ISSN 1330-3651 (Print), ISSN 1848-6339 (Online) https://doi.org/10.17559/TV-20210103153044
Subject review

Software Engineer Competencies in Global Software Development: An Indonesian Perspective

Anita HIDAYATI, Eko K. BUDIARDJO*, Betty PURWANDARI

Abstract: Indonesia, as a country with great potential, should become an active Global Software Development (GSD) player. This study aims to identify and rank software engineers' competencies when working in a GSD environment. Having gathered competencies with a literature study, we then compiled the hierarchy of competencies, questionnaires and data processing using the Analytic Hierarchy Process. We involved Indonesia's experts in giving a portrait of the potential for global software engineers in Indonesia. From the research results, we obtained 40 competencies along with their ranking and clustering. We conclude that software engineers must master both hard and soft competencies by paying attention to the ranking and clustering generated in this study. This research is useful for practitioners, the Indonesian government and other countries to prioritise software engineer competencies. Further, it is useful for developed countries, as GSD clients, to see a portrait of software engineers' competence in Indonesia.

Keywords: Analytic Hierarchy Process (AHP); competencies; Global Software Development (GSD); software engineer

1 INTRODUCTION

Global Software Development (GSD) is a phenomenon that has lately been gaining more and more attention in the software industry. The concept of GSD became known in the early 1990s and continues to be a developing topic today [1]. Stakeholders from various countries have involvement in developing software using GSD [2]. GSD can be a means for all countries in their economic development. Currently, countries such as South Africa, Egypt, Mauritius, Ghana, India, China and the Philippines have an active role in GSD. Many factors are influential, including geographic location, common language, cultural compatibility, political stability, low cost of living and availability of skilled and cheap labour [3]. On the other hand, many countries have very high potential but cannot become active players in GSD.

MINT (Mexico, Indonesia, Nigeria, Turkey) is a group of countries that have the potential for rapid economic growth in the future, in terms of certain geographical, demographic and economic factors. When compared to Nigeria, which is also included in MINT, Indonesia's conditions are very different. Nigeria is not considered a potential country for GSD projects [3], but Indonesia has the potential to become an active player in GSD. Many data show the great potential of Indonesia. Indonesia is the world's fourth most populous country. With its large population, Indonesia will be ranked seventh in GDP by 2050. It was the 16th largest economy in the world in 2018 according to the World Bank GDP ranking. It is also a member of the G-20 economies, having the largest economy in Southeast Asia.

A report from The Kearney Global Services Location Index in 2019 shows that Indonesia remains in the 4th position out of 50 countries selected as attractive locations for offshore operations. Indonesia has shown a significant improvement in its political environment, which helps offset a slight decline in financial attractiveness, people skills and availability. Meanwhile, Indonesia is at the lowest level in the IMD World Digital Competitiveness Ranking 2020, namely 'emerging'. It gets a lower rating for human capital than its business environment, infrastructure and financial attractiveness.

From these data, we can conclude that Indonesia has a very high chance of being considered a global software development destination. However, some things still need to be improved, namely in terms of human resources. Indonesia needs to improve people skills, especially in software engineers, to compete with GSD players from other countries. Software engineers working in GSD need to have other competencies than traditional software development [1]. GSD faces many challenges, especially those related to human factors and their impact on software development [4]. One of the biggest challenges is maintaining quality in the development process and the resulting product [5]. They must be able to adapt and master the competencies required in GSD. Competency at the individual level describes individual characteristics to perform the work [6]. There has been some research discussing GSD and exploring competencies within the global spectrum. However, among case studies in Indonesia, there is no specific study discussing the portrait of Indonesia's potential in GSD in general and global software engineers in particular.

This study aims to compile a complete list of competencies, consisting of hard competencies and soft competencies. Furthermore, using the Analytical Hierarchy Process (AHP), we determined the competency ranking based on the level of importance. With a list of competencies and their resulting rankings, a software engineer will develop a strategy to master these when working in GSD. We involved GSD experts, competencies and software engineers in Indonesia because they can assess Indonesia's software engineers'actual conditions. The results of this research, however, apply not only to the Indonesian context. Other developing countries can use them to improve their software engineers. Besides this, the study will also educate GSD companies worldwide about Indonesian software engineers' current status and potency.

In the next section, we present our literature review to identify articles that have discussed topics related to this research. We present the research methodologies in Section 3 and results and discussion in Section 4. The last section contains conclusions and proposition for further research.

2 LITERATURE REVIEW

We searched for previous research that discussed competence in GSD. There has been some research discussing GSD and exploring competencies within the global spectrum. We need to consider many factors when working at GSD. One of the most important is human resources. The authors in [4] discussed the role of people in GSD. GSD itself is different from a collocated environment, meaning that there are automatically differences in the competencies needed. In [2] the authors discussed how competency changes and adapts when applied in GSD. The authors in [7] defined and implemented general and technical competency models specifically designed to face the challenges of working in a GSD team regarding SWEBOK, PMBOK, ACM Curricula, Career space, ANECA and VTManager Lifecycle.

Additionally, we also searched for articles from writers in Indonesia that discuss GSD because they are the ones who best understand the country's conditions. The two papers obtained discuss the capability of vendors in Indonesia in a global context. In [8] the authors evaluated IT vendors' capability level in Indonesia in supporting global IT projects in various aspects of capabilities using a new framework that adds adaptability. Meanwhile, the authors in [9] developed a framework that describes the development of offshore vendor capabilities during the client-vendor relationship stage. To simplify the description of our literature study results, we summarise these studies in Tab. 1. From Tab. 1 we can conclude that there is no research that focuses on discussing the competence of Indonesian software engineers in the GSD environment.

