

## UAS KECERDASAN TIRUAN

---

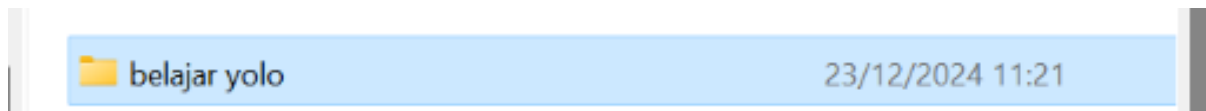
Nama: Diwa Mulki

Nim: 230741112

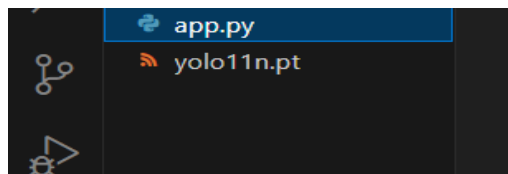
Prodi: Ilmu Komputer

Dosen Pengampu: Zikri Wahyuzi

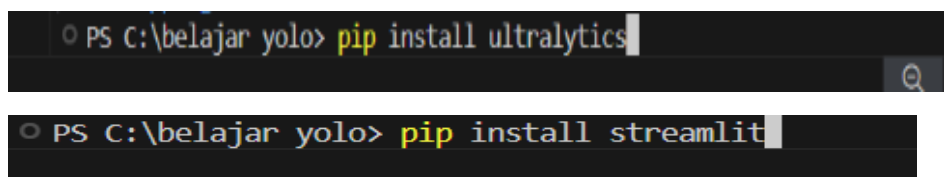
Yang pertama kali kalian lakukan adalah membuat folder , dengan nama belajar yolo



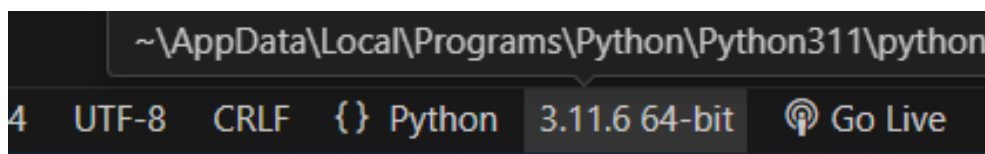
Setelah itu kalian mendownload YOLO11, dan menambahkannya di dalam folder belajar yolo yang telah anda buat sebelumnya. Kemudian masuk ke visual studio code dan import folder tersebut , dan buat file baru dengan format app.py



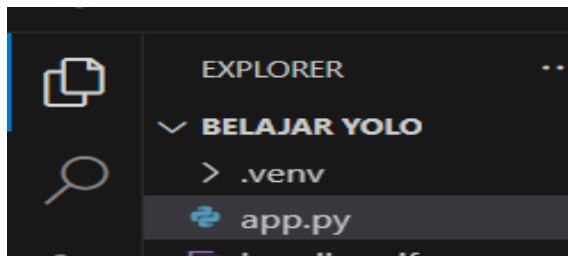
Selanjutnya adalah menginstall library yang dibutuhkan sebagai berikut:



Pilih interpreter dengan memilih yang global atau recommended



Jika kalian menggunakan yang global akan muncul .venv



Kemudian kalian lanjutkan dengan menambahkan code dan saya juga telah menjelaskan fungsi fungsi dari code di atas dengan tanda (#) sebagai berikut:

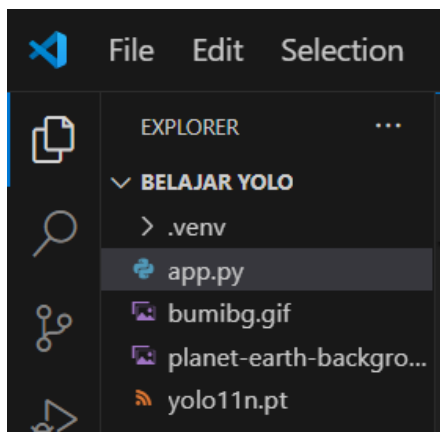
```
1 #import library yang dibutuhkan
2 from ultralytics import YOLO
3 import cv2
4 import streamlit as st
5 from PIL import Image
6 import numpy as np
7 from collections import Counter
8 import base64
9
10 # ini untuk Muat model YOLO
11 @st.cache_resource
12 def load_model(model_path):
13     return YOLO(model_path)
14
15 # ini code untuk Memproses dan menampilkan hasil deteksi
16 def display_results(image, results):
17     boxes = results.boxes.xyxy.cpu().numpy() # [x1, y1, x2, y2]
18     scores = results.boxes.conf.cpu().numpy() # Skor Tingkat keyakinannya
19     labels = results.boxes.cls.cpu().numpy() # Indeks kelas
20     names = results.names # nama kelas
21
22     detected_objects = []
23
24     for i in range(len(boxes)):
25         if scores[i] > 0.5: # Ambang Batas kepercayaan
26             x1, y1, x2, y2 = boxes[i].astype(int)
27             label = names[int(labels[i])]
28             score = scores[i]
29             detected_objects.append(label)
30             cv2.rectangle(image, (x1, y1), (x2, y2), (0, 255, 0), 2)
31             cv2.putText(image, f"{label}: {score:.2f}", (x1, y1 - 10), cv2.FONT_HERSHEY_SIMPLEX, 0.5, (0, 255, 0), 2)
32
33     return image, detected_objects
34
35 # Fungsi untuk menambahkan latar belakang
36 def set_background(image_path):
37     with open(image_path, "rb") as file:
38         base64_image = base64.b64encode(file.read()).decode()
39
40     css = """
41     <style>
42     .stApp {
43         background-image: url("data:image;base64,{base64_image}");
44         background-size: cover;
45         background-repeat: no-repeat;
46         background-attachment: fixed;
47     }
48     </style>
49     """
50     st.markdown(css, unsafe_allow_html=True)
51
52 # Fungsi untuk menampilkan GIF di sidebar
53 def display_animation(gif_path):
54     with open(gif_path, "rb") as file:
55         if data = file.read()
```

```

app.py x planet-earth-background.jpg
app.py > main
51 # Fungsi untuk menampilkan GIF di sidebar
52 def display_animation(gif_path):
53     with open(gif_path, "rb") as file:
54         gif_data = file.read()
55         base64_gif = base64.b64encode(gif_data).decode()
56     st.sidebar.markdown(
57         """
58         <div style="text-align: center;">
59             
60         </div>
61         """,
62         unsafe_allow_html=True,
63     )
64
65 # ini untuk App Streamlit Utama
66 def main():
67     # Tampilkan animasi di bawah sidebar
68     display_animation("bumibg.gif") # Path ke file GIF
69
70     # ini code menambahkan image(foto)
71     set_background(["planet-earth-background.jpg"]) # URL ke gambar latar belakang
72
73     st.title(" DETEKSI OBJEK DIWA MULKI ")
74     st.sidebar.title("MENU")
75
76     model_path = "yolo11n.pt" # Jalur menyambungkan ke model YOLO kite
77     model = load_model(model_path)
78
79     # Buat tombol sakelar untuk mode gaming
80     if "detection_active" not in st.session_state:
81         st.session_state.detection_active = False
82
83     if st.sidebar.button(" MULAI DETEKSI " if not st.session_state.detection_active else "✗ STOP DETEKSI ✗"):
84         st.session_state.detection_active = not st.session_state.detection_active
85
86     # Buka pengambilan video jika mode gaming aktif
87     if st.session_state.detection_active:
88         st.sidebar.success("MENYALAH CUVI: ● Active")
89         cap = cv2.VideoCapture(0)
90         st_frame = st.empty() # Placeholder untuk bingkai video
91         st_detection_info = st.empty() # Placeholder untuk informasi deteksi
92
93         while st.session_state.detection_active:
94             ret, frame = cap.read()
95             if not ret:
96                 st.warning("Failed to capture image.")
97                 break
98
99             # Jalankan deteksi YOLO
100             frame = cv2.cvtColor(frame, cv2.COLOR_BGR2RGB) # Konversikan ke RGB untuk tampilan
101             results = model.predict(frame, imgsz=640) # Lakukan deteksi
102
103             # ini Gambarkan hasil dan kumpulkan objek yang terdeteksi

