3-4-2015

Requirements Catalog for Business Process Modeling Recommender Systems

Michael Fellmann

Novica Zarvic

Dirk Metzger

Agnes Koschmider

Requirements Catalog for Business Process Modeling Recommender Systems

Michael Fellmann[1], Novica Zarvic[1], Dirk Metzger[1], and Agnes Koschmider[2]

[1] Osnabrück University, Information Management and Information Systems, Germany
{michael.fellmann, novica.zarvic, dirk.metzger}@uos.de
[2] Karlsruhe Institute of Technology (KIT), Germany
agnes.koschmider@kit.edu

Abstract. The manual construction of business process models is a time-consuming and error-prone task. To improve the quality of business process models, several modeling support techniques have been suggested spanning from strict auto-completion of a business process model with pre-defined model elements to suggesting closely matching recommendations. While recommendation systems are widely used and auto-completion functions are a standard feature of programming tools, such techniques have not been exploited for business process modeling although implementation strategies have already been suggested. Therefore, this paper collects requirements from different perspectives (literature and empirical studies) of how to effectively and efficiently assist process modelers in their modeling task. The condensation of requirements represents a comprehensive catalog, which constitutes a solid foundation to implement effective and efficient Process Modeling Recommender Systems (PMRSs). We expect that our contribution will fertilize the field of modeling support techniques to make them a common feature of BPM tools.

Keywords: Business Process Modeling, Recommender Systems, Requirements.

1 Motivation and Relevance

Business process modeling and reorganization are still among the top-ten of relevant topics of today's CIOs [1]. However, the construction of semi-formal process models is even today, after two decades of research on business process modeling, a highly manual task involving substantial human effort. Regarding the modeling activity, effort is required to create models conforming to specified rules regarding the naming of model elements and the abstraction level of model elements. Concerning the naming of model elements, terminological problems are amongst the main problems when using conceptual (process) models [2]. Moreover, effort and difficulty arises due to the complexity of today's business processes. It might not be easy to figure out where to start modeling a process and where to stop and on which abstraction level to model [3, 4] since guidance in modeling is largely missing in current tools. These barriers call for process modeling support features, which assist users during process model-

12th International Conference on Wirtschaftsinformatik,
March 4-6 2015, Osnabrück, Germany

Fellmann, M.; Zarvic, N.; Metzger, D.; Koschmider, A. (2015): Requirements Catalog for Business Process Modeling Recommender Systems, in: Thomas. O.; Teuteberg, F. (Hrsg.): Proceedings der 12. Internationalen Tagung Wirtschaftsinformatik (WI 2015), Osnabrück, S. 393-407

ing and make suggestions how to complete a currently being edited process model. Such assistance functions are common features in programming environments (in terms of auto-completing e.g., Java code) or e-commerce systems (e.g., amazon.com uses a recommender system to help users dealing with information overload). Although it has been demonstrated that assistance functions are beneficial in these domains [5] [6], assistance functions are not considered in commercial BPM tools. Recommender systems "generate meaningful recommendations to a collection of users" [7], therefore, it should be a priority to offer assistance functions in process modeling tools. Generally, the auto-completion of programming snippets seems to be easier than to auto-complete graphical modeling tasks where a variety of attributes (e.g., syntactic consistency, semantic validity and completeness, readability) influence the decision for an appropriate subsequent fragment.

This contribution presents a requirements catalog for Process Modeling Recommender Systems (PMRSs). This is not a trivial task and it should be noted that the elicitation and specification of requirements are considered to represent quite difficult processes in the area of requirements engineering [8]. In the research underlying the paper, a comprehensive list of requirements has been gathered from a literature analysis as well as from three different studies carried out within two years that also involved business users. Such a multi-perspective approach provides a solid foundation in order to build effective and efficient PMRSs aiming to facilitate the creation of high-quality process models with less time and effort. The merit of this contribution lies in providing a holistic view on requirements for PMRSs. This holistic view is not sought to build the basis for a particular implementation, but serves as a structured basis for future research works that may devise sophisticated solutions. In this way, we expect that our contribution will fertilize the discussion on assistance function around process modeling, which already has been identified as useful [9].

This contribution is structured as follows. At first, we describe methodological aspects (Section 2). We then elicit requirements by conducting a systematic literature analysis (Section 3) before we present additional requirements by practitioners that have been identified in the course of three studies performed (Section 4). The requirements are consolidated and synthesized into a structured catalog of requirements for PMRSs (Section 5) providing a backbone for possible future implementations. Finally, we summarize and discuss our work (Section 6).

2 Methodological Considerations

The goal of this paper is to provide a holistic view on requirements for PMRSs. For achieving this, relevant scientific works were inspected as well as the wishes and needs of practitioners were gathered in order to better understand the design space for PMRSs. Understanding both, the research side with the current literature as well as the practitioner side, is crucial for requirements researchers [10]. Firstly, we have consulted relevant literature by conducting a systematic literature review [11, 12]. This step assures to have considered relevant research output from the scientific community itself. Secondly, requirements of business users (i.e. practitioners and students that may become prospective practitioners) were gathered from different studies. This step assures that users who create business process models and are familiar with BPM tools were also involved in the elicitation of our collection of requirements. With regard to this we performed three studies, namely (i) a short online-survey about modeling support functionality, (ii) a case study, and (iii) a survey at a major fair that was based on a live-demonstration of a prototypical implementation. Summing up, the research process followed can be characterized to be *exploratory* in nature [13], where the results from literature as well as from users gradually consolidate the set of requirements, which are finally synthesized into a structured collection. Our research process is depicted in Fig. 1.

