

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ



An-Najah National University
Faculty of Engineering and Information Technology
Computer Engineering Department

Graduation Project 2

MedRem

Done by:
Ahmad Shaer

Supervisor: **Dr. Aladdin Masri**

Submitted in partial fulfillment of the requirements for the
Bachelor degree in Computer Engineering

June-2024

Acknowledgment

I would first want to express My sincere appreciation to the Almighty. Without His blessings that led us every step of the way our route, leading to the successful completion of this project.

I'm are really grateful to my parents and other loved ones. I have been continuously inspired and uplifted by your constant encouragement along our journey. I have gained strength and stability from your kind and supportive remarks, which have created a solid foundation.

And to my supervisor, Dr. Aladdin Masri, I express our gratitude for your guidance and your significant suggestions. The precise insights, insightful remarks, provoking debates, and useful feedback you have provided have greatly improved our project's capability.

We would especially want to thank An-Najah National University's Computer Engineering Department for their continuous support.

Disclaimer

The team is comprised of one computer engineering student, Ahmad Shaer, currently pursuing their education at An-Najah University. Collaborated on the completion of this paper. The authors acknowledge full responsibility for any errors identified within. The ideas, suggestions, findings, and opinions expressed in this paper solely belong to the authors and do not reflect the viewpoint of An-Najah National University.

Table of Contents

Acknowledgment	2
Disclaimer	3
Abstract	7
1 Introduction.....	8
1.1 Problem Statement	8
1.2 Problem Objective.....	8
1.3 Scope of work.....	9
2 Constraints, Standards and Codes and Earlier Work	9
2.1 Constraints.....	9
2.2 Standards / Codes	9
2.3 Earlier work:.....	9
3 literature Review	10
4 Methodology	11
4.1 Technologies – Components:.....	11
4.1.1 Esp32:	11
4.1.2 Arduino Mega 2560	12
4.1.3 Servo Motor	13
4.1.4 LCD.....	14
4.1.5 Temperature Sensor	15
4.1.6 Buzzer	16
4.1.7 IR Obstacle Sensor.....	17
4.1.8 Ultrasonic Sensor:.....	18
4.1.9 Sound Sensor:	19
4.1.10 Piezo Sensor:.....	20
4.1.11 Wires:.....	21
4.2 Mechanical Part:.....	22
4.3 Mechanical Design:.....	22
4.4 Features:	28

4.4.1	Storage place:.....	28
4.4.2	Patient drawer / Detection System:.....	29
4.4.3	Components area:.....	30
4.4.4	LCD:	31
4.4.5	WIFI – Serial – API – DNS	33
4.4.6	Pill dispensing:.....	35
4.4.7	Mobile app:	35
5	Conclusion/ Recommendations and Future Works.....	38
6	References.....	38

Table of Figures:

Figure 1:	Esp32.....	11
Figure 2:	Arduino Mega 2560.....	12
Figure 3:	360-degree Servo Motor.....	13
Figure 4:	20x4 LCD	14
Figure 5:	LM35 Sensor	15
Figure 6:	Buzzer.....	16
Figure 7:	IR Sensor	17
Figure 8:	Ultrasonic Sensor	18
Figure 9:	Sound Sensor.....	19
Figure 10:	Pressure Sensor - Piezo Sensor	20
Figure 11:	male to male wires.....	21
Figure 12:	male to female wires.....	21
Figure 13:	female to female wires	21
Figure 14:	First design	22
Figure 15:	First design - Sectors	23
Figure 16:	Second design - initial prototype.....	24
Figure 17:	Second design – Enhanced	24
Figure 18:	Second design - Enhanced2.....	25
Figure 19:	Second design – Testing.....	25
Figure 20:	Final design	26
Figure 21:	Final design - Prototype	26
Figure 22:	Final design - Prototype drawer	27
Figure 23:	Final design - IR sensor.....	27

Figure 24: Single Container	28
Figure 25: All Containers.....	29
Figure 26: Patient drawer.....	29
Figure 27: Front side.....	30
Figure 28: Components Area	30
Figure 29: Backside	31
Figure 30: Initial Screen.....	31
Figure 31: Runtime Screen	32
Figure 32: Empty Alarm	32
Figure 33: Temperature alarm	33
Figure 34: Postman - API testing.....	33
Figure 35: ESP32 and Arduino.....	34
Figure 36: DNS service.....	34
Figure 37: Mobile app Screens	37

Abstract

In today's era of increasing reliance on internet and technology to streamline our daily lives, there is a growing recognition of the potential for these advancements to enhance various aspects of our well-being, particularly in healthcare. This becomes particularly critical for individuals who are ill or elderly, as ensuring timely and accurate medication intake is crucial for effective treatment outcomes. Leveraging Today's technologies, this study introduces an innovative pill management system designed to improve patient medication adherence. By organizing a patient's prescribed medications according to dosage schedules and facilitating timely consumption, the system aims to optimize treatment effectiveness. Incorporating features such as personalized reminders and notifications, it seeks to mitigate the risk of patients forgetting to take their medication. Through the integration of user-friendly interfaces and robust data management protocols, this project addresses the growing challenge of medication non-adherence, offering a reliable and efficient solution to promote patient well-being and healthcare efficacy.

1 Introduction

1.1 Problem Statement

The efficacy of medicine depends on taking it at the appropriate time and dosage. Many people have serious difficulties sticking to their medicine regimens, especially elderly persons. They may provide their prescription at the improper time, in the wrong amount, or forget to take it altogether. When people consume multiple medications, these problems get much more problematic since it is simple to mistake one prescription with another. Furthermore, to maintain the effectiveness of medications, it is crucial to store them properly. Medications must be stored in a proper setting, which is usually a cold, dry location free from moisture and sunshine. Medication might get misplaced or harmed if it is not stored correctly. It can be extremely difficult to remember schedules, manage several prescriptions, and maintain appropriate storage conditions, especially for the elderly or people with cognitive problems. Medication non-adherence is frequently the result of this circumstance, and it can lead to less than desired health outcomes, a larger number of hospital admissions, and increased healthcare expenses.

1.2 Problem Objective

Creating a system that can store up to four different medications, each with up to seven pills, and set up the specific times for taking these medications is essential for effective medication management. This system would need to be programmed with the prescribed times for each medication using a mobile application. When it's time to take the medications, the system should automatically alert the person with an alarm and dispense the correct medications in the right dosages. Such a system would help ensure proper medication adherence, significantly reducing the risk of errors such as taking the wrong dosage or missing a dose. Additionally, this system would be particularly beneficial for older adults or individuals with cognitive impairments who may struggle with managing multiple medications.

1.3 Scope of work

In my project, the focus was on individuals who take medication in general, but it could be especially useful for older adults who take multiple medications and those with cognitive impairments. Older adults often have complex medication regimens that can be challenging to manage, increasing the risk of errors such as missed doses or incorrect dosages.

Additionally, individuals with cognitive impairments may struggle with remembering to take their medications on time and in the correct amounts. Our project aims to address these challenges by providing a reliable system that ensures proper medication adherence, thereby improving health outcomes and reducing the burden on caregivers.

2 Constraints, Standards and Codes and Earlier Work

2.1 Constraints

1. Designing the application in a way that is both functional and effective.
2. Time to get prototypes (3d printing) of product and testing them out.
3. Working alone: meant doubling the amount of the workload and the cost to creating such a system from scratch.

2.2 Standards / Codes

I used Arduino IDE to write the code and test it into the Arduino mega 2560 and ESP32 for WIFI connection.

2.3 Earlier work:

My project solves problem that might occur in the previous work such as free falling object that could lead to opening up the medicine and spilling it and also my project is much smaller to

make it easier to take it anywhere and its cost effective and its all made out of plastic to ensure a good environment (that doesn't rust or corrode) to store the medicine in.

3 literature Review

According to (1) a study was conducted on 239 patients starting new medications for a chronic condition found that of the 67 non-adherent patients, 55 Per cent were unintentionally nonadherent and also talked about patient nonadherence with medicines through application of Reason's (1990) human error theory in organizations. Based on Reason's description of the causes of unsafe acts, he identifies the causes of nonadherence in terms of: Unintentional actions, that consist of slips due to lack of attention (ie, inadvertently taking the wrong tablet) and lapses (due to a failure of memory such as forgetting to take a dose) and intended actions that comprise mistakes and violations. Also, (2) says that Nonadherence may lead to increased use of medical services, including more ER visits and longer hospital stays. greater health care expenses as a result of hospitalizations and extra diagnostic tests (Brickman & Yount, 1996; Rapoff, 1999). Even though following medical advice has clear advantages and irregular adherence can have catastrophic consequences, nonadherence to prescriptions is still a major issue that is challenging to quantify, comprehend, and address. Furthermore, (3) concludes that Medication non-adherence may be significantly associated with all-cause hospitalization and mortality in older people. And that medication adherence has to be monitored and managed in order to minimize hospitalization, enhance medical results, and save medical expenses.

4 Methodology

4.1 Technologies – Components:

4.1.1 Esp32:

the ESP32 microcontroller is made for a variety of uses, such as wearable electronics and Internet of Things devices and small Projects. With its dual-core CPU, built-in Wi-Fi, and Bluetooth capabilities, it's perfect for tasks requiring connectivity and wireless communication. In addition, the ESP32 has a wide peripheral set that enables extensive interaction with sensors and actuators. These peripherals include several GPIO pins, ADCs, DACs, and hardware PWM. Professionals and enthusiasts alike like the ESP32 due to its strong performance and low power consumption. So, I used ESP32 to interact with the mobile application to gather information about the times which the patient needed to take a certain medicine and then transfer the data received to the Arduino to then give commands to the servo motor to dispense one pill from that medicine to the drawer.

