On the Use of Requirement Patterns to Analyse Request for Proposal Documents

Dolors Costal[0000-0002-7340-0414], Xavier Franch[0000-0001-9733-8830], Lidia López[0000-0002-6901-9223], Cristina Palomares[0000-0003-4722-5584], Carme Quer[0000-0002-9000-6371]

Universitat Politècnica Catalunya (UPC-BarcelonaTech), Barcelona, Spain
{dolors|franch|llopez|cpalomares|cquer}@essi.upc.edu

Abstract. Requirements reuse is still today a difficult goal to achieve. One particular context in which requirements reuse may give more benefits than costs is that of call for tenders projects, due to the similarity of the requirements documents (which take the form of requests for proposal documents, RfPs) from one project to another. In this paper, we present an approach aimed at making systematic the assessment of RfPs that technology providers need to conduct in order to decide whether they present a bid or not in a call for tenders project. The approach extends a metamodel we already defined for the former PABRE method, which has a similar goal but from the perspective of the organization that issues the call for tenders. The method is illustrated with an exploratory case study in the field of the railway systems domain.

Keywords: Requirements Reuse, Requirement Patterns, Call for Tenders, Request for Proposals, Bidding Process.

1 Introduction

Reuse is a cornerstone activity in all facets of engineering, and Requirements Engineering (RE) is not an exception. There are several recent works reporting on approaches to requirements reuse (see [1] for a literature review) and in particular, evidence exists that in industry, reuse practices are not yet commonplace [2].

One of the contexts where requirements reuse may pay off occurs with call for tenders processes articulated around the needs for some technological solution exposed in Request for Proposals (RfPs). Multiple RfPs in a same domain (e.g., railway domain) or for the same type of systems (e.g., business applications) may be similar to each other, which opens the way to requirements reuse. In previous work, we have explored one particular approach to requirements reuse, namely the use of requirement patterns, in order to help customers to efficiently produce new RfPs [3][4]. Therefore, the focus of this previous work has been on the customer side. In this paper, we want to explore if the adoption of a pattern-based approach to requirements reuse can also be of help to technology providers when processing the RfP.

2 Background

We consider background on requirement patterns and on RfPs. The seminal book by Withall [5] proposed a first exhaustive catalogue of patterns and, since then, other approaches have been proposed [6][7]. Requirement pattern approaches differ in several respects, see Barros-Justo et al.'s [8]. The most obvious one is the language used to express the requirements, being natural language and use cases the two most popular cases. Other factors that need to be considered are the granularity of the object under reuse, the intended impact in the RE process and the scope.

Regarding RFPs, there are approaches focused on customers and others on providers. Lauesen has approached the customer perspective of call for tenders' processes in several works (e.g., [9]). He has provided some guidelines for the customer, which he reports are not always applied. Our PABRE method [3] was designed for supporting the customer in the preparation of the RfPs. It is based on the creation of pattern catalogues [10][11] with a well-defined metamodel [12]. The requirements that are part of a RfP are created by instantiating the patterns. On the providers side, Paech et al. [13] report the challenge to deal with large RfPs in a tight period with little or no communication with the customer. They propose a risk-based approach in which different types of risks are sought and identified in new RfPs. The resulting analysis is the input to the RfP evaluation and then it is used to decide about the actions to be taken in the bidding process. From a similar perspective, Breiner et al. [14] propose a 4-phase process to deal with RfPs in IT providers to be tailored in every individual bidding. Both providers approaches are similar in their methodological stand, but they lack of a central repository of knowledge and have little tool support.

In this paper we address the provider perspective to call for tenders processes by applying a pattern-based approach to the assessment of RfPs. We use assets built in the PABRE method and evolve them to include the new relevant information.

3 Research Goal and Research Questions

Our goal is to *evaluate* the benefits of *a pattern-based approach* on *assessing RfPs* from the point of view of *technology providers* in the context of *multiple call for tenders processes in the same domain*. This goal is decomposed into two research questions:

RQ1. What type of information needs to be added to requirement patterns to help technology providers in their assessment of RfPs?

RQ2. Does the use of requirement patterns bring benefits to technology providers when organizing their bidding processes?

