

# **Escape This Class**

## **Nonthapat Phoungrod (Nod, 64991037)**

## **Pathawee Sreepromma (Project, 64991040)**

### **Abstract**

*Escape This Class is an educational VR game designed to enhance learning by letting users interact with and apply classroom concepts in an escape room like environment that the player will have to use their knowledge that they've learn from the class to try and solve the puzzles inside and "Escape This Class". The game works with any virtual reality headset and requires an internet connection for download. By integrating 3D models, puzzles, and animations, it helps students visualize and practice topics from subjects like programming, chemistry, physics and several other subjects. Players can also choose a difficulty level that matches their knowledge, making the game suitable for learners at different stages.*

### **Keywords**

virtual reality (VR), educational technology, puzzle-solving games, immersive learning, student engagement

### **1. Introduction**

In today's world, various technologies have advanced rapidly, but education has not developed as broadly. Some subjects have become more difficult to understand due to the increasing complexity, making it hard to visualize concepts. To enhance and develop education, we are utilizing VR technology to create an educational experience in the form of a puzzle-solving game. This approach allows learners to engage in practical, immersive experiences through virtual reality. Additionally, learning through games helps make lessons more visual and engaging, leading to increased student interest and enjoyment in learning.

#### Purpose of developing this game

- 1) To take advantage of virtual reality technology (VR) for educational purposes and program creation.
- 2) To create using the C# language for advantages.
- 3) The ability for the player to apply their past knowledge from classroom instruction to simulation scenarios.
- 4) To strengthen the player's ability in observation, problem-solving, and forming connections.

- 5) For the player to reinforce and review lessons through gameplay.

### **2. Methods**

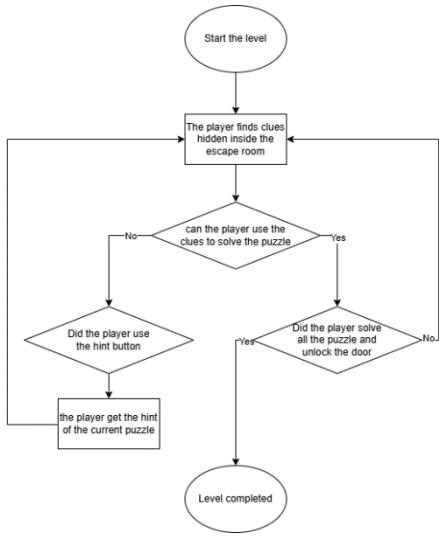
#### Tools and software used to develop this project

- 1) PC or laptop
- 2) Meta Quest 2 (Virtual Reality headset)
- 3) Blender (3D modeling software)
- 4) Visual Studio 2022 (coding software)
- 5) Unity (game engine)

#### Technology, Technology used to develop this project

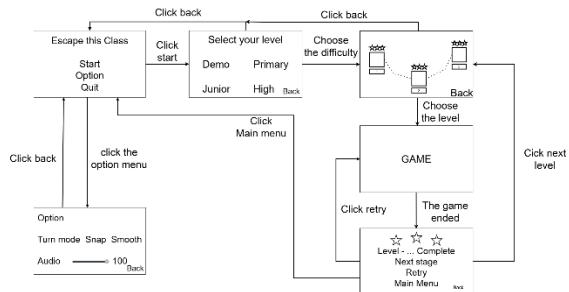
- 1) Virtual Reality (VR)  
a computer-generated environment that simulates and conveys the feeling and experience of being in a virtual world. Viewing this virtual reality requires a specialized device that receives signals from a computer to immerse the user in the simulated environment.
- 2) XR Interaction Toolkit  
An advanced user interaction system specifically designed for creating games within VR (Virtual Reality) and AR (Augmented Reality) environments. It includes a framework for creating 3D visuals and responding to user interfaces (UI) based on user input. The key components of the XR Interaction Toolkit are the set of interactors, the set of interactable elements, and the response controllers that manage the interaction between them. This system can be used to create object manipulation systems in the virtual world and to interact with user interfaces (UI).

### Overall software design



**Figure 1. Flow charts represent the logic of Escape This Class level in general**

Figure 1 presents a schematic of the flow charts that illustrate the general functioning of each level within the game. Upon entering a level, the player must search for clues scattered throughout the room. If the player is unable to locate a clue, they can click the clue button, which will provide a hint. Conversely, if they successfully find and use the clues to solve all the puzzles, they will receive a key to unlock the door.



**Figure 2. Model of Menus in Escape This Class.**

Figure 2 is the representation of each menu inside the game, which will consist of the main menu, a difficulty selection menu, an option menu, and a level selection menu.

