Pakistan Journal of Engineering and Technology, PakJET
ISSN (p): 2664-2042, ISSN (e): 2664-2050
Volume: 5, Number: 2, Pages: 133- 139, Year: 2022

Time Pressure's Impact on Software Quality

Ali Raza*, Habiba Sattar and Nargis Fatima

Software Engineering Department, National University of Modern Languages, Islamabad, Pakistan
Corresponding author: Ali Raza (e-mail: ali.raza4873@gmail.com).

Received: 30/06/2022, Revised: 30/08/2022, Accepted: 31/08/2022

Abstract- **In the software industry, time pressure is unavoidable, and it has an impact on software developers. It could aching programming quality or predisposition impression of execution on got done with responsibilities, making them contrast from actual implementation. Time pressure is frequent in the software industry, where shorter and shorter deadlines and high client demands lead to more tight deadlines. On the other hand, the effects of time constraints have received little attention in software engineering research. This study aims to survey final-year undergraduate students studying software engineering and computer science at Pakistani universities. We select the quality factors from the software quality model, i.e., McCall's classic model – 1977 and FRUP+ – 1992. This study considers these factors: performance, productivity, accuracy, creativity, reliability, efficiency, reusability, interoperability, flexibility, maintainability, integrity, testability, and portability. All selected factors for this study are affected by the time pressure software completed on time and delivered. Still, the quality of the software system is affected and sometimes cause error and issues.**

Index Terms-- **Quality Attributes, Software Engineering, Software Quality, Survey, Software Testing, Time pressure**

I. INTRODUCTION

The time pressure (TP) discernment that the time accessible to get done with responsibility is scant, corresponding to the overly optimistic project timetables are known to have an impact on software quality [1], and they are sometimes referred to as "the single worst foe of software engineering" [2]. Empirical research [3] on the influence of time pressure (i.e., the impression of looming deadlines) has produced mixed results. According to studies [4,] time pressure, for example, negatively and positively impacts work performance. Furthermore, studies have discovered an inverted U-shape association between time pressure and performance [4], implying that the strength of time pressure affects the relationship. Requests of the undertaking are normal in organizational setups.

Time pressure (TP) is predominant in the product business, where increasingly short and high client requests lead to progressively close cutoff times. Be that as it may, the impacts of time pressure stand out in software programming research.

In software program writing, time pressure is regularly connected with adverse results. Time pressure is [3]:

- Deters cautious preparation and defiles a designing norm of value.
- It makes designers and developers pursue faster routes.
- Diminishes the time on software programming exercises.
- It is a de-motivator for programming process improvement.
- Makes disappointment gain from botches.
- Causes lower experiment quality.
- It is a component of burnout in programming groups.

Time pressure concentrate on product management. The executive's area has uncovered that designers working under time limitations be guaranteed to perform worse, yet rather work quicker and that time tension can hinder programming quality [3]. Numerous parts of time strain's effect on the product advancement process stay unexplained, as proven by these different discoveries [5]. These investigations, for instance, don't consider designers' ability to unveil their failure to accomplish a cutoff time and solicitation of a period of expansion.

While the actual prevalence of time pressure in the software business is unknown, we know that most projects (60-80 percent) have overruns [6]. Because there is always pressure to complete a project on time, and overruns are typical, we can assume that time pressure is ubiquitous in many software development projects.

Issues emerge while engineers, feeling that they are feeling the squeeze to fulfill task time constraints, pursue faster routes in managing unforeseen difficulties. "Easy routes" are confidential choices spurred by a longing to remain on time yet are not to the greatest advantage of the venture. When a choice is made, it may not be sure that unfriendly results will follow, and it is far-fetched that the potential outcomes are completely known to the designer. What is pivotal is that a designer worried about quality would have gone with an alternate confidential decision assuming apparent time pressures were somehow reduced. Shortcuts are not really because of cunning (Brooks 1975), nor

are they essentially the consequence of a conscious choice cycle. Rather, they mirror an engineer's inclinations to pray for divine intervention, to let possible causes of trouble alone, and to decipher necessities helpfully when confronted with time pressures.[7]

Most of the literature is based on experiments studies, and some research was conducted as interviews as the research methodology. We want to conduct research in different ways from which we can conclude results in better ways.

In this study, we survey the final year students from different universities' undergraduate computer science and software engineering students from their final year projects on the TP effect on software quality. We select the quality attributes to form the software quality models, i.e., McCall's classic model 1977 and FRUPS PLUS 1992 as software quality factors. According to the points of these models, we prepare the questionnaire and fill from the final year students of Computer Science and Software Engineering on their final year project. We will discuss the attributes and their feedback in the results section.

