International Journal of Electrical and Computer Engineering (IJECE)
Vol. 12, No. 6, December 2022, pp. 6472~6485
ISSN: 2088-8708, DOI: 10.11591/ijece.v12i6.pp6472-6485

Communication, culture, competency, and stakeholder that contribute to requirement elicitation effectiveness

Ajchareeya Chaipunyathat, Nalinpat Bhumpenpein
Department of Information Technology, Faculty of Information Technology and Digital Innovation, King Mongkut's University of Technology North Bangkok, Bangkok, Thailand

Article Info

Article history:

Received Aug 29, 2022
Revised Jun 14, 2022
Accepted Jul 9, 2022

Keywords:

Requirement elicitation
Requirement engineering
Software development
Software engineering

ABSTRACT

In the context of software development, requirement engineering is one of the crucial phases that leads to software project success or failure. According to several disruptive changes in the software engineering landscape as well as the world's challenge of virus pandemic, the provision of practical and innovative software applications is required. Therefore, issues resolution in requirement elicitation is potentially one of the key success factors resulting in enhanced quality of system requirement. The authors have striven to create new ways of requirement elicitation according to factor effects of communication, culture, competency, and stakeholder, by incorporating tools, processes, methods, and techniques to solve the problems comprehensively, and then proposed an adaptive and applicable conceptual framework. To illustrate these effects, the authors performed a literature review from the past 8 years, and then data analysis from interviews of 27 practitioners, observations and focus groups of software development in real-life projects.

Corresponding Author:

Nalinpat Bhumpenpein
Department of Information Technology, Faculty of Information Technology and Digital Innovation, King Mongkut's University of Technology North Bangkok
1518 Pracharat 1 Road, Wongsawang, Bangsue, Bangkok 10800, Thailand
Email: nalinpat.b@itd.kmutnb.ac.th

1. INTRODUCTION

In the context of software development, eliciting requirement is one of core activities in a requirement engineering (RE) process that includes requirement elicitation, requirement analysis, negotiation, system modelling, requirement specification, requirement validation and requirement management [1]. Effectiveness of RE technology could gain more benefits to software industry over their RE activities including eliciting requirements, modelling and analyzing requirements, communicating requirements, agreeing requirements and evolving requirements because requirement changes from misunderstood or missed requirement in later project life cycle are more expensive to fix [2]. Specifically, according to (1) [3] from that is:

$$sw = t(req) \quad (1)$$

In the equation, *sw* represents software, *t* represents team, and *req* represents requirements. It can be interpreted that software is an output of the correlated input of how team process and interpret requirements. Team is the software implementation team who are both the sender and the receiver of requirement communication, e.g., requirement engineer or business analyst, software architecture, software developer, and software quality control. Furthermore, the requirement engineer is the first person and the overall contact

point to bridge gaps between user requirements and software development. Although there are several methods for achieving quality RE effectiveness, it still requires new blueprints to proactively apply advanced RE research to digital industry practices. The study in [4] presented three main causes of distance between RE research and industry that are i) lack of tools, ii) lack of technical support, and iii) slow development pace. These three causes possibly lead to future direction in requirement elicitation research applied to companies such as best practice training provided. Besides, the study by [5] stated the opinions of the insufficient collaboration between academia and industry that root to an unnecessary research and a difficulty to apply RE research in practice. However, there also was the positive opinion on research works perceived by practitioners on problem relevance and solution utility. In addition, the study of [6] catered five future trends of RE research study from 2017-2042 on an automation and its impact that are i) humanoid requirement elicitation, ii) data-driven requirement evaluation, iii) automated documentation, iv) computation-enabled quality assurance, and v) machine-assisted evolution, which affirmed that humans still drive relationship in the future of software engineering. According to human on RE, there are four important philosophical elements that concerns i) interpreting and understanding stakeholders' terminology, concepts, viewpoints, and goals, ii) epistemology-understanding of beliefs of stakeholders, iii) phenomenology in terms of the question being observable in the world, and iv) the question of what can be agreed on as objectively true [2].

However, RE for information system is different from RE for embedded control systems and different from RE for generic services, e.g., networking and operating systems which mean that the type of software will also affect the type of RE methods, as in an accordance with [7] it can be pointed out an overview of elicitation techniques and its purpose related to requirements gathering, design and idea-generating as well as thinking tools. In terms of an RE process, requirement elicitation is the first task to be performed. Its goals are to focus on problem domain and need of stakeholders by collecting or capturing requirements, which can simply do by asking the right question to the users and then gathering user's current pain points or what problem needs to be solved based on requirements boundaries. The user plays a central role in an RE process along with other identified stakeholders, which can be individuals, organizations, customers, clients, and any member of software development teams.

Looking into modern software development affecting by disruptive changes of an advance technology and theories of software changes that influence by work context and are decided by people and tools in, in different industry contexts such as government in [8], and by emerging technologies like agility, open source system, internet of things, mobile applications, artificial intelligence, cyber-physical-system, big data and cloud technology [9], [10] plus the new trend of low-code development platform, researchers are required to study innovative ways to resolve RE challenges in such technologies, i.e., crowd RE and user feedback. Moreover, applications become apparently from coronavirus disease (COVID-19) pandemic such as self-screening and self-health check application, Chiang Mai COVID-19 hospital information system [11], and vaccine reservation system. These software requirements required to serve the large number of users plus rigorous concerns on quality attributes, e.g, contact-tracing application in [12], it required to meet everyone's requirement challenges that are Bluetooth functionality, battery consumption, security and privacy, trust and transparency, reliability and effectiveness and user acceptance. They were found that application should has a potential to understand during pandemic. Everyone is matter. This paper is organized into five sections. Next section describes research method. Section 3 presents research results. Section 4 proposes a conceptual framework and how to apply it. At the end, discussion and conclusion is posed in sections 5 and 6.

2. RESEARCH METHOD

In accordance which the study in [2], RE is taking place in the context of human activity system, related to cognitive and social sciences to provide both theoretical grounding and practical techniques for eliciting and modelling requirement. Therefore, this requirement elicitation research groundwork is conducted under multidisciplinary including linguistic, sociology, cognitive psychology, and philosophical elements in the aspects of communication effect, cultural effect, stakeholder effect, and competency of requirement engineer and business analyst. Following by the study that stated nine categories of requirement elicitation issues in [13] and seven factors that affect the activities of the requirements elicitation process in [14], authors then classified them in four areas of problems as shown in Table 1. This research is conducted under the research process flow in Figure 1. It comprises of four steps, i.e. i) theoretical study from a literature review; ii) industry study from real-life software development projects, the data collection and data management from step 1 and step 2 then performed to be an input of step no; iii) which is gap analysis, and finally step; and iv) that is a development of conceptual framework based on the contribution of communication, culture, competency of requirement engineer, and stakeholder for software development requirement elicitation.

