VISVESVARAYA TECHNOLOGICAL UNIVERSITY
BELAGAVI

Project Report on

"System to Detect Illegal Logging of Trees"

Submitted in the partial fulfillment for the requirements of the degree of

BACHELOR OF ENGINEERING
IN
COMPUTER SCIENCE AND ENGINEERING

Submitted By

FAZAL UR REHAMAN	**1BY19CS402**
MALLIKARJUN HALAGALI	**1BY19CS405**
NIKHIL M	**1BY19CS407**
PAVAN V	**1BY19CS408**

Under the guidance of

Dr. ARCHANA R A
Asst. Professor
Department of CSE, BMSIT&M

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

BMS INSTITUTE OF TECHNOLOGY & MANAGEMENT
YELAHANKA, BENGALURU - 560064.

2021-2022

VISVESVARAYA TECHNOLOGICAL UNIVERSITY
BELAGAVI

BMS INSTITUTE OF TECHNOLOGY &MANAGEMENT
YELAHANKA, BENGALURU – 560064

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CERTIFICATE

This is to certify that the Project work entitled **"System to Detect Illegal Logging of Trees"** is a bonafide work carried out by **Fazal Ur Rehaman(1BY19CS402), Mallikarjun Halagali (1BY19CS405), Nikhil M (1BY19CS407), Pavan V (1BY19CS408),** in partial fulfilllment for the award of **Bachelor of Engineering Degree in Computer Science and Engineering** of the **Visvesvaraya Technological University, Belagavi** during the year 2021-2022. It is certified that all corrections/suggestions indicated for Internal Assessment have been incorporated in this report. The project report has been approved as it satisfies the academic requirements in respect of project work for B.E Degree.

Signature of the Guide	**Signature of the HOD**	**Signature of Principal**
Dr. Archana R A Asst. Professor Dept. of CSE, BMSIT&M	Dr. Thippeswamy G Professor & HOD, Dept. of CSE, BMSIT&M	Dr. Mohan Babu G N Principal, BMSIT&M

External VIVA-VOCE

Name of the Examiners	Signature with Date
1.	
2.	

ACKNOWLEDGEMENT

We are happy to present this project after completing it successfully. This project would not have been possible without the guidance, assistance and suggestions of many individuals. We would like to express our deep sense of gratitude and indebtedness to each and every one who has helped us make this project a success.

We heartily thank our Principal**, Dr. MOHAN BABU G N, BMS Institute of Technology & Management,** for his constant encouragement and inspiration in taking up this project.

We heartily thank our **Head of the Department, Dr. Thippeswamy G, Department of Computer Science and Engineering, BMS Institute of Technology &Management,** for his constant encouragement and inspiration in taking up this project.

We gracefully thank our Project Guide, **Dr. Archana R A, Asst. Professor, Department of Computer Science and Engineering,** for her guidance, support and advice.

Special thanks to all the staff members of Computer Science Department for their help and kind co-operation.

Lastly we thank our parents and friends for the support and encouragement given to us in completing this precious work successfully.

FAZAL UR REHAMAN	**1BY19CS402**
MALLIKARJUN HALAGALI	**1BY19CS405**
NIKHIL M	**1BY19CS407**
PAVAN V	**1BY19CS408**

ABSTRACT

Smuggling/theft of most important trees such as sandal wood in forests, poses a serious threat to forest resources, causes significant economic damage and ultimately has quite a devastating effect on the environment all over the world. This paper proposes a microcontroller based anti-poaching system employing WSN technology, which is capable of detecting theft by monitoring the vibrations produced by the cutting of trees/branches using a 3-axis MEMS accelerometer. A microcontroller is used along with PC so that the information can be uploaded. WSN is widely used technology in remote monitoring applications. The embedded system architecture and the hardware/software designs are described in detail. Vibration data collected by various tests on wood and simulated using Arduino IDE.

DECLARATION

We, **FAZAL UR REHAMAN, MALLIKARJUN HALAGALI, NIKHIL M, PAVAN V**, students of 8th semester B.E , in the Department of Computer Science and Engineering, BMS Institute of Technology and Management, Bengaluru declare that the project work entitled "**System to Detect Illegal Logging of Trees**" has been carried out by us and submitted in partial fulfilment of the course requirements for the award of degree in Bachelor of Engineering in Computer Science and Engineering of Visvesvaraya Technological University, Belagavi during the academic year 2021-2022. The matter embodied in this report has not been submitted to any other university or institution for the award of any other degree or diploma.

FAZAL UR REHAMAN	**1BY19CS402**
MALLIKARJUN HALAGALI	**1BY19CS405**
NIKHIL M	**1BY19CS407**
PAVAN V	**1BY19CS408**

Department of Computer Science and Engineering,

BMS Institute of Technology and Management

Bengaluru -560064

TABLE OF CONTENTS

LIST OF FIGURES

Chapter 1:

INTRODUCTION

Chapter 1

INTRODUCTION

Illegal logging has been recognized as a major problem worldwide, with studies indicating that more than 100 million cubic meters of timber are still being cut illegally each year. In addition, the destruction of the world's forests contributes up to 20% to global carbon dioxide emissions. Forest fires are as old as the forests themselves. when there is no rain for months during summer, the forests become littered with dry leaves and twinges, which could burst into flames initiated by even a slight spark They pose a threat not only to the forest wealth but also to the entire regime to fauna and flora seriously disrupting the ecosystem and biodiversity of a region, this poses a great danger to wildlife and domestic crops nearby. So, it is necessary to avoid the losses due to forest fire by controlling the fire in its early stages.

This project aims to detect illegal logging using sound and vibration sensors, fire and smoke detection using flame and smoke sensors respectively, to provide a warning that illegal logging and/or fire may be occurring, And the system that can help protect forests and the species that inhabit them.

1.1 Background

Illegal logging has been recognized as a major problem worldwide, with studies indicating that more than 100 million cubic meters of timber are still being cut illegally each year. In addition, the destruction of the world's forests contributes up to 20% to global carbon dioxide emissions. Forest fires are as old as the forests themselves. when there is no rain for months during summer, the forests become littered with dry leaves and twinges, which could burst into flames initiated by even a slight spark They pose a threat not only to the forest wealth but also to the entire regime to fauna and flora seriously disrupting the ecosystem and biodiversity of a region, this poses a great danger to wildlife and domestic crops nearby. So, it is necessary to avoid the losses due to forest fire by controlling the fire in its early stages.

1.2 Literature Survey

The environmental effects of illegal logging include deforestation, the loss of biodiversity and the emission of greenhouse gases. Illegal logging has contributed to conflicts with indigenous and local populations, violence, corruption and the worsening of poverty.

"The World Bank estimates that governments worldwide lose between US$ 10 billion and 15 billion each year as a result of illegal logging"

While other people and countries may think that it only affects the producing countries, long term climatic, economic, and environmental problems will affect most of the countries in the world. One example is the illegal logging in the Amazon forest, which is projected to lessen rain in northern China and the Midwest US while increasing rain in Northern Europe and Eastern Africa, both with disastrous consequences.

In India According to data accessed from the Ministry of Environment, Forests and Climate Change (MoEF), a total of 10.8 lakh trees from 15 states and two UTs have been reported through 1.76 lakh cases, at an average of six trees felled under each case

Cases Detected and Trees Lost in 3 Years:

Year	Cases Detected	No. of Trees Lost
2016-17	68823	411094
2017-18	66932	356419
2018-19	40348	312588
Total	**176103**	**1080101**

Source: MoEF | States/UTs included: Andhra Pradesh, Bihar, Haryana, Himachal Pradesh, Karnataka, Kerala, Maharashtra, Punjab, Telangana, UP, West Bengal, Arunachal Pradesh, Manipur, Mizoram, Sikkim, Chandigarh.

Apart from Illegal logging, fires are also a great menace to ecologically healthy grown forests and protection of the environment. Every year, thousands of forest fires across the globe cause disasters beyond measure and description. This issue has been the research interest for many years; there are a huge amount of very well studied solutions available out there for testing or even ready for use to resolve this problem.

Hence, we can conclude that Illegal Logging and Forest Fires is driving the loss of forests the poses the biggest threat to our natural eco-system and biodiversity. It is our responsibility to reduce deforestation as much as possible to maintain balance and for a sustainable future.

Disadvantages of the System:

- Requires more man Power as everything is done manually.
- Time Consuming
- No mechanism to detect Illegal logging and Forest fire in Real-Time.

1.2.1 Existing System

The existing system consists of hiring security personals for monitoring the entire area for suspicious activity, however due to physical limitations in human it is hard to monitor the entire area continuously, thus hiring of guards proves unreliable and inadequate.

Another existing system is the installation of CCTV cameras for covering large area proves very costly and is hard to implement. Also, the latest trend for protection of trees is to tag an RF-ID to trees just like tagging an animal for knowing the whereabouts of a particular tree. However, this technology does not give the real time information while the activity is happening. Activity is detected only when the tree leaves its initial position.

System to detect Illegal Logging through ZigBee wireless module.

The system is implemented using Arduino Micro controller along with ZigBee Wireless module. The system is used to detect illegal logging using ZigBee wireless technology to transfer data to cloud.

Disadvantages of the System:

- Does not send alert message to responsible personal but sends notification to the host website; user is required to keep tabs of notifications.
- No real time response and action.

1.2.2 Proposed System

The Smart Module is placed on tress in forests that require monitoring, the sensors on the module continuously collect and analyse the data every five seconds. The following steps depict the step-by-step working of the smart module. The Temperature and Humidity sensors continuously sense the environment for change in temperature and humidity and transmit this data to ThingSpeak and Blynk via Wi-Fi [1]. The Flame sensor activates the buzzer and relay when it senses fire nearby, the relay is used to control a water pump to sprinkle water in the vicinity as a precautionary measure from stopping wild-fires [2]. An alert notification and e-mail are sent to the people responsible (forest officers) [6]. The Sound sensor is used to measure the sound frequency, the average of peaks (for a period of 10 seconds) is calculated and compared with the predefined frequency range of chainsaw sounds [4]. If the measured sound frequency falls within the defined range then the buzzer is activated and an alert notification and e-mail are sent to the people responsible like forest officers to take immediate action. The Tilt/Vibration sensor is used to detect frequent vibrations when the tree is being cut or when trees fall. When the sensor picks up vibrations it sends an alert to the people responsible (forest officers). The Smart Module continuously sends data to Blynk and ThingSpeak platforms where it can be monitored and analysed. The Blynk app can be used to control the smart module remotely as it is connected to the internet [7]. It can be used to turn ON/OFF the water pump and buzzer.

Advantages of Proposed System:

- Send alert message to all responsible personal via Blynk and email.
- Combining sound and tilt/vibration sensors creates a powerful combination that can effectively address incidents of unlawful logging.
- Cloud-based server helps analyse the collected data and sends alerts to responsible people/department.
- Using Blynk app can control the smart module remotely as it is connected to the internet. It can be used to turn ON/OFF the water pump and buzzer.

1.3 Motivation

Illegal logging has been identified as a major problem in the world, which may be minimized through effective monitoring of forest covered areas. The manual monitoring of the forest to prevent unauthorized activities is practically difficult job. The system is capable of recognizing illegal logging detection using sound and vibration sensor, fire

and smoke detection, temperature and humidity detection, flame sensor, smoke sensor, and soil sensor respectively, in protected areas, thus warning that illegal logging and fire may be occurring, and system that can help protect forests and the species that inhabit them.

1.4 Problem Statement

- Illegal logging refers to what in forestry might be called timber theft by the timber mafia.
- It refers to the harvesting, transportation, purchase or sale of timber in violation flaws.
- Selective logging almost often diseased or malformed trees.
- As a preventive measure to the above problem, the system based on IoT that can be used to avoid the smuggling of the trees which would stop the deforestation.

1.5 Aim and Objective

This project aims to detect illegal logging using sound and vibration sensors, fire and smoke detection using flame and smoke sensors respectively, to provide a warning that illegal logging and/or fire may be occurring, and to use this value when analysing the potentials of an anti-poaching project like Project Ngulia. Project Ngulia is an NPO and is situated in Tsavo West National Park in Kenya, in a sanctuary named Ngulia. The project provides a bottom-up technological security and a border control solution, which also meets sustainability requirements.

Within the thesis an estimate valuation of a rhino's value is calculated. This information is used in the argumentation for wildlife conservation and to identify the market potential. The market potential is presented in a strategic plan and the conclusion made is that the venture will have to expand to other areas that are in need of a new innovative security solution, and thereby possibly save the rhinos and other endangered animals.

1.6 Scope

Although the design was successful there are improvements that could be made in future adaptations of this project. The future scope of work is implementation of multi-node network and incorporation of microphone motion detector sensor to make systems more effective to acquire data such human or animal interference.

1.7 Challenges

- **High Maintenance:** The smart device consists of various different sensors and actuators working together so failure of sensors can lead to performance issues and wrong outputs.
- **The size and scale of the rainforest:** In India rain forests cover about 30,000 Square Kilometres, covering such a vast area is the biggest challenge.
- **Moving throughout the terrain:** The forest land has many irregularities so navigation and deciding on installation points is a difficult task.
- **Sound Detection:** Differentiating the chainsaw sound from other environment sounds is also a challenge; improper sound detection can cause false alarms.
- **Power Supply**: The Smart device runs on battery and maintaining battery can be a hectic task.
- **Network connectivity:** Network connectivity is one of the most common challenges faced by smart device owners. When smart devices stop communicating with each other due to network problems, it can be annoying and seriously disruptive.