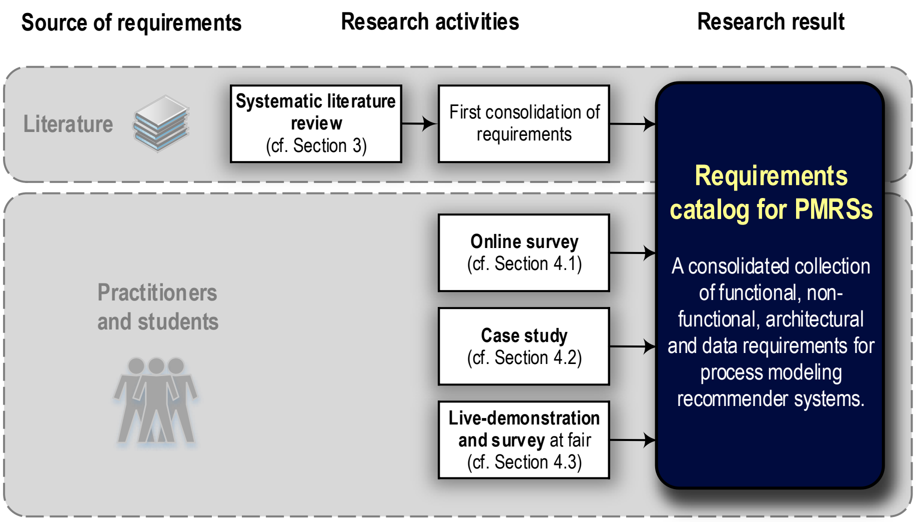

Fig. 1. Overview on research process taken

3 Requirement Elicitation from Literature

3.1 Overview of the Literature Analysis

The first step in collecting requirements was performed by reviewing the literature of related work. By doing so, we aimed – among others – to extract relevant capabilities that should be reflected by the PMRSs and which have already been deemed desirable and valuable in this context. In the beginning of our literature search, we abstained from using our already known literature. Instead, we have followed literature reviewing guidelines as proposed by WEBSTER and WATSON [11] and VOM BROCKE et al. [12]. The selected databases used for literature analysis (cf. Table 1) cover a broad field of scientific disciplines (SCIENCE DIRECT, ISI WEB OF KNOWLEDGE) as well as emphasize Computer Science (SPRINGER) and Economics (EBSCO) and in sum cover approx. 950.000 journals, books and conferences. We used the following query:

```
("business process" OR "process model") AND ("recommenda-
tion system" OR recommender OR "modelling recommendation"
OR "element recommendation" OR "model recommendation" OR
"recommendation of model" OR "recommendation of element"
OR "auto-completion" OR "content assist").
```

Table 2 lists the papers that we retrieved with the query in conjunction with the settings we used when executing the query on the respective database.

Table 1. Overview of the literature search

Source	*Restriction and settings*	*Hits*
Retrieval of relevant papers		
Springer Link	Include Preview-Only content	895
Science Direct	All Sources	420
EBSCO Host	Databases: All; Field: TX All Text	116
ISI Web of Knowledge	Field: Topic	9
Consolidation, selection and review of papers		
Hits from all hosts	Selection of relevant papers from the hits	20
Addition of papers	Extension of the result set by forward- and backward search	22
	Sum of reviewed papers:	**42**

In the next step of our literature analysis we read all the titles and abstracts of the papers retrieved and only selected papers for an in-depth analysis that are relevant for the topic and not only satisfy the query, leading to 20 papers. These papers were used to extend the result set by a forward- and backward search [11] leading to an additional set of 22 relevant papers. After an in-depth analysis of the 42 relevant papers, we arrived at a total of 28 pertinent works that were in line with our previously stated aims. Articles that *directly* state requirements for PMRSs are marked with ◆ in the rightmost column "R", those with indirect statements with ◇.

Table 2. Overview of the relevant literature

Ref.	*Year*	*Main content and focus of the research work*	*R*
[6]	2014	Pattern-based auto-completion of UML modeling activities	◆
[14]	2014	Recommendation Method for Improving Business Process Modeling	◇
[15]	2013	Development of an Auto-Suggest Component for Process Modeling Tool	◆
[16]	2013	Bayesian Networks for Recommendations in Business Process Modeling	◇
[17]	2012	Action patterns in business process models	◆
[18]	2012	SemReuse – Semantics-based Reuse of Business Process Models	◆
[19]	2012	Optimized execution of business processes through recommendations	◇
[20]	2012	Action patterns in business process model repositories	◇
[21]	2011	Autocompletion for Business Process Modelling	◆
[22]	2011	Context-based service recommendation for assisting business process design	◇
[23]	2010	Supporting process design for e-business via an integrated process repository	◆
[24]	2010	Towards a Framework for Business Process Models Reuse	◆
[25]	2010	Action patterns in business process model repositories	◆
[26]	2010	Self-adjusting recommendations for people-driven ad-hoc processes	◇

