**/**

A PROJECT REPORT

Submitted by:

**CHAITHANYA P**

**LMC17MCA002**

to

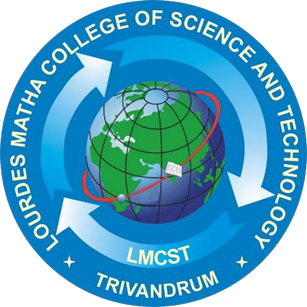
*The APJ Abdul Kalam Technological University*

*in partial fulfillment of the requirements for the award of the Degree*

*of*

*Master of Computer Applications*

**SEWAGE WORKERS SAFETY MONITORING SYSTEM**

****

**Department of Computer Applications**

LOURDES MATHA COLLEGE OF SCIENCE AND TECHNOLOGY

KUTTICHAL, THIRUVANANTHAPURAM 695574

MAY 2019

**SEWAGE WORKERS SAFETY MONITORING SYSTEM**

A PROJECT REPORT

Submitted by:

**CHAITHANYA P**

**LMC17MCA002**

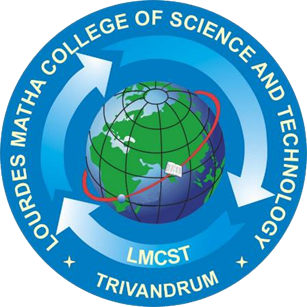
to

*The APJ Abdul Kalam Technological University*

*in partial fulfillment of the requirements for the award of the Degree*

*of*

*Master of Computer Applications*

****

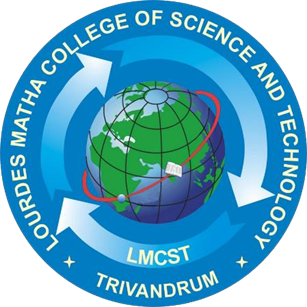
**Department of Computer Applications**

LOURDES MATHA COLLEGE OF SCIENCE AND TECHNOLOGY

KUTTICHAL, THIRUVANANTHAPURAM 695574

MAY 2019

**DEPARTMENT OF COMPUTER APPLICATIONS LOURDES MATHA COLLEGE OF SCIENCE AND TECHNOLOGY KUTTICHAL, THIRUVANANTHAPURAM**

****

## **CERTIFICATE**

This is to certify that the report entitled **‘Sewage Workers Safety System’** submitted by **Chaithanya P** to the APJ Abdul Kalam Technological University in partial fulfilment of the requirements for the award of the Degree of Master of Computer Applications is a Bonafede record of the project work carried out by her under my guidance and supervision.

Prof. Bismi K charleys

(Internal Supervisor)

Prof. Justin G Russel Prof.Selma Joseph

(Project Co-ordinator) (Head of the Dept.)

## **DECLARATION**

I undersigned hereby declare that the project report ‘Sewage Worker Safety Monitoring System’, submitted for partial fulfilment of the requirements for the award of degree of Master of Computer Applications of the APJ Abdul Kalam Technological University, Kerala is a Bonafede work done by me under supervision of Prof. Bismi K Charleys. This submission represents my ideas in my own words and, I have adequately and accurately cited and referenced the original sources. I also declare that I have adhered to ethics of academic honesty and integrity and have not misrepresented or fabricated any data or idea or fact or source in my submission. I understand that any violation of the above will be a cause for disciplinary action by the institute and/or the University.

Place: Trivandrum Signature

Date: 18/05/2020Chaithanya P

**CONTENTS**

ACKNOWLEDGEMENT i

ABSTRACT ii

Chapter 1. INTRODUCTION

* 1. General Background 1
  2. About the Project 2
  3. Objective and Scope 2

Chapter 2. LITERATURE SURVEY

2.1 Study of Similar Works 3

2.1.1 Existing System 5

2.1.2 Drawbacks of Existing System 5

2.1.3 Reports and Images Related to the Current System 6

Chapter 3. OVERALL DESCRIPTION

3.1 Proposed System 9

3.2 Features of Proposed System 9

3.3 Functions of Proposed System 10

3.4 Requirement Specification 10

3.5 Feasibility Analysis

3.5.1 Technical Feasibility 11

3.5.2 Economic Feasibility 12

3.5.3 Behaviour Feasibility 12

3.5.4 Operational Feasibility 13

Chapter 4. OPERATING ENVIRONMENT

4.1 Hardware Requirements 14

4.2 Software Requirements 14

4.3 Tools and Platforms 15

4.3.1 Arduino IDE 15

4.3.2 EmbeddedC 15

4.3.3 Windows 10 16

4.3.4 Blynk 16

4.3.5 ATMega328 17

4.3.6 DHT11 Sensor 18

4.3.7 Bluetooth Module 18

4.3.8 Gas Sensors (MQ Sensors) 19

4.3.9 Buzzer 19

4.3.10 NodeMCU 20

Chapter.5. DESIGN

5.1 System Design 21

5.1.1 Block Diagram 22

5.1.2 Project Activity Diagram 23

5.1.3 UML Use Case Diagram 24

5.1.4 Working Diagram of DHT11 26

5.1.5 Working Diagram of MQ7 27

5.1.6 Working Diagram of MQ9 28

5.1.7 Working Diagram of Buzzer 28

5.2 Input Output Design 29

5.3 Program Design 30

Chapter 6. FUNCTIONAL & NON-FUNCTIONAL REQUIREMENTS

6.1 Functional Requirements 39

6.2 Non-Functional Requirements 39

Chapter 7. TESTING

7.1 Testing Strategies 41

7.2 Unit Testing 42

7.3 Integration Testing 43

7.4 System Testing 44

7.5 User Acceptance Testing 45

7.6 Data Validation Testing 45

7.7 Output Testing 46

7.8 Testing Result 46

Chapter 8. Results and Discussion

8.1 Results 47

8.2 Screenshots 48

Chapter 9. Conclusion

9.1 System Implementation 49

9.2 Conclusion 49

9.3 Future Enhancement 50

BIBLIOGRAPHY

1. Books 51
2. Website 51

APPENDICES

1. SCRUM Board 52
2. List of Figures 54
3. List of Tables 55
4. Abbreviations and Notations 56
5. Coding

# **ACKNOWLEDGEMENT**

If words are considered as symbols of approval and tokens of acknowledgement, then let words play the heralding role of expressing my gratitude. First of all, I would like to thank God almighty for bestowing us with wisdom, courage and perseverance which had helped us to complete this project **Sewage Worker Safety Monitoring System**. This project has been a reality as a result of the help given by a large number of personalities. I am extremely thankful to **Rev.Dr. Tomy Joseph Padinjareveettil,** Director and **Prof. Mohanlal P,** Principal, Lourdes Matha College of Science and Technology Kuttichal for providing me with the best facilities and atmosphere which was necessary for the successful completion of this project. I would like to remember with gratitude **Prof. Selma Joseph**, Head of Department, Department of Computer Applications, Lourdes Matha College of Science and Technology Kuttichal, for the encouragement and guidance rendered. I express our profound and sincere gratitude to **Prof. Justin G Russel**, Department of Computer Applications, Lourdes Matha College of Science and Technology Kuttichal for the immense support and guidance provided. I express my sincere thanks to **Prof. Bismi K Charleys**, Department of Computer Applications, Lourdes Matha College of Science and Technology Kuttichal for his valuable guidance, support and advices that aided in the successful completion of my project. Finally, I wish to express my sincere gratitude to all our friends, who directly or indirectly contributed in this venture.

**ABSTRACT**

In India sewage cleaning from manholes and drains are a difficult and risky job for anyone. In 2013, Supreme Court of India gave a decision that all such jobs would be done by machines and wherever human intervention is required, proper safety equipment should be provided to the workers. However, the municipalities give contract to contractors who hire local people to clean the sewage.

A large number of sanitation workers die every year due to erratic and lack of facilities available, and harmful toxic gases released while cleaning the sewage. Real time health and safety monitoring system for such workers will prove helpful. In the existing system there are no efficient real time health and safety monitoring system. So many researchers proposed so many ideas, but they are not reliable.

The proposed system is an IoT based project. It will mainly be focusing on the safety of the sewage workers. Here the system will be using a helmet which consist of various sensors to detect hazardous gases inside the sewage. The readings from various sensors will send to the mobile of the person who stand outside the sewage through a mobile application. Through this we can save the life of the worker from the deadly poisoning gases inside a sewage. Also, the helmet has an emergency button which will help the worker to notify his helper when he feels suffocated or uncomfortable.

**CHAPTER 1**

**INTRODUCTION**

* 1. **GENERAL BACKGROUND**

With the rapid development of urbanization, urban hardening area is also increasing. At the same time, global warming, water cycle changes, atmospheric circulation anomalies and other factors caused by complex situation which the abnormal weather and climate events continue to occur. Heavy precipitation and other disastrous weather frequency, intensity has increased, the trend of strengthening, the management of sewers is more and more demanding. However, the current sewer monitoring system is only for a certain aspect of the sewer to detect, the most common is to monitor the sewer toxic gas situation. There is no comprehensive sewer monitoring program, which gives the sewer management a great deal of inconvenience, which makes it impossible for departments involved in the operation of the sewer system to be able to grasp and monitor in real time, while the data systems in the management department are inconsistent, the data cannot be organic integration, in the event of danger cannot be timely warning and targeted, effective exclusion. So, it is particularly important to place sensors in areas where people are not easy to reach. Therefore, the establishment of an efficient sewer monitoring system is not only an important part of urban environmental monitoring, but also an important component of safety early warning of facilities.

In the future development of technology IoT has a profound influence. In addition, with the development low power embedded technology, sensor technology is widely used. Constructing the wireless sensor network of sewer can increase the monitoring area, reduce the artificial blind spot of the sewer and improve the accuracy of monitoring. In India sewage cleaning from manholes and drains are a difficult and risky job for anyone. In 2013, Supreme Court of India gave a decision that all such jobs would be done by machines and wherever human intervention is required, proper safety equipment should be provided to the workers. However, the municipalities give contract to contractors who hire local people to clean the sewage.

**ABOUT THE PROJECT**

In this project we are mainly focusing on the safety of sewage workers. Here we are using a helmet which consist of various sensors to detect hazardous gas and other contents in the sewage. Also, we have sensor to monitor the body temperature of the sensor. The data from these sensors will be send the mobile of the team outside the sewage. In their mobile they can see the data send by the sensors and they will get alert if any abnormality is found. Also, the helmet has an emergency button. If the worker feels any hesitations or any discomfort, he can press the emergency button and an alert will be send to the mobile of the co-worker.

