Shazamboni: Backyard Ice Resurfacer

Katharina Golder, Bryan Jaimes, Robert Ling, Yanni Pang

Executive Summary—Backyard ice rinks are gaining popularity. Especially due to the effects of Covid-19, families are building skating rinks in their backyards for kids to practice skating, develop hockey skills, and simply to have fun. These same rinks often contain imperfections such as cracks, pit holes, and chipping, and existing products to maintain the surface of the ice are often expensive, inefficient, too large, and require an extensive amount of manual work. We propose to deliver an affordable and compact semi-autonomous backyard ice resurfacer that will be remote-controlled via a mobile device application. We will create this device by using a Raspberry Pi computer to control the various working components of the machine and connect the device to the user interface via the Raspberry Pi's Bluetooth and WiFi. The machine will consist of hardware components including distance detecting sensors for object avoidance, motors to control and steer the wheels, and a camera for assisted user vision. These processes will be implemented through Python and Django software downloaded to the Raspberry Pi along with Flutter Framework and Dart programming language used to develop the cross-platform application: the Shaz App. There are currently no remote-controlled ice resurfacers on the market for at-home use. (Katharina)

- *Katharina Golder is with the Department of Electrical and Computer Engineering, Boston University, MA 02215. E-mail: kxgolder@bu.edu.*
- *Bryan Jaimes is with the Department of Electrical and Computer Engineering, Boston University, MA 02215. E-mail: bjaimes@bu.edu.*
- *Robert Ling is with the Department of Electrical and Computer Engineering, Boston University, MA 02215. E-mail: rling123@bu.edu.*
- *Yanni Pang is with the Department of Electrical and Computer Engineering, Boston University, MA 02215. E-mail: yanni@bu.edu.*

1 INTRODUCTION

The development of artificial ice rinks in the 19th Century has transformed the seasonal pastime of skating into a professional and recreational sport. While many people find enjoyment from skating today, the unavailability of public ice rinks (especially due to COVID-19) has garnered more popularity for the development of backyard ice rinks. Many homeowners install and build these rinks during the winter months for skating practice and leisure. However, backyard ice rinks often contain flaws such as cracks, pits, holes, and bumps. Furthermore, many homeowners lack access to professional ice surfacer devices. Thus, homeowners find it necessary to have a device for resurfacing and smoothing the ice in order to skate on these ice rinks without much hassle.

Currently no professional and affordable ice resurfacing machines for backyard use exist on the market, leading people to make homemade ice resurfacers. People often make these devices too large, nonoptimal, and amateurish, requiring a lot of manual work. In order for backyard ice rinks to properly become smooth, homeowners need a backyard ice resurfacer or a backyard Zamboni-type device.

The purpose of the Shazamboni is to give backyard ice rink owners the opportunity to smooth their ice rinks and to avoid skater injuries due to uneven surfaces. The Shazamboni will be affordable and eliminate the need for owners to manually pull or push the machine in the freezing temperatures. Therefore, it will be a low-maintenance, compact machine that is easy to maneuver and store. The user will be able to control the motion of the Shazamboni through a remote cross-platform application called the Shaz App.

The Shaz App will allow users to pair their Shazamboni to their phones. Users will be able to power on and off their machine to then be able to control it. Users can manually control their Shazamboni using a joystick feature on the App and see a live video stream of the Shazamboni's vision via a camera included with the machine. Other features of the App will include the ability to check the battery level, how much water is remaining in the tank, and approximately how much time is remaining for the ice rink to be completely smooth.

The Shazamboni machine will also have safety features to automatically stop the device when in close proximity to walls or objects.

Fundamentally, backyard ice rink owners will be able to smooth the surface of their rinks, eliminating bumps and cracks, with the ease of the Shazamboni. (Katharina & Robert)

2 CONCEPT DEVELOPMENT

While the appendix elaborates the

requirements our client and other team members had discussed and agreed upon, this paragraph will summarize all the major requests and requirements for the ECE team. The client needs a device suitable for shaving and cleaning the blemishes and hazards on the ice rink without requiring much manual labor. In conclusion the ECE team agreed that the client needs a device that scrapes the surface of the ice smoothly and effectively, removing any hazards and blemishes and leaving the ice surface clear and fresh. The objectives included the Shazamboni operating with snow cover after the client has shoveled the snow, leaving a mostly flat icy surface for the device to operate on and allowing the user to either manually control the device from an application or full autonomy. Constraints include sizing of the device, proper motor control, operating in appropriate climates (i.e., cold working temperatures for the rink at <= ~32 °F and ambient weather around <= ~55°F) appropriate sensors and having a budget no greater than $ 500 USD for each team (overall, the budget should not exceed $1000 USD).

