

IoT Technology for smart life and sustainable development



A capstone project submitted to the faculty of Industry and Energy Technology, New Cairo Technological University, In partial fulfillment of the requirements for the Degree of Higher Diploma

Information and Communications Technology

SUBMITTED BY

NADER WAEL GALAL

IBRAHIM ARAFA ELSAYED

SHIMAA SAIED

OMAR ANWAR

MAHMOUD HOSSAM ELDINE

AHMED EHAB

EYAD OSAMA

MOHAMED GAMAL ISSA

KAREEM SAYED

ABDELRAHMAN AHMED ABDELAZIZ

SUPERVISED BY

Prof.Dr. Adly Tageldin Eng. Shereen Eng. Maha

STUDENT DECLARATION

Students declaration

We certify that the capstone submission is entirely our work and we fully understand the consequences of plagiarism. We understand that making a false declaration is a form of malpractice.

Also, we acknowledge that we have received feedback about our work from the assessor.

Name	Signature
Nader Wael Galal	-----
Ibrahim Arafa Elsayed	-----
Abdelrahman Ahmed Abdelaziz	-----
Mahmoud Hossam Eldine	-----
Ahmed Ehab	-----
Shimaa Saied	-----
Eyad Osama	-----
Omar Anwar	-----
Kareem Sayed	-----
Mohamed Gamal Issa	-----

Date: 25 / 7 / 2022

Approval Sheet

IoT Technology for smart life and sustainable development

A capstone project submitted to the faculty of Industry and Energy Technology, New Cairo Technological University, In partial fulfillment of the requirements for the Degree of Higher Diploma of Information and Communications Technology.

Submitted by

Nader Wael Galal
Ibrahim Arafa Elsayed
Ahmed Ehab
Eyad Osama
Omar Anwar

Mahmoud Hossam Eldine
Abdelrahman Ahmed Abdelaziz
Shimaa Saied
Kareem Sayed
Mohamed Gamal Issa

This capstone project has been approved by the examining committee:

Score:-----

Name of the Examiner

Signature

Prof.Dr. Adly Tageldin

Information and Communications Technology Department
faculty of Industry and Energy Technology
New Cairo Technological University

Dr. Hatem Yousry

Information and Communications Technology Department
faculty of Industry and Energy Technology
New Cairo Technological University

Dr. Rasha Stohy

Information and Communications Technology Department
faculty of Industry and Energy Technology
New Cairo Technological University

DEDICATION

This project is dedicated to our parents who have never failed to give us financial and moral support, for giving all our needs during the time we developed our system, and for teaching us that even the largest task can be accomplished if it is done one step at a time.

We dedicate this Project to all the people who have worked hard to help us complete this project.

ACKNOWLEDGMENT

First of all, I thank my "*God*" for helping me achieve this work and giving me the ability to finish this thesis satisfactorily.

I would like to express my sincere appreciation to my supervisors: ***Eng. Shereen and Eng. Maha***. I am very grateful for their strong effort, continuous support, and encouragement during the research study in this capstone project. They really influenced my way of thinking and developing the research ideas adopted in this thesis. Really, I can't find the appropriate words to thank them. I am very grateful for their strong effort, continuous support, and encouragement during the research study in this capstone project.

I can't forget my dearest ***Prof.Dr. Adly Tag Eldeen***, Head of Information and Communications Technology Department, the man who learned me not only how to make scientific research but also more diverse things in my practical life.

I would like also to express my deepest thanks to *all the members* of my colleges for their cooperation during the period I spent with them to prepare this work.

I am extremely grateful to *my family, especially my father, and mother.*
Capstone Project Team

ABSTRACT

Making robot car can be remotely controlled by GUI to achieve the goal of sustainable development. If there is a danger to humans in the place in the case of gas leakage or fire. In this case, the location cannot be determined as quickly as necessary to make a decision. Therefore, in light of recent developments and the entry of the internet of things and the use of Graphical User Interface, Software developments, we came up with the idea of making a car by remote control and with IoT devices that measure the temperature and the percentage of gas in the air, and also with GPS so that we can know the location of the malfunction to make the necessary decision to prevent damage. By the science of software development. A website has been created, to present how this project was built, the people working on it, the components, and, the cost of this project. And show the customer how to buy this product.



TABLE OF CONTENTS

ACKNOWLEDGMENT	V
ABSTRACT.....	VI
LIST OF FIGURES	IX
LIST OF TABLES	X
CHAPTER 1	11
1. INTRODUCTION.....	12
1. 1 Overview.....	12
1. 2 Problem Statement.....	12
1. 3 Capstone Project Objective.....	13
1. 4 Capstone Project Methodology.....	13
1. 5 Significance of Proposed Capstone Project	16
CHAPTER 2.....	17
2. Hardware & Implementation	18
2. 1 Introduction.....	18
2. 2 Circuit Diagram	19
2. 3 Specifications, How they work	24
2. 4 Arduino IDE programming language	32
2. 5 Code implementation.....	33
2. 6 Components	38
2. 7 Conclusion	39
CHAPTER 3.....	40
3. Windows form programming with C#.....	41
3. 1 Introduction.....	41
3. 2 Properties of GUI.....	44
3. 3 Basic windows form	47
3. 4 Problem Formulation	51
3. 5 The Applied Approach.....	51
3. 6 Design and code	52
CHAPTER 4.....	60
4. Website Implementation	61
4. 1 Introduction.....	61
4. 2 Languages and technologies	62
4. 3 Code and screens.....	71

4. 4 Summary	75
CHAPTER 5.....	77
5. CONCLUSION AND FUTURE WORK	77
5. 1 Conclusions.....	77
5. 2 Future Works	79
Bibliography	81

LIST OF FIGURES

Figure 1-1 Car with Bluetooth diagram	19
Figure 1-2 Ultrasonic & Servo diagram.....	21
Figure 1-3 Temperature diagram	21
Figure 1-4 Gas Sensor diagram.....	22
Figure 1-5 GPS diagram	22
Figure 1-6 Buzzer diagram	23
Figure 1-7 Final diagram	23
Figure 1-8 Arduino Mega	24
Figure 1-9 Gas Sensor.....	24
Figure 1-10 Temperature Sensor	25
Figure 1-11 Temperature measures	25
Figure 1-12 Ultrasonic Sensor	26
Figure 1-13 Ultrasonic Timing diagram	26
Figure 1-14 Bluetooth	27
Figure 1-15 GPS Module	28
Figure 1-16 L298N Motrор driver.....	28
Figure 1-17 Servo motor.....	29
Figure 1-18 BreadBoard	30
Figure 1-19 Car & accessories	31
Figure 1-20 Buzzer	31
Figure 2-1 Visual studio 2022.....	42
Figure 2-2 Create new project.....	42
Figure 2-3 New project window	43
Figure 2-4 Start windows form	43
Figure 2-5 Basic controls	47
Figure 2-6 Choose event	48
Figure 2-7 Design of page car driver	51
Figure 3-1 Sign in	71
Figure 3-2 Home screen.....	72
Figure 3-3 Team members	72
Figure 3-4 Specifications	73
Figure 3-5 Models.....	73
Figure 3-6 Cart.....	74
Figure 3-7 User data.....	74
Figure 3-8 Contact-us	75

LIST OF TABLES

Table 1-1 – Bluetooth Communication.....	20
Table 1-2 – Car directions.....	20

CHAPTER I

INTRODUCTION

CHAPTER 1

INTRODUCTION

1. 1 Overview

IoT these days very are important because it lowest the cost of workers and can be remotely controlled from a long distance and take action, in our case we use IoT for keeping people from danger in the event of gas leakage or high temperature due to a fire, for example

And knowing the location of the damage through the GPS and measuring the temperature using a sensor and the leakage of gases by using a gas sensor

The car can be controlled by the application of the GUI, and it also displays measurements and location coordinates using the GUI application.

1. 2 Problem Statement

- Humans can't go into a dangerous area filled with natural gas.
- measuring the amount of gas requires sensors.
- sending reads requires communication.
- Detection from where the gas leaked.

1. 3 Capstone Project Objective

- Safety if gas leaked
- Car remote control by using GUI
- Detection car place by using GPS
- Measuring the amount of gas and sending it if It's dangerous or not
- Buy Product by using the Website

1. 4 Capstone Project Methodology

The methodology of this research is summarized as follows:

- IoT:

The Internet of Things or the Internet of Everything is a new and potentially disruptive technology paradigm. It describes several technologies such as RFID, short range wireless communications, and research disciplines that can connect physical objects from the real world to the internet. To implement IoT solutions, they exist software development approaches like Scrum and Kanban, also, others are adapted viz; Large-Scale Scrum, Scaled Agile Framework and so on., and many methods have been proposed such as Ignite | IoT Methodology and IoT Methodology. Most of them have taken agile thinking as a strategy. However, they are monolithic, which is not easy to adopt. Then, a new method is needed to handle the real nature of IoT, particularly, to handle distributed, mobile, and human out-of-the-loop concerns, which can grow

as new products evolve and new problems emerge. This research studies and analyses some of the existing IoT methodologies. Particularly, it focuses on these methodologies Scrum, Kanban, Scaled Agile Framework, Ignite | IoT Methodology, and IoT Methodology. Our study evaluates their capabilities and compares their main characteristics and behaviors in terms of various methodologies of IoT.

- GUI:

Software testing in general and graphical user interface (GUI) testing, in particular, is one of the major challenges in the lifecycle of any software system. GUI testing is inherently more difficult than traditional and command-line interface testing. Some of the factors that make GUI testing different from traditional software testing and significantly more difficult are: a large number of objects, different look and feel, of objects, many parameters associated with each object, progressive disclosure, complex and inputs from multiple sources, and graphical outputs. The existing testing techniques for the creation and management of test suites need to be adapted/enhanced for GUIs, and new testing techniques are desired to make the creation and management of test suites more efficient and effective. In this article, a methodology is proposed to create test suites for a GUI. The proposed methodology organizes the testing activity into various levels. The tests created at a particular level can be reused

at higher levels. This methodology extends the notion of modularity and reusability to the testing phase. The organization and management of the created test suites resembles closely to the structure of the GUI under test.

- WEBSITE :

The aim of the website is to display detailed data about the parts used and about the techniques and languages that were used in this project. We also wanted to include this project in the labor market, so if any of the users wanted to buy our project to include it in some of the things that it is used in, they will buy it on our website, We have also provided tools to communicate with us if some users or critics want to leave some comments for us or communicate with some specific project members

1. 5 Significance of Proposed Capstone Project

With robot car helps to protect humans in the event of a gas leakage in the place without risking the presence of a worker, and the ability to remotely control it using the GUI and locate the gas leakage via GPS, send sensors readings to relief workers if necessary and you can buy the product from the website and see how we build the project.

CHAPTER 2

Hardware & Implementation

CHAPTER 2

Hardware & Implementation

1. 1 Introduction

The project aims to design a car that can be remotely controlled. The robot car contains Arduino microcontroller with basic mobility features. Arduino programs contains instructions mediating between the controller and Arduino car. GUI controller uses different sensors to supervise motion. An appropriate program in the Arduino microcontroller to interact with the GUI controller has to be created. The program has been successfully complied through Arduino IDE to the Arduino microcontroller & loaded into it after proper checking of logic to decrease any loss/damage of hardware. We have to create an application that will provide a user interface to interact with the Arduino powered car. The interface is easy to use and provide feedback from the Arduino microcontroller through Bluetooth after giving instruction to Arduino for various actions through interface via Bluetooth module. The android application is to create with the help of android studio that provide us with more capability & stability. After doing all of this we have test this project thoroughly and find the maximum no. of error & wrong logic in the microcontroller program. After doing this only we can say that we have been able to create as per our goal described

2. 2 Circuit diagram

- Robot Car with Bluetooth

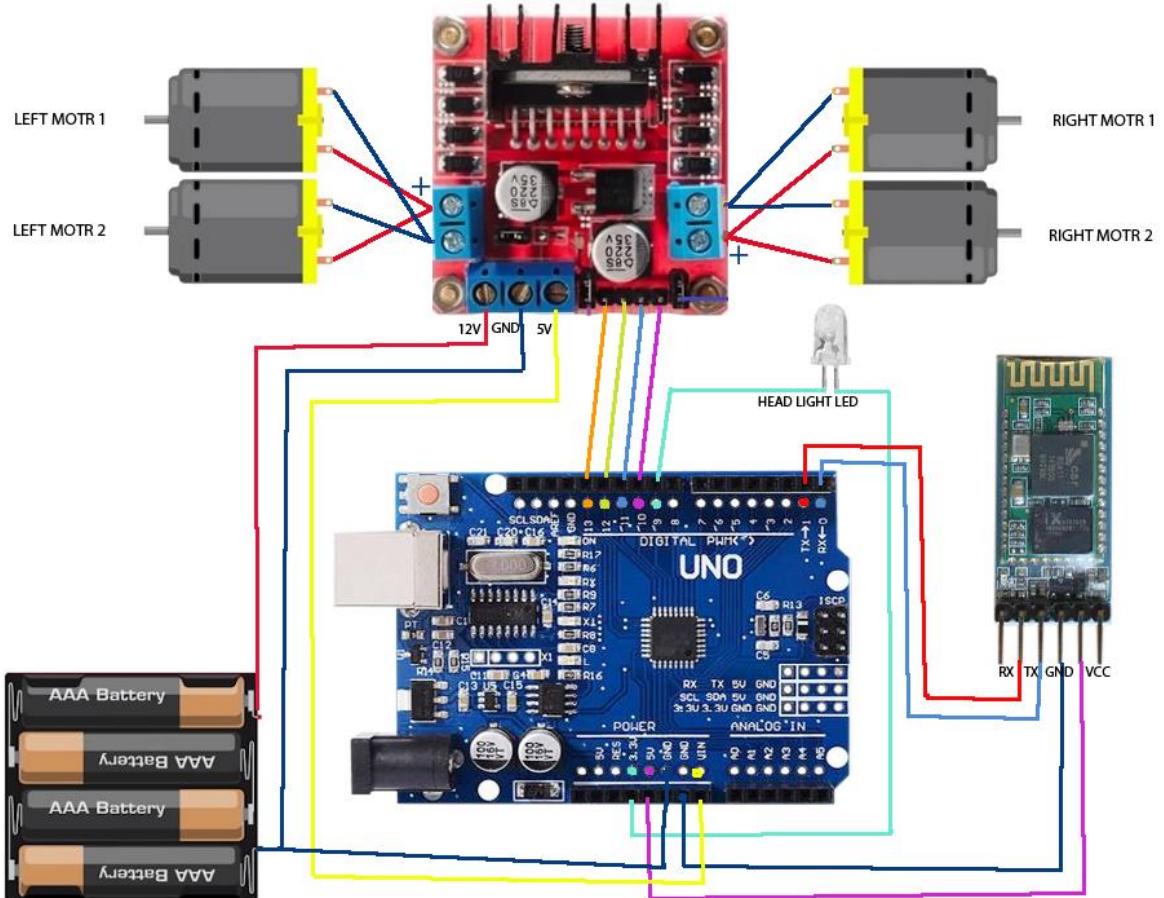


Figure 1-1 - Car with Bluetooth diagram

Resultant & Analysis

Here we work on common mode and when we want to change settings of HC-05 Bluetooth module like change password for connection, baud rate, Bluetooth device's name etc. To do this, HC-05 has AT commands. To use HC-05 Bluetooth module in AT command mode, connect —Key|| pin to High (VCC). Default Baud rate of HC05 in command mode is 38400bps. Following are some AT command generally used to change setting of Bluetooth module. To send these commands, we have to connect HC-05 Bluetooth module to the PC via serial to USB converter and transmit these command through serial terminal of PC.

Command	Description	Response
AT	Checking communication	OK
AT+PSWD=XXXX AT+NAME=XXXX	Set Password e.g. AT+PSWD=4567	OK
	Set Bluetooth Device Name e.g. AT+NAME=MyHC-05	OK
AT+UART=Baud rate, stop bit, parity bit	Change Baud rate e.g. AT+UART=9600,1,0	OK
Change Baud rate e.g. AT+UART=9600,1,0	Respond version no. of Bluetooth module	+Version: XX OK e.g. +Version: 2.0 20130107 OK
AT+ORGL	Send detail of setting done by manufacturer	Parameters: device type, module mode, serial parameter, passkey, etc.

