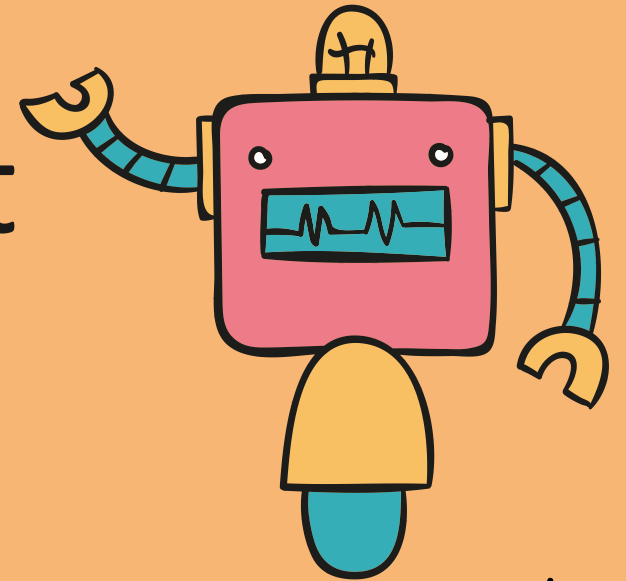
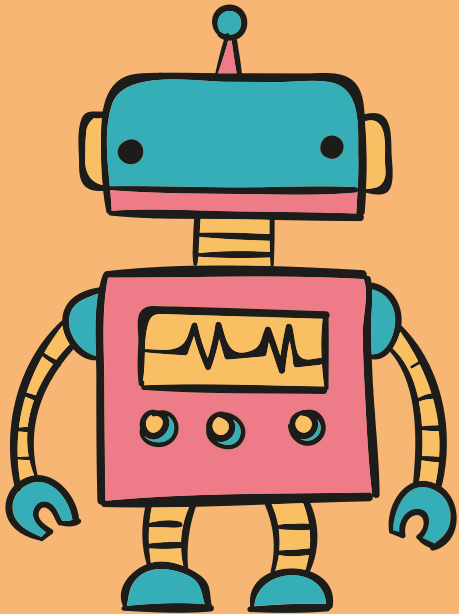


