American International University-Bangladesh (AIUB)

Department of Computer Science
Faculty of Science & Technology (FST)

Space Rover Application

A Software Engineering Project Submitted

By

Semester: Fall 2023-2024		Section: G	Group Number: 2	
S N	Student Name	Student ID	Contribution (CO1+CO2)	Individual Marks
1	Hasan, Shakib	22-46786-1	17.2%	
2	Abha, Esm-e Moula Chowdhury	22-47166-1	21.6%	
3	Akash, Susham Moula Choudhury	22-47168-1	20.5%	
4	Suva, Md. Wahiduzzaman	22-47171-1	20.6%	
5	Sajin, Md. Imtiaj Alam	22-47185-1	20.1%	

Background to the Problem

In recent decades, space exploration has captivated the imagination of humanity, pushing the boundaries of scientific knowledge and technological innovation. Mars exploration is captivating due to its potential to provide insights into our solar system's history, the possibility of extraterrestrial life, and future human colonization. The challenge lies in remote planetary exploration, necessitating advanced robotic systems.

The problem arises from Mars' extreme inhospitality to humans due to factors like extreme cold, low atmospheric pressure, radiation, and dust storms. Robotic systems, like rovers, are essential to overcome these challenges and explore Mars safely and effectively.

This problem is super important because it affects our ability to explore and understand Mars. Mars can teach us a lot about how planets evolve, whether there's life beyond Earth, and if we can live on other planets. When we send robots to Mars, we learn more about robots, computers, and how they can help us everywhere, not just in space. In simple words, solving this problem helps us explore space, survive as a species, and make cool technology.

Solution to the Problem

The objective is to develop software to remotely control space rovers for planetary exploration missions, collect scientific data, enable self-guided movement, ensure real-time communication, store, and analyze data, and to learn the extraplanetary environment. The proposed solutions aim to improve space rover control and data management in remote environments. These include a user-friendly interface, advanced data transmission, and navigation with obstacle detection, onboard data tools, integrated AI to learn the environment and security measures. These solutions collectively enhance efficiency, reliability, and security in space missions. It is feasible and aligns well with the business objective of enhancing efficiency, reliability, and security in space missions.

Key functionalities include advanced data analytics, real-time communication, autonomous navigation with obstacle avoidance, and robust security measures. This creative solution not only enhances mission efficiency and safety but also contributes to scientific knowledge, inspires public interest, and influences space policy. Astronomers and astrophysicists gain access to high-quality imagery and spectral data. Meanwhile, engineers and technologists find in the rover a platform for pushing the boundaries of robotics, autonomy, and communication systems, and technological innovation. Educational institutions and the public are not left behind, as they are inspired and educated through real-life data, mission updates, and captivating discoveries. This application also promotes global collaboration among space agencies, forging international cooperation and diplomacy while sharing the costs and risks associated with space exploration.

Our project will allow the rover to automatically operate faster using artificial intelligence and machine learning technology to scan the unknown terrain, collect data samples of any unusual and rare objects and the atmosphere in the outer space as well as the terrain itself and send the data back to the Base of operations more efficiently than before. There have been other studies and experiments done on this topic, but our project will be more robust, easier to use, the size of the designed software will be smaller, which will require less processor requirement which is followed by less or lighter equipment. This will also decrease the weight of the rover allowing faster execution of operation, faster movement on the terrain in outer space and will minimize the requirement of human control resulting in lesser cost of production, operation as well as maintenance of the rover for their field operations in outer space giving promising result for space expeditions.

In the Mars rover field, existing studies, and solutions primarily center on NASA's missions like Curiosity and Perseverance, plus ESA's ExoMars rover. These missions employ advanced software for navigation, instrument control, and communication. Their solutions include ROS (Robot Operating System), navigation algorithms, DTN (Delay-Tolerant Networking), and tools like Maestro and SAP. Our software builds on these by integrating machine learning, onboard data analysis, and safety protocols for enhanced autonomy and efficiency, aiming to benefit users and meet evolving mission needs.

Model Selection

We have chosen a plan-driven model, specifically the Waterfall model, for our space rover application development project. The choice of the Waterfall model is driven by the following considerations:

Well-Defined Requirements: Given the critical nature of space missions and the need for precise specifications, our project's requirements are expected to be well-defined and stable. The Waterfall model's linear and sequential approach aligns with this characteristic.

Safety and Regulatory Compliance: Space missions, particularly those involving rovers, demand strict adherence to safety standards and regulatory requirements. The Waterfall model's emphasis on documentation, rigorous testing, and traceability ensures compliance with these essential criteria.

Robust Documentation: In the development of software for space rovers, comprehensive documentation is crucial for design, verification, and validation. The Waterfall model encourages the creation of detailed documents at each phase, facilitating precise documentation.

Reasons for Not Choosing Other Models

While the Waterfall model aligns well with our project's characteristics, other models may not be as suitable due to the following factors:

Agile Models: Agile methodologies, such as Scrum, are highly adaptive to evolving requirements but may lack the level of documentation and formality required for space missions, which often involve safety-critical systems.

Iterative Models: Iterative models are beneficial for rapidly evolving projects but might not offer the same level of predictability and regulatory compliance as plan-driven models, which are favored in safety-critical domains.

Hybrid Models: While hybrid models combine the strengths of different approaches, they can add complexity and may require more extensive management in ensuring compliance and traceability.

Roles and Responsibilities:

In our plan-driven Waterfall model, the following roles and responsibilities are essential:

- **Project Manager:** Responsible for project planning, scheduling, and overall project coordination. Ensures that the project stays on track, within scope, and on budget.
- **Business Analyst:** Gathers, analyzes, and documents project requirements in detail. Works closely with stakeholders to ensure a clear understanding of their needs.
- **System Architect/Designer:** Designs the system architecture and components based on the detailed requirements. Creates the overall system design and interfaces.
- **Development Team:** Programmers and engineers responsible for implementing the design and developing the space rover software and hardware.
- **Quality Assurance/Test Team:** Performs comprehensive testing and quality assurance activities at various stages of development to ensure that the rover's software and hardware meet the specified requirements.
- **Documentation Team:** Responsible for creating project documentation, including user manuals, technical documents, and specifications.
- **Customer/Client:** Approves project milestones.
- **Safety and Compliance Experts:** Ensure that the project adheres to all safety and regulatory requirements.

These roles collectively contribute to the successful development of our space rover application while meeting the critical safety and regulatory standards associated with space missions.

Software Requirement

1. Hardware Maintenance

    1.1. In the first step, the system will incorporate dedicated sensors designed for measuring weather parameters, including wind speed, atmospheric pressure, temperature, and UV radiation levels.

1.2.   Prior to any mission, system check up should be performed

1.3.   If system failure is detected, report to base and reboot if possible

1.4.   If the system does not work after system reboot, go to the base station for repair.

1.5.   If the problem is solved, the rover will resume its mission

Priority Level: High

Preconditions: The system is currently not engaged in any type of mission. Or shuts down any mission, if it discovers any issues.

Cross Reference: 2.1, 3.1

2.   Mission Status

2.1.   In the first step, the system will incorporate dedicated sensors designed for measuring weather parameters, including wind speed, atmospheric pressure, temperature, and UV radiation levels.

2.2.   The user can check vehicle status, tire pressure, battery health and percentage and predict any problems and allow the user to make any alternate possible decision to prevent mission failure.

2.3.   The user can check progress of the mission assigned to the rover, and estimated time for mission completion

2.4.   Mobility and visibility of the vehicle can be predicted to ensure maximum efficiency of the mission

2.5.   Weather, route and atmospheric status can be checked to ensure minimization of vehicle damage during any mission

Priority Level: High

Precondition: The system is under continuous maintenance and supervision, all weather data are being collected simultaneously, while the rover is sent on a mission

Cross Reference: 2.1, 3, 4.1, 5.2

3.   Mobility

3.1.   In automatic mode, the rover will autonomously navigate diverse terrains using all-terrain drive mechanisms and obstacle avoidance algorithms.

3.2.   In manual mode, operators can control the rover's movements, including speed, direction, and stopping.

3.3.   In order to manually control the rover the operator has to communicate with the rover.

3.4.   In manual mode, Rover will provide all the data including live camera feeds, telemetry data, and status updates, to ensure precise control.

3.5.   A safety protocol will be available that prevents conflicts when transitioning between modes. The software will have a seamless switching between automatic and manual modes without compromising the rover's safety or mission objectives.

Priority Level: High

Preconditions: The system is operational and functional

Cross Reference: 5.1, 2

4. Terrain Scanning and Way Pointing

4.1. The first requirement involves the integration of multiple onboard sensors, including cameras, radar etc to gather terrain data.

4.2. After data collection, algorithms will process the sensor data to create a detailed terrain map, which will also include the detection of obstacles within the terrain.

4.3. Using the terrain data, the system will generate waypoints. These waypoints will be determined based on the mission objectives.

4.4. A path planning algorithm will be used to determine an efficient route between the waypoints.

4.5. Safety measures include obstacle avoidance algorithms to prevent collisions and other tasks, such as safe stops and mission aborts.

Priority Level: High
Precondition: The rover is engaged in a mission
Cross Reference: 5.1

5. Weather Data Collection

5.1. In the first step, the system will incorporate dedicated sensors designed for measuring weather parameters, including wind speed, atmospheric pressure, temperature, and UV radiation levels.

5.2. To ensure timely access to weather data, algorithms will be implemented to facilitate real-time data acquisition from the integrated sensors.

5.3. A data logging mechanism will be developed for recording the collected weather data, complete with timestamps, for reference and analysis.

5.4. The system will be equipped to transmit the gathered weather data to mission control or storage systems for further analysis and forecasting. The weather data collection shall include the following parameters:
- Wind Speed
- Atmospheric Pressure
- Temperature
- Ultraviolet (UV) Radiation Levels

Priority Level: Low
Precondition: The rover finds the right place and all the actuator is functioning correctly
Cross Reference: 5.1

Priority Level: Low
Precondition: The rover is on a mission and simultaneously collecting weather data, Sensors working fine.
Cross Reference: 5.1, 1

6. Communication

6.1. System will initialize the communication module when it's not in idle state or when it goes for a new mission.

6.2. Then this communication module uses a data streaming mechanism to establish

real-time communication between the rover and Base station.

6.3.   Using this base station system will connect with earth station for task execution and exchange of critical mission data, including scientific findings and rover health.

6.4.   If the system fails to communicate with the base station for a certain time but it's on a mission, the rover will change course towards the last communication location, and it will implement DTN protocols for handling intermittent and delayed communication.

6.5.   When needed, the system shall support communication with accompanying aerial vehicles (drones) for collaborative tasks.

Priority Level: Medium
Precondition: The rover is in any state.
Cross Reference: 5.1, 1

7.   Soil Sample Collection

1.1.   Find a suitable location to collect samples.

1.2.   After scanning the surface and decide whether sample collection is possible, if not relocate to new location

1.3.   Drill to a specific depth and collect the sample.

1.4.   After collecting the sample, store them in containers for further analysis.

1.5.   After completing a mission the rover shall reset to its exploration mode

Priority Level: Low
Precondition: The rover is on a mission and simultaneously collecting weather data, Sensors working fine.
Cross Reference: 5.1, 1

UML Diagram

The following section will present the UML (Unified Modeling Language) diagram of the software. This diagram serves as a visual representation of the software's architecture, illustrating the relationships and interactions between various components. It provides a comprehensive overview of the system's structure and design.
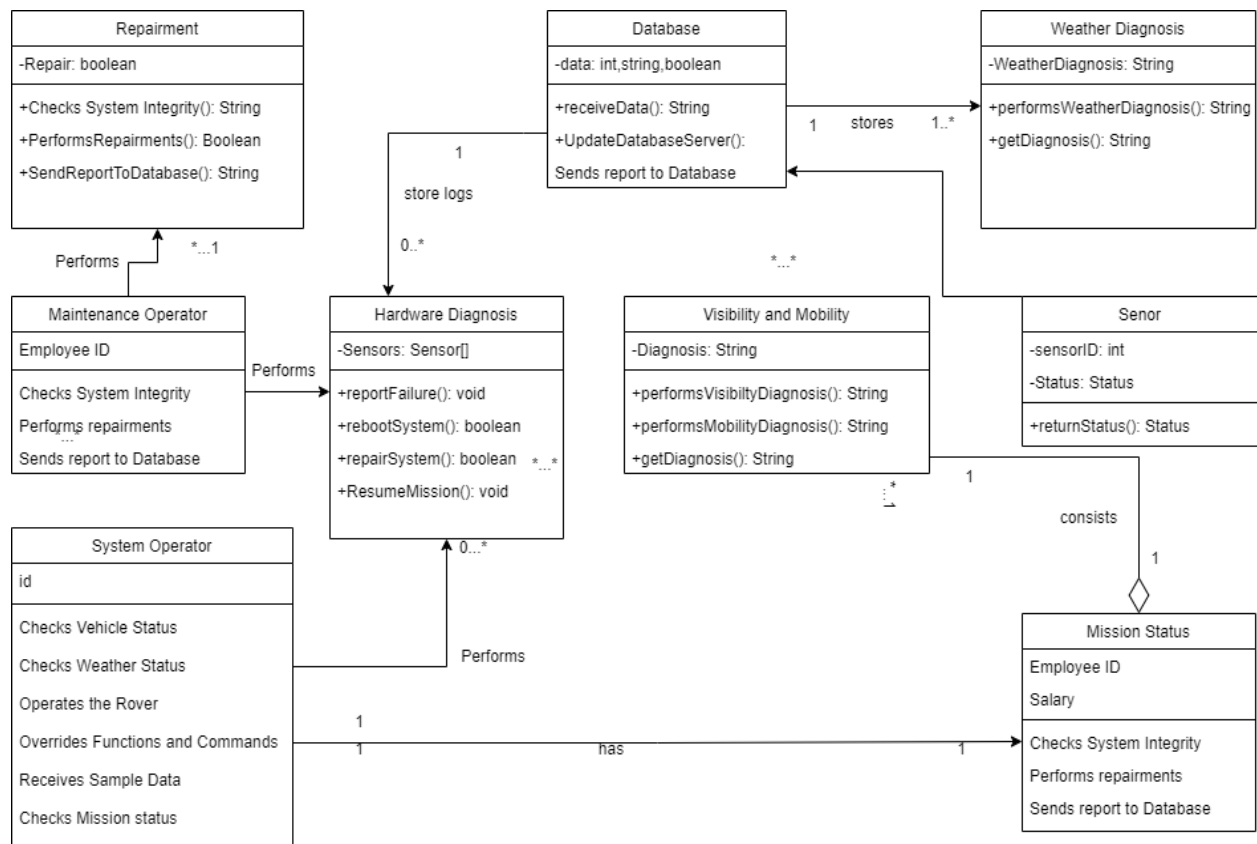
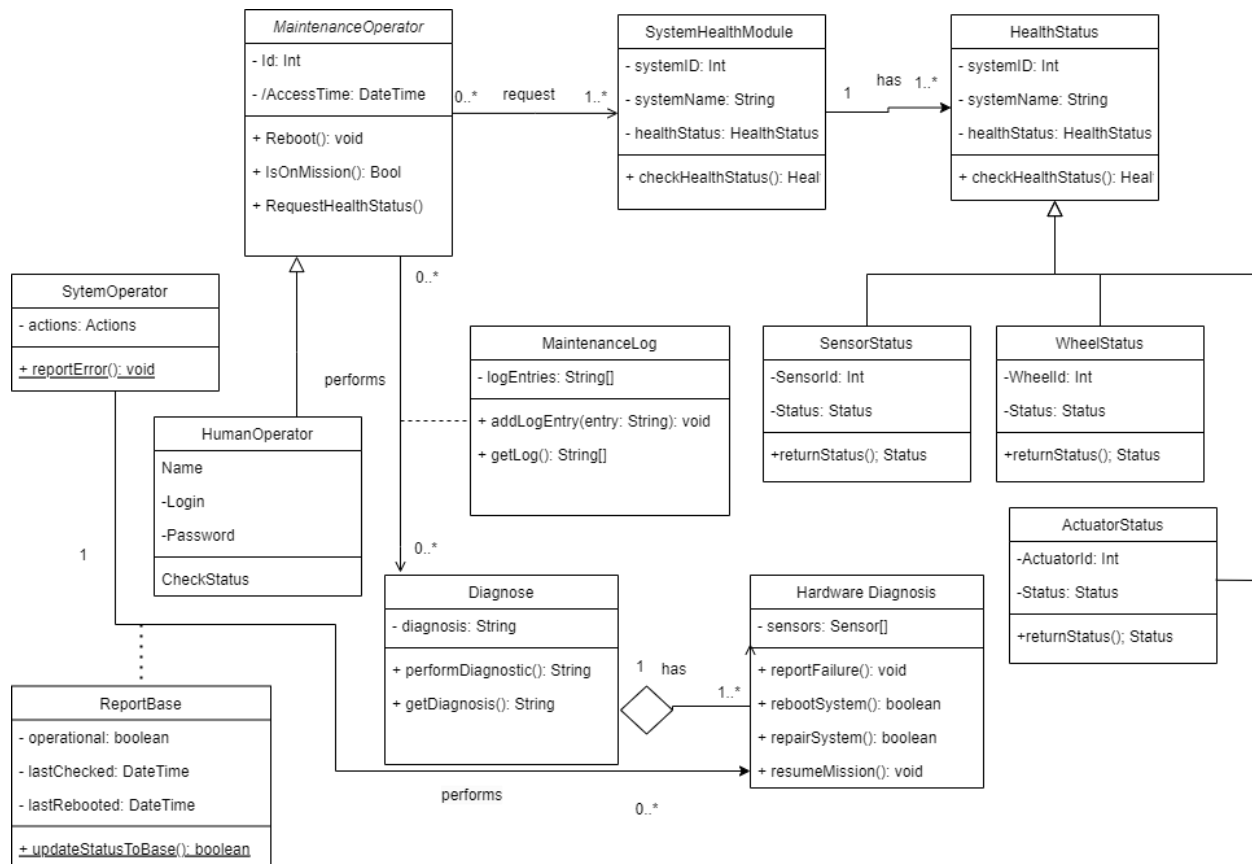## Class Diagram
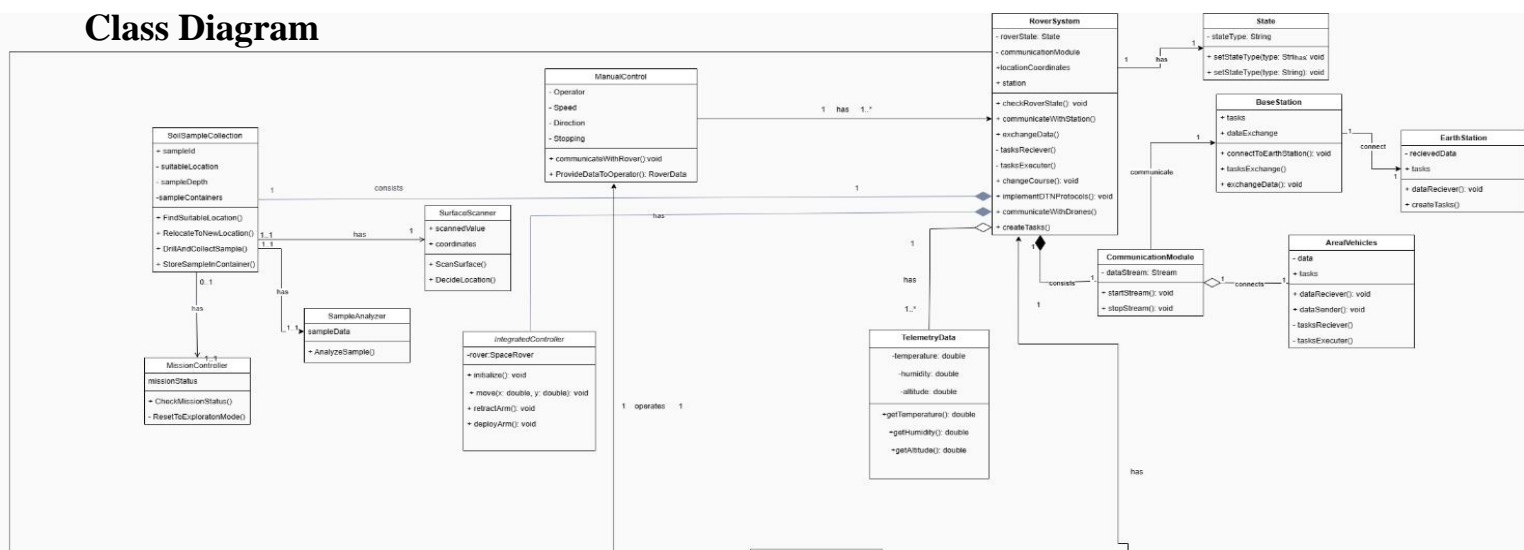
Figure 2.2: Class diagram of mission status

