Applied Information Technology And Computer Science Vol. 3 No. 2 (2022) 1644-1652
© Universiti Tun Hussein Onn Malaysia Publisher's Office

AITCS

Homepage: http://publisher.uthm.edu.my/periodicals/index.php/aitcs
e-ISSN :2773-5141

AM-CSD Standards Metrics Support Project Management To Enhance Assessment of CSS Development

Abdulaziz Ahmed Thawaba[1], **Azizul Azhar Ramli**[2], **Mohd. Farhan Md. Fudzee**[2]

[1]Faculty of Technology and Computer Science,
University of Saba Region Marib, YEMEN

[2]Faculty of Computer Science and Information Technology,
Universiti Tun Hussein Onn Malaysia, Parit Raja, 86400 Batu Pahat, Johor Darul Ta'zim, MALAYSIA

Corresponding Author Designation

DOI: https://doi.org/10.30880/aitcs.2022.03.02.000 2022.03.02.102
Received 24 August 2022; Accepted 02 November 2022; Available online 30 November 2022

Abstract : Critical Systems (CSS) are found in our everyday surroundings, such as medical, transportation, and aerospace systems. CSS failures may be a direct cause of harm to human life or environmental destruction. According to some studies, 60% of CSS failures can be reduced by optimizing the development processes. Therefore, CSS developers face difficulties in choosing appropriate metrics that help find errors, enhance implementation processes, and ensure compliance with standards. There are many metrics for project management (MPM) that improve project evaluation, but there is still a gap to reduce errors and ensure a reliable product. The Assessment Metrics for Critical System Development (AM-CSD) is an MPM tool. AM-CSD metrics improve CSS development by evaluating development processes from the initial stage to the closure stage. This paper discusses AM-CSD metrics using a case for train system project development.

Keywords: Critical Systems (CSS); Metrics of Project Management (MPM); Critical System Development (CSD); Area: Project management, Development Measurement Tools.

1. Introduction

Critical Systems (CSS) are found in many important systems in our lives, such as transportation, medical, aerospace, and military systems. CSS is like other systems prone to failure but the failure of some SCS systems may negatively affect the environment or/and lead to death or injury to humans [1]. Perhaps the reasons for failure are CSS complexity, distribution, or other reasons, which is why developers face huge challenges to build reliable CSS [2]. However, developers try to improve development processes by integrating metrics with other technologies to help software developers find

**Corresponding author: azizth@usr.ac*

and reduce errors. There are many types of measurements used to assess development, some measuring a specific part and others measuring more than one part. Measurements should be made with every part of the CSS project to avoid any glitches during development phases [3] [4].

Through this paper, we will discuss how AM-CSD metrics will help reduce errors during the stages of critical systems development (CSD). The sections of this paper are organized as follows; Section (2) reviews some of the previous work that discussed topics related to CSS and CSD. Section (3) introduces AM-CSD and explains its features. Section (4) analyzes the implementation of AM-CSD metrics using a case for a train system project. The final section contains the conclusion and proposed some future works to reduce CSS failures.

2. Critical Systems

Failures in the use of CSS will have serious consequences, human losses, and significant financial loss [5]. The functionality and complexity of CSS increase failures in various domains [6]. The most common failures in CSS are insufficient specification requirements or misunderstandings attributed to requirements engineering problems [7]. Errors in the design and implementation of CSS can lead to serious consequences during the operation phase of the system [8]. However, CSD is similar to other system development processes, but it needs a more precise and accurate structure to avoid errors that may lead to failure [9]. CSS projects are different from any other projects because they are difficult due to the sensitivity of the system and the consequences of failure. An important determinant of CSD is a clear understanding of the system's environment, complexity, capabilities, and information [10].

