

A Framework for Agile Global Software Development: The Case of an International Software Company in Egypt

Taghreed A. A. Motawi*, Amr Kamel and Hesham Hassan

Computer Science Department, Faculty of Computers and Information, Cairo University, Cairo, Egypt; taghread.motawe@gmail.com, a.kamel@fci-cu.edu.eg, h.hassan@fci-cu.edu.eg

Abstract

Objectives: Presenting the results of an Egyptian branch of an international company as a case study. The existing gap in the knowledge types and the impact of the lack interaction was investigated within the structure of the company team. **Methods:** This study is an exploratory case study. To enhance the data collection process and to answer the study' questions, the following three phases were performed: interviews, online survey, and interviews. A descriptive statistical analysis and root - cause analysis were used to analyze the online survey in quantitative and qualitative attributes and a general framework coding for a thematic analysis to analyze the interviews. **Finding:** The results explore that: (1) the current structure of the Egyptian branch of an international company case study distributed teams faces an existing gap in the knowledge types (domain, technical, and application). (2) The existing gap in both of domain and technical knowledge leads to a gap in the application knowledge. (3) The lack of using qualified document, recording audio and video meetings between the distributed teams, Co-located practice, wikis (asynchronous tool) besides email, and the yearly planned formal training and training course create root causes for the existing knowledge gap. (4) Qualified document is standardized and contains the both test cases and user stories. (5) Recording audio and video meetings between the distributed teams especially in complex project is an excellent feedback. (6) The strategy of the Co-located practice, where the designers (offshore members) travel to collocate with Client engagement team (onshore team) during the requirements document preparing, is preferred by the most participants than the opposite. **Application/Improvement:** Capturing the existing gap in knowledge types for the agile partial offshoring model will lead to improving this model. Which increase the software quality and decrease delivery time and cost for international software development companies and software development' clients.

Keywords: AGSD, Communication, Coordination, Distributed Teams, Knowledge Sharing, Knowledge Types

1. Introduction

The appearance of the Agile Manifesto in 2001 has brought significant changes to the field of software development and project management. The agile's practices and approaches were spread and approved by scholars and practitioners. An agile Software Development (ASD) methodology has improved the process of software development by increasing the quality of software and decreasing delivery time. The combination of both Global Software Development (GSD) and Agile brings more benefits as "continuous integration, configuration management, improved communication, sprint reviews, increased trust"¹, and it brings new challenges¹. The challenges that have been produced

from this combination were categorized into three categories: temporal distance, geographical distance and socio-cultural distance, in relation with communication, coordination and control². In view of the fact that values and principles of agile and global projects environment are opposite extremes on a continuum, a study³ had summarized the contradictory characteristics of agile and GSD to understand the fundamental differences between the two approaches. It mentioned that communication is the main issue in GSD and considered it as one of the fundamental agile values. As for agile teams, face-to-face meeting is the most effective method of knowledge exchange to and within the members of the development team. This is difficult in global teams where the members are located in different

*Author for correspondence

geographical sites. Thus, we can realize the importance of two factors besides the importance of communication. These two factors are knowledge sharing and human resources. Nevertheless, knowledge sharing depends on technology and coordination between human resources. In this context, this study attempts to provide a practical framework model for Agile Global Software Development (AGSD), which takes into consideration knowledge sharing and communication and coordination between human resources in relation to agile GSD challenges. In order to make the process of knowledge exchange more easily and enhance the interaction between distributed teams.

The purpose of this paper is to identify the existed gap in literature in the field of AGSD specifically in the area of knowledge type (domain 'business', application, and technical), and to discuss in detail, the results of the Egyptian branch of an international company case study which based on two sub questions:

RQ1.1: Is there an existing gap in the knowledge types (domain, technical, and application) between distributed parties, at the structure of the current team?

RQ1.2: What is the impact of the lack of using the substitutions that mitigate challenges of both communication and coordination, between the distributed parties at the structure of the current team on the existing knowledge gap? These two sub questions are for the objective of identifying the weakness of the structure of the distributed teams of the Egyptian branch of an international company case study from the perspective of the existing gap in the knowledge types (domain, application, and technical) and the impact of the lack interaction between distributed teams on the existing knowledge gap.

2. Theoretical Base

This section highlights GSD, agile implementation challenges in GSD, more focus in some strategies/practices/techniques to overcome AGSD challenges, and knowledge challenges in AGSD. In addition, a conclusion is written at the end of this section.

2.1 Global Software Development (GSD)

A study⁴ proposed a conceptual framework and experimental analysis for case study to a company that has two teams in UK and India. It analyzed the knowledge gap between the distributed teams' members, its conceptual framework explored the knowledge gap on two main

keys: (1) the type of knowledge (domain knowledge, application knowledge, technical knowledge), and (2) the prior knowledge (education, experience). The importance to use the same language for effective coordination and enquiries was mentioned by a previous study⁵. This study⁵ was about a case study of a company with two branches: one in UK, where the team is responsible for artifact, and the other one in Bangladesh, where the team is performing code and technical analysis. Some researchers in their study⁶ noticed that the previous studies focused on the software development process; therefore, they decided to expand their study towards that direction. In their study, they examined two units from multinational software development organizations as a case study and proposed a reference model as a result to their findings in that case study.

2.2 Agile Implementation Challenges in GSD

The number of studies that had investigated the challenges of using agile methodologies in GSD was not enough taken into consideration that the arising challenges of agile or GSD were separately identified but the combination of both agile and GSD was under examination in real context⁷.

Several studies^{8-14,1} identified communication, coordination, collaboration, cultural differences, documentation, training on agile practices, and distribution of work as the challenges that arose from AGSD. The importance of conducting empirical studies on agile global software development was highlighted¹. There are few studies that investigated the knowledge field in AGSD^{15,16}. The number of studies that focused on the sharing of knowledge in the agile distributed development is rare¹⁵. Also, the reasons behind the occurrence of difficulties in knowledge sharing are cultural, temporal distance, and spatial, which happened because of the lack of face-to-face contact¹⁶. There are three practices: "maintain product / process repository, focus on well-understood functionality rather than critical new functionality, and short cycle but not time-boxed development"⁹ (p.34) in order to aid the developers to acquire a common realizing of both application and business domains⁹.

