recommendations are called Prescriptions. Prescriptions provide detailed instructions of what medicine should be administered, in what formulation and dose, by what route, when, how frequently, and for how long [4]. A prescription may specify one or more medications to be taken by a patient. Once a prescription is given, the recipient is expected to adhere to the prescribed drug. Adherence means taking the correct dosage of each prescribed regimen at the right times, in the proper manner and for the prescribed duration [5]. Computationally, a patient is considered adherent if their medication adherence, calculated as the percentage of the number of pills absent in a given period to the number of pills prescribed by the physician in that same period, is at least 80% [6, 7]. The assumption is that the number of pills absent was administered.

1.1.1 The Problem of Non-adherence

Non-adherence to prescribed regimens is a big problem in healthcare. On average, 50% of patients do not take their medication as prescribed [6, 7]. Poor adherence has been repeatedly documented for various conditions, drugs, and patient populations [8]. Non-adherence hurts both non-adherent patients and healthcare systems. For the patients, poor adherence to treatment regimens carries consequences such as increased risk of disease severity, recurrence of the disease, drug resistance, increased outpatient visits, reduced quality of life, hospitalizations, and mortality [9, 10, 11, 7]. For healthcare systems, poor adherence reduces optimum clinical benefits and affects the overall effectiveness of the systems [3]. The cost of healthcare resulting from non-adherence is also quite significant. Non-adherence to medication prescription is responsible for about 33% to 69% of all medication-related hospital admissions, costing billions per year [7, 5]. A study conducted in 2015 indicated that the annual adjusted disease-specific economic cost of non-adherence per person ranged from $949 to $44190 [12]. In Canada, medication non-adherence accounts for 5% of hospital admissions and physician visits, resulting in an additional $4 billion in health care costs annually [13].

1.1.2 Factors Affecting Adherence

Factors that lead to non-adherence are diverse, covering patients, physicians, and healthcare systems. Medication adherence is, however, primarily in the domain of the patient [3]. Patient-related factors are classified into two categories: intentional and unintentional omission [14, 15]. Fear of side effects, cost implications, too many medications, failure to perceive benefit, and mistrust of prescribing physician are cited among intentional factors contributing to non-adherence [3, 6, 16]. In unintentional omission, patients forget to take their medications [17, 14]. Forgetfulness is one of the most common factors affecting adherence and accounts for about 30% of all non-adherence cases [15]. Patients sometimes forget to take their medications without external influence [15]. At other times, forgetfulness is driven by the failure to effect lifestyle changes that embrace prescription schedules[18, 19]. Patients wish to take their medications. However, working between busy daily schedules and consistently taking medication at set times becomes difficult to manage, resulting in missed doses.

1.1.3 Problem Statement

The focal point of my research revolved around addressing the issue of non-adherence stemming from challenges in effectively integrating prescriptions into many patients' busy and active lifestyles. Many conducted studies aim to mitigate non-adherence primarily attributed to forgetfulness by employing reminders. These reminders are strategically designed to prompt users to adhere to their prescribed medication regimens at predetermined intervals [20, 21, 22]. However, as aptly expressed by Tabi et al., efforts to enhance adherence should transcend the simple objective of ensur-

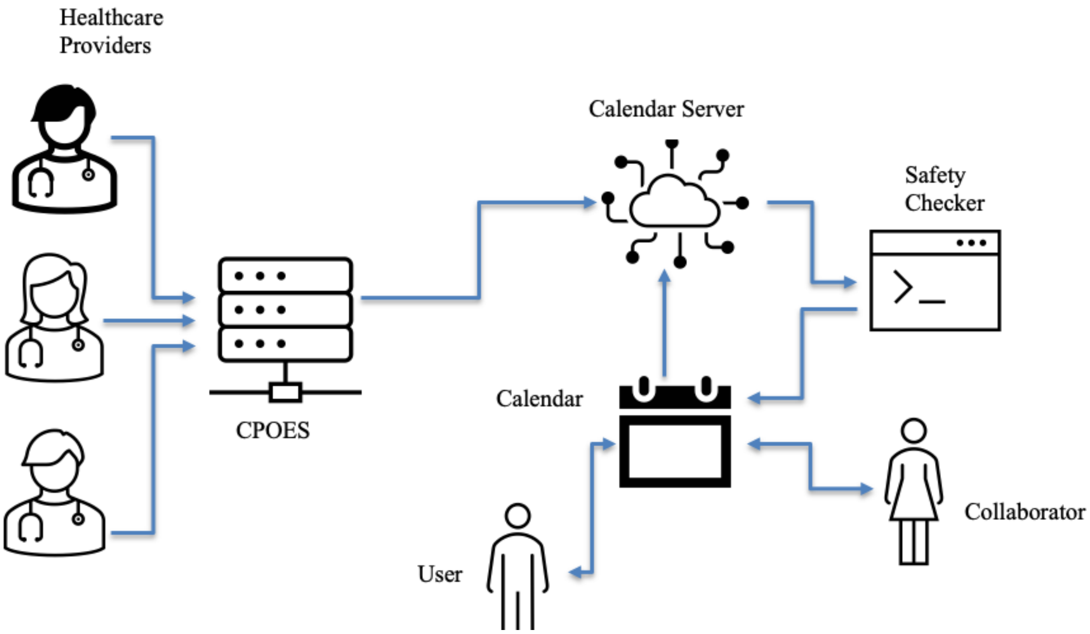


Figure 1.1: A diagram showing the flow of information between healthcare providers and patients. Prescriptions are retrieved from a Computerized provider order entry (CPOE) and stored on the calendar server. The patient refines the schedule, and the Safety Checker validates refinements to ensure harmony with the original prescription.

ing patients' compliance with their medication regimens. Instead, the focus should broaden to aiding patients in efficiently managing their medications [23]. Hence, the main problem addressed in my dissertation is how medication management could be integrated into tools used to manage daily activities. For this, we adopted calendars, pivotal tools for effectively managing day-to-day activities [24].

1.2 Research Questions

Based on this foundational understanding, we advocate for the formalization of theories, the formulation of methodologies, and the development of tools to aid patients in effectively integrating prescriptions into their daily routines. The primary objective is to assist patients in seamlessly incorporating their prescribed treatment regimens into their preferred lifestyles while adhering to the various constraints outlined in their original prescriptions.

The work presented in this dissertation centers on the creation of a system that provides:

1. Mechanisms for encoding prescriptions.
2. Methods for generating, managing, and refining prescription schedules, with active involvement from the patient
3. Platforms to support user reflection for improved collaboration and adherence.

As illustrated in Figure 1.1, the envisaged process starts with physicians from different healthcare providers creating prescriptions for a patient. The patient receives the harmonized prescriptions, free from conflicts. A structured schedule is derived from these prescriptions, allowing patients to refine their daily routines. However, each adjustment must undergo safety validation checks to ensure that the refinement agrees with the specifications in the original prescriptions.

To guide the research in the right direction, the following research questions were formulated:

1. How can we formally represent care plans reflecting multiple prescriptions from multiple health care providers?
    (a) What data modelling techniques exist that would support prescription schedules?
    (b) How can temporal reasoning formalisms be employed for reasoning with prescription schedules?
2. How can visualization be used to communicate and inform positive adherence behaviour in a team care setting?
    (a) How can prescription management be effectively integrated into electronic calendars?
    (b) How can medication data be visually differentiated on the calendar from normal calendar entries?
    (c) How can unsafe medication interactions be effectively visualized on the calendar?
3. How can calendars be extended to include prescription management?

(a) Which medication-related data and operations should the calendar support?

(b) How can patients who refine prescription schedules do so within the specified safety constraints of the prescription?

(c) How can visualization be used in the calendar to support self-reflection?

The main objective of this research was to develop theories, methodologies, and practical tools to assist patients in effectively managing their medication within a safe, self-reflective, and collaborative healthcare system. The research delved into conceptualizing the diverse processes involved, spanning from the prescription's inception to the treatment regimen's conclusion. The approach employed during the study underscored adherence as a collective endeavour, considering the critical roles played by physicians, patients, and informal caregivers in the treatment process. Central to this approach was developing a calendar tool to empower patients in prescription management.

1.3 Methods and Results

Three studies were conducted addressing the research questions and objectives indicated above. The undertaken studies covered prescription formalization, calendar design, and calendar implementation. Different research methods were used for each one of the three studies. These are discussed below. The methods used, and a summary of the results are presented for each study.

1.3.1 Study 1 - Prescription Formalization

In this study, we employed two approaches: firstly, a comprehensive review of temporal reasoning frameworks, and secondly, an experimental evaluation of specific frameworks for formalizing prescriptions. The study involved finding the optimum way of formalizing prescriptions for easy integration into medication management support tools. A literature review was conducted to determine already existing frameworks that are generic and expressive enough to support prescription management. As a prerequisite, the framework should support prescription-related data such as Drug name (D1), Drug dose (D2), Drug dose frequency (D3), Duration (D4), and Drug indications (D5). The framework should also support algorithms for reasoning with the prescription's temporal features and mechanisms for handling unsafe drug interactions within refinements of prescription schedules.

The Framework

The literature review considered many frameworks such as Allen's Integer Algebra [25], Villain's Constraint Propagation [26], Malik's Reasoning in Time and Space [27], and Detcher's Temporal Constraint Satisfaction Problem (TCSP) [28]. The Simple Temporal Problem (STP), a limited TCSP network, in which time points are represented as nodes of a graph and the intervals between the time points (or events) as weighted edges were adopted [28]. In the formalization of prescriptions, an STP fitted well since we were dealing with time points (which could represent administration times) and intervals (which could represent distance between administration times). STP also supports binary constraints (Intervals are bidirectional), which are expressive enough to support reasoning about unsafe drug interactions. Other frameworks similar in operation to STP, but with extensions, were considered. These included STP with preferences (STPP) [29], Disjunctive Temporal Problems (DTP) [30], and DTP with uncertainties [31].

Study Procedure

A dedicated system was developed to assess Simple Temporal Networks (STP) 's efficacy in formalizing prescription data. This system not only implemented STP but also could support multiple prescriptions. To facilitate testing, a grammar was designed based on Restricted Natural Language Grammar (RNLG) (refer to Appendix A) to enable prescription authoring. The RNLG allowed for the specification of prescriptions using requirements (for medication administration) and restrictions (for constraints about medications).

The implemented parser, coded in Python, was responsible for interpreting the prescription and generating the associated STP, which was then stored in a Neo4J database. Subsequently, algorithms were developed to query the graph, create the medication schedule, and verify constraint adherence.

Performance evaluation of the model involved rigorous testing through the introduction of multiple prescriptions and deliberate conflicts between selected medications. This testing primarily focused on two core modules: the parser and the consistency checker. A prescription scenario was designed, necessitating the patient to take a medication once a day for a specified duration. The medication was subject to a constraint mandating a minimum interval of 20 hours between doses. The administration period was changed incrementally from 1 to 30 days. Performance data for both modules was recorded twice for each medication administration day, enabling a thorough assessment of the system's efficiency.

1.3.2 Results

The results of the study revealed that the parser exhibited a linear time complexity of $\mathcal{O}(n)$, where n represents the input size. In practical terms, this translated to processing schedules of approximately two months within around 28 seconds. This time encompassed the system's capability to generate a schedule based on a given prescription. The running time was meticulously measured in relation to the number of nodes generated from the prescription. It's noteworthy that reporting the running time in this context against the number of nodes ensured that the experiment outcomes would remain consistent even with an increase in the number of medications taken per day, as exemplified in the prescription.

On the other hand, the time complexity of the consistency checker was polynomial, specifically $\mathcal{O}(n^3)$, where n denotes the input size. The running time for this operation was observed to be within the order of milliseconds, aligning with the performance characteristics of Floyd-Warshall's All-pairs-shortest-path algorithm [28], which served as the foundation for the consistency checker. Both measurements involved network communication with a Neo4J database.

1.3.3 Study 2 - Exploring Design Variations

In this study, we created personas, employed prototyping, and conducted user studies. In the study, we crafted three calendar prototypes, with each prototype being modeled after a distinct persona and showcasing a weekly medication schedule. The prototypes were static and featured design variations of layout, visual representation of medication entries, and representation of conflicts between medication entries. The main goal of this study was to assess the quality of design variations. Three usability requirements were tested, including (i) the users' ability to correctly and efficiently read the calendar's temporal features, (ii) the users' ability to accurately and efficiently identify the calendar's medication entries, and (iii) the users' ability to identify conflicting medication entries in the calendar correctly.

Study Procedure

We evaluated the designs with twelve participants (P = 12). Each participant completed 16 tasks for each of the calendar designs. Questionnaires were also administered to the participants, measuring their design preferences and likelihood of adopting calendars for medication management. Each study session lasted one hour, and we collected audio and video streams. The study's data was analyzed qualitatively (for user sentiments) and quantitatively (for task completion accuracy).

Study Results

The results obtained from our study indicate the viability of designing calendars to support medication prescriptions while maintaining a familiar user interface. Notably, eight out of twelve participants expressed a favourable inclination towards utilizing such a calendar to manage their medication schedules, ranging from somewhat likely to very likely. Furthermore, these findings significantly influenced the identification of five design guidelines that an integrated calendar system should encompass: incorporating a familiar design (DG1), ensuring a clutter-free interface (DG2), allowing personalization options (DG3), facilitating personal reflection (DG4), and implementing features to highlight information for user attention (DG5). It's important to note that these design guidelines were conceived based on the insights derived from this study and played a foundational role in shaping the subsequent phase of our research. Overall, users could read the calendar's temporal features, identify the calendar's medication entries, and identify conflicting medication entries in the calendar.

1.3.4 Study 3 - Calendar Implementation

In this study, we developed personas, implemented the calendar, and carried out user studies to assess its effectiveness. In the study, we implemented a calendar that seamlessly integrated events and medication management. Leveraging the FullCalendar framework [32], we developed a prototype that featured robust support for three critical aspects: (i) efficient scheduling of medication, (ii) effective handling of conflicts in medication scheduling, and (iii) facilitating self-reflection.

The designed prototype was a desktop calendar, carefully modelled to resemble popular calendars like Google Calendar [33]. The primary objective of this study was to evaluate the calendar's usability, specifically in the context of medication management. We devised a usability assessment focusing on four key requirements:

1. **Efficient Interpretation of Calendar Features:** Assessing users' proficiency in accurately interpreting both event-related and medication-related features presented in the calendar.
2. **Effective Management of Entries:** Evaluating users' ability to efficiently add, modify, remove, and personalize routine events and medication entries within the calendar.
3. **Conflict Resolution Skills:** Examining users' proficiency in identifying and resolving conflicting medication entries within the calendar.
4. **Comprehension of Visualizations and Adherence Summaries:** Analyzing users' capability to comprehend visualizations and effectively summarize adherence levels as presented in the charts.

Study Procedure

We evaluated the calendar implementation with twelve participants (P = 12). Participants were presented with a calendar that reflected the life of a persona with five medications and a relatively busy schedule. Participants were required to perform numerous tasks spanning medication management, conflict resolution, and self-reflection over one hour. The speed and accuracy of task completion were recorded for selected tasks. For some tasks, user sentiments were also recorded. Participants' data was collected through text, Audio, and Video streams. The collected data was analyzed quantitatively and qualitatively.

Study Results

The study's outcomes affirm that managing medications within a calendar interface can be accomplished with a similar level of proficiency required to handle calendar events. While specific medication-related tasks took slightly longer than pre-existing calendar tasks, participants capitalized on their familiarity with calendar interfaces to navigate the new medication management features successfully. Participants demonstrated the ability to manage medications effectively, including the correct and efficient resolution of conflicts arising from potential unsafe refinements of medication schedules. However, it is essential to note that the self-reflection component within the calendar posed challenges for participants. Insights and levels of reflection were limited, indicating a need for further enhancements or guidance in promoting greater engagement with self-reflection features. Despite these challenges, the study provides valuable insights into the potential and feasibility of seamlessly integrating medication management within standard calendar interfaces.

1.4 Conclusion

This research focused on improving medication adherence by integrating prescriptions seamlessly into patients' daily routines through digital calendar systems. The study comprised three interrelated investigations: prescription formalization, calendar design, and calendar implementation. Research methodologies included Literature Review, Personas, and User Studies. The findings indicated the viability of managing medications through familiar calendar interfaces, despite encountering challenges, particularly in self-reflection. The system, implemented using FullCalendar, underwent targeted performance evaluations, demonstrating the efficiency of the prescription manager and consistency checker modules. The study also assessed the effectiveness of formalizing prescription data using the STP approach. The resulting system showcased competence in handling multiple prescriptions and ensuring compliance with diverse constraints. In summary, this research was dedicated to advancing medication management, promoting adherence, and utilizing technological solutions to seamlessly integrate prescription management into patients' daily lives.

Chapter 2

Fundamentals and Literature Review

2.1 Overview

This chapter provides a review of literature focusing on the subject of non-adherence to medication, encompassing key aspects such as the underlying causes, consequences, methods of measurement, and interventions devised to address non-adherence to medication regimens. Additionally, the chapter offers a general overview of adherence-related technological intervention, including discussions on prescription formalization and prescription calendaring. It's important to note that literature exclusively centered on prescription formalization, visualization, calendar design and implementation, and self-reflection is not presented in this chapter. Such specific literature is explored in subsequent chapters, each featuring a dedicated related works section relevant to the particular subject under consideration.

2.2 Introduction to Adherence

Adherence is defined by the World Health Organization (WHO) as the extent to which a person's behaviour of taking medication, following a diet, or executing lifestyle changes corresponds to the agreed recommendations from a healthcare provider [3]. The agreed-upon recommendations, called Prescriptions, provide detailed instructions of what regimen[1] (or course of action in terms of lifestyle changes) should be given

---

[1] In my work, I concentrated on medication prescriptions. There are instances when non-medication regimens such as physical exercises were incorporated into user studies. However, the emphasis was on medication regimens. The use of the word *regimen* in this dissertation, therefore, implies *medication regimen* - unless explicitly qualified.

to the patient [4]. The instructions detail the formulation and dose, administration route, administration cycle, frequency, and regimen duration [4].

A prescription often specifies one or more regimens to be taken by a patient. Once it is given, the recipient is expected to adhere to the prescription. Adherent (or compliant) means taking the correct dosage of each prescribed regimen at the right times, in the proper manner and for the prescribed duration [5]. Computationally (with consensus challenging to arrive about), a patient is considered adherent if their medication adherence level, calculated as a percentage of the number of pills absent in a given period to the number of pills prescribed by the physician in that same period, is at least 80% [6, 7].

2.3 Types of Non-adherence

Adherence, in simple terms, refers to how closely a person's medication-taking behavior aligns with what their healthcare provider recommends. When individuals deviate from these recommendations, it's typically categorized as non-adherence. Non-adherence can be broken down into four key categories, each shedding light on different aspects of this behavior [34, 35, 36].

Primary Non-Adherence: This situation arises when a patient doesn't even take the first step to fill a prescription or obtain the medication prescribed by the healthcare provider. It is essentially a case of not getting started with the treatment.

Initiation Non-Adherence: Here, the patient takes the initial step of getting the prescribed medication but does not follow through with taking it as directed. It is like having the medication in hand but not using it as recommended.

Persistence Non-Adherence: In this category, patients begin taking the medication but, for various reasons, decide to stop without consulting a healthcare professional. It's discontinuing a treatment that was initially started.

Implementation Non-Adherence: Implementation non-adherence encompasses a range of behaviors where patients do not follow the prescribed medication regimen precisely. This can include skipping doses, taking medication at the wrong times, or taking the wrong doses. It also covers instances where patients take either more or less medication than prescribed within a specific time frame.

In my work, I specifically focused on persistence non-adherence, with a particular emphasis on cases where this non-adherence is unintentional, resulting from accidental omissions or lapses in the medication regimen. This type of non-adherence can have significant implications for patient outcomes, and addressing it is a critical aspect of healthcare management.

2.4 The Problem of Non-Adherence

Non-adherence to prescriptions is a big problem in healthcare. The WHO calls it a "worldwide problem of striking magnitude" [3]. On average, 50% of patients do not take their medication as prescribed [6, 37]. This is true even for Canada. A 2015 Canadian Pharmacists Association (CPA) survey found that 30% of patients stop taking their medications as instructed by their healthcare providers, and about one in four people do not fill their prescriptions or take less medication than prescribed [38]. Poor adherence has been documented for a wide range of conditions (with adherence decreasing as the number of medical conditions increases [39, 40]), number of medications (with adherence reducing as the number of medications increases [41, 42]), and patient populations (with adherence higher for older population [41, 8]). Non-adherence hurts both healthcare systems and patients who are non-adherent. For the patients, poor adherence to treatment regimens carries consequences such as increased risk of disease severity, recurrence of the disease, drug resistance, increased outpatient visits, reduced quality of life, hospitalizations, and mortality [9, 10, 11, 7]. For healthcare systems, poor adherence reduces optimum clinical benefits and affects the overall effectiveness of the systems [3]. The cost of healthcare resulting from non-adherence is also quite significant. For instance, non-adherence to medication prescription is said to be responsible for about 33% to 69% of all medication-related hospital admissions in the United States alone, with a resultant cost of approximately $100 billion per year [7, 5]. A study conducted in 2015 indicated that the annual adjusted disease-specific economic cost of non-adherence per person ranged from $949 to $44190 [12]. In Canada, medication non-adherence accounts for 5% of hospital admissions and physician visits, resulting in an additional $4 billion in health care costs annually [13].

2.5 Factors Affecting Adherence

Factors affecting adherence are broadly classified into five categories by WHO. These are (i) socioeconomic factors, (ii) factors associated with the health care team and system in place, (iii) disease-related factors, (iv) therapy-related factors, and (v) patient-related factors [3]. These factors overlap and fall under three categories: patient-related, physician-related, and health system-related factors [6]. My research focused on patient-related factors. According to Brown et al., patient-related factors have a more significant influence on predicting adherence than provider-related or payment-related variables [43].

Several patient-related factors contribute to non-adherence. I will classify them into three categories: literacy factors, socioeconomic factors, and psychological factors.

2.5.1 Health Literacy Factors

Sub-optimal health literacy and lack of involvement in the treatment decision is considered one of the contributors to non-adherence [6]. Certain patients have limited knowledge about their illness and lack information regarding optimal disease management. Insufficient information about adverse effects, poor understanding of medication instructions, limited access to care, and lack of health information technology contribute to non-adherence [44, 6]. Another contributor is the lack of involvement in the treatment decision–making process. Patients feel excluded from important decisions about their health and hence lack the motivation to adhere to medication as prescribed [45, 46].

2.5.2 Socioeconomic Factors

The socioeconomic status of the patient affects adherence. High cost of medication, lack of access to pharmacy transportation, and long wait times at the pharmacy are cited among reasons contributing to non-adherence [47, 6]. Lack of support from the family is also a contributor [48].

2.5.3 Psychological Factors

Psychological factors contributing to medication non-adherence include intentional and unintentional omission [14, 15]. Intentional omission is where patients deliberately decide to stop taking their medication. This may be due to factors such as the fear of side effects, perceived wellness, depression, the fear of becoming too dependent on medication and having to take too many medications [49, 50, 51, 52]. In unintentional omission, patients forget to take their medications [17, 14]. Forgetfulness is one of the most common factors affecting adherence and accounts for about 30% of all non-adherence cases [15]. Patients sometimes forget to take their medications without external influence [15]. At other times, forgetfulness is driven by the failure to effect lifestyle changes that embrace prescription schedules[18, 19]. Patients wish to take their medications. But, working between busy daily schedules and consistently taking medication at set times becomes difficult to manage, leading to missed doses.

2.6 Measuring Adherence

Measuring adherence implies determining the level of agreement between the prescription and the implementation thereof by the patient [3, 5]. In the measure, the patient agrees with and follows the recommendation by the healthcare provider. The ratio of the medication taken to the medication prescribed is the measure of adherence. An adherence score of 80% and higher is considered good [6]. For some conditions, the value may be different. For example, a near-perfect adherence ($\geq$95%) to antiretroviral therapy (ART) is often cited as necessary for HIV viral suppression [53]. The level of deviation of the implementation from the prescription affects this score as the patient is expected to follow the recommended times and dosages. In practice, however, patients follow a schedule tailored to their convenience. This is explained in conceptual models such as the Adherence Interaction Model (AIM) [54]. In AIM, adherence is explained with three measures that consider the patient's preferences: *compliance*, *agreement*, and *persistence*. Compliance and agreement start with a provider-recommended medication plan (i.e., *prescription*). The patient's response to the prescription distinguishes between the two. Compliance measures the distance between the prescription and *description* (that is, the patient-enacted plan). Agreement measures the distance between the prescription and *conscription* (that is, the patient-adopted plan). Persistence is the distance between the conscription and description [54]. This explanation of the measure of adherence is essential. It emphasizes measuring adherence concerning the patient-enacted (or patient-adopted) plan, not the provider-imposed prescription. This explanation of measuring adherence (emphasizing determining the agreement between the healthcare provider's prescription and the patient's refinement) is employed in the rest of this dissertation, focusing on ensuring that the refinement by the patient does not violate constraints borne by the prescription.

Many methods are used to measure (or monitor) adherence. According to Brown et al. [6], these mainly fall into three categories: objective methods, subjective methods, and biochemical measurements. Subjective measurements are obtained by asking patients, family members, caregivers, and physicians about the patient's medication use. Objective measurements are obtained by counting pills, examining pharmacy refill records, or using electronic medication event monitoring systems. Biochemical measurements are obtained by adding a nontoxic marker to the medication, detecting its presence in blood or urine, or measuring serum drug levels [6]. Directly observing patients as they take their medication is considered the most accurate way of monitoring adherence [7]. Next to that is biochemical measurement. The most common method is self-reporting by the patient, a measure considered simple and effective [55, 56]. However, it is widely acknowledged that patients tend to accredit themselves with higher adherence levels.

2.7 Interventions to Improve Adherence

Medication adherence is primarily in the domain of the patient. While there is no generally accepted metric for adherence, there is consensus that considering adherence is important to ensure an effective treatment process. As former U.S. Surgeon General Dr. Koop expressed, "Drugs don't work in patients who don't take them" [57]. The secret to improving adherence lies in technological interventions designed to support patients in their medication-taking initiatives. A prerequisite for improving adherence is thoughtfully considering the factors contributing to (non-)adherence. Adherence can be improved by addressing some of the challenges already presented in Section 2.5. Lars et al. proposed patient education, improved dosing schedules, increased hours when the clinic is open, and improved communication between physicians and patients as four methods that can be used to improve adherence [7]. Improved education can help reduce many health literacy problems. Increasing the operating hours of clinics and improving communication between physicians and patients can help minimize non-adherence resulting from limited access to care. Improving adherence also involves combinations of behavioural interventions and reinforcements [58]. Behavioural interventions should supplement an increase in the convenience of care, educational information about the patient's condition and the treatment, and other forms of supervision or attention [59]. Interventions such as the use of pill boxes to organize daily doses, simplifying the regimen to daily dosing, and cues to remind patients to take medications can be used to improve dosing schedules and help patients whose non-adherence is as a result of psychological factors [7]. Below is a discussion of technological interventions that are employed to address the problem of non-adherence.

Figure 2.1: A regular Weekly Pill Organizer (left) and a smart pill-box (LiveFine Smart WiFi Automatic Pill Dispenser) paired to a smart phone (right) .

2.7.1 Smart Pill-Boxes

Smart pillboxes are electronic containers that are used to store medication. They come in many forms, such as pill holders, alarm-based pill holders, and pill monitoring devices [60]. Some pillboxes are used for organizational purposes and offer no electronic functionality (see weekly pill organizer in figure 2.1). Others are built with technology that automatically detects events (such as opening the container lid) and performs a predefined operation (such as reporting the action to a remote server) [61]. The user can program smart pillboxes to select times for administering the medication. They can record medication taking action and send reminders to the user when a predefined medication administration time is skipped without the action [62, 63]. Although most of these are not smart pill boxes, some pillboxes have become so advanced that they can even be paired with smartphones for monitoring purposes (see figure 2.1 showing a regular pill box and a smart pill box).

Many studies have been conducted to determine if Smart pillboxes can improve adherence [64, 65, 66, 67]. Results indicate that Smart pillboxes improve adherence, especially in patients who have to sort large amounts of pills each day [62, 68]. They have proved even more helpful when implemented with remote monitoring by healthcare providers [69].

2.7.2 Reminder Systems

Reminders are among the most common technological interventions to improve medication adherence [70, 37, 71]. Reminder systems address the problem of adherence resulting from forgetfulness. They are designed to remind users to take their medications at set intervals. Reminder systems take many forms such as video calls [72], voice calls [73], Short Message Services (SMS) [74], and Mobile Applications [75, 76]. Video and voice calls require human intervention to initiate the reminders. These two synchronous interventions are intended to improve patient care and support. Hence, although it requires more effort, video intervention is more effective than other reminder systems because of the patient support provided [72]. Mobile Applications are user-programmed to generate reminders at times predefined by the patients automatically. Some reminder applications allow the user to enter a prescription with information that should be used to trigger reminders. Other reminder applications are designed to work like alarm clocks, with no context information about the prescription. These applications often lack the involvement of healthcare providers and are mainly designed without end-user involvement [77].

Studies have been conducted to determine whether Reminder Mobile-Applications do improve adherence. They have covered various medication conditions, environments, and patient populations [20, 78, 79, 80]. Reminder systems improve adherence to varying extents [81, 82].

2.7.3 Calendar Packaging

Calendar packaging involves using an innovative unit-of-use packaging system in which the container for the medication is labelled with a day or date feature designed to provide a visual record of when the patient last took the medicine [83]. The packaged calendars may be digital (with an implementation like Smart Pill-boxed), physical (and stand-alone), or as part of a blister packaging. The user must indicate the last time they took their medication on the calendar. As in the other interventions, Calendar packaging has improved medication adherence [83, 84, 85].

2.8 Towards Self-Management

Reminder systems, Smart Pill-boxes, and Calendar-supported Blister Packaging have been proven to improve adherence in diverse setups where patients constantly forget to take their medications [69, 81, 82]. However, these technologies are not designed to improve dosing schedules that evolve around a busy lifestyle. They are intended to remind the patient about pre-set medication administration times. Improvement of dosing schedules is an intervention that directly addresses forgetfulness driven by the failure to effect lifestyle changes. In this instance, the patients do not just forget to take their medication. They have problems integrating medication-taking into their daily routine. Reminders can help support the cognitive demands of managing daily and future health tasks, but there is little understanding of how they fit into people's daily lives [86]. To illustrate this, consider a patient who is supposed to administer a medication at some point during the day. If that time coincides with an important event, reminding the patient to take the medication at the set time is not ideal. The ideal intervention would allow the patient to reschedule for an alternative time. The system should then confirm that the new schedule does not violate the temporal constraints of the prescription. Reminder systems lack semantical support for prescriptions. In short, they are not "prescription aware". Reminder systems should be much more than just systems that trigger reminders; they should support self-management of health. A class of applications called Personal Health Applications (PHA) attempts to fill this gap.

2.8.1 Personal Health Applications

Personal Health Applications (PHAs) go beyond conventional reminder systems by seamlessly integrating self-management into the realm of healthcare [87, 88, 89]. These applications empower patients to input their health data, including conditions, symptoms, and prescriptions, and provide essential tools for efficient health management. Functionality includes setting reminders, monitoring health data, and managing medication schedules. PHAs also offer features to monitor adherence to prescribed regimens, a crucial aspect of effective self-management [90, 91]. Notably, the self-management approach has shown promise in improving adherence to complex medication routines [92, 93]. However, existing studies on PHAs often overlook the seamless integration of personal health management into established lifestyle management tools. Instead, they develop standalone applications as proof of concepts [90, 94]. Incorporating personal health management into already familiar utility tools could potentially address adoption barriers associated with health technologies [95].

2.8.2 Designing to fit Into Lifestyles

To address non-adherence resulting from failure to effect lifestyle changes that evolve around the integration of medication into individuals' everyday activities, we must take advantage of the tools used to manage daily activities. For example, Calendars, already proven to effect positive change when deployed as part of blister packaging [83, 84], are among the most popular tools used in the management of daily schedules [96]. Calendars have been tested, and they have proved effective in health-related activities such as fitness tracking by individuals and home health management [97, 98]. Even so, research has yet to be conducted to investigate the possible integration of prescription management into calendars. The integration referred to does not entail building calendars for the sole purpose of managing prescriptions. Prescription management should be a feature of the same calendars, such as Google Calendar [33], that are used to manage all other events, reminders, and appointments. Popular calendars should be designed to support all PHA-related functionalities such as medication schedule management, reminder triggering, medication administration tracking, and providing safety checks for prescription refinement. In essence, we imagine a PHA that is calendar-based. To achieve this, there is a need to establish frameworks that can support the inclusion of prescriptions in the calendar. The calendars should also introduce recognizable visualization designs meant for medication-related entries. Research already exists that addresses underlying technologies such as formal modelling of prescriptions and on-calendar visualizations. I summarize these contributions here. A detailed analysis of the two technologies is presented in subsequent chapters.

Modelling Prescription Data

There are different ways in which researchers have attempted to model prescription data to solve problems related to prescription scheduling. Researchers have proposed systems based on graph transformation [99, 100] and mathematical modelling [101, 102] to write prescriptions, generate schedules, and check for unsafe interactions between medications. New prescription entry forms (such as APAMAT: A Prescription Algebra for Medication Authoring Tool [101]) are proposed using these approaches. After the prescription is supplied using a suggested format, formal models are introduced to prepare schedules and determine the drug-interaction safety of the schedule. Researchers have also attempted to model prescriptions as Constraint Satisfaction Problems (CSP). Anselma et al. considered a prescription an example of a repeating clinical guideline and represented it as a Simple Temporal Problem (STP) [103]. The representation of the prescription as an STP brings with it an exposition of temporal reasoning algorithms, such as the algorithm for consistency checking, which can be used to determine whether a given graph is consistent [28]. These formalisms (discussed in detail in Chapter 4) have been used mainly as proof of concepts (targeting a specific functionality) and have not been proposed as PHAs that can be integrated into general utility tools like calendars. This is one of the problems addressed by our Collaborative Health Adherence Optimization System (CHAOS).

Scheduling Prescription Data

Scheduling of medication on calendar-like systems has been a subject of past research. Self-management Medication schedules, used for out patients, are laid out in tabular format and indicate details about the medication (name and dosage) and a daily routine of medication administration [104, 105]. They are used to create calendars for medication administration. The schedules are mostly week-based. Rich visual cues such as images and icons represent the calendar's temporal elements (when medication is supposed to be administered) and the list of medications. The management of prescriptions using illustrated medication schedules has been studied and proven effective [106, 104, 107].

2.9 Conclusion

Adherence, as defined by the World Health Organization (WHO), refers to the degree to which an individual's actions, encompassing medication intake, dietary habits, or lifestyle adjustments, align with agreed-upon recommendations from a healthcare provider. Within healthcare, non-adherence to medication presents a significant challenge, affecting approximately 50% of patients. Patients deviate from medication adherence due to a range of factors, including literacy issues (such as a lack of comprehension about their condition and information regarding optimal disease management), economic circumstances (such as an inability to cover medication costs), and psychological aspects (such as forgetfulness). Effective interventions like smart pill-boxes, reminder systems, and calendar-based blister packaging have successfully addressed instances of non-adherence arising from forgetfulness and dose management. However, these systems often fail to mitigate non-adherence resulting from the struggle to integrate medication into busy lifestyles. Household tools that seamlessly integrate prescription management are essential for addressing this aspect. While Personal Health Applications (PHAs) are crafted to provide health management functionalities, including prescription management, they are typically disseminated as standalone tools and may not comprehensively address all the complexities surrounding medication management. This dissertation endeavors to bridge this gap by proposing a calendar system that seamlessly integrates prescription management.

The literature review in this chapter provides a broad overview of the subject area, particularly focusing on topics such as prescription formalization and prescription calendaring. A more exhaustive review is presented in the subsequent chapters. Chapter 4 delves into extensive details regarding prescription formalization, while Chapter 5 explores additional literature related to prescription visualization. PHAs are thoroughly discussed in Chapter 6 and Chapter 7.

Chapter 3

CHAOS System Overview

3.1 Introduction

The CHAOS (Collaborative Health Adherence Optimization System) project was aimed at developing a tool that can be used to support medication adherence. The tool developed is a calendar that supports the scheduling and management of both regular events and prescription entries. The targeted users are patients who have difficulties managing their prescriptions alongside busy and dynamic schedules. When event schedules suddenly change, medication entries in the calendar are also liable to be influenced to change. When that happens, the calendar should provide checks that ensure the safety of the schedules (based on the provider-recommended prescription). CHAOS fosters collaboration and provides a platform that patients can use to optimize their adherence.

Optimization

The main goal of CHAOS is to support users in the management of their prescriptions. The system allows the user to schedule their medication, alongside other events, in a calendar. Integration of prescriptions into the calendar allows the user to fit their medications into different activity cycles. Users may move things around with a busy calendar to find appropriate time slots to schedule their medication. When this happens, checking the safety of the new schedule is essential. To address this, the calendar provides constraint checks to ensure that the changes made to the schedules are within the requirements of the prescriptions. Additionally, when using the calendar, patients may track their medication-taking behaviour. This record, used alongside other user-supplied information in the medication tracking process, can explain past medication-taking behaviour and transform the schedule to optimize adherence.

In CHAOS, the patient manages the prescription using the calendar. The prescription is delivered to the patient by the healthcare providers through a computerized provider order entry (CPOE) system. The system fetches the prescription and generates a schedule (prompted by the user) based on the prescription. The patient may share the calendar with other users, such as healthcare providers and informal caregivers. The interactions involved, together with participating entities, are explained below.

Collaboration

CHAOS allows collaboration between the patient, healthcare providers, and informal caregivers. In adherence terms, collaboration commences when the healthcare provider determines the patient's condition and chooses an intervention for them. It continues when the prescription is agreed upon by the patient and the healthcare provider and throughout the prescription implementation period. CHAOS supports collaboration during the implementation period. The system offers mechanisms for tracking and visually representing the patient's medication-taking trends. These visualizations may be used to inform and assess outcomes by the providers during patient visits. CHAOS also supports the sharing of calendar schedules. The patient may grant access to the calendar to both formal and informal caregivers for collaboration purposes.

3.2 The CHAOS Model

I have used the iStar modelling framework (Version 2.0) [108] to present the various components that make up the CHAOS system. The iStar strategic rational (SR) model consists of actors (active entities that aim at achieving goals), association types (links between actors), intentional elements (things actors want), and social dependencies (that is, relationships) between actors and intentional elements [108]. The iStar notation represents actors as circles, goals as ovals, quality goals as curved cloud-like shapes, tasks as hexagons, resources as triangles, and dependencies as relationships linking a depender role to a dependee role via a dependum (such as goal, resource, quality, or task). Figure 3.1 shows these components being used to model the CHAOS system. In the following sections, I will present the components that form CHAOS. I will concentrate on explaining roles (a type of actor). Associations and dependencies will be presented in the discussion within the participating roles. Please refer to figure 3.1 for a detailed view of all association types, intentional elements, and social dependencies.

3.3 System Roles

There are six roles in the CHAOS framework. Four are user-based roles, and two are application-based roles. The four user-based roles are Patient, Calendar User, Healthcare Provider, and Social Peer. The two application-based roles are Calendar App and Calendar and Medication App. The roles of the Calendar User and Patient have been separated to make a distinction between goals and quality elements associated with them. The separation of roles has also been applied to Calendar App and Calendar and Medication App for the same reason.

3.3.1 Calendar App and Calendar User

Calendar App and Calendar User are the default roles available in conventional calendars. The Calendar App (app) offers the Calendar User functionalities associated with everyday calendars. The app supports users' primary goal of organizing their general work and life schedules. The calendar helps the user organize a general schedule of events (allowing the user to add, edit, delete, and personalize events) and the pursuance of commitments (using notifications and reminders). The calendar is structured to manage events as daily, weekly, or monthly workflows. The app guarantees privacy by ensuring only authorized users can access a particular user's calendar. The calendar also supports the user's need for flexibility by allowing them to update their schedules frequently and efficiently.

3.3.2 Calendar and Medication App

The active extension CHAOS introduces to the Calendar App role is the Calendar and Medication Application (CMA) role. CMA is an application that allows patients to manage their prescriptions within the confines of everyday calendars. It is an

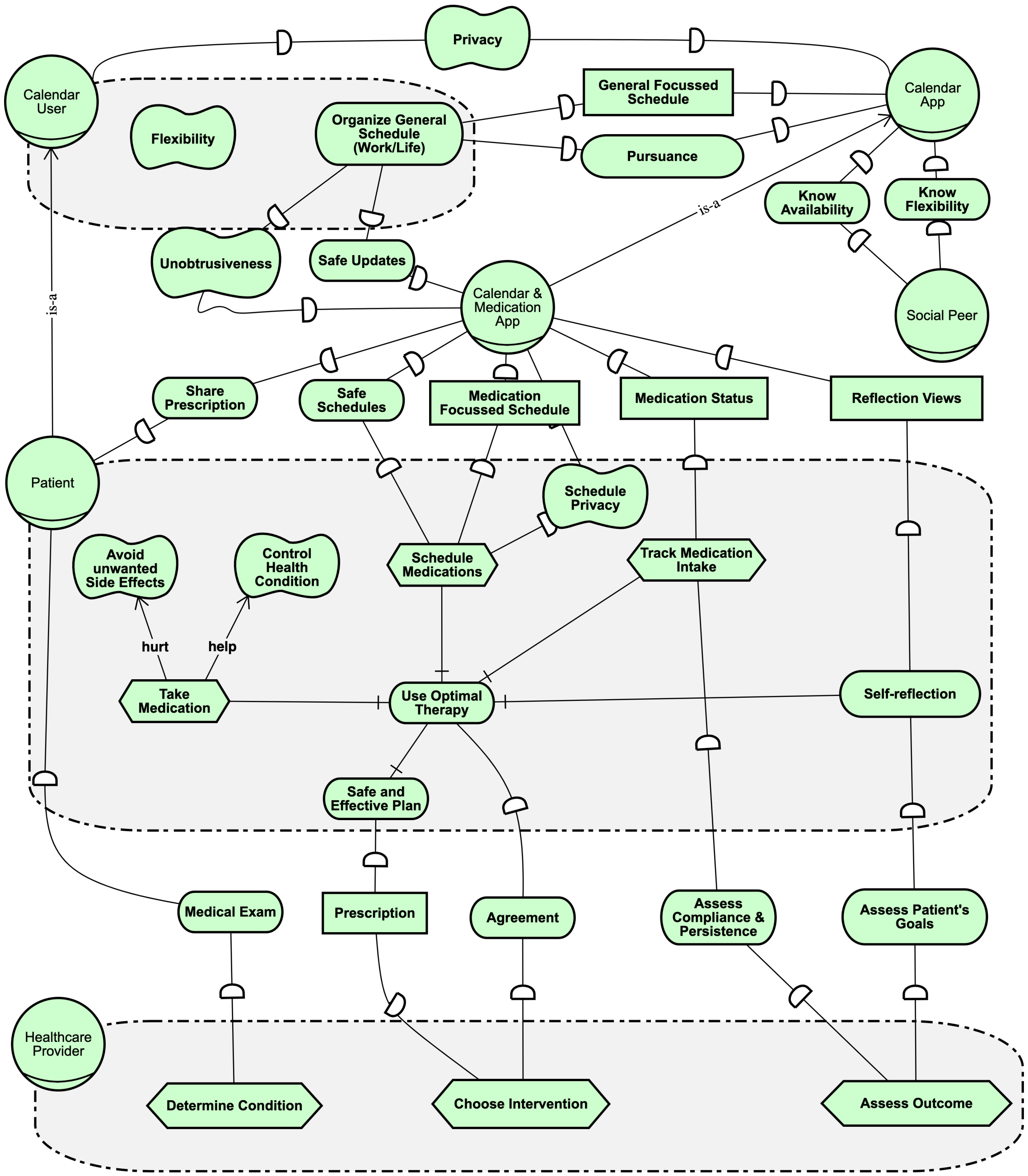


Figure 3.1: Strategic Rational Model for the CHAOS system

instance of the Calendar App and hence the is-a relationship in the model (see figure 3.1).

CMA is designed to achieve three main goals: (i) ensuring the user maintains a safe medication schedule, (ii) facilitating safe updates to the schedules, and (iii) enabling the user to share the prescription with third-party stakeholders, including healthcare providers and caregivers. Scheduling events and medication entries in the calendar may introduce clutter in the display. CMA guarantees unobtrusiveness when using the app. Scheduling medication in the calendar may also temper the confidentiality of the prescription. CMA ensures medication privacy by allowing users to hide all medication-related data from the visual calendar display.

CMA provides three resources to the targeted users (the patients). These resources are Medication Focused Schedules, Medication Status, and Reflection Views.

Medication Focused Schedules

CMA allows the users to create schedules from the prescriptions. By design, the prescriptions should come from a CPOE. CMA authenticates with CPOE and retrieves prescriptions owned by the user. Once received, the system generates the schedule based on the prescription and stores the schedule in the calendar. CMA allows the user to optimize their schedules. The user can refine the schedule to fit in with their daily routines as part of the optimization process. For every refinement, the system checks to ensure that the changes to the schedule do not violate all constraints associated with the original prescription. Violations are brought to the attention of the user, who may then decide to rectify the refinement or ignore the warning.

Medication Status

CMA provides a platform that allows the user to track their medication-taking behaviour. Within the calendar, four statuses indicate the administration status of past medication entries. The statuses include taken (indicating that a medication entry was administered), missed (indicating that a medication entry was not administered), unsure (indicating that the administration status of a medication entry is unknown), or pending (indicating that no action has been taken yet, regarding the entry). Another attribute associated with medication statuses is a tag. Tags (user notes) may be included during the update of the medication status for past entries. They are used to add contextual information to past entries. The information may include symptoms, reactions, observed outcomes, rationale, or measured biomarkers. Dosage modification (indicating whether the user took the complete prescribed regimen or modified the dosage) tracking is also part of CMA.

Reflection View

CMA provides visualizations that can be used to reflect on past medication-taking routines. The visualizations show medication summaries for a specified period. Visualizations show an overview (expressed as a percentage for selected medication), weekly patterns, and a timeline (showing a calendar view with only status information) for a specified period. The visualizations are based on the statuses of past medication.

3.3.3 The Patient

The patient's quality goals are to control their health condition and avoid unwanted side effects associated with the treatment. The patient receives a prescription from the healthcare provider that indicates the medication they are supposed to take over a defined period. This prescribed medication, which the patients take to help control their health condition, may hurt their goal to avoid side effects if not administered according to recommendations. Patients have the goal to find the optimal therapy for them. This means that the therapy must be safe and effective (as assured by the healthcare provider's prescription) and work for them upon tracking the medication intake, reflecting on past patterns, and considering alternative schedules. Patients must schedule their medications using the Calendar and Medication App (CMA). The schedule entered in the calendar must agree with the prescription. Patients depend on the CMA to validate their schedules and ensure that the modified schedules are safe and effective.

3.3.4 Healthcare Provider

When it comes to drug-based therapies, healthcare processes can be characterized as iterative loops where the healthcare providers determines the condition of their patient (for example, by using lab tests or other forms of examination), choosing an intervention (for example, writing a prescription), and assessing the outcome of the intervention (considering the changes to the patient's condition as well as whether the patient adhered to the recommended therapy). Decisions on interventions have shifted from healthcare providers as the primary (or sole) decision-makers to a shared decision model, empowering patients as essential partners in the process [109]. Effective medical decisions require agreement between healthcare providers and patients.

The ability of healthcare providers to assess the outcome of therapy depends on their patient's reflection goal. At a more fundamental level, healthcare providers and patients may engage in a reflection process to revise a treatment or contextual factors like lifestyle choices, etc. For this process, the patient and the healthcare provider can collaboratively look at the tracked medication-taking routines and generated reflection views to inform patient-provider discussions.

3.3.5 Social Peer

Social Peers refer to users with whom the Calendar User collaborates within the Calendar App. Their need revolves around being informed about the Calendar User's availability for coordinating joint or individual meetings on the user's behalf. Moreover, they require insights into flexibility constraints to propose conflict resolutions effectively. It's important to note that certain events possess flexibility and can be rescheduled, while others do not allow such alterations. These constraints can become intricate, particularly in the context of drug therapies. While Social Peers can be provided access to the calendar, a crucial stipulation is imposed on medication-related data – it should be visually abstracted to ensure privacy. When a Social Peer gains access to CMA (Calendar Medication App), the patient should retain control over the visibility of sensitive medication-related information.

3.4 Conclusion

The CHAOS project aimed at developing a calendar that supports the scheduling and managing of regular events and prescription entries. Interactions between different system components have been presented using the iStar Modelling framework. The different parts of the system were presented as roles comprising the Patient, Healthcare Providers, Social Peers, Calendar App, Calendar User, and Calendar and Medication App (CMA). CHAOS is centred around the use of CMA by the patient (a calendar user). CMA extends the functionality of the Calendars App (which primarily is event management) by adding prescription management. The primary user of CMA is the patient. The Social Peer and Healthcare Provider support the patient with different levels of interactions. These two roles also have patient-moderated access to CMA (depending on the patient's collaborative needs). CMA was designed to support the optimization of adherence by the patient. Using the system, the patient can create a medication-focused schedule, track medication intake, and view medication history through reflection views provided by the system. CMA has mechanisms for checking the safety of the refinements made by the user to the medication schedule. By utilizing these features, the user can create safe and effective schedules. This can improve the patient's health condition and minimize unwanted side effects.

Prescriptions need to be formalized in preparation for their integration into calendars. The next chapter presents a semantic framework that supports prescription formalization. The framework supports the generation of prescription schedules and checking the safety of refinements made to the schedule by the patients.

Chapter 4

Prescription Formalization

4.1 Introduction

Formalizing prescriptions is a challenging process. Prescriptions have varied degrees of complexity (a mix of dosage, frequency, and route of administration) and contraindications that should be carefully considered by developers who seek to integrate them into patient support tools. The prescription process is a transaction between a patient and a care provider. The prescription indicates regimens tailored to the patient receiving it (the patient and the prescribing clinician are supposed to agree concerning the implementation of the prescription). The regimens (or prescription entries) come with different attributes such as name, dosage, duration, and administration cycles. These attributes inform the patient of the expected implementation of the prescription. Prescriptions can indicate a single drug regimen or multiple regimens. The latter is widespread in older patients and patients with multimorbidity [110]. The source and time of delivery of the prescription also vary. A patient can receive a single prescription from a single healthcare provider, multiple prescriptions from a single healthcare provider, or multiple prescriptions from various healthcare providers. This information about prescriptions is essential when contemplating the integration of prescriptions into any patient support tool.

With the increase in the number of drugs per given prescription also comes the increase in the likelihood of medication errors resulting from unsafe drug interactions [111]. Medication errors[1] occur when a patient takes two or more drug regimens in a time frame when the regimens should not be taken together. These errors usually

---

[1] Numerous categories of medication errors have been documented [4]. However, in this context, we are specifically referring to errors arising from unsafe drug interactions.

result in adverse effects on the patient's health. Medication errors are among the most common types of errors affecting patient safety, occurring most often at points of transition in care—on admission to a hospital, at transfer from one department to another, and at discharge to home or to another facility [112]. Drug regimens must be reconciled to avoid such medication errors and unsafe drug interactions. The process of identifying discrepancies in drug regimens prescribed in different care settings or at different time points, to inform prescribing decisions and prevent errors is called medication reconciliation [113]. Medication reconciliation errors occur across transitions in patient care. Over 25% of all medication errors in hospitals are caused by a failure to reconcile new prescriptions with ongoing home treatments [114]. Therefore, patients' tools that support the management of prescriptions should have mechanisms for identifying and communicating medication errors resulting from unsafe medication schedules and schedule refinements.

Considering all these constraints associated with prescription integration, this chapter presents a framework for the semantic representation of prescriptions. The proposed framework is generic and can be adopted by any patient support tool designed to work with prescription data. The framework supports three features: (i) modelling of prescription data, (ii) algorithms for reasoning with the prescription's temporal features, and (iii) mechanisms for handling unsafe drug interactions within refinements of prescription schedules. The main research question addressed in this chapter is how we can formally represent care plans combining multiple prescriptions. In answering the main research question, I also answer the following two sub-questions: (i) What data modelling techniques exist that would support prescription schedules? (ii) which temporal reasoning formalisms exist and how can they be employed in reasoning with prescription schedules?

To answer these questions, I begin by presenting a data model that can be used to represent medication prescriptions. By supplying all the required data in the model, as is the case when using Computerized Physician Order Entry Systems (CPOE) [115], a healthcare provider's authored prescription could be shared with the patient's medication management support tool. I propose a way of transforming the formulated prescription into a formal semantic model. The model, expressed as a Temporal Constraint Satisfaction Problem (TCSP) [28], can represent the entire prescription duration and time points of when the prescribed drugs should be administered. I also present an algorithm, based on Floyd-Warshall's all shortest path algorithm [116], that can be run against the model to check whether the schedules derived from the prescriptions have violations of associated temporal constraints.