Table 1. Requirement elicitation issue and its classification

Source	Requirement elicitation issue/factor	Classification
[13]	There are human aspects of requirement elicitation that preclude simple communication between consultants and clients.	Communication and Competency
	The language of humans is not always suitable for technological solution.	Communication
	Requirements change as the project proceeds	Stakeholder
	Clients will sometimes ask for requirements that the organization does not need.	Stakeholder
	Clients cannot say what the business needs.	Communication
	Some clients do not want to support team members in the project.	Culture
	Requirement elicitation is failed because it is not done properly.	Culture
	Symptoms that are not problems are often reported.	Stakeholder & Competency
	Requirement elicitation is not deterministic.	Culture
[14]	Learning capacity	Stakeholder
	Negotiation capacity	Communication & Competency
	Permanent staff	Stakeholder
	Perceived utility	Stakeholder
	Confidence	Competency
	Stress	Culture
	Semi-autonomous	Culture

2.1. Research question

There are two research questions for this research.

- RQ1: What are the factors in the aspects of communication, culture, competency, and stakeholder effecting on requirement elicitation ineffectiveness/effectiveness?
- RQ2: What are practical guidelines (including framework, model, process, method, technique, and tool) for executing effective requirements elicitation in aspects of communication, culture, competency, and stakeholder?

2.2. Research process

The research process flow in Figure 1 presents the research process study on RQ1 and RQ2 in the four areas that are communication, culture, competency and stakeholders. Researchers aim to obtain the practices in terms of methodologies, framework, models, process, methods, techniques, and tools which are used for executing effective requirement elicitation. Theoretical study and industry study are conducted along the same time, perform data collection and gap analysis are then achieved and lastly develop a conceptual framework.

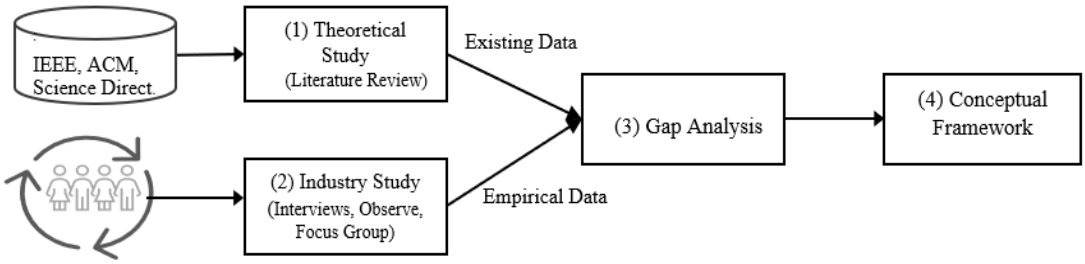

Figure 1. The research process flow

2.3. Literature review

The authors apply a content-analysis process to capture of effectiveness and ineffectiveness factors in a requirement elicitation process including problem definition, conduct requirement elicitation, prototyping and confirm and measurement and document, from the related RE papers published during 2013-2021. The search string included the following keyword: “Requirement”, “Software Development” OR (“elicitation”, “engineering”, “gathering”), “Communication”, “Culture”, “Competency”, “Stakeholder”, “Effectiveness”, “Challenges”, “Factor”. The findings were grouped into four categories (i.e., communication, culture, competency, and stakeholder).

2.4. Industry study

To obtain data for analysis and answers RQ1 and RQ2, the authors used semi-structured interviews and focus group observations from software development practitioners and business users in six projects as

depicted in Tables 2 and 3. The interviews took about 20-60 minutes for each. Most of the data were collected online because of COVID-19 pandemic, during February-September 2021.

Table 2. Overview of investigate practitioners and business users

Organization-business area	Number of practitioner /business user	Based location	Year of experience	Project involved	Interview/observation
A-System Integration	21 -0	Thailand	2-15	P1, P2, P3, P4, P5, P6	Interview
B-Business in Financial Sector	9 -11	Thailand	3-20	P1, P2, P3, P4, P5	Interview and observation
C-Software Product and License Provider	9- 0	Multi-national	6-20	P1, P2, P3, P5	Interview and observation
D-Business in Financial Sector	3-15	Thailand	3-20	P6	Observation

Table 3. Project size

Project	Project size approximately in USD	Number of practitioners involved	Project methodology	Project duration
Project P1	1,000K	40	Agile	12 months
Project P2	360K	20	Hybrid (Agile & Waterfall)	10 months
Project P3	350K	17	Hybrid (Agile & Waterfall)	10 months
Project P4	500K	30	Agile	17 months
Project P5	200K	16	Agile	10 months
Project P6	250K	25	Waterfall	9 months

2.5. Data collection

Data was collected from our three sources that are interview, focus group observation and content-analysis. Common effective influences on communication, culture, competency, and stakeholder were gathered and analyzed from the literatures, and from the real-world software development projects. These activities were conducted, and the data were recorded as described in Table 4.

Table 4. Techniques and process description

Techniques	Process Description
Interviews	Conducting semi-structured interviews, the interviewees aware that the information that they provided would be confidential and made anonymous. The first question is their background the role in the projects that they are involved in, afterward, asked for ways how they elicit, communicate, and the requirements of their system. In the final part of the interviews, the researchers asked the participants about the requirement elicitation problems/challenges and the solution or the best practice that they have been used. Then made note of the result.
Observations	Observing requirement elicitation activities, we made notes from the session, recorded the video and then transcript the result.
Content Analysis	Going through literatures and collecting keywords about "Communication", "Culture", "Competency", and "Stakeholder" factors that effect to Requirement Elicitation and RE. Then noted the results in Mind-Mapping tool.

3. RESULTS

3.1. The findings from literature review

The findings from literature review according to RQ1 are described in four areas. The authors present notations of effectiveness factors by using plus "+" sign, ineffectiveness factors by using minus "-" sign, and factors that differently effect on another context or required further study and more evidence by using "*" sign. Firstly in the communication area, 10 factors were found including articulate-related problem (-), unawareness of needs (-), misunderstanding among stakeholders (-), lack of verbal and presentation skill (-), problem in culture and perspective related (-) [15], language barriers and cultural differences (-) [15], [16] communication challenge in terminology and semantic difference between customer and developer (-) [16], no in-depth communication between requirement analyst and software architect in terms of cost effectiveness (-) [17], difficulty in communication channel and medium articulation between end-users and project team (-) [18], and uneasiness to adopt new communication artifact because organization culture has determined communication artifact for team members to use (-) [18].

Secondly in the culture area, the study on national culture effect in [19] found 16 factors as. First, culture plays important role in choosing RE technique in Kuwait (*). In Saudi Arabia, deference to elderly people and too high priorities (-), autocratic decision making (-), limited trust (-), belief in expertise (-), relationship (+), empathy with client (+), letting the strongest win (-), gender segregation (*), dress code (*), and using English for requirement document (*) were scrutinized. Moreover, (12) people usually avoiding

conflicts and uncertainties (-), (13) leaders' co-operation and top management support being important to RE process (+), and preferred prototyping interview and focus group (+) in China, (15) decision by top management (*) and (16) employee staying away from decision making (requirements that are not backing by top and possibly not to be done) (-) in Thailand were analyzed. All of the above factors can be grouped in three types including generative organization (trust and sharing behavior) (+), bureaucratic organization (rule and process behaviors) (-), and pathological organization (fear and power orient behavior) (-) [20]. In addition, studies on software feedback differences between Germany and China found that individualism (*)/ collectivism (*), power distance (-) and uncertainty avoidance (-) based on Hofstede cultural model [21] has significant differences in behaviors and attitudes to online feedback of these two countries [22].

