

FINAL CODE:

```
#include <Arduino.h>
#include <WiFi.h>
#include "Config.h"
#include "FunctionRegistry.h"
#include "AIController.h"
#include "pitches.h"
#include <Temperature_LM75_Derived.h>

int sensorPin = 1;
int sensorValue;
bool pressed = false; // Keeps track of whether the sensor is currently pressed
int threshold = 500; // Adjust this threshold based on your sensor

// Global objects
WiFiClientSecure client;
FunctionRegistry funcRegistry;
AIController* aiController;
TI_TMP102 temperature;

String read_touch_sensor() {
    sensorValue = analogRead(sensorPin);

    if (sensorValue > threshold && !pressed) {
        // Sensor is pressed, and it wasn't pressed before
        pressed = true;

        Serial.print("Pressure: ");
        Serial.print(sensorValue);
        Serial.print(" -> ");

        // Assigning the value to one of the prompts
        if (sensorValue >= 500 && sensorValue <= 1000) {
            return "create a new material that is lightweight, transparent and flexible";
        } else if (sensorValue >= 1001 && sensorValue <= 2000) {
            return "create a new material that is spongy, bubbly and colorfull";
        } else if (sensorValue >= 2001 && sensorValue <= 3000) {
            return "create a new material that is dense, opaque and rigid";
        } else if (sensorValue >= 3001 && sensorValue <= 4069) {
```

```

        return "create a new material that is strong, reflective and alive";
    } else if (sensorValue >= 4070 && sensorValue <= 4095) {
        return "create a new material that has random features";
    } else {
        return "invalid"; // Just in case
    }
}

// Reset the flag when pressure is released
if (sensorValue < threshold && pressed) {
    pressed = false; // Ready for the next press
}

return "waiting";
}

void setupWiFi() {
    WiFi.mode(WIFI_STA);
    Serial.printf("Connecting to %s\n", WiFiConfig::SSID);
    WiFi.begin(WiFiConfig::SSID, WiFiConfig::PASSWORD);

    while (WiFi.status() != WL_CONNECTED) {
        Serial.print(".");
        delay(500);
    }

    Serial.printf("\nConnected! IP: %s\n",
WiFi.localIP().toString().c_str());
}

void setup() {
    Wire.begin();
    Serial.begin(115200);

    // Setup network
    setupWiFi();
    client.setInsecure();

    // Init LED
    pinMode(PinConfig::ONBOARD_LED, OUTPUT);
    // Init photo transistor
    pinMode(PinConfig::PHOTO_TRANSISTOR, INPUT);

```

```

// Setup function registry
// funcRegistry.attachFunction("TURN_ON_LED", turn_on_led);
// funcRegistry.attachFunction("TURN_OFF_LED", turn_off_led);
// funcRegistry.attachFunction("PLAY_MELODY", play_melody);

// Initialize AI controller
aiController = new AIController(client);
}

void loop() {
    // Capture sensor data here
    String touch_read = read_touch_sensor();
    if(touch_read != "waiting" && touch_read != "invalid") {
        String inputData = "Material: " + touch_read;
        Serial.printf("Input data: %s\n", inputData.c_str());
        // Get AI response
        String result;

        if (aiController->processTextData(inputData,
funcRegistry.getBULLETList(), result)) {
            Serial.printf("AI Response: %s\n", result.c_str());
        } else {
            Serial.printf("AI Error: %s\n", result.c_str());
        }
        delay(5000);
    }
    delay(50);
}

```