PROMETHEUS INSTRUCTION MANUAL

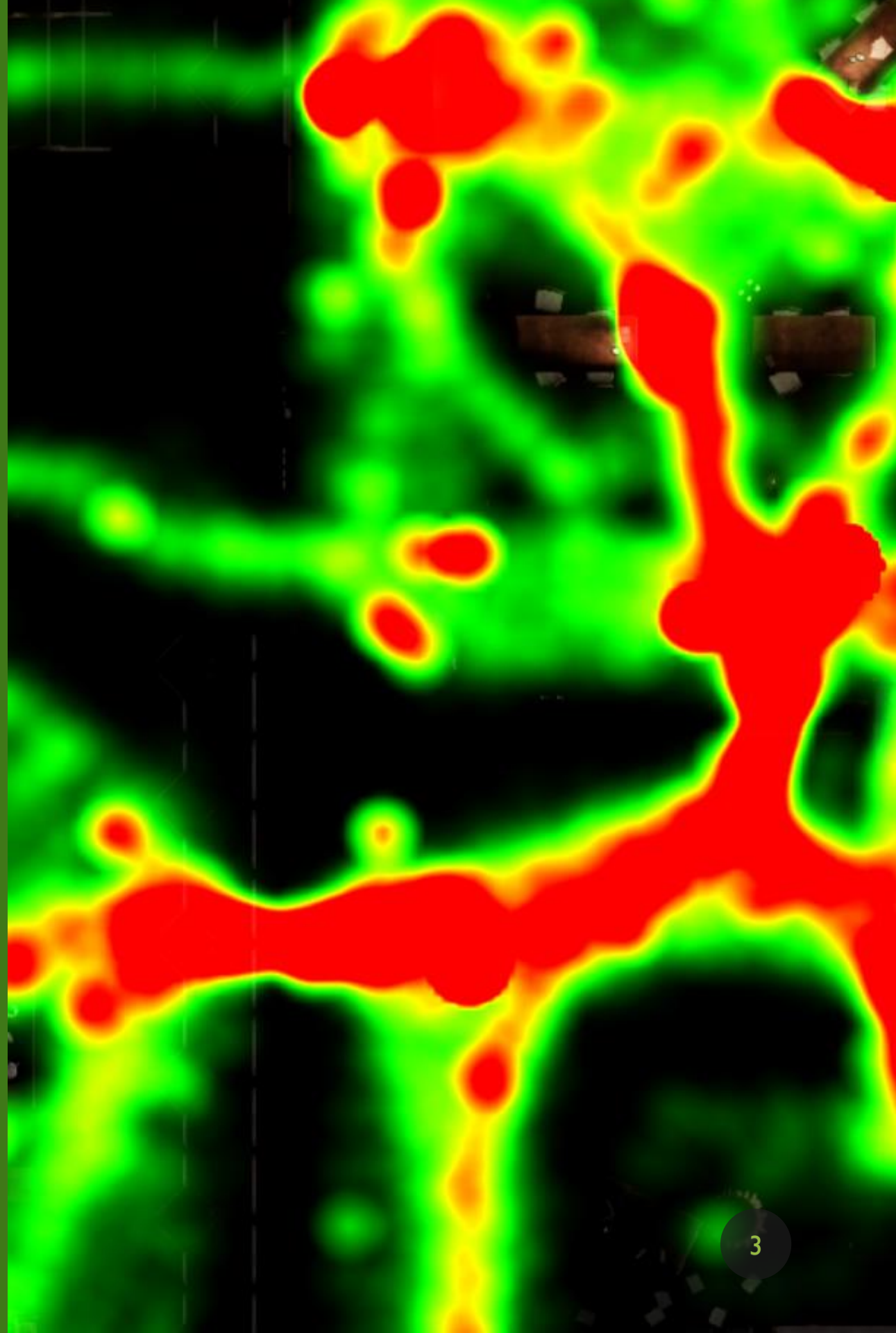
Justin Gibbs

Producer

C26

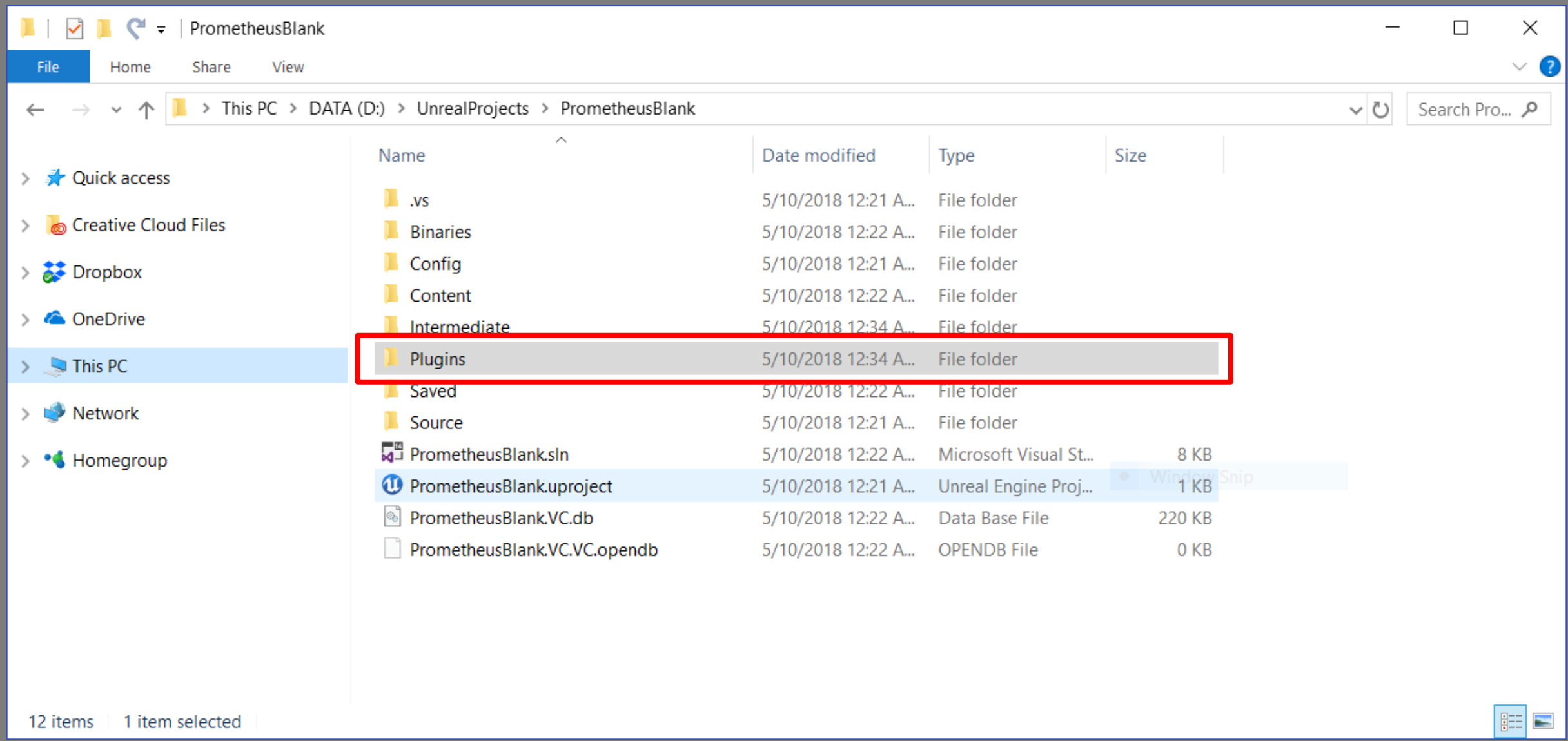
CONTENTS

INSTALLING PROMETHEUS



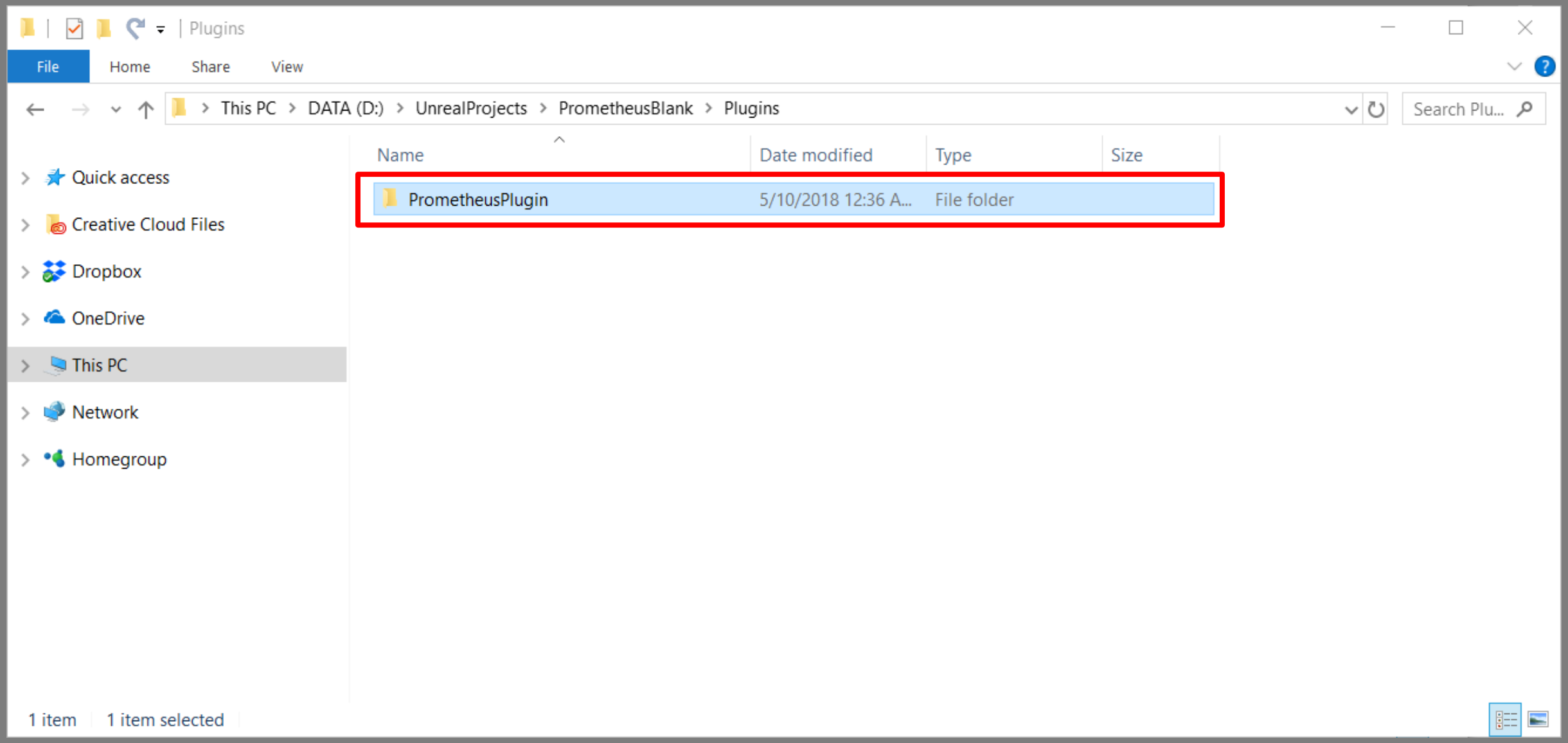
CREATE/LOCATE PLUGIN FOLDER

- Create a new folder, name it "Plugins"
- Unreal knows to look in this folder for plugins



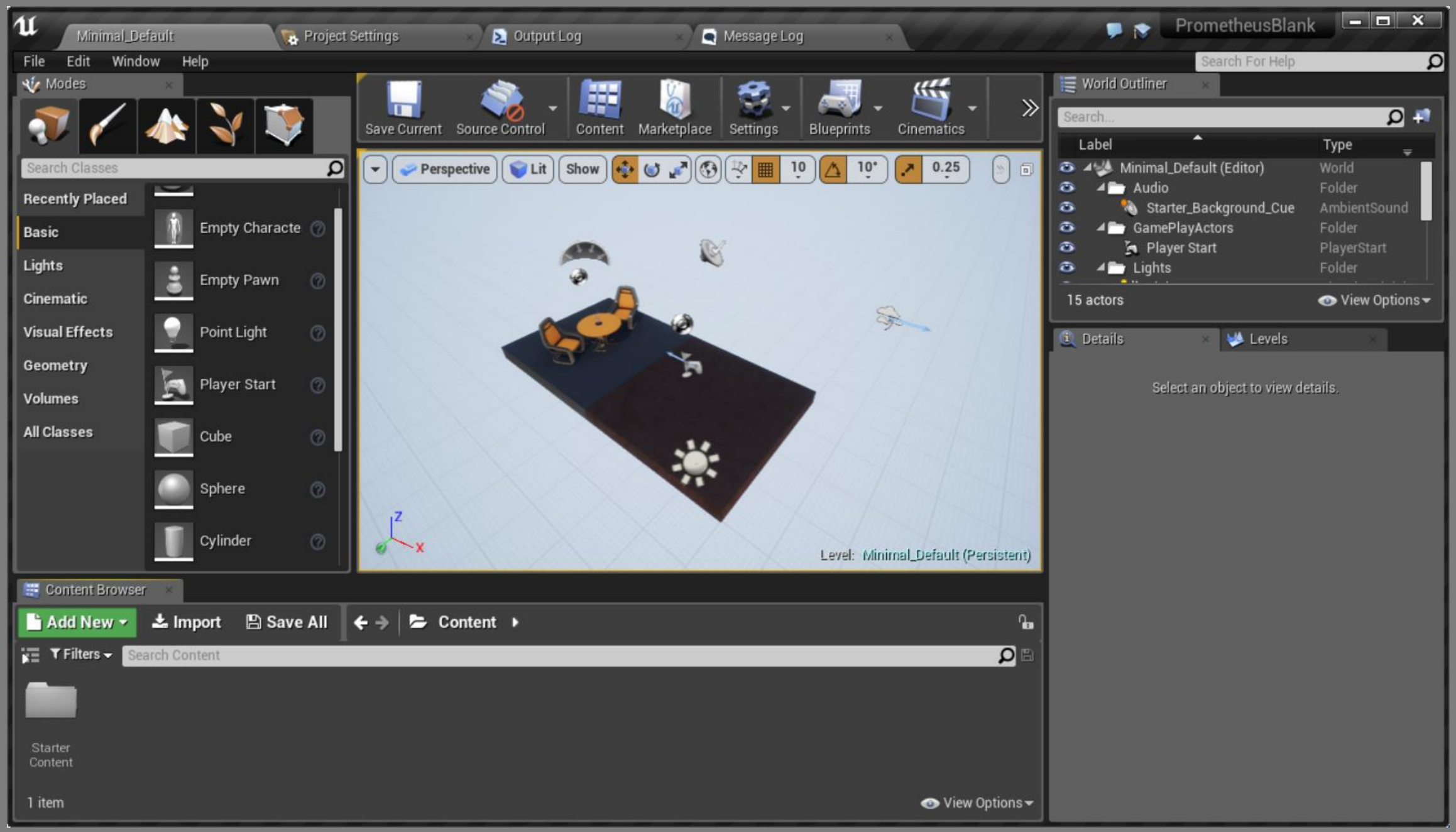
PASTE PROMETHEUS FOLDER INTO PLUGIN FOLDER

