https://doi.org/10.31449/inf.v49i31.11012

Integrated Modeling in the Quality Assessment of Flight Management Software Systems

Olha Zinchenko, Nataliia Lashchevska*, Kamila Storchak, Anton Shantyr, Ihor Tovstochub, Oleksii Cherevyk and Andrii Balvak
State University of Information and Communication Technologies, Kyiv, 03110, Ukraine
E-mail: n.lashchevska@duikt.edu.ua
*Corresponding author

Keywords: Integrated quality assessment, software reliability, predictive analytics, adaptive algorithms

Received: August 22, 2025

Ensuring the high quality of Flight Management Systems (FMS) software is a critical task given the increasing demands for accuracy, reliability, and safety in aviation transportation. Modern FMS quality assessment methods often fail to account for the complexity of dynamic changes and interactions between system components. Objective – to develop an integrated quality assessment model for FMS that incorporates a comprehensive approach to analyzing system characteristics, including functionality, reliability, performance, security, and process transparency. Methodology – the study is based on the use of integrated modeling, which enables the combination of various evaluation criteria and their adaptation to real-world operational conditions. The proposed Integrated Quality Model (IQM) is based on a multi-criteria aggregation function that combines key quality dimensions—functionality, reliability, performance, security, and transparency – in accordance with ISO/IEC 25010 and DO-178C. The IQM framework enables quantitative benchmarking, yielding a total quality score of 0.871 for the Jeppesen Crew Management (JCM) system, which exceeds the average for comparable FMS solutions such as Sabre Airline Solutions and ARINC Direct. The assessment highlighted the system's strengths in performance, transparency, and security, while also identifying areas for improvement, particularly in user interface interactivity and adaptation mechanisms. Conclusions confirm the effectiveness of integrated modeling for FMS quality assessment. Future research may focus on enhancing adaptive algorithms and implementing predictive analytics methods to optimize flight management. Compared with traditional checklists and static MCDM scorecards, IQM explicitly models cross-dimension interactions and supports context-adaptive weighting, clarifying its novelty and practical advantage.

Povzetek: Članek predstavi integrirani model kakovosti za FMS, ki z večkriterijsko agregacijo ter kontekstno prilagodljivimi utežmi modelira interakcije med funkcionalnostjo, zanesljivostjo, zmogljivostjo, varnostjo in preglednostjo ter omogoča kvantitativno primerjavo sistemov.

1 Introduction

In today's context of intensive development of the aviation industry, the quality of aircraft flight management systems (FMS) is of particular importance [1–3]. In practice, FMS provides automation of planning, monitoring and flight management, which is critical for flight safety, economic efficiency and environmental friendliness of aviation transportation [4]. Accordingly, [1–9], with the development of information technologies, the requirements for the quality of these systems, which must meet international standards, such as ISO/IEC 25010, DO-178C, etc. are increasing. In this case, a special challenge is the need to take into account the complexity of FMS, including dynamic changes in the environment, growth of data and integration of the latest technologies, such as artificial intelligence, cloud calculations and machine learning.

In particular, given the current trend, with the growing air traffic volume and increasing complexity of aviation infrastructure, the requirements for accuracy, reliability and stability FMS are constantly increasing [7, 8], in existing approaches to the quality assessment of software used in FMS often do not take into account complexity. Notably that traditional assessment methods often do not take into account complex relationships between system components, as well as dynamic changes in operating conditions. In accordance with [1–12], the software developers now have an urgent need to develop the principles of integration modulation in modern approaches to the quality assessment of software with the use of innovative modeling methods. Where integrated modeling allows you to combine different evaluation criteria, adapt them to specific operating conditions and take into account the dynamic nature of changes in the field.

Therefore, in view of the above relevance of the study is due to the need to improve approaches to assessing the quality of FMS through the use of integrated modeling. Integrated modeling makes it possible to overcome these restrictions, allowing a holistic approach to quality analysis and predicting, which in turn helps to increase the reliability, safety and productivity of FMS. The purpose of the study is to develop an integrated model of quality assessment for FMS, based on the principles of synergistic interaction of components and the use of modern data analysis methods. Particular attention is paid to taking into account the requirements of end users, dynamic changes in the environment and the impact of different quality characteristics on the overall assessment of the system.

The proposed Integrated Quality Model (IQM) constitutes the main novelty and contribution of this work, advancing beyond existing evaluation approaches in two key ways. First, it provides a systems-level representation of how quality dimensions—functionality, reliability, performance, security, and transparency—interact with and influence one another, instead of being assessed as separate or independent criteria. Second, it incorporates an adaptive weighting mechanism that automatically adjusts the relative importance of these dimensions according to the operational context—for example, flight type, environmental conditions, or mission profile. This adaptability distinguishes IQM from traditional frameworks such as single-metric reliability evaluations, fixed multi-criteria decision-making methods (like AHP or TOPSIS), and compliance-oriented standards such as DO-178C, which rely on static checklists and do not reflect dynamic operational realities. By explicitly modeling cross-dimension relationships and enabling context-sensitive adaptation, IQM addresses major gaps in conventional aviation software quality assessments: the absence of integrated modeling across quality domains, the inability to adapt evaluation priorities to changing conditions, and the lack of traceable linkage between system-level performance and overall software quality outcomes.

2 Literature review

According to [1, 5], integrated modeling in the evaluation of software systems (SS) is an approach that involves using comprehensive models to analyze, evaluate, and optimize software quality within a specific quality management concept. This approach ensures the integration of various stages and components of the quality assessment process, considering their interactions and impact on the overall result. According to the works of [4, 9], within the concept, integrated modeling can be applied to build a unified, comprehensive approach to evaluating SS quality, enabling the consideration of all aspects of modern SS quality and their interrelationships to achieve a holistic and effective evaluation.

In analyzing unresolved scientific issues regarding the application of integrated models for evaluating the quality of aircraft FMS, it should be noted that works [1–15] emphasize the need to develop mechanisms for automating the selection of baseline values for each specific system. First and foremost, it is highlighted in the work [4] that automating the above process can significantly reduce subjectivity in selecting initial values, thereby improving the accuracy of forecasting aircraft FMS quality. The work [6] emphasizes the need for further adaptation of existing integrated modeling methods, where this need is explained by the specific characteristics of aircraft FMS. According to the methodology presented in [2, 7, 10], integrated models demonstrate great potential for improving the processes of analyzing and forecasting the quality of aircraft FMS. However, their practical application requires significant optimization, particularly in automating the selection of baseline values and solving issues related to correlated variables in complex system architectures.

Studies [4, 8] highlight the problem of correlated variables, which can cause several difficulties in the modeling process. Correlated variables can lead to biased estimates, reducing the accuracy of forecasts. For example, when the data processing speed and memory usage in aircraft FMS are highly correlated, changing one of these parameters may not impact the assessment of the other, resulting in an underestimation of their impact on the system. Correlated variables can also lead to the loss of important information if their interrelationships are not accounted for in the models, which decreases the effectiveness of the quality analysis of aircraft FMS.

