LifeAutomation

March 26, 2021

Dr. Craig Scratchley
School of Engineering Science
Simon Fraser University
Burnaby, BC, V5A 1S6

Re: ENSC 450W/440 Design Specification for an Autonomous Waste Cart System

Dear Dr. Scratchley,

Please find the document attached below that outlines the design specifications for implementing our smart moving waste cart, Bin Buddy. The purpose of Bin Buddy is to eliminate the need of people having to manually bring out their own garbage. After the easy initial calibration of the boundaries by the homeowners, Bin Buddy will take care of moving the waste carts at scheduled days and times.

The document enclosed lists the design specifications for each individual subsystem. Design choices are detailed and considered to fulfill the requirements as outlined in the requirements document.

Four computer engineering students and one systems engineering student - Aigerim (Ayoka) Omirbekova, Emily Chen, Jeongwon (Julia) Kim, Kira Nishi-Beckingham and Sheetal Puri - make up our amazing LifeAutomation team. Strong with the computer engineering background, our team brings our software experience to the table with a blend of robotics to tinker our project into realization.

Thank you for your time in reviewing our design specifications. Should there be any questions regarding the document, please reach out to our Chief Operations Officer, Jeongwon (Julia) Kim, via email jka183@sfu.ca or phone at (604) 441-7044.

Sincerely,

Kira Nishi-Beckingham
Chief Executive Officer
LifeAutomation

Enclosed: Design Specification for an Autonomous Waste Cart System

Design Specification

Bin Buddy

Partners: Aigerim (Ayoka) Omirbekova
Emily Chen
Jeongwon (Julia) Kim
Kira Nishi-Beckingham
Sheetal Puri

Contact Person: Jeongwon (Julia) Kim
jka183@sfu.ca
(604) 441-7044

Submitted To: Dr. Craig Scratchely, ENSC 405W
Dr. Andrew Rawicz, ENSC 440
School of Engineering Science
Simon Fraser University

Issue Date: March 26, 2021

Abstract

This document outlines the design specifications of Life Automation's autonomous waste cart moving system, Bin Buddy. Design options and choices are explored and justified for each of the subsystems and subcomponents. For each of these subsystems, design requirements are carefully chosen to satisfy the requirements as described in the requirements document, exploring technical details with respect to the mechanical, hardware, and software components of the system. Supporting test plans are also included in the appendix to address testing the subcomponents and components.

Changelog

Table 0. 1: Changelog

Date	Version	Notes
2021-03-26	1.0.0	Published

Table of Contents

Abstract ii
Changelog iii
Glossary ix
1. Introduction 1
  1.1 Scope 1
  1.2 Intended Audience 2
  1.3 Design Classification 2
2. System Overview 3
3. Mechanical Design 4
  3.1 Body Structure 4
  3.2 Chassis Structure 6
  3.3 Braking Mechanism 7
4. Hardware Design 8
  4.1 Microcontroller Unit 8
  4.2 Sensors System 11
    4.2.1 Obstacle Detection 11
    4.2.2 Empty Waste Cart Detection 14
    4.2.3 Navigation 17
  4.3 Motor 19
  4.4 Motor Driver 23
  4.5 Power Supplies 24
    4.5.1 Chassis Power Supply 24
    4.5.2 MCU Power Supply 26
    4.5.3 Emergency power button for motor 27
5. Software Design 28
  5.1 Line Recognition 28
  5.2 Front-End: User Application 30
  5.3 Back-End 31
    5.3.1 System Communication 31
    5.3.2 Database 32
6. Conclusion 33
7. References 34

Appendix A: Test Plan 36
A.1 Introduction 36
A.2 Software Testing 36
A.3 Electronics Testing 38
A.4 Mechanical Testing 39
Appendix B: User Interface and Appearance Design 41
Appendix C: Supporting Design Options 42
C.1 Body and Chassis Structure 42
C.2 Braking Mechanism 44
C.3 MCU 45
C.4 Obstacle Detection 46
C.5 Empty Waste Cart Detection 47
C.6 Navigation 48
C.7 Motors 49
C.8 Motor driver 51
C.9 Chassis Motor Power Supply 52
C.10 Emergency Power On/Off button 53
C.11 Line Recognition 54
C.12 Front-End: User Application 55
C.13 Backend: System Communication 56
C.14 Backend: Database 57

List of Tables

Table 0. 1: Changelog iii

Table 1. 1: Product Phases 2

Table 3. 1: Body Structure Design Requirements 5
Table 3. 2: Chassis Structure Design Requirements 6
Table 3. 3: Braking Mechanism Design Requirements 7

Table 4. 1: Microcontroller Design Requirements 8
Table 4. 2: Raspberry Pi 3 Model B+ specifications 9
Table 4. 3: Ultrasonic Sensor Specifications 11
Table 4. 4: Ultrasonic Sensors Design Requirements 13
Table 4. 5: MPU6050 Specification 15
Table 4. 6: Inertial Sensor Design Requirements 16
Table 4. 7: Camera Design Requirements 18
Table 4. 8: Motor Design Requirements 22
Table 4. 9: Motor Driver Design Requirements 23
Table 4. 10: Chassis Power Supply Design Requirements 25
Table 4. 11: MCU Power Supply Requirements 26
Table 4. 12: Emergency Power Button Requirements 27

Table 5. 1: Line Recognition Design Requirements 29
Table 5. 2: Front End Design Requirements 30
Table 5. 3: Web Server Design Requirements 31
Table 5. 4: Database Design Requirements 32

Table C. 1: Requirements for the Body and Chassis Structure 42
Table C. 2: Design Options for Body Structure 42
Table C. 3: Design Options for Drive Assembly 43
Table C. 4: Requirements for the Braking Mechanism 44
Table C. 5: Design Options for the Braking Mechanism 44
Table C. 6: Requirements for the MCU 45
Table C. 7: Design Options for the MCU 45
Table C. 8: Requirements for Obstacle Detection 46
Table C. 9: Design Options for Obstacle Detection 46
Table C. 10: Requirements for the Empty Waste Cart Detection 47
Table C. 11: Design Options for the Empty Waste Cart Detection 47
Table C. 12: Requirements for the Navigation 48
Table C. 13: Design Options for the Navigation 48
Table C. 14: Requirements for the Motors 49
Table C. 15: Design Options for the Motors 50
Table C. 16: Requirements for the Motor Driver 51
Table C. 17: Design Options for the Motor driver 51
Table C. 18: Requirements for Chassis Power Supply 52

Table C. 19: Design Options for the Chassis Motor Power Supply 52
Table C. 20: Requirements for Chassis Power Supply 53
Table C. 21: Design Options for the Line Recognition 53
Table C. 22: Design Options for the Line Recognition 54
Table C. 23: Requirements for the Front-End User Application 55
Table C. 24: Design Options for the Front-End User Application 55
Table C. 25: Requirements for the System Communication 56
Table C. 26: Design Options for the System Communication... 56
Table C. 27: Requirements for the Database 57
Table C. 28: Design Options for the Database 57

List of Figures

Figure 1. 1: General Overview of the Bin Buddy system 1

Figure 2. 1: General System Diagram 3

Figure 3. 1: Structure of POC version of Bin Buddy 4
Figure 3. 2: Angled view [left] and side view [right] of Bin Buddy attached to a waste cart 5

Figure 4. 1: Raspberry Pi 3 Model B+ 8
Figure 4. 2: Raspberry Pi 3 Model B+ GPIO Layout 10
Figure 4. 3: Ultrasonic Sensor HC-SR04 11
Figure 4. 4: Ultrasonic Sensor Timing Diagram 12
Figure 4. 5: Ultrasonic Sensor Operation Diagram 12
Figure 4. 6: Orientation of Axes of Sensitivity and Polarity of Rotation 14
Figure 4. 7: MPU-6050 Block Diagram 15
Figure 4. 8: i2cdetect and I^2C Block Diagram 16
Figure 4. 9: Common Line Following Cases 17
Figure 4. 10: A One-wheeled System on an Incline of θ Degrees. 19
Figure 4. 11: Mathematical Symbols used in Torque and Power Calculations 19
Figure 4. 12: Applied force vs frictional force graph showing static and kinetic friction regions 20
Figure 4.13: Circuit diagram shows chassis motors connected in parallel with the power source24

Figure 5. 1: Line Recognition Demo 28
Figure 5. 2: System Communication Flow Diagram 31
Figure 5. 3: Database Schema 32

Glossary

Calibration: A process that involves a general product setup.

Large Obstacles: Cars, waste carts, bikes.

Small Obstacles: Branches and stones less than 3.5 cm in height.

Stuck: The state when the product is not able to move around a large obstacle, also, when the product is not standing straight (i.e. tipped over).

1. Introduction

Stand-alone or semi-detached houses are allocated waste collection schedules. To date, most cities adopt a rotating bi-weekly collection schedule for the different carts (organics, recyclables, and garbage) and have by-laws, imposing narrow time windows to place the carts out [1] [2] [3]. With these restrictions, bringing the waste cart to the front of the driveway becomes a cumbersome task for the typical working population and the elderly, resulting in missed collections due to mistiming, or wrong carts being placed on the curb.

LifeAutomation believes that Bin Buddy will be a solution to automating the waste collection service. Bin Buddy attaches to existing city waste carts, autonomously wheeling the cart to the front of the driveway and back once emptied after its initial boundary and app calibration. The smart waste cart system will detect large obstacles, such as cars on the driveway, while navigating to its docking location. A general overview of the Bin Buddy system is outlined in Figure 1.1.

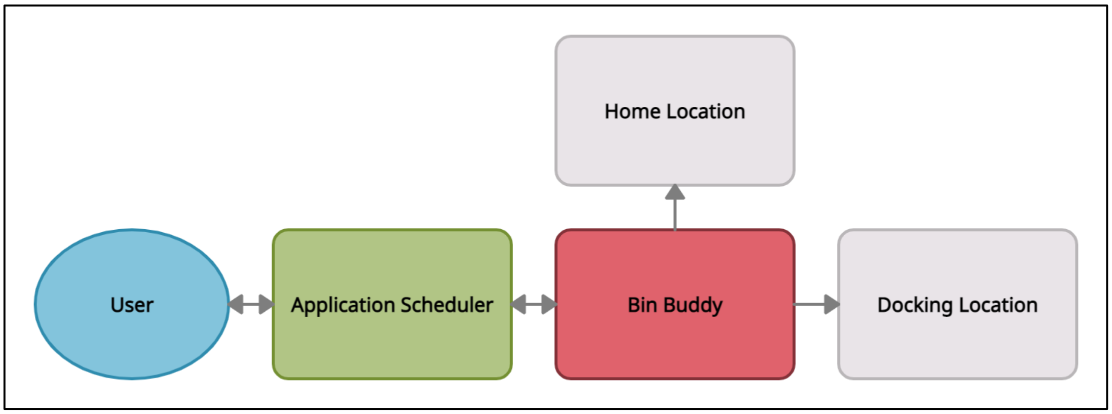

Figure 1. 1: General Overview of the Bin Buddy system

1.1 Scope

The purpose of this document is to specify the design requirements of the Bin Buddy device. Design choices, challenges, and alternatives are explored for each component, and justifications are provided for each subsystem and their components. For elaborated details on all design options, please refer to Appendix C.

1. Mechanical Design
   The mechanical design section will describe the design of the body and chassis structure.
2. Hardware Design
   The hardware design section will analyze all hardware components such as the microcontroller, navigation sensors, motors and driver motor, and the power supplies.
3. Software Design
   The software design section will detail the line recognition system as well as the front-end and back-end servers.

1.2 Intended Audience

This document will serve as LifeAutomation's design requirements guide for the LifeAutomation team, potential clients/partners, Dr. Craig Scratchley, Dr. Shervin Jannesar, Dr. Andrew Rawicz, Mike Hegedus, and Chris Hynes.

1.3 Design Classification

The three stages of the project are categorized in the following format:

Des {Section}.{Sub-Section}.{Sub-Des Requirement (optional)}-{Product Phase Acronym}

Each product phase has an associated acronym for better readability.

Table 1. 1: Product Phases

Product Phase	Acronym
Alpha Phase (Proof of Concept)	POC
Beta Phase (Engineering Prototype)	EP
Production Phase (Final Product)	FP

2. System Overview

Bin Buddy consists of two main components: the Bin Buddy attachment, and the user application. Figure 2.1 shows a system overview of the design. The Bin Buddy waste cart attachment consists of various sensors that work with the motor driver and motors for navigation. This system is paired with the mobile application and will communicate with the Node.js server hosted on the Raspberry Pi microcontroller via Wi-Fi. Additionally, there is a MongoDB database which will be hosted on the cloud through MongoDB Atlas. The database will be responsible for storing and retrieving all user, device, and schedule information input by the user.