The rest of the chapter is structured as follows: I begin with a look at past efforts that relate to formalization of Medication Prescription in Section 4.2. In Section 4.3, I give an overview of the Medication Prescription process, highlighting the core blocks of the proposed model. Using a sample model that facilitates the creation of prescriptions, I detail the process of transforming the prescription into a semantic model that represents the prescription. In Section 4.6, I explain how to verify the consistency of the model. I evaluate the framework's performance in Section 4.7 and conclude with a discussion and recommendation for future enhancements under Section 4.8.

4.2 Related Works

There has been significant research on formalizing the writing of medication prescriptions. There are many efforts that have been directed at simplifying the prescription writing process [117, 118, 119]. Many of these are concerned with automating the dispensing of drug regimens in medication dispensers and integration with Computerized Physician Order Entry Systems. I do not address these research efforts. Instead, I review approaches that formally specify and verify prescriptions. To do that, I begin with a research presentation that presents mechanisms for reasoning about data with temporal attributes. I conclude the section with research that has employed these temporal frameworks to address problems in healthcare.

One of the most popular frameworks for reasoning about time is that proposed by Allen [25]. Allen proposes a formal model, called Interval Algebra, for representing temporal relationships. With 13 possible relationships between events (such as before, after, and during), the framework provides a foundation for accurate and consistent reasoning regarding temporal relations between events. Because I intend to deal with temporal specifications that are disjoint between medications (such as; *Take A at 10 AM, Take B at 12 PM* instead of *Take B 2 hours after A* - with the second implying that if A does not happen, then B won't happen), this formalism provides a good foundation but is not expressive enough to support prescriptions fully. The foundation provided relates to approaches for specifying temporal constraints between medication entries between and within prescriptions. Allen's 13 relationship shows how events can be explained relative to the temporal position of other events. In prescriptions, however, events should first have temporal definitions of their own before temporal constraints can be specified that relate them to other events. This is the limitation implied by insufficient expressiveness. Villain et al. [26] also proposed a mechanism for dealing with temporal reasoning. Their work explored the challenges of reasoning about temporal relationships and provided insights into the representation, propagation, and optimization techniques for handling temporal constraints. Most relevant to my work, they proposed interval-based and point-based temporal representations of events. This separation of the two is something that my work leverages (*Take A at 12 PM* may imply the exact time but also an interval when drug administration is still considered adherent). Malik et al. [27] and Valdes-Perez [120] proposed mechanisms of representing temporal constraints using linear inequalities and suggested approaches for constraining the temporal distance between time points. Using linear inequalities to represent temporal constraints also appears in works such as Temporal Constraint Satisfaction Problems (TCSP) and Disjunctive Temporal Problems (DTP). These frameworks are discussed below.

The TCSP framework, proposed by Detcher [28], builds on previous works by Allen, Villain, and Malik [25, 26, 27] in which time points are represented using variables. Temporal information is represented by a set of unary and binary constraints, each specifying a set of permitted intervals. The framework allows for assessments of time differences between events. It includes algorithms for finding all feasible times that a given event can occur, algorithms for finding all possible relationships between two events, and algorithms for generating scenarios consistent with the information provided. A subset of the TCSP, called Simple Temporal Problem (STP), is also proposed. The STP admits at most one Interval constraint on any pair of time points. Advancements in the STP framework have also been proposed. Examples include the Simple Temporal Problem with Preferences (where the preferences may be indicated amongst the available solutions) [29], the Disjunctive Temporal Problem (DTP) (that admits more than one disjuncture in a temporal constraint) [30], and the DTP with uncertainty (which extends DTP to accounts for contingent events varying notions of controllability, moving away from consistency) [31]. Since I do not consider addressing either preferences or uncertainty (prescriptions are not “contingent events over which the agent has no control [29]”. That is, when a prescription is being prepared, the prescribing physician (agent) has control (albeit in agreement with the patient) over what medication they have to give to the patient. The patient doesn’t receive multiple “outcome uncertain” medications that leave them to rely on chance (“roll a dice”) in order to decide which one to take). The STP framework will be utilized to formalize the prescription later in the chapter. The rationale for settling on the STP was determined by the level of expressiveness of the framework regarding the functionality supported by the proposed model. Below, I present research that has utilized these or related frameworks to solve problems in healthcare.

APAMAT [101] (A Prescription Algebra for Medication Authoring Tool) is a medication authoring tool designed to complement CPOE by checking if multiple prescriptions contain unsafe interactions (also called drug conflicts). APAMAT allows an actor to describe a prescription using a Domain Specific Language (DSL) and checks for interaction conflicts. The tool then indicates whether there are unsafe interactions along with a drug administration schedule that can be fed into a drug dispenser. Although APAMAT supports a formal description of prescriptions and checks for unsafe interactions, it does not detect conflicts in the time scheduling of medicines that would or would not otherwise be taken together. APAMAT uses drug information external to the prescription to detect if a schedule contains medications that are not safe to be taken together. Like APAMAT, Weber et al. proposed a medication prescription authoring tool based on a graph-transformation system (GTS) [99]. This tool consists of an interface language to be used by clinicians to author prescriptions using a Domain Specific Language and a graph transformation system that transforms the prescription into a formal model that defines the semantics of what was proposed. The proposed tool cannot handle conflicts arising from drug-drug interactions. It does not provide a way of authoring drug prescriptions that should or should not be taken together in a selected period. Tsai [102] proposes a system of automatic medication dispensers. The system is designed to help users improve compliance by preventing misunderstanding medication directions and making medication schedules more tolerant of delay and negligence. The checks, called constraints, validate both dosage and interval limits. While dosage validation is not a feature employed in my work, interval checking, not only for a single drug but multiple drugs, is a feature that is addressed.

The research presented above addresses the issue of formalizing medication prescriptions. There is also research that addresses the broader area of the formalism of temporal representation. Anselma et al. [103] propose a formal approach for dealing with repetitions and constraints in clinical guidelines. They present an expressive representation formalism together with associated temporal reasoning algorithms. They rely on STP [28] to resolve inconsistencies in drug prescriptions. According to Anselma, the process of reasoning about prescriptions can be conceptualized as temporal reasoning or, simply, reasoning about time. Prescriptions indicate when drug regimens should be administered and intervals between two such administrations. Hence, one approach that can be employed to better understand how medication prescriptions could be represented in a semantic model is to look at it from the concept of time points and intervals and the relationships between such intervals. The approach used here has been adopted in my work. Anselma's approach has two significant differences from my approach: (i) I do not remove temporal demarcations (week, day, morning, etc.) from the prescription but maintain their temporal context, and (ii) I deal with constraints within a prescription and between multiple prescriptions. The approach by Anselma is also proposed by Combi et al. [121] as a way of modelling time for the management of or reasoning about time-oriented clinical data.

Similar works present ways of representing and reasoning with temporal data in a clinical setup. Asbru was a machine-readable language (with an associated interpreter) proposed by Shahar et al. [122]. The language was used to represent and annotate guidelines based on the task-specific ontology. It supported actions, plans, and states related to clinical guidelines. A Graphical User Interface (GUI) was used for data entry. Gordon et al. proposed a model called DILEMMA for representing guidelines and protocols in healthcare [123]. The proposed model detailed the knowledge (tasks, characteristics, and procedures) that should be part of a generic structure. It also comprised a set of constructs intended to allow any clinical guideline's knowledge content to be fully represented in a uniform style [123]. EON was a general-purpose software designed to interpret abstract protocol specifications and construct patient-specific treatment plans [124]. The software had provision for performing time-oriented queries on a time-oriented patient database [124]. Shiffman et al. proposed the Guideline Elements Model (GEM), a document model for practice guidelines that could store and organize the heterogeneous information [125]. The document used Extensible Markup Language (XML) to store and organize information contained in guidelines. ONCOCIN was an interactive decision-support system designed for use by physicians in the treatment of cancer patients [126]. ONCOCIN employed a reasoner (which managed the data-entry and consultation sessions), a knowledge base (representing the skeletal plans and refinement heuristics), and a patient database (represents patient data and therapy plans) to solve cancer-chemotherapy planning problem [126].

These works, especially Simple Temporal Problems, form the foundation of my work. In the next section, I present a formalization of prescriptions using the STP framework.

4.3 Process Requirements

The first step in the formalization of a prescription is to understand the relevant characteristics of a prescription. Once determined, I will then proceed to proposed methods for organizing and reasoning about prescriptions.

4.3.1 Prescription Characteristics

According to Ethier et al., [127], a prescription is made of the different parts which define the administration modalities for a given drug. The underlying building block is the drug prescription item. The drug prescription item comprises the drug administration, healthcare objective, and drug distribution specifications. The drug administration specification is the part that contains information that pertains to drug administration. This information includes the drug product specification (which indicates the name of the drug), drug dosage specification (which indicates the dosage, administration route, and dosing conditions), and drug course specification (which indicates the starting condition and the duration of the prescription). Kumar et al. [128] summarize these building blocks as superscription (directive to take), inscription (name and dose), subscription (directions to the pharmacists), and signature (instructions for Patient). Fox [129] uses the naming convention: drug name, drug dose, drug dose units, drug dose frequency, duration (comprising start and end date), and indication. Like others (e.g.[76, 130]), when deciding which convention to adopt, I adopt the latter classification by Fox [129] because of the simplicity and directness of the naming convention used.

Following this classification, both the data and temporal reasoning model will support medication entries with the following attributes: Drug name (D1), Drug dose (D2), Drug dose frequency (D3), Duration (D4), and Drug indications (D5). These attributes are all employed by prescriptions in both short-term (acute) and long-term (chronic) disease management [131]. These characteristics of a prescription (which I will also refer to as data requirements) will also be employed in the rest of the dissertation.

As an illustration of how these data items exist in a prescription, let us consider the following scenario presented by Diemert et al. [132].

> *Ms. Smith's physician has prescribed 6mg of Warfarin to be taken orally once daily to address her deep vein thrombosis. Additionally, she was given a prescription for 15 mg of long-acting Morphine orally twice daily and 600 mg of Ibuprofen to be taken three times a day as needed with food to help manage her pain.*

The scenario above specifies three medications: Warfarin, Morphine, and Ibuprofen. The classification above describes the prescribed Ibuprofen, for example, with a name (Ibuprofen - D1), dosage (6mg - D2), frequency (three times a day - D3), duration (no end specified - D4), and indication (with food - D5). In the rest of the chapter, I use the term *medication* more often than the term *drug* because it refers more broadly to an entry in a prescription [132]; it also has a broader usage as it encompasses the prescription of non-drug entities such as food, rest, and physical activity.

4.3.2 Data Exchange Format

When integrating prescriptions into a support tool, it is important to consider avenues through which the prescriptions will be received. There are four options for this. The first option is that the prescription is given to the user by the prescribing physician. The user then has the duty of supplying the required details (D1 to D5) into the support tool via the support tool's Graphical User Interface (GUI). The second option is through a dispensing system (such as Pharmacist's Information System). The third option is through a Personal Health Record (PHR) or provincial Electronic Health Record (system). The fourth option is one where the prescribing physician enters the prescription in a server-hosted CPOE system. The tool should then be able to retrieve the user's prescription from the CPOE system using the user's credentials. In this chapter, and the rest of the dissertation, the fourth option is considered. With the growing adoption of connected health [133], it is arguably more convenient for the tool, after the user has authenticated, to retrieve the user's active prescriptions from some CPOE system. Henceforth, for the proposed formalism, we assume that there is such a source from which the the system would fetch the user's prescriptions and that the source transmits data in a format that is standardized.

With the selection of the data source also comes the question of the data format into which the prescription should be received by the tool. For that, we look to the developers of international standards for the exchange of health data. One such organization is Health Level 7 (HL7) [134]. HL7 is an organization dedicated to providing frameworks and related standards for the exchange, integration, sharing, and retrieval of electronic health information [134]. They are the developers of standards such as the Clinical Document Architecture (CDA) [135] and Fast Healthcare Interoperability Resources (FHIR) [136], two standards that are popular in the exchange of clinical data. CDA is a standard that specifies the structure and semantics of clinical documents, such as discharge summaries, progress notes, and diagnostic reports [135]. It does not address the sharing of prescriptions [135]. FHIR, on the other hand, focuses on providing a flexible, web-based approach to exchanging health data [136]. It has support for medication prescription using modern technologies like RESTful Application Programming Interfaces (API), Unified Modeling Language (UML), Extensible Markup Language (XML), and JavaScript Object Notation (JSON) [137]. The FHIR standard specifies multiple attributes that are supposed to be shared as part of the prescription. Two provisions are of particular interest. They are patient (specifying the identity of the patient, which is important for the retrieval of the correct prescriptions) and dosingInstruction (indicating which the medication should be taken). The attribute dosingInstruction itself is an array of entries where each entry consists of multiple attributes. These attributes include medication (specifying D1), scheduledPeriod (specifying D4), scheduledTiming (specifying D3), doseQuantity (specifying D2), and additionalInstructions (specifying D5). The FHIR standard therefore supports our data requirements (see JSON extract in table 4.1 and full template in Appendix B) and will be considered as the format into which the prescriptions will be received by the patient support tool.

4.3.3 Modelling the Prescription

In order for a prescription to be integrated into a support tool in a manner that supports temporal reasoning, it has to be transformed into a data structure that can easily be processed. Such a data structure should be expressive enough to support multiple prescriptions together with their inherent characteristics (D1 to D5). The FHIR standard has a specification that supports health data exchange in JSON format. The standard supports the data requirements that we wish to consider (D1 to D5). For purposes of prototyping, I employ a simple Restricted Natural Language Grammar (RNLG) that looks like English but avoids the ambiguities of natural language. The grammar restricts the syntax to a small set of interpretation rules [138],

```
{
  ...
  "patient":{ Reference(Patient) },
  "medication":{reference(Medication)},
  "dosageInstruction" :[{
    "text" : "<string>",
    "additionalInstructions":{CodeableConcept},
    "scheduledDateTime" : "<dateTime>",
    "scheduledPeriod" : { Period },
    "scheduledTiming" : { Timing },
    "asNeededBoolean" : <boolean>,
    "asNeededCodeableConcept":{CodeableConcept},
    "site" : { CodeableConcept },
    "route" : { CodeableConcept },
    "method" : { CodeableConcept },
    "doseRange" : { Range },
    "doseQuantity" : { Quantity },
    "rate" : { Ratio },
    "maxDosePerPeriod" : { Ratio }
  }],
  ...
}
```

Table 4.1: An extract from the FHIR JSON specification showing patient identifier, medication name, and dosing instructions.

but provides for additional features, such as restrictions between medications (addressed later in the chapter), which are not yet supported by FHIR. The idea of using RNLG to formalize the creation of plans has been used before to formalize temporal repetitions. For example, Asbru [139] is a language that supports the expression of durative actions and plans caused by durative states of an observed agent. This allows for the representation of temporal repetitions, which is similar to prescriptions.

With this approach, using a restricted set of words, prescriptions of varying complexities and supporting multiple drug regimens could easily be modelled. The RNLG could however be replaced with the FHIR JSON object representing individual prescriptions without functional lapses. In the proposed approach, the root of the grammar is a prescription (See appendix A for full grammar). Every prescription can have one or more **requirements** and/or zero or more **restrictions** (See figure 4.1).

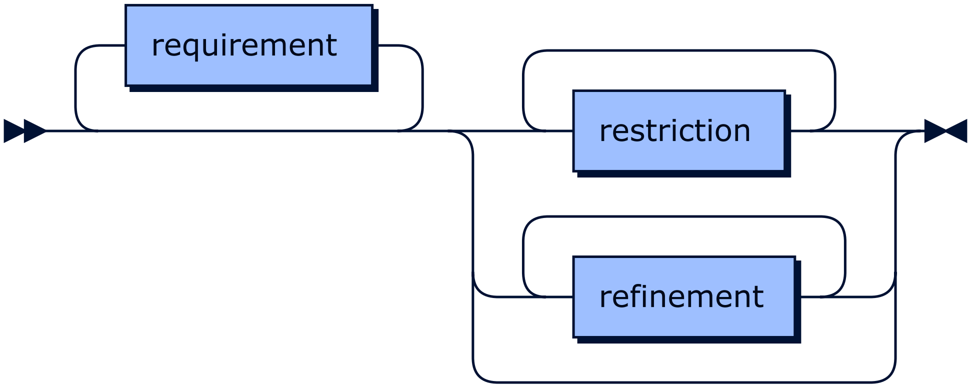


Figure 4.1: A Prescription is made up of one or more requirements, zero or more restrictions, and zero or more refinements.

Requirements indicate the drug regimen to be administered. Restrictions provide a way of preventing unsafe drug interactions in the prescription. The grammar also supports refinements, statements that allow the user to refine the schedule to be derived from the prescription. The underlying implementation of the grammar uses the library Tatsu [140] implemented in Python.

4.3.4 Requirements

In the grammar (Appendix A), a requirement specifies the drug regimen to be administered. It has three main facets: action, duration, and administer.

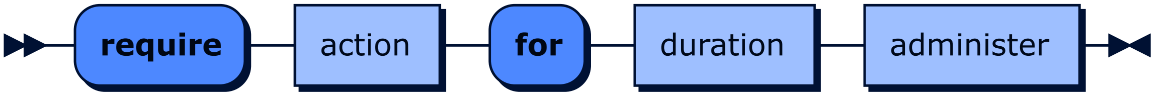


Figure 4.2: A Requirement specifies the action, duration, and administer.

Action

The *action* part of a requirement indicates the drug regimen to be administered (that is, the name of the medication). For example, a prescription may indicate a requirement for *Paracetamol* or *Tylenol*.

Duration

The *duration* part of a requirement specifies the entire period of the prescription implementation. The duration can cover days, weeks, months, or years. The smallest unit for duration is a day and, whatever unit is indicated in the prescription, it is ultimately represented as days (7 days for a week, 30 days for a month, and 365 days for a year). When specifying duration, active days (i.e. days when the drug should be administered) can be specified using either **Iteration** or **Indexing** language constructs.

Iteration is used to indicate a selection of the n^{th} day of the duration of the prescription. For example, for a prescription covering 10 days, an n^{th} iteration of 2 indicates that the drug is to be administered every 2^{nd} day of the period. In this case, the implied days are the first day, third day, fifth day, etc. If no iteration is indicated, then by default the drug is to be administered every day.

Indexing is used to indicate days of the week. A prescription may indicate that a given drug regimen be administered on Monday, Tuesday, Wednesday etc. For a prescription covering months or years, weekly cycles are initiated from the first day. For a period covering days, Sunday is always assumed as the start of the administration period.

Administer

The *administer* part of a requirement is a frame that covers the administration pattern for each medication. Administer indicates the times of the day when the medication is to be administered and the medication unit (or dosage) to be used. The grammar supports three ways of specifying the administration time: **Day-Period**, **Time-Period**, and **Any-Time**.

Day-Period splits the day into three parts: Morning (00:00 to 11:00), Afternoon (12:00 to 18:00), and Evening (19:00 to 23:00). The upper interval specification covers the entire hour (i.e., 00:00 to 11:00 implies 24 AM to 12 PM (exclusive)). The following is an example of a prescription with a Day-Period of Morning:

> *Require Paracetamol for 2 weeks every day, administer 2 in the **morning**...*

Time-Period splits the day into 24 1-hour long intervals and the prescription has to specify the exact time of the day in hours that the drug is to be taken. A prescription with a Time-Period of 6 AM will read as follows:

> *Require Paracetamol for 2 weeks every day, administer 2 at **6AM**....*

Any-Time considers any period of the day as potential drug administration time. Hence, using Any-Time, there is no indication of what time of the day the drug should be administered as the entire 24 hours is considered valid time. The absence of day sectioning (Day-Period or Time-Period) indicates that Any-Time is to be used. A prescription without time specification will read as follows:

> *Require Paracetamol for 2 weeks every day, administer 2.*

The specification of a requirement with the features explained above provides enough expressiveness to author prescriptions with varying levels of complexity. Prescriptions that use Day-Period and Any-Time for specifying administration times are more common than Time-Period specification.

For conditions with multiple drug regimens, the grammar supports inclusion of more than one action in the requirement. For example, a prescription for a patient who is supposed to take Paracetamol and Coartem (a prescription medicine used to treat the symptoms of Malaria) is written with comma-separated requirements as follows:

> *Require Paracetamol for 2 weeks every day administer 2, require Coartem for 3 days every day, administer 4 in the morning and evening;*

4.3.5 Restrictions

Prescriptions come with specified restrictions that are associated with drug administration times. The purpose of the restrictions is to avoid unsafe drug interactions. The restrictions may be within the same drug or between two or more drugs. An example of such a restriction may be that a given time interval be observed between two consecutive medication administrations (that is, medication intake). Such an interval will specify the minimum or maximum time that should be honored by the interval. The FHIR standard supports quasi-restrictions within a drug regiment. The attribute maxDosePerPeriod (see table 4.1) can be used to indicate the upper limit on medication per unit time. An extension would thus be needed to support restrictions between entries from different prescriptions. The grammar supports explicit specification of both types of constraints.

As an example, consider a prescription the requires the user to take 500mg of Paracetamol every morning, afternoon, and evening for 3 days. A patient who implements the prescription by taking the first dosage at 11AM in the morning, the second dosage at 3PM in the afternoon, and the last dosage at 6PM in the evening will have honored the prescription. However, this valid implementation only uses 7 hours of a 24-hour day, which might not be an optimal drug administration cycle that the physician would recommend. Restrictions are introduced to avoid such uneven distribution of administration times, if required.

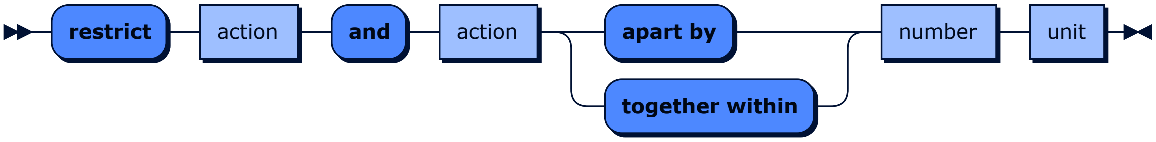


Figure 4.3: A Restriction has three components; action, guide, and temporal constraint.

To ensure that drug regimens are spread evenly in any given period, the physician may decide to dictate the time that should pass between any consecutive drug administration times. For example, the physician may indicate that a 6-hour break be observed between any two intakes. The proposed formalism supports this through the **restrict** statement (see figure 4.3), that provides a way of restricting certain regimens from being taken together (or apart) in a specified time frame. A **restrict** statement has three components:

(i) The **actions** to be restricted,

(ii) A restriction **guide** indicating whether the actions should be taken together or apart, and

(iii) A **break period** that is minimum and/or maximum time period that should be observed between the actions.

An an example, let us consider the following restriction (which would exists as part of a prescription statement):

> *Restrict Paracetamol and Paracetamol* ***apart by*** *6 hours.*

From the statement above, "Paracetamol" is the action, "apart" is the restriction guide, and "6 hours" is the break period. This restriction indicates that any consecutive drug administration for action Paracetamol should have a minimum of 6 hours between them (that is, any two Paracetamol regimens should be taken 6 hours apart). In this example, the action is repeated when referring to the same regimen (Paracetamol) because this restriction is about the same drug regimen. From the earlier prescription, the time periods (Morning, Afternoon, and Evening) will supply the upper bounds on the time intervals. With the example of a patient who administers the regimen at 11AM, 3PM, and 6PM, the restriction is violated because there is a 4 hour gap between 11AM and 3PM and a 3 hour gap between 3PM and 6PM). On the other hand, a cycle with 6AM, 1PM, and 8PM administration times does not violate the restriction.

Prescriptions often have more than one drug regimen specified for the patient. These drug regimens can be taken together or spaced according to how the physician deems most effective. For example, the following prescription indicates that a patient must take Paracetamol and Coartem together within a time frame of 4 hours:

> *Require Paracetamol for 3 days, every day administer 2; require Coartem for 3 days, every day administer 1;* ***restrict*** *Paracetamol and Coartem* ***together within*** *4 hours*

Such a rule, in the model, will always flag a constraint violation if Paracetamol is taken and no Coartem is administered within the 4 hour window. Replacing"together within" with "apart by" in the prescription above would reverse the restriction leading to a constraint violation if the two medications are administered together within a 4 hour window.

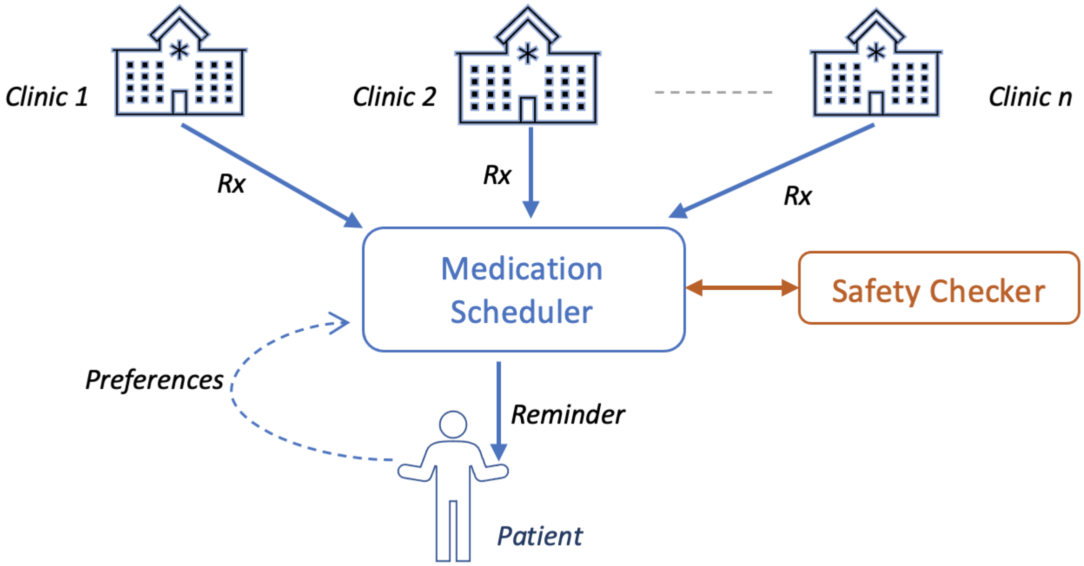


Figure 4.4: Overview of the proposed connected health system. All prescriptions given to the patient are maintained in a *medication scheduler* service which issues adherence reminders. The *safety checker* ensures that harmful drug-drug interactions are avoided.

4.3.6 Refinement

A refinement is a modification made by the user to a prescription schedule after it has been generated. During implementation, a refinement can be performed by manipulating medication schedule entries through a interface provided for the user. We added it to the grammar for testing purposes. Syntactically, a refinement statement specifies three things; (i)the action, (ii) the duration of the refinement, and (iii) type of refinement to be made. There are two options for the type of refinement to be made. The first option is by specifying a list of time points that should be used for the entire schedule. An example is changing the time points for a medication scheduled at 9 AM to 8 AM for a specified number of days. The section option is to modify the time point of a single entry in the schedule. For this, both the old timestamp and the new time stand should be specified.

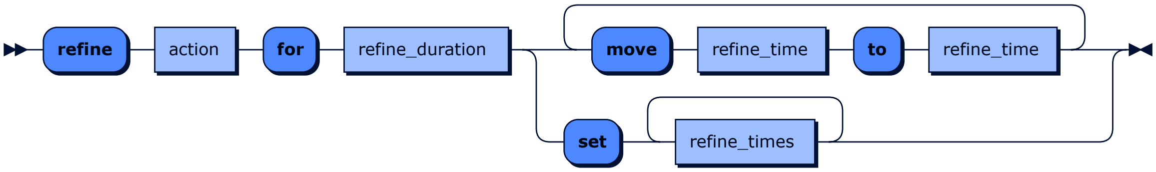


Figure 4.5: A Refinement specifies an action, a duration, and new timestamps for entries in a medication schedule

4.4 Prescription Transformation

Once a prescription has been represented using the RNLG detailed in the previous section, the prescription is transformed into the formal model that supports temporal reasoning. The problem of interest in my work is to verify that a refinement of the prescriptions schedule is consistent. The usage scenario is explained below.

4.4.1 Medication Scheduler

Consider a case in which a Physician writes a prescription a for patient. Let us call the Patient, Jane Doe. The prescription indicates a set of medications (requirements) that Jane is supposed to take. The prescription also includes a set of constraints (restrictions) that are supposed to be observed by Jane as she administers the medication. Jane downloads the prescription into a medication scheduler. The scheduler generates a schedule that spans the duration of each medication in the prescription. She now has a schedule. As part of tasks that can be performed in the scheduler, Jane can refine some medications to match her preferences. This process is obvious in many cases where the prescription does not indicate the exact time when a given medication should be administered. For example, if the prescription is for a medication that should be taken once a day for a year, it is up to Jane to decide the time of the day that the medication should be administered. The scheduler tool should then provide for mechanisms to ensure that the changes made by Jane do not violate the restrictions specified within the prescription. In this case, the check is performed by a Safety Checker. See figure 4.4. A medication schedule should therefore support a temporal knowledge base and algorithms to check the consistency of the schedule refinement.

Dechter et al. have formalized these types of problems as Temporal Constraint Satisfaction Problems (TCSP), where variables represent time points and temporal information is represented as a set of unary and binary constraints, each specifying a set of permitted constraints [28]. I use a particular specialization of the TCSP, called a Simple Temporal Problem (STP), in which all constraints are specified as intervals. The STP framework is explained below.

4.4.2 Simple Temporal Problem

The STP is a network in which time points are represented as nodes of a graph and the intervals between the time points (or events) as weighted edges [28]. Each edge is labelled by an interval [a_{ij}, b_{ij}] that represents the constraint between any two given nodes a and b. The interval represents constraints of the form $a_{ij} \leq X_j - X_i \leq b_{ij}$ where X_j and X_i are two events to be constrained. An STP is associated with a directed edge-weighted graph G_d = (V, E_d), called a distance graph. Dechter et al. also propose a way of determining whether or not the distance graph generated from STP is consistent by applying Floyd Warshall's all-pairs-shortest-path algorithm. A given STP is deemed consistent if and only if its distance graph G_d has no negative cycles.

In the formalization of prescriptions, an STP fits well since we are dealing with time points (administration times) and intervals (distance between administration times). Regarding restrictions, all constraints are binary (Intervals are bidirectional) and there is no dis-junction involved (I do not address cases where the user is given an option to choose between two medications. That is, requirements of the form "Take A or Take B". Each medication has an exclusive administration pattern). This also fits into the STP framework. Hence, to transform a prescription into an STP graph; nodes denote the endpoints of calendrical intervals such as days, sections, and times of the day when drug administration will be actualized; and edges connect the nodes and represent the time difference between any two connected nodes. Prescription transformation handles the requirements and the restrictions of the prescription separately. I begin with the requirements transformation process and then follow it up with the restriction application process.

4.4.3 Requirements Transformation

A Requirement has two parts to be transformed: the Duration and the Administer parts. *Duration*, as detailed earlier, specify two time portions: the entire length of the prescription and the selection of days when the drug should be administered. *Administer* indicates the periods or time points within the days when the drug is to be administered. This can be a section of the day (such as the morning) or the indicative time of the day (such as 6 AM). The process of transforming requirements into a distance graph is a three-step process that includes *Expansion*, *Selection*, and *Reduction*. I illustrate this concept using the following example prescription with a duration that covers a period of 2 weeks, a drug that is to be administered every second day, and the administration period of morning:

> *Require Paracetamol for 2 weeks every 2nd day, administer 1000mg in the morning;*

Expansion

The first stage of the prescription transformation process is the expansion of the prescription duration. This entails transforming the entire duration into days. From the example above, 2 weeks is unrolled into 14 days. The 14 days result in 14 nodes called Day Start (DS) nodes and labelled DS1 through DS14. These DS nodes are then connected to each other from the nearest to the furthest, with each node connected to adjacent nodes. The edges connecting the nodes are bidirectional, with weights of 24 and -24 (there are 24 hours in a day) going forward and backward respectively, as per STP rules.

The first node in the graph is a start node labelled START which represents the directive to commence implementation of a prescription. This is the node to which the first DS node in the requirement will connect. Since a prescription can have multiple requirements, each requirement's first DS node will connect to the START node. Figure 4.6 shows the expansion of 2 weeks into 14 days.

Once all the days have been unrolled, each of the DS nodes is then expanded. This expansion is influenced by the indication in the Administer part. For medication that is supposed to be administered once a day without an indication of the time, no expansion is necessary. But when the administration period is indicated, then the nodes must be expanded to reveal the periods. The granularity of the expansion includes Day Start (DS), Section Start (SS), and Hour Start (HS) - which is the minimum. DS is for the day, SS is the period of a day such as Morning, Afternoon,

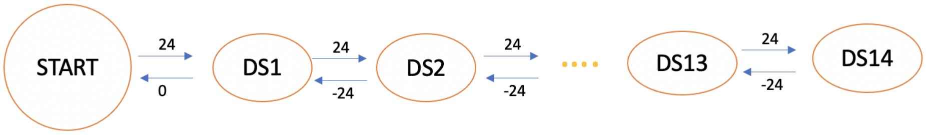


Figure 4.6: The duration of 2 weeks is expanded into 14 days introducing 14 DS nodes. The Start node is the beginning of time and represents the directive to commence implementation of a prescription.

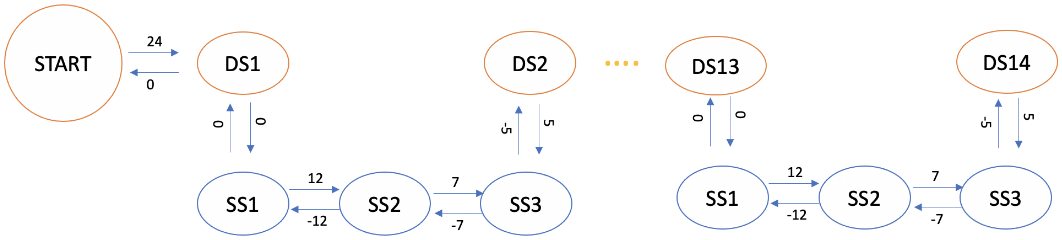


Figure 4.7: Expansion of Day Starts (DS) into Section Start (SS) nodes. The 24-hour direct connection between DS nodes is removed. Three SS nodes are introduced the connect the DS nodes back together. The 24-hour distance between two DS node is maintained.

and Evening (MAE), and HS is for the duration spanning 1 hour. HS is set as the minimum because hour is usually the lowest granularity for medication timing. The attribute scheduledTiming in the FHIR standard does not impose any restriction on the smallest value that can be passed through the attribute. The examples given, however, employ hour as the minimum [137].

In the case of the example above, the expansion creates three additional Section Start nodes. The three periods of the day are numbered SS1, SS2, and SS3 for MAE respectively. A node is thus created for morning, afternoon, and evening for each one of the DS nodes in the graph. During the expansion, the existing direct edges between any two candidate DS nodes are removed. New edges are created that connect DS to SS1, SS1 to SS2, SS2 to SS3, and SS3 to the next DS. Given that the three sections of a day do not have the same duration (using the same duration would not affect the technical implementation of the model), the edges connecting DS to the morning SS (SS1) are weighted 0, those connecting SS1 to afternoon SS (SS2) are weighted 11, those connecting SS2 to evening SS (SS3) are weighted 7, and those connecting SS3 to the subsequent DS node are weighted 5 (12 AM → 11 AM, 12 PM → 6 PM, and 7 PM → 11 PM for MAE respectively). The total weight of the newly introduced edges sums up to 24 for a 24-hour day. Each weight is negated for the reverse edge. Figure 4.7 shows the expansion of DS nodes into SS nodes.

The last step in the expansion phase is the introduction of Hour Start (HS) nodes. This step expands the SS nodes into HS nodes, creating 24 HS nodes between any two DS nodes (spread across SS intervals) and removing the direct link between them. The new HS nodes are labelled HS1 through HS24. The 24 nodes are divided into 11 nodes between SS1 and SS3, 7 nodes between SS2 and SS3, and 5 nodes between SS3 and the subsequent DS node. All HS nodes are connected to each other with edges that are weighted 1 and -1 (for 1 hour) forward and backward respectively. Algorithm 1 shows the determination of the duration and creation of nodes (DS, SS, and HS) representing the duration of the prescription (p).

Selection

The next stage after Expansion is the Selection stage. This stage identifies which nodes in the graph should be connected to Administration Nodes (AD). AD nodes are nodes which indicate the presence of drug administration activities (medication taking event). The selection again consists of at most three phases:

1. In the first phase, all DS nodes with occurrences of AD activities are selected. From the example above, the drug is administered every second day. Hence, every 2^{nd} DS node in the graph is marked as a candidate node for the introduction of AD nodes. If the administration period indicated in the prescription covers the whole day, then the administration nodes are attached to the candidate nodes and the selection process halts. Otherwise, the second phase is executed.

2. The second phase traverses through all selected DS sections and identifies SS sections indicated in the prescription. In our example, the administration period is indicated as the morning, therefore the morning SS (SS1) is selected as a candidate for AD introduction. If the administration period indicated in the prescription covers the section of a day, then the administration nodes are

Algorithm 1 Requirements Transformation Algorithm

```
procedure TRANSFORMREQUIREMENT(requirement)
    duration ← getDuration(requirement)
    admin_level ← getAdminLevel(requirement)
    dayNode ← NULL
    counter ← 1
    while (counter <= duration) do
        dayNode ← createDSNode(dayNode)
        if (admin_level < DS) then
            sectionNodes ← createSSNodes(dayNode)
            if (admin_level < SS) then
                hourNodes ← createHSNodes(sectionNodes)
            end if
        end if
        counter ← counter + 1
    end while
    allNodes ← fetchAllNodes(prescription)
    for (node in allNodes) do
        if (isCandidateNode(requirement, node)) then
            createADNode(node)
        end if
    end for
    allNodes ← fetchAllNodes(prescription)
    for (node in allNodes) do
        if (hasNoADNode(node)) then
            collapseNode(node)
        end if
    end for
end procedure
```

attached to the candidate SS nodes and the selection process halts. Otherwise, the third phase is executed. In the example being used, the selection process would terminate at the second phase, as shown in Figure 4.8.

3. If the drug administration time indicated in the prescription is a time of the day (e.g. 6 AM), this phase traverses the graph marking as candidate nodes all HS nodes that have an administer instruction. The AD nodes are then attached to each candidate node and the process halts.

At the end of the selection stage, Administration Nodes are introduced and attached to each selected candidate node. Thus, new edges are introduced between candidate nodes (that is, DS, SS, or HS) and DA nodes. The weights of the edges correspond to the weight of the candidate node to the next similar node in the graph. For example, edges between DS nodes and AD are weighted at 23 (one less than 24, the weight between any two given DS nodes). The reverse edges from the DA nodes are always weighted 0 (as is the rule in STP when using intervals on edges, the lower value, 0 in our case, becomes the reverse edge.

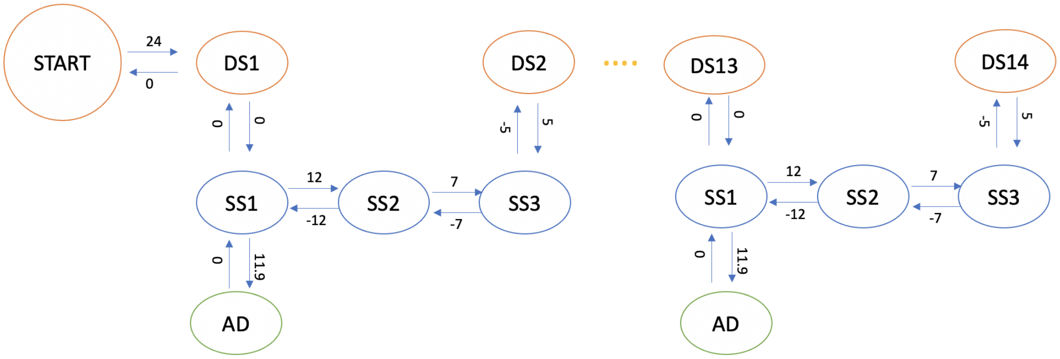


Figure 4.8: All Section Nodes for which medication is supposed to be administered are identified. AD nodes are then connected to each selected SS node with appropriate weights. In this example, the weights are 11.9 and 0 to and fro respectively

Reduction

Reduction is the last stage in the prescription transformation cycle, in which all nodes that are not attached to a DA node are collapsed (reduced). The processing of collapsing the nodes reduces the number of nodes while still maintaining the distance between them, as shown in figure 4.9. Hence, all DS, HS, and SS nodes that do not have DA links are removed and a direction link is formed between two nodes that are attached to AD nodes. This mode significantly reduces the size of the graph and makes it efficient when search algorithms are run against the graph. The reduction stage is merely for performance enhancement and has no effect on the expressiveness of the model.

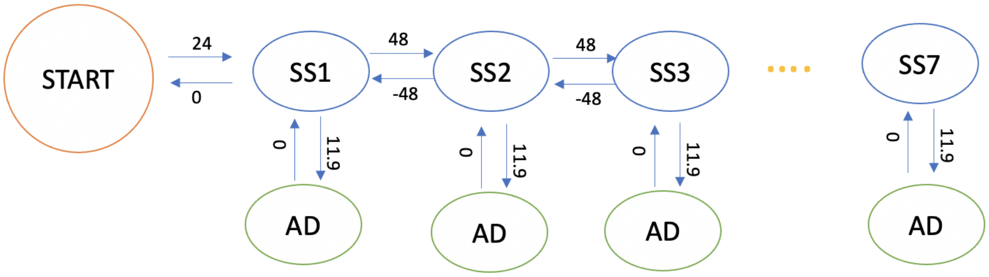


Figure 4.9: The graph in figure 4.8 is minimized removing all nodes that do not have a direct connection to an AD node.

4.5 Restriction Transformation

A successfully transformed requirement will have the semantic model represented as a graph with two types of nodes; Sectioning Nodes (DS, SS, HS) and Administration Nodes (AD). These nodes are connected by edges with weights representing the temporal distance between them. For AD nodes, the temporal distance indicated overlaps the parent sectioning node (even though the pictorial representation appears to indicate a branching off from the path).

The process of transforming the requirements to include restrictions comprises the introduction of restrictions between all affected AD nodes. The restrictions are applied to both nodes of the same requirements and of different requirements. Algorithm 2 is representative of the introduction of restrictions between a pair of requirements. The process of introducing restrictions follows three stages: querying, linking, and weighting.

1. **Querying**: This stage queries the graph for all AD nodes whose requirement is mentioned in the restriction. The details of the nodes are placed in a list sorted by the temporal distance of each node to the start node. This stage produces one list per requirement.

2. **Linking**: This stage fetches the type of restriction indicated in the prescription. As indicated earlier in subsection 4.3.5, the restriction can be one that constrains two medications to be taken together or apart. Regardless of the type, this stage introduces edges that interconnect all selected AD nodes. Each AD node

is connected to every other AD node within the list and to nodes of other list participating in the restriction.

3. **Weighting**: During interconnection, the type of restriction determines the weights of the edges being introduced. For every two nodes, a bidirectional edge is introduced. The weight of the edge is the temporal constraint specified in the restriction. An interval, with lower and upper limit is required for this. The temporal constraint specified in the restriction may indicate both the lower bound and upper bound (as in take B within 4 to 6 hours after A). These are used as weights on the edges. The restriction specification also supports temporal constraints where only the lower bound or upper bound is indicated. Such is the case for "apart" restrictions where only the lower bound may be specified and for "together" restrictions where only the upper bound may be specified. For "apart" restrictions, the unspecified upper limit becomes infinity (∞). The unspecified lower limit for "together" restrictions becomes Zero (0). When specifying the weights of the newly introduced edges between the nodes, the lower value (negated as per STP rules [28]) becomes the weight on the edge connecting one node to the successor node (that is, for any pair of nodes, the source of this edge is the node that has a shorter temporal distance to from the start node). The larger value (taken as is) becomes the weight for the reverse edge. The edge between any two nodes is introduced by the node with a shorter temporal distance from the start node. This is done to avid duplicity of edges between nodes, especially for nodes coming from different requirements.

As an illustration, let us consider the graph in figure 4.9 which is a pictorial representation of the prescription rewritten below. As an extension to the prescription, we have added a restriction so that there is a minimum of 4 hours between any two administration times.

> *Require Paracetamol for 2 weeks every 2nd day, administer 1000mg in the morning; restrict Paracetamol and Paracetamol apart by 4 hours;*

The introduction of the restriction here implies that there should be a 4-hour minimum interval between any two paracetamol AD nodes. Since the period only indicates a minimum, the maximum is becomes infinity. For computational purposes, the value 876000, representing the number of hours in 100 years, is adopted in lieu of the maximum. Figure 4.10 shows the interconnection of the AD nodes.

Algorithm 2 Restriction Transformation Algorithm

```
procedure TRANSFORMRESTRICTION($restriction$)
    $reqs \leftarrow getActions(restriction)$
    $adNodes = fetchADNodes(reqs)$
    $[min, max] \leftarrow getConstraints(restriction)$
    for ($i\_node$ in $adNodes$) do
        for ($o\_node$ in $adNodes$) do
            if (($i\_node$ NOT $o\_node$) AND ($i\_node.time > o\_node.time$)) then
                $to \leftarrow linkNodes(i\_node, o\_node)$
                $from \leftarrow linkNodes(o\_node, i\_node)$
                $addEdgeWeight(to, max)$
                $addEdgeWeight(from, min)$
            end if
        end for
    end for
end procedure
```

Following the rules of graph transformation of STP, the shorter distance in the interval is always negated and becomes the weight of the incoming edge. Hence, every connection with an interval of [-4, 876000] in figure 4.10 represents two edges between the nodes, the smaller one coming in, the larger one going out. If the restriction specified in the prescription is between two requirements, AD nodes of the same requirement are not connected.

The prescription has now been transformed into a Simple Temporal Problem. All the information about the prescription can now be queried. With the graph in its current form, it is possible to generate an administration schedule with all temporal constraints available in the schedule. In the next section, I present the process of determining whether a refinement of the prescription is consistent.

4.6 Consistency Checking

I define a Consistency Check as a mechanism for determining whether there is a conflict (or violation of the restriction) in the refinement of a schedule. Such a conflict can arise when the administration times indicated in the refinement derived from the prescription schedule violates restrictions attached to that prescription. Let us consider the scenarios below. A simple scenario and one that has some level of

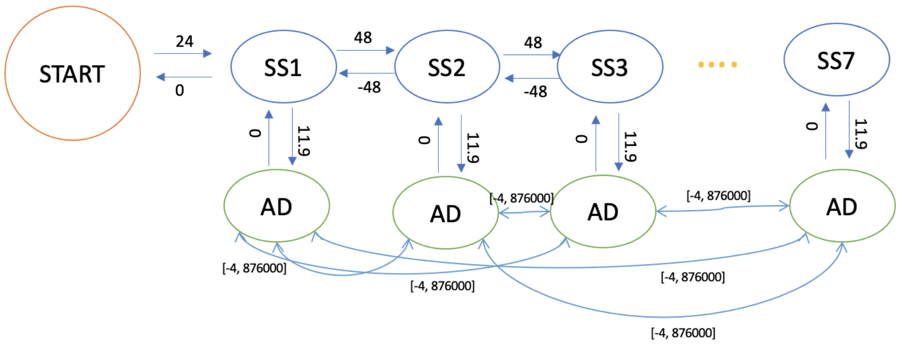


Figure 4.10: The figure shows the interconnection of AD nodes for the restriction that specifies a minimum of 4 hours between any two administration nodes of the same prescription. The lower edge weight of 4 is negated and the number 876000 is used in place of infinity.

complexity.

4.6.1 A simple problem

Jane Doe receives a prescription from her clinician. The prescription requires that Jane takes paracetamol for three days, three times a day. Represented in our format, the prescription would have the following form.

> *Require Paracetamol for 3 days, every day administer 1000mg in the morning, afternoon, and evening; restrict Paracetamol and Paracetamol apart by 6 hours;*

The prescription above specifies that Paracetamol should be taken three times a day; Morning, Afternoon, and Evening. There is also a restriction which specifies that between every two intakes, there should be a 6-hour minimum interval. Since there are 24 hours in a day, Jane has enough time to spread the administration times so that the constraints are not violated. If Jane took the first dosage at 10 AM. And then she has a very bad headache at 2 PM. So, she refines the schedule, moving the medication which she had originally put at 4 PM to 2 PM. Jane has now violated the constraints of the prescription. One would argue that Jane should be able to remember these constraints. For a single medication, maybe she can. But the process becomes even more difficult when the number of medications increases.

4.6.2 A bigger problem

Let us consider an example prescription that has multiple medications.

> ***require*** *Paracetamol for 5 days every day administer 1000mg in the morning, afternoon, and evening;* ***require*** *Metronidazole for 5 days every day administer 2 caps in the morning, afternoon, and evening;* ***require*** *Coartem for 3 days every day administer 4 tablets in the morning and evening;* ***restrict*** *Metronidazole and Coartem apart by 4 hours;* ***restrict*** *Metronidazole and Metronidazole apart by 8 hours;* ***restrict*** *Coartem and Coartem apart by 12 hours;*

The prescription has three requirements and three restrictions. Two of the restrictions are within the same medication. One is between two medications (Metronidazole and Coartem). Since the prescription does not indicate the time that the medication should be taken, it is up to the user to solve the puzzle and refine the schedule in such a manner that all constraints or restrictions specified in the prescription are respected. Scheduling Coartem at 4 AM and Metronidazole at 8 AM AM raises no flags. After 8 AM, the next available slots for Metronidazole are 4 PM and 12 AM. Since Coartem requires a 12-hour space, The only available space is 8 PM. Any other times violate the constraints. This is a problem that requires brevity in solving. A trial-and-error refinement process where either the system suggests available slots or confirms the safety of the user-generated schedule is the rationale behind the consistency check presented here. Below, I present how the STP consistency check can be applied to problems of this nature to help flag errors resulting from unsafe scheduling.

4.6.3 Identifying Conflicts

To identify conflicts in the prescription, I employed the process of detecting inconsistencies using Simple Temporal Problems [28]. Since, as earlier explained, the entire prescription is represented as a distance graph with nodes as administration times and edges as intervals between the administration times, finding inconsistencies entails detecting whether or not the representative distance graph contains negative cycles [141]. This is achieved by applying Floyd-Warshall's all-shortest-paths algorithm to the distance graph and computing the values of all cycles in the graph. If all cycles are positive, then the STP is consistent. Consider the following distance graphs based on the prescription in section 4.6.1 for paracetamol taken three times a day with a 5-hour interval.

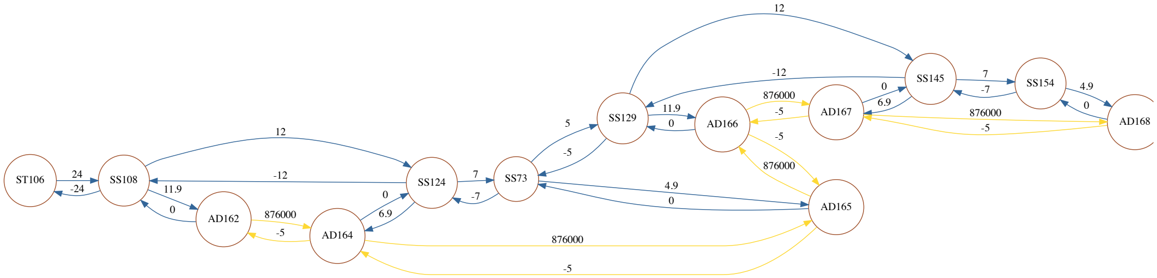


Figure 4.11: A distance graph representing the prescription "require Paracetamol for 3 days, every day administer 1000mg in the morning, afternoon, and evening; restrict Paracetamol and Paracetamol apart by 6 hours"

.

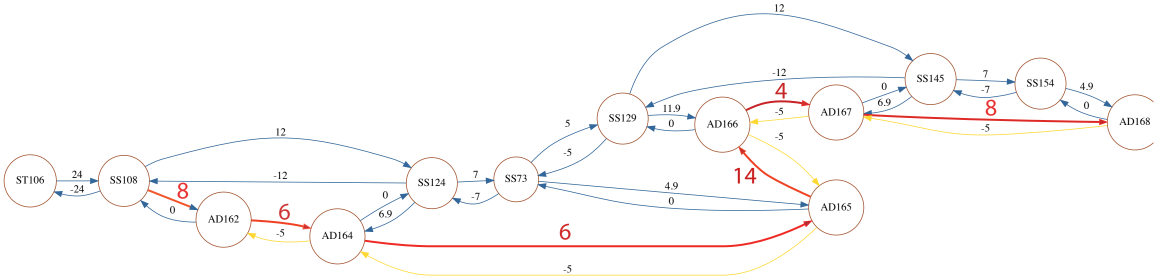


Figure 4.12: A distance graph showing the possible implementation of the prescription in figure 4.11. The 4 hour separation between AD166 and AD167 introduces a negative cycle that makes the schedule inconsistent.

