**FireGuard: Real-Time Smoke Detection Using IoT and Machine Learning**

***B.Tech in Computer Science and engineering***

**Name-shiven gupta**

[**Mail-gshiven11@gmail.com**](mailto:Mail-gshiven11@gmail.com)

**Contact-7428945114**



# ABSTRACT

In this paper, we discuss a Smart Smoke Detection System based on the IoT concept that will also augment security through signal monitoring and alert communication. We incorporated high-tech sensors that help the system to identify smoke and toxic gases to enable timely

actions that may be required in case of fire occurrences. The system is tied to a cloud-based platform which is capable of processing the data coming from various sensors and send

instant alert to the specified recipient, or to emergency services through the use of SMS, email or mobile app notifications. The built-in alarm also adapts learning algorithms to reduce false alarm and can connect with other smart home appliances for auto safety solution like turning off the electrical circuit or triggering the sprinkler system. Not only does this unique method offer appropriate security measures for people’s homes, businesses, and industries but it does so in ways that are flexible and adaptable for the broadest range of customers, thus contributing to the creation of a safer and smarter society.

# TABLE OF CONTENTS

|  |  |  |
| --- | --- | --- |
| **Sl.No** | **Contents** | **Page No.** |
|  | **Abstract** | **2** |
| **1.** | **INTRODUCTION** | **1** |
|  | 1.1 Background | **4** |
|  | 1.2 Motivations | **5** |
|  | 1.3 Scope of the Project | **6** |
| **2.** | **PROJECT DESCRIPTION AND GOALS** |  |
|  | 2.1 Literature Review | **7** |
|  | 2.2 Gaps Identified | **11** |
|  | 2.3 Objectives | **14** |
|  | 2.4 Problem Statement | **15** |
|  | 2.5 Project Plan | **16** |
| **3.** | **REQUIREMENT ANALYSIS \*** | **18** |
|  |  |  |
|  |  |  |
|  |  |  |
| **4.** | **SYSTEM DESIGN\*** | **20** |
|  |  |  |
|  |  |  |
|  |  |  |
| **5.** | **REFERENCES** | **23** |

**Note \***

**Respective guides can decide on the contents to be prepared in consultation with the students.**

# 1.INTRODUCTION

1.1BACKGROUND

The conventional smoke detection systems have used either the ionization technique or the photoelectric system to detect smoke. Ionization smoke detectors contain a small radioactive

source which ionizes the air in a sensing chamber. The detection of smoke in this chamber

interferes with the ionization process hence letting out an alarm. There are two types of smoke detectors which include the ionization type and the photoelectric type; the

photoelectric smoke detectors use a light source and light sensitive element. Smoke particles

in the chamber cause the light to bounce back in all directions hence triggering the alarm.

While these systems are highly useful in many situations, they have remarkable disadvantages. Over 90% of conventional detectors are local only, and once the smoke is

detected, it sounds a local alarm only, and is unable to interface with other systems or call outside help. Furthermore, these detectors can often give false positives because they can

pick up on activities that are non-threatening such as cooking; in turn, there may be alarm

fatigue where the people ignore the alarms completely. Moreover, conventional systems are not as advanced as they offer no real-time delay notifications and compatibility with smart home devices as smart systems provide for enhancing safety.

1.2 MOTIVATIONS

This paper argues that there is a need to have IoT-based Smart Smoke Detection System that would address the limitations of the traditional smoke Detector. Conventional systems are generally predisposed to not offer real-time alerts and adaptive learning to least generate

alarms. This makes it hard to link up to other networks thus if an alarm is sounded then it may not reach all the people within that area this would slow t the response during an

emergency. The coming of IoT technology creates a chance to introduce innovative smoke detecting devices for the purpose of monitoring the situation in real time and deliver

notifications about it to the users via SMS, email or mobile applications. It also assures that the users will be immediately notified of the possible risks irrespective of the geographical

location. In addition, the adaptive learning algorithms can be used to distinguish between real fire threats as well as other inconsequential incidents more frequently, thereby minimizing on false alarms. While implementing the new system through IoT, is also capable of interacting

with other smart devices, thus promote safety through automatic reactions like switching off the circuits or turning on the sprinkler

1.3 SCOPE OF THE PROJECT

This paper presents the IoT-based Smart Smoke Detection System which seeks to overcome the shortcomings associated with the conventional smoke detector through the incorporation of technology and connectivity. This system will use specific sensors that will not only

identify the smoke but also the toxic gases such as the carbon monoxide making it a better security measure. Integrated with an IoT device, the system will be able to access a cloud-

based platform to monitor, process and notify relevant parties in real-time, through multiple

media. It will be possible to use machine learning algorithms that will help in decreasing false alarms since it is capable of identifying real fire threats as well as benign sources. This

function will also incorporate smart home functions in order for the system to independently

activate safety mechanisms such as turning off electricity or opening up the sprinkler in cases of an identified dangerous element. The project will then be aimed at developing a flexible

design that could be adapted to the different buildings including homes, offices among others.