Thirdly in the competency of requirement engineer area, there are 14 competencies comprising of generic and specific ones [23]. Seven generic competencies include understanding of cultures and customs of other countries (+), capacity to adapt to new situations (+), ability to learn quickly about domain or technology (+), virtual team skills (+), ability to work in an international context (+), appreciation of diversity and multiculturality (+), and living with ambiguity in remote teams. Seven specific competencies include comprehension of global software development (GSD) (+), detection of elicitation sources (+), comprehension of software requirements specification (+), knowledge about analyzing quality requirements (+), capacity to manage changing requirements (+), elicitation of real requirements based on stakeholder's need using interview technique and computer mediated communication (+), representation of functional and non-functional requirements for different type of systems (+).

Lastly in the stakeholder area, the study found that stakeholders that affect to requirement elicitation were grouped in two classifications. The first group is the stakeholder who may affect the project (+) [18] that could depict by the stakeholder onion model, or the stakeholder taxonomy based on how each stakeholder effecting to the system. The second group is the stakeholder who is in different degrees of collaboration, representation, accountability, and committed knowledge (+) [24].

The findings from literature review according to RQ2 are also described in four areas. Firstly in the communication area, many studies addressed 12 language-related guidelines including use of communication media to support, a sense of co-location, synchronous interaction, understanding of the language and business culture of clients, encouraging face to face meetings, selecting a vendor with knowledge of the client 's language, reviewing project documents by native speaker, encouraging team members to use standard language or common language in order to avoid mis-interpretation, appointing team members to have fluency in English language, and appointing language translator [24], improving requirement elicitation communication through listening types that are useful for different purposes, e.g., discriminative, comprehensive, therapeutic, critical, and appreciative types [25], team having to leverage requirement and architecture by communication intensity, and levels of interaction and established project methodologies for tackling non-functional requirement [25]. Moreover, five guidelines for solving lack of informal face to face communication were also advised in [26] including adopting appropriate communication tools (e.g., videoconferencing, teleconferencing, data conferencing, and web-based technologies), encouraging frequent communications through latest technologies, daily exchange of project status by technologies (e.g., telephone calls, video conferences, and emails), creating communication protocol, and increasing frequency of communication between teams and users.

Secondly in the cultural area, cross-cultural software development teams cause individual cultural difference factors. Furthermore, there are i) cultural aspects between the vendor and the client organization level and ii) different cultural viewpoints in the global software development team at the country level that also affect requirement elicitation activities. In the matter of individual culture, there are five positive factors of having people culture center among agile teams including effective communication, motivated members, collaboration among members, proactive members, and present leaders [27]. Practical guidelines from [24] also suggested to i) establish open communication between stakeholders through face-to-face meetings, instant messaging and onsite visits; ii) use online tools for team-building if visits would not work; iii) define a cultural ambassador for the project to create teams with complementary skills and cultures; iv) create close cooperations between team members involved at both client and vendor sides to build trust-worthy relationship; v) build mixed teams with memberships from different cultural backgrounds; vi) create roles, relationships and rules to facilitate coordination and control over geographical, temporal and cultural distance; and viii) increase project members' domain knowledge.

Thirdly in the competency area, practical analytical skills in [28] suggested to build comprise of creative thinking, decision making, understanding the best way to learn and ability to design problem, system thinking, conceptual thinking, and visual thinking. Besides, communication and listening skills in [29] suggested to enhance comprise of verbal communications (e.g., restating concept, driving conversation, positioning content, neutral party), non-verbal communications (e.g., avoid defects, build trust, body language), and listening skills (e.g., word choices of voice, inject feelings, active listening from stakeholders

about pains and process and from developers about limitations and possibilities). Furthermore, the study in [30] stated that requirement engineer are required competencies for RE methods, RE experience, RE project management knowledge, and RE tool; while RE analysts are required to ask probing questions (PQs), because PQs could help them to elicit architecturally significant functional requirements that architects need [28]. Besides, the right business analyst (BA) for the right job should be assigned [31] BA's competency can be categorized in 3 levels, i.e., i) junior (novice) who has rule-based, strongly limited, and inflexible behaviors, ii) intermediate level who has ability to incorporate aspects of situation, and iii) senior (expert) level who has intuitive understanding of the situation and ability to zoom in on the central aspects.

Lastly in the stakeholder area, seven interesting techniques and guidelines for stakeholder interaction were suggested by [32]. It including 5 general practices that are i) getting all stakeholders to speak up and letting their opinions be known; ii) giving everyone a voice, possibly by letting stakeholders to contribute and discuss requirements anonymously; iii) using tools to support anonymous contributions; iv) educating all project stakeholders about the importance of engagement, and listening; and v) using techniques that classify stakeholders' attributes in terms of role, knowledge, power, interest, relationship, interpersonal skills [33], [34], and geographical position [34]. Combined with 2 specific practices according to the situation that are i) not counting an argumentative and non-collaborative stakeholder as an asset to the project and removing that person from the project critical path and ii) approaching any situation calmly and objectively. Besides, practical approach to select stakeholders for a software development project suggested by [34] can be performed by five steps, i.e., analyzing project scope, analyzing relationship between stakeholders, specifying stakeholder types, analyzing stakeholder's influence and interest to the project, and analyzing stakeholder's skill.

3.2. The findings from real-life software development projects

The findings based on the interviews 27 practitioners and focus group observation. The demographic attribute comprises different roles, experiences, and organizations in six software projects. Those of interviewees as shown in Figure 2, is presented as.