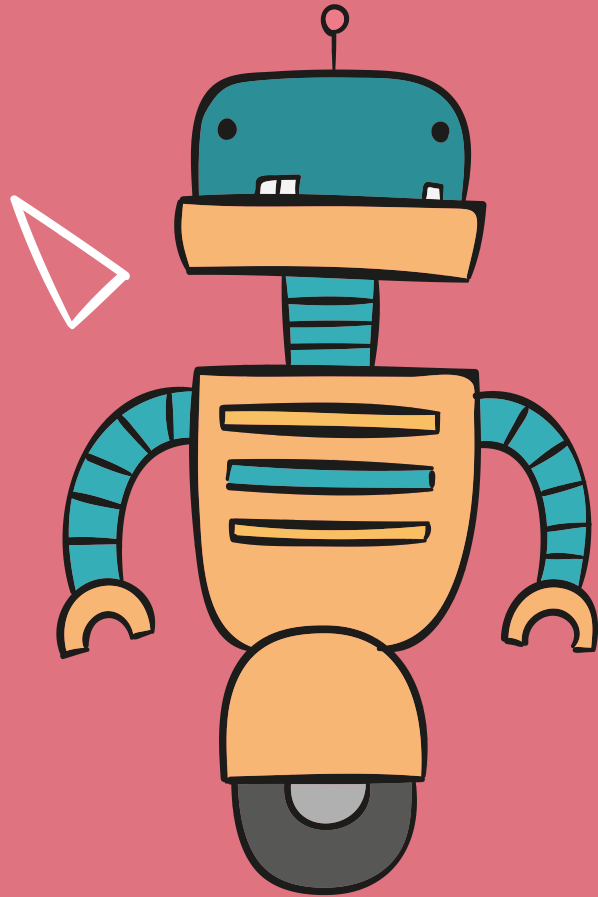
Grabby

The Warehouse Robot

Hendrikse Jeremy & Ortstadt Julius
Robo3

<https://github.com/JeremyHendr/GrabbyBotics>



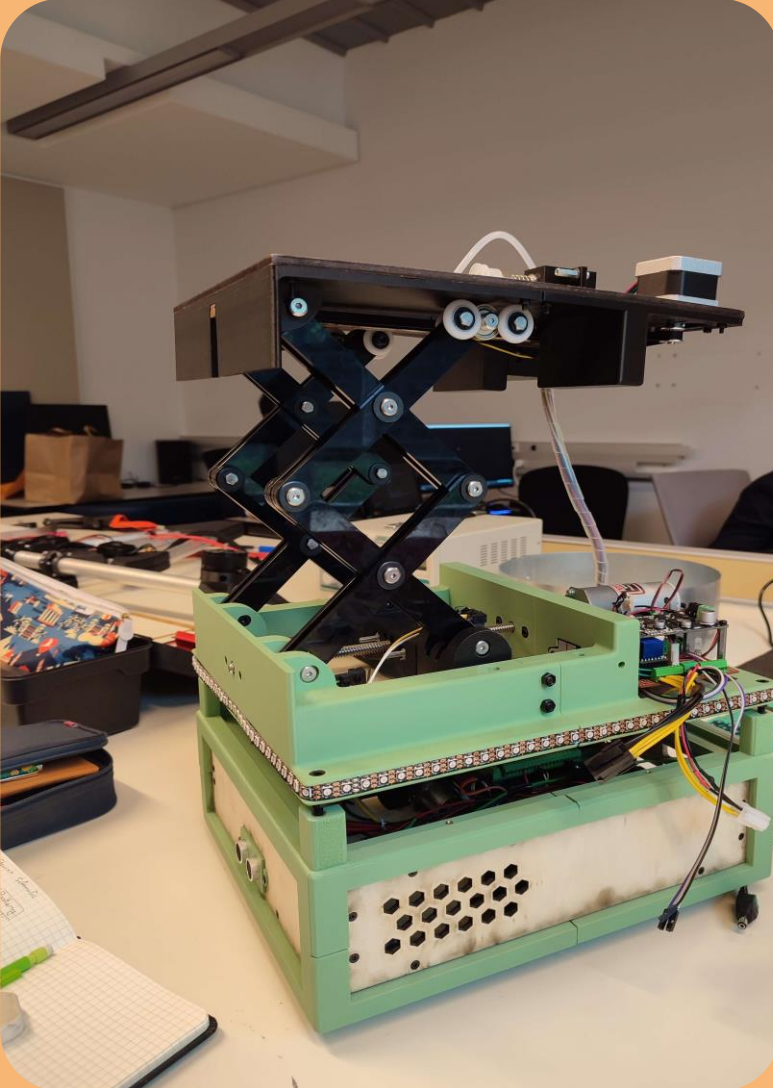
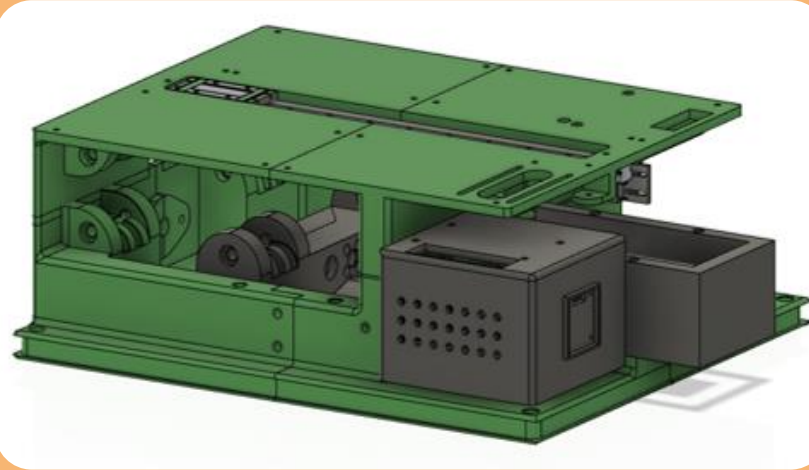
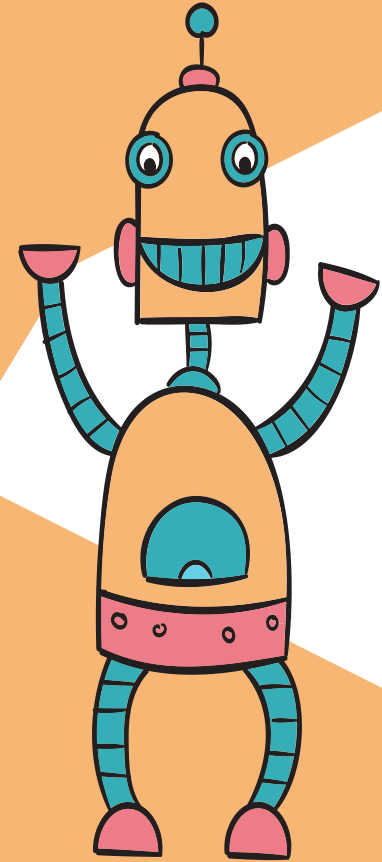