To determine whether or not the graph depicted in figure 4.11 is consistent, we begin by creating a minimal table for the distance graph. The minimal table will indicate the shortest distance between every pair of nodes in the graph. Table 4.2 shows the distances. Since we are only interested in administration nodes, other nodes are ignored in the table. The creation of the minimal table is only for performance enhancement when repeated checks are being performed. Algorithm 3 shows the consistency check when bypassing the minimal table.

From table 4.2, it is easier to verify that there are no negative cycles in the graph. Hence the schedule represented in figure 4.11 is consistent. It is important to know

Algorithm 3 Consistency Checking

```
procedure CheckConsistency(prescription)
    $req \leftarrow getRequirements(prescription)$
    $res \leftarrow getRestrictions(prescription)$
    $adNodes = fetchADNodes(req)$
    for ($i\_node$ in $adNodes$) do
        for ($o\_node$ in $adNodes$) do
            if (($i\_node$ NOT $o\_node$) then
                $to \leftarrow allShortestPaths(i\_node, o\_node)$
                $fro \leftarrow allShortestPaths(o\_node, i\_node)$
                if ($to + fro < 0$) then
                    return $false$
                end if
            end if
        end for
    end for
    return $true$
end procedure
```

Table 4.2: Table of short distances between AD nodes

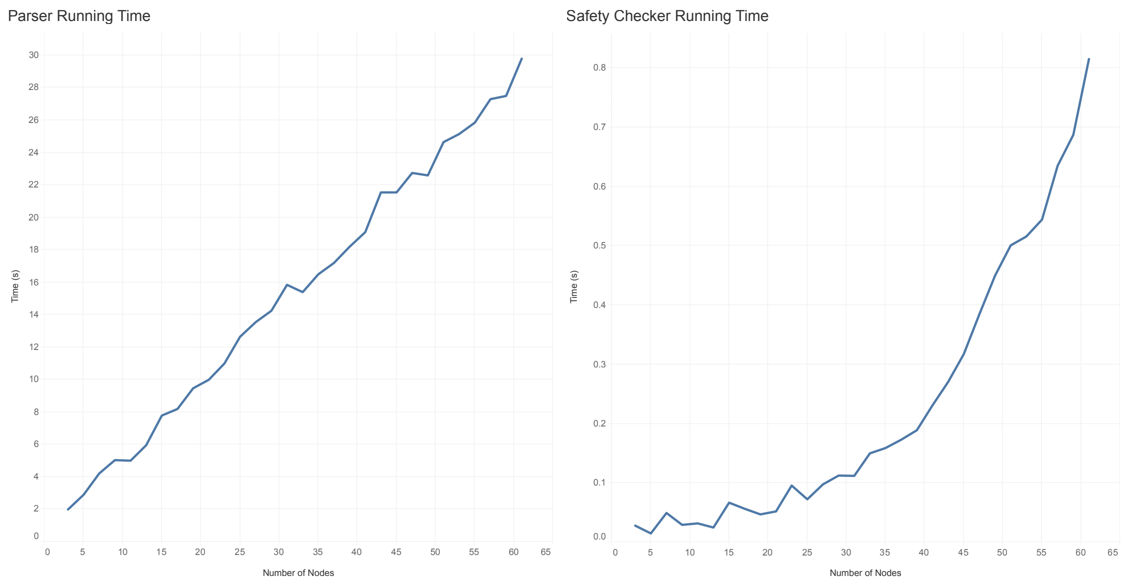
Node	AD162	AD164	AD166	AD165	AD167	AD168
AD162	0	18.9	35.9	23.9	42.9	47.9
AD164	-5	0	23.9	11.9	30.9	35.9
AD166	-12.1	-5.1	0	-5	18.9	23.9
AD165	-10	-5	16.9	0	23.9	28.9
AD167	-20	-15	-5	-10	0	11.9
AD168	-25	-20	-10	-15	-5	0

that the schedule given in this example is the default schedule derived from the original prescription that would be authored by the physician (before refinement). It is not expected to be inconsistent. Let us assume that the person who is implementing this prescription refines the schedule by following the trail given in figure 4.12. In the trail, the second dose (AD167) on day 2 has been moved to within 4 hours of the first dose (AD166). Since -5 reflects a lower bound and was specified as part of the restriction, it remains as the lower bound on the link. The 876000 is however replaced by 4 (the new upper bound). This move introduces a negative cycle between AD166 and AD165 (-5 + 4 = -1) and make the refinement inconsistent.

In the proposed model, the consistent check is supposed to be performed every time there is a refinement to the schedule. No check is performed after the generation of the default schedule. The assumption is that the prescription given to the patient is always consistent. The evaluation of the model is presented below. Details of how the framework would work as part of a support tool are presented in Chapter 6.

4.7 Model Evaluation

The proposed module was evaluated as a stand-alone system, using the Python Programming Language, in Jupyter Notebook [142]. Several libraries were used in the implementation. Tatsu was used for the grammar specification [140], py2neo was used for connecting to the Neo4J database [143], and Graphviz was used for visualizing the model [144]. The model was created with five main modules as shown in figure 4.14.


Parser Running Time

Safety Checker Running Time

Figure 4.13: Running time of two modules with an increase in the number of nodes. Left: The running time of the Parser rises linearly with the number of nodes. Right: The running time of the consistency (safety) checker over the same number of nodes. Running time is smaller but rises sharply as the number of nodes increases.

1. The first module specified the grammar.
2. The second module was the parser that would receive the prescription statement, parse it and generate the representative graph in the database. When

using the FHIR framework, this is the module the would get the data from the JSON object, parse it, and generate the graph.

3. The third module was the consistency checker that fetched the graph from the database and run on it to determine if the prescription represented was consistent or not. The output of this module was a statement indicating whether the prescription was consistent.

4. The four module was the refiner that was used to refine the prescription. Refinement meant indicating the actual times when a given medication was supposed to be taken. This component manipulated the graph by changing the temporal distance between administration nodes (replacing the weights on the outgoing edges).

5. The fifth module was the graph generator that was used to show a pictorial view of the prescription and also generate the schedule based on the graph.

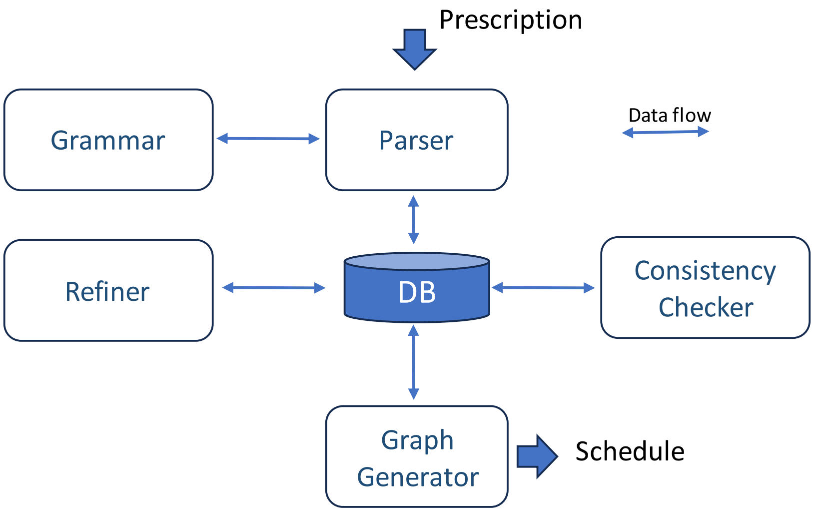


Figure 4.14: Module interaction of the semantic model. The prescription is processed by the Parse to creates a graph based on the prescription and stores it in the database. Graph generator shows the visualization of the graph and generates the schedule.

4.7.1 Performance

The model was tested for performance. The testing targeted two modules: the parser module and the consistency checker module. To test both modules, I used a prescription that requires a patient to take a medication once a day for a specified period. The medication had a constraint that imposed a minimum interval of 20 hours. The period of administration varied from 1 day to 30 days. For each day, the performance of both modules was recorded twice. The average execution duration is shown in figure 4.13. As can be concluded from the graph, the parser's time complexity was linear ($\mathcal{O}(n)$). The running time was measured against the number of nodes generated from the prescription. Reporting the running time against the number of nodes implies that the experiment would yield the same results even if the number of medications taken per day, in the example prescription, was increased.

For example, according to figure 4.13, generating a schedule from a prescription that has 30 nodes would take about 15 seconds. With the running time being linear, a prescription that requires the user to administer medication twice a day for 30 days would yield 60 nodes with about 28 seconds of processing time. The response time is quite high. However, the generation of a schedule is a one-time operation, executed every time a new schedule is being generated. Repeat operations are performed when doing consistency checks. This is done after refinements. Checking the consistency of such a prescription, according to figure 4.13, would only take about 0.7 seconds. The cost could further be minimized by localizing checks. If the last check that was performed was consistent, the check after a single refinement can be localized to that day or week. An overhead cost of less than 1 second is reasonable enough for the semantic model to be used in support tools for prescription management. With localization, the cost is almost negligible. The time complexity of the consistency checker was polynomial ($\mathcal{O}(n^3)$). This was an expected performance. Floyd-Warshall's all-pairs-shortest-path algorithm that was used to generate the distance graph promises the same time complexity [28].

4.8 Discussion and Recommendations

The formal model proposed in this chapter provides a simplified way of authoring and reasoning about prescriptions. The model can work with both single and multiple prescriptions and supports reconciliations between prescriptions in the form of restrictions. The model comprises two primary modules. The first module is responsible for transforming prescriptions into a formal semantic model that is based on the Simple Temporal Network. While in my work the input for this module is a statement based on the restricted grammar specified, the technical implementation of this module will not have significant changes if the input comes from a source such as Computerized Provider Order Entry systems. The second module runs to check for consistency between prescriptions. This consistency check takes as input the temporal constraints present in the prescription. The module creates links between all nodes that are marked as administration nodes and check to ensure that no negative cycles are present in the resultant graph.

The model was tested with prescriptions of different duration and number of medications. The tests conducted were to validate the performance. The term validate is used because the model has as its benchmark the performance of the all-shortest-path algorithm which runs in polynomial time. The model supports prescriptions for medication that have repeat cycles based on calendar days (nth day cycles) or days of the week (specified as Sunday, Monday etc.). The model supports prescriptions of any duration (limited only by the size of the host system). The proposed model is not intended to work as a stand-alone system. For the model to be useful, the modules must be bundled together into a support tool that can run on ubiquitous devices. Another application area would be to integrate this type of functionality into Medication Reconciliation functions in Shared Health Record (SHR) systems, Pharmacy Information Systems (PIS) or even Personal Health Record (PHR) systems where patients maintain their up-to-date prescriptions coming from different provider systems. This integration is demonstrated in the design of calendars for medication management as presented in Chapter 6. The chapter will also discuss performance issues that relate to this model when incorporated into a system.

There are many improvements that can be made to the proposed model for it to work seamlessly. In its current form, the model does not support prescriptions with changing dosages based on time (i.e. Titration). The model does not support varying intervals such as a medication whose administration cycle has no repeat pattern (e.g. 1^{st} day, 10^{th} day, 15^{th} day, n^{th} day - where n is arbitrary). Prescriptions with alternating medications (where you can take medication A or B) are also not supported. These features should be considered for future work as they form part of clinical guidelines concerning prescriptions. The model is also not extensive in the implementation of all features supported by the FHIR standard. FHIR attributes such as rate (medication administration without predetermined repetition cycles) and asNeeded (for medication with optional administration), which may affect the interpretation of requirements and restrictions, are not currently addressed by the model.

4.9 Conclusion

In this chapter, I proposed a formal model for reasoning with Medication Prescriptions. The model comprises; (i) a transformation system that reads prescription data and formulates a semantic model representing the prescriptions, and (ii) a validation mechanism that determines whether the authored prescription is inconsistent (that is, contains conflicting prescriptions). The derived semantic model was represented as a Simple Temporal Network. I used an implementation of Floyd-Warshall's Algorithm to find the shortest temporal distance between medication administration times and determine the consistency of prescription schedule refinements. The proposed model provides support for authoring prescriptions and checking for inconsistencies in single and multiple prescriptions. The model is designed in such a manner that it can be incorporated into any patient support tool that works with prescription data. By supplying a pre-formatted prescription as an input, the model generates a schedule based on the prescription. Authoring and refinements of schedules can be employed in reminder systems and medication management tools.

Chapter 5

Exploring Design Variations

This chapter is based on a previous publication [145]. Hence, any use of the pronoun "we" in this chapter refers to Maybins Lengwe, Jens Weber, and Charles Perin.

5.1 Introduction

Reminder Applications are among the most common form of patient adherence support tools [146, 147, 148]. They are used to trigger notifications to remind the patient on when they are supposed to take their medications [75, 76, 16, 149, 150]. In most of the reminder applications, the patient is required to input the schedules (based on their prescriptions) for when the alarm should be triggered. For patients with simple prescriptions and static schedules, these applications often suffice. But for patients with frequent and dynamically changing schedules, this form of "context unaware" static reminders often fall short of addressing their adherence problem. Scheduling of prescriptions for such type of patients is one that involves laborious interplay between daily activity management and medication management.

To appreciate this problem, consider a patient who manages a busy schedule of dynamically changing day-to-day commitments using an application (typically a calendar), while using a separate medication reminder app. A good number of questions is here posed. What happens when a medication reminder is triggered during the time when the patient is busy? How does that patient reschedule the reminder taking into considerations other activities already planned for? How does one resolve clashes between other activities and medication administration without violating the constraints specified in the prescription? To address these problems, an integrated approach where calendar activities and medication management are bundled into one support tool is necessary. Such a redesign should be centered around electronic calendar.

Electronic calendars have become instrumental in the management of daily activities [96, 151, 152]. They are used to coordinate interactions among individual schedules of family or team members and can convey meaning and values behind the priorities of scheduling [153]. Calendars have been used to visualize temporal trends that include everyday activities such as energy use in work places, fitness tracking, and work routines [154, 97, 155]. In short, they are versatile. The question is, can their versatility be extended to support medication management? As a medication management tool, we posit that calendars would be very effective.

In this chapter, we explore the possibility of integrating prescription management into electronic calendars. Calendars are already used by many patients for other health related purposes [156, 157]. The integration under consideration comes with a couple of challenges. The first challenge is how to render the prescription entries so that the user can differentiate between a normal calendar entry (that is, an entry representing a general activity) and one that is part of a prescription. The second challenge is how to ensure that, once integrated, patients who reschedule prescriptions do so within the specified safety constraints of the prescription. The second challenge implies that the calendar should be “aware” of the prescription related entries. Overall, this chapter is dedicated to addressing the research question regarding the utilization of visualization techniques to effectively communicate and encourage favorable adherence behavior within a care team setting. The sub-questions include: (i) The optimal integration of prescription management into electronic calendars, (ii) The means of visually distinguishing medication-related data from standard calendar entries, and (iii) The effective visualization of unsafe drug interactions on the calendar. To address the research questions, we conducted a study that revolved around the design goals associated with these questions.

The main design goal of the study presented in this chapter is to integrate prescriptions into calendars. We call this goal; Design Goal number one ($\mathbf{Goal}_1$). Integration pf prescriptions into calendars should have support for safety checks. Safety checks prevent (or at least indicate) changes to schedules that do not adhere to prescribed constraints. We call this goal; Design Goal number two ($\mathbf{Goal}_2$). To achieve these goals, we considered ways of rendering medication entries so that they are easily identifiable and that all medication-related information is clearly conveyed. we also explore ways of rendering unsafe drug-interactions that may arise when medication administration times are being scheduled or changed. This is an important aspect to consider because administration times that conflict with a prescription are considered a form of non-adherence [158].

To explore design variations for Goal_1 and Goal_2, we begin by identifying and discussing considerations for the design of calendars that support medication prescriptions. The envisioned calendars should allow for the scheduling of medication actions alongside other everyday activities and provide a way for rendering and resolving conflicts when they are raised by unsafe schedules (i.e., schedules that violate constraints specified in the prescriptions). Later on in the chapter, we present the results of a qualitative study with twelve participants interacting with alternative calendar designs. Results indicate the potential benefit of equipping electronic calendars, that are already in use by many patients, with additional functionality to support the scheduling of medication prescriptions. Users are generally in favour of using such an integrated approach that leverages their familiarity with existing tools. Results also show that it is feasible to design calendars that effectively communicate unsafe medication schedules.

In the remainder of this chapter, we first provide a review of relevant literature. We then describe the data format of medication prescriptions and basic usability requirements for an on-calendar prescription visualization, before introducing three alternative designs. We subsequently report on the design of the user study, its methodology and results, before deriving a set of design guidelines for integrating prescriptions into calendars.

5.2 Related Work

In this section, we present a review of the literature relevant to on-calendar prescription visualization. This includes work related to medication reminders, visualization of schedules, and on-calendar visualizations.

5.2.1 Medication Reminders

Reminders are among the most common technological interventions targeted at improve adherence to medication [70, 37, 71]. Reminders can take many forms, including interventions of caregivers through video and voice calls [159] and text messages [160], smart pill boxes [149], and computer applications [75]. Focusing on systems that visually represent reminder events to stay within the scope of this research, we identified two relevant technological interventions.

The first one is a health literacy tool called Medication Calendar [76]. It was designed to improve antihypertensive medication adherence. The Medication Calendar provides a graphical view of medication to be taken during a given section of a day. Its layout shows Morning, Afternoon, Evening, and Bedtime as columns and the medications as rows. It displays (in text format) the name of the medication, the time of day that it should be administered, the number of times daily that it should be taken, dosage information, and additional clinical indications. The application then triggers reminders when it is time to administer the medication.

The second relevant intervention is a tool for representing graphically enhanced interventions for coronary heart disease [130] The tool shows the time of day (Morning, Afternoon, Evening, and Bedtime) as columns and the list of medication as rows. An additional column indicates the purpose of the medication. Row headers include medication name, dosage, and the time of day when it should be administered. The novelty in this work is that the table cells contain graphical images of the corresponding medication. The idea of using graphical elements to indicate administration events is something we have also adopted. We however do not intend to use pictures of actual drugs as is the case here, as we believe this is something that can only work for limited number of drug samples (unless a generic image is used).

These two contributions provide foundational design elements to consider for visualizing medications and their reminders in a calendar-style design. Specifically, the layout (columns and rows) and design concepts (graphical representation of drug entities) provide a starting point for an integrated design.

5.2.2 Visualization of schedules

To introduce medication reminders and schedules in general calendar applications, we turn to related work on visualization of schedules. Defined broadly, visualization of schedules involves the representation of planned events on a timeline that depicts a sequence of events (where each event has a single time point) or interval event (continuous quantitative time-series data) [161]. The events are often represented as bars that span multiple time points or nodes that are attached to single time points.

Gantt charts [162] and Pert charts [163] are two standard ways of representing event schedules that have been extended (addressing issues such as visual clutter and event prioritization) to solve planning problems in the field of Engineering (e.g., [164, 165, 166]). While not exclusively focusing on visualizing *planned* events, in the healthcare context a large number of systems have been developed for visual analysis of patient's cohorts temporal data (that also consist of series of temporal events). Lifeline [167, 168], Lifeflow [169], Prima [170] and TimeSpan [171] are some of the works that have explored visualization of timelines to show patients' temporal information such as medication histories, hospital visitations, and treatment processes. Walker et al. [172], developed patients' visualizations for emergency department wait times.

In this research, we focus on i) visualizing a single patient's data, and ii) visualizing events on typical calendar layouts, which often consist of two-dimensional charts where one dimension shows days and the other dimension shows the time of day [173]. Alternative layouts exist and we discuss them in the next subsection.

5.2.3 On-Calendar Visualization

Two dominant models are used to visualize events on a calendar: the radial model and the linear model [174]. With the radial model, data points are positioned along a circle, ellipse, or spiral [175]. Lines are usually drawn from the center of the circle to its circumference at equal adjacent spacing to represent time points. Rings drawn from the center of the circle extending outwards are added to further divide the temporal dimension. Popularized by William Playfair [176] and Florence Nightingale [177], radial calendars have proven effective in visualizing univariate calendar events. Variations of these include Radar Bars [178] and Radar Plots [179]. Radial calendars have been used to show daily consumption of provisions of everyday supplies for a whole calendar year by varying the color and size of marks along the rings [174]; to visualize personal data obtained from different body sensors [180] by positioning stacked bars on a single ring with 24 partitions for a 24-hour day; or quantified-self data [181] by showing streamgraphs and heatmaps of multi-year data along a spiral. Although hailed as state-of-the-art [178] and space-efficient [182], the radial model has limits when it comes to representing multivariate data and data without a defined time limit.

The linear model does not suffer from the same limitations as the radial model. It entails the conventional calendar layout with rows and columns representing days and times of day (or vice versa), and each cell representing a time interval of a particular day [183, 33]. It can also be the section of a day such as morning, afternoon, and evening, or a minute. Huang et al. incorporated quantitative user feedback data, collected from daily activities using Fitbit, within a personal digital calendar [97]. Basing their design on Google calendar [33], they overlaid the calendar with horizontal line graphs to show a user's level of activity each day. A follow up user study confirmed that a personal calendar can provide rich context for people to reason about their data. Wijk et al. used a linear calendar to show both the power demand and employee presence at a research facility [154] using color hue, height, and color saturation. Pagno et al. used a linear calendar, as an extension to their radial calendar already discussed, to show the activity level of users per given day [180]. MineTime Insight [184] is a tool for improving short and long-term scheduling decisions. It uses a calendar design to show multiple coordinated views for the exploration of personal calendar data. This tool however, concentrates on the analysis of calendar data and not on augmenting the calendar with additional information, such as medication events. Researchers have also investigated shared calendars [185], and more recently, how calendars can be used to enhance team communication and collaboration [186]. In the latter, they augmented a calendar with visualization support for the exploration of conversation data generated by team messaging platforms like Slack. They used color to identify keywords in messages, size to indicate word frequency, and lines to show connections between messages within the channel.

Incorporating non-standard data, such as fitness data or daily activities, on a calendar view has been studied at length [180, 174, 184, 185, 186]. Calendars have been effectively used to visualize personal information within the user's daily workflow. However, these efforts do not fully support the visualization of medication schedules requirements because most research efforts i) focus on representing univariate and quantitative data; ii) tend to ignore the standard calendar events already present; and iii) are usually limited to supporting events with only two attributes (time and a quantitative attribute) [171, 156, 167].

Visualizing events with a temporal and a quantitative attribute is usually achieved through adding line graphs [97], shading [186], or bars [180] to the calendar. Those are the only visual elements added to the calendar view. My work differs from this scenario because in our context, events have more than two attributes (allowed placement interval, preferred time, duration, and other descriptive units), are to be presented alongside the normal calendar events, and mechanisms are needed to visually communicate conflicts between events. To the best of my knowledge, there is no existing calendar design that presents a way of visually communicating conflicts that may arise between two or more events, and visualizing prescriptions with their temporal constraints alongside other activities in a standard calendar has not previously been investigated.

5.3 Medication Prescriptions

To design an on-calendar visualization of prescriptions, we need to i) understand the relevant characteristics of a prescription; and ii) the usability requirements such a system should support.

5.3.1 Prescription Characteristics

As presented in Section 4.3.1, the design of an on-calendar prescription visualization should support medication entries with the following attributes: Drug name (D1), Drug dose (D2), Drug dose frequency (D3), Duration (D4), and Drug indications (D5).

5.3.2 Usability Requirements

An on-calendar prescription visualization should support tasks that relate to reading calendar entries (e.g., reading calendar events and accessing event information) and to managing calendar entries (e.g., adding, editing and deleting calendar entries).

Representing medication prescriptions in general-purpose calendars requires adding visual information about these prescriptions on top of an existing planned schedule. This means that such a calendar must support tasks related to reading calendar entries (basic function of a calendar) as well as reading information about prescriptions and their potential conflicts.

Calendar Layout

Tasks related to reading standard calendar entries have been discussed in previous work [187, 188, 189]. These include tasks related to retrieving temporal features and reading event-related information such as date, time, location and purpose. From this, we derive the following *basic* usability requirement:

> R_{layout}: *The user should be able to correctly and efficiently read the calendar's temporal features. These features include the calendar's current day, month and year; the days of the week; and the times of the day.*

Medication Entries

The addition of medication entries in the calendar introduces new visual elements. Medication entries communicate more details (D1 - D5) than the standard title, time, and location of regular calendar entries [104, 106, 105]. Integrating prescriptions in a general-purpose calendar involves integrating Personal Health Information (PHI) and related activities into calendars [190, 191, 98]. Therefore, a design that integrates prescription visualization to a calendar should satisfy the following *prescription-related* usability requirement:

> R_{medic}: *The user should be able to accurately and efficiently identify the calendar's medication entries. They should also be able to read the entries' name (D1), dosage (D2), frequency (D3), duration (D4), and indications (D5).*

Medication Conflicts

Prescriptions come with constraints, e.g., drug dosage, administration frequency and other indications. Constraints communicate what should not be done during prescription implementation. They may relate to a single medication or a set of different medications. An example of within-medication constraints is found in the the prescription *Take 600 mg of Ibuprofen three times a day as needed with food*, that has three constraints: 1) that 600mg should be taken at a given time; 2) that the maximum number of intakes per day is three; and 3) that the drug must be taken with food.

But medications are often more complex. Consider for example the following prescription: *Take Tenofovir 1 tablet once daily and Metformin 500mg once daily. Take Tenofovir and Metformin at least 6 hours apart.* This example includes within-medication constraints: both Tenofovir and Metformin must be be taken once daily, and the dosage is 1 tablet for the former and 500mg for the latter. This example also includes a between-medication constraint: the two medications should be taken at least 6 hours apart. A violation of constraints (within- or between-medications) is what we call a scheduling conflict. In this chapter, my focus is on conflicts that deal with restrictions in the scheduling times of medications that may be either unsafe or ineffective when taken together. As explained in Section 4.3.5, the nature of these conflicts falls into two categories: too close (those with lower limit time constraints) and too far apart (those with upper limit time constraints). Previous work has highlighted that a calendar that supports event attributes should also have mechanisms for dealing with conflicts that arise between entries [192]. This is true for medication entries. Therefore, a design that integrates prescription visualization to a calendar should satisfy the following *conflict-related* usability requirement:

> $R_{conflict}$: *The users should be able to correctly identify conflicting medication entries in the calendar. They should also be able to name the medications involved in the conflict, identify the nature of the conflict, and envision actions that may be taken to resolve the conflict.*

These three usability requirements are basic requirements that an on-calendar prescription visualization should satisfy. The proposed requirements align with the two design goals outlined for the study. R_{layout} and R_{medic} address Goal_1 and inform the requirements necessary to integrate prescriptions into calendars. $R_{conflict}$ addresses Goal_2 and informs the integration of safety checking in the calendar. In the following section, we present three calendar designs that were created to satisfy these requirements.

5.4 On-Calendar Prescription Visualization Design

In this section, we propose three designs that were created to show the data (D1 - D5) and meet the usability requirements R_{layout}, R_{medic} and $R_{conflict}$. We start with an explanation of the ideation session that was held to come up with the designs and end with a presentation of the three designs.

5.4.1 Ideation Session

The design process was started with an ideation session involving eight researchers with background in human-computer interaction and visualization. The goal of the session was to explore layout variations and features that a calendar with integrated medication prescriptions should have. The researchers were asked to sketch calendars that show all the data (D1 - D5) and satisfy the given requirements. One ideation

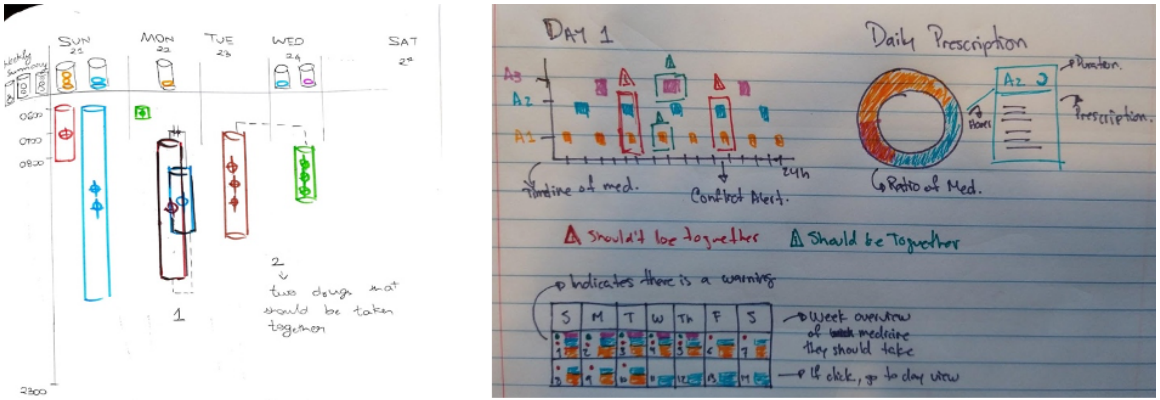


Figure 5.1: Sample designs from the ideation session showing the use of position, shape, color, and size to represent medication entries within a calendar.

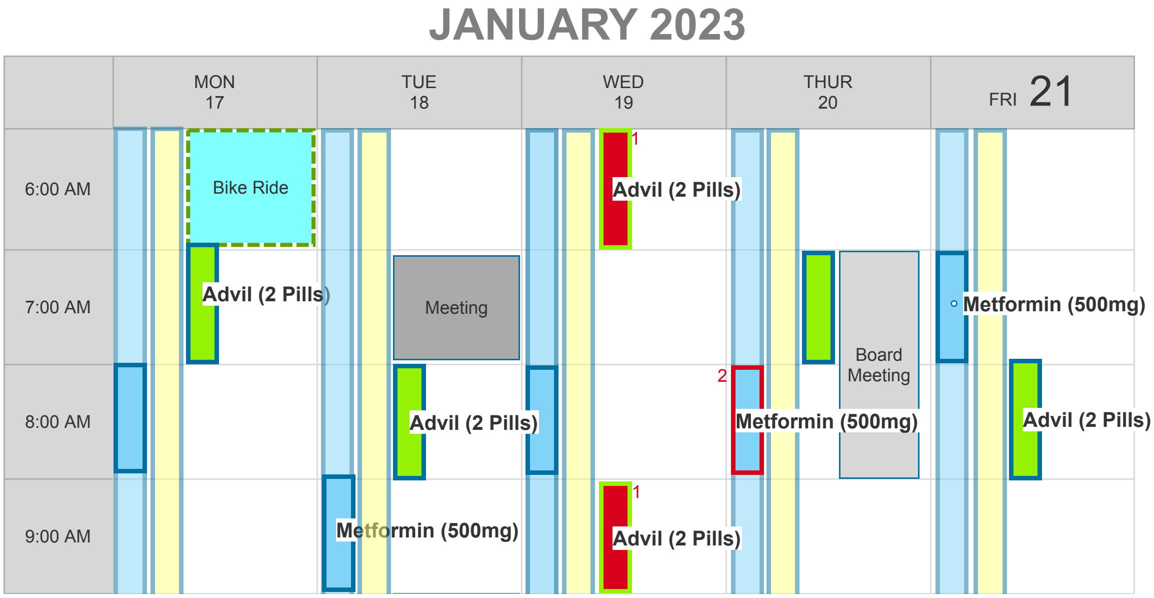


Figure 5.2: $Design_A$ with colored vertical bars used for medication entries and gray rectangular entries used for other activities.

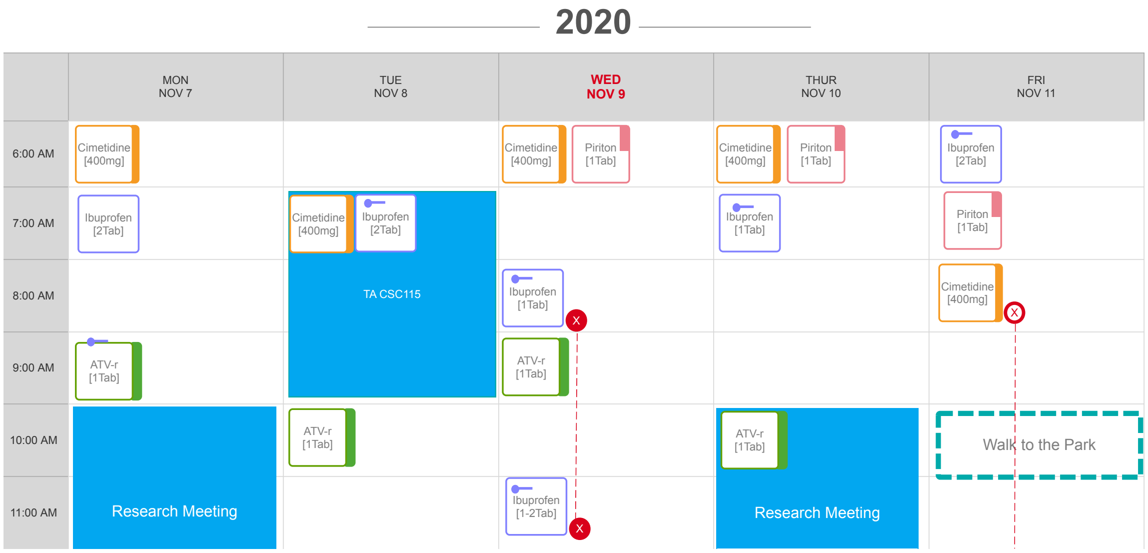


Figure 5.3: $Design_B$ with marked rectangles bearing a colored outline used to represent medication entries. Rectangular entries with dashed outlines represent Physical Activities. Blue full-width rectangular entries show other activities.

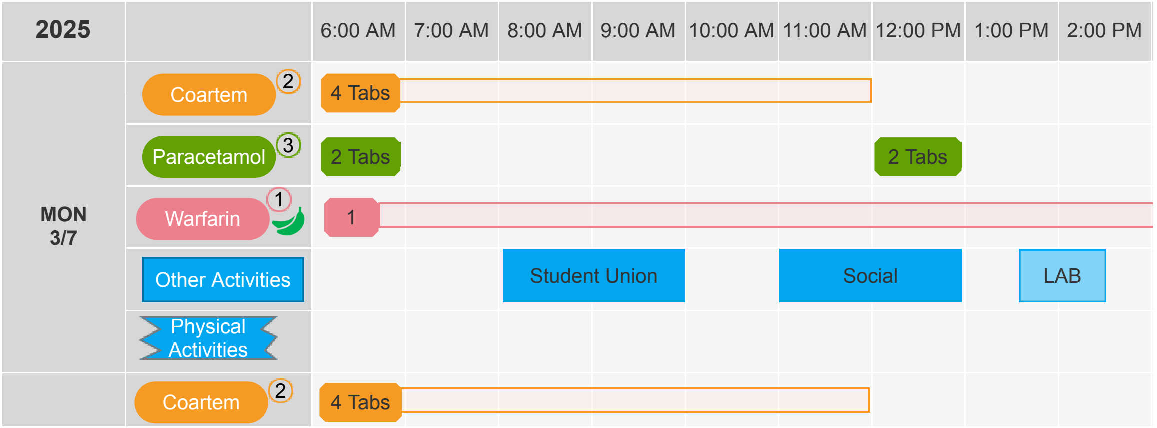


Figure 5.4: $Design_C$ with medication entries represented as sliders that communicate *allowed* administration time interval and preferred administration time. Each medication entry has its own row. Rectangular entries with and without rugged side-edges represent physical activities and other activities, respectively.

session was held for this task and it lasted for 30 minutes. Participants used pens, colored markers, pencils, and regular printing paper for their designs.

Several design dimensions were identified together with their variations from the different sketches that were produced. These included: 1) the layout of the calendar (linear or cyclic), 2) the positioning of the days and times of the day (on the left or at the top), 3) the shape of drug entries (rectangular, cylindrical, or circular), and 4) the orientation of the calendar (vertical or horizontal). Various sizes, colors, and shapes were also used in the designs. Connecting lines were predominantly used to denote the presence (and absence) of conflicts. Fig 5.1 shows some designs that came out of the ideation session.

A review of the designs from the ideation session led to a realization that the space for design alternatives was large. Narrowing down the variations, a decision was made to go for three overlapping calendar designs. The term "overlapping" is here used in the sense that the three calendars shared a lot of common designs.

Three designs ($Design_A$, $Design_B$, $Design_C$) were created by considering i) variations according to these design dimensions, ii) the constraint of compatibility with already existing calendars, and iii) the intention to remain as close as possible to the design of regular medication schedules. The three designs cover a range of design variations regarding layout, representation of medication entries, and representation of conflicts. This allowed us to assess the usefulness of design variations at a component level, rather than at an overall design level. The following subsections discuss the resulting three design variations according to our usability requirements.

5.4.2 Calendar Layout (R_{layout})

The linear layout [174] was employed for all three designs. We discarded the radial layout to stay as close as possible to both regular calendars [33, 183] and medication schedules [104]. $Design_A$ and $Design_B$ show the days of the week at the top of the calendar and the time of day on the left, like most digital calendars. $Design_C$ shows the days of the week on the left and time of day at the top, like most medication schedules. The days of the week are written in abbreviated form in all designs; the month of the year is written in a numerical form for $Design_A$ (E.g., 1 for January) as presented in numerical calendars [193]. This was adopted for $Design_C$ because it offered the most efficient use of space. The year was written on top of the calendar for $Design_A$ and $Design_B$ and inside the calendar for $Design_C$. The name of the month was written on top of the calendar in A and inside the calendar in $Design_B$ and $Design_C$.

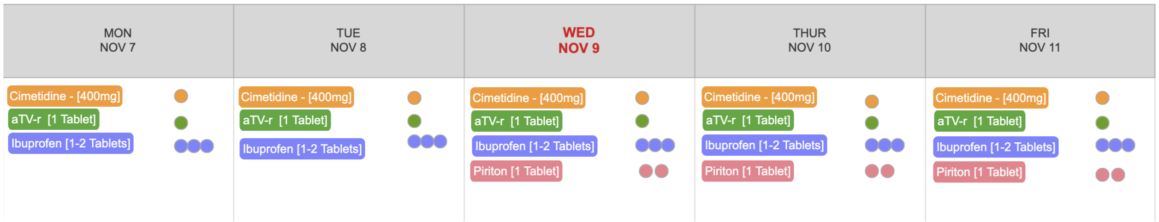
To differentiate headers from entry cells, headers have a grey background whereas cells have a white background. Each design highlights the current day: $Design_A$ does so with a bold and bigger font size like in MS Outlook [183]), $Design_B$ with a red font color, and $Design_C$ with a grey background like in Google Calendar [33].

5.4.3 Calendar Entries (R_{medic})

All three designs show entries using rectangular shapes like most digital calendars do. However, design variations in terms of position, color and size were explored with each calendar design. From the ideation session, these entries had the forms of rectangles, ellipses, cylinders, and circles. For all three designs, a decision was made to use rectangles as they are effective in being extended to cover extended width or height without using a lot of space.

$Design_A$, shown in Figure 5.2, maintains the layout of existing calendars such as Google Calendar [33] and MS Outlook Calendar [183]. The height of rectangular entries indicates their duration, color is used to differentiate types or categories of entries (as set by the user), and their names are conveyed with textual labels. In this design, medication entries are also represented with rectangles (or bars), whose vertical position and height indicates start and end of the allowed administration period for the medication. Medication entries have an embossed horizontal marker placed at some point along the bar to indicate the planned administration time (at which point the reminder would trigger if programmed). As an illustration, a user can set the administration time of 8 AM for a drug that is supposed to be taken anytime in the morning. In such a case, the marker would appear at 8 AM. Preferred administration time of a medication entry is shown with higher opacity and allowed administrative time with lower opacity. Color hue encodes the type of medication.

$Design_A$ supports medication (or drug) entries and physical activities. Each drug entry in the calendar is labelled with the name of the drug and suffixed with bracketed drug dosage. The suffix -WF indicates that the drug should be administered with food. Physical activity entries have a full-color fill, a dashed border, and a label indicating the name of the activity. All other calendar entries are represented with rectangles filled with different shades of grey. Because the shape used by other entries was similar to the one used for medication entries, color hue was the intended primary



MON NOV 7	TUE NOV 8	WED NOV 9	THUR NOV 10	FRI NOV 11
Cimetidine - [400mg] ●	Cimetidine - [400mg] ●	Cimetidine - [400mg] ●	Cimetidine - [400mg] ●	Cimetidine - [400mg] ●
aTV-r [1 Tablet] ●	aTV-r [1 Tablet] ●	aTV-r [1 Tablet] ●	aTV-r [1 Tablet] ●	aTV-r [1 Tablet] ●
Ibuprofen [1-2 Tablets] ●●●	Ibuprofen [1-2 Tablets] ●●●	Ibuprofen [1-2 Tablets] ●●●	Ibuprofen [1-2 Tablets] ●●●	Ibuprofen [1-2 Tablets] ●●●
		Piriton [1 Tablet] ●●	Piriton [1 Tablet] ●●	Piriton [1 Tablet] ●●

Figure 5.5: Part of $Design_B$- Daily summary showing the name of the medication, the dosage, and number of times that a medication was supposed to be administered per given day.

identifier for Medication entries. This dictated the avoidance of using hue as an identifier for other activities. Hence, all other activities were represented with different shades of grey to differentiate differentiate them from medication entries. A white full-width bar was included for sections marked *Do-not-disturb* which would not allow an event to be scheduled in the specified window.

$Design_B$, shown in Figure 5.4, maintains the layout of existing calendars and maintains the standard way of representing calendar entries. As such, this design consists of not altering the representation of the existing base calendar and its entries, and to *overlay* a representation of medication entries. $Design_B$ has the same layout as $Design_A$. Medication entries, like other entries, are shown with rectangular shapes. They have no fill color, have an outline color hue that conveys the type of medication, a solid outline or a dashed outline if they represent drug or physical activity respectively, and a label that displays the name and dosage of the medication. A filled circle and protruding vertical bar (to illustrate a spoon) in the top-left corner of the rectangle indicates whether the medication should be taken with food or not. A small filled rectangular inset indicates the period of the day when the medication should be administered through its vertical position and height. No marker indicates that the medication should to be taken during the hour on which the entry is positioned, a full-height marker indicates that it can be administered at any time of the day, and a partial-size marker specifies the period of the day that it should be administered (top for morning, middle for afternoon, and bottom for evening). This encoding makes it possible to visually convey larger time periods for administration without cluttering the calendar with large rectangles. $Design_B$ also has a daily summary for medication that is supposed to be taken on a given day (See figure 5.5). The summary indicates the name of the medication, the dosage, and the number of times that a medication should be administered. Color hue was used to distinguish

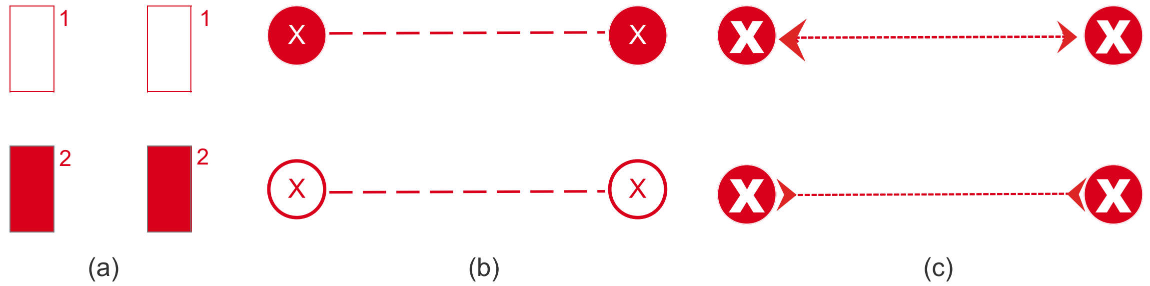


Figure 5.6: Conflict representations: (a) filled and outlined entries for $Design_A$, (b) dashed lines with filled and outlined circles for $Design_B$, and (c) directed solid and dotted lines for $Design_C$.

between different medications.

$Design_C$, shown in Figure 5.4, has a layout different from $Design_A$ and $Design_B$ as it maintains the layout of a common medication schedules. It shows days of the week on the left and times of the day in columns at the top, giving a layout with daily entries that read from left to right (as opposed to top to bottom). Each row in the calendar corresponds to a medication entry within a day.

The first column of a row shows the name of the medication, the number of times the medication should be administered that day, and possibly an indication if the medication should be administered with food (visualized with a banana icon). The medication dosage and administration time are represented with a slider. The position and length of the slider indicate the period in which that medication can be taken. The rectangular buckle indicates the preferred time.

Two additional rows are displayed: one for physical activities, and one for all other, standard calendar entries. They are visually differentiated by using different types of edges. The start and width of a block indicates the start and duration of the activity.

5.4.4 Conflict Visualization ($R_{conflict}$)

Visualization of conflicts between medication entries is shown differently in the three designs, considering i) that conflicts can be of two types (too-apart or too-close) and ii) the results from the ideation session (color and connecting lines were dominantly used to denote conflicts). Figure 5.6 shows how conflicts are shown in each design.

Some designs featured rectangular enclosures for conflicting pairs. For $Design_B$ and $Design_C$, lines were used to show conflicts and match conflicting entries because lines do not overwhelm the calendar (and would generally be effective in connecting points as employed in graphs [194]). For $Design_A$, conflict annotations were matched using numbers.

$Design_A$ stays as close to the features already supported in the calendars as possible. As such, fill and stroke were used to denote the presence of conflicts between two entries. An entry that is flagged as being part of a conflict has its block filled with red (for too-close) or its outline changed to red (for too-far-apart). Numeric labels indicate the conflict identifier i.e., two entries marked with the same number are part of the same conflict.

$Design_B$ indicates conflicts between entries with connecting red dashed lines – an approach that is effective in showing connection between points as employed in node-link diagrams [194]. Start and end of line markers convey conflict type: a filled circle with a cross mark inset indicates a too-close conflict, and an outlined circle with a cross mark inset indicates a too-far-apart conflict.

$Design_C$ indicates conflicts using connecting lines like in $Design_B$. However, in this design the line style encodes the type of conflict (solid line with arrows pointing outwards for too-close and dotted line with arrows pointing inwards for too-far-apart). Start and end of line markers are filled circles with a cross mark inside.

These are the three designs that were arrived at. As earlier indicated, the design space offers unlimited possibilities. Any other design could fulfill both data and usability requirements and be just as well rationalized.

5.5 Study Rationale and Methodology

Three prototype calendars were designed with the aim of studying design variations of layout, representation of entries, and representation of conflict. The goal of this study was to assess the quality of design variations, rather than the quality of each design as a whole.

The three designs were hosted on a public server. Remote moderated think-aloud sessions were conducted via Zoom. Participants were asked to complete a series of tasks with each design variation. The order in which the designs were presented to the participants was counterbalanced and each participant was randomly assigned an order. Each session lasted for approximately one hour. The study was approved by our institutional ethics board.

5.5.1 Participants

Participant recruitment was informed by the focus of this research on active people with frequently changing appointment schedules and who manage medication prescriptions [195]. The recruitment was targeted at people who were between the ages of 35 and 65, who were taking (or helping someone taking) medication regularly, and who considered themselves to have a reasonably busy schedule. Participant recruitment was conducted through social media platforms, university mailing lists, and posters displayed on campus. Those who were interested reached out to the researcher and were recruited on a first-come-first-serve basis. Participants were compensated with a CAD20 Amazon e-gift card. Twelve participants (P) were recruited and completed the study (10 female, 2 male). They were aged between 35 and 54 years old. All participants were either taking and/or helping someone taking multiple medications: nine were on medication, two were both on medication and helping another person on medication, and one was only helping another person with their medication. There were five students, two nurses, three information and communication technology workers, one human resource consultant, and one professor. Although participants self-selected based on our criteria (we did not question their perceived business), the frequency at which they used a calendar was a proxy for us to measure this. The frequency with which participants used a calendar to manage their schedule was daily (9), very often (2) and none (1 – this participant indicated frequent use of Medication Administration Record).

5.5.2 Study Procedure

The study began with the participants agreeing to the consent form. Then, they were asked to provide demographic information and were provided a URL to access the designs from their browser. They went through a familiarization phase during which they could study the guide (legend), the layout, and the content of the three designs. The participants were encouraged to think aloud throughout the session.

The participants were then asked to complete a series of tasks with each design. The participant performed 16 tasks: 5 addressing R_{layout}, 7 addressing R_{medic}, and 4 addressing $R_{conflict}$. Table 5.1 summarizes these tasks. After they had completed each task, the participants were asked whether there were features of the design that they found either useful or not useful to complete that task. After tasks $T_{MED;entry}$ and$T_{MED;day}$ under R_{medic}, and tasks $T_{CON;entry}$ and $T_{CON;type}$ under $R_{conflict}$, partic-

Table 5.1: Tasks participants performed with each of the designs, grouped by usability requirement each task addresses. The last two columns indicate which measures were collected for each task.

Usability Reqs	*Task Identifier*	*Task Description*	*Correct-ness*	*Completion Time*
R_{layout}	$T_{LAYOUT;year}$	Reading the current year	x	x
	$T_{LAYOUT;month}$	Reading the current month	x	x
	$T_{LAYOUT;week}$	Locating the day of the week	x	x
	$T_{LAYOUT;day}$	Identifying the current day	x	x
	$T_{LAYOUT;hour}$	Locating the hour of the day	x	x
R_{medic}	$T_{MED;entry}$	Identifying medication and non-medication entries present in the calendar	x	
	$T_{MED;day}$	Reading medication to be administered on a given day	x	x
	$T_{MED;repeat}$	Reading the number of times a medication is repeated in a given day	x	x
	$T_{MED;cycle}$	Reading other days of the week when a given medication was administered	x	x
	$T_{MED;dosage}$	Reading the dosage of a given medication	x	x
	$T_{MED;food}$	Indicating whether a given medication is to be taken with food or not	x	
	$T_{MED;slots}$	Identifying alternative slots for a given medication entry	x	
$R_{conflict}$	$T_{CON;entry}$	Identifying conflicts that are present in the calendar	x	
	$T_{CON;med}$	Naming the medication involved in a conflict	x	
	$T_{CON;type}$	Determining the type of a conflict	x	
	$T_{CON;resolve}$	Suggesting actions for resolving an identified conflict	x	

ipants were asked to evaluate the difficulty of performing the task. We selected these tasks because other tasks within each usability requirement are dependent on them. Once a participant had completed all tasks for a given design, they were asked for design suggestions before moving to the next design.

At the end of the session, participants were asked which design they preferred overall and how likely they were to adopt any of the designs for the management of their medications. They were given an opportunity to suggest improvements to each design in order to better suit their medication management routines and asked if they had any concluding remarks before the session ended.

5.5.3 Data Collection

Both audio and video streams of the sessions were recorded. The audio component also included three verbally administered questionnaires.

The *demographics questionnaire* was completed at the beginning of the session. The *difficulty questionnaire* asked participants to indicate the difficulty of: (i) differentiating medication entries from non-medication entries ($T_{MED;entry}$), (ii) reading the medication to be taken at a given time ($T_{MED;day}$), (iii) identifying the presence of conflicts in the schedule ($T_{CON;entry}$), and (iv) knowing the nature of the conflict ($T_{CON;type}$). Answers to these questions were provided on a 7-point Likert scale (Very

Easy, Easy, Somehow Easy, Unsure, Somehow Difficult, Difficult, and Very Difficult). The *adoption questionnaire* was for evaluating the likelihood of using a calendar that employs visualizations to show schedules that include visualization of medication prescriptions alongside other activities. Answers to these questions were provided on the same 7-point Likert scale.

We recorded the correctness (measure of success), and the completion time for tasks with succinct answers. We do not report completion time for tasks that had participants provide descriptive answers that could range from one or two seconds to dozens of seconds, as this would report the time it takes to provide a description and not the time it takes to find the answer to a question.

5.5.4 Data Analysis

We performed a qualitative analysis of participants' think-aloud, reactions, opinions, and suggestions. We transcribed the entire sessions using Otter [196] and used NVivo [197] for the analysis. We employed three rounds of coding: first open coding (identifying any interesting concepts), then selective coding (grouping the concepts), and finally axial coding (relating the concepts) [198].

In the open coding stage, one researcher analyzed the transcribed data and coded the data according to the task addressed. The codes from this round were grouped according to the design referenced. Data for all the 12 participants was open coded by the same researcher. This resulted in twelve codes replicated across the three designs.

