Requirements Eng
DOI 10.1007/s00766-012-0148-1

ORIGINAL ARTICLE

Uni-REPM: validated and improved

Mikael Svahnberg · Tony Gorschek · Thi Than Loan Nguyen · Mai Nguyen

Received: 22 September 2011 / Accepted: 14 February 2012

Abstract Software products are usually developed for either a specific customer (bespoke) or a broader market (market-driven). Due to their characteristic, bespoke and market-driven development face different challenges, especially concerning requirements engineering. Many challenges are caused by an inadequate requirements engineering process, and hence there is a need for process improvement frameworks based on empirical research and industry needs. In a previous article we introduced Uni-REPM, a lightweight requirements engineering process assessment framework based on a review of empirically motivated practices in market-driven and bespoke requirements engineering literature. In this article, we validate this framework in academia as well as industry, in order to prepare Uni-REPM for widespread industry use. We conduct two validations; a static validation based on interviews with seven academic experts and a dynamic validation where Uni-REPM is applied in four industrial organisations. Uni-REPM is refined according to the feedback obtained in the validations. The study shows that Uni-REPM is a quick, simple, and cost-effective solution to assess the maturity level of the requirements engineering process of projects. Moreover, the assessment method using checklists is highly usable and applicable in various international development environments.

Keywords Requirements engineering · Process assessment · Lightweight · Empirical validation

M. Svahnberg (✉) · T. Gorschek · T. T. L. Nguyen · M. Nguyen
Blekinge Institute of Technology, 371 79 Karlskrona, Sweden
e-mail: Mikael.Svahnberg@bth.se

T. Gorschek
e-mail: Tony.Gorschek@bth.se

T. T. L. Nguyen
e-mail: thanhloan2812.nguyen@gmail.com

M. Nguyen
e-mail: mainguyentp@gmail.com

1 Introduction

The main idea of process improvement frameworks such as CMM, CMMI, and ISO9000 [4, 38] is to assess the current state of processes in an organisation in order to detect existing problems and to provide an improvement path for the organisation and projects within the organisation.

It has been shown that significant business benefits could be achieved by preventing problems as early as the requirements engineering (RE) phase instead of waiting until the project finished [33]. For example, Hall et al. [16] reports that a large proportion (48%) of development problems stem from problems with the requirements. Moreover, fixing requirements-related problems consumes a high cost of rework in later states [3, 26].

However, despite its important role, research on industrial projects still indicates poor RE practices [1, 14, 16, 21, 29–31]. Problems reported include that the organisations lack well-defined processes and guidelines for using tools, methods, there is little user involvement in the processes, traceability is usually overlooked, and almost none of the available modelling techniques are used [21, 28]. In market-driven RE, the vast number of stakeholders makes it difficult to elicit and manage the requirements, especially since the mass of requirements is continuously expanding and requirements may be stated on different levels of abstraction [10, 24]. Moreover, the requirements are often volatile and changed [1], and there is a need to balance between market pull and technology push [10].

There exist several requirements engineering process improvement frameworks aiming at bridging the gap between best practices and practised best, for example the Good Practice Guide by Somerville and Sawyer [34], and the requirements engineering process maturity model (REPM) [12, 13, 31]. Process assessment frameworks such as CMMI [4] and SPICE [35] also cover requirements engineering, although only shallowly since the scope of these frameworks is much bigger than just requirements engineering. Common for these frameworks are that they are focusing on bespoke requirements engineering and that they have not evolved along with requirements engineering practices in industry. Hence, there are practices not covered at all by these frameworks (for example, practices related to market-driven requirements engineering), and other practices are ranked as being very advanced whereas in contemporary state of practice they are the common norm. Attempts have been made to introduce process assessment frameworks for market-driven requirements engineering, for example MDREPM [11]. However, these attempts usually focus too much on market-driven requirements engineering and thus makes the framework unusable in a bespoke setting.

To this end, Uni-REPM is introduced, in a previous publication,[1] as a modern, light-weight requirements engineering process assessment framework based on empirically proven good requirements engineering practices in both market-driven and bespoke requirements engineering settings. Even if Uni-REPM is created based on empirically validated best practices for market-driven and bespoke requirements engineering, there is still a need to validate whether the collection of practices work well together and whether the structure of the framework is usable. This validation is the focus of this article. We do this in two steps; first we perform a static validation by interviewing a set of domain experts, and then we do a dynamic validation where Uni-REPM is applied in several industry projects. The framework is modified accordingly, and the end result is a refined framework where the correctness, completeness, and applicability are empirically validated.

The remainder of this article is organised as follows. In Sect. 2 the research questions guiding the work are presented, followed by a background on requirements engineering process improvement and a brief presentation of Uni-REPM in Sect. 3. In Sects. 4 and 5 the static and dynamic validation of Uni-REPM is presented. This is followed by a discussion of validity threats in Sect. 6. The updated version of Uni-REPM is presented in Sect. 7, and the paper is concluded in Sect. 8.

[1] Under submission. Can be obtained from the main author.

2 Research questions

The following research questions are formulated for this study:

- *RQ1* To what extent is Uni-REPM suitable for industrial piloting, in terms of its correctness, completeness, and applicability?
- *RQ2* To what extent is Uni-REPM applicable , usable, and useful for industry application?
- *RQ3* What improvements can be done to Uni-REPM based on the findings in RQ1 and RQ2?

These research questions loosely follow the technology transfer framework presented by Gorschek et al. [9], as presented in Fig. 1. In this figure we see how a problem is identified and formulated together with industry, after which a candidate solution is devised. For Uni-REPM, these steps are covered in a previous publication, and the subsequent steps are covered in this article: The candidate solution is validated in academia and through static industry validation, which maps to RQ1. After this, the candidate solution is dynamically validated in industry, which maps to RQ2. Throughout this process, the candidate solution is refined, which maps to RQ3.

3 Background

3.1 Requirements engineering challenges

Requirements engineering constitutes the first stepping-stone in software development, since it generates the input that governs subsequent development and testing of a software product. This has long been recognised in software engineering research as well as in software engineering practice, and a number of studies show that the most cost-efficient place to correct many problems is in the requirements engineering phase [3, 26, 33]. Despite this, requirements engineering remains a neglected area [16, 21, 27, 30]. Challenges with requirements engineering include vague initial requirements, requirements specification, undefined requirements process, requirements growth, requirements traceability, and confusion between methods and tools [16, 21, 22].

In addition to the aforementioned challenges that traditional requirements engineering faces, market-driven requirements engineering introduces or emphasises new challenges [15, 23]. For example, requirements elicitation becomes a challenge of not only asking the right questions

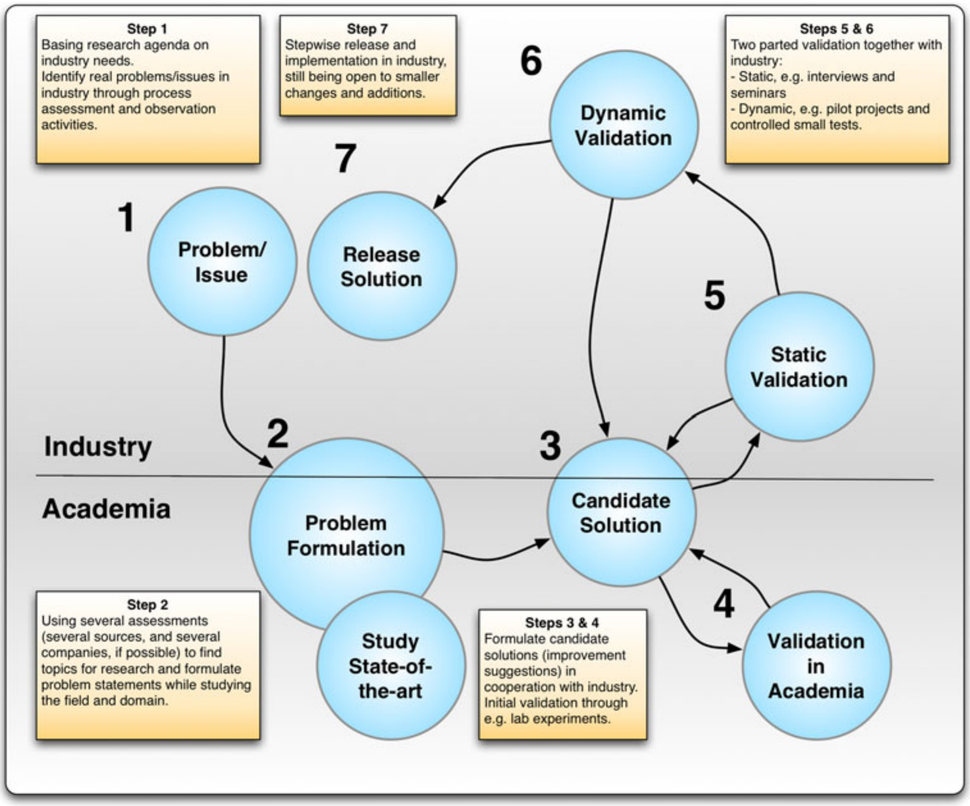


Fig. 1 Technology transfer framework [9]

in the right way, but also finding the stakeholders and gaining access to them in the first place [23]. Given the number of potential sources of requirements, the organisations risk being overloaded with requirements [23], and thus, the mechanisms are needed for triage [6] of requirements. The different requirements sources understand the system and the requirements in different ways based on their needs, which means that requirements are likely to be stated in many different ways and on different levels of abstractions [15]. This introduces challenges when comparing or prioritising requirements. Without a direct channel between customer and development team, the marketing department gains importance, and this accentuates communication issues between marketing and development units [6, 23]. Also, in the absence of a direct channel between customer and development, it is up to the developing organisation to address challenges such as release planning [37], striking a balance between technology push and market pull and dealing with re-prioritised requirements. In addition, coordination and communication between problem (e.g., owned by product managers), and solution space (e.g., owned by development organisation), is complex, and requirements-based defects inherent to misunderstandings threaten functionality, quality, and time to market [7, 8].

