

Team Member: Kitty Li (server), Peizhong Gao (client)

1. The link to your edited code is on GitHub. You can either create a new repo or push the updates to your existing repo for all labs (recommended)

Kitty Li:

[https://github.com/xinyili7/514\\_Lab\\_Four\\_Bluetooth\\_Service.git](https://github.com/xinyili7/514_Lab_Four_Bluetooth_Service.git)

Peizhong Gao: [https://github.com/GPZ12138/TECHIN514\\_labs/tree/main/lab4](https://github.com/GPZ12138/TECHIN514_labs/tree/main/lab4)

2. Screenshot of your serial monitor displaying the number of Bluetooth devices detected using your MCU as BLEScanner

The screenshot shows the PlatformIO IDE interface with the following details:

- EXPLORER**: Shows the project structure: UNTITLED (WORKSPACE) > 514\_Lab\_Four > src > main.cpp.
- PROBLEMS**: No errors or warnings.
- OUTPUT**: Displays the serial monitor output of the BLEScanner program, listing multiple Bluetooth devices found by the MCU. The output includes device names, addresses, and RSSI values.
- TERMINAL**: No content.
- PORTS**: No content.
- CHAT**: A message from Claude Haiku: "what is theb error?" and a response: "Read and checked main.cpp file. The errors are missing includes. Your code uses Serial and delay() functions, but these require the Arduino header to be included. Add #include <Arduino.h> at the top of the file."
- STATUS BAR**: Shows the current file is main.cpp, line 36, column 2, and the build status: 1 file changed +1 -0.

3. Screenshot of the serial monitor of your client device to show successful connection with your server device

The screenshot shows the Visual Studio Code interface with the following details:

- Explorer View:** Shows the project structure under "UNTITLED (WORKSPACE) / 514\_Lab\_Four".
- Editor View:** The main editor window displays `main.cpp` code related to BLE advertising. A red error message at the bottom right states: "Characteristic defined! Now you can read it in your application".
- Terminal View:** The terminal window shows repeated notifications: "Notify value: Kitty's MCU".
- Bottom Status Bar:** Shows file name "Default (514\_Lab\_Four)", line number "Ln 19, Col 9 (58 selected)", and workspace "PlatformIO".
- Bottom Right Panel:** A "CHAT" panel titled "IDENTIFYING THE ERROR IN CODE" contains a message: "Used 1 reference" and "Displayed server name in client's terminal output". It also includes a note about modifying the server side to "Hello World".
- Bottom Right Buttons:** Includes icons for "Edit", "Run", "Build", and "Upload".

## Server: Kitty's MCU

#### 4. Screenshot of the serial monitor of your server device to show the raw and denoised sensor data.

```

BLEServer* pServer = NULL;
BLECharacteristic* pCharacteristic = NULL;
bool deviceConnected = false;
bool oldDeviceConnected = false;
unsigned long previousMillis = 0;
const long interval = 1000;

// HC-SR04 Pins (for XIAO ESP32S3)
#define TRIG_PIN D5 // D0/GPIO5
#define ECHO_PIN D4 // D1/GPIO4

// DSP Filter Parameters
const int FILTER_SIZE = 5;
float filterBuffer[FILTER_SIZE] = {0};
int filterIndex = 0;

// Distance threshold for transmission
const float DISTANCE_THRESHOLD = 30.0;

```

The screenshot shows the PlatformIO IDE interface with the main.cpp file open. The terminal window displays the output of the task "platformio device monitor". It shows raw distance measurements (Raw) and denoised distance measurements (Denoised) for each of the five samples taken by the filter. The raw values fluctuate between 17.97 cm and 18.30 cm, while the denoised values are much more stable, ranging from 18.04 cm to 18.11 cm. A note in the top right corner provides a technical explanation about increasing the pulseIn() timeout from 30000 to 38000 microseconds to account for the sensor's needs to receive an echo.

#### 5. Screenshot of the serial monitor of your client device to show the currently-reading, maximum, and minimum data transmitted from your server device.

```

void setup() {
  if (connect()) {
    _isConnected = true;
  }
}

// Client side only receives notifications for this lab.

if (_isConnected && doScan()) {
  BLDevice::getScan()->start();
}

delay(50);

```

The screenshot shows the PlatformIO IDE interface with the main.cpp file open for a project named "xiao-bluetooth-client". The terminal window displays the output of the task "platformio device monitor". It shows a series of distance measurements being transmitted from the server to the client. The distances start at 6.27 cm and increase in increments of approximately 0.2 cm up to 28.19 cm. The client side of the code is shown, which includes a connection check and a scan initiation.