The second round of coding involved carefully examining the text to identify emerging concepts across designs. A single participant's data was selectively coded by the same researcher to gain insights into the depth of user sentiments on various aspects of the designs. A second researcher who was not involved in the study independently coded the data for the same participant. Notes were then compared and the categorization and naming conventions used for the codes were harmonized. This stage resulted in 34 codes.

These codes were further analyzed in a round of axial coding to identify and define relationships between them. This analysis resulted into three categories: *temporal features of the calendar*, *design of medication entries*, and *rendering of conflicts*. These findings are discussed in detail in the results section (below). We structure the report of the study results under these three themes.

To complement the qualitative analysis, we analyzed the quantitative data (correctness and completion time) using descriptive statistics. We used this approach because of the small sample size. The small amount of data collected yields too low statistical power to confidently draw conclusions between design variations. We conducted the quantitative analysis using Tableau Software [199] and relied on the median as the measure of central tendency.

5.6 Study Results

We first report the study results for each task using completion rate and completion time as well as contextualizing qualitative feedback. Then, we present the participants' preferences regarding the different designs and individual design elements.

5.6.1 Task-based Results

We break down the task-based results according to our three task categories: those that deal with the temporal features of the calendars and address R_{layout}; those that deal with the design of medication entries and address R_{medic}; and those that deal with the rendering and conflicts and address $R_{conflict}$. Figure 5.7 shows task completion (correctness) for all tasks and for each design. We report on completion times for all tasks addressing R_{layout} and for four of the seven tasks addressing R_{medic} (see Figure 5.8). Figure 5.9 shows participant's evaluation of the difficulty of performing tasks $T_{MED;entry}$, $T_{MED;day}$, $T_{CON;entry}$, and $T_{CON;type}$.

Reading Temporal Features

Here we present the results for the five tasks that addressed R_{layout}. Correctness and completion time were recorded for all five tasks.

Most participants successfully completed all R_{layout} tasks quickly with all designs. No comments were made about $T_{LAYOUT;week}$, $T_{LAYOUT;hour}$, and $T_{LAYOUT;year}$.

For $T_{LAYOUT;month}$, five participants (P3, P4, P6, P8, and P9) commented that showing the month with a number in $Design_C$ made reading the date difficult because they could not tell which number represented the month and which one represented the day. For example, P6 said *"I don't like the fact that March is represented by a number. I find that confusing when I've got two numbers slashed beside each other."*.

For $T_{LAYOUT;day}$, two participants (P3 and P6) thought the larger font size used to indicate the current date in $Design_A$ was an error. For example, P6 said *"That's* *why the 21 is so big. I was gonna say there's an error. There's something weird going on with the 21. Maybe Friday was a special day and it gets to be big."* P3 proposed that instead of just showing the current day, there should also be an indication of the current hour, saying *"So right now we're in the 10 o'clock block. So to have maybe a dotted outline around that block, just to let the user know, this is the time of day."* P3 also suggested that as opposed to an indication of the current day on the header, the entire column should be highlighted, and that the header should be floating to remain in place when scrolling.

Reading Medication Entries

Here we present the results for the seven tasks that addressed R_{medic}. Correctness was recorded for all seven tasks, and completion time was recorded for those with non-descriptive answers ($T_{MED;day}$, $T_{MED;repeat}$, $T_{MED;cycle}$, and $T_{MED;dosage}$).

Most participants successfully completed the R_{medic} tasks with all designs. These tasks were longer to perform than the R_{layout} tasks.

All participants successfully completed $T_{MED;entry}$ with all designs. Once they had provided their answer, participants were asked to describe which visual variables (color, size, position, and shape) they relied on to provide their answer. Participants relied on color only (6 with $Design_A$, 2 with $Design_B$, 3 with $Design_C$), shape only (3 with $Design_A$, 5 with $Design_B$, 7 with $Design_C$), size and shape (1 with $Design_A$), shape and color (1 with $Design_A$, 4 with $Design_B$), and shape and position (1 with $Design_C$). One participant did not rely on any visual variable to provide their answers with all three designs. P6, P7 and P4 used the name of the medication to identify medication entries in $Design_A$, $Design_B$, and $Design_C$ respectively. For example, P7 said after they had completed the task with $Design_B$, *"I can tell which ones the medication entries are by the names of the medications. I also noted that there was a dosage size or a number of tablets."*. One participant (P9) indicated that the difference between medication and non-medication entries was not clear until they read the legend, saying *"For the first moment it was not so clear. But after [reading the legend] the shape is thinner than the [one for] regular meetings."*

All participants successfully completed $T_{MED;day}$ with all designs in similar times. Three participants (P3, P5, and P7) complained about the need to scroll to the end of the day with $Design_A$. For example, P3 said *"I find it's a lot of scrolling down. It would be helpful if there was a way to condense it or to make it possible to see*

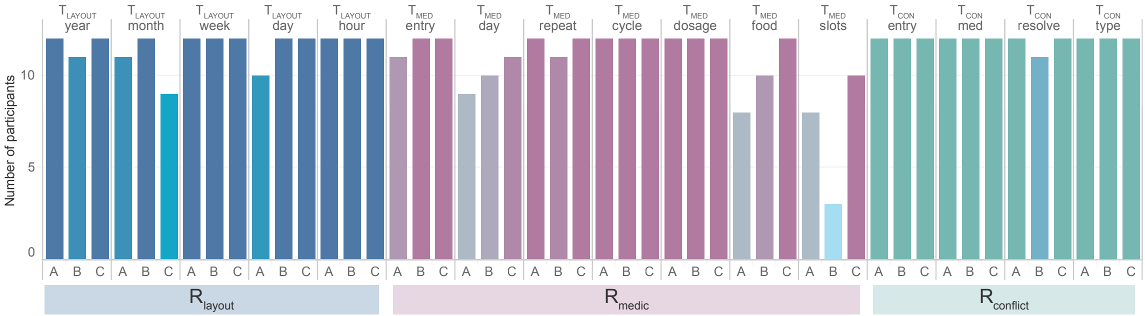


Figure 5.7: A summary of task completion for the three designs A, B, and C. Tasks in $R_{conflict}$ had the most successful completion rate. $T_{MED;slots}$ was the most failed task with Design B recording the lowest score from the entire study.

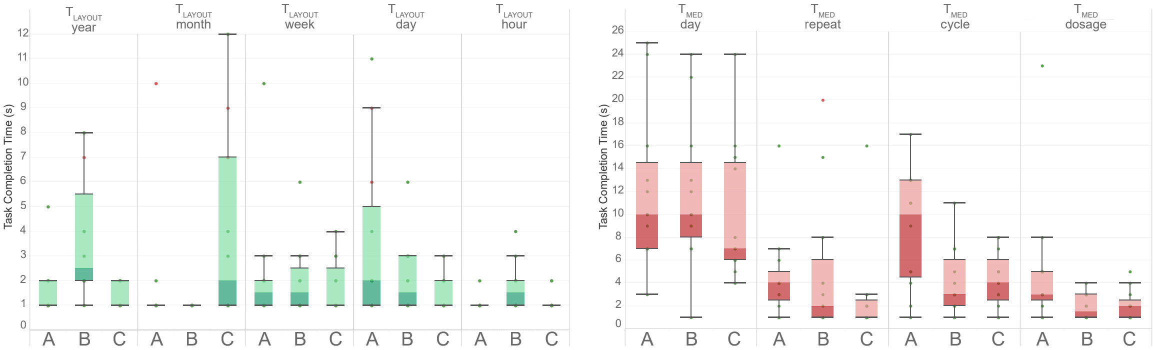


Figure 5.8: Box-plots showing task completion times for five layout tasks (R_{layout}- Left) and four medication tasks (R_{medic}- right) for the three designs. Each box-plot shows the minimum time (lower whisker), first quartile (Q1), median, third Quartile (Q3), and maximum time (Upper Whisker).

the entire calendar available in terms of morning, afternoon, and evening.”, and P8 said *“The time frames are a bit big. So it makes like I said, it really makes it scroll off that you can’t see it all in one consolidated view”*. With $Design_B$, participants were expected to use the daily medication summaries provided at the top. Three participants (P1, P6, and P9) found the daily summaries helpful in performing this task. For example, P9 said *“Yeah, I like the idea of having the first row on the calendar dedicated only for the medications that needs to be taken. I think it brings an overall idea [of] what should be taken during that day.”*. Three participants (P3, P4, and P5) also complained about the lines demarcating days not being clear. For example, P4 said *“I have a harder time differentiating the calendar component the days, because there’s not a strong border between the days of the week.”*.

All participants successfully completed $T_{MED;repeat}$ quickly with all designs. Similar complaints about the need to scroll as for $T_{MED;day}$ were made about $Design_A$ by

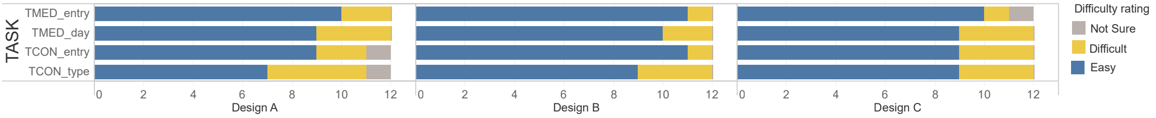


Figure 5.9: Participant's evaluation of the difficulty of performing tasks $T_{MED;entry}$, $T_{MED;day}$, $T_{CON;entry}$, and $T_{CON;type}$ for the three designs.

P3, P5, and P8. With $Design_B$, participants could count the number of dots beside the name of the medication in the daily summaries at the top. Five participants (P3, P6, P7, P9, and P10) complained that the dots were not intuitive. For example, P6 said *"So why is there three circles for purple and one circle for green and one circle for Orange? I'm not sure what that means. It's not immediately intuitive"* P3, P7, and P9 thought that the dots indicated the maximum number of pills that was supposed to be taken per day, not the actual number of pills that should be taken. For example, P7 said *"And then these numbers in the circles next to the names of the medicines, I don't know if that means the maximum number of times allowed per day. So the numbers don't really make sense to me."* No participant had trouble with $Design_C$, where the number was directly indicated beside the name of the medication.

All participants successfully completed $T_{MED;cycle}$ with all designs. With $Design_A$, they again had to scroll through the entire week, therefore took longer; with $Design_B$ they could rely on the daily summaries; and with $Design_C$ the information was readily accessible on the row headers. No comments were made regarding this task.

All participants successfully completed $T_{MED;dosage}$ quickly with all designs. Four participants (P1, P4, P7, and P10) commented on the inconsistency in the unit used for the dosage, saying that only milligrams (mg) should be used. For example, P10 said *"It's a little bit inconsistent that it's milligrams of Metformin, but not for the Advil [...] those could be in milligrams too."*. P1 agreed, *"It shows that its two pills. It doesn't say the dosage."*

Most participants successfully completed most $T_{MED;dosage}$ with all designs. 10 participants commented that an indicator for take-with-food medication should use a food icon (as used in $Design_C$). For example, with $Design_B$ P3 said *"I'd love to see a symbol that is food-related, as opposed to a slider-style tab."*. P9 added, *"I think the line for to be taken with foods is not so useful. Maybe adding like an apple or some icon that reminds us that is of food"*. P7 indicated that the *WF* (for "With Food") suffix used in $Design_A$ could also be read as "Without Food". P7 also thought that when a food icon is used (such as in $Design_C$), it should be a realistic icon and alternatively be personalizable, saying *"But I would want to make it look more like a banana. If that's supposed to be banana, they're green, but bananas are typically yellow. [...] It'd be fun if someone could choose the emoji they want to use or the picture they want to use to indicate this action"*.

Participants had difficulty completing $T_{MED;slots}$, especially with $Design_B$. To complete this task, participants could rely on the bars that indicate allowed medication intake times with $Design_A$ and $Design_C$. With $Design_B$, the marker on the medication entry indicated the allowed time and five participants said $Design_B$ did not support that task. For example, P1 said *"it doesn't show any other time of the day that you can take it"*. Three participants (P5, P11, and P12) said they could reschedule in any free slot, for example, P9 said *"It seems that 7am is a possibility because there is no other and there is no indication of conflicts."*. P10 commented they would move it to a slot and observe if a conflict was flagged, saying *"I don't know. I think I would just move it and see if a conflict came up."*. P8 indicated the use of the bar to indicate allowed schedule times could also be read as extended release time, saying *"That would mean that it's something that it's an extended release. Warfarin is not an extended release."* P10 and P6 reasoned that a medication that is supposed to be taken at a specific time point should not occupy a full hour on the calendar. For example, P10 said *"I really don't like this fact that it says 6am on the side and then it makes it a block of time." "I think I like having it be a smaller amount of time, instead of occupying the full hour"*, added P10.

Reading Conflict Annotations

Here we present the results for the four tasks that addressed $R_{conflict}$. Correctness was recorded for all four tasks, but not completion time because they all asked for descriptive answers. Overall, participants had no difficulty completing these tasks and they did so quickly.

Participants were asked to perform four tasks in this category. The four tasks centered around the ability to read conflicts and suggest possible conflict resolution strategies. Participants were asked to identify all the conflicts present in the calendar ($T_{CON;entry}$), name the conflicting medication entries ($T_{CON;med}$), indicate the nature of the conflict ($T_{CON;type}$), and suggest actions that may be taken to resolve the conflicts identified ($T_{CON;resolve}$).

Yet, participants provided comments about the designs in light of completing these tasks. For example, P4 said *"At first glance, it did not seem clear. It looked unfriendly just because there's red and axes and stuff. But talking through it with you, and I was trying to explain the differences, it made more sense."* For P8, the fill and outline used for $Design_A$ was not as clear at with the other two designs. They said *"I think switching the coloring doesn't necessarily work to bring out the conflict because you lose the fact that you expect to see this green with blue border for Advil, and now I'm instead seeing this red with green border. Is that a different pill?"* Two participants (P3 and P6) thought the design should not allow users to schedule in slots that would cause conflict, for example P6 said *"You're supposed to take one tablet every four to six hours. They shouldn't have scheduled it there."*. Four participants (P1, P10, P3, and P8) said the designs should show available slots and flag an error when the user tries to schedule a medication entry in an illegal slot. For example, P10 said *"So I don't know if there wouldn't be like, the potential spots, if you're going to ask them to move it like where you could have included the constraints to make them blank spots in there."* The four participants (P3, P8, P7, and P10) who noticed that information about the conflict was available when hovering over the conflicting entries with $Design_A$ and $Design_C$ were positive about the feature. For example, P7 said *"when you hover over it, it gives you more information about the conflict that you're having. I do like that."*

2 participants found $T_{MED;entry}$ difficult with $Design_A$, 1 with $Design_B$ and 1 with $Design_C$. The number of those who said $T_{MED;entry}$ was difficult was two for Design A, and one each for designs B and C. One participant wasn't sure for Design C. The rest said the task was easy. For $T_{MED;day}$, participants who said the task was difficult were three for designs A and C, and two for Design B. The rest said the task was easy. One participant wasn't sure on the difficulty of $T_{CON;entry}$ for Design A, two said it was difficult. For designs B and C, one and three participants, respectively, said the task was difficult. The rest said the was easy. Results also indicate that five participants thought $T_{CON;type}$ was difficult for Design A. Designs B and C had three participants each, saying $T_{CON;type}$ was difficult. The summary of the difficulty rating of the four tasks is shown in

5.6.2 Design Preferences

When asked about the likelihood of using a calendar that integrates medication entries, eight answered positively (1 somehow likely, 3 likely, 4 very likely) and four answered negatively (1 somehow unlikely, 1 unlikely, 2 very unlikely). When asked which design they preferred, 1 answered $Design_A$, 10 answered $Design_B$, and 1 answered $Design_C$. Below we present participants' rationale for their reservations and preferences.

Reservations

Four participants had reservations to adopt such a calendar. They provided different reasons for this. P6 wanted to avoid having a constant reminder of medication in their calendar. They said *"I live by calendars. But I have so much going on in my calendar as it is, if I start putting my medication in there as well. It gets way too much emphasis."* P7 was satisfied with their current paper-based system. They said *"Because I have a system that I enjoy using on my own. I have my pills in a pill case. I have a list that I can pull up at any time, that will remind me how many times per day what the dosages per day."* P8 did not see the need of a calendar for medication, because in their case all their medications are to be taken at the same time daily. They said *"I have my medications [which] I take once a day. I make sure that myself, it's the first thing I do when I get up in the morning. It's one of the first activities."*

Familiarity

Familiarity (or unfamiliarity) was one reason why participants opted for one design as opposed to another. This was the case for five participants (P7, P6, P5, P4, and P10). For example P7 commented about $Design_C$, *"I haven't actually used a calendar that looked like this in quite some time. And so right now, for me, this isn't a design that's useful"*. P5 said about $Design_B$, *"I think this is more of a calendar view that I'm used to seeing items in slots like this. So that's familiar."*, and P4 said about $Design_B$ *"This one reminds me a little bit more of a Google Calendar, which I'm familiar with."*

Clutter

Clutter, or lack thereof, was another reason for favoring a design raised by 6 participants (P1, P2, P5, P6, P8, and P9) indicated that $Design_A$ and $Design_C$ were too cluttered. For example, P8 said about $Design_C$: *"My first thing seeing it, it's very cluttered. It's got a lot of information on it."* P6 thought that medication entries were given too much space at the expense of other entries, saying *"Well, I don't* *like again, but there's so much emphasis on the medication, it's almost like the medication takes precedence. And everything else that you've got going on your life is almost insignificant."* P1 suggested toning down medication entries, saying *"instead of having the name of the medication, just put four tablets."* P2 found that too much color was used in $Design_C$, saying *"I think there are a lot of colors here, which makes me confused."* P6 found that the squares used for $Design_A$ could be made smaller, saying *"We [should] make the squares a bit smaller somehow or less predominant, less colorful."*

Privacy

One (P10) participant raised privacy concerns. They suggested that the entries should be discrete enough to avoid exposing information in the event that someone else accessed the calendar. *"So I don't know if it would be something that was like, more discrete or if someone was to accidentally see my calendar and like, saw that I was taking all these medications or something like that."*

5.6.3 Other considerations

Participants described a few features they thought were missing in the designs. These included the timing of reminder, the inclusion of a notation for medications that should be "taken as needed", and the inclusion of "over the counter" medication. P8 said that there should be feedback mechanism embedded into the system so that the calendar should confirm that the user has taken the medication. They asked: *"I guess to all this calendar stuff is when you need to take it. And then did you actually get confirmation of taking it?"*

5.7 Discussion and Design Guidelines

The results from our study confirm that prescriptions can be incorporated into mainstream calendars to allow for management of medication prescriptions ($\mathbf{Goal}_1$) and that unsafe drug interactions can be rendered on the calendar in a way that is easily identifiable by the users ($\mathbf{Goal}_2$). Results also indicate that patients would generally be willing to adopt a calendar system that supports this aspect. Our results also show that the design of such a calendar should not deviate too much from the way conventional calendars are designed and that medication entries should be integrated with other non-medication calendar entries. They should not occupy too much space and should be dismissible by users.

Participants preferred $Design_B$ over $Design_A$ and $Design_C$. $Design_B$ is the design that leverages features from both conventional calendars and prescription schedules. It resulted in better task completion correctness than $Design_A$ and $Design_C$ for layout related tasks (R_{layout}). $Design_C$, with its vertical layout, resulted in better correctness than $Design_A$ and $Design_B$ for medication entry related tasks (R_{medic}). The three designs led to similar correctness for conflict-related tasks ($R_{conflict}$). In terms of task completion time, $Design_A$ and $Design_C$ performed better than $Design_B$ for layout related tasks while $Design_C$ outperformed both $Design_A$ and $Design_B$ for medication entry-related tasks.

Below, we discuss the results under the following five themes that emerged from analyzing the qualitative data: use familiar design, avoid clutter, allow for personalization, support personal reflection, and highlight for attention. Each theme informs a new design guideline ***(DG)*** for integrating prescriptions into calendars.

5.7.1 Use Familiar Design

*The design of a calendar should not radically deviate from calendar interfaces that users are familiar with (**DG1**).* This was observed in various aspects of the design such as layout, medication entries, and icons used to annotate entries such as those which should be taken with food. Over 80% of the participants preferred $Design_B$ because of its probable similarity to already existing calendars. This was surprising to us because $Design_A$ was the design that was intended to resemble existing calendars. The sidelining of $Design_A$ is attributed mainly to the height of medication entries which spanned the entire allowed administration period and introduced too much clutter. The results indicate that the preferred layout should be vertically oriented with days of the week at the top and times of the day on the left.

The dosage used on the medication entry should be one that users are familiar with. The unit used should be consistent with the one used in the prescription. It should show the actual quantity (e.g., milligrams) as opposed to relative classifications such as number of pills or tablets. Similarly, realistic food-related icons (e.g., a banana) should be used to denote that medication that should be taken with food. In this case, bananas are more effective in indicating the take-with-food action than an icon resembling a utensil such as spoon. Such icon should be positioned together with the entry and not as part of medication summaries.

When dealing with conflicts, arrows are effective in communicating the suggested conflict resolution action. End-of-line arrows can be used to indicate that medication entries that have been scheduled too close together should be taken apart and vice-versa. The calendar should also have support for indication that a given entry is optional. Such entries would be used for medications that should be administered “as needed" and non-prescription medications that are sold “over the counter". These design decisions are influenced by everyday activities that users are familiar with.

5.7.2 Avoid Clutter

*The design of the calendar should avoid design elements that introduce clutter to the calendar (**DG2**).* Design elements such as colors, sliders, labels, and markers should be carefully employed to avoid overwhelming the calendar. One of the reasons why $Design_B$ was preferred is because it is less cluttered: medication entries can be rendered effectively using position, shape, and size. The size of a medication entry should be as small as possible so as not to occupy too much space. Size should not be used to indicate either allowed or preferred administration time of a medication entry, and the size of the entry should also be uniform regardless of the length of the allowed period of administration. Using shapes with a colored outline and transparent fill was associated with less noise by participants. While the slider design was effective in communicating both the allowed and preferred administration period, it made the entry occupy a lot of calendar space and was also misread by some participants. Familiar icons such as tablets can be used to indicate medication entries.

Color should be made less dominant and should not be used as the primary identifier for medication entries. While solid fill color was effective in indicating busy slots, using color fill for medication entries was cluttering the designs. The amount of medication information shown (e.g., labels, including name and dosage) should also be minimized. Labels were a source of confusion as to which entry they referred to when multiple entries occupied the same cell. They should be abstracted from the overview and instead be made available as details on demand.

Conflict overlays were easily identifiable on all the designs. Participants preferred the use of indicators for the position of medication entries that are involved in the conflict. The connectors (lines) for conflicting entries should use thin or dotted lines rather than thick solid lines. Participants found that different line styles may appear similar at a distance and hence fail in communicating the nature of a conflict.

5.7.3 Allow for Personalization

To design an effective calendar, we need to tailor the design to individual needs, values, and preferences [200]. Therefore, *the design of the calendar should have provision for users to personalize some of its features (**DG3**).* Such desirable personalization includes adding color to medication entries, choosing icons to be used for medications that should be administered with food, deciding which medication information to display on the entry, and choosing whether to use the default entry shape or substitute it for other Emojis or Icons. Personalization should also cover data privacy. The design of the calendar should have features that will protect the users' sensitive medication data from unauthorized reading. This is particularly important when a calendar is accessed by more than one person. Calendar owners should be able to hide features of the calendar that they do not wish anyone without privilege to see.

5.7.4 Support Personal Reflection

*Medication entries should have separate designs for entries that are future and those that are past (**DG4**).* Past entries should allow for reflection of past medication-taking behavior by confirming whether the user took the medication or not. The design should therefore have a way of letting users confirm that they have taken the medication to aid personal reflection.

5.7.5 Highlight for Attention

*The calendar should highlight entries to which the users' attention should be drawn on any given day (**DG5**).* This includes basic calendar layout requirements: the top of the calendar with the year, month, and weekday labels should be floating so that they are always visible; day separators should be clear and the entire current days should be highlighted; and the current hour should be highlighted. The legend did not do much to help participant understand the meaning of entries, as participants were able to read and correctly interpret the design without referring back to the legend. Less emphasis should therefore be placed on the legend. Making the legend available on request is another option.

Medication entries should have markers that communicate the times that their reminders will be triggered. The markers should not communicate time ranges but points in time when reminders are triggered. The calendar should have daily summaries of the list of medications to be administered each day. These summaries should only contain the name of the medication and users should be able to show or hide them.

Medication conflicts should be emphasized on the conflicting entries rather than on the connectors. The user should be notified of a newly created conflict upon rescheduling an entry, preferably via dismissible error messages that describe the conflict. When rescheduling medication entries, cells that are either safe or unsafe should be highlighted to the user to guide their action. Although some participants felt that the design should not allow them to schedule an entry in the space that is likely to cause a conflict, there might be situations where this possibility is unavoidable. The user should, in this case, be guided on possible moves that will resolve the conflict. This can be done by shading or using an outline for all the cells to which an entry may be rescheduled to resolve the conflict, and letting users configure the amount of warnings and error messages they want to receive.

5.7.6 Limitations

One limitation of our study is its relatively small sample size. While 12 participants is appropriate for the qualitative analysis of collected data, more participants are required to make task-based statistical comparisons between designs. That being said, the purpose of this study was exploratory. The findings from this study will allow us to turn to high-fidelity prototyping of calendar designs and to conduct such a quantitative task-based follow-up study.

Another limitation is that since the study was online, we did not have the privilege of observing participant's full activity cycles. It is likely that remote sessions also lead to participants employing less think-aloud than when participating in person. We also constrained participation to people between the age of 35 and 65 who were either on multiple prescription medications or played the role of caregivers to others on multiple medications. While this allowed us to capture insights for that specific population, these insights do not necessarily generalize to other populations.

Our calendar designs were suited for relatively large screens such as laptops and tablets and were not evaluated on mobile devices. Given the focus of our study on medication entries, we opted for assigning the same color to all non-medication calendar entries. However, events in real-life calendars are often of several colors. The added colors likely increase visual complexity and visual clutter that must be considered in future studies. Finally, the study was only limited to tasks that relate to reading calendar entries. In the future, tasks such as adding and modifying medication entries should be included.

5.8 Conclusion and Recommendations

In this research, we explored the possibility of integrating prescriptions into calendars (Goal_1). We considered ways of rendering medication entries so that they were identifiable and that all medication-related information was conveyed. We also considered ways of rendering unsafe medication schedules (Goal_2). We designed three calendars that leverage features available in both conventional calendars and medication schedules. We conducted a study with twelve participants, to evaluate the effectiveness of the calendar designs. Results show that calendars can be designed to support medication prescriptions while remaining familiar to use. Eight out of twelve participants indicated that they were between somehow likely and very likely to use such a calendar to manage their medications. The findings from our study informed five design guidelines that an integrated calendar should address: using of familiar design (DG1), avoiding clutter (DG2), allowing for personalization (DG3), supporting personal reflection (DG4), and highlighting for user attention (DG5). The successful adoption of these design guidelines will lead not only to a calendar that expressively incorporates all prescription data but to one that is compatible with everyday use of those calendars.

Chapter 6

Calendar Prototype

6.1 Introduction

Calendars are important in the management of day-to-day activities [187, 24]. Many digital calendars are designed primarily for the management of events [33, 183, 201]. Some calendars, however, offer extended functionality. Google Calendar, for example, has inbuilt support for additional entities such as tasks and appointments [33]. The extended functionality, supporting the selection of the type of entry to be added to a calendar implies that the processing of the entry is handled differently. There is a deliberate implementation of functionalities to handle those extensions.

Calendars are also used as health support tools. They are used for activities such as programming health-related reminders and managing clinical appointments, a form of healthcare support. Another important aspect of healthcare support is the management of prescriptions. In their current state, calendars can be used (with a considerable amount of effort) to manage prescription schedules, albeit without "seamless" support, because calendars are not designed to sustain them. The prescription schedule would have to be programmed into the calendar like a normal event or reminder. And it would be treated as such. Because the calendar is built with features that pertain to normal events.

In this chapter, we describe the process of building a web-based calendar prototype with support for the management of prescriptions. Through this approach, we address the primary research question of expanding the functionality of calendars to encompass prescription management. Subsidiary questions within this scope include: (i) Determining the necessary medication-related data and functionalities that the calendar should accommodate, and (ii) Exploring methods by which patients can refine prescription schedules while adhering to the safety constraints specified in the prescription. As a foundation, the prototype was developed to manage events like a conventional digital calendar. Then, we added support for the management of medication schedules. The prototype was built using an open-source web-based digital calendar library, called FullCalendar [32]. Calendar events and prescription-related entries were rendered and processed differently. User Interactions supported by the two types of entries were also feature-specific.

The prototype was also built with support for safety checks between medication entries. Since prescriptions come with an indication of temporal constraints associated with the medications contained therein, the calendar should be able to recognize violations of such constraints arising from schedule refinement by the user. The prototype used an implementation of the Simple Temporal Problem (STP) framework to support constraint checking [28].

Support for self-reflection was also added to the prototype. The system was designed to allow users to track their medication-taking behavior by tagging and updating statuses associated with medication entries. The tags and statuses would later be used during self-reflection. Visualizations for self-reflection were implemented using Apache eCharts[202].

Details of the implementation of the prototype are presented below, as follows. I begin with a review of related works in Section 6.2. The architecture of the prototype is presented in Section 6.3. Section 6.4 discusses the user interface. I end the chapter with an introduction to self-reflection in Section 6.5 and a conclusion in Section 6.6.

6.2 Related Works

In this section, I will present research that relates to the management of medication within and outside calendars. I will not present research that addresses calendar design. This has already been discussed in Chapter 5.

Siek et al. developed a Personal Health Application (PHA) for older adults to manage medications [87]. Their system was open-source and web-based. They used an iterative participatory design process that provided older adults and their caregivers the ability to manage their personal health information. The system was developed in PHP, JavaScript, and HTML and allowed the user to add medication schedules. They conducted a study using the system in which they concluded that Web-based applications have the potential to address all the medication management functions of their targeted user group. Khan et al. improved on the prototype by Siek et al. by integrating the system into an inter-operable Personal Health Record (PHR) system and linked it to authoritative information such as an established database used to obtain National Drug Codes (NDCs) [91]. They also added a local MySQL database to store patient data. A study was conducted that resulted in the proposal of guidelines that should be used in the development of PHAs. The guidelines included (i) that PHAs should be designed with the option to add advanced management functions as users grow accustomed to basic features, (ii) that the designs should have automated mechanisms that require minimal interaction to perform basic medication management tasks, and (iii) that developers work with stakeholders to find the right balance between metaphor and textual information [91].

Kendall et al. conducted research in which they interviewed and compared the experiences of 20 older adults with diabetes and 19 mothers of children with asthma to understand the use of reminder systems for at-home chronic disease management [86]. They proposed that the design of reminder systems should support self-management and account for errors by making it easier to detect, evaluate, and respond to failures when they do occur. They also observed that reminder systems often lacked a feedback loop to track if an activity was performed or to support the evaluation of errors that may have occurred [86]. Like Kendall et al., Stawarz et al. reviewed user sentiments on reminder apps and proposed design requirements for building medication reminders that support medication-taking [77]. They proposed three design requirements. These are Routine Creation (the system should suggest pairing medication-taking with an existing routine), Back-Up Notifications (users should be able to control when and how they are reminded), and Post-Completion Checks (users should be able to check whether a dose has been taken) [77].

The feedback loop identified as a missing feature by Kendall et al. [86] and proposed as the Post-Completion Check design guideline by Stawarz et al. [77], was partially addressed by Silva et al. [90]. Silva et al. created UbiMeds, a mobile application aimed at supporting prescription medication adherence for the aging and disabled population. UbiMeds had a user interface that would allow patients to easily access prescription information. The system was based on an architecture that allowed automation of prescription scheduling, reminders, and constant monitoring from physicians about the patient's medication adherence behavior. UbiMeds also supported intake tracking, where the user would indicate whether they had taken the medication [90].

The concept of checking unsafe drug interaction was researched by Wang et al. [203]. Wang et al. created Wedjat, a mobile phone-based medicine in-take reminder and monitor. Wedjat was built to remind its users to take the correct medicines on time and record the in-take schedules for later review by healthcare professionals. The application had the functionality to alert the patients about potential unsafe drug interactions and revise the in-take schedule automatically when a dose was missed [203]. The interactions considered here are unsafe drug interactions by presence, not time constraints. That implies constraints targeting two or more medications deemed unsafe to be taken together.

These works, when consolidated, provide design considerations important for the integration of medication management into calendars. Checking for unsafe drug interactions, having a feedback loop, and using minimal interactions for basic tasks, are guidelines that repeat some of the design considerations proposed in Chapter 5 and are used as the basis for the development of the calendar prototype. In the sections that follow, I discuss the implementation of the prototype, beginning with the system architecture.

6.3 Architecture Overview

The calendar prototype was built with 5 interacting client-server components (see figure 6.1). On the client side, there are two components: the Calendar Interface and the Client Controller (or API). On the server side, there are three components: the Server Controller (or API), Safety Checker, and Database. The Calendar Interface was built using Hypertext Markup Language (HTML), Cascading Style Sheets (CSS), and the FullCalendar Library [32]. Details about the calendar interface are given in Section 6.4. The rest of the components are explained below.

6.3.1 Client Controller

The Client Controller was built in JavaScript. The controller is responsible for getting user input from the calendar, sending requests and receiving responses from the server, and maintaining the user interface. The client controller comprises five major modules. These are Calendar, Interface, Listener, Forms, and Reflect. The Calendar module is the one that integrates and extends the FullCalendar library. It fetches

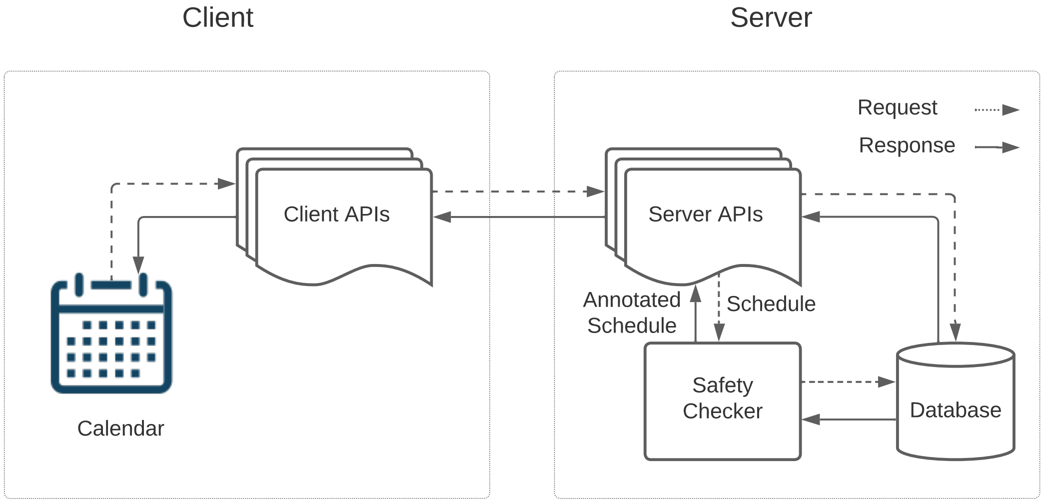


Figure 6.1: Calendar architecture showing interactions between client-side and server-side components.

events and medication entries from the server. The Interface module renders events and medication in the calendar. This module overrides the default appearance of FullCalendar. The Listener module is responsible for handling all user interactions performed on the calendar. The Forms module handles all user input supplied through dialog boxes. Self-reflection is handled by the Reflect module.

6.3.2 Server Controller

The server controller was built in PHP: Hypertext Preprocessor (PHP) and Structured Query Language (SQL). The job of the controller is to process requests from the client controller (such as creating, reading, updating, and deleting events and medication entries) and send the result back to the client. The server controller comprises three major modules. These are User (for all user-related operations), Event (for operations related to regular events), and Prescription (for all prescription-related operations).

6.3.3 Database

The database is used to store data about the users, events, and prescriptions. It was built using MySQL [204]. The database has seven tables (see figure 6.2). Two tables store data about the user (user attributes such as name, email address, and Personal Health Number (PHN)) and event (all event attributes such as title, details, timestamp, and color). The remaining five tables are dedicated to the storage of prescription-related data: Prescription, Medication, Prescription Entry, Constraints, and Status.

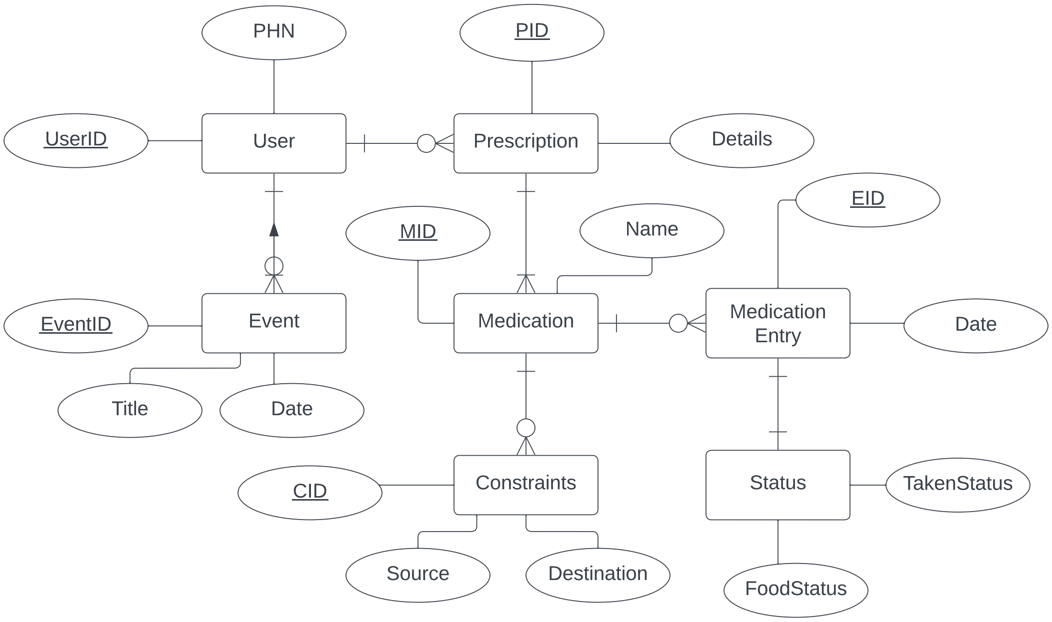


Figure 6.2: An Entity Relationship (ER) diagram showing linkages between database tables. Some attributes have been hidden.

Prescription details

All prescriptions, once received, are stored in the Prescription table in a JSON format. The design of the system was centered around the idea that the prescription will be sourced from a remote prescription service. The assumed prescription data-sharing standard is FHIR [134] and the data is received in JSON format.

Medication Details

The Medication table stores data about each individual medication specified in the prescription. For each medication, the name (D1), dosage (D2), frequency (D3), duration (D4), and indications (D5) are stored. Unlike the prescription, which is stored as a single JSON object, each medication attribute is stored in its own column. The table also stores attributes used for personalization such as display name and color.

Medication Entry Details

The Medication Entry table store data about each medication's administration entries. This is the table that stores the medication schedule. Each entry identifies the medication and a timestamp when that medication should be administered.

Status Details

The Status table stores data associated with the status of each medication entry. The status is an indication of the action that was performed by the user in response to a medication administration prompt. Four possible statuses were arrived at in consultation with a medical specialist. For all medication entries, one of the following statuses should be stored.

1. **Pending** - The *pending* status will indicate that the user has not updated the status of the medication. This will be the default status for all past actions.
2. **Taken** - The *taken* status will indicate that the user has confirmed taking the action.
3. **Missed** - The *missed* status will indicate that the user has confirmed missing the action.
4. **Unsure** - The *unsure* status will indicate that the user is not sure whether they took the action. This is usually the case when the user fails to remember actions taken when updating the statuses later.

Constraints Details

Data about restrictions are stored in the Constraints table. The table stores the identifiers of the medications involved in the restriction together with the time constraint and the nature of the restriction (that is apart or together).

6.3.4 Safety Checker

The Safety Checker performs two functions: (i) It generates schedules based on the prescription and user preferences, and (ii) it checks the consistency of the schedule refinements. The checker is always invoked by the Prescription module in two instances. First, every time the user is generating or refining a schedule, the information about the requested change is passed to the checker which generates the schedule based on supplied information and the information stored in the database about the affected medication. Second, when the Calendar module (client-side) is fetching the calendar entries. The checker receives the schedule to be sent to the client from the Prescription module. The checker validates the consistency of the schedule using the information stored in the database about the medications that form part of the schedule. The checker sends the schedule back to the Prescription module after adding drug interaction information to each entry in the schedule. The added information is used by the Interface module (client-side) to render conflicts in the calendar.

6.4 Interface Design

The Calendar Interface was built using Hypertext Markup Language (HTML), Cascading Style Sheets (CSS), and the FullCalendar Library [32]. Before settling on FullCalendar as the library to be used for the implementation of the prototype, we looked at alternative libraries such as Webix Scheduler [205], Tui Calendar [206], and Angular Calendar [207]. The selection criteria used were that the library should be open-source, extensible, have ample documentation, and be distributed freely. FullCalendar is the library that met all these requirements.

6.4.1 FullCalendar Library

FullCalendar is a JavaScript library that is used to create web-based calendars [32]. The library is based on JavaScript. It is also compatible with JavaScript frameworks such as React, Vue, and Angular. FullCalendar has two distributions; the Standard version, which is free (at the time of writing), and the Premium version, which requires payment (for additional features). For our implementation, we used version 6.1.0 of the standard edition. FullCalendar is open source. Hence, it comes with flexibility in terms of extending the functionality of the calendar. The library comes with event handlers and UI functions that can be overridden to change the appearance of the calendar and control underlying interaction mechanisms. The library handles the rendering of the user interface within a web page on which it is integrated. Figure 6.3 shows the default look of a calendar that uses the FullCalendar library.

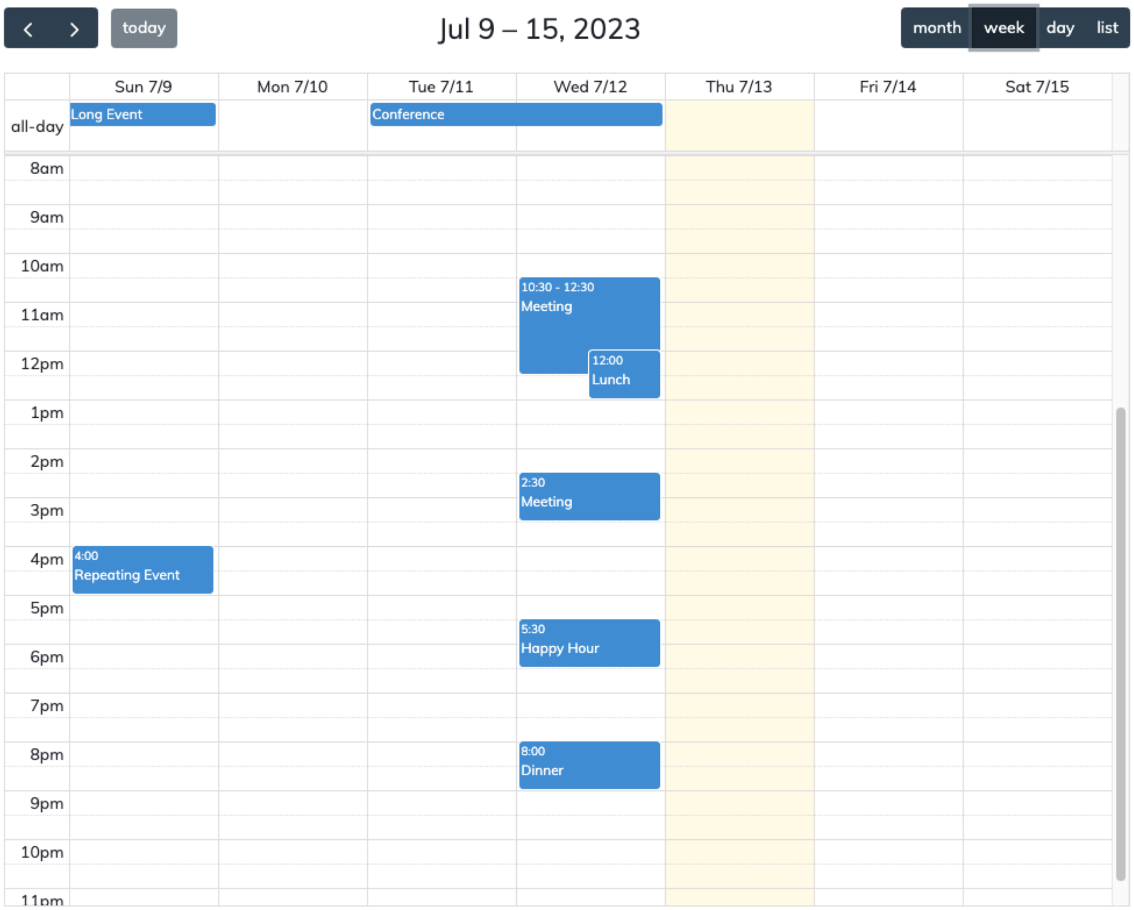
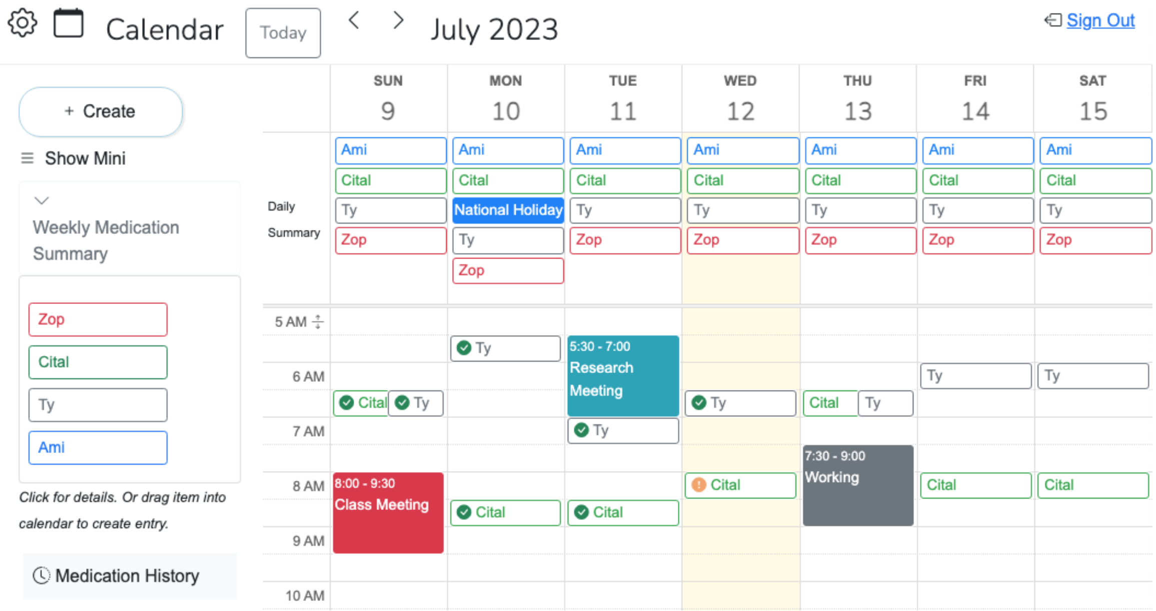


Figure 6.3: Default weekly view of a calendar using the FullCalendar library.

FullCalendar has four calendar layout options. These are monthly layout, weekly layout, daily layout, and list. The calendar has three navigation options; Next (which would change the view to the next unit of display), Previous (which would change the view to the previous unit of display), and Today (which would change the view to the display that shows the current day). The dates are shown on the top of the calendar and the time is on the left. By default, events in the calendar are rendered with a blue color fill and white text showing the event title and time of the event. The calendar supports two types of events; all-day events (which are rendered in the header section of the calendar - the all-day section) and fixed-time events (which appear inside the calendar). Event interactions such as dragging and clicking are also supported.

6.4.2 Calendar Prototype Interface

One of the design requirements we got from the study on exploring design variations for the calendar (presented in Chapter 5) was the *use of familiar design*. Although the calendar functionality was implemented as a proof of concept, the goal is to have prescription management supported by digital calendars. Based on that, we made changes to the appearance of the default FullCalendar system to make it look like popular calendars such as Google Calendar and Outlook Calendar [33, 183]. The following changes were made to the basic appearance of the calendar.



Click for details. Or drag item into calendar to create entry.

Figure 6.4: Personalization of a calendar designed using the FullCalendar library. The calendar shows entries for both ordinary events (with a color fill) and medication entries (with no color fill).

Calendar Layout

We adopted the weekly view of the calendar layout (See figure 6.4). This is because all the designs that were used in the design variations study utilized the weekly view. The rationale for this layout choice (as given earlier) hinged on the prevalence of weekly layouts in medication schedules.

Date and Time

The original date had the form “ddd M/D” with ddd being a three-letter abbreviation for weekdays and M and D being numerical values representing month and day respectively. This was changed to the form “‘ddd D” with both ddd and D representing the default units but presented on two rows. The display for Time was maintained with only minor changes, such as capitalization and spacing, in the format. The range of the visible time was shortened from [12 AM to 11:59 PM] to [5 AM to 11:59 PM]. This was done to reduce the amount of scrolling performed by the user. The adjustment also ensured that the default view seen by the user focused on the time when the calendar had activities. The calendar has controls to toggle between 5 AM and 12 AM. The label for the current week and year was changed to indicate the current month and year.

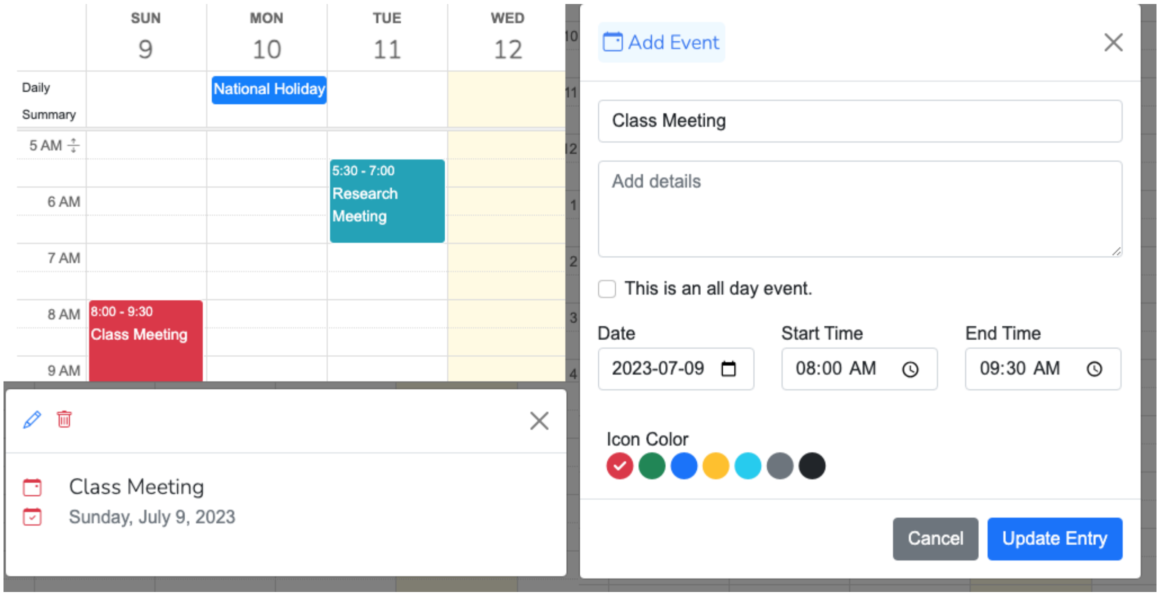
Navigators

Default navigators for today, the previous week, and the next week were overridden. The default navigators were attached to the calendar and could not be put outside a predefined placeholder (left, right, or center), thus limiting personalization. New navigators were introduced with the same functionality but different appearances (simulating Google Calendar and Outlook Calendar).

These are the changes that were made to the general appearance of the calendar. Changes made to the UI elements that related to calendar entries are presented in individual sections below.

6.4.3 Event Management

The display of events did not deviate from the provisions of the library. Events are displayed with only the time of the event and title. The user could personalize the event entries by changing their color. Event entry operations that are supported include adding, editing, and deleting. The user can add an event by using the “Create” button on the web page or by highlighting a selected time frame within the calendar. A dialog box would then appear prompting the user to supply event details such as title, details, date, and time. The event can also be classified as an all-day event, in which case it would appear under the daily summary section of the calendar. To change an event’s temporal attributes, the user can either drag the event to the required slot or click to bring up the details dialog box which has event edit functionality.



Class Meeting

Add details

☐ This is an all day event.

