MILA UNIVERSITY DU043(N)
[Formerly known as Manipal International University (MIU)]

A FLOOD AND FIRE DETECTION SYSTEM WITH CORRECTIVE ACTIONS

HEWAGE INURIGEEMA MANDULEE

JAYARATHNE

(1104201003)

THIS PROJECT REPORT IS SUBMITTED TO FULFILL THE REQUIREMENTS FOR THE AWARD OF THE DEGREE OF **BACHELOR OF COMPUTER ENGINEERING**

OCTOBER 2023

MILA UNIVERSITY DU043(N)
[Formerly known as Manipal International University (MIU)]

DECLARATION AND COPYRIGHT PAGE

Name : **Hewage Inurigeema Mandulee Jayarathne**

Student ID : **1104201003**

Title of the FYP1 & FYP 2: A Flood and Fire Detection System with Corrective Actions

I hereby declare that this project is the result of my own investigations, except where otherwise stated. Other sources are acknowledged by footnotes giving explicit references and a bibliography is appended.

Signature Date 21/12/2023

MILA UNIVERSITY DU043(N)
[Formerly known as Manipal International University (MIU)]

APPROVAL PAGE

A FLOOD AND FIRE DETECTION SYSTEM WITH CORRECTIVE ACTIONS

BY

Hewage Inurigeema Mandulee Jayarathne

Approved by:

"I hereby declare that I have read this Project and in my opinion this Project is sufficient in terms of scope and quality for the award of the degree of Bachelor of Computer Engineering".

Supervisor:

Date:

Name & Seal: Ir. Ts. Gs. Keith Chong Peng Lean

Examiner 1:	Examiner 2:
Date:	Date:
Name & Seal: Dr. Nor Hafizah Binti	Name & Seal: Mr. Muhammad
Mohamed Halip	Aqmal Ashraf Mohd Bostomi

MILA UNIVERSITY DU043(N)
[Formerly known as Manipal International University (MIU)]

CLEARANCE PAGE

Title of theProject	A Flood and Fire Detection System with Corrective Actions	
	Name of the Student	**I/D Number**
1.	Hewage Inurigeema Mandulee Jayarathne	1104201003

I hereby declare that I have read this report and in my opinion this report is sufficient in terms of scope and quality and met the project objectives. Also, I approve that the plagiarism level of this report is found to be 26% which is close to the limit (20%) and all the similarity contents are re-phrased.

Checked and verified by:

FYP Coordinator

____

Date:

Name & Seal: Dr Hidayah Azamin

Approved by:

Head of the Department

____

Date:

Name & Seal: Ir. Ts. Gs. Keith Chong Peng Lean

ABSTRACT

The goal of the project is to create an Arduino based flood and fire detection system using the concept of internet of things (IoT) with corrective action against flood and fire. The system aims to detect floods and fires in the home and alert homeowners even when they are not at home. The project involves the integration of Node MCU and ESP32-CAM microcontrollers for real-time flood and fire detection. The corrective actions, including drainage pumps and water sprinklers, are triggered upon detection. Furthermore, the system incorporates Telegram messaging for instant alerts and visual confirmation through a dedicated bot. The prototype, designed within budget constraints, is tested for scalability and practicality, ensuring its effectiveness in real-world scenarios. This research showcases a holistic approach to IoT-based emergency management, emphasizing both hardware and software integration for enhanced monitoring and response capabilities. The scope of the project includes designing and building the system, testing it in different scenarios, and improving the system if necessary. The limits of sensors used in the system are studied to ensure their effective use. This system aims to provide an affordable and efficient solution for flood and fire detection in smart homes, with the added benefit of taking corrective action itself.

ACKNOWLEDGMENT

I would like to extend my sincerest gratitude to all those who have supported me thus far in the development of the Flood and Fire Detection System project. First and foremost, I want to express my deepest appreciation to my project supervisor, who has provided invaluable guidance and support throughout the process. Their expertise and knowledge in the field have been instrumental in shaping the direction and progress of the project. I am also immensely grateful to the project coordinator, who has been instrumental in keeping me on track and providing valuable feedback and insights. Their organizational skills and feedback have helped me refine and improve the system's design and functionality. Furthermore, I want to acknowledge the head of the department of Computer Engineering and Science and all the lecturers who have contributed to my knowledge and understanding of relevant concepts and technologies. Their teachings have played a crucial role in the development of the Flood and Fire Detection System. I would also like to express my gratitude to the Dean of the School of Engineering for their support and provision of resources. Their assistance has been vital in acquiring the necessary equipment and materials for the project's implementation. Lastly, I want to acknowledge the unwavering support of my family and friends. Their encouragement and motivation have been crucial in keeping me inspired and focused throughout the project's development. To everyone mentioned above and anyone else who has contributed to the progress of the Flood and Fire Detection System, I am deeply thankful. Your support, guidance, and belief in me and the project have been invaluable. I look forward to continuing the development process with your ongoing support and bringing this system to completion.

CONTENTS

FIGURE INDEX

TABLE INDEX

CHAPTER ONE : INTRODCUTION

1.1 Overview

Experiencing a fire or flood can be a traumatic event that can cause significant mental stress for those affected. The aftermath of such events can be overwhelming, with people having to deal with the loss of property, displacement from their homes, and potential injuries or fatalities. In addition to the immediate impact of the event, people may also experience long-term psychological effects such as anxiety, depression, and post-traumatic stress disorder (PTSD) [1] [2]. Extreme weather events have increased around the world in recent years, increasing the frequency and severity of floods. Additionally, an accidental fire can break out at any time and the consequences can be devastating.

Having effective fire and flood safety systems in place can help reduce the risk of these events occurring and minimize their impact if they do occur. Knowing that one's home is protected by reliable safety systems can provide peace of mind and reduce anxiety related to potential disasters.

Modern early warning systems have come a long way from the primitive warning signals of bonfires and wind instruments. As technology has advanced, so too have the capabilities of early warning systems to warn vulnerable people more quickly and reliably. From the use of mechanical warning devices in the 18th century to this development of the electromechanical siren in the 20th century, early warning systems have become increasingly centralized and sophisticated [5]. Fire protection systems have been in use for centuries, with early examples such as bucket chains and hand pumps. Today's modern fire protection systems are increasingly electronic driven; use a combination of sensors, alarms, and suppression systems to rapidly detect and extinguish fire and rely on faster and more secure data transmission.

Flood safety systems have also been used for centuries, with early examples such as levees and dams. The 20th century saw the development of more sophisticated flood control measures such as reservoirs and drainage systems [4]. Today's modern flood safety systems combine sensors, alarm systems and evacuation plan to protect people from flooding.

Flood and fire alarm systems based on Arduino technology and IoT help homeowners effectively identify and respond to potential hazards in their homes. Some of the hazards that can occur in homes are flooding and fire, which can cause serious property damage and even endanger human life. In the event of flooding, the system can detect rising water levels and turn on the alarm, send real-time alerts to the homeowner's mobile his device and as well as the surveillance for the event. In the event of a home fire, the system can detect the presence of heat and turn on the alarm, send real-time alerts to the homeowner's mobile his device and as well as the surveillance for the event. The proposed system be using Node MCU, ESP32 Camera module to enable camera view by taking snapshot by a telegram chat bot, flood detection using flood level sensors and fire using fire sensors and upon detection shall alert the user app alert using blynk app and trigger the house alarm. Additionally, the system itself can take corrective action such as water sprinkler and water drainage pump. One of the main benefits of this system is that it gives homeowners peace of mind by proactive approaches to preventing damage to their homes and property, allowing them to not be burdened by insurances, loss of lives and property damage. This feature is especially important for homes that may be absent in an emergency, allowing the system to act when no one is there. In summary, his Arduino-based IoT flood and fire alarm system is an essential investment for homeowners looking to protect their homes and property from potential hazards. With real-time notifications and the ability to take self-remedial actions, this system provides homeowners with valuable information and peace of mind. Additionally, the system is highly customizable, cost- effective, and easy to use, making it an ideal solution for a wide variety of homeowners.

1.2 Problem Statement

The problem is that floods and fires can occur unexpectedly and spread rapidly, causing serious damage to homes and endangering the lives of residents. The speed and intensity of these disasters make it difficult for homeowners to take corrective action quickly enough to mitigate the damage. Fires can release toxic gases and cause explosions, further endangering residents. Therefore, an efficient and affordable IoT-based flood and fire alarm system that can detect these disasters in real time, notify homeowners while enabling the surveillance for the event, and take corrective action to mitigate the damage caused is required.

1.3 Objectives

i. To build a fire detection system using IoT which would detect the fire, send an emergency alert, trigger the house alarm as well automatically take corrective actions to prevent fire from spreading by automating water sprinkler system.
ii. To create a flood detection system using IoT which would detect the flood water level and send an emergency alert, trigger the house alarm as well automatically take corrective actions to pump out water by turning on the water draining pump.
iii. To enable a camera view for surveillance purposes for both flood and fire detection.

1.4 Scope of Project

This proposed system will be using a microcontroller and external camara to view flood and fire scene upon flood detection using flood level sensor and fire using fire sensor. Upon detection shall alert the user App alert using an app and trigger the house alarm with the corrective actions to put off the fire and divert the water away using barrier. The flood will be detected by each water level sensor due to the increase in the water level.

The system can be broken down for 4 units and the scope of the work for each unit follows:

1. **Fire Detection System**
   - 3 Flame sensors will be used.
   - The alarm goes ON
   - An app alert will be sent to the user’s device.
   - Water sprinkler turns ON
2. **Flood Detection System**

3 water level sensors will be used to detect the increasing water level by 3 levels.

- Level 1 (18 cm bellow from the ultrasound sensor) -

  A pop-up notification to the user's device will be sent. Alarm goes ON.

- Level 2 (16 cm bellow from the ultrasound sensor) -

  A pop-up notification to the user's device will be sent. Draining Pump ON.
  Alarm goes ON.

- Level 3 (14 cm bellow from the ultrasound sensor) -

  A pop-up notification to the user's device will be sent. Draining Pump ON.
  Alarm

goes ON. When the water level is decreasing:

- Level 3 (14 cm bellow from the ultrasound sensor) -

  Draining Pump ON.

- Level 2 (16 cm bellow from the ultrasound sensor) - Draining Pump ON.
- Level 1 (18 cm bellow from the ultrasound sensor) - Draining Pump ON.
- Ground Level/ Threshold Level (22 cm below from the ultrasound sensor) - Draining Pump OFF.

3. **ESC 32 Camera Module**
   - The user can surveillance the area to make sure the app alert is true.
4. **Blynk App**
   - To get the pop-up notification to each flood and fire detection according to the specific condition.
5. **Chat Bot in telegram**
   - To create a chat bot in Telegram app is used to receive snapshots when the user is requested.

1.5 Significance of Study

The importance of this research lies in revolutionizing home security through Arduino technology and IoT-controlled flood and fire alarm systems. These systems seamlessly integrate sensors and monitoring capabilities to help homeowners proactively respond to potential threats such as flooding and fire. During floods, the system instantly detects rising water levels, activates alarms, sends real-time alerts to homeowners' mobile devices, and captures monitoring snapshots. Similarly, in the event of a house fire, the system detects heat, activates an alarm, sends an immediate alert, and captures a snapshot of the incident. The Node MCU, ESP32 camera, and Telegram chatbot make the system more versatile. This study pioneers a comprehensive protection mechanism for homes by using Blynk app alerts and home alarm triggers in combination with corrective measures such as watering and pumping. Ultimately, this proactive technology not only prevents property damage, but also provides homeowners with valuable peace of mind, especially when they are out and about in an emergency.

1.6 Organization of Thesis

The work is well-organized, starting with Chapter 1, which provides the background of the research, problem statement, purpose, scope, significance, and overall organization of the work. Chapter 2 deals with existing knowledge based on 10 literature reviews and lays the foundation for the subsequent methodology. Chapter 3 systematically describes the project approach and presents block diagrams, flowcharts, and components. Chapter 4, Results and Discussion, reveals the results of the project and describes the problems and solutions encountered. The final chapter, Section 5, concludes the study by summarizing the results, discussing limitations, and suggesting future work. Other elements include reference materials, budget breakdowns, Gantt charts, and appendices with relevant code to enhance the completeness of your work.

CHAPTER TWO : LITERATURE REVIEW

2.1 Overview

Literature review is an important part of any research project that involves reviewing existing literature, research papers, books, and other relevant sources of information to assess and summarize existing knowledge in a particular area of research. It aims to identify gaps and limitations in current literature and to highlight important findings, concepts, and theories relevant to the research question. In this chapter, 10 study papers have been taken in reviewing to determine basis for research and indicate the state of knowledge and the need for further research in the field and provide a framework for assessing the impact of the research on the existing body of knowledge.

2.2 Theorical Review

2.2.1 Fire Detection

Fire detection refers to the process of identifying the presence of fire or smoke in a given area. It is crucial for early warning and prompt response to minimize property damage, injuries, and loss of life. Fire detection systems use various technologies and sensors to detect signs of fire, including smoke detectors, heat detectors, flame detectors, and gas detectors. These systems can be standalone devices or part of a larger fire alarm system that includes alarms, control panels, and notification devices. Once a fire is detected, the system triggers an alert, such as sounding an alarm or activating sprinklers, and may also notify emergency services for immediate response. Efficient fire detection plays a vital role in ensuring safety and mitigating the impact of fires in residential, commercial, and industrial settings. In general, fire detection involves the use of sensors and technologies to identify the presence of fire or smoke. The primary objective is to detect fires as early as possible to initiate appropriate responses and mitigate the potential risks. Fire detection systems typically include different types of detectors that are strategically placed throughout a building or area. Smoke detectors are the most common type of fire detectors and use optical, ionization, or photoelectric sensors to detect the presence of smoke particles. Heat detectors monitor the temperature and can trigger an alarm when it exceeds a certain threshold. Flame detectors are designed to detect the presence of flames by sensing their optical characteristics. Gas detectors can identify the presence of specific gases associated with fires, such as carbon monoxide or natural gas. Flame detection can be done typically utilizing either ultraviolet (UV) or infrared (IR) sensing techniques to detect the unique light signatures emitted by flames.

2.2.2 Flood Detection

Flood detection refers to the process of identifying the occurrence or likelihood of a flood in a specific area. It involves the use of various methods and technologies to monitor water levels, precipitation, and other relevant parameters to detect and anticipate potential flooding events.

Flood detection systems typically utilize sensors, gauges, and monitoring devices strategically positioned in areas prone to flooding, such as rivers, streams, or coastal regions. These sensors continuously measure and collect data on water levels, rainfall intensity, flow rates, and other variables that indicate the potential for flooding. The collected data is analyzed in real-time or periodically to identify abnormal patterns or threshold exceedances that may indicate an imminent or ongoing flood event. Sophisticated algorithms and modeling techniques may be employed to process and interpret the data, providing accurate and timely information about the flood conditions. Flood detection systems can generate alerts, warnings, or alarms when predefined thresholds or criteria are met, triggering appropriate responses from emergency management agencies, local authorities, or residents in the affected areas. These responses may include evacuation procedures, activation of flood control measures, and dissemination of timely information to ensure public safety and minimize damage. The primary goal of flood detection is to provide early warning and actionable information to mitigate the impact of floods. By promptly identifying and communicating flood events, authorities can make informed decisions, implement effective emergency measures, and allocate resources efficiently to protect lives and property. Flood detection is an essential component of flood management and preparedness, complementing other strategies such as flood forecasting, floodplain mapping, and infrastructure development to mitigate the risks associated with flooding and enhance resilience in flood-prone areas.

2.3 Previous Work Review

2.3.1 Prediction and Effective Monitoring of Flood Using Arduino System Controller and ESP8266 Wi-Fi Module

D. Dinesh and I. Anette Regina (2019) [1] designed a cost-effective system focused on flood prediction and effective monitoring using an Arduino system controller and an ESP8266 Wi-Fi module. The proposed system uses Arduino Uno controller works with temperature sensor, rain fall sensor, and humidity sensor to manage, monitor, display and alert the flood forecast and warnings as and when required using the advantage of cloud service which make the data can be accessed from anywhere through ThingSpeak website with the help of ESP8266 Wi-Fi module. In this paper, an inexpensive miniature prototype radio frame has been proposed for flood warning stations. The system is designed to solve the timeliness problem of SAR data using sensor modules. It was found that the designed system allows pre-programming of the controllers used to monitor the flood system. The main advantage of this model is the reduction of hardware components in this system.

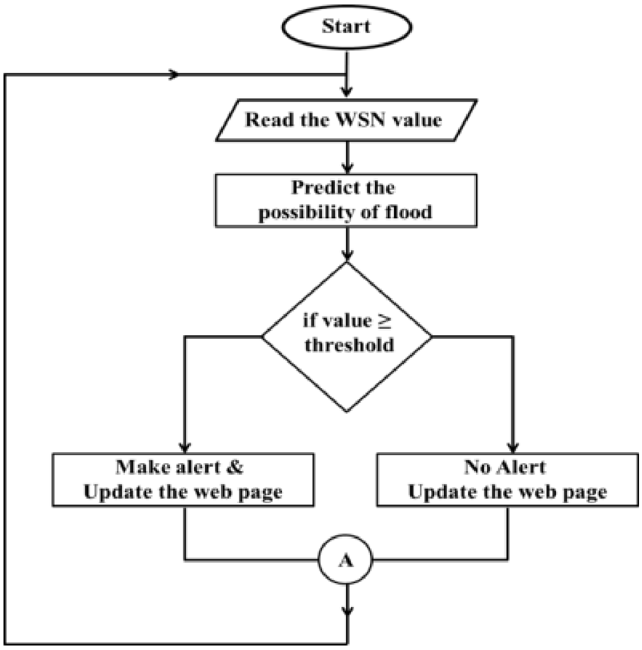


Figure 2.1: Flowchart mentioned in D. Dinesh and I. Anette Regina (2019) [1]

2.3.2 Arduino Based Smart Home Warning System

Qusay Idrees Sarhan (2022) [2] has been presented a smart home warning system that uses an Arduino Uno microcontroller and multiple compatible sensors and actuators to efficiently detect fire, gas leak, and intrusion situations. The instruments used in the smart home warning system are Arduino Uno microcontroller, GSM module, MQ2 gas sensor, Flame sensor, PIR motion sensor, DHT22 temperature and humidity sensor, Buzzer, LED bulb, Solenoid valve and Ventilation fan. The system can send notifications to users via GSM radio communication, send SMS messages, emails with photos attached, calls the owner and also warn the owner/residents by sounds a buzzer and flashes a light bulb in an event of a danger. The system also allows homeowners to take proper actions such as stopping fire via water and decreasing gas concentration in the air via a fan. The system also continuously captures images and saves them in the MicroSD card module to send to the owner. The proposed system is very useful in preventing robbery by detecting movements by thieves. Overall, this study presents a comprehensive solution for smart home security that is affordable, easy to implement, and highly effective.

