

Appendices



Appendix 1. Blackbox Test Table

Test Case ID	Module/Feature Tested	Test Description	Test Steps	Data Input (User/Gesture/Environment)	Expected Outcome (Robotic Hand/Logs)	Actual Outcome	Status (Pass/Fail)	Notes
BB-001	Hand Gesture Recognition (First Number), System State Transition	Test the system's ability to recognize the first numerical hand sign input and internally store it.	1. Start the Raspberry Pi system. 2. Present the hand gesture for "Number 5" to the camera. 3. Maintain the gesture for a short duration, then remove the hand from the frame.	Hand gesture "Number 5"	Raspberry Pi logs indicate successful detection of "Number 5" and transition to "WAIT_OPERATOR" state. Robotic hand remains in its initial or neutral position.			
BB-002	Hand Gesture Recognition (Operator), System State Transition	Test the system's ability to recognize an arithmetic operator hand sign input after the first number.	1. Continue from BB-001 (system in WAIT_OPERATOR state). 2. Present the hand gesture for "Operator +" to the camera. 3. Maintain the gesture for a short duration, then remove the hand from the frame.	Hand gesture "Operator +"	Raspberry Pi logs indicate successful detection of "Operator +" and transition to "WAIT_SECOND_NUM" state. Robotic hand remains in its initial or neutral position.			

Test Case ID	Module/Feature Tested	Test Description	Test Steps	Data Input (User/Gesture/Environment)	Expected Outcome (Robotic Hand/Logs)	Actual Outcome	Status (Pass/Fail)	Notes
BB-003	Hand Gesture Recognition (Second Number), System State Update	Test the system's ability to recognize the second numerical hand sign input.	1. Continue from BB-002 (system in WAIT_SECOND_NUM state). 2. Present the hand gesture for "Number 3" to the camera. 3. Maintain the gesture for a short duration, then remove the hand from the frame.	Hand gesture "Number 3"	Raspberry Pi logs indicate successful detection of "Number 3" and the number_2 variable being updated to "3". Robotic hand remains in its initial or neutral position.			
BB-004	Arithmetic Calculation, Result Mapping, Robotic Hand Actuation	Test the system's ability to perform the arithmetic calculation based on stored inputs and actuate the robotic hand to	1. Continue from BB-003 (inputs "5", "+", "3" stored). 2. Ensure no hand is visible in the camera frame for a duration exceeding max_no_hand_threshold (e.g., 10 frames).	No hand visible in the camera frame (trigger for calculation).	Raspberry Pi logs show calculation "5 + 3 = 8". Raspberry Pi logs show conversion of "8" to corresponding servo configuration. The robotic hand moves to physically display the "Number 8" gesture according			

Test Case ID	Module/Feature Tested	Test Description	Test Steps	Data Input (User/Gesture/Environment)	Expected Outcome (Robotic Hand/Logs)	Actual Outcome	Status (Pass/Fail)	Notes
		display the result.			to its servo mapping.			
BB-005	System State Logic, Error Handling	Test the system's response when only the first number is provided, and no further input follows within the timeout.	1. Start the Raspberry Pi system. 2. Present the hand gesture for "Number 7" to the camera. 3. Remove the hand and ensure no further input is given for a duration exceeding max_no_hand_threshold.	Hand gesture "Number 7", followed by no hand input.	Raspberry Pi logs indicate "Incomplete input" or similar waiting state. The robotic hand remains inactive or in its neutral position.			
BB-006	Input Overwriting Logic	Test if the system correctly uses the last numerical input as the first number if the user changes	1. Start the Raspberry Pi system. 2. Present "Number 2" to the camera. 3. Immediately present "Number 4" to the camera. 4. Present "Operator x" to the camera. 5. Present "Number 3"	Sequence of gestures: "Number 2", "Number 4", "Operator x", "Number 3".	Raspberry Pi logs indicate the first number being "4" (overwritten from "2"). Logs show calculation "4 x 3 = 12". The robotic hand displays the corresponding gesture for "12" (or its defined			

