Meta-Modelling Kindness

Faeq Alrimawi
faeq.alrimawi@lero.ie
Lero - the SFI Research Centre for Software
University of Limerick
Limerick, Ireland

Bashar Nuseibeh
bashar.nuseibeh@open.ac.uk
School of Computing & Communications
The Open University
Milton Keynes, United Kingdom

ABSTRACT

Kindness is a psycho-social phenomenon that is also recognized as an important pro-social behaviour. The use of digital technology provides opportunities to promote kindness in various ways, such as in social media campaigns and online communities. In principle, software engineers are well positioned to develop automated systems that can facilitate software-mediated kindness. However, in practice, incorporating kindness concerns explicitly in the development and use of software systems is challenging: kindness is highly context dependent, affected by a range of factors such as intentions and opportunity.

In this paper, we explore systematic ways in which kindness concerns can be considered by software engineers. We propose a novel meta-model that captures essential entities and relations associated with kindness. The meta-model enables the representation of possible instances or *opportunities* for performing acts of kindness, by considering the *actors* involved (such as giver, receiver, and observer), their psychological and social attributes that promote kindness (such as emotional states and social relatedness), the *acts* needed to fulfil kindness opportunities (such as motivation, ability, and timeliness), and other *contextual* factors (such as location and time). Our meta-model is demonstrated through two software application scenarios that enable charitable donations and kindness in business. Overall, our proposal offers a first, tentative, but concrete step towards enabling kind computing, and promoting kindness in software systems.

CCS CONCEPTS

• **Software and its engineering** → **Software notations and tools**; • **Human-centered computing** → *Collaborative and social computing*;

KEYWORDS

Kindness, Meta-Modelling, Software Engineering, Kind Computing

ACM Reference Format:
Faeq Alrimawi and Bashar Nuseibeh. 2024. Meta-Modelling Kindness. In *ACM/IEEE 27th International Conference on Model Driven Engineering Languages and Systems (MODELS '24), September 22–27, 2024, Linz, Austria.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3640310.3674095

MODELS '24, September 22–27, 2024, Linz, Austria

ACM ISBN 979-8-4007-0504-5/24/09
https://doi.org/10.1145/3640310.3674095

1 INTRODUCTION

Kindness is a psycho-social phenomenon that involves prosocial behavior towards others, motivated by genuine concern and goodwill [23, 31]. Acts of kindness can range from small gestures of appreciation, such as a smile or a compliment, to larger acts of generosity, such as volunteering time and skills, or donating money. Research has shown that kindness has significant psycho-social benefits, such as increased well-being [10, 15, 19], social connectedness [20], and positive affect [39], both for the giver and the receiver. However, the traditional ways of performing acts of kindness, such as in-person interactions, are not always practical or feasible (for example, supporting a friend who lives in a different country during a difficult time).

Digital technology offers distinct opportunities for promoting and acknowledging acts of kindness in inventive ways, while also exposing the possibility of unkindness, either by design or by accident. Social media platforms, for example, can facilitate the spread of positive messages, encourage users to engage in acts of kindness, and help overcome barriers to kindness such as distance, anonymity, and social isolation by providing new ways to connect and engage with others [34]. However, these same platforms can also be used to spread negative messages and enable cyberbullying, and anonymity and distance may lead to online aggression and hate speech [33, 45].

While the potential of digital technology to promote kindness and to mitigate unkindness is significant, modelling kindness is challenging. This is because kindness involves a complex interplay of psychological and social factors, which are context-dependent and diverse [9, 10, 30]. Kindness is a multifaceted phenomenon that encompasses various acts (such as helping someone), actors (such as giver and receiver) and contexts (such as home or workplace) [11, 31]. The use of digital technology to promote kindness thus requires careful and responsible consideration of the ethical implications of modelling such a fundamental aspect of human behavior [9, 34].

To address these challenges, we propose a meta-model that captures key entities and relationships relevant to kindness. The meta-model enables the representation of kindness *opportunities*—which encompass the combination of actors, acts, and contexts that lead to the performance of kindness. It offers a structured representation that can potentially be integrated into various applications in order to systematically encourage and sustain kind behaviors. The meta-model contribution is presented in four parts.

First, drawing on the social psychology literature, we identify and categorise key concepts and their relationships. The meta-model captures primary *actors* involved in performing kindness, which are: the *giver*, who performs the kindness act; the *receiver*, who receives the act; and the *observer*, who witnesses the act. It also allows for the representation of the specific *acts* involved in kindness opportunities. These acts encompass both kindness acts

and supporting acts, with the latter being those that assist in the delivery of kindness. Drawing on the literature of behavior design [13, 14, 16, 28], we identified three main types of supporting acts: motivation acts, which seek to increase the giver's desire for performing kindness (such as emphasizing the benefits associated with kindness); ability acts, which aim to increase the giver's capacity to perform kindness (for example to reduce time or effort required); and prompt acts, which aim to deliver proper and timely alerts to the giver to perform kindness acts (such as app or email notification). Our meta-model also includes entities that enable the representation of the *context* in which kindness occurs, allowing for the inclusion of key contextual dimensions such as location and time.

Second, it allows the explicit representation of actors' psychological and social factors that can have impact on their kindness. To better represent the key factors, we conducted a review of the literature and identified the most significant factors, such as emotions and social relatedness. We then developed a supporting meta-model that captures these factors, allowing for a more comprehensive and nuanced representation of kindness.

Third, we introduce two algorithms to offer opportunities for automation. The first algorithm determines if a given situation qualifies as a kindness opportunity, while the second assesses overall motivation and ability to trigger prompts. Fourth, we made the meta-model accessible as an Eclipse plugin to allow its potential exploitation in software development environments

To demonstrate the value of the meta-model, we applied it to various scenarios, such as charitable donations and kindness in business. In a charitable donation context, for example, the meta-model can be used to design algorithms that facilitate transparency and trust, by providing clear and concise information about the impact of donations and how they are being used.

Our meta-model may be of value to the software engineering community as it captures a common, perhaps standardized, representation of kindness that can guide the design of systems with explicit psycho-social benefits. The meta-model also has the potential to enhance opportunities for automated interventions that promote kindness when using software systems. More generally, we suggest that the meta-model is a step towards more responsible software engineering[1] for the design of pro-social digital interventions.

The rest of our paper is organized as follows. In section 2, we discuss related work, primarily around kindness and the utilization of meta-models to facilitate its modelling. Then, in section 3, we introduce and describe our meta-model for representing kindness. In section 5, we discuss two scenarios in which we apply and demonstrate our meta-model, those scenarios being charitable donations and kindness in business. Next, we describe implications for software engineering in section 6. Finally, in section 7, we conclude and present future work.

2 RELATED WORK

2.1 Kindness

Kindness has been studied extensively in the fields of psychology and social science due to its many psycho-social benefits. It can be defined as a behavior that benefits others and/or the self, and while this informal definition is used in this paper, kindness has been described in various ways in the literature. Peterson and Seligman [31] describe kindness as an action driven by compassion or concern, while Lyubomirsky et al. [23] define it as a behavior that benefits others but comes at a cost to the self.

