JOBSHEET

WORKSHOP SMART CLASSROOM

FACE RECOGNITION DAN FINGERPRINT

(COMPUTER VISION)

SMKN 6 Malang

Disusun Oleh:

PT. KASTARA DIGITAL TECHNOLOGY

SEPTEMBER 2023

DAFTAR ISI

1. PENDAHULUAN

Gambar 1. 1 Computer Vision

Dalam era digital yang berkembang pesat, Computer Vision (CV) telah menjadi konsep revolusioner yang mengubah paradigma interaksi manusia dengan teknologi dan lingkungan sekitar. CV memungkinkan komputer untuk "melihat" dan "memahami" dunia visual seperti manusia, membuka pintu untuk inovasi yang mendalam dan beragam dalam berbagai bidang.kenyamanan.

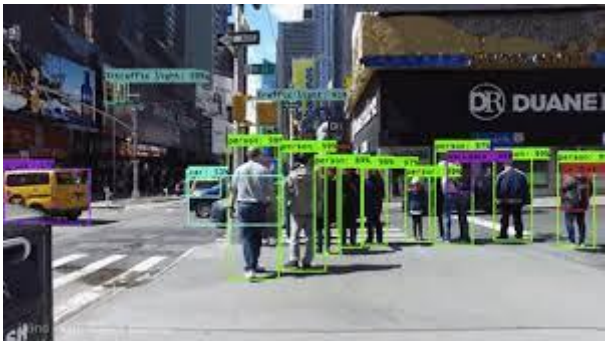
1.1 TEORI DASAR

Sebelum kita memahami Computer Vision, penting untuk memahami bahwa computer vision adalah cabang dari ilmu komputer yang berkaitan dengan pemahaman, analisis, dan interpretasi informasi visual dari dunia nyata, yang biasanya diperoleh melalui kamera atau sensor visual lainnya. Tujuannya adalah untuk mengizinkan komputer untuk memproses, memahami, dan mengambil keputusan berdasarkan data visual. Berikut beberapa teori dasar dalam computer vision:

- **Pengolahan Citra**: Ini adalah langkah awal dalam computer vision di mana citra visual dari dunia nyata diterjemahkan ke dalam format yang dapat diolah oleh komputer. Ini melibatkan filter, segmentasi, ekstraksi fitur, dan transformasi citra.
- **Segmentasi**: Segmentasi mencoba untuk memisahkan objek-objek yang berbeda dalam citra, yang berarti mengidentifikasi batas antara objek-objek tersebut. Ini memungkinkan komputer untuk mengenali objek secara terpisah.

- **Pelacakan Objek**: Pelacakan objek adalah kemampuan untuk melacak pergerakan objek dalam rangkaian citra atau dalam video. Ini berguna dalam aplikasi seperti pengawasan video, kendaraan otonom, dan augmented reality.

1.2 PENGERTIAN COMPUTER VISION



Gambar 1. 2 Computer Vision

Computer Vision (CV) adalah cabang dari kecerdasan buatan yang berkaitan dengan kemampuan komputer untuk memahami, menganalisis, dan menafsirkan gambar dan video yang dihasilkan oleh dunia nyata. Secara singkat, computer vision memungkinkan komputer untuk "melihat" dan memahami dunia seperti yang dilakukan manusia, dengan tujuan seperti pengenalan objek, deteksi pola, pelacakan objek, dan banyak lagi, semuanya berdasarkan data visual yang diterima oleh kamera atau sensor optik. Ini merupakan teknologi kunci dalam berbagai aplikasi, termasuk kendaraan otonom, pengenalan wajah, pengolahan medis, dan banyak lagi.

1.3 PERKEMBANGAN COMPUTER VISION PADA ERA SEKARANG

Dalam era digital yang terus berkembang, perkembangan Computer Vision (CV) telah menjadi lebih menarik daripada sebelumnya. Saat kita terus berinteraksi dengan teknologi dan lingkungan sekitar kita, CV menjadi semakin terintegrasi dalam kehidupan sehari-hari. Penggunaan pengenalan objek yang dapat dihubungkan dengan aplikasi ponsel pintar, dan dihubungkan ke kendaraan otonom, semuanya adalah contoh konkret dari bagaimana Computer Vision telah merubah cara kita berinteraksi dengan dunia di sekitar kita. Berikut merupakan contoh perkembangan Computer Vision pada era sekarang:

1. Pengenalan Objek (You Only Look Once).
2. Pengenalan Wajah (Haarscade).
3. Mobil Otonom.
4. Klasifikasi Medis.
5. Pencocokan Gambar (Fingerprint).

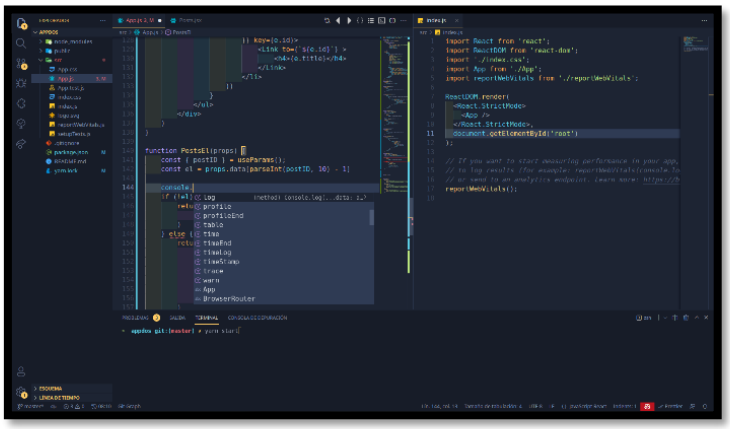
2. PERSIAPAN PRAKTIKUM

2.1.1 HARDWARE

No.	Item.	Type.	Picture.
1	Webcam Laptop	Menyesuaikan	Menyesuaikan
2	Laptop	Windows/Mac	Menyesuaikan

