Game Design Document

Catch 2022 - a bullet hell office game

Group 4
Martin Falke
Edvin von Platen
Marius Lelouard
Elena Palombini
Boning Li

22/12/2020

DH2650 Computer Game Design 2020/21

Table of Contents

1. Game Overview

In this section we present a general overview of the game, addressing the game concept, the main features, the genre and the intended target group.

1.1. Game Concept

The game takes place in an office building. Inside this building, the plants and appliances have come alive and attack the player. The player has to defeat the appliances to proceed with the levels of the building. It is a fast paced and engaging game as the enemies shoot continuously and death is permanent. What makes the game more fascinating are the little details in the setting, such as the shooting plants and the power outlet that will zap the player with electricity.

1.2. Feature Set

The most notable features of the game are:

- **Fast paced action.** As mentioned, the quantity of enemies attacking continuously makes for a fast paced and exciting game, where getting through the levels will be increasingly difficult.

- **Power ups.** The player will find several power ups placed in the levels, each with their own design, quirks and unique conferred abilities. For example, coffee to give a speed bonus.

- **Great replayability.** The sudden death of the player and the subsequent quick nature of each run makes for good replayability of even the first levels, as with each replay the player will be more proficient with the game and get further.

- **Score.** Including a final score based on speed after game over allows the player to compete with themself, and creates a wish to keep playing to eventually achieve a record.

1.3. Genre

The game falls within the genre of Bullet Hell. Catch 2020 features extremely reactive and fast gameplay. The players have to be continuously focused on dodging enemy projectiles while destroying them with theirs, and they have to make sure to not get cornered.

1.4. Target Group

The intended age group for the game are teens and young adults. The main characteristic of the average player of this game is that they should enjoy competitive, fast-paced indie games. The ideal scenario where this game would be played is in between longer games queue times (e.g. Counter-Strike, League of Legends) as a way to pass time, jumping in and out of the game quickly. The game allows for very short runs, so it would be a good way for them to enjoy themselves during the wait. The other way the game might be played is in longer sessions, where the player would grind to achieve a better run and get further ahead in the game.

The main motivation for the choice of this target demographic is that most of the members of the team of developers fall into this same demographic, so we feel more connected to the actual needs and desires of the users. This is particularly useful to "get the design right" in lack of a thorough testing process with the end users.

2. Gameplay and Mechanics

This section will explain the basic mechanics, objectives and gameplay of the game, and will go into the details of what makes it engaging and fun for the players.

2.1. Game Progression

The player has to progress through a succession of levels, each one harder than the one before. At each death of the player, the game restarts from the very beginning. It is therefore mainly the skills of the player that will improve the more they play and let them progress further in the levels. The random power-ups also allow some variability in the game. Once the player has cleared one level by killing all the enemies, they can proceed to the next one by using the elevator at the end of the level. In the final version of the game, a lore will be included. The player will be able to discover it piece by piece by the mean of collectibles randomly found in the levels.

2.2. Objectives

The objective of the game is to climb the levels of the office building. The player does that through elevators. The idea is that going up the levels of the building, the player will be able to confront stronger and stronger enemies and harder levels. The larger goal is to get the best score possible by going faster and further in the levels. The player will be able to compare themself to the other players of the game.

2.3. Play Flow

The game is fast paced and flows very quickly. The player must be constantly on the move or they'll die permanently once their health reaches 0, and will have to restart the game from the beginning. This makes for a very engaging gameplay that keeps players on the edge of their seat.

The game is not easy to beat but proves challenging, especially at the beginning. This is on purpose, and is meant to make the players want to complete the levels even more, by tapping into their "completionist" and competitive traits.

2.4. Physics

Catch2022 uses the Unity2D physics implementation. Physics are used for both movement and collision detection. The 2D physics system assumes a side scroller perspective, e.g. gravity pulls objects down, this leads to issues when using a top-down perspective. To enable the top-down perspective the Z-rotation axis is locked and gravity is disabled for all units.

2.5. Movement

In this subsection we detail the specifics of player movement.

2.5.1. General Movement

Movement is fast-paced and keyboard controlled (WASD). Keyboard controlled movement (i.e. not in the mouse direction) allows for 4- or 8-directional movement. To increase the control of the player we use 8-directional. However, enemies are not restricted to 8-directional movement due to technical limitations of our pathfinding. An Unity asset is used for A* pathfinding and the paths computed by it are linear or Bezier curves.

2.5.2. Other Movement

The player has a *dash* ability (SPACE). This ability gives a short burst (0.05 sec) of movement-speed. In a bullet-hell game the player needs tools to handle dangerous situations, e.g. surrounded by bullets. The dash ability was added to increase the players reactive mobility and enable them to handle such situations.

2.6. Combat

For any bullet-hell game, combat is key. The goal of Catch2022's combat is to be quick, reactive, and fair. Quick so that the pace of the game is fast, reactive in the sense that the player is always in control, and fair so that the player is never forced by the game to end up in a situation where it is impossible to win, such situations should always be the players fault. To enable these three concepts we equipped the player with a shoot attack, a parry ability to protect themselves, and a dash movement ability. See the Character section for more details.

During the game, the player encounters various enemies. The main strategy for combat will be to move fast and precisely to avoid enemies' projectiles or exploding melee enemies and shoot at them at the same time. The player cannot avoid combat as they have to defeat every enemy of a level to be able to access the next one.

2.7. Power ups

Power ups are special elements in the scene that the player can pick up by stepping on them. Each power up confers a special ability or otherwise has some effect, it is either permanent or active for a limited period of time. In the demo, only the briefcase attack speed power up is implemented, but we present all of the power ups that we ideated and that belong in the final version of the game. Most of the power ups also add nicely to the "office" setting as they are items that can actually be found in offices, such as coffee and briefcases.