4 Patterns in the Railway Domain

As case study for the research questions, we used six RfPs from the railway domain that the Viennese Siemens Mobility department made available for this research in the

context of the OpenReq EU project [15]. The RfPs are composed of 17,556 candidate requirements, classified depending on whether they were considered as a real requirement (DEF) or as merely informative (Prose). For each candidate requirement the document also includes the domain or department of Siemens that is the one that had to do the analysis of the compliance of the requirement. One example of requirement is "*On the body of the half barrier 3 light units are mounted. Lights on the half barrier must be visible at night from at least 20 m under normal visibility conditions*", while an example of Prose is "*The purpose of the new computerized interlocking system is described in the present Requirements*".

From the RfPs we constructed 25 patterns of 6 diverse categories. Table 1 summarizes the classification of these patterns and the number of requirements out of the RfPs from which each one has been obtained.

Table 1. Classification of the generated patterns

Type	Category	Pattern	#Reqs
Infrastructure Management	Facility Removal	Remove Facility	3
	Equipment Replacement	Replace Equipment	18
Supporting Systems	Video Surveillance System Installation	Require Video Surveillance System	2
		Establish Video Cameras Location	2
		Establish Video Cameras Mounting	1
		Establish Video Cameras Protection	1
		Establish Monitor Computers	1
		Establish Monitor Location	2
		Establish Monitor Screen	1
		Establish Monitor Display Options	1
		Establish Network Connection Features	2
		Establish Recording Functions	1
	Automatic Block Signaling System Installation	Install Automatic Block Signaling System	10
		Modify Automatic Block Signaling System	16
Non-technical	Training	Make Training Plan	2
		Supply Training Documents	2
		Supply Training Equipment	2
		Establish Trainees	2
		Establish Training Language	1
		Evaluate and Certify Trainees	1
	Warranty	Establish Warranty Period	1
		Provide Assistance	1
		Provide Monitoring Equipment	1
		Remove Defects During Warranty Period	5
		Replace Product During Warranty Period	1

One of the constructed patterns, *Remove Facility*, is depicted in Fig.1 to present requirement patterns elements. The *Remove Facility* pattern can only take one form (*Facility basic* pattern form). The pattern form has a core part (Fixed part) that expresses its basic linguistic template. It also contains three optional extensions to this core part (Extended parts) to describe the levels of the facility to remove, its size and its location, respectively. The bold tags enclosed among "%" are representing parameters that would correspond to specific values in the RfPs (for instance, the parameter %typeOfFacility% could be instantiated with the value "Watchman's Post or Family House"). The Domain and Compliance clauses are explained in the next section.

```
Remove Facility
Goal:           Remove a room or an existing building
Pattern form:   Facility basic
Fixed part:     In %railwayLocation% the contractor shall remove a
                   %typeOfFacility%.
Extended parts:
EP1:            Levels of the facility
   Template:    The facility is a %facilityLevels% facility.
   Compliance:  %facilityLevels% is under 9 levels
EP2:            Size of the facility
   Template:    The facility has %units% %unitMeasure%
EP3:            Facility location
   Template:    The facility is identified in the cadastral plot %ca-
                   dastralIdenfication% of the %districtName%
   Compliance:  - The facility is located in a place accessible by road
                - The facility is situated on a stable ground
Domain:         Installation_Local
```

Fig. 1. Remove Facility Requirement Pattern

5   RQ1: Pattern Attributes

As result of RQ1, we plan to extend the metamodel of the existing PABRE method [12] with new classes, associations and attributes required. Fig. 2 shows the result.

We include in the figure only the relevant excerpt of the PABRE metamodel (e.g., we hide information about classification schemas) over which we include the new elements. Original PABRE classes are filled in salmon color and they show the structure introduced in the previous section: a `Requirement Pattern` can take one or more `Pattern Forms`; each `Pattern Form` is characterized by a `Fixed Part` and one or more `Extended Parts`. In the metamodel, an abstract class `Pattern Part` is introduced for convenience. `Dependency` allows establishing dependencies between patterns. `Glossary Term` and `Relationship` between terms facilitate to deal with synonymy, ambiguity, etc. For the rest, we distinguish:

Information at the organization level (classes in white background). This information needs to be defined only once by the organization:

