The Journal of Systems & Software 168 (2020) 110649

Contents lists available at ScienceDirect

The Journal of Systems & Software

journal homepage: www.elsevier.com/locate/jss

PRISE: A process to support iStar extensions

Enyo Gonçalves [a,*], João Araujo [b], Jaelson Castro [c]

[a] Universidade Federal do Ceará, Av. José de Freitas Queiroz, 5003 – Cedro CEP 63900-000, Quixadá, CE, Brazil
[b] NOVA LINCS, FCT, Universidade Nova de Lisboa, Lisboa, Portugal
[c] Universidade Federal de Pernambuco, Recife, PE, Brazil

ARTICLE INFO

Article history:
Received 5 July 2019
Received in revised form 11 May 2020
Accepted 14 May 2020
Available online 20 May 2020

Keywords:
Process
Extension
Modelling language
Goal modelling
iStar

ABSTRACT

iStar is a goal-based requirement modelling language, being used both in industrial and academic projects of different domains. Often the language is extended to incorporate new constructs related to an application domain or to adjust it to practical situations during requirements modelling. These iStar extensions have been proposed in an ad hoc way resulting in many problems of incompleteness, inconsistency and conflicts. Recently, the language was standardised, but it continues being extended. Thus, we consider that this is an adequate moment to study how to support the proposals of the next iStar extensions. In this paper, we define PRISE, a process to support the creation of iStar extensions which is driven by model-based development concepts, reuse of existing iStar extensions and guidelines of experts. This process can be customised. We illustrate the usage of PRISE by recreating five existing iStar extensions. Finally, we evaluated PRISE with interviews and a survey with experts; and, we performed an interview to analyse the opinion about the usage of the PRISE to create a new iStar extension by a novice. The evaluation and validation indicate good results to avoid problems and increase the quality of the proposals and well receptivity by the experts and novice.

1. Introduction

Requirements Engineering (RE) is concerned with real-world goals for software functionalities or services and constraints on software (Zave, 1997). RE is an important part of software development process which covers from elicitation and specification to requirements maintenance (Kotonya and Sommerville, 1998). Evidence shows that requirements errors, such as misunderstandings and omissions, are more expensive to repair in later phases, such as design, implementation and testing (Boehm, 2007).

Over the last two decades, much attention has been paid to the area of goal-oriented requirements engineering (GORE) and goal-oriented modelling is one of the most important approaches in RE with considerable research devoted to it (Horkoff et al., 2019). While object-oriented analysis fits well to the late stages of requirement analysis, the goal-oriented analysis is more appropriate to the earlier stages where the organisational goals are analysed to identify and justify software requirements and frame them within the organisational context (Mylopoulos et al., 1999). Some goal-based modelling languages have been proposed, such as KAOS (Knowledge Acquisition in autOmated Specification) (Dardenne et al., 1993), and iStar (Yu, 1995).

iStar had some variations in its default syntax (e.g., Toronto iStar Yu, 1997 and Trento iStar Yu, 1995). Efforts were made towards unifying the language notation and establishing a core. Horkoff et al. (2008) analysed the iStar constructs variation (Trento and Toronto), while Cares and Franch defined a reference metamodel (Cares and Franch, 2011). Recently a new version of iStar has been proposed (Dalpiaz et al., 2016), called iStar 2.0.

Extending a Modelling Language (ML) is to add new constructs or modify old ones (Brambilla et al., 2012). Both DSML (Domain Specific Modelling Languages) and GPML (General Purpose Modelling Languages) can have their syntax changed for various reasons. However, GPML are more prone to extensions than DSML because they are proposed in a generic way for any domain. In general, DSML are proposed to specify systems in a domain/application area and GPML are proposed to model a great variety of domains/application areas. Based on this point of view, iStar can be classified as GPML.

Due to the increasing complexity of system development and the diversity of application areas, the proposal of iStar extensions increased in the last few years. Several extensions have been made to suit them to specific application areas, such as Data Warehouse (Giorgini et al., 2005) and Security (Mouratidis and Giorgini, 2007). Autonomic computing systems application area was source for extensions of iStar related to configuration, optimisation, healing and protection of autonomic applications (Lapouchnian et al., 2006). Social Technical Systems (STS) is

* Corresponding author.
E-mail addresses: enyo@ufc.br (E. Gonçalves), joao.araujo@fct.unl.pt (J. Araujo), jbc@cin.ufpe.br (J. Castro).

https://doi.org/10.1016/j.jss.2020.110649
0164-1212/

another application area for iStar extensions, such as the extension involving conflicts of interest in health care (Chung, 2006). Legal aspects were also considered representing regulations in GRL (Ghanavati et al., 2014). These extensions were identified in a Systematic Literature Review of iStar extensions (Gonçalves et al., 2017a) which points to 42.7% of extensions did not present or partially presented the definition of concepts introduced; 62.5% represented the extensions only in concrete syntax; 77.8% of extensions that represent abstract or both syntaxes have an absence of nodes and links of default syntax of iStar; 37.14% of extensions have no compatibility between abstract and concrete syntax; 53.6% of the extensions are not supported by a modelling tool and 108 conflicts (Symbol redundancy, symbol overload, symbol deficit or symbol excess) were identified in iStar extensions.

We understand that there are different forms to present an iStar extension, but all of them introduce new elements to iStar or change old ones. For example, a set of extensions described in detail the new elements and its representations in the iStar metamodel and concrete syntax (see work described in Ali et al., 2008 and Morandini et al., 2015). This kind of extensions describes how the new elements were introduced and how to use them.

Due to the proposal of a new version of iStar (Dalpiaz et al., 2016), we believe this is the best moment to discuss how iStar extensions could be supported by a process. This paper presents in detail the PRocess to support iStar Extensions (PRISE). It is the result of a series of studies (Gonçalves et al., 2017a,b) to understand how extensions were performed and what could be done to improve. The experts in iStar extensions suggested the creation of a process for supporting iStar extensions in research (Gonçalves et al., 2017b). PRISE is important to help novices and experts in iStar extensions to propose their extensions avoiding common mistakes identified in the existing iStar extensions, such as incompleteness, inconsistencies and conflicts. It is work noting that a preliminary and limited version of PRISE was published at Gonçalves et al. (2020).

This paper is organised as follows. Section 2 discusses iStar extensions and the related work. Section 3 presents the related work. Section 4 shows the steps followed in this research. PRISE is presented in Section 5. Section 6 presents a customisation of the PRISE. A validation of the PRISE is presented in Section 7, where we recreated an existing iStar extension to illustrate its usage. Section 8 shows the evaluation of the PRISE based on interviews and a survey. Finally, the conclusions and future work are presented in Section 9.

2. iStar extensions

In this section, we presented an overview about iStar extensions. There are different ways to present an iStar extension (Gonçalves et al., 2017a), but all of them introduce new concepts to iStar, for example, a set of extensions described in detail the new concepts and its representations in the iStar metamodel and concrete syntax (e.g., Ali et al., 2008 and Morandini et al., 2015). These kinds of extensions describe how the new concepts were introduced and how to use them.

On the other hand, another set of extensions was presented together with the method to create the model, with the iStar modifications presented by illustrations with the usage of new concepts. Examples of this kind of extensions are the work of Guzman et al. (2016) and the extension of Islam et al. (2012).

In some cases, an approach presents an illustration or a modelling tool with a set of new concepts introduced in iStar, for example in Gans et al. (2006) and Siena et al. (2008).

We do not consider as an extension any work that used iStar without changes in abstract syntax (changes in metamodel or validation rules) or concrete syntax (new graphical representation) because in this case the iStar is used with default syntax without any changes (extension).

In previous work (Gonçalves et al., 2017a), the iStar extensions were analysed and classified. The results indicate that 77.8% of extensions which extended both syntaxes are non-conservative. To check this information, it was required to analyse both the metamodel and the concrete syntax of the extensions. Therefore, only extensions that presented the abstract syntax and concrete syntax were considered in this analysis.

We also classified the extensions as lightweight, heavyweight or both. The results point out to seventeen extensions (17.7%) were only lightweight, thirty-seven extensions (38.5%) were only heavyweight, and forty-two extensions (43.8%) were both.

TROPOS4AS (Morandini et al., 2015) is an example of an iStar extension, in the Tropos context (Bresciani et al., 2004). It models characteristics of adaptive systems in Tropos, providing conceptual models, a graphical language and its semantics, to enable capturing requirements needed for defining and driving adaptation.

Fig. 1 shows an example of an insulin pump (Martins et al., 2015) modelled with TROPOS4AS. An Insulin Infusion Pump is a device intended to deliver rapid-acting insulin dosages through a catheter placed under the patient's skin to treat Type I diabetes mellitus while maintaining the adequate glucose level in the patient's blood. The model of Fig. 1 presents the patient depending on the insulin pump to have the glucose level controlled. The insulin pump establishes two maintained goals to achieve the main goal, i.e. the patient's glucose checked, and insulin injected. The injection of insulin occurs when the condition of the *patient's glucose level is low* is satisfied. A failure due to the lack of insulin may occur during the injection of insulin, and an error can happen if there is no insulin available. Finally, the recovery plan is performed. We can easily identify elements that are not part of iStar default syntax, such as maintained goal, condition, error, failure and plan.

3. Related work

We did not find a process or guidelines to support extensions in iStar or other modelling languages. Thus, as a related work we have considered an approach to join iStar with other MLs, a study that defines a set of qualities to evaluate iStar, approaches to transform iStar models in models of other phases of the software development and a method to investigate the usability of DSL.

The work by Franch et al. (2011) defines an approach to join iStar with other MLs, i.e., they present results of a search for the existing articles which present approaches of modelling that join iStar with other MLs. The authors also present guidelines to be followed considering the theoretical, technical, methodological and community aspects. However, the results are not related to iStar extensions when the concepts to be introduced to iStar are not from an existing modelling language. It has not considered important concepts such as Physics of Notations (PoN) (Moody, 2009) and the compilation of opinion of iStar experts. It also did not present a repository of existing works.

In Lòpez et al. (2016), the authors proposed a list of qualities to evaluate iStar. This list is composed by 11 qualities grouped in three categories (Syntactic, Semantic and Pragmatic). Extensibility appears as a quality and is described as the facility to add new concepts to iStar. However, this paper does not present how to extend iStar or achieve these qualities in iStar extensions.

Many works treated the transformation M2M (Model-to-Model) from iStar models. (Alencar, 1999) shows the mapping from organisational modelling with iStar to precise specification of systems. Lucena (2010) created a process to derive architectural models from iStar models to acme models (Garlan et al., 1997). This process includes horizontal and

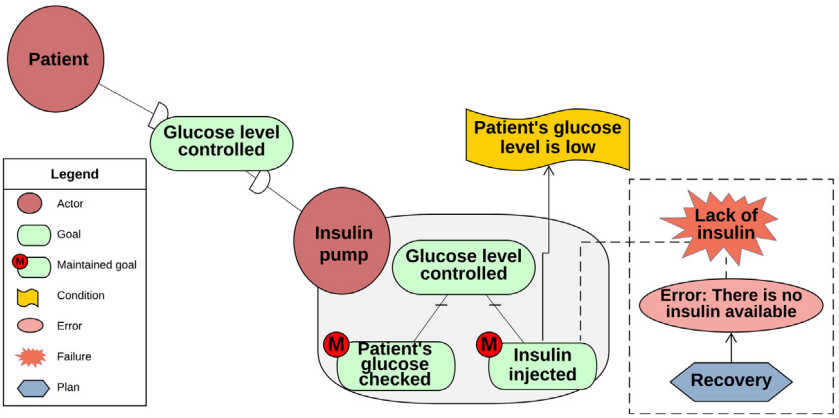


Fig. 1. Modelling of insulin pump with TROPOS4AS.

vertical transformations rules. The horizontal transformations are applied to the requirements models resulting in intermediary requirements models closer to architectural models. Vertical transformations map these intermediary models into architectural models. The activities related to architectural design involves the selection and application of architectural patterns that best satisfy non-functional requirements. In Santander (2002), the author considers the integration of iStar with the UML use case model. A set of guidelines was proposed to integrate the modelling both ML. In that work, the author uses iStar to create organisational modelling and the user case models are derived from it. Bastos (2005) considers the transformation from iStar models to architectural models in context of TROPOS, it is composed of schemas, templates and guidelines to support the transformations. However, all these works do not consider the iStar extensions development.

Barišić et al. (2011) defined a method to investigate the usability of Domain Specific Languages (DSL). They represent the development of a Domain Specific ML in three parts, the domain engineering, design the language, implementation and testing. The domain engineering is related to specify the context of use, design the language concerns to produce design solutions, and implementation and testing are considered the phases to produce tools and evaluate the proposal respectively. However, the main contribution of the process is the evaluation of usability in DSLs development. It did not create a process to support iStar extensions.

4. Methodology

We had developed a set of previous works (Gonçalves et al., 2017a,b, 2018a,b, 2019c,a) to identify how the iStar extensions have been performed and what could be done to improve future iStar extensions. Initially, we established a set of principles and fundamentals about iStar extensions. We had identified the existing iStar extensions by a Systematic Literature Review (SLR) to identify and analyse them (Gonçalves et al., 2017a). We found out 96 iStar extensions identified until 2016. We also had performed a study to identify how researchers proposed the iStar extensions and their point of view related to what could be done to improve the proposal of future iStar extensions (Gonçalves et al., 2017b). PRISE also uses the results of the following previous works: towards extensions mechanisms in iStar 2.0 (Gonçalves et al., 2018a); a catalogue of iStar extensions (Gonçalves et al., 2018b); and the results of an experiment for mitigating problems of symbol redundancy (Gonçalves et al., 2019c) and a survey for mitigating problems of symbol overload (Gonçalves et al., 2019a) in previous iStar extensions. We also analysed the state-of-art of the modelling languages and extensions which we can highlight (Brambilla et al., 2012; Moody, 2009; Caire et al., 2013).

Therefore, we considered these results in the proposal of PRISE. Initially, we listed what is required in PRISE definition, based on the principles and fundamentals. Thus, we modelled an initial version of PRISE using BPMN. Moreover, PRISE was analysed by five researchers with knowledge in process modelling and three experts in iStar extensions (initial analysis). We made the refinement of PRISE considering the suggestions. As a result, 71 out of 84 suggestions of corrections/improvements to PRISE were implemented. Note that 6 of remaining 13 suggestions were similar to the implemented ones and 7 were considered unsuitable.

We validated and illustrated the usage of PRISE recreating a set of five existing iStar extensions. New corrections identified in this step were applied. Finally, we evaluated PRISE with the researchers who proposed iStar extensions. We interviewed three experts in iStar extensions and also received feedback from a survey of 20 other experts in iStar extensions. Final corrections were applied. Fig. 2 shows the steps followed to define this process.

5. The PRISE process

PRISE (PRocess to support iStar Extensions) aims to systematically guide the proposals of iStar extensions to make them as complete, consistent and without conflicts as possible. This section presents the main process and the subprocesses of the PRISE in each sub-section. Therefore, Section 5.1 presents main process, Section 5.2 describes the subprocess *1. Analyse the need for extension*, Section 5.3 details the subprocess *2. Describe concepts of the iStar extension*, The Section 5.4 presents the subprocess *3. Develop iStar extension*, the subprocess *4. Validate and evaluate the iStar extension* is presented in Section 5.5, Section 5.6 details the subprocess *6. Publicise the iStar extension* and, finally, Section 5.7 presents general considerations about the PRISE.

5.1. Main process

The process starts when there is the *intention to extend iStar*. The main process consists of five sub-processes, one task and the evolution of the *Extension specification* artefact. In Fig. 3 we have used BPMN (Business Process Modelling Notation) to depict the PRISE approach. We describe this process in detail as follows.

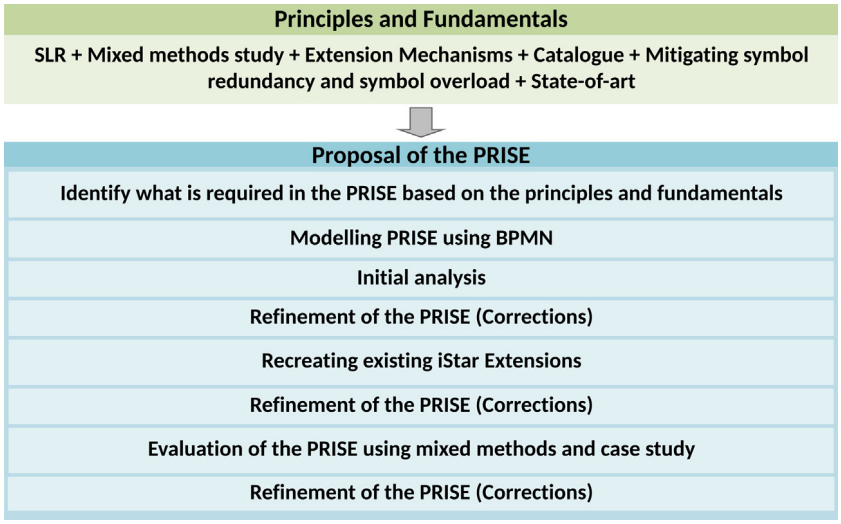


Fig. 2. Methodology for creating PRISE.

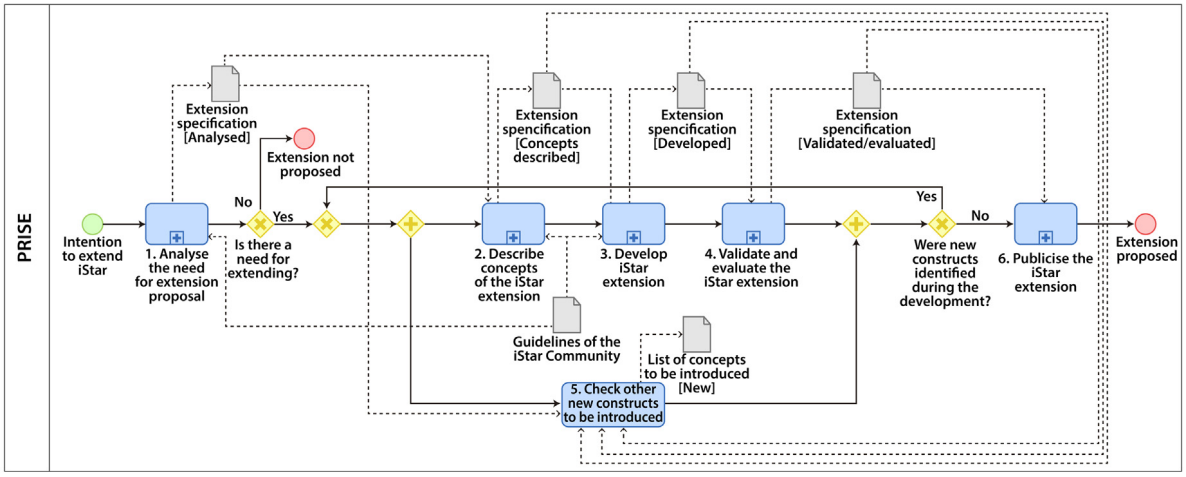


Fig. 3. Main process of the PRISE.

1. Analyse the need for extension. This sub-process consists of verifying the need for the iStar extension. It is important to analyse the need for proposing the extension and to establish the list of the concepts to be introduced (without description of them). It generates the *Extension specification [Analysed]*. This sub-process is detailed in Section 5.2.

Then, a decision is made based on the conclusions of the analysis of the need for an extension proposal. If there is a need for the extension, the process execution continues with the proposal. Otherwise, the process ends without proposing an iStar extension as there is no need for it. When the process execution continues, three sub-processes (2, 3 and 4) are executed in sequence and the task 5 is executed in parallel to these three sub-processes.

2. Describe concepts of the iStar extension. This sub-process describes in detail the concepts identified in sub-process 1. It is important to try to identify constructs to be reused and to analyse if it is possible to relate the constructs to be introduced with the original iStar constructs. It creates the *Extension specification [Concepts described]*. This sub-process is detailed in Section 5.3.

3. Develop iStar extension. This is the main part of PRISE. The *Extender* should consider the set of guidelines provided by iStar experts for the elaboration of extensions before starting this sub-process. This sub-process introduces new concepts in the iStar modelling language. It produces the *Extension specification [Developed]* with the description of the iStar extension. This sub-process is detailed in Section 5.4.

The guidelines are based on the opinion of experts as presented in paper (Gonçalves et al., 2017b), as well as new insights gained during the execution of the SLR (Gonçalves et al., 2017a) and development of the PRISE process. The *Guidelines of the iStar Community* are listed as follows:

- G1 — Preserve the language (iStar) original syntax: It recommends that all nodes and links of the default iStar syntax should be maintained in the extension. In other words, the non-conservative extensions, which remove iStar default constructs, are discouraged. Thus, it is not considered as a good practice, for example, the *Quality* model element be removed during the iStar extension proposal;
- G2 — Carry out consistent, complete and without conflicts extensions and follow a process/method to do them: The iStar extensions have been proposed in an ad hoc fashion manner. This fact has a negative impact on the proposal and many problems of inconsistencies, incompleteness and conflicts have been occurring. Therefore, this guideline highlights the importance of following a process (PRISE) to avoid inconsistencies, incompleteness and conflicts;

- G3 — Perform a literature review, consider the participation of domain experts and iStar experts, and model systems of application area before extending: the literature review can contribute with the understanding of the domain and to set clear definitions to the concepts of this domain. The participation of domain experts and iStar can contribute to avoid errors during iStar extensions proposal;
- G4 — Describe a clear definition of the extension concepts: Concepts which are not described cannot be understood by those who intend to use the extension;
- G5 — Propose concrete and abstract syntax of the extension: The metamodel and well-formedness rules are important to a modelling language and their extensions, once they represent what is allowed and what is forbidden in its usage. Therefore, both syntaxes (abstract and concrete) should be considered during the extension proposition;
- G6 — Check consistency between abstract and concrete syntaxes: it is important to consider checking consistency between abstract and concrete syntaxes. The inconsistency can be described by the representation of a new construct in only one of them, for example including a new graphical representation in concrete syntax, without adding a new metaclass to represent it in the extension metamodel;
- G7 — Relate concepts introduced by the extensions with the iStar concepts: The new concepts introduced by the extension can be related to the iStar concepts, specialising the iStar constructs or be connected by iStar relationships or other relationships;
- G8 — Define extensions with a smallest possible number of modifications and new representations in order not to complicate the use of the modelling language (iStar): the extension should minimise the number of modifications and new representations in iStar as much as possible. It is important that the iStar extension does not introduce a great number of constructs, making the usage of the iStar extension hard. Otherwise, it can mischaracterise iStar and worsen its scalability;
- G9 — Propose careful and simple graphical representations, able to be drawn on paper without a tool: the proposal of new constructs should be simple enough for people to hand draw and do it on the whiteboard and scrap it off and make changes to it very quickly. They should maintain the principle of the symbols of iStar, which are simple and easy to draw. If possible, in order to choose an appropriate graphical representation, try to perform an experiment such as the one suggested by Caire et al. (2013).

