A Hybrid Assessment Model for Programming Education: Combining Automated Grading with Activity Monitoring

Ilkhom Boynazarov[1], Baxtiyor Radjabov[2], Uranchimeg Tudevdagva[3], Jamshid Salimov[4], Jasur Mamanov[5] and Fazilat Ochilova[1]
[1]Samarkand Branch of Tashkent University of Information Technologies named after Muhammad al-Khwarizmi, Ibn Sina Str. 2, 140100 Samarkand, Uzbekistan
[2]Chirchik State Pedagogical University, Amir Temur Str. 104, 111700 Chirchik, Uzbekistan
[3]Mongolian University of Science and Technology, Baga Toiruu Str. 34, 14191 Ulaanbaatar, Mongolia
[4]Jizzakh Branch of the National University of Uzbekistan named after Mirzo Ulugbek, Sh. Rashidov Str. 259, 130100 Jizzakh, Uzbekistan
[5]Denau University of Entrepreneurship and Pedagogy, Bog' Str. 112, 190100 Denau, Uzbekistan
ilhomboy1974@gmail.com, bahtiyorradjabov52@gmail.com, uranchimeg@must.edu.mn, jamshidsalimov8@gmail.com, jasurmamanov93@gmail.com, fazilat1816@gmail.com

Keywords: Hybrid LMS, Programming Education, Online Judge, Hybrid Assessment Model, Student Engagement, Activity Monitoring, Gamification, Learning Analytics, Automated Grading.

Abstract: Traditional Learning Management Systems (LMS) often fail to adequately capture student engagement and mainly rely on summative outcomes when assessing learning performance. This limitation is particularly critical in programming education, where the learning process itself is as important as the final code output. This paper proposes a Hybrid Intelligent LMS and a Hybrid Assessment Model designed specifically for programming courses. The system integrates an automated code evaluation module (Online Judge), gamified assessment components, and a real-time activity monitoring mechanism. Based on students' learning mastery (P) and engagement level (A), an overall efficiency index (E) is calculated. To evaluate the effectiveness of the proposed approach, a controlled quasi-experiment was conducted with 82 first-year Software Engineering students. Participants were divided into control and experimental groups and studied using a traditional LMS and the proposed Hybrid LMS, respectively. The collected data were analyzed using Student's t-test and Pearson correlation analysis. The results show that students in the experimental group achieved significantly higher learning mastery scores ($p < 0.001$) and that strong positive correlations exist between engagement and learning mastery, as well as between engagement and overall efficiency. These findings confirm that the proposed hybrid approach effectively improves student engagement and academic performance in programming education.

1 INTRODUCTION

The rapid development of information technologies has significantly transformed the requirements for teaching programming courses. Modern educational environments are expected not only to deliver theoretical knowledge but also to foster practical skills, independent thinking, and problem-solving abilities. Consequently, programming education requires technological solutions that combine theory and practice, provide immediate feedback, and consider students' engagement throughout the learning process. Digital tools specifically designed to support the preparation, submission, and evaluation of practical programming activities have also been proposed to improve the organization of programming education [1].

Traditional LMS platforms (e.g., Moodle) mainly organize instruction and assessment through test results and final assignment scores. The transition of coding education to online environments has also highlighted challenges related to learning time, student work patterns, and interaction within fully digital courses [2]. However, such approaches do not sufficiently reflect the time students spend learning, the number of attempts made during problem solving, or their learning progress. As a result, students' actual effort and engagement are often underestimated.

Previous studies indicate that automated assessment systems reduce instructor workload and provide fast, objective feedback to learners [3], [4]. Online Judge systems, in particular, offer fair and scalable mechanisms for evaluating programming assignments [5]. Nevertheless, most existing systems focus primarily on code correctness and pay limited attention to the learning process and student engagement.

In parallel, numerous studies demonstrate that gamification elements (points, badges, leaderboards) increase students' motivation and cognitive engagement [6]-[8]. Learning analytics dashboards further support self-regulated learning by visualizing students' progress [9]. However, these approaches are rarely integrated into a unified assessment framework.

To address these limitations, this paper proposes a Hybrid Intelligent LMS and a Hybrid Assessment Model specifically designed for programming education. The proposed approach combines learning mastery (P) and engagement level (A) into a single efficiency index (E), enabling evaluation of both learning outcomes and learning processes.

The main contributions of this paper are as follows:

- Design of a Hybrid LMS architecture integrating Online Judge, activity monitoring, and gamification;
- Proposal of a formal hybrid assessment model with mastery (P), engagement (A), and integrated efficiency (E) indicators;
- Validation through a controlled quasi-experiment providing statistical evidence of the model's effectiveness;
- Implementation of a security model for sandboxed code execution within the educational platform.

