Final Design Review

School of Engineering

MSc Robotics AERO62520 Robotic Systems Design Project

The University of Manchester

15th March 2024

Group: 5

Authors: Alexander Morley, Michalis Iakovides, Ziyi Cheng, Geetik Mamillapalli

Contents

List of Figures

List of Tables

1 Introduction

1.1 Summary of the problem statement and objectives

This document serves as the preliminary design review for team 5's project, the development of an advanced robot specifically engineered to autonomously retrieve a desired object from an environment. The essence of this review is to examine the initial design concepts, ensuring alignment with the standards and detailed specifications established in the Design Requirements Analysis coursework shown in Section 7.

The problem statement that team 5 is challenged with is to:

"Develop a robot which can autonomously retrieve an object from an environment."

To achieve this, team 5's project centres around the creation of a sophisticated mobile manipulator, combining a base platform of the Leo Rover and an onboard manipulator, the Trossen PincherX 150. These two main robotics systems are complemented by an array of essential peripherals, such as a RPLidar A2M12 and a Intel Realsense Depth Camera. The core objective of this endeavour is not only to conceptualise but to bring into existence a robot that can independently navigate and interact to perform the targeted task of object retrieval in an unmapped environment.

The review herein will delve into various details of the robots design, and the use of components. Explaining the integration strategies the team has put into effect to transform the Leo Rover with its manipulator counterpart, into a robot that adheres to the predetermined requirements.

1.1.1 Summary of how the team addressed feedback from Preliminary Design Review

- System Architecture : The systems architecture was slightly changed to include what design requirements were achieved with each component.
- Electrical System: The electrical systems diagram was updated to consider current in the system, and if the battery has sufficient power to energise the whole robot.
- Mechanical Design: The mechanical design discussion was elaborated, explaining the fabrication of the Leo Rovers platform with regards to LiDAR Sensor Obstructions. Additionally the two payload containers were integrated into the structure for object collection. The payload containers are equipped with Dynamixal motors to pivot the inclined storage container with collected objects that will be dumped in front of the fore wheels of the Rover. The structural strength of the platform with the chosen manufacturing material was calculated and simulated in the review to validate the strength of the platform under a maximum payload.
- Software systems: The team considered the progression of the robots algorithm and implemented a behaviour tree for the system to use for navigation and exploration, a flow chart is used to expand on the states that the algorithm uses for its tree.
- Design Requirements analysis: The team also improved the analysis of the solution provided to the problem statement, by implementing more specificity in the requirements. Additionally, a success criteria that summarised the functional and product design requirements was formed, and can be used to validate requirements and assess the quality of the solution based on its success.

1.1.2 Summary of any modifications made since Preliminary Design Review

After completing the Preliminary Design Review , our team made a strategic enhancement to our "object_detect" node, specifically tailored to improve the robotic system's interaction with their environment. Initially designed to integrate with the camera on the Leo Rover, the team shifted our focus towards employing the Realsense Camera D435. This change was motivated by the need for clearer visual and depth data, which is essential for object manipulation tasks. The "object_detect" node is now adept at subscribing to the Realsense Camera D435's colour and depth data streams. By implementing colour-based segmentation, it detects objects within the camera's view. The pivotal advancement comes with the node's capability to calculate the 3D positions of these objects using the depth information provided by the Realsense Camera. This enriched spatial data is then published to other components within the ROS2 framework, significantly enhancing the robotic system's ability to intelligently and precisely interact with its surroundings. To accommodate this upgrade, it was imperative to adjust the algorithm to subscribe to the new data nodes published by the depth camera. This adjustment ensures integration and optimizes the system's performance, making it an essential step in our continuous efforts to enhance the robot's functionality and task efficiency.

The structure of the robot was changed to allow for a mechanism to unload the collected objects onto the end goal by using a mechanism that deploys the collected wooden cubes in front of the robot. Considering the collected objects are not delicate this system can be utilised without concern of breaking the collected object or failing the mission.

The manipulator can demonstrate basic functionality with python launch files, that enable the robot to use the joint state topic to pass along geometric pose messages of the co-ordinates of the object being collected. These co-ordinates will be passed by the object detection node as geometry messages to be configure the robotic manipulator joint states into a position where it could collect the object.

Regarding the LiDAR, our team uses a different method to prevent the detection of the robot's wheels. The old approach was still using the RPLIDAR package to read the scans from the '/scan' topic and the data was processed by an additional filtering node before publishing the data for SLAM. Now, we make use of the RPLIDAR package's configuration files which allows us to set the minimum detection distance to 0.3 meters. This way the process becomes simpler and easier to debug.

Regarding the IMU, our team decided to use the Leo core board's integrated IMU for implementing the extended kalman filter instead of relying on the external Sparkfun IMU. The decision was made due to the fact that the Leo OS publishes the imu reading through the '/imu_raw' topic, which takes away the need of creating a new node and the need to read or filter the data coming from an external IMU.

For navigation and exploration the team implemented a wavefront frontier algorithm, which is based off behaviour trees following a state machine model. It allows the robot to explore an unknown environment, as the object that needs to be found could be anywhere inside this environment. Using this method increases the probability of finding the object regardless of it position in the unknown environment.

1.2 Sustainability Checklist

This document assesses the sustainability of our automatic retrieval robot. The assessment follows the 10-point checklist derived from the BS8622 Guide to Robot Sustainability[1].

1.2.1 Materials

Our robot incorporates various materials, including metals for the manipulator and LeoRover, plastic components for part of the casing, and electronic components for sensors and cameras. In line with the growing trend towards sustainability in technology, the team is exploring the use of recycled plastics and sustainably sourced metals to reduce the environmental impact of our project. This approach is inspired by the broader industry movement towards more sustainable practices[2], although our application is on a much smaller scale suitable for a course project.

1.2.2 Software

The robot's software has been optimized for low power consumption and efficient data processing. Algorithms for object detection and navigation have been refined to minimise computational load, thereby reducing energy usage.

1.2.3 Energy

The robot is powered by rechargeable battery, which allows for the battery to be reused, after the completion of the project.

1.2.4 Waste

We aim to catalogue and responsibly dispose of electronic waste, adhering to our university's e-waste recycling policies[3][4]. By doing so, the group will not only adhere to sustainability principles but also educate ourselves and our peers on the importance of responsible waste management in technological projects.

1.2.5 Emissions

Given the nature of our project, direct emissions from the robot are minimal. However, we're aware of the broader environmental implications, such as the carbon footprint associated with manufacturing the components being used. To address this,the team will make efforts to source components locally where possible, reducing transportation emissions, and we're exploring the use of recycled or second-hand parts to further reduce our project's environmental impact.

1.2.6 Communications

Data transmission is optimized to send only necessary information, reducing the energy required for data storage and processing. Also, the team is implementing data compression techniques in testing to further reduce the data footprint.

1.2.7 Modularity

Our robot's design strategically emphasizes modularity, utilizing 3D printed components to enhance this aspect significantly. This approach allows for streamlined repairs and straightforward component replacement, ensuring long-term usability and adaptability.

1.2.8 Location/Placement

Our robot's operational testing is confined to lab environments, minimizing the need for transportation and its associated environmental impacts. When transportation is necessary, thee team opt for walking or using public transit to move the robot to different testing locations, further reducing our project's carbon footprint.

1.2.9 Maintenance

We've implemented a straightforward maintenance plan for our robot, emphasizing regular checks and diagnostics to prevent malfunctions. This plan is designed to be executed with minimal resources, using standard tools available in our lab, ensuring the robot's longevity and reliability without excessive resource use.

1.2.10 Repurposing

At the conclusion of our project, the team intends to disassemble the robot and repurpose its components for future projects. This practice not only reduces waste but also provides valuable learning opportunities for the team, reinforcing the principles of sustainability through practical application.

1.2.11 Conclusion

Our automatic retrieval robot is designed with sustainability at its core, from material selection to end-of-life repurposing. The team is committed to ongoing improvements in sustainability and welcome feedback and suggestions for further enhancements.

1.3 Cyber Security Considerations

Some considerations were made by the team for the cyber security of this project, which the possibility of another individual connecting to the rover wirelessly and by manually plugging it in. The probability of a cyber attack is quiet low however, to counter act this possibility there are two passwords put in place, one prevents a wireless connection to the rover and the other protects the hardware connection to the Intel NUC. These passwords are changed every week during the team meetings in the instance that it is leaked to anyone outside the team.

2 System

2.1 Architectural Block-Diagram

The following block-diagram illustrates the connection between all the components of the robot. The components can be categorised into the following categories: computing, sensors, peripherals, and power. The colour code in figure 1 can help guide as to which category each component belongs.

- Battery: Provides the necessary energy for the entire robotic system to function autonomously. (Design Requirement P-21)
- LEO Powerbox: Acts as a centralised power management unit, regulating and distributing power to various components.
- LEO Core Board: Drives the Rover's motors and interprets feedback from the encoders, enabling precise control over the movement and navigation of the robotic system.
- DC Motors with Encoders: Offers precise control and feedback for the robot's movement. (Design Requirement P-9)
- Raspberry Pi 4 Model B: Used as a computing platform for reading camera and IMU data, while exchanging sensor data with the Intel NUC and wheel velocity data with the LEO Core Board.
- Leo Core built-in IMU: Provides real-time information about the robot's orientation, acceleration, and angular velocity, which are used in navigation, and motion control.(Design Requirement P-6)
- Camera: Enables visual perception for the robot, which will be used for tasks such as object recognition and navigation. (Design Requirement P-1)
- 2.4GHz WiFi Modem with External Antenna: Facilitates wireless communication, enabling the robot to send and receive data, commands, and updates, during development and testing. (Design Requirement P-25)
- Intel NUC: Powerful computing unit for processing data and performing SLAM (Simultaneous Localisation and Mapping) and navigation algorithms.
- RPLIDAR A2M12: Provides 360-degree laser scanning, allowing the robot to create detailed 2D map of its surrounding environment. (Design Requirement P-2)
- RPLIDAR USB Adapter Board: Acts as an interface between the RPLIDAR sensor and the Intel NUC, facilitating seamless communication and data transfer.
- INTEL Realsense Depth Camera: Enables the robot to perceive its surroundings in three dimensions, and perform tasks such as object recognition and depth sensing. (Design Requirement P-1)
- Trossen PincherX 150 Manipulator: Robotic arm which is attached on the top of the rover. It will be used for grabbing the object when detected. (Design Requirement P-10)

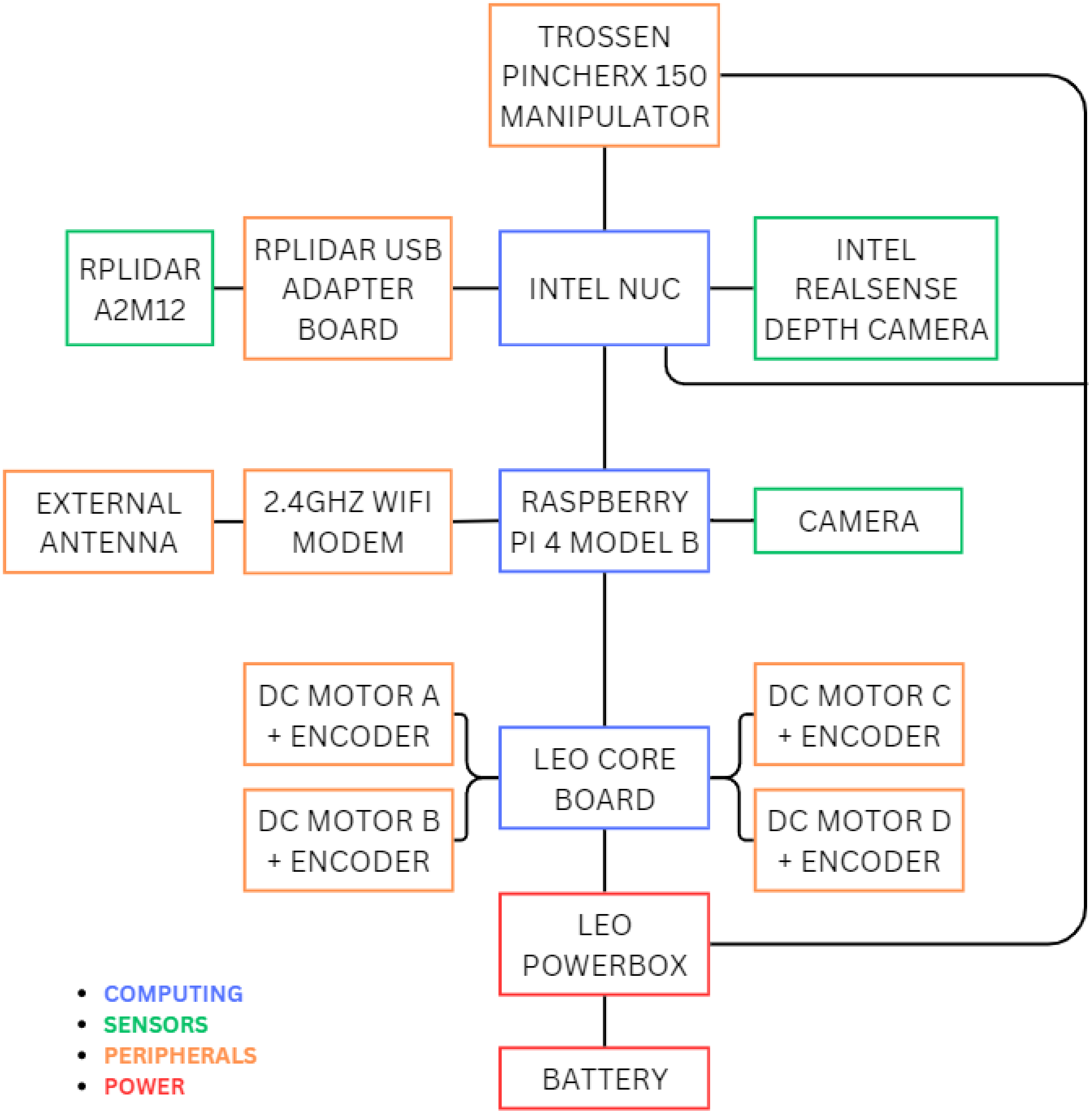


Figure 1: System block diagram showing all the components in the robot

3 Mechanical Design

After analysing both, the functional and product requirements that arise for our problem structure, for the smooth functioning of the peripherals in harmony, it was essential that the payload sled design resonate with the same principles. The mechanical design structure for the payload sled is divided into three major levels of height and operation:

1. Lower Scanning Level (L-1): Includes a structure connected with ribs to mount the LiDAR below the belly of the rover.
2. Base Computing Level (L0): Consists of the Intel NUC and the RealSense Depth Camera.
3. Upper Manipulation Level (L1): Mounts the Robotic Arm onto the rover and has space for storing retrieved objects.

These three levels are connected by unibody ribs which make sure the levels stay in position during operation. The structures are designed to have enough clearances for the operation of these peripherals as well as allows ample motion to the rover suspension for ease in manoeuvrability while mapping and retrieval.

3.1 3D CAD Design Model

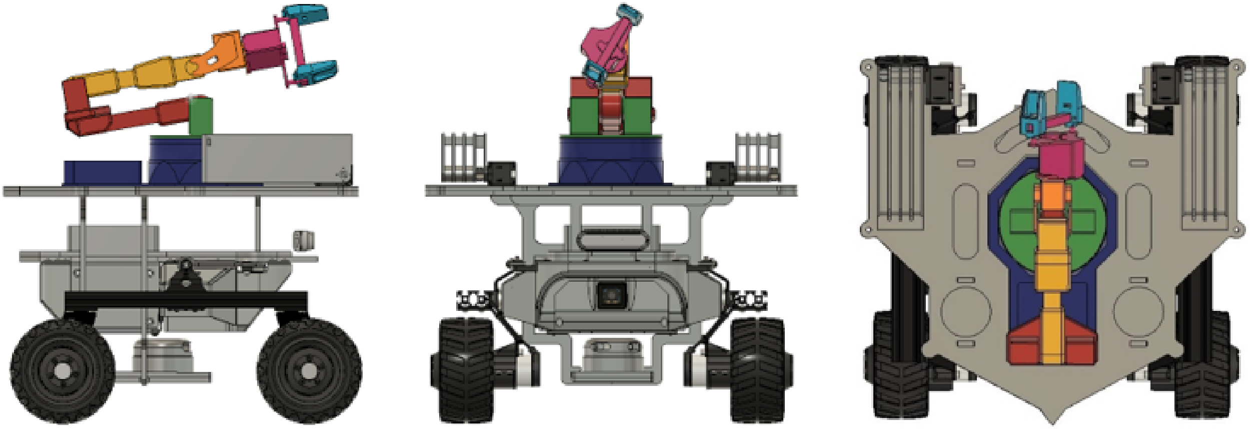


Figure 2: 3D CAD model design

3.2 Design Considerations

After analysing both, the functional and product requirements that arise for our problem structure, for the smooth functioning of the peripherals in harmony, it was essential that the payload sled design resonate with the same principles. The problem was divided into three tasks of Localisation & navigation, Object Detection and finally manipulation and storage of detected objects.

3.3 Sensor Obstructions:

A critical aspect of our design strategy involved addressing potential sensor obstructions to ensure unhindered operation and accurate data acquisition. To mitigate this challenge, a sophisticated three-level payload sled system was proposed, strategically positioning each layer to fulfil specific objectives while minimizing interference from other components.

Level -1: The LiDAR, operating at the lowest level has a better chance of detecting obstacles and map the layout of the environment by eliminating other obstructions in its working plane. Placing the LiDAR lower also makes sure that it just detects obstacles and not the other structures used for mounting the other components.

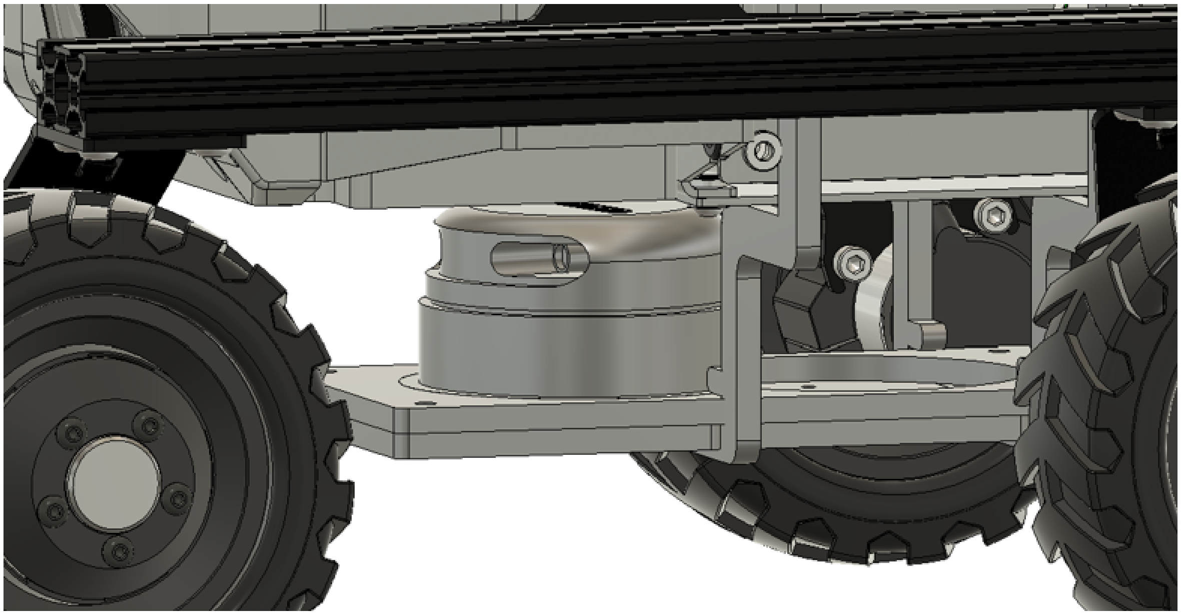


Figure 3: Level -1 of the platform structure

Level 0: The Base Level, connecting both the upper and lower levels firmly to the rover as well as mounting the NUC and the Depth Camera for the detection of the target wooden objects. The Decision to mount the camera on the base level was to reduce wiring complexities and give the camera a clear unobstructed view of the path travelled by the rover. The camera is adjusted to capture the entire field of view of the camera for the detection. This was ensured by mounting the Camera in the frontmost part of the Base Level.

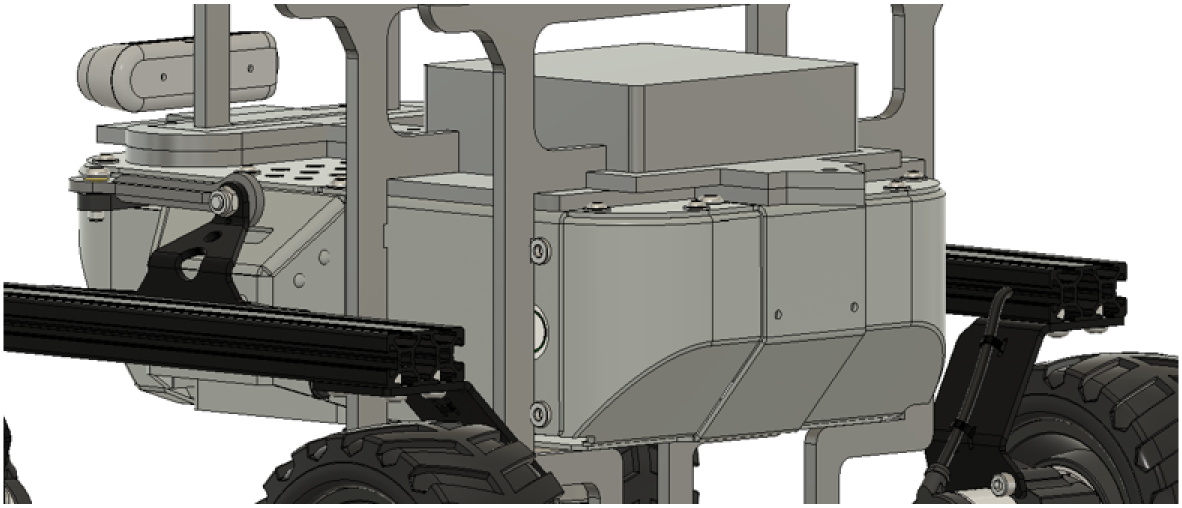


Figure 4: Level 0 of the platform structure

Level 1: The Robotic Manipulator is mounted on this Level along with the storage system for the objects. The decision to mount the manipulator on a different level was to give it freedom to move around and The design decision taken of mounting the Depth Camera on the Lower Level gave rise to a new issue. The Robotic Arm can drop the objects into the storage space but as the rover is required to empty its contents as well, it cannot detect the positions of the objects and empty them as it would require object detection capabilities at the end effector. Mounting the camera at the end effector would be problematic as the vibrations caused due to motion may affect the end results of detection. Furthermore, it causes wire routing issues and as the camera is a delicate sensor it did not seem like a good decision to mount it on the arm. The problem of the storage was approached from a different perspective.

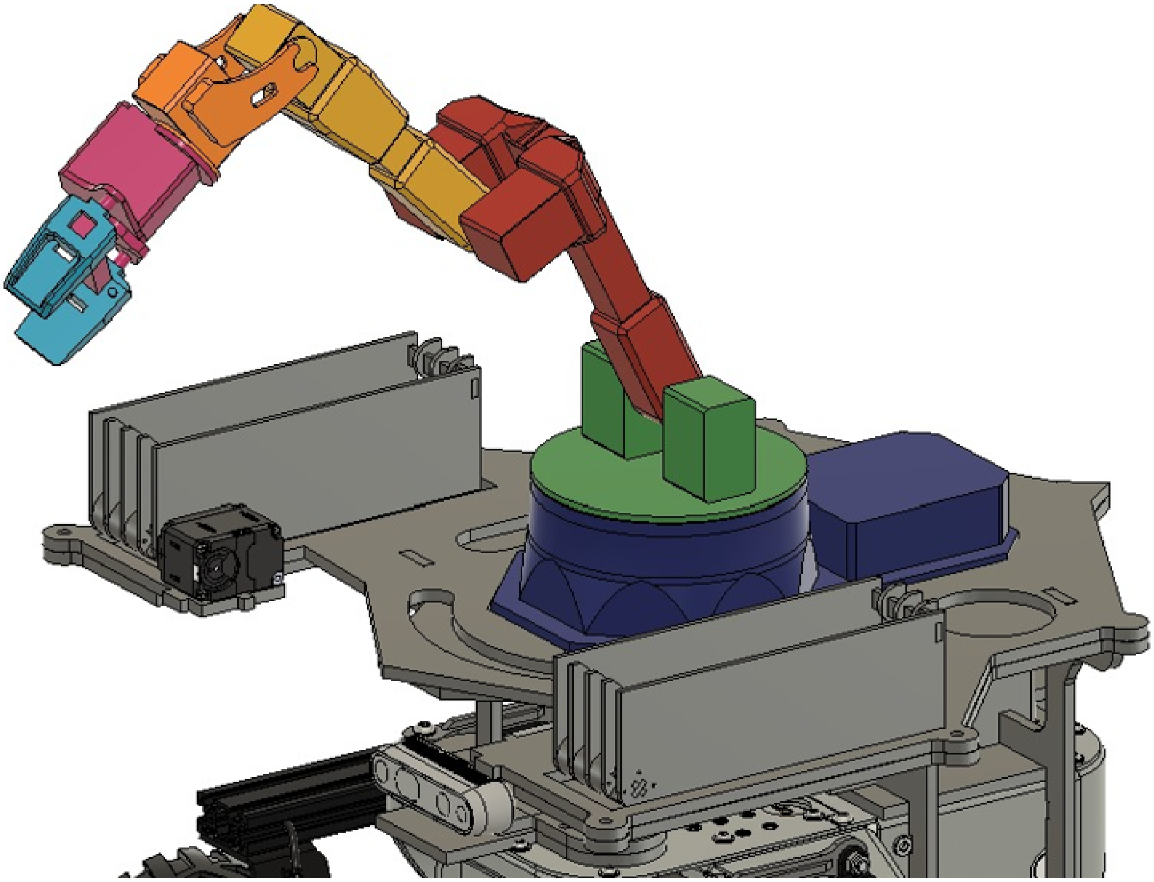


Figure 5: Level 1 of the platform structure

3.4 Storage Container Design:

A light weight stacked payload container was designed which can be altered based on the size of the target object needed. This system enables the team to deploy the rover to pick up and even potentially segregate two different types of target objects into two separate storages. Since the decision was taken to mount the camera on the rover's Base Level, an active storage unit was made that will tilt the stacked storage unit about a pivoting point and dump the objects in front of the fore wheels of the rover. The Motors will be integrated to pivot the inclined storage container with objects that will be dumped in front of the fore wheels of the Rover.

