

PEMAIN 1



KELOMPOK X



PEMAIN 2

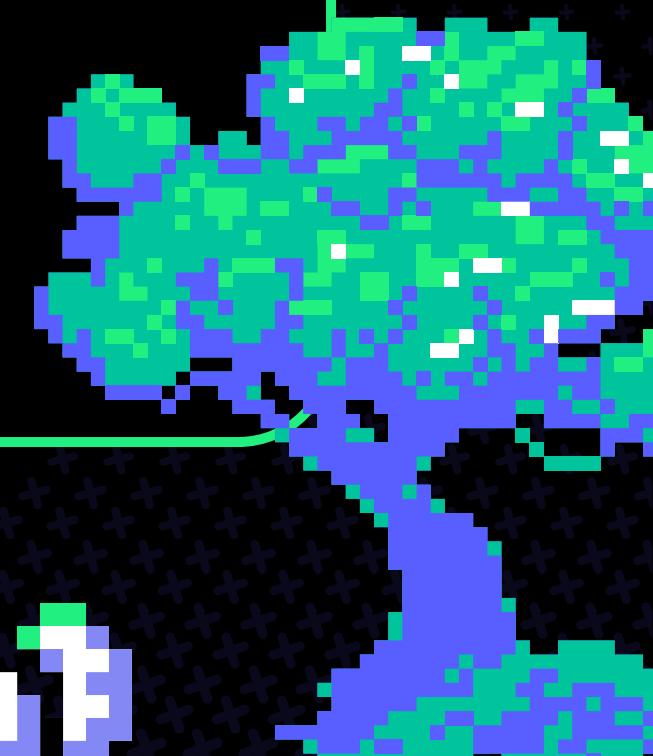
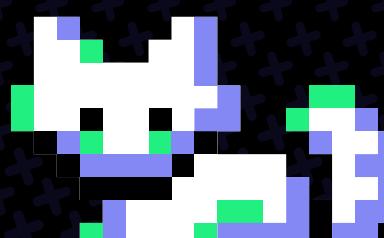
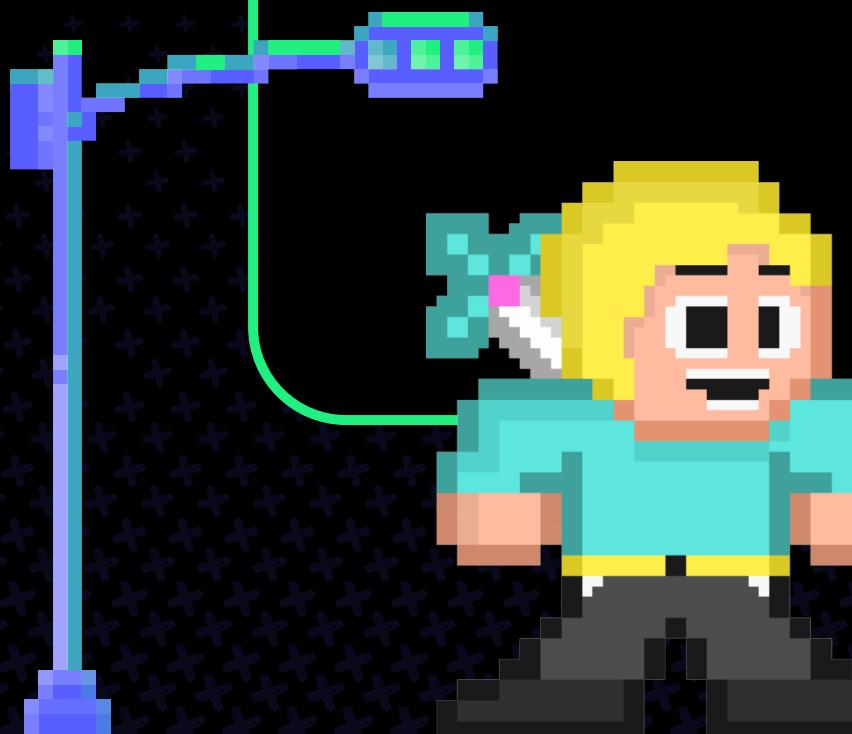
FINAL PROJECT

START

MENU

SIGN IN

ROCK PAPER SCISSORS WITH OPENCV AND CVZONE



[KE HALAMAN AWAL](#)