II. RELATED WORK

Time pressure-stated as the time accessible to follow through with a job is scant compared to the requests of the undertaking [8,9] is normal in organizational setups [10-12]. Notwithstanding, research is uncertain concerning the connection between time forces and workers' way of behaving and execution [4]. Various outcomes involve markers that time strain and work execution have a positive [9,13], negative [14,15], or a transformed U-moulded relationship [4]. These works additionally recommend that the degree of time pressure is conclusive for the kind of effect: though moderate time pressure is important to guarantee inspiration, high or no time pressure prompts interruption or absence of excitement.

The overall uncertain discoveries concerning time pressure hold for the area of programming advancement also, finding no impact [16], beneficial outcomes [3], or a rearranged U-shape impact [17].

A game-hypothetical model that is nonsensical from the beginning, since it turns the standard leeway put together methodology concerning its head and claims a constructive outcome of more significant levels of time tension on programming quality, has been proposed by Austin [7]. The model conceptualizes the "product quality under time tension" situation. Two specialists (i.e., programming engineers) seek the blessing of a head (i.e., the venture director). This incorporates rewards (e.g., advancements, salary increases, future business) from the head to whom they report at customary spans. The players in the game are these two specialists who freely settle on whether to report their failure to fulfill a given time constraint or to pursue a faster route to meet it. The common cutoff time setting strategy is addressed by the likelihood $p > 0$ of a product designer being stood up to with an unreasonable cutoff time and is taken as given. The focal finish of the model is that a basic worth of p (hence alluded to as p_{crit}) exists; for $p < p_{crit}$, the number of easy routes taken increments with p, though for $p => p_{crit}$, easy routes are kept away from.

As per Austin [7], easy routes "are choices made in private [by programming developers] that are roused by a craving to remain on time, yet are not to the greatest advantage of the undertaking." Alternate routes can prompt genuine programming disappointment during runtime, and programming engineers are typically not (completely) mindful of the potential results of pursuing such faster routes. It means quite a bit to reference that product engineers with worries for quality who pursue a faster route under time tension wouldn't do as such if there should be reduced pressure. Furthermore, it is assumed that product designers don't need to fear individual results while pursuing faster routes since it is challenging for non-subject matter experts (like the head) "to follow complex framework issues to causal sources" [7].

Even though numerous choices in financial aspects and money must be made under serious time tension, the impacts of time pressure are a generally neglected area in these fields. Trial brain science has been keen on those impacts for quite some time, emphasizing the tradeoff between speed and precision of navigation (Woodworth,1989).

Earlier work on time strain on programming undertakings utilizing estimated exertion, viability, and effectiveness is restricted. Our underlying review [1] demonstrated that time pressure diminished adequacy (fewer deformities viewed as altogether) yet expanded productivity (more imperfections figured out per opportunity unit) in programming testing undertakings. Additionally, Topi et al. [16] figured out that more limited opportunity accessible was related to diminished rightness on information base inquiry improvement undertakings. The review found no proof supporting expanded effectiveness (rightness/minute).

In applied psychology, Beilock et al. [18] concentrated on hitting the fairway under directions that either (a) featured exactness with taking as much time depending on the situation or (b) taught to proceed as quickly as conceivable while yet being precise. They found that learners created average speed-precision tradeoffs, i.e., playing quicker diminished exactness. Furthermore, a concentration on bookkeeping space showed that bookkeepers with high information performed better under time tension while low information bookkeepers performed more regrettably [19]. Likewise, a concentration on chess players figured out that under time pressure, the nature of chess moves decreased less for chess aces than for weaker players [20].

Observational investigations of time pressure frequently notice time strain as a negative component. For instance, professionals see it as an obstruction to programming quality [21], as a de-motivator for programming process improvement [22], as a variable of burnout [23], and as a reduction of occupation fulfillment [24].

TABLE I
TIME – PRESSURE (TP) STUDIES IN SE INDUSTRIES

Study	Context & Aim	Key Notes
[25] 2013, Modern case study, Software testing, worldwide programming improvement	No deliberate results (qualitative study)	Time pressure was seen as great and terrible. Test groups experienced more time tension than other groups. GSD eases the negatives of time pressure
[1] 2013, Explore,	Number of	Time pressure increased

Programming testing	imperfections identified	efficiency 71%.
[17] 2009, Industrial Case Study, Software projects	Process duration and exertion	Medium time pressure delivered the most elevated efficiency
[16] 2005, Experiment, Database query Development	Exertion, rightness	Time pressure affected exertion or accuracy
[22], 2003, Industry interview study.	None - Qualitative study)	Time pressure is a main de-motivator for process improvement

III. RESEARCH METHODOLOGY

This research is conducted as quantitative research, and the survey is designed as a questionnaire following the steps followed in the survey, as shown in Fig.1.

