

Analysis of the Relationship Between Sustainability and Software Performance

Koray Cirak, Istanbul Technical University, Turkey

Hur Bersam Sidal Bolat, Istanbul Technical University, Turkey*

ABSTRACT

Sustainability problems are getting more critical and increasingly threatening human life. Software, which is developing rapidly and entering into every aspect of our lives, is one of the most fundamental components of the technological society. The widespread use of software applications and limited natural resources has led researchers to focus on research that will ensure sustainability in the software development process. In this study, the authors conducted a questionnaire study concerning the sustainability factors that affect the software development process. Then the effect of these factors and the level of education, age, and experience of the people involved in the software development process on the software performance were investigated. As a result, it has been determined that the factors affecting the software development process in terms of sustainability and the descriptive attributes of the individual have an effect on software performance.

KEYWORDS

Path Analysis, Software Development Process, Software Performance, Sustainability

1 INTRODUCTION

The rapid increase in the world population has caused the rapid consumption of natural resources and the increase of environmental pollution problems. This situation puts the benefiting from natural resources in order to sustain the lives of future generations in danger. Information technologies (IT) and software systems, which take place in every area of our daily life, also have important responsibilities in this regard. Since software has become an important part of our daily life, it is impossible to imagine the future without software. The biggest goal of sustainable IT is to find solutions to increasing energy consumption and electronic waste problems. For this reason, an efficient software and software development process is inevitable in terms of energy, efficiency and performance. In order to develop such software, software developers and people involved in the software development process need to know how to achieve sustainable software development and have appropriate tools to achieve this goal. Especially in the software development process, concepts related to sustainability should be clarified. After determining the concept of sustainability and other factors such as maximum amount of task per unit time, the amount of resources used to fulfil a request and maximum number of users that can continue to work as desired; it is possible to manage the software development process more effectively in terms of sustainability.

1.1 Purpose

In the first part of the study, it is aimed to analyse and reveal what criteria sustainability is evaluated by these people, who take an active role in the software development process, during the software development phase. In line with this purpose, the sustainability factors affecting the software development process have been determined by investigating which dimensions of sustainability are involved in the software development process and which ones already are applied.

In the second part of the study,, it is aimed to investigate the effects of sustainability factors and age, experience and educational status of individuals who take role in software development process on software performance. After that, the statistical significance of the hypotheses created has been tested.

1.2 Literature Review

With the increasing demand for complex software applications, serious negative effects on the environment have started to occur in the field of Information Communication Technologies (ICT) due to the increasing resource and power consumption. ICT's impact on sustainable development, especially in software, is one of the most popular topics in the field of Green Computing (Dick et al., 2010), (Diamond, 2005). Sustainable development means that people use resources optimally to meet their vital needs, taking into account ecological, economic and social impacts. Although efficient solutions are tried to be found for the environment in the field of ICT, Whether the energy and resource savings provided by ICT will exceed the amount of resources consumed in this area is not certain. this regard.

Many studies on Sustainable Information and Communication Technology have focused on environmental sustainability and are mainly addressed in terms of computer hardware. However, examining energy consumption issues in software can be a more important aid in achieving sustainable computing. As well as in the hardware area, the software is also responsible for CO₂ emissions. The software has an indirect impact on the environment as it operates and manages the hardware it runs on. Some software-based sustainability solutions can efficiently track and use resources. Other solutions can be sustainable, limiting the need to add more hardware due to updates. But these are not quite enough. Unfortunately, there is a lack of models and studies in terms of sustainability in the field of computer software and software development processes. This study aimed to give the literature a model that determines the factors and their degree of influence on the sustainable software development process and measures the impact of sustainability factors on software performance.

Shalabh et al. (2012) suggest that due to the increase in power consumption and daily carbon emissions as a result of the use of IT infrastructure, gathering requirements with electronic tools, using teleconferencing instead of physical interviews, shutting down the application or computer when not in use, constructing efficient algorithms and data structures and using virtualized systems.

Fábio et al. (2014) suggest that in order to minimize the amount of energy consumed by applications, each thread be run on only one core and other cores should be closed.

Sunil Kumar et al. (2015) suggest that software languages that consume less energy should be used. For example, it is stated that Java can be preferred because it consumes less energy than C, C ++, FORTRAN and PASCAL. In addition, it is recommended to use Integrated Development Environments (IDE) that help programmers to apply green software and tools to measure the sustainability of codes.