The ECE team agreed the device would feature remote controls and ultrasonic sensors. Implementation of remote controls would allow the client to pilot the device on their phone and ultrasonic sensors worked best for the Shazamboni environment operation. Remote control implementation would be the easiest and most effective option for the ECE team to work on in relation to the smartphone application. Piloting movement requires reasonable amounts of code to work with and further modifications like autonomy will integrate seamlessly with the finished remote controls. An ultrasonic sensor (see section 6 for more information) price proved to be the cheapest option for the project, covered a measurable distance and worked regardless of environmental hazards like smoke, dust, fog, steam. In addition, these sensors would work independently of color, noise, light, temperature and pair well with fluid materials, which ice and snow transform into at the temperatures the Shazamboni may operate in. Finally, having a smartphone application would be the easiest option for the team and the client to install and operate. Features like Bluetooth or wireless integration would work well with the Shazamboni itself.

During the concept development, the ME and the ECE teams discussed other options for the Shazamboni. While the team agreed an application for the device would be necessary, the ME and ECE debated on the appropriate sizing and chassis design of the model. The team spent time debating whether it would be as big as a Roomba (about 14”-16” wide by 3”- 4” tall, depending on the model) or similar in size to a retail Zamboni (the first Zamboni measuring 14 ft, 9 inches in length and 9 ft, 6 inches in height). Eventually, the team decided the specified dimensions would range around 30” x 18” x 14” length by width by height. Of course, the body itself would primarily house the components. The ME team will further design and improve this aspect. While we considered the dimensions similar to a standard Roomba at first, the ME team brought up problems such as ice and water tank storage, allowing us to reconsider the sizing to the size listed in the appendix.

Another issue the ECE team faced during the design process involved the sensor type for the Shazamboni. The ECE team researched a few sensor designs based on the Roomba robot, and eventually came up with a few possibilities: LiDAR (Light detection and ranging), VSLAM (visual simultaneous localization and mapPing), infrared or ultrasonic. For LiDAR and VSLAM techniques, they proved expensive and more complex than needed for the design of the Shazamboni. While the Roomba uses vSLAM and infrared sensors to detect the best path for cleaning, we thought a similar logic could apply to the Shazamboni. The infrared sensor would detect changes in light and adjust itself accordingly. For the Shazamboni, that would mean it would trigger according to changes in light. We thought that changes in light would prove useless as most of the track stays light. In addition, the user would use the Shazamboni during the day, making light change detection more difficult. We opted to use ultrasonic sensors for these reasons.

Finally, we opted to do a fully autonomous remote-controlled device for the client. We searched for various ways to make the device move on its own, including using machine learning, reinforcement learning, or computer vision algorithms to pilot the device. We settled on just making the device remote controlled before implementing some form of autonomy due to the time and budget constraints.

(Robert)

3 System Description

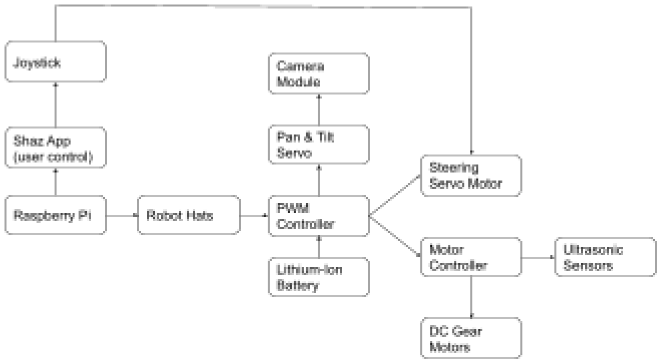
The system is also composed of two main control systems: obstacle avoidance of the Shazamboni and remote control navigation of the Shazamboni as shown in Appendix 7.5. The first control system of obstacle avoidance uses the controller of the Raspberry Pi to process the motors and wheels of the

Shazamboni. From there, the actual distance from obstacles is outputted where the ultrasonic sensors measure the distance from any obstacles. The measured distance is the negative feedback put back into the system.

