THE UNIVERSITY OF WESTERN AUSTRALIA

CITS 5506

The Internet of Things

Smart Livestock Tracking System

Smart Livestock Tracking System

- Project: Smart Livestock Tracking System
- Group Members:
  - Cedrus Dang 24190901
  - Ella Zhang 23844446
  - Faraz Khan 24427672
  - Yifan Lu 24094198
  - Yapei Chen 23764722
- Date: 17th Oct 2024
- University of Western Australia

1. Introduction

THE UNIVERSITY OF WESTERN AUSTRALIA

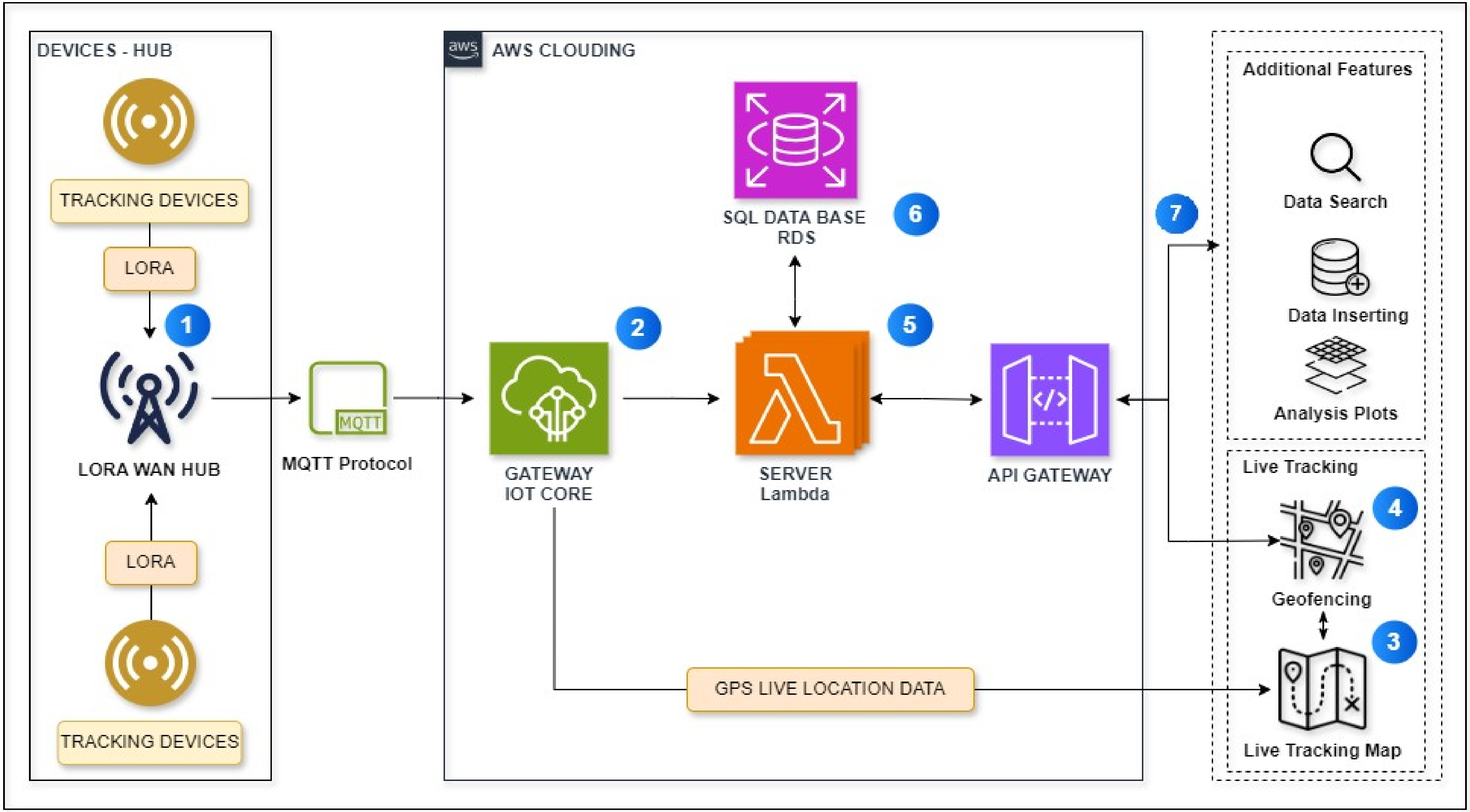
Why we do this project:

- Managing large herds across vast grazing areas has become more **complex in modern livestock farming.**
- **Traditional methods h**ave to face with various challedges due to the lack of tracking information, such as limited visibility, increased risk of theft, and operational **inefficiencies.**
- Effective tracking solution will impact greatly to not only the agriculture perfomance but also create positive improvement in the impact between livestock farming and the society.

