

PAPER

A Mobile-Enabled Adaptive Gamification Framework for Programming Education

Ibrahim Alkore Alshalabi,
Tawfiq Alrawashdeh, Anas
AbuKaraki, Malek Zakarya
Alksasbeh(✉)

Al-Hussein Bin Talal
University, Ma'an, Jordan

malksasbeh@ahu.edu.jo

ABSTRACT

While traditional teaching and learning methods in education have turned to be more active and engaging with the help of technology, this study proposes a Holistic Gamification Learning Adventure (HGLA) framework, which integrates gamification principles, multidimensional progression, and narrative-driven experiences for programming education. Different from most other gamification models, adaptive pathways and collaboration in the HGLA framework ensure novelty in accommodating diverse learning styles and proficiency levels of both novice and advanced programmers. It's a framework that uses responsive design to ensure access across devices, creating ubiquitous learning environments. The embedment of gamified missions and narrative-driven challenges shall make students active participants instead of passive observers, while the motivation, engagement, and retention concerning programming concepts shall be developed. The multi-dimensional progression within the framework caters to individual learning paces, ensuring an individualized educational experience for deeper conceptual understanding and long-term retention. The present study shows how the HGLA framework has transformed programming education into an adaptive, inclusive, and impactful learning journey—a beacon of change in innovative teaching methodologies.

KEYWORDS

gamification, multidimensional progression, narrative-driven experiences, adaptive learning, programming education

1 INTRODUCTION

Nowadays, education has transformed due to technologies and innovative approaches; thus, it is taught and learned differently compared to several decades ago [1, 2]. One of the crucial problems educators and researchers currently face involves bridging the confidence gap in learners while mastering programming skills. Gamification and multidimensional progression are two of the most promising strategies for solving this problem [3]. Programming education, in particular,

Alshalabi, I.A., Alrawashdeh, T., AbuKaraki, A., Alksasbeh, M.Z. (2025). A Mobile-Enabled Adaptive Gamification Framework for Programming Education. *International Journal of Interactive Mobile Technologies (ijim)*, 19(8), pp. 42–69. <https://doi.org/10.3991/ijim.v19i08.52823>

Article submitted 2024-10-12. Revision uploaded 2025-01-06. Final acceptance 2025-01-06.

© 2025 by the authors of this article. Published under CC-BY.

requires creative insights to bridge obstacles brought about by traditional pedagogies and the nature of programming concepts themselves [4].

Traditional approaches, often characterized by lecturing and conventional problem-solving using code, lack any depth in the effective engagement of learners, resulting in adverse issues related to poor retention and shallow skill development, as outlined in [5, 6]. Such difficulties worsen because of diverse individual ways of learning and thinking among the trainees, as referred to in [7, 8]. All this requires breaking barriers through innovative mechanisms allowing active participation and individual paths to understanding. Gamifications, including the addition of elements such as point-based awards, badges, or story-driven narratives, offers newer, perhaps more playful ways, to explore educational materials [9, 10]. It facilitates learning in an active fashion for students by reimagining lessons as interactive missions and challenges [11]. In this regard, multi-dimensional progress further enhances learners' opportunities for following a learning path unique to their learning style and/or pace and toward collaboration and learning with their peers [12, 13]. Recent researches [14, 15] have underlined the gamification potential enabled by mobile to create engagement, motivation, and adaptability in education. The HGLA framework integrates a mobile-responsive design based on these principles, with the assurance of accessibility and smooth user experiences across devices.

Our paper explores the transforming power of gamification, multi-dimensional progression, and narrative-driven experiences within programming education to innovatively address some persistent challenges and change the way students learn through personalization and engagement. By transforming lessons of programming into immersive adaptive journeys, the framework seeks to provide a personalized, engaging path to mastery, using mobile technologies to enable inclusive and effective learning experiences.

2 LITERATURE REVIEWS

The convergence of technology and pedagogy has brought about profound changes in education, reshaping how learners interact with content and improving student learning skills. This literature review focuses on three key concepts in this adapting aspect: gamification, multidimensional progression, and narrative-driven experiences. These concepts play a significant role in transforming pedagogical practices, particularly in the field of programming education.

This section presents the various models and frameworks developed to enhance learning through gamification. We will reflect on these in relation to our HGLA framework, considering dimensions such as adaptive learning, multidimensional progression, narrative-driven experiences, engagement metrics, support for problem-solving, performance progress measurement, collaborative elements, and knowledge retention.

Among all the active areas of research, the integration of gamification in educational frameworks is one, with an ever-growing number of models and approaches reviewed in our literature with the aim of amplifying student engagement, motivation, and learning outcomes. Gamification, or the use of approaches and game design elements in non-game contexts is particularly promising in terms of rendering learning experiences more engaging and effective. However, despite all the developments in this area, most of the currently available models have certain drawbacks concerning adaptability, narrative-driven experiences, and personalized learning pathways. This has motivated the further development of new frameworks, among which the holistic gamification learning adventure-analog framework, presented in this

section, focuses on uniting the best aspects of the previous models while simultaneously addressing the limitations imposed by their predecessors, as shown in Table 1.

Table 1. Comparative analysis: HGLA vs. existing frameworks

Framework	Adaptive Learning	Multidimensional Progression	Narrative-Driven Experience	Engagement Metrics	Problem-Solving Support	Performance Progress Measurement	Collaborative Elements	Knowledge Retention
Game Object Model (GOM) [16]	No	No	No	Moderate	High	Moderate	Low	Moderate
Experiential Gaming Model (EGM) [17]	No	High	No	Moderate	High	Moderate	Moderate	Moderate
Four-Dimensional Framework (4DF) [18]	No	Moderate	High	Moderate	Moderate	High	Low	High
Serious Games Conceptual Framework (SGCF) [19]	No	Low	No	High	High	High	Low	Moderate
Triadic Game Design (TGD) [20]	No	Low	High	Moderate	High	Moderate	Low	Moderate
Six Facets of Serious Games Design [21]	No	Moderate	High	High	High	Moderate	High	Moderate
Learning Mechanics-Game Mechanics Mapping [22]	No	High	No	High	High	Low	Low	Moderate
Gameful Design Heuristics (GDH) [23]	High	Low	No	High	Moderate	Moderate	Moderate	High
Octalysis Framework [24]	High	Low	No	High	Moderate	Low	Moderate	High
Self-Determination Theory-Based Gamification (SDT) [25]	Moderate	Low	Moderate	High	High	Moderate	Low	Moderate
Playful Learning Design Framework [26]	Low	Moderate	Moderate	Moderate	Moderate	Low	Moderate	Moderate
Adaptive Gamification Model [27]	High	Moderate	Low	High	Moderate	High	Low	High
HGLA Framework	High	High	High	High	High	High	High	High

An influential early model in this direction is the Game Object Model developed by [16]. While this model is aimed at integrating game objects, rules, and scenarios within the design of educational games, it has focused almost exclusively on engagement and problem-solving. Although this model has driven the development of some of the most engaging learning environments, it does not contain adaptive learning capabilities that will help in tailoring learning to individual learners' needs. Additionally, it lacks the provision of experiences based on narratives, which reduces its full immersion ability in educational content for learners across diverse learning styles and age groups. The Experiential Gaming Model [17] is based on notions of learning through experiences, with a strong focus on problem-solving. The model caters to multidimensional progress, and for this reason, it is well set for a broad range of learners. However, much like the Game Object Model, it does not fully integrate adaptive learning or narrative-driven experiences, which are key aspects for personalized and engaging learning experiences desired by participants.

Another influential contributor is the Four-Dimensional Framework [18]. This underscores the context, learner profile, and pedagogical concerns related to game-based learning. Though it allows support for narrative-driven experiences and provides a framework that supports performance progress measurement, it lacks

adaptive learning and engagement metrics. These elements constitute the core of dynamic personalized learning environments, continuously adapting to the evolving needs and capabilities of each learner.

Another approach is the Serious Games Conceptual Framework [19], which also integrates engagement metrics and problem-solving support into an educational game. While this framework provides tools to measure the progress of performance, adaptive learning and the narrative-driven component to fully address individual learning needs and preferences are not available. Thus, the enhancement of certain aspects of engagement and problem solving will be developed, while this framework does not answer how a personalized learning experience will be comprehensively ensured.

A Triadic Game Design was proposed by [20] to balance reality, meaning, and play within educational games. This model has adopted narrative-driven experiences with problem-solving but embeds neither adaptive learning nor engagement metrics. Without them, the Triadic Game Design could offer meaningful and contextually relevant learning experiences but may fall short in adapting such experiences to the needs of individual learners or keeping their interest over time.

The Six Facets of Serious Games Design [21] expanded earlier models by putting a tight focus on six facets: motivation, integration, learning, immersion, emotion, and social interaction. The model supports narrative-driven experience with engagement metrics, thus making it a more holistic approach to gamification for education. However, even this is deficient in an adaptive learning system for compiling personalized feedback that could seamlessly escalate the process into individual learning trajectories for learners to traverse at their own pace.

Another technique, Learning Mechanics to Game Mechanics Mapping proposed by [22], determines which game mechanics are the best to engage learners, effectively identifies learning mechanics, and rediscover how to use them in an educational context. The scaled model supports multidimensional progression and engagement metrics but excludes narrative-driven experiences or adaptive learning. Therefore, it works as a basic structure for intermingling educational content with game elements, but it might fail to hook learners or slowly adjust to their personal learning demands. On the other hand, Gameful Design Heuristics [23] was aimed at contexts where games were not participants but rather modulators of human behavior in non-game design. This approach benefits from adaptive learning and enhanced engagement metrics, yet misses story-driven experiences and multidimensional progression. However, the experiences that we build with it will only be engaging and motivational and not truly immersive or sufficiently connective if all components are missing. The Octalysis Framework [24] is the science of human motivation gamification that emphasizes eight core drives. Although adaptive learning and engagement metrics experience are facilitated, they fail to foster the layering of three-dimensional progression and narrative-driven experiences that are crucial to holistic immersion. This limitation means that the Octalysis Framework is likely to motivate learners, but it may not provide as much granularity or personalization in the pursuit of a transformative educational experience. Other models, such as Self-Determination Theory-Based Gamification [25], Playful Learning Design Framework [26], and Adaptive Gamification Model [27], offer unique perspectives and tactics within educational gamification. Although they are technologically similar, there is still a great degree of personalization, so much so that the result is most definitely story-based and takes into account the needs of individual learners.