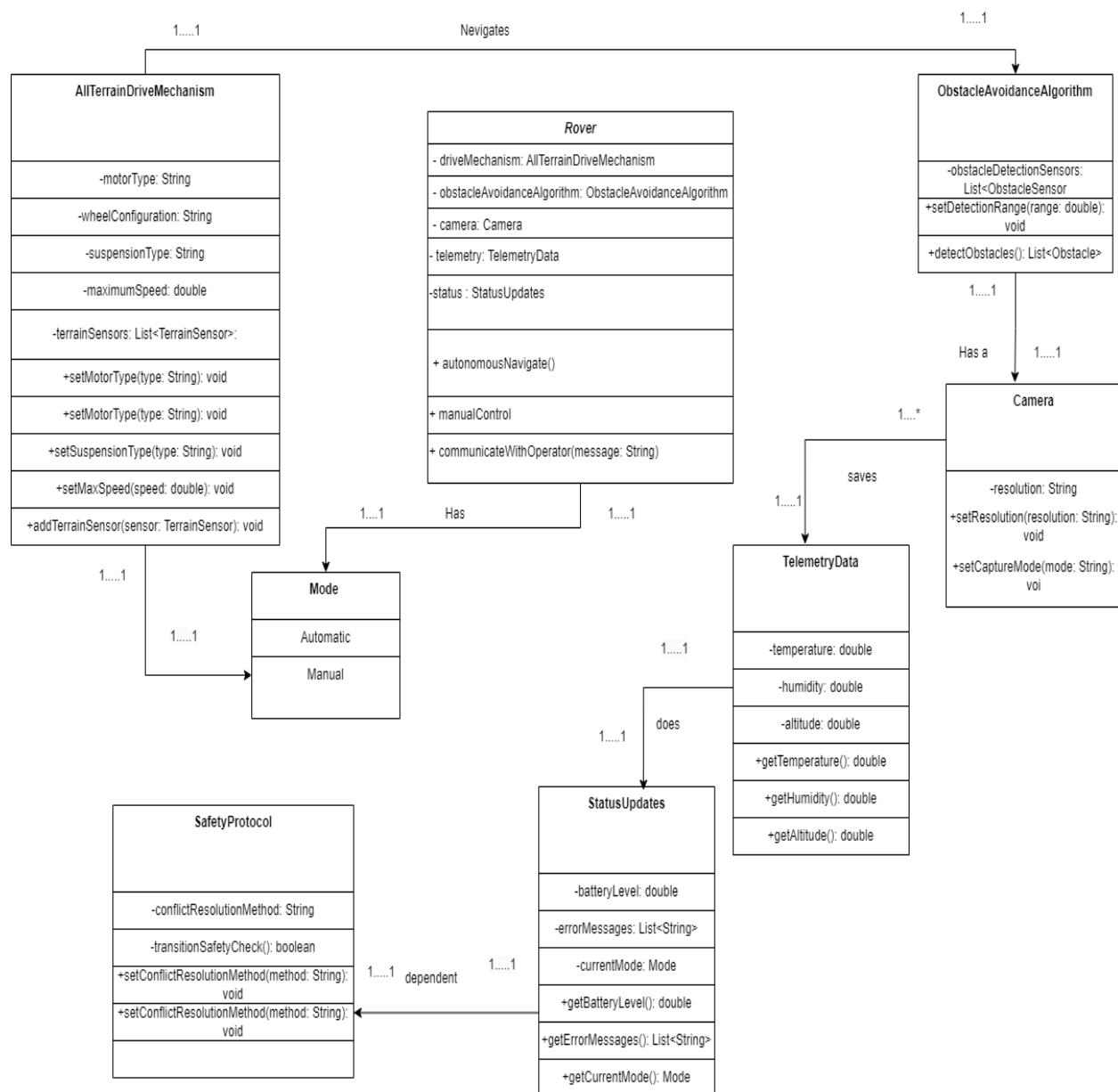Figure 1.2: Class diagram of hardware maintenance

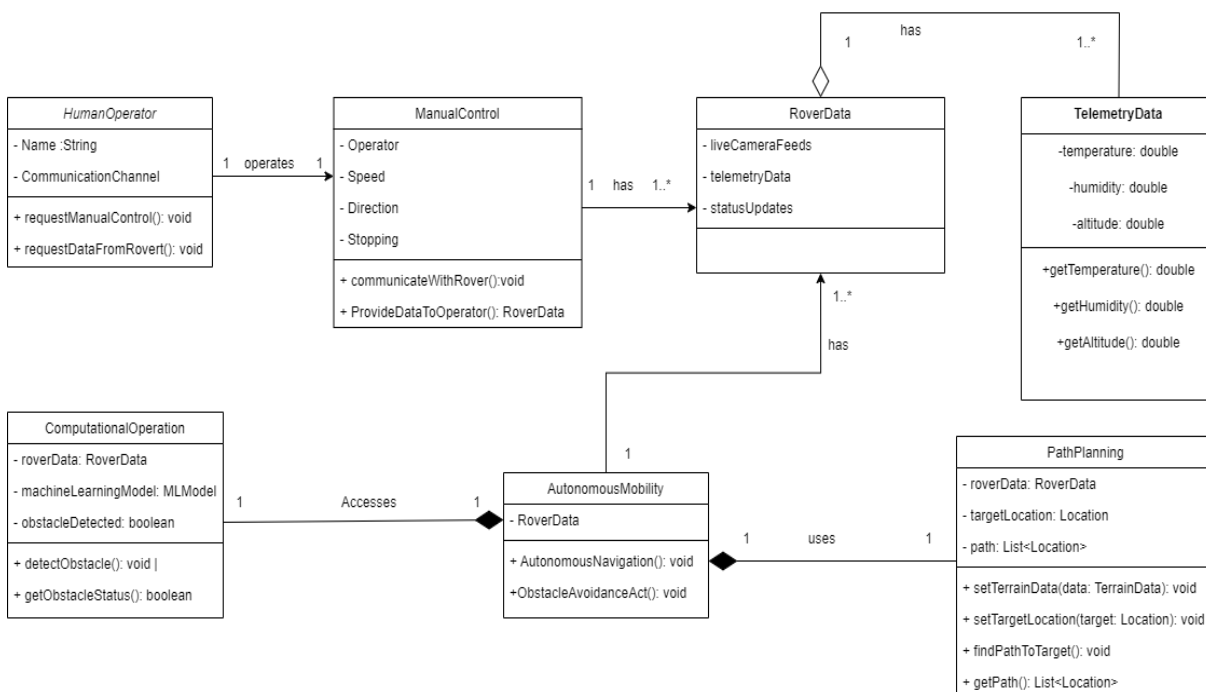Figure 3.2: Class diagram of Mobility

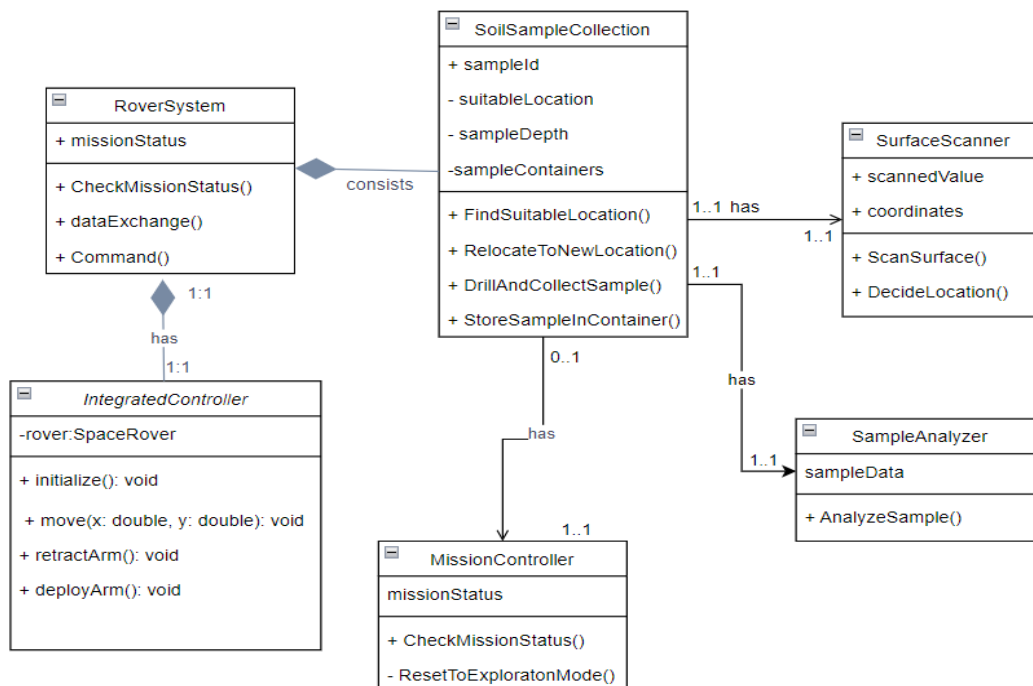Figure 4.2: Class diagram of Terrain scanning and way pointing.

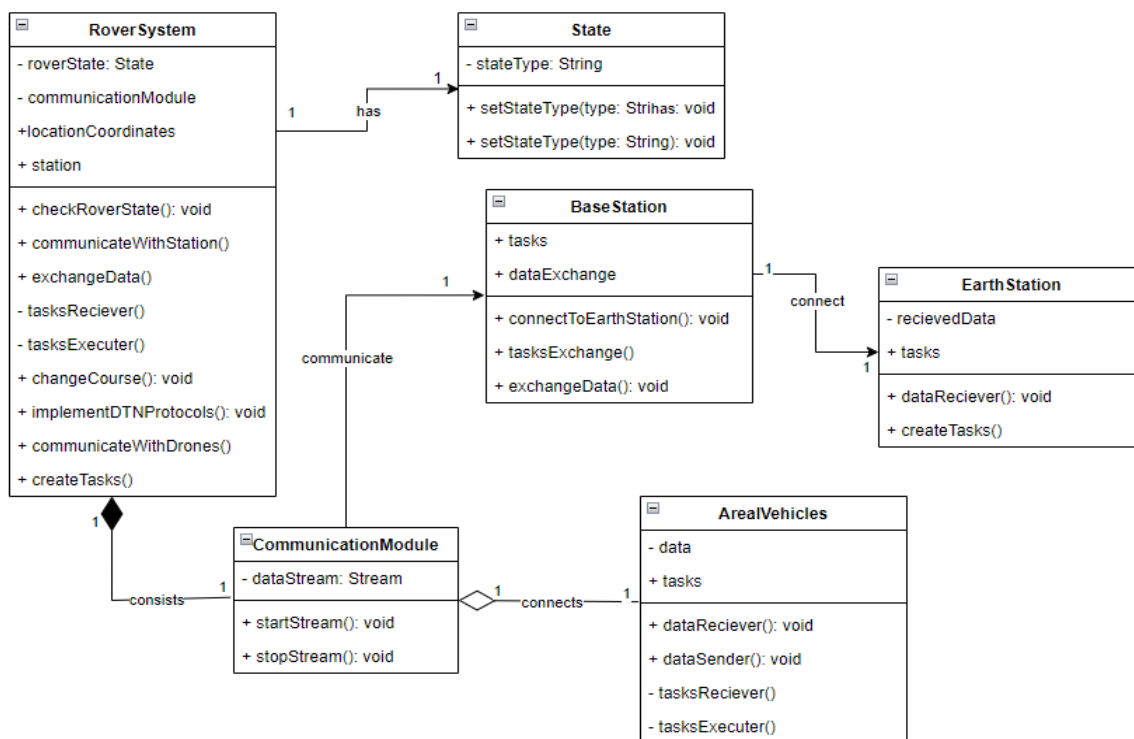Figure 5.2: Class diagram of Soil Sample Collection

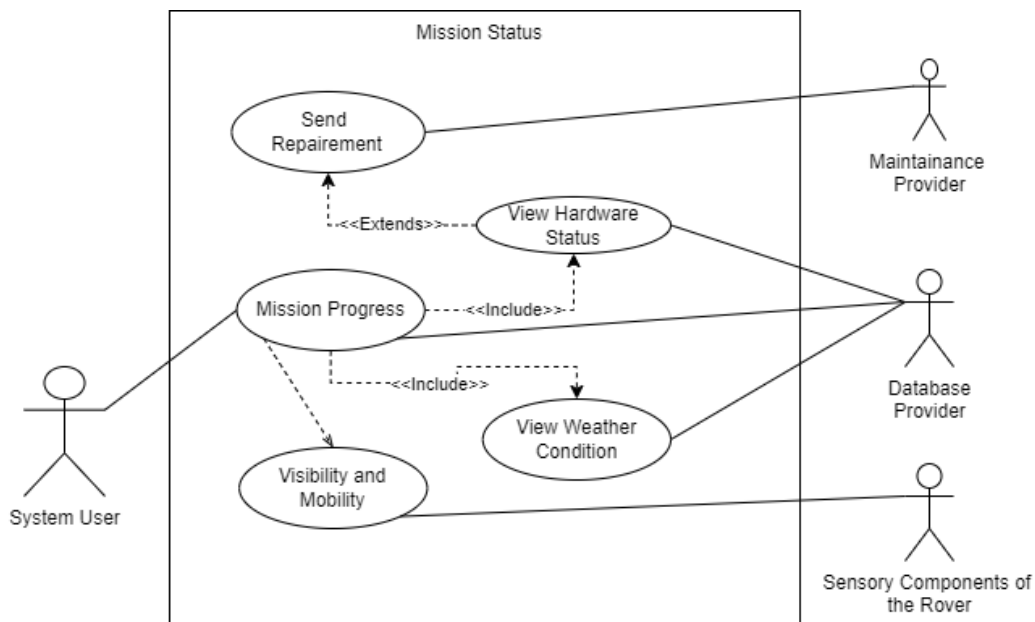Figure 6.2: Class diagram of Communication

