

INVESTIGATING THE RELATIONSHIP OF PERSONALITY TYPES AND TIME PRESSURE AMONG SOFTWARE DEVELOPERS BASED ON A RULE-BASED APPROACH

Ruqaya Gilal, Mazni Omar and Mawarny Md Rejab School of Computing, Universiti Utara Malaysia, Malaysia E-Mail: ruqayagilal@hotmail.com

ABSTRACT

Several developers exhibit dysfunctional behaviour under time pressure (TP). It is believed that "situations and personality are two sides of a coin that cannot be separated". Different personality traits are believed to produce different results under TP. There are numerous studies conducted in the past finding relations between different job roles &TP and personality. However, a lack of studies has been found in software development to measure the relationship between TP and the developer's personality. Therefore, this study has set an objective to find out the relationship between software developer's personality types (based on Myer-Briggs Type Indicator) and TP and their impact on software quality. The "completely randomized design" was used over 30 senior software engineering (SE) students for data collection. The experiment design contained six experimental rounds with different task complexity levels (i.e., easy, medium and hard) and TPs. Each even-numbered round (i.e., 2, 4, 6) was induced with TP, whereas odd rounds were without TP (NTP). The McNemar test was used to find personality differences between groups. Moreover, a Genetic Algorithm (GA) was used under Rough Set Theory (RST) to develop the model. Model validation was performed by applying Naïve Bayes classifier and considering prediction accuracy, F1-score and Area under Curve (AUC)-Receiver Operation Characteristics (ROC) values. The McNemar test results showed that except round 1 and round 2, all rounds had a significant number of personality differences. Later, GA techniques returned 43 decision rules with 72.20% prediction accuracy. Based on the results, it can be concluded that certain personality types can deal with TP effectively. For instance, intuiting, thinking and judging personality-based combinations can handle TP effectively.

Keywords: software development, time pressure, personality, MBTI, genetic algorithm, model development.

INTRODUCTION

Time pressure (TP) has been considered an important factor for software development projects. TP is described by Cooper *et al.* [1]as a lack of time to accomplish a task.TP is a psychological state that occurs when there is insufficient time to complete a task [2]. Basten [3] found that TP is an individual's perception. It is all about how people perceive TP. There are always several developers who produce dysfunctional behaviour under TP. TP is defined as the perception or feeling that there is inadequate time to finish a task or series of tasks to meet job obligations or that workers must work quicker than usual [4], [5], [6]. Overall, TP is a factor or situation interpreted differently by different people because it is based on the individual's perception of how they can handle it. It was discovered that various people had varying reactions to TP [7], [8].

TP has been considered both positive and negative reasons for software developer performance. For instance, it is a positive stressor when managers want to engage developers in activities to extract the best performance[9]. Some individuals claim that TP increases software developers' burdens, eventually resulting in burnout [7]. Due to TP, several developers quit jobs and, more seriously, several face physiological and psychological problems. Team performance, project quality and low productivity are frequently related to TP. TP is mostly associated with negative outcomes for software projects [10].

Software engineering (SE) is overall centred on human factors [11], [12]. Because software is generated and utilized by a wide range of people, it is vital to comprehend individual behaviour when developing software [13], [14]. Humans are the major source of software development and are responsible for ensuring that operations are reliable, effective, and of high quality [15]. The human aspects are difficult to comprehend because everyone approaches challenges in their way. As a result, managing people in the SE is increasingly important [16].Software failure is caused by a lack of knowledge of human factors. As a result, the issue occurs when the actual cause of software failure is overlooked or misconstrued. Blaming technology alone would provide no relief. As a result, the exact cause of the software failure is due to pay less attention to human issues [17], [18], [19], [20].

