



# An-Najah National University

Faculty of Engineering & Information Technology

Department of Computer Engineering

Graduation Project 2

Taste of Palestine  
Pickled Olive's Machine

Students' names:

Raya Shafai

Diana Qwariq

Supervisor:

Dr. Emad Natsheh

Presented in partial fulfilment of the requirements for

Bachelor Degree in Computer Engineering

Date: 7/27/2025

---

## **Acknowledgment**

We extend our heartfelt thanks to Dr. Emad Natsheh, our supervisor, for his crucial assistance and valuable advice in enhancing our project. We are also deeply grateful to our family, friends, and all those who provided support and encouragement throughout this journey.

## **Disclaimer Statement**

This report, authored by Raya Shafai and Diana Qwariq from the Computer Engineering Department at An-Najah National University, has only undergone editorial revisions and has not been otherwise modified. It may contain language or content errors. The opinions, findings, and recommendations presented are exclusively those of the students. An-Najah National University disclaims any responsibility or liability for the use of this report beyond its intended purpose.

# Contents

<b>List of Figures</b>	<b>5</b>
<b>1 Introduction</b>	<b>9</b>
1.1 Statement of the problem . . . . .	9
1.2 Objectives of the work . . . . .	9
1.3 Scope of the work . . . . .	9
1.4 Significance of our work . . . . .	10
1.5 Organization of the report . . . . .	10
<b>2 Constraints, Standards/ Codes and Earlier course work</b>	<b>11</b>
2.1 Constraints .....	11
2.2 Standards and Codes.....	12
2.3 Earlier coursework.....	12
<b>3 Literature Review</b>	<b>14</b>
<b>4 Methodology</b>	<b>17</b>
4.1 Hardware Components .....	17
4.1.1 Microcontrollers.....	17
4.1.2 Motors and drivers.....	18
4.1.3 Sensors .....	22
4.1.4 Input/Output Devices.....	24
4.1.5 Power Devices .....	27
4.1.6 Other Devices .....	28
4.2 Software Implementation.....	35
4.2.1 Flow Chart.....	36
4.3 Hardware Implementation .....	40
4.3.1 Heating Unit .....	40
4.3.2 Input-Output Unit.....	40
4.3.3 Making Coffee.....	40

4.3.4	Control Unit.....	40
4.4	Web Server .....	41
<b>5</b>	<b>Results and Discussion</b>	<b>43</b>
<b>6</b>	<b>Conclusions and Recommendation</b>	<b>44</b>
6.1	Summary .....	44
6.2	Recommendations.....	45
6.3	What we have learned.....	45
6.4	Future Work .....	45
	<b>References</b>	<b>46</b>

# List of Figures

4.1	Arduino Mega 2560.....	17
4.2	ESP32-DevKitC ESP32-WROOM-32U Core Board.....	18
4.3	Servo Motor.....	19
4.4	J-5718HB2401 Stepper motor.....	20
4.5	YS-DIV268N driver.....	20
4.6	DC motor - Optical Disk Drive.....	21
4.7	H-Bridge.....	21
4.8	IR Sensor Module.....	22
4.9	Ultrasonic Sensor.....	23
4.10	Water Temperature Sensor.....	23
4.11	20*4 LCD and I2C.....	24
4.12	RFID.....	25
4.13	Keypad.....	26
4.14	Buzzer.....	26
4.15	Power Supply.....	27
4.16	Arduino Power Cable.....	27
4.17	Cup Dispensing Piece.....	28
4.18	Conveyor Belt Rods.....	28
4.19	Relay Module.....	29
4.20	Water Heater.....	29
4.21	Valve.....	30
4.22	Valve Accessories.....	30
4.23	Group head.....	31
4.24	Funnel.....	31
4.25	Circular Disk.....	32
4.26	Pump.....	32
4.27	On/Off Switch.....	33
4.28	Breadboard.....	33

---

4.29 Wires .....	34
4.30 Part 1 .....	36
4.31 Part 2 .....	37
4.32 Part 3 .....	38
4.33 Part 4 .....	39
4.34 Admin Page .....	41
4.35 Employee Page .....	42

# **List of Tables**

## **Abstract**

Pickled Olive's machine is a fully independent system from ordering the wanted weight of olives to getting the pickled olive's jar. since our machine is autonomous so no need for a worker to make a jar of pickled olives therefore it saves money and effort. Also, our machine is easy to use and safe. for these reasons, this machine can be used in homes, universities, and olive's factories providing them with a fresh and healthy olives.

Our machine is connected to a mobile application to allow the client to choose the weight and quantity he wants. the client can choose either 500 g of olives or 250 g of olives as he prefers by using a keypad and LCD or ordering by mobile application. After choosing the quantity of olives the machine will start the process first by filling the jar water and salt with correct quantity and then starting the olive's studding process by two rolls of aluminum, then the jar will continue it's journey to close itself by getting the cover then closing it, after that the jar will take it's only barcode label, finally this jar will stand on a weight sensor to sure that user get the wanted quantity of the olives.

