

Traçage de véhicule par GNSS

Objectif :

Objectif principal :

Il s'agit d'utiliser l'environnement Pycom afin de transmettre les données GNSS d'un véhicule à travers le réseau LoRaWan de The Things Network [TTN] et de visualiser sa position sur une carte en se connectant sur un serveur.

Objectif final :

Dans l'idée d'un système embarqué où l'autonomie par batterie est un critère important, ce système doit pouvoir consommer peu tout en permettant le suivi d'un véhicule lorsque c'est nécessaire.

Ainsi, l'électronique embarquée doit être mise dans un mode de faible consommation comme le proposent les modes « sleep » des cartes et modules. Ce n'est que lorsque le véhicule se déplace, que le GNSS et la connexion au réseau LoRaWan se fait, et ceci, grâce à l'accéléromètre par son interruption dédiée.

Le système sera (re)mis en mode faible consommation par un message de l'utilisateur depuis une page web.

L'utilisateur sera aussi prévenu par SMS que la voiture se déplace !

Contexte :

- Vous serez en groupe de 3 ou 4 étudiants.
- Temps global : 27.5 heures, réparties de la manière suivante :

Temps alloué encadré : CM : 1.5 heure TD : 2*1.5h = 3 heures TP : 1*3.5h = 3.5 heures TOTAL : 8 heures	Temps alloué en autonomie : TP : 4*4h+3,5H = 19.5 heures
--	---

L'environnement matériel :

- Une carte Pytrack et un module Fipy ou Lopy qui sera dans le véhicule et qui servira de Node ou de End-Device
- Une carte Expansion 3 ou Pysens et un module Fipy ou Lopy qui serviront de Gateway LoraWan et qui devront être partagés entre 2 groupes
- Une antenne Lora par module
- Un Raspberry Pi [RPi] pour le serveur.
- Un Raspberry Pi comme AP WiFi pour la Gateway TTIG (le tout est déjà configuré 😊)

L'environnement logiciel :

- ATOM ou VS Code avec le package/l'extension Pymakr
- Node Red à installer sur le RPi afin de bénéficier d'un serveur web avec cartographie
- The Things Network [TTN] :

<https://www.thethingsnetwork.org/>

Compte TTN pour avoir accès au « Live data » de la GATEWAY TTIG [The Things Indoor Gateway]:

- Name : sae501
- Password : sae501lorawan

Ne rien rajouter sur ce compte, ni modifier, ni supprimer quoique ce soit !

Les validations :

Validation 1 : Envoyer les coordonnées en MQTT et les visualiser sur une carte depuis une page web (hors TTN)

Etape 1 : Envoyer en MQTT, depuis le End Device, des coordonnées de géolocalisation via une connexion WiFi

Etape 2 : Visualiser le positionnement d'un véhicule sur une carte depuis un lien http

Validation 2 : Créer sur TTN une Application et une Gateway

Etape 3 : Créer une Application liée à votre End-Device et voir les messages transiter depuis une Gateway (la TTIG mise à disposition). **Penser à respecter 99 IT ! Pourquoi ???:**

<https://avbentem.github.io/airtime-calculator/ttn/eu868> ou
<https://www.thethingsnetwork.org/airtime-calculator> ou
<https://www.thethingsnetwork.org/docs/lorawan/duty-cycle/>

Etape 4 **[A faire à la fin]** : Créer une Gateway et voir les messages des End-Device/Application

Validation 3 : Formatage des données côté TTN

Etape 5 : Récupérer le Payload et le formater en JSON : « Uplink » dans rubrique « Payload formatters »

Validation 4 : Visualisation des positions du véhicule sur la carte

Etape 6 : Utiliser le broker MQTT fourni par TTN (« Integrations »)

Etape 7 : Prise en compte du message MQTT par Node Red

Validation 5 : Visualisation des positions du véhicule – Version optimisée

Etape 8 : Optimiser les données de géolocalisation à transmettre sur la couche physique LoRa. A ce titre, la latitude et la longitude seront codées chacune sur 3 octets.

Validation 6 : Connexion au GNSS et LoraWan lorsque la voiture démarre (par défaut le Node est en faible consommation)

Etape 9 : Utiliser l'accéléromètre et le mode « deep sleep ».

Validation 7 : Arrêt du suivi par demande de l'utilisateur depuis la page web et passage en mode faible consommation.

Etape 10 : Utiliser le retour au niveau du LoRaWan : « Downlink » dans rubrique « Payload formatters »

Etape 11 : Utiliser un topic descendant par appui d'un bouton depuis la page web

Validation 8 : Utilisation des SMS et notifications pour être prévenu

Etape 12 : A l'aide d'une clé 2G/3G qui vous sera fournie, envoyez-vous un SMS pour être prévenu que le véhicule se déplace.

Etape 13 : Donner la possibilité de mettre le End-Device en mode veille par SMS avec un SMS confirmant cette mise en veille.

Etape 14 : Utiliser l'envoi de notifications sur votre smartphone par des services gratuits appropriés pour avoir les coordonnées au fur et à mesure.

Barème (sujet à modifications) :

NOMS :	NOMS :

Les validations sont conditionnées à votre capacité à expliquer ce que vous avez fait, notamment le contenu (vision globale et fonctions utilisées) de vos scripts.

Etape	Points	Environnement logiciel et matériel	Notation
Etape 1	2	End Device + MQTT Box	
Etape 2	4	End Device + carte (Node red)	
Etape 3	3	End Device + Gateway TTIG + TTN	
Etape 4	2	Gateway (avec carte en partage) + End Device + TTN	
Etape 5	3	End Device + TTN	
Etape 6	2	End Device + MQTT TTN + MQTT Box ou Node Red (debug)	
Etape 7	2	End Device + MQTT TTN + carte (Node Red)	
Etape 8	3	End Device + MQTT TTN + carte (Node Red)	
Etape 9	3	End Device + MQTT TTN + carte (Node Red)	
Etape 10	2	End Device + MQTT TTN	
Etape 11	4	End Device + MQTT TTN + page web (Node Red)	
Etape 12	4	End Device + MQTT TTN + Clé 2G/3G sur RPi	
Etape 13	3	End Device + MQTT TTN + Clé 2G/3G sur RPi	
Etape 14	3	End Device + MQTT TTN + Clé 2G/3G sur RPi	
TOTAL	30		