GESTURE RECOGNITION

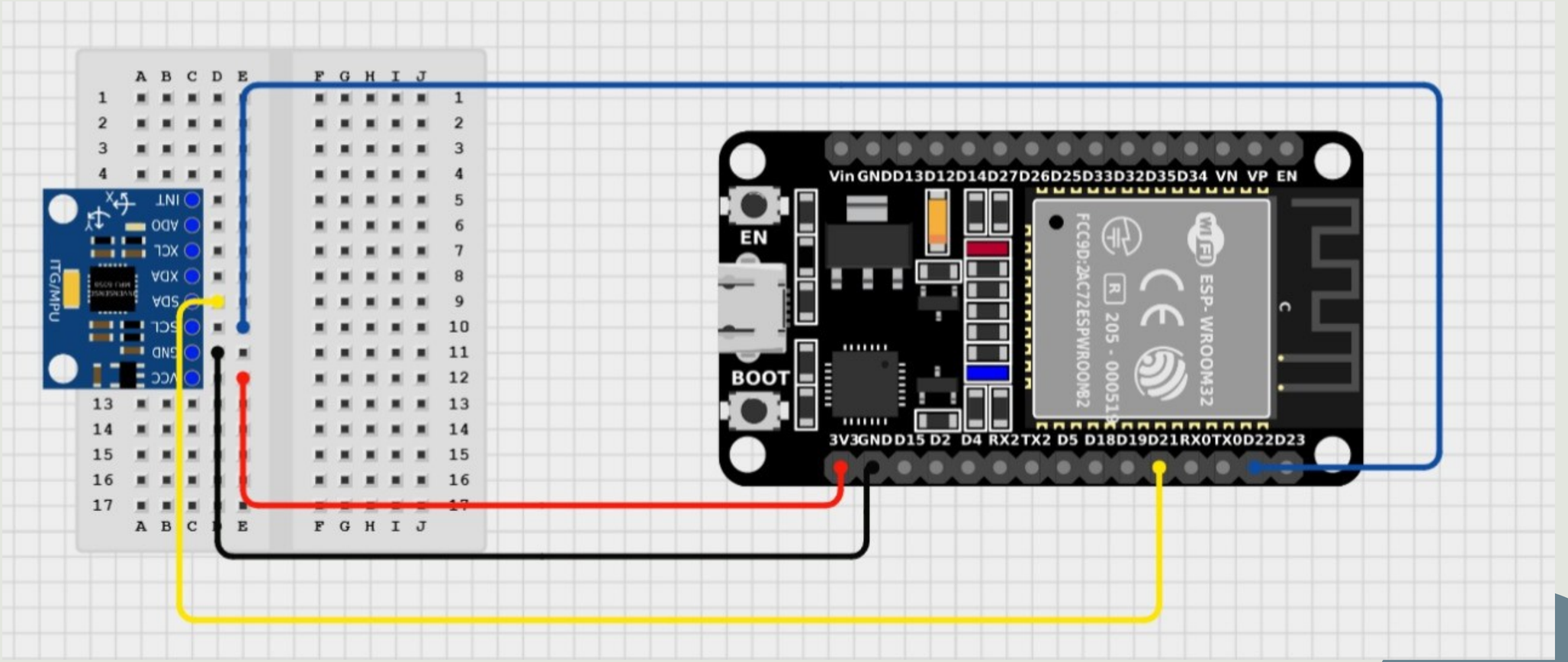
REAL WORLD APPLICATIONS

- **Human-Computer Interaction (HCI)**: Control devices like smart TVs, gaming consoles, and VR systems using natural hand gestures without physical controllers.
- **Retail and Marketing**: Analyze customer behavior in stores by tracking gestures to optimize layout and product placement.
- **Virtual and Augmented Reality**: Enable immersive interactions within virtual spaces through hand and body motion tracking.
- **Robot Control**: Gesture commands help robots perform tasks and interact with humans in industries and assistive services.
- **Sign Language Recognition**: Translate sign languages in real-time, aiding communication for the deaf and mute.

WORKING OF THE PROJECT

- Connect the MPU6050 sensor to the ESP32 and use Edge Impulse’s data forwarder to send gesture data during motion.
- Label the collected data in Edge Impulse Studio by categorizing each sample according to its gesture.
- Design and train a model in Edge Impulse Studio using the labeled data with motion-optimized processing and classification blocks.
- Download the trained model and deploy it to Arduino IDE to enable real-time processing of live sensor data.
- The device now instantly detects and classifies hand gestures using new data from the MPU6050 sensor.

PEHLE CONNECTION TOH DEKHLO!!



FIRMWARE CODE

sketch_feb2a.ino

```
6
7   // 1. Include MPU6050 libraries
8   #include <Adafruit_MPU6050.h>
9   #include <Adafruit_Sensor.h>
10  #include <Wire.h>
11
12  // 2. Set the sampling frequency (Hz)
13  // Common frequencies for gesture recognition are 62.5Hz or 100Hz
14  #define FREQUENCY_HZ        100
15  #define INTERVAL_MS         (1000 / FREQUENCY_HZ)
16
17  // Globals for sensor and timing
18  Adafruit_MPU6050 mpu;
19  unsigned long last_interval_ms = 0;
20
21  void setup() {
22      Serial.begin(115200);
23
24      // Initialize the MPU6050
25      if (!mpu.begin()) {
26          Serial.println("Failed to find MPU6050 chip");
27          while (1) {
28              delay(10);
29          }
30      }
31
32      // Set a reasonable accelerometer range
33      mpu.setAccelerometerRange(MPU6050_RANGE_8_G);
34
35      Serial.println("MPU6050 Found! Ready to collect data.");
36  }
37
38  void loop() {
```

Output Serial Monitor

THIS CODE IS USED A FIRMWARE TO YOUR ESP WHICH ENABLE IT TO SEND YOUR DATA THROUGH A BRIDGE CALLED EDGE IMPULSE CLI.THIS IS ALREADY UPLOADED IN YOUR ESP..

sketch_feb2a.ino

```
23
24      // Initialize the MPU6050
25      if (!mpu.begin()) {
26          Serial.println("Failed to find MPU6050 chip");
27          while (1) {
28              delay(10);
29          }
30      }
31
32      // Set a reasonable accelerometer range
33      mpu.setAccelerometerRange(MPU6050_RANGE_8_G);
34
35      Serial.println("MPU6050 Found! Ready to collect data.");
36  }
37
38  void loop() {
39      // Ensure we are sampling at the correct frequency
40      if (millis() < last_interval_ms + INTERVAL_MS) {
41          return;
42      }
43      last_interval_ms = millis();
44
45      // Get a new sensor event
46      sensors_event_t a, g, temp;
47      mpu.getEvent(&a, &g, &temp);
48
49      // Print the accelerometer data in the required format (CSV)
50      Serial.print(a.acceleration.x);
51      Serial.print(',');
52      Serial.print(a.acceleration.y);
53      Serial.print(',');
54      Serial.println(a.acceleration.z);
55  }
```