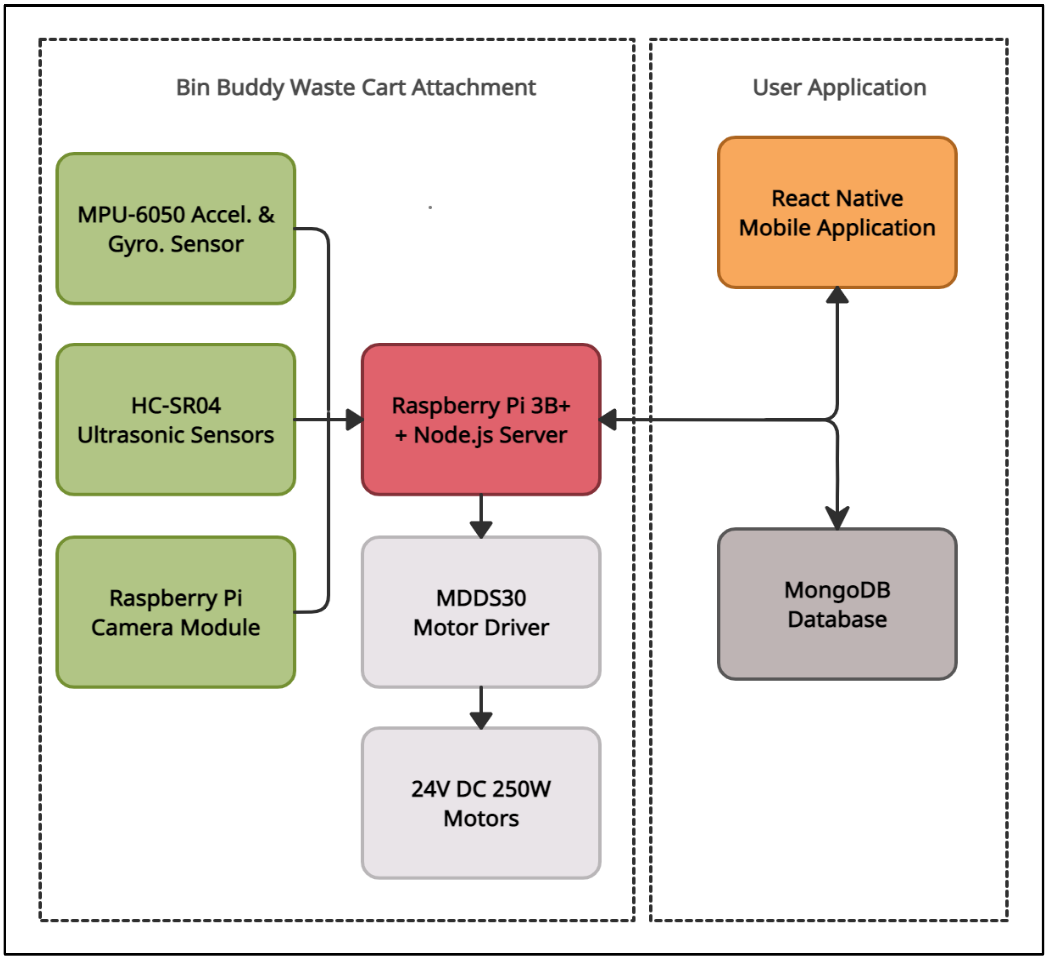

Figure 2. 1: General System Diagram

3. Mechanical Design

3.1 Body Structure

The POC version of Bin Buddy's body will be made from light weight but strong wooden beams and panels. The body structure of Bin Buddy is composed of a back wooden panel attached at 90 degrees to two supporting wooden beams as shown in Figure 3.1. Two stainless steel L brackets are added for support as shown in Figure 3.1. The back plate will slide behind the metal bar of waste cart and two U brackets will be secured onto the back plate to hold the device in place as shown in Figure 3.2 (left). This installation design will be made more user friendly for the EP version. The U-bracket will be placed at a height to ensure that Bin Buddy lifts the waste cart up from the front as shown in Figure 3.2 (right).

This wood and metal bracket design is chosen for the POC because it is lightweight, strong and highly customizable for the mechanical design testing phase.

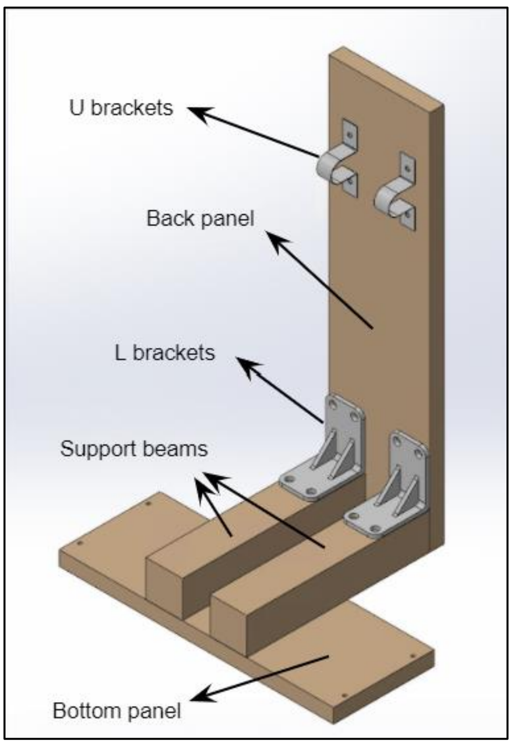

Figure 3. 1: Structure of POC version of Bin Buddy

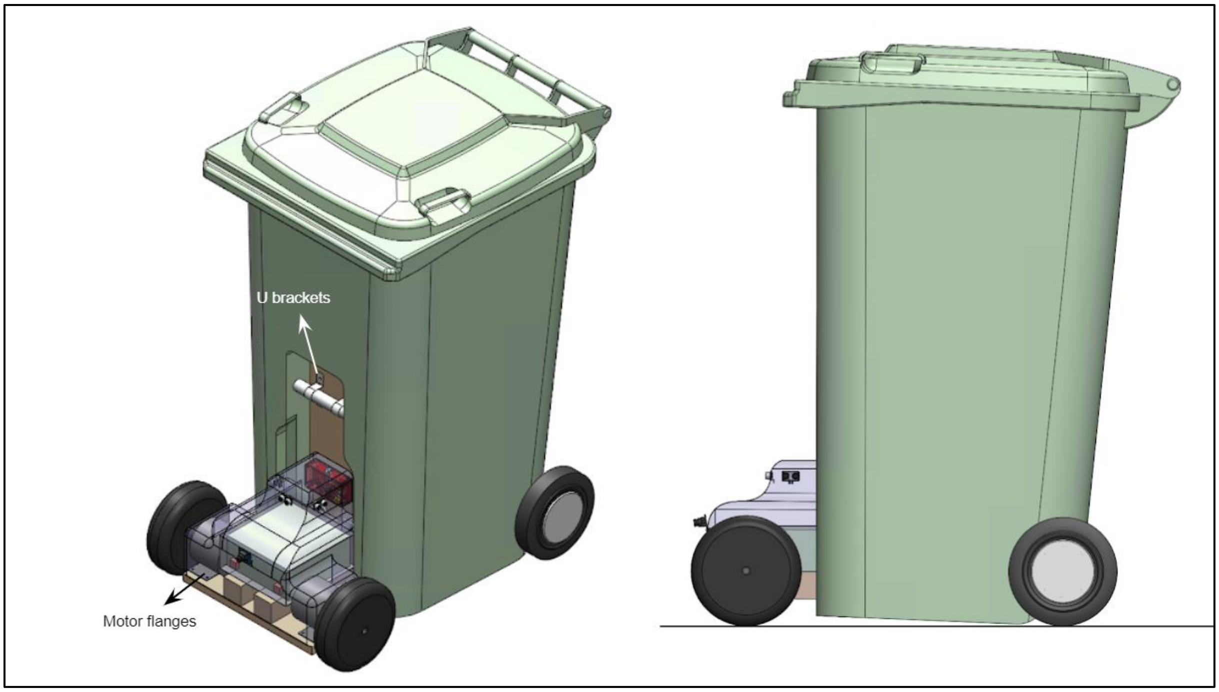

Figure 3. 2: Angled view [left] and side view [right] of Bin Buddy attached to a waste cart

Table 3. 1: Body Structure Design Requirements

Design ID	**Design Description**	**Corresponding Requirements**
Des 3.1.1 - POC	The back panel of body will have a height of 56 cm.	Req 4.1.1 - EP
Des 3.1.2 - POC	The length of the support beams will be 25.5 cm.	Req 4.1.2 - EP
Des 3.1.3 - POC	The U brackets will secure the back panel onto the bar tightly.	Req 4.1.6 - EP Req 8.1.4 - EP
Des 3.1.1 - EP	The electronics, motors and battery will be enclosed in a casing.	Req 8.1.1 - EP

3.2 Chassis Structure

The POC chassis structure includes a bottom plate as shown in Figure 3.1, two motors bolted onto the bottom plate using motor flanges as shown in Figure 3.2 (left) and two wheels attached to the motors. The wheels are rubber with a diameter of 20.32 cm.

To steer the device, a differential steering mechanism will be used where the speed of one DC motor is controlled relative to the other to change the direction of travel. Differential steering was chosen to decrease the number of mechanical parts needed for the assembly. This reduces size and weight of Bin Buddy.

Since differential steering is being used, each motor will be attached separately to each wheel. A flange coupler will be bolted to the disk of the wheel. The motor's shaft will then be coupled to the flange coupler using a set screw.

Table 3. 2: Chassis Structure Design Requirements

Design ID	**Design Description**	**Corresponding Requirements**
Des 3.2.1 - POC	The wheels will be 6.2 cm wide with a diameter of 20.32cm to maximize the device stability.	Req 3.1.2 - POC Req 3.1.3 - POC Req 8.1.5 - EP
Des 3.2.2 - POC	The bottom plate will be 38 cm long to increase the stability of the device-cart system.	Req 8.1.5 - EP
Des 3.2.3 - POC	Two DC motors will be attached directly to both wheels to use differential steering method to change Bin Buddy's direction.	Req 5.3.2 - POC Req 4.1.1 - FP
Des 3.2.4 - POC	The flange coupler will have a bore size of 1.2 cm to fit the motor shaft.	Des 3.2.3 - POC
Des 3.2.1 - EP	The wheels will be made from the rubber.	Req 4.1.4 - EP

3.3 Braking Mechanism

To avoid Bin Buddy from rolling away while parked, there must be a braking mechanism in place. To keep the braking system simple, an electromagnetic spring contact brake will be used to impose friction on one of the wheels.

Table 3. 3: Braking Mechanism Design Requirements

Design ID	Design Description	Corresponding Requirements
Des 3.3.1 - EP	The braking system will be a physical locking mechanism that prevents Bin Buddy from rolling away while not in motion.	Req 8.1.3 - EP
Des 3.3.2 - EP	The braking system will apply friction on one of the wheels using a spring contact and an electromagnet as control.	Req 8.1.3 - EP

4. Hardware Design

4.1 Microcontroller Unit

Bin Buddy must support web server communication, multiple sensors for different purposes, a camera, and a medium-sized motor system.

Raspberry Pi 3 Model B+ is a 64-bit quad-core microcomputer that allows that wide range of functionalities. Raspberry Pi is a powerful compact microcomputer that carries multiprogramming. Raspberry Pi 3 is the most suitable for the Bin Buddy system in power, energy consumption, and extra connectivity. The Model B+ was specifically chosen because it supports HATs (Hardware Attached on Top) which can be used for the drive system. The overall price for the MCU does not differ significantly, however, the performance with Raspberry Pi is multiple times greater than with other microcontrollers. Figure 4.1 and Table 4.1 show the layout and specifications of the MCU.

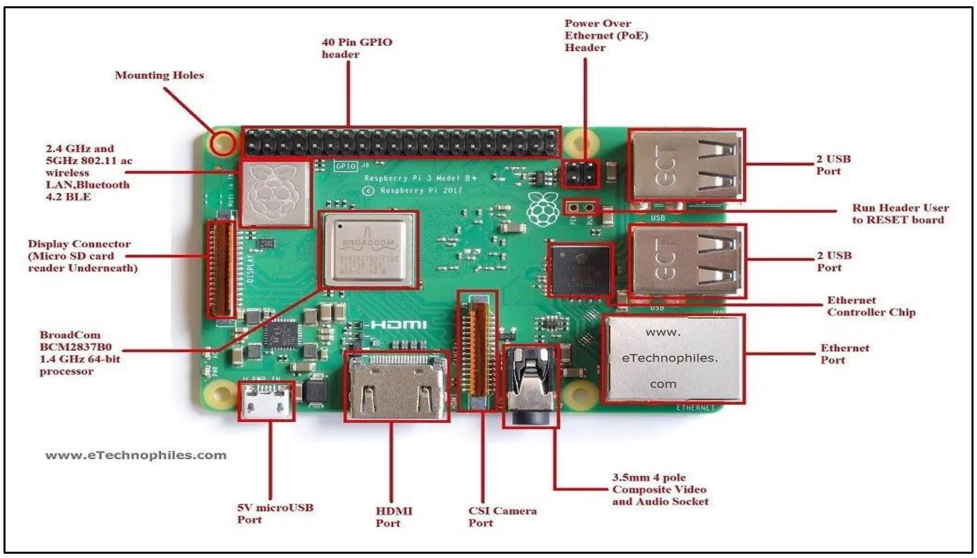

Figure 4. 1: Raspberry Pi 3 Model B+ [4]

Table 4. 1: Microcontroller Design Requirements

Design Requirement	Requirement Description	Corresponding Requirements
Des 4.1.1 - POC	The MCU will support 2.4GHz wireless LAN.	Req 5.1.1 - POC
Des 4.1.2 - POC	The MCU will carry a 2.5A-3A output.	Req 5.1.2 - POC Req 5.1.3 - POC
Des 4.1.3 - POC	The MCU will support camera streaming.	Req 5.1.3 - POC Req 5.1.5 - POC

Table 4. 2: Raspberry Pi 3 Model B+ specifications [5]

Specification type	Value
Dimensions	85mm x 56mm x 17mm
Weight	45g
Processor	Broadcom BCM2837B0, Cortex-A53 64-bit SoC @ 1.4GHz
Memory	1GB LPDDR2 SDRAM
Access	Extended 40-pin GPIO header
SD card support	Micro SD format for loading operating system and data storage
Connectivity	● 2.4GHz and 5GHz IEEE 802.11.b/g/n/ac wireless LAN, Bluetooth 4.2, BLE ● Gigabit Ethernet over USB 2.0 (maximum throughput 300Mbps) ● 4 × USB 2.0 ports
Video & sound	● MIPI CSI camera port
Input Power	● 5V/2.5A DC via micro-USB connector ● 5V DC via GPIO header
Environment	Operating temperature, 0–50°C

GPIO Specifications

Figure 4.2 shows the layout of Raspberry Pi pins. For an input pin, any voltage between 1.8V and 3.3V is read as “HIGH” and a voltage lower than 1.8V as “LOW” by the Raspberry Pi. No devices with an input voltage higher than 3.3V should be connected to the microcomputer, or it may result in the frying the pins. For the output pin, if the pin is set to “HIGH”, the voltage is 3.3V. If it is set to “LOW”, the voltage at the output is 0V.