The literature of gamification, multidimensional progression, and narrative-driven experiences typifies potential profiles for education. More broadly, the integration of these ideas into the HGLA framework provides a convincing synthesis

that circumvents many challenges endemic to typical programming education. This approach transports learners through gamified experiences and then iteratively nudges them on an educational journey that is personalized to their own learning paths, revealing not only a framework but also a transformative process as it gives a new meaning for the learners to begin their quest for meaningful educational adventures.

- **Gamification in Educational Contexts:** Gamification, defined as the incorporation of game design elements into non-game environments, has gained significant traction in educational research [28, 29]. By taking advantage of innate human tendencies for challenge, competition, and achievement, gamification improves student engagement and motivation [30, 31]. Elements such as points, badges, leaderboards, and narratives have been shown to positively impact student engagement and knowledge retention in educational environments [32, 33].

Research has underscored that gamification strategies cultivate a sense of autonomy and competence, both essential components of intrinsic motivation [34]. The narrative-driven experiences submerge in gamified learning environments, providing learners with a clear sense of purpose, supporting profound understanding of the course content, and contextualizing learning objectives [35, 36, 37].

- **Contextualizing Gamification in Jordanian Programming Education:** Gamification has proven effective in Western educational contexts, where infrastructure, digital literacy, and teaching methodologies are often conducive to its adoption [38]. However, the Jordanian educational landscape presents distinct challenges that demand context-sensitive strategies. One key issue is the uneven levels of digital literacy and access to technological resources across student populations, which can significantly influence the adoption and success of gamified learning environments [39, 40]. Additionally, Jordan's cultural and pedagogical norms emphasize collaborative and community-oriented approaches, necessitating the adaptation of gamified mechanics to reflect these values [40, 41].

For example, while individual achievement may drive motivation in Western models, Jordanian students might respond more positively to gamification elements that reward group achievements or emphasize collective problem solving. Addressing these nuances, future iterations of the HGLA framework will incorporate culturally aligned game mechanics, such as team-based challenges and localized narratives, to increase engagement [42]. Furthermore, ensuring equitable access to digital tools through institutional support and infrastructure development will be pivotal in overcoming technological barriers.

By tailoring the HGLA framework to the Jordanian context, this study not only enhances its applicability but also provides a blueprint for implementing gamified learning systems in regions with similar educational and cultural dynamics.

- **Multidimensional Progression and Adaptive Learning:** Multidimensional progression is an approach that adapts learning paths to individual needs and addresses the limitations of traditional (one-size-fits-all) education [43, 44]. Recognizing the diverse backgrounds, interests, and learning styles of learners, this approach offers both adaptive content and challenges. It accommodates varying proficiency levels while maintaining an optimal level of challenge for each learner [45]. The associated personalized learning pathways have been linked to heightened learner motivation, improved knowledge retention, and enhanced critical thinking [46, 47]. Learners are more likely to engage deeply with content aligned with their existing knowledge and interests, supporting a sense of ownership over their educational journey and promoting self-directed learning skills [48].
- **Applications in Programming Education:** Gamification and multidimensional progression have been most effectively applied to programming education, with

an emphasis on challenges to engagement, knowledge retention, and learning pace. The idea of earning points when finishing challenging codes and advancing to levels has been useful for increasing learners' interest levels and motivation to learn programming [49]. The concept of multidimensional progression easily fits the complexity of how programming concepts develop graphically. Tailoring learning pathways to accommodate both beginners and advanced learners has resulted in increased mastery and deeper understanding of coding principles [50].

The HGLA framework is an advanced model that incorporates the best features of previous models as well as a workable approach to their shortcomings. Engaging in influential learning with the HGLA framework ensures that learning is competent, challenging, and customized to individual needs in real-time learning.

Another important gap is the lack of standardized m-learning frameworks that are inclusive of gamification principles while also considering equal access to technology. According to [51], a need exists for those kinds of frameworks that may correlate the gamification mechanics with unique affordances of mobile technologies, such as portability and connectivity, taking into consideration various challenges relating to digital literacy and inequalities in infrastructure.

The HGLA is a novel perspective for gamifying a programming course by combining elements from existing models and frameworks that are worth keeping and introducing new features aimed at solving some of their deficiencies. Offering students the content and services they need to begin an enriching learning experience or continue with it within a flexible framework so that they can succeed. As such, the HGLA model presents an important development with regard to learning in terms of a new way of looking at gamification from the perspective of education design and development, dealing with both holistic and effective aspects of learning design.

3 METHODOLOGY

Our methodology was driven by the overarching goal of revolutionizing programming education by synthesizing the principles of gamification, multidimensional progression, and narrative-driven experiences. The framework's foundation was built upon an extensive review of existing literature on gamification, learning games, and adaptive learning. This preliminary research helped to identify gaps in conventional programming education methods and paved the way for the development of an integrated approach that addresses these deficiencies.

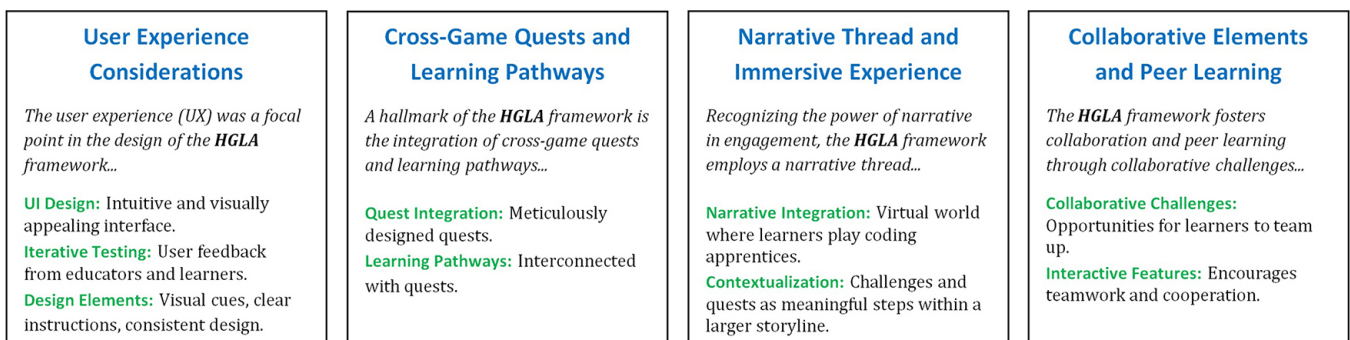


Fig. 1. Guiding design principles

3.1 Design principles guiding the framework

As shown in Figure 1, the implementation of the HGLA framework is underpinned by a set of carefully curated design principles aimed at creating a seamless, engaging, and effective learning experience for programming education. These principles were drawn from pedagogical best practices, gamification theories, and insights from user-centered design.

User experience considerations. Designing for user experience (UX) is a top priority in the design of the HGLA framework. We ensured that the user interface was intuitive and visually appealing to improve the navigation and engagement. We iteratively tested the framework by using educators and learners. The interface was similarly refined through an iterative process to maximize engagement and minimize cognitive load [52, 53]. To provide guidance and continuity, visual cues, explicit instructions, and design elements were placed for similar activities across the framework to help learners.

Cross-game missions and learning pathways. The HGLA framework features cross-game missions and learning pathways. All missions were designed to be applied to programming code in the sense that students had to use programming concepts, information, and programming code that they learned from different programming games to solve problems [54]. This leads to a better and broader understanding of how things are related in programming language, is useful for making them think critically, and also helps in problem-solving techniques. In this regard, the learning pathways and mission enable learners to navigate their learning around programming primitives in a self-paced and style manner that allows each person's pathway of learning to be uniquely personalized [55].

Narrative thread and immersive experience. With engagement performance driven by the narrative, bringing structure to this design approach, HGLA architecture uses a sort of narrative thread that ties back all the different kinds of learning games and challenges provided. The learner is placed in a virtual environment with the coding apprentice as part of an interactive narrative story. The narrative thread places challenges and missions in contexts that transition these activities from discrete exercises into stages along a story in a dynamic narrative, converting them from one-off exercises into steps in a larger storyline, giving quests specifically designed not only context but also meaning [56]. This is a fun way to get learners excited and show how programming concepts are grounded in something that they can understand and relate to.

Collaborative elements and peer learning. The HGLA framework supports collaboration and peer learning through competitive challenges and interactive mechanisms. Collaboration: Working on difficult coding puzzles together, learners can form teams to solve more complex side-by-side challenges. This collaborative experience helps build the community to share, learn from one another, and solve problems with code tricks [57, 58].

Combining these design principles within the HGLA framework results in a distinctive, immersive learning ecosystem. The combination of user experience, cross-game missions, and narrative thread not only overcomes the flaws in traditional programming courses but also changes the image learners have of programming concepts.

