

# Capturing and Organizing Reusable Interaction Practices Using Justification and Feature Models

Sébastien Mosser<sup>†</sup>, Shyam Ravichandran<sup>\*,†</sup>, Calvin Thuan-Phong Khuu<sup>†</sup>, Brian Detlor<sup>‡</sup>, Denise Y. Geiskkovitch<sup>\*</sup>, Anne-Marie Pinna-Déry<sup>§</sup>, and P.J. White<sup>\*\*</sup>

<sup>†</sup>McMaster Centre for Software Certification, Canada

<sup>\*</sup>HuRoN Lab, McMaster University, Canada

<sup>‡</sup>DeGroote School of Business, McMaster University, Canada

<sup>§</sup>I3S Laboratory (CNRS), Université Côte d'Azur, France

<sup>\*\*</sup>designCORE, South East Technological University (SETU), Ireland

## ABSTRACT

Engineering teams working on complex socio-technical systems frequently rely on tacit knowledge when addressing stakeholder-specific needs, hindering systematic reasoning about design choices. In this paper, we report on a qualitative case study within the EMPOWRD project to observe how best-practice knowledge can be captured using justification diagrams, a safety-inspired argumentation formalism. We explore the organization of these practices as a configurable product line using feature models to manage their variability. By applying well-defined composition operators, we demonstrate how assembling models from selected practices can automate the construction of coherent argumentation structures. Our observations within the EMPOWRD project, specifically regarding interactions with older adults, illustrate how this integrated modelling approach enables the systematic reuse and justification of practices across different experimental scenarios.

**KEYWORDS** Model composition, Software Product Line, Argumentation models, Older Adults.

## 1. Introduction

Developing complex socio-technical systems requires balancing technical constraints with evolving stakeholder expectations. While safety-critical domains rely on rigorous standards such as ISO 26262, many software engineering fields rely on experiential best practices. However, these practices often remain tacit, making their underlying rationale difficult to scrutinize, reuse, or adapt. As system complexity increases, this reliance on implicit knowledge leads to unforeseen interactions, hidden assumptions, and poor traceability. While software product lines (Kang et al. 1990) offer a structured way to manage variability, they often lack the means to capture the detailed rationale

behind modelled elements (Gutiérrez-Fernández et al. 2024). Conversely, safety-inspired assurance cases (Chowdhury et al. 2017) provide structured argumentation but are rarely integrated with variability management to handle collections of practices.

In this paper, we report on a qualitative case study conducted within the EMPOWRD project<sup>1</sup>, a \$1M initiative funded by *McMaster Institute for Research on Ageing* (MIRA) to explore older adults' wellness through digital inclusion. Our contribution is to examine how argumentation models (using Justification Diagrams) and software product lines (using Feature Models) can be combined into a consistent framework for modelling and managing heterogeneous practices captured by *Subject Matter Experts* (SMEs). We demonstrate the feasibility and utility of this approach using the participation of stakeholders and external auditors involved in EMPOWRD.

In Section 2, we describe the case under study within EMPOWRD, focusing on practices observed over two years while working on digital literacy training. In Section 3, we report on

### JOT reference format:

Sébastien Mosser, Shyam Ravichandran, Calvin Thuan-Phong Khuu, Brian Detlor, Denise Y. Geiskkovitch, Anne-Marie Pinna-Déry, and P.J. White. *Capturing and Organizing Reusable Interaction Practices Using Justification and Feature Models*. Journal of Object Technology. Vol. 25, No. 3, 2026. Licensed under Attribution - NonCommercial - No Derivatives 4.0 International (CC BY-NC-ND 4.0) <http://dx.doi.org/10.5381/jot.2026.25.3.a14>

<sup>1</sup> <https://mdtrc.mcmaster.ca/empowrd/>

the related work for the kind of interactions with older adults we are considering, as well as how software modelling techniques are classically used in the state-of-practice to address the capture and organization of practices. Section 4 describes the technical background used to support our approach, justification diagrams and feature models. We then present in Section 5 how to capture the rationale for each practice using argumentation models, and how composition operators can assemble practices into a consistent argument when multiple are used. Then, Section 6 describes how feature models can support organizing a collection of practices, and guide their composition. Finally, Section 7 discusses limitations and perspectives, and Section 8 concludes the paper.

## 2. The EMPOWRD case

The EMPOWRD *Major Program of Research* is an interdisciplinary initiative (2024–2028) investigating how digital inclusion improves older adults’ mobility, social connection, and well-being. The program is structured into six research dimensions operating in real-world settings. Among these dimensions, the “*meta*” dimension captures qualitative evidence from the others (such as interviews, observations, and study artifacts), to model these real-world interactions using Justification Diagrams and Feature Models. This approach aims to support software engineers and business stakeholders to systematically reason about experimental designs, ensuring that implicit rationales are made explicit and reusable.

### 2.1. Site Description

To align with software engineering research norms, we follow the ACM SIGSOFT Empirical Standards (Ralph et al. 2020). We frame our investigation as an exploratory embedded multi-case study, where the EMPOWRD program serves as the primary case, and the digital literacy sub-case provides rich, site-specific detail. The methodology utilizes triangulation across diverse qualitative data sources to capture the justification models. In accordance with the standard, we prioritize the trustworthiness and credibility of our findings through team-based analysis (Deltor, White, Geiskovitch) and external audit by experts (Pinna-Déry), rather than seeking statistical generalizability.

The case study site described in this paper is centred on the research dimension led by Deltor within the broader EMPOWRD program. This specific dimension, investigated over two years (Deltor 2025; Abouei et al. 2024; Elgamal & Deltor 2024), addressed the community-based delivery of digital literacy training tailored for marginalized older adults in collaboration with Hamilton Public Library. Operating in real-world settings, the site served as a sub-case where researchers observed and facilitated the meaningful engagement of older participants with digital technologies. The environment produced rich qualitative evidence, including interview transcripts, direct observations of training sessions, and various study artifacts. These data sources provided the empirical grounding necessary to identify, capture, and validate the interaction practices and their justifications using the proposed modelling framework.

**Table 1** Example of Practice Specification: *Peer Buddy*

Practice	Peer Buddy
<b>Purpose</b>	Encourage confidence and comfort by pairing older adults, supporting each other socially and cognitively during tasks.
<b>Applicability Conditions</b>	Suitable when tasks do not require isolated individual responses and the participant pool is large enough to form pairs.
<b>Pre-conditions</b>	Participants must be willing and able to collaborate; pairing should consider comfort and communication skills.
<b>Procedure</b>	Form pairs with similar levels of familiarity or ability; provide a short briefing on mutual support; monitor pairs and assist as needed.
<b>Expected Benefits</b>	Reduces anxiety, increases task completion rates, and offers peer learning opportunities.
<b>Potential Risks</b>	One individual might dominate; periodically check interactions and rebalance pairs where necessary.
<b>Resource Impact</b>	Minimal — organizational effort required to create and monitor pairs.
<b>Supporting Evidence</b>	Peer-support approaches have been shown to reduce anxiety and promote successful learning in older adults (Strom & Strom 2011; Tsai et al. 2017). The benefits of peer companionship for social engagement are also well documented (Gleason et al. 2018).

### 2.2. EMPOWRD Practice Example: *Peer Buddy*

A recurring interaction practice identified through observations of digital literacy training is the *Peer Buddy* approach. This practice involves pairing older adult participants to support one another socially and cognitively during technical tasks. It was observed that this method is particularly effective in community settings where participants may not be tech-savvy, as it fosters an immediate cycle of trust and mutual assistance. Best practices are typically communicated through informal channels such as verbal discussions during team meetings or unstructured email exchanges. While these methods facilitate quick information sharing, they often leave the underlying justification implicit and difficult to scrutinize or adapt across different projects.

To move beyond this reliance on tacit knowledge, these practices are, at best, described in a structured specification table (TAB. 1). It defines the *Practice name* and *Purpose* to establish identity and intent, while *Applicability Conditions* and *Pre-conditions* set the necessary context and requirements for effective use. The *Procedure* provides operational instructions, supported by *Expected Benefits* and *Potential Risks* to clarify outcomes and mitigation strategies. Finally, the *Resource Impact* assesses feasibility, and *Supporting Evidence* anchors the practice in empirical observations and/or established literature to provide the argumentative grounding.

## 3. Related Work

### 3.1. Practices to Interact with Older Adults

According to the World Health Organization, the population of older adults is projected to reach 2.1 billion by 2050, with one in six people aged 60 or older as early as 2030 (World Health

organization 2024). While the fields of Psychology, *Human-Computer Interaction* (HCI) and Universal Design have long focused on solutions suitable for this demographic, it remains unrealistic to expect non-specialists to be well-versed in the nuanced challenges of interacting with older adults.

Practices for conducting technology-related research with older adults are well established within HCI but are primarily disseminated through published manuscripts. These papers provide experimental methodologies and reflections, yet they require the reader to manually extract specific practices and extrapolate their applicability to new, unique situations (Sayago 2019). This extraction process is particularly difficult for individuals outside the field who may lack the background to identify the subtle nuances that drive specific study decisions. Over the past few decades, research has established essential design heuristics, such as providing site maps, designing larger targets, and avoiding specific colours to accommodate age-related changes in vision and motor skills (Zaphiris et al. 2007). Further work has highlighted the specific challenges of recruiting older participants and the preferred methods for engaging this equity-deserving group (Ellis & Cochran 1999). However, because this knowledge is often embedded in large, narrative-heavy research papers, it is not available in a cohesive, easily searchable format for practitioners in other engineering domains.

