



# Mini-projet NSIUM

**Antoine MAROT** ([antoine.marot@ensemblescolaire-niort.com](mailto:antoine.marot@ensemblescolaire-niort.com))

**David SALLÉ** ([david.salle@ensemblescolaire-niort.com](mailto:david.salle@ensemblescolaire-niort.com))

**Julien SIMONNEAU** ([julien.simonneau@ensemblescolaire-niort.com](mailto:julien.simonneau@ensemblescolaire-niort.com))

Ce document est mis à disposition selon les termes de la licence

Creative Commons BY-NC-SA 4.0



Version du document v0.3

Date 09/05/2023

# Table des matières

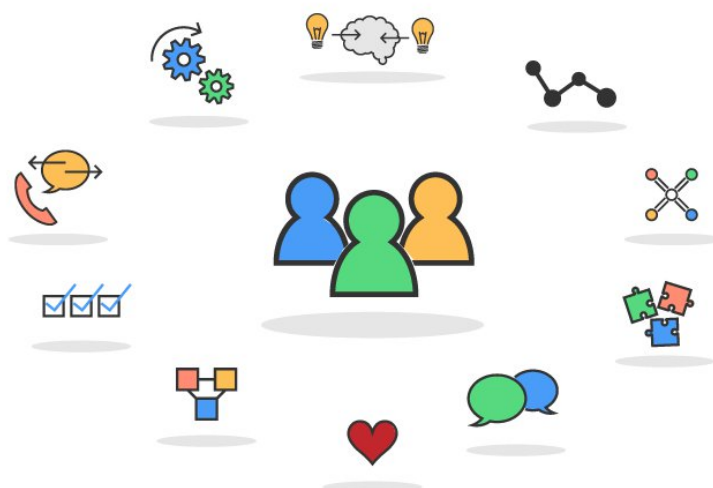
1 - Introduction .....	2
2 - Cahier des charges.....	3
2.1 - Contexte.....	4
2.2 - Synoptique .....	4
2.3 - Expression du besoin.....	5
2.4 - Contraintes .....	5
2.5 - IHM .....	6
2.5.1 - IHM pilotage/visualisation.....	7
2.5.2 - IHM exploitation.....	7
2.6 - Ressources .....	9
2.6.1 - Matérielles .....	9
2.6.2 - Logicielles .....	10

# 1 - Introduction

Ce document présente un mini-projet à réaliser en section terminale NSI. L'objectif est de réinvestir différentes notions abordées depuis la classe de première :

- langage Python
- programmation objet, évènementielle, multitâches
- base de données et requêtes SQL
- page web dynamique avec Flask
- client/serveur en réseau
- sécurisation des échanges

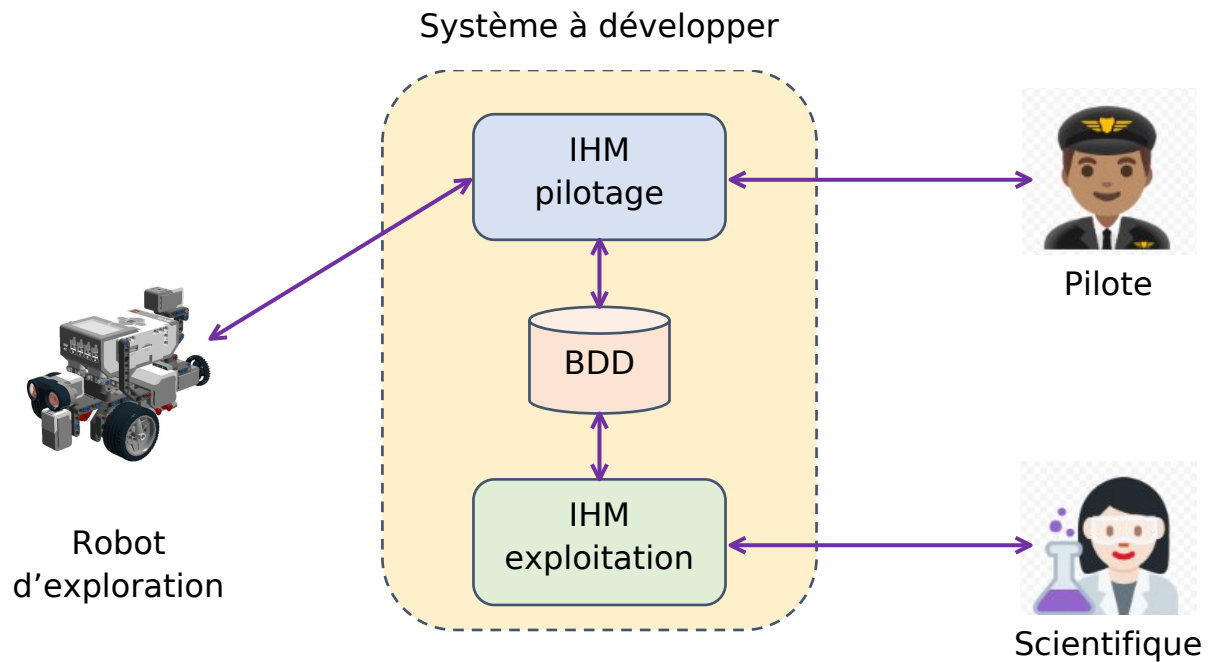
Ce mini-projet sera réalisé en équipe de 2 à 3 élèves. Chaque membre travaillera sur une problématique particulière et collaborera avec les autres membres de l'équipe.