- Paste the whole folder in there



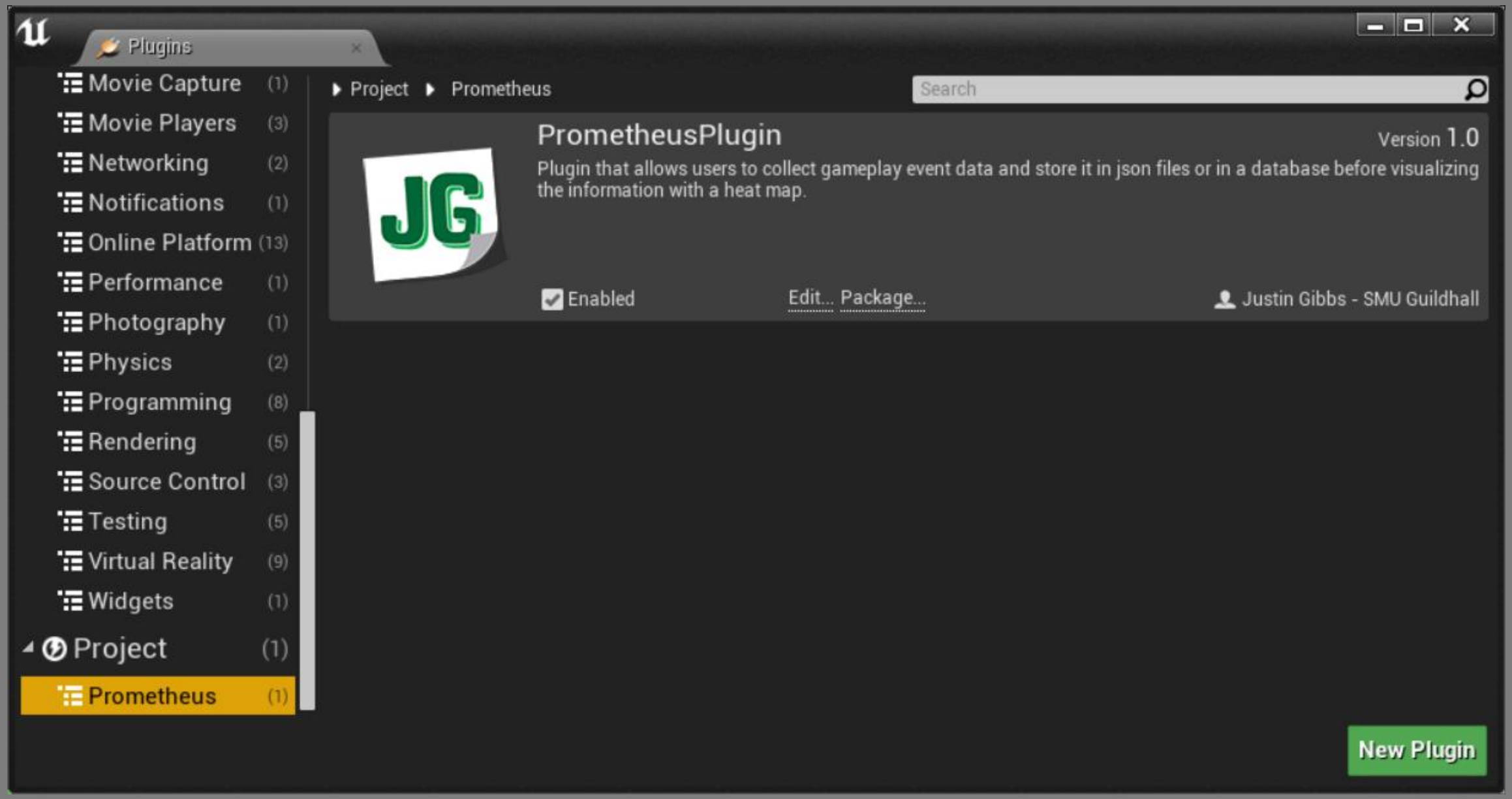
OPEN YOUR PROJECT

- I'm using a blank C++ project called "PrometheusBlank"
- My engine version for these instructions was 4.16.3



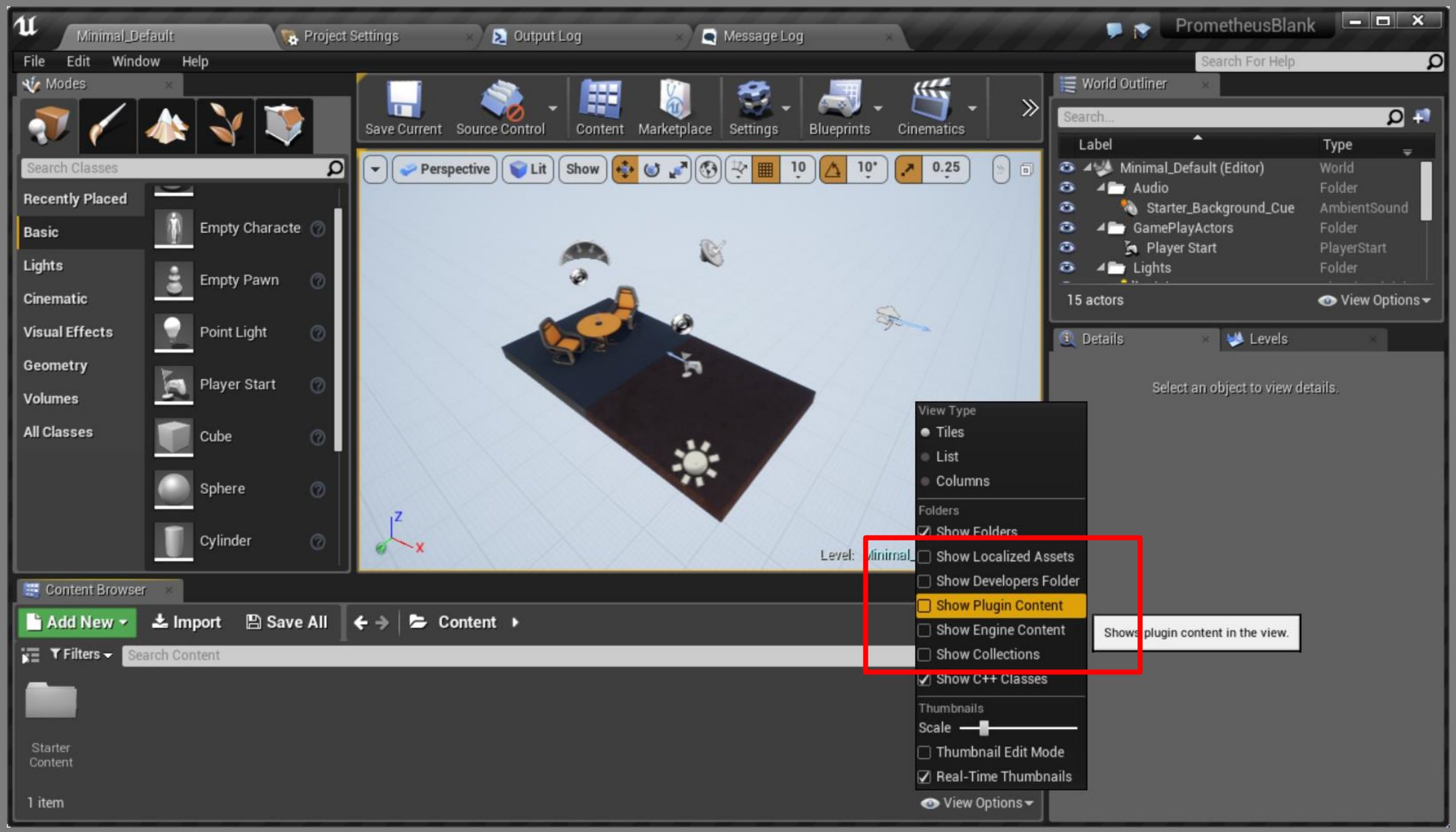
VERIFY THE PLUGIN WAS RECOGNIZED

- Edit > Plugins
- Scrolling to the bottom should reveal Prometheus in the list



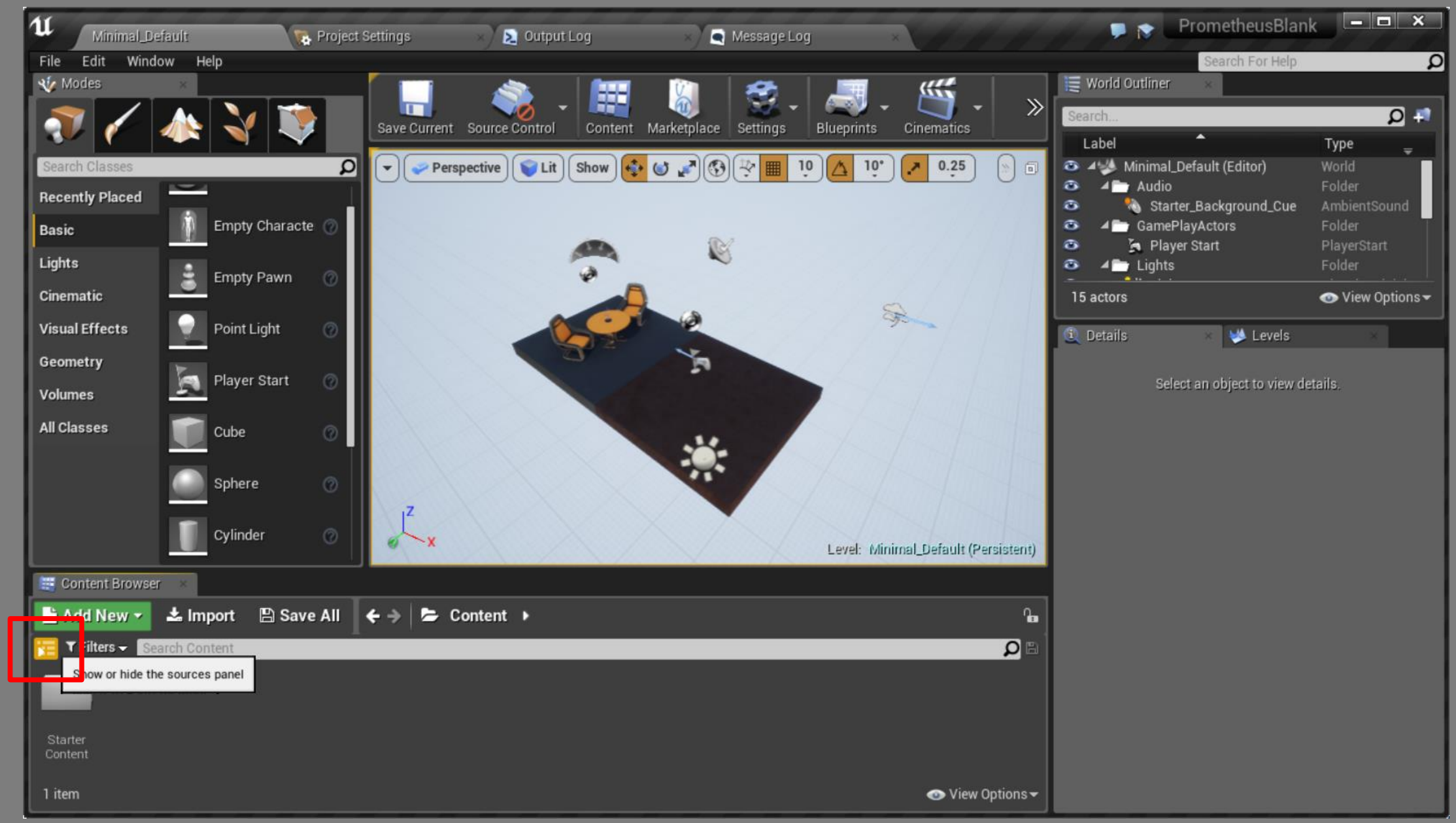
SHOW PLUGIN CONTENT

- Click "View Options" at the bottom right of content browser
- Make sure "Show Plugin Content" is checked