The second major control system is the remote control navigation of the Shazamboni. First, the desired position of the Shazamboni is inputted into the system where the user controls the device through the Shaz App, a cross-platform mobile application built with the Flutter Framework and written in Dart. The app will be an interface to control the Shazamboni. The user will connect to a WiFi network that is emitted by the Shazamboni (Raspberry Pi wireless access point). After the WiFi connection, the mobile application will display a device in the home screen along with a "Start" button. Tapping the "Start" button will take the user to another screen with a joystick and various buttons to control the vehicle's movement, water control, and blades. The actuator is a joystick controlled by the user on the App that controls the process of the motors and wheels on the Shazamboni. The actual position of the device is outputted and the measurement is done by the user's vision assisted by the live stream video, which then loops back into the system giving the controller the error information.

In order to power and achieve the controlled systems, the system can be split into both hardware and software components.

Essentially, the Shazamboni will be a remote control car with additional features: scraping the surface of the ice, smooth the ice, and being able to detect and avoid obstacles. The ECE team is responsible for the remote control functionality and the object avoidance of the device. Therefore, the system will closely resemble a remote control car. Below is the system block diagram of the hardware components of the Shazamboni.



First, a lithium-ion battery will be powering the system. Since the device will be operating in freezing and below freezing temperatures, a lithium battery will be used, for it can function in extreme temperatures. The battery we choose will have rechargeable capabilities, and the battery percentage can be checked by the user through the Shaz App interface. Since the exact electrical components of the Shazamboni still need to be finalized with the size and weight of the mechanical engineering team's product, the current consumption for each component of the prototype is included in the Appendix 7.4.

The battery will then power the PWM controller which is connected to the Raspberry Pi through the Robot HATS. The Robot HATS supplies power to the Pi's GPIO ports outputting 5V to the PWM driver. For the prototype, the specific model for the PWM controller is the PCA9685 16-channel 12-bit I2C Bus that receives a 12V max input and outputs 3.3V/5.5V. The purpose of the PWM controller is to control the power supply from the battery and output a lower voltage than the battery itself. The PWM controller then gives the lower power the three servos: 2 for the camera and 1 steering servo. The lower voltage output from the PWM controller also powers the motor controller.

The motor controller TB6612 supplies power to each of the DC gear motors controlling the motors' direction. The speed of the motors is controlled by the PWM input to the motor driver.

Distance detecting sensors will also be necessary to avoid collisions with obstacles. For this project, ultrasonic sensors will be used since they are reliable and affordable. Additionally, sunlight will not interfere with the ultrasonic sensors unlike infrared sensors. This capability is important because the Shazamboni will be operating outdoors. The sensors should be controlled as close to the DC gear motors as possible so that they can stop the motors at an instant. Therefore, the sensors should be controlled via the motor controller. For the prototype, the sensors were not included with the car kit, so they must be integrated into its design for the prototype and implemented in a similar way in the final design.

Lastly, the main controller that holds all the necessary code is the Raspberry Pi 4. Ultimately, the Raspberry Pi was chosen over an Arduino since it already has built-in WiFi and Bluetooth capabilities. The user application can be paired to the Shaz App using Bluetooth, and then it will communicate via WiFi once the device is paired. The Shaz App will then allow the user to control the hardware components.

All the hardware components that are part of the prototype will ultimately be transferred to the final Shazamboni device. The system will remain the same; however, there will be some changes to consider.

First, the PWM controller will need to be adjusted since the battery used for the final product will be greater than 12V. Additionally, the servos that control steering and the motor controller will most likely need more power than the 5V outputted to the prototype. The DC gear motors will need to be substantially larger and thus, require more power. Therefore, the PWM controller must handle different input and output voltages; however, the functionality of the controller remains the same. The software may need to be fine-tuned for the possibility of higher voltage requirements and slower movement of the Shazamboni.