Acts of kindness (AoKs) are a manifestation of kindness in the world. An AoK is an action performed by a giver with the intention of benefiting a receiver [10, 23]. These acts can range from simple verbal expressions such as compliments to extraordinary physical gestures such as organ donation. AoKs have been found to have numerous benefits for individuals and society, including fostering wellbeing of givers and receivers, reducing anxiety and stress, increasing resilience, and promoting trust among people [6, 10, 15, 19–21, 39, 44].

Kindness is not just a single behavior but a complex psycho-social phenomenon driven by various factors. The quality of kindness is a deep and natural tendency that humans possess, which is influenced by a range of psychological and social factors [9, 10, 30]. Psychological factors include the emotional state of a person and personality traits such as agreeableness [5, 7]. Social factors include the level of relatedness to the receiver, whether it be family, community, or strangers, as well as the level of need, whether it be emotional, instrumental, or health-related [30, 40].

2.2 Meta-Modelling

Meta-modelling has been recognized as an effective technique for creating domain-specific languages (DSLs) in software engineering [22, 27]. Such DSLs can be used to describe the requirements, design, and implementation of software systems, and they enable software engineers to create models that are easier to understand and manipulate [46, 47]. As a result, DSLs can facilitate faster and more effective software development.

Meta-modelling has also been recognized as a major step in the automation of software development processes [37]. By using meta-models to describe the requirements, design, and implementation of software systems, software engineers can automate the generation of software artifacts and reduce the need for manual intervention [12, 46]. This can lead to more efficient and effective software development, potentially reducing the risk of errors and inconsistencies in software artifacts.

The use of meta-models may also enable the integration of kindness into digital technologies. By providing a shared, perhaps standard, representation of kindness, it may be possible to design algorithms and systems that promote and recognize acts of kindness while still respecting the importance of genuine human interaction and empathy.

3 A META-MODEL FOR REPRESENTING KINDNESS

We present a novel meta-model for kindness, which serves as a foundational step towards its automation. Drawing from the literature in social psychology and related fields, our meta-model captures the essential entities and relationships that characterize kindness. By representing the critical actors, psychological and social factors, and actions that support the delivery of kindness opportunities,

[1] https://www.linkedin.com/pulse/software-without-boundaries-bashar-nuseibeh

we suggest that our meta-model will enable a systematic approach to modelling kindness. The aim of the meta-model is to forge a bridge between requirements of prosociality, as expounded upon in social psychology, and the processes of software system design and implementation.

Our approach to developing the meta-model involved a literature review of relevant studies in psychology that have systematically and scientifically examined kindness in recent years, such as those by Cotney et al. [9], Curry et al. [10], and Shillington et al. [39]. Based on our findings, we manually extracted the common entities and relations between the various definitions and uses of kindness found in the literature. We then developed a meta-model that describes these entities and relationships, and implemented it using the Unified Modeling Language (UML).

To support the implementation process, we used the Eclipse Modeling Framework (EMF)[2], which offers graphical tools for model creation, code generation, and easy extension of models for further automation. We also integrated multiple critical capabilities into the meta-model to enable increased automation, such as the ability to evaluate whether an opportunity can be classified as a kindness opportunity based on motivation. The meta-model and its implementation is available publicly[3]. A simplified version of the meta-model is presented in Fig.1.

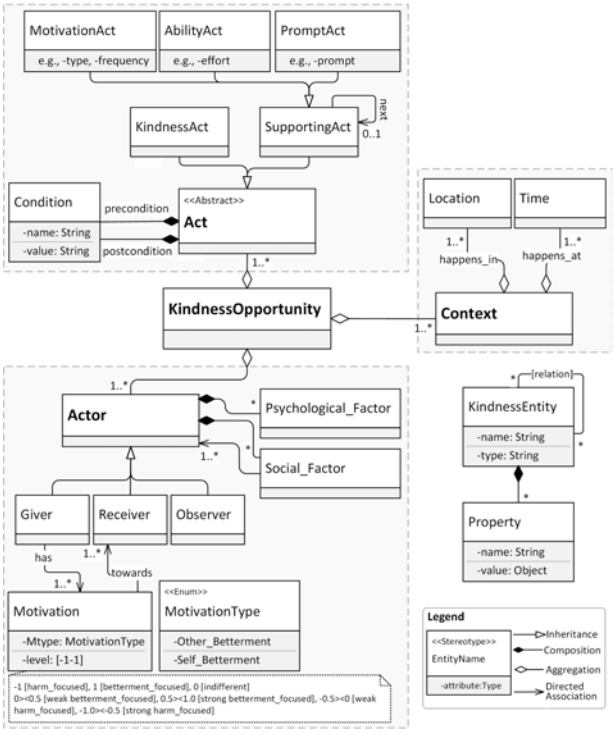

Figure 1: A meta-model for representing kindness. (simplified)

[2]https://www.eclipse.org/modeling/emf/
[3]https://github.com/kindness-metamodel/mm

In subsequent sections, we illustrate the various entities and relations found in the meta-model, elaborate on the capabilities integrated to enhance automation, and elucidate the various domains where it may find application.

3.1 Entities & Relationships

To help illustrate the entities and relations within the meta-model, we will employ the scenario of "sharing chocolate" as an act of kindness involving three individuals: Dara, Tom, and Sarah. Specifically, this act entails "Dara sharing chocolate with Tom in the kitchen, with Sarah coincidentally witnessing it while passing by."

At the heart of the meta-model an entity called *KindnessOpportunity*, which represents information about the opportunity itself, by capturing three main aspects of it: *Actor*, *Act* and *Context*. We expound on these three aspects in the following subsections.

All entities of the meta-model extend *KindnessEntity*, which is characterized by a unique *name* and a *type*, which can specify further the type of an entity (e.g., "Human", "Software"). *KindnessEntity* is also characterized by properties defined using the class *Property*. A *Property* has a unique *name* and a *value*. These properties can be used to provide further information on various entities of the meta-model, such as specifying a property for a *Location* with a type as a *name* and "kitchen" as a *value*.

3.1.1 Actor. An *Actor* is any agent (human or non-human) that is involved in a *KindnessOpportunity*. We identify three main *Actor*s: A *Giver*, which denotes the agent performing the act of kindness (Dara in our example); a *Receiver*, which refers to the agent at which the act is directed (e.g., Tom); and an *Observer*, which denotes an agent perceiving the act (e.g., Sarah).

We explicitly represent *Giver*'s *Motivation*—the driver for performing an act—*towards* a *Receiver*, since it has been highlighted in the literature to be of at most importance when it comes to doing and being kind (e.g., see [10, 23, 43]). *Motivation* is characterised by a type (*Mtype* in our meta-model), which can be: *Other_Betterment*, indicating that the act is performed to benefit others (e.g., help a friend move because he needs help); or *Self_Betterment*, which indicates that the act is done to benefit the person performing the act (e.g., help a friend move because you want to borrow money from him later). A *Motivation* is also characterized by a *level*, indicating the motivation's level of focus, which can be negative or positive, as well as low or high (we indicate this by choosing a value between -1 and 1, where -1, for instance can indicate [highly negative, i.e. harm_focused]). For example, Dara can have a *Motivation* of type *Other_Betterment* and a level that is *[highly positive, i.e. betterment_focused]* (e.g., with a value of 1). In other words, Dara is primarily focused on benefiting Tom. This can also be used to recognize unkind acts, for example, by representing the *Motivation* of the giver as of type *Other_Betterment* but with a level that is *[highly_negative, i.e. detriment_focused]* (e.g., with a value of -1).