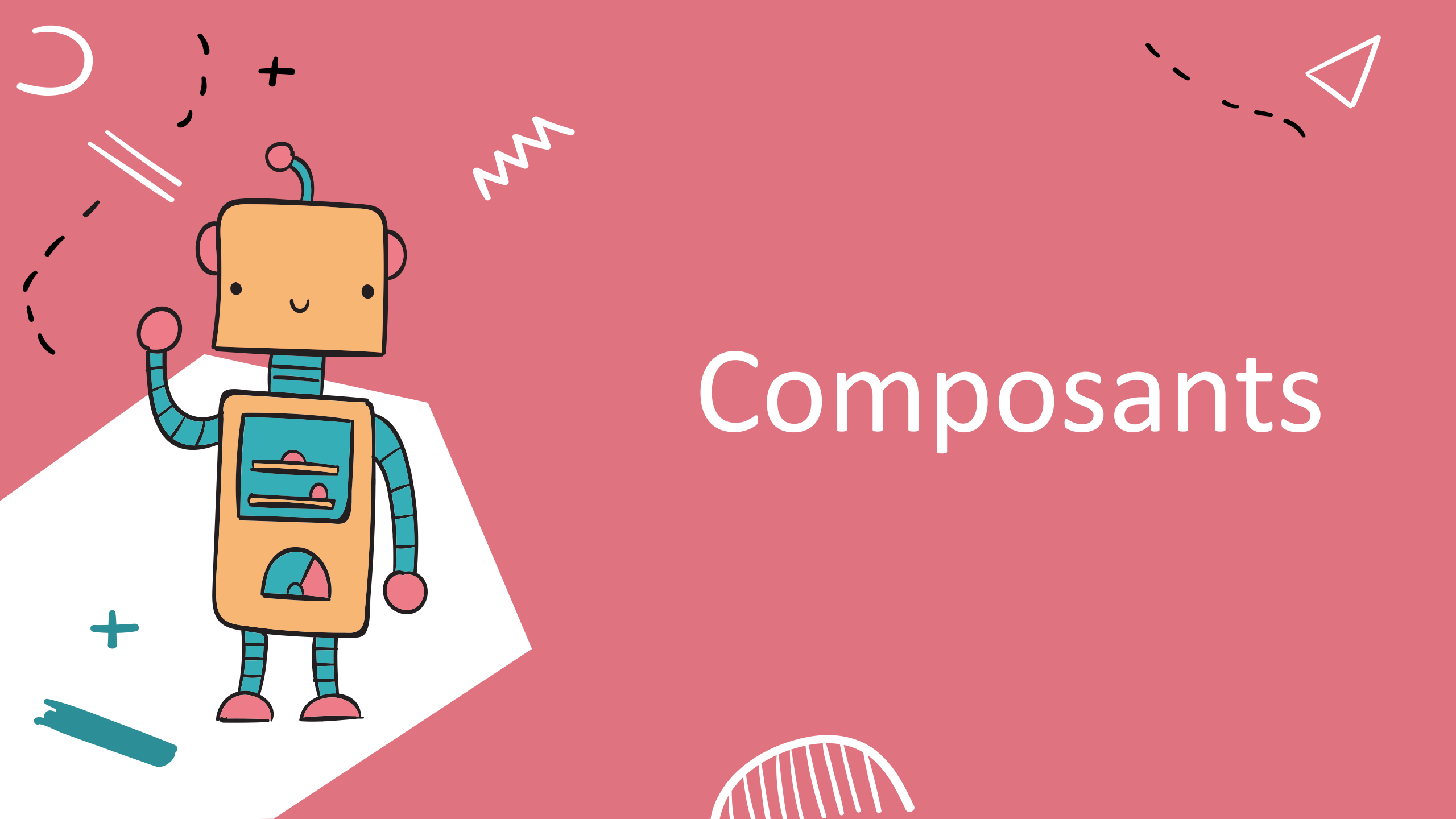
Présentation

Grabby est un robot d'entrepôt dont le but est de naviguer de façon autonome dans ce dernier et de récupérer des colis.

Grabby

+





Composants

Mobile Base

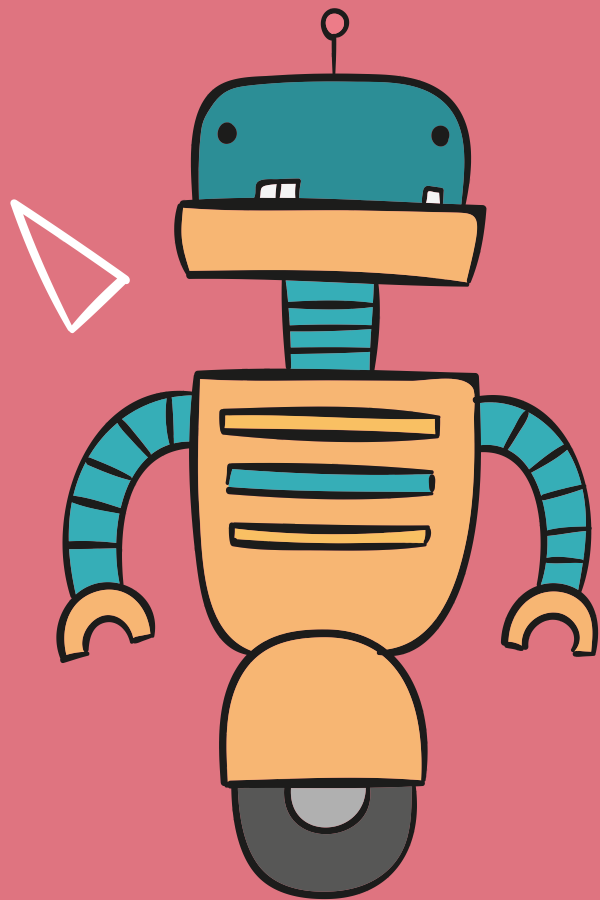
Voici une liste des composants installés dans la base mobile :

