Boston University
Electrical & Computer Engineering
EC464 Capstone Senior Design Project

User's Manual

Shazamboni

The Backyard Ice Resurfacer

Submitted to

Alan D. Pisano Jr.
112 Brookline Street
Needham, MA 02492
617-817-0113
alandpisano@gmail.com

by

Team 10
Shazamboni

Team Members

Katharina Golder kxgolder@bu.edu
Bryan Jaimes bjaimes@bu.edu
Robert Ling rling123@bu.edu
Yanni Pang yanni@bu.edu

Submitted: April 18, 2022

User Manual

Table of Contents

Executive Summary

Team 10: Shazamboni

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 for relaxing entertainment. These same rinks often contain hazards such as cracks, pit holes, and chipping due to unpredictable weather. Even just skating on its surface can cause major defects. 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 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.

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.

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.

The remainder of this manual will entail details on how the system works, the user interface (Shaz App), and how to install and operate the Shazamboni.

2 System Overview and Installation

There are two components to the Shazamboni product: the physical device used to navigate on the ice to leave it smooth and the ShazApp that the user uses to control the Shazamboni machine. Information about how the system works and how to set up the device is outlined in this section.

2.1 Overview block diagram

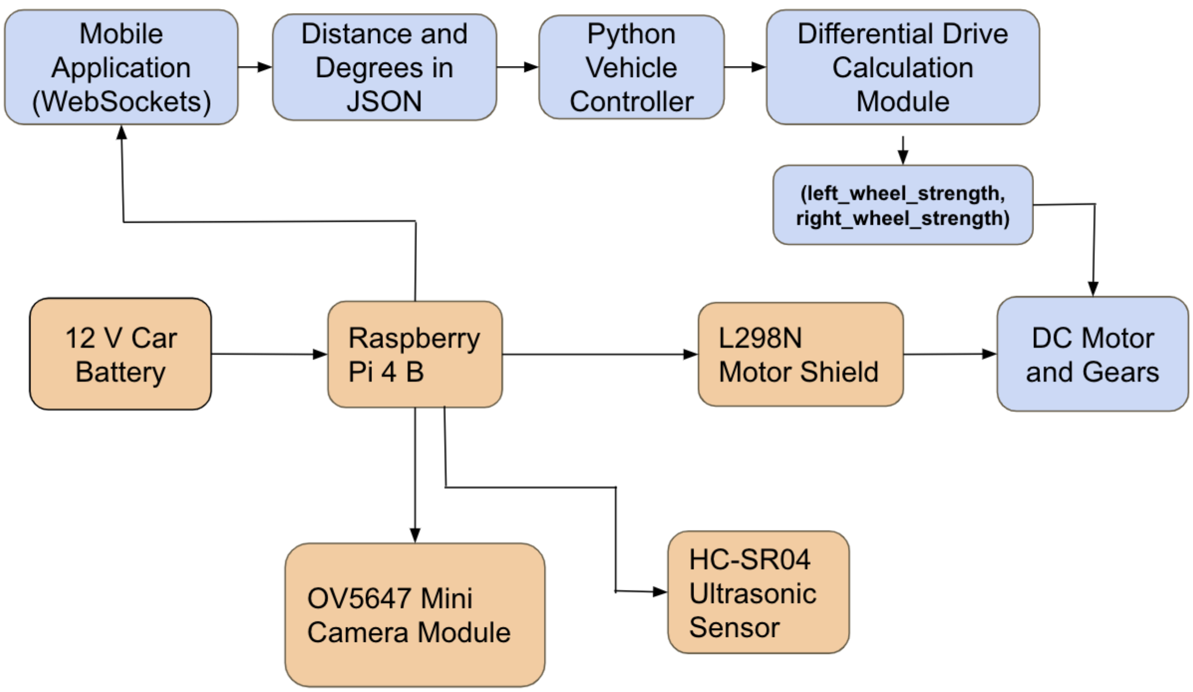

Figure 2.1 The system overview diagram of the Shazamboni. The blocks shaded in blue are the software components, and the blocks shaded in orange are the hardware components of the system.