Part of these guidelines can be related to the Moody's principles (Moody, 2009). G2 and G6 are related to the Semiotic clarity since they recommend avoiding inconsistencies and conflicts. G7 concerns to the Cognitive integration because it recommends relating the introduced constructs with the existing constructs. G8 is related to the Manageable complexity and Graphic economy due it recommends the proposal of the smallest possible number of modifications and new representations. G9 impacts on the application of the Cognitive fit, Perceptual discriminability, Visual expressiveness and Semantic transparency since it recommends create simple symbols to keep the look and feel with the iStar default symbols and consequently it reduces the expressiveness when the proposal is free to create any symbol.

4. Validate and evaluate the iStar extension. This sub-process illustrates the usage, validates the extension with experts, refines and evaluates the iStar extension developed in sub-process 3. It generates the *Extension specification [Validated/evaluated]*. It is detailed in Section 5.5.

5. Check other new constructs to be introduced. PRISE is an iterative process. Thus, new constructs (different of the set identified in the sub-process 1) can be identified during sub-processes 2, 3 and 4 by this task. Thus, this task is performed in parallel during their execution of these three subprocesses. When new constructs are identified during an iteration of sub-processes 2, 3 and 4, they are listed by this task to be considered in the next iteration of them. This task generates the *List of concepts to be introduced [New]*.

When new constructs are identified in task 5, the execution of PRISE returns to task 2 to include them in the iStar extension. Otherwise, the execution continues to the sub-process 6 (*Publicise the iStar extension*).

6. Publicise the iStar extension. This sub-process details how to make the iStar extension accessible to the community. It is detailed in Section 5.6. Finally, the proposal of the iStar extension is considered finished.

We describe the sub-processes of the PRISE in next sections. PRISE involves three roles in their sub-processes: *Extender* (represents the researcher who develops the extension), *Expert in iStar extensions* (represents the experienced researchers in proposals of previous iStar extensions, which are external to the process), *Experts in domain/application area* (represents the experienced researchers in the domain/application area, which are external to the process). When the experts in iStar or domain/application area propose the iStar extension, they play the role of *Extender*.

We now describe the first subprocess.

5.2. Analyse the need for extension proposal (sub-process 1)

This sub-process is needed to analyse whether the extension should be proposed. It introduces a set of tasks to avoid the proposal of unnecessary extensions such as study the domain, model an example with iStar and consult experts in iStar extensions. It is also important to the extender better understand the context of the extension to be proposed. This sub-process is composed of eleven tasks and five artefacts. Fig. 4 relies on BPMN to depict sub-process 1. We detailed this sub-process in the next paragraphs.

Initially, this sub-process tests if the proposed iStar extension is related to an application area (such as security or robotics, for example) or if it is related to practical aspects of iStar (such as modularisation, cardinality or duration of tasks, for example). It is also possible that an iStar extension involves both.

When the iStar extension involves an application area, the *Extender* should *study/review the domain/application area* (task 1.1). This task generates a set of references (Books, Book chapters and papers) identified in the study and, if applicable, use a *list of the contacted researchers* (see this artefact in Section 7.1).

The execution of the process continues with task *1.2. Identify the concepts to be introduced by the extension*. This task consists of identifying the *list of concepts to be introduced*. These concepts can be related to a domain/application area or practical aspects. Task 1.2 considers the references listed in the previous task and generates a *List of concepts to be introduced* (see Section 7.2).

PRISE has a sequence of six tests (gateways) described as follows. The first test (*What is the purpose of the extension?*) verifies the need of analyse a domain/application area.

The second test (*Is there any issue about domain/application area?*) verifies if there is any issue about the domain/application area. When there is any issue, the *Extender* contacts *experts in domain/application area* (task 1.3), the experts in the domain/application area receives and mitigates the issues (task 1.4).

Other opportunities to resolve the questions are during validation and evaluation (sub-process 4, during the sub-process 4.3 which contact experts, the modelling of an example, evaluating the extension by the participants) and publicising the iStar extension (sub-process 6, during the submission of the extension).

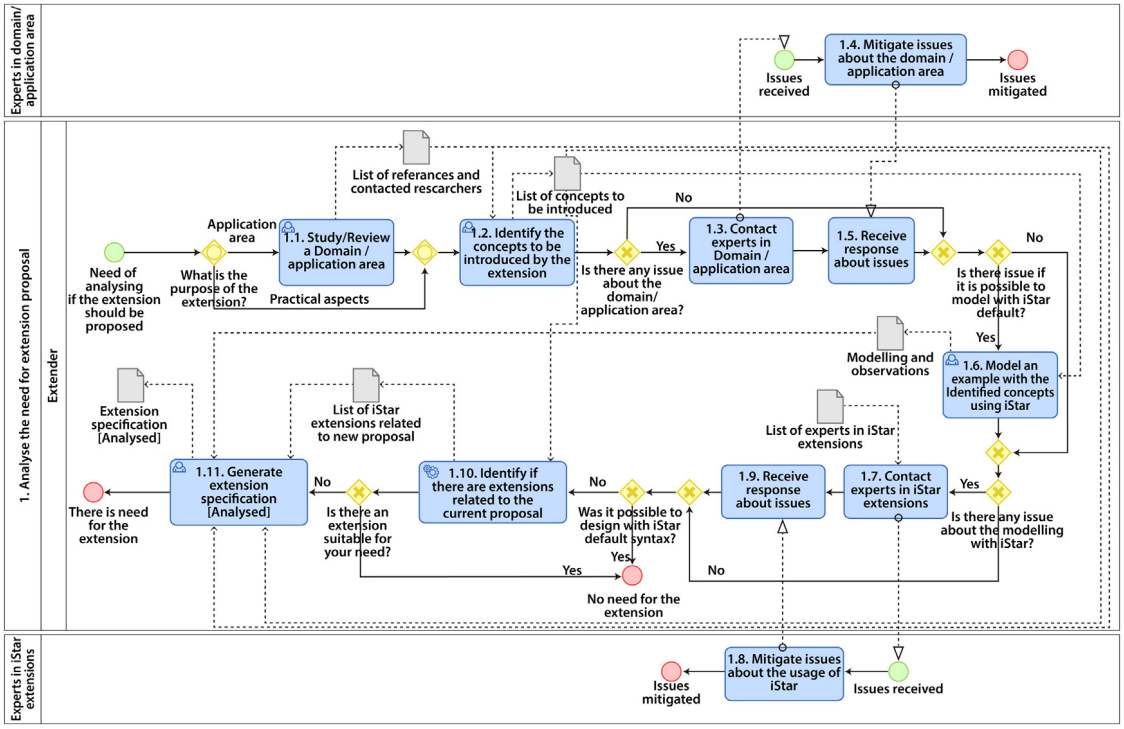


Fig. 4. Sub-process 1. Analyse the need for extension proposal.

The third test (*Is there any issue when modelling the extension using the default iStar?*) is about the possibility to use the default iStar to model the new concept. Sometimes, it is possible to model the systems of a targeted domain without extending iStar. When there are hesitations about it, the *Extender* should use the current version of iStar to try to model the concept targeted by the extension (*List of the concepts to be introduced*) (task 1.6). It is interesting to use a complete (real) example, not a "toy" one. When the *Extender* identifies new concepts to be introduced by the extension, these concepts should be included in the List of concepts to be introduced. This task generates an artefact with the *modelling and observations* (see Section 7.3).

An iStar extension can be done by inexperienced researchers, so in this case hesitations can appear. These may be related to the possibility or not to model the domain concepts with iStar (or other concern about the modelling) with iStar default syntax.

The fourth test (*Is there any issue about the modelling with iStar?)* can be related to the modelling of new concepts with iStar (task 1.6). When there is any issue, the *Extender* contacts *Experts in iStar extensions* (task 1.7), the *Expert* in iStar extensions *receives and mitigate the issues* (task 1.8). There is the possibility that the extension needs to wait a lot for an answer from the expert or do not receive any response. Thus, the *Extender* should continue the execution of the PRISE when did not receive answer from the Experts.

The fifth test (*Was it possible to design with iStar default syntax?*) analyses if it is possible to use the iStar default syntax. If it is the case, there is no need for the extension. Otherwise, the *Extender* should i*dentify if there are extensions related to the current proposal* (task 1.10), which generates the List of iStar extensions related to the new proposal (see Section 7.4). The existing iStar extension is considered suitable when introducing the concepts identified in the *List of concepts to be introduced*.

This action can be performed with the help of iStar extensions' catalogue[1] (Gonçalves et al., 2018b). The *Extender* should search the catalogue by application area and after that search by the name of the constructs to be introduced. Additionally, if the *Extender* did not find the extension in the catalogue, he/she can examine the search sources, as done in Gonçalves et al. (2017a), to analyse the period between the last update and the current date.

The sixth test analyses if there is an extension suitable for the extender's need. If there is a suitable extension, there is no need to propose a new extension.

Otherwise, the proposal continues with the task *1.11. Generate extension specification [Analysed]*, which joins the artefacts generated in this flow in the *Extension specification [Analysed]*. Finally, this sub-process finishes with the extension identified.

The next section details the sub-process *2. Describe concepts of the iStar extensions*.

5.3. Describe concepts of the iStar extension (sub-process 2)

This sub-process is concerned with the reuse of existing constructs since if the creation of new symbols to constructs proposed by existing extensions implies in symbol redundancy. The description of the concepts' meaning is important as the extender should know their purpose and the users of the extension better understand the new constructs introduced by the extension. Extensions defined without the meaning of the introduced constructs are incomplete and could be difficult to use by other users. Finally, it is important to integrate the new constructs to the iStar constructs since they will be used together in the same model.

This sub-process starts when the need for extending iStar is confirmed in sub-process 1 (Section 5.2). It has seven tasks, six artefacts and involves the *Extender* and the *Expert in iStar extensions*. *Extension specification [Concepts described]* is the main artefact generated by this sub-process add the results of this sub-process to the *Extension specification*. Fig. 5 depicts the sub-process 2. (*Describe concepts of the iStar extension*). We detail each element of this sub-process in the next paragraphs.

[1] http://istarextensions.cin.ufpe.br/catalogue/.

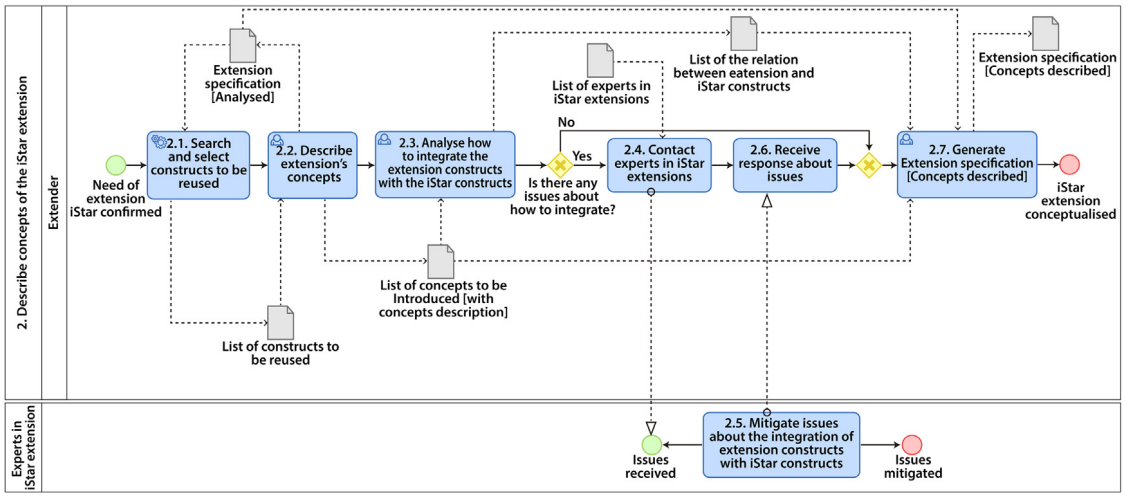


Fig. 5. Sub-process 2. Describe concepts of the iStar extension.

The first task of this sub-process is *Search and select constructs to be reused* (task 2.1). The *Extender* access the catalogue[1] to search and select the constructs identified in *Extension specification [Analysed]*.

The *Extender* can choose not to reuse some construct, because it may be considered unsuitable. For example, the *Extender* may not consider reusing constructs that have the same name, but with different meanings from the intended one. It is also possible that the graphical representation is considered inadequate. When the catalogue shows conflict involving the construct to be reused, the *Extender* can consider the prioritisation or the new graphical representation of the construct in conflict with the ones present in the catalogue. This task produces the *List of constructs to be reused* (See Section 7.5).

Additionally, if the *Extender* did not find the constructs in the catalogue, he/she can examine the search sources, as done in Gonçalves et al. (2017a), to analyse the period between the last update and the current date.

The execution of this sub-process continues with the task *2.2. Describe extension's concepts*. This task is concerned with describing in detail the list of constructs to be introduced and/or reused by the iStar extension. It uses the list of the concepts to be introduced in *Extension specification [analysed]* and the list of constructs to be reused, generating the *List of concepts to be introduced [with concepts description]*, with the meaning of constructs to be introduced (see Section 7.6).

Next, the *Extender* should *Analyse how to integrate the extension constructs with the iStar constructs* (task 2.3). This task is related to the identification of the relation between the extension constructs and the iStar constructs.

The first level of integration is specialising an existing construct of iStar, e.g., textual markers as stereotypes can be used to do it (see our proposal for this kind of representations in Gonçalves et al., 2018a). When this is not possible, new graphical representations should be proposed. iStar has two diagrams SD and SR. So, it is important to identify the impact of constructs in each diagram. This task produces the *List of the relations between the extension and iStar constructs* (See Section 7.7).

Sometimes the *Extender* can have issues about the execution of the extension, e.g., if it is suitable to specialise the task metaclass to represent a specific domain concept. If there is any issue on how to integrate, the *Extender* contacts experts in iStar extensions (task 2.4) in the same way presented in the tasks 1.7, 1.8 and 1.9, the *Expert* in iStar extensions receives and mitigates the issues (task 2.5). Finally, the *Extender* receives a response about the issues (task 2.6).

We believe these alternatives are enough to obtain an answer. However, if the Extender did not receive an answer, he/she should continue to the next task of the PRISE and try to mitigate the questions in other task of the process such as the validation and evaluation (sub-process 4, during the sub-process 4.3 which contact experts, the modelling of an example, evaluating the extension by the participants); and publicising the iStar extension (sub-process 6, during the submission of the extension).

Finally, task 2.7 (*Generate Extension specification [Concepts described]*) joins *Extension specification [Analysed]* with the artefacts proposed in this sub-process (*List of constructs to be reused [with concepts definition]* and *List of the relation between extension and iStar constructs*). This sub-process finishes with the iStar extension conceptualised.

The next section details the sub-process 3 Develop iStar extension.

5.4. Develop iStar extension (sub-process 3)

This sub-process is concerned with the definition of the metamodel, validation rules and concrete syntax. Extensions defined without a metamodel or the concrete syntax are incomplete and difficult to understand and use. The check and correction of problems are important to avoid the occurrence of incompleteness, inconsistency and conflicts which cause misunderstanding in the extensions. The proposal of tools is focused on this process, which is important to make feasible the usage of the extension.

This sub-process starts when the iStar extension has its concepts defined in sub-process 2 (Section 5.3). It has five tasks, one sub-process, six artefacts and only involves the *Extender*. *Extension specification [Developed]* is the main artefact generated by this sub-process, it adds the results of this sub-process to the Extension specification. Fig. 6 depicts the sub-process 3 (Develop iStar extension) in BPMN. We describe each element of this sub-process in the next paragraphs.

The first step of this sub-process is to Define metamodel for extension (task 3.1). The Extender should consider the concepts to be introduced and the relationship of the extension's constructs and iStar constructs, both are present in the Extension specification [Concepts described]. These concepts should be introduced in the iStar metamodel.

The guidelines of the iStar community G1 (Preserve the language (iStar) original syntax) and G7 (Relate concepts introduced by the extensions with the iStar concepts) are related to this task.

Some specific recommendations about the extension of the iStar metamodel are presented as follows:

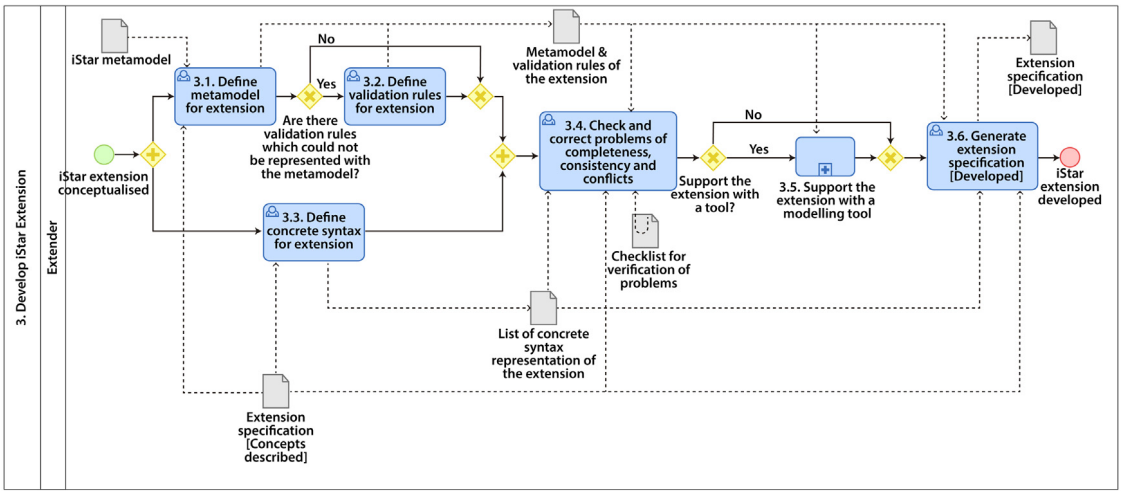


Fig. 6. Sub-process 3. Develop iStar extension.

- When there are concepts which are specific kinds of iStar nodes and links, the extender may specialise it. If the construct is a specialisation of the iStar nodes – actor, agent, role, goal, task, resource, quality (for example a safety goal which specialises goal and simple reflex agent which specialises agent) – or a specialisation of the iStar links — refinement, qualification, contribution, neededby, dependency and association (for example restriction in contribution link). This representation can be done by the creation of stereotypes or by a new metaclass that specialises it.
- The metaclass intentional element is a kind of classifier that groups related constructs that are desired to the system. The metaclasses refinement and goal/task element also are used to group related constructs in the iStar metamodel. Therefore, when a new construct is not a sub-type of iStar nodes and links, the extender has to try to specialise one of these metaclasses to group the new constructs with the existing ones;
- When a metaclass of an iStar node (or a metaclass that groups nodes) is specialised, the links which that node is a target or source are also applied to the new one.
- Similarly, when a metaclass of an iStar link (or a metaclass that groups links) is specialised, the target or source nodes are also applied to the new one. It is possible to restrict the target and source by constraints.

The following references can help the extender to perform this task. A tutorial about the metamodeling creation and representation is presented in Paige et al. (2014). A standard about metamodeling is presented in ISO/IEC, ISO/IEC 24744 (2007). There is a section related to the extension of the metamodels which presents guidelines related to it. Papers Henderson-Sellers et al. (2013) and Atkinson and Kühne (2003) describe some problems that can occur and practical solutions to avoid them.

Also, validation rules should express the constraints which are not possible to represent directly in the metamodel. Thus, the task *3.2. Define validation rules for extension* is performed when there are validation rules which cannot be represented with the metamodel. The tasks 3.1 and 3.2 create the artefact *Metamodel & validation rules* of the extension (see Section 7.9).

The task 3.3 (*Define concrete syntax for extension*) runs in parallel to tasks 3.1 and 3.2. In this task (3.3), the Extender should define the graphical representation of the constructs. One possibility to achieve this is to conduct an experiment similar to Caire et al. (2013). The *List of concrete syntax representation of the extension* is the result of this task (see Section 7.10).

The recommendation of the task 2.3 should be considered also here. It suggests that the first level of integration is specialising an existing construct of iStar, e.g., textual markers as stereotypes can be used to do it (see our proposal for this kind of representations in Gonçalves et al., 2018a). Textual markers have a low impact since it introduces new representations in the metamodel without the need to create new metaclasses. When this is not possible, new graphical representations should be proposed. iStar has two diagrams SD and SR. Therefore, it is important to identify the impact of constructs on each diagram.

We have two observations related to this task:

- The *Extender* should propose graphical representations different from existing iStar extensions;
- The *Extender* should try to reuse a construct of an existing iStar extension. However, he/she can choose not to reuse it when the graphical representation is considered unsuitable to represent the concept.

3. 4. Check and correct problems of completeness, consistency and conflicts, this is performed when the proposal of both syntaxes is finished. This task consists of analysing the following items:

- Completeness of the extension regarding the definition of its concepts, abstract and concrete syntaxes;
- Consistency between its concepts, abstract and concrete syntaxes;
- Absence of nodes and links of the iStar standard syntax;
- The occurrence of conflicts with existing iStar extensions.

The *Extender* should consider the concepts listed in the *Extension specification [Conceptualised]*, *Metamodel & validation rules* of the extension and *List of concrete syntax representation*. Thus, we have some indicators to be considered for each concept in this analysis presented in Table 1.

