IST-Africa 2024 Conference Proceedings
Miriam Cunningham and Paul Cunningham (Eds)
IST-Africa Institute and IIMC, 2024
ISBN: 978-1-905824-72-4

Assessing Software Safety Knowledge and Skill Gaps in Tanzania

Leonard Peter BINAMUNGU[1], Salome MARO[1], Godfrey JUSTO[1], Jasson NDANGUZI[2]
[1]*Department of Computer Science and Engineering, University of Dar es Salaam, P.O. Box 33335, Dar es Salaam, Tanzania*
Email: {lepebina, maro.salome, justo}@udsm.ac.tz
[2]*Tanzania Information and Communication Technologies Commission, 14 Jamhuri Street, Dar es Salaam, Tanzania*
Email: jasson.ndanguzi2@ictc.go.tz

Abstract Safety-critical systems are systems whose failure could result in loss of life or significant damage to property or the environment. Software systems that control and support the safe operation of such systems are safety-critical software. In Tanzania, such systems are becoming common in different domains, including the health domain. The government of Tanzania is increasingly encouraging home-grown solutions over acquiring such systems from outside, implying a future where safety-critical software systems will be increasingly developed and maintained by local software professionals and graduates. However, it is unclear whether software developers in Tanzania have the requisite knowledge and skills to develop safety-critical systems. A survey of 70 industry software practitioners along with 25 follow-up interviews was conducted. A significant software safety knowledge and skills gap was uncovered, suggesting the need to carefully consider the development of software safety skills in professional programs and ICT-related curricula in higher learning institutions.

Keywords: Software safety, software safety competencies, software safety practice

1. Introduction

University graduates are expected to take up roles in industry almost immediately after graduation. However, these graduates tend to struggle in industry roles due to the gap that exists between what the universities teach and what the industry practices. While skills for well-established fields such as mechanical or civil engineering have stabilised over time, for more dynamic fields such as Software Engineering, the skills needed from graduates vary rapidly due to rapid advancements in information and communication technologies (ICT) [1]. It is therefore important for universities to analyse the knowledge and skill gaps and update the curricula accordingly.

In this paper, we address this issue by exploring the knowledge and skills gaps on developing safety-critical systems. Safety critical systems are systems whose failure could cause loss of life, damage to equipment or the environment, or significant financial losses [2]. These systems are now used in various aspects of life, including transportation and healthcare, where we increasingly rely on software-intensive systems for safety and efficiency. In Tanzania, the adoption and use of software-intensive systems is rapidly growing, and the government has recently been actively promoting the development of home-grown solutions, encouraging Tanzanian software professionals and graduates to take on the task of creating these systems. These systems are now gradually finding their way into hospitals, transportation, and even household appliances. Given this context and the fact that these systems are largely developed by graduates trained by Tanzanian higher learning institutions, there is a risk that such systems may be unsafe and yet used to support critical activities in the society. Therefore, it is crucial to examine the knowledge and skills gap to ensure the safety of these systems.

While one might assume that graduates who have the knowledge and skills to develop software systems should be equipped to develop safe safety-critical systems, specific knowledge and skills are needed for safety critical systems. For safety-critical systems, software professionals must understand concepts like functional safety, hazard analysis, and safety cases. They also need to conduct comprehensive quality assurance (QA) activities, which might not be essential for non-safety-critical systems. Various studies have mapped the skills gaps in the computing field. For instance, the study by Simmons and Simmons [3] has explored the gaps in the Computer Science curriculum and what the industry needs by interviewing IT professionals from different companies. Their study found that soft skills such as negotiation skills and management skills are missing in the curriculum. Similar studies are by Ayofe et al. [4] who explored the gaps in the Computer Science curriculum but went as far as studying the causes of and solutions to the misalignment between universities and industry. More specific studies that map the skills gap in different IT roles and professions have also been conducted. These studies are important as specific job roles in the computer field evolve. For instance, there exist studies mapping the skills gap among software engineers [5], robotics engineers [6], software testers [7], game developers [8] and data scientists [9]. With respect to software safety, to the best of our knowledge, there exists one study by McDermid [10] which examined skills needed for the development and evaluation of safety-critical systems. Some of the identified skills are the ability to analyse and choose proper technologies, knowledge of the application domain covered by safety-critical systems and the ability to make ethical decisions. While some aspects of this study are still relevant, the paper is quite old (1990) and newer and more context-specific studies on the skills gap among software safety practitioners are needed. Understanding software safety knowledge and skill gaps is therefore important to enable the software industry and higher learning institutions to prepare accordingly to close the gaps.