Table 1-1 - Bluetooth Communication

We have created functions for different directions of car. There are five conditions for this Bluetooth controlled car which are used to give the directions:

Touched button in Bluetooth controller app	Output for front and rear side to move forward or reverse or to give direction				
Button	M22	M23	M24	M25	Direction
Forward	0	1	1	0	Forward
Reverse	1	0	0	1	Reverse
Right	0	1	0	1	Right
Left	1	0	1	0	Left

Table 1-2 - Car directions

- Ultrasonic Sensor , Servo motor

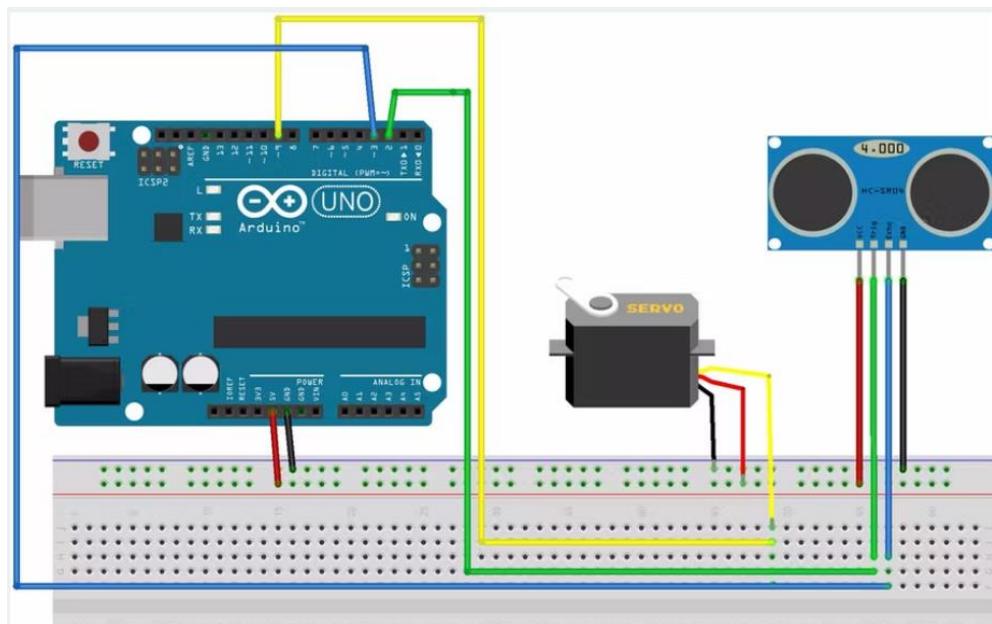


Figure 1-2 - Ultrasonic & Servo diagram

- Temperature Sensor

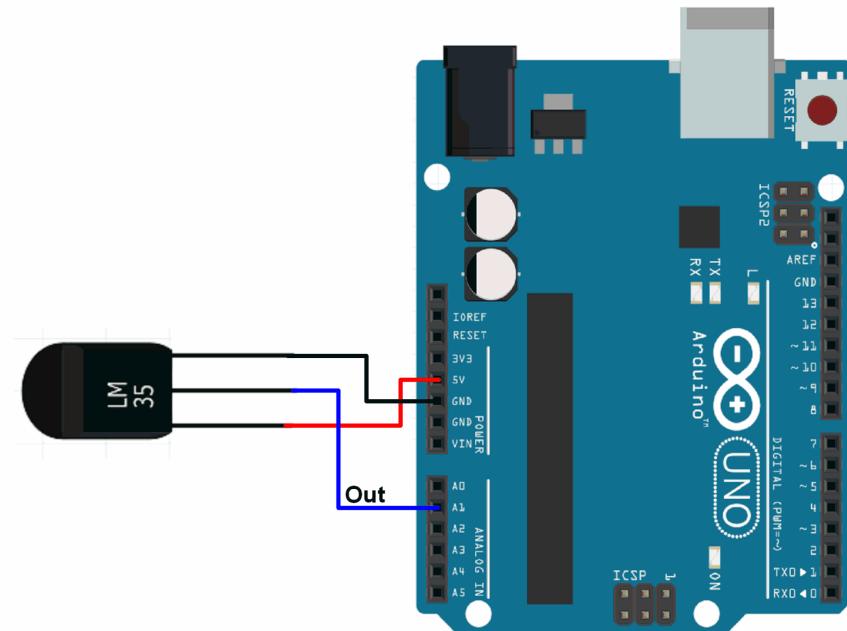


Figure 1-3 - Temperature diagram

- Gas Sensor

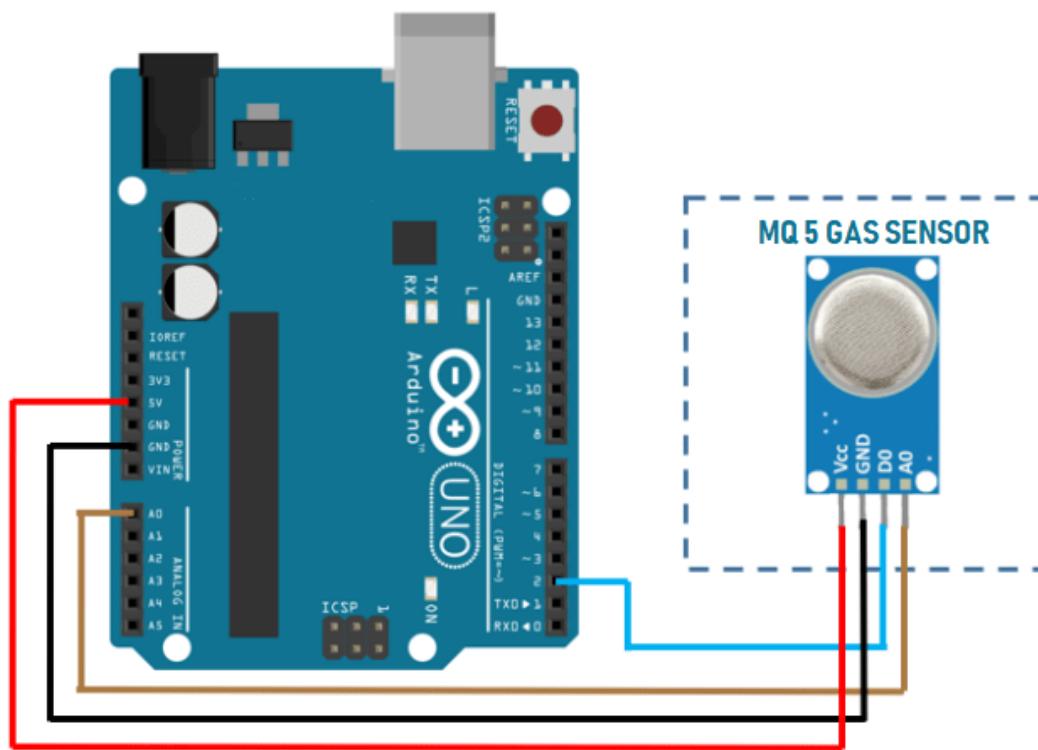


Figure 1-4 - Gas Sensor diagram

- GPS

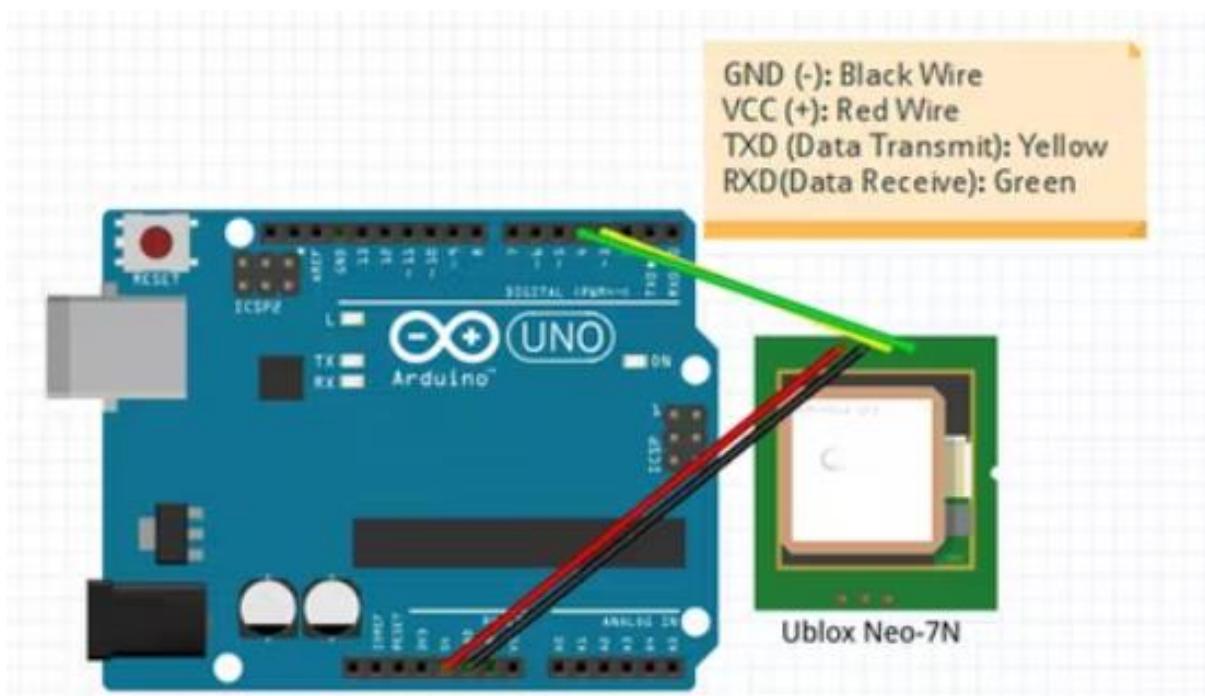


Figure 1-5 - GPS diagram

- Buzzer

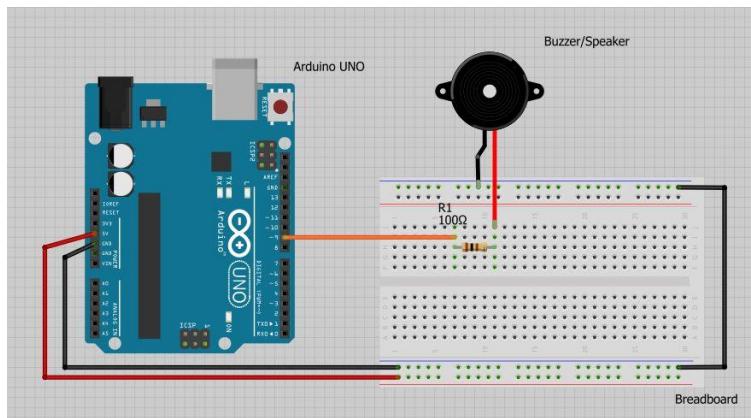


Figure 1-6 Buzzer

- Simulation by using proteus

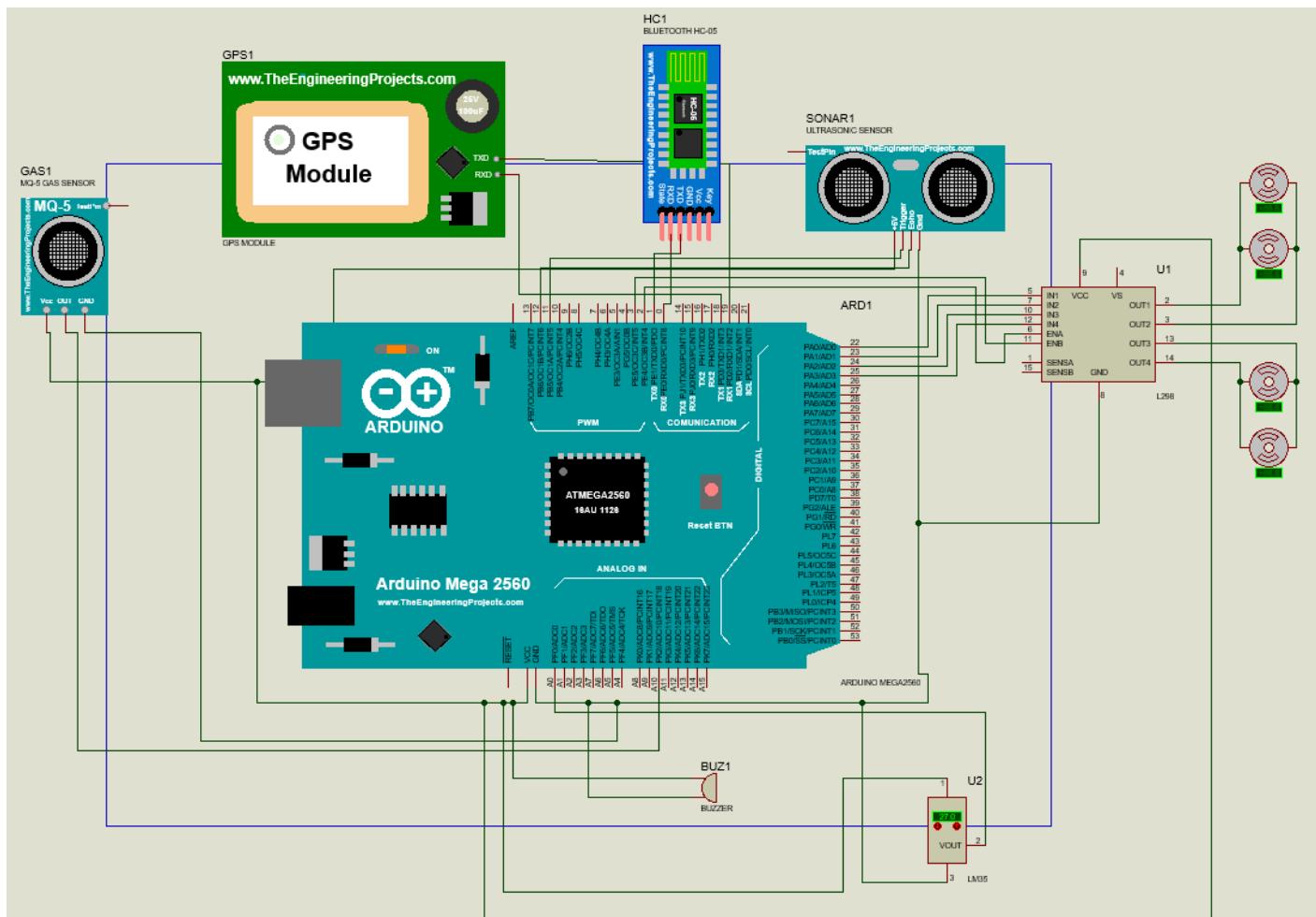


Figure 1-7 Final diagram

1. 3 Specifications, how they work

1. Arduino Mega

The **Arduino Mega 2560** is a microcontroller board based on the ATmega2560. It has 54 digital input/output pins (of which 15 can be used as PWM outputs), 16 analog inputs, 4 UARTs (hardware serial ports), a 16 MHz crystal oscillator, a USB connection, a power jack, an ICSP header, and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started. The Mega 2560 board is compatible with most shields designed for the Uno and the former boards.

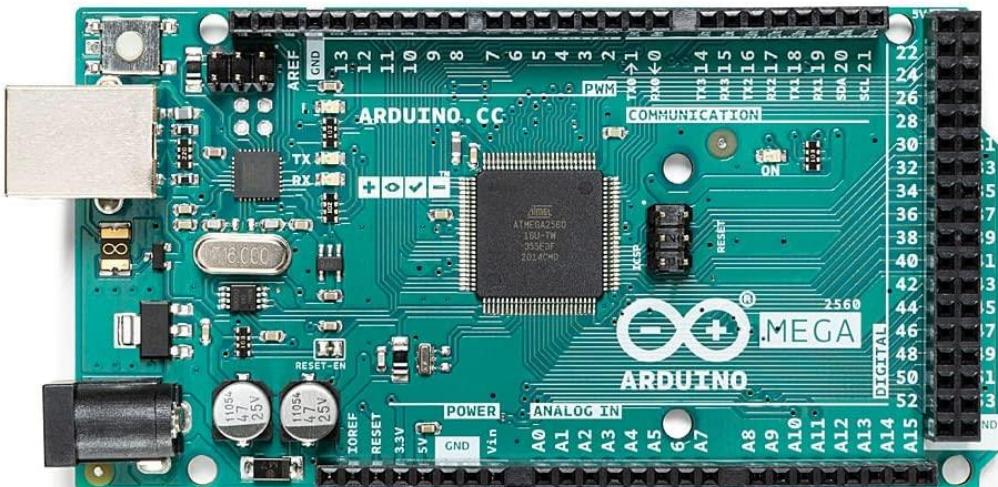


Figure 1-8 Arduino Mega

2. gas sensor MQ-5

- **Overview**
- Sensitive for LPG, natural gas, coal gas
- Output voltage boosts along with the concentration of the measured gases increases
- Fast response and recovery
- Adjustable sensitivity
- Signal output indicator



Figure 1- 9- Gas Sensor

- **Specifications**

- Power: 2.5V ~ 5.0V
- Dimension: 40.0mm * 21.0mm
- Mounting holes size: 2.0mm

- **Applications**

- Gas leakage detector

- **How to Use**

- In the case of working with a MCU:
- VCC \leftrightarrow 2.5V ~ 5.0V
- GND \leftrightarrow power supply ground
- AOUT \leftrightarrow MCU.IO (analog output)
- DOUT \leftrightarrow MCU.IO (digital output)

3. temperature sensor

- LM35 is a temperature measuring device having an analog output voltage proportional to the temperature.
- It provides output voltage in Centigrade (Celsius). It does not require any external calibration circuitry.
- The sensitivity of LM35 is 10 mV/degree Celsius. As temperature increases, output voltage also increases.

