# Reusable Unity-based Puzzles for the Escapp Educational Escape Room Platform

Maja Cosic[1[/]], Sonsoles López-Pernas[2[0000-0002-9621-1392]], Aldo Gordillo[3[0000-0001-9785-4827]], Enrique Barra[3[0000-0001-9532-8962]], Alexandra Santamaría-Urbieta[4[0000-0003-0935-0616]], Milos Kostic[1[0009-0005-0912-9518]], Miljan Milosevic[1,5[0000-0003-3789-2404]]

[1] Belgrade Metropolitan University, 11000 Belgrade, Serbia
[2] University of Eastern Finland, Joensuu 80130, Finland
[3] Universidad Politécnica de Madrid, 28040 Madrid, Spain,
[4] Universidad Internacional de La Rioja, 26006 Logroño, La Rioja, Spain
[5] Institute for Information Technologies, 34000 Kragujevac, Serbia
miljan.milosevic@metropolitan.ac.rs

**Abstract.** Educators have shown growing interest in escape rooms, as they engage students while fostering teamwork, leadership, creativity, and communication. Consequently, educational escape rooms are emerging as a novel form of learning activity that seeks to enhance students' learning through immersive and highly engaging experiences. This paper presents a novel approach to designing reusable educational puzzles within Unity and integrating them seamlessly into the Escapp platform for creating educational escape rooms, as part of the IGLUE project. We describe the workflow that enables educators to configure customizable puzzle instances via Escapp's interface—each instance accessible through a unique URL that exposes configuration data in JSON format. Four distinct Unity puzzles have been developed according to a flexible, parameter-driven architecture, enabling them to adapt dynamically based on educator-defined configurations. During runtime, the Unity game fetches the JSON data to instantiate and render the puzzle, while solution validation occurs via secure API communication with Escapp—puzzle attempts are sent back to the platform for validation, and correct solutions trigger automatic notifications of completion. This bidirectional integration supports modularity, reusability, and real-time feedback and learning analytics, enhancing the design of escape-room-style learning scenarios in a scalable and extensible way.

**Keywords:** educational escape rooms, Escapp platform, reusable puzzles, game-based learning.

## 1 Introduction

In recent years, game-based learning has gained increasing attention as an effective approach to enhance learner engagement and motivation in digital education. Among the many forms of game-based learning, educational escape rooms have emerged as a particularly promising method, as they combine immersive storytelling with problem-solving activities that require creativity, critical thinking, and collaboration [1]. Educational escape rooms situate learners in scenarios where they must solve

Research Paper
DOI: https://doi.org/10.46793/eLearning2025.225C
Part of ISBN: 978-86-89755-37-4

challenges to progress and offer opportunities to reinforce knowledge acquisition while simultaneously fostering transversal skills such as teamwork and communication [2].

Though leisure escape rooms —those designed primarily for entertainment— have been mostly physical, the digitalization of the concept has contributed to the adoption of educational escape rooms, since they can be applied in both classroom and remote learning contexts, provide multiple levels of difficulty, and allow for easy reusability and scalability across different subject domains [2].

However, most existing digital educational escape rooms rely on static designs, which limits flexibility for educators and the possibility to adapt them to the peculiarities of their classroom [3]. The IGLUE project addresses this challenge through the Escapp platform [3], a web-based environment that allows educators to create and orchestrate escape rooms, configure puzzle parameters, and track student progress (https://escapp.es/). The contribution of the present study focuses on developing reusable Unity-based puzzles integrated within Escapp. Each puzzle is dynamically configured via JSON files generated by the platform and validated through API communication, ensuring adaptability and security. Four Unity puzzles have been implemented as proof-of-concept, demonstrating how reusable design can enhance flexibility and scalability of educational escape rooms.

The aim of this paper is to present our work on the design and integration of reusable Unity puzzles into the Escapp platform. We describe the educational and technical background of escape rooms, introduce the Escapp platform, detail the architecture for puzzle configuration and solution validation, and present four specific puzzle implementations. Finally, we discuss lessons learned and potential directions for future development of reusable and scalable digital escape rooms.