Project objective:

- **Provide an effective tracking solution** by leveraging technological innovation, in this case: **GPS low power mode, LoRa and MQTT.**

2. Project Description: Diagram



2. Project Description

Hardware:

- Devices: LILYGO TTGO ESP32 Lora32 module is used to collect the GPS data and send data to base station using LoRa protocol.
- **HUB:** Another TTGO Lora32 will act as LoRa WAN hub to receives data from the trackers and transmits it to the AWS server using Wifi connection.

Software:

- **AWS is our system's main cloud service**. GPS data are sent to AWS IoT Core Gateway with MQTT protocol, stored in SQL database RDS, and transmitted to the front end by two way: Live data and stored data. API gateway also used in this connection.
- **A user website** was created to show the real-time positions of livestock, making it easier for farmers to manage their livestock.
- **A geofencing and Alert System** was created to send pop up alert and *can be* connected with SMS alert sender in commercial version if required.

Technics:

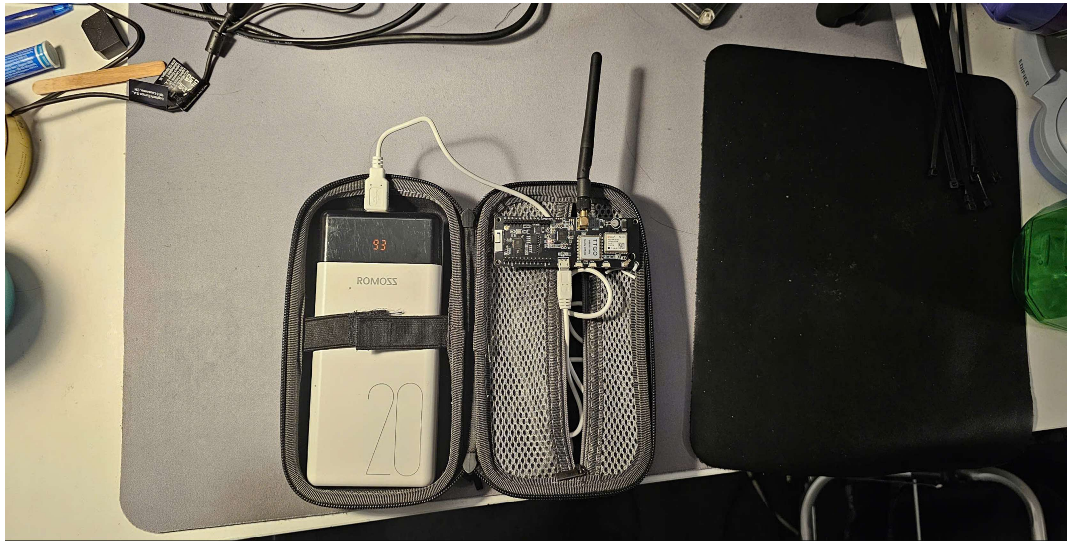
- GNSS (GPS), LORA, MQTT protocol, Clouding Computering, API protocol

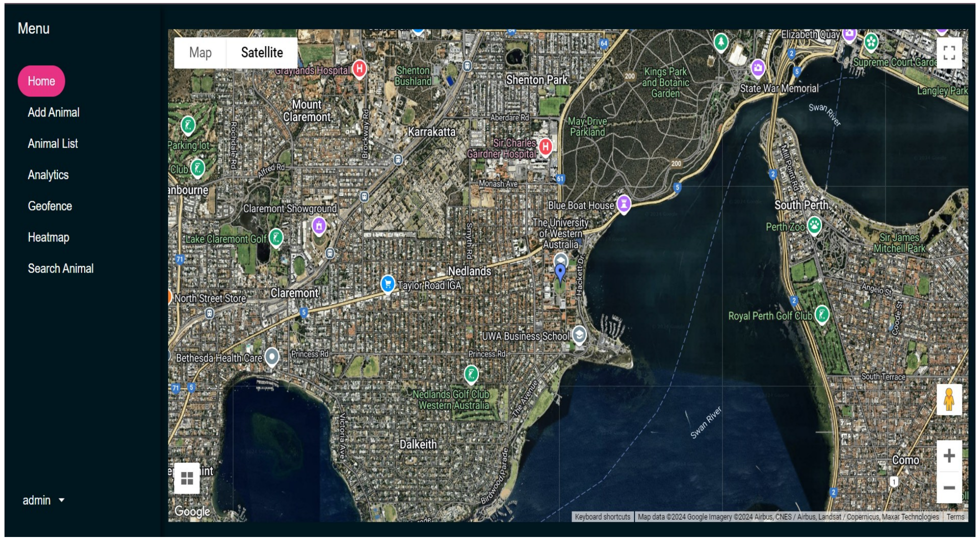
3. Design and Development Process

Key challenges and solutions:

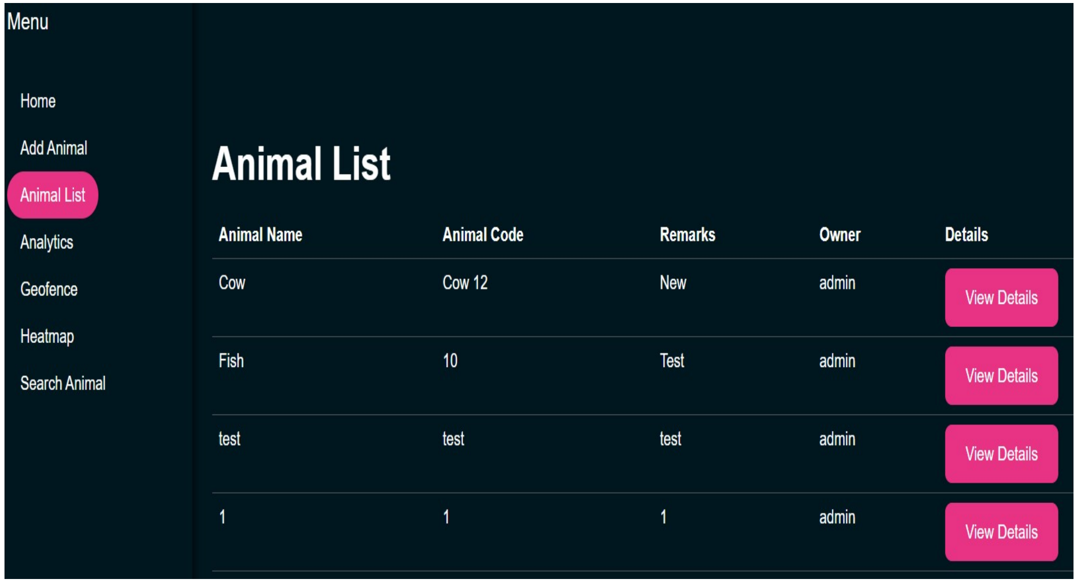
- **High GPS data transfer rate: MQTT** instead of API for live data.
- **High power consuming and difficult to charge** devices due to large scale animal hears: GPS Save Power mode, Sleep period of 30s (Sample), and deep sleep period from 6 am to 6pm (Sample).
- Need **cheap and long-range** communication technology: LoRa.
- The **balance between harware cost and computer power**: ESP32 instead of Andruino or Raspberry Pi.