Chapter 2:
OVERVIEW

Chapter 2

OVERVIEW

2.1 Arduino IDE

Arduino is an open-source electronics platform based on easy-to-use hardware and software. Arduino boards are able to read inputs - light on a sensor, a finger on a button, or a Twitter message - and turn it into an output - activating a motor, turning on an LED, publishing something online. You can tell your board what to do by sending a set of instructions to the microcontroller on the board. To do so you use the Arduino programming language (based on Wiring), and the Arduino Software (IDE), based on Processing.

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

Arduino was born at the Ivrea Interaction Design Institute as an easy tool for fast prototyping, aimed at students without a background in electronics and program- ming. As soon as it reached a wider community, the Arduino board started changing to adapt to new needs and challenges, differentiating its offer from simple 8-bit boards to products for IoT applications, wearable, 3D printing, and embedded environments. All Arduino boards are completely open-source, empowering users to build them in- dependently and eventually adapt them to their particular needs. The software, too, is open-source, and it is growing through the contributions of users worldwide.

Arduino is a computer hardware and software company, project, and user community that designs and manufactures microcontroller kits for building digital devices and interactive objects that can sense and control objects in the physical world. The project's products are distributed as open-source hardware and software, which are licensed under the GNU Lesser General Public License (LGPL) or the GNU General Public License (GPL), permitting the manufacture of Arduino boards and software distribution by anyone. Arduino boards are available commercially in preassembled form, or as do-it-yourself kits.

The project's board designs use a variety of microprocessors and controllers. These systems provide sets of digital and analog input/output (I/O) pins that may be interfaced to various expansion boards ("shields") and other circuits. The boards feature serial communications interfaces, including Universal Serial Bus (USB) on some models, for loading programs from personal computers. The microcontrollers are mainly programmed using a dialect of features from the programming languages C and C++.

2.2 Why Are We Using Node MCU ESP8266?

NodeMCU is an open-source firmware for which open-source prototyping board designs are available. The name "NodeMCU" combines "node" and "MCU" (microcontroller unit). The term "NodeMCU" strictly speaking refers to the firmware rather than the associated development kits. Both the firmware and prototyping board designs are open source. Nodemcu ESP8266 and Nodemcu ESP32 are becoming very popular and are almost used in more than 50% IoT based projects today.

The firmware uses the Lua scripting language. The firmware is based on the eLua project and built on the Espressif Non-OS SDK for ESP8266. It uses many open-source projects, such as lua-cjson and SPIFFS. Due to resource constraints, users need to select the modules relevant for their project and build a firmware tailored to their needs. Support for the 32-bit ESP32 has also been implemented.

The prototyping hardware typically used is a circuit board functioning as a dual in-line package (DIP) which integrates a USB controller with a smaller surface-mounted board containing the MCU and antenna. The choice of the DIP format allows for easy prototyping on breadboards. The design was initially was based on the ESP-12 module of the ESP8266, which is a Wi-Fi SoC integrated with a Tensilica Xtensa LX106 core, widely used in IoT applications.

Reasons Why ESP8266 has Been So Popular

- **Inexpensive** - Yes, it is cheap. Like what this article mentions, $5 ESP8266 is even cheaper than the sensors-launched-out-of-a-cannon. It means the sensors cannons used are usually made with low-cost approaches because it will be useless after cannons launched. Even though, ESP8266 is still a lot cheaper than the sensor.
- **Do Something Different** - The 2nd step is to do something different. Even if you break it, you won't feel you are wasting money. So, some people started trying to

apply the chip in “unusual” ways. In general, Wi-Fi transmission is usually used between 60 to 140 meters, but some people apply it to 366 meters (PCB antenna), 479 meters (exterior antenna). And then they uploaded the video to YouTube (2014, video) and had more people to know ESP8266.

- **Incentive program** - Online incentive programs also accelerated the ESP8266 software development. For example, anyone who can implement some function or application with ESP8266 will get the prize. Besides, Expressif also has an incentive program that once anyone who finds bug in the SDK and got approve will get cash USD$1000 – 2000 in cash prize.
- **Active maker community** - Like what we’ve mentioned above, both official and unofficial maker communities eagerly share and discuss the related topics. The active communities will create a positive cycle on the conditions above. So far the official doesn’t need to hold any campaigns but the community will have their own activities.
- **Abundant Learning Resources** - ESP8266 is from Expressif, located in Shanghai so that there’s only chip information available in Chinese at the beginning. But as it became more famous, the related information was translated to English, helping the global promotion.
- **Flexible Design and Enhanced Function** - Besides the software progressive contributed by the community, ESP8266 chip and board have been enhanced as well, like chip clock rate acceleration, a new analog digital converter (ADC) to improve the sensitivity. Connecting with soil humidity sensor is a common application.
- **More compatible development environments** - Since more and more people knew, bought, and tried using ESP8266, they found the disadvantage of the chip and tried to improve it. For example, to develop ESP8266, people had to understand FreeRTOS and coded with the professional C language. But an Arduino IDE plug-in was then developed so that developers can use Arduino IDE to write ESP8266 command code to diminish the difficulty of development.

2.3 NodeMCU Pinout and Functions

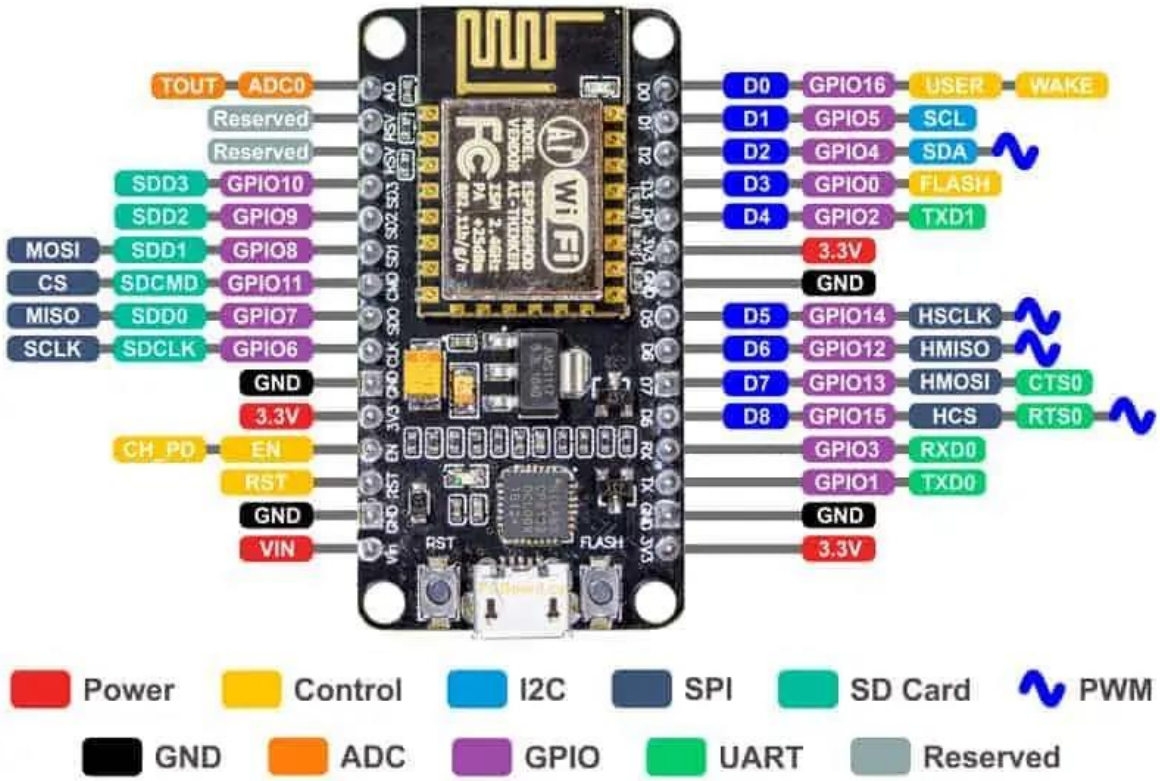

Figure 2.1: NodeMCU ESP8266 pinDiagram

- Power Pins There are four power pins. **VIN** pin and three **3.3V** pins. **VIN** can be used to directly supply the NodeMCU/ESP8266 and its peripherals. Power delivered on **VIN** is regulated through the onboard regulator on the NodeMCU module – you can also supply 5V regulated to the **VIN** pin. **3.3V** pins are the output of the onboard voltage regulator and can be used to supply power to external components.
- GND are the ground pins of NodeMCU/ESP8266.
- I2C Pins are used to connect I2C sensors and peripherals. Both I2C Master and I2C Slave are supported. I2C interface functionality can be realized programmatically, and the clock frequency is 100 kHz at a maximum. It should be noted that I2C clock frequency should be higher than the slowest clock frequency of the slave device.
- GPIO Pins NodeMCU/ESP8266 has 17 GPIO pins which can be assigned to functions such as I2C, I2S, UART, PWM, IR Remote Control, LED Light and Button programmatically. Each digital enabled GPIO can be configured to internal

pull-up or pull-down, or set to high impedance. When configured as an input, it can also be set to edge-trigger or level-trigger to generate CPU interrupts.

- ADC Channel The NodeMCU is embedded with a 10-bit precision SAR ADC. The two functions can be implemented using ADC. Testing power supply voltage of VDD3P3 pin and testing input voltage of TOUT pin. However, they cannot be implemented at the same time.
- UART Pins NodeMCU/ESP8266 has 2 UART interfaces (UART0 and UART1) which provide asynchronous communication (RS232 and RS485), and can communicate at up to 4.5 Mbps. UART0 (TXD0, RXD0, RST0 & CTS0 pins) can be used for communication. However, UART1 (TXD1 pin) features only data transmit signal so, it is usually used for printing log.
- SPI Pins NodeMCU/ESP8266 features two SPIs (SPI and HSPI) in slave and master modes. These SPIs also support the following general-purpose SPI features: 4 timing modes of the SPI format transfer, Up to 80 MHz and the divided clocks of 80 MHz, Up to 64-Byte FIFO
- SDIO Pins NodeMCU/ESP8266 features Secure Digital Input/Output Interface (SDIO) which is used to directly interface SD cards. 4-bit 25 MHz SDIO v1.1 and 4-bit 50 MHz SDIO v2.0 are supported.
- PWM Pins The board has 4 channels of Pulse Width Modulation (PWM). The PWM output can be implemented programmatically and used for driving digital motors and LEDs. PWM frequency range is adjustable from 1000 µs to 10000 µs (100 Hz and 1 kHz).
- Control Pins are used to control the NodeMCU/ESP8266. These pins include Chip Enable pin (EN), Reset pin (RST) and WAKE pin.
- **EN:** The ESP8266 chip is enabled when EN pin is pulled HIGH. When
- **RST:** RST pin is used to reset the ESP8266 chip.
- **WAKE:** Wake pin is used to wake the chip from deep-sleep.
-  Control Pins are used to control the NodeMCU/ESP8266. These pins include Chip Enable pin (EN), Reset pin (RST) and WAKE pin.

- **EN:** The ESP8266 chip is enabled when EN pin is pulled HIGH. When pulled LOW the chip works at minimum power.
- **RST:** RST pin is used to reset the ESP8266 chip.
- **WAKE:** Wake pin is used to wake the chip from deep-sleep.

2.4 INTERNET OF THINGS

IoT (Internet of Things) is an advanced automation and analytics system which exploits networking, sensing, big data, and artificial intelligence technology to deliver complete systems for a product or service. These systems allow greater transparency, control, and performance when applied to any industry or system.

IoT systems have applications across industries through their unique flexibility and ability to be suitable in any environment. They enhance data collection, automation, operations, and much more through smart devices and powerful enabling technology. IoT systems allow users to achieve deeper automation, analysis, and integration within a system. They improve the reach of these areas and their accuracy. IoT utilizes existing and emerging technology for sensing, networking, and robotics.

IoT exploits recent advances in software, falling hardware prices, and modern attitudes towards technology. Its new and advanced elements bring major changes in the delivery of products, goods, and services; and the social, economic, and political impact of those changes.

2.4.1 Features of IOT

The most important features of IoT include artificial intelligence, connectivity, sensors, active engagement, and small device use. A brief review of these features is given below

- AI IoT essentially makes virtually anything smart, meaning it enhances every aspect of life with the power of data collection, artificial intelligence algorithms, and networks. This can mean something as simple as enhancing your refrigerator and cabinets to detect when milk and your favorite cereal run low, and to then place an order with your preferred grocer.
- Connectivity New enabling technologies for networking, and specifically IoT networking, mean networks are no longer exclusively tied to major providers.

Networks can exist on a much smaller and cheaper scale while still being practical. IoT creates these small networks between its system devices.

- Sensors IoT loses its distinction without sensors. They act as defining instruments which transform IoT from a standard passive network of devices into an active system capable of real-world integration.
- Active Engagement Much of today's interaction with connected technology happens through passive engagement. IoT introduces a new paradigm for active content, product, or service engagement.
- Small Devices Devices, as predicted, have become smaller, cheaper, and more powerful over time. IoT exploits purpose-built small devices to deliver its precision, scalability, and versatility.

