

DOCUMENTATION

Objective Manager

Index

Index..... 2

Update Log 3

Introduction 4

Getting Started 5

 Installation..... 5

 First Run 5

 Creating a new Objective 6

 Objective 6

On the Run 8

How To..... 9

 Create an Objective Type Reach 9

 Creating an Objective Type Action..... 11

 Creating an Objective Type Kill 12

Road Map 13

Credits 14

Update Log

Version: 1.00 (17/01/2021) – Initial Documentation

Introduction

This addon was created with the while I was working on a University project, when I started, I had to do every single bit of code and it was annoying how much effort I had to put, just to make a simple objective. So, I have decided to create a simple way to create and maintain objectives.

Further updates will improve the workflow of this addon as new features will be added.

You can check the Road Map of this project down on the **Road Map** Section!

Getting Started

Getting started with Objective Manager!

Installation

Download the package, open *Unity*, then go to **Assets > Import Package > Custom Package**.

After following these steps, you should be ready to go.

First Run

After installing the package, you should now have a new Menu Item in top bar, called ***Objective System***, as shown below.

File Edit Assets GameObject Component **Objective System** Mobile Input Window Help

Figure 1 - Toolbar navigation.

When you press that button a dropdown will show with only one option, ***Objective Manager***. After you press this option a new window will appear as shown below.

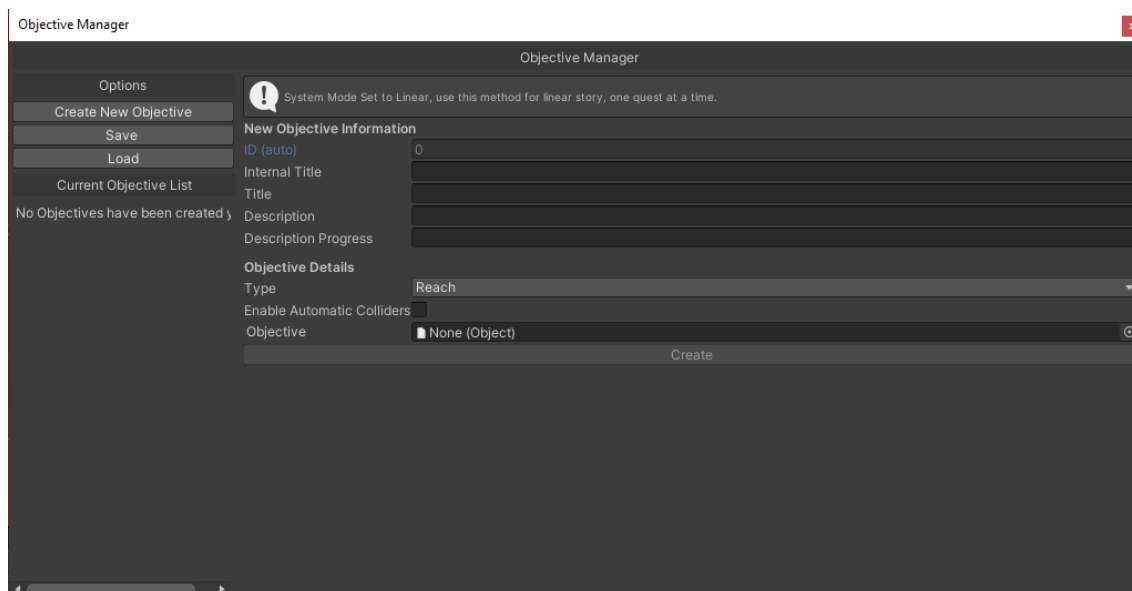


Figure 2 - Objective Manager Window.

Creating a new Objective

When you open the *Objective Manager Window* you will have now access to 3 buttons, *Create New Objective*, *Save* and *Load*.

Note: *Save and Load buttons aren't working at the moment, check Road Map Section to more information.*

Although when you open the manager, the default configuration is to start creating a new Objective, we highly recommend that you press the **Create New Objective**.

Objective

1. New Objective Information:

- a. **ID(Auto):** This feature is to create a unique ID so you can easily track your objectives.
- b. **Internal Title:** Set the internal name of your objective, useful for tracking your objectives.
- c. **Title:** The Objective Title.
- d. **Description:** Write the description of your objective.
- e. **Description Progress:** This feature is used while you need to give more information or shorter information about your Objective.