Test Case ID	Module/Feature Tested	Test Description	Test Steps	Data Input (User/Gesture/Environment)	Expected Outcome (Robotic Hand/Logs)	Actual Outcome	Status (Pass/Fail)	Notes
		their mind before providing an operator.	to the camera. 6. Ensure no hand is visible for a duration exceeding max_no_hand_threshold.		behavior for out-of-range numbers, as per system design).			
BB-007	Robotic Hand Response to Out-of-Range Results	Test the robotic hand's behavior when the calculation result is a number it cannot directly display (e.g., a two-digit number).	1. Start the Raspberry Pi system. 2. Present "Number 5" to the camera. 3. Present "Operator +" to the camera. 4. Present "Number 5" to the camera. 5. Ensure no hand is visible for a duration exceeding max_no_hand_threshold.	Sequence of gestures: "Number 5", "Operator +", "Number 5".	Raspberry Pi logs show calculation "5 + 5 = 10". The robotic hand performs its predefined response for out-of-range numbers (e.g., remaining stationary, displaying an "error" gesture, or displaying only the last digit "0" if that's the design).			
BB-008	Specific Gesture Recognition	Test if the system correctly recognizes	1. Start the Raspberry Pi system. 2. Present the hand gesture for	Hand gesture "Start"	Raspberry Pi logs indicate detection of "Start" and a readiness			

Test Case ID	Module/Feature Tested	Test Description	Test Steps	Data Input (User/Gesture/Environment)	Expected Outcome (Robotic Hand/Logs)	Actual Outcome	Status (Pass/Fail)	Notes
		the "Start" gesture to initiate interaction.	"Start" to the camera. 3. Remove the hand.		indication (e.g., "System ready for input"). Robotic hand remains neutral.			



Appendix 2. Whitebox Test Table

Test Case ID	Modul/Fungsi/Kode yang Diuji	Deskripsi Uji	Kondisi/Path yang Diuji	Data Input (Internal/Parameter Fungsi)	Hasil yang Diharapkan (Internal/Return Value)	Hasil Aktual (Internal)	Status (Lulus/Gagal)	Catatan
WB-001	Fungsi Ekstraksi Fitur (HandTrackingModule)	Menguji deteksi 21 titik landmark tangan dalam kondisi ideal.	Frame citra tangan jelas, pencahayaan baik, satu tangan terlihat.	frame_image (citra tangan tunggal yang jelas)	Output berupa array/list berisi 21 koordinat (x,y,z atau x,y) untuk landmark tangan dan bounding box.			
WB-002	Fungsi Ekstraksi Fitur (HandTrackingModule)	Menguji penanganan jika tidak ada tangan terdeteksi dalam frame.	Frame citra tanpa tangan atau tangan tidak jelas.	frame_image (citra tanpa tangan)	Output berupa array/list kosong atau nilai None untuk koordinat landmark dan bounding box. no_hand_counter bertambah.			

Test Case ID	Modul/Fungsi/Kode yang Diuji	Deskripsi Uji	Kondisi/Path yang Diuji	Data Input (Internal/Parameter Fungsi)	Hasil yang Diharapkan (Internal/Return Value)	Hasil Aktual (Internal)	Status (Lulus/Gagal)	Catatan
WB-003	Logika State Machine (Transisi dari WAIT_FIRST_NUM)	Memverifikasi transisi status setelah prediksi angka valid diterima saat status WAIT_FIRST_NUM.	current_state == "WAIT_FIRST_NUM" DAN prediction in 0-9.	current_state = "WAIT_FIRST_NUM", prediction = "7"	current_state berubah menjadi "WAIT_OPERATOR". number_1 diisi dengan "7".			
WB-004	Logika State Machine (Transisi dari WAIT_OPERATOR)	Memverifikasi transisi status setelah prediksi operator valid diterima saat status WAIT_OPERATOR.	current_state == "WAIT_OPERATOR" DAN prediction in ["+", "-", "*", "/"]	current_state = "WAIT_OPERATOR", prediction = "*"	current_state berubah menjadi "WAIT_SECOND_NUM". operator diisi dengan "*".			