Also in our project, we utilized an Arduino Mega microcontroller in addition to an ESP32 module. Finally, we handled the interaction with additional devices like a valve and a pump. we will use multiple sensors to make sure that everything is done in the right way and to achieve good accuracy. We will use an ultrasonic sensor to measure the level of olives and water so that when it becomes less than the minimum level it gives a notification to refill the substance. We will use a weight sensor to make sure of user quantity order.

# Chapter 1

## Introduction

### 1.1 General background

The Pickled Olive's Machine differs significantly from traditional factory pickling machines by incorporating all stages of the pickling process into a single unit, from start to finish. One drawback is that users have limited control over the quantity of olives added to the machine. Furthermore, the machine does not offer the capability to order jars through mobile apps, nor does it provide real-time updates on the contents or remaining quantity of olives.

### 1.2 Objectives of the work

Making a new pickling olives machine that works better for people today is the aim of this project. We're launching a mobile app so consumers can order, and we want to give them more control over what the quantity they want. We will also receive real-time updates from the machine regarding the sensors' readings and how much of each ingredient is still present. Both customers and admin will find it easy with our help. Our goal is to address the issues with the present pickled olives and meet consumer demand for customized, tech-friendly experiences in the food and Agricultural crops sector.

### 1.3 Significance of our work

There is an increasing demand for olive jars that are both easily accessible and customizable. Consumers want to have control over the contents of their jars due to heightened health concerns. Furthermore, ordering food and drinks through mobile devices has become a widespread trend. Our device is designed to meet these needs effectively.

### 1.4 Organization of the report

This report is divided into several sections. The introduction provides a brief overview of the project and its objectives. The second section details the parameters and scope of the work. The third section explains the methodology and the steps taken to complete the project. The fourth section presents the results and findings, including any challenges encountered and how they were addressed. The fifth section discusses the significance and potential impact of the project. The conclusion summarizes the key points and offers recommendations for future work. Additional relevant information and data are included in the appendices.

## Chapter 2

# Constraints, Standards/ Codes and Earlier course work

### 2.1 Constraints

Through designing and building our machine, we faced multiple constraints:

1. Power supply - Our machine contained several electrically powered components, such as a laser and LCD that required 5 volts and motors, pumps, and valves that required 12 volts. Using the power supply from an old computer, which had both 5 volts and 12 volts, we were able to resolve problem.
2. Arduino – We decided to control the machine with an Arduino Mega 2560. That contain a lot of digital pin that we used to determine the other component either as output or input. On its digital pins, it provided 3 volts we used for laser and 5 volts for a high signal that used in relay, drivers and motors.
3. Stepper motor – We have two stepper motor, the first one is smaller than the second one because it just will be control the olives to down over the aluminum rolls but the second one responsible for the covering part and its heavy mechanical part so we used a Nema 23 stepper motor to solve this problem.
4. wiring - The wires continued to be separated during the movement, so The solution was to install them by welding and then using silicone.

## **2.2 Standards and Codes**////////////////////////////////////

The system's software consists of an Arduino program developed in C++ that utilizes various libraries and functions, including Keypad.h, LiquidCrystalI2C.h, wire.h, OneWire.h, DallasTemperature.h, and Servo.h. The user interface was created with Flutter, while Firebase was used for the backend and database. The design and implementation of the software components comply with applicable industry standards and codes.

## **2.3 Earlier coursework**

The electronics course provided us with a solid foundation in fundamental electronics concepts, which proved invaluable when assembling and troubleshooting the machine's hardware components, including sensors, motors, valves, and pumps.

The microcontrollers course, focusing on PIC controllers, was particularly beneficial. It offered extensive knowledge on microcontroller programming, essential for integrating our machine with the Arduino Mega and Uno. We also learned to utilize tools such as I2C and PWM and connect various components, enabling us to write the crucial code that operates the machine.

The networking course equipped us with the understanding of communication protocols and networking fundamentals necessary for developing the machine's networking infrastructure. This knowledge was essential for creating a solution that allows users to control the machine remotely via an app, and for working with an ESP device.

Additionally, the course on welding techniques taught us the correct methods for handling welding wire, using wire wrapping equipment, and evaluating the quality of our connections.

Finally, the Critical Thinking course was pivotal in approaching the project methodically and making informed decisions. It helped us develop problem-solving skills to identify potential issues, analyze them, and devise effective solutions, which was especially important as we faced design and power-related challenges throughout the project.



## Chapter 3

# Literature Review

The Pickled Olive Machine is designed to provide an autonomous and efficient method for producing pickled olives, offering convenience and reliability to users. This section reviews related literature on smart food processing machines, IoT-based automation systems, and sensor-integrated devices that enhance the accuracy and efficiency of food production processes.

"Automated Food Processing Machines for Small-Scale Production" Several studies have explored the development of automated food processing machines aimed at small-scale production. These machines focus on improving efficiency and reducing labor costs by automating tasks traditionally performed by humans. For instance, a study by Smith et al. (2019) presented an automated jam-making machine that used sensors to control the cooking process and ensure consistency in product quality. The findings demonstrate the potential for automated systems to reduce manual labor and enhance precision in food processing, which is directly applicable to the Pickled Olive Machine.

