PERSONAL ASSISTANCE FOR SENIORS WHO ARE SELF-RELIANT

TEAM ID PNT2022TMID12805

TEAM MEMBERS

19L150-Ghowshik

19L141-Suryaa

19L137-Sowbarnika

19L105-Awinstan

1.INTRODUCTION

In this world, there is a 5 billion people have access to medicine. It improves the patients health, and as well as improves the physical and mental strength. But, sometimes elderly people forget to which medicine to take their medicine at the correct time. 75% of people forget their medicines. The proposed idea is creating a voice based medicine reminder notification alert can be set for multiple medicines and timings including date, time and medicine description, this application will remind their user about the medicine in-take schedule. The project idea will be implemented with python IDLE software. If the medicine time arrives the web application will send the medicine name to the IoT device. The device will receive the medicine name and time with buzzer sound. Based on the received command, the person take the appropriate medicine at correct time. Thus the medicine reminder application can be made with proposed idea. This can be implemented in the field where the peoples are taking medicine. It is suitable for memory loss people. This methodology is used to patients take their medicine in correct time. It reduces the burden of allotting health care professionals in homes to monitor elder persons. This part contain the information about the overview of the project and the purpose of the project.

1.1PROJECT OVERVIEW

The basic outline of our project is to create a personal assistance for seniors who are self-reliant because, sometimes elderly people forget to take their medicine at the correct time. They also forget which medicine He / She should take at that particular time. And it is difficult for doctors/caretakers to monitor the patients around the clock. To avoid this problem, this medicine reminder system is developed. An app is built for the user (caretaker) which enables him to set the

desired time and medicine. These details will be stored in the IBM Cloud . If the medicine time arrives the web application will send the medicine name to the IoT Device through the IBM IoT platform. The device will receive the medicine name and notify the user with buzzer sound.

1.2 PURPOSE

The purpose of our project is to make the patient to feel confident while taking the medicine, make them to take the wright medicine at the right time. There is no need for caretakers. using our project they can get notification at the right time .

2.LITERATURE SURVEY

PAPER 1:

Vikas Anna Mali, " IoT Based Intelligent Medicine Box with Assistance"International Research Journal of Engineering and Technology (IRJET),Volume: 09 Issue: 07 | July 2022

PROPOSED WORK:

In this paper, the intelligent medicine box has been designed. This medicine box stores the medicines and which alerts and remind to patients to take the medicines time to time. So that it avoids the health hazards due to negligence of not taking medicines time to time. Also in this paper different health parameters such as pulse rate, blood pressure and temperature and ECG of heart are measured. These parameters are sent to firebase IoT. Doctors can monitor health parameters on firebase website and accordingly decisions can be taken about patient treatment.

PAPER 2:

Padmini V,Pallavi KS, Prema M , Priya Hegde, Dr.Sreenivasa Setty,"INTERNET OF THINGS (IOT) BASED SMART HEALTHCARE MEDICALBOX FOR ELDERLY PEOPLE" , IJARIIE-ISSN,Vol-7 Issue-3 2021

PROPOSED WORK:

The iot-based smart healthcare system has been advised here, which will hold an intelligence medicine box affiliated with sensors and servers for usual health supervising. The smart medicine box which is having wireless internet connectivity will help the patients to get easy communication between the doctor and the patient without influencing physically and also to get a regular health checkup. This advised medicine box assists the user/patient to use up the exact medicine with an email at the exact time which will facilitate the patients to take the medicines. A laptop is used to store the elaborate data about the doctor and the patient along with the prescription and designation date. Here, the laptop is used as a server. Both the doctors and the patients will have their unique IDs' and passwords for approaching the server. For the doctor's comfort, the temperature and the medication information of the patients are stored on the server. The patient's prescription can be modified by the doctors if essential or in a serious condition, which will also be advised through email.

