## 

## 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** | |
| --- | --- | --- | --- | --- |
| SN | Student Name | Student ID | Contribution (CO1+CO2) | IndividualMarks |
| 1 | Hasan, Shakib | 22-46786-1 |  |  |
| 2 | Abha, Esm-e Moula Chowdhury | 22-47166-1 |  |  |
| 3 | Akash, Susham Moula Choudhury | 22-47168-1 |  |  |
| 4 | Suva, Md. Wahiduzzaman | 22-47171-1 |  |  |
| 5 | Sajin, Md. Imtiaj Alam | 22-47185-1 |  |  |

**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.

**Software Requirement**

1. Hardware Maintenance
   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. Prior to any mission, system check up should be performed
   3. If system failure is detected, report to base and reboot if possible
   4. If the system does not work after system reboot, go to the base station for repair.
   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

1. Mission Status
   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. 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.
   3. The user can check progress of the mission assigned to the rover, and estimated time for mission completion
   4. Mobility and visibility of the vehicle can be predicted to ensure maximum efficiency of the mission
   5. Weather, route and atmospheric status can be checked to ensure minimization of vehicle damage during any mission

**Priority Level:** High

**Precondition:** T

he 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

### **Mobility**

* 1. In automatic mode, the rover will autonomously navigate diverse terrains using all-terrain drive mechanisms and obstacle avoidance algorithms.
  2. In manual mode, operators can control the rover's movements, including speed, direction, and stopping.
  3. In order to manually control the rover the operator has to communicate with the rover.
  4. In manual mode, Rover will provide all the data including live camera feeds, telemetry data, and status updates, to ensure precise control.
  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

### **Terrain Scanning and Way Pointing**

* 1. The first requirement involves the integration of multiple onboard sensors, including cameras, radar etc to gather terrain data.
  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.
  3. Using the terrain data, the system will generate waypoints. These waypoints will be determined based on the mission objectives.
  4. A path planning algorithm will be used to determine an efficient route between the waypoints.
  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

### **Weather Data Collection**

* 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. To ensure timely access to weather data, algorithms will be implemented to facilitate real-time data acquisition from the integrated sensors.
  3. A data logging mechanism will be developed for recording the collected weather data, complete with timestamps, for reference and analysis.
  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

1. Soil Sample Collection
   1. Find a suitable location to collect samples.
   2. After scanning the surface and decide whether sample collection is possible, if not relocate to new location
   3. Drill to a specific depth and collect the sample.
   4. After collecting the sample, store them in containers for further analysis.
   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

1. **Communication**
   1. System will initialize the communication module when it’s not in idle state or when it goes for a new mission.
   2. Then this communication module uses a data streaming mechanism to establish real-time communication between the rover and Base station.
   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.
   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.
   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

