Contents

EXAMINE SYSTEM V1.4.3
DOCUMENTATION

Introduction

Thank you for purchasing the Examine System. This is a Raycast based system where the aim is to interact with any number of 3D objects you have in your scene, then allow you to examine or view those up close to reveal secrets. This allows the player to look at interesting things. Just as you see in a lot of horror and adventure games. It is extremely easy to setup, add your 3D object to the scene, apply a script, add a few tags. Duplicate the object and parent to your camera for rendering, apply another script and you will be able to interact with them! The system uses the emission value of your materials to create an additional selection highlight.

V1.4 Now includes:
► Inspection points, text highlight, custom events and more!

►Raycast system that detects any number of 3D objects you have in your scene
►Zooming, initial rotations, UI customisation and more!
►Add as many objects as you desire to your game.
►Simple setup and fully tool-tipped and commented scripts.
►A basic test scene included for you to see functionality.
►System to allow you to view, rotate and specify a visual name for your object.
►Scripts which can be easily modified.
►Disable and audio managers for easy editing and use.
►Sound effects for the system are included.

CHECK OUT THE ONLINE DOCUMENTATION FOR SETUP HERE:
https://app.gitbook.com/@speedtutoruk/s/examine-system-v1-doc/

MATTHEW RUKAS – SPEEDTUTOR
WWW.SPEED-TUTOR.COM
SPEEDTUTORUK@GMAIL.COM

FAQ

Q). How do I Import the asset?

A). Go to the Unity asset store and visit your "**Download manager**". Download the asset if not already downloaded and click "**Import**", import all required features of the asset for your use. It should have appeared in your project under "**Examine System**".

Q). Should I import "Project Settings" when choosing to import this asset?

A). It is always recommend NOT to import project settings unless important for your project, you can alternatively create an entirely new project and import this asset with the project setting, then transfer the package folder over to your current project. To save any issues! Always back up your project before importing any assets – If you have any issues, do let me know!

Q). Is there an example of this asset working?

A). Yes, you can open the "**Examine System Demo**" to see the flashlight system in action, or use this scene as your initial base of your project.

Q). The blur isn't working or has errors in my version of Unity?

A). This can be removed if you have a different solution, another good solution would be to use a panel (With some transparency) instead of the blur. You would have to edit the script to make the panel appear instead of the blur, but you'll only need to change a couple of lines. (Might be more prevalent in HDRP)

Q). My object isn't rendering properly when attached to the Examine Camera and examining, I also have children objects within this object.

A). If your object is not rendering properly or has children, remember that you must set all the children (and parent) with the layer of "**ExamineLayer**" to make it render above the blur.

Q). I'm using children on my object I interact with, what should I do?

A). Only add the scripts and/or collider to the parent object but make sure this parent object has a material, if you want to use the "ShowHighlight" function – If not, the highlight will not work and functionality will be broken. Just untick "ShowHighlight" if that is the case!

Q). My managers don't stay between scenes properly?

A). Take make these work best, they shouldn't be in an empty game object, but loose within the hierarchy. This is just how the Don'tDestroyOnLoad works, I only put them with a managers parent gameobject to keep it looking neat and tidy!

Getting Started - Setup

Up to date setup on the ONLINE DOCUMENTATION HERE:
https://app.gitbook.com/@speedtutoruk/s/examine-system-v1-doc/

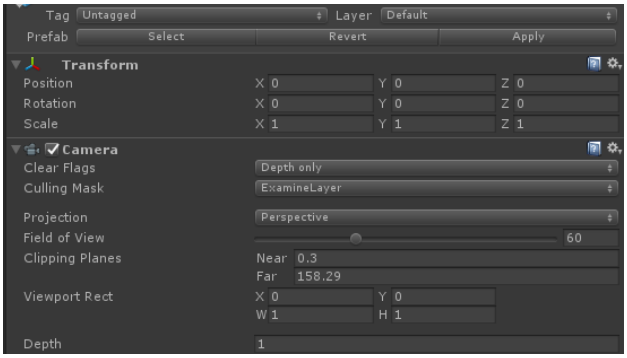
BEFORE YOU START: You could add the "EntireDemoScene" prefab to your scene if you want a really quick start!

