

Transitioning from motivational goal models to user stories within user-centred software design

Eduardo A. Oliveira
Computing and Information Systems
University of Melbourne
Melbourne, Australia
eduardo.oliveira@unimelb.edu.au

Varsha Maram
Computing and Information Systems
University of Melbourne
Melbourne, Australia
v.maram@unimelb.edu.au

Leon Sterling
Centre for Design Innovation
Swinburne University of Technology
Melbourne, Australia
lsterling@swin.edu.au

Abstract—Motivational goal modelling has evolved from agent-oriented models to allow a shared understanding of a project by diverse stakeholders. Building a motivational model is in the spirit of user-centred design. Requirements artefacts such as user stories and personas should be developed consistently with the model. This paper describes a method to generate user stories from motivational models. The generated stories are checked by users and developers to ensure readability and clarity. The method has been partially automated within an extension to an editing tool.

Index Terms—model-based requirements, motivational modelling, requirements engineering, engineering education

I. INTRODUCTION

Motivational goal models are valuable for building a shared understanding of a project. Agent-oriented goal models were described in Sterling and Taveter [1] as a way of designing and implementing agent-oriented systems, however, they did not focus on Requirements Engineering (RE) methods. Since then, the do/be/feel elicitation method [2] incorporated the use of motivational goal models to RE.

The do/be/feel method produces several lists of project goals and stakeholders. The lists can be translated into motivational goal models. These lists and models have been used extensively for teaching at the University of Melbourne and at Swinburne University of Technology as they help students to understand a problem or desired system that is accessible to all stakeholders [1].

The success of software products is highly dependent on validations performed on users' goals. In this context, the relationships between goal models, personas and user stories are critical at early design phases [3]. Non-technical approaches like motivational goal models can be used to facilitate communication and better overall understanding among stakeholders when combined appropriately. The hierarchical diagram of the goals of a system at high abstraction levels has potential to improve communication between students and industry partners during requirements elicitation and validation tasks. Motivational goal models have evolved and been used extensively over the past 5 years in software engineering, especially in RE.

In this paper, we extend the work of [1] and [2] and present a method for semi-automatically generating user stories from motivational goal models to support requirements validation

between software engineers and clients. The method has the advantage of extending the high-level understanding as conveyed by the motivational model with concrete artefacts. As such, they are also consistent with user-centred software design.

The remainder of the paper is organized as follows: Section 2 briefly discusses major challenges related to requirements engineering and the role of motivational models to support the creation of personas and user stories. Section 3 provides a comprehensive example on how to generate user stories from motivational models. Section 4 provides conclusions and highlights future work opportunities.

II. BACKGROUND AND LITERATURE REVIEW

In the past few years, previous studies have identified that face-to-face communication, customer involvement and interaction are major challenges related to Requirements Engineering (RE) as they often involve different stakeholders with different backgrounds [4]–[7]. Agile processes advocate minimal documentation [8] in the form of user stories and discourage long and complex specification documents. Frequent face-to-face communication helps clients steer the project in an unexpected direction according to their own understanding of the project. The frequent meetings lead to informal communication among stakeholders, which aids in the evolution of the requirements. The frequency of communication depends on the availability and willingness of team members. Customers may be accustomed to traditional methods and are unable to comprehend and trust agile methods. [6] discusses risks when customers' IT groups and managers, for example, understand requirements in different ways and how their inability to comprehend agile methods can impact decision-making and project execution. Without a common understanding of requirements, many degrees of formal separation between stakeholders can be created. The separation can lead to biased and incomplete views that diminish transparency in eliciting, negotiating, and validating requirements. A lack of ability to communicate properly can compromise the whole development of a system. There is an urgent need to ensure that requirements are appropriately defined, clarified and prioritised among different stakeholders [9].