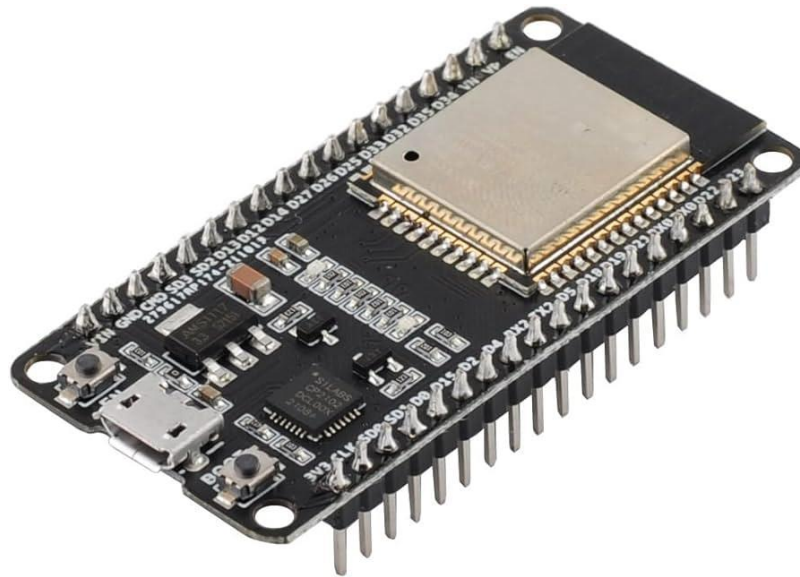


Figure 1: Esp32

4.1.2 Arduino Mega 2560

I used Arduino as a main controller for the machine in which it receives information from the esp32 using serial communication and act accordingly and it control the sensors and the motors and lcd in the system and provide a power supply for them as well.



Figure 2: Arduino Mega 2560

4.1.3 Servo Motor

I used the Servo Motor to act as the main mechanical part to dispense a pill from each container when it gets a command from the Arduino when it's the time for the patient to take the medicine it will move for a certain angle (20° degree) to be exact, in which moves one sector of the container that stores the pill.



Figure 3: 360-degree Servo Motor

4.1.4 LCD

I used a 20x4 (20 characters, 4 lines) to give the patient information about the system.

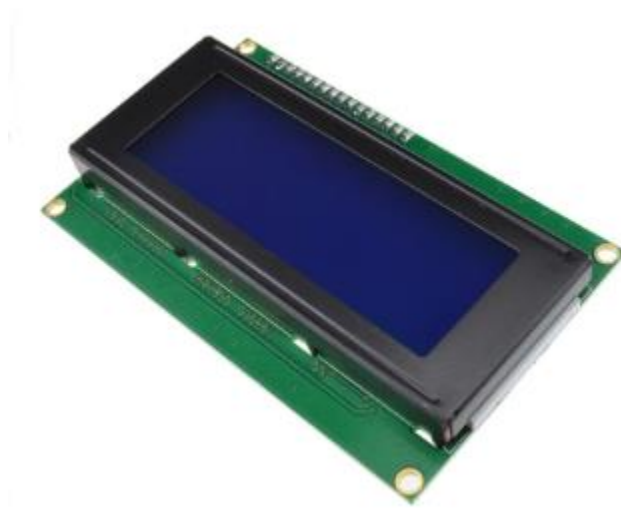


Figure 4: 20x4 LCD

4.1.5 Temperature Sensor

I used LM35 component to measure the temperature of the device and if the device temperature is above 29 °C then the lcd will show a warning.



Figure 5: LM35 Sensor

4.1.6 Buzzer

When the ESP32 gives the command to the Servo motor to dispense a pill from one of the containers or number of containers and then when the pills are in the cup the application will alarm the caretaker or the patient to take his medicine by airing a sound for around 5 seconds

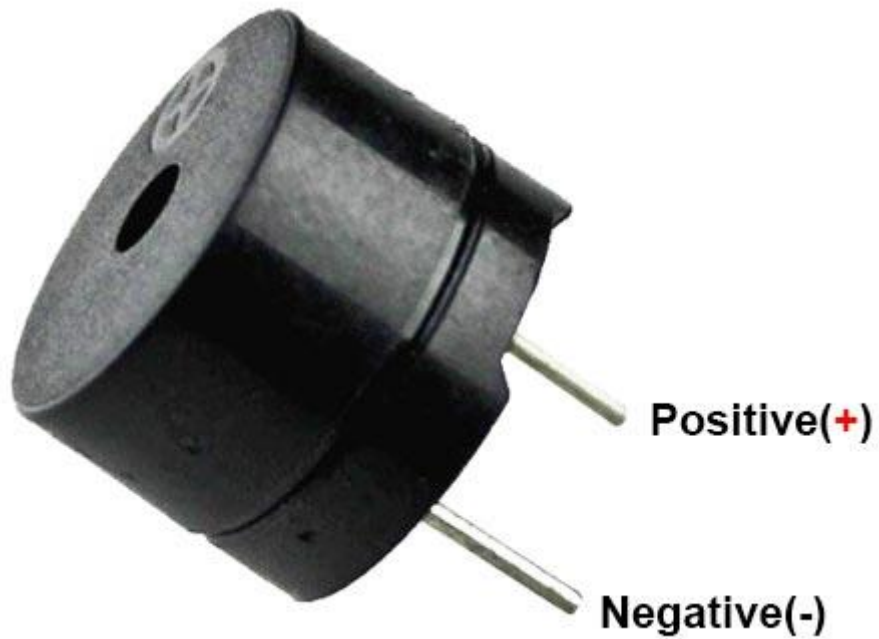


Figure 6: Buzzer

4.1.7 IR Obstacle Sensor

I used this sensor for each container to detect if a pill was dispensed after the servo pushed the drawer and if there were no pills it gives the servo another command to push the drawer once more until there a pill is detected but the accuracy of this sensor wasn't as high as I wanted because the sensor was put in a confined space and the reflection of the light from the walls affected the accuracy of the sensor so then I tried another sensor which is Ultrasonic sensor.

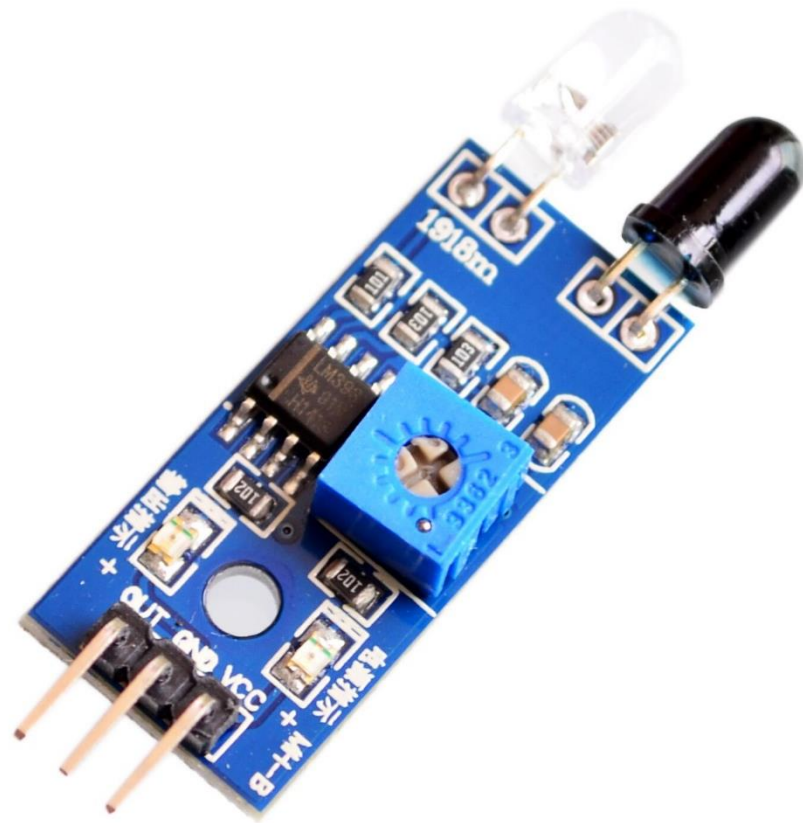


Figure 7: IR Sensor

4.1.8 Ultrasonic Sensor:

I used this sensor to detect if there were a pill dispensed by aiming it at the hole in which the pill would come out of the container and by first measuring the distance between the sensor and the hole of the container and then if the sensor detected a distance less than that value then there were a pill that traveled through that path and pill was detected but after testing it I found that the accuracy of this approach still low due to the small size of the pill so I thought of different approach which was using a sound Sensor to detect the sound of the pill hitting the drawer in which the patient takes his medicine from.

