26th March 2021

Dr. Craig Scratchely,
School of Engineering Science
Simon Fraser University
8888 University Dr.
Burnaby, B.C. , V5A 1S6

SaniTech

RE: ENSC 405W/440 Design Specification for *B-Clean*

Dear Dr. Scratchley,

The enclosed document is the design specification document prepared by SaniTech as part of the engineering design process for our product *B-Clean*. *B-Clean* is a fully integrated basket cleaner system aimed at grocery stores in the lower mainland and in the future, nationwide. It is a product intended to make the shopping experience safer while minimizing human efforts in sanitizing baskets for consumer use. It takes in stacks of used and uncleaned baskets and outputs stacks of baskets that are clean and ready to use .

The purpose of this document is to outline the design specifications of B-Clean from a high-level system overview of the product to design details of each major component. It provides details about the design choices for each stage of our product including the Proof of Concept Stage (Alpha phase), the Prototype Stage(Beta phase), and the Production Stage (Final Product). It will consist of design details of the system architecture, the mechanical specifications,electrical specifications, software and hardware specifications. A detailed justification will be given as well for each design approach .

Our team at B-Clean would like to thank you for taking the time to review our design specification document. If you have further questions or comments related to our report, please reach out to our Chief Communication Officer at mtahmour@sfu.ca .

Sincerely,

Dean Fernandes
Chief Executive Officer
SaniTech

Design Specification:

B-Clean

Partners:

Wendsongdo Prisca Youma
Mahsa Tahmouresi
Chih Yu(David) Teng
Dean Fernandes
Keu Kang
Chaoran(Gerald) Li

Contact:
Mahsa Tahmouresi
mtahmour@sfu.ca

Submitted to :

Dr. Craig Scratchely, ENSC 405W
Dr. Shervin Jannesar, ENSC 405W
Dr. Andrew Rawicz, ENSC 440
School of Engineering Science
Simon Fraser University

Date Issued :
March 26th 2021

Version 1.0.0

Abstract

This document describes the design specifications and outlines the details for our automated shopping basket cleaner machine, B-Clean. B-Clean is a compact, lightweight and comfortable device that facilitates the basket cleaning process by intaking a number of baskets, sanitizing them and outputting them in a stack form to the user. B-clean will be placed in supermarkets beside the cashier to minimize the amount of effort for customers as well as the employees. Unlike other existing cleaning machines that are used for big shopping carts, our product focuses on cleaning baskets with higher accuracy such as cleaning the handles, all the corners, blow drying, and garbage emptying. Furthermore our product is coupled with an LCD spray display monitor that is aimed to provide the end user information regarding the spray usage settings.

This paper will review from the general system design to more detailed design specifications such as all the high-level system architecture, functionality, and implementation for each section of the B-Clean. Parts will include the subsection of :

- Control Unit Module
- Basket Setup Module
- Basket Sanitization Module

The document also outlines the details of the three phases of the product development: Alpha phase/ proof of concept. Beta phase/ prototype phase, and Final phase. The related test plans and all the additional applications related to the user interface design are also included.

Table of Contents

Table of Figures

Table A.1: List of Figures used in Design Documents

List of Tables

Table A.2: List of tables used in Design Documents

Glossary

Acronym	Definitions
LCD	Liquid Crystal Display
LED	Light Emitting Diode
GUI	Graphical User Interface
B-Clean	Product name for the automated basket sanitizing machine
ISO	International Organization for Standardization
IEEE	Institute of Electrical and Electronics Engineers
NEMA	National Electrical Manufacturers Association
AC	Alternating Current
DC	Direct Current
Max.	Maximum
FETs	Field-Effect Transistor
RAM	Random Access Memory
GPIO	General-Purpose Input/Output
CPU	Central Processing Unit
SoC	System on Chip
GB	Gigabit

Table A.3: List of Acronyms used in Design Documents

1 Introduction

B-Clean is a fully integrated and automated basket sanitization device with the goal of making the shopping experience safer and enjoyable for customers. This device offers not only cost-effectiveness, reliability, sustainability and convenience; but also offers grocery store customers comfort and peace of mind. B-Clean is intended to be a new way to clean baskets effortlessly, and tool-free. Indeed, the traditional way of cleaning baskets is no longer able to fulfill the current growing demand of thoroughly sanitizing a considerable number of baskets in the stores. However, today more than ever before, sanitizing store's baskets is a key requirement in order to keep people safe. Our system B-Clean offers the features that satisfies these requirements. B-Clean replaces the manual and repetitive task of sanitizing the baskets and also ensures full coverage of the basket's surface. It takes in single or stacks of baskets and outputs cleaned baskets ready to be used again. With our system in place, stores can focus on their goal of serving customers.This document highlights the design specifications and design approach of our product B-Clean.

1.1 Background

In the light of the Covid-19 pandemic and other rapidly spreading diseases around the world, sterilizing and sanitizing have become essential in order to not only protect self, but others as well. In order to abide by these safety requirements from health authorities most stores have started to sanitize their baskets. However, this task is done manually as stores are attempting to be held at higher standards by creating a safe shopping environment for their customers.This repetitive task is often not done 100 % accurately due to human error in reaching all the surface area of the basket with the sanitizer. In order to attempt to solve this problem, our team conducted interviews in different local grocery stores. We discovered that a substantial number of baskets have to be cleaned daily in medium to large size stores. This is done manually for long hours using a sanitizing liquid by store employees or third parties hired to fulfill this role. These observations come to demonstrate that there is a need for this task to be automated.

1.2 Scope

The aim of this document is to outline the technical design specification for our product B-Clean. This design specification includes comprehensive details and requirements that differentiate between various phases of production, such as proof-of-concept, prototype, and final product. Thus, through this document, the reader would be able to understand the product's intended functionality.

1.3 Intended Audience

This document is intended to be used by all project members at SaniTech for the development and implementation of B-Clean. It will be used as a fundamental guideline for our proof-of-concept demo though some of the design approach may change as we move forwards in the development of our project. This document is also prepared for the instructional team of ENSC 405/440 as well as potential stakeholders seeking to understand our design choices and approach.

1.4 Design Specification Classification

For ease of the reader, the document will use the following convention to identify design specifications:

DES-<Section>.<Subsection1>.<Subsection2>.<Number>-<Development Stage>

The format of the convention is :

- DES: short form of design
- Section: major group of design specifications
- Sub-Section 1: sub-group for section
- Sub-Section 2: optional sub-group for sub-section 1
- Number: identification of requirements in sub-section (1 or 2)
- Development Stage: approximate stage for design specification, see Table 1.4.1

Development Stage Encoding	Description
A	Proof of Concept Stage (Alpha Phase)
B	Prototype Stage (Beta Phase)
F	Production Stage (Final Product)

Table 1.4.1 - Development Stage Encoding Look Up Table

2 Control Unit Module

2.1 Embedded System Control Unit

The Embedded system consists of three major electrical modules: basket setup module, sanitization module and central control module. The central control module is physically wired to other electrical blocks and it hosts software control programs for the corresponding sub-systems. The heart of the central control module is the system control unit, i.e, Main processing Unit. The control unit is selected based on requirements regarding SoC's size and portability, peripheral compatibility, system expandability, power supply configuration, system inputs, digital control, multitasking capability, system memory, external display and chip heat management. The chosen central control unit is Raspberry PI 4 Model B with 8GB RAM, this section focuses on design specifications related to the chosen SoC.

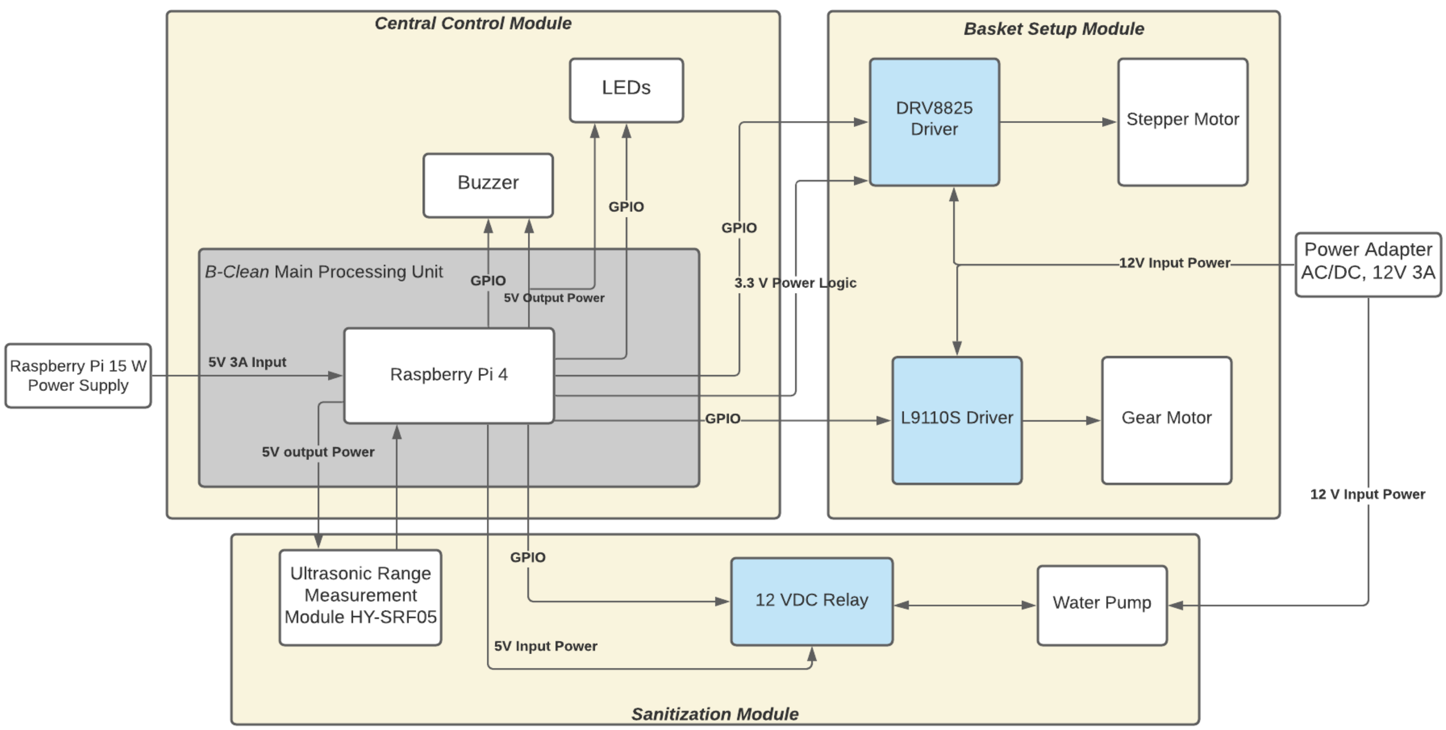


Figure 2.1.1 - Overview of B-Clean Embedded System

2.1.1 Size and portability

Overall Specifications:

Requirement [REQ-3.1.2.1-H.B] implies the control unit must be compact and portable.

Calculations and Analysis:

The control unit will be mounted on one side of the machine frame which has a dimension of 540 mm x 1000 mm (width x height). Control unit's width is expected to be less than or equal to ⅓ of frame width and its height is expected to be less than or equal to ⅕ of the frame height, thus the desired dimension is less than or equal to 18 mm x 200 mm (width x height). Raspberry PI 4 Model B has a dimension of 85.6mm × 56.5mm, which satisfies expectation.

2.1.2 Peripherals compatibility and system expandability

Overall Specifications:

Requirements [REQ-3.1.2.2-H.A] and [REQ-3.1.2.3-L.B] imply that the control unit must support buttons as peripherals. Requirements [REQ-3.1.2.5-L.B] implies that the control unit must be equipped with a speaker or support speaker peripheral. Requirements [REQ-3.3.1.1-L.F] implies object detection sensor must be supported, requirement [REQ-3.3.1.6-H.A] implies pressure sensor or distance detection sensor must be supported.

Calculations and analysis:

The central control unit must be highly expandable to be compatible with various types of system peripherals. Raspberry PI 4 Model B supports peripherals including buttons, speaker, analog sensors and digital sensors.

2.1.3 Power supply

Overall Specifications:

Requirements [REQ-3.2.3.3-H.F] implies the control unit must run with 5V @ 2A DC power supply.

Calculations and Analysis:

Raspberry PI 4 Model B can run with a power supply of 5V @ 2A DC power supply, it's recommended power supply is 5V @ 3A DC power for peak performance [6].

2.1.4 System Inputs and Digital control

Overall Specifications:

Requirements [REQ-3.1.2.2-H.A] , [REQ-3.1.2.3-L.B], [REQ-3.3.1.1-L.F] and [REQ-3.3.1.6-H.A] imply that the control unit must support collecting inputs from buttons, object detection sensor and pressure sensor or distance detection sensor.Requirements [REQ-3.3.1.4-H.A] and [REQ-3.3.1.5-H.A] implies the central control unit must support digital control of motors and water pump.

Calculations and Analysis:

The central control unit must be equipped with GPIOs to collect system inputs via system peripherals and to digitally control external electronic devices. The system requires two ultrasonic sensors, two bi-polar stepper motors, two geared motors and one relay to switch on the water pump. Each ultrasonic sensor, bi-polar stepper motor and geared motor requires two GPIO pins and the relay only requires a high DC voltage pin instead, thus the control unit must be equipped with a minimum of 12 GPIO pins to fully control the system modules. Raspberry PI 4 Model B is equipped with 40 GPIO pins which satisfies minimal GPIOs requirements [6].

2.1.5 Concurrent Programming

Overall Specifications:

Requirements [REQ-3.3.1.1-L.F], [REQ-3.3.1.4-H.A], [REQ-3.3.1.5-H.A], [REQ-3.3.1.6-H.A], and [REQ-3.3.1.7-L.F] imply the control unit must support multitasking to monitor and control multiple system modules at the same time.

Calculation and Analysis:

If the central control unit has a solo-core configuration, it must be coupled with an operating system that has a reliable scheduler to achieve multitasking. Or the central control unit may have multiple CPUs to achieve real parallel programming. Raspberry PI 4 Model B is equipped with 4 CPU cores where each CPU does not have hyper-threading, thus 4 hardware threads are supported by the processing unit.[6] The chosen hardware supports reliable operating systems such as Ubuntu, Yocto Projects, Rasbian etc. Multitasking libraries are supported along with hardware thus fulfilling the system's multitasking requirement.

2.1.6 Memory

Overall Specifications:

Requirements [REQ-3.3.1.8-L.F] and [REQ-3.3.2.6-L.F] imply the central control unit must support non-volatile memory for data storage between power cycles.

Calculation and Analysis:

The central control unit must support a form of non-volatile memory such as a hard disk to store data between power cycles. Raspberry PI 4 Model B is booted from a SD card where the SD card contains rootfs, thus it serves as a non-volatile memory of the system. Additional memory can also be easily expanded when needed.Thus the chosen control unit fulfills the memory requirement.

2.1.7 Display

Overall Specifications

Requirement [REQ-3.3.2.1-H.B] implies the central control unit must support LCD screen and graphic encapsulation.

Calculation and Analysis

The central control unit must be able to compose graphics via X11 protocol, Wayland or frame buffer and it must support LCD as peripheral. Raspberry PI 4 Model B is compatible with Linux based operating systems, graphical user interface can be developed and deployed onto the control unit when the proper toolchain and libraries are installed. With 8G RAM the control unit will ensure fluent graphical user experience. Touch sensitive LCD screens are also supported for easy user-machine interaction.

2.1.8 Heat Management

Overall Specification

Requirement [REQ-3.5.1.6-H.B] implies the central control unit must support chip temperature monitoring mechanism, and there should be a heat sink available in the market to install on critical computation units.

Calculations and Analysis

The central control unit must report when any extensive heat has been detected during operation to prevent system damage. Raspberry PI 4 Model B reports to the user when system overhead has been found and there are head sinks to be installed to the SoC for better temperature management.

2.1.9 Design Specifications

Design Specification	Description	Corresponding Requirement	Supplement Material
DES-2.1.1.1-A	Raspberry PI 4 Model B is compact with dimension smaller than 18 mm x 200 mm (width x height)	[REQ-3.1.2.1-H.B]	Section 2.1.1

DES-2.1.2.1-A	Raspberry PI 4 Model B supports peripherals including buttons, speaker, analog sensors and digital sensors.	[REQ-3.1.2.2-H.A] [REQ-3.1.2.3-L.B] [REQ-3.1.2.5-L.B] [REQ-3.3.1.1-L.F] [REQ-3.3.1.6-H.A]	Section 2.1.2
DES-2.1.3.1-A	Raspberry PI 4 Model B can run with a power supply of 5V @ 2A DC power supply	[REQ-3.2.3.3-H.F]	Section 2.1.3
DES-2.1.4.1-A	Raspberry PI 4 Model B is equipped more than 12 GPIOs pins	[REQ-3.1.2.2-H.A] [REQ-3.1.2.3-L.B] [REQ-3.3.1.1-L.F] [REQ-3.3.1.6-H.A] [REQ-3.3.1.4-H.A] [REQ-3.3.1.5-H.A]	Section 2.1.4
DES-2.1.5.1-A	Raspberry PI 4 Model B is capable of running 4 hardware threads with a reliable operating system. Multitasking libraries are supported.	[REQ-3.3.1.1-L.F] [REQ-3.3.1.4-H.A] [REQ-3.3.1.5-H.A] [REQ-3.3.1.6-H.A] [REQ-3.3.1.7-L.F]	Section 2.1.5
DES-2.1.6.1-A	Raspberry PI 4 Model B is equipped with Non-Volatile memory and memory can be expanded.	[REQ-3.3.1.8-L.F] [REQ-3.3.2.6-L.F]	Section 2.1.6
DES-2.1.7.1-A	Raspberry PI 4 Model B is compatible with an external LCD display.	[REQ-3.3.2.1-H.B]	Section 2.1.7
DES-2.1.8.1-A	Raspberry PI 4 Model B is equipped with heat management.	[REQ-3.5.1.6-H.B]	Section 2.1.8

Table 2.1.9.1 - Control Unit Summary

2.2 Software Specifications

2.2.1 Central Control Module

[REQ-3.3.1.1-L.F] implies that the software must be able to read data from sensors, record that there are baskets to clean, and start the Basket Setup Module and the Basket Sanitization Module. [REQ-3.3.1.9-H.B] implies that once the sensor detects no more baskets are left to clean, the system must stop all the modules and go into standby mode. [REQ-3.3.1.2-H.B] implies that the software must have an algorithm that can record from sensors about the basket size and set up gears in Basket Setup Module to handle the recognized size. [REQ-3.3.1.3-L.F] implies that the software must have an algorithm that can process all the physical inputs using sensors and cameras to detect if the input is valid. If the input is not valid, the software must stop all the modules in the system and alert the operator through the GUI. If the input is valid, [REQ-3.3.1.10-L.F] implies that the software must check for garbage in the basket and initiate the necessary modules to remove the garbage. [REQ-3.3.1.4-H.A], [REQ-3.3.1.5-H.A] implies that the software must be able to fully control all the modules in the system: Basket Setup, Basket Sanitization and Basket Roundup. [REQ-3.3.1.7-L.F], [REQ-3.3.1.8-L.F] implies that the software must be able to keep track of all the modules, their last service dates and notify operators when a module is malfunctioning. [REQ-3.3.1.6-H.A] requires the software to rely on the data provided by sensors to notify the user on the amount of sanitizer left in the system.

2.2.1.1 Central Control Module Programming Language

As the control unit being used to run the system is a Raspberry Pi 4, there are certain native languages setup by default in the system. C, C++, Python are all natively in the system while Java can be installed easily. A comparison of the languages can be found in Table B.1.2.1 of Appendix B.