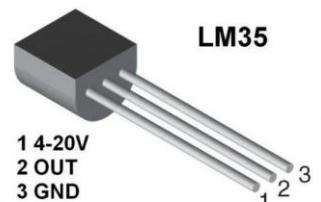


Figure 1-10 - Temperature Sensor

- It is a 3-terminal sensor used to measure surrounding temperature ranging from -55 °C to 150 °C.
- LM35 gives temperature output which is more precise than thermistor output.

- **Pin description**

- **VCC:** Supply Voltage (4V – 30V)

- **Out:** It gives analog output voltage which is proportional to the temperature (in degree Celsius).

- **GND:** Ground

As shown in Figure 4

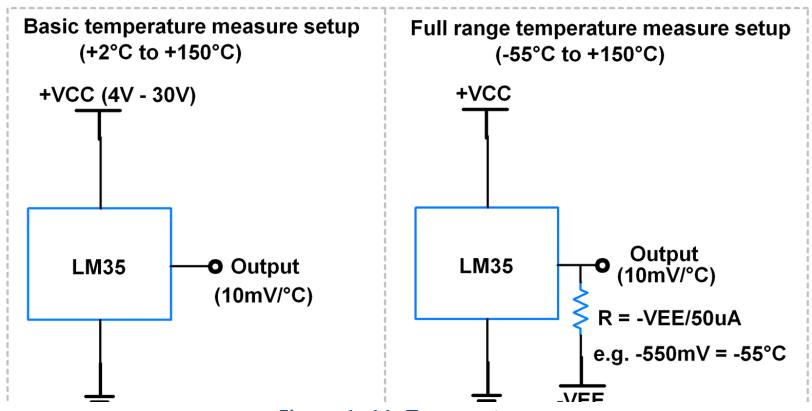


Figure 1- 11- Temperature measures

4. Ultra-Sonic Sensor

Ultrasonic Sensor HC-SR04 is a sensor that can measure **distance**. It emits an **ultrasound** at **40 000 Hz (40kHz)** which travels through the air and if there is an object or obstacle on its path It will bounce back to the module. Considering the travel time and the speed of the sound you can calculate the distance.

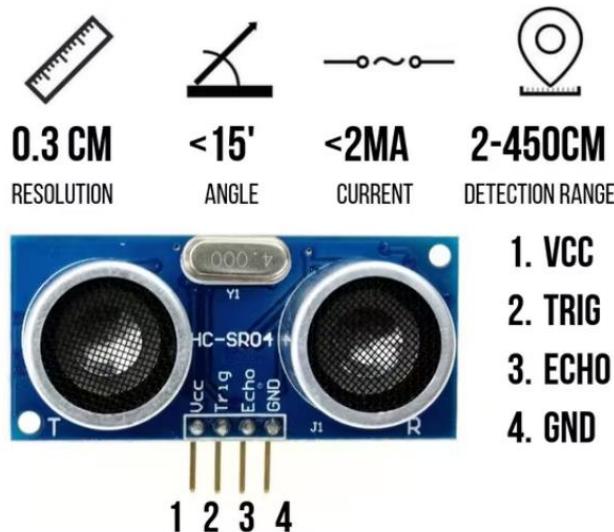


Figure 1-12 Ultrasonic Sensor

In order to generate the ultrasound we need to set the Trigger Pin on a High State for $10 \mu\text{s}$. That will send out an 8 cycle sonic burst which will travel at the speed of sound and it will be received in the Echo Pin. The Echo Pin will output the time in microseconds the sound wave traveled.

Ultrasonic HC-SR04 moduleTiming Diagram

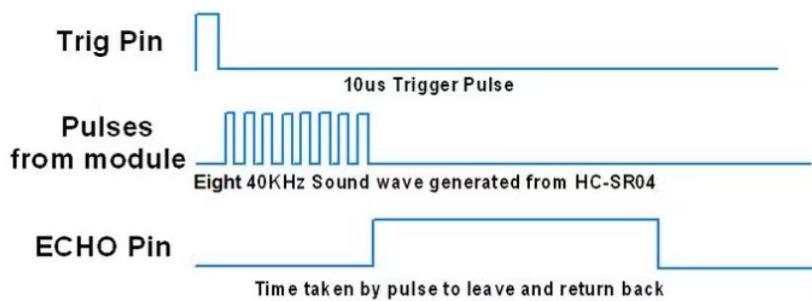


Figure 1-13 Ultrasonic Timing Diagram

5. Bluetooth Module (hc-05)

- It is used for many applications like wireless headset, game controllers, wireless mouse, wireless keyboard and many more consumer applications.
- It has range up to <100m which depends upon transmitter and receiver, atmosphere, geographic & urban conditions.
- It is IEEE 802.15.1 standardized protocol, through which one can build wireless Personal Area Network (PAN). It uses frequency-hopping spread spectrum (FHSS) radio technology to send data over air.
- It uses serial communication to communicate with devices. It communicates with microcontroller using serial port (USART).

• Pin Description

- Bluetooth serial modules allow all serial enabled devices to communicate with each other using Bluetooth.
- It has 6 pins,
 1. **Key/EN:** It is used to bring Bluetooth module in AT commands mode. If Key/EN pin is set to high, then this module will work in command mode. Otherwise by default it is in data mode. The default baud rate of HC-05 in command mode is 38400bps and 9600 in data mode.
- HC-05 module has two modes,
- 1. **Data mode:** Exchange of data between devices.
- 2. **Command mode:** It uses AT commands which are used to change setting of HC-05. To send these commands to module serial (USART) port is used.
- 2. **VCC:** Connect 5 V or 3.3 V to this Pin.
- 3. **GND:** Ground Pin of module.
- 4. **TXD:** Transmit Serial data (wirelessly received data by Bluetooth module transmitted out serially on TXD pin)
- 5. **RXD:** Receive data serially (received data will be transmitted wirelessly by Bluetooth module).
- 6. **State:** It tells whether module is connected or not.



Figure 1-14 Bluetooth

6. GPS Module

Ublox Neo-7N. Utilising high-tech Ublox hardware and exceptional software, this sensor can connect to multiple satellites in order to locate where you are currently at. I would really recommend using an external antenna with this module as it can increase its range and signal strength. Also, a wide open space is required for this project to work and the module to collect data as a direct line is required to establish a stable connection to satellites.



Figure 1-15 GPS Module

7. Motor driver

This L298N Motor Driver Module is a high-power motor driver module for driving DC and Stepper Motors. This module consists of an L298 motor driver IC and a 78M05 5V regulator. L298N Module can control up to 4 DC motors, or 2 DC motors with directional and speed control.

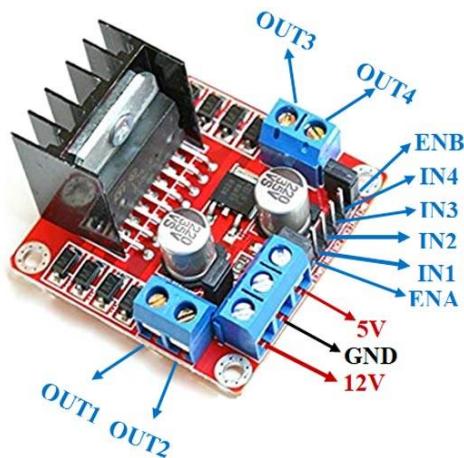


Figure 1-16 L298N Motor driver

The L298N Motor Driver module consists of an L298 Motor Driver IC, 78M05 Voltage Regulator, resistors, capacitor, Power LED, 5V jumper in an integrated circuit.

78M05 Voltage regulator will be enabled only when the jumper is placed.

When the power supply is less than or equal to 12V, then the internal circuitry will be powered by the voltage regulator and the 5V pin can be used as an output pin to power the microcontroller. The jumper should not be placed when the power supply is greater than 12V and separate 5V should be given through 5V terminal to power the internal circuitry.

ENA & ENB pins are speed control pins for Motor A and Motor B while IN1& IN2 and IN3 & IN4 are direction control pins for Motor A and Motor B.

8. Servo motor

Servos, are electronic devices and rotary or linear actuators that rotate and push parts of a machine with precision. Servos are mainly used on angular or linear position and for specific velocity, and acceleration.



Figure 1-17- Servo Motor

Servo Motor Working Mechanism

It consists of three parts:

1. Controlled device
2. Output sensor
3. Feedback system

It is a closed-loop system where it uses a positive feedback system to control motion and the final position of the shaft. Here the device is controlled by a feedback signal generated by comparing output signal and reference input signal.

Here reference input signal is compared to the reference output signal and the third signal is produced by the feedback system. And this third signal acts as an input signal to the control the device. This signal is present as long as the feedback signal is generated or there is a difference between the reference input signal and reference output signal. So the main task of servomechanism is to maintain the output of a system at the desired value at presence of noises.

9. Bread Board

It is self-adhesive and compatible with the Arduino Proto Shield.

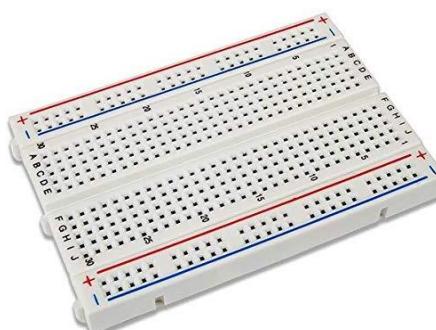


Figure 1-18 BreadBoard

- **Car**

10. 4 dc motors

11. 4 wheels

12. Jumper wires

13. Rechargeable batteries (appropriate capacity)

14. Battery holder



Figure 1-19 Car & accessories

15. Buzzer

An arduino buzzer is also called a piezo buzzer. It is basically a tiny speaker that you can connect directly to an Arduino. You can make it sound a tone at a frequency you set. The buzzer produces sound based on reverse of the piezoelectric effect.



Figure 1-20 Buzzer

2. 4 Arduino IDE programming language

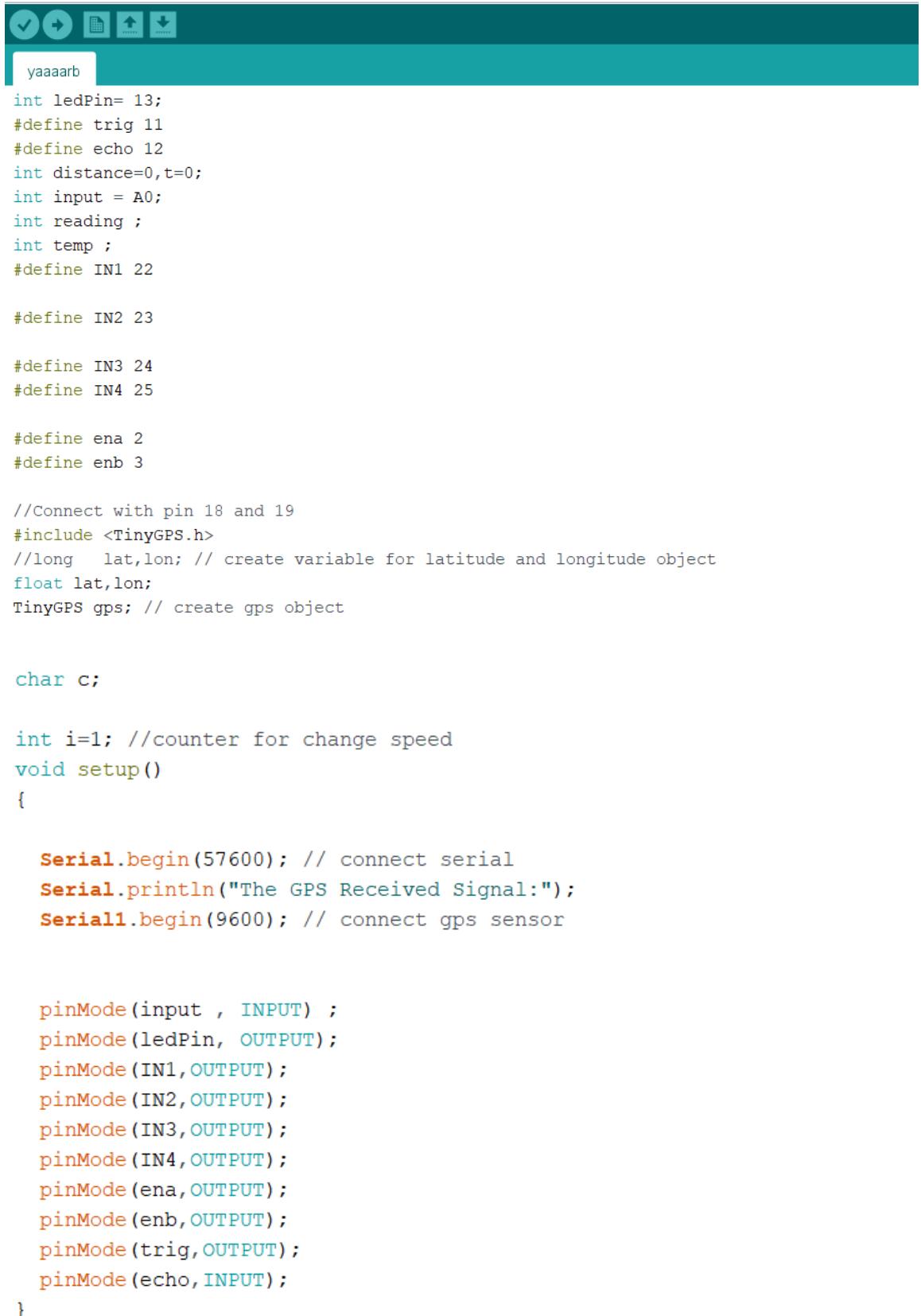
Arduino software is used to put the instruction of whole functions of this system to the microcontroller. Here we language. The program is burnt in the microcontroller using burner software. The program is stored in the EEPROM of the microcontroller, which is present in the NodeMCU ESP8266. By this software we put the data and instruction for forward, backward, left, right operation of this system. In android application when we press a button, a corresponding signal is sent through the Bluetooth to Bluetooth module (HC-05) which is connected with the NodeMCUESP8266 . Similarly an android application is been built for Wi-Fi module and when the buttons been pressed through the application the corresponding signal is been sent through the NodeMCU ESP8266 and the motor driver drives the wireless car. When signal data arrives the NodeMCU ESP8266 the pin which corresponds to the particular input is set to high. Now that pin gives the output to the motor driver section. Motor driver switches accordingly the data bit, if the data bit is low then the corresponding pin of the motor driver doesn't work else high bit then the corresponding pin of the motor driver is on. We have used Arduino IDE version 1.8.1 for writing program. There are two steps of the programming. First set up section where we define all the variables. Second loop part where the program runs continuously.

The screenshot shows the Arduino website's navigation bar at the top, featuring links for PROFESSIONAL, EDUCATION, STORE, a search bar, and SIGN IN. Below the navigation is a main content area with a large 'Downloads' heading. To the left is a card for 'Arduino IDE 1.8.19', which includes a download icon, the version number, a brief description, installation instructions, and a link to source code hosted on GitHub. To the right is a 'DOWNLOAD OPTIONS' sidebar listing download links for Windows (Win 7 and newer, ZIP file, and app), Linux (32-bit and 64-bit), ARM 32-bit, ARM 64-bit, and Mac OS X (10.10 or newer). It also includes links for Release Notes and Checksums.

Downloads

This is a detailed view of the Arduino IDE 1.8.19 download page. On the left, there's a large image of the Arduino logo, the title 'Arduino IDE 1.8.19', and a brief description: 'The open-source Arduino Software (IDE) makes it easy to write code and upload it to the board. This software can be used with any Arduino board.' Below this is a link to the 'Getting Started' page for installation instructions. A 'SOURCE CODE' section provides a link to the GitHub repository. On the right, a 'DOWNLOAD OPTIONS' sidebar lists download links for various operating systems: Windows (Win 7 and newer, ZIP file, and app), Linux (32 bits, 64 bits, ARM 32 bits, ARM 64 bits), and Mac OS X (10.10 or newer). It also includes links for 'Release Notes' and 'Checksums (sha512)'.