The initial powerup system designed for the game was heavily inspired by roguelike games, for example Risk of Rain. Each playthrough would have power ups randomly placed throughout the levels. This introduces randomness and replayability to the game, sometimes the player will be lucky and get very good power ups, allowing them to progress further. On the other hand they can be unlucky, only getting low impact ones. However, during development the scope of the game was reduced and the genre shifted from roguelike and bullet hell to only bullet hell. And only one power up was implemented with static placement in the levels.

2.7.1. Briefcase
In the demo, there is a special glowing red briefcase that when picked up confers to the player a permanent attack speed bonus.

The briefcase attack speed power up

2.7.2. Golden Apple
This power up makes the player invulnerable for 5 seconds. The appearance of this power up is of a glowing golden apple, and it also makes the player glow for as long as it's active.

2.7.3. Coffee

This power up gives a walking speed bonus. The appearance of this power up is of a coffee cup, and for the duration of the effect a coffee cup icon appears close to the parry one.

2.7.4. Yoghurt

This power up causes the player's bullets to start bouncing on walls. The appearance of this power up is of a yoghurt container, and for the duration of the effect a yoghurt container icon appears close to the parry one.

2.7.5. Cake

The cake power up has no effect but spawns a small text that says "The cake is a lie". This is a reference to the famous game Portal.

3. Setting and Characters

We now give an overview of the game setting and its characters.

3.1. Game World

The player is exploring an office building level by level. The game is set a few years in the future. The building is empty of life. The atmosphere is dark. The objects that the player interacts with and fights against are all the furniture and appliances that are usually in an office: desks, boxes, computers, decoration, machines, cleaning ware… The only light that is coming in is from the windows and some of the objects in the level, like computer screens. All of this is made to create a mysterious atmosphere, with no specific clue on where and when exactly the game takes place.

3.2. Characters

Here is a list of all the characters that appear in the game, including the main character and the enemies.

3.2.1. Main Character

The main character is a former employee with a mysterious ninja-like appearance. Only their shape and glowing blue eyes are visible. The rest of their body is entirely black. See image below.

The Main Character

3.2.1.1 Shoot attack

The main attack of the player is shooting. No weapon is used for the attack, instead the bullets are spawned from the body and in the direction of the crosshair. A weapon system was considered, where the player finds weapons in the levels with different behaviour. However, due to scope concerns and limited time we went with a single bullet attack.

The attack speed of the player starts at 1.5 attacks per second and can be permanently upgraded by collecting the briefcase attack speed powerup.

The bullet sprite is a yellow ball which emits light around the projectile. The light increases the visibility of the projectile and makes it "pop".


Player Projectile

3.2.1.2 Parry Ability

Using the right mouse mouse button the player can activate the *parry* ability. The parry ability acts as a shield, reflecting incoming attacks. The purpose of the parry is to enable reactive

gameplay for the player, e.g. when cornered the player can activate the parry and get some breathing room. The parry is also an offensive tool, when timed right it can be used to reflect two projectiles from a single enemy.

The parry has a duration of 1.5 seconds and a cooldown of 2 seconds. Initially the duration was shorter, only 0.3 seconds, but we thought that this was too punishing for the player. One reason being that a common scenario is getting shot by two plants simultaneously but with some time offset you could then only parry one of the projectiles. Another alternative would be to have a very low duration (< 0.3) and a very low cooldown, this would make the ability very reaction based. However, during testing we felt that the longer duration felt better and that the visual feedback for the low duration was lacking.
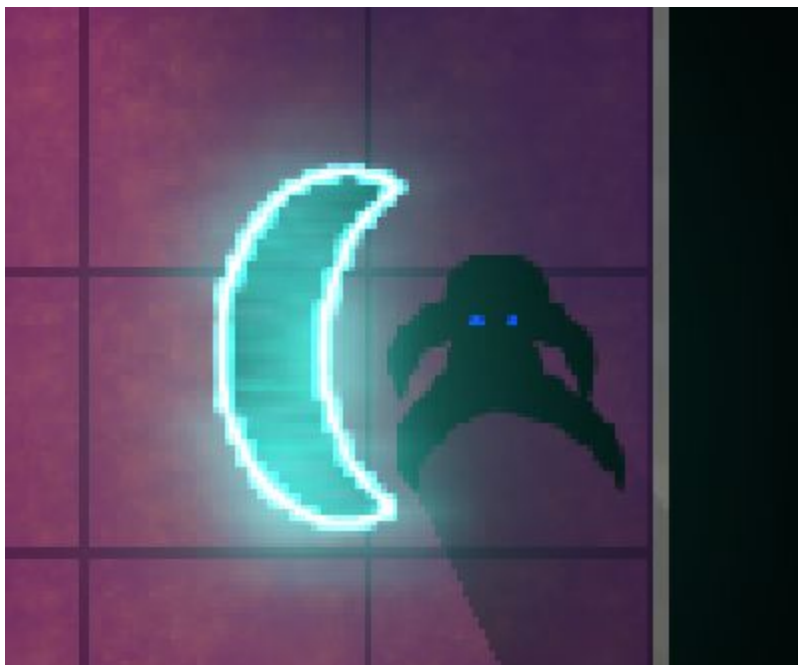
Player Parry Ability

3.2.1.3 Dash

The *dash* is a movement ability the player can use by pressing the spacebar. Dash increases the movement speed of the player by a large amount for a very short duration. The goal of the ability is to increase the reactive and movement ability of the player. It allows the player to run away or dodge enemy attacks more effectively.

3.2.2. Evil Plant

The evil plant is the most common enemy in the game. It looks like an ordinary office plant, but it moves around and starts chasing the player. It shoots yellow bullets, and if the player takes 4 shots, they die. The plant dies if it is hit by 4 fireballs.

