



Smart Vending Machine Counter using IoT

Athir Jaafar¹, and Mohd Norzali Mohd^{1*}

¹Faculty of Electrical and Electronic Engineering,
Universiti Tun Hussein Onn Malaysia, Parit Raja, 86400, MALAYSIA

*norzali@uthm.edu.my

1. Introduction

Vending machines (VM) normally operate in areas such as public transit platforms, stadiums, freeway rest stops, or corridors in shopping malls. These types of locations cause problems for IoT solutions exclusively reliant on the cloud for the purpose of data processing aspects [1]. Within smart VM, the user can order products using a smart phone without interaction with the vending machine in this system. Tallying smartness to a machine similarly increases complexity. While the cost of sensors continues to drop, it is important to outfit an entire cabinet with sensors on every shelf, slot, or spiral. In addition, software versions and configurations require periodic updates. These considerations raise the stakes for retailers seeking to retrofit existing vending machines for today's connected world [3]. The next generation of intelligent vending machines proposes huge opportunities for business transformation through the (IoT) and cloud-based technologies [2]. VM vendors and operators can look forward to more effective ways to engage customers, increase sales, and save money through remote management and predictive maintenance. Advanced data analytics help them better match machines and stock to each location, as well as obtain valuable insights on the effectiveness of promotions, pricing, location, weather, season, and other factors [2]-[4].

Internet of Things (IoT) was first introduced in 1999 by Kevin Ashton in which was applied into Supply Chain Management (SCM), [5]. Sooner or later the utmost of the systems and devices as well as entities surrounding us will openly or ultimately be accessible on the present working network. For the purpose of connecting with IoT, plentiful of sensor network technologies engendered and produced such as Radio Frequency Identification (RFID) [5].

With traditional vending machines, information is collected when machines are restocked and serviced. This creates a serious delay in inventory monitoring. It also impacts sales reporting compliance in jurisdictions requiring vending machine operators to report taxes on sales on a timely basis. Infrequent visits also make it hard to know when a machine is malfunctioning and losing sales. Traditional vending machines require service visits to update prices or begin a promotion, such as two-for-one deals. Since many vending machine owners rely on third parties to stock machines, there is no safeguard to prevent the vending of unauthorized products. The vending machines would experience downtime issues periodically. Sometimes they'd be entirely offline, while other malfunctions might leave the vending machine still functioning, especially when inventory run out of stock. Every minute of downtime was money the company is losing [6][7]. This is a paramount problem especially when the

VM is completely in the “Blackout Status” and keep sending their workers and salespeople to find out and check the inventory status. This is completely a waste of time, money and deficiency in their sales capability to generate. Their Point of Sale (POS) of every vending machine highly deteriorated due to the above causes.

The following applications were identified and developed as part of the solution.

1. Open software development board like NodeMCU ESP8266 have been setup in the implementation in order to get an affordable solution. The system requirements for remote monitoring is control application run on the vending machine itself, gathering data from sensors and handling operation of the electromechanical aspects of the solution (e.g., turning the vending coils) as well as data exchange with both human users (e.g., workers and administrators) and with the cloud.
2. The VM staff would be able to launch an IoT system that monitors the condition of their remote equipment from any internet-connected mobile phone or computer so as for the company to know the status of their inventory management.
3. Administration application operate on a PC or tablet and allow for a detailed view into the operation of the vending machine, including events, status, and logs, as well as access to the cloud data and analytics. This application report back to the data base of the company that stocking of that machine is lacking or no stockings status or certain percentage volume.
4. Each VM have been given a certain index number to be defined and identified. The reporting status of that particular VM have been deliver to the company database the status of that identified VM. This can be depicted in Figure 1 [3].

