Final Report

Lunar ROADSTER

Team I

Author: **Ankit Aggarwal**
Andrew ID: ankitagg
E-mail: *ankitagg@andrew.cmu.edu*

Author: **Deepam Ameria**
Andrew ID: dameria
E-mail: *dameria@andrew.cmu.edu*

Author: **Bhaswanth Ayapilla**
Andrew ID: bayapill
E-mail: *bayapill@andrew.cmu.edu*

Author: **Simson D'Souza**
Andrew ID: sjdsouza
E-mail: *sjdsouza@andrew.cmu.edu*

Author: **Boxiang (William) Fu**
Andrew ID: boxiangf
E-mail: *boxiangf@andrew.cmu.edu*

Supervisor: **Dr. William "Red" Whittaker**
Department: Field Robotics Center
E-mail: *red@cmu.edu*

February 13, 2026

Carnegie Mellon University
Robotics Institute

Abstract

The Lunar ROADSTER (Robotic Operator for Autonomous Development of Surface Trails and Exploration Routes) is an autonomous lunar-working rover designed to groom traversable trails that enable sun-synchronous circumnavigation and long-duration mobility on the Moon. The system features a custom front dozer, developed through iterative design and testing, capable of crater grooming and backblading using a high-force linear actuator.

The rover uses a global 3D Moon Yard map with a preprocessing pipeline that classifies craters and provides geometric data for navigation. The perception system, powered by a custom-trained YOLOv8 model, detects target craters in real time and computes source, sink, and backblade poses for manipulation. Combined with a Pure Pursuit controller for Ackermann navigation, the system enables smooth, accurate motion through complex terrain while the tool planner coordinates blade actuation for excavation and grading. A dedicated validation module evaluates terrain slope from depth data to verify traversal requirements.

During the Fall Validation Demonstration (FVD), the system successfully performed fully autonomous crater identification, navigation, grooming, and revalidation across multiple craters. The rover achieved an average path-following deviation of 7%, well within the 10% requirement. Localization robustness was significantly improved through the integration of the SkyCam-based global localization method and reduced delay in pose updates. Additionally, upgrading to the Jetson Orin improved onboard compute, enabling real-time perception, validation, and navigation without bottlenecks.

Lunar ROADSTER lays the foundation for autonomous lunar surface preparation, supporting future infrastructure, resource transport, and sustained human presence on the Moon.

Contents

1 Project Description

Humanity is preparing to return to the Moon, with the Artemis missions focusing on exploring the South Pole, a region rich in sites of interest. Establishing a circumnavigating route around the lunar pole will serve as a critical "highway" connecting these sites and enabling key activities such as transportation, human settlement, and resource extraction.

A solar-powered rover capable of sun-synchronous circumnavigation could achieve perpetual operation by avoiding lunar sunsets. At high latitudes, this is feasible at low speeds, as shown in Table 1. However, these assumptions rely on the terrain being flat and traversable, free from major topographical challenges. A mission to manipulate the lunar regolith in the circumnavigating path to make it more traversable for future missions is thus, a clear step forward. A robotic system can be designed to conduct these operations efficiently for extended durations.

Table 1: Average Speed Required to Circumnavigate at Different Latitudes on the Moon

Latitude	Distance (km)	Speed (kph)
Equator	11,000	16
50°	7,040	10
60°	5,500	8
70°	3,700	6
75°	2,800	4
80°	1,870	3
81°	1,529	2.5

The Lunar Robotic Operator for Autonomous Development of Surface Trails and Exploration Routes (Lunar ROADSTER) is an autonomous moon-working rover, capable of finding exploration routes and grooming the lunar surface to develop traversable surface trails. These groomed trails will become the backbone for the colonization of the Moon by enabling transportation, logistics, and enterprise development.

2 Use Case

The conceptual use case for the Lunar ROADSTER system is illustrated in Figure 1. The process begins with the system receiving detailed maps of the user-specified latitude from prior exploration missions, such as orbiters or exploratory rovers. Due to the natural irregularities of the lunar surface, this initial reference latitude (white dotted line) often intersects with craters, dunes, and rough terrain, resulting in a non-traversable original path (blue line). These irregularities make the path non-traversable for standard solar-powered rovers, which rely on relatively flat terrain for safe and efficient motion. Traversing such terrain would not only consume excessive energy but also increase the risk of mechanical failure or mission interruption.

To address this, Lunar ROADSTER autonomously navigates this original path and identifies areas requiring terrain conditioning. Using onboard perception and terrain analysis, it classifies craters into two categories: those that are too deep or wide to groom (red), and gradable craters (orange) that fall within the rover's manipulation capabilities. The rover uses a custom-designed dozer blade to push regolith from the rim into the crater, smoothing the terrain and forming a groomed, traversable path (green line). This path is shorter, safer, and more energy-efficient than the original route.

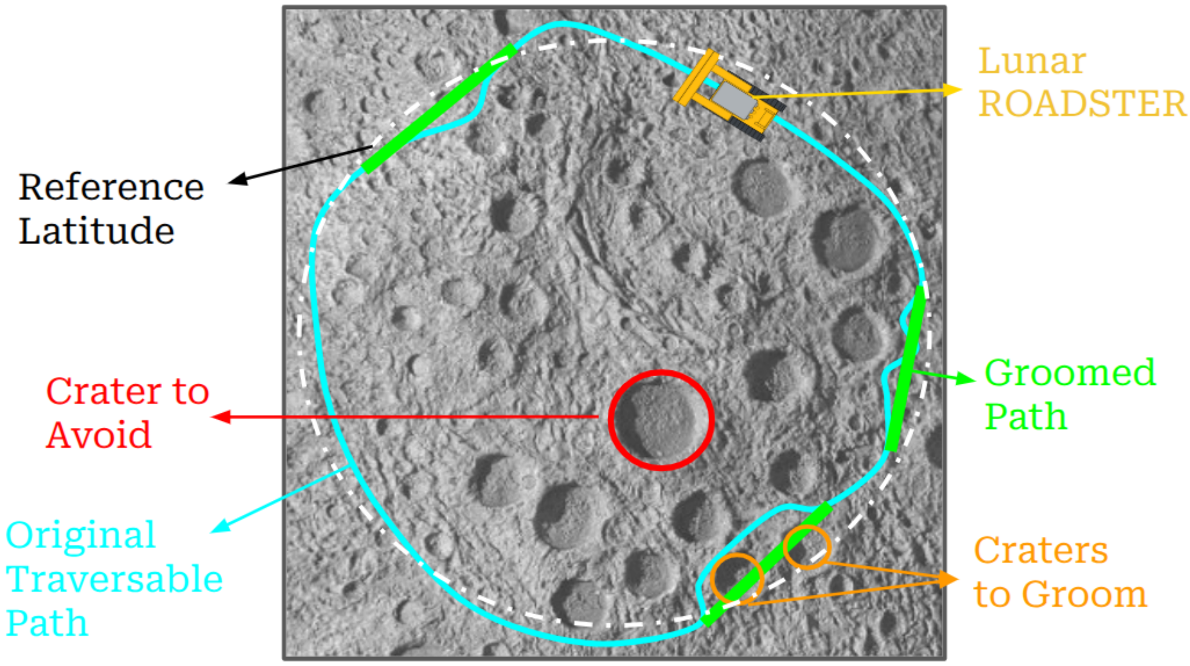


Figure 1: Conceptual System Graphic Representation

After completing the initial grooming operation at a designated crater, the Lunar ROADSTER does not immediately continue along its path. Instead, it retreats slightly to a vantage point where it can scan and assess the modified terrain using its onboard perception system, such as a depth camera. This evaluation step is critical to ensure that the crater has been adequately filled and leveled to meet defined traversability criteria. If the surface still exhibits irregularities such as steep inclines, or dips, the rover autonomously initiates a re-grooming cycle. It re-plans its manipulation trajectory and performs another round of dozing or backblading, adjusting the terrain further. This closed-loop grooming-evaluation cycle continues iteratively until the rover confirms, through sensor feedback, that the modified surface is suitable for safe traversal. Once validated, the rover marks the crater as complete and advances to the next waypoint, progressively transforming the rugged trail into a continuous, traversable path.

This use case demonstrates Lunar ROADSTER's ability to autonomously transform hazardous lunar terrain into a continuous, navigable highway, enabling long-duration sun-synchronous missions and supporting future lunar exploration and infrastructure.

3 System-Level Requirements

The system requirements for the Lunar ROADSTER project are derived from a comprehensive understanding of the problem statement, its use cases, and the high-level objectives. These objectives shown in Figure 2, informed by inputs from stakeholders, provide a clear framework for defining the system requirements.

The requirements are organized into mandatory and desirable categories, further classified into functional, performance, and non-functional requirements. The mandatory requirements form the core functionalities essential for the project's success, while the desirable requirements, though initially out of scope, aim to enhance the system's overall performance. The requirements may evolve as the system develops, further research is conducted, and tests refine the design. The team will focus on meeting all mandatory requirements by project deadlines while working to implement desirable ones as resources permit.

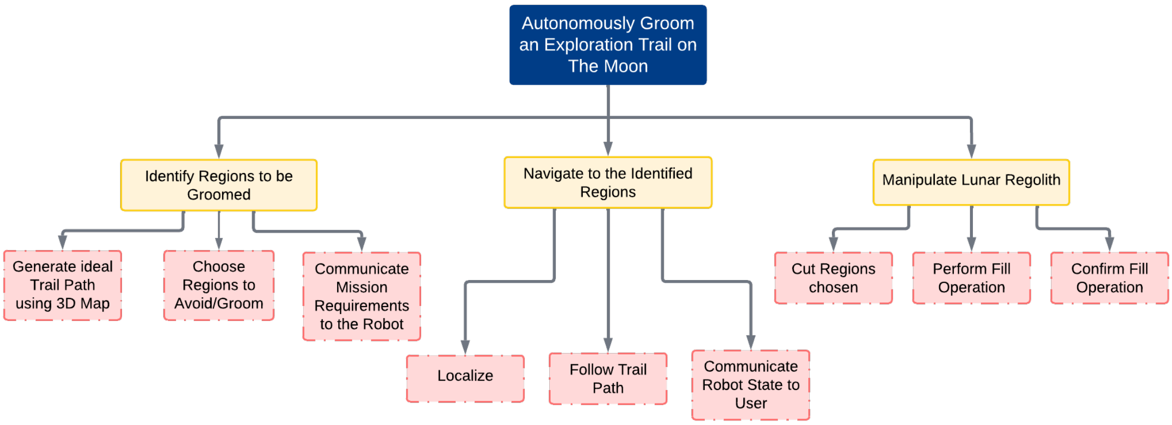


Figure 2: Objectives Tree

3.1 Mandatory Requirements

3.1.1 Mandatory Functional Requirements

Table 2: Mandatory Functional Requirements

Sr.No.	Mandatory Functional Requirement (Shall)
M.F.1	Perform **trail path planning**
M.F.2	Operate **autonomously**
M.F.3	**Localize** itself in a GPS denied environment
M.F.4	**Navigate** the planned path
M.F.5	Traverse **uneven terrain**
M.F.6	Choose craters to **groom and avoid**
M.F.7	**Grade** craters and **level dunes**
M.F.8	**Validate grading** and trail path
M.F.9	**Communicate** with the user

3.1.2 Mandatory Performance Requirements

Table 3: Mandatory Performance Requirements

Sr.No.	Performance Metrics
M.P.1	Plan a path with **cumulative deviation of** $\leq$ 25% from chosen latitude's length [2]
M.P.2	**Follow planned path** to a maximum deviation of **10%**
M.P.3	Have a **contact pressure of less than 1.5 kPa** [3]
M.P.4	**Avoid craters** $\geq$ 0.5 **metres**
M.P.5	Fill craters of **up to 0.5 meters** in diameter and **0.1 meter** in depth [1]
M.P.6	Groom the trail to have a **maximum traversal slope of** 5°

3.1.3 Mandatory Non-Functional Requirements

Table 4: Mandatory Non-Functional Requirements

Sr.No.	Parameter	Description
M.N.1	Weight	The rover must weigh **under 50 kg (Achieved: 23.8 kg)**
M.N.2	Cost	The cost for the project must **be under $5000 (Achieved: $4,993)**
M.N.3	Computing Capacity	The onboard computer should be able to **run all required tasks (Achieved: Shown in Appendix)**

3.2 Desirable Requirements

3.2.1 Desirable Non-Functional Requirements

Table 5: Desirable Non-Functional Requirements

Sr.No.	Parameter	Description
D.N.1	Technological Extensibility	The system will be **well documented** and designed so that future teams can easily access and build on the work
D.N.2	Aesthetics	Requirement from sponsor, the rover must **look presentable and lunar-ready**
D.N.3	Modularity	To enable **tool interchangeability**, the tool assemblies must be modular and easy to assemble/disassemble
D.N.4	Repeatability	The system will **complete multiple missions without the need of maintenance**

4 Functional Architecture

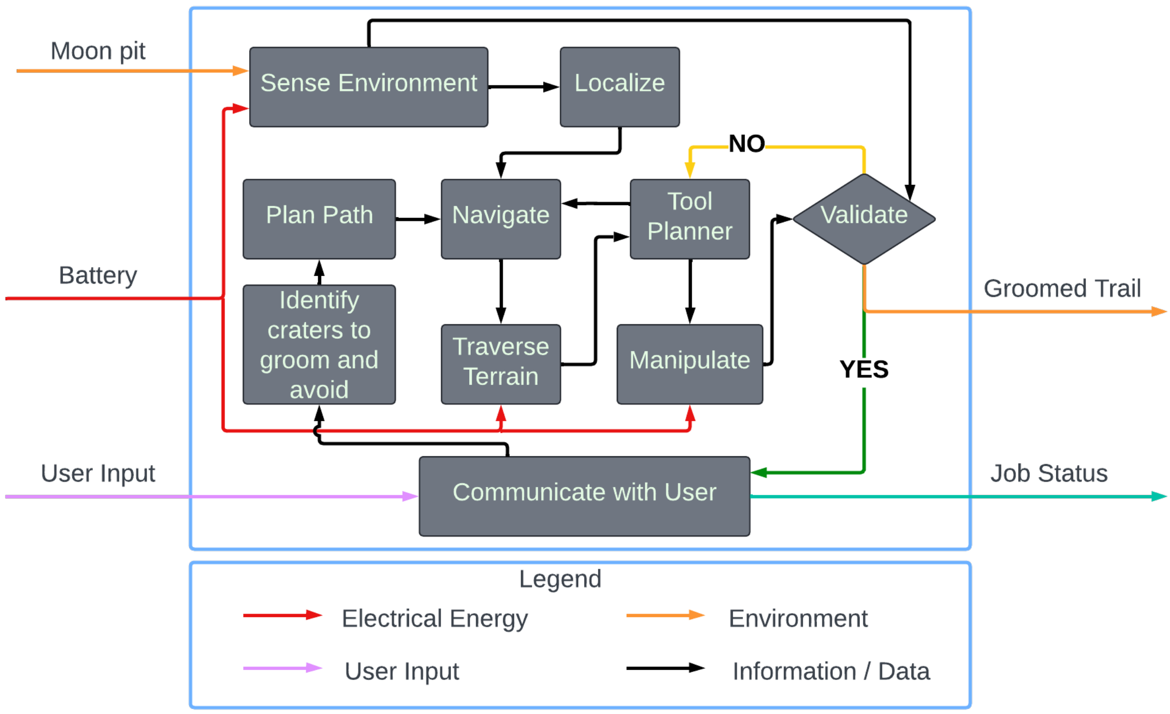


Figure 3: Functional Architecture

Figure 3 illustrates the functional architecture of our system. The system receives three types of input: user input in the form of a map of the environment from the operational terminal, battery input as the electrical energy that powers the components, and environment input from the moonpit (worksite).