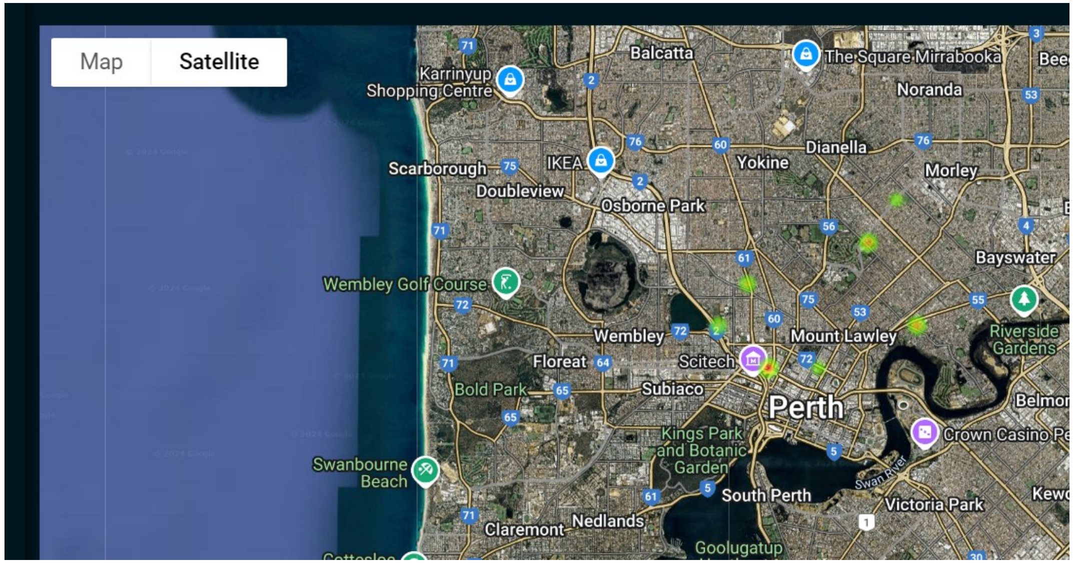
Prototype

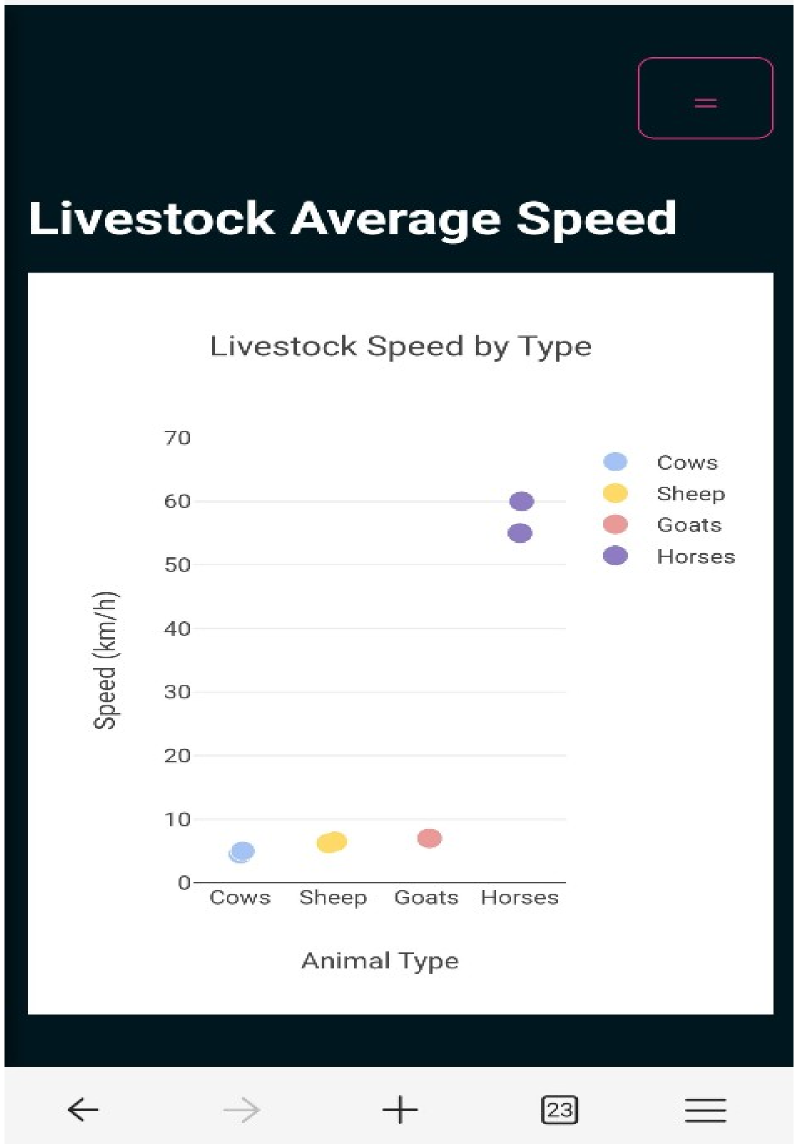



Menu
Map
Satellite
Home
Add Animal
Animal List
Analytics
Geofence
Heatmap
Search Animal
admin