- Class `Domain`. The classification of patterns into domains allows selecting the department that will assess every requirement in the RfP. More precisely, every atomic component inside the structure of a pattern, i.e., a part, should be assigned to one domain. We allow this to be made at three different levels: individually at every part, at the level of a pattern form (meaning that all the parts of a pattern form belong to the same domain) and at the level of a pattern (meaning that all the forms of a pattern –and transitively all of its parts– belong to this domain). To model this comfortably, we introduce an abstract class `Level`. One or more `Departments` will participate in the assessment of all requirements of a given `Domain`. In Fig. 1, we show that the *Remove Facility* pattern has *Installation_Local* as domain.
- Class `Assessment Factor`. Companies will assess RfPs with respect to factors like cost, effort or risk. Its instances are linked to `Value` so that the values for every assessment factor can be explicitly defined. For instance, companies can define risk as `Assessment Factor`, with six possible values (Type1 to Type6) [13].
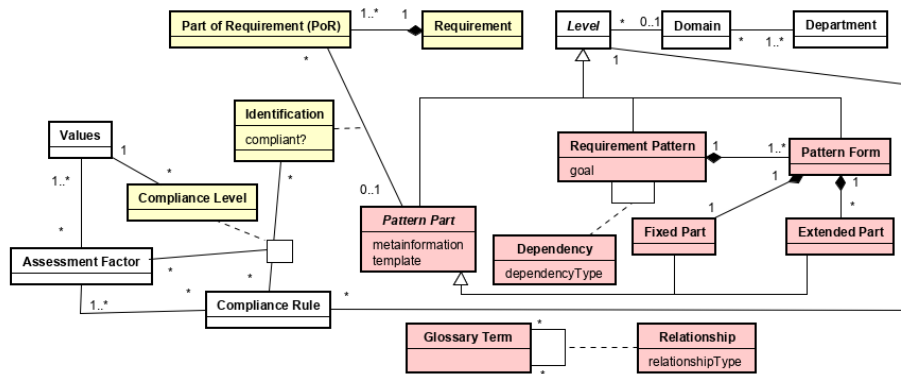
Fig. 2. Extending the PABRE metamodel with information fit for technology providers

- Class `Compliance Rule`. Every `Pattern Part` may have, directly or indirectly, either (through its pattern form or its pattern), one or more `Compliance Rules` that express a condition to be measured with respect to some `Assessment Factors`. The purpose of `Compliance Rules` is to give providers a checklist to decide if every (part of a) requirement appearing in the RfP can be eventually satisfied or not. The class `Level` is used with the same purpose as in `Domain`. In Fig. 1, we show three compliance rules attached to two different extended parts, expressing a restriction on the number of levels and two restrictions on the facility location.

Information at the project level (classes in yellow background). This information needs to be defined at every call for tenders project:

- The RfP includes `Requirements` that can be decomposed into atomic `Parts of Requirements (PoR)`. E.g., "*on cadastral plot 362, cadastral district Acme Acres*" is a `PoR` about removing a facility that provides its location.
- Association class `Identification`. These `PoR` are the ones matching `Pattern Parts` by giving values to the parameters (not shown in the figure): a `PoR` may match one `Pattern Part`, while one `Pattern Part` can eventually be matched to several `PoR`. This matching is kept by the `Identification` association class. For instance, the identification "*The facility is identified in the cadastral plot cadastral plot 362 of the Acme Acres district*" corresponds to the matching of the previous `PoR` example and the *Facility location* part of the *Remove Facility* pattern (see Fig. 1). If a `PoR` does not match any `Pattern Part`, this means that this `PoR` is not covered by the current catalogue (i.e., must be handled manually).
- Association class `Compliance Level`. For every `Compliance Rule` associated to a `Pattern Part` identified in a `PoR`, it is required to propose a `Compliance Level` in the form of a `Value` for each of its `Assessment Factors`. For the previous `Identification` example the `Compliance Rules` are: "*The facility is located in a place accessible by road*" and "*The facility is situated on a stable ground*". These rules can be used to propose the `Compliance Level` of the iden-

tification. As a result of this assessment, the `PoR` will be labelled as compliant, compliant under conditions or non-compliant. This assessment will be recorded in the `compliant?` attribute from the `Identification` association class.