Capretz (2014) stated that software application is a product resulting from human activities, and because a human is less predictable and more complex than software, human activities will create complex dynamics in the software development process depending on their descriptive properties, and therefore cannot be considered independent from any concept that affects the software development process. For this reason, human qualities belonging to individuals involved in the software development process are included in the structural equation model in our study.

Akinyele et al (2021) suggest that one thing is to design and implement an energy system, another crucial aspect is the capability to sustain what has been implemented. This is why the concept of sustainability needs to be given adequate considerations in energy systems planning an design, which

is practical in terms of covering all the associated with the system, including the success and the failure factors.

Almadi et al (2021) stated that two dimensions of sustainability are affected by design patterns, the technical dimension and the environmental dimension. Design patterns greatly impacts software understandability, testability, usability, code testability, change in size, frequency, and defects, further affecting program understandability and maintainability. This result is a direct threat to the technical dimensions of software sustainability. Besides, the occurrences of bad smells in design patterns could increase the energy consumption of software applications which, in turn, affects the stability of the green dimension of software sustainability.

Amrit et al. (2014) emphasized that since people work individually or as a team during the software development process, whether this process is successful depends deeply on the characteristics that define people, that is, the qualities of people. In our study, H1, H2 and H3 hypotheses were created based on these inferences. These hypotheses were established to identify the relationship between age, education, and experience and software performance. Age, education and experience were chosen to represent the descriptive characteristics of individuals.

2 SUSTAINABILITY

“Sustainability” is the work done to protect the ecosystem and sustain diversity while producing everything needed for people to survive (United Nations General Assembly, 1987). In other words, it means that meeting our own needs without jeopardizing source of future generations. The key idea behind this concept is innovation and proactive decision making with the objective of maintaining a balance between ecological stability, economic growth, political justice, and cultural vitality to make sure that the world is harmonious for all inhabitants (Purvis et al, 2019). Throughout human history, there are numerous examples of civilizations that have harmed their environment and endangered their chances of survival (Diamond, 2005). Today, many natural resources are used to sustain modern life. In this sense, sustainability explores the ways we can live in harmony with the universe, protecting our world from damage and destruction. Sustainability, as a science, practice, and movement, has made significant intellectual progress beyond unfettered environmental destruction for the sake of economic growth. The field has expanded and adapted over time as more voices have contributed to theory and practice (Moschis et al, 2020).

There are three key dimensions of sustainability: Environmental sustainability, economic sustainability, and social sustainability. Environmental sustainability focuses on non-renewable resource depletion, pollution, and the use of renewable resources. Economic sustainability focuses on issues related to levels of economic growth, production, and consumption that can be supported indefinitely. Social sustainability focuses on the ability of the social system (community, country, or the whole of humanity (Purvis et al, 2019). The three dimensions of sustainability are interdependent and reciprocally influenced. For example, economic development leads to society’s overconsumption that impacts the environment and level of social well-being. Therefore, to address issues of sustainability, one must focus on all three dimensions (Moschis et al, 2020).

2.1 Sustainability for the Software Industry

For the software industry, we can examine sustainability under two headings. The first is sustainable software, the other is sustainable software engineering.

Sustainable software is software that has a very low impact on the economy, social life and the environment or positively affects sustainable development (Dick et al, 2010). If the positive and negative effects that may arise when using the software can be known, a complete eco-friendly and sustainable software product can be created.

The second definition, called sustainable software engineering, is the expertise of creating a nature-friendly and sustainable software application with a sustainable design procedure. In other words, it is the expertise to characterize and create the software product as it should be. Thus, during

the entire product life cycle of the software product, the negative and positive effects of the practical improvements applied are constantly evaluated, reported and used to further improve the software product (Dick et al, 2010).