This approach makes sure that the system has ability to not only improve safety but also

# 2.PROJECT DESCRIPTION AND GOALS

2.1.Literature review

1. **AN ARCHITECTURAL APPROACH TOWARDS THE FUTURE INTERNET OF**

**THINGS Uckelmann, D., Harrison, M., & Michahelles, F. (2011)**

This paper explores an architectural approach to the development of the Internet of Things

(IoT) as part of the future internet. It discusses the technological evolution of IoT, various architectural models, and possible future applications, providing a foundation for

understanding how IoT will shape global communication and device interaction.

1. **. SMART FIRE ALARM SYSTEM USING IOT AL Shereiqi, I. M., & mad Sohail,**

## M.(2020)

This research introduces a smart fire alarm system using IoT technologies. The authors highlight the system's effectiveness in detecting fires in real-time and transmitting alerts through IoT devices. The study focuses on how IoT can enhance fire prevention methods with more accurate and timely responses.

**3. APPLICATION OF NB-IOT IN INTELLIGENT FIRE PROTECTION SYSTEM Li,**

## T.,&Hou,P.(2019)

This paper focuses on the application of Narrowband IoT (NB-IoT) in intelligent fire protection systems. Presented at an international conference, the authors discuss how NB-IoT can be utilized for monitoring and detecting fires in real-time, offering solutions for improving safety in various environments through intelligent systems.

1. **LOW-COMPLEXITY HIGH-PERFORMANCE DEEP LEARNING MODEL FOR REAL-TIME LOW-COST EMBEDDED FIRE DETECTION SYSTEMS Jadon, A.,**

**Varshney, A., & Ansari, M. S. (2020)**

The study presents a low-complexity, high-performance deep learning model designed for embedded fire detection systems. The authors aim to create a low-cost and real-time fire detection system that can operate effectively with limited resources, providing an innovative approach to fire safety in IoT environments.

1. **IOT-BASED INTELLIGENT MODELING OF SMART HOME ENVIRONMENT FOR FIRE PREVENTION AND SAFETY Saeed, F., Paul, A., Rehman, A., Hong, W.**

## H., & Seo, H.(2018)

In this article, the authors develop an IoT-based intelligent model for smart home environments, focusing on fire prevention and safety. The system integrates various sensors to detect fire hazards and offers a scalable and efficient solution for ensuring safety in smart homes through IoT technologies.

**6. IOT-BASED FIRE ALARM SYSTEM Mahgoub, A., Tarrad, N., Elsherif, R., Al-Ali,**

## A., & Ismail, L. (2019)

This paper presents an IoT-based fire alarm system aimed at improving fire detection and alerting capabilities. The authors discuss how IoT integration enhances traditional fire alarm systems by allowing real-time monitoring and response, leading to better safety measures.

The system is tested in various environments to evaluate its efficiency and scalability.

## 7. WEBSERVER BASED SMART MONITORING SYSTEM USING ESP8266 NODE MCU MODULE Aziz, D. A. (2018)

The author introduces a web server-based smart monitoring system using the ESP8266 Node

MCU module. This system facilitates remote monitoring and control of devices via the internet, with a focus on smart environments such as homes or offices. The study highlights the effectiveness of using IoT modules to create affordable and scalable smart monitoring solutions.

## 8. CARBON NANOTUBE BASED MULTIFUNCTIONAL FLAME SENSOR Mohanty, S., & Misra, A. (2014)

This research explores the use of carbon nanotube-based multifunctional flame sensors. The sensors are designed to detect flames with high sensitivity and low power consumption. The authors propose that such sensors can be integrated into IoT systems to provide early fire detection in a variety of applications, improving safety and response times.

## 9. IOT FOR ENERGY EFFICIENT GREEN HIGHWAY LIGHTING SYSTEMS: CHALLENGES AND ISSUES Mukta, M. Y., Rahman, M. A., Asyhari, A. T., & Bhuiyan, M. Z. A. (2020)

This paper discusses the application of IoT for energy-efficient green highway lighting systems. While primarily focusing on energy efficiency, the paper also touches on the

challenges of integrating IoT into infrastructure systems. The authors propose solutions to optimize lighting based on traffic and environmental conditions, contributing to sustainability efforts.

