A New Approach to Interactive Education: Game Engine-Based Frameworks for Teachers to Develop Interactive Lessons

Mocanu Andrei Ionuț[1], **Nicolae Mihnea Vlad**[2], **Duțescu Răzvan Alexandru**[3], **Apostol Mihai**[4], **Petre Ionuț**[5]

1. National Institute for Research & Development in Informatics – ICI Bucharest, Bucharest, Romania, ionut.mocanu@ici.ro
2. National Institute for Research & Development in Informatics – ICI Bucharest, Bucharest, Romania, mihnea.nicolae@ici.ro
3. National Institute for Research & Development in Informatics – ICI Bucharest, Bucharest, Romania, razvan.dutescu@ici.ro
4. National Institute for Research & Development in Informatics – ICI Bucharest, Bucharest, Romania, mihai.apostol@ici.ro
5. National Institute for Research & Development in Informatics – ICI Bucharest, Bucharest, Romania, ionut.petre@ici.ro

DOI: 10.63467/alls16.art16

Abstract

Recent studies in the field of education show a significant increase in students' level of concentration during classes when the learning experience is highly immersive. During lessons, students are engaged with immersive instructional materials, whether through exercises or video presentations. On average, this results in better information retention and a deeper understanding of the subject matter. Existing applications, whether in VR or in 2D/3D for computers and mobile devices, already offer rich curricula that approach abstract topics through practical student experiences.

This article introduces a new approach to creating teacher-centered interactive educational applications. In conventional applications, teachers must find the software that best fits their course, and in some cases, they must even adapt their existing teaching materials to align with the chosen application. To address these challenges, this paper proposes a framework based on a game engine (Unity or Unreal Engine) that enables teachers to design interactive lessons through intuitive interfaces. These panels contain blocks representing sample lessons or educational scenarios, which teachers can modify via drag-and-drop to match their own instructional content. Furthermore, the article outlines a system for collecting student performance data during lessons. At the end of each session, students may also provide feedback, enabling teachers to refine and improve their lessons based on results and reviews.

The proposed framework aims to increase teacher autonomy, lesson customization, and student engagement in immersive learning environments.

Keywords: interactive education, game engine, ICT

1.Introduction

1.1 The Current State of the Romanian Educational System

In the OECD report for 2025, published on September 9, 2025 (Organisation for Economic Co-operation and Development [OECD], 2025), the amounts of money the Romanian state invests per pupil and per university student are presented as: USD 6,069 and USD 10,329, respectively. Compared with other countries, these figures are well below the average. For university students, the average investment per student is USD 15,102. The gap for pupils is also significant, with Romania ranking among the last EU countries on this metric.

Due to an underfunded education system, phenomena such as school dropout and socio-economic segregation arise, and in families facing extreme hardship, children may not be enrolled in any level of education at all. A high share is reported among children (ages 6-14) and young people (15-19) who are not enrolled in any form of education - 16% and 32%, respectively. These percentages are very high compared with the European average of: 2% and 16%.

Beyond the problems arising from insufficient funding, social constructs end up dividing educational institutions into “schools with a good reputation” and “schools with a poor reputation". In many cases, an elitist and segregationist approach is present, which translates into deepening the educational inequalities among students. Thus, creating an environment that runs counter to the foundations of education - one that fails to ensure equal chances and opportunities for all young people, regardless of family economic situation, ethnicity, or other socio-economic criteria (Țoc, 2024). A high-performing educational system is measured by its capacity to impart core competencies to every student - an inclusive system that does not permit the gradual exclusion of students facing socio-economic hardship.

1.2 Motivation and research objectives

The main objective of this article is to propose a game-engine-based framework that enables teachers to create interactive educational applications. The framework's usage model relies on pre-existing, general examples for each subject. These examples are presented as an intuitive menu where the user can deploy visual blocks by dragging them into the scene in an intuitive manner. The user is free to modify parameters or specific features to suit their needs.

The main strength of this framework is the freedom presented to the teacher. With existing applications, the teacher has to find the preexisting option that most closely matches their subject; in some cases, the teacher even has to rely on modifying the course curriculum to integrate such applications.

Another important aspect is student feedback. At the end of each lab or lesson, students can give a rating and, if they wish, indicate what changes should be made or what they liked the most.