See Appendix 7.3 for the functional analysis design representation of the device.
(Katharina and Yanni)

4 FIRST SEMESTER PROGRESS

This semester, we were able to complete some of the tasks required for our first milestone. The two ultrasonic sensors were wired to the Raspberry Pi 4 via its GPIO ports and controlled using Python3. During the First Deliverable Testing, we were able to demonstrate that the ultrasonic sensors measured various distances of 4, 8, and 12cm correctly. These ultrasonic sensors will ultimately be used to stop the Shazamboni from impacting a wall or other object. We were also able to get a skeleton of the cross-platform application developed. During the First Deliverable Testing, we were able to demonstrate the application running within the iOS simulator. The home page and the button to advance to the control pages were demonstrated as working. On the control page, we were able to show the joystick element. As we shifted the joystick around on the screen, the angle and distance successfully printed in the console. This semester, we purchased a car kit to work on methods of control. We were able to build the car kit and demonstrated that it was able to live stream from its camera and drive around from the Django web interface. The next step for our project is to figure out how to integrate a livestream into the mobile app, then control motors. Concurrently, we have held meetings with the mechanical engineering team to discuss the mechanical requirements of the Shazamboni, its functionalities, and its physical form. Aside from working on the Shazamboni, we spent considerable time during the semester working on presentations, planning and writing reports. (Yanni)

5 TECHNICAL PLAN

The requirements include the following: Hardware and Software integration and testing. With regard to hardware, testing sensors, remote control, and wiring of the Shazamboni take the main priority of focus. In regard to software, user accessibility and seamless connectivity take top priority. The milestones include finishing touches on the sensors and remote control, integration of controls with the ME (sub) team and eventual completion of the overall Shazamboni and Shaz App.

Milestone 1: Finishing touches on the sensors and remote control
Completion of Milestone 1 began on December 10, 2021 and will conclude on January 29, 2022. The ECE team will attempt to carry out the finishing touches during the period and finish them no later than February 5, 2022, which is a hard deadline. Tasks include integrating sensors and remote control with the Shaz App.

Task 1.1 Ultrasonic Sensors
We will apply and test four HC-SR04 ultrasonic sensors for our Shazamboni. They will help the Shazamboni detect objects or obstacles close to the Shazamboni. These sensors will meet specifications for detection of objects. A Raspberry Pi will serve as the main computer for the sensors as we adjust the Pinouts to account for the echo,trigger, 5V power supply and ground. The distances the four sensors will measure will vary between 4 to 12 cm, with 12 cm flagging the Shazamboni to stop. We will test each sensor with objects including a card, a block of wood, and finally cotton balls acting as left behind snow pile up. Lead: Robert Ling; Assisting: Katharina Golder, Yanni Pang

Task 1.2 Remote Control
We will fully integrate and test a remote-control application from our Shaz App to pilot the test car based on the remote-control application we received from the kit. This remote control will meet specifications for movement of the Shazamboni (i.e., pilot the device forwards, backwards, left and right) as well as stream live footage of the Shazamboni when activated. To provide a live video stream, we will be using MJPG Streamer and its corresponding Flutter package. Movement of the motors will be interfaced via a WiFi connection. We will have to write some Dart code to communicate with a Python module on the Raspberry Pi 4 which in turn sends GPIO signals to the various actuators or sensors. A mock-up device on our team's computers will feature the UI for the remote control which features a joystick as an actuator which in turn pilots the motors and wheels of the Shazamboni. We will test the device outside with a mock up arena. Lead: Yanni Pang; Assisting: Bryan Jaimes, Katharina Golder, Robert Ling

Task 1.3 Shaz App
We will further develop and test the Shaz App for better user integration and accessibility. We will integrate the camera features and joystick movements, as stated previously. In addition, we plan to find a way to connect the Shaz App and Shazamboni device via Bluetooth set up or Wi-Fi set up. The Shaz App will launch on Android and iOS devices and meet their respective requirements. Again, the app will be using the Flutter framework along with Dart code. The app should have a home screen that directs the user to a control page. On the control page, there will be a livestream with a view of the camera mounted on the Shazamboni. Buttons will be provided for the user to control the water flow along with the spinning blades. We will continue testing on our computers before moving the application to the actual smartphones. Lead: Bryan Jaimes; Assisting: Yanni Pang

Milestone 2: Integration of controls with ME team
Completion of Milestone 1 will begin on January 29, 2022, and conclude in the week of April 1, 2022, which is a hard deadline. Tasks include power supply, icebox controls and potentially autonomy, along with integration with the ME team's Shazamboni design.