Setup some basic tags and layers before we start!
Make sure to create a tag called: "**ExaminePoint**" and "**InspectPoint**".
Create a set of layers: "**ExamineLayer**" and "**InspectPointLayer**"

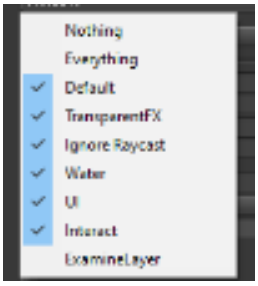
1. When starting your new project please import the "**Characters**" Standard assets or any FPSController you wish. **NOTE: I have provided this for you, but you may use your own if you wish!**

2. Add the "**FPSController**" from the "**Prefabs**" folder of my asset package to your scene. You can drag this into the hierarchy or the main scene.

3. The "**FPSController**" will have a main camera parented to it and another camera called the "**ExamineCamera**" – This camera is a "Depth ONLY" camera (Which means it will only render things we specify and will lower the overhead on rendering your scene in a different camera view. **NOTE: Make sure the ONLY culling mask on this camera is "ExamineLayer"! (Also make sure Depth is set to 1!**



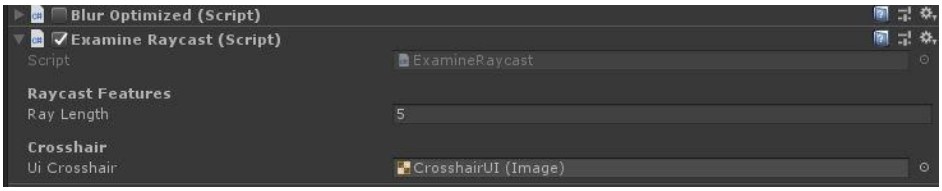
4. On your "**MainCamera**" make sure the "**Culling Mask**" has "**ExamineLayer**" unticked!



5. Please navigate to the scripts folder and add the "**ExamineRaycast**" script to your "**MainCamera**". (If not already from the FPSController_Prefab I have provided)
6. Navigate to the "**Scripts**" folder again and add the "**BlurOptimized**" script to your "**MainCamera**". (If not already on the prefab I provided). Untick the script so it is not active.

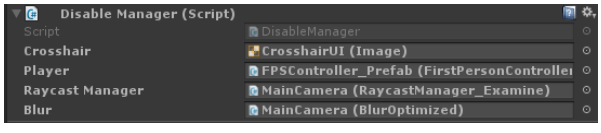
7. Now drag the "**ExamineManagersContainer**" and the "**ExamineCanvasContainer**" from the "**Prefabs**" folder into the hierarchy. **NOTE: Only drag this prefab into the hierarchy, not the scene.**

8. Click on the "**MainCamera**" again and find the "**ExamineRaycast**" crosshair slot in the inspector, and add the new "**CrosshairUI**" to that slot. (Navigate to "**Managers**" > "**Canvas Container**" > "**Canvas**" > "**CrosshairUI**" to find the crosshair).



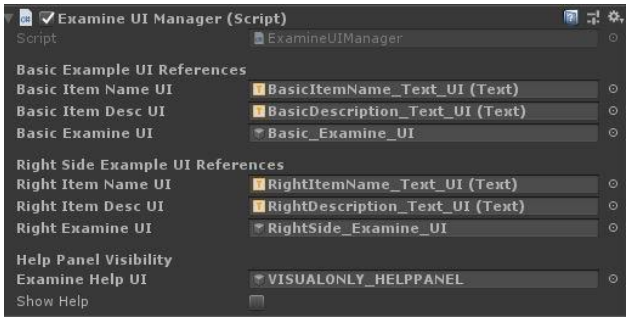
9. Make sure the **"MainCamera"** is tagged as **"MainCamera"**

10. Inside the **"ExamineManagersContainer"** container in the Hierarchy, find the **"Disable Manager"** gameobject. Click on it and you will need to add the specific references in the inspector.
    Add the "**CrosshairUI**" to the "**Crosshair**" slot like we did on the camera.
    Add your "**Player**" object (Usually "**FPSController**" to the "**Player**" slot).
    Add the "**ExamineRaycast**" which will be located on the "**MainCamera**" to the "**Raycast Manager**" slot.
    Add the "**MainCamera**" to that slot which will automatically find the "**BlurOptimized**" script for you. See screenshot.