2.4.2 IOT Advantages

The advantages of IoT span across every area of lifestyle and business. Here is a list of some of the advantages that IoT has to offer

- Improved Customer Engagement Current analytics suffer from blind-spots and significant flaws in accuracy; and as noted, engagement remains passive. IoT completely transforms this to achieve richer and more effective engagement with audiences.
- Technology Optimization The same technologies and data which improve the customer experience also improve device use, and aid in more potent improvements to technology. IoT unlocks a world of critical functional and field data.
- Reduced Waste IoT makes areas of improvement clear. Current analytics give us superficial insight, but IoT provides real-world information leading to more effective management of resources.
- Enhanced Data Collection Modern data collection suffers from its limitations and its design for passive use. IoT breaks it out of those spaces, and places it exactly where humans really want to go to analyze our world. It allows an accurate picture of everything.

2.4.3 IOT Software

IoT software addresses its key areas of networking and action through platforms, embedded systems, partner systems, and middleware. These individual and master applications are responsible for data collection, device integration, real-time analytics, and application and process extension within the IoT network. They exploit integra- tion with critical business systems (e.g., ordering systems, robotics, scheduling, and more) in the execution of related tasks.

- **Data Collection**

  This software manages sensing, measurements, light data filtering, light data security, and aggregation of data. It uses certain protocols to aid sensors in connecting with real-time, machine-to-machine networks. Then it collects data from multiple devices and distributes it in accordance with settings.

- **Device Integration**

  Software supporting integration binds (dependent relationships) all system de- vices to create the body of the IoT system. It ensures the necessary cooperation and stable networking between devices. These applications are the defining software technology of the IoT network because without them, it is not an IoT system. They manage the various applications, protocols, and limitations of each device to allow communication.

- **Real-Time Analytics**

  These applications take data or input from various devices and convert it into viable actions or clear patterns for human analysis. They analyze information based on various settings and designs in order to perform automation-related tasks or provide the data required by industry.

- **Application and Process Extension**

  These applications extend the reach of existing systems and software to allow a wider, more effective system. They integrate predefined devices for specific purposes such as allowing certain mobile devices or engineering instruments access. It supports improved productivity and more accurate data collection.

2.4.4 IOT Technology and Protocols

IoT primarily exploits standard protocols and networking technologies. However, the major enabling technologies and protocols of IoT are RFID, NFC, low-energy Bluetooth, low-energy wireless, low-energy radio protocols, LTE-A, and WiFi-Direct. These technologies support the specific networking functionality needed in an IoT system in contrast to a standard uniform network of common systems.

- NFC and RFID

  RFID (radio-frequency identification) and NFC (near-field communication) pro- vide simple, lowenergy, and versatile options for identity and access tokens, con- nection bootstrapping, and payments.

  – RFID technology employs 2-way radio transmitter-receivers to identify and track tags associated with objects.
  – NFC consists of communication protocols for electronic devices, typically a mobile device and a standard device.
  – Low-Energy Bluetooth This technology supports the low-power, long-use need of IoT function while exploiting a standard technology with native support across systems.

- Radio Protocols

  ZigBee, Z-Wave, and Thread are radio protocols for creating low-rate private area networks. These technologies are low-power, but offer high throughput unlike many similar options. This increases the power of small local device networks without the typical costs.

- LTE-A

  LTE-A, or LTE Advanced, delivers an important upgrade to LTE technology by increasing not only its coverage, but also reducing its latency and raising its throughput. It gives IoT a tremendous power through expanding its range, with its most significant applications being vehicle, UAV, and similar communication.

- WiFi-Direct

  WiFi-Direct eliminates the need for an access point. It allows P2P (peer-to-peer) connections with the speed of WiFi, but with lower latency. WiFi-Direct

eliminates an element of a network that often bogs it down, and it does not compromise on speed or throughput.

2.4.5 ARCHITECTURE OF IOT

IoT is a three layer architecture. The layers include:

1. Perception Layer
2. Network Layer
3. Application Layer

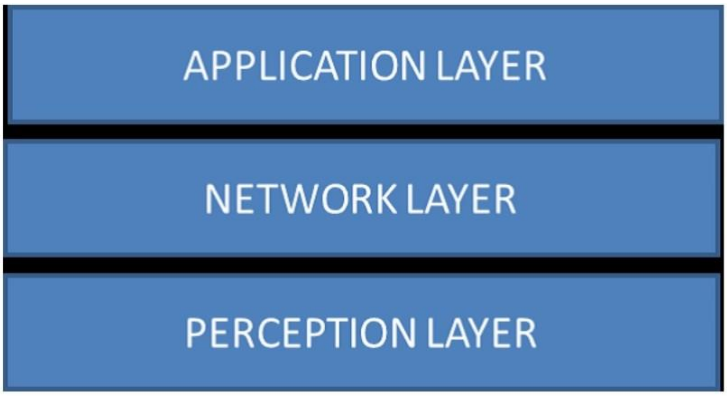

Figure 2.2: Internet of Things Architecture

- **Perception Layer**

It is the first layer of IoT architecture.It is mainly used for identifying objects and collecting information. It is tied to the hardware device like a GPS, sensor, RFID tags, or sensor network and linked to any intelligent system. It also called physical layer as the information from the physical devices is changed into a digital signal that is suitable for network transmission. The primary work of this layer is to gather information from the sensing technology

- **Network Layer**

It is a second layer of the IOT architecture .Its main function is to conduct and obtain data or information. It is a network management center for IoT. It gains data or information from the perception layer that has been collected and transferred to different networks via wired or wireless network. It also transfers huge amount of data between dissimilar networks.

- **Application Layer**

It is a third layer of the IoT architecture. It ties the application to the network. The application layer uses the processed data sent by the network Layer. In fact, this layer

constitutes the front end of the whole IoT architecture through which IoT potential will be exploited.

2.5 Sensors

A sensor is a device that detects and responds to some type of input from the physical environment. The specific input could be light, heat, motion, moisture, pressure, or any one of a great number of other environmental phenomena. The output is generally a signal that is converted to human-readable display. These sensors send their output to micro-controllers

Temperature Sensor

An analog temperature sensor is pretty easy to explain, it's a chip that tells you what the ambient temperature is! .These sensors use a solid-state technique to determine the temperature. That is to say, they don't use mercury (like old thermometers), bimetallic strips (like in some home thermometers or stoves), nor do they use thermistors (temperature sensitive resistors). Instead, they use the fact as temperature increases, the voltage across a diode increases at a known rate. (Technically, this is actually the voltage drop between the base and emitter - the Vbe - of a transistor. By precisely amplifying the voltage change, it is easy to generate an analog signal that is directly proportional to temperature. There have been some improvements on the technique but, essentially that is how temperature is measured.

DH11 Sensor Specifications

The most commonly used Temperature and humidity sensor is DHT11. The Sensor comes with a built-in NTC to measure temperature and an 8-bit microcontroller to output the values of temperature and humidity as serial data. The sensor is factory calibrated and easy to integrate with other microcontrollers.

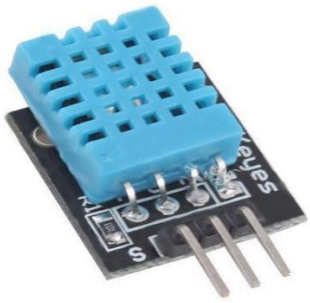

Figure 2.3:DH11 Sensor Pinouts and Packaging

DH11 Sensor Background and Applications

DHT11 Temperature & Humidity Sensor features a temperature & humidity sensor complex with a calibrated digital signal output. By using the exclusive digital-signal-acquisition technique and temperature & humidity sensing technology, it ensures high reliability and excellent long-term stability. This sensor includes a resistive-type humidity measurement component and an NTC temperature measurement component, and connects to a high-performance 8-bit microcontroller, offering excellent quality, fast response, anti-interference ability and cost-effectiveness

Each DHT11 element is strictly calibrated in the laboratory that is extremely accurate on humidity calibration. The calibration coefficients are stored as programmes in the OTP memory, which are used by the sensor's internal signal detecting process. The single-wire serial interface makes system integration quick and easy. Its small size, low power consumption and up-to-20 meter signal transmission making it the best choice for various applications, including those most demanding ones. The component is 4-pin single row pin package. It is convenient to connect and special packages can be provided according to users' request.

Tilt Sensor

The Paris Air show is one of the biggest platforms for various airplane manufacturers to showcase the swiftness of their planes. Some of the valiant pilots of the world perform amazing stunts. These pilots are supported by the power of fast computing machinery in their airplanes. One of the critical parts of this computer assisted circuitry is the tilt sensor. A type of transducer, tilt sensor aids in giving information about the vertical as well as horizontal inclination of the airplane so that the pilot can understand how can he tackle the obstacles during the flight and perform the stunts.

Keeping the pilots informed about the current orientation of the plane, the angle at which they are inclined to earth's surface, tilt sensors play a very important role in decision making for the pilots. These types of transducer produce an electric signal proportional to the degree of inclination with respect one or multiple axes. This article will detail more about tilt sense or, the types of tilt sensors, need and their applications. Lets dig in deep to know about tilt sensors.

Figure 2.4: Tilt Sensor

Tilt sensors play a vital role in numerous applications. Diversely popular in multiple fields, tilt sensors are needed in:

- Portable Computers: A popular advertisement about a portable computer says There is no wrong way or right way you hold the device. This phrase means that the display of that computer aligns itself in the manner user is holding it and he can tilt it the way he wants. Tilt sensors are required to keep the device display in the correct position with the way user holds it. Often, motion based gaming devices use this application of tilt sensor or tilt sensor derived accelerometer.
- Vehicular Security Systems: Several vehicle security alarm systems are based on tilt sensors. In the cases of unauthorized vehicle towing or movement of steering, tilt sensor engages the alarm. This in turn produces an electric signal and the alarm starts sounding.
- Aviation: In airplanes and helicopters, tilt sensors, along with inclinometers form an altitude monitoring system through which the pilot can monitor the inclinations of the flying machine.
- Robotics: For any type of robot, balance is one of the most important criteria to be taken care of. Whenever a robot inclines to any direction, tilt sensor aids in giving details about corresponding incline in form of electrical signals. It thus makes the robot judge on its own that whether it is supposed to get aligned or not.

Relays

A relay is usually an electro-mechanical device that is actuated by an electrical current. The current flowing in one circuit causes the opening or closing of another circuit. Relays are like remote control switches and are used in many applications because of their relative simplicity, long life, and proven high reliability.Relays are used where it is necessary to control a circuit by a separate low-power signal,or where several circuits must be controlled by one signal.

Working of Relays

All relays contain a sensing unit, the electric coil, which is powered by AC or DC current. When the applied current or voltage exceeds a threshold value, the coil activates the armature, which operates either to close the open contacts or to open the closed contacts. When a power is supplied to the coil, it generates a magnetic force that actuates the switch mechanism. The magnetic force is, in effect, relaying the action from one circuit to another. The first circuit is called the control circuit; the second is called the load circuit.

Functions of Relays:

There are three basic functions of a relay:

- On/Off Control
- Limit Control
- Logic Operation

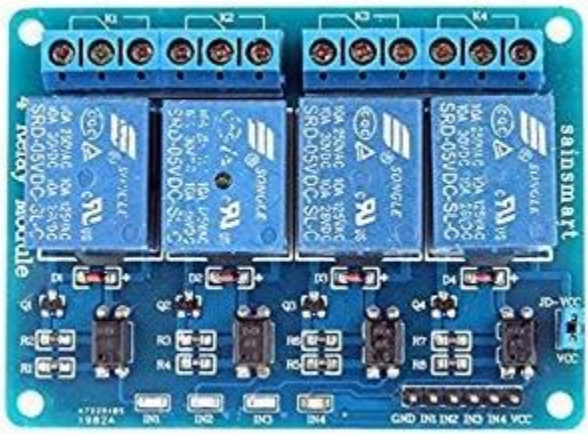

Figure 2.5: Relay Board

Chapter 3:

REQUIREMENT SPECIFICATION

Chapter 3

REQUIREMENT SPECIFICATION

3.1 Introduction

This chapter describes about the requirements. It specifies the hardware and softwarerequirements that are required in order to run the application properly. The Software Requirement Specification (SRS) is explained in detail, which includes overview of dissertation as well as the functional and non-functional requirement of this dissertation. An SRS document describes all data, functional and behavioural requirements of the software under production or development. It not only lists the requirementsof a system but also has a description of its major feature. Requirement Analysisin system engineering and software engineering encompasses those tasks that go intodetermining the need or conditions to meet for a new or altered product, taking ac-count of the possibly conflicting requirements of the various stakeholders, such as beneficiaries or users. Requirement Analysis is critical to the success to a development project. Requirement must be documented, measurable, testable, related to inidentified business needs or opportunities, and defined to a level of detail sufficient for System design.

The SRS functions as a blueprint for completing a project. The SRS is often referred to as the parent document because all subsequent project management documents, such as design specifications, statements of work, software architecture specification, testing and validation plans, and documentation plans, are related to it. It is important to note that an SRS contains functional and non-functional requirements only. Thus the goal of preparing the SRS document is to:

- To facilitate communication between the customer, analyst, system developers, maintainers.
- To serve as a contrast between purchaser and supplier.
- To firm foundation for the design phase.
- Support system testing facilities.
- Support project management and control.
- Controlling the evolution of the system.

3.2 Functional Requirements

Functional Requirement defines a function of a software system and how the system must behave when presented with specific inputs or conditions. These may include calculations, data manipulation and processing and other specific functionality. In this system following are the functional requirements: -

- Accurately measuring the environments temperature, tilt angle.
- The micro-controller must update the relay to change the status of each component
- Sending the data from the sensors to the application on detection.
- The application must not stop working when kept running for even a long time.
- The application should generate on-demand services.