PAPER 3:

Pampapathi B M , Chandana Murthy , Supritha Kumar , Pooja M , SupriyaK,"Survey on IOT Based Medical Box for Elderly People" ISSN 2278-3091 ,Volume 10, No.3, May - June 2021

PROPOSED WORK:

A smart IOT-based healthcare system, which includes a sensor associated intelligence medicine box and a server for routine health monitoring, has been proposed here. This wireless internet access smart medicine box allows patients to get daily health care and to establish easy contact without physically meeting between doctor and patient. The recommended medicine box allows the patient to take the correct medicine at the right time, along with an email to help the patient take the medicine. A laptop is used as a server where, along with prescription and appointment date, accurate information about the doctor and patient is kept. The doctor and the patient both have IDs and passwords to access the server. In addition, the drug data and patient temperature are kept on the server for the convenience of the doctor. The doctor can if necessary,

change the patient's prescription, which will also be notified by email. In addition, in the event of an emergency, the doctor should take immediate action.

PAPER 4:

Prof. R.M Gawande, Miss Shinde Gayatri, Miss Supekar Pragati, Miss Shevkar Vaishali,

Mr Shevkar Akash ,"Smart Medicine Box for Old Age People", IJARIIEISSN(O)-Vol-5 Issue-1 2019.

PROPOSED WORK:

Proper Medication is necessary to become a healthy but failure of that can create big trouble for a patient. This is extremely problematic for the elderly patient who had problem in keeping track of their medicine. So to overcome this we made this Smart Medicine Box which keep tracks of

the dosage and duration between each consumption. Poor eyesight as one of the contributors for medicine consumption errors such as miss dosage since the elderly finds it troublesome to read the instruction on the medicine case and identifying the right dosage of the medicine along with that memory loss is common in old age due to that decrease in speed of information being retrieved. Hence, this Smart Medicine Box will track their medication and inform patient to take right dosage of right medicine at the right time.

2.1 EXISTING PROBLEM

Elderly patients they are facing a lot of problems while taking medicines First thing was Patients may be frightened of potential side effects. They may have also experienced previous side effects with the same or similar medicine. Additionally, patients report not taking their medication because they may have witnessed side effects experienced by a friend or family member who was taking the same or similar medication. From seeing those side effects experienced by someone else, it may have led them to believe the medication caused those problems.so they avoid to take the prescribed medicine by the doctor. major barrier to adherence is often the cost of the medicine prescribed to the patient. The high cost may lead to patients not filling their medications in the first place. They may even ration what they do

fill in order to extend their supply.so they can avoid to buy the medicine once it was empty. Nonadherence can also happen when a patient does not understand the need for the medicine, the nature of side effects or the time it takes to see results. When a patient has several different medicines prescribed with higher dosing frequency, the chances that they should take the medicines wrongly it will lead to other sever problems Patients who don't feel any different when they start or stop their medicine might see no reason to take it. Additionally, once a patient's condition is controlled, they may think the problem has resolved and may discontinue using the medication without doctors concern. These are the major issues facing by the older patients.

2.2 REFERENCES

https://www.irjet.net/volume9-issue7

https://ijariie.com/AdminUploadPdf/INTERNET_OF_THINGS__IOT__BASED_SMART_HEALTHCARE_MEDICAL_BOX_FOR_ELDERLY_PEOPLE_ijariie14798.pdf

https://www.warse.org/IJATCSE/static/pdf/file/ijatcse531032021.pdf

http://ijariie.com/AdminUploadPdf/Smart_Medicine_Box_for_Old_Age_People_ijariie9536.pdf