CLI INSTALLATION

CONNECT THE ESP SETUP TO YOUR COMPUTER AND OPEN THE COMMAND PROMPT. THEN ENTER THE COMMAND

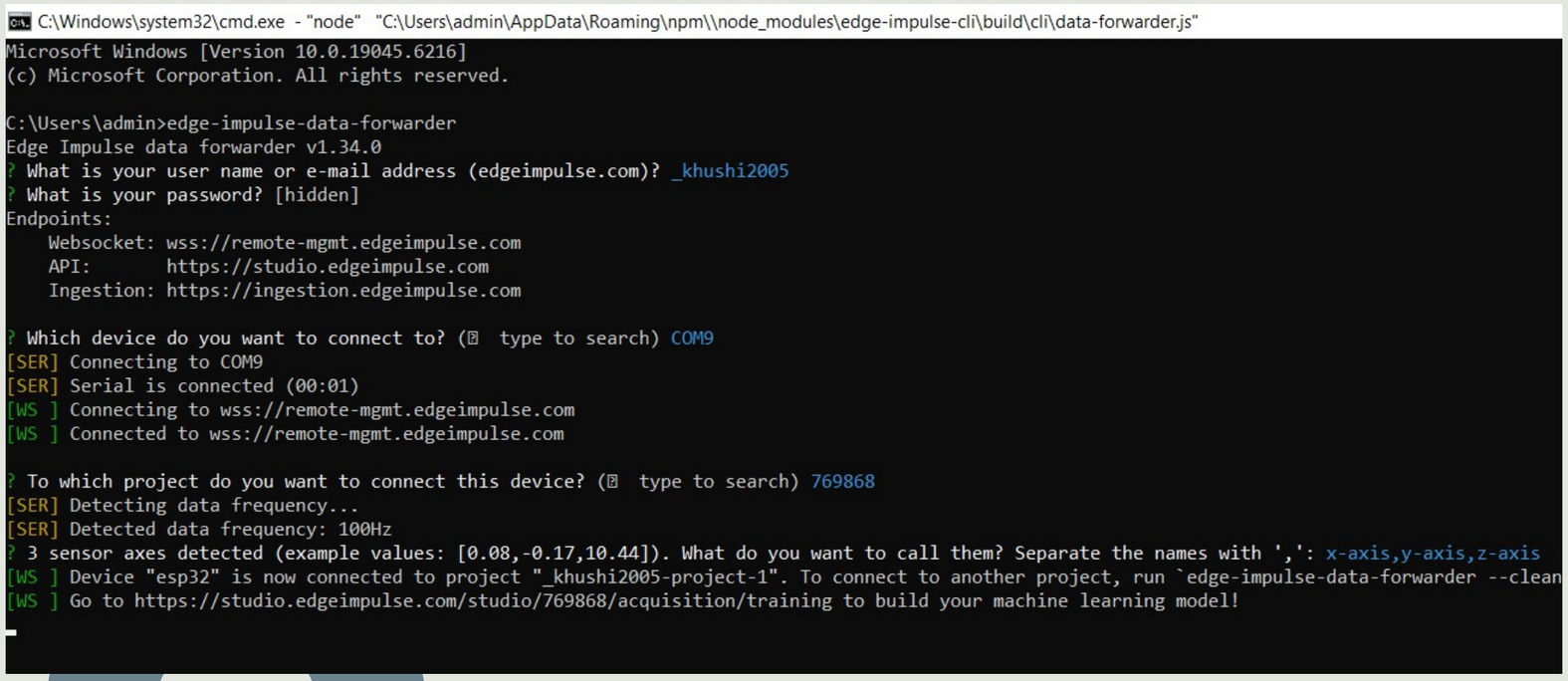
NPM INSTALL -G EDGE-IMPULSE-CLI --FORCE

THIS WILL SUCCESFULLY INSTALL THE EDGE IMPULSE CLI IN YOUR COMPUTER WHICH WILL HELP US TO FORWARD DATA TO EDGE IMPULSE

ENABLING DATA FORWARDER

NOW IN YOUR EDGE IMPULSE CREATE A NEW PROJECT. THEN OPEN THE COMMAND PROMPT AND ENTER

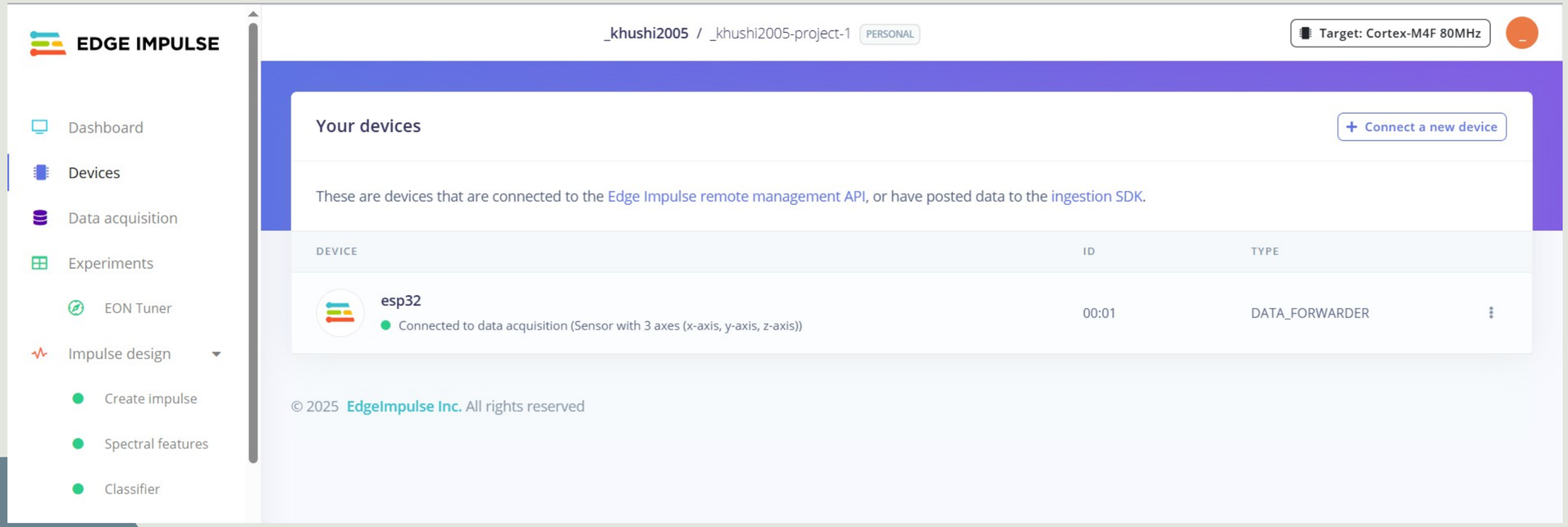
EDGE-IMPULSE-DATA-FORWARDER