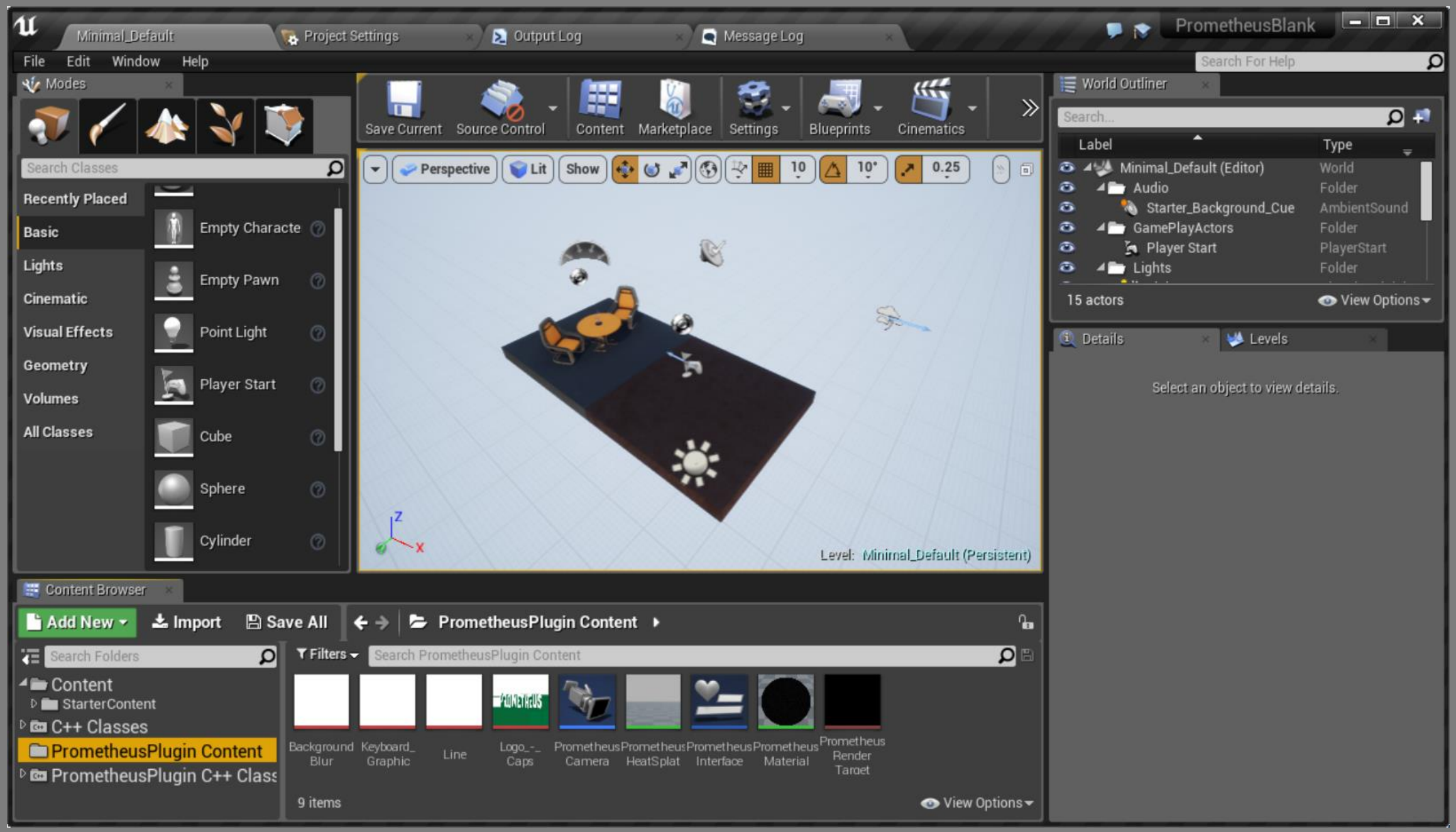
SHOW SOURCES PANEL

- Click the button at the top left of the content browser to reveal the sources panel