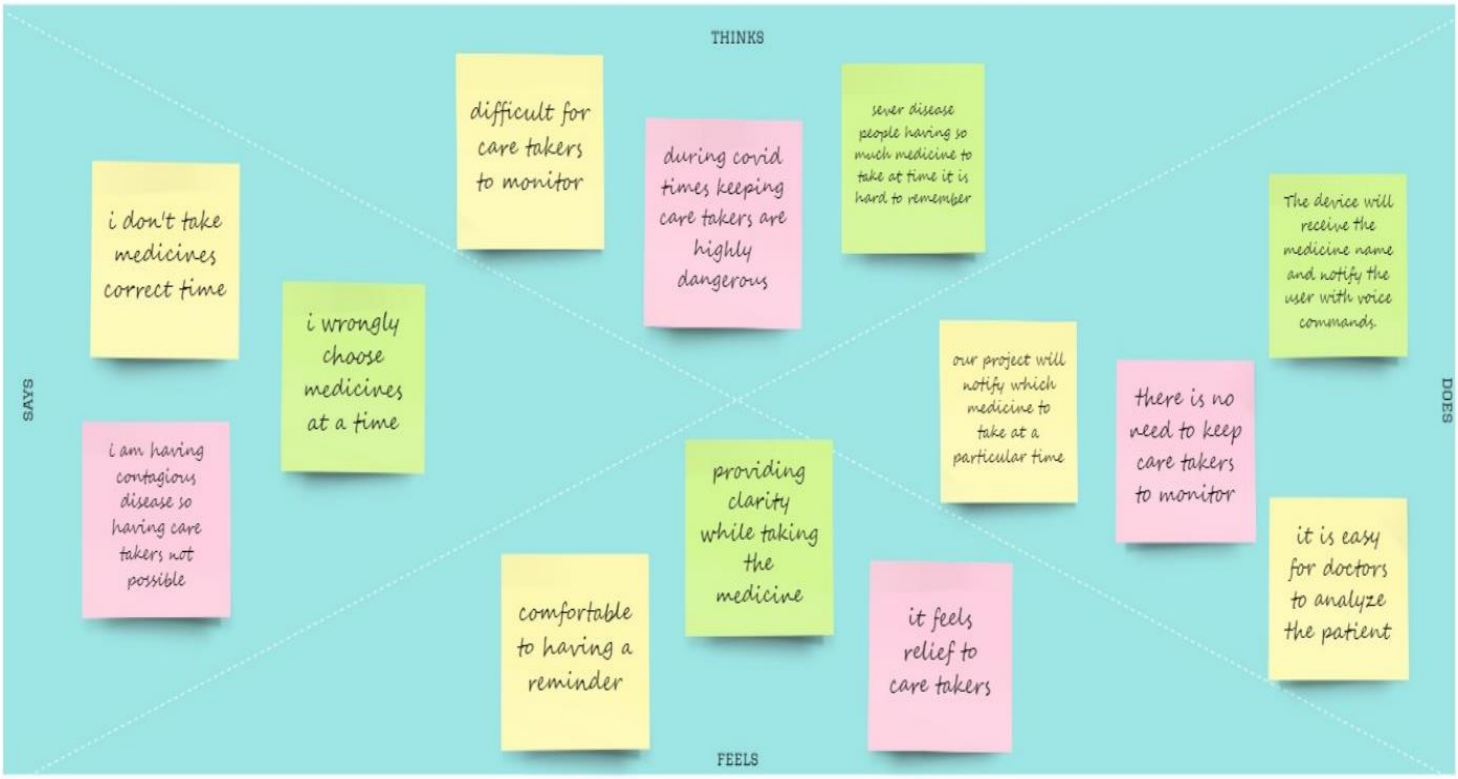
2.3 PROBLEM STATEMENT DEFINITION

A medicine reminder app designed for people who frequently forget to take their medications. An app is built for the caretaker which enables him to set the desired time and medicine.You may also keep track of your appointments. Its parental feature distinguishes it from other apps on the market, allowing you to keep track of and remotely assist your loved ones who find it difficult to utilize such an app with their reminders. The device will receive the medicine name and modify the user with voice command.

