2015

Specific Technical Features

SPOKEN

DADIU

Cinematics

The cinematic system consists of a scene with a canvas that contains a series of panels called frames. Each frame contains a Frame Controller script, an Animator and an Audio Source. The script contains a whole slew of public variables that allows a non-coder to control the behavior of each frame. Most of the variables also contain tool tips, allowing the user to get information on the function of each option. The script is linked to the next frame in the sequence by a public reference and will start the script in the next frame when it is done. The user can control the time a frame stays active directly but if an animation is attached to the frame, it will use the length of that animation instead. If there is a clip assigned to the Audio Source it will play the whole clip as no further functionality was necessary. With this system a user only needs to understand the animation and animator systems of unity, which are already quite familiar to anyone who has experience with Flash or After Effects.

It would be possible to remove the step of linking all the frames together with a tag, finding them and then sorting the list but this is slow and not really needed when the cinematics are using less than 20 pictures.

Level Specific Scripting

The waypoint system was implemented to address the problem with getting the AI to perform scripted behavior for tutorial levels and levels that just require specific actions. When a level is started the AI will first check to see if the current level uses waypoints. If it does, it will find all waypoints with a certain tag and sort them alphabetically. The AI will start at the first in the list and then go the next etc. This means that the user has to be observant of the naming of the waypoint but other than that there is no other knowledge or work required. Each waypoint has a Waypoint script that contains a large assortment of public variables that control the AI's behavior when it reaches the position of the waypoint. When the AI moves towards a waypoint it activates a co-routine that first checks if the AI has reached it, and when that is true, it will execute the rest of the script. There are many different options for the user: playing a specific animation, creating a new sign, interacting with an object etc. After the AI has performed any actions detailed by the script it will wait until it should move to the next point. The behavior of the wait is determined by the script as well: time, player distance, player interaction, camera movement etc. Other functions, not directly related to the AI, includes immobilizing the player, advancing text on the screen and making certain objects blink. Non-coders can set up the AI behavior by themselves and the system is very customizable and can easily be expanded upon if a new functionality is required.