```

Pada set\_bacground dan display\_animation anda bisa menggunakan foto atau gambar yang sudah anda download, dan pada display\_animation kalian bisa menambahkan video dengan format (gift) sebagai berikut:



```

67 # Tampilkan animesi di bawah sidebar
68 display_animation("bumibg.gif") # Path ke file GIF
69
70 # ini code menambahkan image(foto)
71 set_background(["planet-earth-background.jpg"]) # URL ke gambar latar belakang
72

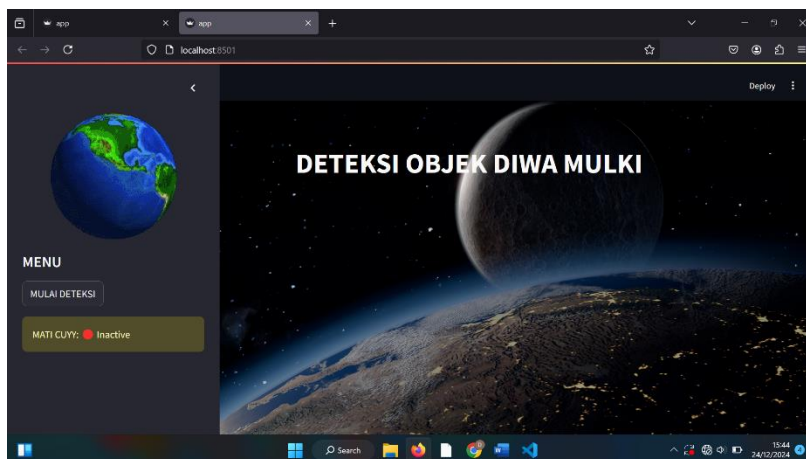
```

```
app.py > main
66 def main():
102
103     # ini Gambarkan hasil dan kumpulan objek yang terdeteksi
104     frame, detected_objects = display_results(frame, results[0])
105
106     # ini untuk Tampilkan video
107     st_frame.image(frame, channels="RGB", use_column_width=True)
108
109     # ini untuk menampilkan informasi deteksi
110     if detected_objects:
111         object_counts = Counter(detected_objects)
112         detection_info = "\n".join([f"{obj}: {count}" for obj, count in object_counts.items()])
113     else:
114         detection_info = "No objects detected."
115
116     st_detection_info.text(detection_info) # Update detection info text
117
118     # Keluar dari loop jika mode gaming dinonaktifkan
119     if not st.session_state.detection_active:
120         break
121
122     cap.release()
123 else:
124     st.sidebar.warning("MATI CUY: 🚫 Inactive")
125
126 if __name__ == "__main__":
127     main()
128
```

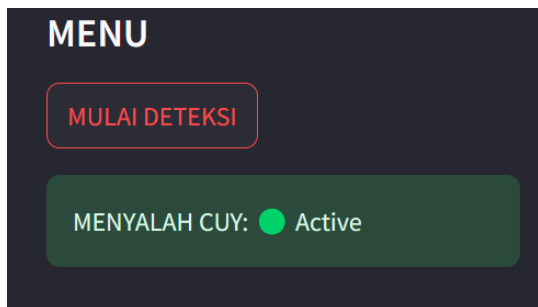
Jika sudah menerapkan code di atas dilanjutkan dengan menjalankannya di terminal sebagai berikut:

```
PS C:\belajar yolo> streamlit run app.py
>>
```

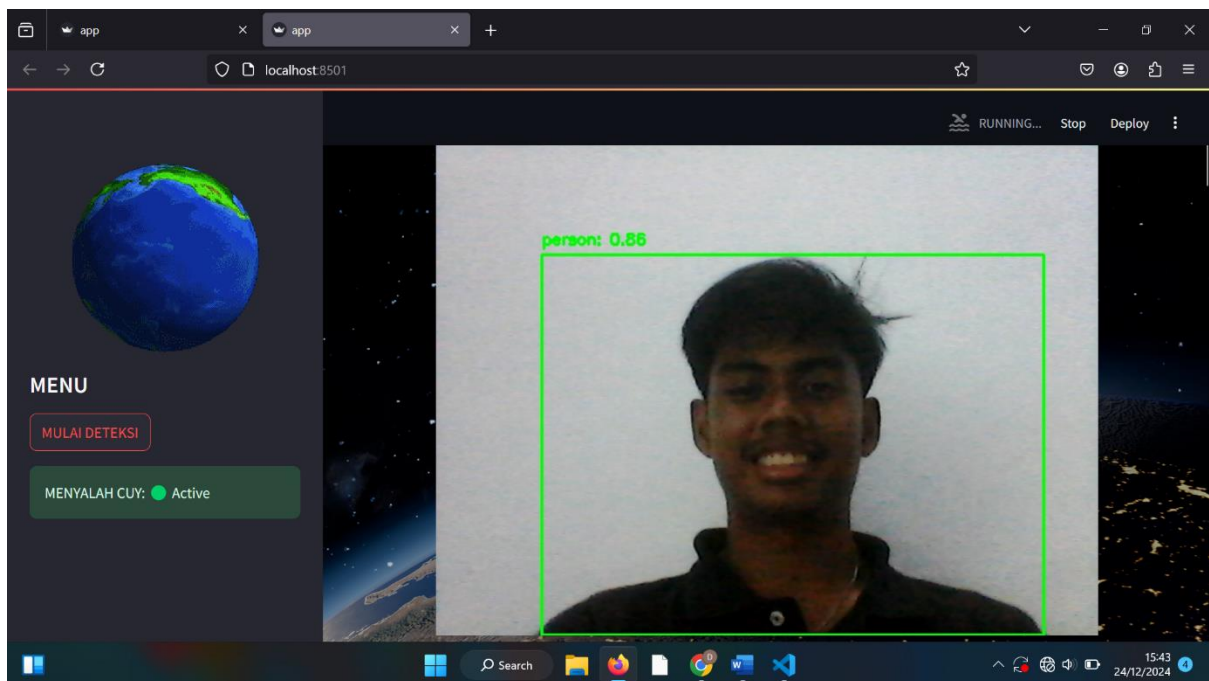
Kemudian kalian di suruh untuk memasukkan email kalian untuk mengakses streamlit Dan akan di arahkan ke browser yang kalian pilih sebagai berikut:



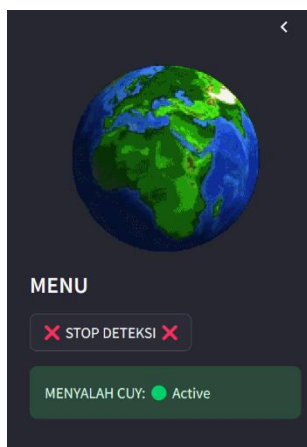
Beginilah tampilan awal masuk ke websitenya teman teman, kemudian untuk mencobanya langkah pertama adalah dengan menyalakan deteksinya, dengan menekan tombol (MULAI DETEKSI) apabila sudah seperti ini tandanya sudah aktif:



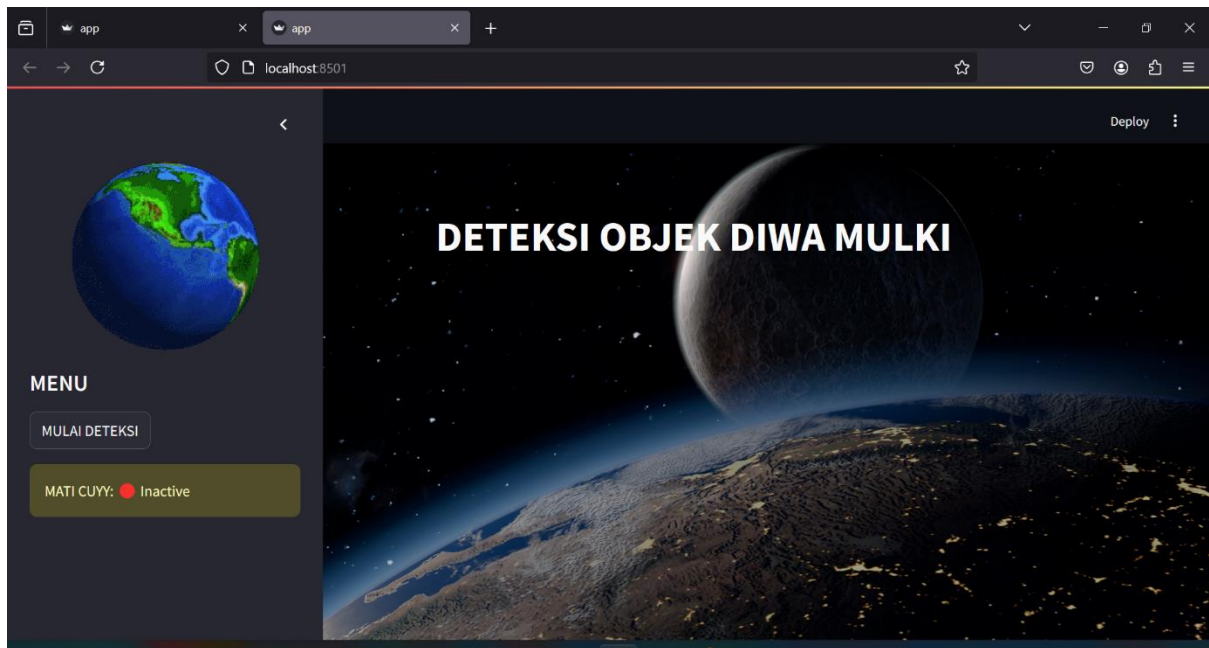
(MENYALAH CUY) dengan tanda seperti ini, dan otomatis camera anda akan otomatis menyala dan YOLO akan otomatis mendeteksi secara real time seperti gambar di bawah:



Dan apabila anda ingin mematikan pendeteksinye anda cukup tekan tombol (STOP DETEKSI)



Kemudian akan masuk ke tampilan awal seperti berikut:



Demikian tutorial singkat dari membuat deteksi YOLO11 ,semoga membantu teman teman dalam belajar dan memahami teknologi sekarang dan masa mendatang, dan menjadi acuan teman teman agar semangat dalam belajar dan menggapai impian teman teman , saya Diwa Mulki pamit undur diri salam sukses.

**TERIMA KASIH**