**UML Diagram**

1. Hardware Maintenance

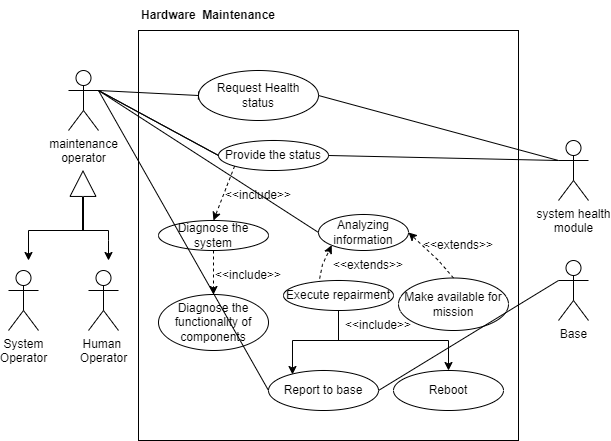


Figure 1.1: Use case diagram hardware maintenance

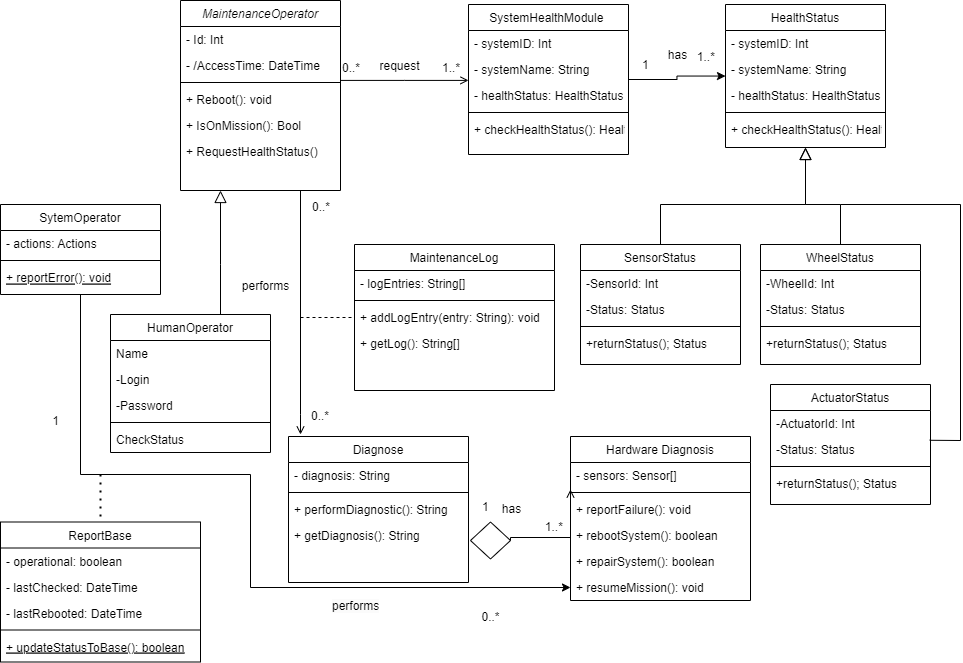
**

Figure 1.2: Use case diagram hardware maintenance