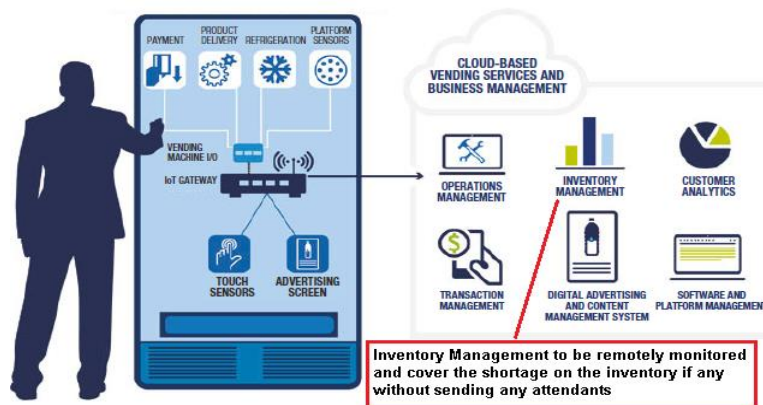


Figure 1: Vending Machine (VM) with IoT operability including inventory management option [3]

2. Methodology

Commonly, the aim of this project is to build a prototype of vending machine which has the capability to remotely monitor the status of inventory stocks and provides a quick response once the status inventory stock alarm alerts the main server of the headquarter of the company that the stock inventory is about to run out. This signal transmitted throughout the cloud by IoT topology to the server to initiate a work plan for the company to refill the inventory shortage.

This chapter entail the mechanism of methodology to design smart Vending Machine (VM) using IoT approach. This include major components of the design process namely: hardware development, simulation & coding development and experimental works, simplify as in Figure 2.

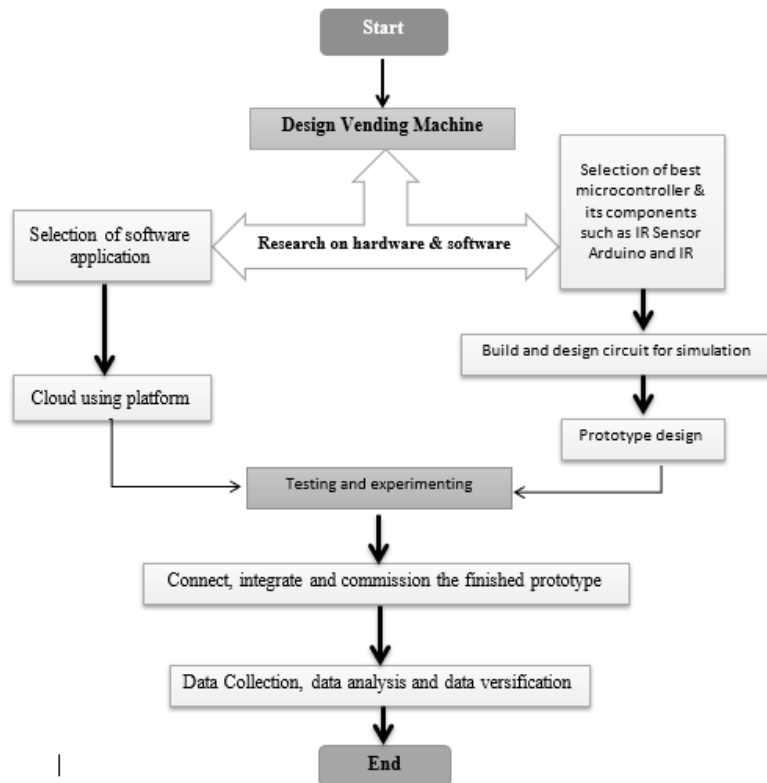


Figure 2: Design process of the proposed project

2.1 NodeMCU ESP8266

The NodeMCU is a development board featuring the popular ESP8266 WiFi chip. As it turns out, ESP8266 could be programed just like any other microcontroller. Its obvious advantage over the Arduino or PIC, that it can readily connect to the Internet via WiFi. However, the ESP8266 breakout board has limited pins although the chip itself has a lot of output ports. The NodeMCU solves this problem by featuring 10 GPIO pins each capable of using PWM, I2C and 1-wire interface.