2. Objectives

The objective of this study was therefore to identify the knowledge and skill gaps that exist among software practitioners in Tanzania when it comes to the development of safety-critical systems. Doing so would point out areas that need to be added to computer-related curricula in institutions of higher learning to ensure that graduates can be trusted with developing safety-critical systems.

The specific objectives of the research were:

1. To identify knowledge and skills needed by software practitioners involved in the development of safety-critical systems.
2. To identify potential domains where safety critical systems are being developed in Tanzania.

3. Methodology

To establish the knowledge and skill gaps in software safety in Tanzania, we used a mixed method approach employing a survey and semi-structured interviews. First, we conducted an online survey of software professionals from the public and private sectors in Tanzania. The survey was conducted from September to October 2022, focusing on assessing the knowledge and skills of software professionals on various aspects of software safety, and whether and how they practise software safety. A convenience sampling with snowballing technique was used: the survey was sent to ICT practitioners through the email group maintained by the ICT Commission of Tanzania and through personal contacts of the authors, but these practitioners were also asked to distribute the survey to other software engineering professionals in their network. Specifically, informed by software safety knowledge and skill areas covered by the work of Kang and Do [11], we developed a questionnaire that focused on establishing the respondents' awareness of and prior training on software safety; knowledge, skills and use of best practices when developing safety-critical systems; understanding and use of software safety standards; and the domains in which safety-critical systems are developed.

A preliminary review of survey responses revealed a total of 70 respondents had answered the key questions of the survey and provided the respondents' demographics and other personal information. Second, the survey of software professionals was followed by in-depth semi-structured interviews, to ascertain what had been reported in the survey. Some of the threats of self-assessment questions in surveys are bias, lack of self-awareness and lack of honesty among the respondents. In our study, this was mitigated by the follow up semi-structured interviews. These interviews were conducted with 25 software professionals purposely sampled from five regions of Tanzania (Dar es Salaam, Dodoma, Mbeya, Mwanza and Unguja (Zanzibar)). The respondents were shortlisted based on participating organisations' regional representations and the potential that the portfolio of software development the respondents were engaged in were safety critical. In this case, the respondents from the health domain were targeted as the systems they engaged portrayed a greater potential for being safety-critical. Some interview respondents were already involved in the development of safety-critical systems (especially those from the health domain), while others were selected based on their potential to develop safety-critical systems, given the domains they worked in. Mirroring the survey questions, interview participants were asked to explain how they considered their level of knowledge/skills in general software safety-related competencies. The quantitative survey data were analysed by counting the number of responses to various software safety aspects represented by the survey questions, while qualitative content analysis was used to deduce various themes from the qualitative data collected through the survey and interviews. The analysis of survey and interviews data enabled us to establish the knowledge and skill gaps for software safety in Tanzania.

4. Findings

The results are organised into three key subsections: the respondents and work domain profile, the gaps related to software safety competencies, and the domains with safety-critical systems. The discussion of the results is also provided.

