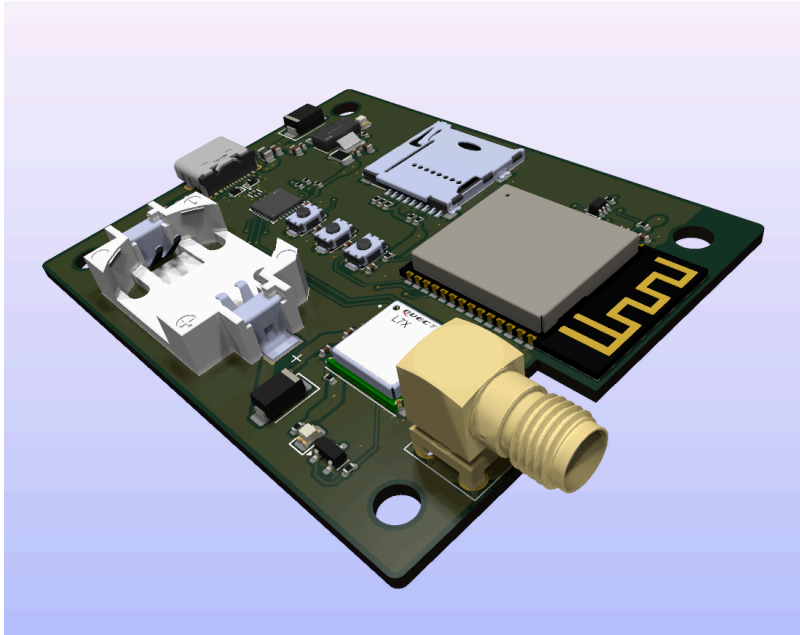


Jake Odgers

Mileage Logger Project



Scope

To log and display mileage for business purposes. With the ability for switching from personal journey to business journey.

Technical Design Skills

- RF waveguide design with a GCPW (Grounded Coplanar Waveguide)
- Differential pair routing for USB C datalines
- EMC and transient considerations
 - TVS (Transient Voltage Suppression)
 - Common mode chock
 - 4 layer board design with ground pours and power plane