Use Case Diagram

Figure 2.1: Use case diagram of mission status

Hardware Maintenance
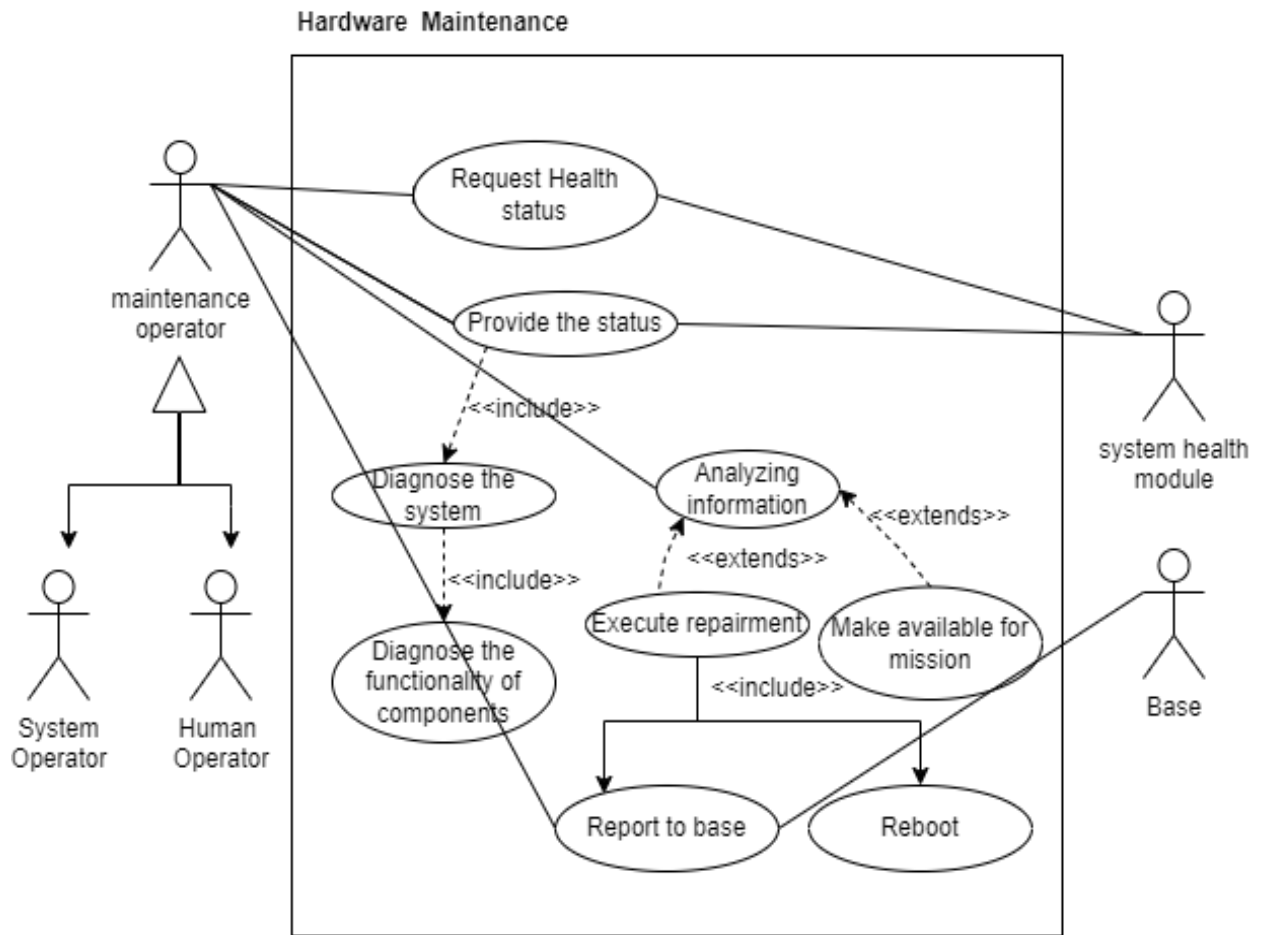

Figure 1.1: Use case diagram hardware maintenance

Figure 3.1: Use case diagram of Mobility
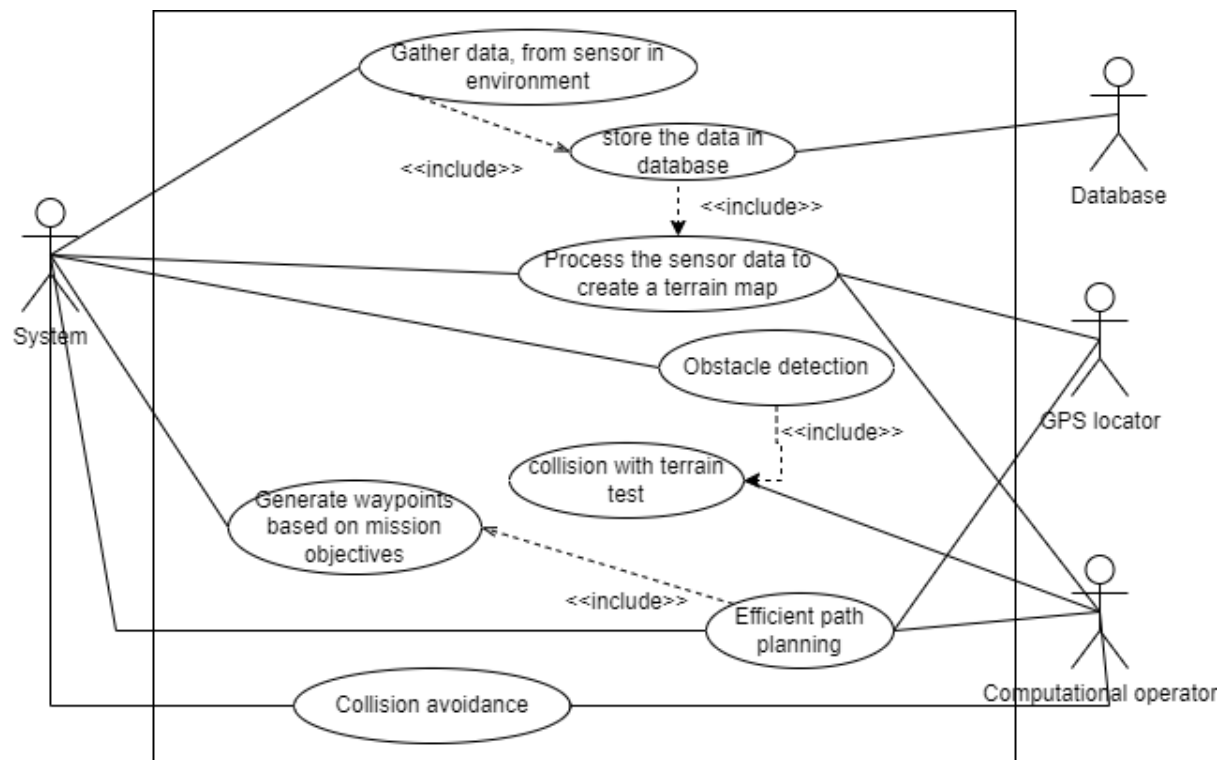
Figure 4.1: Use case diagram of Terrain scanning and way pointing.
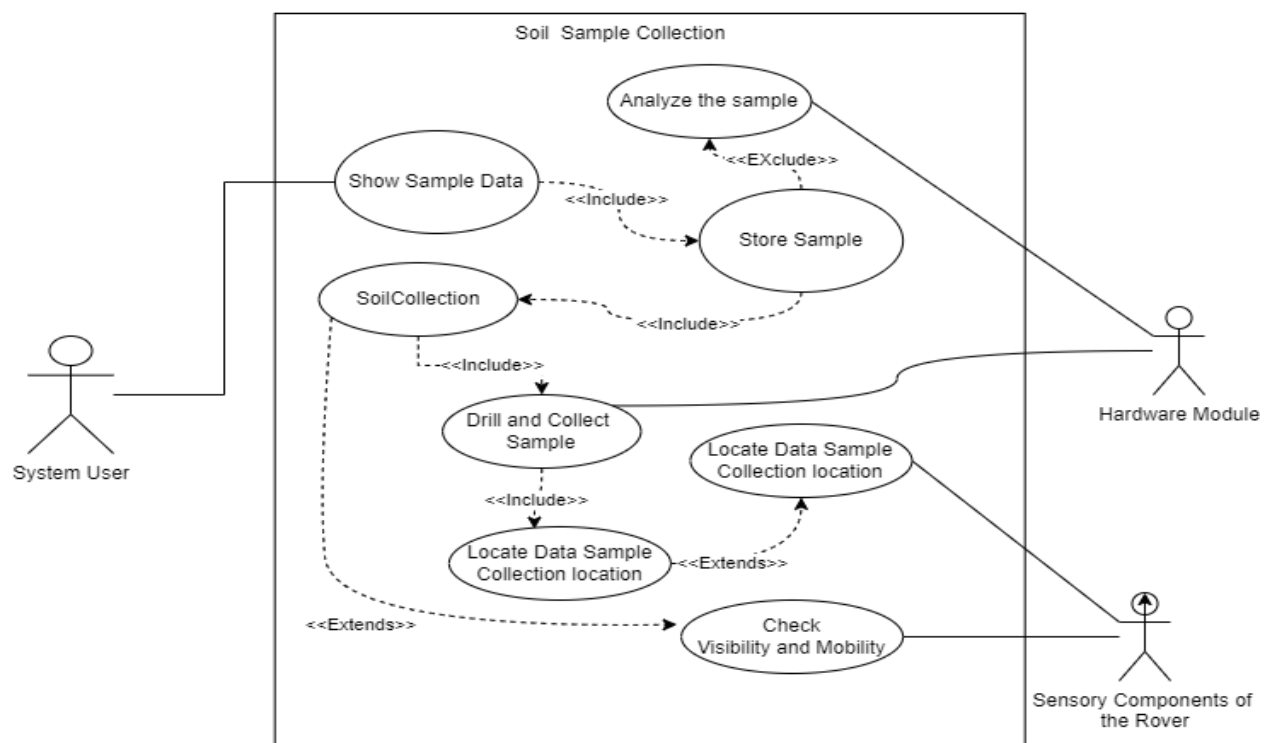
Figure 5.1: Use case diagram of Soil Sample Collection.
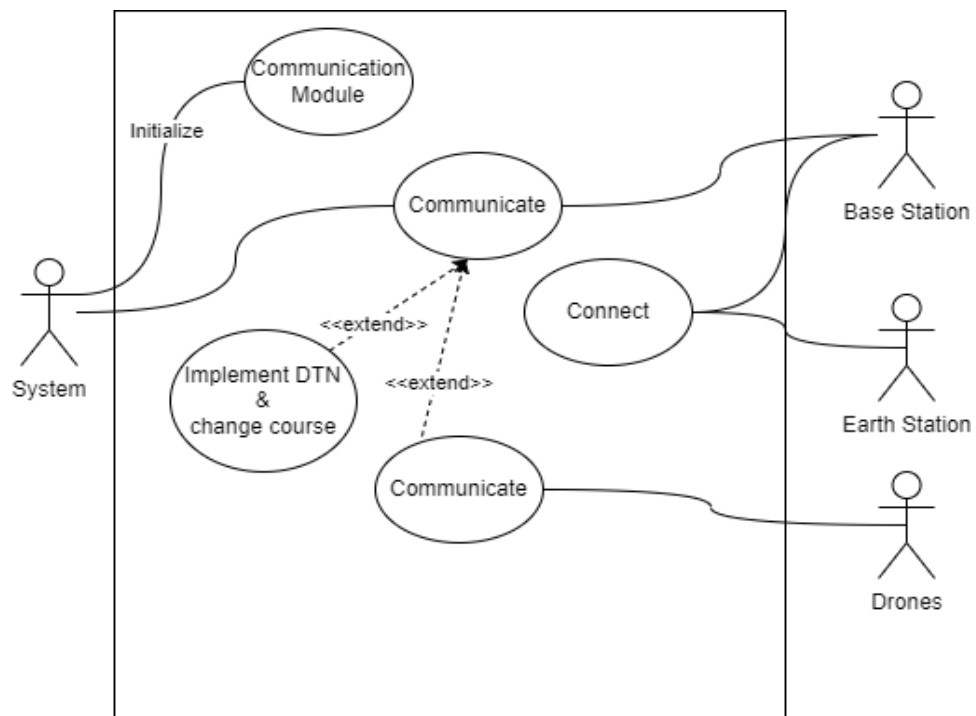
Figure 6.1: Use case diagram of Communication

Sequence diagram
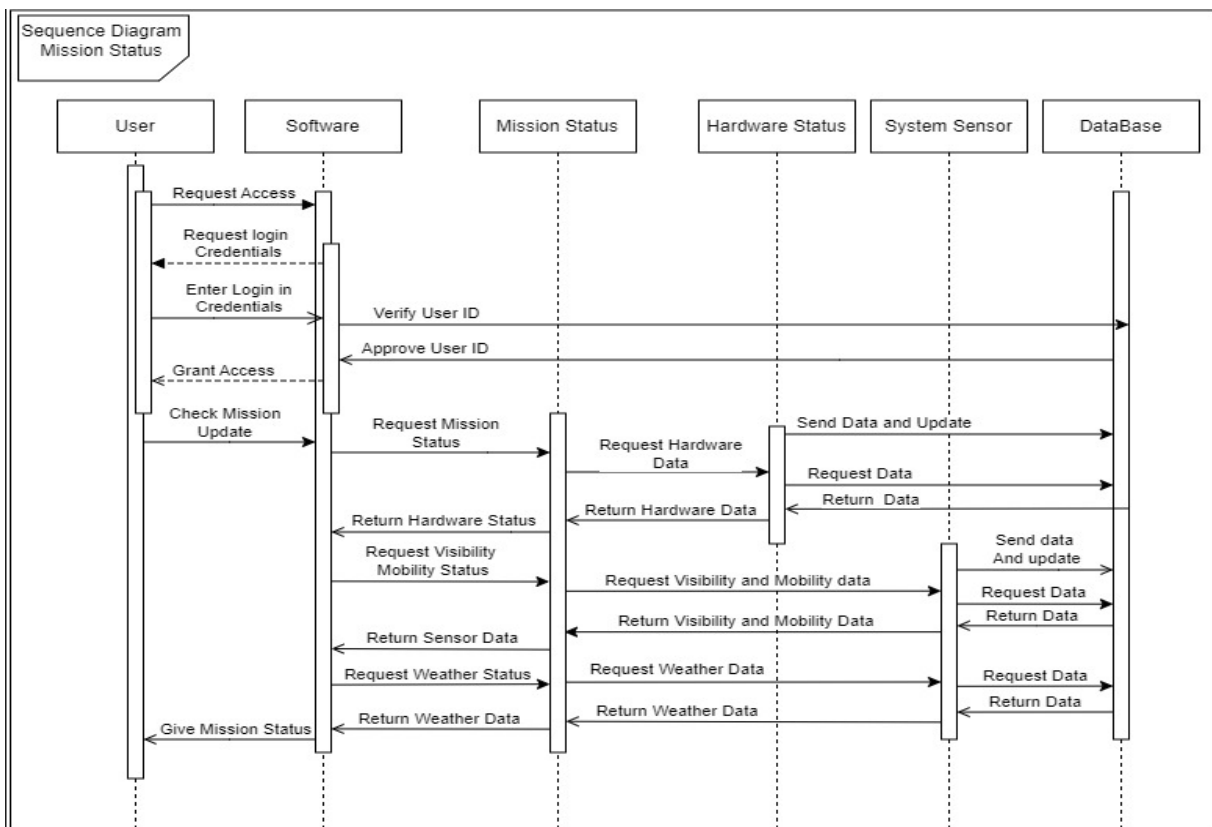
Figure 2.3: Sequence diagram of mission status