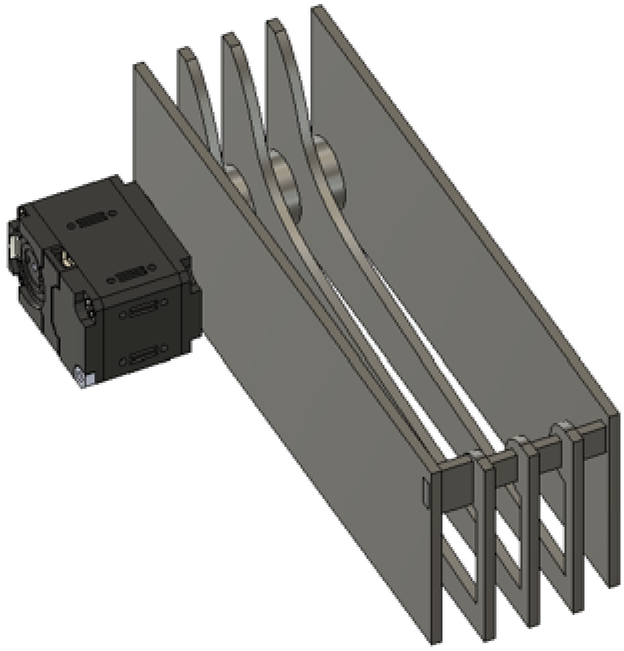


Figure 6: Storage Container Mechanism

3.5 Design Strategy for mounting and assembly

Modular Design for Varied Mounting Options:

Implementing modular structures for peripherals is a strategic and advantageous decision for an autonomous rover tasked with the retrieval of an object.

The dynamic nature of retrieval missions demands adaptability, and modular components provide the flexibility needed to accommodate various scenarios.

With interchangeable modules for sensors, manipulators and computing devices, the rover can swiftly customize its configuration based on the specific requirements of the retrieval task at hand. This modular approach not only enhances the rover's versatility but also simplifies maintenance and upgrades, as individual components can be easily replaced or improved without the need for extensive overhauls.

Furthermore, it facilitates a cost-effective and scalable solution, allowing the autonomous rover to evolve and integrate new technologies seamlessly. By embracing a modular structure, the rover is well-positioned to navigate diverse environments and efficiently retrieve objects with precision and reliability.

3.6 Design Files

The design files show in Figure 7 are the overall dimensions and clearances of the frame for the rover.

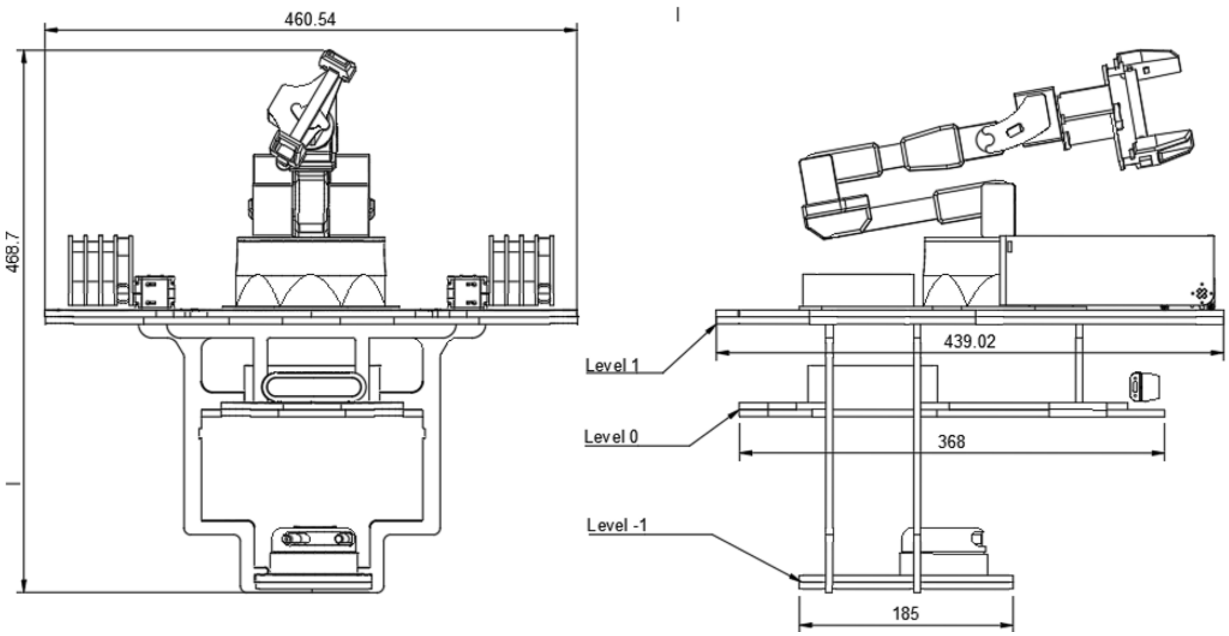


Figure 7: Frame drawing file

3.7 Structural Strength:

To analyse the critical component in the assembly, the component most likely to experience multiple constraints and loads is the Support Ribs added to connect all the layers. Since the support ribs undergo maximum loading out of any other component, the design dimensions were checked against the max load capacity of the ribs. There are two possible types of failures that can occur, Buckling and Shearing and can be seen in figure 8.

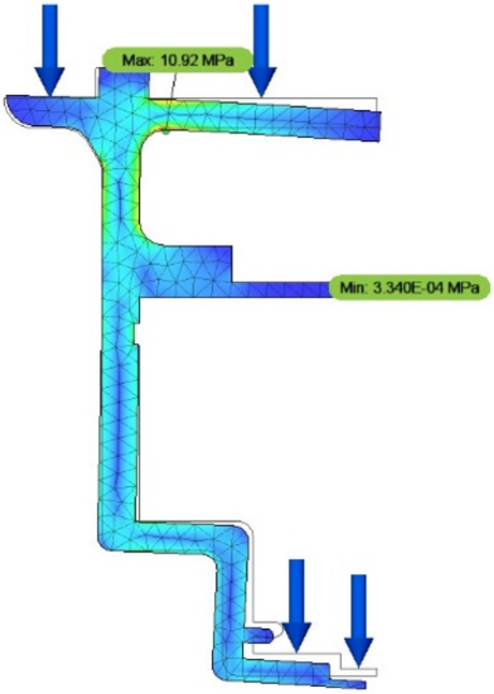


Figure 8: Stress type: Von Mises, Material Compressive Yield Strength (Acrylic): 100-125 MPa

To make sure that the shearing is prevented, we have simulated the structural analysis in static conditions and found that the result for each rib is more than satisfactory even if applied with all the loads. For the loading conditions we considered that the Leo Rover can only take a maximum payload of 5 kg. So that meant our entire design structure and the total weight of the objects collected should always be under 5 kg.

Although, mounting the entire assembly on one rib makes it prone to flexural as well as vibrational forces. To avoid that, mounting of the top layer has been divided into six different stress distributing points that will avoid deformation of any kind in all directions.

3.8 Manufacturability:

Manufacturing has been one of the primary design objectives since operating under deadlines can cause unexpected delays. The entire design of the payload sled has been done in a way that minimizes time consuming methods of manufacturing like Milling or 3D printing. Compared to traditional cutting methods, laser cutting minimizes material waste and reduces production time, as it requires minimal setup and tooling changes. Laser cutting significantly enhances manufacturability by offering precision, versatility, and efficiency in material processing. This enables manufacturers to fabricate intricate designs with high accuracy, allowing them to produce complex components with tight tolerances. Additionally, the non-contact nature of laser cutting eliminates the risk of tool wear, ensuring consistent quality over long production runs.

4 Electrical Design

4.1 Electrical Block-Diagram

The power connection diagram outlines how electrical power is distributed among the robot's peripherals. In addition, the diagram shows connection types and interfaces used between the robot's peripherals, and is shown in Figure 9.

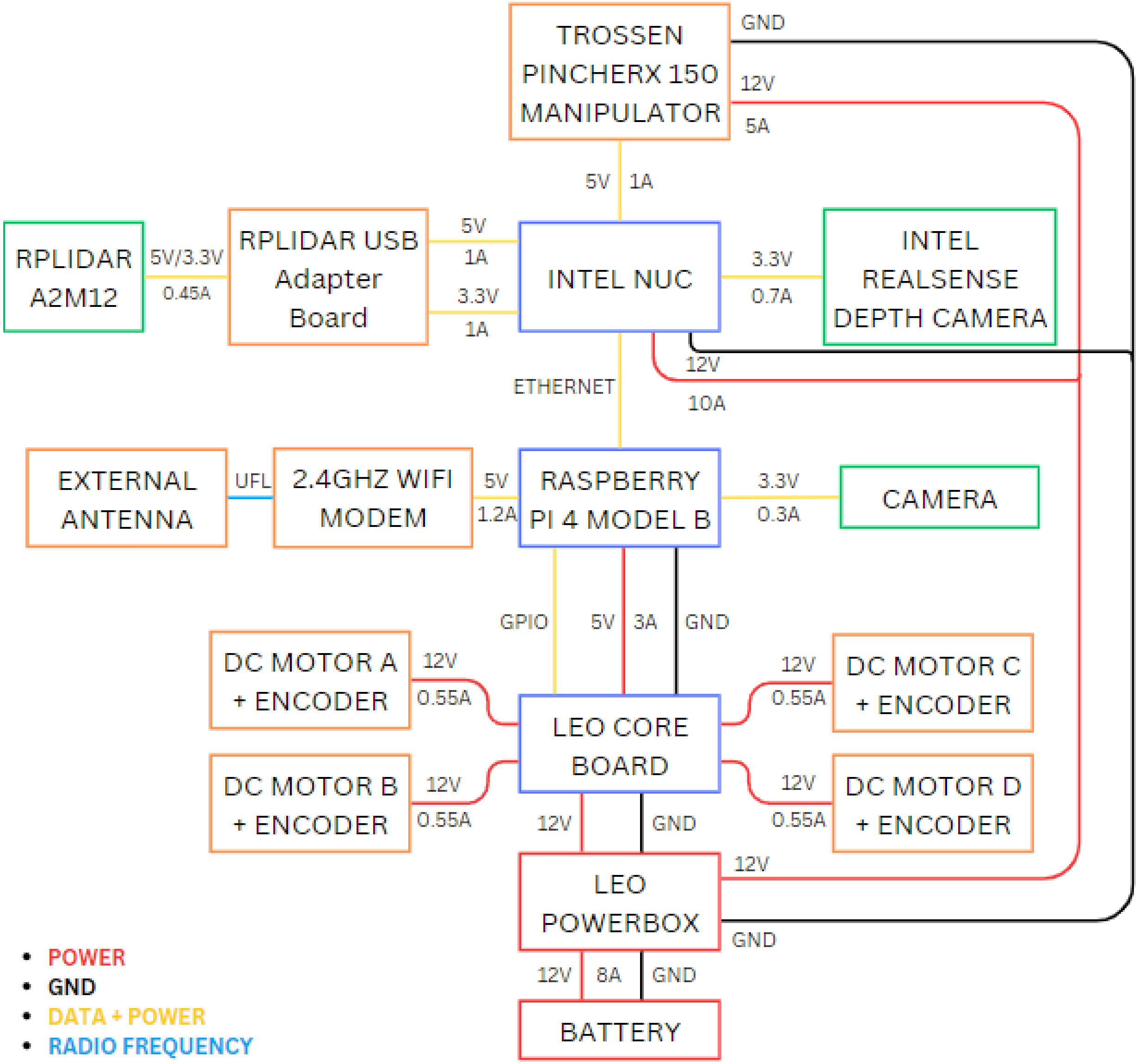


Figure 9: Power connection diagram showing all peripherals on the robot

The Intel NUC operates with a voltage requirement of 12V and a current requirement of 10A(120W). The Leo Rover Battery provides 12V and 8A. In this case, the Leo Rover Battery is not suitable for powering the Intel NUC and an external power supply might be required.

