INTRODUCTION

1. INTRODUCTION

Our daily life, the environment and its condition is very important for our health as it will impact the quality of life for all of earth's inhabitants. Consequently, the issues from environment and the air quality in industrial area are discussed to increase the alertness and responsibility health. The dangerous gases such as CH4, and CO will bring harmful effect towards human as they may cause explosions and CO poisoning accident in most industrial areas Thus, a gas detector is invented to ease human on detecting the presence of those dangerous gases within an area to prevent disaster happen. Nowadays, the gas detector has been innovated into various ways of detection, for example infrared thermal imaging gas leak detection ,wireless gas sensor network.

So, our team decided to use this opportunity to develop a project which could help to decrease the dangers caused by gas leakage which leads to various health issues caused to the human beings, animals and birds and also materialistic losses for the industries so for these, A monitoring system for gas leakage detection needs to be developed. For the development of this system, the combustible gas sensor (MQ9) was used in order to detect the presence of methane (CH4) and carbon monoxide gas (CO) and LPG. Monitoring system by using NodeMCU as the microcontrollers for the whole system. This sensor will detect the concentration of the gas according to the voltage output of sensor and a buzzer acts as the alert message which acts as the alarm for the whole system. And by connecting to the cloud the user get a message to the PC, laptop or mobile so he can access from anywhere and get alerted.

These systems are designed to quickly detect any kind of gas leakage in premises, pipes and other protected areas. These send a signal to the panel that alarms the sounder/hooter. Gas Leak Detection Systems have been designed with an IOT technology which is user friendly and with no reset or mute options for a fail safe operation at low cost.

LITERATURE SURVEY

2. LITERATURE SURVEY

2.1. HISTORY OF IOT

The concept of a network of smart devices was discussed as early as 1982, with a modified Coke able to report its inventory and whether newly loaded drinks were cold or not. Mark Weiser's 1991 paper on ubiquitous computing, "The Computer of the 21st Century", as well as academic venues such as UbiComp and PerCom produced the contemporary vision of the IoT. In 1994, Reza Raji described the concept in IEEE Spectrum as "[moving] small packets of data to a large set of nodes, so as to integrate and automate everything from home appliances to entire factories". Between 1993 and 1997, several companies proposed solutions like Microsoft's at Work or Novell's NEST. The field gained momentum when Bill Joy envisioned device-to-device communication as a part of his "Six Webs" framework, presented at the World Economic Forum at Davos in 1999. The term "Internet of things" was likely coined by Kevin Ashton of Procter & Gamble, later MIT's Auto-ID center, in 1999, though he prefers the phrase "Internet for things". At that point, he viewed radio-frequency identification (RFID) as essential to the Internet of things, which would allow computers to manage all individual things. Defining the Internet of things as "simply the point in time when more 'things or objects' were connected to the Internet than people", Cisco Systems estimated that the IoT was "born" between 2008 and 2009, with the things/people ratio growing from 0.08 in 2003 to 1.84 in 2010.

The key driving force behind the Internet of things is the MOSFET (metal-oxide-semiconductor field-effect transistor, or MOS transistor), which was originally invented by Mohamed M. Atalla and Dawon Kahng at Bell Labs in 1959. The MOSFET is the basic building block of most modern electronics, including computers, smartphones, tablets and Internet services. MOSFET scaling miniaturization at a pace predicted by Dennard scaling and Moore's law has been the driving force behind technological advances in the electronics industry since the late 20th century. MOSFET scaling has been extended into the early 21st century with advances such as reducing power consumption, silicon-on-insulator (SOI) semiconductor device fabrication, and multi-core processor technology, leading up to the Internet of things, which is being driven by MOSFETs scaling down to Nano electronic levels with reducing energy consumption.

The Internet of Things (IoT) is a system of interrelated computing devices, mechanical and digital machines, objects, animals or people that are provided with unique identifiers (UIDs) and the ability to transfer data over a network without requiring human-to-human or human-to-computer interaction. The definition of the Internet of Things has evolved due to the convergence of multiple technologies, real-time analytics, machine learning, commodity sensors, and embedded systems. Traditional fields of embedded systems, wireless sensor networks, control systems, automation (including home and building automation), and others all contribute to enabling the Internet of Things. In the consumer market, IoT technology is most synonymous with products pertaining to the concept of the "smart home", covering devices and appliances (such as lighting fixtures, thermostats, home security systems and cameras, and other home appliances) that support one or more common ecosystems, and can be controlled via devices associated with that ecosystem, such as smartphones and smart speakers. There are a number of serious concerns about dangers in the growth of IoT, especially in the areas of privacy and security; and consequently industry and governmental moves to begin to address these.

2.2. APPLICATION

The extensive set of applications for IoT devices is often divided into consumer, commercial, industrial, and infrastructure spaces.

2.2.1. Consumer applications

A growing portion of IoT devices are created for consumer use, including connected vehicles, home automation, wearable technology, connected health, and appliances with remote monitoring capabilities.

2.2.2. Smart home

IoT devices are a part of the larger concept of home automation, which can include lighting, heating and air conditioning, media and security systems. Long-term benefits could include energy savings by automatically ensuring lights and electronics are turned off. A smart home or automated home could be based on a platform or hubs that control smart devices and appliances. For instance, using Apple's Home Kit, manufacturers can have their home products and accessories controlled by an application in iOS devices such as the iPhone and the Apple Watch.

2.2.3. Medical and healthcare

The Internet of Medical Things (IoMT), (also called the Internet of health things), is an application of the IoT for medical and health related purposes, data collection and analysis for research, and monitoring. The IoMT has been referenced as "Smart Healthcare", as the technology for creating a digitised healthcare system, connecting available medical resources and healthcare services. IoT devices can be used to enable remote health monitoring and emergency notification systems. These health monitoring devices can range from blood pressure and heart rate monitors to advanced devices capable of monitoring specialised implants, such as pacemakers, Fitbit electronic wristbands, or advanced hearing aids. Some hospitals have begun implementing "smart beds" that can detect when they are occupied and when a patient is attempting to get up. It can also adjust itself to ensure appropriate pressure and support is applied to the patient without the manual interaction of nurses. A 2015 Goldman Sachs report indicated that healthcare IoT devices "can save the United States more than $300 billion in annual healthcare expenditures by increasing revenue and decreasing cost. Moreover, the use of mobile devices to support medical follow-up led to the creation of 'm-health', used "to analyse, capture, transmit and store health statistics from multiple resources, including sensors and other biomedical acquisition systems".

Specialized sensors can also be equipped within living spaces to monitor the health and general well-being of senior citizens, while also ensuring that proper treatment is being administered and assisting people regain lost mobility via therapy as well. These sensors create a network of intelligent sensors that are able to collect, process, transfer, and analyse valuable information in different environments, such as connecting in-home monitoring devices to hospital-based systems. Other consumer devices to encourage healthy living, such as connected scales or wearable heart monitors, are also a possibility with the IoT. End-to-end health monitoring IoT platforms are also available for antenatal and chronic patients, helping one manage health vitals and recurring medication requirements

2.2.4. Industrial applications

Also known as IoT, industrial IoT devices acquire and analyse data from connected equipment, (OT) operational technology, locations and people. Combined with operational technology (OT) monitoring devices, IIOT helps regulate and monitor industrial systems.

2.2.5. Manufacturing

The IoT can realize the seamless integration of various manufacturing devices equipped with sensing, identification, processing, communication, actuation, and networking capabilities. Based on such a highly integrated smart cyber-physical space, it opens the door to create whole new business and market opportunities for manufacturing. Network control and management of manufacturing equipment, asset and situation management, or manufacturing process control bring the IoT within the realm of industrial applications and smart manufacturing as well. The IoT intelligent systems enable rapid manufacturing of new products, dynamic response to product demands, and real-time optimization of manufacturing production and supply chain networks, by networking machinery, sensors and control systems together. Digital control systems to automate process controls, operator tools and service information systems to optimize plant safety and security are within the purview of the IoT. But it also extends itself to asset management via predictive maintenance, statistical evaluation, and measurements to maximize reliability. Industrial management systems can also be integrated with smart grids, enabling real-time energy optimization. Measurements, automated controls, plant optimization, health and safety management, and other functions are provided by a large number of networked sensors.

2.2.6. Agriculture

There are numerous IoT applications in farming such as collecting data on temperature, rainfall, humidity, wind speed, pest infestation, and soil content. This data can be used to automate farming techniques, take informed decisions to improve quality and quantity, minimize risk and waste, and reduce effort required to manage crops. For example, farmers can now monitor soil temperature and moisture from a far, and even apply IoT-acquired data to precision fertilization programs.

2.2.7. Infrastructure applications

Monitoring and controlling operations of sustainable urban and rural infrastructures like bridges, railway tracks and on- and offshore wind-farms is a key application of the IoT.The IoT infrastructure can be used for monitoring any events or changes in structural conditions that can compromise safety and increase risk. The IoT can benefit the construction industry by cost saving, time reduction, better quality workday, paperless workflow and increase in productivity. It can help in taking faster decisions and save money with Real-Time Data Analytics. It can also be used for scheduling repair and maintenance activities in an efficient manner, by coordinating tasks between different service providers and users of these facilities. IoT devices can also be used to control critical infrastructure like bridges to provide access to ships. Usage of IoT devices for monitoring and operating infrastructure is likely to improve incident management and emergency response coordination, and quality of service, up-times and reduce costs of operation in all infrastructure related areas.Even areas such as waste management can benefit from automation and optimization that could be brought in by the IoT.

2.2.8. Energy management

Significant numbers of energy-consuming devices (e.g. switches, power outlets, bulbs, televisions, etc.) already integrate Internet connectivity, which can allow them to communicate with utilities to balance power generation and energy usage and optimize energy consumption as a whole. These devices allow for remote control by users, or central management via a cloud-based interface, and enable functions like scheduling (e.g., remotely powering.) The smart grid is a utility-side IoT application; systems gather and act on energy and power-related.Using advanced metering infrastructure (AMI) Internet-connected devices, electric utilities not only collect data from end-users, but also manage distribution automation devices like transformers.

2.2.9. Living Lab

Another example of integrating the IoT is Living Lab which integrates and combines research and innovation process, establishing within a public-private-people-partnership. There are currently 320 Living Labs that use the IoT to collaborate and share knowledge between stakeholders to co-create innovative and technological products. For companies to implement and develop IoT services for smart cities, they need to have incentives. The government play key roles in smart cities projects as changes in policies will help cities to implement the IoT which provides effectiveness, efficiency, and accuracy of the resources that are being used. For instance, the government provides tax incentives and cheap rent, improves public transports, and offers an environment where start-up companies, creative industries, and multinationals may co-create, share common infrastructure and labor markets, and take advantages of locally embedded technologies, production process, and transaction costs. The relationship between the technology developers and governments who manage city's assets, is key to provide open access of resources to users in an efficient way.

2.2.10. Military Applications

The Internet of Military Things (IoMT) is a term that describes the application of IoT technologies in the military domain for the purposes of reconnaissance, surveillance, and other combat-related objectives. It is heavily influenced by the future prospects of warfare in an urban environment and involves the use of sensors, munitions, vehicles, robots, human-wearable biometrics, and other smart technology that is relevant on the battlefield.

2.2.11. Internet of Battlefield Thing

The Internet of Battlefield Things (IoBT) is a project initiated and executed by the U.S. Army Research Laboratory (ARL) that focuses on the basic science related to IoT that enhance the capabilities of Army soldiers In 2017, ARL launched the Internet of Battlefield Things Collaborative Research Alliance (IoBT-CRA), establishing a working collaboration between industry, university, and Army researchers to advance the theoretical foundations of IoT technologies and their applications to Army operations.

2.2.12. Ocean of Things

The Ocean of Things project is a DARPA-led program designed to establish an Internet of Things across large ocean areas for the purposes of collecting, monitoring, and analysing environmental and vessel activity data. The project entails the deployment of about 50,000 floats that house a passive sensor suite that autonomously detect and track military and commercial vessels as part of a cloud-based network.

