



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



**Sahar Saleh**  
**Hala Abduljalil**

**Project supervisors:**  
**Dr. Aladdin Masri**

## Acknowledgement

First and foremost, praise be to God Almighty and thanked him for our success in completing this project after a long and exhausting effort. We would also like to express my sincere gratitude and thanks to our supervisor, Dr. Aladdin Masri, for his patience and for directing us with his comments, which were always appropriate for the success and development of the project, and we do not forget to provide him with information and ideas that helped us in completing the project. Also want to thank the doctors of our Department of Computer Engineering for their patience and cooperation throughout the project completion period.

We are deeply grateful to the Qaddumi foundation for it is support during our years in the engineering department.

We would also like to commend our parents and families for their continuous support, love, care, and prayers for us, and their encouragement towards a bright future. Finally, we thank all the people who have supported us directly or indirectly in completing the research work.

## Disclaimer

This report was written by Sahar Saleh and Hala Abduljalil at the Computer Engineering Department, Faculty of Engineering, An-Najah National University. It has not been altered or corrected, other than editorial corrections, as a result of assessment and it may contain language as well as content errors. The views expressed in it together with any outcomes and recommendations are solely those of ours. An-Najah National University accepts no responsibility or liability for the consequences of this report being used for a purpose other than the purpose for which it was commissioned.

## Contents

1	Introduction .....	6
1.1	Organization of the report.....	7
2	Constraints and earlier work.....	8
2.1	Constraints.....	8
2.2	Earlier Work .....	8
3	Literature review.....	9
4	Methodology.....	10
4.1	Hardware Components .....	10
4.1.1	ESP32-Wroom.....	10
4.1.2	ESP32-CAM.....	11
4.1.3	chassis body.....	11
4.1.4	DC Motors .....	12
4.1.5	L298N Driver .....	12
4.1.6	Wheels .....	14
4.1.7	12 Volt Battery .....	15
4.1.8	MG995 Servo Motor .....	15
4.1.9	Camera Mount .....	16
4.1.10	Buck Converter.....	16
4.1.11	Micro Servo 9g .....	17
4.1.12	Infrared Sensor .....	18
4.1.13	Ultrasonic Sensor.....	18
4.1.14	MQ4 Gas Sensor.....	19
4.1.15	LDR Sensor Module.....	20

4.1.16	PIR Motion Sensor .....	21
4.1.17	Wires.....	22
4.2	Frameworks .....	22
4.3	Tools .....	22
4.4	Technical Choices.....	23
4.4.1	Web page Application .....	23
4.4.2	Process of work .....	25
5	Conclusion and Future Work.....	30

## List of Figures

Figure 1:	ESP32-Wroom .....	10
Figure 2:	ESP32-CAM.....	11
Figure 3:	chassis body.....	11
Figure 4:	JGA25 -Motor .....	12
Figure 5:	DC-Motor with Driver .....	12
Figure 6:	L298N Driver .....	13
Figure 7:	Driver Pins .....	13
Figure 8:	Wheels.....	14
Figure 9:	12Volt Battery .....	15
Figure 10:	MG995 Servo Motor.....	15
Figure 11:	Camera Mount .....	16
Figure 12:	Buck Converter .....	16
Figure 13:	Micro Servo 9g .....	17
Figure 14:	IR Sensor.....	18
Figure 15:	Ultrasonic Sensor .....	18
Figure 16:	MQ4 Gas Sensor.....	19
Figure 17:	LDR Sensor.....	20
Figure 18:	LEDS.....	20
Figure 19:	PIR Motion Sensor.....	21
Figure 20:	Wires .....	22
Figure 21:	Webpage that control the robot .....	23
Figure 22:	Webpage on Autonomous mode. ....	24
Figure :	process of work figures 23 .....	29

## Abstract

The world suffers from many natural disasters such as earthquakes, volcanoes, fires and others, humanity stands only before discovering those dangers, and there are many examples, and the results of exploration attempts endanger humans, Hence the idea of our project to try to reduce this inevitable danger to humanity and to create an automated alternative that performs the task of exploring danger in smart and technological ways without exposing humans to any danger, in addition to collecting the largest possible amount of useful information that can carry the warning inside.

Our project is a Surveillance robot used to monitor and detect certain areas the robot is sent to, by filming and displaying a streaming video, the robot's movement is controlled by two modes:

- The first mode: controlling its movement through a website, where it can move forward, backward, right and left, in addition to controlling the speed of the robot also the amount of Flash, and the camera angle by using Servo.
- Autonomous mode: Movement individually where the robot moves alone, changing its movement to the right or left if a foreign object appears in front of it.

