Software Requirements Specification for ParkingLotHawk

MECHTRON 4TB6 Capstone Design Project

Team #34
Fady Zekry Hanna, zekryhf
Winnie Trandinh, trandint
Muhammad Ali, alim102
Muhammad Khan, khanm120

October 6, 2022

Contents

Revision History

Date	Version	Notes
Oct 5, 2022	1.0	Team Finalized SRS

1 Introduction

1.1 Purpose

The purpose of the project will be to design an aerial drone, called ParkingLotHawk, that can detect how many parking spots are available in any given parking lot. ParkingLotHawk can be operated by property personnel to communicate to drivers on how many spots are available at a parking lot. ParkingLotHawk will explore over any parking lot designated by the user and will aggregate visual information and output the amount of parking spots available. Many parking lots today do not have this data and this data can be used for many reasons. The data can be used by retailers to know how long customers stay at specific stores or location. This data can also be used for drivers to know if there are parking spots available in a specific location allowing for drivers to not waste time and resources in a full parking lot.

1.2 Scope

The product specified in this SRS 1 is about an autonomous aerial drone for helping parking lot operators understand the state of their parking lot. The product specified does not require the operator to manually control or move the drone, rather the requirement describes various autonomous flight modes. The specified drone shall support the ability to both create a path to reach its specified location, as well as create and follow a path to explore large parking lot sections autonomously. During flight, the specified drone shall transmit live information about the parking lot sections it detects. The completed product, a combination of the physical drone and any equipment/application intended to be kept by the parking lot operator to communicate with the drone, is called the ParkingLotHawk. A solution that implements the requirements will help parking lot authorities of outdoor lots quickly gain valuable information without requiring permanent solution, large monetary and time investments, or complex training.

1.3 Definitions, Acronyms, and Abbreviations

Table 1: Key terms, acronyms, and abbreviations are defined.

Term	Definition.
Inclement Weather	Weather with rain, snow, fog, and/or winds over 50 km/hour.
Parking Lot Authorities	Event organizers for whome parking lot management is one of thier tasks. Examples include property managers, and concert organizers.
Parking Lot Operator (Operator)	Person responsible for managing the product during operation.
Clear Boundaries	The border surrounding the parking lot. For example the border between the lot and sidewalks, buildings, curbs, and sections of grass.
Flight Controller	The central part of the drone that receives most sensor input and is responsible for controlling the propellers.
Flight States	The states of the state machine that require the drone to be aerially flying, detailed in the requirements 2.2.10.
PC	Personal Computer.
MVP	Minimum Viable Product.
SRS	Software Requirement Specifications.

1.4 References

There are no external software documents referenced in the SRS.

1.5 Overview

The SRS is organized to follow the IEEE 1998 template. Introduction contains the purpose of the SRS as well as the scope of the product and the problem it solves. Overall Description refines the scope of the product further. It provides more detail about the products environment, primary functions, intended users, constraints, and finally assumptions. Specific requirements contains a detailed description of all requirements, organized into sections for readability. Finally, the Appendixes contains Appendix A: Reflection regarding new knowledge and skills the team needs to create the specified product, along with approaches to how the team will acquire the knowledge. The Appendix B: Formal Transition Table also contains the formal transition tables between the states of the finite state machine which will be later introduced within Product Functions.

2 Overall Description

General factors that affect the product and the requirements are described in the following subsections. A high-level overview of the product functions are also described (see Product Functions).

2.1 Product Perspective

The system specified is an independent and stand-alone parking lot tool. It does not fit into or interface with a larger system of parking lot and security technologies the operator may have available.

The environment consists of an outdoor parking lot and the operator's PC. The operator's PC shall be operating with Windows 10 or Windows 11.

2.2 Product Functions

This subsection describes the behaviour overview of the ParkingLotHawk by splitting its functionality into various interconnected states of operation. These states are shown within the Informal Finite State Machine Diagram and are further refined within Functional Requirements.

2.2.1 Idle State

The product is powered on and communicating with the operator application, but all motors are turned off.

2.2.2 Hover State

The product maintains its current longitudinal and latitudinal position, and hovers at a pre-defined operator selected height.

2.2.3 Manual Location Move State

The product shall move to an operator specified location, and hover at a pre-defined operator selected height. Note that this height does not need to be the same as the height specified in the Hover State.

2.2.4 Autonomous Explore State

The product shall autonomously explore within the current parking lot without operator input.

2.2.5 Configure State

During this state, the operator is able to define parameters of operation that are not configurable in other states.

2.2.6 Off State

The product shall be unpowered and stops communication with the operator application.

2.2.7 Land State

The product shall land at its initial launch location.

2.2.8 Desired Location Error State

This error state occurs when the operator specifies an invalid desired position and shall provide the appropriate error message to the operator.

2.2.9 No Parking Lot Detected Error State

This error state occurs when the product does not detect a valid parking lot, and shall provide the appropriate error message to the operator.

2.2.10 Malfunction State

This error state occurs when an internal/external malfunction occurs in such a way that nominal performance is not possible. The resulting action by the product will be further specified within Functional Requirements.

The normal modes of operation include the following:

- Idle State
- Hover State
- Manual Location Move State
- Autonomous Explore State
- Configure State
- Off State

The flight modes of operation include the following:

- Hover State
- Manual Location Move State
- Autonomous Explore State
- No Parking Lot Detected Error State
- Desired Location Out of Bounds Error State

With the exception of the Configure State and Off State, the product shall update continuously update c_CurrentView and c_OccupancyMap for the operator. These requirements are further refined within Functional Requirements.

