TOTAL DISRUPTION OF PERFORMANCE FOR SOFTWARE PROGRAMMER TEAM SELECTION

Fitria[1]; Neneng Rachmalia Feta[2]

Sistem dan Teknologi Informasi[1,2]
BRI Institute of Technology and Business
http:/bri-institute.ac.id
Fitria.fitrib@gmail.com[1*)], nenengrachmaliafeta@gmail.com[2]

(*) Corresponding Author

Abstract—In 2020, there were 50% of software development projects failed. Lack of development team skills is one of the failure factors of software development projects. On the other hand, the method of selecting a team based on the interdependence of team members which is considered to increase the chances of team success does not consider the skills in forming a development team. Therefore, in this study, the formula for calculating the total disruption of the performance of the development team was improved on the interdependence-based development team selection method, by adding a skill variable to increase the chances of team success. The results of improvement the calculation formula for the total disruption to the performance of the development team by adding a skill variable, shows that it can reduce the risk of project failure, in terms of the number of late requirements and bugs/errors.

Keywords: Total Disruption Of Performance, Interdependence, Team Selection, Skill

Abstrak—*Pada tahun 2020, terdapat 50% proyek pengembangan perangkat lunak mengalami kegagalan. Kurangnya keterampilan tim pengembang merupakan salah satu faktor kegagalan proyek pengembangan perangkat lunak. Dilain sisi metode pemilihan tim berbasis interdepensi anggota tim yang dianggap dapat meningkatkan peluang keberhasilan tim, tidak mempertimbangkan keterampilan dalam pembentukan tim pengembang. Oleh sebab itu, pada penelitian ini dilakukan peningkatan formula perhitungan total disrupsi kinerja tim pengembang pada metode pemilihan tim pengembang berbasis interdependensi, dengan menambahkan variabel keterampilan untuk meningkatkan peluang keberhasilan tim. Dari hasil perbaikan formula perhitungan total disrupsi kinerja tim pengembang dengan menambahkan variabel keterampilan menunjukkan dapat mengurasi risiko kegagalan proyek, dari sisi jumlah requirement yang terlambat dan bugs/errors.*

Kata Kunci: *Total Disrupsi Kinerja, Interdepensi, Pemilihan Tim, Keterampilan*

INTRODUCTION

In 2020, there were 50% of software development projects failed [1]. One of the reasons was the lack of skills of members of the development team [1]. Skill is one of the factors that affect the success of software development projects [1], [2]. Based on the results of research done by the Standish group, skills are ranked fifth as a factor that affects the success of software development projects [1]. Therefore, the project manager needs to form a team that has skills that are suitable to the project's needs [3], [4]. Informing a team of software developers, there are several team orientations used to form a team, such as homogenous, heterogeneous, and interdependency team member selection [5]. The method that is frequently used and considered to be able to form the best team in developing software is an interdependency-oriented team between team members [5], [6]. This method selects team members based on the social element connection and calculates the total disruption of performance of team members' candidates which formed into several teams [5]. The team with the smallest total disruption of performance value will be selected as a team of software developers. Social elements are assessed based on the relationship between team members in completing work according to the project leader [5], [6], [7].

Calculation of the total value of disruption of performance is calculated based on the sum of disruption values multiplied by the value of the performance of each team member [5], [6]. The value of disruption is obtained from how many dependent partners are gone while the value of performance is assessed from the direct supervisor of the team members from previous software

development projects [5], [6]. The calculation of the total disruption of performance will impact the opportunity for the formation of teams that do not have the skills to meet the needs of software development projects.

In 2018, Baskara conducted the research by determining dependent partners based on skills using the bee colony method [6], [8]. In this research, we only make improvements to the stages of determining the dependent partner, which is by determining the dependent partner candidate team members based on the results of skill measurement [6]. But the formula for calculating the total value of disruption of performance used is still similar, which causes the skills of the formed team to be not optimal and can affect the success of the project [6].

Therefore in this research, we will make improvements to the method of determining the dependent partner of each team member by paying attention to the relationship of candidate team members based on social elements and skills and formulating the formula for calculating the total disruption of performance by adding skill variables as dividers into calculation formulas. Skill distribution in formulas is to prevent the high value of disruption from social elements and to optimize the value of social element disruption to expertise.