Two Evil Plants shooting at the player.

3.2.3. Automatic Vacuum Cleaner

The Automatic Vacuum Cleaner is a lot faster than the Evil Plants. It tracks the player and reaches them very quickly. It then explodes, and if the player is caught up in the explosion, they die instantly. The explosion animation is composed of a full, glowing red circle and a smaller white circle outline expanding from the vacuum cleaner. Once the white outline reaches the edge of the red circle, the explosion kills the player if they are in the area of the explosion. If the Automatic Vacuum Cleaner is hit twice by the player, it is destroyed.

The goal of the vacuum cleaner enemy is to force the player to move around as staying still will lead to instant death.
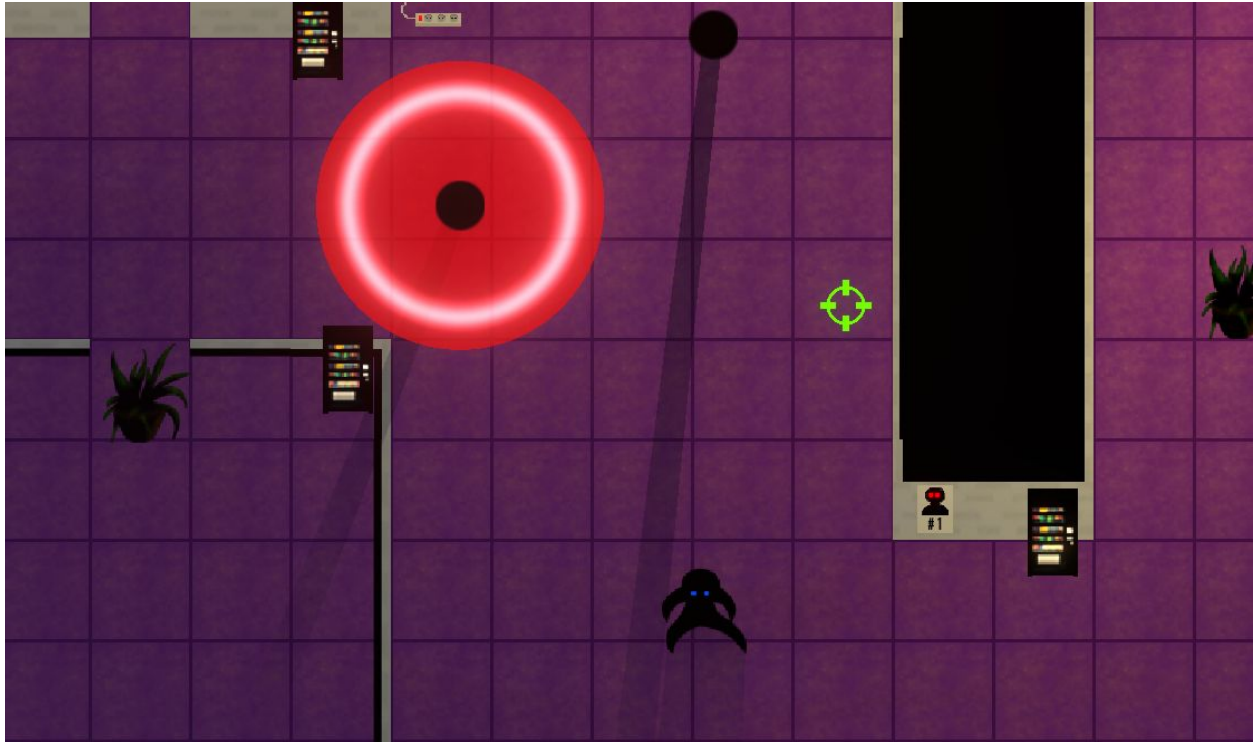
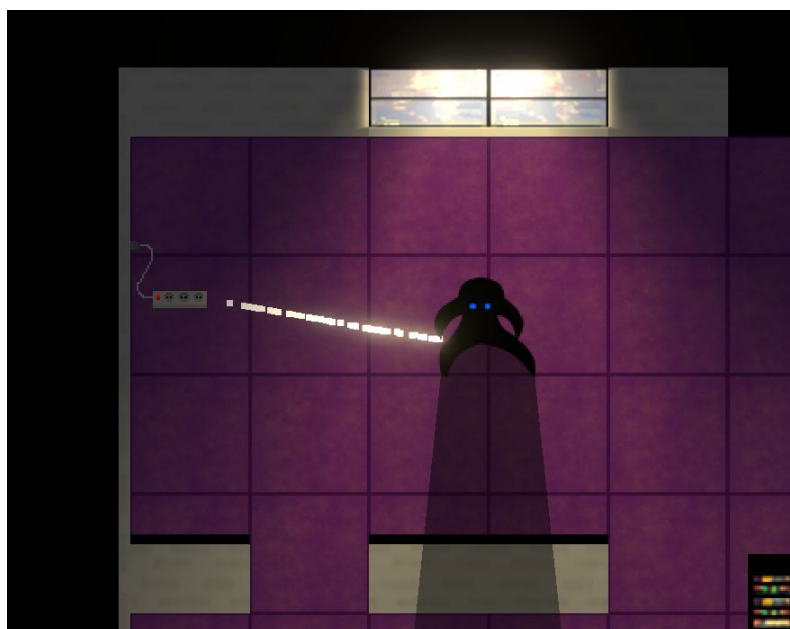*Two Automatic Vacuum Cleaners, of which one is exploding.*

3.2.4. Power Plug

Another enemy that wasn't included in the demo version but belongs in the final game is the Power Plug. It doesn't move but keeps shooting electricity bolts at the player.

A Power Plug, shooting electricity.