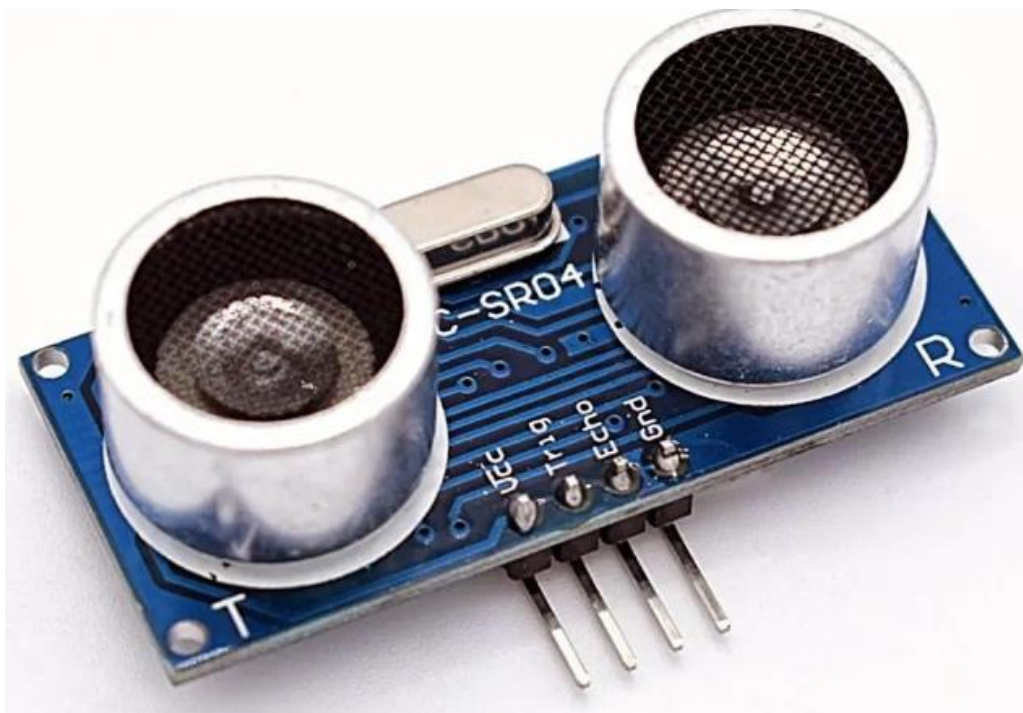


Figure 8: Ultrasonic Sensor

4.1.9 Sound Sensor:

I used this sensor to detect if the pill was dispensed by setting a threshold that when the sound sensor would pass then a pill was detected and that's due to sound of the pill hitting the drawer but after consideration this approach could be compromised due to the sound surrounding the device that could affect the accuracy of the sensor so I thought of a different approach which is pressure sensor.

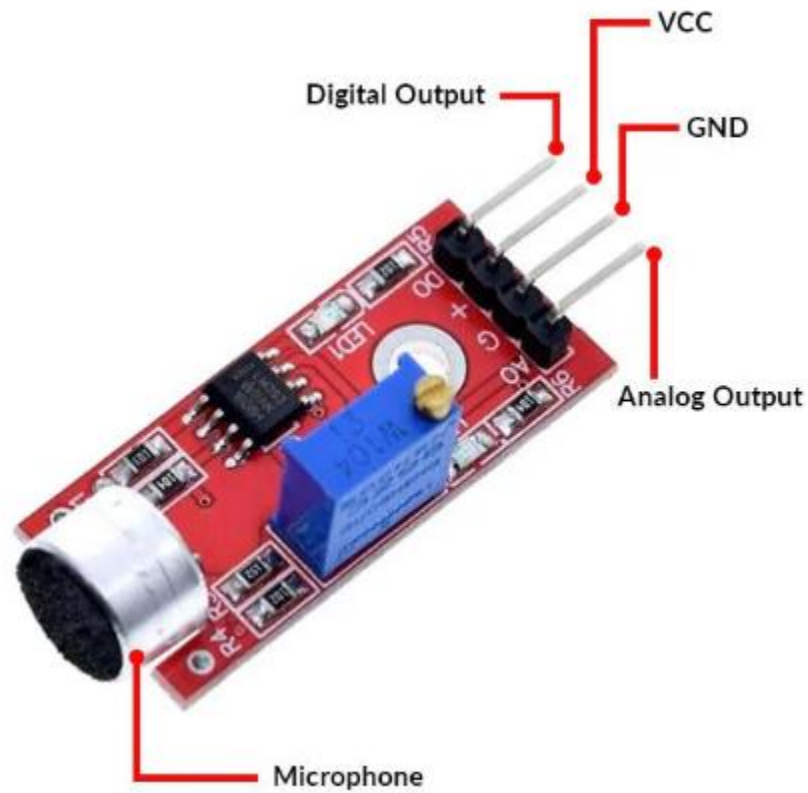


Figure 9: Sound Sensor

4.1.10 Piezo Sensor:

Better known as Pressure sensor, which I used this sensor to detect if pill was dispensed by sticking it to the patient drawer under where the pill would come out so when the pill is dispensed and it would eventually come in contact with the drawer it will hit the sensor and if the sensor produced an output more than a certain threshold (I determined this value after testing) then a pill would be detected.

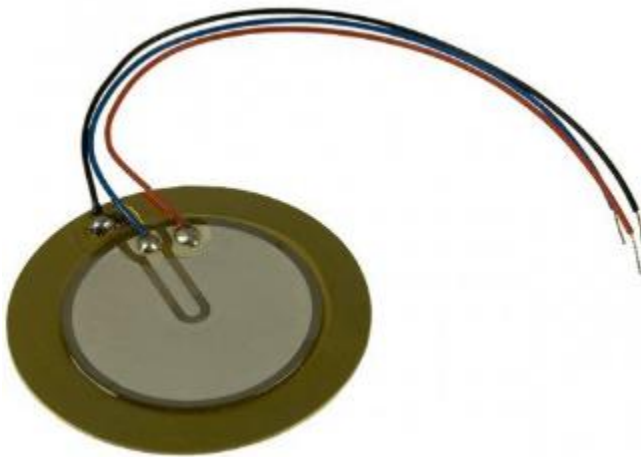


Figure 10: Pressure Sensor - Piezo Sensor

4.1.11 Wires:

I used different type of wires to adapt to the type of connectivity for each component.



Figure 11: male to male wires



Figure 12: male to female wires



Figure 13: female to female wires

4.2 Mechanical Part:

The mechanical portion of the project is what moves the medication pills out of the drawer (container). To do this, it needs a number of servo motors to create movement in conjunction with circular gears that are attached to each servo. The drawer was also made to absorb the torque that the circular gears provide in order to move the slots, and the number of slots that are moved also depends on how much torque the servo generates and also it has Pressure sensor to detect whether a pill was dispensed or not.

4.3 Mechanical Design:

First, I had to design how the application will look like using 3D drawing applications and I had few ideas how to implement the application and first design looked like this:

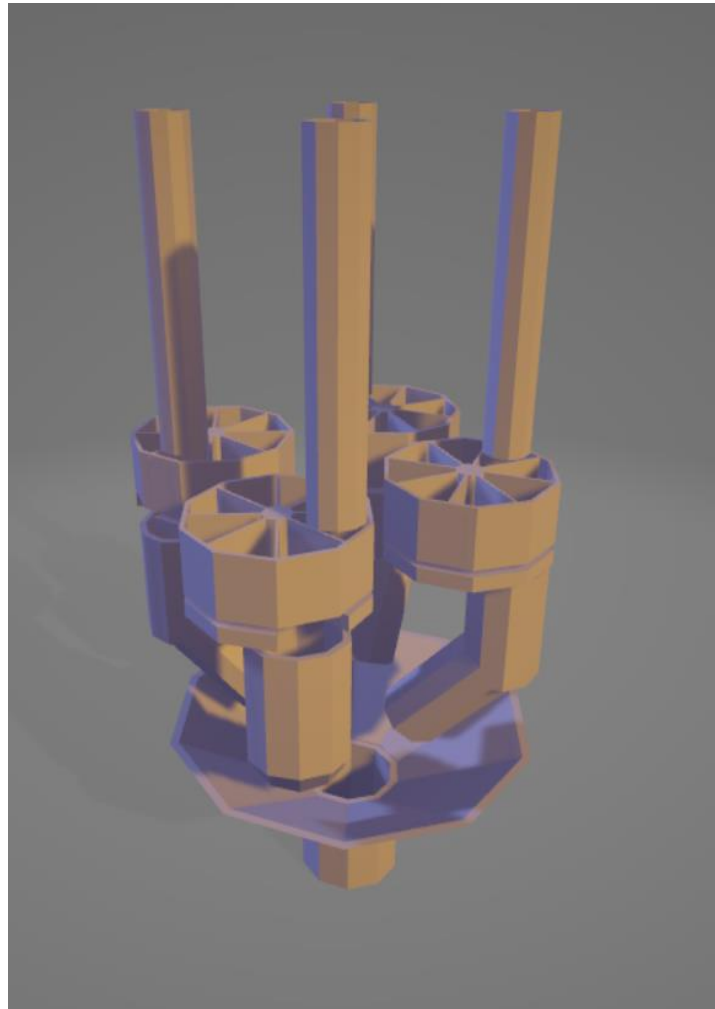


Figure 14: First design

For this design my approach was that the person drop the pill in the pipe and it is stored in the containers under it in one sector and then the servo motor moves to the next free sector to store another pill as shown in the picture below and when it's the time for the patient to take the pill the servo will move in the other direction dropping the pill into a cup but the problem with this design that it introduced free falling for the pills which could lead to the damaging the pill or even spilling its material.