No human activates himself in the same way in a different scenario. Tett et al. [21] mentioned that "trait and situation form two sides of a coin that cannot be separated from each other." The initial motivation of this study was taken from the trait activation theory (TAT). The principle of TAT [21] has a basic philosophy that personality traits are hidden tendencies to behave in certain ways in a different scenario. TAT development also focuses on the individual's reactions based on situations and performance outcomes depending on personality traits. TP situations are never easily manageable by every individual in software development workplaces. The assignment of work to employees suited for their personalities is the most important part of project success [22]. Heaton and Kruglanski [23] observed it more closely by segregating and claiming that introvert and extrovert personality traits would produce different perceptions under TP. Because

ARPN Journal of Engineering and Applied Sciences ©2006-2022 Asian Research Publishing Network (ARPN). All rights reserved.



www.arpnjournals.com

every human is different in dealing with TP, personality can help us recognize the way developers are dealing the TP. Several studies have been conducted in social sciences to find the relationship between TP and personality and their impact on project outcome. For example, a person who is high in neuroticism may act impulsively in such circumstances and make a wrong decision under TP[24]. Fisher [25] studied the relationships between personality with job stress, satisfaction and performance while performing the audit. He found a significant association between personality and job satisfaction and performance. Similarly, In the study of Kocher et al.[26], they worked to check who could cope with the TP better than others; the result showed that individuals with high cognitive ability and intellectual efficiency can perform better under TP.

VOL. 17, NO. 19, OCTOBER 2022

Unfortunately, a scarcity of work in SE highlights using such a strategy to lessen or mitigate TP and personality type difficulties. However, empirical SE literature contains few studies which have worked on TP. MiikkaKuutila and Mika Mantylaare among those few authors highlighting TP's importance in SE. They have conducted a literature study [10], concluding that several psychological problems occur due to TP. Furthermore, in their study, TP has been found as a major factor for more errors and low quality. Authors also claimed that software development and auditing are cognitive jobs requiring serious attention to detail and understanding of rules. However, several studies conducted in the past find relations between auditor personality types and TP[3], [19]. There is still a lack of research in SE that reveals which personality types are better at dealing with TP. Capretz et al. [28]; and Smith et al. [29] discovered that evaluating employee's personalities before assigning work is vital since the success or failure of software projects depends on personality types. As a result, it is vital to apply TP in SE to study the personality of employees because human personality influences employee performance and affects software quality [15], [22], [23]. Therefore, this study conducts an experiment to identify the personality types suitable to work efficiently under TP.

METHODOLOGY

The empirical data was collected by arranging the controlled experimental setup on the students of the Sukkur IBA University (SIBAU). A total of 30 senior students voluntarily participated in the study. Those students were selected based on 1) Age of 20 years or 20+ and2) Academic grades. The minimum age of 20 years was set up to avoid internal threats to validity. Because researchers from the personality domain believed that personality under changes[32]. Similarly, 20 academic grades of programming subjects were considered to ensure that technical weaknesses did not influence the results of the experiments. Moreover, Myers-Briggs Type Indicator (MBTI) [33] was used to obtain the participant's personality types. It is widely used instrument and recommended by SE researchers [26], [27], [28], [29].

The "completely randomized design" was used for the experiments. A completely randomized design requires all the participants to obtain all the experiment's treatments. It is the simplest of all the randomization and replicationbased designs (Xian & Liu, 2019). This ensures that all experimental unit has an equal chance of receiving treatment. It is simple to figure out the design. It is adaptable in terms of the number of treatments and replication, which can vary greatly from one treatment to the next.

This study's experiment had six rounds in which each participant was required to undergo the task in each round. Six rounds were setup to minimize the internal threats to validity and increase the model acceptance. The experiment's goal was to create a software coding program based on a suggestion from a subject expert or lecturer. The tasks were assigned time-based on the pre-test approach. This method involves measuring the amount of time required to accomplish a task, giving the TP group limited time and the NTP group ample time to do the task. Topi, Valacich and Hoffer [38] employed the pre-test strategy in their controlled experiment. Setting time restrictions for assessing TP in experiments is useful [39]. The tasks in the odd rounds were with no time pressure (NTP). This way, we set up the benchmark to compare with TP results. On the other hand, tasks in the even rounds were induced with time pressure (TP) by reducing the actual time limits.

Once the tasks were submitted to the teacher, the subject teacher evaluated the submitted tasks based on the requirements. The tasks were given a score ranging from 1 to 100. Based on the marks, participants were classified into two performance classes: effective or ineffective. Table-1 denotes the overall study variables and their values.