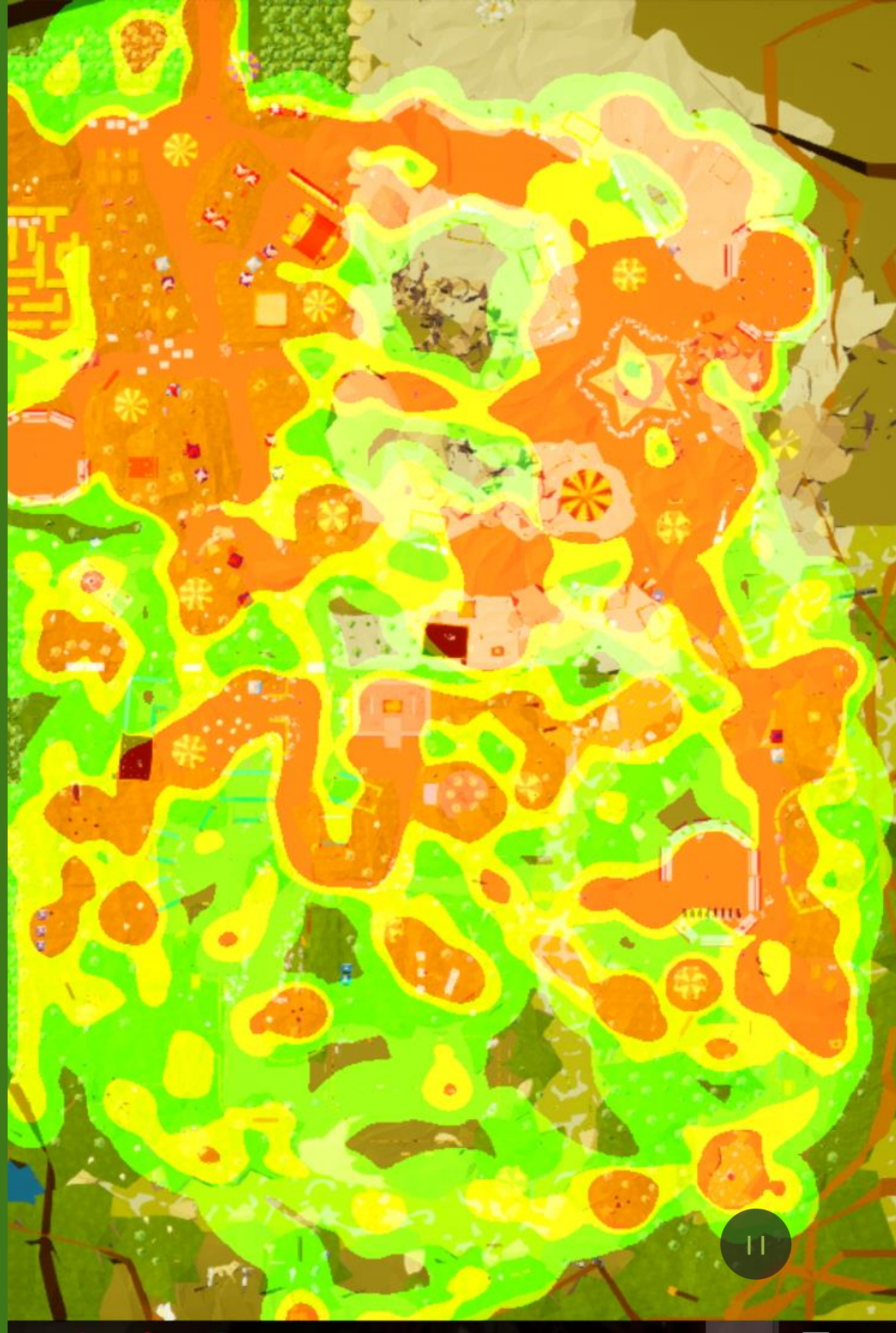


VIEW PROMETHEUS CONTENT

- There should be a folder available that displays the Prometheus content

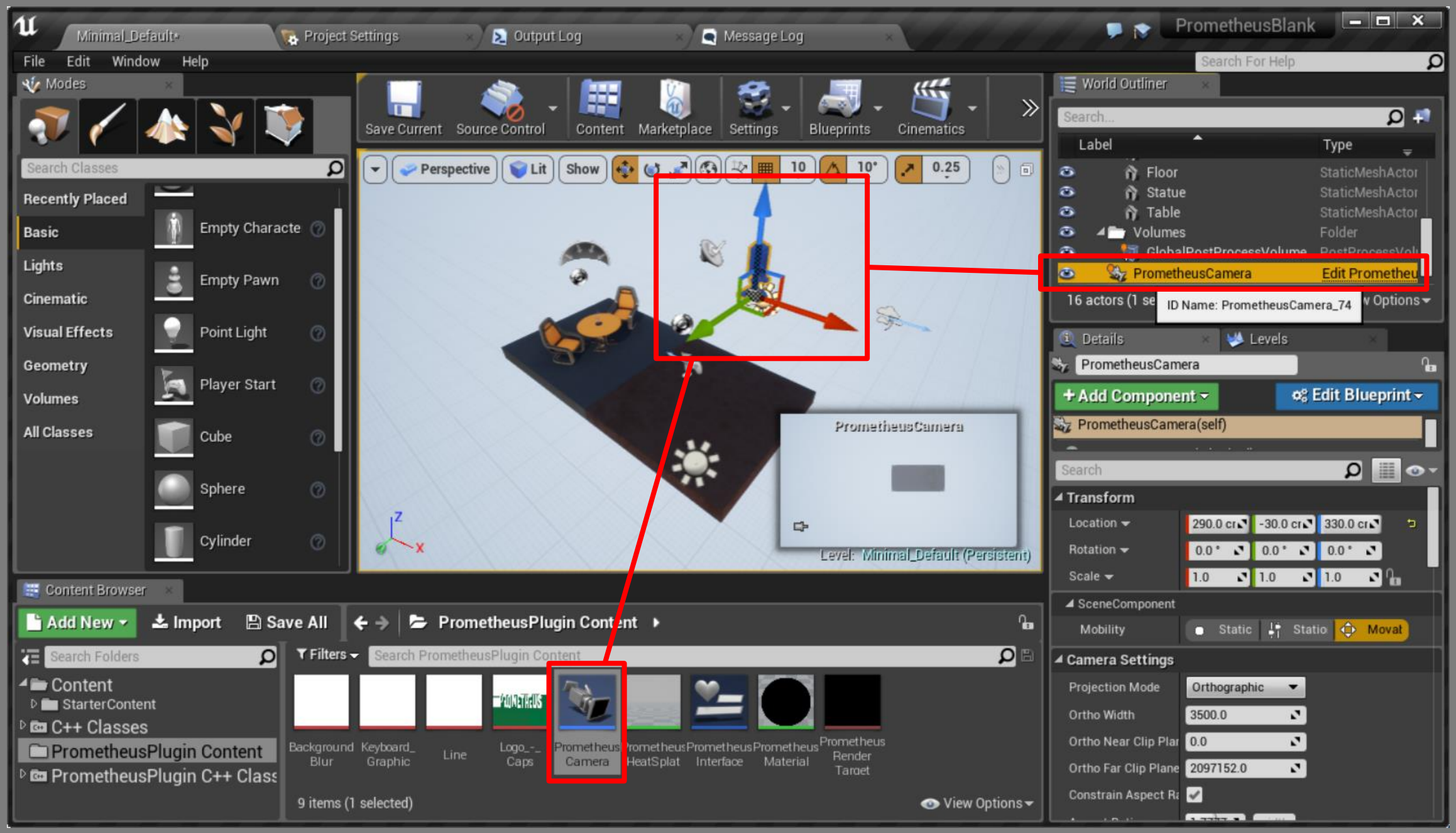


INTEGRATING PROMETHEUS



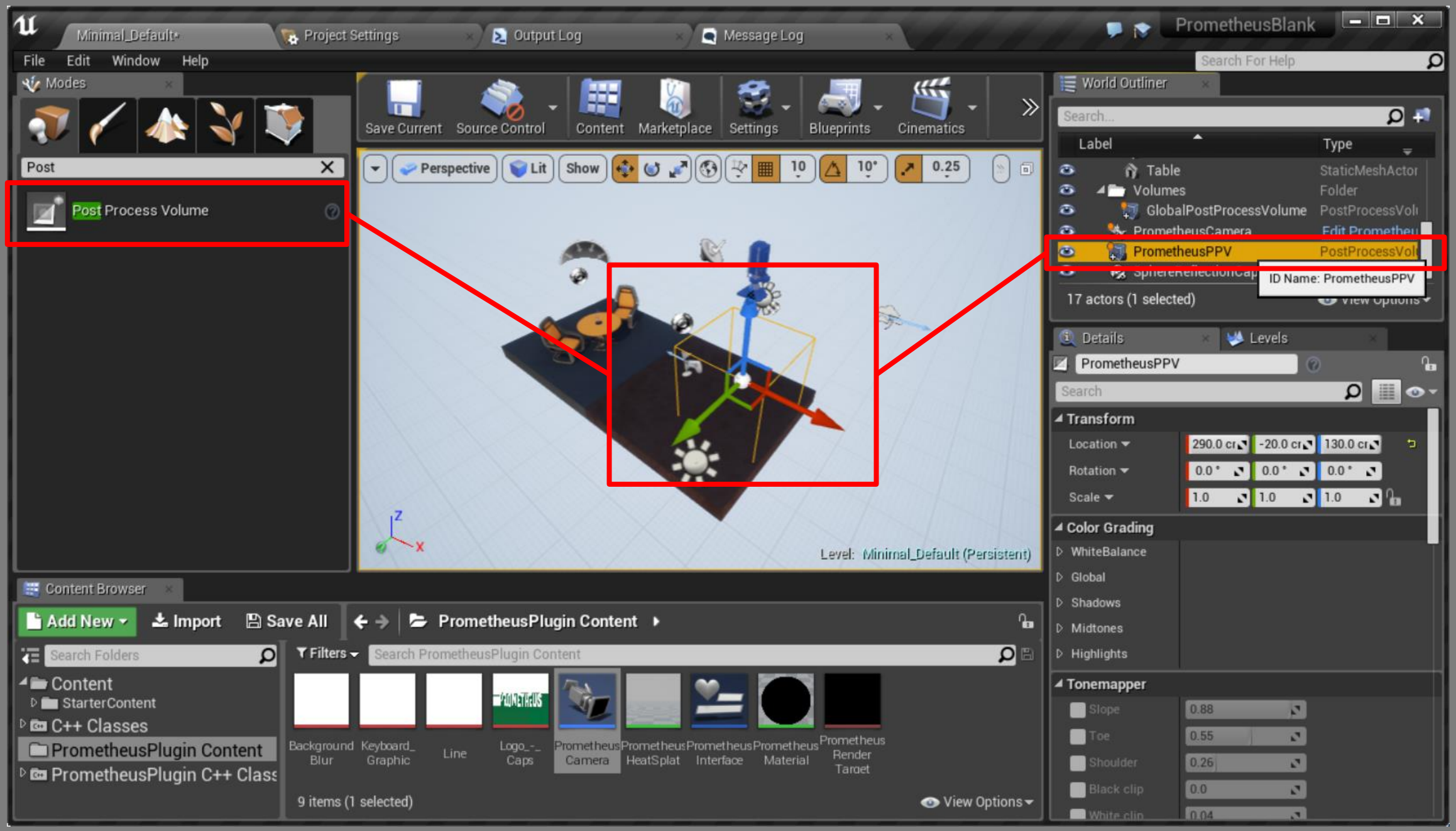
DRAG PROMETHEUS CAMERA INTO EACH SCENE

- Every scene needs a Prometheus camera if there is data being collected in it



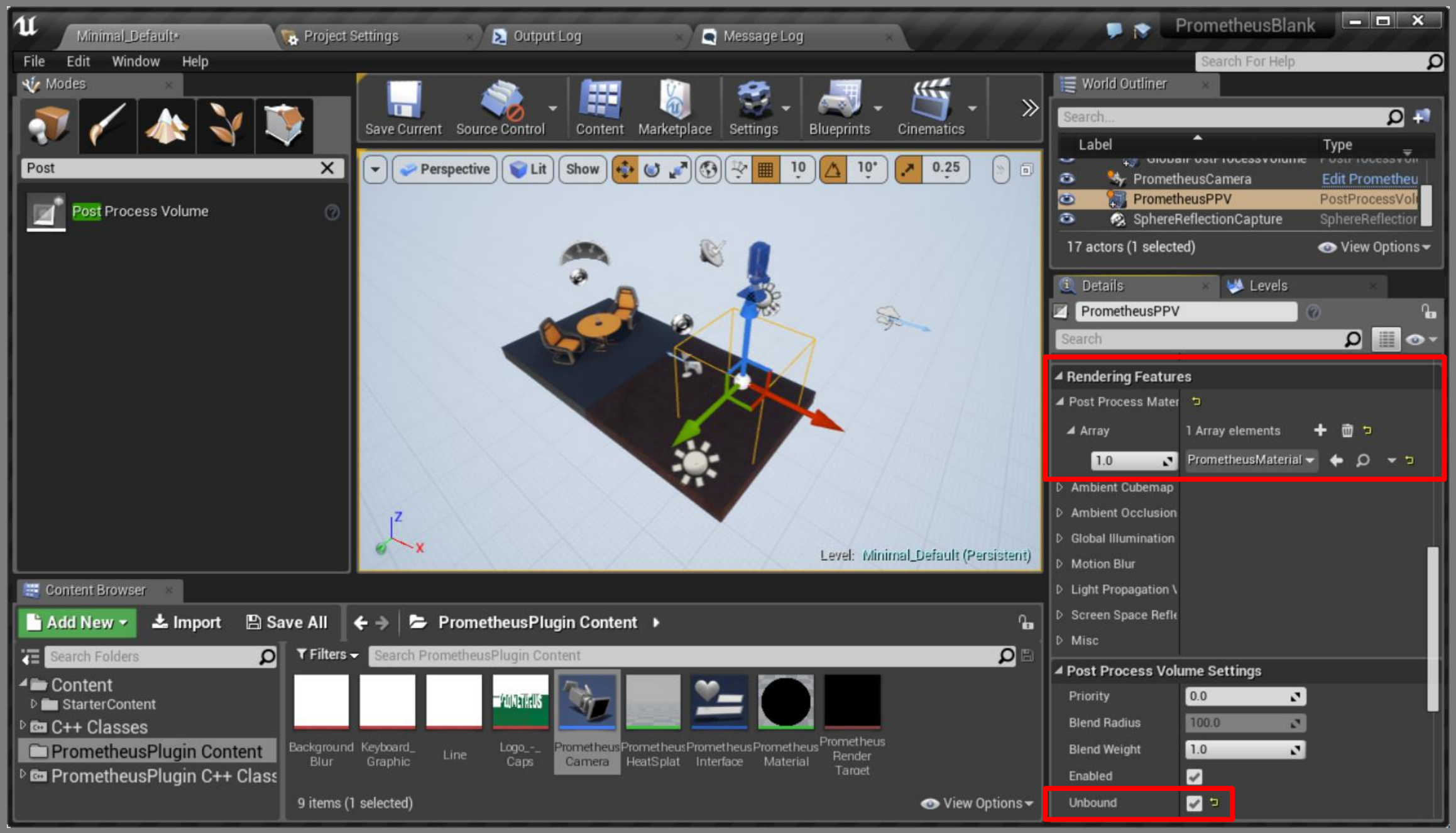
DRAG A POST PROCESSING VOLUME INTO EACH SCENE

- Name them "PrometheusPPV" so the cameras can find them
- Every scene that has a camera needs a PPV



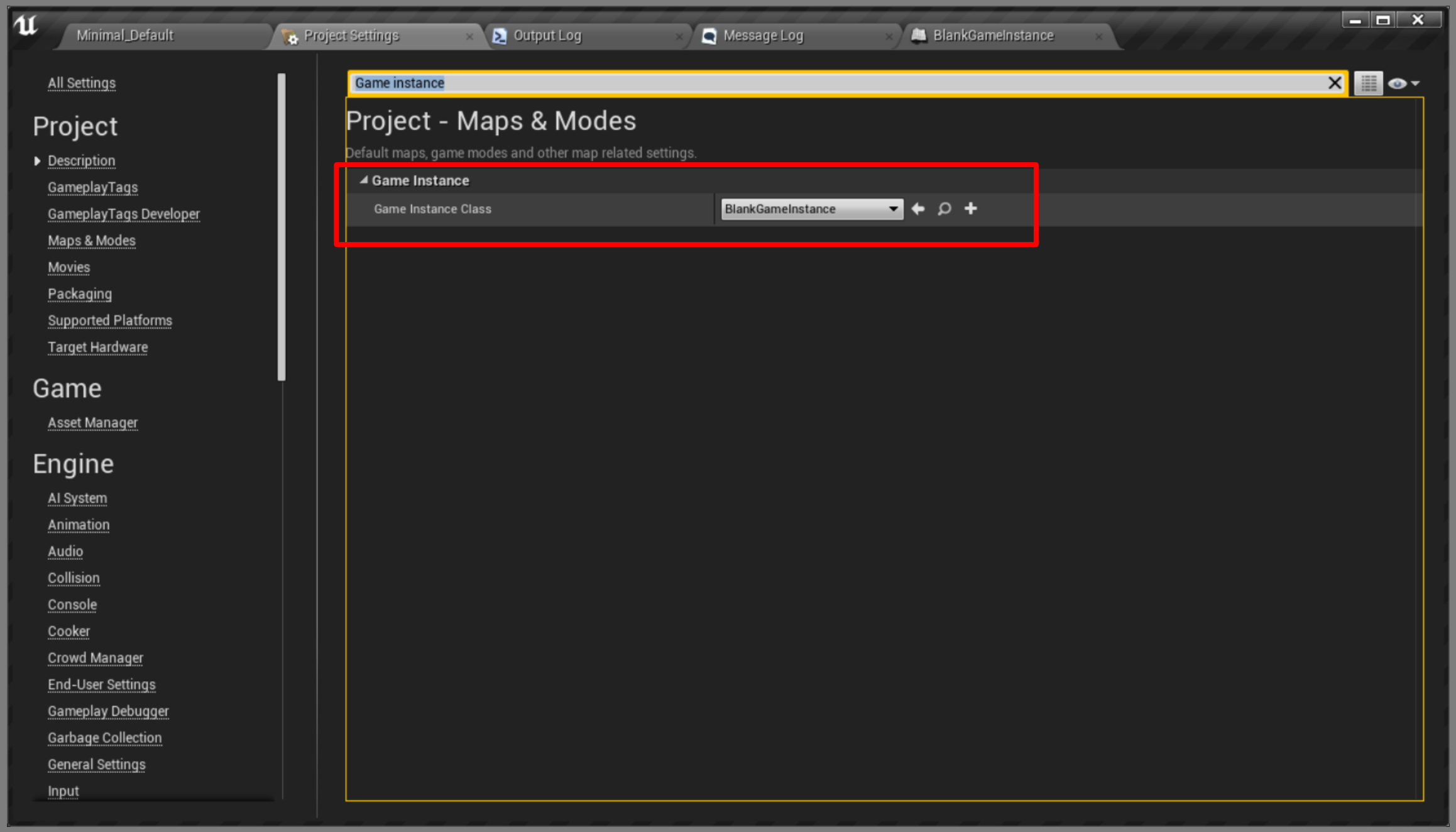
MODIFY SETTINGS ON THE PPV

- "Post Process Materials" needs a reference to PrometheusMaterial
- Check the "Unbound" box so the camera gets the effect anywhere



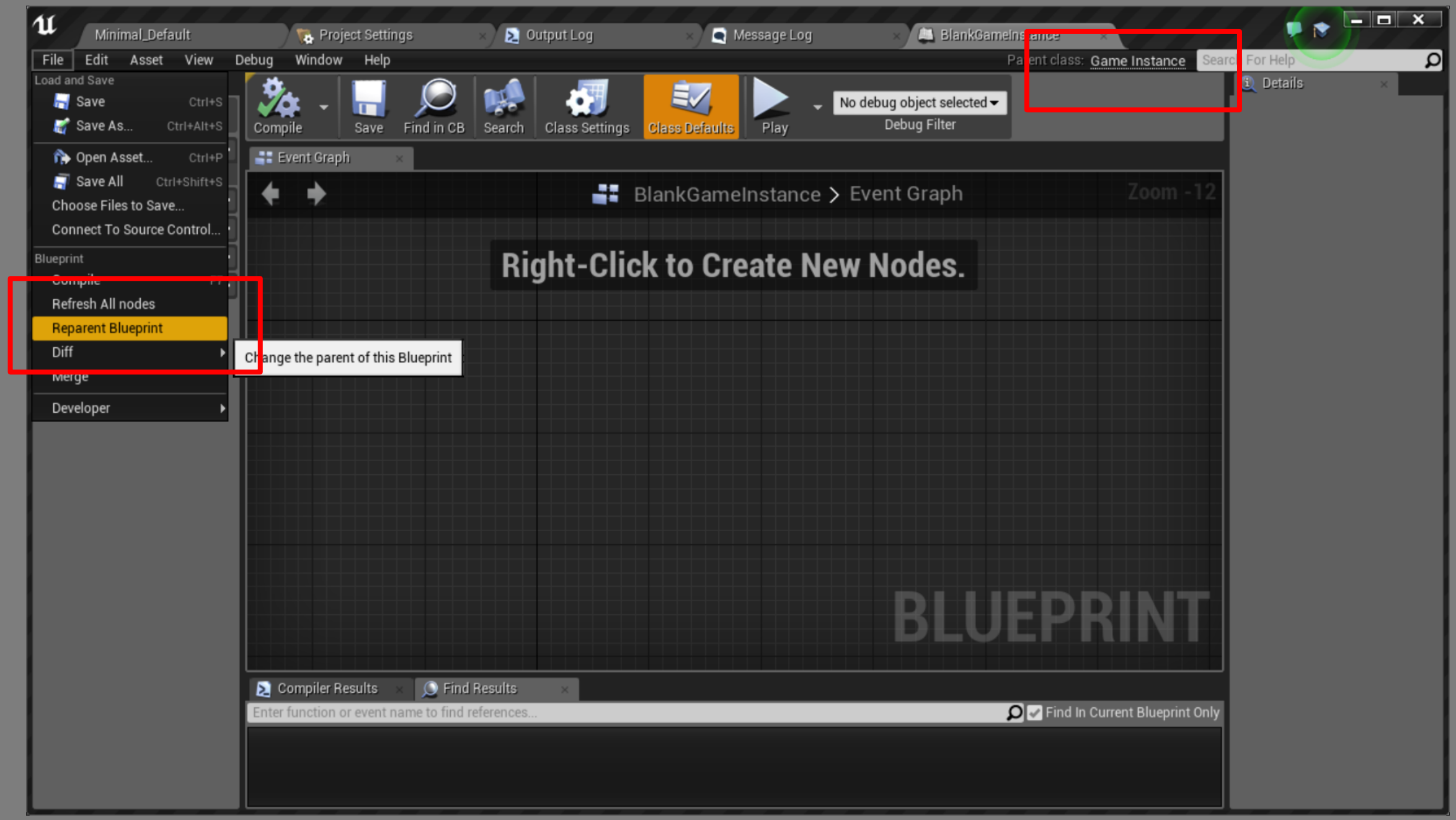
SET THE GAME INSTANCE TO MYGAMEINSTANCE

- Edit > Project Settings
- If your game uses a custom game instance it needs to be reparented