2.2 SOFTWARE

2.2.1 VISUAL STUDIO CODE



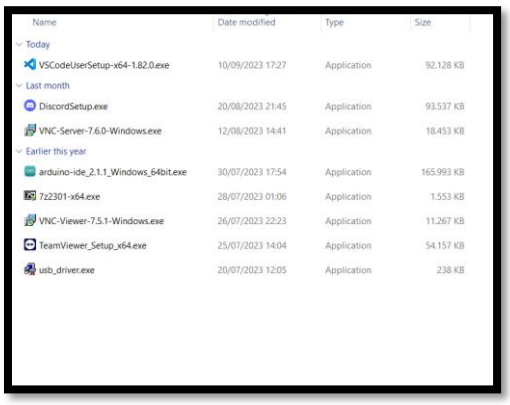
Gambar 2. 1 Visual Studio Code

Visual Studio Code (VS Code) adalah sebuah editor kode sumber terintegrasi yang sangat populer yang dikembangkan oleh Microsoft. Ini adalah perangkat lunak yang digunakan oleh pengembang perangkat lunak untuk menulis, mengedit, dan mengelola kode dalam berbagai bahasa pemrograman. VS Code memiliki banyak fitur berguna, seperti penyorotan sintaksis, saran otomatis, pemecah kode, dan integrasi dengan alat pengembangan yang berbeda.

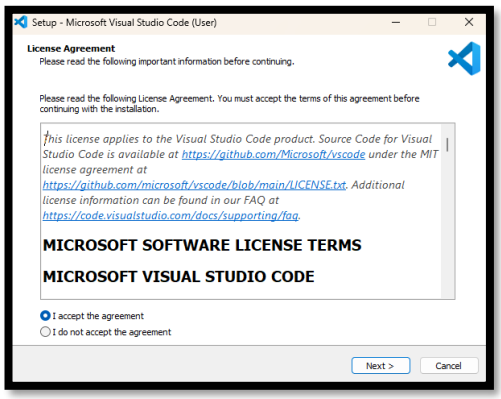
3. LANGKAH-LANGKAH PRAKTIKUM

3.1 INSTALASI VISUAL STUDIO CODE

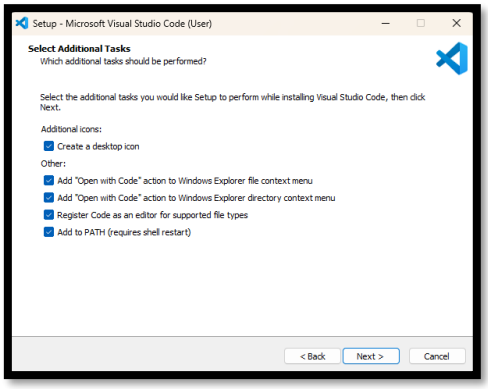
1. Download Visual Studio Code melalui https://code.visualstudio.com/download
2. Install Visual Studio Code dengan cara klik dua kali pada Installer nya



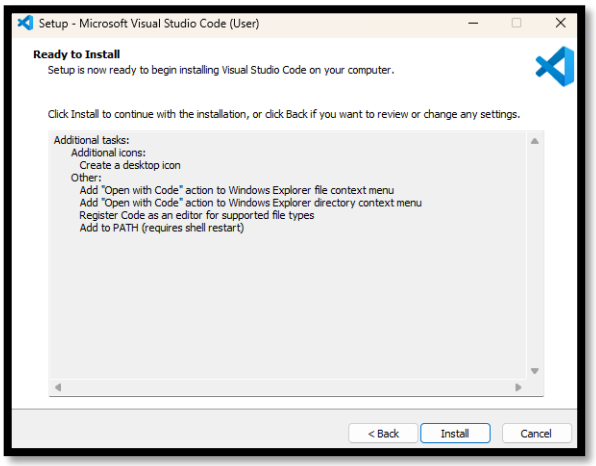
3. Pilih **I Accept the agreement**



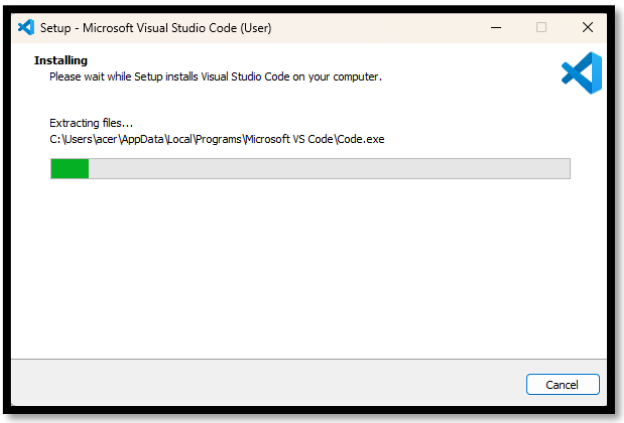
4. Centang semua pilihan dan pilih **Next**