2. State of the Art

2.1 Immersive learning and Virtual Reality use in education - STEM learning

The rapid advancement of technology has changed the way information is being shared and perceived. One of the most impacted fields by these innovations is education. Among emerging technologies, Virtual Reality (VR) has shown particular promise in developing Science, Technology, Engineering, and Mathematics (STEM) learning even more. One of the challenges of traditional teaching methods is making students understand abstract concepts. VR addresses these problems by providing an immersive, interactive, and safe environment where children can visualize and experiment with scientific phenomena that would otherwise be difficult, or impossible to experience in real life. VR also increases the motivation of students, the engagement with the subject matter, and the retention of information, positioning it as a novel way of modernizing learning and education.

VR technology can help teachers overcome the limitations of traditional teaching methods by creating a hands-on experience in a simulated and controlled environment. The benefits of VR being used in STEM learning are: creating an immersive learning experience - VR creates realistic and engaging simulations of scenarios, allowing students to learn actively rather than just observe; enhanced motivation for students - it catches the students' attention by offering a hands-on experience; better retention of information; and access to a personalized and self-directed learning environment.

2.2 Using game engines (Unity, Unreal) in education

In the article (Paul et al., 2012), a game engine is described as a platform that handles common game tasks - such as rendering, physics, and input. It is a collection of reusable components that can be manipulated to bring a video game to life, allowing developers to focus on the specific details that make the game unique.

Thanks to the technological advances of the last decade, game engines such as Unity and Unreal Engine have moved beyond entertainment, becoming robust tools in key fields due to their capabilities for simulations and training. Classic game engines - Unity, Unreal Engine, and Godot - offer extensive support for industries including engineering (automotive, industrial, aerospace), the military/defense sector, medicine, and education (Duțescu et al., 2025). Beyond company support, the community is another important factor. There are numerous open-source community projects that aid learning, creating a conducive environment for education - further reinforced by the fact that the engines mentioned above are free and offer student packages, courses, or discounts.

The table below presents several comparisons between the two most popular game engines on the market today: Unity and Unreal Engine. The comparisons were taken from the article (Salama & Elsayed, 2021). The conclusion was that no single engine is a clear winner across all categories, and the choice should be made according to one's specific needs.

Game Engine	Pros	Cons
Unity	- Supports 25 different platforms - Optimised graphical performance for every platform - Uses C# and JavaScript - Huge Asset Store - Drag and drop option - Involved community	- Hard to learn - Graphical optimisation can be time-consuming
Unreal Engine	- Very good graphical capabilities - Good templates - Blueprint system (drag and drop programming)	- Uses C++ - Not optimized for low-resource / low-spec devices.

Table 1. Unity vs. Unreal Engine

Unity stands out on its own, given the characteristics of the framework proposed in this article. It comes with lower hardware requirements and costs, being a platform that is friendly to mobile phones, tablets, and similar devices.

Using a game engine in education supports the following claim: a person perceives 80% of information through sight (Sobota & Pietriková, 2023). This high percentage indicates the need for a high-performance visual system running on appropriate hardware resources. Even though there are challenges - such as the high cost of hardware and teacher training - it offers a different approach to education, serving as a tool that benefits both teachers and students. It gives teachers curricular freedom, while students can learn abstract concepts in interactive environments without fearing the consequences of mistakes.

3. The Proposed Architecture

To constitute an effective and comprehensive learning environment, the proposed framework must integrate a set of essential functional features. It must ensure multiplayer compatibility to enable real-time collaborative work sessions and provide functionalities for configuring virtual workspaces, such as choosing a virtual classroom or setting the number of available seats. A central component is the educational "prefab" system—predefined, reusable packages for specific subjects, such as physics experiments or interactive maps—supplemented by the integration of subject-specific teaching materials. Furthermore, the platform must include a grading system with LMS integration to ensure interoperability with platforms like Moodle. Finally, the framework must comprehensively manage session data, incorporating audio-video recording, a data-saving system for storing participant lists and generated materials, and data collection on student progress, difficulties, and engagement.

3.1 Workflow example

From the educator's perspective, the life cycle of a single educational session within the framework is divided into four sequential stages. This design provides a clear and manageable process, beginning with the complete configuration of the session, followed by the pre-lesson initialization of materials, proceeding to the live student activity, and concluding with a formal session termination and data archival. The specific actions within each stage are detailed below.

Creating a session: The teacher can create a session using a dedicated menu that allows configuring the workspace (choosing the room, furniture, interactive whiteboard background color etc.), selecting participants or a participant list, and the subject. Session settings can be saved and reused, with a “Load settings” option.

Session initialization: Before allowing participants to join the virtual classroom, the teacher will have access to a subject-specific menu to bring into the virtual space the materials and objects needed for the activity - such as graph generators, geometric shapes, maps, electrical circuits, and more.