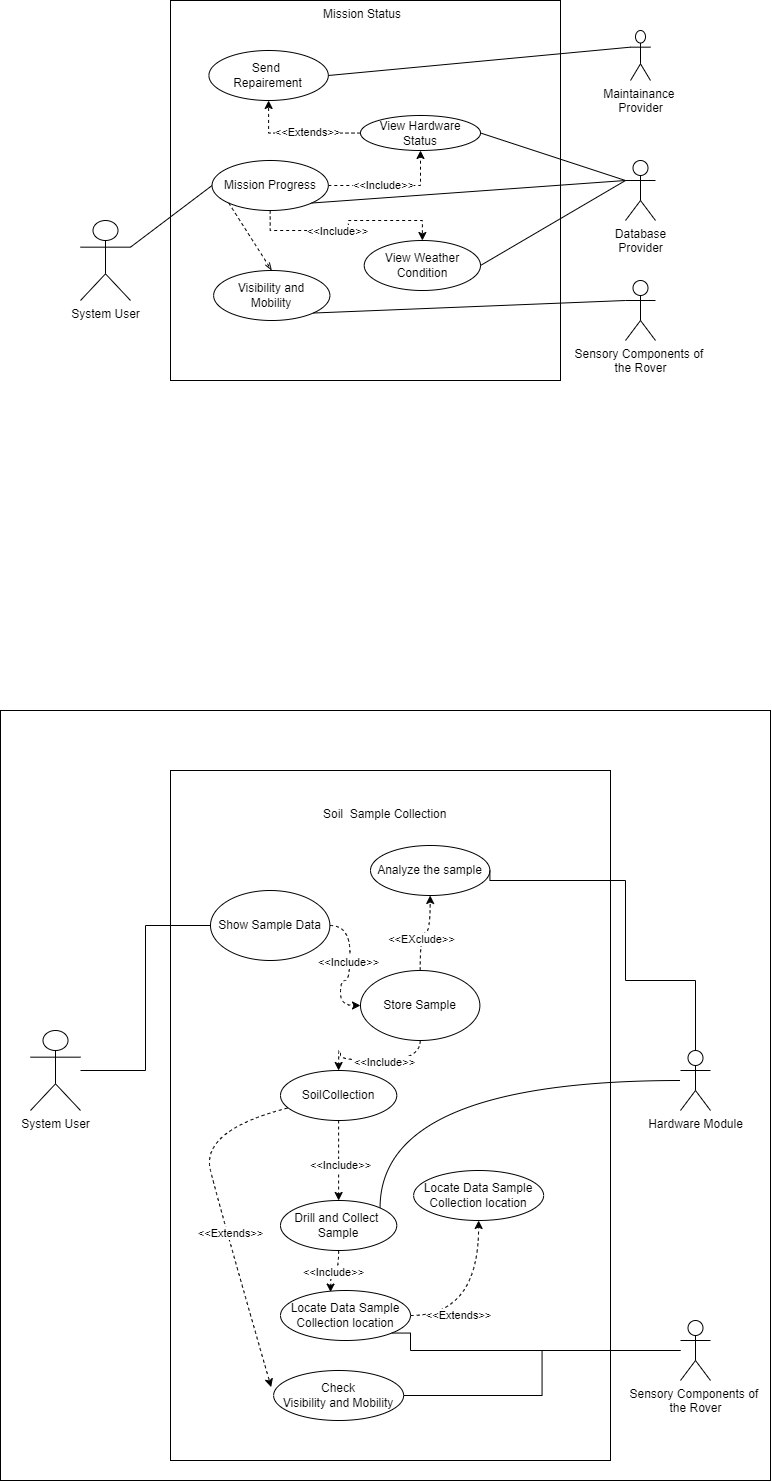
1. Mission Status

Figure 2.1: Use case diagram hardware maintenance

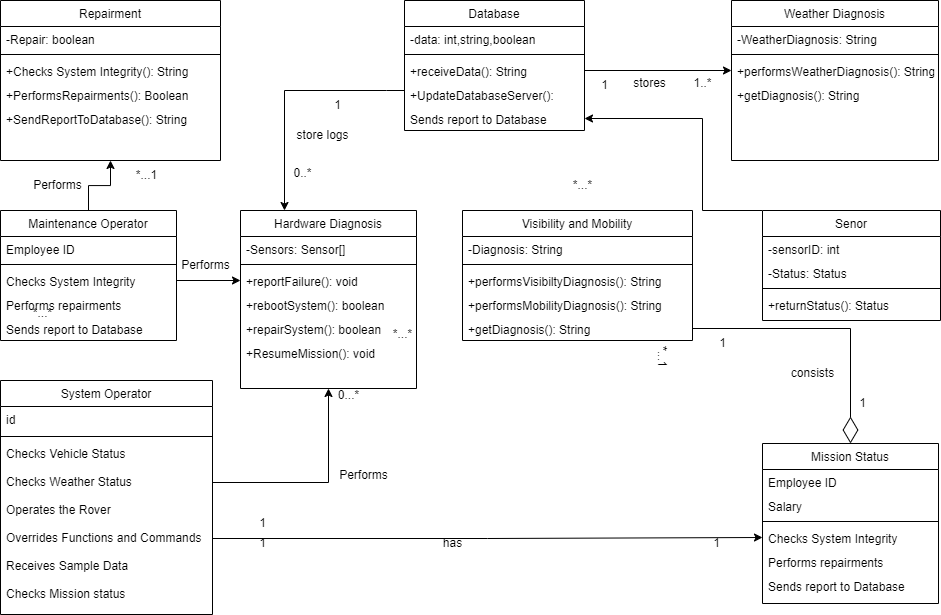
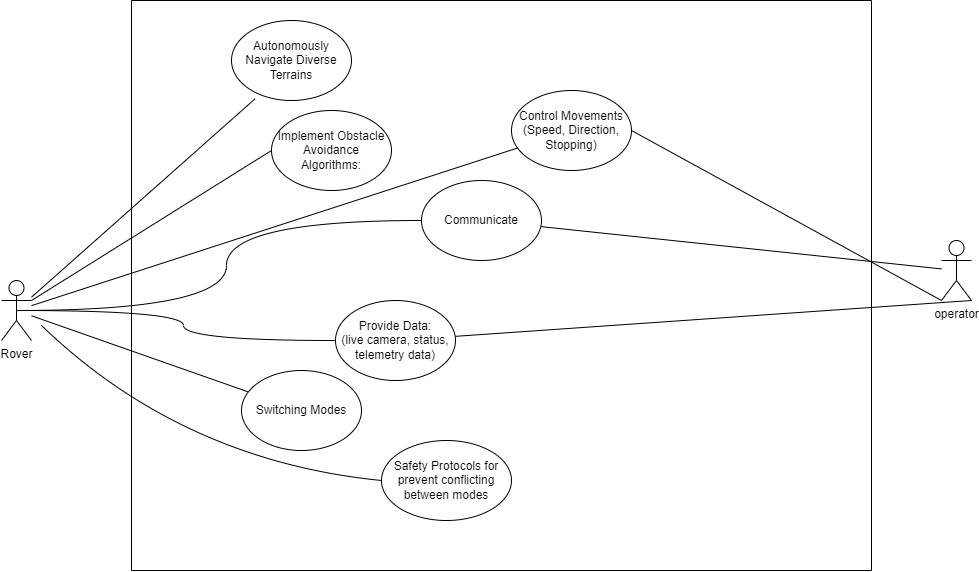