## 2 Background

### 2.1 Educational escape rooms

Educational escape rooms are a type of educational game that combines problem-solving, collaboration, and critical thinking within a narrative framework [1]. Unlike traditional escape rooms, which focus purely on entertainment, educational escape rooms are designed to meet specific pedagogical objectives [2]. Students are placed in immersive scenarios where they must solve puzzles and challenges in order to progress, thus reinforcing learning outcomes in a playful yet structured manner. Key benefits identified in the literature include increased learner engagement, development of transversal skills such as communication, collaboration, creativity, and critical thinking [4], [5], [6], [7], as well as the possibility of integrating domain-specific content in areas such as healthcare [8], or STEM [9]. Digital escape rooms further expand these possibilities by allowing both classroom and remote participation, adaptability to different difficulty levels, and the reuse of puzzle templates across multiple educational contexts [10].

A growing body of reviews illustrates the breadth of these applications. Early surveys highlighted escape rooms as promising tools to enhance student motivation and engagement across disciplines [11], while later analyses examined design principles linking puzzles, narratives, and learning goals [1]. Research has also emphasized the importance of curricular integration and 21st-century skill development, though with limited theoretical grounding [12], a gap that has been noted in recent scholarship [13]. Empirical syntheses provide further evidence: meta-analyses report that educational escape rooms are generally effective across levels of education and modes of delivery [2], while domain-specific reviews in STEM [9] and medical education [14] indicate both motivational benefits and measurable knowledge gains.

At the same time, educational escape rooms are challenging to design, create, and conduct. Developing pedagogically sound activities requires instructors to construct an engaging narrative alongside a sequence of puzzles that are logically connected, employ meaningful game mechanics, and target the desired learning outcomes [15]. Such design processes demand imagination, creativity, and a strong pedagogical orientation. Beyond the conceptual stage, the practical implementation of puzzles and interactive elements can be difficult and time-consuming, particularly for instructors who lack technical expertise. This tension between the pedagogical potential of escape rooms and the practical challenges of their design motivates the need for platforms that lower barriers to creation and provide reusable puzzle templates.

### 2.2 The Escapp platform

Escapp is a web-based platform specifically developed to support the creation, orchestration, and execution of educational escape rooms [3]. It enables educators to design escape room scenarios without requiring advanced technical skills by offering a catalog of configurable puzzles, a scenario editor, and tools for monitoring student progress. One of Escapp's strengths is its modular architecture, which allows external applications—such as Unity-based puzzles—to be seamlessly integrated into escape room narratives. Each puzzle can be parameterized through a JSON configuration generated within Escapp's interface, ensuring flexibility and reusability. In addition, Escapp provides APIs for solution verification and event notifications, making it possible to synchronize game logic between the Unity puzzles and the platform. The tight coupling between customization, gameplay, and progress tracking makes Escapp a powerful tool for educators who aim to leverage game-based learning in both face-to-face and online environments (see Fig. 1).


**Fig. 1.** Screenshot of the Escapp platform team interface.

### 2.3 The IGLUE project

Educational escape rooms often face limitations such as the static nature of puzzles, restricted opportunities for reuse, and the high technical effort required for customization. Addressing this limitation requires a framework that supports the reusability and configurability of puzzles. This is the core idea behind the IGLUE project, in which the Escapp platform has been developed as a web-based environment to design, orchestrate, and manage educational escape rooms. Escapp enables educators to create customized puzzles and scenarios without advanced technical knowledge, while also providing APIs for solution verification and player progress tracking.

Within this project, our contribution focuses on the development of reusable Unity-based puzzles that are directly integrated with the Escapp platform. Each puzzle is configured dynamically through a JSON file generated by Escapp and validated via API communication. In this way, puzzles can be easily adapted to different educational contexts while ensuring security and modularity. Four Unity puzzles have been designed and implemented as part of this approach, serving as concrete examples of how a reusable architecture can enhance the effectiveness of digital escape rooms.