Often, experimental design for older adults relies on general knowledge of age-related abilities (e.g., changes in vision, hearing, and mobility) or software design heuristics (Farage et al. 2012; Zaphiris et al. 2007). While helpful for building a final product, these approaches do not provide guidance on how to reuse or adapt established research protocols. Current literature highlights considerations such as session length, morning versus afternoon timing, and specific assessment types (Dickinson et al. 2007), as well as specialized requirements engineering methods, such as design thinking for seniors (Jussli & Gewalt 2021). Nonetheless, this information is often tailored for domain specialists, leaving it unclear to an outside engineer when or why to choose one assessment over another. To bridge this gap, engineers require explicit access to the rationale used by *Subject Matter Experts* (SMEs) to reach informed conclusions about structuring their own research and applying these insights effectively to their work.

### 3.2. Modelling Software Practices

Requirements engineering has long sought to capture the rationale behind design decisions to make reasoning explicit and analyzable. Early frameworks like IBIS, QOC, and DRL introduced structured representations for issues and alternatives (Kunz & Rittel 1970; MacLean et al. 1991; Lee 1997), while goal-oriented approaches such as KAOS, i\*, and Tropos extended this to model stakeholder intentions and trade-offs (Dardenne et al. 1993; Yu 1997; Castro et al. 2002). However, these methods often leave the justification for specific practices informal. Similarly, traceability research links requirements to implementation but typically assumes decisions are already justified, failing to represent underlying argumentative reasoning (Gotel & Finkelstein 1994; Ramesh & Jarke 2001). Consequently, the rationale

is often treated as static documentation rather than a reusable, composable asset.

Knowledge management research highlights the challenge of externalizing tacit knowledge. Approaches like the Experience Factory capture lessons learned (Basili et al. 2002), and method engineering seeks to tailor development practices to specific contexts (Brinkkemper 1996; Rolland 1998). Yet, best practices remain largely documented in informal checklists or narratives. While situational method engineering provides selection mechanisms, it focuses on procedural composition over the explicit justification of a practice's suitability (Henderson-Sellers & Ralyte 2010). This lack of formalization makes it difficult to validate or reason about practices when assumptions conflict, motivating the need for structured representations that support principled reuse.

In software safety engineering, modelling is primarily used to provide a structured, rigorous argument that a system is safe for its intended purpose. Traditional approaches use notation systems such as Assurance Cases to visually map how high-level safety goals are decomposed into specific strategies and supported by empirical evidence (Chowdhury et al. 2017). These models move beyond descriptive documentation by explicitly linking architectural decisions to safety requirements and potential hazard mitigations, allowing for a formal audit of the underlying reasoning.

Despite these foundations, research has yet to treat best-practice knowledge as a first-class, composable artifact. While the Object Management Group's *Semantics of Business Vocabulary and Business Rules* (SBVR) bridges the gap between experts and engineers, it focuses on operational constraints (the "what" and "how") rather than the underlying rationale (the "why") (Object Management Group 2017). Most current models remain project-specific, lacking mechanisms for systematic justification and for managing interactions. Consequently, the combination of argumentative justification, variability management, and formal composition to reason about practice sets remains largely unexplored, leaving the potential for principled reuse in complex sociotechnical systems untapped.

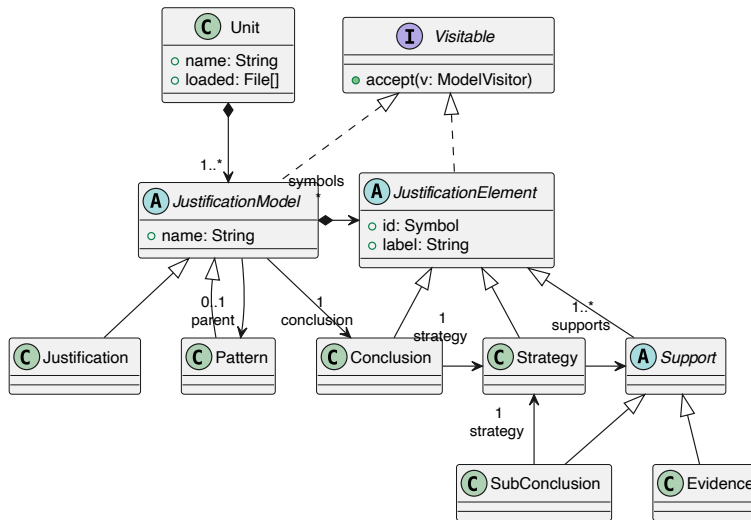
## 4. Approach Overview

Our work further develops argumentative approaches by capturing each methodological choice in a *Justification Diagram* (JD) format and using variability models such as *Feature Models* (FM) to organize these practices into reusable assets. This section provides the necessary background to understand both approaches. Artifacts used in the paper are available online<sup>2</sup>.

### 4.1. Capturing Practices: Justification Diagrams (JD)

Coined by Polacsek (Polacsek 2016), *Justification Diagrams* offer a lightweight approach to modelling assurance claims. Where classical safety cases are expressed using language like the *Goal Structuring Notation* (GSN) (SCSC Assurance Case Working Group 2021), JDs were created as a simplification of GSN-like approaches. The objective was to reduce the heavy

<sup>2</sup> <https://github.com/jpipe-mcscert/jpipe-examples/blob/main/empowrd>



**Figure 1** Meta-model for Justification Diagrams

safety component of GSN to offer a lighter language that supports its use outside the safety realm. As a consequence, JDs focus only on expressing argumentation diagrams: starting from pieces of *evidence*, we systematically apply *strategies* to reach a given *conclusion* logically.

Since their creation in 2016, JDs have been successfully applied to diverse domains, from healthcare (Duffau et al. 2018) to DevOps configuration pipelines (Mosser et al. 2023) and Large Language Model safety (Khuu et al. 2026). This approach is also adapted to capture good practices that support older adults, as one needs to claim (*i.e.*, “conclude” in JD terms) that the right practice is used in the right situation. The complete abstract syntax of the JD language is described as a UML class diagram in FIG. 1. In a nutshell, a justification `Unit` is composed of multiple `JustificationModel`s. A model can be a justification or a `Pattern`<sup>3</sup>, and always reach a single `conclusion`. To support a given conclusion, a `strategy` must be used to make explicit how such a conclusion can be reached. A strategy relies on `Support`ing elements, *i.e.*, atomic evidence (s) or sub-conclusion (s) obtained from more detailed argumentation.

For this paper, we are using JPIPE (McSCert 2023). Users can model their justifications using a textual syntax, as shown in LST. 1. The compiler can then be used to produce a graphical representation of the justification using Polacsek’s graphical syntax, as shown in FIG. 2. In this example, we capture the rationale for using a “peer buddy” system to enhance learning efficiency for older adults, as a first-class-citizen claim reified as a justification. According to this syntax, conclusions are represented as grey boxes, strategies as green parallelograms, and evidence as blue terminal leaves. For example, in the “peer buddy” practice, it makes explicit to the experiment designer that:

1. The pool of participants needs to be large enough. It is up to the designer to reflect on what “large enough” means

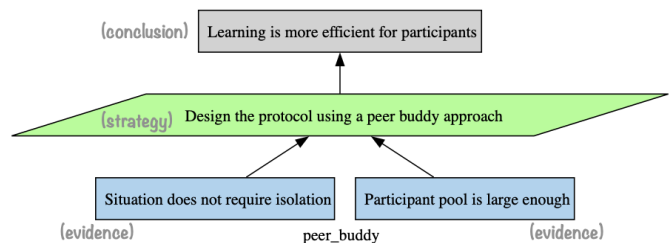
<sup>3</sup> Patterns are a way to express requirements compliance, and will not be covered in this paper.

**Listing 1** Concrete syntax for Justification Diagrams

```

1 justification peer_buddy {
2   conclusion learning is "Learning is more efficient for
   participants"
3
4   strategy design is "Design the protocol using a peer
   buddy approach"
5   design supports stress
6
7   evidence participants is "Participant pool is large
   enough"
8   participants supports design
9
10  evidence non_bias is "Situation does not require
   isolation"
11  non_bias supports design
12 }

```

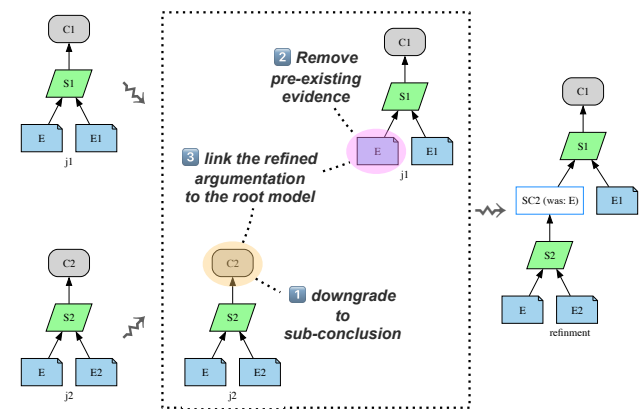


**Figure 2** “Peer Buddy” practice: Increase learning efficiency

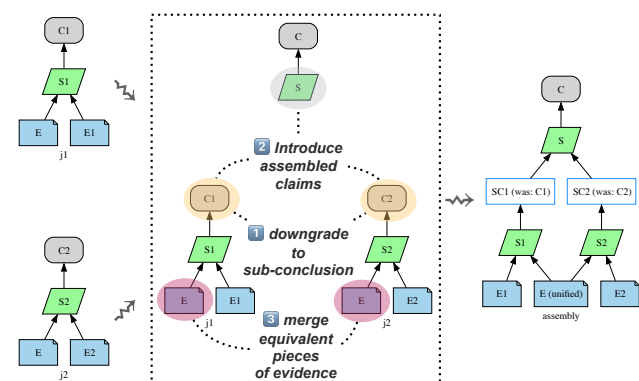
in their context, but to make the claim they expect from this practice, they must provide concrete evidence that this condition is satisfied in their experimental setup.

