

## **Récapitulatif des dépendances :**

- Frameworks et fichiers de lancement du lidar ld06 :  
[lien du dépôt](#)
- Ros 2 Humble :  
[lien de la documentation ros ubuntu](#)
- Slam toolbox :  
[Lien du dépôt de Toolbox](#)

## **Utilisation du framework ld06 :**

- Création d'un espace de travail et importation du dépôt de code.

```
cd ~  
  
mkdir -p ldlidar_ros2_ws/src  
  
cd ldlidar_ros2_ws/src  
  
git clone https://github.com/ldrobotSensorTeam/ldlidar_stl_ros2.git
```

- Accord des permissions au port série de la raspberry :

```
cd ~/ldlidar_ros2_ws  
  
sudo chmod 777 /dev/ttyUSB0
```

Nous avons remplacé ttyUSB0 par ttyS0.

- Compilation des fichiers de lancement :

```
cd ~/ldlidar_ros2_ws  
  
colcon build
```

- Ajout des fichiers de lancement à la source :

```
cd ~/ldlidar_ros2_ws  
  
source install/setup.bash
```

- Lancement du nœud ROS du lidar via son fichier de lancement :

```
ros2 launch ldlidar_stl_ros2 ld06.launch.py
```

## **Utilisation de la bibliothèque SLAM\_toolbox :**

Après configuration des fichiers de lancement de slam\_toolbox, on utilise la commande suivante pour traiter les données du lidar :

```
$ ros2 launch slam_toolbox online_async_launch.py
```

Ici, le fichier « online\_async\_launch.py » permet de récupérer les données du lidar sur le topic /scan et de les traiter pour permettre l’affichage sur le logiciel RVIZ2.

Une fois ces étapes faites, on lance le logiciel rviz2 que l’on peut configurer pour afficher le nuage de point généré par notre lidar.