3.3.   Levels

Two levels were manually designed and implemented for the demo. While the initial idea was to include procedural generation of levels it was not included in the demo due to time constraints, it may be implemented in a future patch.

The two levels are designed with separate goals. The first should introduce the game and its elements to the player while the second acts as a real challenge.

3.3.1. Level #1

Editor view of Level 1

This level is an office floor, see image above, with purple floor, where the light comes from windows on the northern wall; the shadows subsequently point south. The map is composed of a corridor that leads to an open office space with 4 rooms, placed at the corners of the room. The objective of the level is to defeat all of the enemies and get to the elevator door, which is located in the upper right corner room.

The player spawn point is to the top left, from the spawn two plant enemies are visible. The player has a chance to move around and try the controls before moving into line of sight of the plants and engaging them. After defeating the two plants there is a single vacuum cleaner-enemy that engages when the player comes close. This early section gives the player an introduction to

the two enemies of the game. In the rest of the level more enemies are faced at the same time.

3.3.2. Level #2

The second level was made to be harder than the first. Here the player starts in a safe room in the middle of the level, see the image below. Desks are placed at the side of the level, but with enough distance to the walls so that the player can move between the desk and the walls. Enemies are placed on both sides of the level, no area should be safe and the player has to keep moving.
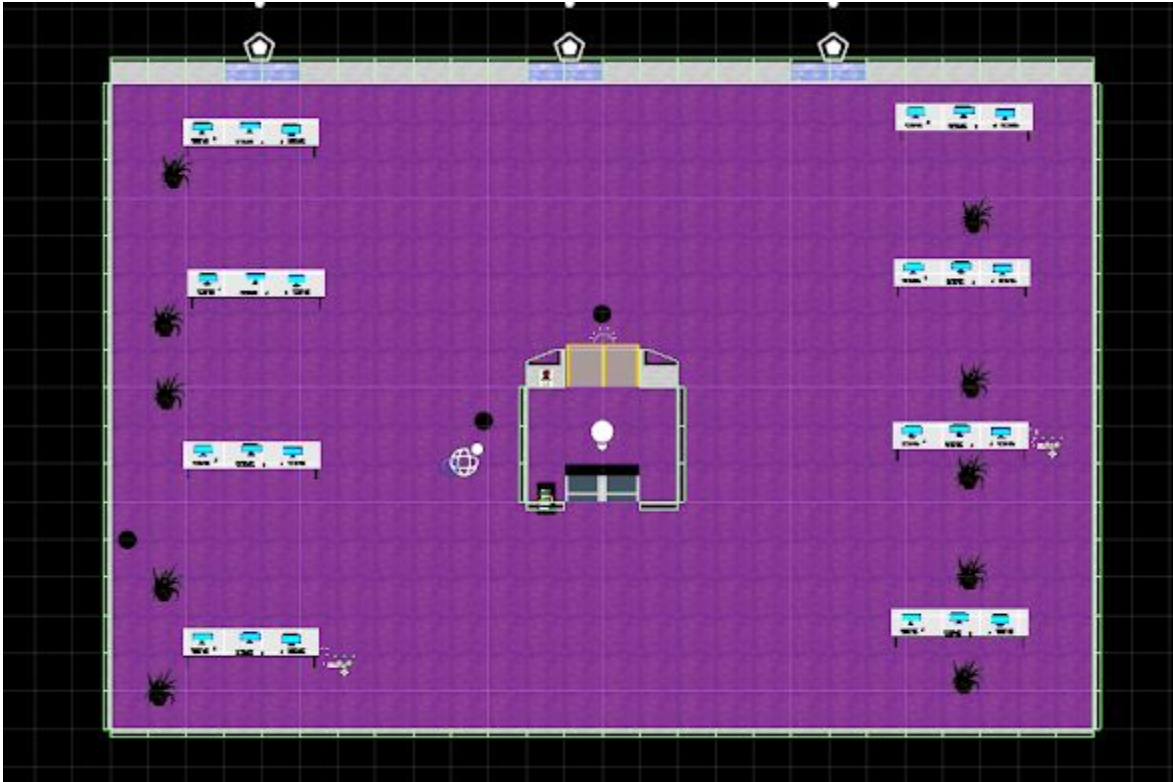
Editor view of Level 2

4. Interface

In this section we will present the different components of the interface: the visual system, the control system, music and sound effects.

4.1.   Visual System

This section will present all the visual components of the interface, namely the HUD, the menus, the rendering system, the camera and the lighting models.

4.1.1. HUD

The HUD is minimal to increase the immersion. Along with the crosshair, only the cooldown times of abilities, the health, and the score is shown in one corner of the screen each.

The parry ability HUD element indicates when the ability is available and when it is on cooldown. The left image below shows the HUD element when the parry ability is available for use and the right during the ability cooldown. The cooldown is shown using an animation where more of the HUD element is shown the less cooldown time is remaining.

*The parry ability HUD element*

The crosshair is a green sprite which replaces the mouse pointer. The crosshair shows in what direction the player will shoot.

Crosshair

There is a HUD element that displays how much health the player has left. It shows one heart out of four for every projectile that the player can be hit by before they die. This is shown in the top left corner of the screen.
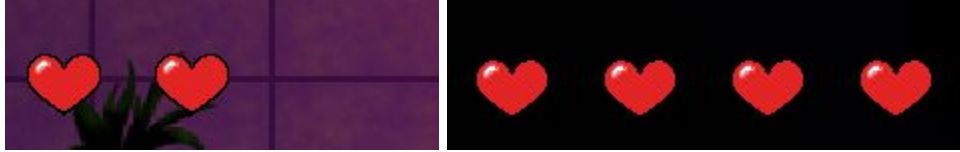
Health HUD element with two and four lives remaining respectively.

