Requirements Engineering (2023) 28:75–96
https://doi.org/10.1007/s00766-022-00387-3

ORIGINAL ARTICLE

Specifying requirements for collection and analysis of online user feedback

M. Astegher[1] · P. Busetta[1] · A. Gabbasov[2] · M. Pedrotti[1] · A. Perini[2] · A. Susi[2]

Received: 22 June 2021 / Accepted: 9 August 2022 / Published online: 16 September 2022

Abstract
According to data-driven Requirements Engineering (RE), explicit and implicit user feedback can be considered a relevant source of requirements, thus supporting requirements elicitation. However, limited attention has been paid so far to the role of online feedback in RE tasks, such as requirements validation, and on how to specify what online feedback to collect and analyse. We performed an action research study, together with a company that developed a platform for online training. This paper presents the design and execution of the study, and a discussion of its results. This study provides evidence about the need of practitioners to follow a simple but systematic approach for specifying requirements for data collection and analysis, at design time. Another outcome of this study is a method to tackle this task that leverages goal-oriented requirements modelling combined with Goal-Question-Metric. The applicability of the method has been explored on two industrial evaluations, while the perceived effectiveness, efficiency and acceptance have been assessed with practitioners through a dedicated survey.

Keywords Data-driven requirements engineering · User feedback · Goal-question-metric · Goal-oriented requirements analysis

1 Introduction

Data-Driven Requirements Engineering (DDRE) provides methods and techniques at support of software developers and analysts willing to exploit user feedback for eliciting, prioritising, and managing requirements for their software products [1]. RE research has devoted huge attention to automating DDRE, but several challenges remain to be addressed in order to better integrate DDRE into a continuous software development process, as discussed, for instance, in [2–4]. The opportunity to further research on how to leverage user feedback not only at requirements elicitation, but also at other stages of the software requirements life-cycle is highlighted in [4, 5] along with the need to enact traceability of feedback to software design artefacts.

In our research, we focus on online user feedback that is generated upon usage of a software application or service. It includes implicit user feedback, that is data generated during a usage session and collected via dedicated monitoring mechanisms, and explicit user feedback, e.g. user reviews, which is collected on dedicated channels or social media [6]. More specifically we investigate how requirements for collection and analysis of online user feedback are identified and what method could support developers in performing this task. This is particularly important for online implicit user feedback to avoid the risk that developers and analysts struggle to interpret collected data, or even worse miss opportunities of collecting the right ones that would help validating if the system they built meets stakeholders' goals.

Our research objective is twofold. Firstly, we aim at understanding what usage data should be collected and analysed for the purpose of system requirements validation and

✉ A. Perini
perini@fbk.eu

M. Astegher
maurizio.astegher@deltainformatica.eu

P. Busetta
paolo.busetta@deltainformatica.eu

A. Gabbasov
agabbasov@fbk.eu

M. Pedrotti
matteo.pedrotti@deltainformatica.eu

A. Susi
susi@fbk.eu

[1] Delta Informatica SpA, Trento, Italy

[2] Fondazione Bruno Kessler, Trento, Italy

evolution in a DDRE approach; we also analyse if developers need a method for the systematic identification of requirements for such user feedback collection. Secondly, we aim at defining a method for specifying requirements for data collection and analysis (that we call requirements for user feedback management, or *UF* requirements in short), in a systematic way. Moreover, we aim at assessing the applicability of the proposed method.

Motivations for this work derive from an industrial project in which a platform for online training was adapted as a citizen information service during the COVID-19 pandemic. Towards achieving our research objectives, we adopted an action research approach organised in four cycles [7, 8]. We first analyse examples taken from this software application for COVID-19 management and discuss with members of the project team about state of practice on exploiting online user feedback. Then, we take inspiration from goal-oriented approaches for Business Intelligence, e.g. [9, 10], and investigate whether concepts from Goal-Question-Metric (GQM) [11] can be exploited to define a systematic approach at support of developers in understanding why and what online user feedback to collect and analyse. In order to evaluate the applicability of the proposed method, we use it in two company's projects. Efficiency and acceptance of the method, as perceived by practitioners, is assessed with the help of a *Quality-in-use* evaluation model [12].

Main results can be summarised as follows: from the interaction with practitioners, we understood that in order to leverage user feedback to assess the success of specific features of a software application we should go beyond selecting mechanisms provided by app deployment platform, such as number of app downloads. A systematic approach is needed to guide practitioners at design-time to identify what feedback to collect and how to analyse it in order to assess the satisfaction of a specific stakeholder's goal; the proposed *GO+GQM* method can provide such guidance to practitioners. Its applicability has been assessed on two industrial projects, and a preliminary evaluation of its Quality-in-Use has been obtained by a survey.

A preview of this work was first presented at the 27th International Working Conference on Requirement Engineering: Foundation for Software Quality (REFSQ'21) [13]. This journal paper extends the conference paper along the following aspects:

- The problem we focus on, which we characterise with more details;
- The proposed method, called *GO+GQM*, for which a more consolidated version is presented;
- The overall design of the 4-cycle action research study;
- The results of those cycles in our study include a more structured evaluation of the applicability, perceived efficiency and acceptance of the *GO+GQM* method. The conference paper was focusing on the experience of the first two cycles of the study.

The rest of this paper is organised as follows. We recall background concepts and discuss related work in Sect. 2. The *GO+GQM* method is presented in Sect. 3. We describe the action research study's objectives and the research design in Sect. 4. Execution of the activities planned for each cycle of the action research and their main findings are discussed in Sect. 5. In Sect. 6, a discussion of lessons learnt and limitations of the study are presented. Then, Sect. 7 concludes the paper highlighting ongoing and future steps in our research.

2 Background and related work

2.1 Background

Goal-Question-Metric (GQM) is a top-down method for deriving and selecting a set of metrics to assess the achievement of high-level goals [11]. A high-level goal is decomposed into sub-goals. Questions referring to what could help stating that those goals are achieved are then identified. Metrics are derived, which individually or in an aggregated form can help answering each question. GQM has been introduced first in software engineering but it has been widely applied in different contexts, including business strategies assessment [14]. Our research applies GQM to project stakeholders' goals to evaluate if a software application that was envisioned as a means to achieve strategic goals meets the purpose, as well as to functional and quality goals that represent users' requirements for a software application, with the aim of defining, in a top-down breakdown, what data to collect and how to analyse them in order to get evidence about user's requirements satisfaction.

A second relevant ingredient of the approach is the Tropos [15] goal-oriented methodology, that allows to represent actors, their goals and tasks, and their social dependencies. This method for specifying requirements for data collection and analysis takes inspiration from goal-oriented approaches for Business Intelligence, e.g. [9, 10]. These works propose using the strategic representation and reasoning typical in goal-oriented modelling, connected with indicators that allow to measure the satisfaction of business and strategic goals.

2.2 Related work

Online, explicit user feedback is largely applied in software personalisation, e.g. [16, 17] and in recommendation systems [18], as a way to improve the user satisfaction.

By contrast, business process analysis and process mining collect implicit user feedback by extracting knowledge

about processes from transaction logs, aiming at detecting or preventing misbehaviour and monitoring process quality. In process mining, three main data analysis perspectives have been proposed, namely the process perspective, the organisation perspective and the case perspective [19]. Several examples of the application of process mining in medical and healthcare environments can be found in literature; for instance, [20] presents a methodology for the application of process mining techniques that leads to the identification of regular behaviour, process variants, and exceptional medical cases, while in [21] the process mining and predictive monitoring is used for the analysis related to health problems diagnoses and therapies. In [22], a literature-based metamodel is presented, which captures the most relevant knowledge elements that have been considered so far, including the notion of an actor's goal.

By contrast, in our work we consider the user's goals perspective as part of the software system requirements specification.

In RE, online user feedback, as defined in [1, 6], is considered as an important source for requirements elicitation and evolution. A huge amount of research on the analysis of explicit user feedback, such as App reviews[1], for software engineering purposes has been developed in the last fifteen years. Research progress and open problems are discussed in literature reviews, e.g. [23]. Dabrowski et al. [24] extend [23]: more than 180 research papers published between 2012 and 2020 were analysed. Among the purposes of this systematic literature review is that of identifying which software engineering activities are mostly supported by app reviews analysis. Fourteen tasks are considered, which correspond to requirements engineering, design, testing, and maintenance goals. Results indicate that 34% of the considered primary studies (i.e. 62 studies) analyse app reviews for requirements engineering purposes, with the following frequency: (i) requirements elicitation (30%), (ii) requirements prioritisation (10%), (iii) validation by users (11%). Our study positions in group (iii) as it contributes to investigate how to support the exploitation of online user feedback for requirements validation and evolution.

In [5], the use of explicit feedback in app stores is investigated with the purpose of understanding how it can influence the behaviour of practitioners in different software engineering tasks and what are the main objectives behind feedback analysis. The authors conducted a set of interviews and administered a questionnaire to elicit this information directly from app developers. The main results show that the majority of the practitioners analyse user feedback to validate and elicit new requirements to support alpha/beta testing of the apps, and to maintain and evolve the apps. Moreover, the activities for the validation and evolution of the app are strongly influenced by the directions emerging from the feedback. A difficulty expressed by the developers has been that of extracting useful information from the feedback because of its high volume and sometimes high degree of noise, so expressing the need of methods for feedback management and analysis. Similarly, in our study we investigate how practitioners can leverage user feedback for the purpose of software requirements validation and evolution, but in the context of an action research study. Differently from [5], we also propose a method to be used at design time for identifying what user feedback can help assess the satisfaction of specific requirement goals.

Combining implicit and explicit user feedback analyses for requirements engineering purposes has also been investigated, e.g. [25], as well as the possibility to consider as implicit user feedback other data about the user, such as multisensorial and physiological data of the users that can be collected through monitoring mechanisms [26].

In our research, we focus on how to specify what online user feedback to collect and analyse for a requirements engineering purpose, namely requirements validation and evolution.

3 The *GO+GQM* method

The proposed approach aims at supporting the elicitation and specification of requirements for user feedback collection and analysis (*UF* requirements for short) for the purpose of requirements validation and evolution in a DDRE approach. Key steps of the method are depicted in Fig. 1. Given a Goal-Oriented (GO) model of the requirements of the software application at interest (step 1. in Fig. 1), the development team can identify which requirements could be evaluated thanks to the analysis of user feedback to be collected upon deployment of the software application. Following the *GO+GQM* method a GQM model is built in this step. The *evaluation goal* for a given requirements of the software application is stated first, then *questions* that should be answered in order to satisfy the evaluation goal are identified. Appropriate metrics based on user feedback are stated along with *indicators* and *indicator levels* that help identify the success and quality in answering the associated question. Such *UF* requirements will be then operationalised via the selection and/or configuration of suitable feedback gathering and analysis tools such as those described in [25].

UF requirements specification can be done together with the specification of functional and non-functional requirements of the software application under consideration supporting their evolution during the application life-cycle. In the rest of the section an illustrative example is presented first, then a meta-model of the basic concepts of the

[1] That is textual feedback associated with a star rating that app users can provide through App store channels.

