

# Manual de instalación y uso del proyecto: Interfaz Unity con procesamiento por Python

---

Autores:

Daniel Castillo López A01737357

Emmanuel Lechuga Arreola A01736241

Paola Rojas Domínguez A01737136

Repositorio de Github: [https://github.com/PaolaRojas24/wuupi\\_argos/tree/main](https://github.com/PaolaRojas24/wuupi_argos/tree/main)

Video de la demo: <https://youtu.be/0E-hnjby2FA>

## Descripción general

Este proyecto permite que una interfaz hecha en **Unity** interactúe con dos scripts de Python (`manos_argos.py` y `piernas_argos.py`) que procesan video en tiempo real y detectan zonas tocadas por la mano o zonas relacionadas con piernas. Unity se encarga de gestionar el inicio de sesión, la selección del modo, la cámara a usar, y visualizar el video procesado.

---

## Requisitos del sistema

- Sistema operativo: Windows 10 u 11
  - Mínimo: Intel core i3 (décima generación o superior) o AMD Ryzen 5 (2da generación o superior).
  - Memoria RAM: 8GB DDR4 Suficiente para cargar sistema operativo, las librerías (mas la de media pipe) y manejar stream de video en memoria sin agotar recursos.
  - Memoria gráfica: Integrada (intel iris Xe Graphics, AMD Radeon Graphics).
  - Cámara: Webcam HD (720p x 480p) recomendado: webcam full HD (1080p a 30 fps o 60 fps ) o celular con Droidcam
  - Unity 2022 o superior
  - Python 3.9 o superior
  - Conexión a internet si se usará cámara por IP (DroidCam)
- 

## Librerías de Python usadas

- `opencv-python`: procesamiento de video
  - `mediapipe`: detección de manos/piernas
  - `numpy`: operaciones con arreglos
  - `openpyxl`: manipulación de archivos Excel
  - `socket`, `struct`, `time`, `sys`: librerías estándar para comunicación y control
- 

## Instalación

### 1. Instalar Python y dependencias

1. Asegúrate de tener Python instalado. Puedes descargarlo desde: <https://www.python.org/downloads/>
2. Descargar los archivos de la carpeta librerías: [https://github.com/PaolaRojas24/wuupi\\_argos/tree/main/Librerias](https://github.com/PaolaRojas24/wuupi_argos/tree/main/Librerias)
3. Ejecuta el script `setup_env.bat` para instalar las librerías necesarias:
  - Doble clic en `setup_env.bat`
  - Espera a que finalice la instalación