The `Communicate with User` block serves as the critical interface between the user and the system. It transmits the map to the `Plan Path` algorithm and updates the user with real-time job status information for active monitoring. The `Plan Path` algorithm processes environmental information from the map to identify craters to groom and avoid. These constraints are defined in the performance requirements (M.P.4 and M.P.5). Based on this analysis, it generates precise waypoints near the craters requiring grooming and sends them to the `Navigate` block. Before initiating navigation, the robot undergoes localization through the `Localize` block, using information from the sensor stack to accurately determine its position within the environment. The `Navigate` block translates the planned waypoints into motor commands for the chassis, which are executed by the `Traverse Terrain` block, enabling the robot to maneuver through the moonpit and approach the target crater effectively.

Once positioned near the crater, the `Tool Planner` is activated, sending motor commands to the `Manipulate` block to initiate tool operations, such as excavation and grading. The grooming process is then evaluated by the `Validate` block to determine if the crater meets the specified grooming criteria, as defined in the performance requirements (M.P.6). If validation fails, the system repeats the cycle, navigating the robot back to the position near the crater and restarting the tool operation. If grooming is successful, the `Communicate with User` block updates the user with the job status, and the system outputs a groomed trail. This iterative and modular workflow ensures precise grooming operations while maintaining active user monitoring and operational reliability.

5 System and Subsystem-Level Trade Studies

Trade studies are an integral part of decision making in the systems engineering process. It identifies the most balanced technical solution among a set of proposed viable solutions and determines which viable architecture or system we should use. Rigorous trade studies were performed on the systems-level and important subsystems to determine the most viable solutions. A summary of the chosen architecture from the conducted trade studies can be found in the morphological chart in Figure 4.

5.1 Systems-Level Trade Study: Lunar Grader

A systems-level trade study was conducted to determine which lunar grader concept is best suited for meeting our performance requirements. Figure 5 identifies 4 potential concepts to use for a lunar grader. The criteria and weight factors were obtained via a weighted objectives tree and can be found in Appendix A.4.

Based on available concepts, we identified 3 different autonomous rovers suitable for lunar grading. They are the Lunar ROADSTER, Crater Grader (made by MRSD 2022 Team A), and the Offworld Dozer (made by Offworld.ai). We also include a benchmark and compare against human performance. Since lunar earth-working is a very dangerous task, we determined that safety should be of utmost importance. This is why it has a

Morphological Chart	Option 1	Option 2	Option 3	Option 4	Option 5
Path Planning	**A***	Dijkstra's Graph Search	Greedy Best First	D*-Lite	
Localization Method	**Total Station, IMU**	Sun/Star Sensor, Visual Odometry, Wheel Odometry, IMU	LRO Correspondences, Wheel Odometry, IMU	Motion Capture, IMU	Visual Odometry, Wheel Odometry, IMU
Navigate	Pure Pursuit	RRT	**Dynamic Window**	Incremental Search	
Wheels	Air Filled	**Metal**	Plastic	Treads	
Chassis	Space Frame	**Ladder Frame**	Unibody	Monocoque	
Suspension	Rocker Bogie	**Double Rocker**	Multi-Link	Trailing/Leading Arm	Macpherson Strut
Motors	BDC	**BLDC**			
Drive System	**Gearbox**	Belt Drive	Chain Drive		
Powertrain	**Lithium Based Battery**	Solar Cells	Isotope		
Decision Architecture	**Finite state machine**	Single state machine			
Cut/Fill Methodology	**Custom Algorithm**	Kubla Software			
Manipulate	Front loader	Front grader	Chassis grader	**Front loader & chassis grader**	
Validate	**Depth Camera on belly of rover**	LiDAR	Camera on top	IR Sensor on belly of rover	RADAR
Communicate With User	2.4 GHz Wi-Fi	**5 GHz Wi-Fi**	Bluetooth		
Sensor Fusion Method	**Extended Kalman Filter**	Particle Filter	Bayes Filter		

Figure 4: Morphological Chart of Cyberphysical Architecture

Trade Studies	Systems Level	Lunar Grader			
Value Ratings *	**Concept**	**Lunar ROADSTER**	**Crater Grader**	**Offworld Dozer**	**Human**
0: Inadequate		LUNAR ROADSTER			
2: Tolerable					
4: Adequate					
6: Good					
8: Excellent					
10: Perfect					
*** Subjective Value Method**					
Criteria	**Weight Factor**	**Value (1 - 10) ***			
Safety	12	7	7	9	0
Navigate autonomously	11	8	8	9	5
Ability to localize	11	8	7	9	1
Ability to grade	8.25	9	9	0	3
Ability to excavate	8.25	9	0	9	3
Traversability	8.25	7	7	5	8
Reliability	6	7	7	8	9
Weight	6	8	10	2	6
Cost	6	10	10	3	2
Tool Size	6	7	2	9	4
Repeatability	6	5	5	7	7
Operation time	4.95	7	7	9	2
Ability to communicate	3.3	8	8	8	8
Adaptability	3	6	5	5	10
Final Score	**100**	**7.673**	**6.6105**	**6.8145**	**4.158**

Figure 5: Systems-Level Trade Study on Lunar Grader Concept

weight factor of 12%. The 3 rovers are comparatively safe since they can operate autonomously. Contrastingly, the benchmark scores very low in safety due to human-prone accidents and a high fatality rate from space suit punctures and explosive decompression.

The ability to localize itself and navigate autonomously is also highly prioritized. This is because the aim of our system is to create a circular lunar polar highway. The concept needs to be able localize and navigate by itself so the path created does not deviate too far from its objective path. Additionally, the tool planner and the navigation planner both requires accurate localization to be able to function effectively. The three rovers all tend to perform well in these aspects, with the commercial Offworld Dozer arguably performing slightly better due to the use of commercial-grade sensors. However, the human benchmark performs poorly in this criteria. Without the proper navigational tools, humans can quickly become lost in the relatively featureless lunar surface.

Our next priority is the concept's ability to grade and excavate while maintaining traversability. This is what differentiates the Lunar ROADSTER concept from the other rover concepts. The Lunar ROADSTER concept can both grade and excavate, whereas the Crater Grader concept can only grade, and the Offworld Dozer and only excavate. The versatility of having both a grader and an excavator prove to be highly appropriate for our functional requirement of grading craters and leveling dunes (M.F.7). In the end, this is arguably the deciding factor to use the Lunar ROADSTER concept.

5.2 Subsystems-Level Trade Study: Manipulation

A trade study on which manipulation subsystem to implement is shown in Figure 6. This is arguably our most important subsystem as it pertains to our primary objective of grooming an exploration trail on the lunar surface.

Trade Studies	Sub-Systems Level	Manipulation			
Value Ratings *	**Concept**	**Front loader**	**Front grader**	**Chassis grader**	**Front loader & chassis grader**
0: Inadequate					
2: Tolerable					
4: Adequate					
6: Good					
8: Excellent					
10: Perfect					
*** Subjective Value Method**					
Criteria	**Weight Factor**	**Value (1 - 10) ***			
Excavation volume	17.5	9	1	1	7
Grading area	17.5	1	8	9	7
Manipulation effort	15	5	6	7	4
Dust contamination	12	1	5	7	7
Controllability	10	4	5	6	6
Degrees of freedom	10	7	5	5	9
Size	9	5	5	4	4
Weight	4.5	5	7	7	4
Cost	4.5	5	5	5	4
Final Score	**100**	**4.62**	**5.065**	**5.64**	**6.11**

Figure 6: Subsystems-Level Trade Study on Manipulation Concept

For the manipulation subsystem, we put heavy emphasis on the concept's excavation volume and grading area. This is because our path can be groomed faster when the excavation volume and grading area is large. Comparing the different concepts, the front grader and chassis grader have a high grading area, but negligible excavation volume. Contrastingly, the front loader has a high excavation volume, but is not able to grade efficiently. While not being able to perform at the level of the specialized concepts, the front loader plus chassis grader concept uniquely can achieve both a high excavation volume and a high grading area. This is the differentiation factor from the other 3 specialized concepts.

However, one drawback of the dual-machinery concept is that it requires a higher manipulation effort to operate. The size and weight of the rover chassis will also need to be larger to accommodate both a grader and loader. Despite this, the advantages of having two tools on trail grooming efficiency greatly outweigh its downsides. Thus, we have decided to use a combination of a front loader plus a chassis grader for our manipulation subsystem.

5.3 Subsystems-Level Trade Study: Localization Method

Our second most important subsystem is the localization method. This is because virtually all aspects of our rover requires accurate localization for it to function effectively. The tool planner subsystem requires localization to plan motor commands, whereas the navigation planner subsystem requires it to plan trajectories and paths. A trade study on which localization method to use is shown in Figure 7.

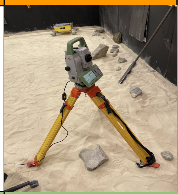
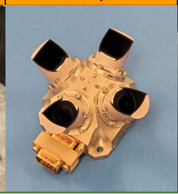
Trade Studies	Sub-Systems Level	Localization Method				
Value Ratings *	Concept	Total Station, IMU	Sun/Star Sensor, Visual Odometry, Wheel Odometry, IMU	LRO Correspondences, Wheel Odometry, IMU	Motion Capture, IMU	Visual Odometry, Wheel Odometry, IMU
0: Inadequate						
2: Tolerable						
4: Adequate						
6: Good						
8: Excellent						
10: Perfect						
* Subjective Value Method						
Criteria	Weight Factor	Value (1 - 10) *				
Accuracy	30	8	4	6	9	4
Robustness	18	8	4	6	8	2
Computational efficiency	12	8	2	3	7	2
Lunar transferability	12	4	9	9	1	9
Reliability	12	8	5	7	9	6
Ease of use	8	9	5	5	7	2
External infrastructure dependency	8	3	9	1	1	9
Final Score	100	7.2	4.96	5.64	6.82	4.48

Figure 7: Subsystems-Level Trade Study on Localization Method

An accurate and robust localization is a necessity. This is why these two criterion constitute 48% of the total weight factor. The total station and motion capture concepts score highly in these categories due to their low localization errors. The motion capture system arguably scores slightly higher due to its ability to discern both location and orientation whereas the total station can only determine location. The sun/star sensor, Lunar Reconnaissance Orbiter (LRO), and visual odometry concepts all score relatively low due to their high localization error rates.

However, a downside of the total station and motion capture system is that they are highly dependent on external infrastructure. The total station requires at least 1 surveyor whereas the motion capture system requires at least 4 cameras to function. This is why both score relatively low in external infrastructure dependency and lunar transferability. Despite this, a total station setup is relatively reasonable for a well-established moon station. In conjunction with its high localization accuracy and robustness, we have decided to utilize a total station as our localization method.

6 Cyberphysical Architecture

The Cyberphysical architecture, depicted in Figure 8, shows how our Lunar rover is physically realized. It integrates a network of the following major subsystems: Sensors, Computations, External Infrastructure, Mechanical subsystem, Actuation and Electronics, and Electrical Power. Each component plays a specific role, with all of them working together in unison to meet the unique demands of lunar surface operations.

6.1 Sensors

The rover relies of the following set of sensors for the essential data, which are crucial for navigating and executing tasks:

- Wheel motor encoders

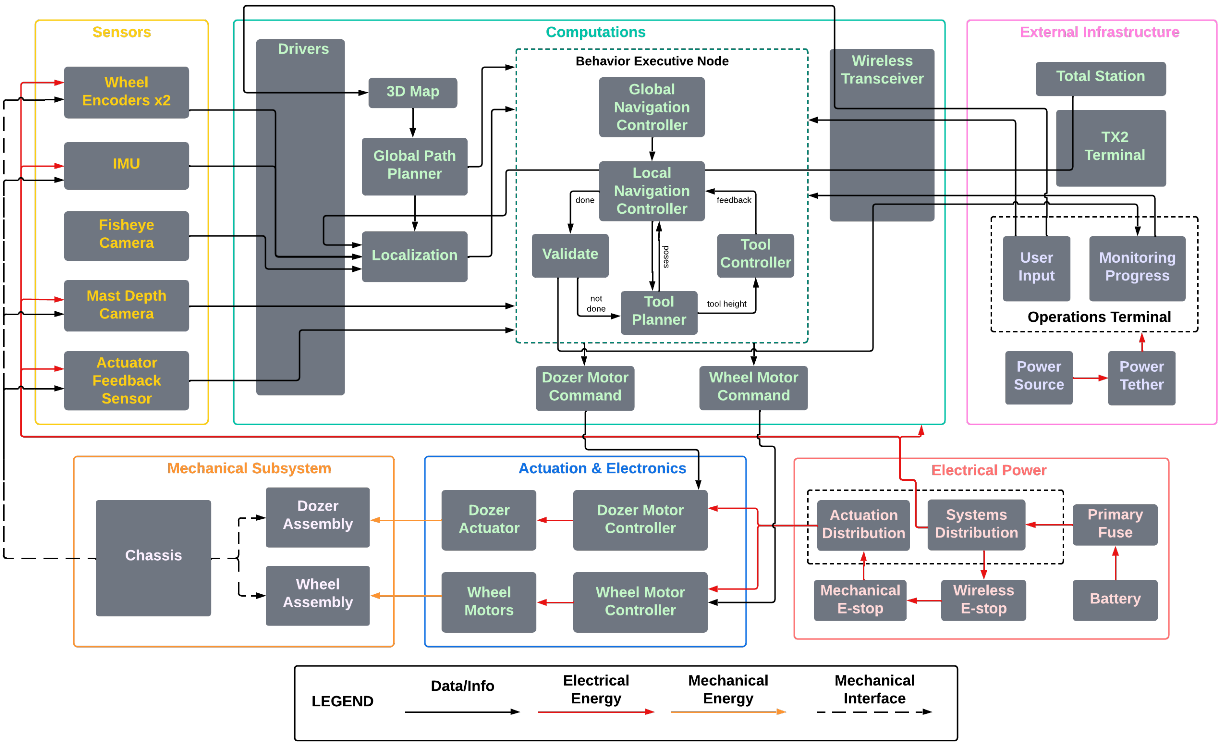


Figure 8: Cyberphysical Architecture

- Mast depth camera (ZED 2i)
- Inertial Measurement Unit (IMU)
- Actuator feedback sensor
- Fisheye camera

These sensors provide critical feedback on the rover's position, orientation, and material to manipulate.

6.2 External Infrastructure

The extrernal infrastructure comprises the Total Station, a Wireless Transceiver, and an Operations Terminal. The robotic total station provides precise robot pose estimates, and the operations terminal allows for seamless communication between the rover and mission control through the wireless transceiver, by providing a user interface to monitor progress and receive updates.

6.3 Computations

The computations subsystem is the processing powerhouse of the rover, where data from sensors are transformed into actions. It includes the following components:

1. The drivers form the interface between sensors and processing units.
2. A 3D map is fed into the 3D Map block through a wireless transceiver from the operations terminal. A global path planning algorithm runs directly on the Jetson Orin that serves as the brain of the rover, which processes the map, identifies gradable and ungradable craters based on their diameters, fits a ring through all

the gradable craters, and allows for planning a smooth path between craters for the rover to follow.