### 3. Results

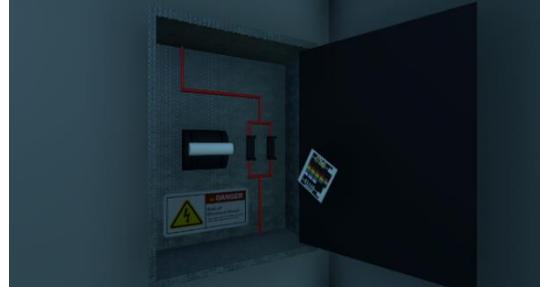
When entering the game, you awaken in a mysterious classroom where all doors and exit routes are locked, leaving you trapped. Your main objective is to escape this classroom and uncover its mysteries. The game features four primary puzzles: the electrical puzzle, programming puzzle, math puzzle, and chemistry puzzle. These puzzles must be solved in a specific order. Each puzzle consists of multiple functions to ensure proper operation. For clarity, each puzzle will be detailed and explained individually.



**Figure 3. Overall scenery of the game**

Figure 3 shows the overall scenery of the only level currently in Escape this class.

#### 1. Electric Puzzle



**Figure 4. Electric box in electric puzzle**

Figure 4 shows the electrical box, which is currently missing a resistor, resulting in a power outage. It awaits the player's intervention to resolve the issue.



**Figure 5. Resistor in electric puzzle**

Figure 5 displays the resistor that the player needs to insert into the electrical box to resolve the power outage issue.

As the player examines the classroom, they will notice that the electricity is currently turned off. Upon inspecting the electrical box, the player will find that a resistor is missing. Using the specifications provided for the electrical box, the player must select and insert the correct resistor to restore power.

- **FirstResisterTest.cs**

This script verifies whether the first resistor has been correctly inserted into the socket by checking the object tag of the resistor that collides with the socket.

- **SecondResisterTest.cs**

This script performs a function similar to FirstResisterTest.cs, but it checks the second resistor instead.

- **LeverText.cs**

This function determines whether the switch on the electric box is in the 'on' position.

- **LightControlPanel.cs**

This function checks if all conditions for the puzzle have been met both resistors must be correctly inserted, and the switch must be turned on. If all conditions are satisfied, it activates the class right and makes the next puzzle accessible.

## 2. Programming puzzle



**Figure 6. Laptop used in programming puzzle**

Figure 6 shows the locked laptop after the power outage issue is resolved.



**Figure 7. Programming puzzle**

Figure 7 presents some parts of the programming puzzle.

The programming puzzle is the second challenge to complete. Upon entering the room after solving the electrical puzzle, you will notice that the laptop in the center of the classroom is now operational. When you attempt to log in, you'll encounter an option to engage in a Pseudocode puzzle to unlock the computer. This puzzle will consist of fundamental programming concepts, including the use of loops and the print function.

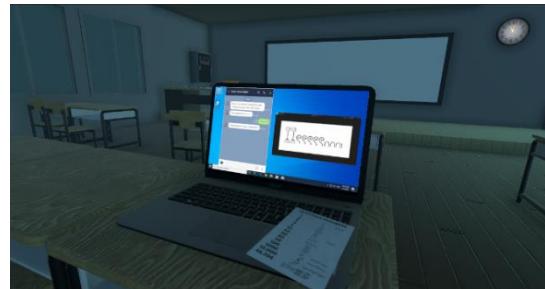
- **MainPuzzle.cs**

This script allows players to input specific code snippets through UI buttons, stores those snippets, and has the infrastructure for running and clearing that code.

- **ButtonScript.cs**

The script is designed to control the visibility of a GameObject based on user interactions and create a functional programming platform for user to code.

## 3.Math puzzle



**Figure 8. hieroglyphic characters in the math puzzle**

Figure 8 illustrates the hieroglyphic characters along with a reference sheet that provides the meaning of each character. The player must translate these characters to solve the puzzle.

Once you solve the programming puzzle, you will discover some hieroglyphic characters that need to be translated into numbers. This translation will provide the password required to unlock the locker at the back of the classroom.

- **displayPassword.cs**

The script is designed to manage a password display and trigger an animation for unlocking a door when the Unlocked variable is set to true

- **Mathlockerkeypad.cs**

This script allows for the entry of a password via a keypad, clearing input, and handling incorrect attempts with an animation. When the correct password is entered, it sets a flag to unlock an associated mechanism.

#### 4.Chemistry puzzle



**Figure 9. Revealed hidden smelter room**

Figure 9 shows the revealed passageway to the hidden smelter room after pressed the hidden button in the locker.



**Figure 10. Smelting room**

Figure 10 presents the hidden smelter room, which the player needs to access to complete the final puzzle and create the key. The room contains the smelter, various materials on the table, and molds for shaping the key.