2. The experiment does not need participants to be isolated from one another. As in the previous case, it is up to the designer to instantiate this condition in their experiment, but, again, concrete evidence supporting the claim must be provided to reach the conclusion.

As a consequence, by selecting a practice, experiment designers obtain a *to-do list* of the things they need to exhibit (the different pieces of evidence), and a clear path forward (the



(a) Refinement Operator:  $refinement = refine(j_1, j_2, j_1.E)$

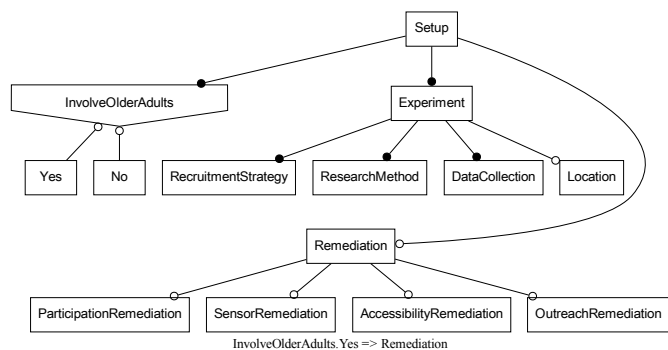


(b) Assembly Operator:  $assembly = assemble(\{j_1, j_2\}, c, s) \wedge j_1.E \equiv j_2.E$

**Figure 3** Composition mechanisms for justification diagrams

strategies) to reach their conclusion. The JPIPE environment relies heavily on the separation-of-concerns paradigm to support the expression of justification as reusable assets. As a consequence, it provides *composition operators* that one can use to compose elementary justification into more complex argumentation patterns (Chaudhari 2024). In this case study, we used two of such operators (FIG. 3):

- *Refine*: to refine evidence using an existing argumentation. In the previously described example, a designer might want to refine why they think that “The participant pool is large enough”. To do this, they would express their final justification as a refinement of the peer buddy one, specifying which justification needs to be used instead of the initial evidence (e.g., they can be reusing a participant pool which is considered large enough, or have taken particular care for their recruitment method).
- *Assemble*: to use multiple justifications to reach a new conclusion. In the previously described example, a designer might want to assemble multiple learning-efficiency methods tailored to their specific situation. As the assembly of individually relevant elements does not guarantee its relevance, users must explicitly state which strategy is used for this assembly and what the new conclusion is.



**Figure 4** High-level feature model for experimental setup (UVL tooling)

## 4.2. Organizing Practices: Feature Models (FM)

Assuming we can capture best practices as JDs, we still need to organize the practices in a way that supports their reuse. There are multiple ways of organizing such knowledge, e.g., ontologies, goals models, predicates. For this paper, we chose to use *Feature Models* (FMs) (Kang et al. 1990), a classical variability modelling approach coming from the product line engineering community. FMs are used to represent the common and variable parts of a system, graphically. At the formal level, they represent a logical formula as a tree, and formal methods such as *SAT-solving* can be used to analyze the tree (e.g., verifying its satisfiability, enumerating all possible configurations).

We depict in FIG. 4 an FM that captures an experimental setup at a high level of abstraction. In this model, we represent such a Setup (root feature) as a mandatory (—●) Experiment and an optional (—○) Remediation strategy for older adults participation. An Experiment is required to describe its RecruitmentStrategy, ResearchMethod, DataCollection, and can optionally describe its Location. The identified Remediation for older adults includes things related to ParticipationRemediation, SensorRemediation, AccessibilityRemediation, and OutreachRemediation. All these elements are optional, as their usage is contextual. On the left-hand side, we described an exclusive choice (represented by an inverted trapezium in FMs) to require the user to indicate whether older adults are involved in the experiment. If the answer is Yes, then they must indicate a remediation strategy, as per the logical constraints  $InvolveOlderAdults.Yes \Rightarrow Remediation$ .

One advantage of FMs is that tools can process them efficiently and at scale (Benavides et al. 2025). For this case study, we use the *Universal Variability Language* (UVL) (Sundermann et al. 2021) to express our variability models, and the FLAMAPY tool suite (Galindo et al. 2023) to process them. Running FLAMAPY on the model represented in FIG. 4 gives us the following information (among others): (i) the user can make 10 decisions (the leaves in this case), and (ii) it models a space of 66 different ways of configuring it. The tool also provides feedback on the model’s soundness, by ensuring it is satisfiable, does not contain any false optional features (features that were modelled as optional but are actually mandatory) or

dead features (features that can never be selected in a valid configuration).

In SEC. 6, we will refine this high-level experimental setup model, which is used as a skeleton for aggregating knowledge coming from *Subject Matter Experts* (SMEs).

## 5. Capturing Practices

Conducting research with older adults can present different challenges than other populations, especially for individuals who lack training for this purpose (Lachapelle-Dagenais et al. 2021). Good practices can come from various sources, such as reference books (Sayago 2019) or individual knowledge shared by researchers. In this section, we focus on the knowledge derived from the previously described case study site.

### 5.1. Modeling Atomic Practices

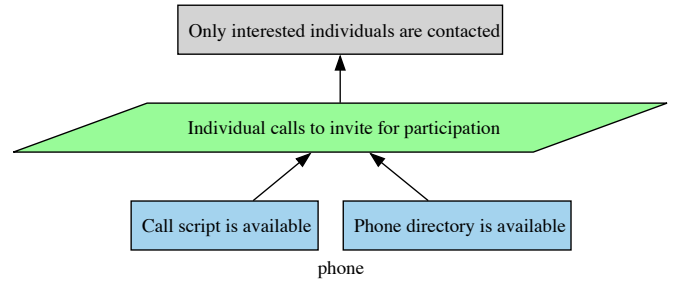
In this subsection, we focus on practices related to participant recruitment. Even if not entirely specific to older adults, we use this example to demonstrate that what might initially look simple can be more complex than expected (if the goal is to do it properly, *e.g.*, when preparing an ethics approval application). There are multiple ways to recruit participants for a given experience. For the sake of concision, we will only consider three in this section: (i) phone campaign, (ii) poster display, and (iii) on-site discussions. FIG. 5 depicts the three justification models for these practices.

First, it is worth noting that each practice leads to a different conclusion. Using a phone-based campaign will help achieve the objective of pre-selecting interested participants, as one will immediately know during the call whether the potential participant is interested in the study. On the other end of the spectrum, displaying posters serves a different purpose, raising awareness of the study among a broader audience. In the end, the chosen method depends on the experiment under design, but capturing the ultimate rationale for each method as an argument makes the objective explicit.

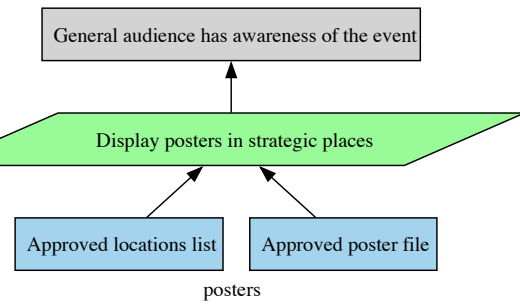
Another benefit of capturing practice as argumentation patterns is that it explicitly exhibits the different pieces of evidence necessary to support the practice. For example, if selecting the posters practice, the argumentation model forces the experiment designer to provide two pieces of evidence to support their practice: (i) a list of approved locations, and (ii) an approved poster file. By itself, the practice does not require the experiment designer to be explicit about how the poster file or the location list was approved; it only requires them to present the associated clues, ensuring they are using the practice with sufficient assumptions. The refinement operator can be used to provide more information if needed (as described in SEC. 5.3).

### 5.2. Assembling Practices for a Given Experiment

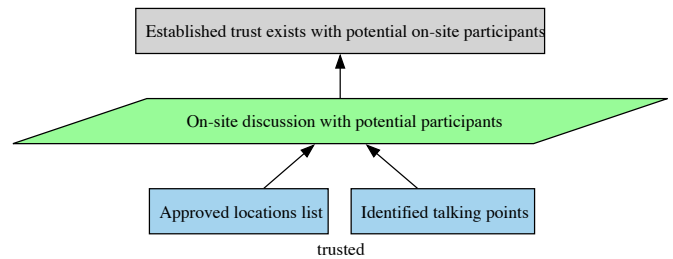
Assuming recruitment practices are captured as described in the previous section, an experiment designer might consider combining multiple of these practices for their experiment. To do so, JPIPE provides a composition operator named *assemble*. It allows one to select an arbitrary number of justifications and express the strategy they use to reach their new conclusion.



