# JEM 485 SENIOR DESIGN FINAL REPORT

## Course Information

Course: JEM 485

Semester/Year: Spring 2025

Project Title: Final Report

Team Members: Toby Desotelle, Logan Earp, Drew Jones, Nimue Meller, Brandon Murry, Bryan Prim, Ben Tucker

## Table of Contents

1. Introduction

2. Problem Identification & Working Criteria

3. Research Sources

4. Summary of Research Findings

5. Project Management + Multidisciplinary aspects

6. Feasible Alternatives

7. Analytical Methods + Alignment to Industry Standards

8. Project Development

9. Component Selection

10. Overall Design Process

11. Design Drawings

12. Project Testing

13. Analysis of Results

14 Ethical, Health, and Professional Impact on Society and Economy

15. Conclusion

## 1. Introduction

* Context: The objective of this project is to create a robot that will be able to compete and excel in the IEEE Mining Mayhem competition.
* Design Process: Group collaboration, prototyping, and testing were employed to attain the initial product, followed by integration and further testing to refine the product.

## 2. Problem Identification & Working Criteria

* Objectives: By the end of the year, our goal is to have a robot that can do the following:

1. Move out of the landing / start pad
2. Move out of the landing pad within 3 seconds of the start LED
3. Collect all pieces outside of the cave
4. Sort pieces correctly into their bins
5. Break the cave’s plane
6. Place the beacon
7. Place the containers with pieces into one of the landing pads

* Criteria: The robot must adhere to the specifications and requirements outlined in the IEEE rules document. The minimum criteria for a robot entry are as follows:

1. Size Limit: The robot must fit within a 12-inch x 12-inch x 12-inch cube at the start of the game and weigh under 26 lbs.
2. Excessive Reaching: The robot must not reach more than 12 inches past the playing field.
3. Field Damage: The robot must not damage the playing field, game elements, or other robots.
4. Safety: Robots must not pose a threat or risk to human safety.
5. Material and Component Usage: Flammable, toxic, or hazardous substances are prohibited, along with electromagnetic, optical, or acoustic devices that interfere with the playing field or other robots.
6. Control: Robots must operate autonomously without external communication after the start
7. Emergency Stop Button: Robots must have an emergency stop button.
8. Ethical considerations: The robot must not intentionally break any game rules or exhibit egregious behavior that violates the “spirit” of the competition.

## 3. Research Sources

* References:

1. Buckner, Eli. “Pose algebra, intro to ROS.” JEM 484 lecture, August 28th, 2024, UNCA, Asheville, NC
2. Buckner, Eli. “2D Pose.” JEM 484 lecture, September 4th, 2024, UNCA, Asheville, NC
3. Buckner, Eli. “Forward Kinematics.” JEM 484 lecture, September 11th, 2024, UNCA, Asheville, NC
4. Buckner, Eli. “Inverse Kinematics 1.” JEM 484 lecture, September 18th, 2024, UNCA, Asheville, NC
5. Buckner, Eli. “Inverse Kinematics 2.” JEM 484 lecture, October 30th, 2024, UNCA, Asheville, NC
6. *"Mining Mayhem - Game Manual 2."* Version 1.1.1, IEEE SoutheastCon Hardware Competition, 3 May 2024.
7. Phunopas, Amornphun, and Shinichi Inoue. "Motion Improvement of Four-Wheeled Omnidirectional Mobile Robots for Indoor Terrain." *Journal of Robotics, Networking and Artificial Life*, vol. 4, no. 4, 2018

## 4. Summary of Research Findings

* Our research findings included:

1. Information on Pose Algebra, which was used to determine the location of our robot.
2. Information on Kinematics, which was employed to make our robot move.
3. Information about ROS, which serves as middleware to our robot.
4. Information on the rules and regulations of the game as determined by IEEE.