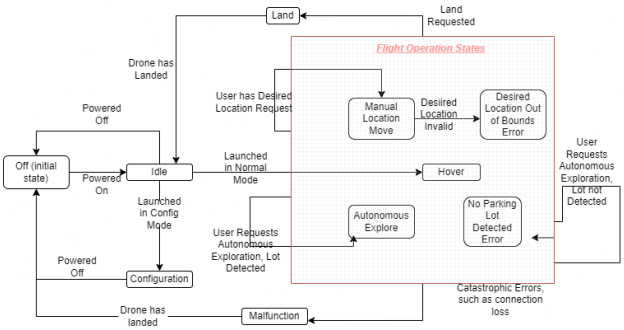
The error modes are listed below:

- Desired Location Error State
- No Parking Lot Detected Error State
- Malfunction State

These modes are used to handle errors either from the operator, or internally from the product.

The behaviour overview of the product is described below, with each directional arrow representing a transition between states:

Figure 1: Informal Finite State Machine Diagram



2.3 User characteristics

The stakeholders are Dr. Spencer Smith, the MECHTRON 4TB6 teaching assistants, parking lot authorities 1, parking lot operators, and visitors of the parking lot. All demographics mentioned would find the data of the ParkingLotHawk useful. There is no technical or software knowledge expected for users of the ParkingLotHawk. The user will need to turn the drone on and place it close to a parking lot. The ParkingLotHawk is made mindful of the society and community and its health impacts; therefore, little to no air or noise pollution shall occur, and no invasion of privacy shall ensue.

2.4 Constraints

The purpose of the system is to provide parking lot information to parking lot operators, who in turn could use that information to decrease the amount of time visitors spend to find a parking spot. As the user can come from a non-technical background, the constraints on the usability of the product should be considered. If the system chooses to use radio communication between the operator's laptop and the product, it must abide by national radio frequency regulations of 2.4MHz. Furthermore, radio communication can only work within approximately 2 km from one point to the other while still abiding by the national regulations. The project constraint present is a maximum budget of $ 750. Canadian regulatory policy does not allow for drone flight within 1 nautical mile (about 2 km) from heliports and 3 nautical miles (5.6 km) from airports[1]. If the drone weighs over 25 kg, the team will need to get special permission from Transport Canada before flying the drone[2]; therefore, the product should be under a weight of 25kg to support widespread adoption.

2.5 Assumptions

The assumptions of the project are:

- Operator does not fly the drone exceeding a specified amount of time.
- Birds do not interfere with the drone.
- Operator's PC has a Windows 10 or 11 OS.
- Parking lot lines are visible to the naked eye.
- Operation done under non-inclement weather1.

2.6 Apportioning of Requirements

The Phase in Plan is composed of three main releases:

- Phase I: Proof of Concept - November 14, 2022
- Phase II: Revision 0 (Minimal Viable Product (MVP)) - February 6, 2023
- Phase III: Revision 1 - March 27, 2023

Each requirement will then be assigned to one of these phases within Specific requirements, indicated by their Phase.

3 Specific requirements

This section of the SRS contains all requirements of the product in order to further refine the scope of the product.

3.1 External Interfaces

Input Variables: Input variables are set/configured before any flight operations are entered 2.2.10 . They are constant throughout the drones flight operation 2.2.10 .

Figure 2: System Context Diagram

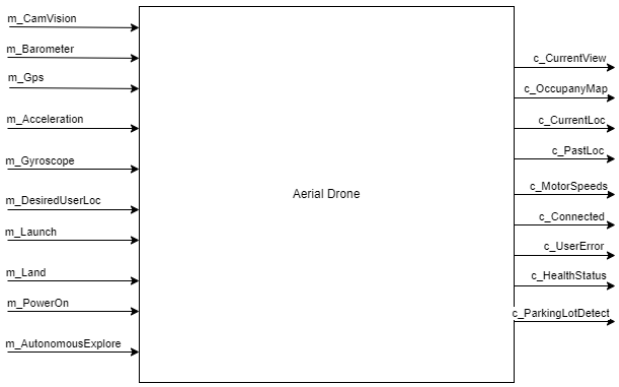


Table 2: Input Variables

Variable Name	Type	Unit	Description
i_Mode	Enumeration	0 - Normal, 1 - Configure	Set by the user to choose whether they want to configure internal parameters or if they want regular operation.
i_MinHoverHeight	float	m	Minimum aerial hover height set by user
i_MaxHoverHeight	float	m	Maximum aerial hover height set by user
i_DesiredHover-Height	float	m	Desired aerial hover height set by user

Monitored Variables are variables that are continuously monitored by the product.

Table 3: Monitored Variables

Variable Name	Type	Unit	Description
m_Acceleration	Vector	m/s^2	three-dimensional vector containing acceleration relative to frame of the drone.
m_Gyroscope	Vector	Rad	three-dimensional vector containing orientation relative to frame of the drone.
m_Gps	Tuple	GPS coordinates (degrees-minutes-seconds (DMS), height (m))	current GPS coordinates of the drone with height in the second tuple.
m_Barometer	Float	atm	Altitude detection using atmospheric pressure measurement
m_DesiredUserLoc	GPS Location	GPS coordinates (DMS)	Desired location of the aerial drone set by user.
m_Launch	Boolean	-	Indicates if the operator desires the drone to begin operation and turn on all peripherals.
m_Autonomous-Explore	Boolean	-	Indicates if the operator desires the drone to autonomously explore the parking lot.
m_PowerOn	Boolean	-	Indicates if the operator desires the drone to be On or Off.
m_Land	Boolean	-	Indicates if the operator desires the drone to land.
m_CamVision	Image	Array of Pixels	Latest image of the section of the parking lot currently visible to the drone.

Controlled Variables are variables that are outputted by the system. Some are visible to the operator on their application, while others help to indirectly accomplish functional requirements.

