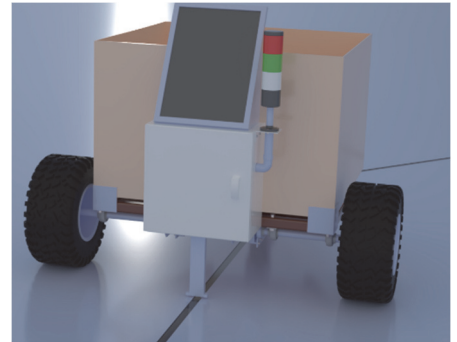


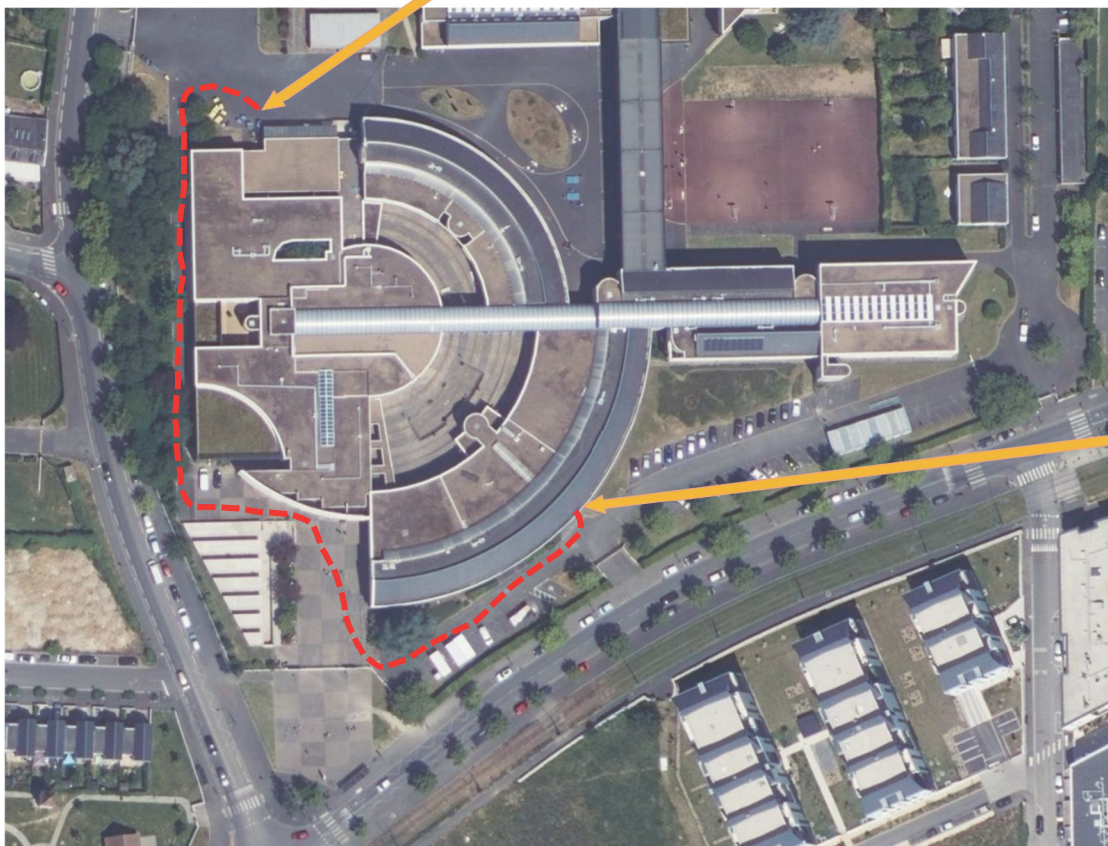
**Produit : Chariot AGV** (Automated Guided Vehicle)**1. Problématique et performances attendues**

Le responsable du magasin de l'enseignement technologique d'un lycée réceptionne toute l'année les colis issus des commandes des professeurs. L'emballage des matériels réceptionnés génère beaucoup de cartons à évacuer qui doivent être recyclés.

