

System Requirements

Group #12 – Botanica

SproutBot

Arun Mistry

Mina Demian

Nicholas Levantis

Usman Minhas

Table 1: Revision History

Date	Developer(s)	Change
November 10, 2023	Arun Mistry, Mina Demian, Nicholas Levantis, & Usman Minhas	Initial System Requirements
March 25, 2024	Arun Mistry, Nicholas Levantis	Added Feedback Integration table. Added comment mentioning Hazard ID link to Safety Requirements.
March 26, 2024	Nicholas Levantis	Added monitored and controlled variables to corresponding requirements.
March 27, 2024	Arun Mistry	Fixed Figure 1

Table of Contents

1. Project Purpose and Overview	5
2. System Overview	5
2.1. Project Scope	5
2.2. Project Constraints.....	5
2.3. System Boundary Diagram.....	6
2.4. System Context Diagram.....	7
2.5. Behaviour Overview.....	8
2.6. Functional Decomposition	9
3. Variables and Constants.....	10
3.1. Monitored Variables	10
3.2. Controlled Variables.....	10
3.3. System Constants.....	11
4. Functional Requirements.....	12
4.1. Robot Requirements	12
4.1.1. Physical Movement of the Robot (RM).....	12
4.1.2. Navigation of the Robot (RN).....	12
4.2. Plant Requirements	13
4.2.1. Plant Maintenance Monitoring (PM).....	13
4.2.2. Plant Maintenance Delivery (PD).....	13
4.3. Environment Requirements	14
4.4. Human Requirements.....	14
5. Safety Requirements.....	14
5.1. Safety Requirements for Robot (SRR)	14
5.2. Safety Requirements for Plant (SRP).....	15
5.3. Safety Requirements for Environment (SRE)	15
5.4. Safety Requirements for Human (SRH).....	15
6. Non-Functional Requirements.....	16
7. Operational Scenarios.....	16
7.1. Normal Operation	16
7.2. Undesired Operations & Handling.....	17
7.2.1. No path is available.....	17
7.2.2. Insufficient water available.....	17
7.2.3. Foliage in the way	17

7.2.4. System tips over.....	17
7.2.5. Loss of communication	17
8. Requirement Modifications	17
8.1. Requirements Likely to Change	17
8.2. Requirements not likely to change	18
9. References	18
Feedback Integration	18

List of Figures

Figure 1: System Boundary Diagram.....	6
Figure 2: System Context Diagram.....	7
Figure 3: Functional Decomposition Diagram	9

List of Tables

Table 1: Notations.....	8
Table 2: Monitored Variables.....	10
Table 3: Controlled Variables	10
Table 4: System Constants	11

1. Project Purpose and Overview

SproutBot is a plant watering robot that aims to provide a reliable and efficient solution for ensuring the health and well-being of indoor greenery, especially when plant owners are away from home. Its primary objective is to water plants around the house as and when necessary. SproutBot will be considered a success if it is able to move, navigate towards a plant's location, and deliver water to it, repeating this process for other plants as required, and finishing its trip by returning to its base location.

SproutBot is divided into 2 main subsystems, with its movement and navigation system as one, and the watering mechanism as another. These systems will be developed separately and combined in the end. Further details are given under [System Overview](#).

2. System Overview

2.1. Project Scope

The scope of this project is to have the robot navigate toward plants with minor adaptability (such as stopping & going around obstacles) to various plants around the same height and deliver a specific quantity of water on a schedule basis. As the potential of this project is massive, we plan to limit the scope of the project to features we believe we can implement successfully.

The scope of this project does not include:

- Plant pots of varying heights. The range of plant pot's height will be within [POT HEIGHT RANGE](#).
- Advanced terrain navigation. The robot will not be required to climb any obstacles to navigate towards a plant. Basic movement and obstacle detection, with rudimentary obstacle avoidance will be within scope, and expanded upon as necessary.
- Elevated plants. The robot will only be required to tend to plants located on the floor of the home.
- Plants on different floors of a home. The robot will not be required to travel up and down stairs or ramps.
- Outdoor operation. This robot will be for indoor use only and will not be built resistant to different weather elements such as strong winds and rain or snow.

2.2. Project Constraints

There are 2 main constraints to keep track of when working on this project.

1. Budget. There is a maximum budget of \$750 for the Bill of Materials (BOM) that should not be exceeded.
2. Time. Revision 0 is due by January 26, 2024, which must capture the major components of the project. Revision 1 is due by March 27, 2024, which must build upon Revision 0. The project's ultimate completion date is to be the date of the Final Presentation, March 27, 2024.

