



- Vidéo " Notre-Dame_Colossus"
- Vidéo : "Notre-Dame_Robot"



- Identifier un besoin et énoncer un problème technique, identifier les conditions, contraintes (normes et règlements) et ressources correspondantes.(CT 2.1)
- S'approprier un cahier des charges (CT 2.3)
- Regrouper des objets en familles et lignées (CT 7.1)

S 23



TRAVAIL A RÉALISER

Problématique de départ : Suite à la catastrophe de notre Dame de Paris, des robots ont été utilisés pour aider les pompiers à éteindre l'incendie et d'autres pour sécuriser le bâtiment.



SÉANCE 1

Sur le document « Les outils pour élaborer le cahier des charges », répondre aux questions suivantes :

- 1 – A partir des vidéos fournies, expliquer les risques encourus par les pompiers lors de l'incendie et par les ouvriers actuellement sur le chantier.
- 2 – La problématique fut de déblayer les gravats afin de sécuriser l'intérieur de la cathédrale, pour répondre à cette exigence, un robot est utilisé. Indiquer quelles sont les solutions qui peuvent être envisagées afin de guider ce robot à l'intérieur.
- 3 – Compléter les diagrammes (SysML et bête à cornes) qui permettent d'identifier le besoin auquel répond le robot

SÉANCE 2

- 1 – Au cours de cette séance, vous devez trouver un nom à votre groupe qu'il faudra être capable d'expliquer à l'oral.

A partir de cet instant il faut vous répartir les tâches :

- ✓ Un élève : coordinateur du groupe
- ✓ Un élève : secrétaire, mettre au propre les réponses du groupe
- ✓ Deux élèves qui cherchent les réponses dans le document ressource sur le CDC

- 2 – A partir du cahier des charges fourni, répondre aux questions suivantes sur le document « Lecture du CDC » ,

- ✗ Quel est le robot qui va être utilisé ?
- ✗ Combien de moteurs sont installés sur le robot MBot ?
- ✗ Quels sont les capteurs disponibles sur le robot Mbot ?
- ✗ Quelles sont les dimensions maximum du robot ?
- ✗ D'où part le robot sur le circuit ?
- ✗ Ou le robot doit arriver à la fin du parcours ?
- ✗ Est-il obligatoire que le robot suive la ligne noire ?



- Vidéo "Cycle de vie d'un produit"
- Vidéo "Comprendre la courbe de cycle de vie d'un produit"



- Identifier un besoin et énoncer un problème technique, identifier les conditions, contraintes (normes et règlements) et ressources correspondantes. (CT 2.1)
- S'approprier un cahier des charges (CT 2.3)
- Regrouper des objets en familles et lignées (CT 7.1)

S 23

3 - Qu'est ce qu'un robot de téléprésence ?

4 – Rechercher des objets communicants uniquement par le son, puis par le son et l'image statique et enfin par le son et l'image dynamique

5 – Compléter les diagrammes du robot qui va être utilisé en classe (il doit éviter les obstacles, être capable de suivre une ligne noir et éventuellement reconnaître les couleurs au sol : Exigences système, diagramme pieuvre et contraintes. Pour cela il faut lire le cahier des charges de notre robot

6 – On veut créer un robot de téléprésence qui permette à la fois de suivre une mission (suivre un parcours, éviter des obstacles....de manière autonome : programmation) et d'être piloté à distance pour palier à toutes défaillance ou imprévus. Pour répondre à ce besoin , un smartphone va être fixé de manière temporaire sur le robot, il va falloir créer un support de téléphone qui devra être fixé sur le robot choisi. A partir de ce « cahier des charges » simplifié et du diagramme des exigences fourni "[SysMI Téléprésence](#)", indiquer quelles sont les contraintes que vous devrez respecter afin de respecter le cahier des charges.