In the following section you can choose between three types of objectives listed below, since some features may differ from what type of objective you choose, I'll put it in an easy way to understand each type:

2. Type:

- a. **Reach:** Used when you want your objective to be reachable. (example: Go from A to B)
- b. **Action:** Used when you want your objective to be and Action. (example: Press a button to enable/disable something)
- c. **Kill:** Used when you want your objective to be a Kill Type. (example: Kill X enemies)

3. Objective Details (Reach Type):

- a. **Enable Automatic Colliders:** This feature creates a collider, usual (+1, +1, +1) more than your Mesh. Not very friendly when you have detailed objects that you need to have a specific collider there.
- b. **Objective:** Select which GameObject is the objective.

4. Objective Details (Action Type):

- a. **Enable Automatic Colliders:** This feature creates a collider, usual (+1, +1, +1) more than your Mesh. Not very friendly when you have detailed objects that you need to have a specific collider there.
- b. **Action Key:** Set which key needs to be pressed to set the action.
- c. **Time to Interact:** Set the amount of time you will need to successfully complete an action. (default time is set to 3 seconds)
- d. **Targets:** Choose how many targets you need to interact to complete the objective.
- e. **Element X:** Select which GameObject is the objective.

5. Objective Details (Kill Type):

- a. **Amount:** Set the amount of kills you need to complete this type of objective.
- b. **Target Tag:** Set the **TAG** used by the objective.
- c. **Search Tag:** This button will output None if there is not any tag within the criteria that you wrote, or it will output the tag meaning that there is a valid tag.

On the Run

You probably noticed when you opened the *Objective Manager Window* that some *GameObjects* were created. When you run this addon the first thing it will do, is create a workflow to track and register everything you do.

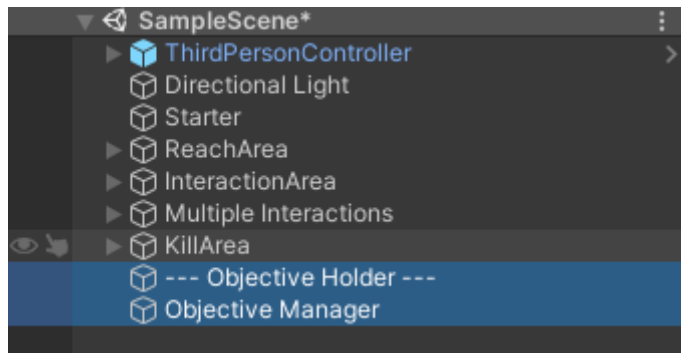


Figure 3 - Automatic Game Objected Created.

Objective Holder is created with the purpose of store all the Objective that you create.

Objective Manager is created with the purpose to manage the flow of the objectives.

If any of this *GameObjects* is deleted, whenever you open the *Objective Manager Window* they will be created again.

Note: If you delete Objective Holder and you have objectives there, they will disappear. We will work on a “Fail-Safe”, further information on **Road Map**.

How To...

We recommend that you access our YouTube channel and check how we do it!

You can always use our example in the example folder 😊.

Create an Objective Type Reach

You design your map, and now you want your player to go from anywhere in the Scene to the X.

So, create an Empty GameObject (I will call it "Reach_Here"), and navigate to where you want your player to go, and put there the GameObject.