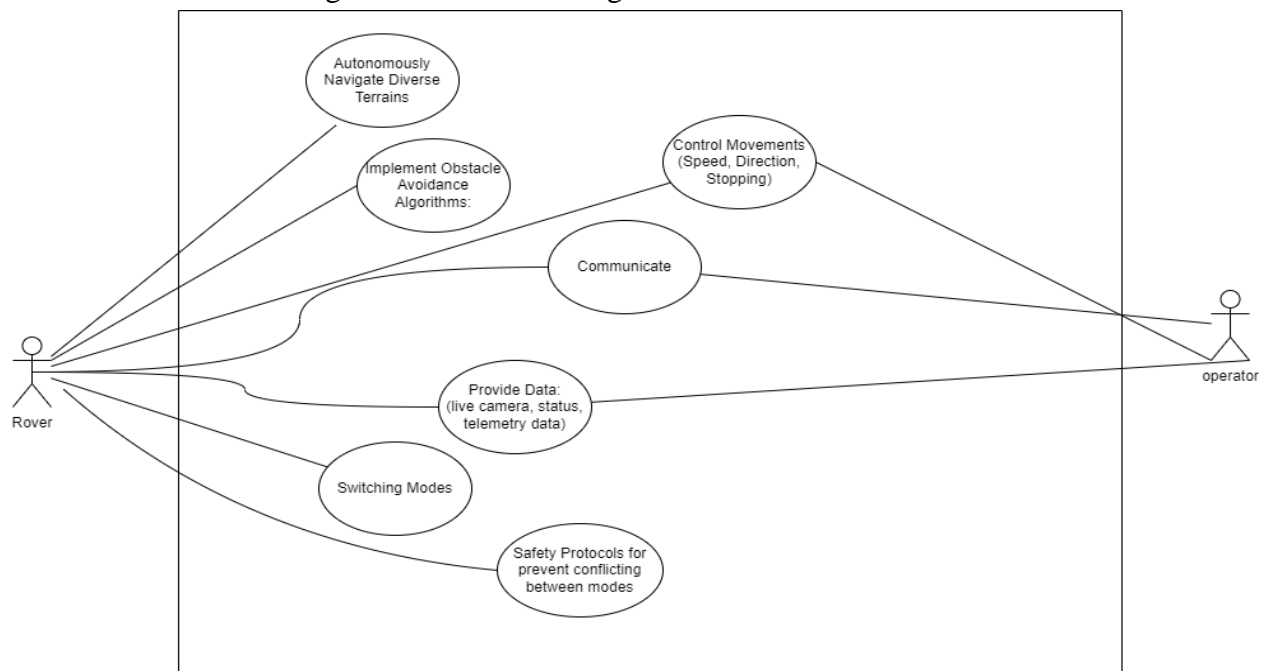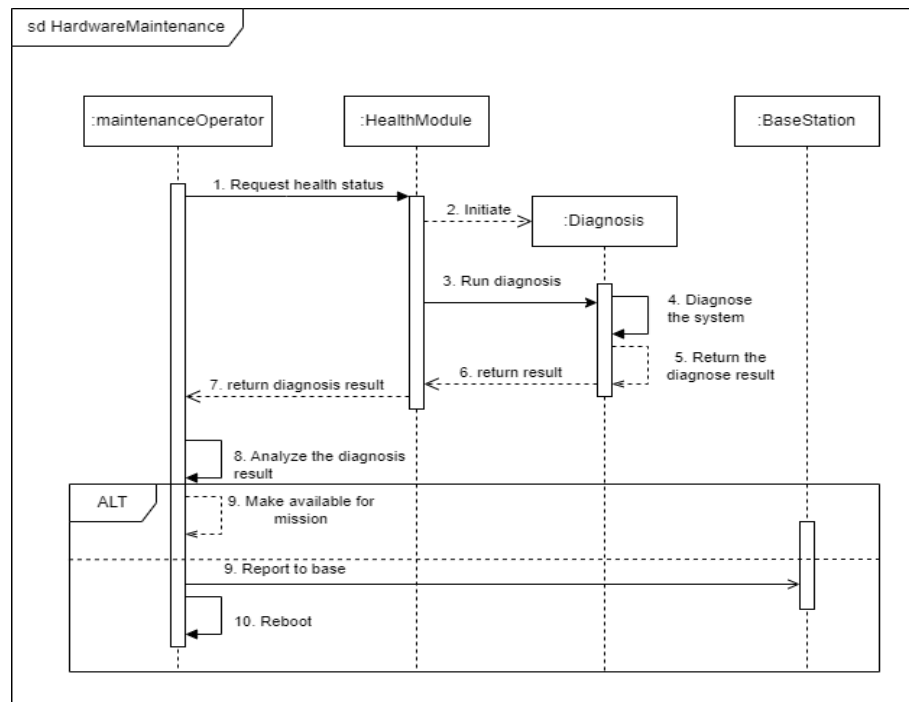Figure 1.3: Sequence diagram of hardware maintenance
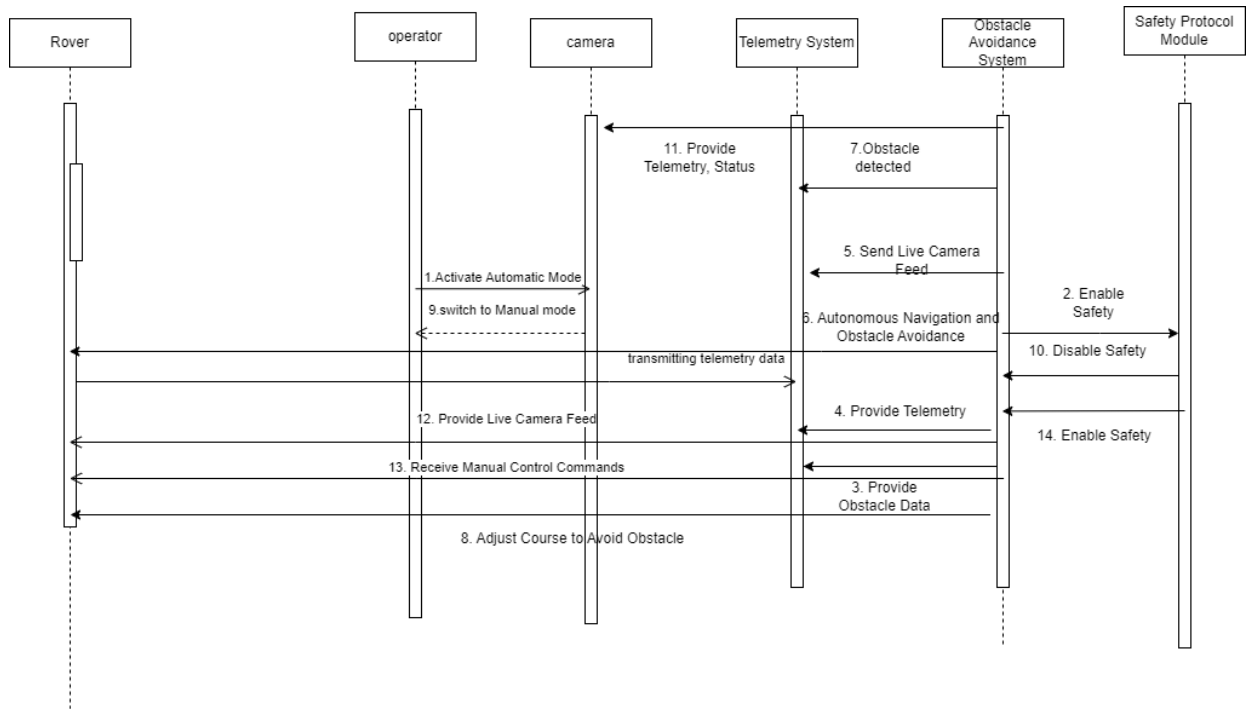
Figure 3.3: Sequence diagram of Mobility
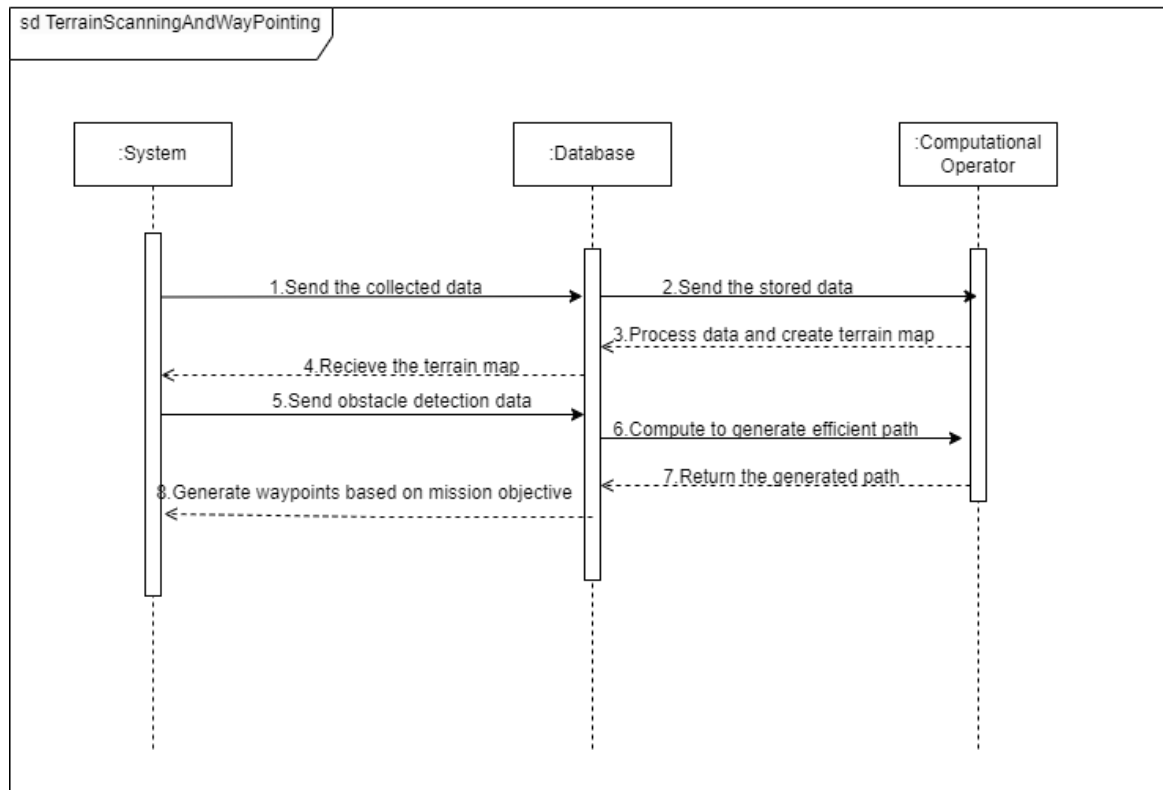
Figure 4.3: Sequence diagram of Terrain scanning and way pointing.
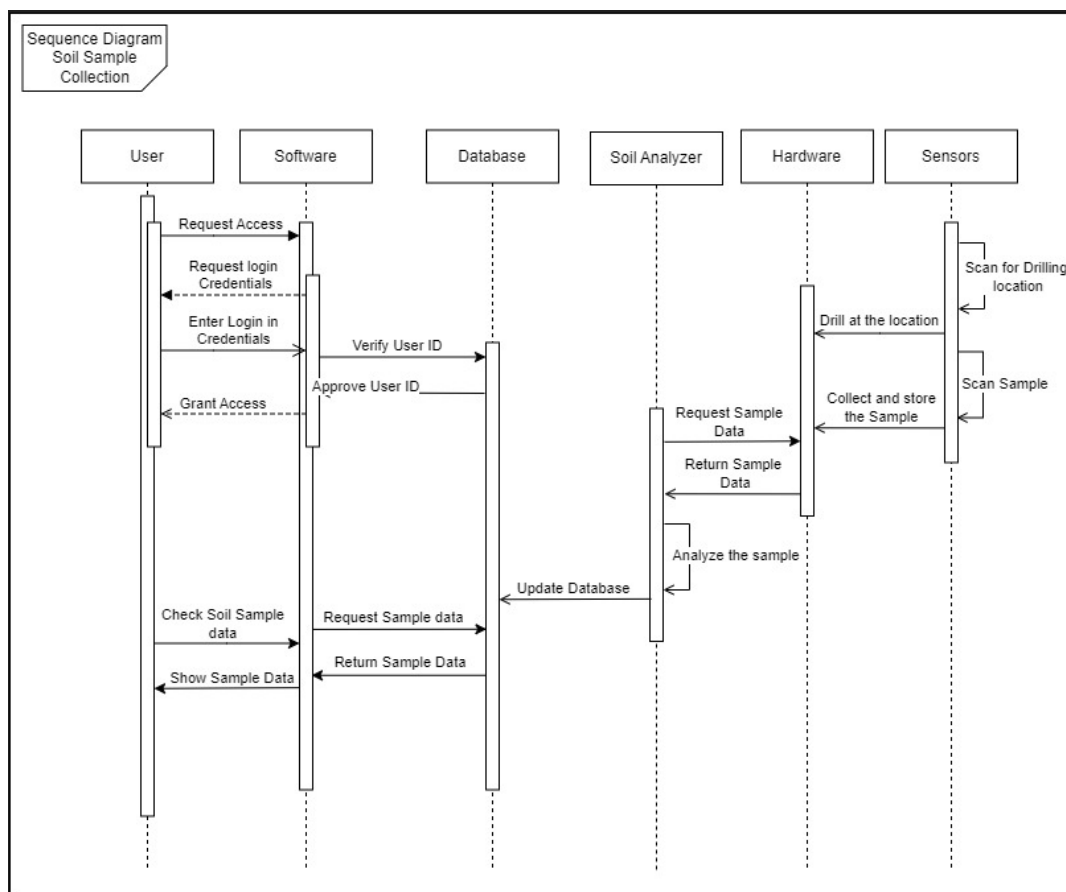
Figure 5.3: Sequence diagram of Soil Sample Collection.
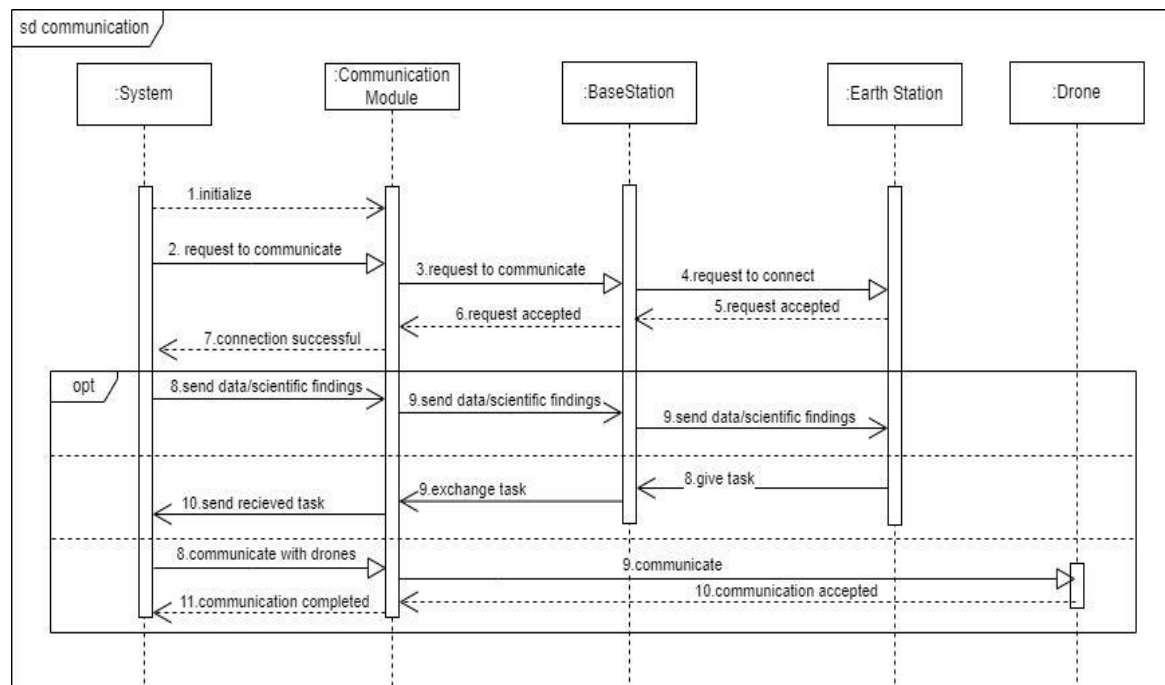
Figure 6.3: Sequence diagram of Communication