6   RQ2: Preliminary Evaluation

We ran a questionnaire inside Siemens to have a preliminary evaluation of our work. We used the TAM evaluation questionnaire [16]. Specifically, we use a simplified version given that the technology is not fully available. We asked the participants to evaluate their vision on the adoption of the pattern-based approach using two simplified scenarios that focus on the patterns usage. The scenarios presented a RfP for which the respondent company wanted to present a bid and described the steps of the pattern usage proposal with the help of mock ups of an envisaged PABRE system.

Questions can be consulted in the online document that presents the questionnaire[1]. Table 2 presents the results of the evaluation The averages computed should be considered for informal reference purposes only, because the values are given in a Likert scale 1 (strongly agree) to 7 (strongly disagree), therefore in an ordinal scale, not with a ratio. However, they are still useful for intuitive explanations. All questions are positive (i.e., 1 means the most positive answer) except for question 2.2; therefore, in the averages, we have computed the inverse value of its responses (i.e., from N to 7-N+1).

The results show that 3 respondents (E2, E3 and E5) were receptive to the summative Question 6: "*Based on the previous scenarios, and assuming that the PABRE system were available, I would intend to use it*". Instead, E1 and E4 were reluctant. Respondents with positive attitude were cautious anyway, as clearly stated by E3: "*Although being open-minded, I am not sure whether this approach could work in practice*". We consider that this position is normal when it comes to considering an emergent, not yet available technology in a mature and complex process.

On the positive side, we can see that the system is perceived as easy-to-use (Question 2), even by E1 and E4. E2 likes the general layout of the solution, while E5 nuances that "*this will depend on the quality of the pattern identification*". The positive respondents are unanimously positive with respect to relevance (Question 3) and even E1 was neutral at this respect. E3 justifies his/her particularly positive rating "*because it can potentially increase productivity*". Anyhow, E2 doubts "*[...] that all relevant decisions can be expressed as patterns (due to complexity and efforts)*". This is a valid point that is in line with our general understanding that patterns cannot be realistically expected to embed all possible knowledge in a bidding process. Finally, results demonstrability is also well considered except for E3 (no rationale provided).

On the negative side, respondents were especially concerned with the expected output quality (Question 4). For instance, E2 expresses, "*I think it [the approach] needs human intelligence to solve the task. Wrong results can do harm!*". Concerning the two negative respondents, both of them remarked that the separation of require-

---

[1] https://www.upc.edu/gessi/PABRE/OPENREQ-PABRE-Questionnaire.pdf

ments and prose is considered "*a misleading approach. E.g., a header gives the para-graph the right frame [...]*" (E4). We consider this not a fundamental problem to a pattern-based approach but to the way in which we proposed our process in RQ3. We could then modify the output of the requirements triage in a way that the requirements list keeps the context of every requirement.

Also, E5 made the point that a particular RfP may not fit well with a pattern-based approach, e.g., "*productivity and effectiveness are expected to vary depending on the nature of the requirements document and the quality of the pattern recognition*". The importance of the nature of the requirements document has already been shown in RQ1, where some RfPs were more aligned to the identified patterns than others.

Table 2. Questions to Siemens practitioners to evaluate the pattern-based solution

Criteria		E1	E2	E3	E4	E5	Avg
1.Perceived usefulness		5,00	3,67	3,00	6,00	3,00	4,13
1.1	productivity	5	4	3	6	3	
1.2	effectiveness	5	4	3	6	3	
1.3	useful	5	3	3	6	3	
2. Perceived ease of use		2,33	1,33	6,00	3,33	3,33	3,27
2.1	understandable	3	2	7	2	2	
2.2	requires a lot of mental effort	6	7	2	2	4	
2.3	easy to use	2	1	5	2	4	
3. Relevance		4,00	2,50	3,00	6,00	2,00	3,50
3.1	pertinent	4	3	3	6	2	
3.2	relevant	4	2	3	6	2	
4. Output quality		3,00	6,00	7,00	6,00	3,50	6,38
4.1	high quality	3	6	7	6	3	
4.2	no problems with quality	3	6	7	6	4	
5. Result demonstrability		4,33	2,00	6,33	2,67	2,33	3,53
5.1	no difficulty to explain	6	2	6	2	2	
5.2	communicate consequences	3	2	6	2	2	
5.3	results apparent	4	2	7	4	3	
6. Behavioral Intention		6	3	2	6	3	4,00
6.1	intent to use	6	3	2	6	3	

7   Conclusions and Future Work

We have presented a pattern-based approach to support IT providers when assessing RfPs and deciding whether to bid for them or not. The main results are an extension of the PABRE metamodel with the information needed to give support to the provider during the bidding, and the results of a questionnaire to get early feedback from our proposal.