2.3. System Boundary Diagram

The system boundary diagram gives a visual representation of how the system interacts with the environment. It shows the monitored variables coming into the system on the left, and the controlled variables coming out on the right. The relationship between the monitored and controlled variables that occurs within the system will be described further in this document.

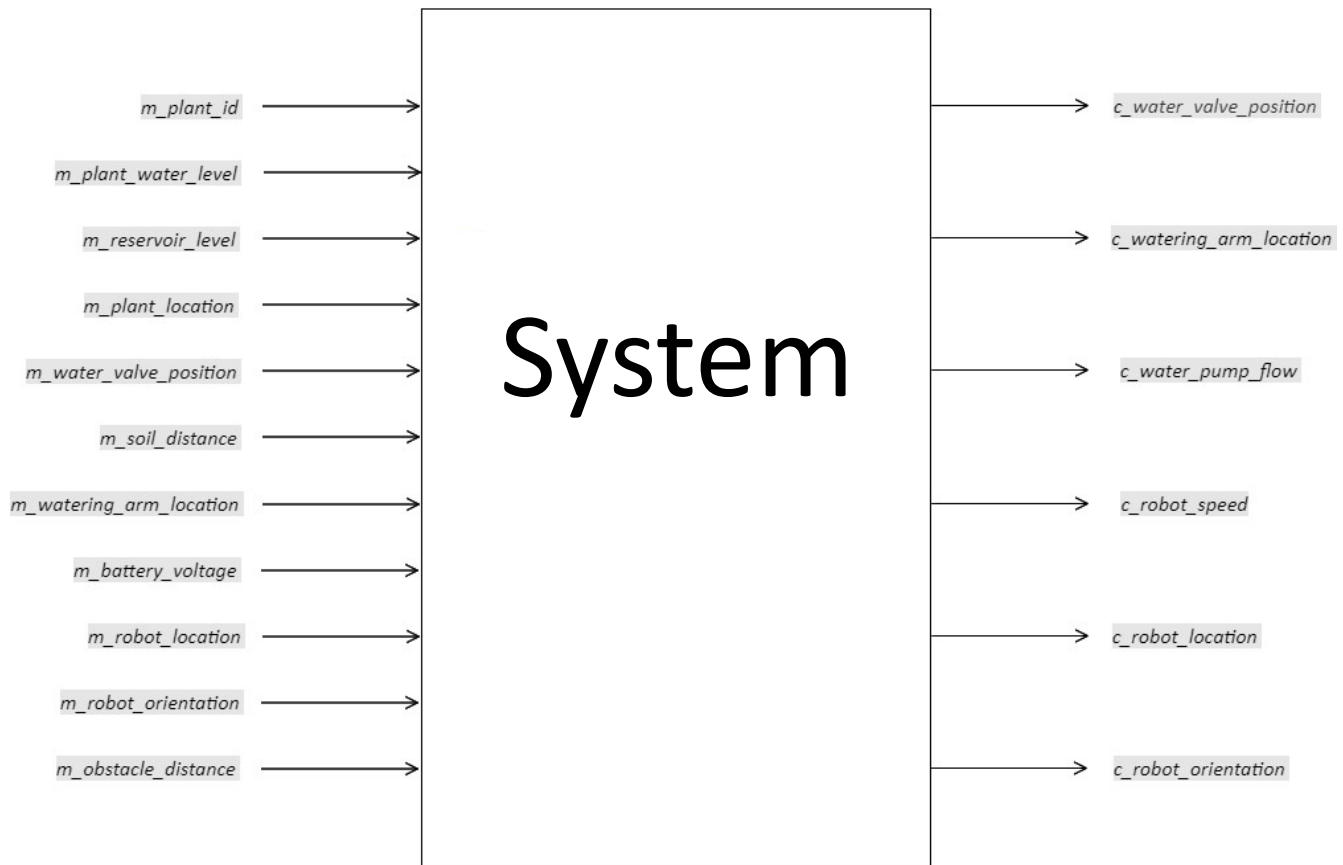


Figure 1: System Boundary Diagram

2.4. System Context Diagram

The system component diagram gives a detailed representation of the components within the system and how they interact with each other and the environment. The dotted red line shows the system boundary.