In this context, personas and user stories, typically structured into epics, can be created as specifications of requirements. An epic is a large user story that cannot be delivered as defined within a single iteration or is large enough that it can be split into smaller user stories [10]. A user story description generally consists of, among other things, a statement. To be amenable to humans as well as to machines, a user story statement should relate to both a persona and a goal [11]. A persona is a description of an archetypical user of a software system, created based on research performed with potential real users of a software system. A goal is an intended outcome of a persona interacting with a software system. A role is an abstraction of a persona. Personas provide the rationale for the existence of user stories. The success of final software products is highly dependent on validations performed on users' goals. The relationships between personas and goals are critical.

Personas and user stories facilitate communication and better overall understanding among stakeholders [9], [12], [13]. Both artefacts shift the concentration from written documentation to communication. User stories are also believed to be capable of eradicating the challenge of constant updating of requirements specification documents in traditional requirements engineering [14] by keeping team members updated. User stories emphasise “user goals” and are generally validated against created personas. The use of personas and user stories briefly explain the user perception, focus on “what” is needed to be done, and support collaborative and iterative development. User stories are usefully structured into epics which is a collection of related user stories.

In recent years, interest in personas and in the use of user stories has extended to the software engineering communities. Previous studies have identified and discussed the importance of personas and user stories in RE [13], [15]–[18]. Surprisingly, however, despite several proposals for integrating personas and user stories methodologically with software en-

gineering [11], [19], there has been little work describing how software engineering tools should be augmented to support their creation, usage, and on-going maintenance. To date, there has been little work or tools supporting the integration, consistency and validation between personas, user stories and requirements engineering activities. According to [16], the focus of software engineering tools has been to support the design and development of software, not user-centered design artifacts like personas or user stories. Therefore it is necessary for us to understand how personas and user stories can be integrated into software tools to ensure they help, rather than hinder, software engineering practice.

Recently we have used motivational goal models to effectively bridge requirements artifacts such as system goals, personas and user stories during the requirements elicitation and design phases of software engineering subjects at the University of Melbourne. They improve communication between students and industry partners. To our knowledge, no motivational model tool has supported an automated extraction of user stories as part of their systems before. This paper contributes to the field by presenting a method for semi-automatically generating user stories from motivational goal models to support readability and clarity in RE, and requirements validation between software engineers and clients.

III. MOTIVATIONAL MODELS AND USER STORIES

Motivational modelling emerged from agent-oriented goal modelling to describe projects to non-technical stakeholders. It has been applied in diverse contexts, including space planning, brand development and strategic planning for an eCommerce platform. The essence is developing a shared understanding between diverse stakeholders in a user-centred manner. The general abstract nature of the models has been useful for software engineering students who sometimes struggle to get a high-level view of a project.