Table 4: Controlled Variables

Variable Name	Type	Unit	Description
c_CurrentView	Image	Array of Pixels	Live visual display of parking lot section the drone's currently sees, as well as any further annotations and text.
c_OccupancyMap	Image	Array of Pixels	Map of available parking spots based on the drone's previous paths.
c_CurrentLoc	Tuple	{GPS coordinates (DMS), height (m)}	GPS coordinates are stored at the first index, height is stored in the second index. Estimated longitudinal coordinate, lateral coordinate and height of the drone.
c_PastLoc	Vector	1/ Degrees, minutes, and seconds (DMS)	Trace of the drone's location in the past 60 seconds (vector of GPS locations).
c_MotorSpeeds	Vector	rad/s^2	n-dimensional vector containing the motor speeds of however many motors the drone chooses to use (2 for helicopter, 4 for quadcopter, 6 for hexcopter, etc.). The vector contains speeds of each motor clockwise from front of the drone.
c_Connected	Boolean	-	Indicates if connection between the drone and the operator's application is established
c_ParkingLotDetected	Boolean	-	Indicated if a parking lot is detected in the c_CurrentView.
c_UserError	Enumeration	0 - None, 1 - Desired_Location_Out_Of_Bounds, 2 - No_Lot_Detected_State	Indicates if a command the user requested is not feasible.
c_HealthStatus	Enumeration	0 - Healthy, 1 - Unhealthy	Indicates if the drone's mechanical and electrical state allows it to safely fly. For example if there is mechanical damage, the value should be Unhealthy.

3.2 Functional Requirements

The following requirements are required in order to achieve the product's functions, as stated in Product Functions.

3.2.1 General Functional Requirements

Table 5: GEN_001

Description	The product shall be able to recognize Clear Boundaries 1. This requirement is a refinement of the Autonomous Explore State.
Rationale	This requirement ensures that the product is able to implement basic autonomy, such as not traveling past the parking lot boundaries.
Phase	II
Likely to Change	No. This requirement is required to implement the Autonomous Explore State.
Associated Inputs and Outputs	m_CamVision.

Table 6: GEN_002

Description	The product shall provide live update of c_CurrentLoc, c_CurrentView and c_OccupancyMap during all normal and non-configurational operation states. This requirement is a refinement of the normal and non-configuration operation states specified in Section 2.2.
Rationale	This requirement ensures that the product always provides the latest controlled variable information to the operator.
Phase	II
Likely to Change	No. This requirement is a part of the MVP and must be present to make the product achieve its product functions.
Associated Inputs and Outputs	m_CamVision, c_CurrentLoc, c_CurrentView, and c_OccupancyMap.

Table 7: GEN_003

Description	The product shall allow the operator to configure the i_MinHoverHeight, i_MaxHoverHeight, and i_DesiredHoverHeight. This requirement is a refinement of the Configure State.
Rationale	The value of these parameters depends on the operators view preferences and parking lot conditions. For example a parking lot with a lot of large trucks may be better suited to higher hovering heights.
Phase	I
Likely to Change	No. This requirement is vital to the operation of the product, as it must be suited to different parking lot environments.
Associated Inputs and Outputs	i_MinHoverHeight, i_MaxHoverHeight, and i_DesiredHoverHeight.

Table 8: GEN_004

Description	The condition i_MinHoverHeight <= i_DesiredHoverHeight <= i_MaxHoverHeight shall always be true. This requirement is a refinement of the Configure State.
Rationale	This requirement ensures logical values for the parameters are set by the operator.
Phase	I
Likely to Change	No. This requirement is required to check the inputted values by the operator.
Associated Inputs and Outputs	i_MinHoverHeight, i_DesiredHoverHeight, and i_MaxHoverHeight.

Table 9: GEN_005

Description	The product shall be able to identify non-occupied parking spots. This requirement is a refinement of the normal and non-configuration operation states specified in Section 2.2.
Rationale	This requirement ensures that the product is able to create the occupancy map.
Phase	III
Likely to Change	No. This requirement is required to create the occupancy map (c_OccupancyMap), which is one of the main functions of the product during the Phase III Release.
Associated Inputs and Outputs	m_CamVision, c_ParkingLotDetected, and c_OccupancyMap.

Table 10: GEN_006

Description	The product shall be shall highlight non-occupied parking slots on the operator's display (update c_CurrentView). This requirement is a refinement of the normal and non-configuration operation states specified in Section 2.2.
Rationale	This requirement ensures that the product is able to create the occupancy map.
Phase	III
Likely to Change	No. This requirement is a key feature of the occupancy map in order to efficiently communicate the data to the operator.
Associated Inputs and Outputs	m_CamVision, c_CurrentView, c_ParkingLotDetected, and c_OccupancyMap.

3.2.2 State Implementation Requirements

Table 11: STA_000

Description	The product shall implement an Idle state. During this state, the solution is powered on but motors are stationary. This requirement is a refinement of the Idle State, and further refined by PERF_007, SAFE_001, SAFE_003, and USE_003.
Rationale	This state is used to ensure that the operator can safely hold the drone and access the mechanical switch that controls m_PowerOn.
Phase	I
Likely to Change	No. This requirement is required for the safety of the operator.
Associated Inputs and Outputs	m_PowerOn and c_Connected.

Table 12: STA_001

Description	The product shall implement a Hover State. During this state, the solution shall fly and hover to height i_MaxHoverHeight. The drone shall keep the same lateral location it is currently at. This requirement is a refinement of the Hover State and is further refined by PERF_002, PERF_004, PERF_005, PERF_006, PERF_007, SAFE_001, SAFE_003, and USE_003.
Rationale	This state is used for when the product is waiting for further operator commands. Hover height is selected to be i_MaxHoverHeight, so that the drone can see as much of the parking lot section as it can. This makes the transition to Autonomous Explore State more likely.
Phase	II
Likely to Change	No. This requirement is a key feature of the MVP, and is required in order to gather any useful information to the operator.
Associated Inputs and Outputs	m_Barometer, m_Acceleration, m_Gyroscope, m_Gps, i_MaxHoverHeight, and c_MotorSpeeds.