CUSTOMER PROBLEM STATEMENT:

3. IDEATION & PROPOSED SOLUTION

3.1 EMPATHY MAP CANVAS

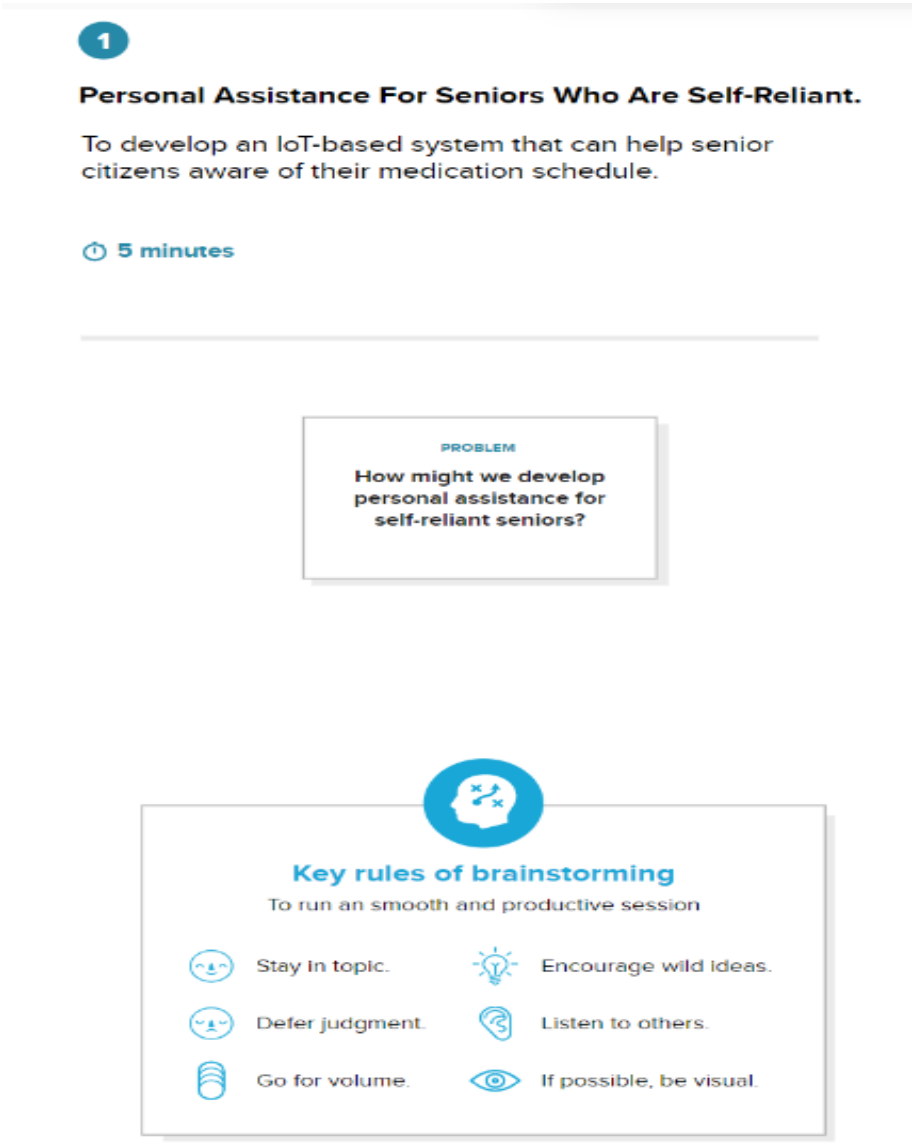


PAIN
ELDERLY PEOPLE WILL TAKE MEDICINES WRONGLY
IT IS VERY HARD TO REMEMBER
SOMETIMES THEY WILL MISMATCH THE MEDICINES

GAIN
PROVIDING REMINDER THROUGH NOTIFICATION ABOUT WHICH MEDICINE TO TAKE
PROVIDING CONFIDENT TO THE PATIENT WHILE TAKING THE CORRECT MEDICINE
THROUGH THIS SPREADING CONTAGIOUS DISEASE CAN BE AVOID

3.2 IDEATION & BRAINSTORMING

Brainstorm & Idea Prioritization: Brainstorming provides a free and open environment that encourages everyone within a team to participate in the creative thinking process that leads to problem-solving. Prioritizing volume over value, out-of-the-box ideas are welcome and built upon, and all participants are encouraged to collaborate, helping each other develop a rich number of creative solutions.