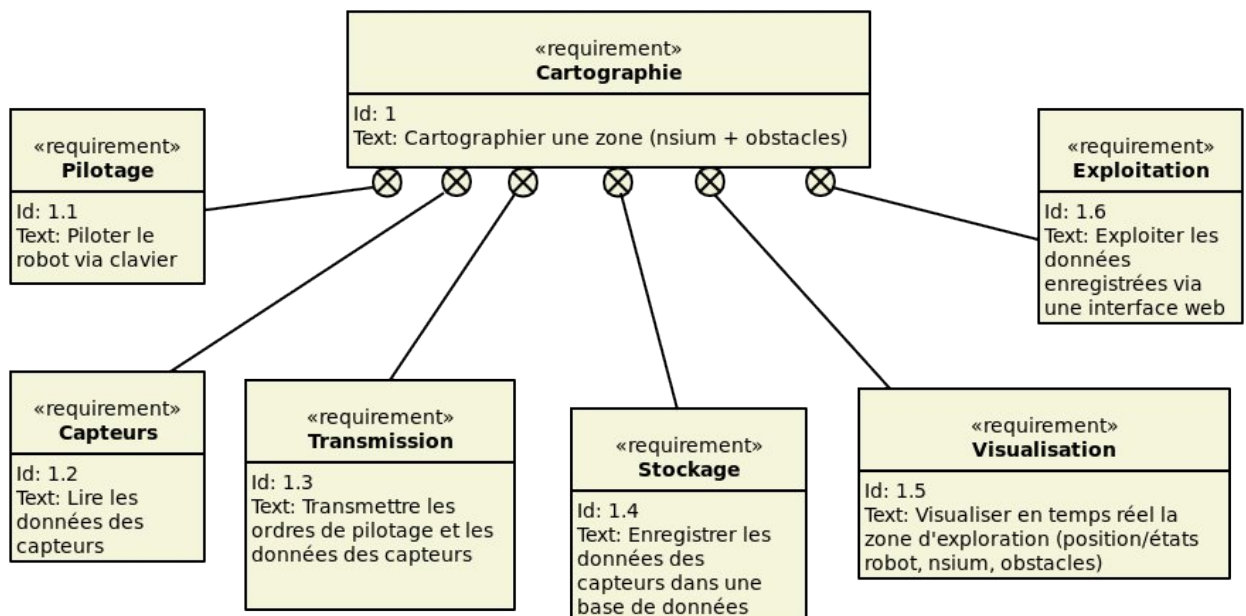
## 2.2 - Synoptique

Ci-dessous le synoptique du système :



## 2.3 - Expression du besoin

Ci-dessous le diagramme SysML des exigences. Chaque bloc exprime une fonctionnalité à réaliser.



## 2.4 - Contraintes

Afin de ne pas surcharger le diagramme des exigences, les contraintes et précisions liées à chaque exigence sont rappelées dans le tableau ci-dessous.