2.2 User interface

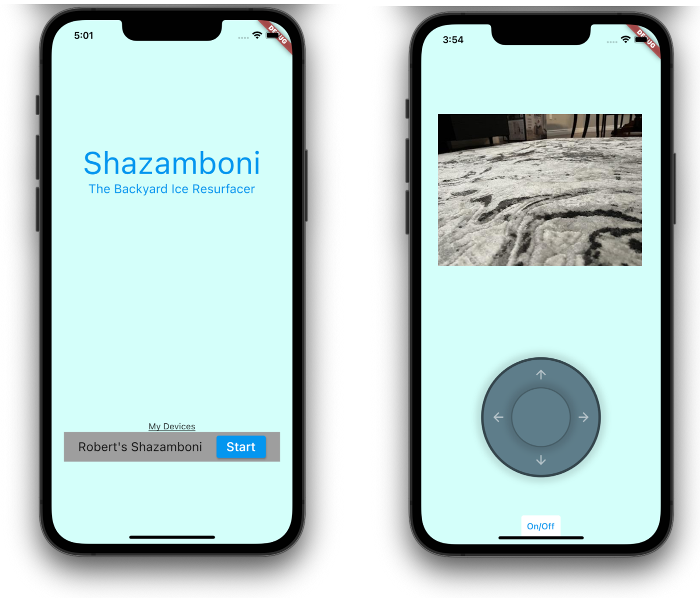

Figure 2.2 The Shaz App UI.

2.3 Physical description

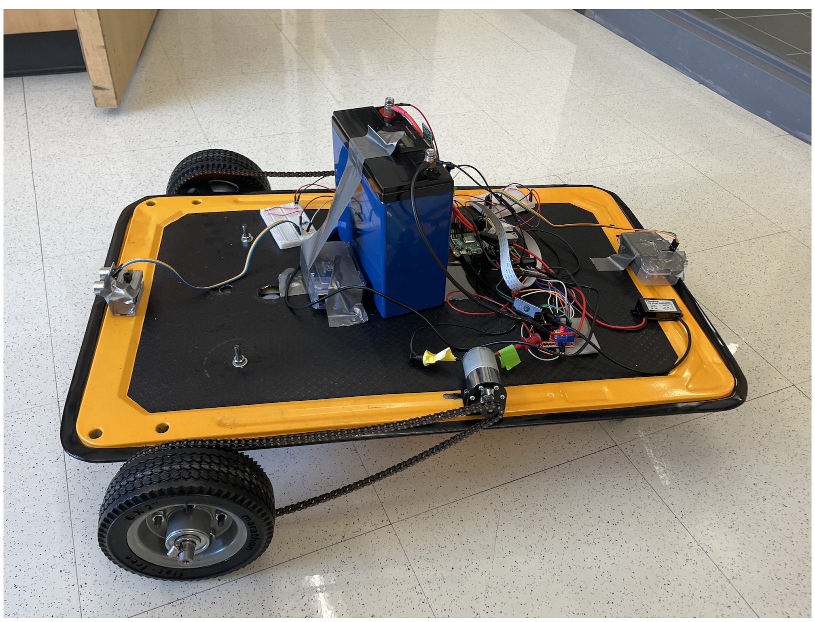

Figure 2.3 Electrical and computer components of the Shazamboni.