The purpose of this study in improving the team selection method is to produce a method of determining the contributing partner of a dependent partner objectively by adding skill variables of team members. So that each dependent partner can depend on a contributing partner who has more skills than him in completing project tasks and can increase the chances of team success

MATERIALS AND METHODS

The research is divided into several activities, starting with conducting preliminary observations, problem identification & formulation, solution analysis, solution design, testing & evaluation of methods, results, and discussion and conclusions. In general, it can be seen in Figure 1.

Preliminary observations. The first step is to conduct preliminary observations related to the project failure risk analysis from resource factors, team-building methods, comparisons of related previous studies, research attributes, research problems based on literature studies, and previous research observations.

Identification and formulation of research problems. At this stage, the variables that will be used in the research are determined and the research problem is formulated into a research problem formulation. Solution analysis. Solution analysis is done by analyzing the appropriate parameters to determine interdependency between

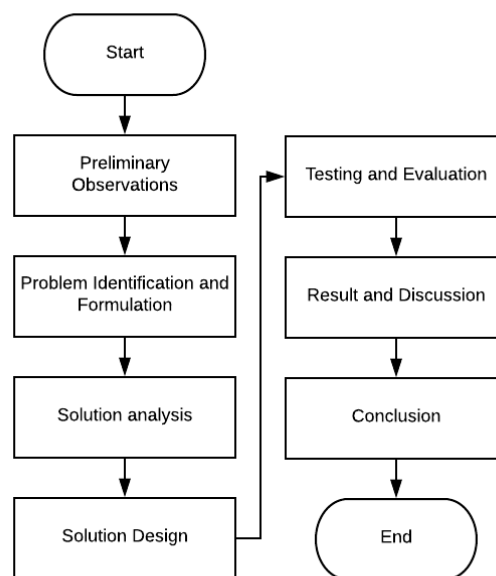candidates and analyzing the method of determining team members in the previous method.

Figure 1. Research Methodology

Solution design. Designing a team formation method that has interdependence in terms of skills and obtains team results that have the minimum total value of disrupted performance. The design carried out includes the design of the method for determining the contributing partner of each candidate based on the skill level and risk analysis of each candidate

Testing and evaluation. Testing the accuracy of the method in determining the team that has the minimum total value of disrupted performance, comparing the teams generated by the proposed method with the previous method.

Results and Discussion. This stage contains the results of testing the developed method and also a discussion of the results of the tests that have been carried out. And Conclusions from research activities.

The formation of interdependency-oriented teams is formed by calculating the total value of the team's disruption of performance [4], [5]. The team with the lowest total disruption of performance value will then be selected as a team of software developers. Calculation of the total value of disruption of performance is calculated by summing the results of the multiplication of performance and disruption of each candidate member of the team [5]. The performance value is assessed by the candidate team members obtained from the candidate's superior team members in the previous project [4]. While the value of disruption is obtained by calculating the number of dependent partner

candidates for the team that is not placed with one team with him [5]. Dependent partners are people who can be relied on by someone to complete their work [5], [6]. The stages of team formation using interdependency-oriented methods are called the determination of interdependency and team formation [5].

Determination of interdependency is a step to determine the dependent partner of each candidate member of the team [5]. Dependent partners are people who can be relied upon to complete work or contribute [5]. The determination of interdependency between candidates for team members with dependent partners is assessed from the proximity or social elements assessed by superiors [5]. Furthermore, after getting the dependent partner of each candidate member of the team, sorting the candidate team data is based on the contribution of team members [5]. Then the data of the team members that have been sorted are divided into two parts or two teams [5]. If the number of team members needed is smaller, then delete the team members who are low contributors. If after eliminating the number of team members formed still exceeds what is needed, then delete the team members who have the least support [5]. Perform calculation of the value of Total Disruption Performance (TDP) of each team using formula 1. The selected team is the team that has the lowest total disruption performance value [5].

$$TDP = d_1p_1 + d_2p_2 + ... + d_np_n \quad .... (1)$$

d = disruption
p = performance
n = quantity of employee.

The development team data was obtained from the developer team data at the BubatDev software house in Bandung and Jakarta. Testing the team formation method is carried out by evaluating the risk analysis of the team formed on the number of late issues in the last project and the number of issues with bug categories in the last project (Table 1).