2 RELATED WORKS

2.1 Automated Assessment and System Performance Factors

Automated assessment systems have been widely studied as effective tools for reducing instructor workload and providing rapid feedback in programming education. The design of effective automated formative feedback is particularly important because the quality and structure of feedback directly influence students' programming learning processes [10]. Ala-Mutka [3] presented a comprehensive survey of automated assessment approaches and emphasized their pedagogical benefits. Ihantola et al. [4] reviewed modern systems for automatic evaluation of programming assignments and showed that test-based and static analysis approaches are dominant.

Online Judge systems provide fair and scalable mechanisms for evaluating programming tasks [5]. In such systems, student code is evaluated using hidden test cases and automatic judging procedures. However, most existing systems primarily focus on code correctness and give limited attention to code quality and the learning process.

Recent studies emphasize the need to enrich automated assessment with additional quality dimensions. Parvathy et al. [11] proposed a comprehensive model that evaluates code quality, style, and semantics. Moreover, system-level factors play a crucial role in e-learning success. Ghazinoory and Afshari-Mofrad [12] demonstrated that technical support and system reliability are among the most critical factors affecting e-learning outcomes. Similarly, Attia et al. [13] showed through an AHP-based analysis that system and service quality has the strongest impact on student satisfaction.

2.2 Student Engagement, Gamification, and Cognitive Analysis

Enhancing student engagement and motivation is a key objective of modern educational systems. Numerous studies confirm that gamification elements such as points, badges, and leaderboards increase students' interest and participation. Ibáñez et al. [7] reported that the use of points and badges in programming courses increases cognitive engagement. Ortiz-Rojas et al. [6] found that leaderboards positively influence learning outcomes. Triantafyllou et al. [8] further emphasized that gamification strengthens intrinsic motivation based on Self-Determination Theory. Recent systematic reviews further emphasize that these elements significantly impact student motivation and academic performance across various educational settings [14].

Learning analytics approaches support self-regulated learning by visualizing student activity and progress [9]. The broader integration of learning analytics and educational data mining also provides a methodological basis for analyzing learner behavior and educational activity data [15]. Studies conducted in the local context using the SURE model revealed deficiencies in course design and administrative systems regarding student engagement [16], [17]. In addition, cognitive diagnostic models aim to analyze students' internal knowledge states. The R-DINAbased model proposed by Cao and Tan [18] provides a way to visualize students' knowledge states. However, these approaches are rarely integrated into unified assessment frameworks.

Overall, existing studies typically address automated assessment, gamification, and learning analytics separately. In contrast, this work integrates these approaches into a single Hybrid LMS and proposes a hybrid assessment model that combines process-oriented and outcome-oriented indicators.

3 SYSTEM ARCHITECTURE

The proposed Hybrid LMS platform is designed based on a microkernel architecture to ensure scalability, security, and extensibility. The system consists of three main layers: presentation layer, application layer, and data layer (Fig. 1).