3. 5 Code implementation



The screenshot shows a code editor window with a dark theme. At the top, there are several icons: a checkmark, a circular arrow, a document, and two arrows pointing up and down. Below the icons, the file name 'yaaaarb' is displayed. The main area contains the following Arduino code:

```
int ledPin= 13;
#define trig 11
#define echo 12
int distance=0,t=0;
int input = A0;
int reading ;
int temp ;
#define IN1 22

#define IN2 23

#define IN3 24
#define IN4 25

#define ena 2
#define enb 3

//Connect with pin 18 and 19
#include <TinyGPS.h>
//long lat,lon; // create variable for latitude and longitude object
float lat,lon;
TinyGPS gps; // create gps object

char c;

int i=1; //counter for change speed
void setup()
{
    Serial.begin(57600); // connect serial
    Serial.println("The GPS Received Signal:");
    Serial1.begin(9600); // connect gps sensor

    pinMode(input , INPUT) ;
    pinMode(ledPin, OUTPUT);
    pinMode(IN1,OUTPUT);
    pinMode(IN2,OUTPUT);
    pinMode(IN3,OUTPUT);
    pinMode(IN4,OUTPUT);
    pinMode(ena,OUTPUT);
    pinMode(enb,OUTPUT);
    pinMode(trig,OUTPUT);
    pinMode(echo,INPUT);
}
```

```
void loop()
{
    char c = Serial.read();
    // bool ultra()=false ;
    /////////////////////////////////
    // ultraread();
    // tem();
    if(distance > 20 && c != 'o')
    {
        if( c == 'w')
        {
            forward();
            ultra();
        }
        else if ( c == 'd')
        {
            Tright();
            ultra();
        }
        else if ( c == 'a')
        {
            Tleft();
            ultra();
        }
        else if ( c == 's')
        {
            reverse();
            ultra();
        }
    }
}
```

```

        else if ( c == 'o')
        {
            off();
            ultra();
        }

    }
else
{
    off();
}
///////////////////////////////
if(c =='t')
{
    tem();

}
else if(c =='l'){
ultraread();
}
else if(c == 'g'){
gps1();
}
}

void forward()
{
    digitalWrite(IN1,LOW);
    digitalWrite(IN2,HIGH);
    digitalWrite(IN3,HIGH);
    digitalWrite(IN4,LOW);
    analogWrite(ena,100);
    analogWrite(enb,100);
}

void reverse()
{
    digitalWrite(IN1,HIGH);
    digitalWrite(IN2,LOW);
    digitalWrite(IN3,LOW);
    digitalWrite(IN4,HIGH);
    analogWrite(ena,100);
    analogWrite(enb,100);
}

void Tleft()
{
    digitalWrite(IN1, HIGH);
    digitalWrite(IN2, LOW);
    digitalWrite(IN3, HIGH);
    digitalWrite(IN4, LOW);
    analogWrite(ena,100);
    analogWrite(enb,100);
}

```

```
void Tright()
{
    digitalWrite(IN1,LOW);
    digitalWrite(IN2,HIGH);
    digitalWrite(IN3,LOW);
    digitalWrite(IN4,HIGH);
    analogWrite(ena,100);
    analogWrite(enb,100);
}

void off()
{
    digitalWrite(IN1,LOW);
    digitalWrite(IN2,LOW);
    digitalWrite(IN3,LOW);
    digitalWrite(IN4,LOW);
    analogWrite(ena,100);
    analogWrite(enb,100);
}

void ultra()
{
    digitalWrite(trig,LOW);
    delayMicroseconds(5);
    digitalWrite(trig,HIGH);
    delayMicroseconds(10);
    digitalWrite(trig,LOW);
    t=pulseIn(echo,HIGH);
    distance=t/57;//Distance = (Speed of Sound * Time/2) = t/(1/((350*0.0001)/2))

}

void ultraread()
{

    digitalWrite(trig,LOW);
    delayMicroseconds(5);
    digitalWrite(trig,HIGH);
    delayMicroseconds(10);
    digitalWrite(trig,LOW);
    t=pulseIn(echo,HIGH);
    distance=t/57;//Distance = (Speed of Sound * Time/2) = t/(1/((350*0.0001)/2))

    serial.println(distance);
}
```

```
void tem()
{
    reading = analogRead(input);
    temp = (reading * (5.0/1024))*100 - 2.5 ;
    Serial.println(temp);
}

void gps1()
{
    while(Serial1.available()) { // check for gps data
        if(gps.encode(Serial1.read()))// encode gps data
        {
            gps.f_get_position(&lat,&lon); // get latitude and longitude

            Serial.print("Position: ");

            //Latitude
            Serial.print("Latitude: ");
            Serial.print(lat,6);

            Serial.print(",");
           

            //Longitude
            Serial.print("Longitude: ");
            Serial.println(lon,6);
        }
    }
}
```

3. 6 Components

- Arduino Mega
- gas sensor MQ-5
- temperature sensor
- Ultra-Sonic Sensor
- Bluetooth Module (hc-05)
- GPS Module
- Rechargeable batteries (appropriate capacity)
- 4 dc motors
- 4 wheels
- Jumper wires
- Motor driver
- Servo motor
- Bread Board
- Buzzer
- Battery holder

2. 7 Conclusion

To us the need of internet and the things which are internet based are very much important nowadays. IoT or internet of things is the very important part in both computer and our daily lives. The above model describes how the Arduino programs the car motor module and by IoT we actually rotate the wheels and give direction to the car. IoT gives us the opportunity to work with different platforms and it helps us to create various interesting modules to work on. We also tested the applications used to drive the car. Due to the new concept of Wireless Controlled Car using Bluetooth and IoT, we were able to come up with various possibilities that can take place.

CHAPTER 3

Windows form programming with C#

CHAPTER 3

Windows form programming with C#.

3. 1 Introduction

Windows Forms is a Graphical User Interface (GUI) class library that is bundled in .Net Framework. Its main purpose is to provide an easier interface to develop applications for desktops, tablets, and PCs. It is also termed the WinForms. The applications which are developed by using Windows Forms or WinForms are known as the Windows Forms Applications that run on the desktop computer. WinForms can be used only to develop the Windows Forms Applications not web applications. WinForms applications can contain different type of controls like labels, list boxes, tooltips, etc.

A graphical user interface (GUI) application is one that is designed using graphical features that make the application easy to use. When you compile a GUI application, an executable file with start-up code is created. The executable usually provides the basic functionality of the program, and simple programs often consist of only an executable file.

GUI was created because text command-line interfaces were complicated and difficult to learn. The GUI process lets you click or point to a small picture, known as an icon or widget, and open a command or function on your devices, such as tabs, buttons, scroll bars, menus, icons, pointers, and windows. It is now the standard for user-centered design in software application programming.

Programs that use GUI are known as “GUI programs.” The program creates small pictures of tasks or functions and waits for the user to interact with them. The user controls when and how they will be used. To select functions, users can use a keyboard or pointing device, such as a mouse, touchpad, or touchscreen depending on the device.

Creating a Windows Forms Application Using Visual Studio 2022

Create a project

First, you'll create a C# application project. The project type comes with all the template files you'll need, before you've even added anything.

1-Open Visual Studio.

2-On the start window, select Create a new project.

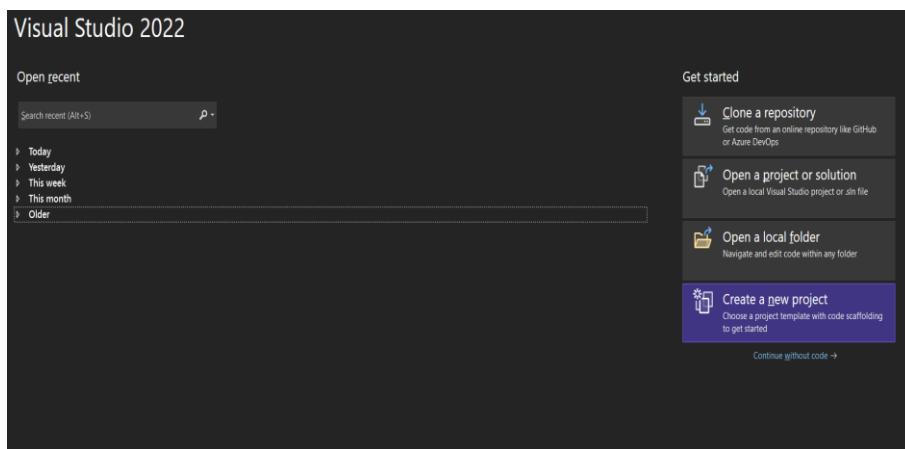


Figure 2-1 Visual studio 2022

3-On the Create a new project window, select the Windows Forms App (.NET Framework) template for C#.

(If you prefer, you can refine your search to quickly get to the template you want. For example, enter or type Windows Forms App in the search box. Next, select C# from the Language list, and then select Windows from the Platform list.)

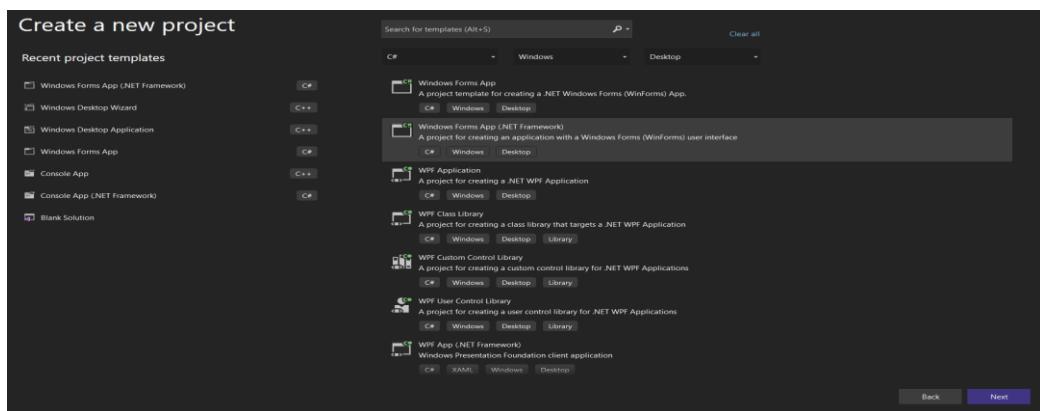
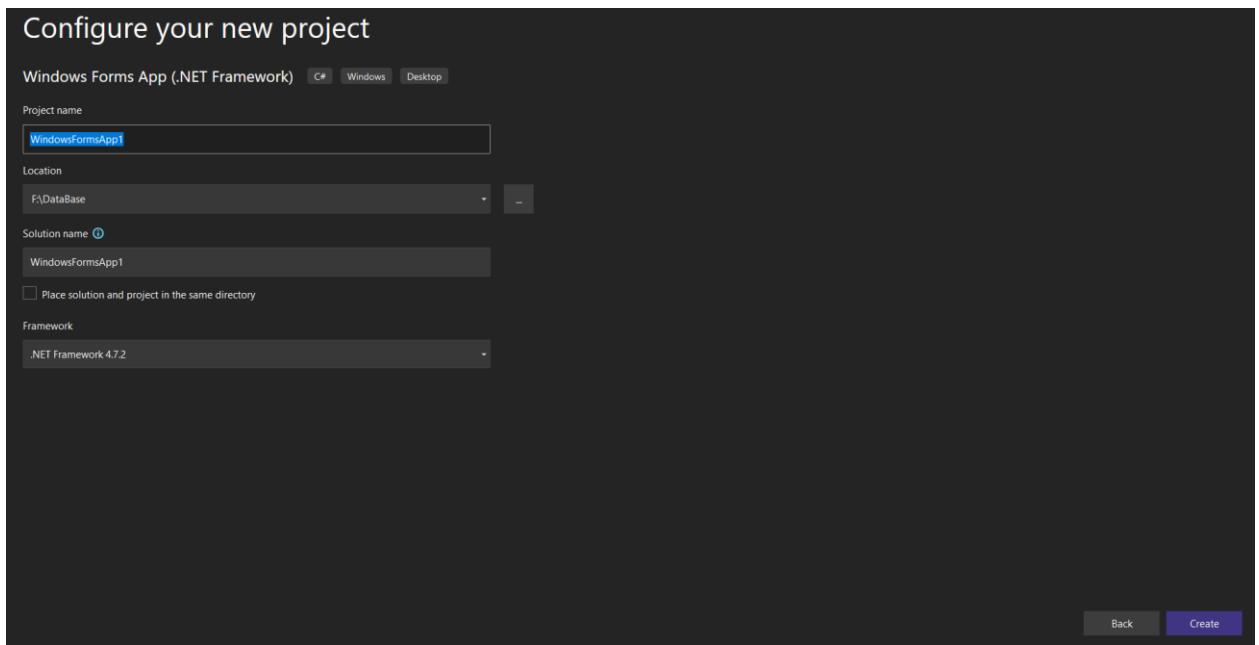
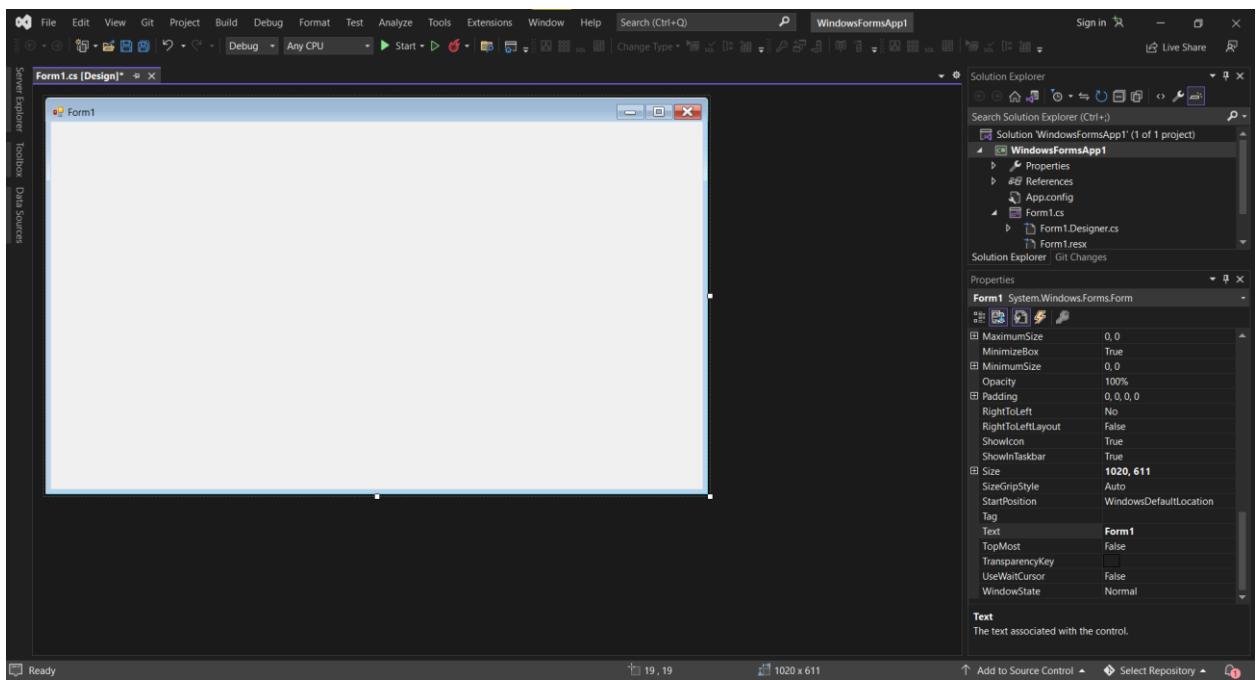


Figure 2-2 Create new project

4-In the Configure your new project window, type or enter HelloWorld in the Project name box. Then, select Create.



2-3 New project window



2-4 Start windows form

3.2 Properties of GUI:

Input controls:

Checkboxes: Checkboxes are square boxes in a list of one or more options. When you click the box, it stays selected. They are best presented in a vertical list. A checkbox can be a single box, such as acknowledging a statement, or a list of related items, such as a shopping list.

Date picker: A date picker lets you select a date and/or time. The creator can choose a calendar or a fill-in option. It ensures that a consistent format is used, such as “day, month, year.”

Dropdown lists: Dropdown lists let you select one item at a time. Several items can be included compactly. Consider adding directions, such as “select one” to let the user know what to do. The creator can add or delete items to keep the list up to date.