For the purpose of setting up internal firmware of the system, Python seems to be a good option considering that development time for Alpha Stage and Beta Stage of the product is very less; Python's fast coding attributes will help speed up software development time. The system may need to expand into image processing and AI to detect basket size to process multiple basket sizes and to detect if there is garbage in the system so that it can be cleaned carefully; Python has established libraries for AI that can be quickly imported for B-Cleans future development. As the system requires a reactive user interface while running multiple modules to process a basket and at the same time log the actions in the system; B-Clean requires a language that can handle multi-threading which is found in Python.

2.2.1.2 Architecture

In the current system configuration, there are four major internal software modules that form the basis of the system's central Control Module: Graphical user interface, Basket Setup Control Module, Basket Sanitization Control Module and System Logging Module (Figure 2.2.1.2.1). A two tier software architecture is implemented, where Graphical User Interface serves as master to other three modules. Refer to section 2.2.2 for detailed Graphical User interface design. The remaining section will provide an overview regarding Basket Setup Control Module, System Logging Module, Basket Sanitization Control module and system workflow design.

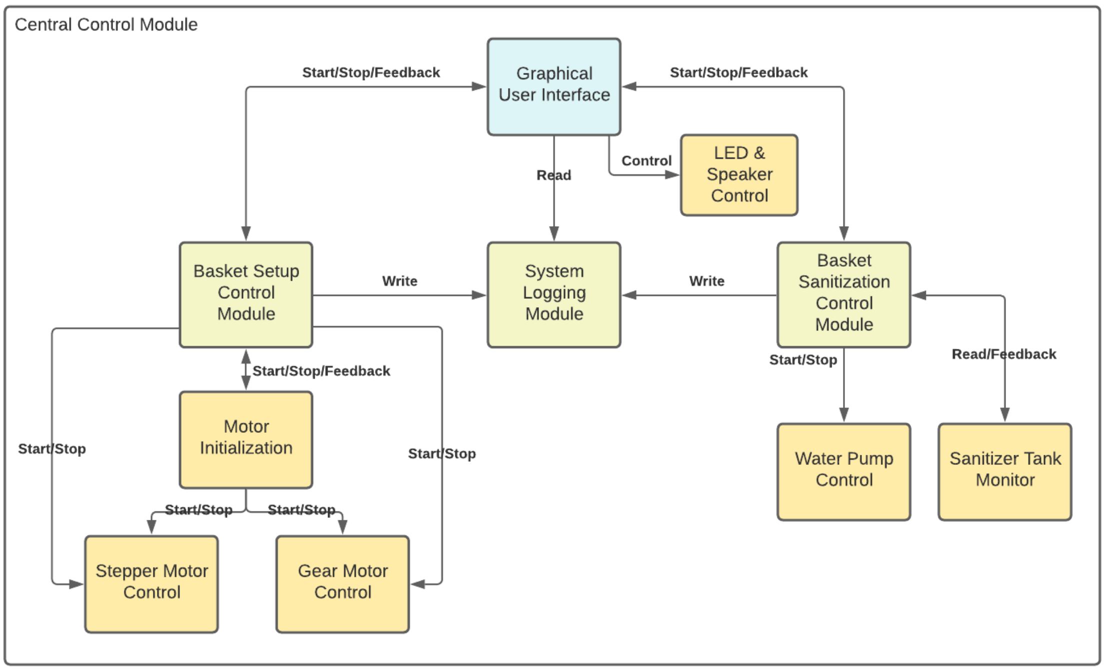


Figure 2.2.1.2.1 - Overview of B-Clean Software System

Basket Setup Module

The Basket Setup Module will need to control two stepper motors for basket separating and two geared motors to deliver the basket to the sanitization chamber. The justification, explanation and design of the motors can be found in Section 3 Basket Setup Module. The motors need to be controlled separately as they need to be operational simultaneously and need to be delineated clearly for multi-threading to work on the tasks efficiently and error free. These motors can then report back the operational status for logging purposes to the GUI. The Motors will be controlled using voltage controls on motor drivers (talked about in Section 3.1 Electrical Design). These will help control the speed and consistency of the motors during rotation. The algorithm used will also ensure that the software will be able to keep the gears even to prevent baskets getting stuck within the system.

Basket Sanitization Module

The Basket Sanitization Module will need to control a water pump to spray the sanitizer in the spray system. The module will also monitor sensors in the Sanitation Liquid Tank to keep track of the amount of Sanitizer left in the system. The justification, explanation and design of the water pump and the sensors can be found in Section 4 Basket Sanitization Module. Similar to the motors in Basket Setup Module, the water pump and the sensors need to be able to work simultaneously during the operation and hence need to be separated for multi-threading to work. The water pump and sensors can then report back the operational status for logging purposes to the GUI. The Water Pump will be controlled by Voltage relays which can control the outflow of the pump to ensure that sprays will only turn on when the basket is reachable by the sprays. The sensors will be placed close to the motor so that it can monitor and report back the remaining sanitizer liquid.

System Logging Module

The System Logging Module will need to serialize system event logging. It has interfaces to other system modules and will be used for GUI display and system error handling and maintaining purposes.

System Workflow

System modules run in parallel, events are triggered by either user inputs via GUI or system sensors. System notifies users when certain system states have been achieved or an error has been detected. See Figure 2.2.1.2.2 for detailed system workflow.

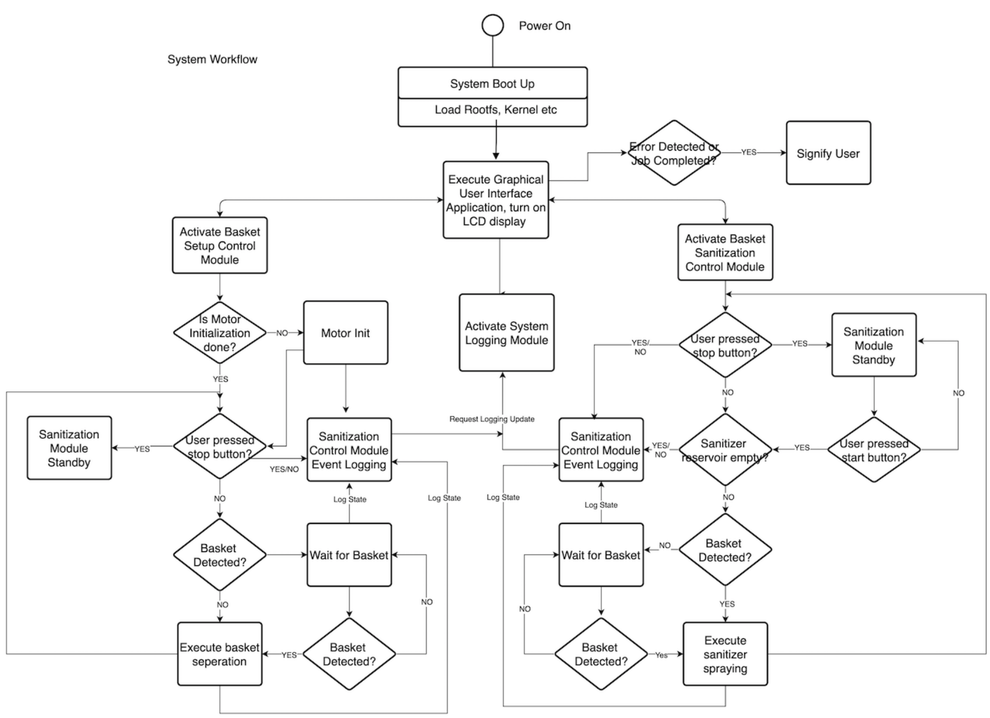


Figure 2.2.1.2.2 - System WorkFlow

2.2.2 Graphical User Interface (GUI)

[REQ-3.3.2.1-H.B] implies that the system needs a screen that can display a GUI to notify an operator about the system and let the operator modify the system for their use. [REQ-3.3.2.2-H.A], [REQ-3.3.2.3-L.F], [REQ-3.3.2.4-H.B] imply that the software must notify through the GUI about completion of a basket cleaning process and if not completed, the reason why it was not completed. The GUI must also display the health of individual modules in the system and the amount of sanitizer left that can be used. [REQ-3.3.2.5-H.B] defines that there will be two interfaces for the GUI: Operator Interface and Maintenance Interface. [REQ-3.3.2.6-L.F] defines the operator interface as displaying the sanitizer liquid level, total number of baskets cleaned, and notifications related to the system. [REQ-3.3.2.7-L.F] defines the maintenance interface as displaying the sanitizer liquid level, total number of baskets cleaned, last service of the mechanical modules, and currently malfunctioning modules

2.2.2.1 GUI Programming Language

As the control unit being used to run the system is a Raspberry Pi 4, there are certain native languages setup by default in the system. C, C++, Python are all natively in the system while Java can be installed easily. A comparison of the languages can be found in Table B.1.2.1 of Appendix B.

For the purpose of setting up a GUI for the system, Python seems to be a good option again. Considering that development time for Alpha Stage and Beta Stage of the product is very less; Python's fast coding attributes will help speed up software development time. Python allows for multi-threading which helps keep the system reactive. With libraries like Tkinter, Kivy and QTpy, there are a significant number of UI frameworks that can be used to create a user friendly and reactive GUI. A comparison of these frameworks can be found in Appendix B, Table B.1.2.2.

For the purposes of keeping dependencies low, lots of resources on the framework and for a quick development cycle, the GUI has been implemented with Tkinter Python Framework.

2.2.2.2 GUI Architecture

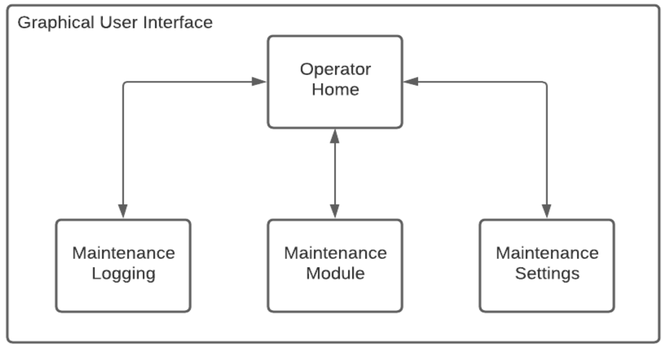


Figure 2.2.2.2.1 - GUI Architecture

The interface based on Figure 2.2.2.2.1, there are 4 major sub-modules that build up the GUI: Operator Home, and Maintenance Logging, Maintenance Module and Maintenance Settings.

Operator Home

The Operator Home page will display information on the system status, the amount of sanitizer level remaining in the system, the total number of baskets cleaned, date/time, and notification alerts related to the system errors. The page will also allow the user to input actions such as: starting and stopping the entire system; logging on to the maintenance module. An example of the page layout can be found in Appendix A Figure A-5.

Maintenance Logging

The Maintenance Logging page will display all the logging input from the Basket Setup Module and the Basket Sanitization Module. Certain events will be colour coded to quickly direct maintenance staff on the logs of interest. The page will also have all the information displayed in the Operator Home page. The page will also allow the user to input actions such as: starting and stopping the entire system; logging back to the Operator Home Page or switch to Maintenance Module page and Maintenance Settings page. An example of the page layout can be found in Appendix A Figure A-6.

Maintenance Module

The Maintenance Module page will display all the electronic modules from the Basket Setup Module and the Basket Sanitization Module. The modules will be listed with their name, number of months since last maintenance, number of issues logged for that module since last maintenance, and the system recommendation on whether the module needs to be looked at in detail by SaniTech staff. The page will also have all the information displayed in the Operator Home page. The page will also allow the user to input actions such as: starting and stopping the entire system; logging back to the Operator Home Page or switch to Maintenance Logging page and Maintenance Settings page. An example of the page layout can be found in Appendix A Figure A-7.

Maintenance Settings

The Maintenance Settings page will display all the modifiable settings allowed by the control system from the Basket Setup Module, Basket Sanitization Module and the Control Unit itself. The settings will include controlling the amount of sanitizer used per basket, manually setting the input basket size, setting the system in standby mode for maintenance and resetting to system defaults. In future development, mode settings will be added to allow for a mode user oriented machine. The page will also have all the information displayed in the Operator Home page. The page will also allow the user to input actions such as: starting and stopping the entire system; logging back to the Operator Home Page or switch to Maintenance Logging page and Maintenance Module page. An example of the page layout can be found in Appendix A Figure A-8.

2.2.3 Design Specifications

Design Specification	Description	Corresponding Requirement	Supplement Material
DES-2.2.1.1-A	The central control module will control the systems internal firmware and will be coded on Python.	REQ-3.3.1.2-H.B REQ-3.3.1.3-L.F REQ-3.3.1.10-L.F	-N/A-
DES-2.2.1.2-A	The central control module will have modularized code similar to the system modules: Basket Setup Module Code and Basket Sanitization Module code.	REQ-3.3.1.4-H.A REQ-3.3.1.5-H.A	-N/A-
DES-2.2.1.3-A	The Basket Basket Setup Module Code will be able to control and monitor the Stepper Motor and Gear Motor.	REQ-3.3.1.7-L.F REQ-3.3.1.8-L.F REQ-3.3.1.1-L.F REQ-3.3.1.9-H.B	-N/A-
DES-2.2.1.4-A	The Basket Basket Sanitization Module Code will be able to control and monitor the Sanitizer pump and the Sanitizer Tank Level Sensor.	REQ-3.3.1.7-L.F REQ-3.3.1.8-L.F REQ-3.3.1.6-H.A	-N/A-
DES-2.2.2.1-A	The GUI module will be controlling the LCD Screen for User-System Interaction and will be coded on Python.	REQ-3.3.2.1-H.B	-N/A-
DES-2.2.2.2-A	The GUI module will have modularized code for 4 different pages: Operator Home, Maintenance Logging, Maintenance Module and Maintenance Setting.	REQ-3.3.2.2-H.A REQ-3.3.2.3-L.F REQ-3.3.2.4-H.B REQ-3.3.2.5-H.B	-N/A-

DES-2.2.2.3-A	The Operator Home Page will display operator necessary information: sanitizer level, number of baskets cleaned and system error notifications	REQ-3.3.2.6-L.F	Appendix A Figure A-5
DES-2.2.2.4-A	The Maintenance Logging Page will display all logs from the system modules.	REQ-3.3.2.7-L.F	Appendix A Figure A-6
DES-2.2.2.5-A	The Maintenance Module Page will display information related to specific electrical parts in the system.	REQ-3.3.2.7-L.F	Appendix A Figure A-7
DES-2.2.2.6-A	The Maintenance Settings Page will let the operator modify system settings.	REQ-3.3.2.7-L.F	Appendix A Figure A-8

Table 2.2.3.1 - Software Specification Summary

3 Basket Setup Module

3.1 Electrical Design

3.1.1 Motor Driver Specifications

There are two stepper motors and two gear motors that are driven by two separate motor drivers. The motor drivers will then be connected to the Raspberry Pi4 to allow the process unit to control the motors. In order to execute desirable control of the motors, the following design specifications must be met, where the general design specifications apply to all motor drivers and the design specifications for stepper motor apply to the stepper motor driver exclusively.

Design ID	Design Description	Corresponding Requirement	Supplemental material
DES-3.1.1-A	Motor drivers must be able to operate with DC 12V input voltage.	REQ-3.2.3.2-H.F REQ-3.2.3.4-H.F	*Table 3.1.2.1* *Table 3.1.3.1*
DES-3.1.2-A	Motor drivers must have their own circuit protection.	REQ-3.2.2.1-H.F	*Table 3.1.2.1* *Table 3.1.3.1*
DES-3.1.3-A	Motor driver’s PCB must be compact and relatively small in dimension in comparison with the frame of the *B-Clean* system.	REQ-3.1.2.1-H.B	*Figure 3.1.2.1* *Figure 3.1.3.1*
DES-3.1.4-A	Motor drivers must be compatible with Raspberry Pi 4.	NA	NA

Table 3.1.1.1 - General Design Specifications for Motor Drivers

Design ID	Design Description	Corresponding Requirement	Supplemental material
DES-3.1.5-A	The stepper motor driver must have at least five different stepping modes.	NA	*Table 3.1.2.1*

Table 3.1.1.2 - Design Specifications for Stepper Motors

According to the Table 3.1.1.1 and Table 3.1.1.2 design specifications, the motor drivers DRV8825 and L9110S meet the specifications; hence, these motors are chosen to be employed in the *B-Clean* system.

3.1.2 Stepper Motor Driver Modules Specifications

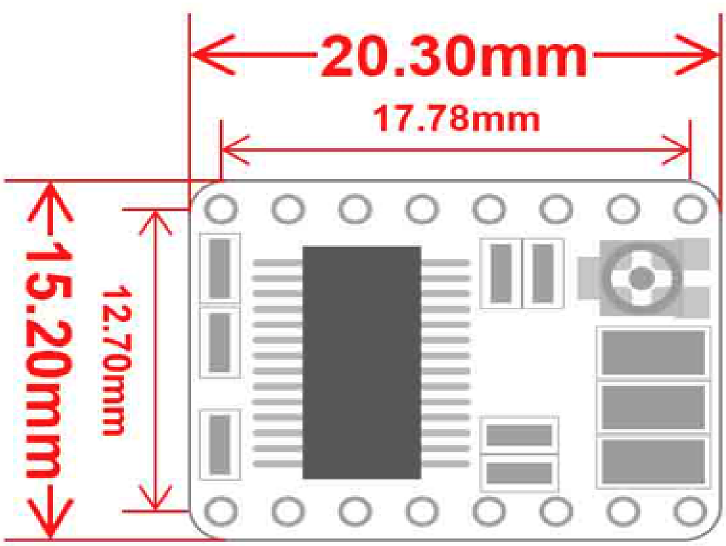


Figure 3.1.2.2: Pin Label of DRV8825 Driver

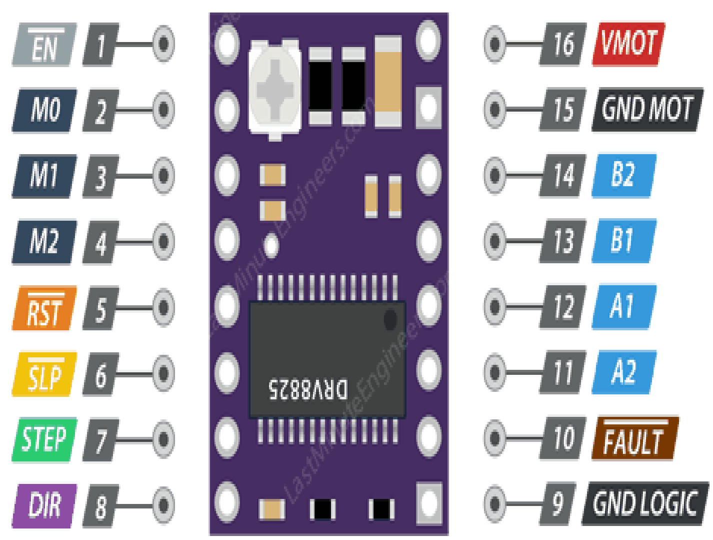


Figure 3.1.2.1 : Dimension of DRV8825 Driver

DRV8825 Specifications	
Minimum Operating Voltage	8.2V~45V
Continuous Current Per Phase	1.5A~2.2A
Logic Voltage	2.5V~5.25V

Microstep Resolution	Full, ½, ¼, ⅛, 1/16, 1/32
Operating Junction Temperature Range	-40~150 ℃