A study¹⁷ had investigated the both the challenges that related to software development when the team is geographically distributed, and the benefits that arising from implementing agile process in the distributed development. Also, it proposed the following two mod-

els for agile distributed teams' structure: (1) complete offshoring, and (2) partial offshoring. The partial offshoring model consists of two main parts: (1) the design team, where it located onshore with the client and sends the requirements (features list) to the onshore team, and (2) the onshore team, which is responsible for delivering the product.

But, the number of studies which have investigated knowledge types (domain, application, and technical) gaps in AGSD is rare. Besides, the use of framework, tools, and methods in agile distributed environment is small. Three researchers¹⁴, who presented a systematic review of agile distributed teams, built on 63 papers from the year 2007 to the year 2012, illustrated that 2% are using framework, 3% are using methods, and 9% are using tools. In fact, their evidence has lead them to the fact that the agile methods in distributed environment. Such evidence needs to be customized. The customized process could include several combinations of framework, tools, practices, and methods¹⁴.

2.3 More Focus on Some Strategies/Practices/Techniques to Overcome AGSD Challenges

2.3.1 Documentation

Three researchers¹⁸ pointed out that much of knowledge remains tacit as a result for using agile in GSD because there is less consideration to documentation. A study⁹ proposed for documenting requirements in different formal representation as a balancing performance to achieved agility and distribution. In addition, two studies^{10,11} proposed preserving valuable documentation to increase the collaboration between distributed teams as a practice to overcome AGSD challenges. Also, the technique of utilizing documentation and standards for the shared design and goals in order to mitigate the confusion between AGSD teams, which is a communication process challenge was presented by a previous study¹³.

2.3.2 Co-Located Distributed Teams for a While

Collocate is the best practice to get better trust among the distributed teams and it also supports the collaboration between distributed teams as mentioned by a study¹⁴, which also proposed the following two kinds of face-to-face meeting strategies in order to raise the comfort level between distributed teams and to support the created trust. These two kinds of strategies are (1) "Seed

Visits"¹⁴ and (2) "Maintenance Visits"¹⁴ as a study called it. Furthermore, the following two different co-located practices to mitigate different communication challenges: (1) the need of physically co-located of one team and (2) promote orderly visits between distributed locations¹³. In addition, "Use Contact Visits" identified as a practice to overcome AGSD challenges^{1,11}. Besides that, this practice is where the project team members are collocated together during the first iterations¹¹. Some studies identified the orderly visits by the product owner to offshore location as a way for increasing the domain knowledge¹⁰.

2.3.3 Using Asynchronous Tools as Wikis besides Email

Some researchers¹⁹ define wikis in this term: "Wikis are an asynchronous platform for agile developers' communication and, thus, mostly helpful for progress tracking". Various previous studies indicated that (1) the founded fact that in the implementation stage of the project, wiki is considered as the most beneficial communication conduit because of the nature of the divided effort and that (2) the gain of using wiki-based systems gives a clear environment by making it simple to check project state, to upgrade tasks list, and to observe the work progress of the team members¹⁴.

The process of sharing knowledge considered as one of the three scenarios of how the agile team members can use wikis. In addition, the team member can share knowledge by posting their questions to a wiki and their colleagues can notice their questions and then give them an answer. And, they can post their experience of solving serious problems, which can help others, who are in the same situation¹⁹.

2.3.4 Recording Audio and Video Meetings

Several studies^{9-11,14} have mentioned the effectiveness of using video meetings for increasing collaboration. However, the coordination costs increase according to temporal distance, and the use of asynchronous tools such as video and audio recordings may reduce the coordination cost because it provides the ability to distribute relevant data of the meetings after and /or before. Moreover, the operation of recording meetings of the sprint planning phase and the ability of offshore members to play it back increase the cooperation by strengthening the understanding²⁰. Two researchers²¹ highlighted the technique of recording conference calls and replay for raise understanding²¹.

2.3.5 Training

As for formal training, according to a study¹⁶, the practice, formal training is the practice where the formal training managers have the ability to standardize the contents of the training programs across the distributed teams. Therefore, where it includes technical skills, agile practices, soft-skills, and business knowledge and which consider helpful tools to strengthen both knowledge and skills of the members of the team. Furthermore, it suggested that the formal training, which the organization' management support it by designed it and implemented it across the organization, is required to avoid the consuming money and time¹⁶. Three researchers¹⁴ mentioned some studies that recommended that the employees of the organization be subjected to a weekly training course. They identified the area of the training course as: (1) Agile, (2) distributed team structure, and (3) processes¹⁴.

3. Case Study Methodology

This section highlights four parts: (1) the Egyptian branch case study design, (2) organizational distributed teams of the Egyptian branch case study, (3) boundaries of the Egyptian branch case study, and (4) the both of data collection and analysis for the Egyptian branch case study are included under the method of the Egyptian branch case study title at the end of this section.

3.1 The Egyptian Branch Case Study Design

This is an exploratory case study type²². This research case study was developed in Egypt software development branch. This branch belongs to International Turnkey Systems Group Company (ITS). This is an international software development company located in Kuwait (head quarter). The company was selected based on the following criteria: different cultural context with the same language (Arabic), but with different dialects, Agile Global Software Development (AGSD), and the organizational structure for the distributed teams. The data collection was collected via interviews (online survey and individual interviews)²². For data analysis, the study used a descriptive statistical analysis and Root – Cause Analysis (RCA) to analyze the online survey in quantitative and qualitative attributes in order to get the purpose of data collection.

3.2 Organizational Distributed Teams of the Egyptian Branch Case Study

The company' model for the team structure is based on the agile “partial offshoring”¹⁷ model. It consists of two main parts: 1) The Client engagement team (CET), and 2) The Global Delivery center as shown below (Figure 1).

