BOUNZY GAME DESIGN DOCUMENT

2023

https://htramu.itch.io/bounzy

GAME SUMMARY

BounZy is a shooting gallery puzzle game designed to be simplistic yet engaging. Main goal of the player is to shoot the zombie, which is hiding behind different kinds of obstacles throughout the game levels. In order to achieve this player must utilize the bouncing bullet mechanic with the help of interactable objects.

GAME OVERVIEW

THEME / SETTING / GENRE

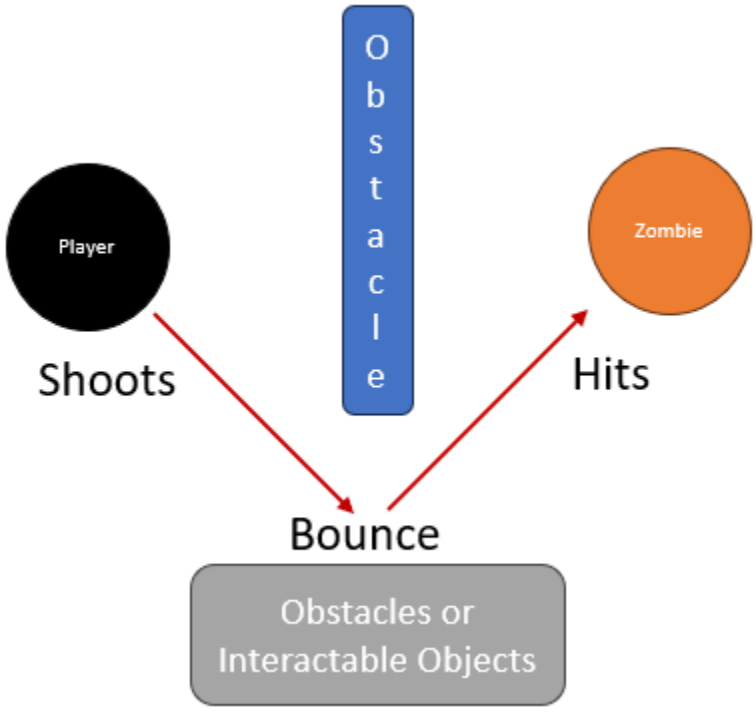
Scientific research of newly developed bouncing bullets, in the controlled shooting test chamber environment is the main theme. Game represents an alternative modern world where zombie menace exists and Zombie Interdiction Agency is combating them. Genre is physics-based puzzle.

TARGET AUDIENCE AND PLATFORM(S)

- Game audience is +7 ages
- PC – current, web based
- Mobiles (IOS, Android) – future

Core Gameplay

Core gameplay is designed to be simple: Shoot the zombie behind obstructions before the time runs out. A static player avatar, which can be angled with mouse movement, fires limited bouncing bullets in order to achieve this goal. In order to help player, several interactable objects are to be placed on levels.