(a) Calling participants (phone)



(b) Displaying posters (posters)



(c) On-site discussions to maximize trust (trusted)

**Figure 5** Capturing atomic practices for participant recruitment

In our case, the three previously described practices were combined to create a participant pool large enough for the experience ( $N = 18$ ). To achieve such a task, assuming the three practices are modelled as independent justifications, the experiment designer describes their recruitment practice as an assembly of existing justifications. To create such an assembly, the user must specify the conclusion they want to reach by combining these justifications, as well as the strategy they use to do so in a relevant way.

Considering  $\mathcal{J}$  the space of justification models,  $\mathcal{C}$  the space of conclusions and  $\mathcal{S}$  the space of strategies, the *assemble* operator is specified with the following signature:

$$\text{assemble} : \mathcal{J}^n \times \mathcal{C} \times \mathcal{S} \rightarrow \mathcal{J} \\ (\{j_1, \dots, j_n\}, c, s) \mapsto j$$

The operator ensures the following properties in the result:

- The provided conclusion  $c$  is the final conclusion of the resulting justification  $j$ ;
- The provided strategy  $s$  is used to reach conclusion  $c$ ;

## Listing 2 Combining recruitment practices with the assemble operator

```

1 // Atomic off-the-shelf practices
2 justification phone { /* ... */ }
3 justification posters { /* ... */ }
4 justification trusted { /* ... */ }
5
6 // Composition operator call: assemble
7 composition {
8   justification recruitment is
9   assemble(phone, posters, trusted) {
10    conclusion: "Potential participant pool is large
11    enough"
12    strategy: "Combining recruitments methods"
13 }

```

- All conclusions from the composed justifications  $\{j_1, \dots, j_n\}$  are demoted to sub-conclusions (white squares with blue border), and used to support the provided strategy  $s$ ;
- Equivalent pieces of evidence in  $\{j_1, \dots, j_n\}$  are unified.

Assuming  $\{phone, posters, trusted\} \in \mathcal{J}^3$ , building the recruitment strategy is achieved as the following:

$$c \leftarrow \text{“Potential participant pool is large enough”}$$

$$s \leftarrow \text{“Combining recruitment methods”}$$

$$recruitment \leftarrow assemble(\{phone, posters, trusted\}, c, s)$$

LST. 2 shows the concrete syntax provided by JPIPE to implement such a call. FIG. 6 depicts the resulting justification. One can observe that the “Approved location list” evidence from the `posters` and the `trusted` independent justification was identified as equivalent and unified in the final result.

**Equivalence classes (Match) & Heuristics.** When composing models, it is classical to use a “match and merge” approach: elements are *matched* according to heuristics to identify equivalence classes, and then such classes are reduced into one *merged* element (Clavreul 2011; Jeanneret et al. 2008). In the context of our case study, we empirically decided to use text-based similarity: two pieces of evidence  $e$  and  $e'$  are equivalent ( $\equiv$ ) iff their Levenshtein distance is under a user-defined<sup>4</sup> threshold  $T$ :  $e \equiv e' \Leftrightarrow levenshtein(e, e') \leq T$ . Overall, JPIPE allows one to use other equivalence classes if needed, by extending the `EquivalenceClass` concept, or reusing off-the-shelf ones (such as strict equality).

**Evidence Unification (Merge) & Cycles.** When elements are unified, there is a risk of introducing inconsistencies in models. For example, in the context of argumentation models, cycles are not allowed, as they would represent self-fulfilling prophecies. To ensure that the composed models are consistent, JPIPE only unifies pieces of evidence when assembling claims, which, by design, cannot introduce cycles. At the language level, the compiler would refuse to compile a cyclic justification.

<sup>4</sup> When not provided as a parameter, the default threshold is 90% of similarity between the two labels, allowing to deal with typos while not jeopardizing results.

## Listing 3 Calling the refine operator to provide more details

```

1 // Atomic off-the-shelf practices
2 justification build_directory { /* ... */ }
3 justification existing_directory { /* ... */ }
4
5 // Composition operator call: refine
6 composition {
7   justification phone_build is
8   refine(phone, build_directory) {
9     hook: "phone:directory"
10 }
11 justification phone_reuse is
12 refine(phone, existing_directory) {
13   hook: "phone:directory"
14 }
15 }

```

### 5.3. Refining Practices with Detailed Argumentation

As practices are captured independently, experiment designers’ needs can vary depending on several factors. For example, an ethics review board might require more information about the interactions with participants during phone calls and on-site recruitment. Regulations (*e.g.*, GDPR in Europe) would require specific care in the way the phone directory is built and maintained, as well as how the participant list is stored. In this section, we will use refinement related to the phone directory, considering that (i) it can be built from scratch or (ii) it can be reused when already existing, under certain conditions.

To achieve such a goal, one can rely on the *refine* operator to have a justification subsumed by another one. The operator takes two justification models and a *hook*, *i.e.*, the identifier  $h \in \mathbb{I}$  of the evidence that needs to be replaced by the detailed justification.

$$refine : \mathcal{J} \times \mathcal{J} \times \mathbb{I} \rightarrow \mathcal{J}$$

$$(original, sub, h) \mapsto j$$

The operator ensures the following properties in the result:

- The evidence  $h \in original$  does not exist in  $j$ ;
- The conclusion  $c \in sub$  is demoted to a sub-conclusion  $sc$ ;
- Strategies in *original* supported by  $h$  are now supported by  $sc$ ;
- Equivalent evidences in *original* and *sub* are unified in  $j$ .

Considering two justification models associated with building a phone directory (*build\_directory*) and reusing one according to given terms of use (*existing\_directory*), we can now refine the practice *phone* justified in FIG. 5a by justifying the way the phone directory is obtained:

$$i \leftarrow id(phone, \text{“Phone directory is ...”})$$

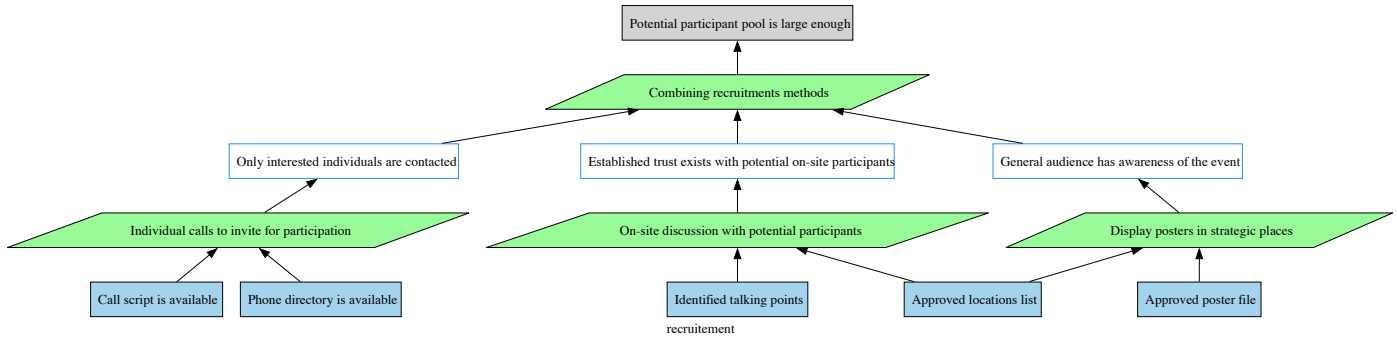
$$phone\_build \leftarrow refine(phone, build\_directory, i)$$

$$phone\_reuse \leftarrow refine(phone, existing\_directory, i)$$

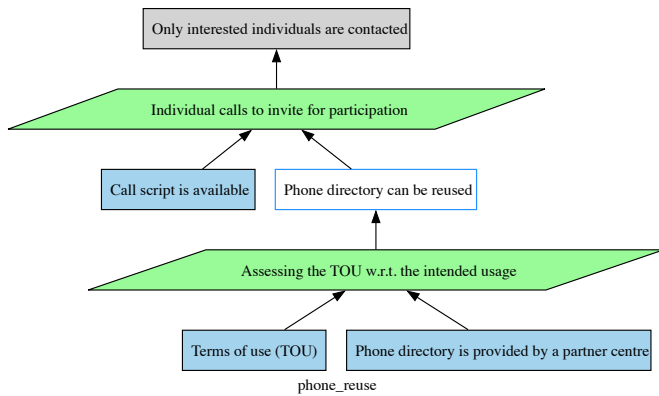
We provide in LST. 3 the concrete syntax used by JPIPE to call the refinement operator, and in FIG. 7 two alternative versions of the *phone* practice, depending on how it was refined.

### 5.4. Summary

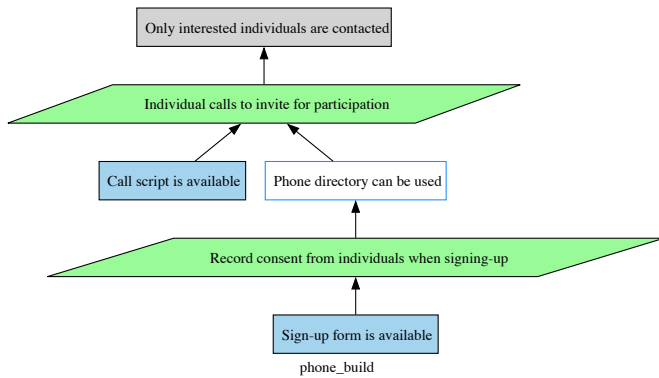
Using our approach, independent practices and their rationale can be captured as justification diagrams, providing actionable



**Figure 6** Combining phone, posters and on-site discussions as a recruitment strategy (LST. 2).



**(a)** Reusing an existing phone directory



**(b)** Building a phone directory from scratch

**Figure 7** Alternative refinements (LST. 3) for the phone practice (FIG. 5a).

insights for experiment designer who would want to reuse such practices. To support such a task, two composition operators are used to automatically compose justifications into more complex ones. The main advantage of using well-defined operators relies on the guarantees obtained on the result: final justifications are the result of the systematic application of operators rather than prone to human error.