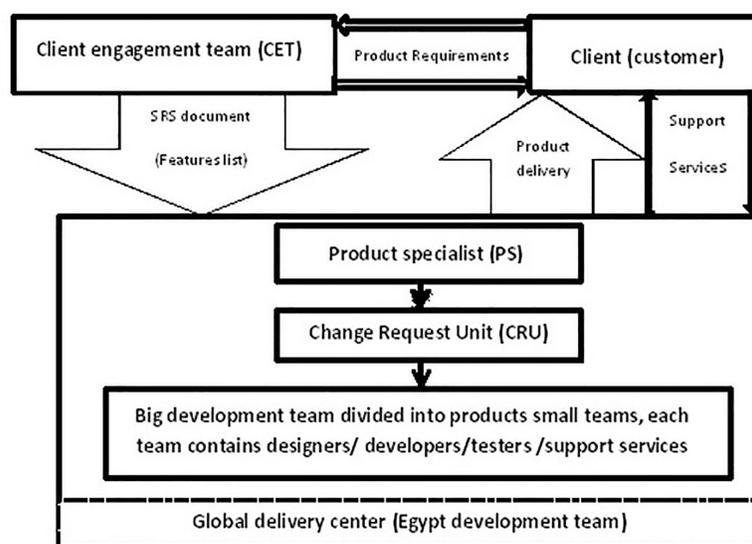


Figure 1. Company distributed teams.

3.2.1 The Client Engagement Team (CET)

Client Engagement Team (CET) exists wherever the client (customer) exists. Such a team may be located in Kuwait, Bahrain, Lebanon, and Malaysia... etc. In fact, CET is responsible for gathering the system requirements from the client, preparing the System Requirements Specification (SRS); and then, sending SRS to the global delivery center (Cairo). Also, CET is responsible of the direct interaction with the development team in global delivery center as the product owner. In addition, CET contains of at least one member. It may consist of more than one member according to the size of product and availability of members.

3.2.2 The Global Delivery Center

This center is responsible for technical design, coding and delivering the client product. In addition, the development team is divided into smaller development teams. Each small team contains ten members including: (1) designers, (2) developers, and (3) quality assurance members (testers). Moreover, each team develops one more than one special product. In fact, the designers are responsible for analysis and design the requirements specification. Also, they do not have a direct interaction with the client except in some few cases when the project is complex.

3.3 Boundaries of the Egyptian Branch Case Study

To overcome the challenges that appear in the Egyptian branch case study, the following boundaries have been selected in order to clarify and limit the problem area. The Egyptian branch case study focused on the first stage where the communication, coordination, and knowledge sharing between CET and development team occurs at the process of analyzing and designing the requirements specification. The Egyptian branch case study included designers' participants. However, the main concern of the Egyptian branch case study was on the knowledge sharing gap and the interaction between distributed members.

3.4 The Method of the Egyptian Branch Case Study

To clarify the Egyptian branch case study methodology, the available resources had been controlled the procedures of the study data collection. The main and only

available resource was the human resource with a consideration of the communication challenge faced during the process of data collection. In order to enhance the process of the data collection and to answer the study' questions, the study performed three phases in the following order: 1) Phase I: level 1_ semi-structured interviews, 2) Phase II: online survey, and 3) Phase III: level 2_interviews& feedback.

3.4.1 Phase I: Level 1_Semi-Structured Interviews

The objective of this data collection procedure was to understand: 1) the implemented steps of the process of the software development and the involved people in this process and 2) the structure of the development team.

The participants of the level1_semi-structured interview were five key persons who have worked in global delivery center at Cairo, Egyptian. An in-person interview was conducted with PMP manager – PMO (Software Development) and senior project manager for almost an hour. Their selection was based on a recommendation from the chief operations officer. Moreover, other interviews were performed by phone with three heads of the design unit for different teams which works on different products. The participants were two females and three males. All interviews were conducted in Arabic language while software engineering terminologies were in English. And notes were written in Arabic as well software engineering terminologies in English.

3.4.2 Phase II: Online Survey

The aim of designing this online survey was to answer the study' two sub-questions: RQ1.1 and RQ1.2.

The survey was designed on website monkey surveys (2016). <https://www.surveymonkey.com>, retrieved in March 8, 2017. The original survey was designed based on the following three resources: (1) the results from the Phase I: Level1_semi-structured interviews. (2) A study⁴ that addressed the existing gap in the knowledge types (domain, application and technical) in the GSD. (3) A master thesis²¹ which mentioned the mitigated solutions of both of communication and coordination challenges in AGSD. The survey was designed in the English language, with 60 questions. The available participants were mainly working or still working designers at the global delivery center. The study had 16 responses out of 21 invitations sent (Table 1).

Table 1. Related response by participant type

Participant	Total Number of Participants	Complete Response	Partial Response	No Response
Design head	5	4		1
Designer member	13	7	2	4
Senior designer	2	2		
Previous design head	1	1		
Total	21	14	2	5

The survey was distributed among the participants by sending daily email invitations. The responses were received for 46 days, starting from December 27, 2016 until February 8, 2017. The researcher preceded three focal points to get, at least, the participants' email addresses. In fact, the response of the three focal points took about 164 days, starting from July 14, 2106 until December 24, 2016. The response was weak despite the length of time. Quantitative and qualitative data analysis methods were used. First, the descriptive statistical analysis was used to briefly summarize a part of the given sample of the data, in addition to answer the RQ1.1. Second, root causes analysis method was used to analyze the second part of the sample data in order to get the root cause of the existing gap and to answer the RQ1.2. In addition, in root cause analysis, tree diagram was used as a tool.

3.4.3 Phase III: Level 2 Interviews and Feedback

In order to enable the researchers to communicate with the designers and to get more clarification and feedback about the survey results, the following three kinds of interviews were conducted:

3.4.3.1 An In-Person Interview

An in-person interview was conducted by Skype, with the Chief Operations Officer for about half an hour, to discuss with him (1) the survey progress, and (2) the communication challenges that have been faced during the interview phase. The chief operation officer recommended communicating with a quality assurance manager who could give more accurate answers to the questions' raised because of his knowledge of the existing challenge, and who could help in the communication problem with other participants.

3.4.3.2 An Unstructured Interview

An unstructured interview was conducted and recorded, with a quality assurance manager and his team (three

team members). The interview duration was for an hour and a half in order to allow flexible discussion and to share with them a list of desirable participants' emails. In return, clarifications and feedbacks for the survey' results were obtained as well as a confirmation of commitment to support the research.

3.4.3.3 In-Depth Interviews

In-depth interviews were conducted by phone with twelve designers where ten of whom has participated in the online survey. The study used confirmation and disconfirmation case strategy from the purposeful sampling strategies in selecting participants for the interviews, in order to get more clarification and feedback about the survey results. The researcher took detailed hand-written notes for six interviews and recorded audio recordings for six interviews. Besides, the researcher took hand-written notes as a reference for using in the process of data analysis. The used language during the interviews was Arabic while software engineering terminologies were in English. Finally, the interview process lasted 193 days, starting from April 10, 2017 until October 31, 2017. The researcher used a "general framework coding"²³ for a thematic analysis²³.