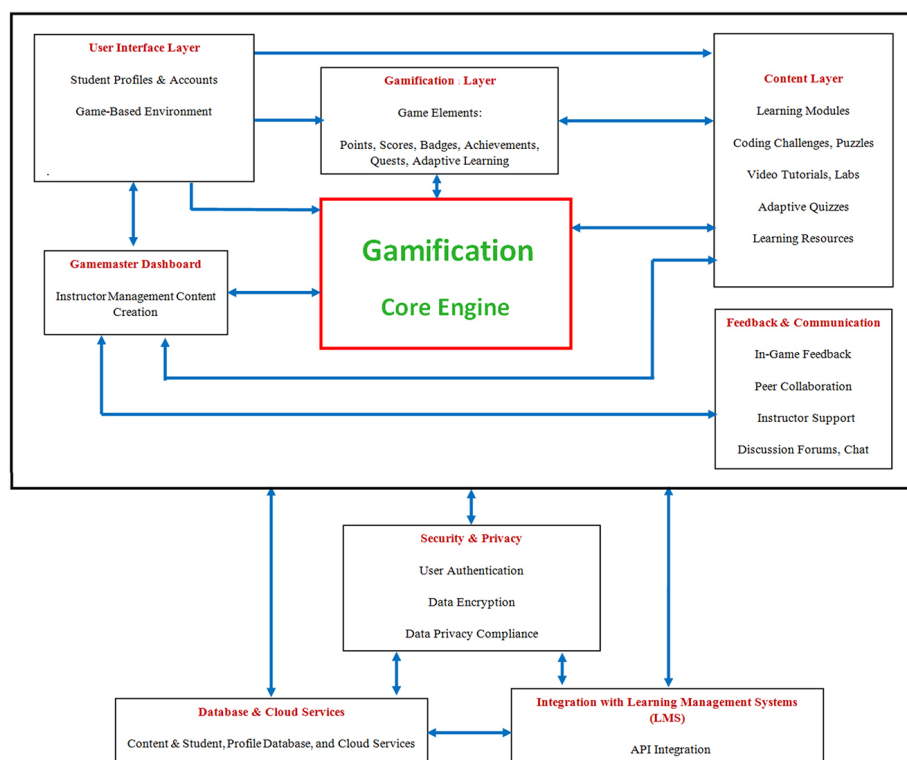


Fig. 2. Holistic Gamification Learning Adventure (HGLA) framework

3.2 Holistic Gamification Learning Adventure (HGLA) framework

As shown in Figure 2, the foundation of the framework was built on the design principles shown in Figure 1 and an extensive review of the existing literature on gamification theories, user-centered design ideas, learning games, adaptive learning, and best practices in education.

User Interface Layer. The User Interface Layer, as shown in Figure 2, is the entry point for students to the gamified learning platform, offering a visually appealing environment for the students to interact with the learning platform, with a simple and nice interface. Student Profiles and Accounts as part of this layer keep the information, progress, and accomplishments of each student and metadata about the users, enabling the creation, editing, and safekeeping of usernames and passwords. Such systems of control also include authentication of users to prevent unauthorized users from using the platform. Data collection looks for user interaction data in terms of how well the user interacts with the system, performs on it, and what preferences the student has for the purposes of personalizing the learning path and learning style.

Content Layer. The Content Layer is important in the delivery of educational content within a gamified learning platform, offering a wide variety of materials and challenges. It works as an active intermediate access point to learning modules, challenges, and resources and utilizes Recommendation Engines for personalized and optimized learning experiences. It comprises structured educational content, practical application challenges, supplementary materials, and a student progress tracker.

Gamification layer. The gamification layer offers huge value to the learning experience of the gamified learning platform by seamlessly integrating game elements, challenges, and missions into the educational content. It interacts with other modules, such as the Content Layer, Recommendation Engine, and Data Analytics & Personalization of the Core Engine, to ensure a consistent and engaging gamified

environment. User data and feedback were analyzed continuously to improve and optimize the gamification elements.

Core Engine. The Core Engine as shown in Figure 3, consists of three primary components: 1) Data Analytics and Personalization, 2) the Recommendation Engine, and 3) Multidimensional Progression Principles.

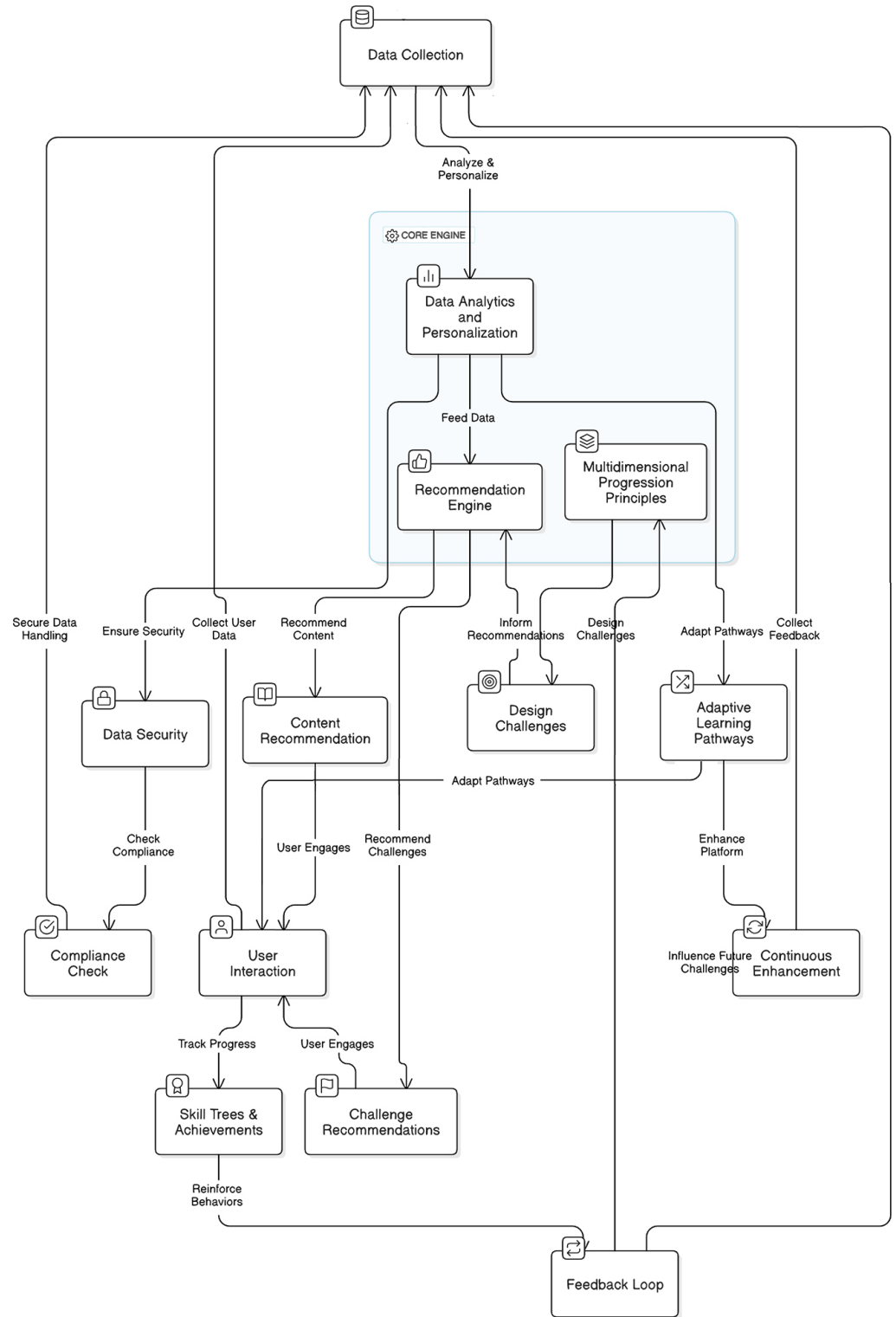


Fig. 3. Gamification core engine of HGLA

Data Analytics and Personalization. The Data Analytics and Personalization component forms the Core Engine for improving user learning within the gamified learning platform by capturing, analyzing, and using data. Integrated layers such as the Recommendation Engine, Gamification Layer, and Content Layer contribute to personalized content recommendations, adaptive learning pathways, and the continuous enhancement cycle of the platform. Considerable importance is attached to data security and compliance with privacy for the ethical handling of users' information. Collecting user data includes personalization by feeding the data collected into the Recommendation Engine, which then makes personalized recommendations of the contents. Continuous data collection adapts to changes in user behavior, preference, and learning patterns over time; hence, the platform is dynamically aligned with student needs.

Recommendation Engine. The Recommendation Engine is a crucial part of our framework, which would make this learning journey for each student unique within the gamified learning platform through recommendations of different content based on the profile, preference, and performance of the learner. Because it was integrated with both the Content Layer and the Gamification Layer, it was designed to make the experience cohesive and engaging, adding value to effectiveness.

Multidimensional Progression Principles. The Multidimensional Progression Principle is one of the core parts of the framework for forming game-based learning, as they relate to both the Gamification Layer and Recommendation Engine in driving the design of challenges, missions, and learning modules. Such principles lead to diversity and personalization within the educational process and ensure that the learning experience caters to various competencies and preferences of students as individuals. The Recommendation Engine is informed by Multidimensional Progression Principles, which provide insights into multiple ways of progressing through learning paths and challenges. They recommended challenges reflecting the principles of multidimensional progression that offer a range of tasks with gradually increasing complexity. Skill trees and achievements integrated with the Gamification Layer serve as motivational elements that create a feedback loop, reinforce positive learning behaviors, recognize user accomplishments, and influence future challenges.

Gamemaster Dashboard. The Gamemaster Dashboard is a comprehensive tool for teachers to manage content and see how students are doing, allowing them to help in real time. It provides a unified view of the learning environment by integrating components such as the Data Analytics engine, personalization engine, Recommendation Engine, and Content Layer, enabling instructors to maximize student learning experience. On the other hand, the Gamemaster Dashboard allows instructors to design educational content and challenges (Instructor Management and Content Creation) that can control the Recommendation Engine and Content Layer. Instructors can adapt learning modules, challenges, and resources to specific learning objectives and curriculum needs.

Feedback & Communication. In our gamified learning framework, the most salient component is the Feedback & Communication component, which allows peer collaboration, teacher support, and communication through discussions and chat forums. This shows a way to improve the learning experience because help is provided in real time and at a collaborative level while generating valuable data for the Core Engine's Data Analytics & Personalization part.