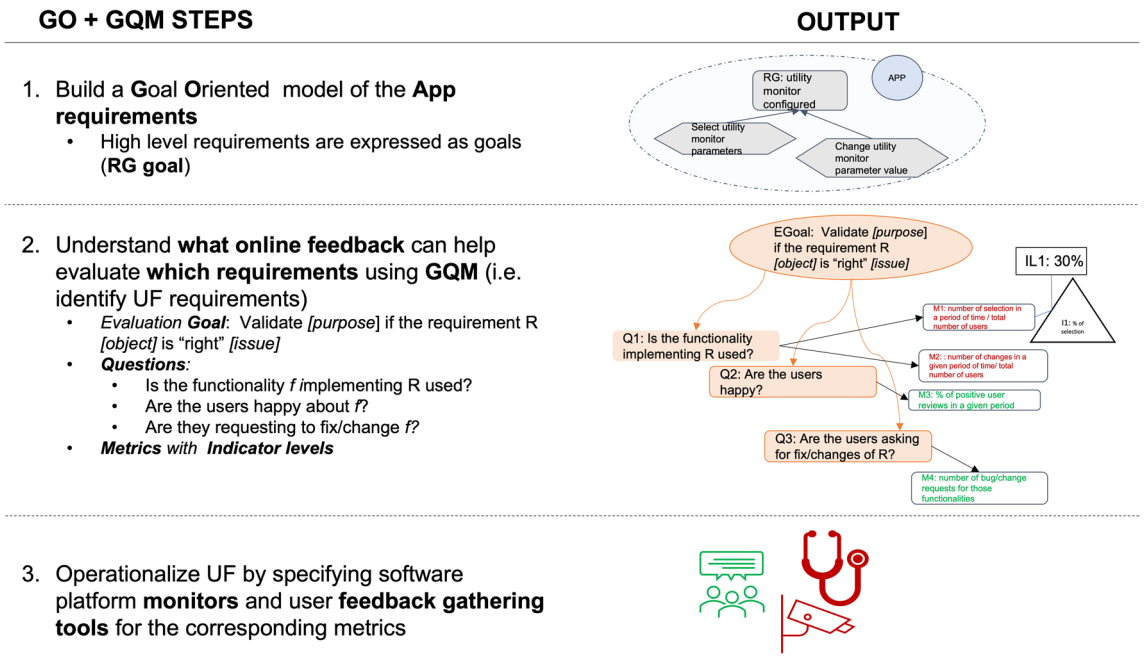


Fig. 1 Key steps of the *GO+GQM* method. An illustrative example is given in Sect. 3.1

approach and a description of how the method can integrate with development tasks are given.

3.1 Illustrative example

To introduce the proposed approach, we use an illustrative example taken from apps for home energy efficiency management[2]. Users of such apps usually aim at contributing to sustainable energy usage. Typical features offered by these apps aim at meeting three types of users' goals: (1) reduce energy consumption costs; (2) configure home utilities operation; and (3) configure utility monitors. An excerpt from the apps requirements corresponding to these user needs is depicted in Fig. 2, right side. The goal model refers to the requirement goal RG1, *Enable end-users to monitor and analyse their energy consumption*, that is further AND-decomposed into the three subgoals RG1.1, *utility added/removed*, RG1.2, *utility status monitored*, and RG1.3, *target demand & action configured*. The tasks implementing the three subgoals are T1.1, *Add/remove utility monitor* (for RG1.1), T1.2a, *visualise utility monitor status* and T1.2b, *query utility status* (for RG1.2), T1.3a, *select utility*, and T1.3b, *set target & action* (for RG1.3).

Besides the users, the app developer and the domain expert are key stakeholders for building a successful app. We consider RG1.3 from the goal model in the right side of Fig. 2 taking the perspective of the practitioners and applying GQM (see the left side of the same figure).

The left side depicts a goal EG1 and its associated questions (Q1-EG1, Q2-EG1, Q3-EG1) to be answered while the app is in use, that is, via the analysis of its logs or by analysing users' feedback upon their usage of the app. Specifically, to reply question Q1-EG1, implicit feedback obtained via suitable logs is needed, while to answer the questions Q2-EG1 and Q3-EG1 explicit user feedback channels need to be adopted.

The related metrics are defined by looking at the detailed requirements in the requirements model. The measures considered to respond to Q1-EG1 are M1, *average # of utility selection in a time period* and M2, *average # of target + action setting in a time period*. The two measures have their associated indicators and indicators levels I1 (*% of selection of utility*) and I2 (*target + action*) with levels 30% and 40%, respectively. Similar considerations can be done for the Measures M3 and M4 and related indicators I3 and I4 for the explicit feedback.

The bottom part of Fig. 2 shows an excerpt from the connections between the measures in GQM and the goals and tasks in the goal model. Specifically, the measure M1 *evaluates* the goal RG1.3 by providing *measurements* on the app functionality corresponding to the task T1.3a that operationalises the goal via the means-ends relationship.

3.2 Meta-model

Figure 3 depicts the meta-model of the *GO+GQM* method, which is composed of three main parts: the left part concerns the concepts taken from GQM and is related to the

[2] In particular, we inspire to the software application used in [27].

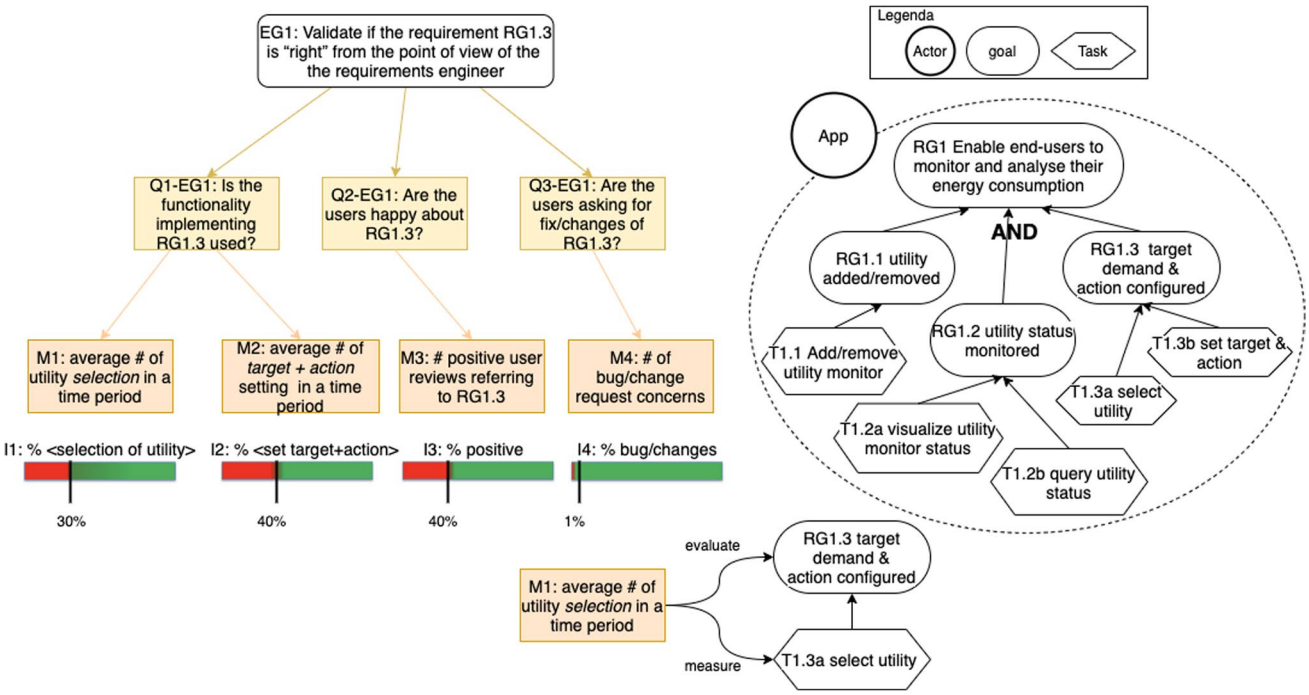


Fig. 2 Excerpt from the requirements model (right side). GQM applied to the requirement goal RG1.3 (left side). The connection between the measure M1 and the goal RG1.3 and the task T1.3a (at the bottom)

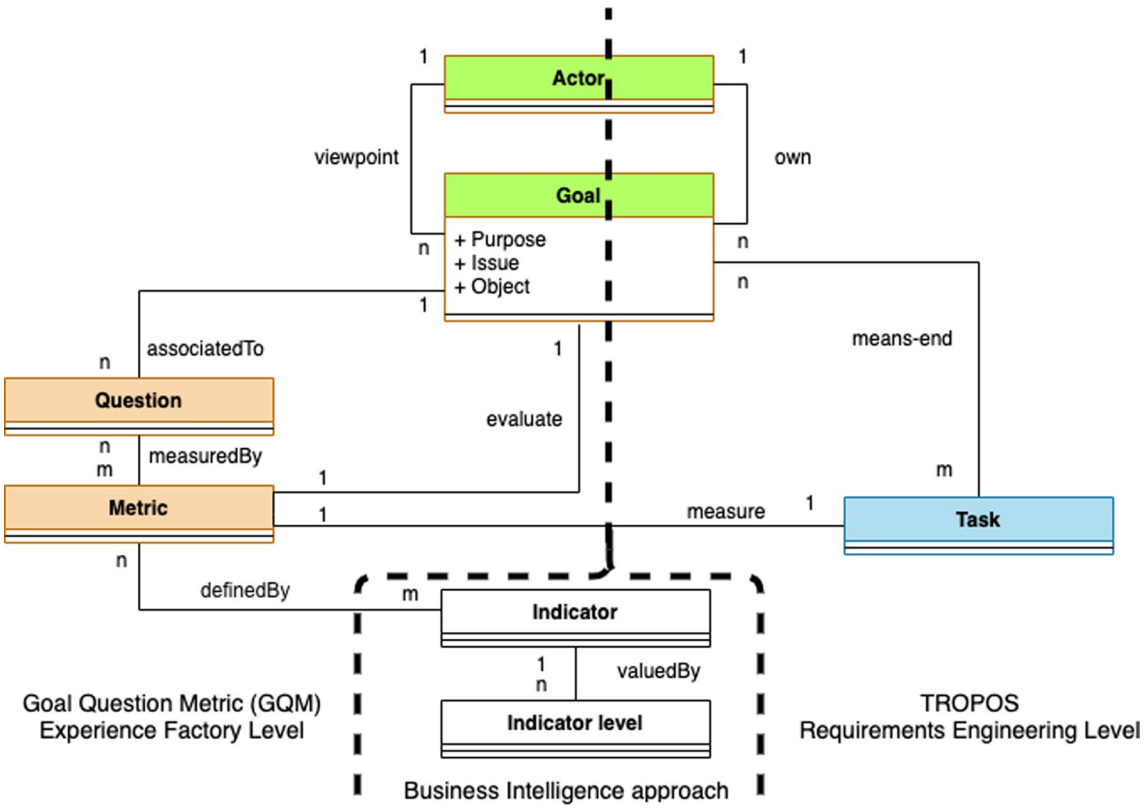


Fig. 3 The metamodel: on top, in green, the concepts that are common to Tropos [15] and GQM [14], on the left, in orange, the concepts from the GQM approach, on the right, in blue, the concept from an excerpt of the Tropos metamodel, in the middle the concepts from the business intelligence approach from Barone et al. [10]