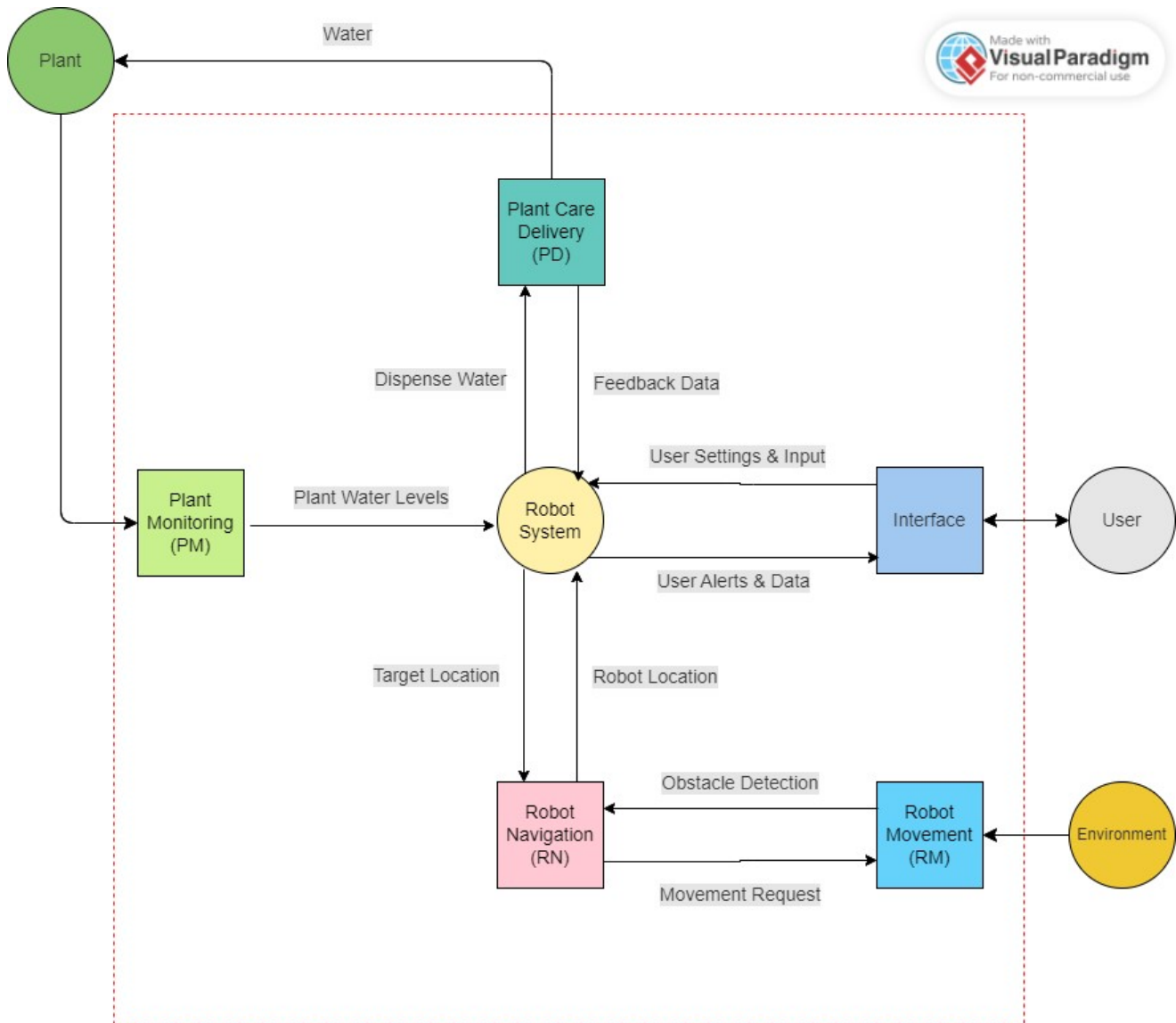


Figure 2: System Context Diagram

2.5. Behaviour Overview

Notation	Type	Description
m_	Prefix	Denotes monitored variables. See 3.1. Monitored Variables
c_	Prefix	Denotes controlled variables. See 3.2. Controlled Variables
NAME	Capitalization	Variable names in full uppercase refer to constants. See 3.3. System Constants
RM	Acronym	Refers to requirements related to 4.1.1. Physical Movement of the Robot (RM)
RN	Acronym	Refers to requirements related to 4.1.2. Navigation of the Robot (RN)
PM	Acronym	Refers to requirements related to 4.2.1. Plant Maintenance Monitoring (PM)
PD	Acronym	Refers to requirements related to 4.2.2. Plant Maintenance Delivery (PD)
E	Acronym	Refers to requirements related to 4.3. Environment
H	Acronym	Refers to requirements related to 4.4. Human
SR	Acronym	Refers to Safety Requirements. See Safety Requirements
SRR	Acronym	Refers to Safety Requirements for 5.1. Safety Requirements for Robot (SRR)
SRP	Acronym	Refers to Safety Requirements for 5.2. Safety Requirements for Plant (SRP)
SRE	Acronym	Refers to Safety Requirements for 5.3. Safety Requirements for Environment (SRE)
SRH	Acronym	Refers to Safety Requirements for 5.4. Safety Requirements for Human (SRH)
NF	Acronym	Refers to Non-Functional Requirements