According to Escribano's study, 60% of failures during the SCS run-in phase can be reduced or avoided by optimizing the work in the SCS development phases [11]. There are many measurement tools and techniques used to improve development and risk management processes. Moreover, there are several metrics used to improve project management, most of which focus on measuring a specific phase or part of the project lifecycle [12]. Project management or developers apply a lot of measurement techniques to enable them to control the development process and produce a product that meets standards. Several well-known measurement approaches/techniques enable the management team to enhance development processes, the most famous of which are Agile metrics and key performance indicators (KPIs) [13]. KPIs work on project objectives and provide means of measurement that enables project management to control actual cost (AC) and planned value (PV) [14]. Kanban is one of the measurement techniques of Agile and is used to ensure specifications are analyzed and tested before they reach the increasing limits of the program. Also, the Epic Burn-up Chart from Agile Technologies serves as a portfolio metric that tracks epic achievement progress [15]. However, developers are still trying to find additional techniques and metrics to reduce CSS failures, but there is a gap in ensuring the reliability of the final CSS product, see **Figure 1**. Therefore, the focus should be on finding a more in-depth measurement and monitoring of CSS failures so that developers and the project management team can improve the reliability of the end product.

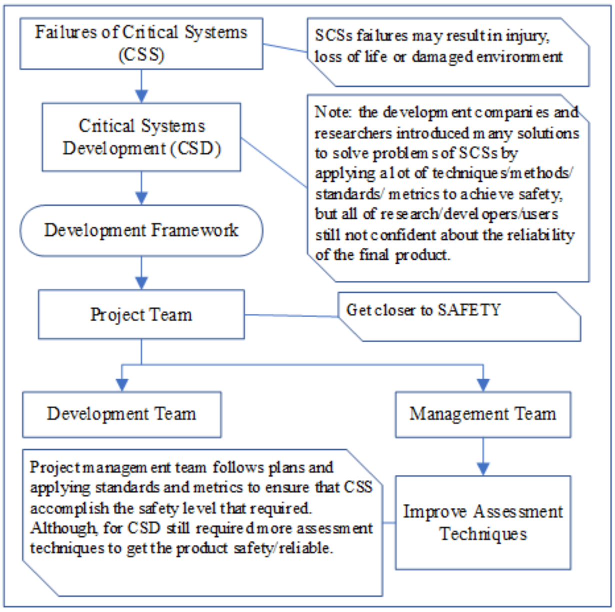


Figure 1: Gap of CS Development Processes

3. Assessment Metrics for Critical System Development (AM-CSD)

Assessment Metrics for Critical System Development (AM-CSD) is a benchmarking tool created to support developers and management teams to measure CSS development stages by monitoring achievement, forecasting, and verifying standards. Standards are the main indicator that AM-CSD uses to measure the development stages of CSS. Accurately measures the extent to which tasks are being carried out against the Standards at all stages of development and anticipates the extent to which the Standards will be achieved in the final stage. AM-CSD has two types of measurements; function metrics and standard achievement metrics. Function metrics measure the implementation of task functions through the project development phases and anticipate completion. Standard Achievement Metrics measure whether standards are met during the development stages of each task and predict how well the implementation of tasks meets the standards in the completion stage. The standard achievement metrics are Measuring Standards Achievement (MSA) and Measuring Standards According to Time (MSAT). The predicting standards metrics are Standards Achievement Prediction (SAP), Sub-Tasks Standards Achievement Prediction (TSAP), and Sub-Systems Standards Achievement Prediction (SSAP).

3.1 Measuring Standards According to Time (MSAT)

MSAT is a time-based measure of task implementation. Time usage improves measurement techniques for accurate results by ensuring that tasks are performed according to time and standards. The MSAT works by using a single formula and then applying a mechanism that takes into account the standards classification and the time to complete the tasks.

$$SATP = \frac{SA * TNS}{TE} \qquad \textbf{Eq. 1}$$

Equation **(1)** calculates Standard Achievement over a specific Time Period (SATP) by multiplying standards achievements (SA) of the task with the time needed for standard (TNS) and then divided it with the Time Elapsed (TE).