As can be seen, the list of challenges for market-driven requirements engineering is quite different and on a higher level than the challenges in traditional requirements engineering. However, a successful organisation must be able to master both levels in order to achieve successful requirements engineering as most companies exist in a mix of bespoke and market-driven contexts. For example, a company developing products for markets often also have central key-customers that can act as bespoke customers, putting requirements to the development organisation that circumvents the normal MDRE elicitation. Thus, there is a need for process improvement that address both the traditional challenges and, where applicable, the market-driven challenges.

3.2 Requirements engineering process assessment

It is reported that the transfer of research results into industry practice is less than optimal [19–21] and sought after by industry practitioners [30]. One method to transfer research results into practice is through process improvement and process assessment frameworks. For entire software development organisations, frameworks such as CMMI [4], ISO9000 [38], and Spice [35] are available to assess process maturity. However, the fact that these frameworks cover the entire project lifecycle and, to some extent, beyond project boundaries also means that they do not go into detail into any particular practice area, such as requirements engineering. Moreover, they tend to be time- and resource consuming [17, 36].

There are also several requirements engineering process assessment frameworks, including the Requirements Engineering Good Practice Guide (REGPG) [34], Requirement Engineering Process Maturity Model v1.0 (REPM 1.0) [13], Market-Driven Requirements Engineering Process Maturity Model (MDREPM) [11], and others [2, 14, 18, 25, 31, 32, 38].

Common to the assessment frameworks listed above are that they are prescriptive; that is, they contain a set of actions that are expected to suit any average company or project. These are then structured according to different principles, for example, logical coherence, difficulty to implement, or required maturity to gain any value of the actions. All of them have been useful and used since their introduction. However, most of them are quite old which means that some practices prescribed by them are outdated, some have been superseded by newer practices, and some practices listed as advanced practices are now commonplace. In addition, few of them explicitly address the challenges faced by organisations in a market-driven development context, and rather, they were originally aimed at bespoke development efforts and rely on a heavy project focus. In market-driven development context the project is a consequence of the market-driven requirements engineering—consisting of large scale handling and triage of requirements, resulting in requirements selection (Several thousand requirements are narrowed down to hundreds that are selected for realisation in projects). Once this is done, the bespoke view can be used.

MDREPM, that was only recently published, does cover the market-driven view, but has the weakness of ignoring the bespoke view.

Moreover, no single framework of the mentioned covers the high-level needs of both market-driven requirements engineering and the lower-level needs of traditional requirements engineering. This implies that companies may need to execute two different process assessments to get a full picture of their current requirements engineering practices.

Thus, there is a need for a modern, light-weight process assessment tool for companies that operate in bespoke as well as in market-driven development contexts. Uni-REPM is the result of a research project[2] that brings in state-of-the-art research and practice in both contexts. Since Uni-REPM is new and as yet untried, it is important to validate it together with industry, in order to prove its applicability, usability, and usefulness. Ultimately, this is the goal of this article.

3.3 Uni-REPM

The aim of Uni-REPM is to serve as a universal light-weight model presenting the maturity of a requirements engineering process through sets of necessary activities. Besides the assessment purpose, Uni-REPM is also expected to function as a guideline giving organisations a recommended improvement path towards a better requirements engineering process from basic practices to an advanced level.

Uni-REPM is based on a set of design objectives, namely

- *Feasibility* The practices have to be validated in industry.
- *Universality* Practices in Uni-REPM shall be applicable in as many contexts as possible.
- *Light-Weight* Uni-REPM shall be a light-weight process assessment framework, not overladen with "good to have" practices or complicated assessment methods. The tools for assessing shall be easy to use, the structure of the model shall be simple and easy to navigate, and the contents of the model shall be well presented and self-contained to an as high degree as possible.

Uni-REPM is an evolution of previous requirements engineering process assessment models such as REPM v1.0 [13], REGPG [34], and MDREPM [11]. It is hierarchically structured into different process areas where each area potentially contains one or several subprocess areas. At the leaf nodes in this hierarchy, there are specific actions that a company can or should perform. Each action is placed on a specific maturity level, and if all actions on one maturity level are performed, the company has a consistent and coherent requirements engineering process of a specific maturity. Thus, the structure of Uni-REPM is relatively straightforward. The details of which process areas, subprocess areas, and above all the actions that are included in the model are, however, more complex. In a previous publication,[3] Uni-REPM was constructed as a result of a systematic literature review. In this article, the focus is instead on validating Uni-REPM as a whole.

4 Static validation

In this section the static validation performed in order to answer RQ1 is presented. The intention of this validation is to evaluate three particular aspects of Uni-REPM:

- *Completeness* Does Uni-REPM present all necessary requirements engineering practices with adequate information? Are there any missing necessary practices?
- *Correctness* Are the contents and presentation of Uni-REPM, especially names, maturity levels, placements

[2] Please see http://www.bth.se/tek/mdrepm.nsf.

[3] See footnote 1.

of actions on maturity levels, and descriptions of actions, correctly presented?
- *Applicability* Is Uni-REPM applicable for industry settings, and to what extent?

The intention of a static validation is to spread the candidate solution to a wide selection of domain experts or industry users and gather feedback in order to improve the candidate solution [9]. This involves presenting the candidate solution to a set of subjects and in some form collect feedback from them. Below, the design of the static validation, its execution, and improvements identified as a result of the static validation are presented.

4.1 Static validation design

Figure 2 presents the overall design of the static validation. As can be seen, the process starts by selecting subjects and schedule time with them. After this, a presentation of Uni-REPM and a list of questions is distributed to the subjects, so that they may prepare themselves before the interview meeting in step three. The main validation step in this part of the study is step three, where interviews are conducted with the subjects. The interviews are summarised and sent back to the subjects for verification to avoid misunderstandings. Finally, the feedback from the subjects is consolidated and analysed in order to improve Uni-REPM. The combination of structured and semi-structured interviews was chosen in order to get rich feedback with the ability to ask follow-up questions. Moreover, since the material to go through is relatively large, by booking time for the interviews early it was possible to get commitment from several, otherwise busy, domain experts that may not have answered at all on, for example, a questionnaire.

Not shown in Fig. 2 is that all interview instruments were carefully reviewed and piloted internally before distribution to the study subjects. This was done to remove any unclarities or ambiguities so that the subjects would be able to easily answer the questions correctly.

4.1.1 Subject selection

In this study, academic domain experts are used since the intentions of the static validation are better addressed by subjects with a broader domain knowledge and also some experience in education. Thus, in order to acquire accurate, valuable, and useful feedback, the following subject selection criteria were taken into account:

- Subjects should have a research interest in Requirements Engineering or Product management. This is to ensure the commitment of the subjects through the whole validation.
- Subjects should have contributed relevant important publications in the study area. This is to ascertain that the subjects have appropriate knowledge to evaluate the model.
- Subjects should have close collaboration with industry. Since the model targets to be applied in industrial organisations, it is necessary that the subjects have industrial experience to be able to evaluate the applicability of the model.

On the basis of these criteria, several sources (e.g., publications and personal recommendations) were searched and identified experts were contacted through e-mail. After one month, seven out of 17 experts contacted accepted to participate in the validation. The list of experts is shown in Table 1.

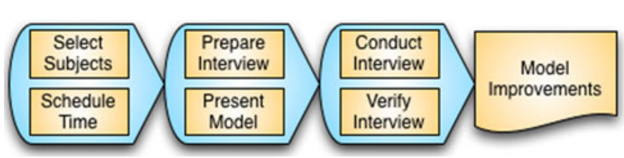


Fig. 2 Static validation process

4.2 Static validation execution

The study was conducted through the use of semi-structured interviews, in order to be able to adjust the questions asked based on each subjects' area of expertise. A minimal set of questions were prepared and distributed beforehand in order to ensure that certain themes and issues were covered. Five of the interviews were conducted by audio conference and the remaining two through face-to-face meetings. In all but one case, only one interview was conducted. Dr. Fricker was gracious enough to participate in a follow-up interview one week after the main interview.