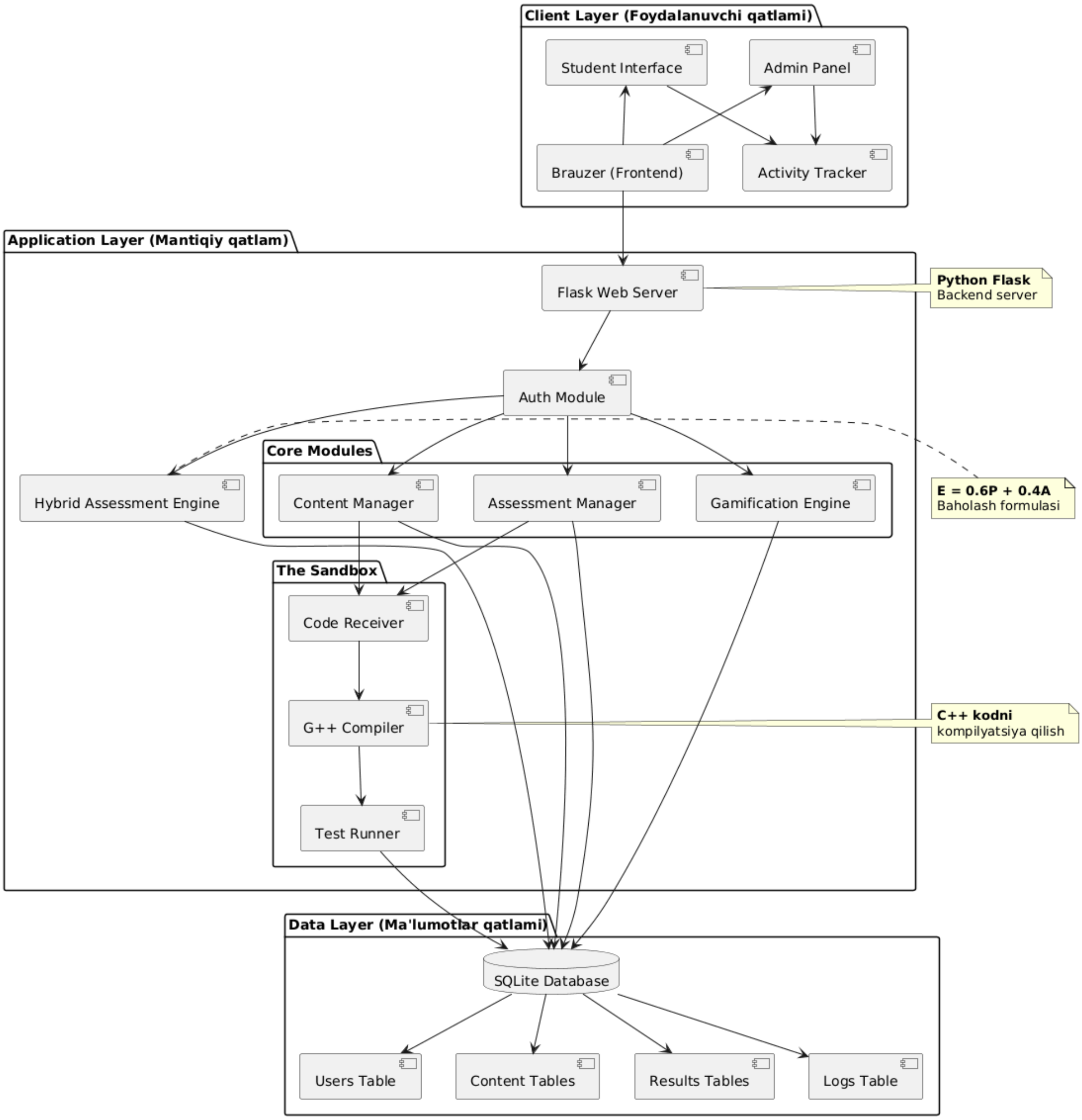

Figure 1: Three-layer architecture of the system. This architecture ensures secure and scalable integration of automated assessment and activity monitoring.

The presentation layer provides user interaction through a web interface and supports three roles: student, instructor, and administrator. A dedicated Student Dashboard visualizes the learning mastery indicator (P), engagement index (A), and overall efficiency index (E).

The application layer contains the core functional modules:

- Online Judge module – compiles and executes student code in real time and evaluates it using hidden test cases. The module supports multiple difficulty levels and performs automated result processing (Fig. 3).
- Gamification module – enhances motivation through time-limited questions, points, badges, and leaderboards.
- Activity Monitoring module – a JavaScript-based background service that tracks active window time and sends heartbeat signals to the server every 30 seconds.
- Assessment and Calculation module – computes the P, A, and E indicators according to the mathematical models.

3.1 Security and System Integration

For security purposes, code execution is performed in an isolated sandbox environment. Following the recommendations of Kurnia et al. [5], chroot and ulimit mechanisms are applied to prevent malicious code execution (Fig. 2).

The process flow demonstrates the effectiveness of the sandbox environment in isolating student code execution. The data layer stores information about users, learning materials, assignments, activity logs, and computed indicators. The database schema is designed using an ER diagram (Fig. 3).

The system supports three main roles: administrator, instructor, and student. Passwords are hashed using the Werkzeug library, while authentication and authorization are managed through @login_required decorators.

This architecture enables the integration of Online Judge functionality with traditional LMS features within a single platform and provides a solid foundation for implementing the hybrid assessment methodology described in the next section.

3.2 Implementation Details

To ensure the reproducibility of the proposed platform, the following technical specifications are provided:

- Tech Stack. The backend of the Hybrid LMS is implemented using the Python Flask framework, while the frontend is developed with HTML5, CSS3, and JavaScript.
- Database: High-performance data management is handled by PostgreSQL, which ensures ACID compliance for storing user profiles, assignment logs, and activity heartbeat signals.
- Online Judge Environment. The code evaluation module uses the g++ compiler (C++17). To ensure security, student code is executed in an isolated chroot sandbox with resource limits enforced via ulimit (e.g., 2.0s CPU time and 256MB memory per submission).
- Monitoring Mechanism. The activity tracking service sends heartbeat signals to the server every 30 seconds. It specifically monitors the active window focus; if the window loses focus or remains idle for more than 5 minutes, the engagement counter is paused to ensure data accuracy.
- Scalability. The microkernel architecture allows the system to process multiple submissions concurrently, with an average judging time of less than 1.5 seconds per code file.