5 Software Design

5.1 RQT Graph

The following section delves into the visualisation of the nodes and topics used, along with their respective links to the System Block diagram. The following graphs were created using RQT_Graph, a ROS tool that provides a dynamic and real-time graphical representation of the relationships and connections between these nodes.

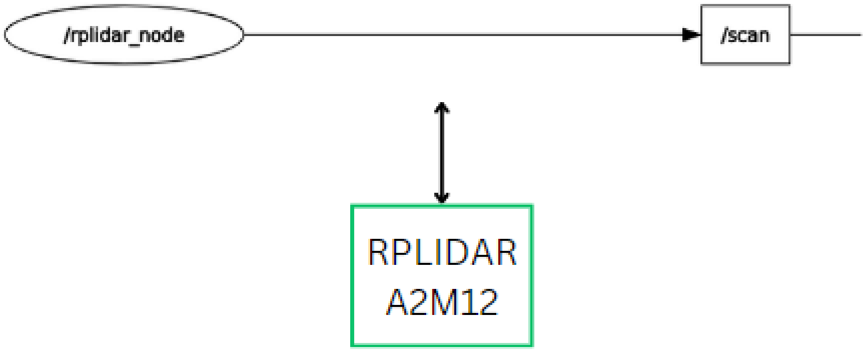


Figure 10: RQT Graph LiDAR section

The RPLIDAR A2M12 utilises the RPLIDAR ROS Package for reading the scanned data, with the ‘rplidar_a2m12_launch.py’ launch file. The launch file then launches the ‘rplidar_node’ node that publishes the data through the ’/scan’ topic in the form of ’sensor_msgs/LaserScan’ messages. Since the LiDAR is placed underneath the robot the minimum detection distance was changed to 0.3 meters(range_min value from rplidar_node.cpp file). This prevents the LiDAR from detecting the robot’s wheels, ensuring the avoidance of inaccurate data.

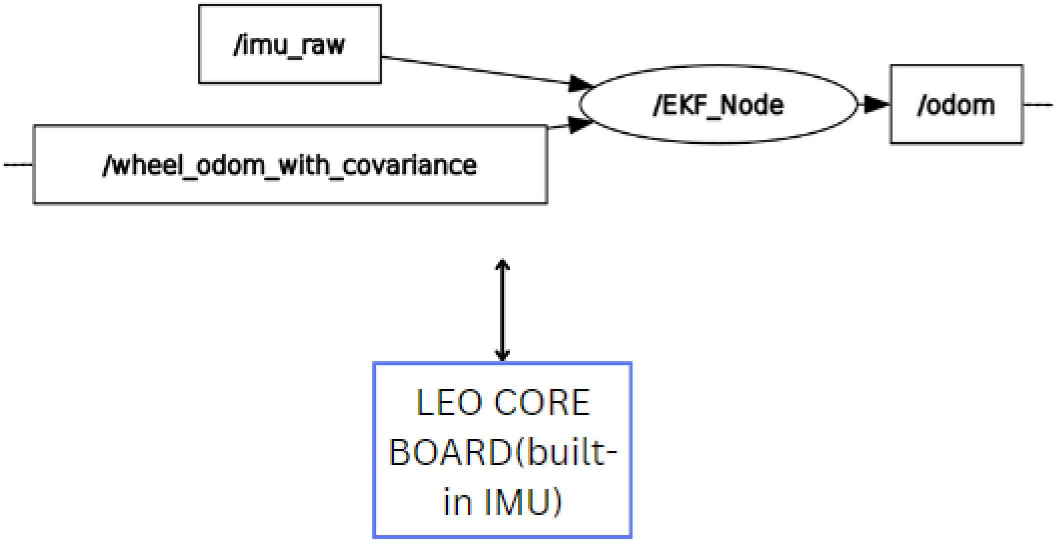


Figure 11: RQT Graph IMU section

The ’EKF_Node’ node is designed to read the Leo core board’s built-in IMU data which include angular velocity and linear acceleration(‘/imu_raw’ topic) and the covariance of wheel odomentry data(‘wheel_odom_ with_covariance' topic). It then implements an extended kalman filter and publishes the new odometry('/odom' topic) in the form of 'nav_msgs/Odometry' message, which is used for the Navigation.

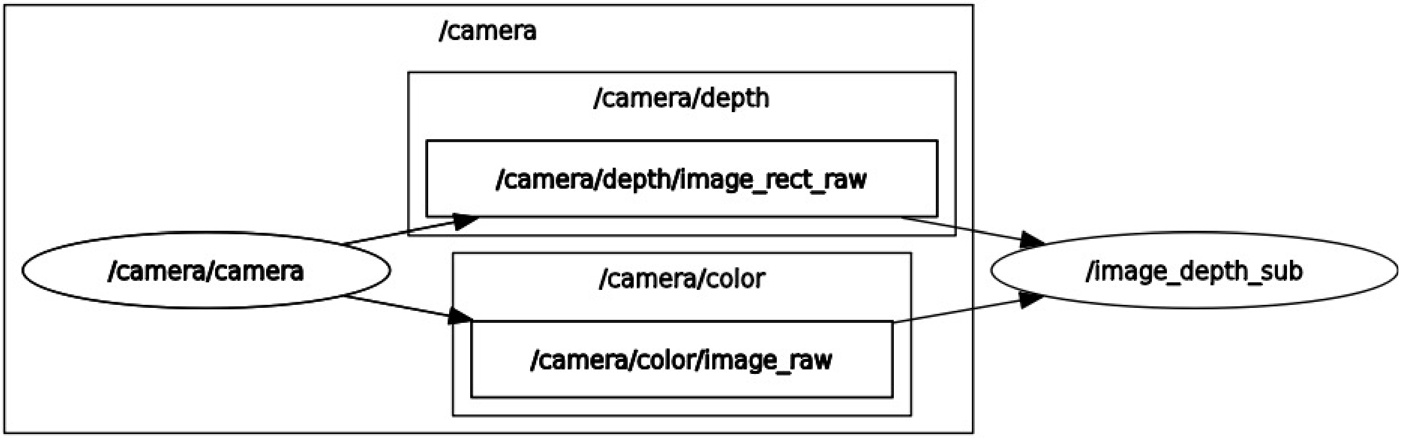


Figure 12: RQT Graph depth camera section

The "object_detect" node is designed to interface with a Realsense Camera D435. It subscribes to two topics published by the camera: "/camera/camera/colour/image_raw" and "/camera/camera/depth/image_rect_raw," both of which publish messages of the type Image. These messages contain the colour and depth information captured by the camera, respectively. The "object_detect" node processes this information to determine the position and orientation of objects within the camera's field of view. The processed data is then published to the "object_pose_to_camera" topic, which emits messages of the type 'geometry_msgs/Point'. This information is intended for use by robotic arms or other devices that require spatial information about objects in their environment.

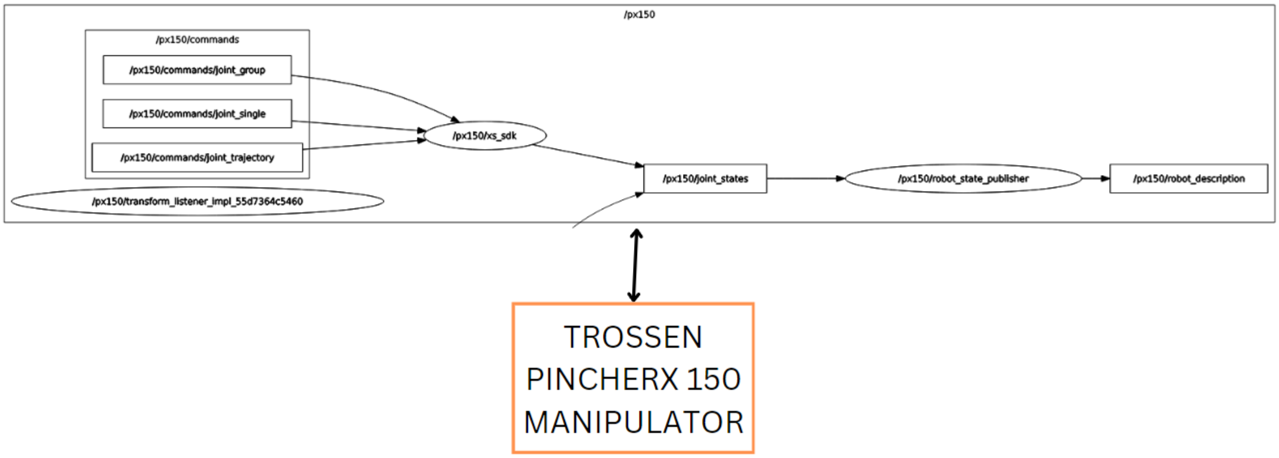

/px150
/px150/commands
/px150/commands/joint_group
/px150/commands/joint_single
/px150/commands/joint_trajectory
/px150/xs_sdk
/px150/joint_states
/px150/robot_state_publisher
/px150/robot_description
/px150/transform_listener_impl_55d7364c5460
TROSSEN PINCHERX 150 MANIPULATOR

Figure 13: RQT Graph Trossen PincherX 150 section

The 'interbotix_ros_manipulators' package was utilised to integrate the Trossen PincherX 150 Manipulator into the robotic system. This package provides a comprehensive set of interfaces and functionalities to control and command the manipulator. The primary focus will be on utilising the '/px150/joint_states' topic, which holds '/sensor_msgs/JointState' messages. The '/sensor_msgs/JointState' messages include data such as name, position, velocity, and effort. The robot system will utilise an 'Arm_Control' node that will provide the Trossen PincherX 150 with a joint state that will be calculated from the '/sensor_msgs/geomtry_msgs' messages; provided from the "object_pose_to_camera" topic. To calculate the end effector position to successfully grasp the object and retrieve it, placing it inside the retrieval mechanism.

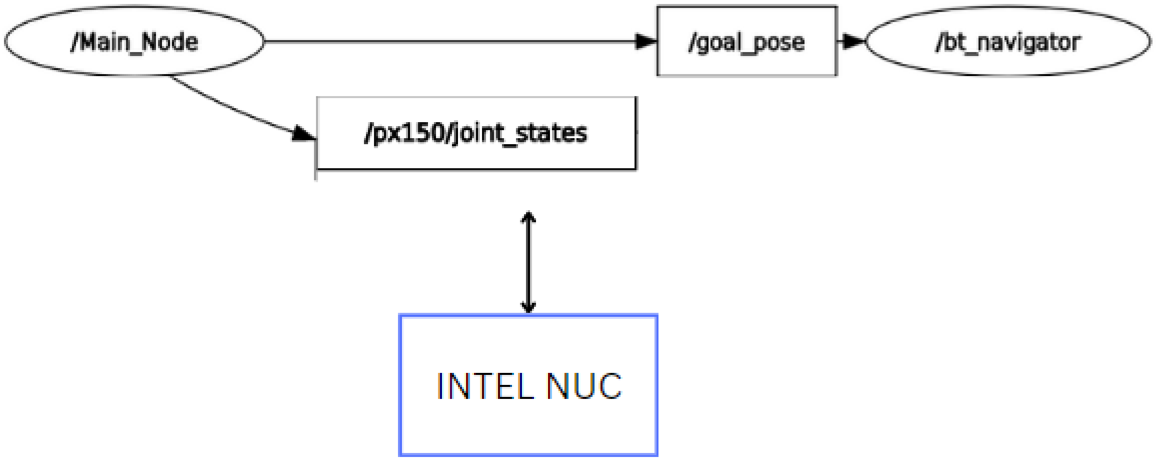


Figure 14: RQT Main Node section

The 'Main_Node' node is designed to process various sensor and navigation message types, including 'OccupancyGrid', 'Odometry', 'Path', 'LaserScan','Image','PointCloud2' and 'IMU' messages.

The node receives messages from topics such as '/map', '/local_costmap', '/global_costmap' and '/odom'. These data, in conjunction with the SLAM-Toolbox and Nav2 Packages, will be utilised for performing localisation and mapping of the environment.

Moreover, the node features two publishers for sending messages. The first publisher controls the robot's movement by publishing '/geometry_msgs/msg/PoseStamped' messages to the '/goal_pose' topic. The second publisher manages the robot arm's movement by publishing '/sensor_msgs/JointState' messages to the '/px150/joint_states' topic.