The interviews were designed to last for two hours and were conducted by two of the researchers; one asked questions while the other took notes and asked clarifying questions. The interviews were recorded with the interviewees

Table 1 Static validation participants

Name	Title	Country
Kristian Sandahl	Professor	Sweden
Jürgen Börstler	Professor	Sweden
Samuel Fricker	Ph.D.	Switzerland[a]
Inge van de Weerd	Ph.D.	The Netherlands
Christof Ebert	Ph.D.	Germany
Richard Berntsson Svensson	Ph.D. Student	Sweden
Krzysztof Wnuk	Ph.D. Student	Sweden

[a] At the time of our study Dr. Fricker was still in Switzerland. To the authors' great joy, Dr. Fricker has since moved to Sweden

consent, and notes were taken during each interview in the event that recording equipment failed [5]. Moreover, taking notes can also provide a real-time "sanity check" to discover aspects that need to be discussed further [31].

Before conducting the first interview, the interview questions were carefully reviewed and then piloted with the help of a PhD student not involved in the research project. During the pilot interview, different ways to ask some of the questions were tested in order to remove any ambiguity in the question or the answers. As a result of the pilot, some of the questions changed form, and some valuable input for how to modify Uni-REPM was also provided.

After the interview, the interview content was transcribed from the recording in a summary form (i.e., not a verbatim transcription). The notes and transcription were then compared to double-check the consistency as well as to avoid losing information. The summary of the interview result was sent back to the interviewee for review before conducting any further analysis based on the results.

Details on the duration of each interview are presented in Table 2. In this table, the time each subject spent on reviewing Uni-REPM prior to the interview is also presented. The subjects are presented in the order interviewed.

4.3 Static validation results

The individual opinions of each interviewed subject are consolidated into a number of suggestions per aspect, as summarised in Table 3 and Table 4. After grouping similar suggestions from different subjects, a total of 65 suggestions remain.

The 65 suggestions were analysed according to a predefined process as illustrated in Fig. 3. First, suggestions were analysed to determine their relevance to the model. Correctness suggestions were checked against literature. Completeness suggestions were weighed against the scope of Uni-REPM and the design goal to be light-weight. The remainder of the suggestions were assessed to ensure that they would be beneficial to the model (according to the researchers) and whether omission would detriment the model. In the end, 47 of the 65 suggestions were implemented. In two cases the implementation was postponed due to time and effort, and since it would not negatively affect the model if the suggestions are not implemented. In Table 5 we present the number of implemented and not implemented suggestions per aspect.

The 47 implemented suggestions are presented on the project homepage http://www.bth.se/tek/mdrepm.nsf (along with reasons for not implementing the remaining 17 suggestions). Significant changes include:

- The notion of optional actions was removed; the only occurrence was anyway more complementing than competing actions.
- "Satisfied/Explained" was changed to "Inapplicable" as this is more easily understood (even if this may result in it being overused).
- More neutral names for the maturity levels were adopted: "basic", "intermediate", and "advanced", respectively.
- The name of the MPA Quality Assurance was changed to "Requirements Validation".
- The order of some actions was changed, and some actions were moved to a different MPA (e.g., the actions concerning prototyping and system modelling were moved to the MPA Requirements Analysis).
- Many actions were clarified in order to make them more understandable.

Table 2 Interview and model review durations

Id	Subject	Interview duration	Model review duration
E1	Dr. v.d. Weerd	55 min	1 h
E2	Dr. Fricker	1 h 35 min + 15 min	45 min
E3	Mr. Wnuk	1 h	1 h
E4	Prof. Börstler	1 h 20 min	2 h
E5	Mr. Berntsson-Svensson	1 h 30 min	3 h
E6	Dr. Ebert	15 min	Unknown
E7	Prof. Sandahl	1 h 25 min	1 h

Table 3 Suggestion summary

Aspect	Suggestions
Correctness	23
Completeness	19
Applicability	2
Others	21

Table 4 Suggestions summary per category

Category	#	Suggestions
Model Structure	7	SG1–SG7
Maturity level Structure	1	SG8
Model Contents Details	3	SG9–SG11
MPA: Organisational Support	14	SG12–SG25
MPA: Requirements Process Management	9	SG26–SG34
MPA: Requirements Elicitation	8	SG35–SG42
MPA: Requirements Analysis	11	SG43–SG53
MPA: Release Planning	3	SG54–SG56
MPA: Quality Assurance	4	SG57–SG60
Others	5	SG61–SG65

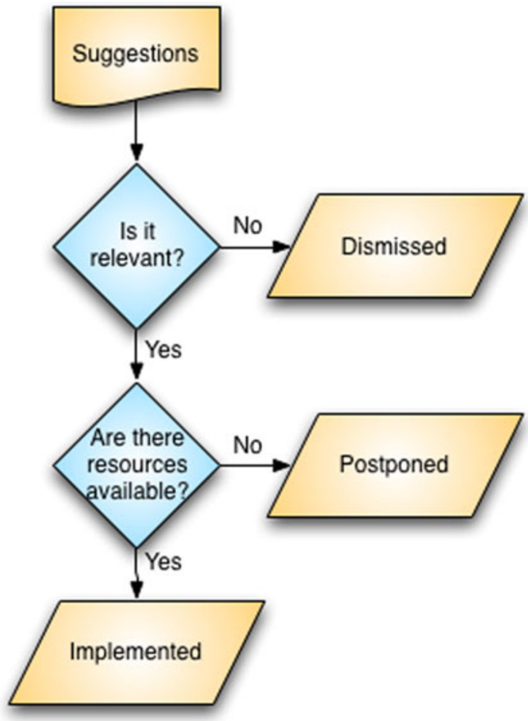


Fig. 3 Suggestion analysis process

4.4 Static validation summary

The aim of the static validation is to ensure correctness, completeness, and applicability of Uni-REPM. By eliciting comments from domain experts and improve the model accordingly, the model is enhanced with respect to all three of these aspects. Under the assumption that if the suggestions provided by domain experts are addressed, the correctness, completeness, and applicability of the model are ensured, the aim of the static validation is thus met.

According to the interviewed domain experts' opinions, all of the actions in the model are applicable in real settings (with few additional actions being suggested). However, the model has to be validated in industry in order to confirm its applicability. This is the focus of the next section.

5 Dynamic validation

In this section the dynamic validation performed in order to answer RQ2 is presented. The intention of this validation is to evaluate three particular aspects of Uni-REPM:

- *Applicability* To what degree can Uni-REPM be applied in industrial projects with different development environments (bespoke, market-driven, or a combination of both)?
- *Usability* The usability of the model is evaluated through three subaspects: *efficiency*, that is, the time it takes for a practitioner to use Uni-REPM to assess the requirements engineering process maturity; *understandability*, that is, the ease by which a practitioner understands and answers the evaluation tools correctly; and *satisfaction*, that is, how pleasant a practitioner feels about Uni-REPM, including the model, the evaluation tools, and the whole validation session.
- *Usefulness* To what degree does Uni-REPM provide a clear view of what needs to be improved in order to enhance a company's requirements engineering practices?

With dynamic validation the candidate solution is applied in an industrial context in order to evaluate the candidate solution under realistic conditions and to identify where further improvements are needed before conducting full scale tests [9]. Below, the design of the dynamic validation, its execution, including evaluation results for the evaluated projects, and a detailed analysis thereof, is presented.

5.1 Dynamic validation design

Figure 4 presents the overall design of the dynamic validation. As can be seen, the first step is to select subjects (participants in industry projects) and distribute a brochure about Uni-REPM to them. After this, the subjects may take two different paths: the upper path where the Uni-REPM assessment is conducted as an interview. For the purposes of the dynamic validation, this is the preferred path. If the subjects are unavailable, the lower path is offered where the subjects complete a self-administered questionnaire using a precis of Uni-REPM, after which a mini-interview

Table 5 Suggestion responses

Response	Suggestion type				
	Correctness	Completeness	Applicability	Others	Total
Implemented	20	12	0	15	47
Postponed	0	0	0	2	2
Dismissed	3	7	2	4	16
Total	23	19	2	21	65

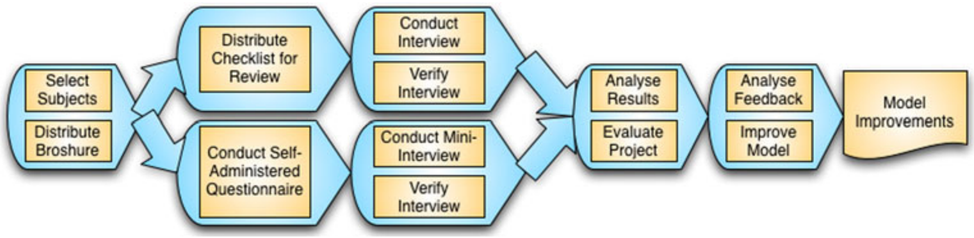


Fig. 4 Dynamic validation process

over phone may be organised for clarification purposes. In both the upper and the lower paths, the results so far are summarised and sent back to the subjects for verification, in order to avoid misunderstandings. After this, the results are analysed and the projects are evaluated. This, together with analysis of additional feedback about Uni-REPM from the practitioners, then serves as a basis for further model improvements. As with the static validation, all instruments were carefully reviewed and piloted internally beforehand.