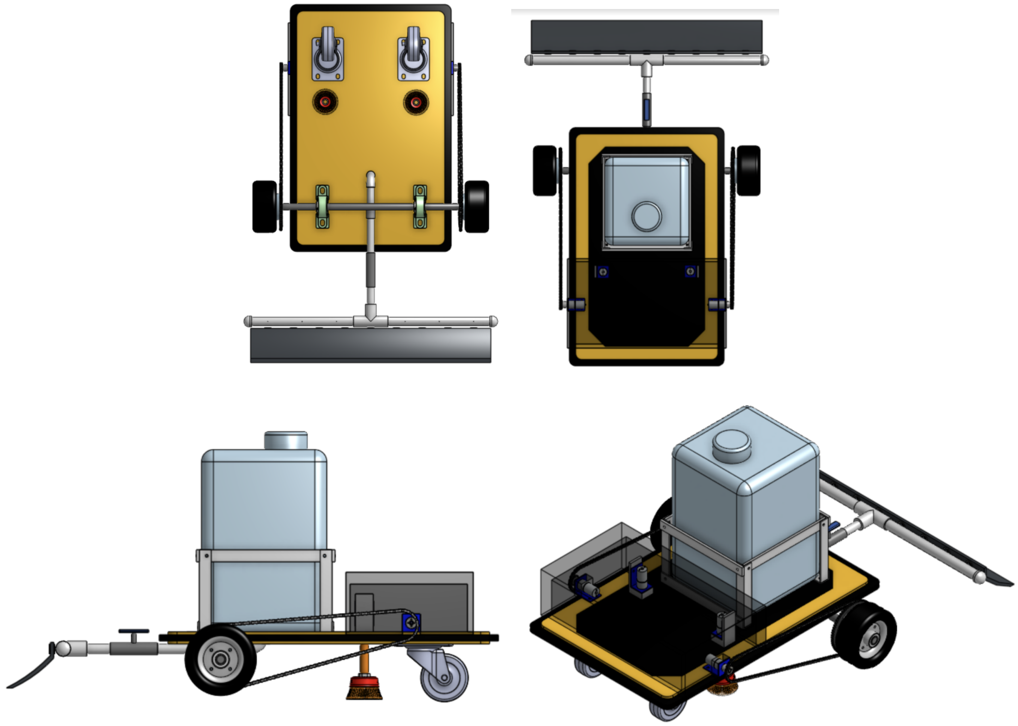

Figure 2.4. Mechanical Engineering team's CAD drawings of the Shazamboni.

The final physical design of the Shazamboni includes the ME team's final designs for the overall shape and dimensions of the product as well as the integration of the wires and hardware components (Figure 2.3) the ECE team designed separately. A specialized compartment which is a 7x8 area on the drive train (Figure 2.4) will store all the hardware components.

2.4 Installation, setup, and support

Installation

The Shazamboni machine will come assembled and fully charged. For charging, a separate charger is needed. A standard "Battery Tender" may be used to charge the device. No home WiFi or hotspots are required to use this device.

Setup

1. Place the Shazamboni machine onto the surface of the ice face-forward towards the area that will be resurfaced first.
2. Download the Shaz App from the App Store, Google Play Store, or sideload the app through the source code.
3. Connect to the Shazamboni's WiFi network.
4. Open the Shaz App, and press **Start**.
5. Use the joystick aided by the live stream video to navigate the Shazamboni across the ice, leaving a smooth finish.

Support

- The Shazamboni’s operation will work entirely in the ShazApp. The Shazamboni will need to be charged by the user’s battery charger.
  If there is a connection issue, toggle the On/Off button.

3 Operation of the Project

3.1 Operating Mode 1: Normal Operation

For normal operation, the user will do the following:

1. The user will use the ShazApp to login and communicate with the Shazamboni
2. The user will use the joystick controls to pilot the Shazamboni to clean the ice
3. If the user encounters a wall, the user should be able to back up after a period of five seconds has elapsed.

The user will have access to the livestream and joystick controls when logged in.

3.2 Operating Mode 2: Abnormal Operations

Abnormal operations include the Shazamboni failing to stop near obstacles, running out of battery, short circuiting wires, and wheels failing to turn.

To safely exit and stop any issues, be sure to unplug the wires if possible or wait for the device to run out of power. In addition, restart the ShazApp by shutting off and re-entering the ShazApp on the user's device. Follow the instructions outlined in 2.4 for set up instructions after these steps are complete.

3.3 Safety Issues

1. The Shazamboni has been tested with a 12V (Volt) 24Ah (Amp hour) car battery that powers all the hardware components. It works based on the North American standards which is 120V at 60Hz and thus may not work as well in other countries. As a result, usage in countries besides the U.S.A and Canada is strongly discouraged. Users must use the standard car charger outlet for charging the battery. Users should follow appropriate guidelines for proper charging and powering on for the best results.

2. The Shazamboni must only be used for ice resurfacing. Users should not allow live entities to ride on top of the Shazamboni. In addition, the Shazamboni should work only on ice surfaces that users have constructed themselves and know beforehand the dimensions of the ice surface.

3. The Shazamboni is resistant to cold weather elements but the user should ensure proper care of the wires and connections. Make sure that elements like water or ice have not breached the sensitive components of the Shazamboni. While hazards like fire or explosions are unlikely, the user must ensure the wires do not appear to short circuit.

4. The Shazamboni must be housed in a closed environment similar to a warehouse, garage, or shed ensuring the longevity of the device.

4 Technical Background

Hardware

Computer System:

The main computer system of the Shazamboni is a Raspberry Pi 4B. Raspberry Pis are cost effective computers that run on Linux on which programs can be downloaded. The Shazamboni's hardware components are controlled via the programming language Python. They also have a variety of GPIO (general purpose input/output) pins that allow the Pi to control electrical components – in this case the Pi runs the operation of the motors, sensors, and camera. The particular model used for the Shazamboni is the Raspberry Pi 4B which includes built-in WiFi and Bluetooth. These features are utilized in communication with the Shaz App and the physical Shazamboni.

