

A Metamodel for iStar-p: Requirements Prioritization with Goal Models

Maria Lencastre¹ [0000-0002-8032-8801] and João Pimentel² [0000-0002-7441-0796]

¹ Universidade de Pernambuco, Brazil

² Universidade Federal Rural de Pernambuco, Brazil
mlpm@ecom.poli.br, joao.hcpimentel@ufrpe.br

Abstract. Requirements prioritization is a key activity to assist in delivering the essential features within time-to-market constraints. When requirements priorities are specified as attributes in a list of requirements, or as ranked requirements, it is difficult to grasp the relationship between these requirements, such as their dependencies. In this paper, we propose the metamodel for iStar-p, a language that extends the *i** framework by including essential prioritization information, such as prioritization technique, prioritization criteria, the involved stakeholders in the prioritization and their weight, as well requirements priority values.

Keywords: Requirements Engineering, Requirements Prioritization, Goal Oriented Requirements Engineering, Goal Modelling.

1 Introduction

Requirements prioritization is an often-neglected sub-activity of the analysis and negotiation phase of the Requirements Engineering process. According to [14], there is a lack of visual mechanisms to help requirements engineers defining prioritization strategies. Even though the literature presents many modelling languages specific to the domain of RE, such as KAOS [15], *i** [4] and AGORA [7], they often do not support the representation of specific concepts related to requirements prioritization, such as prioritization techniques, criteria, stakeholders, and their weights. Based on the relevance of these concepts [1], we have proposed the iStar-p modelling language [5] for expressing prioritization concerns on top of regular *i** (namely iStar 2.0) [4] models.

In this paper we propose a metamodel and constraints for the iStar-p language, based on the original iStar 2.0 metamodel [4]. Section 2 presents an overview of the modelling language, whereas Section 3 presents its metamodel. Section 4 discusses related work. Lastly, section 5 presents conclusions and directions for future work.

2 The iStar-P Model for Requirements Prioritization

The core idea of iStar-p is to include prioritization onto regular *i** models. We chose to extend the *i** modeling language, as we observed that *i** already has some elements that

help support the prioritization and that i^* current approaches for prioritization specification lack expressiveness. Three concepts of the i^* language, particularly, make it a good starting point for prioritizing requirements:

- **Actors:** by including different actors and their dependency relationships with elements of the System-To-Be actor, it is possible to infer some priorities if one considers the weights of these actors.
- **Refinements:** since it is possible to visualize why each task is included in the system, one can partially infer the priority of sub-elements.
- **Contribution links:** the contribution towards qualities can aid the selection of alternative tasks.

The iStar-P model is composed of two sub-models: SPlan and SPrio. On the one hand, there is (meta) information about the prioritization process: stakeholders’ weights, prioritization technique, and the weight of each prioritization criteria. This is represented with as planning sub-model (SPlan). On the other hand, there are the prioritization values assigned by the stakeholders to each element of the model (i.e., the prioritization itself). This is represented through a prioritization sub-model (SPrio).

Fig. 1 shows an example, using the SPlan model, of a prioritization strategy for the Medi@ project. It presents a new kind of actor: Prioritization Team. The icons on top of this actor represent the stakeholders that participate in the prioritization process: project manager, developers, and users. The circle, on top of each of these icons, represent their weights: 2, 3, and 5, respectively. The resource element with a “system” topic, indicates the system (or part of the system) being prioritized (Medi@).

The resource with a “technique” topic, indicates the prioritization technique adopted (Wiegiers’ Matrix) and the prioritization criteria to be used (if any). In this example, the criteria are: benefit (weight 4), penalty (weight 2), cost (weight 2), and risk (weight 2). In Fig. 1-B we display the equivalent text information, as a table, for comparison’s sake.