3. A robotic total station in the external infrastructure, which provides precise robot pose, sends its data to the localization block. This can also be replaced by the fisheye camera, which can be used to look at the ceiling in the Moon Yard and obtain position information of the rover by regressing points to the lights.
4. The data from the total station or fisheye camera is fused with the data from wheel encoders, and IMU, and sent to the localization block, which keeps track of the rover's position on the lunar surface.
5. The Behavior Executive Node manages high-level decision-making and receives inputs from the localization block and the global path that the robot has to follow. The set of waypoints obtained from the Global Path Planner is used by the Global Navigation Controller to move the robot close to the crater.
6. Once the robot is close to the crater, the Local Navigation Controller, Tool Planner (or Perception module), Validation, and Tool Controller all work together in order to achieve reliable grading.
7. The Tool Planner (or Perception module) is responsible for identifying the crater and extracting geometric information from it such as crater centroid and diameter. Using this information, grading poses are calculated and sent to the Local Navigation Controller.
8. The Local Navigation Controller takes robot poses from the Tool Planner as targets and sends commands to the Wheel Motor Command block in order to move the rover to the source and sink poses of the crater. During this operation, the Tool Controller also generates commands for terrain manipulation and passes them directly to the Dozer Motor Command block, which actuates the dozer blade for grooming operations.
9. Once a single pass of the grading operation is completed, the Validation block looks at the groomed terrain and verifies if the grading has been performed satisfactorily or not. If yes, it moves on to the next crater by going to the Global Path Planner block. Otherwise, it repeats another pass of grading.

6.4 Actuation and Electronics

This subsystem translates electrical signals into physical movements. The dozer motor controller receives commands from the dozer motor command block in the computations subsystem. Similarly, the wheel motor controller receives commands from the wheel motor command block in the computations subsystem. These controller blocks provide signals to the respective dozer actuator and wheel motors, which then make the respective assemblies connected to them in the mechanical subsystem move.

6.5 Mechanical Subsystem

It forms the structural backbone of the rover. The main components include the Chassis, Dozer Assembly, and Wheel Assembly. This subsystem provides both the physical support required for the rover and the mechanisms needed to interact with the lunar surface. All the sensors and hardware sit on the chassis of the rover, and the dozer assembly is used to groom the trail on the Moon.

6.6 Electrical Power

This subsystem is responsible for supplying energy to the entire rover. In the operations terminal, a power source supplies power to the robotic total station as well as the operations terminal via a tether. On the rover, a battery provides the electrical power and is connected to the Power Distribution Board (PDB) through a primary fuse for safety. The PDB allocates power to the Systems Distribution block and the Actuation Distribution block. The systems distribution block supplies power to all the subsystems on the rover, and the Actuation Distribution block supplies power to the actuators – motor controllers and motors. We also have a wireless emergency stop (E-stop), which translates to a mechanical E-stop to cut off all power to the actuators in case of an emergency.

7 Current System Status

7.1 Overall System Depiction

Figure 9 shows the overall system hardware depiction and Figure 10 shows the overall system software depiction.

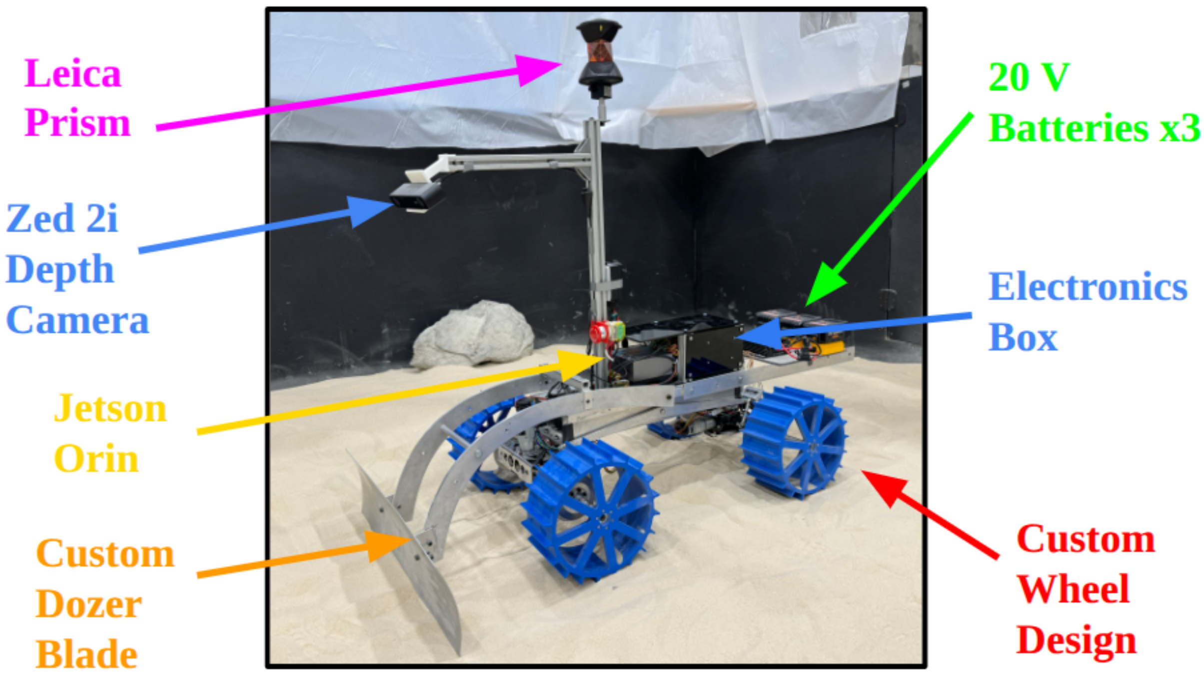


Figure 9: Overall System Depiction - Hardware

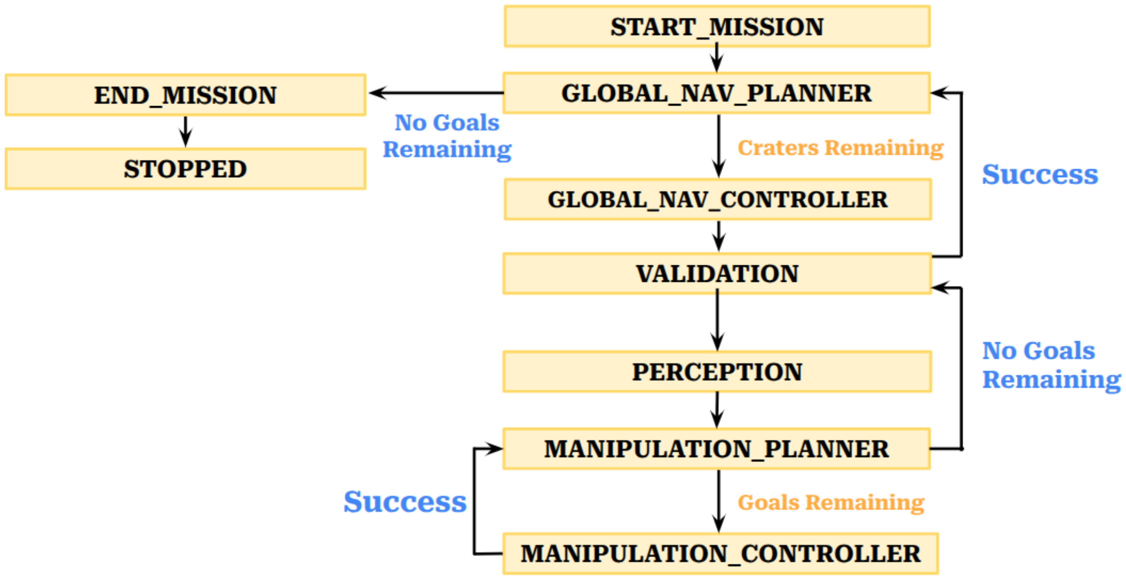


Figure 10: Overall System Depiction - Software

7.2 Subsystem Descriptions

The following section presents a comprehensive overview of all subsystems developed during the spring semester. It also summarizes the modeling, analysis, and testing conducted to meet design specifications and validate subsystem functionality. Appendix A.1 shows the completion status of each subsystem.

7.2.1 Sensors Subsystem

The sensors subsystem is responsible for capturing data from the environment and the internal state of the rover to support various computational tasks. It includes four wheel encoders, an ZED 2i depth camera, a VectorNav IMU, and a linear actuator.

The IMU is integrated using the VectorNav ROS2 package, and the wheel encoders are interfaced via micro-ROS. The linear actuator came with an onboard potentiometer for feedback. The depth camera is interfaced using ZED ROS wrappers, which provided point cloud data for perception.

We 3D printed the camera mount and explored optimal mounting angles for depth perception. The functionality of this subsystem was verified through targeted unit tests, including Optimal Mast Depth Camera Placement (T10) and a Maintenance and QA Test (T14).

7.2.2 Computations Subsystem

Jetson & Docker:

The Jetson and Docker unit hosts all key system packages and manages device drivers through a Docker-based ROS2 framework running on the NVIDIA Jetson Orin.

We built a custom Docker container building on the ZED-SDK wrapper image, and runs ROS2 Humble, micro-ROS, and several other necessary system packages and device drivers. We also configured the Jetson with a static IP for remote SSH access.

Localization:

The localization unit enables the rover to estimate its pose within the Moon Yard. It combines data from the Leica TS16 Total Station, VectorNav IMU, and wheel encoders, to support both global and local localization. We also calibrate the rover's yaw to ensure consistent orientation w.r.t the `map` frame.

The Leica TS16 Total Station acts as our "GPS on the Moon". While it provides reliable position data by tracking the prism on the rover, it is not Lunar accurate. In order to eliminate dependency on the Total Station, we implemented a new localization technique using the fisheye camera, which also provides position information of the rover, thereby mimicking a Sun-Star Tracker. Detailed explanation regarding the two implementations is provided in Appendix A.2.

Navigation:

The Navigation subsystem is responsible for identifying gradable craters, retrieving their coordinates, and generating feasible paths that allow the rover to reach these targets while avoiding obstacles.

A high-resolution 3D map of the Moon Yard is generated using a FARO laser scanner. From this map, craters are extracted and classified as either gradable or obstacles based on geometric thresholds defined during the Mapping the Moon Yard Test. These gradable crater centroids form the basis for the global navigation procedure.

For global planning, we employ a Lattice A* planner that incorporates yaw orientation as a third dimension, ensuring adherence to the rover's Ackermann steering constraints. Unlike differential-drive robots that can execute abrupt turns, the rover requires kinematically feasible, smooth curvature transitions. The planner generates paths that satisfy this constraint and incorporates several cost terms, including penalties for backward motion, maintaining counter-clockwise traversal, enforcing smooth heading evolution, and encouraging the rover to "hug" a reference spline.

This reference spline also referred to as the grading ring is derived from a polynomial fit passing through the centroids of all gradable craters. It provides a consistent, circular-like path around the Moon Yard, guiding the global navigation solution.

For control, we use a Pure Pursuit controller for real-time path following. During manipulation tasks (approaching crater sources, sinks, or backblade poses), the rover operates in configurations that are much closer to obstacles, where the global planner settings are too conservative. In these scenarios, we switch to a manipulation-specific configuration of the same Pure Pursuit controller, referred to as the "manipulation controller." A simplified A* planner, referred to as "manipulation planner," is used to generate these short manipulation paths, as they are usually nearly linear and may require controlled forward or backward motion.

Perception:

The perception subsystem is the eyes of the rover system. It uses the Zed 2i camera to detect crater information from the live feed and processes the information to provide goal poses for terrain manipulation.

We trained a YOLOv8-nano model on a custom dataset to detect craters in the Moonyard. Using the RGB and Depth data, the subsystem provides the crater centroid coordinates in the robot frame, as well as the crater diameter, which are used to compute the goal poses (Source->Sink->Source Backblade->Sink Backblade) for terrain manipulation.

The manipulation involves the following motions, executed by the manipulation planner and controller:

1. Reach Source Pose and actuate the tool down
2. Drive to sink pose (Forward Grading)
3. Actuate tool up and drive to Source Backblade
4. Actuate tool down
5. Drive backwards to Sink Backblade (Backblading)

The tool height is calculated based on depth data collected by the Zed Camera.

Validation:

The validation unit verifies whether the groomed crater satisfies M.P.6, which is to groom the trail to a maximum traversal slope of 5°. We use the ZED 2i stereo camera to evaluate the terrain by computing metrics such as mean elevation and elevation RMSE. We calibrated the camera to ensure that flat ground is perceived as having zero elevation and no tilt.

Behavior Executive Node (BEN):

The BEN is the node that executes the flow of the software system shown in Figure 10. It is a ROS2 node that maintains a finite state machine (FSM) that:

1. Switches rover MUX modes between `IDLE`, `FULL_AUTONOMY`, and `FULL_TELEOP`.
2. Plans and executes global navigation to crater sites.
3. Triggers crater perception and goal-pose generation.
4. Plans and executes manipulation/tool trajectories around craters.
5. Runs terrain validation and uses grading results to decide on second passes or next craters.
6. Provides `DEBUG` and `MANUAL_OVERRIDE` states for safe field debugging and teleoperation.

The FSM advances on a periodic timer callback, consuming odometry, crater detections, and planner/validation responses to drive the mission from `START_MISSION` to `END_MISSION` / `STOPPED`.

7.2.3 External Infrastructure Subsystem

The external infrastructure subsystem includes the off-board components required for localization and network communication between the rover and the operations terminal. It comprises the Leica TS16 Total Station, an NVIDIA TX2 relay chip, a LAN router, and the team's laptop, as shown in Figure 11. The total station continuously tracks the Leica

Figure 11: Left, Leica TS16 Total Station connected to TX2 Relay. Right, LAN Router and NVIDIA TX2 Relay

prism mounted on the mast of the rover and sends this data to the TX2 relay chip. The relay forwards these messages over the LAN to the Jetson onboard for localization. We validated the subsystem through the External Infrastructure Test (T05), ensuring reliable message transmission.

7.2.4 Mechanical Subsystem

The Dozer Assembly and the Wheel Assembly form the major Mechanical advancements made by us to the rover platform. The dozer assembly is the primary terrain manipulation component of the rover, designed to push and grade sand in the Moon Yard. The wheel assembly, shown in Figure 9, enables the rover's mobility by serving as the interface between the drivetrain and the lunar terrain.

7.2.5 Actuation Subsystem

The actuation subsystem is responsible for delivering motion to the rover's drivetrain and controlling the tool for sand manipulation. It includes four DC motors with encoders and a feedback-enabled linear actuator.

We selected high torque planetary gear motors and interfaced them with RoboClaw motor controllers. We also integrated the linear actuator, used for lifting and lowering the dozer blade, with the Actuonix LAC board and Arduino Due. A major issue that we faced during spring tests was that the pinion gears wore out or broke off of the motor shaft frequently. We addressed this by drilling a hole through the motor shaft and fixing a dowel pin to secure it. We also observed oscillations near the linear actuator's setpoint under load. We plan to further tune the PID or replace the component during the Fall.

7.2.6 Electrical Power Subsystem

The electrical subsystem manages power distribution across all components of the rover. It includes three 20V batteries, a DC-DC buck converter, a custom Power Distribution Board (PDB), and controllers for motors, actuators, and sensors. We designed a system-level circuit diagram and manufactured a compact electronics box.