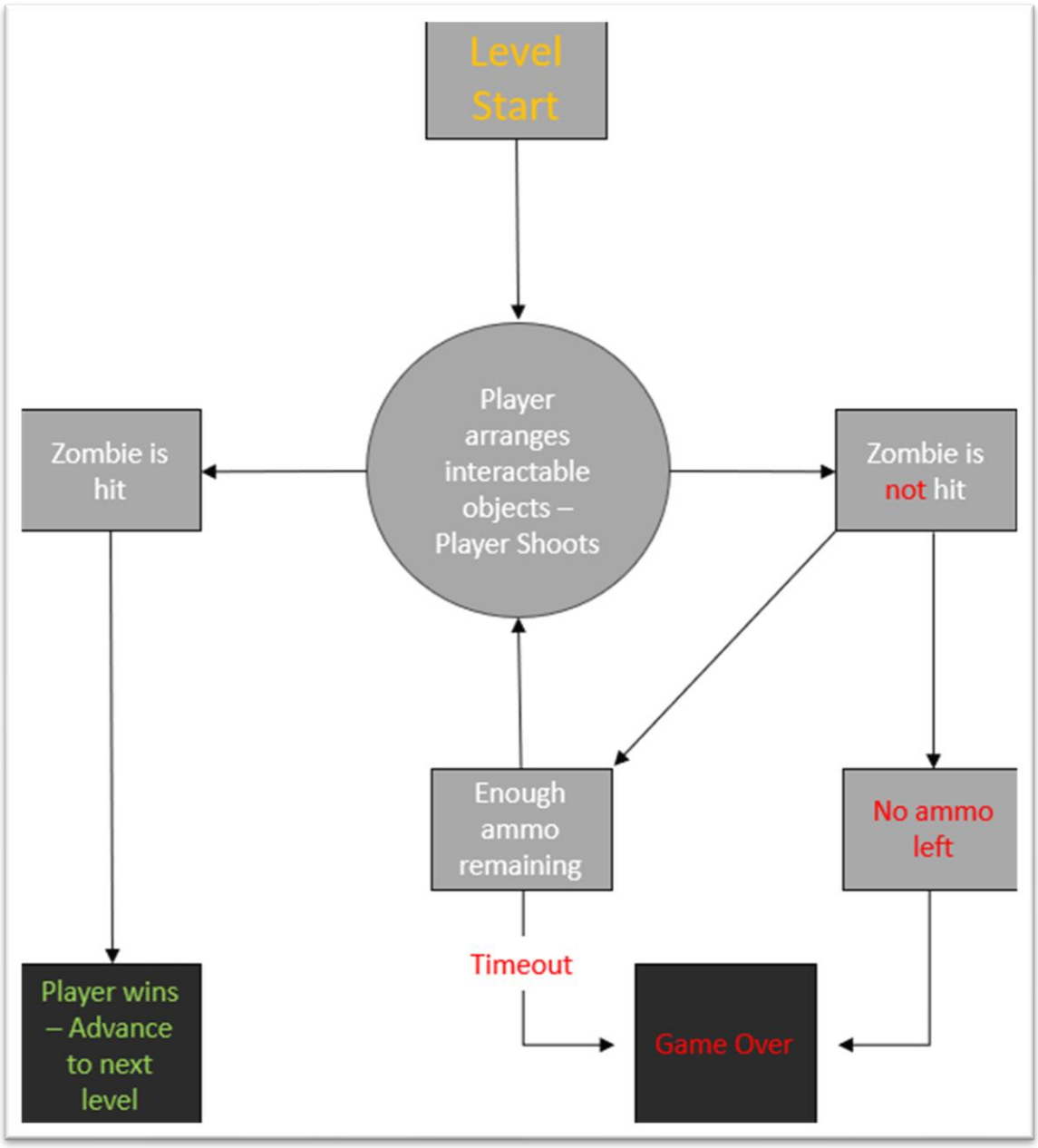
Story And Gameplay

Story

Set in the alternative version of our current world state, in the universe of BounZy zombies do exist but denied. Governments put out the Zombie Interdiction Agency (Z.I.A.) to combat ever-growing zombie menace and leave no trace behind. Here you fill the shoes of a Z.I.A. field agent to help Z.I.A. scientists to test newly developed bouncing bullets, which can bounce solid surfaces but very deadly to zombies.