Pulse-width modulation is available on all the pins on the software side and on GPIO12, GPIO13, GPIO18, GPIO19 for the hardware.

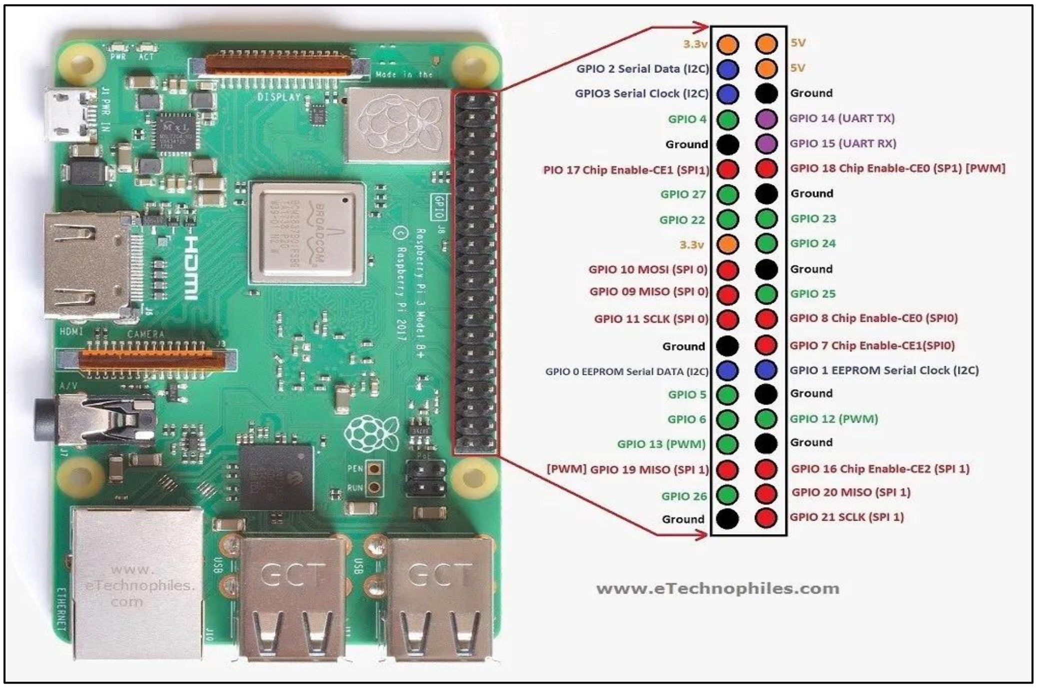

Figure 4. 2: Raspberry Pi 3 Model B+ GPIO Layout [6]

4.2 Sensors System

4.2.1 Obstacle Detection

For the obstacle detection with Raspberry Pi, ultrasonic sensors HC-SR04 are chosen as they can operate in an outdoor environment.

The Bin Buddy system will have one ultrasonic sensor placed at the front and one on each side except the rear of the waste cart to avoid driving into large obstacles. To maneuver around a large obstacle, Bin Buddy requires at least 90 cm to operate. Therefore, the ultrasonic sensors will be used to detect large obstacles from within 180 cm away.

Table 4.3 lists some of the sensor's specifications that are relevant to Bin Buddy.

Table 4. 3: Ultrasonic Sensor Specifications [7]

Parameter	Value
Working Voltage	DC 5 V
Working Current	15mA
Working Frequency	40Hz
Min, Max Range	2cm, 400cm
Practical Measuring Distance	2cm - 80cm
Measuring Angle	15°
Trigger Input Signal	10us TTL pulse
Dimensions	45mm x 20mm x 15mm

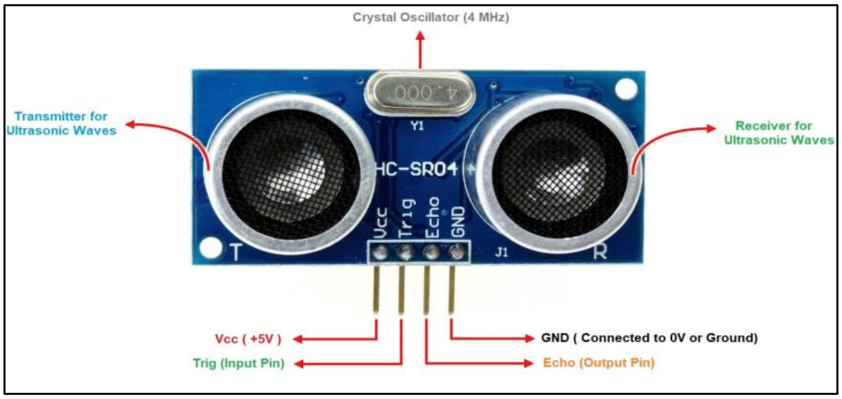

Figure 4. 3: Ultrasonic Sensor HC-SR04

For the trigger input to start the ranging, a short 10us high (5V) pulse must be supplied. This initiates the sensor to transmit 8 cycles of ultrasonic burst at 40kHz, and to wait for the reflected burst of ultrasound. In order to prevent a trigger signal to the echo signal, the measurement cycle must be over 60ms. After the sensor detects ultrasound from the receiver, it will set the Echo pin to a high voltage (5V) and delay for a period, proportional to the distance. The distance can be calculated at followed:

$$distance = \frac{time * sound\ velocity}{2}\text{, where the sound velocity} = 340\ \text{m/s}$$

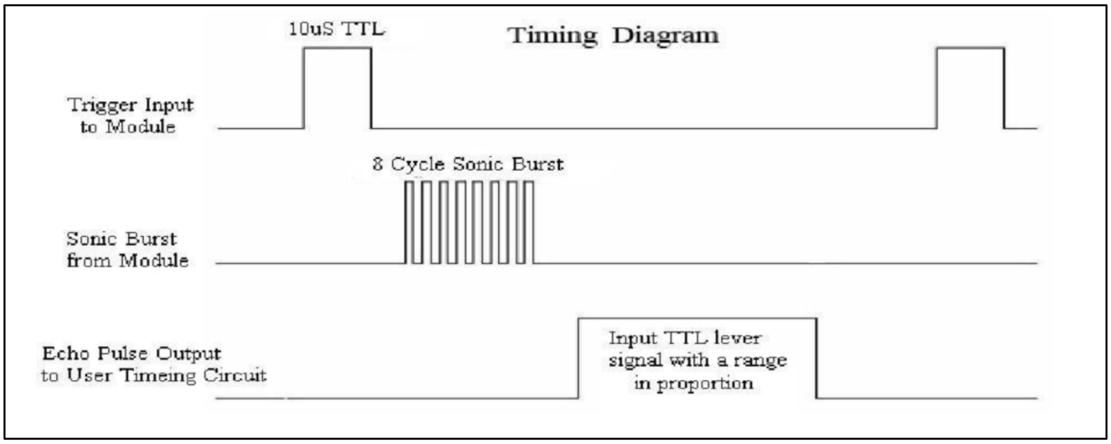

Figure 4. 4: Ultrasonic Sensor Timing Diagram [7]

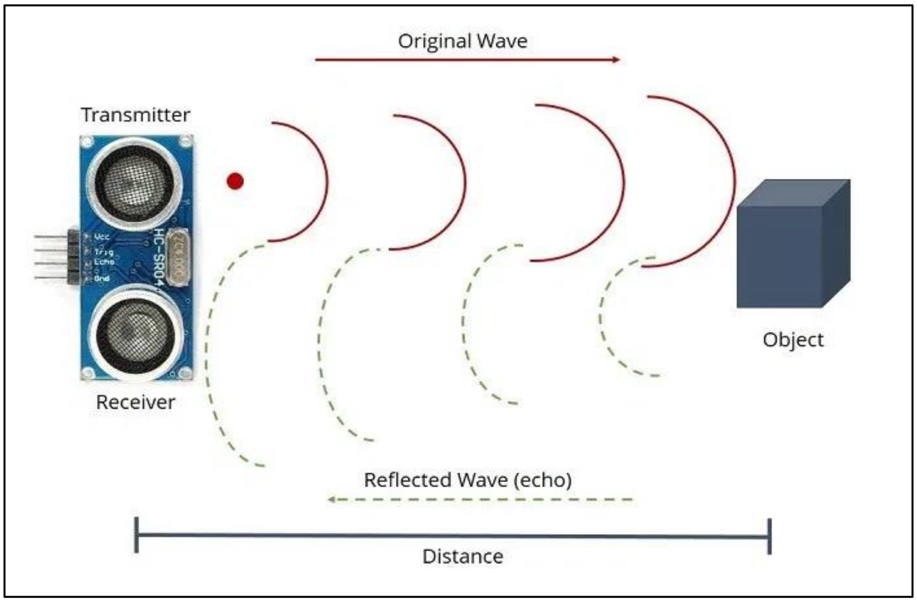

Figure 4. 5: Ultrasonic Sensor Operation Diagram [8]

Table 4.4 lists the design requirements for each phase of the project.

Table 4. 4: Ultrasonic Sensors Design Requirements

Design ID	Requirement Description	Corresponding Requirements
Des 4.2.1.1 - POC	The HC-SR04 sensor will detect large obstacles within 180 cm, which falls within its non-contact measurement range of 2-400cm.	Req 5.5.1- POC Req 5.5.1 - EP
Des 4.2.1.2 - POC	The ranging accuracy of the HC-SR04 sensor will not exceed 3 mm.	Req 5.5.1 - POC Req 5.5.1 - EP
Des 4.2.1.1 - EP	The Bin Buddy system will have one ultrasonic sensor placed at the front and one on each side except the rear of the waste cart to allow maintaining at least 90 cm of distance between Bin Buddy's body structure and large obstacles.	Req 3.2.2 - EP Req 5.5.1 - EP Req 6.2.5 - EP

4.2.2 Empty Waste Cart Detection

The MPU-6050 sensor is used to detect when the waste cart is emptied to signal Bin Buddy to return home. It will also detect when Bin Buddy is tipped over to alert the user. This sensor is chosen as it has both the accelerometer and the gyroscope.

Based on feedback from the instructional team, the accelerometer and the gyroscope sensor is sufficient for detecting the tilt. Changes that are as drastic as the angle change of 180 degrees to Bin Buddy itself would rarely occur unless a waste cart is being emptied out by the waste collection truck. If Bin Buddy topples over due to an unfortunate event instead of being emptied out, the MPU-6050 sensor will be used to detect that Bin Buddy is unable to travel due to being flipped over and the detected condition will be used to trigger a function call to alert the user.

The MPU-6050 sensor defines the x, y and z axes as shown in Figure 4.6.

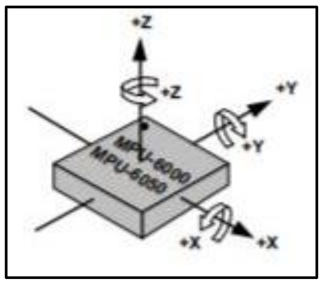

Figure 4. 6: Orientation of Axes of Sensitivity and Polarity of Rotation [9]

As shown in Figure 4.7, the MPU-6050 sensor has three independent axes each for gyroscope and accelerometer where the gyroscope detects rotation to provide angular velocity and the accelerometer measures displacement on axes. On a flat surface, x and y axes measure 0g and 1g on z-axis [9].

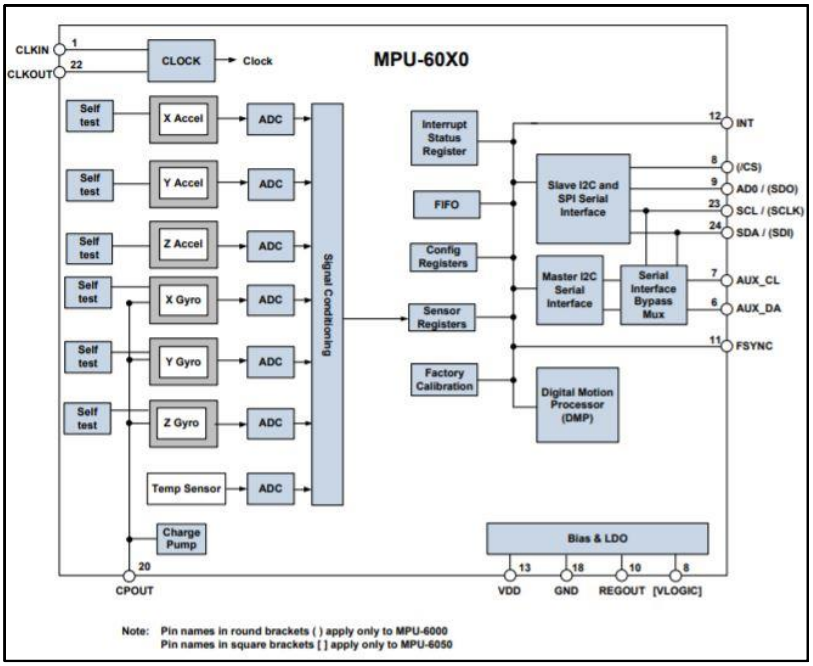

Figure 4. 7: MPU-6050 Block Diagram

Table 4.5 lists some of the important parameters of MPU-6050.

Table 4. 5: MPU6050 Specification

	Specification	Gyroscope	Accelerometer
 Figure #: MPU 6050	Measurement Ranges	$\pm 250°/sec$, $\pm 500°/sec$, ±1000°/sec, ±2000°/sec	$\pm 2g$, $\pm 4g$, $\pm 8g$, $\pm 16g$
	Operating Current	3.8mA (gyroscope + accelerometer only, DMP disabled)	
	Temperature Range	-40 to 85 ℃	
	VDD	2.375V-3.46V (absolute maximum rating: -0.5V to 6V)	

Pins that are of interest to us are Pin 13 (VDD), Pin 18 (GND), Pin 23 (SCL), Pin 24 (SDA). VDD and GND are power supply rails. SCL and SDA are I^2C serial clock and data, respectively. The MPU6050 sensor will use the Inter-Integrated Circuit (I^2C) communication protocol, which allows for the ease of identifying the address of the sensor as shown in Figure 4.8 [10].