In step 1, we plan for the research to complete the introduction about our desired research topic, "Time Pressure effect on software quality," and study different research papers and articles related to our selected topic in the past.

In step 2, we identified the objective and scope of the research study that the effect of the deadline on the quality of the software. In the result, we can conclude the TP impact on the software industry.

In Step 3, we identify and select the quality factor/attributes that are affected due to time pressure. We selected two software quality models (I) McCall's classic model for software quality factors - 1977(ii) FURPS + - 1992 model for the software quality factor. Using these two-quality model factors, we selected 13 quality factors/attributes and made a questionnaire for the survey.

FIGURE 1. Research Methodology.

In Step 4, we select our targeted audience from which we have to fill out our questionnaire. So, we select undergraduate students in their final year or do their final year projects in the field of Software Engineering (SE) and Computer Science (CS) from different universities.

In steps 5 and 6, we make a google form for our questionnaire and share this form to the final year students of undergraduate studies in the department of SE and CS of different universities. Students fill out the questionnaire. In the last step, we get 69 responses to form the final year students. We analyze all the records and prepare the results for every attribute mentioned above, and the outcomes are discussed in detail in the next section.

IV. RESULTS

In this research, we surveyed through a questionnaire from the final year students of undergraduate studies on their final year project from the department of SE and CS from different universities. The questionnaire selects 13 software quality attributes from the software quality model. A total of 69 records received from the students' details are given below:

A. PERFORMANCE

Performance is about asking how efficiently we have done our work or how good was service delivered. In software quality assurance, performance determines how a system performs regarding responsiveness and stability under the workload [26]. The question related to performance is performance increase due to time pressure, shown in Fig.2. These research results are based on a survey. The questioner fills by final-year students of different universities. We get different results from a survey of people. According to our survey, 40.6% of participants agreed that product performance increased due to TP, 23.2% disagreed with the performance increase due to TP, and 24.6% were fair that performance may or may not be affected due to the TP effect.

B. PRODUCTIVITY

Productivity measures the capability or how much a person performs in converting inputs into valuable outputs [26]. The success of a project depends on team productivity. Questions related to productivity are *productivity of the team increase due to time pressure* and result shown in Fig.3. As the result shows 33.3% are fair not agree nor disagree and 26.1% disagreed productivity of the team increased due to time pressure and 29% are agreed that productivity of team increases due to TP effect.

Another question related to productivity is *whether code refactoring aims to improve software or code quality.* Results are shown in Fig.4. From the result, we can say that code refactoring improves the software quality results percentages are 55.1% agreed, 26.1% are fair (neither agree nor disagree), and 15.9% participants disagree with this.

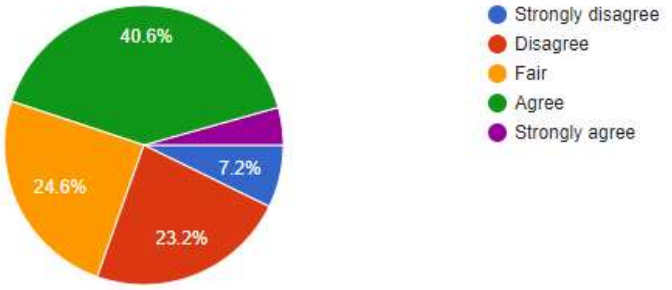

FIGURE 2. Time pressure effect on performance.

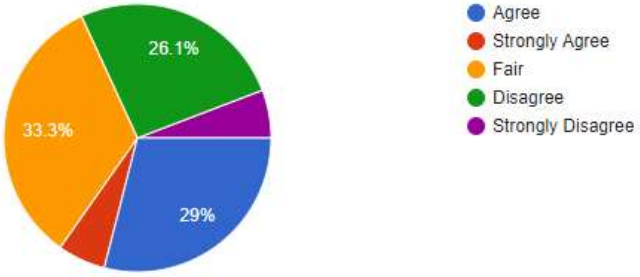

FIGURE 3. Time pressure effect on the productivity of the team.

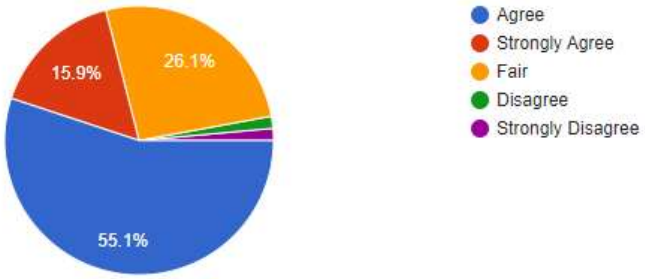

FIGURE 4. Time pressure effect on Code Quality.