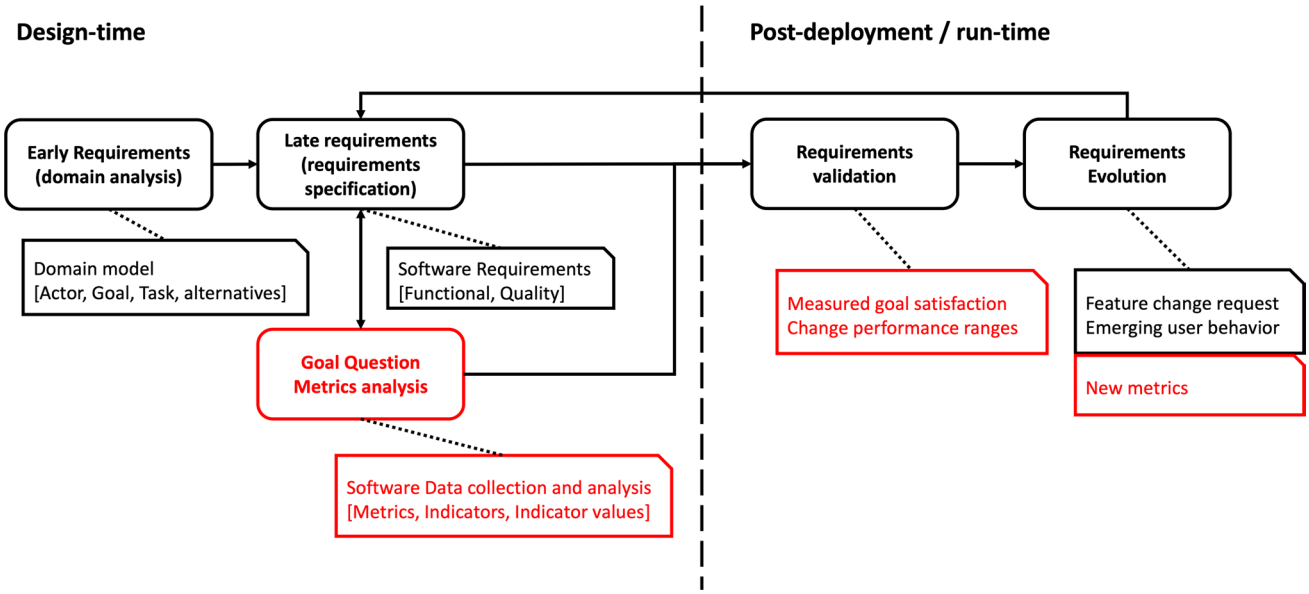


Fig. 4 The envisioned process with new tasks and artefacts highlighted in red

Experience Factory level as introduced in [11], the right part is related to the TROPOS goal-oriented methodology [15] and refers to the *Requirements Engineering level*, the bottom part represents the *Indicator* and *Indicator level* concepts proposed by the business intelligence approach presented in Barone et al. [10]. The concepts of *Actor* and *Goal*, reported in green, are shared by GQM and TROPOS. The choice of the three methodologies is motivated by the presence of concepts in the respective meta-models that can be interconnected. These meta-models can be easily composed in a coherent framework for the representation of both the system goals and the metrics for the monitoring of the satisfaction of the goals themselves.

Considering the right part of the meta-model, the concepts of *Actor*, *Goal*, *Task* are the concepts that in TROPOS allow to describe the domain of a given organisation and in particular the network of actors, their goals and the processes that characterise the way the goals are accomplished in the organisation. This part describes the instruments a Requirements Engineer may use to analyse and describe the user requirements or the requirements of the software system.

The left part of the meta-model describes the concepts related to the GQM framework that are used to build and evolve the model of requirements (the Experience Factory level as introduced in [11]). As observed above, also in this part of the meta-model the concepts of *Actor* and *Goal* are present. The *Goal* is characterised by a perspective (the goal belongs to an actor) and by three aspects: *Purpose*, *Issue*, *Object*. The three aspects describe a quality (the Issue), the change that would be produced on this Issue (the Purpose) and the Object to which the Issue applies [11]. For instance, considering the evaluation goal EG1 in Fig. 2, we have that "validate" is the Purpose, "right" is the Issue, requirement RG1.3 is the Object.

The other concepts in the meta-model are the *Question* and *Metric* that describe the other two pillars of the GQM framework. *Question* evaluates a Goal, while the *Metric* is the way one or more Questions are measured. Moreover, the *Metric* evaluates the Goal and measures the Task in the Requirements Engineering level, so being a bridge between the two parts of the meta-model.

Finally, the concepts of *Indicator* and *Indicator Level* provide concrete definitions for *Metric*. The *Indicator level* represents the threshold that determines whether a metric in a GQM path gets a meaningful value, according to a given indicator. Looking at the example depicted Fig. 2, the *Indicator* "I3: % of positive user reviews" provides a concrete way to define the metric "M3: # positive user reviews", that refers to RG1.3 in the example. The *Indicator level* related to I3 specifies that if the % of positive user review is greater than 40% we can answer positively to the question about the users being happy with the App for monitoring and analysing energy consumption.

3.3 User feedback requirements specification and related tasks

Figure 4 shows an overview of a possible process that integrates the activities and artefacts related to the analysis of the requirements for user feedback gathering. The process is inspired to the TROPOS development process [28] that

has been enriched with activities concerning the systematic identification of requirements for user feedback analysis. The new activities and related artefacts are reported in red. We distinguish between design-time and post-deployment (run-time) tasks. At design-time, during the Early requirements activity the key stakeholders are identified together with their main goals and tasks as well as the alternative ways goals and tasks can be decomposed [15]. The subsequent late requirements activity consists in the specification of functional and quality requirements of the intended software application or service. In parallel with those activities the process includes a new activity, namely *Goal Question Metrics analysis*, that aims to define metrics, related indicators, and corresponding range of values (indicator values), which will help to assess to what extent the running application achieves the stakeholders' goals [11]. At deployment / run-time, once the software application has been deployed on a platform instrumented with data collection and analysis mechanisms and it is accessed by its intended users, two tasks can be performed, namely *Requirements validation* and *Requirements evolution*. Concerning Requirements validation, the implemented mechanisms for data collection and analysis can help requirements engineers to evaluate if (and to what extent) the software application meets the stakeholders' goals and, in addition, to validate knowledge used to define the data collection and analysis requirements, as for instance value ranges of the indicators used in the metrics for goal assessment. As for requirements evolution, feature change requests can be collected through explicit user feedback and ideas for new requirements that can emerge by the analysis of session logs. Moreover, we also foresee the possibility of eliciting ideas for new metrics from implicit user feedback, e.g. by aggregating indicators or by leveraging on process mining techniques. This information can then be reported to the design phase in order to evolve the metrics and the requirements of the system.

4 Research goal and design

We performed an action research study [7, 8] with the aim to investigate how to support practitioners' decisions about what type of online user feedback to collect and for what purpose. The following research questions were guiding us:

- *RQ1: Is online user feedback collected by practitioners? What type?*,
- *RQ2: When is it decided what user feedback to collect? For what purpose?*, and
- *RQ3: How is it decided what user feedback to collect and analyse? Is a systematic approach needed?*

Since a method for specifying *UF* requirements (namely the *GO+GQM* method) was defined, as additional question we considered:

- *RQ4: How effective and efficient is the GO+GQM method as perceived by practitioners? Will practitioners adopt this method?*

We took them into account when formulating specific diagnosis' goals and planning corresponding actions in the different study's cycles.

In the rest of this section, we first describe the context of our action research. An overview of the action research study that has been performed along the four cycles is then presented, followed by a description of the *Quality in Use* model that we adopted in the last cycle of the action research study.

4.1 Research context

Delta Informatica is a small-medium enterprise that developed a platform called ELEVATE for the creation of interactive, multi-media exercises that we call *IMM* exercises for short. *IMM* exercises can be used in online adult training as a complement to traditional in-presence training; the ELEVATE platform also manages the related online education sessions. The structure of an *IMM* exercise is a directed graph, as depicted in Fig. 5 (left side). It may contain cycles. Multi-media content, e.g. video clips or text describing a step in a procedure to be learned, can be associated with each node of the graph. The edges outgoing from a node represent choices that are offered to a trainee, and that correspond to decisions to be taken to progress in performing the procedure. Indeed, the destination of an edge is the point from which the training story will resume if the trainee takes that choice. While exploring an *IMM* exercise, the trainee can experience the effect of a wrong action/choice, by ending in a so-called failure node. When all the right actions/choices in a procedure are taken, the trainee reaches a success end-node.

The two key components of the ELEVATE platform are depicted in Fig. 5 (green colour). The exercise design tool (EDT) is used by a so-called instructional designer to create the exercise structure. The instructional designer can access multi-media resources or tools to create contents for each exercise's node. The management tool (EMT) allows to collect session logs and to produce learning analytics; it can be integrated with a learning management system (LMS), which is accessed by trainees attending online courses.

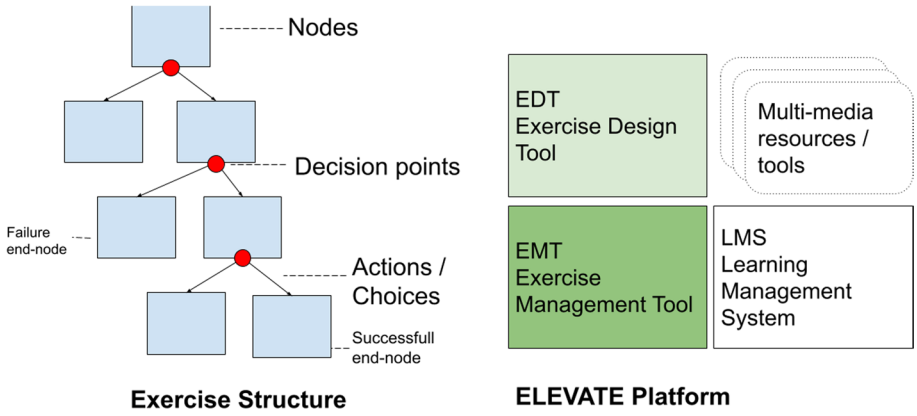


Fig. 5 Structure of an ELEVATE *IMM* exercise (left side); Key components of the ELEVATE platform (green colour, right side) (colour figure online)

Table 1 Members of team of the ELEVATE project and of the 4-cycle Action Research study

ID	Profile	ELEVATE project	Action study research
PM	Delta Informatica project manager	Yes	No
SC	Senior consultant (30-year experience)	Yes	Yes, all cycles
SD	Delta Informatica Senior developer who coordinated the development of the ELEVATE platform (10-year experience)	Yes	Yes, cycles c1, c3, c4
JD1	Delta Informatica Junior developer (4-year experience)	Yes	Yes, all cycles
JD2	Delta Informatica Junior developer (2-year experience)	Yes	Yes, cycle c4
JD3	Delta Informatica Junior developer (3-year experience)	Yes	Yes, cycle c4
SR1	Senior researcher of the SE research unit at FBK	Yes	Yes, all cycles
SR2	Senior researcher of the SE research unit at FBK	Yes	Yes, all cycles
JT1	Junior technologist of the SE research unit at FBK	Yes	Yes, cycle c4
SD-ext	Senior Developer of another company (7-year experience)	No	Yes, cycle c4

A more detailed description of the platform can be found in [29].