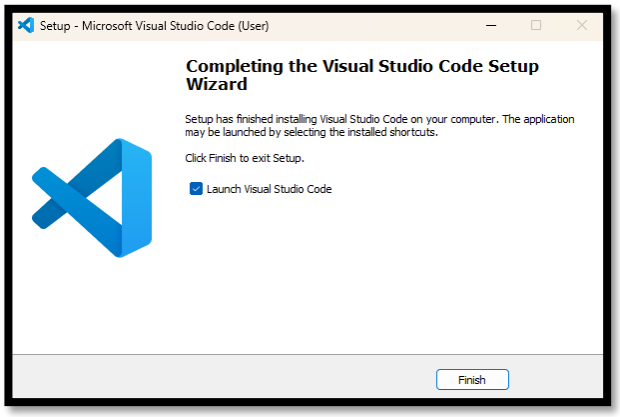
5. Klik **Install**



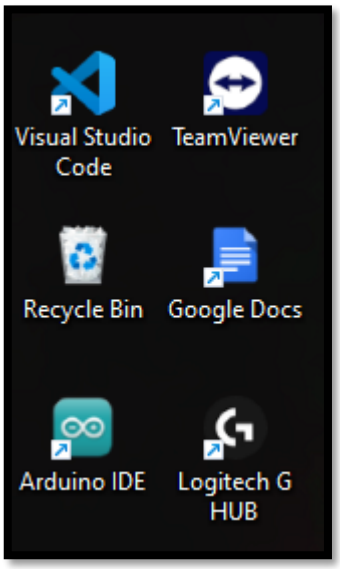
6. Tunggu hingga proses instalasi selesai



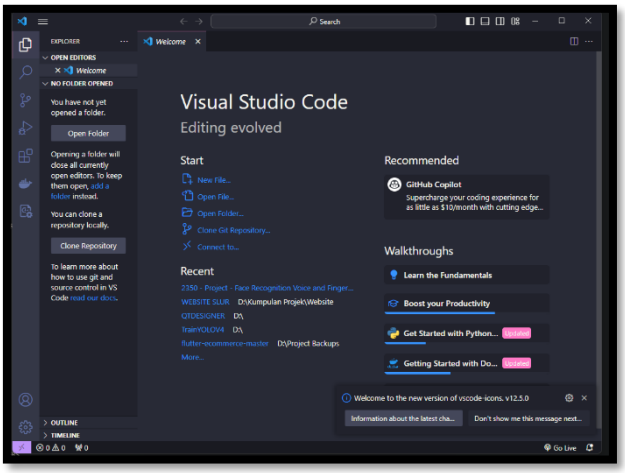
7. Proses instalasi selesai dan pilih **Finish**



8. Visual Studio Code akan muncul pada tampilan desktop

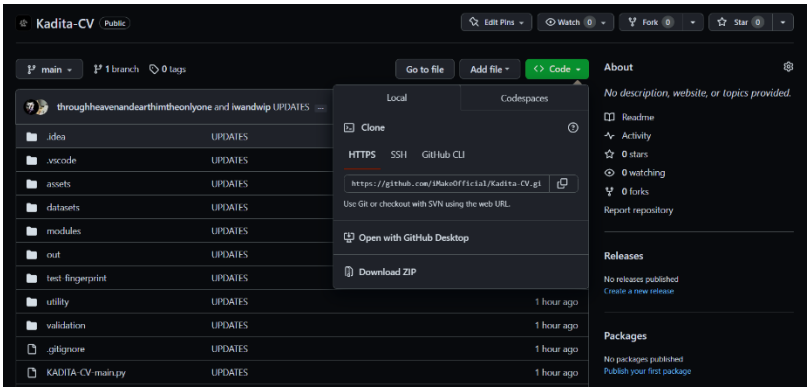


9. Buka Visual Studio Code

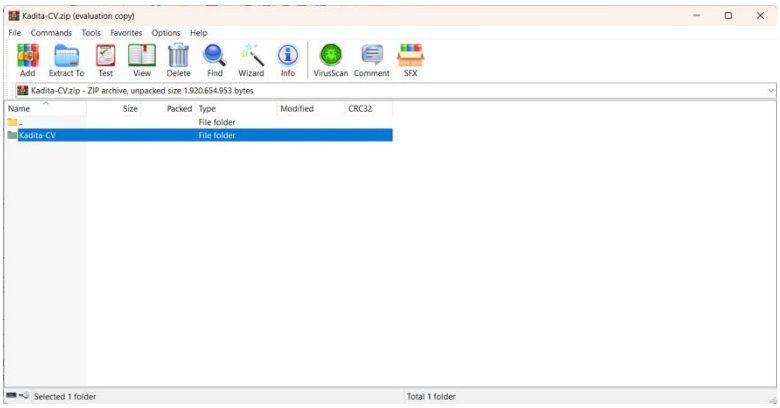


3.2 INSTALASI FRAMEWORK KADITA-CV PADA VISUAL STUDIO CODE

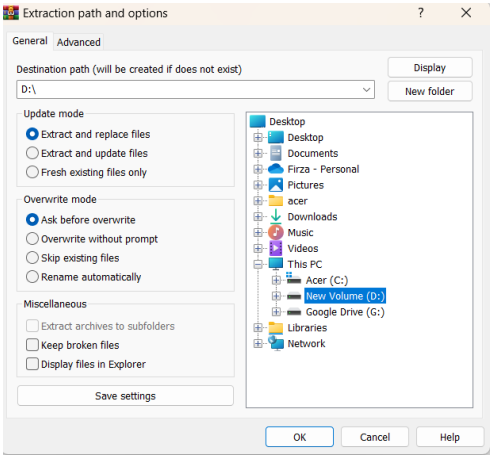
1. Download ZIP file Kadita-CV di Github pada link di bawah:
   https://github.com/iMakeOfficial/240-KDT-WORKSHOP-CV



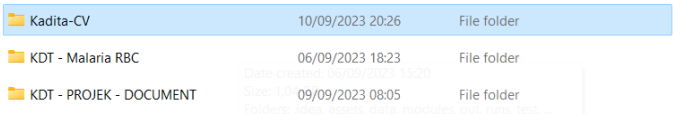
2. Buka file .zip menggunakan RAR extractor (winrar/7zip)