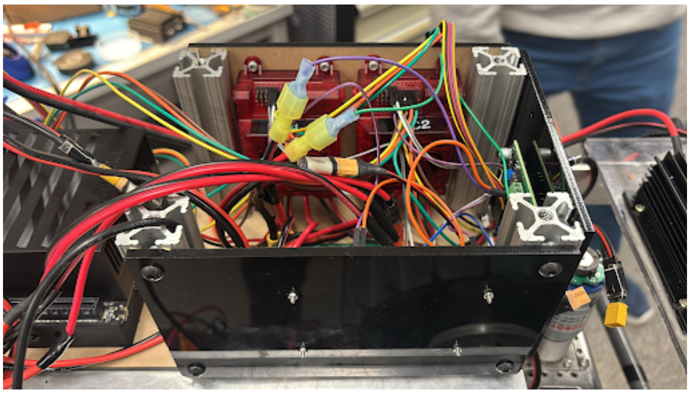


Figure 12: Electronics Box

As part of several hardware tests, we integrated and tested the PDB, which distributes power to all modules while meeting updated current and voltage demands. We also ensured that the electronics box minimizes space while maintaining accessibility and cooling, as shown in Figure 12. Full implementation details, design schematics, and hardware validation results are provided in the Appendix A.3.

7.3 Modeling, Analysis and Testing

7.3.1 Localization

Total Station Localization:

We configured and tuned the EKF parameters to fuse sensor inputs, which include the Leica TS16 Total Station, wheel encoders and IMU. We set up the frame transforms to ensure all sensor data aligns properly at the `base_link` frame. We also perform yaw calibration at startup to ensure consistent orientation data from IMU relative to the `map` frame.

Based on this, we conducted several tests where we analyzed pose stability and drift over time. We also analyzed sensor noise and measurement delays, and tuned EKF parameters to minimize odometry drift. During our Spring Validation Demonstration, we faced a minor frame offset issue each time the total station battery was swapped, leading to localization inaccuracies. In order to tackle this, we implemented a new resection method, which utilizes 3 fixed prism locations in order to fix the `map` frame, and is more accurate compared to the previous orientate-to-line method.

Fisheye Camera Localization:

The fisheye camera (the SkyCam) gives us the rover's position, just like the total station. In order to test the accuracy of the SkyCam, we created a test setup comprising of a cardboard box that holds the fisheye camera and the total station prism very close to each other. We then moved the setup manually within the Moon Yard and recorded estimates from the SkyCam as well as the total station, which acted as our ground truth. The results showed that the SkyCam estimates were much more continuous and reliable than the total station. Figure 13 shows us performing this test.

During this test, we found that any roll and pitch causes the pose to drift. In order to tackle this, we mounted the fisheye camera on a mechanical gimbal to account for any roll, pitch or yaw. Figure 14 shows the fisheye camera on a gimbal and mounted on the rover.

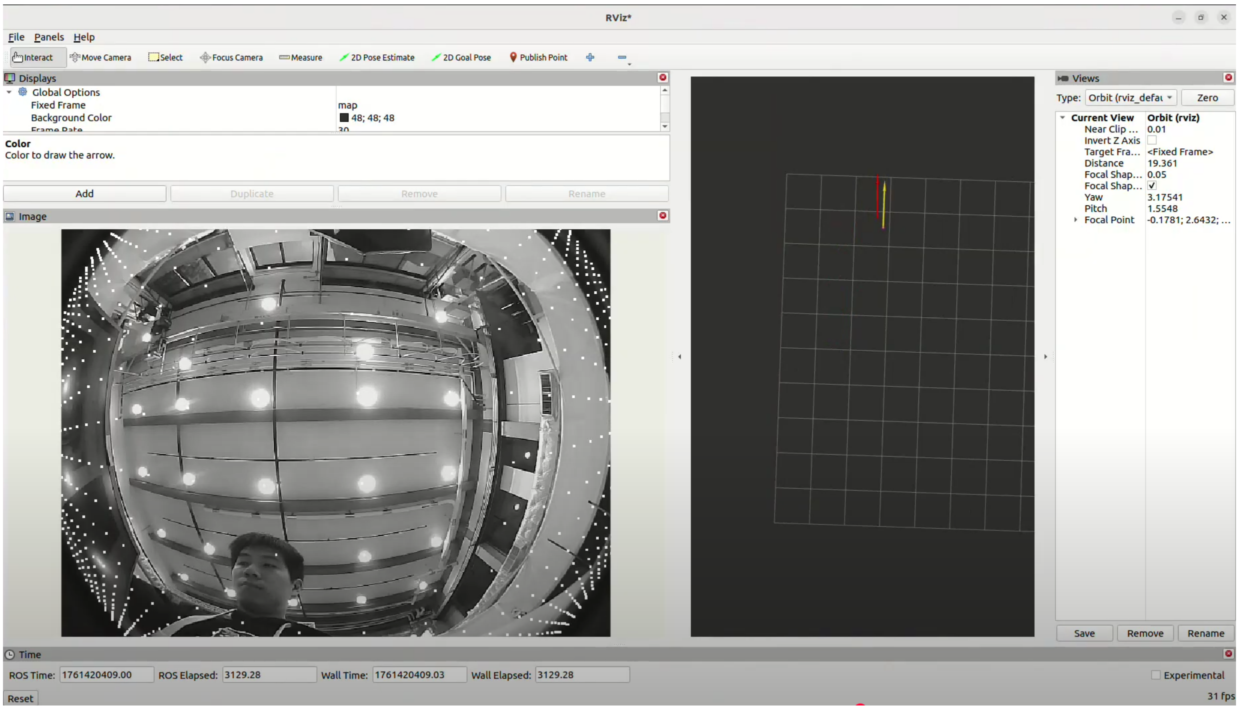


Figure 13: Fisheye Camera Localization Test with SkyCam estimate shown in Red and Total Station estimate shown in Yellow

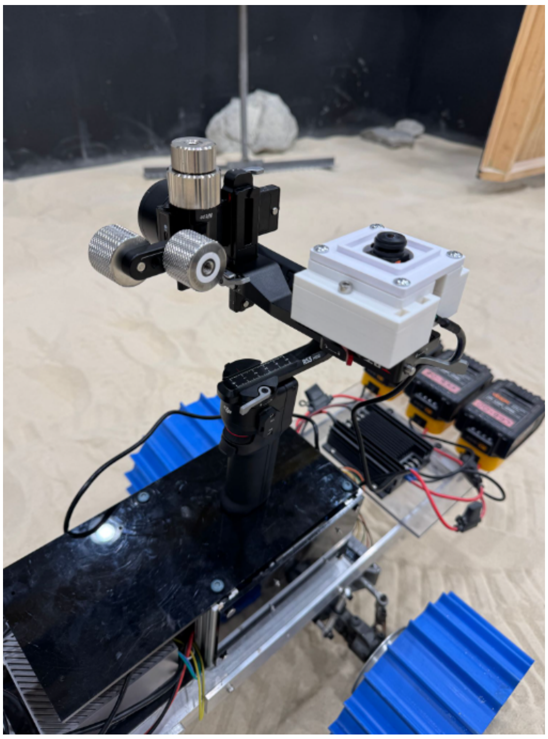


Figure 14: Fisheye Camera mounted on Mechanical Gimbal

7.3.2 Navigation

The navigation subsystem underwent extensive modeling, parameter tuning, and iterative testing to ensure reliable path planning and smooth Ackermann-feasible motion in the Moon Yard. Both the global Lattice A* planner and the Pure Pursuit controller were refined through repeated trials across varying crater sizes, obstacle layouts, and terrain conditions.

The global planner uses a reference spline generated from a polynomial fit through the centroids of all gradable craters. Planner parameters were tuned to keep trajectories close to this spline, avoid ungradable craters, and maintain curvature limits appropriate for Ackermann steering. Instances of out-of-bounds paths or trajectories intersecting ungradable craters informed further adjustment of planner cost weights, resulting in stable and repeatable plans.

Similarly, the Pure Pursuit controller parameters, primarily lookahead distance, goal tolerance, and linear velocity—were tuned through on-robot experiments to achieve smooth, non-oscillatory turning without swirly motion.

Tables 7 and 8 summarize the final navigation performance. Table 7 reports planned path deviations between 21.47% and 24.34% (mean 22.89%), satisfying M.P.1 ($\leq$25%). Table 8 shows actual path-following deviations ranging from 0.09% to 12.98% (mean 7.01%), meeting M.P.2 ($\leq$10%).

Table 6: Planned Path Deviation from Reference Latitude

Radius of reference latitude = 2.33 m, Reference Latitude length = 14.66 m

Planned Path Length (m)	Deviation (m)	Deviation (%)
11.09	3.57	24.34%
11.51	3.15	21.47%
11.11	3.55	24.20%
11.51	3.15	21.47%
11.29	3.37	22.97%

Least: 21.47% **Max:** 24.30% **Mean:** 22.89%

Table 7: Actual Path Followed Deviation

Radius of reference latitude = 2.33 m, Reference Latitude length = 14.66 m

Planned Path (m)	Actual Traversed Path (m)	Deviation (m)	Deviation (%)
11.09	9.65	1.44	12.98%
11.51	12.97	1.46	12.68%
11.11	11.42	0.31	2.79%
11.51	10.76	0.75	6.52%
11.29	11.28	0.01	0.09%

Least: 0.09% **Max:** 12.98% **Mean:** 7.01%

The Autonomous Navigation Validation Test (T02, T03, T04 and T06) confirmed that the rover can consistently reach target craters while avoiding obstacles and maintaining Ackermann-feasible motion.

7.3.3 Perception

This was a new subsystem, so modeling, analysis, and testing began early. We selected the YOLOv8-nano model for its lightweight architecture and real-time performance on edge devices. To build the dataset, we created multiple craters and captured hundreds of images under varied lighting and viewing angles, then manually labeled crater bounding boxes in Roboflow. The dataset was split into train/val/test sets (fig. 15).

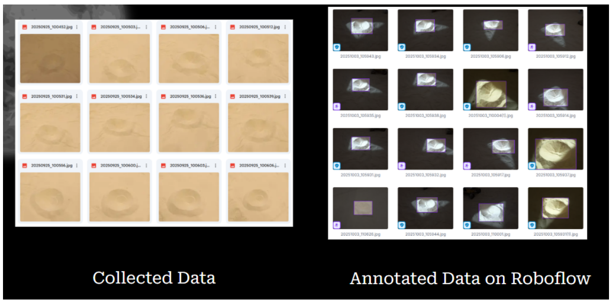


Figure 15: Crater Dataset

After retraining YOLO on this data, we tested it on recorded videos, added more data, and tuned hyperparameters to improve performance. The final model was deployed on the Jetson Orin, running live inference on the Zed 2i camera feed, achieving detection confidence between 55–95%.

We then implemented crater geometry extraction and validated the full detection-and-measurement pipeline across different crater shapes (fig. 16). Initial issues like camera feed latency during YOLO inference and small logic bugs in geometry extraction were resolved using CUDA acceleration and iterative testing.

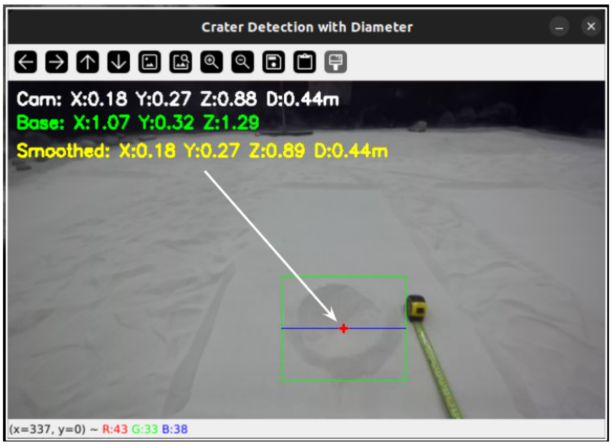
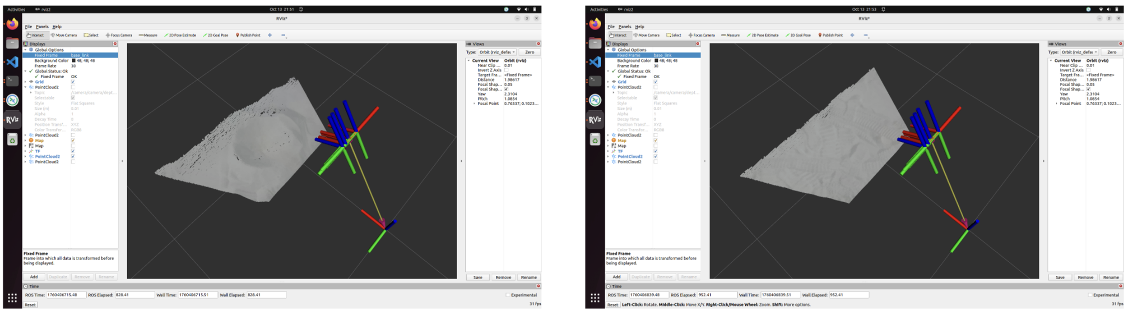


Figure 16: Crater Geometry Extraction

Finally, crater diameter and coordinates were published in the robot frame, and robot poses were generated based on crater size and the rover's current position. The four key poses (Source → Sink → Source backblade → Sink backblade) were tuned and integrated with the navigation stack.

7.3.4 Validation

To ensure that the validation unit meets the required specifications, the ZED 2i stereo camera was first modeled and calibrated against the base link of the rover, ensuring its location and orientation are known exactly. This is published via a static transform TF topic, so the camera feed is transformed to the correct base link frame. Once the validation unit is integrated and deployed, quality assurance and testing are performed to ensure the unit's accuracy. In particular, the gradient estimate from the validation unit was compared against a 15-degree crater and flat terrain. The unit returned an estimate of 14.78 degrees (grading = false) and 1.44 degrees (grading = true), respectively (see Figure 17). The unit's robustness to different terrains (crater, flat ground, smoothed sand, facing wall, facing rock, etc) was also analyzed to ensure its robustness across scenes that it is expected to encounter.



Validation on 15 Degree Crater
Grading Success = False, Slope = 14.78 deg

Validation on Flat Terrain
Grading Success = True, Slope = 1.44 deg

Figure 17: Validation Unit QA Testing

7.3.5 Mechanical

Dozer Assembly:

It consists of a front-mounted blade, support arms, actuator mounts, and a linear actuator that receives commands from the navigation stack during the manipulation state. Figure 9 shows the custom dozer blade mounted on the front.

We conducted extensive research on standard dozing and grading tools to design the blade, arms, and actuator yoke. We machined all the components from lightweight aluminum for strength and durability. We also tested multiple actuator configurations for optimal force transfer. We demonstrated successful autonomous dozing and backblading during both Spring and Fall Validation Demos.

Wheel Assembly:

The wheels are 3D printed using PLA and are mounted directly to the drivetrain. We went through several iterations and tested them in the Moon Yard, with design refinements made based on visual wear, deformation, and observed rover performance. The current wheel design successfully provides sufficient traction for pushing sand and allows for steering on loose terrain, as demonstrated in the Spring nd Fall Validation Demos (SVD).

7.4 FVD and FVD Encore Performance Evaluation

We structured our performance evaluation around 3 main questions: What went well? What didn't go well? What needs to improve?. Furthermore, based on our FVD demonstration, we assessed how well we were able to demonstrate our targeted requirements and identified the necessary improvements. Finally, we scored our entire spring project at a high level.