Finally, the HUD also shows the player's score at the bottom right of the screen as a yellow text, like in the image below. It is updated in real-time as enemies are killed and levels are completed.

Score HUD element

4.1.2. Menus

When the player launches the game they are presented with the main menu, shown in the first image below. It allows the player to start the game, change the options, or quit the application. This menu is also reachable by pausing in-game, when the game is completed, or when the game is lost, through the respective menus of those scenarios.

Main Menu

When the options menu (shown in the image below) is entered, the player can choose to tweak the volume of the music and sound effects individually. When the player is happy with the values, they can press 'Back' to return to the main menu.

Options Menu

4.1.3. Rendering System
The rendering will be done by Unity's 2D Universal Render Engine. This will make the use of shaders a lot easier.

4.1.4. Camera
The camera is oriented top-down, and follows the player across the level. Post-processing effects are applied to accentuate the immersion of the game.

4.1.5. Lighting Models
The level is almost dark. The only real light of the level is coming from the windows. A lot of elements are glowing. It is the case among others of power-ups, computer screens, the eyes of the player and the enemies. There is also a slight global light to illuminate parts that are not directly in line of sight of a window, so that they are playable.

4.2.   Control System

The game is controlled with the help of a mouse and a keyboard. The targeting system, the shooting and parrying commands are controlled by the mouse while the movements and dashing commands are controlled using the keyboard. The navigation in the game menus is also done with the mouse. The game can also be paused using either 'P' or 'Escape' and unpaused the same way or by using the mouse.

Gamepad support was considered but due to the target platform being PC mouse and keyboard was prioritized.

4.2.1. Player Keybindings
Here we give a list of all player keybindings in the game, they cannot be changed in-game:
a: *move left*

d: *move right*
w: *move up*
s: *move down*
left mouse button: *shoot*
right mouse button: *parry*
space: *dash*
p or escape: *pause/unpause*

4.3.   Music

The game soundtrack was made by a musician who our team knows. The overall music style mixes the concept of pixels and future cities, and the fast pace is also in line with the tense atmosphere of the battle part of the game.

4.4.   Sound Effects

This subsection presents sound effects present in the demo, the tools used and the design process, as well as an example of sound clip generation.

4.4.1. Tools
The tools used in the sound design of this project are: Audacity, Mixpad. Among them, Mixpad is mainly used to generate basic sound effects, create remixes, mashups, beats as well as add recorded audio pieces. Audacity is used to post-process the generated audio clips, such as Multitrack mixing, noise reduction, etc.

4.4.2. Design Process
The core idea of the sound design is that all sound effects should be consistent with the game's graphic style and music style. Almost all audio clips are modified and created on 8-bit style and basic audio resources in multiple versions.

4.4.3. Sound Clip List (Catch 2022 Demo version)
- Death sounds for player as well as all different enemies
- Player and enemies' bullets sound.
- Player's power-up sound.
- UI interaction sound.
- Game over and pass the level sound.

4.4.4. An Example of Sound Clip Generation
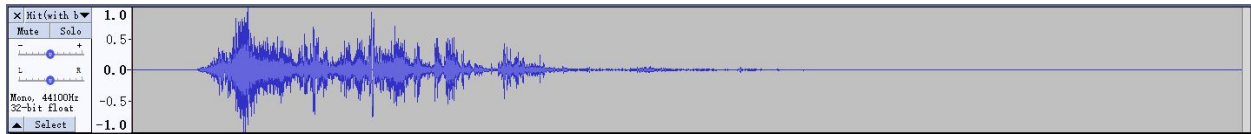
Sound effect of hit with blood splash

This sound clip used 2 original sub-clips as the body, by doing post-processing as well as rhythm control, we got the effect of hitting enemies with blood coming out.
Here are the 2 original subclips:
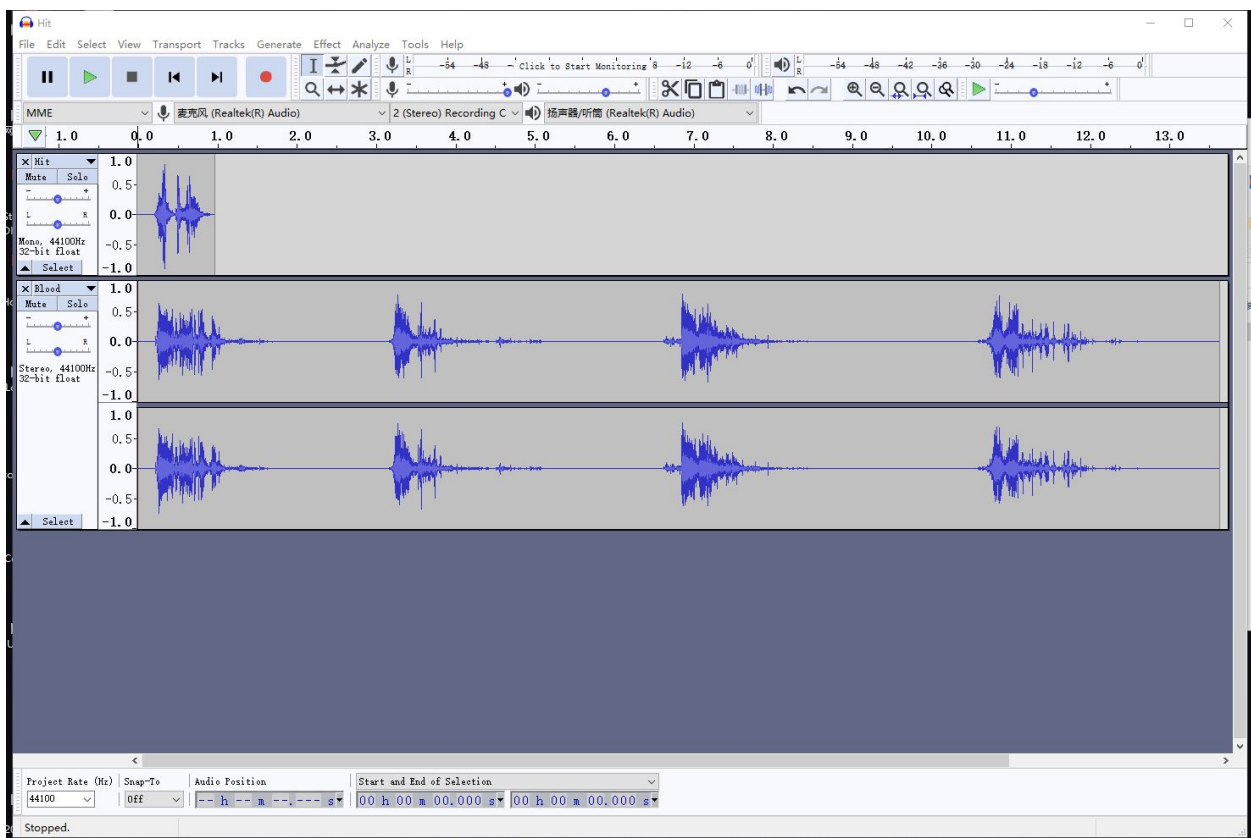
Original Subclips