Fig. 2 shows the Medi@ example with prioritization elements considering the three participants and four criteria, according to the strategy defined in Fig. 1. The prioritization values, assigned to each requirement, are visually displayed as matrices where rows

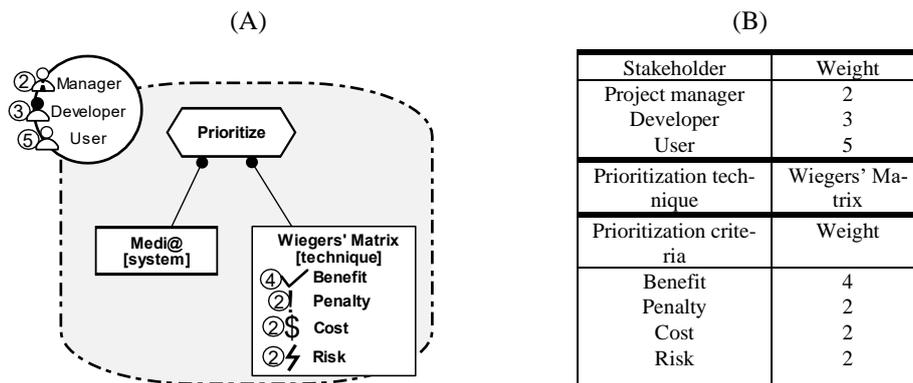


Fig. 1. Meta information instantiated for Wiegiers’ Matrix with the SPlan sub-model

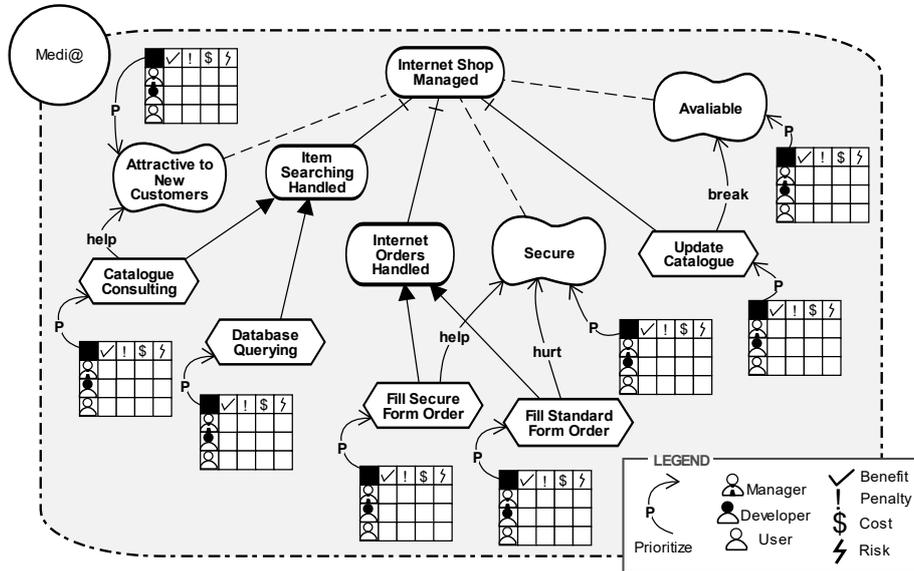


Fig. 2. Empty prioritization elements (matrices) - example of a SPrio sub-model

represent stakeholders and columns represent prioritization criteria (Fig. 2). Thus, each cell in a matrix represents the value assigned by a stakeholder to an element, regarding a particular prioritization criterium. Observe that the participants and criteria, indicated in the matrix, vary according to the project.

In this example, as defined in the SPlan (Fig. 1), the participants are Project manager, Developers, and Users. The criteria are Benefit, Penalty, Development Cost, and Risk. Icons are preferably used, instead of text, to prevent bloating the model with too much text. The prioritization element is linked to the respective element through a Prioritizes link, labeled with the letter 'P'.

The iStar-p model can be used both for data gathering and data visualization. In data gathering, participants must receive a model with empty matrices (such as Fig. 2), where they should write down the values they assign to each requirement/criterium pair. By prioritizing on top of this diagram the stakeholders will be able to consider the relationship between elements and thus make more informed decisions, compared with prioritizing just a list of requirements. For instance, when prioritizing a task, one can analyze why that task is there (dependencies and refinements) and their impact on quality constraints (contribution links).

For data visualization, a consolidated model with the input of every stakeholder and (possibly) calculated priority values can be used as an input for conflict identification, negotiation, release planning, and so on. For instance, the averaged priority values calculated based on stakeholders' and criteria's weights can be displayed in the black cells of the matrices.