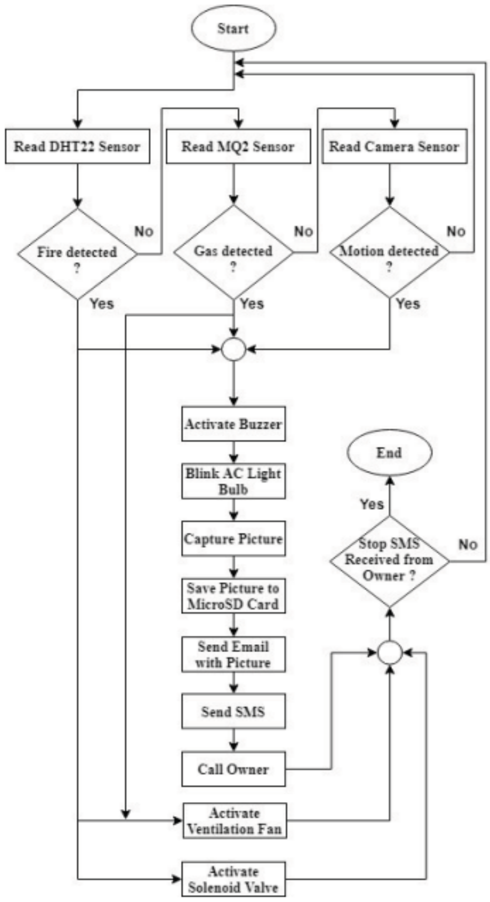


Figure 2.2: Flowchart mentioned in Qusay Idrees Sarhan (2022) [2]

2.3.3 Arduino Based Fire Detection and Control System

Muhammad Shazali Dauda and Usman Saleh Toro (2020) [3] has been presented an Arduino-based fire detection and control systems that is inexpensive for safety and accessible to users of all levels. The to automatically detect heat in certain environments, sound an alarm, turn off buildings, and spray water to reduce the intensity of fires. The system uses a DHT 11 sensor, a buzzer, a 5V DC (direct current) motor, a GSM (Global System for Mobile) module sim800l for sending SMS (Short Message Service), and a 16X2 LCD screen and Atmeg328p microcontroller. The objectives of this project were met, and the system worked effectively. The system continuously monitors the presence of large amounts of heat and activates an alarm, simultaneously shutting down the building's electrical network and sending an SMS

(Short Message Service) alert to take safety measures to contain the situation. extinguish the fire as the proposed system is unique in that it uses an Arduino microcontroller and various sensors to detect and control fire initiation.

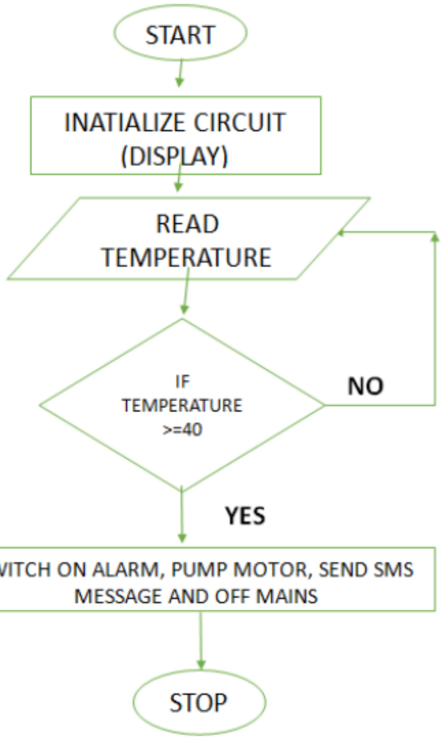


Figure 2.3: Flowchart mentioned in Muhammad Shazali Dauda and Usman Saleh Toro (2020) [3]

2.3.4 Arduino Based Smart Home Automation System

Ma Naing and Ni Ni San Hlaing (2019) [4] designed a system to run on both AC and DC power and uses a hybrid power supply. The two Arduino NANO boards are used to obtain values of physical conditions through sensors connected to them. The temperature sensor reads temperature values, the smoke sensor detects smoke by sending SMS alarms and ringing the buzzer, and the Light Dependent Resistor (LDR) controls automatic switching on and off of the light based on daylight intensity. A motion detector is also integrated using Passive Infrared Sensor (PIR) to detect movement for security purposes. Figure 2.1.4 also shows various other components such as a GSM module, LCD displays, relays, fans, bulbs, and a servo motor that can be controlled by the system.

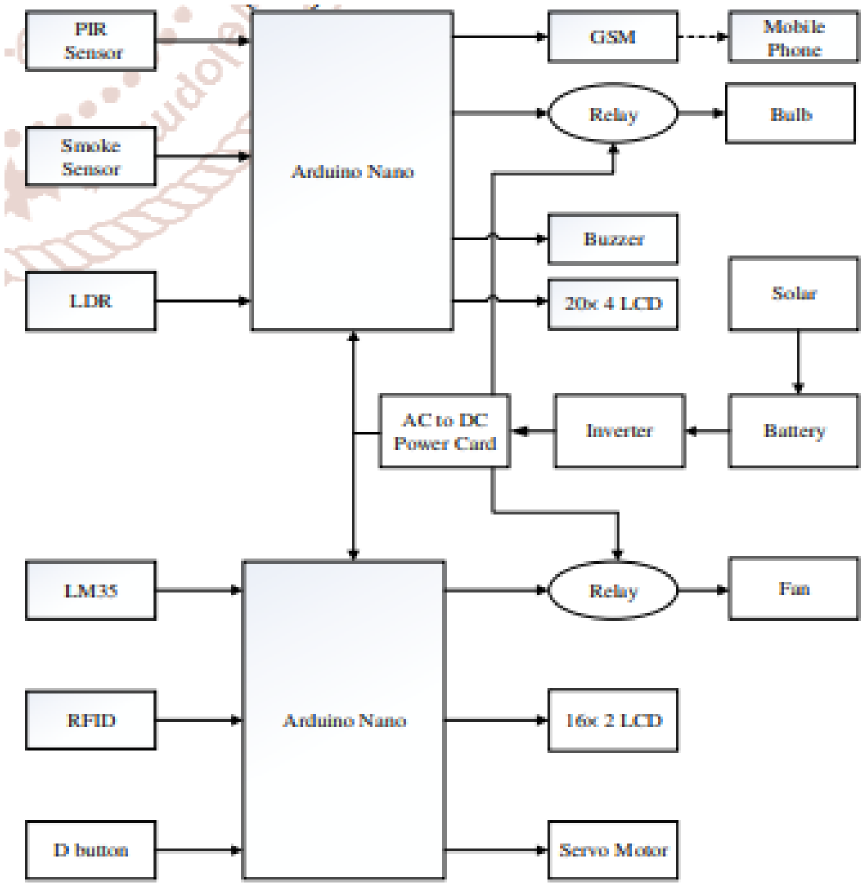


Figure 2.4: Flowchart mentioned in Ma Naing and Ni Ni San Hlaing (2019) [4]

2.3.5 Design and Implementation of Security Systems for Smart Home based on GSM Technology.

Jayashri Bangali and Arvind Shaligram (2013) [5] design a system consists of a sensor, an Atmega644p microcontroller, a sim548c GSM module, a buzzer, an in-system programmer, and relays to control the device. The sensors of the system detect intrusions and dangerous situations such as gas leaks and fires. The Atmega644p microcontroller collects information from sensors and uses a sim548c GSM module to send SMS notifications to the homeowner's preferred number. A buzzer is used to give an audible alarm in the event of an intrusion. Relays are used to remotely control devices such as lights and fans via SMS commands. In the system, all sensor outputs are connected to an ADC. One IR is connected to Windows and another IR sensor is in front of the door. Entering the room through the window will be treated as unauthorized entry as well as entry from door is treated as authorized entry. If the access to the house is permitted LED light will be turned on the switch after checking the illuminance of the room, and sound the buzzer will turn on in case of unauthorized entry. If the temperature is high (above 45 degrees), monitor the temperature continuously.

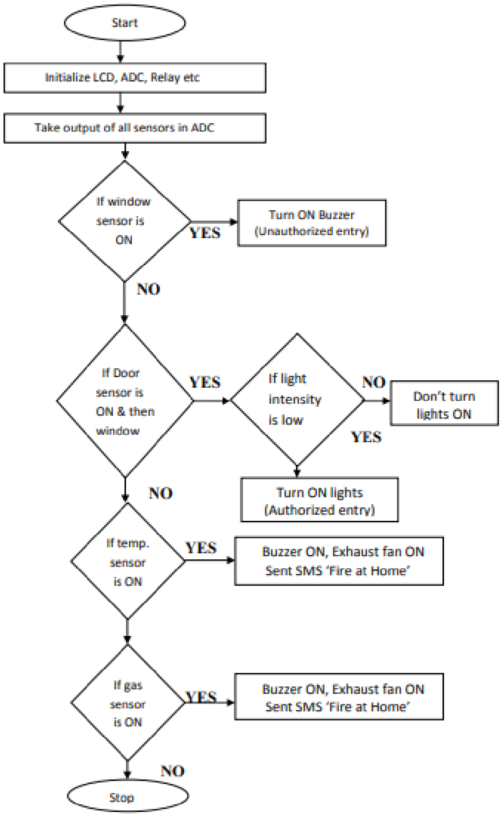


Figure 2.5: Flowchart mentioned in Jayashri Bangali and Arvind Shaligram (2013) [5]

2.3.6 Prototype of Google Maps-Based Flood Monitoring System Using Arduino and GSM Module

Dedi Satria, Syaifuddin Yana, Rizal Munadi, and Saumi Syahreza (2017) [6] designed a flood early warning system using Google Maps and Arduino technology. The system uses sensors to detect the water level and send alerts to users via SMS. The study highlights the potential benefits of implementing this system in flood-prone areas, such as faster response times and reduced damage. Overall, this study represents a prototype flood monitoring system that may improve disaster management in flood-prone areas. The flood monitoring system discussed in this study uses an ultrasonic sensor as a height detector, an Arduino Uno as a processor, a U-Blox Neo 6m GPS module and a GSM module as a water level transmitter and collects flooded information. Coordinate the system station.

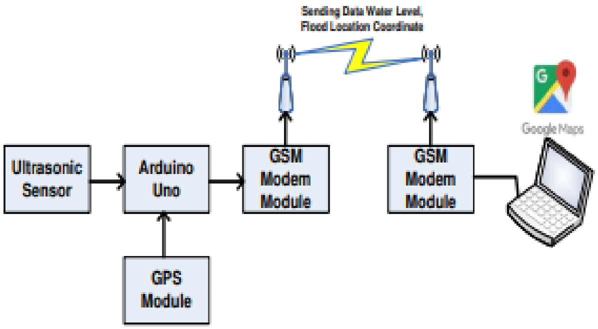


Figure 2.6: Flowchart mentioned in Dedi Satria, Syaifuddin Yana, Rizal Munadi, and Saumi Syahreza (2017) [6]

2.3.7 Quick Fire Sensing Model and Extinguishing by Using an Arduino Based Fire Protection Device

Md. Rawshan Habib, Naureen Khan, Koushik Ahmed, Mahbubur Rahman Kiran, Mohaiminul Islam Bhuiyan, and Omar Farrok (2019) [7] proposed an Arduino-based automatic fire alarm system with fire extinguisher for fire prevention. The proposed device uses mathematical models to represent the thermal properties of the house in which it is installed, the external environment and its heating system. A cost function for maintaining the conditioned environment is also considered. The temperature control system set point is 27°C in winter. This study emphasizes the importance of fire protection for safety purposes and proposes this device as a solution to prevent serious accidents due to mishandling of fire sources. The proposed fire protection system uses several functional sensors, such as smoke detectors, temperature sensors and flame sensors. The system also includes a microcontroller and sensor unit, a fire alarm, a motor and water pump for the fire system, a 12V step-down transformer, a bridge rectifier, and filter capacitors for the power system. Also, the system uses a regular push-button phone, which presses the call button three times to call the owner and pressing is done with a servo motor.

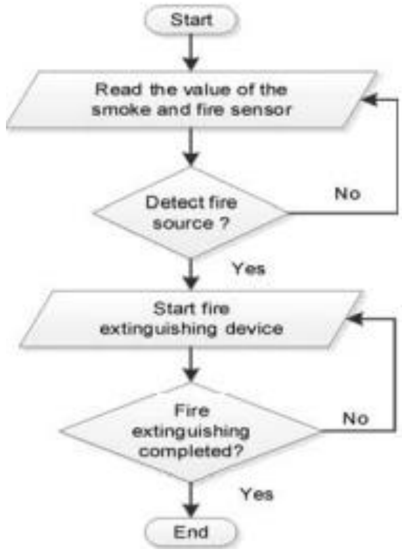


Figure 2.7: Flowchart mentioned in Md. Rawshan Habib, Naureen Khan, Koushik Ahmed, Mahbubur Rahman Kiran, Mohaiminul Islam Bhuiyan, and Omar Farrok (2019) [7]

2.3.8 Flood Early Warning Detection System Prototype Based on IoT Network

Joni Welman Simatupang and Faiz Naufal (2019) [8] presents a prototype of a flood early warning system based on IoT networks. The system uses an ultrasonic sensor device to measure the flood in real time, an Arduino UNO to collect the data, and a SIM900 module to send it via SMS to a central server. The system is intended to deliver early warning messages to measurement point managers, who can distribute data to the population. This study compares this system with other similar systems developed by researchers in the past and highlights its strengths and weaknesses. Overall, this research provides insight into how his IoT technology can be used for early flood detection and warning systems. The Flood Early Warning Detection System prototype is built using several components, including Arduino UNO, Ultrasonic sensor, SIM900

GSM/GPRS module, Breadboard and jumper wires, Power supply (9V battery or adapter) and a Cloud server. The ultrasonic sensor is used to measure the water level, while the Arduino UNO collects and processes the data from the sensor and sends it to the cloud server via SMS using a SIM900 GSM/GPRS module. The cloud server stores and processes the data, which can be accessed by users through their smartphones or other devices.

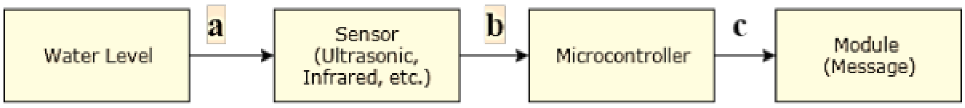


Figure 2.8: Flowchart mentioned in Joni Welman Simatupang and Faiz Naufal (2019) [8]

2.3.9 SMS Based Flood Monitoring and Early Warning System

Sheikh Azid, Bibhya Sharma, Krishna Raghuwaiya, Abinendra Chand, Sumeet Prasad, and A Jacquier (2015) [9] describe a study of a design and implement an SMS-based flood monitoring and early warning system. The system uses an Arduino microprocessor connected to a GSM modem and pressure sensor to measure water level. The system will send timely alerts to endangered or threatened population groups and responsible authorities via SMS. The study also discusses the advantages of using SMS-based systems for flood monitoring and early warning compared to other methods. A potential problem identified in the investigation is the inability of the GSM module to upgrade itself when network operators make changes to the network. The components used in the system include Arduino microprocessor, GSM modem, Pressure sensor, Aluminum box to house the circuit components, External support such as a column of a bridge or a dedicated concrete support, Solar battery charging system (to make the system independent), SIM card (for GSM module), Wires and connectors for circuitry, resistors, capacitors, and diodes for circuitry.