As threat to validity, we have evaluated our approach only in one case (Section 6). This case has several characteristics: the (type of) domain, the characteristics of the organization, the size and type of RfP documents and others. Generalizing our results beyond these contextual characteristics requires careful reflection.

Our future work focuses at the automation of the approach. Our intention is to make our approach particularly appealing and more prone to scale in contexts where a considerable number of bidding processes around large RfPs from the same domain take place. The main functionalities that the platform will support are: requirements

triage, to classify information from RFPs in order to distinguish the requirements from the document prose; patters identification, that will do the match among requirements and the specific patterns in a catalogue; and decision-making support, intended to help in the decision of compliance of requirements in a RfP. Some components are already available, as the web services to manage the patterns catalogue; others are being developed as the NLP components to pre-process RfPs.

References

1. Irshad, M., Petersen, K., Poulding, S.: A Systematic Literature Review of Software Requirements Reuse Approaches. *Information and Software Technology* 93, 2018.
2. Palomares, C., Quer, C., Franch, X.: Requirements Reuse and Requirement Patterns: A State of the Practice Survey. *Empirical Software Engineering* 22 (6), 2017.
3. Renault, S., Méndez-Bonilla, O., Franch, X., Quer, C.: PABRE: Pattern-based Requirements Elicitation. RCIS 2009.
4. Franch, X., Quer, C., Guerlain, C., Renault, S., Palomares, C.: Constructing and Using Software Requirement Patterns. In: *Managing Requirements Knowledge*, Springer, 2013.
5. Withall, S.: *Software Requirement Patterns*. Microsoft Press, 2007.
6. Pacheco, C. L., Garcia, I. A., Calvo-Manzano, J. A., & Arcilla, M.: A Proposed Model for Reuse of Software Requirements in Requirements Catalog. *Journal of Software: Evolution and Process* 27(1), 2015.
7. Barcelos, L.V., Penteaso, R.D.: Elaboration of Software Requirements Documents by means of Patterns Instantiation. *Journal of Software Engineering Research and Development* 5:3, 2017.
8. Barros-Justo, J.L, Benitti, B.V., Cravero-Leal, A.L.: Software Patterns and Requirements Engineering Activities in Real-World Settings: A Systematic Mapping Study. *Computer Standards & Interfaces* 58, 2018.
9. Lauesen, S.: COTS Tenders and Integration Requirements. *Requirements Engineering Journal* 11(2), 2006.
10. Palomares, C., Quer, C., Franch, X., Renault, S., Guerlain, C.: A Catalogue of Functional Software Requirement Patterns for Content Management Systems. SAC 2013.
11. Palomares, C., Quer, C., Franch, X., Guerlain, C., Renault, S.: A Catalogue of Non-Technical Requirement Patterns. RePa 2012.
12. Franch, X., Palomares, C., Quer, C., Renault, S., De Lazzer, F.: A Metamodel for Software Requirement Patterns. REFSQ 2010.
13. Paech, B., Heinrich, R., Zorn-Pauli, G., Jung, A., Tadjiky, S.: Answering a Request for Proposal – Challenges and Proposed Solutions. REFSQ 2012.
14. Breiner, K., Gillmann, M., Kalenborn, A., Müller, C.: Requirements Engineering in the Bidding Stage of Software Projects – A Research Preview. REFSQ 2015.
15. Felfernig, A., Stetinger, M., Falkner, A., Atas, M., Franch, X., Palomares, C.: OpenReq: Recommender Systems in Requirements Engineering. i-Know 2017.
16. Davis, F.F.: Perceived Usefulness, Perceived Ease of Use, and User Acceptance of Information Technology. *MIS Quarterly*, 13(3), 1989.