Table 1: Notations

2.6. Functional Decomposition

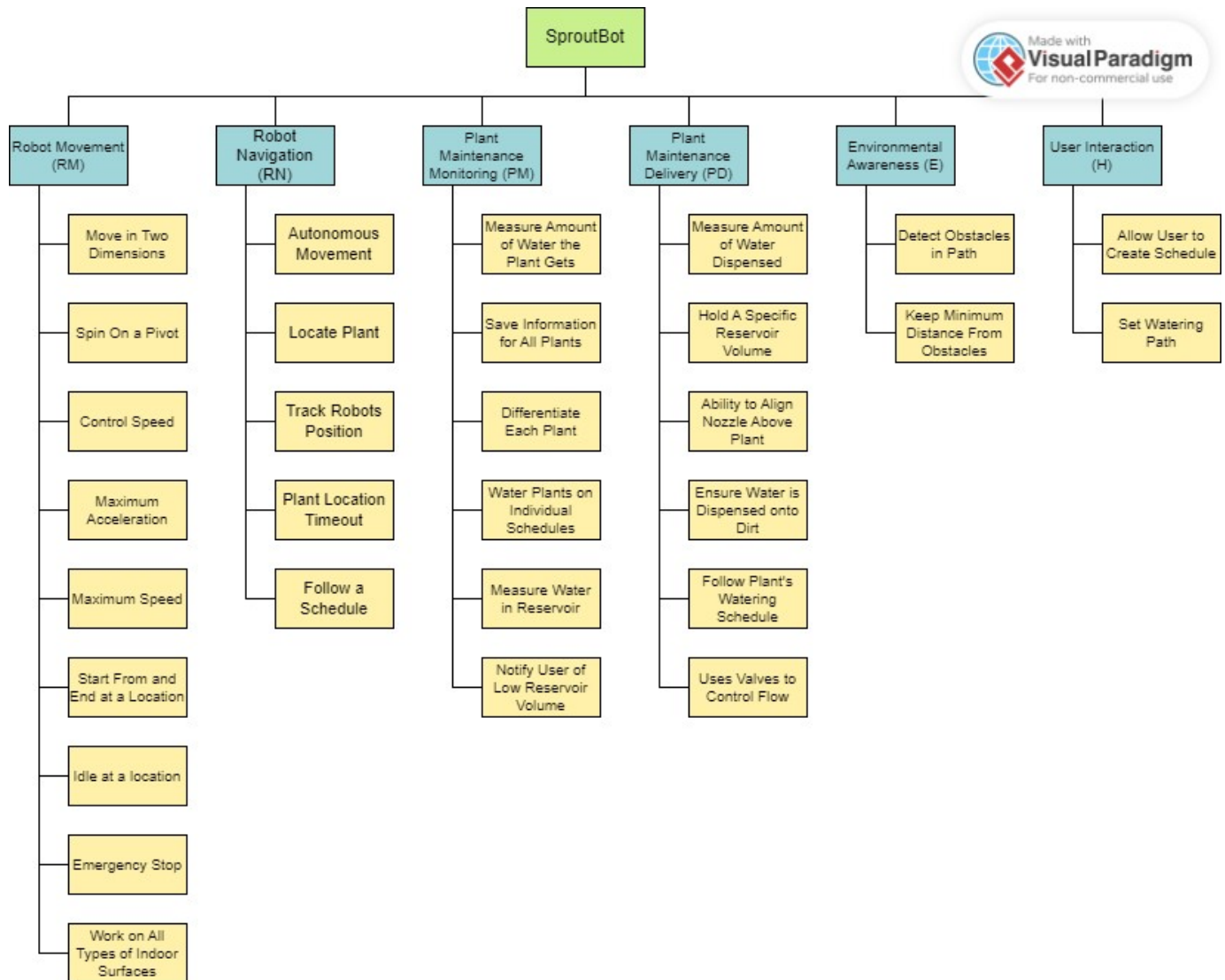


Figure 3: Functional Decomposition Diagram

3. Variables and Constants

3.1. Monitored Variables

Name	Unit	Description
<i>m_plant_id</i>	-	The plant's identification details in order to control how much water is supplied at specific times.
<i>m_plant_water_level</i>	mL	Measures the moisture level of the plant. Each plant will have its own moisture sensor and value.
<i>m_reservoir_level</i>	mL	Measures the volume of water in the robots on board reservoir.
<i>m_plant_location</i>	Vector in m <x, y, z>	Determines the location of the plant relative to the robot.
<i>m_ir_distance</i>	m	Distance between the IR Emitter at the destination and IR Receivers on the robot
<i>m_water_valve_position</i>	-	Tracks whether the valve for watering mechanism is open or closed.
<i>m_soil_distance</i>	m	Determines if the nozzle of the water mechanism is above soil or not.
<i>m_watering_arm_location</i>	Vector in m <x, y, z>	Location of end effector of robot watering arm, relative to the robot.
<i>m_robot_location</i>	Vector in m <x, y, z>	Location of the robot relative to its environment, such as its initial starting position and the plant.
<i>m_robot_orientation</i>	Vector in degrees < α , β , θ >	The orientation of the robot relative to its starting position.
<i>m_obstacle_distance</i>	m	Location of an obstacle relative to the robot.
<i>m_color_sensor_output</i>	Vector in RGB	The RGB output of the color sensor

Table 2: Monitored Variables

3.2. Controlled Variables

Name	Unit	Description
<i>c_water_valve_position</i>	-	Tracks whether the valve for watering mechanism is open or closed.
<i>c_watering_arm_location</i>	Vector in m <x, y, z>	Location of end effector of robot watering arm, relative to the robot.
<i>c_water_pump_flow</i>	mL/s	Speed of the water being pumped.
<i>c_robot_speed</i>	m/s	Overall speed of the robot when it is moving.
<i>c_robot_location</i>	Vector in m <x, y, z>	Location of the robot relative to its environment, such as its initial starting position and the plant.
<i>c_robot_orientation</i>	Vector in degrees < α , β , θ >	The orientation of the robot relative to its starting position.
<i>c_plant_water_needed</i>	Bool	Tracks whether a specific plant needs to be watered or not.

Table 3: Controlled Variables

3.3. System Constants