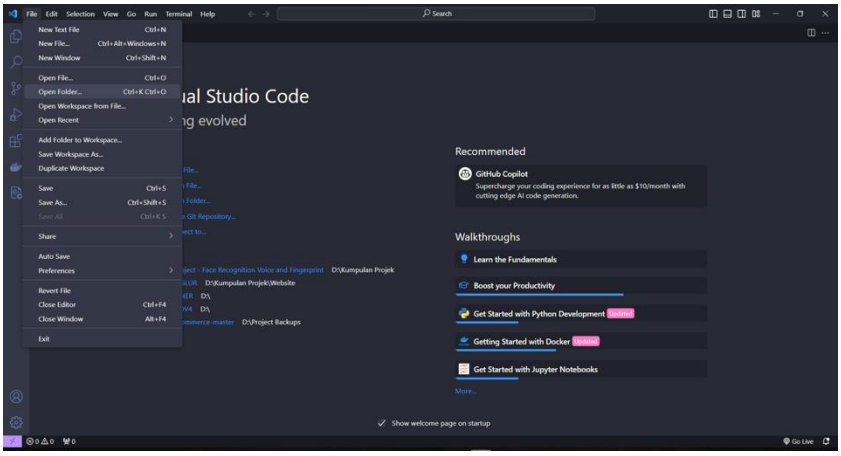
3. Ekstrak ke folder yang anda inginkan



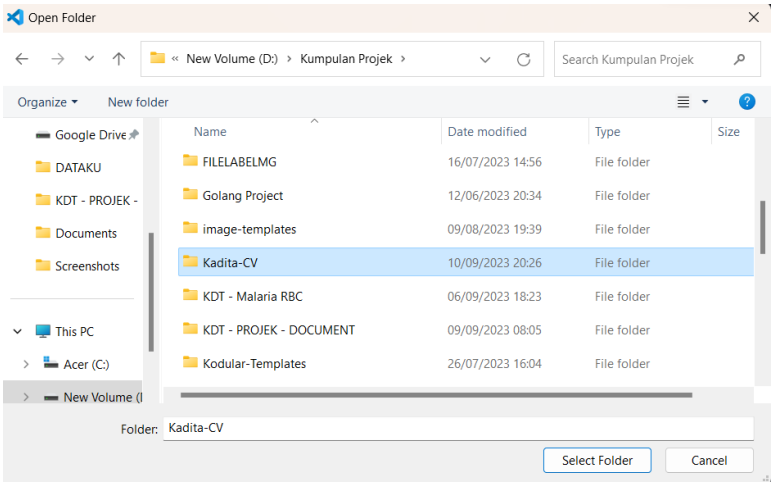
4. Setelah di ekstrak lalu muncul file seperti ini **:**



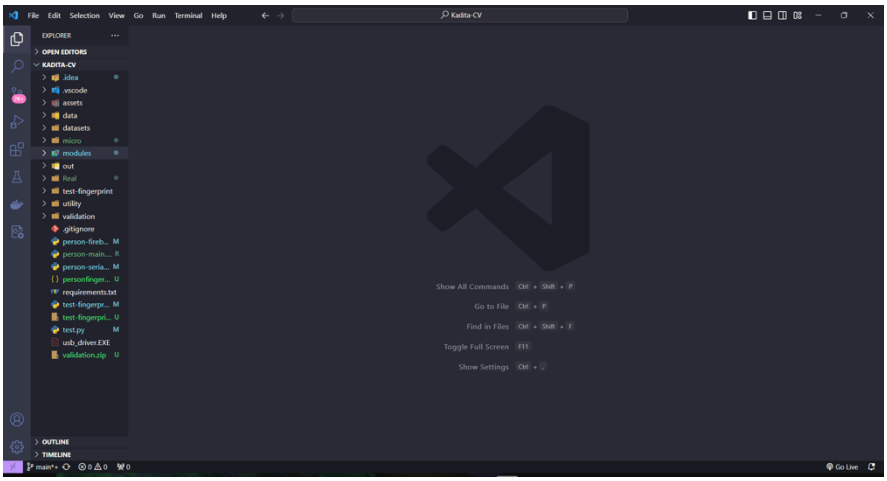
5. Selanjutnya buka Visual Studio Code lalu pilih menu **Open Folder**