4. The Egyptian Branch Case Study Results and Discussion

In this section, the study presents three parts as (1) the results of the Phase I: level 1_ semi-structured interviews. (2) An attempt to answer the two proposed sub-questions by illustrated and discussed the results of both phases: Phase II: the online survey and Phase III: the level 2_ interviews and feedback.

4.1 Level 1_Semi-Strucured Interviews

A Figure of the Egyptian branch case study organizational distributed teams was developed after reviewing inter-

viewees' responses. The results were presented in details at the part (2), organizational distributed teams of case study, of the previous section (3).

4.2 The Weakness of the Structure of the Distributed Teams of the Egyptian Branch Case Study

The study investigated the structure of team for the Egyptian branch of an international company case study from two basics topics: (1) Identifying of the existing knowledge gap and highlighting its root causes, and (2) the impact of the lack of using the substitutions, that mitigate challenges of both communication and coordination, on the existing knowledge gap. The objective of this investigation is to capture the current gap in the literature in the AGSD area specifically the gap in knowledge type (domain 'business', application, and technical), and to identify the weakness of the structure of the distributed teams of the Egyptian branch case study from the perspective of the existing gap in the knowledge types (domain, application, and technical) and the impact of the lack interaction between distributed teams on the existing knowledge gap. As a result, the researcher posed (two) sub-questions. In an attempt to answer the proposed sub-questions, the researcher followed three steps; first and second, the researcher designed the survey & interviews to answer the sub questions: RQ1.1 and RQ1.2. Thirdly, the researcher analyzed the survey designed about the Egyptian case study by mixed methods of quantitative and qualitative analysis to find the knowledge gap and explorative the root causes for the existing challenges, and last, the researcher interviewed with the survey's participants to confirm the existing challenges and to explore the solutions that may mitigate the existing challenges. In the next two paragraphs the following will discuss: Answering the two sub-questions at the first two paragraphs.

4.2.1 The Existing Gap in the Knowledge Types (Domain, Technical, and Application)

The current structure of the distributed teams faces an existing gap in the knowledge types (domain, technical, and application); which negatively influence the development cycle. It is apparent that, from the survey results, the designers faced the following: 1) a real gap in their application knowledge, in addition 2) an expected gap in both

of their domain knowledge and technical knowledge. The next following three paragraphs will illustrate the results that examined the gaps. Then, the fourth paragraph will discuss the results.

4.2.1.1 A Real Gap in the Application Knowledge

More than two-thirds of the designers faced a lack in sufficient information exchange to accomplish tasks; they asked a question about the requirements or client system that was not adequately answered for them. And this lack did not influence the analysis and design phase only, but it also influenced the coding phase (Figure 2).

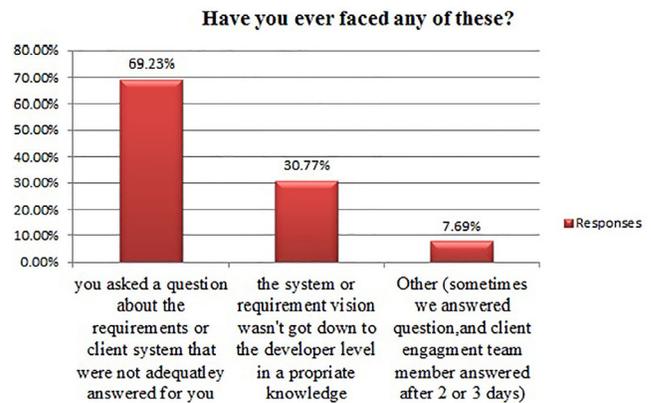


Figure 2. The Percentage of the level of insufficient information exchange.

4.2.1.2 An Expected Gap in the Domain Knowledge

The designers faced an actual lack of the domain knowledge, which was illustrated by a number of real situations (Figures 3, 4).

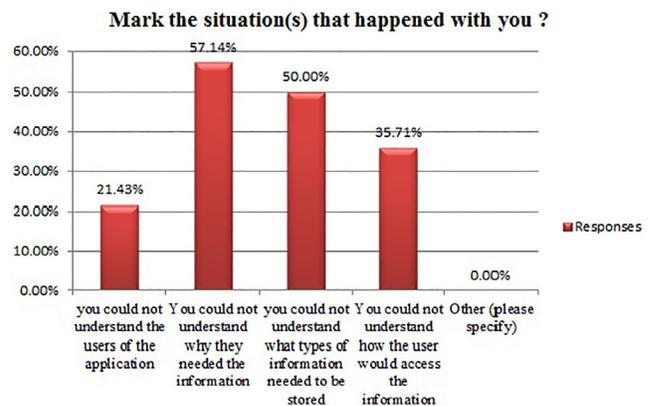


Figure 3. The Lack1 of domain knowledge.

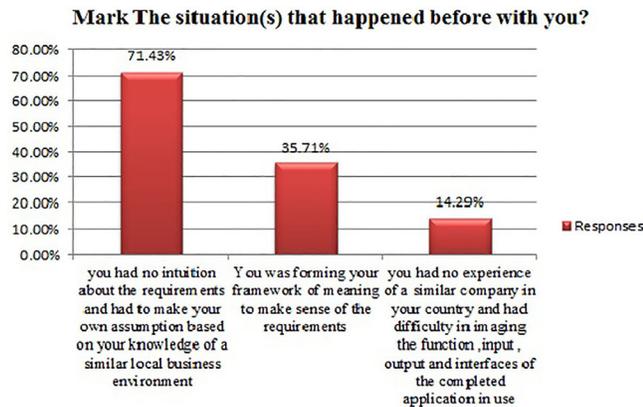


Figure 4. The Lack2 of domain knowledge.

4.2.1.3 An Expected Gap in their Technical Knowledge

Less than half of the designers did not have a suitable technical knowledge, which is required for their application knowledge (Figure 5).