Between March 2019 and June 2020, researchers of the Software Engineering (SE) research unit in Fondazione Bruno Kessler (FBK) were formally involved in the ELEVATE project. Among the objectives of this collaboration were that of defining a methodology for the production of *IMM* exercises with the ELEVATE platform and that of identifying techniques to enable personalisation of a training session on the basis of trainee's characteristics. The profiles of the project team members are summarised in Table 1. The Delta Informatica team included three junior developers (between 2 and 4 years of work experience) who were assigned specific tasks in the platform development; see JD1, JD2, JD3 in Table 1. They were coordinated by a 10-year experienced developer (SD) with the help of a consultant, expert on agent-based technology and virtual reality technology who was the main ideator of the ELEVATE project (SC), and of a project manager. Two senior researchers (SR1, SR2) and one technologist (JT1) were part of the FBK team working in the ELEVATE project. The action research study presented in this paper has been conducted in the period October 2020–May 2021. As indicated in the last column of Table 1, all the participants to this study, apart one, were members of the team working for the ELEVATE project, thus they knew the ELEVATE platform and the type of intended customers.

4.1.1 The ELEVATE-COVID19 software application

During the state of emergency for the first wave of the COVID-19 pandemic, the local government of the Provincia Autonoma di Trento, Italy (PAT for short) needed a way to regularly inform citizens (more than half a million inhabitants) about the prudent and legally permitted behaviours to follow and those that did not comply with health advice and norms. Citizens struggled to know what was allowed and what was not, even if they were strongly motivated to follow the rules because they wanted, for example, (i) to behave in a responsible way, thus contributing to mitigate the risk of crisis for the healthcare system; (ii) to avoid risky behaviours for their own and family's health; (iii) to avoid being fined. Browsing through cryptic and lengthy regulations looking for clues concerning a specific topic can be a tedious and difficult process, and people often preferred to

Table 2 Action research study: diagnosing and planning

Cycle	Diagnosis	Action planning
c1	Did practitioners collect user feedback? What Type? How? Did they analyse it? Did practitioners specify *UF* requirements? When in the software development process? How?	We planned a meeting of researchers and practitioners. We proposed to use a shared document where practitioners were asked to write down their experience in the *ELEVATE-Covid19* project.
c2	Which available techniques can be used to define a systematic approach for *UF* requirements elicitation and analysis?	We planned to look at research literature on user-feedback in RE to select or identify a method for *UF* requirements, and to perform a preliminary evaluation of such method by applying it to examples taken from the *ELEVATE-Covid19* application.
c3	Is the *GO+GQM* method applicable to other case studies, in different application domains?	We planned to: perform another industrial evaluation on a different company's project; Identify the team of practitioners who could try to apply it and instruct practitioners about the method. Designed an interview for Delta Informatica customers (users of the ELEVATE platform).
c4	Is the *GO+GQM* method effective for identifying *UF* requirements? What's the practitioner's perception about efficiency and acceptance of the *GO+GQM* method?	We designed a survey and a 15' video tutorial about the *GO+GQM* method. Target subjects: practitioners.

interact directly with a PAT helpdesk via telephone even for the simplest questions.

To lighten the workload of the helpdesk call centre, PAT asked Delta Informatica to set up a Web-based system for accessing the information contained in the guidelines in an immediate and interactive manner. For each guideline, a detailed description of the allowed behaviours was produced; each of the latter was then associated with a reference category (e.g. *Sports and outdoor activities*) and to additional keywords. Web users could search by category or by keywords, similar to what happens in search engines. In many cases, text-based guidelines were accompanied by infographics and exercises carried out with the ELEVATE platform. As mentioned above, ELEVATE is primarily used for creating *IMM* exercises to be embedded into online training courses as mechanism to test and enforce knowledge. However, an *IMM* exercise can be used also as a *communication tool* that allows its user to explore scenarios. The COVID-19 exercises asked simple questions such as *Do you have to use public transport to reach your destination?* and showed a set of predefined answers to choose from. The exercises proposed alternative scenarios based on how the story progressed, ending when these led to situations of either correct or discouraged (if not prohibited) behaviours. The user was free to repeat any exercise and try different options.

Exercises were created and made available according to this workflow:

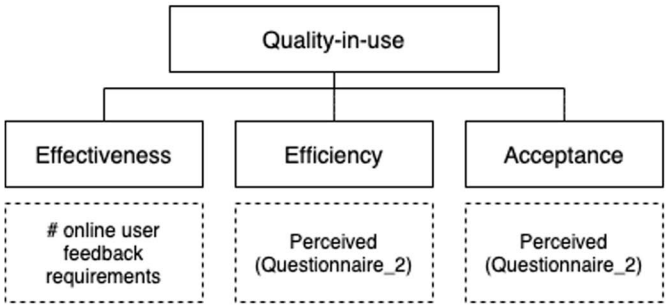
1. Production of multimedia contents by the PAT press office team (mainly infographics with guidelines to be followed);
2. Exercise design by the team of Delta Informatica, acting as software platform consultants for PAT;
3. Approval of the exercises by a domain expert, member of the PAT press office;
4. Exercises deployment—Delta Informatica also provided the hosting services for all ELEVATE components, while exercises were linked from the COVID-dedicated Web site of PAT;
5. Access by citizens to the online material and collection of usage data;
6. Analysis of the data collected via the EMT component of the ELEVATE platform, by both PAT and Delta Informatica.

The ELEVATE *IMM* exercises allowed to organise information so that only the relevant parts of the COVID-19 directives were progressively offered to the user, leading to simplicity and greater engagement compared to the sequential reading of norms written in legal language. Indeed, positive feedback on the exercises was informally collected both from users and from the press office of the local government.

Months after, the Delta Informatica team involved the FBK team to investigate if the positive feedback about the *ELEVATE-Covid19* application could have been confirmed by the data made available directly from the EMT component of the ELEVATE platform. This motivated the action research study we present in this paper, and more specifically contributed to the identification of the diagnosis' goals for the first cycle.

4.2 Action research study

Table 2 summarises design aspects about each cycle such as diagnosis's goals and actions we planned to perform with the purpose to address them. Main goals in the **first cycle** regard extracting information from the practitioners' experience in the *ELEVATE-Covid19* project to answer questions about



QUESTION	TYPE
It is clear how and for what purpose to use GQM+GO	5-likert
I find it easy to specify what user feedback to collect using GQM+GO	5-likert
I found the GQM+GO method easy to use	5-likert
It would be easy for me to become skillfull at deciding what user feedback to collect by using the GQM+GO	5-likert
Using the GQM+GO method will allow me to specify what user feedback to collect in order to cover all relevant aspects to be validated with users	5-likert
Using the GQM+GO method will allow me to specify what user feedback to collect more quickly (than without it)	5-likert
Using the GQM+GO method will allow me to specify what user feedback to collect more effectively (than without it)	5-likert
I will adopt the GQM+GO to specify what user feedback to collect and for which purpose	open
What do you feel are the strengths / weaknesses of the GQM+GO method?	open

Fig. 6 Adopted quality-in-use model, and excerpt of the *Questionnaire* (question, type)

the online user feedback that has been collected. Moreover, we wanted to investigate how practitioners decided about what user feedback to collect, and for what purpose. That is research questions *RQ*1, *RQ*2, and *RQ*3 are considered in cycle 1. Actions for addressing these goals include meeting project's team members, and setting up a shared document to enable collaborative writing to document answers to the proposed questions.

The goal of the **second cycle** was motivated by lessons learned from the first cycle. It concerns the identification of a systematic approach for the definition of what online user feedback to collect during the usage of an application, with the purpose to validate project's goals and software application requirements. Correspondingly we planned to perform an analysis of research literature about online user feedback in Requirements Engineering. The identified method should have been tested on examples taken from the *ELEVATE-Covid19* project.

The main goal in the **third cycle** concerns the assessment of the applicability of the proposed method for *UF* requirements specification. For this purpose, we planned to identify a different project for which the Delta Informatica team was using the ELEVATE platform and to apply the method to both company's projects in an extensive way. We also designed an interview to be executed with customers of Delta Informatica willing to use the ELEVATE platform.

The goal of the **fourth cycle** concerns the assessment of effectiveness, efficiency and acceptance of the *GO+GQM* method. That is the research question *RQ*4 is considered in this cycle. To address it, we defined the Quality-in-use model described in Sect. 4.3, with appropriate metrics, and planned to design and execute a survey.

A description of how we executed the planned activities, what artefacts we used for evaluation, and the lessons learnt by researchers and practitioners are discussed in Sect. 5.

4.3 Quality in use model

With the aim to evaluate the effectiveness, efficiency and acceptance of the *GO+GQM* method, as perceived by practitioners, we used a *quality-in-use* assessment model [30] as shown in Fig. 6.

Three main quality aspects are considered, namely *Effectiveness*, *Efficiency* and *Acceptance*. The metric we defined for evaluating *Effectiveness* is the number of *UF* requirements that help assess a requirement goal, which are defined by applying the method. This number is determined by the variety of questions that are derived from an evaluation goal, and by the number of indicators that are associated with these questions. The number of indicators gives an account of the degree of coverage of the questions derived from an evaluation goal. For *Efficiency* and *Acceptance*, we are interested in understanding the practitioners perspective, upon having used the method. For this purpose, we designed a questionnaire that we administered to practitioners. Questions like *Using the GO+GQM method will allow me to specify what user feedback to collect more quickly (than without it)* and *Using the GO+GQM method will allow me to specify what user feedback to collect more effectively (than without it)* are used to assess perceived efficiency. Questions like *It is clear how and for what purpose to use GO+GQM* and *I will adopt the GO+GQM to specify what user feedback to collect and for which purpose* are used to assess if

practitioners believe the method can be adopted (i.e. *Acceptance*). The full list of questions is shown in Fig. 6.

5 Action research execution and results

In this section, we focus on the evaluation and learning tasks we performed upon executing the activity that we planned for each cycle of the action research study. An overview is given in Table 3.

5.1 Cycle 1

Following the activities that we planned to address the goals of the first cycle, we held a virtual meeting on October 6, 2020. As a follow up to this meeting we decided to set up a shared document where main facts and excerpts of project artefacts that should have been evaluated to address the diagnosis's goals were reported by two practitioners (referred as JD1 and SD in Table 1).

Evaluating: Requirements artefacts for the *ELEVATE-Covid19* project are included in a vision document stating the key objective of the project, i.e. COVID 19 rules dissemination and understanding by citizens. Excerpts of log data collected with the EMT tool during the use of the application in May 2020 were described in the shared document and used for this first evaluation. They are data about the user behaviour, such as the speed of execution or the ability to follow the correct paths of an exercise. These data help to assess the achievement of predefined educational objectives. In training, this data are used to evaluate the performance of individuals and groups of students, and allows to tune the exercises e.g. by making situations harder or easier to interpret and by offering alternatives to follow. Unfortunately, while these data give us a very precise picture of the interests of the *ELEVATE-Covid19* 's users (for instance, how many took a certain path within an exercise, revealing to be interested to specific situations), they do not allow to answer a few basic questions: (i) Did users find the information they were looking for? (ii) Overall, was using the exercise a positive or a negative experience? (iii) Are there extensions or improvements of the *ELEVATE-Covid19* application we should consider after looking at the users' behaviours? Indeed, the main metrics available at that time from EMT reflect the perspective of a teacher who has educational objectives to satisfy and thus need to make sure that students have successfully followed certain paths and that the interactive experience they had is translated in permanent knowledge applicable to real-life situations.