Composant	Quantité	Utilisation
Moteurs 99:1	2	Mouvements du robot
Nvidia Jetson Nano	1	Traitement de données
Arduino Mega	1	Tâches simples
Capteur Ultrason	1	Détection d'obstacles
RPLiDAR A1	1	Navigation
Convertisseur DC-DC	1	Système d'alimentation
Pont en H	1	Alimentation des moteurs
Fusible	2	Protection
Diode Zener	1	Protection
Ventilateur	2	Refroidissement
Switch	1	Démarrage du robot
Antenne	2	Connexion Jetson

Grabbing System

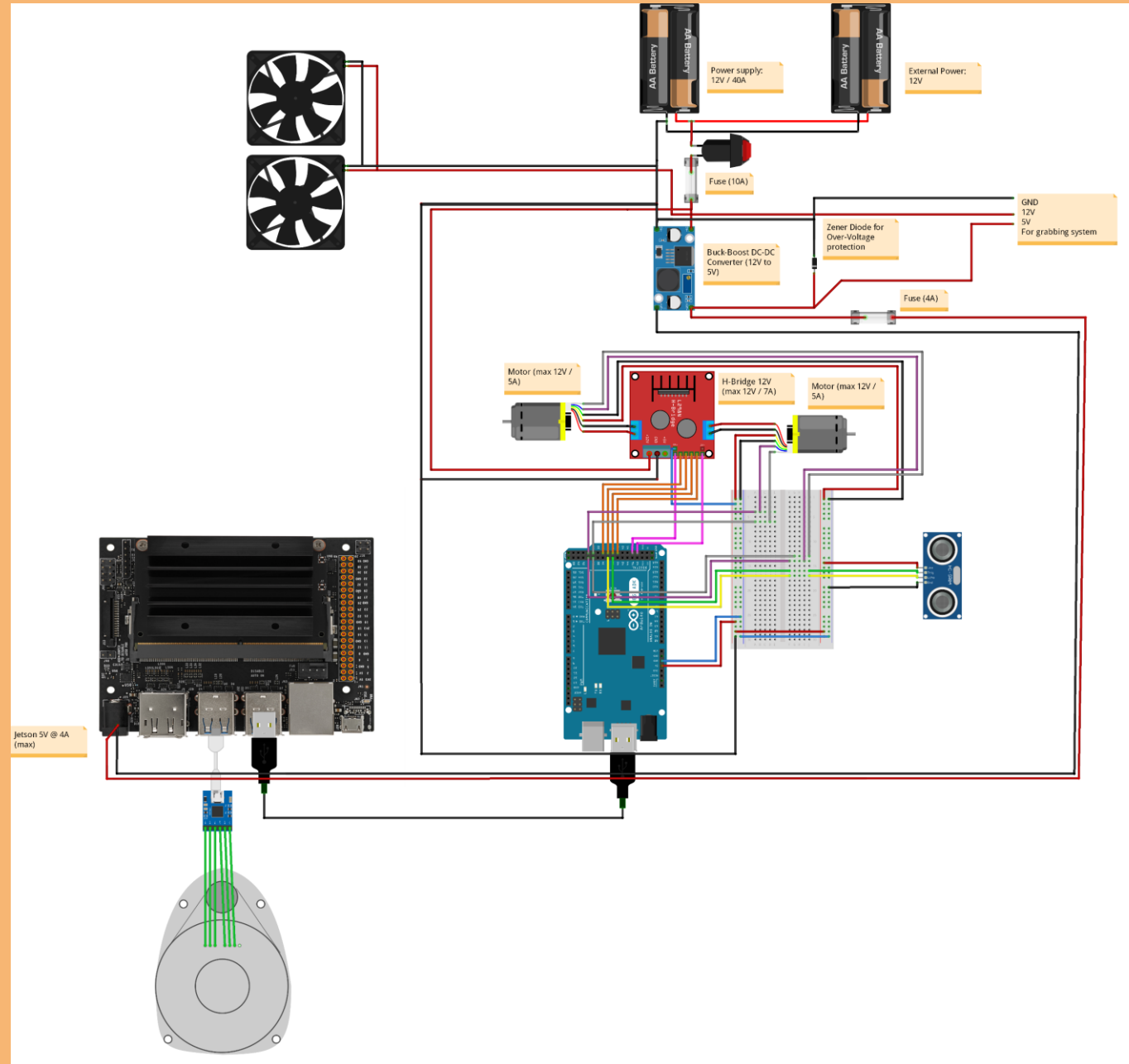
Voici une liste des composants installés dans le système de récupération :

Composant	Quantité	Utilisation
Moteur 148:1	1	Système de levage
Capteur de distance laser	1	Hauteur de la plateforme
Capteur de fin de course	4	Détecter la fin de course des différents mouvements
Pompe et électrovalve	1	Attraper et sécuriser les colis
Stepper Nema17	1	Déplacer le système de récupération
Ecran OLED	1	Afficher des informations sur l'état du robot
Stack de PCB	1	Regroupe tous les composants pour Controller la partie supérieure du robot