4.1 Respondents profile and work domains

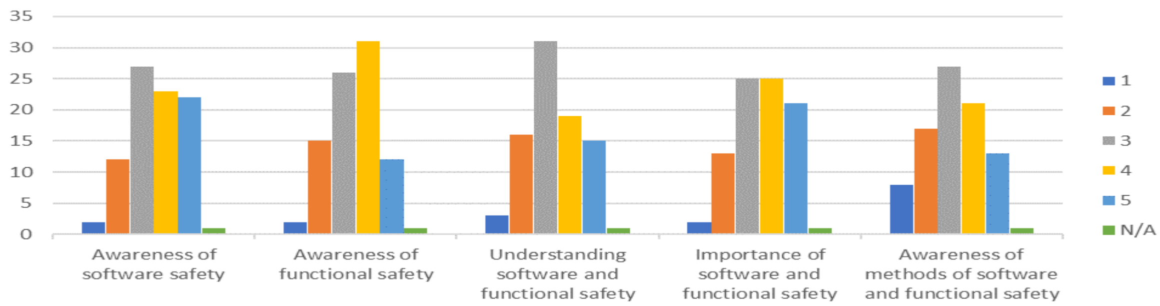
The survey attracted 70 responses from diverse software practitioners. The employing organisations were also diverse–software development organisations (27.6%), education institutions (14.9%), telecommunication organisations (12.6%), energy companies (9.2%), embedded systems companies (5.7%) and health institutions (6.9%). The respondents' characteristics are summarised in Table 1. The employing organisations were further disaggregated into public (58.6%), private (37.9%) and others–freelancer/NGO/internship (3.5%). The organisation domains were software (27.6%), education (14.9%), telecommunication (12.6%), energy (9.2%), health (6.9%), embedded systems (5.7%), and others (23.1%).

Table 1: Respondents' profiles. Note that SE companies are companies whose main business is software development while Non-SE companies are companies that have other business but also develop software.

Respondents Demographics		
1. Age Distribution ● 18-25: 9.2% ● 26-35: 52.9% ● 36-45: 26.4% ● 46-55: 10.3% ● 55+: 1.1%	2. Sex ● Male: 92% ● Female: 8%	3. Highest Education ● Certificate: - ● Diploma: 10.3% ● Bachelor Degree: 57.3% ● Masters: 27.6% ● PhD: 4.6%
4. Employed by: ● SE Specialising Enterprise: 34.5% ● Non-SE Specialising Enterprise: 41.4% ● Independent/Consulting Software Engineer:13.8% ● Unemployed: 10.3%	5. Work experience: ● Less than 1 year: 5.7% ● 1-2 years:14.9% ● 3-4 years:28.7% ● 5-10 years:31% ● 10+ years:19.5%	6. SE Area: ● Software Programming: 51.7% ● Systems Analysis: 49.4% ● System Audit: 21.8% ● Software Design/Architecting: 39.1% ● Software Project Management: 43.7% ● Hardware Programming: 14.9% ● Software Qualification: 16.1%

4.2 Software safety competencies

Figure 1 provides a general view of practitioners' competences on the five selected core software safety general competencies. In general, basic software safety knowledge and skill levels among practitioners were found to be wanting.



Where 1 = No knowledge/skill, 2 = A little knowledge/skill but considerable development required 3 = Some knowledge/skill but development required, 4 = Good level of knowledge/skill displayed, with little development required, 5 = Fully knowledgeable/skilled, no/very little development required, NA =This competence is not applicable to my job.

Figure 1: General software safety-related competencies

Figure 2 provides a general view of the practitioners' competencies on the five selected software safety competencies related to safety analysis and design techniques. Based on Figure 2, about 57%-66% of respondents ranked 3-5 on the Likert scale for competencies across the five software safety competency areas, which implied significant knowledge and skills exist among practitioners on software safety common analysis and design techniques.