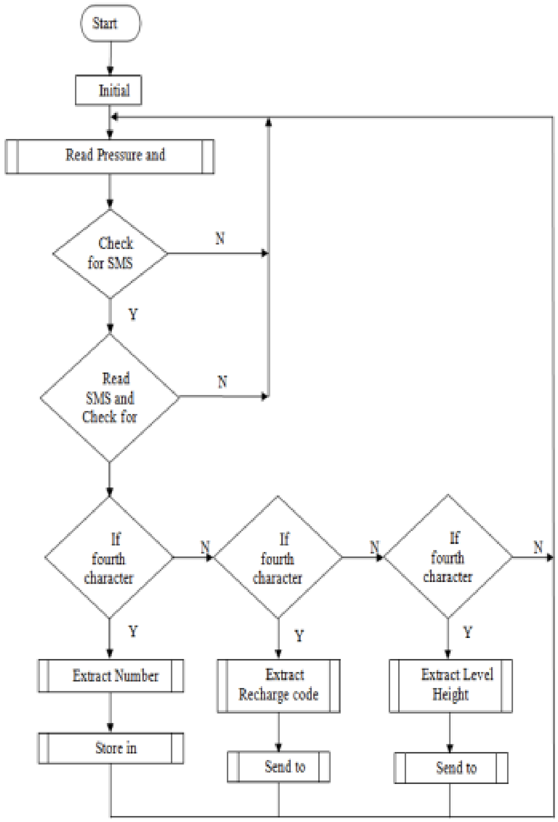


Figure 2.9: Flowchart mentioned in Sheikh Azid, Bibhya Sharma, Krishna Raghuwaiya, Abinendra Chand, Sumeet Prasad, and A Jacquier (2015) [9]

2.3.10 Design of a Home Fire Detection System Using Arduino and SMS Gateway SMS

Suwarjono Suwarjono, along with Izak Habel Wayangkau, Teddy Istanto, Rachmat Rachmat, Marsujitullah Marsujitullah, Hariyanto Hariyanto, Wahyu Caesarendra, Stanislaw Legutko, and Adam Glowacz (2021) [10] designed and implemented a fire alarm system using an Arduino Uno microcontroller and an SMS gateway. The flow of the system is divided into four stages: fire detection, data processing, SMS sending and alarm. The authors used a flame sensor to detect fires and a GSM module to send his

SMS alerts to the homeowner. The system was tested in a real-world environment, and the results showed that it worked as expected with 10 successful attempts to send SMS and trigger alarms. They have used components such as Arduino Uno R3 Atmega328p microcontroller board, DS18B20 temperature sensor, MQ2 gas sensor, Sim900 GSM module, Active buzzer 5 V–12 V, Adapter 12 V–1 A and Alkaline Battery 9 V in the design.

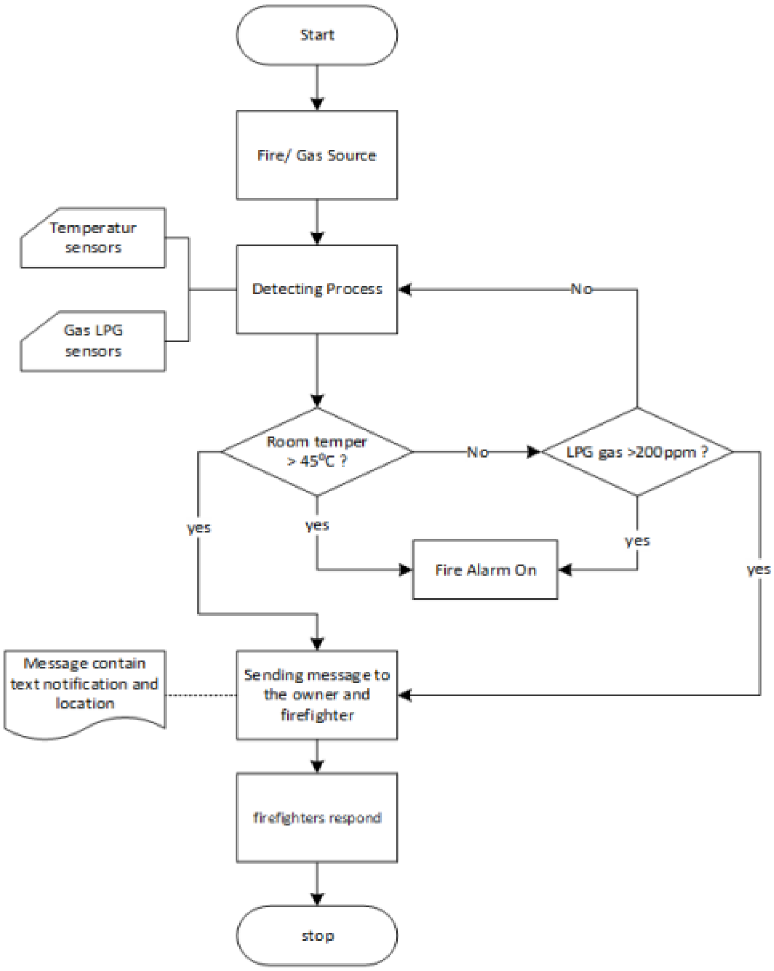


Figure 2.10: Flowchart mentioned in Suwarjono Suwarjono, along with Izak Habel Wayangkau, Teddy Istanto, Rachmat Rachmat, Marsujitullah Marsujitullah, Hariyanto Hariyanto, Wahyu Caesarendra, Stanislaw Legutko, and Adam Glowacz (2021) [10]

2.4 Summary of Literature Review.

The summary of the literature review is shown in Table 2.1

Table 2.1 Summary of literature Review.

No.	Author (s) & Year	Title of Research Paper	Variables Studied/ Research Design	Equipment/ Instruments/ Apparatus used for Experiments/ Analysis/ Characterization, etc.	Important Findings	Limitations of Study	ResearchGap/ Novelty of ResearchStudy
[1]	D.Dinesh, I.Anette Regina 2019	Prediction and Effective Monitoring of Flood Using Arduino System Controller and ESP8266 Wi-Fi Module	The sensing unit contains temperature, rainfall, and humidity sensors, and this unit transfers the sensor value to the Arduino and microcontroller unit based on the threshold value. The Arduino and microcontroller send the signal to the communication unit, which updates the information in internet sources using ESP 8266 Wi-Fi module. The system displays and alerts the flood forecast and warnings as and when	1. Power supply 2. Temperature sensor 3. Rainfall sensor 4. Humidity sensor 5. Arduino Uno controller 6. Relay driver 7. LCD display 8. ESP 8266 Wi-Fi module	The system is designed to solve the timeliness problem of SAR data using sensor modules. It was found that the designed system allows pre-programming of the controllers used to monitor the flood system. The main advantage of this model is the reduction of hardware components in this system.	The system is only limited to measure. parameters like Temperature, Rainfall and Humidity.	It uses cloud technology by using ESP 8266 Wi-Fi module to update the information in cloud. Doesn’t include with fire detection system with the surveillance of a camara. The system doesn’t incorporate a corrective action unit to divert the flood away.

			required using the advantage of cloud service which makes the data can be access from anywhere through ThingSpeak website.				
[2]	Qusay Idrees Sarhan 2022	Arduino Based Smart Home Warning System	The project uses an Arduino Uno microcontroller and various sensors and actuators to create a smart home alarm system that can detect fires, gas leaks, and intrusions. The system has different types of alerts and notifications, such as sending SMS messages, emails with photos, calling owners, playing buzzers, and flashing light bulbs. Only when a fire or gas leak is detected will the fan operate to remove smoke and leaked gas. A solenoid valve operates and stops in the event of a fire. The system continuously captures images and saves them in the MicroSD card module to send to the owner. The system can only	1. Arduino Uno microcontroller 2. GSM module 3. MQ2 gas sensor 4. Flame sensor 5. PIR motion sensor 6. DHT22 temperature and humidity sensor 7. Buzzer 8. LED bulb 9. Solenoid valve 10. Ventilation fan	The system is highly effective in detecting fire, gas leakage, and housebreaking situations using multiple sensors and actuators working together. It also allows homeowners to take proper actions such as stopping fire via water and decreasing gas concentration in the air via a fan.	Other potential threats to home security such as flooding, or carbon monoxide poisoning are not being considered. while the proposed system is designed to be easy to implement, it may still require some technical expertise to set up and maintain. GSM which may not be available or reliable in all areas.	It integrates multiple sensors and actuators to efficiently detect fire, gas leak and intrusion situations. Use SMS messages, emails with attached images, owner calls. A solenoid valve operates and stops in the event of a fire. The system continuously captures images and saves them in the MicroSD card module to send to the owner. It lowers the gas concentration in the air with a fan. It doesn’t integrate with a system to trigger in a flood.

			be stopped when the homeowner sends an SMS message.				. It doesn’t include the surveillance of a camara.
[3]	Muhammad Shazali Dauda, Usman Saleh Toro 2020	Arduino Based Fire Detection and Control System	An Arduino based fire detection and control system designed to automatically detect heat in each environment, sound an alarm, switch off mains of the building, and spray water to reduce the intensity of fire. The system uses a DHT 11 sensor, a buzzer, 5v DC (Direct Current) motor, a GSM (Global System for Mobile) Module sim800l to send SMS (Short Message Service), and LCD screen 16X2 and Atmeg328p Microcontroller. The system continuously monitors the	1. DHT 11 sensor 2. Buzzer. 3. 5v DC motor 4. GSM Module sim800l 5. LCD screen 16X2 6. Atmeg328p Microcontroller 7. Power Supply Unit (PSU)	The system was able to quickly detect and alarm a fire outbreak, shut down the building's power grid, send an SMS alert, and extinguish the fire with water from a tank. Overall, it is found that the proposed fire detection and control system effectively achieves the objective of providing a cost-effective solution for detecting and controlling fire outbreaks in buildings.	the system was tested in a controlled environment, and its effectiveness in real-world scenarios may vary depending on various factors such as the size of the building, the intensity of the fire outbreak, and other environmental conditions.	Extinguish fires using water from a tank. The paper also presents a detailed description of the software design of the system, which can be used as a reference for future research in this field. It doesn’t integrate with a system to trigger a flood. It doesn’t include the surveillance of a camara.

			presence of significant amounts of heat and activates an alarm, simultaneously switches off the mains of the building, sends a Short Message Service (SMS) alert, and extinguishes the fire as a safety measure to contain the situation.				
[4]	Ma Naing, Ni Ni San Hlaing 2019	Arduino Based Smart Home Automation System	The system is designed to operate on AC and DC power and uses a hybrid power supply. Two Arduino NANO boards are used to obtain physical state values via sensors connected to them. The temperature sensor reads temperature readings, and the smoke sensor detects smoke and sends SMS alerts and sounds a buzzer. A light dependent resistor (LDR) controls the lights to turn on/off automatically based on daylight levels. A passive infrared (PIR) sensor motion	1.PIR Sensor 2.Smoke Sensor 3.LDR 4.LM35 5.REID 6.D Button 7.Arduino Nano 8.GSM 9.2 Relay 10.Buzzer 11.20 x 4 LCD 12.AC to DC Power Card x 2 13.Inverter Bulb 14.Solar 15.Battery 16.Fan 17.16 x 2 LCD	The system was designed to monitor and control various home appliances such as lights, fans, and temperature based on signals from related sensors. The paper reports that all tasks of the system were done successfully, but there were limitations in time and expenses.	Limitations in time and expenses.	The system also includes SMS alarm functions that can alert users in case of power supply failure or smoke detection. The system can be run by both AC and DC power. It doesn't integrate with a system to trigger a flood. It doesn't include the surveillance of a camara. The system doesn't incorporate a fire extinguishing unit.

			detector is also integrated to detect movement for security reasons. The diagram also shows various other components that can be controlled by the system, such as GSM modules, LCD displays, relays, fans, light bulbs, and servo motors.				
[5]	Jayashri Bangali and Arvind Shaligram 2013	Design and Implementation of Security Systems for Smart Home based on GSM technology	One IR is connected to Windows and another IR sensor is in front of the door. Entering the room through the window will be treated as unauthorized entry as well as entry from door is treated as authorized entry. If access to the house is permitted LED light will be turned on the switch after checking the illuminance of the room and sound the buzzer will turn on in case of unauthorized entry. If the temperature is high (above 45 degrees), monitor the temperature continuously. In	1.Atmega644p 2.microcontroller 3.ADC 4.IR sensor 5.Gas sensor 6.Temperature sensor 7.Light sensor 8.LEDS & Buzzer 9.GSM module	The paper suggests that GSM-based security systems provide enhanced security as it can quickly send SMS alerts to the homeowner's desired number in case of any intrusion, gas leakage, or fire. The proposed system is controlled by an Atmega644p microcontroller and collects information from sensors to send SMS alerts. The paper also discusses two methods for enhancing	The effectiveness of the system may depend on various factors such as the quality and placement of sensors, network coverage, and reliability of the GSM module. This system does not use a camara	It uses the concept of smart home with GSM technology. It doesn’t integrate with a system to trigger in a flood. It doesn’t include the surveillance of a camara. The system doesn’t incorporate a fire extinguishing unit.

			the case of a fire, an SMS ("Fire at home") will be sent to the homeowner. In case of a gas leak sensed by the gas sensor, the owner will be notified through a SMS ("gas leak").		home security using GSM-based systems, one using a web camera and the other using a sim548c GSM module.		
[6]	Dedi Satria, Syaifuddin Yana, Rizal Munadi, and Saumi Syahreza 2017	Prototype of Google Maps-Based Flood Monitoring System Using Arduino and GSM Module	It begins with an ultrasonic sensor that detects water levels. The GPS module then sends the water level data and flood location coordinates to the Arduino Uno as a data processor. Both data are sent in the form of SMS data to the information system station received by the modem. The data is received by a computer and processor to create a water level information system based on Google Maps. This information is displayed as a map with inundation height	1. Ultrasonic sensors 2. Arduino Uno 3. U-Blox Neo 6m GPS module as a detector of flood coordinate location 4. GSM SIM900 module 5. Computer and processor for receiving and displaying the data on Google Maps through a browser 6. Jumper Wires	A Google Maps-based flood monitoring system developed as expected. The system can detect water level and send alerts to users via SMS to provide real-time information about flood height and location. It has successfully used an ultrasonic sensor, an Arduino Uno, a U-Blox Neo 6m GPS module and a GSM SIM900 module in the system. The system could improve response time and reduce damage in	The system is based on GSM technology for sending SMS notifications, which may not be available in some regions. Water level detection accuracy may be affected by factors such as debris and other obstacles in the water. Systems may require regular maintenance and calibration to ensure accurate readings. In some areas, the cost of implementing and maintaining systems can	Developing a Google-maps based monitoring system. Used GSM model. It doesn't include with fire detection system with the surveillance of a camara. The system doesn't incorporate a corrective action unit to divert the flood away.

			data via a browser, providing real-time information on inundation height and location.		flood prone areas. The study highlights the potential benefits of installing this system in flood-prone areas, such as improved disaster management and damage reduction.	be a barrier to adoption. This study was performed as a prototype and further investigation is required to assess the effectiveness of the system in real-life scenarios.	
[7]	Md. Rawshan Habib, Naureen Khan, Koushik Ahmed, Mahbubur Rahman Kiran, Mohaiminul Islam Bhuiyan, and Omar Farrok. 2019	Quick Fire Sensing Model and Extinguishing by Using an Arduino Based Fire Protection Device	A flame sensor senses the fire and sends an electrical signal to the microcontroller. A microcontroller receives a signal and sends it to five outputs that activate various components of the system. Solenoid relay switches actuate to turn on servo motors, fire sirens, and fire extinguisher induction motors. Two additional signals are sent to the actuator and the mobile phone's LEDs to alert the concerned parties of the fire incident. Water and powder spray systems are activated by activating a relay switch that helps extinguish the	1. Flame sensor 2. Smoke detector 3. Temperature sensor 4. Fire extinguisher 5. Transformer (12V step-down) 6. Servo motor (for mobile phone and call button) 7. Single-phase induction motor 8. Bridge rectifier 9. Filtering capacitor 10. Fire siren 11. LED	The proposed fire protection device shows reliability with fewer false alarms. The time delay is 1.5s for activating. alarm. Thus, it can neglect the smoke created from cigarettes, burning papers etc. Commercial thermal sensors are expensive. A home-made converter is used in this device. Therefore, the proposed system is economical. Smoke, flame, and temperature sensors	This is where short range fire detectors are used. As a result, the system does not perform well in crowded areas. It uses a mobile phone to send updates to the right people, which is unreliable. The display is not used here to show status	System incorporates both fire detection and extinguishing systems in a single unit. The system uses multiple functional sensors to avoid the possibilities of malfunction of alarm circuit and decrease of false alarm. All sensors are employed twice in number to make the system more reliable. The system uses an ordinary button phone to call the owner's number, which is executed by a servo motor that presses the call button three times. The owner is notified of the fire accident via mobile

			fire.	12. Relay (magnetic) 13. Microcontroller (Arduino-based) 14. Wires	have been duplicated to increase system reliability and accuracy.		phone network available in that area. It can neglect smoke created from cigarettes and burning paper. It doesn’t integrate with a system to trigger in a flood. Doesn’t included the surveillance of a camara.
[8]	Joni Welman Simatupang and Faiz Naufal 2019	Flood Early Warning Detection System Prototype Based on IoT Network	An ultrasonic sensor measures the water level and sends the data to the Arduino UNO. The Arduino UNO processes the data and sends it to the cloud server via SMS via the SIM900 GSM/GPRS module. Cloud servers store and process data that users can access from their smartphones and other devices. When the water level reaches a certain level, an early warning message will be sent to the measurement point management and the data will be distributed to the population.	1. Arduino UNO 2. Ultrasonic sensor 3. SIM900 GSM/GPRS module 4. Breadboard and jumper wires 5. Power supply (9V battery or adapter) 6. Cloud server	A prototype flood early warning system has proven useful as one of the solutions that can be implemented to reduce the number of casualties from floods that may occur in the near future. By receiving sensor data from ultrasonic sensors and distributing it through GSM and GPRS modules, the system was able to frequently route, record and publish the data on a website. I was also able to ask questions about current water levels and respond	The ultrasonic sensor used in the system had an accuracy of about 20%. This means that the measurements may not be very accurate. The GSM and GPRS modules used in the system were found to be less responsive to some commands, affecting the functionality of the Arduino. The system requires power, either a 9V battery or an adapter, which may not be readily available.	Integration of Arduino UNO, ultrasonic sensor, and GSM/GPRS module to create a low-cost and effective early warning system for floods. Use of cloud server to store and process data, which can be accessed by users through their smartphones or other devices. Ability to reply directly to messages from anyone asking about the current water level condition. It doesn’t include a fire detection system with the

					directly to messages from people who were interested in knowing if.		surveillance of a camara. The system doesn't incorporate a corrective action unit to divert the flood away.
[9]	Sheikh Azid, Bibhya Sharma, Krishna Raghuwaiya, Abinendra Chand, Sumeet Prasad, A Jacquier	SMS Based Flood Monitoring and Early Warning System	A pressure sensor measures the water level and sends the data to the Arduino microprocessor. An Arduino microprocessor processes the data and sends his SMS alerts to vulnerable and threatened people as well as relevant authorities via a GSM modem. By incorporating a solar charging system, the system will become self-sufficient by continuously charging batteries in the remote areas where the facility is located. Users can check the battery status through her GSM	1. Arduino microprocessor 2. GSM modem 3. Pressure sensor 4. Aluminum box 5. a column of a bridge or a dedicated concrete support 6. Solar battery charging system 7. SIM card (for GSM module) 8. Wires and connectors for circuitry	This system successfully validates the use of pressure sensors in water level monitoring systems as the relationship between pressure and water level is perfectly linear. The system is self-contained and does not require external power, but recharging the SIM card and saving contacts is done via SMS. The whole system is solar powered, and the rechargeable battery can last for about a week.	GSM network and may not be available in some remote areas. It needs external support like a column of a bridge or special concrete support to hold the pressure sensor in place. These may not be available in all locations. The system is limited by the accuracy of the pressure sensor used and may not be able to detect small changes in water level. The GSM module cannot update itself, so	It uses an SMS based system using GSM module. The incorporation of a pressure sensor to measure water level height, which is a more accurate and reliable method than traditional methods such as visual inspection or manual measurement. The incorporation of a solar battery charging system to make the system independent and self-sustaining, which is particularly useful in remote

			module with the ability to check the battery status at any time. The module should be able to report the battery level to the user via SMS. Remote replenishment and resident number addition are also integrated for complete system efficiency.	9. Resistors, capacitors, and diodes	SMS-based flood monitoring and early warning systems are more efficient than other methods such as radio and television broadcasts. This is because it can reach people in remote areas where other methods are not available. A potential problem identified in this research is the inability of the GSM module to upgrade itself when network operators make changes to the network.	if the network provider makes changes to their network, your system may experience problems. Depending on the local government or region, the introduction cost of this system may be high.	areas where access to electricity may be limited. The inclusion of features such as remote top-up and storing contact numbers via SMS makes the system more user-friendly and accessible. Doesn’t include with fire detection system with the surveillance of a camara. The system doesn’t incorporate a corrective action unit to divert the flood away.