ENTER YOUR ID,PASSWORD,NEW PROJECT THAT YOU ARE WORKING ON AND IF ASKED THEN THE COM PORT IN WHICH ESP IS CONNECTED

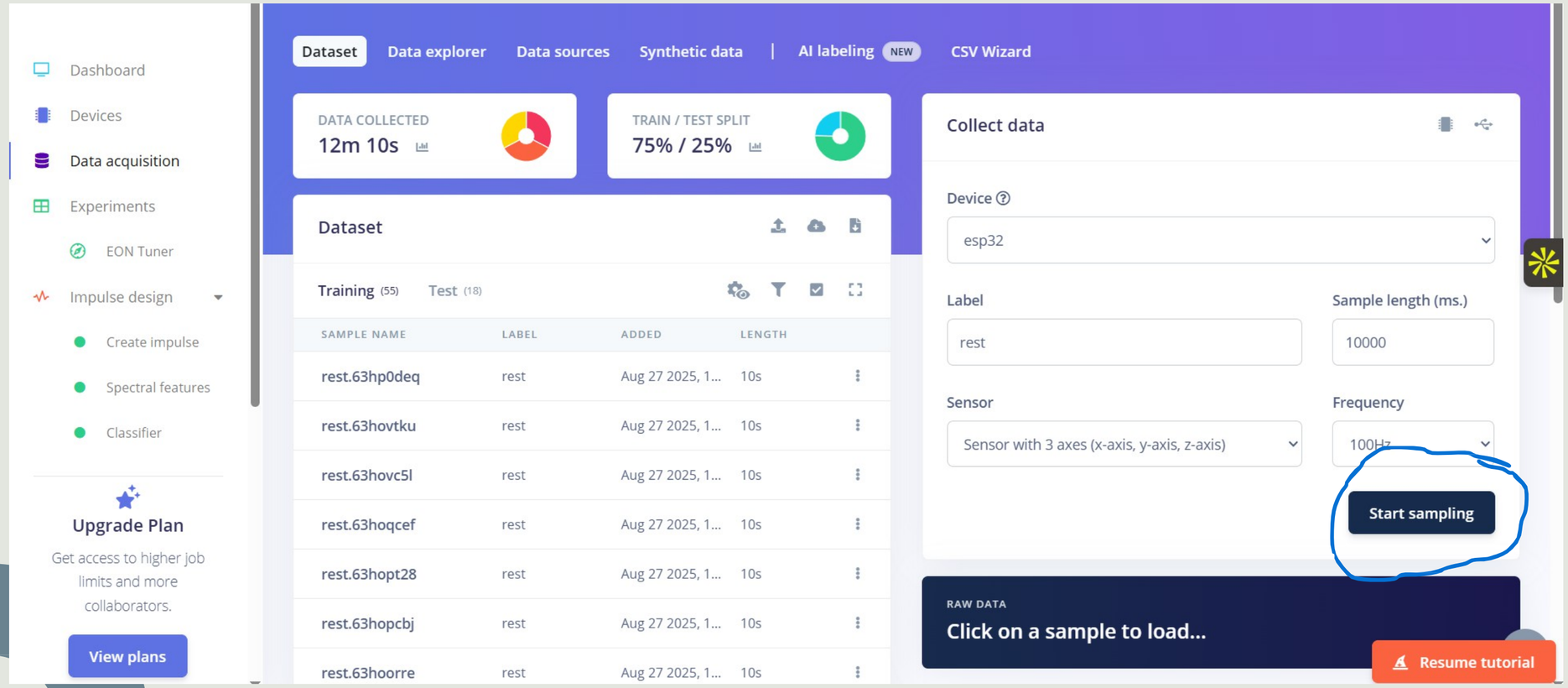
CONNECTED DEVICE IN EDGE IMPULSE

OPEN YOUR EDGE IMPULSE AND GO TO THE DEVICES OPTION. THERE YOU WILL SEE YOUR DEVICE NAME AND A GREEN DOT WITH IT



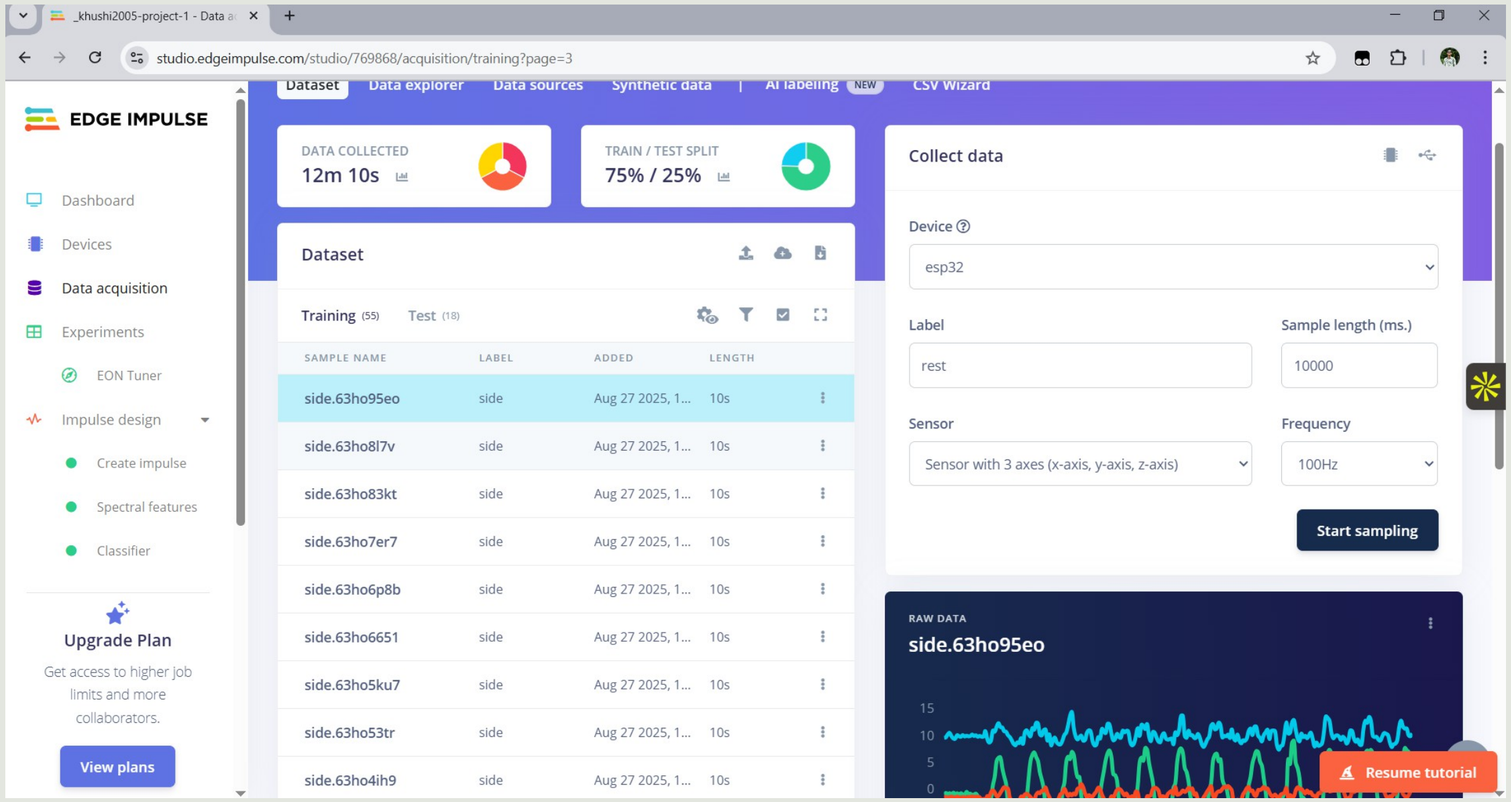
DATA ACQUISITION

GO TO THE DATA ACQUISITION OPTION. IN THE COLLECT DATA WINDOW, YOUR DEVICE NAME WILL BE SHOWN AUTOMATICALLY. SET THE SAMPLE LENGTH AS 10000 AND ENTER A LABEL FOR ANY GESTURE AND THEN PERFORM THAT GESTURE TO RECORD THE DATA.