Figure 2.2: Use case diagram hardware maintenance

### **Mobility**

Figure 3.1: Use case diagram of Mobility

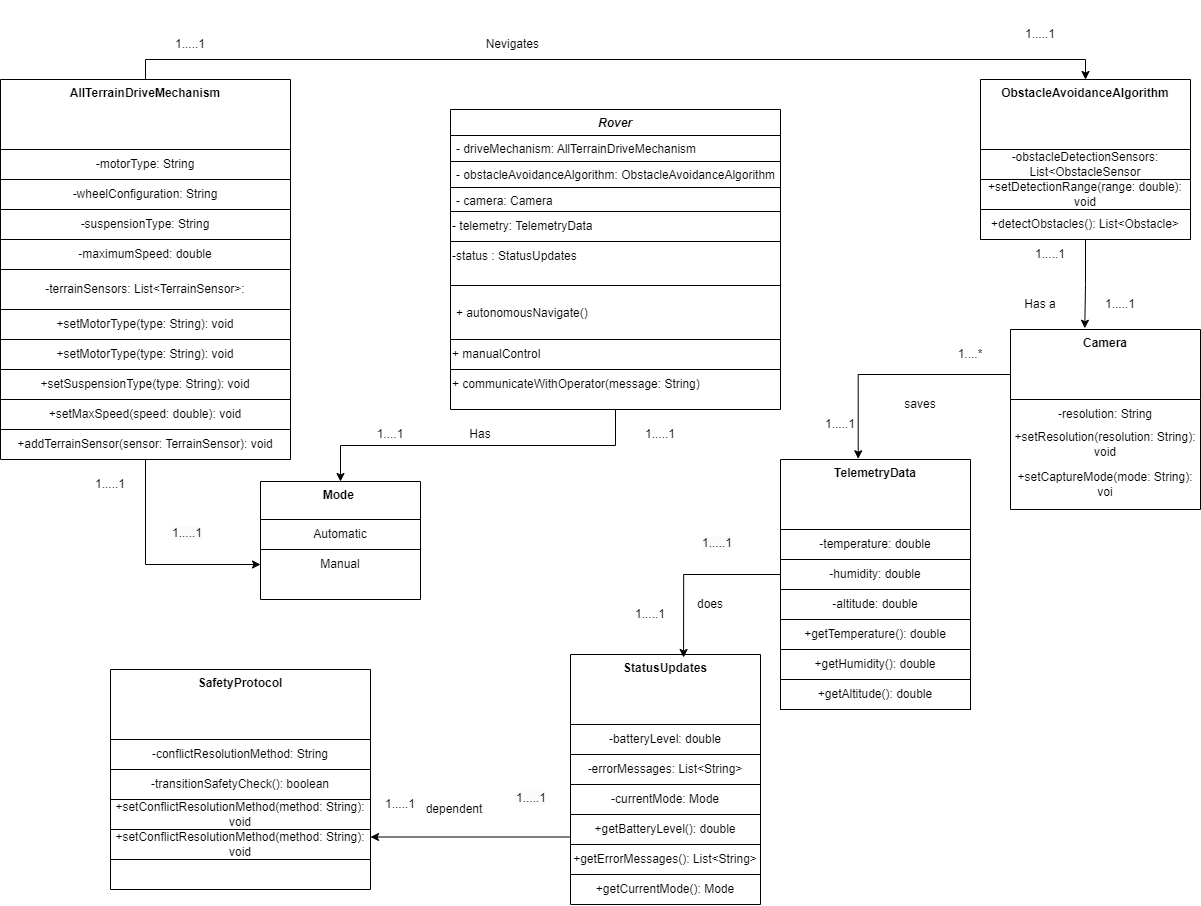
**

Figure 3.2: Class diagram of Mobility

