Full Paper

IoT-Based Smart Power Outage Communication System

A. R. Iyanda
Department of Computer Science and Engineering, Obafemi Awolowo University, Ile-Ife, Nigeria
abiyanda@oauife.edu.ng

O. D. Ninan
Department of Computer Science and Engineering, Obafemi Awolowo University, Ile-Ife, Nigeria

O. S. Odudimu
Department of Computer Science and Engineering, Obafemi Awolowo University, Ile-Ife, Nigeria

ABSTRACT

This paper reports the development of a smart power outage communication system for use in an environment characterized by uncertainty with a view to produce a robust and an accurate performance with the least computational and data resources. Nigerian energy supply issues, which have brought about inadequate electricity supply to domestic households and industrial producers notwithstanding a speedily growing economy, have converted the average energy consumer to a power sensing and reporting tool, since our daily activities and decision-making are dependent on the availability of electricity. Often the traditional human to human communication is the way to communicate power statuses and is prone to error; however, a machine-to-human communication process is non-trivial. The system reported in this work was designed using a visual modeling tool and implemented in Python 3.6 using the Python Django framework mongoDB as Back End and hosted on the Heroku cloud platform. The results show that as power events (outages and restorations) occur, the power sensor sends a notification instantly to the android devices of all subscribed users. The notifications are received a few seconds after the event occurs. The users can also query all historical power outages and restorations using the android application, as well as unsubscribe from the power monitoring device.

Keywords: Power Outages and Restoration, GSM, Real-Time Notifications, Firebase Cloud Messaging, Android Application Software, Power Status Communication

1. INTRODUCTION

The development of the electric motor enabled large scale power generation and transmission (Chatterjee, 2002). Society has subsequently become dependent on electricity to power households, run industries, which improve economic growth, and make infrastructure available to many technological fields, such as the telecommunications industry. Ultimately, failures in power transmission have cause disruptions in both businesses and individual life; this continues to change in form and pattern in an uncertain manner, making its description imprecise and sometimes intractable. An accurate power outage communication system can be a great tool particularly in environments where the power supply is very limited, such as in many Africa countries. According to Quarshie et al. (2017), power outages (also known as failure, cut or blackout) can be defined as a short or long-term loss of electric power supply to an area. This may be caused by faults at power stations, electric transmission lines damage, substations, or damage of other parts of the distribution system and electric mains overload.

PWC (2016) concluded that the power outages in Nigeria have become very embarrassing. Nigeria has only about 25% available and operational generating capacity due to gas constraints, out of 13 Gigawatts of electricity. There is a transmission capacity of 5 GW overall, of which almost 90% of transmitted power reaches electricity consumers through the Distribution Companies (DISCOs). These distributors, however, suffer collection and commercial losses since less than 50% of electricity consumers pay for the power they consume, apart from the 7% loss of the operational generation across the distribution network. These problems have resulted in the failure of the Nigerian power sector to provide reliable and adequate electricity supply to domestic households and industries despite a rapidly growing economy (Arobieke et al., 2012; Aliyu et al., 2013).

From the dawn of electricity in Nigeria, energy utility consumers were typically left in the dark concerning the causes of outages or when power would be restored. Traditionally, consumers needed to be proactive to be in the know of power status since there was no convenient way to communicate with the utility service providers to learn when electricity would be restored. Hence, the people were conditioned to expect little in the way of communication from their power providers. Consequently, this existing system of communicating power status is unreliable and ineffective as it affects decision making.

In today's age of information and technological sophistication, most developed countries across the world have power management systems that constantly collect and process power data which is then communicated to concerned energy consumers. Adderly (2016) examined electrical disturbance data and reliability index data from the Department of Energy and state utility public service commission regulators respectively using statistical approaches with a view to detect signs of improvement. This showed that there is a repository of power data to collect and analyze for effective decision making that will benefit all energy consumers. It is however important that efficient and reliable machine-to-machine (M2M) communication be harnessed to collect power data from grid components using power sensors, which will then be stored on a centralized / distributed database such as Cloud 55 for processing and effective reporting to the concerned entities.

Due to the unreliability and inefficiency of the existing human-to-human communication process, the ubiquity of smartphone devices running the Google Android operating system in the mobile device ecosystem is a strong enabler for massive real-time communication. Thus, relying on the traditional interpersonal interaction to communicate power status is no longer needed. This study created a real-time system that communicates outages to concerned entities using their android devices. Collection of power supply data from one or more geographical locations of study can be very valuable for addressing large scale power problems.

2. LITERATURE REVIEW

It is dangerous to experience power failures at locations where the environment and community safety are at jeopardy. Areas like hospitals, sewage treatment plants, mines, shelters, and telecommunication require emergency (backup) power. It has been widely debated that power grids are self-organized critical systems based on historical data and computer modeling, as they exhibit unavoidable disturbances of all sizes (Carreras et al. 2002; Hoffmann and Payton, 2014). This results in the gradual increase of demand / load, the economics of operating a power company, and the restrictions of modern engineering (Dobson et al, 2007).