5.1.1 Localisation and Mapping

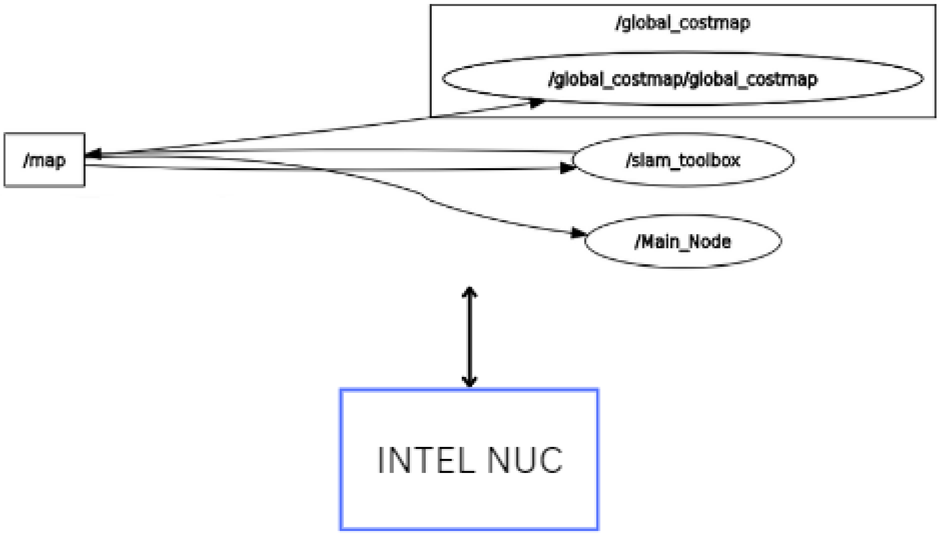


Figure 15: RQT SLAM Algorithm

The slam_toolbox package is used for Simultaneous Localization and Mapping (SLAM). It works in conjunction with the lidar, wheel encoders, IMU and transformation data to enable accurate mapping and localisation capabilities. The slam_toolbox subscribes to inputs such as lidar data ('/scan' topic), wheel encoder and IMU data ('/odom' topic), and transformations between the robot's base frame and sensor frames ('/tf' topic). The fused data is used to estimate the robot's pose (position and orientation) in real-time, providing accurate localisation information. The lidar data captures static features in the environment and thus slam_toolbox continually updates the map as the robot moves through the environment by publishing a 2D occupancy grid map('/map' topic - 'nav_msgs/msg/OccupancyGrid' message).

5.1.2 Navigation

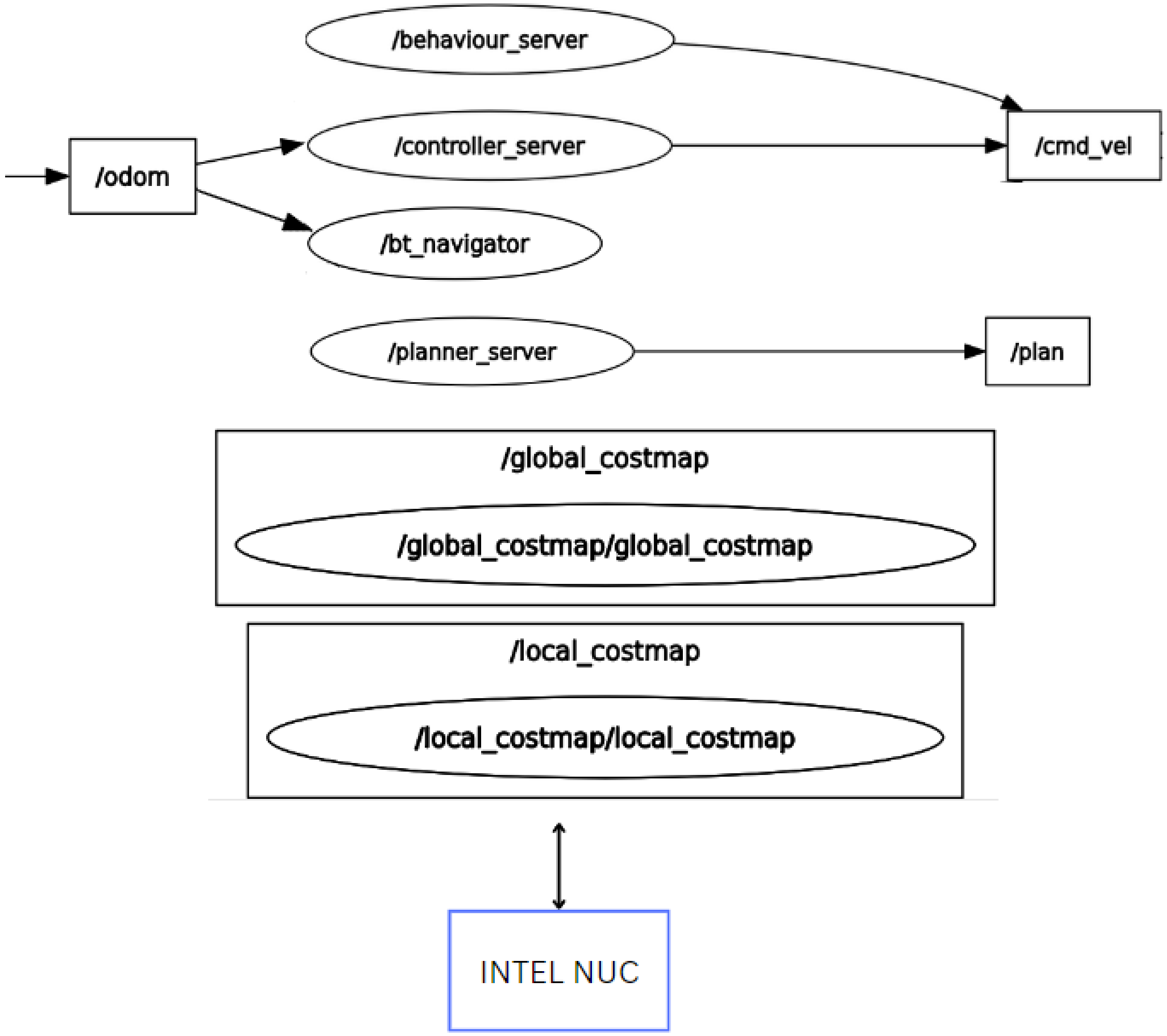


Figure 16: RQT Navigation Algorithm

The NAV2 package is a fundamental element for robot's navigation. The navigation relies on the SLAM_Toolbox which generates an occupancy grid map and localise the robot in the environment. The goal of the NAV2 package is to ensure that the robot can go from point A to point B. The Nav2 stack relies on multiple inputs such as the transformations about the robot's pose and the map('/tf'), lidar scan data('/scan'), occupancy grid map('/map'), odometry data('/odom') and goal pose received from the frontier algorithm('/goal_pose').

The inputs are then passed to a planner and a controller, which are key components responsible for path planning and motion control and are handled by a behaviour tree.

Planner: The planner is used to generate the path between a start and a goal position, while avoiding the obstacles detected from SLAM_Toolbox. The planner plugin used is the NavFn planner with the A* algorithm for path planning and allowing unknown spaces.

Controller: The controller is used to calculate the velocities needed to ensure that the robot is following the path based on its velocity,pose and measurements. The controller plugin used is the DWB controller.

Behaviour Tree: The behaviour tree is used to define and manage complex robot behaviours and decision-making processes, like spin or backup behaviours. It provides a hierarchical structure for organising and executing different behaviours based on conditions and priorities.

The NAV2 stack combines the data from input topics with the planner and controller and is able to navigate autonomously while publish its own topics. Some topics are the following: velocity commands to control the robot's motion('/cmd_vel), a planned path('/path), a global plan('/plan'), a local plan('/local_plan'),a cost map, which represents the occupancy cost of the environment('/costmap), a global costmap('/global_costmap'), information about the planner('/planner_server') and information about the controller('/controller_server').

5.1.3 Frontier Exploration

When the robot is deployed in an unknown environment, it must choose a point to navigate towards. The robot must look for the object(s),which can be located anywhere in the map. For that reason the only available information is the robot's starting position, and thus the robot must run an exploration algorithm to navigate to unknown areas. The chosen exploration algorithm is the Wavefront Frontier, which is the integration of Frontier Exploration and Wavefront Planning. The algorithm is guided by a behaviour tree that ensures that the robot can have one state at a time and achieving the transition between each state. The robot operates within five distinct states: FIND_ROBOT, FIND_FRONTIER, GO_TO_FRONTIER, GO_TO_OBJECT and RETURN_HOME.

Initially, the robot is in the FIND_ROBOT state, which finds the position of the robot in the map frame using the transformation between the 'map' frame and the 'base_link' frame. It then converts the robot's position to occupancy grid coordinates. Once this step is done, the robot's state changes to FIND_FRONTIER state. In this state the Wavefront Planning part is used to search the surrounding area. The algorithm identifies three types of cells: open, unknown and occupied.

In cases where the algorithm detects unknown cells(usually areas behind walls), it stops the search and the Frontier Exploration part determines if the unknown area should be explored. A frontier is determined based on the number of the unknown cells and the existence of occupied cells by checking small areas over the unknown cells. Once the frontier is set, the state changes to GO_TO_FRONTIER, which sets the frontier point as the goal pose and publishes so that the robot starts navigating towards it.

While the robot is driving toward the new point it checks if the camera has detected the object. In case where the object is detected the robot switches to GO_TO_OBJECT state to drive toward the object and pick it up. When the object has been collected the state changes back to GO_TO_FRONTIER to go to the frontier point that was set before. This cycle continues until the entire map is explored.

Our approach assumes that a fully explored map indicates no remaining objects and robot can return to its starting position with the RETURN_HOME state.

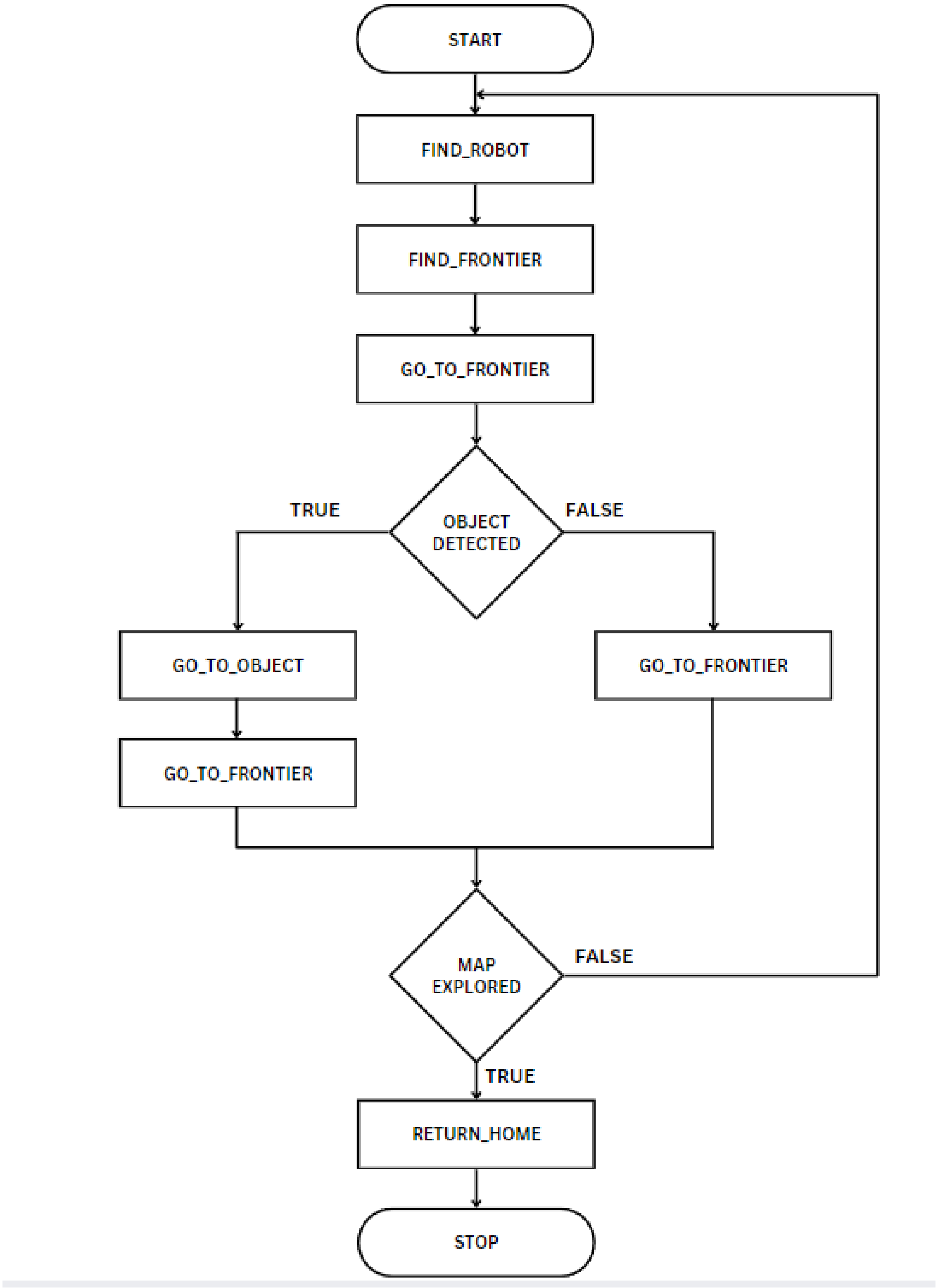


Figure 17: Flowchart of Frontier Exploration Behaviour (State) Tree

5.2 GitHub Repository

The team 5's GitHub repository presents the results of the robot design project that the team completed. Portraying skills in programming, hardware tools and design methodology have been combined to allow the creation of a retrieval robot. The aforementioned repository covers coding and designs used to create the technically advanced robot, alongside a narrative that outlines the development process. This includes documentation of the teams challenges, solutions and subsequent progression. Further outlining project goals, with justification of design, engineering decisions, and testing procedures. The document is not only intended to elucidate an overview of the project, but to serve as a resource for future robotic engineers.