3.2 Standards Achievement Prediction (SAP)

SAP is used to predict that tasks will be accomplished against standards. SAP is considered subsets as well as sub-tasks as individual tasks. This metric applies to the last test of the task to verify whether they will be in the standard when completed. SAP is performed using three equations:

$$TTR = DT - (Td - Sd) \qquad \textbf{Eq. 2}$$

Equation **(2)** calculates the for task time remaining (TTR), by asking the DT task period of the difference between the Td test date and the date of starting the Sd task.

$$\mathrm{SsR} = \mathrm{St} - SA \qquad \textbf{Eq. 3}$$

Equation **(3)** calculates the standards remaining (SsR) for tasks by subtracting standards achieved (SA) of task standards St.

$$TNRS = RT - (RSs * TNS) \qquad \textbf{Eq. 4}$$

Equation (**4**) calculates the time needed for the remaining standards (TNRS) of task by multiplying the remaining standards (RSs) with the time needed for standard (TNS) and then subtracting it from the remaining time (RT) of the task implementation.

3.3 Sub-Tasks Standards Achievement Prediction (TSAP)

The subtask undergoes many tests during its development stages, and the PSEMT is designed to use the results of these tests to make predictions. TSAP uses the results of two or more tests to predict whether a task will meet the standards at the completion stage. TSAP forecasting uses Standards Achievement results from existing tests and also two equations as follows:

$$No.(SBt) = SBt1 + SBt2 + \cdots + SBtN \qquad \textbf{Eq. 5}$$

Equation **(5)** counts the number of tests for a subtask (SBt) by aggregating all tests of sub-task.

$$SAP(SB) = \frac{No.(CSA)}{No.(SBt)} * 100 \qquad \textbf{Eq. 6}$$

Equation (**6**) calculates the achievement of predicting standards for Sub-tasks PSA (SB) by counting the number of classification standards achieved (CSA) and divided by the No. of tests (SBt) for the sub-task and then multiplied by 100.

3.4 Sub-Systems Standards Achievement Prediction (SSAP)

SSAP uses the results of tests that check the achievement of standards during the implementation of subtasks and then predicts the achievement of standards at the completion stage of the subsystem. SSAP uses the SA results for subtasks as well as two equations as follows:

$$No.(SS) = No.(SBt1) + No.(SBt2) + \cdots + No.(SBtN) \qquad \textbf{Eq. 7}$$

Equation **(7)** counts the No. of sub-tasks for sub-system (ss) by aggregate all sub-tasks (tst).

$$SAP(SS) = \frac{SAP(SB)}{No.(SS)} * 100 \qquad \textbf{Eq. 8}$$

Equation **(8)** calculates the prediction of the PSA subsystem standard achievement ratings (SS) by dividing the standard prediction achievement of the PSA subtask (SB) with the number of subsystems (SS) count tests and multiplied by 100.

4. Performance Analysis

4.1 Discussion Case Study

The train system is a CSS system, and if any failure occurs while the system is running it can cause very serious consequences like death, injury, etc. Train safety standards have improved significantly over the past decades, but accidents are still rife. Gou studies found that the errors with the highest rate in the train system are process organization, inappropriate supervision, interpersonal competence, and skill-based errors [16]. There is a lack of government reports showing the occurrence of train accidents, but some media reports show the scale of disasters caused by railway accidents. According to the 2014 Sher studies, there were a large number of casualties and injuries due to train accidents between 2005 and 2009, these accidents were caused either by system failure or human error [17]. The project is divided into two main parts, the train crossing system, and the train safety system; each part contains subtasks.

4.2 Discussion of AM-CSD results

AM-CSD benchmarks used standards as a key indicator to measure CSS development processes. Measures achievement of task standards in the development stages and anticipates achievement of task standards at the completion stage.

4.2.1. Measuring Standards According to Time (MSAT)

MSAT assisted the project team to ensure that standards were implemented according to plan and time for each standard. The mechanism used in the MSAT to determine the rating for achieving criteria was to check whether the criteria were met and the time spent on each standard. The MSAT mechanism also defined the action required by the management team.

