Name: Vedant Patil

Class: TY-AIEC Batch: A

Roll. No.: 2223981

**Title**

**Keyword Spotting Project like “OK, Google,” “Alexa,” on Edge Devices using Microphone**

**Program:**

#include <PDM.h>

#include <Your\_EdgeImpulse\_Library.h> // replace with actual .h name from the .zip library

#define BUFFER\_SIZE 1024

static int16\_t sampleBuffer[BUFFER\_SIZE];

volatile int samplesRead = 0;

void onPDMData() {

int bytesAvailable = PDM.available();

PDM.read(sampleBuffer, bytesAvailable);

samplesRead = bytesAvailable / 2;

}

void setup() {

Serial.begin(115200);

while (!Serial);

Serial.println("Keyword Spotting - Nano 33 BLE Sense");

// Setup PDM microphone

PDM.onReceive(onPDMData);

if (!PDM.begin(1, 16000)) {

Serial.println("Failed to start PDM!");

while (1);

}

// Initialize impulse

run\_classifier\_init();

}

void loop() {

if (samplesRead) {

signal\_t signal;

int err = numpy::signal\_from\_buffer((const int16\_t\*)sampleBuffer, samplesRead, &signal);

if (err != 0) {

Serial.println("Signal creation failed");

return;

}

ei\_impulse\_result\_t result;

EI\_IMPULSE\_ERROR res = run\_classifier(&signal, &result, false);

if (res != EI\_IMPULSE\_OK) {

Serial.println("Classifier failed");

return;

}

// Print prediction

Serial.println("Predictions:");

for (size\_t i = 0; i < result.classification.count; i++) {

Serial.print(result.classification[i].label);

Serial.print(": ");

Serial.println(result.classification[i].value, 3);

}

samplesRead = 0;

}

}