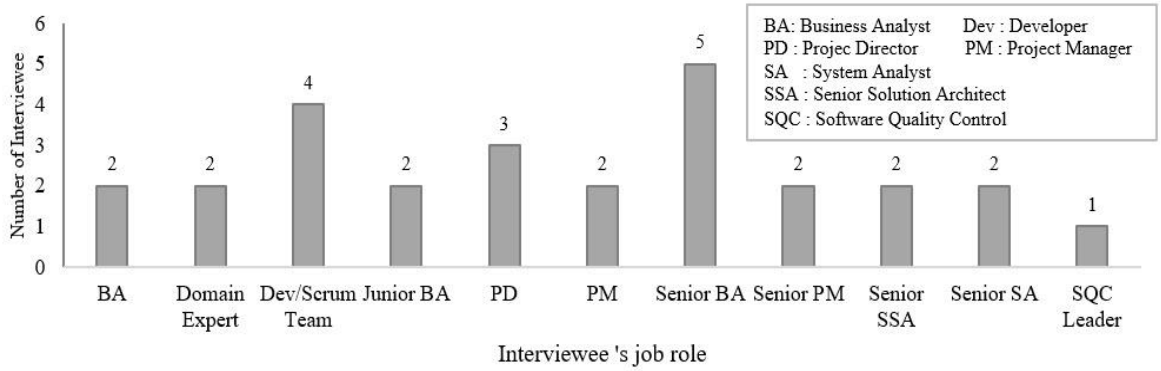

Figure 2. Demographic of interviewees

In the context of industry, the five software projects were implemented for an enterprise company by using an international commercial off-the-shelf (COTS) application along with new implementation of customized functions responsive to its client requirements, new downstream and upstream applications and one software project is to build a web application from scratch based on business requirements. All six projects were built from financial requirements challenges and required high availability and quality because consequences of software faults and delayed project cause daily penalty and financial damages for both clients and vendors. In the aspect of stakeholders, overall people in the five projects involved approximately 100 people comprising of i) business users including business domain experts and the end users; ii) in-house IT members including project managers, solution architectures, business analysts, programmers and testers; iii) vendors including product owners (multinational company) and local implementers (IT professional company); and iv) other stakeholders including senior managements, procurement staffs, acceptance committees, lawyers, steering committees and infrastructure engineers. In the aspect of information sources, all the projects' information was from i) stakeholders, which comprise of the end user domain experts, functional consultants, and technical consultants and ii) analysis documents including terms of reference documents, functional documents, API documents, and end user operation procedures. Regarding to the web

application project, the working team including 15 of business users, 2 business analysts, 1 project manager, 3 senior developers 2 junior developers and 2 testers.

In the aspect of software project management methodology, the five projects used a hybrid agile and waterfall approach. Before the requirement phase, functional consultants set up training courses to their clients for getting familiar with existing COTS features and functions. Output of this phase was requirement gap analysis results. The requirement phase had then conducted according to practitioner sides. Functional consultants and business analysts mainly used an interview as a primary technique to elicit requirements by sequentially performing following steps which include understanding domain area and problem domain, identifying and analyzing stakeholders, and planning requirement elicitation and getting to know what project deliverables are. After that, they performed requirement classification process and requirement prioritization and negotiation process, based on gap analysis documents. The functional consultants, technical consultants, and business analysts then wrote a software requirement specification (SRS) as per the collected requirements and the results of solution mapping. Finally, the project manager sent the SRS to business users, business domain experts, and decision makers to confirm and ask for their sign-off. The COTS software itself has one-week sprint to develop. New patch was scheduled of two-week delivery. In conclusion, in the software project management point of views the comparison results when the project ends up between requirement estimation and actual spend, showed that there were several late requirements providing in the user acceptance test phase instead of requirements providing in the requirement development and design phase. According to the project #6 which was the financial web application, its team used waterfall methodology due to the user's organization culture that reflected to the requirement artefacts. The document requirement specification must conform the organization template. It needed to be confirmed and sign-off before continuing to the development phase. The requirement which did not present in this document means that it was out of scope of the project.

3.2.1. Communication

In the aspect of communication for RQ1, there are 10 communication factors that affect requirement elicitation ineffectiveness supported by practitioners' statements as.

- Communication gap among requirement engineers, business analysts, and users can lead to (1) uncomfortable communication between users and business analysts by using English language as the medium, (2) misunderstanding in requirements due to different levels of English knowledge, (3) different levels of business understanding between functional consultants, business analysts, in-house IT members, and users, as said "*I am not sure the Swedish business consultant can get the point on what the end user wants because I myself don't clearly understand of Accounting terminology, I just interpreted user requirements from Thai to English word by word"* [by Practitioner #8-Senior Business Analyst].
- Unclear requirements and requirement misunderstanding can lead a business analyst to be unable to analyze, classify or prioritize given requirements. For example, a business requirement in one sentence can be differently interpreted among users, in house IT members, and requirement engineer, as said *"I saw the business requirement under this item stated that system must support all message xx format, the xx format is too broad"* [by Practitioner #7-Senior Business Analyst] and "*User states that system must support pricing calculation, e.g., P1 model and P2 model, whether the means that we have to cover other requirements than P1 and P2 model?"* [by Practitioners #6-8-Business Analyst and Senior Business Analyst].
- Different terminologies among functional consultants, technical consultants, and clients can lead to more time for functional or technical consultants to map software terms to user business term and vice versa, as said *"The term "Exception" in requirement means that system should leave this step to user to do the process manually, but "Exception" in developer 's perception means that system error and the process will stop.*" [by Practitioners #8-9-Senior Business Analyst, Senior Developer].
- Lack of communication about non-functional requirements (NFR) among business analysts, technical consultants, and developer can cause a lot of re-work to meet quality requirement acceptance, as said *"…why don't this NFR requirements been visited since in requirements phase instead of user acceptance phase, because I cannot fix them in this tight contract deadline*" [by Practitioner #16-Developer/Scrum Team], "*System performance was extremely slow I can't do my business on this system"* [by Business User #2] and *This performance issue needs to level up the severity to critical."* [by Practitioners #12 and Test Lead, #14-Project Director].
- Lack of communication among project managers, product owners, functional consultants, and technical consultants can lead to missing requirements, omitted requirements, and delayed project progress in a whole, as said *"I cannot attend almost the daily scrum meeting because there is always the schedule conflict with another request from my senior management, so I assigned the new guy of my team to attend*

instead" [by Practitioners #10-11-Project manager, Technical Lead], *"I cannot decide on this requirement I need to ask/consult my boss who was a domain expert in this area"* [by Business User #4], and *"These requirements are not complete the project steering advise us to separate the team to communicate the algorithms with relevant stakeholders to get them agree, we need to schedule the working team meeting more frequent than once a week…"* [by Practitioner #3-Senior Business Analyst].

- All team members lose the same page of requirements because significant project information does not share among all members in a team due to, e.g., no center of requirements repository and no mail group that can be reached all team members, as said "*… the latest document version I have is version 0.1, but now you say that the requirements solution was already updated in version 0.7 that why I can't find it, could you send it again to me*?" [by Practitioner #18-Domain Expert].
- There is lack of requirement negotiation in terms of system behavior or the limitation of COTS to reconcile among success-critical stakeholders and manage to sustain each stakeholder's expectations, as said "*As system show incorrect value which doesn't match user expected value… we have discussed internally and believe that this system limitation as it needs additional logic that is not currently in place"* [by Practitioner #22-Functional Consultant] and "*User informed that the result is not right … unfortunately I think we have come up against a blocking system limitation here."* [by Practitioner #21-Functional Consultant].
- There are a lot of ambiguity requirements in terms of reference document that required handful efforts to interpret, as said *"… the user says that the information to fill-out in this field can imply from the relevant fields or maybe I will put blank in this field"* [by Practitioner #15-Business Analyst].
- Domain experts do not transfer knowledge to in-house IT team members. This causes an issue on missing requirements and requirements change to match with the system in system integration phase, as said *"User just added the missing requirement in system integration phase instead of added it in requirement gathering phase the requirement is about the new system should send email alert to users when the amount is lower the threshold. This is because the in-house business analyst doesn't have clearly business process and knowledge on what data is sending and receiving that's why she did not aware to add this feature for user"* [by Practitioner #6-Business Analyst].
- A standard configuration document of COTS that is sent to clients does not clear enough and causes clients to miss some configuration requirements during go-live deployment. Consultants that should solve the problems cannot contact in time according to lack of communication between project manager and technical consultants to prepare for client support, as said *"This error happened because we are missing to set this flag to "Y", the technical consultants never informed us about this"* [by Practitioners #11 and #16-Technical Lead, Scrum Team].