Database and cloud services. Databases and cloud services are the backbone of data integrity, availability, and scalability. It interacts with the Content Layer, Recommendation Engine, Gamemaster Dashboard, and Core Engine Data Analytics & Personalization. Furthermore, this allows for a personalized experience across the

platform, where user profiling, preferences, learning history, and achievements are stored in real time based on the user's interactions and progress.

Cloud services that prime platforms for scalability, reliability, and accessibility support an ever-growing user base and interactions across locations and devices. The protection of data through encryption, access controls, and privacy standards for mitigating risk with the implementation of disaster recovery processes.

Integration with learning management systems. The integration with learning management systems (LMS) via API within the system architecture ensures that everything goes right with the learning experience data being kept in sync, content is the same, and transitions between the gamified platform and external LMS go smoothly. This provides two-way communication whereby user progress and achievements can be exchanged in real time, keeping courses and learning modules aligned across platforms. This integration facilitates cross-platform accessibility, enabling users to seamlessly access content from both systems.

Security and privacy. Security and privacy within the system architecture provide a safe learning environment within the gamified platform by covering user authentication, data encryption, privacy compliance, and security measures. It interoperates with both the Gamemaster Dashboard and learning management systems.

User authentication involves secure processes for logging, multifactor authentication, and identity verification. Role-based access control manages various levels of access for students, teachers, and administrators through secure account recovery systems. Security updates are aimed at preventing potential threats. Federated authentication makes it easy to seamlessly log into all the integrated systems. This would ensure that data encryption and privacy compliance are of the utmost importance, whereby stringent protocols are in place to protect user information during both transmission and storage. Privacy standards shall have been met, whereby the assurance of safety for user data is within the system. Second, privacy policies should be transparent; regular privacy audits should be carried out with data anonymization techniques. The system should also ensure the consent of users and a policy on data deletion.

3.3 Gamification Core Engine Algorithms and programming course

Student progress analytics. In a programming course, the Data Analytics & Personalization of Core Engine uses advanced programming skill metrics based on features of complexity, efficiency, and correctness that could be extracted from code. Core Recommendation Engine suggests advanced coding challenges based on students' mastery levels, supporting a deeper understanding of programming concepts as shown in Algorithm 1.

Algorithm 1: Student Progress Analytics Implementation

```

Step 1. Analyzing programming progress
Step 2. Generating a personalized analytics report
Step 3. If the average mastery is greater than or equal to Threshold Mastery Average Value GOTO
Step 4 ELSE GOTO Step 5
Step 4. Acknowledging high mastery, GOTO Step 6
Step 5. Handling cases where the average mastery is less than Threshold Mastery Average Value.
    {
        • Provide tips for improvement.
        • Suggest additional learning resources.
        • Encourage practice and continuous learning.
    }
Step 6. End

```

Student challenges. The Core Recommendation Engine, adjusted to a programming environment, can provide intellectual suggestions related to challenges that match the user's selected programming language or focus area. The multidimensional progress within Core Engine justifies a varied set of challenges, whereas one encourages students to explore many different dimensions of programming, as shown in Algorithm 2.

Algorithm 2: Student Challenges Implementation

```

Step 1. Getting programming challenges for the student
Step 2. Assessing smart challenge completion
Step 3. If the number of completed challenges is greater than or equal to Threshold completed challenges Value GOTO Step 4 ELSE GOTO Step 5
Step 4. Unlocking the 'Code Master' achievement, GOTO Step 6
Step 5. Handling cases where the number of completed challenges is less than Threshold Completed Challenges Value.
    {
        • Encourage student to take on more challenges.
        • Provide suggestions for specific challenge types.
        • Offer rewards for completing a certain number of challenges.
    }
Step 6. End

```

Student progress review. Smart recommendations through the Core Recommendation Engine, powered by data analytics and personalization of core engines as shown in Algorithm 3, present adaptive insights into student progress reviews. The dynamic nature of Core Engine's Multidimensional Progression ensures that recommendations align with student learning paths to support the novel approach to reviewing their progress.

Algorithm 3: Student Progress Review Implementation

```

Step 1. Conducting a smart programming progress review for the student
Step 2. Analyzing the progress review analytics
Step 3. Providing adaptive progress recommendations
Step 4. If the star rating is greater than or equal to Threshold Star Rating Value GOTO Step 5 ELSE GOTO Step 6
Step 5. Unlocking the 'Coding Prodigy' achievement, GOTO Step 7
Step 6. Handling cases where the star rating is not greater than or equal to Threshold Star Rating Value to handle non-perfect reviews.
    {
        • Encourage student to review feedback and recommendations.
        • Provide additional resources based on the feedback.
        • Ask for specific areas where improvement is desired.
    }
Step 7. End

```

Student engagement analysis. Data analytics and personalization of the Core Engine refine the analysis of student engagement by considering programming habits, such as code contribution frequency or participation in coding discussions, as shown in Algorithm 4. The Core Recommendation Engine provides dynamic suggestions for maintaining or increasing engagement and enhancing practical coding experience.

Algorithm 4: Student Engagement Analysis Implementation

```

Step 1. Analyzing programming engagement data
Step 2. If the average session time is greater than or equal to Threshold Session Time Value minutes
      GOTO Step 3 ELSE GOTO Step 4
Step 3. Commending high engagement, GOTO Step 5
Step 4. Handling cases where the average session time is less than Threshold Session Time Value
      minutes (handle low engagement).
      {
        • Encourage student to explore more content for an enhanced learning experience.
        • Provide tips for effective learning and time management.
        • Suggest joining collaborative activities or challenges.
      }
Step 5. End

```

Student challenges feedback. The Core Recommendation Engine, powered by Data Analytics & Personalization from the Core Engine as shown in Algorithm 5, intelligently responds to challenges with feedback on how one could have done better, thereby suggesting improvements that align with students' preferences. This smart feedback loop creates a more adaptive and enjoyable experience in addressing challenges.

Algorithm 5: Student Challenges Feedback Implementation

```

Step 1. Retrieving programming challenges feedback
Step 2. Analyzing Programming Challenges Feedback to provide smart challenges improvements
Step 3. If the total points from feedback analysis are greater than or equal to Threshold Feedback Total Points Value GOTO Step 4 ELSE GOTO Step 5
Step 4. Granting a feedback achievement GOTO Step 6
Step 5. Handling cases where the total points are less than Threshold Feedback Total Points Value,
      handle low feedback points.
      {
        • Encourage student to provide more detailed feedback for better improvements.
        • Highlight the importance of constructive criticism for enhancing the platform.
        • Introduce incentives for users who actively contribute valuable feedback.
      }
Step 6. End

```

Student accessibility preferences. In a more specific context such as programming, students' accessibility preferences can involve code readability preferences, syntax highlighting, or even language-specific settings. Core Engine applies accessibility settings as shown in Algorithm 6 for coding preferences to ensure the effectiveness and personalization of the coding experience.

Algorithm 6: Student Accessibility Preferences Implementation

```

Step 1. Retrieving user accessibility preferences
Step 2. Applying personalized accessibility settings
Step 3. If the total points from accessibility preferences are greater than or equal to Threshold Total Accessibility Preferences Points Value GOTO Step 4 ELSE GOTO Step 5
Step 4. Granting an accessibility reward, GOTO Step 6
Step 5. Handling cases where the total points are less than Threshold Total Accessibility Preferences Points Value, handle low accessibility points.
      {
        • Provide student with tips on how to enhance their accessibility settings.
        • Encourage student to explore additional accessibility features for a better experience.
        • Seek direct feedback from users to understand specific accessibility needs.
      }
Step 6. End

```

Student progress comparison across modules. The multidimensional progression comparison refines the comparison of student progress across the coding modules using the Core Engine as shown in Algorithm 7, thus offering an even more complete and personalized cross-module comparison with advanced insights through the Core Recommendation Engine and Data Analytics & Personalization of the Core Engine.

Algorithm 7: Student Progress Comparison Across Modules Implementation

```

Step 1. Comparing student progress across programming modules
Step 2. Providing advanced cross-module insights
Step 3. If the total points from module comparison are greater than or equal to Threshold Total Module Comparison Points Value GOTO Step 4 ELSE GOTO Step 5
Step 4. Granting a module comparison reward, GOTO Step 6
Step 5. Handling cases where the total points are less than Threshold Total Module Comparison Points Value, handle low module comparison points.
    {
        • Encourage student to explore additional modules for a more comprehensive learning experience.
        • Provide insights on how to optimize learning strategies to improve module comparison points.
        • Offer assistance or resources to address specific challenges identified in module comparison
    }
Step 6. End

```

Student exploration of advanced topics. Core Advanced Learning Opportunities by Core Engine as shown in Algorithm 8; therefore, the Core Recommendation Engine and Data Analytics & Personalization influenced the Core Engine recommendation for further advanced learning in programming with the help of real-time exploration carried out by students. This makes exploration more dynamic and adaptive to student performance.

Algorithm 8: Student Exploration of Advanced Topics Implementation

```

Step 1. Gathering data on user exploration of advanced programming topics
Step 2. Analyzing the exploration data
Step 3. Providing adaptive advanced learning opportunities based on the analysis
Step 4. If the total points from exploration are greater than or equal to Threshold Total Exploration Points Value GOTO Step 5 ELSE GOTO Step 6
Step 5. Granting an exploration achievement, GOTO Step 7
Step 6. Handling cases where the total points are less than Threshold Total Exploration Points Value, handle low exploration points.
    {
        • Encourage student to explore a broader range of advanced topics for a more diversified skill set.
        • Provide guidance on specific advanced topics that align with the student's interests.
        • Offer additional resources or challenges to boost points in advanced topic exploration.
    }
Step 7. End

```

Student mastery challenges. The Core Engine's Smart Mastery Improvement Recommendations, as shown in Algorithm 9, use advanced analytics and personalized insights to recommend Mastery Challenges that target the strengths

and weaknesses of students in programming concepts. The Core Engine ensures the multidimensionality of the student progression within the mastery experience.