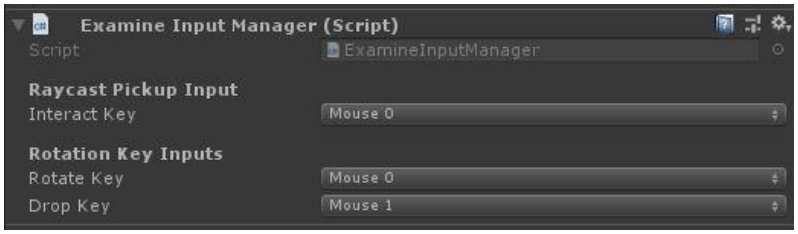


11. Inside the "**Managers_Examine**" again will have a GameObject called "**UIManager**" – Make sure this has the correct connections for enabling and disabling the UI. See screenshot below:

- Add the Basic Item UI's to the respective slots
- Add the Right side UI's to the respective slots



12. You can set the inputs using the “ExamineInputManager”.



13. Everything is ready for you to start adding 3D Objects to your scene. Add your first 3D object or add ones from the “**Prefabs**” folder. Add the “**InteractiveObjects**” prefab to your hierarchy or scene.
14. Make sure that any object you wish to interact with as a tag of “**Pickup**” and a layer of “**Interact**”. You will need to give the object a collider if not already.
    Add the “**ExamineItemController**” script from the “**Scripts**” folder. (If not already)
15. Select your “Main Camera” and make sure it has the tag “**MainCamera**”.
16. Select the “ExaminePoint” object and make sure it has the tag “**ExaminePoint**”

Initial Rotation Offset: Set your default rotation for your object to be displayed when examining by adjusting the X, Y, Z values.

Initial Zoom: The initial distance the object is away from the camera (Lower value is closer)
Zoom Range: The range where you can use the scroll wheel to zoom in and out
Zoom Sensitivity: The sensitivity of the zooming capability

Rotation Speeds: Set the speed of the X and Y rotation speed of for the object.

Show Emissive Highlight: Whether the system should use the emission slot in your material to create an object highlight or not (Requires the object this script is on, to have a mesh renderer – You could create an array if an object has children. Email me if you’re confused!

