Computer Games Software Technology
The University of Cyprus
May 2021

Yiangos Georgiou
Eleni Evripidou

# Chicken Escape

Game Design Document

*Yiangos Georgiou, Eleni Evripidou*

## Story

A chicken has successfully escaped her coop and now she is on a mission to rescue her chicken friends from other coops. Along the road she must avoid obstacles that make her dizzy and unable to continue her mission. She also must collect fruits to keep her motivated and have enough strength to complete her mission. Most importantly she must collect golden eggs to rescue her chicken friends.

## Character

The character of our game is a low-poly chicken, that was designed, sculpted, and animated using Blender by Eleni. The rig of our character includes running, flying, dizziness and idle state animations. Our character's design is displayed in Figure 1.

## Mechanics

This game is in the category of the endless running games. There are no stages, but the game can get more difficult as the hero moves closer to the goal: to collect the requirement number of golden eggs and reach the spot where it can rescue other chickens. The difficultness increases by increasing the speed of the hero and the amount of static and moving obstacles along the track.

### Character Movement

In the game, our hero can be controlled by the user to go left using left arrow, right using right arrow and jump using spacebar key in the keyboard. The forward movement of the character is controlled by the game. When entering a turn, the forward vector of the gameobject that includes the chicken character is adjusted to be aligned with the forward vector of the road. This is the reason we created separate prefabs for a left and a right turn since the forward vector of each differ. This way, the game guarantees that the player cannot go backwards to collect missed items. Lastly, we implemented our game so that the chicken can only jump when it is grounded.

### Skills

Our hero can go into strength mode. During this time, the character can break through obstacles without losing any lives and collecting ten points. It is also able to collect golden eggs inside wooden cage by breaking through the cage. It can also jump very high to collect the golden egg that is in a higher altitude, by using the up-arrow key.

### Animation

Our character is in the running state when is on the ground. When it does not touch the ground is in the flying state. It can go to the dizziness state from any state when it hits an obstacle, expect the rotating obstacle.

### Physics

Our character is a rigid body, so it can collide with objects that have the collider element assigned to, such as obstacles, ground blocks, and fences. Since the character is a rigid body, gravity is affecting it.

Yiangos Georgiou
Eleni Evripidou

*Figure 1: Different animation states of the main character. Starting from left up is a frame captured from the idle state, right up from the flying state, left down from the dizziness state and right down from the running state.*

Our world has a gravitational field with gravitational acceleration equal to g=-50m/s^2. In a force field similar to that of earth the jumping and landing speed was not quick enough for our taste, hence we changed it.

## Camera Movement

The camera is a third person camera, following our hero. It is implemented using Cinemachine provided by Unity. The forward vector of the camera is used to align the character's movement direction with the user input keys. Also, the camera is avoiding objects that are in front of character by preserving the distance from the character, hence changing its height, and not pulling forward. We adjusted the parameters of the camera that are responsible for how gradually the camera returns to its normal position and to resolve an occlusion to our taste.

## Collectables

There are three categories of collectables: fruits, golden eggs, and potions that are provided by a free asset package in unity asset store. Fruits are consisted by apples, bananas, carrots, cherries, watermelons, and pumpkins. The goal of the chicken is to collect as many fruits as possible. Each fruit is valued as one point in the score. Furthermore, by collecting a specific pattern of fruits that the game asks, the chicken goes into strength mode. The category of golden eggs includes a general golden egg, a golden egg inside a wooden cage and a golden egg in a higher altitude. The last category of collectables is the potion collectible which gives the chicken an extra life. All collectables are spinning and oscillating along the y-axis.