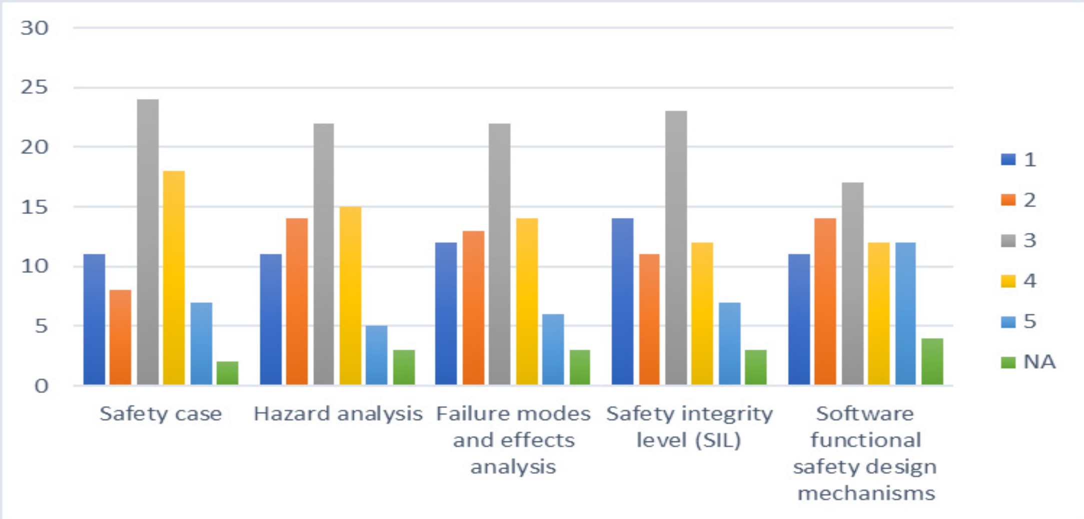


Figure 2: Software safety analysis and design-related competencies

Various safety standards have been defined to guide the development of safety-critical systems. They essentially serve as best guides but also as fundamental technical references. The IEC 61508 is a basic functional safety standard applicable to all industries, published by the International Electrotechnical Commission. The ISO 26262 is an international functional safety standard for the development of electrical and electronic systems in road vehicles. The IEC 62278 standard guides the development of electrical and electronic applications for the railway industry, based on the European Norm EN 50126. Similarly, the IEC 62279, based on EN 50128, guides the development of software for railway control and protection systems. The IEC 62304 safety standard defines the life cycle requirements for medical device software. The DO-178C guides software considerations in aviation systems and equipment certification. The MIL-STD 882E is the Department of Defence standard practice for the design and development of hardware and software, and ISO 13482 is the safety standard for personal care robots. Figure 3 provides a general view of respondents' awareness of the seven selected generic and domain-specific software safety standards. About 27.6%-43.7% of respondents ranked 3-5 on the Likert scale for awareness across the seven sampled standards, which implied relatively low knowledge/skill existed among practitioners on software safety standards, an essential component that guides the development of safety-critical systems across the safety life cycle.