7.4.1 Self-Score

At a high level, this is how we scored our system,

1. **Hardware:** 10/10
2. **Software:** 9/10
3. **Presentation/Management:** 9/10
4. **Overall: 9/10**

7.4.2 Qualitative Evaluation

What went well?

1. Mechanical Design: The drive system and tool assembly were sturdy and well-built.
2. Electronics Box Design: The design is compact, reliable, and integrates the PCB. It also satisfies D.N.2 by providing a more finished look to the ROADSTER.
3. Perception: The perception pipeline worked well on all types of craters in the MoonYard and outputted good manipulation poses.
4. Navigation Planner: The navigation stack found the shortest paths to the goal while avoiding obstacles and executed the manipulation tasks well.

5. Finite State Machine: The behavior tree worked well at 2Hz and switched between the various states well.
6. SkyCam: The SkyCam was able to localize the rover position with accuracy comparable to the Total Station and published at a faster frequency.

What did not go well?

1. Reliability of Networking: In the morning, on Encore, our local network, which is used to connect to the Jetson, stopped working. We do not know why this happened, and the network was not established reliably until hours later.
2. Validation: Due to errors in the rover position and smoothing filters, the validation outputted SUCCESS during Encore despite the manipulation not meeting our performance metrics.

What needs to improve?

1. Pre-demo setup methodology: Our current pre-demo setup is convoluted and time-consuming, as it involves mapping, calibration, and setting up external infrastructure. This needs to be improved to allow us to test more frequently and efficiently.
2. Reliance on tuning: Our perception and navigation stacks require manual tuning of parameters to get adequate performance. Autonomy can be enhanced by incorporating additional sensors and implementing more effective startup methods.

7.4.3 Evaluation of Requirements

Table 8: Requirement Evaluation

Req.	Description	Status	Score
M.P.1	Cumulative Deviation from Latitude	**Demonstrated** - Satisfies requirement and rover plans path with average deviation of 22.89% from reference latitude.	10/10
M.P.2	Navigation Accuracy	**Demonstrated** - Satisfies verification criteria and rover follows planned path with average deviation of 7.01 %.	10/10
M.P.3	Contact Pressure	**Demonstrated** - Contact Pressure of the ROADSTER on the Moon - 1.37kPa (0.199 psi).	10/10
M.P.4	Crater Avoidance	**Demonstrated** - Avoided the large crater in the MoonYard in all test runs.	10/10
M.P.5	Crater Grooming	**Demonstrated** - Groomed craters in both autonomous and tele-operation modes. We built a strong, capable and specialized machine for crater grooming.	10/10
M.N.1	Weight	**Achieved** - The rover weighs 23.8kg.	10/10
M.N.2	Cost	**Achieved** - The cost of the project was $4,995.	10/10
M.N.3	Computing Capacity	**Achieved** - The onboard computer can run all tasks	10/10
D.N.1	Technological Extensibility	**Achieved** - We have maintained an Engineering Wiki and extensively documented our entire design process.	10/10
D.N.2	Aesthetics	**Demonstrated** - Sponsor requirement.	10/10
D.N.3	Modularity	**Achieved** - The tool assembly can be mounted and re-mounted easily	10/10
D.N.4	Repeatability	**Achieved** - Several test runs without maintenance.	10/10

7.5 Strong/Weak Points

7.5.1 Strong Points

1. **Robust Tool Subsystem:** The dozer assembly is well-built and effective.
2. **Compact E-Box:** The new E-box is an efficient design with an accessible casing that integrates the PDB.
3. **Wheels:** The custom wheels reliably enable superior traversability and increased drawbar pull.
4. **Power Distribution Board:** The PDB is a well-organized circuit that reliably protects and powers all subsystems.
5. **Management and Presentation:** The fall semester was well-managed and on schedule. Our presentations during FVD and Encore were well received.
6. **Software:** All software stacks - perception, validation, navigation, and BEN functioned effectively and reliably.
7. **Software Integration:** The behaviour tree, perception, validation, and navigation stack were integrated and functioned seamlessly.
8. **SkyCam:** The SkyCam was able to localize the rover position with accuracy comparable to the Total Station and published at a faster frequency.

7.5.2 Weak Points

1. **External Infrastructure:** The local network did not work reliably during Encore.
2. **Hardware Reliability:** A lot of our system testing was delayed throughout the semester due to hardware connection issues. While we were able to solve this by demo day, we were unable to test as much as we had wanted to.
3. **MoonYard Logistics:** As CMU policies regarding MoonYard activities were changing, we faced constant issues with access and testing throughout the semester. Since these were new, we couldn't plan for them, leading to delays and reduced testing time.

8 Project Management

8.1 Schedule

The capability milestones planned at the beginning of the Fall semester are tabulated in Table 9. Bi-weekly reviews are aligned with Progress Reviews to enhance progress tracking and support the timely achievement of critical external and internal milestones. We also introduced the Objectives & Key Results (OKR) tracker to better align owners and deliverables. The OKR tracker also allowed us to identify earlier what potential risks could arise and develop a mitigation plan should the risk be realized.

Table 9: Fall Semester Subsystem Capability Milestones

Date	Event	Capability Milestones	Tests	Requirements
09/10	PR7	Hardware and software **refinement**	Validate **hardware upgrades**, **software fixes**, and **system stability** improvements	M.F.5, M.F.9, M.N.3
09/24	PR8	**Validation stack** setup. **Navigation** tuning.	Verify **smooth navigation** through path execution tests	M.F.2, M.F.3, M.F.4, M.F.8, M.F.9
10/08	PR9	Autonomous grading of **multiple craters**	Verify **autonomous grading performance** across multiple craters	M.F.2, M.F.3, M.F.4, M.F.5, M.F.6, M.F.7, M.F.9
10/29	PR10	SkyCam-based localization for **improved global positioning**	Test SkyCam-based localization by checking rover's ability to **self-localize accurately with/without external infrastructure**	M.F.3
11/12	PR11	**Full system integration Quality assurance testing**	Check all **subsystems and units are functioning correctly**	M.F.2–M.F.9
11/17 & 11/24	PR12 (FVD and Encore)	**Final system demonstration** involving autonomous grading of multiple craters	Demonstrate **full autonomous operation** by detecting, avoiding ungradable craters, and grading multiple suitable craters according to mission specs	M.F.1–M.F.9, M.P.1–M.P.9

We were generally on schedule for our development and capability milestones over the Fall semester. A major scheduling failure that we encountered was that the rover's hardware was inoperable for the majority of PR8 and PR9. This meant that much of our onboard testing was delayed until the rover was fixed. As such, the validation, navigation, and perception units could not be integrated on board the rover. To partially mitigate this blocker, our team resorted to using static tests and simulations to test unit functionality. This proved to be successful in partially limiting the impact of the hardware blocker.

8.2 Budget

As of the Final Review at the end of the Fall semester, $4,992.71 of the $5,000 MRSD budget has been spent. Including items provided to us free of charge by our supervisor and items inherited from the Crater Grader MRSD team from 2022, our total budget spent is approximately $8,062.71. This figure is not incorporated into the $5,000 MRSD budget allocated to us. Table 10 outlines a refined parts list of major big-ticket item purchases. Figure 18 shows item purchases under the MRSD budget segregated by functionality. Our full parts list includes over 70 entries and is available on our website:

https://mrsdprojects.ri.cmu.edu/2025teami/parts-list/

Table 10: Budget Statement with Big-Ticket Purchases

Part Name	Description	Unit	Quantity	Total
NVIDIA Jetson	Computing board	$800	2	$1600*
VN-100 IMU	IMU	$800	1	$800*
Planetary Gear Motor	Motor	$60	4	$240*
RoboClaw 2x30A	Motor Controller	$135	2	$270*
ZED 2i	Stereo Camera	$562	1	$562
Planetary Gear Motor	Motor	$60	10	$600
Linear Actuator	Actuator	$97	3	$290
DC/DC Power Converter	Power Converter	$316	1	$316
VN-100T-CR	IMU	$1300	1	$1300
			MRSD Budget Total:	**$4,992.71**
			Grand Total:	**$8,062.71**
				*Excluded from MRSD budget

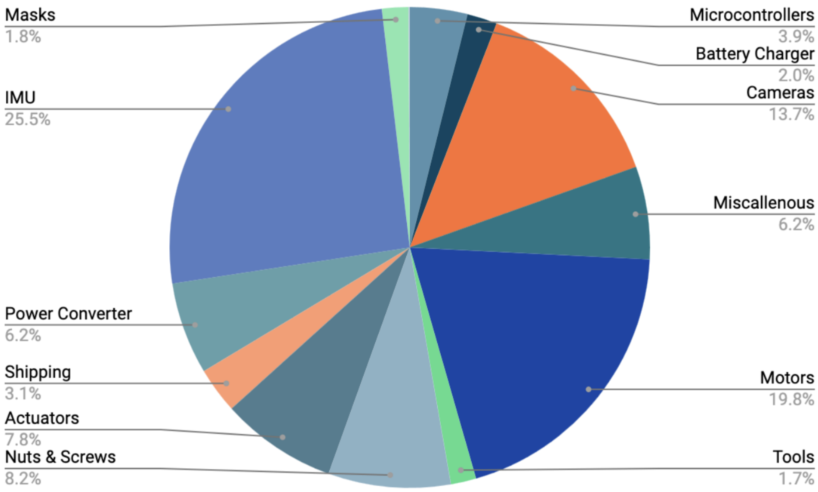


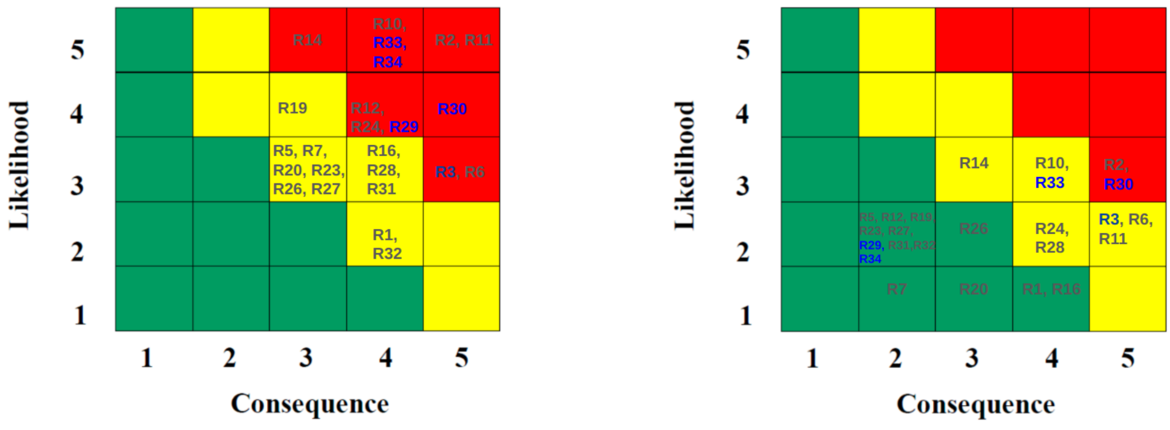
Figure 18: Pie Chart of Budget Expenditures

Overall, our budgeting process has been successful. We remained under-budget for the entire project. The final $500 of our budget was reserved for spare part purchases of rover critical items such as motors, linear actuators, and cameras. This is the reason why we only have around $7 remaining at the end. A budgeting failure was the purchase of the VN-100T-CR IMU. We did not thoroughly look through the MRSD cage inventory list before making the purchase. We found an identical model from the MRSD cage afterwards and could have spared the $1300 to be used for other purchases.

8.3 Risk Management

At the beginning of the Fall semester, the top five highest priority risks were identified. These are tabulated in the subsections below. The full risk management table is provided in Appendix A.5. For each listed risk, we include a preliminary mitigation plan along with any actions taken to address or reduce the associated threat.

Figure 19 shows the Likelihood-Consequence tables for all risks identified so far, with (a) representing the unmitigated risks and (b) the risks after mitigation actions. The highlighted entries correspond to the top five risks actively tracked and addressed in the project.



(a) Unmitigated Risks (b) Mitigated Risks

Figure 19: Likelihood-Consequence Tables

8.3.1 Integration Issues between Subsystems

This risk had an unmitigated **likelihood of 3 and consequence of 5**. Due to the complexity of the system, individual subsystems may work correctly but fail to communicate properly when integrated, leading to schedule delays, requirement changes, and potential demo failure.

To address this, we adopted a common ROS2-based communication framework, performed continuous unit testing, and integrated subsystems incrementally. These actions were implemented successfully in Fall, reducing the **likelihood to 2 while the consequence remains at 5**.

8.3.2 Access to FRC Workshop

This risk had an unmitigated **likelihood and consequence of 4**. Since access to the FRC Workshop was essential for hardware fabrication and repairs, lack of access, especially in the absence of key personnel, posed a serious threat to meeting hardware deadlines.

To mitigate this, the team successfully utilized alternative fabrication labs on campus and coordinated with professor Red Whittaker, Tim and other stakeholders to gain temporary access to the FRC Workshop. These actions were implemented early on in the semester and effectively reduced both the **likelihood and consequence to 2**.

8.3.3 No spares available

This is a critical risk with an unmitigated **likelihood of 4 and consequence of 5**. It arose after discovering that one of the rear axles had broken and disengaged from the driveline. As the rover model is discontinued, spare parts are no longer available, putting the entire project at risk.

With support from Professor Red and Chuck, the team located a twin rover and salvaged compatible parts. Additional mitigations, including stocking similar components and maintaining mechanical subsystems, have reduced the **likelihood to 3, though the consequence remains high at 5**. This remains a top concern.

8.3.4 Localization frame shifts after total station battery swap

This risk was identified after the PDR during system testing for the Spring Validation Demo (SVD). It had an unmitigated **likelihood of 5 and consequence of 4**. Battery replacement in the total station introduced small frame shifts, resulting in localization inaccuracies that affected navigation and posed a risk of missing craters during grading.

To mitigate this, the team plans to use a resection method with three known prism locations instead of orientate-to-line and explore alternatives like SkyCam, reducing the **likelihood to 3 while the consequence remains 4** due to the importance of accurate localization.

8.3.5 Arduino requires reset before teleoperation

This risk was identified after the PDR during system testing for the Spring Validation Demo (SVD), with an initial **likelihood of 5 and consequence of 4**. The Arduino required frequent manual resets when switching between autonomy and teleoperation, impacting setup time and operational readiness.

Mitigations such as correcting Jetson USB permissions, using a USB 3.0 connection, and fixing ROS node frequency mismatches stabilized the connection, eliminating manual resets and reducing **both likelihood and consequence to 2**.

8.3.6 Evaluation

Overall, the risk planning and mitigation of the team was very successful. Potential risks are identified early on in the development phase and shared with all team members. Critical risks had multiple mitigation plans in place, and major concerns such as spare part availability and hardware degradation were addressed by purchasing critical spares ahead of the Final Validation Demo.

Moreover, the team used an OKR framework to track progress and assign clear owners to each unit, enabling early communication of blockers and proactive risk mitigation. The team identified over 20 risks before major rover development, enabling robust mitigation plans to be put in place early. We also kept our issues log up to date throughout the development cycle and reviewed the log every week to add new issues and resolve old ones. Our issues log can be view here: `https://mrsdprojects.ri.cmu.edu/2025teami/issues-log/`

A key improvement area is hardware risk identification. The team assumed the inherited CraterGrader rover was fully operational with minimal modifications, which proved incorrect due to significant wear and required replacements. Earlier identification of hardware risks and component-level mitigation plans would have improved risk management.

