LU2IN013 - Groupe 3

PROJET ROBOTIQUE

GU David

SI MOHAMMED Yaniss

MANOUBI Taysir

YAN Nanlin

DATAYANA BOT CORP

Employé :

- GU David
- SI MOHAMMED Yaniss
- MANOUBI Taysir
- YAN Nanlin

Employeur :

- BASKIOTIS Nicolas
- SIGAUD Olivier

PLAN

1 - Introduction

2 - Présentation du Projet

3 - Difficultées rencontrés

4 - Conclusion

Introduction

1 - Introduction

A - Contraintes du Projet

B - Objectifs du Projet

C - Matériels fournit

A - Contrainte du Projet

- SCRUM Agile

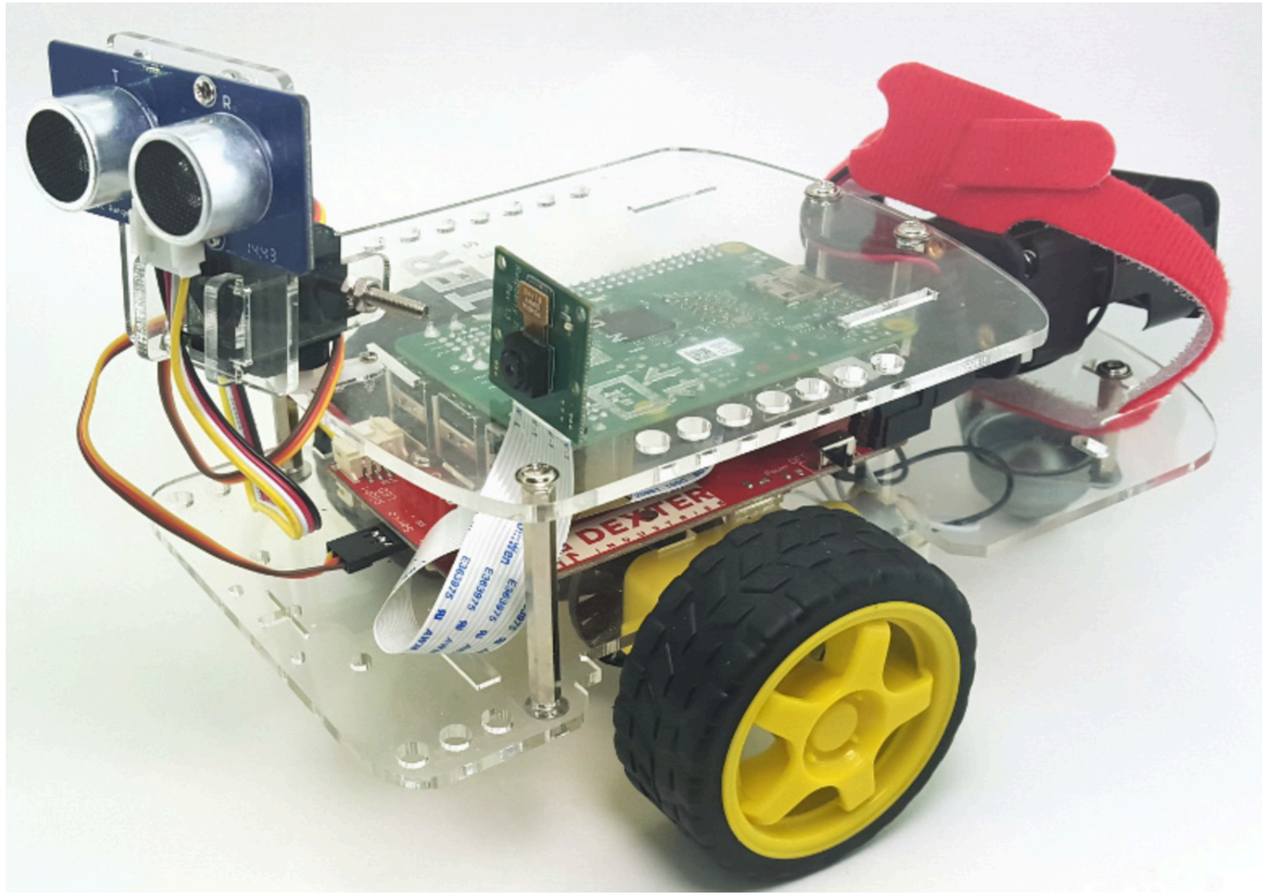
- Langage Imposé

- Autonomie

B - Objectifs du Projet

- Tracer un Carré
- Se Rapprocher d’un Obstacle
- Suivre une balise

C - Matériel fournis



- **Présentation du Robot**

C - Matériel fournis

- **Classe Robot2IN013**
- **Cours**

Présentation Projet