There are two ways of making subclips:

- Using the **Generate** function in Audacity, it has many basic sound clips. By adjusting their pitch, tone as well as rhythm, we can get simple sound effects.
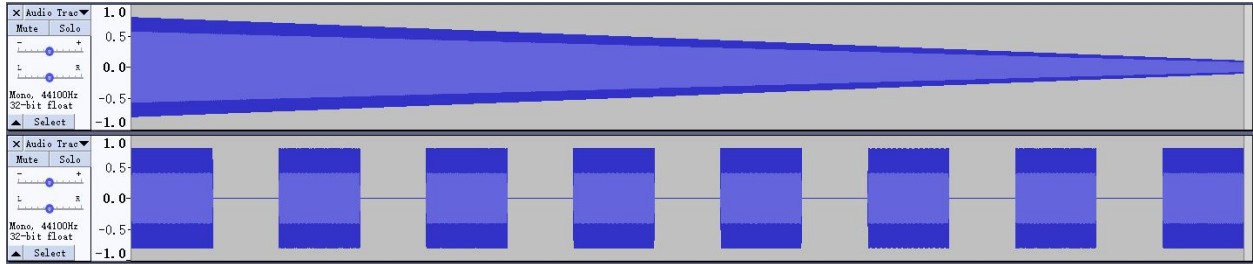
Chrip and DTMF tones

- Using recording equipment simulates sound effects as a sound source. The device we used for this project is:

Zoom recorder

We borrowed this device from the sound studio in KTH and it helped a lot with the sound source-track generation. Basically, in the recording process, we took the form of simulating actions to capture the sound effects. The original sound clip only needs a rough outline for the next work.

Finally it is the post-processing workflow:
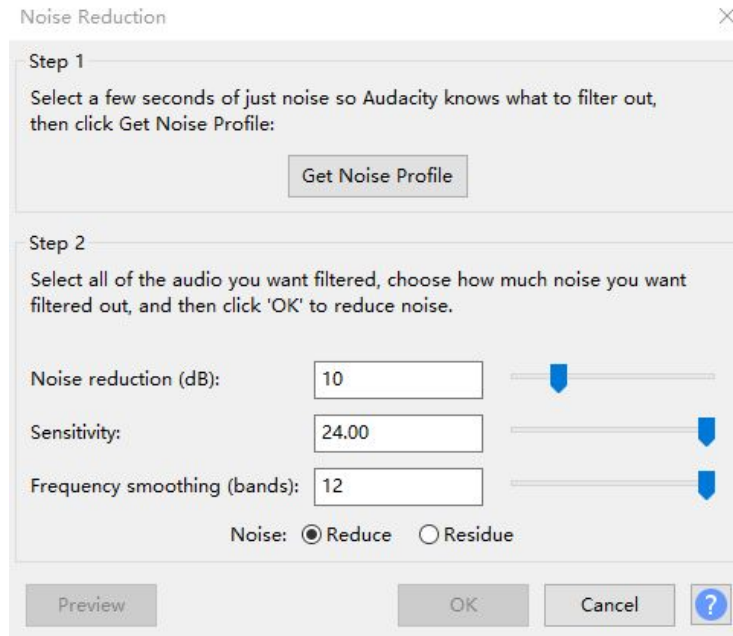
Noise reduction

The first thing is to do the noise reduction, game sound effects have certain requirements for the clarity of sound segments, and unnecessary noise must be removed. The next step is to create an 8-bit style for the clip to cater to our game atmosphere. Here we refer to the rhythm characteristics and distortion of 8-bit music, and carry out the design process of optimization, testing effects, re-optimization. Finally it comes to the parameter settings, rhythm modification according to game actions or animation effects, duration control, etc.

Overall, the sound design of the game draws on the concept of soundscape in sound design. A soundscape is a sound or combination of sounds that forms or arises from an immersive environment. The scenes of the game and the cases where sound effects are applied are also part of soundscape. Using relevant sound design knowledge can effectively enhance the immersive player experience of the game.

4.4.5. Resources
The basic part that constitutes the audio clips mainly comes from the built-in warehouses of mixpad and audacity. In some clips, open source audio clips are mixed up to obtain better sound effects. The advantage of this is that it will not deviate too far from the main style, and can modify the audio to a large extent to freely adjust the effect of the sound in the game.

