**1.0 INTRODUCTION**

Traditionally, projects did not require high levels of standardization and hence there were rare changes in components and completion process. Nonetheless, the effects of globalization have led to an increase in standardization process requirements and particularly in international companies. The field that has realized an increase in change the most is a software development and its processes. With the increased demand for technologies, software development has played an integral role in meeting the needs of people and organizations. However, software development systems in modern society have faced a high level of diversity. Subsequently, with the increasing need to accomplish the requirements of the software projects simultaneously, diversity becomes inevitable.

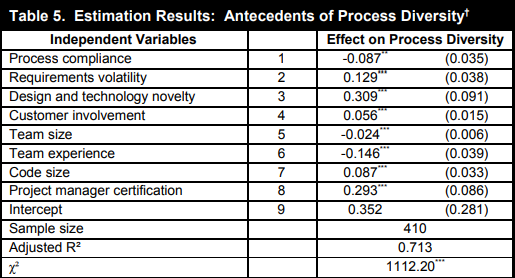
During the early stages of the software development process, there must be a plan for possible system diversity before the actual start. The software development process is the condition of a project that arises out of the need to simultaneously adopt multiple software development processes in a single project (Ramasubbu et al., 2015). The diversity in software process development is a result of different factors. Past studies have reported that diversity in the software development process is a result of external and internal factors. Ramasubbu et al. (2015) noted that the software development process is a result of customer direct involvement, level of compliance to international standardization, volatility in requirements, design, and technological novelty. Therefore, this research reports the conceptualization and measurement of the software process diversity to a software developer. It also analyses the impacts of diversity on the performance of the software development process. The second part discusses the background information on the software development process. Next, there are methods, findings, and discussion. The last section deals with the conclusion of the work.

**2.0 BACKGROUND INFORMATION**

In this part of the research, there is in-depth discussion of the software development and process framework. Further, the development process addresses the individual factors that contribute to the success or failure of the process. It also looks into the subject of research.

**2.1 Software Development and Process Framework**

Lately, the software development process has become an integral step that positively affects the conditions and the product used in the market. Software development refers to a stepwise process of conceptualizing, specifying, designing, analyzing, programming, scheduling, documenting, testing and error fixing entailed in the creation and maintenance of applications and software components. The development and maintenance of software involve different activities. Therefore, in order to maintain the activities during the software development, the appropriate framework is needed. Edwin (2014) opined that software frameworks have powerful tools that enable developers to create flexible and error-free applications. Other than the provision of tools, software frameworks also speed the development process through the provision of necessary functionalities (Edwin, 2014).



**Table 1.** Effects of factors in process diversity (Narayan& Anandhi & Giri, 2015, p801)

The data above shows the factors with both positive and negative effects on the process diversity. These independent variables affect the process diversity differently. It clearly shows that demand volatility, design novelty, and customer participation all have a positive and statistically significant impact on process diversity. The results show that an increase in the percentage of demand volatility will lead to an increase in process diversity by about thirteen percentage points (B=0.129; p<0.001) and it can be seen that for every unit increase in design and technological novelty scores, process diversity will increase by about thirty-one points. percentage points (B=0.309; p <0.001) and so on.

**2.2 Subject of the Research**

Software development often serves different purposes in organizations. First, the software development process has been an important step that affects the status and use of the product in the market. In this regard, companies are increasingly adopting multiple process frameworks for their projects and embracing the diversity of software processes. However, under what circumstances can the increased software process diversity in response to an emergency in a project improve project performance. What is software process diversity is explained in 2015 by Narayan, Anandhi, and Giri that "software process diversity refers to the project conditions produced by using multiple software development process frameworks in a single project at the same time"(p.878). Therefore, it is first necessary to study and analyze what factors contribute to the increase in software process diversity through the data and results obtained. So, this research as a final report focused on what factors help increase the diversity of software processes in real-world projects, as well as the fit of software process diversity and process compliance with project performance.

In 2015, an evaluation by Narayan, Anandhi, and Giri mentioned a concept "When the software team adopts and integrates multiple standardized process frameworks in a single project, they will increase the diversity of the overall software process of the project" (p.789). The concept of software process diversity arises as a result of the project team's response to emergencies such as demand fluctuations, design, and technical novelty, customer participation, and compliance with the organizational standards of project process implementation"(p.790). In essence, these pieces of information help to locate the specific direction and concept of the research. The most important thing is that from the experimental results obtained in a table form, it is found that demand fluctuations, design novelty, and customer participation all have a positive and statistically significant impact on process diversity. Similarly, in 2007, Agarwal and Chari explained what demand volatility is "the degree of non-functional and functional changes in the software development process during the project life cycle" (p.145-156).