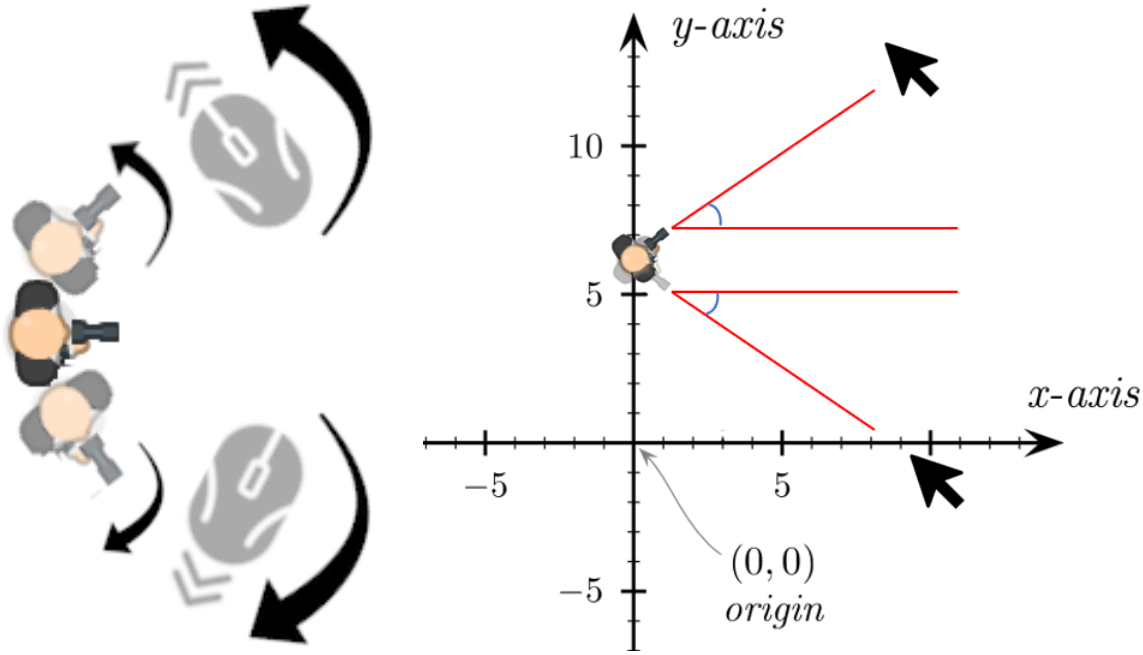
GAMEPLAY

Core Gameplay Loop

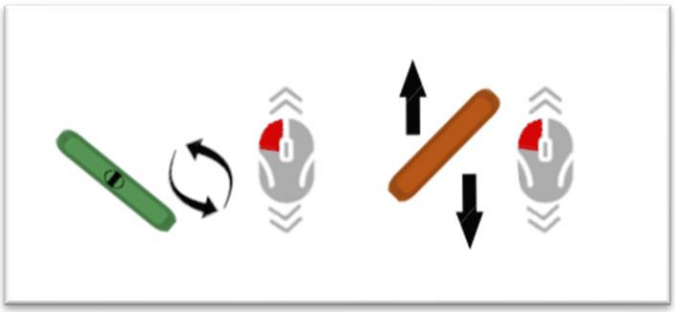


Movement And Controls

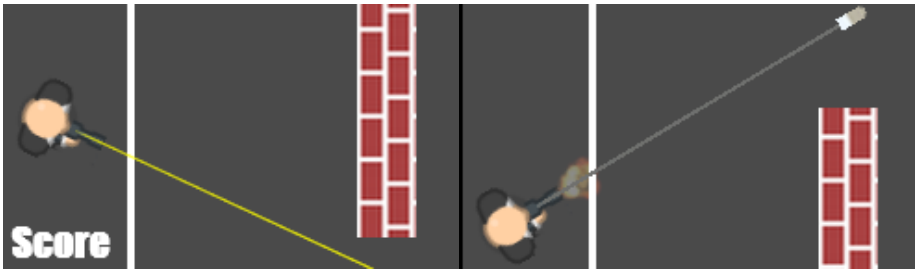
The player avatar is a 48 x 42 px 2D sprite positioned in the middle of the left side of the layout (origin point will be at x:32, y:352, given the layout height is 704 px) facing to the right-hand side of the layout. While static (no vertical or horizontal movement), it can be angled around its origin, in real-time with mouse movements. Avatar angle matches the mouse pointer angle of the 2D coordinates.



Player shoots with right mouse button. Rotates or moves Interactable Objects with holding left mouse button. (Detailed on their respective mechanics page)



Middle button shoots a harmless tracer bullet with a colorful trail that is different from trail color of *Bouncing Bullet*. Tracer bullet is to show the player approximate route of a bouncing bullet.



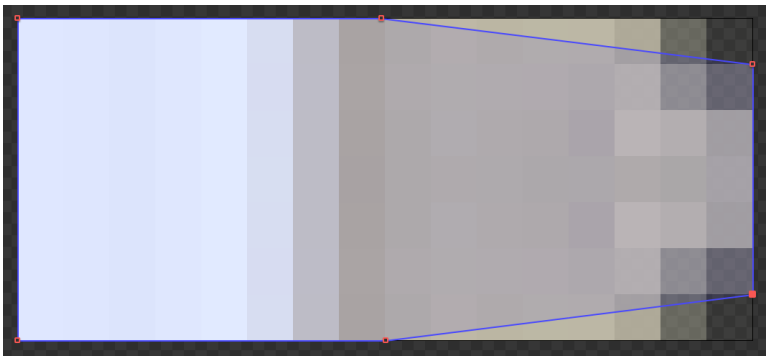
Tracer Bullet trail- Bouncing Bullet trail

Gameplay Mechanics

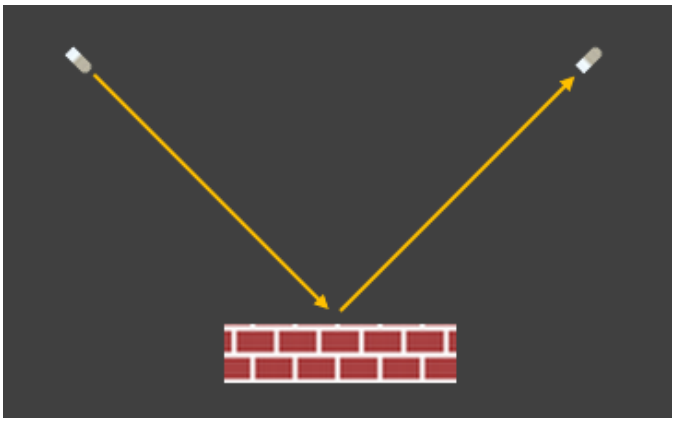
SHOOTING

Shooting is the only mechanic to fulfill the main objective. When the player clicks the right mouse button, a bullet sprite is created. This is accompanied by a fire sprite (which fades out in 0.5 seconds) gets spawned at the same location and along with an "gun sound" audio file is played to create shooting "feel".

- *Bouncing Bullet* is a 16 x 7 px 2D sprite. It is spawned at the gun tip location of the avatar sprite. Exact location of this image point (**Image Point Gun**) is the middle of the right side of the avatar sprite, coordinates of x:48, y:21.
  Properties:
  - Speed: *Bouncing Bullet* moves 500 px per seconds. No friction or gravity.
  - Movement: *Bouncing Bullet* angle is same with the avatar sprites angle upon spawning.
  - Availability: At level start player is given 8 *Bouncing Bullet*s. This number is fixed among levels. Running out of bullets is a game over condition – if last bullet cannot connect with zombie sprite.
  - Bounces from "solid" objects: Bullets will bounce of solid objects - wall, hazard objects and interactable objects sprites. This bounce will be calculated depending of the bullet sprite angle of motion and the shape of colliding objects.
  - Destroyed outside of layout: If bullet gets out of playing area in any way, bullet sprite gets destroyed.
  - Interaction with zombie sprite: If bullet sprite collides with zombie sprite, both sprites get destroyed. This is the win condition of any level.



Boundaries in blue denotes the collision shape of the bullet sprite. Front of the shape is 5 px.



Example of a bullet bounce. Bullet sprite's entry angle is 45 degrees. It collides with wall sprite and exits with an angle of 315 degrees.