Table 3.1.2.1 - DRV8825 Motor Driver Specifications

DRV8825 Pins and Descriptions	
Pin	Descriptions
EN	Enable: default state of this pin is enabled. Default is set to low.
M0	Resolution selector for Microstepping (step size)
M1	Resolution selector for Microstepping (step size)
M2	Resolution selector for Microstepping (step size)
RST	Reset: Connected to 5V to power on the driver
SLP	Sleep: Connected to 5V to power on the driver, else setting this pin to Low will set the driver in sleep mode
STEP	Connect to GPIO of Raspberry Pi for controlling microsteps of the motor. Faster the input pulses, the faster the motor will rotate
DIR	Connect to GPIO of Raspberry Pi for controlling the direction of the motor. Set it high for clockwise motion and low for counterclockwise.
VMOT	Connect to 8.2V~45V
GND	Motor Ground
B2	Connect to Stepper Motor
B1	Connect to Stepper Motor
A1	Connect to Stepper Motor
A2	Connect to Stepper Motor
FAULT	Connect to Logic Power Supply 3.3V. (drives LOW if the H-Bridge FETs are disabled due to over-current or thermal shutdown protection)
GND	Logic Ground

Table 3.1.2.2 - DRV8825 Motor Driver Pin Descriptions

3.1.3 Gear Motor Driver Modules Specification

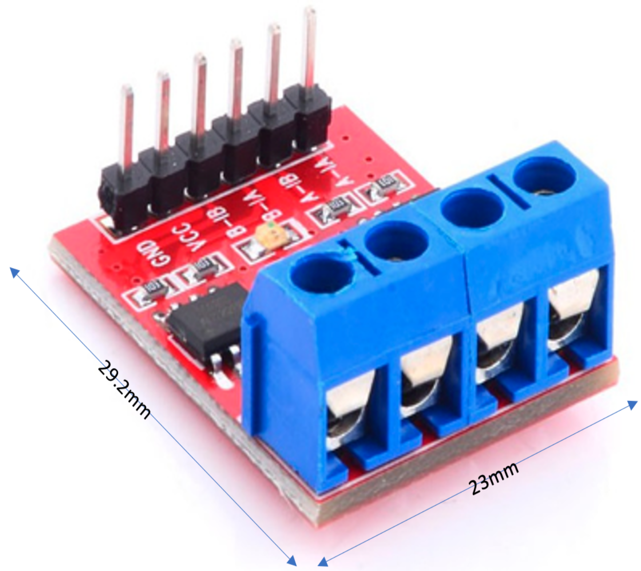


Figure 3.1.3.1 - Dimension of L9110S Gear Motor Driver

L9110S Motor Driver Specifications	
Input Voltage	2.5V~12V
Output Current	800mA
Operating Temperature	0-80℃

Table 3.1.3.1 - L9110S Motor Driver Specifications

L9110S Pin and Descriptions	
Pin	Descriptions
GND	Ground: Connect to External GND
VCC	Connect to 2.5V~12V Input Voltage
B-IB	Motor B Input B
B-IA	Motor B Input A
A-IB	Motor A Input B
A-IA	Motor A Input A

Table 3.1.3.2 - L9110S Motor Driver Pin and Descriptions

3.1.4 Motor Drivers Circuit Design

To separate the baskets one by one as they enter, *B-Clean* uses a circuit that synchronously controls the two motors. As seen in the figure *3.1.4.1*, the power supply provides a 3.3 V input to both the Raspberry pi 4 and the DRV8825-Drive module. The DRV8825-Drive module has eight pins (table 3.1.2.2), we connected the Reset and Sleep pins to the power logic power supply that goes to VCC of the Raspberry pi. Next, we connected the GND of Raspberry pi to that of DRV8825 module. The Step and the DIR pin also connects to the Raspberry pi. For connecting the motos power supply to the module we used a 100 microfarad capacitor as an intermediate to connect it to the GND and VMOT of the module. The {A1,A2,B1,B2} pins are considered for motor connection.

DRV8825 Motor Driver supports current limit functionality and it can be set through the potentiometer on the circuit board. The current limit can be set by setting the reference voltage. The mathematical relationship between current limit and reference voltage is as follow:

$$\text{Current Limit} = VREF * 2$$

The current limit is dependent on the maximum stepper motor current rating per phase. For instance, if the stepper motor has a current rating of 2.8A per phase then the VREF will become:

$$2.8\ A = VREF * 2$$

$$VREF = 2.8/2 = 1.4V$$

Consequently, the corresponding VREF will have to be set to 1.4V to ensure that the current limit is 2.8A.

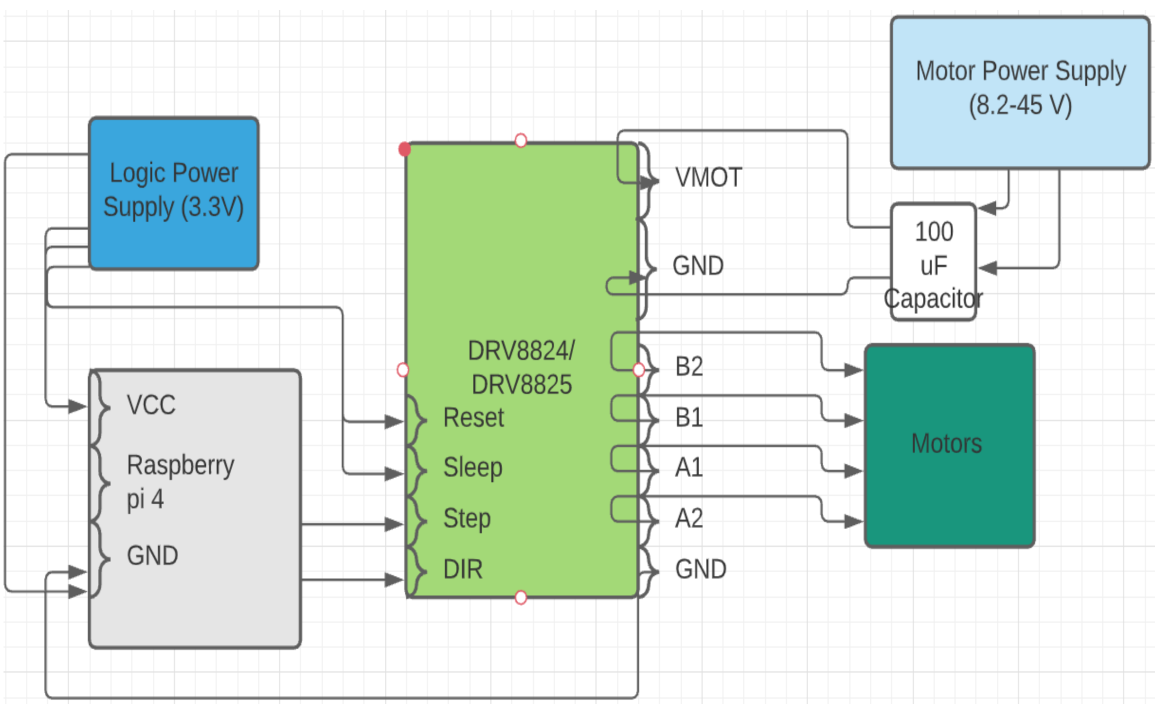


Figure 3.1.4.1 - DRV8825-Drive-Board

In figure 3.1.4.1 the L91105 DC stepper motor driver H-Bridge is used to be connected to the two dc gear motors, gear motor A and gear motor B. The L91105 DC stepper motor has six pins as described in the *3.1.3.2* table. We connected the Raspberry pi to {A-A,...B-B} of the L91105 DC module. The VCC and GND of this module goes to the motor power supply.

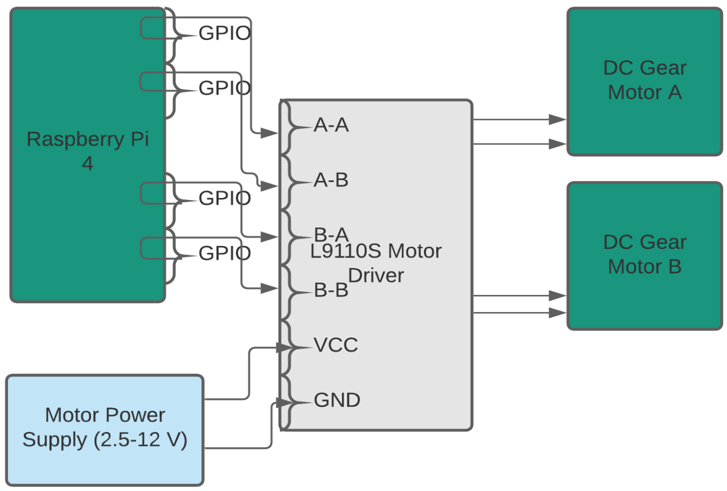


Figure 3.1.4.2 - Gear motor driver L9110S

The below table describes the design requirements related to the circuit used in *B-Clean*.

Design Specification	Description	Corresponding Requirement	Supplemental material
DES-3.1.6-A	Raspberry pi 4 model B, 8GB RAM shall be used to control the components	REQ-3.2.3.3-H.F	Figure 3.1.4.1
DES-3.1.7-A	100 uF Capacitor must be used to connect between VMOT and Motor Power Supply.	NA	Figure 3.1.4.1
DES-3.1.8-A	The ground of the Raspberry pi must be connected to the ground of the drive board.	NA	Figure 3.1.4.1

Table 3.1.4.1 - Circuit design specifications

3.1.5 Microcontroller

The design specification related to the microcontroller is listed in the table. Limited space will restrict the size of the microcontroller. The microcontroller should operate on 3.3V due to minimizing the power consumption, which will lead to longer operating life and will minimize power-supply requirements.

Design Specification	Description	Corresponding Requirement	Supplemental material
DES-3.1.10-A	Raspberry pi 4 model B, 8GB RAM shall be used to control the components	REQ-3.2.3.3-H.F	Figure 3.1.4.1 Figure 3.1.4.2
DES-3.1.11-A	*B-Clean* must use the 400 Holes Breadboard to connect one Raspberry Pi with two motors and other components	NA	NA
DES-3.1.12-A	150MM 10/PKG breadboard jumper wires to connect the raspberry pi to the breadboard	NA	NA
DES-3.1.13-A	Microcontroller must have the following operation range: -40 to 105°C	REQ-3.2.4.4-H.F	NA

Table 3.1.5.1 - Microcontroller design specifications

3.1.6 Power supply

A power system is needed to drive the motors, driver board, the Raspberry Pi 4 and all the related components. Jumpers are used for the data transfer. For the power supply we choose 12V 3A AC/DC since the stepper motor requires a high input power source to be able to have a high output torque. Therefore the power supply needs to accommodate that. Also the power supply has to drive all the stepper and gear motors. The following table gives design details about the power supply. It is noticeable that for the proof-of-concept stage we use one power supply to charge all the components.

Design Specification	Description	Corresponding Requirement	Supplemental material
DES-3.1.16-A	15W Power Supply (Ctype) shall be used to connect the pi to the outlet	REQ-3.2.4.3-H.B	NA
DES-3.1.17-A	The power supply shall have high enough capacity 5000 mAh to power the components.	NA	Section 3.1.5
DES-3.1.18-A	The microcontroller must be powered with a 3.3	REQ-3.2.3.3-H.F	Figure 3.1.4.1

	V power supply		
DES-3.1.19-A	USB 3.0 (A type) to USB 3.0 (C type) cable to transfer power	REQ-3.2.4.3-H.B	NA

Table 3.1.6.1 - Power Supply

As the table 3.1.6.1 suggested, two power supplies will be needed in the project, one is USB-A to USB-C to pi and the other one is used to connect the power Raspberry Pi 4 to the outlet.

3.1.7 Monitoring system circuitry

For the monitoring system initially we will be using the raspberry pi's LCD that comes with it and later we will replace that with the bigger LCD.

3.1.8 Calculations

Considering the maximum current draw for Raspberry pi model B to be 500 mA [1] we can estimate the power supply capacity for a minimum of 10 hours working:

10 h * 500 mA = 5000 mAh

The calculated value is the value of the battery we need for 10 hours working with a minimum current consumption.

3.1.9 Overviews of Motors

Stepper motors and gear motors have distinct characteristics regarding the pins and operating conditions. In this section, each motor's features will be displayed in the form of diagrams and tables.

3.1.9.1 NEMA 23 Stepper Motors

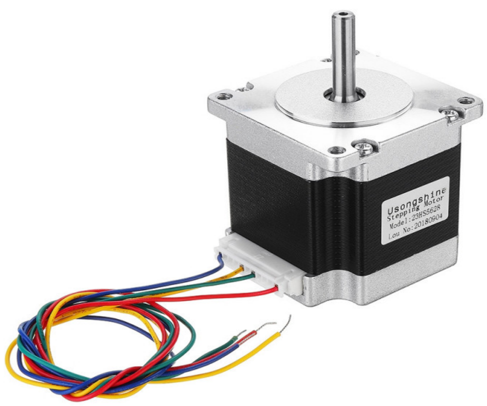


Figure 3.1.9.1.1 - Diagram of the Stepper Motor

NEMA 23 Stepper Motor Specifications	
Voltage	3.2V
Current per Phase	2.8A
Step Angle	1.8 °
Steps Per Revolution	200
Holding Torque	270 oz. in

Table 3.1.9.1.1 - Table of Detailed Specifications of Stepper Motor

3.1.9.2 12V 30 RPM Gear Motors

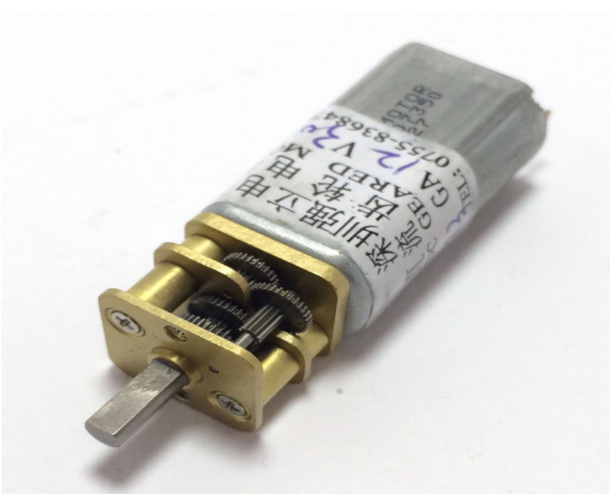


Figure 3.1.9.2.1 - Diagram of the Gear Motor

Gear Motor 12V 30 RPM 13GA	
Voltage	12V
Speed	30 RPM
Motor Size	N20
Weight	11 grams

Table 3.1.9.2.1 - Table of Detailed Specifications of the Gear Motor

3.1.10 Applications of Motors

In the *B-Clean* system, two stepper motors and two gear motors are employed and the combination of these four motors play a significant role in the basket setup module. Two stepper motors are responsible for separating a basket from the stacked baskets. This is achieved by attaching specialized gears to the shaft of the stepper motors. As these two motors move synchronously, it can extract baskets one by one with the customized pitch of the gear, which is tailored to fit in the gaps between two stacked baskets. Once these two motors rotate, it will push the bottom basket away from the stacked baskets.

After a basket has been extracted from the stack, two gear motors will transport the separated basket downward to the sanitization module through a helical system. This execution can be accomplished through integrating gears to the shafts of the gear motors. Whenever the shaft of the motors rotate, it will trigger the helical system to operate as well. The movement of these two motors are concurrent.

3.1.11 Motors Size and Portability

Overall Specifications:

Requirement [REQ-3.1.2.1-H.B] implies the control unit must be compact and portable.

Calculations and Analysis:

Stepper motors and gear motors will be mounted on the same side as the helical gear system. The stepper motor itself has a width of 56.4mm+/-1, which fits perfectly on top of the frame with a width of 70mm+/-0.50. Refer to figure 2.1 for further detailed dimensions of NEMA 23 Stepper Motor used in the *B-Clean* system.

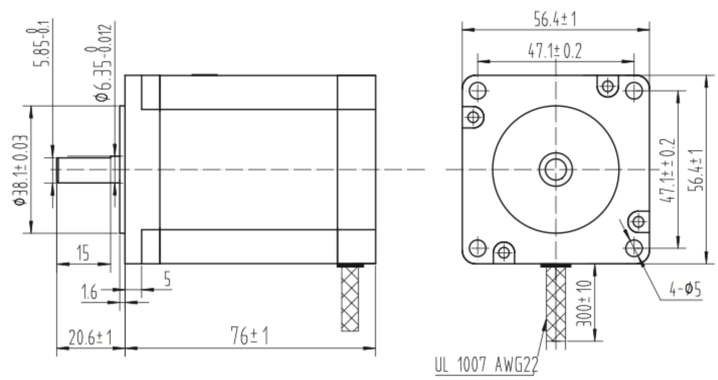
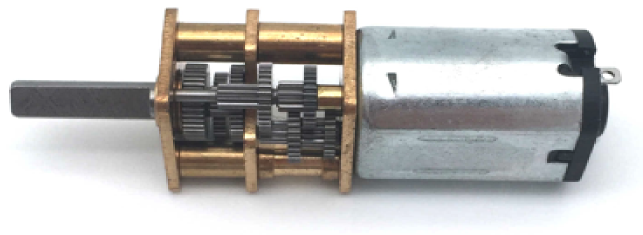


Figure 3.1.11.1 - Dimensions of NEMA 23 Stepper Motor

Similarly, the gear motors have dimensions of 10 x 12 x 36 mm (width x height x length). The dimensions of the gear motors are relatively compact and small; hence, these will qualify for the requirements of having a compact system. Figure 2.2 illustrates in-depth dimensions of the gear motor.



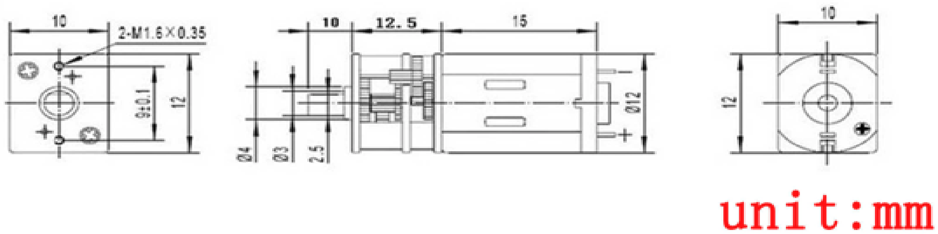


Figure 3.1.11.2 - Dimensions of the Gear Motor

3.1.12 Motors compatibility and system expandability

Overall Specifications:

Requirement [REQ-3.1.1.2-H.A] expects the gears to separate baskets one by one. Requirement [REQ-3.1.3.2-H.B] indicates that the stepper motors for separating a stack of baskets have to withstand the weight of approximately 10-20 stacked baskets.

Calculations and Analysis:

The stepper motor and gear motor are highly expandable and adaptable due to the fact that any customized gear can be constructed through 3D printing and it is effortless to fit any tailored-made gear onto the shaft of both gears.

The application of the stepper motor is solely for separating baskets and it should be capable of handling a stack of 10-20 baskets, which weighs around 15 kilograms. Hence, the stepper motor has to have enough holding torque. The chosen stepper motor has a holding torque of 19 kg-cm, which is more than adequate for the B-Clean system.

3.1.13 Motors DC Power Supply

Overall Specifications:

Requirement [REQ-3.2.3.4-H.F] requires AC-DC converter to power on the stepper motors and gear motors.

Calculation and Analysis:

The output of the conventional power socket in Canada is AC input voltage of 120 V at a frequency of 60 Hz. However, both gear and stepper motors used in the setup module require a constant DC input voltage. Therefore, a chosen AC-DC power supply that converts the AC input voltage to DC 12V output with 3 Amperes can fulfill the requirements.

3.1.14 Motors Design Specification

Design Specification reference	Description	Corresponding Requirement
DES-3.1.21-A	Motors' width must be less than the width of the frame, which is 70mm+/-0.50.	[REQ-3.1.2.1-H.B]
DES-3.1.22-A	Motors must be compatible with the customized gear used in the system to ensure proper operation for separating the basket one by one.	[REQ-3.1.1.2-H.A] [REQ-3.1.3.2-H.B]
DES-3.1.23-A	Motors must have a 12V DC power input.	[REQ-3.2.3.4-H.F]

Table 3.1.14.1 - Summarized Design Specifications for Motors

3.2 Mechanical Design

3.2.1 Outer Frame

The outer frame is designed to be as compact as possible to meet [REQ-3.1.2.1-H.B]. The height of the machine is designed to be around 1m for the user's ease of use. The width and the length are measured to fit gears and baskets in the right position. Figure 3.2.1.1 shows the outer dimension of *B-Clean.* It is designed to stand stable with proper weight distribution to meet [REQ-3.1.3.4-H.A].