These indicators are used to verify the quality of the extensions. The *Checklist for verification of problems* (see Section 7.11) is used to verify the occurrence of the problems listed above. The *Extender* should correct the problems of incompleteness, inconsistency and conflicts identified.

Next, when it is feasible, the iStar extension is supported by a modelling tool (Sub-process 3.5). The feasibility analysis should be done by the Extender considering the criteria of the availability of the selected base-tool and the chronogram of the research. The support of the iStar extension with a modelling tool is detailed in Section 5.4.1 *Support the extension with a modelling tool* (sub-process 3.5).

Table 1
List of indicators of istar extensions.

Indicator	Possible value
List of the concepts	Present/not present
Description of the concepts	Present/not present
Abstract syntax definition	Present/not present
Concrete syntax definition	Present/not present
Number of inconsistencies between the concepts' description, abstract and concrete syntax	Integer
Number of absent of nodes and links of standard syntax of iStar	Integer
Number of changes in well-formedness rules of iStar	Integer
Number of conflicts of one concept with two or more representations in concrete syntax	Integer
Number of conflicts of two or more concepts with only one construct in concrete syntax	Integer
Number of conflicts of new constructs in conflict with the iStar standard syntax	Integer
Number of conflicts of wrong representation of iStar standard syntax construct	Integer
Number of conflicts of representation of constructs that are not part of the extension	Integer

Finally, the *Extender* should *Generate extension specification [Developed]* (task 3.6). This task consists of adding the artefacts generated.

Next, we present the sub-process 3.5 Support the Extension with a Modelling Tool.

5.4.1. Support the extension with a modelling tool (sub-process 3.5)

Fig. 7 details the sub-process 3.5 (*Support the extension with a modelling tool*) which is concerned with the implementation, testing and modification of a modelling tool to support the extension.

The task *3.5.1. Implement the extension in a modelling tool* is concerned with the implementation of a case tool to support the iStar extension. A list of existing iStar tools can be found in this site.[2] Extenders can create their own tool without considering the existing ones as a starting point.

The *Extender* should *Test the modelling tool* (task 3.5.2). So, the modelling tool should be tested to check what is allowed and forbidden in the extension. This involves the following steps:

- Create a list containing what is possible to model (for example, *task*, *resource*, *neededby* from a *task* to a *resource*);
- Create a list containing what is forbidden to model (for example, *neededby* from a *quality* to an *actor*);
- Try to use the modelling with the lists of what is possible and forbidden to do;
- Create a list containing the problems identified.

The *Extender* should *Correct the modelling tool for the extension* (task 3.5.3), when there is any correction that has been identified in the test.

Finally, when there is no correction identified, the *Extender* should *Make the tool available in a link* (task 3.5.4). It is suggested that the extender uses a repository such as github (github.com) or other similar to make the tool available.

Section 5.5 presents the subprocess 4 Validate and Evaluate the iStar Extensions.

[2] istar.rwth-aachen.de/tiki-index.php.

5.5. Validate and evaluate the iStar extension (sub-process 4)

The validation identifies if the extension is valid to model systems and it is also useful to show the users how to use the extension providing an example of use. The evaluation of an extension is important to get feedback about it from other researchers objectifying to identify their opinion and corrections to improve it.

This sub-process starts after the iStar extension is developed in sub-process 3 (Section 5.4). It has six tasks, one sub-process, four artefacts. *Extension specification [Validated/evaluated]* is the main artefact generated by this sub-process and adds the results of this sub-process to the Extension specification. Fig. 8 depicts the sub-process 4 (*Validate and evaluate the iStar extension*) in BPMN. We detail each element of this sub-process in next paragraphs.

This sub-process is composed of three pairs of analysis and the related corrections of the iStar extension, with the validation being performed by two pairs and the evaluation by one pair.

The first pair of analysis (validation)/correction is performed by the *Extender* that uses the proposed iStar extension proposed to model a system (task 4.1). This task represents the usage of the iStar extension proposed in order to identify corrections and improvements in the extension. This can be done by selecting one or more case studies, modelling the scenarios and identifying and documenting limitations. The *Extender* should select non-trivial and real examples to use the iStar extension. The *Extender* should describe how to use the iStar extension, i.e., the required steps to use the proposed iStar extension. When the *Extender* identifies that something is missing or can be improved (such as a node of the extension be a target or source of a proposed link, or there is a need to change the place of a construct's label), the respective corrections are performed (task 4.2).

The second pair of analysis (validation/correction) is performed by the sub-process 4.3 (*Consult experts*). It concerns with the submission of the iStar extension for the analysis of the experts (in iStar extensions and domain/application area), reception and analysis of the feedback. The *4.3. Consult experts* sub-process is detailed in Section 5.5.1.

The *Extender* may *Apply the corrections/improvements from the experts* (sub-process 4.4) to the iStar extension. It is necessary to check whether the changes do not introduce further inconsistency, incompleteness or conflicts problems.

The third pair of the analysis, evaluation/correction, consists of the pair of the tasks 4.5 (*Evaluate the iStar extension*) and 4.6 (*Apply the improvements from the evaluation*). This pair is performed if it is feasible to evaluate the iStar extension. The feasibility analysis should be done by the *Extender* considering the criteria of the availability of participants to the selected kind of evaluation and the chronogram of the research. The evaluation (task 4.5) can be done by an experiment, qualitative study or survey. An experiment can be based on Wohlin et al. (2012) or Juristo and Moreno (2001). The extender can plan a qualitative study based on Merriam (2009). Finally, a survey can be performed following the steps suggested by Kitchenham and Pfleeger (2002).

Note that Task 4.1 is concerned with the use of the extension by the Extender, allowing his/her own analysis and illustration. The objective of tasks 4.1 and sub-process 4.3 is to identify corrections/improvements to be done. Task 4.5 is concerned with the usage of the extension or its analysis by a group of participants and/or experts, its objective is to compare the modelling of the extension with other existing extensions.''.

When the evaluation identifies improvements, these should be applied (task 4.6). Finally, the *Extension specification [Validated/evaluated]* is generated by the *Extender*.

The next section presents the sub-process 4.3 Consult Experts.

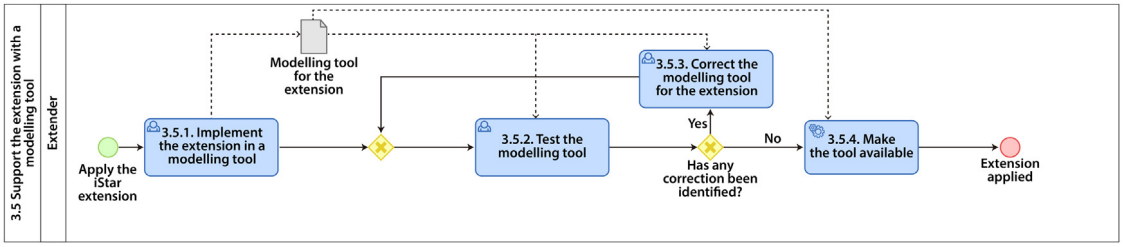


Fig. 7. Sub-process 3.5 Support the extension with a modelling tool.

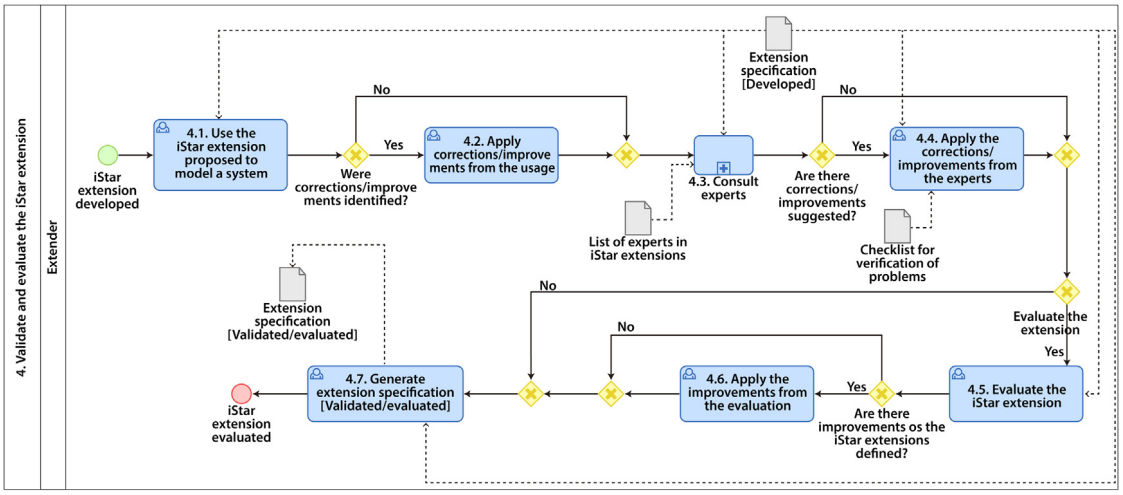


Fig. 8. Sub-process 4. Validate and evaluate the iStar extension.

5.5.1. Consult experts (sub-process 4.3)

This sub-process details the consultation of the *Expert in iStar extension* and the *Expert in domain/application area* regarding their feedback about the proposed iStar extension. Both consultations are made in parallel. They are detailed in next paragraphs of this section. Fig. 9 shows the sub-process 4.3 in BPMN.

The *Extender Consults experts in iStar extensions* (task 4.3.1) to identify corrections in the approach. This task involves the following steps:

- Identify the experts to be contacted and their e-mails in the List of experts of iStar extensions;
- Send an e-mail asking about corrections with the extension specification attached. Or contact personally, showing the extension specification and talking about corrections.

For example, the *Researcher* can ask the experts if the iStar extension is useful for modelling requirements, about the relationship with the existing concepts, about how the concepts were included in iStar (Stereotypes, new graphical representation) and the graphical representation used.

If the contact is not available, the Extender should search for more recent papers of the experts (e.g., dblp, Google scholar). For example, the Researcher can ask the experts if the iStar extension is useful for modelling requirements, about the relationship with the existing concepts, about how the concepts were included in iStar (Stereotypes, new graphical representation) and the graphical representation used.

The *Expert in iStar extensions* receives the description, Analyses and gives the feedback about the iStar extension proposed (task 4.3.3). The Extender receives the feedback and corrects the iStar extension (task 4.3.5) considering the feedback of an Expert in iStar extensions. There is the possibility that the extension needs to wait a lot for an answer from the expert or do not receive any response. Thus, the Extender should continue the execution of the PRISE if no answer is received from the Experts.

The *Expert in iStar extensions* receives the description, *Analyses and gives the feedback about the iStar extension proposed* (task 4.3.3). The *Extender receives the feedback and corrects the iStar extension* (task 4.3.5) considering the feedback of *Expert in iStar extensions*.

In parallel, when the iStar extension is proposed to a domain/application area (as security or robotics, for example), the *Extender* consults experts in domain/application area (task 4.3.2). This involves the following steps:

- Identify the experts to be contacted and their e-mails. It is possible to consider the List of references and researchers contacted, available in *Extension specification [Developed]*;
- Contact experts asking about corrections and with the extension specification attached.

The *Expert in domain/application area* analyses and gives the feedback about the iStar extension proposed (task 4.3.4). Finally, the feedback of the experts in application area is received (task 4.3.6). When the *Extender* do not receive any response, he/she should contact other expert in iStar in domain/application area.

The *Expert in domain/application area* analyses and gives the feedback about the iStar extension proposed (task 4.3.4). The feedback of the experts in application area is received (task 4.3.6). As mentioned in the description of tasks 4.3.3 and 4.3.5, the Extender should continue the execution of the PRISE if no answer is received.

We believe these alternatives are enough to obtain an answer about questions related to iStar or the domain/application area. However, if the *Extender* did not receive an answer, he/she should continue to the next task of the PRISE and try to mitigate the questions in other task of the process such as the evaluation

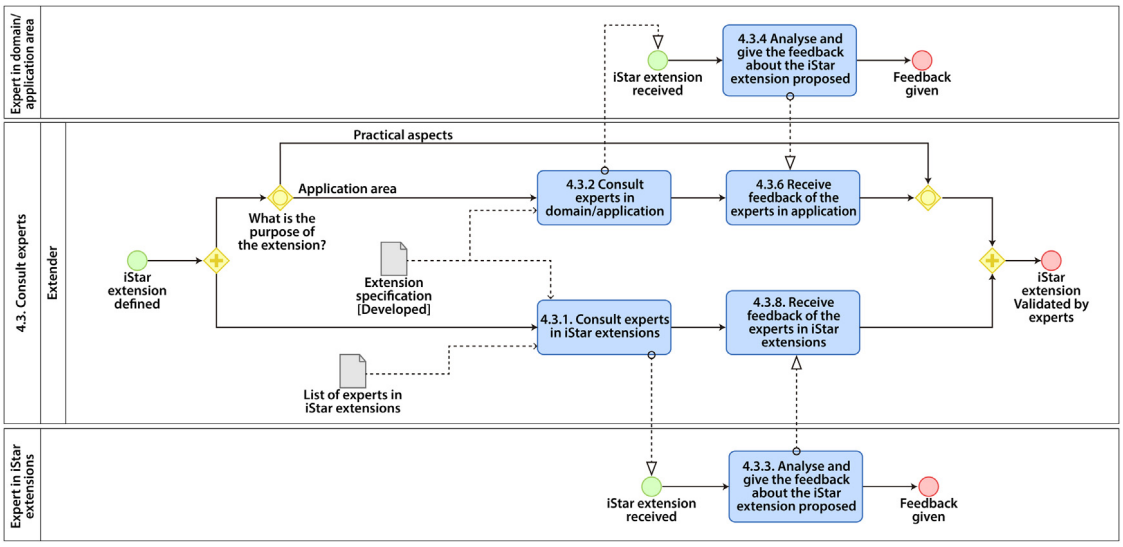


Fig. 9. Sub-process 4.3. Consult experts.

and publicising the iStar extension (sub-process 6, during the submission of the extension).

Next section details the sub-process 6 Publicise the iStar Extension.

5.6. Publicise the iStar extension (sub-process 6)

iStar extensions should be publicised to be used by others or allow its reuse. iStar extensions unavailable cannot be used.

This sub-process starts after the iStar extension is evaluated and validated in sub-process 4 (Section 5.5). It has four tasks, two artefacts and involves the *Extender* and the *Expert in iStar extensions*. The extension is accessible to the community and endorsed by experts at the end of this sub-process. Fig. 10 depicts the sub-process *6. Publicise the iStar extension*, in BPMN. We detail each element of this sub-process in the next paragraphs.

This sub-process starts with the task 6.1 (*Add the new iStar extension to the Catalogue*). It consists of updating iStar extension catalogue already referred in Section 5.2 with the insertion of the new iStar extension. The *Extender* may inform the link to the related paper or update the *Extension Specification*. It is also required to inform data about title, abstract, application area, extension base, level of extension, compatibility between metamodel and concrete syntax of extensions, metamodel completeness, concepts definition, kind of construction proposed, reasoning approach used, tool support, kind of validation, validation rules and constructs added.

When there are *Experts in the iStar extensions* participating in the proposal of the new iStar extension, they *Endorse the iStar extension* (task 6.2). Otherwise, the iStar extensions experts are notified about the extension (task 6.2) by the Extender. The Extender should contact at least one *Expert in iStar extensions* to inform the insertion of a new iStar extension in the catalogue. The contacted experts in iStar extensions can *Endorse the iStar extension* (task 6.2) if they consider it well-defined. The experts can use the PRISE TOOL (Gonçalves et al., 2019d) to endorse the extension.

In parallel, the *Extender* performs a task to *Publish the iStar extension* (task 6.4). For example, the extension can be published in a conference, journal, blog or discussion list.

Finally, this sub-process ends with the iStar extension publicised and endorsed. Consequently, PRISE ends with the extension proposed.

Next section presents additional considerations about PRISE.

5.7. Additional considerations about PRISE

PRISE is a reference process and can be customised by the extender before to be used. We illustrated this customisation in Section 6.

We created the PRISE tool (Gonçalves et al., 2019d), which the extenders can use to manage the proposal of their extensions. It is possible to create new projects of the extensions, register the advance in their proposals and update the generated artefacts. The PRISE tool is available in https://istarextensions.cin.ufpe.br/prisetool.

It presents the main process of the PRISE process and five web screens, one for each sub-process. Each screen is composed of a diagram of the process on top and a grid with the checklist of their tasks and sub-process(es). The tool displays the related artefacts as web forms and template. The templates are files which can be downloaded to be filled as an alternative way to continue developing the extension off-line. It is possible to identify the progress of the performed tasks by the tasks marked as done in the tool. Fig. 11 presents the checklist of the *Analyse the need for extension proposal* sub-process 1, we omitted the top of this screen with the diagram of this subprocess because this image is the same of Fig. 4. First column presents the status of the related task (performed or not), the name of their subprocess or task are listed, and the related artefacts are presented as input, template and output. The checklist of performed allows the identification of the progress, the name of subprocess or tasks has a tooltip with the description of each item, input and output are links to web forms and template is a file which can be downloaded to be filled and uploaded.

The BPMN diagram of the PRISE is available in https://github.com/enyo-goncalves/prise. Next section presents the customisation of PRISE.

6. Customisation of PRISE

PRISE is a reference process to support iStar extensions. Thus, it can be adjusted for the needs of removing some steps that do not make sense in a particular situation. In this section, we will illustrate a customisation of PRISE to the usage of an extender who is an expert in iStar extensions and the application area.

In this case, it is not necessary to consult an expert in iStar extensions once the role Extender is performed by someone who knows very well the modelling language. Thus, the steps 1.7, 1.8,

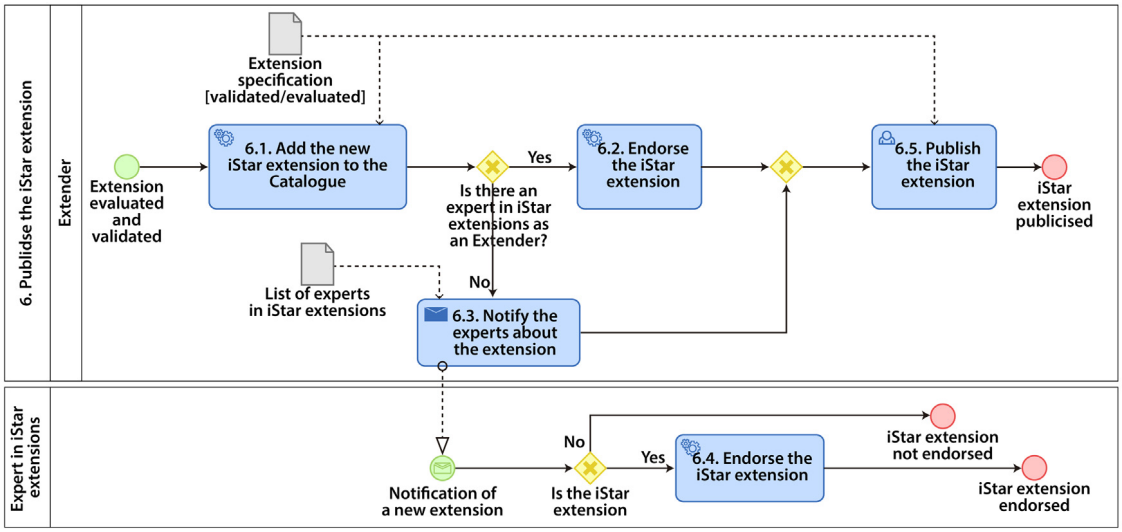


Fig. 10. Sub-process 6: Publicise the iStar extension.

Performed	Subprocess or Task	Input	Template	Output
☑	Study/Review a Domain / application area		1562267821.xls Choose File No file chosen Submit	References and researchers contacted
☐	Identify the concepts to be introduced by the extension	References and researchers contacted	1562267873.xls Choose File No file chosen Submit	Concepts to be introduced

Fig. 11. Details of checklist of the sub-process 1 Analyse the need for extension proposal.

1.9, 2.4, 2.5, 2.6, 4.3.1, 4.3.3, 4.3.5, 4.4, 6.3 and 6.4 which are related to consulting experts in iStar extensions should not be performed. Task 1.6 is required to analyse if it is possible to model the extension concepts with iStar. Thus, it is possible for an expert in iStar extensions to identify this without creating a new iStar model due to his/her previous experiences. Additionally, the task 2.3 related to analyse how to integrate the extension concepts and the iStar concepts can be omitted as a task and be introduced in the description of the tasks to define the metamodel (3.1) and concrete syntax (3.2).

Since we considered the extender also an expert in the application area, it is not necessary to consult an expert in application area once the extender is just one and know very well this field. Thus, the tasks 1.3, 1.4, 1.5, 4.3.2, 4.3.4 and 4.3.6 which are related to consulting experts in the application area should be not performed. The task 1.1 related to study an application area is not required because the extender just knows the application area very well and can easily identify the concepts to be introduced (next task which was maintained) without needing to start a literature review about the theme.

In this customisation we maintained relevant tasks which are the core of PRISE, i.e., we maintained the following principles:

- Search to existing extensions and their constructs to promote the reuse and avoid conflicts;
- Completeness by the description of the meaning of the concepts to be introduced, the metamodel and the concrete syntax of the extension.
- Verification of completeness, consistency and conflicts.
- Extension support with a tool;
- Extension validation and evaluation by an empirical evaluation;
- Make the extension public.

Fig. 12 presents the sub-processes 1, 2, 4 and 6. The main process of the PRISE and the sub-processes 3 and 3.5 were not changed in this customisation and the sub-process 4.3 was removed. The project of this customisation can be accessed in www.cin.ufpe.br/ler/prise/customisation downloaded in https://github.com/enyo-goncalves/prise.

We presented the specification of the extension mentioned in Section 7 considering the customised version of PRISE to illustrate its usage. This version can be accessed in www.cin.ufpe.br/~ler/prise/customisation/illustration.

Next section presents the validation and illustration of the PRISE.

