# **DECLARATION**

We ………………..,, ……………. hereby declare that this Project “IoT MEDICAL PILL FOLLOW UP SYSTEM” submitted in partial fulfillment of the requirement for the Advanced Diploma in ELECTRONICS AND TELECOMMUNICATION TECHNOLY, at Rwanda Polytechnic-Integrated Polytechnic Regional College-Huye, is our original work and has not previously been submitted elsewhere. Also, we declare that a complete list of references is provided indicating all the sources of information quoted or cited.

……………………….

Signature………………………….

Date……. /………/……………….

…………………………………

Signature………………………….

Date……. /………./…………..

# **CERTIFICATION**

This is to certify that the Project work entitled “IoT MEDICAL PILL FOLLOW UP SYSTEM” Submitted by ……………………………. and ………………………, in partial fulfillment of the requirement for the award of the degree Electrical and Electronic engineering department in Electronics and telecommunication technology is a record of work carried out by him under my guidance.

Supervisor:

Signature …………………………...

Date : …... /…... / 2021

Head of department of Electrical and Electronics Engineering

MCS. Theophile BIRAMAHIRE

Signature …………………………...

Date : …... /…... / 2021

# **DEDICATION**

We dedicate this work to our beloved parents, brothers and sisters who may see this work as a fruitful result of their support, encouragement and love. Who have always been by our side in time of need, since we started our academic study till today. Finally, we dedicate it to all our classmates for the part they played in our studies during those long and exciting years.

# **ACKNOWLEDGEMENTS**

Our first gratitude, appreciation and thanks goes to the Almighty God, who guided us through this journey that was not easy.

We express our deepest thanks to Electrical and Electronics Engineering department for taking part in useful decision and giving necessary advices and guidance and arrange all facilities to make life easier. I choose this moment to acknowledge their contribution gratefully.

It is our radiant sentimental to place on record our best regards, deepest sense of gratitude toour supervisor ……………………………… for his careful and precious guidance which were extremely valuable for our study both theoretically and practically, for his technical, wise advice, inspiration, motivation and guidance support during the elaboration of this project.

We highly express our thanks and gratitude for IPRC Huye administration and others staffs for their great help. Moreover, we express and appreciate our school practice supervisor for his help, advice throughout our teaching practice.

We wish to express our sincere thanks to Rwanda Polytechnic/ IPRC Huye for giving us the opportunity of making practice exercises to the department of Electrical and Electronics Technology especially option of Electronics and Telecommunication.

# **ABSTRACT**

Nowadays smart devices are being used in every aspect of our daily life to improve our standards of living, the usage of smart systems has extended in almost every sector including health, payments, transportation and many others in order to improve the standards of. In this report we going to mainly focus on the usage of smart system in health monitoring. Almost 50% of people forget to take their medication at least once a month. Patient adherence to medication regimes is a large problem across the world. Across all medicines, it has been estimated that up to 75% of people do not take their medicines properly and this becomes a problem to people with chronic medication and also elderly people who are likely to forget to take their medication, that why we designed a system that will keep track of the medication schedule and provide a reminder to the patient while also providing the pills in their correct amount, and in case the patient forgets to take their medication, the system must alert the care taker in charge.

**TABLE OF CONTENTS**

[**DECLARATION** ii](#_Toc89941433)

[**CERTIFICATION** iii](#_Toc89941434)

[**DEDICATION** iv](#_Toc89941435)

[**ACKNOWLEDGEMENTS** v](#_Toc89941436)

[**ABSTRACT** vi](#_Toc89941437)

[**TABLE OF FIGURES** ix](#_Toc89941438)

[**LIST OF ABBREVIATIONS** xi](#_Toc89941439)

[**CHAPTER ONE: INTRODUCTION** 1](#_Toc89941440)

[1.1 General background 1](#_Toc89941441)

[1.1.1 Main objective 2](#_Toc89941442)

[1.1.2 Specific objectives 2](#_Toc89941443)

[1.2 Scope of project 2](#_Toc89941444)

[1.3 Project organization 2](#_Toc89941445)

[**CHAPTER TWO: LITERATURE REVIEW** 3](#_Toc89941446)

[2.1 Introduction 3](#_Toc89941447)

[2.2 Electronic components 3](#_Toc89941448)

[2.2.1 Microcontroller 3](#_Toc89941449)

[2.2.1.1 NODEMCU / ESP8266 3](#_Toc89941450)

[2.2.2 Actuators 6](#_Toc89941451)

[2.2.2.1 Servo motor 6](#_Toc89941452)

[2.2.3 Switching devices 7](#_Toc89941453)

[2.2.3.1 Push button 7](#_Toc89941454)

[2.2.4 Indicators 8](#_Toc89941455)

[2.2.4.1 LCD 8](#_Toc89941456)

[2.2.4.2 Buzzer 9](#_Toc89941457)

[**CHAPTER THREE: METHODOLOGY** 10](#_Toc89941458)

[3.1 Introduction 10](#_Toc89941459)

[3.2 Methodology 10](#_Toc89941460)

[3.2.1 Observation 10](#_Toc89941461)

[3.2.2 Documentation 10](#_Toc89941462)

[3.3 Softwares used 11](#_Toc89941463)

[3.3.1 Arduino IDE 11](#_Toc89941464)

[3.3.2 Sublime text editor 11](#_Toc89941465)

[3.3.3 XAMPP 11](#_Toc89941466)

[3.3.4 Browsers 12](#_Toc89941467)

[3.4 Programming languages 12](#_Toc89941468)

[3.4.1 HTML 13](#_Toc89941469)

[3.4.2 CSS 13](#_Toc89941470)

[3.4.3 Java script 13](#_Toc89941471)

[3.4.4 PHP 14](#_Toc89941472)

[3.4.5 SQL 14](#_Toc89941473)

[**CHAPTER FOUR: PRESENTATION AND RESULT ANALYSIS** 15](#_Toc89941474)

[4.1 Introduction 15](#_Toc89941475)

[4.2 Design 15](#_Toc89941476)

[4.2.1 Block diagram 15](#_Toc89941477)

[4.2.2 Block diagram description 16](#_Toc89941478)

[4.2.2.1 ESP8266 16](#_Toc89941479)

[4.2.2.2 Push button 16](#_Toc89941480)

[4.2.2.3 LCD 16](#_Toc89941481)

[4.2.2.4 Buzzer 16](#_Toc89941482)

[4.2.2.5 Servo motor 16](#_Toc89941483)

[4.2.2.6 Power supply 17](#_Toc89941484)

[4.2.2.7 Web application 17](#_Toc89941485)

[4.2.2.8 Device flow chart 24](#_Toc89941486)

[4.2.2.9 Circuit diagram 25](#_Toc89941487)

[4.2.2.10 working principle 25](#_Toc89941488)

[4.2.2.11 Implementation of the project 26](#_Toc89941489)

[**CHAPTER FIVE. CONCLUSION, SUMMARY AND RECOMMENDATION** 27](#_Toc89941490)

[5.1 Conclusion 27](#_Toc89941491)

[5.2 Recommendation 27](#_Toc89941492)

[**REFERENCES** 28](#_Toc89941493)

[**APPENDIES** 29](#_Toc89941494)

# **TABLE OF FIGURES**

[Figure 1: ESP8266 4](#_Toc89941495)

[Figure 2: Servo motor 7](#_Toc89941496)

[Figure 3:LCD 16X2 blue back light 8](#_Toc89941497)

[Figure 4:5VDC piezoelectric buzzer 9](#_Toc89941498)

[Figure 5: popular web browsers 12](#_Toc89941499)

[Figure 6: System block diagram 15](#_Toc89941500)

[Figure 7:Web site organisation 18](#_Toc89941501)

[Figure 8:Index page 19](#_Toc89941502)

[Figure 9: Create account form 20](#_Toc89941503)

[Figure 10: User's dashboard 21](#_Toc89941504)

[Figure 11:User's medication report 21](#_Toc89941505)

[Figure 12:User's profile page 22](#_Toc89941506)

[Figure 13:Admin's dashboard 23](#_Toc89941507)

[Figure 14:Manage users page 23](#_Toc89941508)

[Figure 15: Generate device serial number page 24](#_Toc89941509)

[Figure 16: electronic device flow chart 24](#_Toc89941510)

[Figure 17: Full circuit diagram 25](#_Toc89941511)

[Figure 18: final implemented project 27](#_Toc89941512)