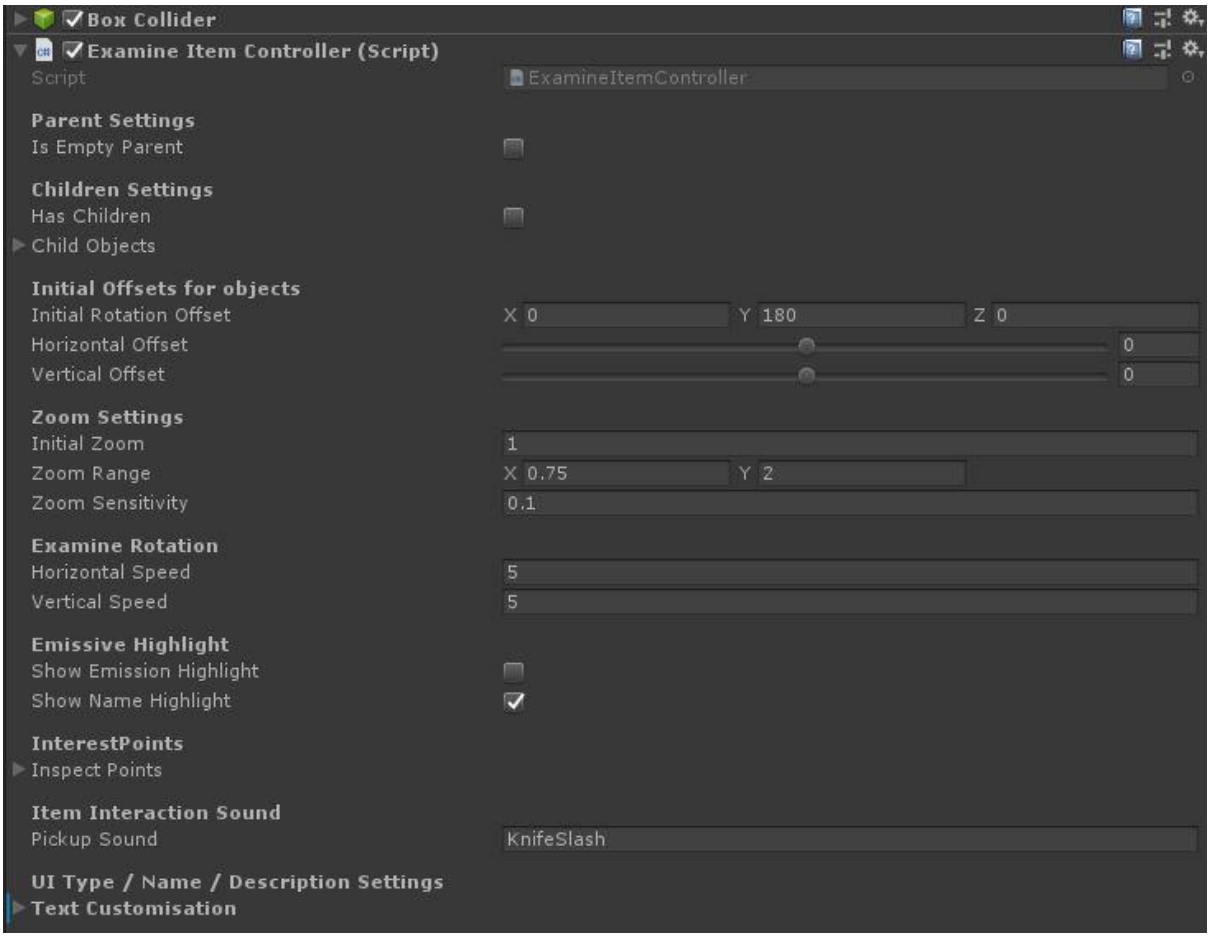
Show Name Highlight: This will show a simple UI next to the cursor to display object name and interaction tip!

UI Type: The type of display UI you would like: Bottom, right or none!

Item Name: Give your object any name you like and it will be add to the text component which sits in the canvas. Edit as you wish.

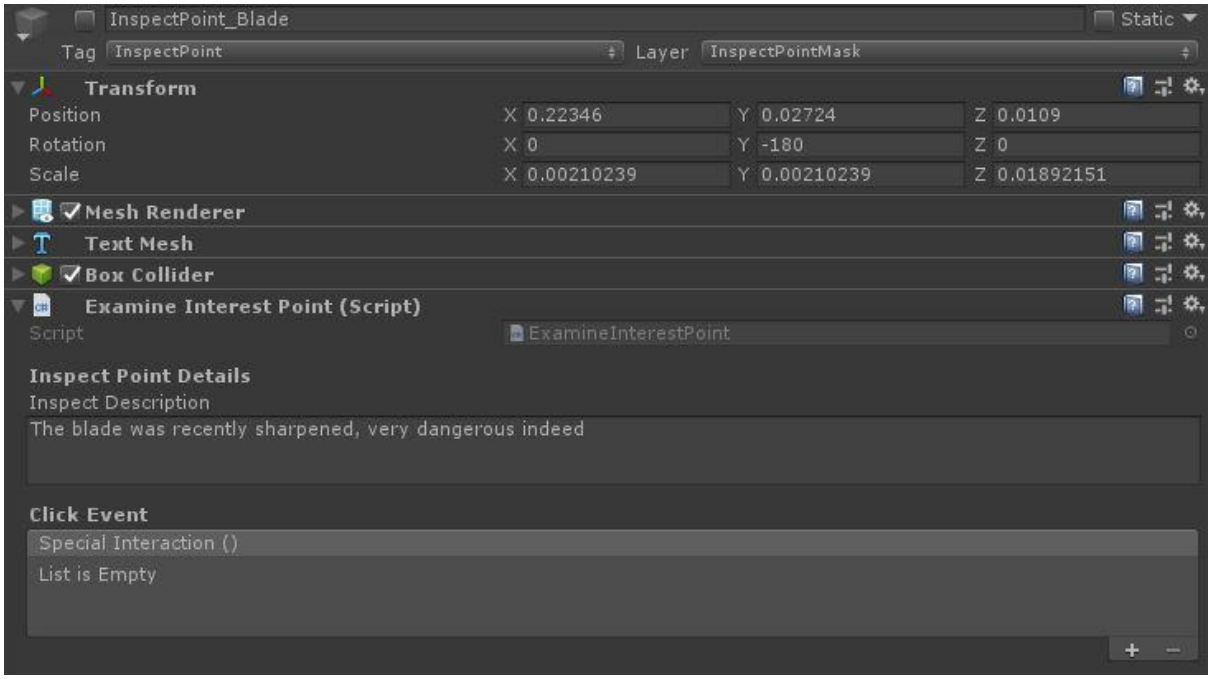
Item Description: Give the objects a short description for examine purposes.