Name	Unit	Description
<i>NOMINAL_TEMPERATURE</i>	°C	The expected room temperature
<i>NOMINAL_PRESSURE</i>	Atm	The expected air pressure in the room
<i>BATTERY_VOLTAGE</i>	V	The maximum voltage available to supply robot components
<i>MOTOR_TORQUE</i>	Nm	The torque of all motors used for movement will be constant.
<i>POT_HEIGHT_RANGE</i>	m	The range of heights that is allowable for a pot.
<i>RESERVOIR_CAPACITY</i>	L	The maximum water the on board reservoir can hold.
<i>RESERVOIR_CAPACITY_LOW</i>	L	The threshold for when the water in the reservoir is considered low, for the user to be notified.
<i>MIN_DISTANCE</i>	m	The minimum distance that the robot must leave from all environmental obstacles.
<i>MAX_FORCE</i>	N	The maximum external force that the robot must receive before stopping operations.
<i>MAX_SPEED</i>	m/s	The maximum speed the robot should be allowed to move at.
<i>MAX_ACCELERATION</i>	m/s ²	The maximum acceleration that the robot should be allowed to move at.
<i>BASE_LOCATION</i>	Vector in metre <x, y, z>	The position where the robot stays when it is not required to water plants.
<i>MAX_ATTEMPT_TIME</i>	s	The maximum time the robot can attempt to reach a plant before it must stop.
<i>MAX_RESPONSE_TIME</i>	s	The maximum duration allowable for the user and system's interaction, in either direction.
<i>MAX_RADIUS</i>	m	The maximum radius within which the robot can detect and identify plants to water.
<i>RESERVOIR_CAPACITY_LOW</i>	%	The percentage water reservoir level that the user will be notified at, once it is reached.
<i>BATTERY_LEVEL_LOW</i>	%	The percentage battery level that the user will be notified at, once it is reached.

Table 4: System Constants

4. Functional Requirements

The functional requirements for this project will be divided into 4 different components as identified above. Each component may include subsections that the parent component will include. These are explained below.

1. Robot

- Physical Movement of the Robot (**RM**)
 - How the robot will physically travel between arbitrary points
- Navigation of the Robot (**RN**)
 - How the robot's software will help navigate between different known points

2. Plant

- Plant Maintenance Monitoring (**PM**)
 - How different plants will be identified and monitored for different scenarios
- Plant Maintenance Delivery (**PD**)
 - How water will be delivered to the plant