## 3 Creating Reusable Unity-based Puzzles for Escapp

### 3.1 Reusable Puzzle Design in Unity

The development of reusable puzzles in Unity followed a modular and parameter-driven approach, ensuring that a single puzzle template could be adapted to multiple learning contexts. Instead of embedding fixed content or solutions directly into the Unity code, puzzles were designed to load their configuration at runtime. This was achieved through a JSON-based mechanism, where external parameters define the behavior and appearance of each puzzle instance.

The configuration file may contain elements such as the puzzle's textual content, difficulty level, number of interactive components, or hints to be displayed to learners. Once the Unity puzzle is launched, it retrieves the JSON file from a unique URL generated by the Escapp platform, parses the data, and instantiates the puzzle accordingly. This approach separates the pedagogical content, defined by the educator, from the technical implementation, designed by the developer.

### 3.2 Integration Between Unity Puzzles and Escapp

The integration between Unity puzzles and the Escapp platform is based on a two-way communication model. In the first direction, JSON → Unity, the puzzle reads the configuration file generated by Escapp. This ensures that when a student launches a puzzle within an escape room scenario, the Unity application automatically adapts to the parameters chosen by the educator. The configuration stage provides flexibility while maintaining a consistent workflow inside the platform.

In the opposite direction, Unity → Escapp, the interaction focuses on solution validation and progress tracking. When a learner attempts to solve a puzzle, Unity does not perform the validation internally. Instead, it sends the proposed solution to Escapp's API, which checks correctness and returns a boolean result. If the answer is valid, Unity notifies Escapp of puzzle completion, enabling the platform to register progress and orchestrate the sequence of the escape room. This mechanism—implemented through REST calls and potentially WebSocket communication—ensures security, prevents unauthorized access to solutions, and guarantees that the educator retains full control over the learning flow.

### 3.3 About reusable puzzles

In this section we present the reusable puzzles created for the purpose of the IGLUE project: Caesar cipher, Bomb, Radio and Safebox.

**Caesar cipher puzzle.** The Caesar cipher puzzle relies on a JSON configuration file that specifies two parameters: a string representing the original code and an integer that defines the shift value. Based on this input, the Unity application generates a cipher that players are required to decode.

```
{
  "code": "ABC 123",
  "shift": -2
}
```

**Fig. 2.** Code of Caesar cipher.

For example, if the code is ABC and the shift is +2, the encrypted output becomes CDE. Conversely, a negative shift produces a reverse transformation; with the same code ABC and a shift of –2, the resulting cipher is YZA.

The game interface is divided into two sections: the upper part displays the encoded text, while the lower part provides the input area for players' attempts, Fig. 3. Input is performed through a virtual keyboard positioned below the screen, with additional options to reset or correct entries via backspace. The system also allows users to select specific letter positions when constructing their solution. The visual design emulates a retro-style handheld device, reinforcing the puzzle's mysterious atmosphere. All graphical elements were hand-drawn to enhance usability and player immersion. The development process followed a two-step approach: an initial prototype was created to validate interaction and gameplay mechanics, which was later refined into the final version implemented in Unity.


**Fig. 3.** Caesar cipher puzzle made in Unity.

**Bomb puzzle.** The bomb puzzle is a 3D puzzle that consists of four distinct smaller puzzles, each contributing to the overall puzzle system, Fig. 4a. To successfully complete the puzzle the player needs to stop the bomb timer by pressing the stop button.


a b

**Fig. 4.** Bomb puzzle example: a) Overall puzzle system, b) Sliders puzzle.

The stop button is covered by a plastic lid, which can only be lifted after the player completes all the smaller puzzles. For the purpose of connecting the puzzle to Escapp, the smaller puzzles must be solved in a specific order. The four smaller puzzles in order are: the sliders puzzle, color button puzzle, wire cut puzzle and number pad puzzle. Each of the puzzles has a lightbulb that indicates if the puzzle is active. If the lightbulb shines yellow it means that the puzzle has been activated, but if the lightbulb is just gray it means that the puzzle is inactive. When the puzzle is completed correctly the lightbulb will turn green. However the player makes a mistake the lightbulb will blink red and reset.

