A participative modelling tool supporting the story card method

by

Petra Opperman

16020155

Submitted in partial fulfilment of the requirements for the degree

MEng Industrial Engineering

in the

FACULTY OF ENGINEERING, BUILT ENVIRONMENT AND INFORMATION TECHNOLOGY

at the

UNIVERSITY OF PRETORIA

Study leader:

Prof. M de Vries

ABSTRACT

When business-oriented software needs to be developed within a scaled context, the story card method (SCM) assists in structuring emerging software requirements within a taxonomy that represents enterprise operation. However, agile team members first need to develop a common understanding about enterprise operation when they construct the enterprise operation taxonomy.

The COVID-19 pandemic emphasised the need to use *digital participative design practices* when in-person face-to-face participation is not possible, especially when team members are geographically dispersed. A key concern was identified during a previous design iteration of the SCM and confirms that the current software modelling tool that is being used, in combination with the SCM, does not encourage active participative modelling (PM), due to the *latency* of the tool. This study aims to investigate whether a new PM modelling tool is useful to post-graduate participants within a tertiary educational context, when they apply digital PM within the context of their own enterprise using the SCM.

The study starts with a literature review, indicating that problems related to PM also exist within a broader context than this study. A design science research (DSR) approach was followed in this study to evolve the existing SCM artefact and address the concern related to the previous software modelling tool. As multiple PM tools are available, a list of minimum requirements was used to short-list two tools. A comparative analysis of the two tools is provided, motivating the selection of a single tool that was used to support the SCM. In applying the SCM, 36 participants were involved. Of the 36 participants, 25 completed a survey to evaluate the *usefulness* of the tool and whether the tooling *encouraged participative design*. Using a demonstration case to illustrate the notion of participative design to the post-graduate participants, using the selected tool in combination with the SCM, feedback was obtained about the participative modelling tool that was used by post-graduate participants. Finally, a conclusion is provided on the usefulness of the PM tool and whether the findings could be generalised beyond the combined use with the SCM.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

Abbreviation	Description
BOK	Body of Knowledge
BPMN	Business Process Modelling Notation
CSD	Coordination Structure Diagram
DEMO	Design and Engineering Methodology for Organisations
DEMOSL	DEMO Specification Language
DSR	Design Science Research
IS	Information Systems
IT	Information Technology
NEMO	Network of European Museum Organisations
OCD	Organisation Construction Diagram
PI	Publication Identifier
PM	Participative Modelling
RQ	Research Question
SCM	Story Card Method
TPT	Transactor Product Table

1. Introduction

The aim of Chapter 1 is to introduce the reader to the *problem context* (Section 1.1), the *initiation of the study* (Section 1.2), *problem statement* (Section 1.3), *research questions* (Section 1.4), *thesis statement* (Section 1.5), *scope demarcation and limitations* (Section 1.6) and *significance* of the study (Section 1.7). A *problem instance analysis* (Section 1.8) is also provided.

1.1 Problem context

When business-oriented software needs to be developed within a scaled context, the *story card method* (SCM) assists in structuring emerging software requirements within a taxonomy that represents enterprise operation. However, agile team members first need to develop a common understanding about enterprise operation when they construct the enterprise operation taxonomy. The COVID-19 pandemic emphasised the need to use *digital participative design practices* when in-person face-to-face participation is not possible, especially when team members are geographically dispersed.

1.2 Initiation of the study

This study was initiated by a real-world department at a tertiary education institution, as a problem was identified regarding the current participative modelling software tool that is being used in combination with the SCM in one of the postgraduate modules that focus on enterprise design. *Participative modelling* (PM) entails the simultaneous modelling of diagrams by two (or more) modellers, using the same software application modelling tool, each using their own laptop or computer.

The software application tool that is currently being used for PM within the SCM is Diagrams.net, or previously referred to as Draw.io. Diagrams.net is relatively easy to learn and use, and available at no cost. It can be used either online or offline (as a desktop application) and enables the user to create/model, edit and save many different types of diagrams such as flowcharts, entity relationship diagrams, block diagrams and network diagrams.

However, Diagrams.net is not completely sufficient in meeting the needs of the SCM and is thus only seen as a short-term solution. The main inefficiency is the delay experienced by the modellers, when using a PM approach to develop coordination structure diagrams (CSDs) as a means to structure emerging software system requirements. The up-to-five second delay occurs when each user has to refresh their browser every time a change is made to the Diagrams.net model. Thus, the co-modelling process is hindered significantly as users are unable to receive immediate feedback from each other.

The primary measure of interest for this study will be the usefulness of a new PM tool, with *usefulness* relating to the multiple requirements that need to be addressed by the tool.

Concepts from the software application development body of knowledge (BOK), related to this study, are explained in more detail in Section 2.2.3. The concepts also provide the context for the initial development and subsequent adaptation of the SCM.

Since the focus of this study is to present alternative tooling for PM, when using the SCM, Chapter 2 presents the SCM and PM in more detail. In order to provide sufficient context with regards to the SCM for the remainder of this chapter, it is important to note that the SCM evolved from an *initial* SCM to an *adapted* SCM, and then further evolved to an *extended* SCM. For the remainder of this dissertation, the SCM (without a preceding adjective) refers to the latest version of the SCM, i.e., the *extended* SCM.

1.3 Problem statement

The problem is that the software modelling tool, Diagrams.net, that is currently being used to demonstrate PM as part of the *adapted* SCM, is inefficient in encouraging active participation during modelling due to its *latency* in reflecting updates performed by co-modellers.

1.4 Research questions

The sub-sections that follow include the *primary* as well as *secondary* research questions related to this study.

1.4.1 Primary research question

The primary research question for this study is as follows:

How can a participative modelling tool support the story card method?

1.4.2 Secondary research questions

The secondary research questions (RQs) for this study, as well as an indication of the chapters where these questions will be answered, are provided below:

RQ1: How were the two alternative solutions compared to derive at a single solution? [Answered in Chapter 4]

RQ2: How was the chosen solution developed and demonstrated? [Answered in Chapter 5]

RQ3: How was the final solution evaluated and what are the findings? [Answered in Chapter 6]

1.5 Thesis statement

A thesis statement communicates the primary aim of a dissertation to the reader in one concise sentence. The thesis statement (related to the primary research question) for this dissertation is as follows:

> Implementing the story card method steps within MURAL as participative modelling tool, supports the story card method.

1.6 Scope demarcation and limitations

The scope of this study will include an extension of the SCM, specifically focusing on the PM *software tooling* in support of the SCM, when applying the SCM steps prior to system requirements elicitation. A design science research (DSR) approach (see Chapter 3) will be followed in order to develop and evaluate the main artefact of this study, i.e. the *software tool implementation* of the SCM.

A limitation of the study includes the list of minimum requirements specified for the PM tool (provided in more detail in Chapter 4). Another limitation that will be considered is whether the PM is only restricted to the combined use with the SCM, or whether the tool could be generalised beyond the use with the SCM. The related findings are discussed in more detail in Chapter 7.

1.7 Significance

Even though the primary focus is on experimenting with and selecting a PM tool that is suitable for use in combination with the SCM, this study is also significant within a broader context.

Patel et al. (2012) state that it becomes increasingly important for organisations to be able to support collaborative working environments as they move towards more distributed ways of working, whereas Renger et al. (2008) reason that end-users, stakeholders, experts and entrepreneurs are able to create a shared understanding about a system representation through participative/collaborative modelling. They also state that due to the increasing complexity of systems and organisations, it becomes more important to create a shared understanding and joint representations of those systems. The significance of the study is further justified in the synthesised results from the literature review (provided in Section 2.2.6).

1.8 Problem instance analysis

This section focuses on the problem instance that was identified at a department at a tertiary education institution, elaborating on the *data-gathering strategy* for analysing the problem as well as validating the *need for a solution.*

1.8.1 Data-gathering strategy

The primary data-gathering strategy for analysing the problem that existed at a department at a tertiary education institution, consisted of ad hoc interviews between the SCM designer and the researcher of this study. Semi-structured online meetings were used during the initial stages of the study to clarify the minimum requirements for a suitable PM tool, as discussed in more detail in Chapter 4.

1.8.2 Need for a solution

The *adapted* SCM has already been evaluated in a study by De Vries (2022) and currently incorporates a software modelling tool, Diagrams.net. Feedback from respondents indicated that the tool complies with most of the minimum criteria as stipulated in Section 4.1. However, the respondents also indicated that a *delay* exists in refreshing updates to diagrams that are made by the modellers. The SCM is thus in need of a new PM tool that does not demonstrate a delay, and satisfies all of the remaining requirements. In Chapter 2 it is indicated that the problem instance, i.e., the need for an efficient PM tool to facilitate co-modelling within the area of software development, also exists as a bigger class-of-problems.

2. Literature Review

This chapter provides the necessary *background and related work* (Section 2.1) on the research topic as well as a *literature review* (Section 2.2) in order to provide an overview of the existing literature related to the study.

2.1 Background and related work

Enterprises are complex entities, multiple enterprise models and perspectives are needed to represent different facets or domains of the enterprise (Karagiannis et al., 2022). Multiple modelling languages exist to represent a particular domain, such as the organisation design domain, including DEMO (Design and Engineering Methodology for Organisations), BPMN (Business Process Modelling and Notation) and Petri Nets (Karagiannis et al., 2022). The DEMO modelling language is specified using DEMOSL (DEMO Specification Language). DEMO, when compared to other modelling languages, focuses on representing enterprise operation in a consistent and concise way, also hiding operational complexity in a consistent way (Dietz and Mulder, 2020). One of the four aspect models, the cooperation model, represents the essence of enterprise operation in terms of a coordination structure diagram (CSD) and a transactor product table (TPT).

Although the CSD is useful as a taxonomy in structuring user stories that emerge during an agile software development project, an additional method, the SCM, is required to model a CSD in a participative way (De Vries, 2022). However, the latest version of the SCM indicated that the software tool hampered interactive modelling and thus needed further experimentation with alternative tools (De Vries, 2022).

Hence, the history of the *story card method (SCM)* as well as more detail on *participative design and participative enterprise modelling* are provided below. A discussion regarding *related work on tool evaluation* is also provided.

2.1.1 The story card method (SCM)

The SCM was initially developed and introduced in an article by De Vries (2018). The *initial* SCM was developed in order to facilitate logical understanding of the abstract concepts related to an organisation construction diagram (OCD). The OCD is a DEMO construct useful in representing a blue print of enterprise operation when eliciting requirements and developing supporting information systems. The *initial* SCM consisted of 5 inputs and 10 method steps, where *physical collaboration* was incorporated through the use of paper sticky notes.

The application of the *initial* SCM resulted in an OCD, based on DEMOSL version 3, modelled on an A1 sized paper. Positive feedback was obtained regarding the solution artefact i.e. that the (*initial*) SCM was useful in relating to a concrete world in order to understand OCD-related concepts when participants had to apply the (*initial)* SCM. Most participants indicated that they preferred manually modelling with sticky notes, as it is easy to understand, encourages conversation and allows for changes by moving the sticky notes around. However, evaluation results also indicated that participants who applied the *initial* SCM experienced difficulties in applying some of the method steps, leading to the opportunity that the method steps could be further improved.

Therefore, the *initial* SCM was adapted and evaluated again in a second article by De Vries (2022). As DEMOSL evolved from version 3 to version 4, the application of the *adapted* SCM resulted in a different output, namely a CSD, based on DEMOSL version 4. The main purpose of the *adapted* SCM is to provide guidance when agile software development team members first need to obtain a common understanding of enterprise operations, using the CSD as a taxonomy to structure emerging user stories when supporting software application systems are developed. The *adapted* SCM consists of 3 inputs and 12 method steps, incorporating *digital collaboration* due to COVID-19 restrictions as physical meetings incorporating physical sticky notes were not always possible.

The CSD was modelled using a software modelling tool, i.e. Diagrams.net. Feedback from the 2022 study indicated that *Steps 7 to 12* of the *adapted* SCM could be further refined, as participants found it difficult to perform these steps when applying the *adapted* SCM. It was also determined that the latency of Diagrams.net in reflecting updates on the models, performed by co-modellers, is a problem as it hinders interactive co-modelling.

The *adapted* SCM thus presented an opportunity for improvement. Therefore, in parallel with, and for the purposes of this study, the abovementioned *adapted* SCM was further extended by the SCM designer, focusing on refinement of the *adapted* SCM *steps*, and is presented below. The differences between the *adapted* SCM and *extended* SCM are indicated using bold lettering.

The *extended* SCM specifies 3 inputs and 13 method steps.

<u>*Inputs:*</u>

(1) Information Technology (IT) hardware, e.g. laptop/computer and internet connection for co-modellers.

(2) **Participative modelling tool, such as MURAL** that allows collaboration of multiple modellers to co-compose a flow-charting diagram, as well as a coordination structure diagram (CSD) and a transactor product table (TPT).

(3) **Participation-pair**, i.e. <u>facilitator</u> that received appropriate Design and Engineering Methodology for Organisations (DEMO) training, and a <u>colleague</u> that is knowledgeable about some existing operations at a real-world enterprise. **The modelling responsibilities of the facilitator and colleague are also clearly distinguished in the method steps, underlining the responsible role.**

<u>*Method steps:*</u>

- *Step 1:* **<u>Colleague</u>** explains a short process (about 10 to 15 tasks) that s/he is involved with. Ensure that the process incorporates the use of IT (e.g. the process followed from requesting vacation leave up to receiving notification about the approval of the request). Explain to your colleague that s/he needs to formulate the tasks (*verb* + *noun*) using rectangular shapes for tasks, mapping out the *tasks* in sequence of occurrence, left to right. A decision-making *gateway* may be used to represent different paths, based on the gateway's decision-outcomes. Use *swim lanes* to represent task-responsibilities associated with *existing actor roles* (these are often composite actor roles) at the enterprise. A standard flow-charting language may be used.
- *Step 2:* **<u>Facilitator</u>** explains the red-green-blue *production triangle* (i.e. a means to classify production tasks).
- *Step 3*: **<u>Facilitator</u>** explains the *complete transaction pattern* for actor-collaboration regarding production acts. Identify an *original* production act from the process flow chart to explain the collaborative interaction around the *original* production act.
- *Step 4:* **<u>Both</u> take turns** in classifying some process tasks as *original*. Duplicate identified production tasks, using red-color-coding. Duplicate identified original coordination tasks associated with an *original* production tasks, using pink (light-red-color-coding) for the duplicated task.
- *Step 5:* **<u>Both</u> take turns** in formulating alternative descriptions for color-coded tasks. (a) Edit each task that was classified as an *original production task*, adding an alternative

description, i.e. "execute + <transaction kind>". **Complex tasks (sub-processes) may translate into multiple transaction kinds. (b) Edit each task that was classified as an *original coordination task*, adding an alternative description that reflects the applicable coordination act from the *complete transaction pattern*, e.g. "request + <transaction kind>".**

- *Step 6:* **Both take turns** in analysing the remaining tasks, reaching consensus on tasks that are informational acts, i.e. sharing/ remembering/ calculating acts. **Duplicate these tasks and apply green color-coding to those tasks.** *Not all tasks need to be color-coded*, since some may imply documental acts and these will not be shown on the CSD.
- *Step 7:* **Facilitator** duplicates the red and light-red tasks, moving the duplicated tasks to the bottom of the drawing space. For each new <transaction kind> (evident in an original production task or original coordination task), create an internal elementary transactor role (OR environmental elementary transactor role if the executing actor role is outside the scope-of-interest **OR external transaction kind with elementary executor role). It is also possible that a transactor role is self-activating when time alone initiates the transaction kind.**
- *Step 8:* **Facilitator explains** *parent-part-structures*, i.e. how one transaction kind becomes *a part of* one or more *parent* transaction kinds via initiation. Differentiate between a response-related deep-process-structure and flat-process-structure. Also, indicate *cardinality* when a *parent* instance initiates *part* instances if the cardinality deviates from the default of 1..1. **Where a composite transactor role initiates another transactor role, a cardinality of non-definable (nd) should be used.**
- *Step 9:* **Both take turns** in adding initiator links where every transactor role should be initiated at least by one other transactor role on the diagram, **with the following exceptions: (1) The self-activating transactor role cannot be initiated by another transactor role, but self-activating transactor roles may initiate other transactor roles; (2) It is also possible that environmental composite transactor roles or additional elementary transactor roles (not supported by the flow-chart) need to be added as initiators.**
- *Step 10:* **Colleague** duplicates green tasks that are *informational* production tasks, moving the duplicated tasks to the bottom of the drawing space. **Facilitator** uses the *informational* production tasks and inputs from the colleague to complete the *interstriction structure* adding *access links* when transactor roles need to access production facts from the banks of

original transaction kinds, *multiple original transaction kinds* and *external multiple original transaction kinds*.

- *Step 11:* **Facilitator** obtains inputs from colleague to add *wait links*, indicating that progress of an instance of one transaction kind may impede the progress of another transaction kind's instance(s).
- *Step 12*: **Both** validate the CSD, enquiring whether the "leaves" of the upside-down tree-structure are truly elementary. If some of the leaves can be further decomposed, i.e. they are composite (internal or environmental) transactor roles, expand the CSD by adding appropriate elementary (internal or external) transactor roles.
- ***Step 13:*** **Both validate the CSD by compiling a TPT, using the empty sticky notes of the template. Facilitator adds first row of TPT, followed by a validation from the colleague. Colleague adds next row of the TPT, followed by a validation from the facilitator. Repeat for subsequent rows. Change the CSD if an error is detected.**

The outputs of the *extended* SCM include a CSD, based on DEMOSL version 4, as well as a TPT. Also note that the "sticky notes" in *Step 13*, refers to *digital* "sticky notes" and not *physical* paper-based sticky notes as in the *initial* SCM.

The main differences between the *adapted* SCM and *extended* SCM can thus be summarised as follows:

1. Diagrams.net was used for the *adapted* SCM, whereas MURAL was used for the *extended* SCM.
2. The *adapted* SCM did not emphasise the responsibilities of co-modellers, whereas the *extended* SCM clearly indicates the responsibilities of co-modellers by underlining the responsible role.
3. The *adapted* SCM did not include the fragments indicated using bold lettering.
4. *Step 13* did not exist in the *adapted* SCM.

From Chapter 4 onwards, it is determined whether MURAL as PM software tool can be used to support the *extended* SCM, also addressing the shortcoming of the *adapted* SCM.

2.1.2 Participative design and participative enterprise modelling

Two knowledge areas developed in parallel, both sharing a *participative* approach, namely *participative design* and *participative enterprise modelling*.

Participative design emerged as a method within human-computer interaction and software design for more than a decade (Scariot et al., 2012). The main objective of using participatory design is to make the consumers as end-users part of the design process, rather than involving the consumers right at the end of the design (Hansen et al., 2021). According to Simonsen and Robertson (2013), participatory design supports mutual learning between multiple participants in collective "reflection-in-action". Since participative design is used when designing a new artefact, an entire design cycle may be implied, starting with an existing understanding of a current artefact or process that needs to be re-designed, also including participation in selecting among different choices for solution areas and solution constructs (Bratteteig and Wagner, 2016).