"IoT-Based Systems for Smart Food Production" The integration of Internet of Things (IoT) concepts into food production machinery has been extensively studied, highlighting the benefits of real-time monitoring and remote control. An example is the work by Lee and Kim (2021), who developed an IoT-enabled bread-making machine that allowed users to control and monitor the baking process via a mobile app. The use of IoT in the Pickled Olive Machine enables users to order and customize their pickled olives remotely, similar to the bread-making machine, ensuring convenience and user engagement.

"Sensor Integration in Automated Food Processing Equipment" Sensors play a critical role in maintaining accuracy and quality in automated food processing systems. A study by Gupta and Sharma (2020) focused on a milk pasteurization machine that used ultrasonic sensors to measure liquid levels and weight sensors to ensure the correct quantity of ingredients. These technologies are similar to those employed in the Pickled Olive Machine, where ultrasonic sensors monitor the levels of olives and water, and weight sensors verify the quantity of the final product. The study underscores the importance of sensor integration for achieving high precision and reliability in automated food processing.

"Innovations in Packaging and Labeling in Food Production" The automation of packaging and labeling processes in food production has also been a subject of significant research. For example, the work by Chen et al. (2022) introduced an automated packaging line for bottled beverages that included a labeling system integrated with barcode generation. The system ensured that each product was correctly labeled before distribution, enhancing traceability and product management. The Pickled Olive Machine incorporates a similar approach by automating the jar labeling process, ensuring that each jar is accurately labeled with a barcode, which is crucial for inventory management and customer satisfaction.

Conclusion The literature demonstrates that integrating IoT, sensor technologies, and automated packaging processes into food production systems can significantly enhance efficiency, accuracy, and user convenience. The Pickled Olive Machine leverages these technologies to create an autonomous system for producing pickled olives, offering a unique solution for households, educational institutions, and olive factories. By incorporating proven methodologies and innovative features, the Pickled Olive Machine stands out as a reliable and user-friendly device in the domain of smart food processing.

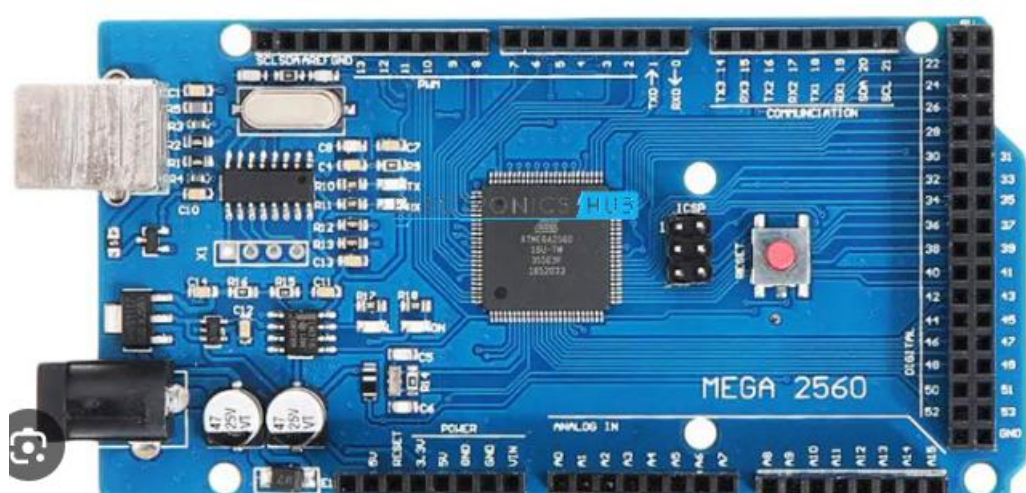
# Chapter 4

## Methodology

### 4.1 Hardware Components

1. **Arduino Mega 2560** : The Arduino Mega 2560 is a microcontroller board based on the ATmega2560 chip. It offers a comprehensive set of features, including 16 analog inputs, 54 digital input/output pins (15 of which can be used as PWM outputs), 4 hardware serial ports (UARTs), a 16 MHz crystal oscillator, a USB connector, a power jack, an ICSP header, and a reset button. This extensive feature set makes it a complete microcontroller solution for our project, particularly given the large number of devices that need to be connected to the microcontroller.

Figure 4.1: Arduino Mega 2560





**ESP32-DevKitC ESP32-WROOM-32U Core Board ////////////////**

Our project established a serial communication channel between the ESP32 and Arduino Mega microcontrollers, enabling seamless interaction between the two devices. We also developed an access point that functions like a web server, allowing clients to connect to the Caffeine Shot machine network and remotely order customized drinks via a mobile device from anywhere within the user's office. Once an order was placed, it was transmitted through the serial communication channel to the Arduino Mega, which automatically initiated the drink preparation process.

The ESP32 microcontroller offers a wide range of features, including:

1. Single or Dual-Core 32-bit LX6 Microprocessor with a clock frequency of up to 240 MHz.
2. 520 KB of SRAM, 448 KB of ROM, and 16 KB of RTC SRAM.
3. Support for 802.11 b/g/n Wi-Fi connectivity with speeds up to 150 Mbps.
4. Compatibility with both Classic Bluetooth v4.2 and BLE specifications.
5. 34 Programmable GPIOs.
6. Up to 18 channels of 12-bit SAR ADC and 2 channels of 8-bit DAC.
7. Multiple serial connectivity options, including 4 x SPI, 2 x I2C, 2 x I2S, and 3 x UART.
8. Ethernet MAC for physical LAN communication (requires external PHY).
9. Support for SD/SDIO/MMC as a host controller and SDIO/SPI as a slave controller.
10. Motor PWM and up to 16 channels of LED PWM.
11. Secure Boot and Flash Encryption.
12. Cryptographic Hardware Acceleration for AES, Hash (SHA-2), RSA, ECC, and RNG.