Table 1 Literature Review

Article	Summary
How software development competencies change in global settings-An explorative study [2]	Identified the factors influencing the adaptation competencies in GSD and the adaptation process
A discussion on the role of people in global software development [4]	Identified human factor challenges in GSD and proposed a solution to solve or mitigate them
Skills and abilities for working in a global software development team: Acompetence model Javier [7]	Defined and implemented a competence model specifically designed to address the challenges people face working in the GSD team
A framework for offshore vendor capability development [9]	Developed a framework to increase the Indonesia vendors' capability in global offshore outsourcing
Vendor capability for offshore IT projects: Analysing a case in Indonesian context [8]	Evaluated the capability level of Indonesian IT vendors in supporting offshore IT projects in various aspects of capabilities

3 RESEARCH METHODOLOGIES

Fig. 1 shows the seven steps of the methodology used to answer the research questions. We conducted a literature review to collect data about software engineers' competencies in a global environment from online databases. The data was processed using thematic analysis with open coding in the NVivo software. The result was a hierarchy of competencies. The third step was to compile a questionnaire containing a pairwise comparison matrix from the list of competencies using the Analytic Hierarchy Process (AHP) concept [10]. Before distributing the questionnaire to the experts, we carried out a readability test aiming for the contents to be consistent, error-free and easy to read.

Revisions were made based on input from the questionnaire examiners. Once ready, the questionnaire was given to the experts to fill out. The results were analysed using AHP formulas to produce a ranking from the list of competencies [11]. The final step was to derive conclusions from all the data that had been successfully processed and collected.

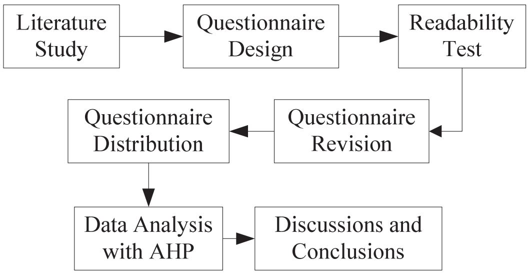

Figure 1 Research flow

The first step was a literature review to obtain all articles that cover the competencies needed by a global software engineer. We used library automation and the digital archive of Universitas Indonesia. We took advantage of Summon's Discovery Federated Search menu, a meta-search engine that can perform article retrieval from several online databases subscribed to by UI Libraries, such as Emerald, Ebsco, Science Direct, Proquest, IEEE and Springerlink. The literature was taken from the last five years using the following terms:
("global software development" orgsd or "global software engineering" or gse or "distributed software development") and (competen*)

Tab. 2 shows the stages and results of the literature review.

Table 2 Data sources

Phase	Results
Beginning	6,693
Publication date (5 years prior)	2,647
Content-type (journal and proceeding)	1,318
Discipline (Computer Science)	84
Language (English)	84
Abstract	27

Of the 28 papers that were among the last results from screenings, we identified competencies, grouping those competencies based on their proximity using thematic analysis. Fig. 2 presents a hierarchy of competence. We refer to eight competencies as criteria, consisting of four hard competencies and four soft competencies. The four hard competencies are Programming Techniques (C1), Requirements Techniques (C2), Verification and Validation (C3), and Collaboration Techniques (C4). In comparison, the four soft competencies are Culture (C5) Project Management and Leadership (C6), Communication and Coordination (C7), Collaboration and Knowledge Management (C8). Each criterion has a set of sub-criteria. Comparisons occur at the criteria and sub-criteria levels. Details for each of the sub-criteria are shown in Tab. 3.

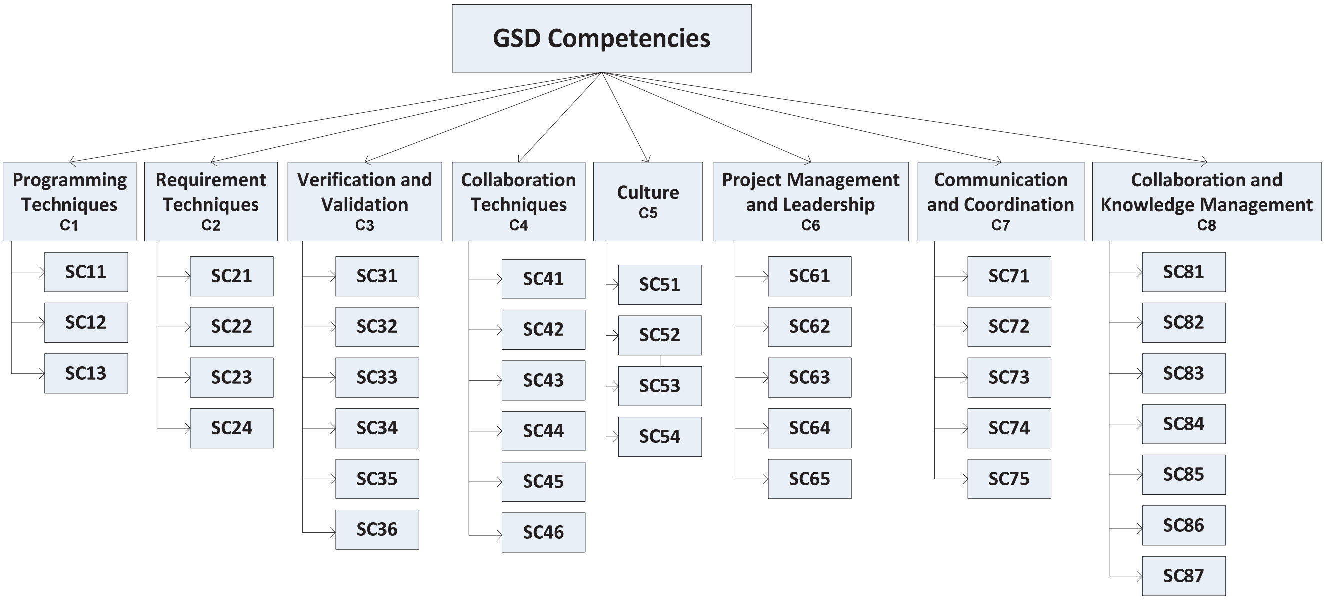
GSD Competencies
Programming Techniques C1
SC11
SC12
SC13
Requirement Techniques C2
SC21
SC22
SC23
SC24
Verification and Validation C3
SC31
SC32
SC33
SC34
SC35
SC36
Collaboration Techniques C4
SC41
SC42
SC43
SC44
SC45
SC46
Culture C5
SC51
SC52
SC53
SC54
Project Management and Leadership C6
SC61
SC62
SC63
SC64
SC65
Communication and Coordination C7
SC71
SC72
SC73
SC74
SC75
Collaboration and Knowledge Management C8
SC81
SC82
SC83
SC84
SC85
SC86
SC87