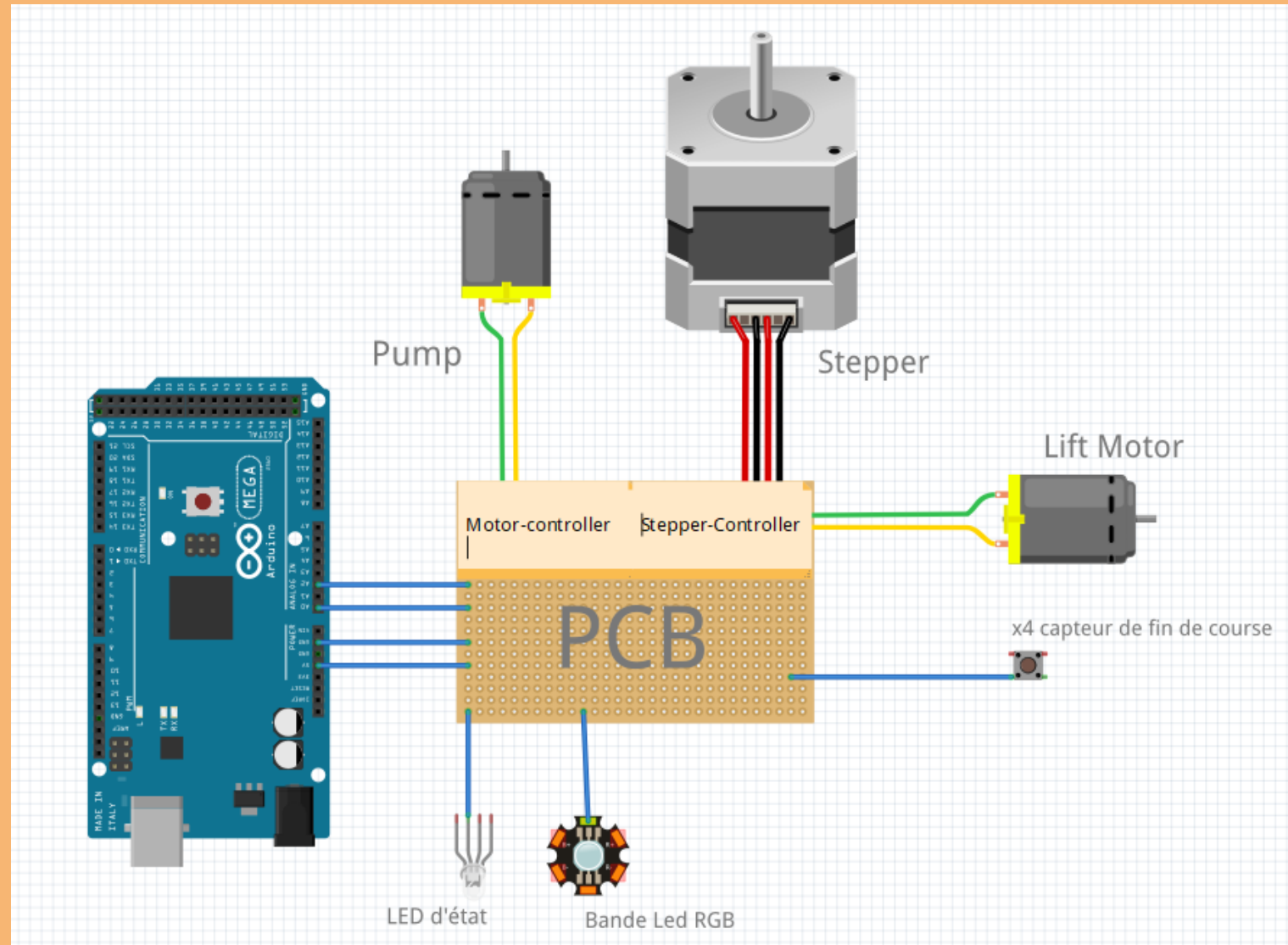


Montage

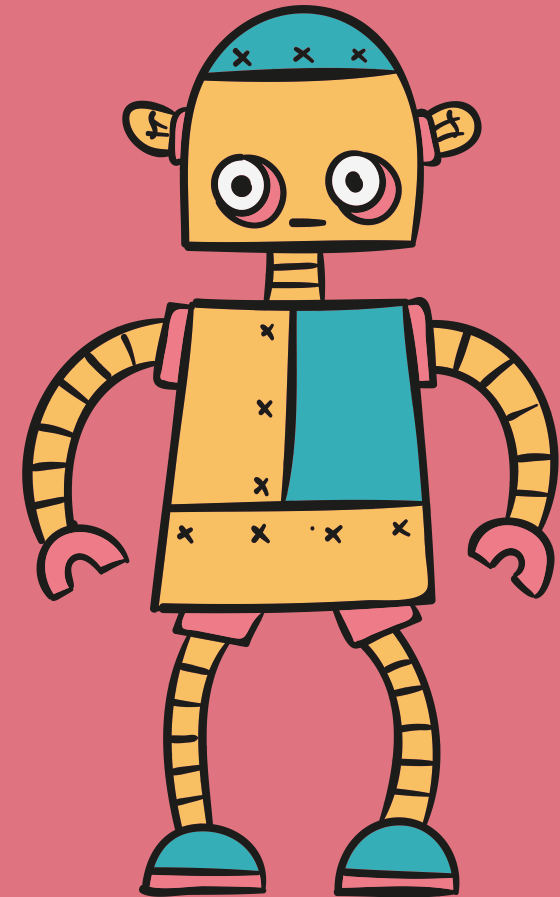
Mobile Base



Grabbing System



Code









Code

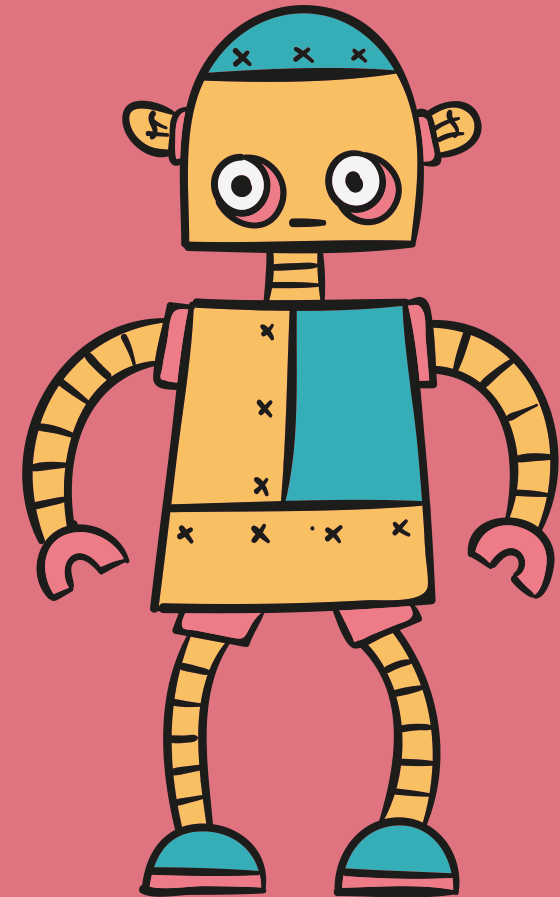


Afin d'avoir un maximum de possibilités et de flexibilités, nous avons opté pour la structure de code suivante:



- Système principal avec ROS Melodic.
 - Packages et nodes correspondants aux différentes tâches.
 - HECTOR-SLAM pour le mapping avec le LiDAR.
 - SLAM et RVIZ pour la navigation autonome.