A human *Actor* has *Psychological_Factor*s and *Social_Factor*s. Both of these entities identify factors that can have an effect on an individual's tendency towards being kind. *Psychological_Factor*s refer to internal aspects to an individual, such as emotionality, that influence an individual's kindness behavior; while *Social_Factor*s refer to societal conditions and circumstances that can affect their kindness behavior, such as social relatedness. Since there can be

many factors that influence kindness behavior, we provide a complementary meta-model to allow the capture of these factors and their use by the kindness meta-model. We describe the factors meta-model in section 3.2.

3.1.2 Act. A *KindnessOpportunity* need also to refer to *Act*s that are involved in it. An *Act* is a specific behavior or action performed by an *Actor*. We represent two main types of *Act*: *KindnessAct*s and those that can support them, i.e. *SupportingAct*s. *KindnessAct*s refer to acts that exhibit kindness between *Giver*s and *Receiver*s, such as Dara sharing its chocolate with Tom, while *SupportingAct*s refer to actions that can facilitate and lead to performing *KindnessAct*s, such as Dara reading about the importance of sharing before meeting Tom. Each *Act* is characterized by a *precondition*, describing the state of the world before performing the act, and *postcondition*, describing its state after performing the act. Both can be created using the entity *Condition*, which is characterized by a *name* and a *value*, which can refer to a specific formalism for representing the condition, such as Bigraphical Reactive Systems [29].

We identify three types of *SupportingAct*. *MotivationAct*s, those that increase the motivation of the *Giver* to perform kindness, such as highlighting to them the benefits of being kind. A *MotivationAct* can be characterized by various properties such as type (e.g., personal, social, functional), time, frequency and intensity (e.g., weak or strong). *AbilityAct*s, those that increase the *Giver*'s ability to perform kindness, such as providing chocolate vending machine in the kitchen of our example. An *AbilityAct* can target increasing abilities or decreasing efforts, for example, physical or digital ability/effort. *PromptAct*s refer to acts that notify the *Giver* to perform a kindness act, such as sending them an app notification or an email. These types are based on a widely used behavior model called "Fogg model" [14], which states that for a person to carry out a behavior they need to have "enough" *motivation* and *ability* (such as money, time, physical or digital ability), which if then *prompted* (by a notification or an email, for example) will likely perform the designated behavior. In our case, this means that to have a better chance of establishing a successful *KindnessOpportunity*, it needs to be accompanied by acts that can support and increase motivation and ability of the *Giver* and when both reach a certain level—termed the "action line" in the Fogg model [13]—, suitable prompts can be delivered, such as app notifications, emails, or an itch to the hand [26]. *SupportingAct* is characterized by the relation *next* which defines partial order between the three types. The main constraint is that *PromptAct*s should be at the end of any sequence. *MotivationAct* and *AbilityAct* do not have constraints on their order.

3.1.3 Context. The third key aspect of the meta-model is *Context*, which refers to the surrounding environment in which a *KindnessOpportunity* takes place. We characterize *Context* by two fundamental entities, *Location* and *Time*. The space and time in which an opportunity takes place have a crucial role on its success. For example, Dara might want to share her chocolate with Tom while they are in the kitchen (location) after lunch (time). *Location* can refer to any place in the real-world or the cyber-world and could be specific (e.g., kitchen) or relative (e.g., a place that has both the giver and receiver present); while *Time* can be absolute (e.g., 1pm) or relative (e.g., after lunch).

While the entities and relationships of the meta-model allow the capture of individual, point-in-time acts of kindness, these acts serve as the fundamental building blocks for kindness in people and places over longer periods of time. Such acts are usually essential for establishing initial behaviors that can be recognized and reinforced. Our meta-model then enables the analysis, evolution, and sustainability of kindness over time by providing the means to develop strategies to handle them.

Our meta-model includes elements that allow for the integration of new data over time such as new social or psychological factors, which can be captured by our complementary meta-model for representing such factors. Furthermore, while point-in-time acts of kindness are crucial for initiating behavior, our meta-model outlines strategies for habit formation, by relying on widely used behavior models such as the aforementioned Fogg model. Finally, capturing individual kindness acts via the meta-model creates opportunities for software engineers to generate datasets that others can utilise. These datasets can be instrumental in building strategies and approaches that account for kindness over the long run.

3.2 Complementary Meta-Model: Representing Psychological & Social Factors

Given the vast number of factors that can influence kindness behavior, we conducted a thorough review of the literature to identify the most salient factors. We captured these factors as a separate meta-model, which can be linked to our kindness meta-model using the two entities *Psychological_Factor*s and *Social_Factors* found in it. This approach allows for greater flexibility in including additional factors and facilitates the meta-model's extension and use by other researchers. The factors meta-model is shown in Fig. 2.

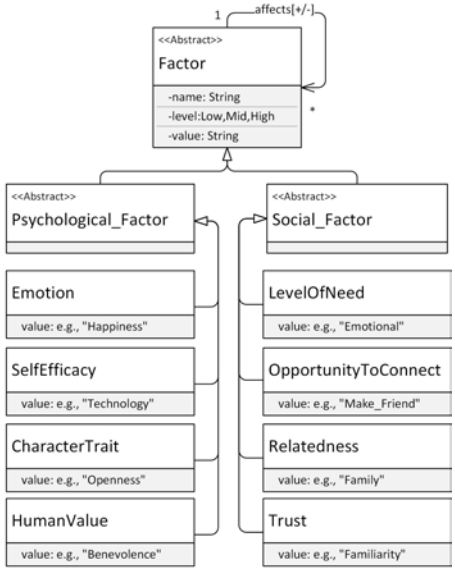

Figure 2: A meta-model for representing psychological and social factors.

The factors meta-model consists of the entity *Factor*, which refers to any type of factor that have influence on individual's kind behavior. The entity is characterized by a *name*, a unique label, *level*, indicating the intensity of the factor (Low, Medium, or High), and a *value*, referring to a specific value in a given factor (such as happiness or sadness as a value for emotion). It also defines a relation between *Factor*s, referred to as *affects* in the diagram. This relation can be used to establish how various factors can have an impact, positive or negative, on other factors. For example, having close ties between giver and receiver (social factor) can increase positive emotions in the giver (psychological factor).

The factors meta-model defines two types of *Factor*s: *Psychological_Factor*s and *Social_Factor*, which correspond to their counterparts in the kindness meta-model. For *Psychological_Factor*s, we identify four key factors that have been salient in the literature on kindness, which are: *Emotion*, *SelfEfficacy*, *CharacterTrait*, and *HumanValue*.

Emotion refers to the emotional state of an *Actor*, such as happiness, sadness, and its level, such as high or low. For example, Dara might have high *Emotion* of happiness, which led her to share her chocolate. Emotions have been cited as a main driver for human kindness [9, 10]. *SelfEfficacy* indicates beliefs that people hold about their capacity to control events in their lives. Having self efficacy, such as beliefs in expressing positive emotions, controlling negative emotions, and meeting others needs, can affect the tendency to perform kindness [4, 7]. For example, Dara believes she is capable of giving part of her chocolate to Tom. *CharacterTrait* indicates a person's dimensions of personality [24], which includes openness and agreeableness, two traits that have been found to be a major determinant in behaving kindly [7]. Finally, *HumanValue* indicates human values, i.e. guiding principles in people's life [36], which includes benevolence and universalism, two key values that have been found to be frequently associated with prosocial behavior and kindness [7].