Algorithm 9: Student Mastery Challenges Implementation

```

Step 1. Gathering data on student mastery programming challenges
Step 2. Analyzing the mastery Programming challenges data
Step 3. Providing smart mastery improvement recommendations based on the analysis
Step 4. If the total points from mastery challenges are greater than or equal to Threshold Total mastery challenges Points Value GOTO Step 5 ELSE GOTO Step 6
Step 5. Granting a mastery challenges achievement, GOTO Step 7
Step 6. Handling cases where the total points are less than Threshold Total mastery challenges Points Value, handle low mastery challenges points.
    {
        • Encourage student to revisit and practice specific challenging concepts.
        • Provide additional resources or hints for mastering complex programming challenges.
        • Suggest participating in collaborative challenges or discussions for shared learning experiences.
    }
Step 7. End

```

Student profile update. The Core Engine, as shown in Algorithm 10, verifies and applies dynamic profile updates to a greater range of user interactions specific to the programming course. The Core Recommendation Engine makes suggestions regarding updates related to the community, while data analytics and personalization of the Core Engine make these suggestions by personalizing evolving student profiles.

Algorithm 10: Student Profile Update Implementation

```

Step 1. Updating the student's dynamic programming profile
Step 2. Validating the dynamic programming profile update, if student profile updated GOTO Step 3 ELSE GOTO Step 5
Step 3. Applying the dynamic programming profile update GOTO Step 5
Step 4. Handling cases where the profile update is not valid
    {
        • Inform student that the profile update was not valid and guide them on the correct.
        • Provide a link or resource for student to review guidelines on updating his/her profile.
    }
    GOTO Step 8
Step 5. Checking if the total points from the student's profile update are greater than or equal to Threshold Total Student's Profile Points Value GOTO Step 6 ELSE GOTO Step 7
Step 6. Granting a dynamic community participation reward, GOTO Step 8
Step 7. Handling cases where the total points are less than Threshold Total Student's Profile Points Value, handle low community participation points.
    {
        • Encourage student to participate in more community activities to earn additional points.
        • Highlight the benefits of active participation and contributions to the community.
    }
Step 8. End

```

Student Satisfaction Survey. Smart student Satisfaction Surveys, run by Core Engine as shown in Algorithm 11, use advanced analytics to understand user sentiments within the programming context. The Core Recommendation Engine,

supported by data analytics and personalization of the Core Engine, further refines the survey results and recommends improvements that are needed for an even more satisfying learning journey.

Algorithm 11: Student Satisfaction Survey Implementation

```

Step 1. Conducting the smart programming student satisfaction survey
Step 2. Analyzing the results of the Programming student satisfaction survey
Step 3. If the satisfaction score is greater than or equal to Threshold Satisfaction Score Value GOTO
Step 4 ELSE GOTO Step 5
Step 4. Commending high satisfaction.
    {
    • Display a message commending student for his/her high satisfaction.
    • Offer additional rewards or benefits for highly satisfied student.
    }
GOTO Step 6.
Step 5. Handling cases where the satisfaction score is below Threshold Satisfaction Score Value,
handle low satisfaction scores.
    {
    • Encourage users to provide specific feedback on areas that need improvement.
    • Offer support channels for student to address their concerns or issues directly.
    }
Step 6. End

```

Student collaboration with peers. Through the multidimensional progression of Core Engine as shown in Algorithm 12, Adaptive Collaborative Learning drives the assurance of collaboration opportunities for programming projects or code reviews. The core recommendation engine with data analytics and personalization builds further collaboration data to continuously improve programming peer learning.

Algorithm 12: User Collaboration with Peers Implementation

```

Step 1. Retrieving data on student collaboration with programming peers
Step 2. Analyzing Student Collaboration with Programming Peers, the collaboration data
Step 3. Encouraging adaptive collaborative learning
Step 4. If the total points from collaboration are greater than or equal to Threshold Total
Collaboration Points Value GOTO Step 5 ELSE GOTO Step 6
Step 5. Granting dynamic community participation reward, GOTO Step 7
Step 6. Handling cases where the total points are below Threshold Total Collaboration Points Value,
handle low collaboration points.
    {
    • Provide suggestions on how student can increase collaboration with peers.
    • Highlight the benefits of collaborative learning and its impact on student progress.
    }
Step 7. End

```

Student time management analysis. Core Engine's Dynamic Time Management Analysis, as shown in Algorithm 13, with suggestions from the Core Recommendation Engine and insights provided by Core Engine's Data Analytics and Personalization, offers customized time management tips dynamically concerning programming. This dynamic nature ensures that time is used effectively and aligns with students' learning styles.

Algorithm 13: Student Time Management Analysis Implementation

```

Step 1. Analyzing the data related to student time management
Step 2. Providing dynamic time management tips based on the analysis
Step 3. If the total points from time management are greater than or equal to Threshold Total Management Points Value GOTO Step 4 ELSE GOTO Step 5
Step 4. Granting module comparison reward for effective time management, GOTO Step 6
Step 5. Handling cases where the total points are below Threshold Total Management Points Value, handle low time management points.
    {
        • Offer personalized advice on improving time management skills.
        • Provide challenges or activities to enhance time management.
        • Display personalized tips for effective time management.
        • Provide specific suggestions for improving time management skills.
        • Propose challenges or tasks to help users practice and enhance time management.
    }
Step 6. End

```

Student reflection on learning journey. Core Engine Data Analytics and Personalization recommend a learning path based on the adaptive learning paths of students that are reflected in programming journeys as shown in Algorithm 14. Core Engine Multidimensional Progression ensures variety and enrichment in the learning process regarding coding.

Algorithm 14: Student Reflection on Learning Journey Implementation

```

Step 1. Obtaining student reflection on their dynamic programming learning journey
Step 2. Analyzing the student's reflection to derive insights
Step 3. Providing personalized dynamic programming learning path recommendations
Step 4. If the total points from the reflection analysis are greater than or equal to Threshold Total Reflection Analysis Points Value GOTO Step 5 ELSE GOTO Step 6
Step 5. Granting accessibility reward for a well-reflected learning journey GOTO Step 7
Step 6. Handling cases where the total points are below Threshold Total Reflection Analysis Points Value, handle low reflection points.
    {
        • Encourage students to reflect more deeply on their learning experiences
            ◦ Send prompts or questions that encourage student to reflect more profoundly
        • Provide additional resources or challenges to enhance reflection
            ◦ Propose specific activities or challenges to enhance student reflective practices
    }
Step 7. End

```

4 EXPERIMENTS AND RESULTS

To verify the analytical research results, experimental results are introduced in this section. The experiment was conducted on students enrolled in an Internet Programming course at Al Hussein Bin Talal University. It is an undergraduate-level course taken by students majoring in computer science, artificial intelligence, and software engineering. This study proposes that our gamification system improves students' performance more than the traditional university classroom method for learning HTML programming. This section summarizes the statistical power analysis performed to test the alternative hypotheses.

4.1 Ethical considerations

Our study has been conducted according to ethical guidelines in the best interest of privacy, data security, and equity in the analysis of students' performance. Key ethical measures include:

- **Participant consent:** Participants were informed of the purpose and the procedures of our research methodology. The participants were informed in detail on how their data would be used through an informed consent form. Participants had the right to withdraw from the study at any time without any consequences. The consent form clearly stated the purpose of the study, how the data would be treated, and the rights of the participants.
- **Data anonymization and security:** All data about participants were anonymized, using unique identifiers that ensured no direct link to personally identifiable information (PII). Anonymized data were securely stored in encrypted formats on password-protected servers, with access restricted to authorized researchers. Only aggregated data used for analysis and publication are effectively minimized risks of identification of participants.
- **Data handling and confidentiality** All data handling followed institutional data protection guidelines. All personnel who handling data signed confidentiality agreements to avoid misuse. Data collection, storage, and disposal procedure were planned in a manner would ensure that the ethical best practices were followed to protect participant data throughout the research.
- **Minimizing potential biases:** The participant sample has tried to be diverse on the dimensions of gender, age, and academic backgrounds. Pre-tests were done to establish a baseline with regard to familiarity with gamified learning systems so that assessment would be on an equitable basis. Double-blind data analysis protocols were followed to avoid subjective bias in interpreting the results.
- **Transparency and accountability:** The research methodology and analysis processes were documented in detail to ensure reproducibility and enable peer review. Where relevant, participants were given access to findings, promoting openness and trust.

4.2 Materials

The materials used in this experiment were as follows:

- **Gamification system:** Develop an online learning game based on the HGLA framework, embedding game elements such as points, badges, leaderboards, and challenges in a manner relevant to learning HTML programming.
- **Traditional classroom materials:** General textbooks, lecture notes, and in-class exercises are deployed in the conventional method of teaching programming in HTML.
- **HTML programming proficiency tests:** A similar standardized HTML proficiency test, pre-test, and post-test were given to both groups, and were designed to measure the student attainment of knowledge and skills in HTML Programming, with scores ranging from 0 to 100 for both tests. At the beginning of the study, all students in both groups took the same standardized HTML proficiency test (pre-test) to ensure knowledge equivalence regarding HTML

programming between the two groups. At the end of the instructional period, all students took the same standardized HTML proficiency test (post-test) in both groups. This test measured students' understanding and practical skills in HTML programming at the end of the course.

4.3 Procedure

The quantitative semi-experimental study, using a convenience sample, was conducted at the IT College of Al Hussein Bin Talal University. It involved four classes of an HTML programming course covering HTML, CSS, and JavaScript. The classes were divided into two groups: an experimental group and a control group (traditional classroom), both randomly assigned to the HTML programming course. The experimental group and the control group followed the same syllabus, content coverage, and study duration.