Figure 2 Hierarchy model for ranking competencies of global software engineers

Table 3 Criteria and Sub-criteria

Criteria	Sub-criteria
C1. Programming Techniques (hard)	SC11. Ability to use social coding sites that facilitate collaborative development [12]
	SC12. Ability to build software applications by reusing existing components (CBS) [13-15]
	SC13. Ability to adopt new programming languages and tools [16]
C2. Requirement Techniques (hard)	SC21. Ability to use typical and advanced CRM applications, tools and standards on the GSD site [17]
	SC22. Ability to build tools and technology for engineering process needs [17]
	SC23. Ability to establish mechanisms for analysing the impact of changing needs on distributed sites [17-19]
	SC24. Ability to collect, formalise and validate geographically dispersed stakeholder and business needs [20]
C3. Verification and Validation (hard)	SC31. Continuous programming capabilities via pair programming [20]
	SC32. Ability to maintain detailed artifacts to avoid misunderstanding between sites [20]
	SC33. Ability to add code inspections or test when coordinating software development work [21]
	SC34. Ability to use testing automated tool support [22]
	SC35. Ability to evaluate results based on test cases and product specifications [23]
	SC36. Ability to use rigorous unit and integration testing [24]
C4. Collaboration Techniques (hard)	SC41. Ability to support synchronous and asynchronous communication technologies [7, 25]
	SC42. Ability to design a system backup and recovery [26]
	SC43. Ability to master advanced collaboration tools/techniques in distributed communication [7, 27-29]
	SC44. Ability to maintain privacy and security [25]
	SC45. Ability to perform data integration among various tools [29, 25]
	SC46. Ability to integrate with issue trackers or other platforms [16, 25]
C5. Culture (soft)	SC51. Ability to understand cultural influences and implications in work-life [2]
	SC52. Ability to assess practices and products from multiple cultural perspectives [2, 30 ,4]
	C53. Ability to work and adapt to different cultures and to overcome interpersonal conflicts [7, 30]
	SC54. Can trust the team and earn trust [2, 31, 32]
C6. Project Management and Leadership (soft)	SC61. Ability to adapt strategies, goals and plans according to the situation [2]
	SC62. Ability to manage diversity in teams so that everyone can contribute and participate [2, 33]
	SC63. Ability to earn trust from the team [2, 33]
	SC64. Ability to influence and direct a team [33]
	SC65. Ability to think critically and innovatively and have a commitment and willingness to work hard [34]
C7. Communication and Coordination (soft)	SC71. Ability to use and organise information and communication technology in a global context [7, 35, 16]
	SC72. Ability to communicate with teams of different cultures and personalities [7, 2, 35, 28]
	SC73. Ability to build, develop and maintain social protocols in a distributed team [36]
	SC74. Ability to hear and consider the thoughts of others [7, 30]
	SC75. Ability to communicate through writing and conversation that is clear and fluent in a foreign languages [32, 28]
C8. Collaboration and Knowledge Management (soft)	SC81. Ability to collaborate with multiple teams in different time zones [35, 31, 28]
	SC82. Ability to share information and knowledge in teams [7, 2, 37, 38, 18]
	SC83. Ability to organise meetings and brainstorming sessions [7]
	SC84. Ability to understand the perspectives, needs and values of others [35]
	SC85. Ability to use other people's skills and knowledge [2]
	SC86. Ability to collaborate on problem-solving [37, 38] [2, 33]
	SC87. Ability to set rules for working with separate data [7]

We compiled a questionnaire from the literature review results regarding the hierarchy of competencies presented in Fig. 2. The questionnaire consisted of two parts, with the first part containing the identity data of the respondents. Tab. 4 shows the details of the data from the expert respondents. In the second part, the respondents must compare the criteria or sub-criteria on the left (A) and the right (B) related to GSD competencies based on Indonesia's actual conditions, using a pairwise comparison form. If A's sub-criteria or criteria are considered superior, the expert chooses the left side column.

Conversely, if B's sub-criteria or criteria are considered more important, the expert chooses the right side. There are ten comparison tables. Eight tables contain a comparison of each sub-criteria, and one table includes a comparison of nine criteria. Respondents must choose one of the values in the column provided. Tab. 5 shows the value scale for comparison. Tab. 6 is an example of a filling scale in the questionnaire.