C. CORRECTNESS

Correctness describes the ability of software products to perform their exact tasks, as defined by their specification. Our software system works properly/correctly in abnormal conditions [26]. Question-related to correctness is the *accuracy of work decrease due to time pressure* results are shown in Fig.5. Results gained from the survey about the accuracy of the system regarding time pressure effect are 44.9% are agreed, 23.2% are fair (neither agree nor disagree),20.3% are strongly agreed, and 11.6% disagree that accuracy effected by the TP.

D. CREATIVITY

Creativity generates useful possibilities in solving problems and developing new ideas [26]. The question related to creativity in the questionnaire is whether Time pressure reduces *creativity*. Results from the survey are shown in Fig.6. It stated that 37.7% agree, 26.1 strongly agree, 20.3% are fair (neither agree nor disagree), and 14.5% of participants disagree that time pressure reduces creativity in software development.

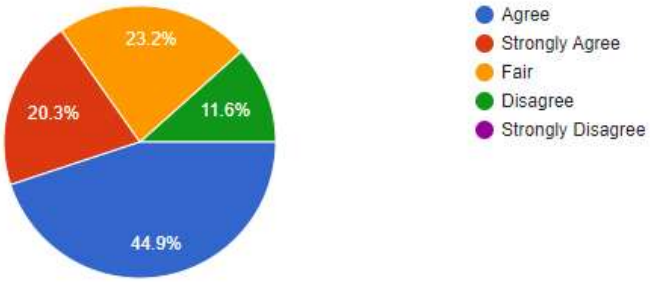

FIGURE 5. Time pressure effect on Accuracy / Correctness.

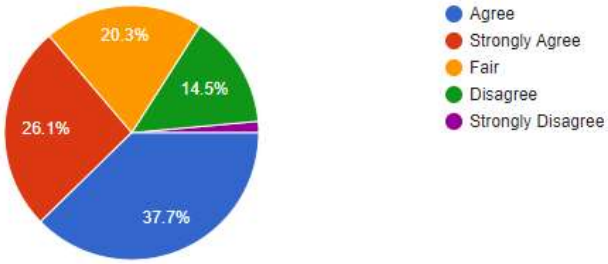

FIGURE 6. Time pressure affects creativity.

Another question from the questionnaire related to creativity is People under time pressure do not solve problems creatively. Results about this question are shown in Fig.7. These results indicate that problem-solving in software development is creatively affected by the time pressure effect, and its statistics are 31.9% agree, 27.5% fair (neither agree nor disagree), 23.2% strongly agree, and 14.5 have countered with our question.

E. RELIABILITY

The following list outlines the different types of graphics. Reliability means that software performs its functionality consistently in any environmental condition and in a specific period [26]. The question is whether *Time pressure affects the response time of the software* results graph shown in Fig.8.

Reliability is an important factor; without good reliability, it is irrelevant artifacts increase. As a result, 39.1% agree, 34.8% are fair (neither agree nor disagree),14.5% strongly agree, and 14.5 disagree that reliability is affected by the pressure effect on the response time of software.

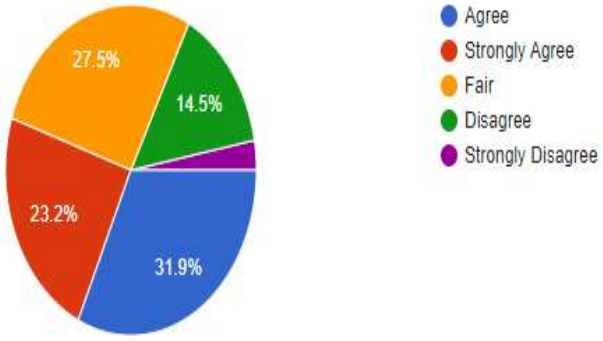

FIGURE 7. Time pressure effect on Problem Solving Creativity.

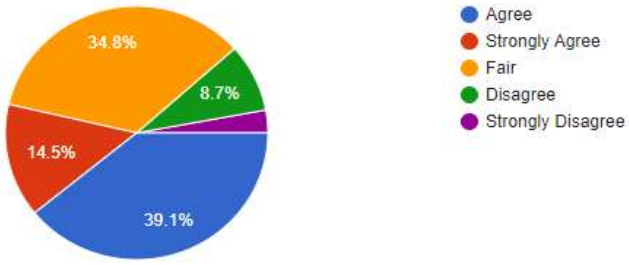

FIGURE 8. Time pressure affects reliability.

F. EFFICIENCY

Efficiency means achieving goals or results; if the software system is efficient, its means that all processes are optimized [26]. The question from our survey related to efficiency is *People under time pressure do not necessarily work more efficiently*. Results graph shown in Fig.9. Results stated as 39.1% agree, 27.5% fair, 18.8% strongly agree and 13% disagree about the impact of efficiency due to TP.