Motor Controls:

The two rear motors of the Shazamboni are controlled by both the Raspberry Pi and a L298N Motor Driver Controller. The motor controller allows both motors to operate independently so that each motor's speed and direction can be controlled separately. Through the use of Python scripts downloaded to the Raspberry Pi, the user will be able to navigate the Shazamboni through the Shaz App's joystick.

Sensors:

In order to avoid potential obstacles on the rink and colliding into walls, two HC-SR04 Ultrasonic Sensors are incorporated into the design, one in the front and one in the rear. Ultrasonic sensors measure the distance between itself and an object using ultrasonic sound waves. Using this technology, the sensors can detect when the Shazamboni is getting too close to obstacles, and when it does, it can immediately stop the machine to avoid any collisions. Additionally, these sensors work particularly well in bright sunlight or in dark environments to accommodate the various weather conditions where it operates.

Camera:

The Shazamboni will use the OV5647 Mini Camera Module for the Raspberry Pi from Arducam. This cost effective webcam will use python scripts on the Raspberry Pi to stream live footage of the Shazamboni's view by communicating with the ShazApp through websockets. The user will be able to see the footage on their phone in real time. The camera has a steady 30fps@1080P and 60fps@720P, but will use the latter for steadier frame rates. The field of view is 50°x41°.

Power Supply:

The Shazamboni is powered by a rechargeable 12V Lithium Iron Phosphate battery (LiFePO4). When taking into consideration all the hardware components that will be operating simultaneously, the fully-charged battery will last approximately 6 hours. The chemical make-up of this battery is built to perform in temperatures between -4°F and

167°F, so operating the Shazamboni during winter time in below freezing temperatures should not interfere with the device's charge.

Software

Flutter / Dart:

The Shaz App is a cross-platform application that is used to control the Shazamboni and developed using Flutter. Flutter Framework is a software development kit created by Google to help users develop UI software to create mobile applications. The platform applications include Android, iOS, Google Fuchsia, Web platform, Linux, macOS, and Windows. The programming language used to utilize Flutter Framework is Dart. Dart is a typed object-oriented language.

Built in WiFi / Ad hoc:

In order for the Shaz App to be able to communicate with the Shazamboni wirelessly, some sort of a connection is required between the two devices. The method this system uses is configuring the Raspberry Pi in Ad Hoc Network mode. In other words, the Raspberry Pi has "built-in WiFi," or its own network with its own unique IP address, allowing users to connect to their Shazamboni without the need of a separate home WiFi or hotspot.

Socket Communication:

Through the WiFi ad hoc network connection, the Raspberry Pi and the cross-platform application use websockets to communicate. Socket communication is used to send JSON packets containing information about the user's movements from the joystick. These packets are decoded on the Raspberry Pi and are used to drive the two motors present on the vehicle.

Camera Live Stream

The camera live stream uses mjpg-streamer on the Raspberry Pi along with the ArduCam camera module. It starts a stream on start up on port 8080 on the Raspberry Pi as it starts up. This stream is played through the cross-platform application.

5 Relevant Engineering Standards

In terms of electrical hardware, the Shazamboni follows the standards of IEEE and the American National Standards Institute. Starting with the power socket, the Shazamboni shall use a Type A plug that will conform to the North American standards. The internal controls, i.e., the motor shield, HC-SR04 ultrasonic sensor and Raspberry Pi, will use a voltage of 5-12 V and have two resistors, 330 Ohm and 470 Ohm connected to the ultrasonic sensor. Standard multicolored jumper wires (cables) will connect all the internal controls together. The wiring of the Shazamboni shall include a defined ground terminal in which all internal controls will find common use. The wires themselves shall follow the standards of the National Electrical Safety Code, which is outlined in detail on Section 9. In terms of mechanical hardware, the mechanical engineering team will follow the standards established by their professors.

Regarding the software components of the Shazamboni, the team shall use the approved libraries of Python and in addition any open-source libraries. The team will be sure to comply with the communications requirements set up also by IEEE and use a standard IEEE 802.11 for any ShazApp connections requiring standard WiFi as well as a custom Ad-Hoc network for direct communication with the Shazamboni motors and webcam.