2

Brainstorm

Write down any ideas that come to mind that address your problem statement.

10 minutes

Awinstan G

the data of the senior medication must be secure

use of Raspberrypi as the IoT device

Use of led/lcd as alearting device

Developing a Medi-bot that can bring medicine to senior citizen to their place using the RSSI value of the senior's smartphone

Suryaa E.N

To Earn revenue, we can give space for advertisement on the website.

To make the Caretaker meditation schedule entry website user-friendly we can add an auto-fill option.

To create a check method that assures the caretaker that the senior has taken the medication

A button can be used as a check system.

Sowbarnika C

The IoT device must be cost effective

We can have voice assistance which can obtain the meditation schedule from senior citizens using audio.

Alert SMS to senior citizens smartphone

we must send message to the care-taker about the medication taken by the senior

Ghowshik Y

Developing a website where caretake or senior can enter their meditation schedule which can be stored in IBM cloud

Use of nodeMCU as the IoT device

We can't develop a Medi-bot that can track the senior to his location using RSSI value because we can't say that senior people will carry their smartphone everywhere they go in an indoor location

Use of LCD/LED and speaker as alert device

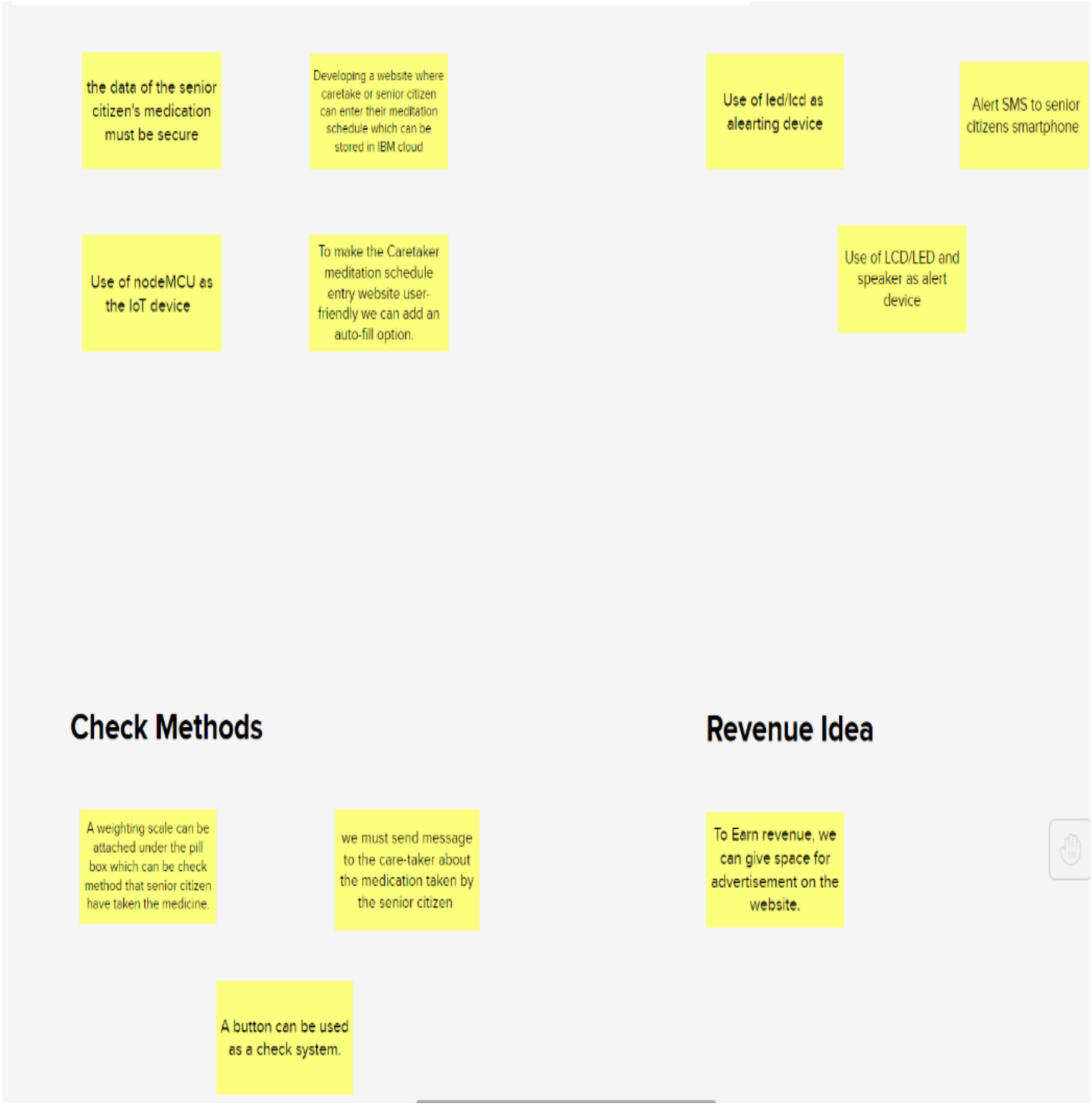
A weighting scale can be attached under the pill box which can be check method that senior citizen have taken the medicine