Table 4 Expert data

Company	Position	Experience
A	Chief Software Scientist	GSD, Software Engineering
B	Director	Software Engineering
C	CEO	Software Engineering
D	GM Training Centre	GSD, Competencies
E	CEO	Software Engineering

Table 5 Scale for pairwise comparison

Comparison	Left Side	Right Side
1	A and B have equal importance	
3	A is slightly more important than B	B is slightly more important than A
5	A is more important than B	B is more important than A
7	A is clearly more important than B	B is clearly more important than A
9	A is absolutely more important than B	B is absolutely more important than A
2, 4, 6, 8	Intermediate values, when compromise is needed	

Table 6 Comparison in the questionnaire

A	9	8	7	6	5	4	3	2	1	2	3	4	5	6	7	8	9	B
Criteria or Sub-criteria																		Criteria or Sub-criteria

Table 7 Matrix comparison

A	9	8	7	6	5	4	3	2	1	2	3	4	5	6	7	8	9	B
C1																		C2
																		C3
																		C4
																		C5
																		C6
																		C7
																		C8
C2																		C3
																		C4
																		C5
																		C6
																		C7
																		C8
C3																		C4
																		C5
																		C6
																		C7
																		C8
C4																		C5
																		C6
																		C7
																		C8
C5																		C6
																		C7
																		C8
C6																		C7
																		C8
C7																		C8

Before distributing the questionnaire to the respondents, we carried out a legibility test involving several laboratory researchers. This step was to ensure that the questionnaire was user-friendly and readable. In terms of content, it does not contain errors: it is easy to understand, it is not confusing, or it does not contain multiple meanings. Based on suggestions and input, we made improvements in terms of presentation and content.

We distributed questionnaires to five experts in software engineering, competency and GSD. We also assisted in the process of experts filling in questionnaires so that there were no misperceptions-this step also maintained the experts' focus and consistency, as the questionnaire was extensive, consisting of 20 pages. As an illustration of the questionnaire compiled, a comparison matrix of criteria is shown in Tab. 7.

4 RESULTS AND DISCUSSIONS

The questionnaire data were processed using AHP [39]. The priority results of five experts were combined using the geometric mean. *GM* is calculated directly by using Eq. (1), where *GM* is the Mean Geometry and *X* is the experts.

$$GM = \sqrt[n]{(X_1)(X_2), ..., (X_n)} \tag{1}$$

We processed the results of the completed questionnaire into nine pairwise comparison matrix tables. The eight tables are the comparison of the sub-criteria for each of the eight criteria in Fig. 2. Each criterion has several sub-criteria that are not the same, so that the matrix dimensions may vary. One more table compares the eight ones, which results in an 8 × 8 matrix.

The next step is to determine the priority vectors by normalising the eigenvector of the matrix. Priority vectors contain the weight of each matrix element. The final step is to test the consistency by using the Consistency Ratio (*CR*), which is a comparison between the Consistency Index (*CI*) of an n x n matrix and the Random Consistency Index (RI). Tab. 8 contains *RI* values. If $CR \leq 0.1$, a matrix is considered consistent. If $CR \leq 0.1$, a matrix is considered consistent. We consider *CR* as Eq. (2).

$$CR = \frac{CI}{RI} \tag{2}$$

Table 8 *RI* values

N	1	2	3	4	5	6	7	8	9
RI	0	0	0.58	0.9	1.12	1.24	1.32	1.41	1.45

CI is retrieved by using the formula in Eq. (3). $\lambda_{\max}$ is the maximum eigenvalue of the matrix.

$$CI = \frac{\lambda_{\max} - n}{n - 1} \tag{3}$$

We performed the above steps for each of the comparison tables. The consistency test results from nine pairwise comparison matrix tables all produce a Consistency Ratio (*CR*) value of less than 10%. This result shows that the answers from all experts are consistent. Tab. 9 shows the calculation results and the priority. The weight of each criterion and sub-criteria are in the priority column.

Table 9 Pairwise comparison for criteria and sub-criteria

Overall Goal (G)	C1	C2	C3	C4	C5	C6	C7	C8	Priority
C1	1	2.69	1.23	0.49	0.58	2.34	2.71	2.27	0.14197
C2		1	0.37	0.24	0.54	0.90	1.38	1.00	0.13997
C3			1	0.21	0.69	1.84	1.84	1.00	0.14175
C4				1	0.83	3.64	3.48	3.16	0.14718
C5					1	2.55	1.90	1.64	0.14462
C6						1	0.92	1.00	0.14224
C7							1	0.41	0.14220
C8								1	0.14205
Inconsistency Ratio: 0.03									
C1. Programming Techniques	SC11			SC12			SC13		Priority
SC11	1			3.11			3.66		0.3366
SC12				1			0.68		0.3313
SC13							1		0.3321
Inconsistency Ratio: 0.03									
C2. Requirements Techniques	SC21		SC22		SC23		SC244		Priority
SC21	1		0.29		0.47		1.48		0.2494
SC22			1		0.99		3.45		0.2521
SC23					1		1.81		0.2511
SC24							1		0.2474
Inconsistency Ratio: 0.03									
C3. Verification and Validation	SC31	SC32	SC33	SC34	SC35		SC36		Priority
SC31	1	0.37	0.77	0.41	0.56		0.77		0.1636
SC32		1	0.71	0.51	1.08		1.20		0.1646
SC33			1	0.47	1.55		1.43		0.1679
SC34				1	2.85		1.93		0.1666
SC35					1		1.84		0.1679
SC36							1		0.1693
Inconsistency Ratio: 0.07									
C4. Collaboration Techniques	SC41	SC42	SC43	SC44	SC45		SC46		Priority
SC41	1	0.25	0.59	0.23	1.81		2.91		0.1651
SC42		1	1.11	1.68	1.55		2.29		0.1744
SC43			1	0.72	1.67		2.50		0.1663
SC44				1	3.63		4.43		0.1717
SC45					1		1.00		0.1614
SC46							1		0.1613
Inconsistency Ratio: 0.06									
C5. Culture	SC51		SC52		SC53		SC54		Priority
SC51	1		0.68		4.99		4.66		0.2532
SC52			1		3.16		3.06		0.2538
SC53					1		1.25		0.2466
SC54							1		0.2464
Inconsistency Ratio: 0.06									
C6. Project Management and Leadership	SC61	SC62	SC63		SC64		SC65		Priority
SC61	1	0.786	0.678		1.070		2.048		0.2026
SC62		1	0.582		0.644		0.582		0.2012
SC63			1		1.000		0.803		0.1999
SC64					1		0.803		0.1988
SC65							1		0.1975
Inconsistency Ratio: 0.05									
C7. Communication and Coordination	SC71	SC72	SC73		SC74		SC75		Priority
SC71	1	1.201	1.380		0.803		1.563		0.20014
SC72		1	0.803		1.000		1.476		0.19990
SC73			1		1.185		1.476		0.19980
SC74					1		1.052		0.20024
SC75							1		0.19993
Inconsistency Ratio: 0.01									
C8. Collaboration and Knowledge Management	SC81	SC82	SC83	SC84	SC85	SC86	SC87		Priority
SC81	1	1.838	1.933	3.519	3.707	2.371	1.821		0.1439
SC82		1	2.408	3.160	3.160	2.290	1.552		0.1434
SC83			1	1.316	1.246	1.000	0.644		0.1426
SC84				1	0.778	0.699	0.422		0.1429
SC85					1	0.412	0.582		0.1421
SC86						1	1.431		0.1430
SC87							1		0.1420
Inconsistency Ratio: 0.02									