[10]	SuwarjonoSuwarjono, along with Izak Habel Wayangkau, Teddy Istanto, Rachmat Rachmat, Marsujitullah Marsujitullah, Hariyanto Hariyanto, Wahyu Caesarendra, Stanislaw Legutko, and Adam Glowacz. 2021	Design of a Home Fire Detection System Using Arduino and SMS Gateway	This research involves designing and implementing a fire alarm system using an Arduino Uno microcontroller and an SMS gateway. The system is divided into four stages: fire detection, data processing, SMS sending and alarm. The authors used a flame sensor to detect fires and a GSM module to send his SMS alerts to the homeowner. The system was tested in a real-world environment, and the results showed that it worked as expected with 10 successful attempts to send SMS and trigger alarms.	1. Arduino Uno R3 Atmega328p microcontroller board 2. DS18B20 temperature sensor 3. MQ2 gas sensor 4. Sim900 GSM module 5. Active buzzer 5 V–12 V 6. Adapter 12 V–1 A 7. Alkaline Battery 9 V	I have successfully designed and implemented a fire alarm system using an Arduino Uno microcontroller and an SMS gateway. The system was tested in a real-world environment, and the results showed that it worked as expected with 10 successful attempts to send SMS and trigger alarms. The authors also explained the limitations of their proposed system, such as the inability to detect smoke and his reliance on the GSM network to send SMS alerts. Overall, this study demonstrates that automatic fire alarm systems can be designed at low cost using off-the-shelf components.	The system is designed to detect flames using a flame sensor, but it cannot detect smoke, which can also be an indicator of a fire. If there is no network coverage or if the network is congested, the system may not be able to send alerts. The range of the GSM module used in the study is limited, which means that the system may not be able to send alerts if it is located far away from a cellular tower. The system requires a stable power supply to function properly. If there are power outages or fluctuations, the system may not work as intended. The flame sensor may trigger false alarms if it detects other sources of heat or light, such as sunlight or incandescent	Automated SMS alerts using GSM module. Integration of multiple sensors: The authors integrated multiple sensors, including a flame sensor and a gas sensor, to detect fires more accurately. It doesn’t include a flood detection system with the surveillance of a camara. The system doesn’t incorporate a fire extinguishing unit.

						bulbs.	
	My project	A Flood and Fire Detection with Corrective Actions	The proposed system integrates a Node MCU microcontroller, an ultrasonic sensor for flood detection, and a single flame sensor for fire detection. It is also equipped with a drainage pump, which is effective in preventing flooding. When rising water levels or the presence of flames are detected, the system triggers real-time alerts via his Blynk app, notifying users instantly. At the same time, the house alarm is activated and alerts the occupants and	1.Node MCU 2.Flame Sensor 3. Ultrasound Sensor 4.Jumper Wires male to male sets 6.ESP 32 Fi-Wi modules with Camara 7.Water Pump 12 V 8.Water Sprinkler 9.Alarm 12V 9.Relay 3V to 12 V 10.Water tube pipe (hose)	To successfully build a fire detection system using IoT which would detect the fire, send an emergency alert, trigger the house alarm as well automatically take corrective actions to prevent fire from spreading of fire by automating water sprinkler system. To successfully create a flood detection system using IoT which would detect the flood water level and send an emergency alert, trigger	This project is limited to flood and fire detection and corrective action and does not include other home automation features such as temperature control, lighting, or security. Flame sensors typically have a range of a few feet so their range can be limited by the strength of the flame and the level of ambient light in the surrounding area. Ultrasound sensors can interfere with other objects in the water.	The system can do both fire and flood detection. The increase of water is detected in 3 levels and response to each level varies making the system not to do unnecessary disturbance in a less important event. The system incorporates corrective action units for both fire and flood detection; to extinguish the fire and to divert the flood away.

			surrounding people. Rephrase A snapshot of the monitored area where the system is installed can be requested via the Telegram bot.		the house alarm as well automatically take corrective actions to prevent flood spreading by turning of the water draining pump. To successfully enable the camera view for surveillance purposes for both flood and fire detection by taking snapshots.	The ESP32 Wi-Fi module requires a strong network connectivity, otherwise the alert message to the user can get delayed and the live streaming for surveillance of the event by the camara can get disrupted. The system is limited to 2 main corrective actions such as water sprinkler for the fire extinguishing and pump to drain the water. Any other actions such as ventilation system to act against a fire is not included. The higher rate of water floor cannot be handled by the 12 V pumps so the water flooding/ filtering should be controlled in the simulation. Sensors can malfunction when an event occurs.	The system includes a surveillance unit with a camera. It uses an ESP 32 Wi-Fi module which is built in one module with a camara which minimizes the cost of the unit.

2.5 Chapter Summary

Overall, this chapter is about what past researchers have done and the comparisons of ideas and knowledge that has been published in journals. This chapter details the previous findings conducted by other researchers regarding fire detection systems and flood detection systems. Most of the studies are found to be done for either a fire detection system or flood detection system as shown most of the studies mentioned above expect for two studies ([2], [4]). These two studies [2] [4] have been done for a designing of a smart home system with the capability of multiple detections; fire and gas leak, house intrusion etc. The three systems above mentioned [2] [3] [7] have an automated system provided to combat fire as in one study [2], a solenoid valve operates and stops the fire, second study, [3] extinguish fires using water from a tank and the other [7], the fire extinguisher induction motors ON in an event of a fire. My project incorporates two detections, 'Fire and Flood Detections' as well as incorporates corrective action units for both fire and flood detection as to extinguish the fire and to drain the flooded water. The system includes a surveillance unit with a camera as well. If a GSM module is included in the system, as mentioned in study [9], if the network provider makes changes to their network, the system can experience problems since GSM module cannot update itself. Therefore, I will be using an ESP 32 Wi-Fi module which is built in one module with a camara which also minimizes the cost of the unit. The increase of water is detected in 3 levels and response to each level varies making the system not to do unnecessary, too much disturbance to the owner or for the surrounding in less important events like other water leaks.

CHAPTER THREE : METHODOLOGY

3.1 Overview

This system be using a microcontroller and ESP32 Camara to enable camera view, flood detection using ultrasound sensor and fire detection using flame sensor and upon detection shall alert the user app alert using Blynk and trigger the house alarm with the corrective actions to bring down the fire and pump out the water away.
The main parts that will be used in the 2 sub-systems are water sprinkler, water draining pump, and ESC 32 Camera. An alarm will go on when fire is detected, and a pop-up notification will be sent to the user's device using the Blynk app. Water sprinklers will also turn on automatically to combat the fire.
Similarly, Flood detection systems continuously monitor water levels using ultrasonic sensors. The sensor measures the distance to the water surface and triggers targeted actions based on predefined thresholds. When the water level rises to level 1 (18 cm below from the ultrasonic sensor), the system activates the water level siren for a set time and sounds an alarm. Users will receive a pop-up notification via the Blynk app with real-time information about rising water levels. Similarly, at level 2 (16 cm below from the ultrasonic sensor), the siren is triggered one after the other, accompanied by activation of the draining pump and corresponding app notification. Additionally, if the water level father increased, and level 3 reached (14 cm below from the ultrasonic sensor), the siren is triggered one after the other and corresponding app notification if not when the water drained out and level decreased to the threshold (22cm below from the ultrasonic sensor), the drainage pump is stopped. The system responds in a timely manner to fluctuations in water levels and uses both visual and audio cues and pop-up notifications to alert users to potential flooding risks. Finally, the ESC 32 Camera will allow users to surveil the area and confirm the app alert is accurate. Telegram chat bot is used to request a snapshot of where the system is installed. Additionally, the ESP32 WiFi camera captures snapshots of Telegram bot broadcasts for enhanced user monitoring. The Blynk app enables wireless notifications through the Node MCU and demonstrates the system's dual wireless and direct output capabilities. As in conclusion,

there are 2 wireless outs and 4 direct outputs in the system.

Prototype flood simulations effectively demonstrate the system's corrective actions. Floods are simulated by filtering water from a storage tank into a simulation tank. When the water level reaches the level 2, the sump pump returns the water to the storage tank. Similarly, if a fire is detected, the system activates a sprinkler system connected to the same sump pump used in the flood simulation. The water in the flood simulation pool is returned to the storage tank depending on the water level in the simulation tank.

3.2 Block Diagram

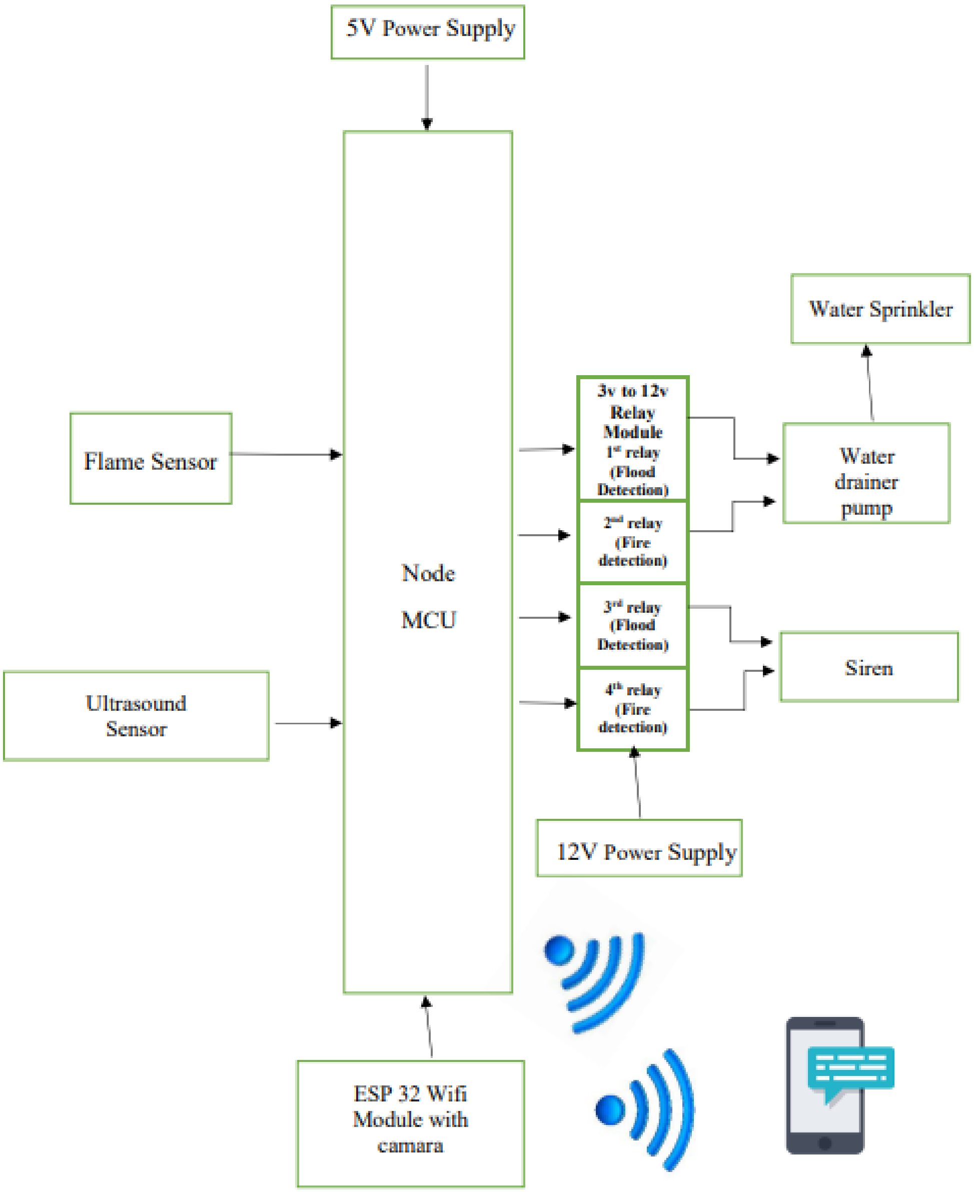


Figure 3.1: The Transmitter Block Diagram

The block diagram in Figure 3.1 provides a comprehensive overview of the project's architecture and details the interconnections of its 14 components. This system includes its three main inputs: a flame sensor, an ultrasonic sensor, and an ESP32 camera module. These inputs are fed to the node MCU, which coordinates the system response via a relay module with four 3V to 12V relays. The 12V power supply is connected to the relay and the node MCU is connected to the 5V power supply. The output of the relay module is routed to both the siren and water pump components, each controlled by a dedicated flood and fire detection relay. Sprinklers act as both flood and fire triggers and are closely related to water pumps.

3.3 Flowcharts

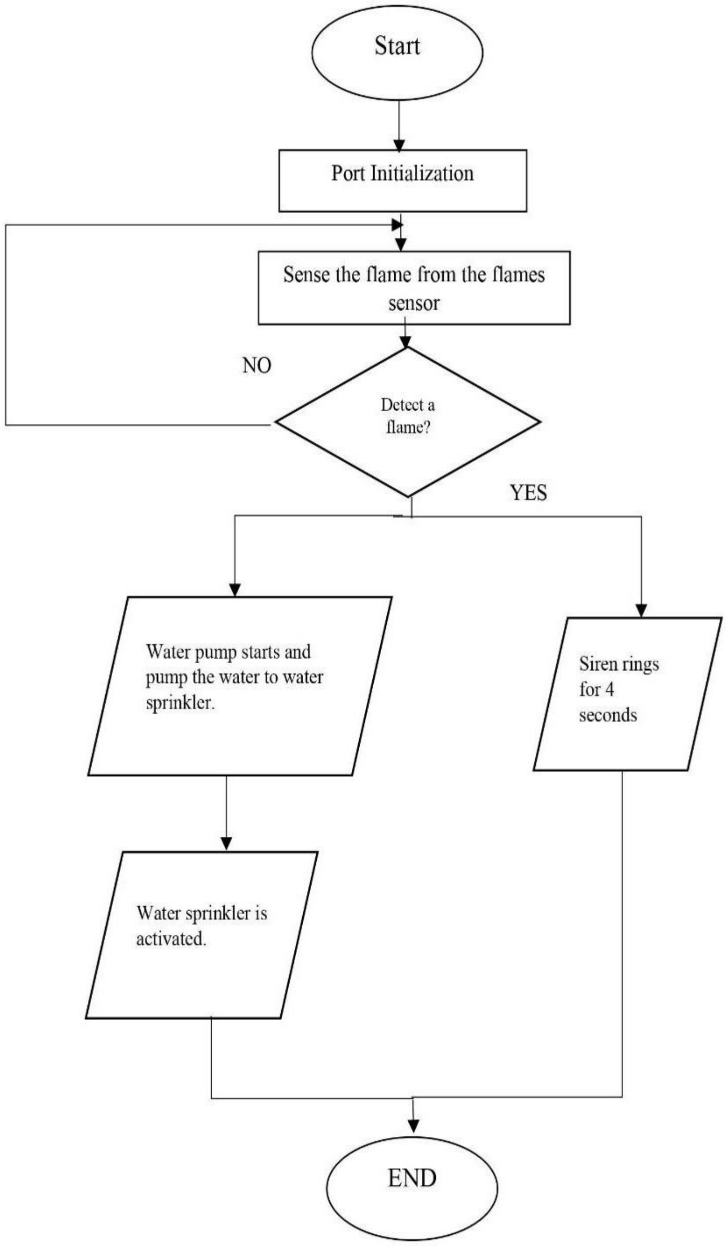


Figure 3.2: The Flowchart of the Fire Detection System

The fire detection system with flame sensor works seamlessly to detect the presence of fire and notify the user immediately. In this intuitive process, each flame sensor acts as an alert node that can detect flame When a sensor detects a fire, this critical information is efficiently transmitted to a central control unit. The control unit, which acts as the nerve center of the system, responds immediately by activating a loud siren. This versatile alarm mechanism includes both visual and audible alarms to notify users of potential fire hazards quickly and effectively. Integrated flowcharts highlight the reliability of the system, demonstrating streamlined and responsive processes and providing real-time alerts for increased safety.