G. INTEGRITY

Integrity refers to methods to ensure that data is real, accurate, and safeguarded from unauthorized user modification [26]. The question related to integrity included in our survey is the *security of software compromise due to time pressure.* The results graph shown in Fig. 10 shows that 46.4% agree, 31.9% are fair, 11.6% disagree, and 10.1% strongly agree with the statement that security is compromised due to time pressure in the software development industry.

H. TESTABILITY

Testability means finding and isolating faults is easy [26]. Question-related to testing is that *the number of defects increases due to time pressure.* The resulting graph related to this statement is shown in Fig.11. A high degree of testability helps you understand more quickly what is wrong. If the testability of the software artifact is high, then finding faults in the system (if it has any) using testing is easier. As results show, 39.1% agreed, 29% were fair, 21.7% strongly agreed, and 10.1% disagreed with the statement that the number of errors or failure increases due to the time pressure effect in software organizations.

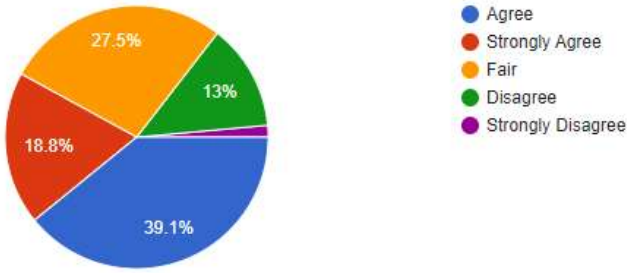

FIGURE 9. Time pressure effect on efficiency.

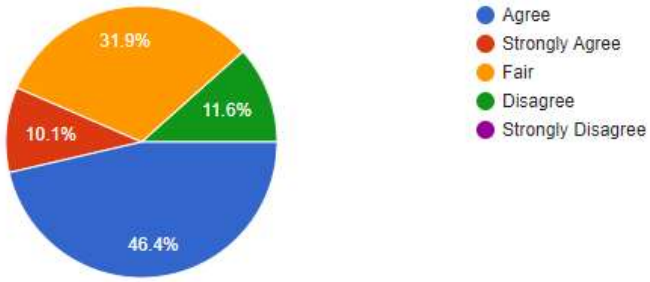

FIGURE 10. Time pressure effect on Security / Integrity.

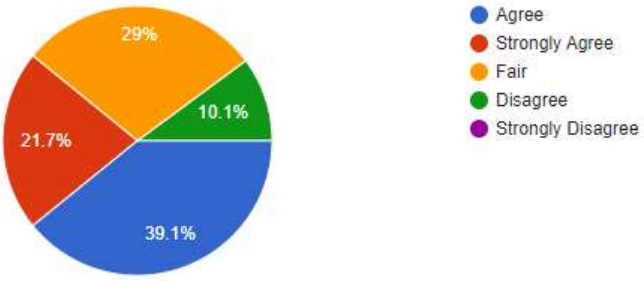

FIGURE 11. Time pressure effect on testability.

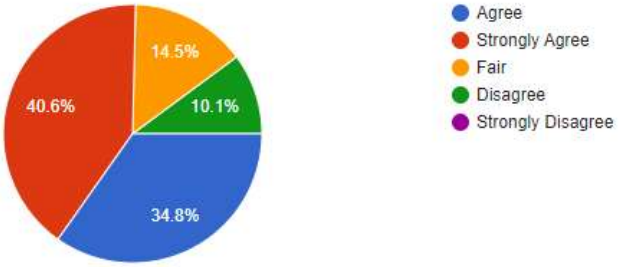

FIGURE 12. Time pressure effect on maintainability.

I. MAINTAINABILITY

The degree to which application can easily understand, repaired, or enhanced [26]. The question from our questionnaire is *Time pressure leads to shortcuts.* The resulting graph regarding maintainability is shown in Fig.12. Maintainability is important in any project because approximately cost is related to it. As a result, 40.6% strongly agree, 34.8% agree, 14.5% are fair, and 10.1% disagree that time pressure leads to shortcuts.

J. FLEXIBILITY

Flexibility means that the software system easily responds to external and internal changes to different users and requirements [26]. The question is the *Adaptation of software systems to other environments ignored due to time pressure* Result graph shown in Fig.13. Results stated as 44.9% agree, 34.8% are fair, and 7.2% strongly agree with our question that changes or updates in the software system ignore due to the effect of time pressure in the software fields.