The next step was to calculate the global priority weight by synthesising all the criteria and sub-criteria local priorities in Tab. 9. We then multiplied the original weight column by the local weight to produce global weight. The overall rank column contains the order for the overall global weight of the sub-criteria. The results are in Tab. 10. The local rank column contains the ranking for each sub-criterion within one criterion. At the same time, the overall rank applies to all sub-criteria.

We can see that the ranking order is spread across all criteria. It is difficult for us to draw conclusions only from Tab. 8. Therefore, we sorted the sub-criteria from the highest to the lowest importance. We also performed clustering to obtain grouping in Fig. 3. Fig. 3 shows that there are five clusters. The number of competencies for each cluster is different. Cluster 1 is the most important competency, and Cluster 5 is the least important one.

Table 10 Rank result

Criteria	Original Weight	Sub-criterion	Local Weight	Local Rank	Global Weight	Overall Rank
C1	0.14197	SC11	0.3366	1	0.04780	1
		SC12	0.3313	3	0.04704	3
		SC13	0.3321	2	0.04716	2
C2	0.13997	SC21	0.2494	3	0.03492	10
		SC22	0.2521	1	0.03529	8
		SC23	0.2511	2	0.03515	9
		SC24	0.2474	4	0.03464	11
C3	0.14175	SC31	0.1636	6	0.02320	33
		SC32	0.1646	5	0.02334	32
		SC33	0.1679	2	0.02381	27
		SC34	0.1666	4	0.02362	31
		SC35	0.1679	3	0.02381	28
		SC36	0.1693	1	0.02401	26
C4	0.14718	SC41	0.1651	4	0.02430	25
		SC42	0.1744	1	0.02567	22
		SC43	0.1663	3	0.02448	24
		SC44	0.1717	2	0.02527	23
		SC45	0.1614	5	0.02376	29
		SC46	0.1613	6	0.02374	30
C5	0.14462	SC51	0.2532	2	0.03661	5
		SC52	0.2538	1	0.03670	4
		SC53	0.2466	3	0.03566	6
		SC54	0.2464	4	0.03563	7
C6	0.14224	SC61	0.2026	1	0.02881	12
		SC62	0.2012	2	0.02861	13
		SC63	0.1999	3	0.02843	17
		SC64	0.1988	4	0.02827	20
		SC65	0.1975	5	0.02808	21
C7	0.14220	SC71	0.2001	2	0.02845	15
		SC72	0.1999	4	0.02843	18
		SC73	0.1998	5	0.02841	19
		SC74	0.2002	1	0.02847	14
		SC75	0.1999	3	0.02843	16
C8	0.14205	SC81	0.1439	1	0.02043	34
		SC82	0.1434	2	0.02036	35
		SC83	0.1426	5	0.02025	36
		SC84	0.1429	4	0.02029	37
		SC85	0.1421	6	0.02018	39
		SC86	0.143	3	0.02031	38
		SC87	0.142	7	0.02016	40

Cluster 1 contains the 'Programming Technique' competencies, with all three sub-criteria in the top three. 'Ability to use social coding sites that facilitate collaborative development' is the highest priority, followed by 'Ability to adopt new programming languages and tools' and 'Ability to build software applications by reusing existing components (Component-based Software) '.This follows the general opinion that a software engineer who has qualifying skills will adapt to new environments with a broader scope quickly.

Cluster 2 contains all the competencies of 'Culture' and 'Requirement Technique'. It shows that soft skills in dealing with cultural differences in GSD are fundamental. Understanding and assessment from multiple cultural perspectives are required. In addition, the ability to resolve conflicts and foster trust in a diverse team is crucial. GSD, which is very dynamic, requires high technical requirements. That is why GSD actors are expected to use and build applications and tools to handle stakeholders' business needs from various sites. They should also be able to establish mechanisms for analysing the changes that have occurred.

Cluster 3 contains competencies from 'Project Management and Leadership' and 'Communication and Coordination', which are soft competencies. In GSD, these competencies are essential. Team members must be able to understand, listen to, and trust other team members. Therefore, the ability to communicate and coordinate is vital. They must also be able to think critically and innovatively in managing projects with dynamic global situations.

Cluster 4 is more about 'Collaboration Techniques'. GSD actors must master backup and recovery; privacy and security; data integration; collaboration tools and techniques; and communication techniques. Additionally, they have to master the 'Verification and Validation' of the system, such as performing various tests and analysing the results. The lowest ranked is 'Collaboration and Knowledge Management'. The competencies here are in the lowest rank because of the competencies in the upper clusters being mastered. It will be easy to master Cluster 5.

Overall, we can see those hard competencies are ranked first, third, sixth and seventh. Meanwhile, soft competencies are ranked second, fourth, fifth and eighth. This means nothing dominates. Although hard competencies are ranked first, they are followed directly by soft competencies. However, we cannot ignore soft competencies, which also have a critical ranking. The two complement each other.