~ **https://github.com/ziyicheng427/RobotDesign_Group5**

6 Analysis

The components provided by the university of Manchester, can be utilised to present a unique solution to the problem statement, which team 5 aims to achieve in their robot. To analyse the effectiveness of the team's robotic solution, requires a closer inspection of the team's design decisions. During the planning and fabrication stages of the robots creation, the verification matrix from section 7.1.5 was examined and heavily referenced throughout. By going through and analysing the verification matrix each requirement has been rigorously considered in the design choices and can be clearly explained:

6.1 Functional Environment Requirements

The functional environment requirements are very basic in that the robot will be operating indoors on a concrete level flooring. The flooring will be dry and have no steps or pitfalls. The requirement for this environment (P-17) is that 'the robot shall maintain full functionality on a flat even concrete surface'. The base of the robot being a Leo Rover, is able to adapt to indoor and outdoor environments. The design of the rover enables for four wheel drive up large inclinations and rough uneven terrain with its large 130 mm diameter rubber wheels, therefore the robot should be able to fully traverse a level indoor environment. The robots platform base is over qualified for this environment and should be fully functional on the required flooring.

6.2 Functional Object Requirements

The functional object requirements refer to the object that needs to be retrieved from the environment. The fundamental requirement for this section is that robotic system 'should be able to detect this desired object in the environment' (P-1). The object that needs to be retrieved is an opaque wooden cube, that is small and light enough to be handled by the onboard Trossen PincherX 150 Manipulator. The object will be in the environment and within the arms 450 mm reach, weighing less than 50 grams. The object will be detected by using a Intel Realsense Depth Camera that will be mounted on the same shelf as the Intel NUC shown in Figure 2. that will be able to use an object detection algorithm to differentiate the desired object and locate it in the robots virtual simultaneous localisation and mapping (SLAM).

6.3 Functional Obstacles Requirements

The obstacle requirements of the problem also revolve around the detection of obstacles in the environment (P-2). These obstacles will be rigid and static and tall enough so that the Leo Rover cannot drive over them; however the obstacles are never stated to be taller than the Leo Rovers height of 433 mm [5]. These obstacles will be within the operating environment and will take the form of large wooden or plastic structures that impede the robots path planning operation. Opposed to the depth camera which will be trying to find the desired object in the environment, a RPLidar A2M12 scanner will be utilised to find all walls and obstacles that will hinder the path of the robot. This dual sensor fusion shall create two different visions for the robot that will enable the robot to distinguish between the desired object and basic path blocking obstacles (P-3). The LiDAR will be mounted beneath the rover frame as shown in Figure 2 ensuring that the obstacles are also seen in the environment regardless of their height. A software range limit will be placed onto the LiDAR sensor to ensure the rovers wheels are not recorded in the scans, and the path planning algorithm will introduce some wiggling to capture a full 360° field of view, around the rubber wheels.

6.4 Functional Rover Path Planning Requirements

The path planning requirements for the robot is the first major task for the robotic system. The rover shall be installed with a path planning algorithm that incorporates SLAM. With the LiDAR and front visual camera being the main two sources of obstacle detection the algorithm will have enough feedback to create a virtual map and plan a correct path through the environment (P4 & P5). The system will be able to localise itself within this map using its on board IMU (P6).

The algorithm will use behaviour trees, a method of building up complex relationships of robot behaviours, to generate velocity messages to send to the wheel motors and correctly path itself autonomously. By improving the complexity and quality of these behaviour trees will enable the rover to tightly manoeuvrer itself down any path it can fit its frame into or detect if it cannot fit at all, having to recalculate its path (P7 & P-8).

Using this greatly increase the chances that the teams solution will be able to find the object in the environment. An additional, software limit shall be placed onto the motors to ensure the speed of the rover does not exceed $0.5ms^{-1}$, ensuring standard safety requirements are met throughout the project (P-9).

6.5 Functional Object Manipulation and Retrieval Requirements

The object manipulation and retrieval requirements associate with the Trossen PincherX 150's capability to work in tandem with the depth camera sensor. The robots manipulator software will implement a Proportional Integral Derivative (PID) controller to control the manipulator, allowing the robot to grasp the object with minimal oscillations in the arm. The robot will use depth camera to detect how far away the object is from the arm giving feedback to the manipulator to increase accuracy of the object's retrieval (P-10). The manipulator will also have a velocity threshold to ensure safety requirements are met throughout the project (P-12). The end-effector will place the object into retrieval section in Figure 5.

and begin its secondary stage of path planning where the rover will try to find the retrieval site to offload the object from the to the desired location.

6.6 Functional Degree of Autonomy Requirements

The degree of autonomy ranges in robotics from complete human teleoperation to fully autonomous robots, team 5 aims to create a robot that could engage and achieve all requirements autonomously with slight human interaction to start the robot in the environment. The three stages of path planning of finding the object, retrieving the object and moving to the end site; will all be autonomous as the path planning algorithm will be able to detect its environment and plan a path without any human interaction (P13 - P16).

6.7 Product Design Performance Requirements

With the Trossen PincherX 150 the robot will be able to handle and retrieve objects no heavier than 50 grams and no wider than 50 mm per side ensuring the performance requirements to retrieve the desired object is met (P-18). The robot will be equipped with a Intel Realsense Depth Camera that can has a depth accuracy of < 2% at 2 meters [6] and a RPLidar A2M12 that has a range of 12 meters [7]; where both sensors have high accuracy at ranges larger than the required one meter of minimum range. The sensor's locations and resolutions will be optimised to reduce false detections below the 5% maximum (P-19). With the implemented SLAM algorithm the system will scale the obstacles to allow for a rover of width 540 mm and length 520 mm (120% larger than the real rover) to traverse the environment without coming into contact with the environment (P-20).

6.8 Product Design Reliability Requirements

The reliability requirements includes the lifetime the robot can operate and if it can operate at indoor temperatures. With the standard lifetime of a Li-Ion 3S battery being four hours of driving or eight hours of sensor operation the minimal lifetime requirement is met (P-21). The rover, as previously mentioned, is able to operate outdoor at hotter temperatures of $40°C$ which is double the average room temperature of $20 - 22°C$ and will not be hindered by indoor humidity levels (P-22).

6.9 Product Design Size, Shape, Mass, and Style Requirements

The Size, Shape and mass requirements of the Leo rover comply with the problem statement being 447 x 433 x 249 mm and weighing 6.5 kg. The sum of components on the payload sled, including the Trossen PincherX 150 manipulator, RPLidar A2M12 and Intel Realsense Depth Camera is 1.81437 kg (4 lbs) [8], 0.19 kg [9] and 0.072 kg [2] respectively. The total weight is 9.25437 kg which is far below the 15 kilogram limit, giving extra weight to print a large payload sled to fit all the mention peripherals.

The aesthetic design requirement was ensured as all acquired and printed components include ergonomic features and safety colours to suit a modern industrial environment.

6.10 Product Design Cost Requirements

There was no further expenses or purchases during the design and construction of the robot, outside of printing a payload sled shown in, abiding to the cost requirements set in the problem statement.

6.11 Product Design Communication Requirements

The communication requirement is a team set requirement to monitor the robots status and current readings at all times during initial testing and assessments. A constantly flow of data will be sent wirelessly through Wi-Fi, every second to aid the team in debugging and validation of the robots software.

6.12 Success Criteria

Finally the analysis can be fused together to formulate a success criteria that incorporates all of the Functional and Design Requirements for the robot, to validate if the problem statement is achieved.

1. To assess that the functional environment requirements have been met, the robot should be able to consistently drive straight in all 4 cardinal directions on a level and dry concrete floor without human intervention. The robot should be able to rotate a complete 360° with a precision of ±1° on the surface.

2. To assess that the functional object requirements have been met, the robots onboard sensors should be able to detect a 50 mm coloured, opaque wooden cube with an accuracy of 99.9% with its object detection algorithm.

3. To assess that the functional obstacles requirements have been met, the robots onboard sensors should be able to detect path blocking obstacles that are within the operating environment with a false detection accuracy of less than 1% with its obstacle detection algorithm.

4. To assess that the functional rover path planning requirements have been met the robot should create virtual map and the robot should be able to localise itself in said map always. The robot should be able to plan a path based on this map and should not make contact with the detected obstacles that will block the robots movement, to be considered correct.

5. To assess that the functional object Manipulation and retrieval requirements have been met, the robot should be able to pick up the detected object in the environment and retrieve it with a success rate greater than 95%.

6. To assess that the functional degree of autonomy requirements have been met, the robot should be able to complete all previous success criteria without human intervention.

7. To assess that the product design performance requirements have been met, the robot will be able to handle and retrieve objects no heavier than 50 grams and no wider than 50 mm per side, the robot should have a depth camera that has a depth accuracy of $< 2\%$ at 2 meters to detect objects and a LiDAR that has a range of 12 meters to detect obstacles.

8. To assess that the product design reliability requirements have been met the robot should be able to operate at full functionality at room temperature for at least 4 hours.

9. To assess that the product design, size, shape mass and style requirements have been met the robot should be less than 500 mm x 500 mm x 500 mm and weigh less than 15 kg at the end of the project.

10. To assess that the product design cost requirements have been met the team should not have made any additional expenses or purchases.

11. To assess that the product design communications requirements have been met the robot should relay its status wirelessly every second.

7 Appendix

7.1 Design Requirements Analysis

7.1.1 Stake Holder Engagement

Introduction

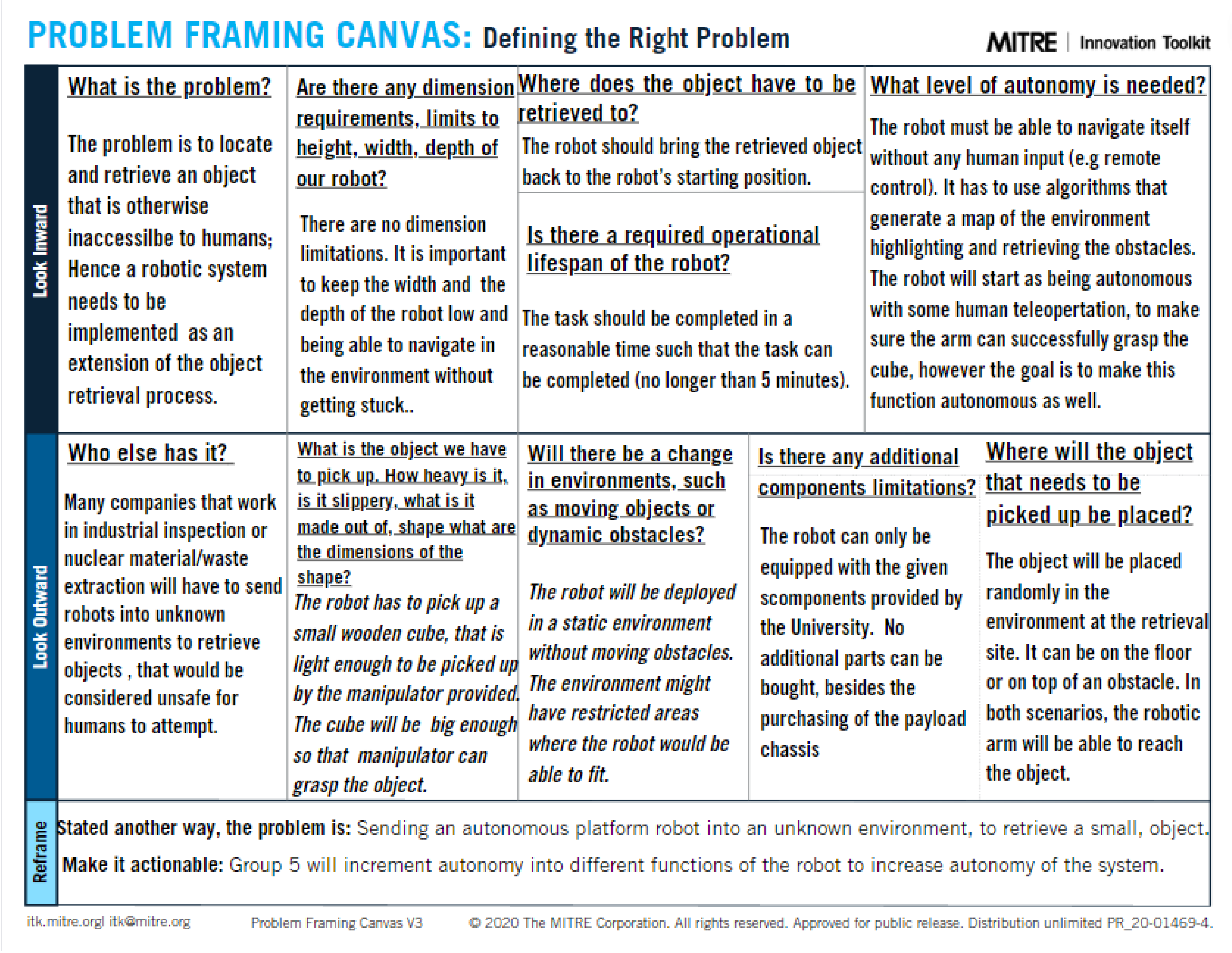
In this project Group 5's aim to is acknowledge and provide a solution to the customers request. The customer requirements statement is as follows:

> **"Develop a robot which can autonomously retrieve an object from an environment."**

As this initial statement was vague, more information was required to be extracted from the customer. The team had two sessions to discuss and validate the design requirements for the product. Each of these sessions was a fifteen minute question and answer formatted meeting, where the team brought questions to outline and recognise the scope of the project. The recording of these questions are summarised in Table 1.

Problem Framing Canvas

Table 1: Problem Framing Canvas fabricated after clarification from the customer.



PROBLEM FRAMING CANVAS: Defining the Right Problem

MITRE | Innovation Toolkit

Look Inward	What is the problem? The problem is to locate and retrieve an object that is otherwise inaccessilbe to humans; Hence a robotic system needs to be implemented as an extension of the object retrieval process.	Are there any dimension requirements, limits to height, width, depth of our robot? There are no dimension limitations. It is important to keep the width and the depth of the robot low and being able to navigate in the environment without getting stuck..	Where does the object have to be retrieved to? The robot should bring the retrieved object back to the robot's starting position. Is there a required operational lifespan of the robot? The task should be completed in a reasonable time such that the task can be completed (no longer than 5 minutes).		What level of autonomy is needed? The robot must be able to navigate itself without any human input (e.g remote control). It has to use algorithms that generate a map of the environment highlighting and retrieving the obstacles. The robot will start as being autonomous with some human teleopertation, to make sure the arm can successfully grasp the cube, however the goal is to make this function autonomous as well.
Look Outward	Who else has it? Many companies that work in industrial inspection or nuclear material/waste extraction will have to send robots into unknown environments to retrieve objects , that would be considered unsafe for humans to attempt.	What is the object we have to pick up. How heavy is it, is it slippery, what is it made out of, shape what are the dimensions of the shape? *The robot has to pick up a small wooden cube, that is light enough to be picked up by the manipulator provided. The cube will be big enough so that manipulator can grasp the object.*	Will there be a change in environments, such as moving objects or dynamic obstacles? *The robot will be deployed in a static environment without moving obstacles. The environment might have restricted areas where the robot would be able to fit.*	Is there any additional components limitations? The robot can only be equipped with the given scomponents provided by the University. No additional parts can be bought, besides the purchasing of the payload chassis	Where will the object that needs to be picked up be placed? The object will be placed randomly in the environment at the retrieval site. It can be on the floor or on top of an obstacle. In both scenarios, the robotic arm will be able to reach the object.
Reframe	**Stated another way, the problem is:** Sending an autonomous platform robot into an unknown environment, to retrieve a small, object. **Make it actionable:** Group 5 will increment autonomy into different functions of the robot to increase autonomy of the system.				

itk.mitre.org| itk@mitre.org Problem Framing Canvas V3 © 2020 The MITRE Corporation. All rights reserved. Approved for public release. Distribution unlimited PR_20-01469-4.

7.1.2 Problem Statement

Discussions with the customer has given rise to a need for a distinct technical solution that can be fulfilled using robotics. The Customers have requested a mobile autonomous solution which can go into unmapped environment and retrieve a specific object and deliver it to a retrieval site. There will be multiple obstacles in the environment blocking the path of the robot. All of these objects and obstacles will be stationary. Additionally, the environment will be an indoor space with a level floor.

Objectives

The goal of Team 5 can be broken down into the following statements:

- The team will use a four wheeled rover and implement any necessary peripherals needed to complete this task.
- The team will successfully retrieve the aforementioned object in the minimal possible time.
- The team will implement an initial level of autonomy for this project so the robot can autonomously path plan in the unmapped environment.
- The team will aim to enable the robot to complete the task with full autonomy.
- The team will aim to extract periodic data from the rover for a better understanding of the status of an ongoing process to get live progress tracking from the operator's end.
- The team will retrieve the object successfully with the robotic solution and the robot should return to its start location.

It is also noteworthy, that the team will take extreme strides to make a robot that operates safely and reliably.

7.1.3 Functional Requirements

The functional requirements define what the robotic system must do and what features and functions are included. These requirements are formed to confirm with the customer and validate what they have said has been communicated correctly.They are constructed under the formatting of NASA's System Engineering Handbook [10].

1. Environment

1. The robot shall only operate on a level surface, such that there is no elevation or pitfalls in the flooring.
2. The robot shall only operate on a dry surface.
3. The robot shall only operate on concrete.
4. The robot shall operate indoors.

2. Object

1. The robot shall retrieve an object that is an opaque painted wooden cube.
2. The robot shall retrieve an object that is small and light enough to be handled by the onboard Trossen PincherX 150 Manipulator.

3. The robot shall be able to detect the object inside the environment.

4. The robot shall be able to reach the object from the ground, as the object will be within the manipulators operational reach.

5. The robot may have more than one object to retrieve.

3. Obstacles

1. The robot shall be able to detect the obstacles in the environment.

2. The robot shall not be able to drive over the obstacles.

3. The robot should be able to detect the obstacles that are lower than its frame, as the obstacles are not necessarily taller than the robot.

4. The robot shall be able to distinguish the blocking obstacles from the object that needs to be retrieved.

5. The robot shall not encounter dynamic obstacles.

6. The robot shall not be bale to move the obstacles as they shall be rigid, static and immovable.

4. Rover Path Planning

1. The robot shall be able to traverse the environment to retrieve the object.

2. The robot shall be able to detect obstacles and create a map of the environment.

3. The robot shall be able to localise its frame inside its map for path planning.

4. The robot's wheels shall move and operate at a reasonable speed, for safety.

5. Object Manipulation and Retrieval

1. The robot may be able to carry and store multiple objects, in a space that is accessible to the manipulator.

2. The robot shall be able to plan a path to the object that needs to be retrieved and a path to the retrieval site.

6. Degree of Autonomy

1. The robot shall be able to traverse and map the environment fully autonomously, without human interaction.

2. The robot shall be to detect obstacles and the object that needs to be retrieved fully autonomously, without human interaction.

3. The robot may be able to carry and retrieve the object to the retrieval site fully autonomously, without human interaction.

4. The robot should communicate its status to the operators, periodically.

7.1.4 Product Design Requirements

The product design requirements are from the group of customers or organisations that need to have their requirements met regardless of their need for the product. These requirements are essential and must be satisfied as failure to do so could result in the banning or recall of the product [11]. The general requirements are as follows:

1. **Performance Requirements**

   - **Surface Navigation:** The robot shall maintain full functionality on a level concrete surface to comply with the environment requirements section (7.1.3.1).

   - **Robot Speed:** A maximum speed 0.5 meters per second is established, for the safety requirement 7.1.4.4.

   - **Object Retrieval Capability:** The robot shall handle an object with dimensions not exceeding 50 mm per side and weigh no more than 50 grams enabling Trossen PincherX 150 manipulator to grasp the object. Retrieval success rate shall be at least 95% under predefined conditions, including variable lighting and slight object position variations and will be validated with testing. This addresses the object manipulation and retrieval section (7.1.3.5).

   - **Obstacle Detection:** The robot shall detect obstacles within a minimum range of 1 meter from its frame with a degree of accuracy, such that false detection are less than 5%. This addresses the obstacles section (7.1.3.3).

   - **Path Planning:** The robot shall have the ability to generate and update a virtual map in real time with a resolution that allows navigation through spaces as narrow as 120% of the robot's frame width. This addresses the rover path planning section (7.1.3.4).

2. **Reliability Requirements**

   - **Operation Consistency:** The robot shall be powered for at least 4 hours of operation at full charge. This the standard lifetime of a Li-Ion 3S battery.

   - **Environmental Adaptability:** The robot shall operate reliably within a temperature range of 0-40 °C. It should also tolerate humidity levels from 20% to 80% for inside building standards. Complying with the indoor environment stated in environment section (7.1.3.1).

   - **Battery Life Span:** The robot's 12 volt battery will have a typical life span of 3-5 years, under proper procedures for charging and discharging.

3. **Size, Shape, Mass, and Style Requirements**

   - **Physical Characteristics:** The robot shall not exceed dimensions of 500 mm x 500 mm x 500 mm and a maximum weight of 10 kg to ensure navigability and ease of transport.

   - **Aesthetic Design:** Functionally aesthetic design to fit a modern industrial environment, with a colour scheme that includes safety colours (e.g., yellow and black), and ergonomic features for human interaction where necessary

4. **Cost Requirements**

   - **Budget Constraints:** No additional components may be funded besides the manufacturing for the payload sled as well as all of the materials in the toolbox and the components from the predefined list.

5. **Communication Requirements**

- **Data Transmission:** The robot shall be equipped the robot with Wi-Fi and Bluetooth for transmitting status and environmental data to a central system every second.

7.1.5 Requirements Verification Matrix

After developing the requirements it is mandatory to identify how these requirements will be verified to the team and to the customer. A verification matrix is introduced as a method of demonstrating how the requirements will be validated [12]. The verification of each requirement is decided by either doing a series of ten tests (Test) or a single demonstration (Demonstrate). The requirement will be verified if the success rate is above 80%, after the ten tests. Some of the requirements will have to be undergo a pretest acceptance, which is verifying the requirement again, before a major demonstration or assessment. The verification matrix can be seen in Table 2.

Table 2: A verification matrix based on requirements proposed in sections 3 & 4.

Requirement No.[A]	**Section**[B]	**Requirement**[C]	**Verification Success Criteria**[D]	**Verification Method**[E]	**Pretest Acceptance**[F]
P-1	7.1.3.2	The robot shall be able to detect the object inside the environment.	1. See if the object can be seen by the camera. 2. Does it confirm the object is the target to be retrieved?	Test	False
P-2	7.1.3.3	The robot shall be able to detect the obstacles in the environment.	1. See if the obstacles can be seen by the robot.	Test	False
P-3	7.1.3.3	The robot shall be able to distinguish the blocking obstacles from the object that needs to be retrieved.	1. Does the robot distinguish the target object from the surrounding obstacles?	Test	True
P4	7.1.3.4	The robot shall be able to traverse the environment to retrieve the wooden cube	1. Can the robot go through the environment and retrieve the wooden cube to the retrieval site?	Test	True
P-5	7.1.3.4	The robot shall be able to detect obstacles and create a virtual map of the environment.	1. Does the robot create a virtual map with its sensors? 2. Does the robot save its knowledge of the map to be reused later?	Test	True
P-6	7.1.3.4	The robot shall be able to localise its frame inside its virtual map for path planning.	1. Does the robot show its frame in the virtual map? 2. Is it in the correct place in the global frame within a radius 20 mm?	Test	True