9 Conclusions

9.1 Lessons Learned

Always prepare for unforeseen delays
We aimed to finish subsystems early to reserve time for integration and testing, yet we still missed our target by several weeks due to unforeseen blockers like mechatronic wear-and-tear. We learned that accurate scheduling requires not just optimistic targets, but realistic estimates that factor in generous buffer time for revising tasks and addressing extreme edge cases.

Communication is Key
Since the inception of this project, our team had decided to meet with our sponsor every week. This turned out to be a very critical decision and a very fruitful one, as we quickly understood that our sponsor's insights helped us plan the upcoming weeks better and discuss issues and risks before they became blockers. We received valuable feedback from the team members and the sponsor through these meetings.
Along with this, our regular stand-up meetings and open communication through multiple channels kept us updated with the progress of the team and the project, and helped us greatly in picking up slack and redistributing tasks during overloading scenarios.

Hardware is indeed, HARD!
At the end of the Spring Semester, we had a well-designed working mechatornic system in place that was tested several times. However, since the beginning of the Fall semester, we faced countless hardware issues, some of them being poor wiring, actuator failure, steering motor failure, driveline failure, etc.
Even though we had decided to focus the team's time on software and integration this semester, we ended up spending 60% of the semester fixing hardware issues, ultimately replacing all the parts. This taught us that never underestimate hardware, be doubly sure than sorry, and always have spares.

9.2 Future Work

After debriefing with the team members and sponsors, we suggest the following advancements to the system:

- **Multi-DOF tool**
  This feature will greatly help in tackling the crater from different directions. This can be achieved by adding different DOFs, like yaw and sliding, to the tool and actuation subsystems.

- **Robust Localization**
  The Total Station lacked precision and speed. The Sky-Cam Localization lacked robustness and repeatability. Moving forward, this is a crucial system (and a pain point) that can be targeted to improve.

10 References

[1] Teng Hu et al. “Population of Degrading Small Impact Craters in the Chang’E-4 Landing Area Using Descent and Ground Images”. In: *Remote Sensing* 14.15 (2022). ISSN: 2072-4292. DOI: 10.3390/rs14153608. URL: https://www.mdpi.com/2072-4292/14/15/3608.

[2] P. Muñoz, P. Bellutta, and M. D. R-Moreno. “Proposing new path-planning metrics for operating rovers on Mars”. In: *Scientific Reports* 13 (2023), p. 22256. DOI: 10.1038/s41598-023-49144-8. URL: https://doi.org/10.1038/s41598-023-49144-8.

[3] Laurent Sibille et al. *Lunar Regolith Simulant Materials: Recommendations for Standardization, Production, and Usage*. Tech. rep. NASA/TP-2006-214605. NASA Marshall Space Flight Center, 2006. URL: https://ntrs.nasa.gov/api/citations/20060051776/downloads/20060051776.pdf.

A Appendices

A.1 Subsystem Completion Status

Table 11: Subsystem Completion Status

Subsystem	**Completion**
Sensors	100%
Computations	100%
Jetson & Docker	100%
Localization	100%
Navigation	100%
FSM Planner	100%
Validation	100%
External Infrastructure	100%
Mechanical	100%
Dozer Assembly	100%
Wheel Assembly	100%
Actuation	100%
Electrical Power	100%

A.2 Localization Unit Implementation Details

A.2.1 EKF-based Sensor Fusion

Localization is achieved using an Extended Kalman Filter (EKF) implemented via the ROS2 `robot_localization` package. We implement two EKF nodes, as shown in Figure 21:

1. **Local EKF**: Fuses data from the VectorNav IMU and wheel encoders.
2. **Global EKF**: Fuses the data from the VectorNav IMU, wheel encoders along with absolute position data from the Leica TS16 Total Station or the Fisheye camera.

We require both global and local localization nodes because they complement each other's weaknesses. The local localization node provides fast and smooth updates on the rover's motion, and is good for real-time control. However, it drifts over time because it is dead-reckoning. The global localization node helps correct any long-term drift and keeps the rover anchored to the `map` frame in the Moon Yard. However, it is too sluggish for steering control. By using both nodes together, we get the benefits of smooth short-term motion tracking and long-term position accuracy.

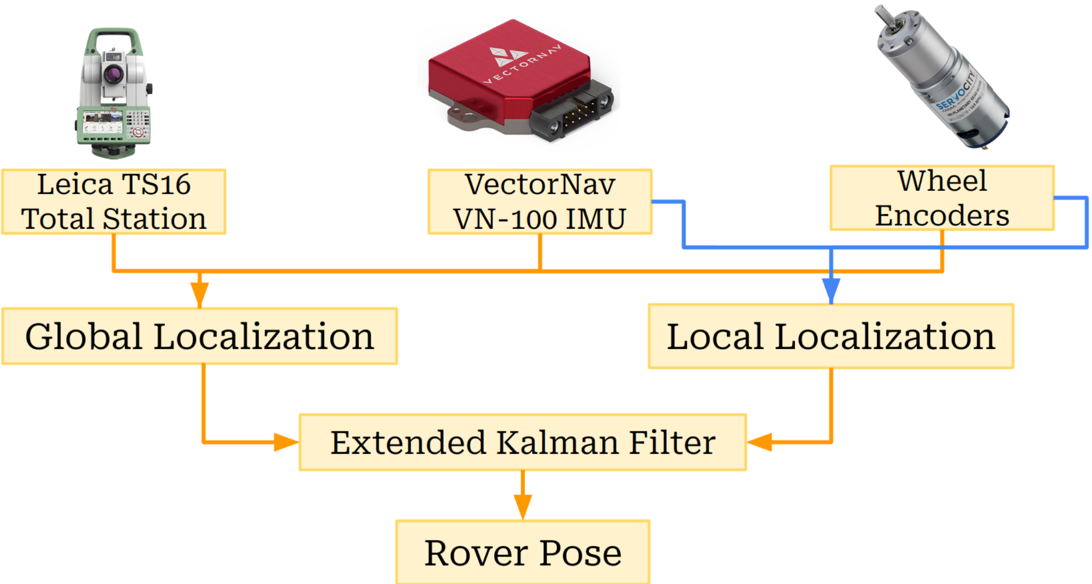


Figure 20: Localization Method using Total Station

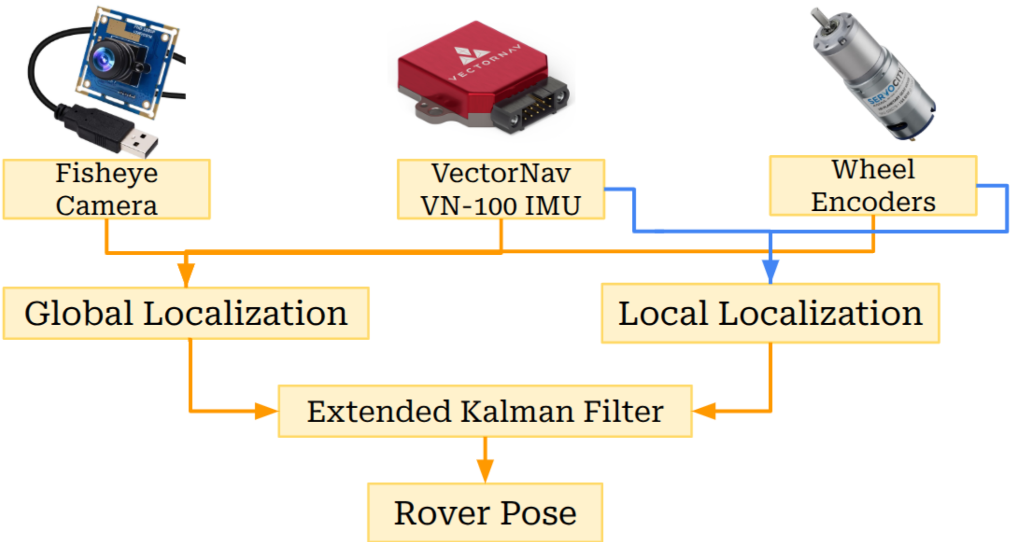


Figure 21: Localization Method using Fisheye Camera

A.2.2 Fisheye Camera

The Leica TS16 Total Station acts as the "GPS on the Moon". While it does provide reliable position information of the rover, it is not Lunar accurate. In order to tackle this, we use a fisheye camera to provide us with the position data. The fisheye camera looks upward at the ceiling of the Moon Yard, which is our test environment. It then finds bright pixels (lights on the ceiling), regresses points to this grid of bright pixels using Levenberg-Marquardt optimization, calculates pixel offset and re-projects back to 3D from the 2D camera coordinates using ceiling height, in order to obtain the rover's position. In multiple tests, we found that the localization information obtained after fusing data from the fisheye camera was much more continuous, reliable and accurate, than with the total station data.

A.2.3 Coordinate Frame Alignment

The following coordinate frames are defined and broadcasted to ensure proper alignment of all sensor inputs:

- `map` : This is the global frame with origin at the corner of the Moon Yard.
- `odom` : This is the local frame origin that offers a continuous homogeneous transform to the `base_link` frame.
- `base_link` : This is the frame of the rover at its Center of Mass.
- `total_station_prism` : This is the frame of the Leica Prism mounted on the mast of the rover.
- `imu_link` : This is the frame of the IMU.

A.2.4 Crater Identification and Classification

We identified gradable craters from the processed map based on geometric features. Each crater was classified by:

- Diameter: Only craters within a 0.5 m diameter, as per our performance requirements, were marked as gradable.
- Depth: Only craters within 0.1 m depth were marked as gradable.

A.3 Electrical Subsystem Implementation Details

- We use three 20V batteries to power the system.
- A buck converter steps down voltage to 12V for logic-level components.
- Power is routed through an Emergency Stop, wireless switch (FOB), and monitored using two battery sensors before reaching the custom Power Distribution Board (PDB).
- The PDB supports connection to RoboClaw motor controllers, actuator controllers, Jetson, Arduino Due, and other peripherals.

Figure 22 shows the detailed circuit schematic. We modeled the PDB, shown in Figure 23, with protective features, including over-current, reverse-voltage, and indicator LEDs. We also evaluated the voltage and current demands of all subsystems to ensure load balancing and stable PDB output.

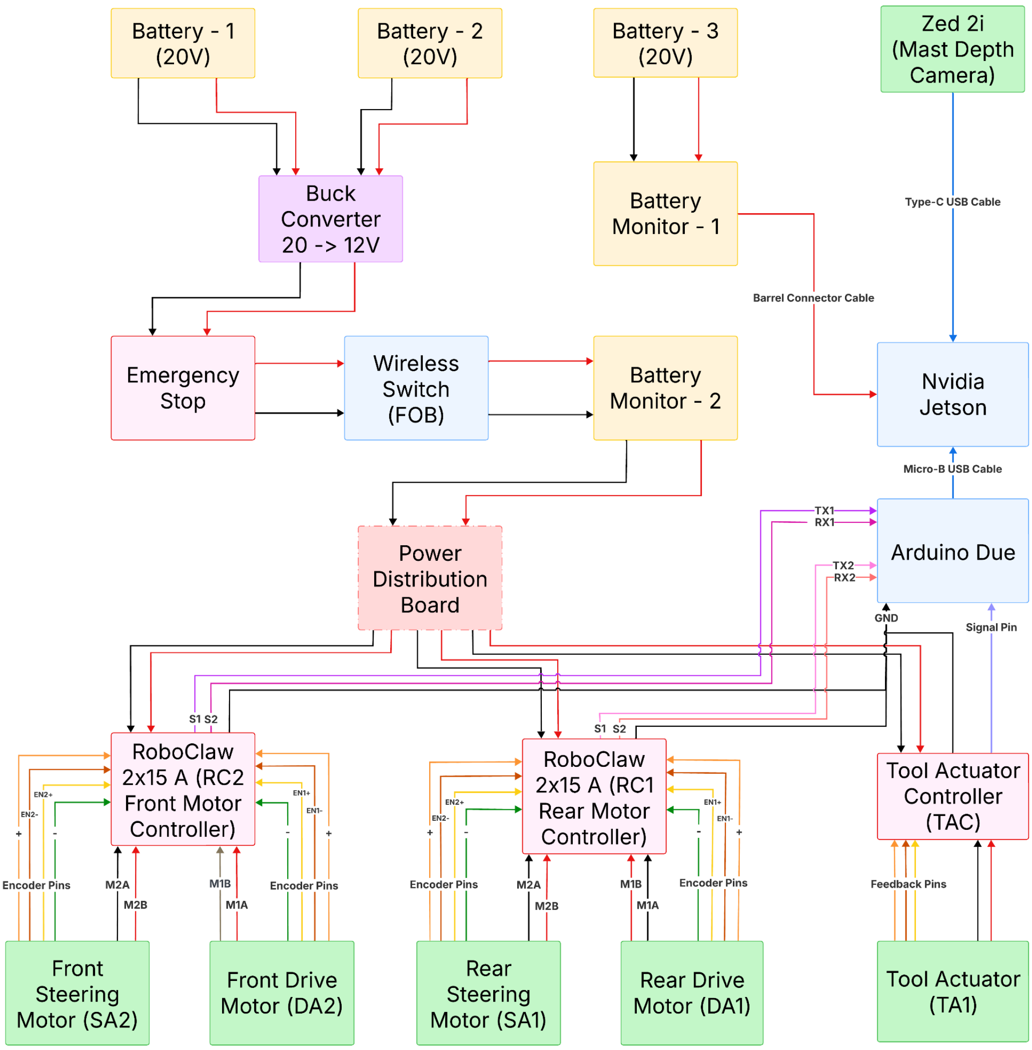


Figure 22: Circuit Schematic

Figure 23: Power Distribution Board

A.4 Subsystem-Wise Detailed WBS

A.4.1 Mechatronics

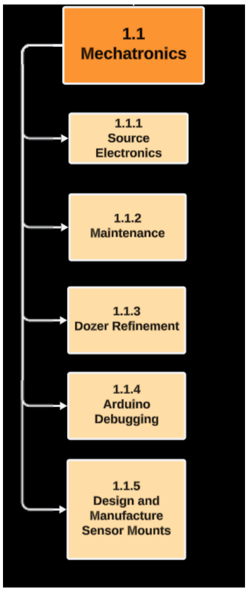


Figure 24: Level 4 WBS for Mechatronics Subsystem

Figure 24 shows the Level 4 Work Breakdown Structure for the Mechatronics Subsystem. The tasks shown are:

- Sourcing electronics and electrical connections like actuators, junction connectors, etc.
- Maintenance task will include QA checks of all mechanical and electronic connections
- Dozer refinement task includes fixing the jittering of the dozer actuation.
- Arduino debugging task includes fixing the Arduino reset issue.
- New mounts will be designed and manufactured for new sensors like Zed 2i camera and the fish-eye camera, and the gimbal.