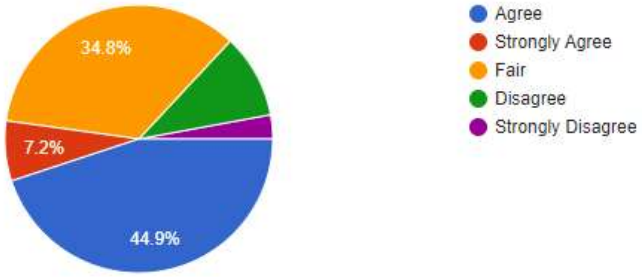

FIGURE 13. Time pressure effect on flexibility.

K. PORTABILITY

The portability of software means that software can easily adapt to the environment when moving from one environment to another [26]. Question-related to portability is the *portability of software system effect due to time pressure.* The resulting graph is shown in Fig. 14. The software requires much environment-related change; if the system is portable, it requires low effort to run on different platforms. As a result, from the survey, 42% are agree, 31.9% are fair, and 13% strongly agree with the portability of software systems affected due to time pressure.

L. REUSABILITY

Reusability uses existing assets within the software product development process [26]. Given the question is *Reusability factor is not focusing due to time pressure.* The resulting graph is shown in Fig.15. Reusability reduces the duplication of effort and enhances reliability. As a result, 43.5% agree, 27.5% are fair, 11.6% are strongly agreed and 7.2% are strongly disagree with the reusability factor that does not focus during software development due to the time pressure effect.

M. INTEROPERABILITY

Interoperability" refers to different solutions' capability to communicate freely and easily [26]. Our question related to interoperability is *Capabilities and effort effect due to time pressure.* The graph of the result is shown in Fig.16. The interoperability allows the system to communicate easily and understand the information they pass to one another. As a result, 44.9% agree, 29% are fair, 15.9% strongly agree and 8.7%

disagree with the statement that efforts and capabilities are affected due to time pressure.

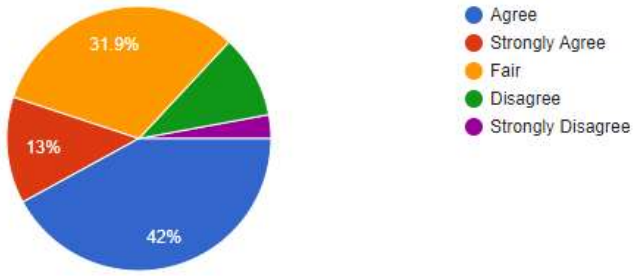

FIGURE 14. Time pressure effect on portability.

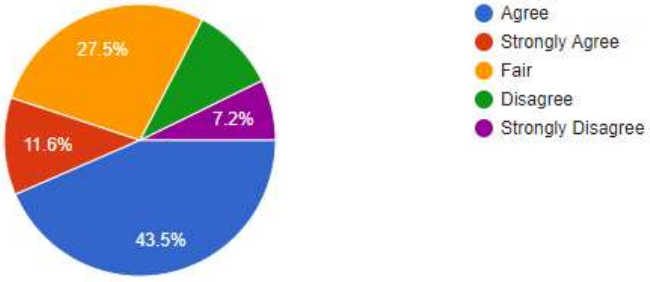

FIGURE 15. Time pressure effect on reusability.

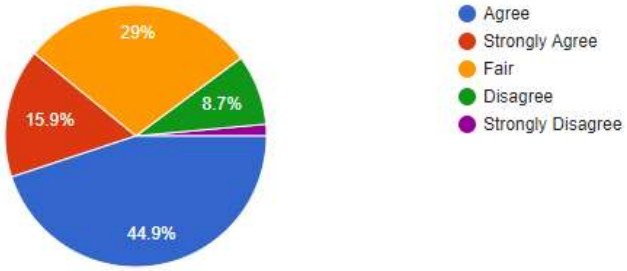

FIGURE 16. Time pressure effect on interoperability.

V. DISCUSSION

The Time pressure affects software quality. Time pressure negatively inputs the quality of the software. Due to time pressure, engineer focus on project completion rather than the quality of the software. In this way, the quality of the project decreased. Software quality is mostly neglected when the project is completed on a short deadline. The main concern is to deliver the project on time. Time pressure maybe teamwork well, but in most cases, developers just ignore the software quality.

We select the quality attributes to form the software quality models, i.e., McCall's classic model 1977 and FRUPS PLUS 1992 as software quality factors. Factors include Product Operation (Correctness, Reliability, Efficiency, Integrity, Usability, performance, productivity, design), Product Revision (Flexibility, testability, supportability), and product transaction (Portability, Reusability, Interoperability). Based on the result, we conclude that the team's productivity increased due to time pressure.