We can list the advantages of sustainable software and what can be done to achieve these advantages as follows:

- Performance problems occur in software applications over time. There may be several reasons for this. Some of these reasons are the increase in the number of users much faster than expected in project planning and performance problems are not taken into consideration while developing the project. One of the methods used for solving performance problems is to replace the existing hardware with higher performance hardware. However, this method is not a suitable approach in terms of sustainability. Preferring faster code instead of this method will significantly reduce hardware replacement.
- Cost analysis is carried out before starting software projects. Including environmental costs in these analyses is a recommended practice for sustainability. In this way, studies will be made on developing more effective algorithms by preventing the use of more servers.
- Memory management is extremely important in software applications. For this reason, the limited memory needs to be used optimally. With optimum memory usage, it will continue to be used in old computers with low memory. In this way, electronic waste will be reduced.
- Realization of the developments with the company's own resources will prevent the use of 3rd Party components and unnecessary resources.
- When software development engineers encounter a problem, they usually apply the first solution that comes to mind. The most important reason for this is that the time allocated for the development process is limited. Reviewing and re-coding the written code will enable more efficient and faster codes to be developed.

Sustainability should generally be considered from the first stage of software development (Dick et al, 2010). But this is not always possible. Because it is not easy to change the way software developers work. Moreover, there are very few guidelines on how software engineering can contribute to improving the sustainability of the systems under development.

In Mourão et al stated that (2018), approaches and frameworks in green and sustainable software engineering are mainly focused on the requirements and design phases. In the former, researchers support sustainability as a non-functional requirement in quality standards, which ratifies the importance of stakeholder identification, and a need to further investigate the impact of sustainability on requirements engineering. The latter mainly addresses the impacts of architecture/design on the software's power consumption. This evidence may indicate the need to consider energy efficiency as an important quality attribute when designing systems architectures.

2.2 Software Development Sustainability Dimensions

Software sustainability is characterized by five sustainability dimensions: environmental, technical, social, individual and economic (Penzenstadler and Femmer, 2013).

The environmental sustainability dimension focuses on how to create, use, protect and discard the software by minimizing its impact on the environment. The product's energy consumption can be controlled by energy efficiency, runtime efficiency, processor density, memory usage, idle and algorithm efficiency. For software engineering, the question we need to ask in terms of environmental sustainability is "How does software affect the environment during development and maintenance?" (Penzenstadler and Femmer, 2013).

Long-term use of systems with changing environmental conditions and related requirements and ensuring their adequate development have been the main goal of technical sustainability. Sustainability of technical software is related to the long use of software systems. For software engineering, the

question we need to ask in terms of technical sustainability is “How can the software be created to adapt easily to the future change?” (Penzenstadler and Femmer, 2013).

Social sustainability focuses on how to develop software that will increase social capital value. Therefore, this dimension focuses on the social added values of the software. For software engineering, we need to ask in terms of social sustainability, “What effects do software systems have on society (communication, interaction, management, etc.)?” (Penzenstadler and Femmer, 2013).

The focus of individual sustainability is on how to create and support developers to ensure long-term satisfaction from their profession. For software engineering, the question we need to ask in terms of individual sustainability is: “How can software be created and maintained in a way that ensures long-term satisfaction in the profession of developers?” (Penzenstadler and Femmer, 2013).

The focus is on the economic sustainability dimension, how software systems can be created so that the long-term investments and assets of stakeholders are as safe as possible from economic risks. Intent from assets includes not only capital but also added value. In order to be economically sustainable, the software must have a low-cost process and a long-term profit. For software engineering, the question we need to ask in terms of economic sustainability is “How can software systems be created to ensure that stakeholders’ long-term investments are as safe as possible from economic risks?” (Penzenstadler and Femmer, 2013).

3 METHODOLOGY

The process that started with the literature research continued with determination of the research subject. Questionnaire questions were prepared to measure sustainability factors after the establishment of hypotheses. Then, the data were collected, and whether the data was normally distributed was checked and reliability analysis was performed. Factors affecting software development process were determined with explanatory factor analysis and factor structure was confirmed with confirmatory factor analysis. After the measurement model compatibility test was performed, the validity of the models was tested by applying confirmatory factor analysis. After testing the hypotheses with path analysis, the results were interpreted at the last stage and suggestions for future studies were presented.

3.1 Model of the Research

The model of the research; is a study to investigate the relationship between the sustainability measurement model and the measurement model used to measure the descriptive characteristics of individuals involved in the software development process on the software performance measurement model. Along with the information about the age, experience and educational status of the people who are actively involved in the software development process, by evaluating the opinions on the concepts that may affect the sustainability in the software development process, it was wanted to be shown that how the sustainability factors affect the software development processes and performance. In addition, explanatory and confirmatory factor analyses were applied to the statements to reveal how sustainability factors affect the software development process. In this way, under which factors the effects are collected has been determined. Along with the obtained software sustainability factor, it is aimed to measure the effects on each other with the structural equation model established by using the descriptive features of the software process and the software performance measurement model.