7. Using PRISE

We recreated five existing extensions with problems to present the usage of the PRISE. In the paper (Mouratidis et al., 2013), an extension is presented to model security and privacy requirements in cloud provider modelling. An extension to model goal models in an holonic way is presented by Louaqad and El Mohajir (2014). The modelling of temporal information like preconditions and effect was explored in (Liaskos and Mylopoulos, 2010). Morales et al. (2015) presented an iStar extension to model teleo-reactive robots. Finally, the paper (Murukannaiah

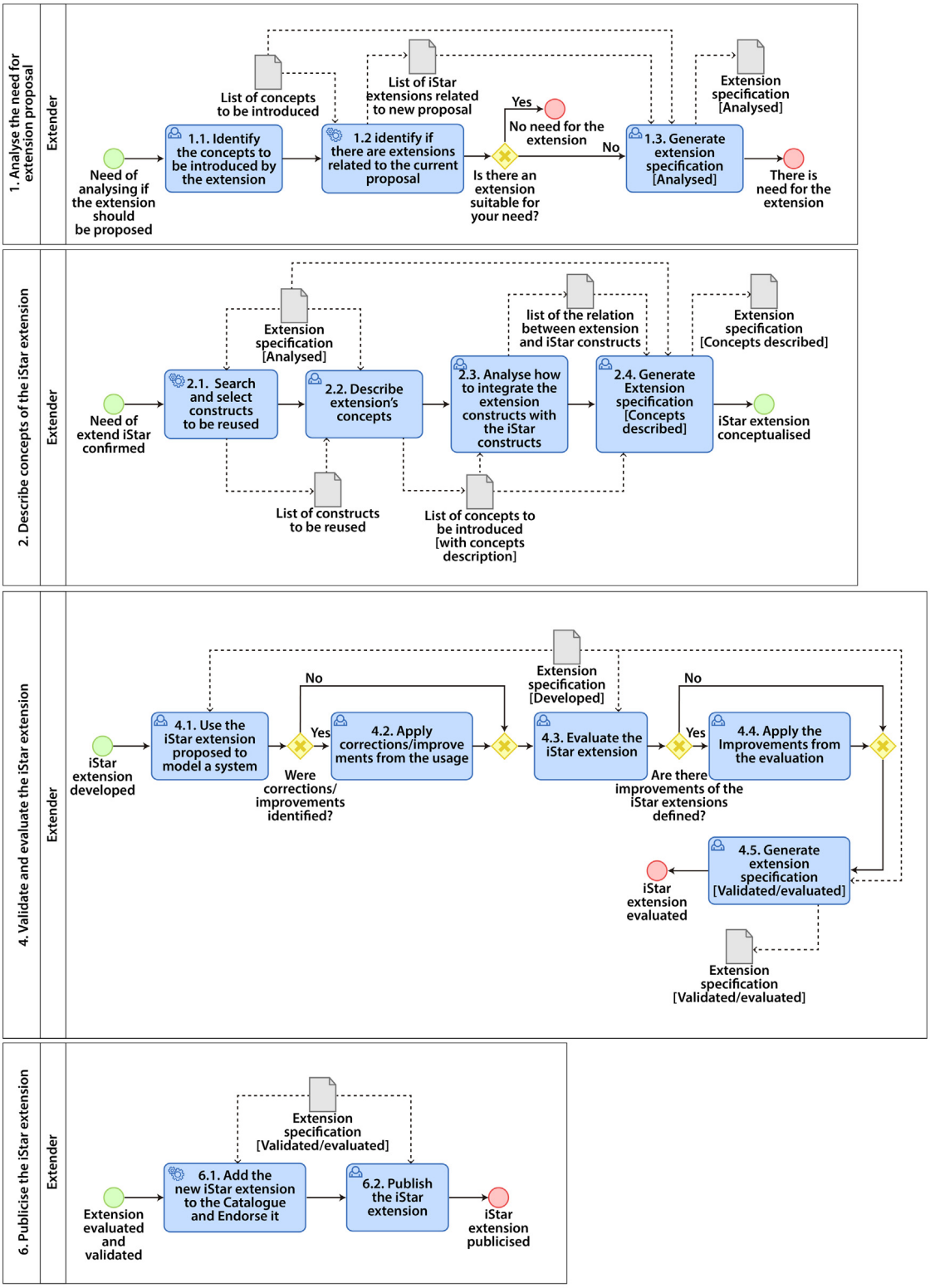


Fig. 12. A customised version of the PRISE.

and Singh, 2014) presents an iStar extension to model contextual intentional elements (such as *belief* and *resource*). The results are available in www.cin.ufpe.br/~ler/validation&illustration and show an evidence that the process is appropriate to create iStar extensions avoiding common mistakes of the existing proposals We used the complete version of the PRISE presented in Section 5.

Two new iStar extensions were proposed by Gonçalves et al. (2019b) and Ribeiro et al. (2019) using PRISE which confirms that it is a good way to support the proposal of new iStar extensions.

Gonçalves et al. (2019b) proposed a new iStar extension to model rational agents in the context of Multi-Agent Systems (MAS). Multi-agent systems (MAS) involve a wide variety of

agents that interact with each other to achieve their goals. Usually, the agents in a MAS can be reactive or proactive, this choice defines the rationale of its elements. Rational Agents is the term used to mention a set of four kinds of reactive and proactive agents. Russell and Norvig (2003) define four types of agents according to their internal structure: Simple Reflex Agent, Model-Based Reflex Agent, Goal-based Agent and Utility-Based Agent.

We represented each kind of rational agent and their roles by stereotypes (simple-reflex, model-based-reflex, goal-based and utility-based) associated with agent and agent role. Organisation, Environment, Planning, Plan and Perception are represented by new symbols. Therefore, Belief, a construct that was removed in iStar 2.0, was added as an intentional element again. We added stereotypes to represent of next-function, formulate-problem, formulate-goal and utility-function and action. The relationship neededby, which is defined in iStar 2.0, was used to link beliefs and next-function tasks. Cause/effect is a relationship included to connect perceptions and actions and connect next-function and action.

We followed PRISE to propose this extension. The results are detailed in Gonçalves et al. (2019b).

In Ribeiro et al. (2019), the authors proposed a new iStar extension to model safety concepts in safety critical systems. The authors of these two papers used the PRISE to propose these extensions and avoid the occurrence of problems that have been occurring in the existing iStar extensions.

In this section, we illustrated the usage of PRISE (not customised version) by the recreation of the iStar extension proposed by H. Mouratidis, S. Islam, C. Kalloniatis, S. Gritzalis in the following paper: *A framework to support the selection of cloud providers based on security and privacy requirements*, which was published in the Journal of Systems and Software in 2013 (Mouratidis et al., 2013). We selected it due to the number of problems of incompleteness, inconsistencies and conflicts (28 in total) identified during the execution of a previous SLR (Gonçalves et al., 2017a). The identification of these conflicts can be checked in the paper (Gonçalves et al., 2017a).

The original paper concerns the elicitation of security and privacy requirements and the selection of a cloud provider based on the satisfiability of the service provider to the relevant security and privacy requirements. The authors introduced 15 constructs in iStar to represent *Security and Privacy goal, Security and Privacy Measure, Security and Privacy Mechanism, Security and Privacy Constraint, Cloud Actor, Threat, Vulnerability, Implements, Satisfies, Restricts and Satisfaction*.

Thus, in next sections we present the results according to the *Extension specification* template artefact of the PRISE process. Additionally, we show a comparison between the original and recreated version of the selected iStar extension.

7.1. Analyse the need for extension proposal — study/review a domain/application area (task 1.1)

We studied the papers referenced in the selected iStar extension, which are probably what the author used to extract the concepts added by the extension. We did not have any issues about the domain/application area, so we did not contact these experts. The list of references is presented in Table 2.

7.2. Identify the concepts to be introduced by the extension (task 1.2)

We identified fifteen concepts to be introduced by the extension, being eleven nodes and four links. Some concepts are entity's specialisations for security and privacy such as security measure and privacy measure. Table 3 shows these results.

Table 2
List of references in the recreated istar extension.

ID_reference	Reference
REF_SPCP1	Bruening, P.J., Treacy, B.C., 2009. Cloud computing: privacy, security challenges, Bureau of Nat'l Affairs, www.hunton.com
REF_SPCP2	Cloud Threat. 2010. Top Threats to Cloud Computing Vr. 1.0, Cloud Security Alliance, https://cloudsecurityalliance.org/research/top-threats/
REF_SPCP3	Erdogmus, H., 2009. Cloud computing: does nirvana hide behind the nebula? IEEE Software 26 (2), 4–6.
REF_SPCP4	Grobauer, B., Walloschek, T., Stocker, E., 2011. Understanding cloud computing vulnerabilities. IEEE Security & Privacy Magazine 9 (2), 50–57.
REF_SPCP5	Islam, S., Mouratidis, H., Wagner, S., 2010. Towards a framework to elicit and manage security and privacy requirements from laws and regulation. In: Proceeding of Requirements Engineering: Foundation for Software Quality (REFSQ), Lecture Notes in Computer Science, vol. 6182/2010, pp. 255–261.
REF_SPCP6	Islam, S., Mouratidis, H., Weippl, E., 2012b. A goal-driven risk management approach to support security and privacy analysis of cloud-based system. In: Security Engineering for Cloud Computing: Approaches and Tools. IGI Global Publication, United States of America by (an imprint of IGI Global) 701 E. Chocolate Avenue, Hershey, PA 17033.
REF_SPCP7	Islam, S., Mouratidis, H., Jürjens, J., 2011. A framework to support alignment of secure software engineering with legal regulations. Journal of Software and Systems Modelling (SoSyM) 10 (3), 369–394.
REF_SPCP8	Rosado, D.G., Fernández-Medina, E., López, J., Piattini, M., 2010. Analysis of secure mobile grid systems: a systematic approach. Information & Software Technology 52 (5), 517–536.
REF_SPCP9	Rosado, D.G., Gomez, R., Mellado, D., Fernández-Medina, E., 2012. Security analysis in the migration to cloud environments. Future Internet 4 (2).
REF_SPCP10	Sriram, I., Khajeh-Hosseini, A., 2010. Research Agenda in Cloud Technologies, CoRR, CoRR abs/1001.3259:(2010)
REF_SPCP11	Takabi, H., Joshi, J., Ahn, G., 2010. Security and privacy challenges in cloud computing environments. In: IEEE Computer and Reliability Societies, November/December, IEEE Computer Society
REF_SPCP12	Mouratidis, Haralambos, and Paolo Giorgini. Secure tropos: a security-oriented extension of the tropos methodology. International Journal of Software Engineering and Knowledge Engineering 17.02 (2007): 285-309.

7.3. Model an example with the identified concepts using iStar (task 1.6)

We did not have any issues about the domain/application area, so we did not contact these experts. However, we were uncertain if it was possible to model some concepts, such as security goals and privacy goals, with the iStar without doing the extension. Consequently, we tried to model part of the example presented in the original proposal of this extension (Mouratidis et al., 2013) and wrote some observations.

The main goal of this task is to try to identify the domain concepts which is possible to represent with iStar without creating new representations and what is not possible to do so.

Table 3
List of concepts to be introduced in the recreated istar extension.

ID_concept	Name of the concept	Reference
Nodes		
CON_DEC_01	Security goal	REF_SPCP5
CON_DEC_02	Privacy goal	REF_SPCP5
CON_DEC_03	Security Measure	REF_SPCP5, REF_SPCP10
CON_DEC_04	Privacy Measure	REF_SPCP5, REF_SPCP10
CON_DEC_05	Security Mechanism	REF_SPCP9, REF_SPCP10
CON_DEC_06	Privacy Mechanism	REF_SPCP9, REF_SPCP10
CON_DEC_07	Security Constraint	REF_SPCP5, REF_SPCP12
CON_DEC_08	Privacy Constraint	REF_SPCP5, REF_SPCP12
CON_DEC_09	Cloud Provider Actor	REF_SPCP5, REF_SPCP9
CON_DEC_10	Threat	REF_SPCP2, REF_SPCP10
CON_DEC_11	Vulnerability	REF_SPCP2, REF_SPCP4, REF_SPCP10
Links		
CON_DEC_12	Implements	REF_SPCP9
CON_DEC_13	Satisfies	REF_SPCP9
CON_DEC_14	Restricts	REF_SPCP9
CON_DEC_15	Satisfaction	REF_SPCP9

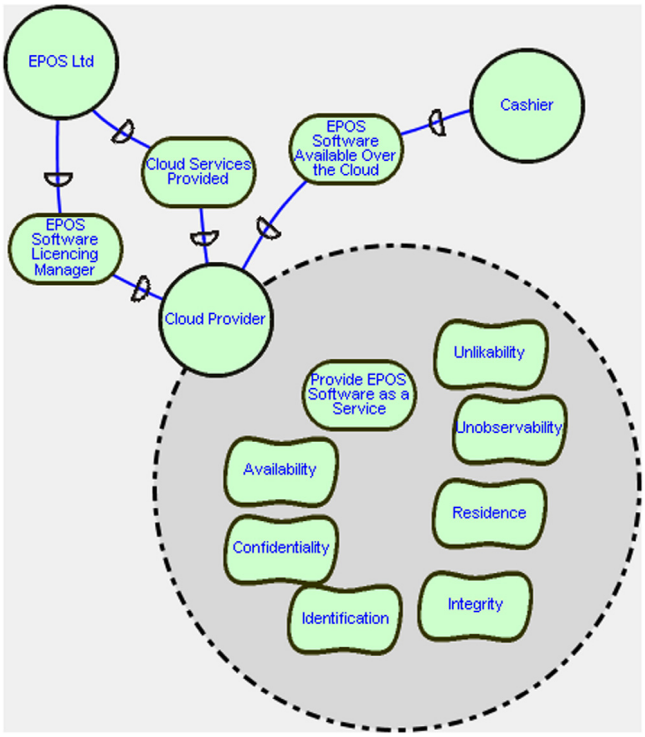


Fig. 13. Modelling with the concepts of the recreated iStar extension.

EPOS is the reduced form of the company Electronic Point Of Sales. We have identified three actors that are relevant to this example: EPOS Ltd, EPOS Software and Cashier. The EPOS Ltd depends on cloud services be provided and EPOS software Licencing managed. The Cashier depends on the EPOS software being available.

In doing so, the Cloud actor has the goal to Provide EPOS Software as a service and ensure the Data confidentiality, Data residence and Availability of Software. Furthermore, the security and privacy properties of Availability, Confidentiality, Identification, Integrity, Residence, Unobservability and Unlikability are required to this actor.

The model is presented in Fig. 13 and the observations are described next.

Observations

Table 4
List of the search results of the recreated istar extension.

Term	Extensions	Year
Cloud	A framework to support selection of cloud providers based on security and privacy requirements (*)	2013
	Security Requirements Engineering for Cloud Computing: The Secure Tropos Approach (*)	2016
Security	A framework to support selection of cloud providers based on security and privacy requirements (*)	2013
	A vulnerability-centric requirements engineering framework: analysing security attacks, countermeasures, and requirements based on vulnerabilities	2010
	Improving Risk-Based Security Analysis with i*	2011
	Integrating Security Patterns with Security Requirements Analysis Using Contextual Goal Models (*)	2014
	Model Based Process to Support Security and Privacy Requirements Engineering	2012
	Modelling security requirements through ownership, permission and delegation	2005
	Secure Tropos: A Security-Oriented Extension of the Tropos Methodology	2007
	Security and privacy requirements analysis within a social setting	2003
	Security and Trust Requirements Engineering	2005
	Security Requirements Engineering for Cloud Computing: The Secure Tropos Approach (*)	2016
	Security requirements engineering via commitments	2011
Privacy	A framework to support selection of cloud providers based on security and privacy requirements (*)	2013
	Model Based Process to Support Security and Privacy Requirements Engineering	2012
	Privacy-Aware Trust Negotiation (*)	2016
	Security and privacy requirements analysis within a social setting	2003
Service	A new service-based approach for enterprise modelling	2013
	Considering Technology Representation in Service-Oriented Business Models	2011
	Exploring Web Services from a Business Value Perspective	2005
	Modelling and Reasoning about Service-Oriented Applications via Goals and Commitments	2010

The Security and Privacy goals should be specialisations of *Quality*. Quality restrictions were modelled when trying to use iStar without the extension as Quality, since they were linked to characteristics such as confidentiality and availability, for example. Representing threats, measure and relationships were difficult. The modelling of direct dependence linking EPOS Ltd to the *Provide EPOS as Service* objective is not possible in iStar, as it lacks a dependum. Also, we felt the need to specialise the actors that represent cloud actor, we believe that the representation indicated by the researcher is sufficient. We consider that *Security/Privacy Mechanism* should not specialise *task*, since the description of these mechanisms are not similar to tasks (see the example of ACID mechanisms, Log data and checkPoint; we do not see them as being a type of task). Modelling relationships is necessary as a specialisation of the contribution relationship. The modelling of security and privacy objectives is necessary to be done through a textual mark, however these should be *Qualities*.

There is no need to create a relationship between these entities and the cloud actor. We consider that only adding these

Table 5
List of concepts to be reused.

ID_concept	Name of the concept	Reference of the extension to be reused
Nodes		
CON_DEC_01	Security goal	Li, T., Horkoff, J., Mylopoulos, J., 2014. Integrating security patterns with security requirements analysis using contextual goal models. In: The Practice of Enterprise Modelling Working Conference
CON_DEC_07	Security Constraints	Mellado, D., Mouratidis, H., Fernandez-Medina, E., 2014. Secure Tropos framework for software product lines requirements engineering. Comput. Stand. Interfaces
CON_DEC_10	Threat	Islam, S., Mouratidis, H., Kalloniatis, C., Hudic, A., Zechner, L., 2012. Model based process to support security and privacy requirements engineering. International Journal of Secure Software Engineering
CON_DEC_11	Vulnerability	Two extensions: Elahi, G., Yu, E., Zannone, N., 2010. A vulnerability-centric requirements engineering framework: analysing security attacks, countermeasures, and requirements based on vulnerabilities. Requirements Engineering Journal Dubois, E., Mayer, N., Rifaut, A., 2011. Improving risk-based security analysis with i∗. Social Modelling for Requirements Engineering Book.

entities inside the boundary makes it part of the cloud. Finally, we consider that there is need to extend the iStar to represent the list of domain concepts.

7.4. Search if there is an extension that considers your proposal (task 1.10)

We searched the catalogue of iStar extensions for some terms and some extensions were found. As we are recreating an existing iStar extension, we added the year of publication to identify the papers published after the extension selected. The list is presented in Table 4, we listed all extensions found. Extensions proposed after the publication of selected extension are highlighted by an asterisk (*). We have not considered the article itself and articles after 2013 at this stage. Thus, we can conclude that there was no extension proposed for this purpose.

The next step of the flow is the gateway: *Is there an extension suitable for your need?* We did not find an appropriate existing extension. Thus, the extension specification [Analysed] is generated with the join of Tables 1–3, Fig. 13 and Observations.

7.5. Search and select constructs to be reused (task 2.1)

We searched the repository of iStar extensions by the name of the concepts to be introduced in Table 3. The list of the constructs found is shown in Table 5.

We found some constructs with the name of the *measure*, but they are not conceptually equivalent to the concept to be added by the extension. We believe the authors reused the existing threat representation as a basis, however, withdrew the textual markup (s).

7.6. Describe extension's concepts (task 2.2)

We identified the description of each concept identified in Table 3. The list of the concepts with their descriptions and the references where the descriptions were identified is shown in Table 6.

7.7. Analyse how to integrate the extension constructs with the iStar constructs (task 2.3)

We tried to relate the concepts to be introduced with the iStar standard concepts. Thus, we concluded that we could specialise six iStar constructs. Cloud Actor represents companies which provide cloud services, so it is a kind of Actor. Security and privacy goals are kinds of goal. Finally, Satisfies, Restricts and Implements are similar to values *help*, *make*, *break* and *hurt* which add information about the kind of contribution is established between two intentional elements. The list of the relations between the extension and iStar constructs is given in Table 7.

Measure and Constraint can be represented as a kind of intentional element. The Other constructs to be introduced by this extension (*Threat*, *Vulnerability* and *Satisfaction*) are represented with new metaclasses without specialising iStar nodes or links and they are not intentional elements.

7.8. Generate extension specification [Concepts described] (task 2.7)

As described in the last paragraph of Section 5.1, when the experts in iStar participate in the iStar extension, they play the role of *Extender*. Thus, we have two experts in iStar extensions participating as *Extenders*. Consequently, we did not need to contact external experts in iStar extensions to mitigate issues about how to integrate (tasks 2.4, 2.5 and 2.6). Finally, the *Extension specification [Concepts described]* was generated with the integration of the *Extension specification [Analysed]* and the results of Tables 4–6.

7.9. Define metamodel of extension (task 3.1)

We included the concepts identified in the List of Concepts to be introduced in the iStar metamodel. We considered the relationship of extension's constructs and iStar constructs in this step. The result is presented in Fig. 14.

The new metaclasses *Threat*, *Vulnerability*, *Measure*, *Constraint* were created without specialising the existing iStar constructs. They were grouped as intentional elements. *Cloud provider* is a new metaclass that specialises *Actor*. *Mechanism* is a new metaclass which specialises *Task*. The relationship of *Satisfaction* is represented by a new relation between *Quality* and *Cloud provider*. We created the Security/privacy values which are applied to the *Quality*, *Measure*, *Constraint* and *Mechanism* to represent the variations of them to Security and privacy. *Implements*, *Satisfies* and *Restricts* are new values added to the *Contribution type*.

Contribution Type and Security/Privacy Values represent Enumerators with values which are related to *Contribution links* and *Quality*, *Constraint* and *Mechanism*, respectively. In an Ecore notation, for example, the relation between this kind of element with its related entities are not visual. However, to illustrate this relation we used dotted lines between these two enumerators and its related entities.

It was not necessary to add validation rules in this extension (task 3.2).

Table 6
List of concepts to be introduced with the concept's description.