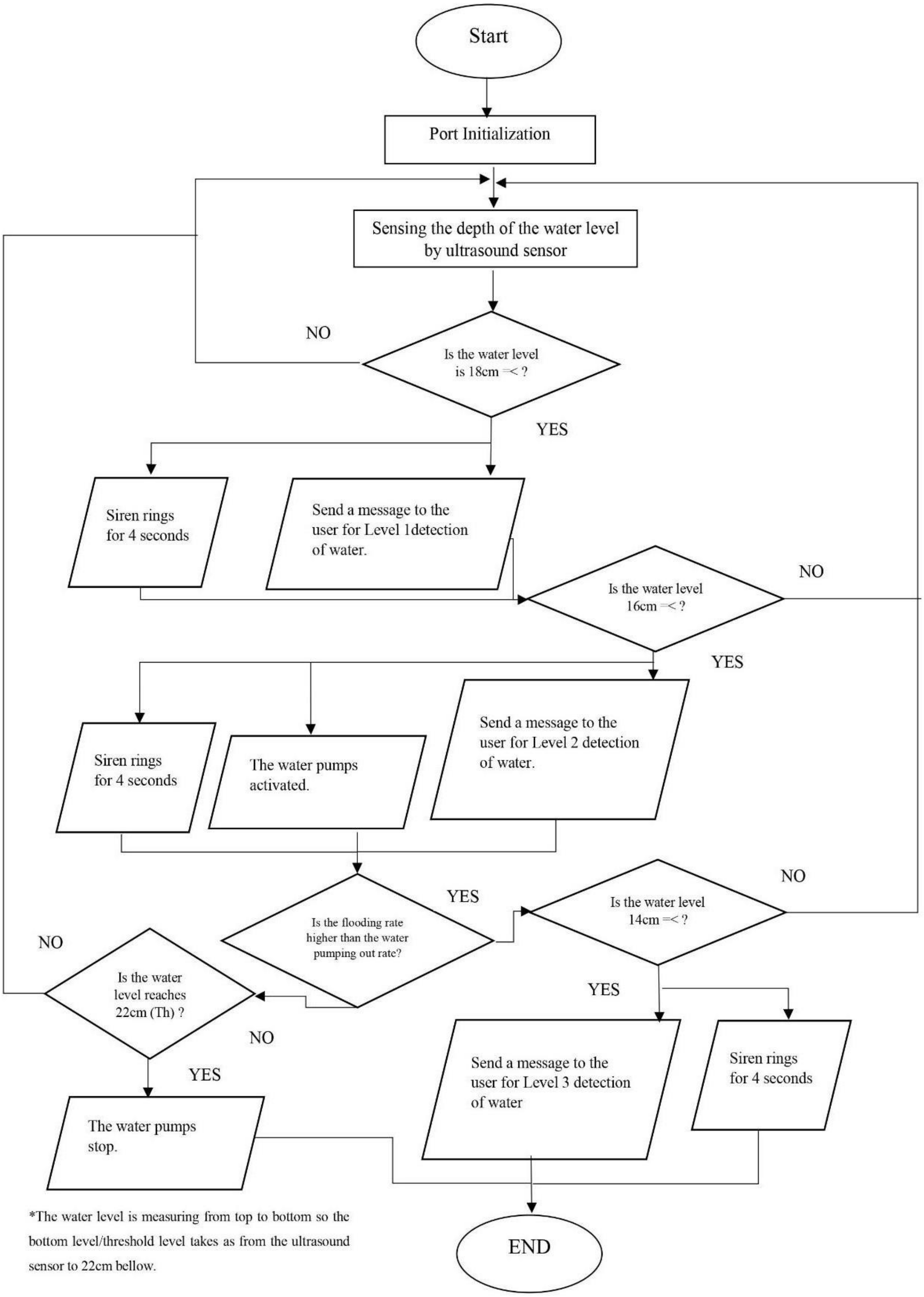


Figure 3.3: The Flowchart of the Flood Detection System

Flood detection systems continuously monitor water levels using ultrasonic sensors. The sensor measures the distance to the water surface and triggers targeted actions based on predefined thresholds. When the water level rises to level 1 (18 cm below from the ultrasonic sensor), the system activates the water level siren for a set time and sounds an alarm. Users will receive a pop-up notification via the Blynk app with real-time information about rising water levels. Similarly, at level 2 (16 cm below from the ultrasonic sensor), the siren is triggered one after the other, accompanied by activation of the draining pump and corresponding app notification. Additionally, if the water level father increased, and level 3 reached (14 cm below from the ultrasonic sensor), the siren is triggered one after the other and corresponding app notification if not when the water drained out and level decreased to the threshold (22cm below from the ultrasonic sensor), the drainage pump is stopped. The system responds in a timely manner to fluctuations in water levels and uses both visual and audio cues and pop-up notifications to alert users to potential flooding risks.

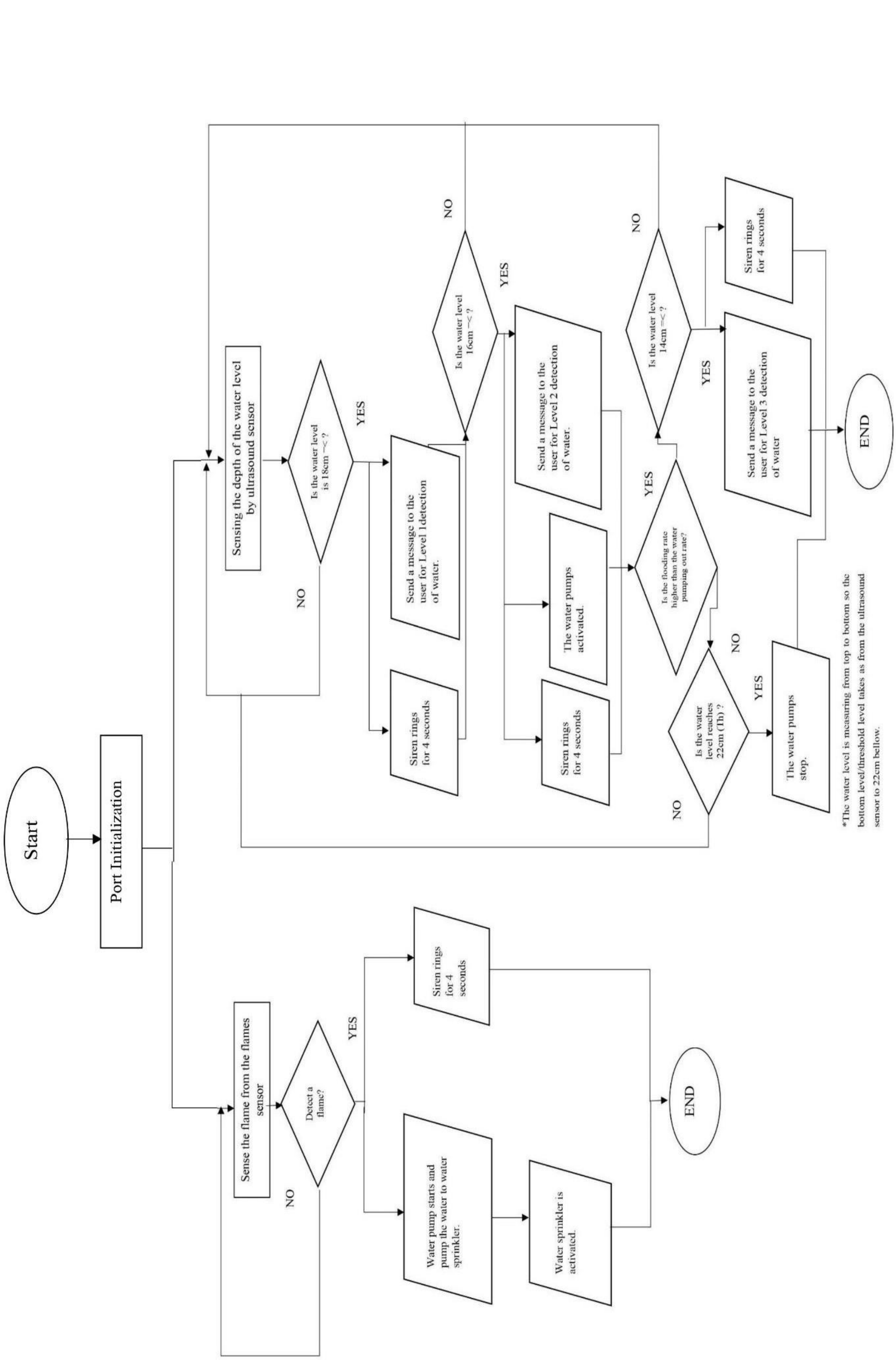


Figure 3.4: The Flowchart of Fire and Flood Detection System

Integrated flood and fire detection systems consist of sensors and control units used to simultaneously monitor flood and fire hazards. In the event of a fire, the flame sensor detects flames and activates alarms such as mobile notifications and sirens. At the same time, the ultrasonic sensor monitors the rising water level, activates the impermeable draining pump, and sends a pop-up notification when reach the assigned water levels. This integrated system provides comprehensive protection by providing early warning and enabling rapid response to fire and flood emergencies, increasing overall safety and minimizing potential damage. keep it to a limit.

3.4 List of Components Used

3.4.1 Node MCU

Figure 3.5: Node MCU Microcontroller

The Node MCU microcontroller based on the ESP8266 WLAN module is a compact and versatile IoT platform. It features a 32-bit Tensilica microcontroller that enables seamless integration with Wi-Fi connectivity. Built-in support for the Arduino IDE makes programming and project development easy. The Node MCU's compact size, GPIO pins, and Wi-Fi capabilities make it ideal for IoT applications, allowing remote monitoring and control. Its affordable price and user-friendly interface have contributed to its popularity in a variety of projects, from home automation to sensor networks. Overall, the Node MCU microcontroller is a powerful and accessible solution for IoT enthusiasts and developers.

3.4.2 Ultrasound Sensor

Figure 3.6: Ultrasound Sensor

Ultrasound sensors are non-contact distance measurement devices that use ultrasonic waves. It consists of a transmitter that emits ultrasound pulses and a receiver that detects the echoes. By calculating the time, it takes for the pulse to return, the sensor measures the distance to the object. Ultrasound sensors are widely used in robotics, security systems, and automation, and are versatile and work well in a variety of environments. It provides accurate distance measurement, works effectively in the dark, and is not affected by color or transparency. These sensors are critical for applications where accurate proximity detection and obstacle avoidance are important to optimize system performance.

3.4.3 Flame Sensor

Figure 3.7: Flame Sensor

Flame sensors are key components for detecting the presence of flames in a given environment. It uses various technologies such as infrared and ultraviolet to detect the radiation emitted by the flame. The sensor is designed to respond to specific wavelengths of light emitted by flames, allowing it to distinguish between the presence or absence of flames. When a flame is detected, the sensor sends a signal to a control system or device so that appropriate action can be taken, such as: B. Adjustments to fuel delivery, ignition control, or safety measures. Flame sensors are widely used in systems involving combustion processes such as: B. Ovens, boilers, gas stoves and industrial burners serve as important safety features to prevent uncontrolled fires and to ensure proper operation.

3.4.4 Male to Female Jumper Wires sets

Figure 3.8: Male to Female Jumper Wire

Jumper wires male to female sets are essential components in electronics projects, providing a convenient and solderless method for connecting various components. With a male pin connector on one end and a female pin connector on the other, these versatile wires enable easy and flexible connections between male headers, sensors, microcontrollers, and other components. The color-coded wires come in different lengths, facilitating organized and efficient circuitry. Whether used for prototyping, circuit testing, or breadboarding, these jumper wire sets offer a reliable and reusable solution for establishing temporary or permanent connections in electronic projects.

3.4.5 Printed Circuit Board (PCB)

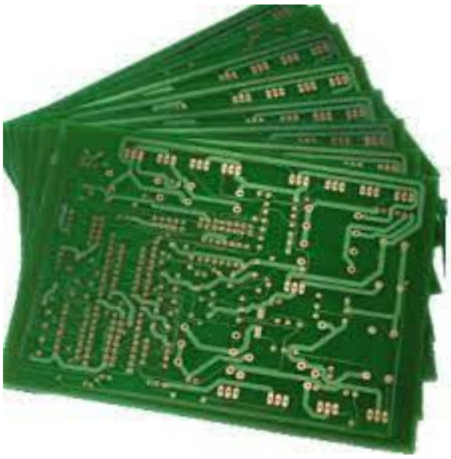


Figure 3.9: PCB

A printed circuit board (PCB) is a flat circuit board made of non-conductive material such as fiberglass or composite epoxy with a thin layer of conductive material on its surface. It serves as the basis for the structured and efficient assembly and connection of electronic components. Conductive layers (usually copper) are etched into intricate patterns to create paths or traces that facilitate the flow of electrical signals between components. Printed circuit boards are designed using CAD (Computer Aided Design) software and manufactured using processes such as etching, drilling, and soldering. They provide a compact and reliable means of connecting and assembling electronic components, ensuring optimal functionality, ease of assembly and efficient production of various electronic devices and systems.

3.4.6 ESP 32 WiFi Module with Camara

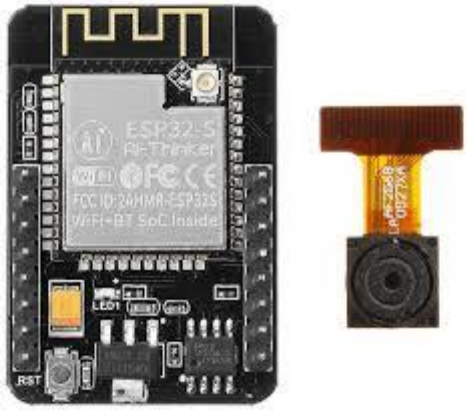


Figure 3.10: ESP 32 WiFi Module with Camara

The ESP32 Wi-Fi Module with Camera is a versatile electronic module that combines the functionality of an ESP32 microcontroller and a camera module. ESP32 is a powerful and widely used microcontroller with built-in Wi-Fi and Bluetooth connectivity, and a camera module for image or video capture and processing. This combination enables the development of applications such as surveillance systems, his IoT devices with image recognition capabilities, and video streaming projects. The ESP32 WiFi module with camera provides a compact and integrated solution for wireless communication and visual data collection, suitable for various projects requiring connectivity and image capture capabilities.

3.4.7 5V Water Pump

Figure 3.11: 5V Water Pump

A 12V water pump is a type of water pump that operates on a 5-volt power supply. It is specifically designed to be powered by low voltage power sources such as USB ports, rechargeable batteries, and microcontroller boards. These pumps are compact and portable, making them ideal for small water cycles and DIY projects that require low flow rates. They are commonly used in applications such as water-cooling systems for electronics, hydroponics, aquariums and small irrigation systems. A 12V water pump usually consists of a motor and an impeller. When the motor is energized, the impeller rotates and creates centrifugal force, drawing water into the pump and out through the nozzle or outlet. It is important to note that 5V water pumps are typically designed for low pressure, low flow applications.

3.4.8 Water Sprinkler

.

Figure 3.12: Water Sprinkler

A watering device is a device used to distribute water to a specific area, usually for irrigation purposes such as gardens, lawns, farmlands and sports fields. Designed to mimic rainfall by dispersing water in a controlled pattern to promote even coverage and efficient irrigation. A sprinkler typically consists of a body, a nozzle, and a rotating or vibrating mechanism. When connected to a water source, the pressure forces water out of the nozzle, creating a fine spray or jet that is thrown into the air and drops into the surroundings. A rotating or vibrating mechanism ensures that the water is evenly distributed and provides greater coverage. Watering sprinklers bring convenience and automation to your irrigation system, saving you time and effort while maintaining proper hydration of your plants, lawns and crops. We have a variety of styles including fixed sprinklers, impact sprinklers, rotating sprinklers and pop-up sprinklers to meet different needs and preferences when it comes to achieving efficient and effective watering.

3.4.9 12V Alarm (Siren)

Figure 3.13: 12V Alarm

A power window motor, also known as a power window motor, is an electrical device used to control the movement of a car window. It is typically used in power window systems that allow automatic or electronic control of window operation. The window motor is responsible for driving the window regulator, the mechanical assembly that raises and lowers the window. When the driver or passenger presses the power windows switch, a signal is sent to the power window motor, which activates and rotates a series of gears or motor drive mechanisms. This rotational motion translates into linear motion, scrolling the window up and down. Window motors are typically designed for the specific weight and size of the window they operate on. They are powered by the vehicle's electrical system, typically powered by the vehicle's battery through fuses and relays. You can control the rotation direction of the motor to move the window in the desired direction.

3.4.10 Water Tube Pipe

Figure 3.14: Water Tube Pipe

Water hoses, also called water pipes or hoses, are flexible hoses used to carry water or other liquids from one place to another. They are typically made from materials such as PVC (polyvinyl chloride), rubber, and reinforced plastics for their strength, durability, and resistance to corrosion and deterioration from exposure to water. Water hoses come in a variety of sizes, lengths and types for a variety of uses, from domestic, horticultural, irrigation systems and plumbing to industrial and construction. One end is usually attached to a faucet, pump, or another water source, and the other end directs the flow of water to its desired location. Water pipes allow water to be transported in an efficient and controlled manner, facilitating critical tasks such as watering plants, watering buildings, and transferring liquids in industrial processes.