Here we are constructing an advanced sewage workers safety monitoring system. For this we are using various sensors. These sensors will be embedded on a helmet and the sewers can wear this helmet. This helmet has a carbon monoxide sensor, hydrogen sulphide sensor, methane sensor, and it has a temperature sensor to detect the temperature and humidity.

* 1. **OBJECTIVE AND SCOPE**

**SEWAGE WORKERS SAFETY MONITORING SYSTEM BASED ON IoT is** a project which take care of the sewage workers, those people who work for the society. This smart safety system will help to monitor the safety measures of a sewage workers who works in the sewages to keeps our city clean.

The proposed system will mainly be focusing on the safety of the sewage workers. Here the system will be using a helmet which consist of various sensors to detect hazardous gases inside the sewage. The readings from various sensors will send to the mobile of the person who stand outside the sewage through Blynk. Through this we can save the life of the worker from the deadly poisoning gases inside a sewage. Also, the helmet has an emergency button which will help the worker to notify his helper when he feels suffocated or uncomfortable.

**CHAPTER 2**

**LITERATURE SURVEY**

**2.1 STUDY OF SIMILAR WORK**

**Underground Drainage Monitoring System**

The underground drainage system is an important component of urban infrastructure. Most management on underground drainage is manual therefore it is not efficient to have clean and working underground system also in such big cities, it is difficult for the government personnel to locate the exact manhole which is facing the problem. Therefore, it is essential to develop a system which can handle underground drainage without human intervention. Underground Drainage involves sewerage system, gas pipeline network, water pipeline, and manholes. This project describes various functions used for maintenance and monitoring of underground drainage system. It provides a system which is able to monitor the water level, atmospheric temperature, water flow and toxic gasses. If drainage system gets blocked and water overflows it can be identified by the sensor system. And that sensor sends information via the transmitter which is located in that area to the corresponding managing station.

As most of the cities in India have adopted underground drainage system, it is very important that this system should work in a proper manner to keep the city clean, safe and healthy. If they fail to maintain the drainage system the pure water may get contaminated with drainage water and can spread infectious diseases. So different kind of work has been done to detect, maintain and manage these underground systems. Also, leaks and bursts are unavoidable aspects of water distribution system management and can account for significant water loss within a distribution network if left undetected for long period. This project represents the implementation and design functions for monitoring and managing underground drainage system with different approaches. It also gives a description of water wise system and detection method to detect leakage defects in sewer pipeline. Also, some part of condition rating model for underground Infrastructure Sustainable Water Mains and Intelligent system for underground pipeline assessment, rehabilitation and management are explained.

**Automated IoT for Underground Drainage Monitoring System**

The Internet of Things (IoT) consists of real-life objects, communication devices attached to sensor networks in order to provide communication and automated actions between real world and information world. IoT came into existence because, without human interaction, computers were able to access data from objects and devices, but it was aimed at, to overcome the limiting factors of human entered data, and to achieve cost, accuracy and generality factors. Sensor Network is a key enabler for IoT paradigm. It represents the implementation and design function of an Underground Drainage and Manhole Monitoring System (UDMS) for IoT applications. The vital considerations of this design are low cost, low maintenance, fast deployment, and a high number of sensors, long life-time and high quality of service. The proposed model provides a system for monitoring the water level and atmospheric temperature and pressure inside a manhole and to check whether a manhole lid is open. It also monitors underground installed electric power lines. In real time, UDMS can remotely monitor current states of the manholes.

**Smart Real Time Drainage Monitoring System Using IoT**

Drainage system monitoring plays an important role to keep city clean. In fact, not all areas have drainage monitoring team. It leads to irregular monitoring of the drainage condition. The irregular monitoring leads to the blocking of the drainage that imply to the salutation which trigger flood. Manual monitoring is also incompetent. It requires a professional but they can only monitor very finite and maintain low accuracy. Also, sometimes due to lack of knowledge the worker may meets to an accident as they have no idea that how will be the conditions in those manholes. This paper represents the application and design function of a smart and real-time Drainage and Manhole Monitoring System with the help of Internet of Things. The manholes present in the drainage will have a module which is having microcontroller interfaced with gas sensor, level indicator, NRF.

The system will monitor if the blockage is occurred in between two manholes and also it will sense the rise in amount of various gases which are harmful to the human beings, and also a system of monitoring the water level then it will trigger an alarm and will provide that information to the health departments from which the particular action will be taken. The system will able to monitor all these things in real-time scenario which will allow us to take proper actions of the particular problem in drainage system.

**2.1.1 Existing System**

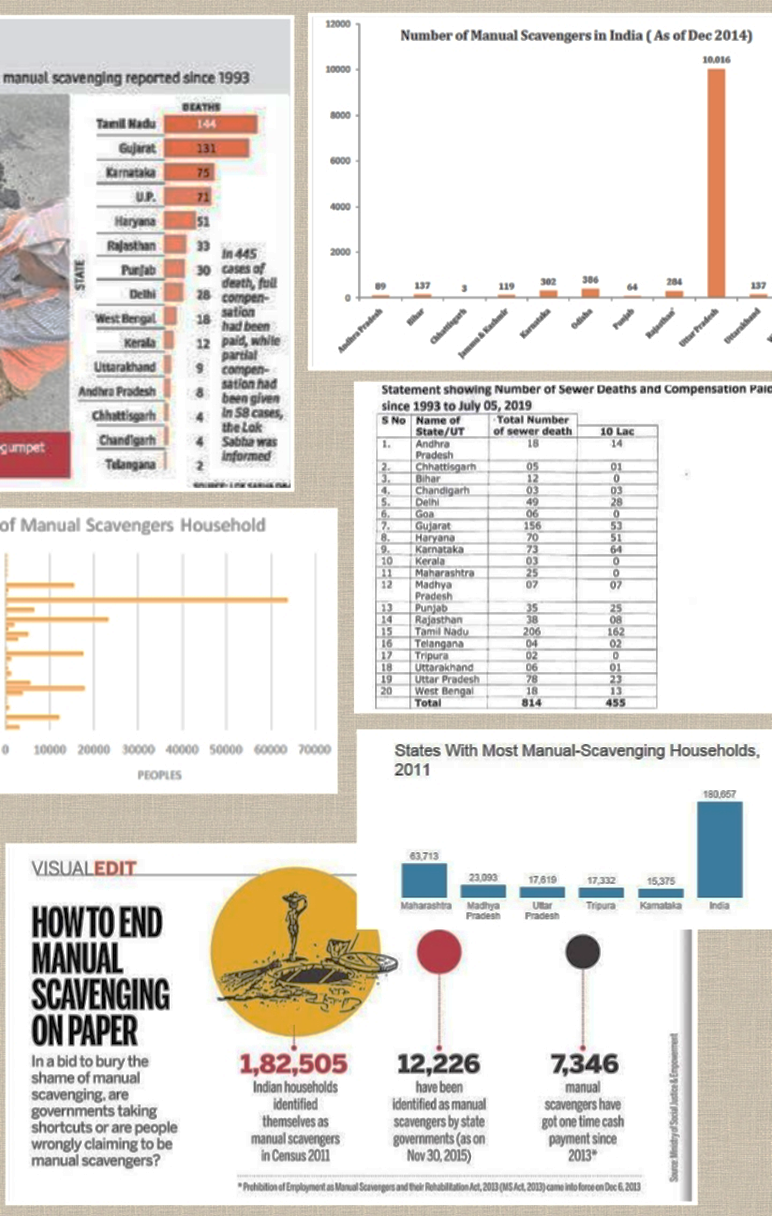
A large number of sanitation workers die every year due to erratic and lack of facilities available, and harmful toxic gases released while cleaning the sewage. Real time health and safety monitoring system for such workers will prove helpful. In the existing system there are no efficient real time health and safety monitoring system. So many researchers proposed so many ideas, but they are not reliable. In the existing system they use very limited facilities like a gas sensor and a heartbeat sensor. Under the drainage the possibilities of current monitoring system implementation are quite difficult. There is no unified equipment that can provide all the services in one place. Today’s drainage system is not computerized due to which it is hard to know if blockage is occurring in particular location. Also, sometimes due the waste in those drainage lines can produce various gases like methane (CH4), carbon monoxide (CO), etc which are harmful and can cause serious problem if inhaled by humans in large amount and these problems are generally faced by the drainage workers due to which death can occurs. Also, we don’t get early alerts of the blockage or rise in amount of those gases or the increase in water level.

**2.1.2 Drawbacks of Existing System**

* Lack of safe and security
* Less Efficient
* Implementation Issues
* High Expenses
* Difficulty in Maintenance

**2.1.3 Reports and Images Related to the Current sewage System (INDIA)**







**CHAPTER 3**

**OVERALL DESCRIPTION**

**3.1 PROPOSED SYSTEM**

In our proposed system we are introducing a new system which can solve all the problems in the existing system. Here we are using a helmet which can be wear by the sewage workers. This helmet consists of various sensors, which will measure the various gas levels, oxygen level, and temperature and humidity in the sewage. Here we are providing a facility for alert the co-workers when the things will go worse. Also, our safety smart helmet has an emergency button that can be used by the sewage worker when he feels any suffocation. Here we are unifying all the services and facilities that we can provide, in one equipment, that is, through a helmet, which will work efficiently to save the life of sewage workers. The expenditure our proposed system is very cost effective.

**3.2 FEATURES OF PROPOSED SYSTEM**

* Cost Effective
* Efficient
* Can Implement Easily
* Safe and Secure
* Easy to maintain
* Can easily add advanced technologies
* User Friendly
* Light Weight
* Ease of Use

**3.3 FUNCTIONS OF PROPOSED SYSTEM**

* **Portability:** Our smart sewage safety helmet can be wearing easily without any hesitation and it can be kept wherever we need.
* **Easy Maintenance:** It is very easy to keep the maintenance of our helmet.
* **Low Cost:** The rate of sensors used in this helmet is affordable and the price of helmet is satisfiable.
* **Adaptive:** This proposed system can be easily adapted to any new technology.
* The proposed system will take care of gas levels and other parameters and will give alert to the mobile of the co-worker if any problem occurs.

**3.4 REQUIREMENT SPECIFICATION**

Requirements specification simply means, “Figuring out what is to be made before making it”. It determines what people need before starting to develop a product for them. Requirement definition is the is the activity of translating the information gathered in to a document that defines a set of requirements. These should reflect what consumer wants.