### **Terrain Scanning and Way Pointing**

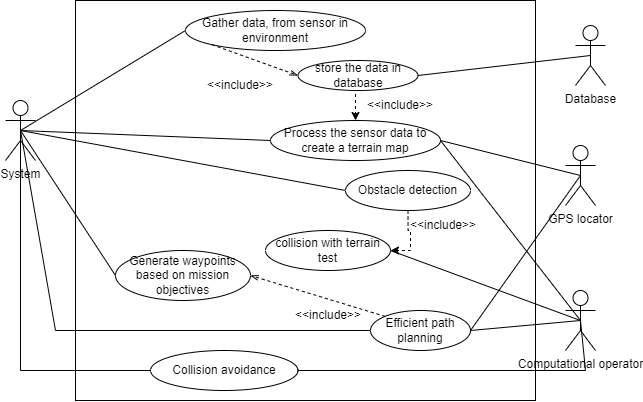


Figure 4.1: Use case diagram of Terrain scanning and way pointing.

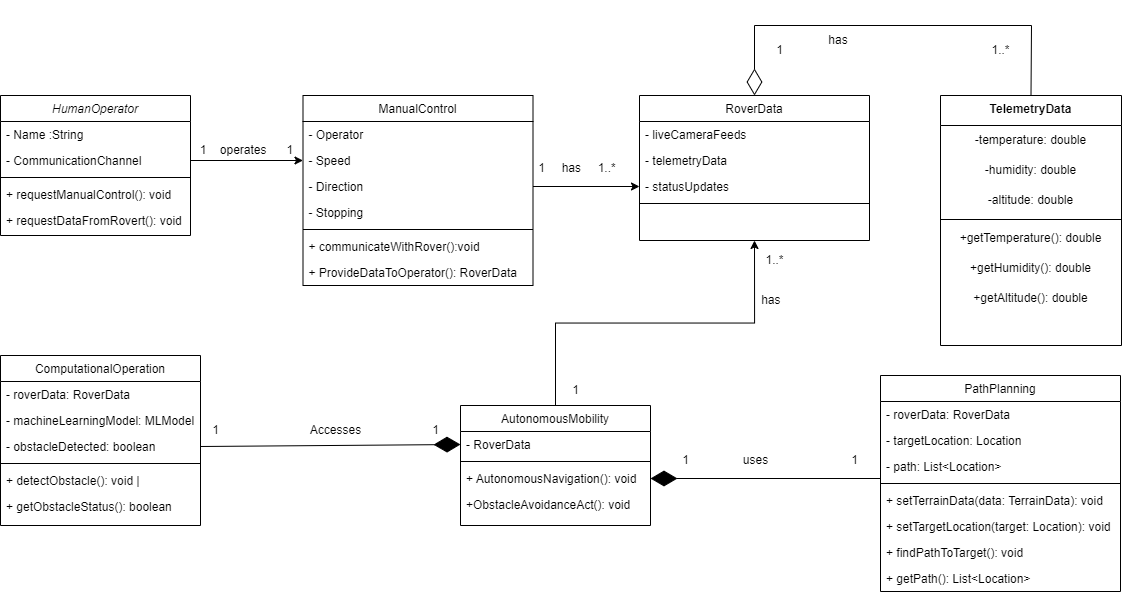


Figure 4.2: Class diagram of Terrain scanning and way pointing.