5. Artificial Intelligence

In this section we will present the various AI methods used in the game, both for Enemy and Support AI.

5.1.   Enemy AI – Villains and Monsters

Some enemies are given artificial intelligence to detect the player and follow it. Pathfinding is therefore implemented for those enemies along with a basic state machine to control the different states of the enemies. Depending on the specifications of the enemies, different kinds of artificial intelligence are implemented. For example, ranged enemies like the *evil plants* will stay in range of the player but not too close. What they have in common is that they switch between states depending on the player's relative position and visibility.

Enemies start in an *idle* state where they wait for a patrol timer to expire. When that patrol timer expires they move to a random grid cell above, below, to the left of, or to the right of them and they then go back to the idle state. As soon as the player gets within a certain "provoke range" they will enter the *provoked* state and start chasing the player. For the ranged enemies, as soon as the player comes in line of sight of them (either through themselves or the player moving) they will switch to the *shooting* state. For the *automatic vacuum cleaner* is a bomber type enemy that will stop chasing the player as soon as it is within explosion range of them. It will then enter the *exploding* state, perform one explosion and re-enter the *provoked* state.

The basis for the ranged AI state machine was initially sketched using Google Draw so that the team had a thoroughly worked-through idea of how the AI would function in-game before any implementation was attempted. It was then implemented for the ranged AI and later adapted to the bomber AI.

5.2.   Support AI

This subsection presents the methods used for Player and Collision Detection and Pathfinding.

5.2.1. Player and Collision Detection
The player is detected by checking the range (and the line of sight with a raycast for the ranged AI) in every update iteration. If the range is lower than or equal to some threshold, the player is considered to be detected.

Collision detection is done with circle colliders for the player, the enemies, and the projectiles. When these overlap, they are considered to have collided. The parry effect uses a polygon collider specifically adapted to its shape, so that it can handle collisions of its own from different directions. Finally, the level completion mechanic uses a box collider to detect that the player has come within range of the level exit.

5.2.2. Pathfinding
Pathfinding is done using the A* search algorithm, which is good for detecting valid paths in a real-time dynamic environment. It is typically only slower than algorithms that can pre-process

the entire environment, which of course is impossible when there are enemies and a player constantly moving throughout the level that change the valid paths. This was more or less completely implemented using a generous and free open source project called the [A* Pathfinding Project](#). The only part that had to be done was to use a "Seeker" script to acquire a path and then move towards that path. This was implemented in a way so that each enemy updates their path every 0.5 seconds so that they are neither too sluggish nor too agile in their movement. Some bugs with the pathfinding emerged as the project moved from each object/tile having the size of exactly one cell to some tiles being smaller than that. These were not fixed as they did not impact the gameplay enough to warrant the time it would take to fix it.

6. Technical Platform

This section will go into details on the technical platform, and all of the hardware and software specifications.

6.1. Target Hardware

The game is developed for Windows on PC. The Unity engine technically supports builds for Mac and Linux distributions. However, our time to develop and test the game is limited so we chose to focus on Windows since all of us have access to a Windows PC. Windows is also the most common operating system used for PC games. Porting the game to other devices, e.g. mobile, would also require extra time that we would rather spend on developing the core of the game.

6.2. Development Hardware and Software

Each of the team members had access to a personal computer of their own at home and these were used for all the development and its surrounding procedures. Some additional pieces of software we used to make the development process easier for the team. Git and GitHub were used to facilitate parallel work on different aspects of the game and the merging thereof. Codecks, an online project management tool designed for game developers, was used to keep track of tasks in the project with respect to the actual game, this document, the website, as well as asset creation. A dedicated Slack workspace was used to communicate and coordinate by text throughout the project while Zoom and Discord were used to hold regular meetings.

6.3. Development Procedures and Standards

To make the development process smoother and most of all make the merging of individual work as pain free as possible, a few guidelines were established early in the project with regards to how work in the game engine was done. One such guideline was that when doing individual work, that work should be done in a separate scene from what the rest of the team is working on.

This way, many false-positive merge conflicts in Git are largely avoided. Normally when this happens, it can cause a scene to break completely because the code of a game object has been placed by Git in the middle of another game object. Additionally, as soon as a task was started it was marked on the Codecks workspace and written on the Slack to make sure no duplicate work was made.

6.4. Game Engine

The selected engine for the game is Unity. The game was developed using version 2020.1.13f1 for Windows. This was a fairly easy choice, we had no specific reason to go with an older version of Unity so we chose the latest version that was easily available for download with Unity Hub.

6.6. Scripting Language

The scripting will be done in C# as it is the main scripting language in Unity.

6.7. Distribution

A demo version of the game will be available for download on [the website dedicated to the game](#). Upon full release of the game, the game will be (if possible) launched on Steam through the Steam Direct program.

7. Business Plan

To provide a more comprehensive view of the future of the game and our strategy, we have added this section to the GDD, that will deal with the business aspects and considerations related to our game.

7.1. Executive summary

Catch 2022 is an indie pixel action game with one time purchases which will be available on Steam, since Steam is the easiest platform to do the game releasing and most gamers in our target group already have a Steam account. Another strategy is that the game will be published on Epic Games platform as a free game, the source of income is from Epic games, the advantage of this strategy is that the commercial risk can be shared by Epic Games and us. All soundtracks of the game are also available for purchase. As for subsequent updates, DLC may be launched for purchase, and season pass will be part of the future commercial revenue of the game.
- Target player age: 10-30.
- Expected ESRB rating: Everyone 10+, Fantasy Violence included.

● Planned release date: December 31st, 2020.

7.2.   Description of Target Sectors

Catch 2022 was primarily made for teens and young adults all over the world. As one article noted, the range of independent gamers can be divided into four different categories, which of we think three are relevant:

● The Cult Gamer. Well-informed and active gamer, enjoys *"hidden gems"* i.e. games not played by the masses.