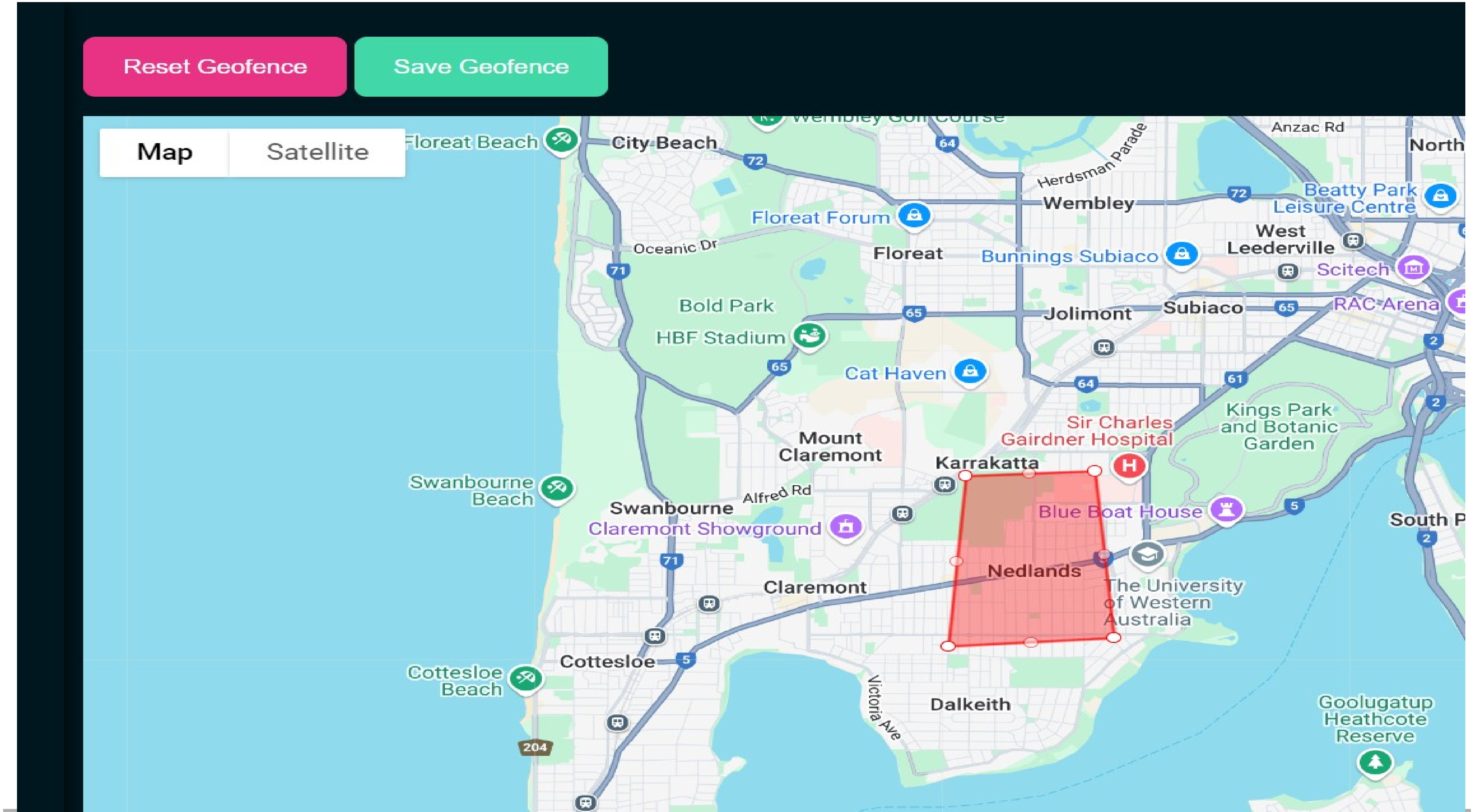
3. Design and Development Process







3. Design and Development Process



4. Results (Testing and Validation)

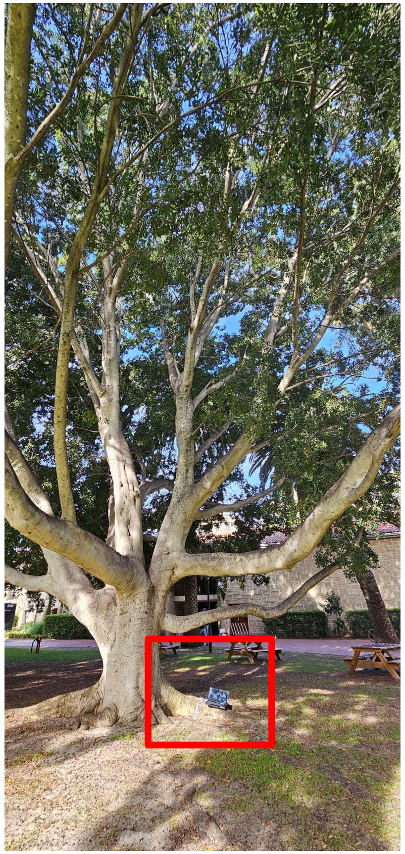
- Testing procedures :
  - GPS accuracy fixing proficiency.
  - LoRa connection.
  - MQTT connection.
  - Use flow in user interface.
  - Geofencing test.
  - Battery capacity test.