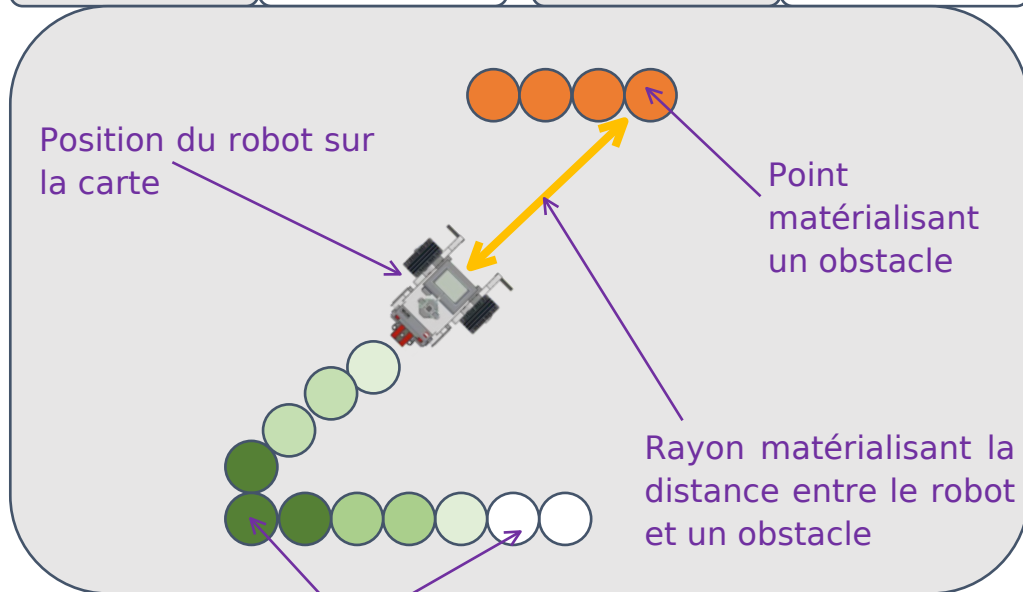
<b>Exigence</b>	<b>Contraintes/précisions</b>
1.1 Pilotage	Le pilotage du robot se fera via le clavier d'un ordinateur. On retrouvera 6 ordres de pilotage associés à 6 touches <ul style="list-style-type: none"><li>• avancer</li><li>• reculer</li><li>• tourner à gauche</li><li>• tourner à droite</li><li>• lever barre</li><li>• baisser barre</li></ul>
1.2 Capteurs	Le robot renverra au système différentes informations <ul style="list-style-type: none"><li>• le taux de nsium mesuré dans le sol (en pourcentage)</li><li>• la distance robot/obstacle (en mm)</li><li>• la position angulaire du robot (en degrés)</li><li>• la position angulaire des 2 roues (en degrés)</li></ul>
1.3 Transmission	La transmission des ordres de pilotage et des informations se fera via une liaison réseau wifi. Idéalement cette liaison sera cryptée pour éviter toute fuite d'informations Les données devront arriver toutes les secondes environ
1.4 Stockage	Le stockage des informations en provenance du robot se fera dans une base de données relationnelle afin de faciliter son exploitation ultérieure
1.5 Visualisation	Le pilote du robot pourra visualiser en temps réel (délai < 1s) sur une IHM la zone d'exploration avec : <ul style="list-style-type: none"><li>• la position du robot (à calculer à partir des données des capteurs)</li><li>• les obstacles</li><li>• les points d'échantillons des taux de nsium</li></ul> Voir le détail de l'IHM attendue au 2.5.1
1.6 Exploitation	Le scientifique pourra à distance depuis une interface web dédiée effectuer différentes actions : <ul style="list-style-type: none"><li>• rechercher une campagne d'exploration particulière en fonction d'un nom ou d'une date</li><li>• visualiser la carte d'exploration avec les obstacles et les zones nsium</li><li>• afficher les valeurs brutes ainsi que les minimum, maximum, moyenne des taux de nsium</li><li>• exporter les données brutes au format CSV</li></ul> Voir le détail de l'IHM attendue au 2.5.2