L'évacuation n'est possible qu'après le broyage des cartons. Le broyeur étant éloigné du magasin, le magasinier perd beaucoup de temps à transporter les cartons vident au détriment de l'ouverture du magasin.

Afin de soulager le magasinier de cette tâche chronophage, un chariot électrique pouvant stocker et transporter les cartons à recycler se chargera de les déplacer de nuit du magasin au broyeur (aller) et du broyeur au magasin (retour) la nuit suivante lorsque les cartons auront été déchargés et broyés :

*Zone de stationnement  
du chariot AGV à  
proximité du broyeur.*



*Zone de stationnement  
du chariot AGV devant  
le magasin.*

*Chemins (230 m) et  
emplacement du  
marquage au sol  
(ligne noire).*

Figure 1 – Vue aérienne du lycée

Le projet impose un cahier des charges, sous forme de diagramme des exigences, précisant les axes techniques à respecter afin d'assurer les déplacements du chariot AGV.

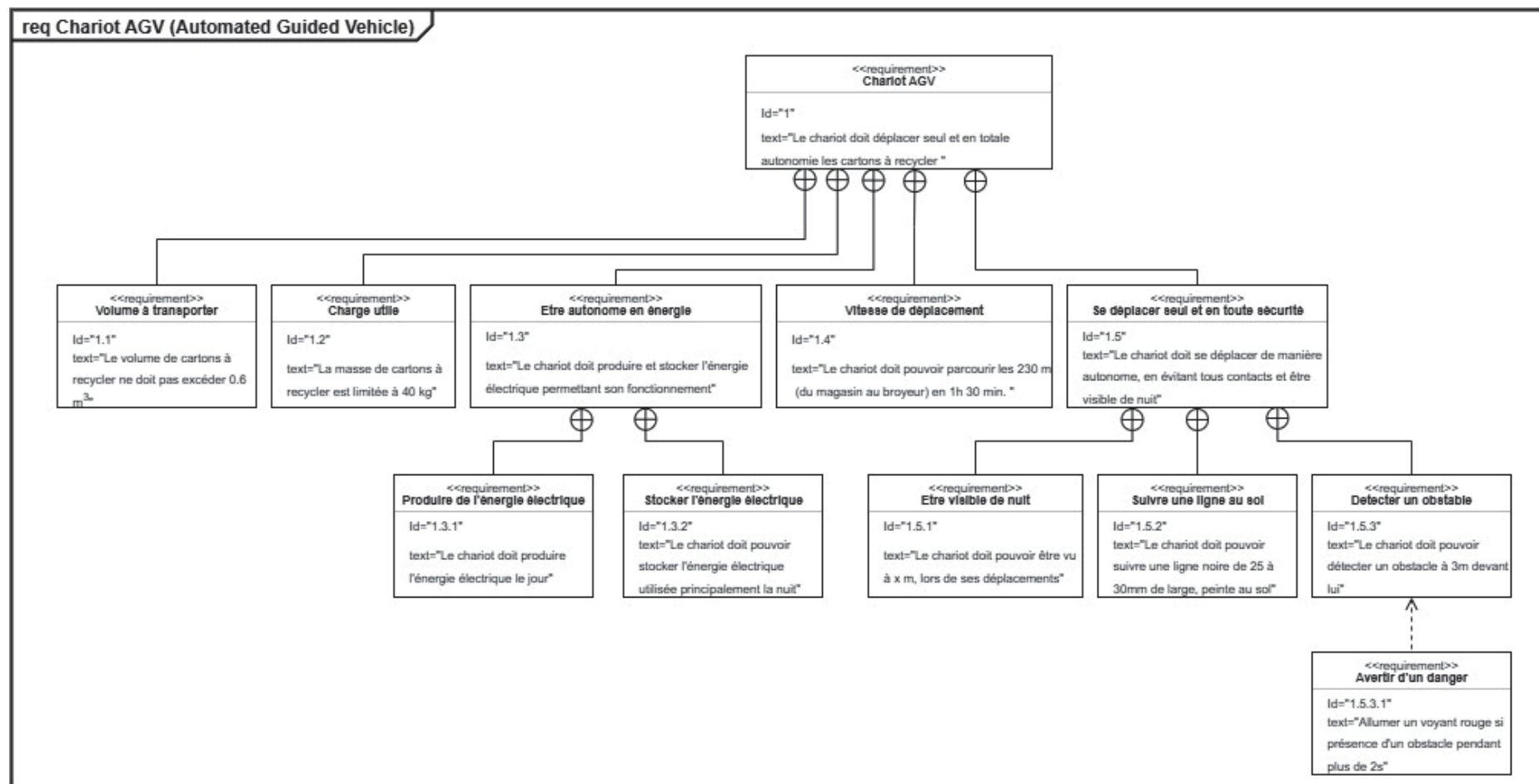


Figure 2 - Diagramme des exigences

## 2. Expérimentation

Matériel à disposition pour cette partie :

- Ordinateur équipé de l'IDE arduino,
- 1 carte arduino
- 1 capteur MAKER-LINE, une feuille avec ligne noire et quelques câbles

Lire la documentation du capteur de suivi de ligne MAKER-LINE et visualiser la vidéo du calibrage du capteur.

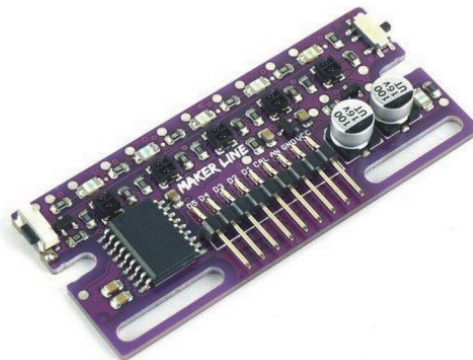


Figure 3 Capteur MAKER-LINE

Ce capteur possède :