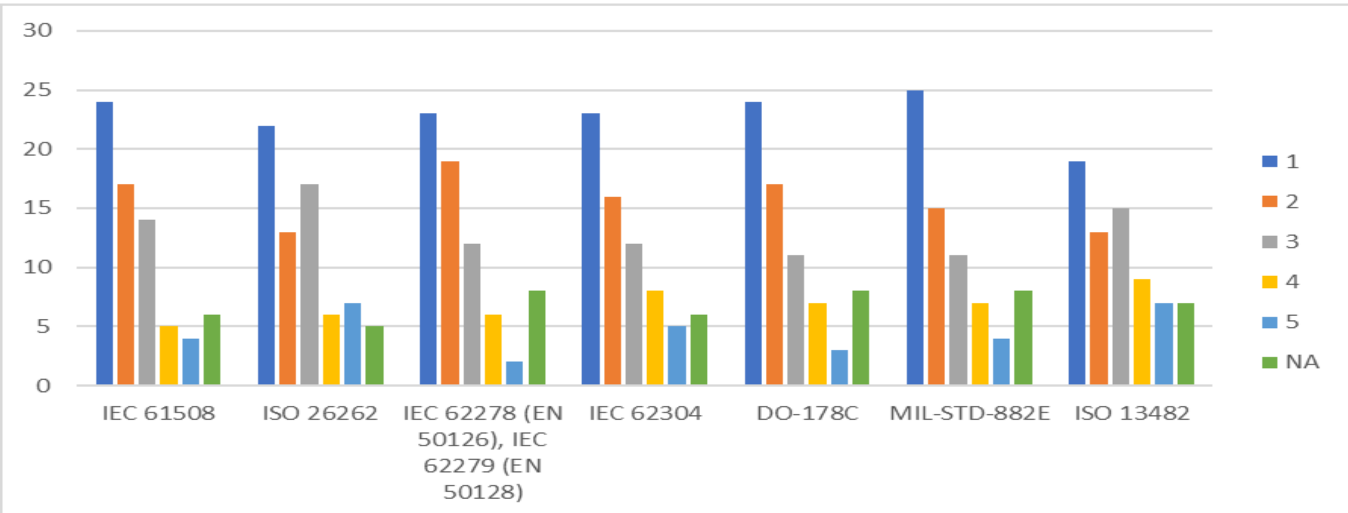


Figure 3: Awareness of safety standards

Importantly, although the survey results showed some degree of software safety knowledge and skills among respondents, the interviews provided an absolute confirmation that 24 of the interviewed practitioners had no competence related to software safety. Throughout interviews, the interviewees responded in the context of software security, the construct that they were eloquent about. The 24 interviewees were expert software professionals with software engineering-related degrees from tertiary education institutions. One interviewee who had a master's degree in biomedical engineering and an undergraduate degree in electronics showed a mastery of most functional safety concepts and standards, but little knowledge and skills in software engineering. The rest of the interviewees thought that safety means security and that if a system is secure, it is automatically safe. Due to this lack of understanding, the interviewed practitioners did not perform any safety-related activities expected in the software development lifecycle such as hazard analysis and development of safety cases, even though some of them had developed safety-critical systems.

4.3 Domains with safety-critical systems

Based on the survey and interview results, it was learned that safety-critical systems are developed and/or used in some domains in Tanzania. Such domains include health, air transportation, energy, and water. For example, the health domain has numerous electronic medical record (EMR) systems and laboratory information systems that are used by different hospitals across the country to interact with patients in ways that could compromise the safety of patients, similar to the situation explained in the work of Thomas and Thimbleby [12]. In the domain of urban and rural water supply, safety-critical software systems are, among other things, used to treat water before it is supplied to homes and offices for consumption. In the energy domain, safety-critical systems are used in the generation, distribution and transmission of electricity. Similarly, in air transportation, safety-critical systems are used to monitor various activities, including those related to safe take-off and landing. While some of these systems are developed in-country, others are imported from abroad. Appropriate knowledge for developing safety-critical systems is required in both cases. Domestic software practitioners need appropriate knowledge and skills to develop better safety-critical systems and, for safety-critical systems that are imported from abroad, domestic software practitioners need appropriate skills to be able to inspect them properly before approving them for operation. This is why domestic software practitioners need the requisite knowledge and skills to develop safety-critical systems. However, as presented in Section 4.2, such knowledge and skills are generally missing in Tanzania.

4.4 Discussion

Although the quantitative survey results suggested that a significant number of software practitioners were equipped with software safety knowledge and skills, the qualitative results from the interviews suggested the opposite, indicating the existence of a dire need for software practitioners' development in software safety knowledge and skills. Software practitioners in Tanzania confused software safety for software security. As such, the qualitative data suggested a prompt need for professional training and updating of curriculum in higher learning computer-related disciplines to accommodate software safety knowledge and skills. The state of practice opinions also confirm that software safety is ill-thought-out in development practices in Tanzania, which seem more security-intensive. Echoing the state of practice, some interviewed software practitioners said the following:

- *"Most organisations have been focusing on security of software rather than safety it is high time to bring on board this knowledge to organisations"*
- *"I believe this is a very ignored aspect in software development"*

- *“High level institutions should consider incorporate into curriculum”*

Moreover, echoing the need for developing software safety knowledge and skills among practitioners, some interviewed practitioners said the following:

- *“By prioritising safety training to Tanzanians, we may be able to identify, estimate and by so be able to prevent hazards”*
- *“Safety should be prioritised, and this is possible if the systems development practitioners are trained well to identify safety issues and develop safety procedures to handle and improve the system safety”*
- *“There should be regular education about this issue to promote greater understanding among developers”*
- *“Tanzania is moving fast in terms of technology adaptation, software safety practices become crucial now and then. I think we as tech/software engineers need to consider this practice while developing software.”*