Yiangos Georgiou
Eleni Evripidou

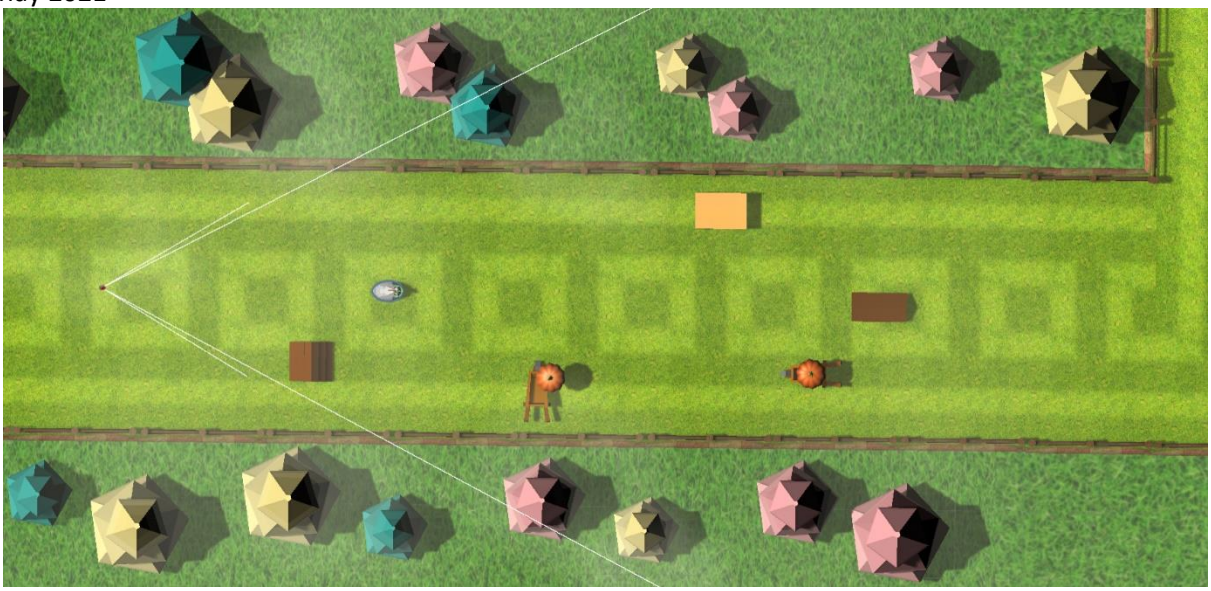

*Figure 2: Demonstration of ground, décor, collectables and obstacles spawning.*

## Obstacles

There are two categories of obstacles: static and moving. The static obstacles include a variety of wooden objects that are provided by a free asset package in unity asset store. To have a variety in our game, the orientation of obstacles is randomized between two faces. Only for the obstacle that represents a wooden wheelbarrow its orientation is randomized between three of its faces. The moving obstacles can be a rotating wood that can spin clockwise or anti-clockwise with different speeds or a moving crate that oscillates in a direction perpendicular to the forward direction of the road with different speeds. The technical difficulty faced here was that since the track has turn, the changes along x and z axis are not the same each time. Hence, the forward and right vector of the ground tiles was used to find the desired orientation of the obstacles. When the chicken hits static obstacles it goes into the dizziness state, but when it hits moving obstacles its mesh blinks (not its rig). For blinking, we used the StartCoroutine function calling an IEnumerator function. After hitting the obstacles, they disappear because otherwise the chicken can hit them again when it wakes up.

## Track

The track is created dynamically. It is consisted by a collection of straight, right turn and left turn block. The blocks were created using grassed ground blocks provided by a free asset package in unity asset store and were adjusted to our demands. The chicken is not able to go out of the track due to colliders. But if it finds a way to do so, the game ends.

## Ground Spawn

The local coordinate system of the ground blocks and the next spawning point are adjusted so that each new block is aligned with the previous block. We randomly choose to generate one of the following types of ground tiles (1) straight, (2) left turn and (3) right turn. Incorporating turns in an endless run needs a global rotation that change the tile direction accordingly. Moreover, to avoid collision created by generating the same turn consecutive times we add a patience threshold to left and right ground tiles. A fixed number of tiles can be activated concurrently in the game. In our first implementation, when chicken reached the end of a tile a new tile was generated and the previous was destroyed after some time. However, we noticed that when the player was hitting an obstacle and the dizziness animation was played, the tile that previously passed through, was destroyed and a

Yiangos Georgiou
Eleni Evripidou

*Figure 3: Demonstration of barn and chicken spawning when the chicken successfully collected the golden eggs*

gap was shown in the scene, or even in some cases the chicken fell out of the track. Hence, we solved this issue by destroying the ground blocks that the chicken has passed through, when the camera exits them and generate new ground blocks when the chicken exits. This procedure continues until the player finish the game.