ID_concept	Name of the concept	Description	Reference
Nodes			
CON_DEC_01	Security goal	The main focus of security goals is to ensure critical security properties such as confidentiality, integrity, availability, authenticity, and non-repudiation as well as the privacy goals within the overall system environment. They are classified based on a privacy taxonomy.	REF_SPCP5
CON_DEC_02	Privacy goal	It defines the goals of the privacy legislation.	REF_SPCP5
CON_DEC_03	Security Measure	It is a measure to ensure security protection.	REF_SPCP5, REF_SPCP10
CON_DEC_04	Privacy Measure	It is a measure to ensure privacy protection and to protect from any accidental and unlawful activities.	REF_SPCP5, REF_SPCP10
CON_DEC_05	Security Mechanism	They are the most appropriate mechanisms used to implement certain security services for the Cloud.	REF_SPCP9, REF_SPCP10
CON_DEC_06	Privacy Mechanism	They are the most appropriate mechanisms used to implement certain privacy services for the Cloud.	REF_SPCP9, REF_SPCP10
CON_DEC_07	Security Constraint	Constraints can represent a set of restrictions that do not permit specific actions to be taken or prevent certain objectives from being achieved and more often are integrated into the specification of existing textual descriptions. This concept is related to constraints of security.	REF_SPCP5, REF_SPCP12
CON_DEC_08	Privacy Constraint	Constraints can represent a set of restrictions that do not permit specific actions to be taken or prevent certain objectives from being achieved and more often are integrated into the specification of existing textual descriptions. This concept is related to constraints of security.	REF_SPCP5, REF_SPCP12
CON_DEC_09	Cloud Provider Actor	It is a concept to represent the providers of cloud services.	REF_SPCP5, REF_SPCP9
CON_DEC_10	Threat	A loss event occurs when a vulnerability is successfully exploited.	REF_SPCP2, REF_SPCP10
CON_DEC_11	Vulnerability	The Vulnerability is a prominent risk factor. ISO 27005 defines risk as "the potential that a given threat will exploit vulnerabilities of an asset or group of assets and thereby cause harm to the organisation", measuring it.	REF_SPCP2, REF_SPCP4, REF_SPCP10
Links			
CON_DEC_12	Implements	Relationships between mechanism and measure, where a mechanism implements a measure.	REF_SPCP9
CON_DEC_13	Satisfies	Relationships between measures and constraints, where a measure satisfies a constraint.	REF_SPCP9
CON_DEC_14	Restricts	A constraint restricts a goal.	REF_SPCP9
CON_DEC_15	Satisfaction	Relationships between security goals or privacy goals and cloud providers, where a goal satisfies cloud provider.	REF_SPCP9

7.10. Define concrete syntax of extension (task 3.3)

The graphical representations of extension's constructs are presented in Table 8. We used the iStar extensions catalogue to identify the representation of reused constructs. Once there were two representations, we considered the results of prioritisation experiment (Gonçalves et al., 2019c) to choose the graphical representation to vulnerability and satisfaction. We reused four graphical representation of existing iStar extensions. We maintained the symbols proposed in the original version of this extension to represent mechanisms, Measure, Mechanism, Cloud actor and the Implements, Satisfies and Restricts links. The security and privacy goals were represented by qualities added of textual marker to security or privacy representations. Textual markers were also applied to Measure, Mechanism and constraint.

Table 7
List of concepts to be reused.

ID_concept	Name of the concept	Relation with iStar constructs
Nodes		
CON_DEC_09	Cloud Provider	Specialises Actor
CON_DEC_05 and CON_DEC_06	Mechanism	Specialises Task
CON_DEC_07	Security goal	Specialises Quality
CON_DEC_08	Privacy goal	Specialises Quality
Links		
CON_DEC_12	Implements	Specialises Contribution link
CON_DEC_13	Satisfies	Specialises Contribution link
CON_DEC_14	Restricts	Specialises Contribution link

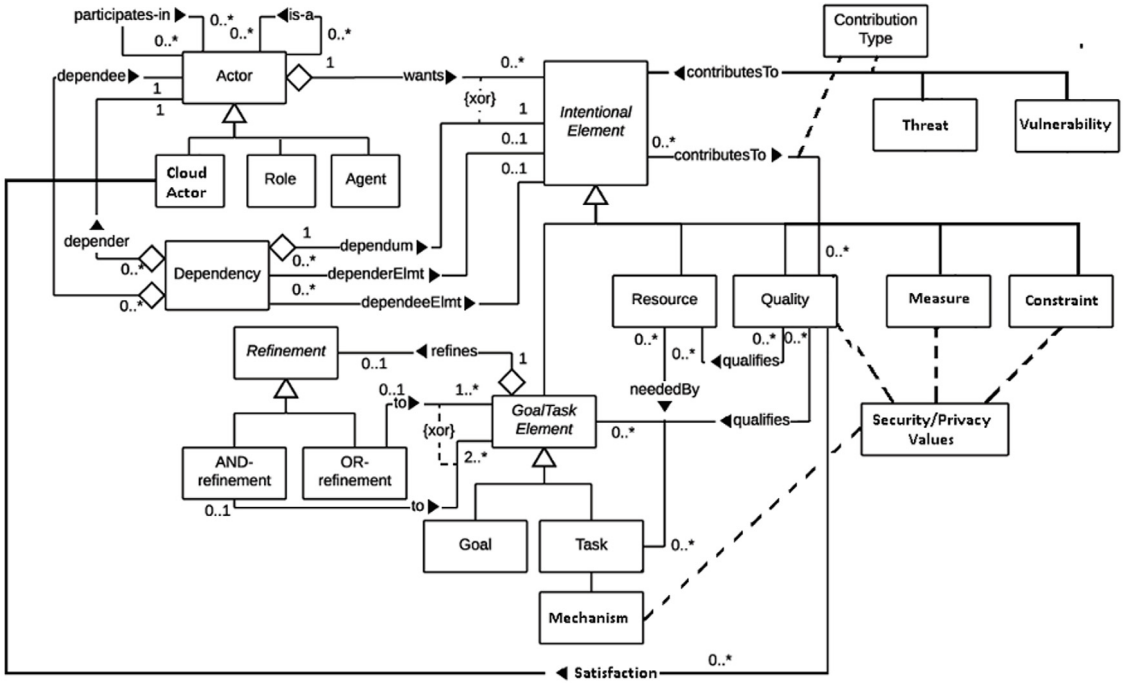


Fig. 14. iStar extended metamodel.

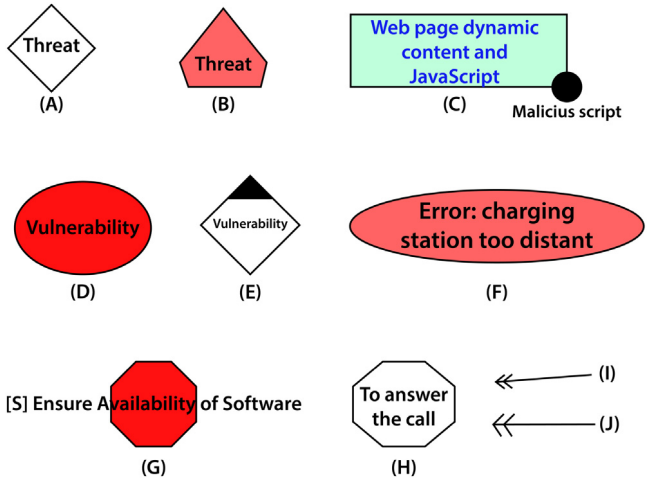


Fig. 15. Graphical representations in conflict with the recreated iStar extension.

7.11. Check and correct problems of completeness, consistency and conflicts (task 3.4)

The problems of the original version of the iStar extension (Mouratidis et al., 2013) are described in next paragraphs and illustrated in Fig. 15.

A diamond (Fig. 15-A) is used in Dubois et al. (2011) to represent *Threat* related to vulnerability and a Pentagon (Fig. 15-B) is used in Mouratidis et al. (2013) to represent a threat as a node related to security.

Vulnerability is represented in three different ways in iStar extensions. In paper (Elahi et al., 2010) it is represented as a black circle attached to an intentional element and the vulnerability has a label outside to inform its description (Fig. 15-C). In paper (Mouratidis et al., 2013), vulnerability is represented by an ellipse (Fig. 15-D), while in paper (Dubois et al., 2011) a diamond with the upper part painted in black (Fig. 15-E) is used to represent vulnerability.

The octagon is used for security extensions (such as Mellado et al., 2014 and Mouratidis et al., 2013) to represent the security and vulnerability restrictions (Fig. 15-G) and it is also used by the extension (Murukannaiah and Singh, 2014) to represent plan (Fig. 15-H). The ellipse is used to represent error (Fig. 15-D) and vulnerability in paper (Morandini et al., 2015) (Fig. 15-F).

In papers Mellado et al. (2014) and Mouratidis et al. (2013), a new link notation is introduced with different meanings. In Mouratidis et al. (2013) it indicates a *Satisfaction link* (Fig. 15-I) while in paper (Mellado et al., 2014; Zave, 1997) it indicates that an actor has a security/privacy property (Fig. 15-J).

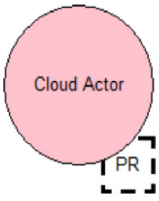
The ellipse is used to represent error in Morandini et al. (2015) (Fig. 15-F) and vulnerability (Fig. 15-D) in paper (Mouratidis et al., 2013). The ellipse is one way used by iStar to represent belief, but this construct is used to represent vulnerability in paper (Mouratidis et al., 2013) (Fig. 15-D).

We analysed the recreated iStar extension proposal according to the *Checklist for verification of problems*. We identified no problem in this proposal. The checklist filled is presented in Table 9.

We summarised the results and compared the two versions (original and recreated) of the selected iStar extension (Mouratidis et al., 2013) according to Problems of Incompleteness (PIM), Problems of Inconsistencies (PIN) and Problems of Conflicts (PCO). The nine criteria used are listed below:

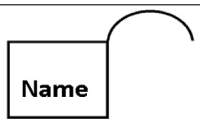
- PIM1: Non-definition of the meaning of the constructors introduced;
- PIM2: Non-representation of the extension in the abstract syntax;
- PIN1: Inconsistencies with iStar default syntax due absence of nodes and links;
- PIN2: Inconsistencies between the abstract syntax and concrete syntax;
- PCO1: One concept with two or more representations in concrete syntax;
- PCO2: Two or more concepts with only one construct in concrete syntax;

Table 8
List of concrete syntax representation of the extension.

ID_concept	Name of the concept	Graphical representation	Explanation about the graphical representation	Reused?
Nodes				
CON_DEC_01	Security goal		The *Quality* was specialised to represent *security goals*.	Yes
CON_DEC_02	Privacy goal		The *Quality* was specialised to represent *privacy goals*.	No
CON_DEC_03	Security Measure		The Figure was proposed as a polygon with many concavities to represent the measurement. The security is represented with (S) to differentiate it from privacy.	No
CON_DEC_04	Privacy Measure		The figure was proposed as a polygon with many concavities to represent the measurement. The privacy is represented with (P) to differentiate it from security.	No
CON_DEC_05	Security Mechanism		*Task* was specialised to represent *Security Mechanism*.	No
CON_DEC_06	Privacy Mechanism		*Task* was specialised to represent Privacy Mechanism	No
CON_DEC_07	Security Constraint		*Constraint* of secure Tropos was reused	Yes
CON_DEC_08	Privacy Constraint		*Constraint* of secure Tropos was reused	Yes
CON_DEC_09	Cloud Actor		Actor was specialised to represent *Cloud Actor*. PR represents provider.	No
CON_DEC_10	Threat		It is a pentagon that represents the tip of a spear. The colour purple was used also to represent the *Threat*.	Yes

(*continued on next page*)

Table 8 (*continued*).

ID_concept	Name of the concept	Graphical representation	Explanation about the graphical representation	Reused?
CON_DEC_11	Vulnerability		It was represented as square with an arch (such as a padlock opened) to represent the *Vulnerability*.	No
Links				
CON_DEC_12	Implements		We added a label *to Contribution link* to represent *Implements relationship*	No
CON_DEC_13	Satisfies		We added a label to *Contribution link* to represent *Satisfies relationship*	No
CON_DEC_14	Restricts		We added a label to *Contribution link* to represent *Restricts relationship*	No
CON_DEC_15	Satisfaction		It represents the satisfaction relationship by the representation of the text OK from the *Quality* satisfied to the *Cloud Actor*.	No

- PCO3: New Constructs in conflict with the iStar default syntax;
- PCO4: Wrong representation of iStar default syntax construct;
- PCO5: Representation of constructs that are not part of the extension.

Table 10 presents the results of the comparative analysis between the two versions, the one proposed by Mouratidis et al. (2013) and the other following the PRISE Process. We can identify that the original version has five PIM1, thirteen PIN1, four PIN2, two PCO1, three PCO2 and one PCO3. The version of the selected iStar extension recreated with PRISE, did not present these problems.

7.12. Generate extension specification [Developed] (task 3.6)

We only recreated an existing iStar extension to illustrate the use of the PRISE process. So, we did not apply the iStar extension to a modelling tool (task 3.5). Consequently, we generated the *Extension specification [Developed]*, adding the metamodel, concrete syntax representation and *Checklist verification to the Extension specification [Concepts described]*.

7.13. Use the iStar extension proposed to model a system (task 4.1)

We described how to use the iStar extension: it is necessary to relate the security and privacy qualities with the cloud providers by Satisfaction relationship. The threats and vulnerabilities should be represented and related to the goals. Next, the constraints of security and privacy should be represented in the model and related to the existing goals by restriction. Privacy and security measure should be represented and related to the existing constraints. Finally, security and privacy mechanisms should be represented and related to the measures.

A summarised description of the system in Mouratidis et al. (2013) is given as follows: the aim of the project was to develop a cloud-based solution that will enable the company to distribute its EPOS software over the cloud and also manage remotely issues such as licencing and maintenance. EPOS Ltd anticipates that a cloud-based solution will reduce their costs (especially maintenance costs), will attract more customers due to the flexibility that can be applied on licencing agreements (Mouratidis et al., 2013).

The model of the recreated iStar extension is presented in Fig. 16. The difference in graphical representation between the original and recreated iStar extension is highlighted next. We have added the textual marker *cp* to the representation of the Cloud Provider. We also have added security and privacy goals to the model and represented their satisfaction by the Cloud Provider (we used the new representation for satisfaction). We added textual markers applied to Measure and Mechanism to represent their kind (security or privacy). We represented the Vulnerability Insecure APIs with a diamond with the up edge in black.

7.14. Execution of the other steps of the PRISE

We did not identify the corrections during the usage of the extension. We consulted an external *Expert in iStar extension* and an *Expert in domain/application area*, but they did not request changes.

We only recreated an existing iStar extension to illustrate PRISE. So, we did not *Evaluate the iStar extension* (task 4.5). Consequently, we generated the *Extension specification [Validated/evaluated]*, adding Modelling of EPOS system to the *Extension specification [Developed]*.

The recreated iStar extension was endorsed by two experts in iStar extensions and this paper is the final step of the PRISE (Publish the iStar extension).

Time and effort to recreate this extension were not analysed. We also do have not the values of the time and effort to create the original version. Thus, it is not possible compare the creation of an iStar extension with PRISE and the creation without PRISE. It can be a threat to the conclusion of the usage of PRISE.

8. Evaluation of the PRISE

Wohlin et al. (2012) define two main categories of validation, i.e., dynamic and static validations. Static validation does not require that the new solution be used, it may be done through a

Table 9
Checklist of completeness, consistency and conflicts.

Completeness			
Which level the iStar extension was applied [X] Concepts definition [X] Metamodel [] Well-formedness rules (optional) [X] Concrete syntax			
Consistency			
Concept	Concept's definition	Metamodel	Concrete syntax
Security goal	[**X**]	[**X**]	[**X**]
Privacy goal	[**X**]	[**X**]	[**X**]
Security Measure	[**X**]	[**X**]	[**X**]
Privacy Measure	[**X**]	[**X**]	[**X**]
Security Mechanism	[**X**]	[**X**]	[**X**]
Privacy Mechanism	[**X**]	[**X**]	[**X**]
Security Constraint	[**X**]	[**X**]	[**X**]
Privacy Constraint	[**X**]	[**X**]	[**X**]
Cloud Provider Actor	[**X**]	[**X**]	[**X**]
Threat	[**X**]	[**X**]	[**X**]
Implements	[**X**]	[**X**]	[**X**]
Satisfies	[**X**]	[**X**]	[**X**]
Restricts	[**X**]	[**X**]	[**X**]
Implements	[**X**]	[**X**]	[**X**]

Presence of nodes and links of iStar default syntax			
Construct		Metamodel	Concrete syntax
Goal		[**X**]	[**X**]
Quality		[**X**]	[**X**]
Resource		[**X**]	[**X**]
Task		[**X**]	[**X**]
Actor		[**X**]	[**X**]
Agent		[**X**]	[**X**]
Role		[**X**]	[**X**]
Refinement		[**X**]	[**X**]
Qualification		[**X**]	[**X**]
Contribution (make, help, break, hurt)		[**X**]	[**X**]
NeededBy		[**X**]	[**X**]
Dependency		[**X**]	[**X**]
Association links	is-a	[**X**]	[**X**]
	Participates-in	[**X**]	[**X**]

Conflicts				
Concept	Conflicts			
	One construct with two or more Symbols	Two of more constructs with only one Symbol	Wrong representation of iStar constructs	Construct which is not part of the extension
Security goal	[0]	[0]	[0]	[0]
Privacy goal	[0]	[0]	[0]	[0]
Security Measure	[0]	[0]	[0]	[0]
Privacy Measure	[0]	[0]	[0]	[0]
Security Mechanism	[0]	[0]	[0]	[0]
Privacy Mechanism	[0]	[0]	[0]	[0]
Security Constraint	[0]	[0]	[0]	[0]
Privacy Constraint	[0]	[0]	[0]	[0]
Cloud Provider Actor	[0]	[0]	[0]	[0]
Threat	[0]	[0]	[0]	[0]
Implements	[0]	[0]	[0]	[0]
Satisfies	[0]	[0]	[0]	[0]
Restricts	[0]	[0]	[0]	[0]
Implements	[0]	[0]	[0]	[0]
Details about the problem(s): Problems were not identified.				

presentation of the candidate solution followed by the analysis. On the other hand, the dynamic validation is an approach in which the new solution is used in a project and the results are analysed using a case study, for example.

The validation of the PRISE is based on two approaches. The first one is a static approach with experts in iStar extensions and the second one is a dynamic evaluation with a novice.

8.1. Evaluation by the experts

[Objective] The purpose of this evaluation is to identify the point of view of the experts in iStar extensions about PRISE. It will contribute to the evaluation of a reference process to guide other iStar extensions.

[Study Design] We used mixed methods research (Creswell, 2014) to analyse the opinion of the experts in iStar extensions about PRISE using a qualitative study based on interviews and a quantitative study based on a survey. We choose to use qualitative and quantitative methods together to gain a more complete understanding of their opinion.

Our qualitative study allowed us to capture valuable evidence about the opinion of the experts about PRISE. Additionally, we performed a survey, different from the ones who participated in the qualitative study, to evaluate their opinion about PRISE. By doing so, we carried out a triangulation of the findings.

[Population] The universe of this research (population) consists of authors of iStar extensions. We considered the list of

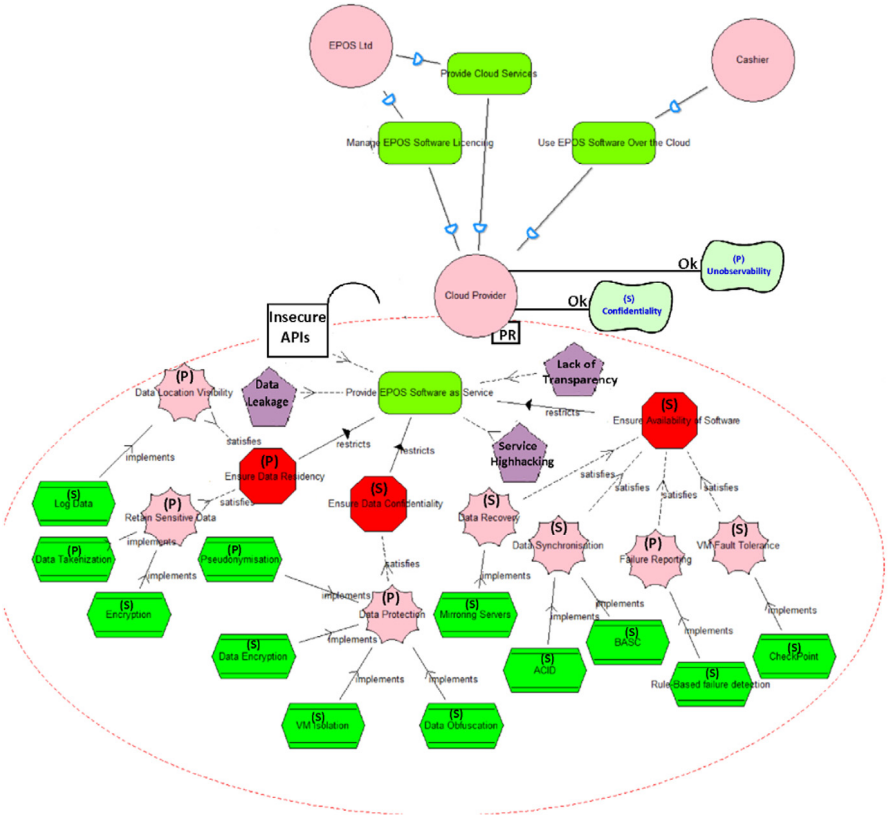


Fig. 16. Modelling of EPOS system with the recreated iStar extension.

authors of iStar extensions presented in paper (Gonçalves et al., 2017a). Thus, our universe (Population) consists of 153 authors of papers from 75 different universities describing iStar extensions.

[Sample of Participants] In the qualitative study, we used a non-probabilistic sample whose purpose was to choose richer cases for study (Merriam, 2009). We selected a sample of the researchers with more experience with iStar and iStar extensions. Therefore, we invited five authors from the list of authors of iStar extensions (Gonçalves et al., 2017a). We reached the saturation of the interviews with these three participants, and it was unnecessary to interview the other two. The saturation was reached since we received similar answers of the three participants, as a kind of consensus/common sense, they diverged only few and not relevant points. Consequently, we invited the other 148 authors to participate in the survey.