Mohamed and Mohamed (2017) covered most of the current communications schemes used to help to provide accurate and precise control of setting up of a power system. The author emphasized that communication has always played a critical part in having a safe and efficient operation of the electric power grid. as well as when implementing an end-to-end and two-way open communication grid infrastructure. In addition, automation and control of electric utility generation, transmission, and distribution systems in real-time rely on dependable and secure communication networks.

Mehmood et al. (2015) surveyed M2M communications based on mobile networks with focus on the latest Long-Term Evolution-Advanced (LTE-A) networks. He also presented the use of M2M in Smart Grid metering and monitoring service being one of the main driving forces for increased M2M traffic. In addition, M2M communications improve the performance of the smart grid which in turn improves the performance of energy generation, distribution, and consumption.

Chatterjee et al. (2018) emphasized that real-time communication is essential for lives. Their application assisted in instant transfer of files and messages with minimal delay or none, subject to the broadcast medium. Android-based real-time communications were achieved using Google Firebase with other features to provide a real-time database server. The authors offered a system that is able to send text-based messages and files (e.g., images, audio, videos, texts) over the Internet between two users on the network in real-time using Android operating system and Google Firebase to handle the backend of the communication operation. The study concluded that "Firebase services are more efficient and faster compared to building a traditional server-side database using a scripting language.

Eskandarpour and Khodaei (2016) proposed improved power grid resilience using a machine learning model (a multi-dimensional Support Vector Machine (SVM)) with a view to predict the component states in response to extreme events (outage and operational). k-fold cross-validation and model benchmarking techniques were used to validate the model and tests used numerical simulations based on a precise and commonly used performance measures.

Prathik et al. (2018) presented energy consumption and effective use of home appliances for saving energy in India by giving information on meter reading, power cuts, and the alert systems for generating an alarm when energy consumption exceeds the specified limit, using IoT. This reduced having to depend on humans for monthly reading collection and hence reduce technical problems associated with the billing process. An Arduino microcontroller and a GSM (Global System for Mobile Communication) module were used to implement the system. This system provided the user with information on the power consumed daily and hence help to conserve energy. Also, information on the bill amount, payment and the pre-planned power shut down details were communicated to the user. In addition, it reduced wastage of energy by terminating power supply when the user was out of station.

Karim (2018) predicted that energy demand will increase globally as the world economy is growing rapidly, and hence become more expensive, affecting the economic development. The study was based on the available low power wide area network (LPWAN) technologies. Results of range tests carried out in an industrial area shows that LoRa (Long Range) technology has higher communication distances as compared to legacy WAN technologies. The study therefore presented a system architecture for an energy management system (EMS) using LoRa as the primary communication technology as well as an ideal implementation of it, integrated into the current EMS of Enisyst.

Santos and Ferreira (2019) presented a system that offers real-time information and a descriptive analytics process using Internet of Things (IoT) LoRa to monitor power consumption with a view to give energy consumption over time and identify waste. The solution can be used in different situations since it allows for easy installation without communication range and obstacle restrictions.

In this paper, we look to create a similar application based on previous works and determine how IoT can be used for sending the data directly to the cloud to prevent problems arising from long distance communication and restrictions of cable transmission. The power monitoring device developed collects power data and sends it to the cloud for storage, creating a repository for power data in the country. In terms of predicting power status, an effective human power predictor is that person with the ability to constantly monitor power outages and restorations at any given time, keep track of past power outages and restorations and solve statistical problems expertly in order to accurately predict the next power event that will occur. The aforementioned capabilities are almost impossible for any human being to achieve, as data will continually grow and require more computational power for accurate predictions to happen. However, this study provides adequate information to develop an algorithm for predicting power outages and restorations.

3. METHODOLOGY

This section provides the design architecture of the entire system as well as the design of the subsystems. The Department of Computer Science and Engineering, Obafemi Awolowo University Ile-Ife National Association of Computer Science Students (NACOSS) Secretariat was used as a case study. The system was modularized into three subsystems which are the Power Monitoring Sensor sub-system, API, and the Android Application Software subsystem. The process flow of the interacting subsystems is shown in Figure 1. The study seeks to implement an IoT based smart power outage communication system (SPOCS), which is a real-time system designed to communicate power outages and restorations.

The power monitoring device captures power events (outages and restorations), sends the captured payload containing the event data (event_data), device number (device_no) and the timestamp of the event over a GSM network using an embedded GSM Module interfaced with an Arduino board, via an HTTP protocol; it sends an HTTP post request over TCP/IP to an API endpoint that serves as the streaming data processor, the streaming data processor, which is hosted on a web server, fires the storage API and returns the timestamp of the server to the power monitoring device for effective clock synchronization between the sensor and the server. The storage API stores the received payload on the database, calls the notification API, which constructs the applicable notification message and data based on the event that occurred, and dispatches the constructed message to Google Firebase Cloud Messaging Platform. This platform ensures that the sent notification is received on the subscribed end-user's phone through the developed android application. Registered users on the application are subscribed to the installed power monitoring device by default. The user can then query historical data as well as predictions made by the machine learning model in the cloud.

The Hardware Subsystem deals with the various hardware components used for the design of the project as shown in Figure 2. It also deals with how the components are interfaced with one another to realize the objectives of the project. The hardware subsystem is subdivided into three:

i. The input unit consists of the rectifier circuit and the real-time clock (RTC) Module.
ii. The control unit consists of the Arduino UNO.
iii. The Output Unit which consists of Indication LEDS and the GSM Mobile device.

The software programming interface of the hardware subsystem is further subdivided into two, which are the Control Unit Programming and the GSM Module Programming. The entity relationship diagram of the entire system is shown in Figure 3.

4. IMPLEMENTATION

The whole system was modularized into three subsystems which are the Power Monitoring Sensor subsystem, the Power Communication System Middleware (API) and the Android Application Software subsystem. The Power Monitoring Sensor subsystem is divided into two parts (Hardware system and Software system).

The hardware system part was implemented using an Arduino Uno, a GSM Module, rectifier circuit, led indicators, active and passive electronic components constituting the Input Unit, a Control Unit, an HTTP communication unit, and a Power supply unit. The input unit of the system consists of the rectifier circuit and the Arduino ADC. The rectifier circuit, which is a part of the power supply circuit, converts the AC input from the power source to DC power, which is then connected to the power input of the Arduino board. The power pin of the Arduino board is constantly monitored for detection of outages and restoration from the connected AC power source. The power state read from the power pin is then

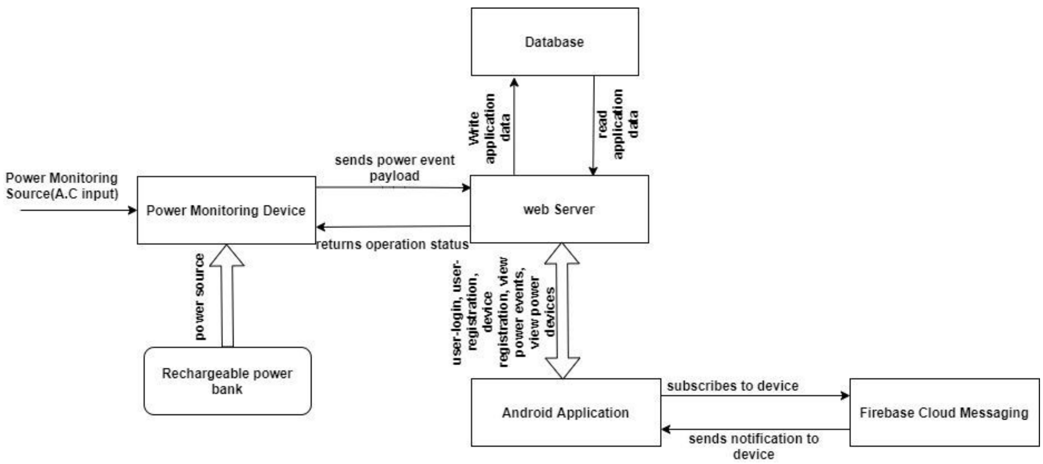


Figure 1: The process flow of the developed system

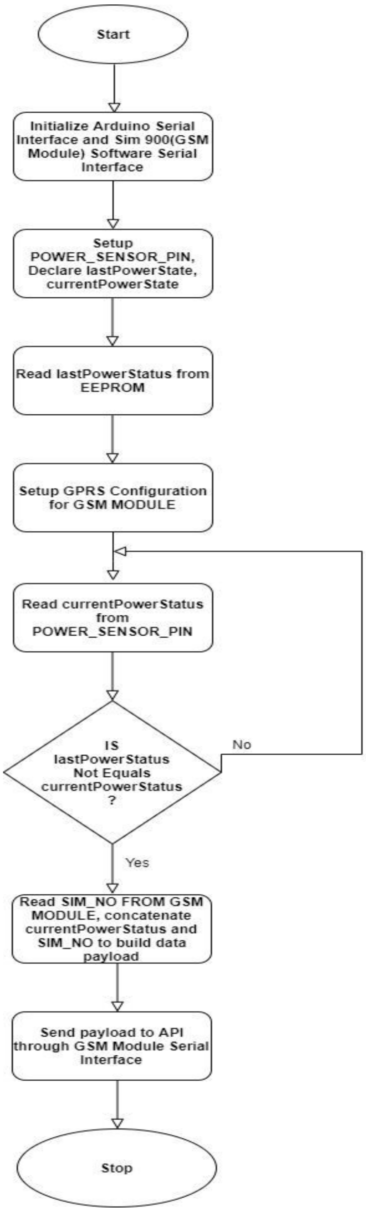


Figure 2: The Hardware Subsystem Flowchart

converted to a digital level through the Analog to Digital converter (ADC) of the Arduino board. Each of these states is stored temporarily in a state variable named currentPowerStatus, in order to keep the last state of the sensor in sync with the last state stored in the cloud (database). The currentPowerStatus gets stored in another state variable named lastPowerStatus which then gets stored in the EEPROM of the microcontroller (Atmega328p) permanently.

The control unit of the system has the Arduino board as its Central Processing Unit. The hardware subsystem, which is the power monitoring sensor, sends the detected power status from the input unit through a GSM modem which uses the HTTP protocol to establish a connection to the cloud by opening a GPRS context with the network operator, set up an HTTP post request with the data to post (payload) following the HTTP 1.1 protocol, terminate the HTTP context, and close the GPRS context. The Arduino board transmits and receives data to / from the GSM module (SIM900A) over Arduino's USART (Universal Synchronous / Asynchronous Receiver / Transmitter) interface by submitting AT commands to the GSM Module (Sim900A) using the Arduino Uno ‹SoftwareSerial› C++ library. The GSM Module then interprets this command accordingly and interfaces with the network operator relative to the command specified. For example, to query the phone number of the sim card present on the module, the AT command "AT+CNUM" returns the phone number present on the device.

A 5 V power supply circuit was built to power the Arduino board and the supporting modules. The circuit has a power supply unit, a step-down transformer, a bridge rectifier, electrolytic capacitor, and voltage regulator (LM7805). The step-down transformer steps down the 240V AC to 12V. The 12V is rectified by the bridge rectifier (converted to DC voltage) and filtered for any ripple with the 470microFarad electrolytic capacitor. The filtered output is then regulated to 5V DC to power the Arduino. However, the Arduino board uses an external rechargeable power bank as a power source which is independent of the digital voltage level measured from mains through the Arduino input pin.

The application programming interface (API) was developed in a Linux environment due to the ease of access to open-source software resources. Also, since the API will be hosted on a Linux virtual machine in the cloud, it made sense to develop locally on a similar operating system so

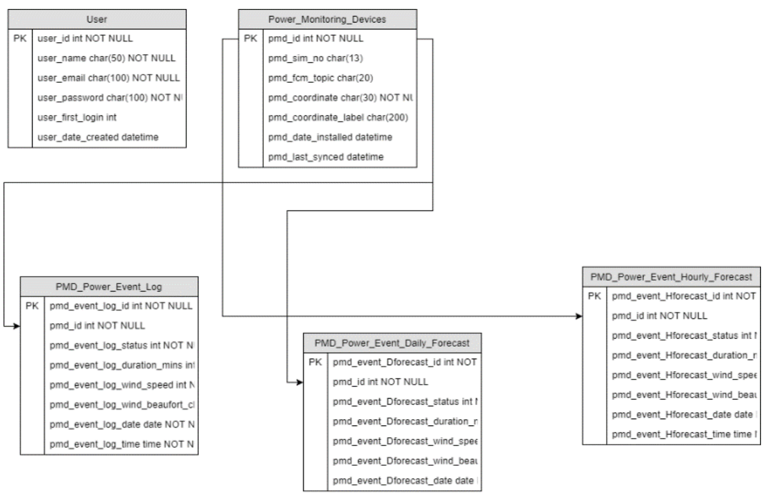


Figure 3: The entity-relationship diagram of the proposed system

that deployment to the production environment would be seamless. The Django web framework is the backbone of the API subsystem of this work; MongoDB was used for the backend database of the API. The MongoDB database was hosted on mLab while the API source code was hosted on a web server sitting on Heroku's cloud platform which is a cloud-based platform-as-a-service (PaaS) supporting several programming languages.

The Android Application Software subsystem was developed in the Windows OS environment because of the simplicity of the OS environment, and the fact that it is reasonably well documented, and its sources is available. The Android Studio IDE provides a unified environment where the android application was built. For the development of the android-based event notification system, the core programming used is java but the front-end, Androids XML was used. It provides the skeletal frame design, flexible enough that users will be able to interact with. The users' interface was built using a hierarchy of layouts that includes text views, buttons, and web view. Android Studio was used to provide some other services aside layout editing such as APK (Application Package) analyzing and Vector Asset Studio work. Firebase Cloud Messaging (FCM) was integrated into the app with just a few lines of code for sending real-time power event notifications to subscribed users with a web console called Firebase Notification.

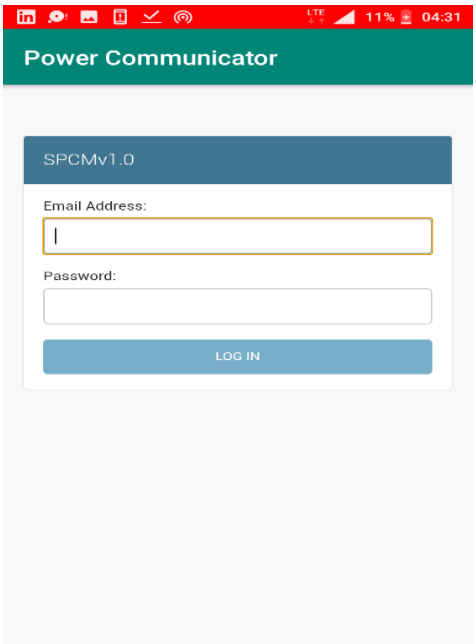
The power monitoring device senses power outage / restoration from the digital input pin of an Arduino, the Arduino board then builds a data payload which contains the power status (digital 0 for an outage and 1 for restoration), the device identification number (the sim number of the sim card in the GSM module(sim900A)) and forwards this information to the GSM module (which is connected to the internet). The GSM module then sends this information to the backend restAPI hosted on the cloud. The power status change (i.e., outage to restoration and vice-versa) sent to the backend is done by tracking every state change using two-state variables currentPowerStatus and lastPowerStatus. currentPowerStatus stores the current digital voltage level measured from the power pin and lastPowerStatus read / write the last power state from / to the Atmega328p. The data payload is only sent to the backend when there is a state change i.e., currentPowerStatus not equal to the lastPowerStatus.

For users to get notified of power changes in a particular location on their mobile devices, these users must subscribe to the power monitoring device installed in such location. To subscribe to a power monitoring device, end users must select the device of interest and subscribe through the user interface of the mobile application. The restAPI receives the payload, uses device serial number to query the list of users subscribed to the device, takes the current timeStamp the event occurred to compute power uptime / downtime (in minutes), uses device serial number to retrieve the geographical coordinate (registered on the device registration page during device installation) where the device is. The restAPI, then saves this information (power event uptime / downtime, power event status, and weather data) into the Power Event Log Table, in order to notify all subscribed end-users, the restAPI consumes Google Firebase Cloud Messaging API by passing the list of subscribed users device token and Notification Message to the API which in turn delivers the message to their mobile device in real-time as indicated. Once the users receive and click on the notification on their mobile device, the power event log details page is then displayed. However, end users on the other hand can view the history of power restoration / outages by navigating to the power event page.

The app pages are hyperlinked which makes it possible and easy to navigate through and explore pages on the app. There are details on each page that users can familiarize with, where any navigation tab of choice can be selected. The pages are as follows:

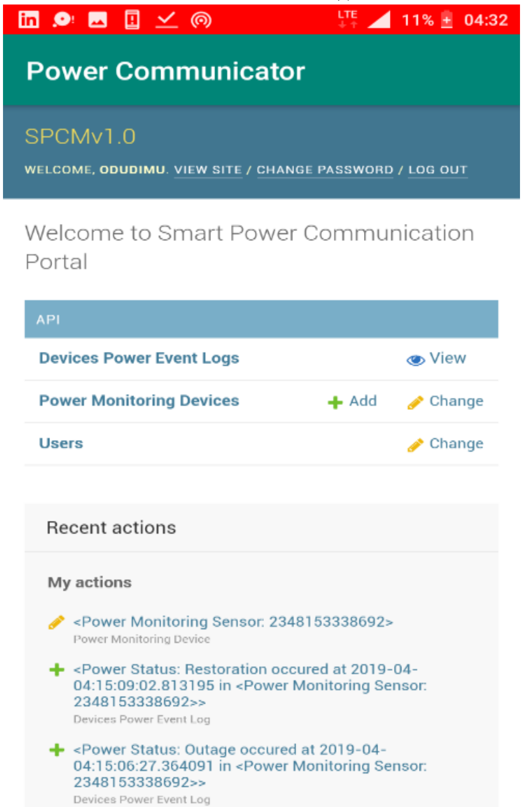
i. Landing page: This is the first page (main activity), the user interfaces with either the login page or the landing page depending on whether he is logged in already, if no user session exists on the app, the first page becomes the login page, where the user enters his/her login details to gain access to the application. The landing pages are shown in both cases mentioned above in Figures 4a and 4b respectively.
ii. User Creation page: This page can be accessed only by an admin user to create users that will be granted access to test the system as shown in Figure 5.
iii. Subscribe page: On installation of the application, a device registration token is generated from FCM and copied into the clipboard by clicking on the subscribe button, on the subscribe page, the user selects one or more power monitoring device(s) to subscribe to and registers the generated device registration token against the subscribed devices by pasting the generated token in the user device token textbox control and click save to submit the data into the database. The subscription page is shown in Figure 6.
iv. Power Monitoring Devices page: Only an admin user is allowed to view this page. The administrator registers power monitoring sensor(s) details on the app by filling in device ID(s) (in this case, the sim number of the device), notification topic and location details of the installation environment to be monitored. The power monitoring devices page is shown in Figure 7.
v. Power event logs page: This is the page where recent and historical power records can be found on the app, it contains a list of power outages and restorations summary which gives an overview of the power event that occurred, where and when it occurred, each of these events have a detailed view that gives the user the full information of the power event including the weather situation that accompanied the occurrence of the event. The power event logs page is shown Figure 8 with records of power outages and restorations captured on the system in NACOSS Secretariat and the detailed view of one of the events is shown in Figure 9.
vi. The Notification view: This is a view created by the android operating system through FCM for every

message sent to the server from the API endpoints; this message is displayed with the notification view. The notification view is shown in Figure 10.



(a) Login

(b)



(b) After Login

Figure 4: Landing Page

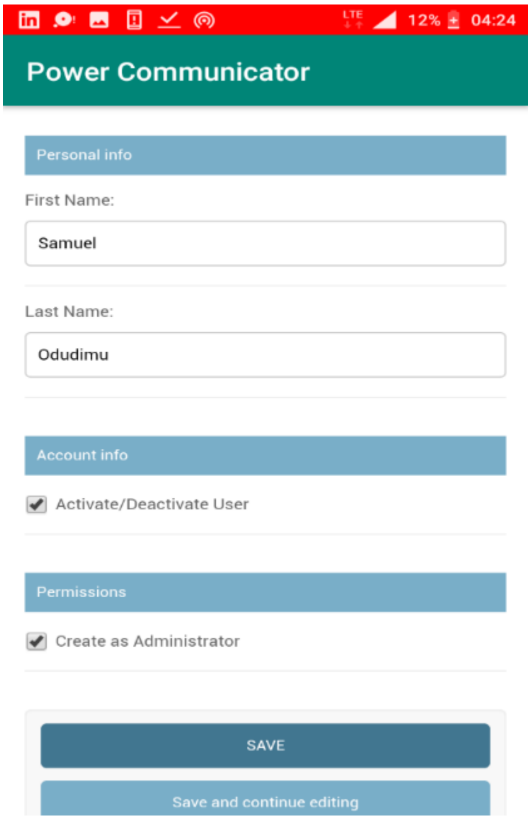


Figure 5: Admin user creation page

Figure 6: Device subscription page

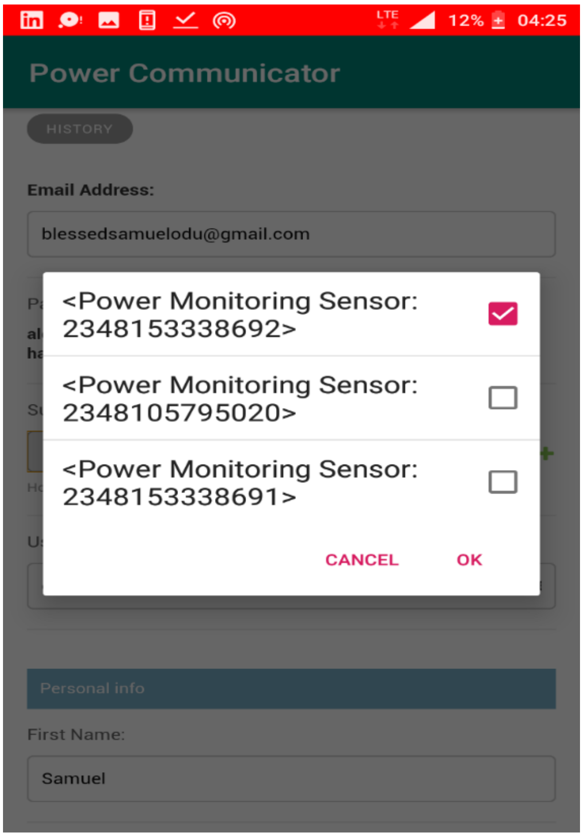


Figure 7: Power monitoring devices page

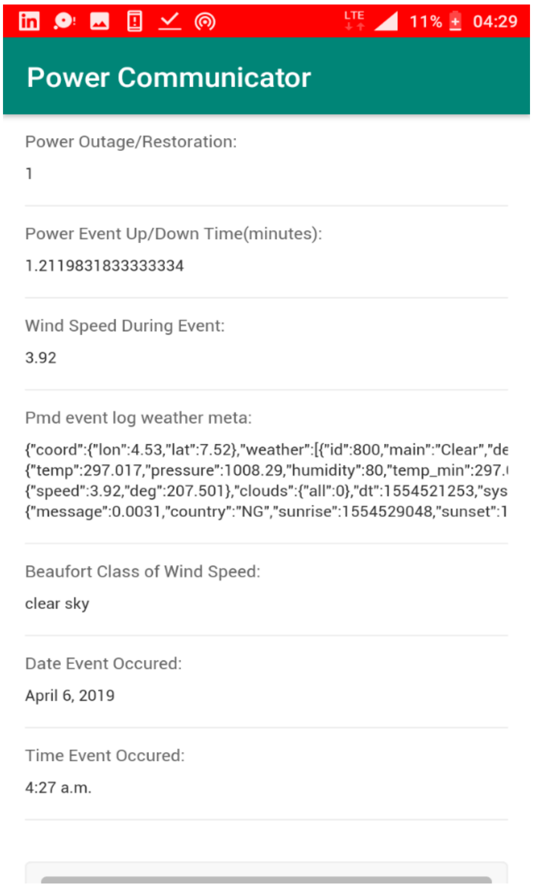


Figure 9: Detail view of a power event

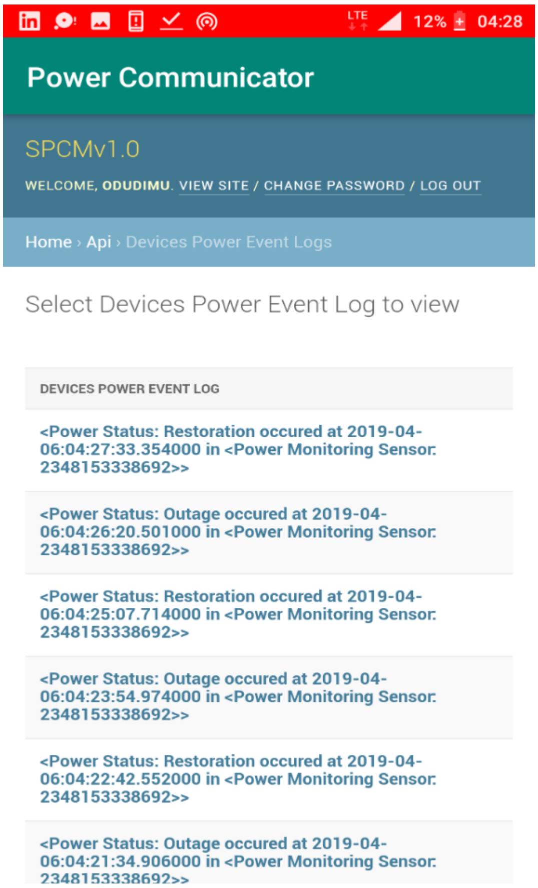


Figure 8: Power event logs page

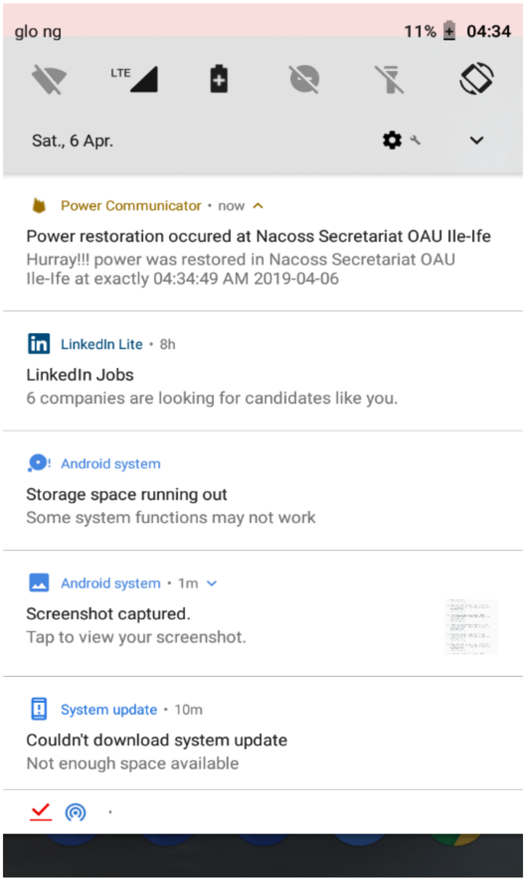


Figure 10: Notification view

5. RESULT

During the testing phase, a total of 12 end users tested and evaluated the system by installing the application on their various devices, each user's details were profiled by an admin user of the system. the login details of the profiled accounts were given to each respondent to access the system. All the respondents were students of Obafemi Awolowo University, nine (9) of which were from the Department of Computer Science and Engineering, two (2) from the Department of Electrical and Electronics Engineering and one (1) from the Department of Management and Accounting. The age range of respondents is between eighteen (18) and twenty-five (25) years. All respondents have previous experience in the usage of mobile applications on their mobile devices as they all installed the software on their devices without guidance. Each evaluation metric was rated on a scale of 1-10, the average rating and percentage of analysis of each of these metrics were taken against the total number of respondents that tested the system. The result of the system evaluation is shown in Table 1. The average rating for Utility was 80.8%, for Efficiency was 69.2%, for Responsiveness was 89.2%, and for Usability was 82.5%.