To make and complete this project, we will mainly rely on sensitive sensors of different types, which will help us in exploring the hazards, we will use the camera that enables us to display the vision of the areas, and then we will use it additionally. For all the hardware requirements to build a bot architecture, and finally use a website with HTML, CSS and vanilla JavaScript to control it.

# 1 Introduction

Our main goal before choosing the project was to find a real problem that a group of people suffer from and to try to find a smart solution that contributes to reducing the occurrence of this problem, so once the problem was chosen, we knew which direction we would go towards the project. 1-Controlling the robot through a web page, through which the robot can be controlled forward, backward, rotate right and left on web page.

The problem we chose was the danger to humans when exploring disasters, and the solution was to design an alternative robot for the human task in exploration, but above all we read in depth about the problem we identified as the title of our project and defined the features in which we want to provide the robot to support the solution. Our project goal is to employ as many features as possible to help detect any threat that exists.

Of course, when we identified the problem, there is a category that we targeted for our project and they are people who go to explore areas after disasters to collect information about the place and explore the victims and injured and speed up access to them.

## 1.1 Organization of the report

The next chapter (Constraints and Earlier Work) will focus on the main issues encountered while working on this project and will mention previous work that helps us to build the application.

Next comes the Literature Review chapter, which introduces and explains some of the relevant works, and shows how this work differs. In chapter four (Methodology) we will talk about our methods of building this project and discuss it in detail. The last chapter is Conclusions , in this chapter, the conclusions will be discussed also, we will mention future work.

## 2 Constraints and earlier work

### 2.1 Constraints

It is normal when working on a project that we encounter many limitations and problems that are difficult to do. The limitations were an obstacle for us in our project, examples of these limitations in terms of software when adding a certain library that is not suitable for the microcontroller used, also some libraries that were not suitable for the special version type with Arduino, also a problem with the communication of two microcontrollers with each other ( esp32 WROOM, esp32CAM), this problem took a lot of time until we found the right solution for it, the solution was to make esp32 as server and esp32CAM as client-server, and the way to communicate between them was API requests. As for the hardware constraints, they are: First, the network we were using in the project testing process was poor in sending requests between two microcontrollers. The second issue was the nature of the infrared Sensor work in a dark environment.

### 2.2 Earlier Work

At the beginning of our project, we gained sufficient experience from the microcontroller course in reading the datasheet for each component and how to deal with it, in addition to the previous programming courses (c++ for how to write code on microcontrollers, Network course for how to send request from the first microcontroller to another), also we learned program electronics on an Arduino.

### 3 Literature review

Before starting the project, we definitely read previous research and articles on various topics, in order to expand our understanding and ideas to solve the problem we set. Previous experiences with the exploratory robot differed from one robot to another. Some of them made an exploratory robot in Indonesia directed to rescue the victims only, as it was made of 3 ultrasonic sensors to avoid obstacles, in addition to a camera to recognize the faces of the victims, as this robot was only moving independently. [1] Hence the first distinction of our project, where the robot is controlled in two ways, in addition, it has many exploratory features that are not limited to obstacles, but send an image when you see anything it senses, a live video broadcast, it senses whether the object Living being, it also senses gas, and if the environment is dark, the place is luminous, also don't forget to send a notification to Signal.

Another example we read about was making a robot, and this robot was intended for more than one monitoring environment, but it was weak with the features supporting the robot [2] Thus, we focus on adding many features to an exploratory environment within stakeholders.

It is worth noting that we are not the first to come up with this idea, but we did a lot of research and identified limited problems to develop solutions for them, which helped us to supply the robot with many features that support our basics. The idea becomes a door to distinguish us from previous projects that carried the same idea with different characteristics and categories.

## 4 Methodology

In this section, each piece is identified individually, the reason for its use, and the pins we used in this piece to connect it with the ESP32.

### 4.1 Hardware Components

In this section, each piece that was used in the project will be identified:

#### 4.1.1 ESP32-Wroom

This microcontroller was used as a server where it receives an API request from the ESP32 CAM, and was used to define modes which is control and autonomous for motion, and to program several features.



