Automatic Cocktail Dispenser

ECE 445 Design Document - Spring 2023

Group No. 32

Carson Van Pelt, Benjamin Thuma, Caleb Kong

Professor: Viktor Gruev

TA: Prannoy Kathiresan

Contents

1. Introduction

1.1 Problem

Once people are 21 years old, enjoying unique and flavorful alcoholic beverages can be a fun social activity to do with friends and family. There are thousands of different concoctions that are ready to be tried, but it's up to the consumer to know what they want and how to make it. Going out to a cocktail bar is also an option, but to go consistently would certainly hurt financially, as the drinks tend to be very expensive (around $10 to $15 on average)[1]. In addition, purchasing drinks in a social environment where all your faith is in a stranger to mix your drinks pose potential dangers. Some bartenders may be interested in serving high-volume cocktails that consumers wouldn't expect, which could lead to overconsumption. Overall, enjoying drinks in a large social setting where the alcohol volume content can vary is a slippery slope to impaired judgment and serious accidents.

Even when making drinks in their own home, most consumers aren't skilled bartenders. This can also lead to similar issues discussed before such as adding more alcohol than intended, or even not enough. Finding the right balance of ingredients can be bothersome and dissatisfying when done incorrectly. Overall, high alcohol intake, cost, and consumer dissatisfaction are all key issues that can be avoided when using an automatic cocktail dispenser.

1.2 Solution

Our solution is to create an automated cocktail dispenser, one that can eliminate cost and complexity by introducing standardized ingredient portions, as well as providing different cocktail choices. The user will choose a cocktail from an online menu, then place the proper ingredients in the appropriate containers. Then, the user will select how much alcohol they want in their drink, and the correct quantities will be dispensed into a cup and mixed with a motorized stirring rod.

1.3 Visual Aid

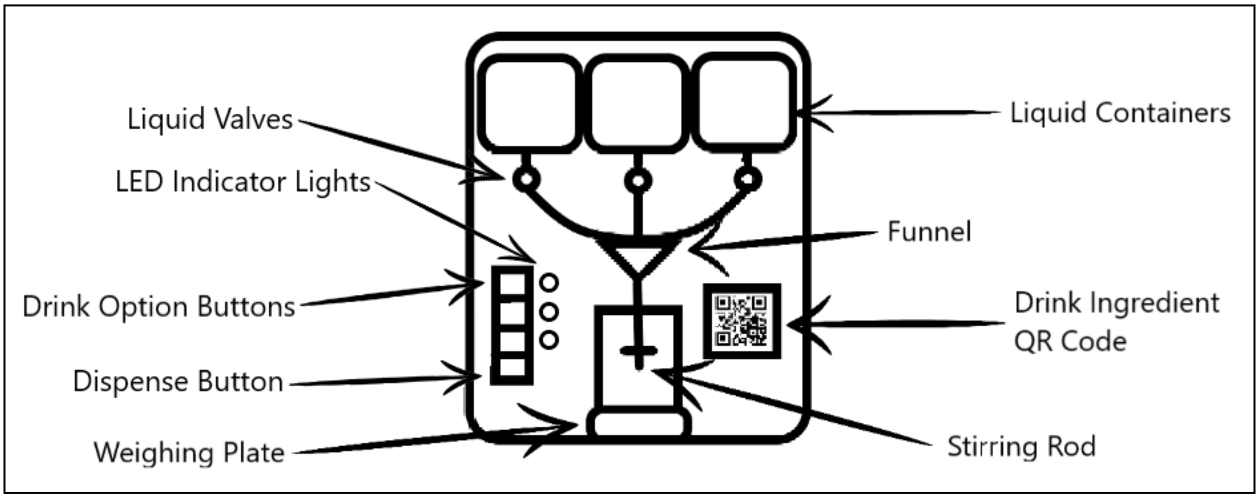

Figure 1: Physical representation of the dispenser

As seen in *Figure 1*, there are three containers that will each hold a separate liquid ingredient at the top of the design. The button panel on the side will be used to select how strong the dispensed drink will be. Each container has a connected tube, and each tube directs the liquid to a funnel that dispenses into the user's cup below. Additionally, the stirring rod pictured above will be inserted through the center of the funnel so that it is centered in the cup. The funnel hole diameter will be large enough so this doesn't cause any issues with the release of ingredients. Lastly, there will be a QR code placed on the design that can be scanned via any smartphone. This will bring the user to a webpage about the types and compositions of possible drinks.