6. **Select Folder** Kadita-CV

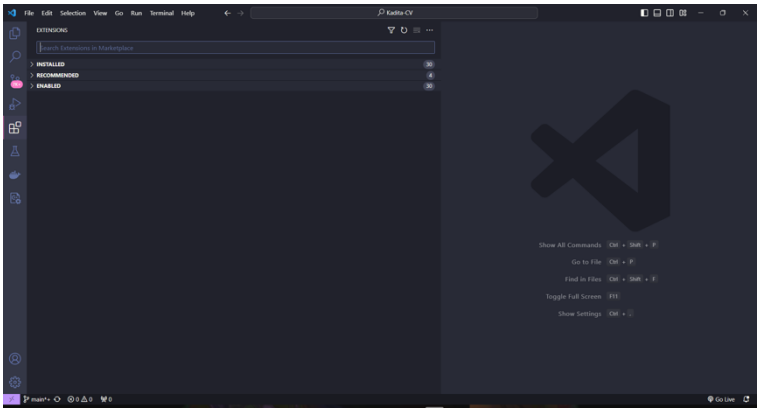


7. Lalu akan muncul menu seperti ini, artinya kamu sudah berhasil.

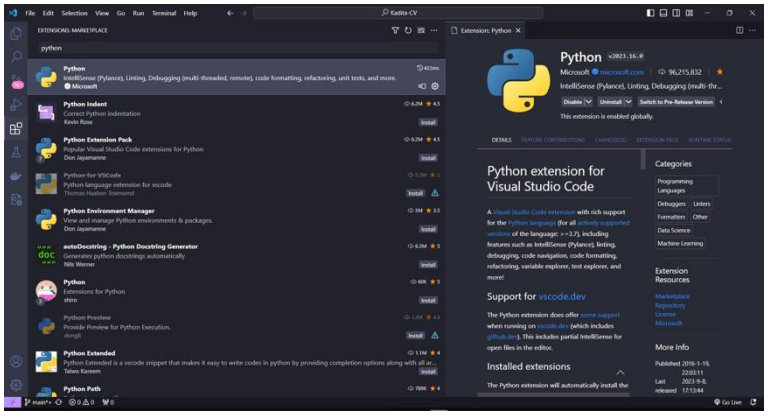


3.3 INSTALL PACKAGE VISUAL STUDIO CODE

1. Buka Extension di Visual Studio Code dengan menekan keyboard **CTRL+SHIFT+X**



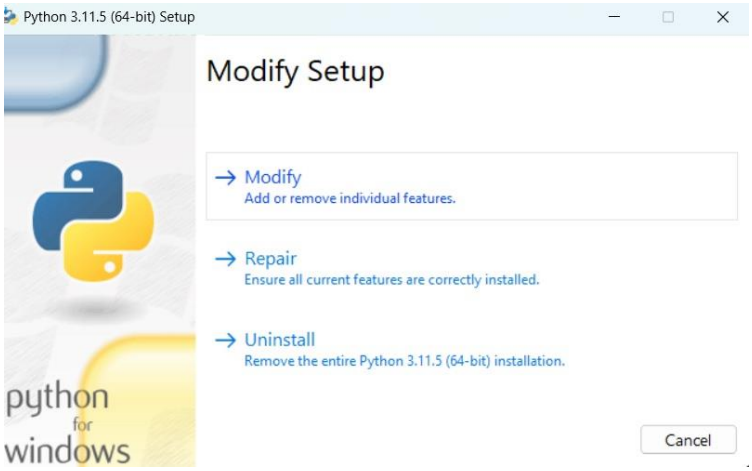
2. Lalu ketik "**Python"** di kolom pencarian lalu install.



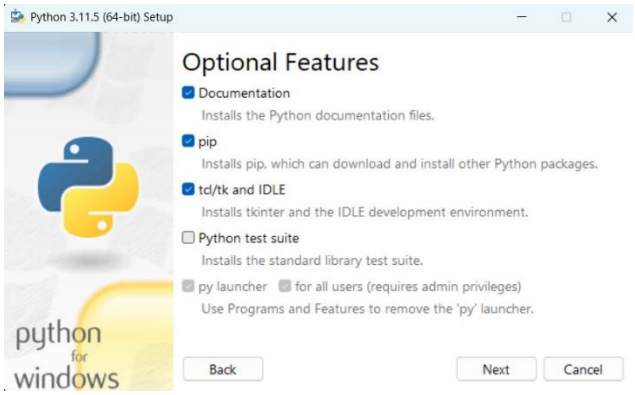
3. Setelah itu download Python dengan link dibawah ini
   https://www.python.org/downloads/

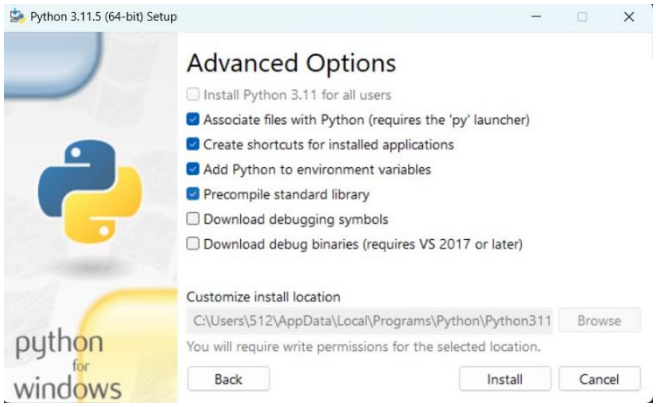
4. Tekan Install Now.