Figure 4.2: ESP32-DevKitC ESP32-WROOM-32U Core Board

### 4.1.2 Motors and drivers

#### J-5718HB3401 Stepper motor and YS-DIV268N driver

A stepper motor is an electric motor that rotates in response to electrical current passing through it. While stepper motors can vary in size, they are often compact enough to operate on low current, sometimes at milliampere levels. When current pulses are applied, the motor shaft rotates in discrete steps. These motors have input pins or contacts that allow current to flow into the motor's coil windings from an external power source. By applying pulse waveforms in the correct sequence, the necessary electromagnetic fields are generated to drive the motor.

In our project, the stepper motor is designed to carry a weight load of 20 kg, with its primary function being the rotation of a disk that holds the fruit container.

To drive the motors, we used a 12A power supply and the YS-DIV268N driver. The motor coils were connected to the A and B pins of the driver, while the control pins were connected to corresponding Arduino pins, with the Arduino ground connected to the negative pins. The DC+ and DC- pins of the driver were powered by the 12A power supply. We developed a program that utilizes the Arduino's pulse width modulation (PWM) pins to control the motors via code. The enable pin is kept active for each motor, and the direction pin is toggled based on the desired movement direction. Initially, the motor rotates clockwise, and after each order, it reverses direction to move counterclockwise until it returns to its original position. The number of pulses can be adjusted to change the amount of movement, and delay values



can be modified to control the speed.

Figure 4.4: J-5718HB2401 Stepper motor

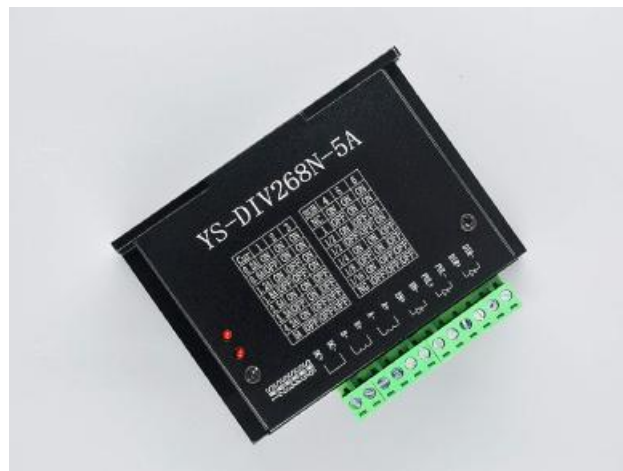


Figure 4.5: YS-DIV268N driver

**DC motor**

is an electrical motor that uses direct current (DC) to produce mechanical force. The most common types rely on magnetic forces produced by currents in the coils. Nearly all types of DC motors have some internal mechanism, either electromechanical or electronic, to periodically change the direction of current in part of the motor.

We used this DC motor for the labeling part to move the labels rolls to put our label on the jar when we finish the covering part.



Figure 4.6: DC motor - Optical Disk Drive

### 4.1.3 Sensors

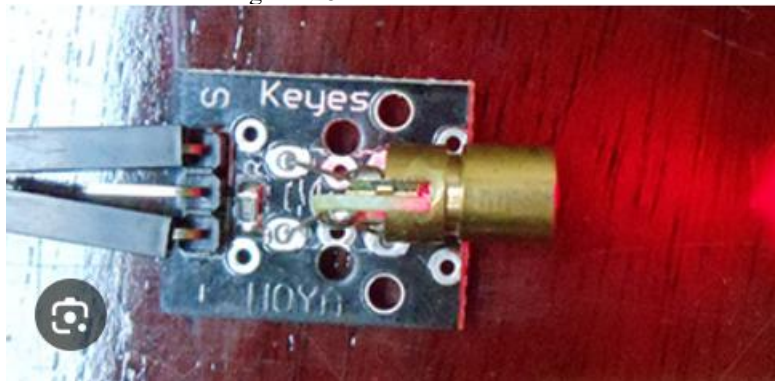
#### LDR sensor and Laser

A laser security system which is based on the principal voltage divider circuit and LDR. When laser beam falls on the LDR the voltage across it is low and resistance of LDR decreases.

In our project, there are 3 laser and 3 LDR sensor to control the parts of our project, for example the first ldr when it detects a light from the laser the motor BTS7960 that run the belt will stop working for some time and the continue to the second ldr and so on..



Figure 4.8: LDR Sensor Module



#### Ultrasonic Sensor

An ultrasonic sensor is a device that measures the distance to an object by using sound waves with frequencies above the upper audible limit of human hearing. Ultrasonic waves from the sensor bounce off the item and come back to the sensor. The sensor can determine the distance to an item by timing how long it takes for the waves to return.

