**Unity Project Setup**

**Project Type: 3D Core (but can be 2D if you don't need physics or perspective).**

**Unity Version Recommended: 2021.3 LTS or newer**

**Folder Structure**

**CopyEdit**

**Assets/**

**├── Scripts/**

**│ └── SerialButtonController.cs**

**├── Scenes/**

**│ └── SerialTestScene.unity**

**Creating a UI Button in Unity**

1. In the **Hierarchy**, right-click → UI → Canvas.
2. Inside the Canvas, right-click → UI → Button (TextMeshPro or Legacy is fine).
3. Rename the Button (e.g., ActivateButton).
4. In the Inspector:
   * Scroll to **Button (Script)** → **On Click ()**
   * Click the **"+"** icon.
   * Drag the GameObject that holds SerialButtonController onto the field.
   * From the dropdown, select → SerialButtonController → SendPatternCommand().

**ESP32 Summary**

The ESP32 code is ready (see file: **Esp32 Serial Pattern**) and:

* Waits for the "ACTIVATE\_PATTERN" command.
* Triggers a synchronized PWM pulse across the **VTA fingers** and **fan**.
* Resets them to 0 after 3 seconds.

**What to Do Next**

1. **Upload ESP32 code** to your board.
2. **Build Unity scene** with a button wired to SendPatternCommand().
3. **Ensure Arduino Serial Monitor is closed** (so Unity can access COM7).
4. **Run Unity** → Click button → Observe VTAs and Fan initialization

NOTE\* review the documentation document!