INTERACTABLE OBJECTS

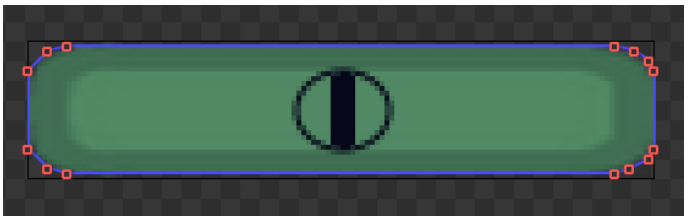
This is the secondary mechanic to help achieving the main goal. Player can interact with two types of objects on the game level and manipulate them in order to make the *Bouncing Bullet* reach the zombie. Also, player interaction with these objects increases Moves counter, which is an end score variable.

- *Rotatable Object* is a 128 x 28 px 2D sprite. While static (no horizontal or vertical movement), it can be rotated around itself, in real-time with mouse movements, by left clicking and holding either end of the sprite. When player released the left click, moves counter is updated with +1. Origin point is the middle of the sprite, coordinates of x:14, y:64.

  Properties:

  - *Solid* behavior
  - Availability: This object is available from level 1 to the end of the game.
  - Screw head like shape in the middle of the sprite imply that the sprite is rotatable around itself.
  - No random generation. Sprites will be placed by hand on level design.
  - Randomized angles: From level 3 onward, the angle of each object within the layout is randomized at the beginning of each new level.

> ***Solid*:** When an object sprite with a *Solid* behavior collides with bullet object sprite, bullet sprite will bounce from this sprite. *This is a universal behavior that is shared by numerous other objects.*



Outer shape in blue denotes the collision shape of the rotatable object sprite.

- *Draggable Object* is a 128 x 28 px 2D sprite. While it cannot be angled (angle will be determined on level design by hand), it can be dragged up and down (x coordinates is fixed, only y coordinate will change) by left clicking and holding anywhere on the sprite. When player released the left click, moves counter is updated with +1.

  Properties:

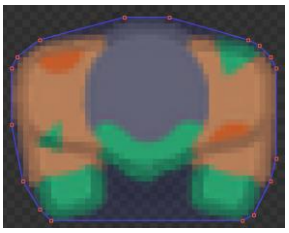
  - *Solid* behavior
  - Availability: This object is available from level 1 to the end of the game.
  - No random generation. Sprites will be placed by hand on level design.
  - Drag boundary will be determined by an invisible sprite placed behind the brown object.

Outer shape in blue denotes the collision shape of the draggable object sprite.

ZOMBIE

Zombie is a 48 x 42 px 2D sprite, the only destructible object in the game and the main in-game object to achieve main goal; Bullet must collide with Zombie sprite (which is dubbed as Zombu in-game). It is static, except for the tween movement which repeats every 1 second and subtracts a random number between -9 to 9 from the Zombie sprite's current angle to give a lifelike sensation to the Zombie. When hit, the Zombie sprite gets destroyed and is replaced by a 96x96 Blood sprite, followed by the playback of a "death" sound. In the event of any lose condition, a 1-second animation is played, depicting the Zombie sprite jumping up with its right fist raised in celebration. Simultaneously, a sound file, featuring a guttural and irritating laughter designed to unnerve players, is played.

Blood sprite

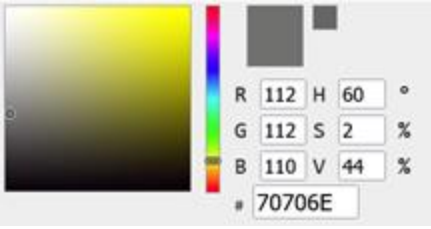


Outer shape in blue denotes the collision shape of the zombie object sprite.

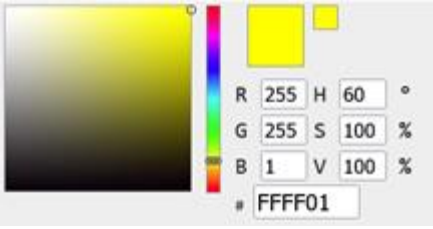
ENABLERS

- *Laser Sight* is a 4 px wide 2D sprite. Its primary function is to indicate the player's aiming direction. While it has a fixed width, its height is determined via ray casting methods. A ray is casted from Image Point Gun every tick (60 times a second). If a ray is intersecting a solid object, Laser Sight sprite height will be set as the distance between the Image Point Gun x and y coordinates to the x and y coordinates of the intersected object. Laser sight will not reflect from solid objects. Angle of the Laser Sight sprite is always equal to the avatar sprite.

- *Tracer Bullet* is a variant of the *Bouncing Bullet* sprite. It has the same characteristics with the *Bouncing Bullet* sprite except:
  - Tracer Bullet is harmless to the zombie
  - Game starts with player possessing 12 Tracer Bullet. Unlike *Bouncing Bullet*, it does not replenish at the start of the levels; Tracer Bullet count is fixed throughout the game.
  - Tracer bullet trail fade out time is 4 seconds compared to the *Bouncing Bullet*.
  - Tracer bullet trail color is different from the *Bouncing Bullet*.