3.4.11 Custom Made Relay 3V to 12V

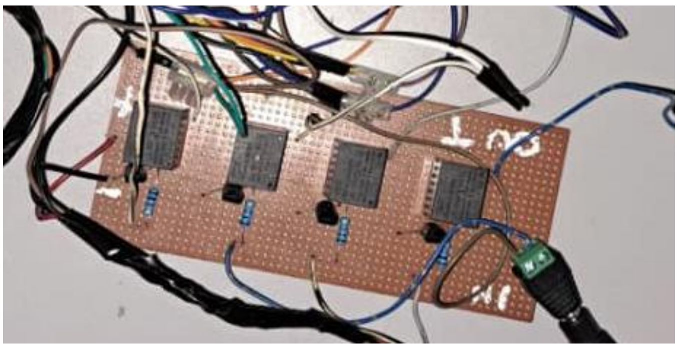


Figure 3.15: Custom Made Relay 3V to 12V

A customized relay module with four relays operating between 3V and 12V contains transistors that control the activation of the relays. Each relay acts as a switch and can handle different voltage levels. The transistor amplifies the low voltage input signal and allows effective control of the high voltage relay. This module enables seamless connectivity between low-voltage microcontrollers such as Arduino or Node MCU and high-voltage devices such as pumps and sirens. It provides a versatile solution for controlling multiple outputs in a circuit, increasing flexibility for automation and electronics projects where precise relay control is essential. The use of transistors ensures reliable and efficient relay switching, contributing to the overall performance and functionality of the module.

3.5 Electric Circuit Design Using Fritzing

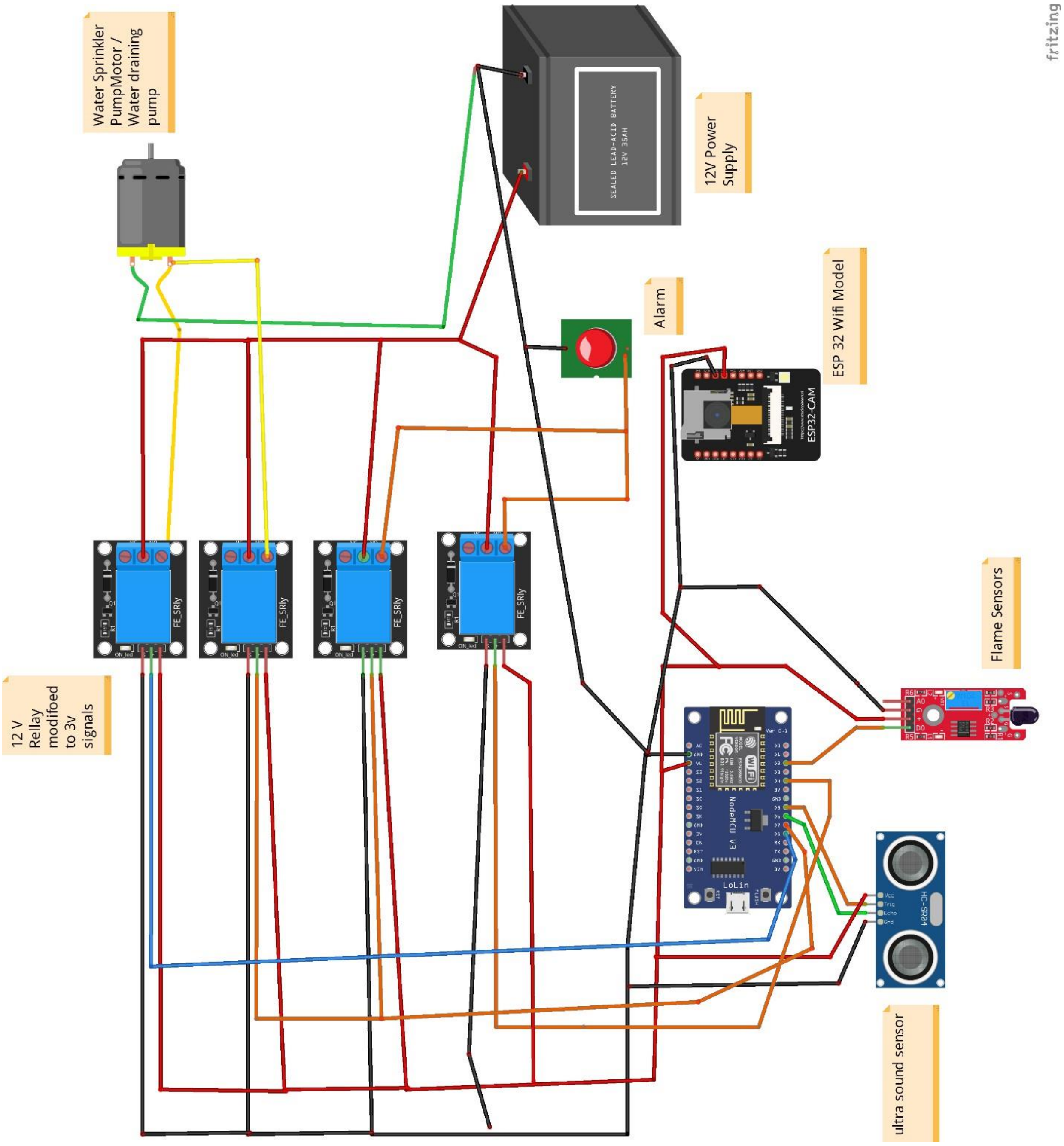


Figure 3.16: The Circuit diagram.

The circuit Diagram illustrates the Advanced Fire and Flood Detection System highlighting the main components and their connections. It uses two different power supplies: the, a 12V power supply for the pump and siren, and a 5V power supply that powers the node MCU, ESP32, and various sensors. This integration includes four relay modules that control sirens, sprinklers, and sump pumps, each with two outputs for

flood and fire detection. Strategically placed flame and
ultrasonic sensors effectively capture the input, whose output is passed to the node MCU for processing. Pumps and watering systems have complex connections and are controlled by node MCUs. Installation of sirens and water level alarms ensures quick warning. Connectivity extends to the Node MCU, which communicates with the Blynk app over Wi-Fi to deliver real- time notifications to mobile devices. Additionally, a Telegram bot called “Alert” enhances monitoring by capturing ESP32 snapshots during events. This complex design ensures a comprehensive and responsive fire and flood detection system.

3.8 Complete Hardware Prototype

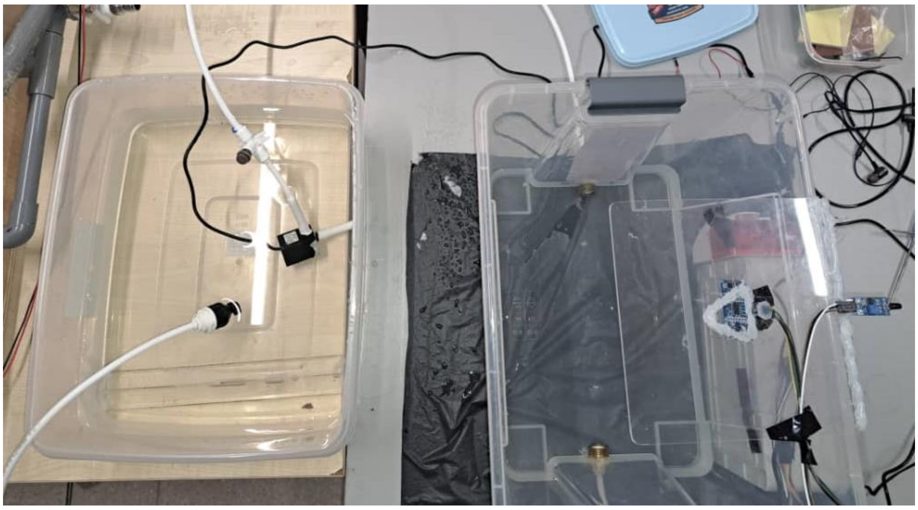


Figure 3.17: Front View

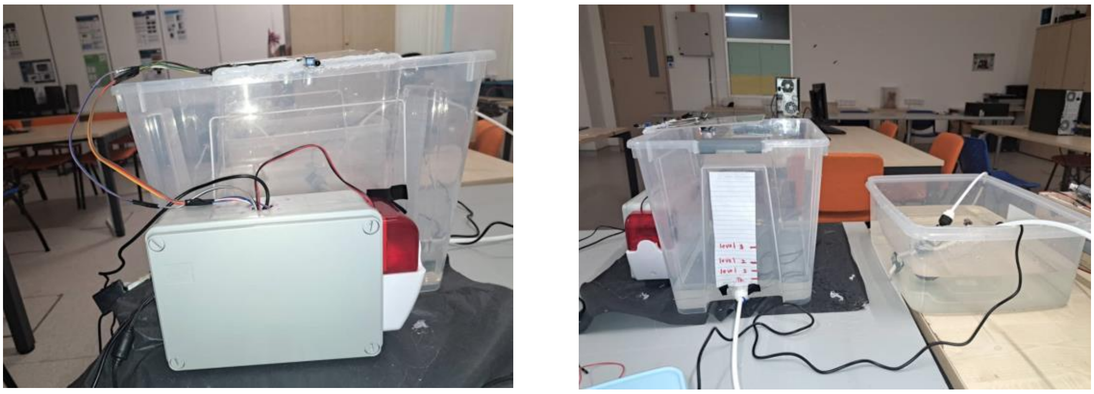


Figure 3.18: Rear View

The flood simulation in the prototype serves as a practical demonstration of the system's corrective measures. In this simulation, the initiation of flooding is orchestrated by channeling water from a storage tank into a designated simulation tank. Once the water level in the simulation tank reaches level 2, which 16cm bellow from the ultrasonic sensor, the drainage pump comes into action, expelling the water back into the storage tank. This cyclic process emulates a controlled flooding scenario. Conversely, in the event of a fire detection, the system activates a water sprinkler mechanism. Notably, this sprinkler system is ingeniously integrated with the same drainage pump utilized in the flood simulation. The sprinkler dispenses water from the flood simulation tank back into the storage tank, contingent upon the water level in the simulation tank. This interconnected use of components underscores the system's versatility and efficiency, showcasing its capability to adapt to varying emergency scenarios. The prototype's design, which employs 12V pumps for practicality and budget considerations, provides a scaled-down yet insightful representation of how the system would respond to real-life situations involving fire and flooding. To assemble the system, cutting hard items by saw, drilling and hot gluing have to be done. The control box houses hardware components such as sirens and water level alarms, creating a comprehensive security system.

Figure 3.19: ESP 32 camara snapshot in telegram chat

Telegram bot called "Alert" (Figure 3.19) captures snapshots from the ESP32 during events to improve monitoring capabilities. The Telegram bot named "Alert" plays a crucial role in enhancing the monitoring capabilities of the system. Integrated with the ESP32, this bot facilitates the capture of snapshots during fire or flood events. When triggered, the ESP32, equipped with a camera module, captures images of the surroundings. These snapshots are then seamlessly transmitted to the user through the Telegram bot. The ESP32-CAM is configured to connect to Wi-Fi, initialize the camera, and continually check for new messages from the Telegram bot. Users can command the system to capture photos ("/photo") or toggle the flash ("/flash"). When a photo is requested, the ESP32-CAM captures an image and utilizes the Telegram API to send it as a response, enhancing real-time surveillance capabilities and providing users with visual data to assess events remotely in their monitored area.

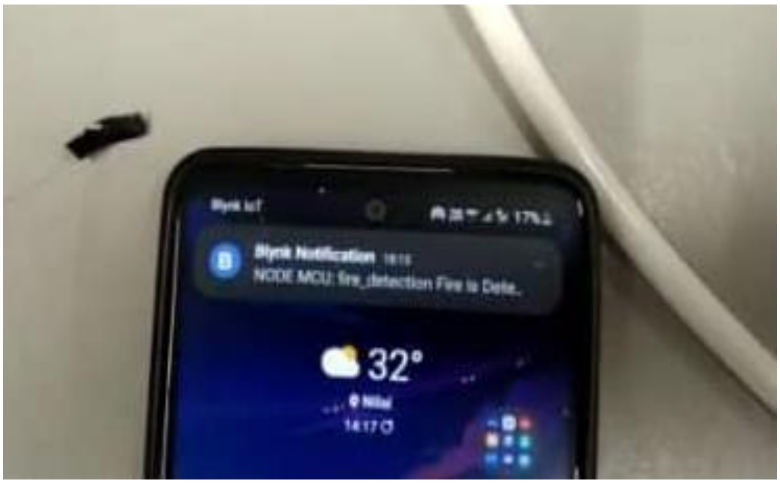


Figure 3.20: Pop up notification using Blynk App

ESP32 connects to its Blynk app and sends instant alerts to your phone. The Node MCU establishes a robust connection with its dedicated Blynk app, creating a dynamic link for seamless communication and real-time updates. In the event of fire or flood detection, the Node MCU triggers an immediate pop notification to the user's phone through the Blynk app, ensuring swift and direct alert delivery. This integration not only enhances the responsiveness of the system but also provides users with instantaneous information about potential threats to their surroundings. The Blynk app acts as a user-friendly interface, allowing individuals to monitor and receive alerts remotely, contributing to a proactive and informed response to emergency situations. The Node MCU's connectivity with the Blynk app exemplifies the system's commitment to prompt and reliable communication, emphasizing its role in delivering timely notifications for enhanced safety and security.

3.6 Chapter Summary

This chapter focus on the system methodology for the proposed system and block diagram which gives a clear view of the inputs and output will be connected to the system. The chapter consists of flowchart which shows the flow of the systems. It elaborates the taking the sensing data from the relevant sensors and analysing the sensory data, gives the relevant corrective actions according to the conditions. This chapter consists of 3 flow charts for the two sub-system such as fire and flood systems consecutively. It consists of the software and hardware circuits of the system as well. In the chapter, the operations for each components using the system also have been explained. the flood simulation within the prototype aptly demonstrates the system's effective corrective actions. The intricately designed simulation involves the filtration of water from a storage tank to a designated simulation tank. Once the water level surpasses a predetermined threshold, the drainage pump efficiently expels the water back into the storage tank. This adaptive approach is mirrored in the event of a detected fire, where the system seamlessly activates a water sprinkler connected to the same drainage pump used in the flood simulation. The water from the flood simulation tank is intelligently sprinkled back into the storage tank, contingent on the water level in the simulation tank. This comprehensive simulation not only showcases the system's responsiveness to both flood and fire scenarios but also underscores its versatility and reliability in addressing diverse challenges. Blynk and Telegram integration adds user interaction and monitoring capabilities to this fire and flood detection system.

CHAPTER FOUR : RESULTS AND DISCUSSION

4.1 Overview

This chapter provides a comprehensive overview of the test results and subsequent analysis of fire and flood detection systems. This includes evaluating algorithm improvements, hardware components, and system responsiveness in various simulated scenarios. The fire detection algorithm works effectively and activates the siren and water pump as soon as the flame sensor is detected. However, issues such as multiple notifications and the timing of siren activation were identified and were resolved by introducing a 4-second delay. Flood detection experiments demonstrated the system's ability to respond to different water levels and operate components accordingly. Challenges such as siren activation during high waters and timing of mobile notifications were identified and solutions were proposed for flag implementation. This section highlights the importance of real-world testing to validate the robustness of the system and highlights the successful integration of key components such as flame sensors, ultrasonic sensors, and dewatering pumps. When building up the protype, there multiple challenges such as Microcontroller Shift, Sensor Selection, Scaling Real-world Simulation within Financial Constraint and malfunctioning Components have been through and they have been explained the problems and solutions section. Overall, the results and discussion provide detailed insights into the performance of the system, address the observed issues, and present solutions for more efficient and reliable operations.

4.1 Obtained Results for Objective 1

- *To build a fire detection system using IoT which would detect the fire, send an emergency alert, trigger the house alarm as well automatically take corrective actions to prevent fire from spreading by automating water sprinkler system.*

Table 4.1: Fire Detection Testing

Measurement	1st Trial	2nd Trial	3rd Trial	4th Trial	5th Trial
Flame Sensor Output	ON	ON	ON	ON	ON
Siren Activation	ON	ON	OFF	ON	ON
Water Sprinkler Pump Activation	ON	ON	ON	ON	ON
Pop-up notification "Fire in the House"	Multiple pop notification for the same event	Multiple pop notification for the same event	Single pop notification for the same event	Single pop notificatio n for the same event	Single pop notificatio n for the same event

The table indicates consistent flame sensor output across trials, leading to siren and water pump activations. However, it highlights an issue with multiple pop-up notifications for the same event, indicating a potential glitch in the alert system. The fire detection algorithm enhanced to exhibit a responsive mechanism, promptly activating the siren and water pump upon flame sensor detection. The introduction of flags (Appendix C.1) ensures streamlined event handling and prevents repetitive actions. The 4-second delay allows for controlled the time water sprinkler and siren are activated which only added for the demonstration purposes and will be adjust accordingly in a real-life fire outbreak. The system's reliance on the flame sensor input, coupled with well-timed actions, enhances its ability to swiftly identify and address fire incidents. The LEDs of the flame sensors are lighting up which means the connectivity has been correctly established. The testing of the fire detection of flame has been done using lighter.