4. Results (Testing and Validation)

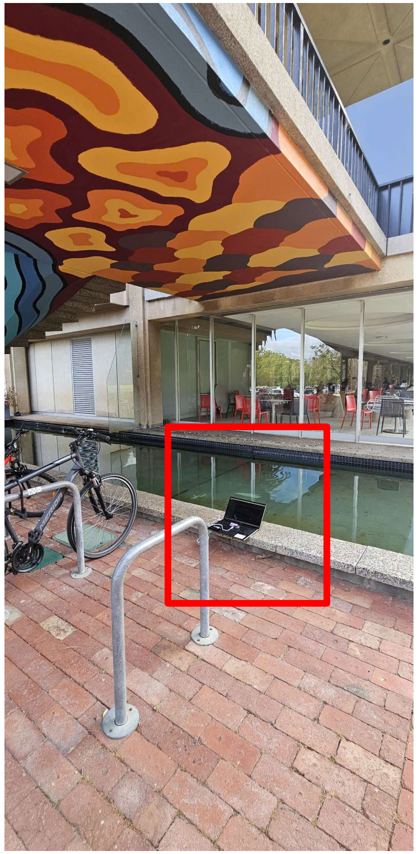
- **GPS: Testing setups**

Open Area

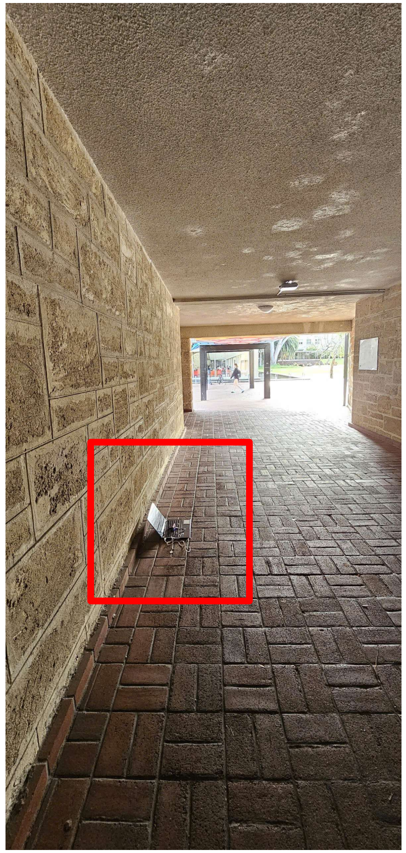
Under Canopy
Low Density



Under Canopy
High Density

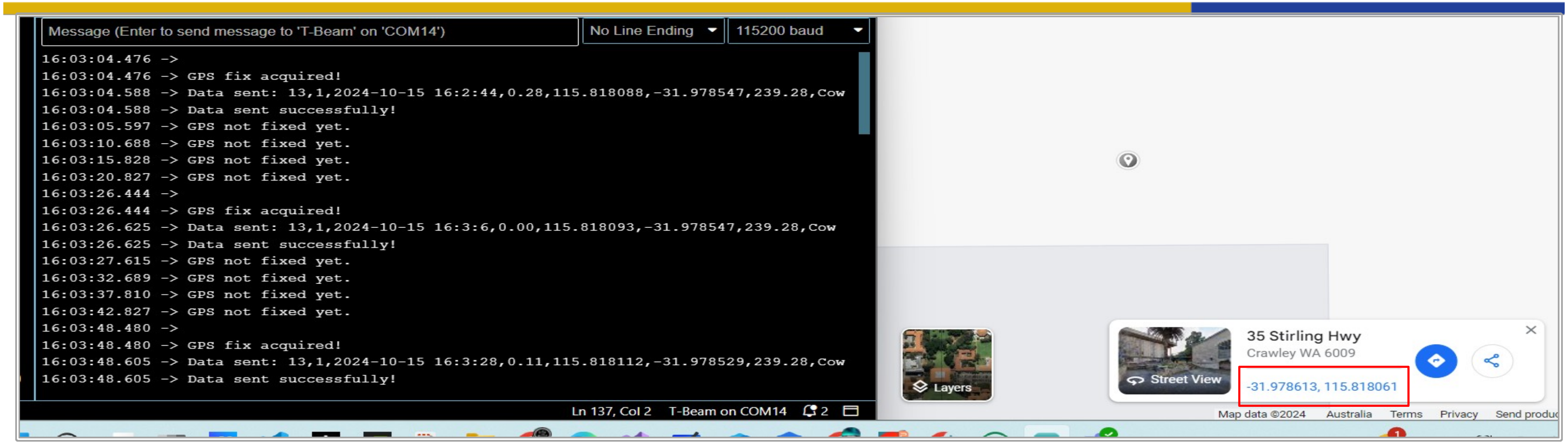


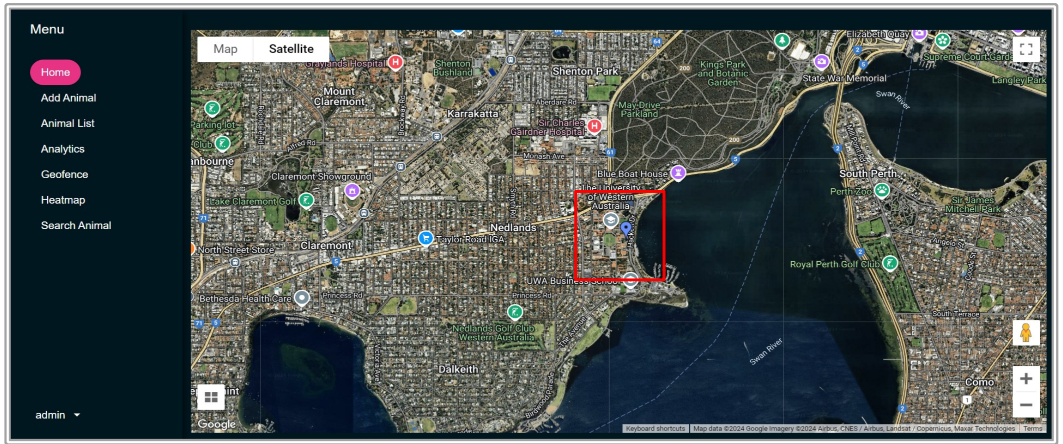