General Font Settings for each type: Adjust these as you wish!



Adding Inspection Points

1). You can create an inspection point by either adding the “InspectionPointPrefab” to your scene which will have the setup as below:
Tag: “InspectPoint”
Layer: “InspectPointLayer”
Add: Box Collider
Add: “ExamineInspectPoint” script
On the paret Object, in this example the knife object select the “ExamineItemController” script and add the inspect points to the inspect point array at the bottom of the inspector! NOTE: This is very important!



Here you can add a brief description in text form to display when you hover over an inspection point and also have an OnClick event when you select a point, you can see my example on the “Cardboard Box” in the test scene!

Final Notes

1). The “ExaminePoint” on the “ExamineCamera” is the object used to transform the examinable objects to it. Making them viewable during the interaction. Move this further or closer to make objects do the same when you examine them!

2). You can disable the hints or tips which explain controls by navigating to the “**UIManager**” and unticking the “ShowHelp” box in the inspector!

3). The system uses the materials “**Emission**” value to make a highlight, you can set each materials emission values to “**0.5**” if you aren’t using the emission value on the object. The script turns emission on and off so no need to do anything special. You can enable this on a per object basis by clicking the “ShowHighlight” on each interactive object within your scene, if you wish!

4). You can change the interaction buttons by selecting the “UserInputManager” script and changing the desired key or mouse button!

5). I have added a method within the “ExamineItemController” which uses “OnDestroy” – This destroys the instanced object material, if you create one using the highlight. Meaning it will save any memory leakage in future!

Remember to take a look at the demo scene if you have any troubles, it might give you an idea on how to fix an issue!

If you find the package helpful, please leave a positive review and star rating as it would really help me out! ☺ If you have any problems, feel free to send an email to me!

MATTHEW RUKAS – SPEEDTUTOR
WWW.SPEED-TUTOR.COM
SPEEDTUTORUK@GMAIL.COM

Extending the Examination System

How do I use a different character controller with this system?

In the "**DisableManager**" script we reference the player controller so we can disable the "**FirstPersonController**" script so when we interact with the object we disable character movement and camera rotation. If you're using a different controller please replace that reference with the way to disable your controllers movement and camera rotation!

Looking at the disable manager below you can edit the reference to your player by creating a new variable. Then within the "DisablePlayer" method disable your camera and movement scripts. You can email me for more information BUT this is very character controller dependant.

```
    public class ExamineDisableManager : MonoBehaviour
    {
        public static ExamineDisableManager instance;

        [SerializeField] private Image crosshair = null;
        [SerializeField] private FirstPersonController player = null;
        [SerializeField] private ExamineRaycast raycastManager = null;
        [SerializeField] private BlurOptimized blur = null;

        void Awake()
        {
            if (instance != null) { Destroy(gameObject); }
            else { instance = this; DontDestroyOnLoad(gameObject); }
        }

        public void DisablePlayer(bool disable)
        {
            if (disable)
            {
                raycastManager.enabled = false;
                Cursor.lockState = CursorLockMode.None;
                Cursor.visible = true;
                blur.enabled = true;
                crosshair.enabled = false;
                player.enabled = false;
            }
            else
            {
                raycastManager.enabled = true;
                Cursor.lockState = CursorLockMode.Locked;
                Cursor.visible = false;
                blur.enabled = false;
                crosshair.enabled = true;
                player.enabled = true;
            }
        }
    }
}
```

