

EASY DIALOGUE 2D

By Moises Gabriel

Description:

Easy Dialogue 2D is a very simple dialogue system with a quick installation and fast set up.

It's very useful for Game Jams, prototypes, and small games.

Out of the box, the system allows for NPCs to hold multiple nodes of dialogue, and within those nodes can be as many lines of dialogue as needed. It has a built-in Typewriter effect so that the text can be displayed one character at a time. You can customize the Typewriter to fit your needs.

Installation:

- Through the Unity Asset Store, add this package to your Packages. Once done, proceed to the Package Manager in the Unity Editor, download, and import the package.
- Through Itch.io, download the asset, and import the EASYDS package to your Assets folder.
-

What's Included:

The EasyDS 2D folder has (5) folders.

- The Audio folder holds the audio you might need for when you display your text.
- The Editor folder holds the script that handles the *Create Dialogue* button.
- The Nodes folder will hold all of the Scriptable Objects containing your dialogue.
- The Scriptable Objects folder holds the script that creates the Scriptable Objects.
- The Scripts folder holds the main scripts to be used with this system.

Setup:

1. Create an empty Game Object and call it "Dialogue Manager". Attach the "DialogueManager" script to it.
2. Attach the "DetectPlayer" script to your NPC(s).
3. Attach the EasyDS script to your NPC(s).
 - a. Add your NPC's name
 - b. Click the *Create Dialogue* Button.

Writing Dialogue:

After clicking the *Create Dialogue* button, the new Dialogue Scriptable Object (Node) will automatically be added to the NPC's dialogue list and if navigate to the Nodes folder you will find that the Node has your NPC's name as its name. All that's left is to click the (+) button while in the Node object and add dialogue for your NPC!

Each string acts as a line of dialogue. If a line will have another line after it, end the line with the **/line** tag. If it will be the final line in the node, end the line with the **/end** tag.

Example:

- You are really cool! /line
- You've inspired me to be cool too. /line
- I hope you have a great day! /end

Once the system reaches the end of a node, it will check for the next node in the List. If there aren't any more nodes, it will restart from the first node.

Typewriter Effect:

In the Dialogue Manager component that was added to the Dialogue Manager object, you'll see different variables under the Typewriter heading.

- Typewriter Speed – the speed at which each letter is revealed.
- Auto Next Line – when this is active, the lines in the node are displayed one at a time automatically without any user input.
- Wait For Next Line – this is the time between each line when Auto Next Line is active.
- Pause On Punctuation – when this is active, the Typewriter will have a brief pause when sentences have punctuation.
- Pause Length – how long the Typewriter should pause when there's punctuation.
- Play Text Audio – when this is active, an audio clip is played when each letter is revealed.
- Audio Clip – the audio clip that is played when Play Text Audio is active.
- Audio Source – the source of the audio clip.

Tips:

- For the Detect Player component, the Alert Object and its corresponding variables/methods are optional and can be commented out in the code.
- The Create Dialogue button will automatically place the objects in the Nodes folder with an automatic naming convention.

This asset utilizes the **Creative Commons Attribution-NoDerivatives 4.0 International**

Public License. For more information on the license please check on the **LICENSE** pdf that is included with this package.