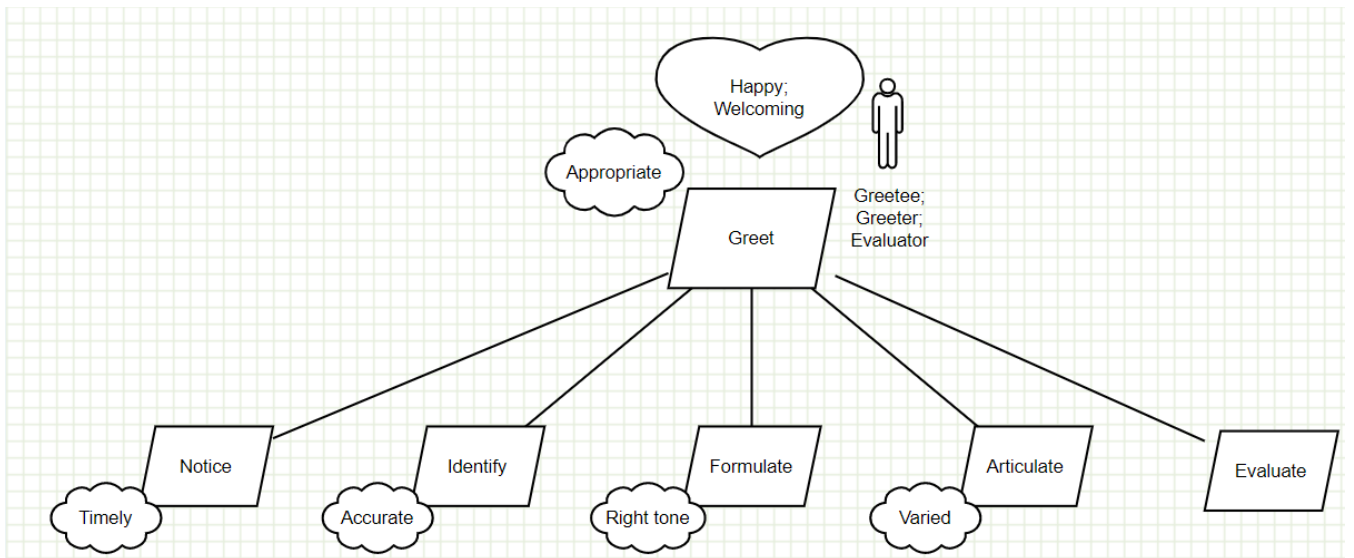


Fig. 1. Greeting goal model

It is beyond the scope of the paper to discuss motivational modelling in detail. Rather this paper has a narrow scope of showing how to generate user stories from motivational models. We illustrate the relationship between user stories and motivational models with two examples in which the complexity of the models and relationships increase with each example.

We begin our discussion with a very simple 'low-level' model taken from the book by Sterling and Taveter [1]. The example we choose is the 'Hello World' of agents, an agent greeting. As this is a simple example, we believe it is easier for us to explain the links between the goal model and user stories, without the need to provide additional details about the problem domain. A goal model for two agents greeting is given in Section 4.6 of 'The Art of Agent-Oriented Modelling' on p. 139. It is reproduced in Figure 1 with some slight modification of wording and the addition of an emotional goal. The model is a high level representation of a greeting scenario. The greeter needs to greet the gretee by a sequence of steps including noticing there is someone to be greeted in a timely manner, identifying the person accurately, formulating an appropriate greeting, and articulating it. In the spirit of quality control, there is an evaluator role to evaluate the greeting. To briefly explain the notation, functional goals are depicted in parallelograms, quality goals are depicted in clouds, emotional goals in hearts, and roles by stick figures. This notation is used consistently throughout the paper. Note the diagram was re-drawn with our editor tool available at motivationalmodelling.com.

The idea for converting from motivational model to user stories is to consider the motivational model as a tree and to generate a user story for each leaf. The basic form of a

user story is 'As a <user> I want to <do> so that <goal>'. Interpreting the roles as users gives an initial attempt to generate user stories. We have extended the motivational modelling tool with an experimental feature that generates user stories automatically. Applying the feature to the model in Figure 1 results in five user stories as listed below.

- 1) As a Greeter, Greeter or Evaluator, I want to be able to Notice
- 2) As a Greeter, Greeter or Evaluator, I want to be able to Identify
- 3) As a Greeter, Greeter or Evaluator, I want to be able to Formulate
- 4) As a Greeter, Greeter or Evaluator, I want to be able to Articulate
- 5) As a Greeter, Greeter or Evaluator, I want to be able to Evaluate

The five user stories do not make complete sense as generated by the tool. For instance, in this example, there should be a separate user story for each different role. Later in this paper that will not be the case. So user story 1 would be better expanded to be three separate user stories. Here is possible wording.

- As a Greeter, I want to be able to Notice the Greeter
- As a Greeter, I want to be able to be Noticed by the Greeter
- As an Evaluator, I want to evaluate whether the Greeter greets the Greeter appropriately

It is an exercise for the reader to expand the other user stories.

We now consider quality and emotional goals. The functional goal 'Notice' has the quality goal 'Timely' attached.