Task 2.1 Power Supply
A 5V battery capable of delivering power to the Raspberry Pi will be designed, tested and implemented for the Raspberry Pi. Right now, we plan to continue using our two 18650 rechargeable batteries to power the toy car for testing and continuing to connect the Pi to a USB-C wired cable, but we will soon look for more powerful batteries that can charge the Raspberry Pi 4 as well. The battery will be chargeable and weather-resistant. Testing will include determining how many minutes the device can operate on a single charge. Lead: Katharina Golder; Assisting: Robert Ling

Task 2.2 Water Tank Control
While the ME team will focus on the major aspects of the water tank design (i.e., the weight, the dimensions and material of the water tank), the ECE team will contend with figuring out the controls of depleting the water tank of dirty water and accumulated snow. The Raspberry Pi will handle dispensing the water as the Shazamboni cleans the rink. The ME team will coordinate with the ECE team in finding a suitable rink for testing. Lead: Robert Ling; Assisting: Katharina Golder, Bryan Jaimes, Yanni Pang

Task 2.3 ME Team Wiring
Once the ECE team has completed the controls and sensors, the final piece will be to wire all the parts together in the Shazamboni with the help of the ME team. Overall, the Shazamboni will have a chargeable port and wires and sensors integrated and hidden from view, along with the additions the ME team will make during the process. Testing will be on various rinks. Lead: Katharina Golder; Assisting: Bryan Jaimes, Robert Ling, Yanni Pang

Task 2.4 Autonomy
Additionally, once the ECE team has figured out the controls, the ECE team will do their best to create an autonomous device, if the user desires. The Shazamboni will automatically stop and reverse if it reaches a nearby obstacle, warning the user to reroute. For autonomy, computer vision algorithms like VSLAM can be used to plot a circular path in the ice rink. Testing will include a small area outside and eventually an ice rink. Lead: Yanni Pang; Assisting: Bryan Jaimes, Robert Ling

Milestone 3: Wrap up
This will occur on the week of April 1, 2022, and conclude on May 6, 2022, which is ECE day. These tasks include customer installation and finishing last minute reports and presentations.

Task 3.1 Customer Installation
The Shazamboni team will work on integrating and completing the full Shazamboni and Shaz App for the customer. The team will work on meeting all the specifications of the customer as well as requirements. The team will test on the customer's homemade rink. Lead: The Team Leads during that period; Assisting: The rest of the team
(Robert)

6 Budget Estimate

The client explicitly told the team the budget of the Shazamboni should be no greater than $500; however, since both the ECE team and the ME team are working together, the budget can be extended to an extra $500. The Bill of Materials for the prototype is listed in Appendix 7.6.

The majority of the components that are used for the prototype will also be used for the final design, so there will not be many items that need to be repurchased. The only item that the team did not have to buy is the Raspberry Pi 4 which was supplied during the Senior Design Hardware MiniProject. However, its price is included in the overall budget statements.

In summary, the working prototype cost a total of $179.95 with the majority of the cost accounting for the SunFounder Car Kit. It was the most expensive purchase of $99.99; however, the components it came with were worth its value. The kit came with all the necessary hardware components for the car to operate and the materials to construct the building of the car. By ordering the kit as a whole, it allowed for the team to save money on buying individual components and save that money for the final design. The car kit also included open-source software which is useful in learning how the car was designed to run. The included software allows us to learn and write our own code for the Shazamboni's final design. Next semester, we will have to purchase more parts to integrate into the actual functional prototype. For example, we will need to purchase a suitable battery, a motor with enough torque, and other motor controllers for the full-sized prototype.

The final budget the team spent so far is around $180.

(Katharina and Yanni)

Acknowledgment

The authors wish to thank Roberta Steinberg for reviewing and revising the paper.

References

[1] "Clean ice. clean air. clear choice.," Zamboni, 11-May-2020. [Online]. Available: https://zamboni.com/. [Accessed: 10-Oct-2021].

[2] "Innovative OlymPia ice resurfacing machines: Resurfice Corp..," Resurfice. [Online]. Available: http://www.resurfice.com/. [Accessed: 10-Oct-2021].

[3] S. Stones, "Homemade ice-surfacing machine gives Old Washing Machine New Life," Winnipeg, 25-Jan-2021. [Online]. Available: https://winnipeg.ctvnews.ca/homemade-ice-surfacing-machine-gives-old-washing-machine-new-life-1.5280237. [Accessed: 10-Oct-2021].

[4] Shelleysteevesglobalnews, "New Brunswick grandfather creates do-it-yourself 'zamboni'," Global News, 23-Jan-2017. [Online]. Available: https://globalnews.ca/news/3200088/new-brunswick-grandfather-creates-do-it-yourself-zamboni/. [Accessed: 10-Oct-2021].