We received a total of 20 responses of which 8 from Brazil, 1 from Germany, 3 from Italy, 2 from Mexico, 1 from Portugal, 1 from Canada, 1 from Chile, 1 from USA and 2 from the United Kingdom. Concerning the educational degree of the participants, we have 17 PhDs, 1 PhD Student, 1 master (participant who finished the master course and is working in a company) and 1 master student. Concerning their current work/research positions we have 14 professors, 2 consultants, 1 requirements engineer and 3 students. They are from 18 different universities. Their experience with the usage of iStar is summarised as follows: 5 experts, 8 advanced, 5 intermediate and 2 emerging. Their experience with the creation and extension of modelling languages consists of: 5 experts, 9 advanced, 4 intermediate and 2 emerging. Finally, concerning their experience with the creation of iStar extensions: 3 experts, 8 advanced, 6 intermediate and 3 emerging.

[Collection Preparation] Clarification and consent terms were sent to participants before each interview session. Semi-structured interviews were conducted using an interview script with open questions. The complete version of the script of the interviews is available in Table 11. A survey was submitted with a set of multiple-choice questions and open questions. The survey is presented in Table 12.

Table 10
Comparative analysis between the two versions of the recreated iStar extension.

Evaluation criteria	Original	Recreated
PIM1	5	0
PIM2	0	0
PIN1	13	0
PIN2	4	0
PCO1	2	0
PCO2	3	0
PCO3	1	0
PCO4	0	0
PCO5	0	0

The script interview and survey were validated by the second and the third authors of this paper. We tested the script interview and the survey with a computer science professor from Universidade Federal do Ceará. This professor did not propose any iStar extension but has experience in requirements engineering and empirical research. They were also tested previously by a pilot with another expert in iStar extensions who helped to improve it. The pilot data were not used during the analysis.

The participants of the qualitative study watched a video about PRISE before the interviews. This video was used in the first step of the survey. This video is available in https://youtu.be/_LF4u-MOsD0.

[Data Collection] Interviews and Survey were conducted in English. The interviews were conducted via Skype individually with each participant and the Survey was submitted by the Google forms during June and August 2018. Each interview was conducted by one author of this paper, while a second author took

Table 11
Script interview of the qualitative study with experts (interviews).

1. What is lacking for researchers to create iStar extensions more systematically? What do you miss?
2. What do you and other researchers do to develop extensions more systematically?
3. Do you know another process for creating iStar extensions?
4. Is it necessary/useful the proposal of a process for supporting iStar extensions? Why?
5. What is your opinion about PRISE?
6. What are the strongest points of PRISE?
7. What are the weaknesses of PRISE?
8. What do you think about the guidelines, catalogue and the verification of completeness, consistency and conflicts proposed in PRISE?
9. What is the effort/difficulty level to understand PRISE? Which are the most difficult and the easiest parts of the PRISE?
10. Is PRISE suitable/important to support the proposal of iStar extensions by the iStar community? Why? To whom?
11 Would PRISE (or part of it) be useful to create your next extensions? How? What about other researchers?
12. How do you think the process would be used to create extensions? As proposed or with changes/part of it?
13. Do you have any changes/improvements/comments related to prise?
14. Is there something about PRISE that we did not mention in the interview and you would like to talk about?

Table 12
Questions of the quantitative study (survey).

1. There is a lack of process to guide the creation of new iStar extensions.
2. The proposal of a process for supporting iStar extensions is necessary.
3. It is relevant following the guidelines of the iStar community.
4. It is important to try to reuse the constructs of existing iStar extensions.
5. The catalogue of iStar extensions can help to identify the constructs to be reused.
6. It is important to verify the completeness of the iStar extension proposed[2].
7. Which value of the scale below represents the level of contribution that PRISE can have to avoid the occurrence of incompleteness in iStar extensions.
8. It is important to verify the consistency between the meaning of constructs, abstract and concrete syntaxes
9. Which value of the scale below represents the level of contribution that PRISE can have to avoid the occurrence of inconsistencies in iStar extensions.
10. It is important to verify the occurrence of conflicts in the iStar extension.11. Which value represents the level of contribution of the PRISE to avoid the occurrence of conflicts in iStar extensions.
11. Which value represents the level of contribution of the PRISE to avoid the occurrence of conflicts in iStar extensions.
12. What is the difficulty level to understand PRISE?
13. The use of PRISE is feasible for creating iStar extensions.
14. How suitable is PRISE to be used in the definition of the future iStar extensions?
15. PRISE is important to create future iStar extensions.
16. The iStar community can benefit from the usage of PRISE in the definition of their next extensions.
17. PRISE can be useful to create my next extensions.
18. How will you use PRISE to propose your next extensions?
19. To whom can PRISE be useful?
20. Would you recommend PRISE to other researchers?
21. What are the strongest points of the PRISE?
22. What are the weaknesses of the PRISE?
23. Do you have any changes/improvements/comments related to prise?
The survey is composed of the following kind of questions:
Likert Scale (Strongly agree, agree, do not know, disagree, strongly disagree) → 1, 2, 3, 4, 5, 6, 8, 10, 13, 15, 16, 17
Measure the contribution (0 to 5) → 7, 9, 11
Likert Scale (Very suitable, suitable, do not know, unsuitable, very unsuitable) 14
Yes/no/maybe question → 20
Open questions → 21, 22, 23
Specific options → 12, 18, 19
Question 12's options: Very easy, easy, medium, hard, very hard
Question 18's options: I will use PRISE as it was proposed, I will use a great part of the tasks and artefacts of PRISE to create mine instance of this process, I will use a few tasks and artefacts of PRISE to create mine instance of this process, I will not use anything of PRISE, Other
Question 19's options: To extenders without experience, To expert extenders, Both, Nobody, Other

annotations. All interviews were recorded (with permission of each participant).

[Data Analysis and Synthesis] Qualitative data from the interviews were then analysed using procedures from the Grounded Theory methodology defined by Strauss and Corbin in Strauss and Corbin (2007). Grounded Theory aims at building a new local theory from collected data rather than from predefined concepts. The statements presented in Strauss and Corbin (2007) were used to categorise and synthesise data, to build an evidence-based theory. We also followed the guidelines presented by Stol et al. (2016). The audios of the interviews were transcribed, and the MAX QDA 12 (a tool used to perform qualitative analysis) was used to support the analysis.

We labelled portions of text using text codes (initially opened coding, then closed coding). Coding consists of giving a label to important portions of the interview transcriptions. The opened coding is used at the beginning of analysis to identify relevant portions of the interview transcriptions and create the codes. The closed coding is used to identify relevant portions of the interview transcriptions based on codes identified in opened coding. We started with opened coding, where several codes were identified. We then performed a closed coding step where interview transcripts were re-evaluated to try to identify codes not found in the first analysis.

Initially, we did an intra-participant analysis (i.e., analysis in the transcription of each participant) to create labelled portions of text using codes and an inter-participant analysis (i.e. analysis between the transcriptions of the participants) to relate these codes giving rise to the categories that were named following a constant comparison method (Strauss and Corbin, 2007). According to Merriam (2009), data are grouped in a similar dimension. Thus, we created a summary of the results of this study with the main findings identified by codes.

We took notes during this analysis (memoing) and contacted part of the three interviewed (two participants) one more time to clarify some issues that emerged during the analysis of their interviews.

Regarding the survey, we represented the results of the survey's questions by graphics which present the number of the responses for each possible response.

In general, the results of both studies point to the agreement of the participants about the usefulness of PRISE and its acceptance as a way to support the proposal of their next iStar extensions.

8.1.1. Results and discussion of the interviews with experts

This section reports the results and discussions of the qualitative study, being organised to present the understanding of the participants about PRISE. In this section, we presented parts of the transcriptions of the interviewees in italic and between double quotation marks.

Lack of studies and need to conduct iStar extensions systematically

Initially, we addressed the interviewee's understanding of what is lacking for researchers to create extensions more systematically. This is covered in the interview script by question 1. The three participants of this study were unanimous when affirmed that, excluding PRISE, they do not know any process proposed to support iStar extensions. P1 stated that the iStar community did not really propose anything (process method or other work) to systematise how to extend iStar yet.

These results are coherent with results of a previous SLR on iStar extensions (Gonçalves et al., 2017a). The result of a research question presented in Gonçalves et al. (2017a) did not identify any process or guidelines to guide the iStar extensions proposals.

For this reason, two participants refer to the need of some guidance. P1 and P2 mentioned that it is useful to propose guidance have some guidance when proposing iStar extensions, but it should not necessarily be a process. Another kind of approach can also contribute, such as a method or guidelines, but a process is a good way. The participant P3 was more emphatic about the definition of a process: "*I think it would be great such a process defined and so can be more systematic and help to pay attention to all the different aspects that would not be necessarily have considered without it*". [P3]. We observed during the interviews that P1 and P2 presented a point of view more generic than P3, but they did not oppose the solution to be a process (in fact they commented that a process is a good way).

We agree with P1 and P2 that there are a great number of ways to propose guidance for the creation of iStar extensions. Processes are important because they give consistency and structure to a set of activities, gather existing experiences making it reproducible (Pfleeger, 2003). Thus, during the development of this research, we decided to propose a process.

Usefulness and Suitability of PRISE

When asked about their opinion about PRISE, all of them considered it to be useful. P1 cites that the tasks are suitable and contribute to the extenders not to forget any step of the proposal such as defining the metamodel or checking inconsistencies. P1, P2 and P3 mentioned that the process is too detailed. P1 also commented about this aspect: "I think this is not really surprising, but I can understand that it is easy to forget to do certain things without the process.".

The participants agree that PRISE is suitable to support the proposal of iStar extensions by the iStar community (question 10). In this sense, P1 said "I think so. Nothing in it that really extraordinary but it may be the ordering of some of the processes. They don't… otherwise, there's nothing that's not sensible. I mean you could always add more things, but it seems consistent with related work and yeah, it seems like it would be helpful.". P2 also complemented that there are no reasons to say it is unsuitable. According to P2, it can definitely improve the way that extensions are proposed helping to avoid common mistakes.

Regarding to whom it is suitable, they told it is suitable to both experienced and novices, but mainly for novices. We highlighted part of the response of the P1 where it is justified this point of view: "Particularly new researchers. I mean the people who usually do this kind of work tend to be students. This is a PhD or master's common type of work. However, it can also be useful for more senior researchers.".

We agree with them that PRISE is mainly useful to novices since they have no previous experience about proposing iStar extensions. This answer can be related to the transcription of P1 when he/she mentioned that PRISE is not extraordinary. Maybe for him/her (a researcher with a great experience) part of the tasks of the PRISE were obvious. However, newcomers need to be reminded of all steps.

Highlights about the contribution

Participants P1 and P3 cited some parts of the PRISE as necessary for this, such as the guidelines and to verify problems of incompleteness, inconsistencies and conflicts. On the other hand, P2 mentioned in a general way that it is necessary the proposal of a process to support the proposal of iStar extensions.

All participants mentioned to do a literature review and that defines the concrete syntax of the extension. P1 cited that he/she generally performs the literature review in a systematic way. P1 and P2 said they have been changing the way how they propose an extension. P1 said that sometimes the researchers propose the abstract syntax and sometimes do not propose it.

These comments highlight the lack of reproducibility and verifiability in the proposal of existing iStar extensions by experts. It is an evidence of the relevance of PRISE to the creation of iStar extensions by experienced extenders.

Strongest points and weaknesses

We enquired directly about the strongest points of the PRISE (question 6). P1 cited some highlights of the PRISE such as the checklist of verification of incompleteness, inconsistencies and conflicts and the reuse of the existing iStar extensions by the catalogue. P2 mentioned that principles and fundamentals behind it are very solid. As presented in Section 4 (Methodology) our process is based on a set of the empirical studies that joined the observation about what is done in previous works, the opinion of the experts and the literature background. P2 also commented that PRISE can help to avoid errors that are common to happen such as incompleteness, inconsistencies and conflicts. P3 said that the description is very clear and detailed and there is no such process defined. Consequently, it comes to fill an existing gap.

The size (number of tasks) of PRISE was mentioned as a weakness by all participants. P3 commented that although the size is considerable, he would not remove any task of the original version of the PRISE. P1 and P2 cited that the repository is a very nice thing, but the chances of it being kept up-to-date are not that great. We agree with them, but the tasks of 1.10 (*Identify if there are extensions related to the current proposal*) and 2.1 (*Search and select constructs to be reused*) mention that the catalogue can helps but it is necessary to do a search considering the last date the catalogue was updated. Therefore, we informed a complementary step to the catalogue during these tasks, i.e., the extender can do a search in the databases of the main venues of publications and try to identify new extensions between the last update of the catalogue and the current date. We also will perform periodic mapping studies to up-to-date the catalogue.

Participants P1 and P2 mentioned that consulting experts in iStar extensions could halt the process if the contacted expert does not respond. We clarified it in the process description by mentioning that the extender should contact more than one expert and if they delay responding to identify others and contact them. There is a list of 30 iStar extensions experts.

The participants consider PRISE easy to understand and that the number of steps may imply it is difficult to follow. Particularly, P3 said that "*someone who has not gone through it may be intimidated right apparent complexity*" and P1 commented that "*PRISE is not difficult, it's very detailed which is again not necessarily a bad thing. So, it takes some time*.". Thus, this response complements the weakness presented in the previous paragraph when the participants told that the size is a weakness of the PRISE. This explanation implies that this weakness is related to time. P2 corroborate with this point of view: "*you have deadlines, then the problem may be a time problem*.".

Also, we asked about the most difficult and the easiest parts of PRISE. Only P1 commented that some of the terminologies are a bit difficult when there needs to be a glossary here a set of definitions. We think this is an interesting suggestion and we created a glossary and made it available here: www.cin.ufpe.br/~ler/prise/glossary.

Adoption of the PRISE

We received a different kind of responses regarding if PRISE is useful to create their next extensions. P1 commented that is not focused on proposing new iStar extensions at the moment, P2 said that will definitely consider using PRISE in the next extensions and P3 will try to use it.

Regarding the usage by the other researchers, they think there are no reasons for not using it. P2 highlighted three topics regarding this: (i) "*it's a bit too ambitious for the community to say we can establish it as a partner now, but I keep the line of that you should continuously collect evidence about the suitability, generality, availability*"., (ii) "*No matter what, some people still use other methods even when improving mental work. For example, after refining iStar 2.0, many people still do not use it*". and (iii) "*Recreate an existing extension and correct the mistakes can help to convince the experts to use*.". We highlighted these parts because we considered a realistic point of view.

An interesting discussion was made around the question 12 (*How do you think the process would be used to create extensions? As proposed or with changes/part of it?*). All participants commented as something natural the users of the PRISE not consider some tasks. P3 explained that would expect that when people start using any process when in humans has done following a process, they might adapt it to some degree. And P2 complemented that the experts certainly would adjust it when to use. P2 and P3 also commented that we should maintain all elements (tasks and artefact) of the PRISE and let people customise it when needed.

P1 gave a very interesting explanation about the usage in the industry context as follows:

> "*I'm doing this with industry I would probably just do whatever I need to do in the short term to make it work but then I would go back afterwards and try and make sure that we actually covered the related work and that we... you know... make sure to define things systematically in the concrete syntax and abstract syntax. So, I can imagine if I was to actually use this in practice and I would probably apply it afterwards*". [P1].

Improvements

Regarding the question 13 (*Do you have any changes/improvements/comments related to PRISE?*), the participant P2 suggested to maintain the tools developed in more robust repositories such as GitHub (github.com/) or others similar because sometimes the researcher make it available in a link in places that are not accessible permanently as the GitHub. We introduced this suggestion in the description of the task 3.5.4. *Make the tool available in a link*.

Finally, they did not comment on the last question (*14. Is there something about PRISE that we did not mention in the interview and you would like to talk about?*).

8.1.2. Results and discussion of the survey with experts

In this section, we describe the results of the survey. We could not compare the data of our results and data of similar studies for other modelling languages despite searching and not finding similar studies.

PRISE received a good evaluation of the experts that participated in the survey. This reiterates the findings of the qualitative study. Table 13 shows the results for the questions 1, 2, 3, 4, 5, 6, 8, 10, 13, 15, 16 and 17. Median, mode and number of responses received for each option (Strongly Agree, Agree, Do not Know, Disagree and Strongly Disagree) were presented for each question.

We can identify that responses Strongly Agree and Agree are the most frequent of the responses, as consequence medians and modes are Strongly Agree or Agree. These kinds of responses received in average 16.91 responses for each question, it represents 84.58% of the responses. The number of these responses varies between 14 and 20 responses. We highlight the responses of the questions 3, 4, 5, 6, 8, 10, 13, 15 and 16 did not receive any disagree or strongly disagree response. We received two responses disagree in Question 1 (*There is a lack of process to guide the creation of new iStar extensions*), one response in Question 2 (*The proposal of a process for supporting iStar extensions is necessary*) and two in Question 17 (*PRISE can be useful to create my next extensions*).

Question 1 (*There is a lack of process to guide the creation of new iStar extensions*) has a related field to inform the existence of the process and the participant mentioned that there is not a process, but the books and papers of Domain-Specific Modelling languages can help to propose iStar extensions. Therefore, this participant did not point an existing process. Furthermore, another participant described that each author makes his/her extensions in an ad hoc way and follow some steps. However, we cannot consider it a process.

Questions 7, 9 and 11 are about measuring the importance level to avoid inconsistencies, incompleteness and conflicts. We used a scale from 0 to 5 in these questions. In Question 7 (*Which value of the scale below represents the level of contribution that PRISE can have to avoid the occurrence of incompleteness in iStar extensions*), we received only responses between 3 and 5 with median 4. In Question 8 (*Which value of the scale below represents the level of contribution that PRISE can have to avoid the occurrence of inconsistencies in iStar extensions*), we received one response 0, one response 2 and the other 18 responses between 3 and 5 with median 4. Finally, in Question 11 (*Which value represents the level of contribution of the PRISE to avoid the occurrence of conflicts in iStar extensions*), we received one response 2 and the other 18 responses between 3 and 5 with median 4. The results of these three questions are shown in Fig. 17.

In Question 12 (*What is the difficulty level to understand PRISE?*), most of the participants (90%) considered the difficulty level to understand PRISE medium or easy, only one response pointed it is hard and none very hard.

In Question 14 (*How suitable is PRISE to be used in the definition of the future iStar extensions?*), most of the participants (70%) considered PRISE Very Suitable or Suitable. Any response pointed it is Unsuitable or Very Unsuitable.

In Question 18 (*How will you use PRISE to propose your next extensions?*), one (1) participant indicated that will use PRISE as it was proposed and nine (9) participants pointed that will use a great part of the tasks and artefacts to create their next extensions. Four (4) participants indicated that will use few tasks and artefacts of PRISE, one (1) participant pointed that will not use anything of the PRISE and two (2) gave other responses. The participant that choose will not use anything of the PRISE

Table 13
Results of the validation survey — questions 1–8, 10, 13 and 15–17.

Questions	Median	Mode	Strongly Agree	Agree	Do not Know	Disagree	Strongly Disagree
1. There is a lack of process to guide the creation of new iStar extensions.	Agree	Strongly Agree/Agree	7	7	4	2	0
2. The proposal of a process for supporting iStar extensions is necessary.	Strongly Agree	Strongly Agree/Agree	8	8	3	1	0
3. It is relevant following the guidelines of the iStar community.	Strongly Agree	Strongly Agree	9	7	4	0	0
4. It is important to try to reuse the constructs of existing iStar extensions.	Strongly Agree	Strongly Agree	12	7	1	0	0
5. The catalogue of iStar extensions can help to identify the constructs to be reused.	Strongly Agree	Strongly Agree	14	5	1	0	0
6. It is important to verify the completeness of the iStar extension proposed.	Strongly Agree	Strongly Agree	11	7	2	0	0
8. It is important to verify the consistency between the meaning of constructs, abstract and concrete syntaxes	Strongly Agree	Strongly Agree	12	7	1	0	0
10. It is important to verify the occurrence of conflicts in the iStar extension.	Strongly Agree	Strongly Agree	11	9	0	0	0
13. The use of PRISE is feasible for creating iStar extensions.	Agree	Agree	3	12	5	0	0
15. PRISE is important to create future iStar extensions.	Agree	Agree	5	10	5	0	0
16. The iStar community can benefit from the usage of PRISE in the definition of their next extensions.	Agree	Agree	8	10	2	0	0
17. PRISE can be useful to create my next extensions.	Agree	Agree	6	8	4	2	0

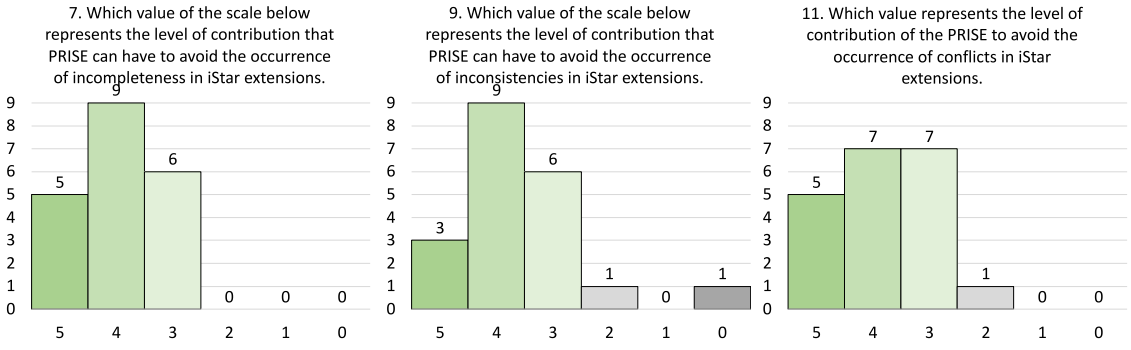


Fig. 17. Results of the validation survey — Questions 7, 9 and 11.

commented that will not propose new iStar extensions and selected this answer for this reason. This response is similar to the participant P2 of the interviews (Section 8.1.1).

Fig. 18 shows the results of Questions 12, 14 and 18.