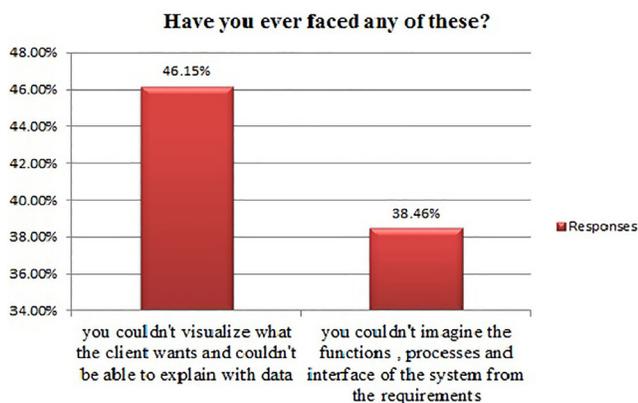


Figure 5. Lack of technical knowledge.

4.2.1.4 Discuss

What is presented here from this study’s results about the existing gap in the knowledge types (domain, technical, and application) is congruent with what was illustrated in the previous study⁴. Even though, this study⁴ investigated in the gap of the knowledge types (domain, technical, and application) in the GSD. The study expects that any effective gap in any one or both of the domain knowledge or the technical knowledge will affect directly of the creation of a gap in the application knowledge; According to what mentioned about the application knowledge ““application knowledge provides a “mapping” between domain knowledge and technical knowledge”^{24,4}. This expectation brought into

agreement with the results of this study. The aim of this study is to investigate first the presence of the gap in the application knowledge. And then if the real presence of the gap in the application knowledge will lead the study to expect a gap in both of domain and technical knowledge or in one of them at the least. And, if the study can prove.

4.2.2 The Impact of the Lack of using the Substitutions, that are Mitigate Challenges of both Communication and Coordination on the Existing Knowledge Gap

Here, the study presents the root-causes analysis results of the online survey and the results of Phase III: level 2 interviews and feedback. In fact, finding the main causes of the existing knowledge gap by using the root-causes analysis is essential to define the impact of the lack of using mitigate solutions which facilitate knowledge sharing and improve the interaction between distributed teams. Also, the results of Phase III: level 2 interviews and feedback have presented under an identification of emergent themes. This has developed after reviewing interviewees’ responses. Each interview question has presented one theme or more.

The lack of using practices/techniques/strategies, that mitigate challenges of both communication and coordination, negatively impact the domain and technical knowledge by creating root causes that responsible of the existing gap of the domain and technical knowledge. This impact had appeared when the study used root cause analysis to find the root causes of both the domain and technical knowledge. We will illustrate below five root causes and discuss each root cause.

4.2.2.1 The Lack of using Qualified Documentation

First, this study identifies the qualified document, which is standardized and contains: (1) test cases, and (2) user stories, is an effective technique for increasing collaboration between the distributed teams. The study’s results showed that this technique as an important key in the analysis and the design phase because it is the first key to understand the requirements business knowledge and has the ability to minimize the communication challenges¹⁴. Second, all the participants of the level 2 interviews& feedback confirmed the results of the survey, which were: (1) SRS document is not stan-

dard in containment test case and other details, and (2) epics are not efficient as user stories. Last, the study's results showed that, the impact of the lack of using qualified document on the existing domain knowledge gap increase the consume time of accomplish the tasks because of the decrease of the appropriate grasp of the business requirements which results of re-doing the work and a need for more explanations.

More details about this impact will be illustrated below in the next two paragraphs. Then, the agreements between the previous studies and this study, and the new addition of this study will be mentioned in the third paragraph.

4.2.2.1.1 Test Cases and Standardization

Based on the study's results, all the participants illustrated that: (1) they faced a major challenge in understanding what the client wants when test cases were lacked in the SRS document. They considered the test cases are basic to illustrate the business requirements and to increase their understandable for the requirements. Some participants of the study believe that the included of the entire available test cases in the SRS document are enough to understand the system requirements. (2) The SRS document sometimes may and may not contain test cases. And if it includes test cases, sometimes the available test cases are incomplete and just two short lines or not enough amount to gain an appropriate business knowledge. As a result, for the above-mentioned points (1) and (2) there is a need to standardize the contents of the SRS document which is essential to guarantee including the qualified test cases in SRS.

4.2.2.1.2 User Stories are more Efficient than Epics (Features Level)

All participants agreed that user stories are more efficient than epics (features level). They explained that there is an essential need for the SRS document to contain user stories when they were asked about if it is better to create user stories by the CET members instead of the designers and why. Their answers highlighted the importance of user stories. Four of the participants said, *'it is supposed to come to us as user stories.'* One participant said, *'it is a fundamental solution.'* Also, two participants said, *'it was one of the things that we requested.'* While the rest five participants used similar expressions such as, *'I think it is better because'*, *'it is an excellent idea because'*, *'it will*

be a good thing', and *'this is definitely a good thing.'* The study' results showed that containment the SRS document on the user stories as substitute to epics (features level) is more efficient according to the following points: (1) The User stories arrange and clarify the business requirements in an organized structure. Also, the CET is closer to the business environment than the developer team; which makes the user stories clearer and understandable from the business vision. And that increases the collaboration and decrease the rework by minimizing the misunderstanding of the business requirements that arising from the existence of the mediator between the client and the development team. According to what was illustrated by four participants. (2) The user stories are clear visions of the client' perception of what he/she (the client) wants. The clarity of their visions will decrease the rework because the client will directly involve in the creation of the user stories. And that was what three participants illustrated. Some participants also expanded that in the analysis and the designing phase, the designers put a number of perceptions to what the client wants in a document called a gap document. And sometimes their perceptions are being far from what the client' wants. As a result, they need additional explanation and to rewrite the gap document to include the right perception. That was what some of the participants explained about the inefficient of containment the SRS to epics wondering, *'why they can't avoid this rework by getting the right perception from the beginning.'* according to one participant. (3) User stories are better strategy to dispense with the more details required with epics. According to the study's results, three participants explained that they wouldn't need for more details with test cases if they get user stories instead of epics. As an explanation for why they think that the user stories are more efficient than epics. This more details have appeared as an essential need to the most of the participants when they were asked about (a) their perception of what are the minimum elements that SRS document need for being more qualified, and (b) the most frequent question designers ask the CET member They believed that the minimum elements that SRS document need for being more qualified is a clear and organized requirements with a business process flowchart for each requirement next to test cases. Furthermore, they explained their demand reason of the existing of a business process flowchart for each requirement next to test cases for increasing their understanding of the business requirements, because the requirements were often writ-

ten in a short description and a lot of business implicit in it and which are not clear to them. And added to that, they indicated to the business process flowchart for the requirements as an answer for the second question (b).

