International Journal on Advanced Science Engineering Information Technology

Vol.10 (2020) No. 2
ISSN: 2088-5334

Sustainable Software Engineering:A Perspective of Individual Sustainability

Sumaira Nazir[ab], Nargis Fatima[ab], Suriayati Chuprat[a], Haslina Sarkan[a], Nurulhuda F[a], Nilam N.A. Sjarif [a]

[a] *Universiti Teknologi Malaysia, Jalan Sultan Yahya Petra, Kualala Lumpur, 54100, Malaysia*

E-mail: nsumaira@graduate.utm.my; fatimanargis@graduate.utm.my; suriayati.kl@utm.my; haslinams@utm.my; huda@utm.my: nilamnur@utm.my

[b] *National University of Modern Languages, H-9, Islamabad, 44000, Pakistan*
E-mail: sunazir@numl.edu.pk; nfatima@numl.edu.pk

***Abstract*— Sustainable software engineering is a mean of developing sustainable software with sustainable software engineering process activities while balancing its various dimensions for instance economic, environmental, social, technical and individual. It is conveyed that the economic, technical, environmental and social dimensions are explored to satisfactory degree however the individual dimension of sustainable software engineering which is concerned with wellbeing of software engineers is not explored to satisfactory degree with respect to its understanding and challenges. Therefore, the aim of the study is to highlight and prioritize the challenges regarding individual sustainability dimension. The study also provides the mitigation strategies for the top five individual sustainability challenges. The systematic literature review has been performed to report the challenges and mitigation strategies. The study finding shows that lack of domain knowledge, lack of methodologies and tool support, lack of education, varying and unidentified situations and lack of sustainable software engineering practices are top most challenges regarding individual sustainability. These challenges need an urgent attention to achieve the goal of sustainable software engineering. The study also reports various mitigation strategies to overcome the risk of identified top most individual sustainability challenges such as to introduce sustainable software engineering education and knowledge in software engineering curricula, development of knowledge sharing frameworks and awareness regarding unclear and varying situations for each software engineering activity etc. The study will be beneficial for sustainable software engineering body of knowledge, sustainable software engineering practitioners and researchers by providing classified list of individual sustainability challenges and their mitigation strategies.**

***Keywords*— challenges; individual sustainability; mitigation strategy; sustainable software engineering.**

I. INTRODUCTION

Sustainable Software engineering is the foremost concerned in present era. Multiple statements are being presented by researcher to convey the notion of the sustainability, sustainable software and sustainable software engineering process [1]–[3]. For instance sustainability refers *'to capacity to endure'* while sustainable software refers to the software that can be continuously assessable, documented and maintainable [1], [4]–[6].

Sustainable software engineering is concerned with the development of sustainable software complemented with software engineering process activities [9], [10]. It enables the software engineering team to perform software engineering activities while managing the negative and positive impacts on sustainable software engineering process activities [2]. In short, sustainable software engineering aims to provide sustainable software while considering sustainability as top most concern. It also refers to commitment to treat sustainability on priority basis in software engineering [7].

The sustainability concern in software engineering is still emerging [8]–[11]. It requires context and considerations of sustainable software engineering dimensions [12]. Numerous researchers have conveyed the multiple dimension context of sustainability from software engineering perspective. For instance economic, social, environmental, technical and individual [2], [13]–[15] . These dimensions must be considered and balanced during sustainable software engineering [2], [13]–[15]. The dimensions are discussed in the following subsections.

A. Technical Sustainability

Technical sustainability is connected to software maintenance, flexibility, evolution, and the ease of transitions [7]. It is concerned with the development of

software while managing its changing technological needs and ensuring that the software can be alive for a longer period [5], [12], [16].

B. Environmental Sustainability

Environmental sustainability ensures that there must be no negative effects on environments throughout the software engineering activities [16], [17]. The major concern is to protect natural capitals for examples energy, air etc. [1], [2], [18].

C. Economic Sustainability

Economic sustainability is concerned with stakeholders' investments for the long term and high return on investment (ROI) [1], [3]. Its foremost concern is to manage risks associated with economic aspects [2], [19], [20].

D. Social Sustainability

Social sustainability is associated with the protection of interest of social communities, group of people or organizations. Software accessibility and usability are major concerns regarding social sustainability[2], [3].

E. Individual Sustainability

Individual sustainability also referred to as personal sustainability [2]. It is concerned with the wellbeing of the software engineers by providing them with education, knowledge, methodologies and tools to sustain their expertise, competencies and skills as well as to enhance their productivity [2], [3]. The details are shown in Figure 1 below.