For the buttons, there will be one button for a normal cocktail, one for a stronger, "double shot" cocktail, and one for a mocktail (no alcohol at all). Two of the containers will have alcohol, and one container will have a mixer (i.e. juice, soda, etc). A cocktail will consist of one shot from each of the two alcohol containers, and three shots from the mixer. A double-shot cocktail will contain two shots from each of the two alcohols, also with three shots from the mixer. The mocktail will contain just three shots of the mixer. For obvious reasons, we will be testing with food coloring-dyed water, rather than actual alcohol.

1.4 High-Level Requirements

1. The product must be able to dispense the user's chosen drink with no selection error - normal and double-shot cocktails have ingredients from all three containers, and the mocktail only has liquid from one container.
2. The product must dispense the correct quantity of each ingredient for each respective cocktail with ~10% accuracy, measured by mass (e.g. +/- 4.5g for 45mL, one shot, of water). The baseline masses for: mocktail is 135g, single shot cocktail is 225g, and double shot cocktail is 315g.
3. The stirring rod properly mixes the drinks after a successful dispense (test with red, blue, and yellow food coloring water in each container, to see if the final cocktail color is a uniform brown).

2. Design

2.1 Physical Design

Our design will use the following main components: three square plastic containers (with screw-on lids) to hold the liquids, three electrically controlled solenoids acting as liquid valves, one stirring rod controlled by a gear motor, and one load cell sensor to measure the quantity of liquid dispensed. As seen in Figure 1, the containers will be mounted above the solenoid valves, to use gravity, rather than pumps, to facilitate liquid transportation. Based on a button that the user presses, the appropriate solenoid valves will activate, one at a time, to dispense the proper amount of liquid into the cup. The load cell sensor will close the valves, once the appropriate liquid quantity is reached. Finally, the stirring rod will mix the drink after all liquids have been dispensed.