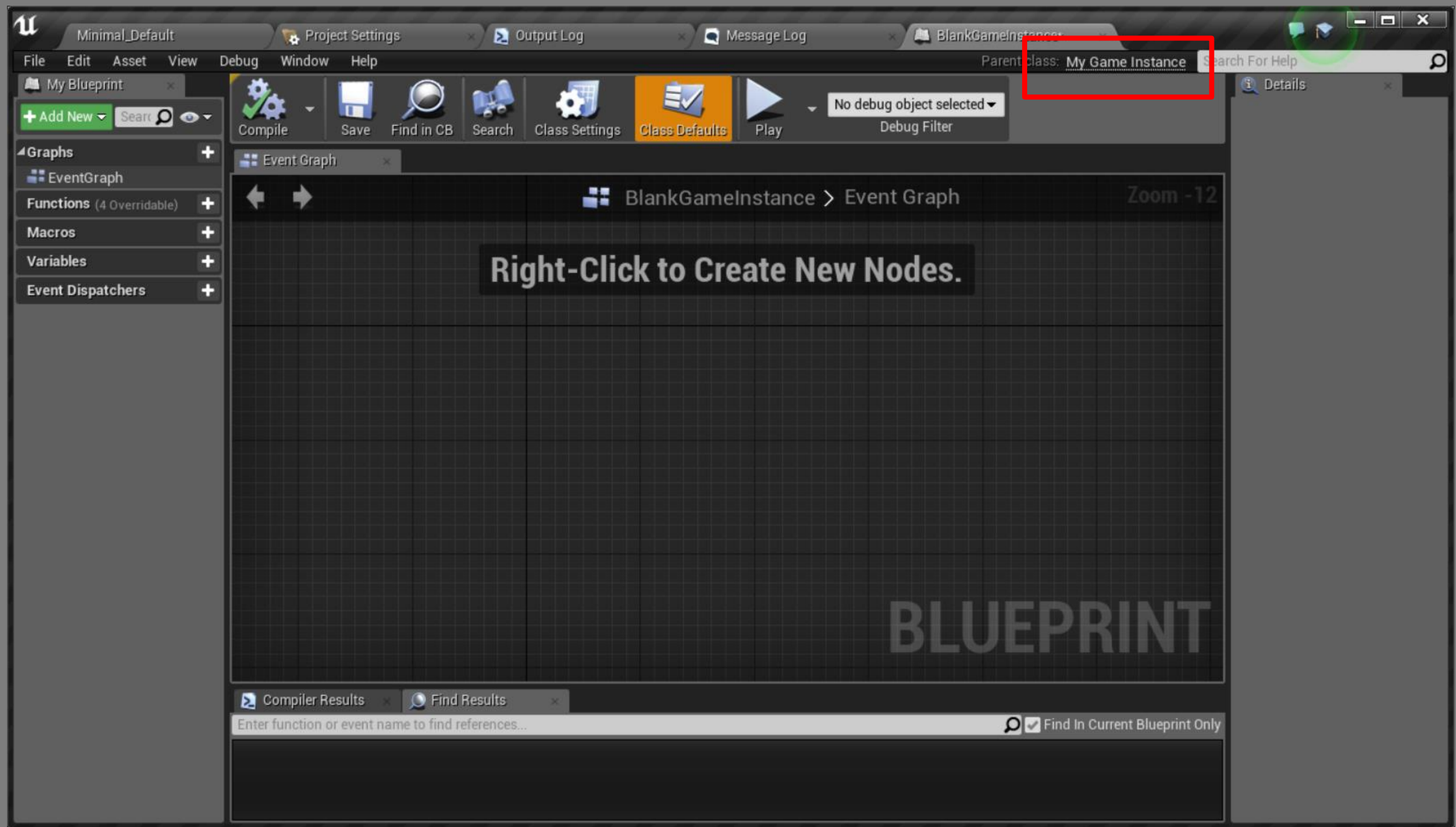
IF NECESSARY REPARENT GAME INSTANCE BLUEPRINT

- Only needs to be done if you use a custom game instance
- Click "Reparent Blueprint" and select "MyGameInstance"

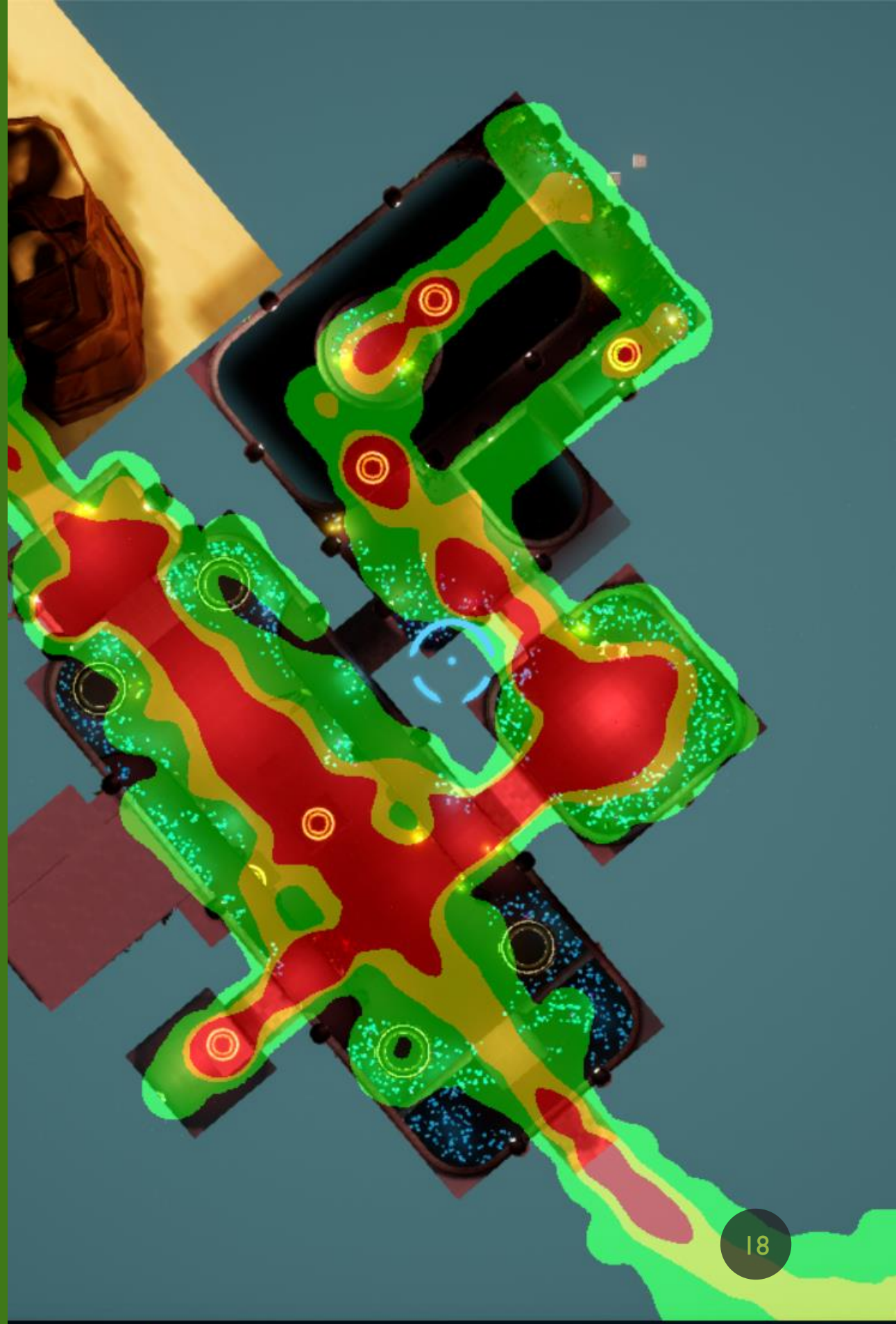


RESULT OF REPARENTING BLUEPRINT

- The parent class is visible in the top right
- This "BlankGameInstance" can now be set in the project settings

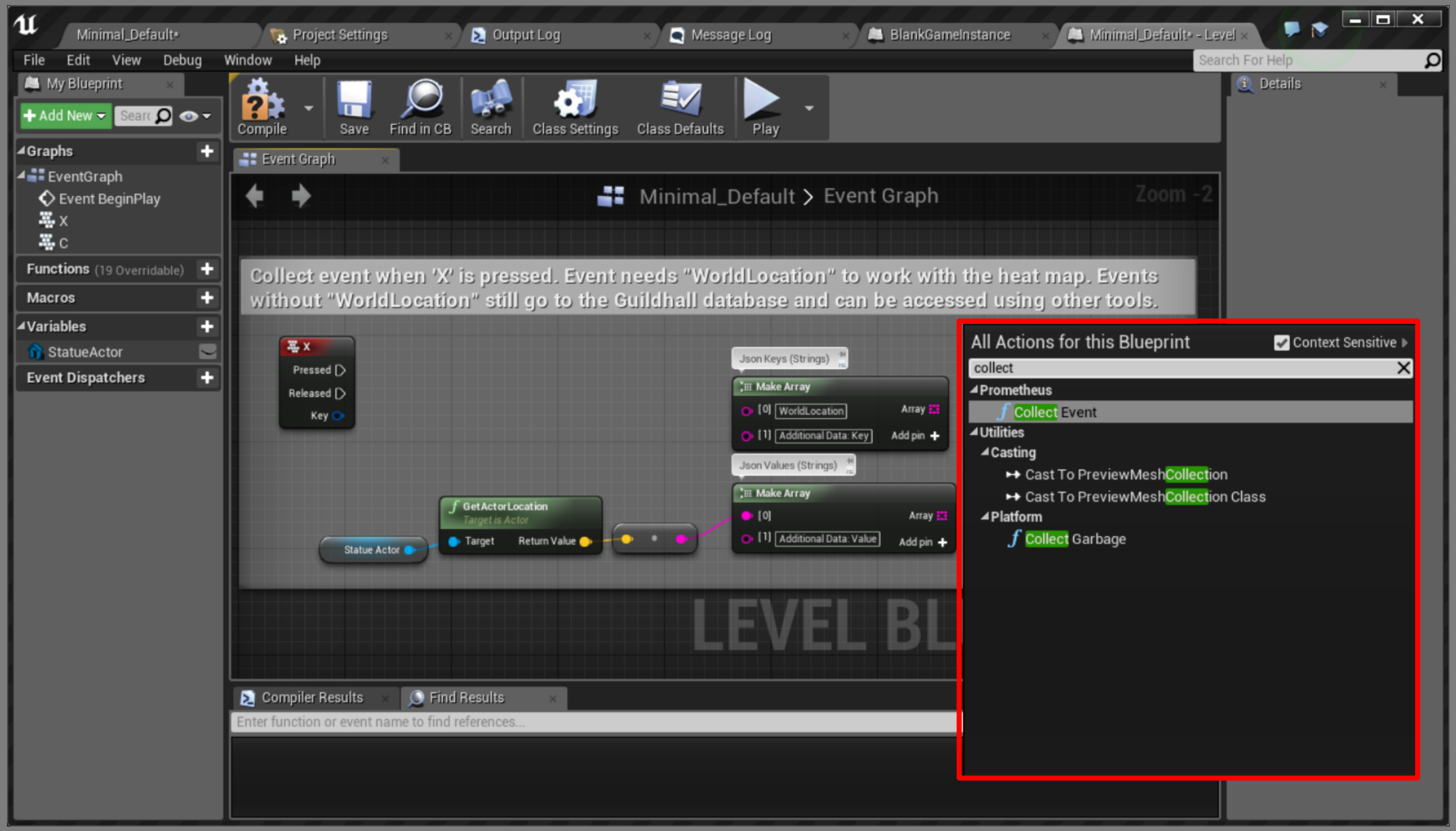


COLLECTING DATA WITH PROMETHEUS



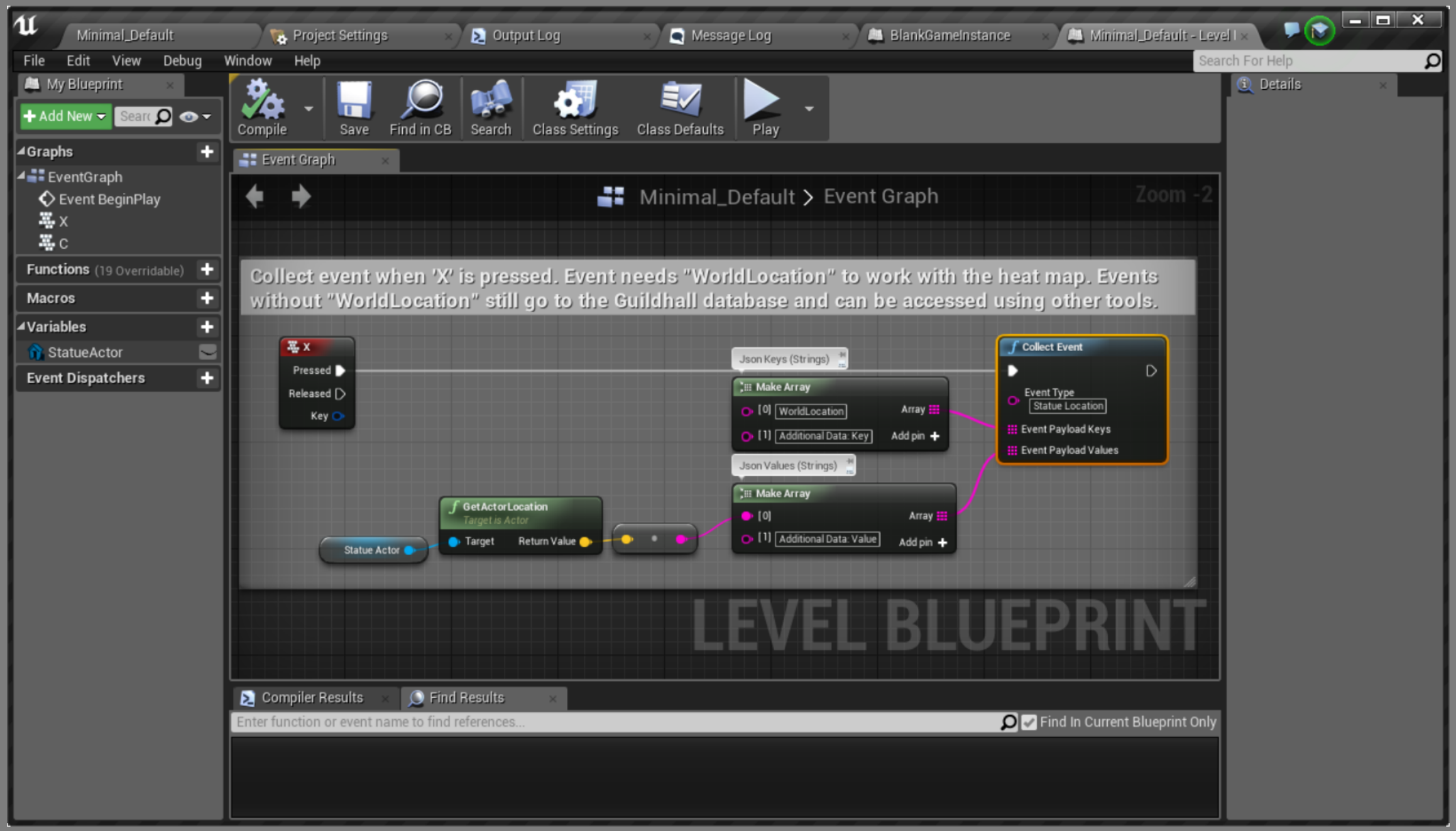
ADD A “COLLECT EVENT” NODE

- Right click and type collect event to place the node where you want to collect data