List boxes: List boxes let you select multiple items from one compact list. Use this GUI feature if you have a long list of options for the user to consider. There are four variations of list boxes: single-line, multi-select, multi-select with checkboxes, and multi-select-dual list boxes.

Toggles: Toggle buttons let you change a setting typically as off/on states.

Informational components

Message box: A message box is a small window with information such as a policy or disclaimer. It requires you to take action before you proceed.

Notifications: A notification is a message box. Typically, they are used to indicate emergency warnings, error messages, or task completion.

Pop-up windows: A pop-up, or modal, window requires you to interact with it before you can return to the system.

Progress bar: A progress bar shows where you are in a series of steps in a process. Typically, progress bars are not clickable. For example, a progress bar might show your pizza order's status in the order, cook, and delivery process.

Tooltips: A tool tip offers you more information when you hover over an item. For example, you might receive a definition and usage examples when you hover over a word or phrase.

Interaction elements of a GUI:

Apart from structural elements, a GUI also features interaction elements, such as:

Cursors: A cursor indicates the place where the system will accept input next. It can either be a pointer, which follows the movements of a pointing device—such as a mouse—or a text cursor, which indicates the point of focus in a current text box.

Selections: A selection refers to a list of items to which a user will apply an operation. A user will select a portion of text for cut, copy and paste operations. Image editing applications allow users to select and modify certain areas of an image by using the magic wand selection or lasso selection tools.

Adjustment handles: A handle serves as the indicator of a drag and drops operation. When a user places the pointer on the handle to initiate the drag process, its shape changes to an icon that represents the drag function.

What are the benefits of a GUI?

GUI uses visual elements to represent those now hidden lines of command. You simply select a button or an icon to call the relevant function. The easy use of GUIs has made it possible for the public in general, regardless of experience or knowledge, to access all kinds of systems for everyday use.

There are many other benefits to using GUI. Here are the most common:

1. Easy to use

Since data is represented by symbols, shapes, and icons, users can easily recognize, classify and navigate options. A simple click is all it takes to acquire a function. Because it's so easy to use and understand, GUI has become the preferred interface for computers and mobile devices.

2. Easy to communicate

Visual representation of data is recognized faster than text. Non-programmers find it easy to use GUIs since it requires no experience with computing commands. They don't have to worry about writing or debugging code. As a result, users find GUI an easy-to-learn interface.

3. Attractive

GUI has visually appealing features and is not cluttered with command line codes. Visual images can portray emotions, comments, and situations with long lines of computer language. Pictures and such are easy to understand and often carry universal meaning.

4. Provides shortcuts

GUI lets users take advantage of shortcut keys to minimize strokes. A combination of two keys in place of several actions saves the user time and increases productivity. For example, a call-to-action button can pull up a form, a prefilled letter, or a list of contact information. That one button saves you from searching for the same information.

5. Allows for multitasking

GUI lets users work and view two or more programs at the same time. For example, you can view a streaming presentation while searching the internet from a web browser. You can watch a video while writing a review of the presentation with a search engine in another tab.

3. 3 Basic Windows Forms

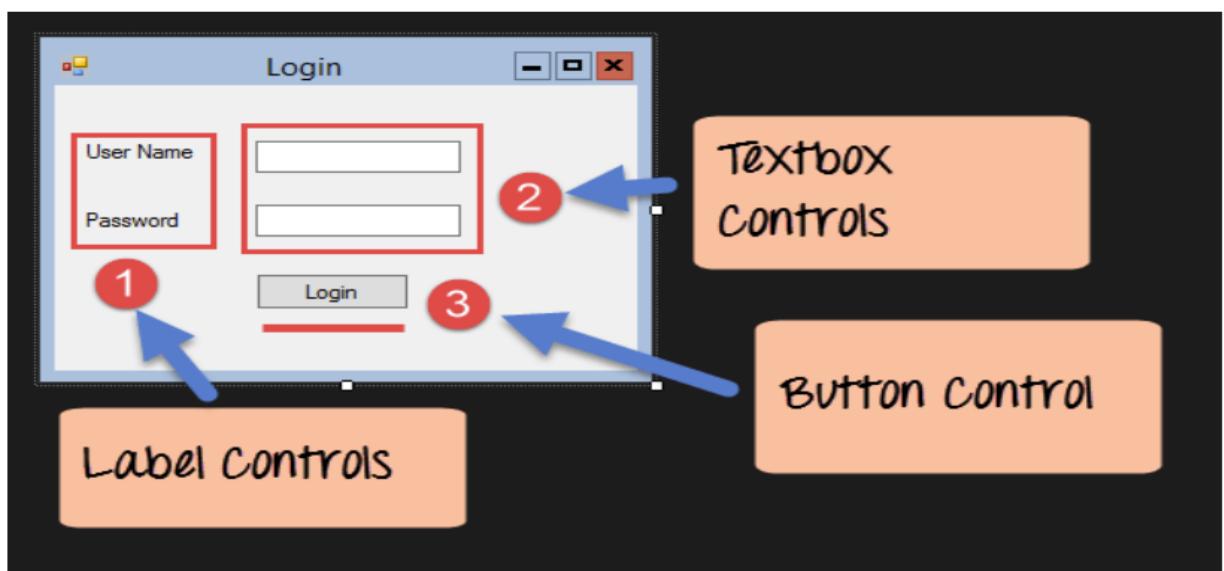
- **Basic controls**

A Windows forms application is one that runs on the desktop computer. A Windows forms application will normally have a collection of controls such as labels, textboxes, **comboBox**, etc.

Buttons are circles that let you make immediate choices and take action. Radio buttons come in groups where only one button can be selected at a time. Label buttons have text on them. Consider using a check box if you want more than one option to be selected.

Textboxes: Text boxes are fields that let you enter text. The creator can control how much text is allowed.

Label: A Label control is used as a display medium for text on Forms. Label control does not participate in user input or capture mouse or keyboard events.



2-5 Basic controls

ComboBox: C# controls are located in the Toolbox of the development environment. You can use them to create objects on a form with a simple series of mouse clicks and dragging motions. A ComboBox displays a Textbox combined with Listbox, which enables the user to select items from the list or enter a new value.

- **CLICK EVENTS**

Of course, a menu is not very useful if you can't make it do something. In this section we'll define some event handlers for our menus and examine how event handlers work in more detail than we covered in chapter 2. This section builds on the MyPhotos version 3.2 project constructed in section 3.2, or available on the book's web site. Events for Windows Forms controls can be added from the Windows Forms Designer window, or in the Properties window. We will discuss each method separately.

How do I detect keys pressed in C#

You can detect most physical key presses by handling the KeyDown or KeyUp events. Key events occur in the following order:

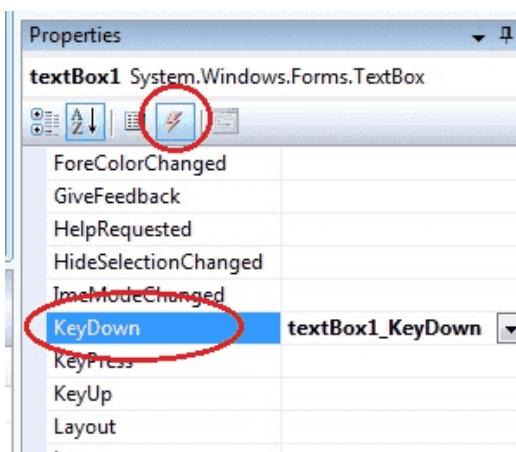
- KeyDown
- KeyPress
- KeyUp

How to get TextBox1_KeyDown event in your C# source file ?

Select your TextBox control on your Form and go to Properties window. Select Event icon on the properties window and scroll down and find the KeyDown event from the list and double-click the Keydown Event. You will get the KeyDown event in your source code editor.

Code:

```
private void textBox1_KeyDown(...) { }
```



2-6 Choose event

Difference between the KeyDown Event, KeyPress Event, and KeyUp Event

KeyDown Event: This event is raised as soon as the user presses a key on the keyboard, it repeats while the user keeps the key depressed.

KeyPress Event: This event is raised for character keys while the key is pressed and then released. This event is not raised by noncharacter keys, unlike KeyDown and KeyUp, which are also raised for noncharacter keys

KeyUp Event: This event is raised after the user releases a key on the keyboard.

KeyDown Event Code :

```

1  using System;
2  [using System.Windows.Forms;
3
4  namespace WindowsFormsApp1
5  {
6      public partial class Form1 : Form
7      {
8          public Form1()
9          {
10             InitializeComponent();
11         }
12         private void Form1_KeyDown(object sender, KeyEventArgs e)
13         {
14             if (e.KeyCode == Keys.Enter)
15             {
16                 MessageBox.Show("Enter key pressed");
17             }
18         }
19     }
20 }
```

KeyUp Event Code:

```

1  using System;
2  [using System.Windows.Forms;
3
4  namespace WindowsFormsApp1
5  {
6      public partial class Form1 : Form
7      {
8          public Form1()
9          {
10             InitializeComponent();
11         }
12         private void Form1_KeyUp(object sender, KeyEventArgs e)
13         {
14             if (e.KeyCode == Keys.Enter)
15             {
16                 MessageBox.Show("Enter key pressed");
17             }
18         }
19     }
20 }
```

KeyPress Event Code:

```

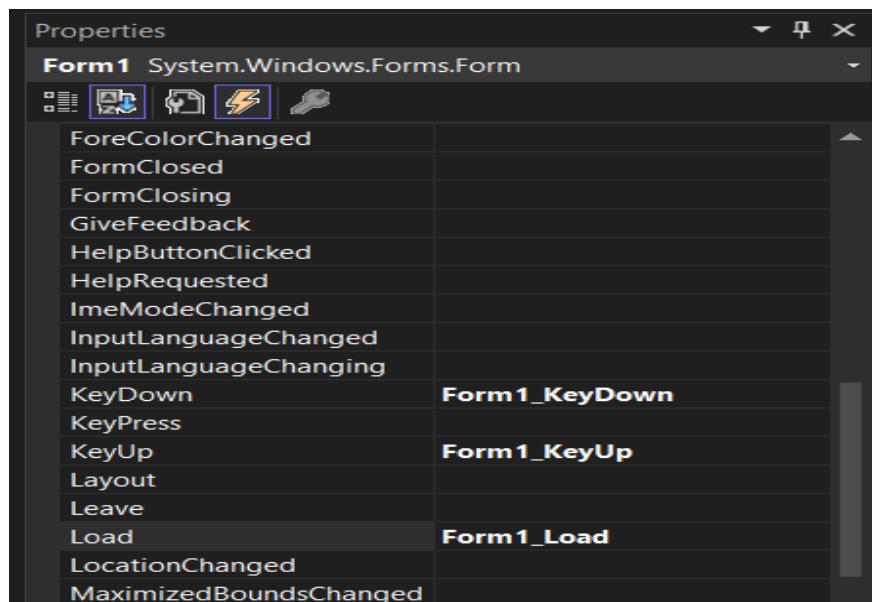
1  using System;
2  using System.Windows.Forms;
3
4  namespace WindowsFormsApp1
5  {
6      public partial class Form1 : Form
7      {
8          public Form1()
9          {
10             InitializeComponent();
11         }
12
13         private void Form1_KeyPress(object sender, KeyPressEventArgs e)
14         {
15             if (e.KeyChar == (char)Keys.Enter)
16             {
17                 MessageBox.Show("Enter key pressed");
18             }
19             if (e.KeyChar == 13)
20             {
21                 MessageBox.Show("Enter key pressed");
22             }
23         }
24     }
25 }
```

Form.Load Event

You can use this event to perform tasks such as allocating resources used by the form.

An important event you'll want to write code for is the **Form Load** event. You might want to, for example, set the Enabled property of a control to False when a form loads. Or maybe blank out an item on your menu. You can do all this from the Form Load event.

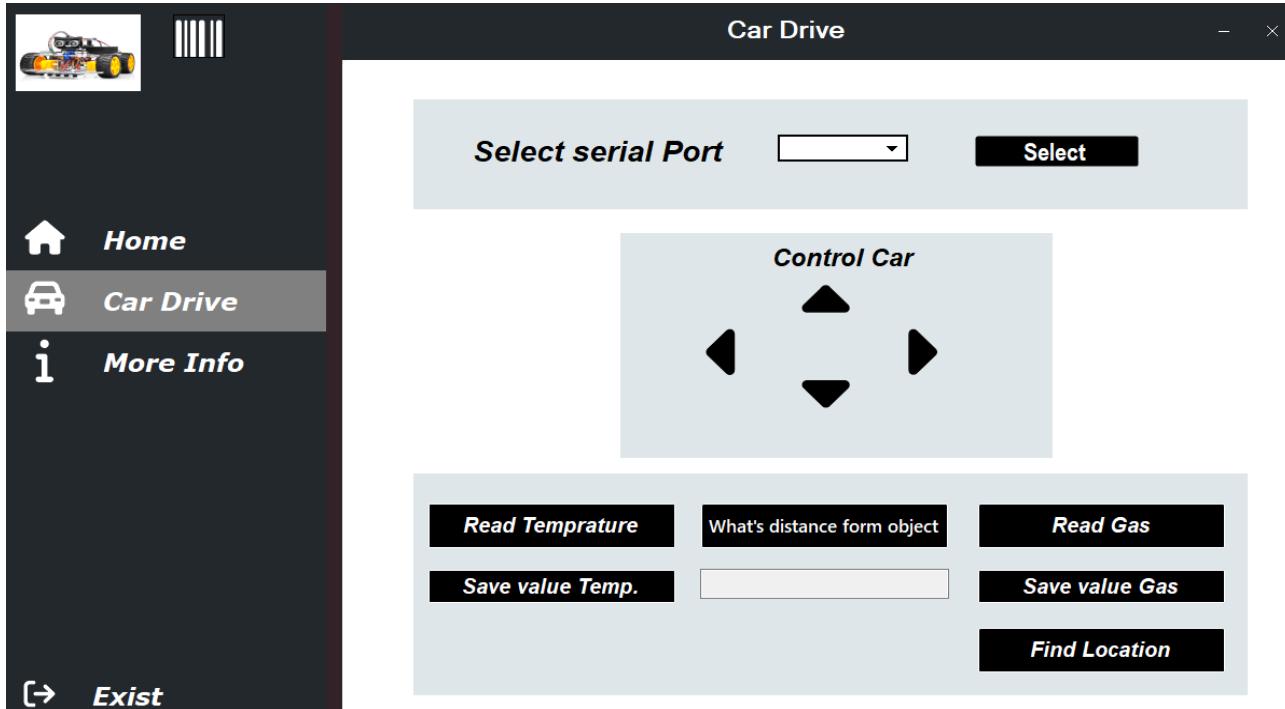
Bring up your coding window, and select the Form1 Events from the drop-down box:



3. 4 Problem Formulation

The user can't deal with the code to reach the information he wants to know or to use the car properly so the application helps him to:

- Control the moving of the car
- Know the read (value) of each sensor
- Easy to deal with a website that contains all the information about the car