3 iStar-p Metamodel

The iStar-p is a conservative extension, in the sense that it keeps all the constructs from the iStar 2.0 language. Fig. 3 depicts its metamodel, with white boxes representing constructs from the original language and light purple boxes representing the new constructs: Prioritization Team, Prioritization Participant, Prioritization Value, Prioritization Element, Prioritization Criterium, and Prioritization Technique. Its constraints are formalized with the Object Constraint Language (OCL), also presented in Fig 3.

The Prioritization Team is a specific kind of Role, on which the prioritization activity planning is detailed. It is associated with one or more participants of the prioritization. The participants are not modelled as a kind of actor, because we are not necessarily concerned with their strategic needs (goals, tasks, etc.). Intentional Elements may be prioritized through Prioritization Elements. A Prioritization Element can be associated with Prioritization Values, which represent the values given by each participant regarding each Prioritization Criterium. Even though the adopted Prioritization Technique influences the selection of prioritization criteria, they are not explicitly linked in the metamodel. Nonetheless, the Prioritization Technique is a resource used by the Prioritization Team.

Since tasks need resources, they should not be prioritized (first OCL invariant); it suffices to prioritize the tasks. Since the Prioritization Team *per se* should not interfere with the system requirements (other than their priorities), they cannot be linked to other elements of the *i** model (second and third OCL invariants). Lastly, the Prioritization Technique can only be defined within a Prioritization Team (fourth OCL invariant) and it cannot be part of a dependency (fifth OCL invariant).

4 Discussion

iStar-p is not able to support the data gathering required by some prioritization techniques but, even in these cases, it is useful for data visualization. For instance, in the Wiegers' Matrix technique a set of weighted criteria can be used to calculate priority values – iStar-p can represent both the values assigned at each criterion and the calculated priority values. On the other hand, techniques such as AHP [11] calculate priorities based on pairwise comparisons – whereas this comparative data cannot be gathered with iStar-p, the calculated priority of each alternative can be displayed with iStar-p.

In general, the techniques that would not benefit from iStar-p, for data gathering, are those techniques that require pair-wise comparison (e.g., AHP), as well as techniques that use a specific form of data representation (e.g., QFD, BST). For data visualization, most techniques can have their final results represented with iStar-p. In particular, rankings can be represented by displaying in the matrix the position of each element in the rank. Additionally, groups or categories (e.g., MoSCoW) can be represented as values in the priority matrix. Considering both data gathering and data visualization, two techniques are not supported in any way by iStar-p: Cost value approach and Win Win.

ommended to prioritize only a sub-set of requirements, such as: planned to a next release, of a certain kind (e.g., only qualities), of a given actor, and so on. Additionally, the use of filters or dynamic views can reduce the effort of analyzing such models [12][13]. Examples of applicable filters include filter by stakeholder; filter by criteria; filter by conflicting values; and filter by kind of element.

Five different visual representations have been considered in our study, as illustrated in Fig. 4. Option A was the one selected for the iStar-p. Option B, with the full name of each criterion, was discarded since it requires even more space than Option A. Options C and D were also discarded since, when compared with Option A, they need more effort to recall what the criteria associated with each column are. Options E, F and G are well suited to represent the values of a single stakeholder, but they are inadequate for the case of multiple stakeholders.

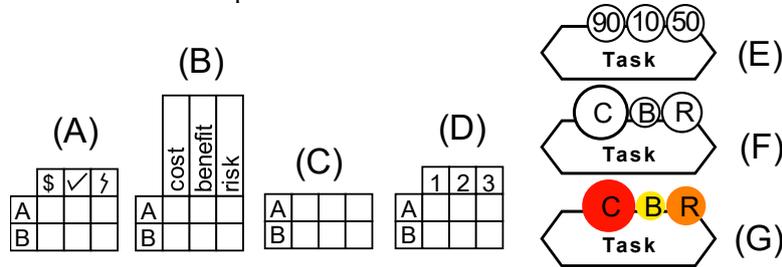


Fig. 4. Alternative notations that were considered

4.1 Related Work

There are some approaches for requirements modelling that support prioritization and decision-making concepts. The work of Kassab [8] demonstrates the application of a decision-making technique, AHP, along with the NFR Framework [3]. [9] proposes the use of AHP with i^* models to obtain quantitative indicators of softgoal satisfaction. [6] presents an approach to select alternatives on i^* models, based on desired levels of satisfaction of softgoals. [10] presents an approach to select tasks (plans) on i^* models based on prioritized temporal preferences. The iStar-p can be considered complementary to these approaches, since the information gathered through its model can be used as input for the decision-making in them. Moreover, iStar-p is technique agnostic, whereas [8] and [9] are limited to AHP.