Enterprise modelling is "an integrated and multi-perspective way of capturing and analyzing enterprise solutions" (Stirna and Persson, 2018). Enterprise models may be created to serve different objectives, also as part of a design cycle to (re-)design a part of the enterprise. When multiple individuals are involved during modelling, enterprise modelling can be further classified as *collaborative* or *participatory*, most effectively conducted when a facilitator leads the collaborative modelling session (Stirna and Persson, 2018). Fellmann et al. (2020) indicate that collaborative modelling emphasises joining of several experts into a coordinated effort, whereas participative enterprise modelling also involve users or enterprise stakeholders. One of the main objectives of a participative approach for enterprise modelling is avoiding conceptual misalignment between the stakeholders and their different perspectives (Fellmann et al., 2020). Participative enterprise modelling is also aligned with the paradigm of the 2018 *Business Information Systems Engineering* conference research note, moving enterprise modelling from an expert discipline towards a more inclusive modelling approach (Sandkuhl et al., 2018).

According to Stirna and Persson (2018), a participative modelling (PM) approach has three characteristics, namely (1) a defined way of working in the form of methodological steps to carry out the modelling sessions with explicit principles of stakeholder involvement, (2) a group of stakeholders responsible for the knowledge that goes into the model and (3) a modelling facilitator responsible for guiding the discussion among stakeholders and the modelling method used.

2.1.3 Related work on tool evaluation

Other research scholars within the knowledge area of participative enterprise modelling have already experimented with online PM tools. Gutschmidt (2021) experimented with two freely-available tools, namely Draw.io and Google Drawings. As already mentioned, the previous version of the SCM (the *adapted* version) also suggested Diagrams.net (a new branding for Draw.io), highlighting the latency problems with Diagrams.net.

Gutschmidt (2021) used the technology acceptance model as a basis to measure the perception of PM tooling in terms of five *perception* criteria: (1) perceived usefulness, (2) perceived ease of use, (3) perceived enjoyment, (4) acceptance and (5) awareness (of changes made by another participant). The study invited four teams of three students to create two different kinds of models, a goal model and process model, where each team had to use a *single* tool, either Google Drawings or Draw.io. Furthermore, the teams used Zoom for communication. Due to the small sample size, factor analysis and significance tests were excluded. Overall, Draw.io scored better than Google Drawings for both goal modelling and process modelling, in terms of most of the five perception criteria. The interviews were analysed to extract positive and negative aspects of the modelling tool. Even though Draw.io was selected as the superior tool, some negative aspects were highlighted.

Since there exists no other study that focus on the investigation and experimentation of the two PM tools included for this study, namely MURAL and Miro, especially in support of the recently developed *extended* SCM steps, this study is novel. Thus, the *gap in the research* that this study aims to address is that the *extended* SCM has not been supported with a suitable PM tool and will therefore investigate how the *extended* SCM can be supported with a PM software application tool.

2.2 Literature review

The purpose of the literature review is to extracted relevant concepts from an existing BOK, related to the research topic. It is also determined whether the problem instance exists as a class-of-problems, by searching for existing solutions or solution areas for the identified class-of-problems.

The *research method for the literature review* is provided in Section 2.2.1, the *protocol* is applied in Section 2.2.2, the *main concepts for the selected body of knowledge* are provided in Section 2.2.3, and the *literature review results* for the *class-of-problems* and *suggested solution areas* are provided in Sections 2.2.4 and 2.2.5 respectively. Section 2.2.6 provides the synthesised results based on the findings of the literature review.

2.2.1 Research method for the literature review

Okoli (2015) provides an 8-step guide to conduct a *systematic* literature review. These steps were only used as a guideline to present the literature review for this study in a *structured* manner. *Step 1* includes the definition of the purpose of the literature review that was already introduced above. *Step 2* requires defining a protocol that can be used in performing the review.

This section provides the protocol that was used to search for existing literature, and details the criteria and techniques that were used in executing the remaining steps, *Steps 3-7*, of Okoli (2015). *Step 8* involves the writing of a review, and does not form part of the protocol.

Research Protocol:

The execution methods of *Steps 3-7* of Okoli (2015) are provided below, and include guidelines for *search execution, practical screening, quality appraisal* as well as *data extraction* and *synthesis*.

Step 3: Search execution

The search execution includes guidelines for *phases, keywords, sources* and *snowballing*.

- *Phases*

  The phases (or steps) that should be followed as a guideline in order to conduct the literature review in a structured manner, is *Steps 3-7* from the 8-step search procedure from Okoli (2015), as mentioned above.

- *Keywords*

  The keywords and search strings to be used in the search execution are based on the following guiding questions:

Q1: What concepts are currently used in literature associated with PM for structuring emerging requirements during software application development? **[Concepts in use]**

Q2: What evidence exists that other enterprises also experience problems with PM in structuring emerging requirements, when they follow an agile software application development approach? **[Class-of-problems]**

Q3: What evidence exists that other researchers already experimented with PM tools that are useful for structuring emerging requirements during agile software application development? **[Existing solutions or solution areas]**

For Q1, no search string was formulated, as the concepts can be derived from the search results of Q2 and Q3.

For Q2 and Q3, the following search string was formulated:

kw:(((“participat* model*” OR “participat* diagram*”) OR (“interactive model*” OR “interactive diagram*”) OR (“collaborat* model*” OR “collaborat* diagram*”)) OR **ti**:((“ participat* model*” OR “participat* diagram*”) OR (“interactive model*” OR “interactive diagram*”) OR (“collaborat* model*” OR “collaborat* diagram*”)) OR **ab**:((“ participat* model*” OR “participat* diagram*”) OR (“interactive model*” OR “interactive diagram*”) OR (“collaborat* model*” OR “collaborat* diagram*”)))

AND **kw**:((“software application*” OR “software tool*”) OR **ti**:(“software application*” OR “software tool*”) OR **ab**:(“software application*” OR “software tool*”))

- *Sources*

  The database sources (digital libraries) that should be used to perform the search process are as follows:

  - ✓ ACM Digital Library
  - ✓ Google Scholar
  - ✓ ScienceDirect
  - ✓ SpringerLink
  - ✓ WorldCat
  - ✓ WorldCat.org

  These sources were selected based on the list of suggested digital libraries related to the field of software engineering, provided by Kitchenham and Charters (2007).

  The Google search engine should also be included as a source to ensure that the search is not too limited, meaning that grey literature such as websites and blogs could be perused. This will have the effect that the literature review is not a pure literature review, but rather a grey literature review.

- *Snowballing*

  According to Wohlin et al. (2014), snowballing is particularly useful in extending a systematic literature review search. Applying this guideline to the literature review for this study, backward snowballing should be applied (to Q2 and Q3’s search) to identify additional sources to include in this study. Backward snowballing refers to the method of searching in the reference lists of selected studies (Wohlin et al., 2014).

Step 4: Practical screening

Practical screening refers to inclusion and exclusion criteria to filter irrelevant literature out of the search process. Table 1 below details the criteria that should be used.

Table 1: Inclusion and Exclusion Criteria for Studies Selection

Inclusion criteria	Exclusion criteria
a) Literature published between 2012 and 2021 (not applied to the articles obtained through snowballing). b) Literature that could possibly have a bearing on the research topic/questions.	a) Literature written in a language other than English. b) Duplicates found in the digital libraries.

Step 5: Quality appraisal

It is necessary to assess the quality of the literature, as all studies are not of equal quality.

For Q2 and Q3, the following checklist provided in Table 2, filtered and adapted from Kitchenham and Charters (2007), should be used to appraise the qualitative studies.

Table 2: Checklist for Qualitative Studies

Number	Question
1	Are the findings credible?
2	Has knowledge or understanding been extended by the research?
3	Are the links between data, interpretation and conclusions clear?
4	Is the reporting clear and coherent?

Each question has three possible answers, “Yes”, “Partly”, or “No”, with a score of 1, 0.5 or 0, respectively. The quality score of a particular study is thus calculated by finding the sum of all the scores of the answers to the abovementioned questions.

As there exists a huge variety of grey literature source types, the grey literature considered for inclusion in this study must be suitably appraised taking into account the specific *nature* of the source, the *origin* of the source as well as the *author* of the source (where applicable).

Step 6: Data extraction

The quality-appraised studies should be displayed in a tabular format, indicating the following:

- Unique publication identifier number
- Author(s') details / Publisher (in the case of grey literature)
- Year
- Title
- Nature of the source

Data must be extracted from the quality-appraised studies, using a codebook (detailed in Sections 2.2.4 and 2.2.5 for the *class-of-problems* and *solution areas* respectively) as well as the software tool, ATLAS.ti, to aid with the coding and analysing process.

Step 7: Data synthesis

Extracted data must be synthesised in the form of a discussion, clearly distinguishing between the results for class-of-problems and solution areas.

2.2.2 Protocol applied

The application of the abovementioned protocol is provided in this section.

Protocol applied for Q2 and Q3:

The applied protocol for Q2 and Q3 distinguishes between the search for white and grey literature. Garousi et al. (2019) define white literature as formal (published) sources such as books, published journal papers and conference proceedings. In contrast, grey literature is defined as unpublished studies, with limited distribution, that are not included in bibliographic retrieval systems e.g. websites, blogs, Wikis and lectures (Garousi et al., 2019).

<u>Search for white literature:</u>

- *Step 3: Search execution*

  After using the search string as formulated in the protocol for Q2 and Q3 to search within the ACM Digital Library, ScienceDirect, SpringerLink, WorldCat and WorldCat.org, a total of 1081 results were displayed.

  The search string as specified in the protocol for Q2 and Q3 could however not be accommodated 'as-is' by the Google Scholar's search engine, thus the search string had to be adapted. It was discovered that the simple keyword, *"Participative Enterprise*

Modelling", displayed the most relevant results in Google Scholar. A total of 77 results were displayed.

- *Step 4: Practical screening*

  Application of inclusion criterion A, reduced the abovementioned 1081 studies to 401 results. Application of exclusion criterion A further reduced the studies to 196 results. Manual application of inclusion criterion B and exclusion criterion B resulted in only 6 studies. Backward snowballing added an additional 3 relevant studies to the results.

  As the results from the Google Scholar search engine were few, and the advanced search engine does not allow for the easy application of the criteria as specified in Table 1, the 77 results were manually screened by reading through the titles and scanning the abstracts. A total of 6 studies were identified that could be useful for the purposes of this report. Backward snowballing resulted in only 1 additional relevant study.

  A total of 16 studies were included in the next step, *Quality appraisal*, discussed next.

- *Step 5: Quality appraisal*

  The quality scores, based on the questions as specified in Table 2, are indicated in Table 3 below for the 16 identified studies. The studies are listed in order of most recent.

Table 3: Quality Scores of Selected Studies

No.	**Study**	**Question**				**Total**
		1	**2**	**3**	**4**	
1	"Participatory modelling from a stakeholder perspective: on the influence of collaboration and revisions on psychological ownership and perceived model quality" (Gutschmidt et al., 2021)	1	1	1	1	4
2	"Identifying HCI patterns for the support of participatory enterprise modelling on multi-touch tables" (Gutschmidt et al., 2019)	1	1	1	1	4
3	"Tools and methods in participatory modelling: Selecting the right tool for the job" (Voinov et al., 2018)	1	1	0.5	1	3.5
4	"Collaborative model-driven software engineering: a classification framework and a research map" (Franzago et al., 2017)	1	1	1	1	4

No.	Study	Question				Total
		1	2	3	4	
5	"The impact of the representatives of three types of process modelling tools on modeller's perceptions and performance" (Polančič and Jošt, 2016)	1	1	1	1	4
6	"Knowledge transfer of software tool development for functional requirements analysis" (Cheng et al., 2016)	1	0.5	1	1	3.5
7	"Modelling collaboration protocols for collaborative modelling tools: Experience and applications" (Gallardo et al., 2013)	1	1	1	1	4
8	"Enabling systems modelling language authoring in a collaborative web-based decision support tool" (Browne et al., 2013)	1	1	1	1	4
9	"Simplifying the development of cross-platform web user interfaces by collaborative model-based design" (Genaro Motti et al., 2013)	1	1	1	1	4
10	"A model-driven development method for collaborative modelling tools" (Gallardo et al., 2012)	1	1	1	1	4
11	"Factors of collaborative working: A framework for a collaboration model" (Patel et al., 2012)	1	1	1	1	4
12	"Online collaboration: Collaborative behaviour patterns and factors affecting globally distributed team performance" (Serçe et al., 2011)	1	0.5	1	1	3.5
13	"Challenges in collaborative modelling: a literature review and research agenda" (Renger et al., 2008)	1	1	1	1	4
14	"Integrating agile modelling with participative enterprise modelling" (Stirna and Kirikova, 2008)	1	1	1	1	4
15	"Participative enterprise modelling: experiences and recommendations" (Stirna et al., 2007)	0.5	1	1	1	3.5
16	"Investigating the influence of situational factors on participative enterprise modelling: making a case for a qualitative research approach" (Persson, 2000)	1	1	1	1	4

As none of the 16 above-listed studies received a total quality score of less than 3.5, or received a zero score for either one of the four questions, it was concluded that none of the studies need to be excluded due to poor quality.

<u>Search for grey literature:</u>

- *Step 3: Search execution*

  The general Google search engine was used to search for applicable grey literature that could be included for this study. As the search procedure for grey literature is of a more informal nature than the search for white literature, a variety of different search terms were used. Snowballing could not be applied, as the identified sources did not have any reference lists that could be perused.

- *Step 4: Practical screening*

  Practical screening was manually and individually applied to *considered* sources only, as the Google search engine does not allow for an advanced search option to filter out all irrelevant results.

- *Step 5: Quality appraisal*

  The websites included as grey literature were deemed as being of sufficient quality as they are the official websites of commercially available software application tools.
  The presentation slides were also appraised as being of sufficient quality, as it is from the NEMO (Network of European Museum Organisations) conference, with J. Stirna (a professor at Stockholm University) listed as an author.

- *Step 6: Data extraction*

  The short-listed studies in Table 4 presents the studies that will be used to relate the research topic to a bigger *class-of-problems*. A total of 11 studies are included, sorted in order of publication dates, starting with the most recent. The first column provides a unique publication identifier (PI) number for each study.

Table 4: Findings for Class-of-Problems

Publication Identifier	**Author(s) / Publisher**	**Year**	**Title**	**Nature of Source**
PI-1	A. Gutschmidt, B. Lantow, B. Hellmanzik, B. Ramforth, M. Wiese and E. Martins	2021	"Participatory modelling from a stakeholder perspective: On the influence of collaboration and revisions on psychological ownership and perceived model quality"	Conference Proceeding
PI-2	A. Voinov, K. Jenni, S. Gray, N. Kolagani, P. D. Glynn, P. Bommel, et al.	2018	"Tools and methods in participatory modelling: Selecting the right tool for the job"	Journal Article
PI-3	M. Franzago, D. Di Ruscio, I. Malavolta and H. Muccini	2017	"Collaborative model-driven software engineering: A classification framework and a research map"	Journal Article
PI-4	J. Gallardo, C. Bravo, M. A. Redondo and J. de Lara	2013	"Modelling collaboration protocols for collaborative modelling tools: Experience and applications"	Journal Article
PI-5	J. Gallardo, C. Bravo and M. A. Redondo	2012	"A model-driven development method for collaborative modelling tools"	Journal Article

Publication Identifier	Author(s) / Publisher	Year	Title	Nature of Source
PI-6	H. Patel, M. Pettitt and J. R. Wilson	2012	"Factors of collaborative working: A framework for a collaboration model"	Journal Article
PI-7	F. C. Serçe, K. Swigger, F. N. Alpaslan, R. Brazile, G. Dafoulas, V. Lopez, et al.	2011	"Online collaboration: Collaborative behaviour patterns and factors affecting globally distributed team performance"	Journal Article
PI-8	M. Renger, G. L. Kolfschoten and G. J. De Vreede	2008	"Challenges in collaborative modelling: a literature review and research agenda"	Journal Article
PI-9	J. Stirna and M. Kirikova	2008	"Integrating agile modelling with participative enterprise modelling"	Conference Proceeding
PI-10	J. Stirna, A. Persson and K. Sandkuhl	2007	"Participative enterprise modelling: Experiences and recommendations"	Conference Proceeding
PI-11	A. Persson	2000	"Investigating the influence of situational factors on participative enterprise modelling: making a case for a qualitative research approach"	Journal Article

The short-listed studies in Table 5 presents the studies that will be used to relate the research topic to possible *solution areas*. A total of 8 studies are included, sorted in order of publication dates, starting with the most recent. The first 3 studies are the identified grey literature sources. Again, each study is provided with a unique PI number.

Table 5: Findings for Solution Areas

Publication Identifier	**Author(s) / Publisher**	**Year**	**Title**	**Nature of Source**
PI-12	Miro.com	2021	"How Agile facilitators use Miro"	Website
PI-13	Mural.co	2021	"Visual collaboration made easy"	Website
PI-14	J. Stirna and B. Lantow	2021	"Participatory enterprise modelling with the 4EM method"	Presentation Slides
PI-15	A. Gutschmidt, V. Sauer, K. Sandkuhl and A. Kashevnik	2019	"Identifying HCI patterns for the support of participatory enterprise modelling on multi-touch tables"	Journal Article
PI-16	G. Polančič and G. Jošt	2016	"The impact of the representatives of three types of process modelling tools on modeller's perceptions and performance"	Journal Article
PI-17	P. H. Cheng, J. M. Fu and L. W. Chen	2016	"Knowledge Transfer of Software Tool Development for Functional Requirements Analysis"	Journal Article

Publication Identifier	Author(s) / Publisher	Year	Title	Nature of Source
PI-18	D. Browne, R. Kempf, A. Hansen, M. O'Neal and W. Yates	2013	"Enabling Systems Modelling Language Authoring In A Collaborative Web-Based Decision Support Tool"	Journal Article
PI-19	V. Genaro Motti, D. Raggett, S. Van Cauwelaert and J. Vanderdonckt	2013	"Simplifying the development of cross-platform web user interfaces by collaborative model-based design"	Conference Proceeding

Refer to Sections 2.2.4 and 2.2.5 for the extracted data provided in a tabular format for the *class-of-problems* and *solution areas* respectively. Also refer to Appendix A and B for evidence of data extraction using ATLAS.ti.

- *Step 7: Data synthesis*
  Refer to Section 2.2.6 for the synthesised results of the class-of-problems and solution areas, respectively.