Similarly, for *Social_Factor*s, we identify four key factors affecting an individuals kind behavior. These are: *LevelOfNeed*, *OpportunityToConnect*, *Relatedness* and *Trust*.

LevelOfNeed refers to the type of need of a *Receiver*, which can include Emotional (e.g., reach to someone feeling down), Instrumental (e.g., help someone move), or Health-Related (e.g., raise money for someone with cancer). For example, Dara wanted to share her chocolate because Tom was feeling down. *OpportunityToConnect* indicates possible opportunities to start, keep, or strengthen relationships with other people, such as making new friend, keeping old friend, or strengthening family ties. For example, Dara wanted to make a new friend by sharing her chocolate with Tom. *Relatedness* indicates the relationship between two *Actor*, such as family, friend, colleagues or stranger. For example, Dara and Tom can be colleagues. Finally, *Trust* indicates the level of trust between two *Actor*s, such as familiarity between them being high or low. For example, Dara and Tom can have low familiarity, and since kindness can increase trust between giver and receiver [20, 44], Dara wanted to share her chocolate with Tom; and hence, have a higher familiarity with Tom.

3.3 Capabilities to Enhance Automation

*MotivationAct*s that affect the overall *Motivation* of the *Giver* are crucial in determining whether a given opportunity qualifies as a *KindnessOpportunity*. To automatically assess the overall motivation of the giver, we developed a functionality in our meta-model that takes into account the impact of these *MotivationAct*s. The algorithm for this function is presented in Algorithm 1. The algorithm takes as input a list of *Motivation*s for the giver, denoted as M_g, and a list of *MotivationAct*s affecting the opportunity, denoted as M_a. The output is a boolean value indicating whether the opportunity qualifies as a *KindnessOpportunity*, i.e., if the other betterment is more than the self betterment. The algorithm first initializes the other betterment and self betterment to zero. It then iterates through each motivation in the giver's list, and if the motivation type is "other", it adds its value to the other betterment; if the motivation type is "self", it adds its value to the self betterment. Next, the algorithm iterates through each *MotivationAct*, and if the act increases the other betterment, it adds its value to the other betterment; if the act decreases the self betterment, it subtracts its value from the self betterment. Finally, the algorithm returns whether the other betterment is more than the self betterment.

Algorithm 1 Determine if an opportunity is a *KindnessOpportunity*

Require: A list of motivations M_g for the giver and a list of motivation acts M_a affecting the opportunity
Ensure: Whether the opportunity is a kindness opportunity or not

```
1:  function IsKindnessOpportunity(M_g, M_a)
2:      O_B ← 0                                   ▷ initialize other betterment
3:      S_B ← 0                                   ▷ initialize self betterment
4:      for all m ∈ M_g do                        ▷ consider giver motivations
5:          if m.type = other then
6:              O_B ← O_B + m.value
7:          else if m.type = self then
8:              S_B ← S_B + m.value
9:      for all m ∈ M_a do                        ▷ consider motivation acts
10:         if m.increaseOtherBetterment = true then
11:             O_B ← O_B + m.value
12:         if m.decreaseSelfBetterment = true then
13:             S_B ← S_B − m.value
14:     return O_B > S_B    ▷ return true if other betterment is more than
    self betterment
```

In order to aid in the determination of whether a prompt ought to be triggered, we devised a functionality within our meta-model to address this concern. The algorithmic process for this function is explicated in Algorithm 2. The algorithm for this functionality determines whether a prompt should be triggered based on the overall motivation and ability effects of the giver. The algorithm takes the giver's motivation, motivation acts, ability acts, and an action line as inputs, and outputs whether a prompt should be fired or not. The algorithm calculates the total motivation by looping through the motivation acts and adding or subtracting the act's value based on whether it increases other betterment or decreases self betterment. Similarly, the algorithm calculates the total ability by looping through the ability acts and adding or subtracting the act's value based on whether its effect is positive or negative. The

action score is then calculated via a given formula, which takes as input the total motivation and total ability. While we do not provide a specific formula within the meta-model, we enable users (such as software engineering researchers) to specify and refine their own formulas to suit their specific needs. If the action score is greater than or equal to the given action line, the algorithm returns that a prompt should be triggered, otherwise, it returns that a prompt should not be triggered.

Algorithm 2 Calculate if overall motivation and ability exceed action line, i.e. if they are enough to trigger prompts.

Require: G giver motivation, M MotivationActs' values, A AbilityActs' values, AL ActionLine
Ensure: Whether prompt should be triggered or not

function canTriggerPrompt(G, M, A, AL)
  $T_M \leftarrow G$ ▷ Set total motivation to giver's motivation
  for $m \in M$ **do** ▷ Loop through motivation acts
    if $m.incOther = \top$ **then** ▷ If act increases other betterment
      $T_M \leftarrow T_M + m.val$ ▷ Add act's value to total motivation
    else if $m.decSelf = \top$ **then** ▷ If act decreases self betterment
      $T_M \leftarrow T_M - m.val$ ▷ Subtract act's value from total motivation
  $T_A \leftarrow 0$ ▷ Set total ability to 0
  for $a \in A$ **do** ▷ Loop through ability acts
    if $a.eff =$ "pos" **then** ▷ If ability effect is positive
      $T_A \leftarrow T_A + a.val$ ▷ Add ability's value to total ability
    else if $a.eff =$ "neg" **then** ▷ If ability effect is negative
      $T_A \leftarrow T_A - a.val$ ▷ Subtract ability's value from total ability
  $AS \leftarrow f(T_M, T_A)$ ▷ Calculate action score using a given formula
  return $AS \geq AL$ ▷ return true if action score is greater than or equal to action line

4 SCOPE AND LIMITATIONS OF THE KINDNESS META-MODEL

The meta-model offers a structured approach for analyzing and fostering prosocial behaviors across different domains by breaking down kindness into three primary components: actor, act, and context. Its utility can extend across various domains, including personal interactions (the individual), organizational behavior (the group), and community engagement (the society).

Personal Interactions. At an individual level, the kindness meta-model offers insights into the motivations driving acts of kindness, the nature of these acts, and the contextual factors shaping interpersonal dynamics. It thus can facilitate self-reflection and promote intentional kindness in everyday interactions, fostering positive social connections.

Organizational Behavior. Within group settings, the meta-model can enable members to cultivate a culture of kindness. By examining the actor's intentions, the nature of their acts, and the group context, stakeholders can identify opportunities to promote collaboration, support members' well-being, and enhance group citizenship behavior.

Community Engagement. On a broader societal level, the kindness meta-model can serve as a guiding framework for designing interventions aimed at promoting collective well-being and social cohesion. By understanding the actor's role within the community, the nature of their contributions, and the contextual factors influencing community dynamics, stakeholders can implement targeted strategies to foster kindness.

Despite its potential versatility, the kindness meta-model is not without limitations. Some key considerations include: subjectivity, contextual complexity and behavioral dynamics.

Assessing kindness can be subjective, as perceptions of altruism and benevolence may vary across individuals and cultures. The model may need to account for diverse perspectives and value systems to ensure its applicability across different contexts.