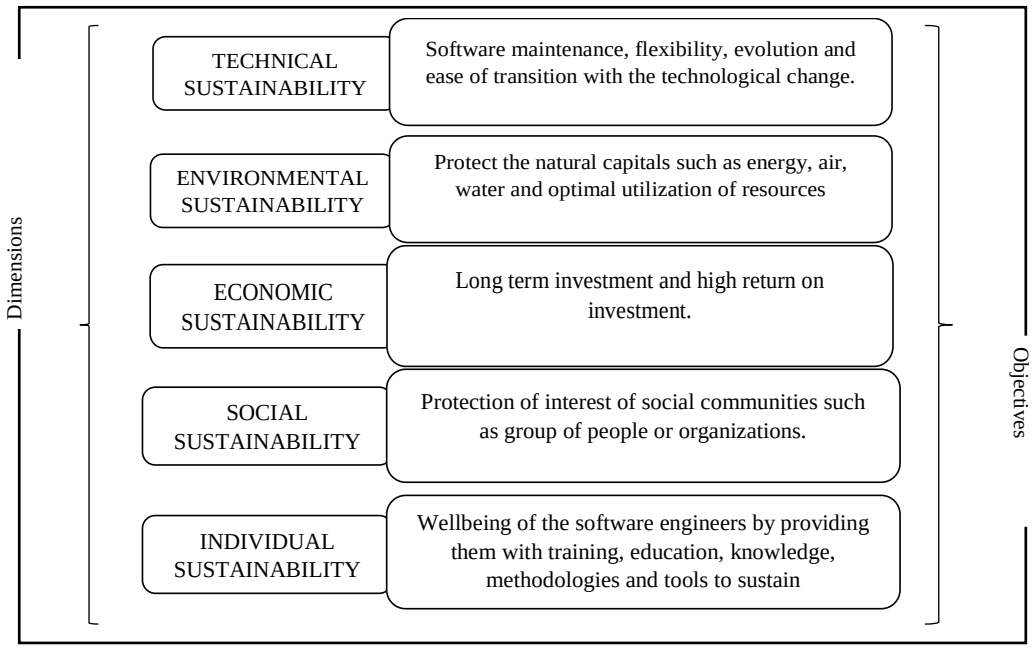

Fig. 1 Dimensions of sustainability in sustainable software engineering

Fig. 1 signifies and summarizes the sustainable software engineering dimensions along with their primary objectives. Though researchers have highlighted the importance of sustainability and sustainable software engineering and its dimensions, however, the literature reported that economic, social, environmental and technical sustainability has been explored by researchers to satisfactory degree and warrant future work on individual sustainability concerns [2], [7], [12], [19].

This paper aims to highlight the importance of individual sustainability concerning sustainable software engineering and specifically to personal sustainability. The study also reports and prioritizes the challenges associated with individual sustainability. The additional aim is to report the mitigation strategies of the top five identified individual sustainability challenges. The paper is structured as the sections I summarize the notion of sustainability and sustainable software engineering, it also highlights the sustainable software engineering facet along with its significant dimensions. Section I also provides related work regarding sustainable software engineering and individual sustainability. Section II provides the research approach and section III describes the challenges associated with the individual sustainability, prioritize the challenges based on their frequency of occurrence in existing literature and highlights top five challenges along with their mitigation strategies. Section IV concludes the paper and provides future work suggestions.

F. Literature Review

Software engineering is known to be well-defined discipline with a clear objective to develop quality software by utilizing minimum resources [21]. However, the software engineering paradigm is shifted towards sustainable software engineering so that the developed software can endure for longer periods. Many researchers have underlined the importance of sustainable software engineering and also highlighted the various challenges and provided efforts to support sustainable software engineering dimensions [1], [7], [16], [22].

Recent study performed by Salam and Khan (2018) to report the challenges related to sustainable software

engineering regarding software multi-sourcing context. The reported challenges include increase in power consumption and carbon emission throughout the software engineering activities, weak or complex software design, lack of ICTs for coordination and communication, high resource requirements, lack of coding standards and awareness etc. [22].

Another study performed by Penzenstadler et al. [3] highlighted the need of sustainable software engineering to address the global, interdependent and complex challenges faced by individuals through sustainable software development, however, according to their study, the sustainability problem associated with various dimensions need to be considered and addressed. Therefore, they provided a leverage point analysis technique to address the problems related to the sustainability dimension and thus to support the individual to deal with risk associated with challenges.

Similarly, Wu et al. [15] studied the role of informational and communication technologies for sustainable development goals. They have performed the extensive literature review and presented the open issues to enhance the social, economic and environmental dimensions for sustainable software engineering.

Likewise, Keran et al. [17] have explored the environmental dimension of sustainable software engineering. They have assessed the existing approaches defining the criteria for environmental sustainability and have introduced a model to identify the impact of software systems on natural resources.