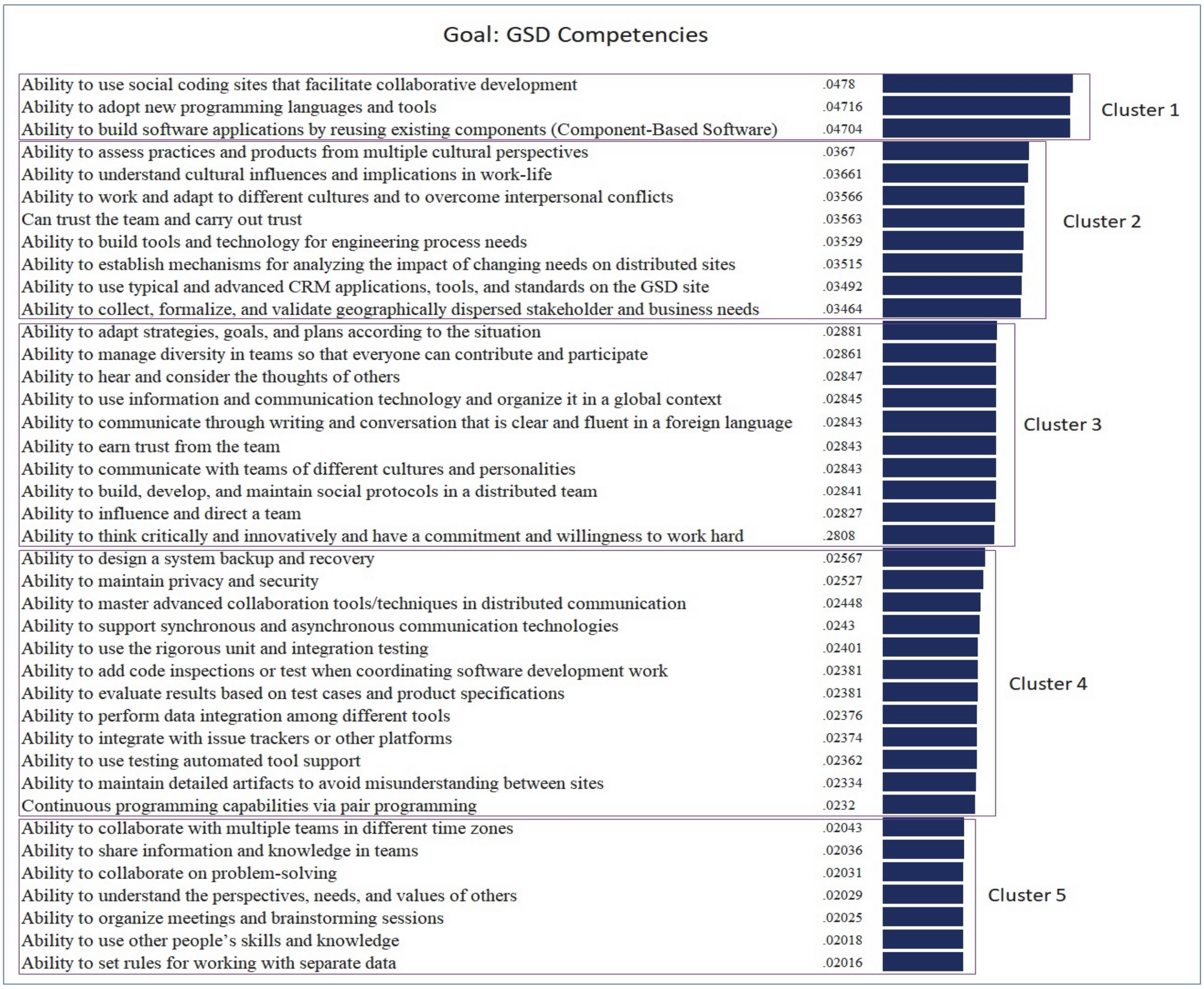

Figure 3 Global rank of global software engineers' competencies

When we relate the results obtained to Indonesia's actual conditions, the global software engineers in this country are already superior in soft competencies. They have mastered 'Collaboration' in terms of hard and soft competencies. For 'Culture', 'some things still need to be improved regarding trust and interpersonal conflict. The soft competencies that still need considerations are 'Communication and Coordination' and 'Project Management and Leadership'. Meanwhile, Indonesian global software engineers' most significant weaknesses fall into the category of hard competencies, especially in 'Programming Techniques', followed by 'Verification and Validation' and 'Requirement Techniques'.

For developed countries, the results of this research can be used as the basis for identifying Indonesia as one of the countries capable of providing qualified global software engineers. The results will provide a picture of the condition of Indonesia, which has so far been less exposed to its potential in the GSD.

5 CONCLUSIONS

This research identified and ranked software engineers' competencies when working in a GSD environment, based on their importance, by examining Indonesia's actual conditions. Findings from this research can be a reference for all interested parties in Indonesia and similar countries to determine mastery of GSD competencies. The summary of the findings is presented in Fig. 3.

We extracted the academic databases' hard and soft competencies through a literature review, followed by distributing a questionnaire to experts to validate the findings. Obtaining 40 competencies, we grouped them into four criteria for hard competency and four criteria for soft competency. We used AHP to compile a hierarchy of competencies and processed data from the completed questionnaires to obtain an order of importance. We then grouped them into 5 clusters. At a glance, it is clear that hard competencies occupy the first cluster, while soft competencies are in the lowest cluster. However, after a more in-depth analysis, it was found that global software engineers must master both hard competencies and soft competencies.

This study's results can guide the self-evaluation of global software engineers in Indonesia and what competencies they need to improve on to succeed in the GSD environment. For the steps they have to take, reference can be made to the global rank of global software engineers' competencies in Fig. 3. The government of Indonesia-along with other developing countries with similar conditions-can list competencies and their rankings as a reference to assess the condition of their software engineer resources and formulate a strategic policy for increasing competence.