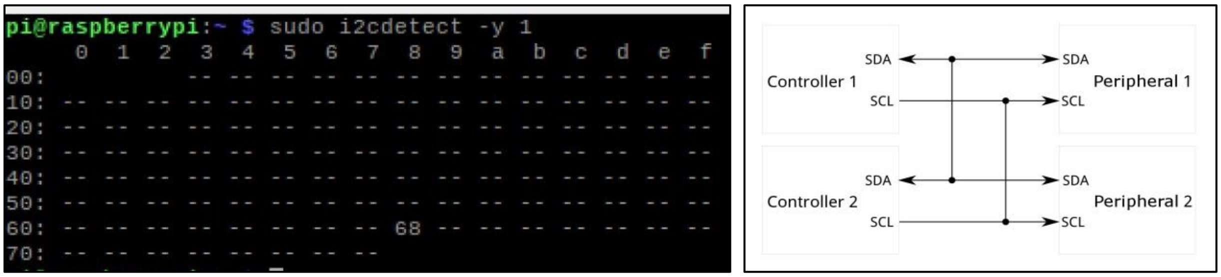

Figure 4. 8: i2cdetect and I^{2}C Block Diagram

The I^{2}C protocol is useful for peripheral communication, requiring only two wires while supporting multiple peripheral devices as shown in Figure 4.8 [10]. Controllers 1 and 2 in the block diagram above would be Raspberry Pi B+ and the peripherals would be the MPU6050 sensor for Bin Buddy.

Table 4.6 contains the design requirements for the inertial sensor.

Table 4. 6: Inertial Sensor Design Requirements

Design ID	Design Description	Corresponding Requirements
Des 4.2.2.1 - POC	The MPU-6050 sensor will detect a rotational change between 40 -180 ^{o} with respect to the base frame.	Req 5.4.1 - POC Req 6.2.6 - EP
Des 4.2.2.1 - EP	The MPU-6050 sensor will distinguish between a rotational change of 40 - 90 ^{o} and 90 -180 ^{o} to determine if Bin Buddy fell over or was emptied out. This condition will initiate the method to either alert the user or return to the home base.	Req 3.1.2 - POC Req 3.2.4 - EP Req 3.2.5 - EP Req 5.4.1 - EP Req 6.2.6 - EP

4.2.3 Navigation

The Raspberry Pi Camera Module will be used to implement line following using the Hallway Problem concept. The Hallway Problem significantly narrows down the scope of the navigation area by constraining Bin Buddy's field of navigation to a set of virtual or physical walls. This allows Bin Buddy to have a path to always follow, eliminating the complexities of path mapping and pinpointing the system's precise location. Line following can be used to exercise this problem using a Raspberry Pi camera module. As the line strays away from the camera's frame of view, Bin Buddy will readjust so the line returns to the center of the frame.

Figure 4.9 shows the common cases that will be addressed, some of which will be incorporated in conjunction with the ultrasonic sensors. The dashed lines are the virtual boundaries while the solid line is the line that will be set up by the user.

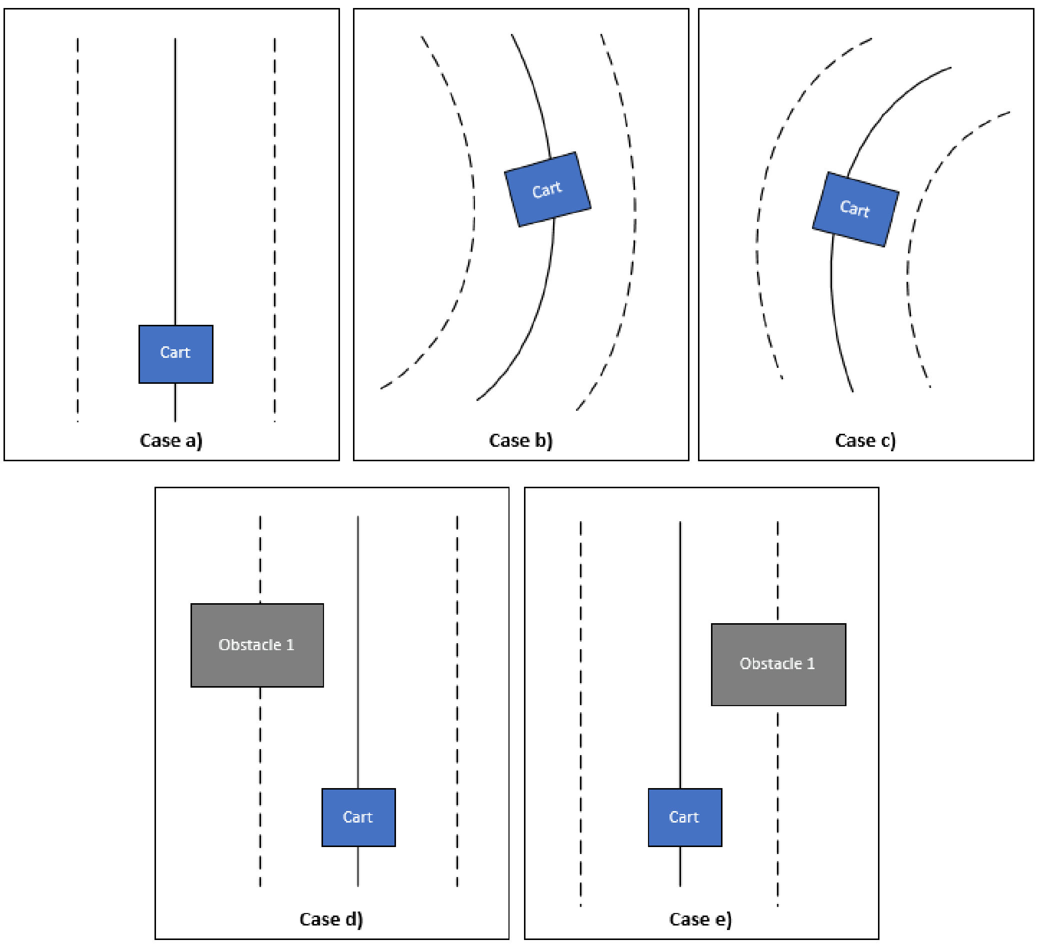

Figure 4. 9: Common Line Following Cases

Case a) is the base case containing a straight line. Bin Buddy will make no changes to its driving direction and can continue driving forwards. Cases b) and c) contain curved lines in either direction. Bin Buddy will traverse these curves based on the instructions of the image recognition software. Cases d) and e) describe the presence of obstacles at both sides. Bin Buddy will attempt to avoid these obstacles while remaining within the scope of the virtual walls. In all other cases, assistance will be required from the user as Bin Buddy will either be located beyond the scope of the virtual walls, or an obstacle is completely blocking the width of the possible path of travel.

The camera module is also used to recognize the docking stations, which is the beginning and the end of the navigation process of Bin Buddy. It will detect a simple, distinct colour at each docking station.

Table 4. 7: Camera Design Requirements

Design ID	**Design Description**	**Corresponding Requirements**
Des 4.2.3.1 - POC	The camera will support a resolution of 1080p.	Req 3.1.1 - POC Req 3.1.1.1 - POC Req 3.2.1 - EP Req 3.2.5 - EP
Des 4.2.3.2 - POC	The camera will support a frame rate of 30 fps.	Req 3.1.1 - POC Req 3.1.1.1 - POC Req 3.2.1 - EP Req 3.2.5 - EP
Des 4.2.3.3 - POC	The camera will be connected to the Raspberry 3B+ microcontroller.	Req 3.2.1 - EP Req 3.2.1.1 - EP
Des 4.2.3.4 - EP	The camera will detect the docking station by recognizing the docking station colours.	Req 3.2.1 - EP Req 3.2.1.1 - EP

4.3 Motor

Two geared PMDC (Permanent Magnet DC) motors of rated power 250W, rated current of 14A and voltage of 24V are used to control the mobility of Bin Buddy. These motors meet the torque and power requirements from calculations shown in sections below. PMDC motors are selected because they offer high power for low cost and offer bi-directional speed control using a simple controller with H-bridge and PWM support.

4.3.1 Motor torque and power calculations

Figure 4.10 is a sketch of a one-wheeled system on an incline. Figure 4.11 shows the symbols used in the calculations.

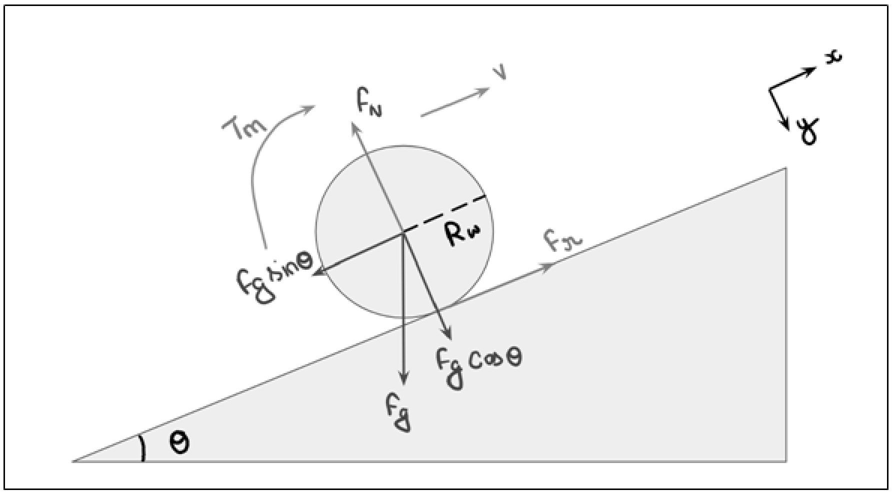

Figure 4. 10: A One-wheeled System on an Incline of θ Degrees.

Radius of the wheel	R_W
Mass of the wheel	m_w
Angular acceleration of the wheel	α
Linear acceleration of the wheel	a_x
Moment of Inertia of the wheel	I
Frictional force between rubber wheel and concrete ground	F_r
Static friction constant	μ_s
Normal reaction force	F_N
Weight of the system	F_g
Mass of the load (includes device's and waste cart's mass)	m_l
Acceleration due to gravity	$g = 9.8m/s$
Angle of incline	θ
Torque required from motor	T_m
Net torque on the system	$\sum \tau$

Figure 4. 11: Mathematical Symbols used in Torque and Power Calculations

All calculations are the worst-case scenarios for torque required from the motor.

For example, more torque will be required by the motor system while the device is climbing up an incline of 10°. The frictional force for impending motion – in other words, when the device is just starting up and is about to move – has the maximum value as shown in Figure 4.12. Therefore, the maximum frictional force (static friction) will be used to calculate the worst-case scenario torque.

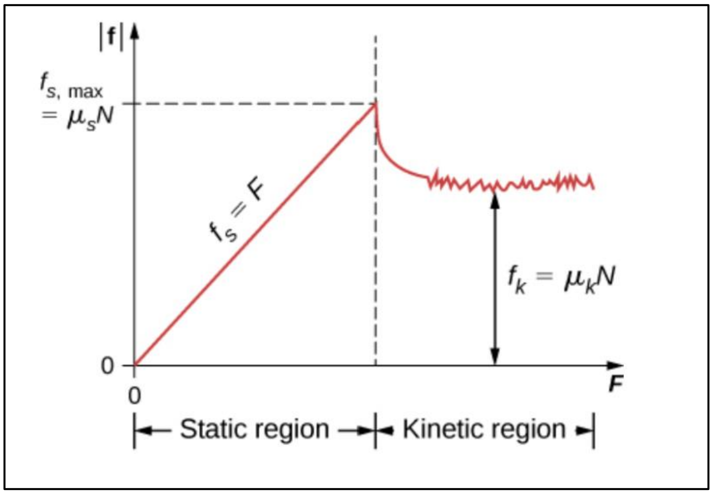

Figure 4. 12: Applied force vs frictional force graph showing static and kinetic friction regions [11]

Given the assumption that the wheel is a solid cylinder,

$$I = \frac{1}{2}(m_w)(R_W)^2$$

and the angular acceleration is

$$\alpha = \frac{a}{R_W}$$

the net torque equation with the clockwise direction set as the positive sense of rotation is:

$$\circlearrowright \sum \tau = T_m - R_W F_r = I\alpha = \frac{1}{2}(m_w)(R_W)^2 \frac{a}{R_W}$$

$$\Rightarrow T_m = R_W F_r + \frac{1}{2}(m_w)(R_W)(a)$$

Assuming the wheel is impending motion, meaning F_r considers only the static friction value,

$$F_r = \mu_s F_N = \mu_s F_g cos\theta = \mu_s (m_l g) cos\theta$$

This is the torque required by the motor to overcome friction between the wheel and ground, and to overcome the wheel's moment of inertia:

$$\Rightarrow T_m = (R_W)(\mu_s)(m_l)(g)\cos\theta + \frac{1}{2}(m_w)(R_W)(a)$$

$$\boldsymbol{T_m = (R_W)[(\mu_s)(m_l)(g)\cos\theta + \frac{1}{2}(m_w)(a)]}$$

From the requirements document, the following parameters and their values are derived.

$R_w = 0.10m$ [Req 3.1.3 $- POC$]
$m_w = 1.45kg$
$\theta = 10°$ [Req 3.2.4 $- EP$]
$Load\ mass\ m_l = 50kg - 1.45kg = 48.6kg$ [Req 5.3.2 $- FP$]
$\mu_s = 1$ [12]
$Max\ Linear\ Velocity = 10\ km/hr = 2.8m/s$ [Req 6.1.3 $- POC$]

$Operating\ linear\ velocity\ on\ a\ 10°\ incline = 0.5m/s$

($based\ on\ the\ average\ speed\ of\ a\ Roomba\ which\ is\ 0.3m/s$ [13]).

At a velocity of 0.5m/s, Bin Buddy will take approximately 40 seconds one way to travel a driveway length of 20m, meeting requirement [Req 5.2.2 - EP].

From this velocity value, the value of acceleration is chosen to ensure that Bin Buddy has sufficient time to accelerate from 0m/s to 0.5 m/s without inducing sudden jerks. The time to accelerate is chosen to be 2 seconds as an upper boundary.