4.2.2.1.3 *The Previous Studies and the Results of this Study*

The results of this study seem to be in agreement with the previous studies^{9-11,13,14}. They considered documentation and standardization as useful techniques/practices for increase the collaboration between distributed and overcome the confusion between AGSD teams. Also, they highlighted the document which contains user stories and uses cases to minimize the misunderstanding. However, this study's results added a new addition to the previous studies when considered the qualified document, which is standardized in the containment of both test cases and user stories, as a valuable sharing technique for the domain knowledge which increases the business understandable. This addition was practically tested in the case that the client and the development team are distributed geographically and there is a mediator team, that is located geographically with the client, collects the business requirement from the client.

4.2.2.2 *The Lack of Recording Audio and Video Meetings which are being between the Development Team and CET*

Recording audio and video meetings between the distributed teams that are related to the meetings data is a useful technique to increase the cooperation by strengthening understanding as mentioned by^{20,21}. All the participants of the Phase III: level2_interviews& feedback denied that the meetings (video, voice) between them and CET team were recorded. This was confirmed by the results of the survey. According to the study's results, the Impact of the lack of recording audio and video meetings, which are being between the development team and CET, is focused on the loss of the local feedback that works on: (1) increasing the required knowledge of the business requirements to accomplish tasks, and (2) decreasing the consumed time in accomplishing tasks. More details about this impact will be illustrated below in the next two paragraphs. In the first paragraph, the participants' evaluation of the lost benefits due to the lack of the existence of the recorded meetings between the CET and the Egyptian team among the phase of analysis and design will be highlighted. In the second paragraph, the experience of some

of the study' participants for using this technique at local meetings will be illustrated.

4.2.2.2.1 *The Lost Benefits of the Lack of the Recorded Meetings*

The participants of this study highlighted the lost benefits due to the lack of the recorded meetings. These benefits focused on supporting the required understanding of the business requirements in a certain time for accomplishing tasks by the followings: (1) erasing any misunderstanding of the business requirements which could make a conflict on a certain point(s) after the meeting between the attendees, sometimes this conflict is small, but still it creates a need for an explanation to complete the task. The majority of the participants emphasized on the importance of having recorded meetings. As one of the participants explained, 'why we wait for an answer when we can have a local feedback?' However, a few of participants related the need of having recorded meetings to the complexity of the project. They believe that they don't need to record meetings unless the change request is a complex. Their explanation to that need was the possibility of occurrence of this conflict in the usual meeting is insignificant unless the meeting is longer than the usual ones and that happens in the complex project. During long meetings several business issues discussed and the possibility of forgetting important things at the end of the meetings actually happens. (2) Extracting needed knowledge to accomplish the certain update. The reason of why the recorded meetings work as data history is that not all details and implicit knowledge were being included in MOM or documents and that would make the recorded data as a reference for the new designers or anyone who need to make an update on an existing feature and he/she wasn't the one who created the task. That was what some of the participants explained about the benefits of the existence of the recorded meetings.

4.2.2.2.2 *The Experience of some of the Study' Participants for using Recorded Meetings*

Two of the participants commended the using of the recorded meetings technique at the local meetings as an excellent feedback.

4.2.2.3 *The Lack of the Co-Located of Development Team Members with CET*

The lack of physical colocation of one team together leads to the loss of both better trust among the distributed team

members and collaboration and the apparent of communication challenges. This lack can be mitigated by practicing various strategies of collocate practice as mentioned by^{13,14}. According to the study's results, the impact of the lack of using the co-located practice on the domain knowledge is the loss of a mechanism that increases the appropriate understanding of the business requirements and business environment. These results based on the participants' evaluation of the lost benefits due to the lack of the existence of the co-located practice.

The Study' Results Identified

Two kinds of the co-located practice. These two kinds are: (1) strategy1: CET member has come to Egypt and collocate with the designers during the analysis and design phase, (2) strategy2: the designers travel to collocate with CET members during the SRS document preparing. Also, three of the participants of the Phase III: level2_interviews& feedback confirmed that, the co-located practice has happened before, but in rare and specific situations.

The Study' Results Defined

The strategy 1 as a way to increases the understanding of the business requirements because the CET member could have the ability to bring the whole perspectives of both the client and the business environment, as it has been explained by five of the study' participants. Also, two of the five participants added that the strategy 1 increases trust and collaboration among team members by supporting the feeling of comfort and satisfaction among the team members about their work and their business requirements understanding.

The Study' Results Show That

Seven of the participants prefer strategy 2 than strategy 1 because they consider it as an excellent practice that gave them a chance to see the business environment and understand it. In addition, it is a chance to communicate directly with the client. This provides them a clear and whole vision of what the client wants. They explained their reasons of their preference in the following points: (1) part of their work challenge is that the client' he/she doesn't know what he/she really wants? And sometimes, they do the required feature in a way he/she doesn't prefer but it does the required tasks. The need to connect directly with the environment and the related persons is essential to avoid the rework and create accurate user stories. (2)

There is a lot of implicit knowledge that the client didn't tell because he/she didn't consider it important enough, but it is important to the designers and knowing it needs the designers to be close to the business environment and see how the business process functions in reality. (3) They need to have the same grasp of the business environment as the CET member. One participant illustrated, *'if I have the same grasp as the CET member, then the discussion would be worthy and I would get the required point in less time.'* Added to that, *'I would be more certain about the results.'* (4) They need to benefit from their business courses in a practical way. One participant illustrated, *'this practice is a chance for the person who takes the business course to practice their business knowledge in a real-life context in order to increase his/her domain knowledge. And, it is a chance for the company to know the benefit of being the person who prepares the SRS document has a technical knowledge.'*

4.2.2.4 The Lack of using Wikis (Asynchronous Tool) besides Email

The lack of face-to-face communication leads to the loss of both the implicit knowledge and experience which rapid the problems solving and increases the required understanding of the knowledge. This lack can mitigate by using wikis platform which works as a data history platform for sharing both of experience and knowledge. The impact of the lack of using wikis besides emails leads to the loss of certain valuable benefits of sharing both of experience and knowledge. Noting that sharing both experience and knowledge can positively; (1) increase the required understanding of the business requirements to accomplish task, and (2) decrease the consumed time to accomplish task. That was what has been illustrated by the results of the Phase III: level2_interviews& feedback.