Date: 2023-07-09
Start Time: 08:00 AM
End Time: 09:30 AM

Class Meeting
Sunday, July 9, 2023

Figure 6.5: Event management features show the appearance of events in the calendar (top left), details view of an event (bottom left), and add/update event dialog (right).

6.4.4 Medication Management

Events in the calendar are native. That is, they are already supported by calendars. That is the reason why their default appearance, defined by the library, was maintained. It is the medication that is being introduced in the calendar. Thus, to be able to distinguish medication entries from event entries, a new design for medication entries was necessary.

Medication Entry Shape

For the shape of the medication entries, we adopted the design that was recommended by the user study presented in the previous chapter. The design is one which used only an outline for shapes representing medication entries. This meant that the design was either going to have a white or transparent fill. Since the calendar's background color is white, we opted to use white as the color fill for the medication entries. White maintained consistency in the display when a medication entry was overlapping with an event entry (if transparent, it would take on the color-fill of the event). Based on the feedback from the study, the side marker that was indicating the drug administration window was removed from the design as it was not successful in communicating the intended purpose. The dosage was also removed from the overview (it was made available when viewing entry details).

Medication Entry Status

Alongside the name, medication entries also have an icon that indicates the status of the entry. As indicated earlier, the status of an entry could be taken, missed, unsure, or pending. A similar type of icon, adopting a filled circle, was used for the statuses. The color fill used was differentiated for each status. As shown in figure 6.6, red was used for missed entries, blue for unsure entries, orange for pending entries, and green for taken entries. Each status also has a distinct mark within the circle. A cross-mark was used for missed entries, a question mark was used for unsure entries, an exclamation mark was used for pending entries, and a checkmark was used for taken entries. A semi-circle was used to indicate entries for modified dosages. These icons were visible for past entries, relative to the current date and time (future entries do not have a status).

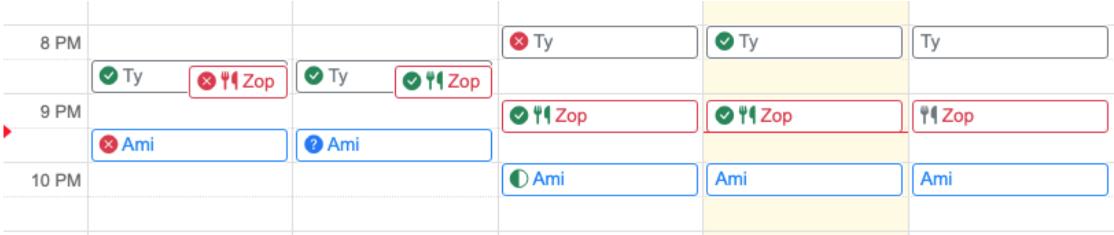


Figure 6.6: Medication entries with four types of statuses (Missed - red icon, Taken - green icon, Modified - green semi-circle icon, Pending - orange icon, Unsure - blue icon).

Medication entry statuses are supposed to be updated by the user. By default, all entries have the pending status. The user updates the status after administering medication. To update the status, a dialog box pops up after clicking a medication entry. The user then chooses an appropriate status to assign to that entry. The action of dragging a medication entry to a different time slot also influences the status of the moved entry. The following rules are applied to determine the status.

1. If the user moves an entry from one date to another when both dates are in the past, then the new status for that entry becomes taken. The act of moving an entry to a past date implies that the medication was taken at a different time.

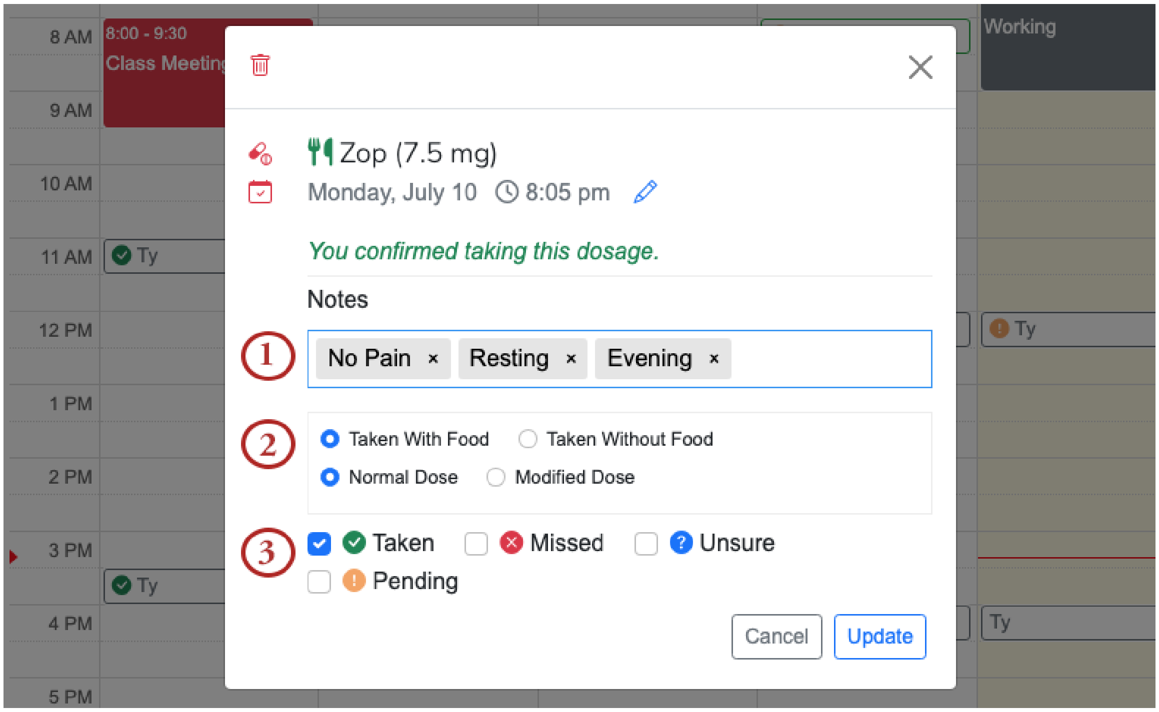
2. If the user moves an entry from a future date to a past date, relative to the current date and time, then the new status becomes taken. This movement implies that the entry, which is future, was taken earlier than planned.
3. If the user moves an entry from a past date to a future date, then the new status becomes pending. The action implies that has not yet been taken.
4. If the user moves an entry from one date to another when both dates are in the future, no action is taken. The user is just refining the schedule.

Taken With Food

We employed a cutlery icon to indicate that a given medication was supposed to be administered with food (See entries for Zop in figure 6.6). The icon appeared on entries for both past and future entries. As with statuses, color was used to indicate the state of the entries as regards food accompaniment. All taken-with-food icons for future entries are colored grey. For past entries, the color of the taken-with-food icon matched the color of the status icons for all statuses except the taken status. For example, if the status is missed, the icon for the taken-with-food icon is red. For the taken status, the user is supposed to indicate whether the medication was taken with food, in which case green is used for taken-with-food and red is used for taken-without-food.

Medication Entry Tagging

When updating the status of a medication entry, the user can choose to assign tags that can be referred to during self-reflection. This is achieved by supplying tags in the “Notes” section of the medication entry’s details dialog box. Tags are intended for the user to attach rationale, observations, and outcomes to a medication entry for purposes of providing context when revisiting the entries. Contextual information such as symptoms (e.g., headaches and tiredness) and measured biomarkers (such as heart rhythm and blood pressure) can also be added to an entry using tags. The tags can later be used individually or in collaborative prescription evaluation with physicians during appointments. The role of tags in self-reflection is explained in Chapter 7.



Zop (7.5 mg)

Monday, July 10 8:05 pm

You confirmed taking this dosage.

Notes

No Pain × Resting × Evening ×

● Taken With Food ○ Taken Without Food

● Normal Dose ○ Modified Dose

☑ Taken ☐ Missed ☐ Unsure

☐ Pending

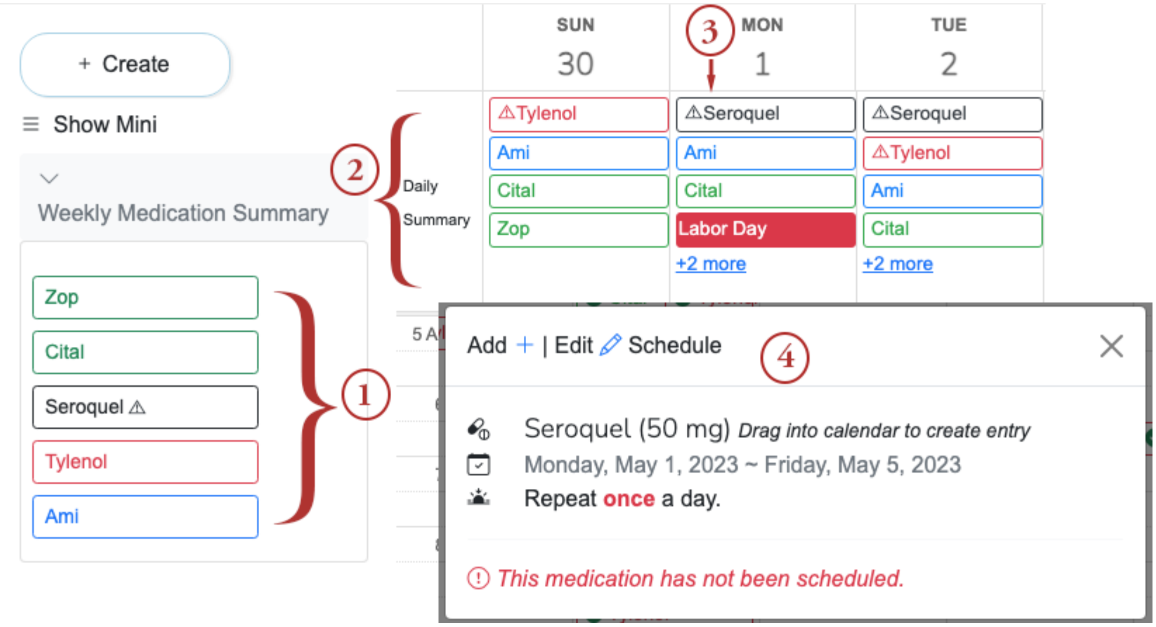
Figure 6.7: A dialog box showing options for (1) tagging a medication entry, (2) indicating whether the medication has been taken with food and whether the dosage is normal or modified, and (3) changing the status of an entry.

6.4.5 Updating Medication Entries

Medication entries can be updated in the same way that normal event entries are updated. They can be rescheduled by either dragging and dropping or editing their temporal properties (date and time) from the entry's details view. Other attributes that can be updated include statuses and notes (see figure 6.7). Medication entries can also be deleted from the calendar.

6.4.6 Medication Summaries

Medication summaries are a feature of the calendar that shows the list of medications for the user. The summaries show the medication together with an indication of whether a schedule has been generated for that medication. There are two types of summaries; daily summary and weekly summary (figure 6.8 (1) and (2)). The daily summary shows the list of medication that is supposed to be administered on a given day. This summary appears in the all-day section of the calendar. Medications for which a schedule is pending or those that have only been partially scheduled have an error icon preceding the name of the medication (figure 6.8 (3)). Clicking on the summary entry brings up an information box that gives more information about the warning icon. The weekly medication summary appears on the left sidebar of the calendar and shows a list of medication that is supposed to be administered in a selected week.



Seroquel (50 mg) *Drag into calendar to create entry*

Monday, May 1, 2023 ~ Friday, May 5, 2023

Repeat **once** a day.

This medication has not been scheduled.

Figure 6.8: Calendar design for (1) weekly medication summary, (2) daily medication summary (3) daily medication summary entry with unscheduled entries, and (4) an inset showing the medication details dialog box.

Viewing Medication Details

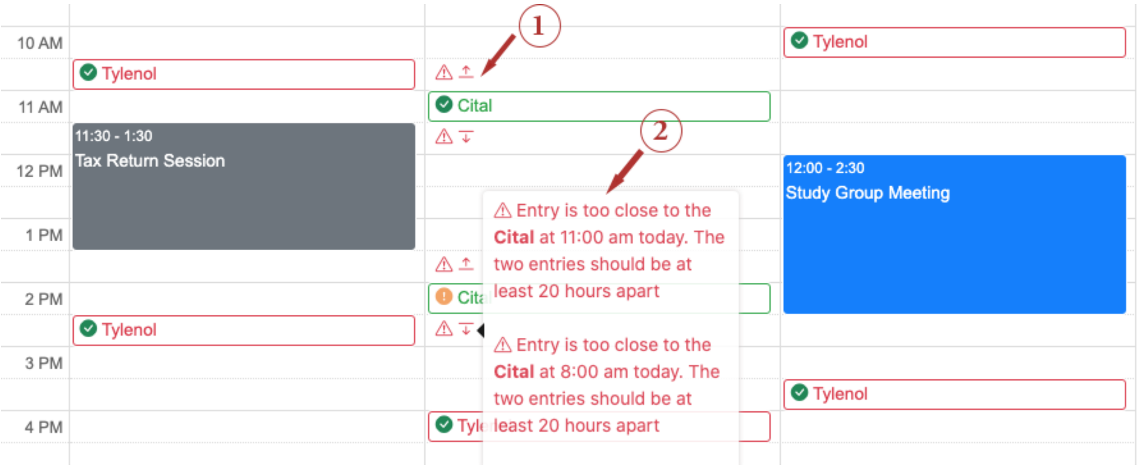
The daily and weekly summaries mirror each other in behavior for all interactions. Clicking on a daily summary or weekly summary entry reveals the details of the selected medication. The resultant dialog box reveals details about that medication such as name, dosage, duration, and repetition cycle. The dialog box also reveals whether the selected medication has entries that have not been scheduled (See figure 6.8 - (4)).

Updating Medication Details

The medication's details dialog box has the Add/Edit menu. Since the prescription is not authored by the user, deleting the prescription is not supported. The user can add a schedule for the selected medication by supplying default times based on the repeat cycles imposed by the prescription. The system would then generate the schedule for a specified period. Editing of the medication is restricted to attributes such as display name and color.

Scheduling Medication Entries

Both daily and weekly summary entries supported the drag-and-drop event. To create an individual entry, the user would drag the medication to the appropriate time point in the calendar.



⚠ Entry is too close to the **Cital** at 11:00 am today. The two entries should be at least 20 hours apart

⚠ Entry is too close to the **Cital** at 8:00 am today. The two entries should be at least 20 hours apart

Figure 6.9: Design showing conflict annotation. (1) Conflict warning with arrows suggesting the direction in which to move the entry to resolve the conflict. (2) Tooltip providing detailed information about the conflict.

6.4.7 Conflict Annotations

Conflicts between medication entries were signified by a red warning icon (triangle with exclamation inset, as shown in figure 6.9). The presence of the icon indicates that the entries conflict with one or more other entries. Next to the warning icon is an arrow suggesting the direction in which the entry should be moved to resolve the conflict. If an entry is involved in multiple conflicts, only one arrow is shown per direction. Arrows on both sides of the entry indicate that the entry is conflicting with entries on both sides. Hovering over the warning icon activates tooltips that provide more information about the conflict. When moving an entry to resolve a conflict, the calendar highlights the “unsafe zone” – a time block in which the conflict would still be active.

6.4.8 Medication Privacy Settings

During the user study of calendar design variations, privacy concerns were raised. The concerns formed part of the *Personalization* design consideration (DG4). The main concern was the exposure of the medication to unauthorized entities. Quoting a user, *“So I don’t know if it would be something [more] discrete or if someone was to accidentally see my calendar and [see] that I [am] taking all these medications ...”* We addressed this issue by adding medication privacy setting to the calendar. Using the settings, a user can hide all the medications from the calendar. When the prescriptions are hidden, no medication is loaded, and the calendar would operate like a normal events calendar. The user might also choose to only hide medication summaries or medication entries. This feature can protect the prescription from “unintentional” exposure. It could also be leveraged in determining the level of caregivers’ access to the prescriptions (In a collaborative setting, if the user has hidden the prescriptions, they are hidden for all users that have been granted access).

6.5 Self-Reflection

Supporting personal reflection (DG6) is another design consideration that came out of the study. Users wanted a way of letting them confirm that they had taken the medication to support personal reflection. I have already explained how we implemented the confirmation of medication-taking action (in the form of statuses). To aid self-reflection itself, we extended the calendar to have charts summarizing the user’s adherence over time. Self-reflection was not part of the original designs used in the study. Therefore, a process was carried out of determining the requirements for the integration of personal reflection in the calendar. The integration process, from design to implementation, is presented in Chapter 7.

6.6 Conclusion

This chapter presents the development of a web-based calendar prototype with built-in support for medication management. The prototype was created using HTML, JavaScript, PHP, MySQL, and the FullCalendar library. It supports both ordinary events and prescription entries. The prototype comprises five major components: two on the client side for managing the interface, user interactions, and client requests, and three on the server side for handling calendar data and responding to client requests. In addition to general medication management tasks, the calendar also includes features for self-reflection and safety checks to prevent unsafe drug interactions resulting from schedule refinements. Through this implementation, we tentatively addressed the primary research question of expanding the functionality of calendars to encompass prescription management. Chapter 8 will provide the methodology and results of a user study conducted to evaluate the calendar's effectiveness, confirming the successful implementation of these features.

Chapter 7

Designing for Personal Reflection

This chapter is based on a previous publication [208]. Hence, any use of the pronoun "we" in this chapter refers to Maybins Lengwe, Jens Weber, Charles Perin, and Morgan Price.

7.1 Introduction

The process of developing assistive technologies to support medication adherence covers a wide spectrum of design considerations. In chapter 5, I reported on five design considerations, derived from a user study, that informs the design of calendars that support medication management. These design considerations include; the use of familiar design (DG2 - as per numbering used in the preceding chapter), avoiding clutter (DG3), allowing for personalization (DG4), supporting personal reflection (DG5), and highlighting for user attention (DG6). Four of these design considerations (DG2, DG3, DG4, and DG6) address the issue of usability of the system. For example, DG3 is a requirement for a minimalistic design, one in which calendar elements occupy less space. This, like the other three, deals with the aesthetics and appearance of the calendar's User Interface (UI) elements. To address them, alterations can be made to the prototypes presented in Chapter 5 without changing the overall functionality and underlying framework of the system. DG5, on the other hand, introduces a whole new extension to the system. It involves the introduction of new UI elements to the calendar and the adjustment of the underlying framework to support the design guideline. As such, I use this chapter to consider processes and approached necessary for introducing the support for personal reflection in the calendar.

Reflection, according to Baumer et. al. [209], is the reviewing of a series of previous experiences, events, and stories and putting them together in such a way as to come to a better understanding or to gain some sort of insight. The process starts with users recording personal data for review later. The data is either entered manually using supporting utility tools or detected automatically using technologies such as smartphones, smartwatches, and smart pill boxes. The process of engagement in the self-tracking of personal data falls into phenomena such as Personal Informatics [210], Quantified Self [211], and Personal Reflection [209]. Designers and device manufacturers have become increasingly interested in the ways that technology might be able to support human reflection on past experiences [212]. More recently, health has become a focus for talking about self-reflection with emphasis on promoting healthy behavioural change, greater awareness and learning to self-manage chronic conditions [213, 214].

There are many reasons why people employ self-tracking. Some people want to set goals and achieve them with the help of self-reflection support systems[215]. Others want to document activities so that they can understand how different factors in their lives relate together [216]. Supported by these reasons, the incorporation of self-reflection into calendars has the potential to help patients better understand their medication-taking habits (or even challenge themselves to improve or maintain their adherence). Put in the words of one of our study participants, “The calendar should not just show which medication is supposed to be taken and when, it should also include a trace of statuses of past medication so that [users] may be able to reflect and optimize their calendars”. Although improved adherence is the targeted outcome, optimization is the driving force. Patients should be able to look at their medication history and identify patterns that may influence their adherence levels (whether positive or negative). They should then be able to optimize their medication and activity schedules so that adherence is maximized. The incorporation of personal reflection can also foster collaborative reflection between the patient and their care providers [217]. In this chapter, I answer the research question of how visualization can be used in the calendar to support self-reflection. To do so, I present design requirements for the integration of reflection views in the calendar. I also present the self-reflection visualizations that were designed as part of the calendar presented in Chapter 6. The visualizations were collaboratively developed with input from experts in Human-Computer Interaction (HCI) and Medicine. The effectiveness of these visualizations for promoting self-reflection was assessed as an integral component of the prototype, which is discussed in Chapter 6. The outcomes of this evaluation are detailed in Chapter 8.

The rest of the chapter is organized as follows. An overview of related work is presented in Section 7.2. I follow that up with a discussion of design requirements for medication reflection in Section 7.3. Section 7.4 presents the proposed high-fidelity design of the reflection views. I end the chapter with a discussion and conclusion in Section 7.6.

7.2 Related Work

Reflection has long been an accepted best practice in healthcare from the perspective of a care provider [218]. Unfortunately, current clinical information systems do not provide much support for reflective practices [219]. Janssen et al. refer to this missed opportunity as a blind spot that has been overlooked so far [219]. From a perspective of patient self-management, technology-supported tools for self-reflection exist and show promise in promoting healthy behaviour [214, 213]. Below, I present literature that is addresses self-reflection in healthcare.

Salib et al. presented a prototype study on the role of reflection on adherence to HIV medications [220]. The study piloted a prototype which they had developed, called TreatYourSelf [221]. The application allowed users to visualize their adherence recorded in weekly, monthly, and daily forms. No detailed explanation of the visualizations used for reflection is given. What the paper features is one visualization that presents a summarized calendar view of adherence. Their results indicated strong patient support for the usefulness of such a tool (80% of participants would use such a tool). Two qualitative findings were highlighted as a result of their study: (1) the need to provide contextual information surrounding the medication routine and (2) the need to easily evaluate the data collected [220]. A similar study, related to HIV, was conducted by Lauren et al. [222]. Their aim was to understand patients' and clinicians' perceptions of expected benefits and preferred uses of feedback to enhance conversations about adherence. They found that real-time feedback could facilitate collaborative adherence problem-solving, motivate patient adherence, and reinforce the importance of optimal adherence [222].

Lee et al. developed a sensing system that was made up of a sensor-augmented pillbox and an ambient display that provided real-time visual feedback about how well medications were being taken [1]. Their system used feedback after medication

Figure 7.1: An in-home tablet-based display, by Lee et. al. [1], showing feedback from the user's tracked data about medication taking, coffee making, and phone usage.

taking to allow the individual to develop routines through self-regulation. That is, unlike reminder systems which trigger alarms so that the user is reminded to take medication, no reminders were here generated. The idea was for the user to reflect on the routines, identify patterns, and consequently self-readjust. The feedback system used contained information such as the time that the medication were taken, an indication of whether the medication was taken on-time, and the time at which the pillbox doors were opened (See figure 7.1). The finding by Lee et al. was that feedback helped improve the consistency of medication taking behavior [1]. They also found that motivated individuals engage in self-regulation to adhere to their medication regimens.

That medication adherence is enhanced by digital health activity tracking is also supported by Quisel et al. who conducted an analysis of health status, behaviors, and medication adherence from health behavioral data. They found that users who adopt digital health activity trackers tend to be more adherent to medications and that adherence increases with tracking frequency [223].

Another importance contribution was made by Kocielnik et al. [2]. They presented a system called Reflection Companion. The system was a mobile conversational system that supported reflection on personal sensed data. Their area of application was physical activity. The data was collected using fitness trackers. Reflection Companion delivered mini-dialogues and graphs to users' mobile phones to promote reflection. The visualization used for reflection featured bar graphs that indicated summaries of activities. The system also supported text-dialog (See figure 7.2). Results from their

Hey Kate, take a look at your graph. Can you spot any weekly patterns in your data?

For one, I am less active on the weekends.

Please think a bit, what can this pattern tell you?

I need to be more active on the weekends. Even if I don't leave my house, I can still walk on the treadmill in the clubhouse.

Hey Jane, looking at your data. Which day(s) were you most physically active?

Saturday (long run while on vacation) and Tuesday (4th of July holiday).

Please take a moment, what was unique on these days and how can you repeat it?

LOL--more vacations and holidays! That dataset was not a typical week due to the holiday.

Figure 7.2: Extract from the system developed by Kocielnik et al. [2]. The system's mini-dialogues with visualizations are shown on the left. On the right is a block diagram of an example mini-dialogue.

study indicated that mini-dialogues were successful in triggering reflection and that the triggered reflection led to improved adoption of new behavior [2]. This contribution is important because it explicitly introduces charts that are not integrated into a calendar or scheduling tool. With the designs presented in chapter 5 already associated with clutter, a design outside of the calendar space has greater likelihood of adoption.

There are many other studies that have been conducted which seek to establish the relationship between self-reflection and increased adherence. The studies have spanned areas such as mental health [224], fertility care [225], multiple chronic conditions [226], physical activities [227, 228], sleep diaries [229], and self-defined activities [230]. These all show that self-reflection is effective in promoting adherence. And, as Hill et al. emphasized, real-time adherence feedback should be integrated into adherence support tools and also, guidance for feedback should focus on optimizing adherence and mitigating negative perception of adherence monitoring systems [222]. In the next section, I discuss guidelines that have been proposed on how to implement self-reflection. I also lay the groundwork for integrating reflection into the proposed calendar.

7.3 Design Requirements

Reflection, as defined earlier, is the reviewing of a series of previous experiences, events, and stories and putting them together in such a way as to come to a better understanding [209]. For prescriptions, the review process involves determining whether a given prescription was implemented according to its specification. The specification comes with data requirements that should guide on the implementation process. These requirements were presented in Section 4.3 of Chapter 4. They include Drug name (D1), Drug dose (D2), Drug dose frequency (D3), Duration (D4), and Drug indications (D5). During implementation of the prescription, data should be recorded on whether the administration schedule followed for any given medication honored these requirements. The recorded data should indicate whether (i) the full (or modified) dosage was administered (D3), (ii) the medication was taken according to the number of times specified (D3), (iii) the implementation period has been fulfilled (D4), and (iv) all contra-indications associated with the medication were honored (D5). The overarching indication is whether the medication was correctly administered. For this, there are four possible statuses; Pending, Taken, Missed, and Unsure. As presented in Chapter 6, here is a restatement of what each status indicates.

1. The *pending* status indicates that the user has not updated the status of the medication. This is the default status for all past actions.
2. The *taken* status indicates that the user has confirmed taking the action.
3. The *missed* status indicates that the user has confirmed missing the action.
4. The *unsure* status indicates that the user is not sure whether they took the action. This is usually the case when the user can not remember the action taken, during the process of updating the status.

Once the status and the temporal information associated with an entry have been known, we can design visualizations that can be used for self-reflection. To better inform the design, an understanding of the rationale behind self-reflection is key. The precursor to self-reflection is self-tracking. We used literature to come up with considerations for the design and development of the visualizations.

7.3.1 Supporting Reasons Why People self-track

There are diverse reasons as to why people keep track of their behavioral data. Fleck et al. suggest many such reasons including; reflection on the process of learning, critical review, and self-development [212]. People also self-track to be more aware of the state of their health or condition, monitor progress towards a preset goal, find associations in health events, take actions, or share information with healthcare providers [225, 226, 216]. The main themes behind all these reasons are self-awareness and self-improvement. People self-track because they want to look at the record and either commend themselves (for met goals), or challenge themselves to do better (for pending or missed goals). Self-tracking also provides a convenient way of sharing personal health data with healthcare providers.

Visualizations designed for reflection should be validated against these reasons. While the indicated reasons may be diverse (anyone can come up with a new reason altogether), the main aim for designing visualizations for reflect should be to present information that enables decision making. The system should therefore be designed with mechanisms for collecting as much medication-taking information as possible from the user, either through manual entry or automated means.

7.3.2 Supporting Different Stages of Self-tracking

Many approaches have been suggested on the different stages of self-tracking [225, 231, 232]. The proposed approaches are mostly extensions to the model proposed by Li et al. [210]. Li et al. suggested five stages that make up self-tracking. These are preparation (planning), collection (gathering of the data), integration (combining data from different sources), reflection (making sense of the data for decision making), and action (follow up actions based on the reflection) [210]. Mechanisms to support *reflection* is what this chapter is addressing. *Action* is a responsibility of the end-users and thus can not be addressed here (it's not a design consideration). *Integration* implies combining data from different sources (mainly sensors). This stage is also not addressed here as the source of data in this case will be a single calendar. The first two stages (*preparation* and *collection*) have to be supported, as a prerequisite, to set up a platform for the latter stages.

Supporting Preparation and Collection

Two extensions were made to the framework proposed in chapter 4 and the designs presented in Chapter 5. To the framework, the schedule generated by the safety checker was extended to include a medication status for each medication entry. When the schedule is exported to the data model from which the calendar loads its content, that presence of the status provides for an attribute that can be manipulated to reflect the four statuses indicated earlier. The designs were also changed to reflect the status of each medication entry in the calendar. Previously, all medication entry shapes in the three designs only indicated information about the medication. The shapes was modified or redesigned to show the status of any given medication. These modifications, once made, extended the prototype to include *Preparation* and *Collection* for self-tracking.

7.3.3 Supporting Different Levels of reflection

In addition to proposing reasons why people self-track, Fleck et al. also proposed five levels of reflection. These include *description* (revisiting of events), *reflective description* (revisiting events and trying to interpret or justify them), *dialogic reflection* (exploring relationships between events), *transformative reflection* (revisiting events with intent to re-organise or do something differently), and *critical reflection* (looking at the wider implication) [212]. These levels were numbered R_0, R_1, R_2, R_3, and R_4 respectively. I will maintained these numbers for reference purposes. The levels form a good basis to start designing visualizations for reflection. The designs presented in Section 7.4 will refer back to these levels for motivation. To support self-reflection, the design of the visualizations should support these different levels of reflection. While not al of them will be explicitly attached to the visualizations (and referred to in the discussion), the designs themselves should provide an encompassing view of the medication history, allowing the user to infer from the data more than originally intended.

7.3.4 Supporting Reflection Heuristics and Insights

The levels of reflection, presented by Fleck et al. [212], give an indication of the expected actions from the user. From the designer's perspective, it is also important to know how the data should be presented and what insights would be of interest to the user. Choe et al. suggest the type of visualization insights that should be supported for self-reflection [233]. These include *details*, *trend*, *comparisons*, *correlation*, *summary*, *distribution*, and *outliers*. These insights will also be revisited in the Section 7.4 to justify certain design choices. Three heuristics will guide the overall design. These heuristics are (i) that the designs should be easy to read, (ii) that patterns should be easy to recognize, (iii) and that trends will be easy to figure out [234].

7.4 Design Ideation

For the ideation stage, I worked on different visualizations that would support the discovery of some of the basic insights described above. Five visualizations were designed, covering a wide range of targeted insights. The design process for the visualizations is described below.

7.4.1 Overall adherence

Two visualizations were designed to show the overall adherence. The two designs were intended to show adherence in form of percentage. Showing adherence in this form would give the user a quick summary of adherence levels.

The first of the two ($Overall_1$ - figure 7.3) was a radial visualization that showed overall adherence using a clock hand that pointed to a percentage position on the gauge. The gauge used two colors to communicate user-set unsafe and safe levels. The color for positions on the gauge that were less that 80% was red, indicating in-adherent levels. The reading above 80% was green, indicating adherent levels. The visualization had radio buttons that could be used to switch between medications.

The second visualization ($Overall_2$ - figure 7.4) showed adherence levels on a linear scale. The scale showed percentage points from 0 to 100. Below the scale was a list of medications. Each medication has a bar that represented adherence levels. This visualization used the same demarcation and coloring of in/adherent levels as $Overall_1$. Unlike $Overall_1$ where only one medication level could be read at any given time, $Overall_2$ was designed to show the adherence levels for all medications at the same time, side by side.

By using these two designs, the user would have access to insights such as adherence *summary* per medication and *comparison* between adherence levels for different medications. $Overall_1$ and $Overall_2$ fall in the level of *reflective* reflection. By using

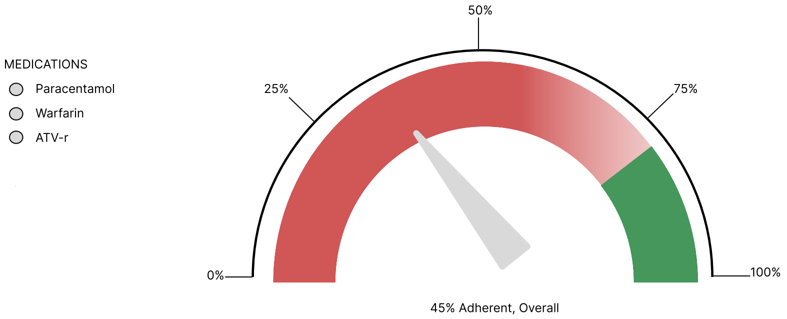


Figure 7.3: A visualization showing user adherence level (as a percentage) for selected medications. The red and green parts of the gauge represents lower and higher levels of adherence, respectively.

these visualizations over a selected period, the user can derive interpretation of overall adherence rates.

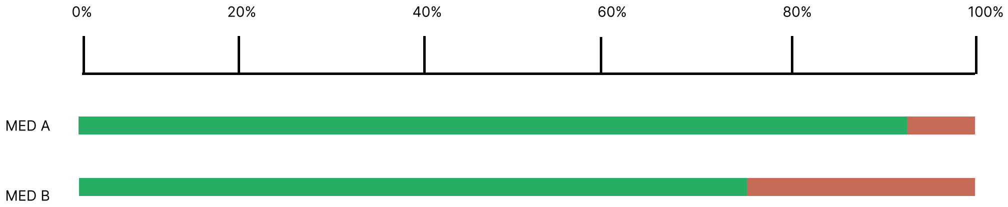


Figure 7.4: A visualization showing adherence levels (as a percentage) for all medications. The green fill in the adherence bar represents medication marked as taken. Red represents the percentage of missed medication.

7.4.2 Adherence pattern

Three visualizations were designed to show adherence pattern over an interval. The three designs were intended to communicate details associated with entries in each medication.

The first visualization ($Pattern_1$ - figure 7.5) was designed to communicate the dosage per given medication and an indication of whether that dosage was administered. The design used a capsule icon to represent a dosage. Each medication occupied a row of capsules. The number of entries (in a row) between two consecutive dates represented the number of times that a given medication was supposed to be administered. Green fill in the lower part of the capsule indicated that a dose was administered. Red indicated that a dosage was missed. The design also featured a larger horizontal capsule that indicated overall adherence.

The second visualization ($Pattern_2$ - figure 7.6) was designed to communicate adherence by showing dosages on a timeline that reflected the calendar view. Each dosage was represented by a tablet icon whose color communicated the administration status of the dosage. The visualization maintained the temporal position of each dosage. A horizontal bar was added outside the timeline to show overall adherence.

The third visualization ($Pattern_3$ - figure 7.7) was designed to communicate adherence using a calendar view. Unlike the second visualization, the temporal data associated with each dosage was here abstracted. Instead, each medication had stacked bars for each day of the calendar. The number of divisions in the bar represented the number of medication administrations per given medication. For example, *Med X* in the visualization is taken three times a day whereas *Med Y* is taken once a day. The ordering of the dosages was also maintained with the top of the stack representing the beginning of the day. Each entry in the stack was colored either green (for taken) or red (for missed).

A number of insights were expected to be derived from these three designs. First, in additional to insights related to overall adherence(*summary* and *comparison*) , the user could deduce if a certain period of time had an effect on adherence (*trend*, *correlation*, and *outliers*). This could be achieved using any one of the three designs. Second, if there was a relationship between adherence and the period of the day when medication is taken, the second and third designs could reveal that. Third, the third visualization could be used to deduce if the number of dosages affect adherence (*distribution*). Table 7.1 summarizes the insights and levels of reflection supported by these visualizations.

To validate the designs, a series of ideation sessions were held in which two other researchers refined the visualization. The last ideation session that was held in consultation with a domain expert. After these sessions, we arrived at three visualizations, each supporting multiple insights and addressing selected levels of reflection. The final visualizations are presented below.

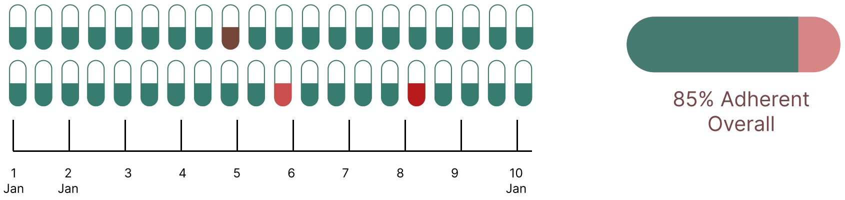


Figure 7.5: A visualization showing medication dosages as capsules colored either green (for taken) or red (for missed). Each row represents a separate medication.

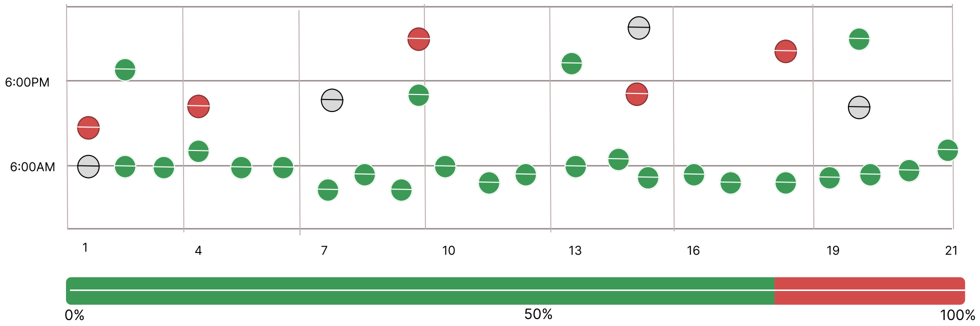


Figure 7.6: A visualization design showing dosages on a timeline. Each tablet glyph represents a dosage with green implying that the dosage was administered and red implying that the dosage was missed.

7.5 Reflection Visualizations

Following up on the ideation stage, we designed three visualizations to help the user self-reflect on their medication adherence record. The three visualizations were (i) Overview (designed to show overall adherence), (ii) Weekly Summary (designed to show adherence levels summarized per week), and (iii) Timeline View (designed to show adherence by timeline). These visualizations summarized the three designs coming from the ideation sessions by removing design redundancies (such as the overall indicator on multiple designs from the sessions) and aligning more clearly with insights and levels of reflections. From the design perspective, the visualizations were intended to support three basic tasks: (i) reading over adherence levels, (ii)

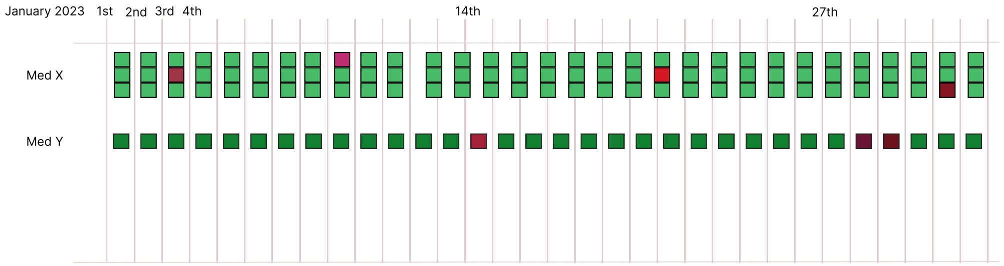


Figure 7.7: A visualization design showing the list of medication on a timeline. Each medication has a row of stacked bars where each entry in the stacked bar represents a dosage.

Table 7.1: Visualization Design - Levels of Reflection and Types of Insights.

	Visualizations				
Levels of Reflection	$Overall_1$	$Overall_2$	$Pattern_1$	$Pattern_2$	$Pattern_3$
Description				✓	✓
Reflective	✓	✓	✓	✓	✓
Dialogic			✓	✓	✓
Transformative				✓	
Critical					
Types of Insights					
Details			✓	✓	✓
Trend			✓	✓	✓
Comparison	✓	✓	✓	✓	✓
Correlation				✓	✓
Summary	✓	✓		✓	
Distribution		✓			
Outliers				✓	✓

identifying extreme values in the calendar, and (iii) characterizing the distribution of medication statuses. The three visualizations (referred to as *Overview*, *Weekly*, and *Timeline*) are discussed below.

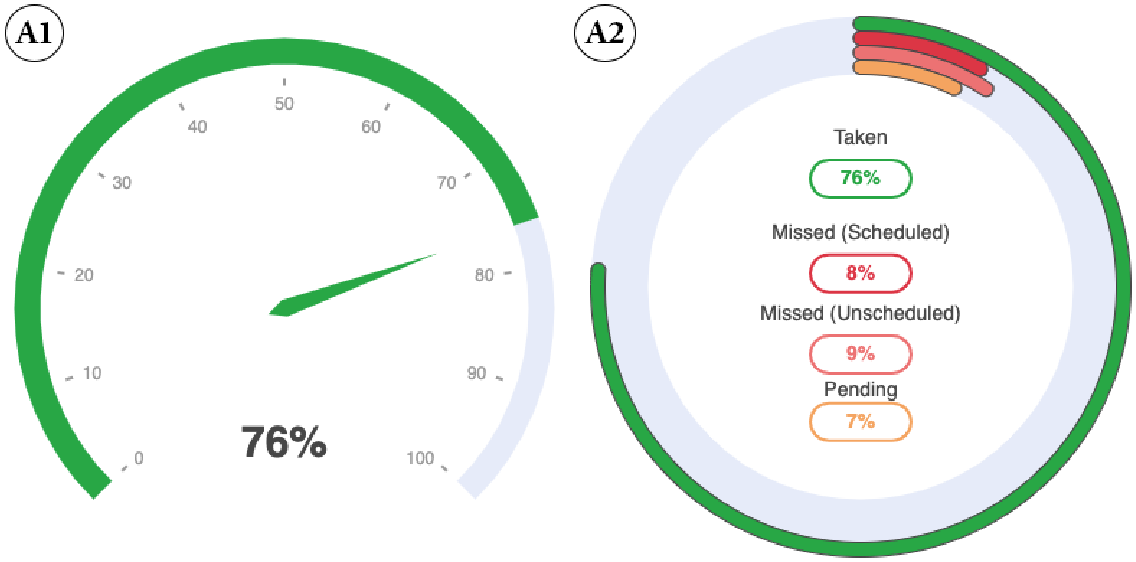


Figure 7.8: *Overview* Visualization showing the overall adherence for all medications over a selected period of time. A1 shows the overall rate while A2 shows the distribution of the statuses

Overview

The *Overview* visualization was designed to show the overall adherence rate by the user. The visualization had two levels of medication action details (presented in separate charts). The first one ($Overview_1$ - shown as *A1* in figure 7.8) visualizes overall adherence for all medications in the calendar. The adherence level are indicated by a dial within the radial chart. The chart has a refinement option which allows the user to view the adherence rate per medication. Using $Overview_1$, the user can see the rate presented as a percentage over a selected period of time. The adherence rate is calculated a ratio of medication entries with a taken status to all entries of that medication in the selected period. The user can select a different period to make comparisons either between entries of the same medication or all medications.

The second level of details ($Overview_2$ - shown as *A2* in figure 7.8) visualizes the breakdown of the medication statuses for the selected medication and period. Using this setting, the user can see the distribution of the medication that was taken relative to the ones that were either missed, pending, unsure, or unscheduled (in the event that a medication that was supposed to be scheduled by the user had no calendar entries, all missing entries for such a medication are considered unscheduled).

Insights The *Overview* visualization was designed to address the *description* level of reflection. The visualization offers a brief overview of the history without showing medication entry details necessary for further interpretation. Using this visualization, the user can derive insights by probing the visualization with questions such as the following.

1. What is my overall adherence (*Summary*)?
2. How does my overall adherence compare between selected time intervals (*Comparison*)?
3. In which medication is my adherence level highest or lowest (*Comparison*)?
4. What is the distribution of medication statuses per selected medication (*Comparison, Distribution*)?

The main goal for the *Overview* Visualization is to show the overall adherence and provide a quick way of comparing adherence between different medications. $Overview_1$ gives this quick access. The rationale for $Overview_2$ was to provide context for the summary provided in $Overview_1$. $Overview_2$ reveals situations where overall adherence reported is affected by unscheduled or pending entries. By default, the visualization is set to show $Overview_1$. The toggle option is provided for the user to switch views.

Weekly Summary

The *Weekly* visualization shows adherence consolidated based on weekdays (See figure 7.9). A summary of the adherence is presented by showing the overall adherence for each day of the week. This provides for a quicker way to compare adherence between weekdays and cover an extended period of time (The Timeline View has period of time as a limitation). The visualization uses a stacked bar-chart to show the weekly summary. Each bar shows the distribution of medication taken (green), missed (dark red for scheduled and bright red for unscheduled), pending (orange), and unsure (blue). The user can also change view to a normal bar chart where each status is shown on a separate bar (See figure 7.9). Refinement the visualization display also exist using both the medication list and the status list (that is, showing one medication and/or on status at a time - as shown in figure 7.9). A word-cloud is part of this visualization. The word-cloud gives the user access to motives or outcomes

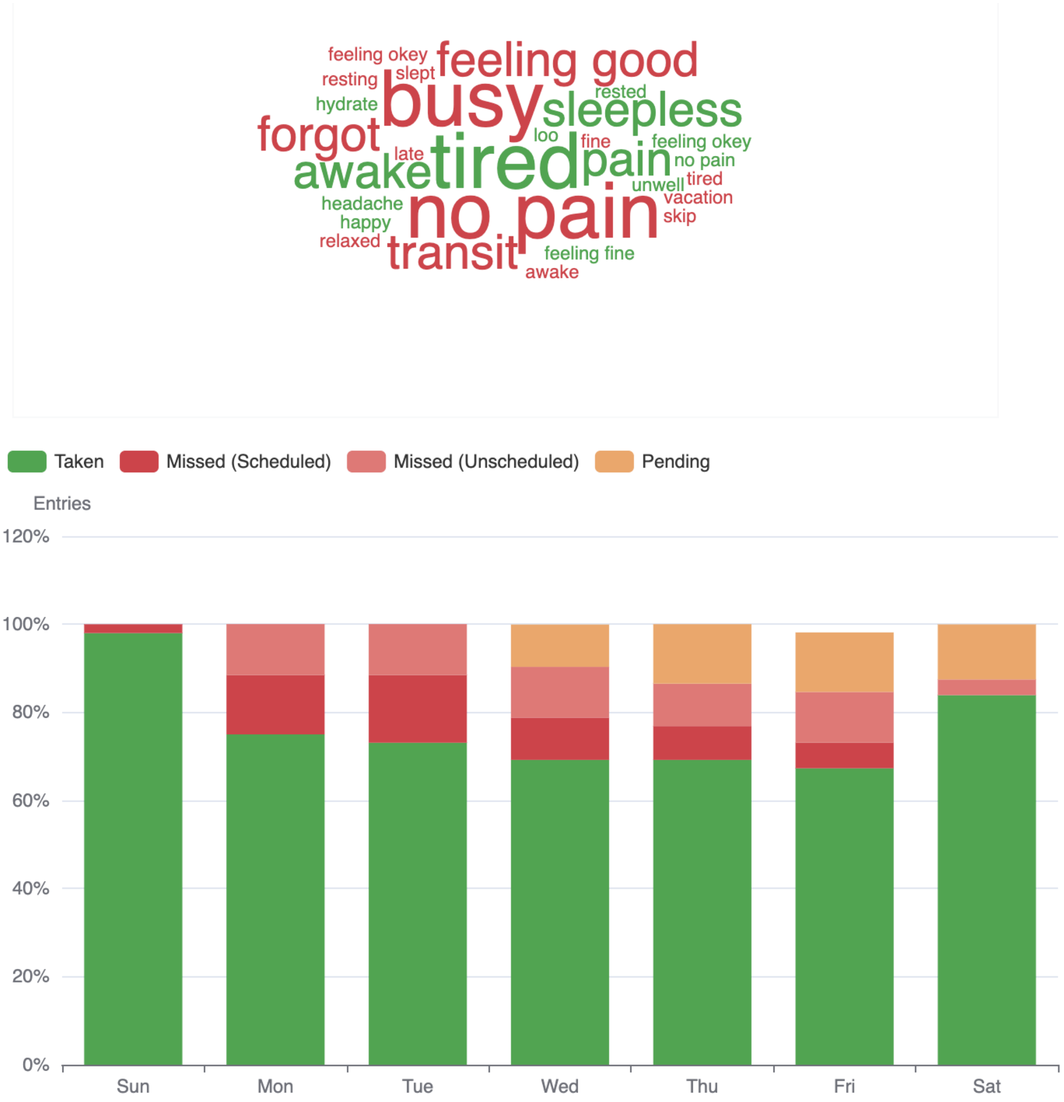


Figure 7.9: A visualization showing adherence rate by weekdays. This particular visualization would be interpreted as having a higher rate on weekends than on other days of the week. A word cloud helps the user to reason about causes and outcomes.

(hashtags) associated with the presented summary. The word-cloud is generated from the tags which the user supplies when updating the status of a medication entry. Using color hue (for whether the status is for a missed or taken medication) and size (for how many times a certain tag is used), the user can reason about what influenced an observed change.

Insights The *Weekly* visualization was designed to address reflection levels R_1 to R_4. The visualization offers a weekly summary view that is explicable ($reflective$).

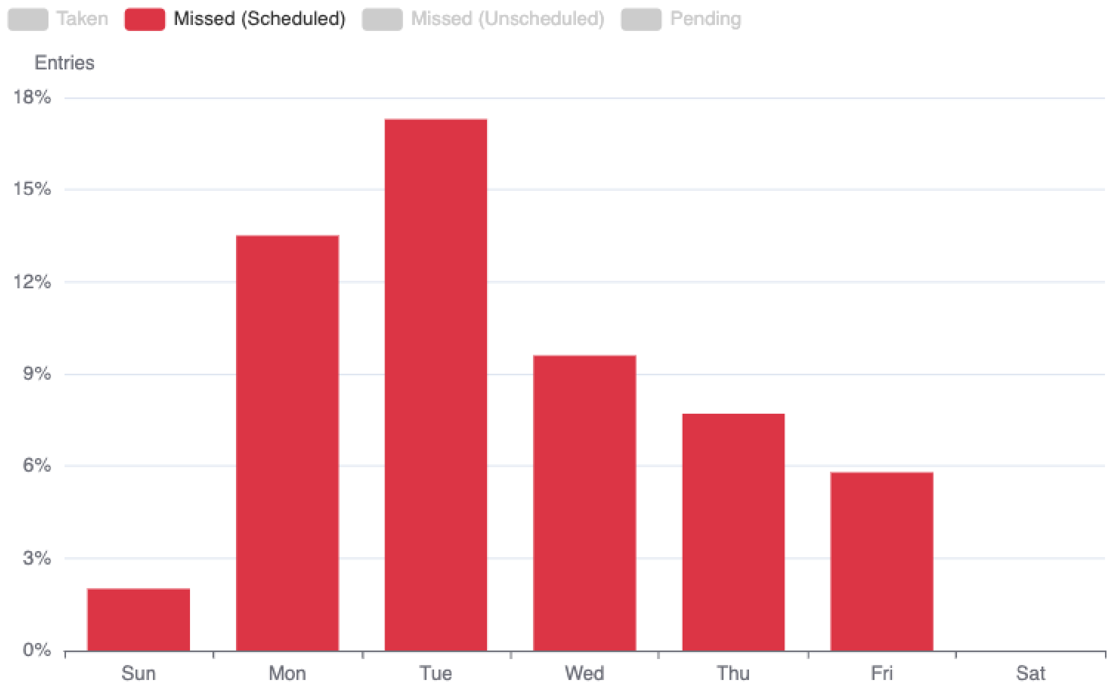


Figure 7.10: A visualization showing adherence rate by weekdays. This visualization is a refinement of figure 7.9 showing only missed actions.

The summarization of adherence according to behavior on weekdays offer the possibility of interpretation based on temporal relationships (*dialogic*). The *Weekly* visualization also includes context information in form of tags. By using the chart, together with the word-cloud, the user can revisit past events and try to justify them (*transformative* and *critical*). Using the Weekly Summary, the user can derive insights by probing the visualization with questions such as the following.

1. On which days of the week am I more (or less) adherent overall (*Trend, Summary, Comparison*)?
2. On which days of the week am I more (or less) adherent per selected medication (*Trend, Summary, Comparison*)?
3. How do the weekly adherence levels compare between different intervals (*Trend, Comparison*)?
4. What do the recorded tags say about what affected my adherence per selected period (*Details, Trend, Correlation*)?