Further, in 2015, an evaluation by Narayan, Anandhi, and Giri made assumptions 1-4 as follows: First, that there was a significant relationship between requirements volatility of a software development project and increased level of software process diversity (Narayan, Anandhi, and Giri, 2015). Such revelations are helpful for the completion of this report.

**2.3 Purpose of Research**

In this research report, the purpose is to conceptualize and measure the software process diversity. Moreover, it’s also aimed at investigating the impacts of diversity factors on the performance of the software development process. Through past research studies, it was important to understand the increase in an organization’s compliance work processes. There is a high expectation that an increase in the diversity of the software process in response to project emergencies has the potential to improve the performance of a project. That is, in the process of responding to emergencies, it is expected that addressing the complexities would lead to a positive outcome.

**3.0 METHODS**

The methodology is an important part of research study. It contributes to the success of the study in ways such as choosing the correct design, and sampling method.

**3.1 Research Design**

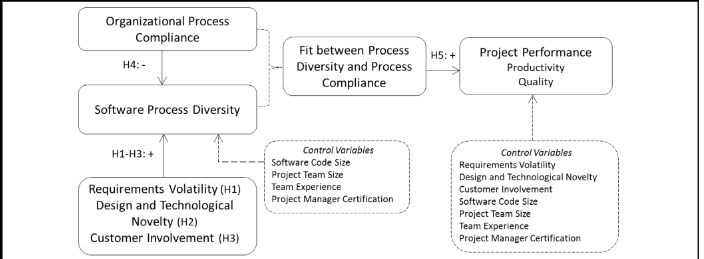
In addressing the topic of this research report, a mixed research approach was used. In the mixed research approach, both qualitative data and quantitative sets of data were used. In the qualitative data, past literary research findings formed part of the investigation of the factors contributing to the complexity of software process diversity. Credible scientific research studies were located from reputable articles and journals. On the other hand, quantitative data was located from one of the studies.

**3.2 Data Sampling**

Data sampling was an important process in the investigation of the factors leading to the conceptualization and measurement of the software process diversity. The main source of data for this report included studies by Ramasubbu et al. (2015).

**4.0 RESULTS AND DISCISSIONS**

The table below reports on the results of the impacts of different factors on the software development process diversity. The factors were categorized into independent variables and dependent variables. In the independent variables, the factors included process compliance, requirements volatility, design and technology novelty, customer involvement, team size, and certification.



**Figure 1.** Software Process Diversity Model (Ramasubbu et al., 2015, p.792)

In the model above, the control variables affect the performance of the project through impacts on the quality and productivity.

**4.1 Process compliance**

First, process compliance had a negative and statistically insignificant impact on the software process diversity. From the results in the table above, it is evident that a decrease in process compliance (1: -0.087, 0.035, p>0.001) led to a decrease in process diversity. The percentage decrease was at 3.5%. Process compliance refers to the mechanisms and procedures that are enacted to comply with applicable standards and ensure there is software design process fidelity to the process frameworks. An increase in compliance efforts is a stabilizing force that prevents process diversity since it blocks variations in processes deemed non-compliant, unnecessary, and unfit (Ramasubbu et al., 2015).

**4.2 Requirements volatility**

This factor had a positive and statistically significant impact on the software process diversity. An increase in the requirements volatility led to an increase in the software process diversity. The percentage increase in the diversity and thus complexity of the software development process was at 0.129 (2:0.129, p<0.001). From these findings, it is important to note that the volatility of the requirements during the software development process cannot be ignored at all. The requirement volatility has been defined as the extent of change in non-functional and functional requirements during the life span of a software system (Ramasubbu et al., 2015). It has been found that unexpected changes in requirements and subsequent alterations have been reported to cause pressure in projects. The results impact of such pressure is the creation of ripple effects leading to project reworks.

**4.3 Design and Technology Novelty**

The third factor was the design and technology novelty of the software development process. The finding shows that design and technology novelty had a positive and statistically significant impact on the software process diversity. It implies that the existence of the design and technology novelty factor during the development process would significantly lead to process diversity (3:0.309, p<0.001). According to Ramasubbu et al. (2015), higher levels of existence of design and technological novelty increase additional requirements on software development teams. Such additional requirements automatically trigger software process diversity. Furthermore, Ramasubbu et al. (2015) also reported that in the performance of a project, the existence of design and technologies contributed to project complexity and hence impacting performance negatively through increased levels of risk in management.