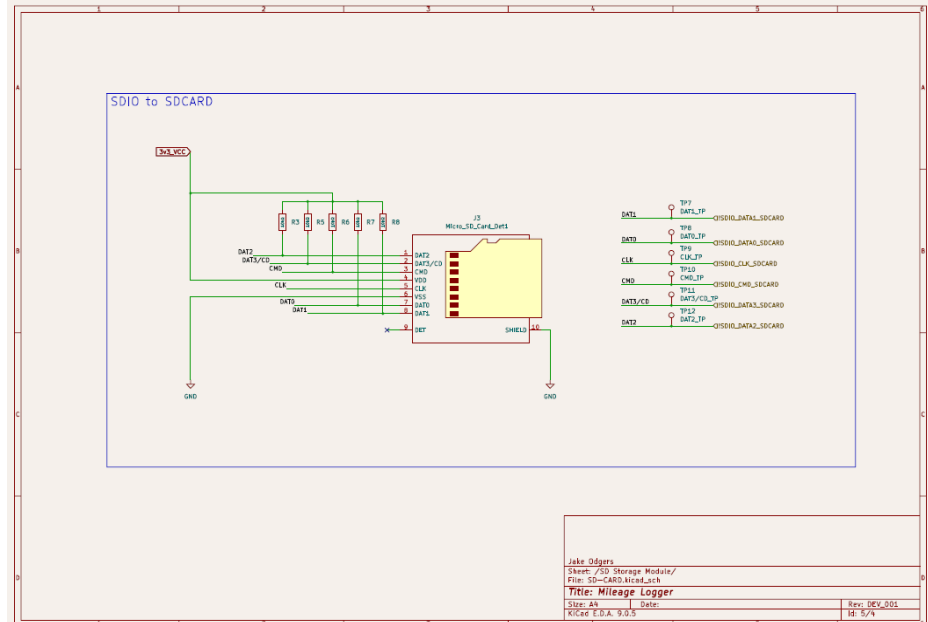
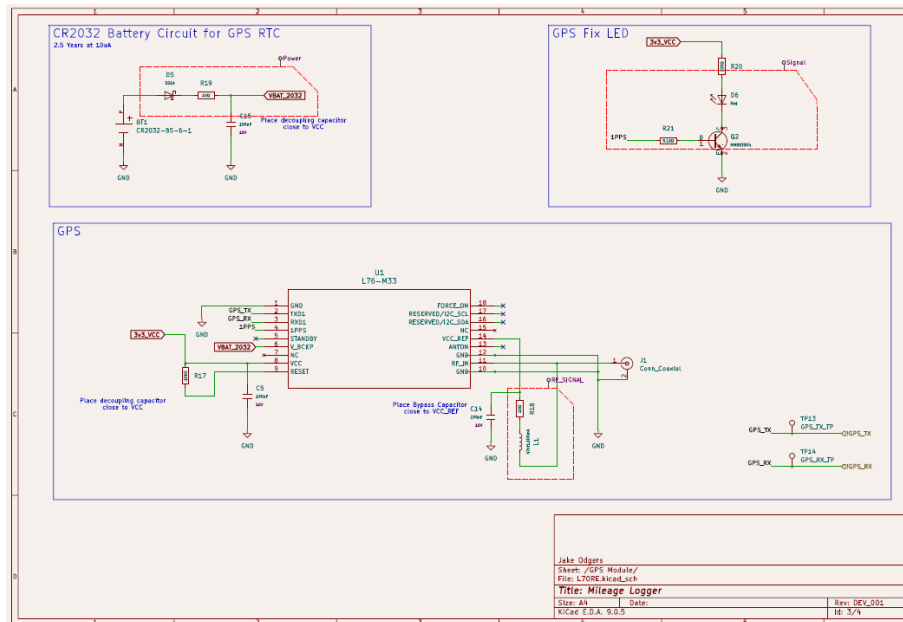