Activity diagram
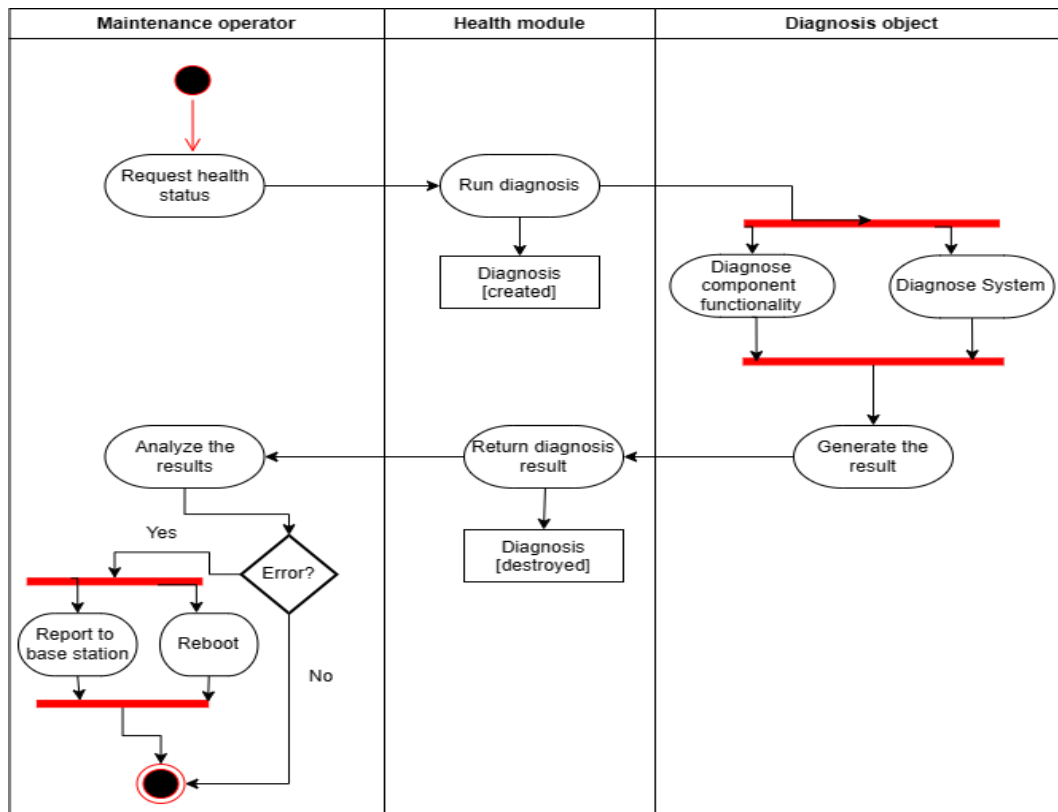
Figure 1.4: Activity diagram of hardware maintenance
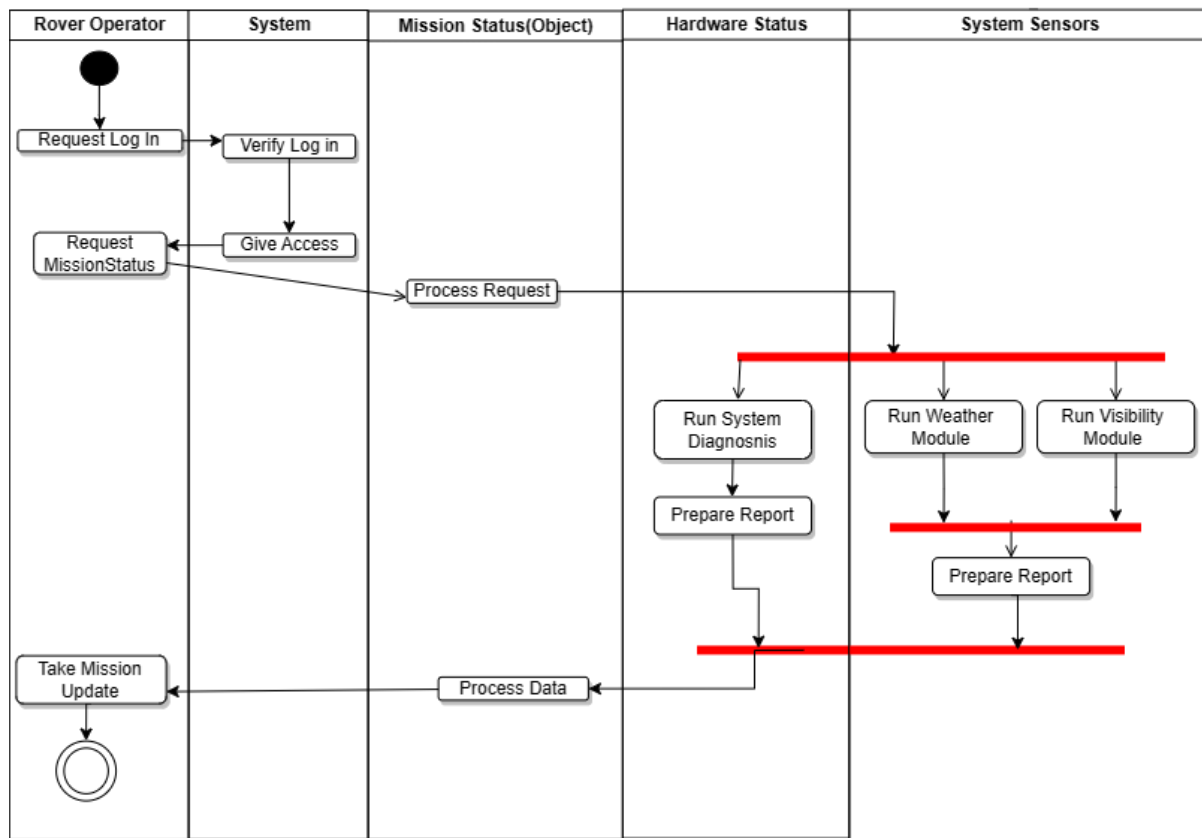
Figure 2.4: Activity diagram of mission status
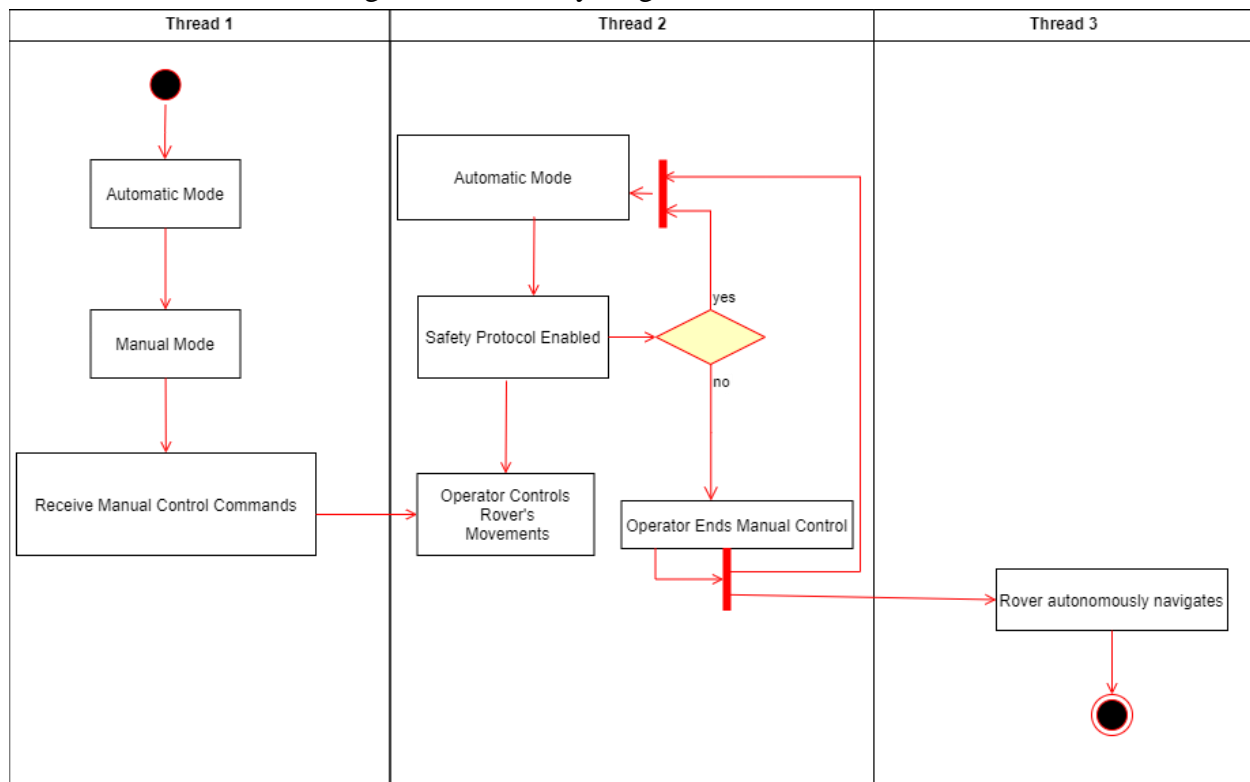
Figure 3.4: Activity diagram of Mobility
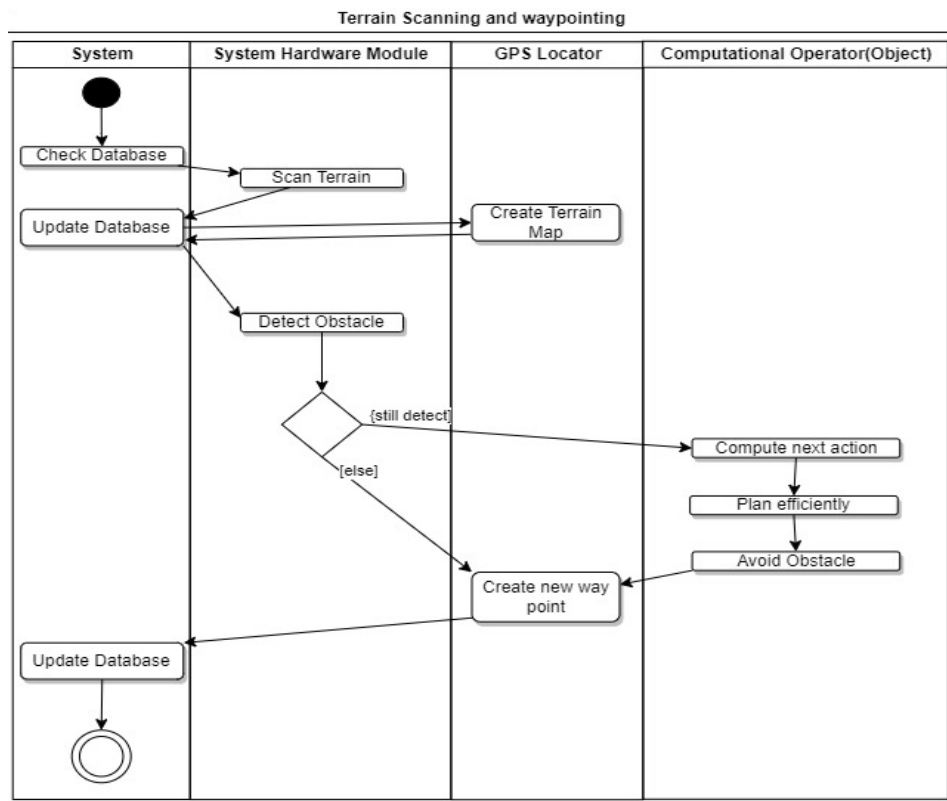
Figure 4.4: Activity diagram of Terrain scanning and way pointing.
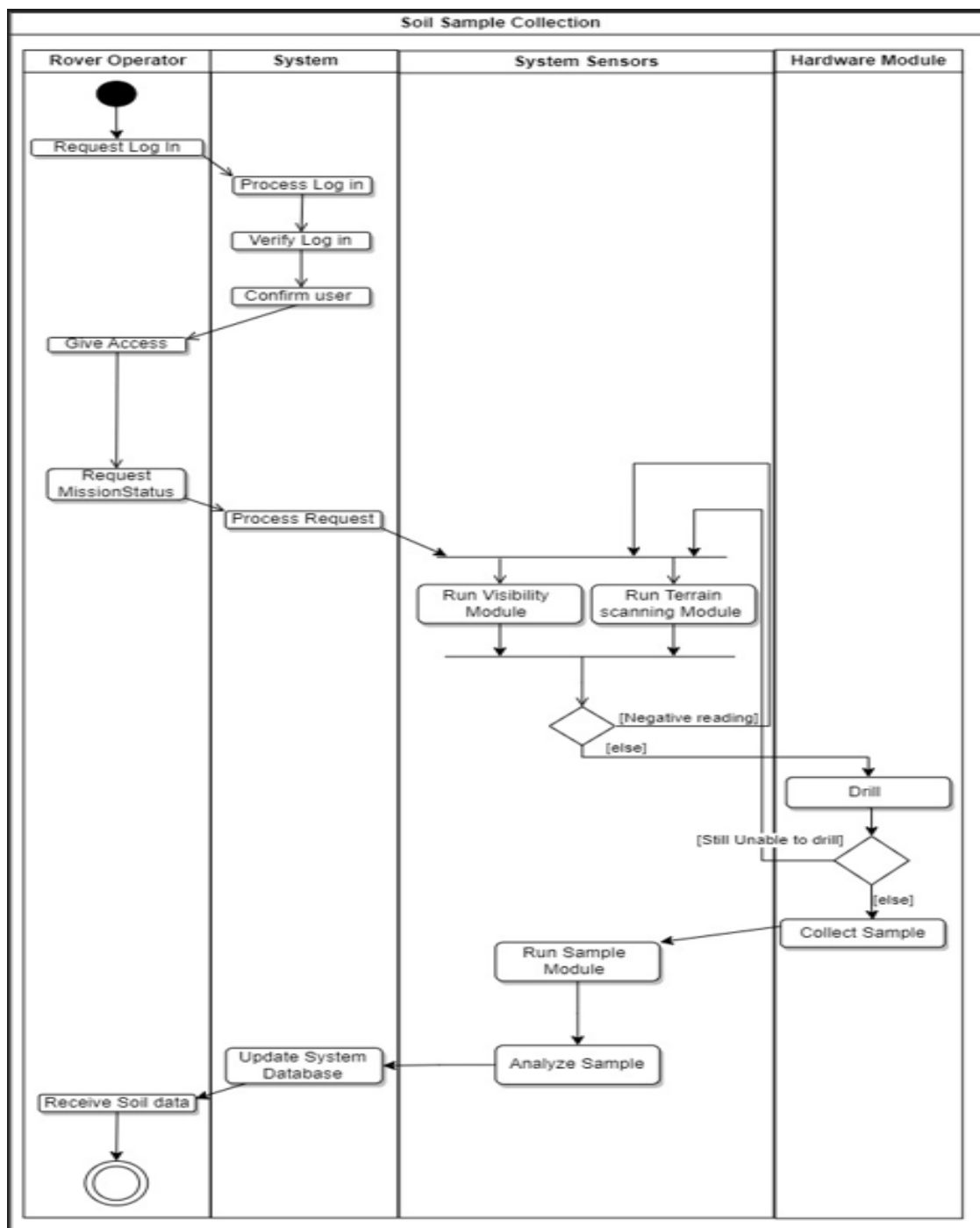
Figure 5.4: Activity diagram of Soil Sample Collection.
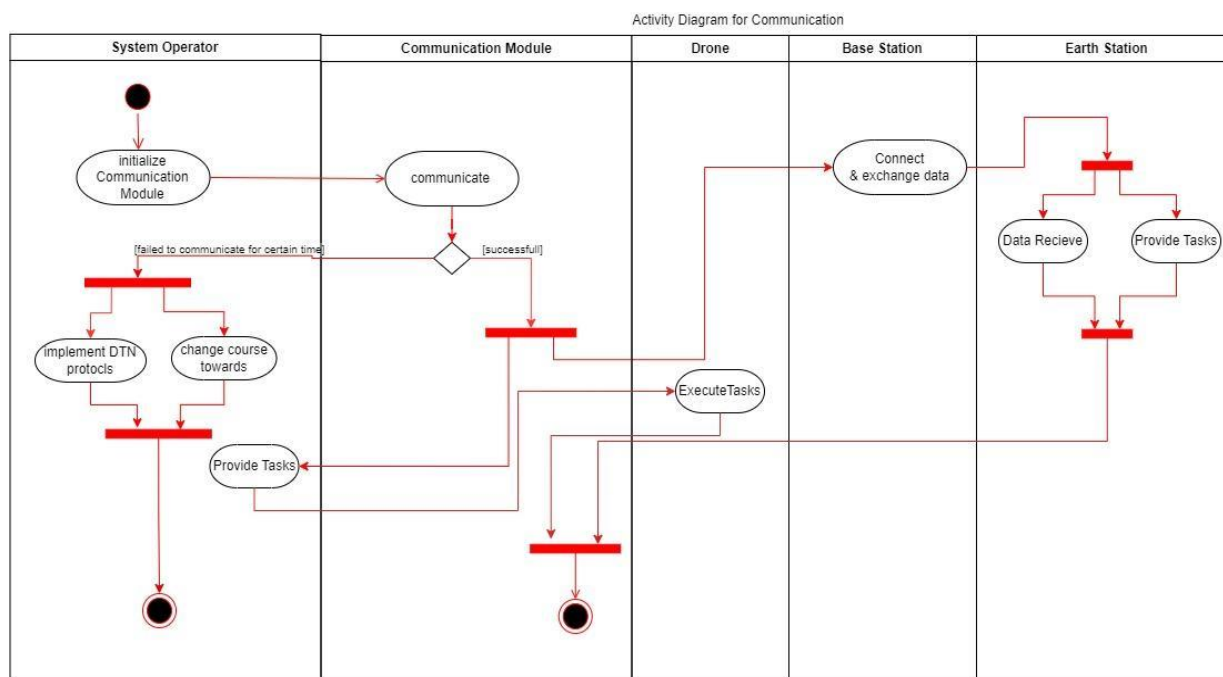
Figure 6.4: Activity diagram of Communication