[27]	2009	Business process models with syntax-based assistance in diagram editors	◆
[28]	2009	Social software for modeling business processes	◇
[29]	2009	Advanced social features in a recommendation system for process modeling	◇
[30]	2009	Real support for perspective-compliant business process design	◇
[31]	2009	Auto-completion for executable business process models	◇
[32]	2009	Advanced social features in a recommendation system for process modeling	◇
[33]	2008	Auto-completion for Diagram Editors based on Graph Grammars	◆
[34]	2008	Recommendation based process modeling support	◇
[35]	2007	Similarity-based support of process modeling	◆
[36]	2007	Pattern-based knowledge workflow automation: concepts and issues	◇
[37]	2007	Measuring similarity between semantic business process models	◆
[38]	2006	Automatic user support for business process modeling	◇
[39]	2002	Hybrid Recommender Systems Survey and Experiments	◇
[40]	1997	Content-based, collaborative recommendation	◇

3.2 Elicitation of Requirements from Literature

In the following, we summarize the outcome of our literature analysis in synthesizing a list of requirements grounded in the available literature. To begin with, we observed that requirements fall into three broad categories: Requirements in regard to the *content* that is recommended, the *recommendation capabilities* and the *recommendation system*.

Content-related requirements address what is recommended by the PMRS and relate to the *type* of and *meta-information* about recommendations. Regarding the *type*, a distinction between elements and the structure of a model can be made [16]. In addition, also labels that are partially typed during modeling may be completed by the system [21]. Moreover, other elements such as resources may be suggested [16].

RL1. *Recommendation of basic process model constructs.* The system should be able to recommend constructs such as elements, their structure and labels.

RL2. *Recommendation of additional process model constructs.* The system should provide recommendations for other constructs such as resources used in a business process.

Regarding meta-information, relevant information items are described in respect to model repositories [25]. They may range from a textual description, business context, supported goals (e.g. regarding execution speed, quality improvement or cost reduction), resources used (e.g. services, goods or money) or actors involved (e.g. customers, suppliers) to the formal classification of a suggested element according to different classification schemes. Another form of meta-information is provenance information. This information should provide insights on the level of authority of a recommendation (e.g. the recommendation conforms to accepted standards or is endorsed by experts) [35] or who already reused this model or a part of it [32].

RL3. *Provide descriptive meta-information about the recommendations.* The system should provide relevant meta-information about the suggested elements.

RL4. *Provide provenance information about the recommendation.* The system should provide information to judge the quality of the recommendation.

Capability-related requirements address the functionality of the recommendation feature and relate to the *semantic quality*, the *flexibility and ease of use during modeling*. Further they cover *personalization* and *customizability* via settings as well as *multi-method techniques* to calculate the recommendations.

Regarding the *semantic quality*, the system should avoid suggesting redundant elements already present in the model [39] or that lead to erroneous models [37]. Moreover, recommendations should be in a meaningful behavioral order (e.g. and activity “check” should occur in the possible execution traces of the process model before the activity “approve”, not vice versa) [17] [31]. Moreover, they should comply with domain specific policies and rules [17] [37] and be provided with the right level of granularity [24] [25] [14]. Concerning the *flexibility and ease of use during modeling*, different recommendation modes such as forward completion, backward completion and auto-completion [16] should be offered. They may be complemented with additional assistance features such as guidance for the selection of recommendations [13] or a pre-view feature [6]. Further, the user should be free to select her preferred modeling orientation (from left to right or top to bottom) [35]. Regarding an easy application of recommendations, *personalization* may also contribute to the overall usability of the system and may require the system to observe the users intention (e.g. topic of the model under construction, the desired abstraction level) and to take previously constructed models into consideration in order to provide personalized recommendations [31].

RL5. *Ensure recommendations with a high semantic quality.* The system should provide recommendations that are adequate and lead to a high semantic model quality.

RL6. *Flexible and easy application of recommendations.* The system should make it easy to work with recommendations and may guide the user in the selection of suggestions.

RL7. *Use personalization mechanisms.* The system should provide personalized recommendations tailored to the needs of the specific user in her specific modeling situation.

Regarding *customizability*, the user may customize the system by adjustments that directly affect the system behavior [6] such as the maximum number of recommendations and the elements displayed per recommendation, which should not exceed 15 elements [35]. In addition, the filtering options for recommendations should be highly adjustable. They may be filtered according to various meta-information about the recommended model element(s) such as their frequency in the knowledge base, the date of the last insertion of an element, previous uses of the suggestion by other users after recommendation and the similarity of the model element labels compared to labels in the knowledge base [15]. Finally, *multi-method techniques* provide multiple recommendation strategies in order to fit the users requirements [30] [21].

RL8. *Adjustable filtering options for recommendations.* The User should be able to adjust the filtering criteria for recommendations.

RL9. *Adjustable amount of recommendations.* The User should be able to adjust the amount of recommendations.

RL10. *Multiple recommendation strategies.* Recommendations should be determined using different calculation strategies in order to fit the user requirements.