**10. REVERSIBLE TEMPERATURE-SENSITIVE LIQUID–SOLID**

**TRIBOELECTRIFICATION WITH POLYCAPROLACTONE MATERIAL FOR**

## WETTING MONITORING AND TEMPERATURE SENSING Li, X., Zhang, L., Feng, Y., Zheng, Y., Wu, Z., Zhang, X., & Zhou, F. (2021)

The authors present a novel temperature-sensitive liquid-solid triboelectrification system for wetting monitoring and temperature sensing. This research offers insights into how

triboelectric sensors can be used in IoT systems to detect environmental changes, which can be applied in fire prevention or safety monitoring systems by tracking temperature fluctuations.

## 11. DATA COLLECTION METHODS SERIES: PART 4: DESIGNING FORMS AND INSTRUMENTS Hutchinson, E., & Harwood, E. M. (2009)

This paper is part of a series focused on data collection methods, specifically discussing the design of forms and instruments for effective data gathering. While not directly related to IoT, the study offers valuable insights into how data collection tools can be optimized, which is essential for IoT systems that rely on accurate data input for decision-making processes.

## 12. SOFTWARE DEVELOPMENT LIFECYCLE MODELS Ruparelia, N. B. (2010)

This article outlines various software development lifecycle (SDLC) models, including their strengths and weaknesses. The author compares traditional models such as Waterfall with

more flexible approaches like Agile, providing a comprehensive analysis that is useful for developing IoT systems, which require robust and adaptable software structures.

**13. WATERFALL VS. V-MODEL VS. AGILE: A COMPARATIVE STUDY ON SDLC**

## Balaji, S., & Murugaiyan, M. S. (2012)

The authors present a comparative study of three popular SDLC models: Waterfall, V-Model, and Agile. Their findings help developers choose the right methodology depending on the complexity and requirements of the project. In the context of IoT, the adaptability of Agile might offer the best approach for evolving and iterative system development.

**14. LOW-COMPLEXITY HIGH-PERFORMANCE DEEP LEARNING MODEL FOR**

## FIRE DETECTION SYSTEMS Jadon, A., Varshney, A., Ansari, M. S., & Kant, C. (2020)

This research focuses on the application of deep learning in fire detection systems, particularly in environments constrained by computational resources. The authors propose a low-complexity deep learning model for fire detection, which can be embedded into IoT devices. This offers real-time monitoring and early detection of fire hazards, optimizing safety in various IoT-enabled infrastructures.

## 15. IOT-BASED INTELLIGENT MODELING FOR FIRE PREVENTION AND SAFETY IN SMART HOMES Rehman, A., Saeed, F., Paul, A., & Hong, W. H. (2018)

The paper discusses IoT-based intelligent modeling for fire prevention and safety in smart homes. The authors developed a model that integrates sensors and IoT devices to detect and respond to fire hazards. Their approach enhances traditional fire alarm systems by offering real-time monitoring and increased automation in smart environments.

2.2 Gaps Identified

1. Lack of Connectivity

More specifically, the traditional smoke detectors are standalone systems which are not connected to a wider network or system. This leads to what they refer to as Localized alarms, that is alarms restricted to the said device. This is more so in emergency situations whereby in the event none of the respondents are near the alarm’s vicinity, the cycle will only begin

after, hence quite a number of minutes may be wasted. This delay is however dangerous since it may deny the receiving party an opportunity to contain the situation thus leading to formation of a physical barrier that can result in damage or even injury if an accident

happens. There is no way of sending a signal to those who are not in the vicinity of that

location for example those still out of their homes or in distinct parts of a huge compound. This isolation reduces the efficiency of the smoke detection system in helping to respond to any probable fire risks on time.

1. Limited Alert Mechanisms

Currently marketed smoke detector offer only audible notification such as loud alarm.

Although these alarms are efficient to wake those within vicinity, they may not be sufficient in some occasions. For instance, if the occupants are asleep, in another room or out of the house, the audio alert may not be effective. However, in such cases, there are no more

alerting mechanisms provided such as for visual, for sending alerts to the mobile devices, or

integration with other communication systems, so that the system is able to guarantee that all the concerned parties are alerted timely. This limitation can becomes a drawback of the system especially when quick decision has to be made.

1. No Real-Time Monitoring

There are a number of shortcomings with the conventional smoke detection systems including the following Ones do not provide real-time monitoring or easy remote access. As mentioned earlier, the lack of real time data makes it impossible for a user to access their property’s status while on the move. Users are unable to monitor the fire risks or even the status of the system in order to make a proper decision and take necessary action when needed due to the lack of remote-monitoring. This lack of remote access also raises

difficulties of users being able to check on the operation of the system or receive any notifications on the detected threats and subsequently, may result in users being unaware of the security condition of their property.