From these approximations, the linear acceleration can be calculated:

$$a = \frac{v - 0}{t} = \frac{0.5 - 0}{2} = 0.25m/s^2$$

These values are used to calculate the torque required.

$$\Rightarrow T_m = (R_W)[(\mu_s)(m_l)(g)\cos\theta + \frac{1}{2}(m_w)(a)]$$

$$\Rightarrow T_m = (0.1)[(1)(48.6)(9.8)\cos(10°) + \frac{1}{2}(1.45)(0.25)]$$

$$\Rightarrow T_m = (0.1)[469.044 + 0.1813]$$

$$\Rightarrow T_m = 46.9\ Nm$$

A service/safety factor of 1.5 will be accounted for in the required torque [14].

$\Rightarrow T_m = 70.35\ Nm$ for the whole chassis system

The load will be distributed evenly on two motors.

Therefore, the rated torque required from each motor is: $\boldsymbol{T_{m1} = 35.2\ Nm}$

Using the rated torque, the power requirements for the motors can also be derived.

Angular velocity, $\omega = \frac{v}{R_w} = \frac{0.5}{0.1} = 5\ rad/s$

RPM = $\frac{(\omega)(60)}{2\pi} = \frac{(5)(60)}{2\pi} \approx 48\ RPM$

Power required for each motor, $P_m = (T_{m1})(RPM)(Conversion\ Factor)$ [15]

$\boldsymbol{P_m = (35.2)(48)(0.1047) \approx 177\ Watt}$

Table 4.8 outlines the motor requirements for Bin Buddy based on the torque and power calculation.

Table 4. 8: Motor Design Requirements

Design ID	Design Description	Corresponding Requirements
Des 4.3.1 - POC	The motor will be bi-directional.	Req 5.3.1 - POC Req 5.3.2 - POC
Des 4.3.2 - POC	The motor will be able to maintain a speed of 0.5m/s or 48RPM.	Req 5.3.1 - POC
Des 4.3.3 - POC	The motor will have rated torque of at least 35.2Nm.	Req 5.3.1 - EP Req 5.3.2 - FP
Des 4.3.4 - POC	The motor will have rated power of at least 177W.	Req 5.3.1 - EP Req 5.3.2 - FP
Des 4.3.5 - POC	The motor will be reasonably quiet.	Req 5.3.2 - EP
Des 4.3.6 - POC	The motor will last a reasonable lifespan of product.	Req 5.3.1 - FP

4.4 Motor Driver

The motor driver used in Bin Buddy will be the MDDS30 SmartDriveDuo Smart Dual Channel 30A Motor Driver by Cytron Technologies. This will be referred to as MDDS30 driver from hereon. This driver was chosen because it offers all POC and EP requirements while being an inexpensive option.

Table 4. 9: Motor Driver Design Requirements

Design ID	Design Description	Corresponding Requirements
Des 4.4.1 - POC	The MDDS30 driver will offer bi-directional control.	Req 5.3.1 - POC
Des 4.4.2 - POC	The MDDS30 driver will offer PWM speed control for each motor.	Req 5.3.2 - POC
Des 4.4.3 - POC	The MDDS30 driver will allow current draw of at least 14A per motor.	Des 4.3.3 – POC Des 4.3.4 - POC Req 5.3.1 - EP Req 5.3.2 - FP
Des 4.4.4 - POC	The MDDS30 driver must have thermal and current limit protection.	Req 8.1.4 - FP
Des 4.4.5 - POC	The MDDS30 driver will be reasonably quiet.	Req 5.3.2 - EP

4.5 Power Supplies

4.5.1 Chassis Power Supply

The power of the motor, P_m = 177 Watt, is calculated in section 4.3. Therefore, the power required for two motors is 354W, which is then rounded to 360W.

In section 4.3, a device moving at a velocity of 0.5m/s device is approximated to take 40 seconds to cross a driveway length of 20m. After accounting for extra distance covered due to maneuvering around obstacles, the roundtrip time will be approximated to have an upper bound of 5 minutes.

Goal: Power a 360W system continuously for 5 minutes.

Power of system in Watts, P = 360W
Time system needs to run for continuously in minutes, T = 5min
Efficiency of battery, e = 98% = 0.98

As shown in Figure 4.13, the left and right motors in chassis are connected in parallel with the power supply using the motor controller. The voltage required from the battery needs to be the rated voltage of each motor. 24V lead-acid batteries are not readily available, so two 12V batteries are used in series.

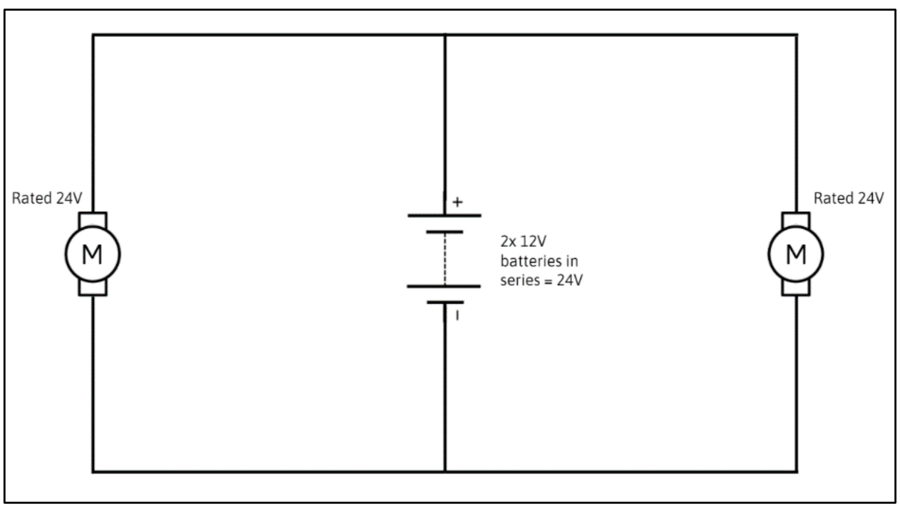

Figure 4.13: Circuit diagram shows chassis motors connected in parallel with the power source

Given

Voltage of battery in Volts, V_b = 24V
Energy stored in battery in watt-hours = E_b
Capacity of battery in amp-hours = C_b

$$\Rightarrow E_b = \frac{(P_m)(\frac{T}{60}hours)}{e} \quad \Rightarrow \frac{(360W)(\frac{5}{60}hours)}{0.98} = \frac{30}{0.98} \approx 31\,watt - hours$$

$$\Rightarrow C_b = \frac{E_b}{V_b} \Rightarrow \frac{31Wh}{24V} \approx 1.3\,amp - hours$$

Therefore, the chassis requires a 1.3AH capacity and a 24V power supply to run continuously for a 5-minute roundtrip.

To keep the battery weight and cost to a minimum, two rechargeable 12V SLA (Sealed Lead Acid) batteries connected in series with a capacity of 4.5Ah will be used. Thus, the device will be able to make approximately 3 roundtrips of 5 minutes each before requiring a recharge.

Table 4. 10: Chassis Power Supply Design Requirements

Design ID	Design Description	Corresponding Requirements
Des 4.5.1.1 - POC	The power supply will supply a voltage of 24V.	Req 5.2.1 - POC
Des 4.5.1.2 - POC	The power supply will provide at least 1.3amps per hour.	Req 5.2.1 - EP Req 5.2.2 - EP
Des 4.5.1.3 - POC	The power supply will last 3 roundtrip cycles before requiring a recharge.	Req 5.2.1 - EP
Des 4.5.1.4 - POC	The power supply will be rechargeable.	Req 5.2.1 - FP
Des 4.5.1.1 - FP	The power supply will weigh no more than 3kg each to minimize the product weight.	Req 4.1.1 - FP

4.5.2 MCU Power Supply

Typically, the Raspberry Pi 3 B+ requires a power of 2.5Ah with 5V [16]. However, since the camera module is present in the project, the Raspberry Pi requires a portable battery of ~3Ah that will output 5V. Due to the implementation of a new board design but keeping some old components, the Raspberry Pi 3 has a ~0.6V voltage drop from the USB port to the chip. Therefore, an adapter that will provide 5.2 - 5.4V is ideal.

Table 4. 11: MCU Power Supply Requirements

Design ID	Design Description	Corresponding Requirements
Des 4.5.2.1 - POC	The power supply will supply a voltage of at least 5V.	Req 5.1.2 - POC
Des 4.5.2.2 - POC	The power supply will provide at least 2.5Ah.	Req 5.1.2 - POC Req 5.1.3 - POC
Des 4.5.2.3 - POC	The power supply will have a micro-USB power connector.	Req 5.1.5 - POC
Des 4.5.2.1 - FP	The power supply will be rechargeable.	Req 5.2.1 - FP

MCU Power Consumption Calculations

According to the Raspberry Pi documentation, the typical bare-board active current consumption for Raspberry Pi 3 B+ is 500mA and ~260mA on average [16].

Therefore, the power consumption by the microcomputer is:

$$P\ (consumed\ by\ Raspberry\ Pi)\ =\ V*I\ =\ 500mA*5.0V\ =\ 2.5W$$
$$P\ (consumed\ by\ Raspberry\ Pi, average)\ =\ V*I\ \simeq\ 260mA*5.0V\ \simeq\ 1.3W$$

The portable battery will output ~5V with 2.5Ah. Therefore, the power * hours provided by the battery before recharged is:

$$P*hours\ (provided\ by\ the\ battery)\ =\ V*I*h\ =\ 5V*2.5Ah\ =\ 12.5Wh$$

As a result, the battery life before recharging in hours can be estimated as:

$$Battery\ lifetime\ =\ \frac{P*hours(provided\ by\ the\ battery)}{P(consumed\ by\ Raspberry\ Pi)} = \frac{12.5Wh}{2.5W}\ \simeq 5\ hours$$

$$Battery\ lifetime\ (average)\ =\ \frac{P * hours(provided\ by\ the\ battery)}{P(consumed\ by\ Raspberry\ Pi, average)} = \frac{12.5Wh}{1.3W} \simeq 9.6\ hours$$

4.5.3 Emergency power button for motor

Since Bin Buddy operates on a 24V battery with a high motor current draw of 28A, a battery isolation system is needed as a safety feature. A battery kill switch will be used to cut off the input terminal connection from battery.

Table 4. 12: Emergency Power Button Requirements

Design ID	Design Description	Corresponding Requirements
Des 4.5.3.1 - POC	An emergency power button will mechanically rotate out to safely disconnect the chassis motors from the power supply.	Req 8.1.1 - POC
Des 4.5.3.1 - EP	A small push-button control will be used to energize the relay coil to be used as an emergency button to isolate the chassis system from the 24V SLA battery.	Req 8.1.1 - POC

5. Software Design

5.1 Line Recognition

Using the camera module mentioned in the section above, Bin Buddy will utilize the following algorithms and Python libraries from OpenCV on Raspberry Pi B+ to recognize a line as needed for the navigation of Bin Buddy [17].

Gaussian Blur is one of the pre-processing techniques used in smoothing images for the purposes of noise reduction through diminishing details [18]. Applying a Gaussian Blur is effectively applying a low-pass filter, removing high frequency components of an image. To use the GaussianBlur function from OpenCV, the width and height parameters of the Gaussian Kernel must be specified [19].

Hough Transform is another important technique used in feature extraction of an image where every edge point is transformed into a line over iterations and the regions with the highest density of lines crossing are considered as true lines [20]. To use the Hough Line Transform function in OpenCV, the function needs to intake the minimum length of line and the maximum amount of gap between the lines as its parameters [21]. In our case, we are adopting HoughLinesP() as the probabilistic Hough Transform is less computationally exhaustive and therefore, faster [22].

Colour Space Conversion is also an important component in image processing and needed to translate the colour space from one basis to another. The cvtColor function is used.

Figure 5.1 demonstrates what these techniques will result.

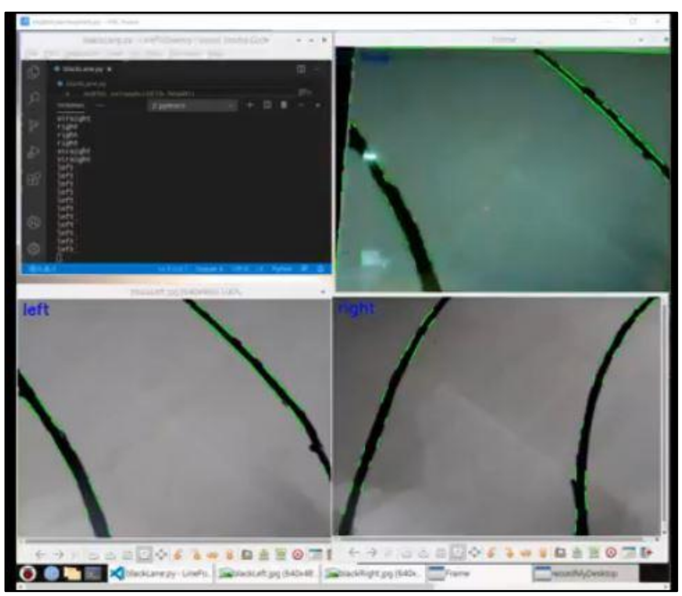

Figure 5. 1: Line Recognition Demo

Table 5. 1: Line Recognition Design Requirements

Design ID	**Design Description**	**Corresponding Requirements**
Des 5.1.1 - POC	The image recognition software will use Python to utilize OpenCV functions, such as cvtColor, HoughLinesP and Gaussian Blur, to process the line as images in a frame rate up to 17 frames per second.	Req 3.1.1 - POC Req 3.1.1.1 - POC Req 6.1.1 - POC
Des 5.1.2 - POC	The image recognition software will detect a light-colored line and update driving directions to Bin Buddy every second.	Req 3.1.1 - POC Req 3.1.1.1 - POC Req 6.1.1 - POC
Des 5.1.3 - POC	The image recognition software will throw a flag when the line is not in the field of view of the camera module after 20 seconds to adjust its travel. The user will be alerted after 60 seconds.	Req 3.2.5 - EP Req 5.6.1 - POC Req 6.1.1 - POC Req 6.2.4.1 - EP
Des 5.1.1 - EP	The image recognition software will try to relocate the line by turning around 0 - 180 degrees to remain on its course of path and direction.	Req 3.1.1.1 - POC Req 5.6.1 - POC Req 6.1.1 - POC

5.2 Front-End: User Application

The chosen user application for the Bin Buddy system is a mobile app developed using the React Native framework. This framework is open-source and cross platform. Using JavaScript, React Native ensures a fast and seamless mobile development cycle. React Native is purely front-end and will communicate with the backend server that will be located on the Raspberry Pi microcontroller.