- 5 sorties logiques donnant indépendamment l'état des 5 capteurs (Les capteurs, espacés de 10mm, sont composés de 5 LEDs infrarouges et de 5 phototransistors mesurant la réflectance d'une surface.)
- 1 sortie analogique délivrant une tension représentant précisément la position de la ligne.

Afin de limiter le nombre d'entrées utilisées sur la carte arduino, on souhaite utiliser ce capteur avec sa sortie analogique.

- 1 **Réaliser le câblage** du capteur sur la carte arduino
- 2 **Réaliser le calibrage** du capteur
- 3 **Noter** les valeurs analogiques dans les différents cas de figures du tableau ci-dessous représentant les différentes positions de la ligne :

		S1	S2	S3	S4	S5			Valeur moyenne	
										gauche
										gauche
										milieu
										milieu
										milieu
										droite
										droite

On considère que la ligne est à droite lorsque seulement 2 capteurs au maximum à droite la détectent.

On considère que la ligne est à gauche lorsque seulement 2 capteurs au maximum à gauche la détectent.

- 4 **Modifier** votre programme pour afficher "DROITE", "MILIEU", "GAUCHE" dans la liaison série en fonction de la lecture analogique de la position de la ligne.

### 3. Simulation

Matériel à disposition pour cette partie :

- Ordinateur connecté à internet
- Fichier sur site TINKERCAD

On souhaite réaliser un programme non bloquant permettant la détection d'un obstacle à moins de 3m pendant plus de 2 secondes répondant à l'exigence 1.5.3 du cahier des charges.

