**Computer Games Development CW208**

**Year IV**

**Games Design Doc**

Shane Conroy

c00270031

11/05/2025

**Contents**

Game Overview...........................................................................................3

Key Gameplay Systems..........................................................................3, 4

Gameplay Loop.......................................................................................4, 5

Game World.................................................................................................5

The Physical World.........................................................................6, 7, 8, 9

Rendering.................................................................................................. 10

Game Engine....................................................................................11, 12, 13

Characters............................................................................................13, 14

User Interface.......................................................................................14, 15

**Game Overview**

My game, “Save our Sheep,” is a farm management game with the main objective of earning as much money as possible and upgrading your farm’s buildings. The player does this by shearing sheep and selling their wool. It’s not as easy as it seems because at night wolves will attack your farm, hunt your sheep and eat them!

**Key Gameplay Systems**

Sheep Behaviour

* Flocking
* Dynamic Leader assignment
* Eating grass and sheering wool
* Prestige system and Genetic combining
* Reacting to influences (Wolf, Dog)

Wolf Behaviour

* Comes out at night
* Despawn during day
* Hunt sheep at night
* Dies if eats an infected sheep

Dog Behaviour

* Follows mouse
* Right click to barks
* Herd sheep
* Stun wolf

Grass System

* Spawn when bought
* Despawn when eaten

Economy System

* Buy and sell sheep
* Buy grass for sheep
* Sell wool for income
* Buy farm upgrades
* Max amount of sheep
* Wool price
* Grass bought at one time
* Sheep bough at one time

UI/HUD

* Pop out panel with buttons
* Buttons for buying/selling/upgrading
* Current funds count

Combining

* Combine sheep based off greatness
* New Sheep has combined stats
* Prestiges sheep

Particles

* Particle effect when sheep die, wolf dies, and sheep sold
* Visual numbers when picking up wool

**Gameplay Loop**

The Player will start off with no sheep and $1000. This is enough to buy some starter sheep and some grass.

When the Player is ready, they will open the fence and let the sheep out. Once out the sheep will either wander around or seek grass, depending on whether there’s grass or not. After eating grass, the sheep will grow out their wool.

Then the Player can sheer the sheep, collect the wool and earn money. After collecting enough money, the player can upgrade their farm, which will help them earn money faster and more efficiently.

Once night arrives, wolves will spawn and seek out sheep. The Player, who controls the sheepdog must stun the wolf and try save their sheep from the wolf.

After night, the wolf leaves and the cycle continues.

**Game World**

Overview

* The game’s world is set in a barren, rundown farmland. The farmland is split into two main parts, the field and the pen. Although the farm starts neglected, with no sheep, decrepit buildings and no edible grass, the player can restore the farmland to a proper, functional farm.

The pen

* The pen is the sheep’s safe zone away from the wolf. When a new sheep is bought, they will spawn in here and during the night will hide in here away from the wolf, with the Player’s help.

The field

* Outside the pen is the main field, this is where the sheep will spend most of their time. They will either graze or wander depending on the availability of grass. However, being out here exposes the sheep to the wolves.

The rundown buildings

* At the edge of the field, there’s rundown buildings. They might not look like much, but the player can repair these buildings and use them to majorly benefit their farm.

**The Physical World**

The physical world of the game is a compact, enclosed farm designed for economy management and herding. It’s based off a small farmland. It is intentionally simple with it’s layout directly serving the game’s core mechanics. Though visually minimal, every area has a purpose.

-Key Locations-

The Pen

* The safe zone for sheep. They spawn here when purchased and will return here during a recall.

The Field

* The larger outer area where grass nodes spawn when bought. Sheep mainly stay out here to graze. It’s the main source of income but also a high-risk zone at night due to wolves.

Ruined Buildings

* Old farm structures like the barn, loom, market, and garden are placed on the field’s outskirts. These are gradually upgraded by the player to enhance their productivity.

The Combiner

* Unique machine that takes in sheep, combines them and spits out a better, prestiged sheep with improved stats. There is a very low chance the Combiner will mess up and spawn an infected Sheep.

-Travel-

Sheep

* For the sheep, the player doesn't directly move the sheep, rather the sheep will move autonomously using flocking to mimic a flock of sheep. They will follow their leader or look for grass if they're the current leader.

Dog

* The dog is controlled by the Player. The Dog follows the mouses current position on screen.

Wolf

* The Wolf enemy will seek out the closest sheep and go directly for them.

-Scale-

The game is bound to one screen, with everything available on screen.

The game is played from a top-down view of the game. The camera is quite high above the field, so the animals look much smaller as to fit them all on screen.

The pen takes up the bottom 1/3 of the screen, the field taking up the rest of the 2/3s.

-Objects-

* Sheep
* Main entity. Eats and can be sheered
* Grass nodes
* Sheeps main food source
* Wolves
* The game’s enemy. Eats sheep
* Dog
* Herds sheep and stuns wolves
* Fence & Gate
* Used to contain the sheep
* Farm Buildings
* Upgrades the player’s game

-Day and Night-

The game has a day and night visualised by the shifting background colour. The background goes from a bright green to a darker green-blue, indicating the time of day.