2.3. WHAT IS IOT

The internet of things, or IoT, is a system of interrelated computing devices, mechanical and digital machines, objects, animals or people that are provided with unique identifiers (UIDs) and the ability to transfer data over a network without requiring human-to-human or human-to-computer interaction. A complete IoT system integrates four distinct components: sensors/devices, connectivity, data processing, and a user interface. Below I will briefly explain each component and what it does.

2.3.1. Sensors

First, sensors or devices collect data from their environment. This could be as simple as a temperature reading or as complex as a full video feed. I use “sensors/devices,” because multiple sensors can be bundled together or sensors can be part of a device that does more than just sense things. For example, your phone is a device that has multiple sensors (camera, accelerometer). However, whether it’s a standalone sensor or a full device, in this first step data is being collected from the environment by something.

2.3.2. Connectivity

Next, that data is sent to the cloud (what’s the cloud?), but it needs a way to get there. The sensors/devices can be connected to the cloud through a variety of methods including: cellular, satellite, WiFi, Bluetooth, low-power wide-area networks (LPWAN), or connecting directly to the internet via Ethernet. Each option has trade-offs between power consumption, range and bandwidth (here’s a simple explanation). Choosing which connectivity option is best comes down to the specific IoT application, but they all accomplish the same task: getting data to the cloud.

2.3.4. Data Processing

Once the data gets to the cloud, software performs some kind of processing on it. This could be very simple, such as checking that the temperature reading is within an acceptable range. Or it could also be very complex, such as using computer vision on video to identify objects (such as intruders in your house).But what happens when the temperature is too high or if there is an intruder in your house? That’s where the user come

2.3.5. User Interface

Next, the information is made useful to the end-user in some way. This could be via an alert to the user (email, text, notification, etc.). For example, a text alert when the temperature is too high in the company’s cold storage. Also, a user might have an interface that allows them to proactively check in on the system. Depending on the IoT application, the user may also be able to perform an action and affect the system. For example, the user might remotely adjust the temperature in the cold storage via an app on their phone. And some actions are performed automatically. Rather than waiting for you to adjust the temperature, the system could do it automatically via predefined rules. And rather than just call you to alert you of an intruder, the IoT system could also automatically notify relevant authorities

SYSTEM ANALYSIS

3. SYSTEM ANALYSIS

3.1. EXSISITING SYSTEM

3.1.1. The Laser Approach

When it comes to gas, it is always a serious business. Gas operators' maintenance programs are specifically designed to limit safety hazards through close monitoring of their pipelines and installations. But in addition to the concern for assets and human safety, global warming must now also be taken into account, along with the need of operational cost reductions. Methane, the main component of natural gas, is an odourless, invisible, yet combustible gas. Like CO2, it is a greenhouse gas, but its effect on climate change is far greater than CO2. Methane is responsible for a quarter of the global warming we experience today. In response, the natural gas industry has to proactively deal with methane emissions mitigation all along the gas chain from production down to the end consumers. This is why pipeline operators and gas utilities worldwide are currently reviewing their integrity management system with tougher Leak Detection and Repair (LDAR) programs

Laser-based gas sensor technology is an extremely effective tool for detecting and quantifying polluting gases such as carbon dioxide or methane owing to some key advantages. Flame ionization consists in passing a gas sample through a hydrogen flame. The presence of gas produces a current between two electrodes.

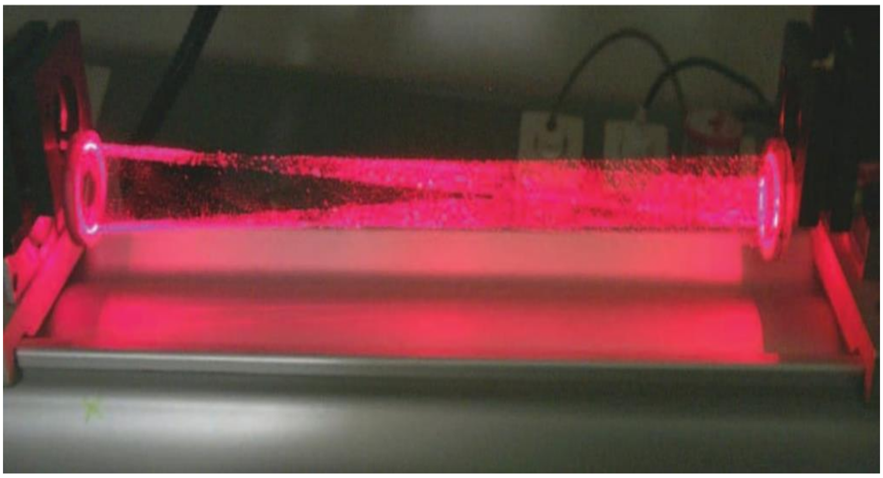

Fig 3.1.1 Laser system for detecting gases.

3.1.2. Advantages

- Because of the very narrow 0.3 nm line width of the laser emission, there is no interference from other gases.
- Response times are in the order 1 second. This allow for fine resolution/control when making process measurements.
- The intense laser light concentrated at the absorption wavelength enables path lengths up to 1 km to be measured.
- An average measurement is taken over the total path so that a narrow plume of gas has less chance of escaping detection.
- Can easily be conformed to be 'Intrinsically Safe'.
- Low maintenance and low operating costs.
- Reliable technology.

3.1.3. Disadvantages

- They are very expensive.
- Only one gas can be measured with each instrument.
- When heavy dust, steam or fog blocks the laser beam, the system will not be able to take measurements. This is also the case when a person or vehicle blocks the path.

3.2 PROPOSED SYSTEM

Internet of Things aim towards making life simpler by automating every small task around us. As much is IoT helping in automating tasks, the benefits of IoT can also be extended for enhancing the existing safety standards. Safety has always been an important criterion while designing home, buildings, industries as well as cities. The increased concentration of certain gases in the atmosphere can prove to be extremely dangerous. These gases might be flammable at certain temperature and humidity conditions, toxic after exceeding the specified concentration limits or even a contributing factor in the air pollution of an area leading to problems such as smog and reduced visibility which can in turn cause severe accidents and also have adverse effect on the health of people. Most of the societies have fire safety mechanism. But it can use after the fire exists. In order to have a control over such conditions we proposed system that uses sensors which is capable of detecting the gases such as LPG, CO2, CO and CH4.

This system will not only able to detect the leakage of gas but also alerting through audible alarms. Presence of excess amounts of harmful gases in environment then this system can notify the user. System can notify to society admin about the condition before mishap takes place through a message.

System consists of gas detector sensors, NodeMCU and Cloud server. One Society authority person can register the all flat member user to our system. Society admin can add the details of per flat user such as user name, mobile number, per user flat sensor details information. Society admin can configure the threshold value of each sensor. System hardware can be deployed on each flat. Sensors can sense the value per time. System can send the values to cloud server. Server can Check that the sensor values was existed the threshold value. If sensor value can cross the limit the server can send the command to hardware for buzzing the alarm. Server also sends the notification message to user.

IoT is an expanding network of physical devices that are linked with different types of sensors and with the help of connectivity to the internet; they are able to exchange data. Through IoT, internet has now extended its roots to almost every possible thing present around us and is no more limited to our personal computers and mobile phones. Safety, the elementary concern of any project, has not been left untouched by IoT. Gas Leakages in open or closed areas can prove to be dangerous and lethal. The traditional Gas Leakage Detector Systems though have great precision, fail to acknowledge a few factors in the field of alerting the people about the leakage. Therefore we have used the IoT technology to make a Gas Leakage Detector having Smart Alerting techniques involving calling, sending text message and an e-mail to the concerned authority and an ability to predict hazardous situation so that people could be made aware in advance by performing data analytics on sensor readings.

SYSTEM REQUIREMENTS

4. SYSTEM REQUIREMENTS

4.1. HARDWARE REQUIREMENTS

4.1.1. NodeMCU

NodeMCU is open source platform, their hardware design is open for edit/modify/build. NodeMCU Dev Kit/board consist of ESP8266 Wi-Fi enabled chip. The ESP8266 is a low-cost Wi-Fi chip developed by Espressif Systems with TCP/IP protocol.

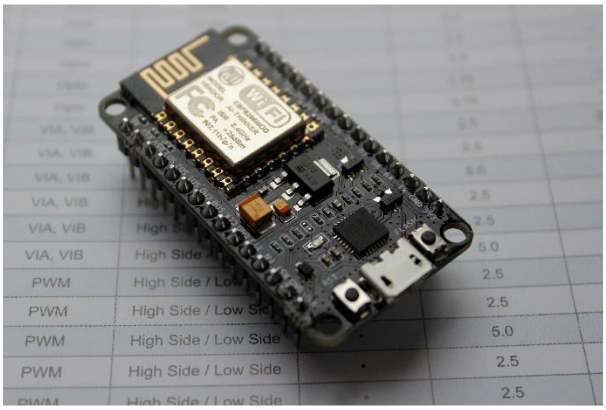

Fig 4.1.1 NodeMCU

4.1.2. MQ9 Sensor

This is MQ-9 Carbon Monoxide, Methane, and LPG Gas Sensor Module can be used to sense Carbon Monoxide and Methane Gas. Sensitive material of MQ-9 gas sensor is SnO_2, which with lower conductivity in clean air. It makes detection by the method of cycle high and low temperature, and detect CO when the low temperature (heated by 1.5V). The sensor's conductivity is higher along with the gas concentration rising. When high temperature (heated by 5.0V), it detects Methane, Propane etc. combustible gas and cleans the other gases adsorbed under low temperature.

Fig 4.1.2 MQ9

4.1.3. Buzzer (Alarm)

Electromagnetic sound emitter (buzzer) HCM12X converts electrical signal into sound. An electromagnet with a magnetic membrane is placed on the housing. With the passage of electric current through the magnet an alternating magnetic field is formed, under the influence of which the membrane begins to oscillate, thereby creating a sound signal.

Fig 4.1.3 Buzzer

4.1.4. Breadboard

Breadboards are one of the most fundamental pieces when learning how to build circuits. Once you are done you should have a basic understanding of how breadboards work and be able to build a basic circuit on a breadboard.

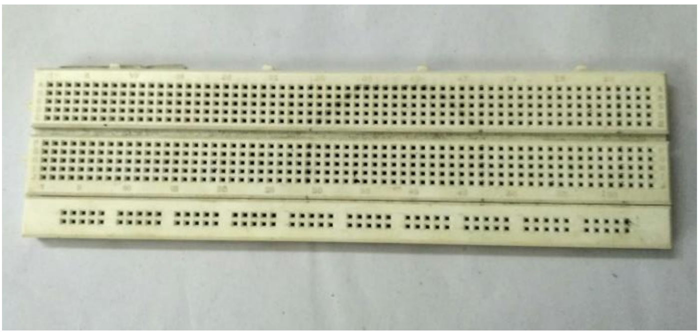

Fig 4.1.4 Breadboard

4.1.5. Jumper Wires

Jumper wires are used to establish an electrical connection between two points in a circuit.

Fig 4.1.5 Jumper Wires

4.2 SOFTWARE REQUIREMENTS

4.2.1 Arduino IDE

Arduino IDE is an open source software that is mainly used for writing and compiling the code into the Arduino Module. It is an official Arduino software, making code compilation too easy that even a common person with no prior technical knowledge can get their feet wet with the learning process. It is easily available for operating systems like MAC, Windows, and Linux and runs on the Java Platform that comes with inbuilt functions and commands that play a vital role for debugging, editing and compiling the code in the environment. A range of Arduino modules available including Arduino Uno, Arduino Mega, Arduino Leonardo, Arduino Micro and many more. Each of them contains a microcontroller on the board that is actually programmed and accepts the information in the form of code. The main code, also known as a sketch, created on the IDE platform will ultimately generate a Hex File which is then transferred and uploaded in the controller on the board. The IDE environment mainly contains two basic parts: Editor and Compiler where former is used for writing the required code and later is used for compiling and uploading the code into the given Arduino Module.This environment supports both C and C++ languages.