System-related requirements comprise a wide spectrum of requirements addressing amongst others the *evolution of the knowledge base* as well as more *general non-functional requirements*. Regarding the *evolution of the knowledge base*, the addition of new knowledge is important. It can be accomplished by importing existing models into the knowledge base [15] or leveraging more general knowledge in process design [21] such as standardized models. Another option is to learn new elements or models [15] e.g. from user behavior [38] or by training approaches [16]. Equally important is the support of changes e.g. via automatic addition or deletion of elements in the process models [24] or change notifications [32] for inserted model structures. A version management of the knowledge base should provide for the possibility to customize it for different projects or customers [25]. In respect to compatibility, the system should either be independent of a specific modeling language (i.e. general-purpose) or tailored to it (i.e. recommendations consider the syntax and semantics of the modeling language in use) [15]. Moreover, interfaces to existing tools should be provided [15].

RL11. *Support knowledge base evolution.* The system should provide capabilities such as versioning, change management, importing new content or learning.

RL12. *Compatibility to existing tools and languages.* The system should work with existing modelling languages and in conjunction with existing tools (e.g. as plug-in).

More *general non-functional requirements* are to a large part not specific to PMRSs (e.g. the typical "cold start" problem of recommender systems [39]) or even to recommender systems as a whole although being discussed in this context (e.g. scalability [35] [6], usability [24] [40] [33], extensibility [18], data protection [15], generic tool development approaches [27]). We therefore do not enumerate them in the context of this work.

4 Practitioners Perception of Recommendation Features

4.1 Results from the Survey

The first relevant study addressed general features of a PMRS. The study has been conducted in 2011 and mainly analyzed the status of process model reuse from a user's perspective. However, the interviewees have also been asked to answer three questions, which address PMRSs as a way to facilitate process model reuse. The foundation of the survey has been a structured and standardized online questionnaire. To attract interviewees we spread the link of the online questionnaire to a popular German BPM network, forum and a widely-used mailing list of business informatics (www.bpm-netzwerk.de, www.bpm-forum.net). The addressees of the questionnaire were experts with long-standing process modeling experience.

Overall, the questionnaire has been answered by 47 people, 48.9% being researchers (23 persons), 40.4% practitioners (19 persons) and 10.6% others (5 persons, e.g., people in education). The respondents have used the following tools for process modeling (multiple selection was possible): ARIS platform (40.4%), MS Visio (31.9%), Signavio (14.9%), Adonis (8.5%), MS PowerPoint (6.4%), MS Excel (6.4%) and BizAgi (4.3%). 26 other tools were indicated only once, which are not mentioned one by one. With respect to the process modeling notation, the following have been used (multiple answers were possible): EPC (78.7%), BPMN (65.9%), UML (44.7%), BPEL (10.6%) and others (2.2%). Also, the use of reference models was quite common. For instance, 31.9% used the SAP R3 model, 23.4% used ITIL3 and 34% an enterprise-internal reference model while 25.5% have never used a reference model (multiple answers were possible). To implement appropriate process modeling support functions, we were interested about the *need* for such a feature and *how* such function should be implemented. The results are given in Table 3.

Table 3. Results of questions about how modeling assistance functions should be applied

Need for a PMRS feature?	
No. I am satisfied with current tools	14.7%
Yes. Suggest single process elements.	21.7%
Yes. Suggest patterns of process fragments.	33.8%
Yes. Suggest similar process models.	24.3%
Other.	5.4%
What kind of source to use as input data for the PMRS?	
Process models out of a repository.	24.8%
Best practices, reference models, laws.	37.1%
Other.	38.1%
How to implement a PMRS feature?	
During model construction displaying support in process model.	14.8%
During model construction displaying in separate window.	22.2%
Only on request displaying support in process model.	33.3%
Only on request displaying in separate window.	29.6%

This analysis shows a clear preference for a PMRS. Particularly, it is preferred to suggest patterns of process fragments (e.g., a group of process elements connected via control flow patterns) (33.8%), followed by suggestions of similar process models (24.3%) and finally single process elements (21.6%). Comparing the preference of the source of the process modeling support feature, then best practices (which might be retrieved online) are preferred over process models from a repository (which are mainly created by previous users). Many respondents selected the option “others” (which was a free text box), however, without making any suggestions for other sources. If a process model support feature is offered by a tool, then a clear preference is given for assistance on request (62.9%) instead of a pro-active provision of recommendations (i.e. “live”) during model construction (37%).

The following observation can be seen from this survey. The respondents clearly expressed a need for a PMRS. They would prefer to suggest a connected group of process elements, which has as source commonly and widely used process models (best practices, reference models). One reason for this preference might be a wide use of BPM tools for business process reengineering (where reference models are applied). Also, users do not wish to be bothered or interrupted by new functions during process modeling. Instead, a PMRS offering its features upon request is preferred. Based on these insights, the following requirements have been derived.

RS1. *Various sources for recommendations.* The recommendation system should be able to generate recommendations from various sources.

RS2. *Provenance information.* The recommendation system should provide background information regarding the source and quality of a recommendation.