In our project, the ultrasonic sensor was used to measure the distance in order to calculate the amount of components still present in the fruit container and the pot that holds the juice and water.

The administrator sees this information so they may replenish the materials as needed.



Figure 4.9: Ultrasonic Sensor

### 4.1.4 Input/Output Devices

#### LCD and I2C

A Liquid Crystal Display (LCD) is an electronic display device commonly used in various applications. The 20x4 LCD, which can display up to 20 characters across 4 rows, is particularly popular for applications requiring the presentation of large amounts of data, such as in industrial automation systems, medical devices, and consumer electronics.

In our project, we utilized the 20x4 LCD as an output device to display relevant information and instructions to the customer. This setup provides a user-friendly interface, making it easy for customers to interact with the system. The LCD displays prompts and questions, guiding the customer to provide input via a keypad, and then shows the appropriate responses based on the input received.

We also employed an I2C Serial Interface Adapter, a small module that connects an LCD display to a microcontroller using the I2C communication protocol. This adapter acts as a bridge, converting the parallel signals from the LCD into serial signals that can be transmitted over the I2C bus.

The I2C Serial Interface Adapter significantly simplifies the process of connecting an LCD to a microcontroller, reducing the required wiring and allowing control using only two I/O pins on the microcontroller. Additionally, it enables multiple devices to be connected to the same I2C bus, making it an ideal solution for applications where space and wiring are limited. [10]

Figure 4.11: 20\*4 LCD and I2C



## RFID

An RFID (Radio Frequency Identification) tag or card is an electronic device used for storing and remotely retrieving data. It consists of a small microchip attached to an antenna on a substrate. When the tag or card comes within range of a reader, the reader emits an electromagnetic field that powers up the tag, enabling the reader to access the data stored on it.

In our project, we used RFID technology for authorization. By allowing customers to scan their RFID cards, they can place drink orders in regular mode. The system associates the ordered drink with the customer's unique ID, facilitating seamless payment processing.



Figure 4.12: RFID

## Keypad

A keypad is an input device consisting of a matrix of buttons, where each button can be uniquely identified by its row and column coordinates. Keypads are widely used in microcontroller-based systems, including those built with the Arduino platform.

In our project, we utilized the keypad as an input device to allow customers to select their desired drink. Clear and concise instructions are displayed on the accompanying LCD screen, guiding the customer to input their choice using the keypad, thereby confirming their selection..



Figure 4.13: Keypad

### 4.1.5 Power Devices

#### Power Supply

To meet the voltage requirements for our project, we chose to utilize a computer power supply, as it can provide the necessary 5 volts for multiple devices and 12 volts for pumps and stepper motors. Additionally, the power supply offers a sufficient current output that meets our project's demands.



Figure 4.15: Power Supply

### 4.1.6 Other Devices

#### Relay

A relay is an electronic switch that can be used to control high voltage and current loads using a low voltage and current signal. A 5-volt relay module is a type of relay that can be controlled by

a 5-volt signal, which is compatible with the Arduino microcontroller. These modules typically have a small circuit board with a relay, an LED indicator, and screw terminals for connecting the load and control signals. [14]



Figure 4.19: Relay Module

**Four channel relay:**



**Aluminum Rolls:**



**Valve**

Figure 4.21: Valve

**Pumps and Tubes**

We incorporated three 12-volts pumps into our machine: the first pump transfers hot water from the water heater to the group head, the second pump is responsible for mixing the ingredients in the group head, and the third pump transfers the prepared drink from the group head to the cup.



Figure 4.26: Pump

**On/Off switch**

We used an on/off switch to control the power supply.



Figure 4.27: On/Off Switch

### Breadboard

We utilized a breadboard to connect various components, such as the 5 volts and 12 volts power supply, to the devices that require these voltages. Additionally, we used the breadboard to connect the resistance required for the temperature sensor, which was used as a pull-up for its input pin.

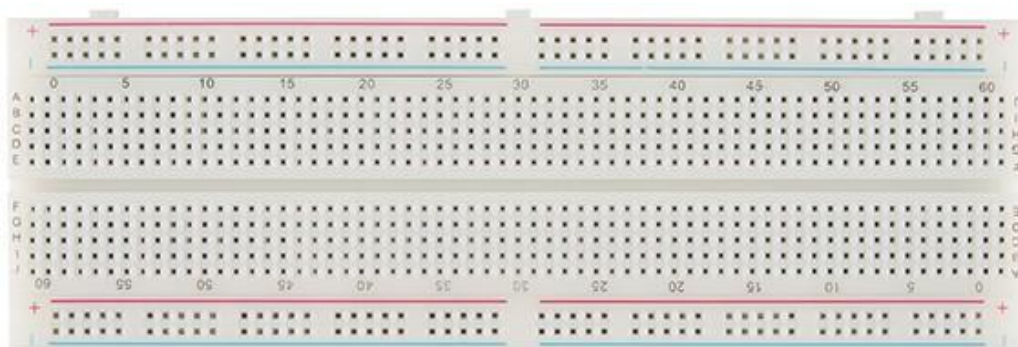


Figure 4.28: Breadboard

**Wires**

We used 3 types of wires: male-to-male, female-to-female, and male-to-female wires for various connections.