Table 13: STA_002

Description	The product shall implement a Manual Location Move State. During this state, the drone moves to the m_DesiredUserLoc, and hovers at that location with height i_DesiredHoverHeight. This requirement is a refinement of the Manual Location Move State and is further refined by PERF_003, PERF_004, PERF_006, PERF_007, SAFE_001, SAFE_003, and USE_003.
Rationale	This state is used for when the the product needs to provide the operator the ability to move the drone to a specific location.
Phase	II
Likely to Change	No. This requirement is a key feature of the product, and is required in order for the operator to make changes to the product's path.
Associated Inputs and Outputs	m_DesiredUserLoc, m_Barometer, m_Acceleration, m_Gyroscope, m_Gps, i_DesiredHoverHeight, and c_MotorSpeeds.

Table 14: STA_003

Description	The product shall implement an Autonomous Explore State. During this state the drone will create its own path to explore and remain within the parking lot it currently detects. This requirement is a refinement of the Autonomous Explore State and is further refined by PERF_001, PERF_004, PERF_007, SAFE_001, SAFE_003, and USE_003.
Rationale	This state is used for when the operator does not need to constantly instruct the drone to move.
Phase	III
Likely to Change	No. This requirement is an important part of the Phase III Release, as it allows for automated creation of the occupancy map.
Associated Inputs and Outputs	m_Barometer, m_Acceleration, m_Gyroscope, m_Gps, m_AutonomouseExplore, m_CamVision, c_CurrentView, and c_MotorSpeeds.

Table 15: STA_004

Description	The product shall implement a Configure state. During this state settings and parameters that cannot be changed during flight can be changed. The Input Variables i_MinHoverHeight, i_MaxHoverHeight and i_DesiredHoverHeight can be changed in this state. The product is powered on but motors are stationary. This requirement is a refinement of the Configure State, and references GEN_003. This is further refined by PERF_004, PERF_007, SAFE_001, SAFE_003, and USE_003.
Rationale	This state is used to allow parameters that are unsafe to change during flight operation, to be safely changed through a special process. During this state the operator can safely hold the drone. Such parameters are outlined in Input Variables.
Phase	II
Likely to Change	No. This requirement is required to ensure safety of the operator, as well as the product from unsafe changes during operation.
Associated Inputs and Outputs	i_Mode, i_MinHoverHeight, i_MaxHoverHeight, and i_DesiredHover-Height.

Table 16: STA_005

Description	The product shall implement an Off state. All modules are powered off. No battery power is consumer. c_UserError is set to None, c_-HealthStatus is set to Unhealthy, and c_Connected is set to false. All values in the matrices c_MotorSpeeds, c, c_OccupancyMap, and c_-CurrentView are set to 0. This requirement is a refinement of the Off State and is further refined by PERF_004, PERF_007, SAFE_001, and SAFE_003
Rationale	This state is used to explicitly state what it means for the drone to be off.
Phase	I
Likely to Change	No. This requirement is required for the safe operation, transport, and handling of the product.
Associated Inputs and Outputs	c_UserError, c_HealthStatus, c_Connected, c_MotorSpeeds, c, c_-OccupancyMap, and c_CurrentView.

Table 17: STA_006

Description	The product shall implement a Land state. In the land state the solution first travel laterally to the initial launch location, and then lands vertically downward. Once physically landed, the drone enters the Idle state. This requirement is a refinement of the Land State and is further refined by PERF_002, PERF_004, PERF_006, PERF_007, SAFE_001, SAFE_003, and USE_003.
Rationale	This state is used to explicitly designate a landing path and command.
Phase	II
Likely to Change	No. This requirement is required for stopping the operation of the product.
Associated Inputs and Outputs	m_Barometer, m_Acceleration, m_Gyroscope, m_Gps, and c_MotorSpeeds.

Table 18: STA_007

Description	The product shall implement an Desired Location Error state. Upon entry to this state, the c_UserError variable is set to Desired_Location_Out_Of_Bounds. The drone proceeds to Hover at its current location. Upon exit of this state, the drone shall set c_UserError to None. This requirement is a refinement of the Desired Location Error State and is further refined by PERF_007, SAFE_001, SAFE_003, and USE_003.
Rationale	This state is used to indicate explicitly that the operator's request cannot be met.
Phase	II
Likely to Change	No. This requirement is required for clear communication with the operator to handle unsupported requests.
Associated Inputs and Outputs	m_Barometer, m_Acceleration, m_Gyroscope, m_Gps, c_UserError, and c_MotorSpeeds.

Table 19: STA_008

Description	The product shall implement a No Parking Lot Detected Error state. Upon entry to this state, the c_UserError variable is set to No_Lot_Detected_State. The drone proceeds to Hover at its current location. Upon exit of this state, the drone shall set c_UserError to None. This requirement is a refinement of the No Parking Lot Detected Error State and is further refined by PERF_007, SAFE_001, SAFE_003, and USE_003.
Rationale	This state is used to indicate explicitly that the product cannot detect a parking lot.
Phase	II
Likely to Change	No. This requirement is required for clear communication with the operator to handle unsupported requests.
Associated Inputs and Outputs	m_Barometer, m_Acceleration, m_Gyroscope, m_Gps, c_UserError, and c_MotorSpeeds.

Table 20: STA_009

Description	The product shall implement a Malfunction state. During this state the drone sets the c_HealthStatus to Unhealthy. It then tries to land at its launch location, which if is not possible the drone instead lands vertically on the land below. After landing the drone enters the Off state. This requirement is a refinement of the Malfunction State and is further refined by PERF_007, SAFE_001, SAFE_003, and USE_003.
Rationale	This state is used to ensure that the product can handle large malfunctions that can occur during operation.
Phase	II
Likely to Change	No. This requirement is required to handle malfunctioning components during operation.
Associated Inputs and Outputs	m_barometer, m_acceleration, m_gyroscope, m_gps, c_healthStatus, and c_motorSpeed.