The requirements (Software and Hardware) of SEWAGE WORKERS SAFETY MONITORING SYSTEM BASED ON IoT are:

* **Helmet, Mobile Phone**
* **Gas Sensors:** To detect various gases like Methane, Carbon Monoxide etc.
* **NodeMCU:** NodeMCU is an open-source firmware and development kit that helps you to prototype or build IoT products.
* **Temperature and Humidity Sensor:** It measures the heat energy or even coldness produced by an object or system.
* **EmbeddedC:** Programming language required for microcontroller-based applications.
* **Arduino IDE:** Used for Arduino board programming.
* **Blynk:** Used to send the readings from the sensors to the mobile.
* **ATMEGA 328:** It is an 8-bit microcontroller chip that are used with the popular Arduino Duemilanove boards.

**3.5 FEASIBILITY ANALYSIS**

The initial investigation points to be question whether the project is feasible. The feasibility study concerns with the considerations made to verify whether the system fit to be developed in all terms. Once the idea to develop hardware is put forward, the question that rises first will pertain to be the feasibility aspects. Feasibility study is a test of proposed system regarding its efficiency, impact on the organization, ability to meet the need of users and effective use of resources.

Thus, when a new project is proposed, it normally goes through a feasibility study before it is approved for development. A feasibility study is conducted to select the best system that meets the system performance requirements. This entitles an identification description, an evaluation of candidate system and the selection of the best system for the job.

During system analysis, a feasibility study of the proposed system was carried out to see whether it was beneficial to the organization. Three key considerations that are involved in the feasibility study. They are,

* Technical Feasibility
* Economic Feasibility
* Behaviour Feasibility
* Operational Feasibility

**3.5.1 Technical Feasibility**

Technical Feasibility centres on the existing computer system hardware, software, etc. and to some extent how it can support the proposed addition. This involves financial considerations to accommodate technical enhancements. Technical support is also a reason for the success of the project. The techniques needed for the system should be available and it must be reasonable to use. Technical Feasibility is mainly concerned with the study of function, performance, and constraints that may affect the ability to achieve the system. By conducting an efficient technical feasibility, we need to ensure that the project works to solve the existing problem area.

Since the project is designed using EmbeddedC as programming language. It is very efficient and user friendly. Here we are using MCU 328 micro controller to feed the program and the readings from various sensors are send to the mobile phones using a data monitoring app which easy to use and maintain.

**3.5.2 Economic Feasibility**

The role of interface design is to reconcile the differences that prevail among the software engineer’s design model, the designed system meets the end user requirement with economical way at minimal cost within the affordable price by encouraging more of proposed system. Economic feasibility is concerned with comparing the development cost with the income/benefit derived from the developed system. In this we need to derive how this project will help the management to take effective decisions.

Economic Feasibility is mainly concerned with the cost incurred in the implementation of the project. Since this project is developed using EmbeddedC which is more commonly available and even the cost involved in the installation process is not high.

This project has various sensors which is available at low cost in the market. The helmet used in this project is also low cost. Also, the price of micro controller MCU 328 is affordable. The installation cost of Data monitoring app, Arduino IDE are also free.

The system once developed must be used efficiently. Otherwise there is no meaning for developing the system. For this a careful study of the existing system and its drawbacks are needed. The user should be able to distinguish the existing one and Proposed one, so that one must be able to appreciate the characteristics of the proposed System, the manual one is not highly reliable and also is considerably fast. The proposed system is efficient, reliable and also quickly responding.

**3.5.3 Behaviour Feasibility**

Proposed projects are beneficial only if they can be changed in to information system that will meet operation requirement of the organization. People are inherently resistant to change and computers have been known to facilitate changes. An estimate should be made of how strong reaction the user staff is likely to have towards the development of a computerized system.

In this project the system is only used by the worker and the data will send to his co-worker only who stand outside the drainage. There is no data storage is needed. The co-worker will get the live readings from the sewage.

**3.5.3 Operational Feasibility**

There is not much difficulty in implementing the system. The proposed system is effective, user friendly and functionally efficient. The user of the system must be unaware of the internal working of the system so that the user will not face any problems running the system. In our system we are using MCU 328 microcontroller and various sensors. We can extend or add any features to the system easily.

The sensors will give accurate value according to the programs that are fed in to the micro controller. The user can easily use the system and app. There is no need to worry about the internal procedures of the system.

**CHAPTER 4**

**OPERATING ENVIRONMENT**

**4.1 HARDWARE REQUIREMENTS**

Processor : Intel Pentium4, Arduino UNO

RAM : 1GB or above

Hard Disk : 250GB or above

Display Size : 14” Colour Monitor or above

Screen Resolution : 1024\*768 pixels or above

Keyboard : PC/AT Enhanced Type

Mouse : Logitech PS/2 Port Mouse

Micro Controller : NodeMCU

Temperature Sensor : DHT11

Gas Sensors : MQ7, MQ9

Bluetooth Module

Helmet

**4.2 SOFTWARE REQUIREMENTS**

Operating System : Windows7 or higher

Front End : Blynk

Back End : EmbeddedC

IDE : Arduino IDE

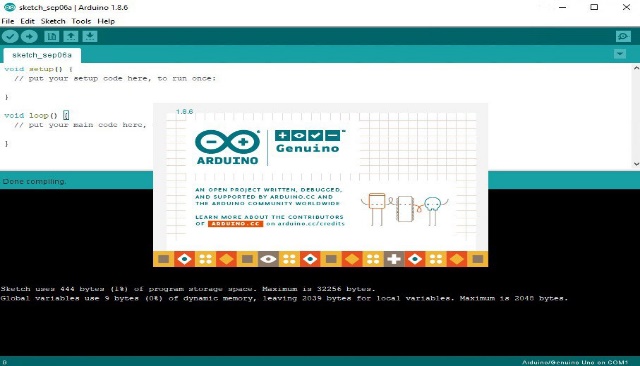
Data Monitoring App : Blynk

**4.3 TOOLS AND PLATFORMS**

**4.3.1 Arduino IDE**

Arduino is an open-source electronics platform based on easy-to-use hardware and software. [Arduino boards](https://www.arduino.cc/en/Main/Products) are able to read inputs - light on a sensor, a finger on a button, or a Twitter message - and turn it into an output - activating a motor, turning on an LED, publishing something online. You can tell your board what to do by sending a set of instructions to the microcontroller on the board. To do so you use the [Arduino programming language](https://www.arduino.cc/en/Reference/HomePage) (based on [Wiring](http://wiring.org.co/)), and [the Arduino Software (IDE)](https://www.arduino.cc/en/Main/Software), based on [Processing](https://processing.org/).

Over the years Arduino has been the brain of thousands of projects, from everyday objects to complex scientific instruments. A worldwide community of makers - students, hobbyists, artists, programmers, and professionals - has gathered around this open-source platform, their contributions have added up to an incredible amount of [accessible knowledge](http://forum.arduino.cc/) that can be of great help to novices and experts alike.



**4.3.2 EmbeddedC**

Embedded C Programming is the soul of the processor functioning inside each and every [embedded system](https://www.elprocus.com/ieee-projects-on-embedded-systems/) we come across in our daily life, such as mobile phone, washing machine, and digital camera. Each processor is associated with an embedded software. The first and foremost thing is the embedded software that decides functioning of the embedded system. Embedded C language is most frequently used to [program the microcontroller](https://www.elprocus.com/how-to-program-the-microcontroller/).

Earlier, many embedded applications were developed using assembly level programming. However, they did not provide portability. This disadvantage was overcome by the advent of various high-level languages like C, Pascal, and COBOL. However, it was the C language that got extensive acceptance for embedded systems. The C code written is more reliable, scalable, and portable; and in fact, much easier to understand.

**4.3.3 Windows 10**

Windows 10 is a major version of the Microsoft [Windows](https://techterms.com/definition/windows) operating system that was released on July 29, 2015. It is built on the Windows NT [kernel](https://techterms.com/definition/kernel) and follows [Windows 8](https://techterms.com/definition/windows_8). Part of the reason Microsoft decided to name the 2015 release "Windows 10" (and skipped "Windows 9") is because the [operating system](https://techterms.com/definition/operating_system) is designed to be a new direction for Microsoft. One of the primary aims of Windows 10 is to unify the Windows experience across multiple devices, such [desktop computers](https://techterms.com/definition/desktop_computer), [tablets](https://techterms.com/definition/tablet), and [smartphones](https://techterms.com/definition/smartphone).

As part of this effort, Microsoft developed Windows 10 Mobile alongside Windows 10 to replaces Windows Phone – Microsoft's previous mobile OS. Windows 10 also integrates other Microsoft services, such as Xbox Live and the Cortana voice recognition assistant. While Windows 10 includes many new features, it also brings back the [Start Menu](https://techterms.com/definition/start_menu), which was dropped in Windows 8. The bottom of the Windows 10 Start Menu includes a search bar that allows you to search both your local [PC](https://techterms.com/definition/pc) and the web.

**4.3.4 Blynk**

Blynk is a Platform with IOS and Android apps to control Arduino, Raspberry Pi and the likes over the Internet. It's a digital dashboard where you can build a graphic interface for your project by simply dragging and dropping widgets. Blynk is a new platform that allows you to quickly build interfaces for controlling and monitoring your hardware projects from your iOS and Android device.

With Blynk, though, the software side is even easier than the hardware. Currently, Blynk supports most Arduino boards, Raspberry Pi models, the ESP8266, Particle Core, and a handful of other common microcontrollers and single-board computers, and more are being added over time.

****

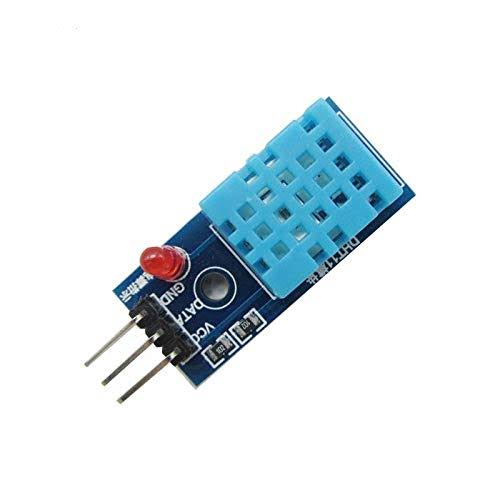
**4.3.5 AT** **Mega 328**

AT Mega-328 is basically an Advanced Virtual RISC (AVR) micro-controller. It supports the data up to eight (8) bits. AT Mega-328 has 32KB internal built-in memory. This micro-controller has a lot of other characteristics. You should also have a look at [Introduction to PIC16F877a](https://www.theengineeringprojects.com/2017/06/pic16f877a.html) (it’s a [PIC Microcontroller](https://www.theengineeringprojects.com/2015/03/pic-microcontroller-projects.html)) and then compare functions of these two Microcontrollers. AT Mega 328 has 1KB Electrically Erasable Programmable Read Only Memory (EEPROM). Moreover, AT Mega 328 has several different features which make it the most popular device in today’s market. These features consist of advanced RISC architecture, good performance, low power consumption, real timer counter having separate oscillator, 6 PWM pins, programmable [Serial USART](https://www.theengineeringprojects.com/2015/06/serial-port.html), programming lock for software security, throughput up to 20 MIPS etc. ATmega-328 is mostly used in [Arduino](https://www.theengineeringprojects.com/2015/03/arduino-projects.html).



