ESCAPE
An Escape Room Games

First Author
Andreas Hennings
Universität Regensburg
andreas.hennings@ur.de

Second Author
Sarah Brand
Universität Regensburg
sarah.brand@stud.uni-regensburg.de

CHI 2020 Extended Abstracts, April 25–30, 2020, Honolulu, HI, USA.

Abstract
In this paper we present Escape, an escape room game we developed for the Game Engineering course at Regensburg University. After having a brief look at Escape Room games in general, we will focus on the designing of the game and introduce its key components. We will conclude this paper with some notes on testing on improving the game.

Author Keywords
Game, Escape Room, Computer, Game Development

CSS Concepts
Software and its engineering

Introduction
In recent years, escape room games have gained increasing popularity. They are considered a subgenre of action games, in which a player has to escape from a certain location, mostly a room or an apartment. Often, the player has to find items or solve puzzles to achieve the mission.

History
The first game to introduce this concept is generally considered to be *Crimson Room* by Toshimitsu Takagi, released 2004. Later, they were devoloped as free browser games or for mobile platforms. Today, many escape room games are browser games or conducted live in the real world. Recently, there's even been a movie produced dealing with this setting (Escape Room, 2019), indicating that the topic is gaining public interest.

Related Work
As mentioned above, *Crimson Room* was the first game to introduce the concept of an escape game. Other examples are "Viridian Room", "MOTAS", and "Droom"[i]. In the scientific field, escape room games have been developed for educational purposes, like escapED[ii] for students, the Nursing Escape Room game for nurses[iii]or to explore the possibilities for teambuilding in a mixed-reality environment[iv]

Game Development

Time Constraints and levels
In our version of an escape room game, we had to limit the time a player can take to find the key to open the door to 5 minutes to meet our requirements. Since we wanted to offer potential customers more than just a 5 minute game, we decided to design the game in a way, that additional levels can be added very easily. All it takes to create new levels is to duplicate a key and hide it somewhere the game manager (see below) will recognize all hidden keys and enable the player to start new levels, as long as there are hidden keys in the game. As a proof of concept, we hid eleven keys in the game.

The Story
At the very beginning of the game, the player is informed that he / she has been locked inside an appartment (probably by some kind of sadistic madman) and a bomb is set to explode in five minutes. In order to escape, the player has to find a key to open the door to get out. To make things a little more difficult, the countdown of the bomb will be set to 10 seconds, as soon as the player has picked up the key. No more information is provided, except that the player can move and look around using ASDW and Space keys. In the background, the beeping of the bomb adds to the tension. Also, the player is deliberately left unclear about his / her situation: After a key is found, the player gets no information about how many times he / she will be sent back into the apartment and only be notified, after all keys have been found.

Architecture / Level layout

In order to make the game neither too easy nor to difficult, we had to carefully plan the layout of our game world. While a single room would have been too small to effectively hide the keys, a large house or even an open world scenario would have made it impossible for the player to find a key within 5 minutes and reach the door within 10 seconds after picking it up. This is supposed to add more tension to the gameplay, since the player has to really rush to the door to make it out within that timeframe. Also, the player needs to make sure, that no object is blocking the way to the door. Thus, we designed a small apartment, consisting of a living room, a bed room, a kitchen and a bath room. For the environment that can be seen, we chose a HDR image, to make the scenario a bit more realistic.

Interior and Objects

After finishing the architecture of our game, we found that it was merely impossible to get all of the furniture and objects needed for our game from the Unity Asset Store. Many items did not exist or were not free to use. Furthermore, we wanted our interior design to match and not look like being compiled of random stuff that would not really fit together. And finally, we wanted to make sure that the game could be published without violating any copyright laws. Therefore, with the exception of the palm tree in the living room and the fire in the fireplace (which were taken from the Unity Standard Assets), all objects were made by ourselves, using the Blender 3D Software. For materials, we chose to use PBR materials exclusively, which we got from CC0textures.

Physics and Object Sounds

All objects in the scenario can be divided into two groups: static and dynamic objects. While static objects (like the safe) can not be moved by the player, dynamic objects can – and have to, in order to find keys. While all objects have colliders to make them impenetrable by the player, only dynamic objects also have a rigidbody component. After extensive testing, we chose to give those a relatively low mass, since otherwise it wouldtake the player too much effort to push them around and make some keys inacessible. To add to realism, all dynamic objects are connected to a little script that checks for acceleration, so a sound can be played when the object falls or is pushed with force.

The Safe

While every key can be found pushing objects, we decide to make the last key of the game quite difficult to get. Using the VIDE-Dialogue Plugin for Unity, we created a little dialogue, in which the player has to enter the right code for the safe to open. The correct answer is – admittedly – difficult to find and represents the last barrier, before the game is finally over. Entering the wrong combination, the countdown is going to be set to 10 seconds, effectively giving the player no chance to just try all combinations.

The Player Asset

As Escape is a first-person adventure game, we chose to use Unity’s Standard Assets First Player as a basis. We added a script that allows the player to interact with the objects in the scene: Using a Ray, the player can detect all interactable objects. As soon as such an object is detected, the mouse can no longer be used to look around. Instead, the mouse arrow will appear, allowing the player to click on a window that will open, offering different options for interaction, depending on

the kind of object in focus. If the player wishes to not interact, he / she can use the keyboard to move away, causing the window and mouse arrow to disappear.

The Game Manager

The Game Manager is the central controlling instance of the game. It is responsible for switching between menu mode and game mode, updates the countdown, and provides methods that can be called from other scripts to determine the state in which the game is in. The game manager was not put in an extra scene, but activates and deactives the game objects that determine the mode of the game. Thus, the objects of the game world are not put back into their original position each time a new level starts. This helps creating the effect of the apartment becoming increasingly messy as the player needs to push around objects, creating barriers that can make the game more difficult. For example, the player may need to push the bed aside if it blocks the way out of or into the bedroom.

Testing and improvement

Because of the Covid pandemia, we were not able to test the game with as many participants as we would have liked to. Instead, we had to ask the people we share accommodations with for their help in trying out the game and finding issues. Nevertheless, this helped us to improve the game, especially the level of difficulty. Initially, we had the locations for the keys determined by empty game objects, and would instantiate one key per level. While testing, we found that this made the game too difficult and decrease the players motivation. Thus, a different approach was discussed and implemented: Instead of one single key per level, all keys are now present at the start of the game, improving the chances of new players to find at least one of them. This helped to boost the players motivation, since they had some success right at the beginning of the game.

Outlook and Improvements

At its current stage, the game is finished and ready to be deployed. We achieved what we were planning for and did the best we could with our limited ressources and considering the constraints and issues we had to deal with, namely the pandemia. After the end of the covid crisis, we are looking forward to testing the game with more participants and getting new ideas how this game could be improved. One idea might be to introduce new locations, items or new ways to interact with objects.

i https://en.wikipedia.org/wiki/Escape_the_room

ii https://projects.cri-paris.org/PSTOCK/2019/07-10/Y1iUuEAo/FILES/46340211-9798-4205-b758-211604fd7e25.pdf

iii https://www.sciencedirect.com/science/article/abs/pii/S0260691718309146

iv https://dl.acm.org/doi/abs/10.1145/3130859.3131436