Wireframe Design
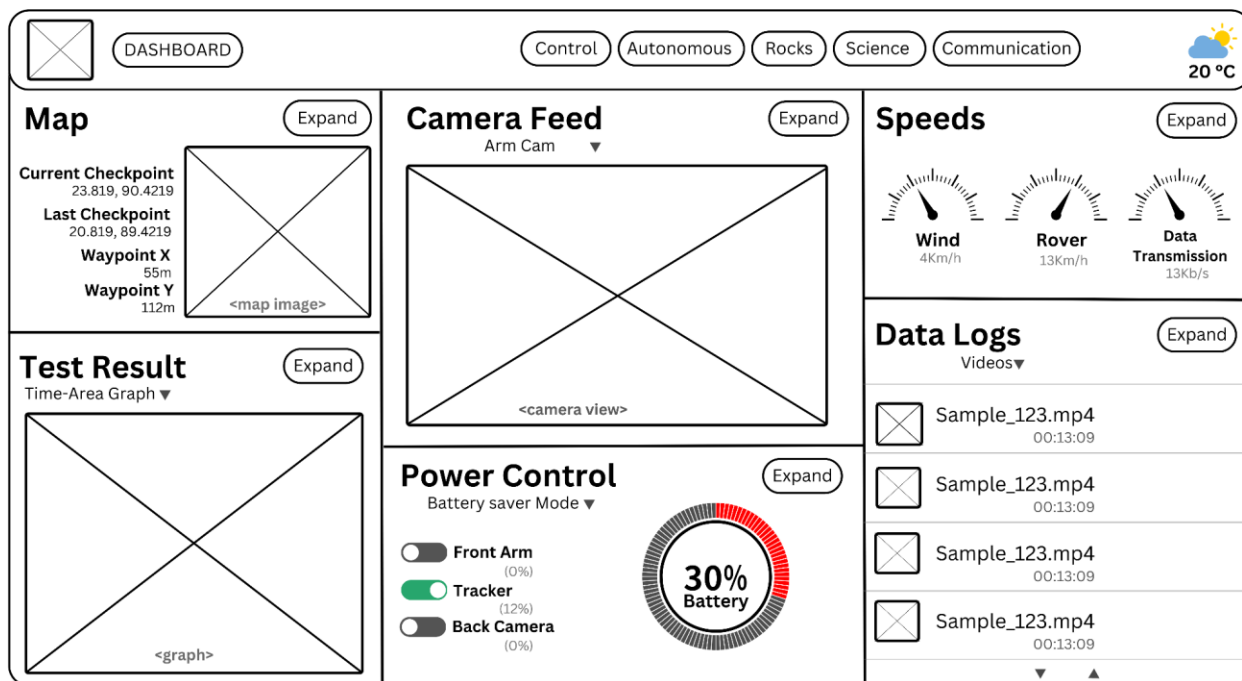
Figure 7.1: Wireframe Design of Dashboard
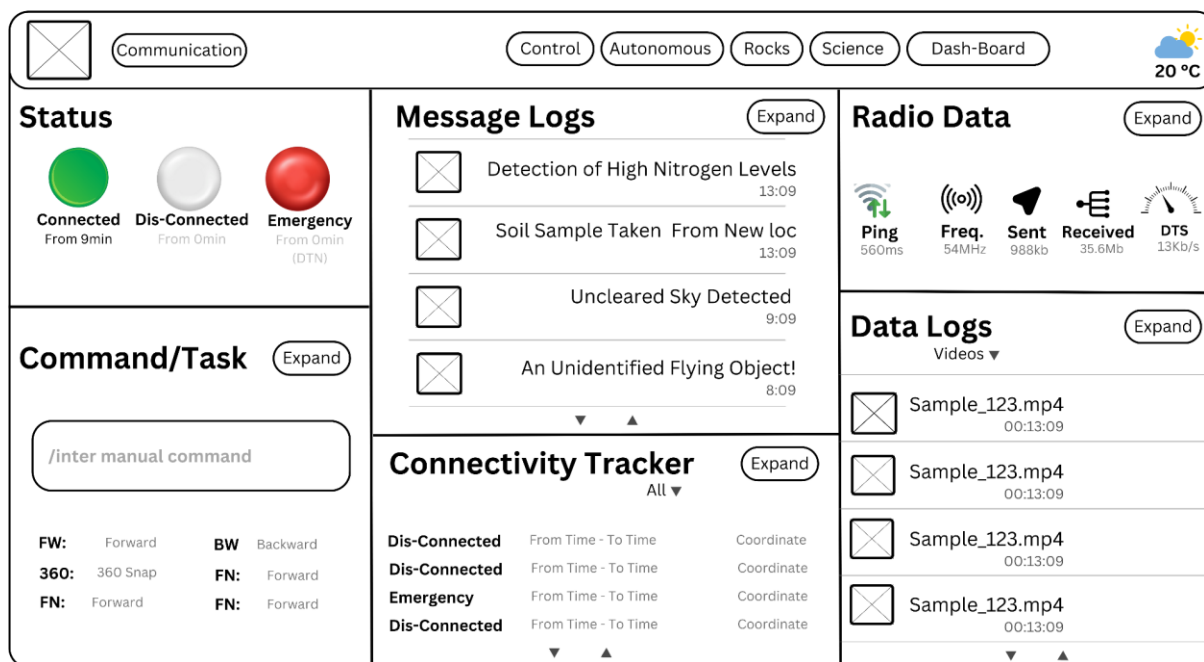
Figure 7.2: Wireframe Design of Communication
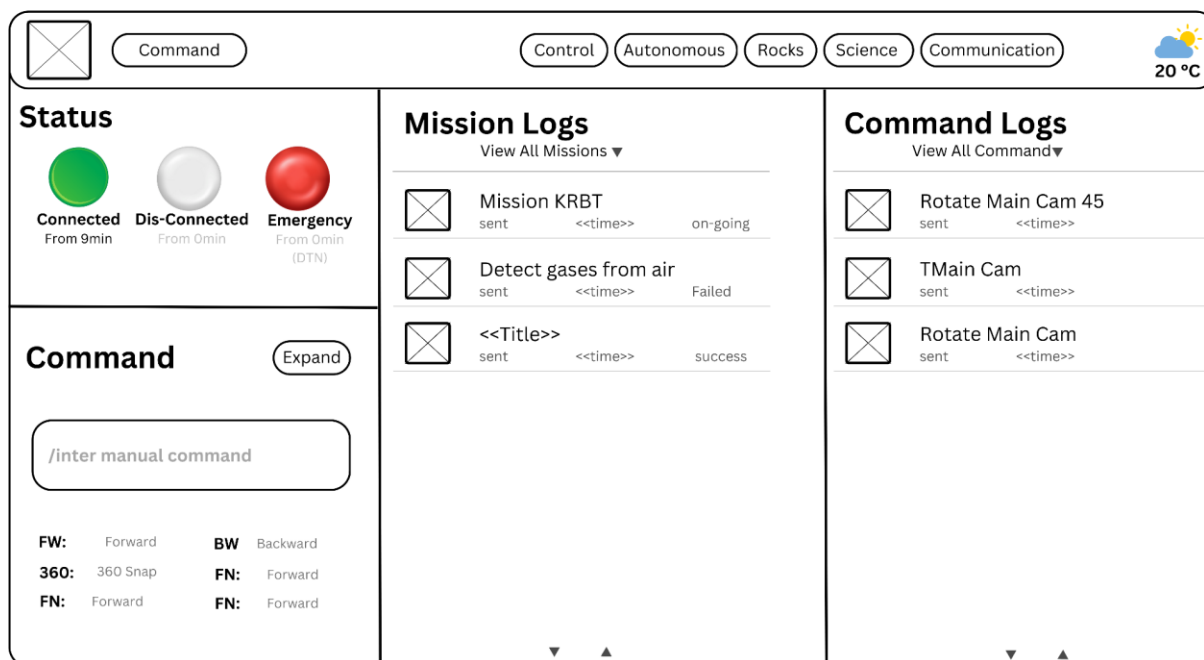
Figure 7.3: Wireframe Design of "Expanded View of Command/Tasks"
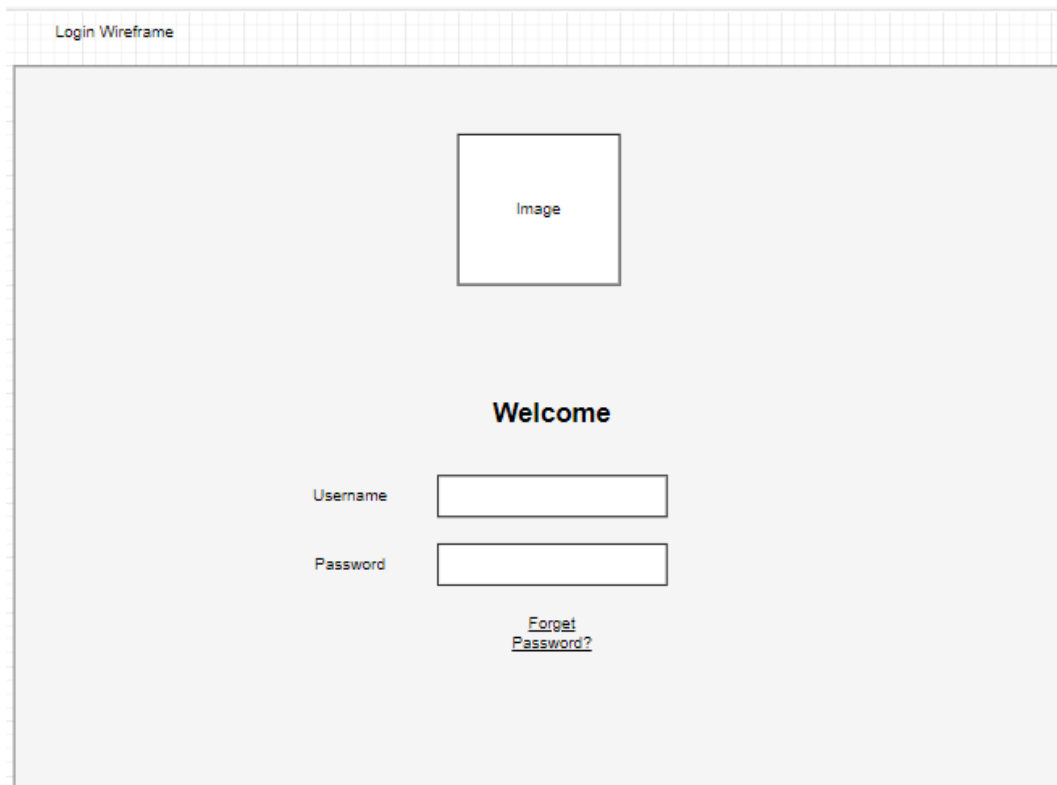
Figure 7.4: Wireframe Design of LogIn
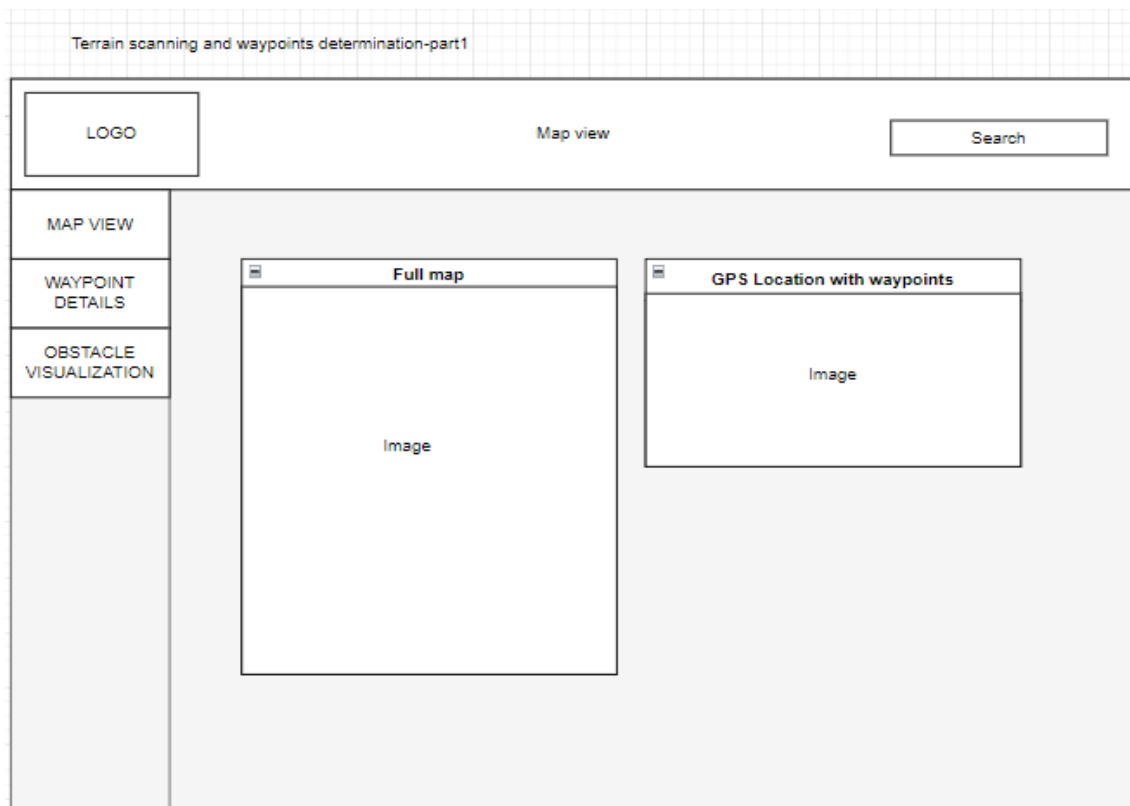
Figure 7.5: Wireframe Design of "Map-View"
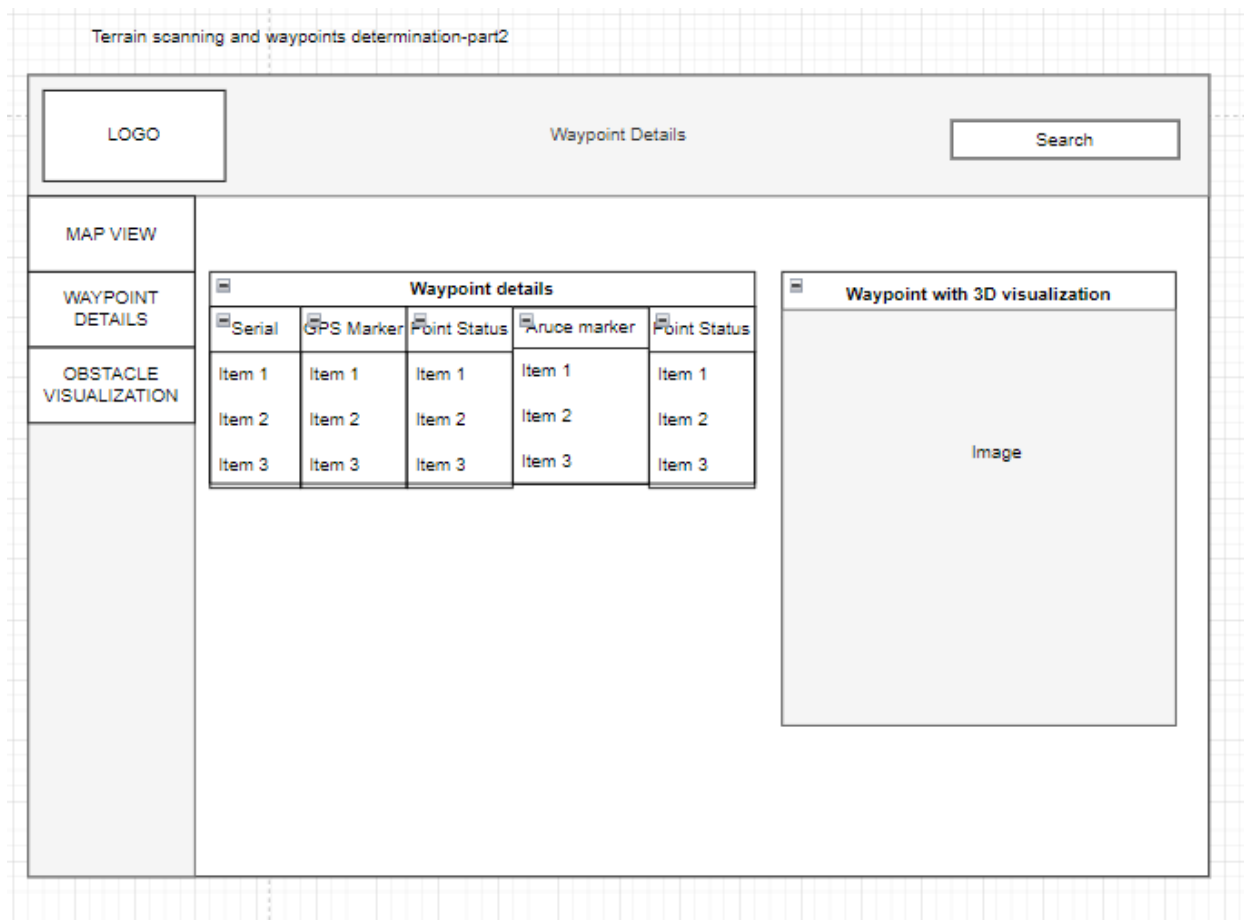
Figure 7.6: Wireframe Design of "Waypoint Details"
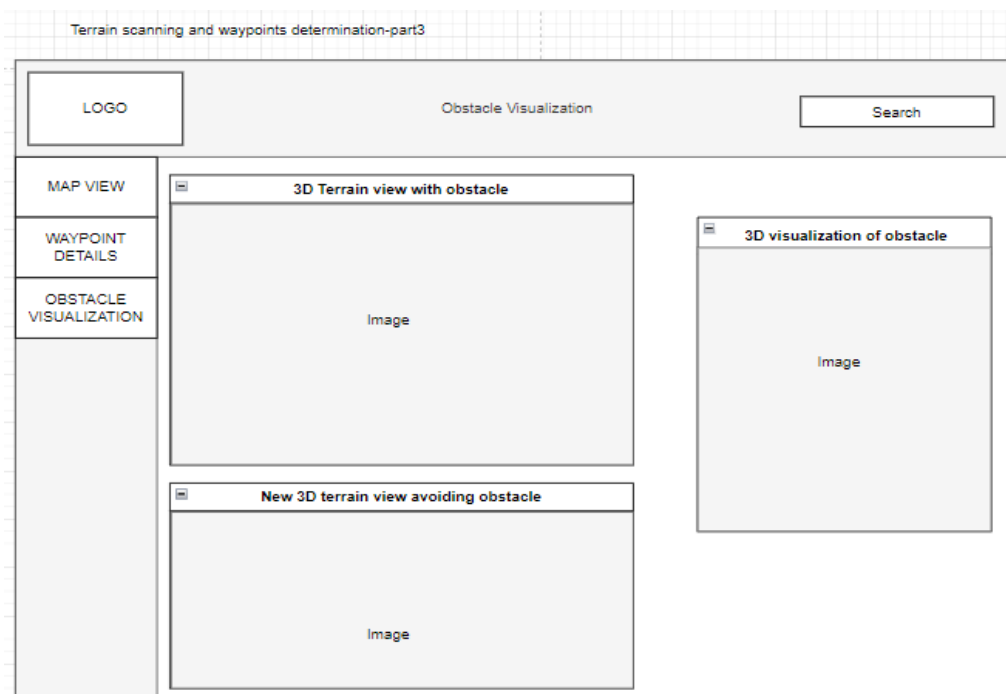
Figure 7.7: Wireframe Design of Dashboard

