**Phase 4**

**Project Development Part 2 Performance Test**

|  |  |
| --- | --- |
| Name | Bhuvaneshwari V |
| College code | 6220 |
| Date | 31 October 2023 |
| Project Name | Public Transportation optimization |

**Model Performance Testing:**

|  |  |  |
| --- | --- | --- |
| **Parameter** | **Values** | **Screenshot** |
| Metrics | Wowki Execution time and Output screenshot  Or  Python accuracy of prediction and output screenshot |  |

Code:

#define BLYNK\_TEMPLATE\_ID "TMPL26V4fGv5q"

#define BLYNK\_TEMPLATE\_NAME "Test"

#define BLYNK\_AUTH\_TOKEN "XEHxNF\_Ur1Nt2p7wB5B20dNI1ZUwj34P"

#include <WiFi.h>

#include <WiFiClient.h> #include <BlynkSimpleEsp32.h>

int duration1 = 0; int distance1 = 0; int duration2 = 0; int distance2 = 0; int dis1 = 0; int dis2 = 0; int dis\_new1 = 0; int dis\_new2 = 0; int entered = 0; int left = 0; int inside = 0; #define LED 2

#define PIN\_TRIG1 15

#define PIN\_ECHO1 14

#define PIN\_TRIG2 13

#define PIN\_ECHO2 12 BlynkTimer timer;

char auth[] = BLYNK\_AUTH\_TOKEN; char ssid[] = "Wokwi-GUEST"; // your network SSID (name) char pass[] = ""; #define BLYNK\_PRINT **Serial**

long get\_distance1() { // Start a new measurement: digitalWrite(PIN\_TRIG1, HIGH); delayMicroseconds(10); digitalWrite(PIN\_TRIG1, LOW);

// Read the result:

duration1 = pulseIn(PIN\_ECHO1, HIGH); distance1 = duration1 / 58; return distance1;

}

long get\_distance2() { // Start a new measurement: digitalWrite(PIN\_TRIG2, HIGH); delayMicroseconds(10); digitalWrite(PIN\_TRIG2, LOW);

// Read the result:

duration2 = pulseIn(PIN\_ECHO2, HIGH); distance2 = duration2 / 58; return distance2;

}

void myTimer() { **Serial**.println("100"); dis\_new1 = get\_distance1(); dis\_new2 = get\_distance2(); if (dis1 != dis\_new1 || dis2 != dis\_new2){ **Serial**.println("200"); if (dis1 < dis2){

**Serial**.println("Enter loop"); entered = entered + 1;

inside = inside + 1; digitalWrite(LED, HIGH); Blynk.virtualWrite(V0, entered); Blynk.virtualWrite(V2, inside); dis1 = dis\_new1; delay(1000); digitalWrite(LED, LOW);

}

if (dis1 > dis2){

**Serial**.println("Leave loop"); left = left + 1; inside = inside - 1;

Blynk.virtualWrite(V1, left); Blynk.virtualWrite(V2, inside); dis2 = dis\_new2; delay(1000);

}

}

}

void setup() {

**Serial**.begin(115200); pinMode(LED, OUTPUT); pinMode(PIN\_TRIG1, OUTPUT); pinMode(PIN\_ECHO1, INPUT); pinMode(PIN\_TRIG2, OUTPUT); pinMode(PIN\_ECHO2, INPUT);

Blynk.begin(auth, ssid, pass, "blynk.cloud", 8080); timer.setInterval(1000L, myTimer);

}

void loop() { Blynk.run(); timer.run(); }

**Link:**

<https://wokwi.com/projects/378905581521445889>

**Simulation:**