## 5. Project Management

### Team Structure and Roles:

* **Toby**: 3D Printing, Navigation Lead, Coding, Team Leader
* **Bryan**: Treasurer, Camera
* **Brandon**: Secretary, Camera Lead
* **Ben**: Mechanical Design Lead
* **Nimue**: Software and Power System Lead
* **Logan**: Electrical Lead, Coordinator, Coding
* **Drew**: Mechanical Design and Assembly Assistant

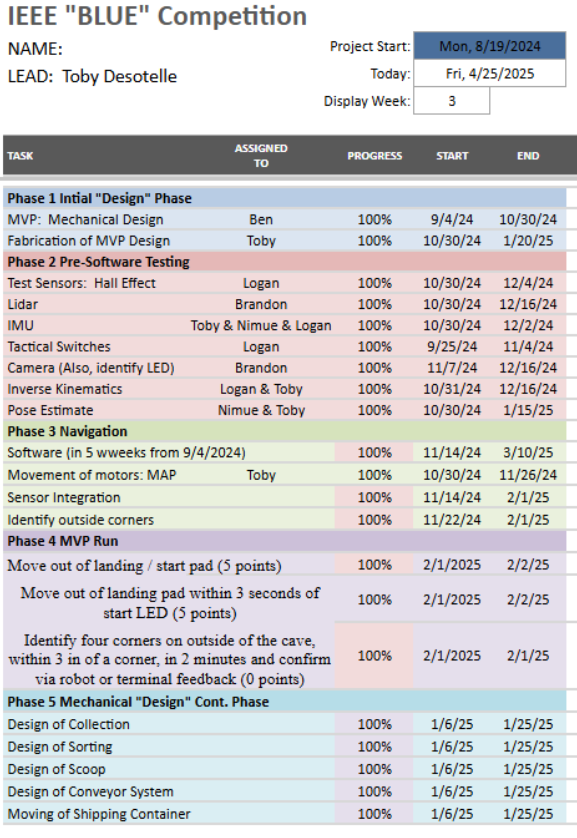
### 5.5 Multi-Disciplinary Participation:

The team members have actively collaborated across disciplines to ensure all components of the robot work cohesively. Below are some examples:

* **Mechanical & Electrical Integration**: Ben, the Mechanical Design Lead, worked closely with Logan to ensure that the mechanical frame was compatible with the electrical components, particularly the sensor mounting points and power system integration. Nimue assisted in ensuring that the power distribution was efficient for the robot’s motors and electronics.
* **Software & Hardware Collaboration**: Nimue, while primarily focused on software, collaborated with Logan to integrate motor control algorithms with the hardware. Toby also worked closely with the hardware team to ensure the navigation software was able to utilize sensor inputs for path planning and obstacle avoidance.
* **Testing & Design Participation**: Drew not only helped with the mechanical design but also contributed to testing the software by running hardware tests and debugging issues with sensor integration. Brandon contributed beyond his camera lead role by offering input on the mechanical aspects of the camera housing and working alongside Ben to ensure it fit seamlessly into the robot's design.

### Task Breakdown and Deadlines:

The task breakdown remains largely unchanged, but tasks have been grouped to ensure clarity:

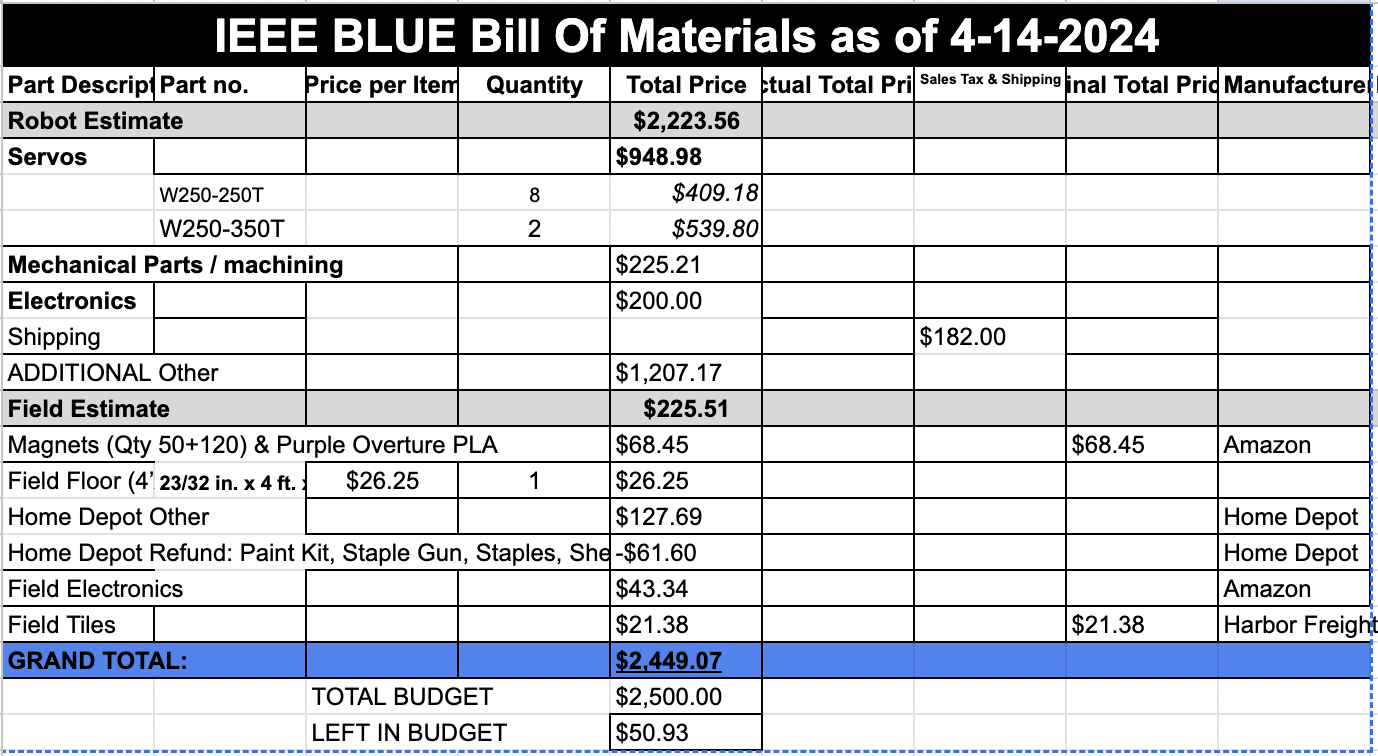
****

### Acquisition of Materials and Parts:

Parts were sourced from various online retailers, with priority given to high-quality, reliable vendors. For example, motors and sensors were sourced from trusted robotics suppliers, while custom parts were fabricated as needed through 3D printing and local vendors. The acquisition process is carefully managed to ensure timely delivery and avoid delays in production.

### Detailed Budget:

A budget breakdown is provided in the accompanying image, detailing expenditures for materials, equipment, and labor. The team utilizes a shared document to track expenses and ensure all purchases align with the project budget. Regular reviews are conducted to ensure financial goals are met and that the project stays within budget.



## 6. Feasible Alternatives

* Alternatives to this design included using various wheel configurations and wheel types. Based on research, the following options were not utilized:

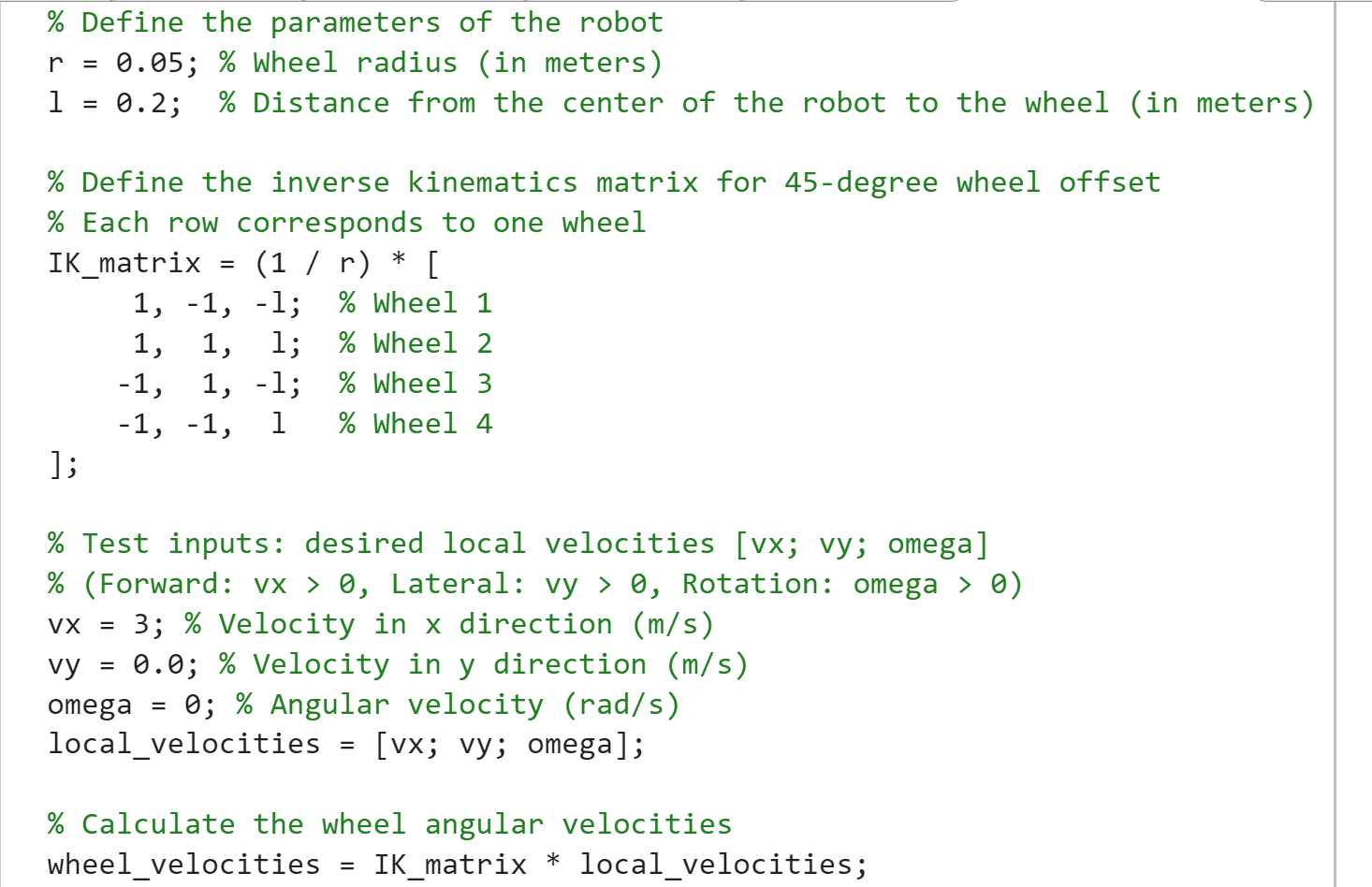
1. Mecanum Wheels: These wheels were shown to be prone to reduced efficiency and slippage, increasing control complexity
2. 3-Wheeled configuration: Provides a lack of stability and terrain adaptability.

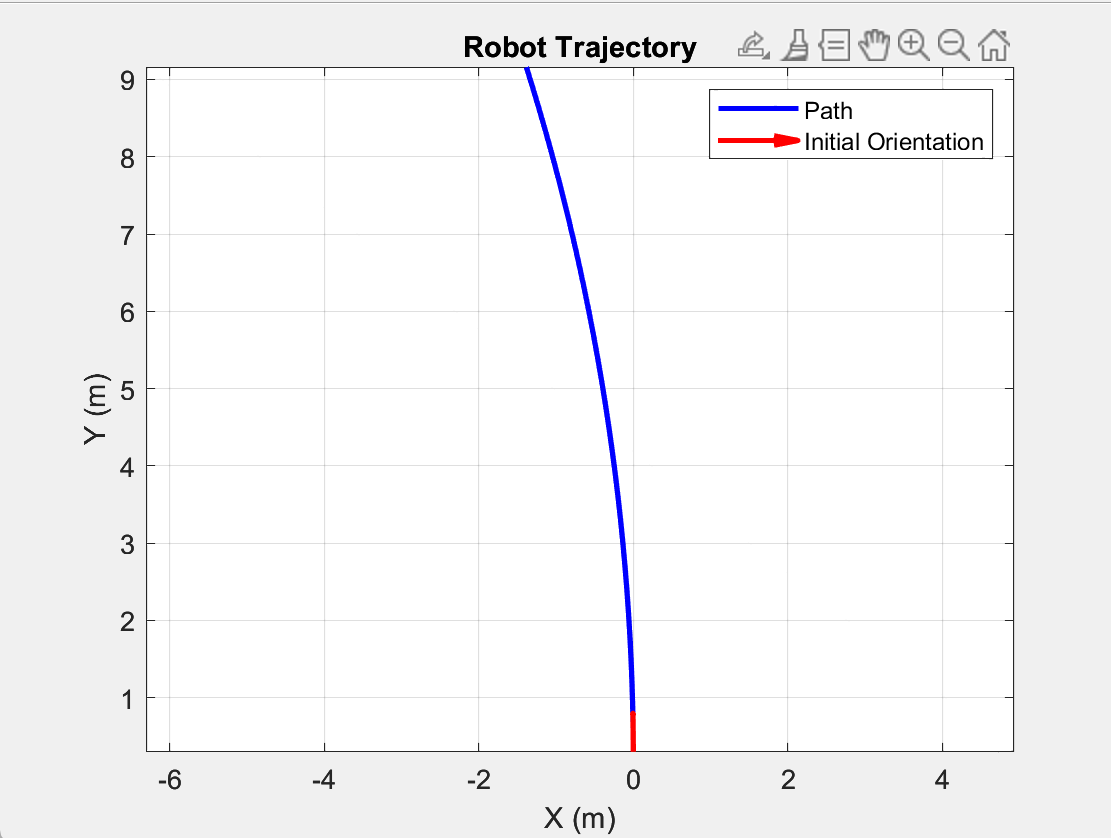
* Alternatives to this design included the use of different electronic components. Based on research, the following options were not utilized:

1. Arduino UNO Board: It was determined that the OpenCR board functioned as an Arduino and provided many additional features.
2. Raspberry Pi 5: Although the Pi 5 has greater computing power, it only supports ROS 2. ROS 2 is a newer version of ROS that does not have all the code libraries that we plan to use.

## 7. Analytical Methods

* Scientific Basis: We needed to determine how our wheel velocity relates to the robot's overall orientation. We used forward kinematics to derive the Jacobian matrix of a 4-omni wheel robot. That gives us an estimate of our current position from reading wheel velocities, but we also need a way to send our desired velocity and have the robot go to a specific position. To do this, we had to take the inverse of the Jacobian to solve for wheel velocities. With both Forward and Inverse kinematics solved, we can control and monitor our robot's position to a high degree.
* Principles of robotic interaction, linear algebra calculations, and data analysis techniques.
* MATLAB for computation and graphics, Gazebo for simulation, and RViz for visualization and debugging.



* This is a MATLAB script to verify that our team correctly derived the Jacobian and inverse Jacobian matrix. The code allows you to adjust the desired velocity to verify the simulated bot will navigate as expected.
* This is a Diagram of the output of the inverse kinematic MATLAB script, showing the results of a velocity in the x direction of 3 m/s and a rotational velocity of 0.1 rad/s

## 8. Project Development

1. Initiation:
2. Define objectives and requirements, and identify solutions for autonomy
3. Identify constraints
4. Planning:
5. Create a Gantt chart to track progress
6. Define team roles and responsibilities
7. Create deadlines for task completion
8. Define measurable goals to accomplish
9. Establish a route of communication among the team
10. Execution:
11. Assign tasks and responsibilities to group members
12. Implement those tasks to develop the project
13. Track progress via the Gantt chart
14. Check-in with group members periodically to update progress
15. Update goals and tasks based on results
16. Completion:
17. Ensure the final product functions as intended and meets requirements
18. Provide documentation in the form of a final report

## 9. Component Selection

1. Depth camera: Used to give feedback on location and allows us to locate specific items. The depth camera can measure the distances of objects in relation to the camera, which can be interpreted as a 3D representation of the environment. The camera can be leveraged to identify and avoid obstacles, identify objects, and estimate pose.
2. LiDAR: The LiDAR provides omnidirectional sensing by providing four sides of coverage. This allows the robot to know its location without the need to rotate. The LiDAR provides high-accuracy measurements that can be used in a variety of environments. The component can be implemented to assist in object tracking and obstacle avoidance.
3. Open CR board: This board is compatible with ROS (Robot Operating System) Dynamixel motor control, which simplifies our robot’s communication and control. The board has high processing power and multiple connectivity options. The board has built-in sensor integration and features, and an IMU (Inertial Measurement Unit), which can provide data for orientation, acceleration, and angular velocity.
4. Dynamixel servos: These servos allow for precise control by offering real-time feedback for position, torque, and voltage. Using Dynamixel servos allows us to daisy chain them together for efficient wiring on a single communication bus. They also feature a high torque-to-size ratio.
5. Raspberry Pi 4 B: Used to manage communication between navigation feedback components and servos via ROS middleware. Provides an ecosystem that supports many different programming languages and facilitates a modular approach to the creation of our robot's code.
6. Omnidirectional wheels: These give the robot full mobility, allowing travel in any direction without the need to rotate first. The wheels provide enhanced maneuverability for tight spaces and precise positioning as well. Using these wheels also simplifies our mechanical design by eliminating the need for a steering mechanism.
7. 3D printed frame: The frame will be 3D printed for cost-effectiveness, weight, production time, and ease of manufacturing.