Conducting the assessment in the form of interviews is preferable since this gives richer material; it is possible to ask follow-up questions and also to identify when a subject hesitates or emphasises in a particular way. This is important for the assessment of Uni-REPM, but equally important for the assessment of a project with the help of Uni-REPM. The self-administered questionnaire is offered as an alternative for two reasons. First, not all practitioners may be available for the full-scale interview, and second, it is likely that the self-administered questionnaire is going to be the preferred assessment form when Uni-REPM is being used in industry. There is, however, a validity threat with the self-administered questionnaire, since subjects may be more inclined to answer in a way that is favourable to their own project. To address this threat during the evaluation of Uni-REPM, mini-interviews are conducted for clarification purposes when subjects use the self-administered questionnaire.

5.1.1 Subject selection

For this validation, the focus is on companies that have a requirements engineering process and on practitioners involved in the companies' requirements engineering process. The subjects thus acquired were asked to select a, for the company, typical project to evaluate. Thus, the subjects are able to select a project where they can discuss freely and openly about the project without breaking any confidentiality agreements.

A large number of Swedish IT companies were contacted at random, but none of them were interested in participating within our time frame, and thus we had to resort to personal contacts in order to arrange for interviews. As a result, four companies agreed to participate (two from Singapore, one from Spain, and one from Denmark), as listed in Table 6 and presented in further detail below.

Within P1, 15 persons worked in the studied business unit at the time of study. The project under assessment involved customising two different modules in an MDRE-developed product for a client, one for internal invoicing and the other for space management, both related to facility management. At the time of study, the project had lasted four months and was still being developed. In the team, two persons, that is, the project manager and the project manager's assistant, were responsible for requirements engineering.

P2 involved programming the interface for robot hardware and consisted of five team members. At the time of study, the project had lasted seven months. The project was a bespoke project with a specific customer.

In P3, a system for a small insurance company in Singapore was developed. This project was developed in a company with around 500 employees. The system was derived from a generic product developed by the company, and new requirements were created by comparing the gap between the customer's expectations and the existing functionality of the system. The new requirements were implemented by customisations and localisation teams. This project lasted one and a half year.

Table 6 Dynamic validation participants

Project	Domain	Means of validation	Size of RE team	RE context
P1	Facility Management Software	Skype Interview	2 persons	Market-driven product, customised for one client
P2	Embedded Software	Skype Interview	5 persons	Bespoke product
P3	Insurance & Banking	Skype Interview	6 persons	Bespoke product derived from a Market-driven product
P4	Insurance	Self-administered questionnaire	10 persons	Market-driven product

P4 was developed by a company in the insurance domain with 800 employees and 200 employees in the IT business unit. The project lasted for one year, involving 10 people in the requirements engineering team. The project developed a point of sale system containing information about insurance products and managing client information along with their purchases. The system also generates benefit illustrations for the clients and risk profiles of the clients. This is an in-house system with the requirements coming from internal users and is being used across Singapore, Indonesia, Vietnam, Malaysia, China, and Brunei.

5.2 Dynamic validation execution

The preferred means of conducting the dynamic validation is through face-to-face interviews, as this enables the researchers to notice cues to when the participants do not understand or do not agree. However, to fit within time and budget constraints, audio conferences were used instead.

The interviews were based on the checklist provided with Uni-REPM and hence, the interviews were conducted as structured interviews. This checklist is constructed such that for each action in Uni-REPM there is a corresponding question in the checklist that can be answered with either "yes", "no", or "inapplicable". This checklist (along with Uni-REPM) was distributed to the subjects before the interview. In addition to the questions in the checklist, questions about the company, the project, and the interviewee's background were also asked in order to obtain an understanding of the projects' contexts. Moreover, a set of closing questions were added to the interview instrument.

The interviews were scheduled to last for 1.5 hours and were conducted by two researchers who worked together such that one asked questions and the other took notes and asked clarifying questions.

During the interview, the interviewees were asked to "think aloud" when answering the questions. This serves as one of our primary means to assess the understandability of Uni-REPM. If Uni-REPM is instead used to evaluate a project (as opposed to being the object of evaluation), thinking aloud may provide rationales when deeming an action "inapplicable". Both researchers were vigilant for "misunderstanding signs" from the interviewees. Examples of such signs include hesitation when answering questions or irrelevant answers. These signs are another primary means to assess the understandability of Uni-REPM. The explanation of the action from the model was provided to clarify and resolve the misunderstandings.

After the interview, the interview content was transcribed from the recording in a summary form (i.e., not a verbatim transcription). The notes and transcription were then compared to double-check the consistency and to avoid losing information. The summary of the interview result was sent back to the interviewee for review before conducting any further analysis based on the results.

5.2.1 Self-administered questionnaire

In one case, a self-administered questionnaire was used instead of an interview. In this case, the questionnaire used was the same as the one used in the interviews, with the addition of questions to capture the subjects' understanding of the model. In addition, a brief version of the model consisting of the process area view and the actions was distributed together with the questionnaire.

5.3 Dynamic validation results

The results can be analysed in two ways: an analysis of the results with respect to the assessed projects' requirements engineering process maturity, as intended by the Uni-REPM model, and the other way is to analyse the results with respect to the goals of the dynamic validation. Focusing on the validation of Uni-REPM, the following aspects and information sources are considered:

- *Usability: Efficiency* The time used for each validation session.
- *Usability: Understandability* All the information about misunderstandings and ambiguity.
- *Usability: Satisfaction* Feedback from the practitioners about the checklist, model, and the validation session.
- *Usefulness* A general discussion with the practitioners after the process assessment.
- *Applicability* Model lag (the number of actions deemed "inapplicable"), along with feedback from the practitioners.

Below, each of these aspects is discussed in further detail.

5.3.1 Usability: efficiency

The interviews were completed in the designated time frame of 1.5 hours while it took 40 minutes to fill in the self-administered questionnaire. In addition, the assessment exercise did not require any costly resources except for the practitioner. The detailed result analysis generally took 10 hours of work, but much of this is analysis for the sake of the dynamic validation. Analysis for Uni-REPM can to a large extent be automated up to the point where specific process improvement actions need to be decided on and planned in the company.

5.3.2 Usability: understandability

Problems in understanding can stem from the question in the checklist, the explanation of the corresponding action in the model, the structure of the model itself, or because the concept is new to the interviewee.

After analysis, 17 questions were rephrased, three explanations were added to the model, two actions were removed, and five additional misunderstandings were resolved with the help of the existing explanations in Uni-REPM[4].

5.3.3 Usability: satisfaction

The interviewee from project 1 commented that he liked the question lists. "I thinks your questions are really nice because some questions make me remember some answer from other questions, so you create synergies between the questions [hence] If the interviewed person forgets about something he/she could remember it with some other questions". However, he also mentioned that some of the questions were long and he only focused on the first part of the question. In accordance, some of the questions have been rephrased and shortened.

It was also recommended to establish the relationships between questions in the list in order to avoid asking inapplicable questions (for example, if there is no product roadmap in the process, it is not necessary to answer question RP.GA.a1) and to develop a classification or description of environments in which the model is suitable.

In project 3, the practitioner mentioned that the checklist was quite long and involved many more actions than the real process. This is an unavoidable deficiency of most prescriptive process assessment tools, however lightweight they are.

In project 4, the interviewee performed a self-administered questionnaire. She read through the whole model description by herself and used it to assess her project by answering all the questions in the checklist. The interviewee said that all the terms used in the checklist are familiar, and the explanation in the model was adequate and easy to understand. Apart from the process assessment, a mini-interview was arranged to confirm and evaluate her result. This interview confirmed the self-administered assessment's results. Moreover, it was confirmed that the interviewee had the same understanding of the "difficult" concepts as intended. Hence, the result from this case provides convincing evidence of the model's usability. It shows that it is possible for an engineer to learn and apply the model in real life without additional help.

[4] Details about these improvements are available on the project homepage: http://www.bth.se/tek/mdrepm.nsf

5.3.4 Usefulness

Besides the purpose of evaluating the usability and applicability of the model, the dynamic validation also aims to validate the usefulness of the model in assessing the process maturity level of a project. Hence, the raw results obtained from the interviews and questionnaires were analysed, and the overall process maturity of the projects according to Uni-REPM was determined. Moreover, each main process area was scrutinised to locate the strong and weak points of the processes. On the basis of those findings, specific improvement actions were recommended in order to increase the maturity of the process.