Table 1. Risk Evaluation Method

Probability (P=T*V)		Impact (I = P*Cost (Assets)
Treat Catalog (T)	Vulnerability (V)	Assets(Cost)
Delay in delivery (PT1) Bug/ Error (PT2)	Skill	Salary

Delay in delivery is the number of late issues in the last project, Bug/error is the number of issues with bug categories in the last project and Issue is the list of project task features, bugs, and project management for bitbucket to be tracked

RESULTS AND DISCUSSION

A. Determination of Interdependency

Determination of interdependency is done by measuring the skills of each candidate member of the team and proximity from the social element side. Candidates for team members will have a dependent partner who has higher skills than him and has a relationship from the social element side. The skills of candidates for team members are assessed in terms of hard skills and soft skills.

Hard skill measurement is done using a programming quiz. The programming quiz is taken from online certification (https://www.brainbench.com/) with the category of java programming language (J2EE). While the soft skill measurement is done using the 360-degree measurement method. The measurement instrument used is the competency dictionary in Spencer's work that has been mapped into the soft skill category that must be owned by the programmer (Table 2) [9]–[11].

Table 2. Softskill Programmer [7] [8]

Softskill Category	Competency	Indicator	Measure
Team Player	Teamwork -kesungguhan-(TW1)	Spencer scale	Competency level
	Teamwork - inisiatif-(TW2)	Spencer scale	Competency level
Group work	Developing others, (DEV)	Spencer scale	Competency level
	Concern for Order (CO)	Spencer scale	Competency level
	Achievement Orientation – Motivated Action (ACH1)	Spencer scale	Competency level
Time Management	Fleksibelitas (FLX2)	Spencer scale	Competency level
Listening Skills	Interpersonal understanding - Kedalaman pemahaman terhadap orang lain - (IU1)	Spencer scale	Competency level
	Interpersonal understanding - Mendengar Dan Merespon Orang Lain - (IU2)	Spencer scale	Competency level
Problem Solving	Analitical Thingking - Ukuran Permasalahan Yang dihadapi - (AT2)	Spencer scale	Competency level

Softskill Category	Competency	Indicator	Measure
Critical thinking	Analitical thinking - Kompleksitas Analisis- (AT1)	Spencer scale	Competency level
	Conceptual thinking, (CT)	Spencer scale	Competency level
Trustworthiness	self-confidence (SCF)	Spencer scale	Competency level
Ability to work under pressure	Self-control, (SCT)	Spencer scale	Competency level
Personal Integrity	Expertise -Penguasaan keilmuan- (EXP3)	Spencer scale	Competency level
	Achievement Orientation – Degree of Innovation (ACH2)	Spencer scale	Competency level
	Inisiatif (INT)	Spencer scale	Competency level
	Berorientasi pada pelanggan (CSO)	Spencer scale	Competency level

Table 3. Dependent partner based on the social element

Person Id	Dependent Partner based on Social Element SE1	SE2	Performance	Vulnerability	Skill
A1	A2	A12	0,8	35	65
A2	A9	A11	0,66	30	70
A3	A12	A4	7	55	45
A4	A12	A1	0,5	45	55
A5	A12	A1	0,7	45	55
A6	A7	A14	0,85	35	65
A7	A16	A2	0,6	20	80
A8	A1	A12	0,78	65	35
A9	A2	A12	0,8	35	65
A10	A5	A12	0,7	25	75
A11	A2	A9	0,45	55	45
A12	A14	A1	0,8	30	70
A13	A12	A2	0,7	45	55
A14	A15	A12	0,67	75	25
A15	A14	A12	0,6	45	55

In this study, the team formation simulation was conducted by forming a team of software development programmers. For comparison to the results of the team formed, a dependent partner was determined based on interdependency social elements based on previous research (Table 2). The assessment of dependent partners based on interdependency social elements is assessed by the project manager who has collaborated with each of the candidates for the project team before.

B. Team Formation

The next step in forming an interdependency-oriented team after determining the dependent partner is to sort the dependent partner data based on their contribution and break the candidate team members into several teams based on the level of contribution.

In Table 4 can be seen the distribution matrix of the initial data of the programmer (Table 4). Table 5 is the result of sequencing the candidate members of the team formed based on contributions in terms of social elements (Table 5), while Table 6 is the result of sequencing the candidate members of the team formed in terms of social elements and skills (Table 6).