Table-1. Study variables and their values in the study.

Input
1. Easy with no TP (NTP)
2. Easy with TP
3. Medium NTP
4. Medium TP
5. Hard NTP
6. Hard TP
1. Introvert
2. Extrovert
1. Sensing
2. Intuiting
1. Thinking
2. Feeling
1. Judging
2. Perceiving
1. (Marks <77) Ineffective
2. (Marks>= 77) Effective

Before applying further modeling techniques, it was important to evaluate whether the difference between the rounds (i.e., NTP and TP) exists or not. The McNemar test was applied over the rounds results by considering a minimum p1-value of 0.05 and a p2-value of 0.10. We found except for round 1 and round 2, the other rounds had a considerable difference between the personality types of NTP and TP. It concluded that there was no difference in the performances when tasks were simple. The level of tasks impacts the performance. Following Table-2 summarizes the McNemar test results between each comparable round.

Table-2. McNemar test results.

Trail	Between		p1-value	p2-value	Difference
1	Round 1	Round 2	0.003	0.007	No
2	Round 3	Round 4	0.091	0.182	Yes
3	Round 5	Round 6	0.342	0.684	Yes

Moreover, a Genetic algorithm (GA) was used under rough set theory (RST) to develop the model. GA is an artificial intelligence algorithm with a heuristic search process that copies natural evolution. GA rely on biologically inspired operators such as mutations, crossover, and selection to develop high-quality solutions and optimise and search for issues [40], [41]. The ability to implement GAs as a "universal optimiser" that can be used to optimize any sort of issue from various fields [42], [43] is one of the advantages of such algorithms. A GA can reveal elegant, multifaceted answers to problems that a researcher may be oblivious of, and those appropriate engineering methodologies are unlikely to find [44], [45]. Furthermore, the dataset was divided into subsets for model training and testing. The Naïve Bayes classifier was applied for the testing, and prediction accuracy, F1-score and Area under Curve (AUC)-Receiver Operation Characteristics (ROC) measures were considered over the developed model.

RESULTS AND DISCUSSIONS

We have sorted results for discussion based on the complexity level of the tasks (i.e., easy, medium and hard). As mentioned earlier, the subject expert suggested the task, and the task complexity was induced in the tasks by the subject expert. The GA algorithm in RST generated some rules. The "rules with TP and NTP" can also be utilized as a decision statement based on a dataset-driven "IF-THEN" condition. Each rule has two interpretations: 1) Situation; 2) Decision The statement before ("=>") is termed the IF-PART in the table above, and the statement following after (i.e., "=>") is called the THEN-PART. The left-PART is the number of objects in the dataset that fit the if-statement. On the other hand, the right part shows how many objects from

ISSN 1819-6608



www.arpnjournals.com

the dataset match the "then-statement/then-part" depending on the if-statement. In the easy level, six decision rules were formed from round 1. In other words, six personality combinations were formed to denote effective and ineffective classes. Based on the rules, only one personality combination (Introvert, Intuiting and Feeling) is appeared to be ineffective for yielding positive results on the given tasks. Whereas remaining five rules were supporting the effective class. These rules were composed of introvert(1), extrovert(3), sensing (2), intuiting (1), thinking (1), feeling(1), judging (1) and perceiving (2) times. It shows that extrovert personality combinations mainly dominated the effective results if the task is NTP. On the other hand, seven rules were formed in round 2. Only three of them were composed for effective class and the remaining four for ineffective class. Based on the rules, it was observed that intuiting and judging traits are composed of the effective class rules only. Moreover, the ineffective class mainly dominated introvert and perceiving-based personality compositions. It explains that one should have intuiting and judging-based personality formations that can tackle the TP. Similarly, introvert and perceiving personality combinations can take TP as these traits were observed effective in the round 1. Table-2 summarizes the outcomes of each rule in their respective rounds.

Table-3. Presents the overall rules which were formed during experiments.