Figure 7.8: Wireframe Design of "Mission Status"
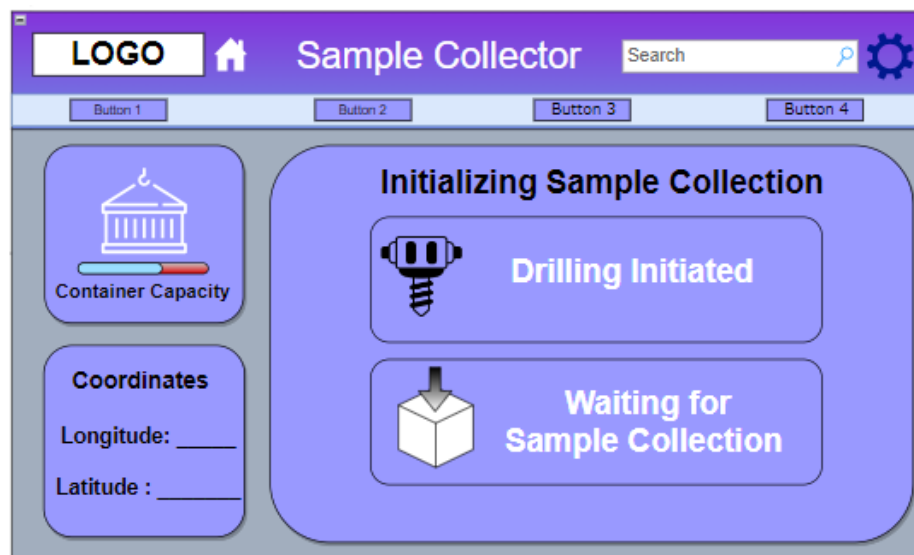
Figure 7.9: Wireframe Design of "Sample Collector"
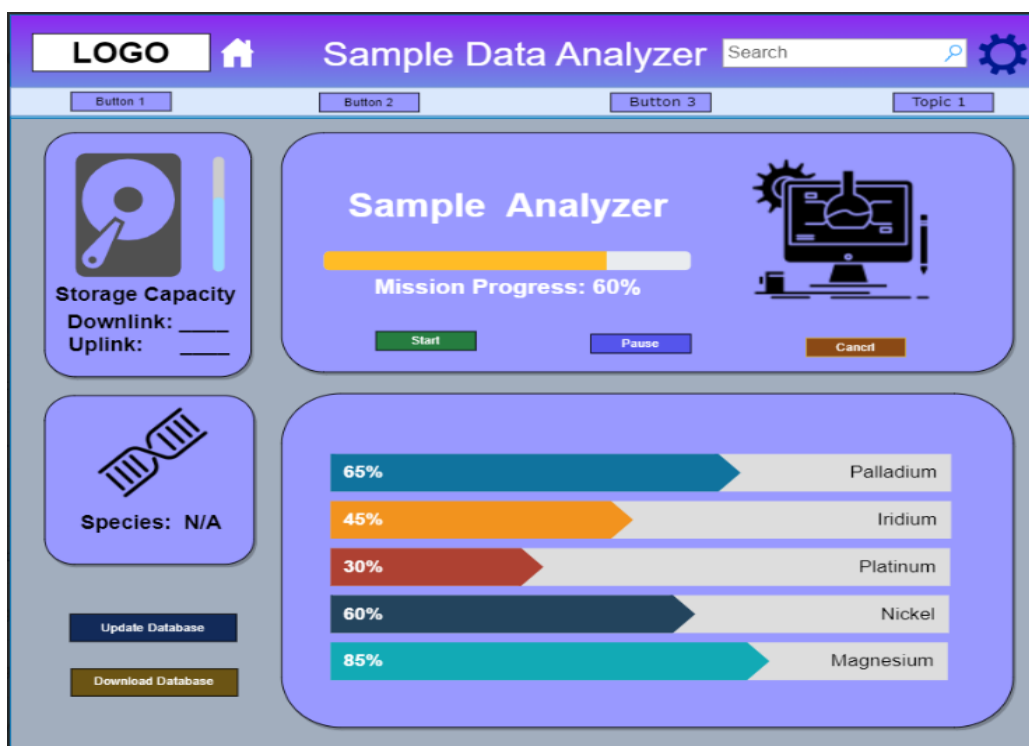
Figure 7.10: Wireframe Design of "Sample Analyzer"
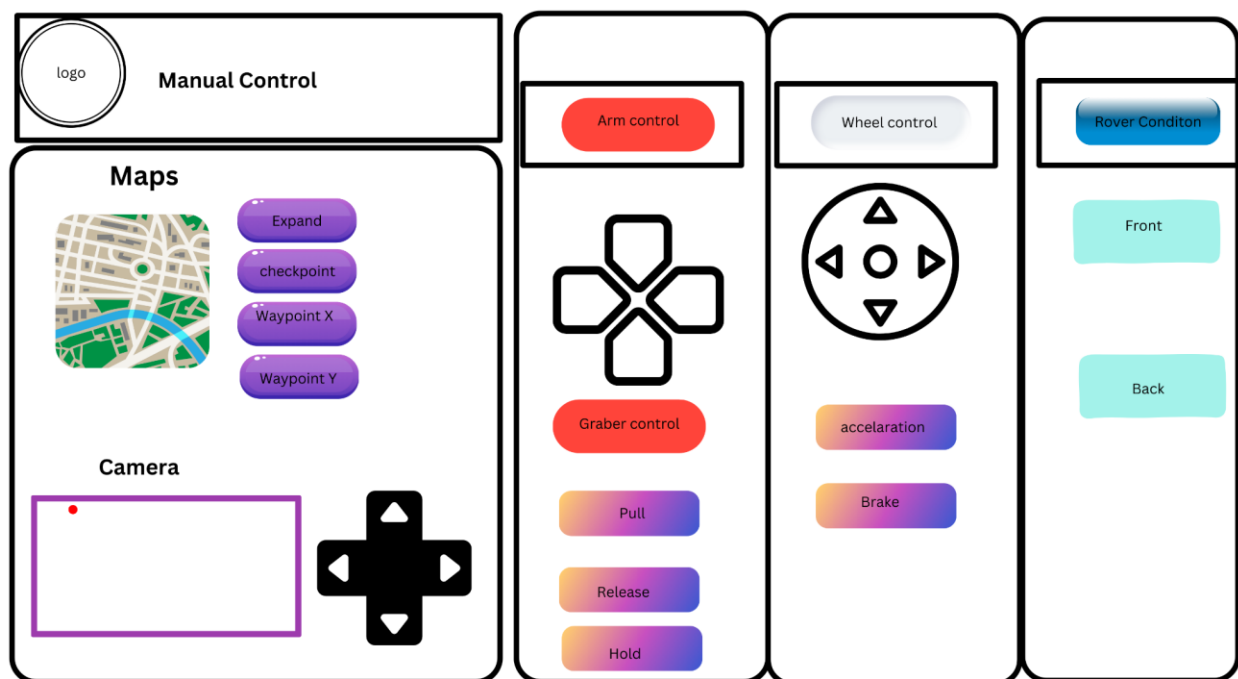
Figure 7.11: Wireframe Design of "Sample Collector"
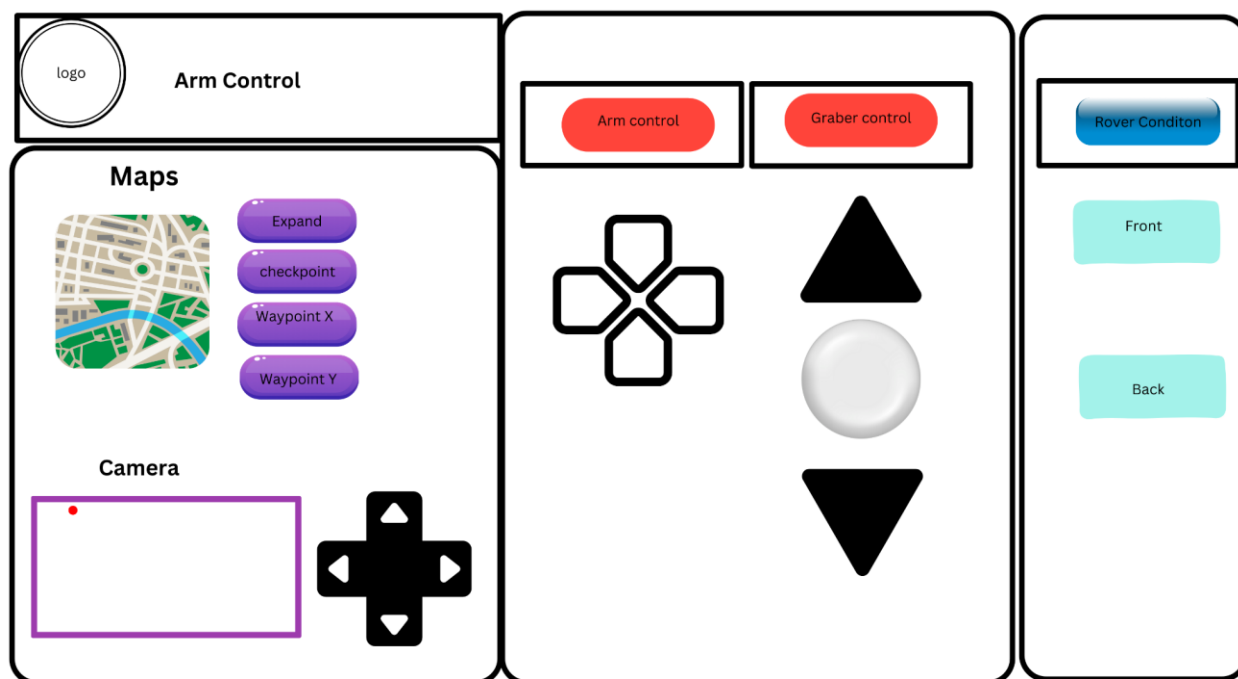
Figure 7.12: Wireframe Design of "Sample Collector"
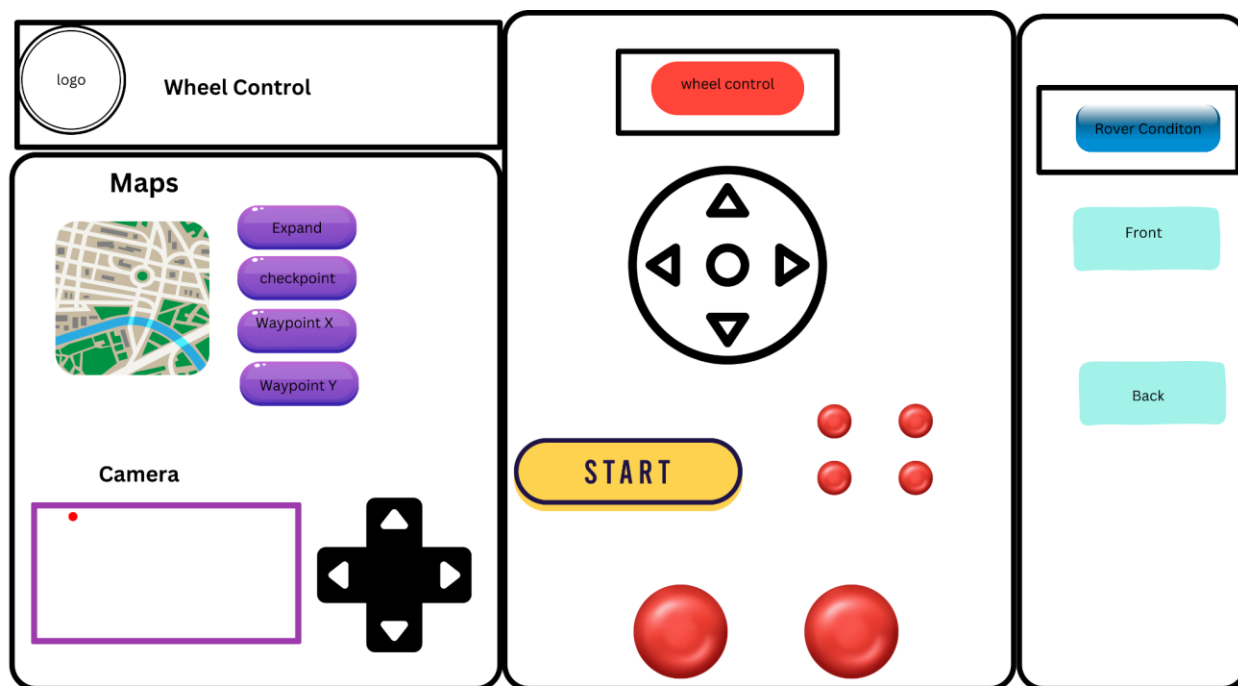
Figure 7.13: Wireframe Design of "Sample Collector"
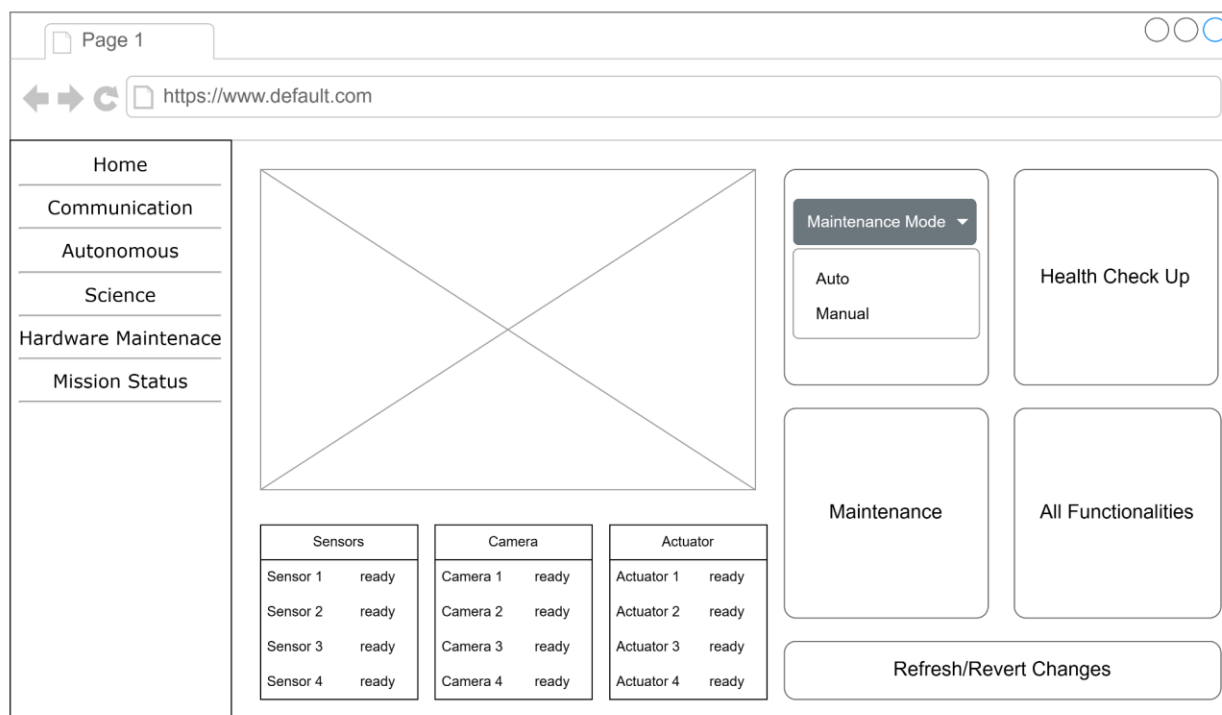
Figure 7.14: Wireframe Design of Hardware Maintenance
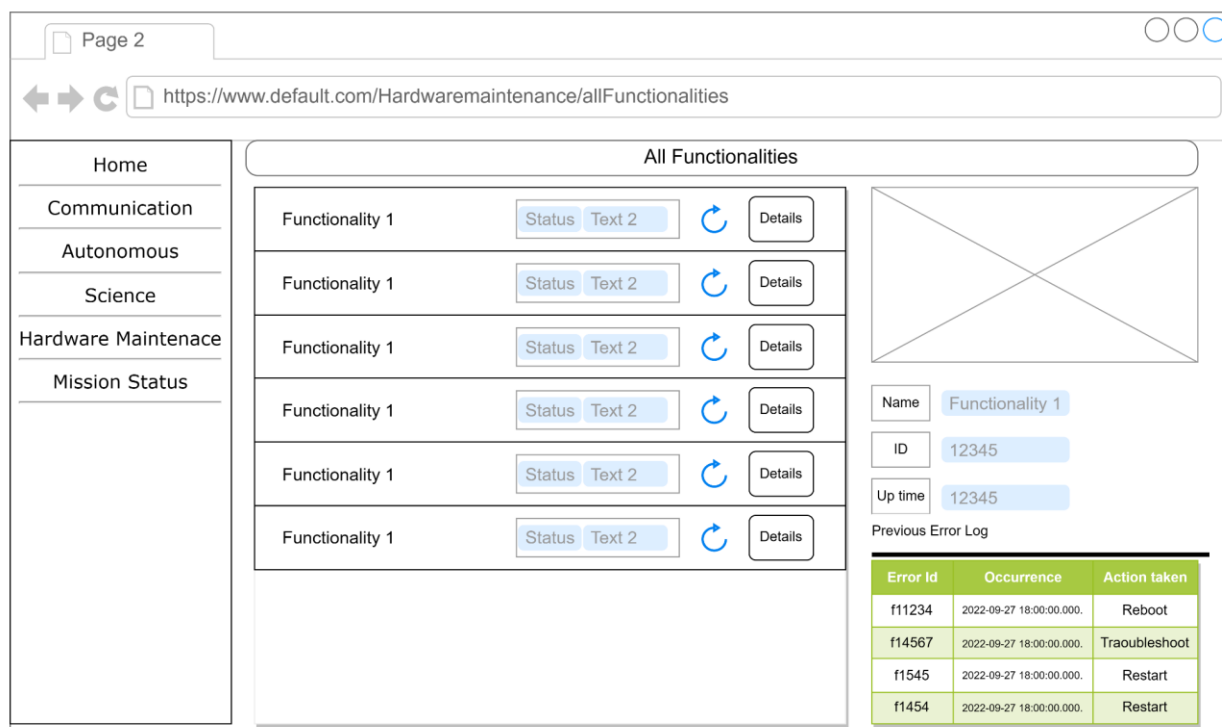
Figure 7.15: Wireframe Design of Hardware-Maintenance/All-functionalities

Figure 7.16: Wireframe Design of Hardware-Maintenance/Maintenance

Test Case Table

Project Name: Space Rover Application	Test Designed by: Sajin, Md. Imtiaj Alam
Test Case ID: COMM_01	Test Designed date: 16/11/2023
Test Priority (Low, Medium, High): High	Test Executed by: Sajin, Md. Imtiaj Alam
Module Name: Communication Module	Test Execution date: 18/11/2023
Test Title: **Verify Initialization of Communication Module**	
Description: Test the initialization of the communication module.	
Precondition (If any): The rover is in any state.	

Test Steps	Test Data	Expected Results	Actual Results	Status
1. Power on the system. 2. Use the interface to send a basic command. 3. Verify the rover receives and responds appropriately.	Command: Forward	Rover responds to the command without errors.	As expected	Pass
Post Condition: The communication module is ready for further commands.				

Project Name: Space Rover Application	Test Designed by: Sajin, Md. Imtiaj Alam
Test Case ID: COMM_02	Test Designed date: 16/11/2023
Test Priority (Low, Medium, High): High	Test Executed by: Sajin, Md. Imtiaj Alam
Module Name: Communication Module	Test Execution date: 18/11/2023
Test Title: **Verify Real Time Data Streaming Mechanism**	
Description: Test the data streaming mechanism between the rover and Earth station.	
Precondition (If any): Communication Module Initialized	

Test Steps	Test Data	Expected Results	Actual Results	Status

Test Steps	Test Data	Expected Results	Actual Results	Status
1. Open the software interface 2. Use the interface to start any camera . 3. Verify the rover broadcasted the live view.	Command: Open Front Cam	Rover responds to the command and consistently streams the real time view.	As expected	Pass

Post Condition: The rover is capable of real-time data streaming for monitoring and control.

Project Name: Space Rover Application	Test Designed by: Sajin, Md. Imtiaj Alam
Test Case ID: COMM_03	Test Designed date: 16/11/2023
Test Priority (Low, Medium, High): Medium	Test Executed by: Sajin, Md. Imtiaj Alam
Module Name: Communication Module	Test Execution date: 18/11/2023

Test Title: **Verify Handling of Communication Failures**

Description: Test the rover's behavior when facing communication failures with the Base station.

Precondition : Communication Module Initialized

Test Steps	**Test Data**	**Expected Results**	**Actual Results**	**Status**

Test Steps	Test Data	Expected Results	Results	Status
1. Simulate a communication failure with the Base station.(e.g., disconnect network cable) 2. Observe how the system handles the communication failure.	N/A	1. Recognizes the failure and initiates the appropriate response (e.g., logging, alerting). 2. Changes the course towards the last communication location and implements DTN protocols.	As expected	Pass

Post Condition: Rover handles communication failure and resumes normal state after resolved.

Project Name: Space Rover Application	**Test Designed by**: Susham Moula Choudhury Akash
Test Case ID:MISS_01	**Test Designed date**: 15/11/2023
Test Priority: High	**Test Executed by**: Susham Moula Choudhury Akash
Module Name: Mission Status Module	**Test Execution date**: 17/11/2023

Test Title: Verify Real Time Mission Progress Update

Precondition : Mission Status Initialized

Test Steps	Test Data	Expected Results	Results	Status
1: Check mission progress	Estimate and Update time for completion	Rover responds to the command and consistently streams the real time data update.	As expected,	Pass

Post Condition: The user interface shows real-time updates on the mission progress and provides an estimated time for completion based on current conditions.

Project Name: Space Rover Application	**Test Designed by:** Susham Moula Choudhury Akash
Test Case ID:MISS_02	**Test Designed date**: 15/11/2023
Test Priority: Medium	**Test Executed by:** Susham Moula Choudhury Akash
Module Name: Mission Status Module	**Test Execution date**: 17/11/2023

Test Title: Verify Real Time Mission Status Update

Precondition : Mission Sensors Detection Initialized

Test Steps	Test Data	Expected Results	Results	Status
1. Activate the weather sensors and initiate data collection. 2. Activate the Hardware sensors and initiate data collection. 2. Update the database and display	1. Simulate the sensors and initiate data collection.	Rover responds to command and updates the database	As expected,	Pass

Description: The rover successfully and activates the dedicated sensors and begins measuring atmospheric pressure, weather, temperature.

Post Condition: Rover responds successfully activates the dedicated sensors and begins measuring atmospheric pressure, weather, temperature.

Project Name: Space Rover Application	Test Designed by: Susham Moula Choudhury Akash
Test Case ID:MISS_03	Test Designed date: 15/11/2023
Test Priority: Medium	Test Executed by: Susham Moula Choudhury Akash
Module Name: Mission Status Module	Test Execution date: 17/11/2023

Test Title: Check Vehicle Status

Precondition: Mission Status Initialized

Test Steps	Test Data	Expected Results	Results	Status
1.Check Tire pressure 2.Check Battery Health, 3.Update Battery percentage 4.Update mission progress	1. Check vehicle status	Rover responds to the command and checks vehicle status	As expected,	Pass

Description: Rover responds to the command and displays accurate information about the vehicle's status, tire pressure, battery health, and percentage.

Post Condition: Rover responds to the command and displays accurate information about the vehicle's status, tire pressure, battery health, and percentage.

Project Name: Space Rover Application	Test Designed by: Susham Moula Choudhury Akash
Test Case ID:SOILC_01	Test Designed date: 15/11/2023
Test Priority: High	Test Executed by: Susham Moula Choudhury Akash
Module Name: Soil Sample Collection Module	Test Execution date: 17/11/2023

Test Title: Search for a suitable location.

Precondition: Search for Suitable

Test Steps	Test Data	Expected Results	Results	Status
1.Initiate the search for a suitable location. 2. Scan the surface for feasibility. 3.Relocate to a new location if necessary.	1.Scan for Drill spot and soil collection 2. Scan for collectable space materials on the ground	The rover successfully scanned for suitable location	As expected,	Pass

Description: The rover navigates and scans the surroundings to identify a suitable location for soil sample collection. The system analyzes the surface conditions to determine if sample collection is possible. If not feasible, the rover should initiate relocation to find a better location.

Post Condition: Rover responds to the command and navigates and scans the surroundings to identify a suitable location for soil sample collection.

Project Name: Space Rover Application	Test Designed by: Susham Moula Choudhury Akash
Test Case ID:SOILC_02	Test Designed date: 15/11/2023

Test Priority: Medium	**Test Executed by:** Susham Moula Choudhury Akash
Module Name: Soil Sample Collection Module	**Test Execution date:** 17/11/2023

Test Title: Sample Collection

Precondition: Initiate drilling process and collect sample

Test Steps	Test Data	Expected Results	Results	Status
1.Initiate drilling process using drilling mechanism 2. Collect the soil sample. 3.Store the samples for further analysis.	1.Drill spot to a specific depth 2. Collect the sample 3. Store the sample in the container	The rover successfully responds to the command and collects the sample after drilling	As expected,	Pass

Description: The rover uses its drilling mechanism to reach the specified depth in the soil. The rover collects a sample from the drilled location and ensures it is securely stored within designated containers. The collected soil samples are properly stored in containers, ready for further analysis.