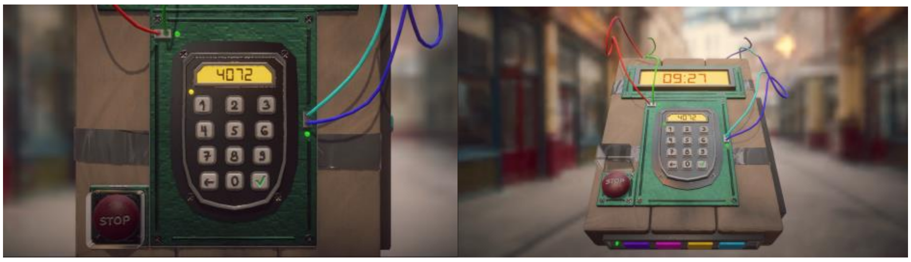
The first puzzle to be completed is the sliders puzzle, Fig. 4b. This mini puzzle features two separate sliders positioned on the sides of the bomb. The player must match each of their values to the corresponding values entered in Escapp. To change the value, the player needs to grab the slider's handle and drag it to the correct position.


a b

**Fig. 5.** Bomb puzzle example: a) Color button puzzle, b) Wire cut puzzle.

The next mini puzzle is the color button puzzle, Fig. 5a. This puzzle consists of four differently colored buttons that are placed at the bottom part of the bomb. The buttons themselves need to be pressed in a predefined sequence. The order of the buttons and their color are entered through Escapp. Furthermore, the button sequence can repeat the same button multiple times or exclude some buttons. The length of the sequence can also be changed to fit each escape room's purposes. If the player makes a mistake during the sequence, it will reset and the player will have to start from the beginning of the sequence.

The third mini puzzle is the wire cut puzzle, Fig. 5c. This puzzle consists of four wires that connect the front of the bomb to its top. Each wire has a specific color and must be cut in a predefined order, which can be entered through Escapp. Since wires are being cut, each wire can only appear once during the sequence, and the sequence can only range from 1 to 4. When the player hovers the mouse over a wire, the cursor changes to a wire cutter icon to simulate cutting. Once a wire is cut, it will visually appear cut in the game. If the player cuts a wire in the wrong order, the sequence resets, and they must start again from the beginning.




a b

**Fig. 6.** Bomb puzzle example: a) Number pad puzzle, b) Completed bomb puzzle.

The last mini puzzle is the number pad puzzle, Fig. 6a. This puzzle consists of a number pad with digits, a backspace, and an enter button, located at the center of the bomb. The player needs to enter the correct number code, which can be set through Escapp. Once the player has inputted the code, they must press enter to check if it is correct. If the code is incorrect, the player will need to reenter a new code.

Finally, once all of the mini puzzles are completed the plastic lid covering the button will lift up and the player can press the stop button to stop the timer, Fig. 6b. With that the player has successfully completed the whole puzzle.

**Safebox puzzle.** The safebox puzzle is a 3D puzzle that requires the player to enter a code using a combination lock, Fig. 7. The lock can be spun both clockwise and counterclockwise to the desired digit. Pressing the key Q on the keyboard turns the rotation counterclockwise, while pressing the key E turns the rotation clockwise. For easier use, the player is using a circular slider to turn the handle to the correct number. The code for the safebox is entered through Escapp, as well as the direction of the lock. Once the correct code has been entered, the player can click on the safebox handle to open it. If the player enters a wrong digit while inputting the code, the lock resets and the player must start over. Inside the safebox there is either a note or an image that provides an additional clue or story element to the game. The content inside can be added through Escapp, including whether it should be a note or an image.