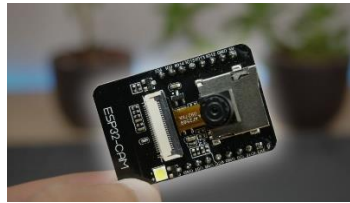
*Figure 1: ESP32-Wroom*

The pins that we used on ESP32-Wroom:

- GPIO34 Pin: Connected with output Pin of Gaz Sensor.
- GPIO25 Pin: Connected with output Pin of LDR Sensor.
- GPIO27 Pin: Connected with echo Pin of Ultrasonic Sensor.
- GPIO12 Pin: Connected with Driver Pin.
- GPIO13 Pin: Connected with Driver Pin.
- GPIO14 Pin: Connected with Driver Pin.
- GPIO15 Pin: Connected with Driver Pin.
- GPIO16 Pin: Connected with Output-orange- Wire of servo for ultrasonic.
- GPIO17 Pin: Connected with Output-orange- Wire of servo for camera.
- GPIO5 Pin: Connected with led for LDR.
- GPIO19 Pin: Connected with buzzer just for test.
- GPIO27 Pin: Connected with Trigger Pin of Ultrasonic Sensor.
- GPIO22Pin: Connected with output Pin for IR Sensor.

#### 4.1.2 ESP32-CAM

The ESP32 CAM microcontroller was used to create a video stream of the area we sent the robot to, and it was also used to show the robot's website using its IP, also it was used to take a capture of the area and send it via Telegram if there was movement.



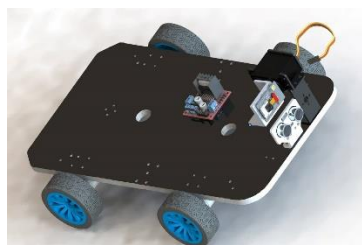
*Figure 2: ESP32-CAM*

This module we used three pins:

- VCC Pin: connected with 5 volts.
- GND Pin: connected with 0 volt.
- GPIO12 Pin: connected with output pin from Motion Sensor.

#### 4.1.3 chassis body

We used a piece of wood to put the components on it such as motors ,battery base , dc-driver.



*Figure 3: chassis body*

#### 4.1.4 DC Motors

We used a four JGA25 motor to convert the electrical energy coming from the batteries into mechanical energy to move the wheel of the robot we connected the motor wires with the driver



Figure 4: JGA25 -Motor

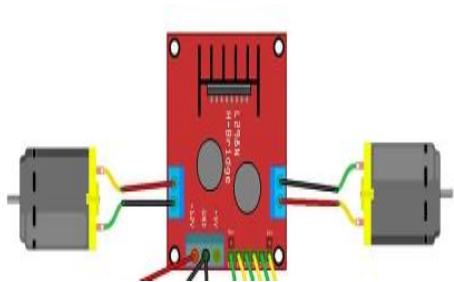


Figure 5: DC-Motor with Driver

#### 4.1.5 L298N Driver

we control JGA25 motors. The easiest and affordable way to control JGA25 motors is to interface the L298N motor driver with the esp32. It can control both the speed and the spinning direction of four JGA25 motors.

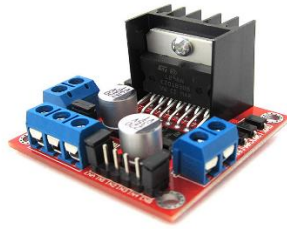


Figure 6: L298N Driver

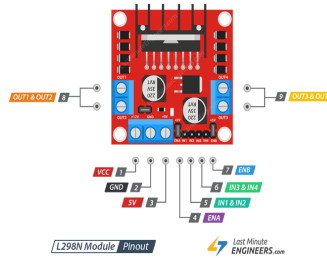


Figure 7: Driver Pins

This module consists of 11 pins:

- VS Pin: We connect the input voltage coming from the batteries to 12 volts to this pin.
- VSS Pin: We used to drive the logic circuitry L298N which can be 5V.
- GND Pin: ground pin.
- OUT1, OUT2, OUT3, OUT4 pins: we used these pins to control the JGA25 -motors.
- IN1, IN2, IN3, IN4: We use these pins to control whether the motor is spinning forward, backward, left or right or autonomous. The direction of the motor can be controlled by applying the value taken from speed slider and connected with pin12,13,14,15 from ESP32.

#### 4.1.6 Wheels

We used four wheels to move the robot.



*Figure 8:Wheels*

#### 4.1.7 12 Volt Battery

12 volts from battery was used to supply the L298N Driver with 12 volts.



*Figure 9: 12Volt Battery*

#### 4.1.8 MG995 Servo Motor

We used the servo to control the direction of movement of the camera from top to bottom in different degrees.