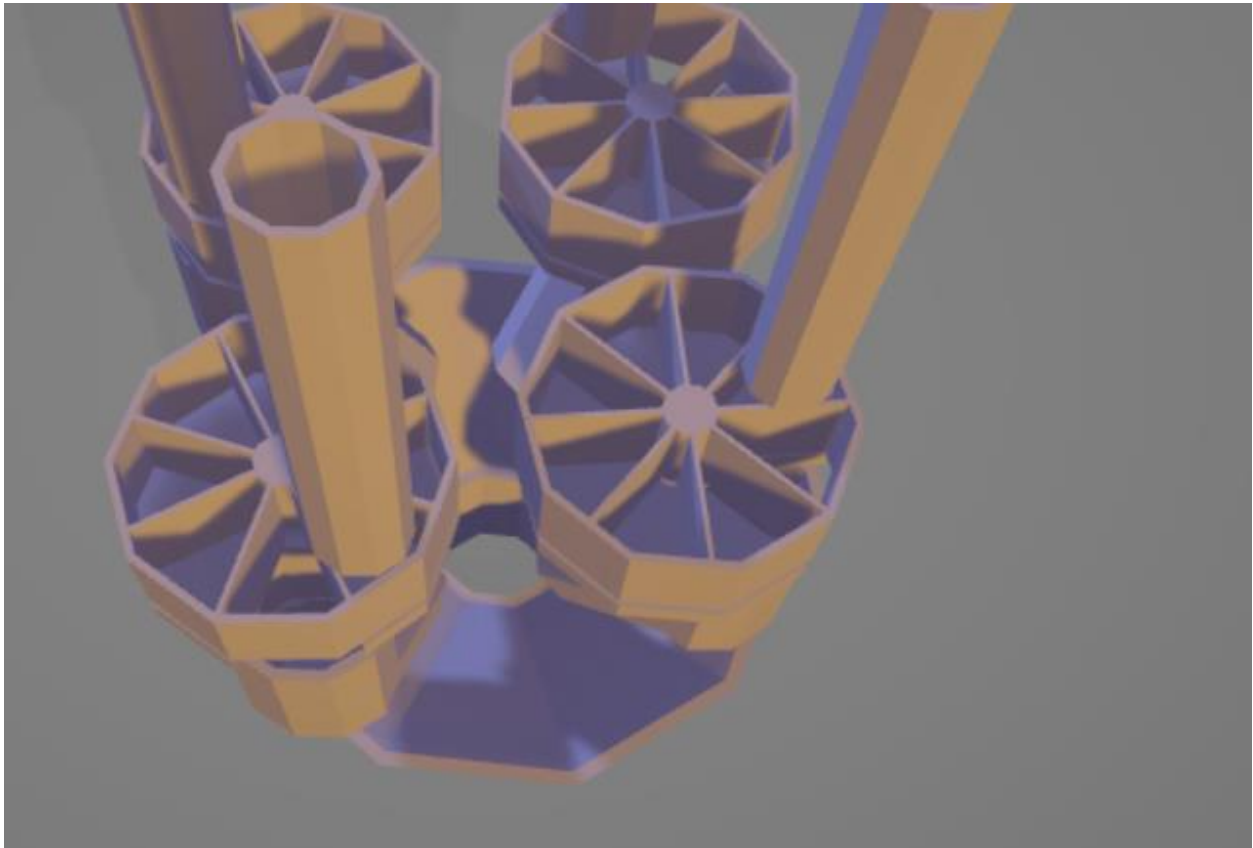


Figure 15: First design - Sectors

Then I went to a different approach minimizing the falling distance that the pill has to travel and design it like this:

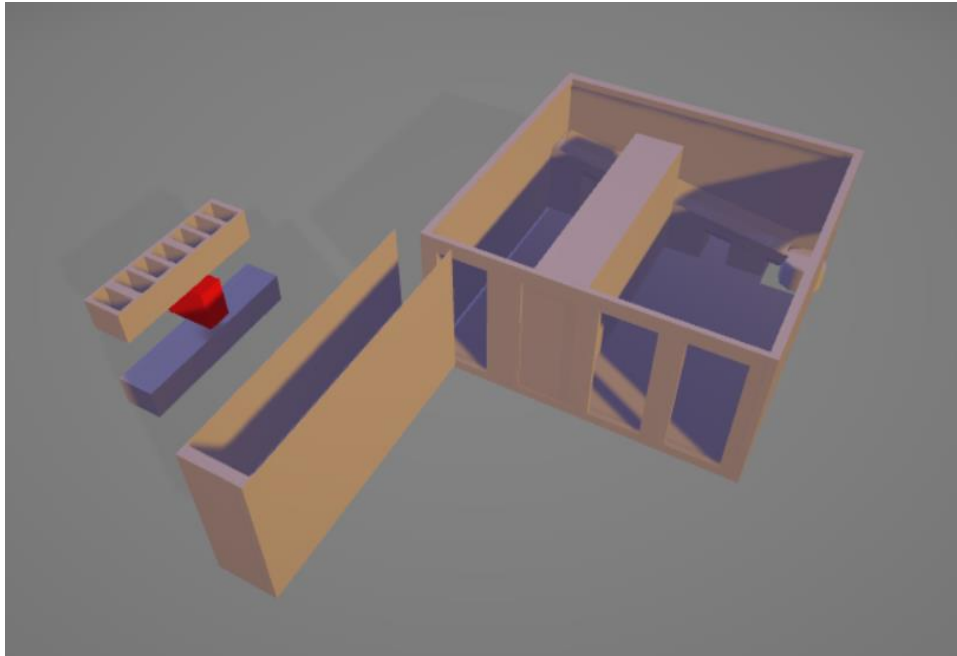


Figure 16: Second design - initial prototype

But then enhanced the design to make it easier for the user to store the pills and to add a place to store the Arduino and the esp32 and the wires.

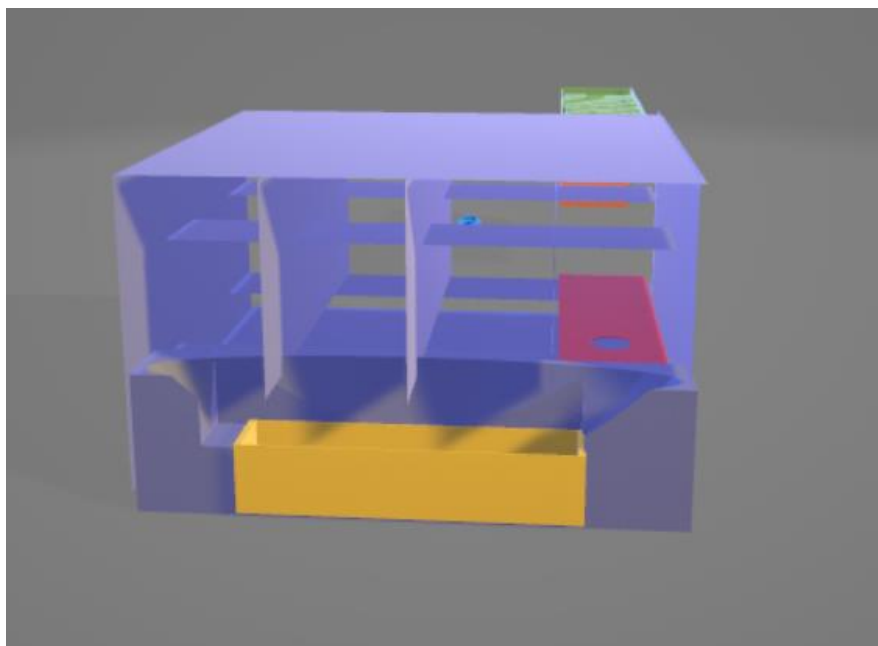


Figure 17: Second design – Enhanced

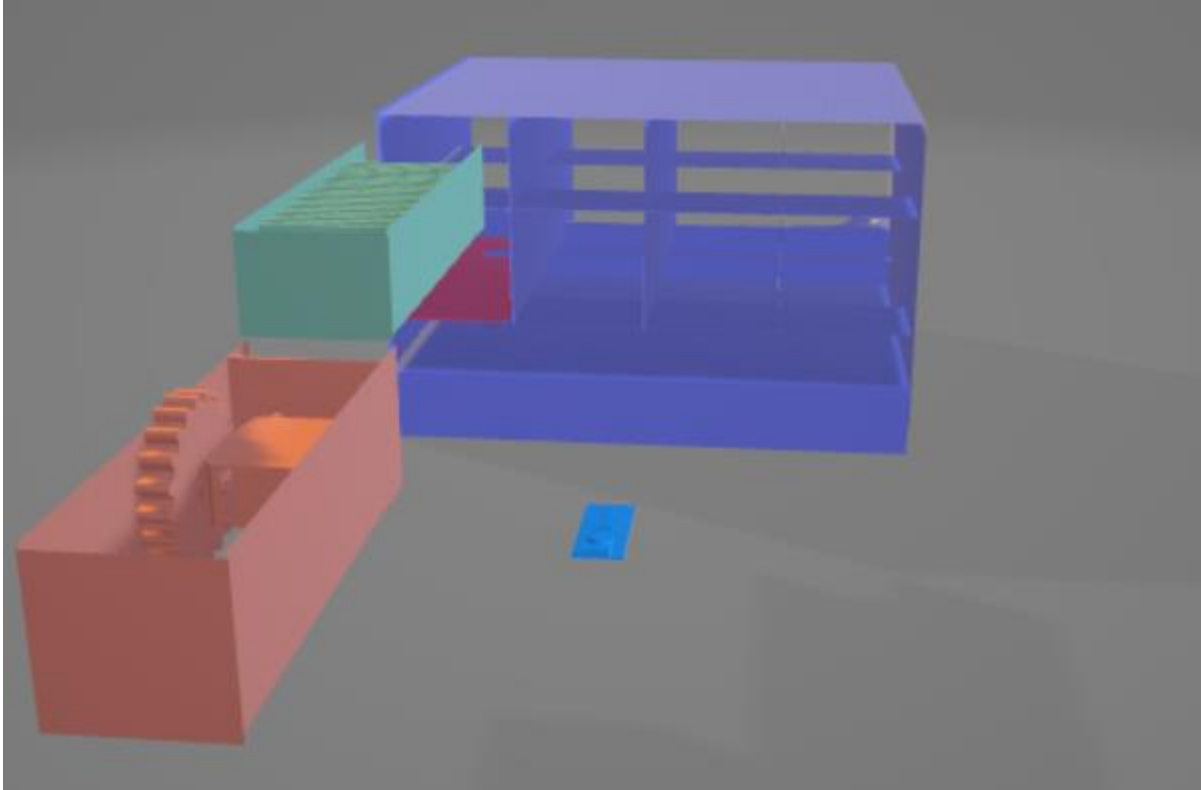


Figure 18: Second design - Enhanced2

And then I printed a prototype of the main core of the application:

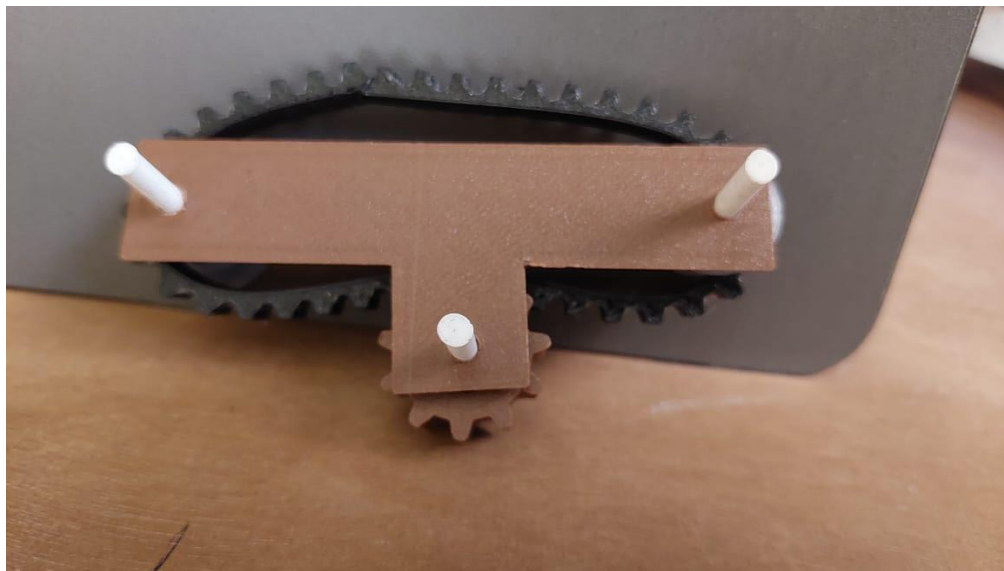


Figure 19: Second design – Testing

but after testing it by trying to move it manually without attaching the servo, it was hard to move and it produced a janky movement and instability which wasn't the result I wanted so I thought of a different approach for the application.

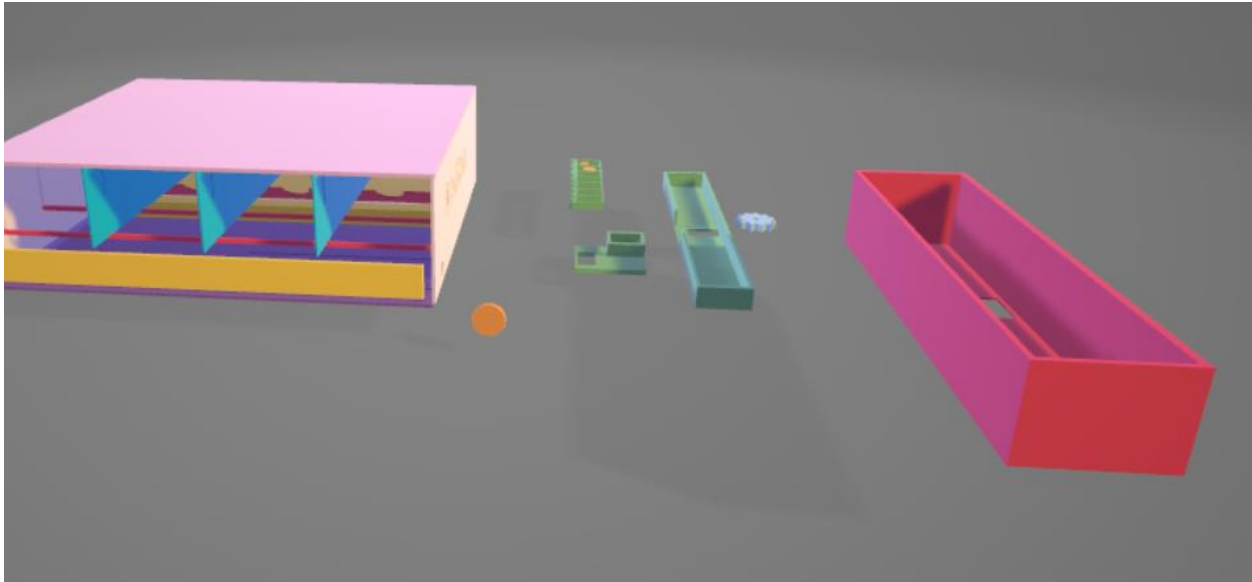


Figure 20: Final design

And then printed the main core of the application.



Figure 21: Final design - Prototype

And tested its movement and it produced the movement and the output that I wanted so I printed a bigger drawer and assembled them together and this is how they looked like.

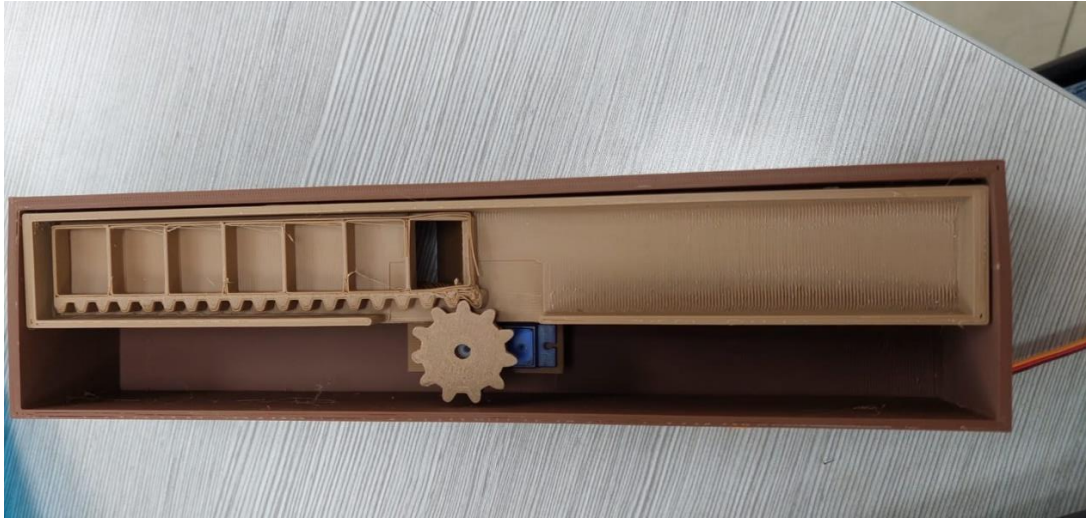


Figure 22: Final design - Prototype drawer

And then tested it and it worked and then went ahead and printed the other drawers and the box of the application and also added a place to put an IR sensor for each output to check whether there was a pill dispensed or not and if not, it will tell the servo to move to the next sector and it will continue to do so until it reaches the maximum number of the sectors or a pill dispense from the sectors.

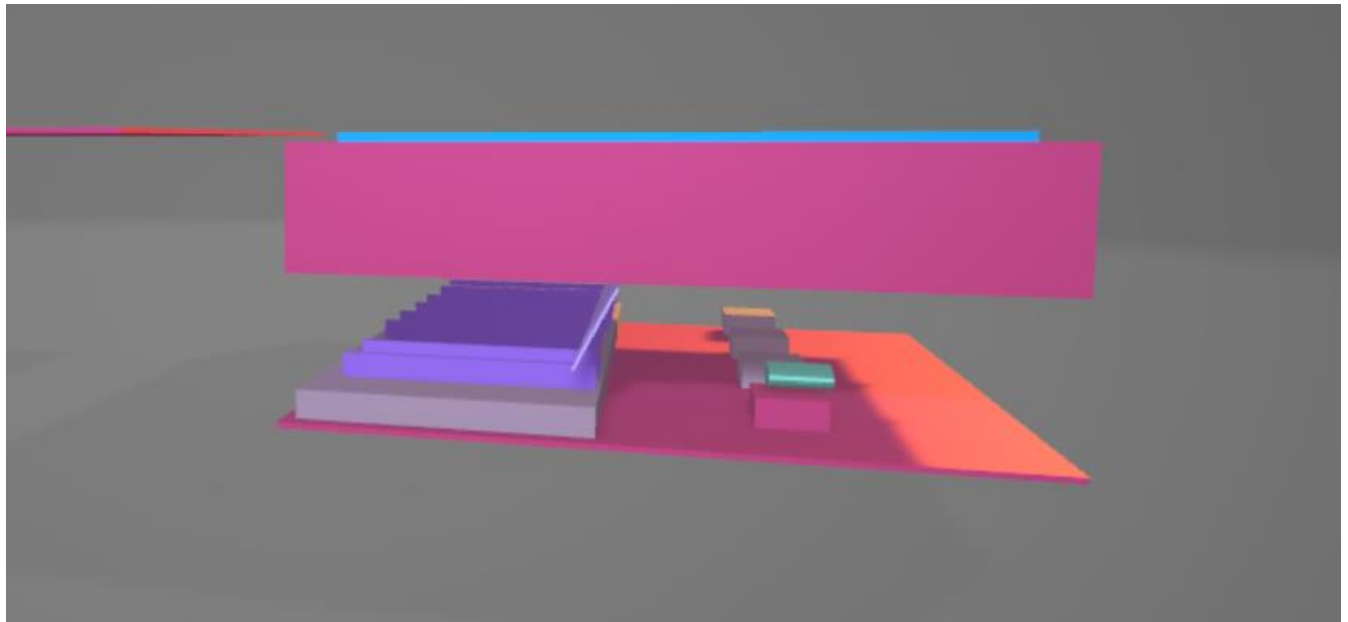


Figure 23: Final design - IR sensor

4.4 Features:

4.4.1 Storage place:

I build the pill container out of plastic because it the most fitting material because it doesn't corrode or rust and its more continent to build the sectors of the container that can contain one pill and also can contain the servo and the conveyer that moves the sectors and I made 7 sectors to store up to 7 pills and I build 4 containers to store up to 4 different types of medicines.