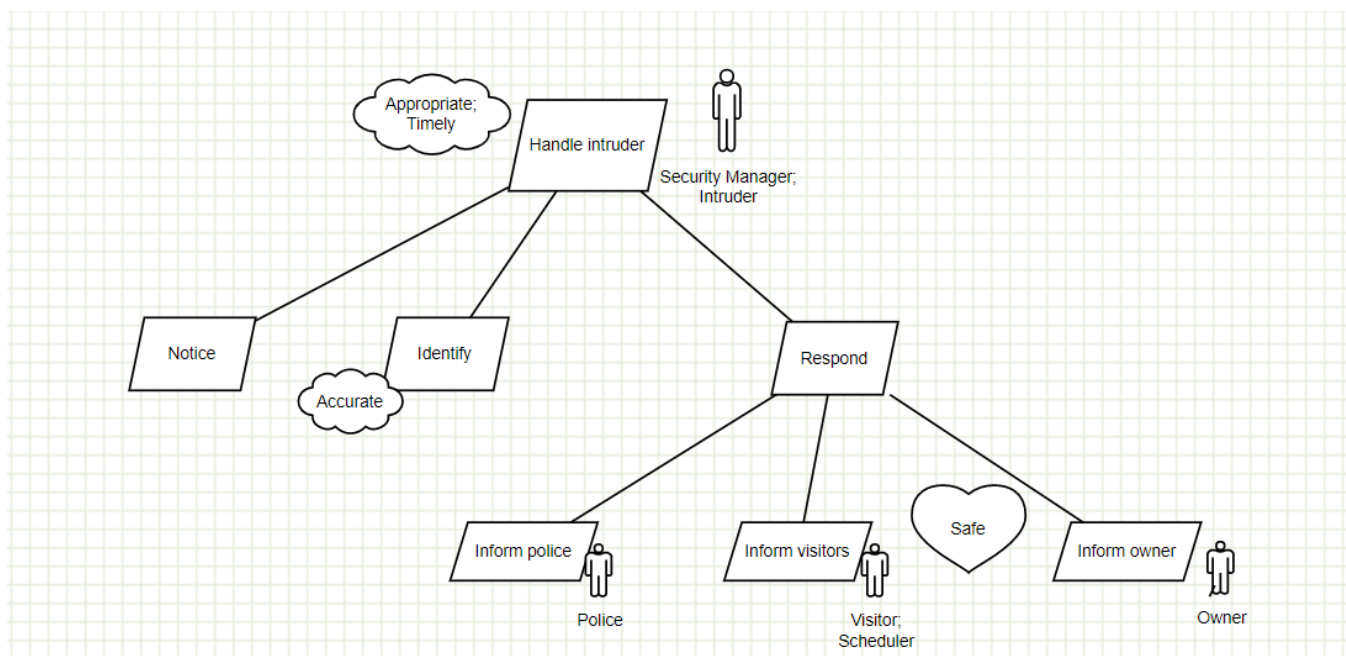


Fig. 2. Intruder handling goal model

An improved user story would be 'As a Greeter, I want to be able to Notice the Greetee in a timely manner.'

An improved version of the second user story might be 'As a Greeter, I want to be able to Identify the Greetee accurately.' Note that the quality goal 'Accurate' expressed as an adjective in the model reads better as an adverb 'Accurately' in the user story. In general, adjustments should be made to improve readability, and the translation process should be semi-automated rather than automated completely.

We now consider the intruder scenario discussed in Chapter 9 of 'The Art of Agent-Oriented Modelling.' Consider Figure 9.4 from the book modified slightly in Figure 2.

Note that the motivational model can be naturally divided into two subtrees. We introduce another translation principle. Significant subtrees should be handled as separate epics. The name of the epic will typically be a node from the motivational model. The experimental feature generates the following.