Productivity of the team is important for the competition of any project in a time when people follow any schedule or work breakdown structure, the chance of affections due to time pressure on software quality decreases. Performance is also affected due to time pressure 40.6 % of participants in the survey agreed with that statement. When people work under pressure, the accuracy of work also decreases, affecting the software's quality. The creativity of the team also decreased, and people were not fully focused or not creative to solve any problem effectively. The team does not work efficiently when everyone on the team has a clear role, it means they have less overlap, and teamwork is the more productive way.

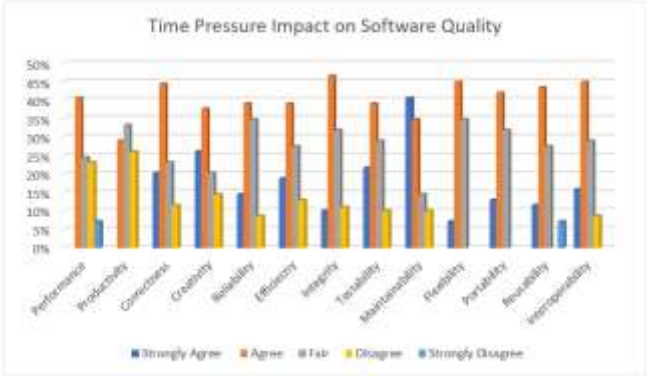

FIGURE 16. Time pressure effect on Software Quality Factors.

When people work under pressure, the chance of increasing defects in the software also increases when defects decrease in software or a project gains more attention from users. The quality is not more compromised due to time pressure. Under time pressure, developers compromise on software quality and are concerned about the quality increase in most industrial sectors. The effect of time pressure is mostly identified during quality assurance. People under time pressure do not necessarily work efficiently. They may reduce the amount of work necessary to be done, which may lead to the software's worse quality, mostly in those cases when the performance cannot be observed or verified.

Most of the projects were overrun due to time pressure Adaptation (updating) of software from one environment to another is another factor that also affected its means that software system is not flexible and the chance of that affection on software quality increased. The accuracy of the software system is affected due to the TP effect developers focus on completing the project on time. They do not focus on accuracy, and testing of the software system is not occur properly, which sometimes causes a large failure of the software.

Reusability and interoperability are other quality attributes from the software quality model considered in our research. These two factors are all affected due the TP effect. Code in the software system is not properly structured and does not focus on the later use of the software; similarly, it is not part to attention by the software developer to interoperability factor system can be used with any other third-party software. All the factors mentioned in this research study are somehow effected due to the time pressure directly affecting the software system's quality.

VI. CONCLUSION

This study looked into the effects of time constraints on software quality factors according to McCall's and FRUP PLUS software quality model. This research study is conducted with a survey of the final year student of undergraduate studies at different universities from the department of Software Engineering and Computer Science student on their FYPs (Final Year Projects).

Data collection is performed through google Forms 69 students filled the questionnaire details results according to each quality factor discussed in the Results section. Different 13 factors are followed in the questionnaire.

Performance of the software increase due to the time pressure effect. The team's productivity is not affected directly due to the time pressure effect—accuracy in the software system decrease, and error and defects in the system increase. Due to the time pressure effect, the developer just focuses on completing the requirements and delivering the project to the client on time. Still, in this way, the developer ignores the important quality factor of the software system, such as reusability, maintainability, interoperability, security, efficiency, testability, and integrity. All these factors are majorly neglected during development due to tight deadlines. The software was completed on time and delivered to the client, but this software is not maintainable and has defects and issues.

VII. FUTURE WORK

In the first step, this study conducts with final year students of different universities of undergraduate studies on their final year projects. In the next step, we will conduct the time pressure effect on the software quality from intermediated level software developers in the software organizations and compare all these researches. We will briefly analyze the comparison of the time pressure impact on software quality.

FUNDING STATEMENT

The author(s) received no specific funding for this study.

CONFLICTS OF INTEREST

The authors declare they have no conflicts of interest to report regarding the present study

REFERENCES

[1] Mäntylä, M. V., & Itkonen, J. More testers–The effect of crowd size and time restriction in software testing. *Information and Software Technology*, vol. *55*, *no.* 6, p.986-1003, 2013.

[2] Costello, S. H. Software engineering under deadline pressure. *ACM SIGSOFT Software Engineering Notes*, vol. *9*, *no.* 5, p.15-19, 1984.

[3] Mäntylä, M. V., Petersen, K., Lehtinen, T. O., & Lassenius, C. Time pressure: a controlled experiment of test case development and requirements review. In *Proceedings of the 36th International Conference on Software Engineering* 2014, (pp. 83-94).

[4] Maruping, L. M., Venkatesh, V., Thatcher, S. M., & Patel, P. C. Folding under pressure or rising to the occasion? Perceived time pressure and the moderating role of team temporal leadership. *Academy of management journal*, vol. *58*, no. 5, p. 1313-1333, 2015.