3.2.3 State Transition Requirements

The following requirements are refinements of all the states specified in Section 3.2.2, and specifies the product's transitions between states. These requirements will not be changed unless the states are changed.

Table 21: TRANS_001

Description	Upon the m_PowerOn becoming false, the drone shall enter the Off state. This requirement references the Off State.
Rationale	This requirement ensures that the product can turn off safely.
Phase	I
Associated Inputs and Outputs	m_PowerOn.

Table 22: TRANS_002

Description	Upon the m_PowerOn becoming true, the drone shall enter the Idle state. This requirement references the Idle State and the requirement SAFE_005.
Rationale	This requirement ensures that the propellers do not damage the operator after touching the m_PowerOn switch.
Phase	I
Associated Inputs and Outputs	m_PowerOn.

Table 23: TRANS_003

Description	Upon the m_Launch becoming true, the drone shall enter the Hover state if the i_Mode was set to normal, and enters the Configure state if the i_Mode was set to configure. This requirement references the Hover State and the Configure State.
Rationale	This requirement facilitates the setup and configuration of the product.
Phase	II
Associated Inputs and Outputs	m_Launch and i_Mode.

Table 24: TRANS_004

Description	If in the Hover state, and c_ParkingLotDetected is equal to true, the product shall enter the Autonomous Explore state and explore the detected lot. If two parking lots are detected at the same time, it arbitrarily picks one to explore. This requirement references the Hover State and the Autonomous Explore State.
Rationale	This requirement ensures that the default mode of operation is Autonomous Explore after entering the Hover state.
Phase	III
Associated Inputs and Outputs	c_ParkingLotDetected.

Table 25: TRANS_005

Description	Once the user enters or changes m_DesiredUserLoc, the drone shall automatically enters the Manual Location Move state. This requirement references the Manual Location Move State.
Rationale	This requirement ensures that the product shall take the operator's request with higher priority than any other kind of operation.
Phase	II
Associated Inputs and Outputs	m_DesiredUserLoc.

Table 26: TRANS_006

Description	If while in the Manual Location Hold state and the product determines that m_DesiredUserLoc is outside parking lot boundaries, the product shall enter the Desired Location Error state. This requirement references the Manual Location Move State and the Desired Location Error State.
Rationale	This requirement ensures that the product shall notify any issues with the operator's request at the moment it is detected, so that the operator can make adjustments.
Phase	II
Associated Inputs and Outputs	m_DesiredUserLoc.

Table 27: TRANS_007

Description	When m_AutonomousExplore is set to true and c_ParkingLotDetected is equal to true, the product shall enter the Autonomous Explore state. This requirement references the Autonomous Explore State.
Rationale	This requirement ensures that the product shall take the operator's request with higher priority than any other kind of operation.
Phase	II
Associated Inputs and Outputs	m_AutonomousExplore and c_ParkingLotDetected.

Table 28: TRANS_008

Description	When m_AutonomousExplore is set to true but c_ParkingLotDetected is equal to false, the product shall enter the No Parking Lot Detected Error state. This requirement references the No Parking Lot Detected Error State.
Rationale	This requirement ensures that the product shall notify any issues with the operator's request at the moment it is detected, so that the operator can make adjustments.
Phase	II
Associated Inputs and Outputs	m_AutonomousExplore and c_ParkingLotDetected.

Table 29: TRANS_009

Description	Upon m_Land being true, the product shall enter the Land state. This requirement references the Land State.
Rationale	This requirement ensures that the product is able to land at the requested time.
Phase	II
Associated Inputs and Outputs	m_Land.

Table 30: TRANS_010

Description	If c_Connected becomes false for more than 5 seconds at any point during operation, then the product shall enter the Malfunction state. This requirement references the Malfunction State.
Rationale	This requirement ensures that the product detects connectivity errors, and is able to handle such occurrences.
Phase	I
Associated Inputs and Outputs	c_Connected.

Table 31: A tractability table linking non-functional requirements to the functional requirements they reference. If a requirement has no dependencies or relationships, then it is not mentioned.

Functional Requirement	Non-functional requirement
STA_000	PERF_007, SAFE_001, SAFE_003, USE_003
STA_001	PERF_002, PERF_004, PERF_005, PERF_006, PERF_007, SAFE_001, SAFE_003, USE_003
STA_002	PERF_003, PERF_004, PERF_006, PERF_007, SAFE_001, SAFE_003, USE_003
STA_003	PERF_001, PERF_004, PERF_007, SAFE_001, SAFE_003, USE_003
STA_004	PERF_004, PERF_007, SAFE_001, SAFE_003, USE_003
STA_005	PERF_004, PERF_007, SAFE_001, SAFE_003
STA_006	PERF_002, PERF_004, PERF_006, PERF_007, SAFE_001, SAFE_003, USE_003
STA_007	PERF_007, SAFE_001, SAFE_003, USE_003
STA_008	PERF_007, SAFE_001, SAFE_003, USE_003
STA_009	PERF_007, SAFE_001, SAFE_003, USE_003

Table 32: A tracability table linking functional requirements to the functions they refine. If a requirement has no dependencies or relationships, then it is not mentioned.