S. No	Rules without TP	Rules with TP				
	Round 1 vs Round 2					
1	Round(1) AND Introvert AND Sensing =>	Round(2) AND Introvert AND Intuiting AND Judging				
1	Outcome(1)	=> Outcome(1)				
2	Round(1) AND Introvert AND Intuiting AND $Facting \rightarrow Outcome(0)$	Round(2) AND Introvert AND Thinking =>				
	Round(1) AND Extrovert AND Intuiting =>	Round(2) AND Introvert AND Sensing AND				
3	Outcome(1)	Perceiving => Outcome(0)				
4	Round(1) AND Extrovert AND Perceiving => Outcome(1)	Round(2) AND Introvert AND Intuiting AND Feeling AND Perceiving => Outcome(0)				
5	Round(1) AND Extrovert AND Sensing AND Feeling AND Judging => Outcome(1)	Round(2) AND Extrovert AND Intuiting AND Thinking => Outcome(1)				
6	Round(1) AND Thinking AND Perceiving => Outcome(1)	Round(2) AND Extrovert AND Feeling => Outcome(0)				
7		Round(2) AND Sensing AND Judging => Outcome(1)				
	Round 3	3 vs Round 4				
1	Round(3) AND Introvert AND Intuiting AND Feeling AND Judging => Outcome(1)	Round(4) AND Introvert AND Sensing => Outcome(0)				
2	Round(3) AND Introvert AND Feeling => Outcome(0)	Round(4) AND Introvert AND Thinking => Outcome(1)				
3	Round(3) AND Introvert AND Sensing AND Judging => Outcome(0)	Round(4) AND Introvert AND Judging => Outcome(1)				
4	Round(3) AND Introvert AND Perceiving => Outcome(0)	Round(4) AND Extrovert AND Sensing => Outcome(0)				
5	Round(3) AND Extrovert AND Intuiting AND Feeling AND Perceiving => Outcome(1)	Round(4) AND Extrovert AND Thinking => Outcome(1)				
6	Round(3) AND Extrovert AND Feeling => Outcome(1)	Round(4) AND Extrovert AND judging => Outcome(1)				
7	Round(3) AND Extrovert AND Sensing => Outcome(1)	Round(4) AND Intuiting AND Perceiving => Outcome(0)				
8	Round(3) AND Sensing AND Perceiving => Outcome(1)					
	Round	5 vs Round 6				
1	Round(5) AND Introvert AND Intuiting AND	Round(6) AND Introvert AND Intuiting AND Feeling				
1	Feeling AND Perceiving => Outcome(1)	AND Judging => Outcome(0)				
2	Round(5) AND Introvert AND Intuiting =>	Round(6) AND Introvert AND Intuiting AND Judging				
	Round(3) AND Extrovert AND Intuiting AND	Round(6) AND Extrovert AND Intuiting AND				
3	Feeling AND Judging => Outcome(1)	Thinking AND Judging => Outcome(1)				
4	Round(5) AND Extrovert AND Perceiving => Outcome(0)	Round(6) AND Extrovert AND Perceiving => Outcome(0)				
5	Round(5) AND Extrovert AND Sensing AND Feeling => Outcome(1)	Round(6) AND Sensing => Outcome(0)				
6	Round(5) AND Extrovert AND Intuiting AND Judging => Outcome(0)	Round(6) AND Feeling => Outcome(0)				
7	Round(5) AND Extrovert AND sensing AND Thinking => Outcome(1)					
8	Round(5) AND Intuiting AND Feeling AND Indiang -> Outcome(0)					
9	Round(5) AND Thinking AND Perceiving => Outcome(1)					
	~ /					

In the round 3 and 4, complexity was slightly greater than round 1 & 2 as mentioned above. Therefore, we expected to extract new personality combinations that

will behave well in both TP and complexity. Keeping it in view, we got a total of eight rules composed for round 3. In which four rules were classified as effective and four as



ineffective. The results quite openly represent that introvert trait-based combinations in round 3 were mostly ineffective.