The Arduino IDE supplies a software library from the Wiring project, which provides many common input and output procedures. User-written code only requires two basic functions, for starting the sketch and the main program loop, that are compiled and linked with a program stub main() into an executable cyclic executive program with the GNU toolchain, also included with the IDE distribution. The Arduino IDE employs the program avrdud*e* to convert the executable code into a text file in hexadecimal encoding that is loaded into the Arduino board by a loader program in the board's firmware.

The IDE environment mainly distributed into three sections

1. Menu Bar

2. Text Editor

3. Output Pane

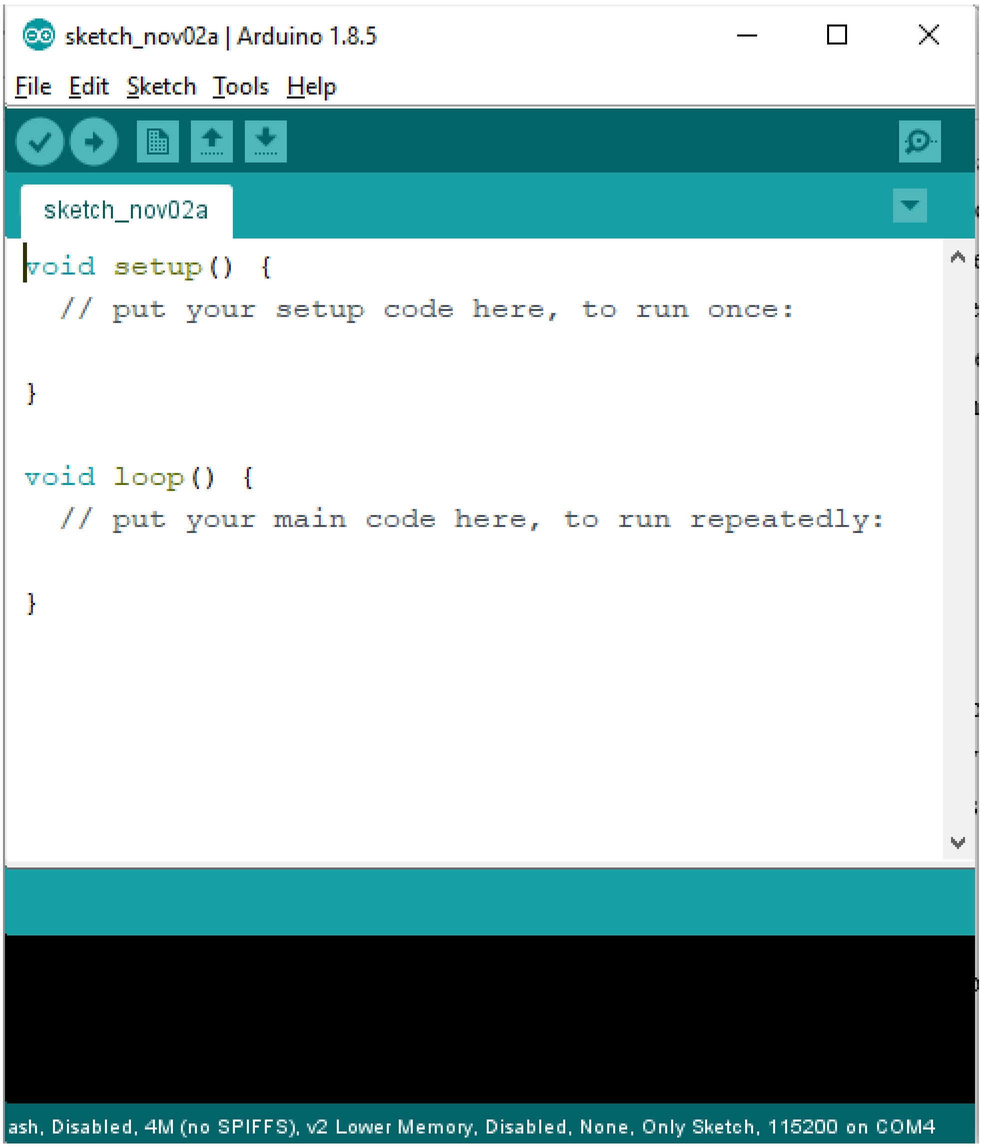

Fig 4.2.1.1 Arduino IDE new page to write programs.

A sketch is the name that Arduino uses for a program. It's the unit of code that is uploaded to and run on an Arduino board. One can only upload a single sketch to a arduino board, it's a rule one arduino board, one sketch to run. If you have multiple sketches each with their own setup() and loop() function then you can't combine them by using the tabbed edit windows, as again you can only compile and load one sketch. Your sketch undergoes minor changes (e.g. automatic generation of function prototypes) and then is passed directly to a C/C++ compiler (avr-g++).

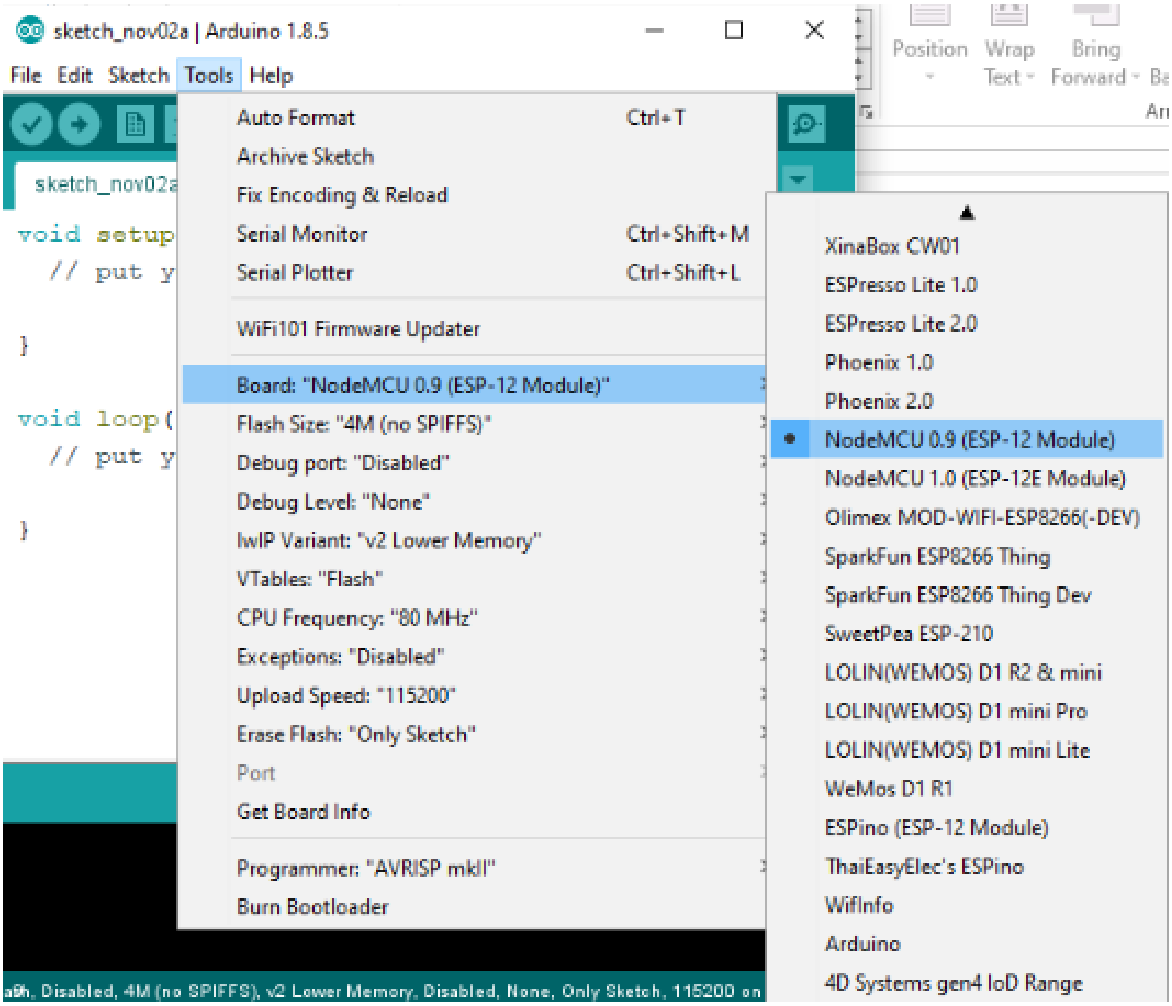

Fig 4.2.1.2 Selecting board in which we are uploading the code.

To program the NodeMCU using the Arduino IDE.

Step 1:

Connect your NodeMCU to your computer. You need a USB micro B cable to connect the board.

Step 2:

Open Arduino IDE. You need to have at least Arduino IDE version 1.6.4 to proceed with this.

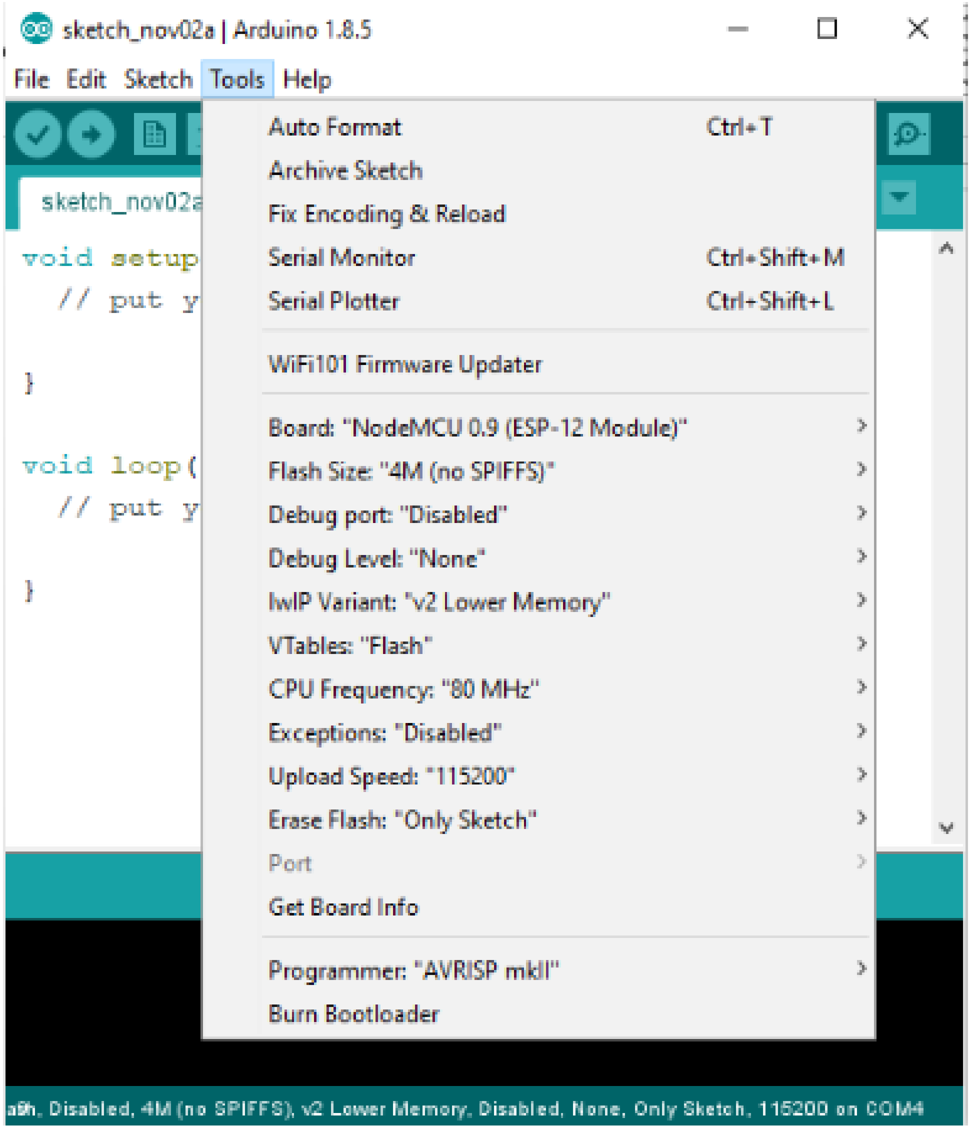

Fig 4.2.1.3 Selecting port in which we are uploading the code.