The interviewee in project 2 expressed that the checklist and the corresponding actions were useful because they gave him ideas on how to improve the process in the next project. This is also one of the contributions of this study, which is to narrow the gap between academic state-of-the-art and industrial state-of-practice. While there has been a lot of evolution of requirements engineering practices in academia, widespread adoption of the same in industry is lacking.

In project 4, the interviewee commented that the idea of the model was very nice, but it would be more beneficial for practitioners to get the information of *how* to perform the actions recommended in the model. Currently, Uni-REPM mostly provides guidance on "what" to do as opposed to "how". However, the missing information for how to perform actions is considered as more important and preferable from the companies' point of view. Hence, implementing these suggestions will improve the usefulness of the model. However, due to time limitations, this was left as an improvement for the next version of the model.

5.3.5 Applicability

In order to assess the applicability of Uni-REPM, the model lag (i.e., the number of actions deemed "inapplicable") is studied, and the actions most commonly deemed as "inapplicable". Model lag was introduced as a term already in REPM v1.0 [13] and is a useful side effect of introducing the "inapplicable" answer category. If an action does not suit a particular company, this is simply viewed as a deficiency of the model itself—a lag between the prescribed and the needed practices.

A summary of the model lag is presented in Table 7. In this table the model lag is presented in per cent along with the number of actions for each project. P2 and P3 were developed in a bespoke context, and hence, release planning (RP) activities are by default not applicable. Therefore, Table 7 also presents the model lag without the release planning actions in these two cases.

Table 7 Model lag summary

Project	Model lag	
	Without RP	Total
P1	n/a	12% (9 of 74 actions)
P2	13% (9 of 67 actions)	22% (16 of 74 actions)
P3	19% (13 of 67 actions)	27% (20 of 74 actions)
P4	n/a	0.002% (1 of 74 actions)

Studying the interviews, a common reason for assessing an action as "inapplicable" was because of the nature of the projects. Since the products being built were mostly derived from existing products, part of the information (and requirements) could be reused, and hence this invalidated the need for certain actions. However, there are also some cases where the subjects claimed that the actions were not suitable for their situations. One example of this is the action "DS.GA.a1 Define Requirements Attributes". In one of the projects, it was argued that since the project was small and the process kept simple, they did not make use of attributes to manage requirements.

Three actions were deemed inapplicable in more than one project:

- *OS.RR.a5* Define Roles and Responsibilities for Product Management
- *RE.SI.a3* Identify Other Requirements Sources
- *RE.GA.a7* Reuse Requirements

The interviewees found that it would be too complicated and not necessary to involve the product manager in their requirements process, since the requirements were often created by the development teams. Moreover, in most of the interviewed cases, the customers are the main and most important source of requirements, and for smaller companies with a close customer connection, other requirement sources such as bug reports were considered unnecessary. This may, however, be more relevant as the project sizes grow. The interviewees thought similarly about the "Reuse Requirements" action; it requires a mature organisation of a certain size to be able to systematically plan a reuse effort and to gain any benefits therefrom.

5.4 Dynamic validation summary

Uni-REPM is thus applied on four different projects, in four different companies in three different countries. Uni-REPM is shown to be applicable in various development environments, including bespoke, market-driven, and a mix of the two, and different development domains, and different project sizes.

In all interviews, the practitioners show a good understanding of the checklist and the actions in the model. They are familiar with most of the terms used as well as the actions. Moreover, Uni-REPM is efficient in assessing the requirements engineering process maturity in organisations given the short duration of the interviews or the self-administered questionnaire. In addition, Uni-REPM indicates opportunities for improvement by clearly identifying actions that are not performed and that would further enhance a company's requirements engineering processes.

In this article the focus is on the assessment of Uni-REPM itself and not the actual assessment of the projects. However, for illustration purposes, summaries of the projects are presented in Fig. 5, and in Table 8, along with a brief discussion of them below. Each graph in Fig. 5 is to be interpreted as follows; for each maturity level, the total number of actions is listed per MPA (dark blue), along with the number of completed actions (blue), and the number of inapplicable actions (light blue).

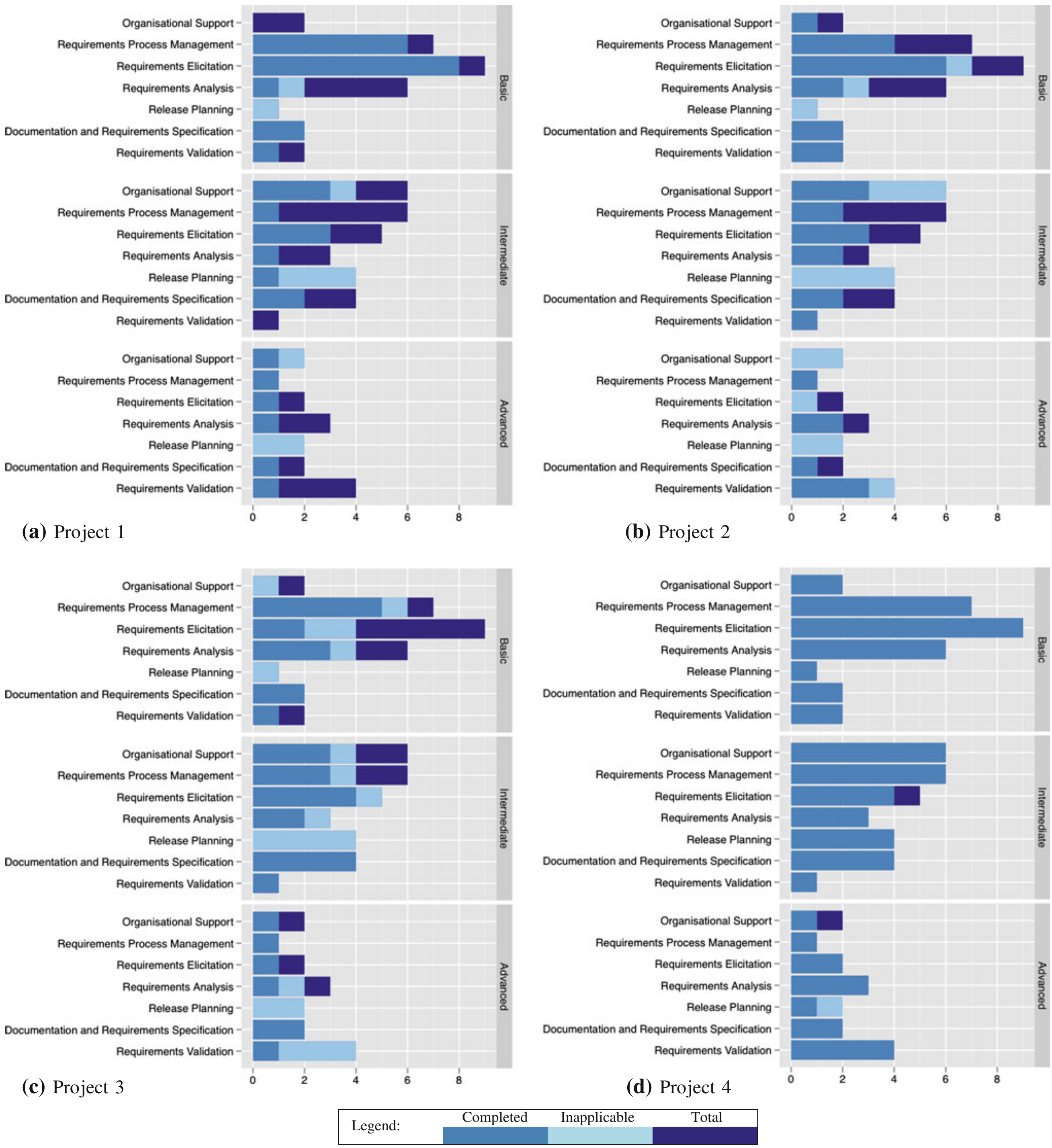
A quick overview shows that project 4 has a very mature requirements engineering process. Project 4 is followed in order of maturity by projects 3, 2, and 1. With the exception of project 4, there are actions on the basic level that need to be addressed before the projects can be assessed to be on this level. Being on a specific level has no value in itself, except that the actions on that level together form a consistent and coherent requirements engineering practice. Thus, actions needing to be addressed on the basic level are listed in Table 9 (for reasons of confidentiality we list them all together so that no individual project can be identified. Some actions were not completed in more than one project). For each of these actions and for each of the projects, a more in-depth analysis is made together with the company to identify the consequences of the current situation, and the consequences of implementing the action in order to determine whether there is an actual need to implement the action. Once the basic level is accomplished, one may then continue with the actions on the intermediate and advanced levels.

6 Validity threats

In this section, the validity threats to both the static and dynamic validation are discussed according to Wohlin et al. [39].

6.1 Internal validity

In order to ensure that the interview instruments were properly designed and able to answer the questions we want answered, all interview instruments were carefully reviewed and piloted before conducting the interviews.