The application will consist of four core functionalities.

1. Registering as a new user or logging as an existing user.
2. Adding and viewing registered devices and their statuses as well as accessing the emergency stop button to halt all moving devices.
3. Viewing the calendar for pre-scheduled waste cart devices.
4. Using the mobile drawer navigation menu to navigate between pages.

Table 5. 2: Front End Design Requirements

Design ID	Design Description	Corresponding Requirements
Des 5.2.1 - POC	The application will be available for Android devices running version 9.0 and iOS devices running iOS 12 or later.	Req 6.3.1 - FP
Des 5.2.2 - POC	The application will send requests to the Raspberry Pi through REST API calls over Wi-Fi.	Req 3.2.3 - EP
Des 5.2.3 - POC	The application will send alerts to the user if the Bin Buddy attachment is unable to navigate to its destination location autonomously, or an unexpected interrupt is triggered.	Req 6.2.4.1 - EP Req 6.2.6 - EP Req 6.3.2 - FP
Des 5.2.1 - EP	The application will send requests to the database to store or retrieve schedules.	Req 3.2.3 - EP
Des 5.2.2 - EP	The application will have a maximum of 6 intuitive screens to increase productivity.	Req 3.2.6 - EP Req 3.2.7 - EP Req 3.2.7.1 - EP Req 6.2.2 - EP
Des 5.2.3 - EP	The application will display the location of each device.	Req 6.2.3 - EP

5.3 Back-End

5.3.1 System Communication

A web server will be used for system communication. A node web server and mobile application will be hosted on the raspberry pi. This will ensure that any mobile device with internet access will be able to pull information from the Bin Buddy device through the web server. The Bin Buddy will be able to send alerts and information to the mobile application through get/post requests and the mobile application will be able to send collection data to a database connected to the web server. The user will have complete control over the system if they have Wi-Fi access regardless of their proximity to the system making the control extremely accessible.

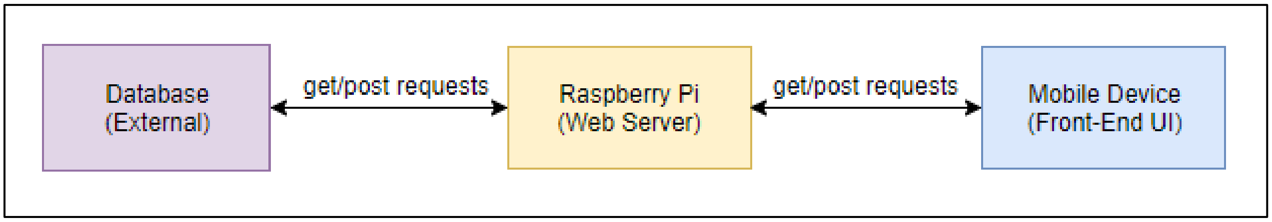

Figure 5. 2: System Communication Flow Diagram

Table 5. 3: Web Server Design Requirements

Design ID	**Design Description**	**Corresponding Requirements**
Des 5.3.1.3 - POC	The system will use a web server to communicate with the mobile application using get/post requests.	Req 3.2.3- EP Req 3.2.5 - EP Req 6.2.4.1 - EP Req 6.2.6 - EP Req 6.3.2 - FP Des 5.2.2 - POC Des 5.2.3 - POC Des 5.2.3 - EP
Des 5.3.1.2 - POC	The web server will be hosted on a Raspberry Pi.	Req 5.1.1 - POC
Des 5.3.1.1 - EP	The web server will connect to a database to store user data.	Req 3.2.3- EP Des 5.2.1 - EP

5.3.2 Database

The data will be stored using MongoDB and MongoDB Atlas. MongoDB Atlas is a cloud database service which is free to use and would allow Bin Buddy access to a database server without the overhead of manually creating one. MongoDB Atlas uses MongoDB which is a noSQL database. This is optimal as the data that needs to be stored for Bin Buddy is minimal and fairly unstructured. The MongoDB database would consist of two collections as seen in Figure 5.3.

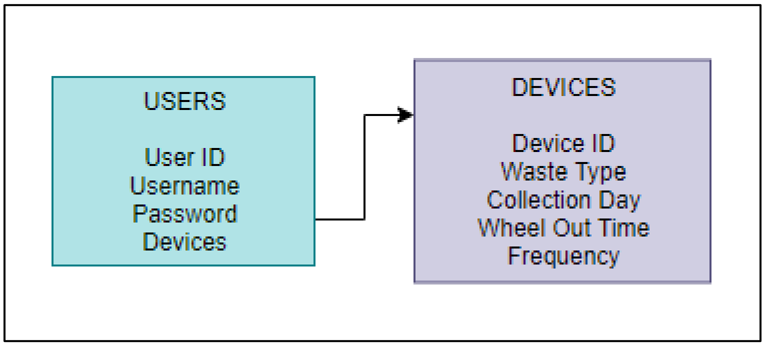

Figure 5. 3: Database Schema

Table 5. 4: Database Design Requirements

Design ID	**Design Description**	**Corresponding Requirements**
Des 5.3.2.1 - EP	The database will use MongoDB and mongoose.	Req 3.2.3- EP Des 5.2.1 - EP
Des 5.3.2.2 - EP	The database will be hosted on MongoDB Atlas.	Req 3.2.3- EP Des 5.2.1 - EP

6. Conclusion

This design document outlines design choices for Bin Buddy, with a focus on the POC. The details of how each design choice is implemented and the design decisions are justified through considering design alternatives. Below is a summary of each key section of the document:

1. Mechanical Design
    a. Body structure
        - The POC version of Bin Buddy's body structure will be made from wooden beams and panels with two U brackets bolted onto the back plate.
    b. Chassis structure
        - Differential steering mechanism will be used with two wheels. The wheels and motors will be attached using a flange coupler.
    c. Braking mechanism
        - A simple spring and contact mechanism will be used to apply pressure on one of the two wheels while Bin Buddy is not in motion.
2. Hardware Design
    a. Microcontroller Unit
        - The Raspberry Pi 3 B+ is chosen as the microcomputer.
    b. Sensors System
        i. Obstacle Detection will be executed using Ultrasonic Sensors HC-SR04.
        ii. Empty Waste Cart Detection will use a MPU-6050 Inertial Sensor.
        iii. Navigation will use the Raspberry Pi Camera Module.
    c. Motor and relevant calculations
        - Two motors with rated power of 250W each are chosen to meet the minimum torque requirement calculated.
    d. Power Supplies
        - A 24V SLA battery with a minimum 3.9AH is required to power the chassis system for 3 roundtrips without a recharge. 5V with 2.5Ah is required to power the Raspberry Pi 3 B+.
        - A battery disconnect switch for the chassis power will be a safety feature.
3. Software Design
    a. Line Recognition will use Python and OpenCV with the Camera Module
    b. User Interface – mobile application
        i. Frontend
            o The React Native framework will be used for cross-platform compatibility.
        ii. Backend
            o A Node web server will be used for system communication.
            o MongoDB will be used as Bin Buddy's database system.

This technical document is a living document that serves as the implementation guidelines for Bin Buddy.

7. References

[1] "Collection Calendar & Guidelines," [Online]. Available: https://www.coquitlam.ca/157/Collection-Calendar-Guidelines. [Accessed 10 February 2021].

[2] "City of Surrey," [Online]. Available: https://www.surrey.ca/services-payments/waste-collection/collection-schedules. [Accessed 10 February 2021].

[3] "Look up your collection schedules," [Online]. Available: https://vancouver.ca/home-property-development/garbage-and-recycling-collection-schedules.aspx. [Accessed February 2021].

[4] "Labists," [Online]. Available: https://labists.com/products/labists-raspberry-pi-3b-32gb. [Accessed 02 March 2021].

[5] "raspberrypi," [Online]. Available: https://static.raspberrypi.org/files/product-briefs/Raspberry-Pi-Model-Bplus-Product-Brief.pdf. [Accessed 05 March 2021].

[6] "eTechnophiles," [Online]. Available: https://www.etechnophiles.com/raspberry-pi-3-b-pinout-with-gpio-functions-schematic-and-specs-in-detail/. [Accessed 05 March 2021].

[7] "sparkfun," [Online]. Available: https://cdn.sparkfun.com/datasheets/Sensors/Proximity/HCSR04.pdf. [Accessed 05 March 2021].

[8] R. Santos and S. Santos, "Random Nerd Tutorials," [Online]. Available: https://randomnerdtutorials.com/complete-guide-for-ultrasonic-sensor-hc-sr04/. [Accessed 05 March 2021].

[9] "InvenSense," [Online]. Available: https://invensense.tdk.com/wp-content/uploads/2015/02/MPU-6000-Datasheet1.pdf. [Accessed 03 March 2021].

[10] "sparkfun," [Online]. Available: https://learn.sparkfun.com/tutorials/i2c. [Accessed 03 March 2021].

[11] "ER Services," [Online]. Available: https://courses.lumenlearning.com/suny-osuniversityphysics/chapter/6-2-friction/ . [Accessed 24 March 2021].

[12] R. A. Serway and R. J. Beichner, Physics for Scientists and Engineering with Modern Physics, New York: Saunders.

[13] "Robot Reviews," [Online]. Available: http://www.robotreviews.com/chat/viewtopic.php?t=9166#:~:text=The%20Roomba%205xx%20actually%20has,last%20few%20inches%20when%20docking.. [Accessed 24 March 2021].

[14] Z. Khan, "Motor Control Tips," [Online]. Available: https://www.motioncontroltips.com/faq-choose-safety-factor-motor-design-lasts/#:~:text=Most%20documentation%20and%20motor%20selection,to%20output%20enough%20energy%20for.. [Accessed 25 March 2021].

[15] "Michigan Tech," [Online]. Available: https://pages.mtu.edu/~wjendres/ProductRealization1Course/DC_Motor_Calculations.pdf. [Accessed 23 March 2021].

[16] "raspberrypi," [Online]. Available: https://www.raspberrypi.org/documentation/hardware/raspberrypi/power/README.md. [Accessed 08 March 2021].

[17] Abhinav, "hackster.io," [Online]. Available: https://www.hackster.io/Abhinav_Abhi/ road-lane-detection-with-raspberry-pi-a4711f. [Accessed 23 Marchh 2021].

[18] "Wikipedia," [Online]. Available: https://en.wikipedia.org/wiki/Gaussian_blur. [Accessed 23 March 2021].

[19] M. Brett, "matthew-brett.github.io," [Online]. Available: https://matthew-brett.github.io/teaching/smoothing_intro.html#:~:text=The%20FWHM%20is%20the%20width,Y%20axis)%20is%20the%20FWHM.. [Accessed 23 March 2021].

[20] "caltech.edu," [Online]. Available: http://web.ipac.caltech.edu/staff/fmasci/home/astro_refs/HoughTrans_lines_09.pdf. [Accessed 23 March 2021].

[21] "OpenCV," [Online]. Available: https://opencv-python-tutroals.readthedocs.io/en/latest/py_tutorials/py_imgproc/py_houghlines/py_houghlines.html. [Accessed 23 March 2021].

[22] R. Singh, "RSDharra," [Online]. Available: https://rsdharra.com/blog/lesson/14.html#:~:text=First%20of%20all%2C%20let's%20detect,hence%20P%20in%20the%20name. [Accessed 23 March 2021].

[23] D. Collins, "Motion Control Tips," [Online]. Available: https://www.motioncontroltips.com/when-are-worm-gears-self-locking-and-where-is-this-useful/. [Accessed 12 March 2021].

[24] "Robotics Tomorrow," [Online]. Available: https://www.roboticstomorrow.com/article/2015/11/low-cost-lidar-based-navigation -for-mobile-robotics/7270. [Accessed 23 March 2021].

[25] "Robot Shop," [Online]. Available: https://www.robotshop.com/ca/en/lidar-lite-3-laser-rangefinder.html. [Accessed 23 March 2021].

[26] C. Pao, "Ceva," [Online]. Available: https://www.ceva-dsp.com/ourblog/how-are-visual-slam-and-lidar-used-in-robotic-navigation/. [Accessed 23 March 2021].

[27] "Locatify," [Online]. Available: https://locatify.com/blog/ble-beacons-no-bull-beacon-review/. [Accessed 23 March 2021].

[28] "Robot Platform," [Online]. Available: http://www.robotplatform.com/knowledge/actuators/dc_motors.html. [Accessed 17 March 2021].

[29] "Oriental Motor," [Online]. Available: https://www.orientalmotor.com/brushless- dc-motors-gear-motors/technology/AC-brushless-brushed-motors.html#:~:text=Motor%20Construction,steel%20ring%20surrounding%20the%20rotor.. [Accessed 12 March 2021].

[30] "EngineersGarage," [Online]. Available: https://www.engineersgarage.com/tech-articles/choosing-motor-for-robots/. [Accessed 18 March 2021].

[31] "MongoDB," [Online]. Available: https://www.mongodb.com/cloud/atlas. [Accessed 15 March 2021].

Appendix A: Test Plan

A.1 Introduction

Test Purpose
The corresponding test plan provides detailed steps for the Proof of Concept prototype testing to demonstrate that all design specifications and requirements are met.

Test Coverage
Typically, test coverage provides testing instructions for all the lifecycles of the product. However, for the purpose of ENSC 405W, this appendix presents a test plan solely for the Proof of Concept. The software and hardware/electronics components and subsystems will be tested followed by the mechanical part of the project. Upon verification, integration testing will be performed to demonstrate the functionality of Bin Buddy as one system.