Similarly, extrovert, feeling and perceiving traits appeared to be effective traits in round 3. On the other hand, in round 4, the introvert trait appeared to be effective in maintaining quality results with TP provided composed of thinking and judging traits. In the same way, extrovert traitbased personality was effective in round 4 if it was composed of thinking and judging traits. In the psychology language, one can say that thinking trait-based personality people are less emotional and directed based on the organizational or project goals. Similarly, judging people are called well organized. Therefore, based on the limited results, we can say that developers with thinking and judging personality traits could manage quality along with TP.

Nine rules were formed in round 5, the highest number of rules in any round. Four rules were formed for ineffective class decisions. Extrovert, intuiting and judging were found frequently in the ineffective class rules in round 5. On the other hand, in the rules composed of effective, extrovert, sensing, feeling, and perceiving appeared to be dominated in the rules. It was observed that the extrovert trait appeared in both effective and ineffective classes, but the other trait's combination matters. For example, extroverts appeared in the ineffective class if it is composed of perceiving and intuiting and judging (see round 5 rule no 4 and 6 in Table-2).

Moreover, in round 6, six rules were composed; four were classified as ineffective class, and the remaining two were classified as effective. During observing the rules in the round 6, we realised that personality type ENTJ appeared effective straight away. Also, introvert, intuiting and judging personality combination appeared to be effective. Hence, we can claim that intuiting and judgingbased personality combinations are suitable for working under TP. On the other hand, sensing and feeling personality traits appeared to be ineffective (see round 6 rules no 5 and 6). Extrovert with perceiving appeared ineffective in both rounds 5 and 6. We also observed that a combination of an introvert, intuiting and feeling do not handle TP effectively to manage the quality outcome.

As mentioned in the methodology section, prediction accuracy, F1-score and AUC-ROC measures were applied to test the results. Using Naïve Bayes classifier in the testing dataset, this study obtained 72.22%, 76.20%, and 73.20% values on prediction accuracy, F1score and AUC-ROC, respectively. According to Pearson [46], prediction accuracy is one of the essential elements to consider when evaluating a model's performance. This study was chosen based on a prediction accuracy criterion of 70%. The testing results of this study are satisfactory as they achieved the minimum validation benchmark in human studies. In the light of literature, it's worth noting that 70% prediction accuracy is considered adequate for model development[47],[48],[32]. As a result, 70 % was established as the effective accuracy benchmark in this study. Finally, the model was validated using a benchmark of 50% or higher for AUC-ROC [49]and for F1-score measure [50], [51] using the following formula:

$$f1_score = 2 * \frac{precision*recall}{precision+recall}$$
(1)

Table-3 summarizes the results of the measures, respectively.

Table-4. Testing results.

Measure	Obtain value	Benchmark			
Prediction Accuracy	72.22%	70%			
F1-Score	76.20%	50%			
(Note: precision=70.5882 & recall=82.7586)					
ROC-AUC	73.20%	50%			

THREATS TO VALIDITY

First and foremost, the findings of this study can be utilized to identify successful personality types for dealing with TP. Personality is a broad term that can be influenced by a variety of internal and external factors. As a result, generalizing these findings remains a major worry for the study's validity. Internal validity refers to how well a test establishes a reliable cause-and-effect relationship between a treatment and a result. Therefore, the same students conducted all rounds to appropriately measured NTP and TP's influence. The tasks were different in each round, but the level of complexity was the same to avoid threats of task familiarity. The students' age may also influence validity; for example, older students may be able to manage the TP better, whereas younger students may not. As a result, the student's age limit was set at 20 or higher to avoid this hazard and assure personality stability. It may lessen the model's validity threats while also increasing its acceptance. External validity relates to how well a study's findings may be applied to other situations. For external validity issues, third-year students were selected because they did better since they were more familiar with the courses and programming assignments; therefore, they were able to complete the tasks more quickly. Furthermore, students are appropriate SE participants because there are no significant performance differences between students and experienced developers [52][53][54].