2.2 ThingSpeak

ThingSpeak is an IoT analytics platform service that allows developers to aggregate, visualize, and analyze live data streams in the cloud. Developers can send data to ThingSpeak from any devices, create instant visualization of live data, and send alerts. Wiki and according to its developers, "ThingSpeak is an open-source Internet of Things (IoT) application and API to store and retrieve data from things using the HTTP and MQTT protocol over the Internet or via a Local Area Network. ThingSpeak enables the creation of sensor logging applications, location tracking applications, and a social network of things with status updates"[8].

2.3 Steps of Developing VM Android Applications

A Vending Machine mobile apps is used to check real-time inventory and also to refill the machine with products, it's developed by an Android app called Sketchware as the following:

Step 1: Sketchware is used to create mobile apps which is called VM apps. First creation of first page which includes Machine 1, Machine 2 and Machine 3, respectively. This is designing the Vending Machine apps first page or home page for VM apps as can be depicted in Figure 3, by divide the first page into 3 sections for 3 machines.

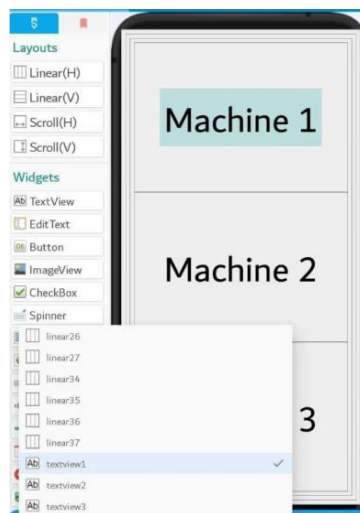


Figure 3: Designing the home page of VM apps

Step 2: Configuring home page by setting the Machine1 as "button1": on click which is ready to open the details of VM1, as shown in Figure 4.



Figure 4: Configuring the first page of the mobile app

Step 3: Creation of each VM page by designing the front view of VM details which has two major sections. The first section contains the gauge, buttons of field 1, 2 and 3 products. The second section is related to refill the Stock Inventory SI as shown in Figure 5.



Figure 5: Designing inventory and refilling page for each Vending Machine

Step 4: Configuring inventory and refilling page of SI by arranging inventories of two sections which are the check status and SI refill as shown in Figure 6.



Figure 6: Configuring inventory and refilling page

2.4 VM Mobile Application

VM Android Mobile App is successfully developed by implementing Sketchware platform as shown in Figure 7. This is a program works on Andriod OS, that intended to prepare products available on every vending machine and additionally used for filling and refilling the storage units of the vending machine.