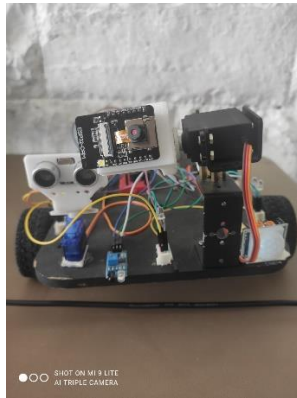
*Figure 10: MG995 Servo Motor*

This module consists of 3 pins:

- Orange wire: This wire is connected with pin 17 with ESP32 to control the direction of the camera.
- Red wire: connect with 5 volts.
- Brown wire: connect with GND.

#### 4.1.9 Camera Mount

It is used to place and carry the ESP32-CAM.



*Figure 11: Camera Mount*

#### 4.1.10 Buck Converter

It is used to reduce the voltage coming from the batteries which is 12V to 5V to supply ESP32 with a Fixed voltage of 5V.



*Figure 12: Buck Converter*

This module consists of 4 pins:

- IN+(12V): It is connected to a 12-volt battery.
- IN-: It is connected to GND.
- OUT+(5V): It is connected to VCC of ESP32.
- OUT-: It is connected to GND of ESP32.

#### 4.1.11 Micro Servo 9g

It is used to move the ultrasonic in order to choose which direction the robot goes, is it right, forward or left.

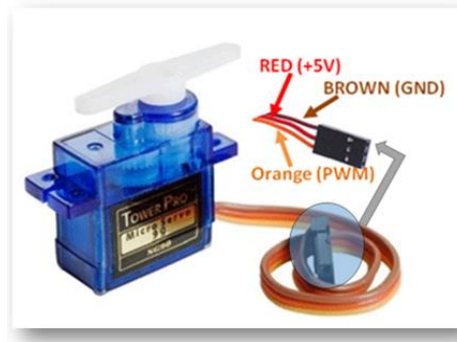


Figure 13: Micro Servo 9g

This module consists of 3 pins:

- Orange wire: This wire is connected with the pin 16 ESP32 to measure the distance using the ultrasonic
- Red wire: connect with 5 volts.
- Brown wire: connect with GND.

#### 4.1.12 Infrared Sensor

It was used to sense some object of the surroundings also can measure the heat of an object, as if there is body heat, a notification will be sent to the Signal program, and as soon as the notification reaches the relevant authorities where they will rush to rescue the wounded as an example.



Figure 14:IR Sensor

This module consists of 3 pins:

- VCC Pin: connected with 5volt.
- GND Pin: connected with GND.
- OUT Pin: Active High Output and connected in pin 22 from ESP32.

#### 4.1.13 Ultrasonic Sensor

It measures the distance between the robot and the object it collides with. If the distance is less than 35, then he compares the distance to the right, left, and the top goes towards it and if both higher than 35 it goes forward .



Figure 15: Ultrasonic Sensor

This module consists of 4 pins:

- VCC: It is connected with system VCC with 5V.
- GND: It is connected with system ground.
- Trig: signal output pin, it is connected with pin 21 of the ESP32.
- Echo: signal input pin, it is connected with pin 27 of the ESP32.

#### 4.1.14 MQ4 Gas Sensor

There may be a cooking gas (methane) leak in the area to which the robot has been sent, and if there is a leak, a warning notification is sent to Signal program, so that the necessary precautions are taken.



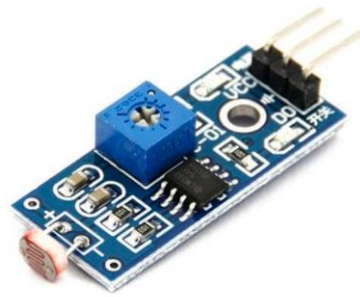
*Figure 16: MQ4 Gas Sensor*

The pins of this module:

- VCC Pin: connected with 5 volts.
- GND Pin: Connected with 0 volt.
- AO Pin: Connected with Pin34 from ESP32.

#### 4.1.15 LDR Sensor Module

If the robot entered a dark area then the LEDs connected with LDR will be on , to give a better lighting for streaming .



*Figure 17: LDR Sensor.*



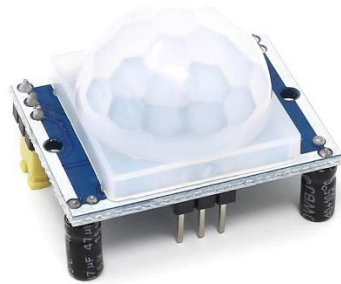
*Figure 18: LEDs.*

The pins of this module:

- VCC Pin: connected with 5 volts.
- GND Pin: Connected with 0 volt.
- AO Pin: Connected with Pin 25 from ESP32

#### 4.1.16 PIR Motion Sensor

This sensor was used to detect any movement. When there is movement, a momentary picture will be taken from ESP32 CAM and uploaded to Telegram.