● The Casual Video Gamer. This type of gamer plays games on the web, and maybe some downloads on PC or console.

● The Ex-Hardcore Gamer. Generally older gamer with lots of experience. Stays up to date on the latest trends although rarely plays for longer sessions anymore, or the latest games.

Since Catch2022 is a fast paced game and can be quite punishing our primary target group is the cult and ex-hardcore gamer i.e. gamers with previous experience.

The market positioning of the game is that we mainly regard the game as a "filler" between other games (especially longer competitive games). This decision affected many developments. For players who like to challenge themselves, they can still spend longer trying to break records.

7.3.   Market Analysis

From the data of an indie game market report, being an indie game developer certainly is appealing with all the creative freedom and in-app purchases rising. Slice Intelligence reports that people in 2018 spent on average $87 in free-to-play in-app purchases. For comparison — players who purchase games for traditional consoles spend only $5 more a year.

Even with the victory march of microtransactions and in-app purchases increasing, the industry still heavily relies on whale players. Only 10% of gamers account for 90% of revenue for in-app purchases. It can be found that the overall consumption trend of players is more inclined to in-app purchases, but the data from different regions shows that the overall sales of one-time purchased games in the Europe market is still very high, which is why we choose to use the on-time purchase model for our game releasing.

On the other hand, with the continuous expansion of the Esports industry, the number of competitive game players continues to increase, which means that more players will be willing to play our game in the spare time of matching competitive games. Therefore, the expansion and development potential of the indie action game market are very promising.

7.4. Competitor analysis

Catch 2022 is an indie pixel bullet-hell experience game. Through market research, our main competitors are divided into two categories:
- Famous indie rogue-like action games: Wizard of Legend, The Binding of Isaac,etc.
- Mini-games to play while in queue: Curveball, Osu, agar, etc.

7.4.1. Strengths and Weaknesses
Compared with most Mini-games to play while in queue, the advantage of Catch 2022 is that the experience of each round of the game is different. Randomly generated maps and events are the basis for infinite repetitive play. Compared with famous indie rogue-like games, Catch 2022 has a faster pace of the game. Players eliminate enemies quickly and there is no need for them to pay too much attention to it. It is a mini game that fills the long waiting time of match making in other games. Catch 2022 will play such a role in the target group of players. The unique selling points of Catch 2022, in its final version, are:
- Fast-paced action as the core gameplay can attract many action game players.
- Infinite Replayability: Each run is unique since the levels are procedurally generated.
- Unlockable Power-ups: Power-ups can be "discovered" and permanently unlocked in each run, allowing players to drop at random in consecutive runs.
- Benchmark High Scores: Provides a Rank system for players who like to challenge and compete with others for scores.
- Price advantage: Compared with other roguelike games, Catch 2022 will not be as expensive as other roguelike games.

The main weaknesses of Catch 2022 are:
Compared with the main competitor games, the richness of roguelike elements still needs to be improved, which means that the number of core gameplays that players can explore is limited. Of course, as the game continues to update and improve, this weakness will definitely be weakened and even resolved. In addition, most mini games usually have a web version, that is, they do not need to be downloaded and can be played directly on the web. Catch 2022 needs to be downloaded and installed. We will consider porting the game to other platforms to better suit the game features of the mini games scenario. Finally, game promotion is also a weakness in our business plan. How to let more players know about Catch 2022? Our team has fewer resources than many famous game studios among competitors, and there is also a certain gap in the investment budget for game promotion.

7.4.2. Environmental Opportunities and Threats

Nowadays, the game live broadcast industry is developing very rapidly. We plan to provide Catch 2022 activation keys to well-known game anchors as a way of publicity to quickly gain a certain user base, with gameplay as the main attraction, and let players Recommending Catch 2022 to each other is our goal and opportunity. On the other hand, as a competitor game of Catch 2022, the threats they may pose to us include:

- Excellent game graphics may be more attractive to players. The graphics of pixel games have been criticized by many players and game evaluation agencies.
- Too high game difficulty causes non-hardcore players to lose interest in the game. Consider adding game difficulty and randomness parameter settings to cater to most players' tastes.
- The problem of pirated games is a common threat from game encryption crackers to all game manufacturers and studios.

7.5. Marketing Plan

Marketing is essential to get the players to know that our game even exists. This section presents our marketing strategy.

7.5.1. Distribution and Platforms

Distribution of the game through Steam. One time purchase as the marketing strategy. Steam is the easiest platform to publish games and is for gamers around the world. At the same time, most of our target users are users of Steam. In the beginning, we will focus heavily on marketing through Youtube, Reddit, Discord, and we will send free game keys to Twitch streamers. We believe our game is particularly suited to being watched in streaming as it's very fast paced and engaging.

7.5.2. Strategy Related to Audiences

Refer to section 1.4, the use situation of our target user group can be: In between longer games (eg Counter-Strike, League of Legends queue times) as a way to pass time, jumping in and out of the game quickly. We are also considering adding the progression system for the game. This means Faster runs give a better ranking. Discrete ranks (amateur, novice, pro, legend…). Hardcore players will always compare their own performance with others, so as to be immersed in Catch 2022 and increase the adhesiveness of the game with players.

7.5.2. Pricing

The price of the game will be between 5-8 US dollars, lower than the average price of popular indie action games.

7.5.3. Community Management and Development

The game will enter the Early access stage in the early stage, during which it will communicate with players through the community, publish announcements, update content and collect ideas from the community, and make full use of initial player resources to polish and adjust the game. We want to foster the growth of a community around the game. We will do that through various social media preferred by gamers like Steam and Discord. We will also keep our users informed about all upcoming updates of the game through our website.