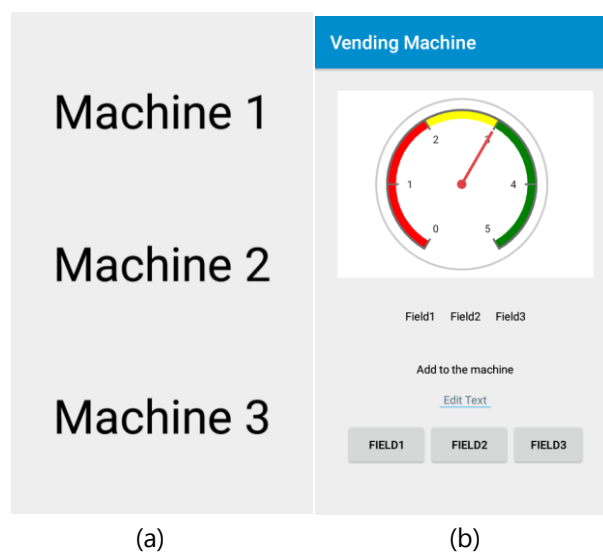


Figure 7: (a) VM app homepage and; (b) VM app gauge for each machine inventory

2.5 System Diagram

The research system diagram can be depicted in Figure 8. The principle operation would be initiated by pressing the pushbutton in which send the electromechanical signal to the NodeMCU ESP8266 to start running. As can be shown in Figure 8, that NodeMCU have been send the signal to selectively either the choice of yes or no. In the case of YES which is represented by $x=zero$ then the functional operation have been dropped. In other words when $x=zero$ it reflects back that there are no more products available and the performance operation become null. In the event that x is not zero which means the state of NO, two simultaneous signals have been sent concurrently. One of these signals have been transmitted towards the DC motor while the other signal have been transmitted towards IR proximity sensor. Both signals can reach the CUTOFF stage in the case of zero then null status have been attained. From the side of IR sensor signal which have been transmitted words IF CUTOFF stage. When the CUTOFF status is YES then the signal proceed back or reflect back towards NodeMCU. On the side of NodeMCU, ThingSpeak been functioning by exchanging the command signals with NodeMCU as input/output status operation while the other side of ThingSpeak, which two signals have been sent towards two Android apps namely: wirepusher and Vending Machine App (VM App). In the case of wirepusher, ThingSpeak have been the sending side while ThingSpeak have been exchange the signals interchangeability with VM App. Wirepusher have displayed the alert status of the products while VM app have displayed the product status with the company headquarter of the products stocks status at the machine whether it's full or out of stock. If full then NodeMCU have been informed and if null then NodeMCU have been instructed to get the data of non-products available at the moment and the products stock need to be refilled. This is bluntly explained in Figure 8.

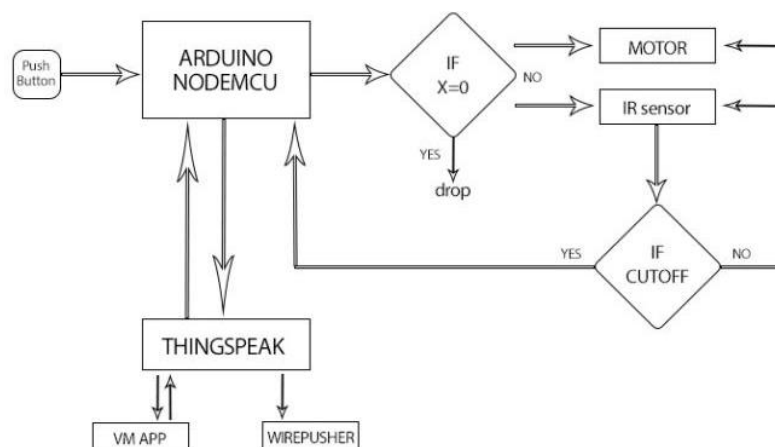


Figure 8: System diagram

2.6 Circuit Diagram

The research circuit diagram can be illustrated in Figure 9. The circuit carries three major line of products. The whole process starts when the button have been pushes representing for example line one. The signals beyond the pushbutton have been transmitted to NodeMCU. The NodeMCU on the other hand have started functioning by sending out the signals to the DC motor by supplying it with 5 Vdc and 0.5 mA. The DC motor have been start operating and turning its bearing to push the selected products throughout the spiral internal screws. The selected product have been move from its containing slot and falls down into the receiving housing. This process have been executed through the IR sensor detection that has been outlined in Figure 9. Once IR sensor receives the signal from NodeMCU simultaneously, DC motor have been push the selected product to position it for the final pick up by the customer. Once the product has been going through the IR sensor, it send a signal to

NodeMCU notifying it that the stock volume is now $x-1$. The research circuit was designed to setup the stock inventory value of the margin of 3. Once the stock value reached 3, then ThingSpeak have been send the signals by exchanging the transactions with wirepusher and VM app that the stock is in critical status of 3. In this critical stage of stock inventory, NodeMCU have been informed that the stock is in low volume margin and need to be refilled and reach the full stage before reaching null stage.

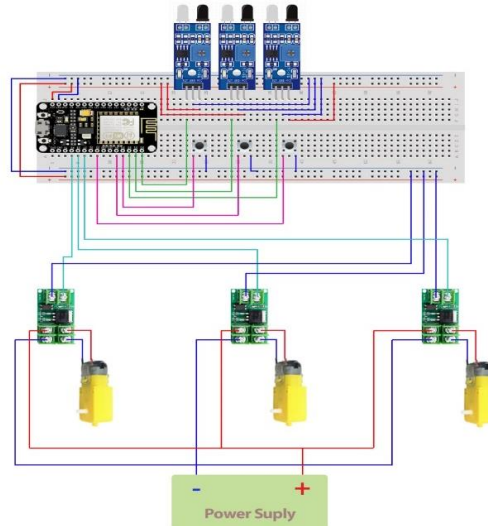


Figure 9: Circuit diagram

3. Results on the Experimental Work

A successful experiments setup is managed and designed completely to ensure the workable of the project, as shown in Figure 10.

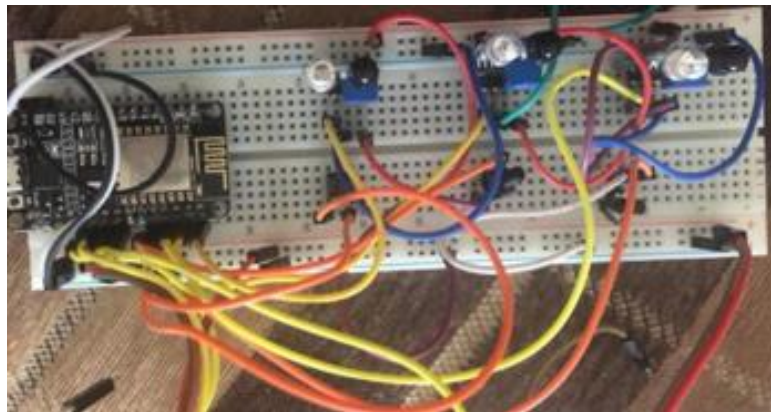


Figure 10: Circuit of proposed system

3.1 Final System Integration for Simulation and Testing

Based on circuit design, the entire system integration, assembly and interface has been structured and formed in which can be seen in the Figure 11.



Figure 11: Prototype

3.2 Wirepusher Android Application

Wirepusher is interconnected through the project with ThingSpeak. The wirepusher function within ThingSpeak environment has the following two major steps:

Step 1: Setup ThingHTTP

Prior to react setup, ThingHTTP is created and the following fields is generated;

- Name: Request 115163
- URL: (wirepusher URL)
- Method: Get
- HTTP version: 1.1

URL is instructed by the commands of wirepusher application, that can be seen in Figure 12 and 13.

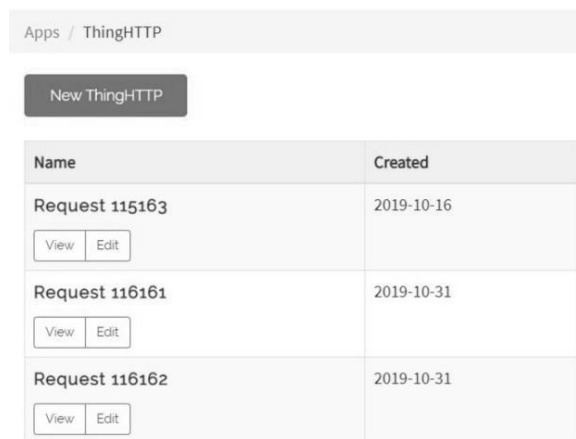


Figure 12: Creating 3 new ThingHTTP for the 3 fields

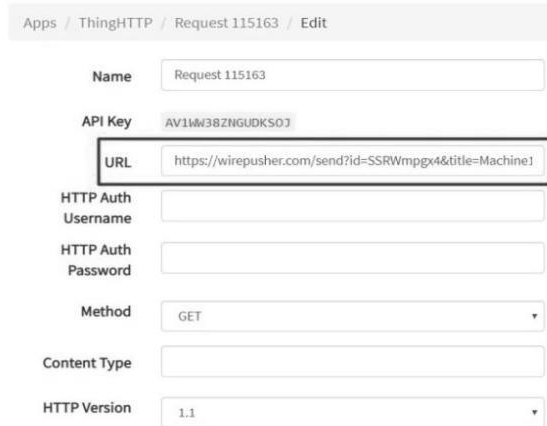


Figure 13: New ThingHTTP setting

Step 2: Setup React

Command called 'react' is created for every field. React 1 for field 1, react 2 for field 2 and react 3 for field 3, as in Figure 14. The setup of new react by plugging in the following information;