4 PROPOSED METHODOLOGY

The main objective of this study is to develop a hybrid assessment methodology that considers not only students' final outcomes but also their engagement and effort during the learning process. For this purpose, learning mastery and engagement indicators are integrated into a unified mathematical model.

The proposed methodology is based on three core components:

- learning mastery indicator (P);
- engagement indicator (A);
- overall efficiency index (E).

This approach enriches traditional summative assessment with process-oriented indicators and enables comprehensive evaluation of student learning.

4.1 Mathematical Foundations of the Hybrid Model

The overall efficiency index is defined as a linear combination:

$$E = \alpha \cdot P + \beta \cdot A,\ \alpha + \beta = 1, \qquad (1)$$

where:

- P is the integrated learning mastery indicator;
- A is the engagement indicator;
- α, β are weighting coefficients. For experimental trials, $\alpha = 0.7$ and $\beta = 0.3$.

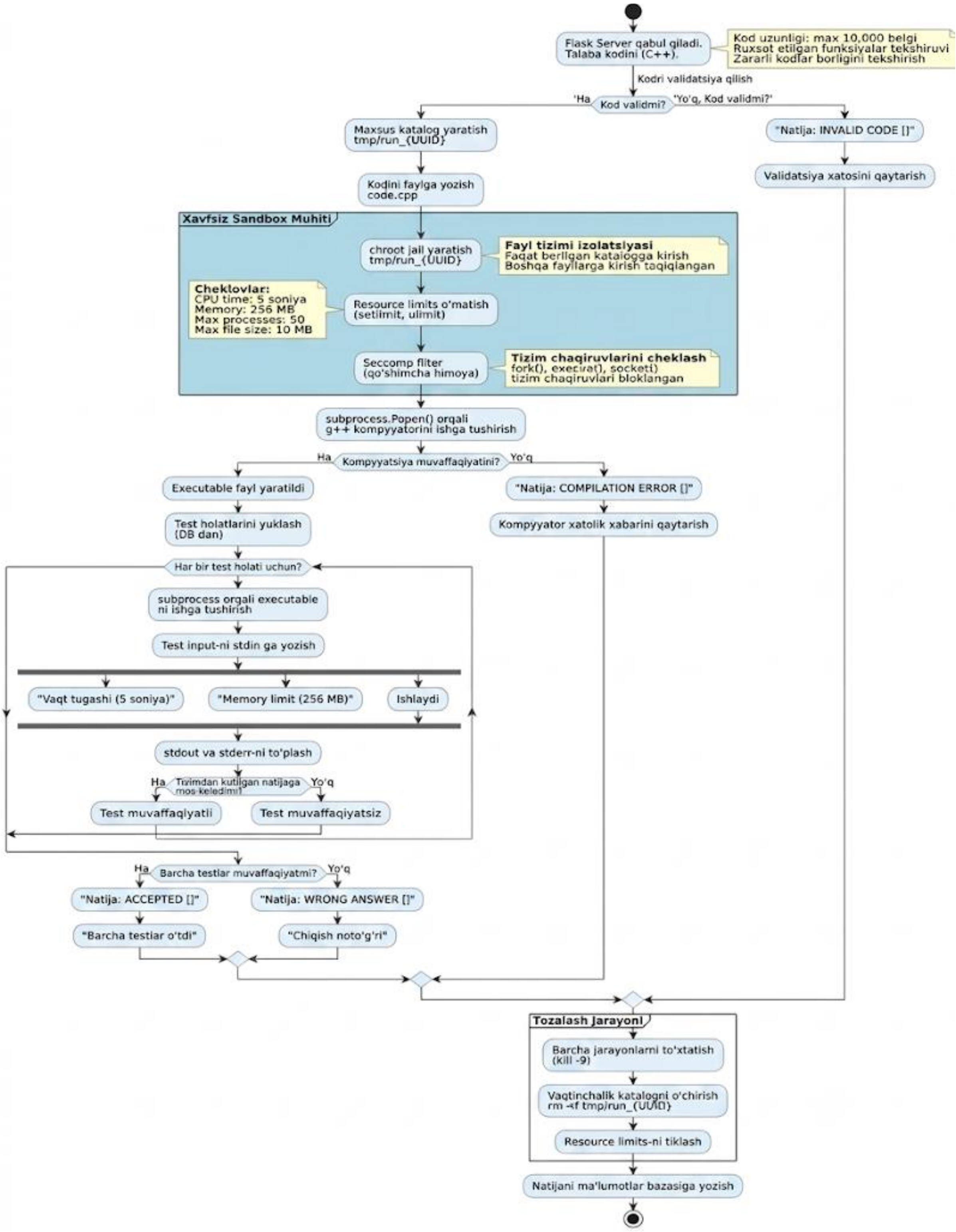

Figure 2: Process flow diagram in the system security module.