A graphical presentation of the protocol applied to Q2 and Q3 is illustrated in Figure 1 below:

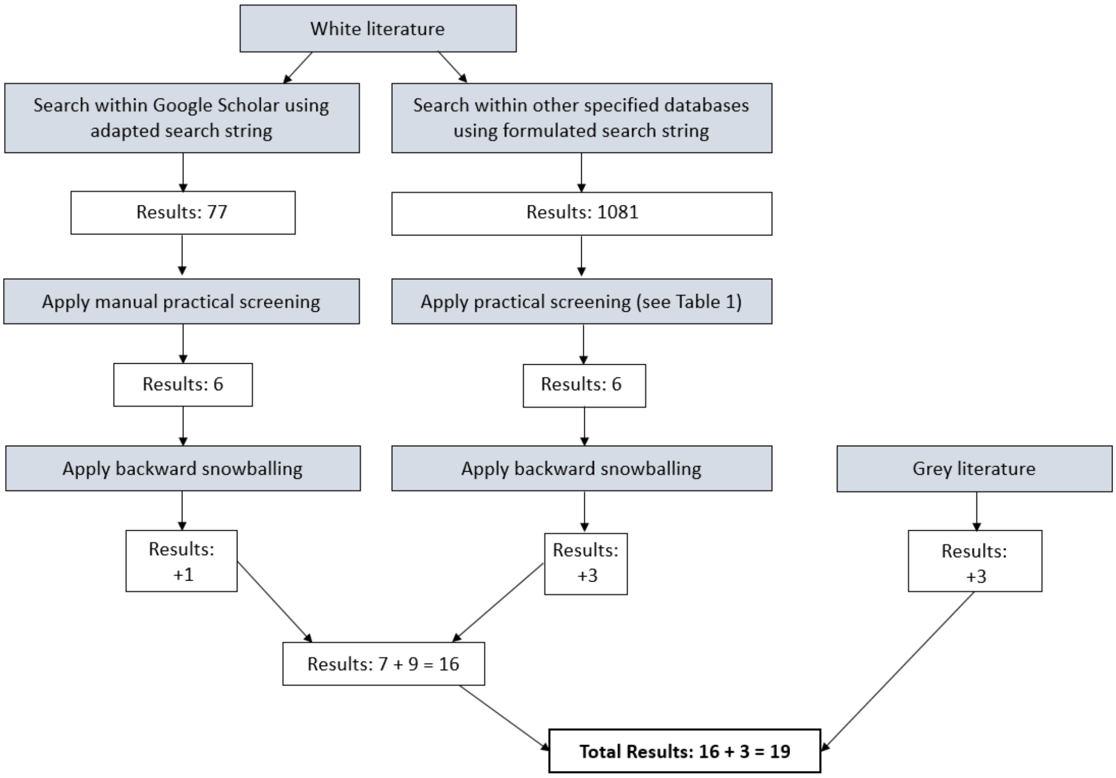


Figure 1: Protocol Applied for Q2 and Q3

As this study was initiated in June 2021, the current literature review was executed between June and November 2021, and does not include any studies published in 2022. To ensure that the literature review is relevant and up to date, the protocol was executed again in July 2022, adapting inclusion criterion A so that it includes the year 2022.

For Q2 and Q3, an additional 13 studies were displayed after extending the year range. None of the additional studies were relevant.

As databases are not static and new studies could be added continuously, the literature review as updated in July 2022 will be the final review to be used for this study. Thus, studies applying to the protocol that may possibly be added to databases during the finalisation and submission period of this dissertation, will not be incorporated.

2.2.3 Main concepts for the selected body of knowledge (BOK)

Main concepts that are used within the software application development BOK, and used in this dissertation are summarised alphabetically in Table 6 below:

Table 6: Main Concepts and Definitions for the Selected BOK

Concept	Definition
Agile software development	Software development agility is defined as the capability of a software team to efficiently and effectively respond to and incorporate user requirement changes during the life cycle of a project. The software development process is viewed as dynamic and evolving, rather than static and predefined (Lee and Xia, 2010).
Aspect models	Dietz and Mulder (2020) present four ontological aspect models that are useful in representing the essence of enterprise operation, namely the cooperation model, action model, process model and fact model.
BPMN (Business Process Modelling Notation)	The goal of BMPN is to provide a modelling notation that is readily understandable by all business users. BPMN is used to produce a business process diagram, which can be defined as a flowcharting technique for creating graphical models of business process operations (White, 2004).
Coordination Structure Diagram (CSD)	The CSD forms part of the cooperation model, and is a graphical representation of actor roles (implemented by humans) that perform certain coordination acts (such as requesting or promising) that relate to certain production acts (Dietz and Mulder, 2020).
DEMO	The design and engineering methodology for organisations (DEMO) is a methodology used to provide the essential (ontological) model of an enterprise in a systematic way, and is an integration of the four aspect models as defined above (Dietz and Mulder, 2020).
DEMOSL	DEMOSL (DEMO Specification Language) is used to express all four aspect models and consists of diagrams, tables and formal textual descriptions (Dietz and Mulder, 2020).
Enterprise ontology	Enterprise ontology refers to the essential knowledge of the construction and the operation of the organisation of an enterprise, completely independent of implementation and realisation (Dietz and Mulder, 2020).

Concept	Definition
Requirements elicitation	Requirements elicitation refers to an in-depth and comprehensive process of discovering information (in the form of requirements) from all relevant stakeholders in relation to a software application that needs to be developed (Ramdhani et al., 2018).
Story card method (SCM)	The aim of the (adapted) SCM is to provide guidance in compiling a CSD in a collaborative way, providing a common understanding of enterprise operations, structuring emerging requirements for agile software application development (De Vries, 2022).
TPT (Transactor Product Table)	According to Dietz and Mulder (2020), a TPT is the list of identified transaction kinds, product kinds and executor roles used to supplement the CSD (as discussed above).
User stories	A user story is a short, semi-structured sentence that illustrates requirements from the perspective of a user, and has been widely accepted as artefacts to capture the user requirements in agile software development (Raharjana et al., 2021).

The timeline presented in Figure 2 shows the development of the (relevant) abovementioned concepts over the years.

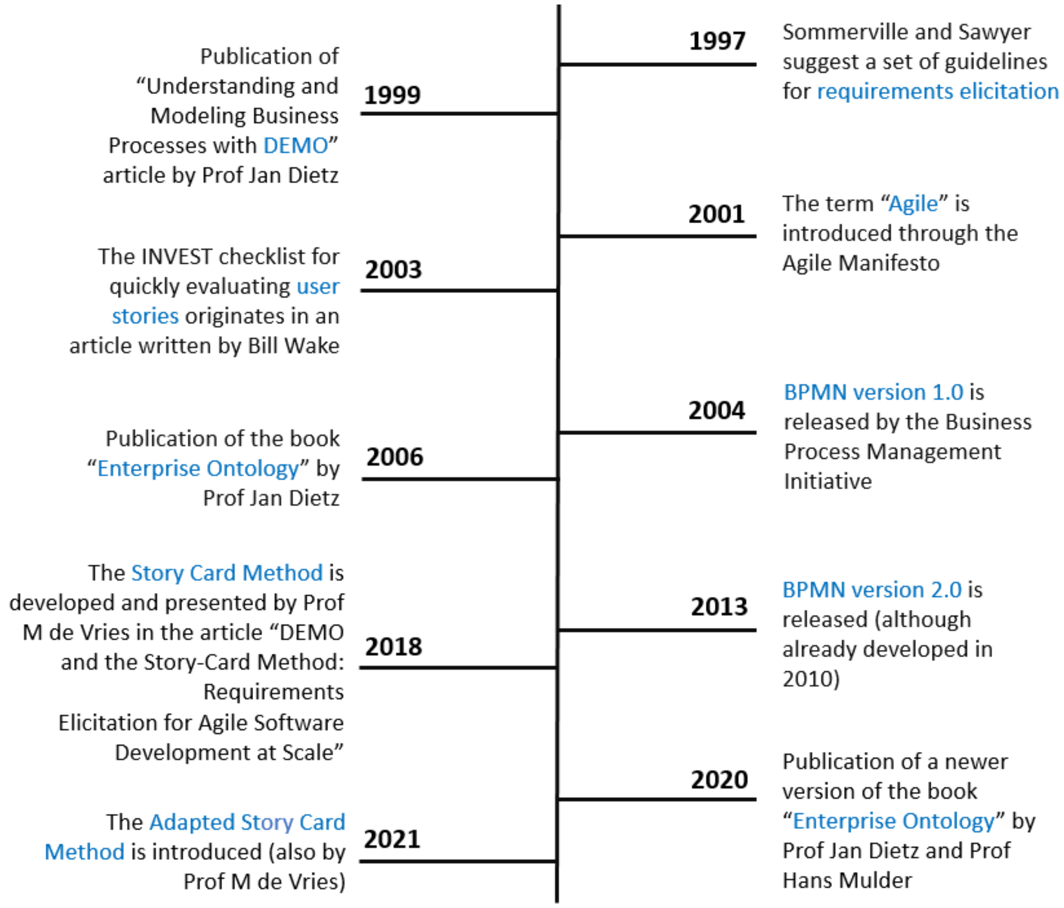


Figure 2: Timeline of Concepts Development

2.2.4 Results for class-of-problems

This section provides a *codebook* that can be used to analyse the text from literature studies related to a *class-of-problems* as well as a table indicating the *extracted themes* from the short-listed studies as presented in Table 4.

Codebook

The following codebook can be used to analyse the text from the identified literature studies. Coding will result in a set of extracted solution *themes* and associated *problems*. An inter-coder agreement is provided in Appendix C, to indicate that the codebook enables coding with an 86% overall agreement when applied by two independent coders.

- *Code label*: **Solution Theme (T)**

  Short definition: A central solution topic that stands out from the text and is frequently used within the software application development BOK.

  Full definition: Solution themes can be seen as the methodologies, practices or techniques that make up the BOK. Without a thorough understanding of solution themes, the BOK cannot be fully understood.

  Example: participative modelling **(T1)**

  When to use: When exploring literature within the software application development BOK.

- *Code label:* **Problem (P)**

  Short definition: A cause for inefficiency or low productivity within an identified solution theme.

  Full definition: A problem relates to a cause for inefficiency or low productivity within an identified solution theme, and can more easily be identified once all themes have first been identified and understood.

  Example: cultural differences among collaborative team members **(P1)**

  When to use: When a fragment relates to factors, criteria, principles or prerequisites that lead to the inefficiency or low productivity of a solution theme that was identified within the software application development BOK, such as the solution theme "participative modelling".

Extracted themes and associated problems

The extracted solution themes with associated problems, using the codebook presented above, for each article/study related to the *class-of-problems*, are provided in Table 7 below:

Table 7: Extracted Themes for Studies Related to the Class-of-Problems

Publication Identifier	Title	Extracted Solution Themes (T) and Associated Problems (P)
PI-1	"Participatory modelling from a stakeholder perspective: On the influence of collaboration and revisions on psychological ownership and perceived model quality"	"Participatory and collaborative enterprise modelling" **(T1)** "Psychological ownership of models" **(P1)**
PI-2	"Tools and methods in participatory modelling: Selecting the right tool for the job"	"Participatory modelling" **(T1)** "Timing and stages of participation" **(P2)** "Number, diversity, background, and skills of the stakeholders" **(P3)**

Publication Identifier	Title	Extracted Solution Themes (T) and Associated Problems (P)
		"Level and intensity of stakeholder participation" **(P4)**
PI-3	"Collaborative model-driven software engineering: A classification framework and a research map"	"Collaborative software engineering" **(T2)** "Model-driven software engineering" **(T3)** "Communication means for allowing involved stakeholders to be aware of the activities of the other stakeholders and exchange messages and information within the team" **(P5)**
PI-4	"Modelling collaboration protocols for collaborative modelling tools: Experience and applications"	"Collaboration protocol" **(T4)** "Groupware" **(T5)** "Collaborative modelling systems usually feature some tools for collaboration support such as communication tools, but these systems are not always made up of a single tool" **(P6)**
PI-5	"A model-driven development method for collaborative modelling tools"	"Groupware" **(T5)** "Collaborative modelling" **(T1)** "Collaborative modelling tools are usually domain specific" **(P7)**
PI-6	"Factors of collaborative working: A framework for a collaboration model"	"Collaborative working environments" **(T6)** "Conflict in collaboration" **(P8)** "Time barriers, for example working across different time zones or working to tight deadlines, may hinder communication and be a constraint on collaboration" **(P9)** "Reluctance of individuals to release, or even share, their own (tacit) knowledge" **(P10)**
PI-7	"Online collaboration: Collaborative behaviour patterns and factors affecting globally distributed team performance"	"Collaborative learning" **(T7)** "Synchronous collaborative technologies" **(T8)** "Time zone differences, individual differences and cultural differences among team members" **(P3) (P9)**

Publication Identifier	**Title**	**Extracted Solution Themes (T) and Associated Problems (P)**
PI-8	"Challenges in collaborative modelling: a literature review and research agenda"	"Collaborative modelling" **(T1)** "The stakes and reasons for involvement" **(P4)** "The modelling expertise of the participants" **(P3)** "The culture and country of the participants" **(P3)** "The number of participants involved in the modelling effort" **(P3)** "The model quality, both from an objective and a subjective perspective" **(P11)**
PI-9	"Integrating agile modelling with participative enterprise modelling"	"Agile modelling" **(T9)** "Participative Enterprise modelling" **(T1)** "Agile teams do not usually perform a thorough business analysis, which means that they do not require the business people to think systemically about their business and to connect their business needs to the new IS (Information System)" **(P12)** "They may also have different opinions about the same issue" **(P3) (P8)**
PI-10	"Participative enterprise modelling: Experiences and recommendations"	"Participative enterprise modelling" **(T1)** "Ensure that all the ideas and wishes of all participants are reflected in the model correctly" **(P11)** "The degree of problem complexity … will need a more experienced and skilled modelling team" **(P13)**
PI-11	"Investigating the influence of situational factors on participative enterprise modelling: making a case for a qualitative research approach"	"Participative enterprise modelling" **(T1)** "Degree of agreement among stakeholders regarding the nature of the problem and the goals for its solution" **(P8)**

2.2.5 Results for suggested solution areas

This section provides a *codebook* that can be used to analyse the text from literature studies related to *solution areas* as well as a table indicating the *extracted themes* from the short-listed studies presented in Table 5.

Codebook

The following codebook can be used to analyse the text from the identified literature studies. Coding will result in extracted *solution areas* and *solution tools* that address the problems or concerns that were identified in Section 2.2.4, associated with a particular solution theme, such as participative modelling. An inter-coder agreement is provided in Appendix D, and indicates that the codebook enables coding with a 100% overall agreement when applied by two independent coders.

- *Code label:* **Solution Area (SA)**

  Short definition: A means of solving an identified problem.

  Full definition: A solution area refers to a means of solving an identified problem within a certain solution theme.

  Example: computerised tools for more efficient participative modelling **(SA1)**

  When to use: When a fragment in the text indicates that a possible solution exists to address the problems or concerns associated with certain solution themes.

- *Code label:* **Solution Tool (ST)**

  Short definition: A commercial participative software application modelling tool.

  Full definition: A solution tool is a sub-code for solution area, and refers to a commercially available software application tool that can be used for modelling, more specifically interactive/participative/collaborative modelling.

  Example: diagrams.net **(ST1)**

  When to use: When a fragment in the text refers to software application tools that can be used for interactive/participative/collaborative modelling.

Extracted themes and associated solution areas/tools

The extracted solution areas and -tools, using the codebook presented above, for each article/source are provided in Table 8 below:

Table 8: Extracted Themes for Studies Related to Solution Areas

Publication Identifier	**Title**	**Extracted Solution Areas (SA) and Solution Tools (ST) for a particular Solution Theme (T)**
PI-12	“How Agile facilitators use Miro”	“Miro is a fast, free, and simple-to-use online whiteboard built to help you collaborate with others anytime, anywhere” **(ST1)**
PI-13	“Visual collaboration made easy”	“MURAL’s intuitive features make visual collaboration creative, engaging and productive” **(ST2)**
PI-14	“Participatory enterprise modelling with the 4EM method”	“Participatory enterprise modelling (EM)” **(T1)** “For Enterprise Modelling framework (4EM)” **(T10)** “EM application in practice is usually supported by computerized tools” **(SA1)** “Efforts on tools supporting 4EM – Miro.com and diagrams.net” **(ST1) (ST3)**
PI-15	“Identifying HCI patterns for the support of participatory enterprise modelling on multi-touch tables”	“Participatory enterprise modelling (PEM)” **(T1)** “New digital devices such as multi-touch tables appear very attractive in the context of PEM” **(SA2)**
PI-16	“The impact of the representatives of three types of process modelling tools on modeller’s perceptions and performance”	“We found that modellers perform their tasks significantly faster with a web tool in case of collaborative work” **(SA1)**
PI-17	“Knowledge Transfer of Software Tool Development for Functional Requirements Analysis”	“Continuous Collaboration Model (CCM)” **(T11)** “It is necessary to help users analyse all of their requirements with an efficient tool” **(SA1)**

Publication Identifier	Title	Extracted Solution Areas (SA) and Solution Tools (ST) for a particular Solution Theme (T)
PI-18	"Enabling Systems Modelling Language Authoring In A Collaborative Web-Based Decision Support Tool"	"The goal is to offer an experience where all users have up-to-date information and can concurrently modify the model" **(SA3)**
PI-19	"Simplifying the development of cross-platform web user interfaces by collaborative model-based design"	"Responsive design" **(T12)** "Visio: commercial editor of Microsoft for creating and sharing diagrams, enables collaborative features" **(ST4)**

2.2.6 Synthesised results

This section provides a discussion on the findings of the literature review, relating to the *class-of-problems* and *solution area(s)*. A shortlist of the *alternative solutions* that will be investigated for this study is also provided.

Class-of-problems

According to Younas et al. (2019), it is important to understand the *nature of participants* during interactive/participative modelling. Therefore, the following factors, and associated problems, need to be considered (Younas et al., 2019):

- Number, diversity, background and skills of the participants, as it may differ greatly.
- Timing and stages of participation – some participants may want to participate in all stages of the process, whereas others may want to only focus on specific areas.
- Level and intensity of participation – participants with a low participation level may benefit from an entirely different set of methods or tools than those with a high participation level.

Renger et al. (2008) refer to *group composition characteristics*, which could also be related to the *nature of participants*. These characteristics include the culture and country of participants, the stakes and reasons for participant involvement as well as the modelling expertise of participants. Renger et al. (2008) also refer to *group size*, where they state that the number of participants is of particular importance in collaboration and that productivity tends to decrease when the size of the group increases.

Another noticeable class-of-problems that appeared in more than one article, is *conflict among participants in collaboration*. Patel et al. (2012) reason that the conflict could be due to a lack of understanding of each other's skills and knowledge or due to incompatibility between participants' personalities, values, goals or opinions. Patel et al. (2012) also state that conflict can threaten work effectiveness, efficiency and even working relationships. Persson (2000) mentions that a high degree of mismatch in the agreement among participants regarding the nature of the problem and its solution goals results in a higher need for conflict resolution, which can be time-consuming.

Communication means is one of the main elements in driving software development activities and other model-based tasks, according to Franzago et al. (2017). They state that it is necessary that a means of communication is available for allowing involved participants to be aware of the activities of the other participants and for them to exchange messages and information within the team (Franzago et al., 2017). Gallardo et al. (2013) however mention that collaborative modelling systems usually feature some tools or components for collaboration support (communication tools such as voice chatting), but that these systems are not always made up of a single tool. Time barriers, such as time zone differences or tight deadlines may further hinder communication and be a constraint to collaborative working (Patel et al., 2012).