Contextual factors influencing kindness can be multifaceted and dynamic, making it challenging to capture the full extent of their impact within the model. Stakeholders must exercise caution when interpreting findings and consider the broader socio-cultural context in which kindness unfolds.

Human behavior is complex and may not always conform to the model's theoretical framework. Actors may exhibit contradictory or unexpected behaviors, necessitating flexibility in applying the model and interpreting its results.

5 DEMONSTRATING THE META-MODEL: CHARITABLE DONATIONS AND BUSINESS

We illustrate how our meta-model can aid in facilitating the development and use of software applications within the realms of charitable donations (highlighting personal interactions and community engagement) and business (showcasing organizational behavior). Specifically, we will describe two software application scenarios that illustrate how our meta-model can enable the automation of kindness in these domains. These scenarios will showcase how our meta-model can be applied to capture the key entities and relationships involved in acts of kindness, as well as the psychological and social factors that facilitate them. By demonstrating the applicability of our meta-model in these real-world scenarios, we aim to provide practical insights into how it can be used to design and develop kind computing technology [2].

5.1 Charitable Donations

Charitable donations play a significant role in supporting various causes and social issues. However, the process of identifying and selecting charities, making donations, and tracking the impact of donations can be time-consuming and complicated for donors [17]. With the help of our kindness meta-model, software applications can potentially be developed and used to make the donation process easier and more effective.

Scenario. A non-profit organization that supports underprivileged children in developing countries wants to increase its charitable donations to expand their reach and impact. They decide to develop a software application that will enable people to donate easily and regularly.

The use of the kindness meta-model. Through the utilization of our meta-model, the application can be designed by appealing to the three key components of the meta-model, giving insights on how to develop and use the application. Below we describe how

the key components can be represented and used to support the development of the application.

Actors. The *Giver* can be a person who donates money to a charity or a cause (i.e. donor). The *Receiver* can be the charity or the cause that receives the donation (i.e. charity organization). The *Observer* can be individuals who may observe or learn about the donation (i.e. potential donors). These actors are shown in Fig. 3.

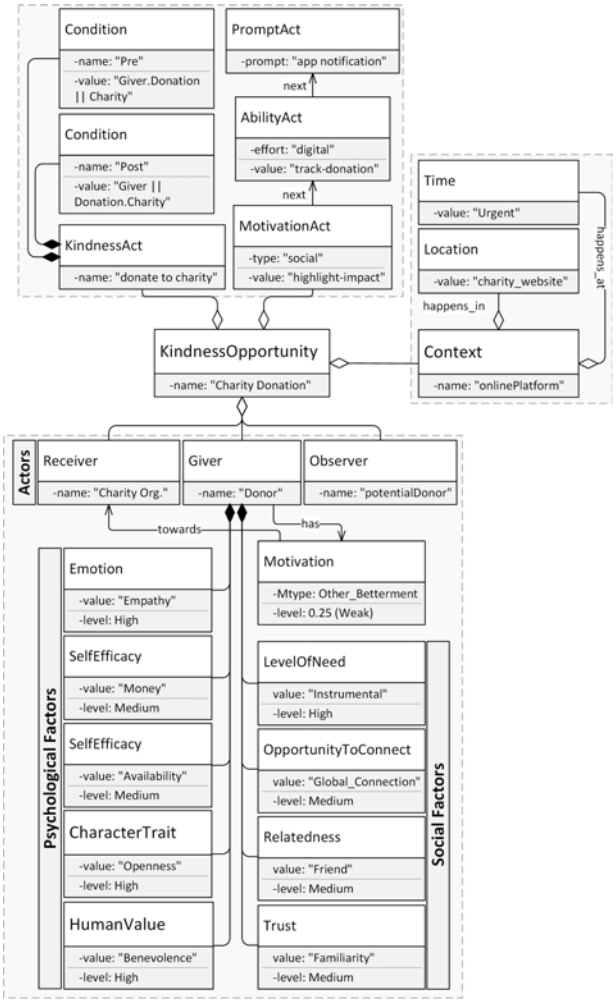

Figure 3: Our meta-model used to represent a *KindnessOpportunity* for charity.

Figure 3 shows also the most significant psychological and social factors that drive the giver to donate. **Psychological factors**. *Emotion*: the giver may strongly feel *empathy* for the people or cause they are donating to, which motivates them to give. *SelfEfficacy*: the giver may have both the financial resources and availability to make charitable donations. *CharacterTrait*: the giver may possess a strong sense of openness, which drives them to donate. *HumanValue*: the giver may prioritize values such as benevolence, which influence their decision to donate. **Social factors**. *LevelOfNeed*: the giver may perceive a high level of instrumental need for the cause they are donating to, which motivates them to give to alleviate the suffering of underprivileged children and enhance their wellbeing. *OpportunityToConnect*: the giver may feel a sense of belonging and global connectedness by donating to the cause. *Relatedness*: the giver may have a personal connection to the cause, such as having a friend who has been affected by the issue. *Trust*: The giver may trust the organization or platform they are donating through, feeling confident that their donation will be used effectively and ethically. However, the giver may have a slight motivation to benefit the charity, which supports underprivileged children.

Act. *Act*s involved in a charitable opportunity can be as follows. The *KindnessAct* is the act of donating money to the charity or the cause. It can be characterized by a precondition and a postcondition. The precondition can indicate that the Giver has a donation and the Charity is present (represented as "$Giver.Donation||Charity$" in Fig 3, where '.' denotes containment and '||' denotes that Giver and Charity may not be in the same location - the notation is borrowed from BigraphER [38], a tool that implements BRS [29]).

There can be a number of *SupportingAct*s to help deliver the act of charity, as shown in Fig. 3. *MotivationAct*: the giver could be motivated, for example, by the desire to make a difference to the society and the world. The application may use various strategies to increase the giver's motivation, such as highlighting the impact of previous donations. In this scenario, a *MotivationAct* is needed since the giver's own motivation is weak (indicated by the number 0.25 (placeholder value) in Fig. 3, i.e. the giver may be reluctant about donating to the charity). *AbilityAct*: the giver may face various barriers that limit their ability to donate, such as financial constraints, time constraints, or lack of information about the donation process. The application may use various strategies to increase the giver's ability to donate, such as providing easy and secure donation methods, offering various donation levels, or providing the ability to track donations. *PromptAct*: the giver may need reminders or nudges to initiate the donation process or to follow through with their intentions to donate. The application may use various strategies to prompt the giver to donate, such as sending personalized app notifications.

Context. The *Context* in which a charitable opportunity can take place can also be represented by the meta-model. *Location*: the donation may take place in virtual locations, such as an online donation platform. *Time*: the donation may occur at various times, such as during a specific campaign or event, on a recurring basis, or in response to a specific, urgent need.