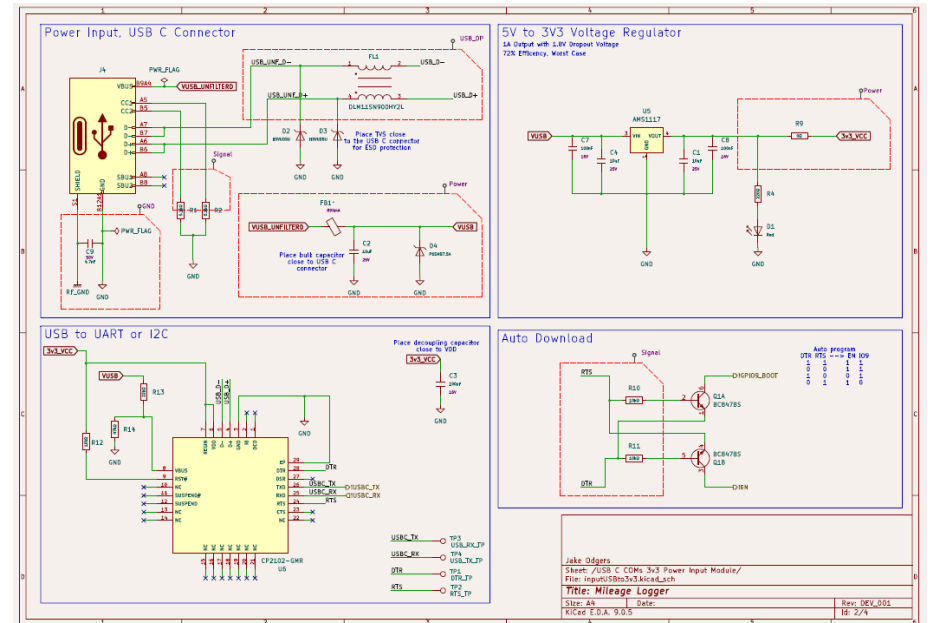
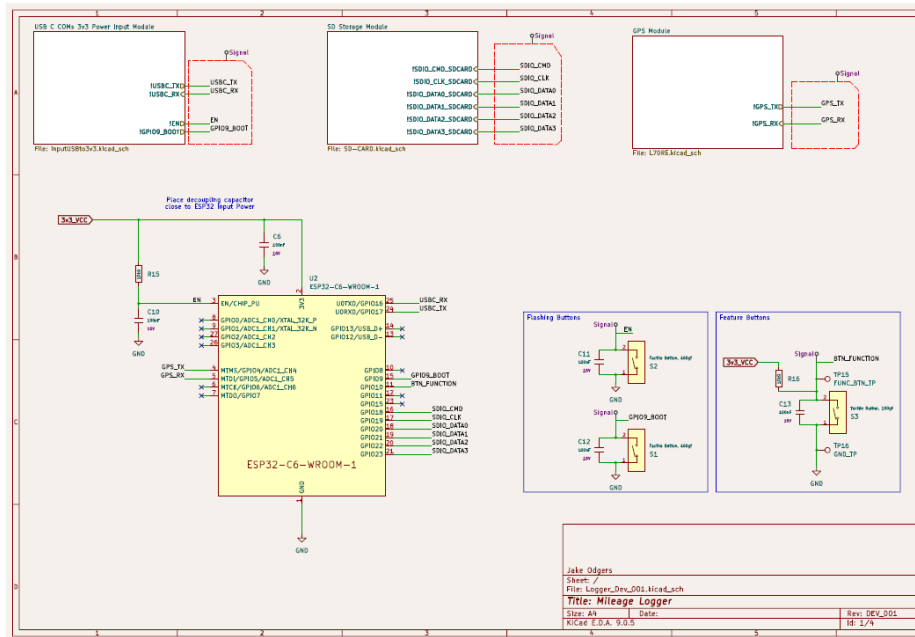
Display