2.2 Block Diagram

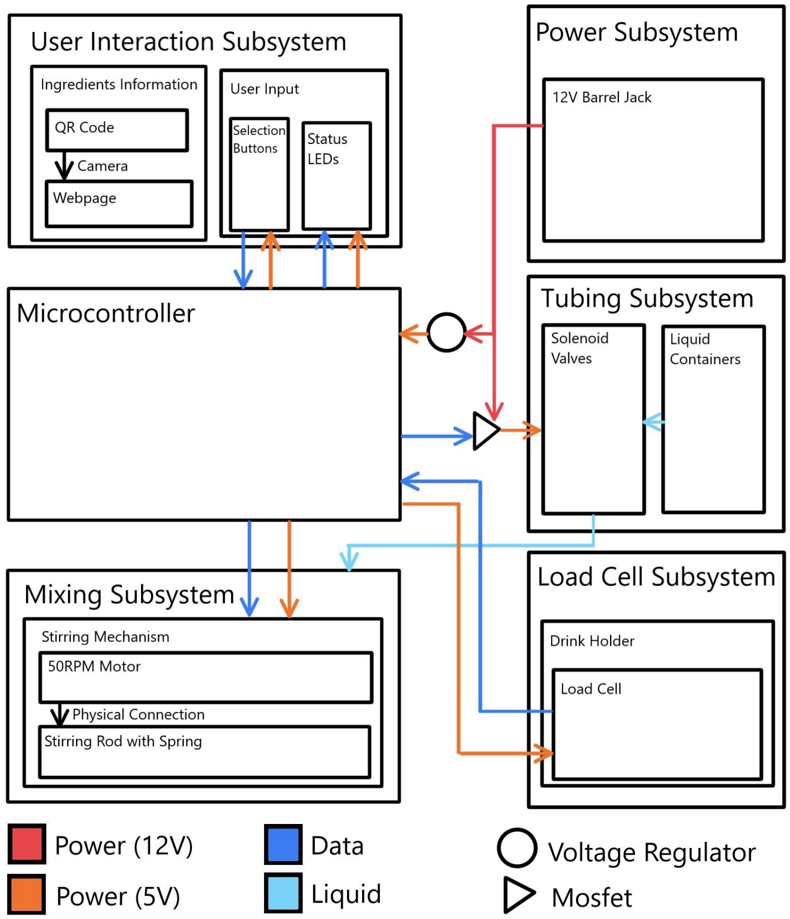

Figure 2: Automatic Cocktail Dispenser Block Diagram

2.3 Subsystems

2.3.1 User Interaction Subsystem

The User Interaction Subsystem is responsible for allowing users to choose the drink they want to dispense through push-button inputs and LED indicators showing their selection. There are going to be four push buttons in total: one for a single shot of alcohol, one for a double shot, one for a mocktail, and a dispense button. Each button has a voltage rating of 120V AC and a current rating of 3A. Each LED has a voltage rating of 2V and a current rating of 35mA. Only one button will be able to be selected at a time, and each button will have its own assigned LED, except for the dispense button. When the device is first turned on, the default alcohol content will be set to mocktail to ensure all users, regardless of age, are consuming a legal beverage.

Selecting alcohol content will need to be explicitly selected by the user to ensure we are following IEEE ethical standards.

Also included in this portion of the unit will be a scannable QR code. By using a compatible device (most modern smartphones), a user will be able to pull up a simple web interface, giving general information on the device as a whole, its functionality, and recommended ingredients for the three containers which reside in the tubing system. This web interface will be hosted through Google Cloud and built with simple html and css to reduce complexity.

Requirements	**Verification**
● Buttons can select drink potency.	● After the user clicks a selection button, the corresponding LED will be lit and the proper alcohol content should be dispensed through the tubing system. We will confirm this by using the mocktail as a baseline which should not activate the alcohol solenoids at all and have an approximate weight of 135g. For a single-shot cocktail, the solenoids should activate, adding ~90g overall, and a double-shot cocktail should add ~180g to the total weight, with a margin of error of 10% for the total weight.
● QR Code can be scanned and the website is accessible.	● After the user scans the QR code with a compatible device, they will be sent to our website. There should be no errors displayed to the user (404, 500, etc.) and the content of the website should be legible on any device.
● Only a single LED is lit at a time so more than a single option can't be selected.	● If we select multiple buttons at once, the lesser alcohol content will be selected, and only that LED will be lit.

	If a selection button and the dispense button are selected at the same time, the selection will be made but the device will not dispense unless the dispense button is selected on its own.
• The LED indicator is lit up according to the correct button press.	• When a user chooses one of the alcohol content selection buttons, the associated LED should be lit up. This should be nearly instantaneous and will confirm to the user that their selection has been registered on our microcontroller. There will be no LED misfires and one LED should always be lit.

Figure 3: User Interaction Subsystem - Requirements & Verification

2.3.2 Power Subsystem

The Power Subsystem will be responsible for supplying power to each of the electrical components of the design. 12V DC batteries will be used to power each of the electric solenoid valves and the gear motor that stirs the drinks. The load cell sensor, push buttons, LEDs, and microcontroller will be operating at 5V so we will need to step down the voltage of this device with a voltage regulator.

Requirements	**Verification**
• The electric solenoid valves respond to the supplied power and activate.	• Measuring the voltage across the solenoids should be within 10% of 12V. When supplied power, the valves should open and they should close again when returned to low voltage.
• The gear motor shows a response when supplied with power, and holds the correct RPM (50 RPM).	• Measuring the voltage across the motor should be within 10% of 12V. To test rpm, we will attach a rod or pencil to the motor and time it with a stopwatch for 12 seconds. If the motor is successfully reaching 50

	RPM, there should be 10 rotations back to the start in that amount of time.
• Microcontroller is supplied with the proper voltage (5V) specified in its spec sheet.	• We will use a multimeter to confirm our current/resistance calculations are correct and the device is being supplied the proper voltage (within 10% of 5V).
• Voltage regulator correctly steps down the voltage from 12V to 5V.	• Using the oscilloscope, we will probe the nodes before and after the regulator on the PCB and ensure the voltages are as expected.