2 - Présentation du Projet

A - Architecture du Code

B - Présentation des stratégies

A - Architecture du Code

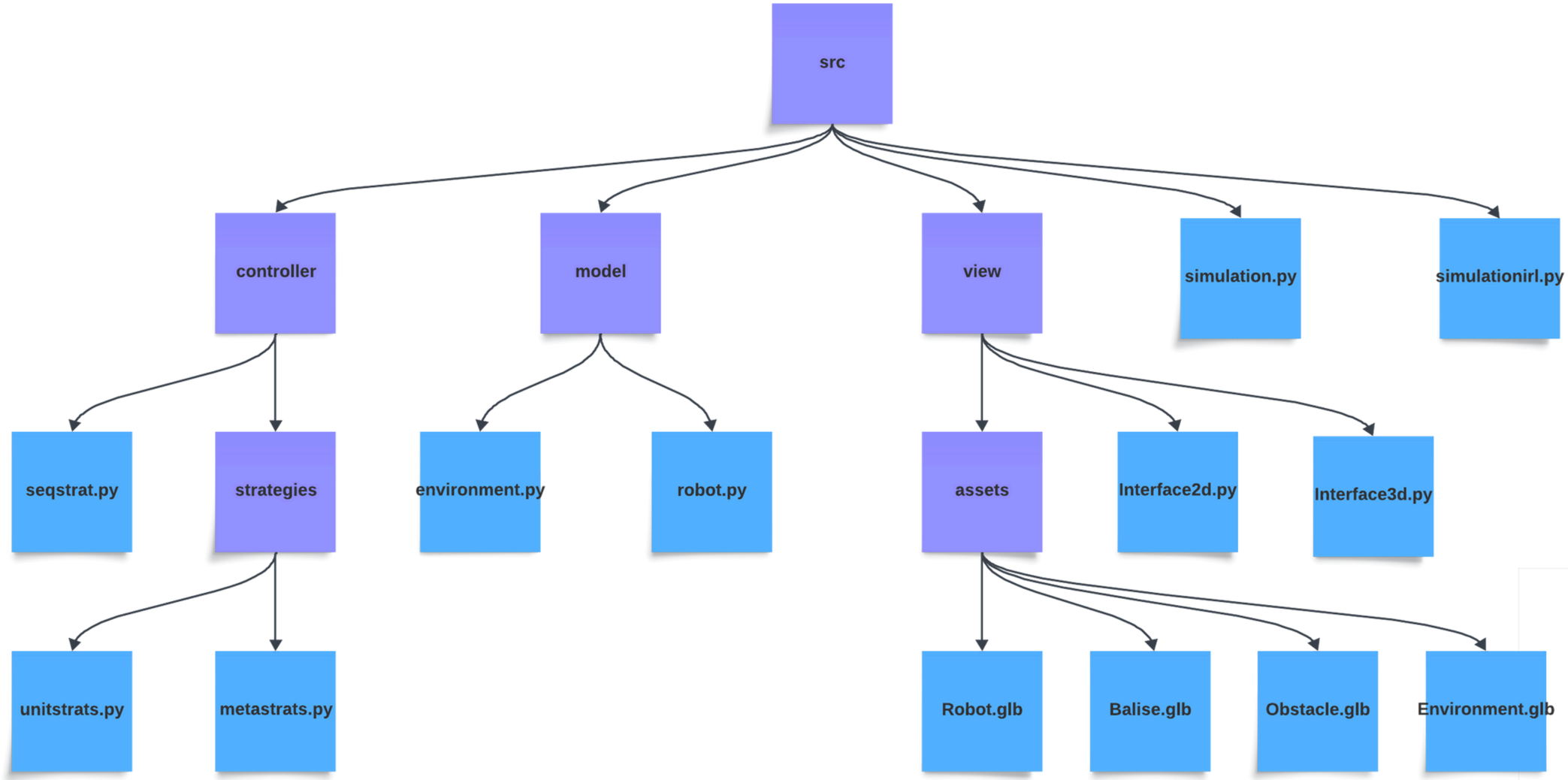


Figure 1 - Arborescence des fichiers

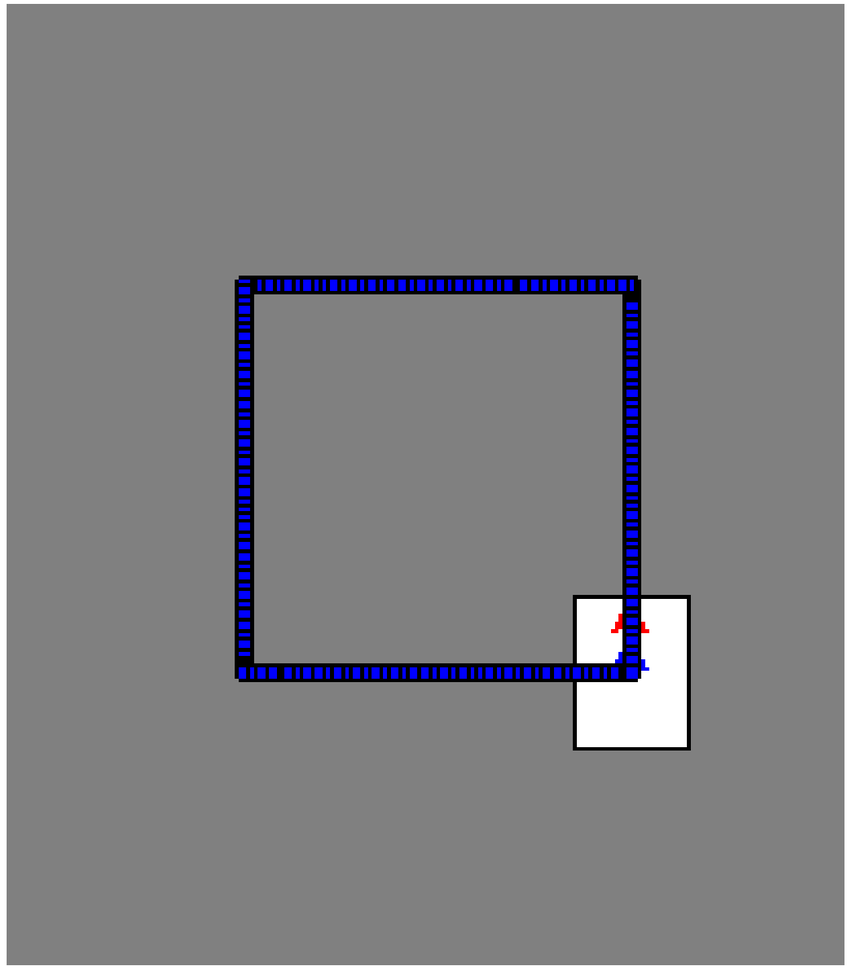
B - Construction des Stratégies

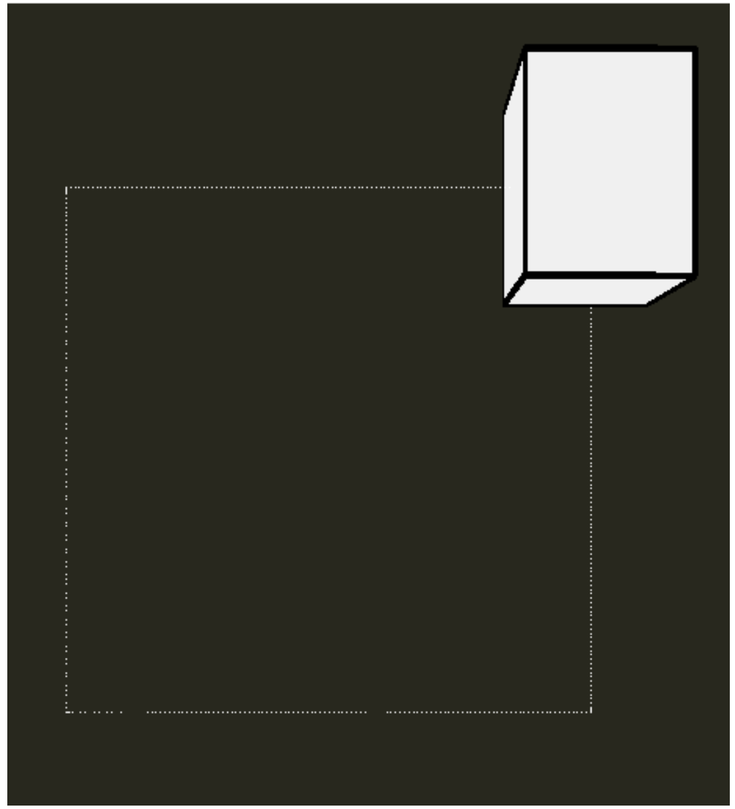
```python
# ----------------------------------------
class UnitStrat:
    """Class abstraite
    """
    def __init__(self):
        pass

    def start(self):
        pass

    def step(self):
        pass

    def stop(self):
        pass
    pass
# ----------------------------------------
```

```python
# -CONTROLLER-----------------------------------------------------------------------
class SequentialStrategy(UnitStrat):

    def __init__(self, strats: list[UnitStrat]) -> None:
        """
        """
        self.strats = strats

    def start(self) -> None:
        """Remet cur à -1 pour remettre le controleur sur la première instruction possible
        """
        self.cur = -1

    def step(self) -> None:
        """Parcours des instructions de la liste self.strats
        """
        if self.cur<0 or self.strats[self.cur].stop_strat:
            logger.info("Passage à la stratégie suivante")
            if self.stop():
                logger.info("Fin de l'ensemble des stratégies")
                return
            self.cur += 1
        else:
            self.strats[self.cur].step()

    def stop(self) -> bool:
        """Condition d'arrêt de step
        """
        return self.cur == len(self.strats)-1 and self.strats[self.cur].stop_strat
```