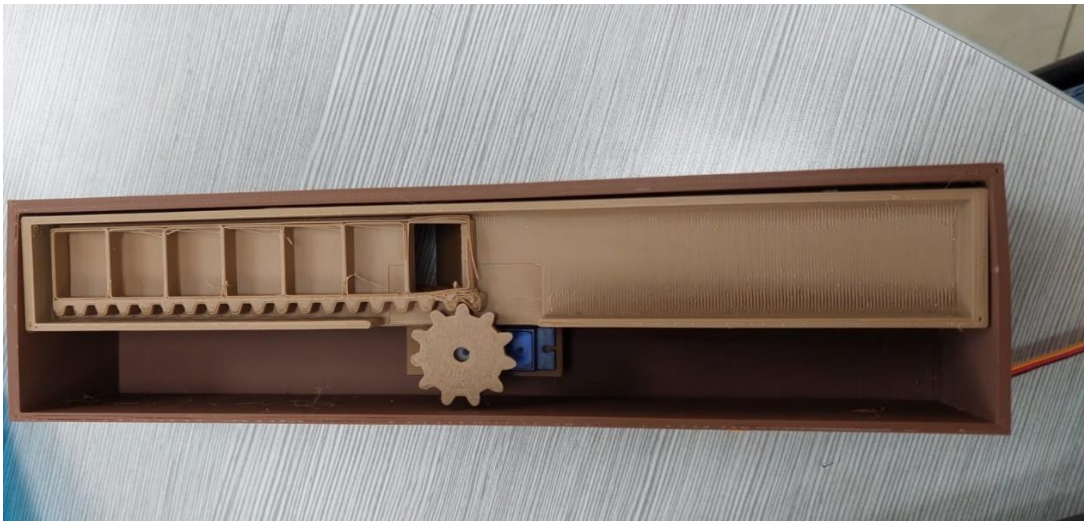


Figure 24: Single Container



Figure 25: All Containers

4.4.2 Patient drawer / Detection System:

The drawer where the pills will be after the system decides that its time for the patient to take his medicine and the sensors to detect whether a pill was dispensed.

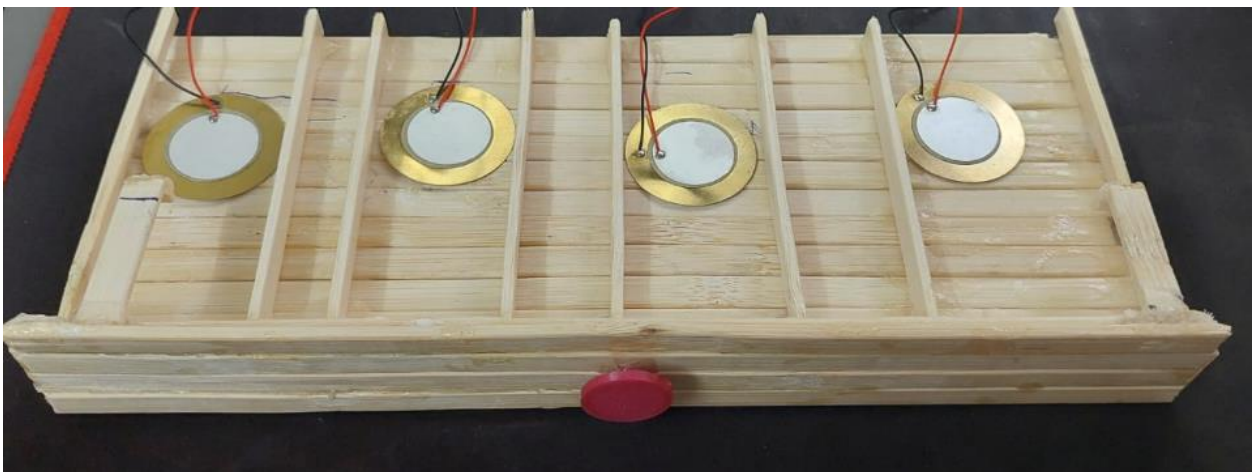


Figure 26: Patient drawer

4.4.3 Components area:

Which is an area under the containers and behind the drawer to store all the wires and components of the system to hide the complexity from the user and also make it accessible for the technician to gain access to parts for maintenance if needed.



Figure 27: Front side

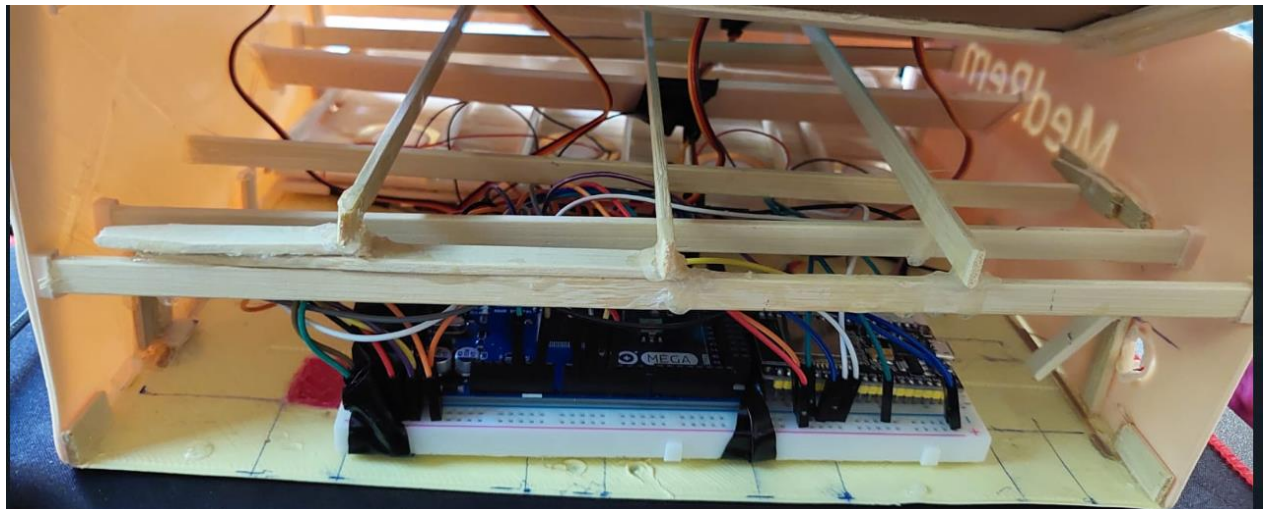


Figure 28: Components Area

4.4.4 LCD:

In the backside of the device is the patient drawer as well as a lcd to give information about the system to the patient.

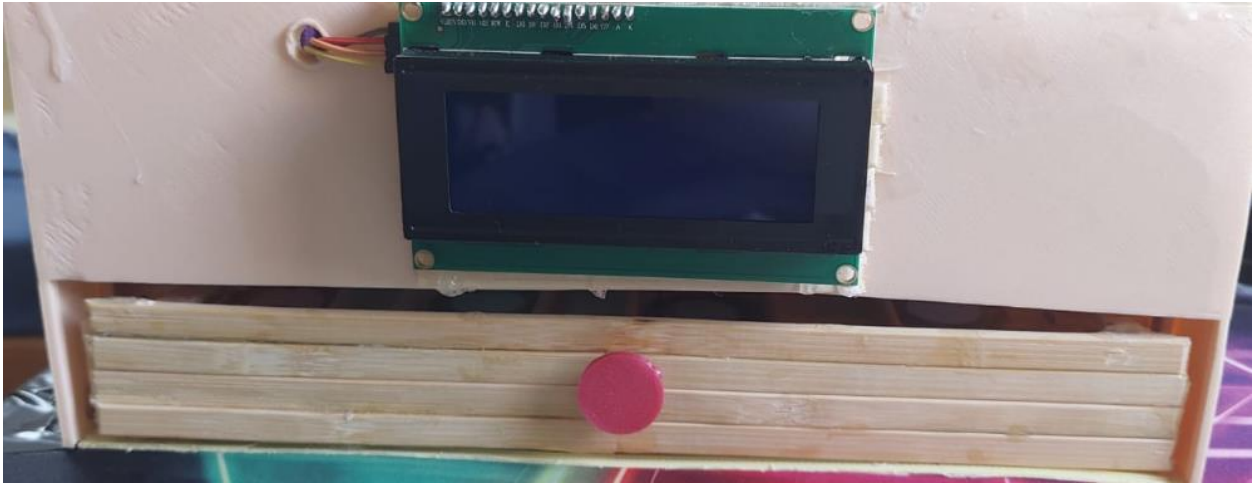


Figure 29: Backside

This is initial screen in which the patient would see before synching the mobile application with the device.