*Figure 19: PIR Motion Sensor.*

The pins of this module:

- VCC Pin: connected with 5 volts.
- GND Pin: Connected with 0 volt.
- D-out Pin: Connected with pin12 from ESP32-CAM.

#### 4.1.17 Wires

Wires of different lengths and types were used to connect the components.



Figure 20: Wires

## 4.2 Frameworks

We have used a webpage using a HTML, CSS, then we added the JavaScript to make the button as a toggle button to work as two mode: the first mode (control mode), and the second one is (Autonomous mode).

## 4.3 Tools

We have used 2 programs that we used to send a notification:

1. Signal program: it used to send a notification if there a gas leakage, also if there an object.
2. Telegram application: it used to upload a photo that taken from ESP32 CAM, if there found a motion by using PIR Motion Sensor.

## 4.4 Technical Choices

### 4.4.1 Web page Application

in the figure below show the webpage that we can open by using PC or mobile by API, we can control the robot forward (almost two meter), backward (almost 1 meter back), turn left (turn the robot 45 degree to left), turn right it the same functionality but to right, finally stop button to stop the robot. Also, the webpage contains a flash slider to control the power of the flash (0-255). on addition the speed slider to control robot speed (0-255), Finally we have servo slider to control degree of freedom of camera servo to change streaming angle.

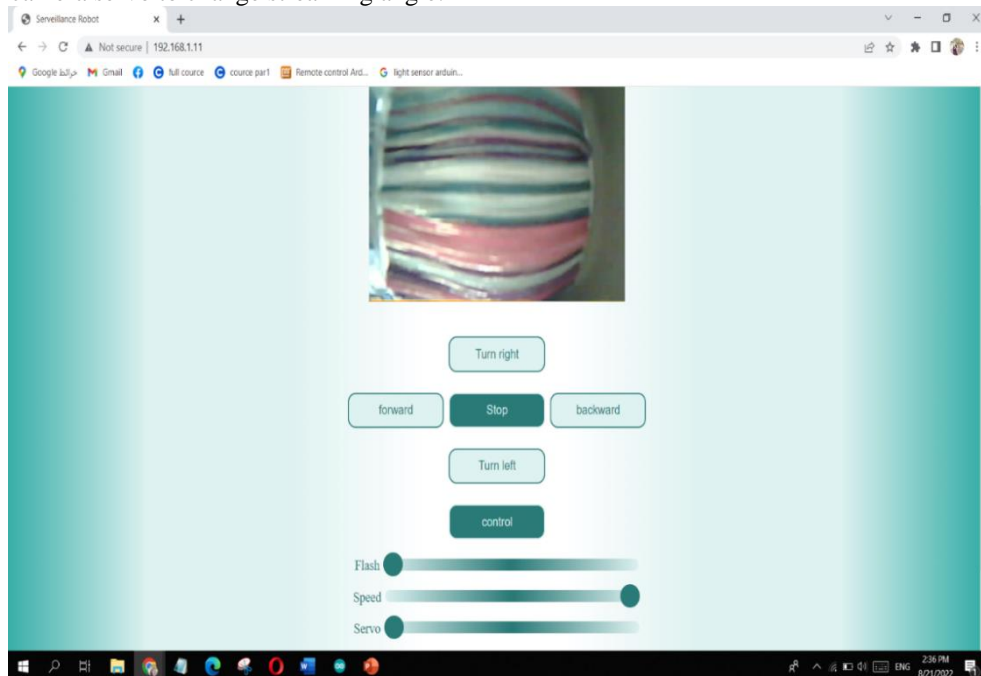


Figure 21: Webpage that control the robot.

As we said above, the robot controls mode 2, so in the webpage when you switch the button, it sets the robot to the autonomous mode, which makes the robot walk forward but if you sense an object less than 35cm using (ultrasonic sensor), the robot will look left and right (using the servo) and compares the large distance and then takes it as his direction.