# **LIST OF ABBREVIATIONS**

**AC**: Alternating current

**API**: application programming interface

**ASCII**: American Standard Code for Information Interchange

**AVR:** Automatic Voltage Regulation

**CRT**: Cathode Ray Tube

**CSS**: Cascading style sheet

**dB**: decibel

**DC**: Direct current

**GPRS**: General Packet Radio Service

**GPS**: Global positioning system

**GSM**: Global System for Mobile Communication

**HAART**: Highly active antiretroviral therapy

**HIV**: human immunodeficiency virus

**HTML**: Hypertext markup language

**I2C**: inter-integrated circuit

**IC**: integrated circuit

**ICSP**: in circuit serial programming

**ICT**: Information Communication and Technology

**IDE**: Integrated Development Environment

**IoT**: Internet of things

**IPRC**: Integrated Polytechnic Regional College

**LCD**: Liquid Crystal Display

**LPG**: liquefied petroleum gas

**MHz**: Megahertz

**MS**: Microsoft

**NTC**: Negative temperature coefficient

**PC**: Personal Computer

**PCB**: Printed Circuit Board

**Ppm**: parts per million

**PWM**: Pulse Width Modulation

**RAM**: Random access memory

**RDMS**: Relational Database Management Systems

**ROM**: Read Only Memory

**RP**: Rwanda Polytechnic

**RURA**: Rwanda utilities regulatory authority

**SDA**: Serial data

**SIM**: subscriber identity module

**SMS**: Short Message Service

**SQL**: Structured Query Language

**TTL**: transistor transistor logic

**UART**: Universal asynchronous receiver transmitter

**USA**: United States of America

**USB**: Universal Serial Bus

**VCC**: Voltage common collector

**VDC**: Direct Current voltage

**XAMPP**: Cross-platform, Apache server, MariaDB, PHP and Perl

# **CHAPTER ONE: INTRODUCTION**

## General background

The World Health Organization identified medication non-adherence as one of the major causes of morbidity, mortality and health care costs. It is estimated that between 30 and 50% of prescribed medication, depending on the disease and the health care system, is not taken as directed. (Victor Fornari, 2019)

Numerous researchers put effort into examining the rates and predictors of adherence to contribute to the development of adherence boosting interventions. These studies usually focus on specific medications in particular populations, such as HAART adherence in HIV-positive individuals or blood glucose lowering medications in diabetes patients. The results of these studies cannot be generalized to other populations as well as other drugs. (Institute of Medicine, 2002)

Nonadherence is a large problem but it should not be seen as the patient’s problem as there are many causes of nonadherence but they fall into two overlapping categories: intentional and unintentional. Unintentional nonadherence occurs when the patient wants to follow the agreed treatment but is prevented from doing so by barriers that are beyond their control. Examples include poor recall or comprehension of instructions, difficulties in administering the treatment, or simply forgetting to take it. Unintentional nonadherence is related to limitations in the persons’ capacity and resources affecting their ability to implement their intention to adhere. Intentional nonadherence occurs when the person decides not to follow the treatment recommendations. (Institute of Medicine, 2002)

Having seen all these problems related to nonadherence , we directed this project to solve and minimize the Unintentional nonadherence meaning forgetting to take medications at the right time or taking a few or more than prescribed by the doctor.

### Main objective

The main objectives of this project is to Design and implement IoT MEDICAL FOLLOW UP SYSTEM.

### Specific objectives

The specific objectives of this project are:

1. To build a circuit that will continuously check the time and remind the patient that the time of medication has reached.
2. To build a circuit that will dispense the medicines in the right amount as prescribed by the doctor at the right time.
3. To build a circuit that will send an http request to send an email in case the patient doesn’t collect the medicine.
4. To build web application where the medication collect report will be visualized and also where the times and amount of pills will be set.

## Scope of project

This project is limited to areas that can have an access to internet since the device uses internet to continuously update the times and also send notifications to the health care taker.

## Project organization

This report discusses the design and implementation of IoT medical pills follow up system. It is subdivided into five chapters the first chapter is the Introduction, the second chapter is the literature review, the third chapter is Methodology, the fourth chapter is presentation and result analysis and the fifth chapter is Conclusion, summary and recommendations

# **CHAPTER TWO: LITERATURE REVIEW**

## Introduction

In this chapter we have reviewed different types of components that were used while implementing this project. these components include, Microcontroller, actuators, indicators and switching devices

## Electronic components

### Microcontroller

A microcontroller is a small computer on a single metal-oxide-semiconductor (MOS) integrated circuit (IC) chip. A microcontroller contains one or more CPUs (processor cores) along with memory and programmable input/output peripherals. While designing and implementing this project we used esp8266 (Steiner, 2005)

#### NODEMCU / ESP8266

The esp8266 is the name of microcontroller designed by espressif system , Espressif system is a Chinese company based out of Shanghai . (Dr. Umesh Dutta, 2021)

The esp8266 advertises itself as it is a self-contained Wi-Fi networking solution. It is self as a bridge from existing to Wi-Fi and it is also capable of running self-contained application. The volume of esp8266 didn’t start until 2014 which means in the scheme of things this is a new bland entry in the line-up processors. A couple of years after Ic production, 3rd party OEMs are taking these Ic chips and building “breakout boards” for them. They are very tiny and virtually impossible for hobbyists to attach wires to allow them to be plugged into breadboards Thankfully, these OEMs bulk purchase the ICs, design Basic circuits, design printed circuit boards and construct pre-made boards with the ICs reattached Immediately ready for our use (Dr. Umesh Dutta, 2021)

There are a variety of board styles available. The two that we focused on have been given the names ESP-1 and ESP-12. It is important to note that there is only one ESP8266 processor and it is this processor that is found on ALL breakout boards. What distinguishes one board from another is the number of GPIO pins exposed, the amount of flash memory provided, the style of connector pins and various other considerations related to construction. From a programming perspective, they are all the same

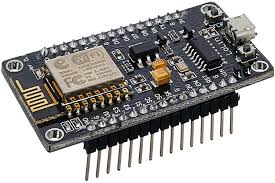


Figure 1: ESP8266

(Dr. Umesh Dutta, 2021)

**The specification**

When we approach a new electronics device, we like to know about its specification. Here bellow are table show us some details points:

Table 1:esp8266 specification

|  |  |
| --- | --- |
| Voltage | 3.3v |
| Current consumption | 10uA – 170Ma |
| Flash memory attachable | 16MB max (512K normal) |
| Processor | Ten silica L106 32 bit |
| Processor speed | 80-160MHz |
| RAM | 32K + 80K |
| GPIOs | 17 (multiplexed with other functions) |
| Analog to Digital | 1 input with 1024 step resolution |
| 802.11 support | b/g/n/d/e/i/k/r |
| Maximum concurrent TCP connections | 5 |

The question of determining how long an ESP8266 can run on batteries is an interesting one. The current consumption is far from constant. When transmitting at full power, it can consume 170mA but when in a deep sleep, it only needs 10uA. That is quite a difference. This means that the run time of an ESP8266 on a fixed current reservoir is not just a function of time but also of what it is doing during that time and that is a function of the program deployed upon it (Kolban, 2016)

The ESP8266 is designed to be used with a partner memory module and this is most commonly flash memory. Most of the modules come with some flash associated with them. Realize that flash has a finite number of erases per page before something fails. They are rated at about 10,000 erases. This is not normally an issue for configuration change writes or daily log writes but if your application is continually writing new data extremely fast, then this may be an issue and your flash memory will fail (Dr. Umesh Dutta, 2021)

This module comes with a built in USB connector and a rich assortment of pin-outs. It is also immediately breadboard friendly

**Applications**

Major Fields of ESP8266 applications to Internet-of-Things include:

• Home Appliances

• Home Automation

• Smart Plug and lights

• Mesh Network

• Industrial Wireless Control

• Baby Monitors

• IP Cameras

• Sensor Networks

• Wearable Electronics

### Actuators

An actuator is a component of a machine that is responsible for moving and controlling a mechanism or system, for example by opening a valve. An actuator requires a control signal and a source of energy. The control signal is relatively low energy and may be electric voltage or current, pneumatic, or hydraulic fluid pressure, or even human power. Its main energy source may be an electric current, hydraulic pressure, or pneumatic pressure. When it receives a control signal, an actuator responds by converting the source's energy into mechanical motion, while implementing this project we used a servo motor (Nesbitt, 2011)

#### Servo motor

A servo motor is a rotary actuator or motor that allows for a precise control in terms of angular position, acceleration and velocity, capabilities that a regular motor does not have. It makes use of a regular motor and pairs it with a sensor for position feedback. The controller is the most sophisticated part of the servo motor, as it is specifically designed for the purpose.

Servo motors are not actually a specific class of motor but are a combination of specific parts, which happen to include a DC or AC motor, and are suitable for use in a closed-loop control system. They are used in robotics, automated manufacturing and computer numerical control (CNC) machining applications (Firoozian, 2014) .

The servo motor is a closed-loop servomechanism that uses position feedback in order to control its rotational speed and position. The control signal is the input, either analog or digital, which represents the final position command for the shaft. A type of encoder serves as a sensor, providing speed and position feedback. In most cases, only the position is reported. The final position is reported to the controller and this is compared to the initial position input, and then if there is a discrepancy, the motor is moved in order to get to the correct position (Firoozian, 2014).