[5] Siau, K., Long, Y., & Ling, M. Toward a unified model of information systems development success. *Journal of Database Management (JDM)*, vol. *21*, no. 1, p. 80-101, 2010.

[6] Molokken, K., & Jorgensen, M.. A review of software surveys on software effort estimation. In *2003 International Symposium on Empirical Software Engineering, 2003. ISESE 2003. Proceedings.* 2003, September, (pp. 223-230). IEEE.

[7] Austin, R. D. The effects of time pressure on quality in software development: An agency model. *Information systems research*, vol. *12*, no. 2, p. 195-207, 2001.

[8] Cooper, C. L., Cooper, C. P., Dewe, P. J., Dewe, P. J., O'Driscoll, M. P., and O'Driscoll, M. P. Organizational stress: A review and critique of theory, research, and applications, 2001.

[9] Kelly, J. R., and McGrath, J. E. Effects of time limits and task types on task performance and interaction of four-person groups. *Journal of personality and social psychology*, vol. *49*, *no.* 2, p. 395, 1985.

[10] Gersick, C. J. Time and transition in work teams: Toward a new model of group development. *Academy of Management Journal*, vol. *31*, no.1, p. 9-41, 1988.

[11] Gevers, J. M., Rutte, C. G., and Van Eerde, W. Meeting deadlines in work groups: Implicit and explicit mechanisms. *Applied psychology*, vol. *55*, no. 1, p. 52-72, 2006.

[12] Waller, M. J., Zellmer-Bruhn, M. E., and Giambatista, R. C. Watching the clock: Group pacing behavior under dynamic deadlines. *Academy of Management Journal*, vol. *45*, no. 5, p. 1046-1055, 2002.

[13] Chong, D. S., Van Eerde, W., Chai, K. H., and Rutte, C. G. A double-edged sword: The effects of challenge and hindrance time pressure on new product development teams. *IEEE Transactions on Engineering Management*, vol. *58*, no.1, p.71-86, 2010.

[14] Karau, S. J., and Kelly, J. R. The effects of time scarcity and time abundance on group performance quality and interaction process. *Journal of experimental social psychology*, vol. *28*, *no.* 6, p. 542-571, 1992.

[15] Perlow, L. A. The time famine: Toward a sociology of work time. *Administrative science quarterly*, vol. *44*, no.1, p. 57-81, 1999.

[16] Topi, H., Valacich, J. S., and Hoffer, J. A. The effects of task complexity and time availability limitations on human performance in database query tasks. *International journal of human-computer studies*, vol. *62*, no.3, p. 349-379, 2005.

[17] Nan, N., & Harter, D. E. Impact of budget and schedule pressure on software development cycle time and effort. *IEEE Transactions on Software Engineering*, vol. *35*, no. 5, p. 624-637, 2009.

[18] Beilock, S. L., Bertenthal, B. I., Hoerger, M., et. al.. When does haste make waste? Speed-accuracy tradeoff, skill level, and the tools of the trade. *Journal of Experimental Psychology: Applied*, vol. *14*, *no.* 4, p. 340, 2008.

[19] Spilker, B. C. The effects of time pressure and knowledge on key word selection behavior in tax research. *Accounting Review*, 49-70, 1995.

[20] Calderwood, R., Klein, G. A., and Crandall, B. W. Time pressure, skill, and move quality in chess. *The American journal of psychology*, p. 481-493, 1988.

[21] Wilson, D. N., & Hall, T. Perceptions of software quality: a pilot study. *Software quality journal*, vol. *7*, *no.* 1, p. 67-75, 1998.

[22] Baddoo, N., and Hall, T. De-motivators for software process improvement: an analysis of practitioners' views. *Journal of Systems and Software*, vol. *66*, no.1, p. 23-33, 2003.

[23] Sonnentag, S., Brodbeck, F. C., Heinbokel, T., et. al.. Stressor-burnout relationship in software development teams. *Journal of occupational and organizational psychology*, vol. *67*, *no.* 4, p. 327-341, 1994.

[24] Linzer, M., Konrad, T. R., Douglas, J., et. al. Managed care, time pressure, and physician job satisfaction: results from the physician worklife study. *Journal of general internal medicine*, vol. *15*, *no.* 7, p. 441-450, 2000.

[25] Shah, H., Harrold, M. J., & Sinha, S. Global software testing under deadline pressure: Vendor-side experiences. *Information and Software Technology*, vol. *56*, *no.* 1, p. 6-19, 2014.

[26] Galin, D. *Software quality: concepts and practice*. John Wiley & Sons, 2018.