Figure 2 - Structure des stratégies

B - Faire un Carré





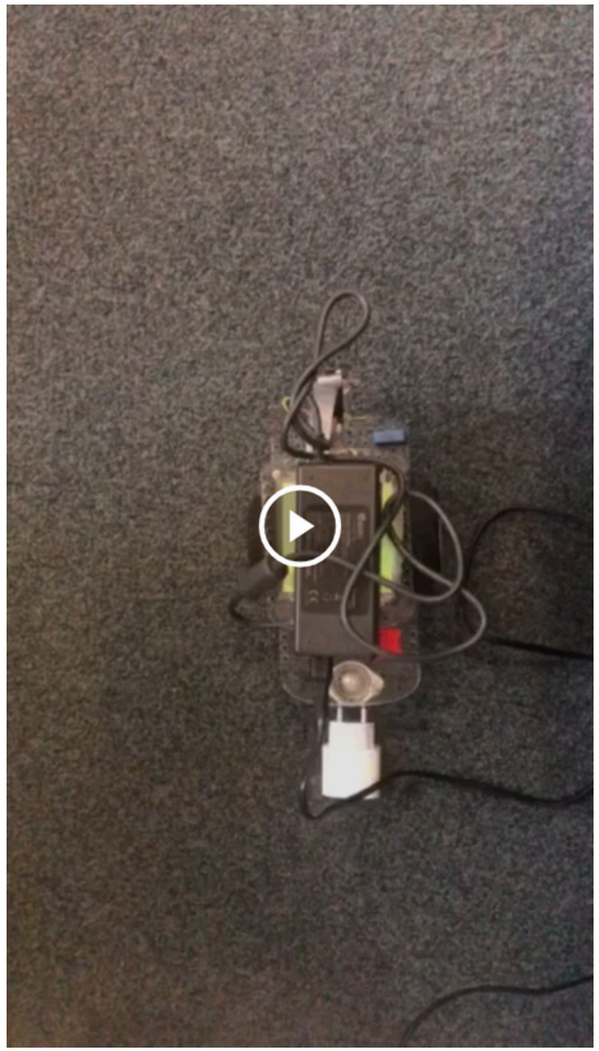


Figure 3 - Faire un Carré 2D,3D,Réalité

B - Se Rapprocher d’un Obstacle

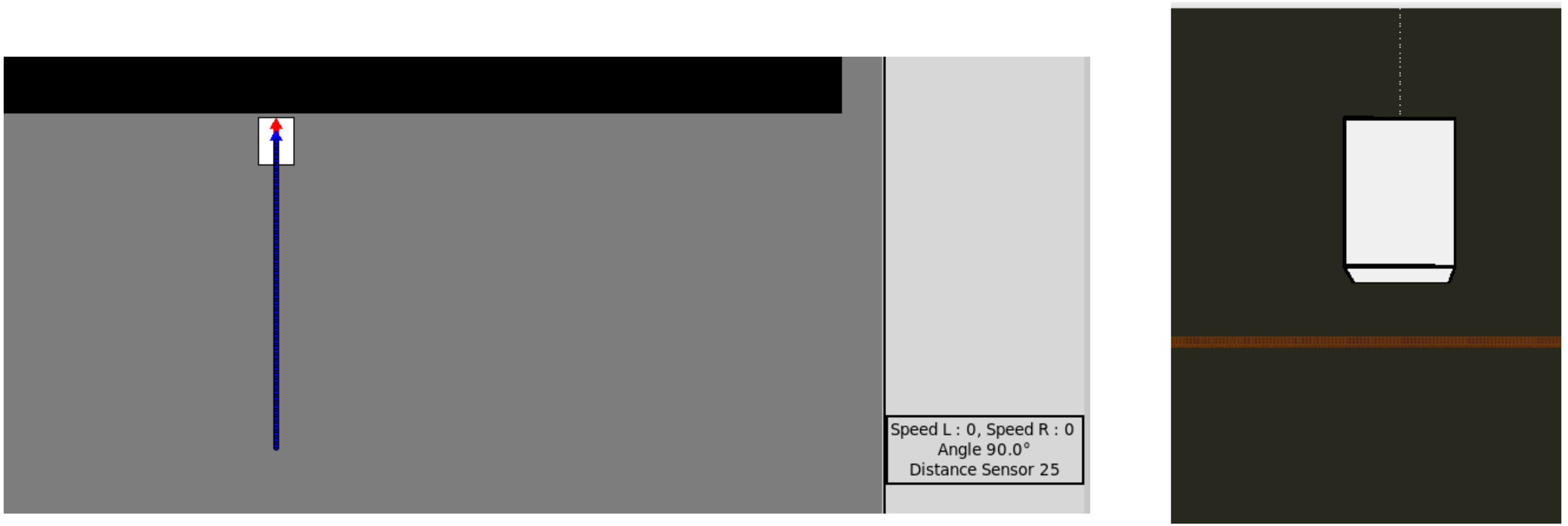


Figure 4 - Se rapprocher d’un obstacle 2D et 3D

B - Se Rapprocher d’un Obstacle