According to Renger et al. (2008) *model quality*, both from an objective and subjective perspective, is another challenge posed by collaborative modelling. Patel et al. (2012) mention that participants are sometimes reluctant to release or share their own knowledge, whereas Stirna et al. (2007) state that it should be ensured that all the ideas and wishes of the different participants should be reflected in the model correctly, which could also have an effect on the quality of the model.

Other less apparent problems that were identified include:

- The degree of problem complexity, where a more complex problem will need a more experienced modelling team (Stirna et al., 2007).
- Collaborative modelling tools are usually domain specific (Gallardo et al., 2012), which means that they might not be usable for all types of modellers.
- Feelings of model ownership could influence collaboration, 'MY' versus 'OUR' model viewpoints (Gutschmidt et al., 2021).

The main problems identified for PM can thus be summarised as problems related to: (1) *the nature of collaborative participants,* (2) *conflict among collaborative participants,* (3) *collaboration communication means* and (4) *collaboration model quality.*

Solution area(s)

Cheng et al. (2016) state that there exists a need for participants (users) to help analyse all of their system requirements during software requirements elicitation with the aid of an *efficient tool*. In addition, Polančič and Jošt (2016) state that modellers perform their tasks significantly faster with a *web-based tool* in the case of collaborative work, whereas Browne et al. (2013) mention that the goal of real-time collaboration is to offer an experience where all users have up to date information and are able to modify the model concurrently.

According to Stirna (2021) in the presentation slides from the NEMO 2021 conference, the application of enterprise modelling in practice is usually supported by *computerised tools*. The computerised tools that are *currently* in support of enterprise modelling are Diagrams.net and Miro (Stirna, 2021). Another tool that facilitates collaborative modelling is Microsoft Visio, according to Genaro Motti et al. (2013).

Gutschmidt et al. (2019) discuss how multi-touch tables can be used to enable participatory enterprise modelling. However, in accordance with the scope of this study, this solution area will not be considered as it is not a software application *tool*, but rather a *device*.

The solution area for the purpose of this study can thus be described as a *computerised (web-based) tool that supports participative/collaborative modelling.*

Alternative solutions

Based on the *solution tools* that were identified and extracted in Section 2.2.5, the alternative solutions considered for this study include Miro, MURAL, Diagrams.net and Microsoft Visio.

However, referring to the *minimum* requirements as listed in Section 4.1, Diagrams.net and Microsoft Visio were eliminated as these tools do not meet all of the stipulated minimum requirements. Diagrams.net is *not* highly interactive, and Microsoft Visio is *not* available as a low/no cost option.

Chapter 4 provides an in-depth comparison between the alternative solutions included for this study, namely Miro and MURAL.

3. Research Methodology

The aim of this chapter is to introduce the reader to the framework that was used to carry out the research for Chapters 4-6 of this dissertation.

An overview regarding *literature on research methodologies* that were considered for the purposes of this study is provided in Section 3.1. Detail on how the selected *research methodology was applied to the study* is provided in Section 3.2. The *rigour* that can be expected from the research methodology and *ethical considerations* are also provided.

3.1 Literature on research methodologies

According to a study by Peffers et al. (2018), design science research (DSR) has recently become a well-accepted research paradigm within the information systems (IS) discipline. The study specifies five DSR genres within the IS discipline, namely (1) *IS design theory*, (2) *DSR methodology*, (3) *design-oriented IS research*, (4) *explanatory design theory* and (5) *action design research*.

The aim of the first genre is to develop *IS design theories* as principal artefacts of knowledge generation and representation. Instantiations of the artefacts is not a requirement for this theory, but could enhance presentation to an audience.

In contrast, *DSR methodology* focuses on artefact development. The research effort could even start with an already designed version of an artefact. The emphasis is to design and construct an applicable artefact that could increase the efficacy of an organisational IS.

Design-oriented IS research is typically practiced within the German-speaking IS community. This genre aims to design better preforming IS solutions, with usefulness for practice as a critical measure.

Explanatory design theory places emphasis on design features and their impact on the environment. It is essential for this genre to have at least one design variable that can be manipulated in a systematic manner, as well as hypotheses that allow for evaluation.

Finally, *action design research*, places focus on DSR within an organisational context. It is particularly useful when researchers have the opportunity to work within an organisation to develop the artefact, incorporating stakeholder and end-user participation as well.

The DSR methodology was selected as an appropriate research methodology, as this study is aimed towards the extension of an existing artefact, namely the SCM, as presented in Section 2.1.1. DSR is discussed in more detail below.

Design Science Research (DSR)

The goal of DSR is to create and evaluate information technology artefacts that can solve problems identified in organisations, thus including any object designed with an embedded solution to an understood research problem (Peffers et al., 2007).

Baskerville et al. (2018) mentions five DSR objectives. The first objective relates to *technology and science evolutions*, and states that DSR projects must be accurately established in the cycles of science-technology evolution for the selected application domain. The second objective relates to *design artefacts*, and states that a DSR project must incorporate the building and evaluation of a novel design artefact. The third objective relates to *design theories*, and states that the value of design knowledge should be defined as both inputs and outputs of a DSR project. The fourth objective relates to *DSR processes*, and states that a reference process should be followed in conducting the research. The fifth objective relates to *DSR impacts*, and states that practice should be impacted by design artefacts and research should be impacted by design theories.

According to Peffers et al. (2007), the DSR methodology consists of the following activities, (also shown graphically in Figure 3 below):

Activity 1: Problem identification and motivation

Define the particular research problem and justify the value that a solution will contribute.

Activity 2: Definition of solution objectives

Derive the objectives of a solution from the outputs of the abovementioned step. The objectives can either be quantitative or qualitative.

Activity 3: Design and development

Create the actual artefact, based on its desired functionality and architecture. The artefact can be a construct, model, method or instantiation.

Activity 4: Demonstration

Demonstrate the ability of the artefact to solve one or more instances of the problem.

Activity 5: Evaluation

Asses and measure to what degree the artefact supports a solution to the problem.

Activity 6: Communication

The last activity, the need for *communication* in order to spread resulting knowledge, was proposed by Archer L (1964) and Hevner et al. (2004). Communicate the problem and its importance, and the artefact and its effectiveness to researchers and other relevant audiences.

A study by Offermann et al. (2010) stated that there is no consensus among researchers as to what the actual output of DSR is and what types of artefacts exist. Hence, the authors conducted a literature review, including 62 papers, to determine 8 distinct artefact types, namely (1) system design, (2) method, (3) language/notation, (4) algorithm, (5) guideline, (6) requirements, (7) pattern and (8) metric. The abovementioned artefact types as described in *activity 3* from Peffers et al. (2007) were included in the literature study by Offermann et al. (2010) as "personal typologies" of the authors.

According to Sonnenberg et al. (2012), the evaluation of the artefact is *crucial* for DSR in order to proof that the artefact is relevant for practice. Sonnenberg et al. (2012) therefore propose four generic evaluation activities within a DSR process, also aligned with the six DSR activities of Peffers et al. (2007).

The first activity relates to the evaluation of the *problem identification* in order to ensure that the selected DSR problem is meaningful. Some evaluation methods that could be applied within this activity include assertion, a literature review, expert interviews and surveys.

The second activity is an artificial evaluation relating to the *design*, with the aim of showing that the artefact design embeds the solution to the problem. Examples of evaluation methods that could be applied within this activity consist of logical reasoning, mathematical proof, simulation and benchmarking.

The third activity relates to *demonstration*, and serves to determine how well the artefact performs during interaction. Possible evaluation methods include prototype demonstration or experimentation, surveys, benchmarking and expert interviews.

The fourth and final activity relates to the *evaluation* activity of Peffers et al. (2007), and serves to ultimately determine whether the artefact is both applicable and useful in practice. Case studies, field experiments, surveys and focus groups are a few of the evaluation methods that could be applied within this activity.

Figure 3 indicates how the DSR process is structured in a nominally sequential order, with more than one entry point. The basis of the nominal sequence, is starting with *activity 1* i.e. following a problem-centred approach. Starting with *activity 2* leads to an objective-centred solution. A design- and development-centred approach is initiated by starting with *activity 3*. The final entry point, starting with *activity 4*, leads to a client- or context-initiated solution.

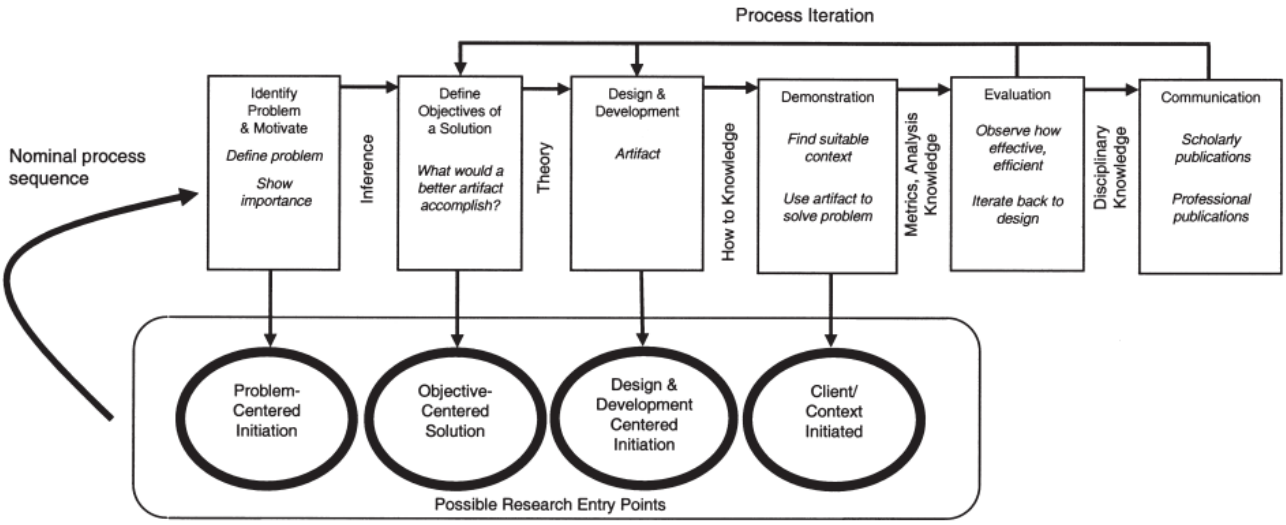


Figure 3: Design Science Research Methodology Process Model (Peffers et al., 2007)

A taxonomy for DSR studies was developed by Smuts et al. (2022), useful in identifying DSR study design principles and also for designing DSR studies in a systematic way. A total of 72 studies were analysed in order to develop the taxonomy, consisting of 7 dimensions with certain characteristics specified for each dimension. Example studies are also provided in the article, illustrating how the taxonomy may support the identification of DSR research patterns in future research. The authors concluded that the taxonomy should provide guidance to DSR researchers, but that the dimensions and characteristics of the taxonomy could include some form of bias.

Hevner et al. (2004) provide practice rules for carrying out DSR in the IS discipline. These practice rules exist in the form of 7 guidelines, and represents the characteristics of well executed research. The guidelines are described as follows: (1) the output of the research must be a feasible artefact in the form of a model, construct, method or instantiation, (2) the objective should be to enable the development of technology-based solutions to unsolved and relevant business problems, (3) the “quality, utility, and efficacy” of the artefact must be evaluated in a rigorous manner, (4) the research must present contributions that are clear and verifiable, (5) rigorous methods must be practiced in both the construction and evaluation phases of the artefact, (6) the search process in deriving a solution to a defined problem should be based on existing theories and knowledge and (7) the research must be presented effectively to relevant audiences.

However, a study by Venable (2010) indicated that the abovementioned guidelines, as stated by Hevner et al. (2004), have come to dominate the DSR criteria, standards, guidelines and expectations, and that the opinions of other scholars should be taken into account as well. A survey was conducted, including participants with the following roles: (1) high quality journal gatekeepers, (2) DSR conference gatekeepers and (3) authors specialising in DSR. Feedback results from respondents indicated that caution should be exercised in using the Hevner et al. (2004) guidelines as a *mandatory* checklist for evaluating DSR projects, and that one should not expect single papers to adhere to *all* guidelines.

3.2 DSR methodology applied to the study

The *software tool implementation* of the SCM is the *artefac*t to be developed and evaluated in this study. This new artefact is based on the existing artefact, namely the SCM steps which can be classified as a *method* type of artefact.

Following a problem-centred approach, the 5 activities/steps of the DSR cycle according to Peffers et al. (2007) as presented in Section 3.1 above, are addressed in this study as follows:

1. *Problem identification* – A study that focused on the development and evaluation of an online-adapted version of the SCM indicated a concern, namely that the previous modelling software (Diagrams.net) does not encourage *active participation during modelling* when applying the SCM steps (De Vries, 2022). The *latency* of Diagrams.net is a problem, since there is a considerable *delay* in updating the diagram, which *hampers* active participation during modelling. Hence, there exists a need to extend the SCM with a suitable PM software application tool that supports the SCM.
2. *Definition of solution objectives* – In addressing the concern as identified in *step 1* above, the objective of the study will be to investigate, experiment with and select a

software modelling tool that promotes PM and supports the SCM by enabling step-wise guidance of the SCM.

3. *Design and development* – Select an appropriate modelling tool by experimenting with two alternative short-listed tools, used in combination with the SCM, involving two research participants during the iterative process. Perform this process by (1) exploring the features of both tools (2) identifying certain entry requirements that the desired tool should meet and (3) select a single tool based on its ability to address the entry requirements. Once a single tool is selected, embed the SCM steps within the new software tool in order to support the SCM and to promote PM.
4. *Demonstration* – Demonstrate during an interactive online session to post-graduate participants how a fictitious case can be applied when using the SCM in combination with the new software modelling tool.
5. *Evaluation* – Obtain survey feedback on the *usefulness* of the modelling tool by requesting that post-graduate participants apply the SCM within their own industry context, using the new software tool. Evaluate whether the selected tool encourages PM and also evaluate to what extend actual participation took place during the PM exercise.

The last step, relating to *communication,* is supported by the writing and submission of this dissertation. The mapping of a single DSR process cycle as described in an article by van der Merwe et al. (2017) is applied in this dissertation in order to provide a structured report on master's degree level. The mapping exists between a research report structure and DSR process model, and consist of the following: (1) mapping of the *introduction* with the *problem awareness* phase of the DSR process model, (2) mapping of the *introduction* with the *suggested solution (artefact)* phase of the DSR process model, (3) mapping of the literature review with the *problem awareness* phase of the DSR process model, (4) mapping of the literature review with the *suggested solution (artefact)* phase of the DSR process model, (5) mapping of the *method* to the *development* phase of the DSR process model, (6) mapping of the *body of the report* to the *development* phase of the DSR process model, (7) mapping of the *body of the report* to the evaluation phase of the DSR process model and (8) mapping the *conclusion* to the *study and research contribution*.

The *data gathering and analysis tools* that were used for this study, mapped to the respective DSR steps, are discussed below:

Thematic analysis

Thematic analysis can be described as a method for identifying, analysing and reporting certain themes that stand out from a data set and is widely used as a qualitative data analysis technique within empirical software engineering research, as it provides an in-depth understanding regarding the content of the data (Wohlin and Aurum, 2015).

In line with the first step of DSR, *problem identification and motivation*, two thematic analyses were conducted in Chapter 2 (refer to Sections 2.2.4 and 2.2.5) in order to analyse extracted themes from identified literature studies. The first analysis was done to identify the *class-of-problems* related to this study, whereas the aim of the second analysis was the identification of different *solution areas.*

Experiments

According to Wohlin and Aurum (2015), experiments are often conducted with the aim of comparing two or more tools, methods or techniques.

Two rounds of experimentation formed part of this study, namely a *pilot* experiment that was conducted during the *solution objectives* phase of the DSR cycle, and the *main* experiment that was conducted during the *evaluation* phase of the DSR cycle. For the *pilot* experiment, an iterative process was used to experiment with two PM tools. Two additional participants (who were not included in the main experiment) formed part of the experimentation team. An exploratory or inductive approach was used to identify the main features of the two tools, as well as to determine which of the features are absolutely necessary when using the tool to apply the SCM. Only one of the two tools was selected to be used for the *main* experiment, where post-graduate participants had to apply the SCM within their own industry context, using the new software tool.

Surveys (questionnaires)

Surveys form part of a quantitative research method that is commonly used in both empirical software engineering and IS research, and may elicit data from individuals, groups or even organisations (Wohlin and Aurum, 2015).

As part of the *evaluation* step of the DSR cycle, a survey was distributed to participants to obtain feedback on tool usefulness. A few questions relating to the backgrounds of the participants were also added to ensure that the results are interpreted within a specific context.

However, in order to protect the confidentiality of the participants, the survey data was analysed in such a manner that results were not directly linked/traced back to individual responses.

The survey questions are as follows:

1. What is your existing role at the enterprise?
2. What is the main type of industry where you are currently employed?
3. What is your tertiary education?
4. Is your enterprise currently using modelling tools to represent process/activity logic at the enterprise? Please specify.
5. From a participant (facilitator) perspective, did you follow the SCM, allowing your colleague to co-model, aligned with instructions provided by the SCM?
6. From a participant (facilitator) perspective, do you believe that the tool hampered participative modelling and created frustration when applying the SCM?
7. From a colleague perspective, do you believe that the tool hampered participative modelling and created frustration when applying the SCM?
8. From an ease-of-use perspective, would you recommend that [insert selected software modelling tool] is used when having to participate with other team members during participative modelling in the future?
9. From a participant (facilitator) perspective, if you had to use the SCM again in future to facilitate teaching on the CSD, would you prefer to use face-to-face facilitation and drawing on a whiteboard, rather than using online modelling?
10. In terms of the [insert selected software modelling tool] tooling, you or your colleague may have experienced some frustrations related to tool functionality. Please elaborate on these frustrations?

The survey results are discussed in detail in Chapter 6.

3.3 Expected rigour

The practise rules discussed in Section 3.1 highlights the importance of rigorous methods. ‘Rigour’ refers to the act of being consistent and thorough. Therefore, the expected rigour for each of the abovementioned techniques are provided below.

Thematic analysis - Data triangulation, defined as the inclusion of data from multiple sources, was used to ensure that the applied technique does not lack thoroughness. An inter-coder agreement was also used to measure the consistency of the data analysis.

Experiments – To ensure that consistent participation was evident in the main experiment (i.e. during the *evaluation* phase), a clear and precise method/set of instructions was defined and provided to participants. An online demonstration, using a fictitious case, was also given to all participants prior to the main experiment. Participants were encouraged to also interact and provide feedback during the online demonstration.