SÉANCE 3 : rappel de programmation / Logigramme

1 – Lancer le logiciel Mblock, changer la langue si nécessaire et paramétrer en respectant les consignes suivantes :

Brancher le robot avec la prise USB

Mettre le robot sur on

Dans le menu **Choix de la carte** sélectionner **mBot (mCore)** et dans le menu **Choix des extensions** sélectionner **Makeblock**

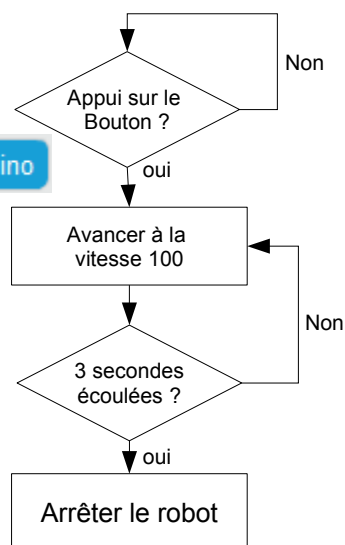


Dans le menu Connecter, sélectionner par port série (COM..)

2 – Réaliser le 1^{er} programme qui a pour objectif de faire fonctionner tous les actionneurs (moteurs, son, lumière)

- Le programme ne s'exécutera qu'une seule fois
- Attendre un appui sur le bouton
- Le robot avance pendant 3 secondes à la vitesse 100
- Il s'arrête (le robot avance à la vitesse 0)
- Faire un clic droit sur et téléverser dans l'Arduino et

Téléverser dans l'Arduino



- Attendre que le téléversement soit terminé.
- Débrancher le câble USB et poser le robot au sol et tester le programme en appuyant sur le bouton noir du robot



- Vidéo "Cycle de vie d'un produit"
- Vidéo "Comprendre la courbe de cycle de vie d'un produit"



- Identifier un besoin et énoncer un problème technique, identifier les conditions, contraintes (normes et règlements) et ressources correspondantes. (CT 2.1)
- S'approprier un cahier des charges (CT 2.3)
- Regrouper des objets en familles et lignées (CT 7.1)

S 23

3 – Réaliser le second programme qui reprend le précédent en ajoutant :

- Les Led s'allument en vert pendant 2 secondes puis s'éteignent
- Le robot émet le son G7 sur une durée entier
- Tester le programme

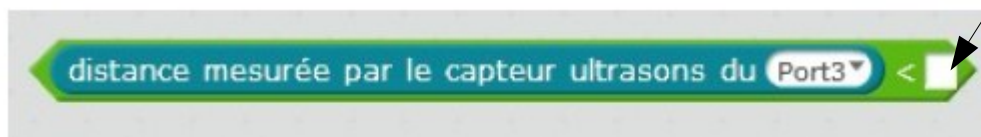
régler la DEL de la carte tout en rouge 0° vert 255° bleu 0°

4 – Réaliser le troisième programme : faire l'organigramme

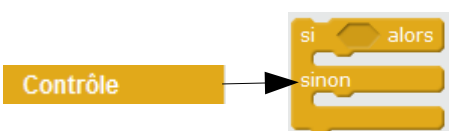
L'objectif est que le robot se déplace en ligne droite avec les LED vertes allumées. Il doit s'arrêter et allumer les LED en rouge quand il arrive à 20 cm d'un obstacle. Si on supprime l'obstacle il repart.



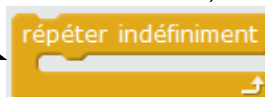
Astuce 1 : Il va falloir aller chercher le comparateur dans **Opérateurs** et insérer la distance mesurée que tu trouveras dans **Pilotage**



Astuce 2 : Il faudra utiliser « si sinon » que tu trouveras dans **Contrôle**



Astuce 3 : le programme se répète de manière indéfinie, il faudra donc utiliser une boucle que tu trouveras dans **Contrôle**



5 – Programme 4 : le robot doit réaliser un carré de 1m x 1m en utilisant le moins de briques possible.

6 - Programme 5 : faire suivre la ligne noire.

état du suiveur de ligne sur le Port 2

Etat du suiveur de ligne	Capteur gauche	Capteur droit
0	Noir	Noir
1	Noir	Blanc
2	Blanc	Noir
3	Blanc	Blanc