Table 1: Standard Classification Metrics According to Time Scheme (MSAT)

Task ID	Task Name	Start Date	Test Date	Time Spent	Time for each standard	Standards Achieved (SA)	SA Acco-rding to Time	Standard Classific-ation	Action Require
T-S13	Integration of T-S11 & T-S12(Part (1))	10/5/2018	10/8/2018	3	0.1	95	3.16666667	Very high satisfaction	DO Nothing
T-S22	Brake Control System (sub-system of Part (2))	9/5/2018	9/20/2018	15	0.15	93	0.93	Satisfaction	Improve -ment some functions
T-S24	Integration of T-S21, T-S22 & T-S23 (Part (2))	10/5/2018	10/8/2018	3	0.1	95	3.16666667	Very high satisfaction	DO Nothing

The MSAT uses this timeline to obtain more accurate results when validating criteria achievement as shown in the **Table 1**. The subtask (T-S22) met 93% of the standards and placed the task in the satisfaction category because the time required to obtain 95% or more of the standards was insufficient. Therefore, the project management must take action and may decide to improve the sub-task.

4.2.2. Standards Achievement Prediction (SAP)

SAP has given the project management team the ability to anticipate standardized outcomes at the final stage of each task. After each test for each task, SAP calculated whether there was still time to implement the remaining standards to indicate whether the task standards would eventually be met.

Table 2: Predicting Final Standards Achievement of tasks (SAP)

Task ID	Test Date	Standards Achieved (SA)	Remaining Time	Remaining Standard %	Time required for each standard	Time remaining after achieving remaining standards	Predict the achievement of standards
T-S11	9/12/2018	85	33	15	0.4	27	Standards will be achieved..
T-S21	9/12/2018	92	8	8	0.15	6.8	Standards will be achieved..
T-S22	9/20/2018	93	0	7	0.15	-1.05	Standards will not be achieved unless action is taken

As shown in **Table 2**, SAP provides a prediction of the achievement of standards for all tasks at the final stage. Developers or project management can immediately review the results of the prediction mechanism after each test. According to test results on September 20, the sub-task (T-S22) seems to require an additional 1.05 days to achieve 95% or more of the standards.

4.2.3. Sub-Tasks Standards Achievement Prediction (TSAP)

TSAP assisted the project management team to obtain more accurate results to predict sub-task completion standards. TSAP has improved the decision-making process of the Train system management team by using several test results for each sub-task during the development stages.