**4.3.6 Temperature and Humidity Sensor (DHT11)**

The DHT11 is a basic, ultra-low-cost digital Temperature and Humidity Sensor. It uses a capacitive humidity sensor and a thermistor to measure the surrounding air, and spits out a digital signal on the data pin (no analog input pins needed). Its fairly simple to use, but requires careful timing to grab data. The DHT11 calculates relative humidity by measuring the electrical resistance between two electrodes. The humidity sensing component of the DHT11 is a moisture holding substrate with the electrodes applied to the surface. The change in resistance between the two electrodes is proportional to the relative humidity.



**4.3.7 Bluetooth Module**

Bluetooth Module is a basic circuit set of chip which integrated Bluetooth functions and which can be used in wireless network transmission. Generally, the Bluetooth Module can be divided into the following types: data transmission module, remote control module etc. HC‐05 **module** is an easy to **use Bluetooth** SPP (Serial Port Protocol) **module**, designed for transparent wireless serial connection setup. The HC-05 **Bluetooth Module** can be used in a Master or Slave configuration, making it a great solution for wireless communication. **HC-05 module** is an easy to use **Bluetooth** SPP (Serial Port Protocol) **module**, designed for transparent wireless serial connection setup. Serial port **Bluetooth module** is fully qualified **Bluetooth** V2.



**4.3.8 Gas Sensors (MQ Sensors)**

A **gas sensor** is a device which detects the presence or concentration of gases in the atmosphere. Based on the concentration of the gas the sensor produces a corresponding potential difference by changing the resistance of the material inside the sensor, which can be measured as output voltage. Based on this voltage value the type and concentration of the gas can be estimated. When the concentration of the gas exceeds this threshold, the digital pin goes high. The analog pin can be used to measure the concentration of the gas. Detecting and measuring pollution and the main gases are carried out by means of specific sensors, that are often very expensive. Luckily, for a while it is possible to find low-cost, reliable sensors for sale: that’s the ones from the MQ series. Moreover, they are very cheap and may be easily acquired, even on the various online stores. Although very popular, they are however not very well documented, and it is therefore not easy to find an online library for Arduino, one that is ready and capable of correlating the signals provided with the concentrations detected in the air.



**4.3.9 Buzzer**

A **buzzer** is a device which makes a buzzing or beeping noise. ... More complex **buzzers** include the oscillator **circuit** and the piezoelectric element or speaker in a single package, so all you need to do is apply a voltage and you will get an annoying beeping or buzzing sound. The **buzzer** consists of an outside case with two pins to attach it to power and ground. Inside is a piezo element, which consists of a central ceramic disc surrounded by a metal vibration disc. When current is applied to the **buzzer** it causes the ceramic disk to contract or expand. The most common sizes for Sound Level are 80 dB, 85 dB, 90 dB and 95 db. There are several **types** available including Electro-Acoustic, Electromagnetic, Electromechanics, Magnetic and Piezo, among others.

****

**4.3.10 NodeMCU**

NodeMCU is an open-source firmware and development kit that helps you to prototype or build IoT products. It includes firmware that runs on the ESP8266 Wi-Fi SoC from Espressif Systems, and hardware which is based on the ESP-12 module. The firmware uses the Lua scripting language. The term “NodeMCU” by default refers to the firmware rather than the dev kits. NodeMCU has 128 KB RAM and 4MB of Flash memory to store data and programs. Its high processing power with in-built Wi-Fi / Bluetooth and Deep Sleep Operating features make it ideal for IoT projects. NodeMCU can be powered using Micro USB jack and VIN pin (External Supply Pin). It supports UART, SPI, and I2C interface.



**CHAPTER 5**

**DESIGN**

**5.1 SYSTEM DESIGN**

System design is a reduction of an entire system by studying the various operations performed and their relationships within the system and the requirements of its success. One aspect of design is defining the boundaries of the system and determining whether or not the candidate system should consider other related system. System can be defined, as an orderly grouping of interdependent components can be simple or complex.

The idea of the systems has been most practical and necessary in computerizing the interrelationships and integration of operations, especially when using computers. Thus, it’s a way of thinking organizations and their problems. An organization consists of several interrelated and interlocking components. The most creative and challenging phase of the system life cycle is system design. The term design describes a final system and the process by which it is developed. It refers to the technical specifications that will be applied in implementing the candidate system. It also includes the construction of programs and program testing.

The first step in the system design is to determine how the output is to be produced and in what format. Samples of the output and the inputs are also presented. In the second step, input data and master files are to be designed to meet requirement of the proposed output. The processing phase’s system’s objectives and complete documentation. System design has two phases:

* Logical
* Physical

The logical design reviews the present physical system, prepares the input and output and also prepares a logical design walk- through.

Physical design maps out the details of the physical system, plans the system implementation, devices a test and implementation plan and new hardware and software.

**5.1.1 Block Diagram**

**NodeMCU**

**Cloud**

**MQ7**

**MQ9**

**Buzzer**

**DHT11**

**Bluetooth Module**

This is the block diagram that represents the project **SEWAGE WORKER SAFETY MONITORING SYSTEM** which is an IoT based project. This block diagram will give a brief idea about the proposed system. Here we are using NodeMCU microcontroller and various sensors. Here we are using various gas sensors to detect various hazardous gases inside the manhole. The temperature and humidity sensor will detect the temperature and humidity level inside the manhole and also, we are using an ultrasonic sensor to detect the obstacles.

Also, a button and buzzer will be there on the helmet. This is an emergency button. The worker can press this button if he feels any hesitations while cleaning the manhole. There is also buzzer for alerting the worker. The readings or data from these sensors and the signal when pressing the emergency button will be send to the mobile through a Data monitoring app. Whenever the gas level goes beyond the threshold value an alert will be send to the mobile of the co-worker who stands outside the sewage.

**5.1.2 Project Activity Diagram**

**EmbeddedC Program**

**Sensor**

DoutVCC Compiling

Ground

Convert into binary form

Input

**RAM/ROM**

**Timer**

**A/D**

**CPU**

Oscillator

**Power**

**PWM**

**Digital**

Output

**Buzzer**

**NodeMCU 5V**

In this project the program codes are written in Arduino IDE using EmbeddedC programming language. For the working of the hardware this program should be loaded to the microcontroller, here the MCU is NodeMCU which is suitable for IoT projects. After writing the program in Arduino IDE, we have to compile it. If there is no error, this program will be converted into hexadecimal form. Then these hexadecimal codes are upload to the NodeMCU. The NodeMCU has parts similar to a computer. It has memory (RAM and ROM), CPU, Timer, Power etc.

In memory unit the program again will be converted into binary form. In this Project the MCU takes the sensor readings as input. Mainly most of the sensors has the following 3 pins: Vcc for power supply, GND for ground and Dout for output. The output of the sensor will be the input of MCU. The sensor output will be connected to the digital pin of NodeMCU. Then after processing the input the MCU will produce output through the connected output devices like Buzzer.

* + 1. **UML Use case diagram**

A [use case](https://searchsoftwarequality.techtarget.com/definition/use-case) is a methodology used in system analysis to identify, clarify, and organize system requirements. The purpose of use case diagram is to capture the dynamic aspect of a system. However, this definition is too generic to describe the purpose, as other four diagrams (activity, sequence, collaboration, and State chart) also have the same purpose. Use case diagrams are used to gather the requirements of a system including internal and external influences. These requirements are mostly design requirements. Hence, when a system is analyzed to gather its functionalities, use cases are prepared and actors are identified.

In brief, the purposes of use case diagrams can be said to be as follows

* Used to gather the requirements of a system.
* Used to get an outside view of a system.
* Identify the external and internal factors influencing the system.
* Show the interaction among the requirements are actors.

Use case diagrams are employed in [UML](https://searchsoftwarequality.techtarget.com/definition/Unified-Modeling-Language) (Unified Modeling Language), a standard notation for the modeling of real-world objects and systems. System objectives can include planning overall requirements, validating a [hardware](https://searchnetworking.techtarget.com/definition/hardware) design, testing and [debugging](https://searchsoftwarequality.techtarget.com/definition/debugging) a [software](https://searchapparchitecture.techtarget.com/definition/software) product under development, creating an online help reference, or performing a consumer-service-oriented task.

### **5.1.3.1 BASIC ELEMENTS IN USE CASE DIAGRAM**

* + - * + **Use case:** A use case in a use case diagram is a visual representation of a distinct business functionality in a system.
        + **Actors:** An actor portrays any entity (or entities) that perform certain roles in a given system. The different roles the actor represents are the actual business roles of users in a given system. An actor in a use case diagram interacts with a use case.
        + **System boundary:** A system boundary defines the scope of what a system will be. A system cannot have infinite functionality. The system boundary is shown as a rectangle spanning all the use cases in the system.
        + **Include:** When a use case is depicted as using the functionality of another use case in a diagram, this relationship between the use cases is named as an includerelationship.
        + **Extend:** In an extendrelationship between two use cases, the child use case adds to the existing functionality and characteristics of the parent use case. An extend relationship is depicted with a directed arrow having a dotted shaft, similar to the include relationship.