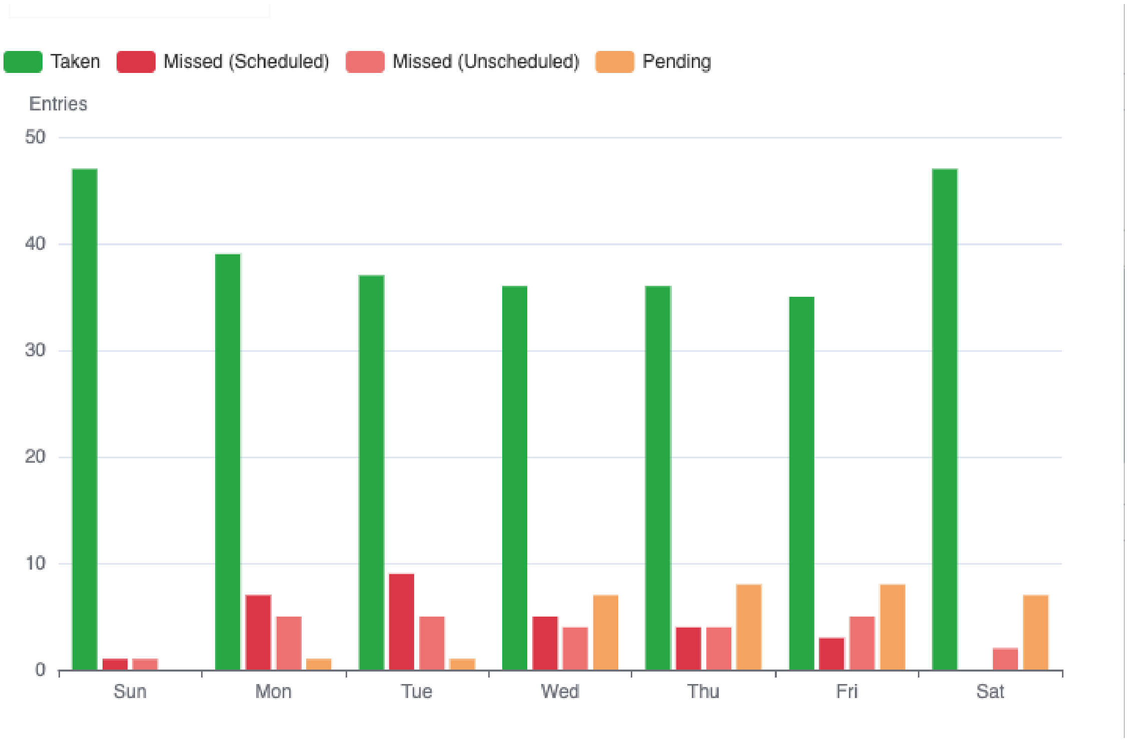


Figure 7.11: An alternative weekly view showing the statuses as separate bars. The bars also use dosage count as the unit instead of the percentage used in figure 7.9

The main goal for the *Weekly* Summary Visualization is to show the correlation between adherence and the days of the week. The rationale is that Weekdays in calendars often carry repeated duties. By design, we assumed that a revelation of days where adherence rates are lower, coupled with context information coming from the tags, might compel the user to consider realigning the medication to better fit into such days.

Timeline View

The *Timeline* visualization showed medication-taking summaries in a timeline that reflected the calendar (See figure 7.12). The timeline is based on the user-selected interval from the calendar. In the visualization, all medication entries are shown at the exact time at which they were supposed to be taken. Shape (randomly assigned to medication at loading time) and color hue (with green for taken and red for missed, as in all the visualizations) are used to differentiate between medications and medication statuses respectively. Each entry in the timeline supported the hover interaction, an action that revealed the name, status, and all tags associated with that medication

Table 7.2: Visualization Implementation - Levels of Reflection and Types of Insights.

	Visualization Types		
Levels of Reflection	*Overview*	*Weekly View*	*Timeline View*
Description			✓
Reflective	✓	✓	✓
Dialogic		✓	✓
Transformative		✓	✓
Critical		✓	
Types of Insights			
Details		✓	✓
Trend		✓	✓
Comparison	✓	✓	✓
Correlation			✓
Summary	✓	✓	
Distribution	✓		✓
Outliers			✓

entry.

Insights The *Timeline* visualization was added primary to support *description* in additional to the other levels of reflection. The visualization features a trace of medication administration action, allowing the user to revisit all events at the point of occurrence (*description*). For each action, the status and tags associated with the action are also visualized. The visualization maintains a calendar view that allows for exploration of relationships between entries (*dialogic*) and provides an opportunity for reorganization by analyzing trend and associated tags (*transformative*). Using the Timeline View, the user can derive insights by probing the visualization with questions such as the following.

1. How is my adherence be selected period of time (*Summary, Trend, Comparison*)?
2. What relationship is there between the time of day and the level of adherence for a selected period of time(*Correlation, Distribution, Outliers*)?
3. What temporal relationships do the associated tags reveal about selected missed actions (*Trend, Correlation*)?

The main goal of Timeline View is to give a detailed overview of medication-taking behavior without abstracting the temporal information of medication entries in the calendar. Using this visualization, the user can identify patterns based on dates and times of the day. For example, according to figure 7.12, adherence rates are higher in the morning as compared to evenings. Timeline view is designed to be reasoned with along such lines.

Medication: Tylenol
Status: missed
Date: 2023-04-17 22:00

Figure 7.12: This visualization shows the calendar trace between two selected dates. Each entry in the calendar is positioned at its original calendar position. For each entry, the color indicates whether it was taken or not. On hover, a popup message appears showing the name, status, date, and associated tags.

Table 7.2 summarizes the insights and levels of reflection supported by the three visualizations. These three designs were implemented in JavaScript. The visualization were extensions of example visualization by Apache eCharts[202]. The charts are accessible directly from the calendar page. The evaluation of the charts is presented as part of the evaluation of the entire calendar in Chapter 6.

7.6 Conclusion

Self reflection has proven to increase engagement with adherence support tool. We conducted a study in which we sought to come up with design guidelines for integrating medication management in digital calendars. Supporting personal reflection is one of the design guidelines that came out from the study. The original design of the calendar did not have provision for personal reflection and needed an extension to support the feature. This chapter introduces a part of the calendar that deals with self-reflection. I began by exploring literature that is related to self-reflection, personal informatics, and quantified self. I proceeded by discussing design considerations for a system that allows users to reflect about their implementation of prescriptions. The considerations included reasons why people self-reflect, levels of reflection, and general insights that should be targeted by reflection systems. I then presented the ideation process that I followed to arrive at three visualizations that would give users different types of insights about their medication histories. The proposed visualizations were intended to support basic tasks such as reading over adherence levels, identifying extreme values in the medication calendar, and characterizing the distribution of medication statuses. A study was conducted to validate the calendar and also learn about user perception of visualizations for self-reflection. The results of the study are presented as part of Calendar Evaluation in Chapter 8.

Chapter 8

Calendar Evaluation

8.1 Introduction

Digital calendars play a crucial role in managing everyday activities [187, 24]. Many digital calendars available today primarily focus on event management [33, 183, 201]. Some calendars, like Google Calendar, offer extended functionality by incorporating entities such as tasks and appointments [33]. This extension of functionality, allowing users to choose the type of entry they want to add to their calendars, opens up the potential for calendars to be utilized as support tools in the healthcare industry. Recognizing this design provision as an exploitable space, we proposed enhancing the calendar's functionality by integrating an aspect of health management into it. To this end, we designed a calendar that supports medication management. The proposed calendar supports (i) scheduling of medication, (ii) handling of medication scheduling conflicts, and (iii) self-reflection. The design and implementation of the calendar prototype have already been discussed in Chapter 6. The design and implementation of the calendar's self-reflection feature have also been discussed in Chapter 7. In this chapter, we present the methodology and results of the study conducted to evaluate the developed prototype.

To evaluate the prototype, we conducted a usability study with twelve participants. During the study, participants were provided with a calendar reflecting the activity cycle of a persona developed in collaboration with a medical expert. They were then asked to role-play and perform tasks covering four usability requirements: event management, medication management, conflict management, and self-reflection. We recorded the correctness and efficiency of task performance, as well as user sentiments for selected tasks. The recorded data was analyzed both quantitatively using Microsoft Excel (for task completion metrics) and qualitatively using NVivo (for user sentiments) [235, 197].

The results indicate that medications can be managed using digital calendars with the same level of competency required for managing calendar events. Discovering calendar features related to medication took slightly longer than discovering features related to time and events. Participants leveraged their familiarity with calendars to complete most of the tasks. Some tasks specific to medication management, such as scheduling medication and adding medication entries to the calendar, took longer than event-related activities during the first attempt but equalized during the second attempt. This leveling up put medication management at par with event management in terms of completion time. Participants were also able to manage conflicts related to unsafe refinement of medication schedules with ease. Self-reflection received mixed reviews, with self-tracked calendar statuses attracting more user engagement. Participants were able to reflect and derive insights using both the calendar and charts. Overall, participants found managing medication using a calendar useful and something that can challenge them to improve their adherence.

The rest of the document is structured as follows: Usability requirements are presented in Section 8.2. The study methodology and results are presented in Section 8.3 and Section 8.4, respectively. I conclude the Chapter with a discussion of the results in Section 8.5 and a conclusion in Section 8.6.

8.2 Usability Requirements

In order to conduct a usability study of the calendar prototype, there was a need to establish the usability requirements against which the prototype would be evaluated. And, as already determined in Chapter 5, a calendar that supports medication entries should support tasks that are associated with the management of both events and medication entries. These tasks include general calendar activities such as adding, modifying, and deleting calendar entries [33, 183]. The tasks also include medication-specific tasks that relate to handling conflicts arising from unsafe schedule refinement [145]. Adding medication management to calendars also introduces tasks related to using the calendar for self-reflection. To support these tasks, certain usability requirements should be satisfied by the calendar. During the design stage, we proposed three usability requirements spanning Calendar Layout (R1), Medication Entries (R2), and

Medication Conflict (R3) (See Chapter 5). To study the usability of the calendar implementation, we modified the three usability requirements to accommodate dynamic user interactions. We also added new usability requirements that addressed the extended functionalities brought in by the introduction of Self-Reflection. The requirements are discussed below.

8.2.1 Calendar Features

The calendar's event and medication management features should be discoverable with minimal effort. This requirement is a unification of Calendar Layout (R1) and Medication Entries (R2) requirements already proposed in previous chapters. R1 addresses the users' ability to correctly, and efficiently, read the calendar's temporal features. R2 addresses the users' ability to accurately, and efficiently, identify the calendar's medication entries (See Chapter 5, Section 5.3.2). These two requirements were merged into one as they both related to *reading*, or discovering, the calendar's features. In this respect, the calendar should satisfy the following usability requirements.

R_A -*The user should be able to correctly, and efficiently, read the calendar's event and medication related features. These features include the calendar's event and medication entries. Other features include temporal attributes such as current year and month; days of the week; times of the day; current day; and current time. The user should also be able to identify Calendar navigators; event creation buttons, medication icons, medication summaries, medication history, and medication privacy settings.*

8.2.2 Manage Calendar Entries

The calendar should support tasks related to the handling of calendar events. These tasks include adding, editing, and deleting events from the calendar. The calendar should also support tasks related to the handling of medication entries. These tasks include adding, editing, deleting, and personalizing medication entries. To evaluate these, we added a new requirement. The following usability requirements should be satisfied by the calendar.

R_B -*The user should be able to efficiently add, edit, delete, and personalize both routine events and medication entries within the calendar.*

8.2.3 Medication Conflicts

The calendar should support easy handling of medication conflicts. This requirement was derived from the Medication Conflicts (R3) requirement. R3 addressed the users' ability to correctly identify conflicting medication entries in the calendar. The requirement was extended to include the actual task of resolving conflicts as opposed to only suggesting possible actions that might be taken to resolve the conflict. The calendar should now satisfy the following usability requirements.

R_C *The users should be able to correctly identify conflicting medication entries in the calendar, know the nature of the conflict, and resolve the conflict. The system should also be fast enough to support efficient rendering of conflicts.*

8.2.4 Understanding Medication History

The calendar should support self-reflection. Self-reflection was not part of the study on the exploration of design variations (Chapter 5). This requirement has been added in response to the results of the previous study, which dictated the need for self-tracking of medication taken by the users. Three visualizations were added to the calendar to show different insights based on the medication history of the user (See Chapter 7.4). By using three types of visualizations, together with self-tracking data shown in the calendar, the users can derive insights about their medication-taking behavior. The calendar should therefore satisfy the following usability requirements.

R_D *The users should be able to read the visualizations and summarize the adherence levels as presented in the charts. The users should also be able to identify relevant insights from their personal histories.*

8.3 Study Methodology

We conducted a user study to evaluate the calendar prototype and get users' perspectives on different components of the calendar. The study was held remotely using the video conferencing platform Zoom [236]. Prior to the study, we hosted the calendar on a publicly accessible web hosting service. The calendar featured a trace of activities for a persona, developed by a medical expert, called Jordan. Jordan was taking five medications. These included Tylenol (15mg taken four times a day, as needed), Zopiclone (7.5mg taken once a day, with food), Citalopram (30mg taken once a day), Seroquel (50mg taken once a day), and Amitriptyline (75mg taken once a day). Jordan also had a schedule that included activities such as Classes, Part-time Work, Research Meetings, Study Sessions, Clinic Visits, Shopping Activities, and Hiking. We populated the calendar with 6 months of simulated events and medication-taking actions.

A moderated study session was then conducted based on Jordan's calendar. During the study, we presented the participants with two weeks of Jordan's history. The two weeks were presented to the participants in the same order. We asked the participants to perform a set of tasks that reflected everyday calendar usage tasks such as reading, adding, editing, and deleting calendar entries. Participants' activity cycles were recorded through video and activity logs. The participants were also given an opportunity to suggest changes that might be made to the calendar in order to better suit their medication management routines. The study was approved by the Institutional Ethics Board at the University of Victoria.

8.3.1 Participants

We made a call for participation through different platforms. These included Social Media platforms (including Facebook, Slack, and WhatsApp), university mailing lists, and posters. The recruitment was targeted at people who were between the ages of 19 and 65 years old, were taking (or helping someone taking) medication regularly, considered themselves to have a reasonably busy schedule, and used digital calendars to manage their daily routines. Participants were also recruited through Snowball sampling. Those who were interested reached out to the researcher and were recruited on a first-come-first-serve basis. Participants were compensated with a CAD20 Amazon e-gift card. From the call for participation, twelve participants (P) were recruited, and all the recruited participants completed the study (Female = 7, Male = 5). The participants were aged between 21 and 40 years old (average age was 31). Of the twelve participants, six were on medication, one was both on medication and helping another person on medication, and five were only helping another person on medication. The group comprised students (n = 8), nurses (n = 1), Human-Computer Interaction (HCI) experts (n = 1), Librarians (n = 1), and Lecturers (n = 1). The frequency with which participants used a calendar to manage their schedule was daily (n = 9), often (n = 2) and not often (n = 1).

8.3.2 Study Procedure

The duration of each study was one hour. The study began with participants being granted access to the calendar which was hosted on a public server. Participants shared their screens to show the calendar. The participants were then introduced to the Persona. They were encouraged to think through the Persona's position. Participants were given four sets of tasks. These tasks were meant to test usability requirements (R_A - R_D) for the calendar. They started by performing tasks in R_A. There were altogether 14 tasks in this category. The tasks involved discovering various features of the calendar including temporal features, event and medication management features, medication privacy, and medication history.

Tasks in R_A were followed up with tasks in R_B. There were 14 tasks in R_B. These tasks involved activities dealing with the management of both events and medication entries. After R_B, they went on to perform 4 tasks that comprised R_C. Tasks in R_C centered around being able to identify and fix warnings resulting from unsafe schedule refinements. Participants then performed a total of 15 tasks in R_D. There were only 6 unique tasks in R_D. Some of the tasks were repeated under different conditions (such as different charts), leading to a total of 15 tasks.

The tasks in R_D were grouped into 4 sets. The first set comprised 5 tasks which were intended to test self-reflection using the status data presented in the calendar view. Each of the remaining three sets had a different number of tasks intended to test self-reflection using the personal history visualizations. The tasks for the three sets were distribution as follows: There were 4 tasks performed for the *Overall Chart*, 4 tasks performed for the *Weekly Chart*, and 2 tasks performed for the *Timeline Chart*. The participants were instructed to rank the three visualizations in order of preference, starting with the one they considered the easiest to read for self-reflection.

The last set of tasks was a repeat of tasks in R_B. Participants were asked to repeat tasks in R_B using a different calendar week. The repetition was done to test the efficiency of doing the tasks after a prior exposure.

Participants performed a total of 38 unique tasks (see table 8.1). Tasks in R_A, R_B, and R_C were delivered to the user via prompts embedded into the calendar. Tasks in R_D were read out to the user. Following these tasks, participants were asked to offer general feedback on the calendar. This feedback encompassed any design features they wanted to applaud or critique. Furthermore, they were invited to share their thoughts on the implementation of privacy and medication conflict features, as well as the usefulness of the visualizations for self-reflection.

To determine the system's efficiency, the response time of the system was also measured for multiple requests involving different numbers of conflicts. We recorded the time it took the server to process a request and send the response back to the client application (calendar). The requests of interest were those in which the full calendar read request was initiated by the client. On average, 5 requests were made involving zero to ten pairs of conflicts.

8.3.3 Data Collection

Participants' data was collected in the form of text, Audio, and Video streams. This came from the recording of the study sessions. The audio component captured sentiments from the participants and included a demographics questionnaire that was also administered at the beginning of the interview. The video components recorded user interaction activities on the calendar. Activities such as the start and end of tasks were recorded through timestamp logging. Timestamp logging was also used to record the time it took for the system to complete user requests.

8.3.4 Data Analysis

Analysis of the results was conducted in two parts. First, we had quantitative data from tasks in R_A, R_B, and R_C. The data was about the correctness and efficiency of performing the tasks. Descriptive statistics were applied for the analysis. This choice was based on several factors: the insufficient volume of data to support meaningful statistical tests, the absence of a baseline for comparative analysis, and the primarily qualitative nature of the analysis, with time and error rates serving to provide context to the qualitative findings. In the second part, we had qualitative data that came from opinions and sentiments expressed by the participants on the way the calendar was implemented. These sentiments came from tasks across all usability requirements (R_A - R_D). We also had qualitative data that came from the participants' interpretation of the medication history charts (R_D). This data was qualitatively analyzed using Open, Axial, and Selective coding [198]. As indicated earlier, the data came from three streams; activity logs, audio, and video streams. Performance data was obtained from the activity logs. User Interface (UI) Interaction behavior was manually obtained from video streams. Sentiments were obtained from the audio transcript provided by Zoom [236].

Table 8.1: Table of calendar features and management tasks.

Req.	*Task Identifier*	*Task Description*
R_A	FEAT_DATE	Locating the current year and month
	FEAT_WEEK	Locating the days of the the week
	FEAT_HOURS	Locating the hours of the day
	FEAT_DAY	Identifying the current day
	FEAT_TIME	Identifying the current time
	FEAT_EVENT	Identifying calendar event entries
	FEAT_DAILY	Locating medication daily summary
	FEAT_WEEKLY	Locating medication weekly summary
	FEAT_MED	Identifying calendar medication entries
	FEAT_ICONS	Identifying medication status icons
	FEAT_NAV	Locating calendar navigators
	FEAT_CREATE	Locating event create button
	FEAT_HISTORY	Locating medication history charts
	FEAT_PRIVACY	Locating medication privacy settings
R_B	EVENT_ADD	Adding event to calendar
	EVENT_EDIT	Editing calendar event
	EVENT_DELETE	Deleting event from calendar
	MED_ADD	Adding medication entry to calendar
	MED_MOVE	Moving calendar medication entry
	MED_STATUS	Changing status of medication entry
	MED_DELETE	Deleting medication entry from calendar
	MED_DISPLAY	Changing display name of medication
	MED_COLOR	Changing color of medication entries
	MED_HIDE	Hiding all medication from calendar
	MED_SHOW	Showing medication entry in calendar
	MED_NOSUMMARY	Hiding medication summaries from calendar
	MED_PENDING	Showing only unscheduled medication in summary
	MED_SCHEDULE	Creating default schedule for medication
R_A	CON_WARNING	Identifying scheduling conflict warnings
	CON_DETAILS	Describing scheduling conflicts
	CON_RESOLVE	Resolving conflicts shown in calendar
	CON_HELP	Describing features that helped conflict resolution

Table 8.2: Table of distinct self-reflection tasks performed.

Req.	*Task Identifier*	*Task Description*
R_D	REF_CONSISTENCY	Reading adherence trends using the calendar or chart
	REF_MOST	Identifying most consistent medication
	REF_LEAST	Identifying least inconsistent medication
	REF_DAYS	Identifying daily adherence patterns
	REF_JUSTIFY	Rationalizing adherence pattern
	REF_ALTERNATIVE	Interpreting adherence trends using alternative designs

Quantitative Analysis

Tasks in R_A had the participant discover some calendar features. For these, we measured both the time that was spent discovering the feature and an indication of whether the discovery was completed (a correct response was given). The task completion time was measured by the prompt controls. There were four controls on the prompt (*Previous, Next, Skip, and Done*). We recorded the timestamp when the task came up (The participant clicked *Next* to load a task) and the timestamp when they marked it complete (The user clicked the *Done* checkbox to mark the task as completed). Correctness was determined by manually looking at the task execution from the video streams. This is because it was possible for the user to indicate a task as completed when the action performed was not the right one. The same metric recording procedure was followed for tasks in R_B and R_C. We also record the timestamp of the request processing time by the server. This record indicated the time when the calendar client's fetch requests were received by the server and the timestamp of when the response was sent to the client by the server. We conducted data processing and aggregation using Microsoft Excel [235]. We analyzed central tendency measures such as mean, median, and standard deviation. Because the data was not skewed, and we only had 12 participants, only the mean has been reported. Tableau was used for visualizing results [199].

Qualitative Analysis

When performing tasks in R_A, R_B, and R_C, participants would often pause between tasks to express their opinions and give suggestions about the implementation. Tasks in R_D involved reading the visualizations presented to the user. participants had to give a verbal answer. The sentiments obtained were analyzed using NVivo [197]. We employed a round of sentiment matching (matching sentiments to tasks) and three rounds of coding; open coding (identifying interesting concepts), axial coding (grouping the concepts), and selective coding (relating the concepts) [198].

During sentiment matching, we skimmed through the transcribed statements and grouped the sentiments according to the task to which they were referring. No interpretation of the participants' statements was conducted at this stage. The full extracts were transferred to the task group without any modification. Blocks of text that referenced more than one task were copied to as many tasks. We then proceeded to the open coding stage. During this stage, we went through the sentiments for each task, identifying interesting concepts. The resultant codes were added to the task group to which they belonged. 32 tasks had codes. Some tasks (especially those under the discovery stage) had no associated user sentiments.

Selective coding was then conducted. This involved grouping all the interesting concepts identified in the previous stage. The tasks were abstracted at this point and the resultant codes were grouped according to the sentiment, thus removing duplicates. This resulted in 21 codes. Examples of these codes included ease of use, non-intuitive placement, privacy demands, and calendar sharing. The final stage was selective coding. These codes were further analyzed to identify and define relationships between them. Some codes directly addressed guidelines employed in the design of the calendar, such as familiarity with design, clutter, and personalization. Others were based on new additions to the calendar design. The sentiments, including commendations and criticisms, addressed general themes such as calendar features, privacy, conflict management, and self-reflection (directly linking to our usability requirements). These will all be presented in the results section below.

8.4 Study Results

Results will be presented in sections that match the usability requirement being tested. This will be in four parts. The first part will present results from the feature discovery tasks. The second part will present results from the event and medication entry management tasks. The third part will present results from the conflict management tasks. This will also include systems performance results. The fourth part will present results from self-reflection tasks. Results from questions about design preferences will be presented as insets in the sections aligning with the expressed sentiments.

8.4.1 Calendar Feature Discoverability

There were 14 distinct tasks related to feature discovery that participants engaged in. These tasks were designed to address usability requirement R_A. They encompassed three primary categories: (i) exploring the calendar's temporal features, (ii) interacting with calendar events, and (iii) managing medication entries. We measured both the correctness and efficiency in task completion. For each task, we documented the number of participants who successfully completed it, along with the average (mean) time taken for completion. We measured task completion times in seconds to determine how quickly participants could perform common calendar-related actions. This data served the purpose of identifying any tasks that took an unusually long time to complete, indicating potential usability issues or areas for improvement in the application's design. Additionally, it helped assess the overall efficiency and user-friendliness of the calendar application in terms of task execution. In cases where participants encountered difficulty, guidance on task completion was provided at the end of the study. Additionally, we gathered user sentiments associated with these tasks, wherever available. The summarized findings, illustrating the task completion rate and the average time taken for completion, are visually represented in Figure 8.1 and Figure 8.2, respectively.

Calendar's Temporal Features

There were 6 tasks in this category (FEAT_DATE - FEAT_NAV). All twelve participants (P = 12) performed FEAT_DATE, FEAT_WEEK, and FEAT_HOURS. The average time for the three tasks was 18.7 seconds, 11.5 seconds, and 14.5 seconds respectively. The time range (24 hourly markers) for the calendar was collapsed to start from 5 AM. This was done to limit the amount of scroll for an active day beginning after 5 AM. Hence, FEAT_HOURS also tested users to see if they would expand the range to view time from 12 AM to 11:59 PM. Only one participant (P = 1) expanded the time to reveal 12 AM. The rest of the participants worked with the default setting (5 AM to 11:59 PM).

Eleven participants (P = 11) successfully performed FEAT_DAY. The average completion time was 13.5 seconds. Ten participants (P = 10) successfully completed FEAT_TIME with an average completion rate of 23.2 seconds. All participants (P = 12) successfully completed FEAT_NAV with an average completion time of 23.

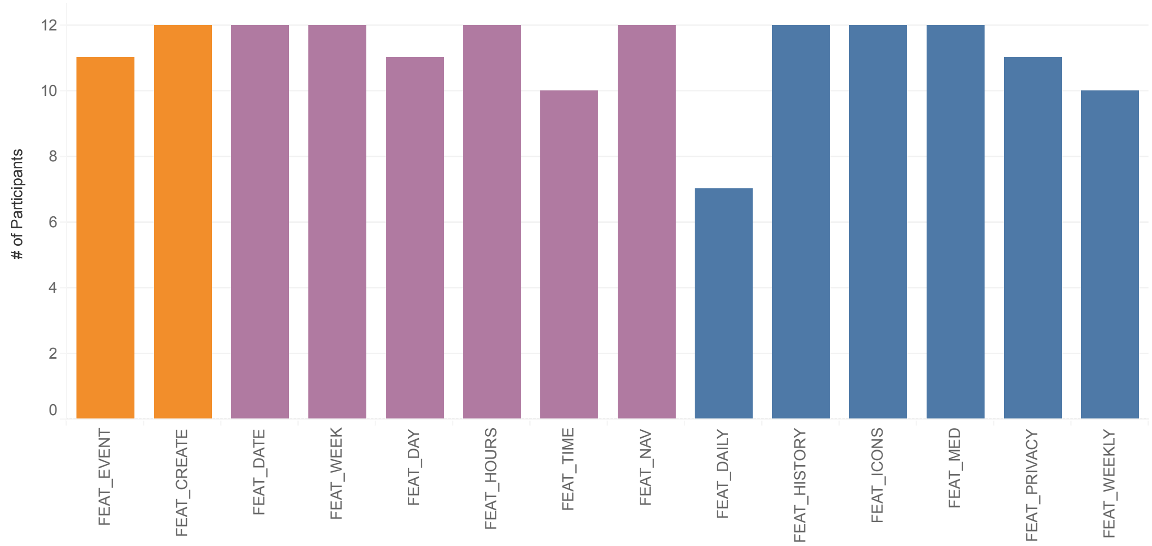


Figure 8.1: Chart showing the completion rate under feature discovery. 8 participants discovered all features. FEAT_DAILY was the most difficult task with only 7 participants completing the task.

Calendar Events

There were 2 tasks in this category (FEAT_EVENT and FEAT_CREATE). Eleven participants (P = 11) successfully completed FEAT_EVENT. The average completion time for FEAT_EVENT was 38.1 seconds. All participants (P = 12) successfully completed FEAT_CREATE with an average completion time of 11.4 seconds respectively.

Medication Entries

There were 6 tasks that were performed in this category (FEAT_DAILY - FEAT_PRIVACY). Seven participants (P = 7) successfully completed FEAT_DAILY with an average time of 67.2 seconds. Five participants (P = 5) failed to perform FEAT_DAILY. Participants who failed to perform FEAT_DAILY went through all the entries in the presented week as opposed to locating the summary UI. Nine participants (P = 9) successfully completed FEAT_WEEKLY. The average completion time was 39.4 seconds. All participants (P = 12) successfully completed FEAT_MED, FEAT_ICONS, and FEAT_HISTORY with an average completion time of 42.2 seconds, 56.7 seconds, and 39 seconds respectively. Eleven participants (P = 11) successfully completed FEAT_PRIVACY in 36.8 seconds on average.

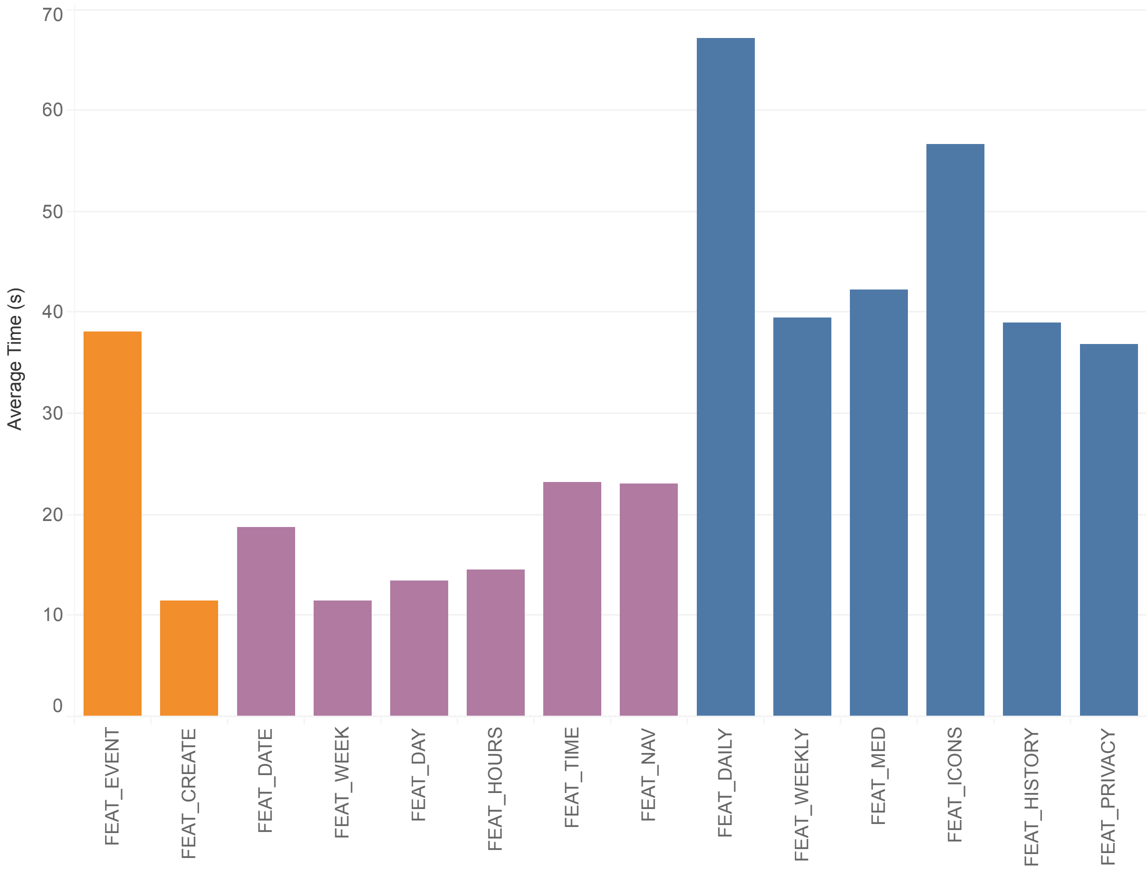


Figure 8.2: Chart showing the completion time under feature discovery. Discovering features related to medication entries took longer than discovering the calendar's temporal features.

As can be seen from the results, all tasks under the discovery stage were performed quickly. All the tasks, apart from FEAT_DAILY, were completed in less than a minute. It is important to re-emphasize at this point that participants had no prior exposure to the calendar. Hence, being able to discover all calendar features in less than 30 seconds on average was quite fast.

8.4.2 Calendar Management Tasks

There was a total of 18 calendar management tasks spread across three categories. The tasks were distributed as follows; 3 tasks for event management, 11 tasks for medication management, and 4 tasks for conflict management. All the tasks in the three categories were performed twice, with an interlude of self-reflection tasks between the attempts. For the first attempt, participants were given an opportunity to ask for a hint if they had difficulties performing the task. A task would only be marked as failed if the participant failed to perform it after getting a hint. In such a case, the participant was given instructions on how to perform the task. There was no provision for hints during the repeat stage. We recorded both the completion rate and average completion time for all tasks in event management and medication management. For conflict management tasks, CON_WARNING and CON_RESOLVE are the only tasks for which completion rate and completion time were recorded. CON_DETAILS and CON_HELP were subjective tasks requiring the participants to give detailed explanations of the task completion process. For these two tasks, we only report an indication of whether the explanation given was satisfactory.

Event Management

There were 3 tasks that were performed in this category (EVENT_ADD - EVENT_EDIT). All participants (P = 12) completed EVENT_ADD, EVENT_EDIT, and EVENT_DELETE with average completion time of 90.3 seconds, 74.6 seconds, and 26.8 seconds respectively. Participants completed EVENT_ADD by using the Create button (P = 7) or clicking within the calendar (P = 5). Participants were given tasks without being told how to go about doing it. They had to use their intuition, aided by previous exposure to calendars, to complete the tasks. No hints were used by any participant.

Medication Management

There were 11 tasks in this category (MED_ADD - MED_STATUS). Results for these tasks will be presented according to the type of the task. The types included Entry Creation, Entry Update, Entry Personalization, Visibility Control, and Schedule Creation.

Entry Creation Ten participants (P = 10) successfully completed MED_ADD. Two participants (P = 2) managed to complete the task with the help of a hint. The hint *"Read all the details that you can see from the current view"* was given to help participants find the instructions on how to add a medication entry to the calendar. The *"current view"* was the medication details page. The average completion time was 111 seconds. To add a medication entry to the calendar, participants were required to drag an entry from either the daily summary or weekly summary into the calendar. Seven participants (P = 7) attempted using the event creation approach before discovering the instruction. participants were repeatedly clicking the Create Event button or clicking within the time slots to bring up the event creation dialog box. Participants contended that medication entries should be added using the Create Event button. Echoing their design preference, P11 said *"I would like to be able to create a medication entry from Create [button]"*. Some participants avoid using the Create Event button and spent time trying to edit the medication's information. *"I'm not hitting the [Create Event] button because I understand this is only for events."* Said P9, while looking for the right approach. Ten participants (P = 10) dragged the medication from the weekly summary. Only two used the daily summary.

Entry Update All participants (P = 12) successfully completed MED_MOVE, MED_DELETE, and MED_STATUS. The average completion time was 34.1 seconds, 31.7 seconds, and 23.5 seconds respectively. No hints were used. There were two approaches to completing MED_MOVE. One was by editing the medication's time using the edit box. The other was by dragging the medication to the new time slot. All but one participant used the drag method. *"I like that I can drag stuff."* Commended P8. The design of the status-switching selectors proved problematic to some participants. The design, which used checkboxes, made three participants click the choices multiple times to deactivate a selection. These participants recommended that radio buttons be used in place of checkboxes. They reasoned that even though the checkboxes used worked like radio buttons, they gave the notion that a user can select more than one status at the same time. To view available statuses, some participants were observed checking through the entire calendar. Others clicked the medication entries for details and obtained the required information from there. There were also participants who used both views to obtain the information.

Entry Personalization All participants (P = 12) successfully completed the task MED_COLOR. Eleven participants (P = 11) successfully completed MED_DISPLAY. One participant (P = 1) managed to complete MED_DISPLAY after the hint *"Check the details of the medication"* was given. The average completion time was 43.5 seconds for MED_DISPLAY and 23.1 seconds for MED_COLOR. When personalizing the medication display, ten participants (P = 10) accessed the functionalities by using the Weekly Summary. Two participants used the Daily Summary.

Visibility Control Ten participants (P = 10) successfully completed MED_HIDE. Two participants (P = 2) needed a hint to complete the task. The question-form hint *"How can you manage access to your calendar entries?"* was given. The average completion time for the task was 37.8 seconds. Participants had difficulties locating the functionality with four participants (P = 4) initially checking for the functionality under the medication's edit window. All participants (P = 12) successfully completed MED_SHOW (average time of 28.5 seconds), MED_NOSUMMARY (average time of 21.9 seconds), and MED_PENDING (average time of 32.3 seconds).

Finding the medication visibility control features was not immediately intuitive for some participants. About five participants were observed checking the medication details and reviewing the medication summaries before turning to the settings icon. Participants believed that managing the visibility of medications from the settings window was not intuitive. *"The privacy setting was very private. It was kind of very hard to find it."* Said P9. Two participants (P = 2) proposed that the functionality to show and hide medication be put next to the summaries. *"It's better to just have like a hide or show button here [pointing to the summary], rather than a separate menu."* Proposed P12. Others, like P1, thought the feature to show and hide medication should be part of an individual medication item. The participant suggested that users should be able to control the visibility of selected medications from the calendar. This suggestion drifts away from the approach used in the design in which a user could either show or hide all medications without exclusion. *"I kind of found it weird how the show or hide medication is in the settings. Maybe this feature could be incorporated when you're trying to create a medication [schedule], like if you want to hide each individual medication."* Said P1. Other participants submitted that naming the visibility control feature a privacy feature was misleading. While they thought managing visibility was a privacy control feature, some participants suggested a change in the naming of the feature. *"The naming of privacy setting is not good for here."* Said P5, pointing to the settings icon. P2 also agreed with P5 that Privacy Setting was not an obvious choice for hiding and showing medication. *"I think for just some of the wordings, like the privacy settings. It's not like obvious."* Argued P2.

Schedule Creation Eight participants (P = 8) successfully completed MED_SCHEDULE. Four participants (P = 4) completed the task after receiving hints such as *"Read all the details in the current view"* (P = 3) and *"Create a schedule for all* *days at once"* (P = 1). MED_ SCHEDULE was the slowest task across all categories, posting an average completion time of 161.2 seconds. Like MED_ADD, four participants (P = 4) tried creating the schedule using the Create Event button. Participants were observed clicking the Create Event button, clicking in the time slots, dragging summaries into the calendar, or trying to edit the repetition cycles for medication. Eleven participants (P = 1) used the Weekly Summary to access the schedule creation feature. Participants indicated that the process of creating medication schedules was not intuitive. They indicated that the most intuitive option was to use the Create option. Participants therefore suggested that schedule creation should be handled in the same way as event creation. This implies that the creation of schedules should be initiated by clicking either the Create Event button or the ideal time slot within the calendar. *"I would like to be able to create a medication entry from Create."* Said P11. P8 had a similar suggestion. *"I wish it was a little bit more intuitive of how to create schedules. I want to be able to go into here [Individual med entry details] and add the schedule."* The suggestion was also supported by P6. "I think there's a need for a quick way of doing it the same way events are created by just clicking on a particular time slot."

Medication Conflict Management

There were 4 tasks in this category (CON_WARNING - CON_HELP). All participants (P = 12) successfully completed all the tasks in medication conflict management. 2 of the tasks (CON_DETAILS and CON_HELP) were subjective. For the two tasks, we did not consider completion time. For CON_WARNING, the average completion time was 61 seconds. CON_RESOLVE lasted 27.9 seconds on average. All participants (P = 12) supplied the correct response on the type of error message. Participants used the popup message (revealed by mouse-over) on the conflicting entries to describe the nature of the conflict (CON_DETAILS). For CON_HELP, eight participants (P = 8) said that the information on the minimum separation and the ability to drag and drop the entries were the features that helped them resolve the conflict. Three participants (P = 3) said they used the popup message to know the minimum separation time and used the arrows to know the direction to move the entries. One participant (P6) said they just randomly moved the entry until the warning message disappeared.

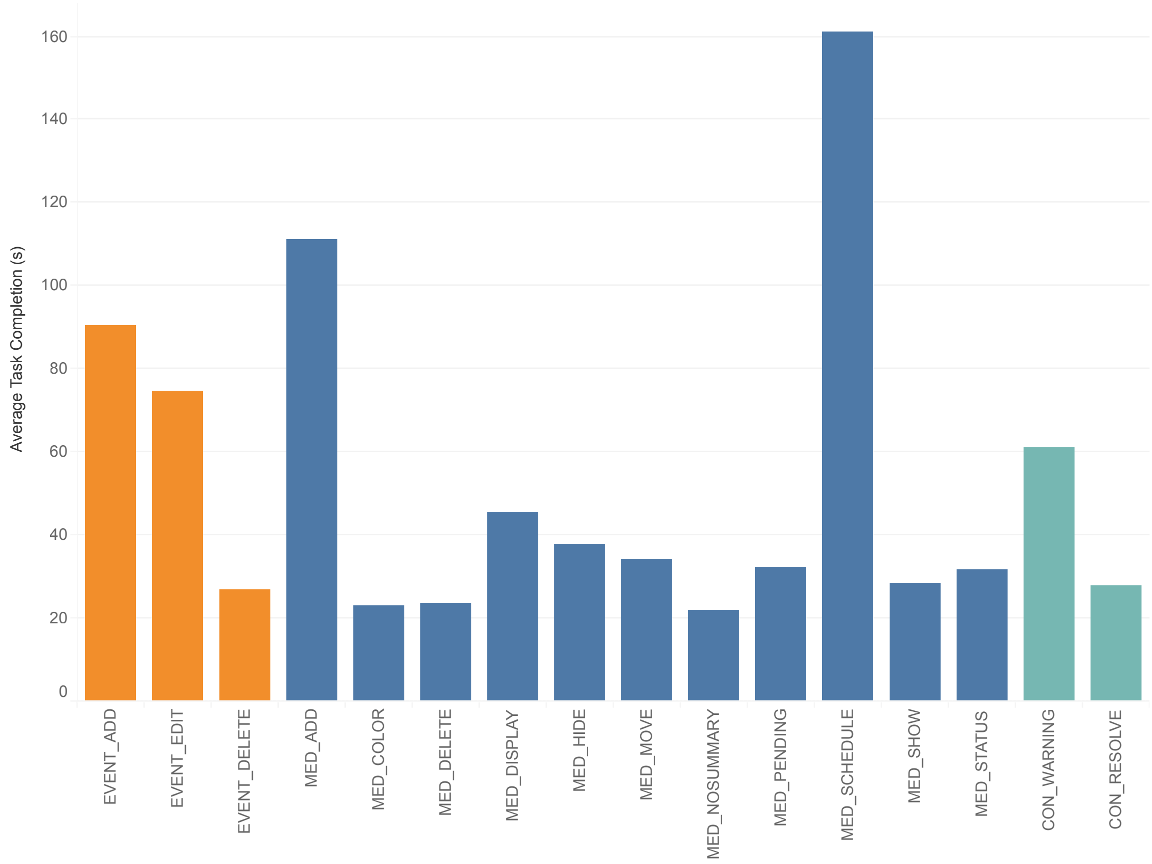


Figure 8.3: Chart showing the completion time for calendar management tasks. The completion time was almost uniform across all medication tasks except MED_ADD and MED_SCHEDULE.

Conflict Response Time

When handling fetch requests for a calendar period that did not have medication conflicts, the server's average response time was 0.117 milliseconds. The observed minimum time was 0.106 milliseconds, and the maximum time was 0.137 milliseconds. The time posted by the system when conflict detection was involved did not deviate much from the one observed when there was no conflict. For some iterations, the system even posted lower times (this comparison was made on independent runs). The overall minimum time observed was 0.092 milliseconds (for a request involving 6 pairs of conflicts) and the maximum time was 0.223 milliseconds (for a request involving 10 pairs of conflicts). The response time for different numbers of conflicts is presented in figure 8.5.

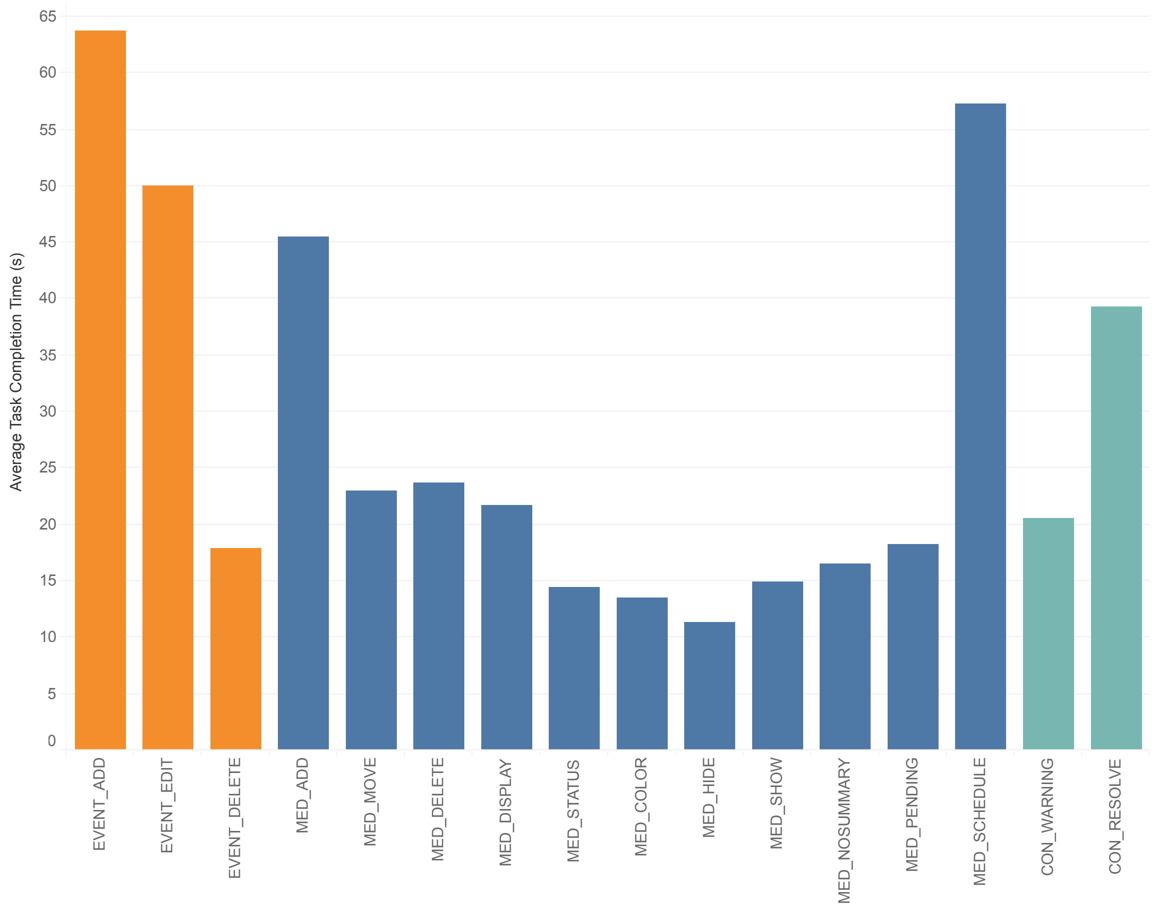


Figure 8.4: Chart showing the completion time for *repeated* calendar management tasks. Task completion time reduced by over a factor of two for all tasks. Completion time for MED_ADD and MED_SCHEDULE matched that of event tasks.

Repeat Stage

All calendar management tasks were repeated after an interlude of self-reflection tasks. The repeat execution of tasks improved in both completion rate (no hints were given) and completion time. There was as much as a threefold reduction in the average completion time across most of the tasks. For some tasks, the difficulty was increased, and this was reflected in the completion times. For example, CON_RESOLVE took longer during the repeat stage because of the additional steps required by the task. During the first iteration, participants were resolving conflicts involving two medications. The repeat stage had a conflict involving four medications. Figure 8.3 summarizes the completion time of all 18 tasks. Figure 8.4 shows the time spent doing the same tasks during the repeat stage.

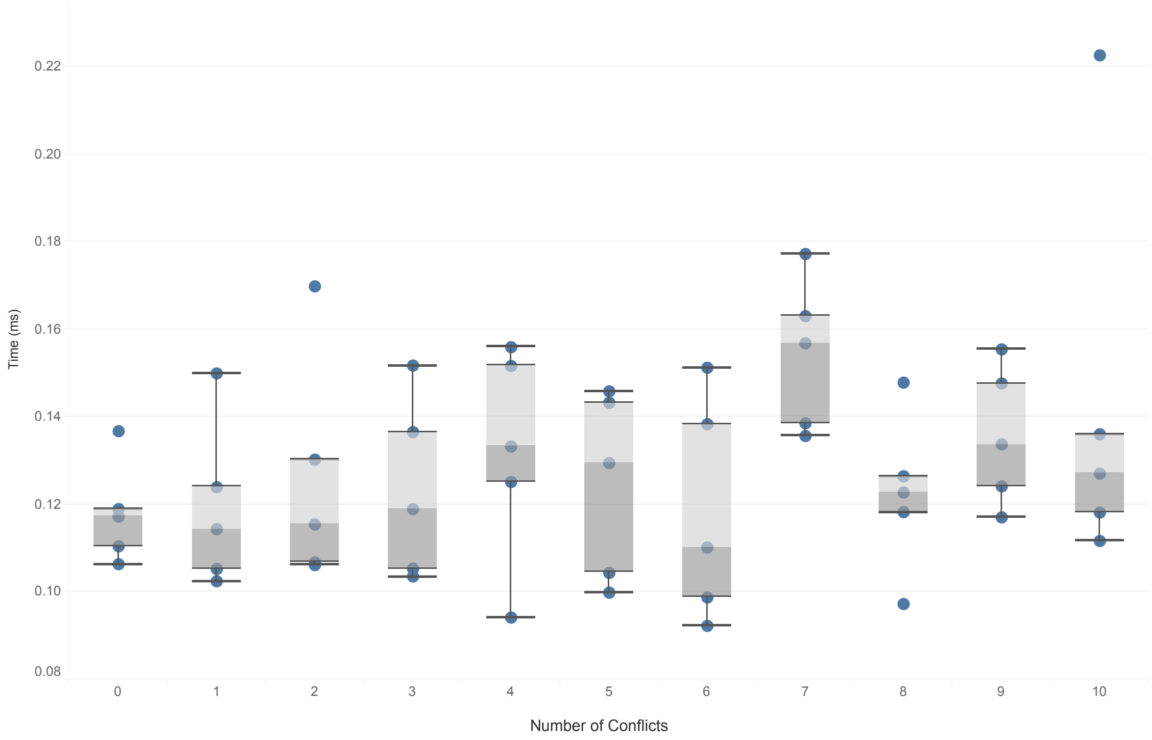


Figure 8.5: Box plot showing average (First Quantile, Median, Third Quantile) request handing time by the server. The median time for requests with no conflicts (zero conflicts) was at par requests with requests involving multiple conflicts (See 6 conflicts).

8.4.3 Self-Reflection Tasks

After completion of calendar management tasks, participants were asked to perform tasks related to the interpretation of medication administration patterns. That is, they were asked to self-reflect. Participants performed 6 unique tasks. The tasks included reading adherence trends using the calendar (REF_CONSISTENCY), identifying the most and least consistent medication from a given view (REF_MOST, REF_LEAST), identifying daily adherence patterns (REF_DAYS), reasoning about possible causes of adherence (REF_JUSTIFY), and reading adherence trends using alternative views (REF_ALTERNATIVE). Tasks were spread across four levels, with selected tasks repeating in some levels, for a total of 14 tasks. The four levels included reading adherence using the calendar view (one level) and reading adherence using medication history charts (three levels). Results for these levels are presented below.

Self-Reflecting via Calendar View

There were 5 self-reflecting tasks that were performed using the calendar view (REF_ CONSISTENCY - REF_JUSTIFY). Participants were given a calendar week showing 89% adherence levels. Two medications (Zopiclone and Citalopram) were given a 100% adherence rate. Two other medications (Amitriptyline and Tylenol) were given 72% and 87% adherence rates respectively. All these medications were scheduled to be taken once a day except for Tylenol (scheduled four times a day). Adherence rates were at 100% on Thursday, Friday, and Saturday. Sunday, Monday, and Tuesday were at 85% adherence. Wednesday was at 60% adherence.