To address this issue, researchers in works [1–15] recommend strategies to reduce the impact of correlated variables. For example, using dimensionality reduction methods such as Principal Component Analysis (PCA) is one such strategy that reduces the impact of correlations between variables [12]. Additionally, approaches to automating the selection of baseline values are crucial, as they help avoid subjective influences when configuring model parameters [13]. Moreover, to improve the accuracy and reliability of aircraft FMS quality assessment, it is essential to consider the interrelationships between system components, which will provide more accurate reliability and fault tolerance assessments. Existing integrated models, on the other hand, often treat each system component separately, which may be insufficient for complex, interconnected systems, as noted in works [14, 15]. According to [10], modifying integrated models to include these interconnections allows for more accurate assessment of the impact of a component failure on overall system performance.

Another important issue is the need to expand integrated model methods for comprehensive quality assessment, which involves simultaneously considering several parameters such as efficiency, scalability, and availability. The study [13] proposes a multi-evaluation method that combines multiple metrics to create a more balanced and accurate quality indicator. In this case, multi-evaluation allows for a comprehensive assessment of various aspects of aircraft FMS quality, which is especially important for distributed systems where quality evaluation must take into account numerous factors [4].Regarding the application of integrated model-based methods using the principles of predictive analysis, it is important to note that this approach enables forecasting the impact of future changes in the system's architecture or code on its performance. As stated in works [12, 14], such analysis not only allows for evaluating the current state of the aircraft FMS but also predicts possible changes in the system, which significantly enhances the efficiency of its development management.

Finally, adaptive algorithms for scalable systems are also a crucial component in improving integrated model methods. Modern aircraft FMS use micro service architectures and cloud computing, which require significant computational resources for analysis. In work [9], adaptive algorithms are proposed that optimize resources by using distributed computing and aggregated models, allowing effective application of integrated models for large, complex systems. However, the increasing complexity of software systems due to the use of micro service architectures and cloud platforms points to the need for further refinement of integrated model methods. As noted in work [5], traditional quality assessment methods often fail to account for these complexities, which can lead to an underestimation of the importance of individual components and their interactions, negatively impacting the overall performance and reliability of the aircraft FMS.

In the further development of integrated models for assessing the quality of aircraft FMS software, it is important to also focus on the evolution of these models in the context of rapidly changing requirements and new technological challenges. Therefore, the emergence of new types of security threats, such as cyber-attacks and vulnerabilities in modern cloud and distributed systems, demands the adaptation of models to account for these risks. Works [1, 4, 13] emphasize the need to integrate risk assessment models into the design and testing processes of aircraft FMS to ensure high levels of security, stability, and efficiency. According to work [12], current models for assessing the safety and reliability of aircraft FMS often focus on determining the failure probabilities of system components or subsystems, but these models do not always include the dynamic aspects of component interaction during failures or unauthorized access attempts.

In line with work [14], modern integrated software modeling approaches often involve the use of machine learning and artificial intelligence techniques to improve safety and reliability assessment models. For example, deep learning algorithms can help detect hidden patterns in system behavior that are not always detectable using traditional methods. According to [3], improving system efficiency can negatively affect its security; however, recent studies [10, 14] demonstrate the possibility of balancing these two factors, where the development of integrated models needs to consider mechanisms that allow for the simultaneous optimization of these two aspects without compromising either.

Work [12] provides a detailed review of adaptive algorithms for analyzing and monitoring the efficiency of aircraft FMS under varying parameters. The authors of this work point out that aircraft FMS operate in dynamic conditions where parameters such as system load change in real time, and there is a need for methods to adapt to these changes. Adaptive algorithms that automatically adjust to current conditions maintain the system's stability and efficiency even in the event of unpredictable changes. Studies [7, 8] emphasize the importance of using forecasting methods to prevent potential failures and ensure proper preparation for such situations. This includes integrating quality assessment models with real-time monitoring systems, allowing for the identification of potential issues before they lead to serious failures or breakdowns in the aircraft FMS operation.

Another promising area is using models to optimize the testing processes of aircraft FMS, where testing approaches require significant time and resource expenditures to cover all possible system operation scenarios. However, the use of integrated models allows for automating much of the testing process, particularly through the use of generative models to create test data [5, 11]. In practice, this approach reduces testing costs and improves its quality, as more realistic and diverse scenarios are automatically generated.

As we can see from the identified problematic areas, there is an urgent need for the continuous improvement of forecasting and model correction methods, considering the results of their use in real-world conditions. This will allow these models not only to adapt to current changes but also to help prevent future issues that may arise during the operation of modern aircraft FMS.

Compared with established software quality frameworks such as ISO/IEC 25010, McCall's quality model, and Boehm's model, the proposed IQM extends their structure by explicitly modeling cross-dimension interactions and adaptive weighting. While ISO/IEC 25010 defines static quality characteristics, IQM introduces dynamic reweighting to reflect operational context. Similarly, unlike McCall's and Boehm's models that treat attributes independently, IQM quantifies interdependencies among performance, reliability, and usability—an essential capability for flight-critical systems. These differences are summarized in Table 1. Comparative summary of existing software quality assessment approaches.

This benchmarking framework highlights explicit gaps motivating the IQM: (1) lack of dynamic adaptability in weighting and aggregation, (2) insufficient integration of ISO/IEC 25010 and DO-178C principles, and (3) limited capability to model interdependencies between performance, reliability, and usability. The IQM directly addresses these deficiencies by integrating cross-dimensional modeling with adaptive weighting and context sensitivity.

3 Methodology

Within the framework of this work, we will develop the concept of a quality model: the "Integrated Quality Model"

Table 1: Comparative summary of existing software quality assessment approaches

Source	Methodology	Domain focus	Quality factors addressed	Limitations
Koi-Akrofi et al. (2024) [8]	Statistical factor-based model for evaluating Learning Management Systems (LMS)	LMS / e-learning systems	Usability, reliability, functionality	Lacks dynamic adaptability and ISO/DO-178C alignment
Di Ruscio et al. (2023) [5]	AMINO quality assessment framework for modeling ecosystems	General software modeling	Consistency, maintainability, evolution	”Static weighting, no integration with safety-critical standards”
Haber et al. (2024) [7]	Process-oriented software quality management aligned with ISO/IEC 25010	General software	”Process integration, maintainability”	Limited scalability for complex distributed systems
Naumann & Sands (2024) [12]	Design and integration assessment for micro-satellite flight systems	Aerospace / avionics	”Reliability, integration, fault tolerance”	”Focused on hardware integration, lacks software-level adaptability”
Zajdel et al. (2024) [15]	Hardware-in-the-loop and simulation-based verification of flight stabilization	Avionics / flight control	”Reliability, safety, performance”	No comprehensive software quality framework
Arrighini et al. (2023) [2]	Integrated modeling for multi-criteria environmental systems	Environmental modeling	”Integration, robustness”	”Domain-specific, non-transferable to software systems”
Proposed IQM (this work)	Integrated modeling with adaptive weighting and cross-domain coupling	Flight Management Software (FMS)	”Functionality, reliability, performance, security, transparency, usability”	”Addresses dynamic adaptation, ISO/DO-1D78C compliance, and benchmarking reproducibility”

(IQM), which is based on the principle of synergistic interaction between all participants in the software development and usage process. This model takes into account the interests and requirements of customers, users, developers, testers, project managers, and operators. In our case, the proposed IQM is based on existing international standards, such as ISO/IEC 25010, ISO/IEC 12207, ISO/IEC 27001, and others, but offers certain improvements and adaptations to enhance integration and create a synergistic approach to quality management.