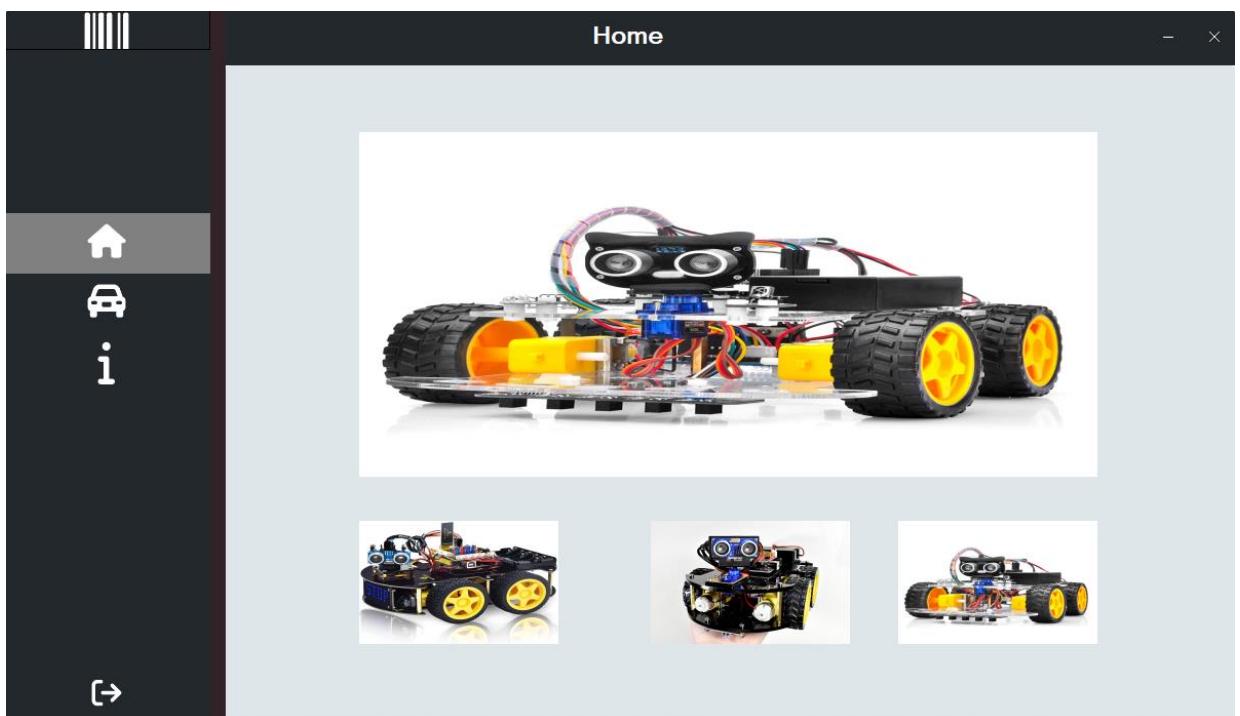
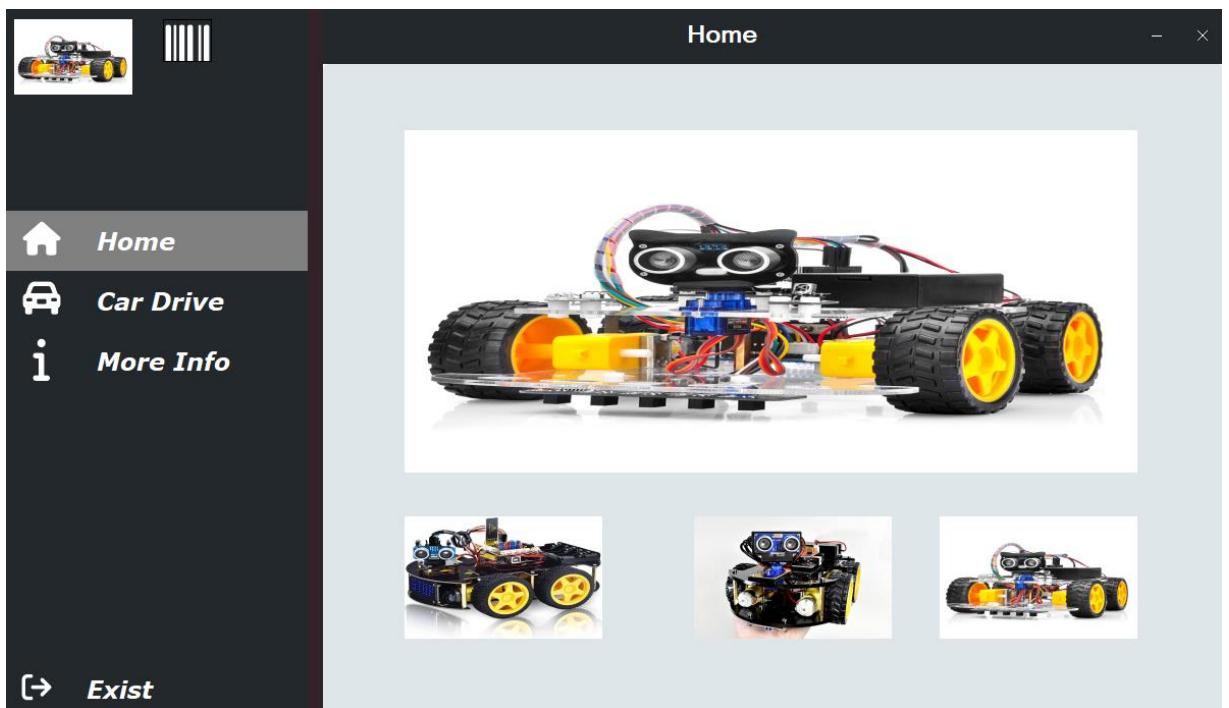


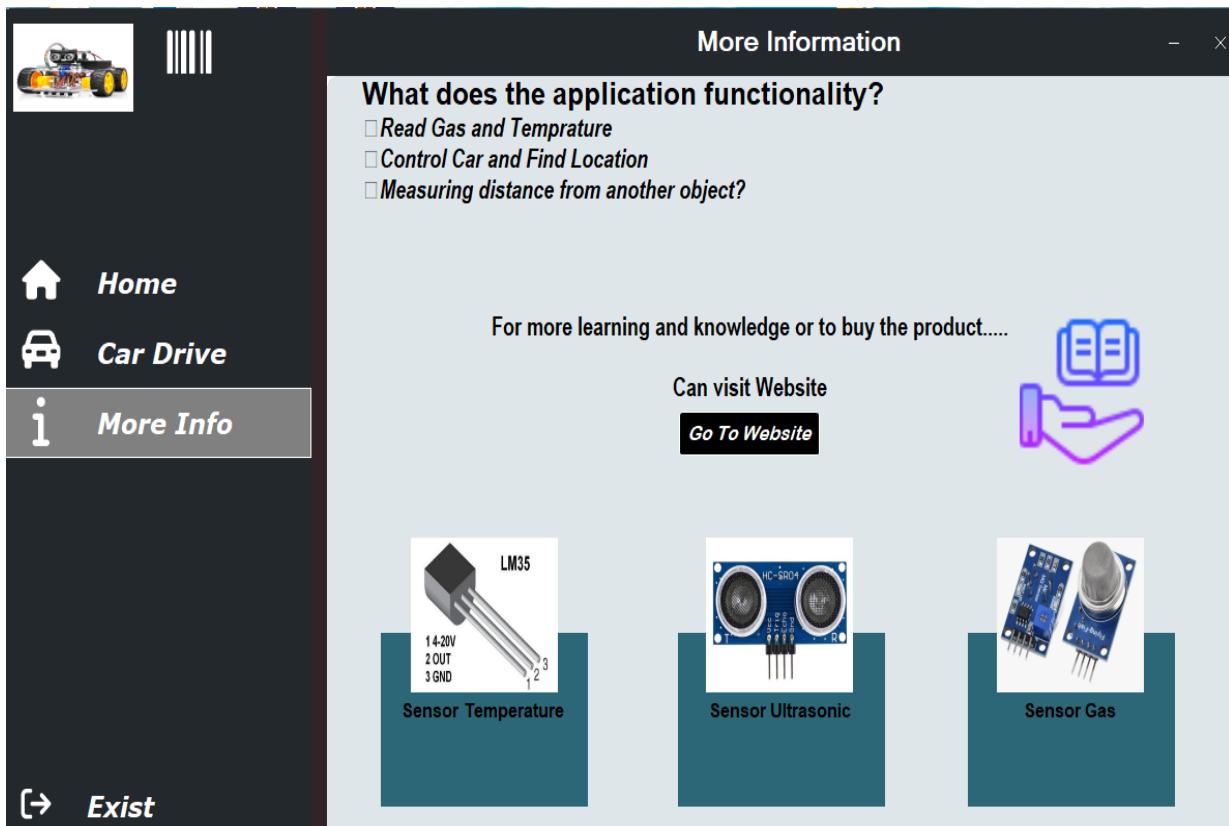
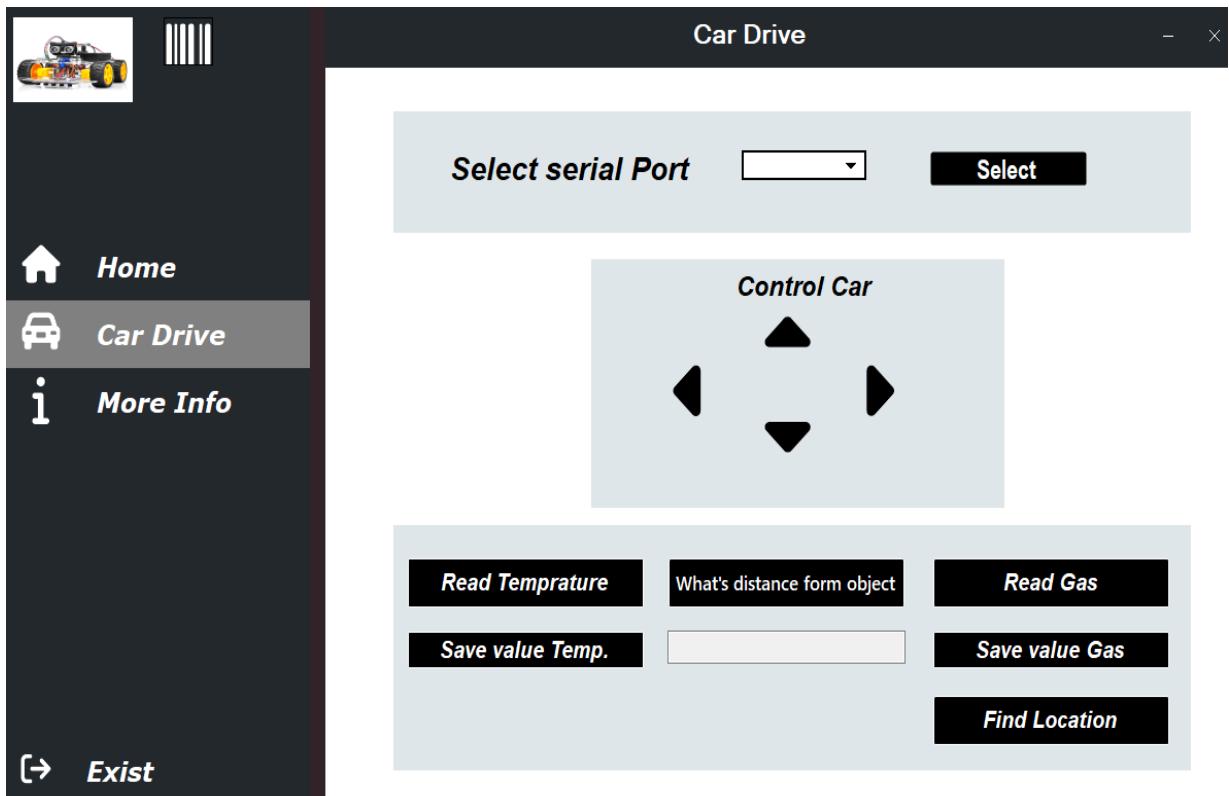
2-7 Design of page car driver

3. 5 The Applied Approach

Design a desktop application to facilitate the use of the car which is simple to the user to deal with, as there is a button to drive and know the readings of each sensor and another one to open the web page which contains all the information about the car using c# language and visual studio

3. 5 Design and Code of Capstone Project





```
1  using System;
2  using System.Drawing;
3  using System.Linq;
4  using System.Windows.Forms;
5  using System.IO.Ports; // connected to serial port
6  using System.Data.SqlClient; // Sql DataBase
7
8  namespace Design2
9  {
10     public partial class Form1 : Form
11     {
12         public Form1()
13         {
14             InitializeComponent();
15             CollapseMenu();
16         }
17         ////////////// Start ///////////
18         // Select Serial Port
19         private void Form1_Load(object sender, EventArgs e)
20         {
21             DriveBtn.BackColor = Color.Gray;
22
23             string[] ports = SerialPort.GetPortNames(); // (Read Serial Port)
24             foreach (string port in ports)
25             {
26                 gunaComboBox1.Items.Add(port); // (Add List)
27             }
28         }
29
30         private void Select_Click(object sender, EventArgs e)
31         {
32             string x = gunaComboBox1.SelectedItem.ToString();
33             serialPort1.PortName = x;
34             serialPort1.BaudRate = 9600;
35             serialPort1.Open();
36         }
37         ////////////// End ///////////
```

```
40 ////////////// Start //////////
41 // Car move
42     1 reference
43     private void Form1_KeyDown(object sender, KeyEventArgs e)
44     {
45         if (e.KeyCode == Keys.W)
46         {
47             up.PerformClick();
48         }
49         else if (e.KeyCode == Keys.S)
50         {
51             down.PerformClick();
52         }
53         else if (e.KeyCode == Keys.D)
54         {
55             right.PerformClick();
56         }
57         else if (e.KeyCode == Keys.A)
58         {
59             left.PerformClick();
60         }
61     }
62     1 reference
63     private void Form1_KeyUp(object sender, KeyEventArgs e)
64     {
65         serialPort1.Write("o");
66     }
67     1 reference
68     private void up_Click(object sender, EventArgs e)
69     {
70         serialPort1.Write("w");
71     }
72     1 reference
73     private void right_Click(object sender, EventArgs e)
74     {
75         serialPort1.Write("d");
76     }
```

```
77     private void left_Click(object sender, EventArgs e)
78     {
79         serialPort1.Write("a");
80     }
81
82     1 reference
83     private void down_Click(object sender, EventArgs e)
84     {
85         serialPort1.Write("s");
86     }
87
88     ////////////// End ///////////
89
90     ////////////// Start ///////////
91     // Data received
92     string RxString;
93     1 reference
94     private void serialPort1_DataReceived_1(object sender, SerialDataReceivedEventArgs e)
95     {
96
97         RxString = serialPort1.ReadExisting();
98         this.Invoke(new EventHandler(DisplayText));
99     }
100
101    1 reference
102    private void DisplayText(object sender, EventArgs e)
103    {
104        TextDRe.AppendText(RxString);
105    }
106
107    1 reference
108    private void ReadTem_Click_1(object sender, EventArgs e)
109    {
110
111        serialPort1.Write("t");
112        TextDRe.Clear();
113    }
114
115    1 reference
116    private void DistBtn_Click_1(object sender, EventArgs e)
117    {
118        serialPort1.Write("l");
119        TextDRe.Clear();
120    }
121
122    ////////////// End ///////////
```

```
123     private void BtnClose_Click(object sender, EventArgs e)
124     {
125         Application.Exit();
126     }
127
128     //// Start ////
129     /// Menu
130     private void btnmenu_Click(object sender, EventArgs e)
131     {
132         CollapseMenu();
133     }
134
135     2 references
136     private void CollapseMenu()
137     {
138         if (this.panelMenu.Width >= 300) //Collapse menu
139         {
140             panelMenu.Width = 200;
141             pictureBox2.Visible = false;
142             btnmenu.Dock = DockStyle.Top;
143             foreach (Button menuButton in panelMenu.Controls.OfType<Button>())
144             {
145                 menuButton.Text = "";
146                 menuButton.ImageAlign = ContentAlignment.MiddleCenter;
147                 menuButton.Padding = new Padding(0);
148             }
149         }
150         else
151         { //Expand menu
152             panelMenu.Width = 300;
153             pictureBox2.Visible = true;
154             btnmenu.Dock = DockStyle.None;
155             foreach (Button menuButton in panelMenu.Controls.OfType<Button>())
156             {
157                 menuButton.Text = " " + menuButton.Tag.ToString();
158                 menuButton.ImageAlign = ContentAlignment.MiddleLeft;
159                 menuButton.Padding = new Padding(10, 0, 0, 0);
160             }
161         }
162     }
163     //// End ////
164 
```

```
164 // Start //
165 // convert between page //
166 private void DriveBtn_Click(object sender, EventArgs e)
167 {
168
169     lbltxt.Text = "Home";
170     moreInfo1.Visible = false;
171     home1.Visible = true;
172     if(DriveBtn.Focused == true)
173     {
174         DriveBtn.BackColor = Color.Gray;
175         HomeBtn.BackColor = panelMenu.BackColor;
176         MInfoBtn.BackColor = panelMenu.BackColor;
177     }
178
179 }
180
181 private void HomeBtn_Click_1(object sender, EventArgs e)
182 {
183     lbltxt.Text = "Car Drive";
184     home1.Visible = false;
185     moreInfo1.Visible = false;
186     if (HomeBtn.Focused == true)
187     {
188         HomeBtn.BackColor = Color.Gray;
189         DriveBtn.BackColor = panelMenu.BackColor;
190         MInfoBtn.BackColor = panelMenu.BackColor;
191     }
192
193 private void MInfoBtn_Click_1(object sender, EventArgs e)
194 {
195     lbltxt.Text = "More Information";
196     home1.Visible = false;
197     moreInfo1.Visible = true;
198     MInfoBtn.BackColor = Color.Gray;
199     DriveBtn.BackColor = panelMenu.BackColor;
200     HomeBtn.BackColor = panelMenu.BackColor;
201 }
202 // End //
```

```
205 // Start //
206 // Connected to DataBase
207 1 reference
208 private void SaveTem_Click(object sender, EventArgs e)
209 {
210     SqlConnection con = new SqlConnection(@"Data Source=DESKTOP-E4LQDM3;Initial Catalog=Capstonproject;Integrated Security=True");
211     string InsertQuay = "Insert into Tem(Temperature)Values('" + TextDRe.Text + "')";
212     con.Open();
213     SqlCommand cmd = new SqlCommand(InsertQuay, con);
214     cmd.ExecuteNonQuery();
215     con.Close();
216 }
217 1 reference
218 private void SaveGas_Click(object sender, EventArgs e)
219 {
220 }
221 // End //
```

CHAPTER 4

Website Implementation

CHAPTER 4

Website Implementation

1.Introduction:

At the first, the website presents many features the first one is to display and introduce team members at the home page, and simple brief talking about the project.