Port Registers. Port registers allow for lower-level and faster manipulation of the i/o pins of the microcontroller.

The Tools menu:

Boards: Selects the board you want to work with.

Serial Port: Selects the Serial Port to which your board is connected.

Serial Monitor: Opens the Serial Monitor Window for the serial port you have selected.

Other Serial Ports.

Programmers.

Upload Using Programmer.

Automatic Debugging.

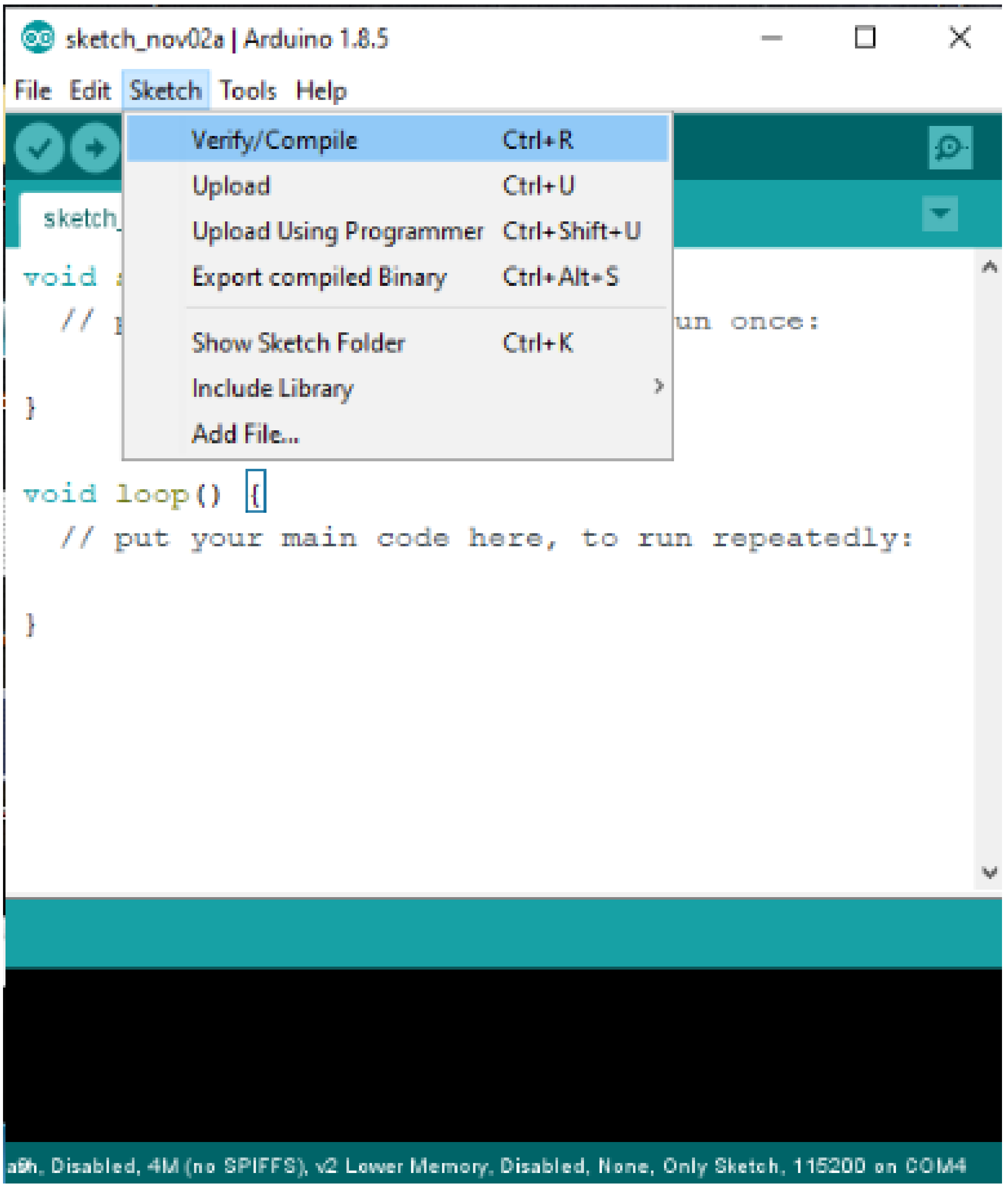

Fig 4.2.1.3 Compiling written code.

Code → Compile → Upload → Run

At the core of Arduino, is the ability to compile and run the code. After writing the code in the IDE you need to upload it to the Arduino. Clicking the Upload button (the right-facing arrow icon), will compile the code and upload it if it passed compilation.

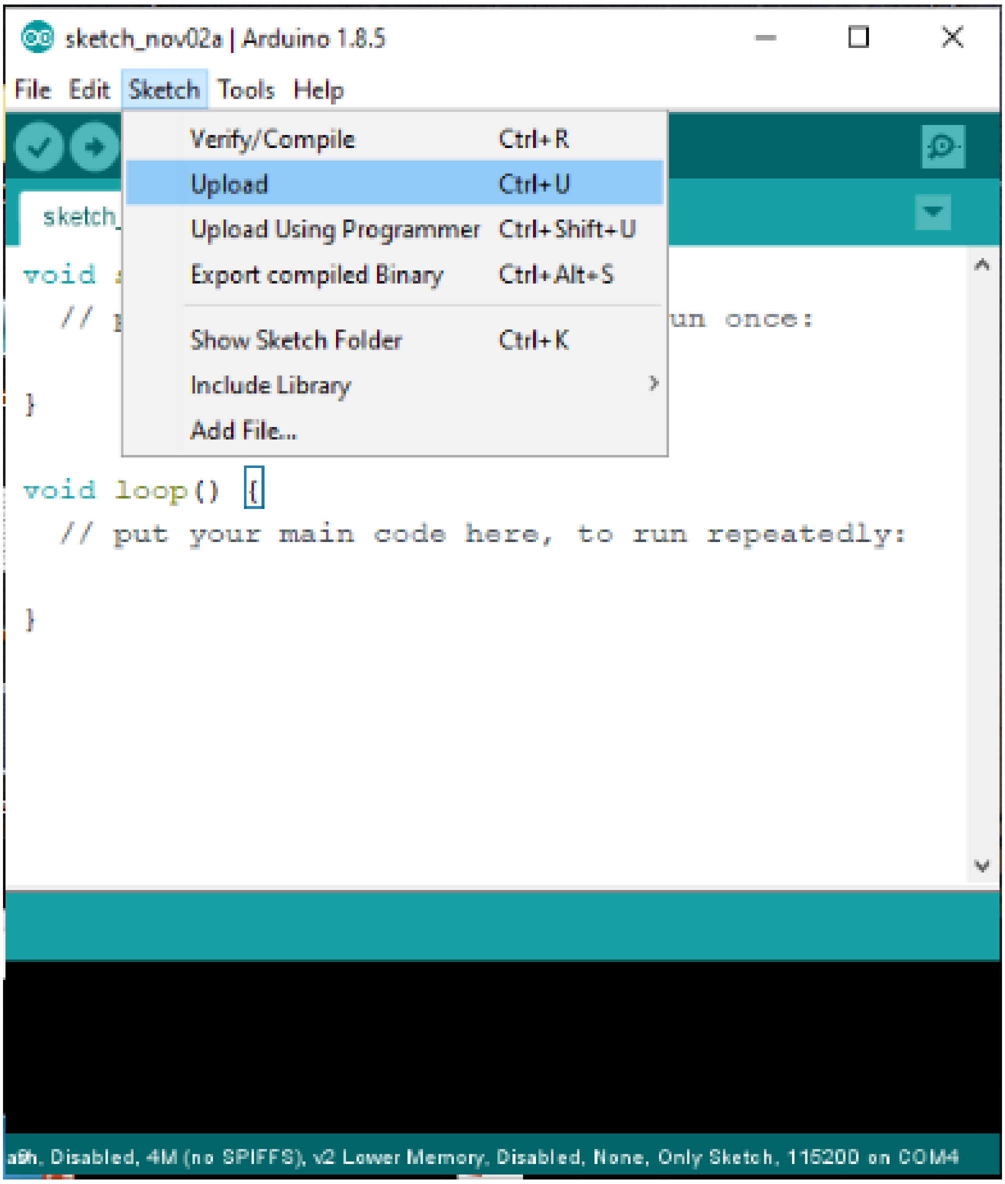

Fig 4.2.1.3 Uploading the compiled code to board.

4.2.2. ThingSpeak

ThingSpeak server is an open data platform and API for the Internet of Things that enables you to collect, store, analyse, visualize, and act on data from sensors. Your device or application can communicate with ThingSpeak using a Restful API, and you can either keep your data private, or make it public. In addition, use ThingSpeak to analyse and act on your data.ThingSpeak is loT analytics platform service that allows you to aggregate, visualize and analyze live data streams in the cloud. ThingSpeak is often used for prototyping and proof of concept loT systems that require analytics.ThingSpeak" is loT analytics service that allows you to aggregate, visualize, and analyze live data streams in the cloud. ThingSpeak provides instant visualizations of data posted by your devices to ThingSpeak. With the ability to execute MATLAB code in ThingSpeak, you can perform online analysis and process data as it comes in. ThingSpeak is often used for prototyping and proof-of-concept loT systems that require analytics.

With ThingSpeak, you can store and analyze data in the cloud without configuring web servers, and you can create sophisticated event-based email alerts that trigger based on data coming in from your connected devices.

Some of the key capabilities of ThingSpeak include the ability to:

- Easily configure devices to send data to ThingSpeak using popular IoT protocols.
- Visualize your sensor data in real-time.
- Aggregate data on-demand from third-party sources.
- Use the power of MATLAB to make sense of your IoT data.
- Run your IoT analytics automatically based on schedules or events.
- Prototype and build IoT systems without setting up servers or developing web software.
- Automatically act on your data and communicate using third-party services like Twilio or Twitter.

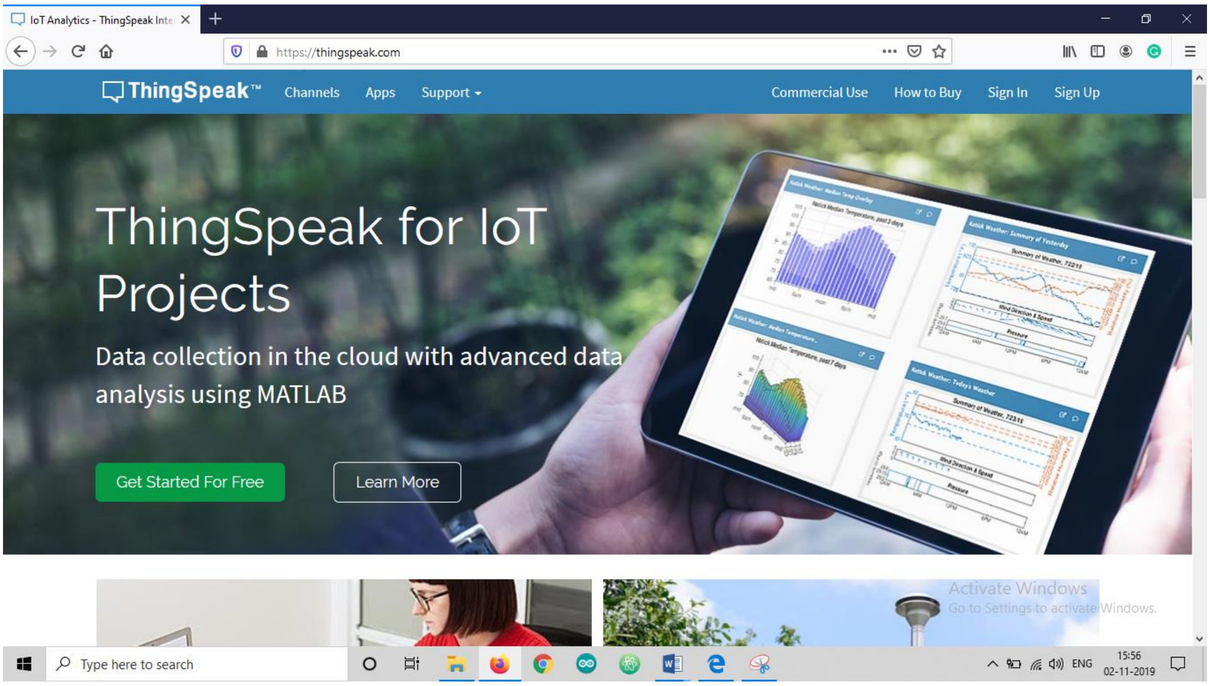

Fig 4.2.2.1 ThingSpeak home page.