The simplest servo motors use DC motors and position sensing through a potentiometer and also use big-bang control, which means that the motor moves at maximum speed until it stops at the designated position or is stopped. This is not widely used in industrial motion control as it can be quite inaccurate, but these kinds of servo motors are popular in radio-controlled devices such as model aircraft and toy cars. Sophisticated servo motors for industrial use have both position and speed sensing as well as implement proportional-integral-derivative control algorithms, allowing the motor to be brought to its position quickly and precisely without overshooting, as the speed of the shaft can also be controlled (Firoozian, 2014).



Figure 2: Servo motor

(Firoozian, 2014)

### Switching devices

switching devices are defined as any devices that open and close electrical circuits. Electrical circuits must form a continuous loop, and a switching device functions like a gate in that loop. A circuit is ON when the switching device is closed, and the circuit is OFF when the switching device is open. In this project we used different kind of switching devices such as pushbuttons, relay board and selector switch (J. Paul Guyer, 2017)

#### Push button

A Pushbutton Switch is a switch designed so that its contacts are opened and closed by depressing and releasing a pushbutton on the Switch in the direction of its axis. Push buttons are available in different structure and operation. Momentary operation here the pushbutton returns to its original position after it is released and in alternate operation the first time the pushbutton is pressed an internal lock mechanism holds it in the same position. The next time it is pressed, the lock is released and the pushbutton returns to its original position (J. Paul Guyer, 2017).

### Indicators

An indicator is a specific, observable and measurable characteristic that can be used to show changes or progress a program is making toward achieving a specific outcome. In this project we used different types of indicators such as light indicators i.e. LEDS, LCD and sound indicators i.e. buzzer (Murphy, 2014)

#### LCD

An LCD (Liquid Crystal Display) screen is an electronic display module and has a wide range of applications. A 16x2 LCD display is very basic module and is very commonly used in various devices and circuits. A 16x2 LCD means it can display 16 characters per line and there are 2 such lines. In this LCD each character is displayed in 5x7 pixel matrix. The 16 x 2 intelligent alphanumeric dot matrix display is capable of displaying 224 different characters and symbols. This LCD has two registers, namely, Command and Data (Murphy, 2014).

Command register stores various commands given to the display. Data register stores data to be displayed. The process of controlling the display involves putting the data that form the image of what you want to display into the data registers, then putting instructions in the instruction register.

The LCD used parallel communication and one of the disadvantages of parallel communication is that it uses many wires for information transfer, to solve this problem we used an I2C module. I2C is a serial communication protocol, so data is transferred bit by bit along a single wire (the SDA line). I2C is synchronous, so the output of bits is synchronized to the sampling of bits by a clock signal shared between the master and the slave. The clock signal is always controlled by the master (Murphy, 2014)

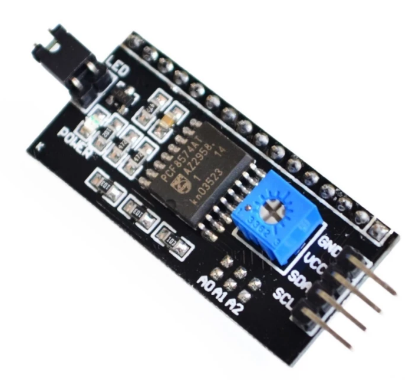


Figure 3:LCD 16X2 blue back light

[ McRoberts.(2011)]

#### Buzzer

The buzzer is a sounding device that can convert audio signals into sound signals. It is usually powered by DC voltage. It is widely used in alarms, computers, printers and other electronic products as sound devices. It is mainly divided into piezoelectric buzzer and electromagnetic buzzer. While designing this project we used a 5VDC piezo electric buzzer (Murphy, 2014)



Figure 4:5VDC piezoelectric buzzer

(Murphy, 2014)

# **CHAPTER THREE: METHODOLOGY**

## 3.1 Introduction

This chapter demonstrates different methods and strategies that were used to conduct this project, we highlighted the methods which are observation method and documentation methods. Furthermore, we reviewed different softwares and programming languages that were also used while designing and implementing this project.

## Methodology

### Observation

Observation, as the name implies, is a way of collecting data through observing. Observation data collection method is classified as a participatory study, because the researcher has to immerse him/herself in the setting where her respondents are, while taking notes and/or recording. The advantages of observation data collection method include direct access to research phenomena, high levels of flexibility in terms of application and generating a permanent record of phenomena to be referred to later. At the same time, observation method is disadvantaged with longer time requirements, high levels of observer bias, and impact of observer on primary data, in a way that presence of observer may influence the behavior of sample group elements.

While collecting the data, it was hard to visit many homes since most of them they like privacy. But we managed to observe starting from our homes and some institutions, we also observed how the business is done on the side of vendors in the business of gas refilling and gas ordering.

### Documentation

This technique is also necessary for getting information and help the researcher also to increase their knowledge so that he can easily solve the problem practically, it permits the researcher to consult books, other previous researchers, class notes, and the internet to find a different definition of words and codes for solving it.

Here we have read many class notes; consult different books from the school even outside the school so that we can easily solve the problems as stated before.

## Softwares used

Application software (app for short) is a program or group of programs designed for end-users to perform specific tasks. While designing and implementing this project we used different kind of softwares like Arduino IDE, Sublime text editor, Xampp and browsers (Rajlich, 2011).

### Arduino IDE

The Arduino Integrated Development Environment (IDE) is a cross-platform application (for Windows, mac OS, Linux) that is written in functions from C and C++.It is used to write and upload programs to Arduino compatible boards, but also, with the help of third-party cores, other vendor development boards. In this project Arduino IDE was used in writing and upload program to Arduino mega 2560 (Murphy, 2014)

### Sublime text editor

Sublime Text is a sophisticated  cross-platform source code editor with a Python application programming interface (API). It natively supports many programming languages and markup languages, and functions can be added by users with plugins, typically community-built and maintained under free-software licenses. This software was used in writing PHP, JavaScript, HTML and CSS codes (McFedries, 2018)

### XAMPP

The full form of XAMPP is X stands for Cross-platform, (A) Apache server, (M) MariaDB, (P) PHP and (P) Perl. The Cross-platform usually means that it can run on any computer with any operating system. XAMPP is an open-source software developed by Apache Friends. XAMPP software package contains Apache distributions for Apache server, MariaDB, PHP, and Perl. And it is basically localhost or a local server. This local server works on your own desktop or laptop computer. XAMPP was used test the website before uploading it to the remote web server. XAMPP server software gave us a suitable environment for testing MYSQL, PHP, Apache, and Perl projects on the local computer (McFedries, 2018)

### Browsers

A web browser, or simply "browser," is an application used to access and view websites. Common web browsers include Microsoft Internet Explorer, Google Chrome, Mozilla Firefox, and Apple Safari. The primary function of a web browser is to render HTML, the code used to design or " markup " webpages. Each time a browser loads a web page, it processes the HTML, which may include text, links, and references to images and other items, such as cascading style sheets and JavaScript functions. The browser processes these items, then renders them in the browser window. While designing the web application we used many popular web browsers in order ensure browser compatibility (McFedries, 2018).



Figure 5: popular web browsers

[ Luimes.A.(2018)]

## Programming languages

A programming language is a set of commands, instructions, and other syntax use to create a software program. Languages that programmers use to write code are called "high-level languages." This code can be compiled into a "low-level language," which is recognized directly by the computer hardware (McFedries, 2018)

High-level languages are designed to be easy to read and understand. This allows programmers to write source code in a natural fashion, using logical words and symbols. In the design of this project different programming languages were which are Arduino C language, HTML, CSS, PHP, Java script (McFedries, 2018)

### HTML

Stands for "Hypertext Markup Language." HTML is the language used to create webpages. HTML was invented by Tim Berners-Lee, a physicist at the CERN research institute in Switzerland. He came up with the idea of an Internet-based hypertext system.

Hypertext means a text that contains references (links) to other texts that viewers can access immediately. He published the first version of HTML in 1991, consisting of 18 HTML tags. Since then, each new version of the HTML language came with new tags and attributes (tag modifiers) to the markup (McFedries, 2018)

### CSS

Cascading Style Sheets, fondly referred to as CSS, is a simple design language intended to simplify the process of making web pages presentable.

CSS handles the look and feel part of a web page. Using CSS, you can control the color of the text, the style of fonts, the spacing between paragraphs, how columns are sized and laid out, what background images or colors are used, layout designs, variations in display for different devices and screen sizes as well as a variety of other effects.

CSS is easy to learn and understand but it provides powerful control over the presentation of an HTML document. Most commonly, CSS is combined with the markup languages HTML. CSS has several advantages among them include CSS saves time, Pages load faster, Easy maintenance, Superior styles to HTML, Multiple Device Compatibility, Global web standards (McFedries, 2018)

### Java script

JavaScript is a dynamic computer programming language. It is lightweight and most commonly used as a part of web pages, whose implementations allow client-side script to interact with the user and make dynamic pages. It is an interpreted programming language with object-oriented capabilities.