(a) Project 1 (b) Project 2 (c) Project 3 (d) Project 4

Fig. 5 Assessment summaries

In order to obtain relevant feedback from the interview subjects in the static validation, a set of specific criteria were used as described in Sect. 4.1 and only invited those that matched these criteria. Of the 17 persons thus identified, seven (41%) agreed to participate. All participating subjects were in general positive to Uni-REPM and gave valuable feedback on improvement opportunities. While different or additional feedback may have been received from those that did not participate, the most pressing issues are likely to have been found by the participating domain experts.

There is a threat that the projects selected by the subjects in the dynamic validation are not representative of the whole organisation. In order to encourage the subjects to

Table 8 Summary over all process areas for each project

Project	Level	Actions			
		Completed	Inapplicable	Incomplete	Total
Project 1	Basic	18	2	9	29
	Intermediate	11	4	14	29
	Advanced	6	3	7	16
Project 2	Basic	17	3	9	29
	Intermediate	13	7	9	29
	Advanced	7	6	3	16
Project 3	Basic	18	6	5	29
	Intermediate	17	8	4	29
	Advanced	7	6	3	16
Project 4	Basic	29	0	0	29
	Intermediate	28	0	1	29
	Advanced	14	1	1	16

Table 9 Incomplete actions on the basic level, union of all four projects

ID	Title
OS.GA.a1	Create a Product-wide Glossary of Terms
OS.RR.a1	Assign Owner(s) of Requirements Development and Management Processes
PM.GA.a1	Define and Maintain Requirements Development and Management Processes
PM.CM.a1	Manage Versions of Requirements
PM.CM.a2	Baseline Requirements
PM.RT.a2	Document Requirements' Source
RE.GA.a1	Elicit Quality Requirements
RE.DC.a3	Elicit Information about System's Business Process
RE.DC.a4	Elicit Information about System's Operational Domain
RA.QA.a1	Analyse for Missing and Double Requirements
RA.QA.a2	Analyse for Ambiguous Requirements
RA.QA.a3	Analyse for Correctness of Requirements
RA.QA.a4	Analyse for Testability of Requirements
RA.PS.a1	Create Prototype
RV.GA.a2	Use Checklist to Ensure Quality of Requirements

actually select representative projects, the assessment result is kept anonymous, and it was emphasised that the main purpose is to validate Uni-REPM and not the company.

The researchers had to resort to convenience sampling in the dynamic validation in order to get any industry responses at all. This is of course a threat, since personal contacts can be expected to be more positive towards the proposed model than others may be. On the other hand, they may also feel more free to "speak their mind" since they are already acquainted. Moreover, convenience sampling resulted in cases of requirements engineering processes from three different countries being studied, as opposed to the original strategy of only sampling swedish companies.

A related threat is that only one subject from each company was interviewed. If more persons from the same projects had been interviewed, a richer picture may have been received. In fact, this can also be used to assess the requirements engineering process in a company. If, by a couple of 1.5-hour sessions with Uni-REPM, it is detected that project participants do not share the same view of the requirements engineering process, this may be an indication of problems in the project.

6.2 Conclusion validity

As stated above, all interview instruments were reviewed and piloted before the actual interviews. A number of measures were taken to ensure a fair treatment of the interview subjects and their answers. Specifically, two researchers participated in each interview, all interviews were recorded, two researchers participated in the analysis of each interview, and each interview was summarised and then sent back to the interviewee for confirmation. Finally, all types of feedback was sought, and the researchers were actively searching for improvement opportunities on Uni-REPM. Therefore, it is argued that the risk that the researchers' hopes and desires have negatively influenced the results is negligible.

6.3 Construct validity

All types of feedback were actively elicited, especially improvement opportunities, which means that most construct validity threats are moot or may even turn out beneficial for the aims of the study, since they often imply that external sources influence the results of the study or that the study setting itself may induce the subjects to act differently.

During the dynamic validation, there is a threat of evaluation apprehension. To address this threat it was emphasised that the primary goal of the interview was to evaluate Uni-REPM and not the company. The result of an Uni-REPM evaluation is also primarily intended as a tool to identify improvement opportunities, and this was also communicated to the participants. Moreover, the participants were informed that the result analysis generated will be kept anonymous.

6.4 External validity

External validity relates to the ability to generalise the result to a larger population [39]. The interaction of selection and treatment can pose a threat in this study as the participating domain experts are academic researchers and not industrial practitioners. However, all of the experts involved have relevant industry experience which may mitigate the threat.

There is a small threat caused by the interaction of selection and treatment as the dynamic validation was only performed in four organisations. The threat is reduced by reporting the characteristic of the environments and providing details about the projects under evaluation. Moreover, the companies represent different development environments, different development domains, and different countries, which further implies that Uni-REPM is applicable internationally and in different contexts.

7 Uni-REPM revisited

Having thus validated and improved Uni-REPM, the new version is briefly presented in this section. A more detailed version, with full descriptions of all the actions, the Uni-REPM model itself, as well as checklists for conducting an Uni-REPM assessment can be found on the project homepage: http://www.bth.se/tek/mdrepm.nsf.

7.1 Structure

Uni-REPM is structured in two views, a Process Area view and a Maturity Level view. These serve two different purposes. The process area view is used to navigate the model and to quickly find practices that logically belong together, whereas the maturity level view describes sets of practices that constitute a consistent and coherent RE process, and where the practices in one level supports each other as well as the more advanced practices on the next level. The process area view is hierarchically constructed of main process areas (MPA), consisting of subprocess areas (SPA), and Actions. This is illustrated in Fig. 6.

Within the description of each action, there can be recommendations and supporting actions. Recommendations give practitioner's suggestions on proven techniques or supporting tools for the practice. This information aims to help practitioners when implementing an action. Supporting actions provide links to other actions which will benefit the practitioners if they are implemented together. Each action is assigned a certain level (from 1 to 3, corresponding to "Basic", "Intermediate", and "Advanced" level) depending on its difficulty to implement, how essential it is for the RE process, and dependencies between actions. This constitutes the Maturity Level View. We would like to point out that the maturity level is only applicable to the RE process and does not indicate anything concerning the overall maturity of the organisation as a whole. It should, however, be possible to compare two RE processes in terms of maturity using the results of an evaluation.

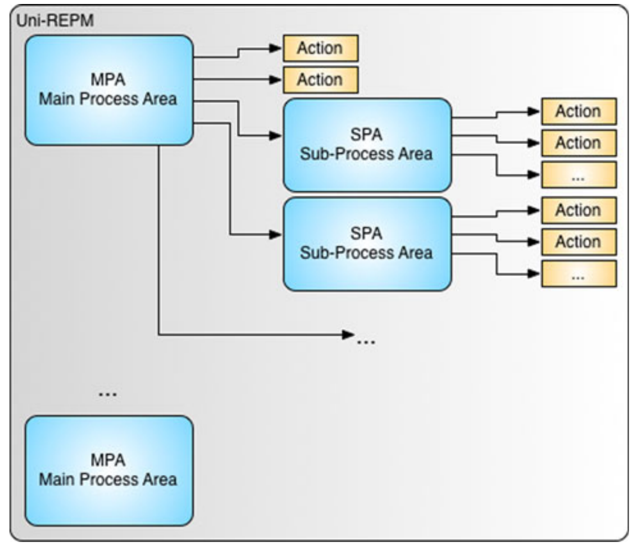


Fig. 6 Structure of Uni-REPM

7.2 Usage

Uni-REPM assessments are made with the help of a checklist. For each action, a question is posed, that can be answered either by "completed", "incomplete", or "inapplicable". The "inapplicable" category ensures that the results of an evaluation can still be interpreted even when there are actions that are not applicable to a particular situation or organisation.

The results can be interpreted for a particular MPA or for the entire RE process. In order to assess the maturity level of an organisation, all actions within an MPA (or overall) at a certain level must be completed or satisfied explained in order to achieve this level. A more important interpretation is to study the actions that are marked as incomplete on the level above the current maturity level, as these indicate which activities should be considered next for process improvement efforts.

Studying, for example, Project 1 and the MPA Organisational Support in Fig. 5 and the corresponding Table 8, it can be seen that no actions are completed on the Basic level, three actions are completed on the intermediate level, plus one more inapplicable, and one action completed and one inapplicable on the advanced level. What this means is that this company should first focus on implementing the activities on the "basic" level and the missing activity on the "intermediate" level. A similar analysis can then be applied to each of the other MPA's.