CONCLUSIONS

This study was started with the objective of finding out the relationship between human personality types and TP. We can conclude that different personality types behave differently to dealing with TP. Based on the results; it concluded that certain personality types could deal with TP effectively. For instance, intuiting, thinking and judging personality-based combinations can handle TP very well. However, sensing and perceiving personality traits can produce satisfactory results but not during TP. As previously said, the real cause of software failure in SE is neglecting the human factors. Personality variables may impact overall software development outcomes if they are



ignored throughout specific jobs. Workers do their best when they are in their comfort zone, but it is harmful to the organization and the individual who works there if they are not. As a result, the findings revealed that matching employees to their personality types can significantly boost the chances of software success.

LIMITATIONS AND FUTURE WORK

This study has some limitations that could lead to intriguing future research. The data for this study was gathered from software engineering students in an educational setting that limited the scope of the research. As a result, a larger sample size is required to perform the study at a higher level and with professionals from the software development industry. There is also a recommendation for future research to consider gender issues while studying personality types. As a result, more attention must be paid to employees' personality types in combination with their gender to deal with TP because personality types significantly impact gender.

REFERENCES

- C. L. Cooper, C. P. Cooper, P. J. Dewe, M. P. O'Driscoll and M. P. O'Driscoll. 2001. Organizational stress: A review and critique of theory, research, and applications. Sage.
- [2] C. Speier-Pero. 2019. Using aggregated data under time pressure: a mechanism for coping with information overload. J. Decis. Syst. pp. 1-19.
- [3] D. Basten. 2017. The Role of Time Pressure in Software Projects: A Literature Review and Research Agenda. Int. Res. Work. IT Proj. Manag. pp. 1-15.
- [4] A. Baethge, T. Vahle-Hinz, J. Schulte-Braucks, and R. van Dick. 2018. A matter of time? Challenging and hindering effects of time pressure on work engagement. Work Stress. 32(3): 228-247.
- [5] S. Kleiner. 2014. Subjective time pressure: General or domain specific? Soc. Sci. Res. 47: 108-120.
- [6] S. Ohly and C. Fritz. 2010. Work characteristics, challenge appraisal, creativity, and proactive behavior: A multi-level study. J. Organ. Behav. 31(4): 543-565.
- [7] L. M. Maruping, V. Venkatesh, S. M. B. Thatcher, and P. C. Patel. 2015. Folding under pressure or rising to the occasion? Perceived time pressure and the moderating role of team temporal leadersip. Acad. Manag. J. 58(5): 1313-1333.
- [8] B. J. C. Claessens, W. Van Eerde, C. G. Rutte and R. A. Roe. 2007. A review of the time management literature. Pers. Rev.

- [9] J. A. LePine, N. P. Podsakoff, and M. A. LePine. 2005. A meta-analytic test of the challenge stressorhindrance stressor framework: An explanation for inconsistent relationships among stressors and performance. Acad. Manag. J. 48(5): 764-775.
- [10] M. Kuutila, M. Mäntylä, U. Farooq and M. Claes. 2019. Time Pressure in Software Engineering: A Systematic Literature Review. [Online]. Available: http://arxiv.org/abs/1901.05771.
- [11] A. Fuggetta and E. Di Nitto. 2014. Software process. Futur. Softw. Eng. FOSE 2014 - Proc., pp. 1-12, doi: 10.1145/2593882.2593883.
- [12] R. E. D. S. Santos, C. V. C. De Magalhaes, J. D. S. Correia-Neto, F. Q. B. Da Silva, L. F. Capretz and R. Souza. 2017. Would You Like to Motivate Software Testers? Ask Them How. Int. Symp. Empir. Softw. Eng. Meas., 2017-Novem: 95-104, doi: 10.1109/ESEM.2017.16.
- [13] R. S. Pressman. 2014. Software Engineering-A Practitioner's Approach (8th edition). McGraw-Hill.
- [14] A. S. Barroso, J. S. Madureira, M. S. Soares, and R. P. C. do Nascimento. 2017. Influence of human personality in software engineering-a systematic literature review. In International Conference on Enterprise Information Systems. 2: 53-62.
- [15] M. Korsaa et al. 2013. The people aspects in modern process improvement management approaches. J. Softw. Evol. Process. 25(4): 381-391.
- [16] P. Liu and Z. Li. 2011. Toward understanding the relationship between task complexity and task performance. In International Conference on Internationalization, Design and Global Development. pp. 192-200.
- [17] A. R. Gilal, M. Z. Tunio, A. Waqas, M. A. Almomani, S. Khan and R. Gilal. 2019. Task Assignment and Personality: Crowdsourcing Software Development. In Human Factors in Global Software Engineering, IGI Global. pp. 1-19.
- [18] P. Holtkamp, J. P. P. Jokinen and J. M. Pawlowski. 2015. Soft competency requirements in requirements engineering, software design, implementation, and testing. J. Syst. Softw. 101: 136-146.
- [19] L. F. Capretz and F. Ahmed. 2010. Why Do We Need Personality Diversity in Software Engineering ? ACM