After solving the math puzzle, the player receives a guide for making brass. They'll find a red button in a locker that, when pressed, lifts another locker to reveal a hidden smelter. The player must create brass and forge a key to escape the classroom.

- **AmountColliderCheck.cs**

This script simulates the process of melting materials in a furnace based on their amounts and proportions. When the right conditions are met (The ratio between copper and zinc is correct according to the brass making guide provided to user from earlier puzzle) while the container is inside the furnace, it triggers a melting reaction that includes visual effects using unity particle system.

- **ChemistryRoomopen.cs**

The script allows for the interaction with a door leading to a chemistry room, ensuring that the door can only be opened if certain conditions are met (The earlier puzzle is cleared).

- **KeyMaking.cs**

The script is designed for a key-making process in a game where liquid metal is poured into a mold to create a key using to escape the classroom.

- **Pour.cs**

The script simulates a pouring effect using particles based on the orientation of the GameObject.

When the object is tilted downward the pouring effect is activated. When tilted upward, the effect stops.

- **SubstanceScript.cs**

This script is responsible for tracking the amounts of zinc and copper when they interact with a designated object.

## Discussion

Users have expressed two main concerns: first, the limited time available to complete the puzzle and the fact that it only has one level, which restricts its appeal to a specific group of players. Additionally, access to the game is currently challenging, as it requires contacting the developer, and the overall experience lacks immersion. To address these issues, we plan to implement the following improvements:

- Develop additional levels to provide users with a greater variety of puzzles related to multiple lessons.
- Develop educational level inputs to recommend puzzles that are appropriately tailored to users' skill levels.
- Implement educational requirement tags for each level, which will be displayed on the level selection screen.
- We will include a section where users can recommend lessons that can be used for future puzzles.
- Add interactive tutorials or hints for each puzzle to help new players understand mechanics without feeling overwhelmed.
- Integrate a storyline or narrative to engage players more deeply, providing context and motivation for solving puzzles.
- Adding cooperative or competitive multiplayer modes, allowing players to work together or challenge each other.
- Improve graphics and sound design to create a more immersive experience, including background music and sound effects that fit the theme.

- Upload the game to platforms such as Steam, Epic Games, or the Microsoft Store to improve accessibility for players.

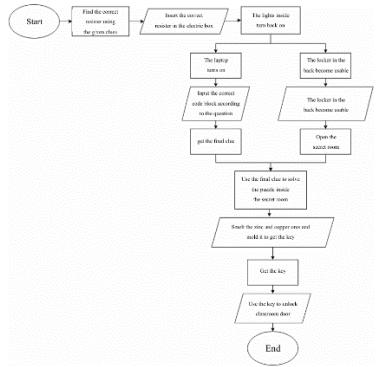
## Conclusion

This project aims to create an educational virtual reality game that works with VR headsets. It includes VR glasses, controllers, connectors for connecting VR, and computers that can run this game if they have enough RAM on them, can support using VR devices, and can connect to the Internet.

This educational effort aims to increase students' enjoyment of learning. They can also do more than just listen in order to improve the quality of their learning and gain a clearer understanding of the lectures. It inspires people to take a greater interest in their education. It might serve as a resource for going over previous lessons. The game's puzzles are created using teachings that are derived from classroom instruction. Through the use of the simulation program, players will be able to put the teachings they have learned in the classroom into practice, helping them to better grasp the material and to help them remember it. The knowledge gained

## References

- [1] Cosmina, Cosmina. (2018). A Glance into Virtual Reality Development Using Unity. *Informatica Economica*. 22. 14-22. 10.12948/issn14531305/22.3.2018.02.
- [2] Fang, Huang & Xiru, Yuan & Kun, Huang & Wenxin, Li & Haodong, Tian. (2021). Design and Application of VR Lab Based on Unity. *Journal of Physics: Conference Series*. 1982. 012167. 10.1088/1742-6596/1982/1/012167.
- [3] Meng, Xiangyu. (2019). Formal Analysis and Application of the New Mode of "VR+ Education". *ITM Web of Conferences*. 26. 01014. 10.1051/itmconf/20192601014.
- [4] Qin, Guohe & Gao, Liang. (2021). Application of VR Technology in Japanese Education. 10.1007/978-3-030-69999-4\_29.
- [5] XR Interaction Toolkit  
<<https://docs.unity3d.com/Packages/com.unity.xr.interaction.toolkit@2.2/manual/index.html> >



**Figure 10. Flow charts represent the logic of Escape This Class demo level**

Figure 10 presents a schematic of the flow charts that illustrate the general functioning of the demonstration level that we've develop which will consist of 4 main puzzles and the player will have to finished them accordingly in order to complete the level.