5 Conclusion and Future Work

In previous work we have performed a systematic mapping review and interviews with practitioners to identify the essential elements of requirements prioritization [1][2]. The key elements are requirements priority, prioritization technique, prioritization criteria, stakeholder identification, stakeholders' weight, requirements identification and the number of stakeholders. Then we propose the iStar-p, a visual model which extends i^*

language to support the first five identified essential elements of requirements prioritization, along with empirical evaluation [5]. In this paper we propose the iStar-p meta-model, which allows a better understanding and formalization of the iStar-p proposal.

A study of the semiotics of the proposal is a promising venue to identify further improvements. Additional mechanisms can also be analyzed to prevent data overload – namely, filtering [12][13]. Additionally, the development of a supporting tool could facilitate the adoption of the proposal.

Acknowledgments: The authors thank FACEPE for their financial support.

References

1. Cavalcanti, C., Lencastre, M., Fagundes, R., Santos, T., Ferreira, D.: Mechanisms to Support Requirements Prioritization: A Systematic Mapping Review. In: Proceedings of 21 Workshop on Requirements Engineering. DOI 10.17771/PUCRio.wer.inf2018-52. (2018)
2. Cavalcanti, C.: Planejamento e Priorização de Requisitos em Modelos i*. Masters dissertation. University of Pernambuco, Brazil. (2017).
3. Chung, L., Nixon, B. A., Yu, E., Mylopoulos, J.: Non-functional requirements in software engineering (vol. 5). Springer (2000)
4. Dalpiaz, F., Franch, X., Horkoff, J.: iStar 2.0 language guide. In: arXiv preprint arXiv:1605.07767 (2016).
5. Flório, C.; Lencastre, M.; Pimentel, J.; Araujo, J.: iStar-p: A Modelling Language for Requirements Prioritization. Proceedings of the 38th International Conference on Conceptual Modeling (in press).
6. Horkoff, J., Yu, E.: Finding solutions in goal models: an interactive backward reasoning approach. In International Conference on Conceptual Modeling. Springer, Berlin (2010).
7. Kaiya, H.; Horai, H.; Saeki, M.: Agora: Attributed Goal-oriented Requirements Analysis Method. Proceedings IEEE Joint International Conference on Requirements Engineering DOI: 10.1109/ICRE.2002.1048501. (2002).
8. Kassab, M.: An integrated approach of AHP and NFRs framework. In: Proceedings of the 7th International Conference on Research Challenges in Information Science. IEEE (2013).
9. Liaskos, S., Jalman, R., Aranda, J.: On eliciting contribution measures in goal models. 20th IEEE International Requirements Engineering Conference (2012).
10. Liaskos, S., McIlraith, S. A., Sohrabi, S., Mylopoulos, J.: Representing and reasoning about preferences in requirements engineering. Requirements Engineering, 16(3), p. 227 (2011).
11. Saaty, T. L.: The Analytic Hierarchy Process. McGraw-Hill International, New York (1980).
12. Savio, D., Poothiyot, A. P.: Extended support for visualizing requirements: Filtering and tracing requirements in ReBlock. In: IEEE 5th International Workshop on Requirements Prioritization and Communication (RePriCo), pp. 11-14. IEEE (2014).
13. Silva, L., Moreira, A., Araújo, J., Gralha, C., Goulão, M., Amaral, V.: Exploring views for goal-oriented requirements comprehension. In: Conceptual Modeling. Springer (2016).
14. Thakurta, R.: Understanding requirement prioritization artifacts: a systematic mapping study. Requirements engineering, 22(4), pp. 491-526. Springer (2017).
15. Van Lamsweerde, A. Requirements engineering: From system goals to UML models to software. John Wiley & Sons (2009).