As stated in [35], COTS's characteristics are as. A buyer has no access to source code. Vendor controls its development. It has nontrivial installed base. It needs more empirical understanding of COTS-based system (CBS) based on the following hypotheses in requirement engineering perspectives that are i) more than half of features of COTS software products going unused and ii) COTS assessment and tailoring efforts to meet user requirement being vary significantly by COTS products classes, i.e., database management systems, GUIs, networking, device drivers and operating systems.

In the aspect of communication for RQ2, there are four found guidelines as. i) there is a need to set up e-mail group and a central requirement repository of team members. Meanwhile, it needs to make sure that all team members receive the same information, ii) daily communication in team members is needed by changing traditional software development methodology to agile such as Scrum, iii) project plan should be updated and communicated to all stakeholders regularly, and iv) team members who have competency in English language skill and requirement engineering skills should be appointed.

3.2.2. Culture

In aspects of culture for RQ1, there are eight found factors supported by practitioners' statements as.

- The first issue is cultural diversity. Functional consultants are from multinational organizations. Users would therefore come from different cultures. For example, functional consultants and technical consultants are from India, while users are Thai. Functional consultants and technical consultants always eat lunch at the same time in their country, so that it causes overlap working hours, as said *"There was some scope-creep occurring in requirement gathering session which is out of scope of the system, the user question about how to do their business, some consultants are OK to give advice based on their experience, but some may not OK because it was not his role, I think the question that user have asked did not expect to happen in this session."* [by Practitioner #21-Project Manager and Functional Consultant].
- The second issue is timing problem. Users do not dedicate enough time for requirement elicitation activities, as said *"I have sent the request ask for a session to clarify user's requirements, but they are not*

providing their available time yet." [by Practitioners #10, #15, and #21-Project Manager, Business Analyst, Functional Consultant].

- Users do not speak their requirements until requirements are the same as their management, as said *"I am not sure whether this feature could be included in the system, I need to confirm with all of my team also my boss internally*." [by Business Users #3-4].
- There is strong power and politics in requirement elicitation activities, e.g., during requirement negotiation and requirement acceptance between steering committee members and team members, as said *"From the requirement session yesterday, I was amazed that the project coordinator had invited all management people and all of them had attended the meeting".* [by Practitioners #2-4-Project Manager, Senior Business Analyst, Assistant Project Manager].
- Some domain experts do not like system behaviors that then lead to plenty of change requests for customization, as said *"Why the system behaviors cannot fulfil this simple requirement, I don't feel comfortable to use it."* [by Business User #6].
- Several cultural transition activities are required, e.g., training, secure buy-in of senior leadership and project staff, readiness assessment of all stakeholders for transition to new software product, as said *"We need Customer Success manager to tender care to customer to keep their trust on our system"* [by Practitioner #27-Project Director].
- Users are reluctant to confirm and give signoffs on requirements before their managers confirm or sign first, as said *"For the next project it is better not to put the SRS sign-off in the payment milestone, according to our experience that users don't want to sign-off because their late requirements will go the change request process which means that those requirements cannot add to this project anymore*." [by Practitioners #10 and #17-Project manager, Project Director].
- Requirement engineers from the West have strong and direct comments to Thai users that cause unsatisfaction among users due to useless requirements, as said *".... If the knowledge around this area is weak, the need cannot be strong."* [by Practitioner #21-Functional Consultant].

In the aspect of culture for RQ2, there are six found guidelines as: i) in terms of national cultures, it should appoint the team who can work and understand well on different cultures as the communication mediator. Business analyst should in advance study of stakeholder's cultural and background, to mitigate the risk on requirements misunderstanding; ii) in terms of organization cultures, business analyst should comprehensively involve management to commit and support the project; iii) there is a need to get stakeholders agreement to support one another; iv) it is important to use contextual technique such as participant observation, ethnomethodology, and conversation analysis to capture tacit requirements from the users in different culture from the software development team; v) an organization should change its behavior to be a generative organization; vi) business Analyst should gain trust and relationship with the users, and vii) specific tool or technique for gathering requirements from anonymous users should be used.

3.2.3. Competency

In the aspect of competency for RQ1, according to [35], personnel capacities and experiences remain dominant factors influencing CBS-development productivity. Therefore, in the requirement perspectives if a practitioner makes a software project contract for software acquisition with no capability and experience to maintain COTS products, this means that there is no ability to tackle difficult COTS integration challenges without skilled support. Based on data collection there are three competency factors that affect requirement elicitation ineffectiveness supported by practitioners' statements as.

- Requirement engineers from the country of the local implementor side do not know COTS system well, so that they cannot clearly explain features or functions to business users, as said *"... this feature required the product owner to give the end user information directly due to this system feature is contrast with the old one and I don't know whether this is system behavior or an invalid business case"* [Practitioner #7-Senior Business Analyst].
- There is lack of communication skills of requirement engineers due to language and domain knowledge barriers, as said *"I don't want to discuss further on this requirement, please move to the next topic and prepare more on this then get back to me again next time"* [In Brainstorming session with Business User #8 and Domain Expert].
- There is lack of technical skills of requirement engineers, so that there are missing pieces of non-functional requirements, e.g., system performances and practical database design, as said *"... this feature took more than 15 minutes running and then timeout, we need to try re-build database index."* [by Practitioner #16-Developer/Scrum Team].

In the aspect of competency for RQ2, there are two found guidelines as: i) an organization should provide education course, training, or coaching for practitioners on technical essential, self-development and

soft skills, as well as business domain knowledge and ii) an organization should assign the right person to the right job according to the BA's competency in 3 levels from [29] as junior (novice), intermediate, and senior (expert) levels.

3.2.4. Stakeholder

In aspects of stakeholder for RQ1, there are two found factors that affect requirements elicitation ineffectiveness supported by practitioners' statements as. The first factor is the difference of users' personalities and characteristics. For example, the sale or the front office users are always proactive at work, but operation users or the business department have slow responses and conservative decision making. The second is a working style conflict between a multinational organization and a customer organization. For example, a customer organization has many levels and many steps of approval which are required an official procedure in requirements signoff, but a multinational organization (or local implementor's organization) has more flexible levels of management approval over software requirements elicitation issues, as said *"Based on the requirements of the last session, I don't know who the owner of this feature is, I have called Miss A and she said that she is not the owner also her department does not do this task."* [by Practitioner #15-Business Analyst] and *"I believe that the requirements from B department are mostly fulfill and get on your attention, but the requirements from my department have very slow-moving progress."* [by Business User #9 and Domain Expert].