3.2 Sample of the Research

The main population of our work, employees are actively involved in the software development process in different sectors in Turkey. Due to limited resources and time factors, data was collected through a survey conducted on the internet and social media. The sample of the study was obtained as a result of a sampling study for individuals actively working in the positions of Software Development Specialist, Business Intelligence Expert, Business Analyst and Manager in companies operating in the Finance, Marketing, Insurance, Tourism and Telecommunication sectors. The sample of the research consists of 132 people. In this research, because the time and cost factors are taken into consideration, the non-

probability sampling method was used among the people believed to be able to consciously respond to sample research questions. For this reason, the sample was created without any chance element.

3.3 Hypotheses

In his study, Capretz (2014) stated that people are more complex and less predictable than software and can create complex dynamics in the software development process. Considering that software is a product created as a result of human activities, any factor that is effective in the software development process cannot be considered without including any human factor. Amrit et al. (2014), in their research, emphasized that the process of software development is individually or as a team, so whether this process, which requires collaboration, is successful or not depends on the characteristics that define people. H1, H2 and H3 hypotheses created from these inferences are established to determine the relationship between age, education and experience expressions and software performance included in the model in order to represent the descriptive characteristics of individuals involved in the software process. Mahmood et al stated that (2018), The use of computing devices has increased dramatically in recent time, which results in huge power consumption. This situation has made the power consumption a critical metric for evaluating the performance of a computing device. Due to their internal execution characteristics, application programs exploit the hardware very differently, which leads to a quite diverse behaviour concerning their performance or the energy consumed for their execution (Rauber et al, 2018). Based on these inferences, all of the H4, H5, H6, H7 and H8 hypotheses, which demonstrate the relationship between economic, environmental, social, technical and individual dimensions, and software performance, were created to measure the impact of sustainability dimensions on software performance.

- H1: There is a significant relationship between age factor and software performance
- H2: There is a significant relationship between education factor and software performance.
- H3: There is a significant relationship between experience factor and software performance.
- H4: There is a significant relationship between the economic sustainability dimension and software performance.
- H5: There is a significant relationship between environmental sustainability dimension and software performance
- H6: There is a significant relationship between social sustainability dimension and software performance.
- H7: There is a significant relationship between technical sustainability dimension and software performance
- H8: There is a significant relationship between the individual sustainability dimension and software performance.

3.4 Explanatory Factor Analysis

The main population of our work, employees are actively involved. Before starting the factor analysis, it was checked whether the data obtained as a result of the questionnaire application was normally distributed. These tests were carried out through the SPSS program. After the analysis, it was decided by looking at the values of “Skewness” and “Kurtosis” whether the data was normally distributed or not. The data of all measurement models were subjected to the normality control separately. When performed, it was observed that Skewness and Kurtosis values took values between -2 and +2. In explanatory factor analysis, eigenvalue calculation is based on to determine the number of factors. Factors with eigenvalue statistics greater than 1 or 1 were used. Factors below this value are not taken into account. After the factor analysis was done, it was not necessary to remove these items since it was seen that the load values of the items that make up the factors have 0.50 or higher values. After the factor analysis was done, there was no need to remove any item since it was seen that the load

values of the items under the factors had a value of 0.50 or higher. Afterwards, it was checked whether there were items loaded with a value of 0.3 or more on more than one factor or not loaded on a factor of 0.5 or higher in a rotated factor matrix. As a result of these process, 5 factors and 23 items under these factors were obtained. As a result of the test, Kaiser Meyer Olkin (KMO) test measurement was calculated as 0.831. This shows that the data can be used for descriptive factor analysis. After determining the number of factors, rotation was applied to determine the statements collected under the factor. The model matrix obtained at the end of the rotation can be seen in Table 1.