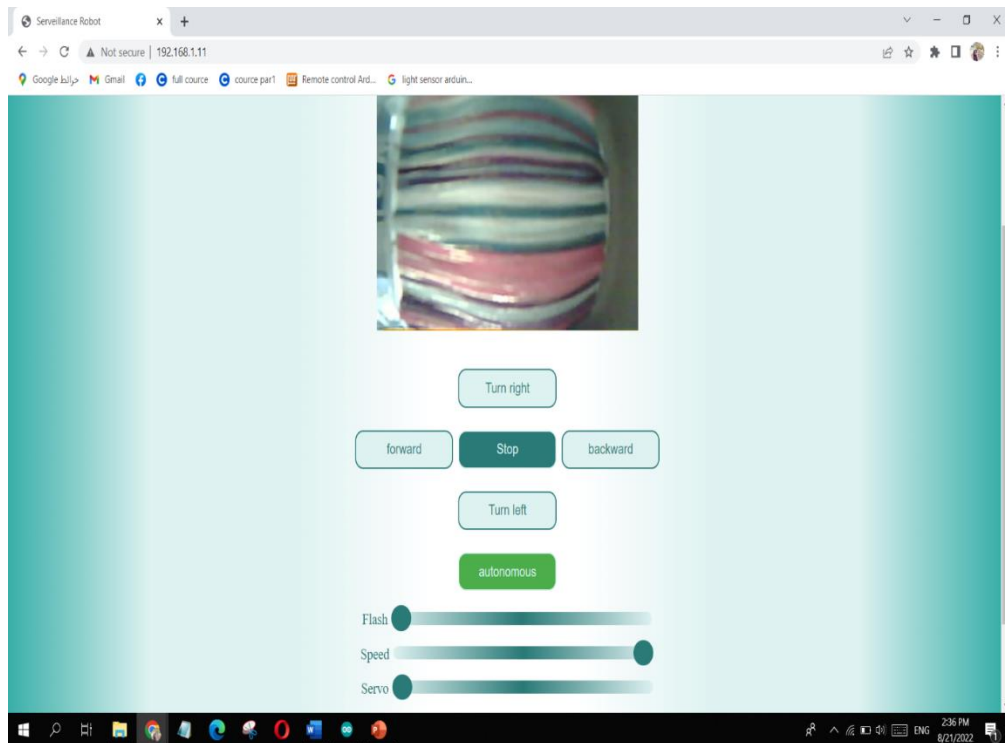
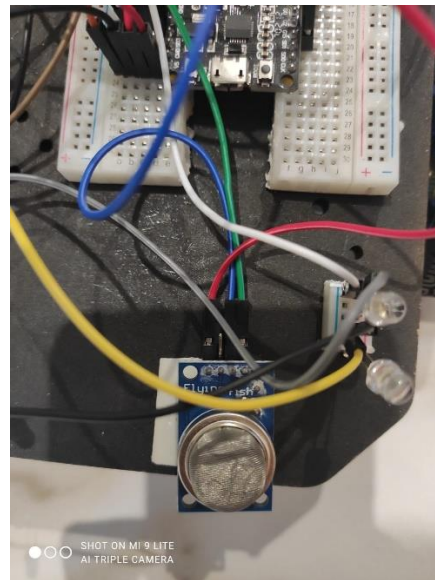
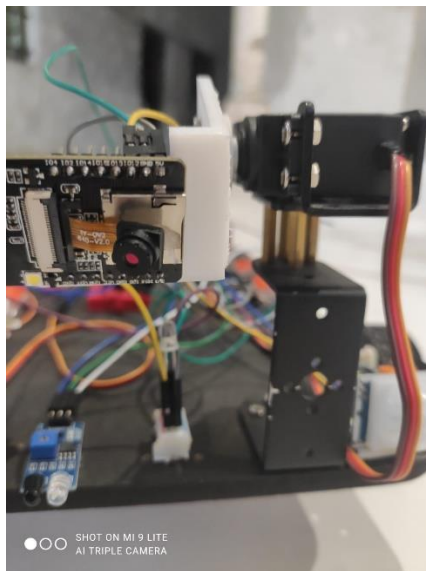


