User Requirements:

The battery efficiency is a crucial aspect for our project, and we can distinguish the duration according to the use of the device. Specifically, we can identify two macro scenarios:

- **At least 24 hours:** in case we lost our pet it is important that the device remain active for at least one day, that could be a sufficient amount of time to track and retrieve him;
- **At least one/two weeks:** in case we do not lose contact with our pet, it is sufficient that the device remains active for at least one/two weeks. This period of time can be used by the user if he will stay out for days, and it's also a good compromise between time required to retrieve the pet and efficiency consumption from a user point of view.

The **accuracy** of the system must be of 10-15m, that is a sufficient range in order to be able for the user to find his pet. (Happily,)The chosen gps sensor is also respecting these constraint:

Precision (Test condition 3)	Horizontal positioning accuracy	2.5	m
	High positioning accuracy	3.5	m
	Speed positioning accuracy	0.1	m/s
	Timing accuracy	30	ns

Position update (and so **duty cycle**) is another important aspect, and also for this requirement we could distinguish two macro scenarios:

- **Never/Once a day/week:** in case our pet is in our view, we could avoid to update the position because we know where it is, or decide to set it depending on the user needs;
- **Each 10 min:** when we lost the contact with the pet, 10 minutes could be a sufficient amount of time to update the position, because in 10 minutes the pet could not do so much road, and the user can rescue him;
- **When required for the user**: if the user wants to know the position of the pet in a certain moment, he can know it by requesting it.

Security is another crucial aspect: data exchanged by the application must be encrypted in order to hide sensitive information (like the position of the pet) to malicious attackers.

Why LoRa?

Since data transfer size are in the order of 2^6 bytes (+ headers) (assuming that latitude and longitude are float data types, and that the data transferred has a form like:

{"Latitude": float, "Longitude": float}

therefore, the total size of the given string would be ca. 48 bytes), and since we need that the system has to be energy efficient, and since we need a communication in the order of km, LoRa is strongly suitable for our project, because allows **long range communication**

(over 10 km in rural areas, 3–5 km in heavily urbanized areas), **low data rate** and **low power consumption**. Moreover, it is also suitable for **security**: indeed, LoRaWAN has several methods that help to prevent the possibility of radio eavesdropping.
LoRa is beautiful but, since there are still issues in making it work on ESP32, we will use Wi-Fi for now. We will make simulations on IOT-LAB, we will make a prototype using WIFI (instead of LORA) to show real measures, and we will integrate simulation and prototype making realistic assumptions.

On **power consumption**, since we do not yet have the necessary knowledge regarding the power consumption behavior of ESP32 Heltec LoRa32 (V2), neither in sleep mode nor in operating mode, we will carry out simulations on IOT-LAB and tests in real-life scenario, in order to make the system work for a reasonable amount of time to satisfy the user requirements.