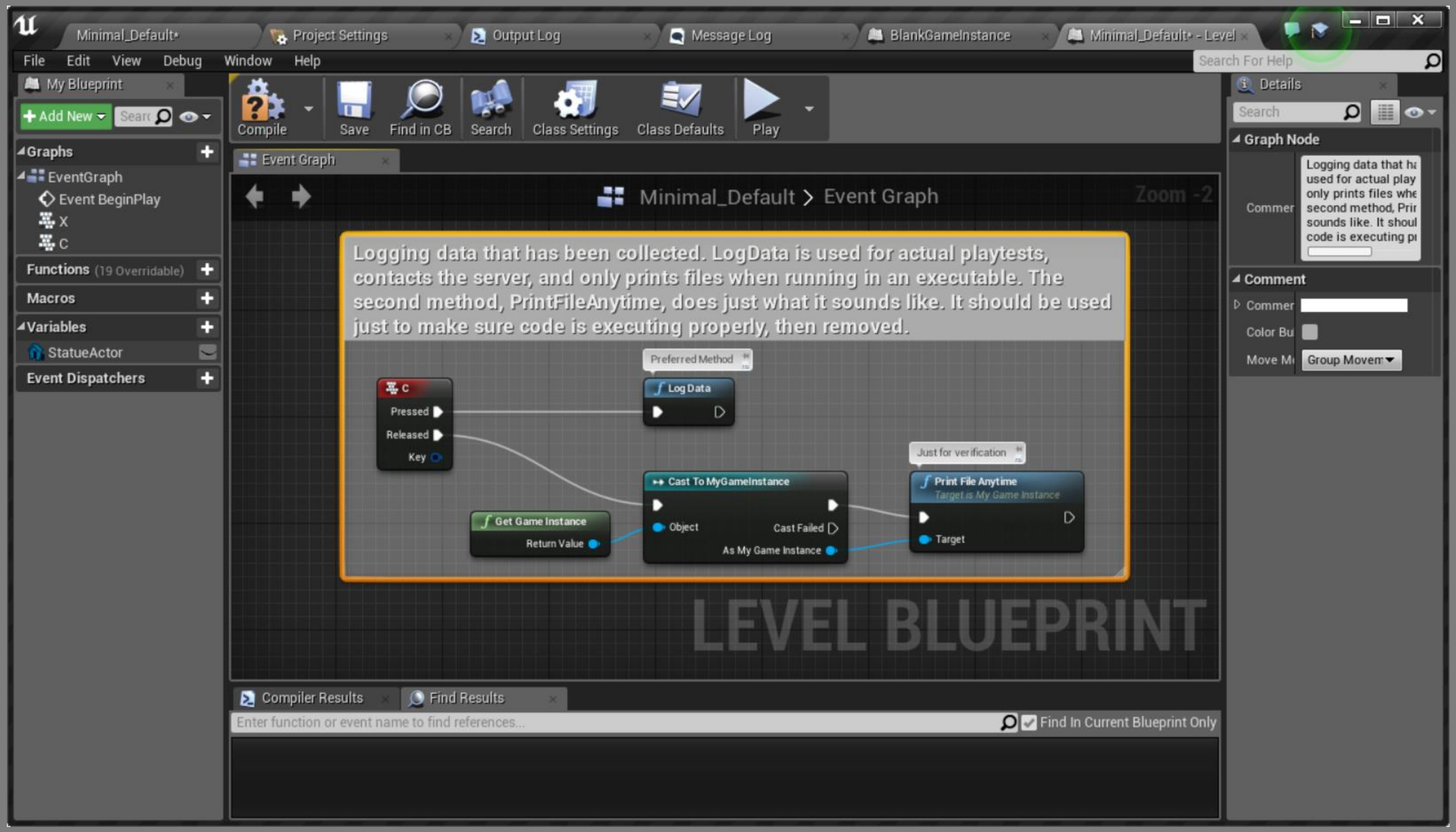
ATTACH THE "COLLECT EVENT" NODE

- The Keys and Values arrays can be anything you want them to be, but you need "WorldLocation" if you want to view the event with the heat map



LOG ANY COLLECTED DATA

- Follow the same process to get to the nodes
- Log Data is preferred, but only works when running in an executable



OUTPUT JSON STRUCTURE

- This is the output json from the events on the previous slides
- This represents the structure a file must have for Prometheus to read it

D:\UnrealProjects\PrometheusBlank\Temp\PrometheusBlank__2018-5-10__1-20-56__PrometheusOutput.json - Notepad++

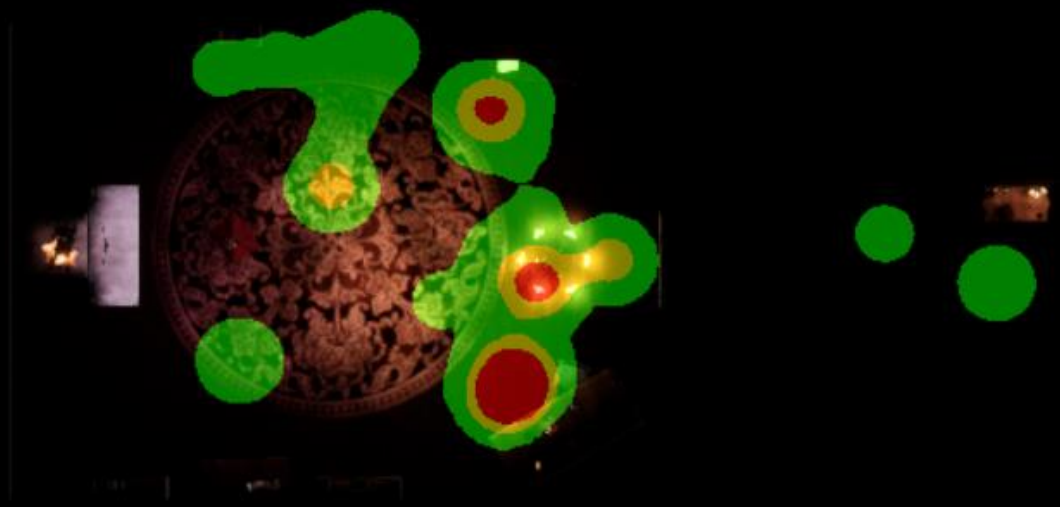
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?

PrometheusBlank__2018-5-10__1-20-56__PrometheusOutput.json


```
[
    {
        "ApplicationId": "PrometheusBlank",
        "EventType": "Statue Location",
        "EventDate": "2018.05.10-06.20.55",
        "EventPayload":
        {
            "WorldLocation": "X=-181.000 Y=1.000 Z=102.000",
            "Additional Data: Key": "Additional Data: Value"
        }
    },
    {
        "ApplicationId": "PrometheusBlank",
        "EventType": "Statue Location",
        "EventDate": "2018.05.10-06.20.56",
        "EventPayload":
        {
            "WorldLocation": "X=-181.000 Y=1.000 Z=102.000",
            "Additional Data: Key": "Additional Data: Value"
        }
    }
]
```

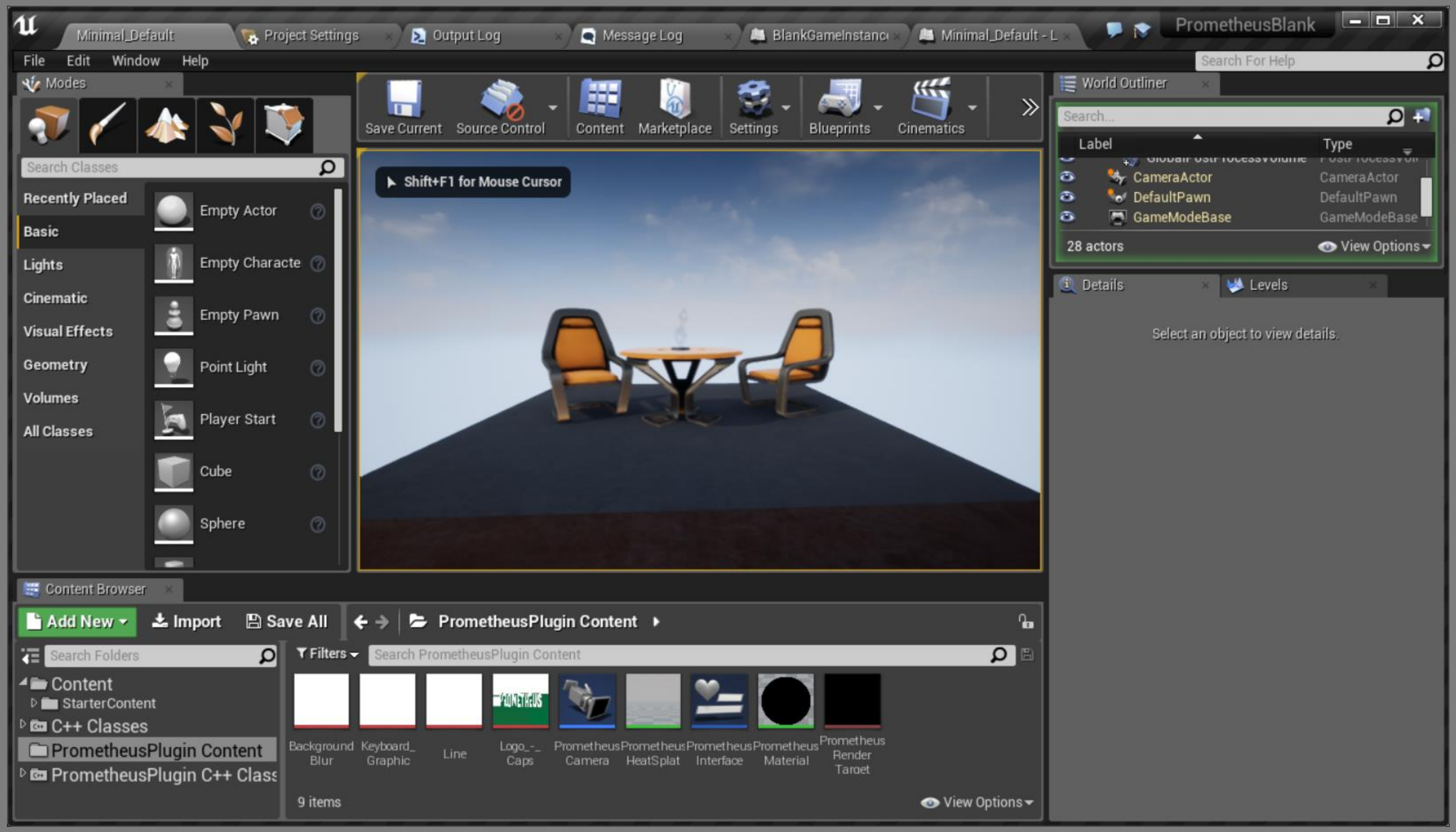

JSON file | length : 517 lines : 22 | Ln : 1 Col : 1 Sel : 0 | 0 | Windows (CR LF) | UTF-8 | INS