Insights derived from the meta-model. Our meta-model can potentially support the development and use of a software application that enhances transparency and trust in charitable giving. The psychological and social factors captured by the meta-model can be incorporated into the software application development process. For example, in the context of requirements elicitation, the meta-model can be used as a structured framework guiding requirements engineers in extracting pertinent specifications. An illustration of this is the utilization of the "openness" *CharacterTrait*, inherent in some users categorized as *Givers*, to articulate a requirement indicating that the software application should facilitate novel experiences—such as introducing new charity events, causes, or beneficiaries—tailored to these users' inclination towards exploration and innovation. During software design, the application, for example, can use the giver's emotions of empathy and

human values of benevolence to design personalized suggestions for charities that align with their values. It can use degree of relatedness and level of need of the receiver to design strategies that prioritize and recommend charities in a more targeted way, such as providing forums or chat rooms for donors to discuss issues or share their experiences. It can also increase givers' trust in the donation process by providing transparency and accountability. For example, it could use blockchain technology to provide a secure and immutable record of all donations made, ensuring that donors can see exactly how their donations are being used. Finally, the meta-model's ability to represent context, such as location and time, can be used to provide information about the impact of donations and how they are being used. For instance, the application can provide real-time updates on how donations are being allocated and the progress being made towards the charitable cause.

5.2 Kindness in Business

In the business world, kindness is often overlooked or perceived as a weakness. However, research has shown that kindness can have positive impacts on employee well-being, productivity, and customer satisfaction [8, 18, 42]. Consequently, fostering kindness in business is necessary, and digital technology can play a significant role in facilitating this goal. By leveraging our meta-model, software engineers can design applications that take into account the actors, acts, and contexts involved in kindness in business. Such applications could include tools for recognizing and rewarding acts of kindness among employees, as well as promoting empathy and emotional intelligence in the workplace.

Scenario. The CEO of a company notices a lack of kindness and positive interactions among himself, the employees, and among the employees themselves. He acknowledges the harmful impact of incivility on productivity and employee well-being, which can lead to decreased performance and various mental and physical health issues [32, 35]. In response, the CEO endeavors to cultivate a *kind space* [3] within the workspace. This endeavor involves the development of a software application that promotes kindness and positive interactions in the workplace. The application could include features such as a recognition system for employees and a chat feature that encourages positive messaging between himself and the employees and between the employees themselves.

The use of the kindness meta-model. By using our meta-model, the CEO can ensure that the application appeals to key elements of kindness, which can be utilized to create more opportunities and promote it within the workspace. Below we describe how these elements can be represented and used to support the development of the application.

Actor. The *Giver* is the CEO who wants to promote kindness within the company culture and recognize employee's good work. The *Receiver* is the employees of the company who will benefit from the kind acts. The *Observer* is the broader public who may observe the company's kind acts and reputation. These actors are shown in Fig. 4.

The **Psychological_Factor**s that impact the giver's willingness to engage in acts of recognition include the feeling of gratitude towards employees (*Emotion*), a belief in the value of kindness (*Self-Efficacy*), and a desire for personal fulfillment through philanthropic work (*HumanValue*). The **Social_Factor**s include a realization of the employees' needs for recognition and support (*LevelOfNeed*), a sense of connection and responsibility towards the local community (*OpportunityToConnect*), and a desire to build trust and goodwill with the public (*Trust*). The giver has two main motivations: a motivation towards benefiting the employees (denoted by the value 0.5 [medium] and a motivation towards benefiting himself (denoted by the value 0.25 [weak]), as shown in Fig. 4.

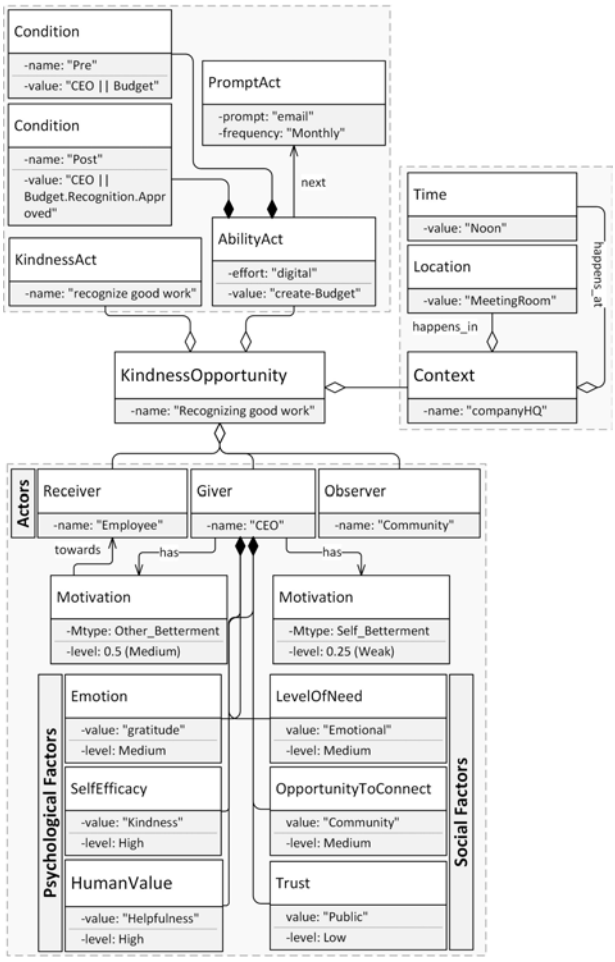

Figure 4: Our meta-model used to represent a *KindnessOpportunity* in business.

Act. The *Act*s involved in the kindness opportunity include a *KindnessAct* of recognizing the hard work of employees by giving out awards and bonuses. To facilitate this act, the meta-model allows for the inclusion of several *SupportingAct*s. *MotivationAct*s include creating a sense of purpose and meaning for the company, and enhancing employees' wellbeing, which reflects positively on their productivity. There is no need for *MotivationAct*s in this scenario as the giver is already assumed to be sufficiently motivated to benefit his employees. *AbilityAct*s involve providing the necessary resources, such as funding and staffing, to make the act of recognition a reality. The application could help the CEO create a

budget for employee recognition programs, allowing the allocation of funds for rewards and incentives. The precondition of an *AbilityAct* that helps setting up a budget could be, for example, that there is a company general budget; and a postcondition could be a recognition budget has been created and approved by the CEO (denoted by *CEO||Budget* and "*CEO||Budget.Recognition.Approved*", respectively, as shown in Fig. 4). Finally, the *PromptActs* involve regular communication and reminders to ensure that the act is carried out in a timely and effective manner. The application may send email reminders to the CEO to recognize employees on a regular basis (e.g., once a month).

Context. The *Context* in which kindness takes place includes the physical *Location* of the company, such as a meeting room, and the *Time* frame in which the acts will occur, such as noon or company-wide event held at the company headquarters.

Insights derived from the meta-model. The use of our meta-model can provide valuable insights into the development of a software application aimed at promoting kindness between a CEO and their employees. For example, the motivation component of the meta-model can be used to ensure that the application encourages both parties to engage in acts of kindness by highlighting the benefits of such behavior, such as increased job satisfaction and team cohesion. The ability component can be utilized to ensure that the application provides opportunities for the CEO and employees to engage in acts of kindness, such as setting reminders for acts of kindness or providing suggestions for ways to show appreciation. The prompt component can be used to ensure that the application prompts the CEO and employees to engage in acts of kindness at appropriate times, such as on work anniversaries or after a successful project. Additionally, the location and time component can be used to ensure that the application suggests acts of kindness that are appropriate for the workplace and that do not disrupt productivity. By incorporating these key elements of the meta-model, a software application can be developed to support the promotion of kindness and gratitude within a business context.