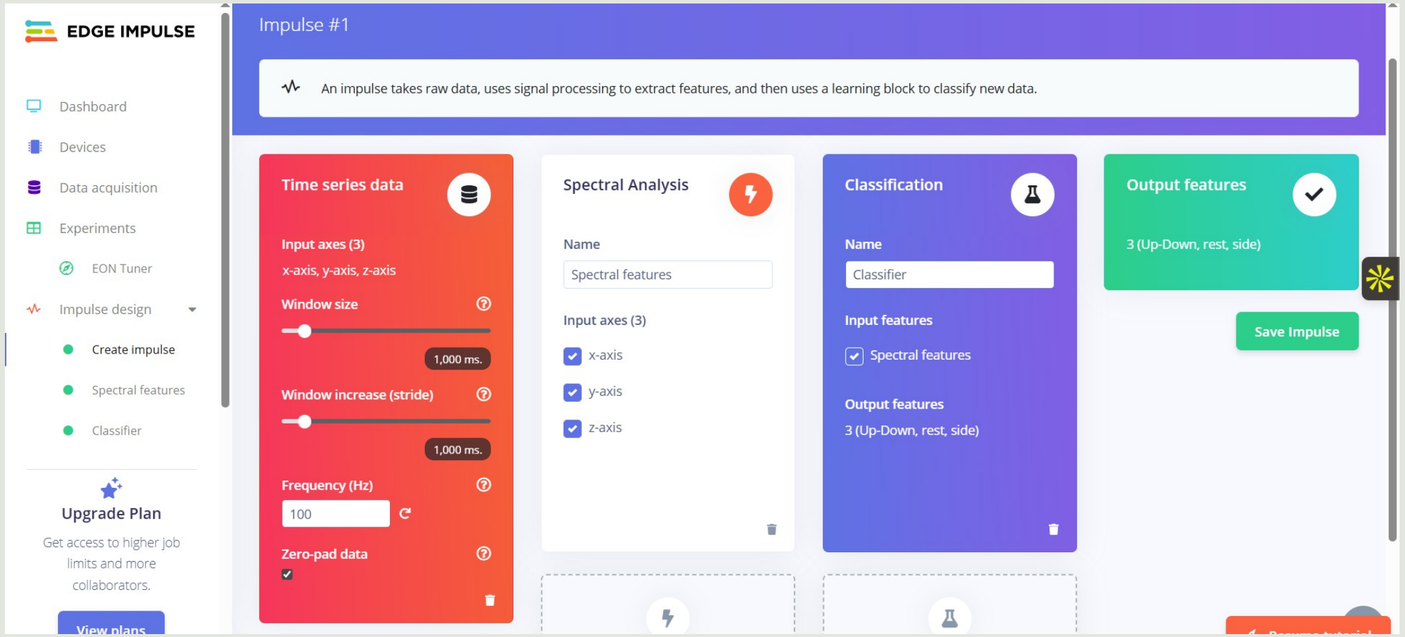
DATA ACQUISITION

PROPERLY LABEL ALL THE DATA AND CREATE ATLEAST THREE CLASSES SUCH AS UP-DOWN,SIDE AND REST. ENSURE THAT MINIMUM 3 MINUTES OF DATA IS PRESENT IN EACH CLASS. ALSO ENTER SOME DATA IN THE TEST WINDOW. THE DATA SPLIT FOR EACH CLASS SHOULD BE MINIMUM 80/20 FOR TRAIN/TEST SPLIT



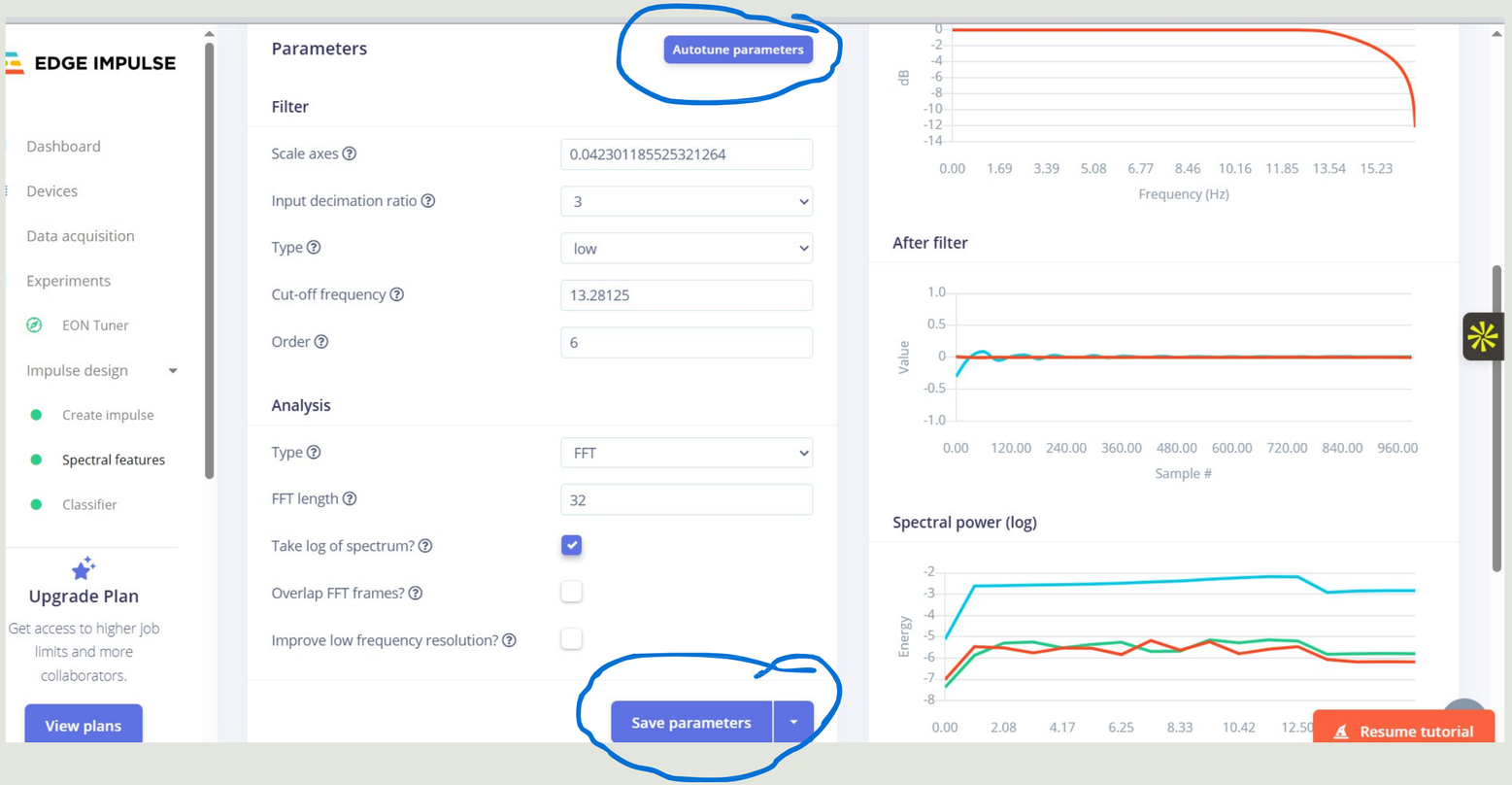
CREATING IMPULSE BLOCK

AFTER ALL THE DATA IS COLLECTED,CLICK ON THE CREATE NEW IMPULSE OPTION. CLICK ON THE ADD A PROCESSING BLOCK AND ADD ‘SPECTRAL FEATURES’ AS THE PROCESSING BLOCK. THEN CLICK ON ADD LEARNING BLOCK AND ADD ‘CLASSIFIER’ AS THE LEARNING BLOCK. THEN JUST SAVE THE IMPULSE