1. False Alarms

But one of the most common problems associated with the standard smoke detector is its tendency to give a false alarm. It produces false alarms when it finds that it is in a position to

respond to non-threatening issues such as smoke from cooking, steam from shower, dust among others. This results in alarm fatigue in which individuals and or system become

insensitive to the alerts hence compromising their usefulness. Such complacency may in the long-run reduce the efficiency and effectiveness of this system to such a measure that its

occupants do not react as would be expected to actual fire incidents. One significant problem is that even triggers that are not genuine threats cannot be identified from other challenges,

which erodes the system’s efficiency even more.

1. Lack of Integration

Conventional smoke detection systems are normally standalone and are not connected with other smart home appliances and emergency services. This level of integration does not allow

the system to connect with other technologies including smart home and emergency services.

For instance, there are no default actions to execute upon threat detection including the shutoff of electrical circuits, closure of sprinklers or calling the emergency services. Without such integration, the system cannot effectively respond to fire dangers in an integrated and

automative manner which reduces the efficacy of the safety measures that are in place.

Modern Enhancements

Some of the modern Smoke Detection Systems have been manufactured with some basic connectivity features such as Wi-Fi, or Bluetooth but these technologies are restricted. They may allow simple things such as triggering notifications to mobile devices but normally are inadequate in giving elaborate monitoring, enriched real time alarms, or synchronisation with other smart home solutions. Such systems may also be missing intelligent learning features

that would help to minimize false alarms, based on a clear differentiation between risks and signs of potential dangers as well as other stimuli. Therefore, even though there may be

certains changes in more modern systems, many of these systems experience numerous issues with connectivity and alerting mechanisms, monitoring real time and integration.

2.3 Objectives

1.Enhance Connectivity: Connect smoke detectors with IoT so as to receive alert notifications and monitor the building’s security through the use of mobile applications even when occupants are not present.

2.Reduce False Alarms: Introduce variable control mechanisms that should allow the system distinguish real fire events from other false alarms such as cooking, thereby enhancing system effectiveness.

3.Provide Comprehensive Alerts: Create different modes of alerting that includes short message service, email, and the mobile application that should alert the users and the emergency services no matter the place they are.

4.Enable Smart Home Integration: Make the smoke detection system compatible with other home appliances like the HVAC systems, sprinklers etc, to enable early safety measures like stopping air circulation or turning on the sprinkler in case of threats as detected by the smoke alarm.

5.Offer Scalability and Customization: Make the system adaptable to different areas including residential use, commercial buildings and industries; the ability to vary different aspects of the sensors and alerts used as well as the flexibility of the system in terms of compatibility.

2.4 Problem Statement

The existing technology of smoke detectable has severities and their biggest disadvantage is that they are not able to immediate alert or self-learning. Usually, these systems are network standalone devices with capabilities only to initiate an alarm within the premises. When no

one is around, there are high chances that one will come across a fire breakout and this may lead to lots of property damage or even deaths in the process. Also, such systems can be

triggered by spurious conditions, which include dust, steam, or fumes from cooking. Using examples, the author explains that often false alarms desensitize the users and make them

passive to bona fide threats. It is with these challenges that this project seeks to has provided a solution through the creation of an intelligent smoke detection system based on IoT

technology. With the implementation of the new system, people will be able to monitor and alert them in real time and even when leaving their property. The use of adaptive learning

algorithms will also assist to distinguish actual fire conditions from other non-real situations

resulting in few false alarms and many realistic conditions. This is so as to improve on safety

by issuing time and accurate alerts and also interacts with smart home systems to provide automatic responses. The system will be able to be modified in size and be able to accommodate the different environments, including homes, businesses and industries.

2.5 Project Plan

Phase 1: Research and Requirement

The first step therefore is to document and evaluate currently used smoke detection technologies and the drawbacks associated with them. This includes understanding the

current systems in order to know the areas that are weak as well as the areas of strength.

From this analysis, it will be therefore possible to deduce the right hardware and software necessary for the internet of Thing based Smoke Detection System. A Feasibility study will be conducted in order to establish whether the project is technically and economically feasible while a Risk analysis will be conducted to identify the risks likely to besete and measures to mitigate the risks.