Participants used different qualifiers when explaining adherence level under REF_ CONSISTENCY. Using only the calendar to estimate the level of adherence, participants used qualifiers such as mostly consistent (P = 5), relatively consistent (P = 1), not very consistent (P = 3), semi-consistent (P = 1), and consistent (P = 2). The consensus was that the user was consistent. However, despite being able to summarize adherence using the calendar, participants suggested that a summary should be provided. Two participants suggested that adherence levels should be summarized in the calendar view. They submitted that the calendar should summarize scheduled medication versus taken medication as part of the weekly summary. *"I would like to have other options like showing me how many I missed here, next to the weekly summary. Like how many I was planning to have taken and how many I really took."* [P9]. This suggestion was echoed by P11.

All participants (P = 12) provided an answer under REF_MOST. Eight participants (P = 8) correctly identified Zopiclone and Citalopram as the medications in which the user is most consistent. Four participants (P = 4) were partially correct (mixing Zopiclone with Amitriptyline). All participants (P = 12) provided an answer REF_LEAST. Four participants (P = 4) answered correctly. Eight participants (P = 8) answered incorrectly, indicating Tylenol as the least consistent.

For REF_DAYS, all participants (P = 12) provided an answer. Five participants (P = 5) answered correctly. Five participants (P = 5) were partially correct. Those who were partially correct indicated a list of days that included days in which the user missed some medications. Two participants (P = 2) answered incorrectly.

REF_JUSTIFY proved very difficult for participants. Out of all participants (P = 12), only three (P = 3) gave a reason to justify the adherence pattern. The rest did not provide a reason. One associated consistency with taking multiple medications at the same time and taking the medication with food. They suggested that when multiple medications are scheduled to be taken at the same time every day, then the user is consistent. They also suggested that medication that was scheduled to be taken with food was never missed. *"They are more consistent on taking 2 or more medications at the same time. And the one that's taken with food is like consistently taken same time every day."* [P12]. Participant P2 married inconsistency with a busy schedule. They suggested that on days when they had a lot of things scheduled in the calendar, they missed some medications. Another participant picked a routine of consecutive days and a break. They suggested that the user either missed two days in a row and then took the medication on the third day or vice versa. "I can see a pattern, an inconsistency of [Amitriptyline] and Tylenol. So, they have not taken it on Monday. They have not taken it on Sunday. But they have it on Tuesday and not on Wednesday, but again on Thursday and Friday." Reasoned P4.

For all tasks at this level, participants were observed tracing events in the calendar and manually counting the medication entries.

Self-Reflecting via Overall Chart

There are 4 tasks that were performed using the *Overall Chart* (See figure 8.6). These are REF_CONSISTENCY, REF_MOST, REF_LEAST, and REF_ ALTERNATIVE.

All participants (P = 12) were able to explore the chart and correctly answer both REF_MOST and REF_LEAST. For REF_CONSISTENCY, all participants used percentages to explain adherence levels. Participants were able to read adherence levels for both individual and all medications. *"So we can see they are 76% consistent. So, they missed 24%, or almost a quarter of the time. Zopiclone is quite highly consistent at 88%. Citalopram is more so at 92%. The overall consistency is 76%"* Answered P4 in response to REF_CONSISTENCY. Other participants, like P8, went as far as trying to determine how some medications were affecting the overall adherence rates. *"Well. So, Seroquel is the one driving down the average. That's an outlier..., not too much. It's definitely not statistically significant."* Reasoned P8 while exploring the chart. Some participants had to be reminded that they could explore the chart by refining the display. These were fixated on the initial display when performing the task.

Participants performed REF_ ALTERNATIVE using *Overall A2* (See A2 in figure

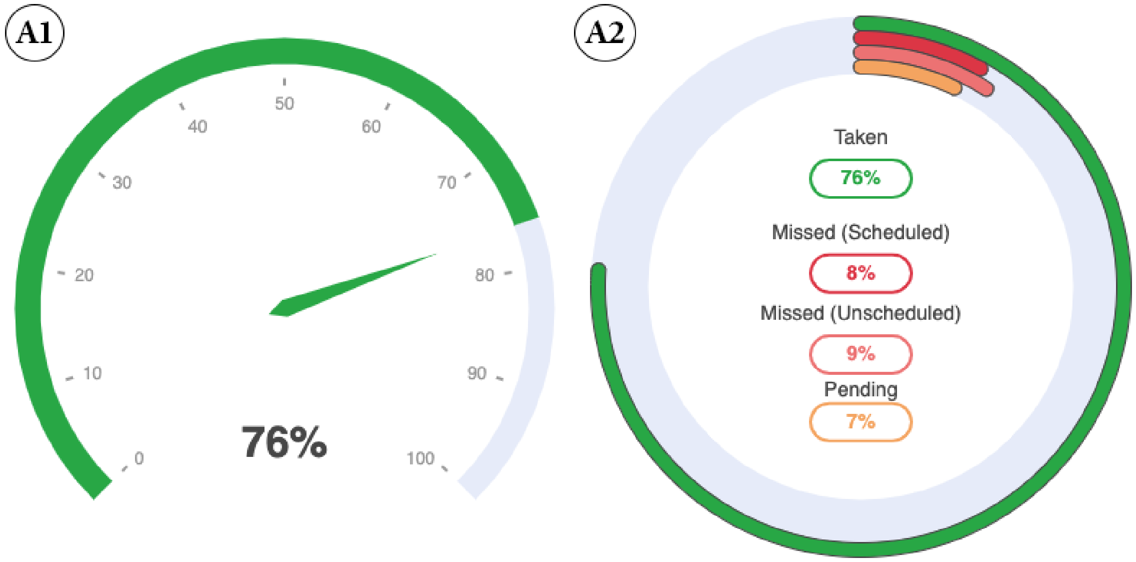


Figure 8.6: *Overview* Visualization showing the overall adherence for all medications over a selected period of time. Participants were able to read adherence levels for both charts.

8.6). Different levels of detail were provided for the task. Some participants could only read the statuses but could not use them to derive insights. *"I'm not sure how I can what I can learn from this. I can see, like the percentage they missed. And what percentage of this is scheduled and unscheduled? But I'm not sure like what insights, I can derive from it."* Said P2. Some were able to relate the statuses to the overall picture. *"This presentation is like much of the medication that was not taken is like pending... So, some medicine was just not scheduled, and I guess that's what contributed to the lowering of the overall."* Said P6. There were also participants who did not understand the meaning of the new status that appeared only in the charts but not in the calendar. In the calendar, the medication that had a pending schedule only carried a warning beside the name. However, in the charts, a summary of all the unscheduled medication entries was also given. This is what some participants did not know what it meant. *"I'm looking at the missed and the missed unscheduled. I'm not really sure what missed unscheduled means. What's the difference?"* Asked P11. The explanation was given.

Self-Reflecting via Weekly Chart

There are 4 tasks that were performed using the *Weekly Chart* (See figure 8.7). These are REF_ CONSISTENCY, REF_DAYS, and REF_ ALTERNATIVE (repeated for Word Cloud and for Bar Chart).

All participants (P = 12) provided a response to REF_ CONSISTENCY and REF_DAYS. All participants were able to read the weekly pattern and identify days when adherence was highest and days when adherence was lowest. Most of the participants identified the daily pattern in response to REF_ CONSISTENCY. For such participants, REF_DAYS was altogether skipped. *"I think during the start of the week the user usually takes the education. Then, in the middle of the week, they miss a little. Then again, at the end of the week, they improve."* Said P7. Some participants even came up with factors that could have affected adherence levels during the week. One participant justified that the low adherence levels towards the end of the week could have been caused by the user's reluctance to update the calendar. *"I would say also that they are perhaps worse at tracking their medications on Wednesday to Saturday."* Said P11. Another participant associated the low adherence towards the end of the week with forgetfulness and a busy schedule. *"I guess between Wednesday and Saturday they have a lot of pending the maybe they just forgot to mark on these days, and I think, just from like, if I recall using my memory, it seems like they had a lot going on these days that they could have just forgotten to mark it".* Justified P1. Eight Participants (P = 8) were also able to read adherence levels across the week.

All participants (P = 12) successfully performed REF_ ALTERNATIVE (Using the Word Cloud). Participants were able to associate tags with both causes and outcomes. *"Okay, so I see that they have had a busy week. I assume that would make adherence lower during the week. If you're in transit and you're busy, obviously you're not taking it. And then I would say that if you have no pain, then of course you just intentionally choose not to take a pain medication. It's not necessary."* Said P11. *"So from here you could see that they when they take the medication. it's either they are feeling rested or feeling okay. Or they are having some pain or are tired or feeling unwell. But again, if they miss the medication, they are probably feeling good."* Said P7. Ten participants (P = 10) were able to associate tag colors with statuses.

All participants (P = 12) successfully performed REF_ ALTERNATIVE (Using the Bar Chart). Eight Participants (P = 8) found the Bar Chart easier to read than the Stacked Bar Chart. *"Yeah, I prefer this one (pointing to bar chart). It makes*

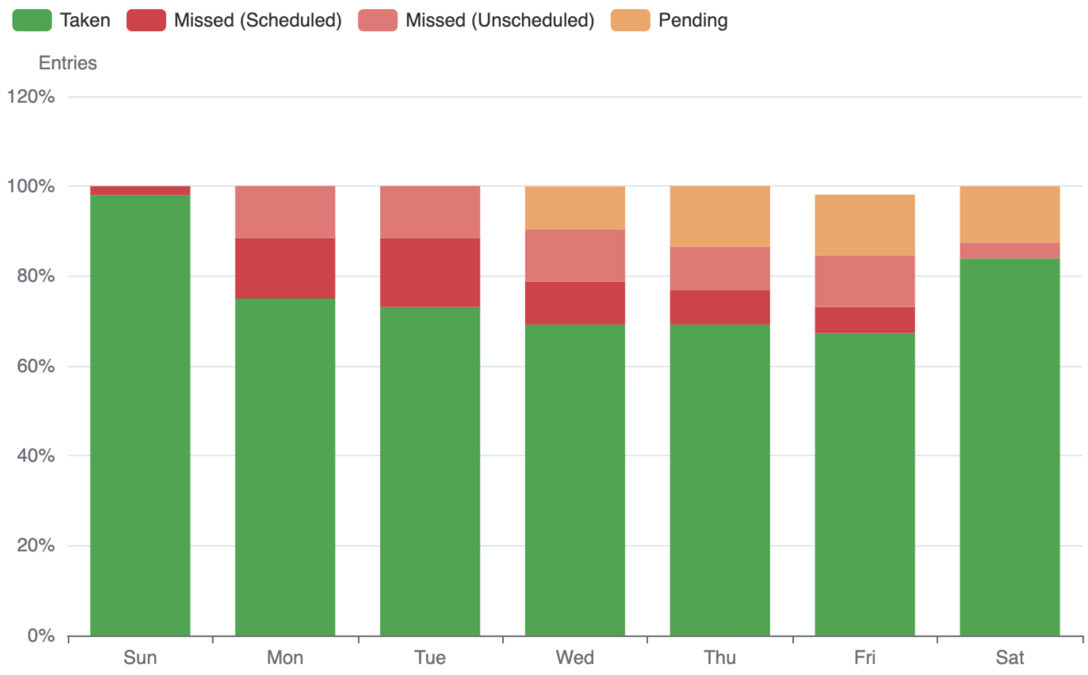


Figure 8.7: A visualization showing adherence rate by weekdays. Participants were able to identify weekly patterns and interpret the reason with the Word Cloud (Not shown in this image).

it a lot easier to notice the differences between the statuses." Said P12 in support of the Bar Chart. Four participants (P = 4) preferred the Stacked Bar Chart. One participant thought the bar chart made the non-taken statuses look insignificant. *"I think the stacked is a little bit more representative. it might make me biased in the way that may make me feel like I'm doing worse than I am, but I'd rather strive to do better and better than to accidentally think I'm doing better."* Said P8.

Self-Reflecting via Timeline Chart

There were 2 tasks that were performed using the *Timeline Chart* (See figure 8.8). These are REF_ CONSISTENCY and REF_JUSTIFY. All participants (P = 12) were able to identify a pattern that was time dependent. Participants were able to see pattern changes over a period. Most of them correctly noticed that the user was consistent at the beginning of the shown period and inconsistent towards the end. *"It seems to be like quite orderly at the very beginning that they took exactly as prescribed. And then things deviated perhaps. That reality looked different than the prescribed. And I would say that over time, adherence got worse."* [P11]. Two participants (P =

2) were able to see patterns based on the time of the day. These noticed that the user was less consistent in the afternoon and more consistent in the evening. *“ The user misses a lot of medication at around 4:00 PM. They take more at 8 PM”*. Observed P10. Only the 12 participants, only three participants interacted with the chart at a level that revealed the tags associated with all missed entries. Three participants suggested that the orientation of the time on the y-axis should be reversed to align with the calendar.

Medication: Tylenol
Status: **missed**
Date: 2023-04-17 22:00

Figure 8.8: This visualization shows the Timeline Chart. Participants were able to identify the high adherence levels ate the beginning and high pending levels at the end. Some participants event identified adherence patterns based on time.

Participants found REF_JUSTIFY difficult. Only one participant (P = 1) supplied a reason to justify the observed pattern. *“Maybe they had a late night out, and they were like, Oh, shoot! I forgot to take my medication, so that’s why they took it then.”* [P1]. Participants were expected to explore the charts and refer to the calendar to derive insights. Participants were expected to explore the charts and refer to the calendar to derive insights. No participant (P = 0) referred to the calendar to find an explanation. The nearest that we got was a participant who tried to recollect from the calendar view experience.

Choice of Chart We asked the participants to indicate the chart that they would use to get a quick view of their personal history. Nine participants (P = 9) indicated that they would use the *Weekly Chart*, two participants (P = 2) selected the *Timeline Chart*, and one participant (P = 1) selected the *Overall Chart*. Reasons for choosing the *Weekly Chart* included (i) the presence of the word cloud, (ii) familiarity with bar charts, and (iii) week-based event cycles. Those who chose the *Timeline Chart* said the chart "provides it a very good overview of the calendar" [p2]. *Overall Chart* was chosen for its "simplicity in showing overall adherence levels" [p4].

8.4.4 Design Considerations

At the end of the tasks, participants were asked to give their general view of the design. They were also asked to indicate which features they would remove or add to the calendar, their view of the privacy settings, their view of the way the conflicts were rendered, and the usefulness of the charts for self-reflection. The results are presented below.

General View of the Calendar

Eleven Participants (P = 11) found the calendar easy to use. These participants described the calendar as having a simple design (P = 8), neat (P = 2), and user-friendly (P = 1). One participant (P = 1) found the calendar difficult to use. *"I find it difficult to use. The design should be improved to work even on old computers like mine"* Said P5, whose computer had display challenges. Four participants (P = 4) likened the design to that of already existing calendars and attributed this to their finding that the calendar was easy to use. *"It's very recognizable because it's seems to be closely modeled after Google Calendar, which I've worked a lot with, and it has the same features as most of the calendars that I've worked with, so it works as I expected it. It kind of fits my mental model of the calendar."* [P11]. Another participant credited their knowledge of calendars with helping them locate features. They cited their previous exposure to Google Calendar to decide where to turn their attention when looking for selected features. *I think it's pretty good. At least I'm familiar with Google Calendar and it looks pretty close to what I have there. So, it helped me to* *identify where I could be finding features."* Said P9. One participant found the design of the calendar cluttered. They could not specify exactly what they thought was causing the clutter but offered suggestions to improve the design. *"I think it's a bit overwhelming, especially because the view is very cluttered with But I can't really tell why. Maybe switch the appearance of events and medication"* They said.

Calendar Features

Participants had some features that they wanted to be added to the calendar. They did not propose any feature which they thought should be removed from the calendar. Some of the features proposed for addition are discussed below.

Ability to share calendars Three participants (P = 3) indicated that they want to be able to share (using a link) their medication calendars (with modified access level) the same way they do with event calendars. *"[I would like] a feature where you can just send a link to your calendar, and they have access to it. They can see when you're busy. I'd recommend having that modification in the link where, like they can see your medications or not depending on how much you trust them."* Said P8.

Another Medication Status One participant (P = 1) made a proposal for a fifth status where you can indicate a medication as missed but that you decided to miss it with a reason. *"I would add another [status] for you missed like maybe make it blue, or whatever, but it was because you didn't feel it was necessary to take it."* Said P8.

Scroll Indicator Two participants (P = 2) indicated their they would like to see an indicator showing that some entries were not visible in the visible window and that they should scroll to see the entries outside the visible window. *"The thing that I missed here was maybe an indication that there [are] more things down."* Complained P9.

Other features Some of the other features that were proposed include; showing medication summary as part of the medication lists (P = 2), adding a legend to the calendar for all icons (P = 3), and adding monthly and daily view to the calendar (P = 1).

Privacy Settings

All participants (P = 12) found the feature to hide and show medication in the calendar useful. While a number of participants did not like the placement of the feature (as reported under Section 8.4.2), the feature itself was well received. *"So I like that you can kind of change the privacy because I actually like my calendars are public at my university. So I like that I could change them not to show, for example."* Said P11. *"I think it's good that you're able to go into the settings and just show the medication as and when required."* Said P4. For one participant, being able to hide medication was not all about privacy. It was a way of looking at the calendar without a reminder to take medication. *"I like that I can show and hide medication. If the medication is always [there] by default. It's kind of a constant reminder that you have to take this. You have to take this. It can be frustrating."* Another participant proposed a visibility setting where you are able to hide all events and just leave the medication in the calendar. *"I just want to view the medications. I don't want to see these other events like the scheduled events."* Proposed P7.

Conflict Rendering

All participants (P = 12) found the idea of showing conflicts between medication entries useful. Ten participants (P = 10) found the way the conflicts were rendered reasonable. *"This is a good representation. I like the use of the exclamation. I relate this to let's say, on a road, if they are like several warnings an exclamation mark like the one you've used is indicated to give you a warning so simply by looking at it. before I even go to a task that required changing. I noticed to say, here is, there should be something that needs where I need to pay attention."* Commended P6. Two participants (P = 2) thought the indicators should be improved. *"The color is not so good."* Complained P5. Participant P7 proposed that conflicts between different medications should be rendered differently from those between the same medication. Two participants (P8 and P12) proposed that the color used to show conflicts (red) should not be used for as a color for either event or medication entries. *"I don't think I might be able to see any other error message just besides there two. But yeah, I do see them. I think they could be a little bit more obvious... Maybe a color that isn't like being used for the medications or events."* Said P8.

Self-reflection Charts

All participants (P = 12) found the self-reflection charts useful. *"It kind of helps me to reflect like. what is it that is causing me to take the medication. Or what is causing me to forget that can help with my behavior change."* Said P6. *"'I think they are all really good."* Commended P3. *"Yes, definitely, it's really helpful. It's faster for me to see if I'm being regular or not."* Added P9. The participant who was a nurse thought the charts would provide quick access to adherence information. *"This would help me find out [what I need] from the clients. I'd be able to tell the doctor how the patient is doing based on these charts. If they're missing their scheduled medications at certain times, for some reason. And I have the charts right there. That would help me give more accurate information."* Said the participant.

8.5 Discussion

Medication administration can be managed using calendars in the same way that everyday events and appointments are managed. Results indicate that users can complete calendar medication management tasks with the same level of fluidity as calendar event management tasks. The piloted calendar was implemented with support for basic functionalities such as reading, adding, editing, and deleting medication entries. In all these tasks, the task completion procedures used for event management by conventional digital calendars were employed. There were also some medication-related features that were new to the calendar such as medication conflict management and self-reflection. For these two, new ways of feature integration were used. The study was conducted with the objective of testing four usability requirements (R_A - R_D) related to the integration of medication management into calendars. Results indicate that users can (i) read calendar features related to both events and medications (R_A), (ii) manage both calendar events and medication entries (R_B), (iii) manage conflicts between medication entries (R_C), and (iv) summarize adherence trends using charts (R_D). Participants were able to quickly acquaint themselves with the calendar after assimilating the calendar to existing calendars such as Google Calendar and Outlook calendar. They utilized their existing knowledge of calendars to perform tasks across a spectrum of medication-related requirements.

8.5.1 Finding Calendar Features

Participants were able to discover all the calendar's temporal features with ease. This is not an unexpected observation since all the temporal features were implemented in a way similar to conventional calendars (layout designed with weeks on top, time to the left, navigation menu to the top left, and the date next to the navigation menu). The few task completion failures observed during the discovery of the calendar's temporal features may be attributed to not getting the prompt right. For example, when prompted to locate the current time, some participants read entries from the entire time axis. Participants were also able to discover medication-related features without much difficulty. The observed worst case, FEAT_DAILY, may be attributed to the implementation change from the conventional "All Day" calendar feature (for events whose duration is the entire day) to the "Daily Summary" (a quick peak for medication that is supposed to be taken during the day). This argument is even supported by the observation that almost all participants opted to use the weekly summary for tasks which could also be performed by using the daily summary. Hence, while familiarity was consciously used to locate many calendar features, it also unconsciously became a pitfall for the discovery of certain new features. Overall, the completion time for all the tasks was relatively quick. The reported time has a considerable level of noise. Participants who talked through the tasks recorded a higher time than those who did not. Perhaps the most surprising thing from the discovery stage is that it took longer to find "Privacy Settings" than anticipated. This is owing to some participants' expectation of the settings wheel icon to be on the right of the calendar or their disagreement with changing the visibility of the medication from Settings. Generally, participants tried other options before resorting to the settings icon. Worth noting also is the observation that medication-related features posted a slightly higher task discovery time because participants had no prior exposure to many of the features. They were seeing them for the first time and had to reason about the ideal location of the feature under question. All the features were thus easy to find (albeit with varying time). The feature discovery trends do however indicate that new calendar features that take advantage of similar existing placements are easily findable.

8.5.2 Managing Medication Entries

Calendar users would not find the task of managing medication entries any more demanding than what they are already used to. According to the results, adding an event entry to the calendar took as long as adding a medication entry. Editing and deleting medication and event entries also had the same time demands. Other medication-related tasks took less time than updating a normal event, proving that managing a medication entry is only as time-consuming as managing a regular event. Regarding new tasks, participants had to figure out how to add a medication entry to the calendar and how to schedule a medication. Participants not only completed the tasks quickly but also self-learned the implied meaning of both a medication entry and a medication schedule. To explain this, in calendars, event entries do not have parent events. You can create a repeating event but you cannot create a child event [33, 183]. This is not the case for medication entries. For medications, the medication itself is a parent, and medication entries (which make up a schedule) are child elements that exist as part of the medication. By design, the medication cannot be added by the user (except if/when the calendar supports user addition of Over the Counter (OTC) medication, otherwise the prescription is fetched from a CPOE system). The slow time posted by the two tasks, therefore, is not representative of the time of performing the task. Participants spent quite a large part of that time in trying to figure out how to perform that task. Once they did, the completion time beat even that of adding an event (a threefold reduction), which only posted a minor improvement during the second try (See figure 8.4).

In the design of calendars, the process of adding a medication entry and generating a schedule should follow after that of adding events. The default routine for almost all participants, when asked to perform the two tasks, was to try and create an event. To fit into the users' mental model, if a medication action involves creating an entity, then the functionality should be attached to the events' creation process. For calendars that already have Event, Appointment schedule, and Task as sub-menus of the event creation process, adding medication entries and creating medication schedules should be extensions of that menu. For the two processes to be seamless, the system should retrieve the list of medications from the calendar and allow the user to choose a medication for which they would like to create entries.

Adding and showing medications in the calendar is "not a privacy setting". That is, going by the opinions of some participants. Notwithstanding, medication privacy is an important aspect of the calendar. Users should be able to hide and show medication in the calendar (and conversely show medication and hide other events). Users should have control over the visibility of both summaries and medication entries within the calendar. The visibility control feature should be easily accessible, allowing the user to switch between views with ease. Since the sharing of calendars between users is supported by popular calendars, users should be able to control the privacy of the medication in the shared calendars.

8.5.3 Managing Conflicts

Conflicts between medication entries should be rendered in a way that draws user attention. While conflict warning using triangles with exclamation mark insets was effective in communicating the presence of conflicts, a dedicated color and highly visible warning icons would achieve better results. As one participant had put it, *"Medication is a critical thing. Something should pop up saying you were doing this the wrong way."* On-calendar warning annotations should therefore be accompanied by an emphasis that will draw the user's attention to the warning. We found that popup messages are effective in communicating the nature of the conflict. These should be hidden by default and only become visible when the user interacts with the warning.

Users should be helped in resolving the flagged conflicts. A suggestion of time slots where conflicting entries should be rescheduled would help the users by providing guidance on the suggested entry placements. The calendar prototype was designed with the capability to highlight slots outside of which the entries should be moved. The highlighting used was however deemed too faint to be noticed by the users. Hence, a more pronounced annotation suggesting the valid time slots would be highly effective. We found that arrows are only slightly useful in communicating the direction in which to move the conflicting entries in order to resolve the conflict. The user's computation of time based on the minimum time requirement indicated by the error was more effective in this aspect.

From the system's point of view, conflict detection does not introduce a significant processing overhead that results in slower processing time of medication fetch requests. The observed times when handling requests involving multiple conflicts and those without conflicts were almost indistinguishable. All recorded times were below 1 second, which according to Nielsen is the limit for the user's flow of thought to stay uninterrupted, even with noticeable delay [237].

8.5.4 Self-reflecting

Research has already determined that self-reflection has a positive impact on adherence and that there is strong patient support for the usefulness of self-reflection tools [220, 226, 1]. Our study did not seek to determine whether self-reflection impacts adherence. Instead, the concentration was on determining the level of user engagement with charts presenting personal medication-taking history. Notwithstanding, participants from our study found the charts very useful. Participants also indicated that the charts would challenge them to maintain or improve their adherence levels.

Four different views were presented to participants in the study. From the results, it can be stated that users are likely to engage more with the self-tracking feedback presented within the calendar than with the specialized charts rendered outside calendars. Past medication statuses drew more exploration and insight generation than individual charts. From the statuses, insights such as details, comparison, summary, trend, and outliers were observed. Insights for trends and summaries were less accurate when derived using the calendar view. Different levels of reflection such as description, reflective, and dialogic were observed from the user's engagement with the calendar view. The charts, considered together, provide the most accurate trend, summary, correlation, and outlier insights. Participant engagement with the charts was however superficial, with few participants going as far as exploring all the different views available for each chart. Hence, while the charts were built to support advanced levels of reflection such as transformative reflection and critical reflection, observed interactions were limited to Reflective reflection. Few participants attempted to explore relationships between events and medication history.

The charts, although they proved difficult to interpret for many participants, especially when asked to reason about the wider implications, provided a quicker way of reading overall summaries. The Overall chart was the easiest to read. The weekly aggregation (using a bar chart) was more effective in communicating interval patterns. The Weekly chart had the highest engagement levels of the three charts, with users observed clicking, mousing over, refining the display, and changing views multiple times. The Timeline chart had the least engagement, attaining the label of most difficult (and most comprehensive). The word cloud was effective in communicating outcomes and reasons attached to medication entries through notes.

Self-tracking of medication taking should be a design feature of calendars integrating medication management. Self-tracking data should be reflected within the calendar as a first level of access to self-reflection. The calendar should be designed to show the statuses of past medication entries. Easily accessible forms of summaries should be provided within the calendar. This can be done by showing the ratio of the taken medication to the scheduled medication as part of the weekly summaries. Details charts are still important for extended reflection. Although limited engagement was observed for the charts, they proved effective in communicating overall, consolidated, and detailed medication-taking summaries and should therefore be considered for addition as part of medication management. The charts employed should be easy to read for naive calendar users. Advanced charts should also be incorporated for users who would want further exploration.

8.6 Conclusion and Recommendations

Calendars play a pivotal role in the effective management of day-to-day activities. While most digital calendars are primarily designed for event management, their potential surpasses this function. They can serve as powerful tools to facilitate the self-management of medication by patients. This chapter presented a comprehensive study evaluating a calendar integrated with medication management functionalities. Participants were engaged in tasks encompassing both event and medication management, shedding light on the competency required to manage medications using calendars. Although some medication-related tasks took slightly longer than familiar calendar tasks, participants leveraged their familiarity with calendars to effectively complete most new tasks. They demonstrated efficient management of medications and adept handling of conflicts arising from the refinement of medication schedules. However, the self-reflection aspect of the calendar posed challenges to some participants. Despite this, they were able to derive insights and apply various reflection levels using the calendar and the accompanying history charts.

The evaluation of the calendar was conducted over a one-hour period, during which participants were introduced to the calendar for the first time. It's worth noting that extended usage of the proposed features holds promise for enhanced performance and adoption, as indicated by participants' feedback and their adeptness at using the features during the study. An ideal evaluation would involve deploying the calendar in a live environment and conducting diary studies to assess whether the calendar effectively improves medication adherence among patients managing their medications amidst busy schedules. Additionally, conducting pilot studies in various setups with a larger participant pool would provide valuable insights and further validate the calendar's usefulness and impact.

Chapter 9

Discussion

9.1 Background

Non-adherence to prescribed medication is a complex issue involving patients, physicians, and healthcare systems, each contributing to the problem in various ways. To address the problem of non-adherence, a holistic approach should be taken that involves these key stakeholders. Factors affecting non-adherence and interventions have been identified for each of the three stakeholders. Many interventions are targeted at patients. Yet, it is important to note that patient-related factors are somehow intertwined with physician and health system-related factors. Prescription of complex drug regimens, communication barriers, ineffective communication of information about adverse effects, and provision of care, as an example, are listed among factors that contribute to non-adherence [6]. These factors fall within the jurisdiction of the physicians. As challenging as improving medication adherence is, physicians can take steps to address these factors and help improve patients' medication-taking behaviour and, subsequently, outcomes. Technologies intended to enhance patient adherence can be more effective if contributing factors from the physician's side are dealt with beforehand. The same applies to health system factors. Factors such as limitation of visitation time, limited access to care, and lack of health information technology can influence the outcome of non-adherence interventions on the patients [158]. Hence, when physicians and health-care-system-related factors are addressed first, a reasonable stage for introducing effective non-adherence interventions tailored to patients is set.

Notwithstanding all the contributing factors already highlighted, medication adherence is primarily in the domain of the patient [3]. Patient-related factors that affect adherence include literacy levels, socioeconomic status, and psychological factors [3]. Patient education and pictorial and audio-visual education materials have effectively addressed literacy-related factors [238]. Actively involving patients in treatment decisions (such as helping patients choose cheaper alternatives when they can't afford the cost of medication) helps alleviate, to some extent, socioeconomic factors. These two, as are physiological factors, are not entirely up to the patient.

9.1.1 Research Scoping

My research addressed non-adherence under physiological factors (intentional and unintentional omission [15]). I considered the challenges associated with integrating prescribed medications into patients' demanding and active lifestyles. Under this setup, patients are willing to adhere to their prescribed medication regimens yet encounter difficulties maintaining consistency due to the dynamic nature of their schedules. The encompassing factor is forgetfulness, a leading contributor to non-adherence [15]. Forgetfulness applies to some extent, particularly in situations where a hectic schedule leads to occasional lapses in medication taking. However, it's also relevant to include individuals who consistently remember to take their medication but struggle to find an optimal scheduling time. Past research leans towards using reminders to solve the problem of forgetfulness.

9.1.2 Don't forget to take your medication!

Reminders are among the most common technological interventions to improve medication adherence [37]. They are designed to remind users to take their medications at set intervals. Using reminders, patients can program times when alarms should be triggered to remind them to take their medication. When the problem is forgetfulness, reminders are effective to varying extents [81]. Reminders are not an optimal solution for the problem of non-adherence resulting from busy schedules. Triggering medication reminders when the user is busy is itself a limitation. Reminders are also not designed to be aware of the temporal constraints associated with medication. Hence, reprogramming a reminder would not warn the user about possible violation of temporal constraints. Reminders are also operated independently of applications used to manage daily schedules. This makes their effectiveness in helping manage medication alongside other activities inadequate. Integrated solutions that bundle events and medication management together could do more to address these shortcomings. Enhancing existing activity management tools can be achieved by incorporating medication management features.

9.1.3 Integrated Planning

Calendars already serve as pivotal tools for effectively managing day-to-day activities, as established by Mennicken et al. [24]. They find extensive use among individuals, families, and organizations to coordinate diverse facets of daily engagements. Moreover, calendars function as instrumental aids for scheduling health-related reminders and overseeing clinical appointments for specific users. Incorporating medication management within the calendar interface presents a notable augmentation, addressing a vital need for a substantial portion of the patient population. The strategy employed here is that of empowerment. The more empowered patients feel, the more likely they will be motivated to manage their disease and adhere to their medications [6].

Three studies were conducted to assess the viability of integrating medication management within calendars that covered the design and evaluation of a calendar-based medication management system. Within two of these studies, a review was performed to determine participants' inclination towards adopting a calendar for medication management. The results yielded a primarily favourable response, with approximately 68% of participants in the first user study and 80% in the second expressing a positive disposition towards the integrated approach. Notably, users prescribed more than two medications were more inclined to utilize a calendar for medication management than those with fewer medications. This observation is rationalized by the ease of establishing a consistent medication routine that falls outside the ambit of their general "active window." This was also noted during the first study.

9.2 The Ideal Medication Manager

The study path employed for CHAOS involved developing and prototyping a novel calendar system. However, the bigger picture of this endeavour extends beyond creating a standalone application. Instead, the studies have led to the formulation of insightful design guidelines. These guidelines present an adaptable framework that established calendars, such as Google Calendar and Outlook Calendar, can readily incorporate to integrate medication management seamlessly. This strategy is essential to help avert user reluctance to adopt an entirely new calendar application to manage their daily affairs instead of what they use.

By embracing the proposed design guidelines, mainstream calendars can transform into health support tools by seamlessly incorporating medication management features. This approach aligns with the objective of enhancing user engagement and utility, making the integration process conducive to widespread acceptance and usage. If calendars are effective in making appointments and setting up events by users from diverse populations, they can be just as effective in helping manage medication for affected users. Calendars are, therefore, the ideal medication managers.

Integrating medication management in calendars involves three significant considerations. These are (i) how to model medication data, (ii) how to visually represent medication entries on the calendar, and (iii) which additional support features to include as part of medication management.

9.2.1 Modelling Medication Data

In the initial study, we established that the calendar should accommodate essential medication details like name, dosage, frequency, duration, and relevant indications for effective medication support. We utilized the Simple Temporal Problem (STP) framework to represent prescriptions, incorporating frequency, duration, and other pertinent information. The formalization process conceptualized a prescription as a two-part statement comprising requirements and restrictions. Requirements addressed medication administration routines. Restrictions addressed contra-indications related to the medication. This formalization was successful. A prescription can be represented as a temporal problem with support for queries such as consistency checking and schedule generation.

In calendars, data is stored in a database, and operations are conducted to store, read, and display events for users. Existing calendars lack a constraint-checking mechanism that is pertinent to medication. To enable medication management, the formal representation of prescriptions needs to be restructured to ensure compatibility with database support while supporting all medication-related queries.

In the prototyping phase, a strategic choice was made to source prescriptions from a regulated Computerized Provider Order Entry (CPOE) system. This decision initially eliminated the necessity for prescription entry support within the calendars.

However, feedback from user studies indicated a desire for a feature enabling users to add unregulated Over-The-Counter (OTC) medications, prompting a reconsideration of the initial approach. Whether sourced or entered would determine what information is stored in the database. Sourcing information from a CPOE system eliminates the need to keep a full copy of the prescription data in the database. If user entry is allowed, the information can be stored as one entity without decomposing the individual attributes.

Different STP concepts can be used at various stages of the integration pipeline. With the prescription as an input, the system can use the STP concept to generate the default schedule. One would argue that schedule generation is not a complex matter. But this is different, mainly when the prescription contains temporal delineation classes such as morning, afternoon, evening, before bed, at night, etc. The STP effectively represented the delineation in a manner that simplified schedule generation. After the schedule is generated, it can be stored in a database that is queried for display on the calendar. Medication information such as name, dosage, and other indications should be stored separately.

Another stage when the STP concept can be employed is checking for conflicts after schedule refinements. Using contra-indications from medication data, an STP-based minimal network can be generated to represent medication interactions upon each schedule retrieval request. The table should contain information about all conflicting entries included in the request. The prototype's minimal table construction was confined to the requested week. However, all entries within the month should be included in a monthly calendar, potentially leading to a slightly longer table generation time. All entries in the schedule were cross-checked for conflicts using this minimal table. Any information about conflicting entries was then relayed to the calendar rendering functions. The employed process in the prototype proved efficient without causing notable delays for requests involving constraint checking. Once the medication details are stored in the database, the remaining task is determining the optimal rendering mechanism of medication data on the calendar.

To summarize, our study confirmed the high level of expressiveness exhibited by the Simple Temporal Networks (STP) in modeling prescription data. This capacity to efficiently represent repetitions and periodicity within clinical guidelines aligns with the findings of Anselma et al. [103]. Furthermore, our research established the effectiveness of STP in representing temporal constraints associated with multiple medications.

9.2.2 Visualizing Medication Data

We explored various design options to represent medication entries on the calendar visually. The studies underscored the need for a distinct visualization of medication entries, ensuring they stand out from other events without cluttering the design. As the number of medications increases, the size of entries should decrease to maintain a balanced and unobtrusive appearance on the calendar. Users prefer medication entries to be discreet and not overshadow other event entries. Therefore, the design should strike a delicate balance between subtlety and visibility.

Familiar Designs

Employing familiar symbols like medication icons and images to depict medication entries is preferred over arbitrary designs. Brown et al. also emphasized this design principle to address adherence problems associated with poor literacy levels [6]. Additionally, the designs should allow users to customize the appearance of medication entries according to their preferences.

Conflict Rendering

Another aspect to consider in the design of medication entries is the representation of conflicts between them. Error symbols like the exclamation mark or warning triangle are commonly used to indicate conflicts. User feedback suggests that the colour employed to signify conflicts should not align with the calendar's theme colour. In cases of interaction, the conflicting entries should display available time slots for potential relocation to resolve the conflict.

The prototype's text message indicated when the conflicting entry was positioned. However, an optimal design should incorporate a visual annotation that links conflicting entries clearly. During the design phase, arrows were utilized to establish these connections between conflicting pairs. Although technological limitations hindered their implementation, they effectively illustrated conflicting pairs.

9.2.3 Additional Calendar Features

The calendar's functionality should encompass features tailored to medication entries, such as privacy, self-reflection, and collaboration.

Privacy and Collaboration

Privacy is an essential component of effective medication management. When it comes to shared calendars, ensuring that sharing does not compromise an individual's medication details is important. The calendar platform should go the extra mile in enabling controlled sharing, putting the power in the users' hands to determine the extent of access to medication information within the shared calendar.

This level of control not only guarantees privacy but also fosters collaboration—an essential element in promoting adherence. Research strongly underscores the positive impact of patients receiving psychological support during their medication regimen on adherence rates [239]. Consequently, patients might willingly choose to share their medication calendar with these invaluable supporters. The access granted should align with defined specifications for query moderation, allowing the patient to tailor the level of access granted to the authorized party.

Another important aspect of privacy pertains to controlling the visibility of medication entries within the calendar. Users should have the seamless ability to toggle the medication display, putting them in the driver's seat regarding what's visible. The sheer presence of medication entries on the calendar can sometimes be overwhelming, and providing an option to hide them ensures that the default view resembles a standard event calendar. Furthermore, users should be able to streamline their view by deactivating all non-medication-related events, enabling them to concentrate solely on medication entries for effective management. The feature allowing control over visibility should be effortlessly accessible, requiring just a single click to toggle the display according to the user's preferences.

Research in the domain of healthcare privacy aligns with the findings of this study. Wilkowska et al. emphasized the significance of privacy in the effective integration of medical assistive technologies within home environments [240]. Similarly, Fox highlighted the role of perceived data vulnerability in the adoption of mobile health solutions, stating that users are less likely to adopt such solutions if they perceive a risk of sensitive data exposure [241]. By effectively addressing privacy concerns within the calendar system, this research not only builds user confidence but also greatly enhances medication management by fostering a sense of ownership and encouraging collaboration.

Self-Reflection

Enabling self-reflection is paramount within the calendar system, empowering users to actively monitor and enhance their medication adherence. Incorporating specific statuses associated with medication entries, such as *taken* and *missed*, marked by easily recognizable symbols like a green checkmark and a red cross, significantly simplifies user interaction. This intuitive visual feedback minimizes the need for extensive user education.

Moreover, expanding the range of statuses to include *pending*, *unsure*, and *deliberately missed* offers users a reasonable toolkit for accurately documenting their medication experiences. This not only enables immediate feedback but also allows users to incorporate supplementary information essential for self-reflection. By providing users with diverse mechanisms like simple bar charts, gauge charts, and timeline charts, showcasing their medication-taking history, the calendar empowers users to assess their adherence levels effectively.

This approach not only aids users in tracking their medication intake but also cultivates a sense of accountability and responsibility, ultimately leading to improved adherence rates. Additionally, the simplicity and user-centric design of the system ensure a seamless and gratifying experience, encouraging consistent engagement with medication management through the calendar interface.

Self-Optimization

Enabling users to rearrange medication timings within the calendar based on their availability is a foundational step toward self-optimization. During the self-reflection process, users can recognize patterns of missed medication, prompting them to modify the calendar and improve the schedule for better adherence and outcomes. This iterative process, known as self-optimization, empowers users to refine their medication schedule based on insights gained from reflection.

The optimal realization of self-optimization involves the system learning from user behavior and autonomously adjusting the schedule while adhering to predefined constraints and leveraging observed trends. In this scenario, users are relieved from the manual analysis of data to reorganize the schedule; the system handles this task. Although automated self-optimization was not incorporated into the current prototype, we strongly advocate for its integration as a valuable feature in medication calendars.

9.3 Conclusion

In conclusion, advocating for the integration of medication management within calendars is crucial. This functionality should be deemed as essential as event management and appointment scheduling, which are already established features. While making a definitive claim about the improvement in adherence requires comprehensive studies, the proven efficacy of reminders in reducing forgetfulness supports the potential of calendars to enhance adherence. By implementing calendars with features like self-reflection and consistency checks, we can mitigate medication errors, aid in health tracking, and facilitate seamless communication with healthcare professionals. Extensive design guidelines proposed in the literature and our additional recommendations ensure a user-centric approach that emphasizes familiarity, declutters the interface, allows personalization, and promotes privacy. Moreover, embracing concepts like self-optimization and checking for unsafe drug interactions within temporal constraints adds further value to the integration of medication management within calendars.

Chapter 10

Future Research

Reviewing the calendar's design, several envisaged features were only partially realized during the implementation phase. In the iStar model discussed in Chapter 3, we introduced four user-based roles and two system-based roles. The user roles were Patient, Social Peer, Healthcare Provider, and Calendar User. The system roles were the Calendar App (CA) and Calendar and Medication App (CMA). During our studies, our main focus was on achieving the goals related to the system roles and mainly supporting the Patient role. We managed to accomplish most goals linked to CA and CMA, like ensuring safe schedule updates and maintaining a secure medication schedule. However, we didn't fully achieve some goals, such as allowing users to share prescriptions with others. These pending goals are part of our suggested future work, key of which I will discuss further below.

Self-Optimization

The achievement of optimal therapy is one of the goals that were part of the patient's role in the CHAOS model.The operational calendar allows users to refine their schedules based on insights derived from their self-tracked data. However, this optimization process may pose challenges, particularly for less technically inclined users who might need help interpreting the self-tracked data. The system should proactively propose feasible schedules based on observed trends to enhance usability. For instance, if the system identifies a consistent pattern of missed doses linked to specific events, it should recommend alternative time slots for administering that medication while ensuring the absence of unsafe drug interactions.

Responsive Design

The calendar was designed to work on desktops. Further considerations should be taken to ensure the calendar's compatibility and accessibility across various platforms (including mobile) to cater to users' diverse needs and preferences, enabling seamless usage irrespective of the device being used. Another limitation was the view. The calendar was only designed to work with a weekly view. But calendars are intended to be toggled, with monthly and daily views being the other options. Studies should be conducted on how medication entries and conflict annotations should be rendered on these alternative views.

Extended Testing

The studies conducted were on a limited number of participants and for a limited testing period. Extensive usability studies should be conducted that involve a diverse user group to gather comprehensive feedback on the calendar's functionality, design, and user-friendliness. This will aid in identifying areas for improvement and fine-tuning the user interface. The calendar should also be tested within a live environment where participating users are given an extended period to use the calendar to manage their medications. This extended usability testing may provide insights into the viability of calendars to enhance adherence and improve outcomes (such as results, symptoms, and side effects). Self-optimization beyond calendars is another dimension for future research.

Collaboration

Two user-based roles, Healthcare provider and Social peer, form the collaborative aspect of CHAOS. Although conceptualized (See figure 3.1), the calendar's collaborative feature remains uncompleted and untested. This feature was intended to enable users to share their calendars with other stakeholders, including healthcare providers or informal caregivers and also give the other stakeholders feature-full access to the system. In the existing implementation, once shared, the secondary user was granted full access to the calendar. However, the system should be enhanced for future extensions to allow the primary user to define and give selective access levels. Moreover, considering user recommendations, the system should facilitate medication calendar sharing. Unlike the broader access level (requiring user login to view the calendar), this specific sharing option would grant other users view-only permissions, ensuring they can observe the medication schedule without making alterations. These additions would significantly enhance the collaborative capabilities of the calendar, aligning with users' needs and preferences for controlled and nuanced sharing options. Although part of the CHAOS model, most of the functionalities related to Healthcare Providers, such as determining condition and choosing intervention, are outside the scope of the system implementation. Assessing outcomes, on the other hand, is a functionality that is key to the system and should therefore be carefully considered for incorporation into the system.

These represent the primary recommendations, accompanied by minor suggestions for future enhancement. Additionally, attention to refining the User Interface, enhancing conflict annotations, and incorporating imagery as medication entry icons should be considered for further improvement.

Chapter 11

Conclusion

The work presented in this dissertation centred on enhancing medication adherence by seamlessly integrating prescriptions into patients' daily routines using digital calendars. The research had three main objectives: (i) To study the adherence Care Plan, identify parts of the plan that can be represented using formal methods, and formulate a formal specification for the plan representing multiple prescriptions coming from multiple providers; (ii) To develop methods that would be used to enhance adherence using the proposed formal specification; and (iii) To design, Implement, and test collaborative tools, employing Information Visualization, that would help patients with medication management. The calendar was selected as the collaborative tool of choice. Three overlapping studies were conducted to achieve the objectives. The studies covered prescription formalization, calendar design, and calendar implementation.

The initial study focused on formalizing prescriptions for efficient integration into medication management tools. We explored various frameworks, ultimately adopting the Simple Temporal Problem (STP) due to its aptitude for representing temporal aspects of prescriptions. An STP network was implemented, allowing the creation and management of multiple prescriptions. Performance testing demonstrated the system's efficiency in generating schedules and checking for consistency, which is crucial for safe medication management.

In the second study, we designed calendar prototypes to support medication management. Three distinct prototypes were crafted, each emphasizing layout, visual representation of medication entries, and conflict display. Through rigorous evaluation with participants, we identified essential design guidelines such as familiarity, clutter reduction, personalization options, and effective highlighting for user attention. Users showed a favourable inclination towards utilizing such a calendar for medication management.

The third study involved implementing a calendar seamlessly integrating events and medication management. Utilizing the FullCalendar framework, a prototype was developed with efficient scheduling of medications, conflict resolution, and self-reflection features. Usability evaluation demonstrated users' ability to proficiently navigate medication-related tasks within the calendar, although self-reflection posed challenges. The study showcased the potential of integrating medication management into standard calendar interfaces.

The collective findings from these studies offer significant insights into the formalization of prescriptions, efficient calendar interface design, and the seamless integration of medication management into daily calendar systems. Being widely adopted, Calendars have substantial potential to enhance medication adherence by aiding patients in managing their medication schedules. The results affirm that patients are open to utilizing calendars as their preferred tool for medication management. Additionally, integrating crucial features like conflict resolution and self-reflection directly into calendars can further enhance their effectiveness in supporting medication adherence.

Bibliography

[1] Matthew L Lee and Anind K Dey. Real-time feedback for improving medication taking. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 2259–2268, 2014.

[2] Rafal Kocielnik, Lillian Xiao, Daniel Avrahami, and Gary Hsieh. Reflection companion: a conversational system for engaging users in reflection on physical activity. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 2(2):1–26, 2018.

[3] Eduardo Sabaté. *Adherence to long-term therapies: evidence for action*. World Health Organization, 2003.

[4] Jeffrey K Aronson. Medication errors: definitions and classification. *British journal of clinical pharmacology*, 67(6):599–604, 2009.

[5] Patrick J McDonnell and Michael R Jacobs. Hospital admissions resulting from preventable adverse drug reactions. *Annals of Pharmacotherapy*, 36(9):1331–1336, 2002.

[6] Marie T Brown and Jennifer K Bussell. Medication adherence: Who cares? In *Mayo clinic proceedings*, volume 86, pages 304–314. Elsevier, 2011.

[7] Lars Osterberg and Terrence Blaschke. Adherence to medication. *New England journal of medicine*, 353(5):487–497, 2005.

[8] Amanda J Cross, Rohan A Elliott, Kate Petrie, Lisha Kuruvilla, and Johnson George. Interventions for improving medication-taking ability and adherence in older adults prescribed multiple medications. *Cochrane Database of Systematic Reviews*, 5, 2020.

[9] Caroline A Walsh, Caitriona Cahir, Sarah Tecklenborg, Catherine Byrne, Michael A Culbertson, and Kathleen E Bennett. The association between medication non-adherence and adverse health outcomes in ageing populations: a systematic review and meta-analysis. *British journal of clinical pharmacology*, 85(11):2464–2478, 2019.

[10] Steve Offord, Jay Lin, Dario Mirski, and Bruce Wong. Impact of early nonadherence to oral antipsychotics on clinical and economic outcomes among patients with schizophrenia. *Advances in therapy*, 30:286–297, 2013.

[11] Richard Long. Drug-resistant tuberculosis. *Cmaj*, 163(4):425–428, 2000.

[12] Rachelle Louise Cutler, Fernando Fernandez-Llimos, Michael Frommer, Charlie Benrimoj, and Victoria Garcia-Cardenas. Economic impact of medication non-adherence by disease groups: a systematic review. *BMJ open*, 8(1):e016982, 2018.

[13] Kwadwo Osei Bonsu, Stephanie Young, Tiffany Lee, Hai Nguyen, and Rufaro S Chitsike. Adherence to antithrombotic therapy for patients attending a multidisciplinary thrombosis service in canada–a cross-sectional survey. *Patient preference and adherence*, pages 1771–1780, 2022.

[14] Konstadina Griva, Andrew Davenport, Michael Harrison, and Stanton P Newman. Non-adherence to immunosuppressive medications in kidney transplantation: intent vs. forgetfulness and clinical markers of medication intake. *Annals of Behavioral Medicine*, 44(1):85–93, 2012.

[15] Elizabeth J Unni and Karen B Farris. Unintentional non-adherence and belief in medicines in older adults. *Patient education and counseling*, 83(2):265–268, 2011.

[16] Marie T. Brown, Jennifer Bussell, Suparna Dutta, Katherine Davis, Shelby Strong, and Suja Mathew. Medication Adherence: Truth and Consequences. *American Journal of the Medical Sciences*, 351(4):387–399, 2016.

[17] Nafis Faizi and Shahwar Kazmi. Universal health coverage-there is more to it than meets the eye. *Journal of family medicine and primary care*, 6(1):169, 2017.

[18] Jia-Rong Wu, Debra K Moser, Terry A Lennie, and Patricia V Burkhart. Medication adherence in patients who have heart failure: a review of the literature. *Nursing Clinics of North America*, 43(1):133–153, 2008.

[19] Lars Osterberg and Terrence Blaschke. Adherence to medication. *New England journal of medicine*, 353(5):487–497, 2005.

[20] Marcia Vervloet, Annemiek J Linn, Julia CM van Weert, Dinny H De Bakker, Marcel L Bouvy, and Liset Van Dijk. The effectiveness of interventions using electronic reminders to improve adherence to chronic medication: a systematic review of the literature. *Journal of the American Medical Informatics Association*, 19(5):696–704, 2012.

[21] Vinay Kini and P Michael Ho. Interventions to improve medication adherence: a review. *Jama*, 320(23):2461–2473, 2018.

[22] Marya E Corden, Ellen M Koucky, Christopher Brenner, Hannah L Palac, Adisa Soren, Mark Begale, Bernice Ruo, Susan M Kaiser, Jenna Duffecy, and David C Mohr. Medlink: A mobile intervention to improve medication adherence and processes of care for treatment of depression in general medicine. *Digital Health*, 2:2055207616663069, 2016.

[23] Katarina Tabi, Abnashi Singh Randhawa, Fiona Choi, Zamina Mithani, Friederike Albers, Maren Schnieder, Mohammadali Nikoo, Daniel Vigo, Kerry Jang, Regina Demlova, et al. Mobile apps for medication management: review and analysis. *JMIR mHealth and uHealth*, 7(9):e13608, 2019.