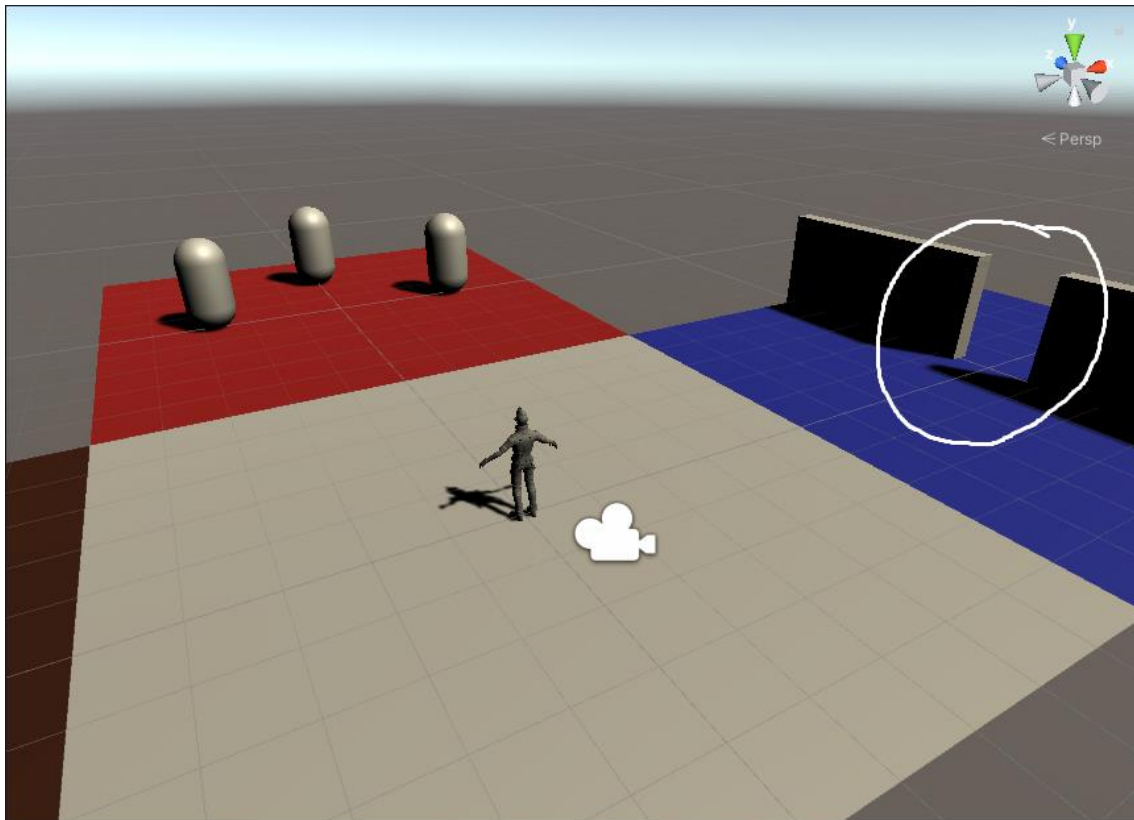


Figure 4 - Example Create Objective Reach, I want my player to go there (white circle).

See next page

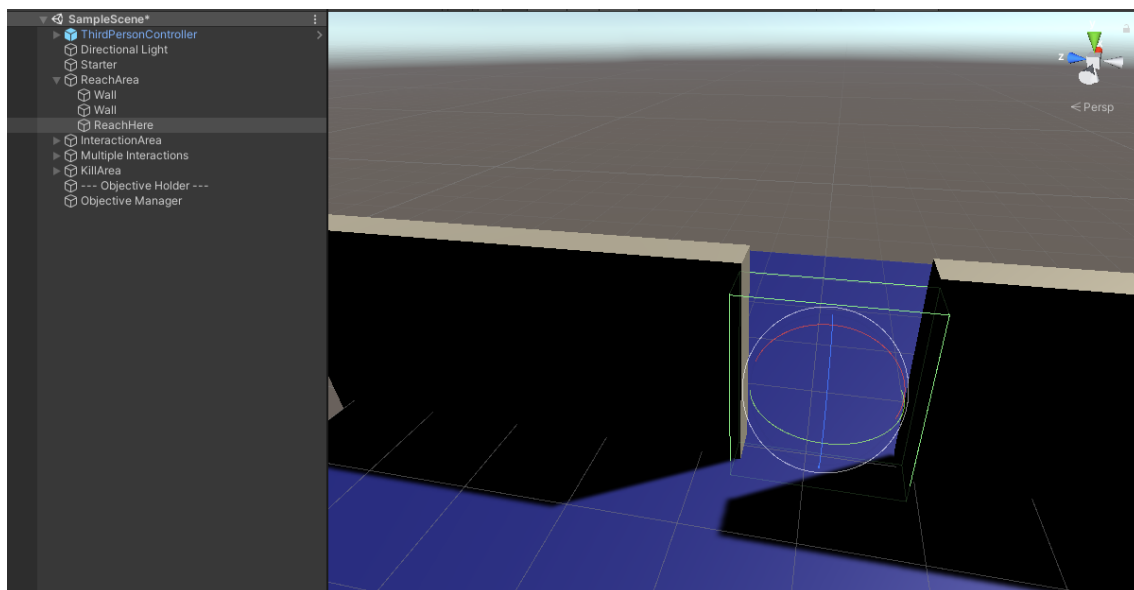


Figure 5 - Create an empty GameObject and position where you want it to go.