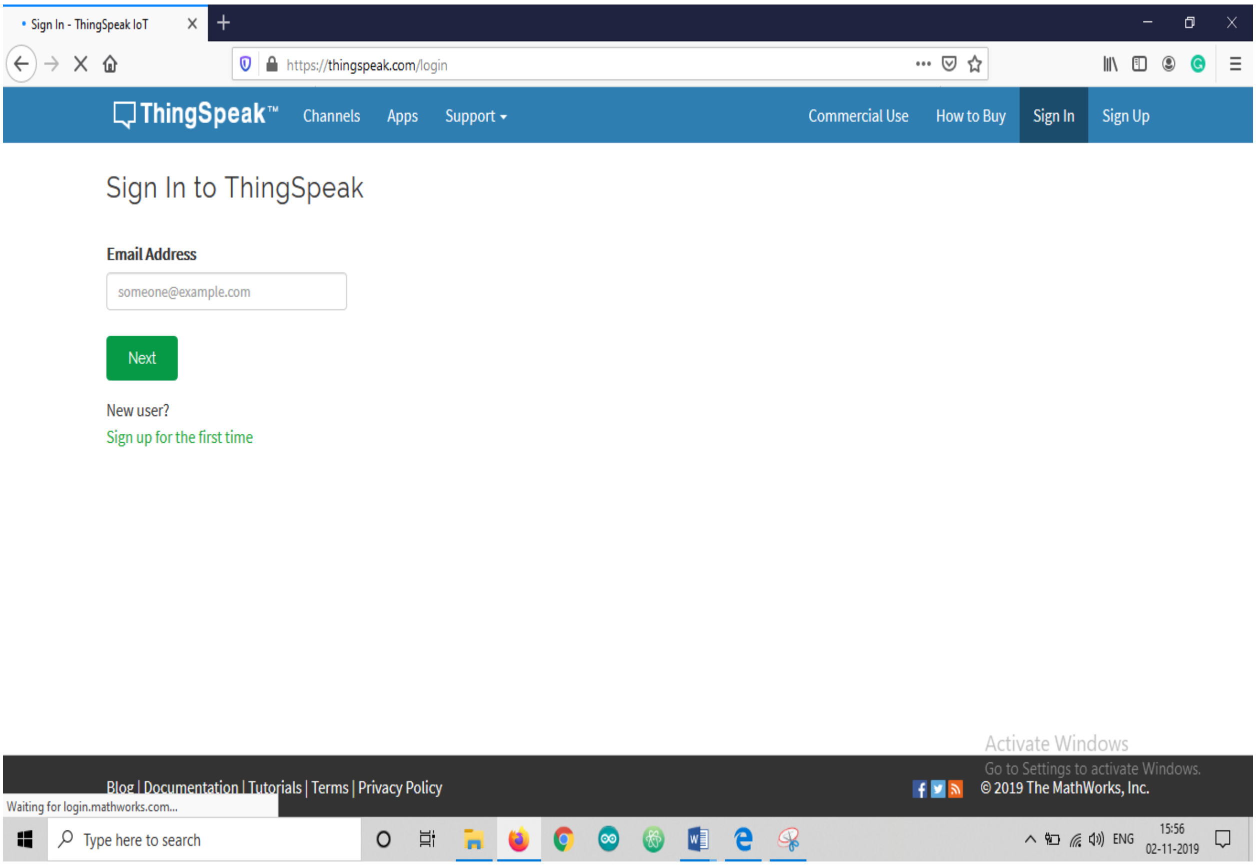

Fig 4.2.2.2 ThingSpeak login page.

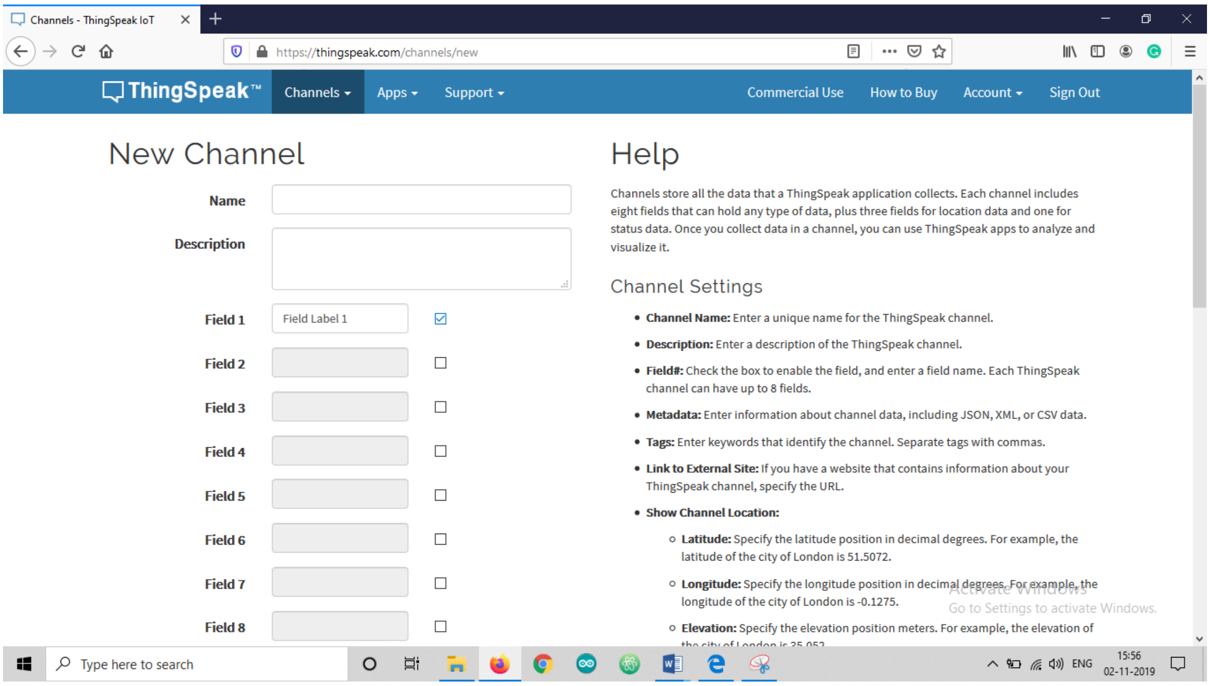

Fig 4.2.2.3 creating a new channel to save sensor data.

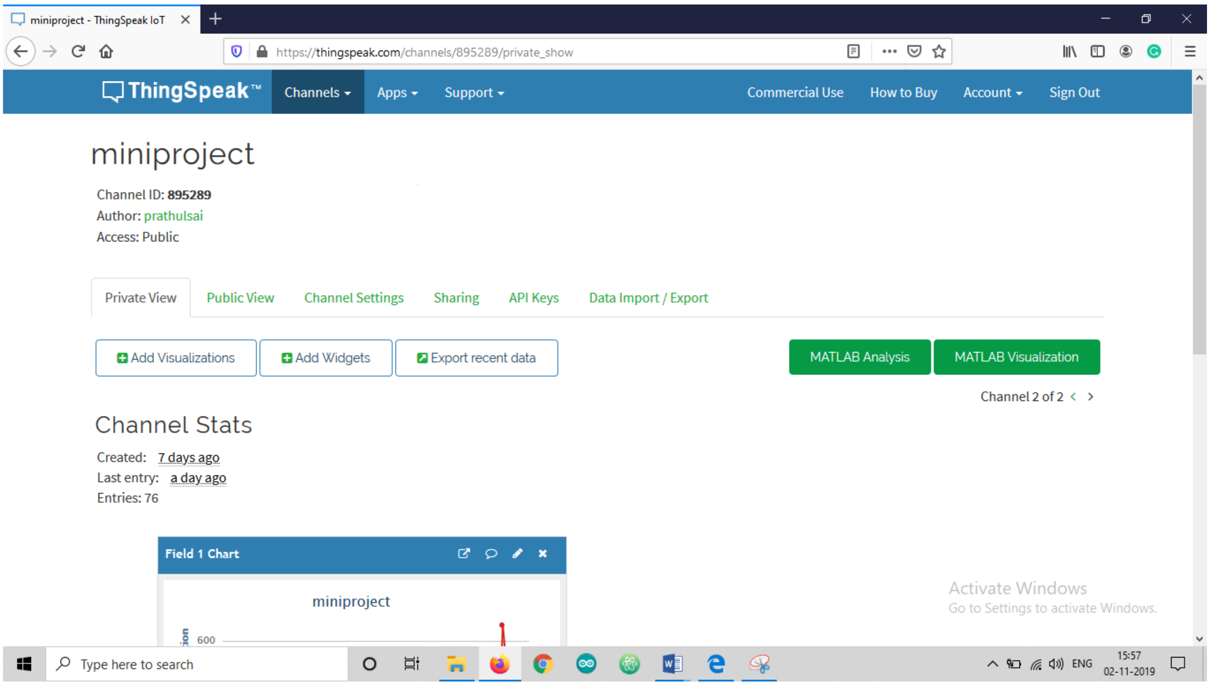

Fig 4.2.2.4 Channel page where we can see the visualization graphs for sensor data.

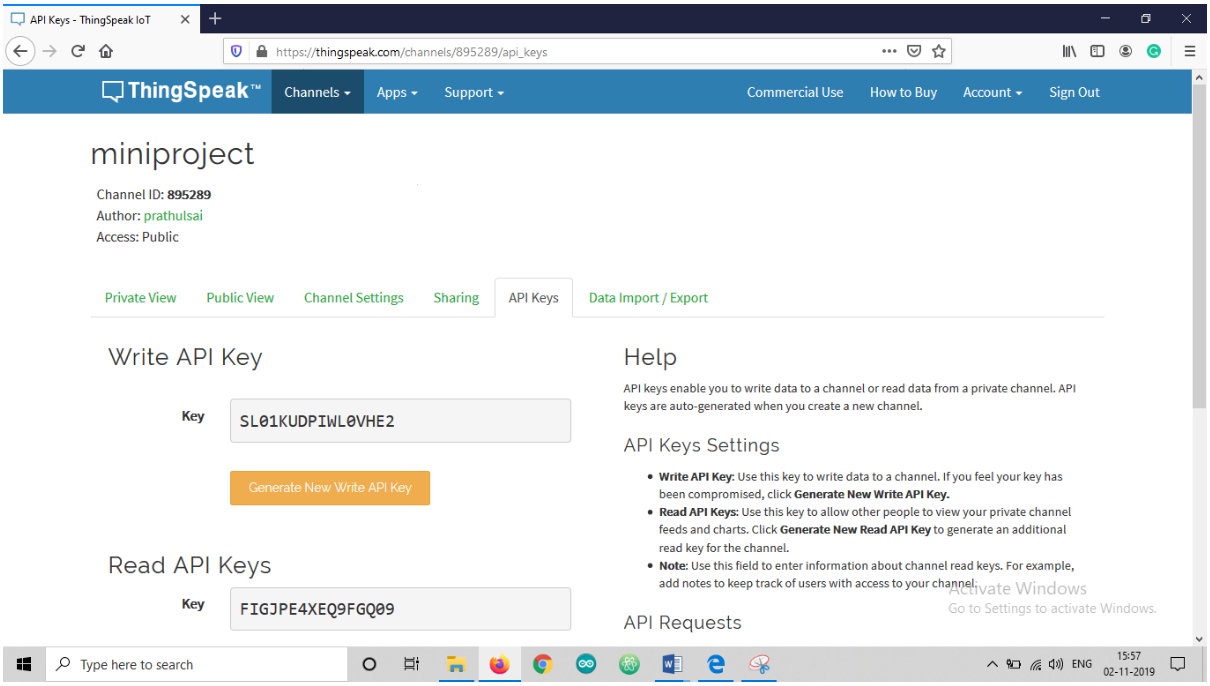

Fig 4.2.2.5 API keys to send or get the data from ThingSpeak server.