3.3 Non-Functional Requirements

- **Response Time:** the time taken by the system to load and the time for responses on any action that happens to the trees.
- **Reliability:** It is the capability of the software to maintain its level of performance when used under specified conditions. Data accuracy is checked to make sure sensors do not send wrong data. Time accuracy is achieved by programming the hardware and sensor.
- **Portability:** It is the capability of software to be transferred from one environment to an- other. The code can work on any platform efficiently as Arduino can run on any platform.
- **Performance:** Power is an important factor when it comes to measurement of performance. The power for NodeMCU is via the USB. The power for the control measures implemented (fan, pump, light, sprinkler) is given via the battery. The battery must be connected to main supply for system to work.
- **Robustness:** This software is being developed in such a way that the overall performance is optimized and the user can expect the results within a limited time with utmost relevancy and correctness.
- **Ease of Use :** The bylnk app is designed in such a way that the user can understand in an easy manner

3.4 User Requirements

The user requirements document (URD) or user requirements specification is a document usually used to software engineering that specifies the requirements the user expects from software to be constructed in a software project. Once the required information is completely gathered it is documented in a URD, which is meant to spell out exactly what the software must do and becomes part of the contractual agreement. A customer cannot demand feature not in the URD, whilst the developer cannot claim the product is ready if it does not meet an item of the URD. The URD can be used as a guide to planning cost, timetables, milestones, testing etc. The explicit nature of the URD allows customers to show it to various stakeholders to make sure all necessary features are described. Formulating a URD requires negotiation to determine what is technically and economically feasible. Preparing a URD is one of those skills that lies between a science and economically feasible. Preparing a URD is one of those skillsthat lies between a science and an art, requiring both software technical skills and interpersonal skills.

3.5 Basic Operational Requirements

Operational requirement is the process of linking strategic goals and objectives to tactic goals and objectives. It describes milestones, conditions for success and explains how, or what portion of, a strategic plan will be put into operation during a given operational period, in the case of, a strategic plan will be put into operation during a given operational period, in the case of commercial application, a fiscal year or another given budgetary term. An operational plan is the basis for, and justification of an annual operating budget request.

Therefore, a five-year strategic plan would typically require five operational plans funded by five operating budgets. Operational plans should establish the activities and budgets for each part of the organization for the next 1-3 years. They link the strategic plan with the activities the organization will deliver and the resources required to deliver them.

An operational plan draws directly from agency and program strategic plans to describe agency and program missions and goals, program objectives, and program activities. Like a strategic plan, an operational plan addresses four questions:

- Where are we now?
- Where do we want to be?
- How do we get there?

The customers are those that perform the eight primary functions of systems engineering, with special emphasis on the operator as the key customer. Operational requirements will define the basic need and, at a minimum, will be related to these following points:

- Mission profile or scenario: It describes about the procedures used to accomplish mission objective. It also finds out the effectiveness or efficiency of the system.
- Performance and related parameters: It points out the critical system parameters to accomplish the mission
- Utilization environments: It gives a brief outline of system usage. Finds out appropriate environments for effective system operation.
- Operational life cycle: It defines the system lifetime

3.6 Hardware Requirements

- **Motors** : Water pump motor
- **Micro-Controllers** : NodeMCU ESP8266
- **Storage** : 128 KB Flash memory 32 KB
- **Sensors** : DH11 Temperature sensor,SW-520D Tilt sensor,KY-026 flame sensor and LM386 Sound sensor

For Mobile

- **Operating System** : Android or IOS
- **RAM** : 512 MB minimum
- **Internal Storage** : 100 MB
- **Internet access** : Yes

3.7 Software Requirements

- **Operating System** : Windows 10
- **Coding Language** : Embedded C
- **Tools** : Arduino IDE, ThinkSpeak and Blynk App

Chapter 4:
DETAILED DESIGN

Chapter 4

DETAILED DESIGN

4.1 System architecture

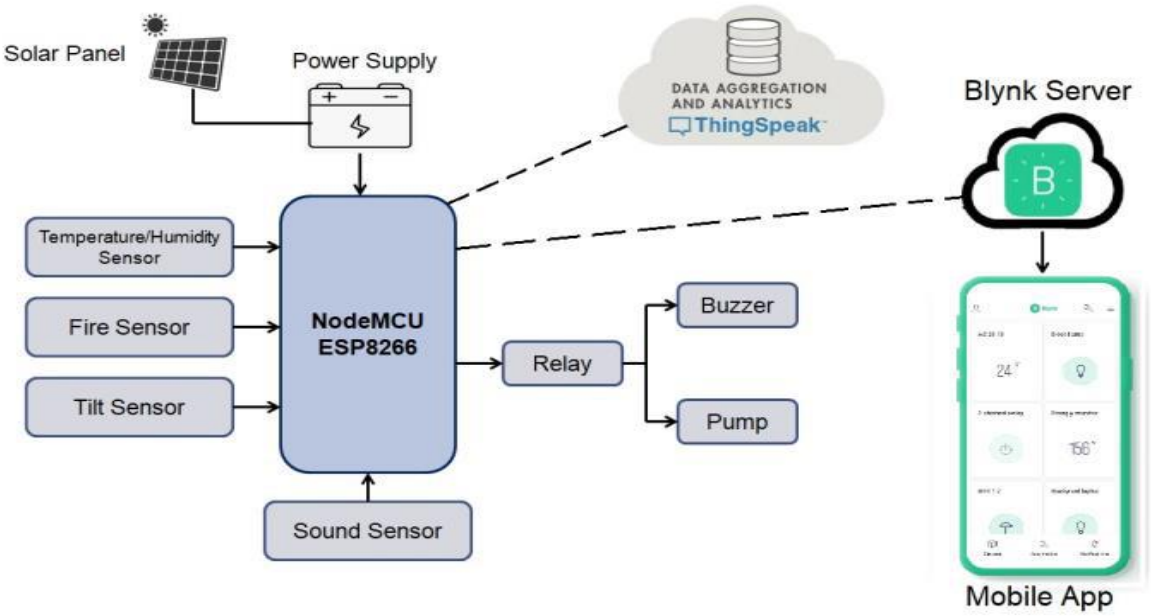

Figure 4.1: System Architecture

4.2 Methodology

The algorithm used to identify the frequency range of chainsaw sound is K-Means clustering algorithm. K-Means clustering falls under unsupervised learning algorithms. In contrast to supervised learning, this clustering does not use labelled data. K-Means divides objects into clusters that have similar characteristics and are different from objects in another cluster.

The Fig 4.2 below shows K-Means clustering of data collected from the sound sensor. The data is shown as Decibel (dBV) vs Peak-to Peak difference.

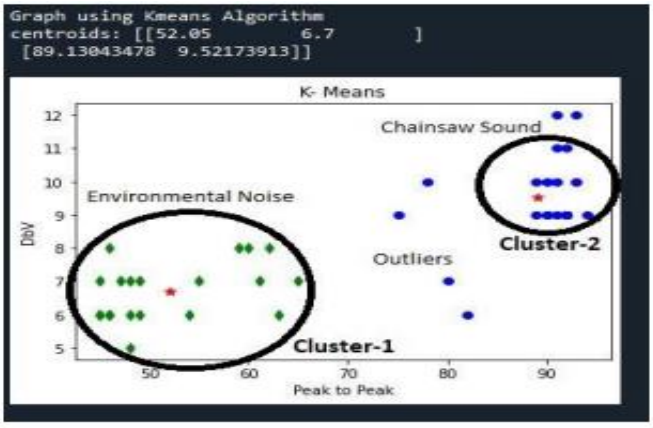

Figure 4.2: K-Means clustering of chainsaw data

Furthermore, by performing local weighted regression on positive dataset i.e only on chain-saw sound we can narrow down the frequency range.

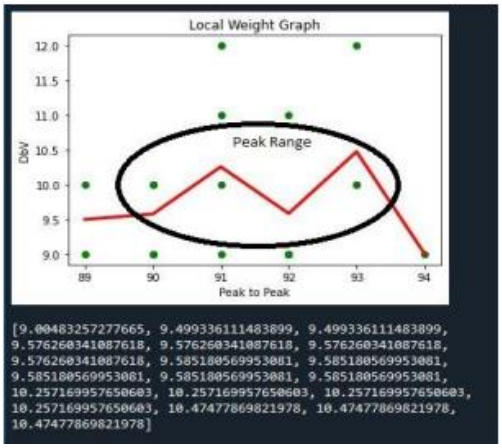

Figure 4.3: Local weighted Regression of data

4.3 Data flow diagrams

A data flow diagram (DFD) is graphic representation of the flow of data through an information system. A data flow diagram can also be used for the visualization of data-processing (structured design).It is common practice for a designer to draw a context-level DFD first which shows the interaction between the system and outside entities. DFDs show the flow of data from external entities into the system, how the data moves from one process to another, as well as its logical storage. There are only four symbols:

1. Squares representing external entities, which are sources and destinations of information entering and leaving the system
2. Rounded rectangles representing processes, in other methodologies, may be called Activities, Actions, Procedures, Subsystems etc. which take data as input, do processing to it, and output it.
3. Arrows representing the data flows, which can either, be electronic data or physical items. It is impossible for data to flow from data store to data store Except via a process, and external entities are not allowed to access data stores directly.
4. The flat three-sided rectangle is representing data stores should both receive information for storing and provide it for further processing.

4.3.1 DFD for Tilt Sensor

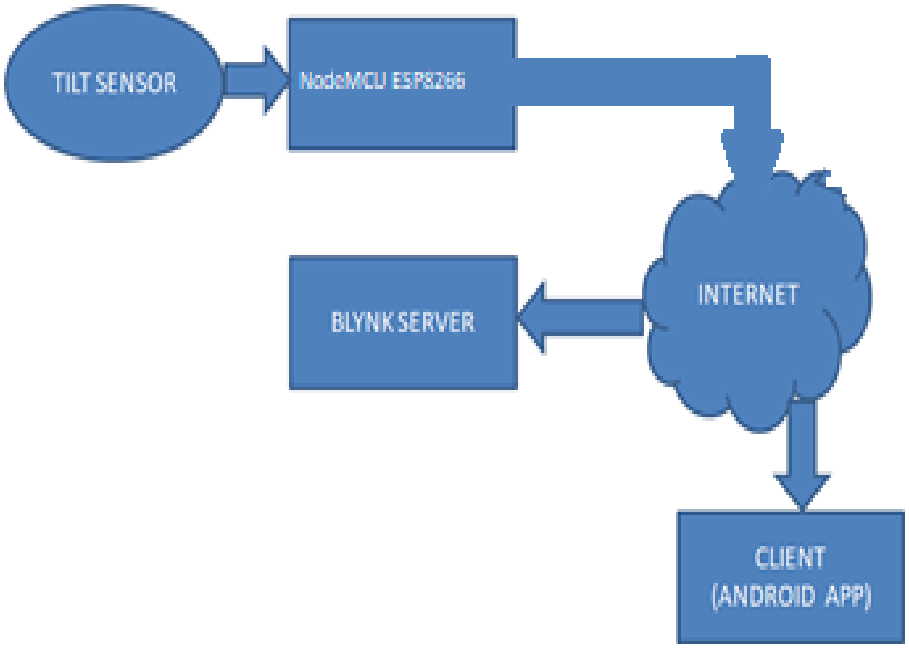

Figure 4.4: DFD for Tilt Sensor

4.3.2 DFD for Temperature Sensor

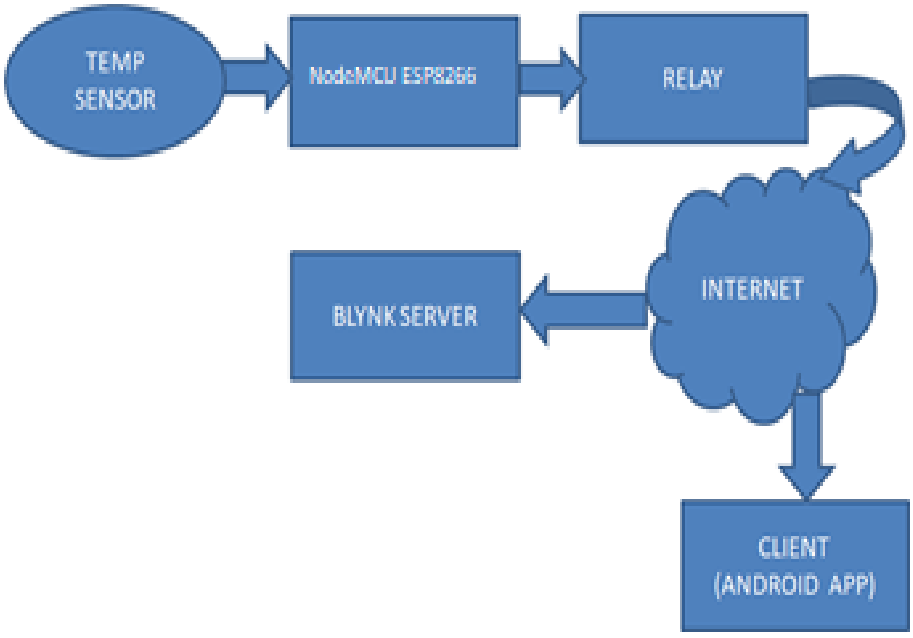

Figure 4.5: DFD for Temperature Sensor

4.4 Use-Case Diagram

A use case defines a goal-oriented set of interactions between external entities and the system under consideration. The external entities which interact with the system are its actors. A set of use cases describe the complete functionality of the system at a particular level of detail and it can be graphically denoted by the use case diagram.