Another study performed by Malik and Khan [16] emphasizes the need to consider economic, social, technical and environmental dimensions for sustainable software engineering. Similarly, the study conducted by khan et al. [23] conveyed the importance of sustainable software engineering and highlights the need to consider situational factor identification for each software engineering activity, for instance, requirement engineering, modeling, construction, code reviews, and deployment concerning various dimensions of sustainable software engineering. Another study presented by Khan et al. [24] explored the sustainability concern in software engineering and conveyed the need to analyze the situational factors to assess software sustainability.

An additional study performed by Chitchyan et al. [2].comprising of 11 case studies analysis explored the context of sustainability in software engineering and reported that the technical, social and economic sustainability are the primary concern, however individual sustainability is lesser prominent compared to other sustainability dimensions. The study also aims to understand the challenges regarding individual sustainability. The further study performed by Gibson et al. [19] with novice developers is subject to analyze their understanding regarding sustainable software engineering. The study aims to highpoint the challenges regarding individual sustainability such as lack of knowledge and lack of education etc. The study also emphasized the need to provide education and knowledge regarding sustainable software engineering along with its dimensions to the software engineering students at the academic level. Similarly the study conducted by Kasurinen et al. with the professional game developers to investigate the understanding of sustainability and its dimensions [25]. The study finding shows that the game developing industry does not focus on the sustainability dimensions and there is a need to improve their understanding of how to maintain the balance between various sustainability dimensions during software engineering. Likewise, the study performed by Groher and Weinreich [26] shows that software engineering practitioners strongly focused on the technical and economic sustainability dimensions while developing software products.

Additional study performed by Chitchyan et al. [7] aims explore the context of sustainability in software requirement engineering activity of software engineering. The study reported that lack of higher management support, resistance to change and lack of responsibility etc. are the hindrances to incorporate sustainability in software engineering.

Becker et al. [12] conveyed that sustainability is a significant challenge and software engineers are responsible for the long term consequences of the developed software. According to the study, requirements are key influencing points for practitioners who want to develop sustainable software. The study emphasized the need for improved curriculum integrated with sustainability principles which ensure the software engineers competencies needed to engineer sustainable software. Moreover, another study by Hinai and Chitchyan [27] shows their effort to relate the sustainability dimension with the domain-specific sustainability requirements.

The study conducted by Lago et al. [14] regarding sustainable software engineering, introduced a framework to tradeoff between sustainability quality requirements concerning various sustainability dimensions. The framework can assist software engineering practitioners to balance quality requirements between social, economic, technical and environmental dimensions. Becker et al. [4] highlight the issue of lack of a common understanding of sustainability and sustainability dimensions regarding sustainable software engineering. The study conveyed a common understanding of software sustainability, sustainable software engineering and its dimension to assist software engineering practitioners while addressing the challenges regarding sustainable software engineering.

Chitchyan et al. [28] emphasized the need to put effort into sustainable software engineering dimensions in education. The study highlights the need to educate software engineers towards the individual and social sustainability dimensions through a practical example. Furthermore, the study conducted by Chitchyan et al. [29] exemplifies the sustainability dimensions to the software product and software process. Nauman et al. [1] have explored the concept of sustainability and sustainable software engineering in detail. Besides, the study has presented criteria for sustainable software and proposed a GreenSoft Model for sustainable software engineering.

Several researchers have also highlighted the need to devise techniques and practices to support sustainable software engineering and also recommended various practices which can be used during software engineering life cycle to develop sustainable software [1], [12], [22]–[24],

[30] Numerous researchers have shown their effort to incorporate the sustainability into software engineering curricula and highlight the competencies of software engineers mandatory for successful sustainable software engineering [3], [19], [28], [31].

The literature shows that the appreciated work has been performed to support the sustainability concern in software engineering [1]–[3], [30], however, the focus towards individual sustainability is not discussed and explored to a satisfactory degree [2]. There is a gap in the literature concerning the identification of challenges and problems associated with individual sustainability [7]. The lack of such research must be addressed to sustain the software engineers and to enable them to deliver sustainable software [2], [7]. Lack of research in individual sustainability motivated us to fill the identified research gap therefore the study aims to identify the challenges associated with individual sustainability. The study also aims to identify the strategies to address the reported challenges concerning individual sustainability to support a software engineers to sustain for longer period.

II. Material and Method

The Systematic Literature Review (SLR) has been conducted to review the challenges associated with individual sustainability along with their possible solutions. SLR involves the detailed review and the critical analysis of existing literature regarding individual sustainability to support sustainable software engineering. SLR has been selected as a research methodology as it provides a rational, thorough and comprehensive search process [32].

