**Task-12 (DEC 2024 Batches)**

**Task description: Campus Monitoring System**

* Sending a live image from an Android App (captured using an Android mobile phone camera) to YOLO V11 (running on a laptop) and getting inference (object detection- campus monitoring- dust bin status)- Both laptop and Android phones are connected to different networks and they are communicating **using NGROK**

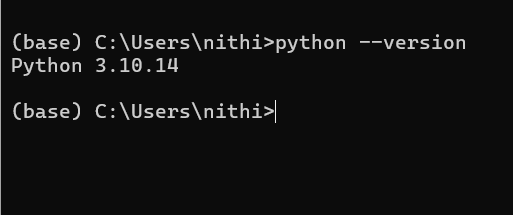
**Files used**

* Server.zip (Python-based application @ laptop)
* client.zip (Android app @ Android mobile phone)

**Procedure**

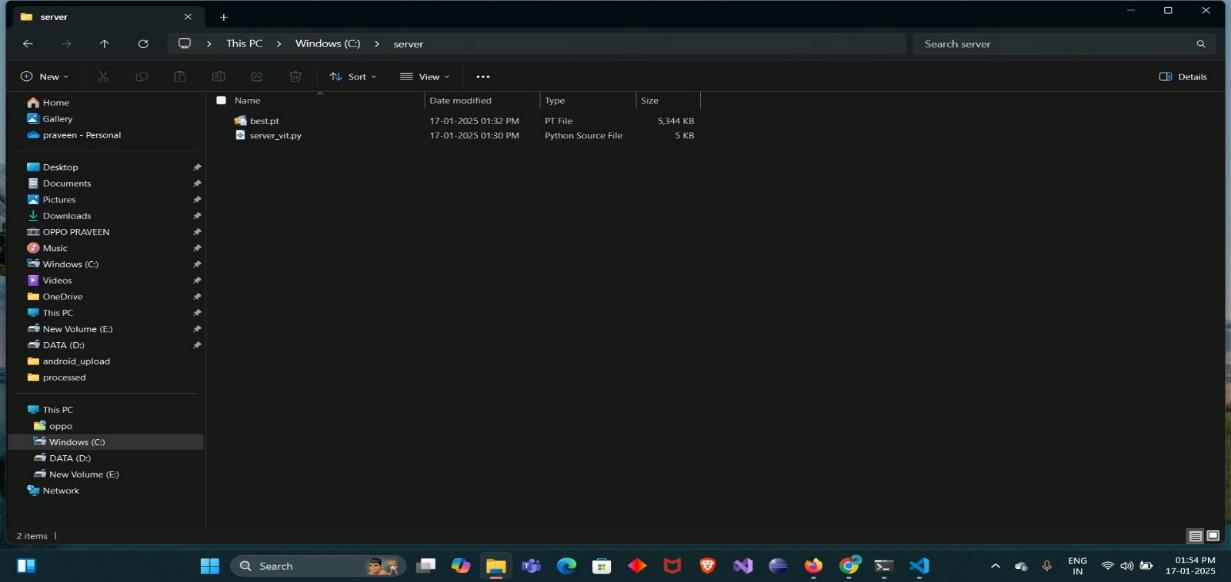
**Step 1: Server side (A laptop):**

* **Install Python 3.10.14** on the the laptop
* Open a terminal and check the python version as follows:

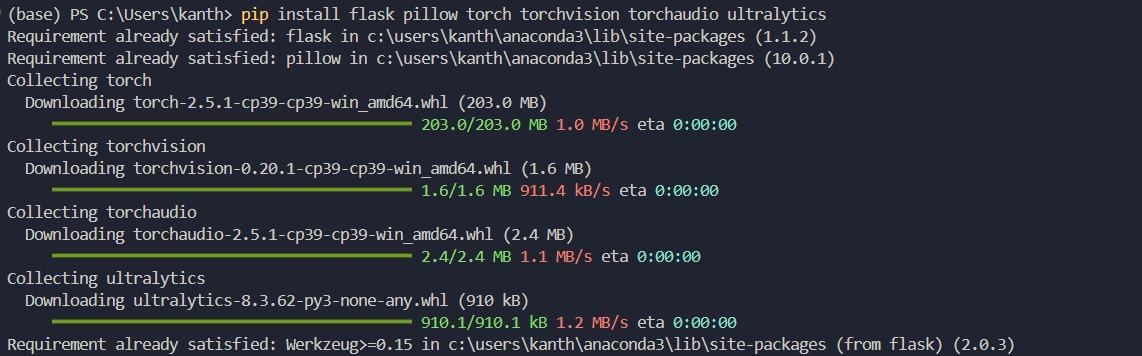


**Step 2: Extract the server.zip into the C drive (on the laptop). This will store the following files in the C:\server\:**

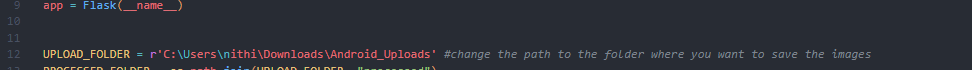
* **server\_vit.py**
* **best.pt**



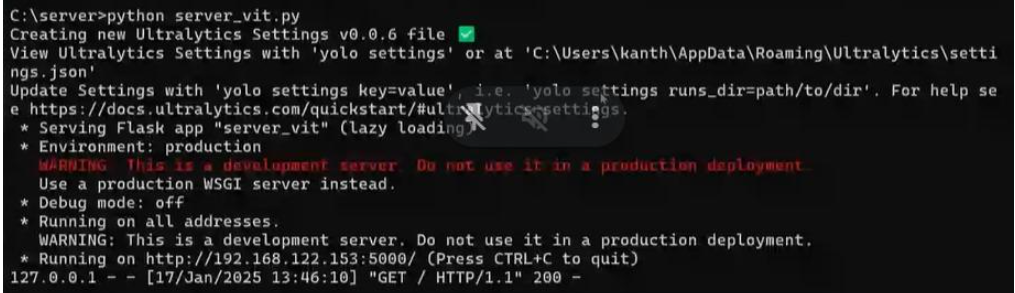
**Step 3:** Give the command **pip install flask pillow torch torchvision torchaudio ultralytics** to download the libraries in the laptop as follows:



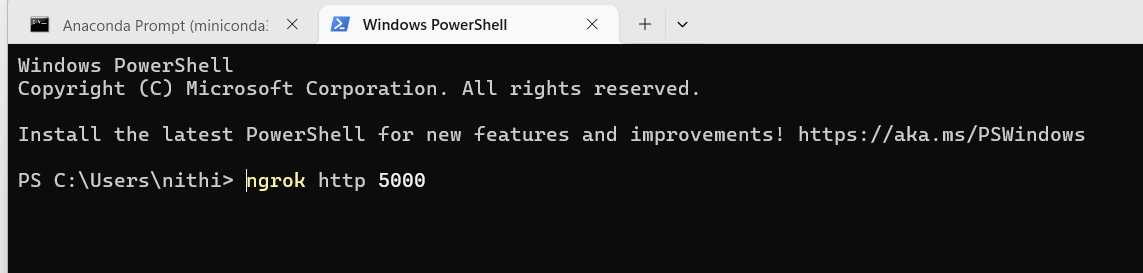
**Step 4:** In server\_vit.py file, give the suitable folder path to store the images uploaded by the Android App as follows:

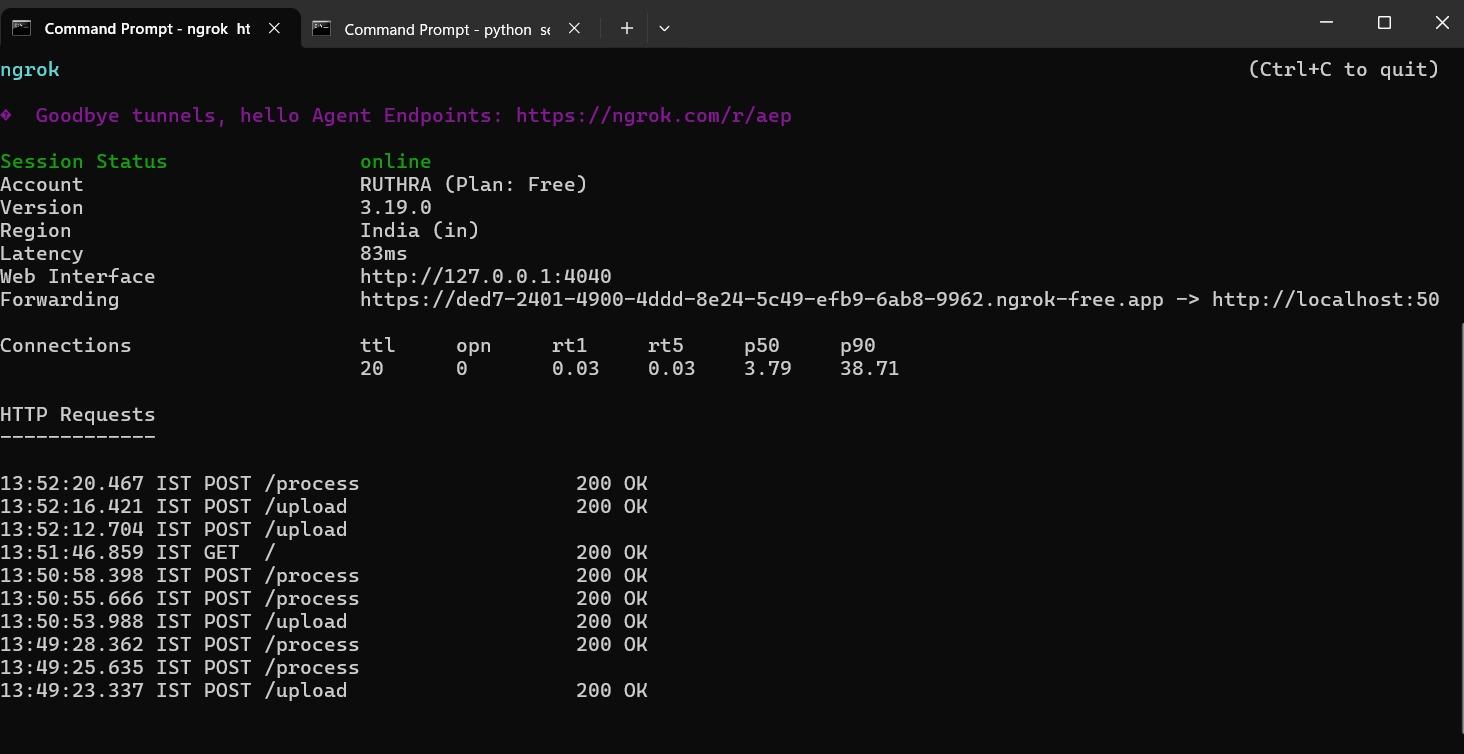


**Step 5:** Run the **python server\_vit.py as follows:**



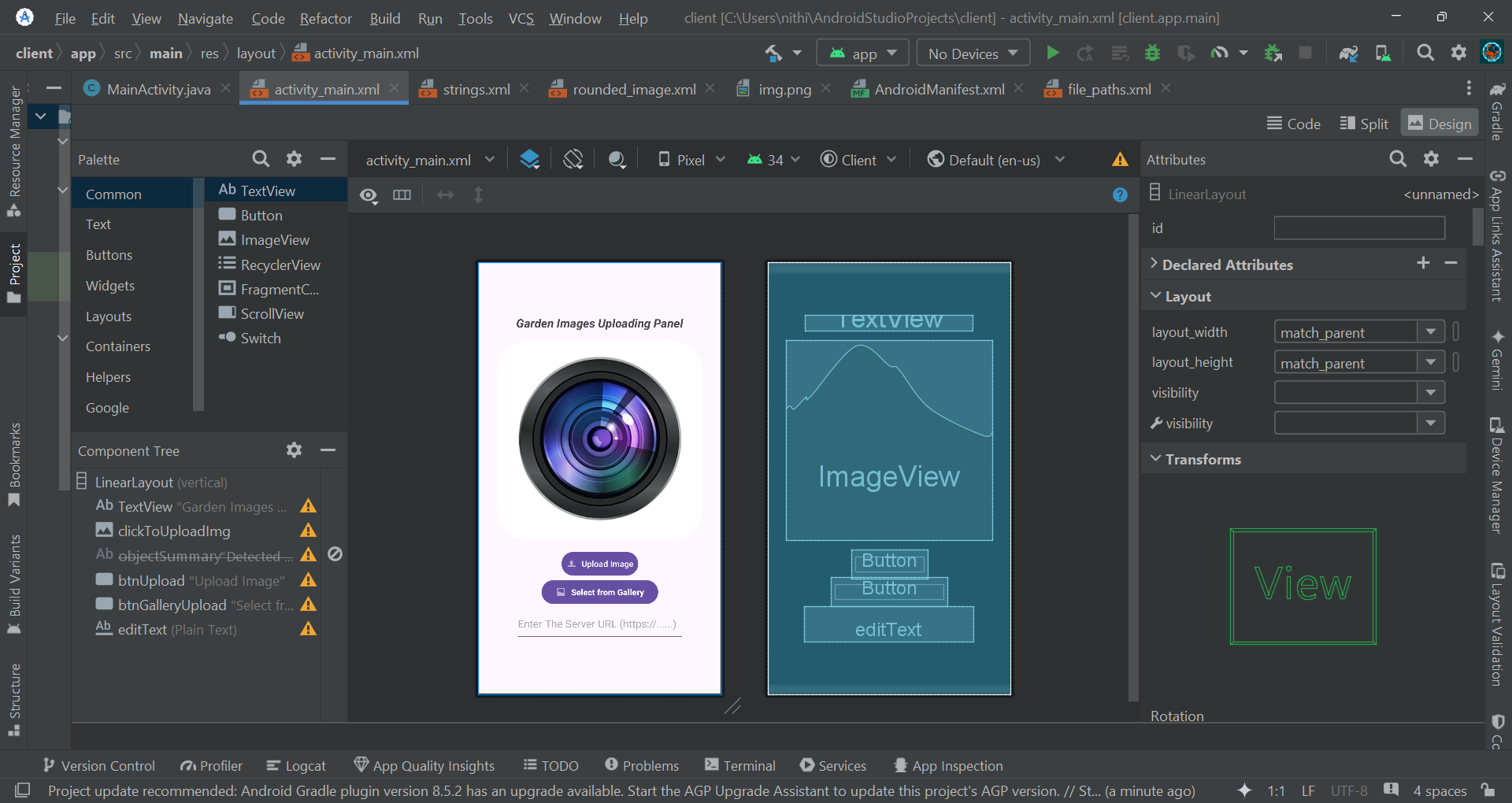
**Step 6:** Exposing the Local Server (running on the laptop) using ngrok using the command **ngrok http 5000** as follows:





* Copy/preserve the ngrok link for connecting the Android app to the Python server (laptop).

**Step 7:** Now extract the **client.zip** and load the Android app into an **android studio**. Then create the .apk file and send it to the required mobile phone (via whatsapp).



**Step 8:** Now run the Android app in the mobile phone (client side).

**Step 9:** Enter the ngrok url (Step 6) to the application as follows:



**Step 10 :** Select any image from the mobile phone (camera / gallery ).

* For accessing the camera:
  + single tap on camera lens image (at the home page)



**Step 11:** After uploading the image, we will inference as follows:

