

INTELLIGENT ENERGY MANAGEMENT AND DISASTER ALERT SYSTEM USING INTERNET-OF-THINGS APPROACH

by

Onyeyili T.C.¹, Nsodukwa A.O.², Dimson I.C.³, Obianyo O.R.⁴

1,3: Dept. of Electronic & Computer Engineering, Nnamdi Azikiwe University, Awka, Nigeria.

2: Dept. of Computer Engineering, Caritas University, Enugu, Nigeria.

4: Dept. of Computer Engineering, Madonna University of Nigeria, Akpugo Campus, Enugu State, Nigeria.

ABSTRACT

There is a need for efficient and sustainable ways of energy utilization and management even as worldwide industrialization continues. Occurrences of natural disasters signify a need for disaster prevention systems that can be used to manage and control the damages caused by the natural disasters. This research proposes a home energy management and disaster alert system which provides remote control and monitoring of home appliances and giving an early warning about impending disasters. The system comprises of two parts: the prototype device part and the web application part. The prototype part consists of a microcontroller, water level sensor, a relay, and other minor electronic components which provide the flood monitoring and control of the output appliance; while the web application part provides an interactive web interface to access the system. The resulting system provides remote monitoring and control of a light bulb and created a web alert every time the water level in its environment reaches a potential flooding level. Similar energy management and disaster alert systems place emphasis on handling larger areas and handling much more data; however, the proposed system serves only for a home.

KEYWORDS/PHRASE: Energy, flood, microcontroller, prototype, disaster, Internet of Things (IoT).

I. INTRODUCTION

The aim of this research is to analyze and develop a prototype of a home energy management and disaster prevention system, and to explore a way through which Internet of Things (IoT) technology

can be applied in energy management and flood disaster prevention.

It is known that Internet is a worldwide computer network, made up of other networks which make use of various protocols to exchange information between each other. Today the Internet has revolutionized the way we relate with devices around us and our environment; we can now connect with any device using the IoT (Kitagami S. et al., 2016).

By definition, Internet of Things (IoT) is a global base that enables advanced services by connecting physical and virtual things through the Internet to share information. IoT provides any physical thing with the ability to transmit data over a network without needing human interaction via APIs over the Internet. Any electrical device can be connected to the Internet to share information, IoT technology can be applied in multiple ways such as remote monitoring of a target or controlling a device from one location to another (Lu H. et al., 2019).

Energy management can be seen as employing various intentional procedures and actions to manage, control and reduce energy consumption by consuming needed energy and conserving the rest. In the context of electrical engineering, energy management can be seen as employing intentional procedures to manage the consumption of electrical energy, from switching off a light bulb to switching off large printing equipment; electrical energy management is concerned with only consuming electrical energy that is needed. Making improvements in energy usage efficiency is widely considered as the most promising, fastest, cheapest

and safest means to mitigate climate change (Pawar P. and Vittal K., 2019).

Disaster can be seen as an unfavorable, undesirable event that causes physical harm or damage to lives and property. There are many forms of disasters like the collapse of buildings, fire incidents, natural disasters like floods and

1.1 IoT and IoT systems

A major goal in IoT is the making of our environment smarter; by providing it with the information it requires, through real-time and historical data feeds and applying computational processing on the information to enable for smart automatic decisions. IoT systems are adopted in various areas across the industrial sector, health sector and social sector. In the home sector, IoT application in implementing smart homes capable of managing themselves has been explored. In the social sector, effective application of IoT for disaster prevention is gaining grounds. Through the use of sensors and incorporating microcontrollers, different household appliances and their operations can be controlled; and things like ambient temperature, house visitors and pests can be monitored. The states of these devices can also be monitored using microcontrollers and information coming from the devices can be relayed to the Internet by connecting a network interface device to the microcontroller. The setup of an electrical device, a microcontroller and a network interface device can be said to be an IoT system (Saleh M. et al., 2017).

An IoT system has the ability to connect to a network to send and receive information automatically. The efficient use and management of energy is one of the most important goals of any industrial sector; this stems mostly from the fact that energy wastage can lead to incurring unnecessary costs and even losses for the industry. Smart IoT systems can be used to manage electrical power usage in a household by automatically or remotely turning on and off appliances and providing real-time information of power usage in

earthquakes. According to the United Nations International Strategy for Disaster Reduction, between the years 1998 and 2017, climate disasters made up 91% of all the 7255 recorded disasters around the world. Among them, floods showed the most frequency at 43%. To mitigate disasters like floods, for example, it is helpful to a great extent to introduce a flood disaster prevention mechanism that helps to warn residents of the flood-prone area. the home.

IoT can be applied to disaster prevention to help prevent disasters like flood, earthquake, etc. To prevent a fire disaster, for example, it would be useful to alert the relevant authorities and reduce the chances for the fire to escalate; this can be done automatically and unsupervised using IoT systems. IoT systems can be introduced in the area of detecting possible disaster scenarios beforehand and sending out notifications to help abate further damage because they can communicate with the Internet in real-time and requires no human supervision in operation.

II. LITERATURE SURVEY

2.1 Energy Management

Energy management has emerged as a relevant problem because of increased demand for energy due to increasing global population and the emergence of the issues of global warming and climate change. Energy gotten from renewable sources helps to mitigate issues like climate change, but it is also in limited supply and it is difficult to store in a large scale. Most energy management implementations employ measures to control and reduce energy consumption by consuming only needed energy and conserving the rest. In the context of electrical engineering, energy management can be seen as employing intentional procedures to manage the consumption of electrical energy, like switching off an unused light bulb to switching off large printing equipment. Electrical energy management is concerned with only consuming energy that is needed (Anastasi G. et al., 2011).

2.2 Energy Management Systems

An energy management system is a system comprised of various components used to monitor, optimize and control the usage, generation or transmission of energy. Energy Management Systems can be categorized into two types: predictive energy management system and non-predictive energy management system.

- **Predictive Energy Management System:** This type of system typically makes use of optimization-based approaches such as dynamic programming, model predictive control, and equivalent consumption minimization strategy.
- **Non-predictive Energy Management System:** This system usually applies the rule-based approach; although in some cases, the filter-based approach and the fuzzy logic-based approach have also been studied in literature.

2.3 Some Techniques Used in Energy Management

- **Energy monitoring:** The major reasons for energy consumption in buildings are electric appliances used for space heating, water heating, air conditioning, lighting, and the use of computers and other electronic devices. A significant cause of energy wastage is due to the careless use of such appliances and devices. By eliminating factors of energy wastage without lowering the level of perceived comfort, the overall energy consumption in buildings could reduce by approximately 30%. Another main source of energy waste can be seen when electrical appliances are in standby mode which accounts for close to 10% of energy consumption in buildings. Based on a number of studies, giving relevant information to consumers about the statistics of their energy usage can cause a significantly reduced rate of wasteful energy consumption by about 5-20%.
- **Smart grids:** A popular and more advanced method used in energy management systems is the use of smart grids. Energy management systems implemented with the use of smart grids that can measure and control energy consumption and integrate with communication systems allow for higher energy efficiency. Smart grids, together with

the use of real-time information systems or predetermined rules, avoid wasting resources by applying energy distribution optimization techniques. Collecting and analyzing consumer energy usage data in real-time can facilitate an energy management system to be effective in controlling and managing the energy consumption of buildings in order to use the best and most efficient path in consuming energy. Using previously collected data in addition to specific algorithms, the system can predict future energy consumption patterns of various buildings and make a realistic estimation of the energy consumption based on processing and analyzing the energy consumption data.

- **Load Scheduling:** Another major goal in energy management is cost optimization, that is, not wanting to bear the cost of using energy that is not really needed. This can be achieved by load schedule. A load scheduling algorithm can be implemented by an energy management system to supply power to appliances only during times the appliance is most likely to be used. A load scheduling algorithm is created with a major purpose of reducing expenses due to energy consumption. This strategy can be effective for managing energy consumption in a situation where the duration of usage of appliances can be predicted. For example, in a home but this proves ineffective in situations where one can not schedule the times one would use an appliance, for example, in a hospital.

2.4 IoT-Based Energy Management System

The emerging technology of the Internet of Things (IoT) can be used to manage energy consumption in commercial, residential, and industrial settings in a better and more efficient way. One of the applications of IoT in residential areas is controlling various electrical appliances or "home automation". IoT technology-based energy management systems are usually composed of the following:

- **A component used to monitor power consumption:** This can be a single smart sensor used to monitor energy consumption or electrical power usage of a single device or consists of a network

of sensors deployed to monitor various appliances and conditions in a building. Smart sensors can, for example, detect when the temperature of a room is up to a suitable level and then trigger signals that cause the heating appliance to be turned off, or monitor how long a device has been on.

- **A component used to control appliances:** This is usually a microcontroller chip embedded into a system connected to relays, or sensors that connect the microcontroller to home appliances and allows it to control energy consumption based on the signals it receives from other parts of the IoT system (e.g. web servers, sensors, appliances, etc).

- **A component used to process and store data:** This is usually a computer server that collects data and performs operations such as projecting real-time energy usage costs or proposing real-time energy optimization using algorithms. This computer server can either be a physical server or one in the cloud. Cloud computing services allow for abstraction of the needed server architecture for web programming. This greatly reduces the cost of managing a physical server which in turn brings down the cost of the entire system.

- **A component used to display power consumption information:** This can be a hosted web application or mobile application that is used to query the computer server for energy consumption information. It is also used to send commands through the web server to the energy management system to switch on or switch off an appliance. Figure 1 shows the diagram of IoT-Based Energy Management System.

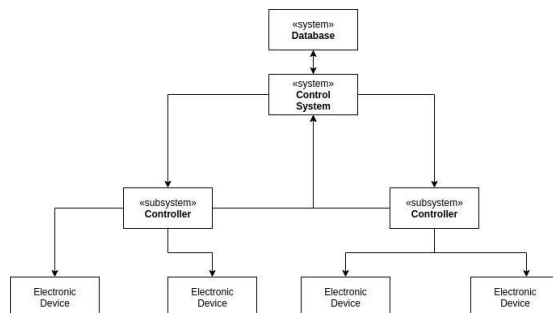


Figure 1: Block Diagram of IoT-Based Energy Management System

2.5 IoT-Based Alert System

The recent growing popularity of IoT systems has brought about its application in various fields of study. For this cause effective applications of IoT systems for disaster prevention, such as monitoring an alarm for when a flood occurrence is expected. The IoT technology that is available in today's industry has evolved to be quite advanced and can be very useful in disaster situations. The main focus of an IoT-based disaster prevention system is mainly to monitor environmental conditions and decide whether or not to send an alarm to the relevant destination, though some systems can be configured to predict possible disasters (Du C., and Zhu S., 2022).

2.6 Disaster Occurrence

A disaster can be seen as an unfavourable, undesirable event that causes physical harm or damage to lives and property. The United Nations International Strategy for Disaster Reduction (UNISDR) defines a disaster as a "serious disruption of the operation of a community or society causing general human, material, economic or environmental damages which exceed the capability of the affected municipal or society to cope using its own resources". Disasters range from structural disasters like a collapse of buildings, man-made fire incidents, etc. These types of disasters are relatively less damaging and easier to control than disasters like floods and earthquakes which are of natural causes and mostly affect a larger population. In a study by the UNISDR, it was discovered that natural disasters make up a large part of the most recently recorded disasters worldwide, with flooding as the most frequent natural disaster.

Research into different ways of managing different kinds of disasters has intensified especially with the advent of climate change and the increasing frequency and magnitude of disasters. Different kinds of disasters require different approaches to

manage them. For example, it is more likely to experience floods in a tropical climate during the rainy season or around an area where there is a water body close by, and hence disaster prevention here is more straightforward. For less predictable disasters like earthquakes, the technique for managing them are different and less straight forward.

2.7 Flood disasters

Flood disasters are not uncommon in many areas around the world, though they do result from a variety of causes. The most identifiable of which are: excessive rainfall, water body bank overflowing and poor drainage systems (Abo-Zahhad M. et al., 2016). There are three school of thoughts on the causes for the increase in flood disasters globally:

- Global warming and climate change that is increasing the amount of rain and also causing polar ice caps to melt is increasing the amount of water in water bodies that can lead to bank overflowing directly and or indirectly.
- Abuse of physical environments such as poor disposal of waste materials and poor drainage planning.
- A combination of the first two and the event of prolonged and torrential showers of rains can also result in devastating flood disasters.

2.8 Strategies Employed in Flood Disaster Prevention

1. Risk assessment and mapping

Flood disaster prevention requires extensive information and data about potential flood causes and the amount of damage that can be done (Dimesah I.E. et al., 2018). Conducting a proper vulnerability analysis for disaster-prone areas helps provide information like past occurrences of flood disasters, major infrastructure that is most likely to be damaged in the event of a flood. Risk assessment and flood hazard mapping make extensive use of vulnerability analysis and technology such as Geographic Information System (GIS) to identify areas most vulnerable to flooding and then predict the frequency, intensity, impact and other vital data that can be used to prepare for potential flood

disasters.

2. Early Warning systems (EWS)

One very popular method in disaster prevention is the use of an early warning system (Poslad S. et al., 2015). An early warning system is an integral type of data-driven Internet of Things system that is used for environment disaster risk assessment and disaster effect mitigation. An EWS is used to ascertain, forecast and mitigate the danger of disaster and any other undesirable events. Early warning systems provide real-time data about the current state of the area that is being monitored. They can incorporate machine learning algorithms to be able to better predict disasters. When using an early warning system, it is important to incorporate both immediate and mass data processing functionality. Systems have to be able to retrieve data about the state of their current environment and should have some good measure of reliability in generating the correct responses for various situations and should be able to send environment data and alerts to the appropriate destination. An early warning system should support four major functions. They are as follows:

- (a) Ability to analyze the risk and vulnerability to predetermined disasters.
- (b) Ability to monitor relevant parameters to be able to generate an accurate warning in time.
- (c) Ability to communicate analysis of risk and alerts to those affected.
- (d) Ability to create disaster response mechanisms that leverage on the local capability and preparedness to react to warnings.

2.9 Review of Past Related Works

Du C., and Zhu S., (2022) proposed a Smart Energy Management System (SEMS) based on the need to develop a system which was able to make electrical power utilization and distribution more efficient. The system was able to keep track of and manage electrical power consumption by consumers. It considered the electrical appliance usage pattern of a consumer and operates the most useful with a higher priority. The power supply utility company sets a maximum demand limit for consumers at various times of the day, and the consumer is not

allowed to surpass this limit as it is enforced by the Smart Energy Saving System. During operation, the SEMS allows operation of the maximum number of high priority devices that can be used without violating the maximum demand limit and it switches off other appliances that would cause the maximum demand limit to be exceeded if they are turned on. It also allows the consumer to reduce energy consumption by allowing certain electrical appliances to only receive power for a specific duration of time each day. The duration of time is causally determined by the SEMS. The SEMS comprises of a Smart Energy Management (SEM) Unit located at the consumer's location and enables monitoring and control functionality for a consumer. The SEM unit also provides a means of communication between the utility and a consumer. The SEM unit receives data like the maximum demand limit from the utility and sends energy usage data back to the utility for processing and updating the maximum demand limit for consumers and billing the consumers. The study on smart energy management using IoT framework by Saleh M. et al. (2017) is similar to this current study in the fact that it seeks to manage energy consumption using IoT and allows for remote control of electrical appliances. It differs in the sense that it allows for a maximum energy demand limit to be set by the utility company to regulate the amount of energy that can be consumed and it allows for the system itself to make decisions on which appliances would be supplied with power. The system proposed in this study does not specify a limit on the amount of energy that can be consumed.

To be able to know how to properly manage the energy, we need to understand how it is being consumed; while on a small scale like in a home, there is a manageable amount of data we would need to analyze on a larger scale handling this data can become problematic.

Al-Ali et al. (2015) proposed an energy management system using IoT and Big Data technologies. The system collects data about energy consumption in an area and uses the data to

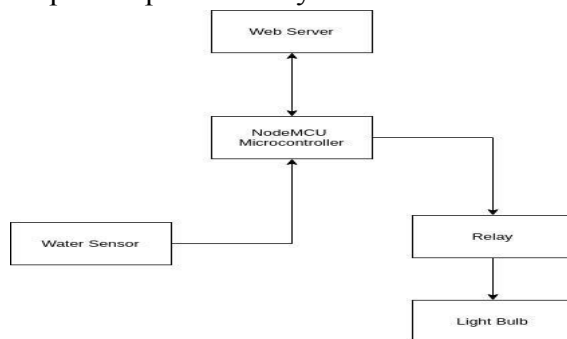
compute the best way to both manage energy consumption in a better way and to satisfy consumers. The data is collected using an array of wireless sensors in residential areas, and then analyzed using the using Business Intelligence (BI) platform. The analyzed data is provided to homeowners in real-time to aid significant insights on their energy consumption statistics. Through the use of these statistics, homeowners are able to better identify ways to save energy in their homes. This study takes advantage of the fact that wireless sensor networks (WSNs) used in IoT systems can make use of other network protocols apart from http to use the MQTT protocol. MQTT is lighter than http and can help the system to achieve better scale in order to accommodate a larger number of users.

Anastasi G. et al. (2011) proposed a mechanism to address the energy resources management in a situation where there is a collaboration between various consumers (that can generate energy locally) in a building and a Building Manager to provide energy. The system proposed was able to plan how energy resources and the task of generating energy would be distributed among consumers in response to various parameters like the availability of locally produced energy. The system aims to optimize the building manager's profit and the total net profit of the consumers collaborating simultaneously. This study finds similarities to the present study based on the fact that it is geared towards energy management. It developed a mechanism that was able to predict effective ways of optimizing energy consumption and profit using mathematical programming without obtaining real-time consumption data which makes it dissimilar to the study at hand.

Du C. and Zhu S. (2022) proposed an energy management system for DC microgrids that functions by employing an energy supporting and consumption strategy based on a priority list. During power generation, the main priority is to support the energy demand from the microgrid by the photovoltaic cells. In this manner the photovoltaic cells reach their Maximum Power

Point Tracker (MPPT) power, then the system shifts priority to support the loads if the demand of the micro-grid is higher than the photovoltaic cells and interlinking circuit. At the consumption stage, the load has higher priority; also, when the battery has a low State of Charge level, it needs to be charged by the photovoltaic cells. Any extra photovoltaic cell power can be put back into the utility grid. When there is no need for PV power, the system would function under no MPPT power. In the event that the energy being generated cannot support the load demand, then load scheduling strategy would be employed to shed all the loads demanding power which are not critical.

Pawar P. and Vittal K. (2019) carried out research on a cloud and android based home automation system that could be used, among other things to control and monitor appliances in a home. The system is organized in a way which allows for multiple users to control appliances through an android application and a web application. The system takes advantage of the available cloud services like the Google Cloud Platform (GCP) to provide one of its component parts; a web server to store customer records and support services to the other components. The other hardware component part of the system is an android mobile



device running the application software for the system. This device is used in accessing the mobile application and sending requests to the server on the cloud.

2.10 Summary of Literature Review

From the preceding works reviewed, it can be seen that energy management and disaster alert prevention are two very unique fields of research

which are hardly related, but this study tries to combine the two using the technology of Internet of things (IoT). The system proposed here can be used to communicate with both electronic devices and sensors to achieve both energy management and disaster prevention, albeit in a smaller scale (like in a home). The system can be likened to a home management system where different appliances are remotely controlled and monitored (and thereby the overall state of the home can be monitored).

III. DESIGN METHODOLOGY AND DISCUSSIONS

3.1 Methodology

This research study will be implemented using the popular NodeMCU board, which contains the ESP8266 WiFi module and a microcontroller. The microcontroller would be responsible for reading the state of a light bulb interfaced with it using a relay and values from a water level sensor. The data from the light bulb and water sensor is sent using an integrated WiFi module to a server hosted on a remote machine with a cloud service provider which in turn processes this data and stores the data in a database.

The circuit with the NodeMCU board updates the state of the light bulb from OFF (0) to ON (1) or the reverse by long polling HTTP requests sent to the cloud server, it also triggers an alarm event on the server with a request when the water level sensor reads a high level of water in its environment. The server continuously sends a notification to the user. The user can also read the state of the light bulb using a web application and sends commands to change the state of the light bulb using the applications.

3.2 The Architecture of the Proposed System

The proposed system is comprised of two major parts as shown in Figure 2: A web server and a prototype device which exchange information with each other over the Internet using the HTTP protocol.

Figure 2: Architecture of the Proposed System

3.3 Design of the System Prototype

The prototype is made up of three main components, namely: The microcontroller board, the load and the water level sensor. The load is comprised of a light bulb connected to a relay module. The prototype is supplied power from two different sources. On the load side, it is powered by 220V AC supply; while

from the perspective of the microcontroller board, it is powered using 9V DC supply provided by cell batteries. The water level sensor and the load (through the relay) are connected to the microcontroller. The microcontroller board is integrated with a WiFi module which allows it to connect with WiFi networks around its vicinity and communicate with a web server hosted in the cloud; the module connects to the Internet. The microcontroller controls the state of the light bulb and turns it off and on by changing the logic state of the relay connected to one of its pins. The water level sensor is connected to one of the pins of the microcontroller and communicates the level of water detected in the prototype's environment. The water level sensor and the relay module receive power from the microcontroller board. The Prototype Circuit Diagram is shown in Figure 3.

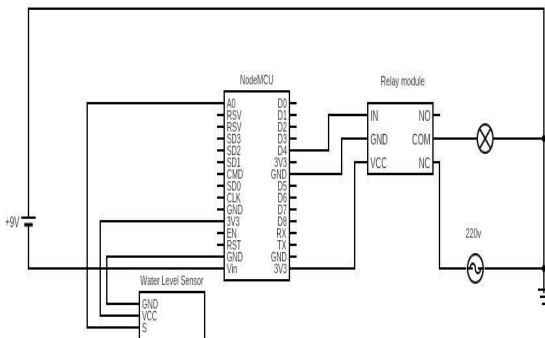


Figure 3: The Prototype Circuit Diagram

3.4 Analysis of the Prototype Components

In this section, the various hardware components which make up the system prototype are explored.

3.4.1 Node MCU Board

This is a microcontroller board which comprises of the Tensilica Xtensa LX106 microprocessor. It also contains an integrated ESP-12E WiFi module which provides the board with WiFi functionality. The microcontroller has 128 KB RAM and 4MB of Flash memory to store data and programs. Its high processing power with inbuilt WiFi features make it ideal for IoT projects. The NodeMCU board is shown in Figure 4, while Table 1 shows the

components of the NodeMCU board.

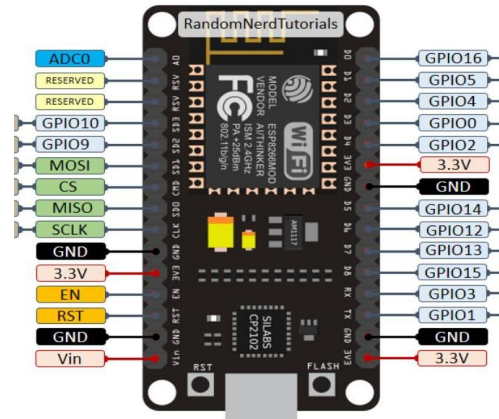


Figure 4: The NodeMCU Board

Table 1: Components of the NodeMCU board

Component	Use
Micro-USB Port	This is used to provide power to the board and also used to load programs onto the board.
ESP-12E Chip	This is used to provide WiFi functionality
2.4 GHz Antenna	This is the WiFi Antenna used for connecting to WiFi networks.
Reset Button	Used to reset the NodeMCU board's microcontroller
Flash Button	Used to flash the NodeMCU board's microcontroller
Voltage Regulator	This is used to configure the voltage levels on the board.

The microcontroller board is programmed and is used to control and get data from other components of the system. The board also provides power for other components of the system to operate

3.4.2 Relay Module

Relays are electrically operated switches. They are used to control a circuit with a low power signal or to control multiple circuits using one signal. The most popular variant of the relay is the electro-mechanical relay which when a voltage control signal is applied to its terminals, converts the magnetic flux generated into a pulling mechanical force that operates the contacts within the relay.

The relay used in the prototype functions as a binary actuator; that is, since the relay has two stable states and completes or interrupts the circuit connecting the light bulb to the 220v power supply based on the control signal it receives from the NodeMCU microcontroller board. The output voltage level of the NodeMCU control signal ranges from 3V to 5V hence a 5V relay is used in order to create enough magnetic flux to drive the relay.

3.4.3 Water Sensor

A water sensor is a device used to detect water in an environment. It can detect the presence, the level, the volume and or the absence of water. A water level sensor represents the volume of water it measures as a voltage level on one of its pins; this can be represented in human readable form. It has an operating voltage of 3 to 5 volts.

3.5 Web Server and database

The web server is hosted on a remote machine offered by a cloud service provider. The program for handling server functions is written with the Node.js runtime which lends itself very well for web programming. The server is responsible for handling various requests both from the web application client and the prototype. The server responds to requests made by the prototype in plain text while responses to the client web application are made in JSON format. This allows the prototype and the device to easily parse responses made by the server. The server program uses an event-driven programming model in its operation. Server events are triggered by requests from the microcontroller and the web client application.

3.6 Web Application

The client that would be used in this research is a web application written with the Vue.js framework. The purpose of the client is to allow a user to control the prototype device via the server by sending HTTP requests and using the WebSockets protocol. The client web application also allows a user to view the state of the prototype device for example whether the lights are turned on if the water sensor has been triggered. It is also used in by a user to receive alerts in the event that the prototype's water sensor is triggered.

3.7 Communication Over Internet

The Communication over computer networks and indeed communication over the Internet is governed by protocols. These protocols specify how data should be transferred and received, among other things within the network. This protocol is usually documented in a stack the most popular of which is known as the Open Systems Interconnection (OSI) model developed by the International Standards Organization.

The stack shows how different parts of network communication can be arranged with the physical layer representing the medium through which communication is conducted, that is for example over a wireless network like WiFi, the data link layer displays how the package will be arranged for transport and also received for transport, the network layer which decides the way in which or the path in which packets would travel, the transport layer provides for the transfer of data in the network and is responsible for error recovery over the entire network and provides for total transfer of data. The session layer establishes, manages and terminates connections between applications. The presentation layer provides independence from differences in data representation. The application layer supports application and end-user processes.

3.8 Internet protocols

The prototype and the client application both use the TCP/IP protocol http for communication. A http client starts a connection to a http server for the purpose of sending http requests the server accepts the connection from clients and services http

requests by sending responses back to the clients. Going by the standard http model, a server cannot create a connection with a client nor send a http response to a client that has not sent a request. This makes http unsuitable for real time duplex communication as only one side can initiate communication and therefore applied to this project there is no way for the server to for example communicate directly with the prototype to notify it to turn off or on the lights.

3.9 Long Polling and WebSockets Protocols

A possible solution to the problem of real-time duplex communication in http would be to employ the WebSocket protocol. The WebSocket Protocol provides duplex communication between a client and a remote host over one TCP connection. However, this method of communication is much more complicated to implement, especially on the side of the prototype. Web socket involves things like clients initiating a handshake and a few others which would bring about complexity when incorporated into the prototype's operation. However, it is suitable for the web application since web browsers provide good ways for abstracting most of these complexities. Another solution was to use long polling http requests. Here a client sends intermittent requests to the server and each request tries to get all available events or data from the server. The server doesn't respond until new events are available. When new data becomes available, the server responds and sends the new data. When the client receives the new data, it immediately sends another request to get more new data from the server, and the process is continued for as long as communication needs to last. The Long Polling Illustration is shown in Figure 5.

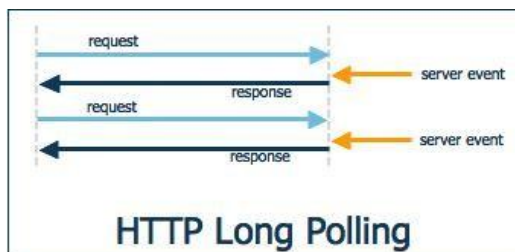


Figure 5: Long Polling Illustration

This makes it look like the server can push or notify its clients about any new events that may have occurred. In this manner, both the prototype, the client web application and the web server can exchange information with each other using a combination of long polling and the WebSocket protocol.

3.10 Flowchart for the System Operation

Figure 6 is a flowchart describing the operation of the system hardware starting from the startup of the NodeMCU microcontroller.

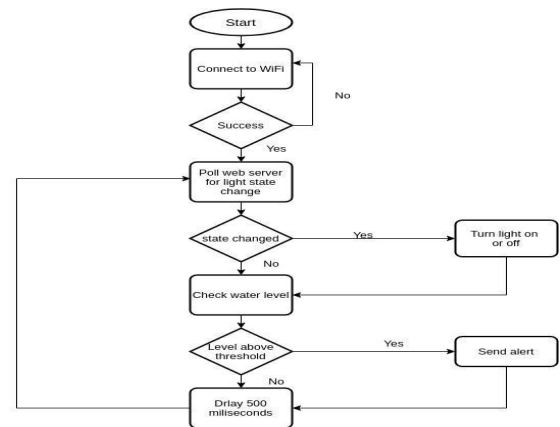


Figure 6: Flowchart for the System Operation

The microcontroller used is Arduino microcontroller, and the program is written in C programming language which is very popular for programmable hardware projects.

3.11 Implementation

The circuit in figure 3 is patched on the Breadboard. Implementation of this project occurs in stages. It is first simulated using Proteus simulation software, upon completion of the simulation exercise, the system is connected to the web.

IV. EXPERIMENTAL RESULTS

4.1 The Working Principle Results

The system starts when plugged into an electrical outlet by first connecting to a WiFi network. If no network is found after a number of retries it goes into an inactive state. After connecting to the network, it sends a message to the server to acknowledge that it is up and working. If this request fails, the system goes into standby mode.

During operation, the system continuously sends an http request to the web server to ascertain if a user has changed the state of the light bulb; if there has been a change, then the microcontroller sets the pin (configured as output) interfacing with the relay module high or low depending on the state of the light bulb received from the server. The water level sensor starts reading when it senses water in its environment and notifies the microcontroller through one of its pins (configured as input) by changing the voltage level of the pin. On receipt of this signal the microcontroller checks if this value is above the specified threshold value. If it is, then the microcontroller sends a request to the server requiring it to acknowledge that it has been notified of this change in the water level in the

	the light bulb has been successfully received and executed by the microcontroller	notifying it that the state of the light bulb has been changed to ON.
WATER_SENS ED	This is triggered by the prototype device whenever the water level in its environment passes a particular point.	The result of this event is that the server sends a push notification to the user through the web application to inform about the high water level.

4.2 Testing Results

Most of the testings done in this project focused on testing the speed of communication between the prototype device and the server and the handling of errors that occurred during operation.

microcontroller environment.

From the server-side, the server program waits for one of its events to be triggered by one of either the microcontroller or the client web application. Once an event is triggered, the server performs the subroutine already defined during programming for the event. The summary results of the possible events and subroutines is shown in Table 2.

Table 2: Web Server Events Description

Event	Trigger	Subroutine
LIGHT_STATE_C HANGED_OFF	This is triggered whenever a command to turn off the light bulb has been successfully received and executed by the microcontroller.	The result of this event is that a push notification is sent to the client application notifying it that the state of the light bulb has been changed to OFF.
LIGHT_STATE_C HANGED_ON	This is triggered whenever a command to turn on	The result of this event is that a push notification is sent to the client application

4.2.1 Latency Result

HTTP is not a persistent connection protocol, that is, a new connection has to be created each time for communication to take place; hence latency is involved.

There is also latency involved when trying to transfer information completely from the client to the web server and finally to the prototype and then from the prototype to the web server and finally to the client. Other areas where latency was introduced were seen to be caused by the network speed.

4.2.2 Latency handling

One way explored in handling latency was to eliminate the presence of the web server and to use the NodeMCU's capability of being able to act as a web server and interfacing the client application with the prototype directly. However, as with most embedded systems, the board is of low memory;

hence it would not be able to handle as many devices and as many client connections as a dedicated web server would do. The prototype would also have to be guaranteed to have only one IP address so it would be easily accessible, but this cannot always be guaranteed as it may be reset or lose connection which in effect can change its IP address. There is also a problem when the need arises to have a database of users; this would lead to too much complexity in setting up.

The system tries to handle the problem of latency by reducing the size of data that is transmitted between its components because even though having a larger size of data would provide more information to be shared it would cause even more latency in communication which would be detrimental to achieving a near to real-time system. Hence rather than requesting to get both the light states and to update the water alarm state (which only happens when the water level has been triggered), it is done sequentially; that is, one after the other.

4.3 Discussions on Error Handling and Security Issues

(a) Error Handling

In the course of operation of the system, errors may occur in which case there have to be appropriate methods for handling. The Table 3 summarizes the detected potential errors and mitigation strategies.

Table 3: Error Handling Strategies

Error	Mitigation
Network connection disconnecting	This occurs whenever the prototype is not able to connect to the internet and as a result is put into an inoperative state. The mitigation found for this was to put the prototype on standby pending when it's network becomes available

	again
Prototype request failed	This occurs whenever the prototype is not able to get a response from the server meaning either it has lost network connection or the server is unavailable. The mitigation found for this is to continuously send requests to the server
	until a response is gotten or until there have been up to 500 requests without response in which case the prototype onstandby
Client request failed	This occurs whenever the web application is not able to get a response from the server meaning either it has lost network connection or the server is unavailable. The mitigation found for this is to continuously send requests to the server until a response is gotten.

(b) Security Concerns

Although IoT is a relatively new field, there are already a lot of ways in which technology could be exploited by malicious entities. This could result in unwanted and unauthorized control of IoT devices that could cause harm.

1. Client to server security

The client to server communication needs to be secured as we do not want just any entity remotely operating the system if they are not authorized. As such a client application has to have a valid API key issued from the server in order to send

successful requests to the server. However, another security risk also exists due to the use of http because it transfers data in plain text form. This did not seem to be a big threat before, but in a situation where traffic from the client application and the server is intercepted attackers can easily read the API key headers by examining the packets. The protocol, called Hypertext Transfer Protocol Secure (https) is used to address this security issue by using encryption provided by the Secure Socket Layer (SSL) protocol to secure communication between a http client and server.

2. Prototype to server security

For the prototype, there exists less of an avenue to secure communication between it and the server due to the fact that it is harder to implement https on very low memory devices so any traffic between the prototype and the server would be visible to anyone who is intercepting it. However, this was deemed to be less of a security concern because the prototype can only act based on the commands it receives from the server. A security concern that was noticed during testing though, is that an attacker can spoof requests from the prototype; that is, they can pretend to be the prototype and send requests to the server. The mitigation found for this was to store the IP address of the prototype and reject requests from other IP addresses for all prototype URLs. The prototype, however, needs to send a onetime request to the server during startup though, containing a secret key stored on the prototype to let the server acknowledge its IP address as legitimate. Though bypasses exist for this method, it is one of the limitations of this system that prototype and server connection is not critically secure.

3. Server security

Web servers are notoriously targeted by attackers as they are the most critical point in systems where most of the data is kept. However, there are little security concerns for the server used in this project as all that is provided by the third-party cloud service provider.

V. SUMMARY AND CONCLUSION

5.1 Summary

This research started with the motive to develop an energy management and disaster alert and prevention system that harnesses the capabilities of IoT technology to control a home appliance and provides an early warning in the case of potential flooding in the home. The system consists of two parts, a web application and a prototype device which exchange information over the Internet using the http protocol and WebSocket. The web application was used to control the prototype and also obtain information from the prototype.

5.2 Conclusion

Results from the testing process showed that the system satisfied all specifications set. The aim and objectives of the system were also met; the energy management and disaster prevention showed good results during the testing process. The error which occurred during testing was properly mitigated.

The major achievements recorded in the development of this research work are as follow:

- An electrical device (a bulb) was controlled over the Internet using a web application and a NodeMCU interfaced to the bulb through a relay module.
- An early warning system was implemented using a web application.
- Long Polling https protocol was used to provide near real-time communication between the web server and the prototype device.

5.3 Challenges Encountered

The challenges encountered during the course of this study were:

- Real-time Communication: The http protocol does not extend itself for real-time duplex communication, but it was chosen for this project; therefore, a compromise had to be made using Long Polling.
- Latency: This was caused because of the overhead involved in connecting a user through a web server to the prototype device, rather than directly. Latency was also introduced based on the

speed of the network.

- Two-way control: There was no way found to enable two-way control of the light bulb, that is the light bulb cannot be simultaneously controlled effectively by a switch and the relay.

REFERENCES

1. Kitagami S., Thanh V.T., Bac D.H., Urano Y., Miyanishi Y. (2016), "Proposal of a Distributed Cooperative IoT System for Flood Disaster Prevention and Its Field Trial Evaluation," *Int. Journal of Internet Things*, vol. 5, no. 1, pp. 9–16.
2. Lu H., Chen M., and Kuang W. (2019), "The Impacts of Abnormal Weather and Natural Disasters on Transport and Strategies for Enhancing Ability for Disaster Prevention and Mitigation," *Transp. Policy*.
3. Pawar P. and Vittal K. (2019), "Design and development of advanced smart energy management system integrated with IoT framework in smart grid environment," *J. Energy Storage*, vol. 25, p. 100846.
4. Saleh M., Esa Y., Mohamed A.A., Grebel H., and Rojas-Cessa R. (2017), "Energy management algorithm for resilient controlled delivery grids," in *2017 IEEE 59 Industry Applications Society Annual Meeting*, pp. 1–8.
5. Anastasi G., Corucci F., Marcelloni F. (2011), "An intelligent system for electrical energy management in buildings," in *2011 11th International Conference on Intelligent Systems Design and Applications*, pp. 702–707.
6. Du C., and Zhu S. (2022), "Research on urban public safety emergency management early warning system based on technologies for the Internet of Things," *Procedia Eng.*, vol. 45, no. 22, pp. 748–754.
7. Abo-Zahhad M., Ahmed S.M., Farrag M., Ahmed M.F, and Ali A. (2016), "Design and implementation of building energy monitoring and management system based on wireless sensor networks," *Proc. - 2015 10th Int. Conf. Comput. Eng. Syst. ICCES 2015*, pp. 230–233.
8. Dimesah I.E., Okine A.N.D., and Kissi Mireku K. (2018), "Influential factors in creating warning systems towards flood disaster management in Ghana: An analysis of 2007 Northern flood," *Int. J. Disaster Risk Reduct.*, vol. 28, pp. 318–326.
9. Poslad S., Middleton S.E., Chaves F., Tao R., Necmioglu O., and Bugel U. (2015), "A Semantic IoT Early Warning System for Natural Environment Crisis Management," *IEEE Trans. Emerg. Top. Comput.*, vol. 3, no. 2, pp. 246–257.
10. Al-Ali A.R., Zualkernan I.A., Rashid M., Gupta R, and Alikarar M. (2017), "A Smart Home Energy Management System Using IoT and Big Data Analytics Approach," *IEEE Trans. Consum. Electron.*, vol. 63, no. 4, pp. 426–434.