## 6. Organizing Practices

The previous section focused on the identification of practices and their composition. The remaining (metaphorical) “*elephant in the room*” is to input all these practices into a framework that would organize them and make them available to experiment designers. In an ideal world, the experiment designer should not be exposed to composition instructions from the underlying JPIPE engine and should only make decisions at the practice level, automatically obtaining the resulting justification.

### 6.1. Approach Overview

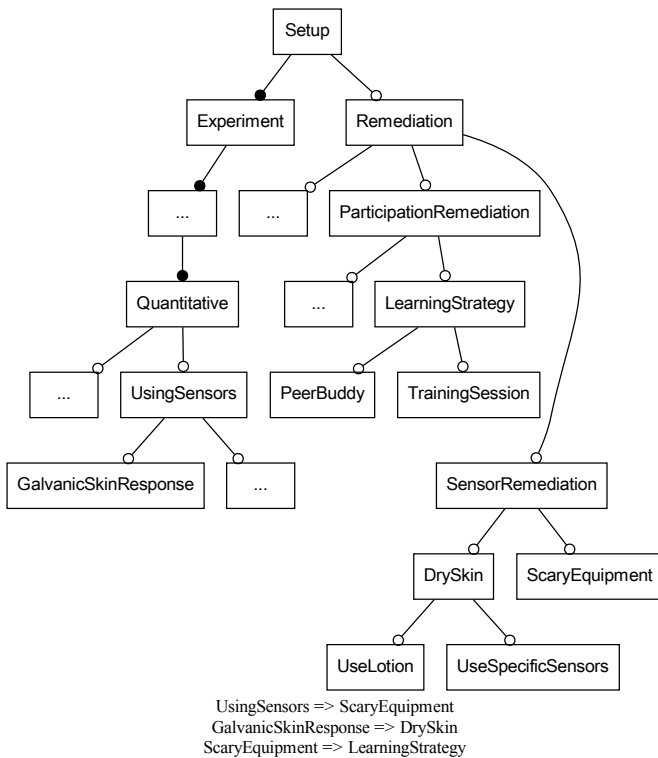
As described in FIG. 4, we consider “*How to design an experiment*” as being an independent domain, with dedicated *Subject Matter Experts* (SMEs) from this area. The key challenge here is to introduce older adult specificity (provided by other SMEs), without interfering with the experiment knowledge.

To achieve this, we consider the Experiment feature model to exist independently and work under the assumption that it will not be modified. This feature model organizes practices for designing empirical experiments. For example, it models the different DataCollection techniques one can use. Among these ones, experiments can collect data in a Qualitative or Quantitative way. Quantitative approaches can rely on Survey, or on Sensors, such as EyeTracking, EEG, or GalavanicSkinSensor. A subset of these features and their hierarchy is represented in FIG. 8, on the left-hand side under the Experiment feature.

We now represent knowledge from SMEs specializing in interactions with older adults. The elements reported here come from the case study, but at the conceptual level, the approach is not restricted to this project: the only requirement is to exhibit a mapping between the feature models and the atomic justification. While observing the digital literacy training dimension in EMPOWRD, we encountered the following situations:

- Issue #1: Using sensors such as EEG, eye trackers, or galvanic skin response sensors is perceived negatively<sup>5</sup>, and might scare older adults, or introduce noise in the collected data because of the induced stress;
- Remediation #1: Several approaches can be used to cope with this kind of stress, such as having researchers provide

<sup>5</sup> In EMPOWRD, using EEG sensors was perceived as extremely invasive by the participants, leading to an overall negative perception of the experimental setup.



**Figure 8** Subset of the Experiment/Remediation feature model

preliminary training to participants to show how the sensing technology works, or organizing orientation meetings to demystify the technology;

- Issue #2: Older adults have a tendency to have dry skin. In such cases, skin sensors calibration is harder, taking a long time and potentially inducing more stress. In some other cases, older adults tend to compensate for this dryness by using hydrating lotion, which can have an opposite effect on the sensors.
- Remediation #2: Remediation techniques can involve the use of specific sensors (e.g., using ring sensors instead of finger clips) that better tolerate these conditions, or instructing the participants to not use skin lotion on their own the day of the experiment and provide such lotion on site, under controlled conditions.

We represent this knowledge in a dedicated Remediation feature model. It organizes the identified remediation strategies as features. For example, PeerBuddy is a StressStrategy technique, which helps with participation. The DrySkin and ScaryEquipment ones are related to the use of sensors in the experiment. A subset of these feature and their hierarchy is represented in FIG. 8 (right-hand side).

To link both independent feature models when experiments involving older adults are designed, we use UVL capacity to import submodels into a bigger one. The Setup model is defined as the import of Experiment as mandatory, and Remediation as optional. In addition, constraints are used to link both in a non-invasive way. Using the previously described techniques,

we obtain the following constraints:

$$\begin{aligned} UsingSensors &\Rightarrow ScaryEquipment \\ GalvanicSkinResponse &\Rightarrow DrySkin \\ ScaryEquipment &\Rightarrow LearningStrategy \end{aligned}$$

Relying on engineering models to organize knowledge gives us the immediate advantage of analyzing how different practices are organized. Feature models are implemented as UVL models, and FLAMAPY is used to process them. By analyzing the model described in FIG. 8 (without the cosmetic “...”), FLAMAPY provides feedback on how the feature model can be used, and on its structural soundness.

From a usage point of view, we are interested in knowing how many decisions are exposed to the experiment designer, and how many different ways of combining the practices are valid:

- An experiment designer can make six decisions in terms of practices to be included in their experiment: GalvanicSkinResponse, PeerBuddy, TrainingSession, UseLotion, UseSpecificSensors, and ScaryEquipment. These decisions could be exposed to experiment designers through a website for example, acting as a wizard to assist in experiment design;
- Among the  $2^6 = 64$  potential combinations of these six choices available to the experiment designer, only 27 satisfy the structural constraints expressed by the SMEs and encoded in the feature model using cardinality and constraints. It represents only 42% of the possible usage of the practices, reinforcing the need to reify SMEs’ knowledge in their organization so that experiment designers can use the appropriate practices in a relevant way.

From a soundness point of view, we are focusing on the SMEs stakeholders more than the experiment designers themselves. It provides meta-information on the soundness of the feature model, offering feedback on how SMEs organize their catalogue of practices.

- The model is satisfiable, i.e., it is sound from a boolean logic point of view;
- The model does not contain any dead features.
- The model contains five false-optional features: Remediation, ParticipationRemediation, LearningStrategy, SensorRemediation, and ScaryEquipment. This is explained by the fact that only the GalvanicSkinResponse sensor method is defined in the experimental part of the model, so as this is the only one that can be defined on this part, everything optionally related to it becomes transitively mandatory. If this situation is a “glitch” here due to the small size of the example, it is helpful at scale to identify issues in the way SMEs have organized the knowledge.

Making decisions according to this feature model results in the instantiation of a so-called configuration. From a technical point of view in UVL, a valid configuration can be exported in a JSON file storing the selected features (see example in

#### Listing 4 Example of a valid configuration in JSON format

```
1 {
2   "decisions": {
3     "PeerBuddy": true, "UseLotion": true,
4     "GalvanicSkinResponse": true
5   },
6   "_transitive": {
7     "UsingSensors": true, "Experiment": true,
8     "Remediation": true, "StressStrategy": true,
9     "ParticipationRemediation": true,
10    "SensorRemediation": true, "Setup": true,
11    "ScaryEquipment": true, "DrySkin": true
12  }
13 }
```

Table 2 EMPOWRD feature models

Feature Model	#decisions	#constraint	#configuration
Experiment	29	4	1,030,080
Remediation	11	4	5,965
Complete	40	14	1,013,780,160

LST. 4). We attach justification models to each concrete feature, by maintaining a binding between the feature name and the location of the associated justification file.

### 6.2. Practice Portfolio identified by EMPOWRD

We report in FIG. 9 the current state-of-practice existing in the EMPOWRD project in the 2024 – 2025 period (phase 1). It is essential to note that this model is not absolute and represents a specific state of knowledge at a particular moment. Overall, the approach is independent of the source of this knowledge, as it focuses on capturing and organizing practices, regardless of their source. The Experiment feature model contains 29 end-user decisions, and 4 internal constraints (e.g., to use posters, one needs to ensure location availability). It describes a space close to one million possible configurations. The Remediation feature model offers 11 decisions to the end-user, and contains 4 internal constraints. It represents a space of close to six thousand different configurations. When combining both, we introduced 6 additional cross-model constraints to link them. Ultimately, it encompasses a vast space of nearly one billion potential combinations for all these practices. This number seems high, but it only represents an infinitesimal fraction (0.09%) of all possible configurations of 40 decisions ( $2^{40} = 1,099,511,627,776$ ).