In Question 19 (*To whom can PRISE be useful?*), two (2) participants pointed that PRISE is useful for experts, one (1) participant pointed it is useful to extenders without experience and eleven (11) pointed it is useful for both (experts and without experience).

We asked the participants if they would recommend PRISE to other researchers. 70% of the participants answered *yes* and 30% answered *maybe*. We had not any *No* response.

The results of Questions 19 and 20 are shown in Fig. 19.

Questions 21, 22 and 23 were open. We synthesised the responses of the Question 21 (*What are the strongest points of PRISE?*) in the following terms used by the participants: standardisation/systematic/well structured; pedagogical/good description of the activities; based on facts/expert feedback; Guidelines; Checklist; Catalogue; Concrete examples; Avoid unnecessary extensions; Maintain the list of extensions; Create extensions more

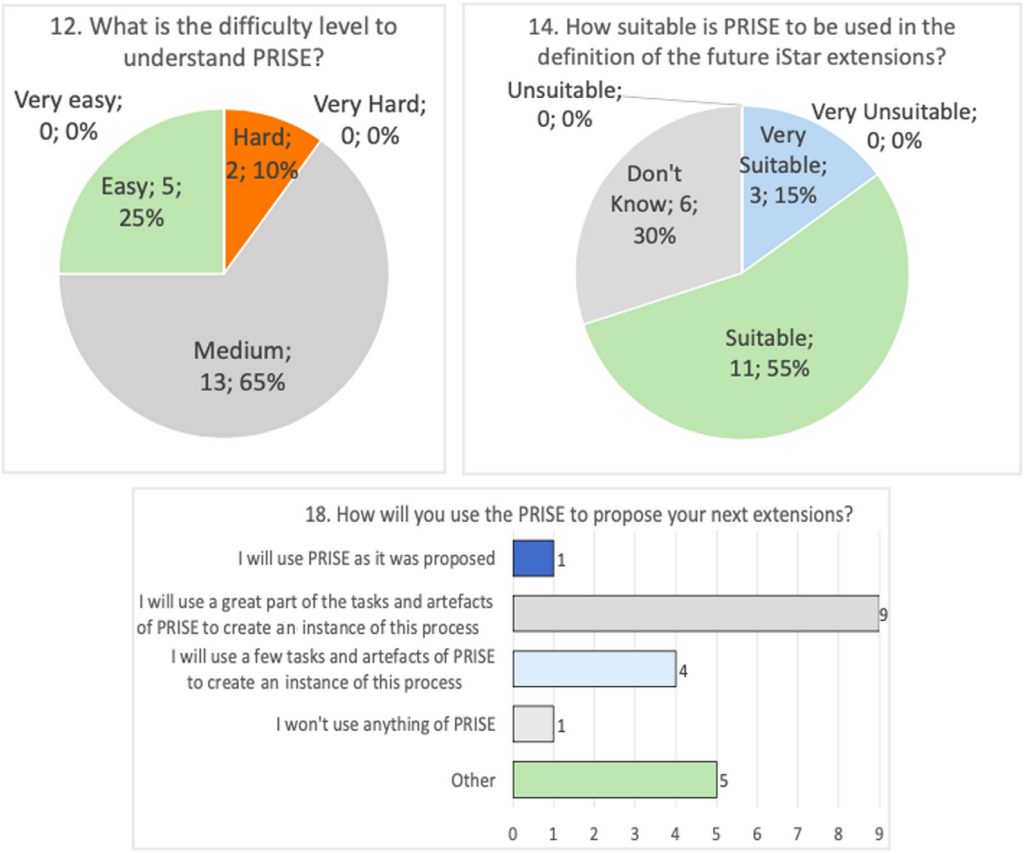


Fig. 18. Results of the validation survey — Questions 12, 14 and 18.

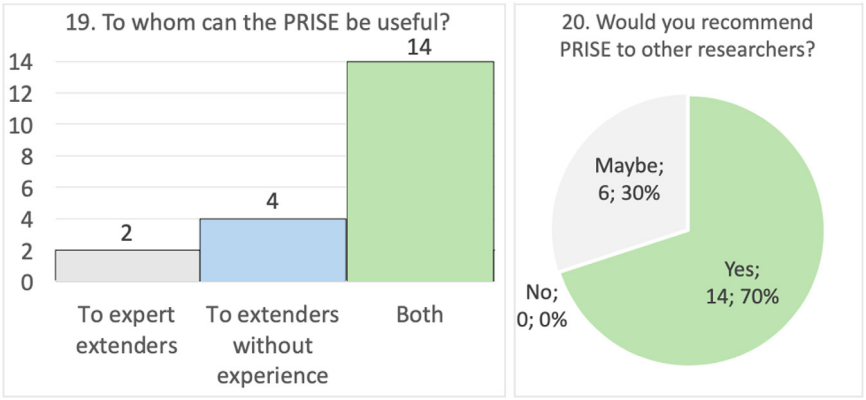


Fig. 19. Results of the validation survey — Questions 19 and 20.

correctly. The Participant 3 pointed out that "*The strength of the Prise is in detailing the steps and careful with the checks that must be performed, to maintain the integrity of the original language*.".

There was a consensus in the responses of all participants pointing the high level of detail/size of the process as its weakness (Question 22 — What are the weaknesses of PRISE?). The Participant 6 did an interesting comment about the process: "*The weakness of PRISE, on the other hand, lies in its own strength. That is, the more details it proposes, but the more charged it gets*.".

The main suggestions of changes/improvements in the PRISE (Question 22 — Do you have any changes/improvements/comments related to PRISE?) was to create a tool to support/manage the usage of the process and the illustration of the instantiation of the process in a soft version. We addressed these comments in this paper.

8.1.3. Final considerations

In summary, the participants of the qualitative study pointed to the lack of existing process, method, guidelines or another way to conduct the proposal of iStar extensions systematically. They recognised that a process could be useful to guide the creation of new extensions. These results are coherent with the results of the questions 1 (There is a lack of process to guide the creation of new iStar extensions) and 2 (The proposal of a process for supporting iStar extensions is necessary) of the survey presented in Section 8.1.2, which received about 75% of acceptance. Research in computer science is motivated to propose solutions to real problems. Both results confirm the need to propose a process to support iStar extensions by the iStar community and points out that a process is considered a way to reach it.

They mentioned specific strong points as guidelines of the iStar community, reuse the constructs of existing iStar extensions, the catalogue of iStar extensions and verification of problems. They also stated that the process is very clear and detailed. Questions 3–10 of the survey of Section 8.1.2 analyses the opinion of participants about the specific points of the PRISE and they agreed with the importance of them. PRISE is based on previous studies about existing extensions; thus, its proposal was made to deal with the needed points identified previously. These results confirmed one more time the real need of them.

The main weakness pointed in both studies is the size (number of tasks) of PRISE. The participants mentioned that due to this level of details, the PRISE could require more time to create a new extension than an extension created without the process. We were concerned about it because this characteristic can have impact on the adoption of the PRISE by the experienced extenders. On the other hand, a less detailed proposal could make difficult the adoption by extenders novices. Thus, we maintained the PRISE in a detailed way, but also presented how to customise in order to simplify to make it more appealing for experienced extenders.

The results point to the intention of its use by the participants as well as the recommend of its adoption by other researchers. It was not considered hard to use, but it was considered too detailed. It was mentioned that PRISE could of value mainly for the creation of new iStar extensions by novices, but the experienced researchers could be benefited as well. The results of questions 13–16 of the survey reiterate these findings. This opinion is one more evidence about the acceptance of the PRISE by the iStar community. Two new extensions have been proposed with PRISE (Gonçalves et al., 2019b; Ribeiro et al., 2019), and we believe that when PRISE be published, this number will increase.

8.1.4. Threats to validity

According to Kitchenham and Pfleeger (2002), there are four aspects that we need to consider: Criterion Validity, Construct Validity, Face Validity and Content Validity. We presented a description of them according to Kitchenham and Pfleeger (2002) in the next paragraphs.

The Criterion Validity is a measure of how well an instrument compares with another predecessor instrument. Criterion validity compares a new instrument against one that is considered a "gold standard". For instance, a proposed new Capability Maturity Model questionnaire might be compared with the existing one to test its concurrent criterion validity. We can use correlations to indicate the extent to which the two questionnaires agreed. Predictive criterion validity assesses the ability of a survey to predict future phenomena. For example, we can survey project managers about on-going projects and their characteristics in order to predict how much effort they will require. We can correlate the predictions with the outcome to assess predictive criterion validity.

Construct Validity is the observation of how an instrument "behaves" when in use. It can be convergent or divergent. Convergent construct validity assesses the extent to which different data collection approaches produce similar results. Divergent construct validity assesses the extent to which results do not correlate with similar but distinct concepts.

Face validity is a cursory review of items by untrained judges. It hardly counts as a measure of validity at all because it is so subjective and ill-defined. This is a way to quickly identify the accuracy of the instrument to the research. In other words, a test can be said to have face validity if it "looks like" it is going to measure what it is supposed to measure. For instance, if a test is prepared to measure whether students can perform multiplication, and the people to whom it is shown all agree that it looks like a good test of multiplication ability, this demonstrates face validity of the test.

Finally, Content Validity is an assessment of how appropriate the instrument looks to reviewers with knowledge of the subject matter. It typically involves a systematic review of the survey's contents to ensure that it includes everything it should and nothing that it should not. If we are developing a new instrument in a topic area that has not previously been researched, it is the only valid form of preliminary validation available.

In this section, we presented the threats to validity of both studies with the experts.

Threats to validity of the qualitative study with experts

Criterion validity — We did not find a previous qualitative study for this purpose so that we could compare it with ours. It can be considered a threat to Criterion Validity. In the paper (Burnay et al., 2014), which is similar to ours since it used mixed methods, the authors analysed documents from previous projects. They used semi-structured interviews in the qualitative stage and the survey with a Likert-type scale of five levels in the quantitative part.

The documental analysis of previous iStar extensions was presented in the paper of SLR (Gonçalves et al., 2017a). The qualitative study of the paper (Burnay et al., 2014) was made to confirm the findings of the documental analysis and discovered aspects not discovered in the documental analysis. Our script interview, therefore, has a broader scope once we have more general questions to confirm and does not bias the responses of the participants.

In paper (Gonçalves et al., 2017b), the authors used mixed methods to analyse the opinion of the researchers who proposed the existing iStar extensions about how they have done their extensions and what can be done to improve the next iStar extensions. The results of this paper based the proposal of the PRISE, but its goal is different from this paper because here we performed a qualitative to validate a process, not to analyse what to do.

Construct validity — The remote location of the participants and the usage of skype may have led to some misunderstandings that might have been relevant. We believe, however, that these effects are at least partially mitigated by the fact that each interview was conducted by one author of this paper while a second author took notes. We also recorded the audio of interviews for a detailed transcription and analysis later. The misunderstandings and inconsistencies identified during the analysis of the qualitative study were discussed. When necessary, further explanations were sought from the participants to mitigate this threat to validity.

We recorded the audio during the interviews to make feasible their transcription and the analysis, so we asked for permission at the beginning of the interview. This information could inhibit the responses of the participants. We mitigated this threat informing the participants that the audio files and transcriptions would be maintained private and they would be anonymous. We also presented a confidentiality and privacy term.

A great part of the questions (questions 4–14) of the qualitative study asked the participants' opinion about PRISE, once this is the goal of this study. For example, we asked about weaknesses, the effort/difficulty level. Thus, we were apprehensive that participants omitted their opinion on these questions, and we had no relevant evidence for them. However, this did not happen once we had enough material in the transcriptions.

Face validity — We tested the script interview with one computer science professor of the Universidade Federal do Ceará. He/she did not propose any iStar extension. We received several comments about the script interview; then, we corrected the script interview accordingly.

We can consider this previous evaluation as a limitation because one unique pilot participant performed this step. However, we mitigated this threat by asking him/her to evaluate again after the corrections of their comments.

Again, the participant of this test did not propose any iStar extension; however, they knew part of the iStar extensions and had already used some of him/her, thus mitigating this threat.

Content validity — We performed a pilot involving one expert in iStar extensions to validate the script interview in the qualitative study. We also can consider this previous evaluation a limitation because of the small number of participants. We mitigated this threat to Content Validity by the validation of the script interview by the fourth and the fifth authors of this paper.

In this survey, we did not have a large number of participants. This threat can be mitigated by inviting additional researchers who will propose the next iStar extensions to answer this survey.

Threats to validity of the quantitative study with experts

We now comment on the threats to the validity of the survey.

Criterion validity — We did not find a previous survey for this purpose so that we could compare it with ours. It can be considered a threat to Criterion Validity.

The survey of the paper (Gonçalves et al., 2017b) aims to analyse the opinion of the researchers who proposed the existing iStar extensions about the main findings of a qualitative study. The results of this paper based the proposal of PRISE, but its goal is different from our because here we performed a qualitative study to validate a process, not to analyse statements of a qualitative study.

Construct validity — We created the survey with different kind of questions: Likert Scale questions (1, 2, 3, 4, 5, 6, 8, 10, 13, 14, 15, 16, 17), Measure the contribution questions (7, 9, 11), Yes/no/maybe question (20), Open questions (21, 22, 23) and Specific options (12, 18, 19). So, it could confuse the participants while they were answering. We mitigated this threat to construct validity presenting an explanation about the kinds of questions at the beginning of the survey to clarify the structure of the survey to the participants. We did not receive any comment about it during the pilot or the execution.

Face validity — The Face Validity of the survey is like the Face Validity of the interviews. We tested the survey with a computer science professor of the Universidade Federal do Ceará. He/she did not propose any iStar extension. However, we received several comments about the survey; then, we corrected it accordingly.

We can consider this previous evaluation a limitation because of the small number of participants (1 participant). We mitigated this threat, however, by asking them to evaluate again after the corrections of their comments.

Despite the re-evaluation, the participants of this test did not propose any iStar extension, although they knew many iStar extensions and had already used some of them, thus mitigating this threat.

Content validity — We performed the pilot involving one researcher. It was done to test the understanding of participant about the survey. We analysed the feedback sent by the participant of the pilot and applied the suggested improvements in the survey.

During the application of the survey, we received some comments from two participants. We tried to mitigate these threats to Content Validity with the participation of two experts in iStar extensions in the development of this research, which validated the survey before the submission to the participants.

8.2. Evaluation by the novice

We performed a case study to analyse the usage of PRISE to propose a new iStar extension to model safety requirements of critical systems. It is an important way to evaluate PRISE by the point of view of a researcher that used this process to create a new iStar extension.

The extenders followed all steps of the original version of the PRISE. Thus, the need to propose the extension started when the extenders felt that it was necessary to explicitly model safety requirements with iStar. They needed the language to be more expressive to represent the specific concepts and decided to create a new extension. The extenders selected five concepts of this application area and added them as new constructs to iStar to model strategies to mitigate hazards in these systems.

[Objective] The purpose of this evaluation is to identify the point of view of a novice extender about PRISE. It will contribute to the evaluation of a reference process to guide other iStar extensions.

[Study Design] We performed an interview to analyse the opinion of a novice in iStar extensions about PRISE. This study allowed us to capture valuable evidence about the opinion of a novice about the usage of PRISE. By doing so, we carried out a triangulation of the findings with the supervisor.

[Participant] The extension was performed by a master student in computer science from Brazil under the supervision of an expert in iStar extensions which proposed four extensions and have more than ten years of experience. The master student has no previous experience with iStar extensions, extensions of other modelling languages or with the domain concepts. She had basic knowledge about the usage of iStar, BPMN and UML. We interviewed the master student and triangulated the findings with her supervisor.

[Collection Preparation] Clarification and consent terms were sent to participants before each interview session. Semi-structured interviews were conducted using an interview script with open questions. The complete version of the script of the interviews is available in Table 14.

The script interview was validated by the second and the third authors of this paper. We tested the script interview with a computer science professor from Universidade Federal do Ceará. This professor did not propose any iStar extension but has experience in requirements engineering and empirical research. The pilot data were not used during the analysis.

[Data Collection] Interview was conducted in Portuguese via Skype in November 2018. After the interview, we contacted the participant twice to clarify some issues that emerged during the analysis. We presented the results of the analysis to the supervisor to receive his feedback about the findings.

[Data Analysis and Synthesis] The audio of the interview was transcribed, and the MAX QDA 12 (a tool used to perform qualitative analysis) was used to support the analysis.

Initially, we did an intra-participant analysis (i.e. analysis in the transcription of the participant) to create labelled portions of text using codes. Finally, we created a summary of the results of this study with the main findings.

8.2.1. Results and discussion of the interviews with the novice

This section reports the results and discussions of the interview with the extender of the new iStar extension, to present the point of view about the usage of the PRISE. In this section, we presented parts of the transcriptions of the interview in italic and between double quotation marks.

Initially, we asked why PRISE was used to support their new iStar extension proposal. The master student answered that the supervisor is an expert in iStar extensions who knows about

Table 14
Script interview of the qualitative study with the novice.

1. Why did you use PRISE?
2. How difficult is it to understand/learn/follow PRISE?
3. How was your experience of using PRISE? Did you use the process as it was proposed or did you make any changes?
4. What do you think about the guidelines, catalogue and checklist for completeness verification and consistency proposed in PRISE?
5. Imagine that PRISE did not exist or you did not use it to create this current extension and answer the following questions: What would be different about creating the extension? What is the impact of using/not using PRISE in the quality of extensions, completeness, consistency, conflicts and reuse of existing extensions?
6. If you have to propose another extension in the future, would you use/not use PRISE? Why?
7. If you knew someone interested in proposing a new iStar extension, what would you tell him/her about PRISE?
8. Do you have any changes/corrections/changes related to PRISE?
9. Is there anything else you would like to comment on your experience using PRISE?

PRISE and that there is a stable version of the process. Thus, the supervisor suggested the usage of the PRISE as a way to create the extension systematically and reach high quality. The extender studied the process and concluded that it could help the creation of the new iStar extension.

Following, we asked her about the difficulty level of understanding and following PRISE.

> "*It was not difficult to understand it, I accessed PRISE modelling available at the website and read the description of the tasks, gateways and artefacts. The diagram is well presented and well described. The video of the presentation of the process also contributed to the understanding. The artefacts guided the execution*". [Novice extender].

The mentioned video is available at https://youtu.be/_LF4u-MOsD0 (It is the same video cited at Section 8.1).

The experience about the usage of iStar was described as follows: The process was used as proposed, without changes. The master student had no previous experience with creation of extensions or knowledge about the creation of modelling languages. Therefore, she did not know the steps involved in the creation of an extension. Thus, PRISE was cited as fundamental to support and arrange the step-by-step of the work to be done and the artefacts to be generated as a result. The PRISE tool (Gonçalves et al., 2019d) was not available to be used when the extender started the extension. Consequently, the extender filled the results the artefacts gradually on her computer. The generated artefacts help to clarify the final result and to understand how the extension representation was reached. Specific parts of the PRISE were mentioned like the creation of the metamodel, that was something not known by her. The participation of an expert in iStar extension was relevant to avoid the creation of an iStar extension with unnecessary constructs. An expert in safety was contacted six times to mitigate issues about the concepts to be introduced, and there was no need for consult other experts on iStar extensions once there was an expert involved in the extension proposal. Modelling a system of the domain during the execution of the first sub-process helped to better identify the constructs involved and the relationship between them and see the need to create the extension (without the process it would not). The catalogue of iStar extensions was used to better know about the existing iStar extensions and to search constructs to be reused. However, there was no constructor in iStar extensions to represent the concepts of safety targeted by the new iStar extension.

Consequently, the supervisor suggested a light-weight representation using the textual stereotypes and specific colours to represent the domain concepts. The checklist of the verification of completeness, consistency and conflicts was used, but no problem was identified. Thus, it was mentioned as an evidence that the process is effective to avoid these problems. The BPMN modelling with well-described tasks, gateways and artefacts were highlighted as other strong points of the PRISE.

We asked directly her opinion about the guidelines. She commented that considers "*the guidelines something important to consolidate the knowledge of the specialists and to facilitate the transfer of this knowledge to the beginners. It was impossible to have access to the information provided by the guidelines without them.*".

The master student tells us that would do a lower quality extension if she had not used PRISE because she would not do a great part of the tasks. According to her: "*I would concentrate on representing the concepts graphically and show the example of a system modelled with the extension.*". We can highlight that a complementary phrase cited by her: "*The big number of tasks is not a problem, once they make possible reach an extension more complete and with high quality*.".

This phrase can be related to the comment of the experts about the number of the tasks (see Section 8.1), it is an evidence that the detailed specification of the PRISE can help the inexperienced extenders to create well-done extensions. She also mentions that now she is able to give real importance to activities that she did not know at the beginning of the extension proposal (such as the creation of a metamodel or to do conservative extensions).

She also mentioned if PRISE did not exist she would depend on the availability of her supervisor to point the steps to be followed and spend more time studying how to do an extension. Therefore, the process also contributed to her autonomy concerning the supervisor. Complementary, the extender commented that, without PRISE, she would have to search constructs in the databases such as ACM, Springer, Scopus and IEE Xplore. This task would take more time and a greater chance of being unsuccessful than the usage of the catalogue.

Regarding the usage of PRISE to propose her following iStar extensions, she affirmed that certainly would use it because there was a considerable gain in organising the work to be done while using the process to create this current extension. She complemented saying that could use the expertise gained during the usage of the PRISE to create the current extension. Thus, the activities and artefacts of the PRISE would be more familiar with the creation of a second extension.

Regarding what would say to a researcher interested in an opinion about the process, she commented that recommend strongly the usage.

> "*PRISE will point you in the right direction, be careful to document everything. At the moment of defining the metamodel, you have to be with the understanding of the concepts to be introduced. Make interactions to include the concepts incrementally. Do not worry initially with the graphic representations of the constructs. The modelling tool is very important to easier the usage of the extension and makes the validation viable.*". [Novice extender]

Finally, we asked her about corrections or improvements to PRISE. The extender had some difficulties to represent the extension in the iStar metamodel and the description of this task

was cited as a weakness of the PRISE. Therefore, we improved the description of this task highlighting the guidelines of experts in iStar extensions related to them, made some recommendations about the iStar metamodel and pointed some references about metamodeling and extension. The participant considered this new information enough to clarify the execution of this task.