Under Roof

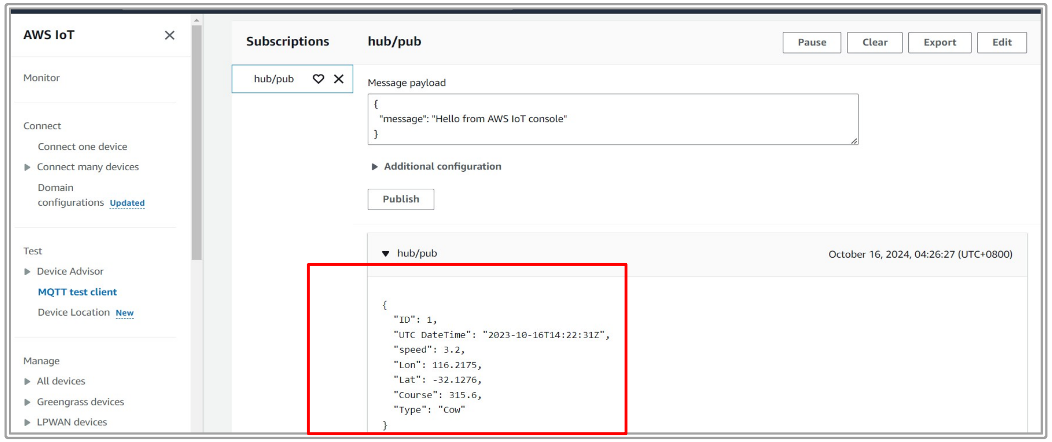


Inside Building

4. Results (Testing and Validation)







Test tools: Andruino IDE serial console, GMap GPS service, AWS test tool for IoT Core, Project live map.

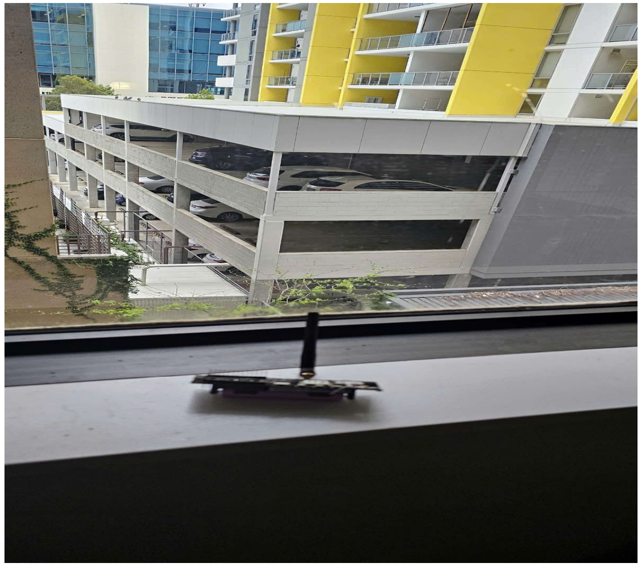
4. Results (Testing and Validation)