Patch Notes

Version 1.4.2 – May 2021

- Fixed an issue where adding "Walls" or an exclude layer would cause issues with object interaction, now no tags or layers are needed on a per object basis. ONLY: Layers of "ExamineLayer" & "InspectPointLayer" are required
- Changed the name of "InspectPointMask" to "InspectPointLayer" for consistency
- Removed some redudant code from the "RaycastManager" and "ItemController" script to reflect these changes
- Added the ability to change the position of the object, on the horizontal axis so you can move it around the screen
- Added a vertical offet to the system to control how far the object should appear up or down, based on your settings
- Added a boolean to show the "Help UI" on a per object basis. Update the "HelpPanel" elements to change the look and feel of the help prompts

Version 1.4.0 - April 2021

- New Features:
    - New variable in the inspector so sound names can easily be added from the "AudioManager"
        - Various sounds effects added for examine interaction sounds
    - Custom text highlight near the cursor to tell you item name and interaction information
        - Custom interaction toggle for both emission highlight and item name highlight
        - Custom Canvas created to modify: Images, text and prompts!
        - Added objects to the "UI Manager" so these only need to be linked once
    - Addition of inspection points to give the system more depth
        - Only become visible when interacting
    - Custom Events - So more things can happen when interacting with an object
        - Custom event interaction when selecting items
        - Lid removal on box example script
        - Key pickup example script
- Minor Fixes:
    - Changed both "MainCamera" and "ExaminePoint" variables so they're found automatically "OnStart()" rather than adding them yourself
    - Refactored the script and re-ordered variables
    - New test scene with more explanations, examples and presentation

Version 1.3 – November 2020

- Added a close button within the UI, so the examine can be dropped using a button
- Set the distance the item is from the camera, based on a value in the inspector
- Set the starting rotation of an object with an X, Y, Z value in the inspector

  - Zoom in and out of the object using the scrollwheel
  - Set zooming bounds with between 2 values
  - Added Text: Size, Font, Style and Color variables in the inspector to easily customise the UI's on a per asset basis
  - Added the built in "OnDestroy" method to make sure objects destroy material instances upon leaving the scene (To avoid memory leaks)
  - Added the "ExcludeLayer" to the raycast to ensure users can negate raycasts through other objects
  - Connected the "UIManager" to the "ExamineItemScript" to avoid any extra setup
  - Refactored the "ExamineItemScript" for refined code and ease of use

Version 1.2 – July 2020

- Scripting Changes
  - Updated the Examine Controller to incorporate a one script setup
  - Added a new Audio Manager, so more sounds can be added and called within script
  - Added a new input manager, which allows easy control of all inputs from one place
  - Refactored the Raycast, ExamineController and more!
- Miscellaneous
  - Updated and edited sounds, textures and materials
  - Added new manager examples to the prefabs and demo scene
  - Update all prefabs and set them to 0, 0, 0 for future use

Version 1.1 – Late May 2020

- Scripting Changes
  - Created an input manager to control custom inputs for the system
  - Custom interaction keys for choosing your own
  - Refactored and renamed code to be more consistent and optimised throughout
  - Changed variable names to make the scripts more consistent
  - Added an "enum" for different types of UI displays when examining
- Miscellaneous
  - Added an additional UI example for displaying item text – Right hand side
  - Set canvas scaling to match screen resolution
  - Anchored all UI's for correct functionality

Version 1.0 – Early May 2020

- Initial Release
  - Code optimisations and improvements

Contact

Thanks very much for downloading! Please be sure to leave a **star rating** and **review** if you liked the package!

Support me on Patreon: https://www.patreon.com/SpeedTutor
Make a small donation: http://www.paypal.me/speedtutor

If you have any problems with the pack, or have some ideas for new models you'd be interested in, please feel free to contact me.

Email: speedtutoruk@gmail.com

Website: http://www.speed-tutor.com