21

27

09



KELOMPOK X

← FREDERICK YONATAN / 5025211121

← NADIF MUSTAFA / 5025211127

← NIZAM HAKIM / 5025211209



MENU

DESKRIPSI SINGKAT

PROGRAM INI MERUPAKAN PERMAINAN SEDERHANA BATU-GUNTING-KERTAS YANG MENGGUNAKAN KAMERA UNTUK MENDETEKSI GERAKAN TANGAN PENGGUNA. SAAT PERMAINAN DIMULAI, PROGRAM AKAN MENGAMBIL GAMBAR DARI KAMERA DAN MENAMPILKAN SKOR. PROGRAM AKAN MENGENALI GERAKAN TANGAN PENGGUNA DAN MEMBANDINGKANNYA DENGAN GERAKAN ACAK YANG DIHASILKAN UNTUK LAWAN (AI). BERDASARKAN PERBANDINGAN TERSEBUT, PROGRAM AKAN MENENTUKAN SKOR PERMAINAN. HASIL PERMAINAN AKAN DITAMPILKAN DENGAN MENGGABUNGKAN GAMBAR LATAR BELAKANG DAN GAMBAR DARI KAMERA. PROGRAM INI MEMBERIKAN PENGALAMAN INTERAKTIF DALAM BERMAIN PERMAINAN BATU-GUNTING-KERTAS MELALUI PENGGUNAAN KAMERA DAN DETEksi TANGAN.

MENU

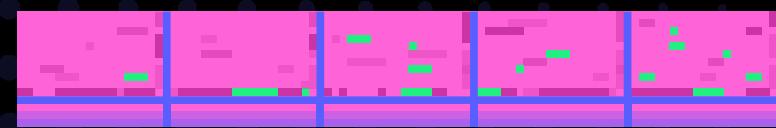
21

27

09



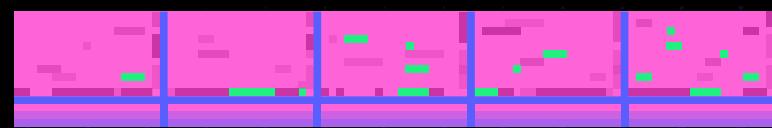
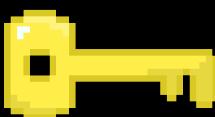
ALUR PROGRAM



MENGCAPTURE GESTUR
TANGAN DAN JARI



MENENTUKAN MOVE
PLAYER BERDASARKAN
HASIL CAPTURE



RANDOMIZE MOVE DARI
KOMPUTER



MENENTUKAN PEMENANG
BERDASARKAN HASIL

MENU

→ 21

◆ 27

★ 09



IMPORT PACKAGE YANG DIBUTUHKAN, YAITU:

- RANDOM: UNTUK RANDOMIZE MOVE DARI KOMPUTER
- CV2 DAN CVZONE : UNTUK CAPTURE VIDEO, IMAGE PROCESSING, DLL
- HANDDETECTOR : UNTUK TRACKING GERAKAN TANGAN
- TIME : UNTUK COUNTDOWN WAKTU MOVE