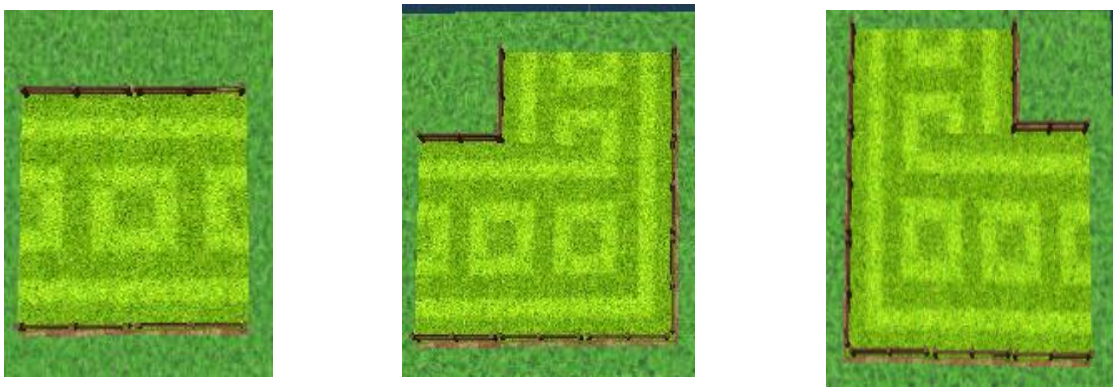

*Figure 4: Ground block prefabs: straight, left turn and right turn tile.*

Ground tiles features: (1) running area (area inside the fences): in this area chicken can move, also obstacles, collectables and eggs are spawned. (2) Décor area (area outside the fences): We keep a small area outside of the fences that is dedicated to randomly spawn decoration such as trees. (3) Next spawning ground tile position (better seen on Ground tile prefab): Each tile contains the position of the next ground tile as we need to avoid gaps and misalignments between adjacent tiles. (4) Invisible colliders limits (better seen on Ground tile prefab): We used colliders located in the same position as fences to prevent the chicken to jump outside the track.

Yiangos Georgiou
Eleni Evripidou

## Décor Spawn

Each ground block has trees on the right and left side, that can be in different colors and sizes and are spawn randomly. Ground Tile contain décor positions that randomly are being activate and assigned trees with different size and color.

## Collectables and Obstacles Spawn

For each object we randomly compute an offset from the center of the ground tile to position the new object. Given that in contrast with other popular running games (temple run, subway surfers) that split the ground area to discrete parts our continuous running area implementation raises several overlapping issues between objects that are being spawned on the ground tile. We follow a collision avoidance strategy that considers the distance between objects for x,y and z axis separately to detect collisions, if one of the distances exceeds the collision distance threshold then the new object is spawned.

## Barn Spawn

The barn is spawned in the middle of the track when player collect the target number of golden eggs to finish the game. The barn is provided by the unity asset store for free.

## Chickens Spawn

When the chicken reaches the barn safely, chickens are spawned. The chickens can be walking or running, in random directions between vectors that are directed towards the user, with random speeds. These chickens are destroyed after some time passes. Those characters and their animations were provided by the unity asset store for free.

Yiangos Georgiou
Eleni Evripidou

*Figure 5: Demonstration of chicken going into superpower mode. Rings are appearing around the character. In the picture, the obstacle breaking simulation is also demonstrated.*

## Super Chicken Mode

Super chicken mode is a feature that urge players to collect a specific pattern of collectables to activate the super abilities to gather golden eggs that are inside a cage (have the strength to break objects) or are higher than a normal chicken can jump (can jump higher than normal). When the pattern is complete the chicken enters superpower mode that last for a specific period after that the pattern is changed randomly, and our chicken returns to its normal form. Our chicken indicates that super mode is activated by changing its colour. When the chicken exits the super mode, the chicken mesh blinks so that the user would be alerted.

## Particle Systems

Obstacle Breaking: We created a simulation for the movement of pieces of wood when breaking through obstacles. The direction of the particles' generation is inside a quarter sphere faced in the forward direction of the ground block. The particles' have randomized different starting sizes and rotations and are affected by gravity so that they do not fly in the air. Also, the particles' texture is assigned to be a picture of a wood surface (Figure 5).

Strength Rings: The rings that appear when chicken enters the strength mode are also a particle system that was provided for free from the unity asset store and was included in the same package as the ground fog. However, this simulation was for the purpose of respawning, so by tweaking the time duration of the simulation we got what we wanted (Figure 5).

Fog: We used fog for 2 reasons. The first was to hide the ground generation from player at the start of the game and the second was to improve game atmosphere by hiding low quality textures of the large background terrain. The simulated fog used, was provided for free from the unity asset store (Figure 6).

