Build-a-Maze

Technical and Game Design Document

Design

Brief description:

Construct a maze and build up your defenses against an evil group of starving villagers trying to steal your sheep. Build mud turrets pitfalls and more to try to stop the unending horde from stealing your valuable livestock. Whilst also managing your balance in the farmers market to allow you to buy more defenses and livestock.

Maze:

Everything inside the maze walls is editable the player can construct a kill zone inside of the maze and the only limit is the players imagination and their bank balance. Each wave gifts the player more money which leads to the player upgrading their maze and building an unstoppable death trap for all the poor villagers trying to steal the sheep.

Art:

All art in the game was made by both designers, from the villagers to the livestock to the traps. The only visual part we didn't make is the font in the game.
All traps except the mud are animated, along with every character in the game being animated.

Farmers Market

Pitfall:

A pitfall trap can be purchased by the player and placed on any grass tile in the maze and when the game starts the player can click on the pitfall to open it and can trap the villagers inside it.

Steppingstone:

The steppingstone trap can be purchased and placed by the player on any grass tile and when a villager walks over it there is a percentage chance that they will fall into the water whilst trying to jump to the tile.

Mud:

Mud can be purchased and placed by the player and will slow down the villagers in their attempt to steal your sheep.

Treadmill:

The treadmill can be purchased and rotated by the player as to which direction they would prefer it can be used to speed up or slow down the villagers and can be turned on or off.

Mud Turret:

The most expensive trap in the shop, it can be placed on any wall facing left or right and can knock out any villager in with just a click. The player must click on the turret to shoot it.

Walls:

Walls are cheap to purchase and can be placed by the player in order to construct their own maze or maze segments if they want. Players will need to purchase walls as the farmer character will break down the walls of the maze.

Sheep:

The sheep is the most expensive item in the shop as they are basically like an extra life to a player they will spawn randomly in the maze when purchased

Undo:

The undo button undoes anything the player has placed but will only undo anything placed in that round. So if on wave one you place 5 walls after wave one is finished you will not be allowed to undo the 5 walls you placed you will have to use the destroy tool.

Redo:

Redo works after something is undone and works the exact same as the undo just with the opposite effect of replacing what you've undo also can only be used on things undone in that round.

Destroy Tool:

The destroy tool can be used to destroy anything within the maze but will only give you half your money back from what it cost to purchase the items. So, make sure to use the undo and redo buttons to save money.

Game AI:

The simpleton:

The simpleton explores the maze much like a drunken college student trying to find his way home after a big night out. The simpleton wanders around randomly just hoping to be lucky enough to find a sheep to bring home to make his parents proud.

The crazy scientist:

The crazy scientist explores the maze meticulously using only left and rights as any good maze explorer knows that if you follow one side you will always find the exit or in this case the sheep.

The adventurer:

The adventure explores the maze and uses his years of experience adventuring to know to never retrace his steps more than once knowing exactly where he has been in the maze and will only retrace his steps if all options have been exhausted.

The Farmer:

The farmer explores the maze much like the simpleton, but he has the strength of ten men and will rip down any hedge he comes across to find another sheep for his collection.

The Sheep:

The sheep are helpless defenseless creatures that you have been tasked with protecting from the angry villagers. They move randomly through the maze grazing on the nice green grass.

Progression:

Waves:

The villagers attack in waves starting with 3 villagers and each wave adds another 3 villagers so wave one has 3 and wave 10 has 30.
Each villager you kill earns you money which in turn allows you to purchase more defenses which in turn allows you to kill more villagers and so on.

UI:

The player starts at the main menu and can choose to instantly exit the game or hit play which brings you to the games shop screen titled the farmers market you can press esc whilst in the farmers market to go back to the main menu which will also allow you to restart the game now as well.

If the player presses the play button the game begins, the player can pause the game using the pause button on the right or stop the game using the red square and go back to the shop screen. if the player presses esc it brings the player back to the shop screen as well.
If the player is defeated, they will be given an option to restart the game or exit the game.

Technical

Maze Generation

The game generates a starting maze for the player using Recursive Backtracking and Depth-First Search like algorithms. The maze is made up of a number of cells, which can be empty, or contain a tile (like a wall, or trap).

The method for generating the maze starts at the entrance of the maze, and picks a neighbouring cell to move to, then picks another neighbouring cell that it has not yet visited, and moves there, and so on and so forth.
Each time the method moves to a new cell, it removes the wall between the two cells, thus creating a pathway.
Once the maze has exhausted all neighbouring cells, it tracks back the way it came until it finds a neighbouring cell it has not yet visited, moves to there, and continues the process.

Once no unvisited cells remain, the process is finished, and you are left with a maze, which can always be solved, as every part of the maze is accessible from any one point.

Animations

The animations for the game AI uses a linear interpolation formula to generate the character's pixel position based on the cell the AI resides in, and the cell he is moving towards. The position between the cells is based on the movement speed of the character and the current time elapsed since he began moving to the new tile.

This method of animation makes the characters very flexible in terms of variable movement speeds and position, meaning the animation will always be synced to the character's movement.

AI and Traps

The AI doesn't necessarily use pathfinding algorithms, instead they make random movements, according to rules, to give them personality.

While the Simpleton and the Farmer take each turn at a completely random chance, the Adventurer and the Crazy Scientist use their own methods to solve the maze.

The Adventurer

The Adventurer uses a similar method to the maze generation, in that every movement he takes is remembered, and once he retraces his steps, he places mental wall tile in the place he just was, so that he never retraces his steps twice, and continues this until he find a sheep, and then his way out of the maze.

The Crazy Scientist

The Crazy Scientist picks either left or right at the start of the game, and takes every available turn of that type from then on. A simple yet effective method of solving the maze.

The above AI were all made generic enough that they could inherit from an abstract Maze Solver class, and mostly call functions from that to achieve their functionality, a method of AI building very similar to that used in the RoboCode tanks.

The Sheep

The final AI, the Sheep, uses a similar technique to a finite state machine, in that it switches from Idle, to Wandering, and to following, based on its circumstances.

Idle

The sheep begins in the Idle state, simply munching on the grass. After a small amount of time passes the sheep will have a random chance of switching to its Wander state, if it does switch, it attempts to find a new direction

Wander

When looking for a new direction, the sheep will try 4 times, looking in a random direction each time, if each direction is blocked off by a wall, the sheep will give up and return to its idle state, if it finds an empty tile, it will move there.
During the sheep's Wander state, upon reaching a new tile, it has a random chance of switching back to Idle, before it ever even looks for a new direction.

Following

If the sheep collides with a villager AI at any point during the game, and the AI does not currently have a sheep following it, the sheep will begin to follow the AI. When following, the sheep moves at the same speed as the AI it follows, and simply moves to the last tile the AI stood in.