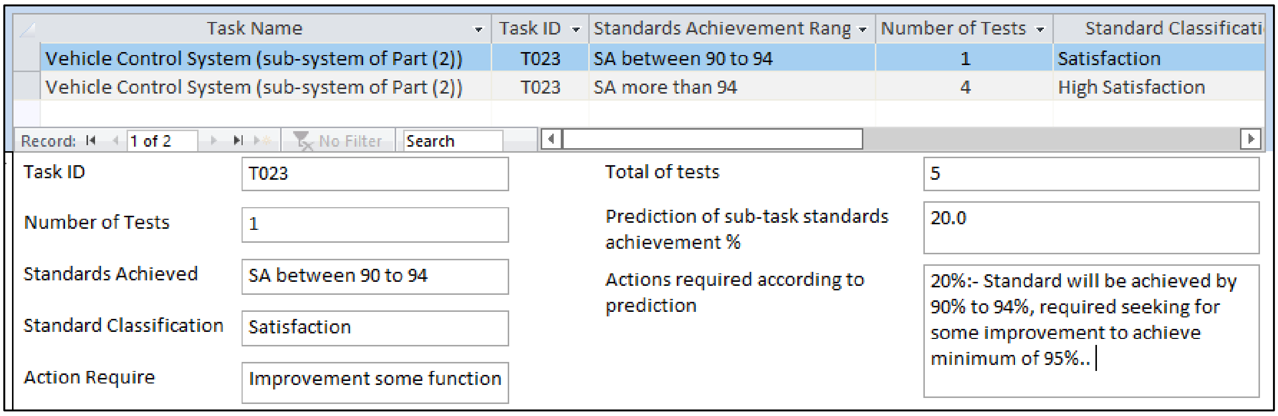
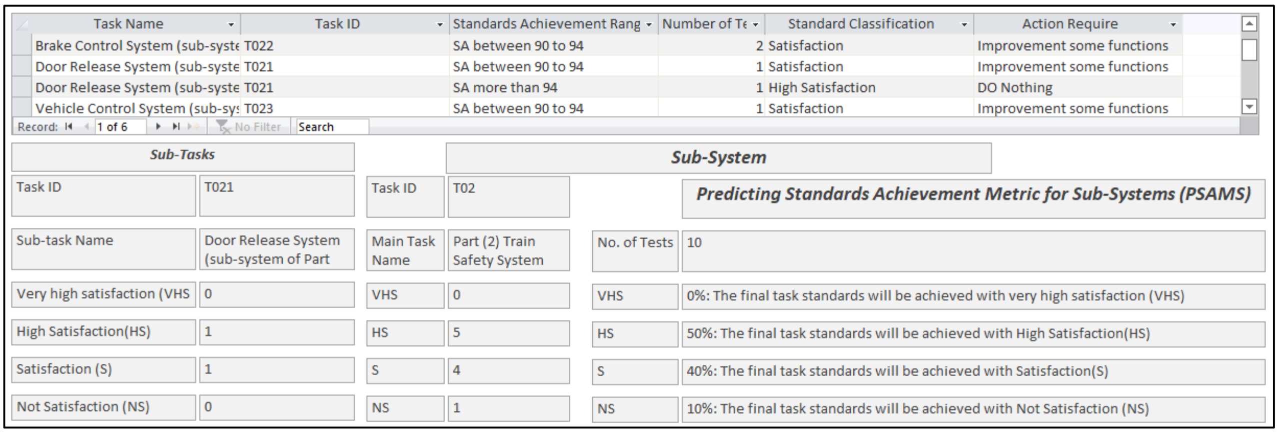


Figure 2: TSAP for sub-task (T-S23)

Figure 2 shows the achievement of classification of sub-task standards (T-S23) according to eleven tests conducted. The tests carried out on the sub-task (T-S23) are eleven tests, seven of which achieved HS, three NS and one S. The following equations used those results in **Table 1** to predict the achievement of Sub-Task standards (T-S23).

4.2.4. Sub-Systems Standards Achievement Prediction (SSAP)

PSALM facilitated decision-making by the management team and in the early stages of the project, which improves implementation and reduces errors.



Task Name	Task ID	Standards Achievement Rang	Number of Te	Standard Classification	Action Require
Brake Control System (sub-syste	T022	SA between 90 to 94	2	Satisfaction	Improvement some functions
Door Release System (sub-syste	T021	SA between 90 to 94	1	Satisfaction	Improvement some functions
Door Release System (sub-syste	T021	SA more than 94	1	High Satisfaction	DO Nothing
Vehicle Control System (sub-sys	T023	SA between 90 to 94	1	Satisfaction	Improvement some functions

Record: 1 of 6 No Filter Search

Sub-Tasks				Sub-System	
Task ID	T021	Task ID	T02		Predicting Standards Achievement Metric for Sub-Systems (PSAMS)
Sub-task Name	Door Release System (sub-system of Part	Main Task Name	Part (2) Train Safety System	No. of Tests	10
Very high satisfaction (VHS	0	VHS	0	VHS	0%: The final task standards will be achieved with very high satisfaction (VHS)
High Satisfaction(HS)	1	HS	5	HS	50%: The final task standards will be achieved with High Satisfaction(HS)
Satisfaction (S)	1	S	4	S	40%: The final task standards will be achieved with Satisfaction(S)
Not Satisfaction (NS)	0	NS	1	NS	10%: The final task standards will be achieved with Not Satisfaction (NS)

Figure 3: Standards achievement results for sub-tasks of the sub-system (T-S02)