 - OpenCV pour l'identification des boîtes (Code QR) et perception des distances à l'aide de la caméra stéréo.
 - Code Arduino pour effectuer les commandes de la Jetson et les tâches “bas niveau”.
- 
- 
- 

Problèmes et Solutions









Les problèmes et les solutions



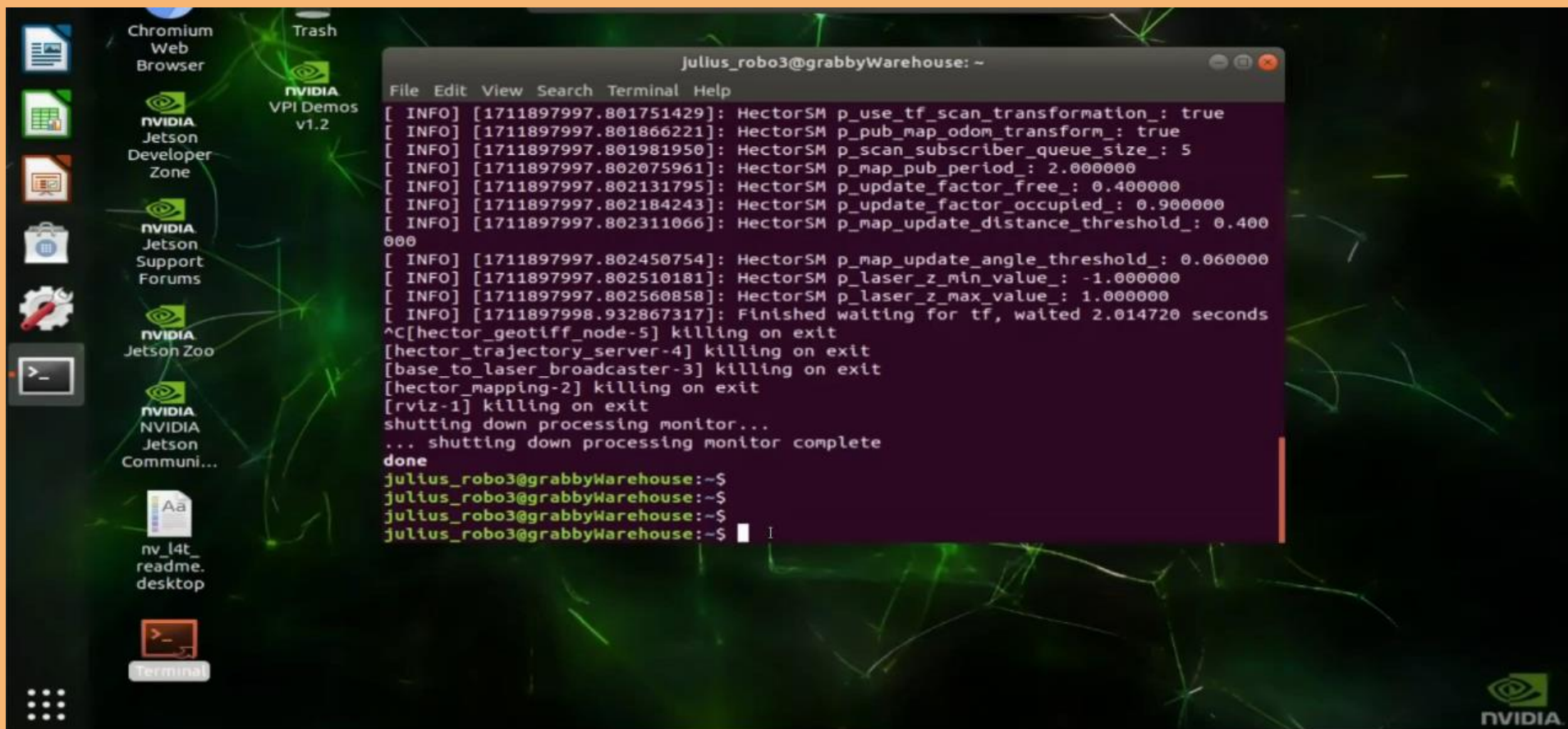
Différents problèmes ont été rencontrés au cours de ce projet. Nous avons réussi à en résoudre mais certains persistent toujours. Voici une liste des principaux problèmes que nous avons rencontrés:



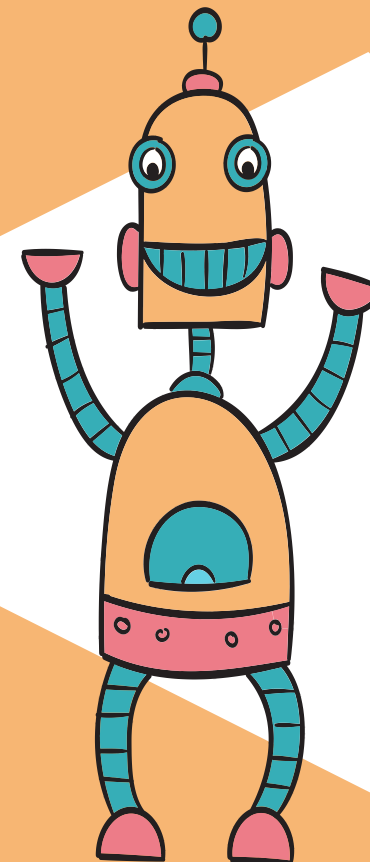
- Navigation par suivi de ligne.
 - Navigation autonome avec SLAM.
 - Détection et contournement d'obstacles grâce au LiDAR.
 - Mapping avec HECTOR-SLAM.
 - Calibrage de la caméra stéréo pour la perception des distances et des objets.
 - Système d'alimentation du robot.
 - Conception et assemblage de certaines pièces.
- 
- 
- 