Java script was used to create and control dynamic website content, i.e. anything that moves, refreshes, or otherwise changes on your screen without requiring you to manually reload a web page. Features like: animated graphics, photo slide shows, autocomplete text suggestions, interactive forms (McFedries, 2018)

### PHP

PHP is a server side scripting language. that is used to develop Static websites or Dynamic websites or Web applications. PHP stands for Hypertext Pre-processor, that earlier stood for Personal Home Pages. PHP scripts can only be interpreted on a server that has PHP installed. PHP is integrated with a number of popular databases, including MySQL, PostgreSQL, Oracle, Sybase, Informix, and Microsoft SQL Server, PHP was used in making logical decisions and connecting the interface with the database through SQL (McFedries, 2018).

### SQL

SQL is Structured Query Language, which is a computer language for storing, manipulating and retrieving data stored in a relational database.

SQL is the standard language for Relational Database System. All the Relational Database Management Systems (RDMS) like MySQL, MS Access, Oracle, Sybase, Informix, Postgres and SQL Server use SQL as their standard database language. SQL was used to access data in the relational database management systems, define the data in a database and manipulate that data, create and drop databases and tables, create view, stored procedure, functions in a database (McFedries, 2018).

# **CHAPTER FOUR: PRESENTATION AND RESULT ANALYSIS**

## Introduction

Using the components, softwares and programming languages discussed before an IoT device and a web application was designed and implemented.

## Design

The design of this project Medical follow up system and pill dispenser is divided into two parts one being the design of the electronic device and another part being the design of the web application.

#### Block diagram

ESP8266

Power supply

Pushbutton

LCD

Cloud

web application

Buzzer

Servo motor

Figure 6: System block diagram

### Block diagram description

The basic block diagram of IoT pills dispenser as shown above is composed of different main blocks:

#### ESP8266

The ESP8266 act as the brain of the electronic device as well as the gateway to the web application, as discussed it is a microcontroller that has the capability to connect to the Wi-Fi network making it as node in a network with the capability to send http request and bring back the response, it also makes logical decisions based on the input devices and the response from the cloud to trigger and print on the output devices.

#### Push button

A Pushbutton Switch is a switch designed so that its contacts are opened and closed by depressing and releasing a pushbutton on the Switch in the direction of its axis. A pushbutton was used to provide a command signal to the microcontroller to dispense the pills when the time for medication is reached.

#### LCD

An LCD (Liquid Crystal Display) screen is an electronic display module and has a wide range of applications. In the design of IoT medical pills dispenser the LCD is used in displaying information to the user of the device, the information displayed include the next medication time, and a message telling the user to collect the pills

#### Buzzer

As said previously a buzzer is a sounding device that can convert audio signals into sound signals. This sound could contain information to the user depending on the frequency tone or the level of volume of the sound produced. The buzzer in this project continuously provide a beeping sound when the time to take the pills has reached.

#### Servo motor

A servo motor as said is a rotary actuator or motor that allows for a precise control in terms of angular position, acceleration and velocity, while designing this project, the servo motor was used to provide a mechanism to dispense the amount of pills needed by controlling its motion and angle of rotation.

#### Power supply

For the circuit to work we needed a source of DC energy, in which we got from batteries and through regulation we obtain a voltage level of voltage as we needed using positive voltage regulators. The value of voltage needed was 6VDC.

#### Web application

A web-based application is a software packages that can be accessed through the web browser over a network such as intranet or internet. The software and database reside on a central server rather than being installed on the desktop system and is accessed over a network. Web based applications are the ultimate way to take advantage of today’s technology to enhance your organizations productivity and efficiency.

In order to achieve the medical pill dispenser and follow up system we designed a web application and bellow is the web organization of the system

**Web organization**

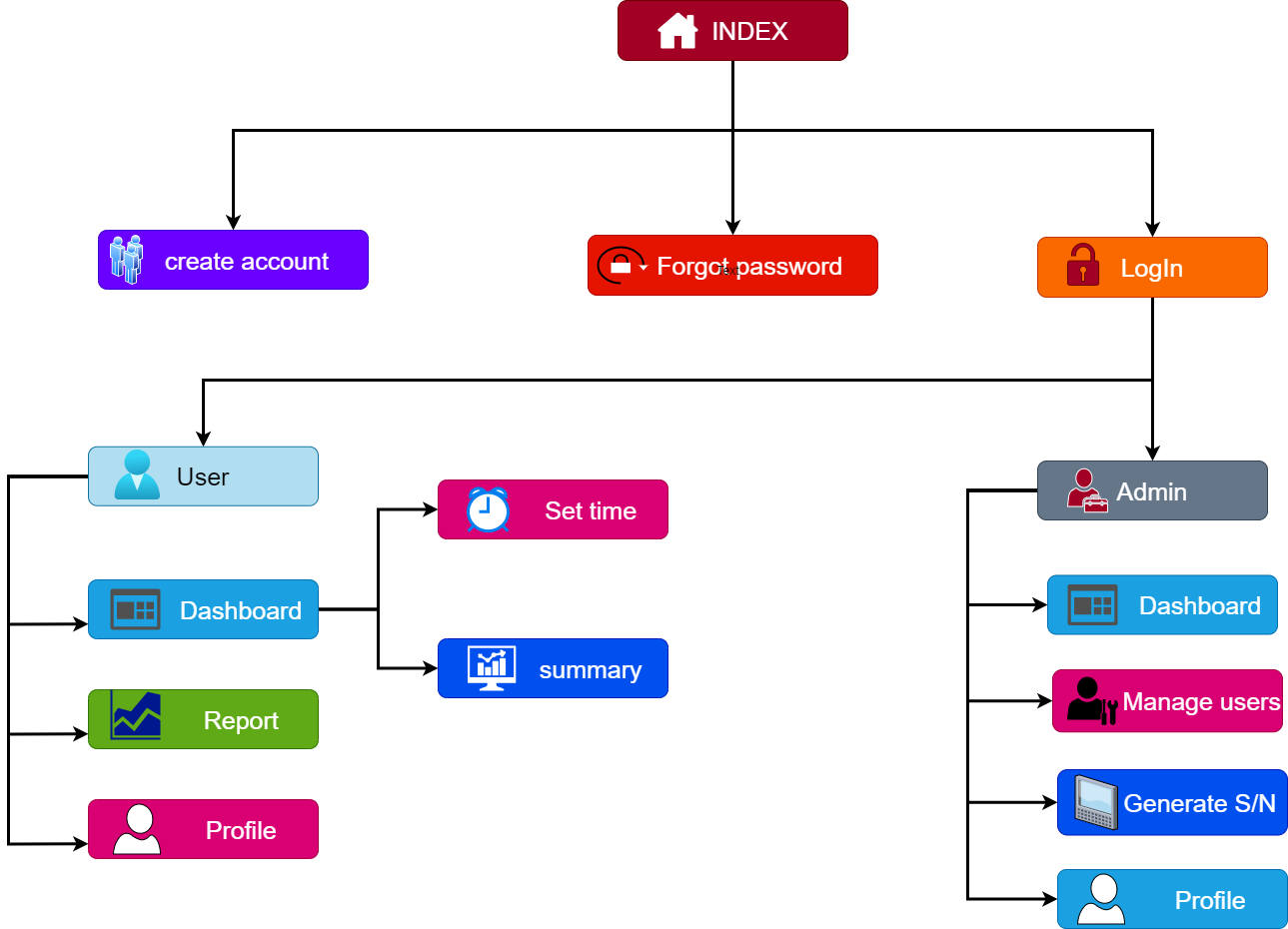


Figure 7:Web site organisation

Upon opening the web application, the user lands on the index page where the user can login in his/her account, create an account if the user is new to our system and also a way to reset the password when the user forgets the password.