3

Group ideas

Take turns sharing your ideas while clustering similar or related notes as you go. Once a sticky notes have been grouped, give each cluster a sentence-like label. If a cluster is bigger than six sticky notes, try and see if you and break it up into smaller sub-groups.

20 minutes

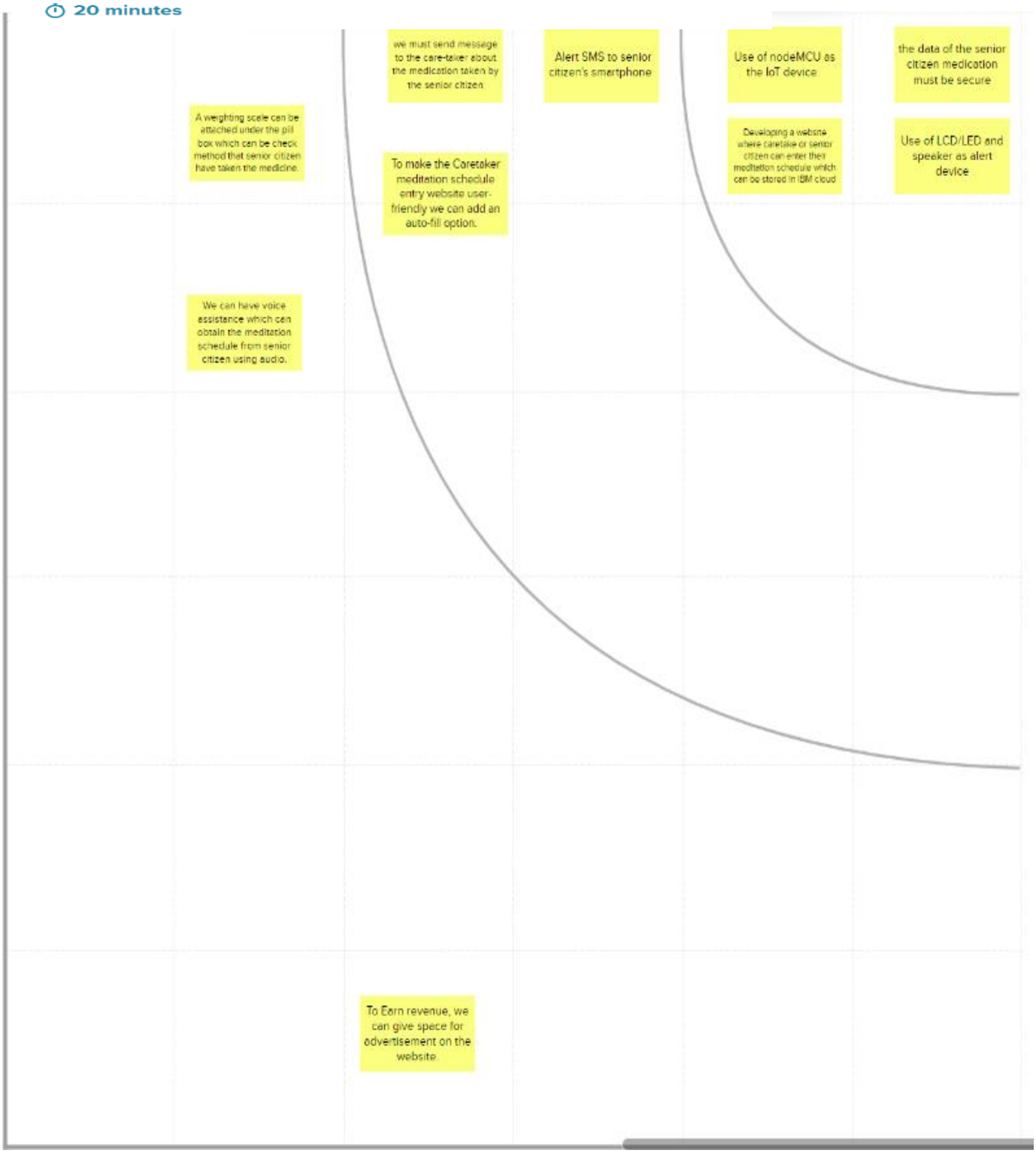


4

Prioritize

Your team should all be on the same page about what's important moving forward. Place your ideas on this grid to determine which ideas are important and which are feasible.