3. Environment (E)

4. Human (H)

4.1. Robot Requirements

4.1.1. Physical Movement of the Robot (RM)

- RM1. The robot must be able to move along the floor in two dimensions.
Rationale: Since the system is only worried about plants on the floor at similar heights, the system only needs to be able to move along two dimensions.
- RM2. The robot must be able to control its speed, [c_robot_speed](#).
- RM3. The robot must have a maximum acceleration of [MAX_ACCELERATION](#).
- RM4. The robot must not move at a speed greater than [MAX_SPEED](#).
Rationale: The system must move in a controlled way and at an acceptable speed and acceleration, so it does not produce any sudden movements and cause any harm to anything around it.
- RM5. The robot must be able to start from and end at a fixed location, [BASE_LOCATION](#).
- RM6. The robot must be able to stay idle at a fixed location, [BASE_LOCATION](#).
Rationale: The system must start and return to the [BASE_LOCATION](#), as that would be the location away from obstacles that the system would wait at until the next watering cycle.
- RM7. The robot must be able to move on all types of indoor flooring surfaces.
Rationale: The system must be able to move on all types of indoor surfaces that are available in rooms, so the system is able to move on surfaces such as hardwood, ceramic, carpets, etc. and does not slip or gets stuck.

4.1.2. Navigation of the Robot (RN)

- RN1. The robot must be able to autonomously travel between different locations, using [m_robot_location](#) to control [c_robot_location](#).

Rationale: The system must be able to move around an indoor environment and go from plant to plant in an autonomous fashion with limited user interaction, as the user might want to water their plants while they are travelling or not at home for a certain period.

- RN2. The robot must be able to locate the position of a plant (*m plant location*) within a distance of *MAX RADIUS*.

Rationale: The system must be able to locate its own position and the position of the plants within a distance of *MAX RADIUS*, so that it is able to navigate to the plants and know when it is within the radius of the plant.

- RN3. The robot must be able to keep track of its own position, *m robot location*.

- RN4. The robot must notify the user if a plant is not reached within a duration of *MAXIMUM ATTEMPT TIME*.

Rationale: The system must notify the user if it is not able to reach a plant within a duration of *MAXIMUM ATTEMPT TIME*, so that the user can manually navigate the system to the plant.

- RN5. The robot must follow a schedule to start navigation towards plants.

Rationale: The system to be able to follow a schedule to navigate to all my plants in the specified times, for it to water them at the appropriate times.

4.2. Plant Requirements

4.2.1. Plant Maintenance Monitoring (PM)

- PM1. The system must recognize how much water is being received by the plant (*m plant water level*).

Rationale: The system must ensure that the plant is receiving correct amount of water and is not underwatered or overwatered.

- PM2. The system must be able to save the watering information pertaining to each plant using *m plant id*.

Rationale: The system must know the information and watering schedules pertaining to each plant and it follows that, so the plants are watered at the appropriate times and frequencies.

- PM3. The system must be able to differentiate plants using *m plant id*.

Rationale: The system must be able to differentiate each plant, to determine the plant's set schedule and know when to water each plant and how much water to provide.

4.2.2. Plant Maintenance Delivery (PD)

- PD1. The system must be able to identify how much water is coming out of the pump (*c water pump flow*).

Rationale: The system must be able to identify how much water is being supplied by the system to the plant, to ensure that the system is supplying the correct amount of water.

- PD2. The system must hold water up to a maximum capacity of *RESERVOIR CAPACITY*.

Rationale: The system must be able to hold water up to a maximum capacity of *RESERVOIR CAPACITY*, so that it can water all plants for the whole duration of the user being away.

- PD3. The system must be able to recognize *m watering arm location* and align its water delivery mechanism (*c watering arm location*) to the plant's soil.

- PD4. The system must directly target the soil when watering.

Rationale: The system must be able to align itself with the plants while avoiding any foliage and directly aiming the watering system to the soil, so that the system can properly water the different plant pots' soils without dropping water outside of the plant pots.

- PD5. The system must follow the plant's watering schedule.

Rationale: The system must recognize the schedule set for each plant and follow it, so it is able to provide each plant the proper amount of water at the proper time.

- PD6. The system must have a valve to allow water to flow and stop. This valve will be monitored by [m_water_valve_position](#) and controlled by [c_water_valve_position](#).

Rationale: The system must have a valve to enable the water to flow and stop it from flowing, so each plant can receive the proper amount of water and no water is dropped outside of the soil.

4.3. Environment Requirements

- E1. The robot must be able to detect obstacles in its path.

Rationale: The system must be able to detect obstacles in its path towards the plants and ignore any obstacles that will not affect its movement, so it could try navigating around the obstacles that will affect its movement on its current path and continue on its path to the plants if the obstacles will not affect its movement on the path.

- E2. The robot must ensure [m_obstacle_distance](#) is greater than [MINIMUM_DISTANCE](#).

Rationale: The system must be able to keep a minimum distance of [MINIMUM_DISTANCE](#) from all obstacles, so nothing affects its movement and causes any spillages or harm to the environment around it.