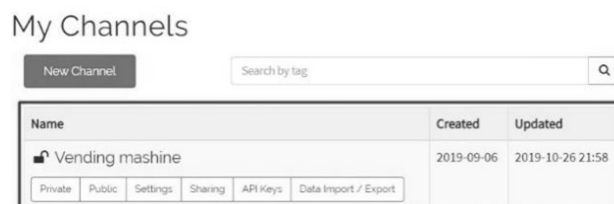
- React Name: React 1
- Condition Type: Numeric
- Test Frequency: On Data insertion
- Condition: Instructed the react to when the field become 4 is created, action by ThingHTTP according to step 1 which is send a notification to the wirepusher mobile application.

Name	Created	Last Ran
<input checked="" type="checkbox"/> React 1 View Edit	2019-10-16	2019-12-08 4:15 pm
<input checked="" type="checkbox"/> React 2 View Edit	2019-10-31	2019-12-10 10:16 am
<input checked="" type="checkbox"/> React 3 View Edit	2019-10-31	2019-12-08 4:14 pm

Figure 14: Creating 3 new react for the 3 fields

3.3 Obtaining the Datum

The data has been obtained from ThingSpeak of the vending machine channel. The channel name is 'Vending Machine' as shown in Figure 15.



Name	Created	Updated
<input checked="" type="checkbox"/> Vending mashine Private Public Settings Sharing API Keys Data Import / Export	2019-09-06	2019-10-26 21:58

Figure 15: ThingSpeak channel

From this channel, the whole data is downloaded from the import/export window. The data is export in the form of csv file extension (.csv) is shown as in Figure 16.

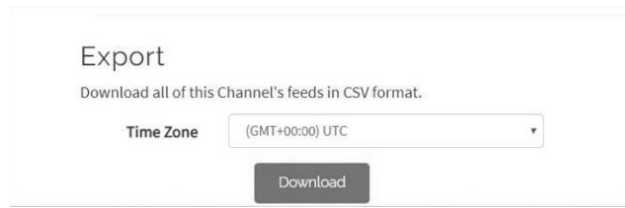


Figure 16: Exporting datasheet from ThingSpeak channel

4. Data Analysis

One of the advantages of the ThingSpeak cloud is the ability to collect data to Microsoft Excel (.CSV) in the form of a table where the table contains several fields, including date, day, year, entry ID and the activity in each of the 3 fields, can be analyzed that data finer shaped curves using solving linear equations.

	A	B	C	D	E
1	created_at	entry_id	field1	field2	field3
2	2019-11-28 16:57:58 +08	1	5		
3	2019-11-28 16:59:53 +08	2		5	
4	2019-11-28 17:00:25 +08	3			5
5	2019-11-28 17:03:00 +08	4	4	5	5
6	2019-11-28 17:03:58 +08	5	3	5	5
7	2019-11-28 17:04:25 +08	6	3	4	5
8	2019-11-28 17:05:00 +08	7	3	4	4
9	2019-11-28 17:05:31 +08	8	3	4	3
10	2019-11-28 17:06:17 +08	9	3	4	2
11	2019-11-28 17:06:49 +08	10	3	3	2
12	2019-11-28 17:07:32 +08	11	3	2	2
13	2019-11-28 17:08:07 +08	12	2	2	2
14	2019-11-28 17:08:45 +08	13	1	2	2
15	2019-11-28 17:09:10 +08	14	0	2	2

Figure 17: Exported data stream from ThingSpeak channel

Figure 17 displays the complete set of data for the 3 lines for 12 days of operation from 28th of November to the 9th of December of 2019. The time counting as shown as in Figure 17, preview the real time transactions for example on 28th of November 2019, that the SI for field 1 was carrying a value of 5 which reflects back to the real time at 8:57:58 +08.