Table 1. Basic factor analysis

		Factor			
		1 2			
	CS07	,871	3	4	5
	CS05	,855			
	CS04	,851			
	CS02	,801			
	CS03	,768			
	CS06	,768			
	CS01	,678			
	ES02	,902			
	ES01	,893			
	ES03	,885			
	ES04	,864			
	ES05	,844			
	SS01				
	SS02		,948		
	SS03		,915		
	SS04		,899		
	TS02		,885		
	TS04			,925	
	TS01			,913	
	TS03			,891	
	BS01			,875	
	BS02				,941
	BS03				,938

When looking at the statements collected under the factors, naming was made as environmental sustainability (1), economic sustainability (2), social sustainability (3), technical sustainability (4) and individual sustainability (5). Together with the 3 generally accepted sustainability dimensions (environmental, economic and social), the technical and individual sustainability dimensions were revealed in the study.

3.5 Confirmatory Factor Analysis

In our study, confirmatory factor analysis was performed to test the hypothesis that there is a relationship between the variables observed through the questionnaire and the factors obtained from these variables. Validity of 5 factors obtained as a result of explanatory factor analysis was investigated by confirmatory factor analysis. The path diagram obtained as a result of confirmatory factor analysis with the AMOS program can be seen in Figure 1. Values on the path diagram should be less than 1. It was also found that all the correlation values were less than 1 in the path diagram resulting from the confirmatory factor analysis. The correlation between the individual sustainability factor and the economic sustainability factor has the highest correlation value which is 0,43. The smallest value belongs to the correlation between the environmental sustainability factor and technical sustainability factor which is 0,06.

3.6 Measurement Models Tests

The first stage of testing the measurement models was started by testing the 1st stage models. In this sense, environmental sustainability dimension, economic sustainability dimension, technical sustainability dimension of software development process, individual sustainability dimension, social sustainability dimension and software performance dimension measurement models were examined. Software development sustainability factor is a two-step measurement model. For this reason, in order to measure this model, a separate analysis study was carried out and model factor loads were checked. As a result of the analyses, it was seen that there was no need to make an improvement on the model since it was seen that the values were in accordance with the criteria.

3.7 Measurement Models Confirmatory Factor Analysis

As seen in Table 2, the values obtained as a result of confirmatory factor analysis meet the criteria. For this reason, it was decided to include all models in road analysis.