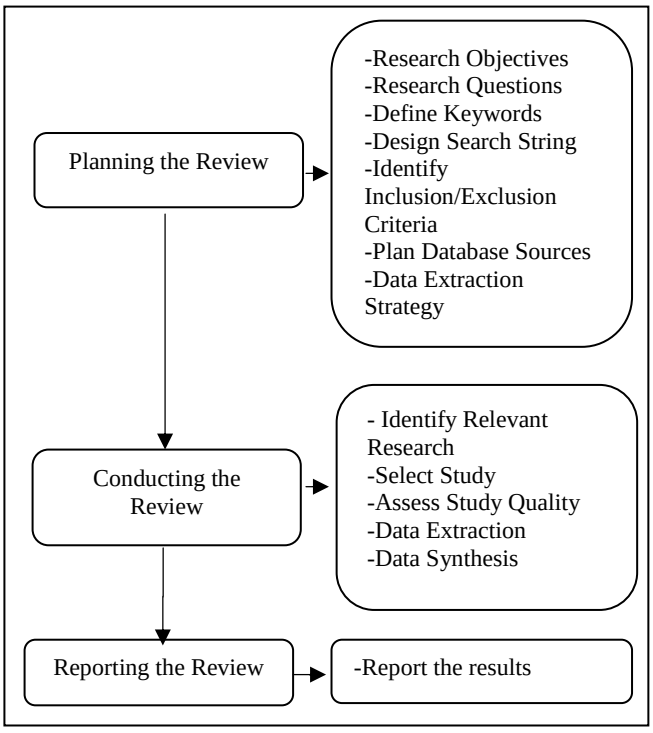

Fig. 2 Steps involved in the systematic literature review

Fig. 2 shows the steps involved in identifying and reporting the challenges and possible solutions to address the challenges. The involved steps are planning of review, conducting of review and reporting the results. The steps are discussed in the subsections.

A. Planning the Review

In this step the study objectives are planned, then the research questions are designed to achieve the study objectives. The keywords are also identified in this step. Further in this step, the search string is defined based on the identified keywords. The inclusion and exclusion criteria are also planned in this step.

1) Research Questions and Objectives: The following research questions have been designed to identify the challenges and their possible solutions regarding individual sustainability to support sustainable software engineering.

- RQ1. What are the challenges associated with individual sustainability?
- RQ2. What is the top five challenges of individual sustainability?
- RQ3. How can the challenges identified in RQ1 be addressed?

Table I shows the objectives and outcomes associated with the research questions.

TABLE I
Research Questions, Objectives and Outcomes

Research Questions	Objective	Outcome
RQ1	To identify the challenges related to individual sustainability	List of challenges related to individual sustainability
RQ2	To identify the top five challenges of individual sustainability based on their frequency of occurrence in the existing studies	List of top five challenges.
RQ3	To identify the possible mitigation strategies to address the challenges identified in RQ2.	Mitigation Strategies to address the challenges identified in RQ2

Fig. 3 shows the relationship between the designed research questions.

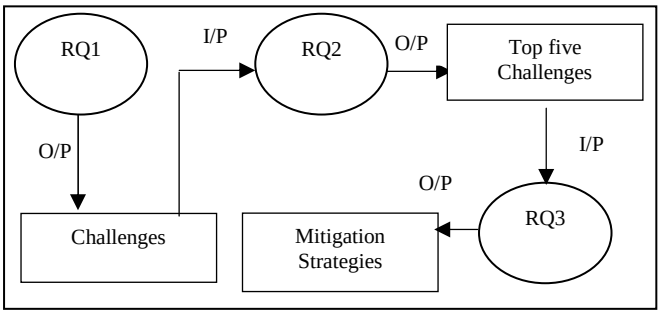

Fig. 3 The relation between research questions concerning inputs and outputs

2) Keywords and Search String: The Search string and keywords for the study are shown in Table II.

TABLE II
KEYWORDS, ALTERNATIVES AND SEARCH STRING

Keywords	Challenge	Individual Sustainability	Mitigation Strategy	Sustainable Software Engineering
Alternate words	The issue, Problem, Barrier, Risk	Personal Sustainability, Human Sustainability, Software Engineer Sustainability, Software Developer Sustainability	Solution, Resolution	Sustainable Software Development, Green Software Engineering, Green Software Development, Green Software Development Life Cycle, Sustainable Software Development Life Cycle
Search String	(Challenge OR Issue OR Problem OR Barrier OR Risk) AND (Solution OR Mitigation Strategy OR Resolution) AND (Individual Sustainability OR Personal Sustainability OR Human Sustainability OR Software Engineer Sustainability OR Software Developer Sustainability) AND (Sustainable Software Engineering, Sustainable Software Development OR Green Software Engineering OR Green Software Development OR Green Software Development Life Cycle OR Sustainable Software Development Life Cycle).			

3) Database Sources: Papers and articles published in journals, *workshops*, conferences, books, reports, and thesis are explored. Database sources such as IEEE, ACM, Scopus, Springer, Science Direct and Wiley are considered for the search process.