4.4. Human Requirements

- H1. The system must allow the user to set a schedule for watering each plant at different times.

Rationale: The user must be able to set a unique watering schedule for their own plants based on their preference/needs.

- H2. There must be a way for the user to control the robot's plant watering path.

Rationale: The user can configure a path for the robot to follow that is not obstructive.

5. Safety Requirements

Mitigation strategies identified in the Hazard Analysis document must be converted to safety requirements. Similar to the functional requirements, they will be identified based on their component (robot, plant, environment, human).

The numbering for each safety requirement will also correspond to the specific hazard as identified in the Hazard Analysis. For example, **SSR1** (Safety Requirement for Robot 1) will correspond to **HR1** (Hazard Robot 1). Rationales for these requirements are also provided in the Hazard Analysis document.

5.1. Safety Requirements for Robot (SRR)

- SRR1. The system must have a water-resistant casing around the robot and any electrical components.
- SRR2. The system must ensure the water dispersal arm is directly over the soil ([m_watering_arm_location](#)) before it is able to release any water.
- SRR3. Same as SRR1.
- SRR4. The robot must avoid collisions with all obstacles around it, and stop immediately if a collision occurs.
- SRR5. The system must identify and reroute accordingly if a plant is relocated.
- SRR6. The system must have a feedback system with sensor redundancy to ensure its position is accurate.
- SRR7. The system must have a rigid movement system to avoid slipping on slick surfaces or transitioning from different floor surfaces, using high friction wheels, treads or points of interaction.

- SRR8. The system must alert the user if its expected position (c_robot_location) does not match its actual position (m_robot_location).
- SRR9. The system must alert the user in the case of a collision or if a collision cannot be avoided.
- SRR10. Same as SRR7.
- SRR11. The system must stop and alert the user in the case of reoccurring collisions.
- SRR12. The system must alert the user if it is unable to move.
- SRR13. The system must have multiple sensors for redundancy should one fail.
- SRR14. The system must detect if it is upright and notify the user if it is not.
- SRR15. The system must stop its movement if it receives a force greater than MAX_FORCE.
- SRR16. The system must notify the user when m_battery_level_voltage reaches BATTERY_LEVEL_LOW.
- SRR17. The system must complete a test at start up to ensure all components are connected and receive power.
- SRR18. The system will require a method for authenticating the user, such as a user/password system.
- SRR19. The system must operate offline using the last schedule set by the user.

5.2. Safety Requirements for Plant (SRP)

- SRP1. The system must correctly recognize the water needed by the different plants.
- SRP2. The system must be able to correctly identify (m_plant_id) and recognize different plants in the environment.
- SRP3. The system must follow a predefined watering schedule for the different plants.
- SRP4. The system must have a way to shut off the watering system at the correct time and not allow any more water than intended (c_water_valve_position).
- SRP5. The system must have a way to determine if the watering system/pump is not working properly.
- SRP6. Using multiple sensors to ensure the water dispersal mechanism is located above the soil before dispensing any water.
- SRP7. Same as SRR16.
- SRP8. Implement software checks to move robot (c_robot_location) and reorient (c_robot_orientation) itself until a plant is located.
- SRP9. Same as SRR9.

5.3. Safety Requirements for Environment (SRE)

- SRE1. Same as SRR4.
- SRE2. The system must have multiple valves for redundancy should one fail.
- SRE3. The system must have a way to detect if there are any water present on the ground.

5.4. Safety Requirements for Human (SRH)

- SRH1. The system must be big enough that it is easily visible from a distance.
- SRH2. The system must be able to notify the user if there is any water present on the ground.
- SRH3. The system must use electrically insulating materials.
- SRH4. The system must have a way for the user to monitor the results of the subsystems, view information about different types of plants, and use all the other features.
- SRH5. The system must have a manual way to shut down the robot in an emergency.

6. Non-Functional Requirements

- NF1. The robot must follow watering schedules accurately.
- NF2. The system must report reliable plant water levels, [m plant water levels](#).
- NF3. The system must report reliable reservoir water levels, [m reservoir level](#).
- NF4. The robot must notify users about events within a maximum duration of [MAXIMUM RESPONSE TIME](#).
- NF5. The system must be able to recognize and scale to watering a varying number of plants.
- NF6. The system must ensure efficiency in its use of power from the power supply.
- NF7. The system's various components must be easy to maintain and replace when needed.
- NF8. The system's must be kept secure.
- NF9. The connection between the system interactions and the user must be stable and reliable.
- NF10. The system must be capable of adapting to minor deviations in [m plant location](#).
- NF11. The system must be capable of adapting to deviations in an obstacle position.
- NF12. There must be a way that the user can monitor the system.
- NF13. There must be a way that the user can obtain information about their plants.