Learning: It was not considered from start how to exploit user feedback to assess the satisfaction of project (stakeholders) goals or software requirements. Whether specific usage data should have been collected in order to evaluate the achievement of project's strategic goals or software requirements was not discussed at design time. The "usual" log data, according to the basic configuration of the e-training platform were collected. As an after-thought, it would have been easy to extend the exercises to collect feedback concerning the questions mentioned above. Even better, it would have been technically feasible to instrument the platform to collect that feedback by default, e.g. asking for a simple rating at the end of a run. Practitioners got aware that a systematic method for *UF* requirements elicitation and specification should be adopted at design time. **Input to next cycle**: researchers got motivated to analyse literature on online user feedback to search for suitable methods to guide practitioners to identify what user feedback could be useful to assess specific feature of a software application, which are linked to a stakeholder's goal, and specifically to set-up indicators and indicator levels to measure goal satisfaction.

5.2 Cycle 2

In our literature analysis we mainly found techniques for mining user reviews for RE purposes. Only a few works focus on approaches that combine the analysis of implicit and explicit user feedback. Concerning approaches for the exploitation of implicit user feedback for assessment purposes we found work in Business Process and Business Intelligence. We shortly recall these research works in Sect. 2. We found that relevant approaches that guide data collection for goal assessment in software project rests on GQM [11] and its extension.

Evaluating: This motivated our proposal of the *GO+GQM* method described in Sect. 3, which we evaluated on one example taken from the *ELEVATE-Covid19* application, as reported in [13].

Learning: Contemporary literature analysis, e.g. [24], and a more focused analysis conducted by the researchers on methods for identifying and specifying *UF* requirements pointed out the lack of suitable methods for *UF* requirements specification. Researchers considered to use basic approaches for assessing software development projects (i.e. GQM). This led to the proposal of the *GO+GQM* method presented in Sect. 3. The preliminary application of the *GO+GQM* method to the *ELEVATE-Covid19* project provided evidence to both researchers and practitioners about the method applicability. **Input to next cycle**: This motivated us to perform a third cycle in our action research study, with the aim to consolidate the *GO+GQM* method and to apply it in two different projects with the purpose of evaluating its applicability.

Table 3 Action research study: evaluating and learning

Cycle /When	Evaluating	Learning
c1 (Oct 2020)	Comments of practitioners (SC, SD, JD1) on the *ELEVATE-Covid19* experience, and their notes reported in the shared document. Vision document of the project. Log data collected with the EMT tool in May 2020.	Artefacts: preliminary analysis of the log data collected in May 2020 in light of stakeholders' goals documented in a project report and partially published in [13]. Practitioners got aware that a systematic method for *UF* requirements elicitation and specification should be adopted at design time. Input to next cycle: researchers got motivated to analyse literature on online user feedback to search for suitable methods.
c2 (Oct–Nov 2020)	Research literature on user feedback in RE. Preliminary results from the application of *GO+GQM* to an example taken from the *ELEVATE-Covid19* application data.	Upon literature analysis the researchers did not find a method for identifying and analysing *UF* requirements, thus they considered basic approaches for assessing software development projects (i.e. GQM). The application of a first version of the *GO+GQM* method provided preliminary evidence to both researchers and practitioners about the method applicability. This work was presented in scientific papers [13]. Input to next cycle: researchers got motivated to consolidate the *GO+GQM* method and to evaluate it more extensively.
c3 (Apr–May 2021)	Comments from a series of meetings between researchers (SR1, SR2) and practitioners (SD, JD1). Artefacts produced during the application of *GO+GQM* to the two case studies.	Main output is the *GO+GQM* method that was described in a project report (and in this paper), and a short video tutorial for practitioners. Practitioners learnt how to use the method. Input to next cycle: results from the application of the method to two projects (reported in Sect. 5.2 in this paper) encouraged researcher to perform a Quality-in-Use evaluation.
c4 (May–Jun 2021)	Data collected from the survey were analysed to assess perceived: Effectiveness; Efficiency; Acceptance.	Main output: survey design on Quality-in-Use of the *GO+GQM* method. Findings from the seven subjects, who participated to the survey, are reported in Sect. 5.4 in this paper.

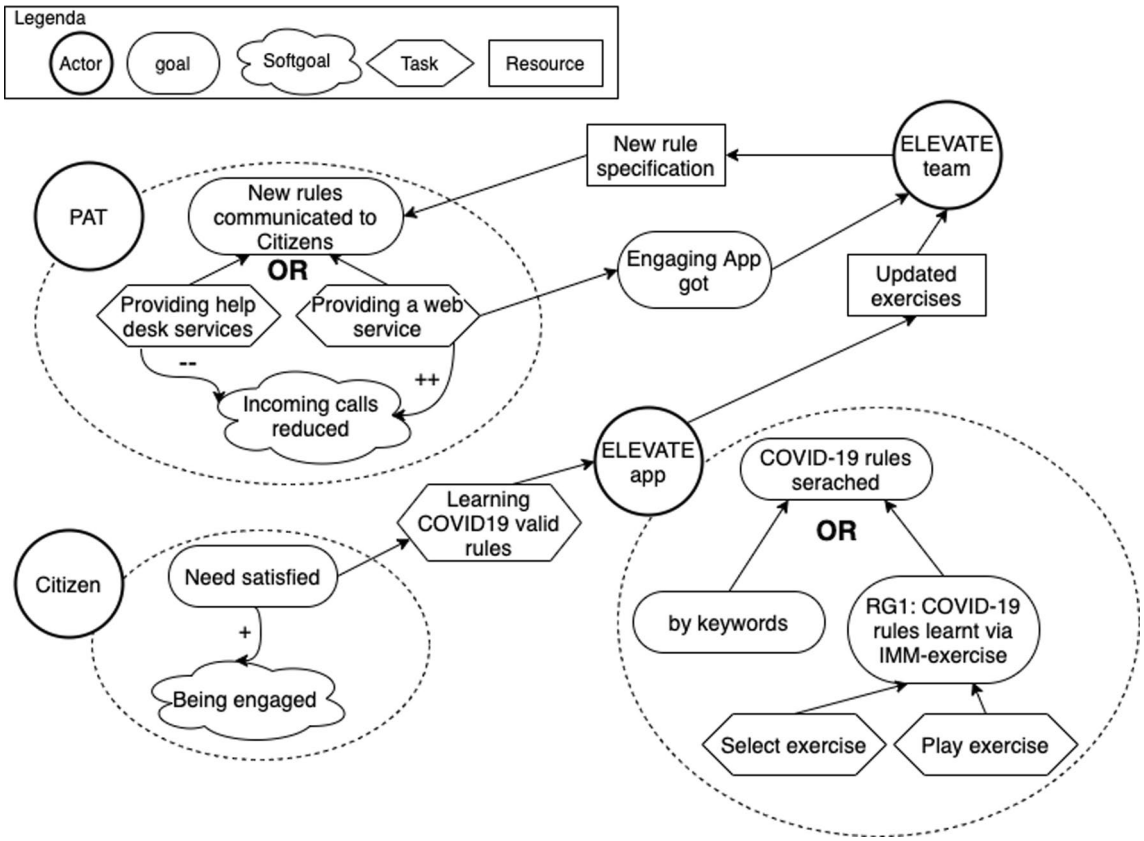


Fig. 7 Requirement goal model in TROPOS notation

5.3 Cycle 3

In this cycle the *GO+GQM* method was applied to two different industrial projects. The first is the same project used in cycle 2, namely the *ELEVATE-Covid19* project, in which the ELEVATE platform is used for mass communication purposes. The second concerns the use of the ELEVATE platform for training purposes. The method was applied by the project teams, and then discussed and revised with researchers. In the rest of this section we present results from the application of the *GO+GQM* method to these two case studies. We then conclude the section recalling main lessons learnt in this cycle.

5.3.1 Industrial evaluation 1

The team of Delta Informatica involved in the *ELEVATE-Covid19* project performed a post-mortem analysis and applied the proposed method. Some of the identified metrics were already implemented in the EMT tool. The data collected with those metrics are analysed to get evidence of the usefulness of the specified *UF* requirements, for the purpose of validating key requirements of the *ELEVATE-Covid19* software application.

UF requirements specification: GQM is applied for specifying requirements for data collection and analysis, considering a key stakeholder in the *ELEVATE-Covid19* project, namely PAT, and its main goals *New rules communicated to Citizens* and *Incoming calls reduced*. These two goals motivated the development of the *ELEVATE-Covid19* software application with a main requirement-goal *RG1: COVID19-rules learnt via IMM exercise*, as depicted in the excerpt of the requirements model in Fig. 7.

The GQM-based approach is applied to validate the requirements goal *RG*1, thus defining the evaluation goal *EG1* that can be stated as illustrated in Table 4.

The GQM for the evaluation goal EG1 is depicted in Fig. 8. The questions *Q1: Are the citizens aware of the tool and interested in it?*, *Q2: Do citizens find the tool useful?* and *Q3: Do the citizens succeed in finding the information they need?* have been associated with EG1. Corresponding identified metrics are the following:

– *M1*, which counts the number of citizens who access an *IMM* exercise, in a given time period. The indicator associated with this metric, *I1*, is defined as % of accesses to the *IMM* exercises over all the accesses to the COVID-19 application, including search by keywords;

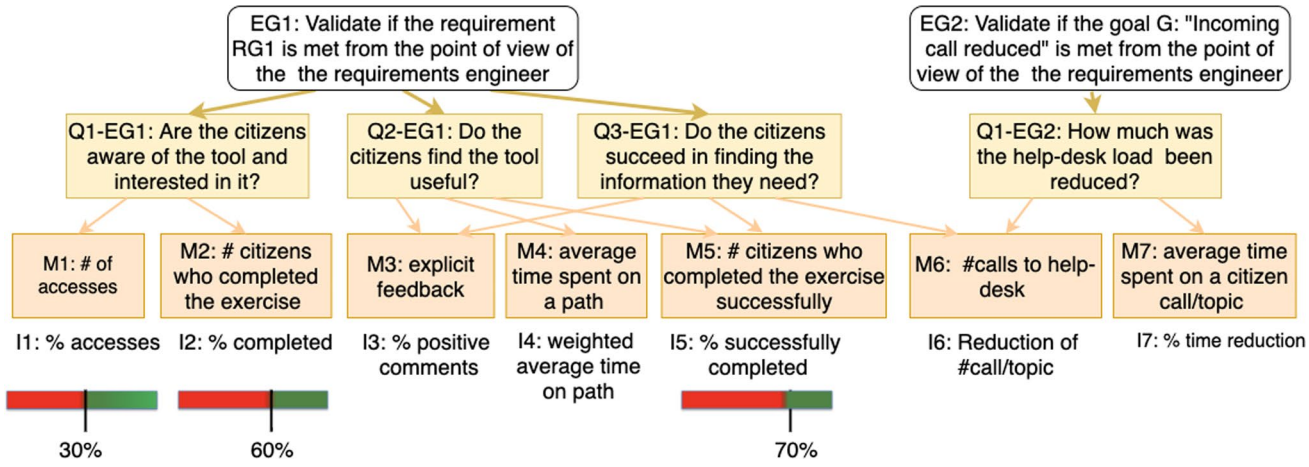


Fig. 8 GQM analysis for online feedback requirements specification (Industrial evaluation 1)

Table 4 The evaluation goal EG1 (Industrial evaluation 1)

[Description]	*EG1: Validate if the requirement RG1: COVID19-rules learnt via IMM-exercise meets citizen's needs*
[Purpose]	Requirements validation
[Perspective]	Requirements engineer
[Object]	RG1: COVID19-rules learnt via IMM-exercise meets citizen's needs
[Issue]	Requirement validity

– *M2*, which counts the number of citizens who completed the accessed exercise. The indicator associated with this metric, *I2*, is defined as the % of exercises that have been completed.