[5] SunFounder [PDF] http://wiki.sunfounder.cc/images/c/c9/Raspberry_Pi_Smart_Video_Car_V2.0.pdf

[6] T. W. Staff, "A brief history of the zamboni," The Week, 08-Jan-2015. [Online]. Available: https://theweek.com/articles/441560/brief-history-zamboni. [Accessed: 10-Oct-2021].

7 ATTACHMENTS

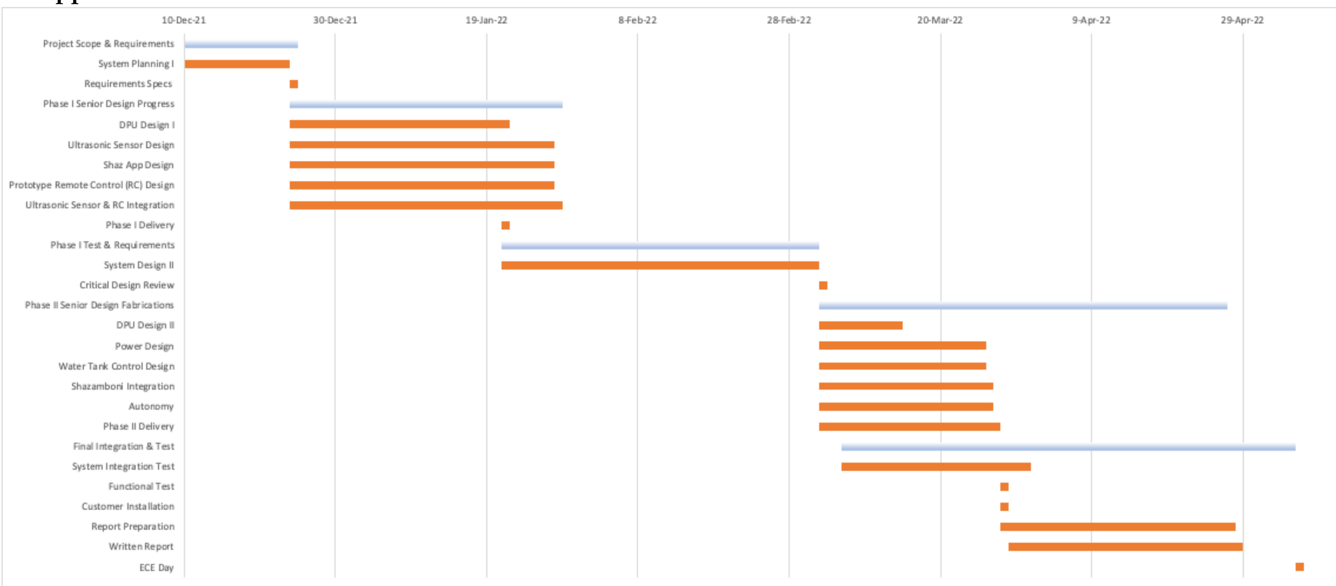
7.1 Appendix 1 - Engineering Requirements

Requirement	Value, range, tolerance, units
Dimensions	30” x 18” x 14” Chassis
Power	12V rechargeable battery, 5V rechargeable battery for Pi
Remote Control / Motor	Raspberry Pi 4 with python code for motor control
Sensors	4 HC-SR04 ultrasonic sensors for detecting objects 12 cm away
Mobile Application	Flutter framework and Dart code designed application for iOS or Android devices
Ice Box / Water Supply	Raspberry Pi 4 with python code for detection and water supply