### 6.3. Ensuring the consistency of selected practices

As mentioned when describing the study site, the EMPOWRD project was designed with a meta-dimension to capture knowledge and support its analysis

We consider *Avery*, an engineer working on another dimension of the project. When Avery observes the success of the Peer Buddy approach in Detlor’s digital literacy sessions, they recognize that the underlying rationale (reducing anxiety through social support) can be applied to co-designing environments for individuals with dementia (another dimension addressed by

#### Listing 5 Assembling off-the-shelf practices

```
1 load phone, posters, trusted from "recruitments.jd"
2 load loc_avail from "locations.jd"
3 load phone_reuse from "outreach.jd"
4 load peer_buddy from "remediation/stress.jd"
5
6 composition {
7
8   justification starter
9     is assemble(phone, posters, trusted) {
10      conclusion: "Potential ..."
11      strategy: "Combining ..."
12    }
13
14   justification temp is refine(starter, loc_avail)
15     { hook: "loc_available" }
16
17   justification recruitment is refine(temp, phone_reuse)
18     { hook: "directory" }
19
20   justification empowrd_design is
21     refine(peer_buddy, recruitment)
22     { hook: "peer_buddy:participant" }
23 }
```

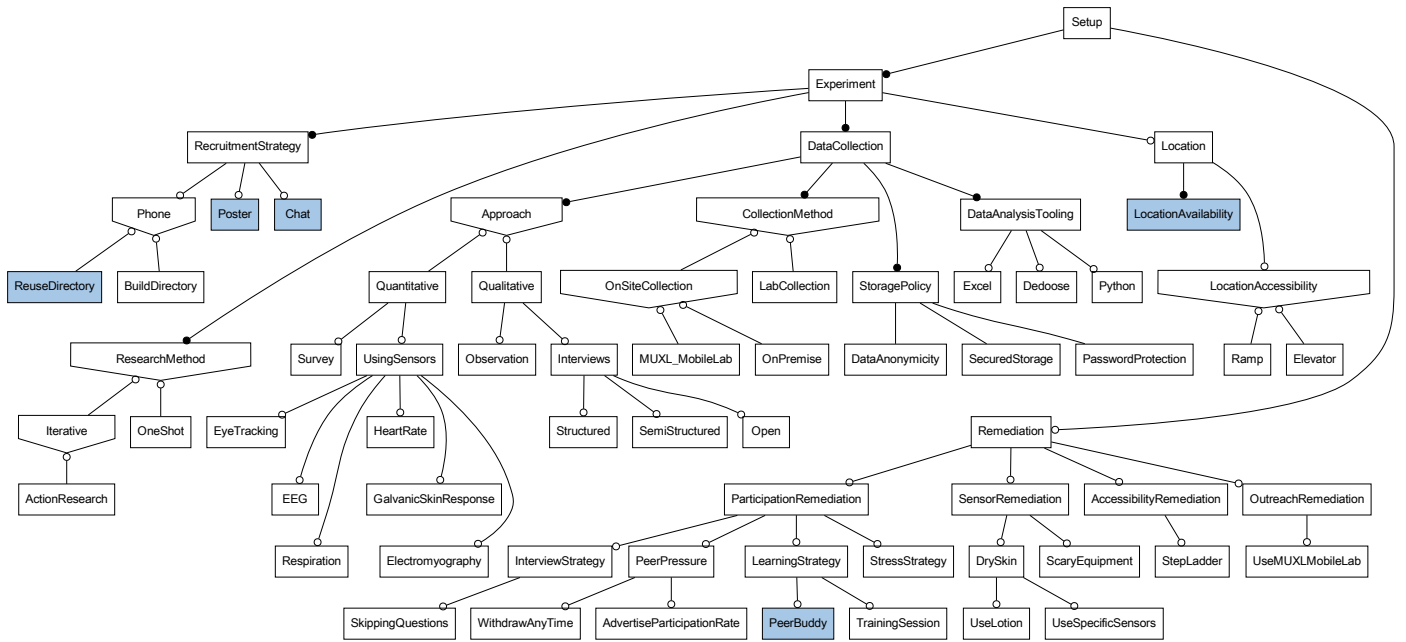
the project). Avery does not have formal training in HCI or in interacting with older adults. Without our contribution, Avery would have to build their campaign from scratch, with no confidence in the outcome. Using our approach, they can reuse the existing catalogue of practices, with confidence that the selected practices will be applied according to their underlying rationale. To illustrate this confidence, we consider the following scenario:

- ① Avery selects the PeerBuddy feature in the catalogue.
- ② As the FM contains the constraint *PeerBuddy*  $\Rightarrow$  *RecruitmentStrategy*, Avery is prompted to select a recruitment strategy;
- ③ Avery decides to use Poster, Chat and Phone strategies simultaneously;
- ④ As the FM defines Phone as an alternative, Avery must choose which approach they will use for their phone directory. They choose PhoneReuse.
- ⑤ As the FM defines a constraint *Poster*  $\vee$  *Chat*  $\Rightarrow$  *LocationAvailability*, Avery will have to consider LocationAvailability in their final campaign.
- ⑥ The configuration is now consistent with the FM. The system derives the justification associated with these decisions by loading them from a reference catalogue;
- ⑦ The justifications are combined<sup>6</sup> into the final justification for Avery (LST. 5);
- ⑧ Avery now has a list of eight pieces of evidence that they must produce to implement a *peer buddy* approach properly for their campaign. Following the argumentation described in FIG. 10, they now have confidence in implementing the *peer buddy* approach in their campaign.

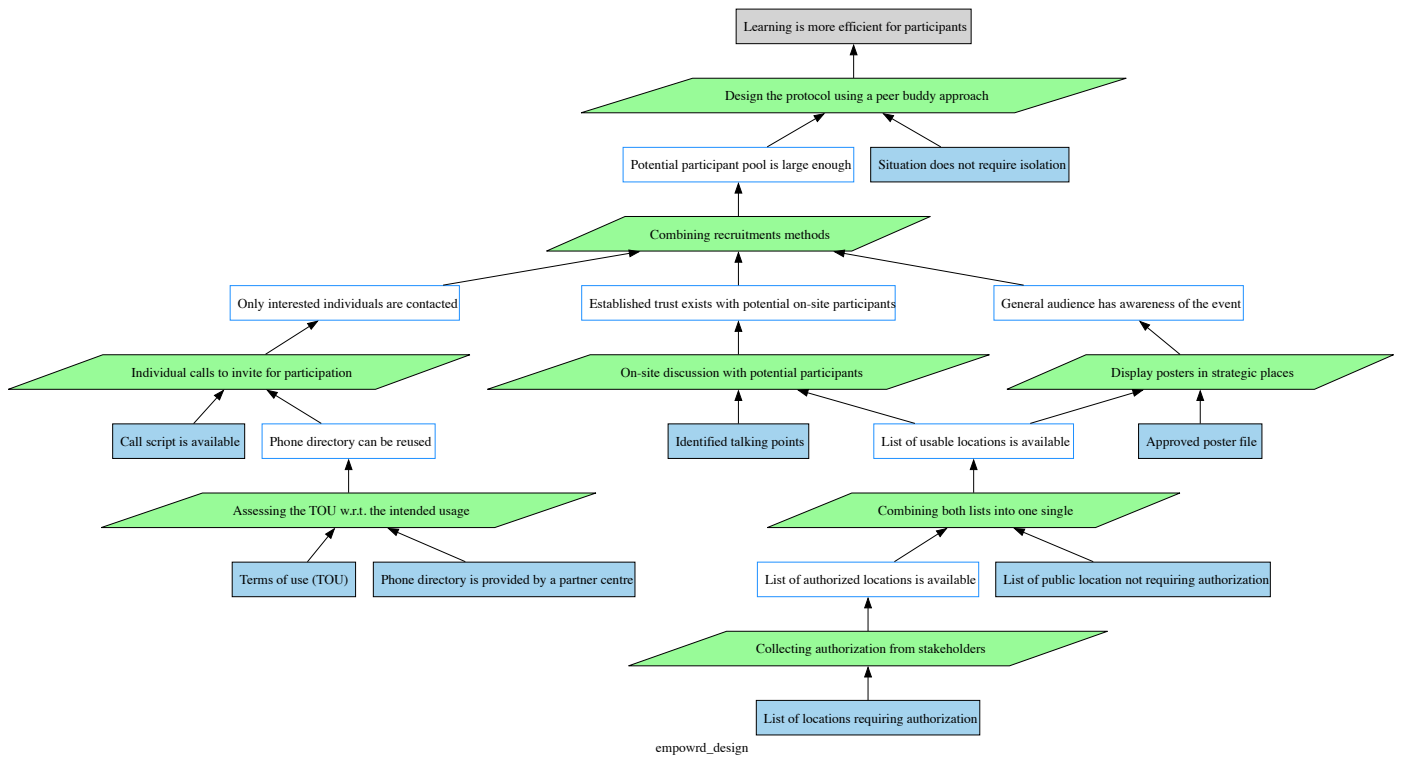
## 7. Discussion

*Transferability & Reuse.* The primary contribution of this paper is the introduction of a systematic framework for capturing and organizing interaction practices, ensuring their internal

<sup>6</sup> For now, this step is manual, but we plan to leverage the FM structure to automate it.