Figure 30: Initial Screen

Then after synching it with the mobile the screen would look like this:

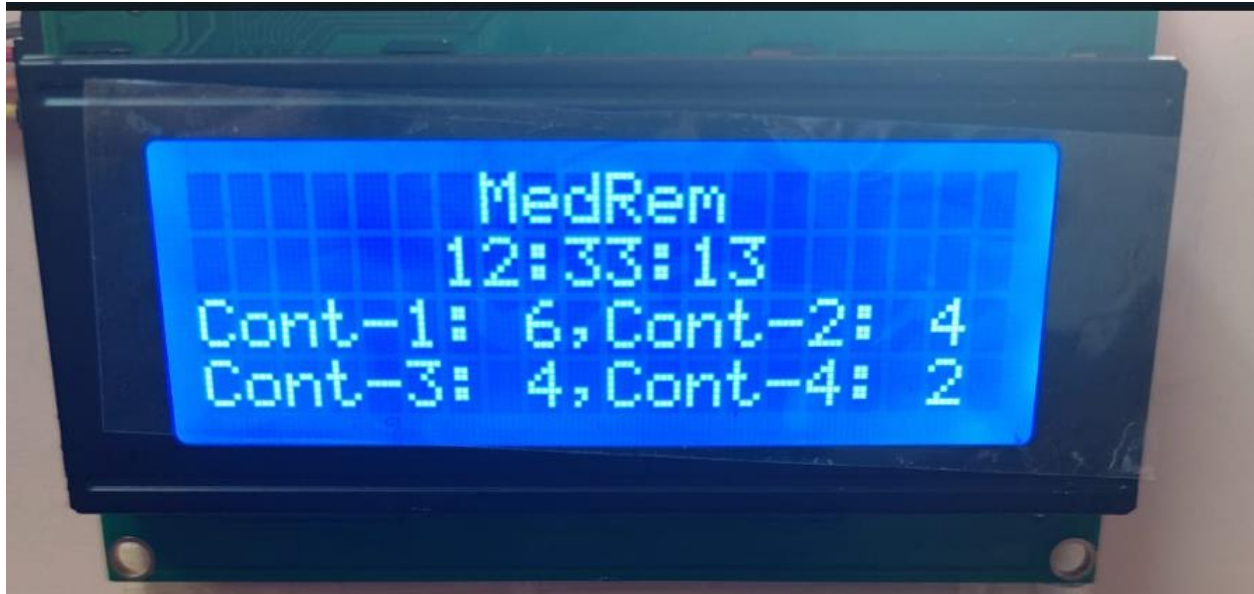


Figure 31: Runtime Screen

In which it shows the time which is synchronized with the mobile application as well as the number of pills remaining in each container.

And this is the screen in which the patient would get after one of the containers is empty.



Figure 32: Empty Alarm

And also, when the temperature is above 29 °C an alarm will be shown in the lcd.



Figure 33: Temperature alarm

4.4.5 WIFI – Serial – API – DNS

I used ESP32 to connect to WIFI and act as a server and accept client(device) and also implemented a serial communication between esp32 and the Arduino to transmit the information that the esp32 gets from the mobile app in a JSON format and also designed the commands as an API so that the device could be implemented in different applications such as IOT or build a web application using these, API'S.

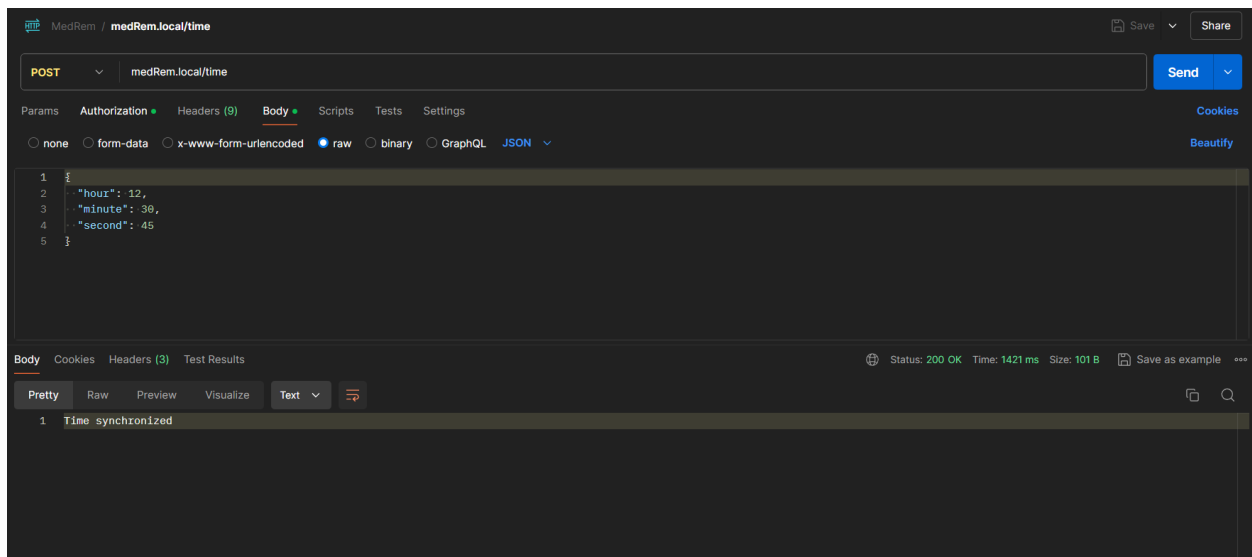


Figure 34: Postman - API testing

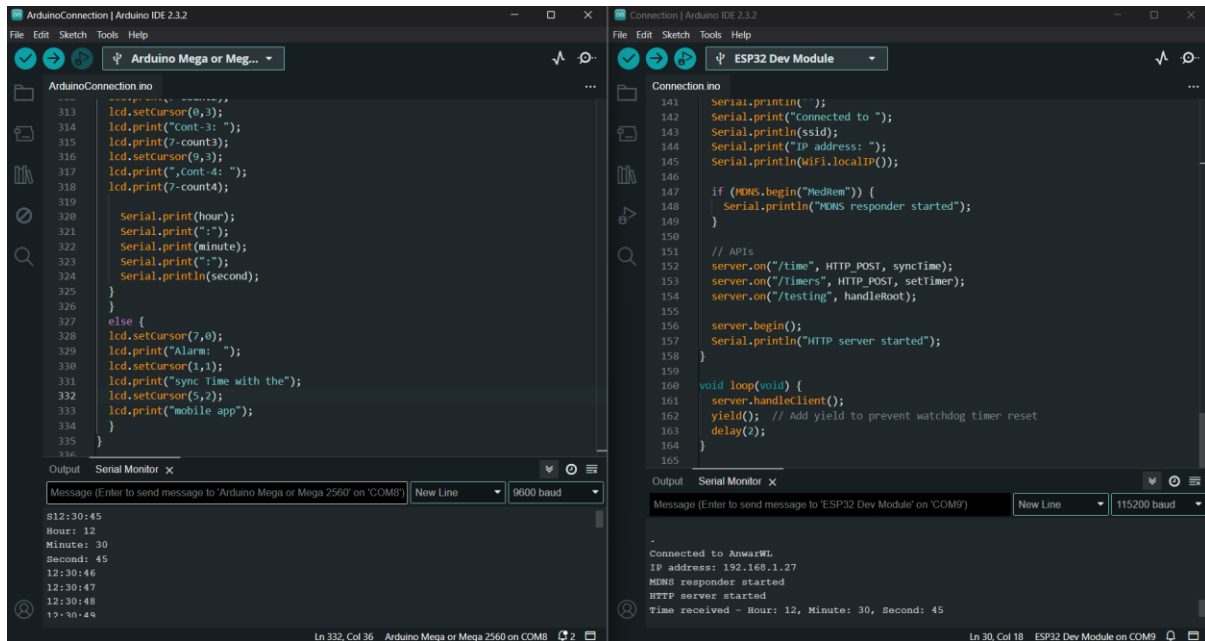


Figure 35: ESP32 and Arduino

The picture above shows the esp32 receiving the data from the postman after sending the request through the URL and then sending the information to the Arduino through serial communication and also added a DNS service in which the URL is typed through a name and there is no need to store the IP address of the esp32 server.

```
#include <ESPmDNS.h> // for dns service
#include <ArduinoJson.h> // JSON library

if (MDNS.begin("medrem")) {
  Serial.println("MDNS responder started");
}

POST medRem.local/time
```