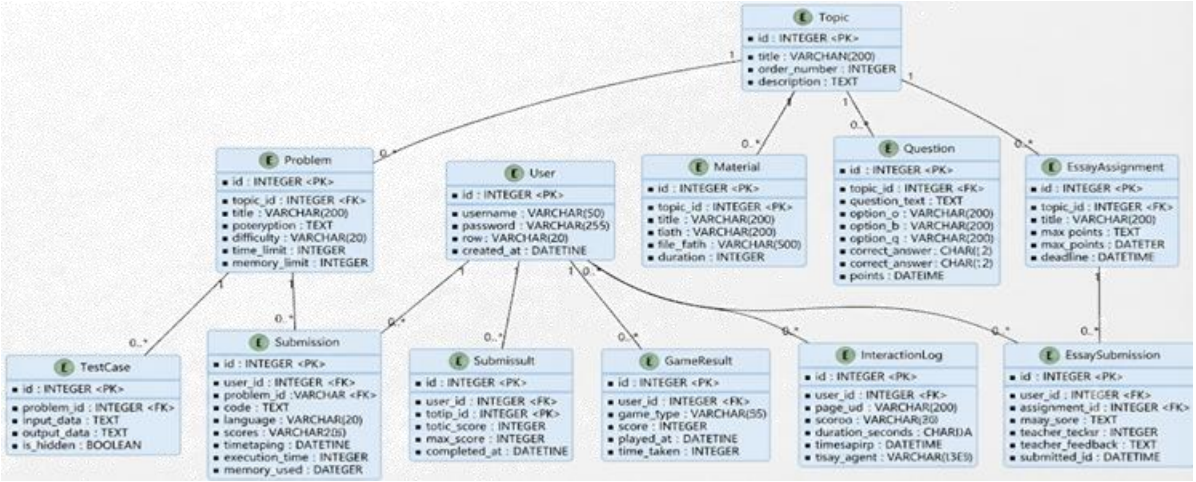

Figure 3: ER diagram of the database. The database schema provides a structured framework for linking learning activities with efficiency indicators.

4.1.1 Learning Mastery Indicator (P)

$$P = w_t \cdot T + w_c \cdot C + w_e \cdot E_s + w_g \cdot G. \quad (2)$$

The components of the learning mastery indicator (P) and their respective weighting coefficients (w_i) are defined as follows:

- T – standardized test score, representing theoretical knowledge ($w_t = 0.3$);
- C – coding task score, evaluated automatically by the Online Judge module ($w_c = 0.4$);
- E_s – essay or creative assignment score, reflecting qualitative understanding ($w_e = 0.2$);
- G – game-based task score, derived from gamified learning activities ($w_g = 0.1$).

4.1.2 Engagement Indicator (*A*)

$$A = \sum_{i=1}^{n} \gamma_i \cdot f\left(\frac{t_i}{t_i^{opt}}\right), \quad f(x) = \min(x, 1). \quad (3)$$

Where t_i denotes the actual time spent in the active window (monitored via heartbeat signals every 30 seconds), and t_i^{opt} represents the optimal time threshold predefined by the instructor for each specific task type.

4.2 Materials and Experimental Design

To evaluate the effectiveness of the proposed hybrid assessment methodology, a controlled quasi-experiment was conducted. The study involved first-year students of the Software Engineering program at the Samarkand branch of TUIT. A total of 82 students were selected and randomly assigned to a control group $(N_1 = 41)$ and an experimental group $(N_2 = 41)$.

The equivalence of the groups' initial knowledge was ensured by the higher education admission policy. That is, students were equally distributed across all academic groups based on their entrance exam scores. In addition, a pre-test conducted at the beginning of the semester confirmed that there were no statistically significant differences in prior programming knowledge between the control and experimental groups ($p > 0.05$).

The experimental design ensured that both groups followed the same syllabus and learning objectives. In both groups, online learning platforms were used alongside traditional classroom instruction to support independent learning. The control group studied using the Moodle/HEMIS LMS and relied on standard test- and assignment-based assessment. The experimental group used the authors' Hybrid LMS, which integrates an Online Judge module, gamification components, and activity monitoring mechanisms.

The experiment was carried out during the autumn semester of the 2025–2026 academic year (15 weeks) in the course PROG16MBK – Programming. Students were provided with electronic learning materials (.docx, .pdf, .pptx, video) as well as self-assessment materials, including tests, essays, practical tasks, and game-based activities. The learning materials were identical in content across both platforms.