Surveys - The objectives of the survey were clearly communicated to participants during online training, as well as in the cover letter of the survey. The purpose was to ensure that the survey was conducted in a consistent and thorough manner. To enable some form of standardisation in the survey responses, the format of questions 5-9 consisted of a five-point scale, i.e. strongly agree / agree / neither disagree nor disagree / disagree / strongly disagree, with the option to provide a motivation. Consideration was also given to the number of questions to avoid the survey from becoming tedious.

It is acknowledged that different forms of *bias* could be present in the survey results, as all participants do not necessarily have the same background and experience with regards to modelling tools.

3.4 Ethical considerations

As stipulated by the faculty of engineering, built environment and information technology ethical clearance committee, (see the following link: https://www.up.ac.za/faculty-of-engineering-built-environment-it/article/15815/faculty-committee-for-research-ethics-integrity) ethical clearance need to be obtained when humans (either as informants and/or subjects) are included as participants in studies. In accordance with the guidelines, the ethical process was followed carefully by the researcher and full ethical clearance was obtained for this study.

4. Alternative Solutions Comparison

In answering secondary RQ1 of the study, namely *"How were the two alternative solutions compared to derive at a single solution?",* this chapter provides the comparison results in tabular format from exploring the alternative solution tools, namely Miro and MURAL. The two alternatives were short-listed using the *minimum requirements* are presented in Section 4.1. The *feature exploration and entry requirement identification* of both tools are presented in Section 4.2, whereas the final *solution tool selected for the study* is discussed in Section 4.3.

4.1 Minimum requirements

The following set of *minimum* requirements for the PM tool were identified in collaboration with the SCM designer, and relates to the SCM that was presented in Chapter 2.

The PM tool should:

1. Be supporting of the SCM operating context, conveyed via a CSD.
2. Be of low/no cost.
3. Be highly interactive with no delay.
4. Facilitate the development of a template, i.e. a template that facilitates PM, using DEMOSL version 4.

The abovementioned requirements were considered when determining possible solution alternatives for inclusion for this study. As mentioned earlier in Section 2.2.6, the possible solution tools identified after synthesising the results from the literature review (Chapter 2), include Miro, MURAL, Diagrams.net and Microsoft Visio. Based on the minimum requirements as listed above, Diagrams.net and Microsoft Visio were eliminated as these tools do not meet all of the stipulated minimum requirements. Diagrams.net is *not* highly interactive, and Microsoft Visio is *not* available as a low/no cost option.

Thus, the two short-listed tools to be considered for this study is Miro and MURAL.

4.2 Feature exploration and entry requirement identification

An iterative process was used to experiment with two PM tools, namely Miro and MURAL. This experiment is referred to as the *pilot* experiment and consists of three main phases: (1) *feature exploration*, (2) *entry requirement identification* and (3) *tool selection*.

During the *feature exploration* phase, an initial SCM template (refer to Section 5.1 for more detail on the template) was developed for Miro and MURAL via their free Education Plan. The researcher of this study, in collaboration with the SCM designer, had full administrative rights on the tools and used an exploratory or inductive approach to identify the main features of the tools. Two additional participants (who did not take part in the *main* experiment) formed part of the experimentation team and were invited to act as *members* or *visitors* on the tooling platforms, with restricted access rights. The identified features are provided in the first column of Table 9, also including sub-headings (indicated in bold and italics) to group features into categories. The second column, *requirement and motivation*, provides a more detailed description of the SCM-related requirement in terms of the indicated feature.

During the *entry requirements identification* phase, some of the identified features were classified as entry requirements i.e. those features that are absolutely necessary when used in combination with the SCM. The column *Ent* in Table 9 indicates with a "1" whether a particular feature has been classified as an entry requirement.

The results from the *pilot* experiment are displayed in the two columns in Table 9 labelled as Miro and MURAL. The results are *qualified* and *quantified* for each of the tools. For qualification, a detailed description is provided on how the specific tool addresses the corresponding requirement. For quantification, the level of addressing a requirement is indicated as 1 (fully addressed), 0.5 (partially addressed) and 0 (not addressed).

The third phase, *tool selection*, is discussed separately in Section 4.3.

Table 9: Tool Comparison Results

Feature	**Requirement and Motivation**	**Ent**	**Miro**		**MURAL**	
			Qualify and Quantify		**Qualify and Quantify**	
Participation enablers						
Interactive modelling	There should not be any latency in reflecting updates performed by multiple participants.	1	Updates reflect immediately.	1	Updates reflect immediately.	1
Integrated voice-chat function	Users should be able to communicate verbally while they produce diagrams. An integrated feature is not an entry requirement, since it is also possible to use the tool in combination with other platforms such as Zoom and Google Meet.		Miro has a video chat feature, but only works for a small number of participants. See: https://redpepper.land/blog/miro-v-mural/	0.5	Own built-in "chat" tool. Visitors* cannot "use the chat tool to chat with other collaborators". See: https://support.mural.co/en/articles/2113739-invite-visitors-to-collaborate	1
Privileges for using the tool						
Workspace membership control	Full members should be prevented from adding new members if member	1	The Education Plan allows for 1 administrator and 100 members. Possible to control	1	The Education Plan allows for 2 administrators and 20 members + 25 guests*.	1

Feature	Requirement and Motivation	Ent	Miro		MURAL	
			Qualify and Quantify		Qualify and Quantify	
	numbers are restricted. During the main experiment, the administrators need to ensure that participants of the experiment have "full member" privileges.		membership sign-up via the option: "New team members from the domain list have to ask team admins for permission to join".		Memberships can be controlled with the option: "Members can invite existing members & guests and *no new members or guests".*	
Workspace privilege control	Administrators should have control over workspace privileges, e.g. preventing members from adding new projects/rooms. The experiment requires monitoring of activity within the projects/rooms and thus projects/rooms had to be created by the administrators.	1	Cannot manage workspace privileges, e.g. preventing members from creating projects. Only possible to control privileges per project (called "rooms" in MURAL).	0	Possible to manage all workspace members on one single page, enabling/disenabling per member: create rooms, publish templates, see open rooms, workspace admin and remove user.	1
Board membership control	Should provide board membership to selected members (i.e. all participants in the SCM experiment).	1	Possible to share a board to the entire Education team or selected Education team members. "Sharing settings" per board allows for removing members.	1	Possible to share a board (also referred to as a mural) to the entire Education team or selected Education team members, up to 50 members. Possible to add unlimited number of visitors* to the board.	1
Board access control	Should be able to manage the users (members and visitors) per board, also removing users from a board.	1	Can remove users that are members. Cannot remove users that are visitors on a public board.	0.5	Full visibility of users per board. Can also remove a user per board. Can change a board's link to "remove" visitors.	1
Board access control relating to new memberships	Should allow Education members to invite visitors from outside with "edit" abilities on a board without occupying full memberships. The SCM experiment has to allow participants to invite their colleagues to join an SCM board.	1	A non-member is indicated as a guest, still occupying a membership. No control is possible on the additional guests that are invited by Education team members.	0	"Visitors are invited via a secure link to access a specific mural without signing in. They do not take up a membership". Also, "Visitors who join with editing permissions can collaborate in a mural just as though they were a member."	1
User type access for board access	Users that are invited to a board should also allow access to "industry" users with editing access. For the SCM experiment, participants need to invite a colleague from industry to participate/edit on a board.	1	Allows for any invitee access, if the board is public, but then only as "commenter". Will occupy full membership if editing is needed.	0.5	Administrator can create private rooms with the setting "Members cannot invite new members/guests" to private room. Member can still send a link via email to industry user that allows full editing rights to a mural/board within a private room.	1
Exporting						
Exporting to .csv	Should facilitate exporting of a table to a .csv file, i.e. exporting the TPT. The .csv format is useful when the TPT has to be further extended into user stories.		Full export possible, keeping the structure of the original table as separate columns in the .csv file.	1	Select a number of sticky notes, right-click to export to .csv. Content of the sticky notes are also grouped into one column in the .csv file.	1
Exporting diagram to .png or .pdf format	Should facilitate exporting of the CSD, used as part of		Select diagram only for export.	1	Can create an "area" to include a diagram and then	1

Feature	Requirement and Motivation	Ent	Miro		MURAL	
			Qualify and Quantify		Qualify and Quantify	
	the SCM, to .png or .pdf. The CSD may be used to specify the operating context of an enterprise and may have to be included in a report.				only export the area as .png or .pdf.	
Ease of modelling						
User interface	Should have an intuitive user-interface in terms of the layout of modelling options, zooming in and out and "undo" a change with a button.		Possible.	1	Possible.	1
Construct editing control	Should be able to group constructs into areas to lock/unlock areas for participating users, indicating its locked status.		Possible.	1	Possible.	1
Moving and re-sizing a construct within a "group"	Should be able to easily move/re-size a single construct within a group. It should be possible to increase the size of a construct that forms part of a group to ensure that the embedded text is readable.		Easy. Double-click to move construct within a group.	1	Cumbersome. Need to un-group, re-size and re-group.	0.5
Support and training						
Response time	Should receive a response within 24 hours from the support team.		24 hours response on Miro Community.	1	24 hours response via e-mail.	1
Self-training	Should facilitate self-training via web pages and videos.		Possible via web pages and videos.	1	Possible via web pages and videos.	1
Version control						
Change tracking	Should be able to track changes per user that are made for the contents of a board for the lifetime of a board. For the SCM experiment, it is necessary to keep track of the level of participation in making changes to a board.	1	Limited, only 30 days.	0.5	No constraints on activity tracking.	1
Version roll back	Accidental deletion of a large part of board content should be rolled back.		Restore content of a previous version.	1	No restoring function.	0
Facilitator Control						
User attention control	Facilitator should be able to take control of participants by locking the screen for participating users, i.e. they cannot add new content until the facilitator releases them.		Not possible.	0	Possible to "summon" participating users.	1
Locking some board contents	Facilitator should be able to lock certain areas that may not be unlocked by participating users of a board.		Only facilitator can unlock some areas for a board owned by the facilitator.	1	Only facilitator can unlock some areas for a board owned by the facilitator.	1

Feature	Requirement and Motivation	Ent	Miro		MURAL	
			Qualify and Quantify		Qualify and Quantify	
Hiding some content	Facilitator should be able to hide some frames/areas on the board to keep participating users focused on a particular step within the SCM.		Possible. Right-click and hide an area.	1	Possible. Use the "outline" of areas to hide an area.	1
Template reusability						
Template design and sharing	A member should be able to design a new template for new languages and/or versions of a language, such as the DEMOSL, and share the template to members of the workspace.		Possible to create a "personal" or "shared" template.	1	Possible to design a template with "Share this link to bring people to this template". A visitor, i.e. not a full member, will be able to use the template to create a mural (i.e. board). Template owner can "delete link to revoke access".	1
Template constructs created as a library of constructs	Should be able to create and share a library of constructs to enable quick drag-and-drop of tailor-made graphical constructs. As an example, each of the graphical constructs that form part of DEMOSL should be selectable from a library.		No library options.	0	Library option to save tailor-made constructs to drag-and-drop onto a workspace. However, the constructs are only available locally, and not shareable to other users (i.e. members or visitors).	1
Template guidelines	The tool should provide a means to guide the template user to sequentially create a new diagram, based on template constructs. The SCM includes a number of steps that should be performed in a very specific sequence to ensure the quality of the diagrams.		Existing templates provide click-through step-by-step guidance on the right-hand-side of the diagram. Unclear whether the Education Plan allows a member to create such click-through guidance.	0.5	Possible to add areas to an "outline". Clicking on the outline items will move the cursor to the linked area.	1

* *Visitors* and *guests* refer to different types of collaborators, with limited permissions.

4.3 Solution tool selection

As seen in Table 9 above, Miro only *fully* addresses 3 out of 8 specified entry requirements. However, MURAL *fully* addresses *all* of the entry requirements i.e. 8 out of 8 specified requirements. Therefore, Miro was disqualified and MURAL was selected as the PM solution tool to be used for the *main* experiment of this study. The detailed comparison of the tools, also including non-entry requirements, is one of the contributions of this dissertation, as will be discussed in Chapter 7.

5. Solution Development and Demonstration

Secondary RQ2 of this study, namely *"How was the chosen solution developed and demonstrated?"*, is answered in this chapter. The development and demonstration phase of the solution is illustrated and explained, including the *SCM development in MURAL* (Section 5.1) and the *SCM demonstration in MURAL* (Section 5.2).

5.1 SCM development in MURAL

A final template was developed in MURAL for participants to use during the *main* experiment of this study. Figure 4 provides an illustration of this template. On the left-hand side, two groups of symbols are available, namely *Basic BPMN Symbols* and *DEMOSL 4+ CSD Symbols*. On the right-hand side, an *Outline* is used to provide methodical guidance in applying the SCM, i.e. including the *13 Steps* that from part of the SCM (as detailed in Section 2.1.1). When a user of this template clicks on the button *Create mural from template*, a new board (also called a mural) is created within a user-selected room. The user may invite several other users to join the board and co-model, following the steps as listed in the *Outline*.

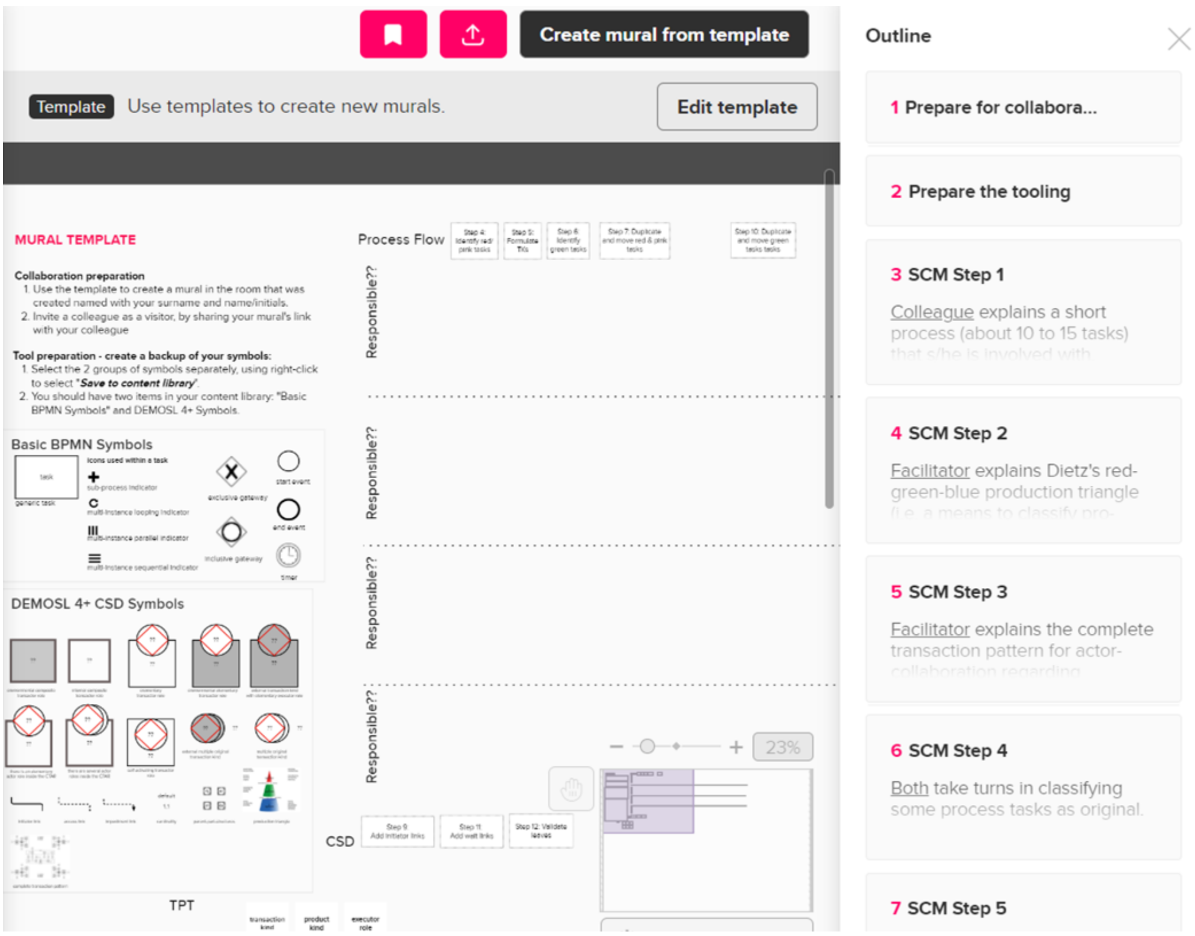


Figure 4: SCM Template in MURAL

Referring back to the three main characteristics, as defined by Stirna and Persson (2018), for a participative approach, MURAL and the SCM template incorporates the characteristics as follows: (1) The template's *Outline* provided a defined way of working in the form of methodological steps to carry out the modelling session with explicit principles of stakeholder involvement indicated in detailed descriptions per step; (2) For each session, a facilitator leads the session, where the facilitator needs to have knowledge about DEMO and the facilitator has to involve a colleague that is knowledgeable about a particular operating context at a real-world enterprise; and (3) Each facilitator, acting as modelling facilitator, was responsible to guide the discussion with a colleague about using the SCM.

5.2 SCM demonstration in MURAL

The SCM method steps were demonstrated, prior to the *main* experiment, to participants during an online session. The demonstration was based on a fictitious case, namely post-graduate operations at some tertiary education institution, and was presented collaboratively by the SCM designer (as facilitator role) and the researcher of this study (as colleague role).

Figure 5 below represents the flow chart (consisting of 12 tasks) obtained from performing *Step 1* of the SCM.

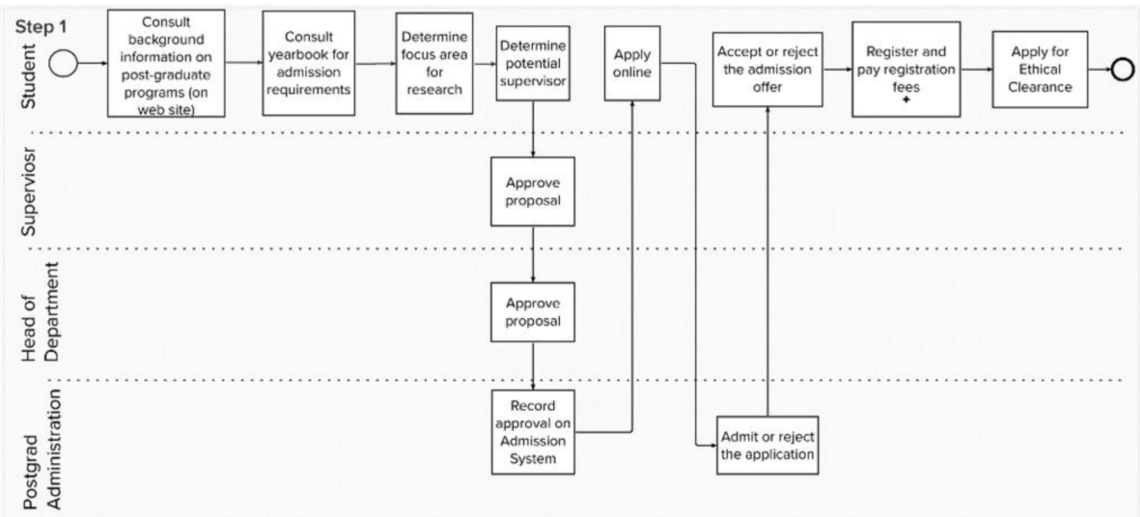


Figure 5: Demonstration in MURAL for SCM Step 1

Since *Steps 2 and 3* from the SCM are explanation steps, no graphical representations are included for these steps. Performing *Steps 4-6* of the SCM, analysing the flow chart, resulted in the extract provided in Figure 6.