Table 4. Dependent partner based on a social element by the project manager

	A1	A2	A3	A4	A5	A6	A7	A8	A9	A10	A11	A12	A13	A14	A15
A1	*	*										*			
A2		*							*		*				
A3			*	*								*			
A4	*			*								*			
A5	*				*							*			
A6						*	*							*	
A7		*				*	*								
A8	*							*				*			
A9		*							*			*			
A10					*				*	*		*			
A11		*							*		*			*	
A12	*											*			
A13		*										*	*		
A14												*		*	*
A15												*		*	*

Table 5. Shorted matrix dependent partner based on a social element by the project manager

	A10	A5	A8	A1	A3	A4	A13	A12	A6	A14	A15	A7	A2	A9	A11
A10	*	*						*							
A5		*		*				*							
A8			*	*				*							
A1				*				*					*		
A3					*	*		*							
A4				*		*		*							
A13							*	*					*		
A12				*				*		*					
A6									*	*		*			
A14								*		*	*				
A15								*		*	*				
A7									*			*	*		
A2													*	*	*
A9								*					*	*	
A11													*	*	*

Table 6. Shorted matrix dependent partner based on social element and skill

	A5	A1	A3	A4	A8	A10	A13	A12	A6	A14	A15	A7	A2	A11	A9
A5	*	*						*							
A1		*						*					*		
A3			*	*				*							
A4		*		*				*							
A8		*			*			*							
A10	*					*		*							
A13							*	*					*		
A12		*						*		*					
A6									*	*		*			
A14								*		*	*				
A15								*		*	*				
A7									*			*	*		
A2													*	*	*
A11													*	*	*
A9								*					*		*

Furthermore, from the data that has been sorted, TDP is calculated. The formula for calculating TDP in the previous study (formula 1) was improved by adding skill variables (formula 2) to prevent the high value of disruption from social elements and optimize the value of social element disruption to expertise.

$$\text{TDP} = \frac{d_1 p_1}{s_1} + \frac{d_2 p_2}{s_2} + \ldots + \frac{d_n p_n}{s_n} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots (2)$$

d = disruption
p = performance
n = total number of employees
s = skill

Based on the results of the calculation of the total disruption of performance, teams formed based on interdependency social elements are teams consisting of A10, A5, and A8 (Table 7). While the team formed using the TDP improvement calculation method produced a team consisting of A2, A11, and A9 as the development team (Table 8).

Table 7. Total disruption of performance team based on the social element

Team			TDP
A10	A5	A8	2.1
A1	A3	A4	4.28
A13	A12	A6	9.15
A14	A15	A7	4.48
A2	A9	A11	3

Table 8. Total disruption of performance team based on social element and skill

Team			TDP
A5	A1	A3	2.58
A4	A8	A10	1.55
A13	A12	A6	4.59
A14	A15	A7	1.92
A2	A11	A9	1.54

The results of the team formed to develop software development projects using the total disruption of performance calculation formula from

JITK (JURNAL ILMU PENGETAHUAN DAN TEKNOLOGI KOMPUTER)

previous research and improvement have different teams produced. The method of forming teams that only use social elements in their TDP calculations produces a team consisting of A10, A5, and A8. While the formation of a team that calculates TDP uses social elements and skills produce team consists of A2, A11, and A9.

The two teams selected with different methods tested with a risk assessment to see a comparison of the chances of possible team failures in developing software [12]. Risk assessment is carried out by considering two types of threats, the number of late requirements (T1) and the number of requirements that are bug/error (T2). Risk assessment is assessed based on possible threats from skills possessed by each team member [12], [13], [14], [15]. The number of late requirements seen from the number of unfinished issues does not match the schedule of the previous project, while the number of requirements that are bugs/errors is seen from the number of issues that have a category of bugs/errors in the previous project of each candidate team. In Table 9, it can be seen the results of the risk assessment carried out on the team formed using the social element calculation method only, the chance of late requirement risk is 18.19 and the requirement risk is a bug / error of 19.66. Whereas the team formed using the calculation of TDP social elements and skills has a risk of late requirements of 13.06 and the requirement risk that is a bug/error of 13.97 (Table 10). This shows that the team formed by using TDP and social element calculations has a lower chance of software development project failure in terms of delays and bug / error requirements.