5. Setelah ditekan Install Now, maka selanjutnya tekan **Modify**

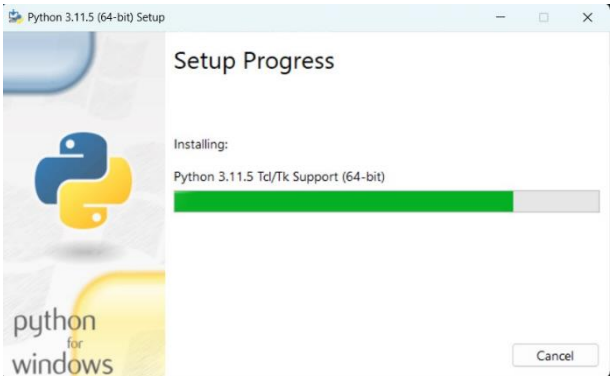


6. Selanjutnya, centang semua option sebagai berikut

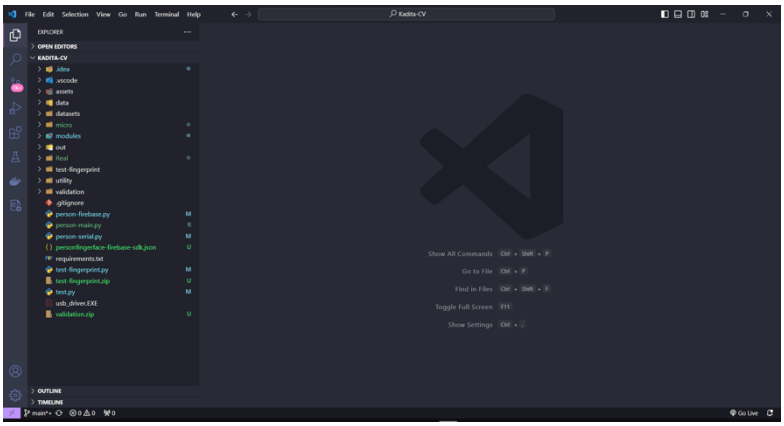




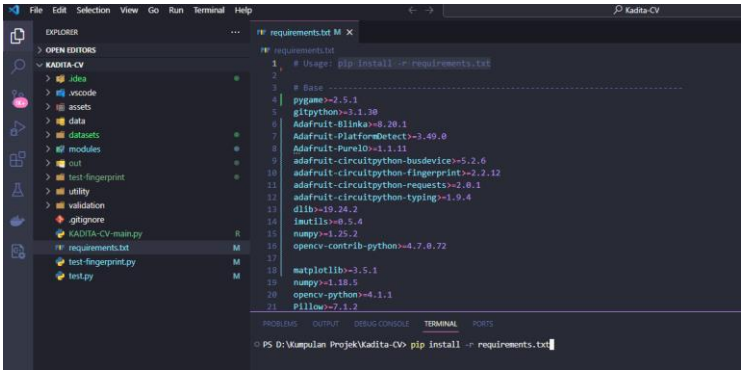
7. Lalu tunggu proses instalasinya



8. Pergi ke folder "**Kadita-CV**".

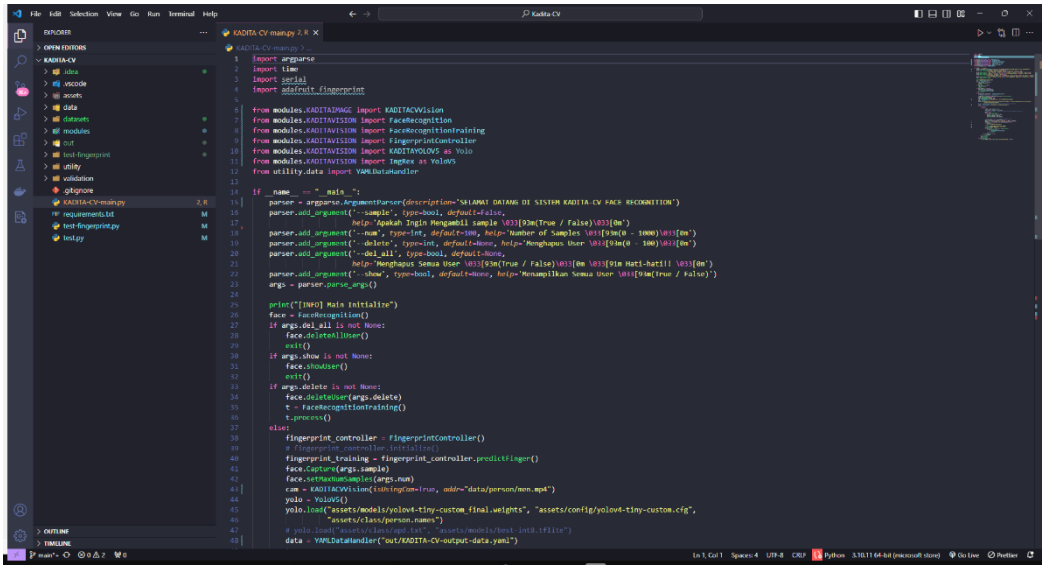


9. Buka terminal, lalu ketik "**pip install -r requirements.txt**"



10. Setelah menginstall semua package, akan keluar keterangan seperti di bawah:

11. **Framework Kadita-CV** siap digunakan.

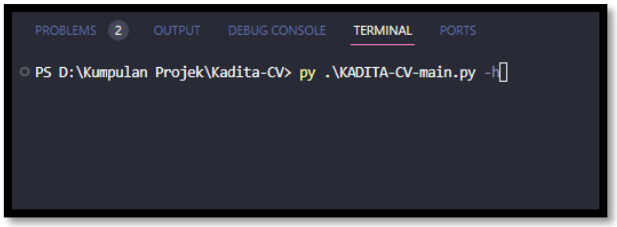


```
import argparse
import time
import serial
import adafruit_fingerprint

from modules.KADITAIMAGE import KADITACVVision
from modules.KADITAVISION import FaceRecognition
from modules.KADITAVISION import FaceRecognitionTraining
from modules.KADITAVISION import FingerprintController
from modules.KADITAVISION import KADITAYOLOV5 as Yolo
from modules.KADITAVISION import ImgRex as YoloV5
from utility.data import YAMLDataHandler

if __name__ == "__main__":
    parser = argparse.ArgumentParser(description='SELAMAT DATANG DI SISTEM KADITA-CV FACE RECOGNITION')
    parser.add_argument('--sample', type=bool, default=False,
                        help='Apakah Ingin Mengambil sample \033[93m(True / False)\033[0m')
    parser.add_argument('--num', type=int, default=100, help='Number of Samples \033[93m(0 - 1000)\033[0m')
    parser.add_argument('--delete', type=int, default=None, help='Menghapus User \033[93m(0 - 100)\033[0m')
    parser.add_argument('--del_all', type=bool, default=None,
                        help='Menghapus Semua User \033[93m(True / False)\033[0m \033[91m Hati-hati!! \033[0m')
    parser.add_argument('--show', type=bool, default=None, help='Menampilkan Semua User \033[93m(True / False)')
    args = parser.parse_args()

    print("[INFO] Main Initialize")
    face = FaceRecognition()
    if args.del_all is not None:
        face.deleteAllUser()
        exit()
    if args.show is not None:
        face.showUser()
        exit()
    if args.delete is not None:
        face.deleteUser(args.delete)
        t = FaceRecognitionTraining()
        t.process()
    else:
        fingerprint_controller = FingerprintController()
        # fingerprint_controller.initialize()
        fingerprint_training = fingerprint_controller.predictFinger()
        face.Capture(args.sample)
        face.setMaxNumSamples(args.num)
        cam = KADITACVVision(isUsingCam=True, addr="data/person/men.mp4")
        yolo = YoloV5()
        yolo.load("assets/models/yolov4-tiny-custom_final.weights", "assets/config/yolov4-tiny-custom.cfg",
                  "assets/class/person.names")
        # yolo.load("assets/class/ggl.txt", "assets/models/best-int8.tflite")
        data = YAMLDataHandler("out/KADITA-CV-output-data.yaml")
```