**Figure 9** Catalogue of EMPOWRD practices: experiments and remediations (highlighted choices lead to the justification described in FIG. 10).



**Figure 10** Final justification based on choices in the feature model (LST. 5).

consistency and enabling the validation of rationales used in experimental designs. While this work focuses on the rigorous documentation of existing knowledge, the a priori transferability of these practices to new research dimensions is currently being explored. This proactive application of the framework to guide future experimental setups represents the second phase of the

project, scheduled for 2026–2028, which aims to transform the modelling repository into a predictive decision-support system for interdisciplinary researchers.

*Counter arguments.* In safety-critical systems, assurance cases are leveraged to rigorously audit the safety process, serving as a structured mechanism to identify flaws in reasoning

that could lead to catastrophic outcomes for end-users. While the current scope of this work prioritizes the representation and organization of sociotechnical practices (transforming narrative expert advice into structured modelling artifacts), a critical subsequent dimension is ensuring that the resulting justification is fit for purpose. This necessitates the inclusion of counter-arguments and "rebuttals" to challenge the validity of established practices. By explicitly modelling these challenges, researchers can move beyond a purely descriptive account and engage in a more robust validation process, identifying where a practice's assumptions might fail

*Contextualization.* By nature, a system is safe when operating under defined conditions (e.g., a self-driving car would work perfectly on a closed circuit on a sunny day, but would behave in an unsafe way in a Canadian back road during winter). In this work, we have assumed that the practices were absolute, i.e., fit for any context. This is true because we focused on general ones, but context-dependent justification will eventually appear. For example, the *European Union* (EU) protects its citizens through the GDPR, which is not in place in North America. However, ethical and *Diversity, Equity, Inclusion* (DEI) compliance is more developed in Canada than in the EU, and to a certain extent, this is time-dependent, as the situation in the USA regarding this point shifted at the beginning of 2025 under the Trump administration. Specifically for older adults, context dependence is amplified by age-related medical issues. For example, a practice may be suitable for regular participants but not well-suited for those with early signs of Alzheimer's disease (Dery-Pinna et al. 2021).

*Meta-justification.* By offering all known practices to the experiment designer, the approach assumes that (i) each practice is good independently, and (ii) the experiment designer makes reasonable choices for their specific context. For the latter, such a rationale could also be expressed through justification diagrams, since an argumentation model can support decision-making. It raises, then, the question of how to validate and audit such meta-justification.

*Evolution & Scalability.* Practices evolve over time, as knowledge increases. What was considered a best practice yesterday might not be considered one today, and there is no guarantee that new practices will not be discovered and published tomorrow. As a consequence, maintaining the catalogue over time is essential. We need ways to add or remove practices incrementally, to understand how they interact, and to expose audit mechanisms to ensure the catalogue's content remains up to date with current practices. As we rely on FMs to model the practice portfolio, we draw on an extensive body of literature on software product line evolution to support this work. In terms of scalability, FMs modelled with UVL are known to scale to orders of magnitude higher than those required to capture practices within a given domain (Benavides et al. 2025).

*Generalization.* While a case study focuses on an in-depth exploration of a specific phenomenon and does not aim for statistical generalization *per se*, it provides the foundational insights necessary to test the transferability of a modelling approach across disparate domains. For instance, while our current findings are rooted in the EMPOWRD digital literacy program for

older adults, we have already achieved promising results by applying a similar methodology to training Large Language Models (LLMs). This parallel research involves formalizing FATES (Fairness, Accountability, Transparency, Ethics, and Safety/Security) practices into structured models to ensure responsible AI development (Khuu et al. 2026). By successfully mapping expert-driven AI safety protocols using the same framework, we are paving the way to demonstrate that the core mechanics of capturing and organizing practice rationales can be generalized.

## 8. Conclusion

In this paper, we addressed the problem of supporting better interactions with older adults by capturing and then organizing good practices related to such activity. Using our approach, SMEs can express the rationale for each of their practices as justification diagrams and organize them in a feature model to show how they interact with one another. Then, an end-user can reuse that knowledge by using the feature model to derive concrete justification diagrams. If this paper focused mainly on the results obtained in the EMPOWRD project, the approach is, in theory) not limited to this single project. In future work, we will expand the kind of practice reified by collaborating with Université Côte D'Azur (Nice, France) and *Centre d'Accueil de Jour Alzheimer* (Biot, France) to apply the approach to practices related to the interaction with older adults suffering from Alzheimer's disease. We are also planning to continue the internal validation using the three upcoming use cases addressed by EMPOWRD.

## Acknowledgments

This work is funded by *McMaster Research Institute on Aging* (MIRA) through the EMPOWRD Major Program of Research, as well as the *Natural Science and Engineering Research Council* (NSERC) of Canada, thanks to the Discovery Grant (RGPIN-2020-05791) and Alliance (ALLRP-576653-22) programs.

## References

- Abouei, M., Elgamal, R., Detlor, B., & Head, M. (2024). Complementing Older Adult Digital Literacy Training Research with a Mobile User Experience Lab. In *Proceedings of the twenty-third annual pre-icis workshop on HCI research in MIS*.
- Basili, V. R., Caldiera, G., & Rombach, H. D. (2002). Experience factory. In *Encyclopedia of software engineering*. John Wiley & Sons, Ltd. Retrieved from <https://onlinelibrary.wiley.com/doi/abs/10.1002/0471028959.sof110> doi: <https://doi.org/10.1002/0471028959.sof110>
- Benavides, D., Sundermann, C., Feichtinger, K., Galindo, J. A., Rabiser, R., & Thüm, T. (2025). UVL: Feature modelling with the Universal Variability Language. *Journal of Systems and Software*, 225, 112326. Retrieved from <https://www.sciencedirect.com/science/article/pii/S0164121224003704> doi: <https://doi.org/10.1016/j.jss.2024.112326>

- Brinkkemper, S. (1996). Method engineering: engineering of information systems development methods and tools. *Information and Software Technology*, 38(4), 275-280. Retrieved from <https://www.sciencedirect.com/science/article/pii/0950584995010599> (Method Engineering and Meta-Modelling) doi: [https://doi.org/10.1016/0950-5849\(95\)01059-9](https://doi.org/10.1016/0950-5849(95)01059-9)
- Castro, J., Kolp, M., & Mylopoulos, J. (2002, September). Towards requirements-driven information systems engineering: the tropos project. *Inf. Syst.*, 27(6), 365–389. Retrieved from [https://doi.org/10.1016/S0306-4379\(02\)00012-1](https://doi.org/10.1016/S0306-4379(02)00012-1) doi: 10.1016/S0306-4379(02)00012-1
- Chaudhari, N. (2024). Pipelines have feelings too: A structured way to design ci/cd pipelines. In *Proceedings of the acm/ieee 27th international conference on model driven engineering languages and systems* (p. 184–187). New York, NY, USA: Association for Computing Machinery. Retrieved from <https://doi.org/10.1145/3652620.3676876> doi: 10.1145/3652620.3676876
- Chowdhury, T., Lin, C.-W., Kim, B., Lawford, M., Shiraishi, S., & Wassying, A. (2017). Principles for systematic development of an assurance case template from iso 26262. In *2017 ieee international symposium on software reliability engineering workshops (issrew)* (p. 69-72). doi: 10.1109/ISSREW.2017.14
- Clavreul, M. (2011). *Model and Metamodel Composition: Separation of Mapping and Interpretation for Unifying Existing Model Composition Techniques* (Theses, Université Rennes 1). Retrieved from <https://theses.hal.science/tel-00646893> (final draft)
- Dardenne, A., van Lamsweerde, A., & Fickas, S. (1993). Goal-directed requirements acquisition. In *Selected papers of the sixth international workshop on software specification and design* (p. 3–50). NLD: Elsevier Science Publishers B. V.
- Dery-Pinna, A., Giboin, A., & Renevier-Gonin, P. (2021). Design of digital therapeutic workshops for people with alzheimer’s disease. In M. Antona & C. Stephanidis (Eds.), *Universal access in human-computer interaction. design methods and user experience - 15th international conference, UAHCI 2021, held as part of the 23rd HCI international conference, HCII 2021, virtual event, july 24-29, 2021, proceedings, part I* (Vol. 12768, pp. 369–383). Springer. Retrieved from [https://doi.org/10.1007/978-3-030-78092-0\\_24](https://doi.org/10.1007/978-3-030-78092-0_24) doi: 10.1007/978-3-030-78092-0\_24
- Detlor, B. (2025). Public Library-Led Digital Literacy Training to Older Adults in Social Housing. In *Proceedings of the 2025 conference of the canadian association for information science (cais)*.
- Dickinson, A., Arnott, J., & Prior, S. (2007). Methods for human-computer interaction research with older people. *Behaviour & Information Technology*.
- Duffau, C., Polacsek, T., & Blay-Fornarino, M. (2018). Support of justification elicitation: Two industrial reports. In J. Krogstie & H. A. Reijers (Eds.), *Advanced information systems engineering - 30th international conference, caise 2018, tallinn, estonia, june 11-15, 2018, proceedings* (Vol. 10816, pp. 71–86). Springer. Retrieved from [https://doi.org/10.1007/978-3-319-91563-0\\_5](https://doi.org/10.1007/978-3-319-91563-0_5) doi: 10.1007/978-3-319-91563-0\_5
- Elgamal, R., & Detlor, B. (2024). Addressing the Digital Divide Among Older Adults in Social Housing. In *Proceedings of the thirtieth americas conference on information systems (amcis)*. Salt Lake City, USA.
- Ellis, R. D., & Cochran, D. L. (1999). Practices to encourage participation of older adults in research and development. In *Chi '99 extended abstracts on human factors in computing systems* (p. 39–40). New York, NY, USA: Association for Computing Machinery. Retrieved from <https://doi-org.libaccess.lib.mcmaster.ca/10.1145/632716.632744> doi: 10.1145/632716.632744
- Farage, M. A., Miller, K. W., Ajayi, F., & Hutchins, D. (2012). Design principles to accommodate older adults. *Global Journal of Health Science*.
- Galindo, J. A., Horcas, J.-M., Felferning, A., Fernandez-Amoros, D., & Benavides, D. (2023). Flama: A collaborative effort to build a new framework for the automated analysis of feature models. In *Proceedings of the 27th acm international systems and software product line conference - volume b* (p. 16–19). New York, NY, USA: Association for Computing Machinery. Retrieved from <https://doi.org/10.1145/3579028.3609008> doi: 10.1145/3579028.3609008
- Gleason, M. E., Iida, M., & Bolger, N. (2018). Supporting older adults through peer companionship: Benefits for well-being and social engagement. *Journal of Gerontology: Psychological Sciences*, 73(2), 202–211.
- Gotel, O., & Finkelstein, C. (1994). An analysis of the requirements traceability problem. In *Proceedings of ieee international conference on requirements engineering* (p. 94-101). doi: 10.1109/ICRE.1994.292398
- Gutiérrez-Fernández, A. M., et al. (2024). Variability management and software product line engineering: an empirical study. *Journal of Systems and Software*. (survey highlighting empirical issues in variability support) doi: 10.1016/j.jss.2024.112114
- Henderson-Sellers, B., & Ralyte, J. (2010, 06). Situational method engineering: State-of-the-art review. *J. UCS*, 16, 424-478.
- Jeanneret, C., France, R., & Baudry, B. (2008). A reference process for model composition. In *Proceedings of the 2008 aosd workshop on aspect-oriented modeling* (p. 1–6). New York, NY, USA: Association for Computing Machinery. Retrieved from <https://doi.org/10.1145/1404920.1404921> doi: 10.1145/1404920.1404921
- Jussli, A., & Gewald, H. (2021). Senior dt – a design thinking method to improve requirements engineering for elderly citizens. In *Proc. ieee 29th international requirements engineering conference workshops*.
- Kang, K., Cohen, S., Hess, J., Novak, W., & Peterson, A. (1990, Nov). *Feature-oriented domain analysis (foda) feasibility study* (Tech. Rep. No. CMU/SEI-90-TR-021). Carnegie Mellon University, Software Engineering Institute’s Digital Library. Retrieved from <https://insights.sei.cmu.edu/library/feature-oriented-domain-analysis-foda-feasibility-study/> (Accessed: 2025-Jun-2)
- Khuu, K. T.-P., et al. (2026). Model Cards: Stop Carding, Start