4) Inclusion and Exclusion Criteria: The studies are included and excluded depending upon the inclusion and exclusion criteria. Different stages of filtration are performed to include the research paper. In stage I the emphasis is on main terms. The papers describing the sustainable software engineering, individual sustainability and challenges associated with individual sustainability are selected. In stage II filtration, the papers selected in stage I are further explored for their references to include more relevant studies. The same inclusion strategy has been followed for the identification of possible mitigation strategies of the top five identified challenges. The papers are rejected depending upon the three-stage filtration process. In stage I the articles which do not contain any relevant content such as articles representing the only table of content, conferences and workshop information are rejected. In stage II the articles excluded based upon the irrelevance of title, keyword and abstract. In stage III the identical papers are rejected. The same exclusion strategy has been followed for the identification of possible mitigation strategies of the top five identified challenges.

5) Quality Assessment Plan: The checklist presented [32] is used to evaluate the quality of the study. The checklist is shown in Table III.

TABLE III
QUALITY ASSESSMENT CHECKLIST [32]

Questions	Answer
Are the goals visibly detailed?	Yes/No/Partially
Are the findings dependable and noteworthy?	Yes/No/Partially
Does the study provide comprehensive knowledge by the research?	Yes/No/Partially
Is the collection of views and contextual concerns been explored?	Yes/No/Partially
Are the relations between data, interpretation, and deductions are clear?	Yes/No/Partially

6) Data Extraction: The data is extracted from the selected studies *with* the help of extraction forms given by [32]. Table IV exemplifies the extraction form to manage the data for this study.

TABLE IV
DATA EXTRACTION FORM [32]

Data	Data to Extract	Comments
ID	Identity of Paper (Pi…Pn) I=1, n=any number	
Title		
Name of Author(s)		
Publication Date (Year)		
Publication Type	Journal/Conference/Report/Workshop	
Publisher		
Type of study	Experiment/Case Study etc.	
Selection status	Inclusion/exclusion	Reason

B. Conducting the Review

In this step of the research, all the steps defined for the review planning are executed. The relevant studies are identified using search string and finally selected for the study after filtration by inclusion and exclusion process as well as by quality assessment.

C. Reporting the Review

In this step, the results of the research study are reported. Various individual sustainability challenges are identified. Top five individual sustainability challenges are also reported based on their frequency of occurrences in the existing study along with their possible mitigation strategies in section IV.

III. RESULTS AND DISCUSSIONS

A. Individual Sustainability Challenges

This section reports the results of the SLR. Twenty-two challenges of individual sustainability have been identified from the existing literature. Table V represents the identified individual sustainability challenges.

TABLE V
CHALLENGES OF INDIVIDUAL SUSTAINABILITY ALONG WITH THEIR FREQUENCY OF OCCURRENCE

ID	Challenge	Reference	Frequency
C1	Lack of domain knowledge	[1], [2], [7], [33], [34], [35]	6
C 2	Lack of methodologies and tool support	[1], [13] [7], [33], [36], [37]	6
C 3	Lack of education	[13], [7], [33]	3
C4	Lack of experience	[7], [33]	2
C5	Poor adaptation of sustainability practice	[7]	1
C6	The resistance of software engineer to changing situations	[7]	1
C7	Varying and unidentified situations	[2], [7], [22] , [23], [33], [35]	6
C8	Lack of higher management support	[7], [22]	2
C9	Poor communication of sustainability values	[7]	1
C10	Unavailability of resources	[7]	1
C11	Time pressure	[2], [7]	2
C12	Lack of software engineer ethical and responsible behavior	[7], [22]	2
C13	Lack of sustainable software engineering practices	[7], [22], [30]	3
C14	Lack of satisfaction of software engineer with their job, facilities, and work environment	[8]	1
C15	Variation in the productivity of individual software engineers	[38]	1
C16	Lack of sustainable software engineering guidelines	[33]	2
C17	Lack of competencies, skills, efficiency, and productivity of software engineers	[22], [33]	2
C18	Misalignment and relevance of sustainable software engineering practices to the software engineering process	[19]	1
C19	Lack of understanding of the difference between individual and social sustainability dimensions	[19]	1
C20	Lack of awareness of sustainable software engineering practices	[31]	1
C21	Lack of sustainable software engineering degree programs	[31]	1
C22	Lack of information and communication technologies	[22]	1

B. *Top Five Individual Sustainability Challenges and their Mitigation Strategies*

This section highlights and discusses the top five individual sustainability challenges based on their frequency of occurrences in the existing literature. The possible mitigation strategies from the literature are also identified for the top five identified individual sustainability challenges. Table VI summarizes the top five challenges of individual sustainability.