Option 1 - Display through a GUI on a computer made through Python Tkinter.

Jake Odgers

Mileage Logger Project

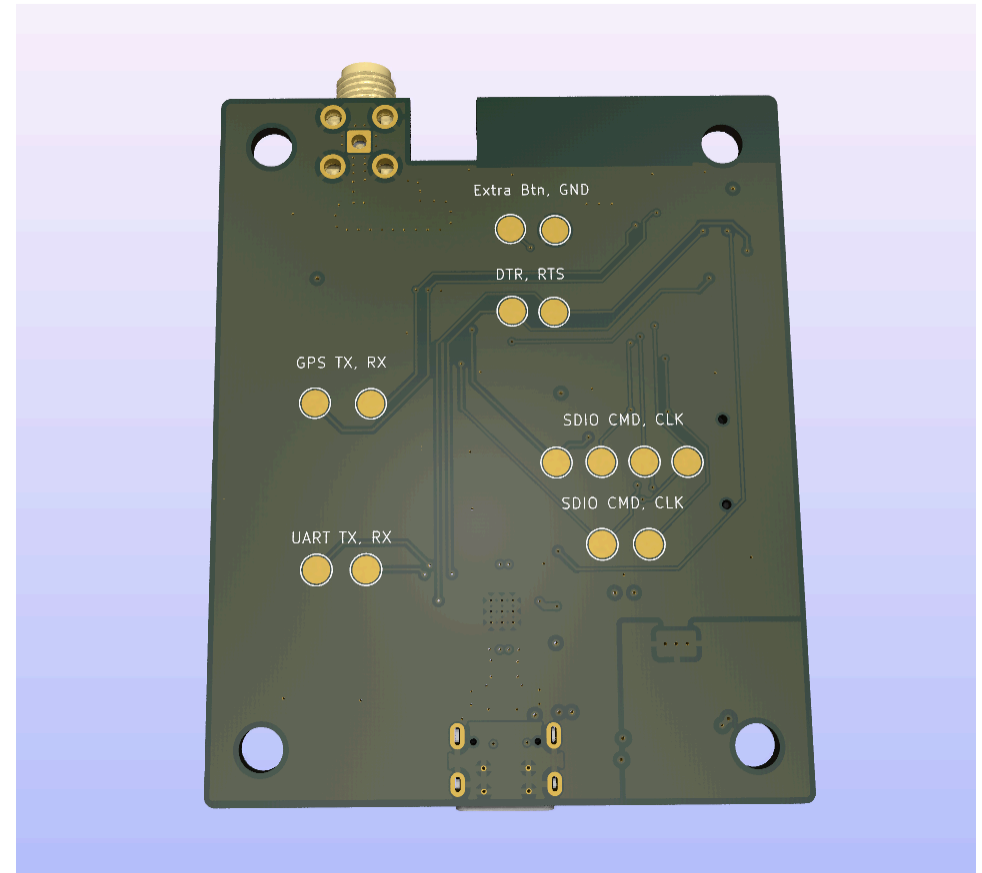
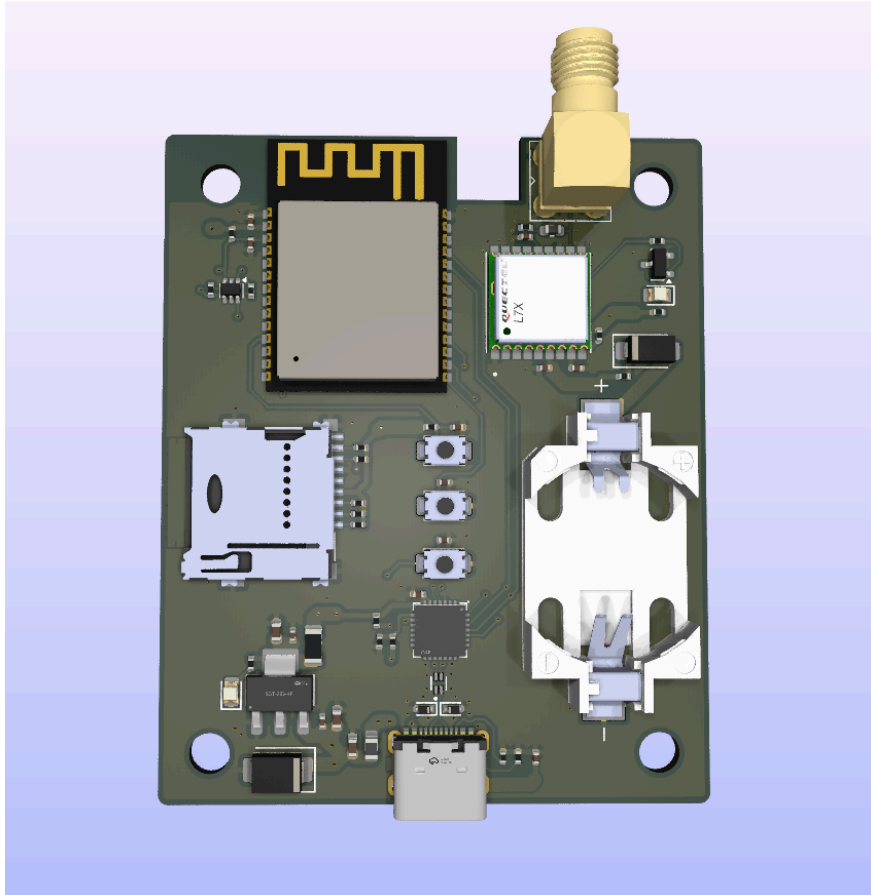
Schematic