RS3. *Display of recommendations on request.* Recommendations should be provided when the user requests the system to do so.

RS4. *Multiple ways of displaying recommendations.* The recommendation system should provide multiple ways of displaying the recommendations varying in their degree of non-obtrusiveness.

4.2 Results from the Case Study

The second relevant study that has been conducted involves participants in a case study. The participants were undergraduate students in the 6th semester with an equal share studying Economics and Information Systems. The participants had to create models using a standardized pre-defined set of process activities. Although the activities were not automatically suggested but had to be retrieved manually by browsing an extensive taxonomy of functions, this allows us to shed light on how users perceive modelling with pre-defined process activities.

In the case study, 52 participants were involved in 2013 and 48 participants in 2014. The participants had to create BPMN models in 11 groups with 5-6 participants in each group. The topic of the models has been product management, human resources and change management. The amount of model elements was prescribed to be in a range of 6 to 24 process elements in order to get comparable models. The participants had to annotate the models with pre-defined activities from the Process Classification Framework (PCF), a collection of more than 1000 enterprise activities arranged in a four-level taxonomy with twelve basic process categories. When the modelling task was finalized, participants were asked discussing arguments in favour and against restricting process modelling to pre-defined standardized activities. Each group produced a text of between one and three pages. For qualitative text analysis, the texts have been fragmented in single arguments. In addition, each participant was requested (as part of a written exam) to provide at least three justified arguments in favour and against using standardized activities in process modelling. These arguments were also captured and added to the list of arguments. In total, 314 arguments (positive and negative) have been collected.

Data analysis has been conducted by two researchers according to the procedure described by MAYRING [41] for qualitative text analysis. In essence, we inductively built categories by applying two abstraction steps on the original statements. These consisted in rewriting the essential semantic content of a sentence in a paraphrase and then finding a suitable category subsuming it. In the first step, one researcher built categories on the arguments originating in the group discussions while the other researcher built categories based on the arguments given by the participants in the exam. In the second step, the inductively created categories have been compared. It turned out that except one category, the two researchers independently created semantically very similar categories with slight variation in naming. After discussing the different names, a consensus regarding the category names was reached (cf. Fig. 2).

ID	Positive Arguments	Amount	Rel. Fequency
P1	Understandability is increased	52	17%
P2	Basis for improved software and benchmarking	29	9%
P3	Process of modelling is improved	28	9%
P4	Internationalisation is facilitated	25	8%
P5	Comparability is improved	19	6%
P6	Relevancy of model contents increases	11	4%
	Negative Arguments		
N1	Pre-defined activitites are incomplete/inadequate	47	15%
N2	No positive impact on competition	33	11%
N3	Loss of flexibility while modelling	31	10%
N4	Effort of modelling increases	31	10%
N5	Irrelevant activities are inserted into the model	8	3%
	Overall amount of statements	314	

Fig. 2. Positive and negative arguments derived from the case study data

Fig. 2 shows the set of consolidated categories derived inductively from the 314 arguments contained in the data. The most frequent argument in favour of restricting modelling to pre-defined activities was that understandability is increased due to an unambiguous naming of the model elements activities acting as a shared terminological basis for all participants that jointly worked together in a group. The most frequent argument opposed to restricting modelling to pre-defined activities was that the set of activities has not been complete since some activities have been missing.

The derivation of requirements from the consolidated 314 positive and negative arguments from the 48 case study participants has been straightforward: From the positive aspects P1-P6, requirements have been derived that when satisfied embody the respective aspect in a functionality of the PMRSs. From negative aspects, requirements have been derived that lead to functionalities that help to mitigate the negative aspect. Each of the two researchers that previously analysed the data of the case study independently proposed a requirement. Afterwards, the requirements have been consolidated to a single proposition. This procedure lead to the mapping of the positive aspects to the following requirements: P1 in RC1; P2 in RC4; P3, N2 in RC2; P4, P5 in RC5. The negative aspects were mapped as follows: N1 in RC3; N2 in RC2; N3, N5 in RC6 (the mapping of N4 is obsolete since this negative aspect results from the process of manually finding relevant activities in the case study).

RC1. *Understandable recommendations.* Since one main positive aspect of using standardized activities has been that their interpretation is less ambiguous, the PMRSs should use such standardized activities to ease the understanding of the recommendations.

RC2. *Recommendation of "uncommon", innovative contents.* For example, the system may suggest activities that are executed typically in another industry and in that way inspire the process design that in turn may exert a positive impact on competition.

RC3. *Extension capability of the pre-defined contents.* To provide a remedy for missing activities, the recommendation system should include a feature to extend the internal knowledge base.

RC4. *Benchmarking feature.* The system should facilitate benchmarking e.g. by suggesting Key Performance Indicators (KPI) or by enabling a comparison of KPI values.

RC5. *Advanced model processing features.* The system should offer advanced features for the translation of models in multiple languages (e.g. process taxonomies such as PCF exist in different languages), to compare models or to show which area of enterprise activities they cover based on their semantics.

RC6. *On/off switch and decent presentation of recommendations*. The recommender system should not tempt the user to insert recommendations that do not match his/her modeling intention or exert any pressure on the modeler. Therefore, the system should be switched off easily and the recommendations should be presented decently.