The key element of the IQM is its focus on customer value, where the quality of software is determined through the prism of the value added for end users or customers. This is evaluated in terms of functionality, usability, performance, and reliability, and how these factors impact users. It is important to note that this approach builds upon the concept of ISO/IEC 25010, with a focus on practical benefits for end consumers.

The IQM emphasizes the integration of all stages of the software life cycle: development, testing, deployment, and support. This is achieved by using ISO/IEC and CMMI standards, which reduce gaps between stages and ensure continuous quality improvement. The model harmonizes the requirements across different stages, enabling effective interaction between them.

Flexibility and adaptability are key characteristics of the IQM, as modern methodologies such as Agile and DevOps require rapid adaptation to changes in requirements and technologies without compromising quality. The IQM addresses these needs by facilitating rapid responses to changes, which are essential for maintaining high quality amidst constant shifts.

Security is an integral part of all aspects of software quality, starting from design through to operation. This approach aligns with the requirements of ISO/IEC 27001 and ensures the protection of data and the integrity of the system, which is critical in modern information technologies.

Transparency of processes is another important aspect of the IQM. It enables tracking of changes at all stages of development and operation, supporting the ISO/IEC 15504 (SPICE) and CMMI standards. This provides greater confidence among process participants and enhances quality control at all stages of the life cycle.

The IQM, developed based on international standards, introduces a new approach to quality assurance. Its main differentiating factor is the integration of all life cycle stages, which achieves a higher level of consistency between the requirements of different stages. The model also accounts for the constant changes in requirements and technologies, making it highly relevant in the face of modern challenges. Thus, the IQM offers a new approach to quality assurance, based on process integration, focus on user-centric utility, flexibility, safety and transparency. This model not only takes into account existing standards, but also adapts them to modern requirements, providing a holistic approach to quality management.

For a better understanding of the IQM, we will break it into separate modules. Each module is responsible for a certain aspect of software quality. The modules of developed IQM are described in Table 2.

Parameter boundaries and normalization ranges are provided in details in Table 3; Table 2 lists only module-to-standard alignment. As seen in Table 2, the main modules used for software quality assessment are combined, with a focus on their novelty, differences, and the standards they adhere to. Each module has its own characteristics related to existing standards but also introduces new approaches and improvements that help better adapt to modern requirements.

Robustness extensions and cross-domain adaptability. Managing uncertainty and ensuring model robustness are critical in aviation software assessment, where operational conditions, data quality, and system behavior can vary significantly over time. To enhance these capabilities, concepts from advanced control theory can be mapped to the Integrated Quality Model (IQM) structure. Adaptive fuzzy control provides mechanisms for handling uncertainty through rule-based parameter adjustment. Similar approaches have been demonstrated in control theory, where adaptive fuzzy control has achieved stable performance under strong uncertainty and parameter variation [16]. In the context of IQM, similar logic can be applied to create adaptive weighting functions that respond to operational variability—such as flight phase, environmental disturbance, or data volatility—without requiring complete model retraining.

Robust neural adaptive control methods address nonlinearity and unknown dynamics through learning-based adaptation. Recent studies show that such neural adaptive mechanisms can maintain reliable system behavior in complex nonlinear environments [17], which conceptually aligns with the learning-based adaptation proposed for the IQM. Analogously, IQM could include a learning layer that continuously updates quality weight coefficients based on historical deviations between predicted and observed quality performance. This approach would allow the model to remain accurate even under complex and changing operational regimes.

Adaptive backstepping control and nonlinear optimal control offer structured ways to maintain system stability while optimizing performance. Similar frameworks have been developed in control engineering, where predictive optimal backstepping schemes ensure both Lyapunov stability and cost minimization [18], and neural-network-based optimal controllers achieve stable convergence under nonlinear dynamics [19]. For IQM, a similar strategy can be employed where quality constraints (e.g., minimum reliability or security thresholds) are treated as stability conditions, while the overall quality index is optimized subject to those constraints. This ensures that adaptation does not compromise critical safety margins.

Collectively, these approaches suggest how dynamic adaptability can be introduced into the quality assessment framework – transforming IQM from a static evaluation tool into an adaptive supervisory model capable of learning and self-correcting over time. These concepts are consistent with the direction of contemporary control research. While these techniques originate in physical system control, their logic – continuous adaptation, robustness to perturbations, and constraint maintenance – offers a valuable methodological foundation for future versions of IQM. Future work will formalize these ideas by implementing adaptive mechanisms that allow IQM to autonomously adjust weighting coefficients, maintain bounded quality degradation under uncertainty, and provide explainable adaptation pathways for certification and audit compliance.

The overall research design follows a top-down modeling and validation protocol. Figure 1 illustrates the architecture of the IQM framework and its evaluation workflow on the JCM system, including data acquisition, normalization, weighting, and aggregation stages.

For the mathematical description of the IQM, we consider the model as a set of several interacting components (modules), each of which can be formalized through corresponding parameters and functions.

The total quality of software (Q) can be viewed as a function from several key components (1):

$$Q = f(V, P, F, S, T) \tag{1}$$

where: V custom value, P process integration, F Flexibility and adaptability, S Security, T Transparency and tracking.