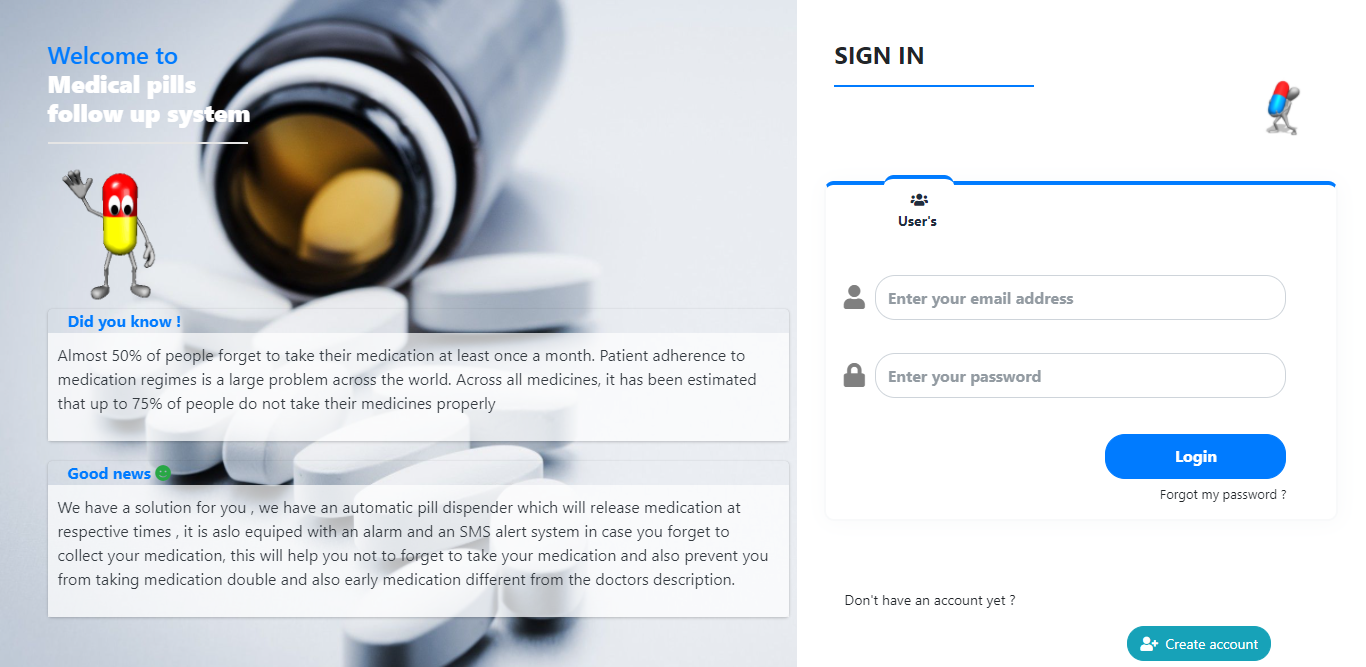


Figure 8:Index page

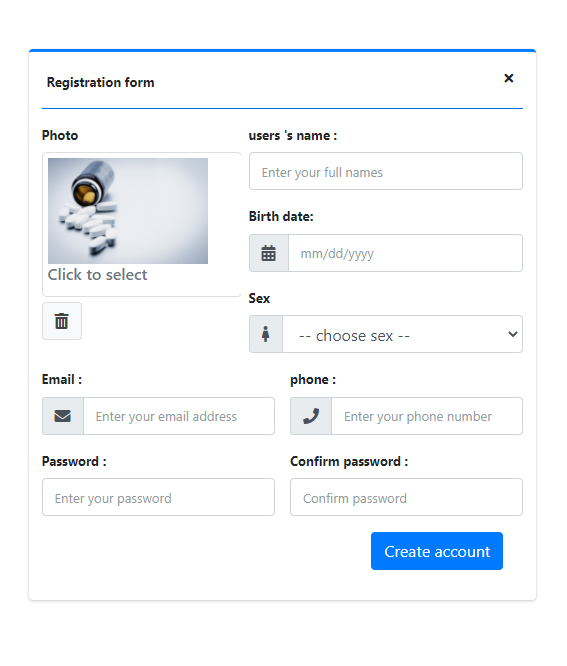


Figure 9: Create account form

After the user has logged in the system, the user lands on the dashboard page, where he/she can set the medication time and number of pills to take, view the summary of the set times, missed pills and taken pills by numeric and by a graph.

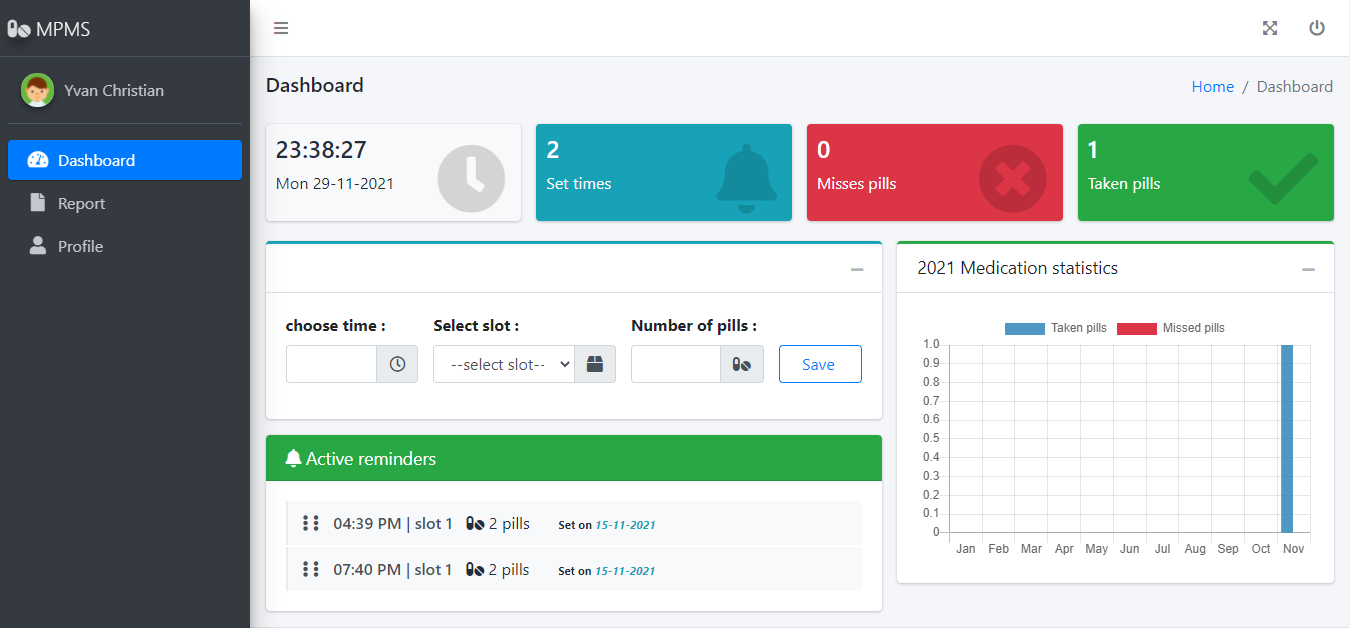


Figure 10: User's dashboard

In the system the user can also view the report of the taken and missed medication by clicking on the report link in the web application. The user also has some features like printing the report, turning it into an excel sheet and also into PDF.

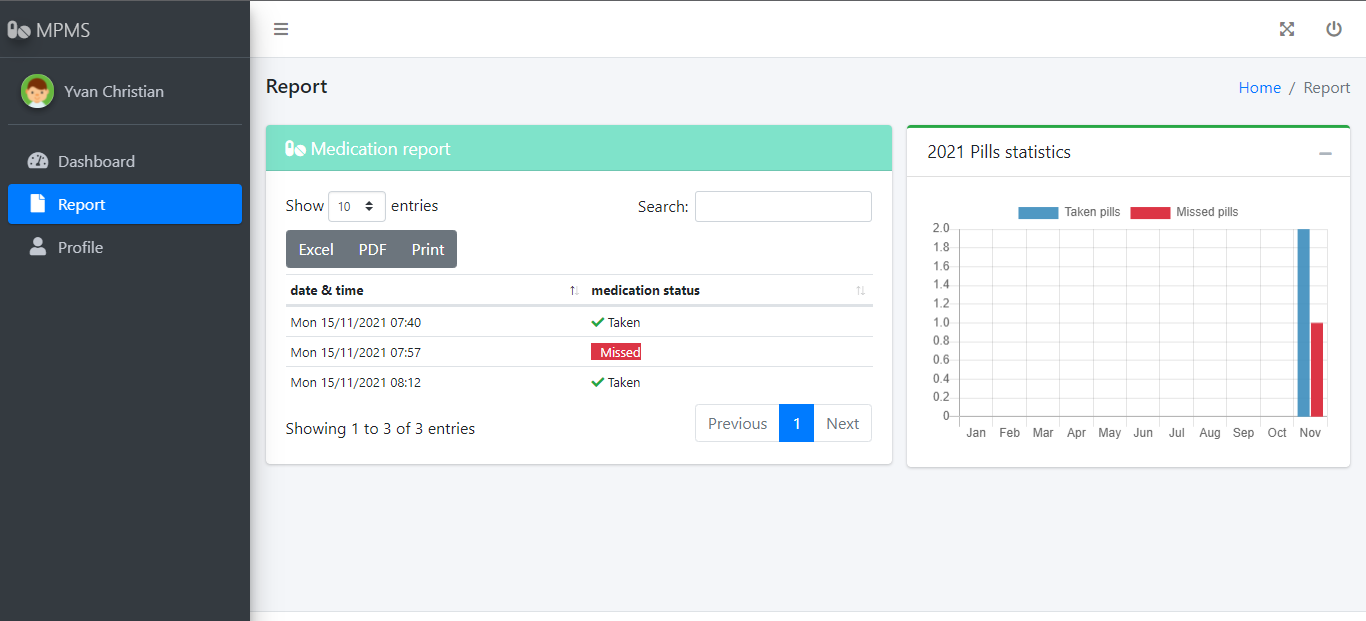


Figure 11:User's medication report

Lastly on the users account, we have another page called the profile page where the user can edit his/her username, email, password, phone number and birthdate