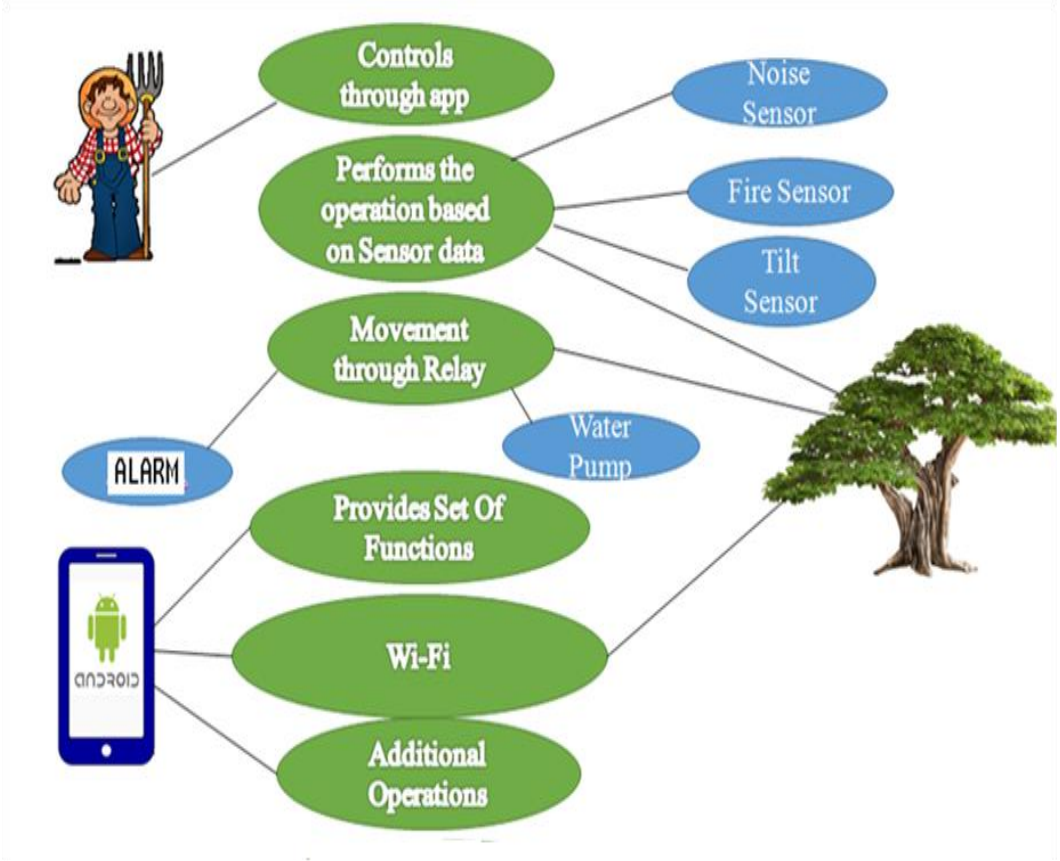

Figure 4.6: Use-Case Diagram Showing Client Transaction

User can controls the Sensor through The Blynk app and the Blynk App do the necessary operation through Relay based on various sensor data from the device deployed on the Tree, operations like switching On/off Pump and Buzzer. The Blynk app provides some set of functions.

4.5 Sequence Diagram

Sequence diagram are an easy and intuitive way of describing the behavior of a system by viewing the interaction between the system and the environment. A sequence diagram shows an interaction arranged in a time sequence. A sequence diagram has two dimensions: vertical dimension represents time, the horizontal dimension represents the object existence during the interaction. Basic elements:

- Vertical rectangle: Represent the objective active (method is being performed).
- Vertical dashed line: Represent the life of the object.
- X:representthelifeendofanobject.(Beingdestroyedfrommemory)
- Horizontal line with arrows: Messages from one object to another.

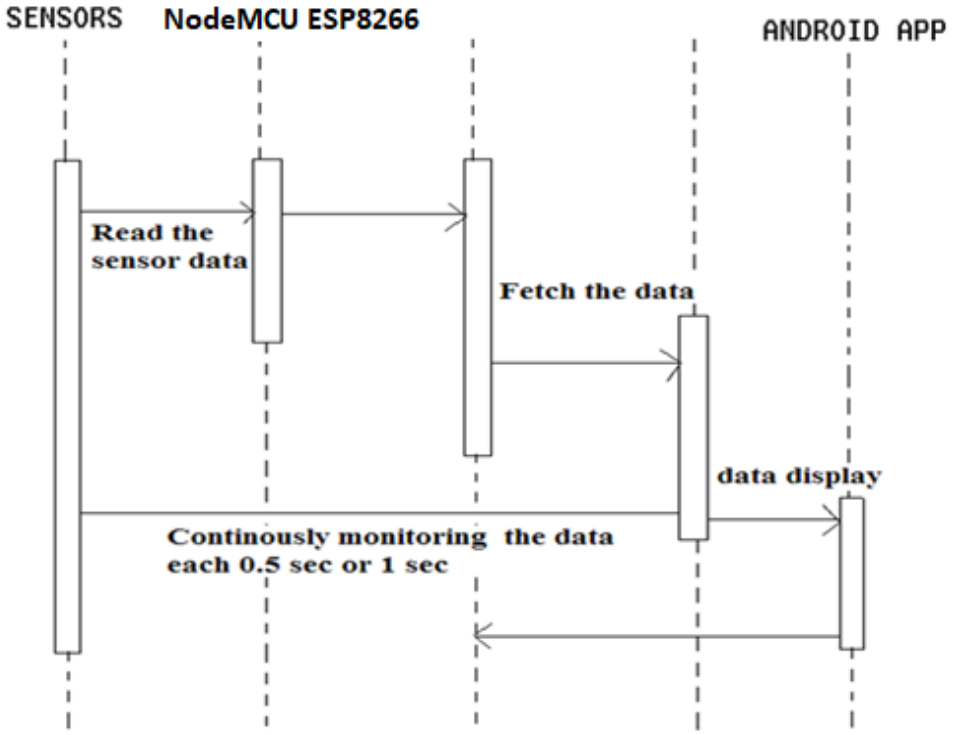

Figure 4.7: Sequence Diagram Showing Transition of Data from Sensors to Cloud

Chapter 5:

IMPLEMENTATION

Chapter 5

IMPLEMENTATION

5.1 Introduction

Implementation is a stage of the project where the theoretical design is turned intoa working system. This phase involves the actual materialization of the ideas, whichare expressed in the analysis document and developed in the design phase. Implementation should be perfect mapping of the design document in a suitable programming language in order to achieve the necessary final product. If the implementation is not carefully planned and controlled, it can cause chaos and confusion.

The implementation stage requires the following tasks:

- Careful Planning.
- Investigation of system and constraints.
- Design of methods to achieve the changeover.
- Evaluation of the changeover method.
- Correct decisions regarding selection of the platform.
- Appropriate selection of the language for application development.

5.2 Code

```
//Illegal logging and fire Detection and  monitoring system with the New Blynk app
//Include the library files
#define BLYNK_PRINT Serial
#include <ESP8266WiFi.h>
#include <BlynkSimpleEsp8266.h>
#include <DHT.h>
#define Buzzer 9
#define relay D3
#define flame D0
#define tilt D4
char auth[] = "nEFRloKBLKFEWe19s2556V46moAF-HxN";//Enter your Blynk Auth token
char ssid[] = "PAVAN";//Enter your WIFI name
char pass[] = "12345678";//Enter your WIFI password
```

```
int Analog_In = A0; // Analog output of the sensor
int Digital_Input= D2;

char status;
WiFiClient client;
float t,h,db;
const int sampleWindow = 50;
String apiKey = "K2YMB91MW2OZUBB0"; //ThingSpeak Write API Key
const char* server = "api.thingspeak.com";//Thingspeak API

unsigned int sample;
DHT dht(D1, DHT11);//(DHT sensor pin,sensor type)
BlynkTimer timer;


void setup() {
  Serial.begin(9600);
  pinMode (Analog_In, INPUT);
  pinMode (Digital_Input, INPUT);
  pinMode(Buzzer, OUTPUT);
  pinMode(flame, INPUT);
  pinMode(relay, OUTPUT);
  digitalWrite(relay, HIGH);
  Blynk.begin(auth, ssid, pass, "blynk.cloud", 80);
  dht.begin();

 //Call the functions
 timer.setInterval(100L,Sound);
 timer.setInterval(100L, DHT11sensor);
 timer.setInterval(100L, flamesensor);
 timer.setInterval(100L, Tiltsensor);
}
```

```
void Sound()
{
  float  Analog;
  int Digital;
  unsigned int sample;
  const int sampleWindow = 50;
  unsigned long startMillis= millis();                  // Start of sample window
  float peakToPeak = 0;                                 // peak-to-peak level

  unsigned int signalMax = 0;                           //minimum value
  unsigned int signalMin = 1024;                        //maximum value

                                                        // collect data for 50 mS
   while (millis() - startMillis < sampleWindow)
   {
     sample = analogRead(0);                            //get reading from microphone
     if (sample < 1024)                                 // toss out spurious readings
     {
       if (sample > signalMax)
       {
         signalMax = sample;                            // save just the max levels
       }
       else if (sample < signalMin)
       {
         signalMin = sample;                            // save just the min levels
       }
     }
   }
   peakToPeak = signalMax - signalMin;                  // max - min = peak-peak amplitude
   db = map(peakToPeak,20,900,49.5,90);                 //calibrate for deciBels
   Blynk.virtualWrite(V2, db);                                          //set text size to 2


  //Current values are read out, converted to the voltage value...
```

```
Analog =  analogRead (Analog_In)   *  (5.0 / 1023.0);
Digital = digitalRead (Digital_Input) ;

//...  and issued at this point
//Serial.print  ("Analog voltage value:");  Serial.print (Analog,  4) ;   Serial.print  ("V, ");
//Serial.print ("Limit value:") ;
int tcount = 0;
int count = 0;
while(tcount<10)
{

  if  (count>=2)
  {
     Serial.println("-----------------------------------------------------------------");
     Serial.print  ("Analog voltage value:");  Serial.print (Analog,  4) ;   Serial.print  ("V,
");
     Serial.print ("Limit value:") ;
     Serial.println (" reached ");

     Serial.print(db);
     Serial.println(" dB");
     Serial.println(signalMax);
     Serial.println(signalMin);
     Serial.println(peakToPeak);
     Blynk.logEvent("sound", "Warning! Chainsaw Sound detected");
     Blynk.email("bgm00ind@gmail.com", "Code Malli GREEN", "ChainSaw Sound
Detected Click here https://thingspeak.com/channels/1769216");
     digitalWrite(Buzzer, HIGH);
     tone(Buzzer, 1000, 200);
  }

  if  (Analog>=1.9500)
  {
     count++;
```

```
    }
    tcount++;
    delay(400);
  }
}
//Get the DHT11  Temperature and Humidity sensor values
void DHT11sensor() {
  h = dht.readHumidity();
  t = dht.readTemperature();

  if (isnan(h) || isnan(t)) {
    Serial.println("Failed to read from DHT sensor!");
    return;
  }
  Blynk.virtualWrite(V0, t);
  Blynk.virtualWrite(V1, h);
}

//Fire Sensor
void flamesensor() {
  bool value = digitalRead(flame );
    if (value == 0)
    {
    Blynk.email("bgm00ind@gmail.com", "Code Malli RED", "Fire Detected Click here
https://thingspeak.com/channels/1769216");
    Blynk.logEvent("fire", "Warning! Fire was detected");
    digitalWrite(relay, LOW);
    digitalWrite(Buzzer, HIGH);
    tone(Buzzer, 1000, 200);
    }
    else
    {
      noTone(Buzzer);
    }
```

```
}
void Tiltsensor() {
  bool value = digitalRead(tilt);
    if (value == 0)
    {
    Blynk.logEvent("fall", "Warning! Tree has Fallen");
    Blynk.email("pavan2020v@gmail.com", "Code Malli FALL", "Tree has Fallen Click here https://thingspeak.com/channels/1769216");
    digitalWrite(Buzzer, HIGH);
    tone(Buzzer, 1000, 200);
    }
    else
    {
      noTone(Buzzer);
    }
}

//Get buttons values
BLYNK_WRITE(V3) {
 bool RelayOne = param.asInt();
  if (RelayOne == 1) {
    digitalWrite(relay, LOW);
  } else {
    digitalWrite(relay, HIGH);
  }
}

void loop() {
  //.............
  Blynk.run();//Run the Blynk library
  timer.run();//Run the Blynk timer
   if (client.connect(server, 80))
   {
    String postStr = apiKey;
```

```
    postStr += "&field1=";
    postStr += String(t);
    postStr += "&field2=";
    postStr += String(h);
    postStr += "&field3=";
    postStr += String(db);
    postStr += "\r\n\r\n\r\n\r\n\r\n\r\n";
    client.print("POST /update HTTP/1.1\n");
    client.print("Host: api.thingspeak.com\n");
    client.print("Connection: close\n");
    client.print("X-THINGSPEAKAPIKEY: " + apiKey + "\n");
    client.print("Content-Type: application/x-www-form-urlencoded\n");
    client.print("Content-Length: ");
    client.print(postStr.length());
    client.print("\n\n\n\n\n");
    client.print(postStr);
    Serial.print("Temperature: ");
    Serial.println(t);
    Serial.print("Humidity: ");
    Serial.println(h);
  }
  client.stop();
  delay(1000);
}
```