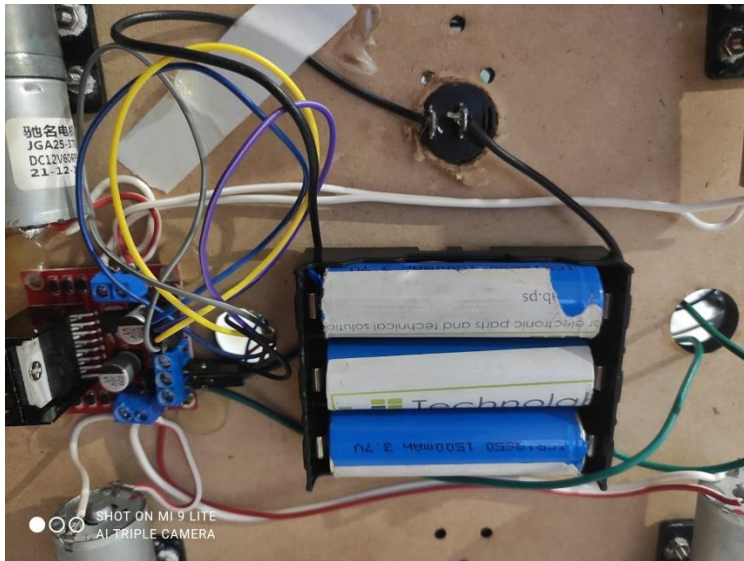
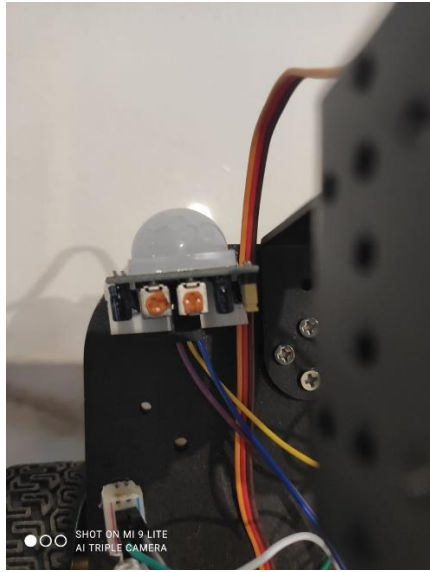
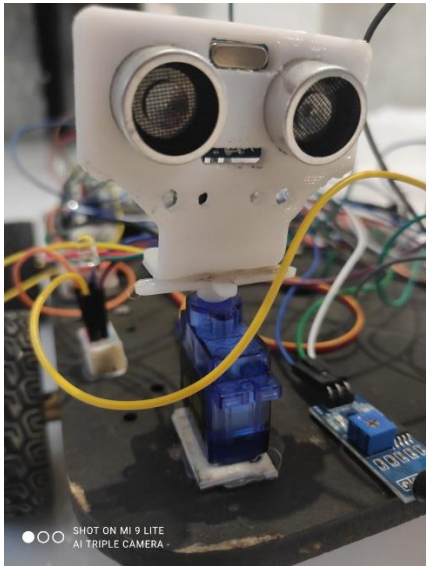
Figure 22: Webpage on Autonomous mode.