7.3 Contents

An overview of the Uni-REPM model after validation is presented in Table 10. This is a brief summary of the

Table 10 Uni-REPM summary

ID	Title	Level
OS	Organisational Support	
OS.GA	General Actions	
OS.GA.a1	Create a Product-wide Glossary of Terms	1
OS.GA.a2	Train personnel in Requirements Development and Management Processes	2
OS.RR	Roles and Responsibilities	
OS.RR.a1	Assign Owner(s) of Requirements Development and Management Processes	1
OS.RR.a2	Define Roles and Responsibilities for Requirements Development and Management Processes	2
OS.RR.a3	Define Roles and Responsibilities for Release Planning	2
OS.RR.a4	Define Roles and Responsibilities for Change Control	2
OS.RR.a5	Define Roles and Responsibilities for Product Management	3
OS.S	Strategies	
OS.S.a1	Define Product Strategies	2
OS.S.a2	Define Product Roadmaps	2
OS.S.a3	Communicate Strategies in Organisation	3
PM	Requirements Process Management	
PM.GA	General Actions	
PM.GA.a1	Define and Maintain Requirements Development and Management Processes	1
PM.GA.a2	Introduce Tool Support for Requirements Development and Management	1
PM.GA.a3	Involve various perspectives in Requirement Development and Management Process	2
PM.CM	Configuration Management	
PM.CM.a1	Manage Versions of Requirements	1
PM.CM.a2	Baseline Requirements	1
PM.CM.a3	Define a Process for Managing Change and Evolution	2
PM.CM.a4	Track change requests	2
PM.RT	Requirements Traceability Policies	
PM.RT.a1	Uniquely Identify each Requirement	1
PM.RT.a2	Document Requirements' Source	1
PM.RT.a3	Define traceability policies	2
PM.RT.a4	Document Requirements' Relation	2
PM.RT.a5	Document Impact of Requirement on Other Artefacts	2
PM.RC	Requirements Communication and Negotiation	
PM.RC.a1	Establish Effective Communication With Requirements Issuers	1
PM.RC.a2	Obtain common understanding of requirements among different involving roles	3
RE	Requirements Elicitation	
RE.GA	General Actions	
RE.GA.a1	Elicit Quality Requirements	1
RE.GA.a2	Qualify and Quantify Quality Requirements	2
RE.GA.a3	Let Business Concern Guide Focus of Elicitation	2
RE.GA.a4	Use Appropriate Elicitation Techniques according to Situation	2
RE.GA.a5	Use Artefacts to Facilitate Elicitation	2
RE.GA.a6	Create Elicitation Channels for Requirements Sources	3
RE.GA.a7	Reuse Requirements	3
RE.SI	Stakeholder and Requirements Source Identification	
RE.SI.a1	Identify and Involve Relevant Stakeholders	1
RE.SI.a2	Identify Other Requirements Sources	1
RE.DC	Domain Consideration and Knowledge	
RE.DC.a1	Elicit Information about System Domain Restrictions	1
RE.DC.a2	Elicit Information about System's Technical Infrastructure	1
RE.DC.a3	Elicit Information about System's Business Process	1
RE.DC.a4	Elicit Information about System's Operational Domain	1

Table 10 continued

ID	Title	Level
RE.DC.a5	Elicit Information about System Boundaries	1
RE.DC.a6	Consider Sociopolitical Influences on Requirements Sources	2
RA	Requirements Analysis	
RA.GA	General Actions	
RA.GA.a1	Perform Requirements Risk Analysis	1
RA.GA.a2	Perform Systematic Requirements Prioritisation at Project-level	2
RA.GA.a3	Analyse Requirements Relations	2
RA.GA.a4	Identify Irrelevant Requirements for Early Dismissal (in/out scope OR Triage)	2
RA.GA.a5	Analyse the Strength of Relations between Requirements	2
RA.GA.a6	Perform refinement and abstraction of requirements	3
RA.QA	Quality attributes analysis	
RA.QA.a1	Analyse for Missing and Double Requirements	1
RA.QA.a2	Analyse for Ambiguous Requirements	1
RA.QA.a3	Analyse for Correctness of Requirements	1
RA.QA.a4	Analyse for Testability of Requirements	1
RA.PS	Problems and solutions analysis	
RA.PS.a1	Create Prototype	1
RA.PS.a2	Perform Systems Modelling	3
RP	Release Planning	
RP.GA	General Actions	
RP.GA.a1	Synchronise Release Plan with Product Roadmap	2
RP.GA.a2	Involve different perspectives in release planning	2
RP.GA.a3	Post Requirement Selection Evaluation	3
RP.GA.a4	Plan multiple release at pre-defined interval	3
RP.S	Requirements Selection	
RP.S.a1	Pack Requirements into Releases	1
RP.S.a2	Estimate Cost and Value of Requirements	2
RP.S.a3	Perform Requirements Prioritisation at Pre-project Level based on Various Dimensions	2
DS	Documentation and Requirements Specification	
DS.GA	General Actions	
DS.GA.a1	Establish Standardised Structure for SRS	1
DS.GA.a2	Define Requirements Attributes	1
DS.GA.a3	Define Requirements States	2
DS.GA.a4	Document Requirements Rationale	2
DS.GA.a5	Record Rationale for Rejected Requirements	3
DS.DD	Documentation Deliverables	
DS.DD.a1	Define User Documentation Deliverables	2
DS.DD.a2	Define System Documentation Deliverables	2
DS.DD.a3	Define Management Documentation Deliverables	3
RV	Requirements Validation	
RV.GA	General Actions	
RV.GA.a1	Validate requirements with relevant stakeholders	1
RV.GA.a2	Use Checklist to Ensure Quality of Requirements	1
RV.GA.a3	Review Requirements	2
RV.GA.a4	Organize Inspections	3
RV.GA.a5	Develop Preliminary Test Case or User Manual	3
RV.GA.a6	Use System Model Paraphrasing for QA	3
RV.GA.a7	Define Acceptance Criteria and Acceptance Tests	3

process areas, the actions and their corresponding maturity levels. For a more detailed description of Uni-REPM please see the project homepage.

8 Conclusions

To validate is a virtue. Too often research results are released without a proper validation. Often when a validation is claimed, it is on small examples or on a small set of subjects. Potentially, this leads to a situation where research results are not trusted or adopted because questions remain about their usefulness. In a previous publication a model for requirements engineering process assessment, Uni-REPM, was introduced. Although this was meticulously constructed of only industry validated requirements engineering practices, the model as a whole was not validated in that publication. This is instead the focus of this article, that is, a thorough validation of Uni-REPM, consisting of a static validation with the help of domain experts, and a dynamic validation on several industry projects. This validation is part of a technology transfer process [9], which represents a responsible release of research results to a larger audience.

The validation resulted in a large number of smaller issues and some larger issues to resolve in the originally proposed Uni-REPM, which further motivates the argument that just because the constituents already are validated this does not remove the need to validate the whole as well. As a result, this article presents a refined Uni-REPM model where feedback on the accuracy, completeness, usefulness, and usability has been incorporated.

Below, the research questions that has guided the research are revisited.

RQ1. *To what extent is Uni-REPM suitable for industrial piloting, in terms of its correctness, completeness, and applicability?* A total of 65 feedback issues were collected from the interviewed domain experts. Most of them relate to the model correctness in terms of the action names, the maturity level the actions reside on as well as scope of the actions. This was expected as currently there are a lot of research going on in this area and there is as yet no well-defined set of terms or activities with any agreement in the research community.

There were only a few suggestions to add new actions. Moreover, the domain experts considered the amount of information presented in each action to be adequate for understanding the action and the benefits it brings, without overwhelming the reader. The domain experts found the "example" section useful for practitioners as it provides ideas on how to implement certain actions and links them to other literature sources for more information.

Regarding the applicability of the model, the experts considered all actions as useful and applicable, and no action should be removed from the model. Although most of the actions are applicable in both bespoke and market-driven development settings, some are more useful and essential in one setting or vice versa.

RQ2. *To what extent is Uni-REPM applicable , usable, and useful for industry application?* To this end the model lag, that is, the number of actions deemed inapplicable in a particular project, is studied. In the four evaluated projects, there is a certain model lag. The lag is smaller in market-driven projects, and in one case it is negligible, but there is still some room for improving the model. However, the overlap between the projects in terms of which actions that are inapplicable is small as only three actions are deemed inapplicable in more than one project. Thus, a deeper understanding of the development settings is required in order to further prune Uni-REPM.

Besides applicability, aspects of understandability and usability were also evaluated. The aspects were first judged on the usage of the checklist and then the model itself. The checklist is the operational form of the model. Except some minor difficulties in understanding the checklists, the practitioners claimed that they were familiar with most of the terms and concepts used. They also had no problem in understanding how the evaluation worked. In project 4 the evaluation was completely made by the practitioner on her own without any outsider's help.

All the interviews were conducted in one and a half hours, and the self-administered questionnaire took 40 minutes. This further confirms the light-weightedness of Uni-REPM.

RQ3. *What improvements can be done to Uni-REPM based on the findings in RQ1 and RQ2?* Although all 65 improvement suggestions contained valid arguments, not all of them were implemented in the improved version of the model. The reason for this is that several trade-offs have to be considered before being able to incorporate the remaining improvement suggestions.