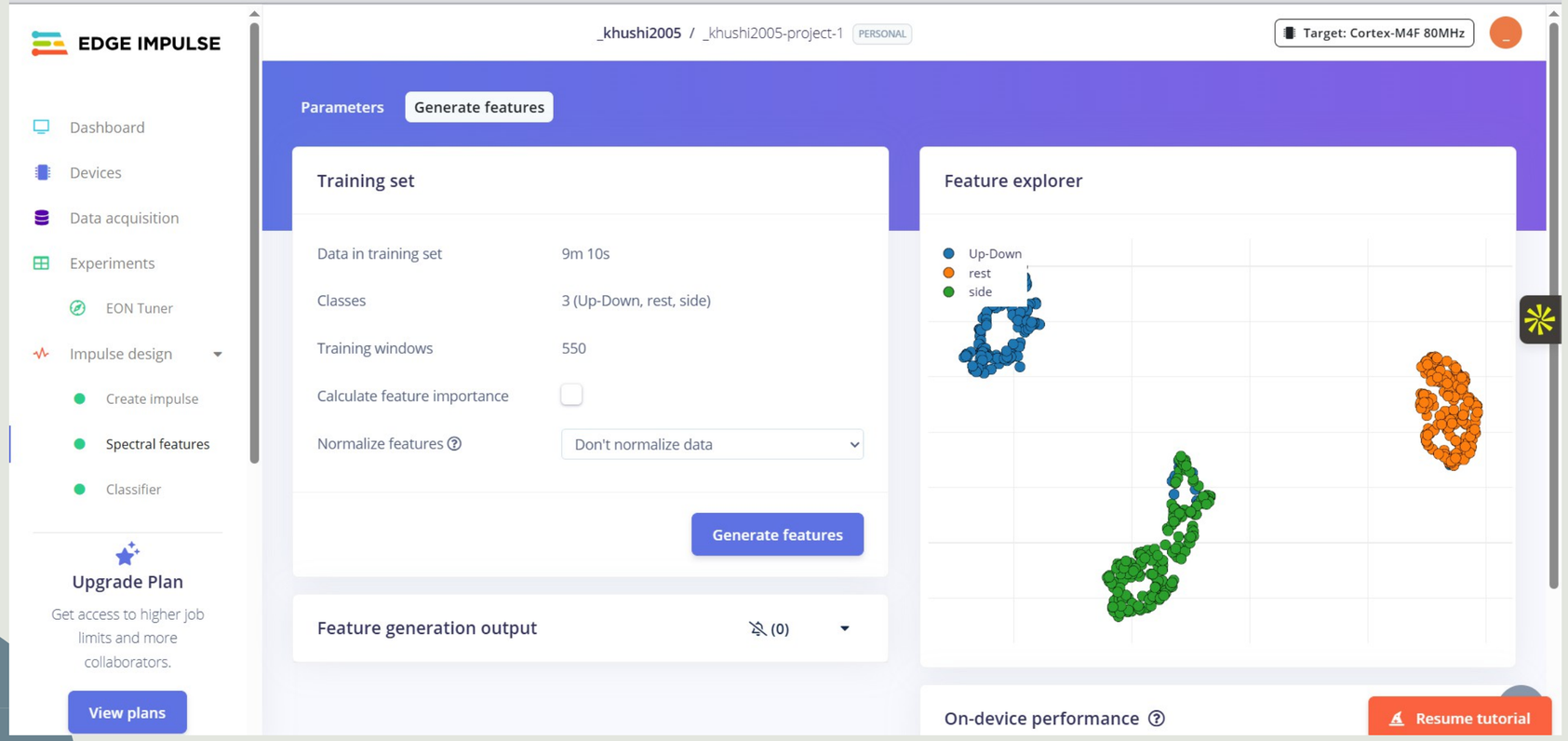
SPECTRAL ANALYSIS AND FEATURE GENERATION

CLICK ON THE SPECTRAL FEATURES OPTION AND CLICK ON THE AUTOTUNE PARAMETERS. AFTER THE PARAMETERS ARE SET, JUST SAVE THE PARAMETERS.



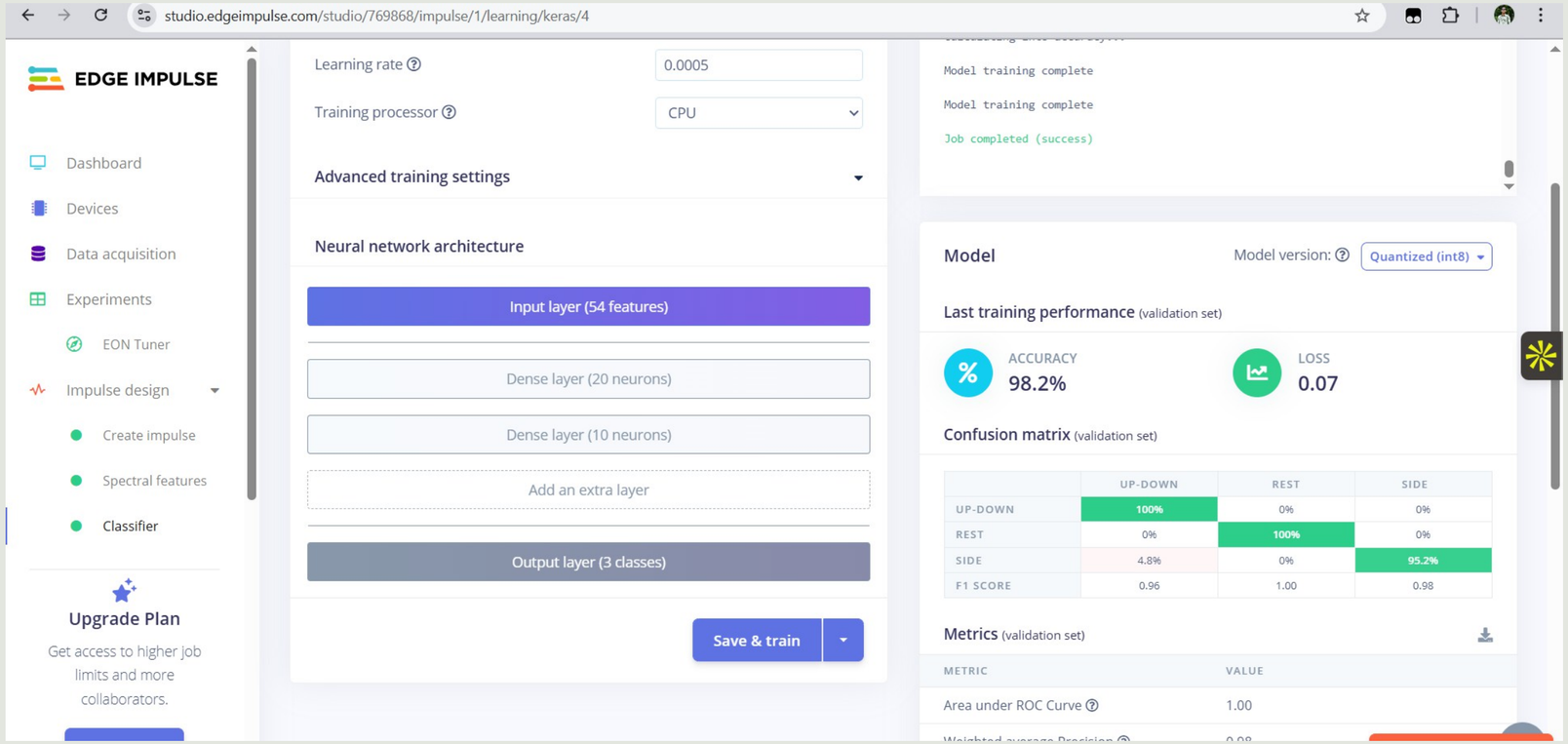
SPECTRAL ANALYSIS AND FEATURE GENERATION

YOU WILL BE REDIRECTED TO THE FEATURE GENERATION WINDOW WHERE YOU WILL JUST VERIFY YOUR DATA PARAMETERS AND GENERATE THE FEATURES OF YOUR DATA.