4.2 Obtained Results for Objective 2

- *To create a flood detection system using IoT which would detect the flood water level and send an emergency alert, trigger the house alarm as well automatically take corrective actions to prevent flood spreading by turning on the water draining* *pump.*

Table 4.2: Water Detection Testing

	Measurement (**distance from top the bottom of the tank**)	**1^ST Trial**			**2nd Trial**			**3rd Trial**			**4thTrial**			**5thTrial**		
		Draining Pump	Siren	Mobile Notification	Draining Pump	Siren	Mobile Notification	Draining Pump	Siren	Mobile Notification	Draining Pump	Siren	Mobile Notification	Draining Pump	Siren	Mobile Notification
Water Rising	**Water until level 1** **(18cm)**	ON	OFF	ON	ON	OFF	ON	OFF	ON	ON	OFF	OFF	ON	OFF	ON	ON
	Water until level 2 **(16 cm)**	ON	OFF	ON	ON	OFF	ON	ON	ON	ON	ON	ON	ON	ON	ON	ON
	Water until level 3 **(14 cm)**	ON	OFF	ON	ON	OFF	ON	ON	ON	ON	ON	ON	ON	ON	ON	ON
Water Decreasing	**Water until level 3** **(18 cm**	ON	OFF	ON	ON	OFF	ON	ON	OFF	ON	ON	OFF	ON	ON	OFF	ON
	Water	ON	OFF	ON	ON	OFF	ON	ON	OFF	ON	ON	OFF	ON	ON	OFF	ON

	until level 2 (16 cm															
	Water until level 1 (14 cm	ON	OFF	ON	ON	OFF	ON	OFF	OFF	ON	OFF	OFF	ON	OFF	OFF	ON

*The water level is measuring from top to bottom, so the bottom level/threshold level takes as from the ultras sound sensor to 22cm bellow.

The presented table summarizes the results of flood detection trials at different water levels, along with the corresponding states of the draining pump, siren, and mobile notifications. The flood detection algorithm effectively responds to rising and decreasing water levels, activating and deactivating components as needed. As the water rises to level 1 (18 cm), the draining pump remains off, the siren stays inactive until the 2nd trail, but the algorithm develops defining in the siren for water detection as a separate output from the fire detection. Therefore, from the 3rd trial, the siren activates, and a mobile notification activation is happened effectively in all trials. Similar responses occur at levels 2 (16 cm) and 3 (14 cm), with appropriate activations and notifications. During the water decreases from level 3 to level 1, the draining pump activates until the water level reaches the threshold level (22cm below from the ultrasound sensor), the siren remains off, and mobile notifications are sent.

4.3 Obtained Results for Objective 3

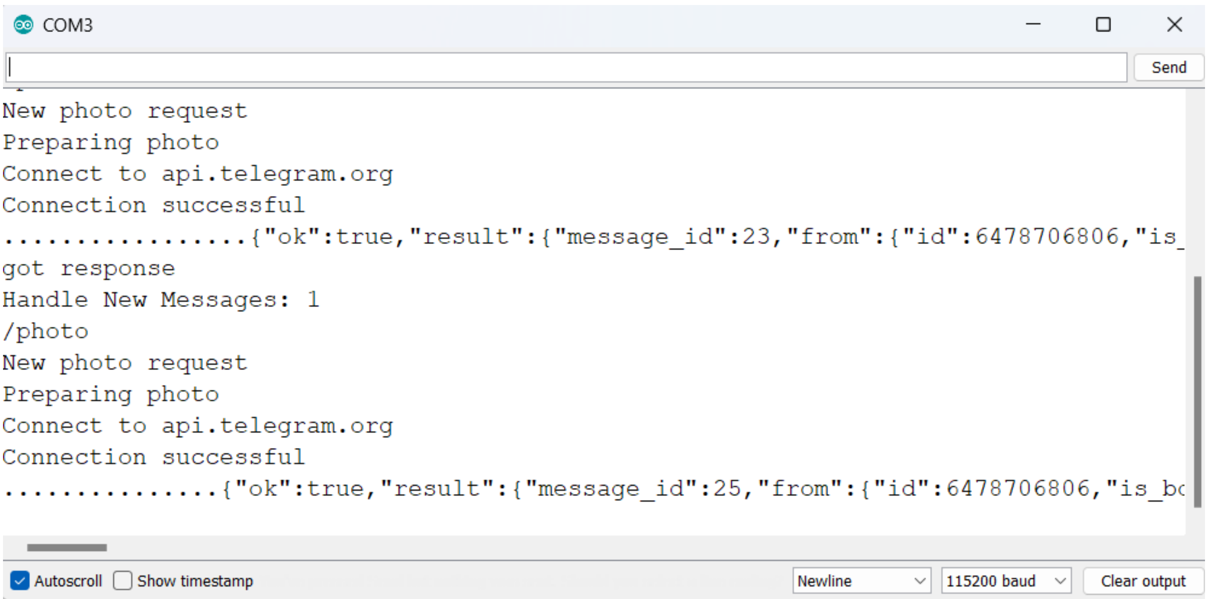
- *To enable a camera view for surveillance purposes for both flood and fire detection.*

Table 4.3: ESP 32 Camara and Telegram Bot called "Alert" Connectivity Testing

Measurement	1st Trial	2nd Trial	3rd Trial	4th Trial	5th Trial
The Wi-Fi connection	connected	connected	connected	connected	connected
Send "/photo" command to Telegram bot	Photo is taken	Photo is taken	Photo is taken	Photo is taken	Photo is taken
Send "/flash" command to Telegram bot	ON	OFF	ON	OFF	ON

The system successfully establishes a Wi-Fi connection in all trials. When the "/photo" command is sent to the Telegram bot, the system consistently takes a photo and sends it to Telegram. The "/flash" command toggles the flash LED state accordingly. The results align with the expectations, demonstrating the functionality and responsiveness of the Telegram communication in the given code (Appendix C.2).

Figure 4.1: ESP 32 communication with Telegram app (Referring through the serial monitor)



The Serial Monitor provides real-time insights into system activities, including Wi-Fi connection status, command receptions, and execution outcomes. This monitoring tool serves as a valuable resource for debugging and ensuring the smooth operation of the ESP32-CAM.

4.4 Problems Faced and Solutions

The provided information indicates several observations and potential issues in both the fire detection and flood detection algorithms. Let's break down the identified problems and discuss possible solutions that have been broken down.

Issues face in terms of programing:

Fire Detection

Multiple Notifications Glitch:

Issue: The 4.1 table suggests a problem with multiple pop-up notifications for the same fire event.

Solution: Introducing flags is a good approach to prevent repetitive actions.

Flood Detection

Multiple Notifications Glitch:

Issue: The 4.2 table suggests a problem with multiple pop-up notifications for the same fire event.

Solution: Introducing flags is a good approach to prevent repetitive actions.

Siren Activation During Water Increase:

Issue: The 4.2 table mentions that the siren is OFF when the water reaches 3 levels until the second trail.

Solution: the algorithm enhanced to define the siren output separately for water detection. Review the conditions for siren activation during water decrease.

Challenges face in terms of protype building:

Microcontroller and Connectivity Shift:

Challenge: The initial plan to use Arduino Uno faced complexities in budget and circuit design in doing serial communication with separate wifi module. Shifting to NodeMCU was a decision to streamline both budget and circuit complexity.

Solution: Adopting NodeMCU offered Blynk connectivity to a single device, simplifying the circuit and reducing costs while ensuring effective communication with the system.

Sensor Selection for Flood Detection:

Challenge: The initial choice of water level sensors for flood detection posed installation challenges inside the tank, leading to potential circuit malfunctions.

Solution: Transitioning ultrasonic sensors addressed the installation issue, allowing for non-intrusive flood detection and maintaining the system's reliability.

Adaptation to malfunctioning Components:

Challenge: Throughout the project, several ESP32 modules have been broken down and pressure pumps with higher water flow capacity was broken down.

Solution: Replacing ESP32 with a function ESP 32 and keep on testing and integrating 12 V pumps which has lower pressure while carefully managing dependencies to avoid exceeding the budget.

Scaling Real-world Simulation within Financial Constraints:

Challenge: Simulating life-size scenarios with budgetary limitations may restrict the scalability of the prototype.

Solution: Adapting the simulation to represent real-world conditions effectively within the available budget requires a meticulous approach. The flood is simulated by filtering water from a storage tank into a simulation tank. When the water level reaches a certain point, the drainage pump expels the water back to the storage tank. Similarly, in the event of a detected fire, the system activates a water sprinkler, which is attached to the same drainage pump used in the flood simulation. The water from the flood simulation tank is then sprinkled back into the storage tank, dependent on the water level in the simulation tank by prioritizing key aspects and essential functionalities ensures a meaningful demonstration within financial constraints.

4.5 Chapter Summary

This chapter focuses on the results and findings obtained from the data collected for this study. Subsequently, analysis was performed and shown in this chapter that the system is functioning successfully establishing its objectives. The assessment of the fire and flood detection algorithms reveals noteworthy issues and corresponding solutions. In the realm of fire detection, a glitch involving multiple notifications for the same event is identified. The recommended solution entails the introduction of flags to forestall repetitive actions, thereby bolstering system stability during fire incidents. Likewise, the flood detection system encounters a parallel challenge with redundant notifications. The proposed remedy mirrors that of the fire detection algorithm employing flags to streamline and optimize event handling. Furthermore, it exposes a discrepancy in siren activation during water increase until the second trial. To rectify this, the algorithm is slated for refinement, with a focus on delineating the siren output specifically for water detection and scrutinizing conditions governing its activation during water decrease. These systematic improvements underscore the commitment to fine-tuning the detection mechanisms, ensuring enhanced reliability and performance in real-world scenarios. The flood and fire detection system successfully navigated challenges through strategic decisions, including the shift from Arduino Uno to NodeMCU for cost-effectiveness and simplified circuitry. Transitioning to ultrasonic sensors addressed installation issues, enhancing reliability. Adaptive measures to address malfunctioning components, such as replacing ESP32 modules and integrating lower pressure 12V pumps, ensured project continuity while managing budget constraints. The flood simulation showcased the system's corrective actions, demonstrating its capabilities in real-world scenarios. This approach, marked by adaptability, strategic decision-making, and prioritization, not only overcame technical challenges but also emphasized the system's functionality within financial constraints, setting the stage for future enhancements and practical applications.

CHAPTER FIVE: CONCLUSION AND RECOMMENDATION

5.1 Conclusion

This report provides the necessary details to support the project's rationale and objectives. This project is to build a flood and fire detection system using Node MCU, flame sensor, ultrasonic sensor, ESP32 Camara module, watering system, and sump pump. The system consists of four main units: fire and flood alarms, water sprinklers, water pumps, and ESC 32 cameras. If a fire is detected, an alarm will sound and a pop-up notification will be sent to the user's device via the Blynk app. Additionally, sprinklers will automatically operate to extinguish the fire. Similarly, flood detection systems use ultrasonic sensors to detect rising water levels and send appropriate app notifications to users' devices. A waters pump was also used to drain water into another tank and filter the water into a flood simulation tank. Therefore, water is recycled within the system. Finally, the ESC 32 camera allows users to monitor the area and ensure that app alerts are correct. A bot created in the Telegram app is used to receive snapshots when requested by users. Fire alarm systems with flame sensors have proven to be effective in detecting fires in a variety of close areas.

Floods are simulated by filtering water from a storage tank into a simulation tank. When the water level reaches level 2, the sump pump returns the water to the storage tank. Similarly, if a fire is detected, the system activates a sprinkler system connected to the same sump pump used in the flood simulation. The water in the flood simulation pool is returned to the storage tank depending on the water level in the simulation pool.

Flood detection systems with water level sensors have been proven to be able to accurately detect water levels at different heights. Both fire and flood alarm systems have consistently provided reliable and accurate results.

The choice of using 12V pumps in the prototype is due to budget constraints and the challenge of creating a life-size scenario. In real-world scenarios, it is essential to use larger capacity pumps and robust hardware, especially in areas with high water flows. In a real-life situation, separate water storage for the sprinkler system would be employed, and the pumped-out water during flooding would be directed to a distant reservoir, avoiding local flooding. The prototype operates on both 5V and 12V power supplies, acknowledging that, in a real-life scenario, the first measure during flooding is often to cut off electricity. Therefore, the system should have the capability to run on backup or remote power sources to ensure its functionality even in adverse conditions. Finally, one of the main benefits of this system is that it gives homeowners peace of mind by proactive approaches to preventing damage to their homes and property, allowing them to not be burdened by insurances, loss of lives and property damage. This feature is especially important for homes that may be absent in an emergency, allowing the system to act when no one is there. Additionally, the system is highly customizable, cost-effective, and easy to use, making it an ideal solution for a wide variety of seekers/homeowners who seek the protection in their own home with less burden.

Table 5.1: Objective Achievement Table

Objectives	Achieved/Not Achieved	Justifications
To build a fire detection system using IoT which would detect the fire, send an emergency alert, trigger the house alarm as well automatically take corrective actions to prevent fire from spreading by automating water sprinkler system.	Achieved	This fire alarm system successfully integrates IoT components such as flame sensors and actuators to enable real-time detection and response. Emergency alerts, alarm activation, and water spray automation confirm the system's ability to comprehensively deal with fire incidents.

To create a flood detection system using IoT which would detect the flood water level and send an emergency alert, trigger the house alarm as well automatically take corrective actions to prevent flood spreading by turning on the water draining pump	Achieved	Flood detection systems effectively use IoT sensors to monitor water levels and respond quickly with emergency alerts, activation of alarms, and automatic control of sump pumps. This holistic approach demonstrates the system's ability to reduce flood-related risks.
To enable a camera view for surveillance purposes for both flood and fire detection.	Achieved	A bot created in the Telegram app is used to receive snapshots when requested by users. Incorporating camera views improves the system's monitoring capabilities and provides visual insight into both fire and flood detection. This feature adds a layer of surveillance, supports. comprehensive situational

5.2 Limitation of Study

The limitation for the project is followed:

1. This project is limited to flood and fire detection with corrective action and does not include other home automation features such as temperature control, lighting, or security.
2. Flame sensors typically have a range of a few feet (in testing, it discovered as 2 feet for the flame sensor use in the prototype) so their range can be limited by the strength of the flame and the level of ambient light in the surrounding area.
3. Ultras sound sensor can interfere from other objects in the water.

4. The ESP32 Wi-Fi module requires a strong network connectivity, otherwise the alert message to the user can get delayed and the live streaming for surveillance of the event by the camara can get disrupted.
5. The system is limited to 2 main corrective actions such as water sprinkler for the fire extinguishing and water draining pump to pump out the water in the system placed room. Any other actions such as ventilation system to act against a fire is not included.
6. Sensors can malfunction when an event occurs.
7. Ther is delay in pop-up notification and snapshot receiving due to poor signal in Wi-Fi.
8. The higher rate of water floor cannot be handled by the 12 V water draining pump so the water flooding/ filtering should be controlled in the simulation.
9. The prototype's simulation manages water within a confined space, whereas real-world flooding requires directing water away from the local area.
10. The prototype assumes dual operation on 5V and 12V power supplies, which may not align with real-world scenarios where electricity is often cut off during flooding.
11. The prototype integrates the water sprinkler system with the flood simulation tank, which may not align with real-world standards.

5.3 Recommendation for Future Work

The choice of using 12V pumps in the prototype is due to budget constraints and the challenge of creating a life-size scenario. In a real-life situation, separate water storage for the sprinkler system would be employed, and the pumped-out water during flooding would be directed to a distant reservoir, avoiding local flooding. The prototype operates on both 5V and 12V power supplies, acknowledging that, in a real-life scenario, the first measure during flooding is often to cut off electricity. Therefore, the system should have the capability to run on backup or remote power sources to ensure its functionality even in adverse conditions.

In envisioning future advancements, the current improvements in fire and flood detection algorithms lay the foundation for several recommendations. First and foremost, ongoing efforts should focus on fine-tuning the algorithms to mitigate any remaining false positives or negatives, ensuring heightened accuracy. Exploring the integration of machine learning techniques offers an avenue for adaptive learning and continuous enhancement. Consideration for sensor redundancy is crucial, providing an added layer of reliability through diverse sensing technologies like smoke sensor and use image processing detect spark in shot of circuit which then led to fire. In real-world scenarios, it is essential to use larger capacity pumps and robust hardware, especially in areas with high water flows. These improved components effectively cope with increased water pressure and ensure system resilience to cope with more severe conditions. Real-world testing in varied environments is paramount to validate the algorithms' robustness, while an improved user interface and clearer feedback mechanisms contribute to user-friendliness. Scalability, energy efficiency, and cybersecurity measures are integral components for the systems' long-term viability and resilience in the face of evolving challenges and technological landscapes.

REFERENCES

[1] "Mental Health Effects of Disasters," Centers for Disease Control and Prevention, [Online]. Available: https://www.cdc.gov/disasters/mentalhealth/index.html. [Accessed 10 04 2023].

[2] "Disaster Distress Helpline," American Psychological Association, [Online]. Available: https://www.apa.org/topics/disaster-distress-helpline. [Accessed 10 04 2023].

[3] "History of Fire Safety," Fire Protection Online, [Online]. Available: https://www.fireprotectiononline.co.uk/info/history-of-fire-safety/. [Accessed 10 04 2023].

[4] "A Brief History of Flood Control," Flood Control America, [Online]. Available: https://www.floodcontrolam.com/a-brief-history-of-flood-control/. [Accessed 10 04 2023].

[5] "History of early warning and emergency notification systems," Electronic outdoor sirens and early warning and emergency notification systems, [Online]. Available: http://www.electronicsirens.com/history-early-warning-emergency-notification-systems/. [Accessed 10 04 2023].

[6] D.Dinesh and Anette Regina, "Prediction and Effective Monitoring of Flood Using Arduino System Controller and ESP8266 Wi-Fi Module," International Journal of Communication and Networking System, vol. 08, no. 01, pp. 50 -55, 2019.

[7] Q. I. Sarhan, "Arduino Based Smart Home Warning System," IEEE 6th International Conference on Control Science and Systems Engineering, pp. 201-206, 2020.