Bouncing Bullet trail color code



Tracer Bullet trail color code

BLOCKERS

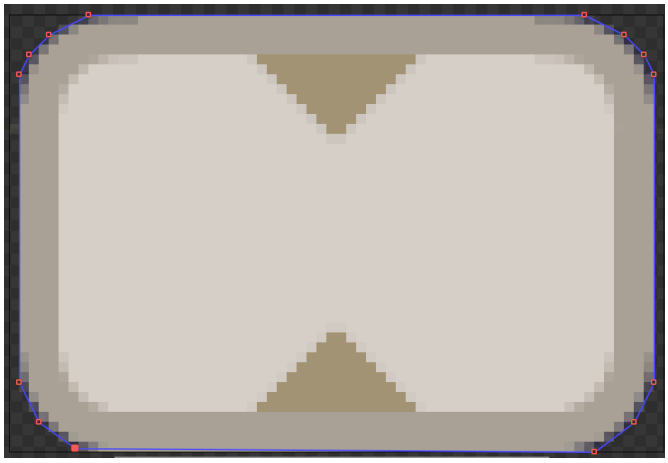
- Moving objects:
  - Downwards Object is a 102 x 272 px 2D sprite with *Solid* behavior. It moves in a linear motion with a jump back to start, from the top edge of the layout downward with a period of 7 seconds, resembling a sawtooth wave. The object starts its movement off-layout and restarts its movement as soon as it exits the layout on the bottom edge. Object angle is fixed.

Outer shape in blue denotes the collision shape of the Downwards Object sprite.

  - Upwards Object is a 102 x 27 px 2D sprite with *Solid* behavior. It moves in a linear motion with a jump back to start, from the bottom edge of the layout upward with a period of 7 seconds, resembling a reverse-sawtooth wave. The object starts its movement off-layout and restarts its movement as soon as it exits the layout on the bottom edge. Object angle is fixed.

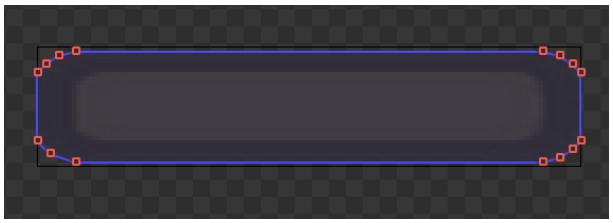
Outer shape in blue denotes the collision shape of the Upwards Object sprite.

  - Pendulum Object is a 66 x 44 px 2D sprite with *Solid* behavior. It moves back and forth, in the directions of left to right and right to left, in a period of 4 seconds in magnitude of 50 pixels, resembling a smooth oscillating motion based on a sine wave. The object starts its movement, where it has been placed on the level design, as soon as the level starts. Object angle is fixed.



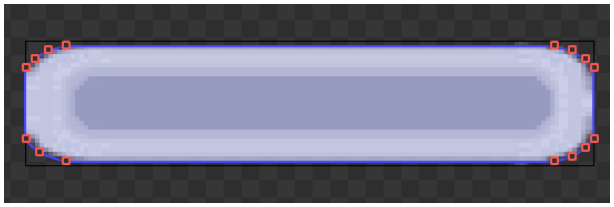
Outer shape in blue denotes the collision shape of the Pendulum Object sprite.

- Rotating Objects:
  - Clockwise Object is a 128 x 28 px 2D sprite with *Solid* behavior. It rotates around itself in a clockwise movement with a speed of 90 degrees per second, completing a full rotation in 4 seconds. Rotation begins at the start of each level and never stops.

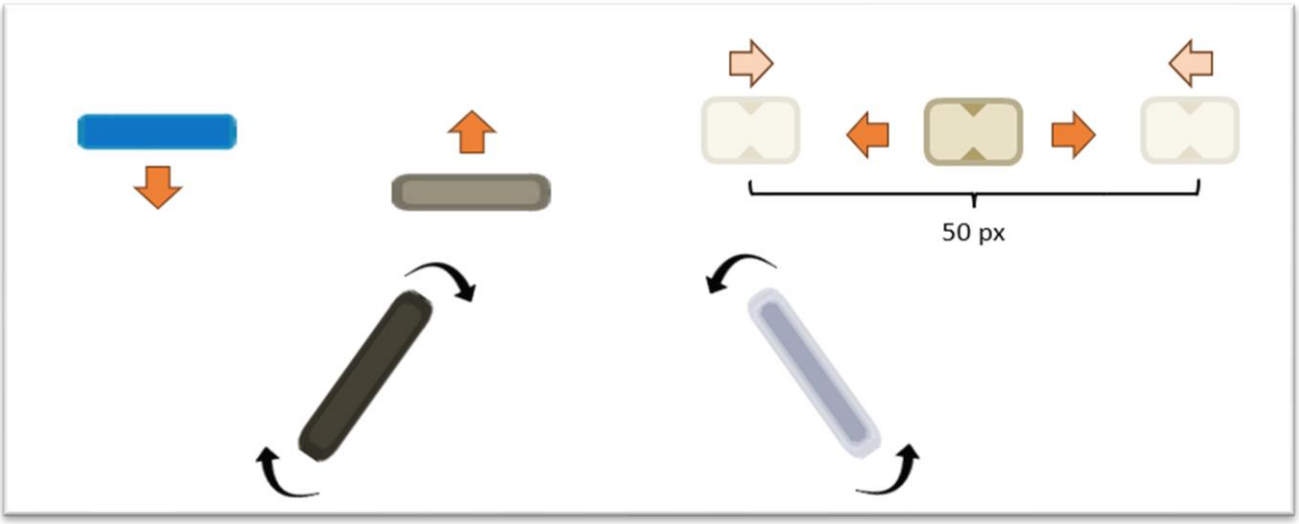


Outer shape in blue denotes the collision shape of the Clockwise Object sprite.