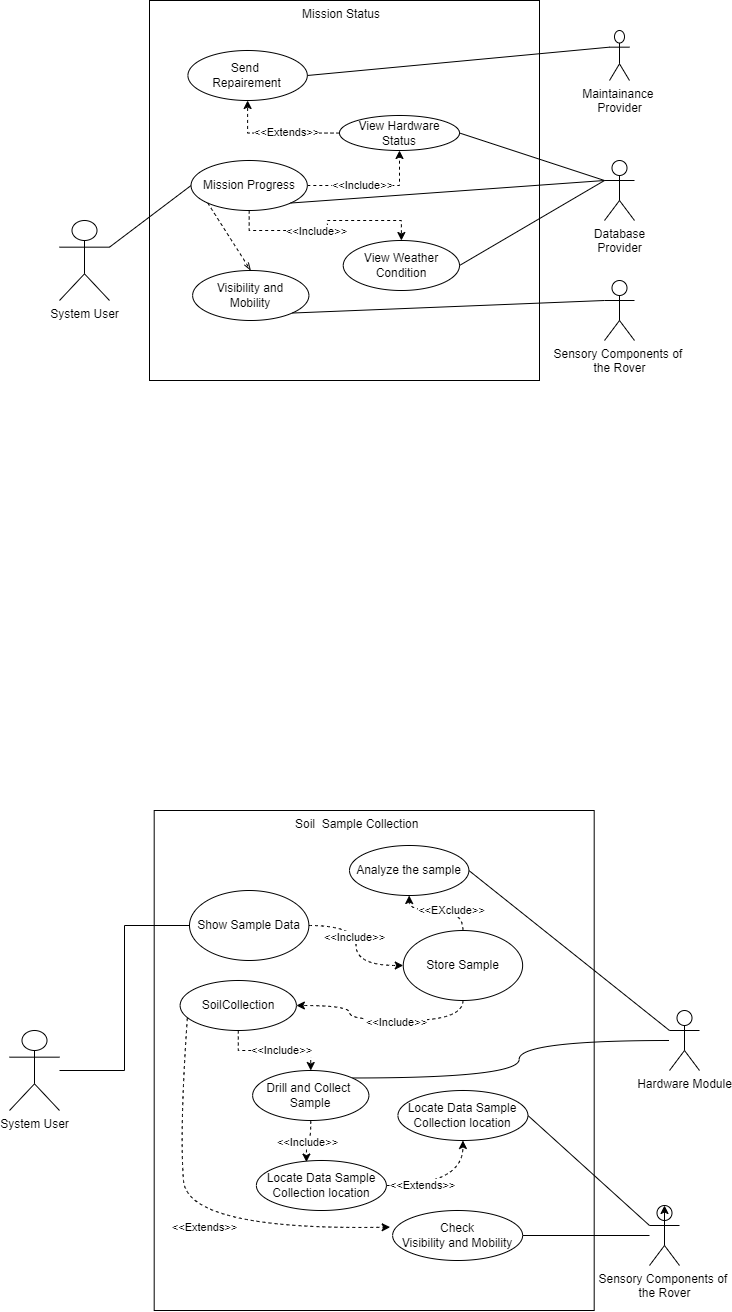
1. Soil Sample Collection

Figure 5.1: Use case diagram of Soil Sample Collection.