1) Epic: Handle intruder

- As a Security Manager or Intruder, I want to be able to Notice
- As a Security Manager or Intruder, I want to be able to Identify

2) Epic: Respond

- As a Security Manager, Intruder or Police, I want to be able to Inform police
- As a Security Manager, Intruder, Visitor or Scheduler, I want to be able to Inform visitors
- As a Security Manager, Intruder or Owner, I want to be able to Inform owner

A better version separates the roles. Here is an updated version with greater detail about the scenario.

1) Epic: Handle intruder

- As a Security Manager, I want to Notice the Intruder

- As an Intruder, I do not want to be Noticed by the User Manager
- As a Security Manager, I want to Identify the Intruder
- As an Intruder, I do not want to be Identified

2) Epic: Respond

- As an Intruder, I do not want police to Respond
- As a Security Manager, I want to Inform police
- As a Police, I want to be Informed
- As a Security Manager, I want to Inform visitors
- As a Visitor, I want to be Informed
- As a Scheduler, I want to be Informed
- As a Security Manager, I want to Inform owner
- As an Owner, I want to be Informed

It is straightforward to add the quality and emotional goals. While simple, these two examples are indicative, we believe, of how a model can be used to generate user stories. The user stories will be followed through in design and implementation to make sure the requirements are met.

A. Software Project Example: A Goal Model for our Motivational Modelling Editor Tool

We now consider a larger example which has emerged from our experience in using motivational modelling in our teaching, research, and consulting activities. For reasons that are beyond the scope of the paper, we developed an initial motivational model of our overall activities around motivational modelling. Part of the model was used to direct teams of University of Melbourne students who have been maintaining and extending the motivational modelling editing tool in software project subjects.

The overall model is presented in Figure 3. Rather than explaining the model directly, we present user stories that could be derived from the model. Note that the effort to think

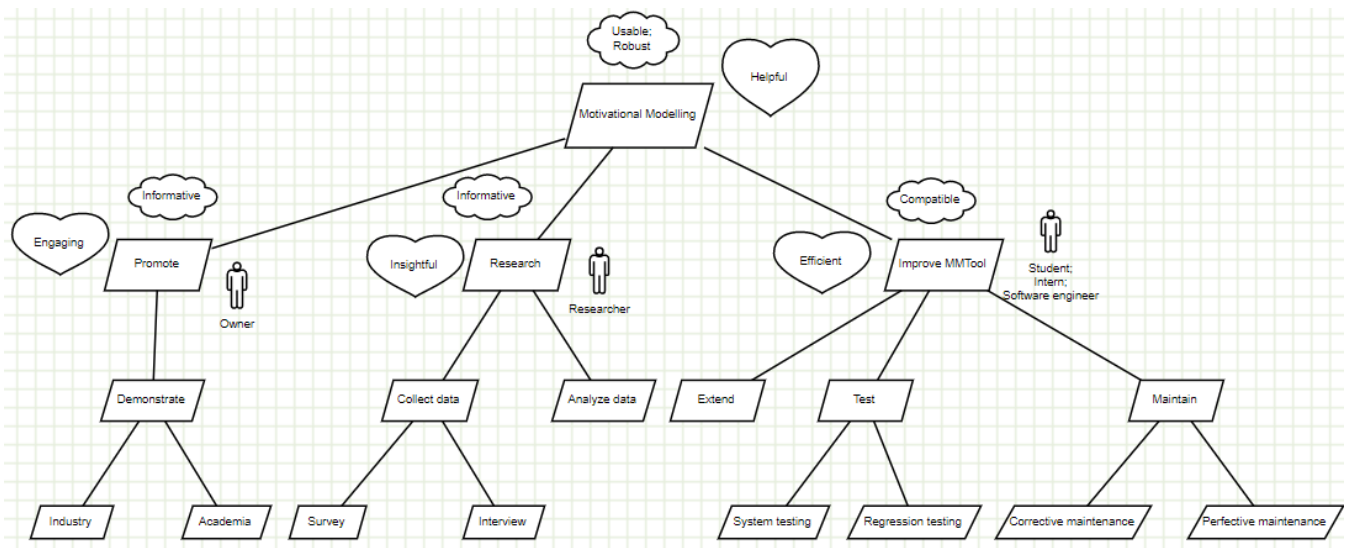


Fig. 3. Motivational Model goal model

about user stories related to the model improved the model, which has been through several iterations, as is expected.