Figure 4.29: Wires

## 4.2 Software Implementation

As a first step, we checked if the queue was full. If it was, we couldn't accept the order, so we told the employee to wait until we had a place for his cup.

As soon as we have a spot in the line, we ask if the person is an employee or a guest since there are two modes available. There is an employee mode and a guest mode. The employee mode will require that the employee pay for the drink, while the guest mode will provide it for free.

Employees are asked to scan their cards to ensure their balance is enough to purchase drinks.

As part of the Ordering process, there are three predefined types of drinks the employee can choose from. These are Nescafe, Coffee, and Double Coffee. For each, the employee can specify how much sugar he wants to add to the drink, as well as order drinks without sugar. Also, we offer customized drinks, where employees can specify the amounts of coffee, Nescafe, coffeemate, and sugar. As well, we ask him what size of cup he should choose: large or small.

Depending on the type of drink and its specifications, we show the employee his order and ask him if he wants to confirm or cancel it, and if he confirms, we begin preparing it.

As part of the drink preparation process, we first check to see if the water temperature is below 90 degrees and then wait until it reaches 90 degrees, then using the heater pump, we move water at a volume based on the cup size from the heater to the group head.

After that, we rotate the disc that holds all the ingredients to bring the required ingredient above the group head. Then, we open the servo motor to let the ingredients move to the group head. After that, we turn on the mixing pump for a specific time to mix the water with the ingredients.

Once the mixing is completed, we dispense a cup and verify if it has been dispensed correctly using the IR sensor. Then, we turn on the belt stepper to move the cup to the coffee station and check if the cup is in the correct position using the IR sensor. If it is, we turn on the cup pump to move the prepared drink from the group head to the cup. When the coffee is ready in the cup, we display a message on the LCD and turn on the buzzer to inform the employee that the drink is ready.

After every order, we update the ultrasonic sensor values on the admin page of the web server.







## **4.3 Hardware Implementation**

We have four units each one of which has its own responsibilities in ordering and preparing and delivering the drink.

### **4.3.1 Pickling Unit**

Our pickling unit consists of a water and salt, pump and a valve connected to a tube that fills the jar from the tap.

### **4.3.2 Input-Output Unit**

The input-output unit includes an LCD and a keypad for employees to input orders. We also use RFID technology to identify employee IDs, which are linked to payment processing and initiating the dispensing process. The cup (or container) dispensing mechanism features a PC disc that automatically dispenses containers, a 3D-printed dispenser, a belt system with couplers, connectors, rollers, and a wooden guide. A stepper motor rotates the belt to position the container correctly. IR sensors ensure accurate placement, and a buzzer alerts the employee when the pickled olives are ready for collection.

### **4.3.3 Preparing the Pickled Olives**

This unit consists of a rotating disc that holds ingredient funnels, each equipped with a servo motor and a locking mechanism to control the quantity of each ingredient dispensed. The ingredients include olives, brine, and any additional seasonings. A stepper motor rotates the disc to align the funnels with the dispensing area. Ultrasonic sensors above each funnel monitor ingredient levels and notify the manager when a refill is needed. The ingredients are collected in a large funnel, mixed with brine, and transferred to the final container using a tube.

### **4.3.4 Control Unit**

The control unit, managed by an Arduino, oversees all machine operations. The Arduino is connected to an ESP32 module, which interfaces with a web server. This setup allows employees to place orders via the web server, while the admin controls and monitors the machine's overall functionality. Additionally, a PC is used to manage more complex tasks and interactions., a PC

power supply is used to power the machine's components.

## 4.4 Mobile Application

The mobile application is structured into two distinct sections. The first section is for the admin, who can monitor ingredient levels in the funnels and control the valve to fill the heater with water from the tap. The second section is designed for employees, allowing them to place orders for pickled olives using a mobile device instead of the traditional keypad and LCD.

The ordering process is streamlined into two screens: one for selecting the type of pickled olives and another for choosing the container size. If the employee opts for a customized order, they can also specify the amount of olives, brine, and additional seasonings to be included.

Additionally, the web server facilitates payment processing. When an employee places an order, the cost is automatically deducted from their account balance.