### **5..1.3.2 COMPONENTS OF USE CASE DIAGRAM**

**System :**

**Relationships :**

**Use cases :**

**Actor :**

5.1.3.3 Use Case Diagram

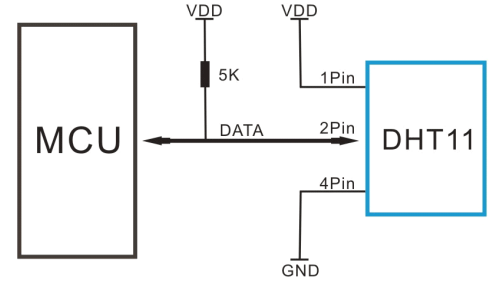
Initialize Sensors

Determine Temp. and Humidity

Detect Gas

Send values to Blynk

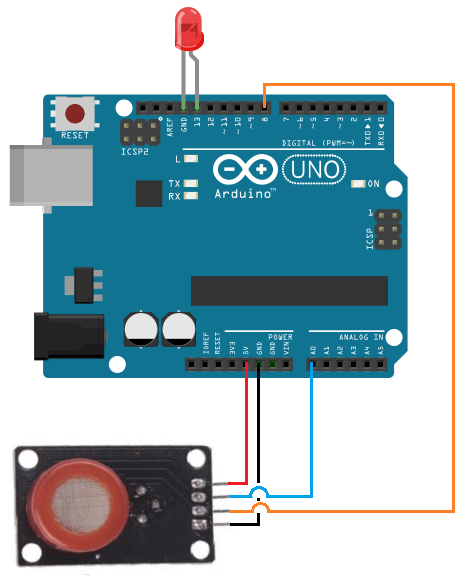
**5.1.4 Working Diagram of DHT11 Sensor**

****

DHT11 is a low-cost digital sensor for sensing temperature and humidity. DHT11 sensor consists of a capacitive humidity sensing element and a thermistor for change them into digital form. For measuring temperature this sensor uses a Negative Temperature coefficient thermistor, which causes a decrease in its resistance value with increase in temperature. To get larger resistance value even for the smallest change in temperature, this sensor is usually made up of semiconductor ceramics or polymers. The temperature range of DHT11 is from 0 to 50 degree Celsius with a 2-degree accuracy. Humidity range of this sensor is from 20 to 80% with 5% accuracy.

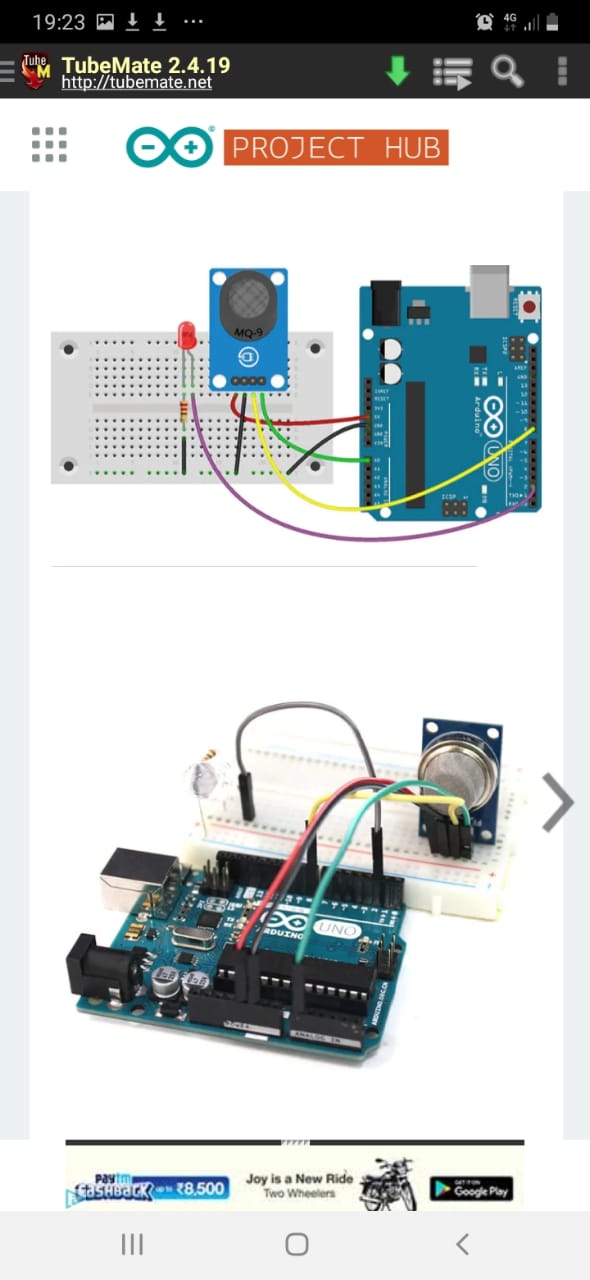
The sampling rate of this sensor is 1Hz.i.e. it gives one reading for every second. DHT11 is small in size with operating voltage from 3 to 5 volts. The maximum current used while measuring is 2.5mA. DHT11 sensor has four pins- VCC, GND, Data Pin and a not connected pin. A pull-up resistor of 5k to 10k ohms is provided for communication between sensor and micro-controller.

**5.1.4 Working Diagram of MQ7 Sensor**

****

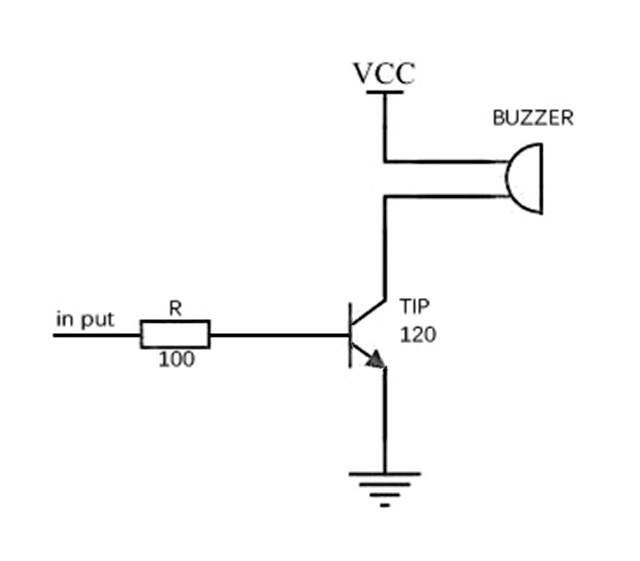
The carbon monoxide sensor we will use is the MQ-7 sensor. This is a sensor that is sensitive to effects of CO. There 4 leads are +5V, AOUT, DOUT, and GND. The +5V and GND leads establishes power for the alcohol sensor. The other 2 leads are AOUT (analog output) and DOUT (digital output). How the sensor works is the terminal AOUT gives an analog voltage output in proportion to the amount of carbon monoxide the sensor detects. The more CO it detects, the greater the analog voltage it will output. Conversely, the less CO it detects, the less analog voltage it will output. If the analog voltage reaches a certain threshold, it will send the digital pin DOUT high. Once this DOUT pin goes high, the Arduino will detect this and will trigger the LED to turn on, signaling that the CO threshold has been reached and is now over the limit.

**5.1.5 Working Diagram of MQ9 Sensor**

****

The MQ9 sensor is sensitive to carbon monoxide and flammable gases. It can detect the detect carbon monoxide density from 10ppm to 1000ppm and flammable gases density from 100ppm to 10000ppm. MQ9 has an internal heater which starts warming up if a 5V voltage is applied. This is an Analog output sensor. This needs to be connected to any one Analog socket in [Grove Base Shield](https://wiki.seeedstudio.com/Base_Shield_V2). The examples used in this tutorial makes uses of A0 analog pin. Connect this module to the A0 port of Base Shield. The output voltage from the Gas sensor increases when the concentration of gas increases. Sensitivity can be adjusted by varying the potentiometer. Please note that the best preheat time for the sensor is above 24 hours.

**5.1.6 Working Diagram of Buzzer**

** **

A **buzzer**is a small yet efficient component to add sound features to our project/system. It is very small and compact 2-pin structure hence can be easily used on [breadboard](https://components101.com/misc/breadboard-connections-uses-guide), Perf Board and even on PCBs which makes this a widely used component in most electronic applications. There are two types are buzzers that are commonly available. The one shown here is a simple buzzer which when powered will make a Continuous Beeeeeeppp.... sound, the other type is called a readymade buzzer which will look bulkier than this and will produce a Beep. Beep. Beep. Sound due to the internal oscillating circuit present inside it. But, the one shown here is most widely used because it can be customized with help of other circuits to fit easily in our application. The buzzer is normally associated with a switching circuit to turn ON or turn OFF the buzzer at required time and require interval.

**5.2 INPUT-OUTPUT DESIGN**

Input design is the basic theory to be considered during system study. The input media used in the system is the sensors. The readings are sent to the mobile of the co-worker. The sensors will detect the level of various gases, temperature, humidity and obstacles. If the gas levels exceed the threshold limit, the co-worker will get alert on his mobile. The readings will be sent to the mobile with the help of a data monitoring app. The user interface design is very important for any application. The interface design defines how the app communicates within itself, to system that interpreted with it and with human who use it. The system is designed in a user-friendly manner.

Product output is the most important one to the user. A major form of the output is the display of the information gathered by the system and the servicing the user requests to the system. Output generally refers to the results or information that is generated by the system. It can be in the form of operational documents and reports. Here the output of the system is the reading that are send by the sensors in the helmet. That data will be shown through Blynk. Output generation hence serves two main purposes, providing proper communication of information to the users and providing data in a form which can be easily understand by the users. The output design phase consists of two stages, output definition and output specification. Output definition takes into account the type of outputs, its contents, formats, its frequency and its volume.

|  |  |  |
| --- | --- | --- |
| **PROCESS** | **INPUT DESIGN** | **OUTPUT DESIGN** |
| Detection of Temperature and Humidity | DHT11 temperature and humidity sensor activated. | Sensing the temperature and density around the sensor. Giving alert if any of the parameter goes beyond the limit |
| Detection of Carbon Monoxide | MQ7 Sensor gets activated. | Sensing the Carbon Monoxide level around the sensor. Giving alert through buzzer if CO level goes beyond the limit. |
| Detection of Methane | MQ9 Sensor gets activated. | Sensing the Methane level around the sensor. Giving alert through buzzer if CH3 level goes beyond the limit. |
| Giving Alerts | Blynk receives data from these Sensors through ESP8266Wifi | Display the data and alert the user if any abnormalities found |

**5.3 PROGRAM DESIGN**

This project has the following major modules according to their functionalities.

1. Software Module
2. Hardware Module

**5.3.1 Software Module**

Software module is a set of instructions, data or programs used to operate computers and execute specific tasks. It’s a generic term used to refer to applications, [scripts](https://whatis.techtarget.com/definition/script) and programs that run on a device. Software can be thought of as the variable part of a computer, and hardware the invariable part. Software is often divided into categories. Application software refers to user-downloaded programs that fulfill a want or need. Examples of applications include office suites, database programs, web browsers, word processors, software development tools, image editors and communication platforms. In this Project the software module has 2 sub modules.

1. **Blynk**

Blynk is a Platform with IOS and Android apps to control Arduino, Raspberry Pi and the likes over the Internet. It’s a digital dashboard where you can build a graphic interface for your project by simply dragging and dropping widgets. Blynk is a Platform with IOS and Android apps to control Arduino, Raspberry Pi and the likes over the Internet. It’s a digital dashboard where you can build a graphic interface for your project by simply dragging and dropping widgets. You’ll also need to install the **Blynk Arduino Library**, which helps generate the firmware running on your ESP8266. Download the latest release from <https://github.com/blynkkk/blynk-library/releases> , and follow along with the directions there to install the required libraries.

