3/02/2018 BLOG – Stealth Game Prototype and Blog Introduction

Intro –

-Who I am?

Hello there. My name is Ian Hudson and I am of the time of writing this a final year University student at the University of Gloucestershire in the UK. This is the first of hopefully many updates which I will create and post. These updates hope to show to you the read (and me) the progress that I have made in my last semester at university.

-When will posts appear?

A new blog will be released either every week or every two weeks. This should be a good time frame to show off what I have been doing.

-What shall be talking about

Each new update will show work which I have under taken since the last update (hopefully). This will include mainly university assignments. However, a personal project might just popup here and there.

That’s the introduction done and dusted. On to what I have been doing for the past week.

My focus for the past week has been my Advance group project. The objective of this project is to work within a team for 4-5 months to create a game.

Stealth Game –

-What is it?

It is a Stealth game. Which incorporates hacking into it to allow the player more choice on paths to take. This will hopefully make a dynamic level which when played will be traversed in different ways.

-explain the project

-My role

I will be one of the two programmers on the project. My main tasks will be to program obviously. The types of programming will probably be gameplay related. However, I can see myself helping with all aspects like AI and graphics (shaders).

On to actual work I have done. I worked on a hacking mini game which could be used to gain access to new abilities, unlock new paths, etc. The mini game was based on a simple pipes puzzle game [image].

How it works?

# Welcome

Hello there. My name is Ian Hudson and I am of the time of writing this a final year University student at the University of Gloucestershire in the UK. This is the first of hopefully many updates which I will create and post. These updates hope to show to you the read (and me) the progress that I have made in my last semester at university.

A new blog will be released either every week or every two weeks. This should be a good time frame to show off what I have been doing.

Each new update will show work which I have under taken since the last update (hopefully). This will include mainly university assignments. However, a personal project might just popup here and there.

That’s the introduction done and dusted. On to what I have been doing for the past week.

<!--more-->

## Overview

My focus for the past week has been my Advance group project. The objective of this project is to work within a team for 4-5 months to create a game.

It is a Stealth game. Which incorporates hacking into it to allow the player more choice on paths to take. This will hopefully make a dynamic level which when played will be traversed in different ways.

I will be one of the two programmers on the project. My main tasks will be to program obviously. The types of programming will probably be gameplay related. However, I can see myself helping with all aspects like AI and graphics (shaders).

On to actual work I have done. I worked on a hacking mini game which could be used to gain access to new abilities, unlock new paths, etc. The mini game was based on a simple pipes puzzle game

![alt text](assets/img/updates/FirstUpdate/ "Ref puzzle game")

The mini game is based on a 6x4 grid with each grid slot being a tile. ![alt text](assets/img/updates/FirstUpdate/hacking\_tile.png "Tile") When the mini game is shown all the tiles are rotated a random number of times. This is done because when the puzzle is made the path is shown. ![alt text](assets/img/updates/FirstUpdate/hacking\_pathCompleted.png "Puzzle Completed")

By randomly rotating all the tiles the path is then hidden and must be found ![alt text](assets/img/updates/FirstUpdate/hacking\_pathRandom.png "Puzzle Random")

After the player has rotated the tiles to make a path from start to end, they press the hack button and if they have correctly done the puzzle then the StartPuzzle function which is called when the hack button is pressed will return true or false depending on if the path is right or not.

Now that the mini game has been explained and how it works time for some actual code.

### Code

The first code snippet is from the Tile class. The code below is a function which is used when a tile has been clicked.