- **Experimental group (gamification system):** This group used a gamification system to learn HTML programming.
- **Control group (traditional classroom):** This group used the traditional classroom method to learn HTML programming.
- **Instructional period:** Both groups underwent an instructional period of 16 weeks, covering the same syllabus and content but using different methods.
- **Teaching method:** Experimental group engaged with the gamification system, which included interactive lessons, quizzes, and game-like elements, to motivate learning. The control group attended traditional classroom sessions involving lectures, discussions, and hands-on exercises, without any gamified elements.
- **Monitoring and support:** Both groups had access to the teacher for guidance and support throughout the instructional period. Additionally, periodic assessments were conducted to ensure that both groups progressed according to the syllabus.
- **Results assessment:** At the beginning of the study all students for both groups took the same standardized HTML proficiency test (pre-test) to ensure the knowledge equivalence of HTML programming between the two groups. also, At the end of the instructional period, all students took the same standardized HTML proficiency test scores (post-test) of both groups, this test measured their understanding and practical skills in HTML programming

4.4 Statistical analysis

This experimental design ensured a controlled comparison between the gamification approach and the traditional classroom method, allowing for rigorous evaluation of the impact of gamification on students' HTML learning outcomes.

The primary analysis involved comparing the mean test scores of the two groups to determine the effectiveness of the gamification system. The following statistical procedures were used.

Descriptive statistics. Descriptive statistics are used to calculate the mean and standard deviation for both groups. The pre-test scores, as shown in Table 2 and Figure 4, were analyzed to ensure that both the experimental and control groups were equivalent at the beginning of the study. The post-test scores shown in Table 3 and Figure 5 were analyzed to evaluate the effectiveness of the gamification system compared to the traditional classroom method.

Table 2. HGLA and control groups pre-test descriptive statistics of scores

Group	Experimental Group	Control Group
Mean	52.450	52.256
SD	3.823	3.324
Sig. Difference	0.8054	

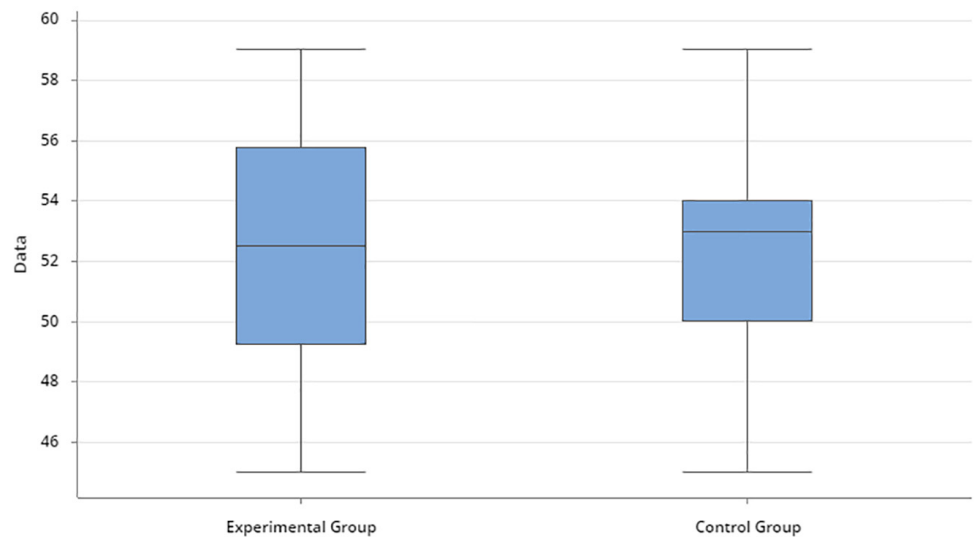


Fig. 4. Pre-test for the HGLA experimental group and control group student's comparison graph

Table 3. HGLA and control groups post-test descriptive statistics of scores

Group	Experimental Group	Control Group
Mean	87.875	75.302
SD	3.770	3.067
Sig. Difference	<0.001	

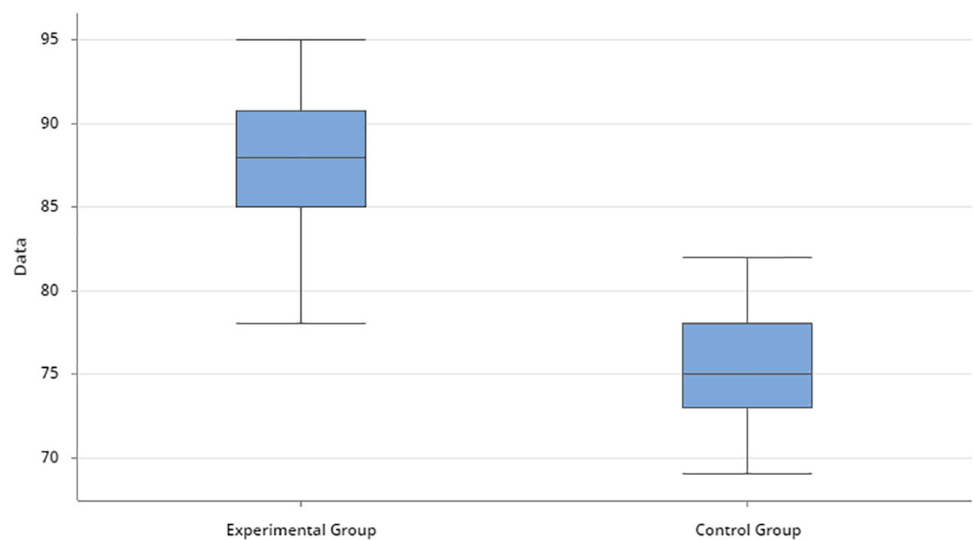


Fig. 5. Post-test for the HGLA experiment group and control group student's comparison graph

Independent samples t-test. The independent samples t-test is compare the mean scores of the experimental group (M_x) and the control group (M_y) and test the hypotheses:

- **Pre-test hypotheses:**

H₀ (Null Hypothesis): There is no difference in the mean HTML proficiency pre-test scores between the gamification and traditional classroom groups ($M_x - M_y = 0$).

H_a (Alternative Hypothesis): The mean HTML pre-test scores of the two groups were not equivalent ($M_x - M_y \neq 0$).

The independent samples t-test for the pre-test scores showed that there was no significant difference between the groups ($t = 0.2472$, $p = 0.8054$), indicating that the random assignment was effective and that the groups were comparable.

- **Post-test hypotheses:**

H₀ (Null Hypothesis): There is no difference in the mean HTML proficiency test scores (post-test) between the gamification and traditional classroom groups ($M_x - M_y = 0$).

H_a (Alternative Hypothesis): The mean HTML proficiency test score (post-test) of the gamification group was greater than that of the traditional classroom group ($M_x - M_y > 0$).

The independent samples t-test for the post-test scores revealed an extremely statistically significant difference between the groups ($t = 16.7181$, $p < 0.001$), with the Experimental Group outperforming the Control Group. This result supports the alternative hypothesis (*H_a*) that the mean HTML proficiency test score of the gamification group is higher than that of the traditional classroom group.

This study investigates the impact of a gamified learning system on HTML programming proficiency among graduate students at Al Hussein Bin Talal University. The results indicate that the HGLA Gamification Framework significantly enhances students' learning outcomes compared with traditional classroom methods. These findings suggest that incorporating game-like elements into educational systems can be an effective strategy for improving student engagement and academic performance.

5 DISCUSSION

The results of this study have highlighted the effectiveness of the HGLA framework in enhancing programming education, particularly for HTML programming. The findings showed that the gamified approach was more effective than the traditional classroom approach in engaging students in efforts to improve their proficiency.

A major advantage of the HGLA framework is its potential to scale in mobile learning environments. The framework, by applying the use of mobile technologies, can offer flexibility and accessibility to a greater number of students. The main key features with high levels of relevance to mobile learning include real-time adaptive feedback, cross-device synchronization, and location-based tasks. These aspects make the HGLA framework suitable for use at distance and low-resource environments where mobile devices have been found as the preferred medium choice for accessing the learning material. Further developments may focus on fine-tuning the framework to use it on more types of mobile devices and operating systems.

Its offline capability and multilinguality will add much worth to make it accessible for various learning environments.

While the study was focused on a convenience sample from the IT College of Al Hussein Bin Talal University, its design lays a foundation for future research in the extension of these findings across multiple programming courses, institutions, and more diverse students. Such extensions would enhance the generalizability of the results and, consequently, validate the applicability of the framework in varied educational and cultural contexts. Therefore, a greater balance of participants across institutions with various demographics and academic profiles might serve to enhance this study in establishing the efficacy of the framework. Further, such a longitudinal design tracking students over semesters will shed more detailed information on the deeper impact that can be brought in by the HGLA framework.

In fact, the HGLA framework bears transformative potential in programming education both over time and across a variety of institutions. Its gamification feature, combined with learning through narrative, maintains consistency in student engagement and retention to attain skills in programming by continuously practicing and learning progressively. Over several semesters, this translates into deeper understanding and long-term expertise. Further standardization of the framework across institutions would provide a basis for resource sharing and collaboration toward developing a unified ecosystem for gamified learning. Its data-driven insights support iterative improvements to ensure ongoing relevance and effectiveness. This framework is thus, in addition to being applied across cultural and educational contexts, a useful tool in the effort to address inequities, foster inclusivity, and expand access to quality education. Periodically updating the curriculum with respect to industry standards will better prepare students to compete in technology-related careers and assure their long-term success in an ever-changing job market.

6 LIMITATIONS AND FUTURE WORK

While the study and the HGLA framework present promising results and provide meaningful insights into programming education, several limitations should be considered to properly contextualize the findings and guide future research efforts.