Click the “Create New Project” in the app to create a new Blynk app. Give it any name. Blynk works with hundreds of hardware models and connection types. Select the Hardware type. After this, select connection type. In this project we have select WiFi connectivity.

The **Auth Token** is very important – you’ll need to stick it into your ESP8266’s firmware. For now, copy it down or use the “E-mail” button to send it to yourself. Then you’ll be presented with a blank new project. To open the widget box, click in the project window to open. We are selecting a button to control Led connected with NodeMCU.

1. **Arduino IDE**

Arduino is an open-source electronics platform based on easy-to-use hardware and software. [Arduino boards](https://www.arduino.cc/en/Main/Products) are able to read inputs - light on a sensor, a finger on a button, or a Twitter message - and turn it into an output - activating a motor, turning on an LED, publishing something online. You can tell your board what to do by sending a set of instructions to the microcontroller on the board. To do so you use the [Arduino programming language](https://www.arduino.cc/en/Reference/HomePage) (based on [Wiring](http://wiring.org.co/)), and [the Arduino Software (IDE)](https://www.arduino.cc/en/Main/Software), based on [Processing](https://processing.org/). It is simple and accessible user experience; Arduino has been used in thousands of different projects and applications. The Arduino software is easy-to-use for beginners, yet flexible enough for advanced users. It runs on Mac, Windows, and Linux. Arduino is a key tool to learn new things. Anyone - children, hobbyists, artists, programmers - can start tinkering just following the step by step instructions of a kit, or sharing ideas online with other members of the Arduino community. It offers some advantage for teachers, students, and interested amateurs over other systems:

* **Inexpensive** - Arduino boards are relatively inexpensive compared to other microcontroller platforms. The least expensive version of the Arduino module can be assembled by hand, and even the pre-assembled Arduino modules cost less than $50
* **Cross-platform** - The Arduino Software (IDE) runs on Windows, Macintosh OSX, and Linux operating systems. Most microcontroller systems are limited to Windows.
* **Simple, clear programming environment** - The Arduino Software (IDE) is easy-to-use for beginners, yet flexible enough for advanced users to take advantage of as well
* **Open source and extensible software** - The Arduino software is published as open source tools, available for extension by experienced programmers. The language can be expanded through C++ libraries.
* **Open source and extensible hardware** - The plans of the Arduino boards are published under a Creative Commons license, so experienced circuit designers can make their own version of the module, extending it and improving it.

**5.3.2 Hardware Module**

Computer hardware is a collective term used to describe any of the physical components of an analog or digital computer. The term hardware distinguishes the tangible aspects of a computing device from software. Computer hardware can be categorized as having either internal or external components. External components, also called [peripheral](https://searchmobilecomputing.techtarget.com/definition/peripheral) components, are those items that are often connected to the computer in order to control either its input or output. In this project we have the following sub modules in the hardware module.

1. **NodeMCU**

NodeMCU is an open-source Lua based firmware and **development board** specially targeted for IoT based Applications. It includes firmware that runs on the ESP8266 Wi-Fi SoC from Espress if Systems, and hardware which is based on the ESP-12 module. There is online NodeMCU custom builds available using which we can easily get our custom NodeMCU firmware as per our requirement.

### **NodeMCU Development Board Pinout Configuration**

|  |  |  |
| --- | --- | --- |
| **Pin Category** | **Name** | **Description** |
| Power | Micro-USB, 3.3V, GND, Vin | **Micro-USB:** NodeMCU can be powered through the USB port  **3.3V:** Regulated 3.3V can be supplied to this pin to power the board  **GND:** Ground pins  **Vin:**External Power Supply |
| Control Pins | **EN, RST** | The pin and the button resets the microcontroller |
| Analog Pin | A0 | Used to measure analog voltage in the range of 0-3.3V |
| GPIO Pins | GPIO1 to GPIO16 | NodeMCU has 16 general purpose input-output pins on its board |
| SPI Pins | SD1, CMD, SD0, CLK | NodeMCU has four pins available for SPI communication. |
| UART Pins | TXD0, RXD0, TXD2, RXD2 | NodeMCU has two UART interfaces, UART0 (RXD0 & TXD0) and UART1 (RXD1 & TXD1). UART1 is used to upload the firmware/program. |
| I2C Pins |  | NodeMCU has I2C functionality support but due to the internal functionality of these pins, you have to find which pin is I2C. |

### **NodeMCU ESP8266 Specifications & Features**

* Microcontroller: Tensilica 32-bit RISC CPU Xtensa LX106
* Operating Voltage: 3.3V
* Input Voltage: 7-12V
* Digital I/O Pins (DIO): 16
* Analog Input Pins (ADC): 1
* UARTs: 1
* SPIs: 1
* I2Cs: 1
* Flash Memory: 4 MB
* SRAM: 64 KB
* Clock Speed: 80 MHz
* USB-TTL based on CP2102 is included onboard, Enabling Plug n Play
* PCB Antenna

1. **DHT11 Sensor**

The DHT11 is a basic, ultra-low-cost digital Temperature and Humidity Sensor. It uses a capacitive humidity sensor and a thermistor to measure the surrounding air, and spits out a digital signal on the data pin (no analog input pins needed). Its fairly simple to use, but requires careful timing to grab data.

### **DHT11** **Pinout Configuration**

|  |  |  |
| --- | --- | --- |
| **For DHT11 Sensor** | | |
| 1 | Vcc | Power supply 3.5V to 5.5V |
| 2 | Data | Outputs both Temperature and Humidity through serial Data |
| 3 | NC | No Connection and hence not used |
| 4 | Ground | Connected to the ground of the circuit |

|  |  |
| --- | --- |
| **For DHT11 Sensor module** | |
| Vcc | Power supply 3.5V to 5.5V |
| Data | Outputs both Temperature and Humidity through serial Data |
| Ground | Connected to the ground of the circuit |

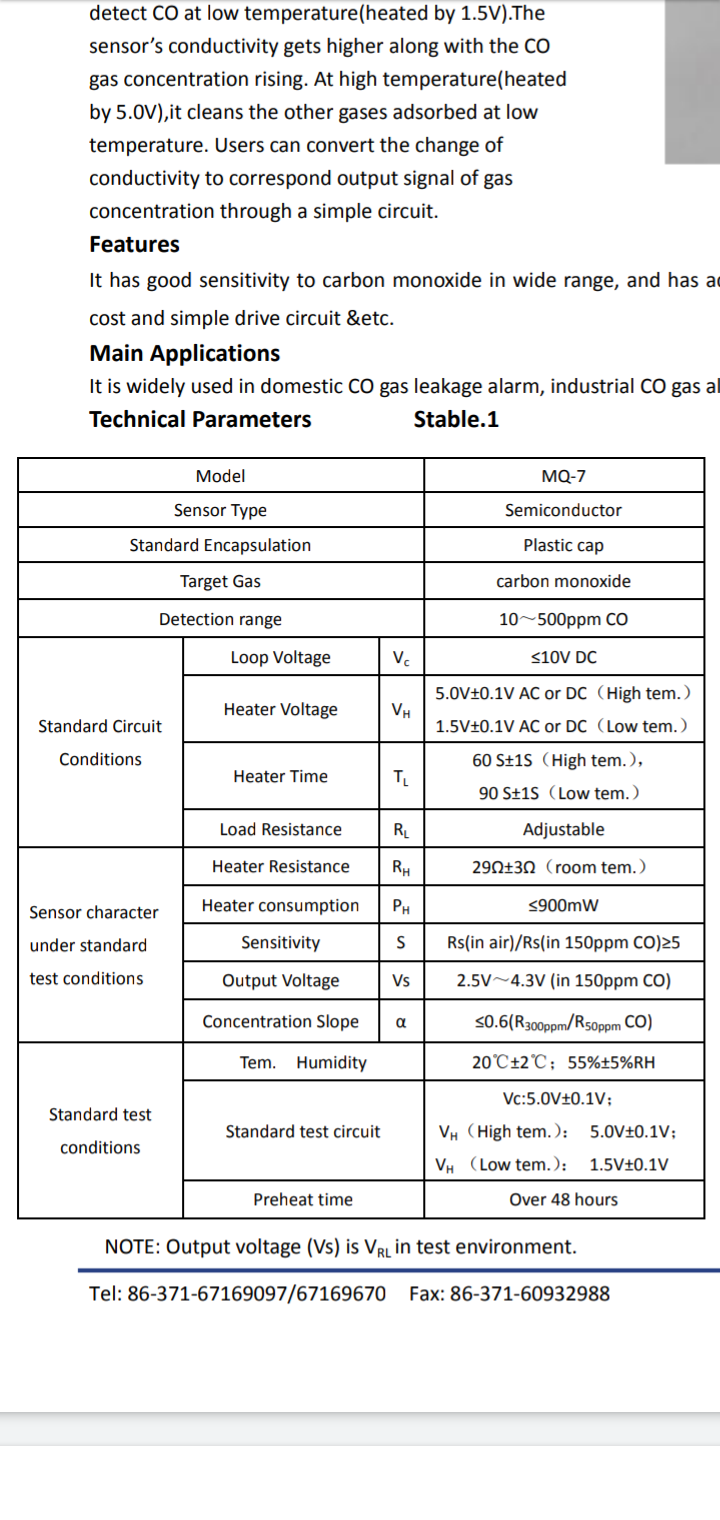
### **DHT11 Specifications:**

* Operating Voltage: 3.5V to 5.5V
* Operating current: 0.3mA (measuring) 60uA (standby)
* Output: Serial data
* Temperature Range: 0°C to 50°C
* Humidity Range: 20% to 90%
* Resolution: Temperature and Humidity both are 16-bit
* Accuracy: ±1°C and ±1%

1. **MQ7 Sensor**

The MQ7 is a simple-to-use Carbon Monoxide (CO) sensor suitable for sensing CO concentrations in the air. It can detect CO gas concentrations anywhere from 20 to 2000ppm. MQ7 is a high sensitivity to carbon monoxide and stable and long-life span. It makes detection by method of cycle high and low temperature. The sensor’s conductivity is higher along with the gas concentration rising.

**MQ7 Sensor Pin Configuration**



1. **MQ9 Sensor**

The Grove-Gas Sensor (MQ9) module is useful for gas leakage detection. It is suitable for detecting LPG, CO, CH4. Due to its high sensitivity and fast response time, measurements can be taken as soon as possible. The sensitivity of the sensor can be adjusted.