In the aspect of stakeholder for RQ2, there are four found guidelines as: i) tool or technique to analyze, classify and prioritize stakeholder (e.g., stakeholder classification model, and stakeholder onion diagram) should be used, ii) in advance study of working style for each stakeholder, if there are stakeholder conflicts, business analyst should address situation calmly and objectively, iii) to increase stakeholder engagement, collaboration tools should be used, and iv) project status between development team and user should regularly updated.

3.3. Gap analysis

There are 16 discovered issues from data analysis of the problems that not yet to be solved based on the study of existing data and empirical data described in Table 5 (see in Appendix). No. 1-10 are related to communication. No. 11-12 are related to cultural problem. No. 13-14 are related to competency. No. 15-16 are related to stakeholder. The authors summarize recommend guidelines for each issue which can be adapted to the requirement elicitation problem space.

4. CONCEPTUAL FRAMEWORK

The result from recommendation solution and benefits to stakeholders can be incorporated in a conceptual requirement elicitation framework as presented in an abstract view in Figure 3 (see in Appendix). Together with a requirement elicitation activity in four phases which are A1: problem definition phase, A2: conduct requirement elicitation phase, A3: prototyping and confirmation phase, and A4: measurement and document phase, the bold line box represented a required activity or factor. While the dash line box represented an optional or adaptable one.

5. DISCUSSION

According to the result to answer RQ1 there are 10 factors categorized as ineffectiveness from literatures review, and 10 ineffectiveness factors from the industry study under communication effect, the cultural has total 10 factors from literature study which is ten ineffectiveness factors, two effectiveness factors then eight ineffectiveness factors has found on industry report. For competency, based on content-analysis 14 competencies were found and three issues that effect to requirements elicitation ineffectiveness were found in real-life project, lastly, in terms of stakeholder factor, the result from the past study has divided the stakeholder into two groups also two factors were introduced from a real-life project.

Further study on RQ2, the result suggested that 12 language-related, listening types, communication intensity, adopting online tool could fix communication problems and four guidelines were introduced from the real cases, in the aspect of culture, seven practices and five factors are provided with six guidelines were raised by real- world project study. In terms of competency, practical skills have found such as analytical skills, communication skills and RE experiences together with six guidelines form industry cases. Finally, for stakeholder section, seven techniques can be implemented also stakeholder selection should execute along with four approaches from the real cases. Based on this result study, it can strengthen that these four areas is contributed to requirement elicitation process as same as the study in [36] state the challenges on stakeholder's difficulties in understanding and prioritizing their need and the challenge that related to communicating and sharing unified view. However, they updated that if business team had educated

stakeholders, the challenge on communicating and sharing unified view could not be defected. Therefore, this can imply that requirement engineer could boost up their business knowledge competency on this effective approach. Moreover, due to more and more global and remote teams because of COVID-19 pandemic, there are challenges in requirement elicitation activity to be performed. These challenges involve i) virtual communication, ii) gaps in sharing knowledge, iii) difficulties in tracking progress and productivity and iv) possible time zone difference and cultural differences.

6. CONCLUSION

The proposed conceptual framework of requirement elicitation is provided in section 4. The authors have stated the 16 issues on requirements elicitation issues gaps then suggest the practices, tools and techniques which could be able to support requirement elicitation effectively. This conceptual framework aims for better system development and system enhancement in the aspects of communication, culture, competency, and stakeholder. The authors also encourage practitioners to adapt these practices towards requirement elicitation phases in modern software development project. However, in order to validate the framework in software engineering indicator the authors aim to use the statistical measures of software project success rate after applying this framework to an industry as a real case study in future.

APPENDIX

Table 5. Issues and guidelines on communication, culture, competency, and stakeholder

No.	Issue	Recommended Guideline
1	Language problem	– Improve language skills or language competency to requirement engineers. – Use several techniques and tools for requirement elicitation, e.g., prototyping, scenarios, and storyboards.
2	Misunderstanding and unclear requirements	– Combine prototyping with other techniques (e.g., questionnaire and think aloud protocol) to provoke discussion in group elicitation technique.
3	Lack of users' ability to say what they want	– Use observation techniques, e.g., field observation, apprenticing, contextual inquiry.
4	Incomplete requirements (e.g., late requirement in user acceptance phase instead of requirement elicitation phase)	– Use model-driven techniques, i.e., goal-based techniques (e.g., KAOS (Keep all objective satisfied) and I*) and scenario-based techniques (e.g., CREWS (Co-operative Requirements Engineering), use cases, and scenarios).
5	Lack of communication between key stakeholders	– Enhance comprehensive communication between stakeholders to share problems, issues, and concerns openly. – Use group elicitation techniques, e.g., brainstorming, focus groups, rapid application development (RAD), joint application development (JAD), workshops and consensus-building workshops with an unbiased facilitator. – Apply sentiment analysis of verbal communication in a software project meeting in order to detect destructive behavior that can danger project success [37].
6	Different terminology between users and requirement engineers	– Provide clear definition of terminology.
7	Unawareness of needs on both users and requirement engineers	– Use techniques to design and generate ideas, e.g., brainstorming, analogy technique, scenario, and storyboards.
8	Missing communication about non-functional requirements	– Extensively involve software architecture in requirement elicitation activities – Enhance technical knowledge for requirement engineer
9	Requirement changes related to management and political rules, acceptance criteria changes, unstable requirements, users' needs and expectations changing overtime	– Use agile requirement engineering methodologies. – Promote continuous integration and continuous delivery culture (generative organization culture) for software development team. – Attain collaborative working tools.
10	More and more global and remote teams (virtual communication) due to COVID-19 pandemic	– Enhance requirement engineering competency on communication skills – Use online collaborative tools, e.g., Zoom, Microsoft Teams, WebEx, Google Meet, Discord and Skype.
11	Cultural differences between business analysts and business users	– Use contextual techniques, e.g., participant observation, ethnomethodology, and conversation analysis
12	Culture differences between organizations	– Comprehensively involve management to commit and support the project – Get stakeholders agreement to support one another
13	Lack of hard skills (e.g., business domain and technical domain)	– Provide appropriate training and experience to perform requirement engineer roles
14	Lack of soft skills (e.g., communication skills and analytical skills)	– Pursue certification, e.g., the International Requirements Engineering Board (IREB) offering the Certified Professional for Requirements Engineering (CPRE) program, the International Institute of Business Analysts (IIBA) offering the Certified Business Analysis Professional (CBAP) program, and INCOSE offering the Certified Systems Engineering Professional (CSEP) program [38]

Table 5. Issues and guidelines on communication, culture, competency, and stakeholder (*continue*)

No.	Issue	Recommended Guideline
15	Differences of individual personalities and characteristics	– Build trust between requirement engineers and users – Be voiced and resolved, whenever a user or a practitioner has a concern
16	Differences of positions (e.g., upper-level managers, middle-manager, end-user)	– Encourage stakeholder commitment – Use cognitive techniques, e.g., protocol analysis and laddering