**4.4 Customer Involvement**

Customer involvement is equally a factor that also affected process diversity. It also had a positive and statistically significant impact on the software process diversity. It means that an increase in the involvement of the customers led to changes in elements and this significantly led to increased levels of process diversity (3:0.056, p<0.001). Projects with high involvement of the owners have high chances of success due to lack of disruptions. As Ramasubbu et al. (2015) put it, the success of such projects is linked to the software teams’ ability to match all the processes to the needs of the customers/clients at the early stages of the development process.

**4.5 Team Size**

Team size had negative impacts on the process diversity. The impact of the team size was statistically insignificant on the process diversity. It means an increase in the size of the team during the software process development did not lead to process diversity (5: -0.024, p>0.001). Proper composition of a team is important in the development process of software projects. Agrawal and Chari (2007) also mentioned that program size enables excellent estimations of the defects in a project. More importantly, small groups ensure there is effective communication among the team members for the successful completion of the projects (Ramasubbu et al., 2015). In addition, small groups enjoy flexibility in the team and hence leading to the effectiveness of the process development (Ramasubbu et al., 2015).

**4.6 Team experience**

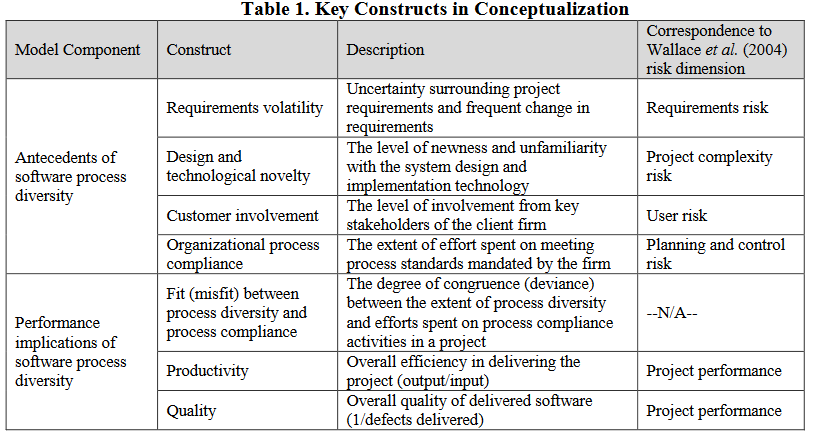
There is also a factorial consequence of the team experience. Statistically, this factor did not have any significant impact on the process diversity. An increase in team experience led to negative impacts and led to a decrease in software development process diversity (6: -0.0146, p<0.001).

**4.7 Code size**

Code size is equally crucial in influencing software process diversity. Its increase causes positive and statistical increases in the software process. The results of the study indicate that code size caused a percentage increase in process diversity by 0.033 (7:0.087, p<0.001).

**4.8 Project Manager Certification**

The certification of the project manager also plays an important role in influencing the diversity of the software process. It shows that an increase in demand for certification of the project manager causes some level of complexity and thus increases the diversity of the software development process. In the aforementioned results, the project manager certification increased process diversity by 0.086. Certification leads to the introduction of different variables and thus leads to the complexity of the process development.

****

**Table 2.** Key Constructs in Conceptualization

As shown in **Table 2**, these key constructs in conceptualization lead to different risks that compromise success of the software development process.

**5.0** **RECOMMENDATIONS**

While there is a wide application of normative models, it is also a fact that not all projects are the same. There are variations in projects and this implies that different software process projects should be accorded unique and distinct approaches. In addition, it is also important that future research consider evaluation of these variables in comparative analyses in projects in different settings.

**6.0 CONCLUSION**

Diversity increases the complexity of the software development process. At the same time, it

is through diversity that the software processes can overcome the complexities and thus reducing project failures. In modern organizations, there are changes in activities and requirements and this triggers diversity in process diversity.

**REFERENCE LIST**

Agrawal, M. & Chari, K. (2007). Software effort, quality, and cycle time: A study of CMM level 5 projects. *IEEE Transactions on software engineering*, *33*(3), 145-156.

Edwin, N.M. (2014). Software frameworks, architectural and design patterns. *Journal of Software Engineering and Applications*, *2014*.

Kaur, R. & Sengupta, J. (2013). Software process models and analysis on failure of software development projects. *arXiv preprint arXiv:1306.1068*.

Liu, L., Kong, X., & Chen, J. (2015). How project duration, upfront costs and uncertainty interact and impact on software development productivity? A simulation approach. *International Journal of Agile Systems and Management*, *8*(1), 39-52.

Ramasubbu, N., Bharadwaj, A., & Tayi, G. K. (2015). Software Process Diversity. *Mis Quarterly*, *39*(4), 787-808.

Varajão, J. (2021). Software Development in Disruptive Times: Creating a software solution with fast decision capability, agile project management, and extreme low-code technology. *Queue*, *19*(1), 94-103.