The model has three subtrees which will be the three major epics: Promote, Research and Improve MM Editor tool. We discuss each epic in turn. Promote is the simplest epic, containing two user stories, one concerning demonstration to industry, and one concerning demonstration to academia. Promotional activities to industry are a little different than promotional activities to academia, so it makes sense to separate the user stories. Further, the reporting on the promotional activities will be different. Ensuring that the user stories would be simple and direct affected how the motivational model was built.

We next briefly discuss the Research epic. We have generated three user stories in this epic. The user stories correspond to conducting surveys about the use of motivational modelling, conducting interviews with students, staff and other users, and analysing the data from the surveys and interviews. Again it was helpful to think about a simple form for the model and the user stories.

The third epic concerns the motivational modelling tool that is being used extensively for workshops and consulting, and also being used for internships and student projects at the University of Melbourne. The tool is also being maintained at work.motivationalmodelling.com outside the university sector. There are five user stories as to the software covering respectively: overall extension of the tool, regression testing, overall system testing, corrective maintenance (i.e. fixing bugs) and perfective maintenance, namely improving features. An example of the latter is automatic sizing of nodes to fit the text in the goals. An example of a tool extension is the ability to colour nodes. Both of those examples have been developed in student projects, and are in the process of being more consistently tested and deployed.

A list of the user stories discussed is given here. Note that these user stories are suitable to be placed on a Kanban board for agile development. In fact they have been. Clearly these user stories are reasonably abstract, but they provide a high level overview of the work being done by the various people involved with motivational modelling including the authors. As reasonably abstract, they capture the flavour of user centred development. They clearly can be refined further as developments occur.

- User Stories about Motivational Modelling
 - Epic: Promote
 - * As Owner, I want to be able to demonstrate to industry
 - * As Owner, I want to be able to demonstrate to academia
 - Epic: Research
 - * As a Researcher, I want to be able to collect data by survey
 - * As a Researcher, I want to be able to collect data by interview
 - * As a Researcher, I want to be able to analyze data
 - Epic: Improve MMTool

- * As a Student, Intern or Software engineer, I want to be able to extend the MM tool
- * As a Student, Intern or Software engineer, I want to be able to test by system testing
- * As a Student, Intern or Software engineer, I want to be able to test by regression testing
- * As a Student, Intern or Software engineer, I want to be able to maintain by corrective maintenance
- * As a Student, Intern or Software engineer, I want to be able to maintain by perfective maintenance

IV. CONCLUSIONS

We have described a method to generate user stories from motivational models. The generated stories are checked by users and developers to ensure readability and clarity. Such checking retains the feeling of user centred development. The method discussed in this paper has been partially automated within an extension to the motivational modelling editing tool.

Future extensions being considered include the integration of a motivational modelling editing tool with Canvas LMS. Users will also be able to upload personas, user stories and epics (text) to the tool, which will convert them into a goal model. Future studies should focus on further examining the impact of user stories generated from goal models on requirements quality and on communication between stakeholders.

ACKNOWLEDGEMENTS

The authors would like to thank the teaching staff and students doing the Software Engineering subjects at the University of Melbourne who have contributed to the ideas of the paper. The work also acknowledges discussions with members of the Future Self and Design Living Lab at Swinburne University of Technology. The work was partially supported by ARC Discovery grant DP200102955, ‘Maturing design-led innovation processes with motivational models’.