TABLE VI
TOP FIVE CHALLENGES OF INDIVIDUAL SUSTAINABILITY

S#	Challenge ID	Challenge	Reference
1	C1	Lack of domain knowledge	[1], [2], [7], [33], [34], [35]
2	C2	Lack of methodologies and tool support	[1], [13] [7], [33], [36], [37]
3	C7	Varying and unidentified situations	[2], [7], [22] , [23], [33], [35]
4	C3	Lack of education	[13], [7], [33]
5	C13	Lack of sustainable software engineering practices	[7], [22], [30]

1) Lack of Domain Knowledge: Domain knowledge refers to the knowledge related to the specific field. It provides relevant context to enhance effectiveness and enables a software engineer to understand the problem quickly. Furthermore, it is established that domain knowledge is fundamental to effective sustainable software engineering [39]. Lack of domain knowledge is reported as a threat to individual sustainability [1], [2], [7], [33], [34], [35].

2) Lack of Methodologies and Tool Support: Lack of *availability* of methodologies and tools is also reported as a critical challenge regarding individual sustainability [1], [13] [7], [33], [36], [37]. In an interview study conducted by Chitchyan et al. one of the respondents stated that the lack of methodologies and tools are significant obstacles in achieving sustainable software engineering objectives [7].

3) Varying and unidentified situations: Unknown situations *are* changed circumstances that are resulted due to varying situational factors. These situational factors are leading causes of unknown situations among the software engineering team [23]. Literature reported that fear of unknown situations is a challenge for sustainable software engineering, specifically individual sustainability [2], [7], [22], [23], [33], [35].

4) Lack of Education: It is reported that lack of education and *experience* is another challenge that hinders the sustainable software engineering thus affecting individual sustainability [13], [7], [33].

5) Lack of sustainable software engineering practices: Lack of sustainable software engineering practices is another challenge that hinders software engineers from sustaining and from maintaining themselves for the more extended period [7], [22], [30].

Table VII summaries the mitigation strategies based on existing literature for the top five identified challenges.

TABLE VII
MITIGATION STRATEGIES OF TOP FIVE CHALLENGES OF INDIVIDUAL SUSTAINABILITY

Challenge	Mitigation Strategies	Reference
C1	Provide knowledge sharing strategies	[1]
	Create knowledge base	[1], [7], [33]
	Acquire knowledge by hiring skillful and knowledgeable software engineers	[33]
	Avoid knowledge loss by effective knowledge sharing	[33]
	Increase organizational learning	[33], [35]
	Develop a framework for individual or group discussions in collective learning and problem solving	[33]
	Provide effective knowledge sharing platforms, for example, social media	[40]
	Top management should provide tutorial and teaching courses to software engineers	[40]
	Introduce sustainable software engineering education and knowledge in Software engineering Curricula	[13], [7], [19], [28]
C2	Provide sustainable software engineering tools and methods to software engineers.	[1], [28], [33], [37]
	Provide the sustainable software engineering methodologies at the tactical and strategic level	[33]
	Avoid tools and methods that affect software engineers in their work negatively	[38]
	Demonstrate the current tools and methodologies applicability to software engineers.	[7]
C7	Make software engineers familiar with the unknown and varying situations	[2], [3], [13], [7], [12], [22], [33]
	Provide software engineers with varying situation awareness for each activity of software engineerings such as RE, Modeling, Construction, Code review and deployment	[23], [24], [28], [35]
C3	Software Engineers must know how to learn and develop a high level of education	[33]
	Software engineers should be able to acquire education from the publications and other data sources	[33]
	Develop a framework for individual or group discussions in collective learning and problem solving	[33]
	Top management should provide tutorial and teaching courses to software engineers	[7], [19], [40]
	Introduce sustainable software engineering education and knowledge in Software engineering Curricula	[13], [7], [19], [28]
C13	Provide sustainable software engineering practices at the tactical and strategic level	[28], [33]
	Introduce sustainable software engineering education and knowledge in Software engineering Curricula	[13], [7], [19], [28]

C. Contribution

The study provides two-fold contribution towards software engineering primarily with respect to sustainable software engineering body of knowledge by providing the challenges regarding individual sustainability along with their possible solutions. Additionally, the study contributed towards the software engineering body of knowledge concerning situational software engineering by highlighting the need of situational software engineering to address the challenge of varying and unidentified situations. The study also helps the researchers who further wish to investigate in the area of individual sustainability and situational software engineering.

IV. CONCLUSIONS

The study presented various challenges regarding individual sustainability. The top five challenges reported in this study include lack of domain knowledge, lack of methodologies and tool support, lack of education, varying and unidentified situations, lack of sustainable software engineering practices which can create obstacles for a software engineer to sustain for longer period of time in the software industry and thus for sustainable software engineering. The study also highlighted the mitigation strategies of the top five identified challenges of individual sustainability. For instance, to introduce sustainable software engineering education and knowledge in software engineering curricula, development of knowledge sharing frameworks and awareness regarding unclear and varying situations for each software engineering activity etc. In future the study will be further enhanced by identifying the degree of challenges associated with identified top five challenges to find better mitigation strategies. The study will also explore the consequences, causes and will suggest the mitigation strategies for all reported challenges so that the software engineering practitioners can be benefited by this research to overcome individual sustainability challenges.