Figure 4: Power Subsystem - Requirements & Verification

2.3.3 Microcontroller

The Microcontroller Subsystem is responsible for several pieces of data and for determining the next step in the process of dispensing a cocktail. This includes taking the data provided by the load cell sensor and deciding when to close a solenoid valve and open the next one. It is also responsible for taking the user's input to the push buttons and deciding whether it is time to start dispensing or not. It is crucial for this subsystem to decipher which button is being pressed and then start the process for the drink that has been selected. It also decides which valve opens at which time and ensures that only one operates at a time. Lastly, it controls when the stirring rod starts/stops to finish off the complete process.

Regarding physical connections, our microcontroller will have five assigned input pins and seven assigned output pins. For input, we will read the three drink potency buttons and the dispense button as well as a constant input from the weighing plate. The output will control the three indicator LEDs as well as the three solenoids and the stirring motor. The programming for this microcontroller will be done in a compatible language for our to-be-decided chip and the general flow will begin by awaiting a dispense button input from the user. If desired, the user can also make an alcohol potency selection through the option buttons, also changing the indicator LED; however, the default drink will be a mocktail. After the dispense button is pressed, the load cell and weighing plate will confirm that weight (the user's cup) is detected and dispensing will begin. The microcontroller will then send a signal to release the seal on the first solenoid, and once the user-specified weight of each ingredient is reached (single, double or mocktail), the solenoid will close and the next will open, repeating the process. After ingredients are dispensed, a stirring rod mounted within the funnel will activate, controlled by a 50 RPM motor and will run for the specified time of each drink. In order to prevent over-dispensing ingredients, the microcontroller will not accept input again from the dispense button until the weight plate has been zeroed out, signifying the user has removed their drink from the dispensing area. We will also prevent dispensing liquid if the weight is above that of our standard cup (within 5 grams), in case the user was to lift up their drink and set it back down on the plate.

The microcontroller being used for this design is the ATMega328P-P. It has an operating voltage of 1.8 - 5.5V and a current rating of 200mA in active mode.

Requirements	**Verification**
• Consistently receive data readings from the load cell sensor without any loss	• We will create a test program which uses the indicator LEDs as debug LEDs (or a separate debug LEDs) which will emit light when the load cell is detecting weight. We will continuously add weight to a cup on the load cell and the LED should stay lit the entire time.
• Correctly pick the order of ingredients to be dispensed.	• Solenoids should open in the order we expect and no two solenoids should ever be open at the same time. We can confirm this by viewing the flow of liquid (water dyed with food coloring) through our tubing system.
• Only allow the stirring rod to operate once the liquids are done dispensing	• While any solenoid is open the stirring rod should not be activated. The stirring rod should only activate after either the first (and only) ingredient is fully dispensed from the mocktail or all three ingredients are fully dispensed for either cocktail selection. Fully dispensed should mean no liquid

	is flowing through the tubing system, so this will require a slight delay in activation even after solenoids are closed. Liquid dripping from the valves is acceptable, but there should be no noticeable flow.
● Choose the correct quantity of liquid for the drink that was selected	● To test that the correct quantity of liquid was dispensed we will precisely measure the three options using our own measuring equipment. By pouring the dispensed drinks into a standard milliliter measuring cup, the three cocktails should fill 135ml, 225ml and 315ml respectively, within a margin of error of 10%.
● Accept user input and correctly recognize the drink selection	● Using a milliliter measuring cup, the button option for mocktail should dispense 135ml, single shot 225ml and double shot 315ml within a 10% margin of error. If any of these are consistently incorrect or flipped it is likely we have mounted the buttons and/or leds incorrectly and we will remedy this.
● The dispense button cannot be selected again when a filled drink resides on the weighing plate.	● This functionality should be limited in our microcontroller code, however, we will test it on the actual machine to confirm. We will dispense a drink and then try multiple options which may lead to failure. We will try removing and replacing the filled drink, changing the potency selection and multiple dispense button presses. As long as the drink resides on the plate, none of these should begin another dispensing cycle.