Note: As said earlier we highly recommend that you create your own Collider as the automatic Collider will create a Collider with the size of a default Cube (1,1,1).

Open *Objective Manager Window* and start filling the fields.

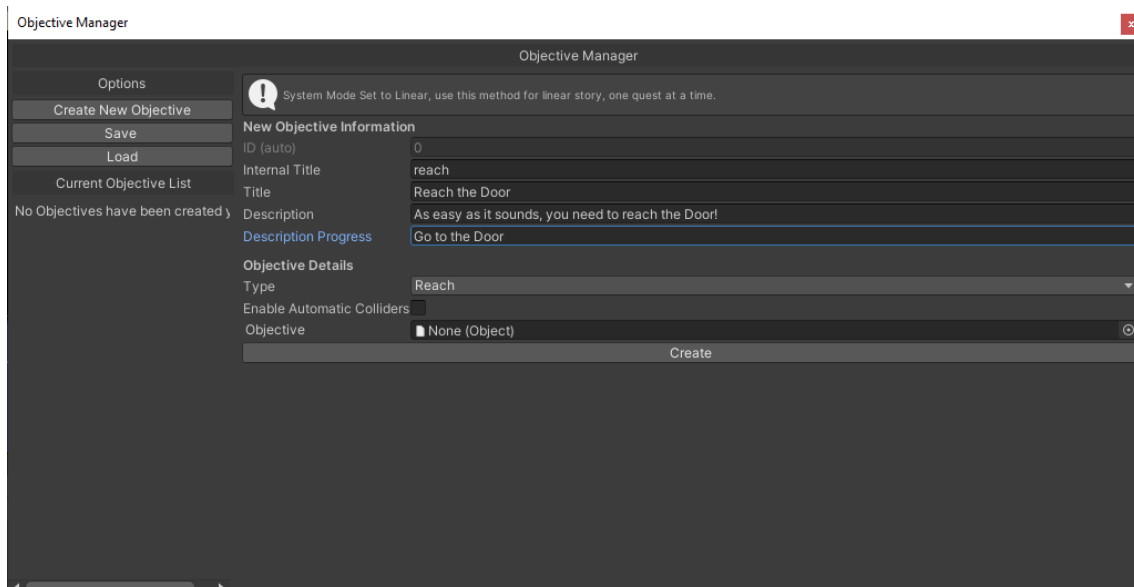
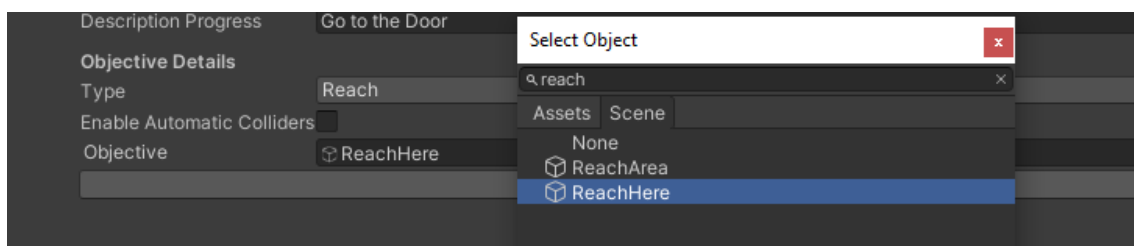
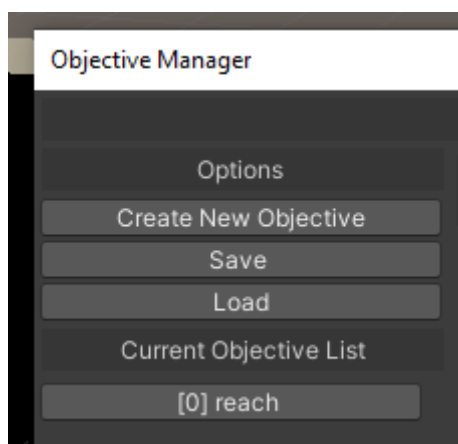


Figure 6 - Example Objective Reach.