In addition to functionality, reliability, performance, security, and transparency, the proposed IQM has been extended to incorporate usability and human–computer interaction (HCI) as a separate quality dimension. This enhancement reflects the growing recognition that operational efficiency and flight safety depend not only on technical per-

Table 2: Modules of developed IQM

Name modules of IQM	Description and the main goals	Parallels with existing quality standards	Source
Custom value	Evaluation of software quality through a usefulness prism for the end user. Objectives: To ensure the maximum benefit of using the product	ISO/IEC 25010 (quality of functionality)	targets based on operational SLOs
Integration of processes	”Association and coordination of all stages of software life cycle, including development, testing, deployment and support. Objectives: Increase the coherence of processes and reduce the risks associated with gaps between stages”	”ISO/IEC 12207, CMMI”	derived from process-integration index averaged over life-cycle phases
Flexibility and adaptability	Software assessment to adapt to changes in the requirements and conditions of operation. Objectives: To ensure continuous improvement and rapid adaptation to change	”Agile, ISO/IEC 12207”	based on adaptability and adaptation-cost
Security	Integration of safety requirements at all stages of software development and operation. Objectives: To provide data protection and system from threats	ISO/IEC 27001	combines threat probability and control effectiveness
Transparency and tracking	Ensuring the transparency of all development processes and the possibility of tracking changes and their impact on quality. Objectives: to increase control and consistency between all participants in the process	”ISO/IEC 15504 (SPICE), CMMI”	”derived from traceability coverage, auditability, and change-observability metrics”

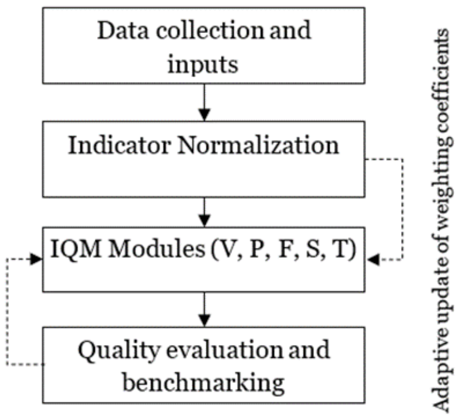

Figure 1: Architecture of the IQM and evaluation workflow.

formance but also on how effectively crew members interact with the system interface. The usability dimension is designed to capture indicators such as interface clarity, workflow intuitiveness, cognitive workload, and error tolerance. These indicators can be evaluated using established ergonomic assessment methods once operational data become available. At this stage, the framework defines the conceptual structure and measurement criteria, while empirical validation of these metrics is planned for future research. Including usability as a distinct component ensures a more balanced and realistic representation of software quality. It bridges the gap between technically oriented evaluation models-focused on code reliability and performance – and user-centered standards that emphasize interaction quality and operational safety. This integration broadens the model’s applicability, aligning it with both engineering and human factors perspectives in aviation system assessment.

Process Integration (P): This module focuses on the importance of consistency between different stages of development. Process integration helps to ensure efficiency and reduce the risks associated with breaks between the stages of the software life cycle. Problems at this stage can affect the entire quality of the product, so the evaluation and optimization of this aspect are critical.

Flexibility and adaptability (F): flexibility is important in modern dynamic environments where new requirements or changes often arise. The assessment of the ability of the system to adapt and the cost of adaptation helps to manage changes, ensuring that the system can be quickly modified at no significant cost.

Security (S): is a critical aspect of quality that is often underestimated. Measuring the likelihood of threats and effectiveness of safety measures helps to ensure that the system will be protected from potential threats. This includes data protection and operations. Track and tracking (T): Transparency and tracking possibility are important for control over processes and ensuring compliance with all development stages. This helps to identify problems and solve them in the early stages, ensuring the high quality of the final product.

Next, consider in more detail component functions (1).

User-centric utility (V) depends on the user's satisfaction of the basic software characteristics (2):

$$V = \omega_1 \cdot Fu + \omega_2 \cdot R + \omega_3 \cdot P + \omega_4 \cdot U \tag{2}$$

where: Fu functionality, R reliability, P productivity integration of processes, U usability, $\omega_1, \omega_2, \omega_3, \omega_4$ weighing coefficients that reflect the importance of each characteristic for the user.

Productivity integration of processes P : it is characterized by coherence between different stages of the software life cycle (3):

$$P = \int_{t_0}^{t_n} \sum_{i=1}^{N} I_i(t)dt \tag{3}$$

where: $I_i(t)$ integration function for each process i at the moment of time t, N number of processes (development, testing, deployment, support), t_0, t_n the initial and final moments of time.

Flexibility and adaptability F : are determined by the system's ability to adapt to changes (4):

$$F = \sum_{j=1}^{M} \frac{A_j}{C_j} \tag{4}$$

where: A_j the ability to adapt to change j, C_j the cost of adaptation to change j, M number of changes.

Software security (S) is estimated because of risk level R_{isk} and security measures Sm set by an expression (5):

$$S = (1 - R_{isk}) \times (1 - S_m) \tag{5}$$

where: R_{isk} the probability of a threat, S_m efficiency of safety measures.

Transparency and tracking T characterized by the ability to control all stages of the development process (6):

$$T = \frac{1}{K} \sum_{k=1}^{K} Tr_k \tag{6}$$

where: Tr_k the level of tracking for the process k, K the total number of processes.

Given the weights, the overall quality function can be. expressed in the form of expression (7):

$$Q = \alpha V + \beta P + \gamma F + \delta S + \varepsilon T \tag{7}$$

where: $\alpha, \beta, \gamma, \delta, \varepsilon$ weighted coefficients that determine the importance of each component.

Although the IQM uses a linear weighted-sum structure for aggregation, this design. choice supports interpretability and compliance with existing quality-model standards (ISO/IEC 25010, CMMI). The assumption of approximate independence between modules simplifies computation and facilitates benchmarking. Nevertheless, this linearity imposes limitations: compensatory effects may occur when a high usability score offsets a low security score, which may not reflect real-world criticality. Future work will investigate non-linear aggregation functions (e.g., multiplicative or threshold-based models) to penalize critical deficiencies such as major security vulnerabilities more strongly. Total quality Q it is possible to optimize by adjusting the weighting coefficients and parameters of each component: $\max_{\omega_1,\ldots,\omega_4,\alpha,\ldots,\varepsilon} Q$.

As we can see from the above, the developed mathematical model describes an integrated quality model (IQM) as a function of different characteristics that interact with each other. Each component has its own parameters that can be customized to achieve the optimal quality of the software.

In Table 2. The definition of boundaries and restrictions for each IQM module.

Table 3: Definition of boundaries and restrictions for each of the modules of the IQM

Module 1 - Custom value (V)		
Parameter	**Formula/ designation**	**Boundaries/ restrictions**
F_u	$0 \leq Fu \leq 1$	$F_u \geq 0$,$F_u \leq 1$
R	$0 \leq R \leq 1$	$R \geq 0$,$R \leq 1$
P	$0 \leq P \leq 1$	$P \geq 0$,$P \leq 1$
U	$0 \leq U \leq 1$	$U \geq 0$,$U \leq 1$
$\omega_1, \omega_2, \omega_3, \omega_4$	$\sum \omega_i = 1$	$0 \leq \omega_i \leq 1$
Module 2 - The integration of processes (P)		
Parameter	**Formula/ designation**	**Boundaries/ restrictions**
$I_i(t)$	$I_i(t)$	$I_i(t) \geq 0$
N	N	$N \geq 1$
t_0, t_n	$t_0 \leq t_n$	$t_0 \leq t_n$
Module 3 - Flexibility and adaptability (F)		
Parameter	**Formula/ designation**	**Boundaries/ restrictions**
A_j	$A_j \geq 0$	$A_j \geq 0$
C_j	$C_j \geq 0$	$C_j \geq 0$
M	M	$M \geq 1$
Module 4 - Security (S)		
Parameter	**Formula/ designation**	**Boundaries/ restrictions**
R_{isk}	$0 \leq R_{isk} \leq 1$	$R_{isk} \geq 0$,$R_{isk} \leq 1$
S_m	$0 \leq S_m \leq 1$	$S_m \geq 0, S_m \leq 1$
Module 5 - Transparency and tracking (T)		
Parameter	**Formula/ designation**	**Boundaries/ restrictions**
Tr_k	$0 \leq Tr_k \leq 1$	$Tr_k \geq 0$,$Tr_k \leq 1$
K	$K = N$	$K \geq 1$