Table 1: Result of Evaluation

User	Utility	Efficiency	Responsiveness	Usability
1	7	5	9	7
2	8	6	9	8
3	10	7	8	7
4	5	8	10	9
5	9	5	9	9
6	8	9	9	8
7	10	8	10	8
8	10	7	7	8
9	7	7	8	7
10	8	8	9	9
11	9	6	10	9
12	6	7	9	10
Average rating	8.08	6.92	8.92	8.25
Percentage (%)	80.80	69.20	89.20	82.50

6. CONCLUSION

In this work, a low cost, low power consuming and efficient IoT-based smart power outage communications system based on GSM technology has been built successfully. Its functionality also shows that it can be used in real-life residential, corporate, and industrial environment monitoring. However, the resulting power monitoring device does not guarantee eventual consistency that is, all captured data from the power monitoring device may not get to the cloud successfully due to bad internet connectivity. Ensuring consistency might require detecting internet connectivity, maintaining an in-memory queue for every failed Http request from the GSM module and resending them when internet connectivity is good which might lead to receiving Out-Of-Order notifications by the end-users.

To ensure the power monitoring device meets with the demands of a real-time system, Priority was given to speed and simplicity over eventual consistency. The study proposed that other technologies such as Zig-bee can be used instead of GSM technology and apart from the use of the HTTP protocol for client-server communications between the monitoring device and the cloud, other simpler, faster, lightweight, and more reliable mobile network communication protocol such as USSD can be used. Based on the power data gathered, a predictive model can be used to forecast the next power event using the captured variables.

REFERENCES

Adderly, S. (2016). Reviewing power outage trends, electric reliability indices and smart grid funding. Graduate College Dissertations and Theses. 531. https://scholarworks.uvm.edu/graddis/531

Aliyu, A., Ramli, A. and Saleh, M. (2013) Nigeria electricity crisis: Power generation capacity expansion and environmental ramifications. Energy, 61(8), 354-367. doi:10.1016/j.energy.2013.09.011.

Arobieke, O., Osafehinti, S., Oluwajobi, F., Oni, O. (2012) Electrical Power outage in Nigeria: History, causes and possible and solutions. Journal of Energy Technologies and Policy, Volume 2, No.6, 2012, ISSN 2224-3232.

Carreras, B. A., Lynch, V. E., Dobson, I., & Newman, D. E. (2002, January). Dynamics, criticality and self-organization in a model for blackouts in power transmission systems. In Proceedings of the 35th Annual Hawaii International Conference on System Sciences (pp. 9-pp). IEEE.

Chatterjee, N., Chakraborty, S., Decosta, A., & Nath, A. (2018). Real-time Communication Application Based on Android Using Google Firebase. International Journal of Advance Research in Computer Science and Management Studies, 6(4).

Dobson, I., Carreras, B. A., Lynch, V. E., & Newman, D. E. (2007). Complex systems analysis of series of blackouts: Cascading failure, critical points, and self-organization. Chaos: An Interdisciplinary Journal of Nonlinear Science, 17(2), 026103.

Eskandarpour, R., & Khodaei, A. (2016). Machine learning based power grid outage prediction in response to extreme events. IEEE Transactions on Power Systems, 32(4), 3315-3316.

Hoffmann, H., & Payton, D. W. (2014). Suppressing cascades in a self-organized-critical model with non-contiguous spread of failures. Chaos, Solitons & Fractals, 67, 87-93.

Karim, A. (2018). Iot based monitoring and control for energy management system, 415 Master's Thesis. University of Stuttgart, Stuttgart

Mehmood, Y., Görg, C., Muehleisen, M., & Timm-Giel, A. (2015). Mobile M2M communication architectures, upcoming challenges, applications, and future directions. EURASIP Journal on Wireless Communications and Networking, 2015(1), 250.

Mohamed, Y. and Mohamed, V. (2017). Current communication media in power system, International Journal of Research in Electrical Engineering 4 (2):145–147. ISSN: 2349-2503

Prathik, M., Anitha, K., & Anitha, V. (2018, February). Smart Energy Meter Surveillance Using IoT. In 2018 International Conference on Power, Energy, Control and Transmission Systems (ICPECTS) (pp. 186-189). IEEE.

PWC (2016) PwC's Annual Power and Utilities Roundtable: The challenges with transforming the Nigerian power landscape (Report). PWC. p.4. ‹https://www.pwc.com/ng/en/assets/pdf/power-roundtable-2016.pdf ›.

Quarshie, D., Agyeman, B., & Bonn, J. (2017). Power outages on performance of selected manufacturing firms on the Ghana stock exchange.

Santos, D., & Ferreira, J. C. (2019). IoT Power Monitoring System for Smart Environments. Sustainability, 11(19), 5355.