A.4.2 Computations

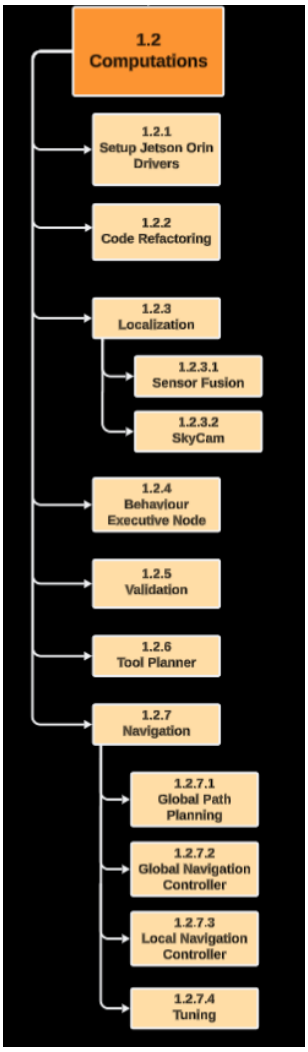


Figure 25: Level 4 WBS for Computations Systems

Figure 25 shows the Level 4 Work Breakdown Structure for the Computations Subsystem. The tasks shown are:

- The first tasks are initial setup tasks involving research, setting up drivers, and docker on the new Jetson Orin.
- The code refactoring task includes adding file headers, documenting the code, and refactoring it to maintain clean coding practices.
- Localization has tasks of sensor fusion tuning and implementing the SkyCam based localization.
- Behaviour Executive Node is the main Finite State Machine, which will integrate all other nodes and susbsystems.
- Validation task includes the implementation of validating the crater grooming using gradients, calculated using the depth information from the Zed 2i camera.
- Tool Planning will include using Perception to determine Crater Geometry and extract poses for terrain manipulation.

- Navigation task will include developing custom planners and controller for global and local navigation

A.4.3 Sense

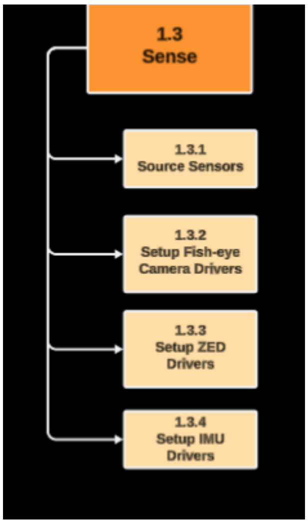


Figure 26: Level 4 WBS for Sensing Subsystem

Figure 26 shows the Level 4 Work Breakdown Structure for the Sense Subsystem. The tasks shown are:

- This subsystem involves sourcing our sensor stack - Encoders, ZED 2i Depth Camera, IMU, Fish-eye Camera and Survey LiDAR.
- The tasks also include setting up the required drivers and obtaining data using the desired communication method.

A.4.4 External Infrastructure

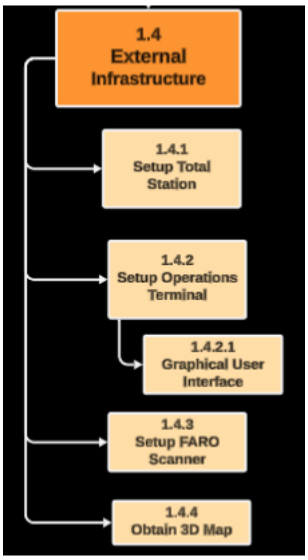


Figure 27: Level 4 WBS for External Infrastructure

Figure 27 shows the Level 4 Work Breakdown Structure for the External Infrastructure Subsystem. This consists of all components of the project that are not on the rover. The tasks shown are:

- Sourcing and Setting up total station will require training with FRC Technicians. It will involve calibrating and tracking the rover's postion during the mission.
- The Operations Terminal consists of the team's laptops, which emulate the Moon Station. This station is used to monitor the mission and verify validation conditions through a Foxglove-based graphical user interface.
- Obtain 3D Map involves obtaining a map of the Moon Yard using the Survey LiDAR. This map will be used to plan the trail path to be groomed by the ROADSTER.

A.4.5 Integration and Testing

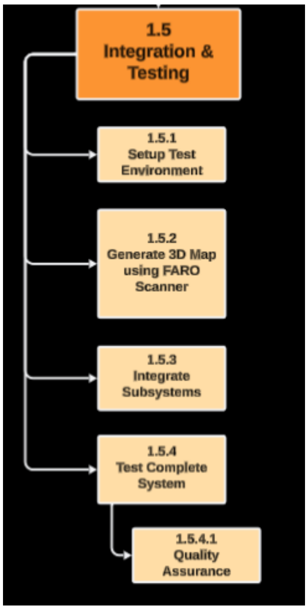


Figure 28: Level 4 WBS for Integration and Testing

Figure 28 shows the Level 4 Work Breakdown Structure for the Integration and Testing Subsystem. The tasks shown are:

- All tasks involve ensuring that all subsystems can work together as a cohesive unit.
- The team will setup concrete testing plans with varying environments to test the ROADSTER
- Set up the test environment and generate a map using the FARO laser scanner, followed by full system testing and quality assurance to improve reliability and performance.

A.4.6 Management

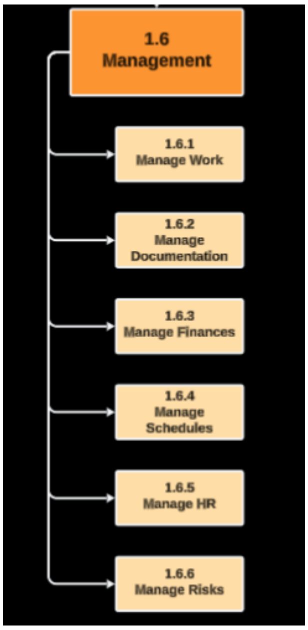


Figure 29: Level 4 WBS for Management

Figure 29 shows the Level 4 Work Breakdown Structure for Management. The Program Manager will be responsible for all tasks in the Management Subsystem. The tasks shown are:

- Manage Work involves tracking progress of the assigned tasks to each member.
- Manage Documentation will cover all reports, presentations and process documentations through the work period.
- Manage Finances involves managing purchases and allocating adequate budget to every subsystem.
- Manage Schedules involves tracking the overall timeline of the project and ensuring that the team follows the planned schedule.
- Manage HR is a task where the team will allocate some time every week to uplift team morale and maintain motivation.
- Manage Risks involves identifying and mitigating any potential risks in the project.

A.5 Identified Risks

The following risks are identified during our preliminary risk management analysis. Potential mitigation actions are put in place to minimize these risks.

Risk ID	Risk Title	Risk Owner	Date Added	Date Updated
R1	PRL Testbed Scheduling	Ankit	11/27/2024	11/27/2024
Description			**Original Likelihood**	**Original Consequence**
PRL Testbed unavailable due to scheduling conflicts with other high priority projects			**2**	**4**
Consequence			**Mitigated Likelihood**	**Mitigated Consequence**
No testbed available for testing and/or SVD			**1**	**4**
Risk Reduction Plan Summary			**Risk Type:**	Scheduling
Action/Milestone		**Success Criteria**	**Date Planned**	**Date Implemented**
Devise and discuss a testing and demo plan with Red and other stakeholders of the PRL testbed beforehand and reserve slots		Successfully reserve slots for using the PRL Testbed	11/30/2024	
Reach out to external testing facilities like Astrobotic or CAT for a backup testing facility		Communicate and discuss possibility of using other testbeds if PRL falls through		
Schedule tests at night		Schedule tests at off-hours to avoid clashes		
Comments				

Figure 30: PRL Testbed Scheduling

Risk ID	Risk Title	Risk Owner	Date Added	Date Updated
R2	Excavator and grader tool planner takes longer than expected to deliver	Simson	11/27/2024	11/27/2024
Description			**Original Likelihood**	**Original Consequence**
Integration of the excavator and grader software with hardware takes longer than expected			**5**	**5**
Consequence			**Mitigated Likelihood**	**Mitigated Consequence**
Unable to meet SVD deadline and potential requirements change			**2**	**5**
Risk Reduction Plan Summary			**Risk Type:**	Technical
Action/Milestone		**Success Criteria**	**Date Planned**	**Date Implemented**
Shift requirements for SVD		Working prototype for SVD	11/28/2024	11/28/2024
Integrate the grader during Fall semester		Working excavator and grader for FVD	11/28/2024	
Potentially use off-the-shelf code if avaliable, preferably from CraterGrader		Successful integration of off-the-shelf components		
Comments				
Decided to move delivery of grader tool planner to the Fall semester				

Figure 31: Excavator and Grader Tool Planner takes Longer than Expected to Deliver

Risk ID	Risk Title	Risk Owner	Date Added	Date Updated
R3	Integration issues between subsystems	Deepam	11/27/2024	11/27/2024
Description			**Original Likelihood**	**Original Consequence**
Subsystems work individually, but integration and communication between the subsystems are flawed			**3**	**5**
Consequence			**Mitigated Likelihood**	**Mitigated Consequence**
Delay in integration causing scheduling overruns, requirements change and failure of the demo			**2**	**5**
Risk Reduction Plan Summary			**Risk Type:**	Technical
Action/Milestone		**Success Criteria**	**Date Planned**	**Date Implemented**
Perform unit testing and subsystem validation continuously		Successful testing of all major subsystems	11/30/2024	04/04/2025
Integrate one subsystem at a time		Successful integration of all major subsystems	11/30/2024	04/04/2025
Use a common framework (e.g. ROS2 interfaces) for communication between subsystems to reduce bugs		Adoption of common framework for communications	11/30/2024	01/18/2025
Keep to planned schedule and have at least 5 weeks for testing and integration		Successful integration of all major subsystems	11/30/2024	03/25/2025
Comments				

Figure 32: Integration Issues between Subsystems

Risk ID	Risk Title	Risk Owner	Date Added	Date Updated
R4	Belly depth sensor is not suitable for validation	Bhaswanth	11/27/2024	11/27/2024
Description			**Original Likelihood**	**Original Consequence**
The belly depth camera is used to validate if a groomed crater is satisfiable. The sensor may not be able to adequately determine depth variations suitable for validation			**4**	**3**
Consequence			**Mitigated Likelihood**	**Mitigated Consequence**
Will result in major revision and changes to the validation architecture and functional requirement, causing delays in scheduling			**2**	**2**
Risk Reduction Plan Summary			**Risk Type:**	Technical
Action/Milestone		**Success Criteria**	**Date Planned**	**Date Implemented**
Mount the depth camera at another location on the rover (e.g. on a mast)		Acceptable validation specifiied by performance requirement		
Use another sensor to determine depth variations (e.g. LIDAR, visual odometry, IR sensor)		Acceptable validation specifiied by performance requirement		
If all else fails, use the total station for validation		Acceptable validation specifiied by performance requirement		
Comments				

Figure 33: Belly Depth Sensor is not Suitable for Validation

Risk ID	Risk Title	Risk Owner	Date Added	Date Updated
R5	Unable to get Crater Grader to perform autonomous crater filling	Bhaswanth	11/27/2024	11/27/2024
Description			**Original Likelihood**	**Original Consequence**
Our rover builds on top of the work acomplished by Crater Grader. If we cannot get Crater Grader to perform autonomous crater filling, we may need to spend time on the navigation stack and design the entire pipeline			**3**	**3**
Consequence			**Mitigated Likelihood**	**Mitigated Consequence**
Extra time commitment to start from scratch or obtaining a suitable replacement			**2**	**2**
Risk Reduction Plan Summary			**Risk Type:**	Technical
Action/Milestone		**Success Criteria**	**Date Planned**	**Date Implemented**
Thoroughly go through Crater Grader's code and the mechanical schematics provided		Thoroughly understand Crater Grader's operations	11/27/2024	
Test each component and wiring to see if they are working		Validate all components and replace broken ones	11/28/2024	
If it is still not working, inherit only the software component from Crater Grader and build hardware ourselves		Working prototype for SVD		
Comments				

Figure 34: Unable to get Crater Grader to Perform Autonomous Crater Filling

Risk ID	Risk Title	Risk Owner	Date Added	Date Updated
R6	Delay in arrival and manufacture of hardware components	William	11/27/2024	11/27/2024
Description			**Original Likelihood**	**Original Consequence**
Shipping delays of components ordered and/or manufacturing delays on custom made components			**3**	**5**
Consequence			**Mitigated Likelihood**	**Mitigated Consequence**
Delays in hardware integration, causing pushbacks in schduling and software development			**2**	**5**
Risk Reduction Plan Summary			**Risk Type:**	Scheduling
Action/Milestone		**Success Criteria**	**Date Planned**	**Date Implemented**
Use off-the-shelf components that are avaliable on hand (e.g. from CMU labs or Red's workshop)		Obtain components before end of December		
Start ordering and designing components during Winter break so there is adequate leeway for delivery and manufacturing before Spring semester starts		Order components before end of December	11/27/2024	12/09/2024
Use simulations to work on software components while we wait for the components to be delivered and/or manufactured		Successful integration of all subsystems on schedule		
Implement other subsystems that are independent from the subsystem that is missing parts		Successful integration of all subsystems on schedule		
In case of delay in wheels, work with the existing wheels and proceed with the timeline while waiting for the new ones to arrive		Successful integration of all subsystems on schedule		
Comments				

Figure 35: Delay in Arrival and Manufacture of Hardware Components

Risk ID	Risk Title	Risk Owner	Date Added	Date Updated
R7	Lack of proper simulation environment	Simson	11/27/2024	11/27/2024
Description			**Original Likelihood**	**Original Consequence**
Inability to accurately simulate the rover in a Lunar-like environment can lead to suboptimal performance			**3**	**3**
Consequence			**Mitigated Likelihood**	**Mitigated Consequence**
The rover's performance in the Moon Pit may be compromised, leading to inefficiencies, mission delays, or potential failure in achieving key objectives			**1**	**2**
Risk Reduction Plan Summary			**Risk Type:**	Technical
Action/Milestone		**Success Criteria**	**Date Planned**	**Date Implemented**
Ask CraterGrader how they ran all their simulations and gather resources		Meet with CraterGrader team	11/28/2024	12/2/2024
Explore LunarSim - https://github.com/PUTvision/LunarSim and check how useful this will be, during the winter break		Working simulation	12/12/2024	
Develop Gazebo environment		Working simulation		
Comments				

Figure 36: Lack of Proper Simulation Environment

Risk ID	Risk Title	Risk Owner	Date Added	Date Updated
R8	Mast depth camera field of view (FOV) is blocked	William	11/27/2024	11/27/2024
Description			**Original Likelihood**	**Original Consequence**
Mast depth camera's FOV can be blocked, partially or completely, due to dust, misalignment of camera, or interference from the rover's own excavator assembly.			**5**	**4**
Consequence			**Mitigated Likelihood**	**Mitigated Consequence**
Hinders the rover's ability to perceive its surroundings accurately, resulting in navigation errors and inefficiencies in excavation tasks			**3**	**4**
Risk Reduction Plan Summary			**Risk Type:**	Technical
Action/Milestone		**Success Criteria**	**Date Planned**	**Date Implemented**
Conduct field tests to choose an optimal height to place the depth camera such that dust does not reach it and it can clearly see in front of the rover, despite the excavator assembly. Ensure that visual data such as depth perception and object detection should not be compromised		Working mast depth camera with a clear FOV		
Comments				