ACKNOWLEDGMENT

Universiti Teknologi Malaysia partially funded the research under Research University Grant (14H76).

REFERENCES

[1] S. Naumann, E. Kern, M. Dick, and T. Johann, "Sustainable Software Engineering: Process and Quality Models, Life Cycle, and Social Aspects," *ICT Innov. Sustain. Adv. Intell. Syst. Comput.*, vol. 310, pp. 191–205, 2015.

[2] R. Chitchyan, I. Groher, and J. Noppen, "Uncovering sustainability concerns in software product lines," *J. Softw. Evol. Process*, vol. 29, no. 2, pp. 1–20, 2017.

[3] B. Penzenstadler *et al.*, "Software Engineering for Sustainability: Find the Leverage Points!," *IEEE Softw.*, vol. 35, no. 4, pp. 22–33, 2018.

[4] C. Becker *et al.*, "Sustainability Design and Software: The Karlskrona Manifesto," *2015 IEEE/ACM 37th IEEE Int. Conf. Softw. Eng. Sustain.*, vol. 2, pp. 467–476, 2015.

[5] H. Koziolek, "Sustainability Evaluation of Software Architectures: A Systematic Review," in *QoSA+ISARCS'11*, 2011, pp. 3–12.

[6] C. C. Venters *et al.*, "Software sustainability: Research and practice from a software architecture viewpoint," *J. Syst. Softw.*, vol. 138, pp. 174–188, 2018.

[7] R. Chitchyan, L. Duboc, C. Becker, S. Betz, B. Penzenstadler, and C. C. Venters, "Sustainability Design in Requirements Engineering: State of Practice," in *2016 IEEE/ACM 38th IEEE International Conference on Software Engineering Companion Sustain*, 2016, pp. 533–542.

[8] C. Calero and M. Piattini, *Introduction to Green in Software Engineering*. 2015.

[9] S. A. Koçak, G. I. Alptekin, and A. B. Bener, “Integrating Environmental Sustainability in Software Product Quality,” *Proc. Fourth Int. Work. Requir. Eng. Sustain. Syst. co-located with 23rd IEEE Int. Requir. Eng. Conf. (RE 2015)*, pp. 17–24, 2015.

[10] N. Deutsch and L. Berényi, “Personal approach to sustainability of future decision makers: a Hungarian case,” *Environ. Dev. Sustain.*, vol. 20, no. 1, pp. 271–303, 2018.

[11] S. Mann, O. Bates, and R. Maher, “Shifting the maturity needle of ICT for Sustainability,” in *5th International Conference on Information and Communication Technology for Sustainability*, 2018, vol. 52, no. March, pp. 209–190.

[12] C. Becker *et al.*, “Requirements: The key to sustainability,” *IEEE Softw.*, vol. 33, no. 1, pp. 56–65, 2016.

[13] C. Becker *et al.*, “Sustainability Design and Software: The Karlskrona Manifesto,” *Proc. - Int. Conf. Softw. Eng.*, vol. 2, pp. 467–476, 2015.

[14] P. Lago, S. A. Koçak, I. Crnkovic, and B. Penzenstadler, “Framing sustainability as a property of software quality,” *Commun. ACM*, vol. 58, no. 10, pp. 70–78, 2015.

[15] J. Wu, S. Guo, H. Huang, W. Liu, and Y. Xiang, “Information and Communications Technologies for Sustainable Development Goals: State-of-the-Art, Needs and Perspectives,” *IEEE Commun. Surv. Tutorials*, no. c, pp. 1–1, 2018.

[16] M. N. Malik and H. H. Khan, “Investigating Software Standards: A Lens of Sustainability for Software Crowdsourcing,” *IEEE Access*, pp. 5139–5150, 2018.

[17] E. Kern *et al.*, “Sustainable software products—Towards assessment criteria for resource and energy efficiency,” *Futur. Gener. Comput. Syst.*, vol. 86, no. 3715, pp. 199–210, 2018.

[18] I. Manotas *et al.*, “An empirical study of practitioners’ perspectives on green software engineering,” *2016 IEEE/ACM 38th IEEE Int. Conf. Softw. Eng.*, pp. 237–248, 2016.

[19] M. L. Gibson, C. C. Venters, M. Palacin-silva, and N. Seyff, “Mind the chasm: A UK fisheye lens view of sustainable software engineering,” 2017.

[20] R. Ahmad, F. Baharom, and A. Hussain, “Software Sustainability Development: Impactibility Characteristic Focuses on Social Approach,” *Proc. 6th Int. Conf. Comput. Informatics, ICOCI 2017*, no. 093, pp. 595–600, 2017.

[21] C. C. Venters *et al.*, “Characterising sustainability requirements: A new species red herring or just an odd fish?” *Proc. - 2017 IEEE/ACM 39th Int. Conf. Softw. Eng. Softw. Eng. Soc. Track, ICSE-SEIS 2017*, pp. 3–12, 2017.