Phase-2: Design and the prototyping

The design and prototyping development phase also decrease but again the costs are split between both the hardware and the software development. Based on aspects to be designed, appropriate hardware will be purchased which include Node MCU, smoke sensors, flame

sensors, power supply units. These sensors will be connected in the circuit and PCB layout in order to facilitate proper interfacing with the microcontroller. On the software side, they will have the use of the embedded software generated in a language termed as Embedded C that will undertake tasks of data acquisition and processing of information from the sensors. Moreover, the cloud infrastructure and the mobile application will be built in order to support real-time monitoring & notification.

Phase 3:Implementation and Development

This phase also encompasses the mechanics of carrying out these detailed development processes especially through project implementation. In the implementation and development

stage, it entails bringing together the hardware components of the assembled system into a working prototype. The actual code to use in Node MCU will be coded, while the cloud-

based platform as well as the mobile application will be designed. To special system accuracy and minimize false alarms machine learning algorithms will be used. The hardware will be interfaced with home intelligent devices and tests will conducted on how the hardware interacts with the software and the cloud services.

Phase 4: Testing and Validation

The evaluation and control phase entails performing unit testing of various components so as to confirm that they work. Final Acceptance testing will be carried out so as to verify

interoperability among the sub system. The system will be exercised in several conditions in order to guarantee that the smoke is detected correctly, and the alerts are issued on time and the smart home integration works as expected. There will be an optimization process of the power consumption as well as the performance.

Phase 5: Deployment and Documentation

In this phase, the real environment, for instance, a home or office environment will be used to disseminate the system. Feedback will also be taken to evaluateorexamine and observe andassess certain performances and, if needed, to fine-tune them. Several user guides, technical descriptions, and installations’ instructions will be provided to the buyers allowing them to understand how the system operates and how to deploy it.

Phase 6: Maintenance and Support

The last process seems to be the maintenance of the implemented system in order to guarantee its efficiency in the long run. There will be supervising of the system to identify the problems and then they will be solved in this way. For improvement measures, feedback from the users will be sought in the attempt to find ways of improving the system in future enhancements and expansion.

# 3.REQUIREMENT ANALYSIS

**Functional Requirement:**

Real-Time Smoke and Gas Detection

Instant Alert Messaging

Adaptive Learning and False Alarm Reduction

Integration with Smart Home Devices

Remote Monitoring and Control

Scalability and Customization

**Non-Functional Requirements:**

Reliability and Accuracy:

Real-Time Data Processing:

Energy Efficiency:

User-Friendly Interface:

Security and Privacy:

Compliance with Standards:

**Hardware Requirement:**

**Node MCU (ESP8266/ESP32):**Central microcontroller for managing sensors and communicating with the cloud.

**Smoke Sensor:** Capable of detecting smoke particles and potentially other hazardous gases.

**Flame Sensor:** Detects flames and light sources within the infrared spectrum, providing an additional layer of fire detection

.**Power Supply:** Reliable power source to keep the system running continuously, with potential battery backup for power outages.

**Software Requirement:**

**Embedded C Programming:** Software for the Node MCU to control sensors, process data, and handle communications with the cloud.

**IoT Platform:** Integration with an IoT platform (e.g., Blynk, Thing Speak) for cloud connectivity, data processing, and alert management.

**Machine Learning Algorithms:** Implemented in the software to reduce false alarms and enhance detection accuracy.

**Mobile Application:** For remote monitoring, alert management, and system control by the

User

# 4.SYSTEM DESIGN

**Hardware Design**

* **Node MCU (ESP8266/ESP32):** This is the main controller that connects to sensors and communicates with the cloud via Wi-Fi. It handles data from sensors and sends alerts.
* **Smoke Sensor (MQ-2):** Detects harmful gases like smoke, CO, LPG, etc., and provides the data to the Node MCU.
* **Flame Sensor:** Detects flame or fire through infrared light and alerts the Node MCU.
* **Power Supply:** A stable power source is essential to keep the system running reliably.
* **Optional Add-ons:**

o **Additional Sensors (Temperature, Humidity):** These sensors can give additional environmental context, improving system accuracy.

**System Design Block Diagram:** Visualize the system as:

**Input (Sensors) → Processing (Node MCU) → Cloud (IoT Platform) → Output (User Alerts)**

**Network Topology:** The Node MCU communicates with sensors locally and connects to the cloud through Wi-Fi. Users receive alerts via the cloud.

1. **Hardware Setup:**

**Connecting the Sensors**: Start by connecting the smoke sensor (MQ-2) and the flame sensor to the Node MCU. This involves connecting the appropriate pins (power, ground, and data pins) from each sensor to the Node MCU. The sensors will use the Node MCU’s GPIO (General Purpose Input/Output) pins to communicate.