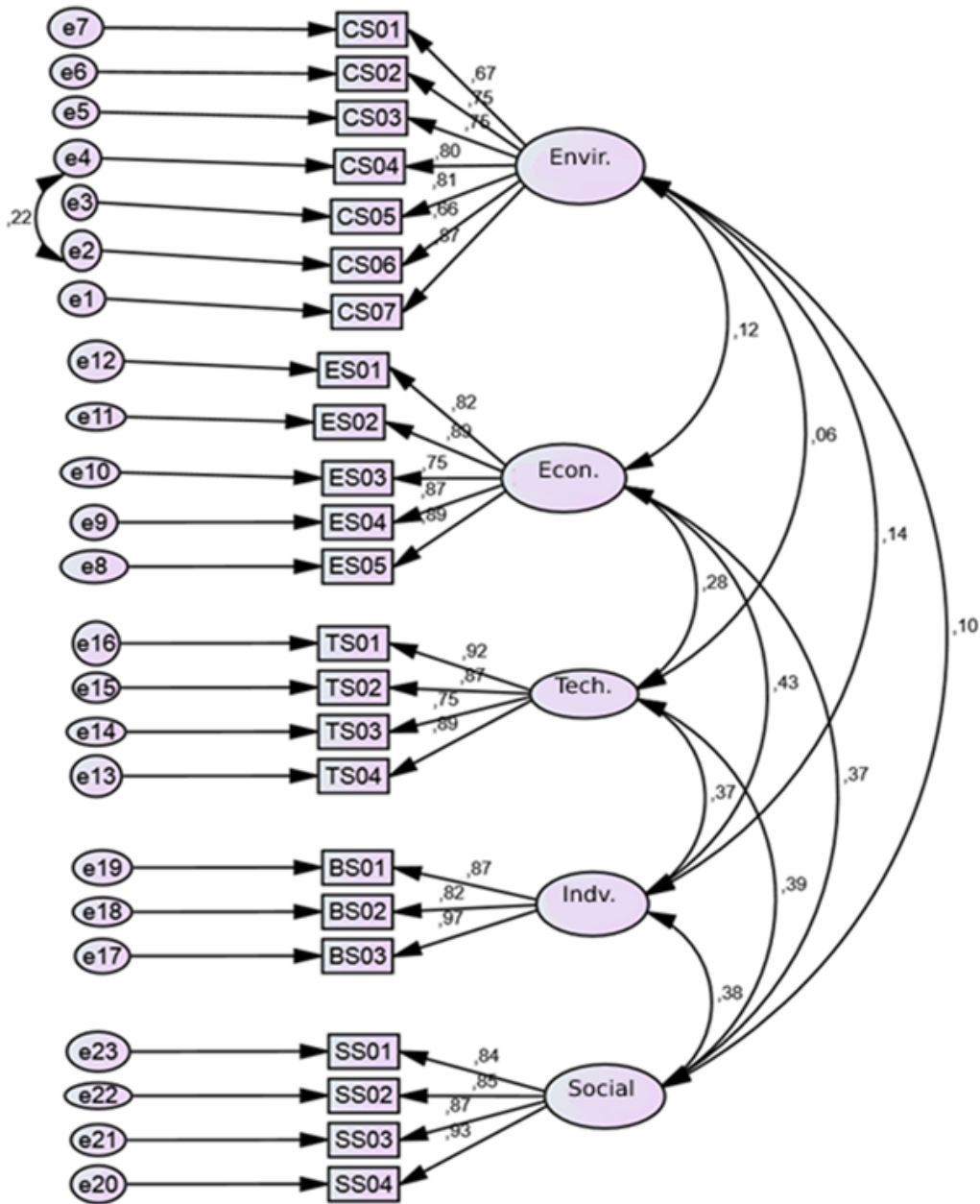
4 RESULTS

In the last stage of the research, a path analysis was conducted to test the validity of the hypotheses created as a result of the literature research. By defining the relationships between the descriptive characteristics of individuals involved in the software development process and sustainability dimensions. Along with the sustainability factor, the effect of age, education, and experience on software performance was measured. However, the relationship between age, education, and experience factors and sustainability dimensions was found to be statistically insignificant. For this reason, these relationships are removed from the model. Analysis results of the main model are shown in Table 3. The Tucker-Lewis Index (TLI) value is nearly acceptable. Only PCLOSE value is out of the desired range. As the data is valid, it was decided not to make any improvement on the model. The validity of the hypotheses was decided by looking at the p values in the model factor loads and their significance table shown in Table 4. For the hypothesis to be valid, p value must be under 0.05. All other hypotheses were found to be significant, except for the H4 hypothesis, which tested the relationship between educational status and software performance.

According to the literature study, 7 hypotheses were accepted from the 8 hypotheses except for the H4 hypothesis, which tested the effect of educational status factor on software performance.

As a result of the analysis, apart from the educational status of the individuals involved in the software process, it was observed that there was a positive effect between their age and experience and software performance. All of the H4, H5, H6, H7 and H8 hypotheses which demonstrate the relationship between economic, environmental, social, technical and individual dimensions, which affect the software development process in terms of sustainability, and software performance, were adopted. It has been observed that all dimensions affecting the software process in terms of sustainability have a negative effect on software performance. Capra et al. (2012) stated that focusing on

Figure 1 Confirmatory factor analysis path diagram model



the studies that prioritize performance in the software process negatively affect software sustainability. When it is desired to increase software performance, processor power and other resource usage in unit time increases. This will cause more energy to be used and more heat to emerge. In this study, it was seen that the dimensions affecting the software process in terms of sustainability had a negative effect on software performance. This result is Capra et al. (2012) has been a result that matches their conclusion.

Table 2. Measurement models confirmatory factor analysis results

Index	Meas.	Accomp.	Comment
CMIN	541,587	–	–
DF	344	–	–
CMIN/DF	1,574	Between 1 and 3	Excellent
CFI	0,931	>0.95	Acceptable
SRMR	0,073	<0.08	Excellent
RMSEA	0,066	<0.06	Acceptable
PClose	0,017	>0.05	Acceptable

Table 3. Main model output analysis

	CMIN/DF	TLI RMSEA	PCLOSE	SRMR	CFI
Criterion	<5	>0,90 <0,10	>0,05	<0,09	>0,80
Realized	2,217	,849 ,096	,017	,0320	,863