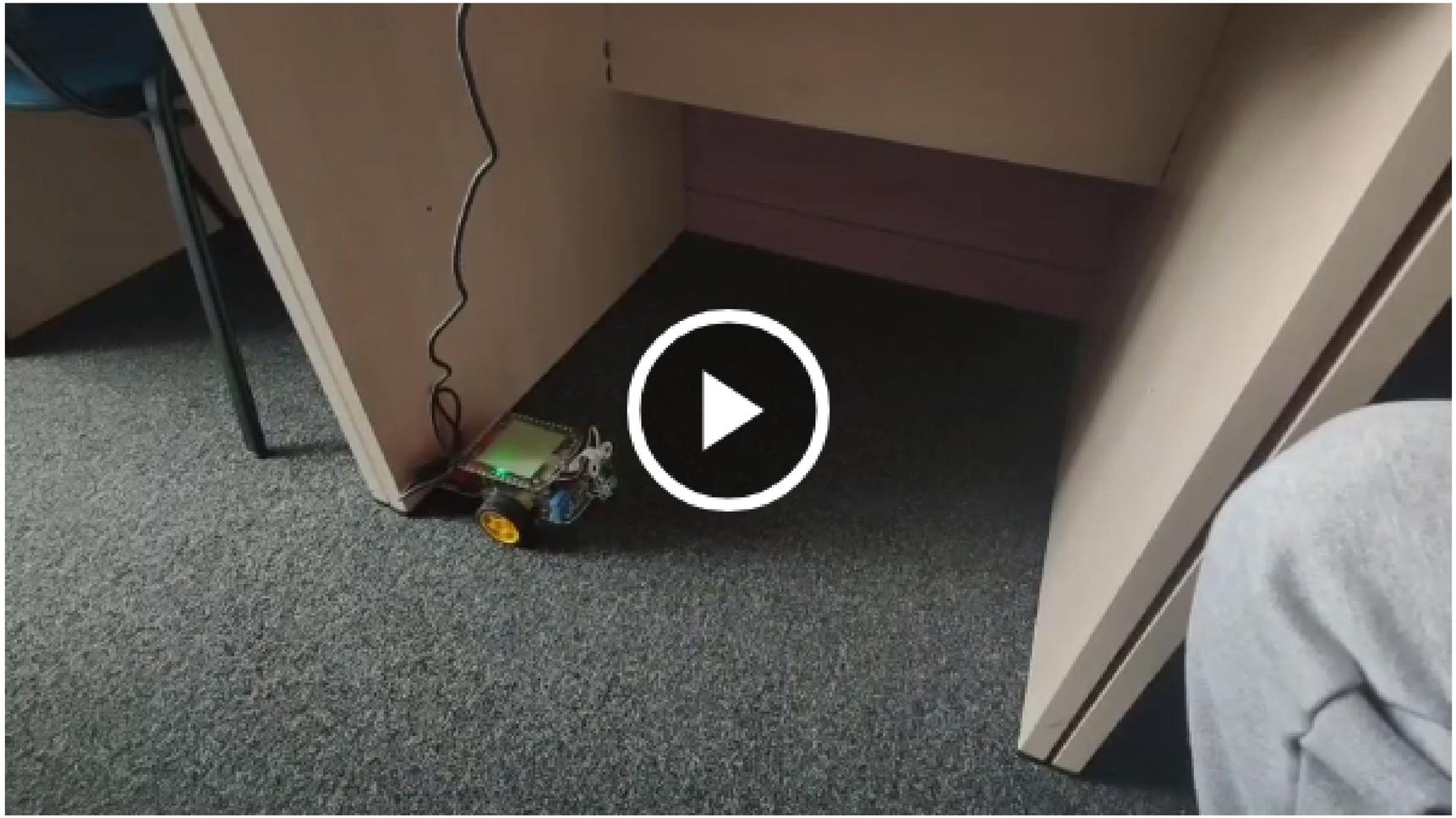


Figure 5 - vidéo “se rapprocher d’un obstacle”

Et la Reconnaissance de Balise ?

C - Suivre une Balise

- Pas achevé
- Reconnaissance fonctionnel

Problèmes Rencontrés

3 - Problèmes rencontrés

A - Organisation

B - Approche du Projet

C - Solutions aux problèmes

A - Organisation