Post Condition: Rover collects the sample after drilling

Project Name: Space Rover Application	**Test Designed by:** Susham Moula Choudhury Akash
Test Case ID:SOILC_03	**Test Designed date**: 15/11/2023
Test Priority: Medium	**Test Executed by:** Susham Moula Choudhury Akash
Module Name: Soil Sample Collection Module	**Test Execution date:** 17/11/2023

Test Title: Sample Data Analyzer

Precondition: Initial Sample Analyzer

Test Steps	Test Data	Expected Results	Results	Status
1.Initiate Analysis 2. Send the report to homebase 3.Reset to exploration mode	1.Start Analysis and update the database	The rover successfully responds to the command and Analyzes the Sample	As expected,	Pass

Description: The rover Analyzes and updates the database After the soil sample collection mission is completed, the rover resets to its exploration mode, ready for the next task

Post Condition: Rover collects the sample after drilling

Project Name: Space Rover Application	**Test Designed by:** Esm-e Moula Chowdhury Abha
Test Case ID: TR001	**Test Designed date:**18.11.23
Test Priority: High	**Test Executed by:** Esm-e Moula Chowdhury Abha
Module Name: Terrain Scanning and Way Pointing	**Test Execution date:** 18.11.23

Test Title: Terrain Data Collection

Description: Terrain data collection for Mars exploration mission, ensuring accurate data gathering by onboard sensors.

Precondition: The rover, equipped for Mars exploration, is engaged in Mission Red Planet.

Test Steps	Test Data	Expected Results	Actual Results	Status
a. Engage the rover in Mission Red Planet. b. Verify that onboard sensors, including Mars-specific cameras and radar, are actively collecting Martian terrain data. c. Check if the collected Martian terrain data is within the expected range and format for Mars exploration. d. Confirm that the rover is transmitting the collected Martian terrain data to the processing algorithms.	Rover engaged in Mission Red Planet.	Martian terrain data is collected accurately.	As expected,	Pass

Post Condition: The rover is prepared with accurate Martian terrain data for further processing and analysis in the mission

Project Name: Space Rover Application	**Test Designed by:** Esm-e Moula Chowdhury Abha
Test Case ID: TR002	**Test Designed date:**18.11.23
Test Priority: High	**Test Executed by:** Esm-e Moula Chowdhury Abha
Module Name: Terrain Scanning and Way Pointing	**Test Execution date:** 18.11.23
Test Title: Terrain Map Generation for Martian Exploration	
Description: Martian terrain map generation for Mars exploration mission, validating effective processing and obstacle detection algorithms.	

Precondition (If any): The rover, configured for Martian exploration, is engaged in Mission Moon. Terrain data is collected successfully.

Test Steps	Test Data	Expected Results	Actual Results	Status
a.Validate that the algorithms process the Martian sensor data effectively to create a detailed Martian terrain map. b. Ensure that the Martian terrain map includes an accurate representation of Martian features and potential obstacles. c. Verify the accuracy of Martian obstacle detection by comparing with known Martian features.	Rover engaged in Mission Red Planet. Valid terrain data collected	The generated Martian terrain map includes detailed information about Martian features and detected obstacles.	As expected,	Pass

Post Condition: The rover is equipped with a detailed Martian terrain map, crucial for planning and executing the Martian exploration mission.

Project Name: Space Rover Application	**Test Designed by:** Esm-e Moula Chowdhury Abha
Test Case ID: TR003	**Test Designed date:**18.11.23
Test Priority: High	**Test Executed by:** Esm-e Moula Chowdhury Abha
Module Name: Terrain Scanning and Way Pointing	**Test Execution date:** 18.11.23
Test Title: Waypoint Generation for Martian Exploration	

Description: Waypoint generation for Martian exploration, confirming the creation of mission-aligned waypoints based on processed terrain data.

Precondition: The rover, equipped for Mars exploration, is engaged in Mission Red Planet.Terrain data is collected successfully.

Test Steps	Test Data	Expected Result	Actual Results	Status
a. Confirm that the system generates waypoints based on the processed Martian terrain data. b. Validate that the generated waypoints align with the objectives of exploring specific Martian landmarks. c. Ensure that the number of waypoints is appropriate for the mission requirements, considering Martian topography.	● Rover engaged in Mission Red Planet. ● · Valid Martian terrain data collected	Waypoints are generated according to the objectives of exploring specific Martian landmarks	As expected,	Pass

Post Condition: The rover is ready to execute the mission with generated waypoints on the Martian surface.

Project Name: Space Rover Application	**Test Designed by:** Esm-e Moula Chowdhury Abha
Test Case ID: TR004	**Test Designed date:**18.11.23
Test Priority: High	**Test Executed by:** Esm-e Moula Chowdhury Abha
Module Name: Terrain Scanning and Way Pointing	**Test Execution date**: 18.11.23

Test Title: Path Planning for Martian Exploration

Description: Path planning for lunar exploration, verifying the efficiency of the algorithm in determining a safe route between generated waypoints.

Precondition: The rover, configured for Mars exploration, is engaged in Mission Red Planet. Terrain data is collected successfully, and waypoints are generated.

Test Steps	Test Data	Expected Results	Actual Results	Status
a. Verify the functionality of the Martian path planning algorithm. b. Confirm that the algorithm determines an efficient route between the generated waypoints on the Martian surface. c. c. Ensure that the path planning considers Martian features and potential obstacles, such as craters, for collision avoidance.	· Rover engaged in Mission Red Planet. · Valid Martian terrain data collected. · Waypoints generated for Martian exploration ·	The Martian path planning algorithm successfully determines an efficient route considering Martian features and obstacles.	As expected,	Pass

Post Condition: The rover is ready for Martian exploration with a planned route.

Project Name: Space Rover Application	Test Designed by: Esm-e Moula Chowdhury Abha
Test Case ID: TR005	Test Designed date:18.11.23
Test Priority: High	Test Executed by: Esm-e Moula Chowdhury Abha
Module Name: Terrain Scanning and Way Pointing	Test Execution date: 18.11.23

Test Title: Obstacle Avoidance in Asteroid Belt Navigation

Description: Obstacle avoidance in asteroid belt navigation, testing rover's ability to navigate around obstacles in real-time during asteroid belt traversal.

Precondition: The rover, designed for asteroid belt navigation, is engaged in Mission Asteroid. Terrain data is collected successfully, waypoints are generated, and the path planning is completed.

Test Steps	Test Data	Expected Results	Actual Results	Status
a. Simulate a scenario with potential obstacles in the asteroid belt path. b. Confirm that the obstacle avoidance algorithms successfully navigate the rover around asteroids. c. Validate that the rover adjusts its path in real-time based on the obstacle detection to avoid collisions.	· Rover engaged in Mission Asteroid. · Valid asteroid belt terrain data collected. · Waypoints generated for asteroid belt navigation. · Path planning completed for asteroid belt conditions.	Obstacle avoidance algorithms prevent collisions during rover navigation through the asteroid belt.	As expected,	Pass

Post Condition: The rover is safely navigating through the asteroid belt towards its destination.

Project Name: Space Rover Application	**Test Designed by:** Suva, Md. Wahiduzzaman
Test Case ID: THM01	**Test Designed date:** 17.11.2023
Test Priority: High	**Test Executed by:** Suva, Md. Wahiduzzaman
Module Name: Hardware Maintenance	**Test Execution date:** 17.11.2023

Test Title: Sensor Calibration and Data Measurement

Description: Verify system logs out-of-range sensor readings for analysis, ensuring the rover remains operational post-test.

Precondition: The system is currently not engaged in any type of mission.

Test Steps	Test Data	Expected Results	Actual Results	Status
1. Power on the rover. 2. Verify that dedicated sensors measure weather parameters (wind speed, atmospheric pressure, temperature, UV radiation levels).	1. Wind speed: 60 mph (beyond acceptable range). 2. Atmospheric pressure: 800 hPa (below acceptable range). 3. Temperature: -60 degrees Celsius (below acceptable range). 4. UV radiation level:15 mW/cm² (beyond acceptable range).	The system should detect out-of-range values, indicating a need for sensor calibration or maintenance.	As expected	Pass

Post Condition: Post-test, system logs out-of-range sensor readings for analysis; rover remains operational.

Project Name: Space Rover Application	**Test Designed by:** Suva, Md. Wahiduzzaman
Test Case ID: THM02	**Test Designed date:** 17.11.2023
Test Priority: High	**Test Executed by:** Suva, Md. Wahiduzzaman
Module Name: Hardware Maintenance	**Test Execution date:** 17.11.2023

Test Title: Verify System Checkup Before Mission

Description: Validate the system logs pre-mission checkup results, including communication failure, for analysis while keeping the rover operational.

Precondition: The system is currently not engaged in any type of mission.

Test Steps	Test Data	Expected Results	Actual Results	Status
1. Initiate a system checkup. 2. Verify each component (sensors, communication systems, mobility systems).	Communication system status: Simulated failure (NOT OK).	The system should report the communication system failure during the checkup.	As expected	Pass

Post Condition: Post-checkup, system logs results, including communication failure, for analysis; rover stays operational.

Project Name: Space Rover Application	**Test Designed by:** Suva, Md. Wahiduzzaman
Test Case ID: THM03	**Test Designed date:** 17.11.2023
Test Priority: High	**Test Executed by:** Suva, Md. Wahiduzzaman
Module Name: Hardware Maintenance	**Test Execution date:** 17.11.2023

Test Title: Verify Handling System Failure During Mission

Precondition: The system is engaged in a mission.

Description: Assess system logging of recovery actions post-handling a simulated communication module failure, ensuring rover state aligns with expected outcomes.

Test Steps	Test Data	Expected Results	Actual Results	Status
1. Simulate a communication module failure during the mission. 2. Attempt to reboot the system (1 for success, 0 for failure). 3. If successful, verify the rover resumes its mission. 4. If the failure persists, verify the rover goes to the base station for repair.	● Communication module failure. ● Reboot success/failure: 0 (reboot fails).	The system should report the failure, attempt to reboot, and either resume the mission or go to the base station for repair based on the outcome.	As expected	Pass

Post Condition: Post-handling failure, system logs outcome and recovery actions for analysis; rover state aligns with expected outcome.

Project Name: Space Rover Application	**Test Designed by:** Shakib Hasan
Test Case ID: MOB_01	**Test Designed date:**18.10.23
Test Priority (Low, Medium, High): High	**Test Executed by:** Shakib Hasan
Module Name: Mobility Module	**Test Execution date:** 18.10.23

Test Title: Verify Mobility of the Rover

Description: Test the mobility of the rover to ensure it can move in response to commands.

Precondition: Communication Module Initialized

Test Steps	Test Data	Expected Results	Actual Results	Status
a. Send a movement command through the interface (e.g., Move Forward).	Command: Move Forward	Rover moves forward as expected, without errors.	As expected,	Pass

b. Verify the rover's response and movement in the specified direction.				

Post Condition: Rover demonstrates successful mobility based on the provided command.

Project Name: Space Rover Application	**Test Designed by:** Shakib Hasan
Test Case ID: MOB_02	**Test Designed date:**18.10.23
Test Priority (Low, Medium, High): Medium	**Test Executed by:** Shakib Hasan
Module Name: Mobility Module	**Test Execution date:** 18.10.23

Test Title: Verify Turning Capability of the Rover

Description: Test the rover's ability to turn in different directions.

Precondition: Communication Module Initialized, Rover in a stationary state

1) **Test Steps**	**Test Data**	**Expected Results**	**Actual Results**	**Status**
A. Send a turning command through the interface (e.g., Turn Left). B. Verify the rover's response and turning in the specified direction.	Command: Turn Left	Rover turns left smoothly without errors.	As expected,	Pass

Post Condition: Rover demonstrates successful turning based on the provided command.

Project Name: Space Rover Application		Test Designed by: Shakib Hasan		
Test Case ID: MOB_03		Test Designed date:18.10.23		
Test Priority (Low, Medium, High): High		Test Executed by: Shakib Hasan		
Module Name: Mobility Module		Test Execution date: 18.10.23		
Test Title: Verify Obstacle Avoidance Capability of the Rover				
Description: Test the rover's ability to detect and avoid obstacles during movement.				
Precondition Communication Module Initialized, Rover in an open area				
2) Test Steps 1. Place an obstacle in the rover's path. 2. Send a movement command through the interface (e.g., Move Forward). 3. Verify the rover's response to detect and avoid the obstacle. The specified direction.	**Test Data** Command: Move Forward	**Expected Results** Rover detects the obstacle, stops, and changes course to avoid collision. As expected	**Actual Results** As expected,	**Status** Pass
Post Condition: Rover successfully avoids obstacles during movement.				

Work Breakdown Structure
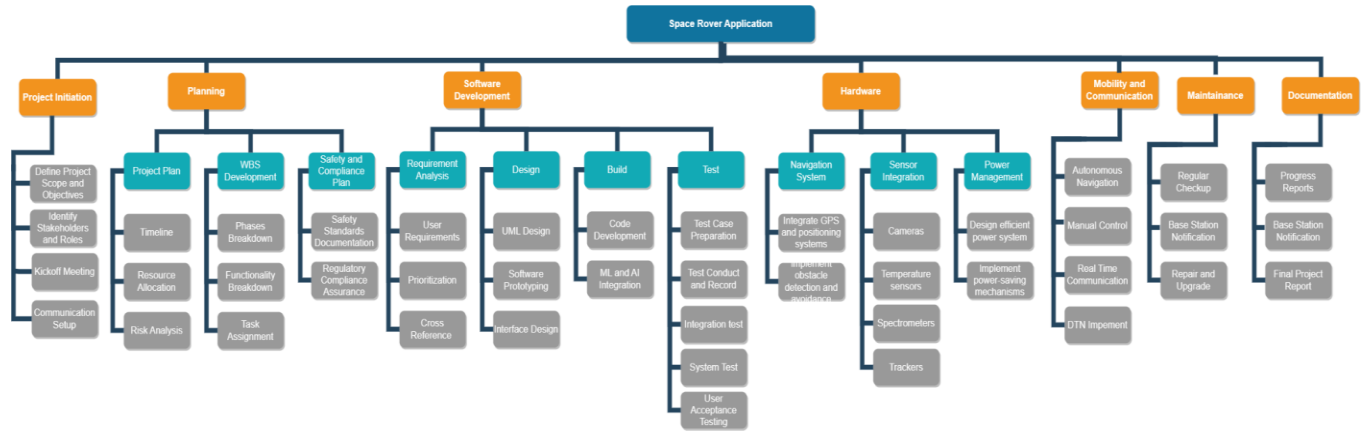
Project Estimation (COMOMO)

For our embedded system project, let's take the number of lines of code to be,

SLOC = 7000,

Effort Factor Coefficient = 3.6,

P = 1.2 and

T = 0.32.

We know,

$$\text{Effort (PM)} = \text{Coefficient}_{(\text{Effort Factor})} \times \left(\frac{SLOC}{1000}\right)^P$$

$$\text{Effort (PM)} = 3.6 \times \left(\frac{7000}{1000}\right)^{1.2}$$

$$\text{Effort (PM)} = 37.2$$

$$\text{Development Time (DM)} = 2.50 \times (PM)^T$$

$$= 2.50 \times (26.52)^{0.32}$$

$$= 7.95 \; Months$$

$$Required \; Number \; of \; People \; (ST) = \frac{PM}{DM} = \frac{37.2}{7.95} = 4.72 \sim 5$$

Timeline Char

[Live link of the following table](#)

	TIME CHART	SEPTEMBER	OCT	NOV	DEC	JANUARY	FEBRUARY	MARCH	APRIL
1.1	Defining project goals and objectives								
	Identifying stakeholders and establishing communication channels								
	Developing a project chart								
	Assembling project team and allocate responsibilities								
1.2	Gathering and documenting functional requirements								
	Defining non-functional requirements (performance, security, etc.)								
	Prioritizing and validating requirements with stakeholders								
	Creating a requirements traceability matrix								
1.3	Creating high-level system architecture								
	Designing data models and database schema								
	Defining user interfaces and user experience (UI/UX)								
	Breaking down the system into modular components								
1.4	Setting up development environment and tools								
	Writting code for individual modules								
	Implementing unit tests and conduct code reviews								
	Integrating modules and perform system testing								
1.5	Conducting functional testing to ensure requirements are met								
	Performing performance and stress testing								
	Identifying and fixing bugs and issues								
	Conducting system testing								
1.6	Preparing for deployment								
	Creating deployment documentation								
	Deploying the software to the rover's system								
	Monitoring for any deployment issues and addressing them								
1.7	Creating user manuals and documentation								
	Developing system documentation for maintenance								
	Documenting any configurations or settings								
	Ensuring all documentation is accessible and organized								
1.8	Evaluating project against								
	Conducting a review with stakeholders								
	Identifying areas for improvement								
	Capturing feedback								
1.9	Completing any remaining administrative tasks								
	Closing out the project in terms of budget, resources, and documentation								
	Archiving project files and making final backups								

Documentation		Entire Team
Project Review		Project Manager
Project Completion		Project Manager

EVA

EVA

Task	Planned Effort	Actual Effort
1	13.0	12.5
2	16.5	17.0
3	10.5	9.5
4	18.0	17.5
5	9.0	9.5
6	12.5	13.0
7	6.5	6.0
8	11.0	11.0
9	15.5	16.5
10	8.0	10.0
11	14.0	-
12	5.5	-

BCWP=120.5 (for Planned Effort tasks 1–10); BCWS=140.0 (for Planned Effort tasks 1–12); ACWP=122.5 (for Actual Effort tasks 1–10)

- BAC = 243.00
- SPI = BCWP/ BCWS =120.5/ 140.0 =0.8607
- SV = BCWP - BCWS =120.5 - 140.0 =-19.5 person-day
- CPI = BCWP/ ACWP = 0.984
- CV = BCWP – ACWP = -2 person-day
- % schedule for completion = BCWS/ BAC = 140.0/ 243.00 = 57.6%
- % complete = BCWP/ BAC = 120.5/ 243.00 = 49.6%

Given Total Task = 36; Effort Estimated=243 Person Day

Risk Table

Risk	Category	Probability	Impact
Size estimate may be significantly low	PS	60%	2
Larger number of users than planned	PS	30%	3
Less reuse than planned	PS	70%	2
End users resist system	BU	40%	3
Delivery deadline will be tightened	BU	50%	2
Funding will be lost	CU	40%	1
Customer will change requirements	PS	80%	2
Technology will not meet expectations	TE	30%	1
Lack of training on tools	DE	80%	3
Staff inexperienced	ST	30%	2
Staff turnover will	ST	60%	2
Team members do not work well	ST	20%	4
Unfamiliar areas of the product take more time than expected	DE	20%	2
Components developed separately cannot be integrated easily,requires redesigning	DE	25%	3
Key personnel are available only part time	ST	20%	4
Security vulnerabilities in the technology stack	TE	20%	4
Changes in market trends impacting product relevance	BU	50%	3
Inadequate backup and recovery procedures	TE	25%	3
Insufficient unit testing, leading to undiscovered bugs	DE	30%	3
Inconsistent or outdated development environments	DE	20%	3