Functional Requirement	Function
GEN_001	Autonomous Explore State
STA_000	Idle State
STA_001	Autonomous Explore State
STA_002	Manual Location Move State
STA_003	Autonomous Explore State
STA_004	Configure State
STA_005	Off State
STA_006	Land State
STA_007	Desired Location Error State
STA_008	No Parking Lot Detected Error State
STA_009	Malfunction State

3.3 Performance Requirements

Table 33: PERF_001

Description	The product shall explore > 90% of the detected parking lot during the Autonomous Explore State. This is a refinement of STA_003.
Rationale	The requirement ensures that Autonomous Explore State is able to accurately survey the majority of the parking lot in order to provide accurate information to the operator.
Phase	III
Likely to Change	Yes. Based on algorithm performance of autonomous exploration and the limitations of the hardware, the threshold may be required to change.
Associated Inputs and Outputs	m_barometer, m_acceleration, m_gyroscope, m_gps, m_AutonomousExplore, and c_MotorSpeeds.

Table 34: PERF_002

Description	The product shall takeoff to i_MaxHoverHeight and land from i_MaxHoverHeight within 25 seconds. This requirement are refinements of STA_001 and STA_006.
Rationale	The requirement ensures that minimal time is required for the product to start transmitting useful data to the operator, and that it does not take excessive time to cease operation once finished.
Phase	II
Likely to Change	No. This requirement refines the Minimum Viable Product and should not be modified.
Associated Inputs and Outputs	m_barometer, m_acceleration, m_gyroscope, m_gps, and c_MotorSpeeds.

Table 35: PERF_003

Description	The product shall move to a specified location with an average speed exceeding 4km/hour. This is a refinement of STA_002.
Rationale	The requirement ensures that the operator spends minimal time waiting for the product to move to the desired location, and that it is quicker than the average speed of a person walking.
Phase	II
Likely to Change	No. This requirement refines the movement of the product to ensure that the product is more optimal than a human.
Associated Inputs and Outputs	m_barometer, m_acceleration, m_gyroscope, m_gps, and c_MotorSpeeds.

Table 36: PERF_004

Description	The product shall transmit all data to the operator at a rate exceeding 0.5 frames per second. This is a refinement of the functions specified in Section Product Functions.
Rationale	The requirement ensures the operator is receiving real time data from the product.
Phase	I
Likely to Change	No. This requirement constitutes the MVP, and ensures that the operator is able to receive the information in a timely manner.
Associated Inputs and Outputs	m_barometer, m_acceleration, m_gyroscope, m_gps, m_CamVision, c_CurrentView, c_CurrentLoc, c_PastLoc, and c_MotorSpeeds.

Table 37: PERF_005

Description	The product shall maintain a longitudinal and lateral position within a 1.5m radius during the Hover State. This requirement is a refinement of STA_001.
Rationale	The requirement ensures that the product does not drift to a different location when the operator expects it to be stationary.
Phase	II
Likely to Change	No. This requirement is required to allow proper operation of the product, as excessive drifting will cause inaccurate data to the operator.
Associated Inputs and Outputs	m_barometer, m_acceleration, m_gyroscope, m_gps, c_CurrentLoc, and c_MotorSpeeds.

Table 38: PERF_006

Description	The product shall always maintain an altitude between i_MaxHoverHeight and i_MinHoverHeight, within a tolerance of ±5%. This requirement are refinements of STA_001, STA_002, and STA_003.
Rationale	The requirement ensures that the operator specified altitude limits are respected during operation of the product.
Phase	II
Likely to Change	No. This requirement ensures that the operator's commands are respected during operation.
Associated Inputs and Outputs	m_barometer, i_MaxHoverHeight, i_MinHoverHeight, and c_MotorSpeeds.

Table 39: PERF_007

Description	The product shall be operable within requirements within non-inclement weather 1. This requirement are refinements of all the functions stated in Section Product Functions, and the assumption of the weather in Section Assumptions. This is further refined within requirement MTNC_003.
Rationale	The requirement ensures that the product is usable in the majority of the environmental conditions, such that its availability is maximized.
Phase	III
Likely to Change	Yes. Based on hardware limitations, this requirement may need to be modified to include less adverse weather conditions.
Associated Inputs and Outputs	m_barometer, m_acceleration, m_gyroscope, m_gps, c_CurrentView, c_CurrentLoc, c, and c_MotorSpeeds.

3.4 Logical Database Requirements

There is no requirement for databases, as the problem requires no long-term storage of data.

3.5 Design constraints

Table 40: DES_001

Description	The product shall cost less than $750 to manufacture. This requirement refines the constraint specified in Section Constraints.
Rationale	The requirement ensures the constraint of the design is met, and ensures that it is not costly for users to implement or replace the product.
Phase	I
Likely to Change	No. This requirement must be met in order to satisfy the externally imposed constraint.
Associated Inputs and Outputs	N/A.

3.6 Standards Compliance

Table 41: STD_001

Description	The product shall weight a total of less than 25kg. This requirement refines the constraint specified in Section Constraints.
Rationale	The requirement ensures the constraint of the design is met, and ensures that the product abides by Canadian regulatory laws.
Phase	I
Likely to Change	No. This requirement must be met in order to satisfy the externally imposed constraint.
Associated Inputs and Outputs	N/A.

Table 42: STD_002

Description	The product shall use radio communication only within the 2.4MHz range. This requirement refines the constraint specified in Section Constraints.
Rationale	The requirement ensures the constraint of the design is met, and ensures that the product abides by Canadian regulatory laws.
Phase	I
Likely to Change	No. This requirement must be met in order to satisfy the externally imposed constraint.
Associated Inputs and Outputs	N/A.

3.7 Software System Attributes

There are no requirements regarding the software used in the product, provided that it meets all the other requirements specified within Section 3.

3.8 Reliability

There are no requirements for the reliability of the product, provided that it meets all the other requirements specified within Section 3.

3.9 Availability

There are no requirements for the reliability of the product, provided that it meets all the other requirements specified within Section 3.