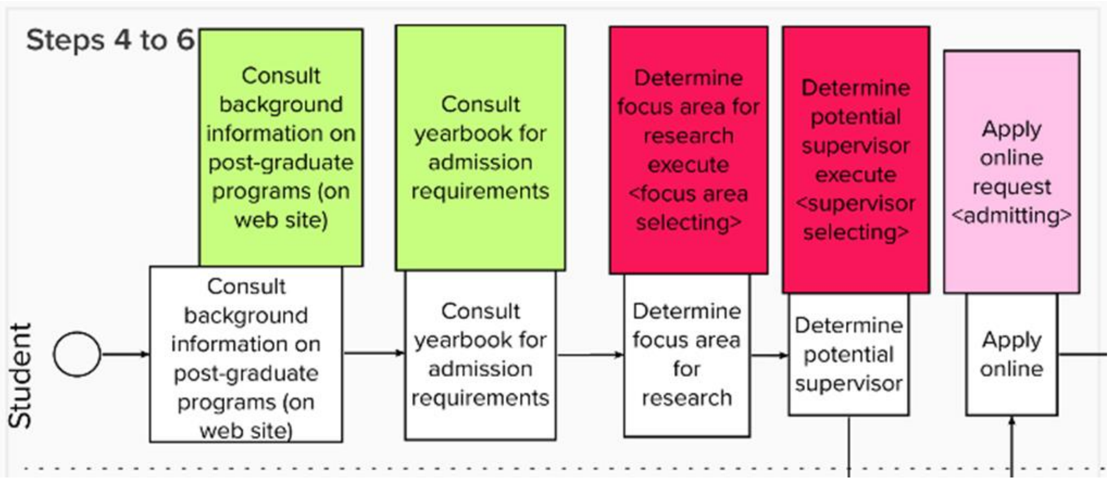


Figure 6: Demonstration in MURAL for SCM Steps 4-6

The remainder of the steps, i.e. *Steps 7-13*, can be seen in Figure 7 below. *Steps 7-12*, converting flow chart constructs into a CSD, are shown on the left-hand side of the figure, whereas *Step 13* i.e. compiling a TPT is shown on the right-hand side of the figure.

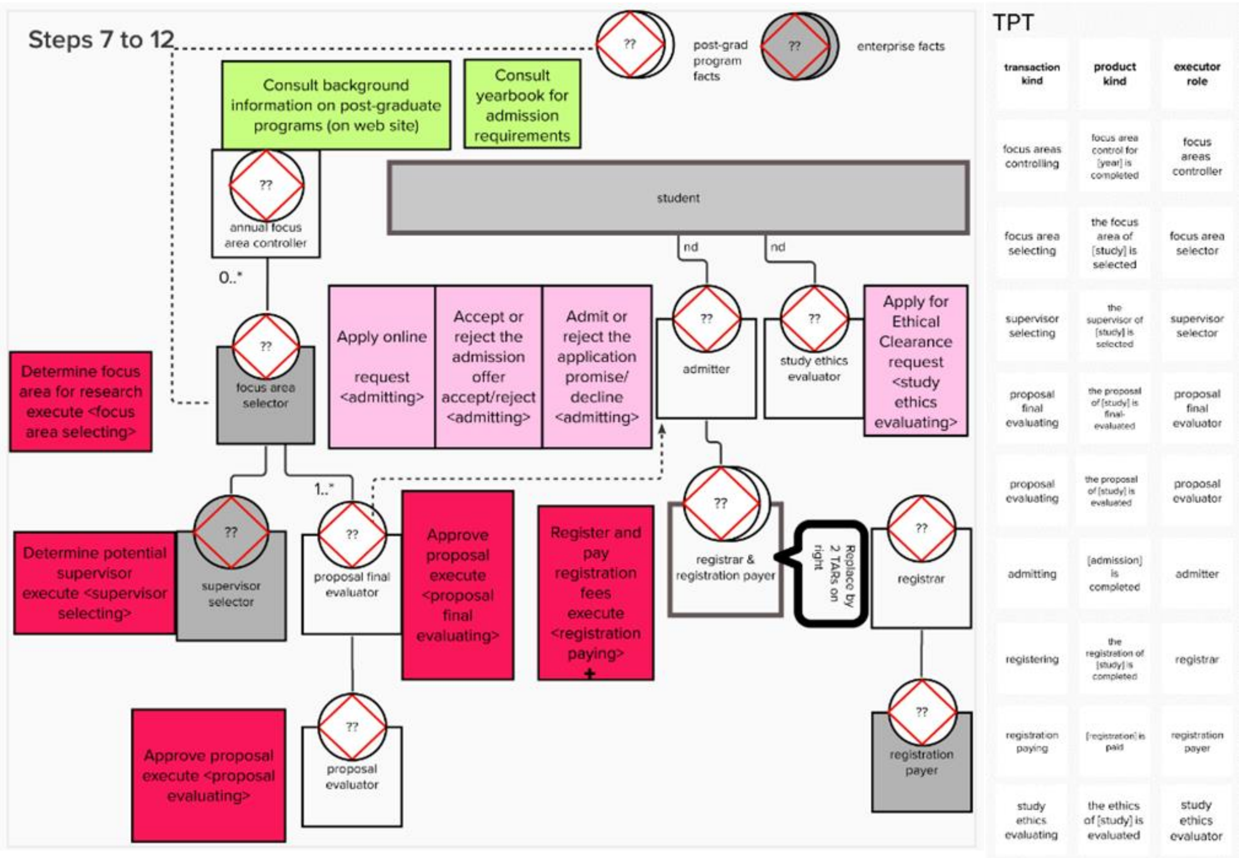


Figure 7: Demonstration in MURAL for SCM Steps 7-13

6. Evaluation Results

This chapter addresses secondary RQ3, namely "*How was the final solution evaluated and what are the findings?*", by presenting the survey results obtained from the *main* experiment. Although 36 participants were actively involved in the experiment, only 25 voluntary surveys were completed. Section 6.1 provides a discussion on the relevant *participant background*, whereas Section 6.2 includes the *participative modelling results. MURAL's participative modelling abilities* as well as the *actual level of participation* form part of the *participative modelling results* presented in Section 6.2. Sections 6.1 and 6.2.1 are based on the reported survey questions as specified in Section 3.2 of this dissertation.

6.1 Participant background

Based on the first survey question, most participants (7 out of 25) indicated that their existing role at the enterprise is *full time student*. The role of *systems analyst* was indicated by 3 out of 25 participants, whereas the role of *business analyst* was indicated by 2 out of 25 participants. *Supply chain/logistics engineer* was also indicated by 2 out of 25 participants. The remainder of the participants (11 participants) indicated the following roles: *implementation specialist, industrial engineer, IT business analyst, production shift supervisor, manufacturing engineer, systems engineer, design engineer, operations planner, mining engineer, support engineer* and *process analyst.*

The second question asked participants to indicate the main type of industry where they are currently employed. Participants who indicated in *question 1* that their existing role at the enterprise is *full time student,* also indicated for *question 2* that they are *full time students*. The *mining* and *financial services* industries were both indicated by 3 participants, respectively. The *automotive* and *retail/wholesale/distribution* industries were both indicated by 2 participants, respectively. The remainder of the participants (8 participants), indicated the following types of industries: *industrial manufacturing, software development, media and entertainment, professional consulting services, construction/engineering, aerospace and defence manufacturing, chemicals* and *telecommunications.*

The tertiary education background of participants was obtained from *question 3*. All participants indicated that they have a Bachelor of Engineering or Bachelor of Technology undergraduate degree, covering multiple disciplines i.e. *industrial* (13 out of 25), *mining* (4 out of 25), *metallurgy* (3 out of 25), *mechanical* (2 out of 25), *electronic* (2 out of 25) and *chemical* (1 out of 25).

For the last question related to participant background (*question 4*), most participants (23 out of 25) indicated that their enterprise is currently using modelling tools to represent business process/activity logic. Some participants even indicated that more than one tool is being used, which included *Microsoft Visio* (16 out of 25), *Diagrams.net* (9 out of 25), *MURAL* (2 out of 25), *Miro* (2 out of 25), *Lucidchart* (1 out of 25), *MagicDraw* (1 out of 25), *Microsoft PowerPoint* (1 out of 25), *ARIS* (1 out of 25) and *Enterprise Architect* (1 out of 25).

6.2 Participative modelling results

The PM results were evaluated in two ways: (1) *MURAL's abilities to encourage participative modelling* and (2) the *level of participation* during modelling.

6.2.1 MURAL's participative modelling abilities

Opinions regarding MURAL as enabler to facilitate PM are consolidated in Figures 8-12 below, based on *questions 5-10* as listed in Section 3.2.

For *question 5* (see Figure 8), the participants who disagreed or strongly disagreed indicated that their colleague who had to co-model the operating context was not from an engineering background, and had difficulties in understanding the concepts used in the SCM.

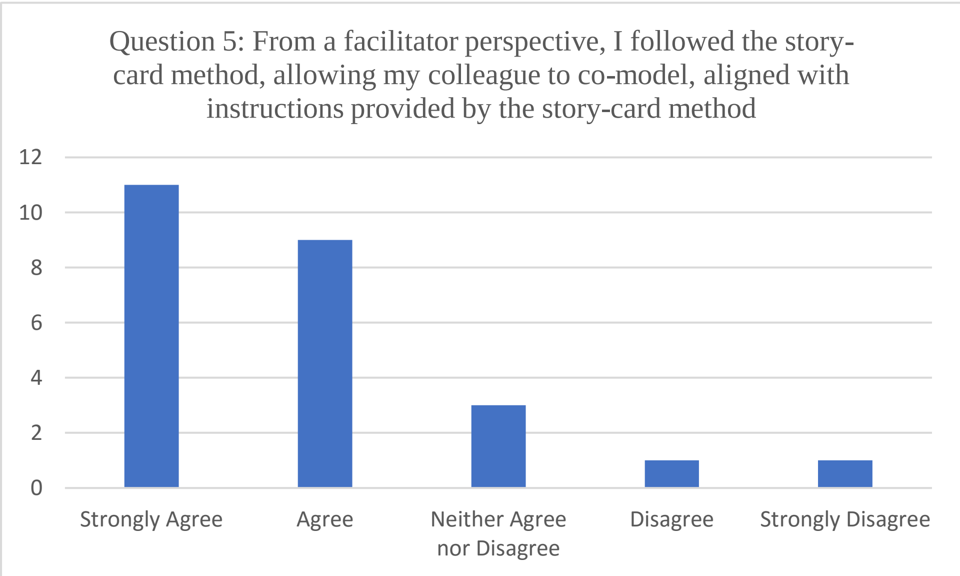


Figure 8: Participant Responses for Question 5

For *question 6* (see Figure 9 below), some participants who agreed, elaborated by indicating the following: (1) some functionalities in MURAL (e.g. ctrl + c) did not work, (2) the "undo" function did not always work and (3) the exports from MURAL are not readable.

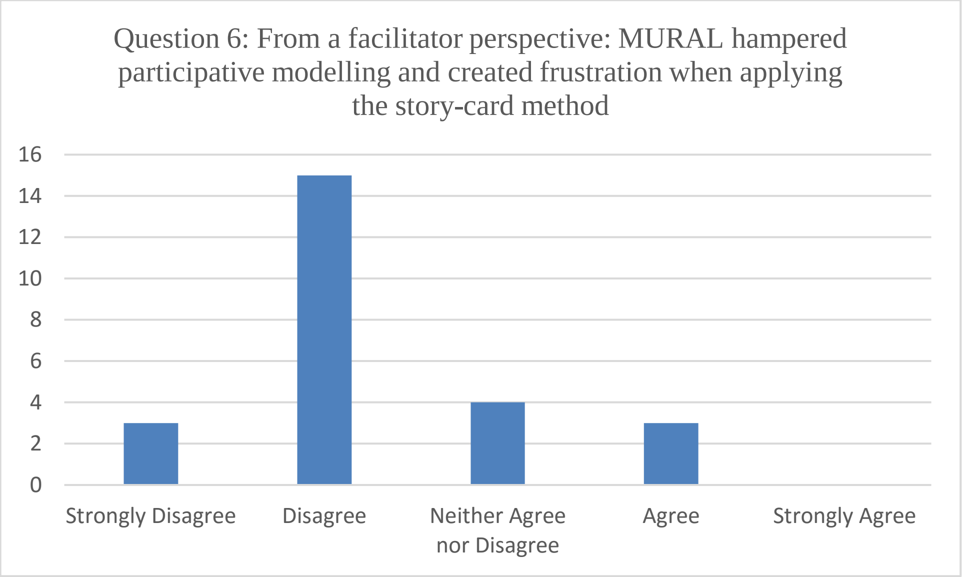


Figure 9: Participant Responses for Question 6

For *question 7* (see Figure 10), participants who agreed/strongly agreed provided additional motivation, indicating that: (1) the colleague found it challenging to complete the process flow, since MURAL is not as user-friendly as Visio, (2) the colleague did not understand the new concepts and did not appreciate the value of the modelling language and (3) MURAL would often refresh automatically causing distraction and time loss.

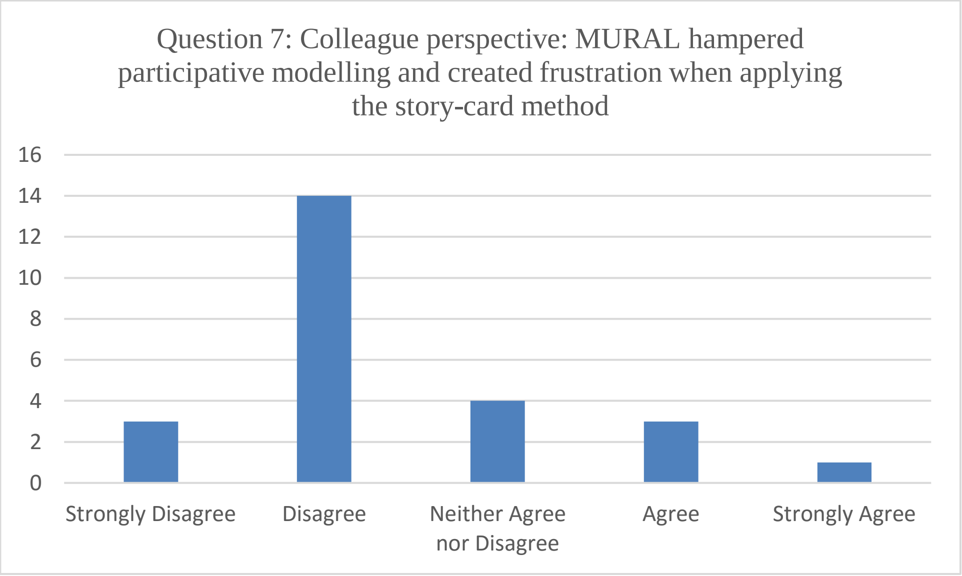


Figure 10: Participant Responses for Question 7

For *question 8* (see Figure 11), one participant provided additional motivation, indicating that many companies already have modelling tools that are more self-explanatory than MURAL.

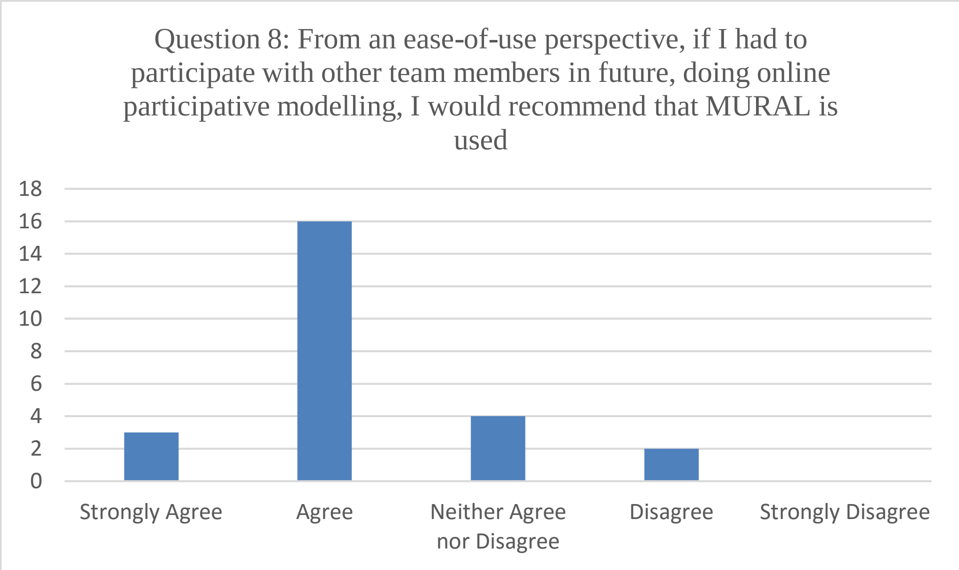


Figure 11: Participant Responses for Question 8

For *question 9* (see Figure 12), the participants who agreed indicated that (1) people engage better with face-to-face facilitation and the first session should be using a whiteboard followed by online examples, and (2) zooming in and out on the SCM caused frustration with navigating.