CAP UNTUK MEMBUKA KAMERA DAN MENGCAPTURE VIDEO DENGAN LEBAR 640 PIXEL DAN TINGGI 480 PIXEL

DETECTOR UNTUK MENDETEKSI TANGAN, MAKSIMAL TANGAN YANG DAPAT DIDETEKSI ADALAH 1

SET TIMER KE 0

STATERESULT UNTUK MENGHITUNG WAKTU SEBELUM MENGELOUARKAN MOVE

STARTGAME UNTUK MEMULAI GAME

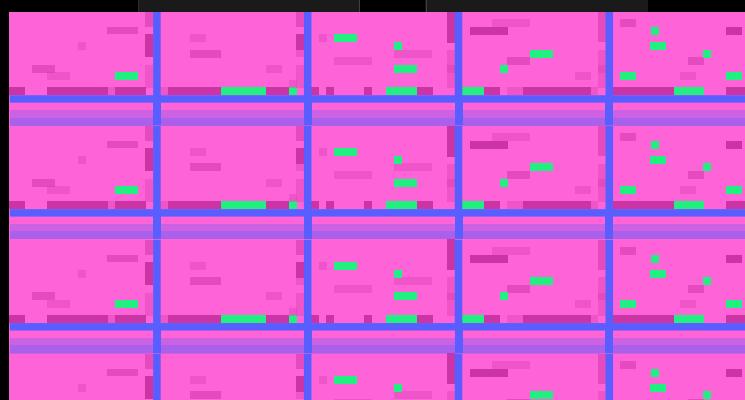
INITIAL SCORE ADALAH 0 0

```
1 import random
2 import cv2
3 import cvzone
4 from cvzone.HandTrackingModule import HandDetector
5 import time

6
7 cap = cv2.VideoCapture(0)
8 cap.set(3, 640)
9 cap.set(4, 480)

10
11 detector = HandDetector(maxHands=1)
12
13 timer = 0
14 stateResult = False
15 startGame = False
16 scores = [0, 0] # [AI, Player]
17
```

PEMAIN 1



```
18 while True:  
19     imgBG = cv2.imread("BG.png")  
20     success, img = cap.read()  
21  
22     imgScaled = cv2.resize(img, (0, 0), None, 0.875, 0.875)  
23     imgScaled = imgScaled[:, 80:480]  
24  
25     # Find Hands  
26     hands, img = detector.findHands(imgScaled) # with draw  
27  
28     if startGame:  
29  
30         if stateResult is False:  
31             timer = time.time() - initialTime  
32             cv2.putText(imgBG, str(int(timer)), (605, 435), cv2.FONT_HERSHEY_PLAIN, 6, (255, 0, 255), 4)  
33  
34         if timer > 3:  
35             stateResult = True  
36             timer = 0  
37
```

1. MEMBACA GAMBAR LATAR BELAKANG ("BG.PNG") DAN MENYIMPANNYA DALAM VARIABEL `IMGBG` MENGGUNAKAN `CV2.IMREAD()`.
2. MEMBACA FRAME GAMBAR DARI STREAM VIDEO MENGGUNAKAN `CAP.READ()` DAN MENYIMPANNYA DALAM VARIABEL `IMG`. VARIABEL `SUCCESS` MENUNJUKKAN APAKAH MEMBACA FRAME BERHASIL ATAU TIDAK.

MENU

21

27

09



3.

MENGUBAH UKURAN GAMBAR **IMG** MENJADI 87.5% DARI UKURAN ASLINYA MENGGUNAKAN **CV2.RESIZE()** DAN MENYIMPANNYA DALAM VARIABEL **IMGSCALED**. PEMANGKASAN (CROPPING) JUGA DILAKUKAN PADA GAMBAR DENGAN MENGAMBIL AREA TERTENTU MENGGUNAKAN INDEXING.

4.

MENDETEKSI TANGAN DALAM GAMBAR YANG SUDAH DIUBAH UKURANNYA MENGGUNAKAN **DETECTOR.FINDHANDS(IMGSCALED)**. HASIL DETEKSI TANGAN DISIMPAN DALAM VARIABEL **HANDS**.



```

38 v           if hands:
39
40             playerMove = None
41             hand = hands[0]
42             fingers = detector.fingersUp(hand)
43             if fingers == [0, 0, 0, 0, 0]:
44               playerMove = 1
45             if fingers == [1, 1, 1, 1, 1]:
46               playerMove = 2
47             if fingers == [0, 1, 1, 0, 0]:
48               playerMove = 3
49
50             randomNumber = random.randint(1, 3)
51             imgAI = cv2.imread(f'{randomNumber}.png', cv2.IMREAD_UNCHANGED)
52             imgBG = cvzone.overlayPNG(imgBG, imgAI, (149, 310))

```

JIKA TANGAN TERDETEKSI, BACA BANYAK JARI YANG TERDETEKSI:

- JIKA [0, 0, 0, 0, 0], ARTINYA SEMUA JARI TIDAK TERDETEKSI, MAKA PLAYER MOVE = 1 (ROCK)
- JIKA [1, 1, 1, 1, 1], ARTINYA SEMUA JARI TERDETEKSI, MAKA PLAYER MOVE = 2 (PAPER)
- JIKA [0, 1, 1, 0, 0], ARTINYA JARI TELUNJUK DAN TENGAH TERDETEKSI, MAKA PLAYER MOVE = 3 (SCISSORS)