4.3 Results from Assessing a Prototype with an Recommendation Feature by Business Users

To complete the understanding from the business user perspective, this section presents survey results of a live-demonstration of a prototypical implementation. The live-demonstration and the survey took place in 2013 at the world's largest computer trade fair CeBIT in Hannover, Germany. Participants were able to have a look at a prototype during 4 days and the concept of recommendations in process modelling. The prototype system comprised a simple web application capable of offering suggestions for a selected activity (usually the last activity that has been added to the model). Recommendations are calculated based on the Process Classification Framework (PCF). It was also possible to use this recommendations mechanism inside a process modeling tool. For this purpose, we extended Microsoft Visio with a simple recommendation feature. With respect to the intention of this paper (requirements catalog), the characteristics and design decisions of the prototype system are not in the center of interest and thus not further considered. After working with both prototypes (Microsoft Visio extension and the web application), participants were asked to give feedback.

Overall the survey was completed by 66 participants, whereof most of the participants were graduates (74.2%). More precisely, a high majority of 66.6% (44 respondents) had a university degree, accompanied by 7.6% with a PhD (5 respondents). A university-entrance diploma was possessed by 18.2% (12 respondents). Only a small number of 6.1% had no university education, but had a vocational education (4 respondents). One participant (1.5%) did not provide an answer with respect to the educational background.

Besides the feedback part of the live-demonstration of the prototypical implementation – which is not in the focus of this paper – also some questions about requirements were included in the questionnaire. Two questions focused on giving certain preferences to requirement issues and additionally a fully open-ended question was included. From an architectural point of view, we were interested to find out the preferences regarding how the tool should be provisioned. We had in total 58 valid answers to this question. A cloud-based Software-as-a-Service solution was ranked first (41.4%). However, for reasons of security many respondents opted for a pure desktop solution (27.6%) or the installation on an own server in the firm (31.0%). Another question was about additional requirements, particularly about elements that should additionally be suggested in a recommendation-based modeling tool. Multiple answers were possible. It is interesting to note that only 14.1% of the respondents to this question stated that no further recommendations are needed (which is in line with the first study in favor of a modeling support presented in Section 4.1). 32.8% wished for a possibility of displaying possible business partners and 71.9% wished for a possibility of displaying time and costs of activities. However, 12.5% expressed the wish for other suggestions. Finally, the fully open-ended question tried to elicit additional important things to be considered in this context.

In total, through the requirements part of the survey 33 suggestions for additional requirements were gathered, where after summarizing 8 were relevant. The criteria for relevance were based both on the question whether they focus on the recommending part of the system (because some focused more on the modeling, frontend or other parts of the system) and on the fact whether they had already been mentioned (to avoid duplicates). The elicited requirements are the following.

- RP1. *Recommendation of organizational units.* The system should recommend additional elements such as organizational units executing the activities.
- RP2. *Recommendation of resources.* The system should recommend resources such as documents, tools or information systems.
- RP3. *Customized specific taxonomies.* To make sure a plethora of potential use cases is covered, the predefined contents in the system should be customizable.
- RP4. *Mobile version of the recommender.* Due to the fact that an increasing amount of work is done on the go, a mobile version should be offered.
- RP5. *Interface to other systems.* Data inside the PMRSs used for recommendations such as taxonomies of pre-defined activities or organizational units should be updated frequently via interfaces to systems containing that data.
- RP6. *Support multiple platforms.* As there are different platforms and architectures used in companies the support of the most important of them is needed to make sure the system gains acceptance.
- RP7. *„Intelligent recommendations".* This requirement is more an overall characteristic of the whole system and demands that recommendations should be made on the right time in the right manner with adequate content.
- RP8. *Show recommendation context.* The user of the system should be informed about the semantic context of a recommendation that is offered.

5 Consolidation and Discussion of the Findings

In order to provide a catalog of requirements, we consolidated the plethora of requirements to a final list of requirements. At first we combined redundant requirements and then we detected and consolidated requirements that are subsumed by others. Finally, we further classified the requirements as being *functional* (FUNC), *non-functional* (NFNC), *architectural* (ARCH) or *data-related requirements* (DATA). Table 4 shows the result of this research step.

Table 4. Consolidated PMRS requirements catalog.

Req. No.	*Name of the consolidated requirement*	*Source-Requirement*	FUNC	NFNC	ARCH	DATA
R01	Recommendation of basic constructs	RL1	■			■
R02	Recommendation of additional objects	RL2, RP1-2	■			■
R03	Innovative and intelligent recommendations	RC2, RP7		■		
R04	Provision of context and meta-information	RL3-4, RC1, RS2, RP8	■			■
R05	Quality and relevance of recommendations	RL5, RL10		■		■
R06	Easy handling of the recommendations	RL6, RL8-9, RS3-4, RC6		■		
R07	Personalized recommendations	RL7	■			
R08	Knowledge base management and evolution	RL11, RP3, RC3	■			■
R09	Advanced features	RC4-5	■			■
R10	Multiple interfaces and platforms	RL12, RP4-6, RS1	■		■	

We decided for this classification for the following reasons. The distinction between functional and non-functional requirements is well known in systems and software engineering. However, we additionally distinguish between requirements concerning the data since these are an important precondition of a PMRS as well as requirements concerning the architectural perspective. The latter ones are relevant in respect to the provisioning of the system.