5.3 Project Implementation

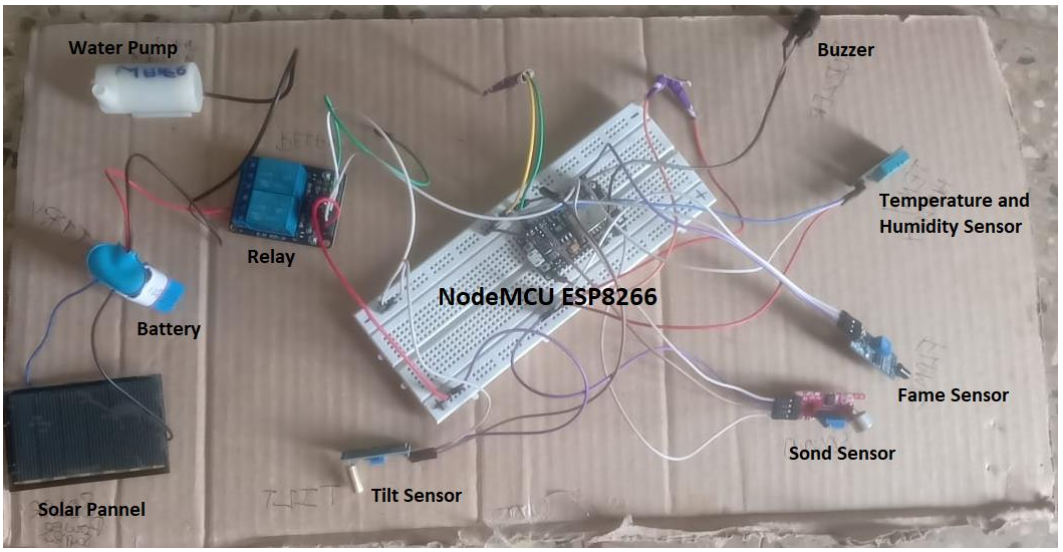

Figure 5.1: Prototype for illegal logging detection System

5.4 Blynk Configuration

Blynk is a toolset for all makers, inventors, designers, teachers, nerds and geeks who would love to use their smartphones to control electronics like Arduino, RaspberryPi and similar ones. Weve done all the hard work of establishing internet connection, building an app and writing hardware code.

With Blynk, you simply snap together an amazing interface from various widgets we provide, upload the example code to your hardware and enjoy seeing first results in under 5 minutes! It works perfectly for newbie makers and saves tons of time for evil geniuses.

Blynk will work with all popular boards and shields. We wanted to give you full freedom when deciding how to plug Blynk into your existing or new project. You will also enjoy the convenience of Blynk Cloud. Which is, by the way is free and open-source.

Blynk is not an app that works only with a particular shield. Instead, it's been designed to support the boards and shields you are already using. And it works on iOs and Android. Blynk also works over USB. This means you can tinker with the app by connecting it to your laptop or desktop while waiting for some internet shield to arrive.

Blynk works over the Internet. So the one and only requirement is that your hardware can talk to the Internet.

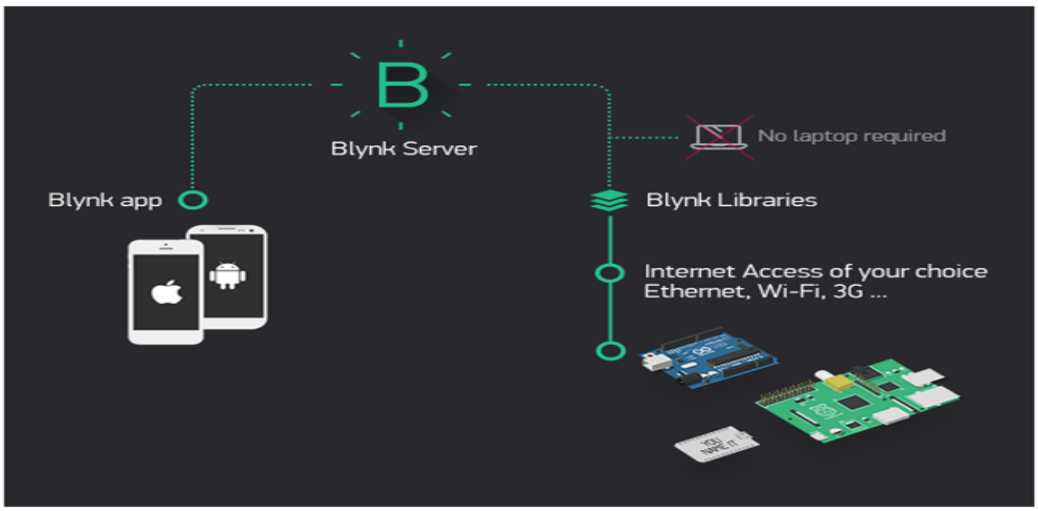

Figure 5.2: Blynk Architecture

Currently, Blynk libraries work with:

- USB
- Ethernet shield
- Wi-Fi shield
- Arduino with Ethernet
- Arduino YN (testing in progress)
- ESP8266
- Raspberry Pi (Blynk will communicate with Pi's GPIOs)
- More Arduino compatible shields and boards

It's not that easy to take Arduino out of your home network, so we've built a Blynk server. It handles all the authentication and communication, and also keeps an eye on your board while the smartphone is offline. Blynk server runs on Java and is open-source. You will be able to run it locally if you really need to. Messaging between mobile apps, Blynk Server and Arduino is based on a simple, lightweight and fast binary protocol over TCP/IP sockets. After downloading the app, we need to create an account and log in. Welcome to Blynk!

Figure 5.3: Blynk first screen

Well also need to install the Blynk Arduino Library, which helps generate the firmware running on your ESP8266. Download the latest release from Blynks GitHub repo, and follow along with the directions there to install the required libraries.

Create a Blynk Project

Next, click the Create New Project in the app to create a new Blynk app. Give it any name you please, just make sure the Hardware Model is set to ESP8266.

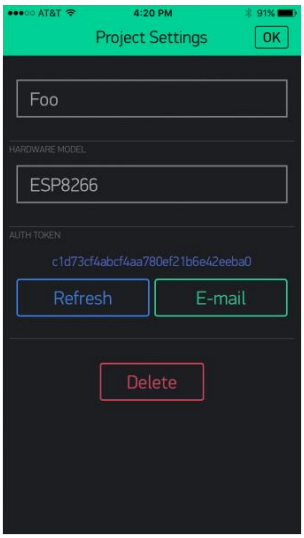

Figure 5.4: Creating a Project

The Authentication Token is very important you will need to stick it into your ESP8266sfirmware. For now, copy it down or use the E-mail button to send it to yourself.

Add Widgets to the Project

Then youll be presented with a blank new project. To open the widget box, clickin the project window to open.

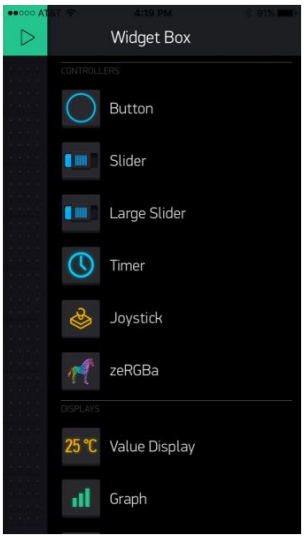

Figure 5.5: Adding widgets

Add a Button, then click on it to change its settings. Buttons can toggle outputs on the ESP8266. Set the buttons output to gp5, which is tied to an LED on the Thing Dev Board. You may also want to change the action to Switch.

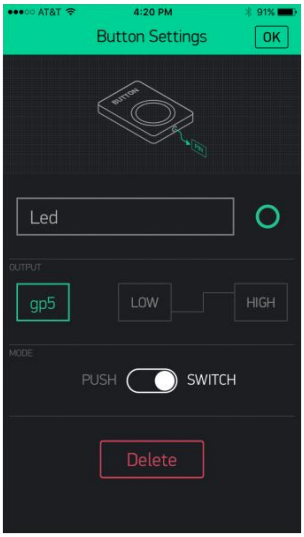

Figure 5.6: Adding a button

Upload the Blynk Firmware

Now that your Blynk project is set up, open Arduino and navigate to the ESP8266 Standalone example in the File Examples Blynk Boards and Shields menu.

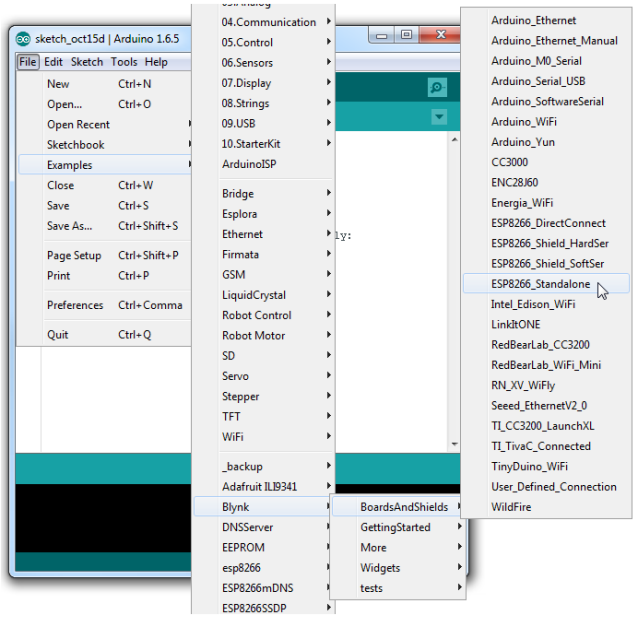

Figure 5.7: Uploading into ESP8266 firmware

Before uploading, make sure to paste your authorization token into the auth[] vari-able. Also make sure to load your WiFi network settings into the Blynk. Begin ("auth","ssid","pass") function.

Run the Project

After the app has uploaded, open the serial monitor, setting the baud rate to 9600.

Wait for the Ready (ping: xms). message.

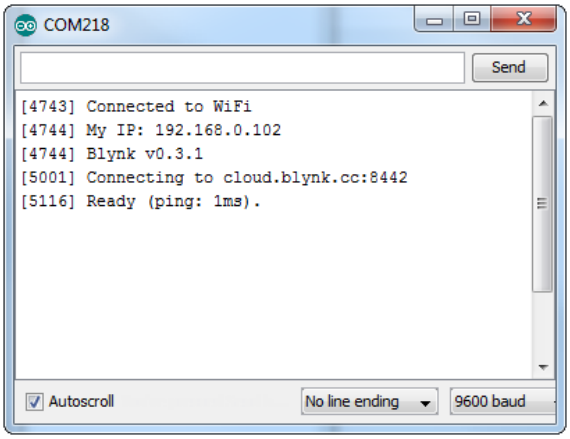

Figure 5.8: Running the Firmware

Then click the Run button in the top right corner of the Blynk app. Press the button and watch the LED!

The following snaps are for the all the widgets and the sensors and for our project:

Blink App Overview

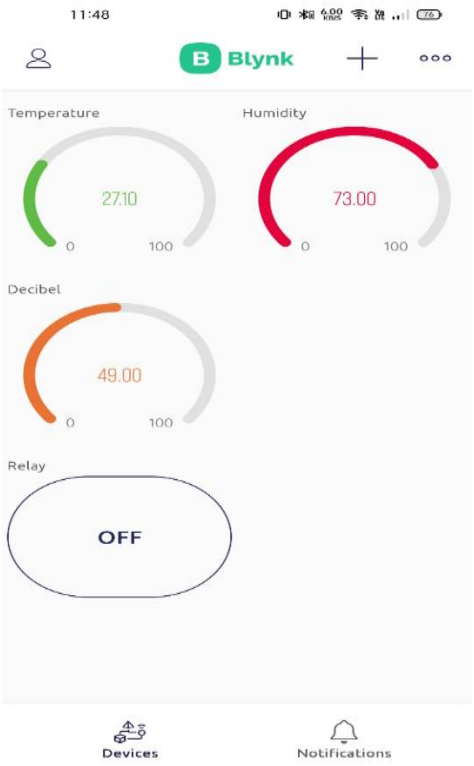

Figure 5.9: Project Overview

Tilt Widget settings

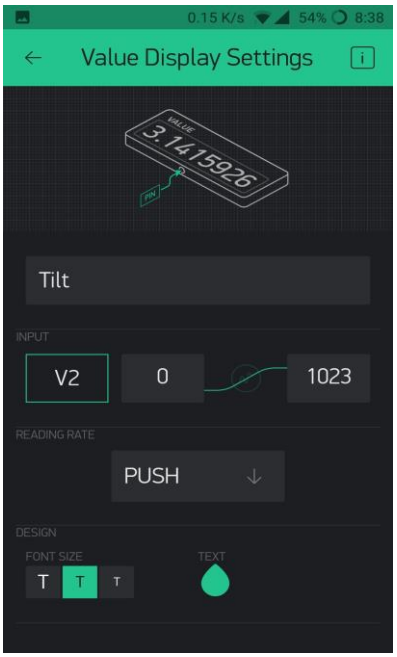

Figure 5.10: Widget settings for Tilt Sensor

Temperature Widget Settings

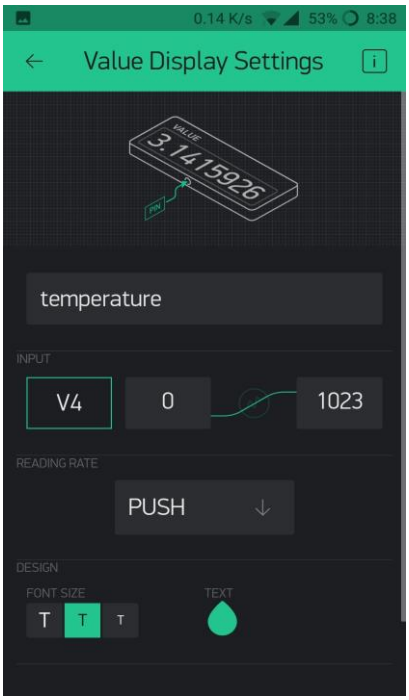

Figure 5.11: Widget Settings for Temperature Sensor

Chapter 6:
TESTING

Chapter 6

TESTING

During this implementation of the system each module of the system was tested separately to uncover errors within its boundaries. User interface is used as a guide in the process. In computer programming, unit testing is a method by which individual units of source code are tested to determine if they are fit for use. A unit is the smallest testable part of an application. In procedural programming a unit may be an individual function or procedure. In object-oriented programming a unit is usually a method. Unit tests are created by programmers or occasionally by white box testers during the development process.