Table 9. Opportunities for team threats to be formed based on the social element

	A10	A5	A8	Risk
T1	4.5	10.73	39.34	18.19
T2	8.15	7.84	42.98	19.66

Table 10. Opportunities for team threats to be formed based on social element and skill

	A2	A11	A9	Risk
T1	16.43	11.99	10.77	13.06
T2	14.46	23.31	4.15	13.97

CONCLUSION

The calculation of total disruption of performance using social element and skill variables can reduce the chance of the risk of the threat of late and bugs/errors. This is caused by the team members depending on the dependent partner who has proximity in terms of social elements with it and has more skills than him. In addition, the improvement of formula total disruption of performance by adding with skill variables can prevent the high value of disruption from social elements and optimize the value of disruption of social elements on the expertise of each member of the development team.

REFERENCE

[1] M. Woźniak, "Sustainable approach in it project management—methodology choice vs. Client satisfaction," *Sustain.*, vol. 13, no. 3, pp. 1–21, 2021, DOI: 10.3390/su13031466.

[2] T. F. Kusumasari, "Instruments Measurement Design of Human Behavior in Collaborative Software Construction."

[3] P. Holtkamp, *Competency Requirements of Global Software Development Philipp Holtkamp Competency Requirements of Global Software Development*. 2015.

[4] S.-C. Hsu, K.-W. Weng, Q. Cui, and W. Rand, "Understanding the complexity of project team member selection through agent-based modeling," *Int. J. Proj. Manag.*, vol. 34, pp. 82–93, Jan. 2016, doi: 10.1016/j.ijproman.2015.10.001.

[5] W. P. Millhiser, C. A. Coen, and D. Solow, "Understanding the role of worker interdependence in team selection," *Organ. Sci.*, vol. 22, no. 3, pp. 772–787, 2011, DOI: 10.1287/orsc.1100.0549.

[6] Fitria and I. G. B. B. Nugraha, *Formation of Software Programmer Team Based on Skill Interdependency*. 2018.

[7] A. R. Gilal, M. Omar, J. Jaafar, K. I. M. Sharif, A. W. Mahessar, and S. Bin Basri, "Software development team composition: Personality types of programmer and complex network," 2017.

[8] D. Karaboga and B. Akay, "A comparative study of Artificial Bee Colony algorithm," *Appl. Math. Comput.*, vol. 214, no. 1, pp. 108–132, 2009, DOI: https://doi.org/10.1016/j.amc.2009.03.090.

[9] J. Lyle M. Spencer and S. M. Spencer, *Competence at Work: Models for Superior Performance. Japan Productivity Center.* Wiley, 1993.

[10] E. S. Mtsweni, T. Hörne, and J. A. van der Poll, "Soft Skills for Software Project Team Members," *Int. J. Comput. Theory Eng.*, vol. 8, no. 2, pp. 150–155, 2016, DOI: 10.7763/ijcte.2016.v8.1035.

[11] A. R. Gilal, J. Jaafar, S. Basri, M. Omar, and A.

Abro, "Impact of software team composition methodology on the personality preferences of Malaysian students," in *2016 3rd International Conference on Computer and Information Sciences (ICCOINS)*, 2016, pp. 454–458, doi: 10.1109/ICCOINS.2016.7783258.

[12] ISACA, *The Risk IT Framework*. ISACA, 2009.

[13] B. C. D. Anda, D. I. K. Sjøberg, and A. Mockus, "Variability and reproducibility in software engineering: A study of four companies that developed the same system," *IEEE Trans. Softw. Eng.*, vol. 35, no. 3, pp. 407–429, 2009, DOI: 10.1109/TSE.2008.89.

[14] D. T. Roberts, "Applying risk assessment at the worker level," in *2017 Petroleum and Chemical Industry Technical Conference (PCIC)*, 2017, pp. 381–386, DOI: 10.1109/PCICON.2017.8188758.

[15] M. T. Riaz, M. S. Jahan, K. S. Arif, and W. H. Butt, "Risk Assessment on Software Development using Fishbone Analysis," in *2019 International Conference on Data and Software Engineering (ICoDSE)*, 2019, pp. 1–6, DOI: 10.1109/ICoDSE48700.2019.9092727.