Figure 5: Microcontroller Subsystem - Requirements & Verification

2.3.4 Tubing Subsystem

The tubing subsystem is responsible for moving drink ingredients from their holding containers to the user's cup. The physical design will consist of three containers each with their respective tubing leading into a common funnel. Each of the three tubes will be opened and closed with a 12V DC electric solenoid, working as a water valve, allowing for precise dispensing. A large part of this subsystem will be built through the machine shop (tubing and fittings), using the containers and solenoids we provided. All functions of this subsystem will be controlled by the microcontroller, with a high voltage opening the solenoids and a low voltage closing them. When the solenoids open, ingredients will be transferred from their holding containers, using gravity, into a funnel combining all tubing above the user's cup. Solenoids will open in series rather than parallel in order to simplify the job of the weighing plate located in the mixing subsystem.

Requirements	**Verification**
• Only one tube dispenses at a time, so when one valve is open the rest are closed	• Through simple test code, we can instruct the valves such that each one cannot open unless all others are closed as well. Test examples can be run with water as the liquid in order to observe the behavior of the valves and ensure they work properly.
• No leakage from the tubing when liquids are being dispensed	• This can be confirmed through several examples of running liquids through the tubes and observing any holes or gaps that the liquid could pass through. This requirement will be fulfilled once there appears to be no leakage for a good stretch of test examples (say, 20 test runs).
• Tubing is stable and remains firm while ingredients are being dispensed.	• Tubing should not shake or move an unreasonable amount while dispensing liquids. The accepted tolerance will depend on the type of tubing used by the machine shop, and we will verify

	this by observing movement (or lack thereof) over multiple test trials.

Figure 6: Tubing Subsystem - Requirements & Verification

2.3.5 Mixing Subsystem

The mixing subsystem is responsible for stirring the container full of liquid once all the ingredients have been dispensed. Once the ingredients have been dispensed, the microcontroller will power the 50 RPM gear motor with a rod attached to it and begin spinning for a predetermined set of time and then cease operation. The gear motor operates at 12V DC and has a current rating of 600mA.

Requirements	**Verification**
• The stirring rod starts operation once the last ingredient is fully dispensed (The last valve is closed).	• Stirring rod should not begin activation until all ingredients are fully dispensed. There should be a clear indication from the tubing system that all valves are closed before its activation (liquid is no longer leaving the containers) and a slight delay should occur between the closing of solenoids and activation of the motor to allow excess liquid in the tubing to enter the user's cup.
• Stirring rod operates for the correct amount of time.	• Stirring rod should run for the amount of time specified for the drink selection. We will confirm this by timing the activation of the motor with a stopwatch and its timing should remain accurate within 1s.
• Stirring rod does not interfere with the funnel it sticks through as well as the ingredients as they are being dispensed.	• This can be confirmed by running several test examples of dispensing the liquid through our funnel/stirring obstacle and keeping track of how many times there is any spillage. Adjustments can be made to the

	stirring rod when necessary to ensure the requirement is met.
• Stirring rod does not disturb the balance of the cup while in operation.	• This can be confirmed by having several examples of a sample cup full of liquid being stirred. Varying the speed of the motor and ensuring that 50 RPM is a reasonable rate should confirm this requirement.
• Cocktail is fully mixed	• Since clear cups are being used, one can see the concoction before and after the mix. Water with food coloring will be used to test this, and pictures will be taken before the drink is stirred and after as well. If the drink is the expected homogenous color after the mixing, the requirement is fulfilled. For example, mixing red, blue, and yellow liquid should result in a brown color.