For further research, we can enrich the competencies by integrating additional databases. We can involve experts from several countries or continents to get a broader perspective. Further, to meet the software industry trend towards Agile, we will continue this research by modelling the Scrum team's competencies in the GSD environment.

Acknowledgments

The authors would like to thank Universitas Indonesia for funding this research through the PUTI Grant, contract number NKB-3232/UN2.RST/HKP.05.00/2020.

6 REFERENCES

[1] Beecham, S., Clear, T., Damian, D., Barr, J., Noll, J., & Scacchi, W. (2017). How best to teach global software engineering? Educators are divided. *IEEE Software*, 16-19. https://doi.org/10.1109/MS.2017.12

[2] Holtkamp, P., Lau, I., & Pawlowski, J. M. (2015). How software development competences change in global setting-An explorative study. *Journal of Software: Evolution and Process*, *27*, 50-72. https://doi.org/10.1002/smr.1701

[3] Casado-Lumbreras, C., Colomo-Palacios, R., Francisca, N., & Misra, S. (2014). Software development outsourcing: Challenges and opportunities in nigeria. *Journal of Global Information Technology*, *17*(2014), 267-282. https://doi.org/10.1080/1097198X.2014.978626

[4] Misra, S., Colomo-Palacios, R., Pusatli, T., & Soto-Acosta, P. (2013). A discussion on the role of people in global software development. *Tehnicki Vjesnik*, *20*(3), 525-531.

[5] Misra, S., & Fernández-sanz, L. (2011). Quality issues in global software development. *ICSEA 2011: The Sixth International Conference on Software Engineering Advances*, (c), 325-330.

[6] McClelland, D. C. (1973). Testing for competence rather than for "intelligence. " *The American psychologist*, *28*(1), 1-14. https://doi.org/10.1037/h0034092

[7] Saldaña-ramos, J., Sanz-esteban, A., García, J., & Amescua, A. (2014). Skills and abilities for working in a global software development team: A competence model. *Journal of Software: Evolution and Process*, *26*, 329-338. https://doi.org/10.1002/smr.1588

[8] Govindaraju, R. & Zairul, R. A. (2014). Vendor capability for offshore it projects: Analysing a case in indonesian context. *Proceedings of 2014 International Conference on Data and Software Engineering (ICODSE 2014)*. https://doi.org/10.1109/ICODSE.2014.7062708

[9] Govindaraju, R. & Irianto, D. (2016). A framework for offshore vendor capability development. *IOP Conference Series: Materials Science and Engineering*. https://doi.org/10.1088/1757-899X/114/1/012078

[10] Wu, Z., Jin, B., Fujita, H., & Xu, J. (2020). Consensus analysis for ahp multiplicative preference relations based on consistency control: A heuristic approach. *Knowledge-Based Systems*, *191*. https://doi.org/10.1016/j.knosys.2019.105317

[11] Mahmudova, S. & Jabrailova, Z. (2020). Development of an algorithm using the AHP method for selecting software according to its functionality. *Soft Computing*, *24*(11), 8495-8502. https://doi.org/10.1007/s00500-020-04902-y

[12] Lu, Y., Mao, X., Wang, T., Yin, G., & Li, Z. (2019). Improving students' programming quality with the continuous inspection process: A social coding perspective. *Frontiers of Computer Science*, *14*(5). https://doi.org/10.1007/s11704-019-9023-2

[13] Ilyas, M. & Khan, S. U. (2017). Software integration in global software development: Challenges for GSD vendors. *Journal of Software: Evolution and Process*, *29*(8), 1-17. https://doi.org/10.1002/smr.1875

[14] Ilyas, M., Khan, S. U., & Rashid, N. (2020). Empirical validation of software integration practices in global software development. *SN Computer Science*, *1*(3), 1-23. https://doi.org/10.1007/s42979-020-00175-2

[15] Khan, M. J. & Mahmood, S. (2018). Assessing the determinants of adopting component-based development in a global context: A client-vendor analysis. *IEEE Access*, *6*, 79060-79073. https://doi.org/10.1109/ACCESS.2018.2878798

[16] Kroll, J., Santos, C. Q., MacHado, L. S., Marczak, S., & Prikladnicki, R. (2016). Challenges and lessons learned on preparing graduate students for GSE work: Brazilians' perceptions on a multi-site course experience. *Proceedings - 11th IEEE International Conference on Global Software Engineering Workshop, ICGSEW 2016*, 37-42. https://doi.org/10.1109/ICGSEW.2016.17

[17] Akbar, M. A., Sang, J., Nasrullah, Khan, A. A., Shafiq, M., & Fazal-E-Amin. (2019). Towards the guidelines for requirements change management in global software development: Client-vendor perspective. *IEEE Access*, *7*, 76985-77007. https://doi.org/10.1109/ACCESS.2019.2918552

[18] Ali, N. & Lai, R. (2017). A method of software requirements specification and validation for global software development. *Requirements Engineering*, *22*(2), 191-214. https://doi.org/10.1007/s00766-015-0240-4

[19] Alsanad, A. A., Chikh, A., & Mirza, A. (2019). A domain ontology for software requirements change management in global software development environment. *IEEE Access*, *7*, 49352-49361. https://doi.org/10.1109/ACCESS.2019.2909839

[20] Zahedi, M., Shahin, M., & Ali Babar, M. (2016). A systematic review of knowledge sharing challenges and practices in global software development. *International Journal of Information Management*, *36*(6), 995-1019. https://doi.org/10.1016/j.ijinfomgt.2016.06.007

[21] Matthiesen, S. (2017). When distribution of tasks and skills are fundamentally problematic: A failure story from global software outsourcing. *PACM on Human-Computer Interaction*, *1*(November), 1-16. https://doi.org/10.1145/3139336