**Power Supply**: Ensure a stable power supply for the Node MCU and sensors. You can either use a USB power adapter or a battery pack (depending on your design needs) to power the Node MCU, which will then power the sensors.

**Optional Add-ons:** If you’re including extra sensors like temperature or humidity sensors, connect them similarly to the Node MCU’s remaining GPIO pins.

1. **Programming:**

**Writing the Code in Embedded C:** Write a simple Embedded C code that reads the data from the smoke and flame sensors and interprets the values. The program should constantly monitor the sensor readings and trigger an alert if smoke or fire is detected.

Example logic: If the smoke sensor detects gas concentrations above a certain threshold, the Node MCU sends an alert. Similarly, if the flame sensor detects infrared light (indicating a fire), an alert is sent.

**Wi-Fi Configuration**: Set up the Wi-Fi connection within the code to ensure the Node MCU connects to your local network, allowing it to send data to the cloud or IoT platform for remote monitoring.

1. **Uploading the Code:**

**Uploading Code to the Node MCU:** Once your code is ready, use the Arduino IDE or any other compatible software to upload the code to the Node MCU. This involves connecting the

Node MCU to your computer via a USB cable, selecting the appropriate board and port in the IDE, and uploading the code.

1. **IoT Platform Setup:**

Configuring the IoT Platform (Blynk or ThingSpeak): Set up an account on your chosen IoT platform and create a project or dashboard to display the sensor data. You will also configure the platform to send alerts (SMS, email, or app notifications) when the sensors detect dangerous conditions.

**Linking Node MCU with the Platform:** In your code, integrate the platform’s API so that the Node MCU sends sensor data to the platform. The platform will then display the data in real-time on your dashboard

1. **Testing:**

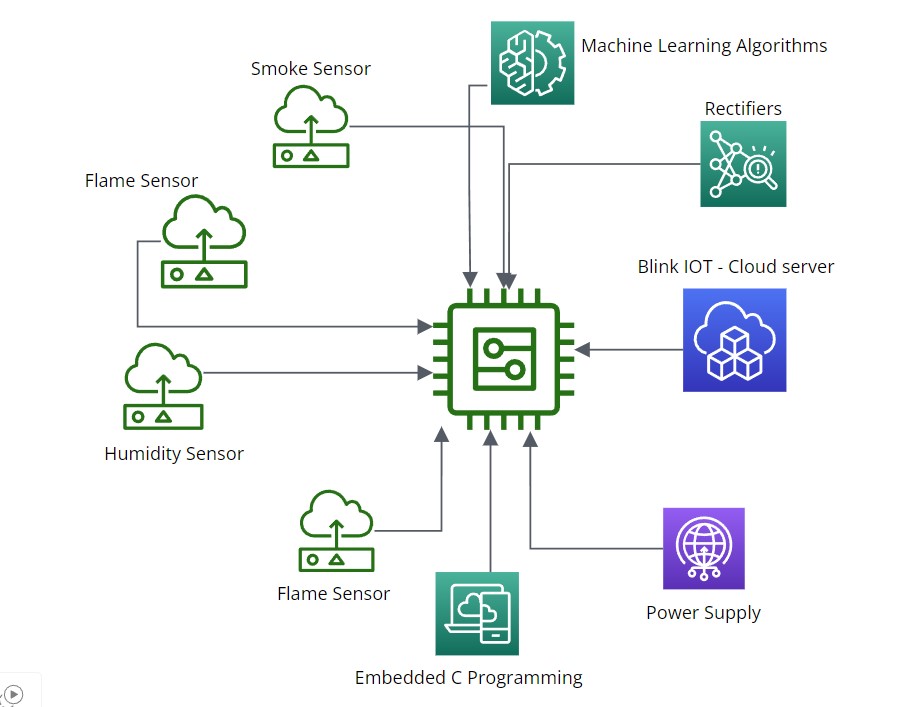
**Sensor Testing:** Test each sensor by exposing them to smoke (e.g., from a burning match) or flame to check if they are detecting correctly. Observe if the Node MCU sends alerts and if the data is displayed on the IoT platform.

**False Alarm Testing:** Run tests to ensure the system doesn’t send alerts in non-dangerous situations, like steam from cooking or a bright light, to finetune its sensitivity.

1. **Deployment:**

**Installing the System:** Once testing is complete, install the system in your home, office, or another environment where you want to monitor smoke and fire risks.