SYSTEM DESIGN

5. SYSTEM DESIGN

5.1 SYSTEM ARCHITECTURE

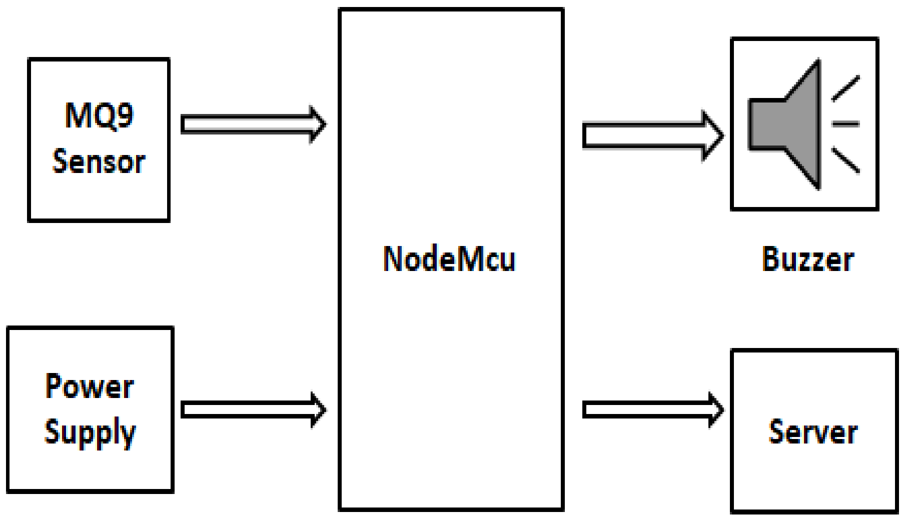

Fig 5.1 block diagram of gas leakage monitoring system.

The block diagram of the Gas Leakage Monitoring design illustrates the detailed system architecture of our project. The system uses the terminal harmful gas monitoring to collect the data of harmful gases at the disaster scene in real time, and then transmits the data to the Website.

The gas levels are sensed through the Mq9 gas sensor is used for sensing Carbon monoxide methane and butane respectively and sent to the Nodemcu (Node Microcontroller unit).The sensed analog signals are converted to digital through ADC (inbuilt in case of nodemcu). The sensed gas levels are sent to the website via Wi-Fi (inbuilt Wi-Fi module); if any one gas level exceeds the set point then the buzzer is generated immediately. A power supply is connected to the Nodemcu to provide the entire system with the required amount of power.

5.2 CICUIT DIAGRAM

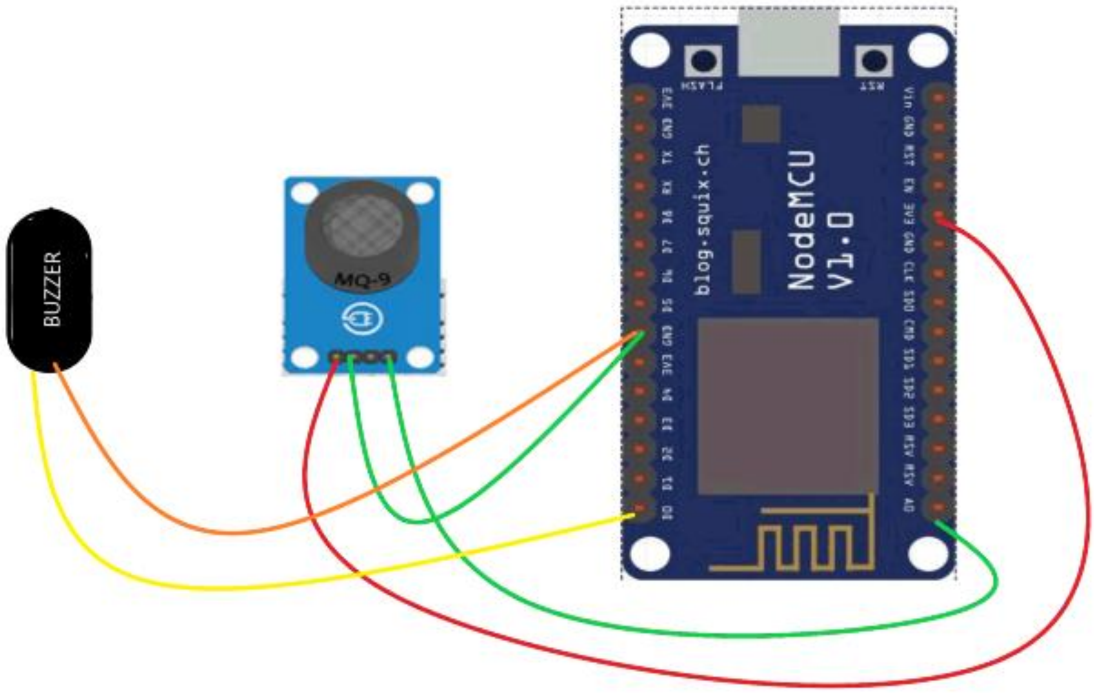

Fig 5.2.1 Circuit diagram of proposed system.

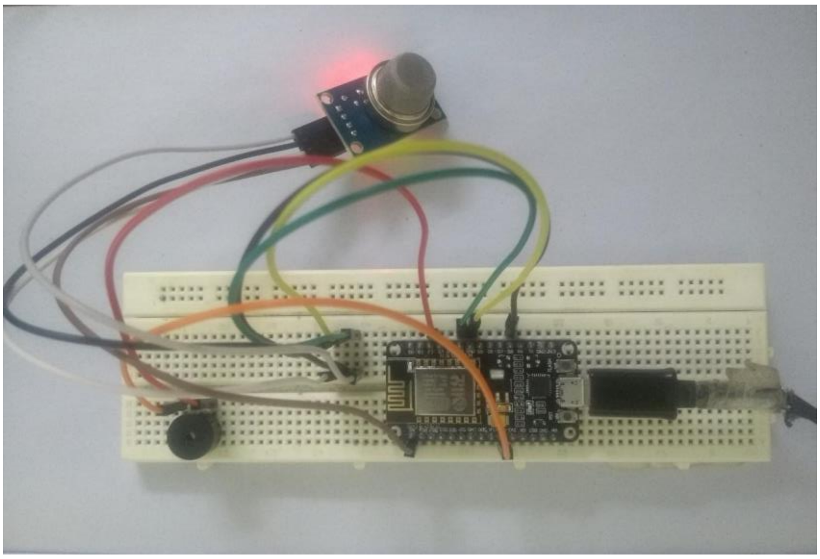

Fig 5.2.2 Picture of our system.

5.3 FLOW CHART

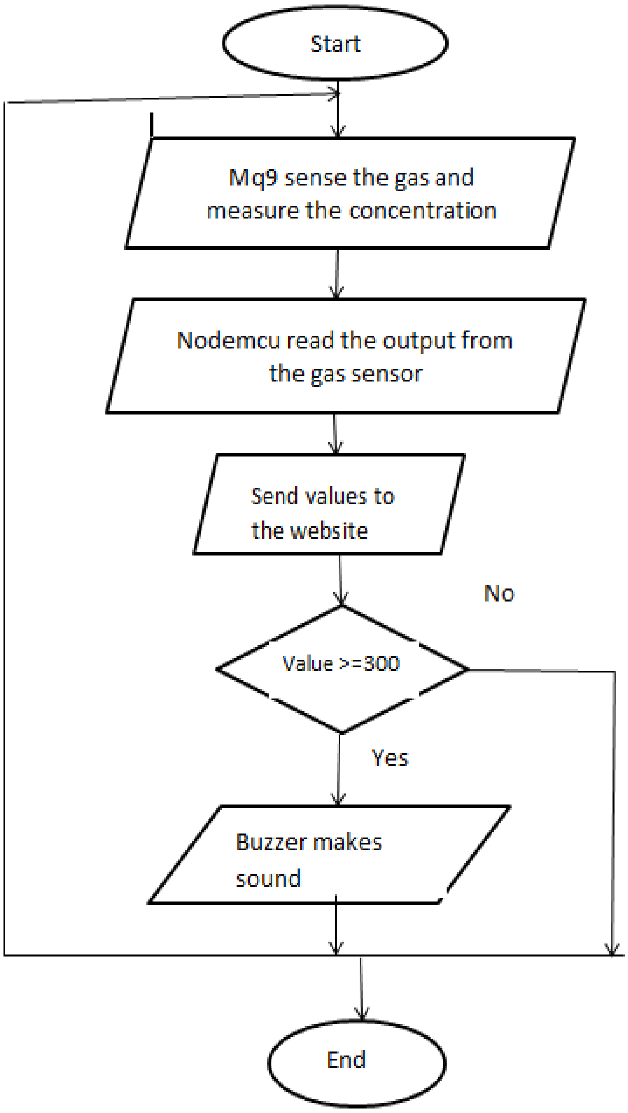

Fig 5.3.1 Flow chart

The chart shows that the whole procedure starts by sensing the harmful gases by MQ-9 sensor and send it to the Nodemcu on reading the sensor of the gas to decide and discover the situation of the environment. Once reading operation is completed, a decision making procedure starts to apply control action according to a given result. In this work, it is proposed to assign precaution based on decision results. The precaution takes place by considering that MQ – 9 Sensor stated that gas concentration is greater than 300 as an analog read. In this case the buzzer is turned ON, the air puller works to pull out the amount of the gas spread in the air, the system send those gas concentration values to the website and the user can view the concentration level of gases through the website.

CODING

6. CODING

6.1. SAMPLE CODE

```
#include <ThingSpeak.h>
#include<ESP8266WiFi.h>
const int buzzer =  4;
int outputpin= 0;
String apiKey = "SL01KUDPIWL0VHE2";
const char *ssid =  "Xender";     // replace with your wifi ssid and wpa2 key
const char *pass =  "zxcvbnm1";
//Thingspeak information
const char* server = "api.thingspeak.com";
WiFiClient client;
void setup()
{
       Serial.begin(9600);
       pinMode(buzzer, OUTPUT);
       delay(10);
       Serial.println("Connecting to ");
       Serial.println(ssid);
       WiFi.begin(ssid, pass);
        while (WiFi.status() != WL_CONNECTED)
       {
              delay(500);
              Serial.print(".");
       }
      Serial.println("");
      Serial.println("WiFi connected");
      ThingSpeak.begin(client);

}
void loop()
{
```

```
digitalRead(outputpin);
  float sensor_volt;
  float RS_air; //  Rs in clean air
  float R0;  // R0 in 1000 ppm LPG
  float sensorValue;
//Average
analogRead(outputpin);
    for(int x = 0 ; x < 100 ; x++)
  {    sensorValue = sensorValue + analogRead(outputpin);
  }
  sensorValue = sensorValue/100.0;
  sensor_volt = (sensorValue/1024)*3.3;
  RS_air = (5.0-sensor_volt)/sensor_volt; // Depend on RL on yor module
  R0 = RS_air/9.9; // According to MQ9 datasheet table
Serial.println(sensorValue);
    if (client.connect(server,80))   //   "184.106.153.149" or api.thingspeak.com
      {
         String postStr = apiKey;
         postStr +="&field1=";
         sensorValue=sensorValue;
         postStr += String(sensorValue);
         postStr += "\r\n\r\n";
         String str = "GET https://api.thingspeak.com/update?api_key=";
         str += postStr;
         client.println(str);
      }
    if(sensorValue>600){
    digitalWrite(buzzer,HIGH);
    }
  else{
    digitalWrite(buzzer,LOW);
    }
client.stop();
```

```
delay(10000);
}
```

Thinspeak Code:

Write a Channel Feed

- GET https://api.thingspeak.com/update?api_key=SL01KUDPIWL0VHE2&field1=0

Read a Channel Feed

- GET https://api.thingspeak.com/channels/895289/feeds.json?results=2

Read a Channel Field

- GET https://api.thingspeak.com/channels/895289/fields/1.json?results=2

Read Channel Status Updates

- GET https://api.thingspeak.com/channels/895289/status.json

6.2 CONNECTION OF PINS

- Connect the GND of MQ-9 sensor to GND of NodeMCU.
- Connect the Micro USB cable to NodeMCU and PC.
- Connect the Vcc of MQ-9 sensor to Vcc (3.3v) of NodeMCU.
- Connect the Digital output to D8 (any Digital pin of NodeMCU).
- Connect positive terminal of Buzzer to A0 pin.
- Connect negative terminal of Buzzer to GND of NodeMCU

TESTING AND VALIDATION

7. TESTING AND VALIDATION

7.1 TYPES OF IOT TESTING

The current challenges of IoT implementation is overwhelming, attributable to the highly complex and unique characteristics of IoT applications. This mandates different test scenarios for normal usage, peak points, and day-long simulations to ascertain if these applications ensure total performance and scalability of the IoT architecture.