Jake Odgers

Mileage Logger Project

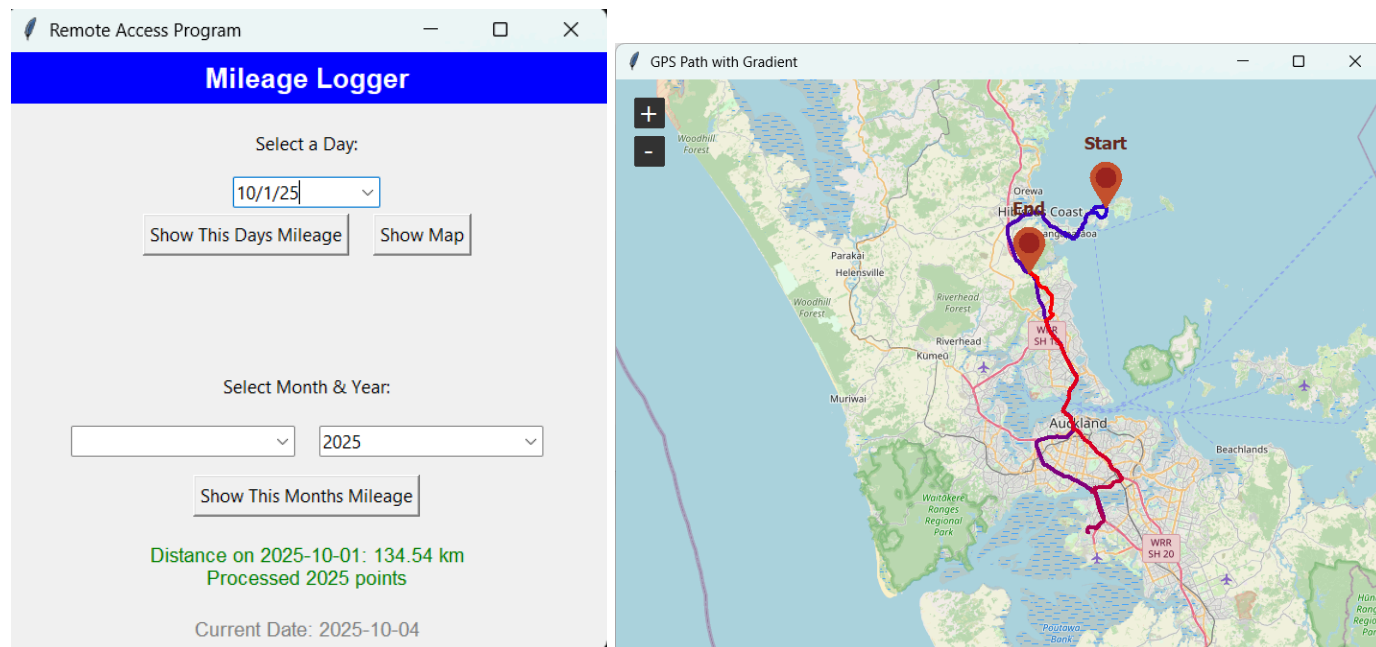
PCB Layout



Mileage Logger Project

GUI OVERVIEW

Option 1 - Computer Display



Mileage Logger Project

ESP32 REMOTE FIRMWARE UPGRADE THROUGH GITHUB

Mileage_logger / Scripts / firmware / version.json

Monkeyhead9459 Update version.json

Code Blame 4 lines (4 loc) · 140 Bytes

```
1 {
2   "latest": "1.0.0",
3   "firmware": "https://raw.githubusercontent.com/Monkeyhead9459/Mileage_logger/main/Scripts/firmware/firmware.bin"
4 }
```