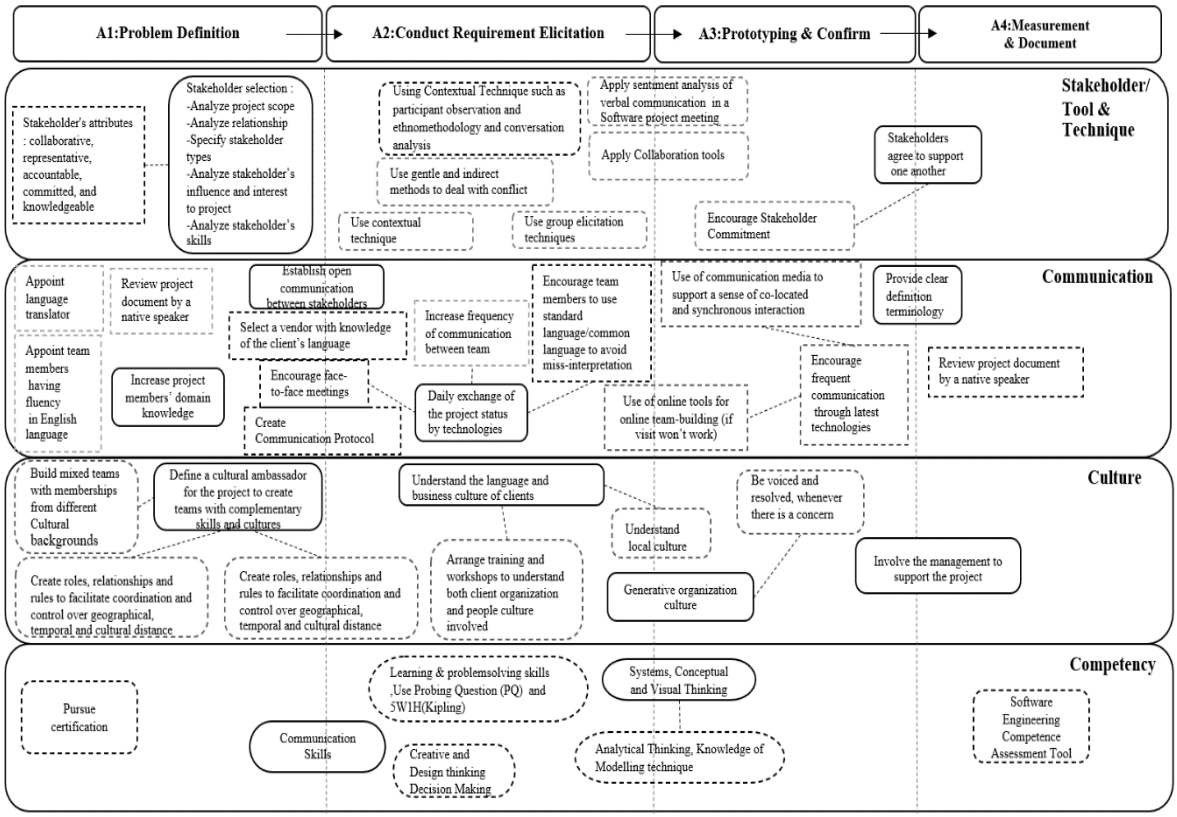

Figure 3. An abstract view of requirement elicitation conceptual framework

REFERENCES

[1] I. Sommerville, "Software engineering: principles and practice," *Software Engineering Journal*, vol. 9, no. 5, 1994, doi: 10.1049/sej.1994.0029.

[2] B. Nuseibeh and S. Easterbrook, "Requirements engineering," in *Proceedings of the conference on The future of Software engineering - ICSE '00*, 2000, pp. 35–46, doi: 10.1145/336512.336523.

[3] G. Starke, M. Simons, S. Zörner, and R. D. Müller, *arc42 by Example: Software architecture documentation in practice*. Packt Publishing Ltd, 2019.

[4] P. Spoletini and A. Ferrari, "Requirements elicitation: A look at the future through the lenses of the past," in *2017 IEEE 25th International Requirements Engineering Conference (RE)*, Sep. 2017, pp. 476–477, doi: 10.1109/RE.2017.35.

[5] X. Franch *et al.*, "How do practitioners perceive the relevance of requirements engineering research?," *IEEE Transactions on Software Engineering*, vol. 48, no. 6, pp. 1947–1964, Jun. 2022, doi: 10.1109/TSE.2020.3042747.

[6] R. R. Lutz, "RE at 50, with a focus on the last 25 years," in *2017 IEEE 25th International Requirements Engineering Conference (RE)*, Sep. 2017, pp. 482–483, doi: 10.1109/RE.2017.50.

[7] K. L. D. Häußer, H. van Loenhoud, A. Schwarz, and P. Steiger, "Handbook of advanced level elicitation according to the ireb standard," www.ireb.org, International Requirement Engineering Board. https://www.ireb.org/en/downloads/tag:handbook (accessed Sep. 09, 2020).

[8] A. Chaipunyathat, "Effects of disruptive change in software development in the context of digital government in Thailand," in *Proceedings of the European Conference on e-Government*, Oct. 2019, pp. 200–209, doi: 10.34190/ECDG.19.047.

[9] J. Cleland-Huang, "Disruptive change in requirements engineering research," in *IEEE 26th International Requirements Engineering Conference (RE)*, Aug. 2018, pp. 1–2, doi: 10.1109/RE.2018.00-58.

[10] K. Villela *et al.*, "Towards ubiquitous RE: a perspective on requirements engineering in the era of digital transformation," in *IEEE 26th International Requirements Engineering Conference (RE)*, Aug. 2018, pp. 205–216, doi: 10.1109/RE.2018.00029.

[11] K. Intawong, D. Olson, and S. Chariyalertsak, "Application technology to fight the COVID-19 pandemic: lessons learned in Thailand," *Biochemical and Biophysical Research Communications*, vol. 534, pp. 830–836, Jan. 2021, doi: 10.1016/j.bbrc.2020.10.097.

[12] M. Bano, D. Zowghi, and C. Arora, "Requirements, politics, or individualism: what drives the success of COVID-19 contact-tracing apps?," *IEEE Software*, vol. 38, no. 1, pp. 7–12, Jan. 2021, doi: 10.1109/MS.2020.3029311.