KEMUDIAN RANDOMIZE ANGKA ANTARA 1--3 YANG DIGUNAKAN UNTUK MENENTUKAN MOVE KOMPUTER

- JIKA KELUAR ANGKA 1, MAKA SHOW 1.PNG PADA STORAGE (ROCK) KE UI
- JIKA KELUAR ANGKA 2, MAKA SHOW 2.PNG PADA STORAGE (PAPER) KE UI
- JIKA KELUAR ANGKA 3, MAKA SHOW 3.PNG PADA STORAGE (SCISSORS) KE UI

 21 27 09

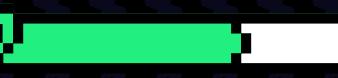
```
53 # Player Wins
54 if (playerMove == 1 and randomNumber == 3) or \
55     (playerMove == 2 and randomNumber == 1) or \
56     (playerMove == 3 and randomNumber == 2):
57     scores[1] += 1
58
59 # AI Wins
60 if (playerMove == 3 and randomNumber == 1) or \
61     (playerMove == 1 and randomNumber == 2) or \
62     (playerMove == 2 and randomNumber == 3):
63     scores[0] += 1
64
```

BANDINGKAN MOVE PLAYER DENGAN MOVE KOMPUTER MENGGUNAKAN ATURAN ROCK , PAPER , SCISSORS SEPERTI BIASA

- JIKA PLAYER MENANG , SCORES[1] += 1
- JIKA KOMPUTER MENANG , SCORES[0] += 1



SKOR TERTINGGI 2500



PEMAIN 2

```
64
65     imgBG[234:654, 795:1195] = imgScaled
66
67     if stateResult:
68         imgBG = cvzone.overlayPNG(imgBG, imgAI, (149, 310))
69
70     cv2.putText(imgBG, str(scores[0]), (410, 215), cv2.FONT_HERSHEY_PLAIN, 4, (255, 255, 255), 6)
71     cv2.putText(imgBG, str(scores[1]), (1112, 215), cv2.FONT_HERSHEY_PLAIN, 4, (255, 255, 255), 6)
72
73     # cv2.imshow("Image", img)
74     cv2.imshow("BG.png", imgBG)
75     # cv2.imshow("Scaled", imgScaled)
76
77     key = cv2.waitKey(1)
78     if key == ord('a'):
79         startGame = True
80         initialTime = time.time()
81         stateResult = False
```



1.

IMGBG[234:654, 795:1195] = IMGSCALED: MELAKUKAN PENGGABUNGAN (OVERLAY) GAMBAR IMGSCALED KE DALAM GAMBAR LATAR BELAKANG IMGBG. BAGIAN GAMBAR YANG DIGABUNGKAN DITENTUKAN OLEH RANGE PIKSEL DARI (234, 795) HINGGA (654, 1195).

2.

IF STATERESULT: BLOK KONDISI INI AKAN DIEKSEKUSI JIKA STATERESULT BERNILAI TRUE.

IMGBG = CVZONE.OVERLAYPNG(IMGBG, IMGAI, [149, 310]): MENGGABUNGKAN (OVERLAY) GAMBAR AI (IMGAI) KE DALAM GAMBAR LATAR BELAKANG IMGBG PADA POSISI YANG DITENTUKAN OLEH KOORDINAT (149, 310).INI DAPAT MEMBERIKAN EFEK VISUAL TAMBAHAN PADA PERMAINAN.

MENU

21

27

09



- CV2.PUTTEXT(IMGBG, STR(SCORES[0]), (410, 215), CV2.FONT_HERSHEY_PLAIN, 4, (255, 255, 255), 6): MENAMPILKAN SKOR AI (SCORES[0]) DI POSISI (410, 215) PADA GAMBAR LATAR BELAKANG IMGBG MENGGUNAKAN TEKS DENGAN FONT CV2.FONT_HERSHEY_PLAIN, UKURAN FONT 4, DAN WARNA TEKS PUTIH (255, 255, 255). PARAMETER TERAKHIR 6 ADALAH KETEBALAN TEKS.
- CV2.PUTTEXT(IMGBG, STR(SCORES[1]), (1112, 215), CV2.FONT_HERSHEY_PLAIN, 4, (255, 255, 255), 6): MENAMPILKAN SKOR PEMAIN (SCORES[1]) DI POSISI (1112, 215) PADA GAMBAR LATAR BELAKANG IMGBG DENGAN KONFIGURASI YANG SERUPA SEPERTI POIN SEBELUMNYA.