REFERENCES

- [1] L. Sterling and K. Taveter, *The art of agent-oriented modeling*. MIT Press, 2009.
- [2] A. Lopez-Lorca, R. Burrows, and L. Sterling, “Teaching motivational models in agile requirements engineering,” in *Proceedings of the Requirements in Education and Training workshop at RE’18*, 2018.
- [3] E. Oliveira and L. Sterling, “Motivational models for validating agile requirements in software engineering subjects,” in *Proceedings of the 19th International Conference on Software Engineering Research and Practice, SERP’21, July 26-29, 2021, Las Vegas, Nevada, USA*, 2021.
- [4] E.-M. Schön, J. Thomaschewski, and M. J. Escalona, “Agile requirements engineering: A systematic literature review,” *Computer Standards & Interfaces*, vol. 49, pp. 79–91, 2017.
- [5] I. Inayat, S. S. Salim, S. Marczak, M. Daneva, and S. Shamshirband, “A systematic literature review on agile requirements engineering practices and challenges,” *Computers in human behavior*, vol. 51, pp. 915–929, 2015.
- [6] J. M. Bhat, M. Gupta, and S. N. Murthy, “Overcoming requirements engineering challenges: Lessons from offshore outsourcing,” *IEEE software*, vol. 23, no. 5, pp. 38–44, 2006.
- [7] E. A. Oliveira, “i-collaboration 3.0: um framework de apoio ao desenvolvimento de ambientes distribuídos de aprendizagem sensíveis ao contexto,” 2013.
- [8] L. Cao and B. Ramesh, “Agile requirements engineering practices: An empirical study,” *IEEE software*, vol. 25, no. 1, pp. 60–67, 2008.

- [9] M. Daneva, E. Van Der Veen, C. Amrit, S. Ghaisas, K. Sikkell, R. Kumar, N. Ajmeri, U. Ramteerthkar, and R. Wieringa, "Agile requirements prioritization in large-scale outsourced system projects: An empirical study," *Journal of systems and software*, vol. 86, no. 5, pp. 1333–1353, 2013.
- [10] M. Cohn, *User stories applied: For agile software development*. Addison-Wesley Professional, 2004.
- [11] P. Kamthan, "Using personas to support the goals in user stories," in *2015 12th International Conference on Information Technology-New Generations*. IEEE, 2015, pp. 770–770.
- [12] J. K. Blomkvist, J. Persson, and J. Åberg, "Communication through boundary objects in distributed agile teams," in *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, 2015, pp. 1875–1884.
- [13] T. Tenso, A. H. Norta, H. Rootsi, K. Taveter, and I. Vorontsova, "Enhancing requirements engineering in agile methodologies by agent-oriented goal models: Two empirical case studies," in *2017 IEEE 25th International Requirements Engineering Conference Workshops (REW)*. IEEE, 2017, pp. 268–275.
- [14] E. Bjarnason, K. Wnuk, and B. Regnell, "A case study on benefits and side-effects of agile practices in large-scale requirements engineering," in *proceedings of the 1st workshop on agile requirements engineering*, 2011, pp. 1–5.
- [15] A. Dittmar and P. Forbrig, "Integrating personas and use case models," in *IFIP Conference on Human-Computer Interaction*. Springer, 2019, pp. 666–686.
- [16] S. Faily and J. Lyle, "Guidelines for integrating personas into software engineering tools," in *Proceedings of the 5th ACM SIGCHI symposium on Engineering interactive computing systems*, 2013, pp. 69–74.
- [17] L. Schneidewind, S. Hörold, C. Mayas, H. Krömker, S. Falke, and T. Pucklitsch, "How personas support requirements engineering," in *2012 First International Workshop on Usability and Accessibility Focused Requirements Engineering (UsARE)*. IEEE, 2012, pp. 1–5.
- [18] B. Ferreira, W. Silva, E. Oliveira, and T. Conte, "Designing personas with empathy map," in *SEKE*, vol. 152, 2015.
- [19] J. W. Castro, S. T. Acuña, and N. Juristo, "Integrating the personas technique into the requirements analysis activity," in *2008 Mexican International Conference on Computer Science*. IEEE, 2008, pp. 104–112.