[8] M. S. Dauda and U. S. Toro, "Arduino Based Fire Detection and Control System," International Journal of Engineering Applied Sciences and Technology, vol. 04, no. 11, pp. 447-453, 2020.

[9] M. Naing and N. N. S. Hlaing, "Arduino Based Smart Home Automation System," International Journal of Trend in Scientific Research and Development (IJTSRD), vol. 03, no. 04, pp. 276-280, 2019.

[10] J. Bangali and A. Shaligram, "Design and Implementation of Security Systems for Smart Home," International Journal of Smart Home, vol. 07, no. 06, pp. 201-208, 2013.

[11] D. Satria, S. Yana, R. Munadi and S. Syahreza, "Prototype of Google Maps-Based Flood Monitoring System Using Arduino and GSM Module," International Research Journal of Engineering and Technology (IRJET), vol. 04, no. 10, pp. 1044-1047, 2017.

[12] M. R. Habib, N. Khan, K. Ahmed, M. R. Kiran, A. Asif, M. I. Bhuiyan and O. Farrok, "Quick Fire Sensing Model and Extinguishing by Using an Arduino Based Fire Protection Device," 5th International Conference on Advances in Electrical Engineering (ICAEE), pp. 435-439, 2019.

[13] J. W. Simatupang and F. Naufal, "Flood Early Warning Detection System Prototype Based on IoT Network," Internetworking Indonesia Journal, vol. 11, no. 01, pp. 17-22, 2019.

[14] S. Azid, B. Sharma, K. Raghuwaiya, A. Chand, S. Prasad and A. Jacquier, "SMS Based Flood Monotoring and Early Warning System," ARPN Journal of Engineering and Applied Sciences, vol. 10, no. 15, pp. 6387-6391, 2015.

[15] S. Suwarjono, I. H. Wayangkau, T. Istanto, T. Istanto, M. Marsujitullah, H. Hariyanto, W. Caesarendra, S. Legutko and A. Glowacz, "Design of a Home Fire Detection System Using Arduino and SMS Gateway," knowledge by MDPI, vol. 01, no. 01, pp. 61-74, 2021.

Appendices A: Budget

Product	Quantity	RM
Node MCU	1	RM 40
Flame Sensor	3	RM 9
Ultrasound Sensor	1	RM 3
Jumper Wires	5 sets	RM15
ESP 32 FiWi module with Camara	1	Rm 20
Water Pump 12V	2	Rm 20
Water Sprinkler	1	Rm 5
Alarm 12V (Siren)	1	Rm15
Relay 3v to 12v module	1	Rm 6
Water tube pipe (hose)	2	Rm 8
PCB	1	RM4
Project assemble materials	-	RM 100
For additional expenses	1	Rm100
Total		Rm 345

Appendices B: Gantt Chart FYP 2

Activities	Weeks												
	2	3	4	5	6	7	8	9	10	11	12	13	14
Experiment and Data Analysis													
Integration of the Project													
Mid- review Progress													
The interaction with supervisor (completion of logbook according to week)													
Report Draft Submission													
Report Draft Submission													
Report Draft Submission													
FYP 2 Final Presentation													
FYP 2 report final submission													

	Completed

Appendices C: Codes

1.1 The Water Level Sensors' and Flame Sensors' Integrated Code

```
#define BLYNK_TEMPLATE_ID "TMPL6yvcmYZwJ"
#define BLYNK_TEMPLATE_NAME "NODE MCU"
#define BLYNK_AUTH_TOKEN "ks5Y1d3rKb9xLzx-PTwSt0IP4sepceLf"

#define BLYNK_PRINT Serial
#include <ESP8266WiFi.h>
#include <BlynkSimpleEsp8266.h>

char auth[] = BLYNK_AUTH_TOKEN;
char ssid[] = "Inuri123";  // type your wifi name
char pass[] = "snob0774";  // type your wifi password

const int sirenPin = D8; // Define the digital pin for the siren for
fire
const int sirenWPin = D2; // Define the digital pin for the siren
for water level
const int trigPin = D5;  // Trigger pin
const int echoPin = D6;  // Echo pin
const int d_pump = D7;   // Draining pump
const int wp_pump = D1; //  water sprinkler pump

const int level3 = 14;
const int level2 = 16;
const int level1 = 18;


int flag1 = 0;
int flag2 = 0;
int flag3 = 0;
int flag4 = 0;
int flag5 = 0;

void fireDetection() {
  int fire_sensor0 = digitalRead(D0);

  if (fire_sensor0 == 0 ) {
    Serial.println("Fire in the House");
    Blynk.logEvent("fire_detection", "Fire is Detected");
    digitalWrite(sirenPin, HIGH);
    digitalWrite(wp_pump, HIGH);
    Serial.println("Siren ON");
    delay(4000);

    flag1 = 1;
```

```
  } else if (fire_sensor0 == 1) {
    flag1 = 0;
    digitalWrite(sirenPin, LOW);
    digitalWrite(wp_pump, LOW);

  }
  return;
}

const int waterThreshold = 22; // Water level threshold value
bool needReset = false; // Global variable to indicate the need for
a reset
void waterDetection() {
  int isButtonPressed3 = analogRead(A0);
  unsigned long currentMillis = millis();

  long duration;
  int distance_cm;

  digitalWrite(trigPin, LOW);
  delayMicroseconds(2);
  digitalWrite(trigPin, HIGH);
  delayMicroseconds(10);
  digitalWrite(trigPin, LOW);

  duration = pulseIn(echoPin, HIGH);
  distance_cm = duration / 58;
  Serial.print("Water level = ");
  Serial.print(distance_cm);
  Serial.println(" cm");

  if ((distance_cm <= level3) && (distance_cm != 0)&& flag2 ==0) {
    Blynk.logEvent("water_detection", "level 3 - 14 cm");
    Serial.println("level 3");
    digitalWrite(sirenWPin, HIGH);
    Serial.println ("Siren ON");
    delay(4000);
    digitalWrite(sirenWPin, LOW);

    flag2=0;
    flag3=0;
    flag4=0;
    flag5=0;

    flag2=1;
  } else if ((distance_cm <= level2 && distance_cm > level3) &&
flag3 ==0) {
    Blynk.logEvent("water_detection", "level 2 - 14 cm");
    Serial.println("level 2");
    digitalWrite(sirenWPin, HIGH);
    Serial.println("Siren ON");
```

```
    delay(4000);
    digitalWrite(sirenWPin, LOW);
    flag3=1;
  } else if ((distance_cm <= level1 && distance_cm > level2) &&
flag4 ==0) {
    Blynk.logEvent("water_detection", "level 1 - 18 cm");
    Serial.println("level 1");
    digitalWrite(sirenWPin, HIGH);
    Serial.println("Siren ON");
    delay(4000);
    digitalWrite(sirenWPin, LOW);
    flag4=1;
  }
  else if(( distance_cm <= level2) && flag3==1){

    digitalWrite(d_pump, HIGH); // Turn ON the pump
    Serial.println("pump on");

  }
  else if (distance_cm >= waterThreshold && flag3==1) {

    digitalWrite(d_pump, LOW); // Turn OFF the pump
    Serial.println("pump off");

      flag2=0;
    flag3=0;
    flag4=0;
    flag5=0;

    return;
  }

}

void setup() {
  Serial.begin(115200);
  Blynk.begin(auth, ssid, pass);
  pinMode(D0, INPUT);
  pinMode(D1, INPUT);
  pinMode(D2, INPUT);
  pinMode(sirenPin, OUTPUT);
  pinMode(sirenWPin, OUTPUT);
  pinMode(trigPin, OUTPUT);
  pinMode(echoPin, INPUT);
  pinMode(d_pump, OUTPUT);
  pinMode(wp_pump, OUTPUT);


 delay (2000);

}
```

```
void loop() {

  Blynk.run();
  fireDetection();
  waterDetection();

}
```

1.2 The ESP 32 web Camara with Telegram Connectivity Code

```
#include <Arduino.h>
#include <WiFi.h>
#include <WiFiClientSecure.h>
#include "soc/soc.h"
#include "soc/rtc_cntl_reg.h"
#include "esp_camera.h"
#include <UniversalTelegramBot.h>
#include <ArduinoJson.h>

const char* ssid = "Inuri123";
const char* password = "snob0774";
// Initialize Telegram BOT
String BOTtoken = "6478706806:AAH_todtCijFg1jhTMzcUwMhPIAGVBjK3-I";
// your Bot Token (Get from Botfather)

// Use @myidbot to find out the chat ID of an individual or a group
// Also note that you need to click "start" on a bot before it can
// message you
String CHAT_ID = "5440097171";

bool sendPhoto = false;

WiFiClientSecure clientTCP;
UniversalTelegramBot bot(BOTtoken, clientTCP);

#define FLASH_LED_PIN 4
bool flashState = LOW;

//Checks for new messages every 1 second.
int botRequestDelay = 1000;
unsigned long lastTimeBotRan;

//CAMERA_MODEL_AI_THINKER
#define PWDN_GPIO_NUM     32
#define RESET_GPIO_NUM    -1
#define XCLK_GPIO_NUM      0
#define SIOD_GPIO_NUM     26
```

```
#define SIOC_GPIO_NUM     27

#define Y9_GPIO_NUM       35
#define Y8_GPIO_NUM       34
#define Y7_GPIO_NUM       39
#define Y6_GPIO_NUM       36
#define Y5_GPIO_NUM       21
#define Y4_GPIO_NUM       19
#define Y3_GPIO_NUM       18
#define Y2_GPIO_NUM        5
#define VSYNC_GPIO_NUM    25
#define HREF_GPIO_NUM     23
#define PCLK_GPIO_NUM     22


void configInitCamera(){
  camera_config_t config;
  config.ledc_channel = LEDC_CHANNEL_0;
  config.ledc_timer = LEDC_TIMER_0;
  config.pin_d0 = Y2_GPIO_NUM;
  config.pin_d1 = Y3_GPIO_NUM;
  config.pin_d2 = Y4_GPIO_NUM;
  config.pin_d3 = Y5_GPIO_NUM;
  config.pin_d4 = Y6_GPIO_NUM;
  config.pin_d5 = Y7_GPIO_NUM;
  config.pin_d6 = Y8_GPIO_NUM;
  config.pin_d7 = Y9_GPIO_NUM;
  config.pin_xclk = XCLK_GPIO_NUM;
  config.pin_pclk = PCLK_GPIO_NUM;
  config.pin_vsync = VSYNC_GPIO_NUM;
  config.pin_href = HREF_GPIO_NUM;
  config.pin_sscb_sda = SIOD_GPIO_NUM;
  config.pin_sscb_scl = SIOC_GPIO_NUM;
  config.pin_pwdn = PWDN_GPIO_NUM;
  config.pin_reset = RESET_GPIO_NUM;
  config.xclk_freq_hz = 20000000;
  config.pixel_format = PIXFORMAT_JPEG;
  config.grab_mode = CAMERA_GRAB_LATEST;

  //init with high specs to pre-allocate larger buffers
  if(psramFound()){
    config.frame_size = FRAMESIZE_UXGA;
    config.jpeg_quality = 10;  //0-63 lower number means higher
quality
    config.fb_count = 1;
  } else {
    config.frame_size = FRAMESIZE_SVGA;
    config.jpeg_quality = 12;  //0-63 lower number means higher
quality
    config.fb_count = 1;
  }
```

```
  // camera init
  esp_err_t err = esp_camera_init(&config);
  if (err != ESP_OK) {
    Serial.printf("Camera init failed with error 0x%x", err);
    delay(1000);
    ESP.restart();
  }
}

void handleNewMessages(int numNewMessages) {
  Serial.print("Handle New Messages: ");
  Serial.println(numNewMessages);

  for (int i = 0; i < numNewMessages; i++) {
    String chat_id = String(bot.messages[i].chat_id);
    if (chat_id != CHAT_ID){
      bot.sendMessage(chat_id, "Unauthorized user", "");
      continue;
    }

    // Print the received message
    String text = bot.messages[i].text;
    Serial.println(text);

    String from_name = bot.messages[i].from_name;
    if (text == "/start") {
      String welcome = "Welcome , " + from_name + "\n";
      welcome += "Use the following commands to interact with the
ESP32-CAM \n";
      welcome += "/photo : takes a new photo\n";
      welcome += "/flash : toggles flash LED \n";
      bot.sendMessage(CHAT_ID, welcome, "");
    }
    if (text == "/flash") {
      flashState = !flashState;
      digitalWrite(FLASH_LED_PIN, flashState);
      Serial.println("Change flash LED state");
    }
    if (text == "/photo") {
      sendPhoto = true;
      Serial.println("New photo request");
    }
  }
}

String sendPhotoTelegram() {
  const char* myDomain = "api.telegram.org";
  String getAll = "";
  String getBody = "";
```

```
  //Dispose first picture because of bad quality
  camera_fb_t * fb = NULL;
  fb = esp_camera_fb_get();
  esp_camera_fb_return(fb); // dispose the buffered image

  // Take a new photo
  fb = NULL;
  fb = esp_camera_fb_get();
  if(!fb) {
    Serial.println("Camera capture failed");
    delay(1000);
    ESP.restart();
    return "Camera capture failed";
  }

  Serial.println("Connect to " + String(myDomain));


  if (clientTCP.connect(myDomain, 443)) {
    Serial.println("Connection successful");

    String head = "--RandomNerdTutorials\r\nContent-Disposition:
form-data; name=\"chat_id\"; \r\n\r\n" + CHAT_ID + "\r\n--
RandomNerdTutorials\r\nContent-Disposition: form-data;
name=\"photo\"; filename=\"esp32-cam.jpg\"\r\nContent-Type:
image/jpeg\r\n\r\n";
    String tail = "\r\n--RandomNerdTutorials--\r\n";

    size_t imageLen = fb->len;
    size_t extraLen = head.length() + tail.length();
    size_t totalLen = imageLen + extraLen;

    clientTCP.println("POST /bot"+BOTtoken+"/sendPhoto HTTP/1.1");
    clientTCP.println("Host: " + String(myDomain));
    clientTCP.println("Content-Length: " + String(totalLen));
    clientTCP.println("Content-Type: multipart/form-data;
boundary=RandomNerdTutorials");
    clientTCP.println();
    clientTCP.print(head);

    uint8_t *fbBuf = fb->buf;
    size_t fbLen = fb->len;
    for (size_t n=0;n<fbLen;n=n+1024) {
      if (n+1024<fbLen) {
        clientTCP.write(fbBuf, 1024);
        fbBuf += 1024;
      }
      else if (fbLen%1024>0) {
        size_t remainder = fbLen%1024;
        clientTCP.write(fbBuf, remainder);
      }
```

```
    }

    clientTCP.print(tail);

    esp_camera_fb_return(fb);

    int waitTime = 10000;   // timeout 10 seconds
    long startTimer = millis();
    boolean state = false;

    while ((startTimer + waitTime) > millis()){
      Serial.print(".");
      delay(100);
      while (clientTCP.available()) {
        char c = clientTCP.read();
        if (state==true) getBody += String(c);
        if (c == '\n') {
          if (getAll.length()==0) state=true;
          getAll = "";
        }
        else if (c != '\r')
          getAll += String(c);
        startTimer = millis();
      }
      if (getBody.length()>0) break;
    }
    clientTCP.stop();
    Serial.println(getBody);
  }
  else {
    getBody="Connected to api.telegram.org failed.";
    Serial.println("Connected to api.telegram.org failed.");
  }
  return getBody;
}

void setup(){
  WRITE_PERI_REG(RTC_CNTL_BROWN_OUT_REG, 0);
  // Init Serial Monitor
  Serial.begin(115200);

  // Set LED Flash as output
  pinMode(FLASH_LED_PIN, OUTPUT);
  digitalWrite(FLASH_LED_PIN, flashState);

  // Config and init the camera
  configInitCamera();

  // Connect to Wi-Fi
  WiFi.mode(WIFI_STA);
  Serial.println();
```

```
  Serial.print("Connecting to ");
  Serial.println(ssid);
  WiFi.begin(ssid, password);
  clientTCP.setCACert(TELEGRAM_CERTIFICATE_ROOT); // Add root
certificate for api.telegram.org
  while (WiFi.status() != WL_CONNECTED) {
    Serial.print(".");
    delay(500);
  }
  Serial.println();
  Serial.print("ESP32-CAM IP Address: ");
  Serial.println(WiFi.localIP());
}

void loop() {
  if (sendPhoto) {
    Serial.println("Preparing photo");
    sendPhotoTelegram();
    sendPhoto = false;
  }
  if (millis() > lastTimeBotRan + botRequestDelay)  {
    int numNewMessages = bot.getUpdates(bot.last_message_received +
1);
    while (numNewMessages) {
      Serial.println("got response");
      handleNewMessages(numNewMessages);
      numNewMessages = bot.getUpdates(bot.last_message_received +
1);
    }
    lastTimeBotRan = millis();
  }}
```

FYP Report Plagiarism Checked. (Check with Turnitin)

fyp

ORIGINALITY REPORT

26%	21%	12%	19%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	randomnerdtutorials.com Internet Source	4%
2	Submitted to Manipal International University Student Paper	3%
3	www.ijcnes.com Internet Source	1%
4	ijeast.com Internet Source	1%
5	www.mdpi.com Internet Source	1%
6	Md. Rawshan Habib, Naureen Khan, Koushik Ahmed, Mahbubur Rahman Kiran, A.K.M. Asif, Mohaiminul Islam Bhuiyan, Omar Farrok. "Quick Fire Sensing Model and Extinguishing by Using an Arduino Based Fire Protection Device", 2019 5th International Conference on Advances in Electrical Engineering (ICAEE), 2019 Publication	1%

repository.usp.ac.fj