- **Charger** le [fichier TINKERCAD](#)

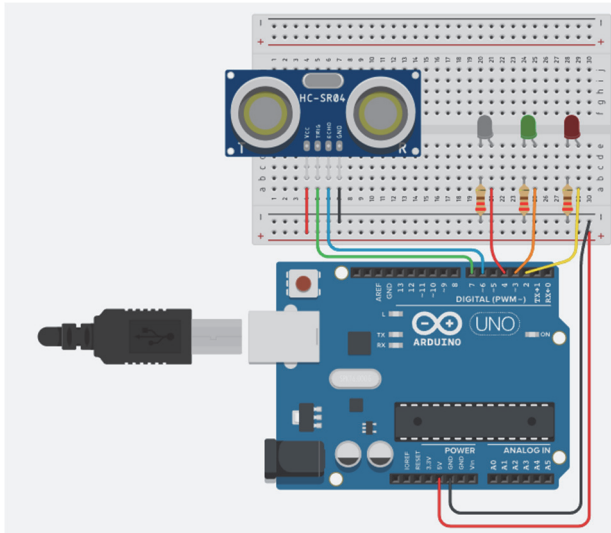


Figure 4 Fichier TINKERCAD



Figure 5 Colonne à LED

- **Ouvrir** le programme (la fonction `readUltrasonicDistance()` est fournie, elle retourne une distance en cm de type float)
- **Justifier** pourquoi l'exigence 1.5.3.1 du cahier des charges exige une détection d'obstacle pendant 2s ?

On souhaite le fonctionnement suivant :

- Lorsqu'**aucun** objet n'est détecté, le voyant vert est allumé.
- Lorsque la présence continue d'un obstacle **sous les 3 mètres** est détectée
- Le programme **enregistre le moment** où l'obstacle est apparu.
- S'il reste présent **plus de 2 secondes**, le voyant rouge est allumé, le chariot s'arrête.

On utilisera pour cela l'instruction `millis()` au lieu de `delay()` :

L'instruction `millis()` en Arduino **retourne le nombre de millisecondes écoulées** depuis le **démarrage du programme (`setup()`)**.

Exemple :

```
unsigned long tempsDebut = millis(); // on lance le chrono

if (millis() - tempsDebut >= 2000) {
    Serial.println("2 secondes se sont écoulées !");
}
```

- **Traduire et tester** l'algorithme suivant :

```
Mesurer la distance
Si la distance < seuil :
    Si un obstacle n'était pas précédemment détecté :
        Commencer le chrono
        Mémoriser qu'un obstacle a été détecté
    Sinon
        Si le chrono dépasse 2s
            Afficher "objet détecté depuis plus de 2s"
            Allumer led rouge
Sinon
    Mémoriser qu'aucun obstacle n'est présent
    Allumer led verte
```

- **Simuler** le fonctionnement et **conclure** sur la nécessité d'un programme de détection d'obstacle non bloquant pour le déplacement du chariot ?

## 4. Conception

Matériel à disposition pour cette partie :

- Ordinateur équipé de l'IDE arduino,
- 1 carte arduino équipée d'un shield moteur et 2 moteurs déjà branchés
- 1 programme "moteur.ino"

L'objectif de cette partie est de réaliser un programme de suivi de ligne en pilotant les 2 moteurs électriques.

Pour cela, il faut suivre la procédure ci-dessous :

- 1- **Lire** la documentation technique du contrôleur de moteurs
- 2- **Brancher** le capteur de suivi de ligne (comme dans la partie précédente)

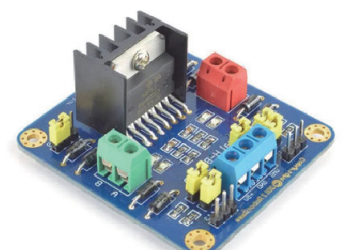


Figure 4 Contrôleur deux Moteurs DC L298 SeedStudio

A partir du programme fourni dans la documentation technique du contrôleur de moteurs (moteur.ino) :

3- **Réaliser** les fonctions suivantes permettant de piloter le chariot :

- **avance()** qui alimente les 2 moteur en marche avant à 100%
- **droite()** qui alimente le moteur gauche à 75% et le moteur droit à 25%
- **gauche()** qui alimente le moteur gauche à 25% et le moteur droit à 75%
- **stop()** qui arête les 2 moteurs

4- **Compléter** avec la prise en compte du capteur de suivi de ligne et **conclure** sur les modifications à apporter pour prendre en compte toutes les parties précédemment vues et le respect des différentes exigences du cahier des charges.