## 2.5 - IHM

### 2.5.1 - IHM pilotage/visualisation

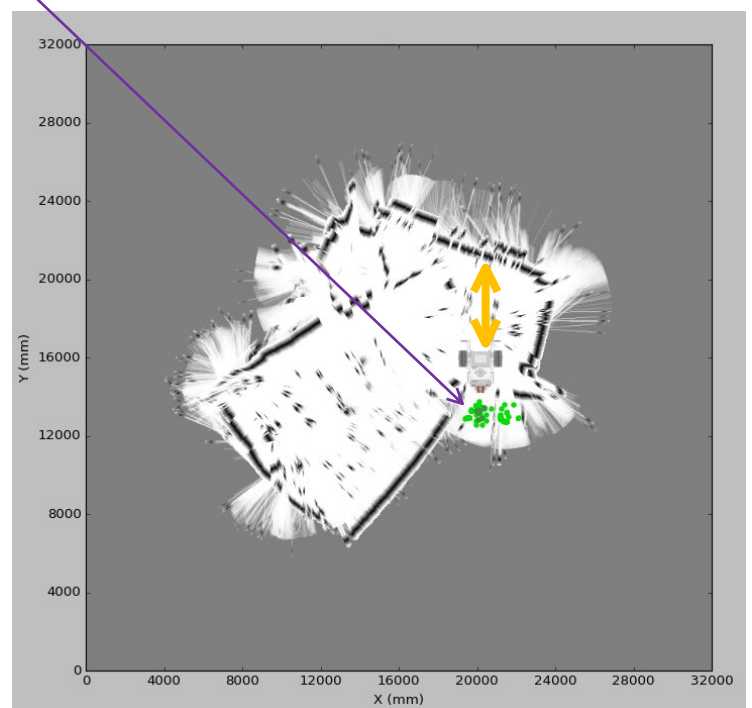
L'IHM de pilotage/visualisation contiendra a minima les éléments suivants.  
Libre à vous de la modifier/améliorer.

Nsium :	10 %	Position :	(32, 20)
Distance :	300 mm	Angle :	45 degrés



Point de mesure Nsium :  
plus le taux est important,  
plus la couleur est foncée

Résultat possible en dessinant  
chaque rayon se terminant par  
un point sombre pour  
matérialiser les obstacles



## 2.5.2 - IHM exploitation

### Page de recherche d'une campagne

#### Recherche d'une campagne

Nom

Date

Pilote

#### Liste des campagnes

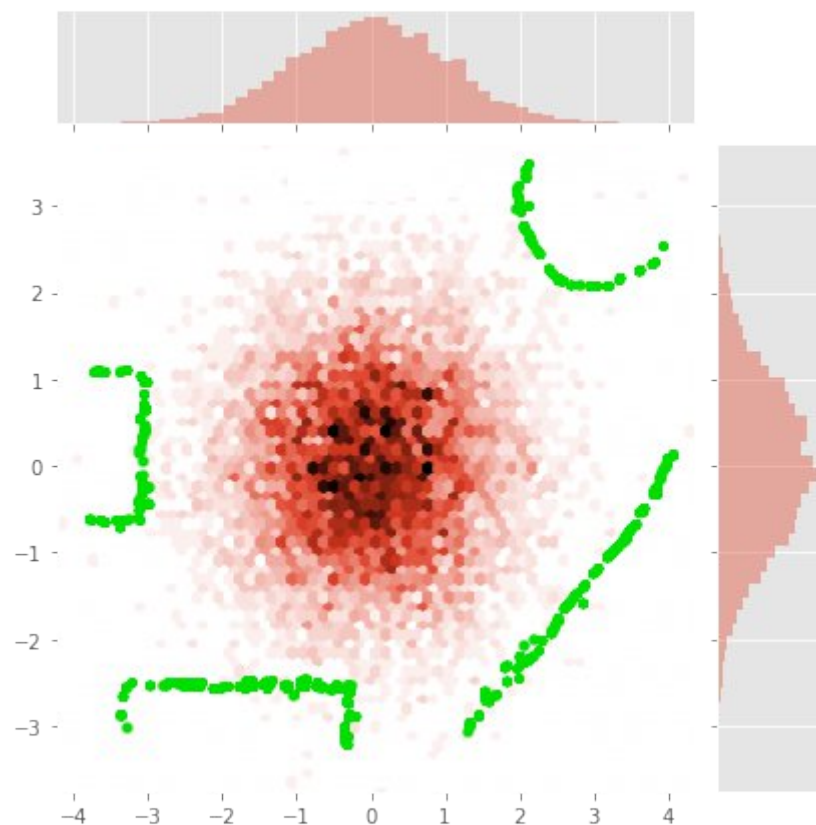
Nom	Date	Pilote	Action
Grotte des marcassins	15/03/2021	BEGOOD Johnny	<input type="button" value="Visualiser"/>
Grotte des papillons	21/03/2021	BEGOOD Johnny	<input type="button" value="Visualiser"/>