**Monitoring and Adjustments:** Keep an eye on the system’s performance over time, and make adjustments if necessary. This could involve tweaking the sensor thresholds or improving the placement of the sensors for better coverage

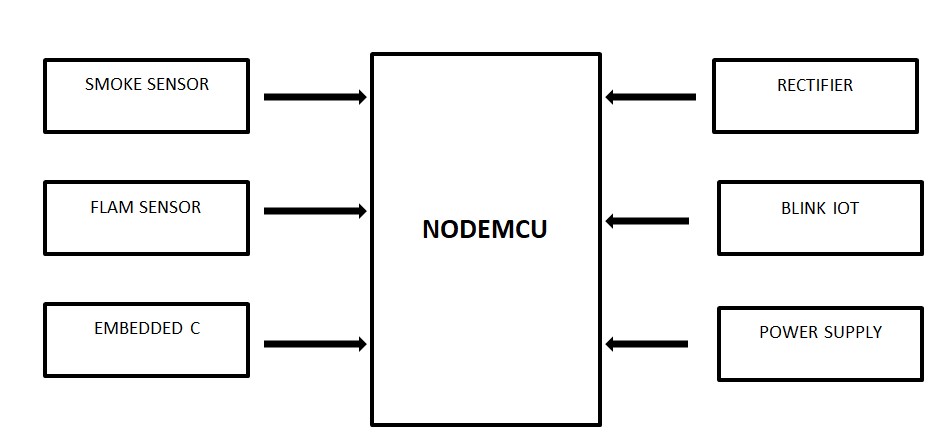


**Software Implementation**

Embedded C for Node MCU: Program the Node MCU to read sensor data and send alerts when abnormal conditions (e.g., smoke) are detected. Write code to handle sensor inputs and connect to the cloud for remote monitoring. IoT Platform Integration: Use platforms like Blynk or Thing Speak to provide a user interface for monitoring and alert notifications.

This allows users to control and monitor the system from a smartphone or computer. MachineLearning: Incorporate basic algorithms to distinguish between real fire events and false positives by analyzing sensor data trends, which helps to reduce unnecessary alerts.

Cloud Services: Platforms like AWS or Google Cloud will store data and handle real-time alerts (SMS, email)



**Machine Learning Process**

**1. Problem Definition**

The primary goal of this project is to classify different types of fire (e.g., electrical, wood, chemical) using real-time sensor data. The system utilizes various parameters such as smoke density, temperature, and humidity to detect fire events and send alerts accordingly. By accurately identifying fire types, the system can facilitate appropriate safety measures tailored to each situation.

**2. Data Collection**

Data is gathered from various sensors integrated with the Node MCU (ESP8266), including:

* **Smoke Sensor (MQ-2):** Measures smoke density and gas concentrations.
* **Flame Sensor:** Detects infrared light emitted by flames.
* **Temperature Sensor (e.g., DHT11):** Monitors ambient temperature.
* **Humidity Sensor:** Provides contextual data that may influence fire behavior.

The data is collected under controlled conditions simulating different types of fires. This includes varying distances, environmental factors, and types of fuel to create a comprehensive dataset.

**3. Data Preprocessing**

Preprocessing is critical for ensuring high-quality input for the machine learning model:

* **Noise Removal:** Filter out anomalous readings from sensors that may arise from environmental factors (e.g., steam, sunlight).
* **Handling Missing Values:** Implement techniques such as interpolation or imputation to manage any missing data points.
* **Normalization:** Scale sensor readings to a consistent range, improving model performance. This can be achieved using techniques like Min-Max scaling or Z-score normalization.
* **Feature Engineering:** Create new features that enhance the model's predictive power:
  + **Smoke Rise Rate:** Calculate how quickly smoke density increases over time.
  + **Time Trends:** Include temporal data such as the time of day or duration since the last detection event, which may influence fire characteristics.

**4. Model Selection**

For this task, a **Random Forest** model is chosen due to its effectiveness in handling complex and nonlinear data, along with its robustness against overfitting:

* **Multi-class Classification:** The model is trained to classify data into multiple categories corresponding to different fire types.

**5. Model Training**

The data is split into training and testing sets (typically a 70-30 split):

* **Cross-Validation:** Implement k-fold cross-validation to ensure the model's generalizability and robustness by evaluating its performance across multiple subsets of the dataset.
* **Hyperparameter Tuning:** Use grid search techniques to identify optimal hyperparameters (e.g., the number of trees, depth of trees) for the Random Forest model, which can significantly improve accuracy.

**6. Evaluation**

To assess the performance of the trained model:

* **Confusion Matrix:** Evaluate true positives, true negatives, false positives, and false negatives for each class.
* **Classification Report:** Analyze precision, recall, and F1-score for a comprehensive view of model performance across different fire types.
* **ROC Curve & AUC:** Use Receiver Operating Characteristic (ROC) curves and Area Under the Curve (AUC) metrics to visualize and quantify the model’s performance at various thresholds, helping to minimize false alarms.