MENU

21

27

09

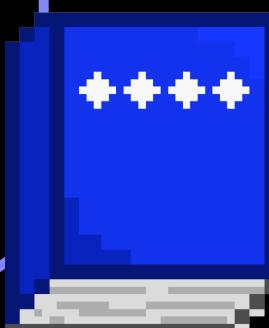


CV2.IMSHOW("BG.PNG", IMGBG): MENAMPIILKAN GAMBAR LATAR BELAKANG (IMGBG) DALAM JENDELA DENGAN JUDUL "BG.PNG" MENGGUNAKAN CV2.IMSHOW().

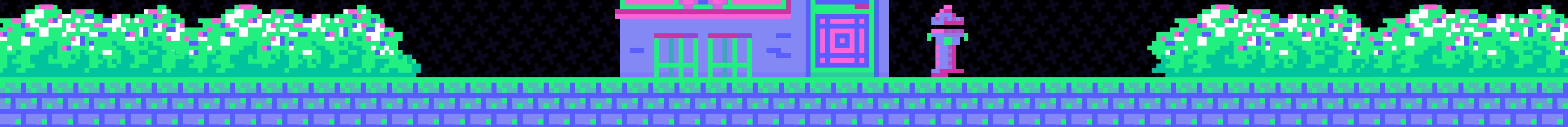
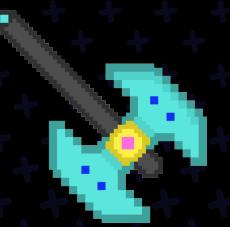
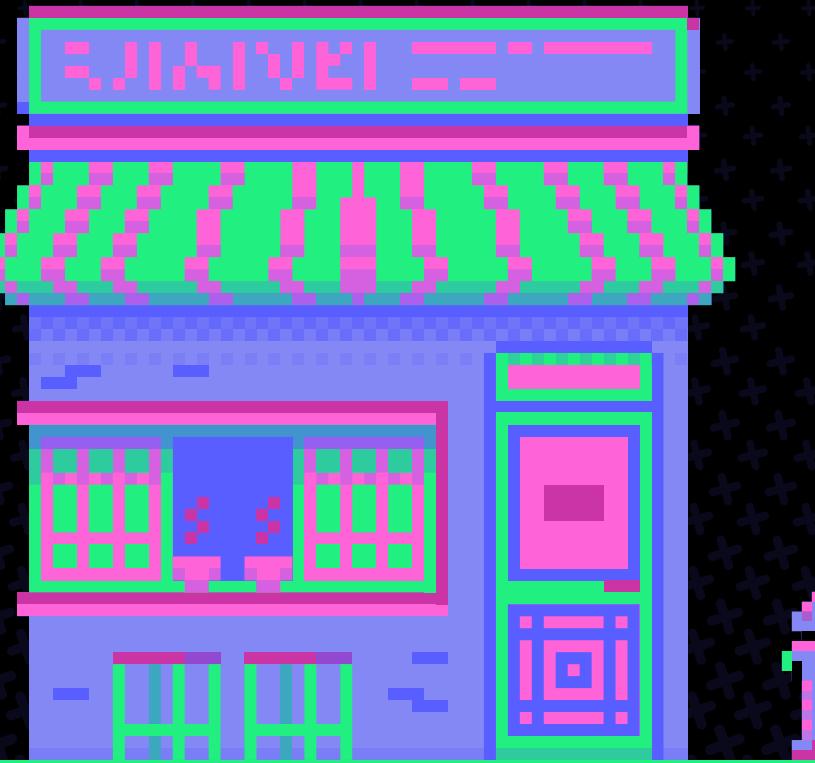
KEY = CV2.WAITKEY(1): MEMBACA TOMBOL YANG DITEKAN DARI KEYBOARD MENGGUNAKAN CV2.WAITKEY() DAN MENYIMPANNYA DALAM VARIABEL KEY.

IF KEY == ORD('A'): BLOK KONDISIINI AKAN DIEKSEKUSI JIKA TOMBOL 'A' DITEKAN.