**MQ9 Pin Configuration**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Item | Parameter | Min | Typical | Max | Unit |
| VCC | Working Voltage | 4.9 | 5 | 5.1 | V |
| PH | Heating consumption | 0.5 | - | 340 | mW |
| RL | Load resistance |  | adjustable |  |  |
| RH | Heater resistance | - | 33Ω±5% | - | Ω |
| Rs | Sensing Resistance | 2 | - | 20000 | Ω |
| CO/CH4/LPG Scope | Detecting Concentration | 200 |  | 10000 | ppm |

#### **Specifications:**

* Power supply: 5V
* Interface type: Analog, Digital
* Type: Semiconductor
* Concentration: Good sensitivity to CO/Combustible gas
* High sensitivity to Methane, Propane and CO
* Long life and low cost
* Size: 32x20mm
* Sensor Type: Semiconductor
* Standard Encapsulation: Bakelite
* Detection Gas: CO and combustible gas
* Loop Voltage: ≤10V DC

1. **Buzzer**

A **buzzer** is a device which makes a buzzing or beeping noise. The **buzzer** consists of an outside case with two pins to attach it to power and ground. Inside is a piezo element, which consists of a central ceramic disc surrounded by a metal vibration disc. When current is applied to the **buzzer** it causes the ceramic disk to contract or expand.

### **Buzzer Pin Configuration**

|  |  |  |
| --- | --- | --- |
| **Pin Number** | **Pin Name** | **Description** |
| 1 | Positive | Identified by (+) symbol or longer terminal lead. Can be powered by 6V DC |
| 2 | Negative | Identified by short terminal lead. Typically connected to the ground of the circuit |

### **Buzzer Features and Specifications**

* Rated Voltage: 6V DC
* Operating Voltage: 4-8V DC
* Rated current: <30mA
* Sound Type: Continuous Beep
* Resonant Frequency: ~2300 Hz
* Small and neat sealed package
* Breadboard and Perf board friendly

s

**CHAPTER 6**

**FUNCTIONAL AND NON-FUNCTIONAL REQUIREMENTS**

## **6.1 FUNCTIONAL REQUIREMENTS**

Functional requirements represent the intended behaviour of the system. This behaviour may be expressed as services, tasks or functions that the specified system is required to perform. The system is should be designed to capture all human faces and recognize it through deep learning. The following functional requirements have been identified for this project.

* Description of data to be entered into the system.
* Description of operations performed by each input.
* Description of work-flows performed by the system.
* Descriptions of system outputs.
* How the system meets applicable regulatory requirements?

## **6.2 NON-FUNCTIONAL REQUIREMENTS**

A non-functional requirement is a requirement that specifies criteria that can be used to judge the operation of a system, rather than specific behaviours. Non-functional requirements are “system shall be requirement ".

Non-functional requirements are often called qualities of a system. Other terms for non-functional requirements are "constraints", "quality attributes”, “quality goals", "quality of service requirements" and "non-behavioural requirements. Some of the non-functional requirements are mentioned below

* **Usability**: The system shall have a clean interface with only needed features, clear terminology and tool tips wherever necessary. Warnings or alerts shall be specified in clear way.
* **Efficiency**: The system shall respond to different searches being conducted like searching particular product, search quantity, etc. in a very fast way.
* **Interoperability**: The system shall be able to interact with other systems. The system should able to be supported at least one software which has a relationship with payment process.
* **Portability**: The system shall be independent of the specific technological platform used to implement it.
* **Reliability**: Reliability defined as a measure of the time between failures occurring in a system (measure show frequently the system fails), so that the system shall operate without any failure for a particular period of time
* **Availability**: Availability measures the percentage of time the system is in its operational state so that the system shall be available for use 24 hours per day and 365days per year.

**CHAPTER 7**

**TESTING**

Software/Hardware testing is critical element of software/Hardware quality assurance and represents the ultimate review of specifications, design and code generation. System testing is the stage of implementation, it is aimed for ensuring that the system works accurately and efficiently before live operations commences.

* Testing is a purpose of executing a programmed with intend of finding errors.
* Preparing a test case that has high probability of finding undiscovered errors.
* Testing to erase out all kinds of bucks from the program.

Before going for testing, first we have to decide the type of test. For this impact system, unit testing is carried out. And the following things are taken to consideration.

* To ensure that information properly places in and out of the program.
* To ensure that the module operates properly at boundaries established to limit or restrict processing.
* To find out whether all statements in module have been executed at least once.
* To find out whether error handling paths are working correctly.

## **7.1 TESTING STRATEGIES**

A strategy for software/Hardware testing integrates software/Hardware test case design methods in to a well-planned series of steps that results in the successful construction of the software/Hardware. The strategy provides a road map that describes the step to be conducted as part of testing, when these steps are planned and undertaken, and how much effort, time and resources will be required. Therefore, any testing strategy must incorporate test planning, test case, design, test execution and resultant data collection and evaluation. A software/Hardware testing strategy should be flexible enough to promote customized testing approach. At the same time, it must be rigid enough to promote reasonable planning and management tracking as the project processes. The project manager, software engineer and testing specialists develop a strategy for s software/Hardware testing. The general characteristics of software/Hardware testing strategy are:

* Testing begins at the component level and works “outward” toward the integration of the entire computer system.
* Different testing techniques are appropriate at different point in time.

A strategy for software/Hardware testing must accommodate low-level testis that are necessary to verify a small source code segment has been correctly implemented as well as high level testing that validate major system function against customer requirements.

**7.2 UNIT TESTING**

Unit test comprises of a set test performed by an individual programmer prior to the integration of the unit into large system. Program unit is usually small enough that the programmers who developed and can it in great detail and certainly in greater than will possible when the unit is integrated into evolving software project. Unit testing should be an exhaustive as possible. In this system, each module was tested individually to ensure that every representation in the module meets the requirements.

Prior to integration of the unit in to a large system, it is necessary to perform a set of tests to each unit of the system. Accordingly, functional efficiency of all sensors used, alert by Blynk application and correct storing of patient data etc. by the android app were tested repeatedly. Details of unit testing are appended in the table below for clarity.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **SL.NO** | **Procedures** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1 | Testing of temperature and humidity sensor for accurate result. | Temperature and humidity of sewage measured must be accurate. | Measuring accurately. | Pass. |
| 2 | Testing accuracy in measuring carbon monoxide rate by the sensor. | CO rate must be measured accurately every time. | CO rate measured accurately every time. | Successful. |
| 3 | Measuring of Methane | Measurement of correct Methane level is a must. | Measured accurately. | Accurate result. |
| 4 | Alert by Blynk application (Notification). | System must send notification to alert co-worker. | System did send alert signal on sensing up normal result. | Pass |
| 5 | All the captured readings from the sewage are sent to the respective co-worker’s android application (Blynk app). | Sewage reading to be accurately monitored in the android application. | Same as expected. | Pass |

## **7.3 INTEGRATION TESTING**

Integration testing is a system technique for constructing the program structure while at the same time conducting test to uncover errors associated with interfacing. The objective is to take unit testing modules and build a program structure that has been dictated by design. Bottom-up integration is the traditional strategy used to integrate the components of a software/hardware system into functioning whole.

Bottom-up integration consists of a unit test followed by testing of the entire system. Subsystem consists of several modules that communicated with other defined interface.

The errors were isolated and corrected to produce a fully functional system. Top-down integration method is an incremental approach to the construction of the program structure. The project was tested to ensure that every representation meets the requirements.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **SL.NO** | **Procedures** | **Expected Result** | **Actual Result** | **Pass or Fail** |
| 1 | Checking integrated output of gas sensors | The values will be correctly integrated. | Correct integration notified. | Pass |
| 2 | Accessibility of helmet by sensors. | grant or denial of output. | Same as expected. | Pass |
| 3 | Correct passage of gas sensors NodeMCU output by mobile application. | Sensor data to be correctly integrated by sensors. | Same as expected. | Pass |

**7.4 SYSTEM TESTING**

In system testing, the software and other system elements are tested as a whole. System testing is actually a series of different tests whose primary purpose is to fully exercise the IOT based Sewage Worker Safety Monitoring system. A table showing the details of various system tests and their results are appended below: -

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **SL.NO** | **Procedures** | **Input** | **Expected Output** | **Pass or Fail** |
| 1 | Temperature and Humidity determination of sewage. | Check temperature and humidity input of sewage. | Actual temperature and humidity of sewage. | Is reading temperature and humidity correctly. |
| 2 | Determination of CO rate . | Level of CO inside the sewage. | Actual measure of CO level. | Reading correctly |
| 3 | Determination of CH3 rate. | Level of CH3 inside the sewage. | Actual measure of CH3 level. | Reading correctly |
| 4 | Blynk operation accurately. | parameters from sensors. | Correct interpretation of readings and detection of abnormalities and notifying the same. | Pass |

## **7.5 USER ACCEPTANCE TESTING**

This testing is generally performed when the project is nearing its end. This test mainly qualifies the project and decides if it will be accepted by the users of the system. The users or the customers of the project are responsible for the test.

## **7.6 DATA VALIDATION TESTING**

Data validation is the process of testing the accuracy of data; a set of rule you can apply to a control to specify the type and range of data that can enter. It can be used to display error alert when users enter incorrect values into a form. In this project data validation testing carried out on all input form pages to test the accuracy.

## **7.7 OUTPUT TESTING**

After performing the validation testing, the next step is output testing of the proposed system since no system could be useful if it does not produces the required output generated or considered into two ways; one is on screen and another is printed format. The output format on the screen is found to be correct as the format was designed in the system design phase according to the user needs. For the hard copy also, the output comes out as the specified requirements by the user. Hence output testing does not result in any correction in the system.

## **7.8 TESTING RESULT**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **SL.NO** | **Procedures** | **Input** | **Expected Output** | **Pass or Fail** |
| 1 | Temperature and Humidity determination of patient. | Temperature and Humidity input of Sewage. | Actual temperature and Humidity of sewage. | Is reading temperature and humidity correctly. |
| 2 | Determination of CO level. | Sewage CO level. | Actual measure of CO level | Reading correctly |
| 2 | Determination of CH3 level. | Sewage CH3 level. | Actual measure of CH3 level | Reading correctly |
| 4 | Blynk operation accurately. | parameters from sensors. | Correct interpretation of readings and detection of abnormalities and notifying the same. | Pass |

**CHAPTER 8**

# **RESULTS AND DISCUSSION**

.