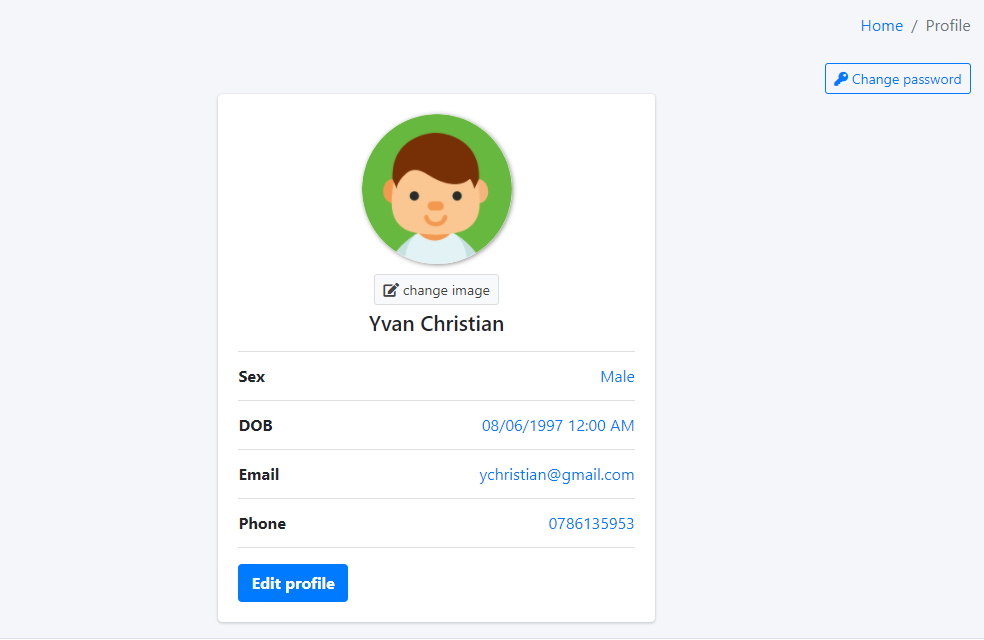


Figure 12:User's profile page

On the other hand the administrator of the system also has to login through the same login as the user, after logging in the system the administrator lands on the dashboard, that provides an overall summary of the registered users, missed medications, taken medication, the summary is represented in both numbers and by graphs.

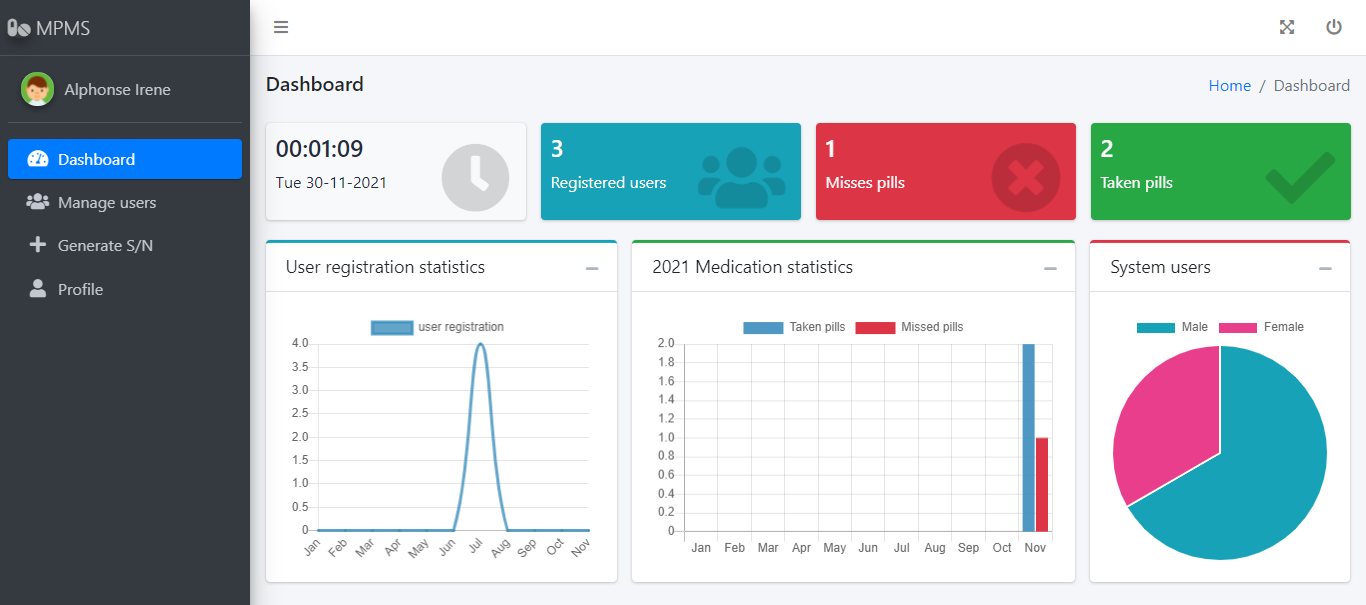


Figure 13:Admin's dashboard

We also have another tab in the system where the administrator can manage users in the system

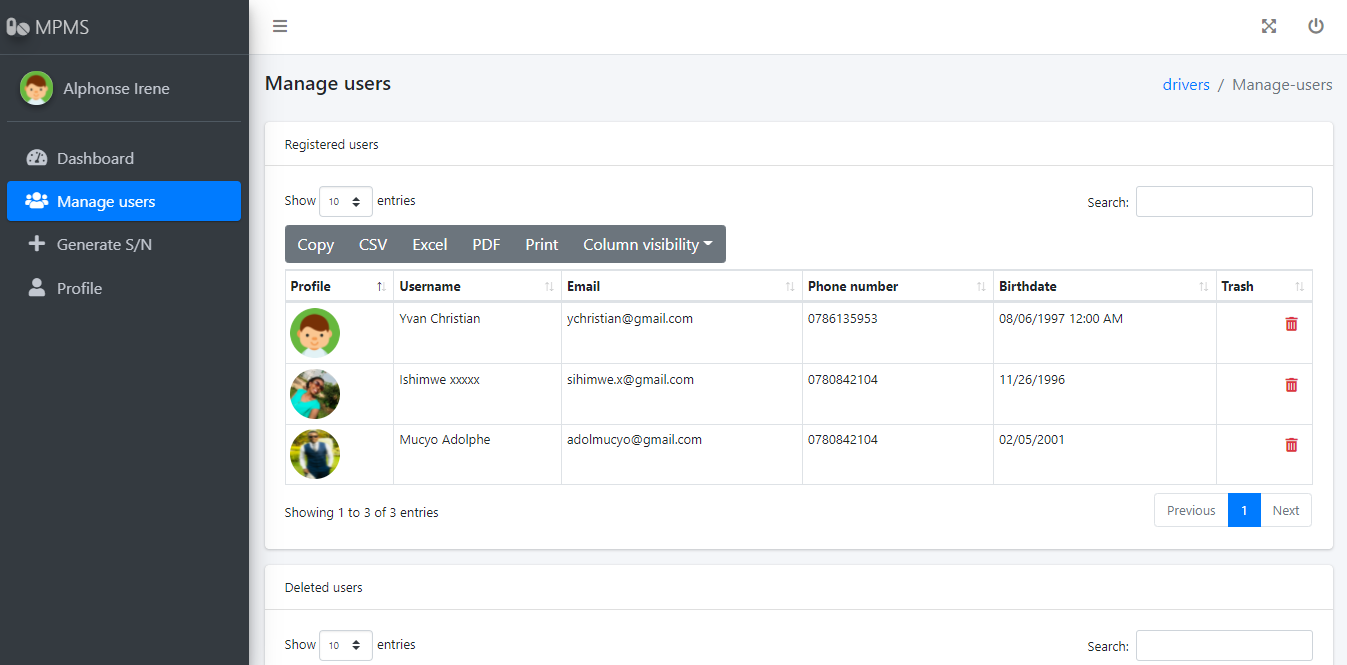


Figure 14:Manage users page

Since every user in the system must have a unique device, with a unique serial number to prevent errors that can be made in serial number generation, the system has a way to generate serial number that are unique.