The second webpage displaying the specifications and the components of the project that our team members used in designing the project and each component has it's datasheet that you can download and explore, it's not only that, there are other features, you can learn about the programs and technologies used for the project (Arduino IDE and C# for the GUI Application);

Also you can see the code used in programming and developing the project, it's not only that, there are some schematics for designed parts.

There are many webpages used in this website like if the users wants to buy the project we developed webpage to sell it with different versions and we offered more than one way to make it easy for user to buy the project like cash on delivery or by credit card, all the user have to do is enter some data like name, email, phone, and address after that the user choose the way that he would pay with, there are webpage if the user wants to leave comments for team members, it will be stored in the database and being displayed in the admin part, which in it admins or team members see information about who logged in, who left comments, and who bought the project.

At the end of the website there is a webpage that user can log out from our website.

2. Languages and technologies:

The website is divided into two parts. The first part (Front-end) consists of languages and techniques for displaying content, while the second part (back-end) consists of languages and techniques used to make the website dynamic.

Front-end languages and technologies:

Html:

- HTML, or **HyperText Markup Language**, allows web users to create and structure sections, paragraphs, and links using elements, tags, and attributes. However, it's worth noting that HTML is not considered a programming language as it can't create dynamic functionality.
- HTML has a lot of use cases, namely:
 - **Web development:** Developers use HTML code to design how a browser displays web page elements, such as text, hyperlinks, and media files.
 - **Internet navigation:** Users can easily navigate and insert links between related pages and websites as HTML is heavily used to embed hyperlinks.
 - **Web documentation:** HTML makes it possible to organize and format documents, similarly to Microsoft Word.
- It's also worth noting that HTML is now considered an official web standard. The World Wide Web Consortium (W3C) maintains and develops HTML specifications, along with providing regular updates.
- This article will go over the basics of HTML, including how it works, its pros and cons, and how it relates to CSS and JavaScript.

CSS:

- Cascading Style Sheets, fondly referred to as CSS, is a simple design language intended to simplify the process of making web pages presentable.
- CSS handles the look and feel part of a web page. Using CSS, you can control the color of the text, the style of fonts, the spacing between paragraphs, how columns are sized and laid out, what background images or colors are used, layout designs, variations in display for different devices and screen sizes as well as a variety of other effects.
- CSS is easy to learn and understand but it provides powerful control over the presentation of an HTML document. Most commonly, CSS is combined with the markup languages HTML or XHTML.

What does CSS do?

- You can add new looks to your old HTML documents.
- You can completely change the look of your website with only a few changes in CSS code.

JavaScript:

- JavaScript is a scripting language that enables you to create dynamically updating content, control multimedia, animate images, and pretty much everything else. (Okay, not everything, but it is amazing what you can achieve with a few lines of JavaScript code.)
- It's a scripting or programming language that allows you to implement complex features on web pages — every time a web page does more than just sit there and display static information for you to look at — displaying timely content

updates, interactive maps, animated 2D/3D graphics, scrolling video jukeboxes, etc. — you can bet that JavaScript is probably involved. It is the third layer of the layer cake of standard web technologies, two of which (HTML and CSS) we have covered in much more detail in other parts of the Learning Area.

AJAX:

AJAX stands for Asynchronous JavaScript And XML. In a nutshell, it is the use of the XMLHttpRequest object to communicate with servers. It can send and receive information in various formats, including JSON, XML, HTML, and text files. AJAX's most appealing characteristic is its "asynchronous" nature, which means it can communicate with the server, exchange data, and update the page without having to refresh the page.

The two major features of AJAX allow you to do the following:

- Make requests to the server without reloading the page
- Receive and work with data from the server

JSON:

- JSON (JavaScript Object Notation) is a lightweight data-interchange format. It is easy for humans to read and write. It is easy for machines to parse and generate. It is based on a subset of the JavaScript Programming Language Standard ECMA-262 3rd Edition - December 1999. JSON is a text format that is completely language independent but uses conventions that are familiar to programmers of the C-family of languages, including C, C++, C#, Java, JavaScript, Perl,

Python, and many others. These properties make JSON an ideal data-interchange language.

- JSON is built on two structures:
 - A collection of name/value pairs. In various languages, this is realized as an object, record, struct, dictionary, hash table, keyed list, or associative array.
 - An ordered list of values. In most languages, this is realized as an array, vector, list, or sequence.
 - These are universal data structures. Virtually all modern programming languages support them in one form or another. It makes sense that a data format that is interchangeable with programming languages also be based on these structures.
- In JSON, they take on these forms:
- An object is an unordered set of name/value pairs. An object begins with {left brace and ends with }right brace. Each name is followed by :colon and the name/value pairs are separated by ,comma.

Back-end languages and technologies:

PHP:

PHP is an open-source server-side scripting language that many devs use for web development. It is also a general-purpose language that you can use to make lots of projects, including Graphical User Interfaces (GUIs).

In this article, I will help you explore the world of PHP so you can learn how it works and its basic features. By the end, you will be able to write your first Hello World program in PHP.

What Does PHP Mean?

- The abbreviation PHP initially stood for Personal Homepage. But now it is a recursive acronym for Hypertext Preprocessor. (It's recursive in the sense that the first word itself is an abbreviation, so the full meaning doesn't follow the abbreviation.)
- The first version of PHP was launched 26 years ago. Now it's on version 8, released in November 2020, but version 7 remains the most widely used.
- PHP runs on the Zend engine, which is the most popular implementation. There are some other implementations as well, like parrot, HPVM (Hip Hop Virtual Machine), and Hip Hop, created by Facebook.
- PHP is mostly used for making web servers. It runs on the browser and is also capable of running in the command line. So, if you don't feel like showing your code output in the browser, you can show it in the terminal.

Advantages of PHP

- PHP has some advantages that have made it so popular, and it's been the go-to language for web servers for more than 15 years now. Here are some of PHP's benefits:
- Cross-Platform: PHP is platform-independent. You don't have to have a particular OS to use it because it runs on every platform, whether it's Mac, Windows, or Linux.

- Open Source: PHP is open source. The original code is made available to everyone who wants to build upon it. This is one of the reasons why one of its frameworks, Laravel, is so popular.
- Easy to learn: PHP is not hard to learn for absolute beginners. You can pick it up pretty if you already have programming knowledge.
- PHP syncs with all Databases: You can easily connect PHP to all Databases, relational and non-relational. So it can connect in no time to MySQL, Postgress, MongoDB, or any other database.
- Supportive Community: PHP has a very supportive online community. The official documentation provides guides on how to use the features and you can easily get your problem fixed while stuck.

Mysql:

MySQL is an open source relational database management system (RDBMS) with a client-server model. RDBMS is a software or service used to create and manage databases based on a relational model. Now, let's take a closer look at each term:

- **Database**

- A database is simply a collection of structured data. Think of taking a selfie: you push a button and capture an image of yourself. Your photo is data, and your phone's gallery is the database. A database is a place in which data is stored and organized. The word “relational” means that the data stored in the dataset is organized as tables. Every table relates in some ways. If the software doesn't support the relational data model, just call it DBMS.

- **Open source**

- Open source means that you're free to use and modify it. Anybody can install the software. You can also learn and customize the source code to better accommodate your needs. However, The GPL (GNU Public License) determines what you can do depending on conditions. The commercially licensed version is available if you need more flexible ownership and advanced support.

- **Client-server model**

- Computers that install and run RDBMS software are called clients. Whenever they need to access data, they connect to the RDBMS server. That's the “client-server” part.
- MySQL is one of many RDBMS software options. RDBMS and MySQL are often thought to be the same because of MySQL's popularity. A few big web applications like Facebook, Twitter, YouTube, Google, and Yahoo! all use MySQL for data storage purposes. Even though it was initially created for limited usage, it is now compatible with many important computing platforms like Linux, macOS, Microsoft Windows, and Ubuntu.

SQL:

- MySQL and SQL are not the same. Be aware that MySQL is one of the most popular RDBMS software's brand names, which implements a client-server model. So, how do the client and server communicate in an RDBMS environment? They use a domain-specific language – Structured Query Language (SQL). If you ever encounter other names that have SQL in them, like

PostgreSQL and Microsoft SQL server, they are most likely brands which also use Structured Query Language syntax. RDBMS software is often written in other programming languages, but always use SQL as their primary language to interact with the database. MySQL itself is written in C and C++.

- Computer scientist Ted Codd developed SQL in the early 1970s with an IBM based relational model. It became more widely used in 1974 and quickly replaced similar, then-outdated languages, ISAM and VISAM. History aside, SQL tells the server what to do with the data. It is similar to your WordPress password or code. You input it into the system to gain access to the dashboard area. In this case, SQL statements can instruct the server to perform certain operations:
- Data query: requesting specific information from the existing database.
- Data manipulation: adding, deleting, changing, sorting, and other operations to modify the data, the values or the visuals.
- Data identity: defining data types, e.g. changing numerical data to integers. This also includes defining a schema or the relationship of each table in the database
- Data access control: providing security techniques to protect data, this includes deciding who can view or use any information stored in the database

How Mysql works:

One or more devices (clients) connect to a server through a specific network. Every client can make a request from the

graphical user interface (GUI) on their screens, and the server will produce the desired output, as long as both ends understand the instruction. Without getting too technical, the main processes taking place in a MySQL environment are the same, which are:

- MySQL creates a database for storing and manipulating data, defining the relationship of each table.
- Clients can make requests by typing specific SQL statements on MySQL.
- The server application will respond with the requested information and it will appear on the clients' side.

That's pretty much it. From the clients' side, they usually emphasize which MySQL GUI to use. The lighter and more user-friendly the GUI is, the faster and easier their data management activities will be. Some of the **most popular MySQL GUIs** are MySQL WorkBench, SequelPro, DBVisualizer, and the Navicat DB Admin Tool. Some of them are free, while some are commercial, some run exclusively for macOS, and some are compatible with major operating systems. Clients should choose the GUI depending on their needs. For web database management, **including a WordPress site**, the most obvious go-to is phpMyAdmin.

3. Codes and screens:

In this section, we will show parts of the codes used in the designing of the website:

config.php

```
<?php  
session_start();  
$host = "localhost";  
$username = "root";  
$password = "123";  
$db = "ecommerce";  
  
$conn = mysqli_connect("$host", "$username", "$password",  
"$db");  
?>
```

The above block of code displays how the website codes connect with the database to deal with it, and it is written by PHP language.

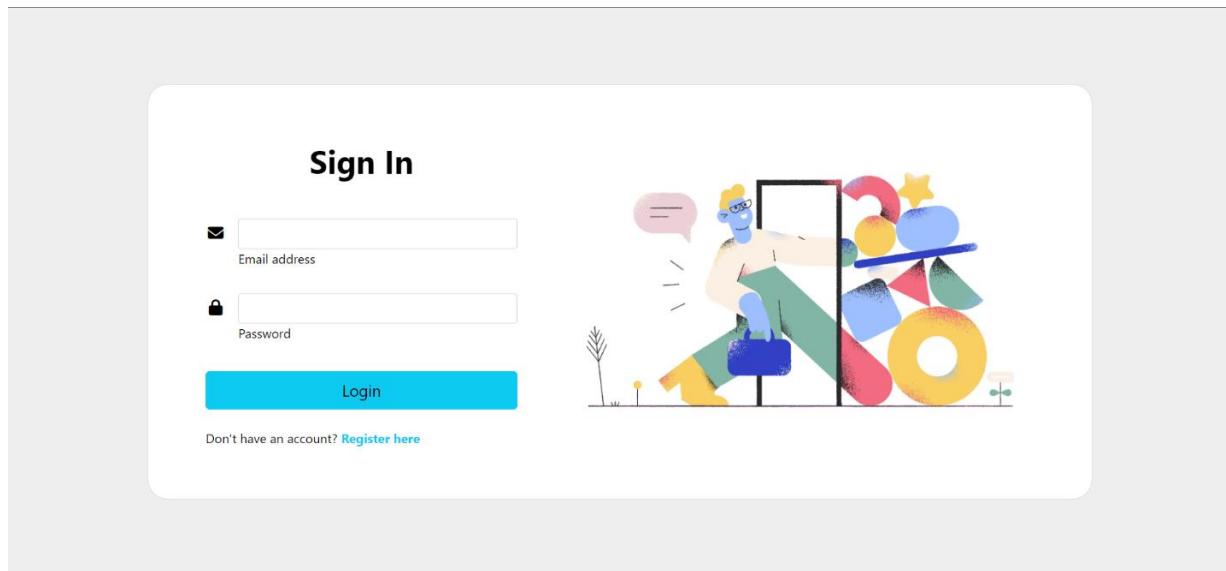


Figure 3-1 Sign in

The above form displays the login form which anyone who wants to access our website must sign in.



Figure 3-2 Home screen

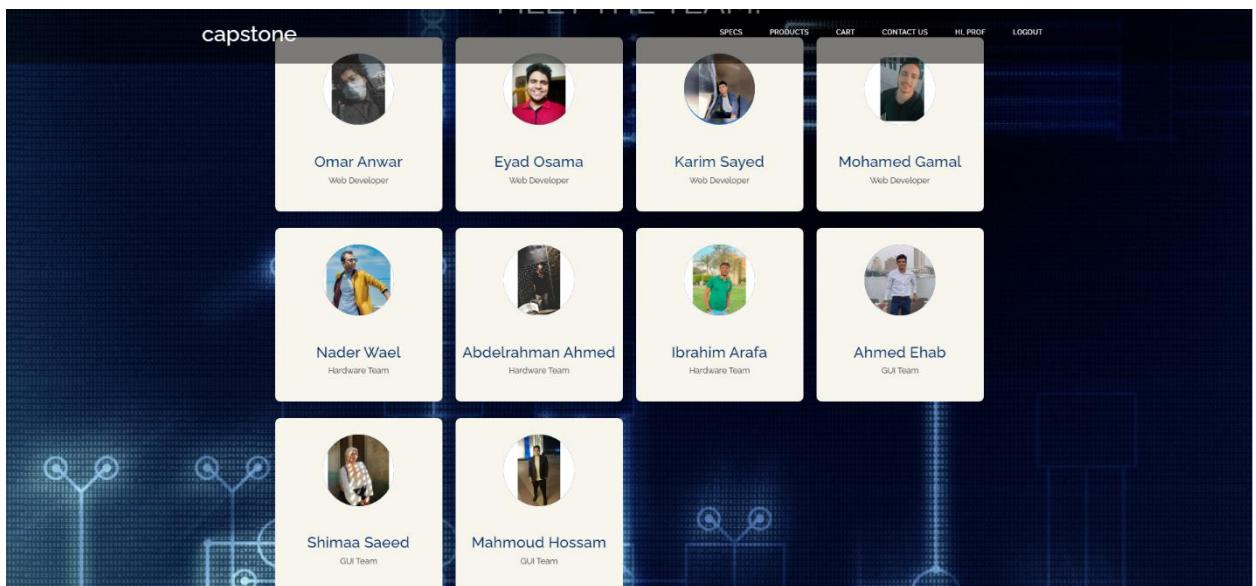


Figure 3-3 Team members

There are from the homepage which represents the team members if anyone wants to be in touch with them.