Note that each metric contributes to answering the associated question; being able to collect data on both can help to have a more complete answer. Indicator levels are defined depending on the specific question and on the project's maturity. In the considered case, 55% of accesses or more is regarded a positive answer to *Q*1, while for the second metric, more than 60% exercise completed can reinforce the positive answer to *Q*1.

Two other questions have been associated with *EG*1, namely *Q2: Do the citizens find the tool useful?* and *Q3: Do the citizens succeed in finding the information they need?*. Three metrics have been identified to answer *Q2*, including *M*3 that corresponds to a requirement for explicit feedback, such as textual user reviews or emoticons. *M*4 (average time spent on an exercise path, from start to end node), and *M*5 (number of citizens who completed the exercise reaching a successful node, i.e. a path that transverse correct decisions). Concerning *Q3*, the metric *M*6, measuring the load of the PAT helpdesk, has been identified in addition to *M*3 and *M*5.

Results from the metrics resting on the logs already available in the ELEVATE platform are discussed in Sect. 5.3.1.

Table 5 The evaluation goal EG2 (industrial evaluation 1)

[Description]	*EG2: Validate if the goal "Incoming calls reduced" is met thanks to the ELEVATE-COVID19 web-app*
[Purpose]	Requirements validation
[Perspective]	Requirements engineer
[Object]	G: "Incoming call reduced"
[Issue]	Goal satisfaction

GQM has been applied also in relation to the goal *Incoming calls reduced* of the stakeholder *PAT*'s communication department. This evaluation goal *EG2* has been defined as in Table 5.

The question derived from EG2 is *How much has the help-desk load been reduced*, and to answer it, in addition to M6, the metric *M7: average time spent on a citizen call/topic* was identified. Both metrics would be measured on the PAT help-desk service; data are not available for our analysis.

UF requirements validation For the purposes of this paper, we consider *IMM* exercises that were created for disseminating COVID-19 rules. They concerned the following topics:

– *Sports and outdoor activities*. In short, sports could only be carried out individually and, for non-professional ath-

Table 6 Values of metrics considered for the EG1 validation, Q1

Exercise name	Time window	*M*1	*M*2	*I*2
Sports and outdoor activities	07/05–18/05	608	412	67.76%
Use of protection equipment	07/05–18/05	243	151	62.14%
Travelling (first version)	07/05–13/05	2139	1291	60.36%
Travelling (second version)	13/05–15/05	254	159	62.60%
Travelling (third version)	15/05–18/05	356	249	69.94%

letes, only outdoors. Moreover, sporting activity outside the region was not allowed, and the use of a mask was mandatory during sports activities only in the presence of other people.
- *Use of protection equipment*. Mandatory use of the mask throughout the region both outdoors and in closed places accessible to the public.
- *Travelling*. Travel within the region was only allowed for reasons of health, work, necessity or visits to relatives.

These exercises were published when the accumulation of new norms and the loosening of previous ones created a sometimes confusing situation. Specifically, the key concepts of the three exercises were extrapolated from an ordinance, issued by the local government, which came into force on May 4, 2020. Their availability was advertised by means of press releases to local media on the day they were put online.

Usage data were collected up to May 18, the day on which a further ordinance entered into force which led to a substantial easing of the restrictions in effect up to that moment. We focus on the online user feedback specifications defined a posteriori through the GQM model of the evaluation goal EG1 (Fig. 8).

Table 6 shows the number of citizens who have accessed each exercise (metric *M*1), and the number of citizens who have completed it (metric *M*2). Table 7 reports the average time (in seconds) spent by citizen on an exercise (*M*4). This average is computed on path of different lengths, e.g. path that can lead to an error end-node in two action/choices, and paths that can include four or five action/choices leading to a successful end-node. Moreover, the number of citizens who have completed an exercise with a positive outcome (*M*5), and the percentage of successfully completed exercises (indicator *I*5) are reported.

According to the metrics *M*1, *M*2, *M*5, and corresponding indicators and indicator-levels, question *Q*1 and *Q*2 associated with the evaluation goal *EG*1 seem to be well satisfied. The application of GQM allowed to specify a requirement for collecting explicit user feedback (motivated by *M*3) and other requirements for implicit feedback in terms of new data to be logged, i.e. exercise path length completed by a citizen to be combined with the time spent for completing the exercise when computing the indicator *I*4.

Table 7 Values of metrics considered for the EG1 validation, Q2 and Q3

Exercise name	Time window	*M*4 (seconds)	*M*5	*I*5
Sports and outdoor activities	07/05–18/05	19.4	340	83%
Use of protection equipment	07/05–18/05	17.3	118	78%
Travelling (first version)	07/05–13/05	24.6	1228	95%
Travelling (second version)	13/05–15/05	25.4	145	91%
Travelling (third version)	15/05–18/05	24.1	211	85%

5.3.2 Industrial evaluation 2

ELEVATE-*Tu Sei* focused on the creation of emergency training exercises. The project was performed in collaboration with a company specialised in Emergency Management and Training (EMT), SEA Srl (Trento, Italy), and a high school class, 4A of Liceo Scientifico Galileo Galilei (Trento), participating as extra-curricular activity during the 2020/2021 academic year. For ELEVATE, the purpose of the project was twofold: demonstrating the usability of the development tool for non-specialists; and, creating examples in an application domain of great interest. SEA wanted to study the feasibility of moving part of their training offering to an online platform.

To this end, one of SEA's training expert selected two scenarios requiring fire management and people evacuation from a scholastic building. He created two storyboards, in the form of a set of possible actions. The correctness or erroneousness of selecting an action was represented as a score (positive or negative), following a gamified approach; some led to either positive or negative endings (i.e. properly handled situations vs disasters with victims). The trainees should play roles with specific responsibilities in case of emergencies (janitors, lab technicians) and choose the proper actions in the right sequence.

Transformation of these storyboards into ELEVATE exercises was left to the high school students. This implied an analysis to extrapolate training objectives and the unfolding of the steps into a graph of situations and choices, whose traversal progressively allows the achievement of objectives until a final node is reached, possibly a failure one in case of errors. For instance, a situation concerned a fire alarm; the trainee had to identify its origin (a lab, in this case) by looking at the alarm board, check the room on fire for victims, properly handle windows to disperse smoke, make sure that people went to safety areas and finally call firefighters. Once

Table 8 Stakeholders and requirement goals

ID	Stakeholder	Requirement type	Requirement description
RG1	Teacher	Learning objective	The procedure is remembered by the trainee
RG2	Teacher	Learning objective	The trainee is exposed to the consequences of errors at the right level for one's training stage
RG3	Instructional designer	Quality goal	Creating exercises of the right complexity
RG4	Instructional designer	Quality goal	Submitting exercises to students of the appropriate level

Table 9 The EG goals for the *Tu Sei* industrial evaluation

[Description]	*EG1: Validate if the requirement [RG1: the procedure learnt via IMM-exercise is remembered by the trainee] is met*
[Purpose]	Learning objective assessment
[Perspective]	teacher
[Object]	RG1: the procedure learnt via IMM-exercise is remembered by the trainee
[Issue]	Learning objective achievement
[Description]	*EG2: Validate if the requirement goal [RG2: The trainee is exposed to the consequences of errors at the right level for one's training stage] is met*
[Purpose]	Learning objective assessment
[Perspective]	teacher
[Object]	The trainee is exposed to the consequences of errors at the right level for one's training stage
[Issue]	Learning objective achievement
[Description]	*EG3: Validate if the requirement goal [RG3: Creating exercises of the right complexity] is met*
[Purpose]	Quality assessment
[Perspective]	Instructional designer
[Object]	RG3: Creating exercises of the right complexity
[Issue]	quality evaluation
[Description]	*EG4: Validate if the requirement goal [RG4: Submitting exercises to students of the appropriate level] is met*
[Purpose]	Quality assessment
[Perspective]	Instructional designer
[Object]	RG4: Submitting exercises to students of the appropriate level
[Issue]	Quality evaluation

the exercise structure was built, students recorded video fragments in their own school.

UF Requirements Specification: As mentioned, *Tu Sei* was a pilot study whose design was done by a domain expert (SEA) and its implementation left to school students. That notwithstanding, the discussions between the ELEVATE development team and the domain expert led to the identification of a number of stakeholders and requirements common to most expected uses of ELEVATE as e-learning tool. Table 8 summarises a few requirements goals relevant to this paper. On one side, the *Teacher* or domain expert is responsible for the type and quality of the content of a set of exercises and aims at the achievement of relevant learning objectives by the trainees. On the other side, the *Instructional designer*, who is responsible for the creation of an *IMM* exercise, wants to ensure the exercises are of good quality for the targeted trainees.

Note that ELEVATE is instrumented for collecting the raw data required to compute a few basic metrics, listed later in Table 10: for instance, the number of nodes reached by a trainee ($BM1$), as well as the average of $BM1$ computed on a group of trainees (called $BM1_{ave}$) or the number of failure nodes visited by a trainee before reaching a successful end-nodes ($BM3$).

The GQM method was applied to define a set of evaluation goals related to the requirements listed in Table 8. The resulting goals are reported in Table 9.

In Fig. 9 we illustrate two evaluation goals with the derived questions and associated metrics, which are defined in terms of basic metrics. For instance, metric $M1$ for $Q1$ is defined as $(BM1(time1) - BM1(time2))$, with $time2 > time1$.

Learning: The practitioners believe that implicit user feedback is a powerful resource for validating project's goal and software application requirements, especially for knowledge dissemination applications. In the *ELEVATE-Covid19* project, five *UF* requirements where identified for the purpose of evaluating if the requirements goal *RG1:COVID-19 rules learnt via IMM-exercise* has been achieved, and two for the purpose of evaluating if an important stakeholder's goal,

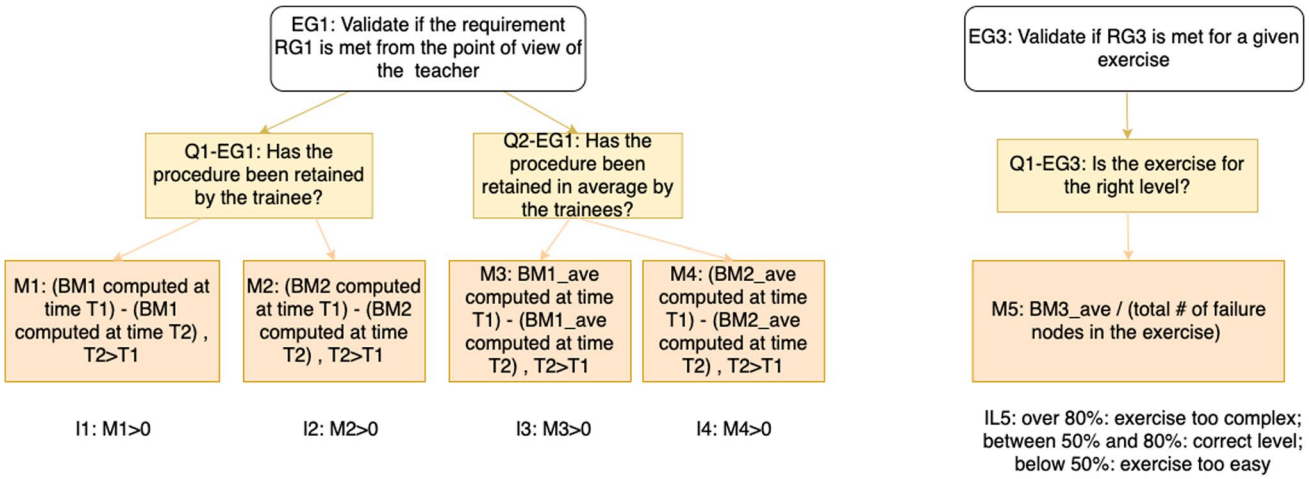


Fig. 9 GQM analysis for online user feedback specification (Industrial evaluation 2)