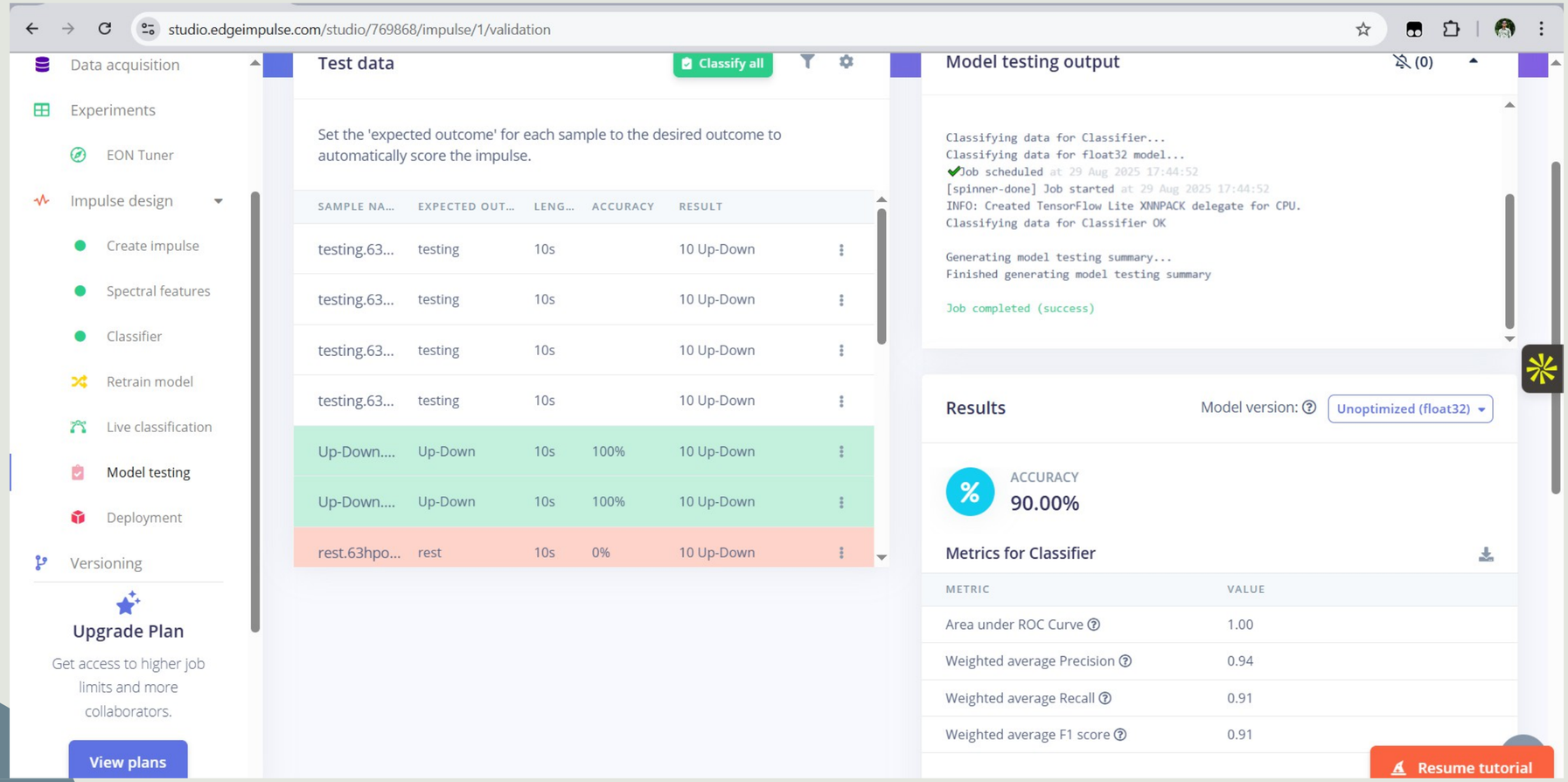
TRAINING OF THE MODEL

GO TO THE TRANSFER LEARNING OPTION AND SET THE NO. OF TRAINING CYCLES AS 30 AND THEN CLICK ON SAVE AND TRAIN. AFTER SOME TIME THE TRAINING STATISTICS WILL BE SHOWN WHICH CAN BE ANALYSED