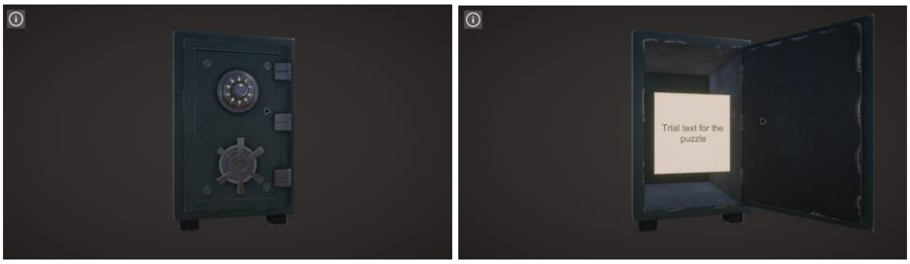



**Fig. 7.** Safebox puzzle example: a) Locked safebox, b) Open safebox.

**Radio puzzle.** The radio puzzle is a 3D puzzle that requires the player to adjust the volume and the frequency of a sound to match the reference wave, Fig. 8. In the first place, the player needs to turn on the radio and then use the volume and frequency handles to adjust the sound itself. The radio screen displays the current volume and frequency values of the sound. In addition to that, the end goal of the puzzle is to align the sound wave with the reference wave shown behind it. Just like in the other puzzles, the correct values and the sound itself should be entered trough Escapp. Once the sound is matched correctly, the puzzle is completed. The player can readjust the audio as much as they need to until they have found the correct values.

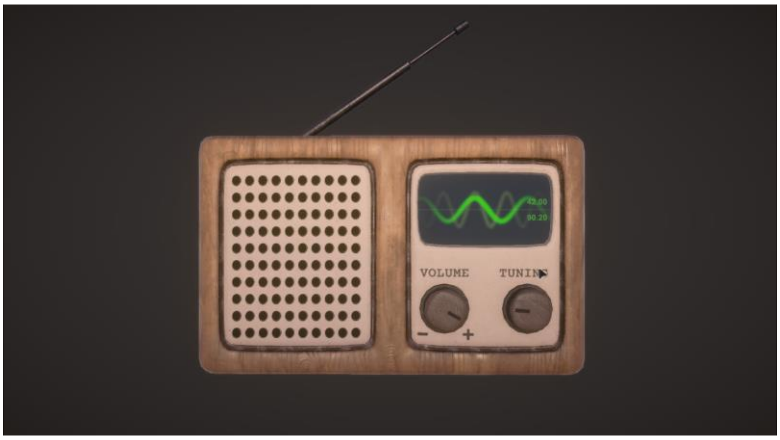



**Fig. 8.** Radio puzzle made in Unity.

## Discussion and conclusion

This work shows how reusable Unity-based puzzles can be integrated into the Escapp platform to enrich educational escape rooms. By using JSON configuration files, puzzle content is separated from logic, allowing educators to easily adapt puzzles to different topics without additional programming.

The integration ensures secure solution validation through Escapp's API and smooth tracking of student progress. This modular design reduces development effort, supports scalability, and encourages puzzle reuse across contexts. Overall, the approach provides a flexible and reusable framework that strengthens the role of digital escape rooms in e-learning.

**Acknowledgments.** This work has been co-funded by the European Union through the Erasmus+ Programme as part of the project "Implementation of game-based learning using escape rooms" (IGLUE) (Reference: 2024-1-ES01-KA220-HED-000256356). Co-funded by the European Union. The opinions and viewpoints expressed are solely those of the authors and do not necessarily reflect those of the European Union or the Spanish Service for the Internationalization of Education (SEPIE). Neither the European Union nor the granting authority can be held responsible for them.

**Disclosure of Interests.** The authors have no competing interests to declare that are relevant to the content of this article.

## References

1. Veldkamp, A., van de Grint, L., Knippels, M.-C. P. J., & van Joolingen, W. R. (2020). Escape education: A systematic review on escape rooms in education. Edu-cational Research Review, 31, 100364. https://doi.org/10.1016/j.edurev.2020.100364
2. López-Pernas, S. (2024). Educational escape rooms are effective learning activities across educational levels and contexts: A meta-analysis. IEEE Transactions on Learning Technologies, 17, 711–724. https://doi.org/10.1109/tlt.2023.3328913.