Ideally, each test case is independent from the others: substitutes like method stubs, mock objects, fakes and test harnesses can be used to assist testing a module in isolation. Unit tests are typically written and run by software developers to ensure that code meets its design and behaves as intended.

6.1 TEST CASES

The Smart Module is thoroughly tested in the developer's environment before its deployment in the working environment. The testing methodology used to test the system is Use-Case Testing. Use case testing is a technique that can be used to identify test cases that cover the entire system, from start to finish. It helps to describe how a particular system is of use to the end user. It is used widely in developing tests to check systems for acceptable levels.

Use-Cases identified for testing and results:

Test-Case 1: Flame Sensor

Function: To detect fire in nearby environment.

Pre-conditions: Fire should have occurred.

ExpectedOutcome: An alert notification and E-mail must be sent via Blynk application and the relay must turn ON the water pump.

Inputs: Readings from the flame sensor.

ActualOutcome: An alert notification and E-mail is sent via Blynk application and the relay turns ON the water pump.

Test-Case 2: Tilt Sensor

Function: To detect unusual vibrations and Tilt Angle threshold.

Pre-conditions: Unusual vibrations or the tree should have fallen.

ExpectedOutcome: An alert notification must be sent to the Blynk application.

Inputs: Readings from the tilt sensor.

ActualOutcome: Alert notification is sent to the Blynk application.

Test-Case 3: Sound Sensor

Function: To detect sound from chainsaw.

Inputs: Environment sound (all sounds excluding chainsaw Sound).

ExpectedOutcome: No change

ActualOutcome: No change

Test-Case 4: Sound Sensor

Function: To detect sound from chainsaw.

ExpectedOutcome: An alert notification and E-mail must be sent via Blynk application.

Inputs: Chainsaw Sound.

ActualOutcome: An alert notification and E-mail is sent via Blynk application.

Sound sensor data was tested with various different sounds and the results were analyzed, the system was found to have ±80% accuracy.

Postconditions: Fall has been detected.