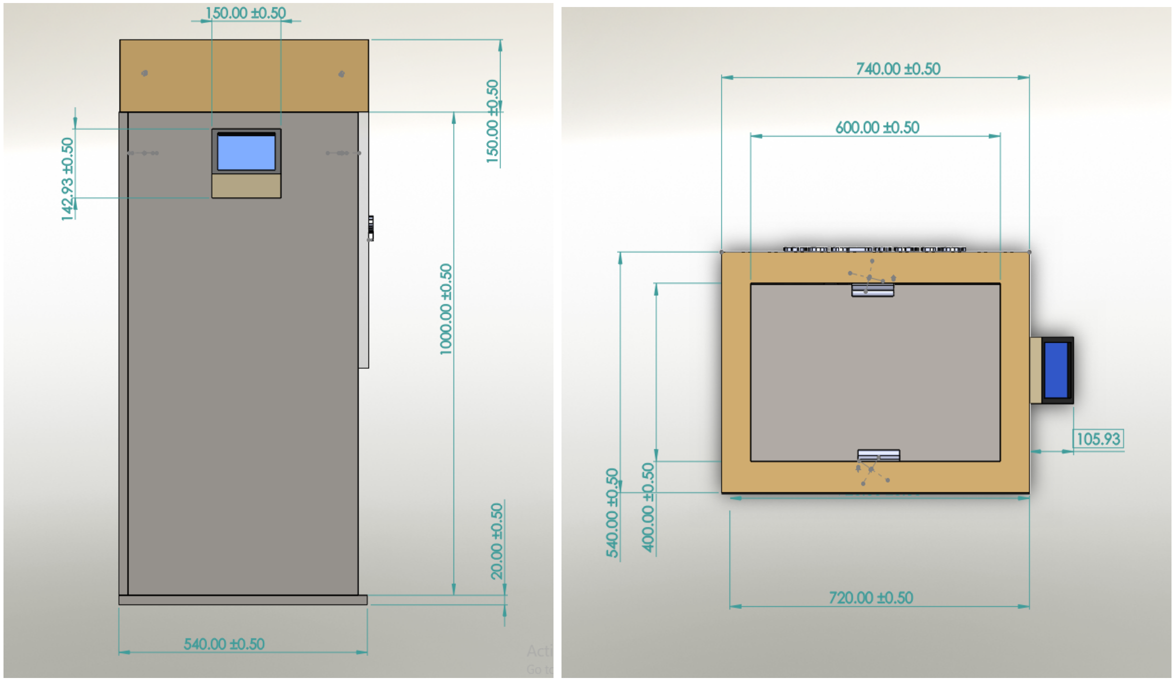


Figure 3.2.1.1 - SaniTech B-Clean with Outer Dimensions in millimeters

3.2.2 3D Printed Gear for separating baskets

Figure 3.2.2.1 illustrates the gear system to separate baskets one by one to meet [REQ-3.1.1.2-H.A]. The gear is designed in Solidworks and the teeth length and the number of teeth are accurately designed to fit in between each basket. Each step turn will release the most bottom basket to feed into the cleaning chamber while holding the remaining upper baskets.

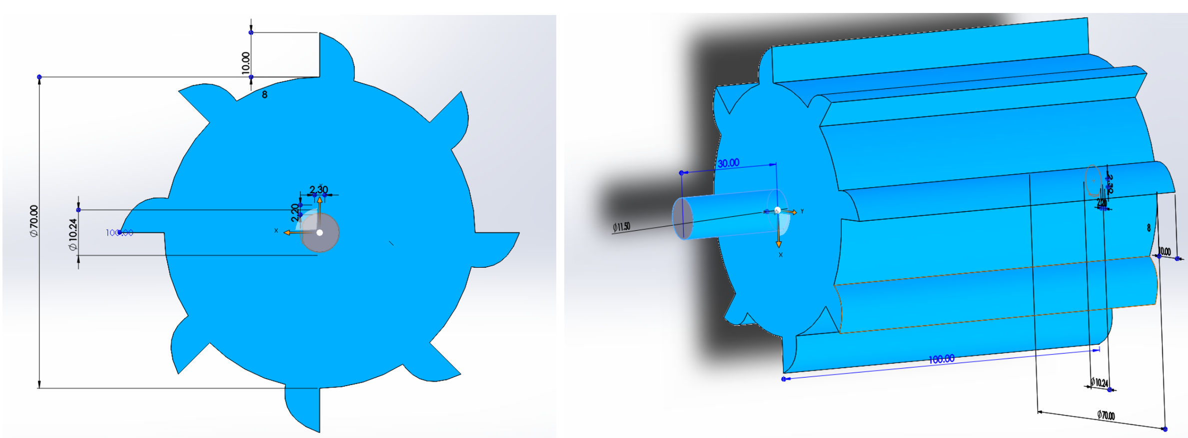


Figure 3.2.2.1 - Top Gears connected to Stepper Motor to separate baskets

3.2.3 3D Printed Helical Gears for Vertical Movements

Figure 3.2.3.1 illustrates the helical gear for the vertical movement of the baskets. This is required to ensure that all the surfaces are exposed to sanitizer and cleaned baskets are stored at the bottom of the chamber. The diameter and the length of the teeth are accurately designed to hold the edge of the basket tight without slippage.

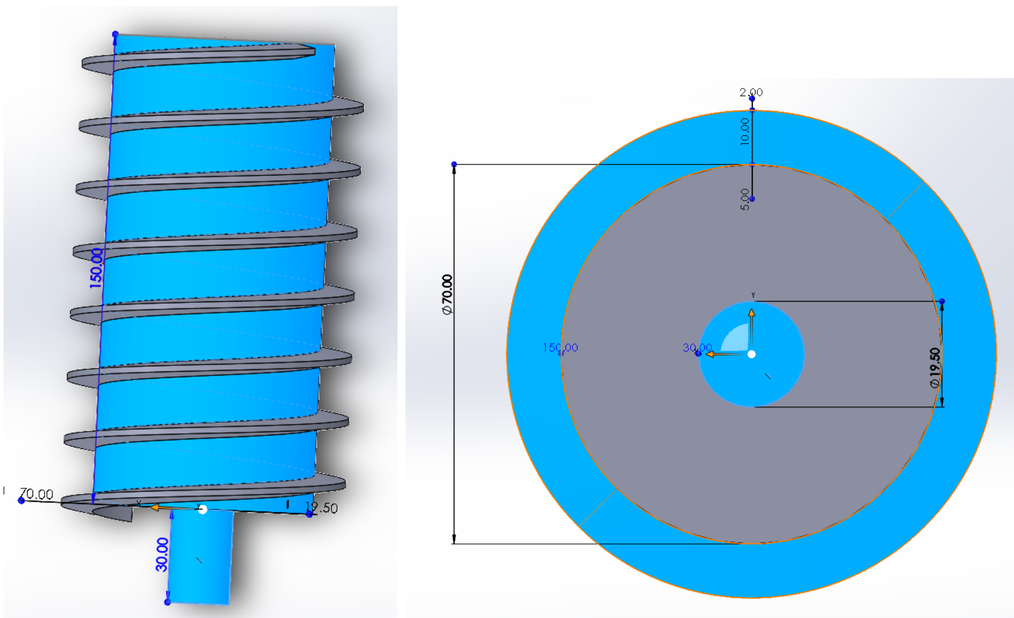


Figure 3.2.3.1 - Helical Gear Connected to Gear Motor for the Vertical Movement of the Baskets

3.2.4 Design Specifications

Design Specification	Description	Corresponding Requirement	Supplement Material
DES-3.2.1-A	Outer Frame	REQ-3.1.1.12-H.A REQ-3.1.2.1-H.B REQ-3.1.3.2-H.B REQ-3.1.3.3-L.F REQ-3.1.3.4-H.A	Figure 3.2.1.1
DES-3.2.2-A	3D Printed Gear for separating baskets	REQ-3.1.1.1-H.B REQ-3.1.1.2-H.A	Figure 3.2.2.1
DES-3.2.3-A	3D Printed Helical Gears for Vertical Movements	REQ-3.1.1.3-H.A REQ-3.1.1.4-H.A	Figure 3.2.3.1

Table 3.2.4.1 - Summarized Design Specifications for Gears and Frame

4 Basket Sanitization Module

4.1 Electrical Design

4.1.1 Sanitization Module workflow overview

The basket sanitization module consists of two subsystems: The spray system and sanitizer level tracking system. As it can be observed in Figure 4.1.1a, both systems communicate mainly with the processing unit which is in our case the raspberry pie 4 already described in section 3.1.5 . Both subsystems will be discussed in great detail in the next sections.

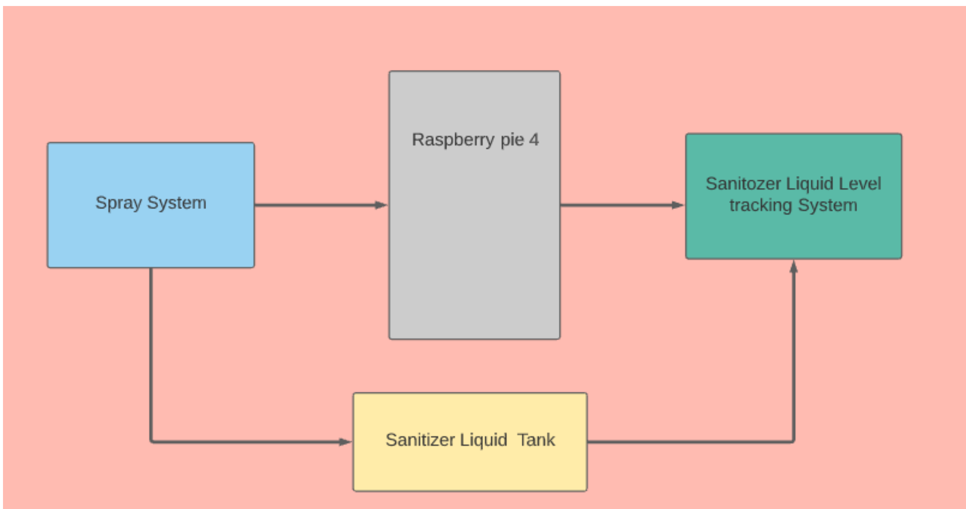


Figure 4.1.1.1 - Sanitization module subsystems

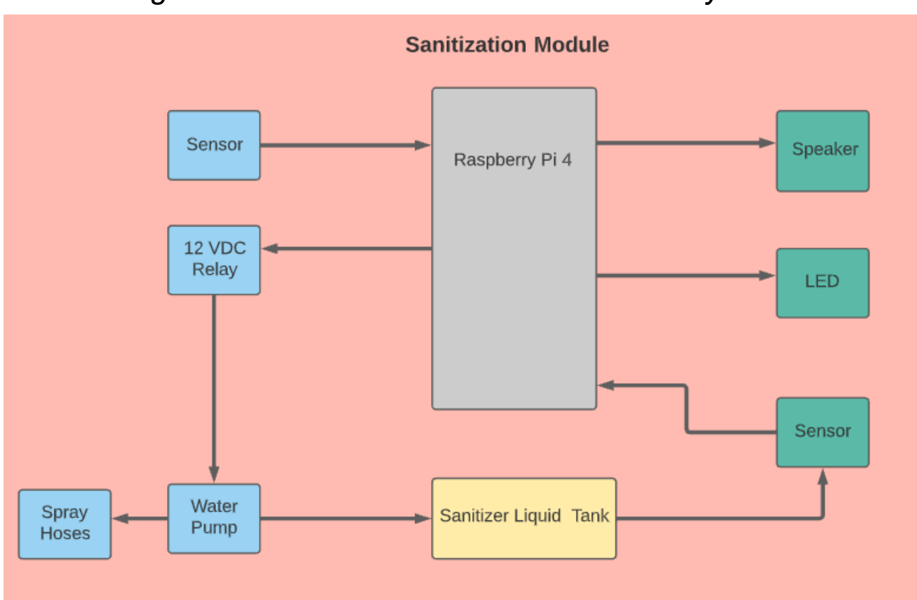


Figure 4.1.1.2 - Sanitization Module Overall workflow diagram

4.1.2 Spray system

4.1.2.1 Electric components specifications

The spray system in the sanitization module has primarily three main electrical components namely, the sensor , the relay and the water pump.

Sensor Specifications

In order to detect whether a basket is within the frame and ready to be sanitized, we need to use a sensor that can detect objects. Below are the specifications of the sensor used for that task:

Design ID	Design Description	Corresponding Requirement	Supplemental material
DES-4.1.2.1.1-A	Sensor shall operate at a range of 4.0V - 5.5V	REQ-3.2.3.2-H.F	Table 4.1.2.1.2
DES-4.1.2.1.2-A	Sensor shall be compatible with the microcontroller	REQ-3.3.1.1-L.F	Table 4.1.2.1.2
DES-4.1.2.1.3-A	Sensor shall detect object within ranges from 5- 30 cm	REQ-3.1.2.4-L.B	Table 4.1.2.1.2
DES-4.1.2.1.4-A	Sensor shall be small in size to fit within the frame	REQ-3.1.3.2-H.B REQ-3.1.3.4-H.A	Figure 4.1.2.1.1 Figure 4.1.2.1.2

Table 4.1.2.1.1 - Spray system sensor Design specification

The Ultrasonic Range Measurement Module HY-SRF05 sensor has been chosen as the sensor to detect baskets inside the sanitization frame. The reason for that choice is that it fulfills all the design specifications listed in Table 4.1.2.1.1 as well as the requirements listed.

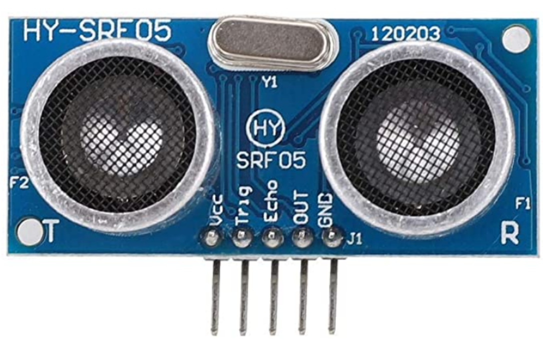


Figure 4.1.2.1.1 - HY-SRF05 Sensor

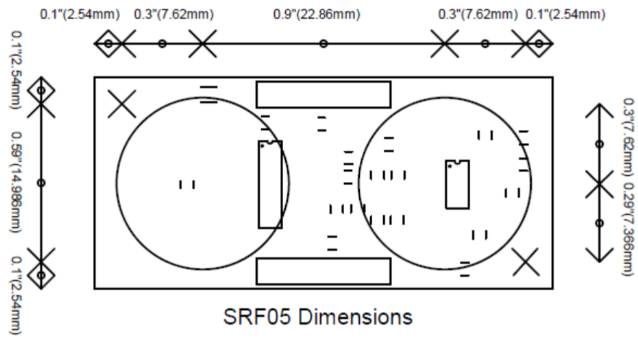


Figure 4.1.2.1.2 - HY-SRF05 Sensor Dimensions

PINS	Vcc Pin: Connect to 5V DC power Trig Pin: Trigger control, signal input Echo Pin: Echo output Out Pin: On-off output GND Pin: Ground connection
Detection distance	2cm-450cm
Measurement Resolution	0.3cm
Measurement Angle	up to 15 deg
Supply Voltage	4.5V to 5.5V
Supply Current	10 to 40mA
Static current	less than 2mA

Table 4.1.2.1.2 - HY-SRF05 Sensor detailed specification

Relay Specification

The role of the relay is to connect with the microcontroller in order to trigger the water pump as the sensor detects a basket in the sanitization frame. Below are the specifications of the relay used for that task:

Design ID	Design specification	Corresponding requirement	Supplemental material
DES-4.1.2.1.5-A	The relay board shall be compatible with the microcontroller	N/A	Table 4.1.2.1.4
DES-4.1.2.1.6-A	The relay board shall operate between 5V- 12V DC	REQ-3.2.3.2-L.F	Table 4.1.2.1.4

DES-4.1.2.1.7-A	The relay board shall be able to trigger the water pump	N/A	Table 4.1.2.1.4

Table 4.1.2.1.3 - Relay Design Specification

Given the design specifications listed on the table above , our team decided to choose the 12V 1 Channel Relay Module as it can perform the task efficiently and it fulfils our design specification and requirement. In addition to that, it is cost effective and readily available locally compared to the other options available to us to control the water pump.

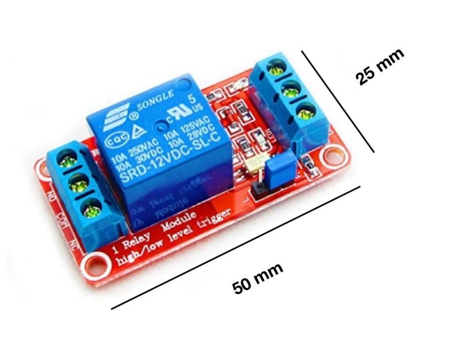


Figure 4.1.2.1.3 - 12V 1 Channel Relay Module

Size (L x W x H)	50mm x 25mm x 18.5mm
Input	VCC: Relay power supply + GND: Relay power supply - IN: Signal trigger input (High or Low Level Signal)
Output	NO: Relay normally open. COM: Relay Common Pin NC: Relay normally closed.

Table 4.1.2.1.4 - Relay module detailed specification

Water Pump Specification

For the purpose of spraying sanitizer liquid inside the sanitization frame, we use a water pump to drive the sanitizer liquid to the spray nozzle in order to sanitize the baskets efficiently .Below are the specifications of the water pump used for that task:

Design ID	Design specification	Corresponding requirement	Supplemental material
DES-4.1.2.1.8-A	The water pump shall operate on voltage between 6V-12V	REQ-3.2.3.2-L.F	Table 4.1.2.1.6

DES-4.1.2.1.9-A	The water pump shall have high pressure operating between range between 6L/min-10 L/min	N/A	Table 4.1.2.1.6
DES-4.1.2.1.10-A	The water pump shall be submersible	N/A	Table 4.1.2.1.6

Table 4.1.2.1.5 - Water pump Design specification

Figure 4.1.2.1.4 - JT-500 DC 12V Water Pump

Current	1.2 A
Working Voltage	6-12 V DC
Size	108 mm (h) x 40 mm (w)
Water Outlet dimensions	21 mm (h) x 10 mm (d)
Maximum Flow rate	10 L/min
Submersible Water Pump	Yes

Table 4.1.2.1.6 - JT-500 specifications

4.1.2.2 Electric circuit

In order to detect baskets within the sanitizing frame , the Ultrasonic Range Measurement Module HY-SRF05 sensor will be connected to the Raspberry pi 4 like we can observe in Figure 4.1.2.2.1 . The Vcc pin of the sensor is connected to the 5V pin of the Raspberry pi, the Trig pin of the sensor is connected to GPIO#4, The Echo pin of the sensor is connected to GPIO#17 and the GND pin of the sensor is connected to Ground in the Raspberry pi.