Démonstration

Test de mapping

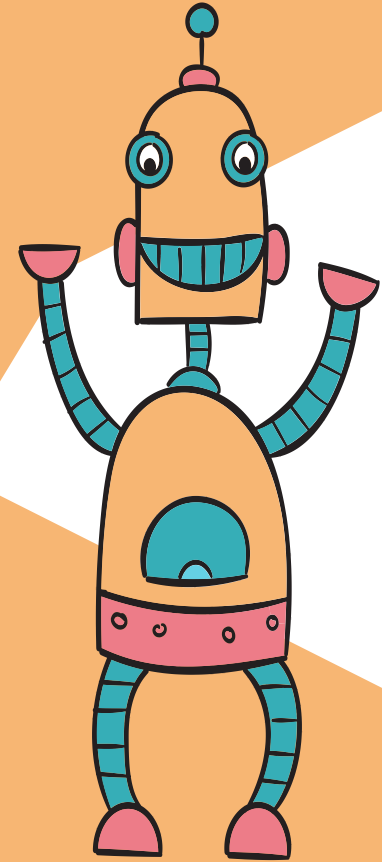


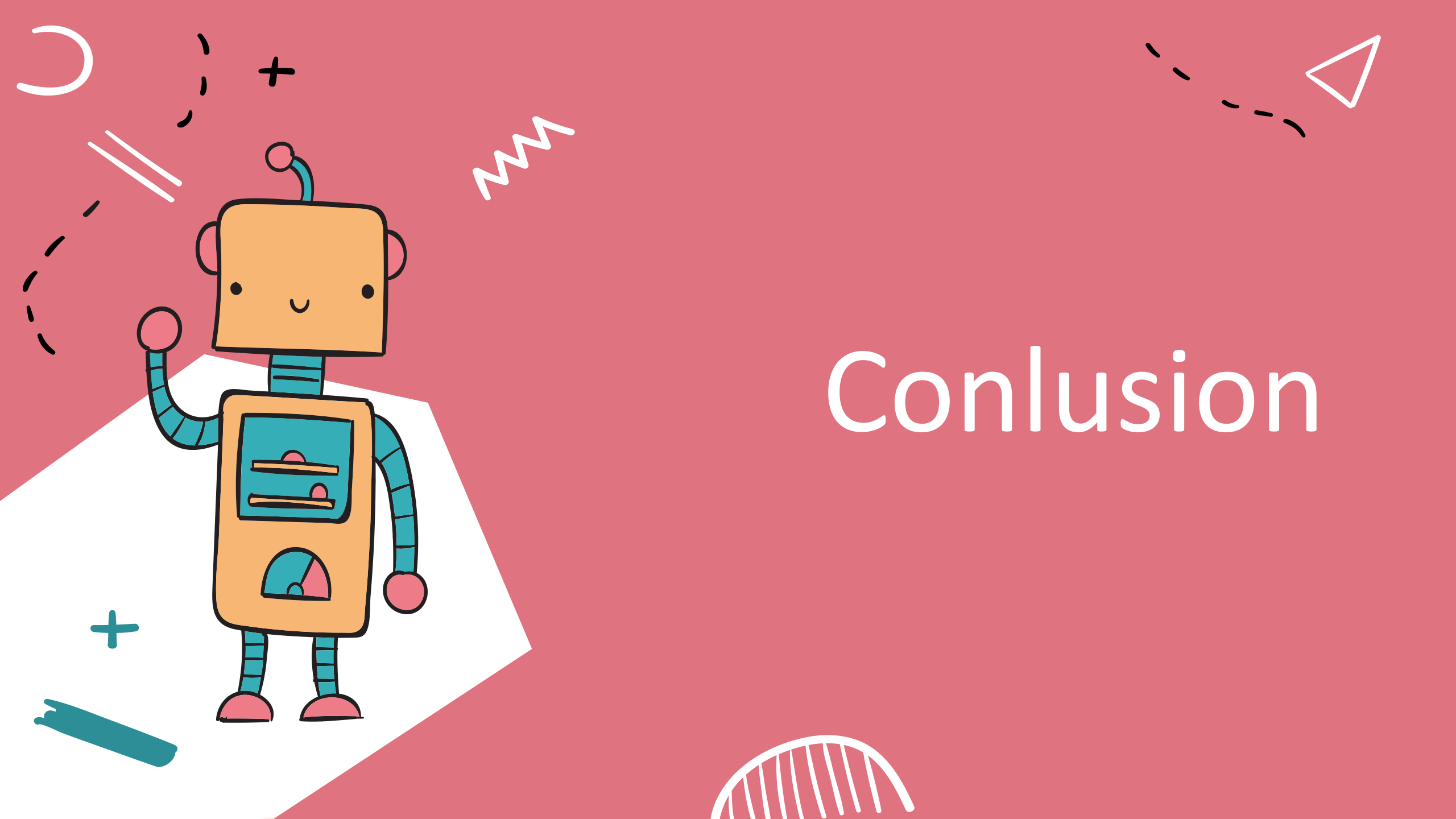
```
julius_robo3@grabbyWarehouse: ~  
File Edit View Search Terminal Help  
[ INFO] [1711897997.801751429]: HectorSM p_use_tf_scan_transformation_: true  
[ INFO] [1711897997.801866221]: HectorSM p_pub_map_odom_transform_: true  
[ INFO] [1711897997.801981950]: HectorSM p_scan_subscriber_queue_size_: 5  
[ INFO] [1711897997.802075961]: HectorSM p_map_pub_period_: 2.000000  
[ INFO] [1711897997.802131795]: HectorSM p_update_factor_free_: 0.400000  
[ INFO] [1711897997.802184243]: HectorSM p_update_factor_occupied_: 0.900000  
[ INFO] [1711897997.802311066]: HectorSM p_map_update_distance_threshold_: 0.400  
000  
[ INFO] [1711897997.802450754]: HectorSM p_map_update_angle_threshold_: 0.060000  
[ INFO] [1711897997.802510181]: HectorSM p_laser_z_min_value_: -1.000000  
[ INFO] [1711897997.802560858]: HectorSM p_laser_z_max_value_: 1.000000  
[ INFO] [1711897998.932867317]: Finished waiting for tf, waited 2.014720 seconds  
^C[hector_geotiff_node-5] killing on exit  
[hector_trajectory_server-4] killing on exit  
[base_to_laser_broadcaster-3] killing on exit  
[hector_mapping-2] killing on exit  
[rviz-1] killing on exit  
shutting down processing monitor...  
... shutting down processing monitor complete  
done  
julius_robo3@grabbyWarehouse:~$  
julius_robo3@grabbyWarehouse:~$  
julius_robo3@grabbyWarehouse:~$  
julius_robo3@grabbyWarehouse:~$
```



Démonstration

Mouvements et fonctionnalités





Conlusion