[22] Fernández-alemán, J. L., Carrillo-de-gea, J. M., Meca, J. V., Ros, J. N., Toval, A., & Idri, A. (2016). Effects of using requirements catalogs on effectiveness and productivity of requirements specification in a software project management course. *IEEE Transaction on Education*, *59*(2), 105-118. https://doi.org/10.1109/TE.2015.2454472

[23] Salger, F. & Engels, G. (2010). Knowledge transfer in global software development: Leveraging acceptance test case specifications, *ICSE '10: Proceedings of the 32nd ACM/IEEE International Conference on Software Engineering*, *2*, 211-214 https://doi.org/10.1145/1810295.1810332

[24] Zafar, A. A., Saif, S., Khan, M., Iqbal, J., Akhunzada, A., Wadood, A., & Alamri, A. (2017). Taxonomy of factors causing integration failure during global software development. *IEEE Access*, *6*, 22228-22239. https://doi.org/10.1109/ACCESS.2017.2782843

[25] Usmani, N., Hassan, R., & Mahmood, W. (2017). Impediments to requirement engineering in distributed team. *International Journal of Information Engineering and Electronic Business*, *9*(6), 10-18. https://doi.org/10.5815/ijieeb.2017.06.02

[26] Ebert, C. (2013). Global software engineering: lessons from industry. In *2013 IEEE 8th International Conference on Global Software Engineering Workshops, ICGSEW 2013*, 49-73. https://doi.org/10.1109/ICGSEW.2013.15

[27] Castro-Hernández, A., Swigger, K., & Ponce-Flores, M. P. (2015). Effects of cohesion-based feedback on the collaborations in global software development teams. *EAI*

Endorsed Transactions on Collaborative Computing, *1*(6), 150818. https://doi.org/10.4108/eai.21-12-2015.150818

[28] Hjelsvold, R. & Mishra, D. (2019). Exploring and expanding gse education with open source software development. *ACM Transactions on Computing Education*, *19*(2). https://doi.org/10.1145/3230012

[29] Datta, S. (2018). How does developer interaction relate to software quality? An examination of product development data. *Empirical Software Engineering*, *23*. https://doi.org/10.1007/s10664-017-9534-0

[30] Frezza, S., Daniels, M., & Wilkin, A. (2019). Assessing students' it professional values in a global project setting. *ACM Transactions on Computing Education*, *19*(2), 1-34. https://doi.org/10.1145/3231710

[31] Vizcaíno, A., García, F., De Guzmán, I. G. R., & Moraga, M. Á. (2019). Evaluating GSD-aware: A serious game for discovering global software development challenges. *ACM Transactions on Computing Education*, *19*(2), 1-23. https://doi.org/10.1145/3218279

[32] Sharma, N., Singh, A. S., & Singh, K. (2015). Agile in global software engineering: An exploratory experience. *International Journal of Agile Systems and Management*, *8*(1), 23-38. https://doi.org/10.1504/IJASM.2015.068607

[33] Niazi, M., Mahmood, S., Alshayeb, M., Qureshi, A. M., Faisal, K., & Cerpa, N. (2016). Toward successful project management in global software development. *International Journal of Project Management*, *34*(8), 1553-1567. https://doi.org/10.1016/j.ijproman.2016.08.008

[34] Noll, J. & Beecham, S. (2017). Motivation and autonomy in global software development: An empirical study. *EASE'17: Proceedings of the 21st International Conference on Evaluation and Assessment in Software Engineering*, 394-399. https://doi.org/10.1145/3084226.3084277

[35] Monasor, M. J., Vizcaíno, A., Piattini, M., Caballero, I., Universidad, P. De, Real, C., … Piattini, M. (2010). Preparing students and engineers for global software development: A systematic review. In *International Conference on Global Software Engineering*. https://doi.org/10.1109/ICGSE.2010.28

[36] Giuffrida, R. & Dittrich, Y. (2015). A conceptual framework to study the role of communication through social software for coordination in globally-distributed software teams. *Information and Software Technology*, *63*, 11-30. https://doi.org/10.1016/j.infsof.2015.02.013

[37] Akgün, A. E., Keskin, H., Ayar, H., & Okunakol, Z. (2017). Knowledge sharing barriers in software development teams: A multiple case study in turkey. *Kybernetes*, *46*(4), 603-620. https://doi.org/10.1108/K-04-2016-0081

[38] Ammad, G., Iqbal Janjua, U., Madni, T. M., Cheema, M. F., & Shahid, A. R. (2019). An empirical study to investigate the impact of communication issues in GSD in pakistan's it industry. *IEEE Access*, *7*, 171648-171672. https://doi.org/10.1109/ACCESS.2019.2953008

[39] Saaty, T. L. & Vargas, L. G. (2012). *Models, Methods, Concepts & Applications of the Analytic Hierarchy Process* 175. https://doi.org/10.1007/978-1-4614-3597-6

Contact information:

Anita HIDAYATI, PhD candidate
1) Faculty of Computer Science, Universitas Indonesia
Kampus Baru UI Depok, Pondok Cina, Beji, Depok West
Java, Indonesia, 16424
E-mail: anita.hidayati@ui.ac.id
2) Department of Computer and Informatics Engineering,
State Polytechnic of Jakarta
Jl. Prof. DR. G.A. Siwabessy, Kampus Universitas
Indonesia, Depok
West Java, Indonesia, 16425
Email: anita.hidayati@tik.pnj.ac.id

Eko K. BUDIARDJO, Professor
(Corresponding author)
Faculty of Computer Science, Universitas Indonesia
Kampus Baru UI Depok, Pondok Cina, Beji, Depok West
Java, Indonesia, 16424
E-mail: eko@cs.ui.ac.id

Betty PURWANDARI, Doctor of Philosophy
Faculty of Computer Science, Universitas Indonesia
Kampus Baru UI Depok, Pondok Cina, Beji, Depok West
Java, Indonesia, 1642
E-mail: bettyp@cs.ui.ac.id