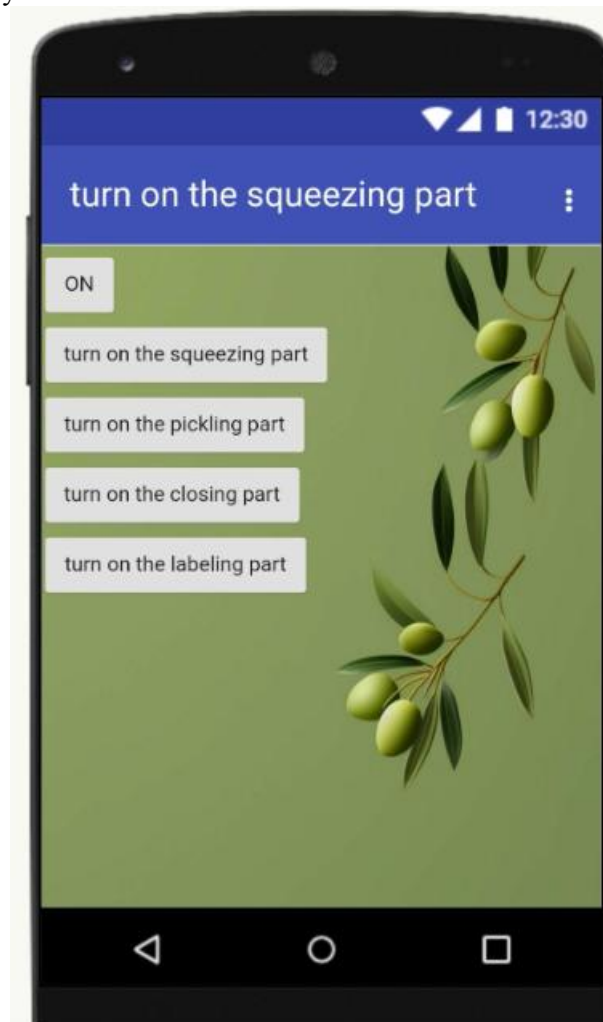
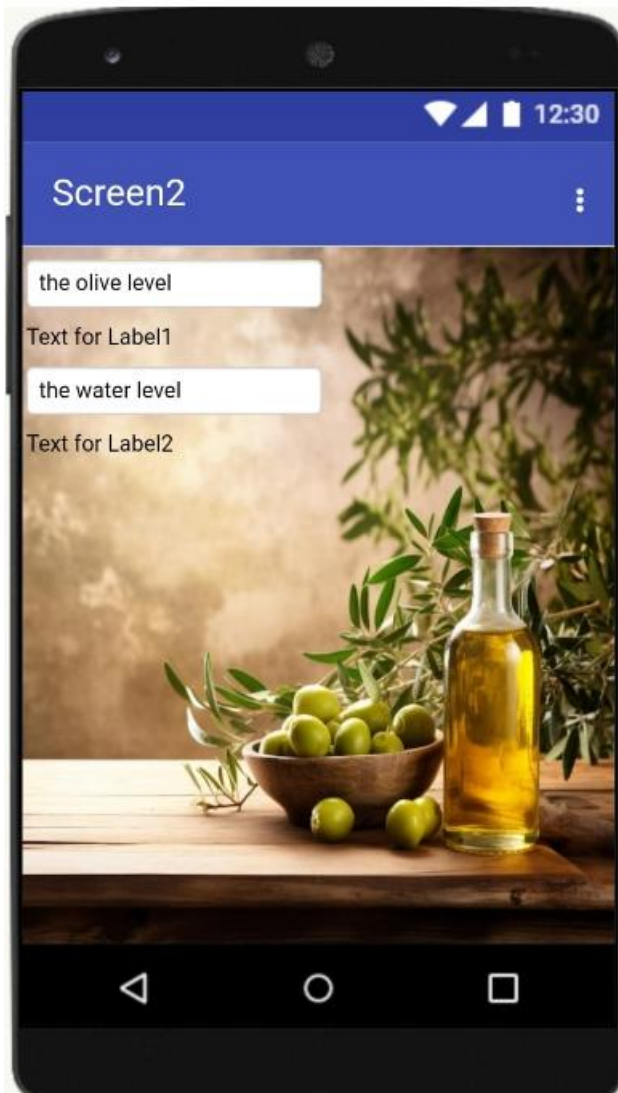


Figure 4.34: Screen1

Screen 2:



## Chapter

# Results and Discussion

By the end of this project, we successfully developed a smart, automatic pickling machine for olives with advanced features, including automated container dispensing. The admin can control the water filling process for the brine preparation through the mobile app, while employees can place orders using the app, streamlining the entire pickling process.

1. **Relay Malfunction:** We encountered an issue where the relays were not functioning as expected. Upon investigation, we discovered that the Arduino output was only 3.1 volts instead of the required 5 volts. Since the relays were designed to operate at 5 volts, they failed to work. We resolved this by switching to another Arduino output that provided the correct 5 volts, ensuring the relays operated correctly and the project proceeded successfully.
2. **ESP32 Uploading Issue:** The ESP32 initially refused to accept the code despite multiple attempts. After extensive research, we discovered that pressing the boot button during the code upload process resolved the issue, allowing us to successfully upload the code.
3. **The ESP32 didn't accept the code for multiple tries,** after doing a lot of searching, we found that we should press the boot button while the code is uploading.
4. **Power Supply Challenges:** Powering the various components of the machine posed a challenge, as some required 220 volts while others needed 12 volts. We addressed this by using a power supply from an old computer that provided both 5 volts and 12 volts, and used standard household electricity for the 220-volt components.
5. **Stepper Motor Insufficiency:** Initially, we attempted to use a Nema17 motor to rotate the disk and rollers, but it lacked sufficient power. After further testing, we determined that using two Nema23 motors was necessary to handle both the disk rotation and roller functions effectively.

## Chapter

# Conclusions and Recommendation

### 6.1 Summary

Our team successfully designed and developed the Pickled Olives Machine, an automated system that simplifies the process of ordering and preparing pickled olives. The machine integrates a web server and keypad for remote ordering, enhancing user convenience.

A standout feature of our machine is its precise cup dispensing mechanism. It operates flawlessly, dispensing containers accurately without manual intervention. The belt system is also efficiently designed, ensuring smooth and swift movement of each container to the processing area.