In-session work: Once the teacher allows participants to connect, they can begin the activity without any additional actions. This makes it easy to understand and use.

Ending a session: This involves closing participant connections (they will be automatically disconnected from the session) and automatically storing in the database both the materials generated during the session and the statistical data of the ended session (video recording, connection duration, number of specific actions, list of participants etc.).

3.2 Steps for Creating an Interactive Lesson Using the Framework

Developing and integrating a new interactive lesson is a straightforward process managed in five distinct phases. This subsection outlines the steps required for a teacher to select a subject, create or customize an experiment, test its functionality, and formally integrate the new application into their lesson planning.

1. Selecting the subject

The teacher begins by choosing the desired subject from the suite offered by the framework. The framework includes a wide range of disciplines - from mathematics, physics, and chemistry to Romanian language, history, and music. Selecting the subject allows for the organization and grouping of experiments or training activities relevant to that field.

2. Choosing or creating an experiment

Once the desired subject has been confirmed, the teacher selects one of the existing experiments available in the platform’s library. Alternatively, they may choose to create entirely new content using a default template, which provides a customizable base structure for lesson development.

3. Customizing the content

In the third step, the teacher makes the necessary adjustments to adapt the lesson to the current educational curriculum or to the specific needs of their students. This customization ensures that the lesson is both pedagogically appropriate and engaging.

4. Testing the lesson

An essential stage in the development of any educational project is testing. During this phase, the teacher checks that all components function correctly and that the application is compatible with different devices and platforms. Any detected errors can also be corrected at this stage.

5. Integrating the lesson into the curriculum

The final step involves updating the existing educational curriculum to integrate the newly created interactive application. The teacher can adjust lesson planning so that the application fits harmoniously within the school's structure and learning objectives.

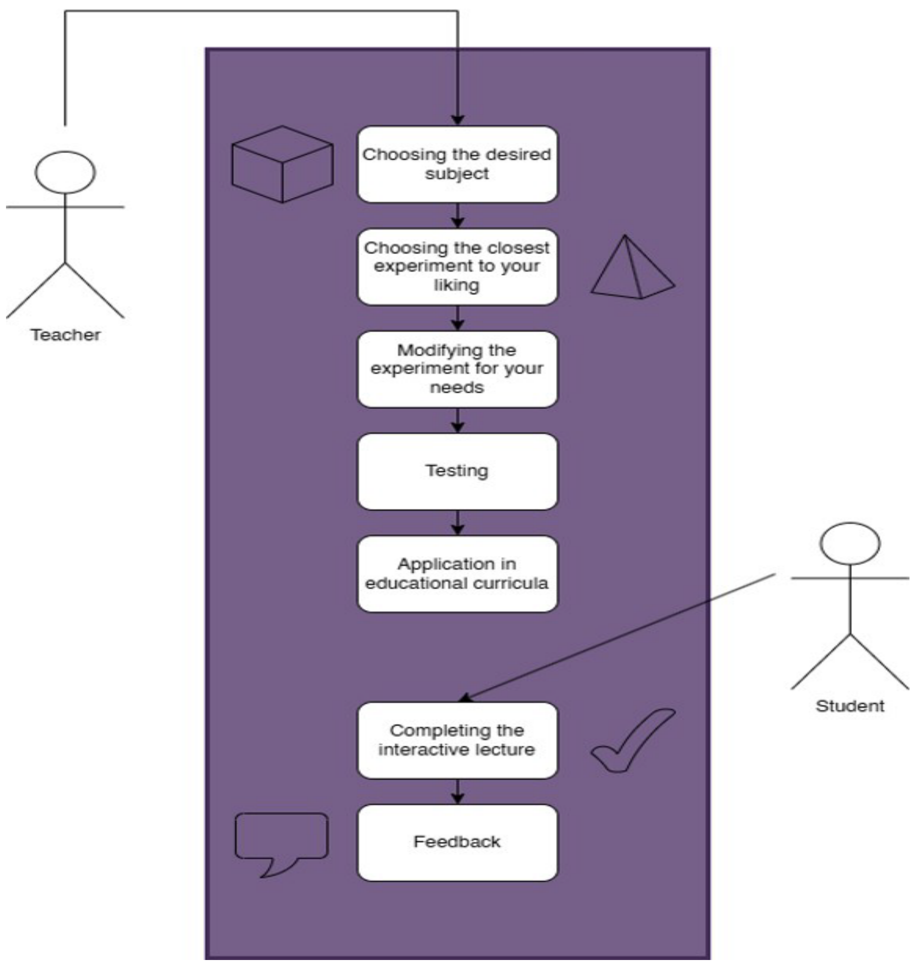

Figure 1. Proposed Architecture

4. Use Cases

This section presents a series of use cases across different courses to demonstrate the practical applicability of the proposed framework. It illustrates how teachers can use the game engine-based framework to create interactive lessons. Through these examples, the flexibility, accessibility, and pedagogical potential of the proposed framework are highlighted.