<pre class="brush: csharp">

/// <summary>

/// Rotate the tile. This will rotate 90 deg clockwise

/// The openDirections are changed to acomadate this

/// </summary>

public void RotateTile()

{

//setup a new bool array for the new directions

bool[] newOpenDirections = new bool[4]

{false, false, false, false };

//Loop though RIGHT, BOTTOM and LEFT directions

//If TOP direction is true then set RIGHT to true etc..

for (int i = 1; i < openDirections.Length; i++)

{

if(openDirections[i - 1] == true)

{

newOpenDirections[i] = true;

}

}

//check if LEFT is true then set TOP to true

if (openDirections[3] == true)

{

newOpenDirections[0] = true;

}

//Set openDirections to newOpenDirections

openDirections = newOpenDirections;

//Rotate the gameObejct's rotation by 90 deg

gameObject.transform.rotation \*= Quaternion.Euler(0, 0, -90);

//Assign the editor script values to openDirections

gameObject.GetComponent&lt;TileEditor>().SetValues(openDirections);

}

</pre>

Here is some code for checking for neighbours of a tile.

<pre class="brush: csharp">

/// <summary>

/// Return a list of tiles which are neigbours and valid

/// </summary>

/// <param name="a\_tile"></param>

/// <returns></returns>

private List&lt;Tile> GetNeighbours(Tile a\_tile)

{

//Setup a new List of tiles for all valid neighbours

List&lt;Tile> returnNeighbours = new List&lt;Tile>();

//Get the x and y values of a\_tile in the array

int tileX = (int)a\_tile.X;

int tileY = (int)a\_tile.Y;

//Check if there is a tile above us

if (tileX - 1 >= 0)

{

//check TOP. Check if a\_tile's TOP is open and the tile above us BOTTOM is open

//If both are true then add the tile above us as it is valid

if (a\_tile.OpenDirections[0] == true && tiles[tileX - 1, tileY].OpenDirections[2] == true)

{

returnNeighbours.Add(tiles[tileX - 1, tileY]);

}

}

//Check if there is a tile below us

if (tileX + 1 < 4)

{

//check BOTTOM. Check if a\_tile's BOTTOM is open and the tile below us TOP is open

//If both are true then add the tile above us as it is valid

if (a\_tile.OpenDirections[2] == true && tiles[tileX + 1, tileY].OpenDirections[0] == true)

{

returnNeighbours.Add(tiles[tileX + 1, tileY]);

}

}

//Check if there is a tile to our left

if (tileY - 1 >= 0)

{

//check LEFT. Check if a\_tile's LEFT is open and the tile to our left is open on the RIGHT

//If both are true then add the tile above us as it is valid

if (a\_tile.OpenDirections[3] == true && tiles[tileX, tileY - 1].OpenDirections[1] == true)

{

returnNeighbours.Add(tiles[tileX, tileY - 1]);

}

}

//Check if there is a tile to our right

if (tileY + 1 < 6)

{

//check RIGHT. Check if a\_tile's RIGHT is open and the tile to our RIGHT is open on the LEFT

//If both are true then add the tile above us as it is valid

if (a\_tile.OpenDirections[1] == true && tiles[tileX, tileY + 1].OpenDirections[3] == true)

{

returnNeighbours.Add(tiles[tileX, tileY + 1]);

}

}

//return all neighbours

return returnNeighbours;

}

</pre>

Test }

}

//Check if there is a tile to our right

if (tileY + 1 < 6)

{

//check RIGHT. Check if a\_tile's RIGHT is open and the tile to our RIGHT is open on the LEFT

//If both are true then add the tile above us as it is valid

if (a\_tile.OpenDirections[1] == true && tiles[tileX, tileY + 1].OpenDirections[3] == true)

{

returnNeighbours.Add(tiles[tileX, tileY + 1]);

}

}

//return all neighbours

return returnNeighbours;

}

</pre>

That’s it for my first update. YAY. As stated early this will hopefully be a weekly/fortnight thing. I hope you have enjoyed this update.

If there is/was anything you spotted which is wrong or have any questions about, please don’t hesitate to contact me. I would be grateful.

Tile code:

-Work which I have undertaken this week

-Hacking mini game

-What is the game?

-How it works?

-A\* to find a path?

-Show code?

-Easy implementation?