-Time-

Minute 1

Player will buy sheep and grass.

The sheep will graze

The player will sheer the sheep and earn cash

Minute 2

The Player will use their earned money to buy upgrades

This speeds up production

Minute 3

Wolves will attack

The Player will stun the wolf

The Player will try and keep sheep alive

**Rendering System**

-Overview-

The game uses SFML window to render all in game visuals. All entities are made up of sf shapes. I am using sprites for the UI and the buildings.

-2D/3D Rendering-

The game is entirely 2D and uses SFML’s immediate rendering with shapes and some textures. There is no 3D in my game at all.

-Camera-

The camera is static and does not move at all during gameplay.

**Game Engine**

-Overview-

I don’t use a typical game engine like Unity or Unreal. I’m using SFML, a graphics library with my own custom logic and core systems. It has core classes like “World” which houses most of the internal logic of my game.

-Game Engine Details-

The engine is responsible for tracking and managing everything such as:

* Sheep management: Position, behavior state, spawning stats
* Wolf: Spawning, hunting, collision with sheep, and stun/death state
* Grass management: Spawning and collision check with Sheep
* Economy system: Tracking funds, upgrades and purchases
* HUD system: Cash, buttons, building upgrades
* Particle and visual effects: Floating text, wool particles
* Day/Night cycle: Manages colour changes and triggers wolf spawn

-Collision Detection-

I use 3 main types of collision detection in my game, each serving their own purpose, AABB, Radial and .getGlobalBounds.

AABB

* The first is AABB which I mainly use for clicking on buttons as the buttons a perfectly square. I just check the mouse’s position off the AABB of any of the buttons and see if there’s an overlap.

.getGlobalBounds

* The next is SFML’s .getGlobalBounds which is very useful for checking the collisions of my entities. If I need to check if a wolf is in contact with a sheep, I can use .getGlobalBounds to check for a collision.

Radial

* The last one I use is a radius-based check. I use this mainly for the sheep and grass, but I also use it for the Dog’s bark and the wolf, to see if the wolf ‘heard’ the bark. I use radius-based checks for sheep as grass nodes can spawn anywhere around the sheep by chance, so I have a radial check around the sheep.

**Game Characters**

-Overview-

My game doesn’t have characters, per se. Unless you consider mindless sheep or wolves “characters.” There isn’t a ‘Player’ character either, the closest thing is the Sheepdog, which loyally follows your mouse to help herd sheep and scare wolves.

Player

* The Player doesn’t have a physical body in game, but still controls how many sheep are bought, what their cash is spent on. They will also try and save their sheep from attacks.

Sheep

* The Sheep has different variations. There’s a gold sheep which when sheered will reward a lot more cash than normal sheep but can’t reproduce.
* Another variation is the infected sheep. The infected sheep will die after a certain amount of time and doesn’t listen to any commands, they may seem useless, but if the wolf eats an infected sheep, it will die and drop a cash orb, rewarding lots of money.
* Then there’s the default sheep, basic and listen to commands. Eats grass and gets sheered. A simple animal, really.

Dog

* A brown, loyal dog. Follows the mouse like his little life depends on it. Barks to scare the wolf and can herd the sheep.

-Enemies and Monster-

Wolf

* The wolf comes out at night, hunts a sheep, eats it then moves onto the next. The wolf will keep eating either until morning or unless it eats an infected sheep, in that case, the wolf dies, dropping a handy cash orb.

**User Interface**

Main Elements

* Current funds are tracked with a money counter in the top right of the screen, it goes up and down depending on purchases or selling stuff.
* The Pop-Out tab is at first hidden on the left of the screen, but if the player hits “Tab” it pops out revealing more buttons.

Pop-Out Tab

* Once opened, the tab expands onto the screen and more additional buttons for gameplay are shown. There are 8 main buttons, 4 core and 4 upgrades.

Core Buttons

* The core buttons are the ones the player will mainly be using. These include the Sheep Buy button, Sheep Sell button, the Grass buy button, all pretty self-explanatory. The last core button is the Combine button, when clicked it will combine the top sheep to create a new prestiged sheep if requirements are met.

Upgrade Buttons

* The upgrade buttons are used for upgrading the derelict buildings on the farm. They include the barn, market, loom, and garden.

Additional Tools

* Inside the pop-out menu is one more button, the “unstuck” button, a helper tool used if the current leader gets stuck somewhere. It’ll return the leader to 400,400, the centre of the field.
* The sheers button is underneath the current funds tracker. Once clicked it’ll place sheers into the player’s hands and allow the sheering of sheep.
* The whistle button will recall all sheep into the pen, if they can hear it.

Feedback and Visuals

* Other non-button UI elements include floating text when the player picks up cash from wool or cash orbs.
* Cooldown displays. When clicking buttons there’s visual feedback of the button greying out for a second to let the player know they hit the button.
* The panel popping in and out. This was a way of keeping the UI tidy and not having it too cramped while still allowing me to put many buttons on screen.