3. Lopez-Pernas, S., Gordillo, A., Barra, E., & Quemada, J. (2021). Escapp: A web platform for conducting educational escape rooms. IEEE Access: Practical Innova-tions, Open Solutions, 9, 38062–38077. https://doi.org/10.1109/access.2021.3063711
4. Pan, R., Lo, H., & Neustaedter, C. (2017). Collaboration, awareness, and communi-cation in real-life escape rooms. Proceedings of the 2017 Conference on Designing Interactive Systems - DIS '17. the 2017 Conference, Edinburgh, United Kingdom. https://doi.org/10.1145/3064663.3064767
5. Warmelink, H., Mayer, I., Weber, J., Heijligers, B., Haggis, M., Peters, E., & Louwerse, M. (2017, October 15). AMELIO. Extended Abstracts Publication of the Annual Symposium on Computer-Human Interaction in Play. CHI PLAY '17: The annual symposium on Computer-Human Interaction in Play, Amsterdam The Neth-erlands. https://doi.org/10.1145/3130859.3131436
6. Williams, P. (2018, March). Using escape room-like puzzles to teach undergraduate students effective and efficient group process skills. 2018 IEEE Integrated STEM Education Conference (ISEC). 2018 IEEE Integrated STEM Education Conference (ISEC), Princeton, NJ, USA. https://doi.org/10.1109/isecon.2018.8340495
7. Wu, C., Wagenschutz, H., & Hein, J. (2018). Promoting leadership and teamwork development through Escape Rooms. Medical Education, 52(5), 561–562. https://doi.org/10.1111/medu.13557.
8. Guckian, J., Eveson, L., & May, H. (2020). The great escape? The rise of the escape room in medical education. Future Healthcare Journal, 7(2), 112–115. https://doi.org/10.7861/fhj.2020-0032
9. Lathwesen, C., & Belova, N. (2021). Escape Rooms in STEM Teaching and Learn-ing—Prospective Field or Declining Trend? A Literature Review. Education Scienc-es, 11(6), 308. https://doi.org/10.3390/educsci11060308.
10. Makri, A., Vlachopoulos, D., & Martina, R. A. (2021). Digital Escape Rooms as In-novative Pedagogical Tools in Education: A Systematic Literature Review. Sustaina-bility: Science Practice and Policy, 13(8), 4587. https://doi.org/10.3390/su13084587
11. Fotaris, P., & Mastoras, T. (2019). Escape rooms for learning: A systematic review. In Lars Elbaek, Gunver Majgaard, Andrea Valente, Saifuddin Khalid (Ed.), Proceed-ings of the 13th European Conference on Games Based Learning (ECGBL 2019) (pp. 235–243). Academic Conferences and Publishing International Limited.
12. Taraldsen, L. H., Haara, F. O., Lysne, M. S., Jensen, P. R., & Jenssen, E. S. (2022). A review on use of escape rooms in education – touching the void. Education In-quiry, 13(2), 169–184. https://doi.org/10.1080/20004508.2020.1860284
13. Vorderobermeier, A., Abel, J., & Sailer, M. (2024). Theoretical foundations and ap-proaches in research on educational escape rooms: A systematic review. Education-al Research ReviewVolume 44, August 2024, 100625.
14. Reinkemeyer, E. A., Chrisman, M., & Patel, S. E. (2022). Escape rooms in nursing education: An integrative review of their use, outcomes, and barriers to implementation. Nurse Education Today, 119(105571), 105571. https://doi.org/10.1016/j.nedt.2022.105571
15. López-Pernas, S., Gordillo, A., Barra, E., & Quemada, J. (2019a). Examining the Use of an Educational Escape Room for Teaching Programming in a Higher Educa-tion Setting. IEEE Access, 7, 31723–31737. https://doi.org/10.1109/access.2019.2902976.