[22] M. Salam and S. U. Khan, “Challenges in the development of green and sustainable software for software multisourcing vendors: Findings from a systematic literature review and industrial survey,” *J. Softw. Evol. Process*, vol. 30, no. 8, pp. 1–21, 2018.

[23] H. H. Khan and M. N. Malik, “Situational Requirement Engineering in Global Software Development,” *Recent Trends Inf. Commun. Technol.*, vol. 5, no. 3, 2018.

[24] H. H. Khan and M. N. Malik, “Software Standards and Software Failures: A Review with the Perspective of Varying Situational Contexts,” *IEEE Access*, vol. 5, pp. 17501–17513, 2017.

[25] J. Kasurinen, M. Palacin-Silva, and E. Vanhala, “What Concerns Game Developers? A Study on Game Development Processes, Sustainability and Metrics,” *Int. Work. Emerg. Trends Softw. Metrics, WETSoM*, pp. 15–21, 2017.

[26] I. Groher and R. Weinreich, “An interview study on sustainability concerns in software development projects,” *Proc. - 43rd Euromicro Conf. Softw. Eng. Adv. Appl. SEAA 2017*, pp. 350–358, 2017.

[27] M. Al Hinai and R. Chitchyan, “Engineering Requirements for Social Sustainability,” *Proc. ICT Sustain. 2016*, 2016.

[28] R. Chitchyan, S. Betz, L. Duboc, B. Penzenstadler, C. Ponsard, and C. C. Venters, “Evidencing sustainability design through examples,” *CEUR Workshop Proc.*, vol. 1416, pp. 45–54, 2015.

[29] R. Chitchyan, W. Cazzola, and A. Rashid, “Engineering Sustainability Through Language,” *Proc. - Int. Conf. Softw. Eng.*, vol. 2, pp. 501–504, 2015.

[30] A. D. Komeil Raisian, Jamaiah Yahaya, “Sustainable Software Development Life Cycle Process Model Based on Capability Maturity Model Integration: a Study in Malaysia,” *J. Threoretical Appl. Inf. Technol.*, vol. 95, no. 21, pp. 5723–5734, 2017.

[31] D. Torre, G. Procaccianti, D. Fucci, S. Lutovac, and G. Scanniello, “On the presence of green and sustainable software engineering in higher education curricula,” *Proc. - 2017 IEEE/ACM 1st Int. Work. Softw. Eng. Curricula Millenn. SECM 2017*, pp. 54–60, 2017.

[32] B. Kitchenham and S. Charters, “Source: " Guidelines for performing Systematic Literature Reviews in SE “, Kitchenham et al Guidelines for performing Systematic Literature Reviews in Software Engineering Source: " Guidelines for performing Systematic Literature Reviews i,” pp. 1–44, 2007.

[33] A. M. S. Al-Raqadi, A. Abdul Rahim, M. Masrom, and B. S. N. Al-Riyami, “Sustainability of knowledge and competencies management on the perceptions of improving ships’ upkeep performance,” *Int. J. Syst. Assur. Eng. Manag.*, vol. 8, pp. 230–246, 2017.

[34] M. Dick and S. Naumann, “Enhancing software engineering processes towards sustainable software product design,” *EnviroInfo 2010*, pp. 706–715, 2010.

[35] M. V. Palacin-Silva, A. Seffah, and J. Porras, “Infusing sustainability into software engineering education: Lessons learned from capstone projects,” *J. Clean. Prod.*, vol. 172, pp. 4338–4347, 2018.

[36] N. Fatima, S. Chuprat, and S. Nazir, “Challenges and Benefits of Modern Code Review-Systematic Literature Review Protocol,” *2018 Int. Conf. Smart Comput. Electron. Enterp. ICSCEE 2018*, pp. 1–5, 2018.

[37] M. Salam and S. U. Khan, “Risks Mitigation Practices for Multi-Sourcing Vendors in Green Software Development,” in *Proceedings of the Pakistan Academy of Sciences*, 2017, vol. 54, no. March, pp. 71–87.

[38] H. Hygerth, “Sustainable Software Engineering: An Investigation into the Technical Sustainability Dimension,” 2016.

[39] K. Aggarwal, T. Rutgers, F. Timbers, A. Hindle, R. Greiner, and E. Stroulia, “Detecting duplicate bug reports with software engineering domain knowledge,” in *2015 IEEE 22nd International Conference on Software Analysis, Evolution, and Reengineering, SANER 2015 - Proceedings*, 2015, no. August 2015, pp. 211–220.

[40] L. W. Chuang and S. P. Chiu, “Encouraging knowledge sharing among green fashion communities,” in *Proceedings - 2017 International Conference on Green Informatics, ICGI 2017*, 2017, pp. 141–144.