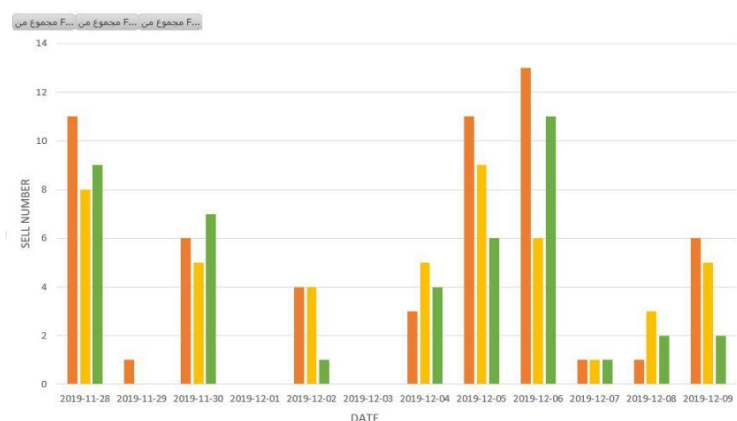


Figure 18: Total sales for 12 days of business operation for line 1, 2 and 3

4. The research involves mechatronics innovations by employing the IR, DC motor and pushbutton in one system environment.
5. The research has implemented low power efficient NodeMCU ESP8266 at 5 Vdc and 0.5 mA respectively.
6. ThingSpeak was in total correlations with VM apps and wirepusher.
7. Schetchwire was efficiently implemeneted connected and integrated to the research system design.
8. The research design circuitry has executed latest NodeMCU ESP8266 version topologies.
9. The research has emphasized the operation of ThingSpeak features.

Finally, the vending machine was developed for the intended purpose. This network had a NodeMCU ESP8266 to notify the vending machine company. This framework was easily set, it required a Wi-Fi connection, and the company had the ability to access the information, which was provided by NodeMCU ESP8266. This model was also used IR proximity sensor to count the products in the machine, DC motors, and buttons. This model helps vending machine companies to be always informed about products quantity inside machines, also to refill before the products get out of stock which is helps with reduce the cost and increases customer trust about the machine, also to analyse the data that collected in ThingSpeak for the sales.

References

- [1] Gwyneth Saldanha, (2015). *Designing IoT Sensor Networks*, Nuvation Engineering, Silicon Valley Headquarters, 151 Gibraltar Court, Sunnyvale, CA 94089 USA.
- [2] Dave McCarthy, (2016). *Vending Machine IoT Makeovers: Retrofit Kit Modernizes Operations and Extends Equipment Life*.
- [3] Antonio Solano, Raquel Dormido, Natividad Duro and Juan Miguel Sánchez, (2016). *Sensors, Article A Self-Provisioning Mechanism in OpenStack for IoT Devices*. 16, 1306.
- [4] Gwyneth Saldanha, (2015). *Designing IoT Sensor Networks*, Nuvation Engineering, Silicon Valley Headquarters, 151 Gibraltar Court, Sunnyvale, CA 94089 USA.
- [5] Muhammad Mohaiminul Islam, Md. Nafees Mahbub, Noor Krystel Fernandez, Azri Bin Azmi and Othman Mohd Yusop, (2016). *Modernizing the Vending Machine Business: Capturing Vending Machine Data on A Remote Server*. Advanced Informatics School (AIS).
- [6] Ashish Kumar Singh, Aditya Sinha, Rajat Vaidya, Hrishikesh Vijay Kulkarni, (2017). *A Review Paper on IoT Based Coffee Vending Machine*, International Journal of Advanced Research in Computer and Communication Engineering (IJARCCE), Vol. 6, Issue 10.
- [7] Aristotle F. Musni, (2014). *Automated Drug Dispenser Using Arduino Microcontroller*, MIT.
- [8] Mark Hung, (2017). *Leading the IoT: Gartner Insights on How to Lead in a Connected World*, Gartner Inc Research.