Regarding RQ2, most of the problems lie in the checklist rather than the model itself. Therefore, modifications were made to the checklist to make it clearer and better reflect the intended practices. Some definitions were also added into the model description to explain specific concepts. Additional actions identified in the real processes were also considered to be added in the model.

In summary, the final version of Uni-REPM consists of 73 actions applicable in both bespoke and market-driven environments. It has performed satisfactory in four industry cases, where the detailed result analysis revealed not only the current state of the process in the evaluated projects but also relevant improvement suggestions for the companies.

Finally, Uni-REPM has managed to lessen the gap between academic research and industry adoption of the research findings by concentrating and presenting them in a practical and usable way that practitioners can easily and quickly apply. In conclusion, Uni-REPM is now verified to be a quick, easy, and cheap way to assess requirements engineering processes in a company.

8.1 Future work

During the construction of Uni-REPM, it was found that there is actually a notable amount of industry validation for many requirements engineering practices, with 50% being validated in one company, roughly 25% in two to seven companies, and 25% being common practice[5]. The four industry validations in this study place the validation status of Uni-REPM in the middle of this spectrum and well above the median. Nevertheless, further industry validation of Uni-REPM is encouraged. As part of the researchers own progress in this respect, an automated tool is being created to enable industry practitioners to use Uni-REPM and at the same time provide feedback on the model.

Since the intention of Uni-REPM is to be light-weight, there is a constant work of pruning the model of unnecessary actions and to add emerging actions that are deemed vital for a workable requirements engineering process.

Conducting more evaluations of requirements engineering practices provides an additional benefit, since it enables generating a map of the current state of practice in requirements engineering. This is, for example, helpful when assessing the feasibility new research ideas and results.

Work on these items are already well underway. For more information, please see the project's homepage: http://www.bth.se/tek/mdrepm.nsf.

There is an underlying assumption in the aforementioned future work that model-based process assessment is inherently beneficial for a company, and therefore, the focus is geared towards improving the model and the assessment tools. An interesting route for future work is thus to take the opposite route by investigating whether an Uni-REPM assessment actually leads to any meaningful process improvements, which in turn reduce for example, the number of product errors, the amount of rework time, or creates an increased understanding of the requirements during the development process. Such a research endeavour is in the early phases of planning together with a selection of industry partners.

[5] This is reported in a previous publication, as yet under submission.

References

1. Adam S, Doerr J, Eisenbarth M (2009) Lessons learned from best practice-oriented process improvement in requirements engineering: a glance into current industrial RE application. In: fourth international workshop on requirements engineering education and training (REET), IEEE Computer Society, Los Alamitos, pp 1–5
2. Beecham S, Hall T, Rainer A (2005) Defining a requirements process improvement model. Softw Qual J 13(3):247–279
3. Boehm B, Papaccio P (1988) Understanding and controlling software costs. IEEE Trans Softw Eng 14(10):1462–1477
4. CMMI-Dev (2006) Cmmi for development version 1.2 cmmi-dev v1.2. Tech Rep, CMU/SEI-2006-TR-008, Software Engineering Institute, Carnegie Mellon University
5. Creswell J (2003) Research design: qualitative, quantitative, and mixed methods approaches. Sage Publications, Thousand Oaks
6. Davis A (2005) Just enough requirements management: where software development meets marketing. Dorset House Publishing Co., Inc., New York
7. Fricker S, Gorschek T, Byman C, Schmidle A (2010) Handshaking with implementation proposals: negotiating requirements understanding. IEEE Softw 27(2):72–80
8. Fricker S, Gorschek T, Myllyperkio P (2007) Handshaking between software projects and stakeholders using implementation proposals. In: Proceedings of requirements engineering: foundation for software quality, vol. Lecture notes in computer science LNCS4542, pp 144–159. Springer, Berlin
9. Gorschek T, Garre P, Larsson S, Wohlin C (2006) A model for technology transfer in practice. IEEE Softw 23(6):88–95
10. Gorschek T, Garre P, Larsson S, Wohlin C (2007) Industry evaluation of the requirements abstraction model. Requir Eng 12(3):163–190
11. Gorschek T, Gomes A, Pettersson A, Torkar R (2011) Introduction of a process maturity model for market-driven product management and requirements engineering. J Soft Maint Evol Res Pract 24(1):83–113
12. Gorschek T, Svahnberg M (2005) Engineering and managing software requirements, chap. Requirements Experience in Practice: Studies of Six Companies. Springer, Berlin
13. Gorschek T, Svahnberg M, Tejle K (2003) Introduction and application of a lightweight requirements engineering process. In: Proceedings of the ninth international workshop on requirements engineering: foundation for software quality (RESFQ)
14. Gorschek T, Wohlin C (2004) Packaging software process improvement issues: a method and a case study. Softw Pract Experience 34(14):1311–1344
15. Gorschek T, Wohlin C (2006) Requirements abstraction model. Requir Eng 11:79–101
16. Hall T, Beecham S, Rainer A (2002) Requirements problems in twelve software companies: an empirical analysis. Softw IEE Proc 149(5):153–160
17. Hockman K, Grenville R, Jackson S (1994) Road map to iso 9000 registration. Qual Prog 27(5):39–44
18. IEEE Std 830-1998 (1988) IEEE recommended practice for software requirements specifications, Los Alamitos
19. Ivarsson M, Gorschek T (2009) Technology transfer decision support in requirements engineering research: a systematic review of rej. Requir Eng 14(3):155–175
20. Ivarsson M, Gorschek T (2010) A method for evaluating rigor and industrial relevance of technology evaluations. Empir Softw Eng 16(3):365–395
21. Juristo N, Moreno A, Silva A (2002) Is the European industry moving toward solving requirements engineering problems? IEEE Softw 19(6):70–77

22. Kamsties E, Hörmann K, Schlich M (1998) Requirements engineering in small and medium enterprises. Requir Eng 3:84–90
23. Karlsson L, Dahlstedt Å, Natt och Dag J, Regnell B, Persson A (2002) Challenges in market-driven requirements engineering—an industrial interview study. In: Proceedings of the eight international workshop on requirements engineering: foundation for software quality (REFSQ), Essen, Germany, pp 37–49
24. Karlsson L, Dahlstedt Å, Regnell B, Natt och Dag J, Persson A (2007) Requirements engineering challenges in market-driven software development—an interview study with practitioners. Inf Softw Technol 49(6):588–604
25. Kautz K, Hansen H, Thaysen K (2000) Applying and adjusting a software process improvement model in practice: the use of the ideal model in a small software enterprise. In: Proceedings of the 22nd international conference on software engineering, ACM Press, New York, pp 626–633
26. Leffingwell D (1997) Calculating your return on investment from more effective requirements management. Am Program 10(4): 13–16
27. Lubars M, Potts C, Richter C (1992) A review of the state of the practice in requirements modeling. In: Proceedings of the IEEE international symposium on requirements engineering, IEEE Computer Society, Los Alamitos, pp 2–14
28. Neill C, Laplante P (2003) Requirements engineering: the state of the practice. IEEE Softw 20(6):40–45
29. Niazi M, Cox K, Verner J (2006) An empirical study identifying high perceived value requirements engineering practices. In: Nilsson A, Gustas R, Wojtkowski W, Wojtkowski W, Wrycza S, Zupancic J (eds) Advances in information systems development, Springer, Berlin, pp 731–743
30. Nikula U, Sajaniemi J, Kälviäinen H (2000) A state-of-practice survey on requirements engineering in small-and-medium-sized enterprises. Tech rep, Lappeenranta University of Technology
31. Pettersson F, Ivarsson M, Gorschek T, Ohman P (2008) A practitioner's guide to light weight software process assessment and improvement planning. J Syst Softw 81(6):972–995
32. Sawyer P, Kotonya G (2001) Sofware engineering body of knowledge (SWEBOK 1.0), 1.0 edn, chap. Software Requirements, IEEE Computer Society, pp 9–35
33. Sommerville I (2004) Software engineering. 7th edn. Addison-Wesley, Harlow
34. Sommerville I, Sawyer P (1997) Requirements engineering: a good practice guide. John Wiley & Sons Inc, Chichester
35. Spice (2011) Software process improvement and capability determination. http://www.sqi.gu.edu.au/spice/
36. Staples M, Niazi M, Jeffery R, Abrahams A, Byatt P, Murphy R (2007) An exploratory study of why organizations do not adopt CMMI. J Syst Softw 80(6):883–895
37. Svahnberg M, Gorschek T, Feldt R, Torkar R, Saleem SB, Shafique MU (2009) A systematic review on strategic release planning models. Info Softw Technol 52(3):237–248
38. TickIt (2001) The tickit guide—using iso9001: 2000 for software quality management system, construction, certification and continual improvement. Tech Rep Issue 5.0, British Standards Institute
39. Wohlin C, Runeson P, Höst M, Ohlsson M, Regnell B, Wesslén A (2000) Experimentation in software engineering—an introduction. Kluwer, Dordrecht