🎯 Purpose of This Setup Guide

This document outlines the required installation and configuration steps needed to successfully build and run Unity projects that include a custom tracking calibration implementation/plugin—specifically using Unity + SteamVR with the Null Driver for simulated VR environments.

🛠 Unity + SteamVR Setup Guide To Run The MR Projects

✅ Install Unity Hub

Download and install **Unity Hub** from the official website:
https://unity.com/download

🧱 Install Unity Versions

- 🔹 **Unity 2022.3.41f1**
  1. Open **Unity Hub**
  2. Add Unity version **2022.3.41f1**
  3. During installation, select the following modules:
     - **Microsoft Visual Studio Community 2022**
       - ✓ Desktop development with C++
       - ✓ Game development with Unity
     - **Download documentation**

- 🔹 **Unity 2020.2.4f1**
  1. Open **Unity Hub**
  2. Add Unity version **2020.2.4f1**
  3. During installation, select the following modules:
     - **Microsoft Visual Studio Community 2019**
       - ✓ Desktop development with C++
       - ✓ Game development with Unity
     - **Download documentation**

📦 Install Microsoft C++ Redistributables

Download and install the latest **C++ Redistributables** for **x64 architecture**:
https://learn.microsoft.com/en-us/cpp/windows/latest-supported-vc-redist?view=msvc-160

(Only the latest should be necessary, but in some particular cases may be needed a previous version)

🎮 Install Steam and SteamVR

1. Download and install **Steam**:
   http://store.steampowered.com/about/
2. Open Steam
   - Go to the **Library** tab
   - Install **SteamVR**
3. (Optional) Open SteamVR and **copy lab computer configuration settings**, if necessary.

⚙ SteamVR Configuration

Go to the path where Steam is installed, once there;

🔧 Modify steamvr.vrsettings

Path:
Steam\config\steamvr.vrsettings

Add or merge the following:

```
"driver_null": {
   "enable": true,
   "id": "Null Driver"
},
"steamvr": {
   "activateMultipleDrivers": true,
   "allowAsyncReprojection": false,
   "allowInterleavedReprojection": false,
   "allowSupersampleFiltering": false,
   "enableHomeApp": false,
   "forcedDriver": "null",
   "showAdvancedSettings": true,
   "showMirrorView": false
}
```

⚠ **Important:** Do **not** duplicate the "steamvr" entry. Just add/merge these parameters into the existing "steamvr" dictionary.

🔧 Modify default.vrsettings

Path:
Steam\steamapps\common\SteamVR\drivers\null\resources\settings\default.vrsettings

Replace contents with:

```
{
   "driver_null": {
      "enable": true,
      "serialNumber": "Null Serial Number",
      "modelNumber": "Null Model Number",
      "windowX": 0,
      "windowY": 0,
      "windowWidth": 1920,
      "windowHeight": 1080,
      "renderWidth": 1920,
      "renderHeight": 1060,
      "secondsFromVsyncToPhotons": 0.01111111,
      "displayFrequency": 60.0
   }
}
```

🔧 Post Setup – Environmental Parameter Adaptation

After completing the installation and configuration steps, you must adapt the project to match the **physical environment** and tracking setup. This is especially important for ensuring that the **tracking system** functions accurately.

📏 Environmental Parameter Adaptation

To adapt the tracking project to your local setup:

1. **Locate the camera script** in the Unity project where the tracking logic is applied.
2. Modify the following values in the script:
    - **Height of the Projection**
      This refers to the real-world height (in meters or your chosen unit) of the projected image
    - **Half the Physical Height of the Environment**
      Measure the total height of the physical environment and divide it by 2.

📐 Tracking Calibration Adaptation

For projects that **include the tracking calibration plugin**, an additional parameter is required:

- **Overlap Percentage (Height-Only)**
  You must provide a value representing the **percentage (out of 100%)** of the **map area** that overlaps **only in terms of height** between the projector space and the tracked physical environment.

⚠ **Disclaimer:** The variable names in the project may differ from those listed in this guide, but they should be intuitive and easy to identify based on their function.