Figure 37: Mast Depth Camera Field of View (FOV) is Blocked

Risk ID	Risk Title	Risk Owner	Date Added	Date Updated
R9	Too many performance requirements	Ankit	11/27/2024	11/27/2024
Description			**Original Likelihood**	**Original Consequence**
We have a lot of performance requirements and we may not be able to meet all of them by April for SVD			**5**	**5**
Consequence			**Mitigated Likelihood**	**Mitigated Consequence**
Delays in testing and validation, impacting project timelines and April SVD Demo results			**2**	**5**
Risk Reduction Plan Summary			**Risk Type:**	Technical, Scheduling
Action/Milestone		**Success Criteria**	**Date Planned**	**Date Implemented**
Have revised performance requirements separately for SVD and FVD (focus more on SVD)		Achievable Performance Requirements	11/28/2024	12/4/2024
Talk to CraterGrader and discuss what is feasible and what is not in the given time		Meeting conducted	11/28/2024	12/2/2024
PM should track schedule properly and team members have to push to meet the timeline		Project follows the schedule	11/28/2024	
Comments				

Figure 38: Too Many Performance Requirements

Risk ID	Risk Title	Risk Owner	Date Added	Date Updated
R10	Drive system wear-and-tear causes malfunction	Deepam	11/27/2024	11/27/2024
Description			**Original Likelihood**	**Original Consequence**
The transmission and steering assembly might be worn out, leading to suboptimal vehicle dynamics, and potentially mechanical failure			**4**	**4**
Consequence			**Mitigated Likelihood**	**Mitigated Consequence**
Rover drive system fails and may require a lot of repair and maintenance			**2**	**2**
Risk Reduction Plan Summary			**Risk Type:**	Technical
Action/Milestone		**Success Criteria**	**Date Planned**	**Date Implemented**
Thoroughly check the Crater Grader's assembly and carry out maintenance of any worn-out parts		Successfully understand and carry out maintenance of existing parts and assemblies		
Completely replace the assembly parts with the same/similar new parts for better performance and reliability		Order and stock spares		
Add limit switches to avoid steering gears to operate beyond their limits		Limit switches added		
Comments				

Figure 39: Drive System Wear-and-Tear Causes Malfunction

Risk ID	Risk Title	Risk Owner	Date Added	Date Updated
R11	Dust ingress	William	11/27/2024	11/27/2024
Description			**Original Likelihood**	**Original Consequence**
Due to significant sand manipulation, the flying sand/dust can enter and accumulate over sensitive electronics (PDB, drivers, Arduino) and sensors (cameras, IMU), leading to component failure or incorrect sensing			**5**	**3**
Consequence			**Mitigated Likelihood**	**Mitigated Consequence**
Component failure during testing or demonstrations. Highly inhibits all future scheduled tasks			**3**	**3**
Risk Reduction Plan Summary			**Risk Type:**	Technical, Cost
Action/Milestone		**Success Criteria**	**Date Planned**	**Date Implemented**
Design proper sand enclosures and mounts for sensitive components		Successfully design and manufacture enclosures		
Review placement of components		Components are placed aptly, away from dust		
Review scale and speed of sand manipulation to eliminate root-cause of flying sand/dust		Select the sweet spot for apt tool speed with least flying dust/sand		
Allocate contingency budget and order spares of the sensitive components in case of component failure		Order and stock spares		
Comments				

Figure 40: Dust Ingress

Risk ID	Risk Title	Risk Owner	Date Added	Date Updated
R12	Code version control	Simson	11/27/2024	11/27/2024
Description			**Original Likelihood**	**Original Consequence**
Code modifications or config parameter changes during testing might not be saved, affecting the final demo. Reverting to a stable version is difficult if changes do not work as expected			**3**	**4**
Consequence			**Mitigated Likelihood**	**Mitigated Consequence**
Delay in code integration and implementation			**1**	**4**
Risk Reduction Plan Summary			**Risk Type:**	Technical
Action/Milestone		**Success Criteria**	**Date Planned**	**Date Implemented**
Implement GitHub version control to store and retrieve the best versions of code and configuration		Successful tracking of code changes		
Use Google Drive to back up important documentation explaining setup processes		Reduces delays during testing due to quick access to setup processes		
Comments				

Figure 41: Code Version Control

Risk ID	Risk Title	Risk Owner	Date Added	Date Updated
R13	Items missing	Ankit	11/27/2024	11/27/2024
Description			**Original Likelihood**	**Original Consequence**
Critical project items may go missing if not stored properly or tracked. Items may be misplaced or borrowed without proper logging			**4**	**3**
Consequence			**Mitigated Likelihood**	**Mitigated Consequence**
Delay in hardware implementation			**2**	**2**
Risk Reduction Plan Summary			**Risk Type:**	Logistics
Action/Milestone		**Success Criteria**	**Date Planned**	**Date Implemented**
Maintain an inventory tracking spreadsheet		Ensures availability of required tools and materials		
Include spare inventory		Reduces downtime caused by missing or damaged items		
Comments				

Figure 42: Items Missing

Risk ID	Risk Title	Risk Owner	Date Added	Date Updated
R14	Sensor ROS packages not available	William	11/27/2024	11/27/2024
Description			**Original Likelihood**	**Original Consequence**
Finalized sensors might lack compatible ROS packages, leading to delays or significant changes in the software architecture			**3**	**3**
Consequence			**Mitigated Likelihood**	**Mitigated Consequence**
Delay in software implementation			**1**	**3**
Risk Reduction Plan Summary			**Risk Type:**	Technical, Scheduling
Action/Milestone		**Success Criteria**	**Date Planned**	**Date Implemented**
Perform trade studies to pick sensors that are compatible with ROS versions before finalizing		Sucessful sensor-ROS compatibility		
Select sensors and ROS versions that minimize potential conflicts		Streamlined integration with minimal issues		
Comments				

Figure 43: Sensor ROS Packages not Available

Risk ID	Risk Title	Risk Owner	Date Added	Date Updated
R15	Lunar-accurate cut/fill regions are not possible to groom	Simson	11/27/2024	11/27/2024
Description			**Original Likelihood**	**Original Consequence**
The rims of the craters may not be enough to fill the whole crater. Going to a different region to carry the sand to the crater may prove to be inefficient			**3**	**3**
Consequence			**Mitigated Likelihood**	**Mitigated Consequence**
The basic assumption of sand availability fails. We may need to rethink the basic concept of tool planner to fit the new parameters of the environment.			**2**	**2**
Risk Reduction Plan Summary			**Risk Type:**	Technical
Action/Milestone		**Success Criteria**	**Date Planned**	**Date Implemented**
Accurately create the environment and assess if the rims are enough to fill		Assessment gives us adequate information		
If not, modify PRs accordingly		Achievable Performance Requirements		
Comments				

Figure 44: Lunar-Accurate Cut/Fill Regions are not Possible to Groom

Risk ID	Risk Title	Risk Owner	Date Added	Date Updated
R16	Sensor data is too noisy to fulfill performance requirements	William	11/27/2024	11/27/2024
Description			**Original Likelihood**	**Original Consequence**
Performance requirements are tough and ambitious, sensor noise may prevent us from achieving it			4	4
Consequence			**Mitigated Likelihood**	**Mitigated Consequence**
Failure to demonstrate performance requirements may cause us to lose marks in the demonstrations			2	4
Risk Reduction Plan Summary			**Risk Type:**	Technical
Action/Milestone		**Success Criteria**	**Date Planned**	**Date Implemented**
Relax the performance requirements enough to ensure that they are achievable		Achievable Performance Requirements		
Ensure enough testing time to tune parameters		Fully planned testing cycle	11/28/2024	
Comments				

Figure 45: Sensor Data is too Noisy to Fulfill Performance Requirements

Risk ID	Risk Title	Risk Owner	Date Added	Date Updated
R17	Off-the-shelf wheels don't interface with the rover	Ankit	11/27/2024	11/27/2024
Description			**Original Likelihood**	**Original Consequence**
No off-the-shelf wheels fit the rover, We'll have to redesign wheel hubs and mountings as per the new wheels.			3	3
Consequence			**Mitigated Likelihood**	**Mitigated Consequence**
Continue with sub-optimal wheels that the rover currently has, thus, not meeting one of the non-functional requirements			2	3
Risk Reduction Plan Summary			**Risk Type:**	Technical
Action/Milestone		**Success Criteria**	**Date Planned**	**Date Implemented**
Shift requirements to FVD		Updated SVD and FVD requirements for wheels		
Good enough market research to see find the best fit, with least amount of changes		Finding and replacing current wheels with new wheels, with least modifications		
Comments				

Figure 46: Off-the-Shelf Wheels don't Interface with the Rover

Risk ID	Risk Title	Risk Owner	Date Added	Date Updated
R27	TX2 Integration	William	1/2/2025	2/24/2025
Description			**Original Likelihood**	**Original Consequence**
Unable to login to TX2 and interface with a LAN network for transmitting data over WiFi ro Jetson			3	3
Consequence			**Mitigated Likelihood**	**Mitigated Consequence**
Delay in finalizing localization stack			2	2
Risk Reduction Plan Summary			**Risk Type:**	Technical
Action/Milestone		**Success Criteria**	**Date Planned**	**Date Implemented**
Set up a new TX2 (Re-flash the TX2).		Successfully set up a new TX2 or re-flash the existing TX2 with updated credentials and settings	01/02/2025	
Reach out to previous teams to understand their methodology and retrieve credentials		Successfully retrieve credentials and interface with LAN	01/02/2025	02/19/2025
Comments				

Figure 47: TX2 Integration

Risk ID	Risk Title	Risk Owner	Date Added	Date Updated
R28	Electrical Hardware Finalization	Ankit	2/14/2025	2/14/2025
Description			Original Likelihood	Original Consequence
E-box Design dependence on to-be manufactured PDB.			3	4
Consequence			Mitigated Likelihood	Mitigated Consequence
Not meeting the hardware deadline			2	4
Risk Reduction Plan Summary			Risk Type:	Technical, Logistics
Action/Milestone		Success Criteria	Date Planned	Date Implemented
Use previous knowledge and account for a placeholder in the design.		Successfully design and manufacture E-box compatible with the new PCB using placeholder PCB design	02/14/2025	04/06/2025
Order PCB and components (and spares) outside of MRSD schedule		Successfully order and assemble the PCB	03/26/2025	04/04/2025
Comments				

Figure 48: Electrical Hardware Finalization

Risk ID	Risk Title	Risk Owner	Date Added	Date Updated
R29	Access to FRC Workshop	Deepam	2/7/2025	2/7/2025
Description			Original Likelihood	Original Consequence
Without access, no hardware fabrication/repairs can be carried out in the absence of Tim			4	4
Consequence			Mitigated Likelihood	Mitigated Consequence
			2	2
Risk Reduction Plan Summary			Risk Type:	Logistics
Action/Milestone		Success Criteria	Date Planned	Date Implemented
Try other fab-labs on campus.		Successfully access other fab-labs and manufacture components	2/9/2025	
Request Tim, John or Red for getting temporary access, if not permanent		Successfully get temporary/permanent access to FRC Workshop	2/12/2025	
Comments				

Figure 49: Access to FRC Workshop

Risk ID	Risk Title	Risk Owner	Date Added	Date Updated
R30	No spares available	Team	3/4/2025	3/4/2025
Description			Original Likelihood	Original Consequence
Discontinued model, spare parts unavailable			4	5
Consequence			Mitigated Likelihood	Mitigated Consequence
The whole project falling through, or redo almost all subsystems on a different rover.			3	5
Risk Reduction Plan Summary			Risk Type:	Logistics
Action/Milestone		Success Criteria	Date Planned	Date Implemented
Check out eBay and other similar platforms for spares		Successfully find exact spares on these platforms	3/6/2025	
Check out and stock similar parts if not same		Successfully find and stock similar parts	3/6/2025	
Find a twin rover that was used by a previous team on campus		Successfully find the twin rover and scavenge parts	3/6/2025	3/7/2025
Maintain all parts, especially mechanical parts		Successfully avoid future breakdowns and part failures	3/7/2025	
Comments				

Figure 50: No spares available

Risk ID	Risk Title	Risk Owner	Date Added	Date Updated
R31	Localization Accuracy	Team	4/2/2025	4/2/2025
Description			**Original Likelihood**	**Original Consequence**
The robot experiences delays in localization and minor positional offsets during movement			3	4
Consequence			**Mitigated Likelihood**	**Mitigated Consequence**
May prevent the robot from reaching the goal location, blocking the transition to the tool planner and affecting grading			2	2
Risk Reduction Plan Summary			**Risk Type:**	Logistics
Action/Milestone		**Success Criteria**	**Date Planned**	**Date Implemented**
Tune EKF parameters and ensure synchronized motor speeds for accurate odometry		Successfully localize the robot in the moon yard during movement	4/2/2025	
Comments				

Figure 51: Localization Accuracy

Risk ID	Risk Title	Risk Owner	Date Added	Date Updated
R32	Navigation and Tool Planner Stack Integration	Team	4/4/2025	4/4/2025
Description			**Original Likelihood**	**Original Consequence**
Navigation and tool planners use differently processed costmaps generated from the point cloud. As the navigation package requires waypoints in a specific format and in a sequence, the tool planner will have to modified and tuned accordingly			2	4
Consequence			**Mitigated Likelihood**	**Mitigated Consequence**
Errors in manipulation of sand, leading to sub-par grading			2	2
Risk Reduction Plan Summary			**Risk Type:**	Logistics
Action/Milestone		**Success Criteria**	**Date Planned**	**Date Implemented**
Extensive testing and tuning		Successfully tune parameters and integrate navigation and tool planner stack	4/5/2025	
Minimize differences between the maps used by tool planner and navigation stack by setting the correct map frame in the point cloud		One ground truth map frame	4/1/2025	04/01/2025
Comments				

Figure 52: Navigation and Tool Planner Stack Integration

Risk ID	Risk Title	Risk Owner	Date Added	Date Updated
R33	Localization frame shift after total station battery swap	Bhaswanth	3/4/2025	4/10/2025
Description			**Original Likelihood**	**Original Consequence**
Battery replacement in the total station causes small frame offsets, leading to localization inaccuracies.			5	5
Consequence			**Mitigated Likelihood**	**Mitigated Consequence**
Leads to poor navigation performance and risk of missing the crater during grading operations.			3	4
Risk Reduction Plan Summary			**Risk Type:**	Logistics
Action/Milestone		**Success Criteria**	**Date Planned**	**Date Implemented**
Implement resection method using three known prism locations instead of orientate-to-line		Successfully fix the frame consistently after battery swaps	4/26/2025	
Explore and test alternative localization methods (using SkyCam)		Successfully maintain localization accuracy	4/26/2025	
Comments				

Figure 53: Localization frame shift after total station battery swap

Risk ID	Risk Title		Risk Owner	Date Added	Date Updated
R34	Arduino requires reset before operation		Bhaswanth	3/4/2025	4/10/2025
Description				**Original Likelihood**	**Original Consequence**
Arduino needs to be manually reset each time before starting autonomy or switching between autonomy and teleoperation modes.				5	5
Consequence				**Mitigated Likelihood**	**Mitigated Consequence**
Slows down setup time and impacts operational readiness, delaying mission start and mode transitions.				3	4
Risk Reduction Plan Summary				**Risk Type:**	Logistics
Action/Milestone			**Success Criteria**	**Date Planned**	**Date Implemented**
Check USB port permissions and drivers issues on Jetson			Successfully establish consistent serial connection without reset	4/26/2025	
Verify that Arduino is connected via USB 3.0 instead of USB 2.0 port			Ensure stable high-speed communication	4/26/2025	
Check for ROS node frequency mismatches causing packet loss to Arduino			Match ROS publish/subscribe rates	4/26/2025	
Comments					

Figure 54: Arduino requires reset before operation