GPS tracking system

Link to vid: https://youtu.be/LdIpNR6gafM

Link to GitHub: https://github.com/Greekfreaks/GPS-tracker

Background:

This system intends to develop an embedded solution that uses GPS and WiFi technologies to notify users of the package's delivery status in real-time, in response to the growing demand for dependable package delivery tracking systems. In order to send automated email notifications when the box is shipped, gets close to its destination, and is delivered, this system interfaces with IFTTT webhooks.

Existing work:

Delivery vehicle GPS tracking is provided by a number of current options, although they frequently call for intricate setups and proprietary software. Our method makes it easier for small enterprises and do-it-yourselfers to implement by utilising readily accessible hardware components and open-source libraries.

Problem statement:

For small firms, the current package delivery tracking systems can be expensive and challenging to incorporate. A cost-effective and simple-to-use system that can instantly inform recipients of the status of their parcels without necessitating a high level of technical knowledge is needed.

Requirements:

This section includes both the hardware and software requirements.

The requirements to run this system on your own device is to have Arduino IDE installed with TinyGPS plus and WiFiNINA. The hardware components of this task require an Arduino Nano 33iot device (with a cable) connected to a NEO 6M GPS module. On top of this it needs an internet connection which can be modified in the secret.h file. The user also needs an IFTTT account where they would have made 3 webhooks which all send emails to the user with the necessary value1, value2 and value3 variables.

Design principles:

The following guidelines inform the system's design:

- Modularity: The use of parts that are easily assembled and replaced, such as GPS and WiFi modules.
- Simplicity: Easy setup with little configuration needed for non-technical users.
- Dependability: Accurate and timely alerts are ensured by robust management of GPS data and WiFi connectivity.
- Scalability: The solution's capacity to grow to accommodate more parcels or more extensive delivery activities.

User manual:

Once all the hardware and software requirements from above have been implemented, you can run the code (with your own wifi SSID and password) which will get the program, to out the message “connected to Wifi”. This allows the Arduino to be connected to wifi before running the program to ensure the emails can be sent. Next, type “Send” into the serial monitor to start the delivery process, from here the code will take care of the rest by constantly updating the location and sending emails depending on the location. *Change the float DestLong and DestLat to change the destination location for the package*.

Conclusion:

This project gave participants practical experience creating an embedded device that combines WiFi and GPS technology with cloud services. Ensuring dependable WiFi access and precise GPS data processing in diverse settings posed a significant challenge. If I had the chance again, I would look into more sophisticated GPS modules with improved signal capture and incorporate more error-handling techniques to increase the robustness of the system.
The next issue I faced was implementing a URL into the emails which provides the real time location of the GPS. It was easy enough to add the coordinates into the email, but adding the data directly into google maps was a little confusing. If I was to do this task again I would implement this straight into a geocoding API where it can take the longitude and latitude data and get back a street address from this to allow the User to have a better understanding of where the package is.