- STARTGAME = TRUE: MENGATUR STARTGAME MENJADI TRUE, YANG DAPAT MEMULAI PERMAINAN.
- INITIALETIME = TIME.TIME(): MENGATUR INITIALETIME MENJADI WAKTU SAAT INI.
- STATERESULT = FALSE: MENGATUR STATERESULT MENJADI FALSE, SEHINGGA DAPAT MEMULAI PERMAINAN DARI AWAL.

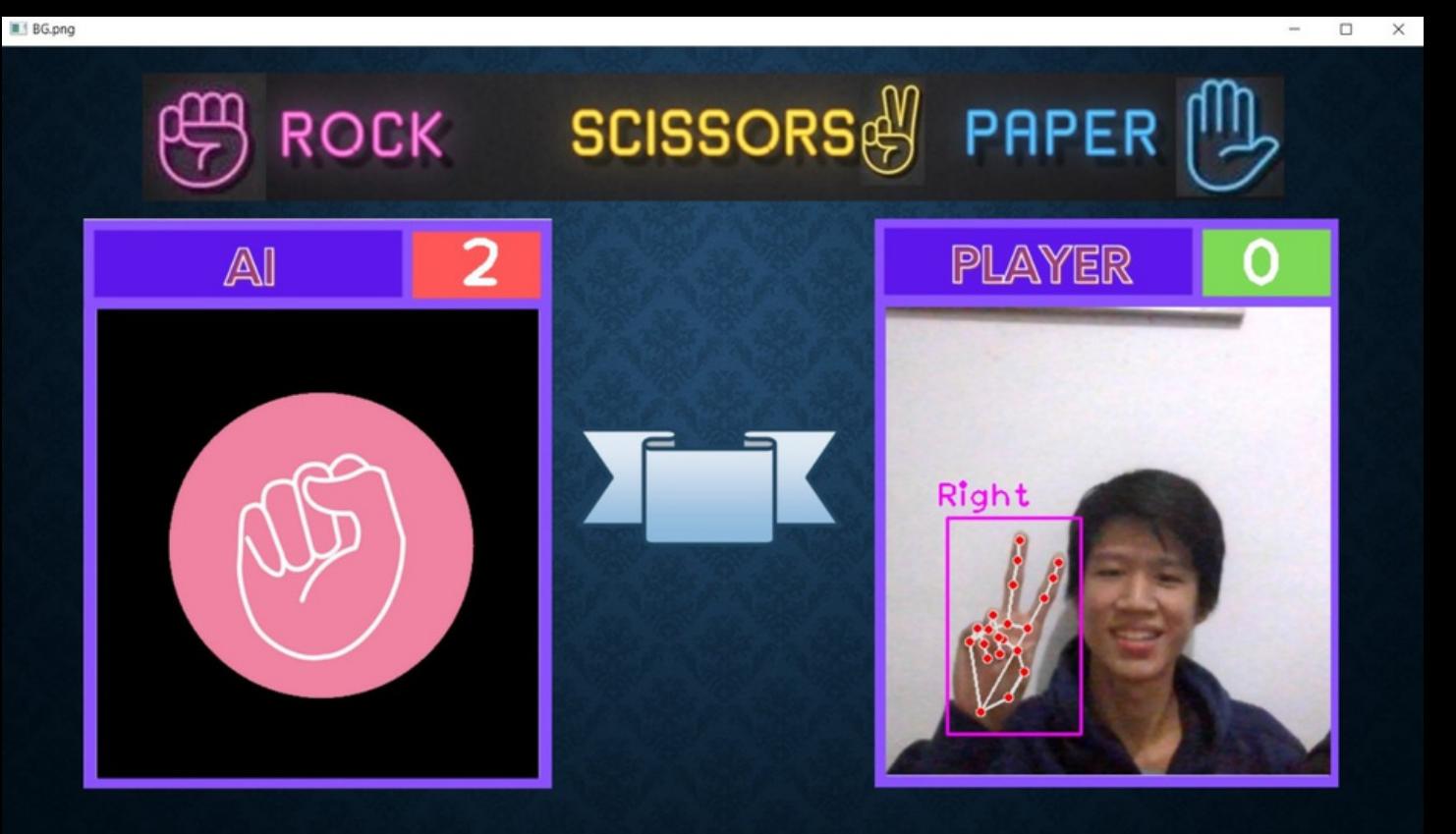


CONTOH RUMAH





1



2



3

MENU



TERIMA KASIH!

→ 21 ♦ 27 ★ 09