[24] Sarah Mennicken, David Kim, and Elaine May Huang. Integrating the smart home into the digital calendar. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, pages 5958–5969, 2016.

[25] James F Allen. Maintaining knowledge about temporal intervals. *Communications of the ACM*, 26(11):832–843, 1983.

[26] Marc Vilain, Henry Kautz, and Peter Van Beek. Constraint propagation algorithms for temporal reasoning: A revised report. In *Readings in qualitative reasoning about physical systems*, pages 373–381. Elsevier, 1990.

[27] Jitendra Malik and Thomas O Binford. Reasoning in time and space. In *IJCAI*, volume 83, pages 343–345, 1983.

[28] Rina Dechter, Itay Meiri, and Judea Pearl. Temporal constraint networks. *Artificial intelligence*, 49(1-3):61–95, 1991.

[29] Neil Yorke-Smith, Kristen Brent Venable, and Francesca Rossi. Temporal reasoning with preferences and uncertainty. In *IJCAI*, volume 3, pages 1385–1386, 2003.

[30] Kostas Stergiou and Manolis Koubarakis. Backtracking algorithms for disjunctions of temporal constraints. *Artificial Intelligence*, 120(1):81–117, 2000.

[31] Kristen Brent Venable and Neil Yorke-Smith. Disjunctive temporal problems with uncertainty. In *Workshop Proceedings June 20–22, 2005 Uppsala, Sweden*, page 178. Citeseer, 2005.

[32] FullCalendar LLC. Fullcalendar javascript library. https://fullcalendar.io/, 2023. Accessed: 2022-07-12.

[33] Google. Google calendar. https://calendar.google.com/calendar, 2022. Accessed: 2022-05-04.

[34] Alex J Adams and Samuel F Stolpe. Defining and measuring primary medication nonadherence: development of a quality measure. *Journal of managed care & specialty pharmacy*, 22(5):516–523, 2016.

[35] Jacqueline G Hugtenburg, Lonneke Timmers, Petra JM Elders, Marcia Vervloet, and Liset van Dijk. Definitions, variants, and causes of nonadherence with medication: a challenge for tailored interventions. *Patient preference and adherence*, pages 675–682, 2013.

[36] Beena Jimmy and Jimmy Jose. Patient medication adherence: measures in daily practice. *Oman medical journal*, 26(3):155, 2011.

[37] Robby Nieuwlaat, Nancy Wilczynski, Tamara Navarro, Nicholas Hobson, Rebecca Jeffery, Arun Keepanasseril, Thomas Agoritsas, Niraj Mistry, Alfonso Iorio, Susan Jack, et al. Interventions for enhancing medication adherence. *Cochrane database of systematic reviews*, 11, 2014.

[38] Canadian Pharmacists Association. A national survey of canadians on their perceptions and attitudes towards pharmacists in canada.

https://www.pharmacists.ca/advocacy/advocacy-activities/medication-adherence/, 2023. Accessed: 2023-02-01.

[39] San Kim, Kathleen Bennett, Emma Wallace, Tom Fahey, and Caitriona Cahir. Measuring medication adherence in older community-dwelling patients with multimorbidity. *European journal of clinical pharmacology*, 74:357–364, 2018.

[40] Kang-Ting Tsai, Jen-Hau Chen, Chiung-Jung Wen, Hsu-Ko Kuo, I-Shu Lu, Lee-Shu Chiu, Shwu-Chong Wu, and Ding-Cheng Chan. Medication adherence among geriatric outpatients prescribed multiple medications. *The American journal of geriatric pharmacotherapy*, 10(1):61–68, 2012.

[41] Seung Jae Kim, Oh Deog Kwon, Eunice Bormee Han, Cheol Min Lee, Seung-Won Oh, Hee-Kyung Joh, Bumjo Oh, Hyuktae Kwon, BeLong Cho, and Ho Chun Choi. Impact of number of medications and age on adherence to antihypertensive medications: a nationwide population-based study. *Medicine*, 98(49), 2019.

[42] Johnson George, Rohan A Elliott, and Derek C Stewart. A systematic review of interventions to improve medication taking in elderly patients prescribed multiple medications. *Drugs & aging*, 25:307–324, 2008.

[43] Marie T Brown, Jennifer Bussell, Suparna Dutta, Katherine Davis, Shelby Strong, and Suja Mathew. Medication adherence: truth and consequences. *The American journal of the medical sciences*, 351(4):387–399, 2016.

[44] Assumpta Ann Ryan. Medication compliance and older people: a review of the literature. *International journal of nursing studies*, 36(2):153–162, 1999.

[45] R Brian Haynes, Heather P McDonald, and Amit X Garg. Helping patients follow prescribed treatment: clinical applications. *Jama*, 288(22):2880–2883, 2002.

[46] DK Theo Raynor. Medication literacy is a 2-way street. In *Mayo Clinic Proceedings*, volume 83, pages 520–522. Elsevier, 2008.

[47] Pranaya Mishra, Ebba H Hansen, Svend Sabroe, and KK Kafle. Socio-economic status and adherence to tuberculosis treatment: a case-control study in a district of nepal. *The international journal of tuberculosis and lung disease*, 9(10):1134–1139, 2005.

[48] R Brunner, Jacqueline Dunbar-Jacob, Meryl S LeBoff, I Granek, D Bowen, Linda G Snetselaar, Sally A Shumaker, J Ockene, M Rosal, Jean Wactawski-Wende, et al. Predictors of adherence in the women's health initiative calcium and vitamin d trial. *Behavioral Medicine*, 34(4):145–155, 2009.

[49] Alessandra Graziottin, Jacqueline Gardner-Nix, Maik Stumpf, and Michael N Berliner. Opioids: how to improve compliance and adherence. *Pain Practice*, 11(6):574–581, 2011.

[50] Huang-Tz Ou, Steven R Feldman, and Rajesh Balkrishnan. Understanding and improving treatment adherence in pediatric patients. *, 29, 2*, 29(2):137–140, 2010.

[51] Robert Horne, John Weinman, and Maittew Hankins. The beliefs about medicines questionnaire: the development and evaluation of a new method for assessing the cognitive representation of medication. *Psychology and health*, 14(1):1–24, 1999.

[52] Janet A Butler, Robert C Peveler, Paul Roderick, Peter WF Smith, Robert Horne, and Juan C Mason. Modifiable risk factors for non-adherence to immunosuppressants in renal transplant recipients: a cross-sectional study. *Nephrology Dialysis Transplantation*, 19(12):3144–3149, 2004.

[53] Kathy K Byrd, John G Hou, Ron Hazen, Heather Kirkham, Sumihiro Suzuki, Patrick G Clay, Tim Bush, Nasima M Camp, Paul J Weidle, and Ambrose Delpino. Antiretroviral adherence level necessary for hiv viral suppression using real-world data. *Journal of acquired immune deficiency syndromes (1999)*, 82(3):245, 2019.

[54] Simon Diemert, Jens H Weber, and Morgan Price. An engagement model for medication management: From prescription to description and conscription. *ITCH*, pages 81–86, 2017.

[55] John C Walsh, Sundhiya Mandalia, and Brian G Gazzard. Responses to a 1 month self-report on adherence to antiretroviral therapy are consistent with electronic data and virological treatment outcome. *Aids*, 16(2):269–277, 2002.

[56] R Brian Haynes, DW Taylor, David L Sackett, Edward S Gibson, Charles D Bernholz, and Jayanti Mukherjee. Can simple clinical measurements detect patient noncompliance? *Hypertension*, 2(6):757–764, 1980.

[57] Mary B Neiheisel, Kathy J Wheeler, and Mary Ellen Roberts. Medication adherence part one: Understanding and assessing the problem. *Journal of the American Association of Nurse Practitioners*, 26(1):49–55, 2014.

[58] R Brian Haynes, Heather McDonald, Amit X Garg, Patty Montague, et al. Interventions for helping patients to follow prescriptions for medications. *Cochrane Database Syst Rev*, 2(11):7–9, 2002.

[59] Annette Zygmunt, Mark Olfson, Carol A Boyer, and David Mechanic. Interventions to improve medication adherence in schizophrenia. *American Journal of Psychiatry*, 159(10):1653–1664, 2002.

[60] Brianna Abbey, Anahita Alipour, Logan Gilmour, Christopher Camp, Crys Hofer, Robert Lederer, Greig Rasmussen, Lili Liu, Ioanis Nikolaidis, Eleni Stroulia, et al. A remotely programmable smart pillbox for enhancing medication adherence. In *2012 25th IEEE international symposium on computer-based medical systems (CBMS)*, pages 1–4. IEEE, 2012.

[61] Murtadha Aldeer and Richard P Martin. Medication adherence monitoring using modern technology. In *2017 IEEE 8th Annual Ubiquitous Computing, Electronics and Mobile Communication Conference (UEMCON)*, pages 491–497. IEEE, 2017.

[62] Diaa Salama Abdul Minaam and Mohamed Abd-ELfattah. Smart drugs: Improving healthcare using smart pill box for medicine reminder and monitoring system. *Future Computing and Informatics Journal*, 3(2):443–456, 2018.

[63] Huai-Kuei Wu, Chi-Ming Wong, Pang-Hsing Liu, Sheng-Po Peng, Xun-Cong Wang, Chih-Hi Lin, and Kuan-Hui Tu. A smart pill box with remind and consumption confirmation functions. In *2015 IEEE 4th Global Conference on Consumer Electronics (GCCE)*, pages 658–659. IEEE, 2015.

[64] Shashank Shinde, Tejas Kadaskar, Pushpak Patil, and Rohit Barathe. A smart pill box with remind and consumption using iot. *International Research Journal of Engineering and Technology*, 4(12):152–154, 2017.

[65] Hsiu-Ling Tsai, Chun Hsiang Tseng, Long-Cian Wang, and Fuh-Shyang Juang. Bidirectional smart pill box monitored through internet and receiving reminding message from remote relatives. In *2017 IEEE International Conference on Consumer Electronics-Taiwan (ICCE-TW)*, pages 393–394. IEEE, 2017.

[66] Alan Naditz. Medication compliance—helping patients through technology: Modern "smart" pillboxes keep memory-short patients on their medical regimen. *Telemedicine and e-Health*, 14(9):875–880, 2008.

[67] Morales Suárez-Varela MT et al. Study on the use of a smart pillbox to improve treatment compliance. *Atencion primaria*, 41(4):185–191, 2009.

[68] Roderick W Treskes, Enno T Van der Velde, Jan W Schoones, and Martin J Schalij. Implementation of smart technology to improve medication adherence in patients with cardiovascular disease: is it effective? *Expert review of medical devices*, 15(2):119–126, 2018.

[69] Marc I Rosen, Michael O Rigsby, Jamelah T Salahi, Caitlin E Ryan, and Joyce A Cramer. Electronic monitoring and counseling to improve medication adherence. *Behaviour research and therapy*, 42(4):409–422, 2004.

[70] Sandra van Dulmen, Emmy Sluijs, Liset Van Dijk, Denise de Ridder, Rob Heerdink, and Jozien Bensing. Patient adherence to medical treatment: a review of reviews. *BMC health services research*, 7(1):1–13, 2007.

[71] Vicki S Conn, Todd M Ruppar, Maithe Enriquez, and Pam Cooper. Medication adherence interventions that target subjects with adherence problems: systematic review and meta-analysis. *Research in Social and Administrative Pharmacy*, 12(2):218–246, 2016.

[72] Michal Gur, Vered Nir, Anna Teleshov, Ronen Bar-Yoseph, Eynav Manor, Gizelle Diab, and Lea Bentur. The use of telehealth (text messaging and video communications) in patients with cystic fibrosis: A pilot study. *Journal of Telemedicine and Telecare*, 23(4):489–493, 2017.

[73] Kerry Y Fang, Anthony J Maeder, Heidi Bjering, et al. Current trends in electronic medication reminders for self care. *Stud Health Technol Inform*, 231:31–41, 2016.

[74] Shama Mohammed, Rachel Glennerster, and Aamir J Khan. Impact of a daily sms medication reminder system on tuberculosis treatment outcomes: a randomized controlled trial. *PloS one*, 11(11):e0162944, 2016.

[75] Q Liu, K Abba, Alejandria Mm, D Sinclair, Balanag Vm, and Lansang Mad. Reminder systems to improve patient adherence to tuberculosis clinic appointments for diagnosis and treatment (Review) Summary Of Findings For The Main Comparison. *Cochrane Database Syst Rev.*, 11:1–59, 2014.

[76] Leah L Zullig, Felicia McCant, S Dee Melnyk, Susanne Danus, and Hayden B Bosworth. A health literacy pilot intervention to improve medication adherence using meducation® technology. *Patient education and counseling*, 95(2):288–291, 2014.

[77] Katarzyna Stawarz, Anna L Cox, and Ann Blandford. Don't forget your pill! designing effective medication reminder apps that support users' daily routines. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 2269–2278, 2014.

[78] Cristian Pop-Eleches, Harsha Thirumurthy, James P Habyarimana, Joshua G Zivin, Markus P Goldstein, Damien De Walque, Leslie Mackeen, Jessica Haberer, Sylvester Kimaiyo, John Sidle, et al. Mobile phone technologies improve adherence to antiretroviral treatment in a resource-limited setting: a randomized controlled trial of text message reminders. *AIDS (London, England)*, 25(6):825, 2011.

[79] Imran Ahmed, Niall Safir Ahmad, Shahnaz Ali, Shair Ali, Anju George, Hiba Saleem Danish, Encarl Uppal, James Soo, Mohammad H Mobasheri, Dominic King, et al. Medication adherence apps: review and content analysis. *JMIR mHealth and uHealth*, 6(3):e6432, 2018.

[80] Kristi Sidney, Jimmy Antony, Rashmi Rodrigues, Karthika Arumugam, Shubha Krishnamurthy, George D'souza, Ayesha De Costa, and Anita Shet. Supporting patient adherence to antiretrovirals using mobile phone reminders: patient responses from south india. *AIDS care*, 24(5):612–617, 2012.

[81] Sarah D Fenerty, Cameron West, Scott A Davis, Sebastian G Kaplan, and Steven R Feldman. The effect of reminder systems on patients' adherence to treatment. *Patient preference and adherence*, pages 127–135, 2012.

[82] Nancy Tran, Janet M Coffman, Kaharu Sumino, and Michael D Cabana. Patient reminder systems and asthma medication adherence: a systematic review. *Journal of Asthma*, 51(5):536–543, 2014.

[83] Barbara K Zedler, Priyanka Kakad, Susan Colilla, Lenn Murrelle, and Nirav R Shah. Does packaging with a calendar feature improve adherence to self-administered medication for long-term use? a systematic review. *Clinical therapeutics*, 33(1):62–73, 2011.

[84] Bee Ying Tan, Asrul Akmal Shafie, Mohamed Azmi Ahmad Hassali, and Fahad Saleem. Assessment of medication adherence and the costs associated with a calendar blister pack intervention among hypertensive patients in malaysia: a randomized controlled trial. *SAGE open medicine*, 5:2050312117709189, 2017.

[85] Philip J Schneider, John E Murphy, and Craig A Pedersen. Impact of medication packaging on adherence and treatment outcomes in older ambulatory patients. *Journal of the American Pharmacists Association*, 48(1):58–63, 2008.

[86] Logan Kendall, Jordan Eschler, Paula Lozano, Jennifer B McClure, Lisa M Vizer, James D Ralston, and Wanda Pratt. Engineering for reliability in at-home chronic disease management. In *AMIA Annual Symposium Proceedings*, volume 2014, page 777. American Medical Informatics Association, 2014.

[87] Katie A Siek, Danish U Khan, Stephen E Ross, Leah M Haverhals, Jane Meyers, and Steven R Cali. Designing a personal health application for older adults to manage medications: a comprehensive case study. *Journal of medical systems*, 35:1099–1121, 2011.

[88] Leah M Haverhals, Courtney A Lee, Katie A Siek, Carol A Darr, Sunny A Linnebur, J Mark Ruscin, and Stephen E Ross. Older adults with multi-morbidity: medication management processes and design implications for personal health applications. *Journal of medical Internet research*, 13(2):e1813, 2011.

[89] Hadi Kharrazi, Robin Chisholm, Dean VanNasdale, and Benjamin Thompson. Mobile personal health records: an evaluation of features and functionality. *International journal of medical informatics*, 81(9):579–593, 2012.

[90] Juan M Silva, Alain Mouttham, and Abdulmotaleb El Saddik. Ubimeds: a mobile application to improve accessibility and support medication adherence.

In *Proceedings of the 1st ACM SIGMM international workshop on Media studies and implementations that help improving access to disabled users*, pages 71–78, 2009.

[91] Danish U Khan, Katie A Siek, Jane Meyers, Leah M Haverhals, Steven Cali, and Stephen E Ross. Designing a personal health application for older adults to manage medications. In *Proceedings of the 1st ACM International Health Informatics Symposium*, pages 849–858, 2010.

[92] Madhukar H Trivedi, EH Lin, and Wayne J Katon. Consensus recommendations for improving adherence, self-management, and outcomes in patients with depression. *CNS Spectr*, 12(8 Suppl 13):1–27, 2007.

[93] Scott R Smith, John C Rublein, Cheryl Marcus, Tina Penick Brock, and Margaret A Chesney. A medication self-management program to improve adherence to hiv therapy regimens. *Patient education and counseling*, 50(2):187–199, 2003.

[94] Abu Sayeed Mondol, Ifat Afrin Emi, and John A Stankovic. Medrem: An interactive medication reminder and tracking system on wrist devices. In *2016 IEEE Wireless Health (WH)*, pages 1–8. IEEE, 2016.

[95] Leslie S Liu, Patrick C Shih, and Gillian R Hayes. Barriers to the adoption and use of personal health record systems. In *Proceedings of the 2011 iConference*, pages 363–370. ACM, 2011.

[96] Carman Neustaedter, AJ Bernheim Brush, and Saul Greenberg. The calendar is crucial: Coordination and awareness through the family calendar. *ACM Transactions on Computer-Human Interaction (TOCHI)*, 16(1):1–48, 2009.

[97] Dandan Huang, Melanie Tory, and Lyn Bartram. A Field Study of On-Calendar Visualizations. *arXiv preprint arXiv:1706.01123.*, 2017.

[98] Jordan Eschler, Logan Kendall, Kathleen O'Leary, Lisa M Vizer, Paula Lozano, Jennifer B McClure, Wanda Pratt, and James D Ralston. Shared calendars for home health management. In *Proceedings of the 18th acm conference on computer supported cooperative work & social computing*, pages 1277–1288, 2015.

[99] Jens H Weber, Simon Diemert, and Morgan Price. Using graph transformations for formalizing prescriptions and monitoring adherence. In *Graph Transforma-*

tion: 8th International Conference, ICGT 2015, Held as Part of STAF 2015, L'Aquila, Italy, July 21-23, 2015. Proceedings 8, pages 205–220. Springer, 2015.

[100] PC Hsiu, HC Yeh, PH Tsai, CS Shih, DH Burkhardt, TW Kuo, JWS Liu, TY Huang, et al. A general model for medication scheduling. *Institute of Information Science, Academia Sinica, Taiwan, Technical Report TR-IIS-05-008*, 2005.

[101] Han-Chun Yeh, Pi-Cheng Hsiu, Chi-Sheng Shih, Pei-Hsuan Tsai, and Jane WS Liu. Apamat: A prescription algebra for medication authoring tool. In *2006 IEEE International Conference on Systems, Man and Cybernetics*, volume 5, pages 4284–4291. IEEE, 2006.

[102] PH Tsai, CY Yu, MY Wang, JK Zao, HC Yeh, CS Shih, and JWS Liu. imat: Intelligent medication administration tools. In *The 12th IEEE international conference on e-health networking, applications and services*, pages 308–315. IEEE, 2010.

[103] Luca Anselma, Paolo Terenziani, Stefania Montani, and Alessio Bottrighi. Towards a comprehensive treatment of repetitions, periodicity and temporal constraints in clinical guidelines. *Artificial Intelligence in Medicine*, 38(2):171–195, 2006.

[104] Sunil Kripalani, Rashanda Robertson, Melissa H Love-Ghaffari, Laura E Henderson, Jessica Praska, Akilah Strawder, Marra G Katz, and Terry A Jacobson. Development of an illustrated medication schedule as a low-literacy patient education tool. *Patient education and counseling*, 66(3):368–377, 2007.

[105] Michelle C Carlson, Linda P Fried, Qian-Li Xue, Carmen Tekwe, and Jason Brandt. Validation of the hopkins medication schedule to identify difficulties in taking medications. *The Journals of Gerontology Series A: Biological Sciences and Medical Sciences*, 60(2):217–223, 2005.

[106] Edward L Machtinger, Frances Wang, Lay-Leng Chen, Maytrella Rodriguez, Sandy Wu, and Dean Schillinger. A visual medication schedule to improve anticoagulation control: a randomized, controlled trial. *The Joint Commission Journal on Quality and Patient Safety*, 33(10):625–635, 2007.

[107] Daniel G Morrow, Von O Leirer, and Jill M Andrassy. Using icons to convey medication schedule information. *Applied Ergonomics*, 27(4):267–275, 1996.

[108] Fabiano Dalpiaz, Xavier Franch, and Jennifer Horkoff. istar 2.0 language guide. *arXiv preprint arXiv:1605.07767*, 2016.

[109] Izet Masic. Medical decision making-an overview. *Acta Informatica Medica*, 30(3):230, 2022.

[110] VWM Drury, J Grimley Evans, CF George, Dorcas Knowles, DA Pyke, and DG Williams. Medication for the elderly. *Journal of the Royal College of Physicians of London*, 18(1), 1984.

[111] Ruth M Seymour and Philip A Routledge. Important drug-drug interactions in the elderly. *Drugs & aging*, 12:485–494, 1998.

[112] Jane H Barnsteiner. Medication reconciliation: Transfer of medication information across settings—keeping it free from error. *AJN The American Journal of Nursing*, 105(3):31–36, 2005.

[113] Kenneth S Boockvar, Heather Carlson LaCorte, Vincent Giambanco, Bella Fridman, and Albert Siu. Medication reconciliation for reducing drug-discrepancy adverse events. *The American journal of geriatric pharmacotherapy*, 4(3):236–243, 2006.

[114] Esther Durán-García, Cecilia M Fernandez-Llamazares, and Miguel A Calleja-Hernández. Medication reconciliation: passing phase or real need? *International journal of clinical pharmacy*, 34:797–802, 2012.

[115] Ross Koppel, Joshua P Metlay, Abigail Cohen, Brian Abaluck, A Russell Localio, Stephen E Kimmel, and Brian L Strom. Role of computerized physician order entry systems in facilitating medication errors. *Jama*, 293(10):1197–1203, 2005.

[116] Robert W Floyd. Algorithm 97: shortest path. *Communications of the ACM*, 5(6):345, 1962.

[117] Jiazhao Li, Corey Lester, Xinyan Zhao, Yuting Ding, Yun Jiang, and VG Vydiswaran. Pharmmt: a neural machine translation approach to simplify prescription directions. *arXiv preprint arXiv:2204.03830*, 2022.

[118] Chittaranjan Andrade and TS Sathyanarayana Rao. Prescription writing: Generic or brand? *Indian journal of psychiatry*, 59(2):133, 2017.

[119] Marie N O'Connor, Paul Gallagher, and Denis O'Mahony. Inappropriate prescribing: criteria, detection and prevention. *Drugs & aging*, 29:437–452, 2012.

[120] Raul E Valdes-Perez. Spatio-temporal reasoning and linear inequalities. *Artificial intelligence Laboratory*, 1986.

[121] Carlo Combi and Yuval Shahar. Temporal reasoning and temporal data maintenance in medicine: issues and challenges. *Computers in biology and medicine*, 27(5):353–368, 1997.

[122] Yuval Shahar, Silvia Miksch, and Peter Johnson. The asgaard project: a task-specific framework for the application and critiquing of time-oriented clinical guidelines. *Artificial intelligence in medicine*, 14(1-2):29–51, 1998.

[123] Colin Gordon, Ian Herbert, and Peter Johnson. Knowledge representation and clinical practice guidelines: the dilemma and prestige projects. In *Medical Informatics Europe'96*, pages 511–515. IOS Press, 1996.

[124] Mark A Musen, Samson W Tu, Amar K Das, and Yuval Shahar. Eon: A component-based approach to automation of protocol-directed therapy. *Journal of the American Medical Informatics Association*, 3(6):367–388, 1996.

[125] Richard N Shiffman, Bryant T Karras, Abha Agrawal, Roland Chen, Luis Marenco, and Sujai Nath. Gem: a proposal for a more comprehensive guideline document model using xml. *Journal of the American Medical Informatics Association*, 7(5):488–498, 2000.

[126] Samson W Tu, Michael G Kahn, Mark A Musen, LM Fagan, and Jay C Ferguson. Episodic skeletal-plan refinement based on temporal data. *Communications of the ACM*, 32(12):1439–1455, 1989.

[127] Jean François Ethier, François Goyer, Paul Fabry, and Adrien Barton. The Prescription of Drug Ontology 2.0 (PDRO): More Than the Sum of Its Parts. *International Journal of Environmental Research and Public Health 2021, Vol. 18, Page 12025*, 18(22):12025, nov 2021.

[128] Ajay Kumar, Saurabh Jain, Indu Dangi, Shalini Chowdary, Omprakash Choubitker, Krishna Kumar Pandey, and RS Pawar. Ideal drug prescription writing. *World J. Pharm. Pharm. Sci*, 8(3), 2019.

[129] Andy Fox. Safer Prescribing Workbook. https://www.uhs.nhs.uk/media/suhtideal /doctors/saferprescribingworkbook/section1-prescriptionwriting.pdf, 2022. Accessed: 2022-06-09.

[130] Sunil Kripalani, Brian Schmotzer, and Terry A Jacobson. Improving medication adherence through graphically enhanced interventions in coronary heart disease (image-chd): a randomized controlled trial. *Journal of general internal medicine*, 27:1609–1617, 2012.

[131] Greg Weeks, Johnson George, Katie Maclure, and Derek Stewart. Non-medical prescribing versus medical prescribing for acute and chronic disease management in primary and secondary care. *Cochrane Database of Systematic Reviews*, 11, 2016.

[132] Diemert Simon. A mathematical basis for Medication Prescription and Adherence. *Journal of Chemical Information and Modeling*, 53(9):1689–1699, 2013.

[133] Brian M Caulfield and Seamas C Donnelly. What is connected health and why will it change your practice? *QJM: An International Journal of Medicine*, 106(8):703–707, 2013.

[134] Health Level Seven International. High level 7. https://www.hl7.org/, 2023. Accessed: 2023-07-17.

[135] Robert H Dolin, Liora Alschuler, Calvin Beebe, Paul V Biron, Sandra Lee Boyer, Daniel Essin, Elliot Kimber, Tom Lincoln, and John E Mattison. The hl7 clinical document architecture. *Journal of the American Medical Informatics Association*, 8(6):552–569, 2001.

[136] Duane Bender and Kamran Sartipi. Hl7 fhir: An agile and restful approach to healthcare information exchange. In *Proceedings of the 26th IEEE international symposium on computer-based medical systems*, pages 326–331. IEEE, 2013.

[137] Health Level Seven International. Fhir medication prescription. https://www.hl7.org/FHIR/2015May/medicationprescription.html, 2023. Accessed: 2023-07-17.

[138] Tobias Kuhn. Acewiki: A natural and expressive semantic wiki. *arXiv preprint arXiv:0807.4618*, 2008.

[139] Silvia Miksch, Yuval Shahar, and Peter Johnson. Asbru: A Task-Specific, Intention-Based, and Time-Oriented Language for Representing Skeletal Plans. *7th Workshop on Knowledge Engineering: Methods and Languages (KEML-97)*, 1997.

[140] Juancarlo Añez. *TatSu*, 2017 (accessed December 22, 2020).

[141] Rina Dechter, Itay Meiri, and Judea Pearl. Temporal constraint networks. *Artif. Intell.*, 49(1–3):61–95, May 1991.

[142] Thomas Kluyver, Benjamin Ragan-Kelley, Fernando Pérez, Brian Granger, Matthias Bussonnier, Jonathan Frederic, Kyle Kelley, Jessica Hamrick, Jason Grout, Sylvain Corlay, Paul Ivanov, Damián Avila, Safia Abdalla, and Carol Willing. Jupyter notebooks – a publishing format for reproducible computational workflows. In F. Loizides and B. Schmidt, editors, *Positioning and Power in Academic Publishing: Players, Agents and Agendas*, pages 87 – 90. IOS Press, 2016.

[143] Py2neo. urlhttps://pypi.org/project/py2neo/. Accessed: 2020-09-30.

[144] Graphviz. urlhttps://graphviz.org/. Accessed: 2020-09-30.

[145] Maybins Douglas Lengwe, Jens Weber, and Charles Perin. Integrating prescriptions into calendars: Design considerations. In *Graphics Interface 2023*, 2022.

[146] Elísio Costa, Anna Giardini, Magda Savin, Enrica Menditto, Elaine Lehane, Olga Laosa, Sergio Pecorelli, Alessandro Monaco, and Alessandra Marengoni. Interventional tools to improve medication adherence: review of literature. *Patient preference and adherence*, pages 1303–1314, 2015.

[147] Katherine R Salisbury, Varun K Ranpariya, and Steven R Feldman. Accountability in reminder-based adherence interventions: a review. *Patient Education and Counseling*, 105(8):2645–2652, 2022.

[148] Asma A Abahussin and Ahmed I Albarrak. Vaccination adherence: Review and proposed model. *Journal of Infection and Public Health*, 9(6):781–789, 2016.

[149] AJ Cross, Ra Elliott, K Petrie, L Kuruvilla, and J George. Adherence in older adults prescribed multiple medications. *Cochrane Database of Systematic Reviews*, 5(5), 2020.

[150] Yuhan Luo, Peiyi Liu, and Eun Kyoung Choe. Co-Designing Food Trackers with Dietitians: Identifying Design Opportunities for Food Tracker Customization. *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, 2019.

[151] Carman Neustaedter and AJ Bernheim Brush. " linc-ing" the family: the participatory design of an inkable family calendar. In *Proceedings of the SIGCHI conference on Human Factors in computing systems*, pages 141–150, 2006.

[152] Andy Crabtree, Terry Hemmings, Tom Rodden, and John Mariani. Informing the development of calendar systems for domestic use. In *ECSCW 2003*, pages 119–138. Springer, 2003.

[153] Predrag Klasnja, Andrea Hartzler, Christopher Powell, and Wanda Pratt. Supporting cancer patients' unanchored health information management with mobile technology. In *AMIA Annual Symposium Proceedings*, volume 2011, page 732. American Medical Informatics Association, 2011.

[154] Jarke J Van Wijk and Edward R Van Selow. Cluster and calendar based visualization of time series data. In *Proceedings 1999 IEEE Symposium on Information Visualization (InfoVis' 99)*, pages 4–9. IEEE, 1999.

[155] Earo Wang, Dianne Cook, and Rob J Hyndman. Calendar-based graphics for visualizing people's daily schedules. *Journal of computational and graphical statistics*, 29(3):490–502, 2020.

[156] Catherine Plaisant, Aaron Clamage, Hilary Browne Hutchinson, Benjamin B Bederson, and Allison Druin. Shared family calendars: Promoting symmetry and accessibility. *ACM Transactions on Computer-Human Interaction (TOCHI)*, 13(3):313–346, 2006.

[157] Carman Neustaedter, Lyn Bartram, and Aaron Mah. Everyday activities and energy consumption: how families understand the relationship. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 1183–1192, 2013.

[158] Marie T Brown and Jennifer K Bussell. Medication adherence: Who cares? In *Mayo clinic proceedings*, volume 86, pages 304–314. Elsevier, 2011.

[159] Holly B Jimison, Stuart Hagler, Gregorij Kurillo, Ruzena Bajcsy, and Misha Pavel. Remote health coaching for interactive exercise with older adults in a home environment. In *2015 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pages 5485–5488. IEEE, 2015.

[160] DeKoekkoek Tracy, Given Barbara, Given Charles, Ridenour Kimberly, Schueller Monica, and Spoelstra Sandra. mHealth SMS text messaging interventions and to promote medication adherence: An integrative review. *Journal of Clinical Nursing*, 24(19-20):2722–2735, 2015.

[161] Matthew Brehmer, Bongshin Lee, Benjamin Bach, Nathalie Henry Riche, and Tamara Munzner. Timelines Revisited: A Design Space and Considerations for Expressive Storytelling. *IEEE Transactions on Visualization and Computer Graphics*, 1, 2016.

[162] Wallace Clark. *The Gantt chart: A working tool of management.* Ronald Press Company, 1922.

[163] TI Kirkpatrick and Norman Ross Clark. Pert as an aid to logic design. *IBM journal of Research and Development*, 10(2):135–141, 1966.

[164] Alan D Russell and Asad Udaipurwala. Construction schedule visualization. In *Computing in Civil Engineering (2002)*, pages 167–178. American Society of Civil Engineers, 2003.

[165] HyounSeok Moon, Nashwan Dawood, and LeenSeok Kang. Development of workspace conflict visualization system using 4d object of work schedule. *Advanced Engineering Informatics*, 28(1):50–65, 2014.

[166] Jaemin Jo, Jaeseok Huh, Jonghun Park, Bohyoung Kim, and Jinwook Seo. Livegantt: Interactively visualizing a large manufacturing schedule. *IEEE transactions on visualization and computer graphics*, 20(12):2329–2338, 2014.

[167] Catherine Plaisant, Richard Mushlin, Aaron Snyder, Jia Li, Dan Heller, and Ben Shneiderman. Lifelines: using visualization to enhance navigation and

analysis of patient records. In *The craft of information visualization*, pages 308–312. Elsevier, 2003.

[168] Brett Milash, Catherine Plaisant, and Anne Rose. LifeLines: visualizing personal histories. *Conference on Human Factors in Computing Systems - Proceedings*, pages 392–393, 1996.

[169] Krist Wongsuphasawat, John Alexis Guerra Gómez, Catherine Plaisant, Taowei David Wang, Shneiderman Ben, and Meirav Taieb-Maimon. LifeFlow: Visualizing an overview of event sequences. *Conference on Human Factors in Computing Systems - Proceedings*, pages 1747–1756, 2011.

[170] D. L. Gresh, D. A. Rabenhorst, A. Shabo, and S. Slavin. PRIMA: A case study of using information visualization techniques for patient record analysis. *Proceedings of the IEEE Visualization Conference*, pages 509–512, 2002.

[171] Mona Hosseinkhani Loorak, Charles Perin, Noreen Kamal, Michael Hill, and Sheelagh Carpendale. Timespan: Using visualization to explore temporal multidimensional data of stroke patients. *IEEE transactions on visualization and computer graphics*, 22(1):409–418, 2015.

[172] Katie Walker, Eden Potter, Indae Hwang, Tim Dwyer, Diana Egerton-Warburton, Keith Joe, Jennie Hutton, Sam Freeman, and Daphne Flynn. Visualising emergency department wait times; rapid iterative testing to determine patient preferences for displays. *medRxiv*, pages 2022–03, 2022.

[173] In Merriam-Webster.com. Calendar definition. https://www.merriam-webster.com/dictionary/calendar, 2022. Accessed: 2022-05-04.

[174] Catarina Macas and Penousal Machado. Radial calendar of consumption. *Information Visualisation - Biomedical Visualization, Visualisation on Built and Rural Environments and Geometric Modelling and Imaging, IV 2018*, pages 103–109, dec 2018.

[175] Wolfgang Mueller, Marc Weber Marc, and Alexa Wolfgang Müller. Visualizing Time-Series on Spirals. In *IEEE Symposium on Information Visualization*, 2001.

[176] Ian Spence. William Playfair and the psychology of graphs. *Proceedings of the American Statistical Association*, Section on Statistical Graphics:2426–2436, 2006.

[177] Florence Nightingale. Notes on matters affecting the health, efficiency, and hospital administration of the British army: founded chiefly on the experience of the late war. *Harrison and Sons*, St. Martin's Lane, WC, 1987.

[178] Paolo Buono, Maria Francesca Costabile, and Rosa Lanzilotti. A circular visualization of people's activities in distributed teams. *Journal of Visual Languages & Computing*, 25(6):903–911, dec 2014.

[179] M. Joan Saary. Radar plots: a useful way for presenting multivariate health care data. *Journal of Clinical Epidemiology*, 61(4):311–317, apr 2008.

[180] Bruno Pagno and Luciana Nedel. Everyday Visualization. *IEEE VIS 2015 workshop*, 2015.

[181] Charles Perin. Cycles and (A)Symmetry | Exploring the Design of Shareable Personal Visualizations. In IEEE, editor, *IEEE VIS Arts Program Annotated Portfolios*, Berlin, Germany, October 2018. IEEE.

[182] Geoffrey M. Draper, Geoffrey M. Draper, Yarden Livnat, and Richard F. Riesenfeld. A Survey of Radial Methods for Information Visualization. *IEEE transactions on visualization and computer graphics*, 2009.

[183] Microsoft. Outlook calendar. https://outlook.office.com/calendar, 2022. Accessed: 2022-05-04.

[184] Marco Ancona, Marilou Beyeler, Markus Gross, and Tobias Gunther. MineTime Insight: Visualizing Meeting Habits to Promote Informed Scheduling Decisions. *IEEE Transactions on Visualization and Computer Graphics*, 27(3):1986–1999, mar 2021.

[185] Joe Tullio, Jeremy Goecks, Elizabeth D. Mynatt, and David H. Nguyen. Augmenting shared personal calendars. *Proceedings of the 15th annual ACM symposium on User interface software and technology.*, page 11, 2002.

[186] Siwei Fu, Jian Zhao, Hao Fei Cheng, Haiyi Zhu, and Jennifer Marlow. T-Cal: Understanding Team Conversation Data with Calendar-based Visualization. *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, 2018.

[187] Manas Tungare, Manuel Pérez-Quiñones, and Alyssa Sams. An exploratory study of personal calendar use. *arXiv preprint arXiv:0809.3447*, 2008.

[188] JF Kelley and Alphonse Chapanis. How professional persons keep their calendars: Implications for computerization. *Journal of Occupational Psychology*, 55(4):241–256, 1982.

[189] David Beard, Murugappan Palaniappan, Alan Humm, David Banks, Anil Nair, and Yen-Ping Shan. A visual calendar for scheduling group meetings. In *Proceedings of the 1990 ACM conference on Computer-supported cooperative work*, pages 279–290, 1990.

[190] Andrea Civan, Meredith M Skeels, Anna Stolyar, and Wanda Pratt. Personal health information management: consumers' perspectives. In *AMIA Annual Symposium Proceedings*, volume 2006, page 156. American Medical Informatics Association, 2006.

[191] Victoria Bellotti, Brinda Dalal, Nathaniel Good, Peter Flynn, Daniel G Bobrow, and Nicolas Ducheneaut. What a to-do: studies of task management towards the design of a personal task list manager. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 735–742, 2004.

[192] Christine M Kincaid, Pierre B Dupont, and A. Roger Kaye. Electronic calendars in the office: an assessment of user needs and current technology. *ACM Transactions on Information Systems (TOIS)*, 3(1):89–102, 1985.

[193] Melissa K Kelly, Kevin F Miller, Ge Fang, and Gary Feng. When days are numbered: Calendar structure and the development of calendar processing in english and chinese. *Journal of Experimental Child Psychology*, 73(4):289–314, 1999.

[194] Douglas Brent West et al. *Introduction to graph theory*, volume 2. Prentice hall Upper Saddle River, 2001.

[195] Olav W Bertelsen, Susanne Bødker, Eva Eriksson, Eve Hoggan, and Jo Vermeulen. Beyond generalization: research for the very particular. *Interactions*, 26(1):34–38, 2018.

[196] Inc Otter.ai. Otter. https://otter.ai, 2022. Accessed: 2022-05-04.

[197] QSR International Pty Ltd. Nvivo.

[198] Jonathan Lazar, Jinjuan Heidi Feng, and Harry Hochheiser. *Research methods in human-computer interaction*. Morgan Kaufmann, 2017.

[199] Tableau Software LLC. Tableau desktop.

[200] Fateme Rajabiyazdi, Charles Perin, Lora Oehlberg, and Sheelagh Carpendale. Communicating patient health data: A wicked problem. *IEEE Computer Graphics and Applications*, 41(6):179–186, 2021.

[201] Apple. Apple calendar. https://www.icloud.com/calendar/, 2022. Accessed: 2022-05-04.

[202] Apache Software Foundation. Apache echarts. https://echarts.apache.org/en/index.html, 2023. Accessed: 2022-07-10.

[203] Mei-Ying Wang, Pei-Hsuan Tsai, Jane W-S Liu, and John K Zao. Wedjat: a mobile phone based medicine in-take reminder and monitor. In *2009 Ninth IEEE International Conference on Bioinformatics and BioEngineering*, pages 423–430. IEEE, 2009.

[204] Oracle Syetems. Mysql. https://www.mysql.com/, 2023. Accessed: 2023-07-17.

[205] Webix. Webix calendar. https://webix.com/scheduler/, 2023. Accessed: 2023-07-17.

[206] Tui. Tui calendar. https://ui.toast.com/tui-calendar, 2023. Accessed: 2023-07-17.

[207] Angular. Angular calendar. https://www.angular-calendar.com/, 2023. Accessed: 2023-07-17.

[208] Maybins Lengwe, Jens H Weber, Charles Perin, and Morgan Price. Supporting reflection on medication adherence: Eliminating a blind spot in our rearview mirror. In *2023 IEEE 31st International Requirements Engineering Conference Workshops (REW)*, pages 432–439. IEEE, 2023.

[209] Eric PS Baumer, Vera Khovanskaya, Mark Matthews, Lindsay Reynolds, Victoria Schwanda Sosik, and Geri Gay. Reviewing reflection: on the use of reflection in interactive system design. In *Proceedings of the 2014 conference on Designing interactive systems*, pages 93–102, 2014.

[210] Ian Li, Anind Dey, and Jodi Forlizzi. A stage-based model of personal informatics systems. In *Proceedings of the SIGCHI conference on human factors in computing systems*, pages 557–566, 2010.

[211] Melanie Swan. The quantified self: Fundamental disruption in big data science and biological discovery. *Big data*, 1(2):85–99, 2013.

[212] Rowanne Fleck and Geraldine Fitzpatrick. Reflecting on reflection: framing a design landscape. In *Proceedings of the 22nd Conference of the Computer-Human Interaction Special Interest Group of Australia on Computer-Human Interaction*, pages 216–223, 2010.

[213] Lena Mamykina, Elizabeth Mynatt, Patricia Davidson, and Daniel Greenblatt. Mahi: investigation of social scaffolding for reflective thinking in diabetes management. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 477–486, 2008.

[214] Ian Anderson, Julie Maitland, Scott Sherwood, Louise Barkhuus, Matthew Chalmers, Malcolm Hall, Barry Brown, and Henk Muller. Shakra: tracking and sharing daily activity levels with unaugmented mobile phones. *Mobile networks and applications*, 12:185–199, 2007.

[215] Eun Kyoung Choe, Bongshin Lee, Haining Zhu, Nathalie Henry Riche, and Dominikus Baur. Understanding self-reflection: how people reflect on personal data through visual data exploration. In *Proceedings of the 11th EAI International Conference on Pervasive Computing Technologies for Healthcare*, pages 173–182, 2017.

[216] John Rooksby, Mattias Rost, Alistair Morrison, and Matthew Chalmers. Personal tracking as lived informatics. In *Proceedings of the SIGCHI conference on human factors in computing systems*, pages 1163–1172, 2014.

[217] Mary Ellen Roberts, Kathy J Wheeler, and Mary B Neiheisel. Medication adherence part three: Strategies for improving adherence. *Journal of the American Association of Nurse Practitioners*, 26(5):281–287, 2014.

[218] Kiron Koshy, Christopher Limb, Buket Gundogan, Katharine Whitehurst, and Daniyal J Jafree. Reflective practice in health care and how to reflect effectively. *International journal of surgery. Oncology*, 2(6):e20, 2017.

[219] Anna Janssen, Judy Kay, Stella Talic, Martin Pusic, Robert J Birnbaum, Rodrigo Cavalcanti, Dragan Gasevic, and Tim Shaw. Electronic health records that support health professional reflective practice: A missed opportunity in digital health. *Journal of Healthcare Informatics Research*, 6(4):375–384, 2022.

[220] Gabrielle M Salib, Juan Fernando Maestre, Kenneth B Nimley, Nadia Dowshen, and Gabriela Marcu. The role of reflection and context in medication adherence tracking for people living with hiv. In *Extended Abstracts of the 2018 CHI Conference on Human Factors in Computing Systems*, pages 1–6, 2018.

[221] Adrian Bussone, Simone Stumpf, and George Buchanan. It feels like i'm managing myself: Hiv+ people tracking their personal health information. In *Proceedings of the 9th nordic conference on human-computer interaction*, pages 1–10, 2016.

[222] Lauren M Hill, Carol E Golin, Allison Pack, Jessica Carda-Auten, Deshira D Wallace, Sruthi Cherkur, Claire E Farel, Elias P Rosen, Monica Gandhi, Heather M Asher Prince, et al. Using real-time adherence feedback to enhance communication about adherence to antiretroviral therapy: patient and clinician perspectives. *The Journal of the Association of Nurses in AIDS Care: JANAC*, 31(1):25, 2020.

[223] Tom Quisel, Luca Foschini, Susan M Zbikowski, and Jessie L Juusola. The association between medication adherence for chronic conditions and digital health activity tracking: retrospective analysis. *Journal of medical Internet research*, 21(3):e11486, 2019.

[224] Christina Kelley, Bongshin Lee, and Lauren Wilcox. Self-tracking for mental wellness: understanding expert perspectives and student experiences. In *Proceedings of the 2017 CHI conference on human factors in computing systems*, pages 629–641, 2017.

[225] Mayara Costa Figueiredo, Clara Caldeira, Tera L Reynolds, Sean Victory, Kai Zheng, and Yunan Chen. Self-tracking for fertility care: collaborative support for a highly personalized problem. *Proceedings of the ACM on Human-Computer Interaction*, 1(CSCW):1–21, 2017.

[226] Jessica S Ancker, Holly O Witteman, Baria Hafeez, Thierry Provencher, Mary Van de Graaf, and Esther Wei. "you get reminded you're a sick person": personal data tracking and patients with multiple chronic conditions. *Journal of medical Internet research*, 17(8):e202, 2015.

[227] Susan Michie, Charles Abraham, Craig Whittington, John McAteer, and Sunjai Gupta. Effective techniques in healthy eating and physical activity interventions: a meta-regression. *Health psychology*, 28(6):690, 2009.

[228] James J Annesi. Effects of computer feedback on adherence to exercise. *Perceptual and motor skills*, 87(2):723–730, 1998.

[229] Eun Kyoung Choe, Bongshin Lee, Matthew Kay, Wanda Pratt, and Julie A Kientz. Sleeptight: low-burden, self-monitoring technology for capturing and reflecting on sleep behaviors. In *Proceedings of the 2015 ACM international joint conference on pervasive and ubiquitous computing*, pages 121–132, 2015.

[230] Bon Adriel Aseniero, Charles Perin, Wesley Willett, Anthony Tang, and Sheelagh Carpendale. Activity river: Visualizing planned and logged personal activities for reflection. In *Proceedings of the International Conference on Advanced Visual Interfaces*, pages 1–9, 2020.

[231] Eun Kyoung Choe, Nicole B Lee, Bongshin Lee, Wanda Pratt, and Julie A Kientz. Understanding quantified-selfers' practices in collecting and exploring personal data. In *Proceedings of the SIGCHI conference on human factors in computing systems*, pages 1143–1152, 2014.

[232] Daniel A Epstein, An Ping, James Fogarty, and Sean A Munson. A lived informatics model of personal informatics. In *Proceedings of the 2015 ACM*

international joint conference on pervasive and ubiquitous computing, pages 731–742, 2015.

[233] Eun Kyoung Choe, Bongshin Lee, et al. Characterizing visualization insights from quantified selfers' personal data presentations. *IEEE computer graphics and applications*, 35(4):28–37, 2015.

[234] Andrea Cuttone, Michael Kai Petersen, and Jakob Eg Larsen. Four data visualization heuristics to facilitate reflection in personal informatics. In *Universal Access in Human-Computer Interaction. Design for All and Accessibility Practice: 8th International Conference, UAHCI 2014, Held as Part of HCI International 2014, Heraklion, Crete, Greece, June 22-27, 2014, Proceedings, Part IV 8*, pages 541–552. Springer, 2014.

[235] Microsoft Corporation. Microsoft excel.

[236] Zoom Video Communications. Zoom. urlhttps://zoom.us, 2022. Accessed: 2022-10-04.

[237] Jakob Nielsen. *Usability engineering*. Morgan Kaufmann, 1994.

[238] Dean Schillinger, John Piette, Kevin Grumbach, Frances Wang, Clifford Wilson, Carolyn Daher, Krishelle Leong-Grotz, Cesar Castro, and Andrew B Bindman. Closing the loop: physician communication with diabetic patients who have low health literacy. *Archives of internal medicine*, 163(1):83–90, 2003.

[239] Alan M Delamater. Improving patient adherence. *Clinical diabetes*, 24(2):71–77, 2006.

[240] Wiktoria Wilkowska and Martina Ziefle. Privacy and data security in e-health: Requirements from the user's perspective. *Health informatics journal*, 18(3):191–201, 2012.

[241] Grace Fox. "to protect my health or to protect my health privacy?" a mixed-methods investigation of the privacy paradox. *Journal of the Association for Information Science and Technology*, 71(9):1015–1029, 2020.

Appendix A

CHAOS Grammar

Algorithm 4 CHAOS Grammar

```
prescription = (((requirement[’,’])+ restriction*)|refine);
requirement = ’require’ action duration frame;
restriction = ’restrict’ (apart|together);
refine = ’refine’ refine_list[’;’]+;
refine_list = action refine_duration refine_times[’,’]+;
frame = frame:([’,’] administer); administer = administer:([’administer’] number [’-’ number] [(day_period|time_period)]);
iteration = iteration:(’every’ [(number ordinal)] long_unit);
duration = duration:(’for’ number long_unit (iteration|indexing));
indexing = indexing:(’on’ weekday);
refine_duration = refine_duration:(’for’ number long_unit ’every’ weekday);
refine_times = ’at’ number short_unit [administer];
day_period = day_period: (’every’ (’morning’|’afternoon’|’evening’)[(’,’|’and’)]+);
time_period = time_period: (’every’ number [(’,’|’and’)]+);
apart = apart:([’taking’] action ’and’ action ’apart by’ number short_unit);
together = together: ([’taking ’] action ’and’ action ’together within’ number short_unit);
number = number:/[0-9]+/;
action = action:/[a-zA-Z]+[0-9]*/;
long_unit = unit:(’year’|’years’|’month’|’months’|’week’|’weeks’|’day’|’days’);
short_unit = unit:(’morning’ |’noon’ |’afternoon’ |’evening’| ’night’|’day’|’days’|’hour’|’hours’|’minutes’|’minute’);
weekday = weekday:((’sunday’|’monday’ |’tuesday’ |’wednesday’ |’thursday’ |’friday’ |’saturday’) [(’,’|’and’)]+);
ordinal = [’st’ | ’nd’ | ’rd’ | ’th’];
```

Appendix B

FHIR Medication Prescription Specification

```
{
  "resourceType" : "MedicationPrescription",
  "identifier" : [{ Identifier }],
  "dateWritten" : "<dateTime>",
  "status" : "<code>",
  "patient" : { Reference(Patient) },
  "prescriber" : { Reference(Practitioner) },
  "encounter" : { Reference(Encounter) },
  "reasonCodeableConcept" : { CodeableConcept },
  "reasonReference" : { Reference(Condition) },
  "note" : "<string>",
  "medication" : { Reference(Medication) },
  "dosageInstruction" : [{
    "text" : "<string>",
    "additionalInstructions" : { CodeableConcept },
    "scheduledDateTime" : "<dateTime>",
    "scheduledPeriod" : { Period },
    "scheduledTiming" : { Timing },
    "asNeededBoolean" : <boolean>,
    "asNeededCodeableConcept" : { CodeableConcept },
    "site" : { CodeableConcept },
```

```
    "route" : { CodeableConcept },
    "method" : { CodeableConcept },
    "doseRange" : { Range },
    "doseQuantity" : { Quantity },
    "rate" : { Ratio },
    "maxDosePerPeriod" : { Ratio }
  }],
  "dispense" : {
    "medication" : { Reference(Medication) },
    "validityPeriod" : { Period },
    "numberOfRepeatsAllowed" : "<positiveInt>",
    "quantity" : { Quantity },
    "expectedSupplyDuration" : { Duration }
  },
  "substitution" : {
    "type" : { CodeableConcept },
    "reason" : { CodeableConcept }
  }
}
```