SIGSOFT Software Engineering Notes. 35(2), doi: 10.1145/1734103.1734111.

- [20] O. Hazzan and I. Hadar. 2008. Why and how can human-related measures support software development processes? J. Syst. Softw. 81(7): 1248-1252.
- [21] R. P. Tett, D. V Simonet, B. Walser and C. Brown. 2003. Trait activation theory. Handb. Personal. Work. pp. 71-100.
- [22] L. F. Capretz and F. Ahmed. 2018. A Call to Promote Soft Skills in Software Engineering. arXiv Prepr. arXiv1901.01819.
- [23] A. W. Heaton and A. W. Kruglanski. 1991. Person perception by introverts and extraverts under time pressure: Effects of need for closure. Personal. Soc. Psychol. Bull. 17(2): 161-165.
- [24] K. A. Byrne, C. D. Silasi-Mansat and D. A. Worthy. 2015. Who chokes under pressure? The Big Five personality traits and decision-making under pressure. Pers. Individ. Dif., 74: 22-28, doi: 10.1016/j.paid.2014.10.009.
- [25] R. T. Fisher. 2001. Role stress, the type a behavior pattern, and external auditor job satisfaction and performance. Behav. Res. Account. 13(1): 143-170.
- [26] E. Karagözouglu and M. G. Kocher. 2019. Bargaining under time pressure from deadlines. Exp. Econ. 22(2): 419-440.
- [27] L. C. Gundry and G. A. Liyanarachchi. 2007. Time budget pressure, auditors' personality type, and the incidence of reduced audit quality practices. Pacific Account. Rev., 19(2): 125-152, doi: 10.1108/01140580710819898.
- [28] L. F. Capretz, D. Varona and A. Raza. 2015. Influence of personality types in software tasks choices. Comput. Human Behav.52: 373-378.
- [29] E. K. Smith, C. Bird and T. Zimmermann. 2016. Beliefs, practices, and personalities of software engineers: a survey in a large software company. In Proceedings of the 9th International Workshop on Cooperative and Human Aspects of Software Engineering. pp. 15-18.
- [30] S. S. J. O. Cruz, F. Q. B. da Silva, C. V. F. Monteiro, P. Santos and I. Rossilei. 2011. Personality in software engineering: Preliminary findings from a systematic

literature review. In 15th annual conference on Evaluation & assessment in software engineering (EASE 2011). pp. 1-10.