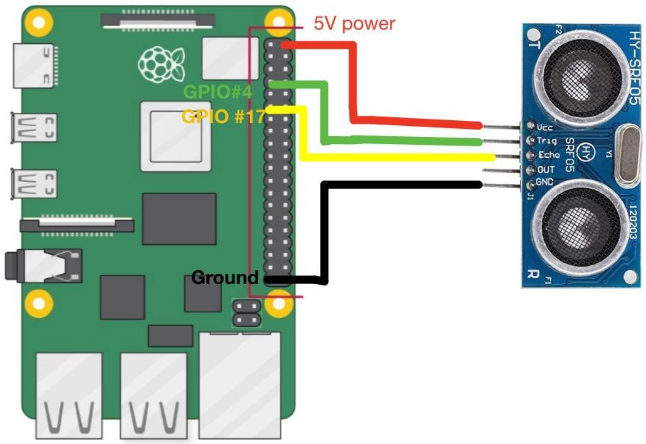


Figure 4.1.2.2.1 : Spray system - sensor- microcontroller circuit connection

In order to control and trigger the water pump, the relay board is connected to the water pump and the Raspberry pi as we can observe in Figure 4.1.2.2.2 below:

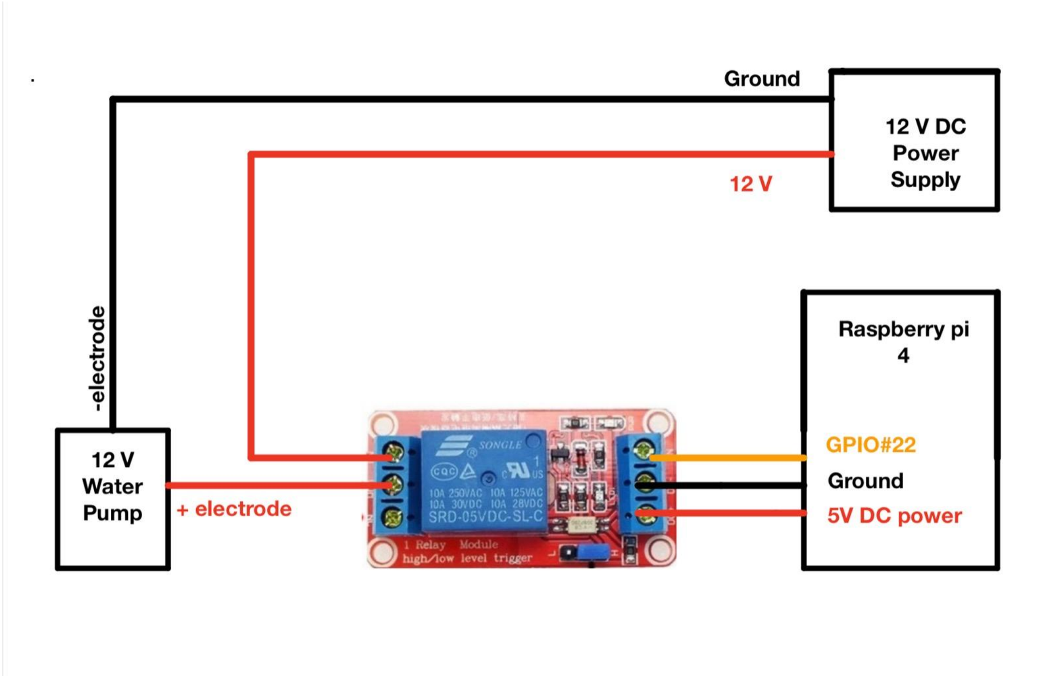


Figure 4.1.2.2.2 - Water pump-Relay board- microcontroller circuit connection

4.1.3 Sanitizer level tracking system

4.1.3.1 Electric components specification

Sensor Specification

The Sanitizer level tracking system has one main electric component which is a sensor. The role of the sensor is to monitor the level of sanitizer liquid remaining in the sanitizer tank before it reaches critical levels. Below are the specifications of the sensor used for that task:

Design ID	Design specification	Corresponding Requirement	Supplemental material
DES-4.1.3.1.1-A	The sensor shall be compatible with the microcontroller	N/A	Table 4.1.2.1.2
DES-4.1.3.1.2-A	The sensor board shall operate between 4V- 5V DC	REQ-3.2.3.2-L.F	Table 4.1.2.1.2
DES-4.1.3.1.3-A	The sensor shall be able to measure fluid level from a given initial position	N/A	Table 4.1.2.1.2

Table 4.1.3.1.1 - Sanitizer Liquid tracking system sensor design specification

Given the table above, our team decided to use the Ultrasonic Range Measurement Module HY-SRF05 sensor to monitor the sanitizer liquid level in the sanitizer tank because it performs the task efficiently and is low cost compared to other sensors in the market.

The dimensions as well as the specification of the HY-SRF05 sensor is given in the section 4.1.2.1 as Figure 4.1.2.1.1 ,Figure 4.1.2.1.2 and Table 4.1.2.1.2

Speaker Specification

As part of the sanitizer liquid tracking system, a speaker is used in order to alert the user when the sanitizer liquid reaches critical levels . Below are the specifications of the speaker used for that task:

Design ID	Design specification	Corresponding requirement	Supplemental material
DES-4.1.3.1.4-A	The speaker shall operate between 3.3- 5 V for low power consumption	REQ-3.2.3.2-L.F	Table 4.1.3.1.3
DES-4.1.3.1.5-A	The speaker shall have SNR at least 60dB	N/A	Table 4.1.3.1.3
DES-4.1.3.1.6-A	The speaker shall be compatible with the microcontroller	N/A	Table 4.1.3.1.3

Table 4.1.3.1.2 - Speaker design specification

Considering the table above, our team decided to choose the GLOGLOW Piezoelectric Model 2312 speaker because it can perform the task efficiently and also fulfils our design specification for the speaker. In addition to that the connector type is lead connector type which makes the connection to the microcontroller effortless. Also, it is small in size, which removes bulkiness from the overall system .

Figure 4.1.3.1.1 - GLOGLOW Piezoelectric Model 2312 speaker

Max current	10 mA
Rated voltage	12 V DC
Working Voltage	3V - 24V
Connector type	leads
Sound pressure level	85 dB to 95 dB
Weight	3 g
Size	12 mm (h) x 23 mm (d)
Cable length	100 mm

Table 4.1.3.1.3 - GLOGLOW Piezoelectric Model 2312 speaker specifications

LED Specification

In order to notify the employees when the sanitizer liquid level has reached critical levels, LED lights are used. Below are the specifications of the LED used for that task:

Design ID	Design specification	Corresponding requirement	Supplemental material

DES-4.1.3.1.7-A	The LED light shall be able to emit RED , GREEN and BLUE light	N/A	Figure 4.1.3.1.3
DES-4.1.3.1.8-A	The LED shall have uniform light output	N/A	Table 4.1.3.1.5
DES-4.1.3.1.9-A	The LED shall have low power consumption	N/A	Table 4.1.3.1.5
DES-4.1.3.1.10-A	The LED shall be durable	N/A	N/A

Table 4.1.3.1.4 - LED design specification

Based on the design specification, listed above, our team decided to choose the LED 5MM RGB 4-PIN C-CATHODE because it can efficiently perform the given task in addition to fulfilling the design specifications.. It is low cost and readily available locally.

Figure 4.1.3.1.2 - LED 5MM RGB 4-PIN C-CATHODE

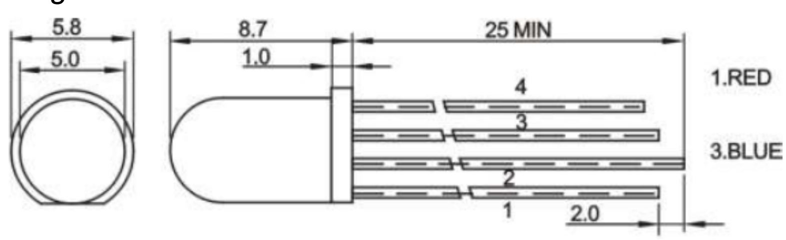


Figure 4.1.3.1.3 - LED 5MM RGB 4-PIN C-CATHODE dimensions

Forward Current	30 mA
Reverse Voltage	5 V
Power Dissipation	140 mW

Table 4.1.3.1.5 - LED 5MM RGB 4-PIN C-CATHODE specification

4.1.3.2 Electric circuit

For the purpose of monitoring the level of sanitizer liquid in the tank , the circuit below is needed to accomplish this task. The sensor is placed on the lid of the tank and reads how far is the sanitizer liquid from that position. The distance is communicated to the raspberry pi and when the threshold level is reached, the user is alerted through LED lights and a speaker as part of the User Interface. In Figure 4.1.3.2.1, the VCC pin of the sensor is connected to the 5V DC power pin of the raspberry pi, the Trig pin of the sensor is connected to GPIO#23, the Echo pin of the sensor is connected to GPIO #24 and the ground pin of the sensor is connected to the ground pin on the raspberry pi.

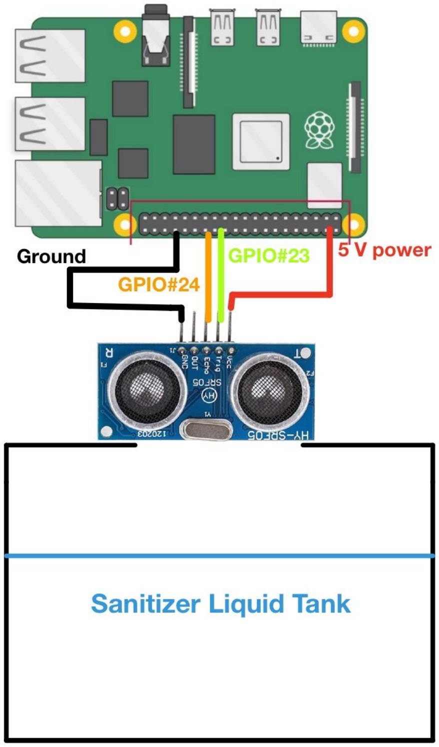


Figure 4.1.3.2.1 - Sanitizer Liquid tracking system sensor - microcontroller circuit connection

For the purpose of alerting the user when sanitizer liquid is at critical level , LED light as well as speaker is used to get the employees attention. Below is the schematique of the circuit configuration of the LED and the speaker :

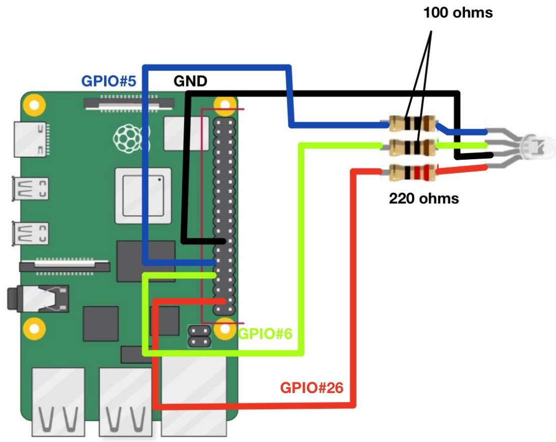


Figure 4.1.3.2.2 - LED-microcontroller circuit connection

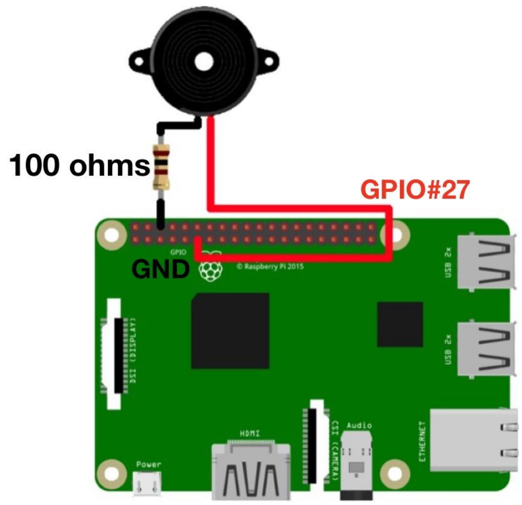


Figure 4.1.3.2.3 - Speaker-microcontroller circuit connection

4.2 Mechanical Design

4.2.1 Sanitizer Distribution Design

Overall Specifications

[REQ-3.1.1.3-H.A], [REQ-3.1.1.4-H.A], [REQ-3.1.1.9-H.B], [REQ-3.1.1.10-L.B] imply that the Sanitization System must be able to sanitize multiple basket sizes and handles. [REQ-3.1.2.8-H.B], [REQ-3.1.2.6-H.B] require that the sanitization system must be able to process baskets quickly and must be convenient to refill sanitizer when necessary. [REQ-3.1.4.3-L.F], [REQ-3.1.3.1-H.B], [REQ-3.4.1.1-H.B], [REQ-3.4.1.2-H.A], [REQ-3.4.1.3-H.A] and [REQ-3.4.1.4-L.B] imply the system must be durable for a significant time frame, withstand sanitizer liquid and eco friendly when possible. [REQ-3.1.3.3-L.F] requires the whole product to be cost effective by having a Maximum production cost of CAD 1000.

Calculations and Analysis

Research on the topic of sanitizing surfaces with sanitizer liquid showed that it was very helpful in disrupting the transfer of viruses [2]. As a lot of the product requirements require a system that can be adaptable and clean every part of the basket, two systems were considered to sanitize the entire basket. Table B.3.2.1.1 in Appendix B talks about a submersible system where we submerged multiple baskets to sanitize them at once and a second solution that sprays each basket with sanitizer.

Considering the pros and cons of both a submersible system and a spray system, B-Clean will choose to implement a spray system as it better satisfies requirements. As can be seen in Figure 4.2.1.1, the placement of the nozzles and tubes will apply the sanitizers to all sides of the basket, even when multiple basket or handle sizes are present. By setting up the spray system around the Basket Setup Module, the system will fulfill the timing requirement by processing each basket faster and not spending time to clean a single basket in two different modules.

There are several market options to implement a spray system. As B-Clean has a maximum cost requirement, market research has focussed on relatively cheaper alternatives to the Sanitization Spray System. It is also necessary to choose an option that can give us sufficient tubing to keep the liquid motor as far as possible so that the operator can easily refill the sanitizer tank. After in-depth examination, option 2 in Table B.3.2.1.2 is the most cost effective.

Brand	MSDADA
Type	flexible plastic tube
Length	15.24 meters (50ft)
Nozzles	20, not fixed, (other extra nozzles)
Diameter	Outer: 7mm, Inner: 4mm
Cost	CAD 28.99
Link	Amazon

Table 4.2.1.1 - Chosen Spray System Specifications

As can be seen in table 4.2.1.1, with over 50 ft of wire, the water pump will be placed close to the surface of the structure so that there will be easy access to refill the sanitizer solution. With more than 20 spray nozzles there will be more than enough spare parts. Additionally, the spray claims to save 70% of water usage compared to other nozzle systems which satisfies being eco friendly by using less sanitizer while still completing the entire cleaning process. It also claims to be made from material that is UV resistant and chemical resistant which is important to satisfy the durability requirements.

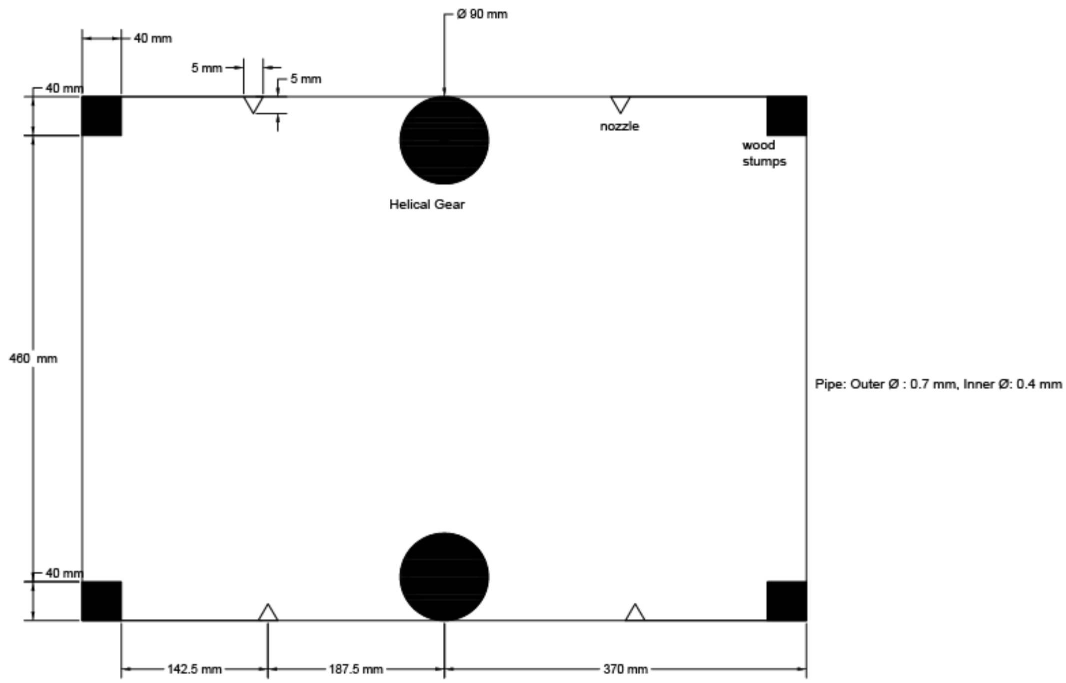


Figure 4.2.1.1 - Structure or the tube and nozzles in the Sanitization Module

As can be seen in Figure 4.2.1.1, there will be 4 spray nozzles connected to a plastic tube that will wrap around the support wooden stumps. The tube will eventually connect to the water pump supplying the sanitizer solution to the tubes. The nozzles will be evenly placed to reach most of the basket, they will be placed 182.5 mm away from the edge on each side. The nozzles are less than 5x5mm and hence will not block the basket movement process from working. The total tubing needed for one layer of the figure is approximately 2560 mm + 500 mm for connecting to the sanitizer source pump. To add additional protection by spraying more sanitizer, additional layers of Figure 4.2.1.1 can be added as the design structure of the tube takes minimal space.

4.2.2 Sanitizer Liquid

Overall Specifications

[REQ-3.1.5.1-H.A], [REQ-3.1.5.2-H.A], [REQ-3.4.1.7-H.F], [REQ-3.5.1.4-H.F] imply that the sanitizing liquid solution must be human friendly and not have severe health effects on humans while effectively killing viruses and germs. [REQ-3.4.1.5-H.F], [REQ-3.4.1.6-H.F] and [REQ-3.1.5.6-L.B] require that the choice of sanitizer must be environmentally friendly. REQ-3.1.5.3-H.A, [REQ-3.1.5.4-H.A] and [REQ-3.1.5.5-H.A], [REQ-3.1.5.7-L.B] imply that the sanitizer choice must be reasonably priced, easily accessible to purchase and must meet industrial standards. [REQ-3.1.5.8-H.A], [REQ-3.1.5.9-H.A], [REQ-3.1.5.10-H.A] and [REQ-3.5.1.5-H.A] imply that the sanitizer must have low viscosity to be able to be sprayed easily, not damage baskets and internal structure of the system.

Calculations and Analysis

There are several chemical solutions that can be used to kill germs and viruses as can be seen in Table B.3.2.2.1 found in Appendix B. Most are very high quality and are used frequently in Medical environments. All of them have a pattern of gaining better efficacy when the solution is left on the surface for extended periods of time. As B-Clean needs a quick solution and does not need an extremely high quality solution to use for medical purposes, it is reasonable to choose a lesser quality and a quicker solution like alcohol.

Alcohol is widely used currently to prevent the transmission of the Coronavirus and in the past for flu seasons. Long standing research recommends using concentrations of between 60-90% Isopropyl Alcohol to have the best effect while sanitizing surfaces [3]. Hence, years of confidence in alcohol based sanitizers which makes it more widely available in the market satisfies the requirements that it should kill germs while being safe for humans to handle. According to research [3], exposure to alcohol for more than 10 seconds (with 60-90% concentration) kills a significant amount of germs. As our basket vertical movement module expects the basket to be moving through the module for ~15-20 seconds, the sanitizer will have time to kill germs and evaporate too. While most chemical solutions will have some adverse effect over time with plastics, alcohol may have the least amount of effect overtime as it will be exposed as a mist for less than 15 seconds by the system [4]. This satisfies ensuring the baskets don't get damaged in the process of being cleaned. Currently to kill germs and viruses effectively, not a lot of environmentally friendly solutions are tested well enough to be approved by EPA or Health Canada. Hence the most environmentally friendly solution that has approval from health authorities are alcohol based solutions [5].