The inclusion of IR sensors allows the machine to handle multiple orders simultaneously, improving time efficiency and operational productivity. Additionally, the machine offers customizable options for pickled olives, enabling users to specify the quantity and type of olives they want, along with choosing between different container sizes for a personalized experience.

We encountered and addressed several challenges during development. One significant issue was providing power to various components, which required both 220 volts and 12 volts. We resolved this by utilizing a power supply from an old computer that provided 5 volts and 12 volts, while standard household electricity was used for the 220-volt components. Another challenge was with the stepper motor; initially, a Nema17 motor was insufficient for rotating the disk. We solved this by upgrading to two Nema23 motors, which provided the necessary power for effective operation.

ingredients or to rotate the roller of the conveyor belt. After a lot of experimentation, we had to use another 2 stepper motors which are Nema23 to do both functionalities.

In summary, our Caffeine Shot Machine is a well-designed and efficient device that makes it easy for employees to order their favorite drinks. With its advanced features, customization options, and smooth automation, the machine offers a unique and user-friendly experience for its users.

## 6.2 Recommendations

Recommendations:

1. Be careful when using the Arduino board, especially the Chinese version, as its output voltage is 3.1 volts instead of 5 volts. We recommend using the Italian version instead.
2. Avoid powering sensors and devices directly from the Arduino board. Use a separate power supply.
3. Always solder wires instead of just connecting them, as they are easily broken.

## 6.3 What we have learned

1. How to work with sensors like ultrasonic and IR, and motors like DC motors, servo motors, and stepper motors, as well as devices like pumps, heaters, and valves.
2. How to connect and use various types of high-voltage sensors and devices with Arduino.
3. How to connect Arduino to ESP32 and use its Wi-Fi features.

## 6.4 Future Work

1- we will add an option to remove the nucleus from the olives, improving the processing efficiency.

2- we will integrate an olive pressing mechanism to extract olive oil, expanding the functionality of the project and providing a complete olive processing solution. These advancements will enhance the system's versatility and cater to a wider range of olive-based products.

# References

- [1] Rahul Jadhv et al. “Smart coffee vending machine using RFID”. In: *Instrumentation Engineering department, Advances in Wireless and Mobile Communications* 10 (2017), pp. 793–800.
- [2] Sawai Pongswatd, Krit Smerpitak, and Teerawat Thepmanee. “Smart coffee vending machine based on IoT concept”. In: *International Journal of Innovative Computing, Information and Control* 16.4 (2020), pp. 1441–1448.
- [3] Kwangsoo Kim et al. “Smart coffee vending machine using sensor and actuator networks”. In: *2014 IEEE International Conference on Consumer Electronics (ICCE)*. IEEE, 2014, pp. 71–72.
- [4] Pavel Smutny Marek Babiuch Petr Foltynek. “Using the esp32 microcontroller for data processing”. In: *International Carpathian Control Conference (ICCC)* (2019), pp. 1–6.
- [5] M. Santhanakrishnan V. Nagarajan S. Sivakumar. “Design and Simulation of DC Motor Speed Control Using PID Controller”. In: *International Journal of Engineering and Technology* (2011).
- [6] S. Muruganand R. Gunasekaran V. Rajamani. “Design and implementation of obstacle detection and avoidance robot using infrared sensors”. In: *International Conference on Circuit, Power and Computing Technologies (ICCPCT)* (2016), pp. 1–6.
- [7] Li L Li W. “Design of Intelligent Ultrasonic Sensor for Internet of Things Applications”. In: *Journal of Physics: Conference Series, 1827(1), 012065* (2021).
- [8] J. Hu X. Zhang and X. Cui. “Design and Implementation of a Wireless Water Temperature Monitoring System”. In: *International Conference on Computational Science and Engineering (CSE) and IEEE International Conference on Embedded and Ubiquitous Computing (EUC)* (2016), pp. 174–178.
- [9] A. Al-Khalidi and I. B. Abu-Sulyman. “Design and Development of a Portable Industrial Multi-channel Data Acquisition System”. In: *2021 IEEE 11th Annual Computing and Communication Workshop and Conference (CCWC)* (2021), pp. 19–26.

- [10] N. A. Popov V. L. Boyarintsev V. V. Koryanov. “LCD Display with I2C Interface”. In: *Procedia Computer Science* 101 (2016), pp. 528–536.
- [11] D. Wadhwa R. Garg. “RFID Based Attendance System Using Arduino and GSM”. In: *International Journal of Computer Applications* 179.23 (2021), pp. 6–11.
- [12] A. Kumar S. Sahu. “Arduino Based Smart Home Automation System Using Android Smartphone”. In: *International Journal of Scientific Engineering Research* 6.7 (2015), pp. 62–65.
- [13] S. V. Dudul S. S. Patil. “Design and Development of Voice Command Recognition System for Smart Home Automation using Arduino”. In: *International Journal of Innovative Research in Science, Engineering and Technology* 9.12 (2020), pp. 8667–8674.
- [14] Y. Lee J. Kim. “Design and Implementation of an Arduino-based Home Automation System”. In: *International Journal of Smart Home* 9.7 (2015), pp. 11–22.