Project Task 0: Proposal

Gabriel Frank, Ryan Botwinick, Jacob Frankel

GitHub Repository

Section 1. Hypothetical Use Case

In modern autonomous vehicle systems, precise environmental awareness is essential for safe and efficient navigation. Our project proposes a distributed IoT-based sensor network that enables real-time obstacle detection and vehicle orientation tracking. The system consists of two Source Nodes:

- Node 1: Ultrasonic sensor to detect objects in the vehicle's path and measure distance (ESP32).
- Node 2: IMU sensor to track vehicle orientation and movement changes (Pi Pico).

These Source Nodes transmit data via XBee radios to a Sink Node, which aggregates the sensor data and uploads it to the ThingSpeak IoT platform. This implementation allows autonomous vehicle systems to analyze real-time data for navigation and collision avoidance. The Sink Node is a laptop with an XBee module that processes the received sensor data before forwarding it to the cloud.

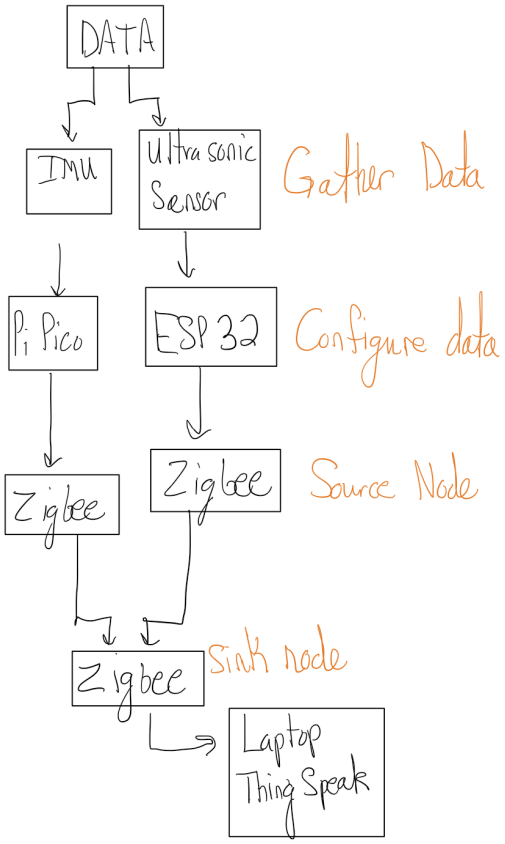

Figure 1: Block diagram of the information system

Section 2. Device, Sensor, and Platform Selection

This section should include figures that show the images of the discussed items.

A. Embedded Devices Selection:

- ESP32 for Node 1
- Raspberry Pi Pico for Node 2

B. Analog Sensors Identification:

- Ultrasonic Distance Sensor
- IMU (Gyroscope and Accelerometer)

C. Wi-Fi enabled Computing Platform:

- Laptop as the Sink Node, running Python scripts to receive data via XBee, process it, and upload to ThinkSpeak

Section 3. Budget

No parts will need to be purchased, as all parts are already accessible. However, a comprehensive list of components and prices are included for reference.

Component	Quantity	Unit Price
XBee	3	$30
Pi Pico	1	$5
ESP32	1	$5.26
IMU	1	$7.69
Ultrasonic Sensor	1	$1.91

Section 4. Team Management

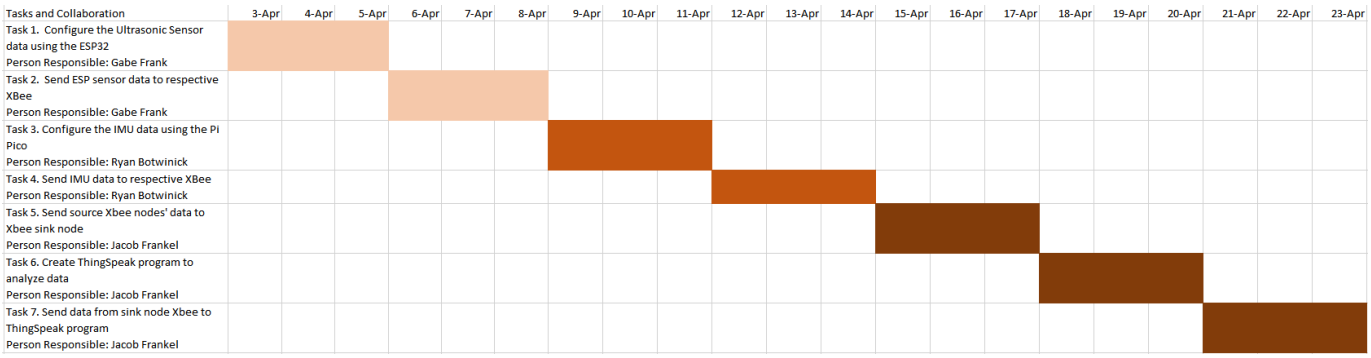

Figure 2: Gantt chart for group progress.

The workload division is as follows:

- Gabe - Configure the Ultrasonic Sensor data using the ESP32, and send it to the corresponding XBee source node.
- Ryan - Configure the IMU data using the Pi Pico, and send it to the corresponding XBee source node.
- Jacob - Ensure receiving of data from the Xbee sink node, create the ThingSpeak program for analysis, and send data from Xbee to ThingSpeak.