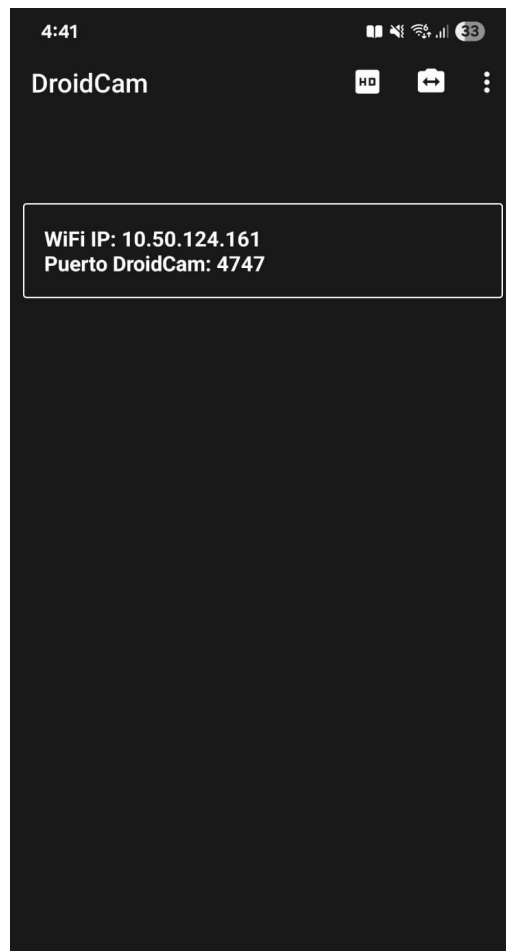
### 2. Abrir el proyecto en Unity

1. Descarga el proyecto de unity de la carpeta Wuupi (se sugiere utilizar [DownGit](https://github.com/PaolaRojas24/wuupi_argos/tree/main/Wuupi)): [https://github.com/PaolaRojas24/wuupi\\_argos/tree/main/Wuupi](https://github.com/PaolaRojas24/wuupi_argos/tree/main/Wuupi)
  2. Abre Unity Hub
  3. Selecciona "Add" y carga la carpeta del proyecto
  4. Abre la escena inicial: `LoginScene.unity`
- 

## Uso del sistema

### 1. LoginScene

- El usuario debe ingresar:
  - Nombre de usuario
  - Contraseña
  - Tipo de procesamiento (`Manos` o `Piernas`)
  - Cámara a usar (`Integrada` o `DroidCam`)
  - Si escoge `DroidCam`, debe ingresar una IP (ej. `10.50.124.161`)



- Al presionar el botón "Iniciar", se guarda la información y se carga la **CamaraScene**.

## 2. CamaraScene

- Unity lanza el script correspondiente de Python (**manos\_argos.py** o **piernas\_argos.py**)
- El script:
  - Procesa el video
  - Detecta zonas de interacción (con MediaPipe)
  - Envía imágenes procesadas por socket a Unity
- Unity recibe y muestra el video en tiempo real en un **RawImage**.

---

## Comunicación Unity ↔ Python

- Unity inicia el script Python con **System.Diagnostics.Process**.
  - El script Python actúa como servidor TCP (puerto **8000**) y envía frames codificados.
  - Unity se conecta como cliente y visualiza los frames.
-

## manos\_argos.py

Este script detecta **manos humanas** utilizando la librería **MediaPipe** y determina qué zonas de la pantalla están siendo tocadas por la mano. Funciona de la siguiente manera:

- Captura video desde una cámara (integrada o IP con DroidCam).
- Usa MediaPipe Tasks para detectar manos y extraer landmarks (puntos clave).
- Divide la imagen en zonas visuales (zona segura y cuatro zonas representando cada uno de los botones).
- Determina qué zona está siendo activada por la mano y la envía en tiempo real a Unity mediante sockets.
- También se visualiza la imagen procesada con anotaciones en tiempo real dentro de Unity.

Este script es útil para sistemas de interacción sin contacto, interfaces gestuales o control de elementos con movimiento de la mano.

---

## piernas\_argos.py

Este script está diseñado para el seguimiento de **piernas u objetos similares**. Su comportamiento es el siguiente:

- Captura video en tiempo real desde la cámara seleccionada.
- Utiliza MediaPipe Pose para detectar posiciones clave de las piernas.
- Calcula en qué zonas de la pantalla se encuentran los elementos detectados.
- Envía la información a Unity, que puede utilizarla para visualización, interacción o toma de decisiones.

Este script puede utilizarse para interfaces controladas por movimientos de pierna, análisis de postura o sistemas de detección de presencia en ciertas regiones del espacio visual.

---

## registro\_usuarios.py

- Este script se encarga de registrar los usuarios que inician sesión.
  - Almacena la información en un archivo Excel (**usuarios.xlsx**) con las columnas:
    - Usuario
    - Contraseña
-

## Solución de problemas

Problema	Solución
Unity no muestra video	Verifica que el script Python esté corriendo correctamente
IP de DroidCam no funciona	Asegúrate de estar conectado a la misma red que el teléfono
Error de librerías Python	Ejecuta <code>setup_env.bat</code> para reinstalar dependencias
El .py no se ejecuta desde Unity	Asegúrate de que Python esté en el PATH o modifica <code>PythonLauncher.cs</code> para usar la ruta completa
¿Otra duda?	Escriba un correo a: <a href="mailto:A01737136@tec.mx">A01737136@tec.mx</a>