3.10 Security

Table 43: SEC_001

Description	The operator's application shall only be launched by a user with authorized access.
Rationale	The requirement ensures that non-authorized users cannot access the product's technology.
Phase	III
Likely to Change	Yes. This requirement may be modified or refined in future revisions of the product.
Associated Inputs and Outputs	N/A.

Table 44: SEC_002

Description	The product shall not upload any gathered data to any external parties.
Rationale	The requirement ensures the user's privacy and the event organizer's privacy.
Phase	II
Likely to Change	Yes. This requirement may be modified in future revisions if the need for off-board computation arises within the software.
Associated Inputs and Outputs	N/A.

3.11 Maintainability

Table 45: MTNC_001

Description	The product shall be fully recharged within 1 hour. This requirement references USE_003.
Rationale	The requirement ensures that the availability of the product is maximized, allowing for frequent uses of the product.
Phase	I
Likely to Change	Yes. This requirement may be limited by hardware requirements of the product. In later revisions, it may be modified to allow easy battery switching in the event that higher availability of the product is required.
Associated Inputs and Outputs	N/A.

Table 46: MTNC_002

Description	The product shall be able to sustain a fall of greater than 1m without sustaining damage that affects operation performance. This requirement references the user classes of the product.
Rationale	The requirement ensures durability. Durability is important because the operators are not technical and may not know how to fix a broken product.
Phase	II
Likely to Change	Yes. This requirement may be limited by hardware requirements of the product. In later revisions, it may be modified to increase or decrease the fall height to better suit the available components.
Associated Inputs and Outputs	N/A.

Table 47: MTNC_003

Description	The product shall be mechanically waterproof, to the point it can sustain a light drizzle for 1 minute of operation while still performing within the requirements. This requirement refines PERF_007.
Rationale	The requirement ensures that if the product experiences a drizzle during flight due to the operator incorrectly assuming the weather was dry, that the product is not severely damaged.
Phase	III
Likely to Change	Yes. This requirement may be limited by hardware requirements of the product.
Associated Inputs and Outputs	N/A.

3.12 Safety

Table 48: SAFE_001

Description	The product shall not influence or interact with dynamic actors positioned in the parking lot. This requirement is a refinement of the product functions stated within Section Product Functions, and is further refined by SAFE_002 and SAFE_004.
Rationale	The requirement ensures the product does not negatively impact the safety of visitors in the parking lot.
Phase	I
Likely to Change	No. One of the fundamental product goals is to assist in parking lot traffic. If the product negatively impacts the traffic, there is no use in the product.
Associated Inputs and Outputs	N/A.

Table 49: SAFE_002

Description	The product shall not allow the operator to set i_MaxHoverHeight, i_MinHoverHeight, or i_DesiredHoverHeight to be below 7m. This requirement is a refinement of SAFE_001.
Rationale	The requirement ensures that the product does not interfere with existing traffic and that the flight path is above the majority of obstacles.
Phase	I
Likely to Change	No. A selected height of below 7m drastically increases the amount of potential obstacles, which will further increase the complexity of the product.
Associated Inputs and Outputs	i_MaxHoverHeight, i_MinHoverHeight, and i_DesiredHoverHeight.

Table 50: SAFE_003

Description	The product shall not require the operator to physically manipulate the product in any way in any states outside of Off State. This requirement are further refinements of all the states outlined in Section 2.2.
Rationale	The requirement ensures that the operator does not interfere with the functions of the product, and that the operator is kept safe from any malfunctions of the product.
Phase	I
Likely to Change	No. To maintain operator safety, the operator must never be required to be near the product during operation. This ensures that in the case of product failure, the operator is not harmed.
Associated Inputs and Outputs	N/A

Table 51: SAFE_004

Description	The product shall not cause distractions or negatively impact greater than 2% of the visitors in the parking lot. This requirement is a refinement of SAFE_001.
Rationale	The requirement ensures that the usage of the product does not negatively impact the visitors in the parking lot.
Phase	I
Likely to Change	No. One of the fundamental product goals is to assist in parking lot traffic. If the product negatively impacts the visitors, there is no use in the product.
Associated Inputs and Outputs	N/A

Table 52: SAFE_005

Description	The product shall include a mechanical Off switch to the product. This requirement is a refinement of the Off State.
Rationale	The requirement ensures that the product can be turned off even in the case of electronic failures, and can double as a kill switch.
Phase	I
Likely to Change	No. A kill switch independent from the electronics is essential to the safe operation of electromechanical products, and must be required within the product.
Associated Inputs and Outputs	N/A

3.13 Usability

Table 53: USE_001

Description	The product shall provide a visual trace of its location for the past 60 seconds +- 1 second. This requirement is a refinement of the c_-CurrentView data.
Rationale	The requirement ensures that the operator is able to keep track of the product's past movement to understand which areas of the parking lot have been explored.
Phase	II
Likely to Change	Yes. Although a method of visualizing the past path is required, the method at which it is done may be changed in future revisions.
Associated Inputs and Outputs	m_barometer, m_acceleration, m_gyroscope, m_gps, c_CurrentView, c_CurrentLoc, and c_PastLoc.

Table 54: USE_002

Description	The product shall allow the operator to save the current visual and raw data into a folder. This requirement is a refinement of the c_-CurrentView data.
Rationale	The requirement ensures that the operator can save the current data for future reference and analysis, and provides a digital record of its operation.
Phase	III
Likely to Change	Yes. The method of which the operator saves the data, or which data is saved, may be changed in future revisions.
Associated Inputs and Outputs	m_Barometer, m_Acceleration, m_Gyroscope, m_gps, c_CurrentView, c_CurrentLoc, and c_PastLoc.

Table 55: USE_003

Description	The product shall be able to operate and provide data to the operator for $>$ 5 minutes without the need to recharge. This requirement is refines the functions stated within Section Product Functions and references MTNC_001.
Rationale	The requirement ensures that the product is able to explore the entire parking lot without the need to recharge in between.
Phase	II
Likely to Change	Yes. This requirement may be changed due to hardware limitations of the product, and is not required for the MVP.
Associated Inputs and Outputs	N/A.