- [31] A. Hasan, S. Moin and M. Pasha. 2019. Prediction of Personality Profiles in the Pakistan Software Industry--A Study. Psych. 1(1): 320-330.
- [32] [32] A. R. Gilal, J. Jaafar, L. F. Capretz, M. Omar, S. Basri and I. A. Aziz. 2018. Finding an effective classification technique to develop a software team composition model. J. Softw. Evol. Process, 30(1), doi: 10.1002/smr.1920.
- [33] I. B. Myers, M. H. McCaulley, N. L. Quenk and A. L. Hammer. 1998. MBTI manual: A guide to the development and use of the Myers-Briggs Type Indicator, vol. 3. Consulting Psychologists Press Palo Alto, CA.
- [34] A. Amin, M. Rehman, R. Akbar, S. Basri and M. F. Hassan. 2018. Trait-Based Personality Profile of Software Programmers: A Study on Pakistan's Software Industry. In 2018 8th International Conference on Intelligent Systems, Modelling and Simulation (ISMS). pp. 90-94.
- [35] M. Z. Tunio *et al.* 2017. Impact of Personality on Task Selection in Crowdsourcing Software Development : A Sorting Approach. pp. 18287-18294.
- [36] A. R. Gilal, J. Jaafar, M. Omar, S. Basri, and I. A. Aziz. 2016. Balancing the personality of programmer: Software development team composition. Malaysian J. Comput. Sci. 29(2).
- [37] A. R. Gilal, M. Omar, and K. I. Sharif. 2014. A rulebased approach for discovering effectivesoftware team composition. J. Inf. Commun. Technol. 13: 1-20.
- [38] H. Topi, J. S. Valacich and J. A. Hoffer. 2005. The effects of task complexity and time availability limitations on human performance in database query tasks. Int. J. Hum. Comput. Stud. 62(3): 349-379.
- [39] M. V Mäntylä and J. Itkonen. 2013. More testers-The effect of crowd size and time restriction in software testing. Inf. Softw. Technol. 55(6): 986-1003.
- [40] M. Buckova, R. Skokan, M. Fusko and R. Hodo\vn. 2019. Designing of logistics systems with using of computer simulation and emulation. Transp. Res. Procedia. 40: 978-985.

- [41] M. Pekarcikova, P. Trebuna, M. Kliment and L. Rosocha. 2020. Material flow optimization through Ekanban system simulation. Int. J. Simul. Model. 19(2): 243-254.
- [42] M. Močilan, M. Žmindák, P. Pecháč, and P. Weis. 2017. CFD simulation of hydraulic tank," Procedia Eng. 192: 609-614.
- [43] Ł. Sobaszek, A. Gola and E. Kozłowski. 2017. Application of survival function in robust scheduling of production jobs. In 2017 Federated Conference on Computer Science and Information Systems (FedCSIS). pp. 575-578.
- [44] L. Durica, B. Micieta, P. Bubenik and V. Binasova. 2015. Manufacturing multi-agent system with bioinspired techniques: CODESA-Prime. MM Sci. J. 2015: 829-837.
- [45] L. Dulina and M. Bartanusova. 2015. CAVE design using in digital factory. Procedia Eng. 100: 291-298.
- [46] R. G. Pearson. 2007. Species' distribution modeling for conservation educators and practitioners. Synth. Am. Museum Nat. Hist. 50: 54-89.
- [47] A. A. Bakar, Z. Kefli, S. Abdullah and M. Sahani. 2011. Predictive models for dengue outbreak using multiple rulebase classifiers. In Proceedings of the 2011 International Conference on Electrical Engineering and Informatics. pp. 1-6.
- [48] T. R. en Hvidsten. 1999. Fault diagnosis in rotating machinery using rough set theory and ROSETTA. Dep. Comput. Inf. Sci. Nor. Univ. Sci. Technol.
- [49] [49] T. Fawcett. 2006. An introduction to ROC analysis. Pattern Recognit. Lett. 27(8): 861-874.
- [50] Z. C. Lipton, C. Elkan, and B. Naryanaswamy. 2014. Optimal thresholding of classifiers to maximize F1 measure. Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics), vol. 8725 LNAI, no. PART 2, pp. 225-239, doi: 10.1007/978-3-662-44851-9_15.
- [51] C. Liu, P. M. Berry, T. P. Dawson and R. G. Pearson, Liu *et al.* 2005 Selectring thresholds of occurrence in the prediction of species distributions.pdf. 3(December 2004): 385-393.
- [52] D. Falessi *et al.* 2018. Empirical software engineering experts on the use of students and professionals in experiments. Empir. Softw. Eng. 23(1): 452-489.

- [53] M. El-Attar. 2019. A comparative study of students and professionals in syntactical model comprehension experiments. Softw. Syst. Model. 18(6): 3283-3329.
- [54] M. Höst, B. Regnell and C. Wohlin. 2000. Using students as subjects-a comparative study of students and professionals in lead-time impact assessment. Empir. Softw. Eng. 5(3): 201-214.