Table 4. Main model factor loads

		Estimate	S.E.	C.R.	P	Label
Performance	<— Economic	-,207	,024	-8,541	***	
Performance	<— Environ.	-,297	,030	-9,937	***	
Performance	<— Social	-,196	,026	-7,432	***	
Performance	<— Technical	-,213	,026	-8,159	***	
Performance	<— Individual	-,124	,019	-6,686	***	
Performance	<— Age	,104	,040	2,600	,009	
Performance	<— Education	-,054	,058	-,932	,351	
Performance	<— Experience	,173	,035	4,959	***	

This result has been in line with the literature. In the analysis made through the structural equation model, it was observed that the factors that positively affect performance negatively affect the software development sustainability. This situation shows that while focusing on the performance of the software product in the software development process, the concepts of sustainability are avoided. However, if it is desired to ensure the continuity of the marginal benefit obtained from the software industry for both the manufacturer and the consumer, care should be taken to use the limited resources in an optimum manner.

5 CONCLUSION AND RECOMMENDATION

Software performance improvements, namely increasing the maximum amount of work per unit time, reducing the amount of resources used when performing a job, and increasing the maximum user limit that it can continue to work as desired, should be provided with the efficiency of the preferred

algorithm during coding rather than extra hardware support. Adopting this awareness to employees may not only be an action that supports sustainability, but may also lead to a sustainability-enhancing result in software performance. In addition, providing an efficient working environment to employees, comparing cost-earnings before starting the software development process and determining a managerial approach may seem to negatively affect the process in terms of time, cost and performance in the short term. However, in addition to sustainability in the long term, it will contribute positively to the areas that appear negative. Bringing these concepts into organizational cultures is important for the sustainability of the software development process.

Another conclusion made on the main model is that software applications are a product that comes out at the end of a process where people work in interaction, so the factors that define people have an impact on software performance. Based on this result, it has been observed that the interaction and communication of the individuals involved in the software process with each other and their environment are healthy, the roles appropriate for the personal characteristics of the individuals in the software development process and the social structure of the company in line with the software development methodology will directly and positively affect the software performance.

Results of the study have been in line with the literature. We learnt that the dimensions affecting the software process in terms of sustainability had a negative effect on software performance from the literature. In the analysis made through the structural equation model, it was observed that the factors that positively affect performance negatively affect the software development sustainability. This situation shows that while focusing on the performance of the software product in the software development process, the concepts of sustainability are avoided. However, if it is desired to ensure the continuity of the marginal benefit obtained from the software industry for both the manufacturer and the consumer, care should be taken to use the limited resources in an optimum manner. On the other hand, we also learnt from literature that process of software development is individually or as a team, so whether this process, which requires collaboration, is successful or not depends on the characteristics that define people. Again, in the analysis made through the structural equation model, it was observed that the factors that define people have an impact on software performance positively affect the software development sustainability.

Sustainability of the software development process will not only leave a more liveable world for future generations. This also contributes to the economic structure and efficiency of companies that have a sustainable process. In an environment where competition is intense, using resources more effectively and having a sensitive profile in terms of sustainability can play an active role in the preference of companies by customers and employees.

As a result of the inferences obtained as a result of the research, suggestions for future work can be made. The questionnaire study applied in this study was prepared only on the basis of people taking an active role in the software development process. Customers who will use the software application and project owners are also included in this study, and the scope can be expanded and new researches can be made. Again, the study can be applied to people who take an active role in software development process in different regions and countries.

In this study, the effect of sustainability on software performance was examined. In future studies, factors such as security, usability, reliability and ease of maintenance may be included in the model in order to examine the effects of sustainability on other dimensions of the software.

FUNDING AGENCY

The publisher has waived the Open Access Processing fee for this article.