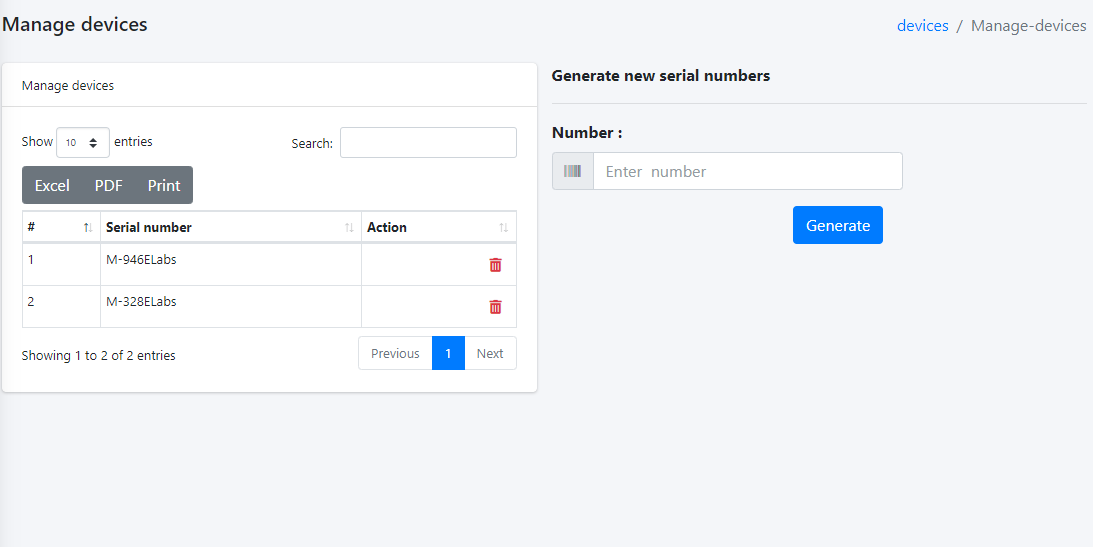


Figure 15: Generate device serial number page

#### Device flow chart

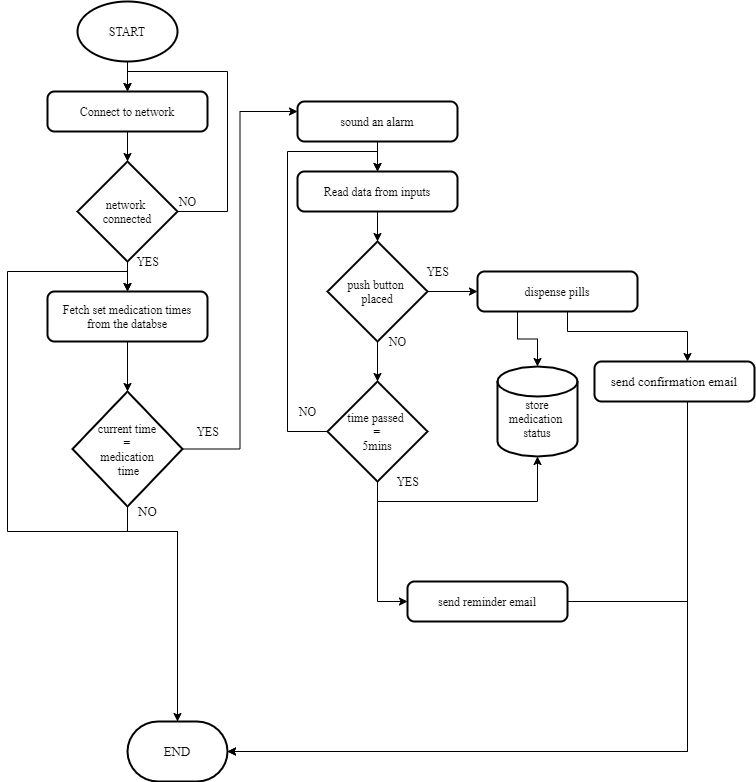


Figure 16: electronic device flow chart

#### Circuit diagram

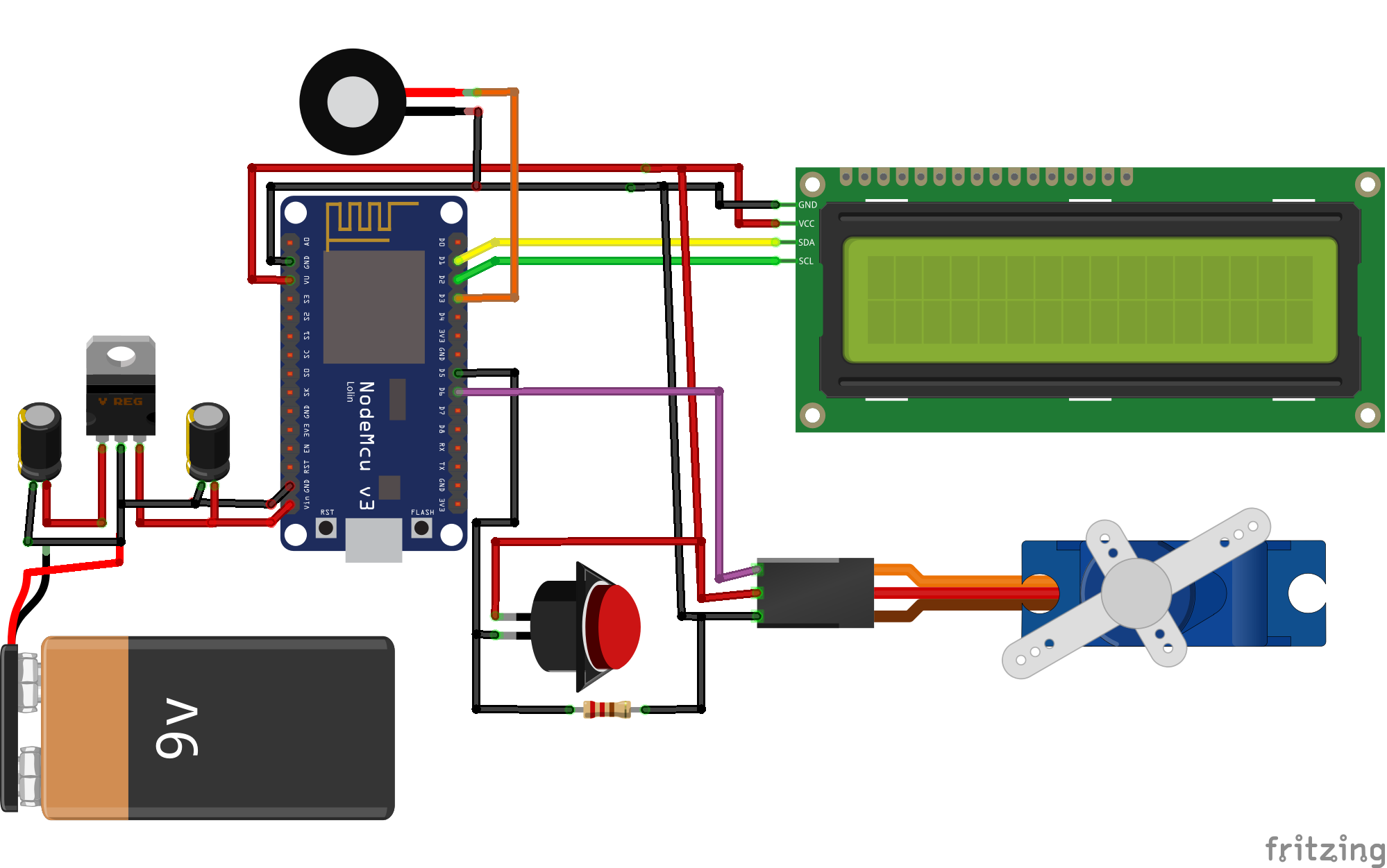


Figure 17: Full circuit diagram

#### working principle

Our project is mainly composed of NodeMcu as the microcontroller used, servo motor for pill dispenser mechanism, LCD to display the information to user, buzzer to provide a sound notification, power supply system to power up the device.

When the device is powered on the device starts by connecting to a wireless network since the ESSP8266 uses Wifi network, when the device has connected to the network, it makes an HTTP request to the webserver, returning the response that contains the set times in the web application and the number of pills that are set.

When the set time matches the current time, the microcontroller then commands the buzzer to make a sound notification alerting the user , that the medication time has reached and also a message will be displayed on the LCD informing the user to press the pushbutton to take the medication, When the user is available and pushes on the pushbutton the microcontroller then instructs the servo motor onto which the pill dispense mechanism is attached to turn at an angle of 30 degrees achieving a mechanism that dispenses one pill at a time until I reaches the correct amount set in the web application.

When the user is not available, means when he/she doesn’t collect the medication, the system will send an email informing him/her that the medication was not taken and also the time and date the medication was supposed to be taken.

Not depending on whether the user collects the medication or not , the system will keep the records for the taken and medication data, thus allowing users to view their medication statistics through the web application.

#### Implementation of the project

While implementing this project, we started by testing component on breadboard then after testing we soldered the whole circuit on PCB after all we covered the project in a wooden cover. Below is the final implemented project.