8.2.2. Threats to validity

We followed the same aspects presented in Section 8.1.4 according to Kitchenham and Pfleeger (2002): Criterion Validity, Construct Validity, Face Validity and Content Validity. Additionally, we analysed threats to the conclusion validity.

Criterion validity. We did not find a previous qualitative study for this purpose so that we could compare it with ours. It can be considered a threat to Criterion Validity. We believe that the study has similarities with the study presented in Section 8.1 Thus, the instruments of these works have similarities and obtained satisfactory results.

Construct validity — We recorded the audio during the interviews to make feasible their transcription and the analysis, so we asked for permission at the beginning of the interview. This information could inhibit the responses of the participants. We mitigated this threat informing the participants that the audio files and transcriptions would be maintained private and they would be anonymous. We also presented a confidentiality and privacy term.

We translated the analysis of the interviews from Portuguese to English. We can be chosen inaccurate terms accidentally during the translation and introduced a kind of bias or in the analysis. We mitigated this threat asking the novice extender and her supervisor about the text with the final results of this study.

Time and effort to create this new extension were not analysed. Thus, it is not possible to compare the creation of an iStar extension with PRISE and the creation without PRISE. We interviewed only one participant because we could not identify other novice extenders that used PRISE. These two threats are considered as future work.

Face validity — We tested the script interview with one a computer science professor of the Universidade Federal do Ceará. He/she did not propose any iStar extension. We received several comments about the script interview; then, we corrected the script interview accordingly.

We can consider this previous evaluation as a limitation because one unique pilot participant performed this step. However, we mitigated this threat by asking him/her to evaluate again after the corrections of their comments.

Content validity — PRISE was not published yet, as a consequence, few researchers know about it. We did not identify other novices using PRISE in this moment. Consequently, we were not able to test it with another novice extender. We believe that in the future, after the publication of a paper which presents PRISE, more researchers will use it and we can replicate this analysis.

9. Conclusion and future work

In this paper, we presented the main results of a process to support iStar extensions, its validation by recreating a set of existing iStar extensions and evaluation by a mixed methods approach.

The process PRISE is based on a set of previous studies to understand how the iStar extensions are performed and what can be done to improve them. It is also based on references of the literature which establishes what is required to create or extend a modelling language and about the quality of the graphical representations of the modelling languages. We presented the modelling of the process in BPMN with the explanation of each task and artefact. This process joins the point of view of the iStar extenders and makes it possible to create iStar extensions in a systematic way, avoiding the mistakes, such as absence of nodes and links of default iStar syntax, that have been happening. We consider the strongest points of PRISE a set of tasks to analyse if the extension proposal is really necessary, a set of tasks to support the creation of a complete extension, the reuse of existing iStar extensions by a catalogue of iStar extensions, a checklist to verify the completeness, consistency and conflicts, the usage of an empirical way to evaluate of the proposed extensions and the consult of the experts during the development of the development. We modelled the process in a very detailed way so that can be used by extenders without experience or can be adapted by experienced ones. We showed how to customise the process to the specific needs of an extender.

We also fixed five existing extensions with problems and showed one of them in this paper to illustrate the usage of the process and confirm its validity.

Finally, we presented the results of mixed methods based on interview and survey which aimed to analyse the point of view of the experts about PRISE process. The participants of the interviews pointed out to the lack of a process and needed to propose something capable of systematising the proposal of the next iStar extensions. They indicated some strong points of PRISE such as the reuse of iStar extensions and the verifications to avoid common mistakes. They also showed some weaknesses such as the catalogue up-to-date. In general, PRISE was well evaluated by the participants. The survey also gives a positive evaluation of the PRISE process although we have received some neutral and negative answers, which is expected. This step was important to better characterise PRISE.

We believe, therefore, that this process could be applied to other modelling languages in the Requirements Engineering area, such as KAOS and NFR. However, it is necessary to perform some adjustment to specificities of each other modelling language such as a specific catalogue for each one, the verification of their specific nodes and links and the list of the experts on their extensions. Furthermore, further studies are necessary to confirm or deny this claim.

We searched for similar papers for other languages and did not find works which investigate extensions for them. As future work, therefore, we intend to analyse our results in iStar with other modelling languages, such as UML, KAOS, NFR. In doing so, we will be able to make a more concrete comparison with the evolution of the extensions in each of them.

We think that a great challenge now is the adoption of PRISE by the community. So, we need to apply it to new extensions, to show evidence that it has been used bringing value to potential extenders. We are currently working on the creation of a new iStar extension. It is being proposed to model rational agents to fill the gap of our approach that just model the architecture of these kinds of software (Gonçalves et al., 2015), generate code automatically (Lopes et al., 2012) and make possible testing (Silveira et al., 2014). This new extension will be illustrated by a real case of a Multiagent system to distance education. Preliminary results of our new extenion to model multiagent systems with rational agents are available in Gonçalves et al. (2019b) Also, researchers from another research group are also using PRISE to create an iStar extension to deception modelling in the context of the security domain.

We did not have a large number of participants in the survey presented in Section 8.1 and in the study presente. Thus, we could not then make statistical inferences or to reveal a true pattern in the data. We intend to invite additional researchers who will propose the next iStar extensions to answer this survey. In the same way, we interviewed only one participant in the study

presented in Section 8.2 because we could not identify other novice extenders that used PRISE. We believe that in the future (about three years) it will be possible to replicate this study with more participants.

Time and effort to use PRISE to create extensions were not analysed in the context of this paper. These analyses will be addressed by an experiment to be performed in the near future. Similarly, it is necessary to evaluate the impact of the usage of a customised version of PRISE to create iStar extensions.

We are also doing an SLR to identify the iStar extensions proposed in the last two years and so update the catalogue of iStar extensions.

CRediT authorship contribution statement

Enyo Gonçalves: Conceptualization, Methodology, Software, Validation, Investigation, Writing - original draft. **João Araujo:** Conceptualization, Supervision, Investigation, Writing - review & editing. **Jaelson Castro:** Conceptualization, Supervision, Writing - review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

The authors thank all participants of this study. We also thank CNPQ/Brazil (Conselho Nacional de Desenvolvimento Científico e Tecnológico), CAPES (Coordenação de Aperfeiçoamento de Pessoal de Nível Superior), Brazil, FACEPE (Fundação de Amparo a Ciência e Tecnologia do Estado de Pernambuco), Brazil and NOVA LINCS UID/CEC/04516/2019 for the financial support to the execution of this work. We also thanks to Prof. Daniel Amyot and Prof. Eric Yu for the contributions to this work.

References

Alencar, F., 1999. Mapeando a modelagem organizacional em Especificações precisas. In: Centro de Informática (Ph.D. thesis). Universidade Federal de Pernambuco.

Ali, R., Dalpiaz, F., Giorgini, P., 2008. Location-based software modelling and analysis: Tropos-based approach. In: International Conference on Conceptual Modelling.

Atkinson, C., Kühne, T., 2003. The essence of multilevel metamodeling, in: C. Atkinson et T. Kühne, model-driven development: A metamodeling foundation. IEEE Softw. 20 (5), 36–41.

Barišić, A., Amaral, V., Goulão, M., Barroca, B., 2011. How to reach a usable DSL? Moving toward a Systematic Evaluation. In: Electronic Communications of the EASST (MPM).

Bastos, L., 2005. Systematic integration between requirements and architecture. In: Centro de Informatica (Ph.D. thesis). Universidade Federal de Pernambuco.

Boehm, B.W., 2007. Software risk management: Principles and practices. In: Software Engineering: Barry W. Boehm's Lifetime Contributions to Software Development, Management, and Research, Vol. 8. p. 387, no.1.

Brambilla, M., Cabot, J., Wimmer, M., 2012. Model-Driven Software Engineering in Practice. In: Morgan & Claypool Publishers series Synthesis Lectures on Software Engineering.

Bresciani, P., Perini, A., Giorgini, P., Giunchiglia, E., Mylopoulos, J., 2004. Tropos: an agent-oriented software development methodology. Auton. Agents Multi Agent Syst. 8 (3), 203–236.

Burnay, C., Jureta, I., Faulkner, S., 2014. An exploratory study of topic importance in requirements elicitation interviews. In: 26th International Conference on Advanced Information Systems Engineering. In: lecture notes in computer science, vol. 8484, Springer, Berlin, pp. 180–195.

Caire, P., Genon, N., Heymans, P., Moody, D., 2013. Visual notation design 2.0: Towards user comprehensible requirements engineering notations. In: 21st IEEE International Requirements Engineering Conference (RE).

Cares, C., Franch, X., 2011. A metamodelling approach for i* model translations. In: Advanced Information Systems Engineering. Springer Berlin Heidelberg.

Chung, V., 2006. Considering role-based conflicts of interest in analyzing and designing e-Health systems with goal-oriented methodologies. In: International Conference on Privacy, Security and Trust: Bridge the Gap Between PST Technologies and Business Services.

Creswell, J., 2014. A Concise Introduction to Mixed Methods Research. Sage Publications, Thousand Oaks.

Dalpiaz, F., Franch, X., Horkoff, J., 2016. IStar 2.0 language guide. arXiv:1605.07767, Available in http://arxiv.org/pdf/1605.07767v1.pdf.

Dardenne, A., Van Lamsweerde, A., Fickas, S., 1993. Goal-directed requirements acquisition. Sci. Comput. Program. 20, 3–50.

Dubois, E., Mayer, N., Rifaut, A., 2011. Improving risk-based security analysis with i*. In: Social Modelling for Requirements Engineering Book.

Elahi, G., Yu, E., Zannone, N., 2010. A vulnerability-centric requirements engineering framework: analyzing security attacks, countermeasures, and requirements based on vulnerabilities. Requir. Eng. J..

Franch, X., Mate, A., Trujillo, J.C., Cares, C., 2011. On the joint use of i* with other modelling frameworks: A vision paper. In: IEEE International Requirements Engineering Conference.

Gans, G., Lakemeyer, G., Jarke, M., Vits, T., 2006. SNet: a modelling and simulation environment for agent networks based on i* and congolog. In: 14th International Conference on Advanced Information Systems Engineering. Springer, Berlin, pp. 328–343.

Garlan, D., Monroe, R., Wile, D., 1997. Acme: an architecture description interchange language. In: Conference of the Centre for Advanced Studies on Collaborative Research. pp. 1–15.

Ghanavati, S., Amyot, D., Rifaut, A., 2014. Legal goal-oriented requirement language (Legal GRL) for modelling regulations. In: 6th International Workshop on Modelling in Software Engineering.

Giorgini, P., Rizzi, S., Garzetti, M., 2005. Goal-oriented requirement analysis for data warehouse design. In: 8th ACM International Workshop on Data Warehousing and OLAP. pp. 47–56.

Gonçalves, E., Almendra, C., Araújo, J., Castro, J., Goulão, M., 2019a. Addressing symbol redundancy representations in iStar extensions. In: 12nd International IStar Workshop in 38 Th International Conference on Conceptual Modelling.

Gonçalves, E., Almendra, C., Goulão, M., Araújo, J., Castro, J., 2019c. Using empirical studies to mitigate symbol overload in iStar extensions. Softw. Syst. Model..

Gonçalves, E., Araújo, J., Castro, J., 2018a. Towards extension mechanisms in iStar 2.0. In: 11th International I* Workshop Co-Located with the 30th International Conference on Advanced Information Systems Engineering.

Gonçalves, E., Araujo, J., Castro, J., 2019b. IStar4RationalAgents: Modeling requirements of multi-agent systems with rational agents. In: 38th International Conference on Conceptual Modeling.

Gonçalves, E., Araujo, J., Castro, J., 2020. A process to support the creation of istar extensions. In: 35th ACM/SIGAPP Symposium on Applied Computing. Brno, Czech Republic, pp. 1363–1370.

Gonçalves, E., Castro, J., Araújo, J., Heineck, T., 2017a. A systematic literature review of iStar extensions. J. Syst. Softw..

Gonçalves, E., Cortés, M.I., Campos, G., Lopes, Y., Freire, E., Silva, V.T., Oliveira, K., Oliveira, M.A., 2015. MAS-ML 2.0: Supporting the modelling of multi-agent systems with different agent architectures. J. Syst. Softw. 77–109.

Gonçalves, E., Heineck, T., Araújo, J., Castro, J., 2018b. CATIE: A Catalogue of istar extensions. In: 21st Workshop on Requirements Engineering.

Gonçalves, E., Monteiro, I., de Oliveira, M.A., Castro, J., Araújo, J., 2017b. Understanding what is important in istar extensions proposals: The viewpoint of researchers. Requir. Eng. J..

Gonçalves, E., Oliveira, L., Heineck, T., Araujo, J., Castro, J., 2019d. PRISE Tool: A tool to support the proposal of iStar extensions using the PRISE process. In: 22nd Workshop on Requirements Engineering.

Guzman, A., Martinez, A., Agudelo, F., Estrada, H., Perez, J., Ortiz, J., 2016. A methodology for modeling ambient intelligence applications using i* framework. In: International IStar Workshop in IEEE International Requirements Engineering Conference. pp. 61–66.

Henderson-Sellers, B., Eriksson, O., Gonzalez-Perez, C., Ågerfalk, et P.J., Eriksson, O., Henderson-Sellers, B., Ågerfalk, et P.J., 2013. Ontological and linguistic metamodelling revisited: A language use approach. Inf. Softw. Technol. 55 (12), 2099–2124.

Horkoff, J., Aydemir, F., Cardoso, E., Li, T., Maté, A., Paja, E., Salnitri, M., Piras, L., Mylopoulos, J., Giorgini, P., 2019. Goal-oriented requirements engineering: an extended systematic mapping study. Requir. Eng. V24, 133–160.

Horkoff, J., Elahi, G., Abdulhadi, S., Yu, E., 2008. Reflective analysis of the syntax and semantics of the i* framework. In: Advances in Conceptual Modelling-Challenges and Opportunities. Springer Berlin Heidelberg, pp. 249–260.

Islam, S., Mouratidis, H., Kalloniatis, C., Hudic, A., Zechner, L., 2012. Model based process to support security and privacy requirements engineering. Int. J. Secure Softw. Eng. 3 (3), 1–22.

ISO/IEC, ISO/IEC 24744, 2007. Software Engineering - Metamodel for Development Methodologies. ISO, Geneva.
Juristo, N., Moreno, A., 2001. Basics of Software Engineering Experimentation. Kluwer Academic Press, Springer, Boston.
Kitchenham, B., Pfleeger, S., 2002. Principles of survey research. Softw. Eng. Notes 26 (6), 16–27.
Kotonya, G., Sommerville, I., 1998. Requirements Engineering: Processes and Techniques. John Wiley & Sons, ISBN: 0-471-97208-8.
Lapouchnian, A., Yu, Y., Liaskos, S., Mylopoulos, J., 2006. Requirements-driven design of autonomic application software. In: Proceedings of the conference of the Center for Advanced Studies on Collaborative Research.
Liaskos, S., Mylopoulos, J., 2010. On Temporally Annotating Goal Models. International i* workshop.
Lopes, Y., Gonçalves, E., Cortés, M.I., Freire, E., 2012. A MDA approach using MAS-ML 2.0 and JAMDER. In: 13th International Workshop on Agent-Oriented Software Engineering in 11th International Conference on Autonomous Agents and Multiagent Systems, 2012, Valencia. Proceedings of 13th International Workshop on Agent-Oriented Software Engineering.
Lòpez, L., Aydemir, F.B., Dalpiaz, F., Horkoff, J., 2016. An Empirical Evaluation Roadmap for iStar 2.0. In: Proceedings of the Ninth International i* Workshop, pp. 55–60.
Louaqad, W., El Mohajir, M., 2014. A Holonic extension of i* framework. In: Third IEEE International Colloquium in Information Science and Technology (CIST).
Lucena, M., 2010. STREAM: a systematic process to derive architectural models from requirements models. In: Centro de Informática (Ph.D. thesis). Universidade Federal de Pernambuco.
Martins, L., Faria, H., Vecchete, L., Cunha, T., Oliveira, T., Casarini, D., Colucci, J., 2015. Development of a low-cost insulin infusion pump: Lessons learned from an industry case. In: Proceedings of the 2015 IEEE 28th International Symposium on Computer-Based Medical Systems.
Mellado, D., Mouratidis, H., Fernandez-Medina, E., 2014. Secure Tropos framework for software product lines requirements engineering. Comput. Stand. Interfaces.
Merriam, S., 2009. Qualitative Research: A Guide to Design and Implementation. Jossey-Bass, San Francisco.
Moody, D., 2009. The physics of notations: Towards a scientific basis for constructing visual notations in software engineering. IEEE Trans. Softw. Eng. 35, 5.
Morales, J.M., Navarro, E., Sánchez, P., Alonso, D., 2015. TRiStar: An i* extension for Teleo-Reactive systems requirements specifications. In: ACM Symposium on Applied Computing, Intelligent Robotics and Multi-Agent Systems Track.
Morandini, M., Penserini, L., Perini, A., Marchetto, A., 2015. Engineering requirements for adaptive systems. Requir. Eng. J..
Mouratidis, H., Giorgini, P., 2007. Secure tropos: A security-oriented extension of the tropos methodology. Int. J. Softw. Eng. Knowl. Eng..
Mouratidis, H., Islam, S., Kalloniatis, C., Gritzalis, S., 2013. A framework to support selection of cloud providers based on security and privacy requirements. J. Syst. Softw. 86 (9), 2276–2293.
Murukannaiah, P., Singh, M., 2014. Xipho: Extending tropos to engineer context-aware personal agents. In: International Conference on Autonomous Agents and Multi-Agent Systems.
Mylopoulos, J., Chung, L., Yu, E., 1999. From object-oriented to goal-oriented requirements analysis. In: Communications ACM. ACM.
Paige, R.F., Kolovos, D.S., Polack, et F.A.C., 2014. A tutorial on metamodelling for grammar researchers. Sci. Comput. Program. 96 (Part 4), 396–416.
Pfleeger, S., 2003. Software Engineering: Theory and Practice, second ed. Pearson.
Ribeiro, S.M., Castro, J., Vilela, J., Pimentel, J., 2019. IStar4Safety: Uma Extensão de iStar para Modelagem de Requisitos de Segurança em Sistemas Críticos. In: 22nd Workshop on Requirements Engineering.
Russell, S., Norvig, P., 2003. Artificial Intelligence: A Modern Approach. Prentice Hall.
Santander, V., 2002. Integrando modelagem organizacional com modelagem funcional. In: Centro de Informatica (Ph.D. thesis). Universidade Federal de Pernambuco.
Siena, A., Maiden, N., Lockerbie, J., Karlsen, K., Perini, A., Susi, A., 2008. Exploring the effectiveness of normative i* modelling: results from a case study on food chain traceability. In: 20th International Conference on Advanced Information Systems Engineering. In: Lecture Notes on Computer Science, vol. 5074, Springer, pp. 182–196.
Silveira, F.R.V., Campos, G.A.L., Cortés, M.I., 2014. A problem-solving agent to teste rational agents: A Case study with reative agents. In: 16th International Conference on Enterprise Information Systems (ICEIS), 2014, Lisboa. 16th International Conference on Enterprise Information Systems (ICEIS).
Stol, K., Ralph, P., Fitzgerald, B., 2016. Grounded theory in software engineering research: a critical review and guidelines. In: 38th International Conference on Software Engineering.
Strauss, A., Corbin, J., 2007. Basics of qualitative research: 2nd edition. In: Techniques and Procedures for Developing Grounded Theory, third ed. Sage Publications, Inc.
Wohlin, C., Runeson, P., Host, M., Ohlsson, M., Regnell, B., Wesslén, A., 2012. Experimentation in Software Engineering. Kluwer Academic Publishers, Springer, Boston, MA.
Yu, E., 1995. Modelling Strategic Relationships for Process Reengineering (PhD. Computer Science). University of Toronto, Toronto.
Yu, E., 1997. Towards modelling and reasoning support for early phase requirements engineering. In: Proceedings of the 3rd IEEE International Conference on Requirements Engineering.
Zave, P., 1997. Classification of research efforts in requirements engineering. ACM Comput. Surv. 29 (4), 315–321.

Enyo Gonçalves received his Ph.D. in computer science at Universidade Federal de Pernambuco, Brazil, in 2019. He completed his Master in Computer Science at the Universidade Estadual do Ceará, Brazil, in 2009. He received a B.S. degree in Computer Science from Universidade Estadual do Vale do Acaraú, Brazil, in 2007. Mr. Gonçalves is currently a Professor at Universidade Federal do Ceará, Brazil. Enyo's research is focused primarily on the software engineering methods and their application to a wide variety of problems in Model-Based Engineering (MBE) applied to Requirements Engineering (RE) and Multi-Agent Systems (MAS). His current research projects include analysis of extensions in modelling languages.

João Araújo is a professor at the Department of Informatics at the Universidade Nova de Lisboa, Portugal and a full member of the Portuguese research center NOVA LINCS. He holds an MSc from Universidade Federal de Pernambuco and a PhD from Lancaster University, UK, both in the area of Software Engineering. His principal research interests are Requirements Engineering (RE), Advanced Modularity, and Model-Driven Engineering (MDE), where, he has published several papers on these topics in journals, international conferences, and workshops. Within these subjects he has also been involved in several projects, such as AMPLE (funded by the European Union), Aspects for Space Domain (funded by ESA), etc. He has served in the organisation of several conferences such as RE, MoDELS, ICSE, ECOOP, AOSD. He was the co-general chair of the IEEE Requirements Engineering conference in Lisbon, 2017. He was awarded with the most influential paper of the AOSD 2013 and best paper award of Caise 2014, among others. He has launched the series of workshops on model-driven RE (MoDRE) that has been held in RE conference.

Jaelson Castro is a Professor at Universidade Federal de Pernambuco, Brazil, where he leads the Requirements Engineering Laboratory. He holds a PhD in Computing from Imperial College, London, UK. His research interests include requirements engineering, model-driven safety critical system development and robotics. He serves on the editorial board of the Requirements Engineering Journal and the Journal of Software Engineering Research and Development and acted as Editor-in-Chief of the Journal of the Brazilian Computer Society — JBCS.