The collected data were prepared for statistical analysis and processed using the methods described in the next section.

While the control group was assessed using P only, the Hybrid LMS automatically computed P, A, and E for the experimental cohort.

4.3 Ethics and Privacy

The data collection process during the quasi-experiment followed strict ethical guidelines. All participants were informed about the nature of the study, the activity monitoring mechanisms, and provided their informed consent before the experiment began. The monitoring system was designed to track only active window focus and interaction timestamps; it did not capture any personal screen content, private files, or external communications. To protect student privacy, all collected data were anonymized and used exclusively for academic research purposes. Data were stored on a secure server and will be retained only for the duration required by the institutional research policy.

5 EXPERIMENTAL RESULTS

The data collected during the experiment were processed using statistical analysis to evaluate the effectiveness of the proposed hybrid assessment model. For each group, the mean value and standard deviation of the learning mastery indicator (P) were calculated.

The statistical significance of differences between the control and experimental groups was examined using an independent samples Student's t-test [19]. The results indicate that the experimental group achieved a significantly higher mean learning mastery score ($M_1 = 84.56, SD_1 = 12.75$) compared to the control group ($M_2 = 63.00$, $SD_2 = 22.52$). The computed t value was 5.34 with $p < 0.001$, demonstrating a statistically significant difference. Furthermore, the calculated effect size (Cohen's $d = 1.18$) indicates a large practical significance according to the criteria established by Cohen [20], confirming the substantial impact of the Hybrid LMS on students' learning performance (Table 1).

Table 1: Descriptive statistics of P for control and experimental groups.

Group	N	Mean (P) [95% CI]	Mean (P)	SD
Experimental	41	84.56 [80.5, 88.6]	84.56	12.75
Control	41	63.00 [55.8, 70.2]	63.00	22.52

Figure 4 presents a comparison of the mean P values for both groups and clearly shows the superiority of the experimental group.

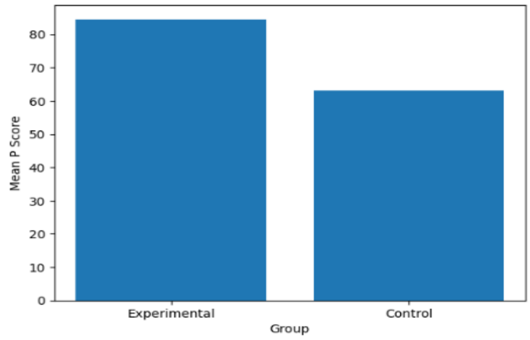

Figure 4: Mean learning mastery (P) by group. The experimental group significantly outperformed the control group.

With in the experimental group, Pearson correlation analysis was applied to examine the relationship between student engagement (A) and learning mastery (P) [21]. A strong positive correlation was found ($r = 0.75$, $p < 0.001$), indicating that higher engagement levels are associated with better academic achievement (Fig. 5).

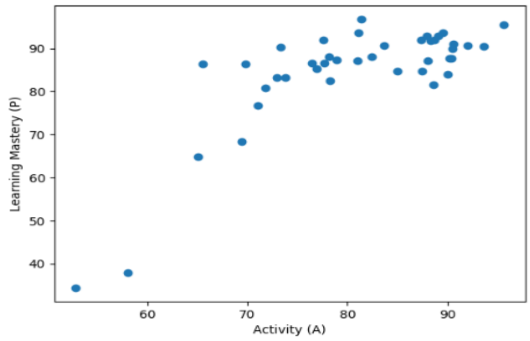

Figure 5: Relationship between Activity (A) and Learning Mastery (P). Higher engagement correlates with better academic achievement.

In addition, a very strong positive correlation was observed between student engagement (A) and the overall efficiency index (E) ($r = 0.89$, $p < 0.001$). This result demonstrates that engagement is a key contributing factor to student efficiency in the proposed hybrid model (Fig. 6).

Overall, the statistical and graphical results confirm that the Online Judge, gamification, and monitoring mechanisms integrated into the Hybrid

LMS positively influence students' academic efficiency by increasing their engagement.

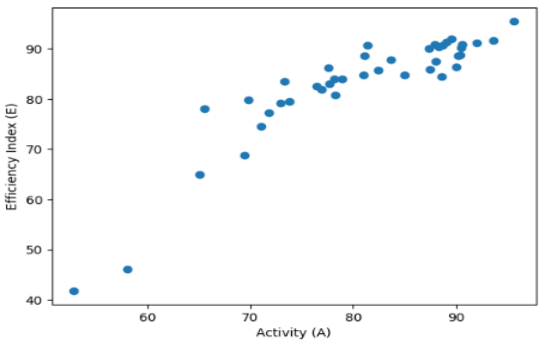

Figure 6: Relationship between Activity (A) and Efficiency Index (E). Engagement is a primary driver of overall student efficiency.