Figure 3 shows the sub-system standards classifications (T-S02) according to the ten tests performed on its sub-tasks. The following equations use the current results of SA to predict the achievement of standards at completion stages for the sub-system (T-S02).

$$Very\ high\ stratification = \left(\frac{0}{10}\right) * 100 = 0\%$$

The result of the prediction showed that the subsystem (T-S02) in the completion stage will not be able to match the standards with Very High Satisfaction (VHS) according to SSAP. Therefore, the developers or the management team should make decisions to improve the development processes of sub-systems.

$$High\ stratification = \left(\frac{5}{10}\right) * 100 = 50\%$$

The prediction results showed that the subsystem (T-S02) will meet the standards with 50% High Satisfaction (HS) according to SSAP.

$$Stratification = \left(\frac{4}{10}\right) * 100 = 40\%$$

With 40% predicting showed that the subsystem (T-S02) will meet the standards with just Satisfaction.

$$Stratification = \left(\frac{1}{10}\right) * 100 = 10\%$$

There is a 10% of possibility that the subsystem (T-S02) will not meet the standards. Therefore, the developers or the management team should make decisions as soon as possible to improve subsystem development processes or redevelop subtasks.

4.3 Project Management Support

The PM-IST metrics are applied to training system development to support the management team with follow-up, task completion verification, achievement forecasting, and standards conformance. The following advantages illustrate the importance of using AM-CSD standards metrics for the project management team:

- Each sub-task is considered a product during the measurement.
- Each sub-task was tested several times to ensure its implementation in accordance with the plan and standards.
- Facilitate tracking of sub-task implementation by providing the percentage of standards achievement time spent from start to completion.

- Predicted the completion of the sub-task and the achievement of standards in the early stages of implementation.
- Expect the completion of the sub-system and the achievement of standards using the results from the sub-tasks.

4.4 Improve Productivity and Safety of CSS

For AM-CSD, each sub-task or subsystem is treated as a separate product, which reduces errors and improves CSD productivity. AM-CSD focuses on safety and minimization of errors during development phases by:

- Using MSAT to ensure that each sub-task or sub-system meets the standards during each test.
- SAP, TSAP, and SSAP predict that standards will be achieved in the next stages of the project.

5. Conclusion

The negative impact of critical systems failure on human life and the environment has made their development very difficult. Therefore, researchers and developers are trying to find new measurement methods and techniques that improve error detection during development processes and reduce the causes of critical systems failure. This paper introduced the PM-IST Standards Metrics to be added to the development framework of CSS. AM-CSD is a development tool that contains several metrics to assist the project management team. AM-CSD Standards Metrics was implemented in the development phases of Critical Systems. AM-CSD uses standards to measure the progress of the Train system project. Standards Metrics reduce errors that may arise during development using MSAT to track the achievement of standards for each project task. Standards Metrics has improved the project management team decision-making process through early prediction of standards for each task using SAP, TSAP, and SSAP matrices. AM-CSD metrics are required to improve their performance by adding some techniques or algorithms to improve the prediction and error detection mechanism during development processes, such as the decision tree algorithm and the random forest algorithm.

Acknowledgement

This work is sponsored by Universiti Tun Hussein Onn Malaysia under Grant FRGS No, FRGS/1/2019/ICT01/UTHM/02/2.

References

[1] P. Singh and L. K. Singh, “Engineering Education for Development of Safety-Critical Systems,” IEEE Trans. Educ., vol. 64, no. 4, pp. 398–405, 2021, doi: 10.1109/TE.2021.3062448.

[2] A. A. Thawaba, A. A. Ramli, M. F. M. Fudzee, and J. Wadata, “Characteristics for Performance Optimization of Safety-Critical System Development (SCSD),” J. Adv. Comput. Intell. Intell. Inform., vol. 24, no. 2, Art. no. 2, Mar. 2020, doi: 10.20965/jaciii.2020.p0232.