### Page d'exploitation graphique d'une campagne

#### Visualisation des données

Campagne

Date



Les points verts matérialisent ici les obstacles



### Exploitation des données

Campagne

Grotte des marcassins

Date

15/03/2021

#### Taux de nsium

Date	Taux nsium	Latitude	Longitude
09h32m45	10	32	21
09h32m46	12	33	20
09h32m47	12	33	20

#### Obstacles

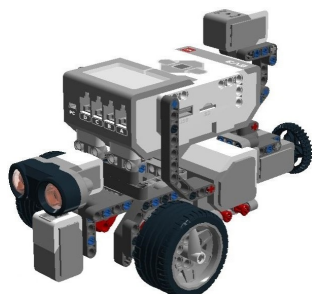
Date	Latitude	Longitude
09h32m45	47	55
09h32m46	48	56
09h32m47	49	57

Exporter en CSV

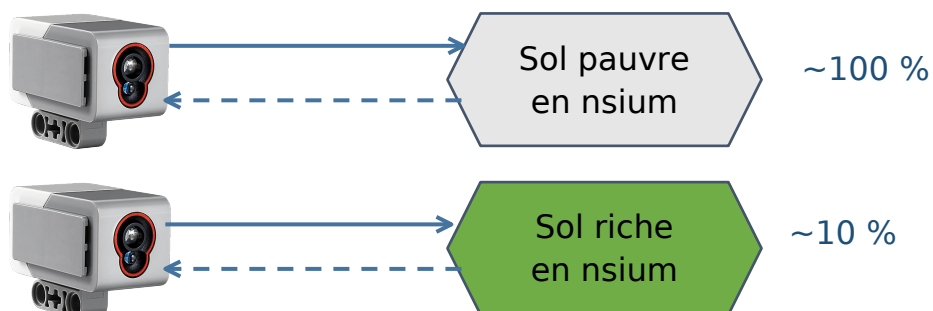
## 2.6 - Ressources

### 2.6.1 - Matérielles

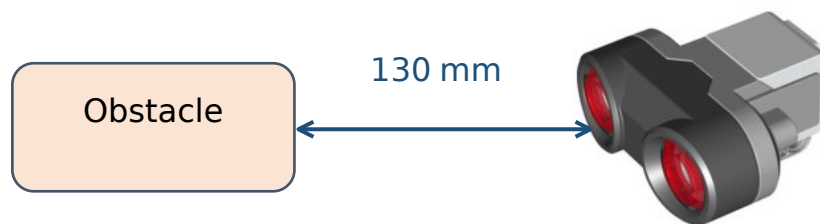
Le **robot** utilisé sera un EV3 de chez LEGO équipé d'un système d'exploitation Linux Debian 9 (stretch) + micropython et d'un dongle wifi afin de pouvoir le piloter à distance et récupérer les mesures effectuées.



Le robot portera un **capteur** de nsium . Ce capteur envoie un signal spécial dans le sol. Si celui-ci revient à 100 % c'est qu'il n'y a pas de nsium en dessous. Si le signal ne revient que faiblement par exemple à 10 %, c'est que du nsium se trouve en dessous et qu'il a absorbé une grande partie de l'énergie du signal d'origine. Son taux serait alors de 90%.



Le robot utilisera également un **capteur ultrason**. Ce dernier permettra de mesurer la distance entre d'éventuels obstacles et le robot. Le pilote à l'instar d'une chauve souris pourra ainsi éviter les obstacles présents sur la zone de recherche.



Pour le positionnement, plusieurs capteurs pourront être utilisés dont le capteur **gyroscopique** et le capteur **tachymétrique** intégré aux moteurs.



Plus d'informations ici : <https://pybricks.github.io/ev3-micropython/>

Informations WiFi :

- SSID : LEGOBOX
- clef : p4ssw0rd!

## 2.6.2 - Logicielles

Quelques pistes concernant les ressources logicielles :

<b>Ressource</b>	<b>Description</b>
Langages	Python, HTML/CSS, SQL
IDE	Visual Studio Code + lego-education.ev3-micropython
Base de données	SQLite
Serveur web	Flask, SVG
IHM pilotage	pygame, tkinter
Réseau	ssh, netcat
Collaboration	Discord, Office365, Google docs, github