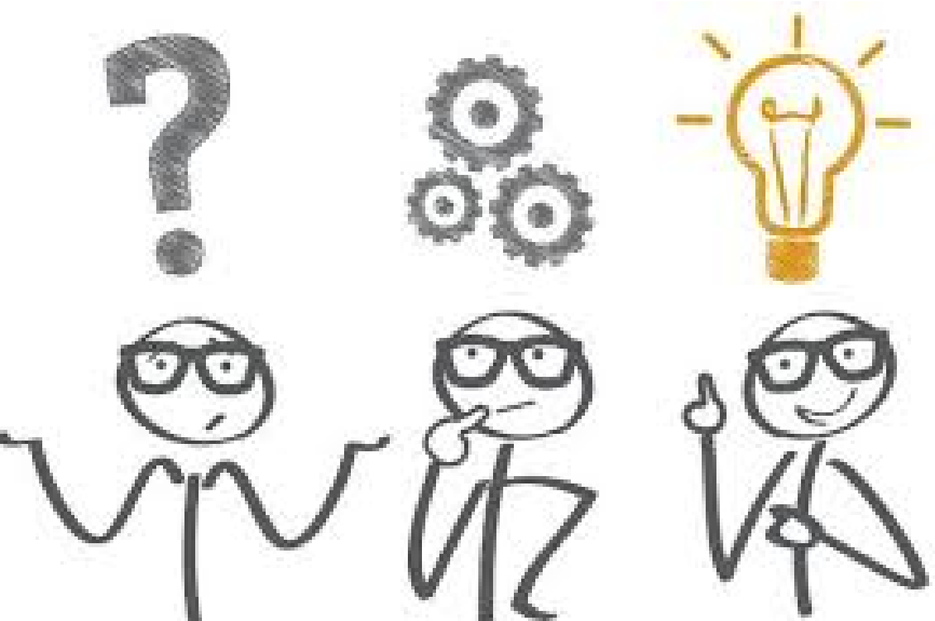
- Disponibilitées
- Communication
- Estimation
- Ressources limitées

B - Approche du Projet

- **Compréhension du sujet**
- **Visions hétérogènes**

C - Solutions aux problèmes

- **Templates Trello**
- **Réunions régulières**
- **Conseils**



CONCLUSION

Remerciements