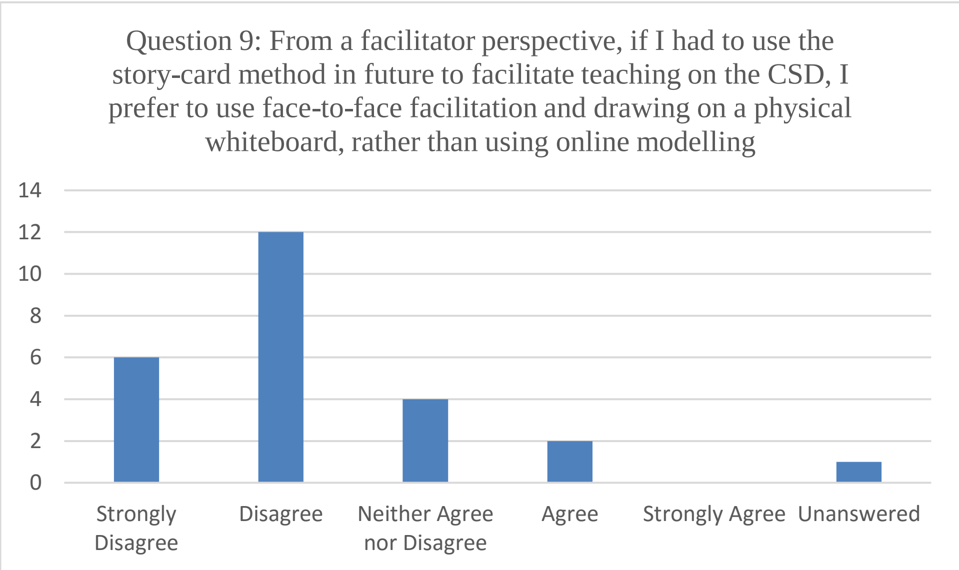


Figure 12: Participant Responses for Question 9

Question 10 was in the form of an open-ended question: *In terms of the MURAL tooling, you or your colleague may have experienced some frustrations related to the tool functionality. Please elaborate on these frustrations.*

The following themes emerged from the responses:

- The 'undo' function is not working properly, e.g. *"MURAL has issues tracking changes chronologically, for example at times we would use the undo function only to revert to a 'previous' instance that didn't actually exist"* and *"The undo function would also often not work".*
- Lack of auto-alignment, e.g. *"my colleague wanted to create a neat flow chart from the beginning but constantly had to re-do portions of the process flow".*
- Editing frustrations, e.g. not being able to *"edit the style of multiple text items simultaneously"* and *"In some cases one person had a block still selected while the other person was trying to edit the text".*
- Using an 'area' as a container for constructs created problems, e.g. *"not always being able to edit constructs inside an area"* and "*When adding an area, at times this would disorient my flow chart".*
- Automatic sticky-note addition with double-click, e.g. a double-click on an editable construct will add a sticky note construct, instead of editing the text of the editable construct.
- Auto-refresh problems, e.g. *"Some of the icons (Gateways and Timer) also did not show, it was just a grey block, I needed to refresh the page a couple of times before it returned to normal"* and *"MURAL would often refresh causing distraction and time loss".*
- Activity tracking issues, e.g. *"did not work properly"* and *"My activity tracking was cleared out".*
- Work loss due to connection problems, e.g. *"My colleague lost connection, but did not realize it until five tasks later (I noticed but I first thought it was from my side), and then he had to reconnect. When he reconnected, the tasks disappeared and he had to re-draw all of them again".*
- MURAL should have the ability to verbally communicate while modeling, e.g. *"It would be easier if MURAL itself came with the option to communicate verbally while collaborating instead of having to use WhatsApp."*

6.2.2 *Actual level of participation*

MURAL offers an activity tracking tool that was useful in determining whether participants engaged with a colleague, using the SCM as intended, where both participants had to apply the co-modelling instructions of the SCM.

For demonstration purposes, extracts of the activity logs of 2 randomly selected participants are provided in Figure 13 below. The activity log on the left-hand side demonstrates method *Steps 4* and *5*, whereas the activity log on the right-hand side demonstrates method *Step 13*. Note that the identities of the participants and colleagues (where necessary) are concealed using blue blocks.

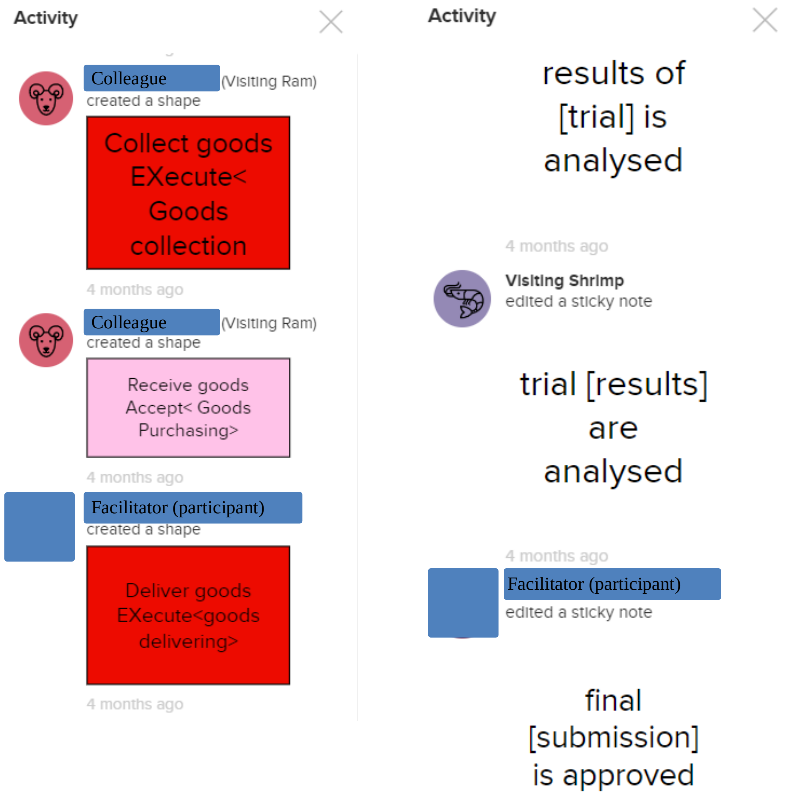


Figure 13: Activity Log Extracts from Participants

Two of the participants indicated that they have lost the detailed history of their interactive modelling activities. Although various possible reasons for this issue were investigated, e.g. the duration from creation date to last-editing date, connection problems as well as the total number of editing activities, none of these possibilities could be confirmed.

Furthermore, the 36 individual murals (also referred to as boards) were investigated to determine if the entire history of the mural (from creation date) reflected in MURAL, and found that only 21 of the murals were traceable from the creation date. A visual inspection of the activity history was required to determine whether participants actually followed the prescribed participation instructions that were indicated for some of the SCM steps. *Step 2, Step 3* and *Step 8* of the SCM requires explanation by the facilitator alone, whereas *Step 12* is a validation step that may not necessarily require additional modelling. The remaining steps include PM with the following dedicated modelling responsibilities:

- *Step 1:* Colleague maps out process
- *Step 4:* Both classify process tasks for original level
- *Step 5:* Both formulate alternative descriptions
- *Step 6:* Both analyse tasks for informational level
- *Step 7:* Facilitator duplicates red and light-red tasks
- *Step 9:* Both take turns to model initiator links
- *Step 10a:* Colleague duplicates green tasks
- *Step 10b:* Facilitator models access links
- *Step 11:* Facilitator models wait links
- *Step 13:* Both take turns in TPT

The results are summarised in Figure 14 below, indicating high participation levels, i.e. participants co-modelled according to the allocated responsibilities, for most of the SCM steps as provided in the list above.

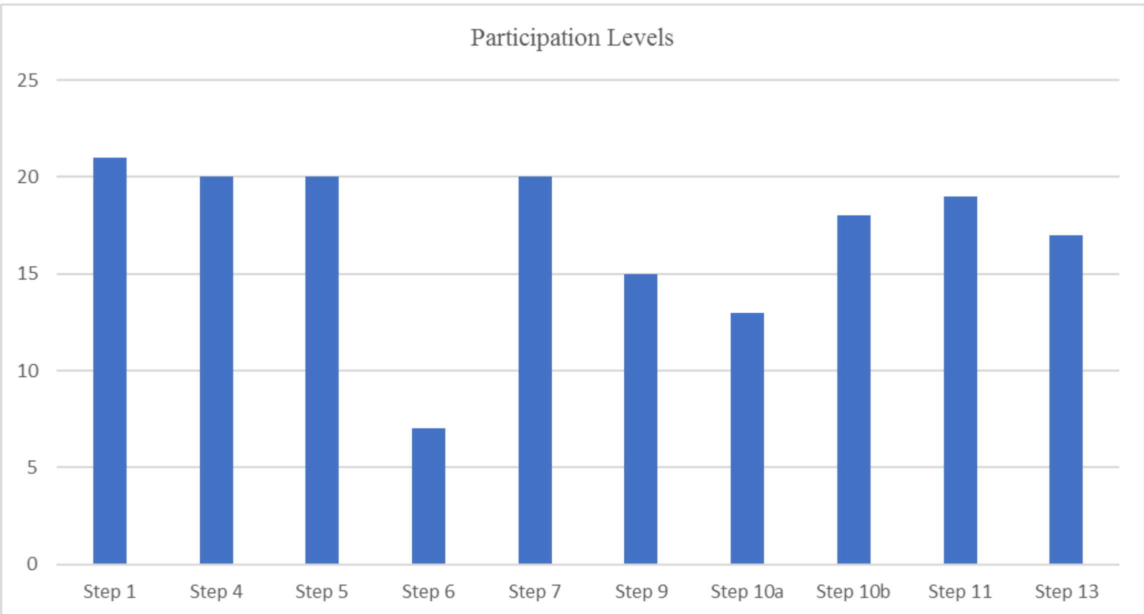


Figure 14: Activity Tracking Results to Indicate Participation Levels

However, 3 steps indicated lower participation and are highlighted as follows:

- *Step 6* required that both the facilitator and colleague identified informational level tasks. This step is problematic, since visual inspection indicated that the facilitator performed the identification, rather than the colleague.
- For *Step 9*, it was expected that the participation pairs would have difficulty in identifying initiation links. The results confirmed that the facilitator controlled this step, rather than the colleague.
- *Step 10a* had to be performed by the colleague, but for some participative sessions, the facilitator performed Step 10a. This may be due to the fact that the facilitator wanted to expedite the session.

7. Summary, Contribution, Limitations and Recommendations

The *summary, contribution, limitations* and *recommendations* of this study are provided below.

7.1 Summary

The SCM provides guidance when agile software development team members first need to obtain a common understanding of enterprise operations, using the CSD as a taxonomy to structure emerging requirements when supporting software application systems are developed. The development and evaluation of an online-adapted version of the SCM as presented in a study by De Vries (2022), indicated that the software modelling tool, Diagrams.net, used in combination with the SCM is inefficient in encouraging active participation during modelling due to its *latency* in reflecting updates performed by co-modellers.

This study started with a literature review, conducted in order to identify *concepts* frequently used within the software application development BOK. The literature review indicated that problems related to PM also exist within a broader context than this study and that other researchers have already experimented with PM tools, such as Microsoft Visio.

In addressing the tool inefficiency concern as mentioned above, DSR was applied to support the SCM by selecting a software modelling tool that promotes PM. A detailed tool comparison was provided, answering *secondary research question 1*, based on an iterative process that was used to conduct a *pilot* experiment with two short-listed tools, namely Miro and MURAL. The *pilot* experiment consisted of three main phases, namely *feature exploration*, *entry requirement identification* and *tool selection*. The third phase, *tool selection*, indicated that MURAL was selected as the solution.

MURAL allowed for the creation of an SCM template, answering secondary *research question 2*. The template was used by participants of this study during a *main* experiment, consisting of pre-designed symbols and method guidance. The pre-designed symbols included *Basic BPMN Symbols* and *DEMOSL 4+ CSD Symbols* in support of diagram construction, whereas the method guidance included the 13 method steps of the SCM, with modelling responsibilities (facilitator, colleague or both), to encourage PM. The SCM method steps were demonstrated, also answering *secondary research question 2*, prior to the *main* experiment to participants during an online session.

Secondary *research question 3* is discussed separately and in more detail in Section 7.2 below, as it relates to the contribution of this study.

7.2 Contribution

One of the contributions of this study is the tooling comparison, as mentioned above. The explorative approach followed, experimenting with two PM tools (i.e. Miro and MURAL) and identifying main features for these tools, could be useful when other researchers may need to compare a different set of tools within their own PM context. The approach could be used by other researchers as a *guideline*, also indicating how results could be quantified and qualified.

The main contribution of this study was to determine the *usefulness* of the PM tool. MURAL's *participative modelling abilities*, as well the *actual level of participation* in accordance with the dedicated SCM modelling responsibilities, were evaluated using a survey, answering secondary *research question 3*. The survey results indicated that participants mostly had a positive experience when they used MURAL in combination with the SCM i.e. that MURAL *encouraged participative modelling*. However, some participants also experienced frustrations with some of MURAL's functions, including: (1) the 'undo' function, (2) auto-alignment, (3) editing, (4) using the 'area' construct, (5) automatic sticky-note addition with double-click, (6) auto-refresh problems, (7) activity tracking not working properly, (8) work loss due to connection problems and (9) the inability to communicate verbally via MURAL.

Activity tracking results indicated that the *actual level of participation* was overall high for most of the SCM steps, except for *Step 6*, *Step 9* and *Step 10a.* The detailed analysis on the activity histories indicated that the facilitator took control of the modelling associated with these steps. This could be due to the facilitator wanting to expedite the session, as the duration of an SCM session could become extensive, or it could be possible that the facilitator found it difficult to impart knowledge on DEMO.

7.3 Limitations and recommendations

The SCM only facilitates a PM session between two participants i.e. a participant pair. Co-modelling an operating context within an agile software development team with five or more members may cause additional tooling related problems and frustrations (other than the 9 different frustrations mentioned above) to surface when using the PM tool for a larger group of participants. The hardware used by participants were not considered, and could have affected the overall modelling experience as well, especially regarding the output devices and the availability of a second monitor. Additional experimentation with a PM tool, incorporating larger participant groups and being more prescriptive in the hardware to be used, is thus recommended.

This study was based on a PM software tool, used to support the SCM, with specific SCM-related entry requirements. However, MURAL is not restricted to the combined use with the SCM. Various pre-designed MURAL templates are available that could be used within different contexts or environments. These templates can be used as-is, or tailored to meet the needs of a completely different set of PM requirements. Users can also design their own custom template (as done for the SCM), if there are no existing templates available that could be useful.

It is acknowledged that PM tools are continuously developing, and that there exist many PM software tools that were not considered for this study (due to certain minimum requirements). However, MURAL continuously improve their product, based on error-reporting and requirements from the end-user community, also communicating updates via their website (see link: https://www.mural.co/changelog).

To conclude, this dissertation indicates that sufficient evidence exists in support of the thesis statement of this study, namely:

> *Implementing the story card method steps within MURAL as participative modelling tool, supports the story card method.*

8. References

ARCHER L, B. 1964. Systematic method for designers. *Design,* 0**,** 56-59.

BASKERVILLE, R., BAIYERE, A., GREGOR, S., HEVNER, A. & ROSSI, M. 2018. Design science research contributions: Finding a balance between artifact and theory. *Journal of the Association for Information Systems,* 19**,** 358-376.

BRATTETEIG, T. & WAGNER, I. 2016. Unpacking the notion of participation in participatory design. *Computer Supported Cooperative Work (CSCW),* 25**,** 425-475.

BROWNE, D., KEMPF, R., HANSEN, A., O'NEAL, M. & YATES, W. 2013. Enabling Systems Modeling Language Authoring in a Collaborative Web-based Decision Support Tool. *Procedia Computer Science,* 16**,** 373-382.

CHENG, P.-H., FU, J.-M. & CHEN, L.-W. 2016. Knowledge transfer of software tool development for functional requirements analysis. *Computer Applications in Engineering Education,* 24**,** 131-143.

DE VRIES, M. Adapting and Evaluating the Story-Card-Method. Enterprise Engineering Working Conference, 2022. Springer, 74-94.

DE VRIES, M. & TH IFIP WG 8.1 CONFERENCE ON THE PRACTICE OF ENTERPRISE MODELING, P. T. 2018. DEMO and the story-card method: Requirements elicitation for agile software development at scale. *Lecture Notes in Business Information Processing,* 335**,** 138-153.

DIETZ, J. L. G. & MULDER, H. B. F. 2020. Enterprise ontology : a human-centric approach to understanding the essence of organisation. Cham: Springer.

FELLMANN, M., SANDKUHL, K., GUTSCHMIDT, A. & POPPE, M. Structuring Participatory Enterprise Modelling Sessions. IFIP Working Conference on The Practice of Enterprise Modeling, 2020. Springer, 58-72.

FRANZAGO, M., DI RUSCIO, D., MALAVOLTA, I. & MUCCINI, H. 2017. Collaborative model-driven software engineering: a classification framework and a research map. *IEEE Transactions on Software Engineering,* 44**,** 1146-1175.

GALLARDO, J., BRAVO, C. & REDONDO, M. A. 2012. A model-driven development method for collaborative modeling tools. *Journal of Network and Computer Applications,* 35**,** 1086-1105.

GALLARDO, J., BRAVO, C., REDONDO, M. A. & DE LARA, J. 2013. Modeling collaboration protocols for collaborative modeling tools: Experiences and applications. *Journal of Visual Languages & Computing,* 24**,** 10-23.

GAROUSI, V., FELDERER, M. & MÄNTYLA, M. V. 2019. Guidelines for including grey literature and conducting multivocal literature reviews in software engineering. *Information and Software Technology,* 106**,** 101-121.

GENARO MOTTI, V., RAGGETT, D., VAN CAUWELAERT, S. & VANDERDONCKT, J. Simplifying the development of cross-platform web user interfaces by collaborative model-based design. Proceedings of the 31st ACM international conference on Design of communication, 2013. 55-64.

GUTSCHMIDT, A. 2021. AN EXPLORATORY COMPARISON OF TOOLS FOR REMOTE COLLABORATIVE AND PARTICIPATORY ENTERPRISE MODELING.

GUTSCHMIDT, A., LANTOW, B., HELLMANZIK, B., RAMFORTH, B., WIESE, M. & MARTINS, E. Participatory Modeling from a Stakeholder Perspective: On the Influence of Collaboration and Revisions on Psychological Ownership and Perceived Model Quality. 2021 Cham. Springer International Publishing, 181-195.

GUTSCHMIDT, A., SAUER, V., SANDKUHL, K., KASHEVNIK, A. & TH IFIP WG 8.1 CONFERENCE ON THE PRACTICE OF ENTERPRISE MODELING, P. T. 2019. Identifying HCI patterns for the support of participatory enterprise modeling on multi-touch tables. *Lecture Notes in Business Information Processing,* 369**,** 118-133.

HANSEN, P., FOURIE, I. & MEYER, A. 2021. Third Space, Information Sharing, and Participatory Design. *Synthesis Lectures on Information Concepts, Retrieval, and Services,* 1**,** i-134.

HEVNER, A. R., MARCH, S. T., PARK, J. & RAM, S. 2004. DESIGN SCIENCE IN INFORMATION SYSTEMS RESEARCH 1. *MIS Quarterly,* 28**,** 75-105.

KARAGIANNIS, D., LEE, M., HINKELMANN, K. & UTZ, W. 2022. Domain-specific conceptual modeling : concepts, methods and ADOxx tools. Cham, Switzerland: Springer.

KITCHENHAM, B. & CHARTERS, S. 2007. Guidelines for performing Systematic Literature Reviews in Software Engineering. 2.

LEE, G. & XIA, W. 2010. Toward Agile: An Integrated Analysis of Quantitative and Qualitative Field Data on Software Development Agility. *MIS Quarterly,* 34**,** 87-114.

OFFERMANN, P., BLOM, S., SCHÖNHERR, M. & BUB, U. Artifact types in information systems design science–a literature review. International Conference on Design Science Research in Information Systems, 2010. Springer, 77-92.

OKOLI, C. 2015. A Guide to Conducting a Standalone Systematic Literature Review. *Communications of the Association for Information Systems,* 37.