Figure 7: Mixing Subsystem - Requirements & Verification

2.3.6 Load Cell Subsystem

The load cell subsystem is responsible for measuring the amount of liquid dispensed into the container and sending the accurate data to the microcontroller. The load cell will detect when the first ingredient has been dispensed enough, send this data to the microcontroller which will then instruct the solenoid to close, and then the next ingredient will dispense. Once again, the load cell will relay information to the microcontroller when enough of this ingredient is dispensed. This sensor will operate at approximately 5V.

Requirements	**Verification**
• The load cell operates at the specified operating voltage (2.6-5.5 V).	• Apply 5V at the VCC pin and observe whether the load cell reads data at this voltage.

• Accurately provides data to the microcontroller with 5% accuracy.	• With an Arduino, write test code that prints load cell reading to the console. • With object's of known weight, place them on the laid cell sensor and check that the values match.
• Load cell input and output resistance is within 5% accuracy of specified calibration in the datasheet.	• A load cell has 4 wires, two sets of positive and negative terminals. Connect an ohmmeter to measure the resistance of both pairs of wires (a positive + negative pair), and compare the values.

2.4 Tolerance Analysis

The most important part of our project is the tubing subsystem. Since our whole project is about selecting which liquids to mix together, without a way to transport the liquids, there would be nothing to mix. Ensuring the correct amounts are dispensed, as well as preventing leaks, are crucial requirements that could lead to immense risk if not performed correctly. Dispensing too much alcohol or leaking liquid over the design poses the greatest dangers, so using extra caution when implementing these parts will be extra important to ensure the safety of the users as well as the creators.

By ensuring that all liquids dispensed are within 10% of one shot, it ensures that total alcohol consumption stays low enough, even when it is +10%. Since there are two shots per normal cocktail, if we have 3 cocktails, that is 2 shots * 1.1 * 3 drinks = 6.6 shots, which is only 0.6 shots more than intended. If we increased the tolerance to 20% of one shot instead, also having 3 cocktails, that would be 2 shots * 1.2 * 3 drinks = 7.2 shots. An entire extra shot, unintentional or not, could mean the difference between staying below the legal BAC (blood alcohol content) level.

Another important component to consider is the electric motor of the mixing subsystem. If the motor draws too much current, it could damage the PCB. For our project, we will be using water with food coloring, the latter of which will not significantly change the liquid, so all the liquid will be the same density and viscosity.

On startup, the motor draws more current than normal, drawing 0.19A. After the motor is smoothly running, it draws a consistent 0.165A. The motor's maximum current rating is 0.6A, and our PCB is designed with a 0.15mm minimum track width, which is rated for 0.5A. Even if the load spontaneously increased somehow, we are currently only at 38% of the max current rating for our PCB, so any spikes in current will be well within this range and not damage the PCB.

3. Cost and Schedule

3.1 Cost Analysis

As seen in the figure below, the total cost in parts before tax ends up being $138.17. Assuming free shipping since most items were ordered from Amazon, and a sales tax of 6.25%, the total cost of components equals $146.81. Assuming an hourly rate of $37 per hour for each partner, $37/hour x 2.5 x 100 hours = $9250 each. Totaling for each partner, the labor cost comes out to $27,750. In terms of overhead costs, electricity costs 13 cents/kWh and the time of operation per day could be approximated to 30 min per day. Using this, energy consumption comes out to 0.012kWh which is $0.00156 per day. Assuming the design is used three times a week for a year, the total overhead costs comes out to be $0.24. The grand total of parts, labor, and overhead costs comes out to be $27,897.05.