## 10. Overall Design Process

1. Definition of Requirements and Objectives:

We began the design process by first determining how our robot should function and the specifications as mentioned in the IEEE rules doc. These requirements also have to be considered with our budgetary restraints.

1. Research and planning:

In this phase, we explored the benefits and detriments of different options such as wheel configuration, sensors, various electrical components, software environment, methods of achieving objectives, and many other aspects. We examined other robots that functioned similarly. We also referenced lectures and publications that have further information on kinematics.

1. Conceptual design:

Here we began collaborating on rough ideas for a design that would fulfill the requirements of our initial robot and give us a path forward for future additions to our final robot. Here we determined the wheel orientation and location of electronic components.

1. Detailed design:

Using SolidWorks, we completed a detailed design for the robot.

1. Assembly:

In this phase, we assembled our robot. We verified that the robot met the size requirements and installed all necessary components for operation.

1. Software:

At this point, we have to create ROS nodes that will be managing calculating of the wheel velocities to achieve our desired position while also reading the feedback from the motors to get an accurate estimation of the pose.

1. Testing: At this stage, we ran tests of our final product, ensuring reliability, repeatability, and robustness.

## 11. Design Drawings

* Figure 1 shows a view of the finished bot, focused on the collection system

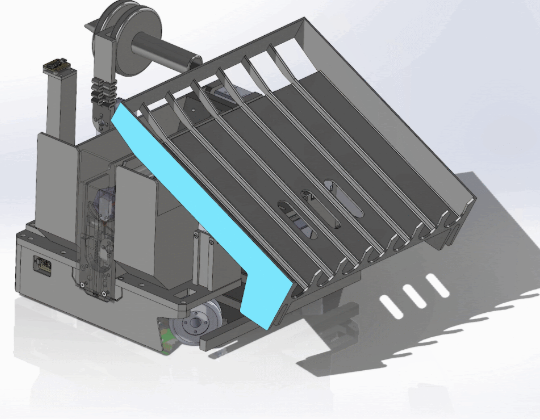


Figure 1

* Figure 2 shows an alternate view, with a focus on the sorting system

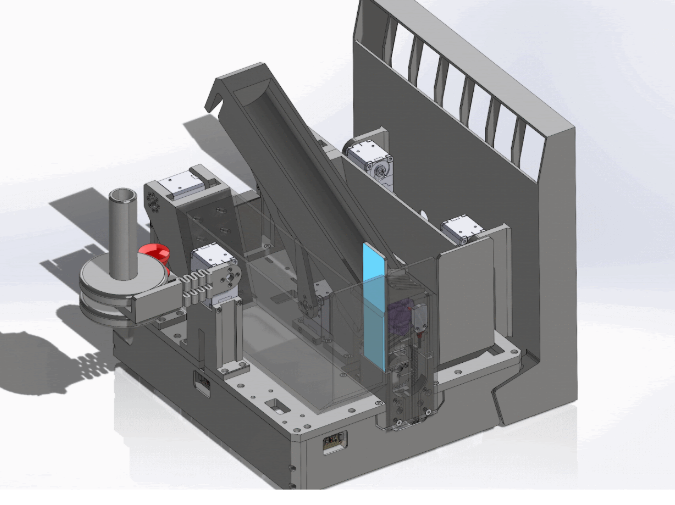


Figure. 2

* Figure 3 shows an exploded view of the full robot with each of its components isolated

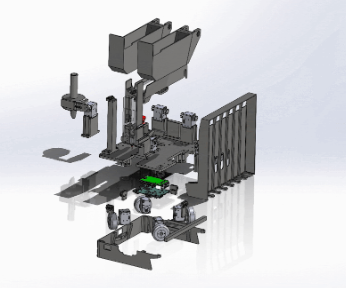


Figure 3

## 12. Project Testing

* Component Testing: The following components were tested individually to ensure that they operate and function as intended:

1. Robot Chassis: 3D printed and assembled. Tested to verify the benefits of modular use to swap out different components. Proved to be useful, but may limit future design potential
2. Dynamixel Servos: Tested to ensure functionality. We were able to determine that we could control the motion of the servos and get accurate encoder data from them.
3. LiDAR: Tested to ensure proper operation.
4. Depth Camera: Tested to ensure proper operation.
5. OpenCR Board: Tested to ensure proper operation.
6. Raspberry Pi: Tested to ensure proper operation.
7. Omnidirectional Wheels: Tested to ensure proper operation.