Test Methods
The test equipment required to verify Bin Buddy's functionality is an empty 120L waste cart. This waste cart will be filled to capacity for testing in the EP phase. Test methods will rely on visual inspection.

Test Responsibilities
All verification will be performed by the members of LifeAutomation using the test plan steps.

The test plans (software, electronics, mechanical) for each subsystem are listed below.

A.2 Software Testing

Test Name: System Communication - Go	**Date:**
Test Description: Press "Start Bin Buddy" on the mobile application.	
Expected outcome: Bin Buddy device will begin moving within 1 second and begin following the provided line.	
Actual Outcome:	

Test Name: System Communication - Stop	**Date:**
Test Description: Press "Stop Bin Buddy" on the mobile application.	
Expected outcome: Bin Buddy device will stop moving within 1 second.	
Actual Outcome:	

Test Name: Line Following	**Date:**
Test Description: Press "Start Bin Buddy" on the mobile application, ensuring there are no obstacles obstructing the expected path.	
Expected outcome: The Bin Buddy device will begin moving within 1 second and continue to follow the line provided.	
Actual Outcome:	

Test Name: Obstacle Detection	**Date:**
Test Description: Place a large obstacle that covers the line path and virtual boundaries, then press "Start Bin Buddy" on the mobile application to start the device.	
Expected outcome: The device will send a notification to the mobile application, alerting the user that it is stuck.	
Actual Outcome:	

Test Name: Empty Cart Detection	**Date:**
Test Description: Lift the Bin Buddy device in the air and tip it over ~180 degrees, then place it back on the ground.	
Expected outcome: The Bin Buddy device will send a notification to the mobile app saying that the device has been emptied.	
Actual Outcome:	

A.3 Electronics Testing

Test Name: MCU Power Up	**Date:**
Test Description: Let the system run and consume power until the MCU stops responding (power supply dies). Recharge the MCU power supply.	
Expected outcome: The MCU will begin responding again.	
Actual Outcome:	

Test Name: Emergency Stop	**Date:**
Test Description: Power up the device and press “Start Bin Buddy” on the mobile application, then press the emergency stop button.	
Expected outcome: The Bin Buddy device will power off and stop all motion.	
Actual Outcome:	

A.4 Mechanical Testing

Test Name: Load Test	**Date:**
Test Description: Place a 45kg load in the waste cart and press "Start Bin Buddy" on the mobile application.	
Expected outcome: The device will begin to move, following the line provided.	
Actual Outcome:	

Test Name: Stability - Basic	**Date:**
Test Description: Set Bin Buddy at the top of a driveway at an angle of 10 degrees, then press "Start Bin Buddy" on the mobile application.	
Expected outcome: The device will follow the expected path down the driveway without tipping over.	
Actual Outcome:	

Test Name: Stability - Obstacles	**Date:**
Test Description: Place multiple small obstacles along the expected path, then press "Start Bin Buddy" on the mobile application.	
Expected outcome: The device should easily roll over small obstacles without tipping over.	
Actual Outcome:	

Test Name: Clamp	**Date:**
Test Description: Lift the waste cart with the Bin Buddy device attached to it in the air and tip it over ~180 degrees while shaking it.	
Expected outcome: The device should remain firmly attached to the trash bin.	
Actual Outcome:	

Test Name: Differential steering	**Date:**
Test Description: Set the expected path so that the device is expected to turn, then press “Start Bin Buddy” on the mobile application.	
Expected outcome: The device should be able to turn, following the expected path, without tipping over.	
Actual Outcome:	

Appendix B: User Interface and Appearance Design

This page is intentionally left blank as Appendix B was submitted separately.

Appendix C: Supporting Design Options

C.1 Body and Chassis Structure

Table C.1 lists the requirements for the body and chassis structure. Based on these requirements, Table C.2 lists the design alternatives for the body structure and Table C.3 lists the design alternatives for the chassis/drive assembly (non-exhaustive).

Table C. 1: Requirements for the Body and Chassis Structure

Requirement ID	Requirement Description
4.1.1 - EP	The product's height should be less than 53 cm.
4.1.2 - EP	The product's depth should be less than 38 cm to leave enough clearance between the waste pick-up truck's arm and the product.
4.1.6 - EP	The product must be secured tightly onto the city waste cart.
8.1.4 - EP	The product must be securely attached to the waste cart such that it does not fall off while the waste is getting collected.
8.1.1 - EP	The product's circuitry must be enclosed in a case.
3.1.2 - POC	The product must travel back and forth without tipping over.
3.1.3 - POC	The product shall drive over small obstacles.
8.1.5 - EP	The product design must be stable as to not tip and fall when going over small branches, rocks, and inclines of less than 10°
4.1.4 - EP	The wheels must have enough traction to navigate with minimal slippage
5.3.2 - POC	The chassis must be able to steer right or left to change directions.

Table C. 2: Design Options for Body Structure

Design Options	Specification
1. Wood and metal brackets	This design is light weight but will not be able to handle the device load requirements overtime. The strength of this design is improved by using metal brackets at critical stress points. This design is highly customizable and does not require any expertise to cut and drill into wooden beams and panels.
2. Stainless Steel Strut/C channels	This design utilizes readily available strut channels in various sizes, materials. These channels come with multiple holes for bolt and nut attachments. They offer high strength for long term use and a

	customizable design but require some expertise to cut and drill into. They also require complex mechanical design and many small brackets and attachment parts to be able to attach different pieces together.
3. CNC machined aluminum frame	This design is the most customizable since it offers the ease of designing in SolidWorks and sending the CAD to get machined and/or sheet-metal bent. Using this type of structure, we can machine various shapes and sizes, and optimize weight and area of the product. This design also gives the option of an in-built electronics enclosure. Although this option offers the most flexibility it is also more expensive than the first two options.

Design option 1 was selected as the POC design because it satisfies all the requirements for POC while being the most flexible and inexpensive option. For the EP either design option 2 or 3 will be chosen to increase the strength of Bin Buddy. Option 3 is preferred because of the ability to optimize the weight and cost and include an in-built enclosure.

Table C. 3: Design Options for Drive Assembly

Design Options	**Specification**
1. Sprocket and chain design	This design utilizes the sprocket that comes attached with the DC motor shaft. In this design, another shaft and a sprocket will be attached to the wheel and a chain will be installed on both motor and wheel sprockets with proper tension. This design offers an advantage where the chassis' output speed and torque can be changed by adjusting the sprocket ratios. The disadvantage of this design is that the wheel and shaft are not concentric. The motor shaft will have to be installed at an offset on the bottom plate to use the sprockets and chain. This increases the breadth of bottom plate.
2. Flange coupler design	In this design, a coupler with a flange will be bolted onto the wheel disk. This coupler will then receive the motor shaft which will be set in place using a set screw on the coupler. The disadvantage of this design is the ability to find a flange coupler with the right size bore. The wheel disk, if metal, needs to have tapped holes, or if plastic, needs to be drilled. The advantage of this design is the smaller number of parts required. The wheel and the motor shaft are concentric and thus require a minimal breadth of bottom wooden plate.

Design option 2, the flange coupler design, was selected as the final POC design because it satisfies all the requirements for the POC while providing a solid coupling of the wheel and motor shaft without using extra number of parts.

C.2 Braking Mechanism

Table C.4 lists the requirements for the braking system. Based on these requirements, Table C.5 lists the design alternatives (non-exhaustive).

Table C. 4: Requirements for the Braking Mechanism

Requirement ID	Requirement Description
Req 8.1.3 - EP	The product must have a braking system that locks the wheels in place to prevent rolling away while not in motion.

Table C. 5: Design Options for the Braking Mechanism

Design Options	Specification
1. Worm-gear	The byproduct of the worm-gear is friction and can be used to brake [23].
2. Disc brake	This disc brake is less dependent on the road conditions. The brake will be triggered either by a cable or hydraulically.
3. Rim brake	The rim brake is a type of brake used predominantly in low-mid priced bikes due to its cheap and simple mechanism; however, it will behave poorly when the rims are wet.
4. Caster brake	The caster brake can be commonly found on shopping carts, but it may increase our steering angle. The latch would be lowered and raised when Bin Buddy reaches its docking stations either hydraulically or using a cable instead of requiring manual labor. This also may not be the most suitable for reducing the speed while Bin Buddy is in motion.
5. Wheel stopper	The wheel stopper is not a sophisticated braking system but it may suffice for the purpose of parking Bin Buddy without affecting the performance of Bin Buddy while in motion. In this case, the wheel stopper will be placed at the docking stations rather than on the wheels directly; however, this would hinder Bin Buddy's ability to move without needing human intervention as it would need assistance to un-stop itself. This also cannot be used to help reduce the travelling speed of Bin Buddy.
6. Frictional spring contact brake with electromagnetic switch	This design will have a normally open relay type mechanism where when the coil is not energized, a physical contact will stick out and apply friction on the rim of the wheel. When the coil is energized, the physical contact will pop in and remove the brake on the device.

To keep the braking system simple, design option 6, an electromagnetic spring contact will be used to impose friction on one of the wheels, eliminating additional complexities to the motors.

C.3 MCU

Table C.6 lists requirements relevant to the microcontroller/microcomputer unit. Based on these requirements, Table C.7 lists the possible design options.

Table C. 6: Requirements for the MCU

Requirement ID	Requirement Description
5.1.1 - POC	MCU must have Wi-Fi capabilities
5.1.2 - POC	MCU must have a 5V/2.5A DC unit.
5.1.3 - POC	MCU must have up to 3A capacity.
5.1.4 - POC	The GPIO (General Purpose Input/Output) header must consist of 40 pins.
5.1.5 - POC	MCU must have at least one USB 2.0 port for powering up and uploading
5.1.6 - POC	MCU should have a micro-SD port to support loading an operating system and storing data.

Table C. 7: Design Options for the MCU

Design Options	Specification
1. Arduino	This microcontroller option is relatively simple and satisfies most of the requirements listed above. However, it does not have built-in modules like the Raspberry Pi, requiring additional units to be purchased and optimized to the system. Additionally, since Bin Buddy's system requires real-time updates, the timing becomes critical. The chips are slower, and the microcontroller does not support multiple programs at a time.
2. ESP32	This design option is considered as the most simple and cost-efficient. However, it would be suitable for the Bin Buddy system if a camera was not intended to be used. Since this is not a microcomputer, additional modules must be purchased as well.
3. Raspberry Pi 3B+	This microcomputer satisfies all the requirements and has options of adding HAT along with having read to be used built-in modules.
4.Raspberry Pi Zero W	This microcomputer does not have an option of adding HAT (Hardware Attached on Top). Therefore, working the medium sized motors can create complications.

Among these options, option 3, the Raspberry Pi 3 B+ was selected as it satisfies all the requirements for Bin Buddy at a higher efficiency and lower cost than the other options.

C.4 Obstacle Detection

Table C.8 lists requirements relevant to the microcontroller/microcomputer unit. Based on these requirements, Table C.9 lists the possible design options.

Table C. 8: Requirements for Obstacle Detection

Requirement ID	Requirement Description
5.5.1 - POC	The sensor must be able to detect large obstacles, given that the obstacles are within 180 cm.
5.5.2 - POC	The sensor must directly connect to the microcontroller.
5.5.3 - POC	The sensor shall have a current consumption of less than 2 mA.
5.5.4 - POC	The sensor shall be connected to a power supply of around 5V DC.
5.5.1 - EP	The sensor must detect large obstacles on each side of the waste cart for at least three sides.

Table C. 9: Design Options for Obstacle Detection

Design Options	Specification
1. Infrared sensors	This design option has certain limitations, especially for outdoor applications, such as Bin Buddy. They are less costly; however, this results in compromising the accuracy, which is not reliable for the small obstacles detection required for the product.
2. Ultrasonic sensors	Ultrasonic sensors provide the accuracy for obstacle detection in outdoor settings and meets all the requirements.

The price difference between infrared and ultrasonic sensors does not significantly affect the budget, while the performance differs considerably. As a result, HC-SR04 ultrasonic sensors are picked for the system.

C.5 Empty Waste Cart Detection

Table C.10 lists the requirements for the empty waste cart detection and Table C.11 lists the design options that were considered.

Table C. 10: Requirements for the Empty Waste Cart Detection

Requirement ID	Requirement Description
Req 3.1.2 - POC	The product must travel back and forth without tipping over.
Req 3.2.4 - EP	The product shall be able to safely travel up and down a driveway slope of maximum 10^o without toppling over.
Req 3.2.5 - EP	The product shall be able to recognize when it is stuck and alert the user.
Req 5.4.1 - EP	The inertial sensor must be able to distinguish between a rotational change of 40-90 ^{o} and 90-180 ^{o} to determine if Bin Buddy fell over or being emptied out.
Req 6.2.6 - EP	The software must alert the users in case of needing attention/user action - theft, stuck somewhere.

Table C. 11: Design Options for the Empty Waste Cart Detection

Design Options	Specification
1. Weight sensor	A weight sensor will detect the weight difference once the waste cart has been emptied. Relying on the weight parameter makes it difficult to tell if Bin Buddy has been put down on the ground after being emptied out or if it is still midair, but now emptied out. Changes that are as drastic as the angle change of 180 degrees to Bin Buddy itself would rarely occur unless a waste cart is being emptied out by the waste collection truck.
2. Magnetometer	Introducing a magnetometer to detect orientation is also an option, but after the discussion from the second progress review meeting and research, we have determined that a magnetometer will not contribute to bettering accuracy.
3. Inertial sensor	An inertial sensor detects angular changes. An accelerometer measures the forces of acceleration and a gyroscope is for measures the angular velocity.

An inertial sensor, specifically the MPU-6050, is chosen as the sensor of choice because it is capable of clearly detecting when a waste cart is being emptied by the waste collection truck.

C.6 Navigation

Table C.12 lists requirements relevant to the navigation. Based on these requirements, Table C.13 lists the possible design options.