5. Conclusions

The increasing use of ICTs in almost all socioeconomic activities has also meant that safety-critical systems have permeated many aspects of life. However, skills to properly develop safety-critical systems are in limited supply among university graduates. We surveyed and interviewed software practitioners in Tanzania and identified an acute shortage of knowledge and skills for developing safety-critical systems. Based on these findings, the study recommends deliberate mechanisms to strengthen the knowledge and skills of software practitioners in Tanzania and specifically recommends professional courses on the development of safety-critical systems for software practitioners and updating university curricula to include software safety competencies. Further, it is vital for software companies to foster software safety culture through regular training of software engineers in the aspects of software safety. A self-learning culture amongst software engineers is paramount to complement the companies’ efforts. As a country, it calls for regulatory and standard bodies to be strengthened in the aspect of software safety, but also promote awareness to software users on safety effects that may be caused by software. Evidence from literature surveys showed little/non-existence of local documentation pertaining to incidents that are software safety related. In the future, one could assess evidence of software safety related incidents within the country. Further, one can investigate how to design and implement relevant professional courses for software engineering professionals in the developing world. Also, in the future, it is important to explore the actual local software industry state of practice in the development of safety-critical systems.

Acknowledgement

Authors would like to thank the Royal Academy of Engineering, UK, for funding this work through grant number ESMN 2123\2\132. Anonymous participants of this study are also acknowledged.

Declaration of Use of Content generated by Artificial Intelligence (AI) (including but not limited to Generative-AI) in the paper

The authors confirm that there has been no use of content generated by Artificial Intelligence (AI) (including but not limited to text, figures, images, and code) in the paper entitled "*Assessing Software Safety Knowledge and Skill Gaps in Tanzania*".

References

[1] A. Verma, K. Lamsal, and P. Verma, “An investigation of skill requirements in artificial intelligence and machine learning job advertisements,” Ind. High. Educ., vol. 36, no. 1, pp. 63–73, 2022.

[2] N. G. Leveson, Engineering a safer world: Systems thinking applied to safety. The MIT Press, 2016.

[3] C. B. Simmons and L. L. Simmons, “Gaps in the computer science curriculum: an exploratory study of industry professionals,” J. Comput. Sci. Coll., vol. 25, no. 5, pp. 60–65, 2010.

[4] A. N. Ayofe, A. R. Ajetola, and A. S. Oyewole, “Assessment of existing gap between industrial IT skill requirements and computer science curriculum in tertiary institutions,” Pac. J. Sci. Technol., vol. 10, no. 2, pp. 326–336, 2009.

[5] D. Akdur, “Analysis of software engineering skills gap in the industry,” ACM Trans. Comput. Educ., vol. 23, no. 1, pp. 1–28, 2022.

[6] N. Shmatko and G. Volkova, “Bridging the skill gap in robotics: Global and national environment,” Sage Open, vol. 10, no. 3, p. 2158244020958736, 2020.

[7] B. Hamid and N. Ikram, “Industry perceptions of the competencies needed by novice software tester,” Educ. Inf. Technol., pp. 1–32, 2023.

[8] J. Kasurinen, S. Mirzaeifar, and U. Nikula, “Computer science students making games: a study on skill gaps and requirement,” in Proceedings of the 13th Koli Calling International Conference on Computing Education Research, 2013, pp. 33–41.

[9] G. Li, C. Yuan, S. Kamarthi, M. Moghaddam, and X. Jin, “Data science skills and domain knowledge requirements in the manufacturing industry: A gap analysis,” J. Manuf. Syst., vol. 60, pp. 692–706, 2021.

[10] J. McDermid, “Skills and technologies for the development and evaluation of safety critical systems,” in Safety of Computer Control Systems 1990 (Safecomp’90), Elsevier, 1990, pp. 163–171.

[11] J.W. Kang and S.R. Do, (2021). The Needs Analysis of Software Safety Education Program for Common Competency Area. Journal of Information Processing Systems, 17(5).

[12] M. Thomas and H. Thimbleby (2018). Computer Bugs in Hospitals: An Unnoticed Killer.