6 IMPLICATIONS FOR SOFTWARE ENGINEERING

Our kindness meta-model can underpin tools for designing algorithms and systems that promote and recognize acts of kindness. By incorporating this meta-model into software development processes, software engineers can potentially create technologies that not only meet functional requirements but that also promote positive social interactions and foster a more inclusive use of software systems.

The software engineering community may also benefit from incorporating the meta-model of kindness in its research and practices. The meta-model can be applied directly and indirectly. Directly, by implementing techniques that use it to build strategies for fostering kindness behaviour, such as a system that matches volunteers with community service opportunities based on their skills and availability. Indirectly, by serving as a guide for eliciting requirements for kindness-focused systems. For example, during requirement gathering, the meta-model can guide developers to consider factors such as user motivation, potential barriers, and the necessary prompts and abilities that encourage kindness. By considering the role of kindness in software engineering processes and outcomes, researchers and practitioners can promote a more collaborative, supportive, and inclusive software engineering culture - a step towards more reflective, responsible software engineering.

However, it is essential to consider the ethical implications of incorporating the meta-model into software design. For example, algorithms that prompt users to engage in acts of kindness could be perceived as manipulative or intrusive. It is therefore crucial to ensure that such interventions are designed with due consideration of user awareness, autonomy and consent.

Cultural sensitivity is also crucial when designing interventions that promote kindness. Acts of kindness may be perceived differently across different cultures, and it is important to ensure that interventions do not inadvertently perpetuate cultural biases or stereotypes.

Finally, software engineers should consider the potential unintended consequences of promoting kindness in automated systems. For instance, a system that automatically recognizes and rewards acts of kindness may inadvertently incentivize performative kindness or discourage genuine kindness acts that are not recognized by the system. Therefore, careful consideration of the intended and unintended consequences of kindness interventions is necessary to ensure that they have the desired impact.

7 CONCLUSIONS & FUTURE WORK

We presented a meta-model for kindness and its potential applications in software engineering. Our meta-model emphasizes the importance of considering different aspects of kindness, including psycho-social factors, acts and context, and provides a structured approach for designing and implementing kindness-related interventions in digital systems. However, modelling kindness has limitations. The complexity of human psychological and social interactions makes it difficult to fully capture kind behaviour, risking the reduction of kindness to formulaic and superficial actions.

There are many opportunities for future work. Firstly, the proposed meta-model can be extended, particularly in terms of incorporating additional dimensions and features of kindness, and expanding the scope of its applications; for example, for developing measures to assess the impact of an act of kindness on its receiver. Secondly, empirical studies can and should be conducted to evaluate the effectiveness of the meta-model in promoting kindness and its impact on well-being. This evaluation can be achieved through various research strategies, such as the Runkel & McGrath's framework [1, 25, 41]. Thirdly, as the meta-model can facilitate automation, future work can focus on the development of software tools and platforms that implement the meta-model in practice. To this end, graphical frameworks, such as Sirius[4], can be utilized by software engineers to build graphical tools to facilitate the creation and management of kindness opportunities. Finally, given the importance of ethical and cultural considerations in this research area, future work should also focus on incorporating diverse perspectives and addressing potential biases in the design of these interventions.

Kindness is an essential element of the human experience, and we believe that promoting and modelling it explicitly is a worthwhile

[4] https://www.eclipse.org/sirius/

goal for all responsible software engineers. Our meta-model for kindness may contribute to addressing this goal. We look forward to further developments and applications by the software engineering community.

ACKNOWLEDGMENTS

This work was supported by the Science Foundation Ireland grant 13/RC/2094_2 and UKRI grant EP/R013144/1.

REFERENCES

[1] R Abelson and RC Schank. 1977. *Scripts, plans, goals and understanding.* Hilladale NJ: Lawrence Erlbaum Associates.

[2] Faeq Alrimawi and Bashar Nuseibeh. 2022. Kind Computing. In *The 44th International Conference on Software Engineering: New Ideas and Emerging Results (ICSE:NIER).*

[3] Faeq Alrimawi and Bashar Nuseibeh. 2023. Requirements for Designing Kind Spaces. In *2023 IEEE 31st International Requirements Engineering Conference Workshops (REW).* 36–41.

[4] Albert Bandura, Gian Vittorio Caprara, Claudio Barbaranelli, Maria Gerbino, and Concetta Pastorelli. 2003. Role of Affective Self-Regulatory Efficacy in Diverse Spheres of Psychosocial Functioning. *Child Development* 74, 3 (may 2003), 769–782.

[5] Alixandra Barasch, Emma E. Levine, Jonathan Z. Berman, and Deborah A. Small. 2014. Selfish or selfless? On the signal value of emotion in altruistic behavior. *Journal of Personality and Social Psychology* 107, 3 (2014), 393–413.

[6] Monica Y. Bartlett and David DeSteno. 2006. Gratitude and prosocial behavior : Helping when it costs you. *Psychological Science* 17, 4 (2006), 319–325.

[7] Gian Vittorio Caprara, Guido Alessandri, and Nancy Eisenberg. 2012. Prosociality: The contribution of traits, values, and self-efficacy beliefs. *Journal of Personality and Social Psychology* 102, 6 (2012), 1289–1303.

[8] Joseph Chancellor, Seth Margolis, Katherine Jacobs Bao, and Sonja Lyubomirsky. 2018. Everyday prosociality in the workplace: The reinforcing benefits of giving, getting, and glimpsing. *Emotion* 18, 4 (2018), 507–517.

[9] Jessica L Cotney and Robin Banerjee. 2019. Adolescents' Conceptualizations of Kindness and its Links with Well-being: A Focus Group Study. *Journal of Social and Personal Relationships* 36, 2 (feb 2019), 599–617.

[10] Oliver Scott Curry, Lee A. Rowland, Caspar J. Van Lissa, Sally Zlotowitz, John McAlaney, and Harvey Whitehouse. 2018. Happy to help? A systematic review and meta-analysis of the effects of performing acts of kindness on the well-being of the actor. *Journal of Experimental Social Psychology* 76 (may 2018), 320–329.

[11] Julie J. Exline, Adrienne Morck Lisan, and Elianna R. Lisan. 2012. Reflecting on acts of kindness toward the self: Emotions, generosity, and the role of social norms. *Journal of Positive Psychology* 7, 1 (jan 2012), 45–56.

[12] Murni Fatehah, Vitaliy Mezhuyev, and Mostafa Al-Emran. 2021. A Systematic Review of Metamodelling in Software Engineering. *Recent Advances in Intelligent Systems and Smart Applications* 295 (2021), 3–27. https://doi.org/10.1007/978-3-030-47411-9_1/COVER

[13] BJ Fogg. 2007. Fogg Behavior Model. Last accessed on 03/05/2023. https://behaviordesign.stanford.edu/fogg-behavior-model.

[14] Bj Fogg. 2009. A behavior model for persuasive design. In *Proceedings of the 4th international Conference on Persuasive Technology.* 1–7.

[15] David A. Fryburg, Steven D. Ureles, Jessica G. Myrick, Francesca Dillman Carpentier, and Mary Beth Oliver. 2021. Kindness Media Rapidly Inspires Viewers and Increases Happiness, Calm, Gratitude, and Generosity in a Healthcare Setting. *Frontiers in Psychology* 11 (2021), 3938.