- Modelling! In *Proceedings of the IEEE/ACM 48th International Conference on Software Engineering: New Ideas and Emergent Results (ICSE-NIER)*.
- Kunz, W., & Rittel, H. W. J. (1970). *Issues as elements of information systems* (Tech. Rep. No. 131). Berkeley, CA: Institute of Urban and Regional Development, University of California. (Google Books ID: B-MaQAAMAAJ)
- Lachapelle-Dagenais, A., Mosser, S., Pinna-Déry, A.-M., & Blay-Fornarino, M. (2021). Requirements engineering for the ageing population: a teaching perspective. In T. Yue & M. Mirakhorli (Eds.), *29th IEEE International Requirements Engineering Conference Workshops, RE 2021 Workshops, Notre Dame, IN, USA, September 20-24, 2021* (pp. 248–257). IEEE. Retrieved from <https://doi.org/10.1109/REW53955.2021.00046> doi: 10.1109/REW53955.2021.00046
- Lee, J. (1997, May). Design rationale systems: Understanding the issues. *IEEE Expert: Intelligent Systems and Their Applications*, 12(3), 78–85. Retrieved from <https://doi.org/10.1109/64.592267> doi: 10.1109/64.592267
- MacLean, A., Young, R. M., Bellotti, V. M., & Moran, T. P. (1991). Questions, options, and criteria: Elements of design space analysis. *Human-Computer Interaction*, 6(3-4), 201–250. doi: 10.1207/s15327051hci0603&4\_2
- McSCert. (2023). *jPipe Language*. Retrieved from <https://www.jpipeline.org>
- Mosser, S., Pulgar, C., Blay-Fornarino, M., Patel, D., Loh, A., & Bruel, J.-M. (2023). Yes, Configuring is Good, But Have You Ever Tried Justifying?.
- Object Management Group. (2017). *Semantics of Business Vocabulary and Business Rules (SBVR) (1.5 ed.)* [Computer software manual]. Retrieved from <https://www.omg.org/spec/SBVR/1.5/>
- Polacsek, T. (2016). Validation, accreditation or certification: A new kind of diagram to provide confidence. In *Proc. IEEE International Symposium on Systems Engineering*.
- Ralph, P., Baltes, S., Bianculli, D., Dittrich, Y., Felderer, M., Feldt, R., ... others (2020). Empirical standards for software engineering research. *arXiv preprint arXiv:2010.03525*. Retrieved from <https://arxiv.org/abs/2010.03525>
- Ramesh, B., & Jarke, M. (2001). Toward reference models for requirements traceability. *IEEE Transactions on Software Engineering*, 27(1), 58-93. doi: 10.1109/32.895989
- Rolland, C. (1998). A comprehensive view of process engineering. In B. Pernici & C. Thanos (Eds.), *Advanced information systems engineering* (pp. 1–24). Berlin, Heidelberg: Springer Berlin Heidelberg.
- Sayago, S. (2019). *Perspectives on human-computer interaction research with older people*. Springer.
- SCSC Assurance Case Working Group. (2021). *Goal structuring notation community standard version 3* (GSN Community Standard No. SCSC-141C). CA, USA: Safety-Critical Systems Club.
- Strom, R. D., & Strom, P. S. (2011). *Peer learning for adults*. R&L Education.
- Sundermann, C., Feichtinger, K., Engelhardt, D., Rabiser, R., & Thüm, T. (2021). Yet another textual variability language? a community effort towards a unified language. In *Proceedings of the 25th ACM International Systems and Software Product Line Conference - Volume A* (p. 136–147). New York, NY, USA: Association for Computing Machinery. Retrieved from <https://doi.org/10.1145/3461001.3471145> doi: 10.1145/3461001.3471145
- Tsai, H.-y., Shillair, R., & Cotton, S. R. (2017). Learning to use mobile devices: A comparison of older adults' learning strategies and younger adults' learning processes. *Educational Gerontology*, 43(4), 209–219.
- World Health Organization. (2024). *Ageing and health*.
- Yu, E. S. K. (1997). Towards modeling and reasoning support for early-phase requirements engineering. In *Proceedings of the 3rd IEEE International Symposium on Requirements Engineering* (p. 226). USA: IEEE Computer Society.
- Zaphiris, P., Kurniawan, S., & Ghiawadwala, M. (2007). A systematic approach to the development of research-based web design guidelines for older people. *Universal Access in the Information Society*.

## About the authors

**Sébastien Mosser** is an Associate Professor at McMaster University, serving as Associate Chair of Computing and Software and Associate Director of McSCert. His research focuses on software composition, domain-specific languages, and modeling. You can contact the author at [mossers@mcmaster.ca](mailto:mossers@mcmaster.ca).

**Shyam Ravichandran** is a Software Engineering PhD student at McMaster University. He researches Human-Computer Interaction, specializing in technology design and evaluation for older adults. You can contact the author at [ravics21@mcmaster.ca](mailto:ravics21@mcmaster.ca).

**Kalvin Khuu** is a Software Engineering PhD student at McMaster University. He uses lightweight safety models to study machine learning practices regarding fairness, accountability, ethics, and safety. You can contact the author at [khuuk2@mcmaster.ca](mailto:khuuk2@mcmaster.ca).

**Brian Detlor** is a Professor of Information Systems at McMaster University. He investigates digital literacy training for marginalized older adults in community settings. You can contact the author at [detlorb@mcmaster.ca](mailto:detlorb@mcmaster.ca).

**Denise Y. Geiskkovitch** is an Assistant Professor at McMaster University and the Barber-Gennum Endowed Chair. Her research focuses on human-computer and human-robot interaction to benefit the aging population. You can contact the author at [geiskkod@mcmaster.ca](mailto:geiskkod@mcmaster.ca).

**Anne-Marie Pinna-Déry** is an Associate Professor at Université Côte d'Azur (Polytech'Nice). Her research interests include human-computer interaction, model-driven engineering, and user interface composition. You can contact the author at [Anne-Marie.PINNA@univ-cotedazur.fr](mailto:Anne-Marie.PINNA@univ-cotedazur.fr).

**P.J. White** is a Senior Lecturer and Research Director at SETU (Ireland) and an Adjunct Professor at McMaster University. He specializes in interdisciplinary co-design and research for aging. You can contact the author at [PJ.White@setu.ie](mailto:PJ.White@setu.ie).