3.4 PRAKTIKUM 2 – FACE RECOGNITION

3.4.1 JALANKAN PROGRAM HELP PADA TRAINING FACE RECOGNITION

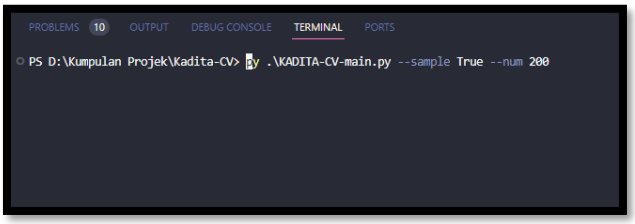
1. Buka Terminal di Visual Studio Code, lalu ketik "**py .\KADITA-CV-main.py -h**"



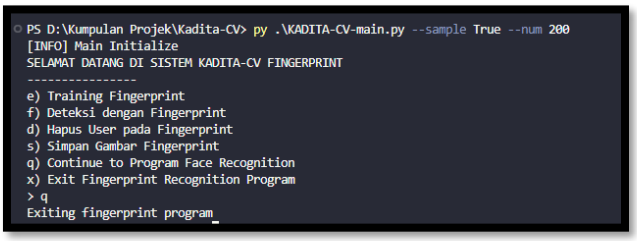
2. Akan keluar tampilan yang berupa beberapa perintah untuk menggunakan Framework:

3.4.2 JALANKAN PROGRAM UNTUK MENGAMBIL SAMPLE WAJAH

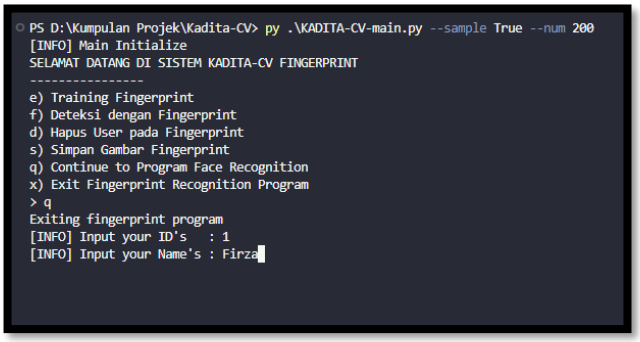
1. Tetap di terminal, lalu ketik "**py .\KADITA-CV-main.py --sample True --num 200**" Untuk pengambilan sampel (--num) di isi menyesuaikan dengan kebutuhan dalam range mulai dari 0-1000. Sebagai contoh diambil sample 200 untuk training wajah.



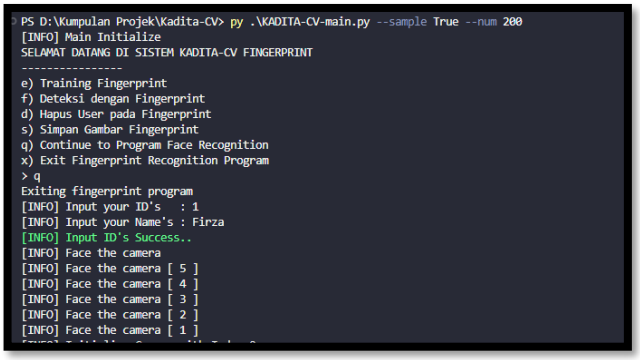
2. Setelah masuk program untuk selanjutnya tekan "**q"** untuk keluar dari Program Fingerprint.



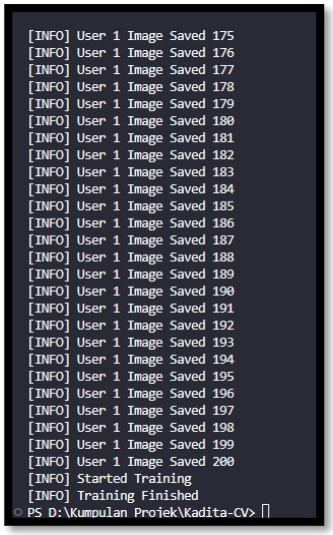
3. Masukan ID dan Nama yang ingin anda input mulai dari (0-100) disini saya ambil **contoh ID:1 dan Nama:Firza**.



4. Masing - masing anda semua harus menghadap ke Webcam pada Laptop untuk mengambil sample wajah sendiri.



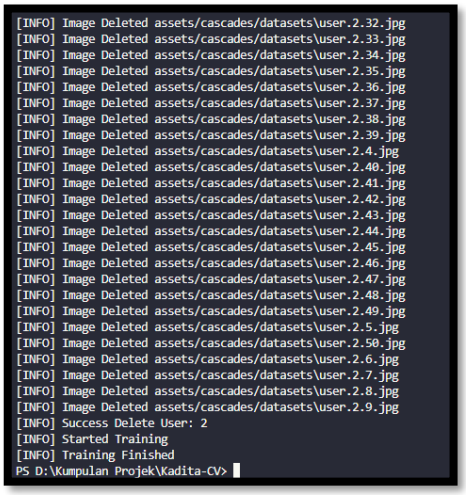
5. Selanjutnya system menunjukkan proses pengambilan data wajah melalui terminal.