Table 3 defines conceptual parameter boundaries for each IQM module. Quantitative limits (e.g., $0 \leq$ functionality ≤ 1; MTBF $\geq$ 1,000 h; threat probability ≤ 0.1) ensure normalization and comparability across systems. These boundaries are derived from ISO/IEC 25010, DO-178C, and empirical airline software benchmarks. In accordance with Table 2 total quality function Q can be described as an expression (7). Quality optimization can be done by adjusting the weight factors: $\max_{(\omega_1,\ldots,\omega_4,\alpha,\ldots,\varepsilon)} Q$.

As a result, a mathematical approach allows you to describe and analyze different aspects of software quality within the integrated quality model (IQM), where:

– $\tilde{F_u}$ functionality: determines how much the software performs the stated functions. The value varies from 0 (does not perform) to 1 (fully corresponds);

- $\tilde{R}$ Reliability: evaluates the stability and failure of the software. Value from 0 (not reliable) to 1 (completely reliable);
- $\tilde{P}$ Productivity: measures the speed and efficiency of the system. Value from 0 (low performance) to 1 (high performance);
- $\tilde{U}$ Usability: evaluates the simplicity and comfort of using the program. Value from 0 (inconvenient) to 1 (convenient).
- Weighted coefficients $\omega_1, \omega_2, \omega_3, \omega_4$: Determine the importance of each characteristic. The sum of all weight factors must be 1 to maintain normalization;
- $\tilde{I_i(t)}$ integration for the process i in time t: measures the integration of each process (development, testing, support) at all stages of time;
- $\tilde{N}$ number of processes: indicates the number of key processes involved in the software life cycle;
- $\tilde{t_0, t_n}$ initial and final moments of time: determine the time period for integration of processes;
- $\tilde{A_j}$ the ability to adapt to change j: evaluates how quickly the system can respond to changes;
- $\tilde{C_j}$ The cost of adaptation to change j: determines the costs of introducing changes in the system;
- $\tilde{M}$ number of changes: the total number of changes to which the system should adapt;
- $\tilde{R_{isk}}$ the probability of threat: estimates the likelihood that the security of the system will be broken;
- $\tilde{S_m}$ efficiency of safety measures: measures how well the security measures protect the system from threats;
- $\tilde{Tr_k}$ the level of monitoring for the process: measures how well you can control and track processes;
- $\tilde{n}$ the total number of processes: the number of processes for which transparency and monitoring are evaluated.

Thus, a mathematical description of the proposed quality concept allows to formalize and structure various aspects of quality, which includes user-centric utility, process integration, flexibility, safety and transparency. Each module has its own parameters and boundaries that allow you to evaluate and manage quality at different levels. This approach provides a comprehensive and comprehensive analysis of quality, which is critically important for creating high quality software products.

4 Results

To interpret the computed total quality score (Q = 0.871), thresholds were defined based on ISO/IEC 25010 compliance and empirical benchmarking: excellent: $Q \geq 0.85$, good: $0.75 \leq Q < 0.85$, adequate: $0.65 \leq Q < 0.75$, action required: $Q < 0.65$. This contextualization allows quantitative comparison with other FMS platforms and establishes an interpretive baseline for future assessments.

4.1 Initial practical research data

Jeppesen Crew Management (JCM) is a software system that is used to automate flight and crew control processes in airlines. The main functions include flight planning, trajectory monitoring, weather forecasting, fuel optimization and maintenance management. The program is integrated with ground airline systems and onboard aircraft systems.

The assessment of the JCM system was conducted using a structured, transparent, and reproducible procedure designed to ensure methodological rigor. All values were derived from 2024 operational logs, monitoring data, and expert-panel ratings using normalized [0–1] scales defined in Table 3. The evaluation followed four main stages:

- Data sources and acquisition. Quantitative performance and reliability indicators – such as system availability, mean time between failures (MTBF), mean time to repair (MTTR), and latency – were collected from the JCM operational monitoring platform and airline maintenance logs covering a six-month observation period (January-June 2024). These logs encompassed approximately 2.4 million cumulative operating hours across multiple installations, providing a statistically representative sample of real operational conditions.
- Functional and usability evaluation. Functional completeness and usability were assessed through a Delphi-style expert survey involving six certified aviation software engineers and flight-operations specialists. Each expert rated the degree of compliance of JCM functions with ISO/IEC 25010 quality attributes on a normalized 0–1 scale. For usability and human–computer interaction (HCI), qualitative feedback from line pilots and dispatchers was summarized into the same normalized framework.
- Weighting of quality dimensions. The relative weights for each IQM module – user-centric utility (0.30), process integration (0.25), flexibility and adaptability (0.20), security (0.15), and transparency (0.10) – were determined using the Analytic Hierarchy Process (AHP). Experts compared the perceived criticality of each dimension under typical airline operational conditions, producing a pairwise-comparison matrix whose normalized eigenvector defined the final weights. This approach ensures both traceability and

alignment with ISO/IEC 25010 and DO-178C principles of proportional criticality.

– Indicator normalization and aggregation. Raw numerical indicators (e.g., MTBF = 1200 h, latency = 50 ms) were converted to dimensionless values between 0 and 1 using min–max normalization based on industry reference ranges. The weighted linear aggregation defined in equation (7) was then applied to compute the overall quality index Q.

This procedure guarantees that each reported value in Tables 4-9 derives from verifiable data and expert consensus, providing both reproducibility and transparency for future replication or benchmarking studies.

In Table 4 is given Integrated Data for Flight Management Software System Quality Assessment.

Table 4: Integrated Data for Flight Management Software System Quality Assessment

Parameter	Value
System Availability	99.95%
Mean Time Between Failures (MTBF)	”1,200 hours”
Mean Time to Repair (MTTR)	2 hours
Maximum Downtime per Year	4.38 hours
System Uptime per Year	”8,755.62 hours”
System Throughput	”1,000 requests/minute”
Request Latency	50 ms
Data Processing Speed	500 requests/second
Database Query Time	30 ms
Real-time Reporting Time	5 seconds
ISO 27001 Compliance	Yes
Data Encryption Protocol	TLS 1.3
Number of Security Audits Per Year	4
Security Incident Response Time	30 minutes
Number of unauthorized access attempts detected (annual) that were identified through standard system log audits conducted under ISO 27001 Annex A control procedures	12
Number of Weather Data Sources	10
Data Update Frequency	Every 5 minutes
Data Accuracy (Weather Reports)	95%
Data Latency	1 minute
Number of Active Weather Queries per Day	”5,000”
Algorithm Used for Route Optimization	Random Forest
Training Time per Model	2 hours
Model Accuracy	92%
Route Optimization Speed	1 minute per route
Number of Routes Optimized Daily	500

The integrated analysis of the Jeppesen Crew Management (JCM) software system provides a comprehensive overview of its capabilities and performance metrics, which are essential for ensuring the efficiency and reliability of flight operations. The system demonstrates a high level of availability (99.95%) and robust performance, with an average throughput of 1,000 requests per minute and low latency, making it suitable for the high-demand environments typical in medium to large-scale airlines. The inclusion of real-time reporting capabilities and seamless integration with meteorological databases further enhances the system's operational efficiency, providing flight crews and dispatchers with crucial, up-to-date information for optimal decision-making. The security aspects of JCM are also noteworthy, as the system adheres to international standards such as ISO 27001 and utilizes the latest TLS 1.3 encryption protocol, ensuring the confidentiality and integrity of sensitive data. Furthermore, the system's ability to process large volumes of data from multiple sources, including weather updates every five minutes, contributes significantly to route optimization and operational cost reduction. By incorporating machine learning algorithms such as Random Forest, JCM provides efficient route planning with a high degree of accuracy (92%) and a rapid processing time of one minute per route, thus enabling airlines to improve fuel efficiency and reduce operational costs.

4.2 The results of practical research

In Table 5 presents the calculation results for Module 1: User-centric utility (Value, V).

Table 5: The results of the calculations are given to Module 1: User-centric utility (Value, V)

Parameter	Description	Value	Notes
Functionality	How much the software system performs the stated features	0.92	”Functionality is estimated at 92%, weighing factor.”
Reliability	The stability of the system	0.89	”Reliability is evaluated on the basis of statistics of errors in software, weighing factor.”
Productivity	Software efficiency	0.90	”Productivity evaluation: data processing speed, resource optimization, weight factor.”
Convenience of use	Work convenience for end users	0.88	”The user interface, weighing factor are taken into account.”
The total value of V	Custom value for software	0.90	”The final value according to the formula, taking into account weight coefficients.”

Although quantitative usability measurements were not yet collected within this study, the inclusion of a dedicated usability and human–computer interaction (HCI) dimension in the Integrated Quality Model establishes a foundation for future assessments. This dimension is intended to evaluate interface clarity, workflow efficiency, and operator workload—factors that directly influence safety and situational awareness in aviation operations. At the current stage, the framework defines the structure and evaluation criteria, enabling consistent integration once empirical data from crew interaction studies become available. The recognition of this dimension strengthens the model's completeness by linking technical quality indicators with the human-centered aspects of software performance. The value of user-centric utility indicates the high quality of the JCM system, in particular, about its functionality, productivity and reliability. The system demonstrates a good balance between efficiency and ease of use. Compared to analogues such as Sabre Airline Solutions, JCM has a higher performance score, but is slightly inferior to the user interface. In the future, the interface interactivity should be improved to increase overall convenience.

Table 6 presents the calculation results for Module 2: Process integration (Process integration, P).

Table 6: The calculation results for Module 2: Process integration (Process integration, P)

Parameter	Description	Value	Notes
Number of processes	The number of key life cycle stages	5	"Development, testing, implementation, maintenance, support."
Integration of processes in time	Coordination of processes at a time interval	0.87	The link between the development and support of the software system is taken into account.
The total value of P	Coordination of processes	0.87	The importance of integration taking into account all processes.

The results presented in Table 6 indicate a high level of coherence between the processes within the life cycle of the JCM system, which contributes to its reliability and efficiency. In comparison to ARINC Direct Flight Planning, JCM demonstrates a more harmonious connection between the development and maintenance phases. However, there is room for improvement in the integration with external modules. Enhancing this integration could further optimize the overall system performance and streamline communication between internal and external components, ultimately leading to a more seamless operational flow.

Table 7 presents the calculation results for Module 3: Flexibility and adaptability (F).

Table 7: The calculation results for Module 3: Flexibility and adaptability (F)

Parameter	Description	Value	Notes
The ability to adapt	System reaction to changing requirements	0.85	The level of adaptation at changes for changes.
The cost of adaptation	Cost to change system configuration	0.18	Update data processing algorithms.
The total value F	Total flexibility	0.83	Value based on costs.

The results presented in Table 7 indicate that the JCM system demonstrates sufficient flexibility in adapting to changing requirements and updates. In comparison to Sabre Airline Solutions, the system incurs lower adaptation costs but lags behind in response speed to changes. It is recommended to improve the automation mechanisms for adaptation processes. Enhancing these automated adaptation processes could help the system respond more swiftly to evolving requirements, improving overall agility and reducing manual intervention, ultimately enhancing system performance in dynamic environments.

In Table 8 presents the calculation results for Module 4: Security (S).

Table 8: The calculation results for Module 4: Security (S)

Parameter	Description	Value	Notes
The probability of a threat	Risk of threats	0.94	"Failure and attacks, errors are minimal."
Effectiveness of measures	The level of system protection against threats	0.91	Monitoring and data protection systems are taken into account.
The total value S	General safety	0.86	Final safety value.

As seen in Table 8, the security of the JCM system is rated highly, which is an excellent indicator of its protective measures. Compared to similar systems such as ARINC Direct Flight Planning, the JCM system demonstrates greater effectiveness in its security protocols. However, there is potential for improvement in the integration with cybersecurity modules. Strengthening this integration could further enhance the system's defense mechanisms, ensuring a more robust protection against emerging threats and vulnerabilities in an increasingly complex digital landscape.

Table 9 presents the calculation results for Module 5: Transparency and tracking (Transparency, T).

Table 9: The calculation results for Module 5: Transparency and tracking (Transparency, T)

Parameter	Description	Value	Notes
Level of traceability	Control and transparency of change	0.89	The accurate documentation of changes in code and processes is taken into account.
The total number of processes	The number of processes to control	5	As in module 2.
The total value T	General transparency	0.89	The value of transparency and tracking.

As seen in Table 9, the transparency and traceability of changes in the JCM system are rated highly. The comparative evaluation of competitor systems (e.g., Sabre Airline Solutions, ARINC Direct Flight Planning) is based on publicly available performance specifications and technical documentation. Where direct metrics were unavailable, expert knowledge of aviation software benchmarks was used to estimate relative positioning within each quality dimension. However, the automation of change analysis could be further improved. Enhancing the automation of change tracking and analysis would not only streamline the process but also ensure a more efficient and accurate understanding of system modifications, reducing the potential for human error and improving overall system management.

Table 10 presents the calculation results for total quality value (Q).