VIEWING DATA WITH PROMETHEUS



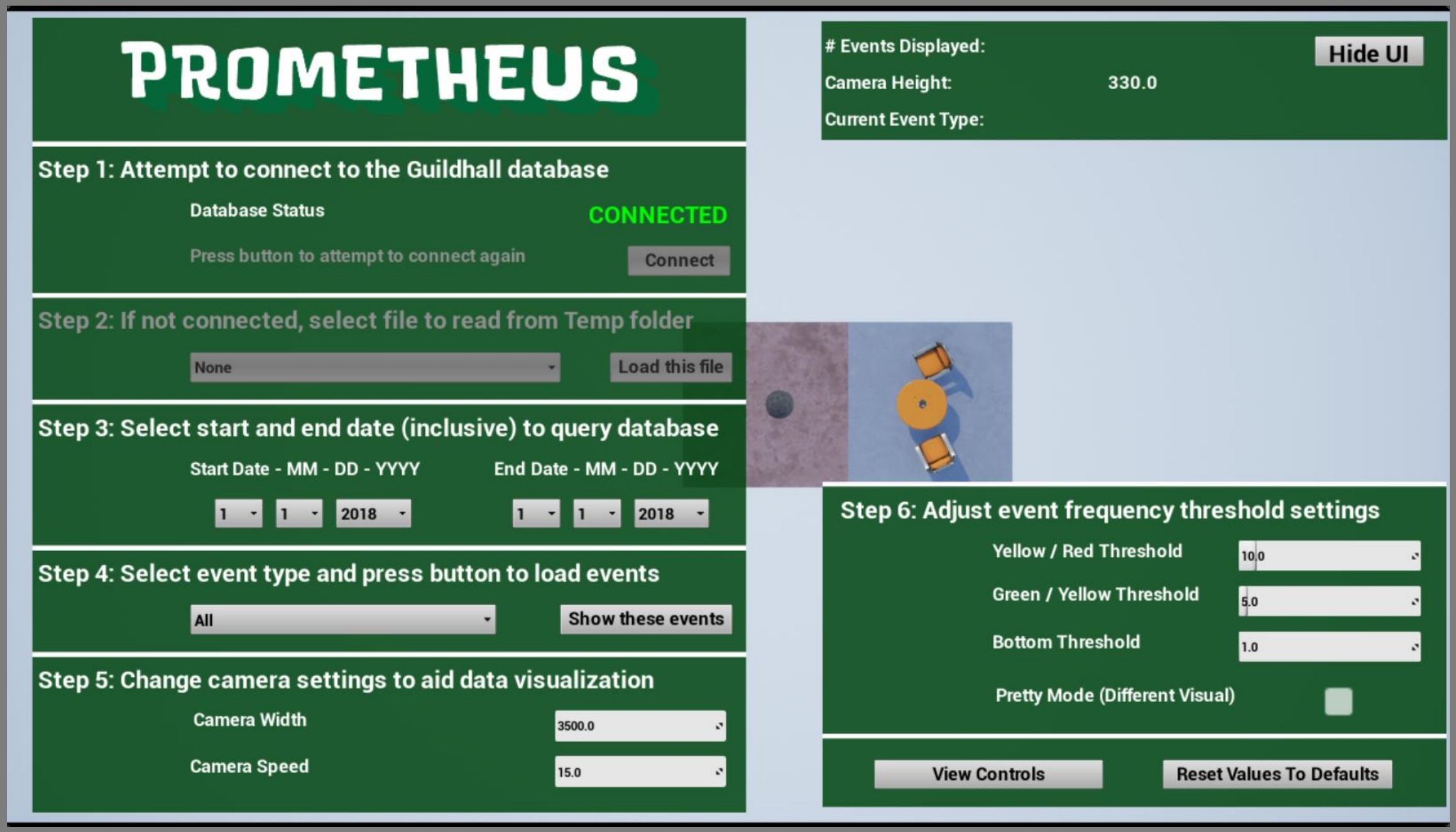
PRESS PLAY

- Run your game to access the heat map visualizer

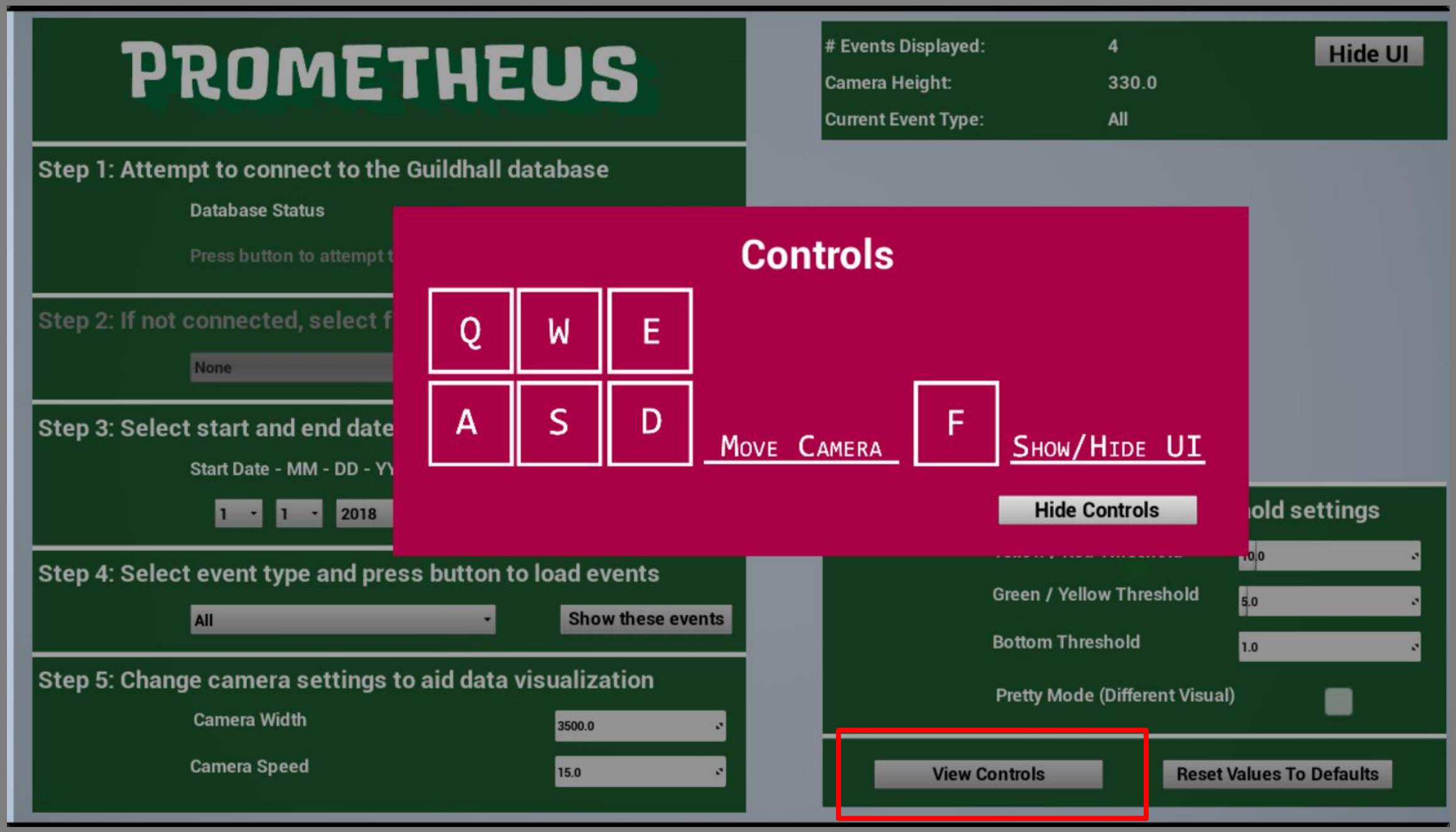


PRESS 'F10' WHILE RUNNING THE GAME TO OPEN PROMETHEUS

- Prometheus's user interface
- Follow the steps to visualize data

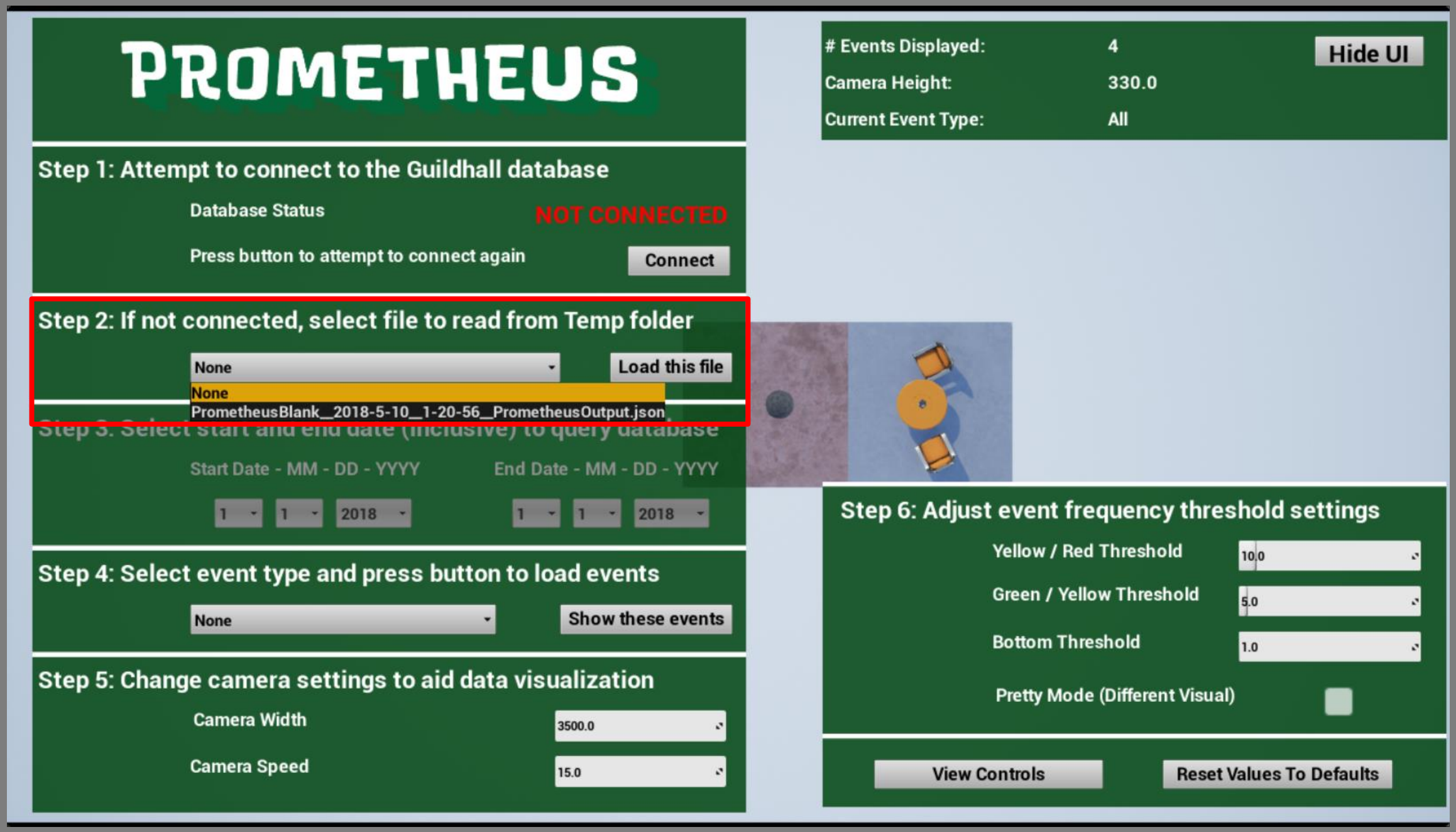


VIEW CONTROLS BY CLICKING THE "VIEW CONTROLS" BUTTON



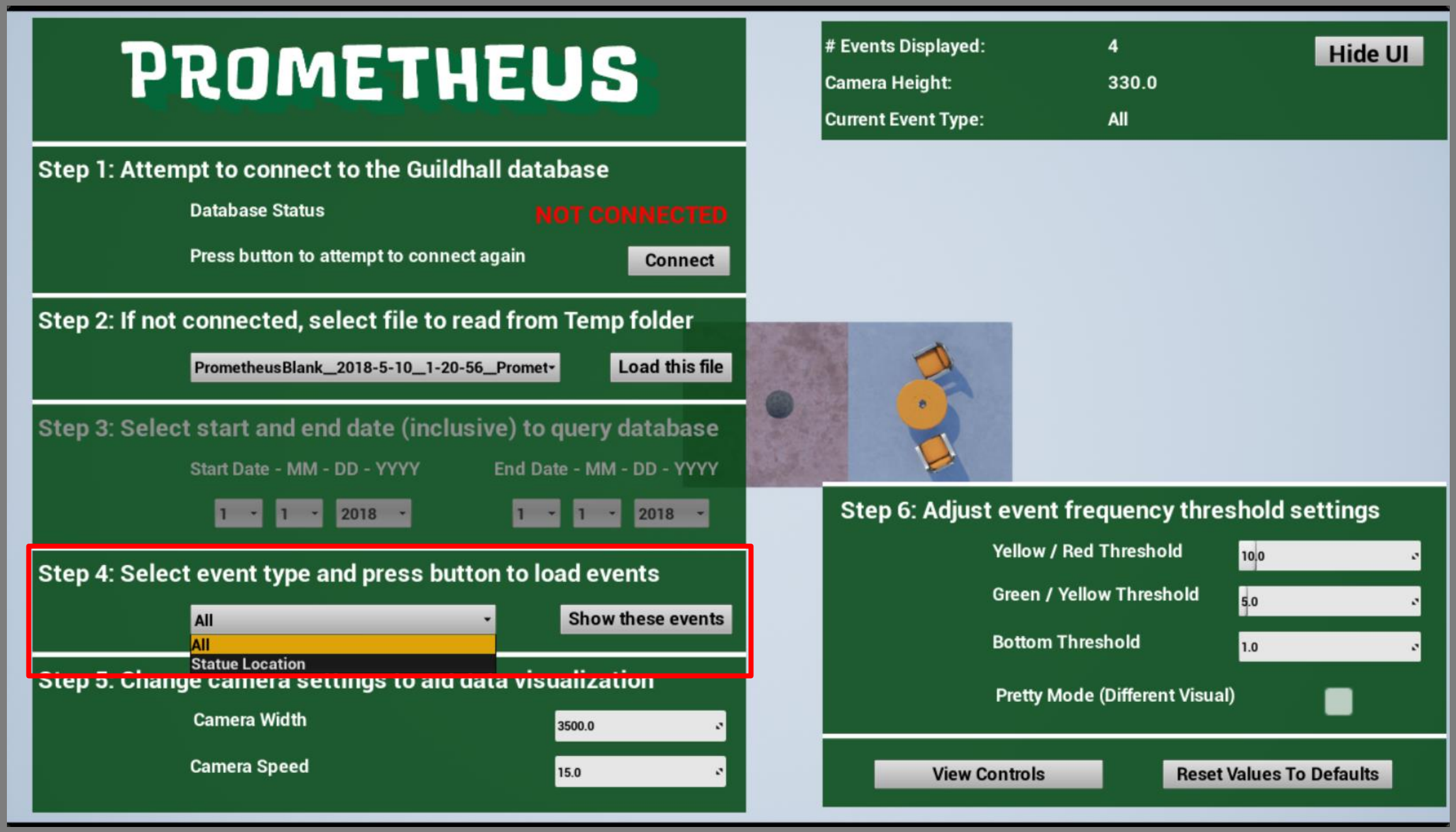
IF NOT CONNECTED: SELECT A FILE FROM THE DROP-DOWN

- We made this file earlier by using the "PrintFileAnywhere" node
- Click the "Load this file" button to populate the event types in Step 4