In India sewage cleaning from manholes and drains are a difficult and risky job for anyone. In 2013, Supreme Court of India gave a decision that all such jobs would be done by machines and wherever human intervention is required, proper safety equipment should be provided to the workers. However, the municipalities give contract to contractors who hire local people to clean the sewage. In the existing system there are no efficient real time health and safety monitoring system. So many researchers proposed so many ideas, but they are not reliable. In the existing system they use very limited facilities like a gas sensor and a heartbeat sensor. Under the drainage the possibilities of current monitoring system implementation are quite difficult. There is no unified equipment that can provide all the services in one place. In our proposed system we are introducing a new system which can solve all the problems in the existing system. Here we are using a helmet which can be wear by the sewage workers. This helmet consists of various sensors, which will measure the various gas levels, oxygen level, and body temperature and humidity in the sewage. Here we are also including an ultra-sonic sensor to detect the obstacle in the drainage. Also, we are providing a facility for alert the co-workers when the things will go worse. Also, our safety smart helmet has an emergency button that can be used by the sewage worker when he feels any suffocation. Here we are unifying all the services and facilities that we can provide, in one equipment, that is, through a helmet, which will work efficiently to save the life of sewage workers. The expenditure our proposed system is very cost effective.

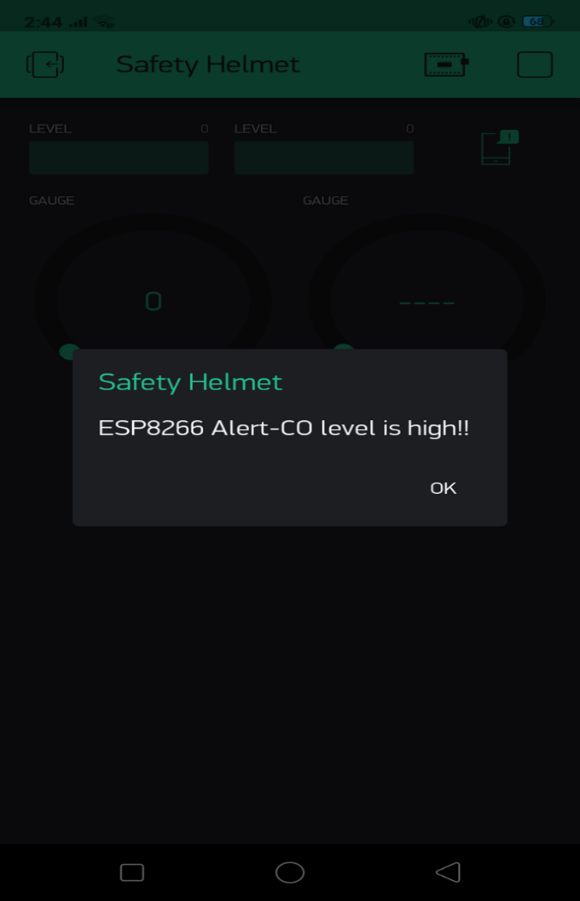
## **8.1 RESULTS**

The proposed system incorporated with the following features.

* Quick and appropriate action can be taken easily for the user.
* Improved efficiency and easy to use.
* Flexibility and Portability.
* Security enhanced for human life.

## **8.2 SCREEN SHOTS**

## 

s

**CHAPTER 9**

# **CONCLUSION**

## **9.1 SYSTEM IMPLEMENTATION**

Implementation means converting a new design into iteration. During implementation there should be a strong interaction between the developer of the software/hardware and the users. Implementation involves installing hardware terminals and training the operating staff. In this phase, user training is critical for minimizing reluctance to change and giving the new system a chance to prove its worth. The new system may be totally new replacing the existing system, or it may be the modifications of existing system. In either case proper implementation is essential to provide a reliable system to meet organizational requirements.

The project “SEWAGE WORKER SAFETY MONITORING SYSTEM” is an implementation of sewage worker safety system using sensors and microcontroller. The sensors and microcontrollers are bind together into a helmet. Here the system uses a data monitoring app called Blynk to receive the data from the sensors which are attached to the helmet.

## **9.2 CONCLUSION**

The project has addressed the security issue of sewage workers. This project is a module which can be incorporated with any other system including hardware for building some fully automated machines. The project is designed in such a way that future modification can be done easily. The following conclusions can be deduced from the development of the project: It provides an abstract platform between the users and avoid manual work.

The experimental results of the system proposed indicate that it may provide a consistent support and assistance for safe and secure life with minimum cost of the system. Arduino Uno is a strong and reliable embedded system device for the complex and challenging tasks. Using these technologies in the proposed system will bring several advantages in providing safety and security for the sewage worker and also comfortable for the co-workers. With the use of Arduino Uno, Sensors and other various important modules, the lives of sewage workers. This technology will definitely improve the security of our sewage workers.

## **9.3 FUTURE ENHANCEMENT**

* To improve with robotic intervention in sewaging so that we can save the lives of innocent peoples.
* To implement body condition sensors or something else to improve the performance and efficiency of the system.
* It can also be used in mining etc.

**BIBLIOGRAPHY**

1. **BOOKS**

* “The Internet of Things” by Samuel Greengard.
* “The Second Machine Age: Work, Progress and Prosperity in a Time of Brilliant Technologies” by Erik Brynjolfsson and Andrew McAfee.
* “The Silent Intelligence’ by Daniel Kellemereit and Daniel Obodovski.
* “Getting Started With internet of Things” by Cuno Pfister.
* “Precision: Principles, Practices and Solutions for the Internet of Things by Timothy Chou.

1. **WEBSITES**

* <https://roboindia.com/tutorials/blymk-introduction-nodemcu/>
* <https://www.arduino.cc/en/Tutorial/HomePage?from=Main.Tutorials>
* <https://www.tutorialspoint.com/arduino/index.html>
* <https://www.open-electronics.org/presenting-mq-sensors-low-cost-gas-and-pollution-detectors/amp/>
* <https://www.keyence.co.in/ss/products/sensors/sensorbasics/ultrasonic/info/>
* <https://www.electronics-tutorials.ws/io/io_3.html>
* <https://www.efe.com/efe/english/world/india-s-sewer-cleaners-dangerous-work-for-little-money/50000262-3788963>
* <https://www.microchip.com/wwwproducts/en/ATmega328>
* <https://playground.arduino.cc/Main/MQSensors/>
* <https://www.hackerearth.com/blog/developers/arduino-programming-for-beginners/amp/>

**APPENDICES**

1. **SCRUM MODEL**
2. **Git**

Git is a version-control system for tracking changes in computer files and coordinating work on those files among multiple people. It is primarily used for source code management in software development, but it can be used to keep track of changes in any set of files. As a distributed revision-control system, it is aimed at speed, data integrity, and support for distributed, non-linear workflows.

1. **Git Repositories**

A Git repository contains the history of a collection of files starting from a certain directory. The process of copying an existing Git repository via the Git tooling is called cloning. After cloning a repository, the user has the complete repository with its history on his local machine. Of course, Git also supports the creation of new repositories.

If you want to delete a Git repository, you can simply delete the folder which contains the repository. If you clone a Git repository, by default, Git assumes that you want to work in this repository as a user. Git also supports the creation of repositories targeting the usage on a server.

1. **Scrum**

Scrum is an agile way to manage a project, usually software development. Agile software development with Scrum is often perceived as a methodology; but rather than viewing Scrum as methodology, think of it as a framework for managing a process. In the agile scrum world, instead of providing complete, detailed descriptions of everything is to be done on a project, much of it is left up to the Scrum software development team. This is because the team will know best how to solve the problem they are presented.

In the agile Scrum world, instead of providing complete, detailed descriptions of how everything is to be done on a project, much of it is left up to the Scrum software development team. This is because the team will know best how to solve the problem they are presented.

Within agile development, Scrum teams are supported by two specific roles. The first is a Scrum Master, who can be thought of as a coach for the team, helping team members use the Scrum process to perform at the highest level. The Product Owner (PO) is the other role, and in Scrum software development, represents the business, customers or users, and guides the team toward building the right product.

1. **Git History**

1. **LIST OF FIGURES**

|  |  |  |
| --- | --- | --- |
| **Figure No.** | **Name** | **Page No.** |
| 2.1.3 | Reports and Images related to the current sewage system | 6 |
| 4.3.1 | Arduino IDE | 15 |
| 4.3.4 | Blynk | 17 |
| 4.3.5 | AT Mega 328 | 17 |
| 4.3.6 | DHT11 Sensor | 18 |
| 4.3.7 | Bluetooth Module | 18 |
| 4.3.8 | MQ Sensors | 19 |
| 4.3.9 | Buzzer | 20 |
| 4.3.10 | Node MCU | 20 |
| 5.1.1 | Block Diagram | 22 |
| 5.1.2 | Project Activity Diagram | 23 |
| 5.1.3.1 | Components of Use case Diagram | 25 |
| 5.1.3.2 | Use Case Diagram | 26 |
| 5.1.4 | Working Diagram of DHT11 Sensor | 26 |
| 5.1.5 | Working Diagram of MQ Sensor | 27 |
| 5.1.6 | Working Diagram of MQ9 Sensor | 28 |
| 5.1.7 | Working Diagram Buzzer | 28 |
| 8.2 | Screenshots | 48 |
| iv | Git History | 53 |
| 1. **List of Tables**  |  |  |  | | --- | --- | --- | | **Table No.** | **Name of Tables** | **Page No.** | | 5.2 | Input-Output Design | 30 | | 5.3.2(a) | NodeMCU Pinout Configuration | 33 | | 5.3.2(b) | DHT11 Pinout Configuration | 35 | | 5.3.2(c) | MQ7 Pinout Configuration | 36 | | 5.3.2(d) | MQ9 Pinout Configuration | 37 | | 5.3.2(e) | Buzzer Pinout Configuration | 38 | | 7.2 | Unit Testing | 43 | | 7.3 | Integration Testing | 44 | | 7.4 | System Testing | 45 | | 7.7 | Output Testing | 46 | | | |

1. **ABBREVIATIONS AND NOTATION**
   * 1. **GIT**

GIT is a distributed version-control system for tracking changes in source code during software development. It is designed for coordinating work among programmers, but it can be used to track changes in any set of files.

* + 1. **IoT**

IoT is Internet of thing. It is simply “A network of interconnected objects able to collect and exchange data”. It is an essential platform where embedded devices are connected to the internet.

* + 1. **DHT11 Sensor**

It is the abbreviation of Digital Humidity and Temperature 11 Sensor. It uses a capacitive humidity sensor and a thermistor to measure the surrounding air, and spits out a digital signal on the data pin (no analog input pins needed). It’s fairly simple to use.

* + 1. **UML**

UML (Unified Modelling Language) is a general-purpose modelling language. The main aim of UML is to define a standard way to visualize the way a system has been designed. It is quite similar to blueprints used in other fields of engineering.

1. **CODING**