6 DISCUSSION

The results of this study demonstrate the effectiveness of the proposed hybrid assessment model in programming education. The student's t-test results confirmed that the experimental group achieved significantly higher learning mastery scores than the control group. This improvement can be attributed to the integration of the Online Judge and automated assessment mechanisms, which provide immediate and objective feedback.

These findings are consistent with previous studies. Ala-Mutka [3] and Ihantola et al. [4] reported that automated assessment systems improve students' academic performance. These findings align with the comprehensive review by Messer et al. [22], who highlighted that such tools not only reduce instructor workload but also provide essential formative feedback for novice programmers. In contrast to these works, the present study extends automated assessment by incorporating engagement monitoring, thereby achieving a more comprehensive evaluation approach.

The Pearson correlation analysis revealed a strong relationship between student engagement (A) and learning mastery (P). This result supports prior research indicating that gamification increases cognitive engagement and learning outcomes [6], [7]. Moreover, the very strong correlation between engagement (A) and the overall efficiency index (E) suggests that student participation plays a central role in determining learning effectiveness.

An important contribution of this study is the integration of process-oriented indicators into the assessment framework. While learning analytics dashboards [9] and cognitive diagnostic models [18] provide valuable insights into student behavior and knowledge states, they are often not directly linked to summative assessment. The proposed hybrid model addresses this gap by combining engagement and learning mastery within a unified index.

Nevertheless, this study has certain limitations. First, the experiment was conducted within a single course at one institution. Second, engagement data were unavailable for the control group, preventing direct comparison of A and E between groups. Future studies should evaluate the proposed model across multiple courses and institutions.

7 CONCLUSIONS

This paper proposed a Hybrid LMS platform and a hybrid assessment model for programming education. The proposed approach integrates learning mastery (P) and student engagement (A) into a unified efficiency index (E), extending traditional assessment methods by considering not only final learning outcomes but also the learning process itself.

The results of the controlled quasi-experiment demonstrate that students using the Hybrid LMS achieved significantly higher academic performance compared with those using a traditional LMS. Statistical analysis revealed strong relationships between student engagement, learning mastery, and overall efficiency, confirming the important role of engagement in achieving successful learning outcomes.

The study demonstrates the practical effectiveness of combining automated assessment, gamification elements, and activity monitoring within an integrated learning platform. The proposed hybrid assessment model provides a flexible framework that can support data-driven evaluation and can be adapted for other practice-oriented educational disciplines.

8 FUTURE WORK

Future research will focus on validating the proposed model across multiple courses, institutions, and diverse student populations to evaluate its generalizability. Further studies should investigate the optimization of weighting coefficients within the efficiency index and examine their influence on assessment reliability.

In addition, the integration of artificial intelligence techniques, including adaptive learning algorithms and personalized recommendation systems, will be explored to provide individualized learning paths, real-time feedback, and more effective support for student development.

REFERENCES

[1] J. O. M. Chaves, A. F. Castro, R. W. Lima, M. V. A. Lima, and K. H. A. Ferreira, "Uma ferramenta para auxiliar na elaboração, submissão e correção de atividades práticas em disciplinas de programação," in Anais dos Workshops do Congresso Brasileiro de Informática na Educação, vol. 1, no. 1, 2012.

[2] H. Jeong and C. Jeon, "Can coding education go completely online? Time, work, and relationship in online courses," East Asian Sci., Technol. Soc., Int. J., vol. 18, no. 1, pp. 4-21, 2024, [Online]. Available: https://doi.org/10.1007/s12280-023-00607-5.

[3] K. M. Ala-Mutka, "A survey of automated assessment approaches for programming assignments," Comput. Sci. Educ., vol. 15, no. 2, pp. 83-102, 2005, [Online]. Available: https://doi.org/10.1080/08993400500150747.

[4] P. Ihantola, T. Ahoniemi, V. Karavirta, and O. Seppälä, "Review of recent systems for automatic assessment of programming assignments," in Proc. 10th Koli Calling Int. Conf. Comput. Educ. Res., 2010, pp. 86-93, [Online]. Available: https://doi.org/10.1145/1930464.1930480.

[5] A. Kurnia, A. Lima, and B. Cheang, "Online judge," Comput. Educ., vol. 36, no. 3, pp. 299-315, 2001, [Online]. Available: https://doi.org/10.1016/S0360-1315(00)00055-8.