- **Limitations in mobile applications:** Despite the potential, the mobile application of the HGLA framework is not without its limitations. First, the reliance on consistent internet connectivity can be a barrier in regions with limited digital infrastructure. While the current framework provides a seamless experience for connected learners, future development should consider offline capabilities to ensure continuity of learning in low-bandwidth or offline scenarios. Second, the user interface design must be device and screen size-compatible, from a smartphone to a tablet computer. Ensuring usability across those platforms is important for keeping learners engaged. Moreover, the design of mobile-specific gamification features needs to consider cultural and educational contexts since they may affect the acceptance of the gamified approach. Third, the integration of mobile technologies introduces problems with privacy and security. While the study has emphasized data anonymization and secure handling practices, mobile devices may add other vulnerabilities that must be addressed through robust cybersecurity measures.
- **Exclusion of multilingual and non-English studies:** One of the limitations is that the literature review was conducted mainly based on research articles published in English, excluding other language publications. This restrains the

possibility of including potentially important findings from regions where gamification in programming education has been studied but remains underrepresented in publications released in the English language.

- **Varied familiarity with gamified learning systems:** The experience with gamified learning systems was manifold among the participants, which may influence their levels of engagement and, accordingly, their learning outcomes. While initial assessments were conducted, these differences in previous experience might result in inconsistent results. This could be further controlled in successive research by having a standardized initial introduction phase that familiarized all participants with the gamified system before the intervention started.
- **Cultural and educational context:** These results may reflect the cultural and pedagogical practices particular to the Jordanian educational system. The cultural and pedagogical practices might influence the acceptance and effectiveness of gamification in ways that could diverge from other educational systems. This would be further supported by comparative studies across different cultural contexts to establish a wider applicability of the HGLA framework.

7 CONCLUSION

The HGLA framework, enriched with Core Engine components, opens a new perspective in programming education. The integration of the Core Engine's Data Analytics & Personalization, Core Recommendation Engine, and Multidimensional Progression into the programming courses offers an innovative adaptive environment. Students receive a dynamic and personalized learning process that helps them engage in mastering their coding skills with satisfaction. The important strength of the HGLA framework is that it is open to further development on mobile learning. Since it is designed to be fully accessible, the framework provides for the learner to work on content easily on a smartphone, tablet, or other mobile platforms. This allows for inclusivity, especially for countries with different levels of digital infrastructure, and gives an equitable opportunity for learning. Besides, the added value of mobile-specific gamification features, such as location-based tasks, real-time adaptive feedback, and cross-device synchronization, extends the practical relevance of the HGLA framework to a wide variety of educational contexts.

Long-term impacts of the HGLA framework on programming education outcomes will be considered in future research, with particular emphasis on mobile learning, along with its adaptability to other subjects and levels of education. The proposed study will involve a larger sample from multiple institutions to ensure that the results are more generalizable. Also, this might be complemented with the inclusion of other triangulation methods and qualitative data through student feedback for deepening insight into students' experiences and preferences to further refine gamified learning environments.

8 REFERENCES

- [1] G. Siemens and P. Tittenberger, *Handbook of Emerging Technologies for Learning*. Canada: University of Manitoba, p. 65, 2009.
- [2] V. Ratten and P. Usmanij, "Entrepreneurship education: Time for a change in research direction?" *The International Journal of Management Education*, vol. 19, no. 1, p. 100367, 2021. <https://doi.org/10.1016/j.ijme.2020.100367>

- [3] J. Hamari, J. Koivisto, and H. Sarsa, "Does gamification work?—A literature review of empirical studies on gamification," in *2014 47th Hawaii International Conference on System Sciences*, Waikoloa, HI, USA, 2014, pp. 3025–3034. <https://doi.org/10.1109/HICSS.2014.377>
- [4] A. I. Zourmpakis, M. Kalogiannakis, and S. Papadakis, "Adaptive gamification in science education: An analysis of the impact of implementation and adapted game elements on students' motivation," *Computers*, vol. 12, no. 7, p. 143, 2023. <https://doi.org/10.3390/computers12070143>
- [5] M. A. Steinmaurer, "Engaging learning concepts and technologies for fostering digital competences and programming in computer science education," Doctoral Dissertation, Goethe University Frankfurt am Main, 2023.
- [6] A. Robins, J. Rountree, and N. Rountree, "Learning and teaching programming: A review and discussion," *Computer Science Education*, vol. 13, no. 2, pp. 137–172, 2003. <https://doi.org/10.1076/csed.13.2.137.14200>
- [7] M. Guzdial, "A media computation course for non-majors," in *Proceedings of the 8th Annual Conference on Innovation and Technology in Computer Science Education*, 2003, pp. 104–108. <https://doi.org/10.1145/961511.961542>
- [8] L. S. Vygotsky, *Mind in Society: The Development of Higher Psychological Processes*, vol. 86, Harvard University Press, 1978.
- [9] C. Kelleher, R. Pausch, and S. Kiesler, "Storytelling alice motivates middle school girls to learn computer programming," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 2007, pp. 1455–1464. <https://doi.org/10.1145/1240624.1240844>
- [10] K. M. Kapp, *The Gamification of Learning and Instruction: Game-Based Methods and Strategies for Training and Education*. Pfeiffer, 2012. <https://doi.org/10.1145/2207270.2211316>
- [11] J. Hamari, J. Koivisto, and H. Sarsa, "Does gamification work? A literature review of empirical studies on gamification," in *2014 47th Hawaii International Conference on System Sciences*, 2014, pp. 3025–3034. <https://doi.org/10.1109/HICSS.2014.377>
- [12] C. A. Anderson and K. E. Dill, "Video games and aggressive thoughts, feelings, and behavior in the laboratory and in life," *Journal of Personality and Social Psychology*, vol. 78, no. 4, p. 772, 2000. <https://doi.org/10.1037/0022-3514.78.4.772>
- [13] R. Azevedo (Ed.), "Using hypermedia as a metacognitive tool for enhancing student learning? The role of self-regulated learning," in *Computers as Metacognitive Tools for Enhancing Learning*, Routledge, 2018, pp. 199–209. <https://doi.org/10.4324/9781315866239>
- [14] A. I. Zourmpakis, M. Kalogiannakis, and S. Papadakis, "A review of the literature for designing and developing a framework for adaptive gamification in physics education," in M. F. Tasar (Ed.), *The International Handbook of Physics Education Research: Teaching Physics*, AIP Publishing LLC, 2023. https://doi.org/10.1063/9780735425712_005
- [15] N. Sirikhvunchai and K. Nakpan, "Designing interactive mobile application to create online-learning experience of the future," in *2024 Joint International Conference on Digital Arts, Media and Technology with ECTI Northern Section Conference on Electrical, Electronics, Computer and Telecommunications Engineering (ECTI DAMT & NCON)*, Chiang-mai, Thailand, 2024, pp. 180–185. <https://doi.org/10.1109/ECTIDAMTNCN60518.2024.10480096>
- [16] A. Amory and R. Seagram, "Educational game models: Conceptualization and evaluation," *South African Journal of Higher Education*, vol. 17, no. 2, pp. 206–217, 2003. <https://doi.org/10.4314/sajhe.v17i2.25314>
- [17] K. Kiili, "Digital game-based learning: Towards an experiential gaming model," *The Internet and Higher Education*, vol. 8, no. 1, pp. 13–24, 2005. <https://doi.org/10.1016/j.iheduc.2004.12.001>

- [18] S. De Freitas and M. Oliver, "How can exploratory learning with games and simulations within the curriculum be most effectively evaluated?" *Computers & Education*, vol. 46, no. 3, pp. 249–264, 2006. <https://doi.org/10.1016/j.compedu.2005.11.007>
- [19] A. Yusoff, R. Crowder, L. Gilbert, and G. Wills, "A conceptual framework for serious games," in *2009 Ninth IEEE International Conference on Advanced Learning Technologies*, Riga, Latvia, 2009, pp. 21–23. <https://doi.org/10.1109/ICALT.2009.19>
- [20] C. Hartevelde, *Triadic Game Design: Balancing Reality, Meaning and Play*. Springer Science & Business Media, 2011. <https://doi.org/10.1007/978-1-84996-157-8>
- [21] T. Marsh, "Serious games continuum: Between games for purpose and experiential environments for purpose," *Entertainment Computing*, vol. 2, no. 2, pp. 61–68, 2011. <https://doi.org/10.1016/j.entcom.2010.12.004>
- [22] S. Arnab et al., "Mapping learning and game mechanics for serious games analysis," *British Journal of Educational Technology*, vol. 46, no. 2, pp. 391–411, 2015. <https://doi.org/10.1111/bjet.12113>
- [23] S. Deterding, D. Dixon, R. Khaled, and L. Nacke, "From game design elements to gamefulness: Defining gamification," in *Proceedings of the 15th International Academic MindTrek Conference: Envisioning Future Media Environments*, 2011, pp. 9–15. <https://doi.org/10.1145/2181037.2181040>
- [24] Y. K. Chou, *Actionable Gamification: Beyond Points, Badges, and Leaderboards*. Packt Publishing Ltd., 2019.
- [25] R. M. Ryan and E. L. Deci, "Self-determination theory and the facilitation of intrinsic motivation, social development, and well-being," *American Psychologist*, vol. 55, no. 1, pp. 68–78, 2000. <https://doi.org/10.1037/0003-066X.55.1.68>
- [26] M. Resnick, *Lifelong Kindergarten: Cultivating Creativity through Projects, Passion, Peers, and Play*. MIT Press, 2017. <https://doi.org/10.7551/mitpress/11017.001.0001>
- [27] A. C. T. Klock, I. Gasparini, M. S. Pimenta, and J. Hamari, "Tailored gamification: A review of literature," *International Journal of Human-Computer Studies*, vol. 144, p. 102495, 2020. <https://doi.org/10.1016/j.ijhcs.2020.102495>
- [28] Y. Xu, "Effective gamification design: A literature review," *The SIJ Transactions on Computer Science Engineering & its Applications*, vol. 3, no. 4, pp. 47–54, 2015. <https://doi.org/10.9756/SIJCSEA/V3I2/03040120201>
- [29] A. Christopoulos and S. Mystakidis, "Gamification in education," *Encyclopedia*, vol. 3, no. 4, pp. 1223–1243, 2023. <https://doi.org/10.3390/encyclopedia3040089>
- [30] Z. Zainuddin, S. K. W. Chu, M. Shujahat, and C. J. Perera, "The impact of gamification on learning and instruction: A systematic review of empirical evidence," *Educational Research Review*, vol. 30, p. 100326, 2020. <https://doi.org/10.1016/j.edurev.2020.100326>
- [31] F. C. Blumberg et al., "Current state of play: Children's learning in the context of digital games," *Journal of Children and Media*, vol. 18, no. 2, pp. 293–299, 2024. <https://doi.org/10.1080/17482798.2024.2335725>
- [32] S. Subhash and E. A. Cudney, "Gamified learning in higher education: A systematic review of the literature," *Computers in Human Behavior*, vol. 87, pp. 192–206, 2018. <https://doi.org/10.1016/j.chb.2018.05.028>
- [33] L. Smirani and H. Yamani, "Analysing the impact of gamification techniques on enhancing learner engagement, motivation, and knowledge retention: A structural equation modelling approach," *Electronic Journal of e-Learning*, vol. 22, no. 9, pp. 111–124, 2024. <https://doi.org/10.34190/ejel.22.9.3563>
- [34] P. Luarn, C. C. Chen, and Y. P. Chiu, "Enhancing intrinsic learning motivation through gamification: A self-determination theory perspective," *The International Journal of Information and Learning Technology*, vol. 40, no. 5, pp. 413–424, 2023. <https://doi.org/10.1108/IJILT-07-2022-0145>