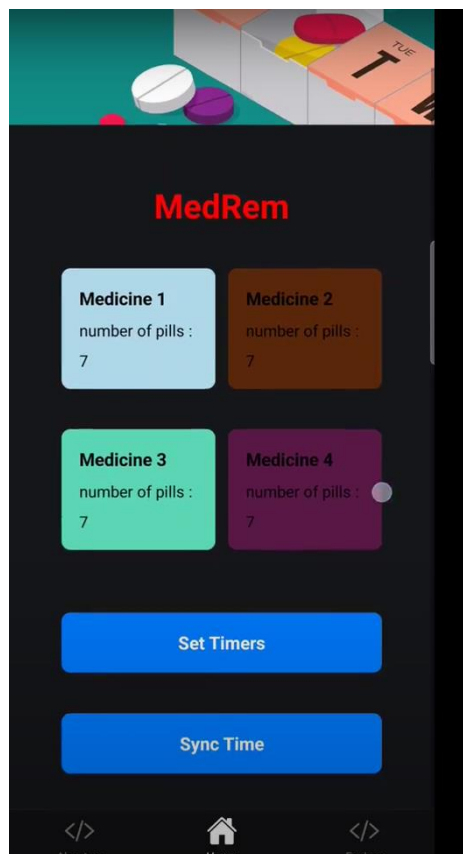
Figure 36: DNS service

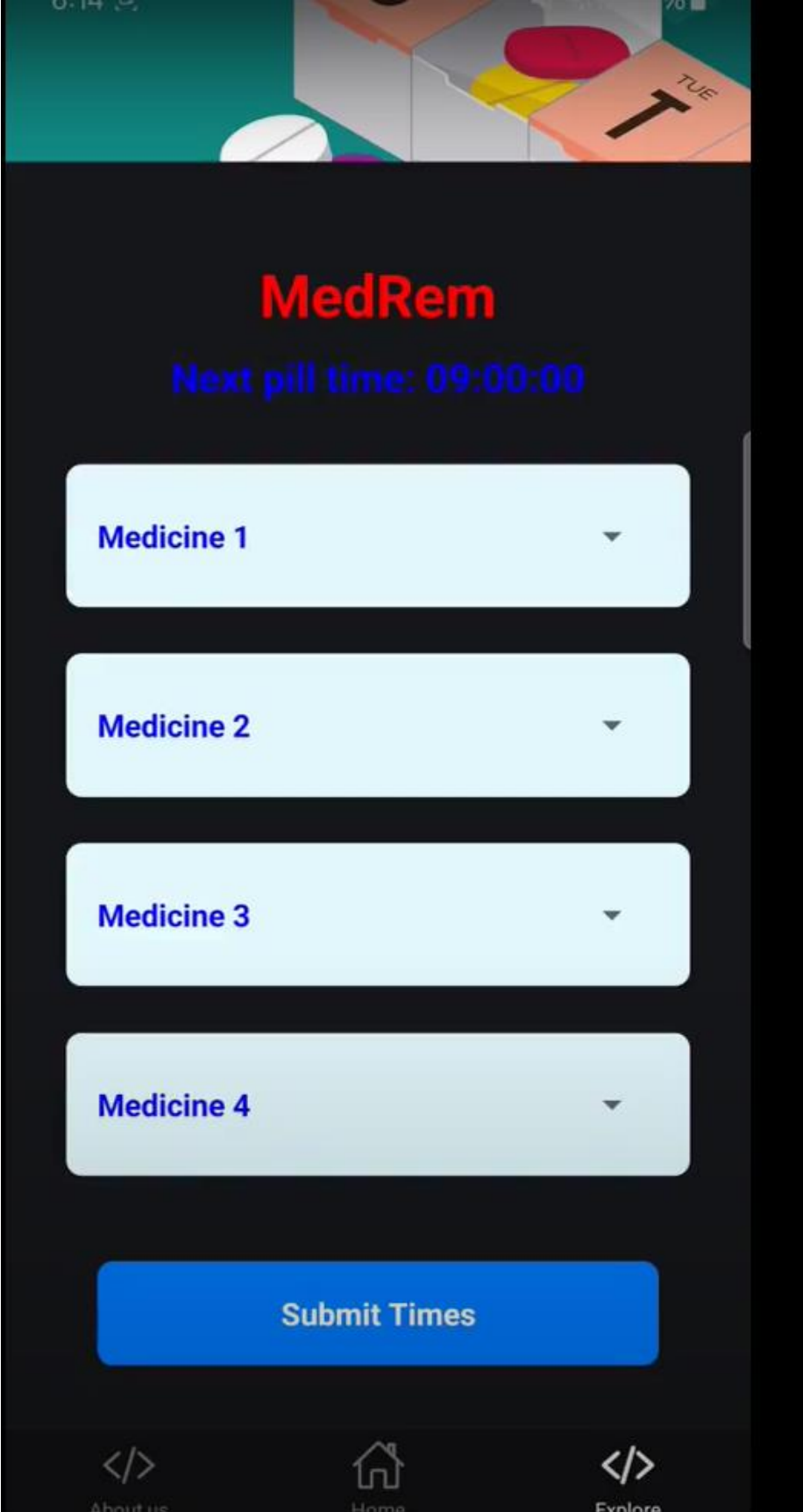
4.4.6 Pill dispensing:

The patient first synchronizes the machine with the mobile application and then selects for each container the number of times to take in a day, whether once a day or twice a day or three times a day. Then, when it's time for the patient to take his medicine, the servo would move to the next sector, dropping a pill to the patient drawer, and the pressure sensor would detect if there were a pill dispensed. If not, the servo would move to the next sector, and when a pill is detected, the system would turn a buzzer for 5 seconds to alarm the patient to take his pills.

4.4.7 Mobile app:

Where the patient or the caretaker can sync time with the device to have the same time and then can submit the number of times to take for a certain medicine and also to keep track of the number of pills remaining in each container.





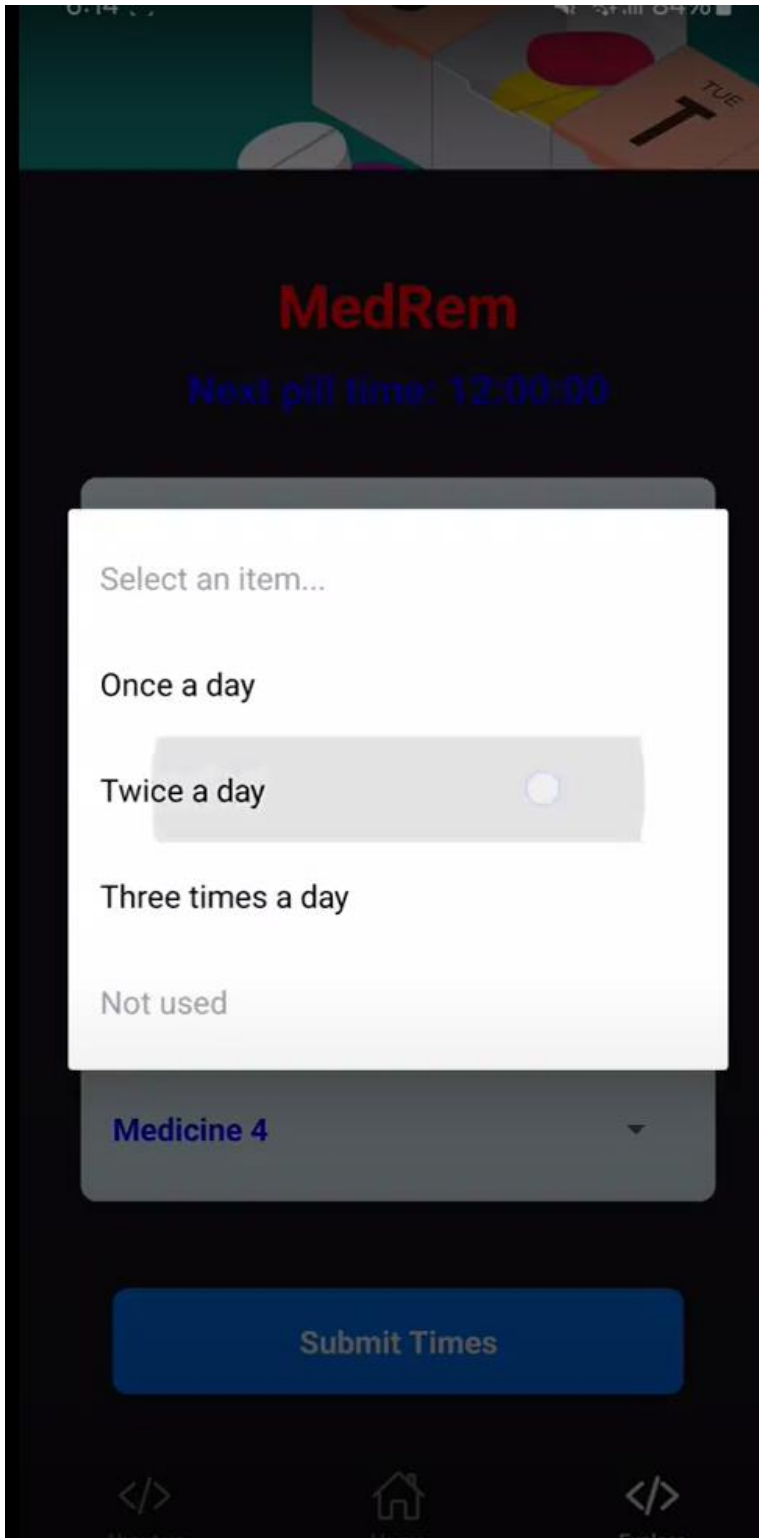


Figure 37: Mobile app Screens

5 Conclusion/ Recommendations and Future Works

The main goal for this application is to preserve the lives of the elderly. I was able to create this system after months of researching, learning and testing new things and in the end I was able to create a system that stores the pills in a container and have a detection system and a conveyer made out of plastic to move the torque from the servo to create momentum to move one sector at a time and also have an alarm system to notify the patient to take his medicine in time and a screen to inform the patient about the current state of the system and also have a server running in the esp32 to accept clients (request) to sync time and also set the timers for each container(medicine)and have 4 different containers that contain up to 7 pills.

And for future work I hope to create a system that contain more types of medicines and for each medicine more pills to store and to create a detection system that have a 100% (my system accuracy is around 95%) and also find a way to power the esp32 and the Arduino internally to create a system that is independent.

6 References

- [1] R. Horne et al., "Concordance, adherence and compliance in medicine taking," NCCSDO, London, 2005, no. 40, p. 6
- [2] N. Zelikovsky and A. P. Schast, "Eliciting accurate reports of adherence in a clinical interview: development of the Medical Adherence Measure," *Pediatric Nursing*, vol. 34, no. 2, pp. 141-146, 2008.
- [3] C. A. Walsh, C. Cahir, S. Tecklenborg, C. Byrne, M. A. Culbertson, and K. E. Bennett, "The association between medication non-adherence and adverse health outcomes in ageing populations: a systematic review and meta-analysis," *British Journal of Clinical Pharmacology*, vol. 85, no. 11, pp. 2464-2478, 2019.