7.1.1. Security Testing

Handling an onslaught of data is fundamental to IoT operations, and therefore, enterprises must conduct security testing to eliminate vulnerabilities and maintain the integrity of data. This includes examining various aspects of the system, including data protection, encryption/decryption, device identity authentication among more.

7.1.2. Performance Testing

This covers real-time and far more cumbersome aspects, such as load testing, streaming analytics, time-bound outputs, and timing analysis, to validate and ensure consistent performance of data reading, writing, and data retrieval.

7.1.3. Compatibility Testing

This testing assesses if the existing working combination of hardware, software, protocols, and operating systems fall on the IoT interoperability radar, and are compatible with the standards and specifications of conventional IoT industrial framework.

7.1.4. Functional Testing

This examines the qualitative and quantitative functional deliverability of deployed IoT applications in the actual conditions. Aspects, like network size, environment conditions, and topologies, are put to test.

7.1.5. Usability Testing

There are so many devices of different shape and form factors are used by the users. Moreover, the perception also varies from one user to other. That's why checking usability of the system is very important in IoT testing.

7.2 TEST STRATEGY AND APPROACHES

7.2.1 Test objectives

- Device should able to reconnect to server when it lost network connectivity.
- Only authorized user can able to access the data.
- Device able to setup connection on restart.

7.2.2 Features to be tested

- Verify that device able to send data when it have interrupted power supply.
- Verify that device tries to reconnect to server when server is down and back off to offline mode if fails to connect to server.

7.3 TEST CASES

7.3.1 TC-DAR1

Purpose	To test the "Always-on" connectivity mechanism for an IoT Device Application that very frequently sends data.
Requirement under test	DAR1
Entry Criteria	IoT Device Application is capable to send frequent data.
Test Procedure	1. IoT Device registers to network and data connection is successfully established. 2. Observe the Radio Resource Control (RRC) state, RRC connection Setup and Release in the Network for certain interval.
Exit Criteria (Pass Criteria)	IoT Device shall not make frequent RRC connection Setup and Release requests and it should be in one of the RRC state machines depending on data payload.

7.3.1 TC-DAR11g

Purpose	Check IoT Device Application behaviour in situations when network communication requests fail: • IoT Service Platform's IP address is unreachable.
Requirement under test	DAR11
Entry Criteria	1. The device is properly configured (APN etc.) 2. SIM Subscription is activate and is configured with the necessary services. 3. Block the IP address of the IoT Service Platform using by a firewall, or configure the device with an IP address (or port) which is not reachable.
Test Procedure	1. Connect the device to the network. 2. Operate the device normally and try to set up a data session. 3. Observe that the data connection shall fail.4.Observe the device behaviour for a period of time.
Exit Criteria	The Device should not retry a service request and "back off" according to the functionality defined within 'network friendly mode' or 'radio policy manager'.

Test Results

All the tests above mentioned are passed successfully. And the device behaving exactly as it is designed in situations when there is a problem with server or power.

SCREENSHOTS

8. SCREENSHOTS

8.1. Uploading code to NodeMCU board.

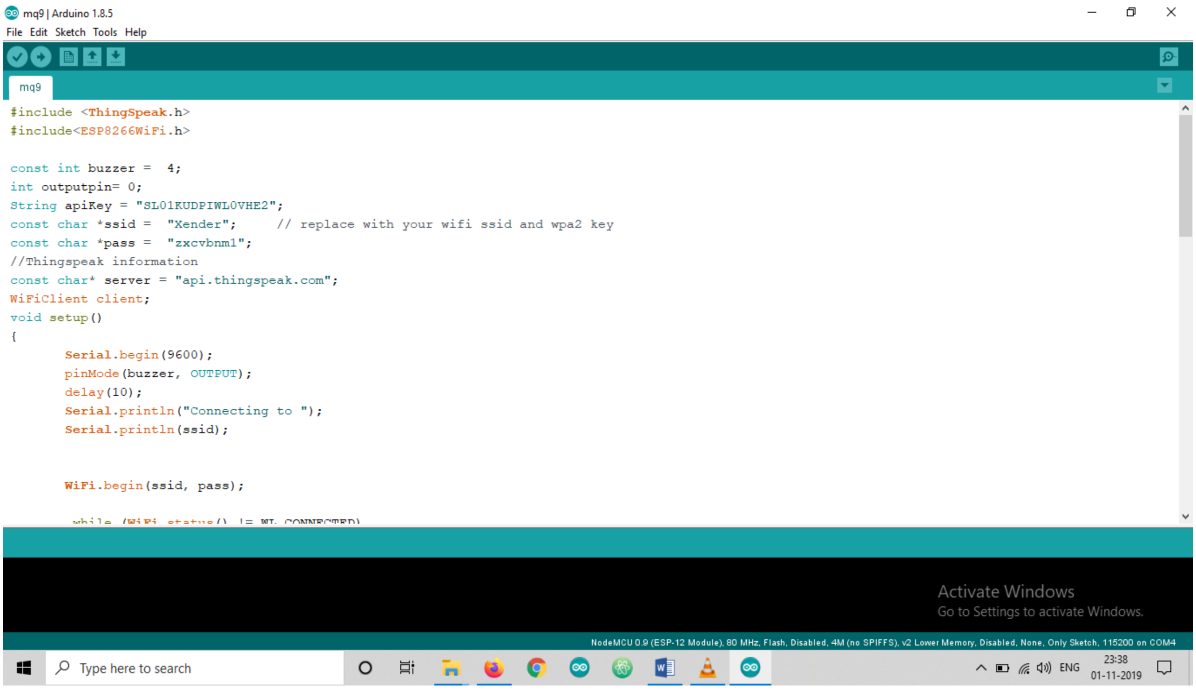

Fig 8.1 code uploading.

8.2. Managing thingspeak platform.

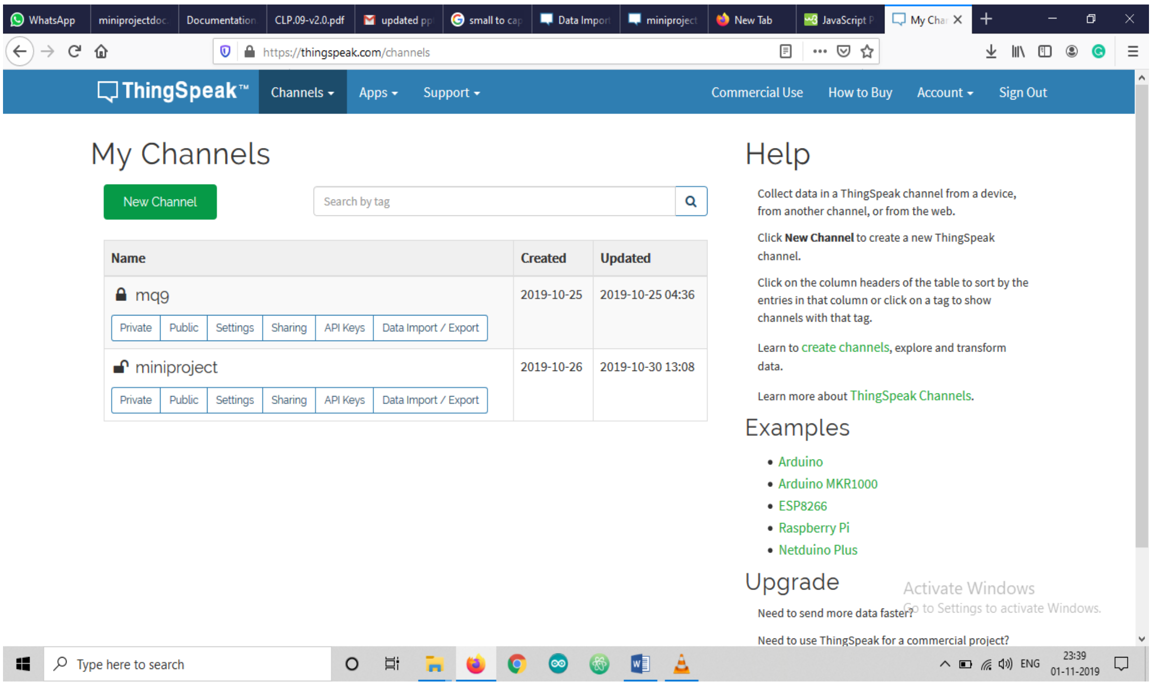

Fig 8.2 ThingSpeak

8.3. User Interface

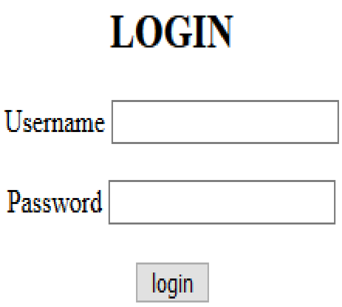

Fig 8.3.1 login page

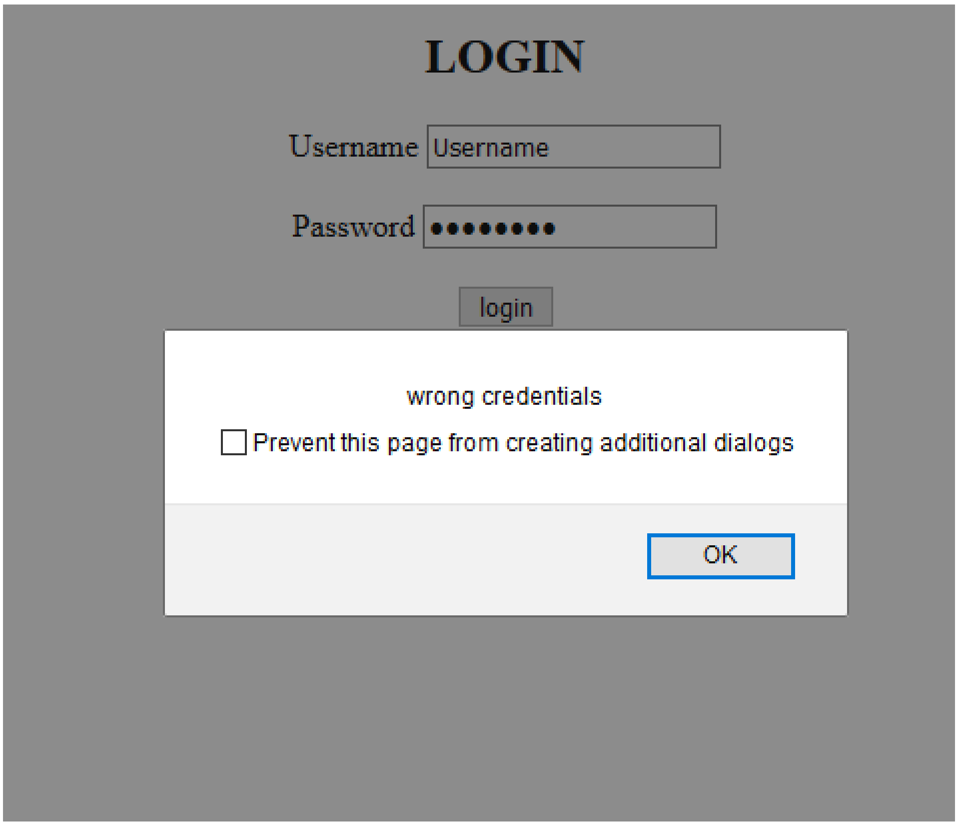

Fig 8.3.2 Login failed due to wrong password.