Yiangos Georgiou
Eleni Evripidou

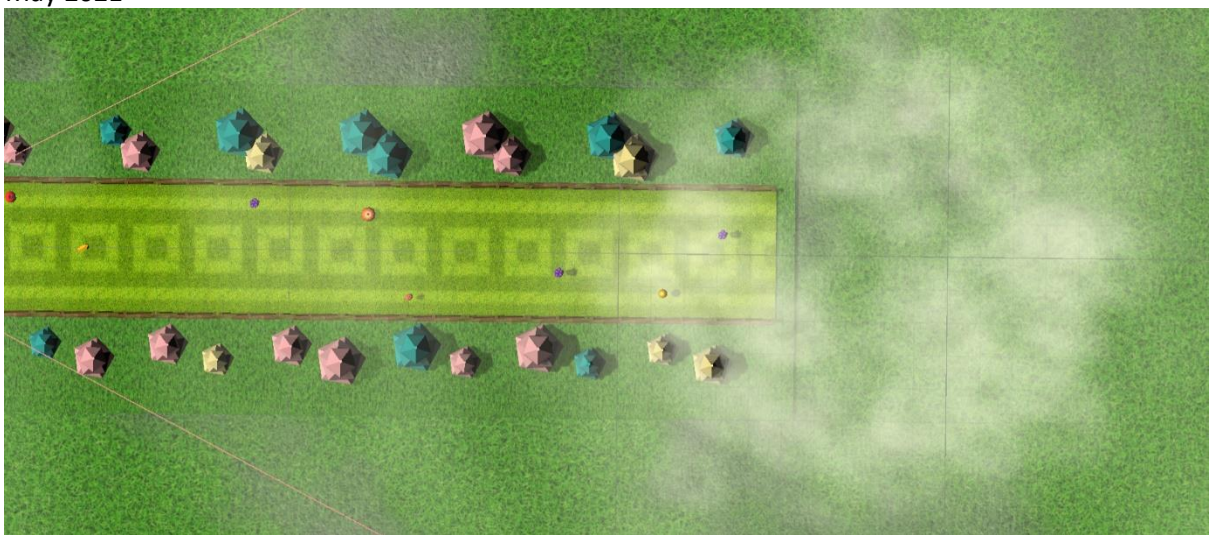

*Figure 6: Demonstration of the use of fog at the start of the game, to hide the gap from the player.*

## User Interface

Start Menu: Our start menu appears at the start of the game. The user can choose to play, or to go to settings to chance the volume of the music of the start menu or quit the game. When the player loses or wins the game, the start menu appears.

Pause Menu: Our pause menu appears when the user hits the escape button. From there, the user can decide to go back to the game or go to the start menu or to quit the game.


*Figure 7: Demonstration of the player status during the game. In this case our character has five remaining lives, has collected one regular egg and the first type of fruit to go into strength mode, which is an apple.*

Current Status Screen Display: During the game, the status of the player is shown. For example, the remaining lives and the maximum number of lives that the player is able to have, are shown on the left-up corner. The current score is displayed on the right-up corner. Furthermore, the pattern of fruit types that the player must collect to go into strength mode, appears in the left-down corner and the pattern of golden egg types in the right-down corner. Each time the user leaves out

the strength mode, a new fruit pattern appears on the screen that they must collect. Since the pattern changes during the game, we created clones of the images that represents the desired fruit types, to appear onto the screen. The items that the player has not collected yet appear with a darker color and those that they have in their possessions appear with a lighter color. (Figure 7)

### Scalable Implementation

Implementation follows a scalable structure as we can modify different features accordingly to the scenario and the difficulty that we want to achieve. Regarding the ground tiles, we can control the number of obstacles spawned, type of obstacle that exist and type of collectables which makes as able to modify the game dynamically. Moreover, the size of the pattern for super chicken abilities can be increased as well as the speed of the chicken. Also, new obstacles and ground styles can be incorporated to enrich the game in future versions.

### Demo-Gameplay presentation.

Our demo consists of a scenario that our chicken must collect 6 eggs. The difficulty of the game increase based on the number of collected eggs. To adjust the difficulty, we modify three important features of the game, namely, chicken's movement speed, number of obstacles per ground tile and the pattern completion that is becoming necessary to progress with the demo scenario after our chicken collect the second golden egg. The number of obstacles per ground tile increase by 1 every 2 golden eggs and the speed of the chicken increase with the same amount every time our chicken collects an egg. Egg gathering in the beginning (first 2) is easy as golden eggs are spawned in convenient positions that a normal chicken can gather, after that to collect the rest 4 golden eggs our chicken must collect specific collectable to activate super abilities to break crates that contain eggs and jump higher than normal to catch eggs that are too high.

## Appendix

### Link for Project:

https://github.com/eevrip/Chicken-Escape-Mini-Game

### Links for Assets

Chickens: https://assetstore.unity.com/packages/3d/characters/animals/meshtint-free-chicken-mega-toon-series-151842

Barn-Trees-Obstacles: https://assetstore.unity.com/packages/3d/characters/lowpoly-simple-house-117348

Road: https://assetstore.unity.com/packages/3d/environments/roadways/grass-road-race-46974

Potion: https://assetstore.unity.com/packages/3d/environments/fantasy/fantasy-treasure-pack-lite-80898

Yiangos Georgiou
Eleni Evripidou

Fruits:

https://assetstore.unity.com/packages/3d/props/food/3d-props-adorable-foods-31249

Audio:

https://freesound.org/

https://assetstore.unity.com/packages/audio/sound-fx/free-casual-game-sfx-pack-54116

Fog and Strength Rings: https://assetstore.unity.com/packages/essentials/tutorial-projects/unity-particle-pack-127325