* Subsystem Integration Testing: The following subsystems were tested using the previously mentioned components to ensure that they function as intended:

1. Navigation:

Components used: Robot Chassis, Dynamixel Servos, Omnidirectional wheels, OpenCR Board, and Raspberry Pi

Tests include:

* Verification of frame design to accommodate the servos and all other electrical components via fabrication and installation.
* Verification and testing of communication between the Raspberry Pi, the OpenCR board, and the servos via programming, feedback, and robot movement. Tuning is applied in between tests.

1. Navigation feedback:

Components used: LiDAR, Depth Camera, OpenCR Board, and Raspberry Pi

Tests include:

* Verification of accurate object detection and getting the position of game pieces for path planning.
* Verification that the LiDAR can sense small enough distances accurately enough for localization and object detection.
* Performance testing: The following tests were done to verify performance and robustness:

1. Navigation Accuracy:

Tests include:

* Publishing a pose we want our robot to achieve, measuring its final position, and making sure it is within an acceptable tolerance of about 95%.

1. Start / Stop Confirmation:

Tests include:

* Trials to ensure that the robot can start and stop upon the push of a button, followed by having it work upon detection of the LED sensor.
* A launch code was made so that upon bootup, the robot’s operating system would immediately start, readying for LED signal.

1. Environmental Variability:

Tests include:

* Versatile Testing Scenarios: Easily change and customize game boards to test various scenarios and robot interactions.
* Simulated Environments: Gazebo allows for creating realistic game board simulations, enabling testing without physical setups.
* Real-Time visualization: RViz provides real-time visual feedback, helping to quickly identify and troubleshoot issues.

1. Randomized Conditions:

Tests include:

* Detecting the April tags and verifying that we can read the April tags in different orientations and angles.
* Placement of astral material in different locations.
* Detecting the astral material using an AI detection model in a variety of lighting conditions and orientations to achieve robust detection.

1. Stress Testing:

Tests include:

* Testing how long our robot can operate on a fully charged battery pack.
* Testing our frame to make sure it can support the full load of our robot with a safety factor of 1.5.

## 13. Analysis of Results + Alignment to Industry Standards

* Comparison to Original Design Objectives: The final robot design adhered closely to the objectives set at the project's inception. It achieved critical capabilities such as moving out of the starting pad within 3 seconds and identifying target corners with high precision. These outcomes were facilitated by the integration of advanced components like the OpenCR board and Dynamixel servos, and through the effective application of kinematics and pose estimation techniques discussed in our lectures.
* To ensure safety and compliance with industry standards, we followed relevant guidelines. For example, the design adhered to IEEE Robotics Competition Rules, ensuring that the robot met the necessary size, weight, and performance requirements. We also incorporated safety features to meet ISO 10218-1 standards, particularly regarding motor operation and emergency stop functionality. In line with ANSI/RIA R15.06, which focuses on safety requirements for industrial robots, we ensured feedback mechanisms were in place to maintain safe control during operations. Measures were taken to prevent electrical hazards such as overloads, ensuring the robot operated within safe parameters.

## 14. Ethics

* While the team never had to be concerned about any ethical dilemmas regarding autonomous decision-making concerns, as the robot’s motion was highly constrained, we have been emphasizing the sensor data to make it very transparent for all of the robot’s information and movement.
* We have also ensured the safety of the electrical and mechanical systems by testing all electrical contact points and ensuring the mechanical design was within our constraints. We have also added an Emergency stop to ensure that, should the robot act in any way too erratically or pose any danger, there is the option to fully disengage it.

## 15. Conclusion

* Lessons Learned: Throughout the project, we learned the importance of a modular design approach, which allowed for flexible testing and easier integration of different subsystems. The challenges of real-time sensor data integration taught us valuable lessons about the need for robust programming and testing environments.
* Achievement of Objectives: The project successfully met its design objectives by constructing a robot that not only fits within the specified dimensions and safety regulations but also performs its designated tasks. The ability of the robot to adhere to competition standards and its performance in test scenarios highlight the success of the project management and design process.