8.4 Serial monitor

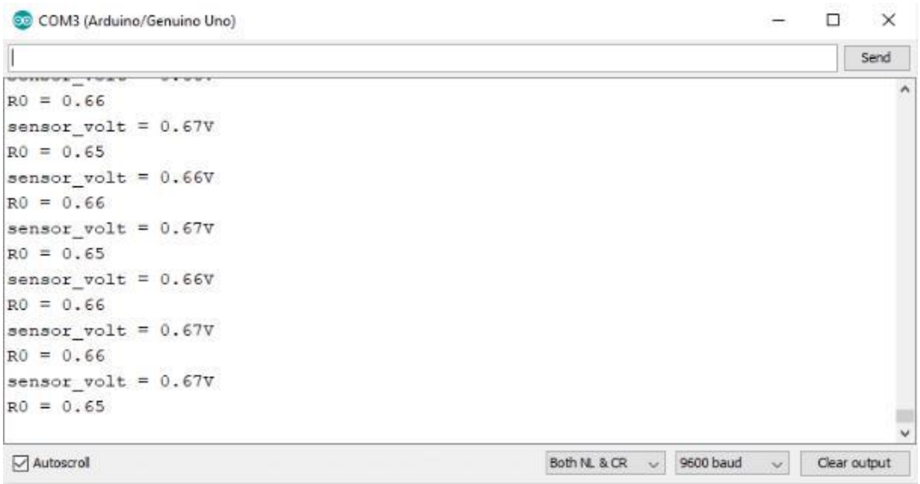

Fig 8.4.1 serial monitor when no gas is detected.

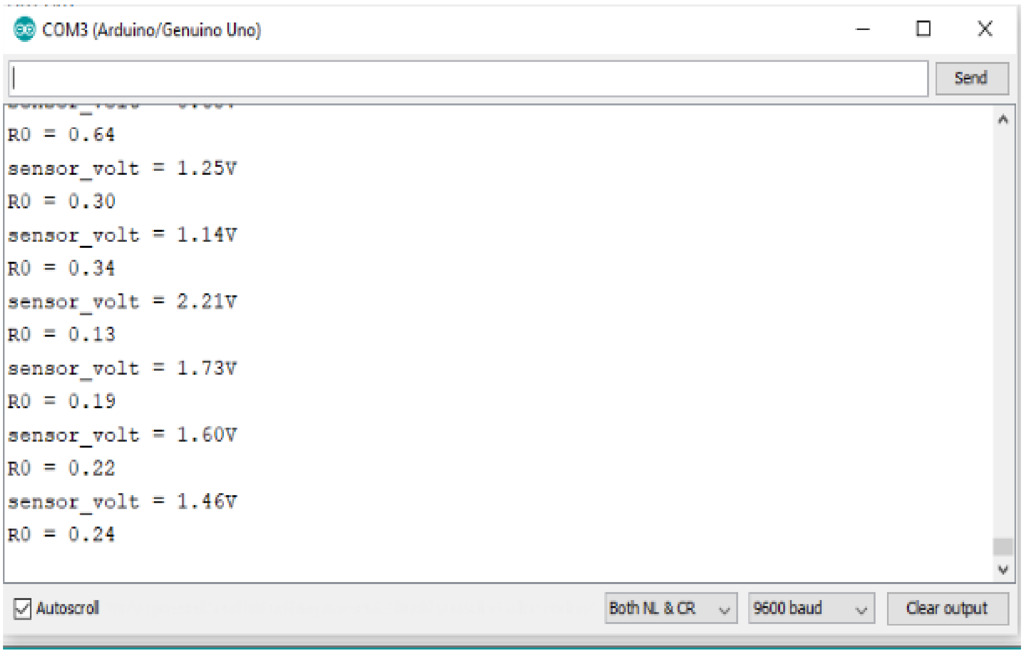

Fig 8.4.2 serial monitor when gas is detected.

8.5 GUI for gas concentration in air.

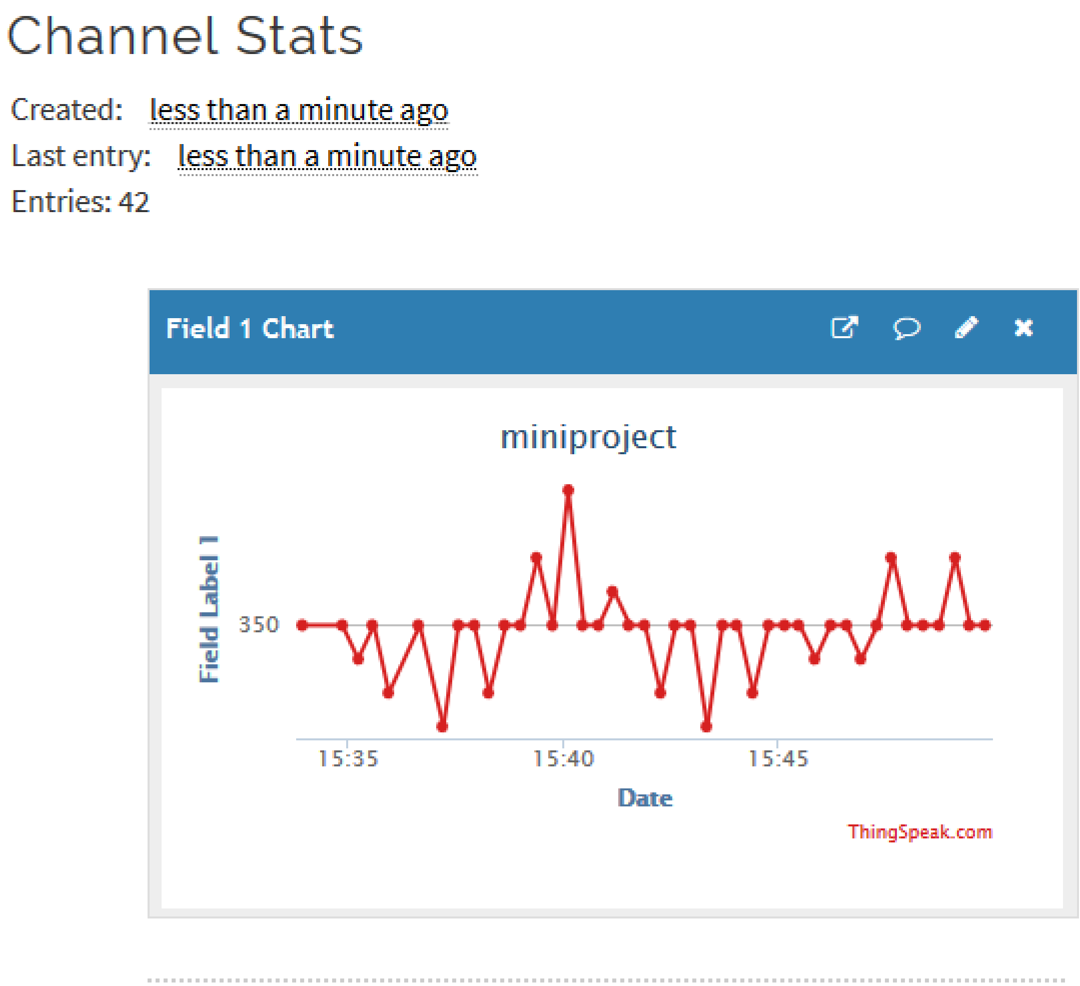

Fig 8.5.1 when the gas is not at a harmful concentration.

Channel Stats

Created: 6 days ago
Last entry: a day ago
Entries: 76

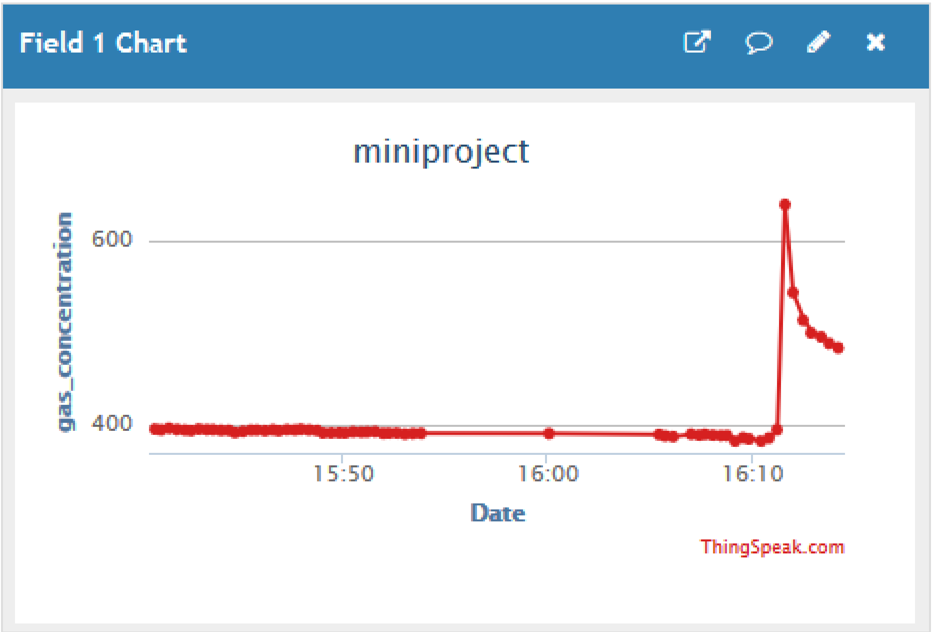

Fig 8.5.2 when gas is detected spike can be observed in the graph.

CONCLUSION

9. CONCLUSION

This system provides a control action during gas leakage by closing the solenoid valve. And it activates the alarm and also sends alert messages to the users within a short time. It is an economical system which can be installed in apartments, hotels and wherever it is needed. The cost of the proposed system is lesser than the commercially available detectors in the market. It can help us to prevent from accidents in all directions. There are some products available which are similar to this gas leakage detector but those are not cost efficient and doesn’t have any safety mechanisms.

Gas leakage leads to severe accidents resulting in material losses and Human injuries. Hence, leakage detection is essential to prevent accidents and to save human lives. This is a cost efficient way to detect harmful gases. And this device is portable so can be used in Industries, home, car. If this product becomes commercial, it will overcome all demerits of other similar products.

BIBLIOGRAPHY

10. BIBLIOGRAPHY

- Thingspeak.com
- researchgate.net/publication/329828278_Intelligent_Gas_Leakage_Detection_System_with_IoT_Using_ESP_8266_Module
- create.arduino.cc/projecthub/electropeak/how-to-calibrate-use-mq9-gas-sensor-w-arduino-e93cb1
- create.arduino.cc/projecthub/use-a-buzzer-module-piezo-speaker-using-nodemcu-89df45
- en.wikipedia.org/wiki/List_of_industrial_disasters
- wiki.seeedstudio.com/Grove-Gas_Sensor-MQ9
- www.indiaenvironmentportal.org.in/category/2543/thesaurus/gas-leakages/?page=3
- images.google.com
- wikipedia.org/wiki/NodeMCU
- arduino.stackexchange.com
- youtube.com