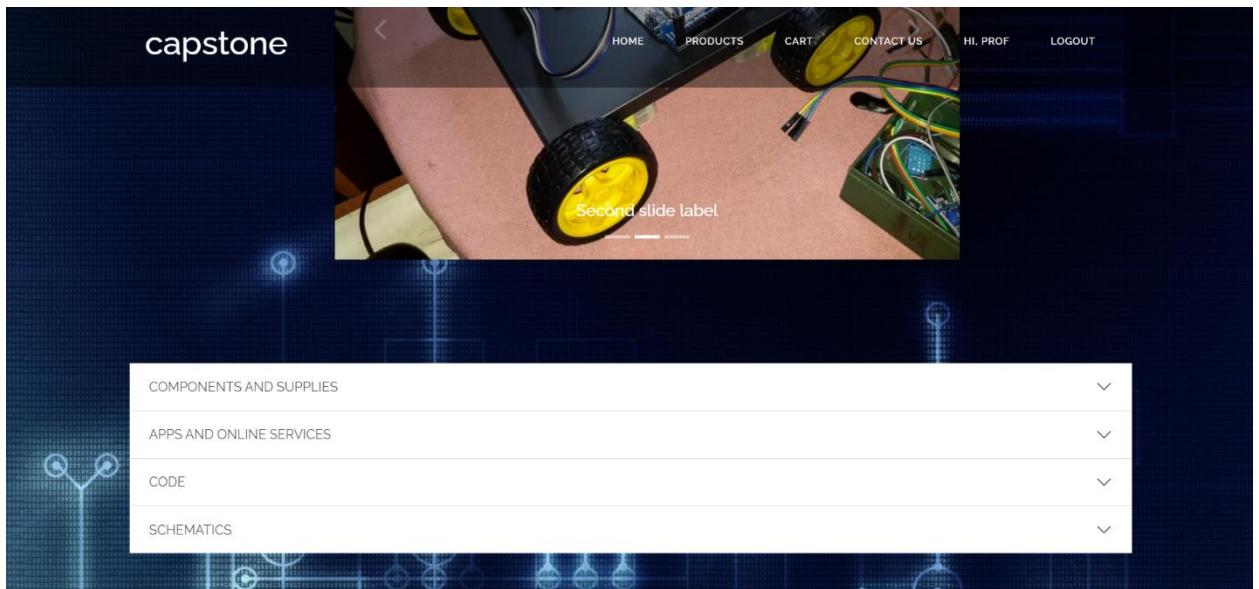


Figure 3-4 Specifications

There is a webpage that represents the specifications of the project and what the team used to build this car.

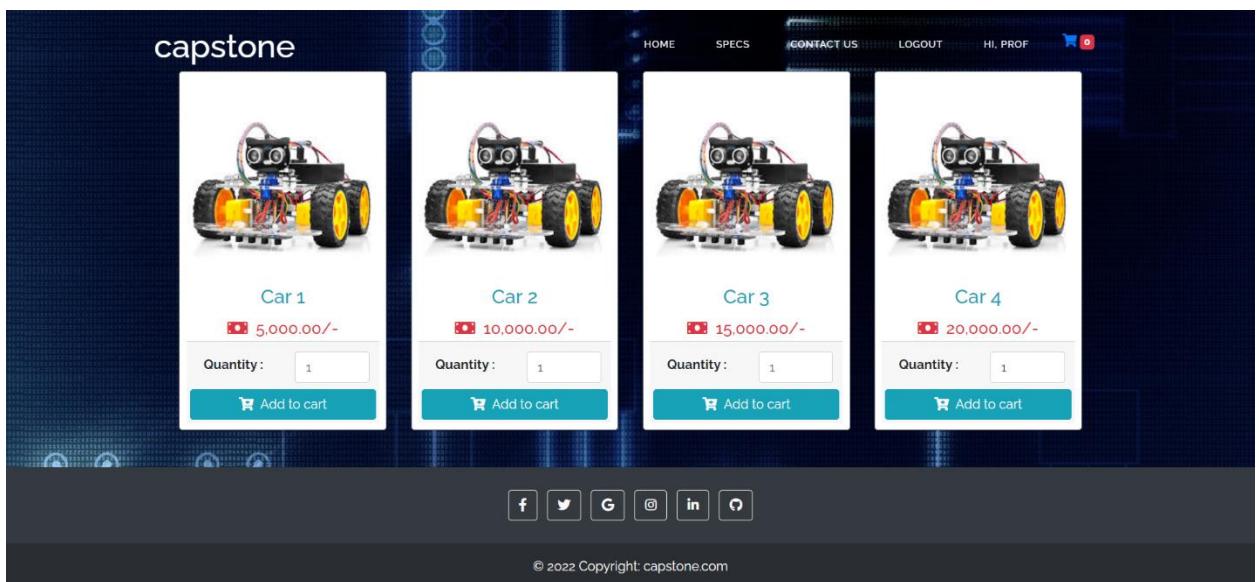
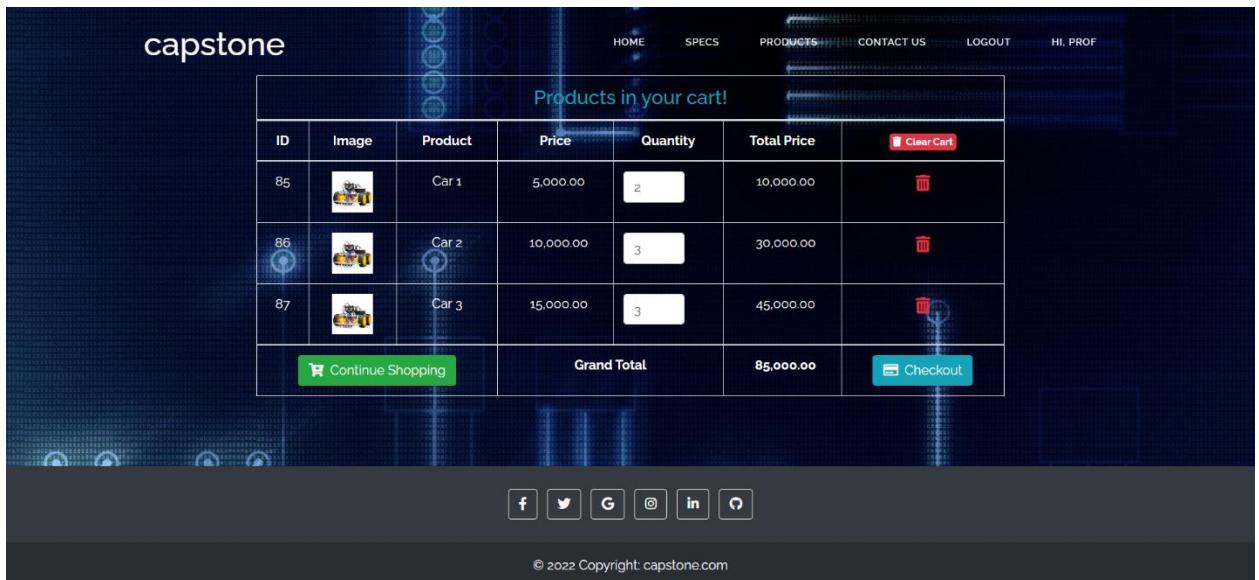


Figure 3-5 Models

It's the section that displays the models of the car and if the customers want to buy any of the project models they will start from here.



The screenshot shows a shopping cart page with a dark blue background featuring a grid pattern and various icons. At the top, there's a navigation bar with links for HOME, SPECS, PRODUCTS, CONTACT US, LOGOUT, and HI, PROF. The main title 'capstone' is at the top left. Below it, a heading says 'Products in your cart!'. A table lists three items: Car 1 (ID 85), Car 2 (ID 86), and Car 3 (ID 87). Each row includes an image, product name, ID, price, quantity, total price, and a 'Clear Cart' button. At the bottom of the table, there are buttons for 'Continue Shopping' (green) and 'Checkout' (blue).

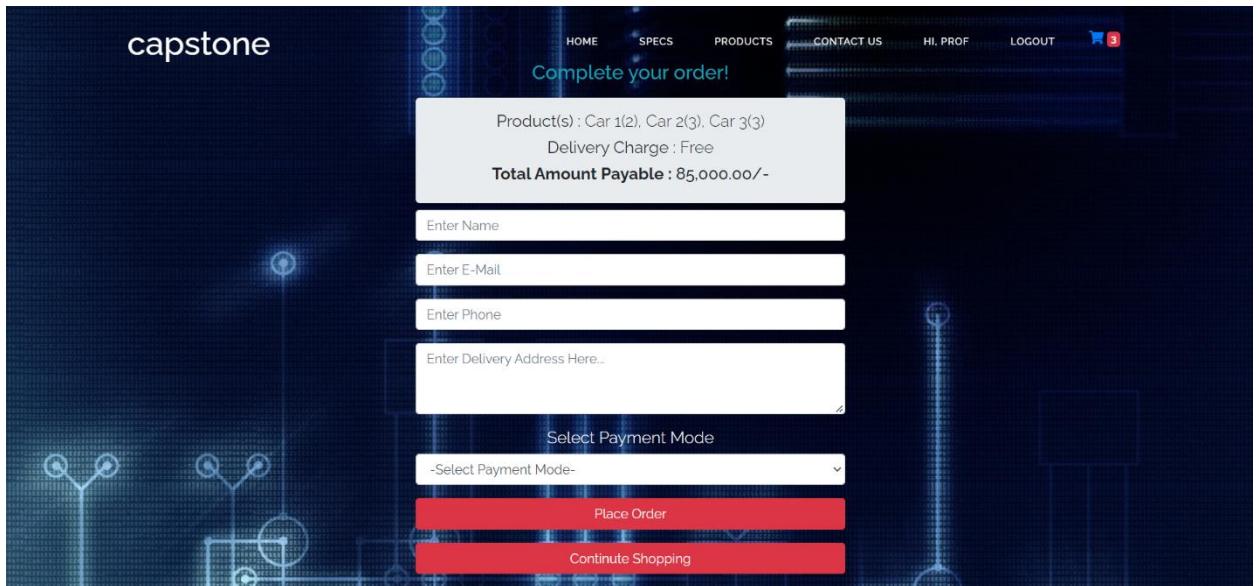
ID	Image	Product	Price	Quantity	Total Price	Clear Cart
85		Car 1	5,000.00	2	10,000.00	
86		Car 2	10,000.00	3	30,000.00	
87		Car 3	15,000.00	3	45,000.00	

Continue Shopping Grand Total: 85,000.00 Checkout

© 2022 Copyright: capstone.com

Figure 3-6 Cart

It's the cart webpage that gives the user more than one choice to choose if he wants to place an order or cancel the shopping or continue.



The screenshot shows a page titled 'Complete your order!' with a dark blue background. At the top, there's a navigation bar with links for HOME, SPECS, PRODUCTS, CONTACT US, HI, PROF, LOGOUT, and a shopping cart icon with a '3' notification. The main area displays the products in the cart: 'Product(s) : Car 1(2), Car 2(3), Car 3(3)', 'Delivery Charge : Free', and 'Total Amount Payable : 85,000.00/-'. Below this, there are four input fields: 'Enter Name', 'Enter E-Mail', 'Enter Phone', and 'Enter Delivery Address Here...'. Further down, there's a section for 'Select Payment Mode' with a dropdown menu containing '-Select Payment Mode-' and two buttons: 'Place Order' (red) and 'Continue Shopping' (red).

Figure 3-7 User data

It's the last step of buying the project models that ask for detailed data from the user to deliver it.

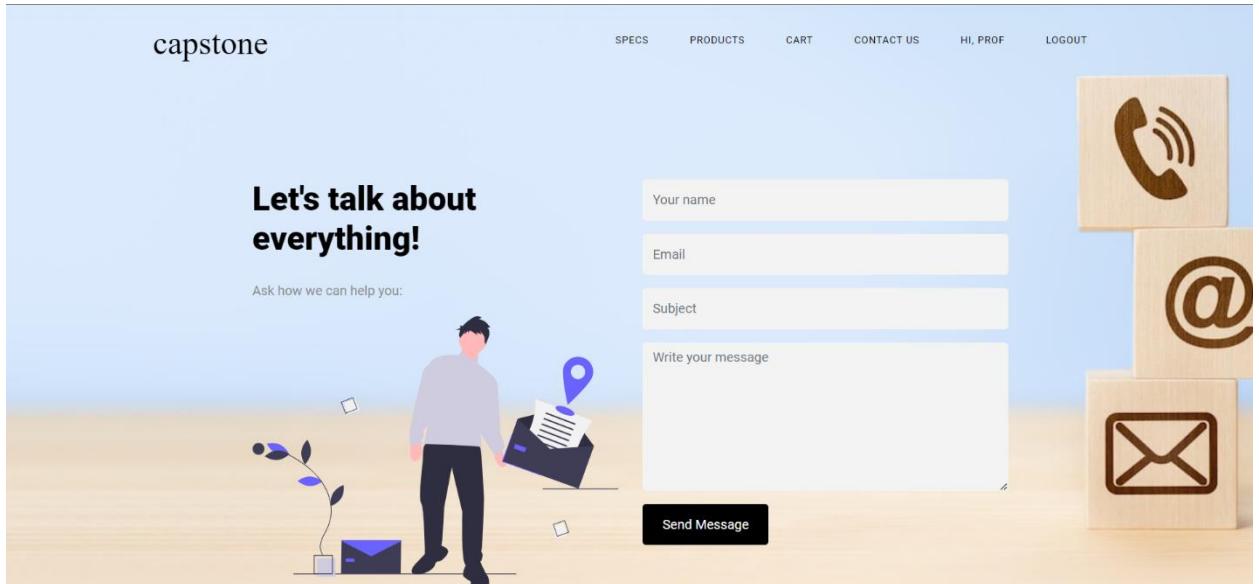


Figure 3-8 Contact-us

It's a contact-us webpage that let users and critics who want to communicate with us leave some comments.

4. Summary:

In light of the modern development in the world of technology, there must be inventions that make it easier for us to coexist and deal with these matters, so we produced this website to simulate this progress.

This website is a mix of displaying information and a simple eCommerce website.

The first section is used to display information about team members and to document the specifications, code, and schematics of the project.

The last section is used to sell our IoT project and to interact with the admins such as by leaving comments.

CHAPTER 5

Conclusion and Future Work

CHAPTER 5

CONCLUSION AND FUTURE WORK

7. 1 Conclusions

A remote control vehicle is designed and we using the GUI application to control the car by using Bluetooth. These devices are always controlled by humans and take no action autonomously just in one case car stop automatically if there is any obstacle the car will stop by using ultrasonic sensor. The main target in such vehicles would be to safely reach a designated point and measures temperature and indicate if there is gas in the area, maneuver the area and reach back to the point of origin.

GPS will send coordinates of the car to GUI application to know where the gas is leaked and send worker to solve the problem.

In this project we make use of the Bluetooth technology to control our machine car. This machine can be controlled by any human using his GUI application, by connecting it with the Bluetooth module present inside our car. User can perform actions like moving forward, backward, moving left and right by the means of command using his-her GUI app. The task of controlling our car is taken care by the Arduino Mega with micro controller ATMEGA2560, 16 MHz crystal oscillator, 8 KB SRAM (Static Random Accessible Memory) and 256 KB flash memory. Arduino play a major role in the control section and had made it easier to convert digital signals and analogue signals into physical movements. The major reason for using a Bluetooth based tech is that we can change the remote anytime – mobiles

phones, tablets and laptops and physical barriers like wall or doors do not affect the car controls.

A **GUI** is represented actions that can be taken by the user

Include GUI object buttons: that performs an action in a program when pressed. Using these objects, a user can use the computer without having to know commands.

The **Website** for selling our product, to show the cost of components and how we collect and made the car, The code of Arduino IDE and GUI using C#.

7. 2 Future Works

- Add decision making from the car by itself example: the car moves into the places every 5 minutes by itself if there is any obstacle the car will avoid it by ultrasonic sensor that will make car rerouting.
- Detect if there any problem and sending reports.
- Increased distance control by using WIFI instead of Bluetooth for remote control ability if required.
- The app will have authorization to allow only the users who buy the car to use it.
- In light of the developments in the world of technology, it is imperative that we develop some technologies and services to keep pace with the situation and to make it better, and among these developments for our website is the online chatting service, it will provide the opportunity to make the user choose the right product for him at the same time and will facilitate communication between supervisors and users.
- The second one that will help the admins is to develop a website that have privilege, so that they can see who signed up and edit or delete them, who commented, also in admins' website they can add, edit or delete the offered products , finally they can see who bought the car and detailed information about that person.

Bibliography

(2021)

Bibliography

- [1] Wang, F.Q. (2012) Research New Type of Supply Chain Management Model Based on the Technology of Internet of Things—With Radio Frequency Identification (RFID) Technology as an Example.
- [2] Shen, S.B., Fang, Q.L. and Zong, P. (2009) Research of Internet of Things and related technology. Journal of Nanjing University of Posts and Telecommunications (Natural Science), 6, 1-11.
- [3] Wang, B.Y. (2009) Summary of IoT Technology Research. Journal of Electronic Measurement and The instrument, 12, 1-7.
- [4] Lin, G. (2012) Research on Complex Event of Supply Chain Decision Support Based on Internet of Things.