REFERENCES

- Agarwal, Nath, & Chowdhury. (2012). Sustainable Approaches and Good Practices in Green Software Engineering. *International Journal of Research and Reviews in Computer Science*, 3(1), 1425-1428.
- Akinyele, D., Amole, A., Olabode, E., Olusesi, A., & Ajewole, T. (2021). Simulation and Analysis Approaches to Microgrid Systems Design: Emerging Trends and Sustainability Framework Application. *Sustainability*, 13(20), 11299. doi:10.3390/su132011299
- Almadi, S. H. S., Hooshyar, D., & Ahmad, R. B. (2021). Bad Smells of Gang of Four Design Patterns: A Decade Systematic Literature Review. *Sustainability*, 13(18), 10256. doi:10.3390/su131810256
- Amrit, C., Daneva, M., & Damian, D. (2014). Human Factors in Software Development. *Information and Software Technology*, 94(12), 1537–1542. doi:10.1016/j.infsof.2014.07.006
- Diamond, J. (2005). *Collapse: How Complex Societies Choose to Fail or Survive*. Penguin.
- Dick, M., Naumann, S., & Kuhn, N. (2010). A Model and Selected Instances of Green and Sustainable Software. In J. Berleur, M. D. Hercheui, & L. M. Hilty (Eds.), *What Kind of Information Society? Governance, Virtuality, Surveillance, Sustainability, Resilience. HCC 2010, CIP 2010. IFIP Advances in Information and Communication Technology* (Vol. 328). Springer. doi:10.1007/978-3-642-15479-9_24
- Diniz, R. F., Gomes, X. M., D'Agostini, C. E., Tiago, F., & De Rose, C. A. F. (2014). Green software development for multi-core architectures. *Computers and Communication (ISCC), 2014 IEEE Symposium on*.
- Eugenio, C., Chiara, F., & Slaughter, S. A. (2012). Is software “green”? Application development environments and energy efficiency in open source applications. *Journal Information and Software Technology*, 54(1), 60–71. doi:10.1016/j.infsof.2011.07.005
- Fernando, C. L. (2014). Bringing the Human Factor to Software Engineering. *IEEE Software*, 31(2), 104–104. doi:10.1109/MS.2014.30
- Kumar, Gupta, & Reza. (2015). Energy Efficient Software Development Life Cycle An approach towards smart computing. *2015 IEEE International Conference on Computer Graphics, Vision and Information Security (CGVIS)*.
- Mahmood, B., Ahmad, N., Saif, U. R., Adeel Anjum, M., & Islam, S. (2018). Power-efficient scheduling of parallel real-time tasks on performance asymmetric multicore processors. *Sustainable Computing Informatics & Systems*, 17, 81–89.
- Moschis, G. P., Mathur, A., & Shannon, R. (2020). Achieving Sustainable Food Consumption: Insights from the Life Course Paradigm. *Sustainability*, 12(13), 5359. doi:10.3390/su12135359
- Mourão, B. C., Karita, L., & Machado, I. C. (2018). Green and Sustainable Software Engineering - a Systematic Mapping Study. In *17th Brazilian Symposium on Software Quality (SBQS)*. ACM. doi:10.1145/3275245.3275258
- Penzenstadler, B., & Femmer, H. (2013). A generic model for sustainability with process and product-specific instances. *Proceedings of the 2013 workshop on green in/by software engineering 2013*, 3-8. doi:10.1145/2451605.2451609
- Purvis, B., Mao, Y., & Robinson, D. (2019). Three pillars of sustainability: In search of conceptual origins. *Sustainability Science*, 2019(14), 681–695. doi:10.1007/s11625-018-0627-5
- Rauber, G. R., & Stachowski, M. (2018). Performance and energy metrics for multi-threaded applications on DVFS processors. *Sustain. Comput. Inform. Syst.*, 17, 55–68. doi:10.1016/j.suscom.2017.10.015
- United Nations General Assembly. (1987). *World Population Prospects: The 2017 Revision, Key Findings and Advance Tables (Raport No.ESA/P/WP/248)*. United Nations.

Koray Çırak received the BS degree in Computer Engineering from the İstanbul University, İstanbul, Turkey, in 2012, the MS degree in Management Engineering from İstanbul Technical University, İstanbul, Turkey, in 2018, where he is currently working toward the PhD degree from İstanbul Technical University. Also, he is a Senior Software Engineer at Softtech.

Hür Bersam Sidal Bolat is a Professor in İstanbul Technical University Faculty of Management, Management Engineering Department. She has got a bachelor degree and doctorate degree from ITU Management Engineering Department and ITU Science and Technology Institute respectively. Beside many research papers; she gives various lectures in different universities in the field of operations management such as production planning, supply chain management, productivity and project management. Her main research areas are supply chain and reverse logistic network design and advanced issues in project management.