**7. Real-Time Detection**

Once trained and evaluated, the model is deployed in the IoT setup:

* **Continuous Monitoring:** Real-time data from the sensors is fed into the model, which classifies the fire type and triggers appropriate alerts.
* **Alert System:** Notifications (via SMS, email, or app) are generated based on the classification results, allowing for immediate action.

**8. Smart Integration**

The smoke detection system can be integrated with smart home technologies:

* **Automated Safety Actions:** Trigger smart home appliances, such as turning off electrical circuits or activating sprinkler systems, based on detected fire types.
* **User Interface:** Provide users with a dashboard to monitor sensor readings and receive alerts in real time.

**9. Continuous Learning**

To maintain and improve the model's accuracy:

* **Data Updates:** Regularly incorporate new sensor data to retrain the model, adapting it to changes in environment or fire behavior.
* **Feedback Loop:** Implement a feedback system where user reports on false positives/negatives can be used to further refine the model and its parameters.

**5.REFERENCES**

1. A. Mahgoub, N. Tarrad, R. Elsherif, A. Al-Ali and L. Ismail, "IoT-Based Fire Alarm

System," 2019 Third World Conference on Smart Trends in Systems Security and Sustainability (WorldS4), 201

1. Sassani, B. A., Jamil, N., Villapol, M., Abbas Malik, M., & Tirumala, S. S. (2020).

FireNot–An IoT based Fire Alerting System: Design and Implementation. Journal of Ambient Intelligence and Smart Environments, 12(6), 475- 489.

1. Vijayalakshmi, S. R., & Muruganand, S. (2017). Internet of Things technology for fire monitoring system. Int. Res. J. Eng. Technol, 4(6), 2140-2147.
2. Alqourabah, H., Muneer, A., & Fati, S. M. (2021). A smart fire detection system using IoT technology with an automatic water sprinkler. International Journal of Electrical & Computer Engineering (2088-8708), 11(4).
3. Sungheetha, A., & Sharma, R. (2020). Real time monitoring and fire detection using internet of things and cloudbased drones. Journal of Soft Computing Paradigm (JSCP), 2(03), 168-174.

[6]Sharma, A., Singh, P. K., & Kumar, Y. (2020). An integrated fire detection system using IoT and image processing technique for smart cities. Sustainable Cities and Society, 61, 102332.

[7]Saponara, S., Elhanashi, A., & Gagliardi, A. (2021). Real-time video fire/smoke detection based on CNN in antifire surveillance systems. Journal of Real-Time Image Processing, 18, 889-900.

[8]Gupta, A., Srivastava, R., Gupta, H., & Kumar, B. (2020, November). IoT-based fall detection monitoring and alarm system for elderly. In 2020 IEEE 7th Uttar Pradesh Section International Conference on Electrical, Electronics and Computer Engineering (UPCON) (pp.

1-5). IEEE.

[9]Jumaa, N. K., Abdulkhaleq, Y. M., Nadhim, M. A., & Abbas, T. A. (2022). IoT-Based Gas Leakage Detection and Alarming System using Blynk platforms. Iraqi J. Electr. Electron.

Eng, 18, 64-70.

[10]Prabha, B. (2019, December). An IoT based efficient fire supervision monitoring and alerting system. In 2019 Third International conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud)(I-SMAC) (pp. 414-419). IEEE.

[11]Yadav, R., & Rani, P. (2020, November). Sensor based smart fire detection and fire alarm system. In Proceedings of the International Conference on Advances in Chemical Engineering (AdChE).

[12]Salman, A. D., Khalaf, O. I., & Abdulsahib, G. M. (2019). An adaptive intelligent alarm system for wireless sensor network. Indonesian Journal of Electrical Engineering and Computer Science, 15(1), 142-147.

[13]Saponara, S., Elhanashi, A., & Gagliardi, A. (2021). Real-time video fire/smoke detection based on CNN in antifire surveillance systems. Journal of Real-Time Image Processing, 18, 889-900.

[14]Jadon, A., Omama, M., Varshney, A., Ansari, M. S., & Sharma, R. (2019). FireNet: a specialized lightweight fire & smoke detection model for real-time IoT applications. arXiv preprint arXiv:1905.11922.

[15]Valikhujaev, Y., Abdusalomov, A., & Cho, Y. I. (2020). Automatic fire and smoke detection method for surveillance systems based on dilated CNNs. Atmosphere, 11(11), 1241.