[16] Sandra Burri Gram-Hansen. 2021. Defining Features of Behavior Design: Considering Context. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, Vol. 12684 LNCS. Springer, 308–319. https://doi.org/10.1007/978-3-030-79460-6_24

[17] Caroline Greenhalgh and Paul Montgomery. 2020. A systematic review of the barriers to and facilitators of the use of evidence by philanthropists when determining which charities (including health charities or programmes) to fund. *Systematic Reviews* 9, 1 (aug 2020), 1–13. https://doi.org/10.1186/S13643-020-01448-W/TABLES/1

[18] Esther Howe, Jina Suh, Mehrab Bin Morshed, Daniel McDuff, Kael Rowan, Javier Hernandez, Marah Ihab Abdin, Gonzalo Ramos, Tracy Tran, and Mary Czerwinski. 2022. *Design of Digital Workplace Stress-Reduction Intervention Systems: Effects of Intervention Type and Timing.* Vol. 16. ACM.

[19] Bryant P.H. Hui, Jacky C.K. Ng, Erica Berzaghi, Lauren A. Cunningham-Amos, and Aleksandr Kogan. 2020. Rewards of kindness? A meta-analysis of the link between prosociality and well-being. *Psychological Bulletin* 146, 12 (2020), 1084–1116.

[20] Dorota Jasielska. 2020. The moderating role of kindness on the relation between trust and happiness. *Current Psychology* 39, 6 (jun 2020), 2065–2073.

[21] Jillian J. Jordan, David G. Rand, Samuel Arbesman, James H. Fowler, and Nicholas A. Christakis. 2013. Contagion of Cooperation in Static and Fluid Social Networks. *PLoS ONE* 8, 6 (jun 2013), e66199.

[22] Steven Kelly and Juha Pekka Tolvanen. 2007. *Domain-Specific Modeling: Enabling Full Code Generation.* John Wiley & Sons. 1–427 pages. https://doi.org/10.1002/9780470249260

[23] Sonja Lyubomirsky, Kennen M. Sheldon, and David Schkade. 2005. Pursuing happiness: The architecture of sustainable change. , 111–131 pages.

[24] Robert R. McCrae and Paul T. Costa. 1997. Personality Trait Structure as a Human Universal. *American Psychologist* 52, 5 (1997), 509–516. https://doi.org/10.1037/0003-066X.52.5.509

[25] Joseph E. McGrath. 1995. Methodology Matters: Doing Research in The Behavioral and Social Sciences. *Readings in Human–Computer Interaction* (1995), 152–169.

[26] Vikram Mehta, Arosha K. Bandara, Blaine A. Price, and Bashar Nuseibeh. 2016. Privacy itch and scratch: On body privacy warnings and controls. In *Proceedings of the 2016 CHI Conference Extended Abstracts on Human Factors in Computing Systems (CHI EA '16).* Association for Computing Machinery, 2417–2424. https://dl.acm.org/doi/10.1145/2851581.2892475

[27] Marjan Mernik, Jan Heering, and Anthony Sloane. 2005. When and how to develop domain-specific languages. *Comput. Surveys* 37, 4 (2005), 316–344. https://doi.org/10.1145/1118890.1118892

[28] Susan Michie, Maartje M. van Stralen, and Robert West. 2011. The behaviour change wheel: A new method for characterising and designing behaviour change interventions. *Implementation Science* 6, 1 (apr 2011), 1–12.

[29] Robin Milner. 2001. Bigraphical Reactive Systems. *CONCUR 2001 — Concurrency Theory* 2154 (2001), 16–35.

[30] Louisa Pavey, Tobias Greitemeyer, and Paul Sparks. 2011. Highlighting relatedness promotes prosocial motives and behavior. *Personality and Social Psychology Bulletin* 37, 7 (2011), 905–917.

[31] Christopher Peterson and Martin EP Seligman. 2004. *Character Strengths and Virtues: A Handbook and Classification.* Oxford University Press.

[32] Christine Porath and Christine Pearson. 2013. The price of incivility. https://doi.org/10.2469/dig.v43.n2.32

[33] Brian A. Primack, Ariel Shensa, Jaime E. Sidani, César G. Escobar-Viera, and Michael J. Fine. 2021. Temporal Associations Between Social Media Use and Depression. *American Journal of Preventive Medicine* 60, 2 (2021), 179–188.

[34] L Rowland and D. Klisanin. 2018. Cyber-kindness : Spreading kindness in cyberspace. *Media Psychology Review* 12, 1 (2018). http://mprcenter.org/review/cyber-kindness-spreading-kindness-in-cyberspace/

[35] Aaron C.H. Schat and Michael R. Frone. 2011. Exposure to psychological aggression at work and job performance: The mediating role of job attitudes and personal health. *Work and Stress* 25, 1 (2011), 23–40.

[36] Shalom H Schwartz. 2006. Basic human values: Theory, measurement, and applications. *Revue Francaise de Sociologie* 47, 4 (2006).

[37] Bran Selic. 2003. The pragmatics of model-driven development. *IEEE software* 20, 5 (2003), 19–25.

[38] Michele Sevegnani and Muffy Calder. 2016. BigraphER: Rewriting and Analysis Engine for Bigraphs. In *International Conference on Computer Aided Verification.* Springer International Publishing, 494–501.

[39] Katie Shillington. 2020. *Kindness as an Intervention for Student Social Interaction Anxiety, Resilience, Affect, and Mood: The KISS of Kindness Study II.* MSc. The University of Western Ontario.

[40] Judith G. Smetana, Marina Tasopoulos-Chan, Denise C. Gettman, Myriam Villalobos, Nicole Campione-Barr, and Aaron Metzger. 2009. Adolescents' and Parents' Evaluations of Helping Versus Fulfilling Personal Desires in Family Situations. *Child Development* 80, 1 (2009), 280–294.

[41] Margaret-Anne Storey, Neil A. Ernst, Courtney Williams, and Eirini Kalliamvakou. 2020. The who, what, how of software engineering research: a socio-technical framework. *Empirical Software Engineering 2020 25:5* 25, 5 (2020), 4097–4129.

[42] Michelle Thomason (Ed.). 2022. *Kindness in Management and Organizational Studies.* Emerald Publishing Limited. https://doi.org/10.1108/9781802621570

[43] Jennifer L. Trew and Lynn E. Alden. 2015. Kindness reduces avoidance goals in socially anxious individuals. *Motivation and Emotion* 39, 6 (dec 2015), 892–907. https://doi.org/10.1007/s11031-015-9499-5

[44] Milena Tsvetkova and Michael W. Macy. 2014. The Social Contagion of Generosity. *PLoS ONE* 9, 2 (feb 2014), e87275.

[45] Jean M. Twenge, Gabrielle N. Martin, and W. Keith Campbell. 2018. Decreases in psychological well-being among American adolescents after 2012 and links to screen time during the rise of smartphone technology. *Emotion* 18, 6 (sep 2018), 765–780.

[46] A Wąsowski and T Berger. 2023. *Domain-Specific Languages: Effective modeling, automation, and reuse.* Springer Nature.

[47] Nebi Yılmaz and Ayça Kolukısa Tarhan. 2020. Meta-models for software quality and its evaluation: A systematic literature review. In *International Workshop on Software Measurement and the 15th International Conference on Software Process and Product Measurement.*