Biology Lessons: In biology classes, students can explore an interactive 3D model of the human circulatory system, navigating through arteries and veins as if traveling inside the body. Teachers can trigger explanatory animations to visualize key biological processes, such as oxygen exchange or blood flow through the heart chambers. This immersive approach transforms abstract concepts into tangible experiences, fostering deeper understanding and engagement.

Physics Lessons: For physics, the framework enables real-time simulations of fundamental laws of motion, gravity, and energy transfer. Students can modify parameters such as mass, force, or velocity, and immediately observe the resulting effects in a virtual environment. This interactive experimentation allows learners to test hypotheses safely, without the constraints or risks associated with real-world experiments, while reinforcing theoretical knowledge through direct application.

Mathematics Lessons: In mathematics, teachers can design visual and interactive representations of equations and geometric functions in a simulated 3D environment. Students can manipulate graphs dynamically - rotating surfaces, changing variables, and observing the effects of transformations in real time. This visual learning approach helps bridge the gap between abstract mathematical concepts and their spatial or functional interpretations.

Chemistry Lessons: In chemistry courses, students can perform virtual experiments in a 3D laboratory, mixing compounds, observing molecular reactions, and analyzing outcomes without exposure to hazardous materials. The system can simulate chemical reactions and display molecular structures at different stages, helping students visualize microscopic processes that are otherwise impossible to observe directly. This not only ensures safety but also enhances conceptual understanding through guided interactivity.

5. Conclusions

This paper has proposed a framework for creating interactive educational applications using powerful game engines like Unity and Unreal Engine. Unlike traditional educational tools, which often constrain teachers to pre-existing templates or content, the proposed system prioritizes educator autonomy and lesson customization, allowing teachers to build immersive, subject-specific experiences that align directly with their curriculum needs. By leveraging the graphical and modular capabilities of game engines, the framework enables the development of highly engaging and visually rich lessons. Its user-friendly drag-and-drop interface supports teachers with little or no programming experience, while its integrated data tracking and feedback mechanisms allow for real-time monitoring of student performance and continuous refinement of instructional strategies.

The application of immersive technologies in education, particularly in STEM subjects, has shown considerable promise in improving student engagement, motivation, and retention. This framework builds on those findings by enabling teachers to use such technologies more easily and effectively, directly integrating them into classroom practice and in the curriculum.

References

Dutescu, R., Apostol, M., Mocanu, I., & Barbu, M. (2025). Virtual Reality in training: A multidimensional analysis of technologies, challenges and paradigms. *Proceedings of the International Conference on Virtual Learning*, *20*, pp. 129–137. https://doi.org/10.58503/icvl-v20y202511.

Organisation for Economic Co-operation and Development. (2025). *Education at a Glance 2025: Romania*. OECD Publishing. https://www.oecd.org/en/publications/education-at-a-glance-2025_1a3543e2-en/romania_e364188c-en.html.

Paul, P. S., Goon, S., & Bhattacharya, A. (2012). History and comparative study of modern game engines. *International Journal of Advanced Computer and Mathematical Sciences*, *3*(2), pp. 245–249.

Salama, R., & Elsayed, M. (2021). A live comparison between Unity and Unreal game engines. *Global Journal of Information Technology: Emerging Technologies*, *11*(1), pp. 01–07. https://doi.org/10.18844/gjit.v11i1.5288.

Sobota, B., & Pietriková, E. (2023). The Role of Game Engines in Game Development and Teaching. In P. K. Zarzycki (Ed.), *Game Engines*. IntechOpen. https://doi.org/10.5772/intechopen.1002257.

Țoc, S. (2024). *Starea învățământului în România. O pledoarie pentru întărirea caracterului public și incluziv al educației*. Friedrich-Ebert-Stiftung. https://romania.fes.de/ro/e/starea-invatamantului-in-romania-o-pledoarie-pentru-intarirea-caracterului-public-si-incluziv-al-educatiei-de-sebastian-toc.html.