- [35] F. Ke, "Designing and integrating purposeful learning in game play: A systematic review," *Educational Technology Research and Development*, vol. 64, pp. 219–244, 2016. <https://doi.org/10.1007/s11423-015-9418-1>
- [36] R. Damaševičius and T. Sidekerskienė, "Virtual worlds for learning in metaverse: A narrative review," *Sustainability*, vol. 16, no. 5, p. 2032, 2024. <https://doi.org/10.3390/su16052032>
- [37] P. Stamatiou, "Can preschoolers learn computational thinking and coding skills with ScratchJr? A systematic literature review," *International Journal of Educational Reform*, vol. 33, no. 1, pp. 28–61, 2024. <https://doi.org/10.1177/10567879221076077>
- [38] C. Cao, "Exploring the effectiveness of gamification beyond traditional mechanics in cultivating Chinese international postgraduate students' sense of belonging in programming courses," Doctoral dissertation, University of Sheffield, 2023.
- [39] A. Y. R. Hmoud, O. H. Salah, and R. A. H. Altalib, "The adoption of gamification in higher education and its impact on academic performance: Empirical evidence from Jordan and Palestine," *Cogent Education*, vol. 11, no. 1, p. 2428907, 2024. <https://doi.org/10.1080/2331186X.2024.2428907>
- [40] K. Louka *et al.*, "Enhancing computational thinking in early childhood education through ScratchJr integration" *Heliyon*, vol. 10, no. 10, p. e30482, 2024. <https://doi.org/10.1016/j.heliyon.2024.e30482>
- [41] S. A. Alwaely *et al.*, "Challenges of using social media in accepting students for E-Learning in higher education institutions: A case study from the Hashemite Kingdom of Jordan," *Kurdish Studies*, vol. 12, no. 1, pp. 2226–2239, 2024.
- [42] A. Bounou, K. Lavidas, V. Komis, S. Papadakis, and P. Manoli, "Correlation between high school students' computational thinking and their performance in STEM and language courses," *Education Sciences*, vol. 13, no. 11, p. 1101, 2023. <https://doi.org/10.3390/educsci13111101>
- [43] N. S. Raj, and V. G. Renumol, "A systematic literature review on adaptive content recommenders in personalized learning environments from 2015 to 2020," *Journal of Computers in Education*, vol. 9, no. 1, pp. 113–148, 2022. <https://doi.org/10.1007/s40692-021-00199-4>
- [44] C. G. Lechuga, "A multidimensional perspective of mathematical ability for eliciting innovations in education," Doctoral dissertation, UC Irvine, 2024.
- [45] R. Dunn and S. A. Griggs, *Practical Approaches to using Learning Styles in Higher Education*. Bloomsbury Publishing USA, 2000. <https://doi.org/10.5040/9798400699962>
- [46] L. Archambault, H. Leary, and K. Rice, "Pillars of online pedagogy: A framework for teaching in online learning environments," *Educational Psychologist*, vol. 57, no. 3, pp. 178–191, 2022. <https://doi.org/10.1080/00461520.2022.2051513>
- [47] H. Uğraş, M. Uğraş, S. Papadakis, and M. Kalogiannakis, "ChatGPT-supported education in primary schools: The potential of ChatGPT for sustainable practices," *Sustainability*, vol. 16, no. 22, p. 9855, 2024. <https://doi.org/10.3390/su16229855>
- [48] F. Uy, O. K. Kilag, and A. Arcilla Jr., "Empowering education: A learning-goals-centric approach to curriculum development," *Excellencia: International Multi-disciplinary Journal of Education*, vol. 1, no. 4, pp. 48–61, 2023.
- [49] D. Zhao, C. H. Muntean, A. E. Chis, G. Rozinaj, and G. M. Muntean, "Game-based learning: Enhancing student experience, knowledge gain, and usability in higher education programming courses," *IEEE Transactions on Education*, vol. 65, no. 4, pp. 502–513, 2022. <https://doi.org/10.1109/TE.2021.3136914>
- [50] A. Abbas, "Turning ideas into reality: Designing and implementing Software Dev Turnkey Labs," OSFPREPRINTS, 2023. <https://doi.org/10.31219/osf.io/uafr>

- [51] S. Papadakis and M. Kalogiannakis, "Evaluating the effectiveness of a game-based learning approach in modifying students' behavioural outcomes and competence, in an introductory programming course. A case study in Greece," *International Journal of Teaching and Case Studies*, vol. 10, no. 3, pp. 235–250, 2019. <https://doi.org/10.1504/IJTCS.2019.102760>
- [52] A. Skulmowski and K. M. Xu, "Understanding cognitive load in digital and online learning: A new perspective on extraneous cognitive load," *Educational Psychology Review*, vol. 34, no. 1, pp. 171–196, 2022. <https://doi.org/10.1007/s10648-021-09624-7>
- [53] A. Pyae *et al.*, "Exploring user experience and usability in a metaverse learning environment for students: A usability study of the Artificial Intelligence, Innovation, and Society (AIIS)," *Electronics*, vol. 12, no. 20, p. 4283, 2023. <https://doi.org/10.3390/electronics12204283>
- [54] S. Assapun and P. Thummaphan, "Assessing the effectiveness of board game-based learning for enhancing problem-solving competency of lower secondary students," *International Journal of Instruction*, vol. 16, no. 2, pp. 511–532, 2023. <https://doi.org/10.29333/iji.2023.16228a>
- [55] X. Li and Y. Li, "Individualized and innovation-centered general education in a Chinese STEM University," *Education Sciences*, vol. 13, no. 8, p. 846, 2023. <https://doi.org/10.3390/educsci13080846>
- [56] S. Mystakidis, "Motivation enhancement methods for community building in extended reality," in *Augmented and Mixed Reality for Communities*, CRC Press, 2021, pp. 265–282. <https://doi.org/10.1201/9781003052838-17>
- [57] S. Zhang *et al.*, "Understanding student teachers' collaborative problem solving: Insights from an epistemic network analysis (ENA)," *Computers & Education*, vol. 183, p. 104485, 2022. <https://doi.org/10.1016/j.compedu.2022.104485>
- [58] L. Pei, C. Poortman, K. Schildkamp, and N. Benes, "Teachers' and students' perceptions of a sense of community in blended education," *Education and Information Technologies*, vol. 29, no. 2, pp. 2117–2155, 2024. <https://doi.org/10.1007/s10639-023-11853-y>

9 AUTHORS

Ibrahim Alkore Alshalabi is currently an Assistant Professor in the Faculty of Information Technology at Al-Hussein Bin Talal University, Ma'an, Jordan. He received his B.Sc. in Computer Science from Al-Isra Private University, Amman, Jordan in 1997, his Master of Computer Applications (MCA) from Bangalore University, India in 2007, and his PhD in Computer Science and Engineering from the University of Bridgeport, USA in 2016. His research interests are in the areas of E-Learning, M-Learning, Instructional Technology, Smart systems, wireless communications, and networks (E-mail: ibrahim.m.alshalabi@ahu.edu.jo).

Tawfiq Alrawashdeh is currently an Assistant Professor in the Faculty of Information Technology at Al-Hussein Bin Talal University, Ma'an, Jordan. He received his B.Sc. in computer science from Al Hussein Bin Talal University (AHU), Jordan, in 2005. He received his master's in computer science from Al-Balqa' Applied University, Jordan in 2011. He received his Ph.D. in 2020 from Universiti Sultan Zainal Abidin, Malaysia. His research interests in the areas of cloud computing, task scheduling and smart systems (E-mail: tawfiq@ahu.edu.jo).

Anas AbuKaraki is currently an Assistant Professor in the Faculty of Information Technology at Al-Hussein Bin Talal University, Ma'an, Jordan. He received his B.S. degree in Computer science from Israa University, Jordan, in 1997, and the

M.S. Degree in Computer network from university of western system (UWS), Australia, he received his Ph.D. degrees in Management Information Systems from Om Durman Islamic University (OIU), Sudan, in 2016. His research interests include decision support, fuzzy, and smart systems (E-mail: anas@ahu.edu.jo).

Malek Zakarya Alksasbeh is currently a Full Professor in the Faculty of Information Technology at Al-Hussein Bin Talal University, Ma'an, Jordan. He received his B.S. degree in Computer science from Mu'tah University, Jordan, in 2005, and the M.S. and Ph.D. degrees in Information Technology from the University Utara Malaysia (UUM), Malaysia, in 2008 and 2012, respectively. His research interests are in the areas of Smart Systems, Information Retrieval, and Instructional Technology (E-mail: malksasbeh@ahu.edu.jo).