While online and retail stores frequently sell alcohol based sanitizers at a competitive price, it can be seen in Table B.3.2.2.2 from Appendix B that the market is competitive in the industrial sector too. All of the options listed satisfy availability requirements and can be widely bought all over Canada. While not all may have Health Canada designation, Health Canada has generally approved EPA based products. Considering this to satisfy the regulatory requirement, option 1 (Alpet D2) at the sale price is the best option for the system. While the user will not need 50 gallon of sanitizer, it is a relatively cheap price considering other options offer around 40% the price for just 1 gallon. As the option has low viscosity, it will also satisfy the requirement to be able to spray like water. Specifications for the chosen solution is described in Table 4.2.2.1

Brand	ALPET D2
Type	Liquid alcohol based sanitizer
Viscosity	Low
Ratings	EPA, CFIA
Quantity	189.27 litres (50 Gallons)
Cost	CAD 299.99 (Sale), CAD 1558.18 (Regular)
Link	Grainger

Table 4.2.2.1 - Chosen Alcohol Solution Specifications

4.2.3 Design Specifications

Design Specification	Description	Corresponding Requirement	Supplement Material
DES-4.2.1.10-A	Sanitization module is implemented with a spray system.	REQ-3.1.1.3-H.A REQ-3.1.1.4-H.A REQ-3.1.1.9-H.B REQ-3.1.1.10-L.B REQ-3.1.2.8-H.B REQ-3.1.2.6-H.B	-N/A-
DES-4.2.1.2-A	The spray system is implemented with the MSDADA spray system.	REQ-3.1.4.3-L.F REQ-3.1.3.1-H.B REQ-3.4.1.1-H.B REQ-3.4.1.2-H.A REQ-3.4.1.3-H.A REQ-3.4.1.4-L.B REQ-3.1.3.3-L.F	Table 4.2.1.1
DES-4.2.1.3-A	The spray system structure will follow the structure found in Figure 4.2.1.1.	REQ-3.1.2.8-H.B REQ-3.1.1.4-H.A	Figure 4.2.1.1

		REQ-3.1.1.3-H.A REQ-3.1.1.9-H.B REQ-3.1.1.10-L.B	
DES-4.2.2.1-B	The sanitizer solution selected for the system is an alcohol based solution.	REQ-3.1.5.1-H.A REQ-3.1.5.2-H.A REQ-3.4.1.7-H.F REQ-3.5.1.4-H.F	-N/A-
DES-4.2.2.2-F	The alcohol sanitizer recommended for use is ALPET D2.	REQ-3.1.5.3-H.A REQ-3.1.5.4-H.A REQ-3.1.5.5-H.A REQ-3.1.5.7-L.B REQ-3.1.5.8-H.A REQ-3.1.5.9-H.A REQ-3.1.5.10-H.A REQ-3.5.1.5-H.A	Table 4.2.2.1

Table 4.2.3.1 - Mechanical Design Specifications for Basket Sanitization Module

5 Conclusion

This document clearly outlines all the general system design specifications of B-clean. The paper discussed different phase deliverables in perspectives of Proof-of-Concept, Prototype and Final Product. The major feature to be implemented is automating the separation of baskets and cleaning them in a pipeline form to save labour, time and efficiency; the control and data processing unit are also in parallel development.

The detailed design requirements and process are presented for software and hardware components. This document also provided a User Interface and Appearance in the Appendix section. Moreover we also provided a test plan for future records.

B-clean consists of helical gears, different sensors including object sensor, spray sensor and electrical parts as main parts as well as LCD monitor user interface details as side parts.

The Proof-of-Concept Deliverables will be presented in mid April. As part of SaniTech our intention is to provide a reliable, safe and eco-friendly product to our consumers through innovative and efficient design requirements and features, as outlined in this document.

6 References

[1] “Power Supply,” *Power Supply - Raspberry Pi Documentation*. [Online]. Available: https://www.raspberrypi.org/documentation/hardware/raspberrypi/power/README.md#:~:text=The%20power%20supply%20requirements%20differ,uses%20a%20USB%2DC%20connector. [Accessed: 23-Mar-2021].

[2] Minnesota Department of Health, *Evaluation of Cleaners, Sanitizers, and Disinfectants for Surfaces*, Dec-2017. [Online]. Available: https://www.health.state.mn.us/communities/environment/risk/docs/guidance/cleaners.pdf. [Accessed: Mar-2021].

[3] CDC, “Chemical Disinfectants,” *Centers for Disease Control and Prevention*, 18-Sep-2016. [Online]. Available: https://www.cdc.gov/infectioncontrol/guidelines/disinfection/disinfection-methods/chemical.html. [Accessed: 25-Mar-2021].

[4] Iconoclast, B. Norris, and Leonardo, “Will alcohol or soap damage plastic or rubber?,” *Chemistry Stack Exchange*, 01-Jul-1961. [Online]. Available: https://chemistry.stackexchange.com/questions/1092/will-alcohol-or-soap-damage-plastic-or-rubber. [Accessed: 25-Mar-2021].

[5] C. Rosenbloom, “'Green' household cleaners and coronavirus: What you need to know,” *The Washington Post*, 11-Aug-2020. [Online]. Available: https://www.washingtonpost.com/lifestyle/home/green-household-cleaners-and-coronavirus-what-you-need-to-know/2020/08/10/e49a24b6-d28c-11ea-8d32-1ebf4e9d8e0d_story.html. [Accessed: 25-Mar-2021].

[6] Raspberry Pi, “Raspberry Pi 4 Model B specifications,” *Raspberry Pi*. [Online]. Available: https://www.raspberrypi.org/products/raspberry-pi-4-model-b/specifications/. [Accessed: 25-Mar-2021].

[7] *C++ vs Java vs Python?* [Online]. Available: https://www.tutorialspoint.com/cplusplus-vs-java-vs-python. [Accessed: 26-Mar-2021].

[8] D. Norman, The Design of Everyday Things: Revised and Expanded Edition, New York: Basic Books, 2013.

[9] “ISO 9241-11:2018,” *ISO*, 04-Apr-2018. [Online]. Available: https://www.iso.org/standard/63500.html. [Accessed: 19-Mar-2021].

[10] “CAN/CSA-C22.2 No. 60529:05 (R2010),” *Standards Council of Canada - Conseil canadien des normes*. [Online]. Available: https://www.scc.ca/en/standardsdb/standards/22548. [Accessed: 19-Mar-2021].

[11] “ISO 14040:2006,” ISO, 16-Sep-2016. [Online]. Available: https://www.iso.org/standard/37456.html. [Accessed: 19-Mar-2021].

[12] CSA Group, “CAN/CSA-C22.2 No. 61508-1:17”, CSA, 2017, [Online], https://www.scc.ca/en/standardsdb/standards/28870

Appendix A - User Interface and Appearance Design

A.1 Introduction

B-Clean is an automated basket sanitizing solution for grocery stores that are looking to provide protection for customers while increasing efficiency of their workforce. As the system will be used by grocery store employees, grocery store customers and SaniTech Maintenance employees; it is necessary to design the user interface and experience in detail. *B-Clean* applies Don Norman's "Seven Elements of UI Interaction" to ensure that the design is usable and requires minimal training for new users. The user interface design of the system will rely on familiarity and appearance to ensure ease of use.

A.1.1 Purpose

The purpose of this document is to demonstrate the user interface design of the hardware and software components for *B-Clean*. It will offer an in-depth justification for the design choices of the system by referring to Norman's 7 principles. To ensure that design choices are not just theory but are practically beneficial, strong user feedback tests are designed to get feedback from potential users.

A.1.2 Scope

The User Interface and Appearance Design Document will cover user analysis, technical analysis, engineering standards, analytical usability testing, and empirical usability testing.

A.2 Graphical Presentation

B-Clean is an automated basket sanitizer and one of the major considerations for the users of our product is to know if the sanitizer liquid has reached critical levels. For maintenance purposes it is important to be able to easily interact with the system to get an idea of how long it can run before the sanitizer liquid gets replaced. Additionally it is important to know which phase of the cleaning process the device is operating on and what is left to completion of the cleaning process. It is equally important for the user to know whether the system is running or at rest. Since our system is an automated system, it does not take much input from the user, but rather it takes the task at the end and informs the user of the process so that the baskets can be retrieved. For familiarity and ease of use, our user interface will be built on models similar to self autonomous and automated systems such as wahermashines and dishwashers.

In this section, there are four different perspectives of the visualized 3D-modelling diagrams for the SaniTech *B-Clean* system. In Figure A.1, it illustrates the general view of the SaniTech *B-Clean* system. Different views of the SaniTech *B-Clean* system are shown in Figure A.2 - A.4 along with the dimensions. All the dimensions present in the diagrams are in millimeters.

In addition, Figure A.5 - A.8 displays the GUI that will be displayed to the users via an LCD screen. Figure A.5, shows the homepage GUI of the SaniTech *B-Clean* system and only the system operation, sanitizer level, and basket cleaned information will be displayed. The display GUI also offers an additional Maintenance interface as shown in Figure A.6 for users and maintainers to check on the logs and the status of the system.

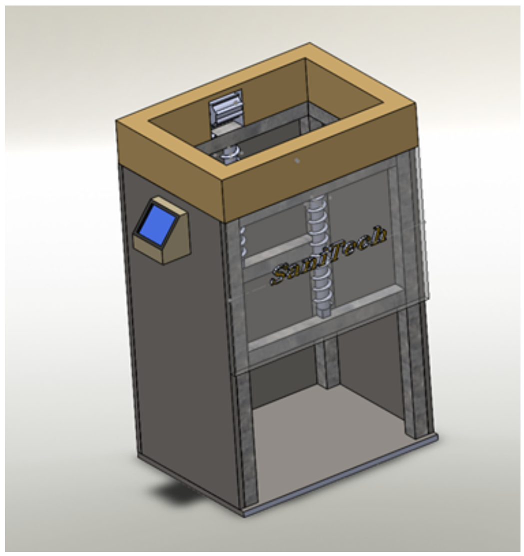


Figure A.1 - 3D Modelling of SaniTech B-Clean

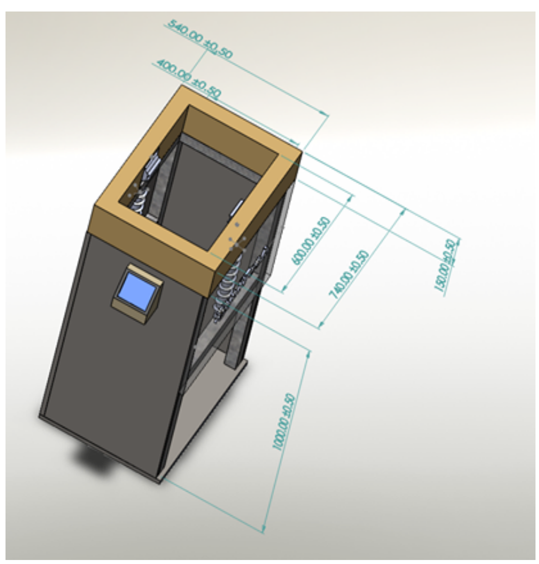


Figure A.2 - 3D Modelling of SaniTech B-Clean with Dimensions in millimeters

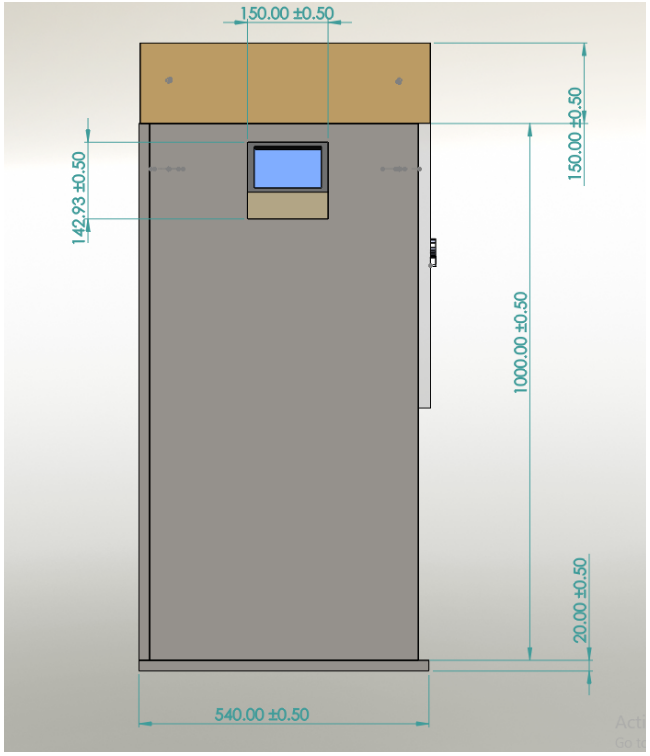


Figure A.3- Side View of SaniTech B-Clean with Dimensions in millimeters

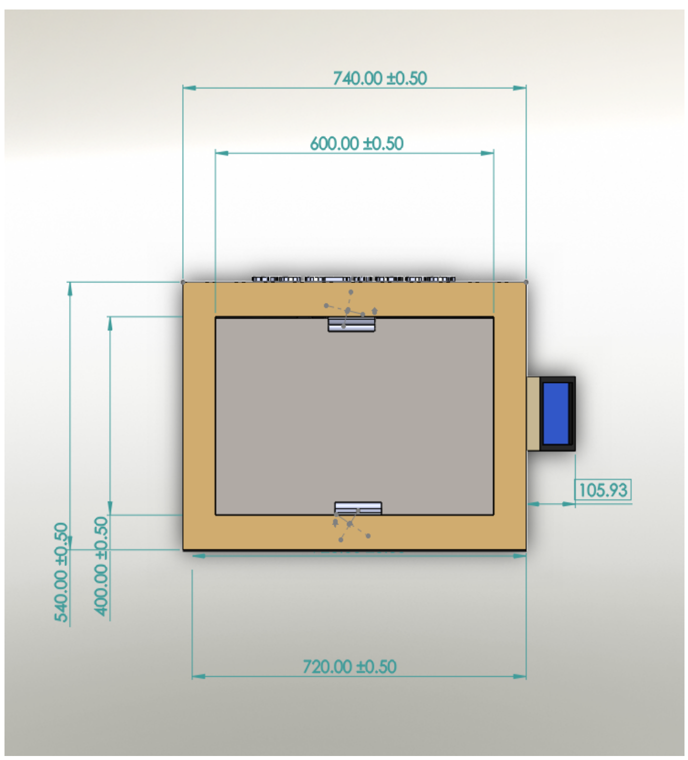


Figure A.4- Top View of SaniTech B-Clean with Dimensions in millimeters

The LCD display spray monitor details:

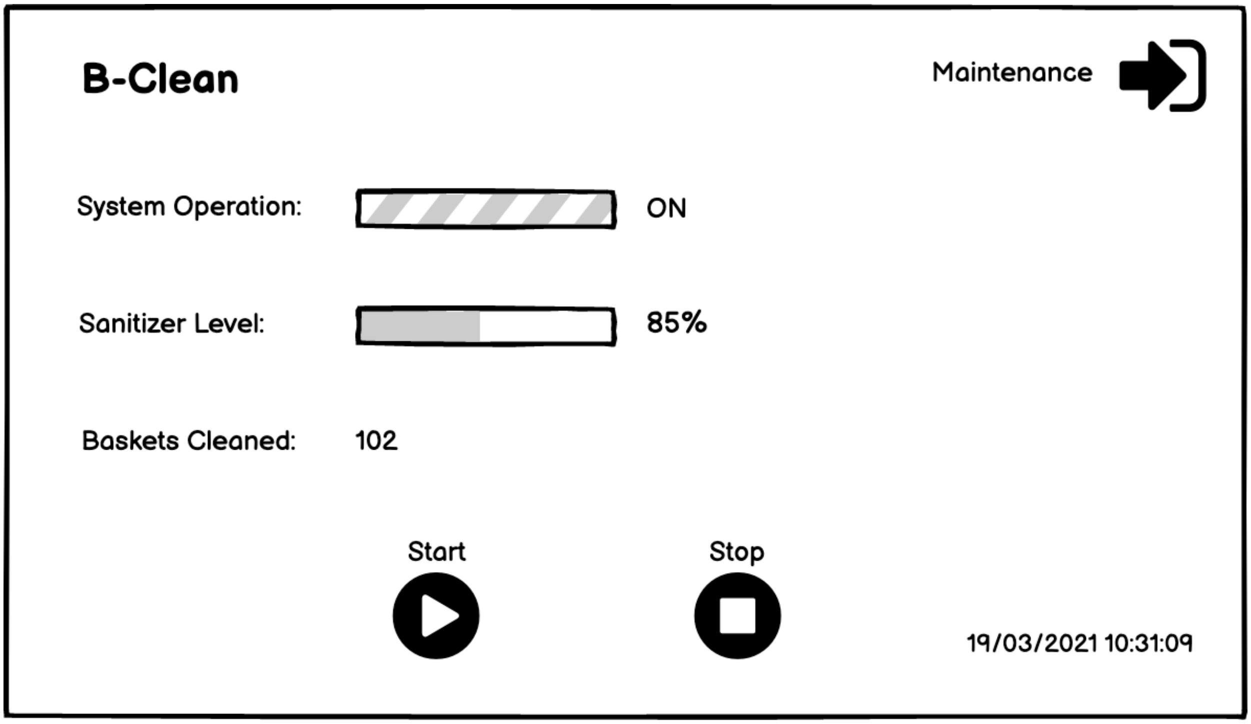


Figure A.5- Operator Home Page

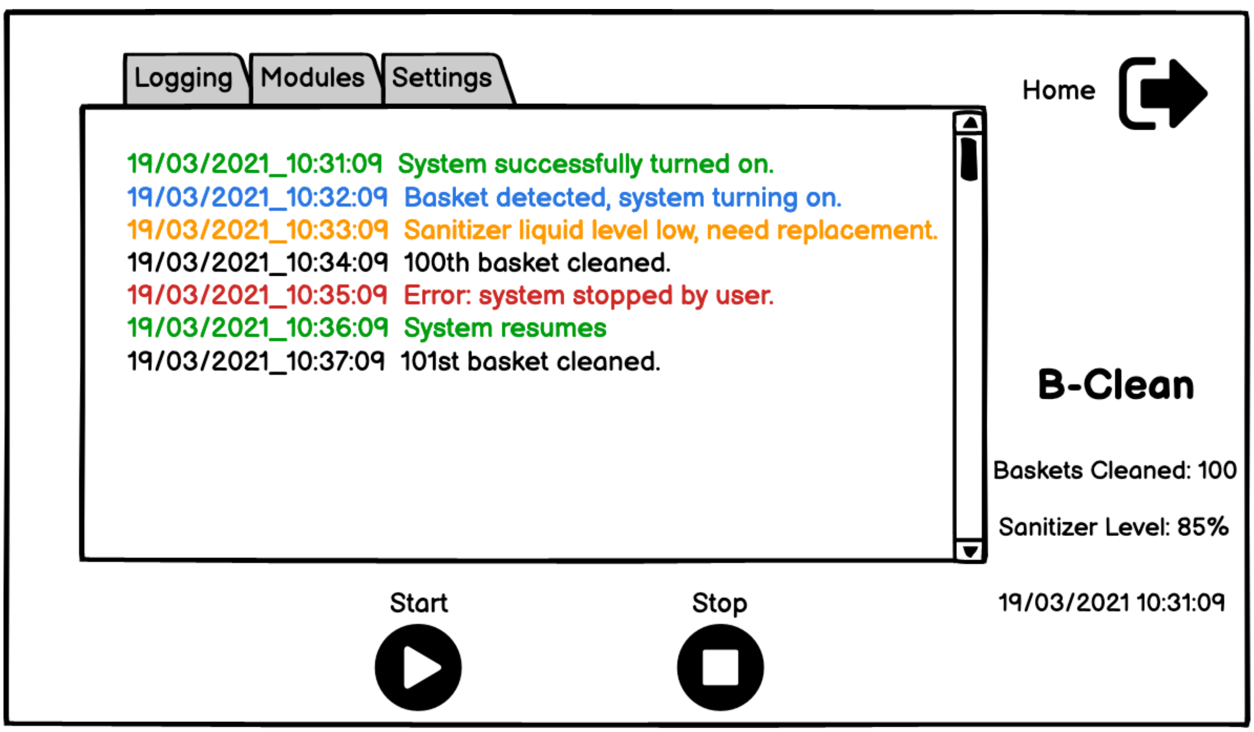


Figure A.6- Maintenance Logging Page

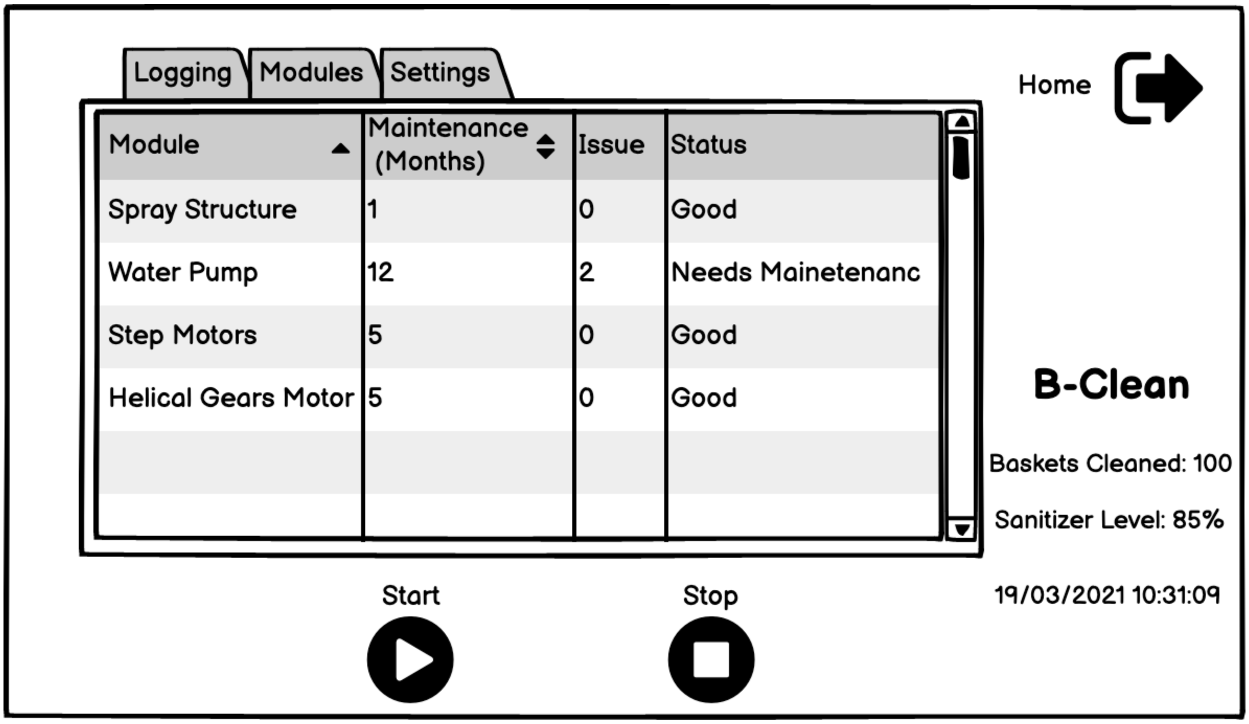


Figure A.7- Maintenance Modules Page

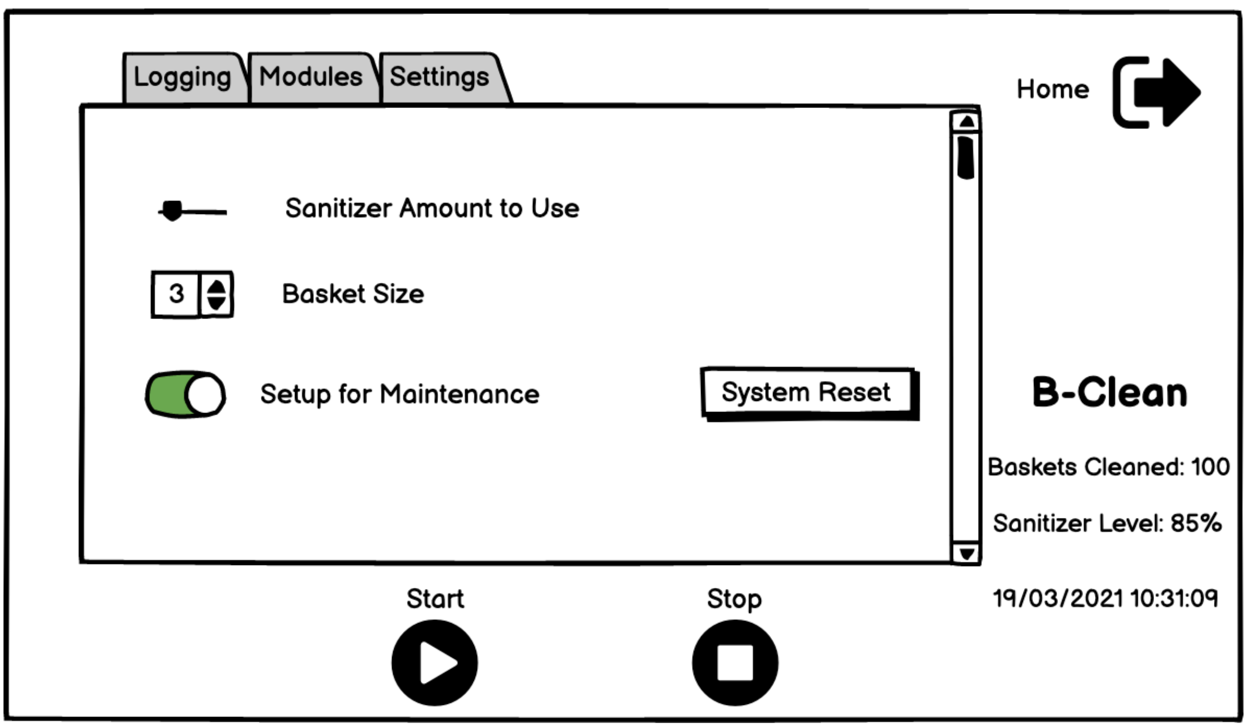


Figure A.8- Maintenance Settings Page

A.3 User Analysis

The SaniTech *B-Clean* system is targeting supermarkets that provide conventional baskets to their customers, particularly for stores that have a significant volume of customers visiting the store on a daily basis. It can replace the repetitive task that employees are required to execute every day that they weren't required to practice before the pandemic, which saves a significant amount of time for the employees. Furthermore, it can ensure that all the surface areas of the basket are sanitized thoroughly which eliminates any doubt from customers regarding the accuracy of sanitization by employees.

The SaniTech *B-Clean* is an user-friendly automated sanitizing system that can be operated by any user effortlessly and it does not require any background knowledge from the user. The operation of the *B-Clean* system is as simple as plugging the power cord to the wall plug to power on the entire system and beginning the fully-automated sanitizing process with a single press of a start button. Then, the user can simply place a stack of baskets on top of the system for the system to process. Once a stack of baskets are feeded into the system, the user can proceed to work on other tasks without the need to monitor it. In case of an emergency or if the user would like to pause the operation at any time, the user can press the stop button on the screen, which will stop the entire system immediately.

The *B-Clean* system also offers an LCD screen for displaying to the user with information regarding the amount of disinfectant solution remaining in the system as well as an overview of the sanitization progress. According to recent interviews with potential customers, a compact system is demanded due the limited store space, thus the system is also subjected to be compact.

A.4 Technical Analysis

This part of the document will discuss *B-Clean*'s user interface design, which is based on Don Norman's Seven Elements of User Interface Interaction. Discoverability, feedback, conceptual models, affordances, signifiers, mappings, and constraints are all explored in detail as design factors.

A.4.1 Discoverability

The key principle of discoverability as a design factor is rooted in the fact that the users should know, just by looking at an interface, what their options are and how to access them [8]. The interface should be as intuitive as possible. In order to make our user interface intuitive, we use a LCD screen that is able to display all the necessary information and configurations a user might want to know and modify. We also design the user interface such that it has minimal menu depth for easy user navigation. It clearly displays essential information such as the cleaning phase and the sanitizer liquid level and eliminates irrelevant information. We will also include physical buttons such as a power-button for direct electrical control and LED’s to indicate the status of the cleaning process and whether an error has occurred during the cleaning process.

A.4.2 Feedback

System feedback is delivered to the user via a LCD-display, LED indicators and a speaker. The LCD-display provides feedback about the current state of the system, potential errors detected in the system, last maintenance of system modules and the setup of the system. The screen will also provide feedback dependent on user interactions, for example when an user clicks on the start button, it will indicate that the system has received and is processing the user action by showing the resulting state changes. LED indicators will provide direct visual indication of system state, it helps provide feedback for users who are far from the machine to easily determine if any user actions are required. The system speaker adds beneficial redundancy in the form of a sound to provide feedback to users such as it will ring an alarm when a clean cycle is done, when maintenance is due or when any error occurs. All the mentioned feedback methods work together to provide clear feedback to the end user to enhance their experience with *B-Clean*.

A.4.3 Conceptual Models

Conceptual models are mental models on how something is supposed to be done. It is the understanding of how users mentally think of tasks and how that is presented in the user interface [8]. The conceptual model helps us to better communicate the intention of our system design, specifically as our conceptual model relates more to the user’s existing mental models, the easier it will be for the user to use the system. In the case of our product *B-Clean*, it resembles existing automated systems like a dishwasher or an automated car wash. Users' previous experience on such systems will help to build the conceptual model for our user interface as these systems take in an object as an input and output cleaned objects at the end of the process.

A.4.4 Affordances

Affordance is a property related to the product that specifies the actions that can be done with the device [8]. To create a friendly user interface, our aim is to make it simple, clear and easy to work with. In the LCD interface in order to prevent the confusion for the user each statement clearly describes the action that needs to be done by the user. Moreover, if the user happens to insert a wrong input the system allows the user to go back and modify the incorrect insertion. The user interface applies design patterns to ensure a user can understand what actions are possible such as using big icons and buttons that are bright and large to be noticeable. In the hardware space, the button will provide ample affordability by labelling them to ensure the user understands it is important for the operation of the system. The system input has also been clearly designed for the user to put the basket at the appropriate place by adding highlighted labels to identify necessary steps.

A.4.5 Signifiers

Signifiers are used to provide subtle but intuitive clues on how the user is supposed to act. They serve as guides as they are perceptible information that signals the user to act as expected [8]. *B-Clean*'s signifiers include LED lights that show whether the system is running, powered-off, paused or if an error has occurred. No light indicates that the system is off while a solid green LED light indicates that the system is on . A solid blue light indicates that the cleaning process has started and is still in process. When the blue light starts flashing, it indicates that the cleaning process is complete and there are no more baskets to be cleaned. Also, a flashing orange light indicates the sanitizer liquid level is critically low. When the LED is red it indicates that an internal error has occurred and needs attention. A flashing red light indicates that the system has been paused by the user
In addition to these signifiers, UI button symbols will be included so that the users intuitively know what each button does. The interface also emits sound to alert the user when an action needs to be taken.

A.4.6 Mappings

Mapping is the connection between control and effect [8]. A good mapping is essential for an intuitive and easy to use interface. When a good mapping is in place, the controls look a lot like the items they control. For our product *B-Clean* the controls of the interface are placed in such a way that the user is informed of the underlying effects of these controls. One such example is using scroll bars to allow the user to scroll up and down on the LCD screen. Another example is that the logging text has the matching color to LED signifer for consistency.

A.4.7 Constraints

According to Norman [8], Constraints are about limiting the number of possible interactions a user can have with the system to increase the usability and guide the user to find the appropriate actions with ease. To apply this design technique in *B-Clean*, our external structure limits the number of inputs to the system. The user has the option to interact by: inserting and removing baskets from the system; an LCD Screen to control the system and buttons to turn on the system. The user interface on the LCD screen has also taken constraints into consideration. There are limited actions for the base operator who can see the information of the system and can easily go into maintenance mode to have a look at more advanced information. Advanced information in the maintenance mode gives the operator more control over the device and needs to be done by a trained professional. This helps the operator have control over the system without sacrificing generality and usability.

A.5 Engineering Standards

Our project focuses on ensuring that the end user gets the best experience from using our product. We believe that UI design directly involves the interaction between the instrument and the customers, thus in our product UI design we will meet the following engineering standards.

ISO 9241-11:2018	Ergonomics of human-system interaction - Part 11: Usability: Definitions and concepts [9]
CAN/CSA-C22.2 No. 60529:05 (R2010)	Degrees of protection provided by enclosures (IP Code) [10]
ISO 14040:2006	Environmental management - Life cycle assessment -Principles and framework [11]
CAN/CSA-C22.2 NO. 61508-1:17	Functional safety of electrical/electronic/programmable electronic safety related systems - part I: General requirements [12]

Table A.5 - Engineering Standards

A.6 Analytical Usability Testing

Analytical usability testing does not involve the target user, but it tells whether or not the target user can use our user interface easily. Our team will test the user interface at different stages of the development of our product to ensure that the errors are detected and addressed before the final product is released. Below are the list of testing components for the user interface of our product *B-Clean*.

A.6.1 Buttons

1. The on button physically powers up the circuitry of the system with the power supply.
2. The off button physically powers off the circuitry of the system from the power supply.
3. The start button initiates the cleaning process.

4. The stop button terminates the cleaning process and the system returns to its initial state.

A.6.2 LED light indicators

1. No light when the system is off.
2. A solid green LED light indicates that the system is on.
3. A solid blue light indicates that the cleaning process has begun and is still in process.
4. A flashing blue light indicates that the cleaning process is complete.
5. A flashing orange light indicates the sanitizer liquid level is critically low.
6. A solid red light indicates that an internal error has occurred and needs attention.
7. A flashing red light indicates that the system has been paused by the user.

A.6.3 Speaker

1. A sound is emitted when the sanitizer liquid level is critical
2. A sound is emitted when an error has occurred in the system and needs the user's attention
3. The sound is loud enough to alert an employee of the store

A.6.4 Graphical User Interface (GUI)

1. LCD Screen will be off when the system is off.
2. The operator home page displays information on the number of baskets cleaned so far, amount of sanitizer liquid available, system state, date and time.
3. The maintenance logging page displays all the logging that will help identify the processing state of the system. It will be helpful in identifying when and what has gone wrong.
4. The maintenance module page lists all the modules in the system that can be maintainable. It lists the last month of maintenance and if it is in need of maintenance.
5. The maintenance settings page lists all the configurable settings for the system such as amount of sanitizer to be used per basket, input basket size and setting the system to maintenance mode for servicing employees.

A.7 Empirical Usability Testing

A.7.1 Internal Testing

In order to test our product for the proof of concept stage, our team will mimic the settings of a grocery store by having stacks of dirty baskets to be cleaned and test the system according to the acceptance test plan. The team members will proceed as employees by feeding in the system with the baskets and retrieving the cleaned baskets at the end of the cleaning process. Our team members will test the interaction with the user interface and how easy it is to interact with the system. Because the team members might be biased on how well the user interface is designed we will have third parties such as friends and family members test our user interface as well for the proof of concept stage.

A.7.2 Grocery store environment Testing

For the final product stage our team intends to operate and test our product in a grocery store setting where we will gather feedback from the employees and from the store manager on how our user interface operates and if it meets their expectations. The feedback will be gathered and reviewed in order to provide our users the best experience while using our product. The feedback from our users is very important to us and it will contribute greatly to improving our design for the user interface of our final product.

A.8 Conclusion

To conclude, we have proposed our user interface for both *B-clean* main body and the LCD monitoring interface. To overcap on some of the materials that we discussed in this appendix we can mention demonstrating our potential design in the graphical presentation section, conducting analysis in A.3 and ensuring the engineering standards are met in section A.5. Moreover, we considered test methods such as Analytical usability testing and empirical usability testing to guarantee the performance of our UI design and to make sure it doesn't have bugs. Even though our proposed design is still under modifications such as the possibility of replacing the conveying belt to helical gears, we are certain that both methods are feasible with their own advantages and disadvantages, so we will keep refining the schematics as we progress more in the future. As for *B-clean*'s prototype stage, the development team will use this appendix as the guideline to implement the *B-clean*'s user interface.

Appendix B - Design Options

B.1 Control Unit Module

B.1.1 SoC

Alternatives	Justifications
Arduinos	Pros: - Affordable and flexible in terms of extending its functionalities - User-friendly IDE development environment - requires only 11.45mA under idle condition - Compact in size - Support easy interface to GPIOs Cons: - Not support full operating system - Low processing power - Difficult to develop multithreading programs
NXP i.mx family	Pros: - Powerful Processing Unit - Coupled with Yocto Projects - Capable of running full Linux kernel - Supports GPIOs - Supports multithreading Cons: - Expensive - Low availability - Relative high power consumptions - Not compact

Table B.1.1.1 - SoC Alternatives

Raspberry PI 4 Model B with 8GB RAM was chosen since it fulfills all requirements. Refer to Section 2.1 for details.

B.1.2 Software Design

Language	Native Support	Description
Python	Yes	-faster coding

		-platform independent -supports multi-threading -advanced library support for UI, AI, Data -small code length -popular QT and other frameworks for UI -syntax flexible
C	Yes	-faster compile time -supports multi-threading -historically popular for embedded systems
C++	Yes	-faster compile time -popular for embedded systems -popular QT framework for UI
Java	No	-faster compile time -platform independent -supports multi-threading -library support for UI

Table B.1.2.1 - Comparison of Programming Languages [7]

Python was selected, justification found in Section 2.2.1.1 and Section 2.2.2.1

Framework	Description
PyQt	-access to signals and slots, a useful framework related to connecting GUI actions to functions. -access to several pre-designed widgets that match the operating platform design. -widely used framework and has lots of documentation.
Tkinter	-free to use for commercial purposes. -already included in the underlying python library. -advanced libraries and common usage, does not require additional dependencies.
Kivy	-open source library for rapid development. -free to use with MIT license -graphics are built over OpenGL ES 2, toolkit comes with 20 highly extensible widgets.

Table B.1.2.2 - Comparison of GUI Frameworks in Python

Tkinter was selected, justification found in Section 2.2.2.1

B.2 Basket Setup Module

B.2.1 Electrical Design

The motors and motor driver selections are mostly based on local store inventories since these parts are easily accessible and it eliminates the need to wait for the parts to get delivered to the door. A list of market options for the stepper motor, gear motor, and the drivers are compiled in Table B2.1.1-B2.1.3. In order to power on these motor drivers and the stepper motors, an appropriate power supply is needed as well. Table B2.1.4 shows a list of AC/DC power supplies that are offered at the local store.