Table C. 12: Requirements for the Navigation

Requirement ID	Requirement Description
3.1.1 - POC	The product must be capable of moving from a defined point A to a defined point B.
3.1.1.1 - POC	The product shall be able to navigate and stay within the preset virtual or physical boundaries
3.2.1 - EP	The product must be capable of recognizing the designated point A (where the user stores their waste cart) and the designated point B (waste pick-up area).
3.2.1.1 - EP	The product must stop within 10 cm of either docking station.
3.2.5 - EP	The product shall be able to recognize when it is stuck and alert the user.

Table C. 13: Design Options for the Navigation

Design Options	Specification
1. LiDAR	LiDAR technology can be used to analyze the distance to an object by measuring the time of flight of a light wave [24]. For example, the LIDAR-Lite 3 Laser Rangefinder can track a target distance accurately up to 40 meters at a cost of $172.65 CAD. Though accurate, LiDAR is too costly for the purpose of Bin Buddy [25]. LiDAR is also prone to inaccuracies and loss of information when directed to objects of inconsistent shape or width [26].
2. BLE Beacon	BLE Beacons can be used at both docking stations. The signals sent can be used to position a device. However, the accuracy decreases as the distance between the beacons increases [27]. Environment factors can also contribute to the blocking of signals, decreasing positioning accuracy [27].
3. Encoders	Encoders work best on smooth surfaces. As a result. bumpy driveways might interfere with the accuracy or contribute to offset that will be required to be accounted for. Additionally, accumulation of error increases as the distance increases. Consequently, encoders will work best alongside a calibration method at a midpoint.

4. Camera Module	A Raspberry Pi Camera module is modular and can directly interface with the Raspberry Pi at a low cost. If it is used alongside image recognition software, such as openCV, line following can be implemented. The camera module can also be used for colour detection.
5. Colour Sensor	A colour sensor is another option that can be used to detect the docking station colours; however, it cannot be utilized well for the main aspect of navigation. The outdoor setting is also not optimal for the colour sensor.

Among these options, the Raspberry Pi camera module was selected as the navigation method because it satisfies all the requirements while having the advantage of better narrowing the scope of the navigation at a low cost.

C.7 Motors

Table C.14 lists requirements relevant to the motors. Based on these requirements, Table C.15 lists the possible design options.

Table C. 14: Requirements for the Motors

Requirement ID	Requirement Description
5.3.1 - POC	The chassis shall move forwards and backwards at less than 10km/h.
5.3.2 - POC	The chassis must be able to steer right or left to change directions
5.3.1 - EP	The chassis must be able to pull the load of the waste cart and the product up an incline of 10º.
5.3.2 - EP	The chassis shall be reasonably quiet while operating.
5.3.1 - FP	The chassis drive motor shall last a reasonable lifespan of the product.
5.3.2 - FP	The chassis drive motor must be able to support a load of 50kg (45kg maximum for waste cart and 5kg maximum for product).

Table C. 15: Design Options for the Motors

Design Options	**Specification**
1. Brushed DC	A brushed DC motor consists of a permanent magnet in the rotor and electromagnets in the stator [28]. Brushes are used to detect a change in orientation to flip the current which in turn generates the rotor's rotation [28]. DC motors are generally inexpensive compared to other motors and require simple controllers for speed and direction control. They also offer the ability to adjust torque to speed ratio [28].
2. Induction	Induction motors use electromagnetic induction to produce torque [29]. These types of motors need an AC power supply [29]. Since the device is portable, a power supply in the form of a battery is more desirable. To use an AC motor with a battery, a DC to AC converter is required which adds extra weight and complexity to the device. AC motors are less efficient than DC motors since some of the energy provided is used on creating an electromagnet whereas DC motors use a permanent magnet and as such, most of the energy is used in the mechanical power output.
3. Servo or Stepper	A servo or stepper motor is not suitable for the Bin Buddy system as these motors are usually embedded in systems that require precise rotational motion. Because they come with precise motion control and feedback control (in servos), these motors are very expensive for high torque and power values [30].

Among these options, the brushed DC motor was selected as the POC and EP motor because it satisfies most of the requirements while being cost effective. It also offers high torque and power for the application.

C.8 Motor driver

Table C.16 lists requirements relevant to the motors. Based on these requirements, Table C.17 lists the possible design options.

Table C. 16: Requirements for the Motor Driver

Requirement ID	Requirement Description
5.3.1 - POC	The chassis shall move forwards and backwards at less than 10km/h.
5.3.2 - POC	The chassis must be able to steer right or left to change directions
5.3.1 - EP	The chassis drive motor must be able to pull the load of the waste cart and the product up an incline of 10^o.
5.3.2 - EP	The chassis drive motor shall be reasonably quiet while operating.
5.3.2 - FP	The chassis drive motor must be able to support a load of 50kg (45kg maximum for waste cart and 5kg maximum for product).

Table C. 17: Design Options for the Motor driver

Design Options	Specification
1. Sabertooth dual 25A motor driver	This motor controller can control two brushed DC motors, offering bi-directional and speed control of the motors. It can provide continuous current values of up to 25A per channel and a peak current of up to 50A per channel. The voltage rating of this driver is 6-30V nominal. It offers thermal and current overprotection. An advantage of this driver is that it offers regenerative topology which means the flyback energy produced from the motors when they are slowing down or reversing can be used to recharge the batteries. A disadvantage of this driver is that it is more costly than other drivers.
2. Cytron SmartDriveDuo Smart Dual Channel 30A Motor Driver	This motor controller has an H-bridge and PWM control to offer speed and direction control of two brushed DC motors. It allows continuous current drawn per channel of 30A and a peak current of 80A per channel. The operating voltage on this controller is 7 to 35V. This controller does not offer polarity protection but does offer overcurrent and thermal protection. It also has an indicator for low battery. An advantage of this controller is that it offers all the requirements for Bin Buddy application while being the more inexpensive option in controllers on the market.

Among these options, option 2, Cytron SmartDriveDuo Smart Dual Channel 30A Motor Driver, was selected because it satisfies most of the requirements while being cost effective.

C.9 Chassis Motor Power Supply

The power requirements are shown in Table C.18. Based on these requirements Table C.19 lists the possible design options.

Table C. 18: Requirements for Chassis Power Supply

Requirement ID	Requirement Description
5.2.1 - POC	The power supply must be able to power the sensors and motors
5.2.1 - EP	The power supply must last 3 roundtrip cycles before a battery recharge
5.2.2 - EP	The power supply must be able to power the chassis motor continuously for at least 5 minutes one way on a driveway that is about 20m long.
5.2.1 - FP	The power supply must be able to self-recharge at docking station
3.3.1 - FP	The product shall cost less than $500
4.4.1 - FP	The weight of the product shall not exceed 5kg
CSA C22.2 No. 0.23-15 (R2020)	General requirements for battery-powered appliances

Table C. 19: Design Options for the Chassis Motor Power Supply

Design Options	Specification
1. Lead-Acid Battery	Lead-Acid batteries are typically used in cars and some electric scooters. Their capacity to cost ratio is high. A disadvantage to using this type of battery is that it is much heavier than using lithium-ion batteries.
2. Lithium-ion Battery	Lithium-ion batteries are lightweight and have high capacity. However, the cost is much higher than lead-acid batteries. Additionally, the battery is not easily accessible for the high-power requirements required to power the motors. The longer shipping time and cost places this option at a disadvantage.

Among the two options, a lead-acid battery was selected as the motor power supply because it satisfies the power requirements at a lower cost. Bin Buddy’s design will take its larger weight and size into consideration.

C.10 Emergency Power On/Off button

Table C.20 lists requirements relevant to emergency power button. Based on these requirements, Table C.21 lists the possible design options.

Table C. 20: Requirements for Chassis Power Supply

Requirement ID	Requirement Description
8.1.1 – POC	The product must have an emergency/power button to disconnect the system from the power supply

Table C. 21: Design Options for the Line Recognition

Design Options	Specification
1. Mechanical battery kill switch	This type of switch is attached to the battery terminal wires, and it physically disconnects the connection of the rest of the circuit to the terminals of the battery. It uses a key with a conductive bolt which when in contact with terminal wires, allows a current flow for the amount of current required. When the key is rotated it disconnects the bolt and thus the current flow into the circuit. The advantage of this switch is that it is easy to implement and can support a current flow of up to 100A for a cheap price. The disadvantage is that this type of mechanical switch is not very user-friendly.
2. Isolation circuit using relay and control push-button	This design uses a relay that can handle a current draw of up to 28A. A push-button controls current flow through the relay coil. When the coil is energized, it closes the relay contacts and completes the power circuit which connects the SLA battery to the chassis motors. Since regular switches cannot handle the amount of current drawn by the Bin Buddy chassis motors, this design is beneficial as it gives the user a simple push-button interface to control a larger relay which in turn controls the large power from the battery. The disadvantage of this design is the complexity to implement.

Among the two options, option 1, mechanical battery kill switch was selected for POC since the POC design needs to be safe while also adding less complexity. Since the EP version needs to be more user-friendly, option 2, relay isolation circuit, will be implemented for safety, design aesthetic and user comfortability.

C.11 Line Recognition

In addition to the design options listed under the Navigation section, the Line Recognition section extends its design options to the techniques that can be used instead of the choices made in the main body of the document. Table C.22 lists some design options that were considered to meet the requirements listed in Table C.12 of the Navigation section.

Table C. 22: Design Options for the Line Recognition

Design Options	Specification
1. Standard Hough Transform	The Standard Hough Transform is an application of Hough transform; however, it is less optimized than the probabilistic Hough transform. Therefore, it is computationally more straining and slower.
2. Probabilistic Hough Transform	The Probabilistic Hough Transform is a more optimized version of the Hough transform application, and therefore, is computationally faster.
3. Homogeneous Smoothing/Box Blur	The Homogeneous Smoothing/Box Blur is a technique that can be used for image smoothing. This is an approximate of Gaussian Blur. If the kernel size is too large, small details of an image may be removed.
4. Gaussian Blur	The Gaussian Blur is another image smoothing technique that is most commonly used [17].

The probabilistic Hough Transform technique was chosen because it is computationally less exhaustive and faster than the standard method. The Gaussian Blur technique was chosen because it eliminates less details of an image compared to the Box Blur technique.

C.12 Front-End: User Application

Table C.23 lists requirements relevant to the front-end user application. Based on these requirements, Table C.24 lists the possible design options.

Table C. 23: Requirements for the Front-End User Application

Requirement ID	Requirement Description
3.2.6 - EP	The device must be able to be set up in less than 1 hour by a moderately mobile person of age 12 or above.
3.2.7 - EP	The product must allow the user to input a monthly collection schedule
3.2.7.1 - EP	A person with 1-2 years of experience using a smartphone shall be able to figure out how to input a monthly collection schedule in less than 20 minutes.
6.2.2 - EP	There must be an interface for users to communicate with the product with little to no user expertise and training.
6.2.3 - EP	The interface shall correctly display the product's destination.
6.2.4.1 - EP	If unable to move around the large obstacles without deviating away from the preset boundary, Bin Buddy shall alert the user.
6.2.6 - EP	The software must alert the users in case of needing attention/user action - theft, stuck somewhere.
6.3.1 - FP	The software shall boot up within 60 seconds
6.3.2 - FP	Any major, unexpected and unaccounted interrupts or failures shall alert the user.

Table C. 24: Design Options for the Front-End User Application

Design Options	Specification
1. On-Board Screen	A screen on the Bin Buddy system can be used for the input of the waste cart schedule. However, because this system will be located outdoors, the screen will be required to be waterproof. An on-board screen also means that the user will not be able to control the system remotely.
2. Website	A website allows for remote control of the system. It can be cross platform and can communicate with the microcontroller. However, it is not very portable and convenient for the typical user.
3. Mobile Application	A mobile application has all the features that a website fulfills while also providing the portability.

Among these options, the mobile application was selected as the front-end user application because it satisfies all the requirements while also providing the portability and convenience for the user.

C.13 Backend: System Communication

Table C.25 lists requirements relevant to the web server. Based on these requirements, Table C.26 lists the possible design options.

Table C. 25: Requirements for the System Communication

Requirement ID	Requirement Description
3.2.3 - EP	The product shall follow the waste collection schedule input by the user.
3.2.5 - EP	The product shall be able to recognize when it is stuck and alert the user.
5.1.1 - POC	MCU must have Wi-Fi capabilities.
6.2.4.1 - EP	If unable to move around the large obstacles without deviating away from the preset boundary, Bin Buddy shall alert the user.
6.2.6 - EP	The software must alert the users in case of needing attention/user action - theft, stuck somewhere.
6.3.2 - FP	Any major, unexpected, and unaccounted interrupts or failures shall alert the user.

Table C. 26: Design Options for the System Communication

Design Options	Specification
1. Bluetooth	Bluetooth communication allows the user to interact with a mobile application causing messages to be sent to the Bin Buddy device. Bluetooth communication, however, requires proximity to the system.
2. Web Server	A web server allows the user to interact with a mobile application which can then implement get/post requests to communicate with the backend web server hosted on the device itself. This ensures that the user has control over the device as long as a Wi-Fi connection is available, meaning proximity will not be necessary

Among these options, the web server was selected as the system communication because it satisfies all the requirements while also ensuring the user has control without requiring proximity to the system.

C.14 Backend: Database

Table 21 lists requirements relevant to the database. Based on these requirements, Table 22 lists the possible design options.

Table C. 27: Requirements for the Database

Requirement ID	Requirement Description
3.2.3 - EP	The product shall follow the waste collection schedule input by the user.

Table C. 28: Design Options for the Database

Design Options	Specification
1. MongoDB Atlas	MongoDB Atlas is a cloud database service that hosts MongoDB servers [31]. This resource is free to use and is very simple to set up.
2. Internally made DB server	Creating a personal database server allows for the choice of a NoSQL or SQL database, however, the data that needs to be stored for Bin Buddy is minimal and relatively unstructured. This option requires more work to set up and will take space and power from the microcontroller.

Among these options, MongoDB Atlas was selected as the database server because it satisfies all the requirements while being simple and free.