  - Counterclockwise Object is a 128 x 28 px 2D sprite with *Solid* behavior. It rotates around itself in a counterclockwise movement with a speed of 90 degrees per second, completing a full rotation in 4 seconds. Rotation begins at the start of each level and never stops.



Outer shape in blue denotes the collision shape of the Counterclockwise Object sprite.

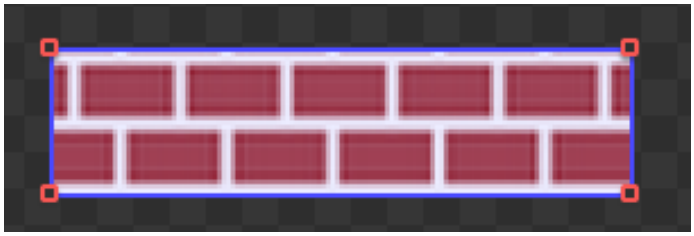


- Time Limit: Each level has a time limit to put pressure on the player and create a sense of tension. It is shown as a decreasing value; at the beginning of the level full time is shown and it decreases over the gameplay. The time limit kept with Integer values, varies from level to level, as indicated in the table below.

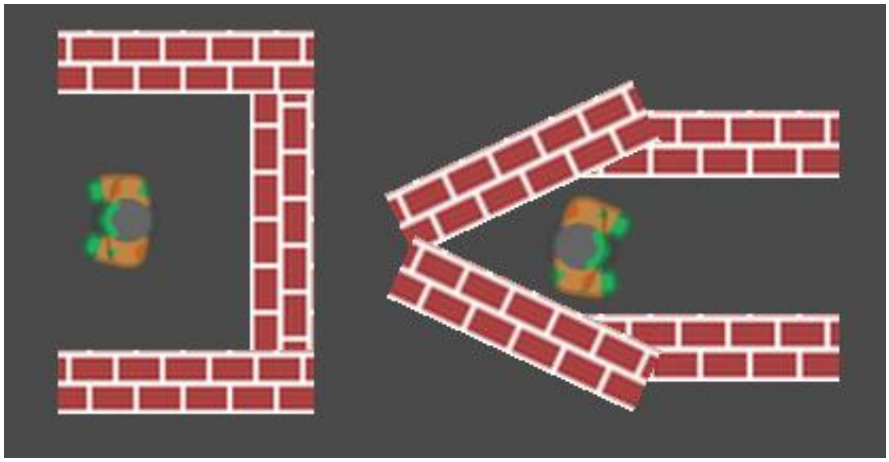
Levels	1	2	3	4	5	6	7	8	9	10
Time Limit	80 sec.	100 sec.	120 sec.	120 sec.	120 sec.	120 sec.	120 sec.	150 sec.	150 sec.	150 sec.

- Limited Ammo: As mentioned in their respective mechanics sections, the player is provided with two types of ammo: the *Bouncing Bullet* and the *Tracer Bullet*. These ammos are limited to add pressure to the player and create a sense of tension.
    - *Bouncing Bullet* is limited to eight. It replenishes start of each level.
    - Tracer Bullet is limited to twelve, and it is fixed throughout the entire game session (no replenishment). The player has to strategically decide whether to use the tracer or keep it for later levels.

- Wall is a 128 x 32 px 2D sprite. Walls are the main-level design object with Solid behavior. They are completely static; there is no movement whatsoever. Although primarily designed to shape the level layout, it can be used as a blocker for secondary usage.



Outer shape in blue denotes the collision shape of the Wall sprite.



Examples of blocker usage of Wall sprite

Score System

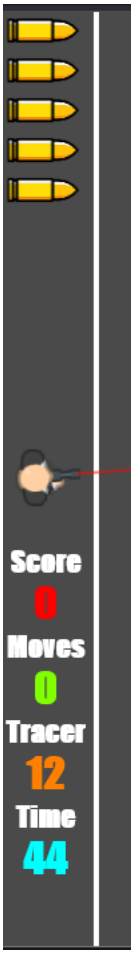
After successfully completing each level, the player is awarded score points. These points are cumulative throughout the game and calculated with two positive and two negative values and kept as an Integer:

- Remaining Bullet count (RBC) is a positive value. Using fewer bullets yields more points.
- Remaining time count (RT) is a positive value. Finishing earlier yields more points.
- Move count (MC) is a negative value. Less manipulation of *Interactable Objects* yields more points.
- Tracer Bullet usage in the current level (TBU) is a negative value. Using fewer Tracer Bullets yields more points.