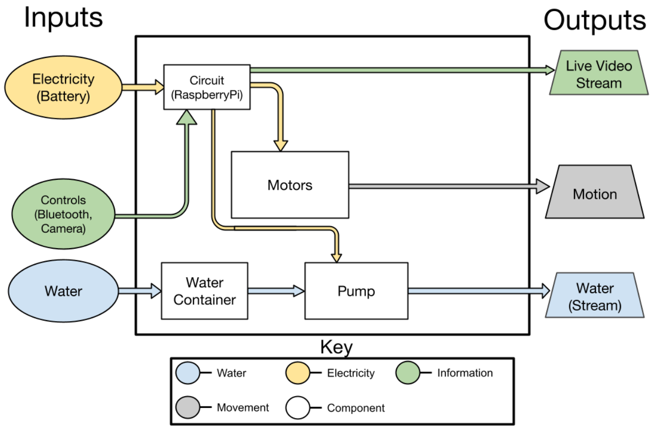
The ECE team will ensure the Shazamboni will have the following required for customer satisfaction: power, remote control and motor, sensors, a smartphone app, ice and water supply controls. In addition, the ME team will oversee the following: Shazamboni dimensions and material, ice box/ water supply wiring, wheels and finally motor selection, which the ECE team will closely monitor and work with. The power supply will stem from a similar source which the Roomba device already uses or in other words a 12V rechargeable battery that the customer can easily plug into when not in use. A 5V battery source will serve as the main source of power for the Raspberry Pi 4, which will derive its power from the 12V battery. Moreover, the Raspberry Pi 4 will control the following: remote controls/motors, sensors and connection to the Shaz App. Wires stemming from the Pinout of the Pi will connect to the wheels of the device for the customer to control. Sensors will include 4 HC-SR04 ultrasonic sensors that will connect to the Pi and each other via the Pinouts of the Raspberry Pi 4. They will detect distances front, back, left and right of the Shazamboni no less than 12 cm away. The Shaz App made from Flutter will serve as the primary user interface for control and sensing Shazamboni power, ice box supply and warnings for the user. Finally, the Raspberry Pi 4 will control how to dispense water and warn users when the ice box reaches full capacity.

(Robert)

7.2 Appendix 2 - Gantt Chart



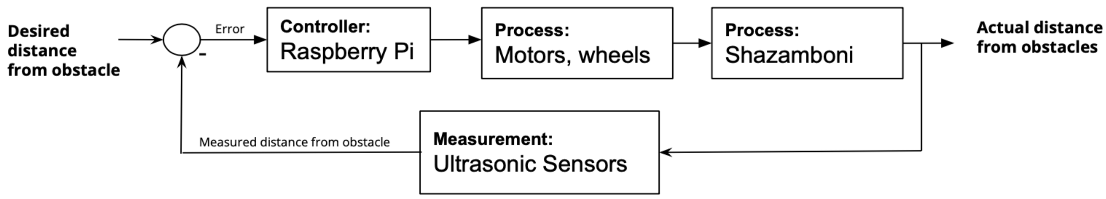
7.3 Appendix 3 - Functional Analysis

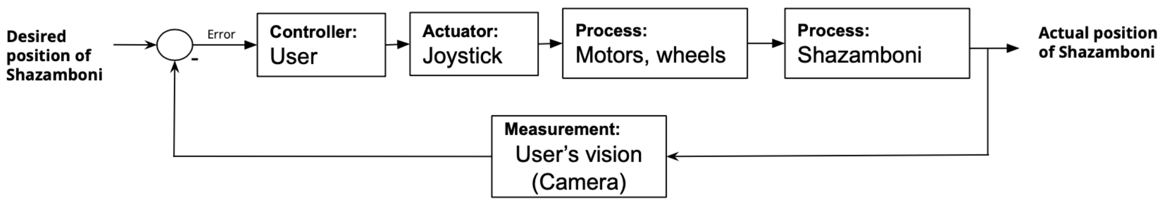


7.4 Appendix 4 - Prototype Current Consumption

Component	**Quantity**	**Current [mA]**
SunFounder SF0180 Servo	3	200
DC Gear Motor	2	<80
DC Gear Reducer	2	<120
Ultrasonic Sensor HC-SR04	2	15

7.5 Appendix 5 - Control Systems





7.6 Appendix 6 - Bill of Materials

Item	**Supplier**	**Quantity**	**Unit Price ($)**	**Being reused? (Y/N)**	**Total ($)**
		Bought Items			
Ultrasonic Sensors	Amazon	1	12.99	Y	12.99
330 Ohm Resistors	Amazon	1	5.99	Y	5.99
470 Ohm Resistors	Amazon	1	5.99	Y	5.99
FunFounder Car Kit	Amazon	1	99.99	Y (parts)	99.99
18650 Rechargeable Batteries	Amazon	1	18.99	N	18.99
Motor Kit (not used)	Amazon	1	12.99	N	12.99
Camera (not used)	Amazon	1	9.99	TBD	9.99
Total					166.93
		Provided Items			
Raspberry Pi 4	Mini Project	1	45.00	Y	45.00
Total					45.00
Total Used Parts					179.95
Total					211.93