7. Operational Scenarios

7.1. Normal Operation

There are various possibilities to the robot's normal operational scenarios. Here is an ideal operational scenario for when the robot autonomously waters plants around the house, when the user is away.

Pre-Conditions:

- The plants are placed in specific areas and are reachable by the robot.
- A watering schedule has been set by the user.
- The robot has executed the watering operation at least once, successfully.
- The robot is located at a predetermined initial position.
- The robot has enough water to complete its operation.

Scenario:

1. The system reaches a scheduled watering time.
2. The robot starts moving from its initial position, [BASE LOCATION](#).
3. The robot navigates to the first plant to be watered.
4. The robot avoids obstacles along its path when moving.
5. The robot reaches the plant.
6. The robot aligns the watering mechanism with the plant's soil.
7. The system pumps the required amount of water.
8. The system repeats the navigation and watering process with other plants if present.
9. The robot returns to its final position, [BASE LOCATION](#).

Success Post-Conditions:

- Plants that require water according to the schedule have been watered.
- The robot causes none to minimal environmental damage.
- The robot reaches its predetermined final position.

7.2. Undesired Operations & Handling

There are various events that could occur, which should not have. A few of these events are given below, along with the desired behaviour, their consequences and how they will be handled.

7.2.1. No path is available

Under desired conditions, the robot has executed a watering operation at least one successfully to ensure all plants are reachable. However, it is possible that the user could skip a test run or that from the time of this test run to the time of use, new obstacles arise preventing the robot from reaching the plant. If the robot is unable to reach a plant after various rerouting attempts and deems it not possible, a notification will be sent to the user expressing that the device cannot reach its plant.

7.2.2. Insufficient water available

A condition may arise where the robot has an insufficient amount of water for its scheduled operation. The system should be filled each time the user goes away, however if they are gone long enough it's possible that after the system will have used all the water. If the robot is scheduled to begin a run but does not have the necessary water, it will complete what it can with the water remaining. When no water remains, it will inform the user that it is unable to complete the task.

7.2.3. Foliage in the way

When watering the soil, the plant's leaves may obstruct the flow of water from all directions, and there may be a very specific direction from which there is an opening. For this scenario, the robot will try to use a little bit of force to attempt in pushing away the foliage, in attempt to reach the soil. Failing that, the robot will have to skip the plant if it cannot find an angle to the soil from any direction and inform the user.

7.2.4. System tips over

There could exist a scenario in which the robot tips over. Some reasons for this include a failed collision detection or human/pet interference. If the system is no longer upright, it must inform the user that it fell over and is unable to complete any tasks. The system will also shut down to conserve battery.

7.2.5. Loss of communication

There is a possibility of the robot failing to communicate with any external components or actors, such as the user. While the user may not be notified of any issues, the system must keep a log of all events that has occurred and inform the user about these events once a connection has been re-established.

8. Requirement Modifications

8.1. Requirements Likely to Change

RM4, RM5 – Depends on pathing implementation, the BASE LOCATION may not be used.

PM1, PD1 – Redundancy of water volume detection depends on budget restrictions and may change.

PD6, SRE2 – Redundancy of valves is dependant on budget restrictions.

SRR5 – Whether the plant reroutes automatically will depend on the routing and navigation implemented. It may be removed from requirements.

SRR6, SRR12 – Redundancy of sensors and ability to detect certain feedback will depend on budget restrictions.

SRP5 – The requirement to confirm if the pump works correctly may be dependent on budget restrictions, if it can't be determined through other means.

SRE3, SRH2 – Water spills may not have a major impact on the system in order to mitigate it.

8.2. Requirements not likely to change

The requirements not listed in **8.1. Requirements Likely to Change** are not likely to change.

9. References

- [1] S. Datta, "What Is Functional Decomposition?," Baeldung, <https://www.baeldung.com/cs/functional-decomposition> (Accessed Nov. 12, 2023).
- [2] D. L. Lempia and S. P. Miller, *Requirements Engineering Management Handbook*. Washington, DC: U.S. Department of Transportation, 2009.

Feedback Integration

TA Feedback	Integration Comments
Provide rationale alongside the requirements.	Rationale for each requirement has been moved to an indented point underneath the corresponding requirement.
Requirements should refer to the monitored variables and how they map to the controlled variables.	Monitored and controlled variables have been mentioned alongside the requirements that require the variable.
Isn't clear how you would consider them completed or how you would test them.	Requirements have been reworded. Completion and testing is described further in V&V.

Table 5: Feedback Integration