Table 56: USE_004

Description	The product shall require less than 2 hours of training for the operator to use. This requirement is a refinement of the assumptions stated within Section Assumptions.
Rationale	The requirement ensures that no technical knowledge is required for the operator to use, allowing for widespread adoption of the product.
Phase	I
Likely to Change	No. Due to the user characteristics of the product, this requirement is required in order for use by the targeted user.
Associated Inputs and Outputs	N/A.

3.14 Portability

There are no requirements related to the portability of the product.

4 Supporting information

4.1 Appendixes

4.1.1 Appendix A: Reflection

Although the team has a solid education in the foundation in Mechatronics, they lack practical experience in building drones and robots in general. The knowledge the team currently does not have is identified in Domain Experts. Each discipline of knowledge was assigned to a specific team members who would become the expert in the subject. They may use any resources to gain the knowledge, such as books, blogs, YouTube videos, and other websites available to them.

The team has also defined what it means to be an expert in their domain:

- In depth understanding of how the component works physically (if applicable).
- Proper reasoning as to why specific component/firmware was chosen.
- In depth understanding of the key parameters and specifications of the product/firmware.
- In depth understanding of the inputs and outputs to the domain.
- In depth understanding of the integration of the domain into the project.

For many of the technical domains, the team has opted to partake in a dual-expert system where two experts are present for the domain. This ensures that in the absence of an expert, the effects on the team are minimized. Furthermore, this promotes collaboration between the experts during their research of their domain. Git, programming languages and Linux are already well understood by team members.

Table 57: Domain Experts

Domain	Assigned Experts	Description of Domain	Recommended Resources
Latex	Zaid	The Latex domain expert is responsible for understanding the Latex syntax and generation process.	Google
Power Management and Motors	Fady and Zaid	This domain relates to the powering of the individual components of the drone, and the control of the motors. Example components include the battery, Power Distribution Board, and the Electronic Speed Controller.	Motor datasheet, Google, Youtube
Mechanical Design	Winnie and Ali	Included within this domain is the creation of the custom drone frame, in addition to minimizing vibrations while maximizing structural integrity.	Google, Youtube
External Sensors and Peripherals	Fady, Winnie and Ali	This domain includes the external sensors that connect to the flight controller, including the camera, radio transmitter, and any other sensors that may be required.	Google, Youtube, Datasheet of components
Flight Controller	Zaid, Fady, Ali, and Winnie	All the members should be responsible for being experts at the Flight Controller as they all need to know how to interface with it. Further sub domains may be created at a later date if the need arises.	Official documentation of Flight Controller
Internal Sensors	Fady, Winnie and Ali	The Internal Sensors domain includes the sensors within the Flight Controller. Such sensors include the Inertial Measurement Unit, barometer, and GPS.	Official documentation of Flight Controller, Google, Youtube

4.1.2 Appendix B: Formal Transition Table

Table 58: Internal State Machine Variable: These variables are defined in order to help to simplify the Transition Table 59

Variable Name	Type	Unit	Description
k_ConnectionLost-Thresh	float	sec	Constant of 5 seconds. Used to define what it means for connection to be weak and/or lost.
k_DesiredLocCha-nged	bool	-	Indicates if the desired location has been changed by the user.
k_OnGround	bool	-	Indicates if the drone is currently on the ground.

Table 59: State Transition Matrix: This is a formal specification of the finite state machine. The current states are written along the top row, and the transitions along the leftmost column. Other cells indicate the next state.

-	Idle	Hover	Manual Location Move	Autonom ousExplo re	Confi gurati on	Off	Desired Loca- tion Out of Bounds Error	Malfuncti on	Land	No Park- ing Lot Error De- tected
m_PowerOn	-	-	-	-	-	Idle	-	-	-	-
!m_PowerOn	Off	-	-	-	-	-	-	-	-	-
m_Launch & i_Mode == Normal	Hover	-	-	-	-	-	-	-	-	-
m_Launch & i_Mode == Co nfigure	Confi gurati on	-	-	-	-	-	-	-	-	-
m_Autonomou- sExplore & c_ParkingLot- Detected	-	Auton omous Explo re	Autonom ousExplo re	Autonom ousExplo re	-	-	Auton omous Explo re	-	-	Auton omous Explo re
c_UserError == Desired_- Location_- Out_Of_- Bounds	-	-	Desired Location Out of Bounds	-	-	-	-	-	-	-
m_Land	-	Land	Land	Land	-	-	Land	Land	Land	Land
(c_Connected) held for k_Con- nectionLost- Thresh	Malfu nction	Malfu nction	Malfuncti on	Malfuncti on	Malfu nction	-	Malfu nction	Malfuncti on	Malfu nction	Malfu nction
c_ParkingLot- Detected	-	Autonomous Ex- plore	-	-	-	-	-	-	-	-
k_DesiredLoc- Changed	-	Manual Loca- tion Move	Manual Location Move	Manual Location Move	-	-	Manual Loca- tion Move	Manual Location Move	Manual Loca- tion Move	Manual Loca- tion Move
k_OnGround	-	-	-	-	-	-	-	Off	Off	-

4.2 Index

References

[1] T. Canada, "Flying your drone safely and legally," Sep. 2020, publisher: AARV. [Online]. Available: https://tc.canada.ca/en/aviation/drone-safety/learn-rules-you-fly-your-drone/flying-your-drone-safely-legally

[2] ——, "Find your category of drone operation," Feb. 2021, publisher: AARV. [Online]. Available: https://tc.canada.ca/en/aviation/drone-safety/learn-rules-you-fly-your-drone/find-your-category-drone-operation