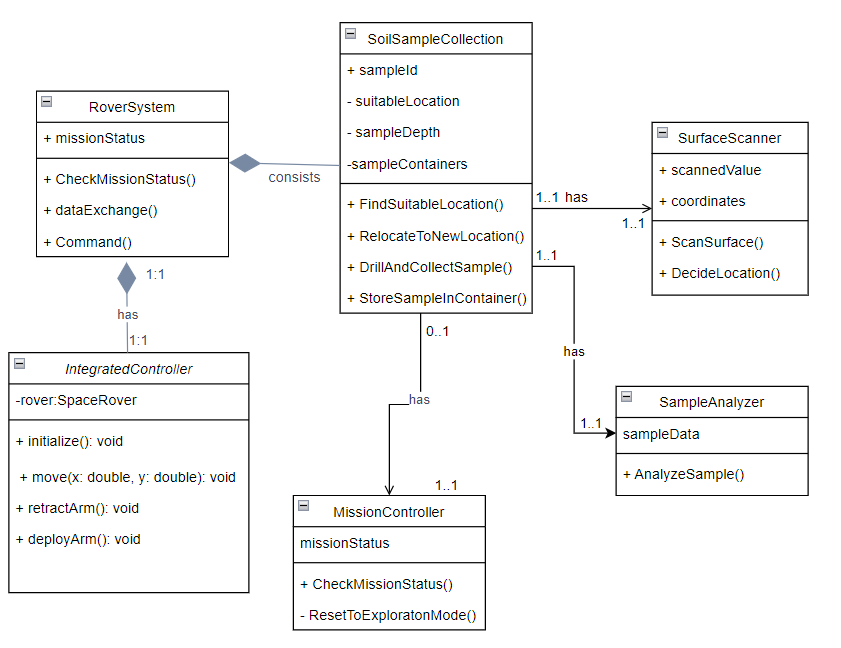


Figure 5.2: Class diagram of Soil Sample Collection.

1. **Communication**

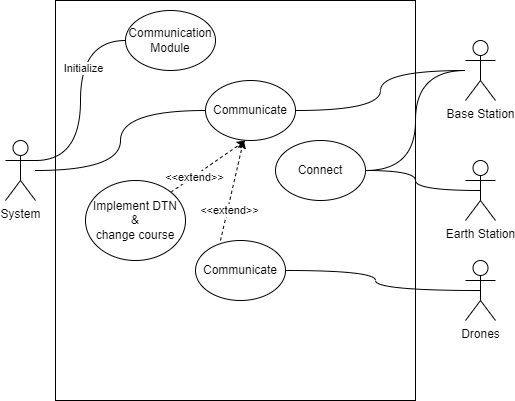


Figure 6.1: Use case diagram of Communication

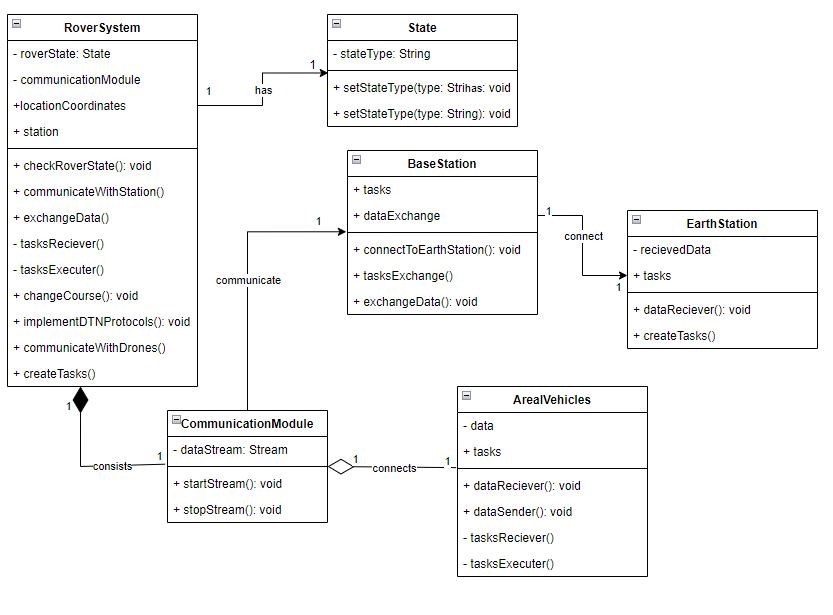
**

Figure 6.2: Class diagram of Communication

**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.

**Sequence Diagram:**

| **Name** | **Task** |
| --- | --- |
| **jim** |  |
|  |
| **abha** |  |
|  |
| **akash** |  |
|  |
| **shuvo** |  |
|  |
| **sajin** |  |
|  |