PATEL, H., PETTITT, M. & WILSON, J. R. 2012. Factors of collaborative working: A framework for a collaboration model. *Applied Ergonomics,* 43**,** 1-26.

PEFFERS, K., TUUNANEN, T. & NIEHAVES, B. R. 2018. Design science research genres: introduction to the special issue on exemplars and criteria for applicable design science research. *European Journal of Information Systems,* 27**,** 129-139.

PEFFERS, K., TUUNANEN, T., ROTHENBERGER, M. A. & CHATTERJEE, S. 2007. A design science research methodology for information systems research. *Journal of management information systems,* 24**,** 45-77.

PERSSON, A. 2000. Investigating the influence of situational factors on participative enterprise modelling: making a case for a qualitative research approach.

POLANČIČ, G. & JOŠT, G. 2016. The impact of the representatives of three types of process modeling tools on modeler's perceptions and performance. *Journal of Software: Evolution and Process,* 28**,** 27-56.

RAHARJANA, I. K., SIAHAAN, D. & FATICHAH, C. 2021. User stories and natural language processing: A systematic literature review. *IEEE Access,* 9**,** 53811-53826.

RAMDHANI, M., MAYLAWATI, D., AMIN, A. & AULAWI, H. 2018. Requirements Elicitation in Software Engineering. *International Journal of Engineering and Technology(UAE),* 7**,** 772-775.

RENGER, M., KOLFSCHOTEN, G. L. & DE VREEDE, G.-J. 2008. Challenges in collaborative modelling: a literature review and research agenda. *International Journal of Simulation and Process Modelling,* 4**,** 248-263.

SANDKUHL, K., FILL, H.-G., HOPPENBROUWERS, S., KROGSTIE, J., MATTHES, F., OPDAHL, A., SCHWABE, G., ULUDAG, Ö. & WINTER, R. 2018. From expert discipline to common practice: a vision and research agenda for extending the reach of enterprise modeling. *Business & Information Systems Engineering,* 60**,** 69-80.

SCARIOT, C. A., HEEMANN, A. & PADOVANI, S. 2012. Understanding the collaborative-participatory design. *Work,* 41**,** 2701-2705.

SERÇE, F. C., SWIGGER, K., ALPASLAN, F. N., BRAZILE, R., DAFOULAS, G., LOPEZ, V. & THIRD INTERNATIONAL COGNITIVE LOAD THEORY CONFERENCE OPEN UNIVERSITY OF THE, N. 2011. Online collaboration: Collaborative behavior patterns and factors affecting globally distributed team performance. *Computers in Human Behavior,* 27**,** 490-503.

SIMONSEN, J. & ROBERTSON, T. 2013. *Routledge international handbook of participatory design*, Routledge New York.

SMUTS, H., WINTER, R., GERBER, A. & VAN DER MERWE, A. "Designing" Design Science Research–A Taxonomy for Supporting Study Design Decisions. International Conference on Design Science Research in Information Systems and Technology, 2022. Springer, 483-495.

SONNENBERG, C., VOM BROCKE, J. & EUROPEAN DESIGN SCIENCE SYMPOSIUM ON PRACTICAL ASPECTS OF DESIGN SCIENCE, E. L. I. R. L. 2012. Evaluation patterns for design science research artefacts. *Communications in Computer and Information Science,* 286 CCIS**,** 71-83.

STIRNA, J. Participatory Enterprise Modelling with the 4EM Method. NEMO, 2021 Stockholm University.

STIRNA, J. & KIRIKOVA, M. Integrating agile modeling with participative enterprise modeling. The proceedings of the CAiSE workshop EMMSAD, 2008. Citeseer, 171-184.

STIRNA, J., PERSSON, A. & SANDKUHL, K. Participative enterprise modeling: experiences and recommendations. International conference on advanced information systems engineering, 2007. Springer, 546-560.

STIRNA, J. & PERSSON, A. 2018. Enterprise Modeling : Facilitating the Process and the People. Stockholms universitet, Institutionen för data- och systemvetenskap.

VAN DER MERWE, A., GERBER, A. & SMUTS, H. 2017. Mapping a Design Science Research Cycle to the Postgraduate Research Report. *ICT Education : 46th Annual Conference of the Southern African Computer Lecturers' Association, SACLA 2017, Magaliesburg, South Africa, July 3-5, 2017, Revised Selected Papers.* Cham : Springer International Publishing : Springer.

VENABLE, J. R. Design science research post Hevner et al.: Criteria, standards, guidelines, and expectations. International Conference on Design Science Research in Information Systems, 2010. Springer, 109-123.

VOINOV, A., JENNI, K., GRAY, S., KOLAGANI, N., GLYNN, P. D., BOMMEL, P., PRELL, C., ZELLNER, M., PAOLISSO, M., JORDAN, R., STERLING, E., SCHMITT OLABISI, L., GIABBANELLI, P. J., SUN, Z., LE PAGE, C., ELSAWAH, S., BENDOR, T. K., HUBACEK, K., LAURSEN, B. K., JETTER, A., BASCO-CARRERA, L., SINGER, A., YOUNG, L., BRUNACINI, J. & SMAJGL, A. 2018. Tools and methods in participatory modeling: Selecting the right tool for the job. *Environmental Modelling and Software,* 109**,** 232-255.

WHITE, S. A. 2004. Introduction to BPMN. *Ibm Cooperation,* 2**,** 0.

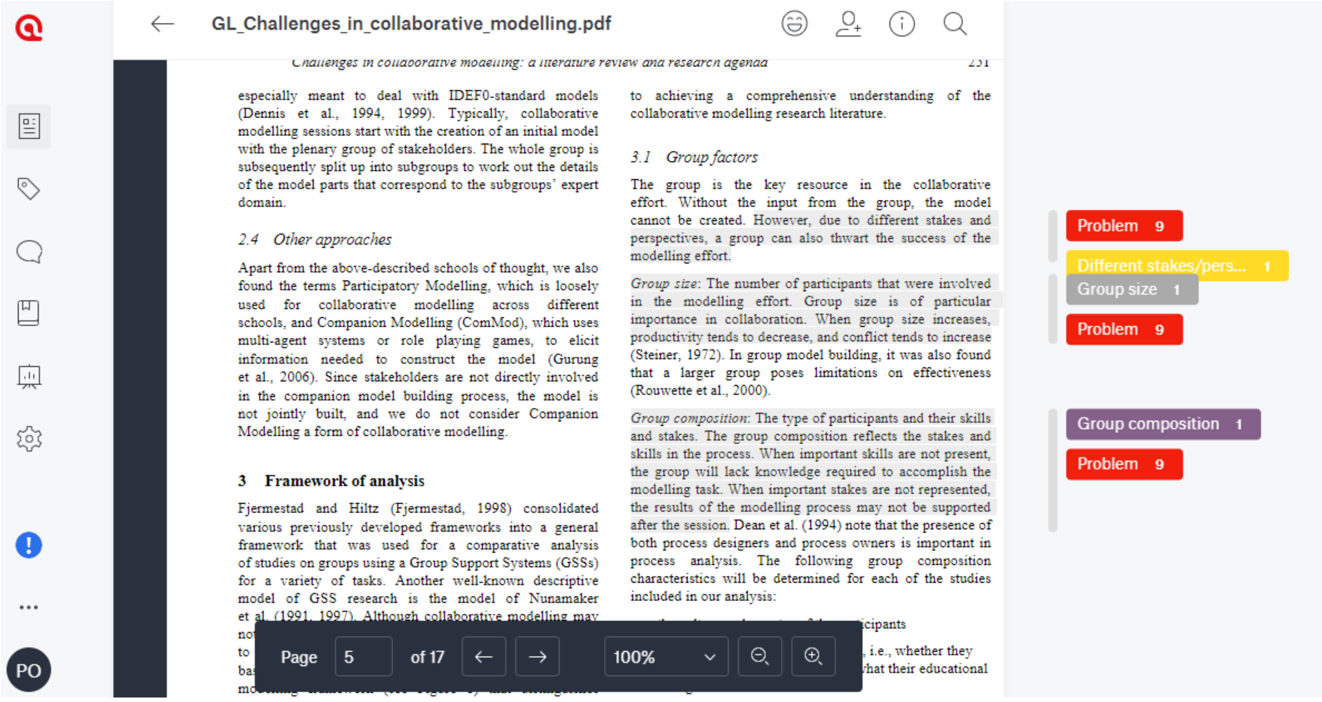
WOHLIN, C. & AURUM, A. K. 2015. Towards a decision-making structure for selecting a research design in empirical software engineering. *Empirical Software Engineering : An International Journal,* 20**,** 1427-1455.

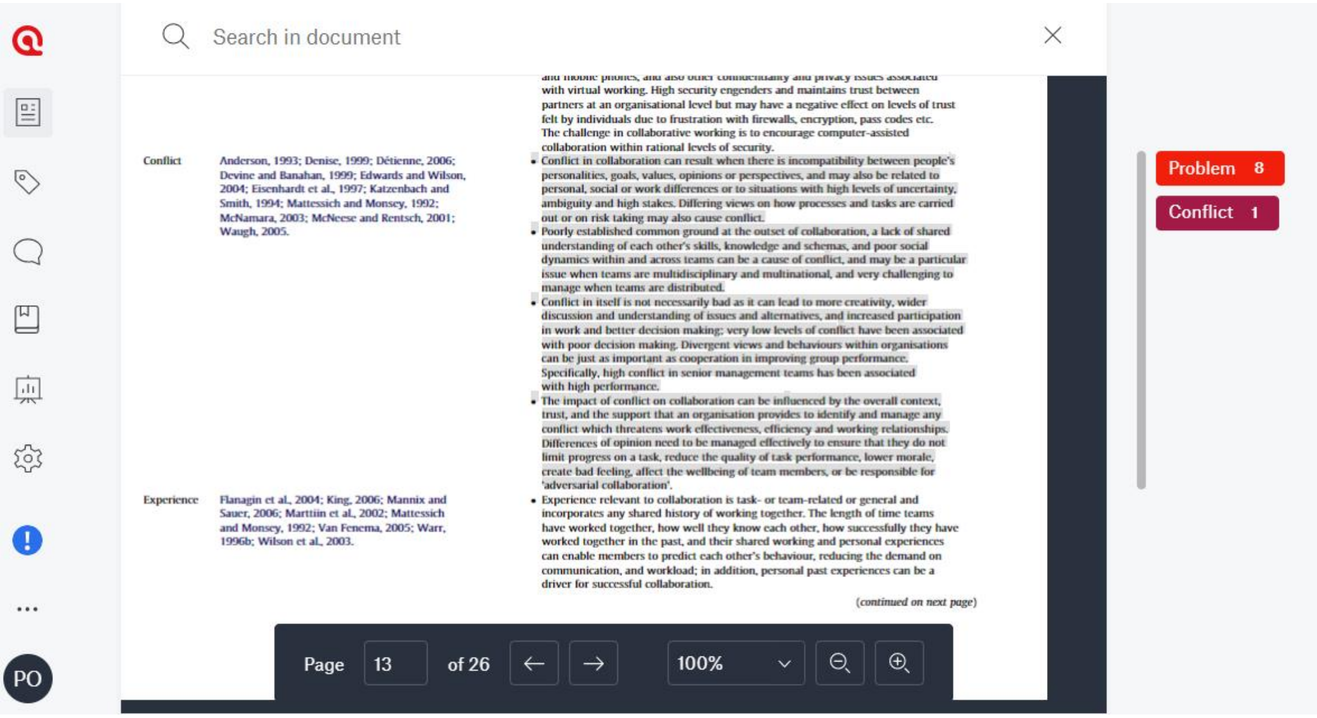
WOHLIN, C., TH INTERNATIONAL CONFERENCE ON, E. & ASSESSMENT IN SOFTWARE ENGINEERING, E. L. G. B. R. 2014. Guidelines for snowballing in systematic literature studies and a replication in software engineering. *ACM International Conference Proceeding Series*.

YOUNAS, M., JAWAWI, D. N. A., GHANI, I., SHAH, M. A., KHURSHID, M. M. & MADNI, S. H. H. 2019. Framework for Agile Development Using Cloud Computing: A Survey. *Arabian Journal for Science and Engineering,* 44**,** 8989-9005.

Appendix A: Evidence of Class-of-Problem Coding

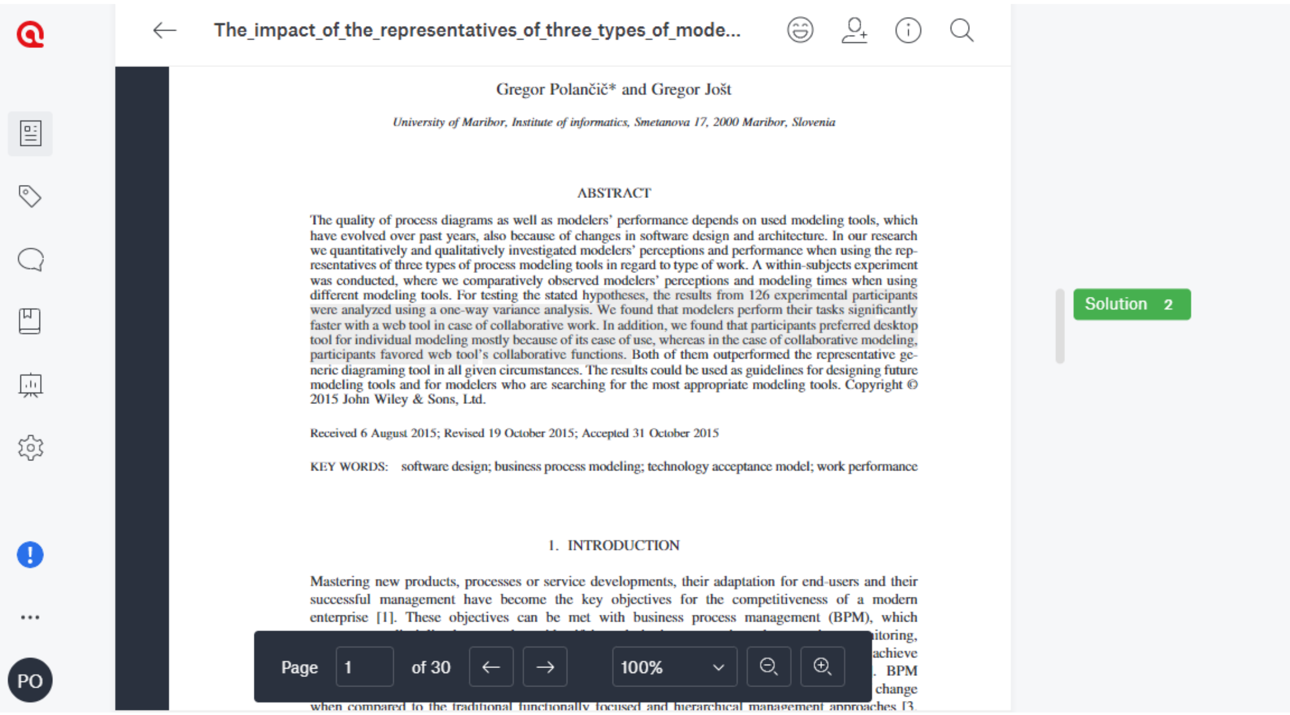
The following illustrations provide evidence of coding, using the online version of ATLAS.ti.





Appendix B: Evidence of Solution Area Coding

The following illustrations provide evidence of coding, using the online version of ATLAS.ti.



Appendix C: Inter-coder Agreement for Codebook related to Class-of-Problems

This appendix provides the inter-coder agreement that was conducted to derive at the overall percentage agreement when using the codebook as specified in the main report (see Section 0).

Code	**Coder A** (Renger et al., 2008)	**Coder B** (Renger et al., 2008)
Theme: **Collaborative modelling**	"..we define collaborative modelling as: The joint creation of a shared graphical representation of a system."	"For the purpose of the research presented in this paper, we define collaborative modelling as: The joint creation of a shared graphical representation of a system."
Problem: **Group size**	"When group size increases, productivity tends to decrease, and conflict tends to increase."	"Group size: The number of participants that were involved in the modelling effort. Group size is of particular importance in collaboration. When group size increases, productivity tends to decrease, and conflict tends to increase."
Problem: **Group composition**	"The group composition reflects the stakes and skills in the process. When important skills are not present, the group will lack knowledge required to accomplish the modelling task. When important stakes are not represented, the results of the modelling process may not be supported after the session."	"Group composition: The type of participants and their skills and stakes. The group composition reflects the stakes and skills in the process. When important skills are not present, the group will lack knowledge required to accomplish the modelling task. When important stakes are not represented, the results of the modelling process may not be supported after the session."
Problem: **Time and cost**	"Collaborative modelling efforts take a significant amount of time, and are therewith costly."	"Collaborative modelling efforts take a significant amount of time, and are therewith costly."
Problem: **Model quality**	"When comparing the different collaborative modelling studies, we found challenges and lessons learned on the following topics: ….. • the model quality, both from an objective and a subjective perspective"	"When comparing the different collaborative modelling studies, we found challenges and lessons learned on the following topics: • the roles and group composition • the interactive process, collaboration and participation • the modelling approach, activities and modelling rules to support the modelling effort • the model quality, both from an objective and a subjective perspective"
Problem: **Productivity**	Not coded.	"…productivity of particular collaborative modelling approaches"
Problem: **Different stakes and perspectives**	"..due to different stakes and perspectives, a group can also thwart the success of the modelling effort."	"However, due to different stakes and perspectives, a group can also thwart the success of the modelling effort."

Total number of codes in agreement: 6
Total number of codes in disagreement: 1
Total number of codes: 7
Overall % Agreement = (Total number of codes agreed / Total number of codes) * 100 ≈ 86%

Appendix D: Inter-coder Agreement for Codebook related to Solution Areas

This appendix provides the inter-coder agreement that was conducted to derive at the overall percentage agreement when using the codebook as specified in the main report (see Section 0).

Code	**Coder A -** Stirna (2021)	**Coder B** – Stirna (2021)
Solution Theme: **Participatory Enterprise Modelling**	"Participatory Enterprise Modelling"	"Participatory Enterprise Modelling"
Solution Theme: **For Enterprise Modelling Framework**	"For Enterprise Modelling framework (4EM)"	"For Enterprise Modelling framework (4EM)" "The 4EM method consists of three core elements: a defined procedure to modelling using a fixed notation, performance of enterprise modelling in the form of a project with predetermined roles and a participatory process to involve enterprise stakeholders and domain experts"
Solution Area: **Group composition**	"EM application in practice is usually supported by computerized tools"	"EM application in practice is usually supported by computerized tools"
Solution Tool: **Miro**	"Efforts on tools supporting 4EM – Miro.com and diagrams.net"	"Efforts on tools supporting 4EM – Miro.com …"
Solution Tool: **Diagrams.net**	"Efforts on tools supporting 4EM – Miro.com and diagrams.net"	"Efforts on tools supporting 4EM – … diagrams.net"

Total number of codes in agreement: 5
Total number of codes in disagreement: 0
Total number of codes: 5
Overall % Agreement = (Total number of codes agreed / Total number of codes) * 100 = 100%