According to, the Phase III: level2_interviews& feedback, all the participants didn't know any information about wiki or its benefits for AGSD. But, when the researcher explained in detail the benefits of using of wikis in AGSD as (1) check project state, (2) upgrade tasks list, (3) observe the work advance of the team members, and (4) share knowledge between distributed team members by sharing experiences and solutions for existent the problems, as mentioned by previous studies^{14,19}. As a result, ten of the participants considered these benefits valuable to save time and to increase the cooperation between them and the CET members which will positively reflect on understanding the business requirements and reduc-

ing the rework. Their summarized explanation was that if the designer has a problem with how doing a certain task or a misunderstanding about business requirements. The designer would search about it in the wiki. Then if the designer found his/her answer, that would save both of the sending time and waiting time for a respond from CET. If the designer didn't find, he/she can post his/her question on the wiki then other team members and CET members will notice his/her question and gave his/her an answer. But, two of the participants said they already have a remedy application which has the first three benefits of the wikis but if wiki has the ability to provide the sharing of experience and knowledge. Then, it would be an excellent reference for enhancing the knowledge and saving time. Also, one participant explained, *'in our work, the experience is an important key to accomplish the task; because the business environment of each region is similar in banks of certain type such as the Islamic banks which have similar processes. This makes the sharing of the experiences has valuable to us.'*

4.2.2.5 The Lack of the Yearly Planned Formal Training and Training Course

The existing of formal training, where it includes technical skills, and business knowledge, is helpful to enhance both the skills and knowledge of the team members¹⁶. Also, some studies recommended that the employees of the organization be subjected to weekly training courses¹⁴. Based on this study's results, yearly planned formal training and training courses are important keys to increase the domain and technical knowledge and recommended to schedule periodical training. This aligns with previous studies^{14,16}. More details about the impact of the lack of the yearly planned formal training and training courses on the domain and technical knowledge will be illustrated below in the next two paragraphs.

4.2.2.5.1 The Existing Domain Knowledge Gap

The impact of the lack of the yearly planned formal training and training courses on the domain knowledge especially with the loss of the direct communication between the development team and the client, is the loss of an excellent domain knowledge source. This domain knowledge source can increase the understanding of both business environment and process, and also illustrate the implicit issues that are gaining by the direct communication with the business environment and the

client. The study's results showed that the geographical distribution between the development team and the client leads to a lack of the understanding of the domain knowledge. The practical solution to increase the domain knowledge, is to adopt planned formal training courses by experienced trainers in the business practices as explained by some of the study's participants who had indicated that their experiment with business training courses by an experienced CET member or an experienced consultant in banks business is a very useful practice; which increases their domain knowledge and improves their work efficiency.

4.2.2.5.2 The Existing Technical Knowledge Gap

The study explored that the gap in the technical knowledge is a result of the lack of the yearly planned formal training. First, this lack has appeared in the amount of the training (Figures 6–8).



Figure 6. The Percentage of the amount levels of received training.



Figure 7. The Percentage of the efficient levels of received training.

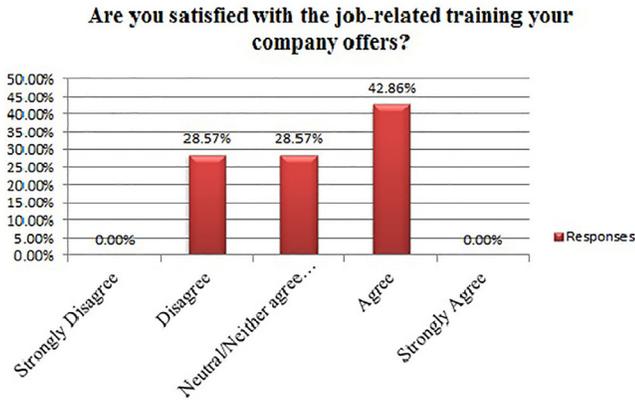


Figure 8. The Percentage of designers satisfaction levels with job-related training.

The Percentage of designers’ satisfaction levels with job-related training here). Second, the formal education of a real percentage of the designers has suffered from a weakness in a technical knowledge side which needs to be improved by the appropriate training (Figure 9). The Various levels of the designers’ formal education and Figure10. The Percentage of the Training efficiency levels received by the newcomers here). Third, the lack of formal training which related to the techniques that enhance the design phase has been appeared from the participants’ answering of the Phase III: level 2_ interviews& feedback when they were asked about the using of wikis. None of the participants known what wikis is. Also, they were not able to provide any information about using it in AGSD. Some of the online survey’s participants suggested the yearly planned training, which related to the last design pattern and the techniques that enhance the design phase, to improve the company training.

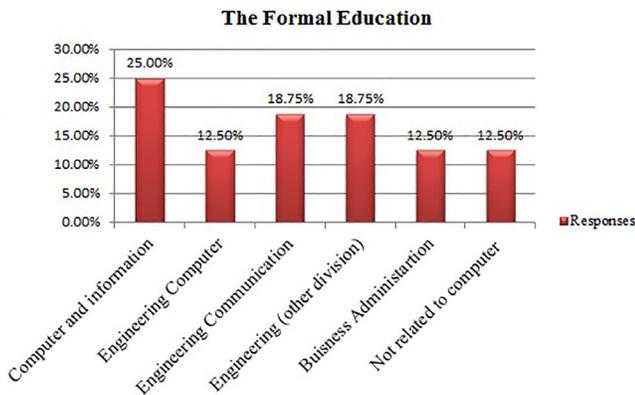


Figure 9. The Various level of the designers’ formal education.

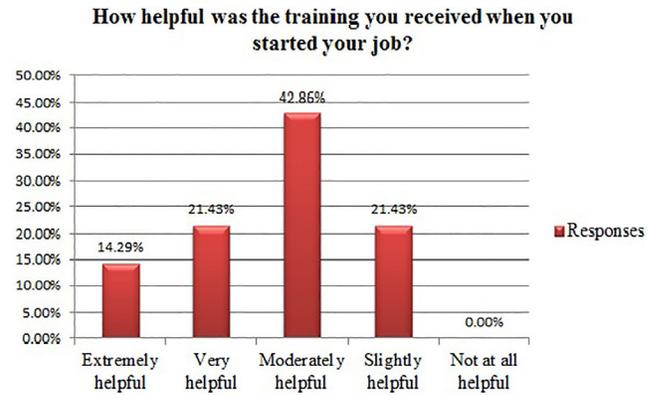


Figure 10. The Percentage of the Training efficiency levels received by the newcomers.