Table 10 GQM for online feedback requirements specification. Basic metrics

Metric Id	Metric
$BM1$	Number of failure nodes reached by a trainee
$BM1_{ave}$	Average of BM1 on a group of trainees
$BM2$	Time required to reach a success node by a trainee
$BM2_{ave}$	Average of BM2 on a group of trainees
$BM3$	Number of failure nodes reached by a trainee before an exercise is successfully completed
$BM3_{ave}$	Average of BM3 on a group of trainees
$BM3'_{ave}$	Average of BM3 computed on a homogeneous group of trainees

namely PAT's goal *Incoming calls reduced*, were achieved. Notice that only three of the seven *UF* requirements were implemented thanks to the basic configuration of the ELEVATE platform. The application of the *GO+GQM* method to the *ELEVATE-Covid19* project provided evidence: (i) about the importance of considering *UF* requirements specification at design time; and (ii) about the effectiveness of the *GO+GQM* method. In case of procedure training applications, practitioners report that often standard questionnaires for trainees are used. These questionnaires can be considered a type of explicit user feedback. For such type of applications, implicit user feedback can be important to test if there are bugs in the deployed exercise, or to get a qualitative assessment on how the exercise was performed by the students. The *Tu Sei* project team identified four key requirements goals, and the Delta Informatica team members were able to apply *GO+GQM* to identify *UF* requirements aiming at providing evidences of the achievements of these goals. Practitioners learnt how to associate metrics and indicator to address questions associated with evaluation goals. **Input to next cycle**: results from the application of the method to the two selected projects encouraged researcher to perform a Quality-in-Use evaluation.

5.4 Cycle 4

In this cycle we aim at evaluating the effectiveness, efficiency and acceptance by practitioners of the *GO+GQM* method. We referred to a *quality-in-use* assessment model [30], which is shown in Fig. 6. Effectiveness is assessed by looking at the number of *UF* requirements that were identified in the two projects evaluated in cycle 3. This gives an account of the degree of coverage of the questions derived from the evaluation goals (output of the GQM step of the proposed approach).

In order to assess efficiency and acceptance of the *GO+GQM* method as perceived by practitioners, we designed a survey based on two questionnaires, called $Questionnaire_1$ and $Questionnaire_2$, respectively, and a short videotutorial on the use of the *GO+GQM* method. During the execution of the survey, the videotutorial was proposed to the survey participants before $Questionnaire_2$ with the aim to reinforce their knowledge on the method. $Questionnaire_1$ is made of three main sections. The first contains questions for characterising the subjects in terms of the roles they played in the software development projects, years of experience, number of users of the software products realised in their projects. The second and third sections focus on explicit and implicit online user feedback, and propose questions related to RQ1, RQ2, and RQ3.

$Questionnaire_2$ is motivated by RQ4. It proposes questions aiming at evaluating developers' perceived efficiency of the method, and acceptance. An excerpt is shown in Fig. 6, right side.

Evaluating and Learning[3]

Following the adopted Quality-in-use model described in Sect. 4, we discuss evaluations of the *Effectiveness* of method upon its application to the two case studies, while an evaluation of its *Efficiency* and *Acceptance* is derived from the answers of six respondents to $Questionnaire_2$.

5.4.1 Effectiveness

Applying the *GO+GQM* method in the first industrial evaluation allowed to identify questions and related metrics that were not considered at the time the *ELEVATE-Covid19* web-application was designed. In the example depicted in Fig. 8 two of the four additional metrics that were identified to answer questions associated with two evaluation goals correspond to new *UF* requirements. Moreover, the resulting GQM model makes more clear the fact that the implemented logs can contribute only partially to the assessment of the evaluation goal.

The application of the method in the second industrial evaluation helped to specify metrics which build on basic log mechanisms already implemented in the ELEVATE platform (basic metrics in Table 10), which can help address four different evaluation goals, from the perspective of two different key stakeholders of the project.

5.4.2 Efficiency

Among the six respondents of the $Questionnaire_2$, five are members of the ELEVATE team, but only two applied the method to the two case studies described in this paper. The sixth respondent is employed in another company. All watched a short video tutorial on the usage of the method and were then asked to perform an exercise before filling in the questionnaire. The exercise refers to the illustrative example presented in Sect. 3. Specifically, the exercise asks a subject to focus on the requirements goal *Home energy plan advice got* and to define *UF* requirements that could help validate if this goal requirement meets users' needs through the analysis of the collected online user feedback. The subjects are asked to derive at least two questions, and at least one metric for one of the questions. All the respondents performed the exercise properly, dedicating in average 15 minutes (minimum time is 5 minutes, maximum is 22 minutes). Regarding the perceived effectiveness (question 6 and 7 in $Questionnaire_2$ depicted in Fig. 6), four respondents agree that the method allows to specify what user feedback to collect more effectively (than without it), one selected the neutral answer, and one disagreed. Three of the respondents agree and the other three disagree on the statement that the *GO+GQM* method allows to specify what user feedback to collect more quickly than without it.

5.4.3 Acceptance

Perceived usefulness, satisfaction and trust about the method are collected by asking subjects to answer the first five questions of $Questionnaire_2$ on a 5-likert scale, and two open questions.

All the six respondents find helpful to have a method that guide them to specify which user feedback to consider. The main motivation for this concerns saving time due to having a predefined feedback collection model rather than creating it from scratch for every project.

Four respondents agree that it's clear how and for what purpose to use the proposed method, one response is neutral and one negative.

Concerning the question about adopting the *GO+GQM* method, two respondents answer positively, two are neutral and two negative. Explanations given for the negative answers include the lack of clear guidelines and the risk to work on the structure of the model instead of "focusing on the target of the feedback itself". Concerning the perceived strengths of the proposed method, a major point was related to the capacity of the approach to structure the design of user feedback gathering methods by "favouring reasoning and formalisation" of the feedback and by the capacity of the method to "clarify the relationship between project's objectives and which feedback to collect, if any" so giving the analysts the possibility to "focus better only on feedback that is really useful to us". On the other side, the weaknesses are mostly related to the need of becoming expert in the use of the approach, so requiring a learning curve to be used in an efficient way. Related to this point some of the subjects required "a good set of templates and examples for becoming proficient at its use" and "guidelines on how to formalise the user feedback" to avoid "leading to possibly useless metrics and overhead on trivial issues". There is also an observation on the fact that the method is too structured (risk of focusing on the building of the model itself rather than on the concrete identification of the measures) and can slow down the capacity to quickly identify questions and measures. It is also important to stress the difference between question and metric and their use in the method, maybe by giving concrete examples of use in different domains and contexts. A final comment concerns the indicators and the difficulty to sometimes "define the indicator level" at the time of the indicator design.

All these observations are important to refine the proposed method and to also plan for the set up of learning material to decrease the learning curve for an efficient exploitation of the approach.

[3] The link to the data repository containing results of the questionnaires is given in [31].

6 Discussion on findings and limitations

Main findings from the post-mortem analysis of this *ELEVATE-Covid19* project, combined with the analysis of the responses to $Questionnaire_1$ can be summarised along our research questions as follows.

6.1 RQ1: Is online user feedback collected by practitioners? What type?

Referring to the responses in the questionnaire that subjects give in relation to the projects they have been involved, the explicit user feedback is collected in several cases. Also implicit user feedback is collected, as stated by 4 of the subjects.

Concerning the specific *ELEVATE-Covid19* project, there was no plan to collect explicit user feedback. The EMT component of the ELEVATE platform automatically collects very fine grained usage data according to its basic configuration for online training applications.

6.2 RQ2: When is it decided what user feedback to collect? For what purpose?

From the responses it appears that in general there are two main phases where the decision about the use of user feedback is performed: during the early requirements analysis and, in the case of explicit feedback, during the deployment of the system. The purpose in both cases is mainly related to bug fixing and requirements elicitation, while, for the specific case of the explicit feedback, the purpose is also to collect statistics related to user acceptance.

In the specific case of ELEVATE, it was assumed that the available statistics were enough for the purposes of the *ELEVATE-Covid19* information tasks. Indeed, they answered a number of important questions, including the number of distinguished users and how the exercises were used. Satisfaction was not considered relevant for e-learning nor at the time of adoption of the ELEVATE platform for the COVID-19 rule dissemination. It is worth noting that, as an outcome of this experience, a user satisfaction poll is now automatically performed by the engine at the end of any type of exercise.

6.3 RQ3: How is it decided what user feedback to collect and analyse? Is a systematic approach needed?

The responses indicate that there are different aspects to consider depending if we refer to explicit or implicit user feedback. Considering the explicit one it seems that the decision is performed in brainstorming sessions internal to the team that also consider the needs of the stakeholders and the objectives of the project. Considering the implicit one, it seems that the decision is based on the criticality of some functionalities, in order to strictly monitoring them, and on the need to follow the behaviour of the users to perform verification activities such as field testing. Finally, there was a general agreement about the need of a systematic method for planning user feedback elicitation. This finding aligns to results of a contemporary survey presented in [5] that concerns the use of app store. That study reveals that developers use metrics made available by the app store platform to measure the success of an app's release, for instance counting the number of download or looking at user reviews. In our study, we focused on the practitioners' interest to assess the success of specific feature of an app, which were designed to satisfy higher level requirement goals.

Focusing on the specific ELEVATE project, some of this feedback is very specific to the information being delivered; in the *ELEVATE-Covid19* case, a critically missing data is if the users have found the information they were looking for. Some feedback could be collected as part of the exercises themselves (e.g. for *ELEVATE-Covid19*, this could be a choice to be followed if the user desires more data on a specific topic), so it has to be taken into account at exercise design time. Without a systematic approach, the risk of forgetting to collect important feedback is very high.

6.4 RQ4: How effective and efficient is the *GO+GQM* method, as perceived by practitioners? Will practitioners adopt this method?

The application of the *GO+GQM* method to the *ELEVATE-Covid19* project in cycle 3 of the action research study provided evidence about its effectiveness and the importance to use it at design time. For example, as shown in Fig. 8, to evaluate if the requirement goal *RG1: COVID-19 rules learnt via IMM-exercise* is achieved, the application of the *GO+GQM* method resulted into the identification of 5 metric-indicators. Among these 5, only 3 correspond to log data that could have been collected using the basic configuration of the ELEVATE platform (namely those corresponding to $M1-I1$, $M2-I2$, and $M5-I5$ in Fig. 8). We discussed results from the survey in a brainstorming meeting we organised at conclusion of cycle 4 of our action research, where two researchers (SR1, SR2) and the senior consultant (SC) were involved. Main conclusions the meeting participants agreed on are summarised here below. Overall, lessons learnt by the ELEVATE team in the *ELEVATE-Covid19* project increased their awareness on the usefulness of online user feedback for requirements validation purposes. The practitioners' opinions collected with our survey about the adoption of the method deserved some reflection. First, it is important to clarify that the practitioners involved in the

action research study tend to work in small and flat teams of 2 or 3 members to implement relatively small projects. It is believed that in the case of more structured teams with roles looking after requirements, such as the product owner, having the opportunity to exploit online user feedback (both implicit and explicit) to assess the satisfaction of strategic goals of the project or, similarly, of users' goals that motivate user stories will be of greater interest. Further, the current lack of integration of the *GO+GQM* method into a tool-supported process, such as a SCRUM development process supported by an issue tracking system to manage the product backlog, could be seen as a barrier towards adopting the method.