6 Cost Breakdown

Project Costs for Production of Beta Version (Next Unit after Prototype)				
Hardware Components				
Item	Quantity	Description	Unit Cost	Extended Cost
1	1	Raspberry Pi 4B	$45.00	$45.00
2	2	HC-SR04 Ultrasonic Sensors	$2.60	$5.20
3	1	12V 24Ah LiFePO4 Battery	$137.99	$137.99
4	2	DC Motors 12V 200 RPM	$11.88	$23.76
5	4	330 & 470 Ohm Resistors	$0.06	$0.24
6	1	L298N Motor Driver Controller	$3.30	$3.30
7	1	Fuse	$1.30	$1.30
8	1	Camera	$9.99	$9.99
9	2	Breadboard	$2.00	$4.00
10	1	12V to 5V Converter	$11.99	$11.99
11		Miscellaneous Wires	$8.99	$8.99
Beta Version-Total Cost				$393.19

Mechanical Components				
Item	Quantity	Description	Unit Cost	Extended Cost
1	1	PVC 3/4" T	$0.76	$0.76
2	1	PVC 3/4" Valve	$2.98	$2.98
3	1	PVC 3/4" M Adapter	$0.63	$0.63
4	2	PVC 3/4" End Caps	$0.75	$1.50
5	1	PVC Cement	$12.34	$12.34
6	1	Storage Box	$10.98	$10.98
7	1	PVC 3/4" Pipe	$5.56	$5.56
8	1	PVC 3/4" F Adapter	$0.93	$0.93
9	2	PVC 1/2 Elbows	$0.87	$1.74

10	2	Motor Brush	$10.72	$21.44
11	2	Adapter Brush	$13.99	$27.98
12	1	Cart	$54.99	$54.99
13	2	Water Jug	$17.86	$35.72
14	1	Zip Ties	$5.49	$5.49
15	2	Brush	$24.99	$49.98
16	1	McMaster Piping	$21.05	$21.05
17	1	Small Sprocket	$10.66	$10.66
18	1	Brush Motor Adapter	$9.32	$9.32
19	1	L Brackets	$8.78	$8.78
20	1	Al Angle Bar	$8.42	$8.42
21	1	Al Flat Bar	$11.31	$11.31
22	1	Axle Nuts	$8.99	$8.99
23	2	6" Wheels	$11.25	$22.50
24	1	Pillow Blocks	$24.99	$24.99
25	1	Axle Nuts	$16.07	$16.07
26	2	Large Sprocket	$16.99	$33.98
27	2	Drive Motor Bracket	$6.99	$13.98
28	1	Shaft Collars	$12.88	$12.88
29	2	Brush Motors	$14.99	$29.98
30	1	Brush Adapter	$12.99	$12.99
31	1	Aluminum	$60.26	$60.26
Beta Version-Total Cost				$539.19

The budget tables above show the combined budget expenses of both the ECE team and ME team. Each team has an allocated budget of $1000, $500 of which Boston University pays for. In total, both teams have a combined budget of $2000 with $1000 provided by the university.

7 Appendices

7.1 Appendix A - Specifications

Requirement	Value, range, tolerance, units
Dimensions	30” x 18” x 14” Chassis
Weight	~5 kg (ECE components only)
Power Supply	• 12V 24Ah Lithium Iron Phosphate Rechargeable Battery • 5V Raspberry Pi
Motion	• 12V 200RPM DC Gear Motors • L298N Motor Driver Controller • Python Script
Object Detection	2 HC-SR04 ultrasonic sensors for detecting objects 7 cm away
Mobile Application	Flutter framework and Dart programming language for iOS or Android devices
Operating Temperature	LiFePO4 battery performs in -4°F and 167°F
Cost	$925.39

7.2 Appendix B – Team Information

Katharina Golder, Bryan Jaimes, Robert Ling, and Yanni Pang make the ECE team of the backyard Zamboni project, known as the Shazamboni and Shaz App. Katharina and Robert are the electrical engineers, and Bryan and Yanni are the computer engineers on the team.

Katharina Golder is a graduating electrical engineering student who enjoyed her courses in signals and systems and electric energy systems. She will be working at AKF Group in New York in the fall as an EIT in electrical engineering.

Bryan Jaimes is a graduating computer engineering student who passionately enjoyed his courses in software engineering. He will be working at Liberty Mutual Insurance in the fall as a software engineer.

Robert Ling is a graduating electrical engineering student who enjoyed his courses in VLSI design, software radios, and reinforcement learning. He plans to work in the industry for a few years before pursuing a masters degree in VLSI chip design and business to contribute to a better future.

Yanni Pang is graduating this May with a degree in computer engineering. After college, he will be working at Intersystems in Cambridge as a quality development engineer. He also has plans to complete a masters degree also in computer engineering.

In addition to the ECE team, there is also a mechanical engineering team working on the Shazamboni: Marianna Natale, Mei Singer, and Beau Walsh.