Santa’s Deliveroo Overview Document

Classes

RTSController:

This class is responsible to manage player user input for gameplay mechanics like select a unit and it also manages the current camera view to transition between fly camera and rts camera mode. It also keeps track of available units for the player to control.

UnitRTS:

This class represent each single player unit, in this case each Santa. It is responsible to manage all the possible interaction for the unit, like moving to a specified position, collecting a gift and set waypoints in the current moving path. This unit implement the IOutlineable interface which is implemented by objects that can be selected (only in the tactical view), also when the mouse is hovering on a IOutlineable object it changes its cursor texture to give the player a visual feedback.

GameManager:

The GameManager keeps track of the game state and manages scene loading and unloading, in this case only between the Boot scene and the Main scene. It instantiates the system prefabs like the UIManager and is a holder for gameplay requirements like the minimum number of gifts the player has to collect to win.

UIManager:

The UIManager manages all the UI. It also manages a pause menu from which is possible to pause and resume the game, restart the game and quit from the application. It also includes a Timer object which keeps track of the remaining time.

AudioManager:

The AudioManager is responsible to manage the audio source for sfx and the audio source for background music.

FlyCam:

The FlyCam is attached to the fly camera object and it is enabled at the start of the level. It allows the player to move the camera with no restriction with WASD keys for movement and Q/E for altitude. It is also possible with left SHIFT to increase the speed and left CTRL to slow down the speed movement. Only in the Unity Editor is possible pressing the END key to change the cursor lock state.

RTSCameraController:

The RTSCameraController allows the player to have a tactical view of the scene. It is possible to move the camera with WASD keys and with the use of the mouse scroll wheel is possible to zoom in/out. In the Unity Editor it is also possible to activate the option to move the camera when the mouse cursor is in the proximity of an edge.

ObjectPooler:

The ObjectPooler class is responsible to spawn a set of pool avoiding instantiations and destructions of objects. This is used in particular when the user sets a number of waypoint for a selectable unit to display line paths with Unity LineRenderer components and waypoint markers.

House:

The House class holds the associated gifts information (which gifts is expecting to receive) and updates it when one of each is delivered. When all associated gifts are delivered to it become unselectable.

Item:

Each item represents a gift, which a selectable object (it implements the IOutlineable interface) and like houses it holds the associated houses information. When it enters in the corresponding house trigger collider it is automatically delivered to that house.

Enemy:

Each enemy is a constant moving entity. It has a set of waypoints which are randomly selected to cover a specific are of the city district. Its range of detection is calculated by distance with the Vector3.Distance function. It has two states, Patrol and Chase. It switches between patrol and chase state when a rts unit is in its range of detection.

Known issues

- The game is missing a level selection menu and it contains only one level. (Unfortunately, I hadn’t much time to make it and I choose to give more importance to gameplay mechanics and to first have a fully functional game).

- The background music doesn’t loop at the end. I forgot to set the loop check in the audio source.