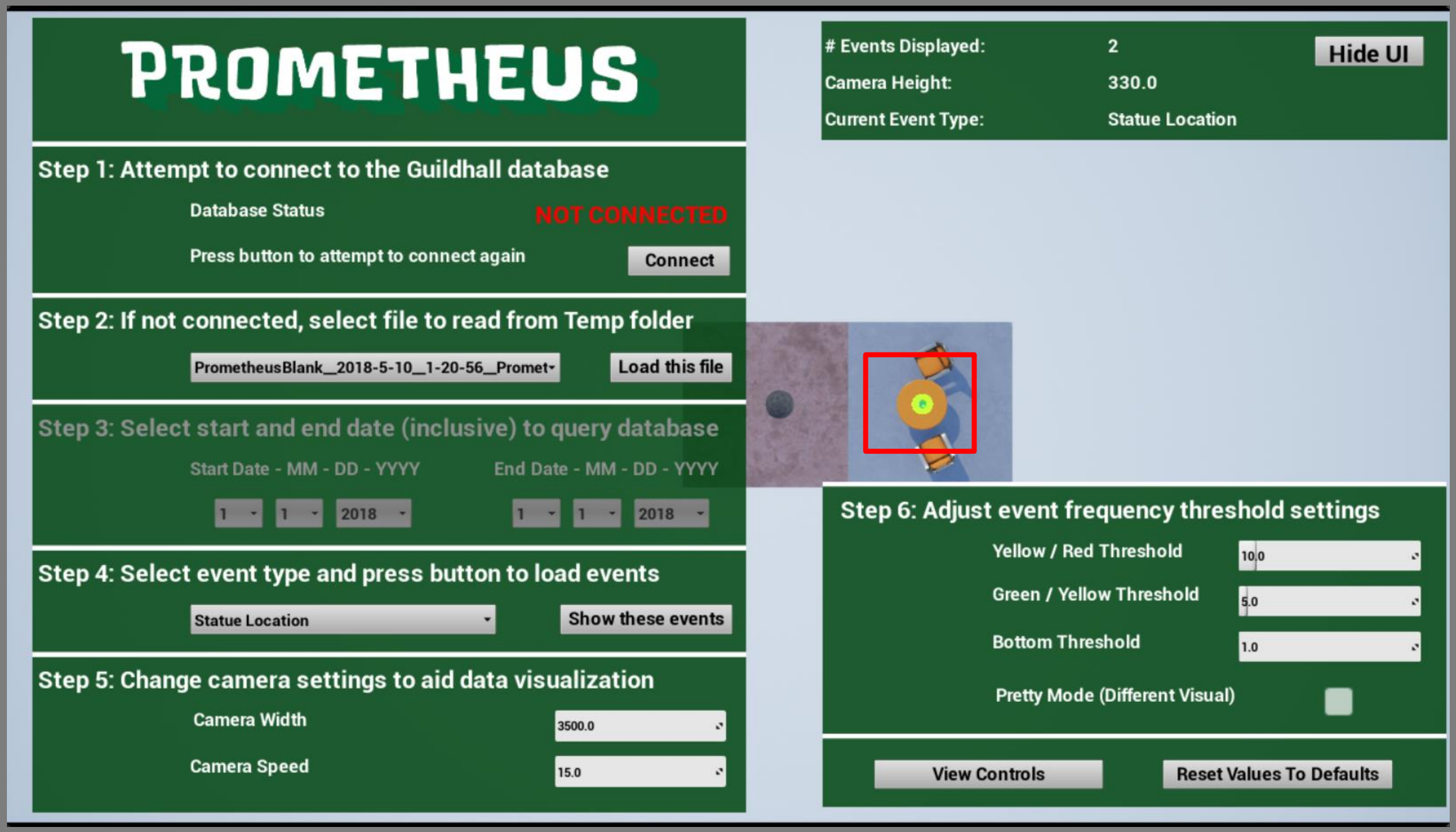
IF NOT CONNECTED: SELECT AN EVENT TO DISPLAY

- Here is our "Statue Location" event that displays the location of a statue
- Click the "Show these events" to display the events



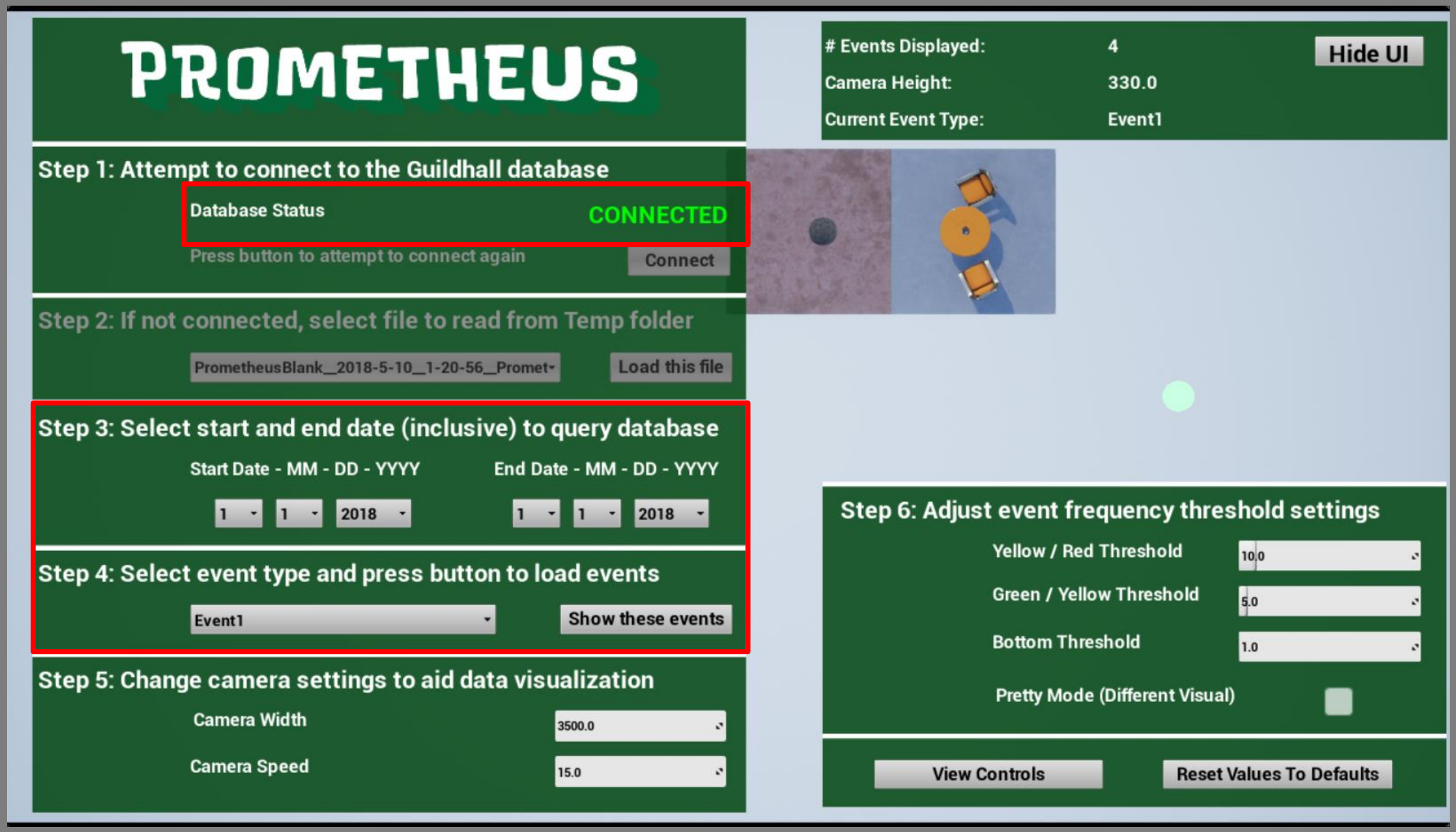
IF NOT CONNECTED: VIEW HEAT MAP DISPLAY

- Here we can see the fruits of our labors as the heat map is displaying over the table, which is where the statue is located



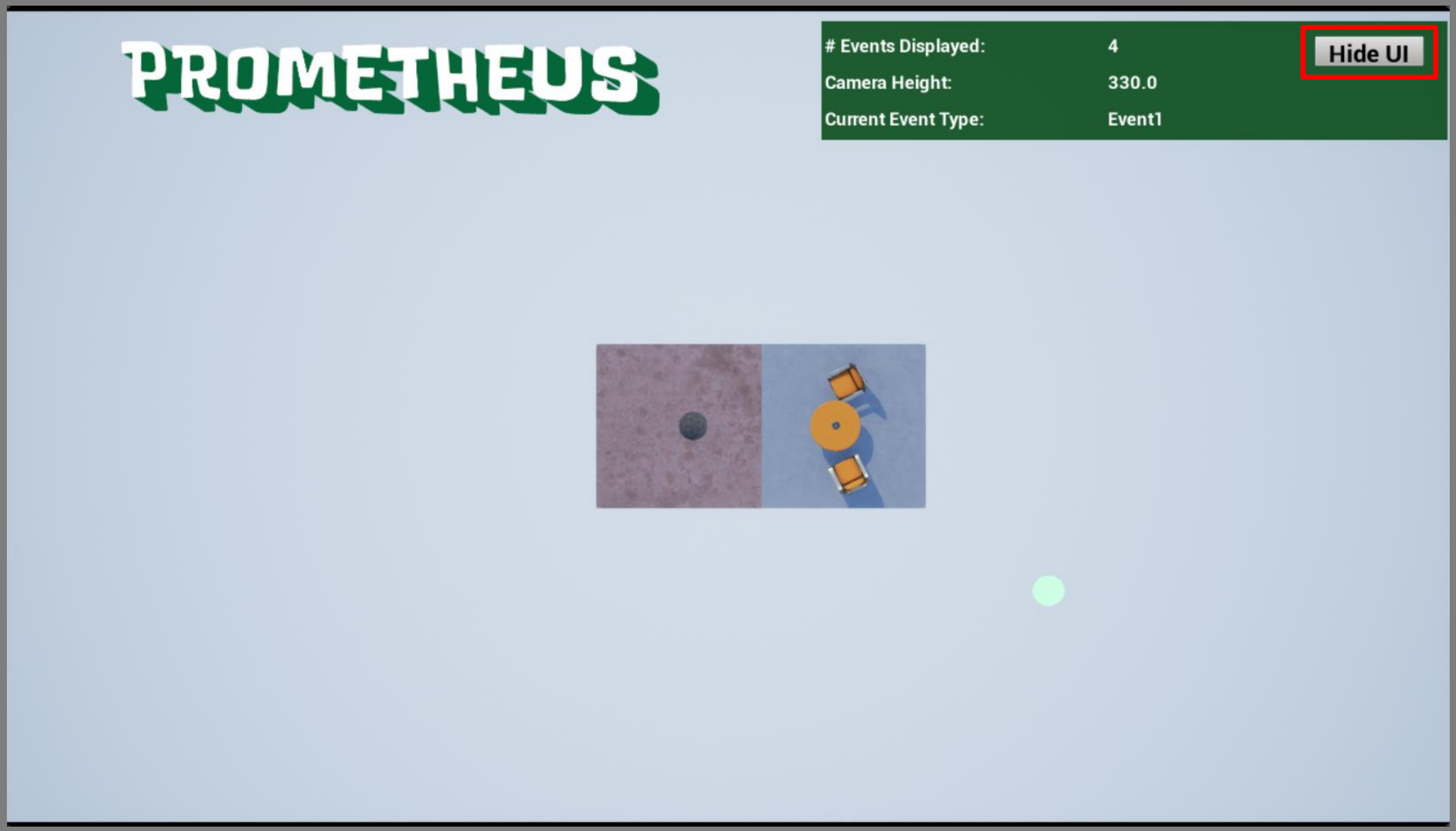
IF CONNECTED: GO STRAIGHT TO SELECT AN EVENT TYPE

- If Prometheus is connected to the database it auto populates Step 4
- The database needs dates and an event type, so don't forget the dates



PRESS ‘F’ FOR A BETTER VIEW

- Using ‘F’ or the “Hide UI” button gives you a better view of your level
- You can move the camera around with WASD to view specific areas



NOTES

- Prometheus was designed to be a part of a specific process, see additional resources for details
- Large datasets can cause long load times as Prometheus needs to loop through all the data
- The Camera can only point downwards, other angles are not supported
- Prometheus is designed to work with the Guildhall database to eliminate users having to edit json files
- If you do end up with a bunch of json files, you may need to combine them into one file yourself to make viewing easier
- There may be bugs that come up when moving to a new engine version

UPGRADING PROMETHEUS TO YOUR ENGINE VERSION

- To upgrade Prometheus to a new engine version, it simply needs to be put into the new Unreal project, and then compiled from Visual Studio
- Some Guildhall projects are not C++ projects (like middle TGP) so I solved this by creating a blank UE4 C++ project in the same engine version and compiling it there
- After compilation, the plugin can be moved to any project with the same engine version

ADDITIONAL RESOURCES

- Justin's thesis presentation on Prometheus and the process:

https://www.twitch.tv/videos/257685488