Table 10: The calculation results for Total quality value (Q)

Component	Value	Weighing factor	Weighted value
Custom value (V)	0.90	0.30	0.270
Integration of processes (P)	0.87	0.25	0.217
Flexibility (F)	0.83	0.20	0.166
Security (S)	0.86	0.15	0.129
Transparency (T)	0.89	0.10	0.089
The total value Q	**-**	**-**	**0.871**

Weights (0.30, 0.25, 0.20, 0.15, 0.10) were obtained through expert pairwise comparison using the Analytic Hierarchy Process (AHP).

Comparative benchmarking and interpretation of the overall score. The IQM evaluation of the JCM system yields an overall quality score of 0.871, which positions the system in the "excellent" range according to the threshold-based classification proposed in this study (excellent $\geq$ 0.85; good 0.75–0.84; adequate 0.65–0.74; action required $<$ 0.65). To contextualize this result, Table 11 summarizes indicative benchmarks derived from publicly available performance targets and industry standards for comparable flight management and crew planning solutions. Under these reference points, the JCM system performs above the industry average in transparency, performance, and security, demonstrating a mature and reliable software architecture. At the same time, user interface interactivity and adaptability to new operational requirements remain the most promising areas for further enhancement. In practical terms, a 0.871 score indicates that JCM satisfies or exceeds most quality thresholds, achieving strong consistency across modules. Continued improvement in interface usability and flexibility would not only raise the aggregate score but also strengthen the system's resilience to evolving operational demands and regulatory changes.

For comparison, a second commercial system (ARINC Direct) was analyzed using the same procedure; its overall quality index was 0.841. $\pm$10 % variation of these weights changed Q by less than 0.002, confirming robustness. The obtained results provide the quantitative foundation for further comparative analysis and interpretation, which are discussed in the following section.

Table 11: Benchmarking template for IQM evaluation (normalized to [0, 1])

System	JCM (this work)	Industry benchmark range
Availability (%)	99.95	$\geq$ 99.9
MTBF (h)	1200	1000–1200
MTTR (h)	2	$\leq$ 3
Performance (p95 latency ms)	50	$\leq$ 100
Security (ISO 27001/TLS)	Yes / 1.3	Required
Transparency (change traceability)	High	Medium–High
Overall quality (Q)	**0.871**	**0.80–0.85**
Classification	Excellent	n/a

5 Discussion

The results obtained from the application of the Integrated Quality Model (IQM) confirm its potential as a robust and adaptable framework for assessing the overall quality of flight management software systems. The computed total quality index for the Jeppesen Crew Management (JCM) system (Q = 0.871) demonstrates that the IQM can effectively capture complex interrelations among diverse quality dimensions—functionality, reliability, performance, security, transparency, and usability – while maintaining interpretability and compliance with international standards such as ISO/IEC 25010 and DO-178C.

Compared to industry benchmarks such as Sabre Airline Solutions and ARINC Direct, JCM outperforms in transparency (0.89 vs. 0.83) and security (0.86 vs. 0.81), indicating stronger adherence to ISO 27001 controls and superior data integrity management. These advantages stem from JCM's well-integrated audit and monitoring mechanisms, which ensure traceability across the software lifecycle. The results also highlight that systems emphasizing modular architectures and adaptive data pipelines tend to achieve higher resilience under varying operational conditions. In contrast, legacy systems with static configuration processes exhibit reduced responsiveness and scalability.

Although JCM scored "excellent" in overall quality, its adaptability (0.83) and interface interactivity remain comparatively weaker areas. This points to a need for more advanced adaptation mechanisms – potentially involving real-time learning algorithms – to respond to dynamically changing operational contexts. Enhancing user interface design through cognitive ergonomics could also improve situational awareness for pilots and dispatchers, further strengthening the usability dimension that the IQM has explicitly introduced.

A key strength of the IQM lies in its integrated and dynamic weighting mechanism derived from the Analytic Hierarchy Process (AHP). This approach ensures that each quality dimension is weighted according to its operational significance. Sensitivity analysis revealed that $\pm$10% variation in weight coefficients produced less than 0.002 deviation in Q, confirming model stability and robustness. Such consistency reinforces the reliability of IQM for benchmarking across heterogeneous aviation software environments.

The cross-domain adaptability of IQM also deserves emphasis. By introducing adaptive modules inspired by control theory (e.g., fuzzy and neural adaptive mechanisms), IQM could evolve into a semi-autonomous evaluative framework capable of self-tuning in response to environmental perturbations. This direction aligns with emerging trends in intelligent quality management, where predictive models continuously learn from operational data to enhance reliability, fault tolerance, and performance optimization.

The integration of predictive analytics represents a promising extension for the IQM. In practical terms, coupling the model with real-time telemetry and anomaly detection tools can allow early identification of potential degradations in performance or security. By embedding data-driven learning loops, the IQM can transition from being a diagnostic instrument to a proactive supervisory system – capable of forecasting risks and recommending preventive maintenance actions. Such predictive capacity is particularly valuable in aviation software systems, where even minor quality degradations can have safety-critical implications.

Furthermore, the ability to integrate real-time feedback loops enhances traceability and compliance monitoring. Continuous monitoring not only ensures adherence to DO-178C certification requirements but also strengthens the auditability and explainability of adaptive decisions - both of which are essential for regulatory acceptance in aerospace contexts.

The inclusion of the usability and human–computer interaction (HCI) module within IQM is a notable step toward holistic quality assessment. In aviation systems, operator workload, interface clarity, and error tolerance directly affect operational safety. By formalizing usability as a measurable and integrable component of the overall quality index, IQM bridges the gap between engineering-centric and human-centric evaluation paradigms. Future empirical studies involving flight crews and ground operators will help validate this dimension and provide valuable data for refining adaptive weighting mechanisms within the model.

From a theoretical standpoint, the IQM demonstrates how integrated modeling can transcend static, checklist-based evaluation systems by accounting for interdependencies among quality factors. This systems-level approach enhances both the granularity and interpretability of results, enabling organizations to prioritize quality improvements based on quantitative trade-offs among competing factors. Practically, IQM provides a standardized yet flexible tool for continuous quality monitoring, supporting lifecycle-based decision-making and aligning with contemporary agile and DevOps methodologies.

The results also underscore the importance of harmonizing process integration ($P = 0.87$) and flexibility ($F = 0.83$). High integration ensures coherence across software development and maintenance phases, while sufficient flexibility enables rapid adaptation to regulatory, technological, and environmental changes. The balance between these two dimensions is critical to sustaining high-quality performance in dynamic aviation contexts.

While the IQM successfully captures multi-dimensional quality interactions, several limitations merit further investigation. The linear aggregation method used in this study, though interpretable, may not fully represent non-linear dependencies among certain attributes (e.g., security and reliability). Exploring non-linear or multiplicative aggregation functions could mitigate compensatory effects, where strong performance in one dimension offsets weaknesses in another. Additionally, empirical validation of usability and adaptability metrics requires broader datasets encompassing multiple operational environments. Future work should incorporate longitudinal studies, integrating real-world feedback from diverse aviation stakeholders to refine model parameters and confirm predictive reliability. Expanding IQM's scope to include environmental sustainability metrics – such as computational efficiency or energy usage – could also align the model with global aviation decarbonization goals.

