Journal 6

Wyatt Marvil Per. 5

21 October 2019

1 Progress Log

Discovered that I had been compiling with API 15, which does not support Bluetooth Low Energy (BLE) functionality. Upgraded to the maximum supported API level of the testing device (currently still Samsung Galaxy S6), which is API 21.

Wrote a callback method to scan for BLE devices. The callback is passed to the system's Bluetooth adapter with the Android API's stopLeScan and startLeScan methods.

Created a DeviceScanner class that contains a method scanLeDevice(boolean enable) to enable and disable Bluetooth scanning. It utilizes a Handler object to synchronously run the BLE scan callback method. The Handler will allow the callback to run for ten seconds before stopping the scan. This is important because scanning is a large burden on bandwidth. If it were not properly stopped, data transmission would be difficult and slow.

Instantiated a DeviceScanner object in the MainActivity's onActivity method. It will initiate a BLE scan on application startup and populate an onscreen device list with the discovered devices. It is also referenced in the "Scan" button's event listener to initiate a BLE scan on button press.

Added a visual cue for when the device is scanning and for when the scan is complete.

Spoke with Mr. Kosek to resolve the application crashing on startup. The issue was an incorrect instatiation of the DeviceScanner's Handler.

Set up a logging system with Android's built in logging API.

2 Next Steps

Implement the functionality to connect to a discovered BLE device. Create a client-server relationship where the user's device is a client querying data from the BLE module which acts as a server.

Receive data (in hexadecimal) from the BLE module and convert it to a plaintext CSV file. Display the CSV within the app (possibly update it live as new data is received).

Give user the option to store the CSV on the device or email it to themselves.

Upgrade to a Samsung Galaxy S7

3 Reflection

A lot of progress has been made, but this application is still behind schedule. As a group we need to also build the stick that will house the gyroscope accelerometers. Another point of interest is that, given that the stick is a strict unbending length, the gyroscope data from all three sensors will be redundant. By sending gyroscope data from only one sensor, perhaps we can make more efficient usage of the BLE bandwidth and also process the resultant data faster.