Requirement No.[A]	**Section**[B]	**Requirement**[C]	**Verification Success Criteria**[D]	**Verification Method**[E]	**Pretest Acceptance**[F]
P-7	7.1.3.4	The robot shall be able to detect if its frame can fit into a pathway, and manoeuvre itself through tight spaces.	1. Does the robot recognise when it cannot go through a path? 2. Does the robot know when its frame cannot fit through tight spaces, if they arise later down that path?	Test	False
P-8	7.1.3.4	The robot shall be able to manoeuvre itself through tight turns.	1. Does the robot path plan correctly for a tight turn? 2. Can it perform this movement.	Test	False
P-9	7.1.3.4	The robot's wheels shall move and operate at a reasonable speed, for safety.	1. Is the robot moving below the established maximum speed? (0.5 ms^{-1})	Demonstrate	True
P-10	7.1.3.5	The robot shall be able to steadily and precisely manoeuvre its manipulator for accurate retrieval of the object.	1 Does the robot manipulator operate without oscillations, that decrease the accuracy of picking up the object. 2. Does the robot manipulator consistently move to the correct location in the global frame, within a radius 2 mm?	Test	False
P-11	7.1.3.5	The robot shall be able to plan a path to the object that needs to be retrieved and a path to the retrieval site.	1. Does the robot create a path to the object and the retrieval site? 2. Is the path correct? 3. Does it follow this path	Test	False
P-12	7.1.3.5	The robot shall operate its manipulator at a reasonable speed, for safety.	1. Does the robot manipulator operate below the safe maximum speed of the arm? (0.5 ms^{-1})	Demonstrate	True

Requirement No.[A]	**Section**[B]	**Requirement**[C]	**Verification Success Criteria**[D]	**Verification Method**[E]	**Pretest Acceptance**[F]
P-13	7.1.3.6	The robot shall be able to autonomously traverse environment and map the environment.	1. Does the robot autonomously traverse its environment? 2. Does the robot autonomously form and remember its virtual map	Test	False
P-14	7.1.3.6	The robot shall be able to localise its frame inside its virtual map for path planning.	1. Does the robot autonomously detect obstacles.? 2. Does the robot autonomously detect the object that needs to be retrieved?	Test	False
P-15	7.1.3.6	The robot shall be able to detect if its frame can fit into a pathway, and manoeuvre itself through tight spaces.	1. Does the robot autonomously carry the object through the environment to the assigned destination.	Test	False
P-16	7.1.3.6	The robot shall be able to autonomously traverse its frame to the end site, once the object has been retrieved.	1. Does the robot autonomously path its way from the retrieval site, upon completion of the task, to the end site	Test	False
P-17	7.1.4.1 7.1.3.1	The robot shall maintain full functionality on a flat, even concrete surface.	1. See if the robot is fully functional on the desired surface.	Demonstrate	False
P-18	7.1.4.1	The robot shall handle wooden cubes with dimensions that do not exceed 50mm per side and weigh no more than 50 grams enabling Trossen PincherX 150 manipulator to grasp the object.	1. Ensure that the manipulator can pick up an object that is within the assigned size parameters. 2. Ensure that the manipulator can pick up an object that is within the assigned weight parameters	Test	False
P-19	7.1.4.1	The robot shall incorporate dual sensors to detect obstacles within a minimum range of 1 meters from its frame with sensor accuracy such that false detection are less than 1%	1. Does the robot use both a LiDAR sensor and real camera. 2. Is the minimum range 1 meter. 3. Are false detections less than 1% of the total sample of readings.	Demonstrate	False

Requirement No.[A]	**Section**[B]	**Requirement**[C]	**Verification Success Criteria**[D]	**Verification Method**[E]	**Pretest Acceptance**[F]
P-20	7.1.4.1	The robot shall have the ability to generate and update a virtual map in real time with a resolution that allows navigation through spaces as narrow as 120% of the robot's frame width, factoring in dynamic obstacles for manoeuvres.	1. Does the robot generate and update its virtual map in real time. 2. Does the robot and map allow navigation for a robot frame that is 120% of what the real frame is. 3. Does this navigation factor dynamic obstacles?	Test	False
P-21	7.1.4.2	The robot shall be powered for at least 4 hours of operation at full charge.	1. Does the robot have a maximum operation time of 4 hours.	Demonstrate	True
P-22	7.1.4.2	The robot shall operate reliably within a temperature range of 0-40°C.	1. Does the robot operate at room temperature. 2. Can the robot operate between the range provided?	Demonstrate	True
P-23	7.1.4.3	The robot shall not exceed dimensions of 500mm x 500mm x 500mm and a maximum weight of 15 kg to ensure navigability and ease of transport.	1. Is the robot's full frame within the assigned dimensions. 2. Is the robots weight below the threshold.	Demonstrate	False
P-24	7.1.4.4	The unit cost of production shall be less than the specified value, namely all materials in the toolbox and the components from the predefined list.	1. Was the robot system built with the defined economic limitations?	Demonstrate	False
P-25	7.1.4.6	The robot shall be equipped the robot with Wi-Fi and Bluetooth for transmitting status and environmental data to a central system every second.	1. Is the robot equipped with Wi-Fi and Bluetooth. 2. Do both of the communication components work as intended?	Demonstrate	True

A Unique Identifier for each unique Robotic System Requirement.

B Section number each unique Robotic System Requirement is contained within.

C Basic Text definition of each Robotic System Requirement.

D Success Criteria of each Robotic System Requirement.

E Verification method of the Robotic System Requirement that will be performed on each verification success criteria for each requirement.

F Indicate whether a requirement is also retested during pretest of the Robotic System, before the assessment.

7.2 Updated EDIA Workplace Charter

Introduction:

Dedicated to fostering an environment of Equality, Diversity, and Inclusion (EDI) within the Group 5 Robot Project team, this charter outlines the team's commitment to addressing EDI challenges in project operations, planning, organisation, and activities.

Equality:

1. Create a transparent process for opportunity allocation, ensuring all team members have access to new roles and learning experiences.
2. Use a standardised, objective system for evaluating contributions, involving multiple reviewers to mitigate individual biases.
3. Actively identify and rectify instances of discrimination or unequal treatment.

Diversity:

1. Celebrate and harness diverse skills, backgrounds, and perspectives.
2. Encourage varied ideas and approaches in robot design and building.
3. Implement a rotation policy in leadership roles to ensure diverse technical leadership

Inclusion:

1. Cultivate an environment where every member feels valued for unique contributions.
2. Implement a cross-disciplinary mentorship program during one of the weekly meetings for 30 minutes, enabling team members from different academic backgrounds to collaborate and share experiences with other members.

Project Operations:

1. Include an EDI checklist in project planning phases to ensure inclusivity in the robot building processes.
2. Develop guidelines for addressing EDI challenges during tasks.
3. Regularly assess and report progress in integrating EDI principles.

Planning:

1. Integrate EDI considerations into robot building task planning.
2. Incorporate EDI considerations into project timelines and milestones.

Organisation:

1. Keeping an EDI focus in all robotic relevant tasks.
2. Enforce a zero tolerance policy for discrimination or harassment.
3. Regularly assess and update EDI policies aligned with best practices.

Activities:

1. Organise team building activities promoting technical collaboration and celebrating diversity.
2. Communicate unwavering EDI commitment through updates, presentations, and external channels.

Conclusion:

By endorsing this Robot Project EDI Workplace Charter, each team member commits to upholding Equality, Diversity, and Inclusion principles. The team recognises that success is tied to creating an inclusive environment where every member contributes unique talents and perspectives.

Working Rules:

There will be two weekly 1 hour meetings each week to increase communication within the team. The first meeting will be at the start of the week on Tuesday to prepare for the Wednesday laboratory session and the second meeting will be on the Friday to discuss each members work that week and plans for the following week. Each team member gets an even distribution of workload that is determined during the weekly meetings.

Each member should work a minimum of four hours per week, including a Wednesday 14:00-16:00 lab session. Additional, online meetings shall be encouraged if a team member needs assistance in their distribution of work or if a deadline needs to be met.

Online sessions will be facilitated through social software applications such as Discord and Whats-app. Professional documentation will be shared through the secure university email service.

Team Member Rules:

1. Ensure all team members have freedom of speech and feel comfortable with proposing ideas to the group.
2. Each team member should record and update their evidence of work to a logbook, and they are responsible for maintaining their own logbook and weekly contribution.
3. One member takes the minutes of the meetings and uploads them, and the responsibility will be rotated each week, during the Friday meeting.
4. If you're unable to attend any of the meetings, inform the team 30 minutes prior to starting.
5. Attendance for at least one weekly meeting is mandatory (Unless informed 24 hours prior).
6. Each technical proposal needs to be discussed and each individual member must have an equal say on the decision.

7. Each team member is responsible for helping to clean the workspace in the lab before leaving.
8. For any technical problem faced, each member must give a potential solution by brainstorming before going for a group vote.
9. All decisions will be settled with a majority vote. In case of an even split in votes, where the decision cannot be resolved, each side must list down pros and cons of their stance and the decision should be then resolved by discussion based on the facts laid down, keeping the team's best interest in mind.

Grievances Procedure:

This team has an internal procedure to manage team member grievances if any arise, due to disrespect, exclusion or mistreatment by other members, regardless of their intentions.

For EDI conflict, the following flowchart should be strictly followed:

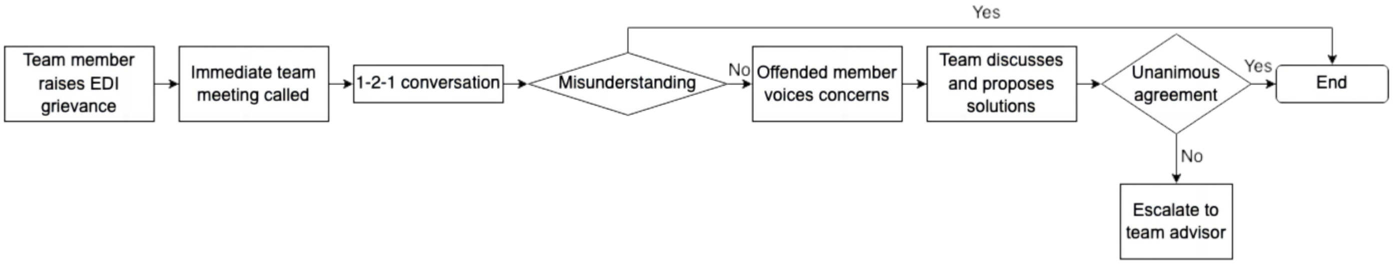

Team member raises EDI grievance
Immediate team meeting called
1-2-1 conversation
Misunderstanding
Yes
No
Offended member voices concerns
Team discusses and proposes solutions
Unanimous agreement
Yes
End
No
Escalate to team advisor

Figure 18: Flowchart showing the resolution of conflict between team members

For technical disagreement, the following flowchart should be strictly followed:

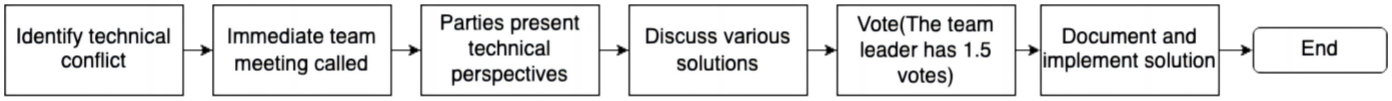

Identify technical conflict
Immediate team meeting called
Parties present technical perspectives
Discuss various solutions
Vote(The team leader has 1.5 votes)
Document and implement solution
End

Figure 19: Flowchart showing the resolution of technical disagreements between team members

References

[1] BS8622: “Guide to the Sustainable Design and Application of Robotic Systems”, https://www.bsigroup.com/

[2] Apple expands the use of recycled materials across its products, Apple, https://www.apple.com/newsroom/2022/04/apple-expands-the-use-of-recycled-materials-across-its-products/

[3] A to Z of Waste Type, University of Manchester, https://www.estates.manchester.ac.uk/media/services/estatesan

[4] Recycling E-Waste safely and responsibly, University of Manchester, https://www.socialresponsibility.manchester.ac.uk/all-news/legacy-news/recycling-e-waste-safely-and-responsibly/

[5] Leo Rover, “Leo Rover Documentation - Specification,” www.leorover.tech. https://www.leorover.tech/documentation/specification

[6] Intel, “Depth Camera D435,” Intel® RealSenseTM Depth and Tracking Cameras. https://www.intelrealsense.com/depth-camera-d435/

[7] Slamtech, “RPLIDAR A2, Low Cost 360 Degree Laser Range Scanner,” Aug. 04, 2022. https://bucket-download.slamtec.com/f65f8e37026796c56ddd512d33c7d4308d9edf94/LD310_SLAMTEC_rplidar_datasheet_A

[8] Trossen Robotics, “Interbotix X-Series Arms — Interbotix X-Series Arms Documentation,” docs.trossenrobotics.com, 2023. https://docs.trossenrobotics.com/interbotix_xsarms_docs/

[9] Intel® RealSenseTM Camera 400 Series Product Family Datasheet

[10] Systems Engineering Handbook. Rev. 1. Washington, D.C: National Aeronautics and Space Administration, 2007. Print. Appendix C

[11] Pfeifer, Michael. Materials Enabled Designs : The Materials Engineering Perspective to Product Design and Manufacturing. Amsterdam ; Elsevier/Butterworth Heinemann, 2009. Print. Chapter 2.3: Product Design Requirements

[12] Systems Engineering Handbook. Rev. 1. Washington, D.C: National Aeronautics and Space Administration, 2007. Print. Appendix D