[3] P. Kumar, L. K. Singh, and C. Kumar, “Model Based Verification of Safety-Critical Systems,” in 2021 2nd International Conference for Emerging Technology (INCET), 2021, pp. 1–9. doi: 10.1109/INCET51464.2021.9456353.

[4] A. A. Thawaba, A. A. Ramli, and M. F. M. Fudzee, “PM-ISD Traceability Metrics Enhance Reliability Assessment for Safety-Critical Systems Development Processes: A Case Study of Oil and Gas Well Drilling Project.” https://ieeexplore.ieee.org/abstract/document/9590906/ (accessed Feb. 19, 2022).

[5] M. Benito, M. M. Trompouki, L. Kosmidis, J. D. Garcia, S. Carretero, and K. Wenger, “Comparison of GPU Computing Methodologies for Safety-Critical Systems: An Avionics

Case Study," in 2021 Design, Automation Test in Europe Conference Exhibition (DATE), 2021, pp. 717–718. doi: 10.23919/DATE51398.2021.9474060.

[6] D. Ratiu, A. Nordmann, P. Munk, C. Carlan, and M. Voelter, "FASTEN: An Extensible Platform to Experiment with Rigorous Modeling of Safety-Critical Systems," in Domain-Specific Languages in Practice: with JetBrains MPS, A. Bucchiarone, A. Cicchetti, F. Ciccozzi, and A. Pierantonio, Eds. Cham: Springer International Publishing, 2021, pp. 131–164. doi: 10.1007/978-3-030-73758-0_5.

[7] L. E. G. Martins and T. Gorschek, "Requirements Engineering for Safety-Critical Systems," 2021.

[8] R. K. Cosner et al., "Safety-Aware Preference-Based Learning for Safety-Critical Control," ArXiv Prepr. ArXiv211208516, 2021.

[9] C.-L. Lin and W. Shen, "Applying Safety Case Pattern to Generate Assurance Cases for Safety-Critical Systems," in 2015 IEEE 16th International Symposium on High Assurance Systems Engineering, Daytona Beach Shores, FL, USA, Jan. 2015, pp. 255–262. doi: 10.1109/HASE.2015.44.

[10] C. Baron and V. Louis, "Towards a continuous certification of safety-critical avionics software," Comput. Ind., vol. 125, p. 103382, 2021.

[11] J. Escribano-Barreno and M. García-Valls, "Supporting the monitoring of the verification process of critical systems'software," ArXiv151204782 Cs, Dec. 2015, Accessed: Feb. 06, 2019. [Online]. Available: http://arxiv.org/abs/1512.04782

[12] P. Mesároš, A. Behúnová, T. Mandičák, M. Behún, and K. Krajníková, "Impact of enterprise information systems on selected key performance indicators in construction project management: An empirical study," Wirel. Netw., vol. 27, no. 3, pp. 1641–1648, 2021.

[13] Z. Kwee, E. Yaghmaei, and S. Flipse, "Responsible research and innovation in practice an exploratory assessment of Key Performance Indicators (KPIs) in a Nanomedicine Project," J. Responsible Technol., vol. 5, p. 100008, 2021.

[14] M. Płaza and Ł. Pawlik, "Influence of the contact center systems development on key performance indicators," IEEE Access, vol. 9, pp. 44580–44591, 2021.

[15] A. S. Silveira Jr, "The Importance of Measurement and Metrics," in Building and Managing High-Performance Distributed Teams, Springer, 2021, pp. 73–83.

[16] H. Gou, C. Liu, R. Xie, Y. Bao, L. Zhao, and Q. Pu, "Running safety of high-speed train on deformed railway bridges with interlayer connection failure," Steel Compos. Struct., vol. 39, no. 3, pp. 261–274, 2021.

[17] Sher Afzal Khana, Nazir Ahmad Zafarb, Farooq Ahmadc, and Saeed Islamd, "Extending Petri net to reduce control strategies of railway interlocking system - ScienceDirect," 2014. https://www.sciencedirect.com/science/article/pii/S0307904X1300365X (accessed Oct. 18, 2018).