The score is calculated with the formula below:

$$Score = Previous\ Level\ Score + (5xRBC) + RT - (2xMC) - (3xTBU)$$

USER INTERFACE, ART AND ASSETS



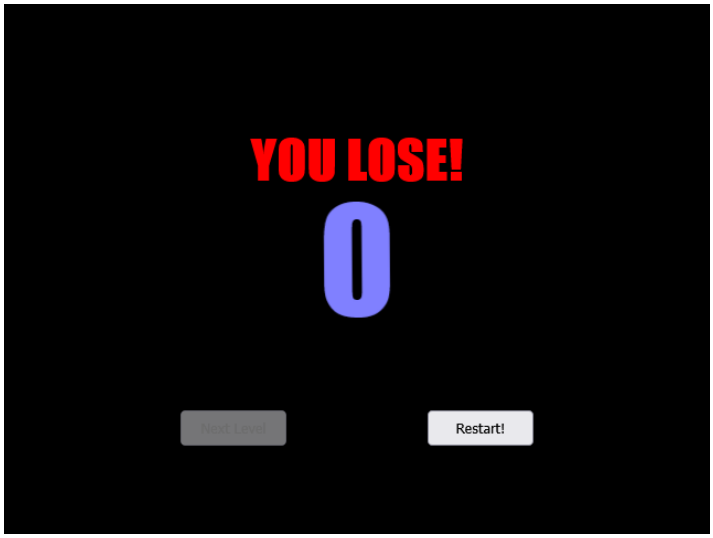
The HUD is composed of four elements: Remaining *Bouncing Bullet* count, current Score, Moves count, and remaining Time count, along with *Tracer Bullet* count. These elements are positioned at the leftmost edge of the layout, as represented in the image on the left. They will be placed around the player avatar, with the Remaining *Bouncing Bullet* count (represented by bullet sprites) positioned on top of the player avatar and the others below. The rationale behind this arrangement is to provide maximum space and visibility to the gameplay area while utilizing the unnecessary space above and below the player avatar.

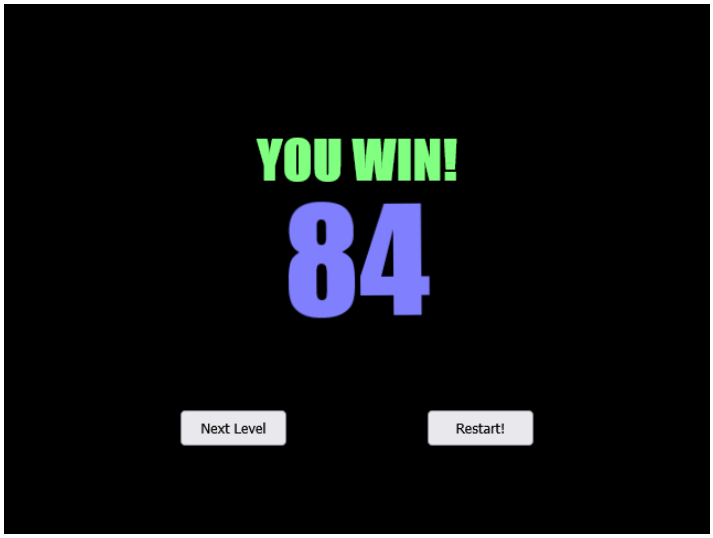
The game begins with a logo (which can be seen on the right) featuring a pseudo-TV screen effect, displaying the colors of the Zombie Interdiction Agency. These colors, inspired by those of the Central Intelligence Agency in real life, represent the current color scheme. The art style should reflect a cold, government-like aesthetic, and this logo serves as a good example of that style.

All assets will be taken and used from Kenny.nl webpage.

https://www.kenney.nl/assets/top-down-shooter

The Victory screen and Game Over screen can be seen in the following images. The Victory screen also features an animation resembling fireworks, which are spawned in random locations six times.





LEVELS

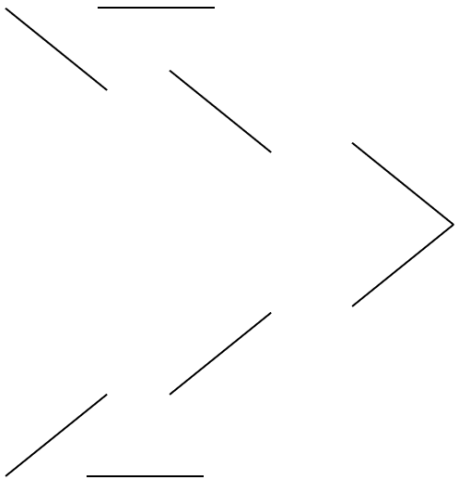
There are ten stages in the game. Every level will be manually designed; each of them will be crafted by hand without any random generation or placement. Every level has 1120 x 704 px layout dimensions, with the exception of the first three, which have layout dimensions of 704 x 704, 832 x 704, and 992 x 704 px, respectively. Each level will have several ways to win. These will be either intentionally designed -each level will feature a designed solution with a direct and minimum number of moves- or unintentional; random interactions between the bouncing mechanic, blocker movement and alike.