Chapter 7:
EXPERIMENTAL RESULTS

Chapter 7

EXPERIMENTAL RESULTS

BlynkApp Notification

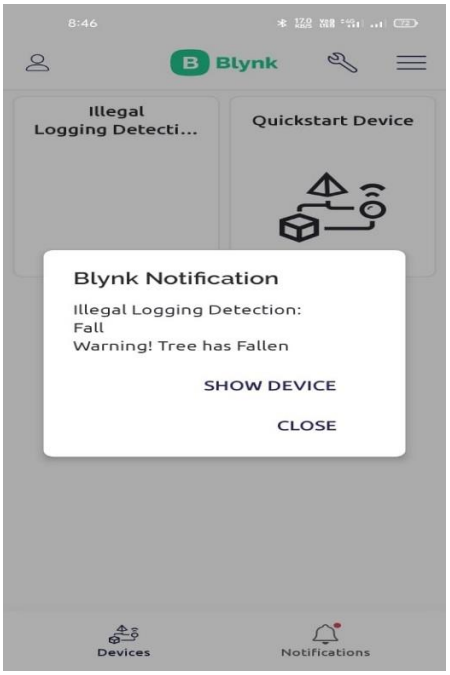

Figure 7.1: Tree fallen notification on BlynkApp

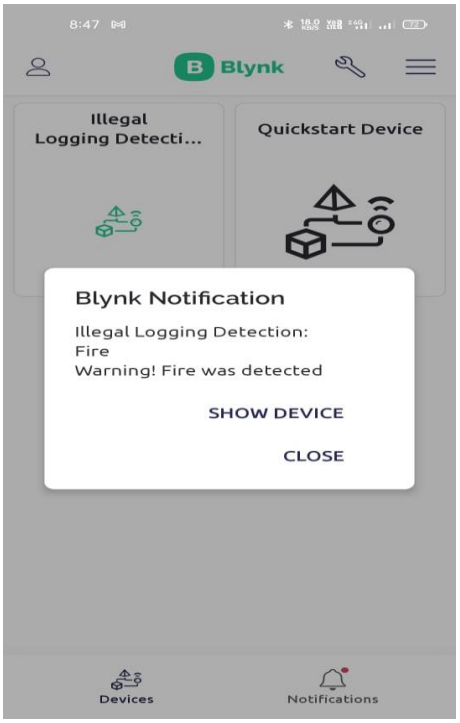

Figure 7.2: Fire detection notification on BlynkApp

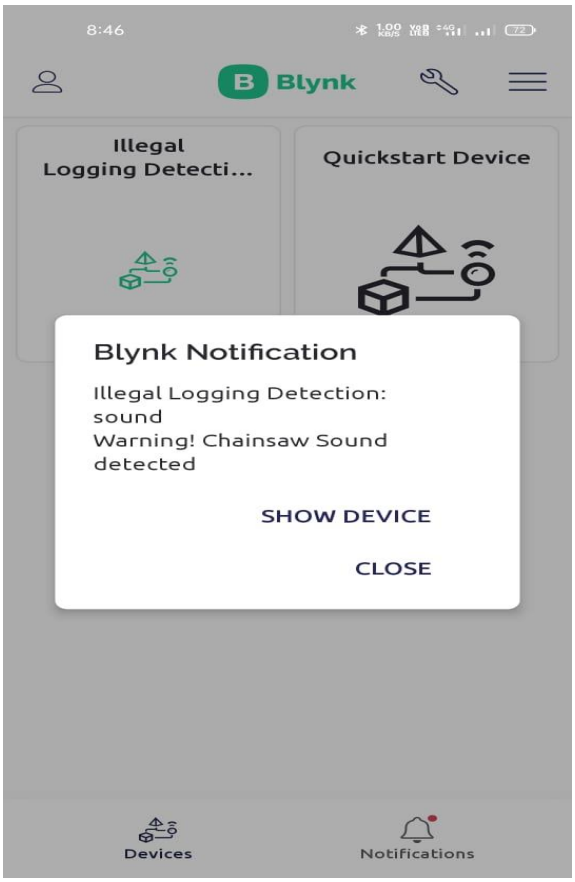

Figure 7.3: Chainsaw sound detected notification on BlynkApp

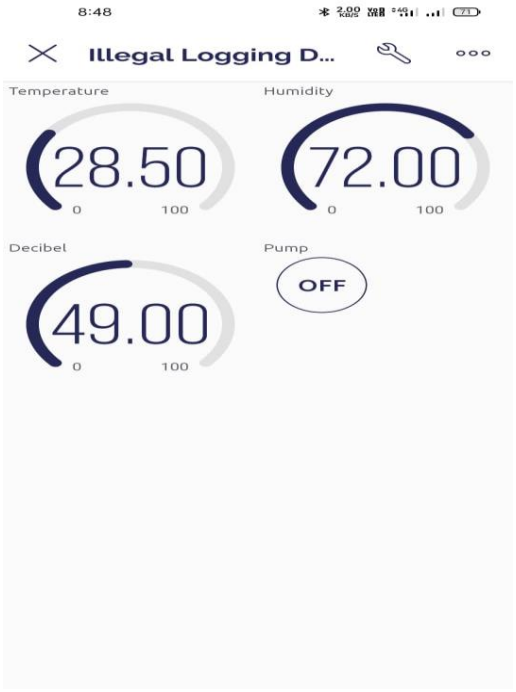

Figure 7.4: BlynkApp dashboard

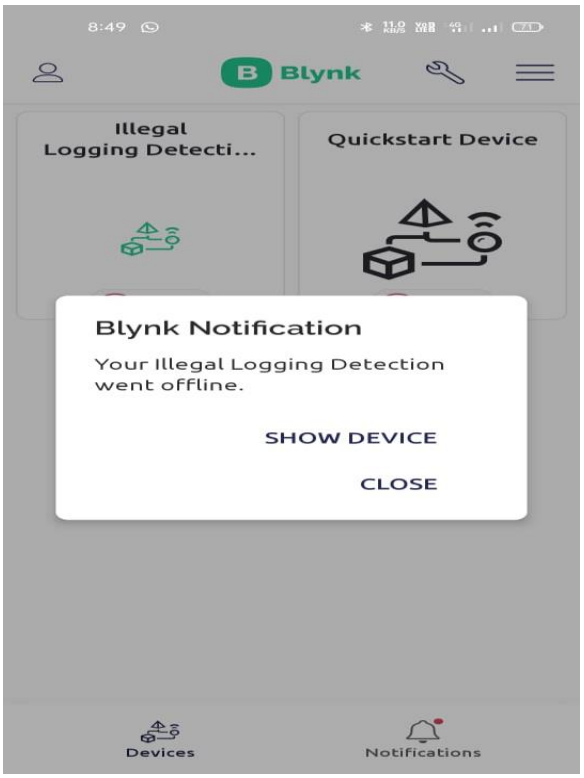

Figure 7.5: Device offline notification

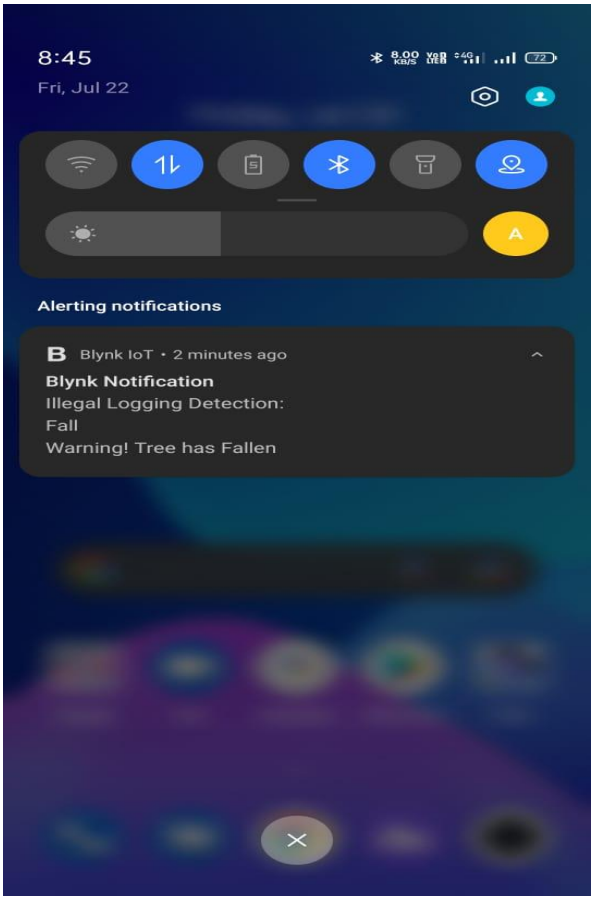

Figure 7.6: BlynkApp notification on Mobile

Email from BlynkApp

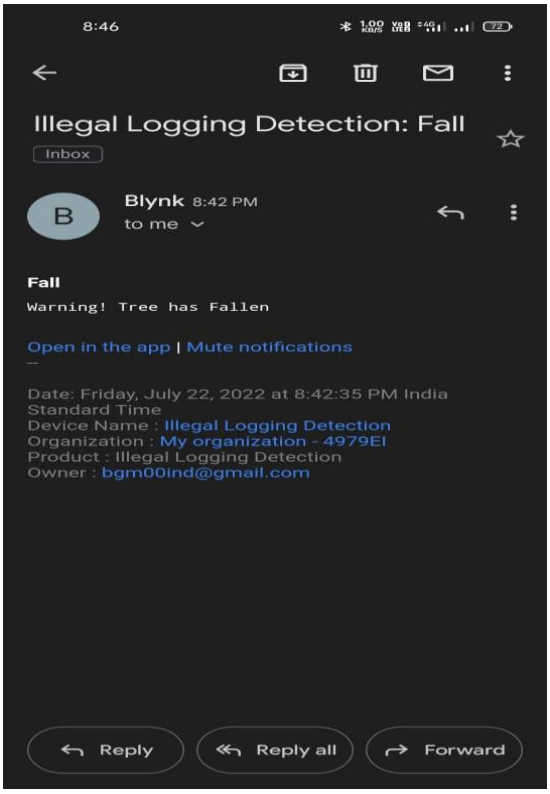

Figure 7.7: Tree fallen email sent by BlynkApp

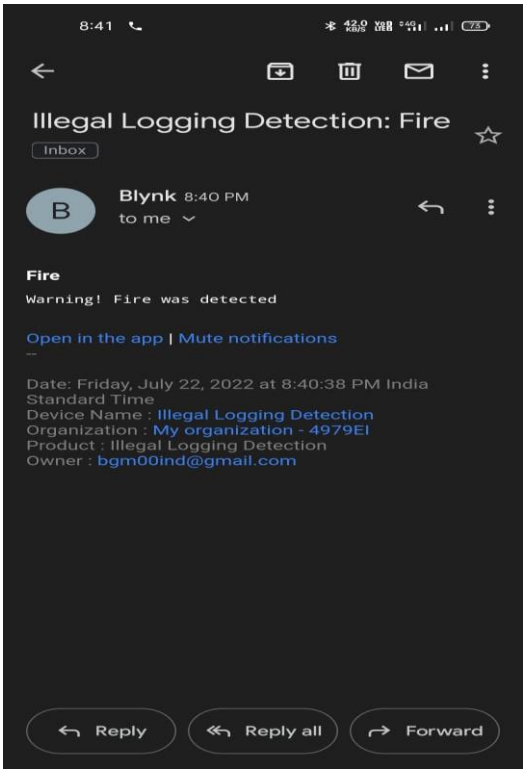

Figure 7.8: Fire detected mail sent by BlynkApp

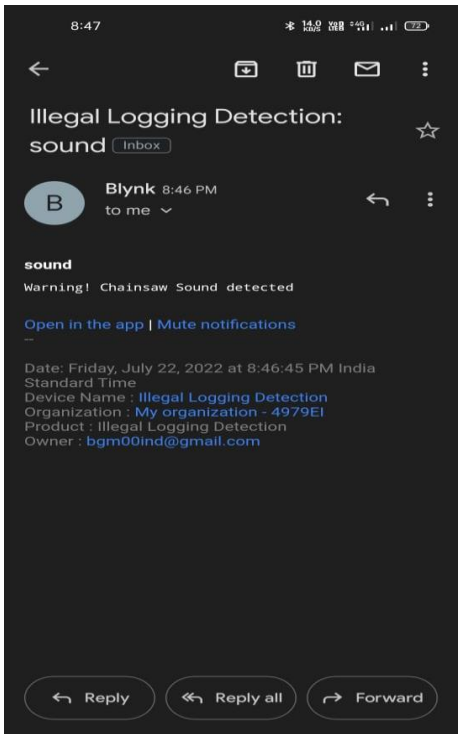

Figure 7.9: Chainsaw sound detected email sent by BlynkApp

ThingSpeak Data

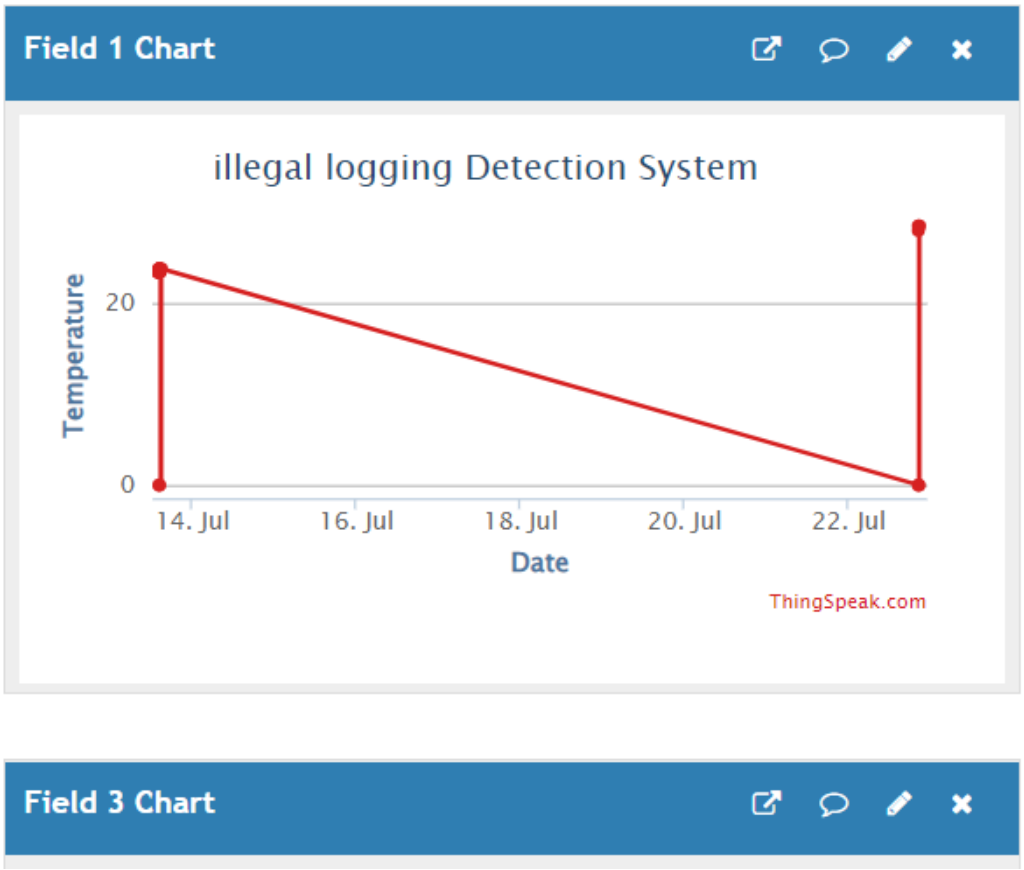

Figure 7.10: Temperature sensor data

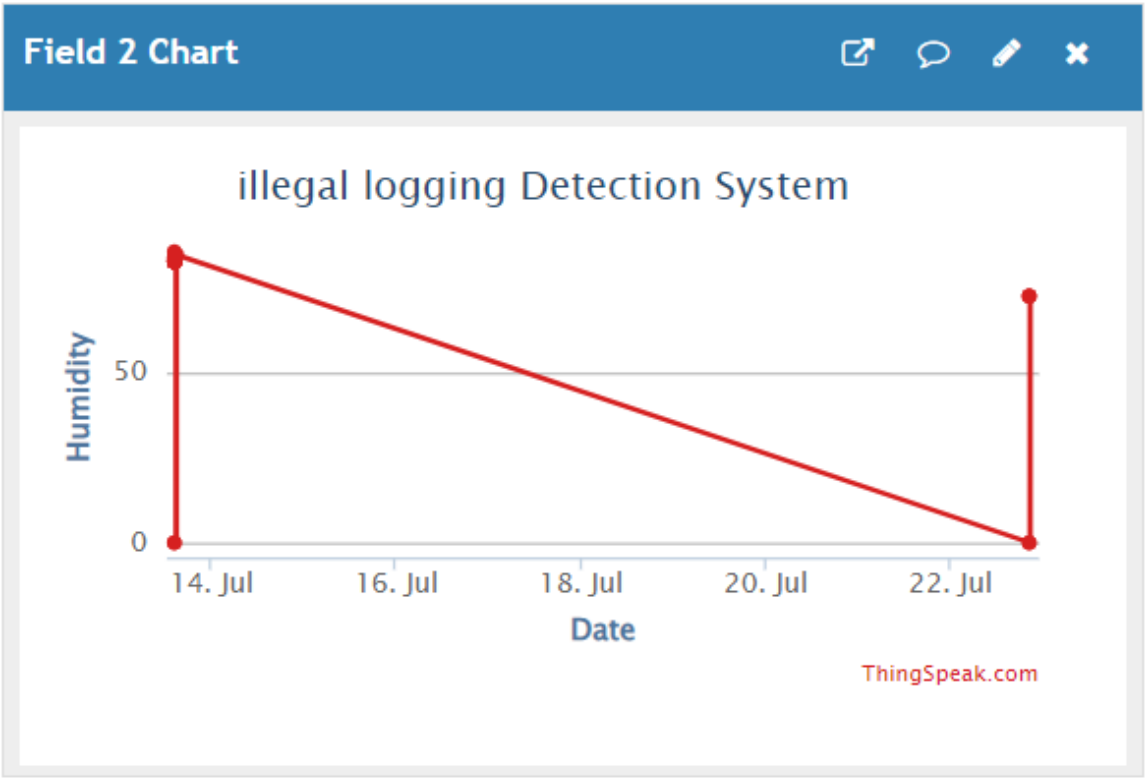

Figure 7.11: Humidity sensor data

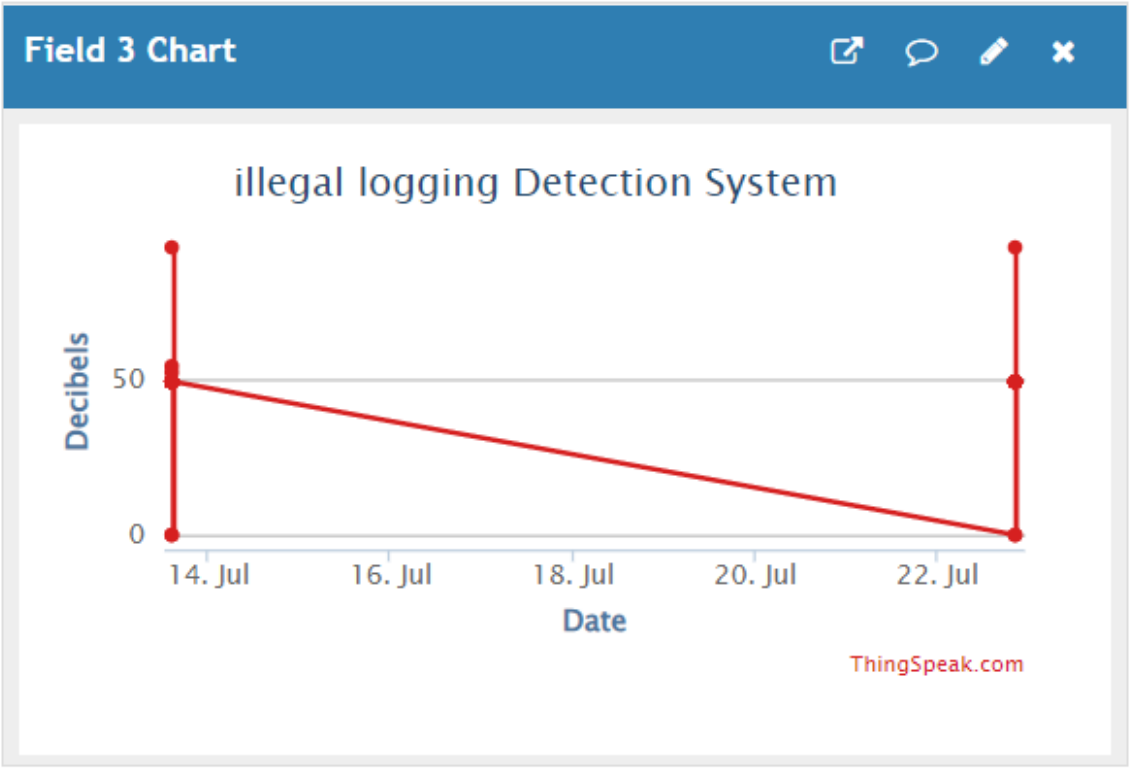

Figure 7.12: Sound data

Chapter 8:

CONCLUSION

Chapter 8

CONCLUSION

Combining sound and tilt/vibration sensors creates a powerful combination that can effectively address incidents of unlawful logging. Here, sound sensor is used to identify that logging has occurred within the bounds of the sound sensor's ability to recognise sounds. A tilt/vibration sensor can be installed to provide two processes verification (sound and vibration) to confirm the presence of illegal logging. In addition to this the flame sensor is used to detect forest fires and activate a water pump in case of fire. In order for the forest patrol officers to respond quickly in the event of logging, the system also sends an alert notification and E-mail through the Blynk IoT app and the data from the sensors is stored and can be visualized on an online cloud platform called ThingSpeak.

REFERENCES

[1] Snehal. A. Jadhav, Shweta. M. Ingawale, Dhanashree G. Mohite "Wireless Sensor Network Based Monitoring System for Forest" in International Journal on Recent and Innovation Trends in Computing and Communication, Volume: 3, Issue: 3, ISSN: 2321-8169, pg no 1733 –1738, 2015.

[2] Duraivel, Beniel Wellington, Arul Nayagam and Kijral, "An IOT based fire alarming and authentication system for workhouse using raspberry pi3", International Journal of Emerging Technology in Computer Science &Electronics (IJETCSE) 2018.

[3] A. Yusoff, S. Shafiril, G. Wills, L. Gilbert, and R. Crowder, "Utilizing the application of sensors to develop low-cost remote sensing system for monitoring forest environmental activity," 2016 IEEE Conf. Wirel. Sensors, ICWiSE 2016, vol. 2017–December, pp. 1–6, 2017.

[4] P. G. Kalhara, V. D. Jayasinghearachchd, A. H. A. T. Dias, V. C. Ratnayake, C. Jayawardena, and N. Kuruwitaarachchi, "TreeSpirit: Illegal logging detection and alerting system using audio identification over an IoT network," 2017 11th Int. Conf. Software, Knowledge, Inf. Manag. Appl., pp. 1–7, 2017.

[5] Y.-Y. Chen and J.-J. Liaw, "A Novel Real-time Monitoring System for Illegal Logging Events Based on Vibration and Audio," 2017 Ieee 8Th Int. Conf. Aware. Sci. Technol., no. iCAST, pp. 470–474, 2017.

[6] 'Ghousia Sultana B' "IOT Based Anti-Poaching Alarm System for Trees in Forest using Wireless Sensor Network" Volume 9, Special Issue No. 3, May 2018.

[7] 'Narhari R. Kotkar M.E (ESD AND VLSI)' "Anti-Smuggling System for Trees in Forest using Flex Sensor and Zigbee" International Journal of Advanced Research in Computer Engineering & Technology (IJARCET) Volume 3, Issue 9, September 2014.