After filling the information spaces, select the Object you wish to select as Objective.



You are all set now, just press the *Create* button and a *GameObject* will appear in the *Objective Holder* and in the *Objective Manager Window*.



Creating an Objective Type Action

Most of the stuff mentioned in the previous section works in this section as well.

When you choose the Type to Action, there will be shown different stuff in the *Objective Details*. Set the Action key needed to perform an action and then set the time needed to perform that action.

In the **Targets** set the size to how much actions you need to complete this objective, and then select which *GameObject* is the objective.

Creating an Objective Type Kill

Most of the stuff mentioned in the previous section works in this section as well.

When you choose the Kill Type, you will have access to two options, *Amount* and *Target Tag*.

You will need to specify how many kills you need to complete the Objective.

You will need to specify the Enemy Tag.

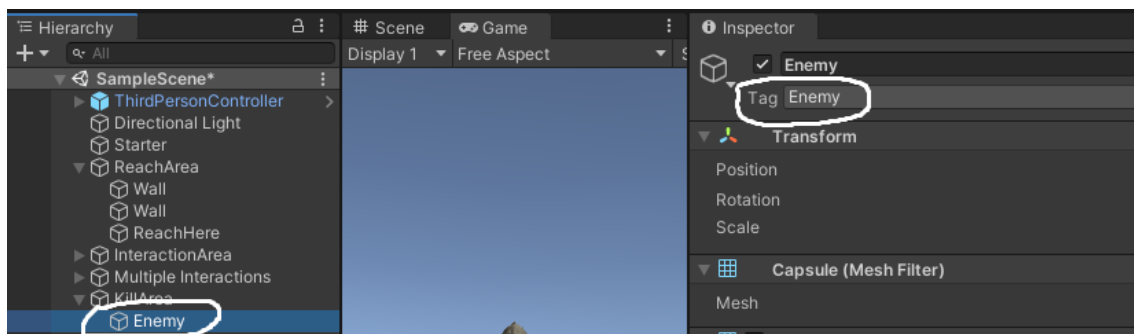


Figure 7 - Setup GameObject of the Enemy.

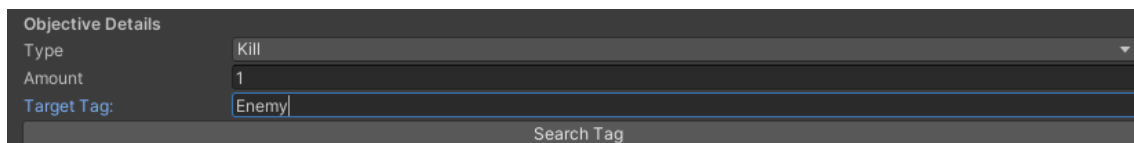


Figure 8 - Specify the "Enemy" Tag.

If everything exists, you will get an information on your console saying "Output: *Value*" if the value in the *Value* is the name of the tag it means that the tag exists, and you can create your objective. If the value is "None" it means that you need to create and assign that tag to the GameObject.

Road Map

In this section we will talk about our future updates and what we intend to do.

1. **Better documentation:** As you can see, there isn't much effort spent in the documentation, hopefully we want to have a full and clean documentation.
2. **Video Tutorials:** We want to an easy step-by-step video tutorial on how you can use this addon.
3. **Multiple Scene Manager:** We are already working on this feature. When ready we will update, and it will be easier to work with this addon.
4. **UI Implementation:** Although some of the features already implemented are design to work with UI, there isn't an easy way to access it. As soon we launch the *Multiple Scene Manager*, we will release this implementation as well.
5. **Fail-Safe Feature:** Well we want to create a XML file where it will store all of your objectives, if by accident you delete the *Objective Holder*, you can always revert to the last Save 😊.

Credits

To Be filled.