[6] M. Ortiz-Rojas, K. Chiluiza, and M. Valcke, "Gamification through leaderboards: An empirical study in engineering education," Comput. Appl. Eng. Educ., vol. 27, no. 4, pp. 1-12, 2019, [Online]. Available: https://doi.org/10.1002/cae.22101.

[7] M. B. Ibáñez, A. Di-Serio, and C. Delgado-Kloos, "Gamification for engaging computer science students in learning activities: A case study," IEEE Trans. Learn. Technol., vol. 7, no. 3, pp. 291-301, 2014, [Online]. Available: https://doi.org/10.1109/TLT.2014.2322354.

[8] S. A. Triantafyllou, T. Sapounidis, and C. Georgiadis, "Gamification in education and training: A literature review," Int. Rev. Educ., vol. 71, pp. 483-517, 2025, [Online]. Available: https://doi.org/10.1007/s11159-024-10111-8.

[9] R. Bodily and K. Verbert, "Review of research on student-facing learning analytics dashboards and educational recommender systems," IEEE Trans. Learn. Technol., vol. 10, no. 4, pp. 405-418, 2017, [Online]. Available: https://doi.org/10.1109/TLT.2017.2783679.

[10] Q. Hao et al., "Towards understanding the effective design of automated formative feedback for programming assignments," Comput. Sci. Educ., vol. 31, no. 3, pp. 319-343, 2021, [Online]. Available: https://doi.org/10.1080/08993408.2021.1924567.

[11] R. Parvathy, M. G. Thushara, and J. M. Kannimoola, "Automated code assessment and feedback: A comprehensive model for improved programming education," IEEE Access, vol. 13, pp. 56658-56675, 2025, [Online]. Available: https://doi.org/10.1109/ACCESS.2025.2999507.

[12] S. Ghazinoory and M. Afshari-Mofrad, "Ranking different factors which affect e-learning outcomes," Int. J. Comput. Theory Eng., vol. 4, no. 2, pp. 237-240, 2012.

[13] M. Attia, M. A. Abdel-Fattah, and S. A. El-Gaber, "A model for assessing and enhancing efficiency of e-learning systems," Int. J. Comput. Appl., vol. 156, no. 6, pp. 12-21, 2016.

[14] L. Jaramillo-Mediavilla, A. Basantes-Andrade, M. Cabezas-González, and S. Casillas-Martín, "Impact of gamification on motivation and academic performance: A systematic review," Educ. Sci., vol. 14, no. 6, Art. no. 639, 2024, [Online]. Available: https://doi.org/10.3390/educsci14060639.

[15] G. Siemens and R. S. J. d. Baker, "Learning analytics and educational data mining: Towards communication and collaboration," in Proc. 2nd Int. Conf. Learn. Analytics Knowl., 2012, pp. 252-254.

[16] I. M. Boynazarov, U. Tudevdagva, R. A. Raxmonova, and G. T. Safarova, "The evaluation result based on the SURE model: Case of Samarkand Branch of TUIT," in Proc. Int. Symp. Comput. Sci. Educ. Technol., 2024.

[17] I. M. Boynazarov and F. K. Ochilova, "Hybrid model of evaluating student activity in e-learning systems," in Proc. Int. Conf. AI Comput. Technol.: Practice, Experience, Problems and Perspectives, Denau, Uzbekistan, 2025.

[18] H. Cao and L. Tan, "Student education assessment model based on programming cognitive diagnosis," in Proc. 2023 Int. Conf. Artif. Intell., Syst. and Netw. Secur. (AISNS '23), New York, NY, USA, 2023, pp. 1-6, [Online]. Available: https://doi.org/10.1109/AISNS57419.2023.10319233.

[19] D. C. Montgomery and G. C. Runger, Applied Statistics and Probability for Engineers, 7th ed. Hoboken, NJ, USA: Wiley, 2018, p. 710.

[20] J. Cohen, Statistical Power Analysis for the Behavioral Sciences, 2nd ed. Hillsdale, NJ, USA: Lawrence Erlbaum Associates, 1988.

[21] J. Cohen, P. Cohen, S. G. West, and L. S. Aiken, Applied Multiple Regression/Correlation Analysis for the Behavioral Sciences, 3rd ed. New York, NY, USA: Routledge, 2003, [Online]. Available: https://doi.org/10.4324/9780203774441.

[22] M. Messer, N. C. C. Brown, M. Kölling, and M. Shi, "Automated grading and feedback tools for programming education: A systematic review," ACM Trans. Comput. Educ., vol. 24, no. 1, pp. 1-43, 2023, [Online]. Available: https://doi.org/10.1145/3636515.