Figure 18: final implemented project

# **CHAPTER FIVE. CONCLUSION, SUMMARY AND RECOMMENDATION**

## Conclusion

The project that we have undertaken has helped us gain a better perspective on various aspects related to our course of study as well as practical knowledge on electronic components, we became familiar with software analysis, designing, implementing, testing and maintaining while carrying out this project, we also improved on team working which of course became the pillar to successfully achieving this project. From the convenience of IoT system that incorporates ICT we managed to make a system the and the device that can solve the problem of medical adherence especially in elder people who mostly have a problem in remembering the medication time and the number of pills to take.

## Recommendation

1. This project is an implication of our concept in automating, controlling of internet of things system, the practical application of this project are immerse and can have a vast level of implementation, this project is highly recommended to be for patients especially those at an elderly age.
2. I recommend my fellow brothers and sisters to explore more in the field of IoT since there is more to learn and this world is tending to prefer IOT systems more.
3. This system can be improved by designing a mobile application hence improving the reliability and easy and fast access.
4. I would like to recommend our institution IPRC-HUYE to provide more practical knowledge in the field of IoT so that it can equip students the knowledge needed in this field.

# **REFERENCES**

Dr. Umesh Dutta, N. K. (2021). *The Internet of Things Using NODEMCU.* mumbai: Blue Rose Publishers, 2021.

Firoozian, R. (2014). *Servo Motors and Industrial Control Theory.* Springer, 2014.

Institute of Medicine, B. o. (2002). *Care Without Coverage: Too Little, Too Late.* Washington, D.C., United States: National Academies Press.

J. Paul Guyer, P. R. (2017). *An Introduction to Protective and Switching Devices for Electrical Distribution.* Guyer Partners, 2017.

Kolban, N. (2016). kolban 's book. In N. kolban, *kolban 's book.* Texas, USA: internet.

McFedries, P. (2018). *Web Coding & Development All-in-One For Dummies.* John Wiley & Sons, 2018.

Murphy, M. (2014). *High-Tech DIY Projects with Electronics, Sensors, and LEDs.* The Rosen Publishing Group, Inc, 2014.

Nesbitt, B. (2011). *Handbook of Valves and Actuators: Valves Manual International.* Elsevier, 2011.

Rajlich, V. (2011). *Software Engineering: The Current Practice.* mumbai: CRC Press, 2011.

Steffi O. Muhanji, A. E. (2019). EIoT: The Development of the Energy Internet of Things in Energy Infrastructure. Springer.

Steiner, C. (2005). *The 8051/8052 Microcontroller: Architecture, Assembly Language, and Hardware Interfacing.* Newyork: Universal-Publishers, 2005.

Victor Fornari, I. D. (2019). *Psychiatric Nonadherence: A Solutions-Based Approach.* Berlin/Heidelberg: Springer.

# **APPENDIES**

APPENDIX ONE

Project codes

#include <SoftwareSerial.h>

#include <ESP8266HTTPClient.h>

#include <ESP8266WiFi.h>

#include <ESP8266WebServer.h>

#include <ArduinoJson.h>

#include <Servo.h>

#include <Wire.h>

#include <LiquidCrystal\_I2C.h>

LiquidCrystal\_I2C lcd(0x27, 20, 4);

Servo myservo;

// WiFi parameters

const char\* ssid = "admin";// network name

const char\* password = "admin123";// network password

HTTPClient http;

//host to send data

String serial\_number = "M-946ELabs";

const char\* host = "http://medical.enlabs.rw/test.php";

int push\_button = D5;

String last\_pill\_time = "";

int taken = 0;

unsigned int miss\_millis = 0;

int missed\_pills = 0;

String missed\_time = "";

int msg\_type = 0;

unsigned int last\_millis\_s = 0;

String next\_m = " Not set";

int buzzer = D7;

void setup() {

Serial.begin(9600);

myservo.attach(D6);

myservo.write(30);

lcd.init(); // initialize the lcd

lcd.init();

// Print a message to the LCD.

lcd.backlight();

lcd.setCursor(3, 0);

lcd.print("Powering on");

delay(3000);

lcd.clear();

lcd.setCursor(0, 0);

lcd.print("Connecting to ");

lcd.setCursor(0, 1);

lcd.print("network... ");

pinMode(push\_button, INPUT);

pinMode(buzzer, OUTPUT);

Serial.print("Connecting to ");

Serial.println(ssid);

WiFi.begin(ssid, password);

while (WiFi.status() != WL\_CONNECTED) {

delay(500);

Serial.print(".");

}

delay(3000);

lcd.clear();

lcd.setCursor(0, 0);

lcd.print("WiFi ");

lcd.setCursor(0, 1);

lcd.print("Connnected ");

Serial.println("WiFi connected");

Serial.println("IP address: ");

Serial.println(WiFi.localIP());

delay(2000);

lcd.clear();

}

void loop() {

unsigned int current\_milli\_s = millis();

if (current\_milli\_s - last\_millis\_s >= 2000) {

lcd.clear();

lcd.setCursor(0, 0);

lcd.print("Next medication");

lcd.setCursor(0, 1);

lcd.print("at : ");

lcd.print(next\_m);

last\_millis\_s = millis();

}

fetch\_data();

}

void fetch\_data() {

String url = host;

url += "?d=" + serial\_number;

http.begin(url);

http.addHeader("Content-Type", "text/plain");

int httpCode = http.GET();

String payload = http.getString(); // get data from webhost continously

Serial.println(payload);

String input = payload;

StaticJsonDocument<200> doc;

DeserializationError err = deserializeJson(doc, input) ;

if (err) {

Serial.print("ERROR:");

Serial.print(err.c\_str());

return;

}

int pill = doc["pill"]; // 0

int times = doc["times"]; // 0

String next = doc["next"]; // "05:33 PM"

String time\_c = doc["time"]; // "05:33 PM"

int qty = doc["qty"]; // 1

if (next.length() > 0) {

next\_m = next;

}

else {

next\_m = " Not set";

}

if (last\_pill\_time != time\_c) {

taken=0;

}

unsigned int last\_millis = millis();

if (pill == 1) {

while (1) {

unsigned int current\_milli\_s = millis();

if (taken == 0) {

digitalWrite(buzzer, HIGH);

delay(250);

digitalWrite(buzzer, LOW);

delay(250);

}

if (current\_milli\_s - last\_millis\_s >= 2000) {

if (taken == 0) {

lcd.clear();

lcd.setCursor(0, 0);

lcd.print("Press to take ");

lcd.setCursor(0, 1);

lcd.print(times);

lcd.print(" pills ");

}

else {

lcd.clear();

lcd.setCursor(0, 0);

lcd.print("Next medication");

lcd.setCursor(0, 1);

lcd.print("at : ");

lcd.print(next\_m);

}

last\_millis\_s = millis();

}

………………………………..

APPENDIX TWO

PROJECT BUDGET

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **#** | **ITEM** | **QTY** | **PRICE/UNITY** | **TOTAL PRICE** |
| 1 | ARDUINO | 1 | 25,000 | 25,000 |
| 2 | LOAD CELL | 4 | 6,500 | 26,000 |
| 3 | GAS SENSOR (MQ5) | 1 | 10,000 | 10,000 |
| 4 | DTH11 | 1 | 10,000 | 10,000 |
| 5 | GPS SENSOR | 1 | 14,500 | 14,500 |
| 6 | GSM | 1 | 17,500 | 17,500 |
| 7 | ELECTRO-VALVE | 1 | 14,000 | 14,000 |
| 8 | LCD+I2C | 1 | 10,450 | 10,450 |
| 9 | FAN | 2 | 2,500 | 5,000 |
| 10 | POWER SUPPLY | 1 | 5,000 | 5,000 |
| 11 | REALY | 1 | 8,000 | 8,000 |
| 12 | SWITCH | 2 | 400 | 800 |
| 13 | RESISTOR | 10 | 100 | 1,000 |
| 14 | LED | 6 | 100 | 600 |
| 15 | BUZZER | 1 | 1,000 | 1,000 |
| 16 | WIRES | 60 | 100 | 6,000 |
| 17 | PCB | 1 | 1,500 | 1,500 |
| 18 | STEP DOWN | 1 | 5,500 | 5,500 |
| 19 | BATTERY | 1 | 1,000 | 1,000 |
| 20 | FEMALE DC JACKS | 3 | 400 | 1,200 |
| 21 | MALE DC JACKS | 3 | 700 | 2,100 |
| 22 | PROJECT COVER | 1 | 5,000 | 5,000 |
| 23 | LOAD CELL PLATE | 1 | 2,000 | 2,000 |
| 24 | MALE CONNECTORS | 2 | 400 | 800 |
| 25 | TRANSPORT |  |  | 15,000 |
| 26 | COMMUNICATION |  |  | 10,000 |
| 27 | BREAD BOARD | 1 | 4,000 | 4,000 |
| 28 | CLOUD/SMS |  |  | 30,000 |
|  |  |  | TOTAL | 232,950 |