GITHUB VERSION CONTROL

Mileage_logger Public

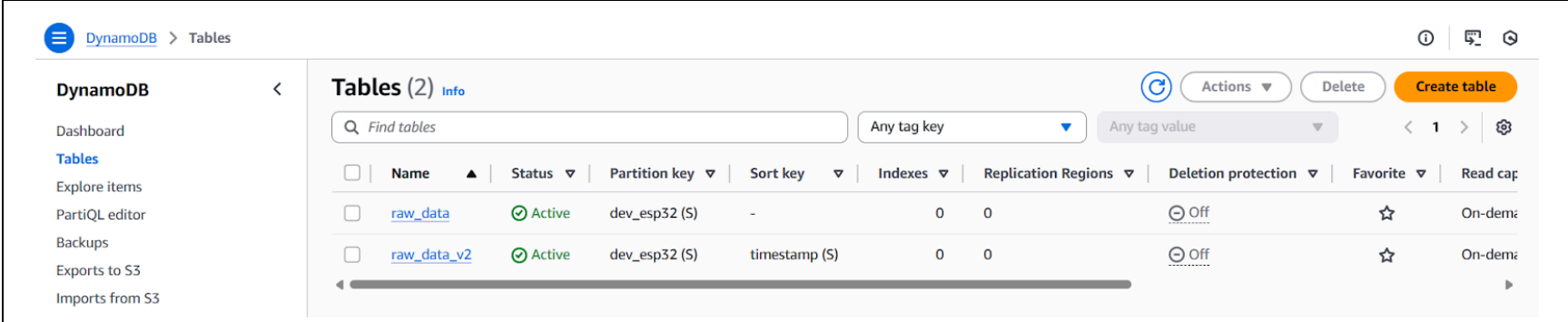
main 1 Branch 0 Tags Go to file

Monkeyhead9459 Update version.json

RAP - GUI	V1
Scripts	Update version.json
LICENSE	Initial commit

Mileage Logger Project

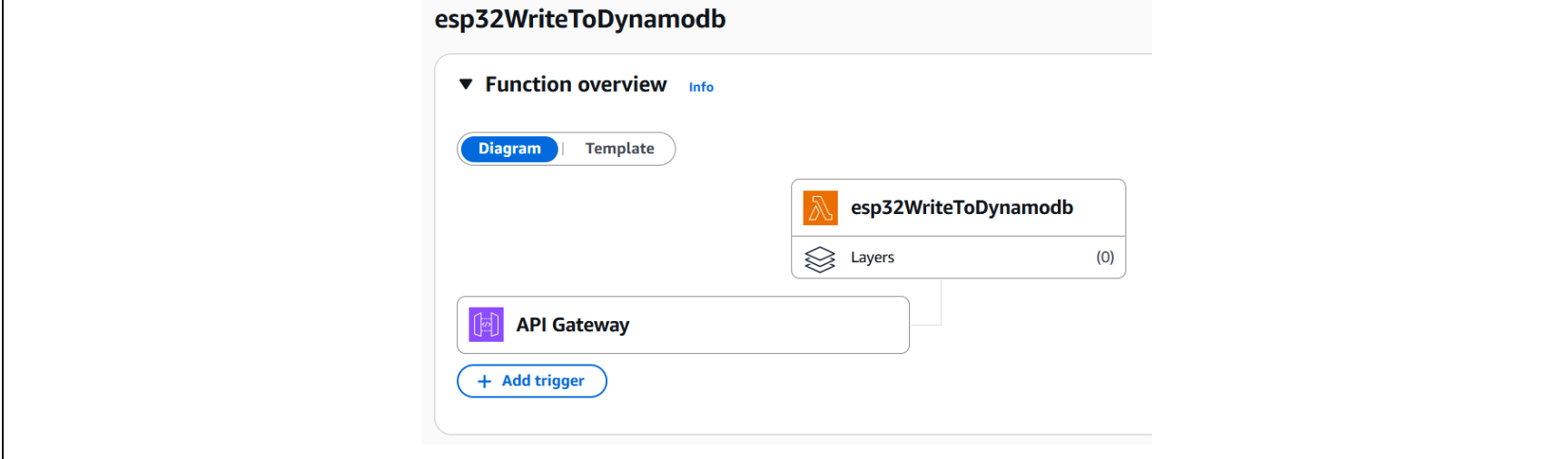
AWS DYNAMODB TABLE FOR CLOUD STORAGE



The screenshot shows the AWS DynamoDB console interface. On the left is a navigation menu with options like Dashboard, Tables, Explore items, PartiQL editor, Backups, Exports to S3, and Imports from S3. The main area is titled 'Tables (2)' and contains a table listing the existing DynamoDB tables.

	Name	Status	Partition key	Sort key	Indexes	Replication Regions	Deletion protection	Favorite	Read cap
<input type="checkbox"/>	raw_data	Active	dev_esp32 (S)	-	0	0	Off	☆	On-demand
<input type="checkbox"/>	raw_data_v2	Active	dev_esp32 (S)	timestamp (S)	0	0	Off	☆	On-demand

API GATEWAY TO LAMDA FUNCTION FOR ESP32 DATA STORAGE



The screenshot shows the AWS Lambda console for the function 'esp32WriteToDynamodb'. It displays the 'Function overview' section with tabs for 'Diagram' and 'Template'. The 'Diagram' tab is active, showing a visual representation of the function's configuration. It includes a box for the Lambda function 'esp32WriteToDynamodb' with '(0)' layers, and a box for the 'API Gateway' trigger. A '+ Add trigger' button is visible at the bottom.

Mileage Logger Project

Improvements:

- Incorporate Google Roads API for more accurate mileage logging by snapping to roads
- Make a portable option for battery connection.
 - SMPS (Increase efficiency)
 - Smaller profile
 - ESP32-C3-MINI-1U
 - Offboard wifi and bluetooth antenna
 - Local/Smaller GPS antenna
- Option 2 for display is making an app through Flutter to connect to the ESP32 through bluetooth and store and display results as well as change journey status from business to personal.