The first level is the easiest and has the smallest size; it will act as a tutorial level. It will consist of only a Rotatable Object and a Draggable Object. Zombie, will be placed directly against the player avatar but it will be surrounded by walls. Four wall segments will be placed to block direct shots, forcing the player to utilize the bouncing mechanic through the only opening on the wall. Before taking a shot at that opening, the player has to manipulate the Rotatable Object and Draggable Object, as they obstruct the path to the opening. Three big direction arrow shaped sprites will act as tutorial objects. Two will be pointed at the interactable objects with accompanying text demonstrating how to operate them. Third one will make a looping tween movement at 1.5 seconds interval and pointing the precise location to shoot in order to hit the zombie. The text displayed on this sprite will convey this information.

The second level will consist of several Rotatable Objects and Draggable Objects, with four of each object. Like in the first level zombie and player avatar will be placed directly against themselves. For the first time, the Player has to use the interactable objects shoot the zombie, because a vertical big wall segment will be placed in front of the player avatar. And also, zombie will be surrounded by walls, a segment will be placed to its back, right and left, forming a reverse “C”. This “C” placement will make also players job easy by creating a bullet trap; The bullet will end up hitting the zombie after bouncing several times between the walls.

The third level will introduce the Downwards Object, which will be placed in the middle of the level. As in the first two levels, the zombie and player avatar will be positioned directly opposite each other. Two horizontal wall segments will be placed in front of the player to prevent direct shooting of the zombie. Two wall segments will be placed behind the zombie, positioned at a 90-degree angle to each other, forming a “>” shape. There will be four Rotatable Objects and three Draggable Objects.

The fourth level introduces a new layout dimension of 1120 x 704 px, deviating from the previous ones. Subsequent levels will maintain this size. It is also the first level that features Upwards Object which will be placed next to the avatar location. Zombie and the player avatar will be positioned directly opposite of each other. Level will consist eight interactable objects, equally distributed among each type. Two wall segments will be placed in front of the zombie to prevent direct shooting, positioned at a 90-degree angle to each other, forming a “>” shape. Four wall segments will be placed up and down of the level layout, with several px wide spaces between them. They will be placed between the player avatar and the “>” shape wall segment, with the angles of 45 degrees for upper wall segments and 315 degrees for the rest, to create a pine tree like shape. An approximate placement shown on the right:



Fifth level will not introduce a new blocker. However, it is the first level that zombie is placed differently from the preceding levels. It will be placed top of the layout, surrounded by a “C” type wall segment placement. Two upwards objects and one downwards object will be placed next to the player avatar placement, with several px spaces between them. They will be placed in order of Upwards-Downward-Upwards respectively. Multiple wall segments will be placed to the right-hand corner of the layout, with six interactable objects, equal of each type.

Sixth level will introduce the Clockwise Object, which will be placed directly against the player avatar and the zombie with the y axis coordinate of “352”. Zombie will be placed bottom side of the layout, it will not be surrounded or blocked by wall segments. Huge spaces will be present to its right and left. This level will utilize heavy usage of wall segments to limit playground and create boundaries. Six interactable objects will be placed. Five of them will be Rotatable Object and one Draggable Object. One Upwards Object and one Downwards Object will be placed in the middle of the layout next to each other.

Seventh level will not introduce a new blocker. Like the previous one, this level will also utilize heavy usage of wall sprites. Horizontal wall segments will be placed to top and the bottom of the layout in order to create up and down boundaries. A huge pixelated “C” like placement will be placed in front of the player, which can be shown to the right. Zombie and player avatar will be placed directly opposite of each other. Zombie will be placed in the middle of the “C” like wall segment placement. A Clockwise Object will be placed directly against the zombie and the middle of the layout. Six interactable objects will be placed to the right side of the layout, equal in numbers of the both objects.

Eight level will introduce Counterclockwise Object. One Counterclockwise and one Clockwise object will be placed to the layout. Zombie will be place directly opposite of the player and the very middle of the layout. It will be surrounded by a reverse “C” like wall segment placement. This segment will also be surrounded by a big “C” like wall segment placement, facing the opposite direction. Ten interactable objects will be placed to the right side of the layout, after the x:576 and y:352 coordinates. Four Draggable Objects and six Rotatable Objects will be placed. No wall segments will be placed to the right-hand side of the layout (after the x:576 and y:352 coordinates), all the edges will be blank.

Ninth level is the last level to introduce a blocker; the Pendulum Object. Unlike the eight level, this level will utilize heavy usage of Wall segments. All edges will be covered with Wall segments to create boundaries to the gameplay area. Also there will be Wall segment placements like hedgehogs to the left side of the layout (before the x:576 and y:352 coordinates), bottom and the top. Zombie and the Player Avatar will be placed directly opposite of each other, almost next to them (they will be four Player Avatar unit pixels apart from each other. Zombie will be surrounded by a directional arrow like Wall segment placement. This placements point will be directed to the player. One Pendulum Object, one Clockwise Object, one Counterclockwise Object, six Rotatable Object and two Draggable Object will be placed unto the layout. Blockers will be placed to the left side and the Interactable Objects to the right side of the layout.

Tenth level will be the last level. All blockers and no Wall segments will be used. Zombie will be placed directly opposite of the player and between one Clockwise Object and Counterclockwise Object. Two Upwards Object and two Downward Object, four Pendulum Object, six Rotatable Object and five Draggable object will be placed upon the layout.