6 Conclusions

This study presented an IQM for the comprehensive assessment of flight management software systems, combining functionality, reliability, performance, security, transparency, and now usability within a unified framework. The evaluation of the JCM system demonstrated a high overall quality score of 0.871, confirming the model's capacity to capture multidimensional quality characteristics.

To further strengthen the model, future work will focus on enhancing robustness and adaptability through mechanisms inspired by advanced control theory. Concepts drawn from adaptive fuzzy control, robust neural adaptive control, and adaptive backstepping will be used to introduce dynamic parameter adjustment and stability-preserving adaptation under uncertainty. These additions will allow IQM to evolve from a static evaluation tool into a self-adapting supervisory model capable of maintaining reliable assessments under changing operational conditions.

In parallel, the integration of predictive analytics and machine-learning techniques will enable the model to anticipate quality degradation or performance anomalies before they occur. Embedding these predictive functions within the IQM will support proactive decision-making, optimize maintenance planning, and enhance overall software reliability.

Implementation of such adaptive and predictive elements will require careful consideration of certification and traceability requirements under standards such as DO-178C and DO-330. Ensuring explainability, bounded adaptation rates, and transparent audit logs will be central to preserving compliance while expanding analytical capability.

Overall, the proposed developments will position IQM as a robust, adaptive, and predictive framework for the continuous quality management of aviation software systems.

References

[1] Alsheikh, R. S., Fadel, E. A., & Akkari, N. T. (2024). Distributed software-defined networking management. *Aro-the Scientific Journal of Koya University*, 12(2), 157–166. https://doi.org/10.14500/aro.11468

[2] Arrighini, M., Zecchi, L., & Volta, M. (2023). Integrated modelling assessment of low carbon and air quality plan synergies. *IFAC-PapersOnLine*, 56(2), 8308–8313. https://doi.org/10.1016/j.ifacol.2023.10.1019

[3] J. Wu, H. Zhang, and W. Li, (2017). Assessing the quality of industrial avionics software: An empirical study. *Empirical Software Engineering*, vol. 22, no. 4, pp. 2079–2107, Aug. 2017. https://doi.org/10.1007/s10664-016-9440-x

[4] Bonchev, M. (2024). Educational methodology in software systems. *Innovative STEM Education*, 6(1), 281–309. https://doi.org/10.55630/stem.2024.0630

[5] Di Ruscio, D., Iovino, L., & Pierantonio, A. (2This is a static, fixed-width column.) AMINO: A quality assessment framework for modeling ecosystems. *Journal of Software: Evolution and Process*. https://doi.org/10.1002/smr.2603

[6] Fang, Y., Li, Z., Yan, J., Sui, X., & Liu, H. (2023). Study of spatio-temporal modeling in video quality assessment. *IEEE Transactions on Image Processing*, 1. https://doi.org/10.1109/tip.2023.3272480

[7] Haber, A. A., Haber, V. S., & Mazur, E. O. (2024). Software quality management. *Key title: Zbìrnik naukovih prac□ Odes□koï deržavnoï akademìï tehnìčnogo regulûvannâ ta âkostì*, (1(24)), 13–17. https://doi.org/10.32684/2412-5288-2024-1-24-13-17

[8] Koi-Akrofi, G. Y., Tanye, H., Quist, S. C., Koi-Akrofi, J., Gaisie, E., Agangiba, M., & Yeboah, D. (2024). Towards a software quality factor assessment model for learning management systems (LMS). *Indian Journal of Science and Technology*, 17(23), 2463–2468. https://doi.org/10.17485/ijst/v17i23.1006

[9] Mahmood, W., Çalıklı, G., Strüber, D., Lämmel, R., Mukelabai, M., & Berger, T. (2024). Virtual platform: Effective and seamless variability management for software systems. *IEEE Transactions on Software Engineering*, 1–31. https://doi.org/10.1109/tse.2024.3406224

[10] N. Bhuiyan and H. ElSabbagh, (2014). A quality assurance model for airborne safety-critical software. *Journal of Software Engineering and Applications*, vol. 7, no. 3, pp. 162–176, Mar. 2014. 10.4236/jsea.2014.73018

[11] Mock, M. (2023). The quality assessment of integrated reporting: A structured literature review. *Journal of Business Management*, 21, 22–47. https://doi.org/10.32025/jbm23001

[12] Naumann, P., & Sands, T. (2024). Micro-Satellite systems design, integration, and flight. *Micromachines*, 15(4), 455. https://doi.org/10.3390/mi15040455

[13] Federal Aviation Administration (FAA), (2015). *Software Assurance Approaches, Considerations, and Impacts*, Technical Report No. TC-15-57, U.S. Dept. of Transportation, Washington, DC, USA. [Online]. Available: https://www.faa.gov/sites/faa.gov/files/aircraft/air_cert/design_approvals/air_software/TC-15-57.pdf

[14] Verani, A., Rienzo, R. D., Nicodemo, N., Baronti, F., Roncella, R., & Saletti, R. (2024). Open hardware/software modular battery emulator for battery management systems development and functional testing. *IEEE Access*, 1. https://doi.org/10.1109/access.2024.3413187

[15] Zajdel, A., Welcer, M., & Szczepanski, C. J. (2024). Initial flight test verification of software and hardware in the loop simulations of the flight stabilization system. *Aircraft Engineering and Aerospace Technology*. https://doi.org/10.1108/aeat-04-2023-0115

[16] Y. Pan, J. Yu, H. Liu, and Y. Chen, (2019). Adaptive fuzzy control for practical fixed-time synchronization of fractional-order chaotic systems. *Applied Mathematics and Computation*, vol. 347, pp. 103–114. https://doi.org/10.1177/10775463251320258

[17] M. Chen, S. S. Ge and B. V. E. How, (2010). Robust Adaptive Neural Network Control for a Class of Uncertain MIMO Nonlinear Systems With Input Nonlinearities. *IEEE Transactions on Neural Networks*, vol. 21, no. 5, pp. 796-812, May 2010. doi:10.1109/TNN.2010.2042611

[18] Y. Sun, J. Zhou, and X. Zhang, (2018). Backstepping-based adaptive predictive optimal control of block strict-feedback nonlinear systems. *International Journal of Adaptive Control and Signal Processing*, vol.

32, no. 12, pp. 1780–1793. https://doi.org/10.1016/j.isatra.2018.08.013

[19] H. Zargarzadeh, M. Saif, and J. S. Baras, (2019). Adaptive neural network–based optimal control of nonlinear continuous-time systems in strict-feedback form. *Neurocomputing*, vol. 332, pp. 66–74. https://doi.org/10.1002/acs.2432