What can be seen when looking at Table 4 is that the distribution of source requirements according to their type being one of RC, RL, RP or RS is not equal. One requirement was detected exclusively by analyzing the case study and three exclusively by the literature analysis. Seven requirements were detected by two or more types of source requirements. Only one requirement was detected by all four types. It thus can be concluded, that the derivation of requirements from different sources such as the literature analysis and the survey, the case study and the prototype presentation in fact is valuable and leads to a more holistic elicitation of requirements.

6 Summary

Although sophisticated modeling tools exist, guidance in process modelling in terms of auto-completion and recommendation features is largely missing even in today’s tools. In this contribution, we therefore systematically collected requirements for such features as a first step towards the stepwise iterative development of PMRSs guiding the modeler in modelling. We derived the requirements deductively from literature as well as inductively by three empirical studies conducted within two years that involved both practitioners and students. The involvement of students always raises discussions about the external validity. However, having both groups in order to get a more balanced and holistic view may justify this [42]. Finally, we hope that our requirements catalog may be useful and serve as a point of reference both for researchers and the industry engaged with the development of PMRSs.

References

1. Luftman, J., Zadeh, H.S., Derksen, B., Santana, M., Rigoni, E.H., Huang, Z. (David): Key information technology and management issues 2012–2013: an international study. J. Inf. Technol. 28, 354–366 (2013).
2. Sarshar, K., Weber, M., Loos, P.: Einsatz der Informationsmodellierung bei der Einführung betrieblicher Standardsoftware : Eine empirische Untersuchung bei Energieversorgerunternehmen. Wirtschaftsinformatik. 48, 120–127 (2006).
3. Nielen, A., Költer, D., Mütze-Niewöhner, S., Karla, J., Schlick, C.: An Empirical Analysis of Human Performance and Error in Process Model Development. Proc. of Conceptual Modeling (ER 2011), Brussels, Belgium. pp. 514–523 (2011).
4. Wilmont, I., Brinkkemper, S., Weerd, I., Hoppenbrouwers, S.: Exploring Intuitive Modelling Behaviour. In: Bider et al. (eds.) Enterprise, Business-Process and Information Systems Modeling : Proceedings of BPMDS 2010 and EMMSAD 2010 held at CAiSE 2010, June 7-8, Hammamet, Tunisia. pp. 301–313. Springer, Berlin (2010).
5. Sen, S., Baudry, B., Vangheluwe, H.: Towards Domain-specific Model Editors with Automatic Model Completion. SIMULATION. 86, 109–126 (2010).
6. Kuschke, T., Mäder, P.: Pattern-based Auto-completion of UML Modeling Activities. Proceedings of the 29th ACM/IEEE International Conference on Automated Software Engineering. pp. 551–556. ACM, New York, NY, USA (2014).
7. Melville, P., Sindhwani, V.: Recommender Systems. In: Sammut, C. and Webb, G.I. (eds.) Encyclopedia of Machine Learning. pp. 829–838. Springer US (2010).
8. Lauesen, S.: Software Requirement: Styles and Techniques. Addison Wesley, Harlow, England (2002).
9. Koschmider, A., Hornung, T., Oberweis, A.: Recommendation-based editor for business process modeling. Data Knowl. Eng. 70, 483–503 (2011).
10. Davis, A.M., Hickey, A.M.: Requirements Researchers: Do We Practice What We Preach? Requir. Eng. 7, 107–111 (2002).
11. Webster, J., Watson, R.T.: Analyzing the past to prepare for the future: Writing a literature review. MIS Q. 26, xiii–xxiii (2002).
12. Vom Brocke, J., Simons, A., Niehaves, B., Riemer, K., Plattfaut, R., Cleven, A.: Reconstructing the Giant: On the importance of rigour in documenting the literature search process. Proceedings of the 17th European Conference on Information Systems (2009).
13. Saunders, M., Lewis, P., Thornhill, A.: Research Methods for Business Students. Pearson Education Limited, Essex (2009).
14. Li, Y., Cao, B., Xu, L., Yin, J., Deng, S., Yin, Y., Wu, Z.: An Efficient Recommendation Method for Improving Bus. Proc. Modeling. IEEE Trans. Ind. Inform. 10, 502–513 (2014).
15. Clever, N., Holler, J., Shitkova, M., Becker, J.: Towards Auto-Suggested Process Modeling Prototypical Development of an Auto-Suggest Component for Process Modeling Tools. Proc. of EMISA 2013, pp. 133–145. Köllen, Bonn (2013).
16. Bobek, S., Baran, M., Kluza, K., Nalepa, G.J.: Application of Bayesian Networks to Recommendations in Business Process Modeling. In: Giordano et al. (eds.) Proc. of the Workshop AI Meets Business Processes 2013 co-located with the 13th Conf. of the Italian Association for AI (AI*IA 2013) Turin, Italy, December 6, 2013. pp. 41–50 (2013).
17. Smirnov, S., Weidlich, M., Mendling, J., Weske, M.: Action patterns in business process model repositories. Comput. Ind. 63, 98–111 (2012).
18. Koschmider, A., Oberweis, A., Schoknecht, A.: SemReuse–Semantikbasierte Wiederverwendung von Geschäftsprozessmodellen. (2012).
19. Barba, I., Weber, B., Del Valle, C.: Supporting the optimized execution of business processes through recommendations. BPM Workshops. pp. 135–140 (2012).
20. Smirnov, S., Weidlich, M., Mendling, J., Weske, M.: Action patterns in business process model repositories. Comput. Ind. 63, 98–111 (2012).