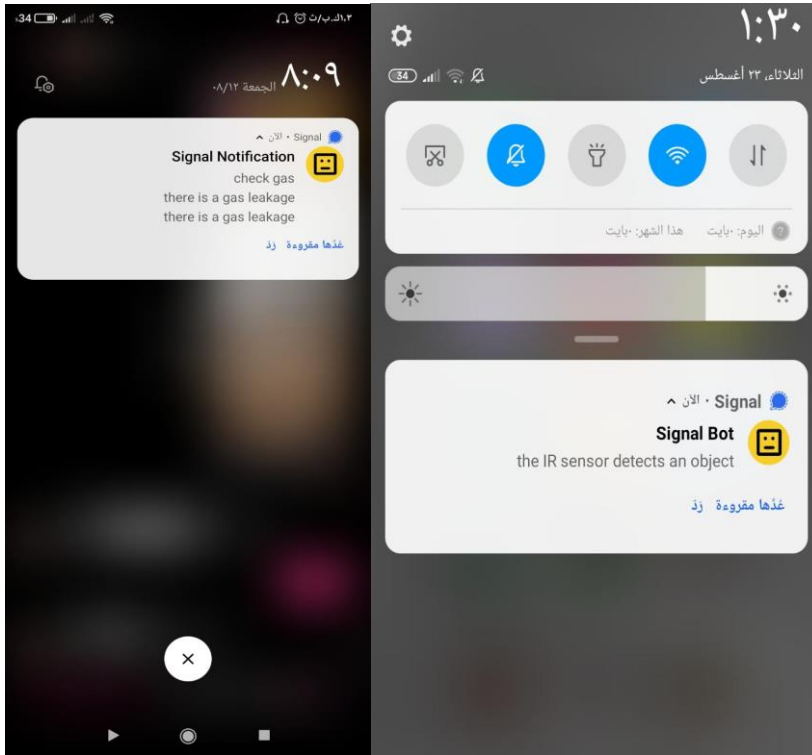
#### 4.4.2 Process of work

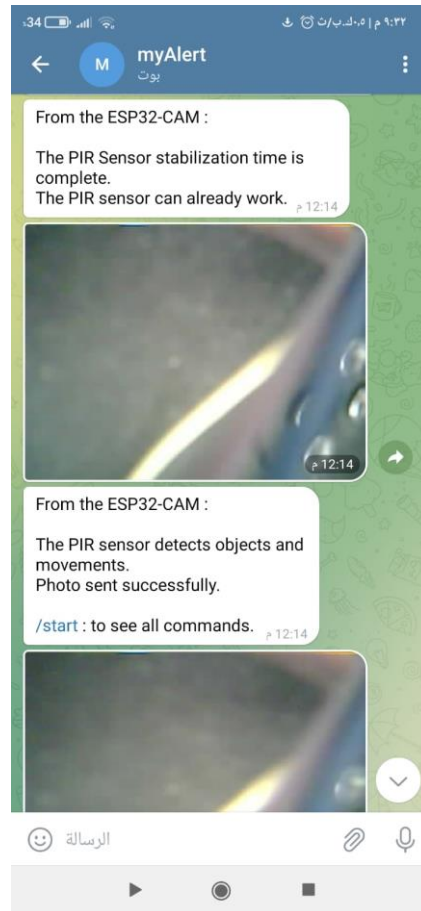
We'll display and discuss process of work of our project in this chapter , we started by testing streaming on ESP32 camera without connect it with esp32 ,after it worked we then moved to connect it with esp32 WROOM and this was the hardest part of our project to make two microcontrollers act like one microcontroller .after that we started to add feature by feature , and test every one alone without other features when it works we edited the main code by adding this new feature code , then we tested the whole code again and we repeated this process . we used apps like signal and telegram to send us notification for some features.

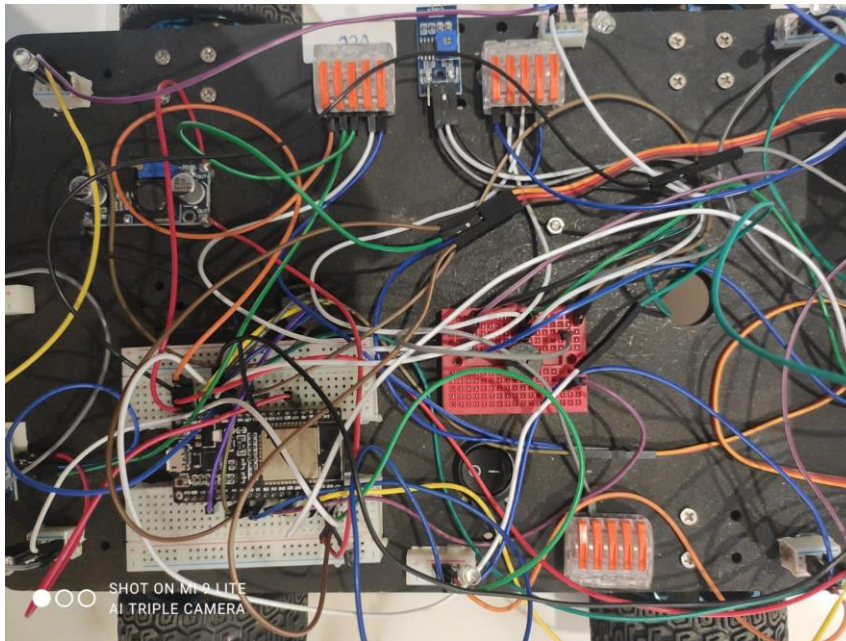
And here are some figures for our process:











23 Figure :process of work figures

## 5 Conclusion and Future Work

As we presented above, our project targets a specific group of people, and they are people who explore risks and disasters, so our project can greatly help them in protecting them and reducing the danger around them.

We hope to use and develop this project by identifying some additional risks so that solutions are found that help them explore these risks, and we also hope when implementing the project to use equivalent and appropriate pieces of those pieces that we used, but with higher quality and accuracy, such as changing the type of wheels to suit the difficult environment, for example The robot is able to climb over the accumulated stones without any difficult.

## References

[1] Budiharto, W., 2015. Intelligent surveillance robot with obstacle avoidance capabilities using neural network. *Computational intelligence and neuroscience*, 2015.

<https://www.hindawi.com/journals/cin/2015/745823/>

[2] Di Paola, D., Milella, A., Cicirelli, G. and Distante, A., 2010. An autonomous mobile robotic system for surveillance of indoor environments. *International Journal of Advanced Robotic Systems*, 7(1), p.8.

<https://journals.sagepub.com/doi/10.5772/7254>