6.5 Threats to validity

Conducting an action research study can be prone to several threats to validity [7, 8]. We took them into account during the design of the study as well as when analysing findings, as discussed here below.

6.5.1 Construct validity

The main construct validity threat which we identified concerns the fact that we mostly worked with one team during the first two cycles of the action research (called mono-operational bias in [8]). We tried to mitigate this threat to validity in the following cycle, by selecting a different company's project (ELEVATE-*Tu Sei*), which was developed by a different team.

6.5.2 Internal validity

Concerning internal validity, we identified as main threat the bias in selecting subjects involved in the planned actions. In particular for the action of the last cycle, based on a survey, we tried to mitigate this threat to validity by involving practitioners outside the company, and by letting subjects to fill in the survey in anonymity.

6.5.3 Conclusion validity

Our study is mostly qualitative. When taking measurements, we paid attention to threats of low reliability of the measures them self. The application of the *GO+GQM* in the two industrial evaluations was double-checked by co-authors not directly involved in the application of the method. Concerning the survey, before administering it to the subjects we performed a pilot study aiming at assessing its quality, and in particular the clarity of the questions.

6.5.4 External validity

Generalisability is an important threat to validity when performing action research. Indeed we worked with one company and in projects using the same software platform for developing interactive video-based exercises (the ELEVATE platform). We tried to mitigate this threat by choosing two projects that involve different stakeholders (i.e. different requirement-goals) and different application domains, namely mass communication (*ELEVATE-Covid19* project) and emergency training exercises (ELEVATE-*Tu Sei* project).

7 Conclusion

This paper has presented our research on how to specify requirements for online user feedback collection and analysis for requirements validation and evolution in the context of data-driven requirements engineering. We have proposed the *GO+GQM* method for the specification of such requirements at design time. The *GO+GQM* method is based on Goal-Question-Metric [11] and goal-oriented modelling (e.g. [15]).

The *GO+GQM* method is part of the output of the 4-cycle action research [8] that we presented in the paper. In the context of this study, we assessed our approach on two industrial projects, one concerning a citizen information service for the COVID-19 pandemic regulations in an Italian region (the *ELEVATE-Covid19* project), the other one a training system for emergency situations (ELEVATE-*Tu Sei* project). The results of these evaluations seem to be promising in terms of perception of the involved practitioners and have already influenced the development plans of the industrial partner involved in the studies.

As future work, we will further assess the proposed method by applying it to projects in different domains in order to verify its general applicability. We plan to perform experiments involving practitioners who will be asked to specify user feedback management requirements with the *GO+GQM* method, considering subjective metrics, such as perceived usefulness, as well as objective metrics, such as the degree of coverage of the questions derived from the evaluation goals associated with requirements. Moreover, administering the *GO+GQM* tutorial and the survey to a larger population of practitioners will help to obtain a subject sample that is statistically representative, and allow to improve generalisability of results.

8 Supplementary information

The data generated with the survey conducted in this study are available at [31].

Acknowledgements This work is part of the ELEVATE research project, which is funded by Provincia Autonoma di Trento, L.P. 6/1999. We would like to thank the anonymous reviewers for their invaluable help to improve the presentation of this work.

Data availability statement The datasets generated in the survey that has been performed in the current study are available at the permanent repository, https://figshare.com/s/b6c48252ce301613242c. A has been added at the end of the Conclusion section ("Supplementary Information" paragraph) and the link to the repository has been included as new reference.

Declarations

Conflict of interest No Conflict-of-interest.

References

1. Maalej W, Nayebi M, Johann T, Ruhe G (2015) Toward data-driven requirements engineering. IEEE Softw 33(1):48–54
2. Groen EC, Seyff N, Ali R, Dalpiaz F, Dörr J, Guzman E, Hosseini M, Marco J, Oriol M, Perini A, Stade MJC (2017) The crowd in requirements engineering: the landscape and challenges. IEEE Softw 34(2):44–52
3. Perini A (2018) Data-driven requirements engineering. the SUPERSEDE way. In: SIMBig 2018, Lima, Peru, 2018. pp 13–18. Springer
4. Franch X, Seyff N, Oriol M, Fricker S, Groher I, Vierhauser M, Wimmer M (2020) Towards integrating data-driven requirements engineering into the software development process: a vision paper. In: REFSQ. pp 135–142. Springer
5. Al-Subaihin AA, Sarro F, Black S, Capra L, Harman M (2021) App store effects on software engineering practices. IEEE Trans Software Eng 47(2):300–319. https://doi.org/10.1109/TSE.2019.2891715
6. Morales-Ramirez I, Perini A, Guizzardi RSS (2015) An ontology of online user feedback in software engineering. Appl Ontol 10(3–4):297–330
7. Wohlin C, Runeson P, Höst M, Ohlsson MC, Regnell B, Wesslén A (2012) Experimentation in software engineering. Springer Science & Business Media
8. Staron M (2019) Action research in software engineering: Metrics' research perspective (invited talk). In: Catania B, Královič R, Nawrocki J, Pighizzini G (eds) SOFSEM 2019: theory and practice of computer science. Springer International Publishing, Cham, pp 39–49
9. Horkoff J, Barone D, Jiang L, Yu E, Amyot D, Borgida A, Mylopoulos J (2014) Strategic business modeling: representation and reasoning. Softw Syst Model 13(3):1015–1041
10. Barone D, Jiang L, Amyot D, Mylopoulos J (2011) Reasoning with key performance indicators. In: IFIP working conference on the practice of enterprise modeling. pp 82–96. Springer
11. Basili VR (1994) Goal question metric paradigm. Encyclopedia of software engineering pp 528–532
12. ISO/IEC 25010:2011 (2011) systems and software engineering-systems and software quality requirements and evaluation (square)-system and software quality models
13. Astegher M, Busetta P, Perini A, Susi A (2021) Specifying requirements for data collection and analysis in data-driven RE. A research preview. In: Dalpiaz, F., Spoletini, P. (eds.) Requirements Engineering: Foundation for Software Quality—27th International Working Conference, REFSQ 2021, Essen, Germany, April 12-15, 2021, Proceedings. Lecture Notes in Computer Science, vol 12685, pp 182–188. Springer. https://doi.org/10.1007/978-3-030-73128-1_13
14. Basili VR, Trendowicz A, Kowalczyk M, Heidrich J, Seaman CB, Münch J, Rombach HD (2014) Aligning Organizations Through Measurement—The GQM+Strategies Approach. Springer, The Fraunhofer IESE Series on Software and Systems Engineering
15. Susi A, Perini A, Mylopoulos J, Giorgini P (2005) The tropos metamodel and its use. Informatica (Slovenia) 29(4):401–408
16. Jeon H (2015) Personalized information gathering using implicit user feedback in a multiple personal device environment. In: Computer Science and its Applications, pp 425–435. Springer
17. Lallé S, Conati C (2019) The role of user differences in customization: a case study in personalization for infovis-based content. In: Proceedings of the 24th International conference on intelligent user interfaces. pp. 329–339
18. Shi X, Gu Z, Chang D, Huang L (2015) How do the users show their interest on line? Eye movement and browsing behaviour. Int J Multimed Ubiquitous Eng 10(4):43–52. https://doi.org/10.14257/ijmue.2015.10.4.05
19. Van Der Aalst WM, Reijers HA, Weijters AJ, van Dongen BF, De Medeiros AA, Song M, Verbeek H (2007) Business process mining: an industrial application. Inf Syst 32(5):713–732
20. Rebuge Á, Ferreira DR (2012) Business process analysis in healthcare environments: a methodology based on process mining. Inf Syst 37(2):99–116
21. Maggi FM, Francescomarino CD, Dumas M, Ghidini C (2014) Predictive monitoring of business processes. In: Jarke, M., Mylopoulos, J., Quix, C., Rolland, C., Manolopoulos, Y., Mouratidis, H., Horkoff, J. (eds.) Advanced Information Systems Engineering - 26th International Conference, CAiSE 2014, Thessaloniki, Greece, June 16-20, 2014. In: Proceedings. Lecture Notes in Computer Science, vol 8484, pp 457–472. Springer. https://doi.org/10.1007/978-3-319-07881-6_31
22. Adamo G, Di Francescomarino C, Ghidini C (2020) Digging into business process meta-models: a first ontological analysis. In: International conference on advanced information systems engineering. pp 384–400. Springer
23. Martin WJ, Sarro F, Jia Y, Zhang Y, Harman M (2017) A survey of app store analysis for software engineering. IEEE Trans Softw Eng 43(9):817–847. https://doi.org/10.1109/TSE.2016.2630689
24. Dabrowski J, Letier E, Perini A, Susi A (2022) Correction to: analysing app reviews for software engineering: a systematic literature review. Empir Softw Eng 27(2):58. https://doi.org/10.1007/s10664-022-10135-4
25. Oriol M, Stade MJC, Fotrousi F, Nadal S, Varga J, Seyff N, Abelló A, Franch X, Marco J, Schmidt O (2018) FAME: supporting continuous requirements elicitation by combining user feedback and monitoring. In: 26th IEEE RE 2018, Banff, AB, Canada, 2018. pp 217–227
26. Lopez FS, Condori-Fernández N, Catalá A (2020) Understanding implicit user feedback from multisensorial and physiological data: A case study. In: ICSE '20: 42nd International conference on software engineering, Workshops, Seoul, Republic of Korea, 27 June–19 July, 2020. pp. 563–569. ACM. https://doi.org/10.1145/3387940.3391466

27. Kifetew FM, Perini A, Susi A, Siena A, Muñante D, Morales-Ramirez I (2021) Automating user-feedback driven requirements prioritization. Inf Softw Technol 138:106635. https://doi.org/10.1016/j.infsof.2021.106635
28. Bresciani P, Perini A, Giorgini P, Giunchiglia F, Mylopoulos J (2004) Tropos: an agent-oriented software development methodology. Auton Agents Multi Agent Syst 8(3):203–236. https://doi.org/10.1023/B:AGNT.0000018806.20944.ef
29. Dellagiacoma D, Busetta P, Gabbasov A, Perini A, Susi A (2020) Authoring Interactive Videos for e-Learning: the ELEVATE Tool Suite. In: MIS4TEL'20. pp. 128–136. Springer
30. ISO/IEC 25000:2014 (2014) Systems and Software Engineering—Systems and Software Quality Requirements and Evaluation (SQuaRE)—Guide to SQuaRE
31. Astegher M, Busetta P, Gabbasov A, Perini A, Susi A (2021) Specifying requirements for collection and analysis of online user feedback—results of the questionnaires (figshare) https://figshare.com/s/b6c48252ce301613242c