- **GPS: Performance metrics**

Scenario	Accurancy	Mode	GPS fixing time (s)	Note and Results
Open area	99%	Normal Mode	20	• Wait period per try is 5s. • There are some unpattern delays in fixing time during test, however, it is rare and in the expectation as the number of sateline sometime not was enough (<3) • Not different in accurancy, but there were variance in the fixing times between scenarios.
	99%	Save Power Mode	22 var: 1 ~ 4	
Under canopy (Low density)	99%	Normal Mode	20	
	99%	Save Power Mode	22 var: 1 ~ 4	
Under canopy (Highdensity)	99%	Normal Mode	20	
	99%	Save Power Mode	22 var: +2/+4	
Under Roof	99%	Normal Mode	22 var: 1 ~ 4	
	99%	Save Power Mode	20	
Inside building	0.0%	Normal Mode	+ inf	
	0.0%	Save Power Mode	+ inf	

4. Results (Testing and Validation)

- **LoRa: Major findings**
  - LoRa connect can only last **under 100m** when Hub is **indoors** (In the apartment, in a room with a window in the same direction as the device). Before disconnecting and reconnecting, the data **can break** and give invalid data.
  - In the **open field**, it can last **at least 1km** without disruption.



4. Results (Testing and Validation)

- **Others:**
  - **MQTT** using IOT Core AWS **took some time** to send data to the web interface.
  - **Geo-fencing tests** show that **accuracy depends** heavily on the sleep period of the GPS sensor and the latency of LoRa and AWS services.
  - User experience testing provides valuable insights into customer needs and potential requirements.
  - **Battery tests** show that each 1% of a 20.000mAh battery can last for 1 hour with a 5s sleep period (~200mA/h). This means if the sleep period is 30s, it can last for 25 days. If adding 6 pm to 6 am deep sleep period, it can last up to 50 days.

4. Cost

No.	Items	Description	Amount	Cost/unit (AUD)	Web Address
1	TTGO LoRa32	LoRa32 GPS device with Wi-Fi and Bluetooth - 915MHz	2	43.00	Temu's store
2	Romoss Battery	20,000 mAh Power storage	1	43.00	Amazone.au's store
3	Cables	Connection components	2	0.00	In TTGO LoRA 32 package
4	AWS Cost	Cloud services, varies based on usage	*vary*	*vary*	https://aws.amazon.com/pricing/
SUM				**129.00**	

AWS Cost is varied. However, this is a very considerate cost. A wrong choices can increase the bill to thousands of USD. Localisation options are recommended as a result.

5. Conclusion

Summary of the project outcomes

- GPS data is **highly accurate** and requires minimal data cleansing and transformation.
- The GPS connection time can be prolonged due to roofs and the lack of satellites in the area (Minimal = 3) and can not be used indoor.
- Save power mode and prolonging sleep time solution is a trade of latency and power saving . However, this also impact geofencing accurancy.

Conclusion

- LoRa with ESP32 915hz **consumes only minimal power** but can work at **very far distances** and can go through walls but with very limited capacity.
- **LoRa connection is fast**, and the data sent to the Hub is almost **immediate**.
- **Clouding solutions** for databases are significantly **expensive** and need constant monitoring;
- **Balance between local and cloud solutions**, such as databases, data transformation, analysis and AI models on Raspberry Pi or the company's server, would be a better option.

Conclusion

- **Improvements:**

1. Deeper analysis on battery life and accuracy trade-off.
2. Local servers and other solutions to save running cost.
3. Additional features such as AI models as competitive key features.
4. AI models to fill no information area in the sleep period using course value + speed to lat and long.
5. Combine GPS with other geofencing technologies like motion sensor fencing (Using literary review). The expected result is that GPS sleep time can be longer and suit solar power capacity while still getting instance alarms.

References

- **References**
  - **Guidance materials:** Andruino Project book, AWS learning platform.
  - **IDE:** Andruino IDE.
  - **Additioanl testing tool:** ESP32 Wifi - Simulating data sender and IoT tester.
  - **Literary review :** Cabezas et al. (2022), Schulthess et al. (2023), Casas et al. (2021), Ilyas et al. (2020), and Handcock et al. (2009) for GPS. Molapo et al. (2018) for potential improvement using tag, beacon, and base station nodes with trilateration methods.

Q&A Slide