21. Wieloch, K., Filipowska, A., Kaczmarek, M.: Autocompletion for Business Process Modelling. In: Abramowicz, W. (ed.) Proc. of BIS 2011, June 15-17, Poznan, Poland. pp. 30–40. Springer, Berlin (2011).
22. Chan, N.N., Gaaloul, W., Tata, S.: Context-based service recommendation for assisting business process design. E-Commerce and Web Technol. pp. 39–51. Springer (2011).
23. Wang, H.J., Wu, H.: Supporting process design for e-business via an integrated process repository. Inf. Technol. Manag. 12, 97–109 (2010).
24. Derguech, W.: Towards a Framework for Business Process Models Reuse. In: Persson, A., Bentallah, Boualem, and Guabtni, Adene (eds.) The CAiSE Doctoral Consortium 2010, Hammamet, Tunisia, June 8, 2010. p. paper3, 10 pages (2010).
25. Shahzad, K., Elias, M., Johannesson, P.: Requirements for a business process model repository: A stakeholders' perspective. Business Information Systems. pp. 158–170 (2010).
26. Dorn, C., Burkhart, T., Werth, D., Dustdar, S.: Self-adjusting recommendations for people-driven ad-hoc processes. In: Proc. of BPM 2010. pp. 327–342. Springer (2010).
27. Mazanek, S., Minas, M.: Business Process Models as a Showcase for Syntax-Based Assistance in Diagram Editors. In: Schürr, A. and Selic, B. (eds.) Model Driven Engineering Languages and Systems. pp. 322–336. Springer Berlin Heidelberg (2009).
28. Koschmider, A., Song, M., Reijers, H.A.: Social software for modeling business processes. Business Process Management Workshops. pp. 666–677 (2009).
29. Koschmider, A., Song, M., Reijers, H.A.: Advanced social features in a recommendation system for process modeling. Business Information Systems. pp. 109–120 (2009).
30. Koschmider, A., Habryn, F., Gottschalk, F.: Real support for perspective-compliant business process design. Business Process Management Workshops. pp. 32–43 (2009).
31. Born, M., Brelage, C., Markovic, I., Pfeiffer, D., Weber, I.: Auto-completion for executable business process models. In: Proc. of BPM Workshops 2009. pp. 510–515. Springer, Berlin (2009).
32. Koschmider, A., Song, M., Reijers, H.A.: Advanced social features in a recommendation system for process modeling. Business Information Systems. pp. 109–120 (2009).
33. Mazanek, S., Maier, S., Minas, M.: Auto-completion for diagram editors based on graph grammars. IEEE Symposium VL/HCC 2008. pp. 242–245 (2008).
34. Hornung, T., Koschmider, A., Lausen, G.: Recommendation based process modeling support: Method and user experience. Proc. of ER 2008. pp. 265–278. Springer (2008).
35. Koschmider, A.: Ähnlichkeitsbasierte Modellierungsunterstützung für Geschäftsprozesse. Dissertation at the University of Karlsruhe (2007).
36. Sarnikar, S., Zhao, J.L.: Pattern-based knowledge workflow automation: concepts and issues. Inf. Syst. E-Bus. Manag. 6, 385–402 (2007).
37. Ehrig, M., Koschmider, A., Oberweis, A.: Measuring similarity between semantic business process models. In: Link, S. and Ghose, A. (eds.) Proc. of APCCM2007, Ballarat, Victoria, Australia. pp. 80–89. Australian Computer Society, Sydney (2007).
38. Betz, S., Klink, S., Koschmider, A., Oberweis, A.: Automatic User Support for Business Process Modeling. In: Hinkelmann, K. et al. (eds.) Proc. of SBPM 2006, Budva, Montenegro, June 2006. pp. 1–12 (2006).
39. Burke, R.: Hybrid Recommender Systems: Survey and Experiments. User Model. User-Adapt. Interact. 12, 331–370 (2002).
40. Balabanović, M., Shoham, Y.: Fab: Content-based, Collaborative Recommendation. Commun ACM. 40, 66–72 (1997).
41. Mayring, P.: Qualitative Inhaltsanalyse. In: Mey, G. and Mruck, K. (eds.) Handbuch Qualitative Forschung in der Psychologie. pp. 601–613. VS Verlag für Sozialwiss. (2010).
42. Svahnberg, M., Aurum, A., Wohlin, C.: Using Students As Subjects - an Empirical Evaluation. Proc. of the Second ACM-IEEE International Symposium on Empirical Software Engineering and Measurement. pp. 288–290. ACM, New York, NY, USA (2008).