[13] B. Davey and K. R. Parker, "Requirements elicitation problems: a literature analysis," *Issues in Informing Science and Information Technology*, vol. 12, pp. 71–82, 2015, doi: 10.28945/2211.
[14] L. R. Wong and D. S. Mauricio, "New factors that affect the activities of the requirements elicitation process," *Journal of Engineering Science and Technology*, vol. 13, no. 7, pp. 1992–2015, 2018.
[15] Neetu Kumari.S and A. S. Pillai, "A study on the software requirements elicitation issues - its causes and effects," in *Third World Congress on Information and Communication Technologies (WICT 2013)*, Dec. 2013, pp. 245–252, doi: 10.1109/WICT.2013.7113143.
[16] S. Aaramaa, S. Dasanayake, M. Oivo, J. Markkula, and S. Saukkonen, "Requirements volatility in software architecture design: an exploratory case study," in *Proceedings of the 2017 International Conference on Software and System Process*, Jul. 2017, vol. Part F1287, pp. 40–49, doi: 10.1145/3084100.3084105.
[17] J. Cleland-Huang, R. S. Hanmer, S. Supakkul, and M. Mirakhorli, "The twin peaks of requirements and architecture," *IEEE Software*, vol. 30, no. 2, pp. 24–29, Mar. 2013, doi: 10.1109/MS.2013.39.
[18] M. Plachkinova, G. Moody, and K. Peffers, "Selecting communication artifacts for requirements engineering," *Journal of Information Technology*, vol. 19, no. 4, pp. 108–130, 2018.
[19] A. Sadig and A.-E.-K. Sahraoui, "Culture effect on requirements elicitation practice in developing countries," *International Journal of Software Engineering and Applications*, vol. 8, no. 1, pp. 49–58, Jan. 2017, doi: 10.5121/ijsea.2017.8105.
[20] P. G. Kim, J. Humble, P. Debois, J. Willis, and N. Forsgren, *The DevOps handbook: how to create world-class agility, reliability, & security in technology organizations*. 2021.
[21] "G. J. H. Geert Hofstede *et al.*, "Six dimensions of national culture," geerthofstede.com, https://geerthofstede.com/culture-geert-hofstede-gert-jan-hofstede/6d-model-of-national-culture/ (accessed Jun. 20, 2019).
[22] J. Tizard, T. Rietz, X. Liu, and K. Blincoe, "Voice of the users: a study of software feedback differences between Germany and China," in *Proceedings of the IEEE International Conference on Requirements Engineering*, Sep. 2021, pp. 328–335, doi: 10.1109/REW53955.2021.00057.
[23] M. Romero, A. Vizcaíno, and M. Piattini, "Competencies desirable for a requirements elicitation specialist in a global software development," in *Proceedings of the Third International Conference on Software and Data Technologies*, 2008, vol. SE, no. GSDCA/M/-, pp. 347–354, doi: 10.5220/0001878703470354.
[24] R. A. Khan, U. Khan, and M. Niazi, "Communication and coordination challenges mitigation in offshore software development outsourcing relationships: findings from systematic literature review," in *The Tenth International Conference on Software Engineering Advances*, 2015, no. c, pp. 45–51.
[25] Z. J. Oster, "Improving requirements elicitation through listening research," in *1st International Workshop on Learning from other Disciplines for Requirements Engineering (D4RE)*, Aug. 2018, pp. 26–28, doi: 10.1109/D4RE.2018.00013.
[26] J. O. Johanssen, A. Kleebaum, B. Bruegge, and B. Paech, "How do practitioners capture and utilize user feedback during continuous software engineering?," in *IEEE 27th International Requirements Engineering Conference (RE)*, Sep. 2019, vol. 2019-September, pp. 153–164, doi: 10.1109/RE.2019.00026.
[27] L. Felipe Amorim, M. Marinho, and S. Sampaio, "How (UN) happiness impacts on software engineers in Agile teams?," *International Journal of Software Engineering and Applications*, vol. 11, no. 3, pp. 39–57, May 2020, doi: 10.5121/ijsea.2020.11303.
[28] A. Kalevela, "Business analysis required skills," www.metabusinessanalyst.com, https://www.metabusinessanalyst.com/business-analysis-required-skills-analytical-thinking-part-1/ (accessed Aug. 08, 2020).
[29] A. Herrmann, M. Daneva, C. Wang, and N. Condori-Fernandez, "Requirements engineering in job offers : who works in re and what competences do they need, particularly soft skills?," *Requirements Engineering Magazine*, 2020. https://re-magazine.ireb.org/articles/requirements-engineering-in-job-offers (accessed May 09, 2021).
[30] C. Wang, P. Cui, M. Daneva, and M. Kassab, "Understanding what industry wants from requirements engineers," in *Proceedings of the 12th ACM/IEEE International Symposium on Empirical Software Engineering and Measurement*, Oct. 2018, pp. 1–10, doi: 10.1145/3239235.3268916.
[31] A. Babar, D. Bunker, and A. Qumer Gill, "Investigating the relationship between business analysts' competency and IS requirements elicitation: a thematic-analysis approach," *Communications of the Association for Information Systems*, vol. 42, no. 1, pp. 334–362, 2018, doi: 10.17705/1CAIS.04212.
[32] J. Cleland-Huang, "Stakeholders on the prowl," *IEEE Software*, vol. 33, no. 2, pp. 29–31, Mar. 2016, doi: 10.1109/MS.2016.45.
[33] F. Anwar and R. Razali, "A practical guideline of selecting stakeholders for requirements elicitation - an empirical study," *International Journal of Software Engineering and its Applications*, vol. 9, no. 2, pp. 95–106, 2015, doi: 10.14257/ijseia.2015.9.2.09.
[34] Y. Zhao, "A systematic stakeholder selection model in requirements elicitation for software projects : a systematic mapping study," M.S. thesis, Department of Software Engineering, Faculty of Computing, Blekinge Institute of Technology, 2018.
[35] C. J. H. Mann, "Software engineering: Barry W. Boehm's lifetime contributions to software development, management, and research," *Kybernetes*, vol. 37, no. 2, Feb. 2008, doi: 10.1108/k.2008.06737bae.001.
[36] C. Palomares, X. Franch, C. Quer, P. Chatzipetrou, L. López, and T. Gorschek, "The state-of-practice in requirements elicitation: an extended interview study at 12 companies," *Requirements Engineering*, vol. 26, no. 2, pp. 273–299, Jun. 2021, doi: 10.1007/s00766-020-00345-x.
[37] M. Herrmann and J. Klunder, "From textual to verbal communication: towards applying sentiment analysis to a software project meeting," in *Proceedings of the IEEE International Conference on Requirements Engineering*, Sep. 2021, pp. 371–376, doi: 10.1109/REW53955.2021.00065.
[38] D. R. R. Young, "Toward better RE the main thing is keeping the main thing the main thin," *Requirements Engineering Magazine*, 2014. https://re-magazine.ireb.org/articles/toward-better-requirements-engineering (accessed May 09, 2021).

BIOGRAPHIES OF AUTHORS

Ajchareeya Chaipunyathat Ph.D. student in Information and Data Science program at the Department of Information Technology, Faculty of Information Technology and Digital Innovation, King Mongkut's University of Technology North Bangkok. Her research interests include requirement engineering, requirement elicitation, software engineering, data science, and natural language processing. She can be contacted at email: s6007011966055@email.kmutnb.ac.th.

Nalinpat Bhumpenpein holds Dr. tech. (Business Informatics), University of Vienna, Austria. Currently, she is an Associate Professor at Faculty of Information Technology and Digital Innovation, King Mongkut's University of Technology North Bangkok. Her research interests include software engineering, requirement engineering, knowledge transfer, agile methodology, and IS/IT management. She can be contacted at email: Nalinpat.b@itd.kmutnb.ac.th.