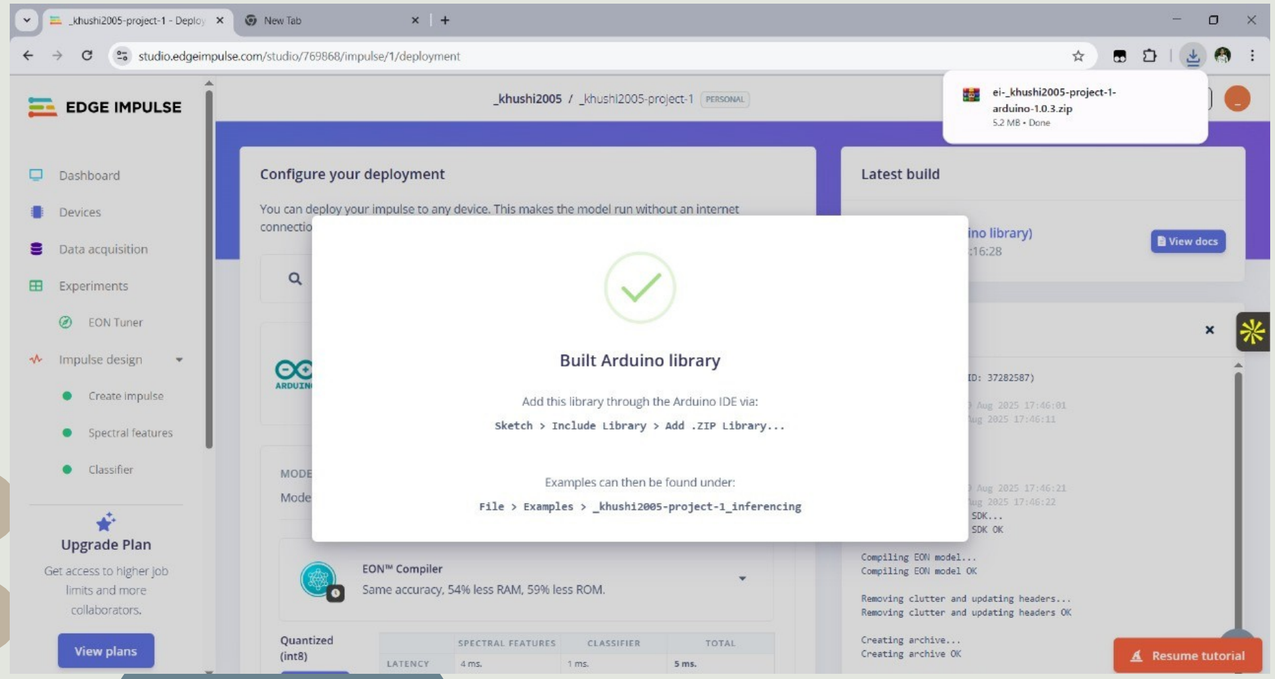
TESTING OF THE MODEL

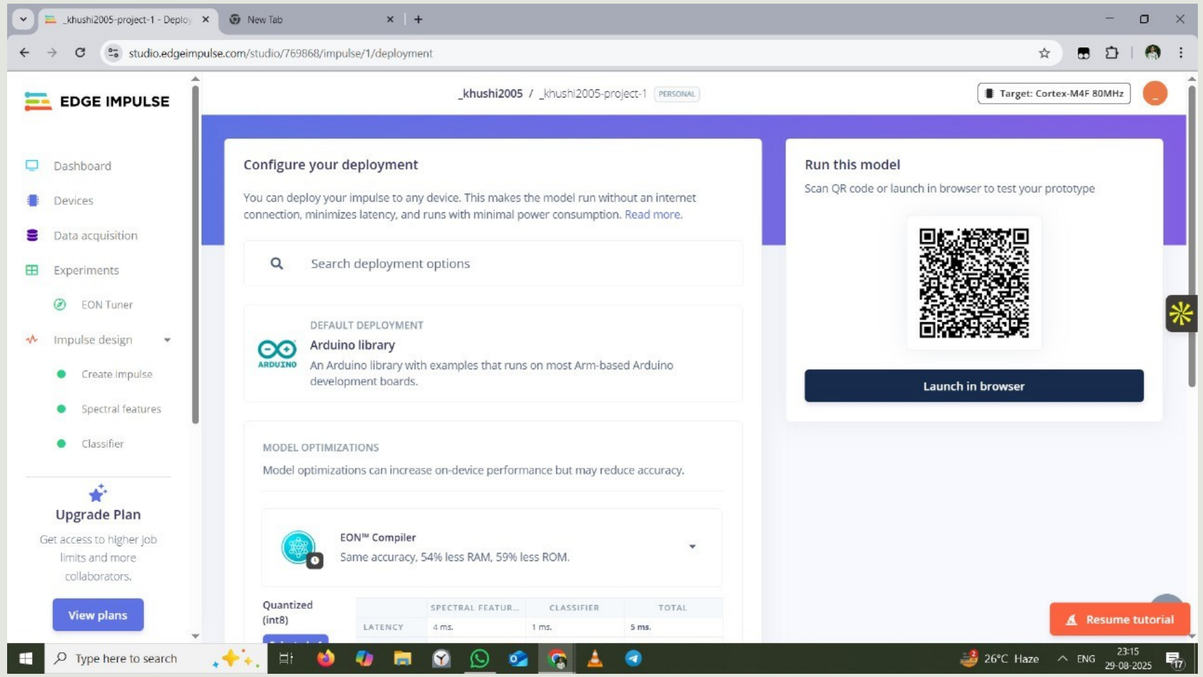
YOU CAN CLICK ON THE MODEL TESTING OPTION TO TEST THE MODEL PREDICTIONS AND ITS ACCURACY



DEPLOYMENT OF MODEL

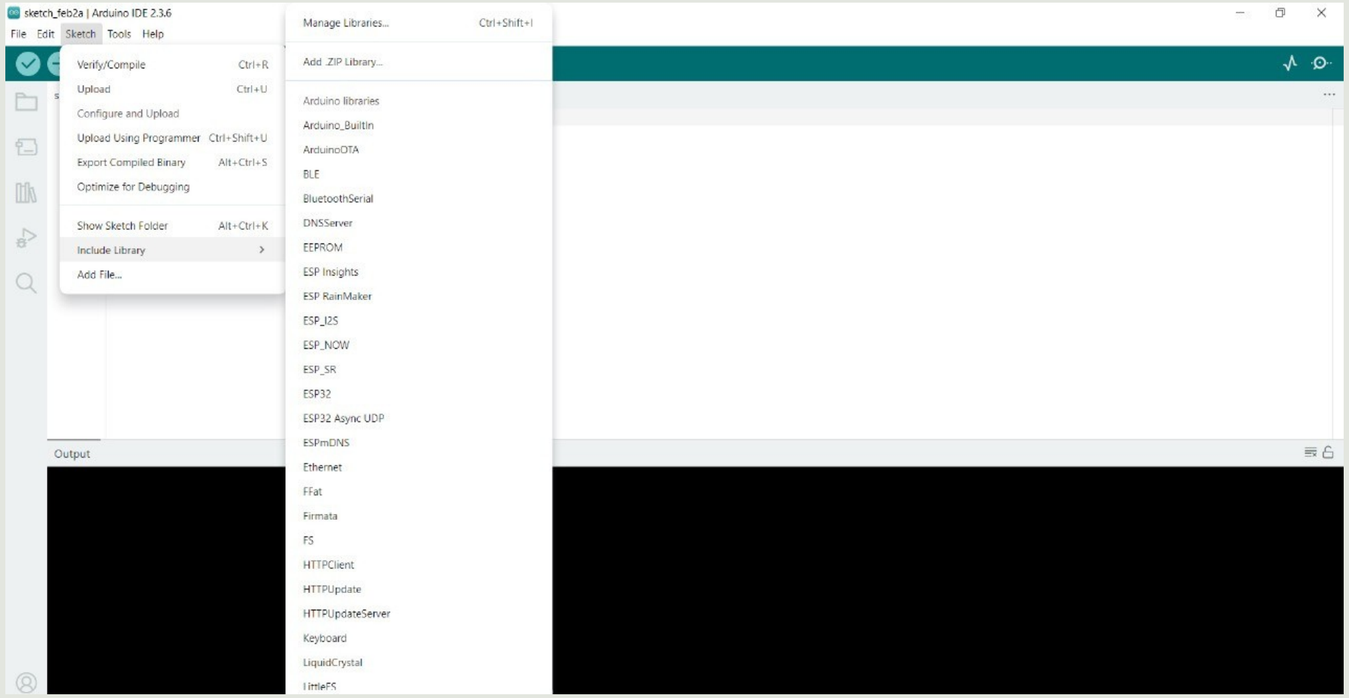
DEPLOY THE MODEL USING THE MODEL DEPLOYMENT TAB AND SELECT THE DEFAULT DEPLOYMENT AS ARDUINO LIBRARY, WHICH WILL DOWNLOAD A LIBRARY FILE INTO YOUR LAPTOP