B.2.1.1 Stepper Motor Alternatives

#	Brand	Description	Cost (CAD)	Link
1	Stepper Motor, NEMA 23, 57BYGH5628, 8mm Shaft	Shaft dimension: 8mm Dimension of the motor: 56.4+/-1 x 96.1 +/- 1 (Width x Length in millimeters) Operating Voltage: 3.2 V Operating Current: 2.8 A/Phase Holding Torque: 270 oz. inch, 19kg.cm Steps/Revolution: 200 Step Angle: 1.8 ° No. of phases: 4	CAD $35	Lee's Electronics
2	Stepper Motor, NEMA 23, 2.4V 3 A, 57BYGH436-06,	Shaft dimension: 8mm Dimension of the motor: 56.4+/-1 x 96.1 +/- 1 (Width x Length in millimeters) Operating Voltage: 2.4 V Operating Current: 3 A Holding Torque: 125 oz. inch, 9 kg.cm Step Angle: 1.8 °	CAD $46	Lee’s Electronics
3	Stepper Motor, NEMA 17, 3V 1.7A 400Steps/Rev 42BYGHM809	Step Angle (degrees) : 0.9 2-Phase Rated Voltage: 3V Rated Current: 1.7A/Phase 5mm Diameter Drive Shaft Holding Torque: 48N.cm = 4.89 kg.cm NEMA 17 form factor	CAD $30	Lee’s Electronics

Table B.2.1.1 - Market options for Stepper Motors

Among all of the alternatives for the stepper motors, Stepper Motor, NEMA 23, 57BYGH5628, 8mm Shaft, is chosen for its holding torque capabilities. In comparison with all other stepper motor options, only Stepper Motor 57BYGH5628 has the highest holding torque, which fulfills the requirement [REQ-3.1.3.2-H.B]. Furthermore, the price of each motor is also part of the consideration in the deciding process. Therefore, in order to satisfy both the budget requirement [REQ-3.1.3.3-L.F] and the performance of the stepper motor, Stepper Motor 57BYGH5628 is the best option in the list.

B.2.1.2 Gear Motor Alternatives

#	Brand	Description	Cost (CAD)	Link
1	Gear Motor	Input Voltage: 3-6 VDC Speed: 240 RPM (3V) W/ LONG Shaft 1.5kg/cm , 12GA	CAD $15	Lee's Electronics
2	Gear Motor	Input Voltage: 6V Speed: 30 RPM , 16 GA	CAD $15	Lee's Electronics
3	Gear Motor	Input Voltage: 12V Speed: 120RPM 16 GA	CAD $15	Lee's Electronics
4	Gear Motor	Input Voltage: 12V Speed: 60RPM 13 GA	CAD $12	Lee's Electronics
5	Gear Motor	Input Voltage:12V Speed: 30RPM	CAD $12	Lee's Electronics

Table B.2.1.2 - Market options for Gear Motors

Table B2.1.2 displays a list of options for the gear motors. Since budget management is crucial in designing the *B-Clean* system, all of the gear motors that are priced at $15 are discarded automatically. After filtering out all the options that cost $15, the selection list now left with options four and five. The key difference between option four and five is the speed, which option four can execute up to 60RPM whereas option five can only operate up to 30 RPM. For the purpose of driving the helical gear, the last option is more than adequate; hence, the 12V gear motor with the speed of 30 RPM is chosen.

B.2.1.3 Motor Driver Power Supply Alternatives

#	Brand	Description	Cost (CAD)	Link

1	L9110S Motor Driver Module	Dual L9110S Chipset Motor Driver Input Voltage: 2.5~12V Current: 800mA Can drive up to two DC motors and one 4-wire 2 phase stepper motor Dimension: 23mm x 29.2mm (width x length)	CAD $8.95	Lee's Electronic
2	DRV 8825 Stepper Motor Driver	6 microstep resolutions (down to 1/32-step) Max. Continuous Current per phase : 2.2A Input Voltage: 8.5V~45V Over-temperature thermal shutdown, over-current shutdown, and under-voltage lockout Dimension: 15.20mm x 20.30mm (width x length)	CAD $11.50	Lee's Electronic
3	A4988 Mini Stepper Motor Driver 8V-35V, 2A	Five different stepping modes: full, half, 1/4, 1/8 and 1/16 Automatic current decay mode detection / selection High heat will result circuit cut, undervoltage lockout, crossover current protection Max. current per phase: 2A Input voltage: 8~35V Dimension: 15.24mm x 20.32mm (width x length)	CAD $7.95	Lee's Electronic
4	DRV 8838 Single Brushed DC Motor Driver Carrier	Max. Continuous Current: 1.8 A Peak Operating voltage: 0~11V built-in protection against reverse-voltage, under-voltage, over-current, and over-temperature Drive only one DC motor Dimension: 10.16 mm x 12.7mm (width x length)	CAD $8.00	Lee's Electronic

Table B.2.1.3 - Market options for Motor Drivers

There are various options for the motor driver module at Lee's Electronic. A driver for stepper motor and a driver for gear motor are needed for the *B-Clean* system. For the stepper motor driver, there are two options, which are DRV 8825 Stepper Motor Driver and A4988 Mini Stepper Motor Driver 8V-35V, 2A. Both driver modules share a few similar features such as circuit protections, input voltage, and the dimensions. Nonetheless, A4988 Mini Stepper Driver is cheaper in price than the DRV8825 Stepper Motor Driver. However, the applications of the stepper motor in the *B-Clean* system will require higher precision, which implies that more phases are needed. Consequently, A4988 Mini Stepper Driver can not satisfy the requirement as it only has five phases whereas DRV8825 Stepper Motor Driver has six phases. Hence, DRV8825 Driver is chosen.

The choices for the gear motor driver are DRV 8838 Single Brushed DC Motor Driver Carrier and L9110S Motor Driver Module. The price of these two motors are relatively close to each other but L9110S Motor Driver Module offers the capability of driving two motors in one module whereas the DRV8838 can only drive one. Therefore, in terms of the value and the features, L9110S Motor Driver Module is the better option than DRV8838. Furthermore, in the *B-Clean* system, there are two gear motors to drive; hence, it is more sensible to purchase one motor driver that can drive two motors at the same time rather than buying two different modules.

B.2.1.4 Alternatives for Motors Power Supply

#	Brand	Description	Cost (CAD)	Link
1	Power Supply, AC/DC, Switching, 12V, 2 A	12V, 2A Switching Power Adaptor. 2.1 mm barrel center positive connector	CAD $12.50	Lee's Electronic
2	Power Adapter, AC/DC, Switching, C7, 12V, 3A	12V DC, 3A Switching Power Adaptor. 2.1 mm barrel center positive connector	CAD $15.00	Lee's Electronic
3	Huntkey Power Adapter, AC/DC, Switching, 12V, 1 A	12V DC, 1A Switching Power Adaptor	CAD $10.00	Lee's Electronic
4	Power Adapter, AC/DC, Switching, 12V, 4A	12V DC, 4A Switching Power Adaptor	CAD $19.50	Lee's Electronic

Table B.2.1.4 - Market options for Motor Power Supply

Table B2.1.4 displays a list of options for the motor power supply. Based on the specifications of the motors that were chosen for the B-Clean system, the NEMA 23 Stepper Motor 57BYGH5628 requires the highest current to operate. Furthermore, according to design specification [Des. 3.1.10] for Motor Power Supply, an AC/DC adapter of 12V with 3A current output is required. Therefore, Power Supply, AC/DC, Switching, 12V, 2 A and Huntkey Power Adapter, AC/DC, Switching, 12V, 1 A are disqualified automatically. The short listed options are Power Adapter, AC/DC, Switching, C7, 12V, 3A and Power Adapter, AC/DC, Switching, 12V, 4A. Between these two options, Power Adapter, AC/DC, Switching, C7, 12V, 3A can satisfy the design specification meanwhile, following the budget requirement, which is to keep the cost down as much as possible. Consequently, Power Adapter, AC/DC, Switching, C7, 12V, 3A is chosen for the *B-Clean* system instead of Power Adapter, AC/DC, Switching, 12V, 4A.

B.3 Basket Sanitization Module

B.3.1 Electrical Design

#	Brand	Description	Cost (CAD)	Link
1	ULTRASONIC RANGE MEASUREMENT MODULE HY-SRF05	- Supply Voltage: 4.5-5.5V - Supply Current: 10-40mA - Output signal: Electric frequency signal, high level 5V, low level 0V - Sensor angle: up to 15 degrees - Detection distance: 2cm-450cm (no contact with the object is required)	15.00	Lee's Electronic
2	Aukru	- PIR Motion Detector Sensor - Cons : will detect motion of helical gears as well as the motion of the baskets introduced in the sanitization frame	6.99	Amazon
3	Marhynchus	- Photoelectric Switch, - Diffuse Reflective Type Sensor Switch with ABS - Cons: costly and bulky	27.99	Amazon

Table B.3.1.1 - Market options for object detection sensor

#	Brand	Description	Cost (CAD)	Link
1	JT-500	- Submersible water Pump - Small, compact and ultra quiet - low consumption, low noise - 12V voltage - Pressure : 10L/MIN	15.00	Lee's Electronic
2	Amarine	- 12V 6L/MIN Water Pressure - Default 80-130PSI; - Suction Range: 3ft-6.5ft; - Design with self-priming function - Easy connecting system, powerful automatic pressure pump - very low current draw and offer high capacity output. - Cut-Off Pressure: It can stop automatically when over voltage, it will restart automatically when	38.99	Amazon

		the pressure is less than standard. - It delivers smooth & consistent flow at all ranges of operation, while drawing low current.		
3	LEDGLE	- 12V Submersible Pump - low noise Pressure : 240L/H	14.89	Amazon

Table B.3.1.2 - Market options for water pump

#	Brand	Description	Cost (CAD)	Link
1	RELAY MODULE 1-RELAY SPDT 12VDC HIGH/LOW TRIGGER	- Pin Maximum load: AC 250V/10A, DC 30V/10A - Using SMD OPTOCOUPLER isolation, the trigger current only 5mA - 12V 1 Channel Relay Module - The module can be set to high or low triggered by jumper - Has the power indicator (green) and relay status indicator (red)	6.00	Lee's Electronics
2	RELAY MODULE 2-RELAY SPDT 12VDC HIGH/LOW TRIGGER	- Relay Type : SRD-12VDC-SL-C - Coil Voltage : DC 12V; - Pole Number : 2 Channel - Weight : 33g	12.00	Lee's Electronics

Table B.3.1.3 - Market options for relay board

#	Brand	Description	Cost (CAD)	Link
1	GLOGLOW	- 12V 85dB Electronic Buzzer - 3-24V Active Piezo Tone Buzzer Alarm Ringer Continuous Sound - Cable Length 100mm	8.30	Amazon
2	PIEZO SPEAKER 5VAC 4.0kHz 85dB LEAD TYPE	- tone type: external drive - operating voltage-:2 38Vac - rated voltage: 5Vac - current consumption: 1.5mA - resonant frequency: ±4.00. 5kHz	3.00	Lee's Electronics

		- sound pressure level: 85dB - connector type: leads		

Table B.3.1.4 - Market options for speaker board

#	Brand	Description	Cost (CAD)	Link
1	LED 5MM RGB 4-PIN C-CATHODE	- Forward Current :30 mA - Reverse Voltage : 5 V - Power Dissipation: 140 mW	1.50	Lee's Electronics
2	LED 5MM RGB 4-PIN C-ANODE	- Forward Voltage (RGB): (2.0, 3.2, 3.2)V - Luminosity (RGB): (800, 4000, 900)mcd	1.50	Lee's Electronics

Table B.3.1.5 - Market options for LED

#	Brand	Description	Cost (CAD)	Link
1	ULTRASONIC RANGE MEASUREMENT MODULE HY-SRF05	- Supply Voltage: 4.5-5.5V - Supply Current: 10-40mA - Output signal: Electric frequency signal, high level 5V, low level 0V - Sensor angle: up to 15 degrees - Detection distance: 2cm-450cm (no contact with the object is required)	15.00	Lee's Electronic
2	Liquid Level Sensor by Waveshare	- Detection depth: 48mm - Power: 2.0V ~ 5.0V	3.99	Robotshop

Table B.3.1.6 -Market options for fluid level sensor

B.3.2 Mechanical Design

B.3.2.1 Sanitizer Distribution System

#	Solution	Pros	Cons
1	Submerged basket in liquid sanitizer	• All surfaces of the basket would be exposed to sanitizer. • Potential for less complicated mechanical	• Complicated and expensive to implement plumbing and sealing. • Significant consumption of liquid sanitizer, not the most

		movement. ● Potential for easier handling of baskets by operators.	efficient use of sanitizer. ● Process of sanitizing a stack may take more time to fully fill and drain the sanitizer.
2	Spray basket with liquid sanitizer	● All surfaces of the basket would be exposed to the sanitizer. ● Efficient use of sanitizer, will not need most plumbing equipment as sanitizer can evaporate. ● Relatively cheaper solution to implement, relatively common in the industry and thus has low costs to implement. ● Can process a stack of baskets relatively quickly.	● Complicated mechanical system to process a stack of baskets.

Table B.3.2.1.1 - Pros and Cons related to Sanitization System

Spray Basket with liquid sanitizer selected, justification in Section 4.2.1.

#	Brand	Description	Cost (CAD)	Link
1	iRuiZhe	Type: flexible plastic tube Length: 8 meters Nozzles: 11, fixed at about 800cm/11 ~ 72cm apart Diameter: Outer 6mm, Inner 4mm	27.99	Amazon
2	MSDADA	Type: flexible plastic tube Length: 15.24 meters Nozzles: 20, not fixed Diameter: Outer 7mm, Inner 4mm	28.99	Amazon
3	LUTER	Type: flexible plastic tube Length: 10 meters Nozzles: 8, not fixed Diameter: Outer 7.5mm, Inner 5mm	21.99	Amazon

Table B.3.2.1.2 - Market options for Spray System

MSDADA System selected, justification in Section 4.2.1.

B.3.2.2 Sanitizer Liquid Solution

Chemical Solution	Mode of Use	Efficacy Time	Cost
Alcohol	Not for medical purposes, commonly used for cleaning surfaces	>10 s	Low
Chlorine	Healthcare purposes	>60 s	Low
Formaldehyde	Specialized healthcare, need care	>10 min	Medium
Glutaraldehyde	High level medical equipment	>5 min	High
Hydrogen Peroxide	Medical related equipment	>1 min	Low
Iodophors	Antiseptic, medical equipment	>1 min	Med
Ortho-phthalaldehyde	Medical equipment with some	>5 min	High
Peracetic acid	Medical equipment	>1 min	High
Phenolics	Medical equipment	>10 min	High
Quaternary Ammonium	Regular surfaces, floors, furniture	> 5 s	Medium

Table B.3.2.2.1 - Solution options for Sanitizer [3]

Alcohol Solution selected, justification in Section 4.2.2.

#	Brand	Description	Cost (CAD)	Link
1	ALPET D2	Type: Liquid alcohol based sanitizer Viscosity: Low Ratings: EPA, CFIA Quantity: 189.27 litres (50 Gallons)	299.99 (Sale) 1558.18 (Regular)	Grainger
2	GERMXTRA	Type: Liquid alcohol based sanitizer Viscosity: Low Ratings: Health Canada Quantity: 4 liters (1+ Gallon)	103.85	Grainger
3	C6 Medical	Type: Liquid alcohol based sanitizer Viscosity: Medium Ratings: EPA, Canadian Made Quantity: 3.78 liters (1 Gallon)	34.40 (Sale) 68.80 (Regular)	C6 Medical

Table B.3.2.2.2: Industrial Market Options for Alcohol based Sanitizer

ALPET D2 Solution selected, justification in Section 4.2.2.

Appendix C - Supporting Test Plans

Control Unit Module			
Testing Design Specifications ID	Test Descriptions	Test Results (Pass/Fail)	Comments
DES-2.1.1.1-A	Raspberry Pi 4 Model B circuit board can be mounted compactly on the side of the frame.		
DES-2.1.2.1-A	Raspberry Pi 4 Model B is able to control peripherals such as sensors, speakers, motor drivers, relay, and LEDs.		
DES-2.1.3.1-A	Raspberry Pi 4 can operate normally with 5V DC and 2A from the power supply.		
DES-2.1.5.1-A	Data is still retained in the memory even after Raspberry Pi is powered off for a while.		
DES-2.1.7.1-A	External LCD is connected to Raspberry Pi 4 for displaying simple GUI.		
DES-2.1.8.1-A	Raspberry Pi 4 should not overheat while it is operating under normal conditions.		
Date:			
Tester:			

Table C.1 - Test Plans for Subsystems and Components for Control Unit Module

Software Module			
Testing Design Specifications ID	Test Descriptions	Test Results (Pass/Fail)	Comments
DES-2.2.1.1-A	The central control module will have functional GUI and can control the system		
DES-2.2.1.2-A	System modules work independently		
DES-2.2.1.3-A	The Basket Basket Setup Module Code will be able to control and monitor the Stepper Motor and Gear Motor.		
DES-2.2.1.4-A	The Basket Basket Sanitization Module Code will be able to control and monitor the Sanitizer pump and the Sanitizer Tank Level Sensor.		
DES-2.2.2.1-A	The GUI module will be controlling the LCD Screen for User-System Interaction and will be coded on Python.		
DES-2.2.2.2-A	The GUI module will have modularized code for 4 different pages: Operator Home, Maintenance Logging, Maintenance Module and Maintenance Setting.		
DES-2.2.2.3-A	The Operator Home Page will display operator necessary information: sanitizer level, number of baskets cleaned and system error notifications		
DES-2.2.2.4-A	The Maintenance Logging Page will display all logs from		

	the system modules.		
DES-2.2.2.5-A	The Maintenance Module Page will display information related to specific electrical parts in the system.		
DES-2.2.2.6-A	The Maintenance Settings Page will let the operator modify system settings.		

Table C.2 - Test Plans for Subsystems and Components for Software Module

Basket Setup Module			
Testing Design Specifications ID	Test Descriptions	Test Results (Pass/Fail)	Comments
DES-3.1.1-A	Motor drivers should be able function under 12V and be able to drive the motors.		
DES-3.1.3-A	Motor driver circuit boards can be placed compactly on the frame of the *B-Clean*.		
DES-3.1.5-A	Stepper motors can operate in various stepping modes under the control of the motor drivers and Raspberry Pi 4.		
DES-3.1.21-A	Two stepper motors can be mounted on the top wooden frame with its width less than the actual width of the wooden structure.		
DES-3.1.22-A, DES-3.1.23-A	Stepper motors and the gear motors can trigger the gears and the helical gears of the *B-Clean* to operate normally with the input of 12V DC		

	from the AC/DC power adapter.		
DES-3.2.2-A, DES-3.2.3-A	The basket can be separated one by one and transported downwards to the sanitizing chamber.		
Date:			
Tester:			

Table C.3 - Test Plans for Subsystems and Components for Basket Setup Module

Basket Sanitization Module			
Testing Design Specifications ID	Test Descriptions	Test Results (Pass/Fail)	Comments
DES-3.1.1-A	Motor drivers should be able to 12V and be		
DES-4.1.2.1.1-A, DES-4.1.2.1.2-A, DES-4.1.2.1.3-A	Raspberry Pi 4 is able to power on the sensor and read data from it for any object that is placed in the range of 5-30 cm away from the sensor.		
DES-4.1.2.1.5-A, DES-4.1.2.1.6-A, DES-4.1.2.1.7-A	Raspberry Pi 4 can control the relay to trigger the water pump.		
DES-4.1.2.1.8-A, DES-4.1.2.1.9-A	The water pump is able to operate between 6V-12V. Depending on the provided input voltage, the pressure can vary between 6L/min to 10L/min.		
DES-4.1.3.1.1-A, DES-4.1.3.1.3-A	Raspberry Pi 4 is able to process data read in by the sensor to realize the amount of fluid remaining in the		

	system.		
DES-4.1.3.1.4-A, DES-4.1.3.1.6-A	Raspberry Pi 4 can control the output of the speaker.		
DES-4.2.1.1-A, DES-4.2.1.2-A, DES-4.2.1.3-A, DES-4.2.2.1-B, DES-4.2.2.2-F	All of the surfaces of the basket are sanitized thoroughly with industrial standard disinfectants.		
Date:			
Tester:			

Table C.4 - Test Plans for Subsystems and Components for Basket Sanitization Module