5. Conclusion and Future Scope

This paper has investigated knowledge types (domain, application, and technical) gaps in AGSD by studying the team’ structure of an Egyptian branch of an international company from the perspective of defining the existing gap in the knowledge types (domain, application, and technical) and the impact of the lack interaction between distributed teams on the existing knowledge gap. A quantitative and qualitative analysis is developed for the online survey and a general framework coding for a thematic analysis is used for the interviews, allowing the following conclusions to be drawn:

- The appropriate understanding of the domain knowledge leads to avoid the rework and save time.
- The periodically planned formal training has a direct proportion effect on the both the domain and the technical knowledge which both have in turn a direct impact on the application knowledge.
- The periodically planned formal training needed to be the first interest by the management of the software company when they seek to develop and increase the software development cycle efficiency.
- The qualified document needs to be considered and not be neglected because it is first key to gain the appropriate understanding of the business requirements.
- Recording audio and video meeting between distributed teams that are related to collected data is required for more customization in the future studies.
- The essential challenges which the CET (onshore team) was faced, is to pass their tacit knowledge of the requirements specification to the designers (offshore members). Therefore, there an essential need to the substitutions

that can mitigate the challenges of both communication and coordination, between the distributed parties.

For future research, the researchers are planning to build a framework based on the results and feedback of this case study and previous studies.

6. References

1. Sriram R, Mathew SK. Global software development using agile methodologies: A review of literature. Proceedings of the IEEE International Conference on Management of Innovation and Technology; 2012. p. 389–93. Crossref.
2. Ågerfalk J, Fitzgerald B, In OP. Flexible and distributed software processes: Old petunias in new bowls. Communications of the ACM. 2006; 49(10):27–34.
3. Dumitriu FL, Oprea DU, Mesnita GA. Issues and strategy for agile global software development adoption. Proceeding of the 3rd World Multi conference on Applied Economics, Business and Development; 2011. p. 37–42.
4. Aman A, Nicholson B. Mind the gap! Understanding knowledge in global software teams. Information Technology in the Service Economy: Challenges and Possibilities for the 21st Century; 2008. p. 321–30.
5. Aman A, Nicholson B. Knowledge management in globally distributed software development teams: Sensemaking challenges. Knowledge Management International Conference (KMICe); 2012. p. 63–8.
6. Prikladnicki R, Audy JL, Evaristo R. A reference model for global software development. Proceeding of 5th Working Conference on Virtual Enterprises; 2004 Aug. p. 369–78. Crossref.
7. Jalali S, Wohlin C. Agile practices in global software engineering-A systematic map. Proceeding of the 5th IEEE International Conference on Global Software Engineering (ICGSE); 2010 Aug. p. 45–54. Crossref.
8. Holmström H, Fitzgerald B, Ågerfalk PJ, Conchúir EÓ. Agile practices reduce distance in global software development. Information Systems Management. 2006 Dec; 23(3):7–18. Crossref.
9. Ramesh B, Cao L, Mohan K, Xu P. Can distributed software development be agile? Communications of the ACM. 2006 Oct; 49(10):41–6. Crossref.
10. Hossain E, Babar MA, Paik HY, Verner J. Risk identification and mitigation processes for using scrum in global software development: A conceptual framework. Proceedings of 16th Asia-Pacific Software Engineering Conference (APSEC'09); 2009 Dec. p. 457–64. Crossref.
11. Shrivastava SV, Date H. Distributed agile software development: A review. Journal of Computer Science and Engineering. 2010 May; 1(1):10–17.
12. Kamaruddin NK, Arshad NH, Mohamed A. Chaos issues on communication in agile global software development. Proceedings of the IEEE Business Engineering and Industrial Applications Colloquium (BEIAC); 2012. p. 394–8. Crossref.
13. Alzoubi YI, Gill AQ. Agile global software development communication challenges: A systematic review. Proceedings of the 18th Pacific Asia Conference on Information System; 2014. p. 1–13. PMID:25278769 PMCid:PMC4167487
14. Rizvi B, Bagheri E, Gasevic D. A systematic review of distributed Agile software engineering. Journal of Software: Evolution and Process. 2015 Oct; 27(10):723–62. Crossref.
15. Razzak MA, Ahmed R. Knowledge sharing in distributed agile projects: Techniques, strategies and challenges. IEEE Proceedings of the Federated Conference on Computer Science and Information Systems; 2014. p. 1431–40.
16. Dorairaj S, Noble J, Malik P. Knowledge management in distributed agile software development. IEEE Proceedings of the Agile Conference (AGILE); 2012 Aug. p. 64–73. Crossref.
17. Phalnikar R, Deshpande VS, Joshi SD. Applying agile principles for distributed software development. IEEE Proceedings of the International Conference on Advanced Computer Control (ICACC'09); 2009. p. 535–9. Crossref.
18. Manteli C, Van Den Hooff B, Tang A, Van Vliet H. The impact of multi-site software governance on knowledge management. Proceedings of the 6th IEEE International Conference on Global Software Engineering (ICGSE); 2012. p. 40–9.
19. Wang X, Maurer F, Morgan R, Oliveira J. Tools for supporting distributed agile project planning. Agility Across Time and Space. Berlin, Heidelberg: Springer; 2010 May. p. 183–99.
20. Hossain E, Bannerman PL, Jeffery DR. Scrum practices in global software development: a research framework. Proceedings of the 12th International Conference on Product Focused Software Process Improvement. Heidelberg: Springer; 2011. p. 88–102. Crossref.
21. Nilsson CG, Karlsson D. Implementing Agile project methods in globally distributed teams. Master Thesis, Project Management. Karlstad Business School; 2014. p. 1–81.
22. Yin RK. Case study research: Design and methods. 2nd Edition. USA: Sage Publications; 1994. p. 1–28. PMID:8026640
23. Lacey A, Luff D. Qualitative research analysis. The NIHR RDS for the East Midlands / Yorkshire and the Humber; 2009. p. 1–46.
24. Guarino N. Understanding, building and using ontologies. International Journal of Human-Computer Studies. 1997; 46(2–3):293–310. Crossref.