3.4.3 MENGHAPUS BEBERAPA USER

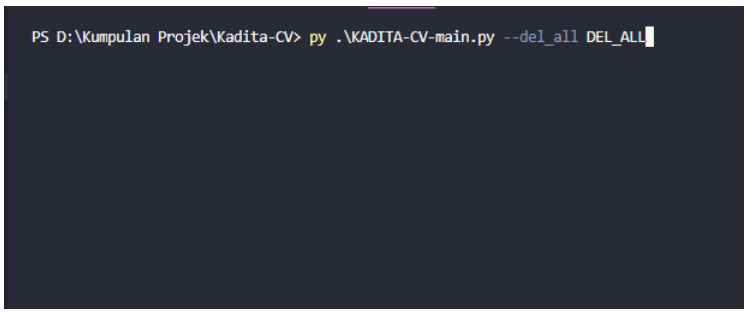
1. Tetap di terminal, lalu ketik **py .\KADITA-CV-main.py –delete 2**. Untuk menghapus user tergantung dari user mana yang ingin dihapus.

2. Selanjutnya system menunjukkan proses penghapusan data melalui terminal:

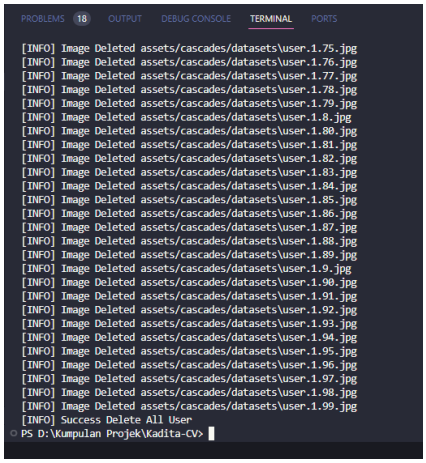


3.4.4 MENGHAPUS SEMUA USER

1. Di terminal, ketik **py .\KADITA-CV-main.py --del_all DEL_ALL**, hati hati jika anda menggunakan perintah ini dikarenakan dengan perintah ini akan menghapus semua User di program.

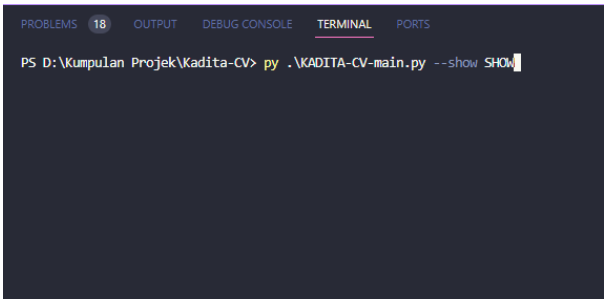


2. Setelah program dijalankan maka system akan menunjukkan proses hapus semua user pada terminal

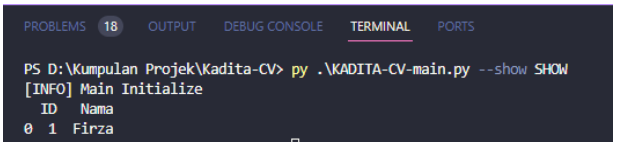


3.4.5 MENAMPILKAN SEMUA USER

1. Untuk mengetahui semua User di suatu sistemnya, maka ketik di terminal **“py .\KADITA-CV-main.py --show SHOW”**

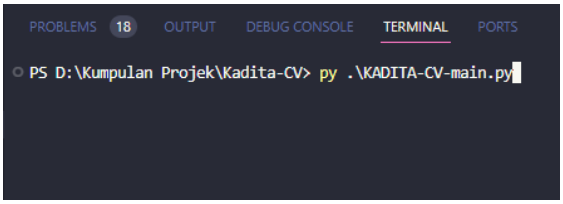


2. Setelah itu program akan berjalan, lalu menampilkan semua Usernya seperti ini



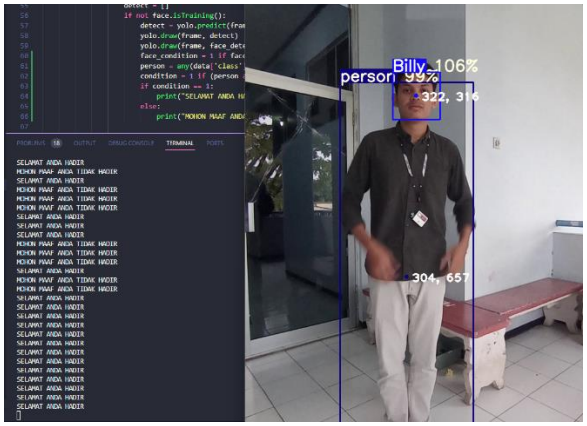
3.4.6 MENJALANKAN PROGRAM UTAMA

1. Setelah data training siap, maka langkah selanjutnya adalah menjalankan program utama dengan cara mengetik "**py .\KADITA-CV-main.py"** untuk mendapatkan hasil dari Face Recognition. Berikut adalah contohnya :



2. Program berjalan maka keterangan di terminal menunjukan gambar seperti di bawah ini:

   Sewaktu terdeteksi wajah maka di terminal keluar keterangan seperti di bawah :



Sewaktu tidak terdeteksi wajah maka di terminal keluar keterangan seperti di bawah :