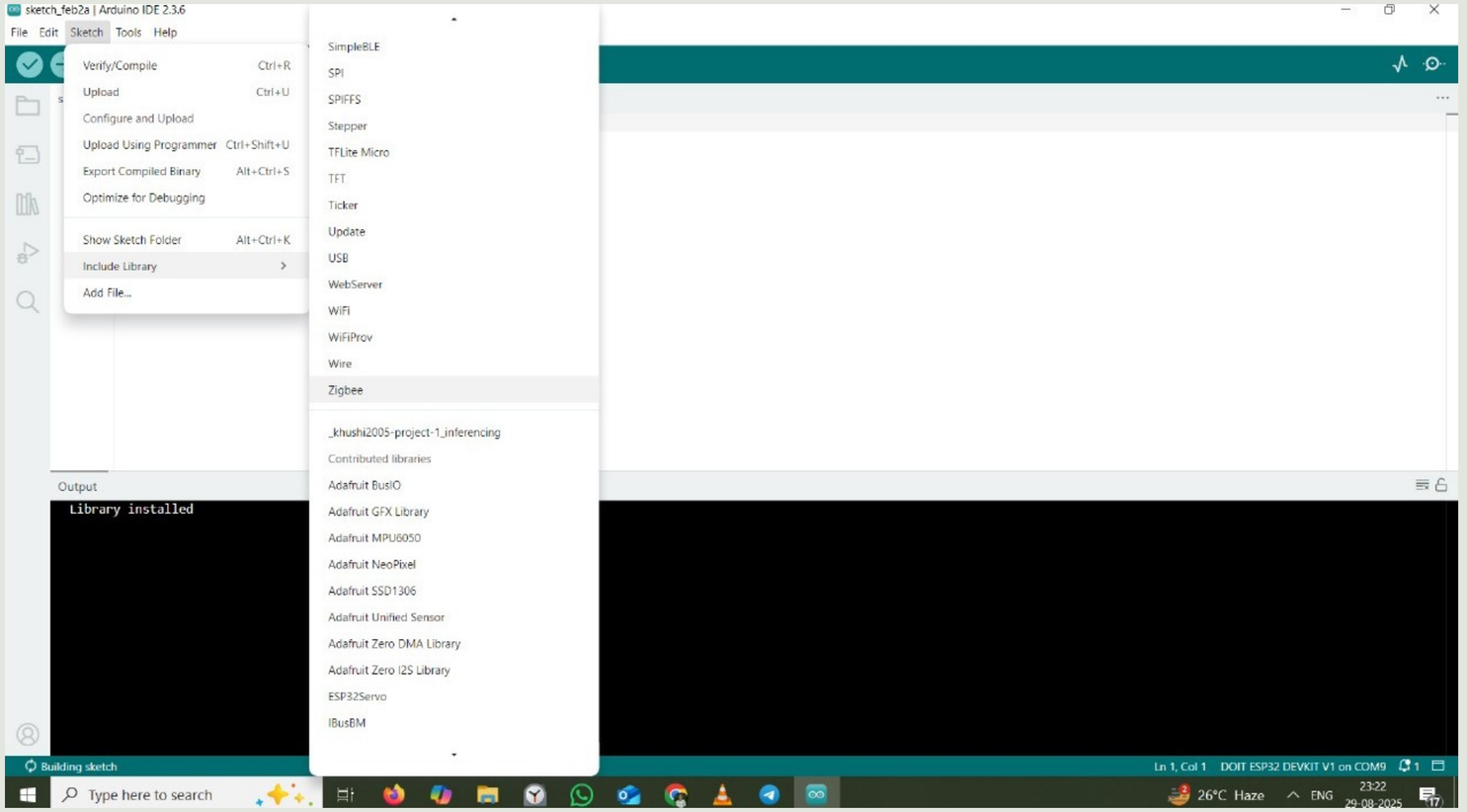
LIBRARY INSTALLATION IN ARDUINO IDE

IN ARDUINO IDE, CLICK ON SKETCH,THEN INCLUDE LIBRARIES AND ON THE TOP, CLICK ON ADD .ZIP LIBRARY AND SELECT YOUR DOWNLOADED ZIP FILE. THIS WILL INSTALL THE MODEL LIBRARY ON YOUR ARDUINO IDE.



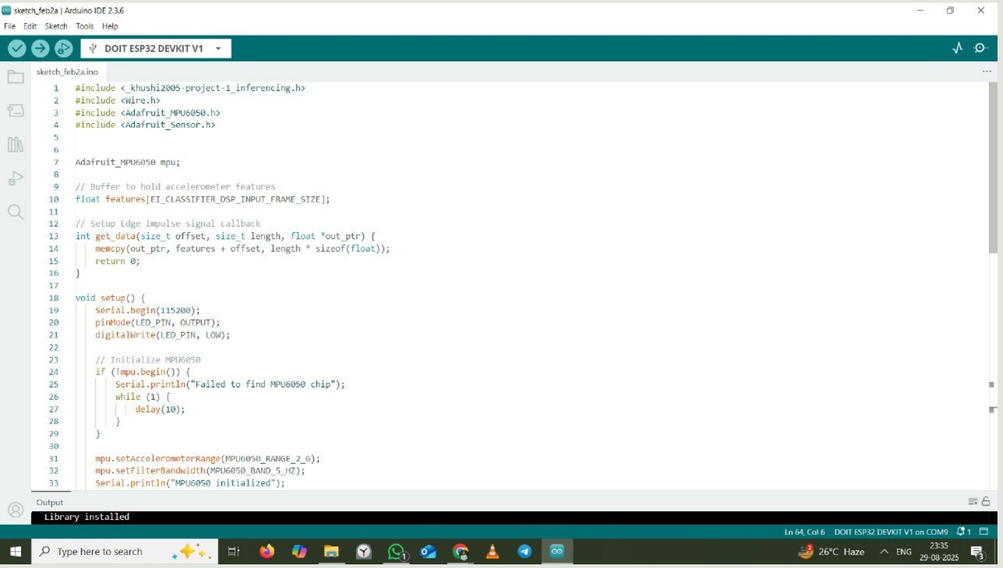
USING THE MODEL

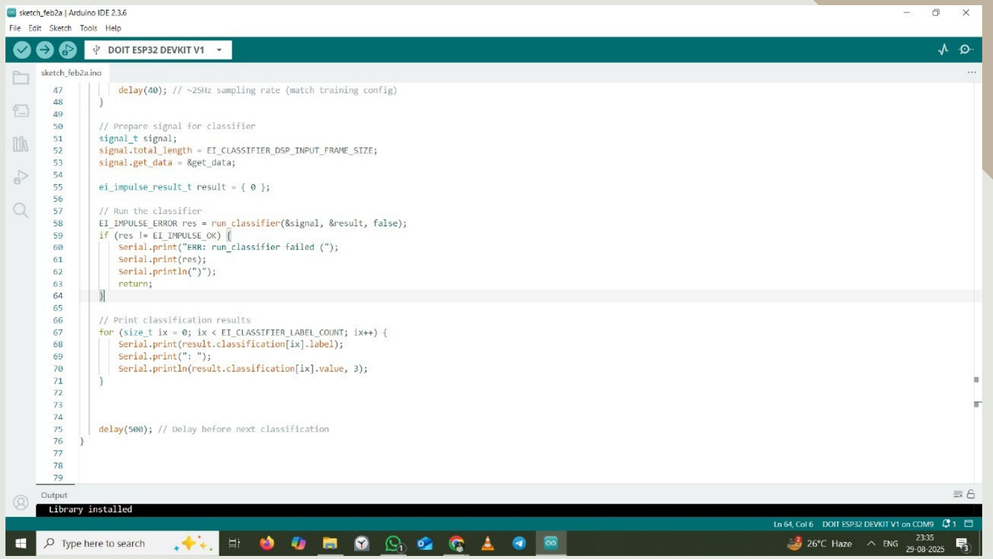
AFTER YOUR LIBRARY IS INSTALLED, ONCE AGAIN GO TO SKETCH AND INCLUDE LIBRARY. THEN CLICK ON YOUR MODEL LIBRARY.

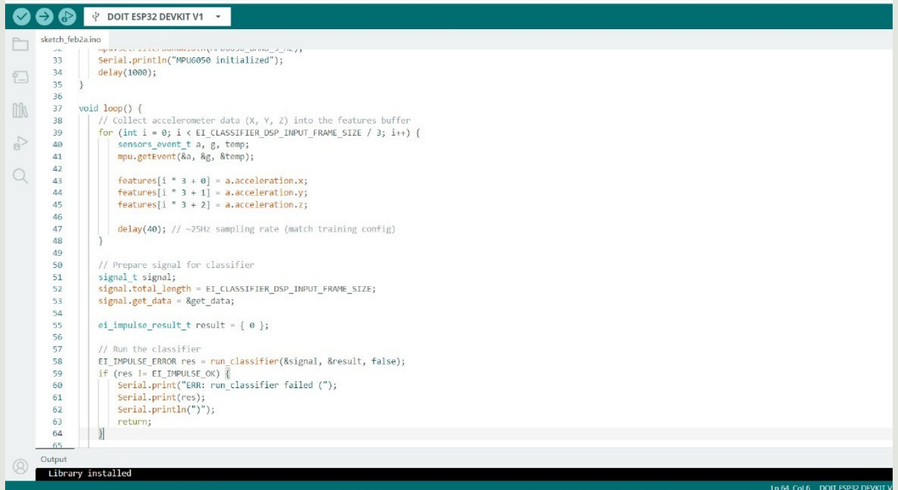


FURTHER CODE AND USE

WRITE THIS CODE WITH YOUR LIBRARY. THIS CODE WILL EXTRACT DATA FROM YOUR MPU6050 AND THEN CLASSIFY IT AND PREDICT THE GESTURE ON THE SERIAL MONITOR







THANK YOU!!