Description	**Manufacturer**	**Quantity**	**Extended Price**	**Link**
Bolsen Digital Load Cell Weight Sensor 1KG/5kg/10kg/20kg Portable Electronic Kitchen Scale + HX711 Weighing Sensors Ad Module for (1kg)	Bolsen Tech	1	$8.29	Link
HFS (R) 12v Dc Electric Solenoid Valve Water Air Gas, Fuels N/c - 1/4IN NPT Available (12V DC 1/4IN NPT)	HFS (Hardware Factory Store)	3	$41.13	Link
Greartisan DC 12V 50RPM Gear Motor High Torque Electric	Geartisan	1	$14.99	Link

Micro Speed Reduction Geared Motor Centric Output Shaft 37mm Diameter Gearbox				
Beer Nuts Bar Mix Containers	Beer Nuts	3	$26.37	Link
Solo Clear 10 Ounce Plastic Cups, 36 Count	Solo	1	$10.99	Link
Push Button	E-Switch	4	$14.52	Link
LEDs	Visual Communications Company - VCC	3	$8.01	Link
IC REG LINEAR 5V 1.2A SOT223	STMicroelectronics	1	$1.16	Link
AC/DC WALL MOUNT ADAPTER 12V 12W	XP Power	1	$12.71	Link

Figure 8: Component Costs

3.2 Schedule

Week	Date	Group Tasks
6	February 20th - February 27th	• Deliver parts to the machine shop (Carson and Ben) • Start PCB Research (Carson and Caleb) • Find and order push buttons (Ben) • Find and order LEDs (Ben)
7	February 27th - March 6th	• Begin PCB Design (Carson) • Test electric solenoid valves via breadboarding (Caleb) • Discuss, decide on, and order a microcontroller (All) • Begin testing load cell sensor (Ben)
8	March 6th - March 13th	• Finish PCB Design (Caleb, Ben) • Get buttons working (Carson) • Machine shop revisions (All) • PCB ORDER MARCH 7th
9	March 13th - March 20th (Spring Break)	• Spring Break
10	March 20th - March 27th	• Integrate load cell sensor into design and continue testing (Ben) • Integrate all solenoids into the design and

		continue testing (Caleb) ● Ensure gear motor works as intended (Carson) ● Revise PCB (Carson)
11	March 27th - April 3rd	● Finish and debug prototype #1 (Caleb) ● Revise PCB Design (Carson) ● Design webpage for drink information (Ben) ● PCB ORDER MARCH 29th
12	April 3rd - April 10th	● Finalize design assembly (All) ● Assemble prototype #2 (All)
13	April 10th - April 17th	● Debug prototype #2 (All) ● Prepare for final presentation (All) ● Mock Demo (All)
14	April 24th - April 28th	● Demo
15	May 1st - May 5th	● Final Paper

Figure 9: Schedule for Project Progression

4. Ethics and Safety

Our project does not pose significant ethical or safety concerns, though it is important to note that, while dealing with liquids, we must ensure that they will not directly interact with any circuitry. A mix of the two could cause a short circuit leading to an electrical shock or fire, which is a significant health risk to the user of the device. To avoid these concerns, all circuitry will be mounted away from any dispensing portions of the unit, and buttons, which may be mounted closer, will be ensured to be waterproof. Continuing, as we will be dealing with 12V Alkaline batteries for this project we must ensure they come from a reliable brand that upholds its own safety and ethical standards. If we were to use non-reliable batteries, we may run into overheating or corrosion issues which pose a risk to the user of the device. Additionally, we must ensure that the batteries are easily user replaceable, as even reliable Alkaline battery brands may begin corroding given the correct amount of time. Giving the user easy access to replace the battery will prevent this issue and will allow the device to safely function for many years to come.

On a separate note, as the recommended ingredients on our webpage may include alcoholic substances, it is crucial that we notify the user of the health risks and laws regarding the consumption of the recommended product. In order to uphold the standards of the IEEE Code of Ethics, which includes "[disclosing] factors that might endanger the public" and “[avoiding] unlawful conduct” [3] we will ensure that standard surgeon general warnings are included on our website in an obvious location.

5. Citations

[1]https://home.binwise.com/blog/alcohol-pricing#:~:text=Most%20restaurants%20are%20aiming%20for,is%20between%20%245%20and%20%2415.
[2]https://www.arrow.com/en/research-and-events/articles/basic-battery-safety-very-different-chemistries-very-different-concerns#:~:text=There%20are%20two%20risks%20with,and%20especially%20to%20the%20eyes.
[3]https://www.ieee.org/about/corporate/governance/p7-8.html