Test Case ID	Modul/Fungsi/Kode yang Diuji	Deskripsi Uji	Kondisi/Path yang Diuji	Data Input (Internal/Parameter Fungsi)	Hasil yang Diharapkan (Internal/Return Value)	Hasil Aktual (Internal)	Status (Lulus/Gagal)	Catatan
WB-005	Logika State Machine (Input Angka saat WAIT_OPERATOR)	Memverifikasi number_1 di-overwrite jika angka baru diinput saat status WAIT_OPERATOR.	current_state == "WAIT_OPERATOR" DAN prediction in 0-9.	current_state = "WAIT_OPERATOR", number_1 (sebelumnya) = "5", prediction = "2"	current_state tetap "WAIT_OPERATOR". number_1 di-overwrite menjadi "2".			Pastikan ini perilaku yang diinginkan sesuai proposal.
WB-006	Logika Kalkulasi (Fungsi safe_evaluate)	Menguji fungsi evaluasi ekspresi aritmetika untuk operasi penjumlahan.	Ekspresi valid, contoh: "5+3".	expression = "5+3"	result = 8.			

Test Case ID	Modul/Fungsi/Kode yang Diuji	Deskripsi Uji	Kondisi/Path yang Diuji	Data Input (Internal/Parameter Fungsi)	Hasil yang Diharapkan (Internal/Return Value)	Hasil Aktual (Internal)	Status (Lulus/Gagal)	Catatan
WB-007	Logika Kalkulasi (Fungsi safe_evaluate)	Menguji penanganan pembagian dengan nol (jika safe_evaluate menangannya).	Ekspresi dengan pembagian nol, contoh: "8/0".	expression = "8/0"	Penanganan error yang terdefinisi (misal, None, Exception, atau nilai khusus seperti "Error").			
WB-008	Fungsi Mapping Hasil ke Konfigurasi Servo (mapping[result])	Menguji mapping hasil angka (misal "1") ke konfigurasi nilai servo yang benar.	Hasil kalkulasi adalah angka "1".	result = 1 (atau string "1")	servo_config berisi nilai PWM yang sesuai untuk gestur "1" (misal, [6,1,6,6,6] berdasarkan tabel proposal).			Cek batas nilai servo 150-800.

Test Case ID	Modul/Fungsi/Kode yang Diuji	Deskripsi Uji	Kondisi/Path yang Diuji	Data Input (Internal/Parameter Fungsi)	Hasil yang Diharapkan (Internal/Return Value)	Hasil Aktual (Internal)	Status (Lulus/Gagal)	Catatan
WB-009	Logika Trigger Kalkulasi (no_hand_counter)	Menguji kalkulasi dipicu ketika no_hand_counter mencapai max_no_hand_threshold dan input valid.	current_state == "WAIT_SECOND_NUM", no_hand_counter >= max_no_hand_threshold, number_1, number_2, operator valid.	no_hand_counter = 10, max_no_hand_threshold = 10, number_1 = "2", operator = "+", number_2 = "3"	Fungsi safe_evaluate dipanggil dengan ekspresi "2+3". actuate_servos dipanggil dengan konfigurasi servo untuk hasil "5".			
WB-010	Model YOLOv11-Pose (Prediksi Kelas Gestur)	Menguji akurasi klasifikasi gestur (angka/operator) setelah keypoint terdeteksi.	Input citra gestur tangan yang jelas (misal, gestur "Angka 7").	image_of_gesture_7	Model mengklasifikasikan gestur sebagai kelas yang benar (misal, "7") dengan tingkat kepercayaan (confidence score) yang tinggi.			Periksa confusion matrix dari hasil validasi.