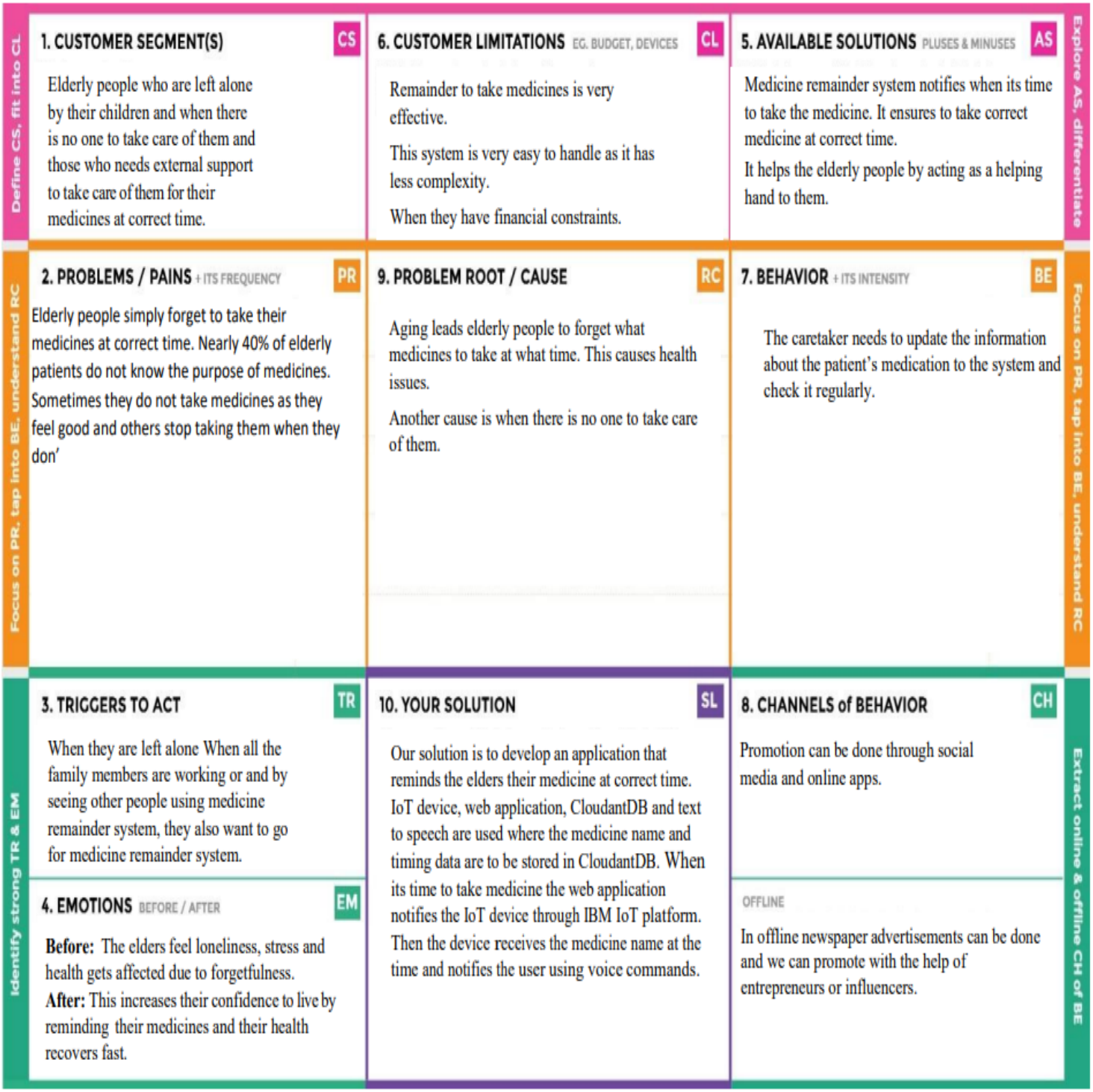
20 minutes



3.3 PROPOSED SOLUTION

S.No	Parameter	Description
1.	Problem Statement (Problem to be solved)	• Elderly people who forget to take their medicine and people who suffer from severediseases need to take their medicines at correct time. Therefore we should create a routine ,a resort to something that reminds them to take medicined at right time. • It is also a tedious task for the doctors/caretakers to monitor the patients around the clock.
2.	Idea / Solution description	• Medicine remainder serves a s a goodway to stay on track and uphold an appropriate schedule. • Hence to avoid the problem faced bythe elderly, we develop a medicine remainder system to ensure that theyare properly taking their medicationswhich avoids unnecessary illness. • Here we use IoT device, web application, CloudantDB and text to speech. • The medicine name and timing data should be stored in CloudantDB. • If the medicine time arrives the web application notifies the IoT device through the IBM IoT platform. The device will receive the medicine name and notify theuser with voice commands.
3.	Novelty / Uniqueness	• This app sends a notification to the user whenthe person should take the medicine. • It can also be used to analyze and track • whether their loved ones are taking their medication on time.

3.4 PROBLEM SOLUTION FIT



4. REQUIREMENT ANALYSIS

4.1 FUNCTIONAL REQUIREMENT

FR no	Functional requirement	Sub requirement
FR 1	User Registration	Registration through Form Registration through Gmail Registration through LinkedIN
FR 2	User Confirmation	Confirmation via Email Confirmationvia OTP
FR 3	User input	input as program of robot
FR 4	Processing of user input	Image acquisition,pre-processing, image segmentation
FR 5	Feature extraction ofuser image	Machine learning algorithms, image classification, data mapping
FR 6	Result of user input	Display the disease name and fertilizer lack for the issues

PROJECT FLOW:

Through a web application, users can set the time and name of the medication. The IBM Cloudant DB will house all of the medication- related information. At the chosen time, the online application will communicate the drug name to the IoT device. The device will voice out the medication name after receiving the name using IBM text to speech service to remind the user to take the medication.

To accomplish this, we have to complete all the activities and tasks listed below:

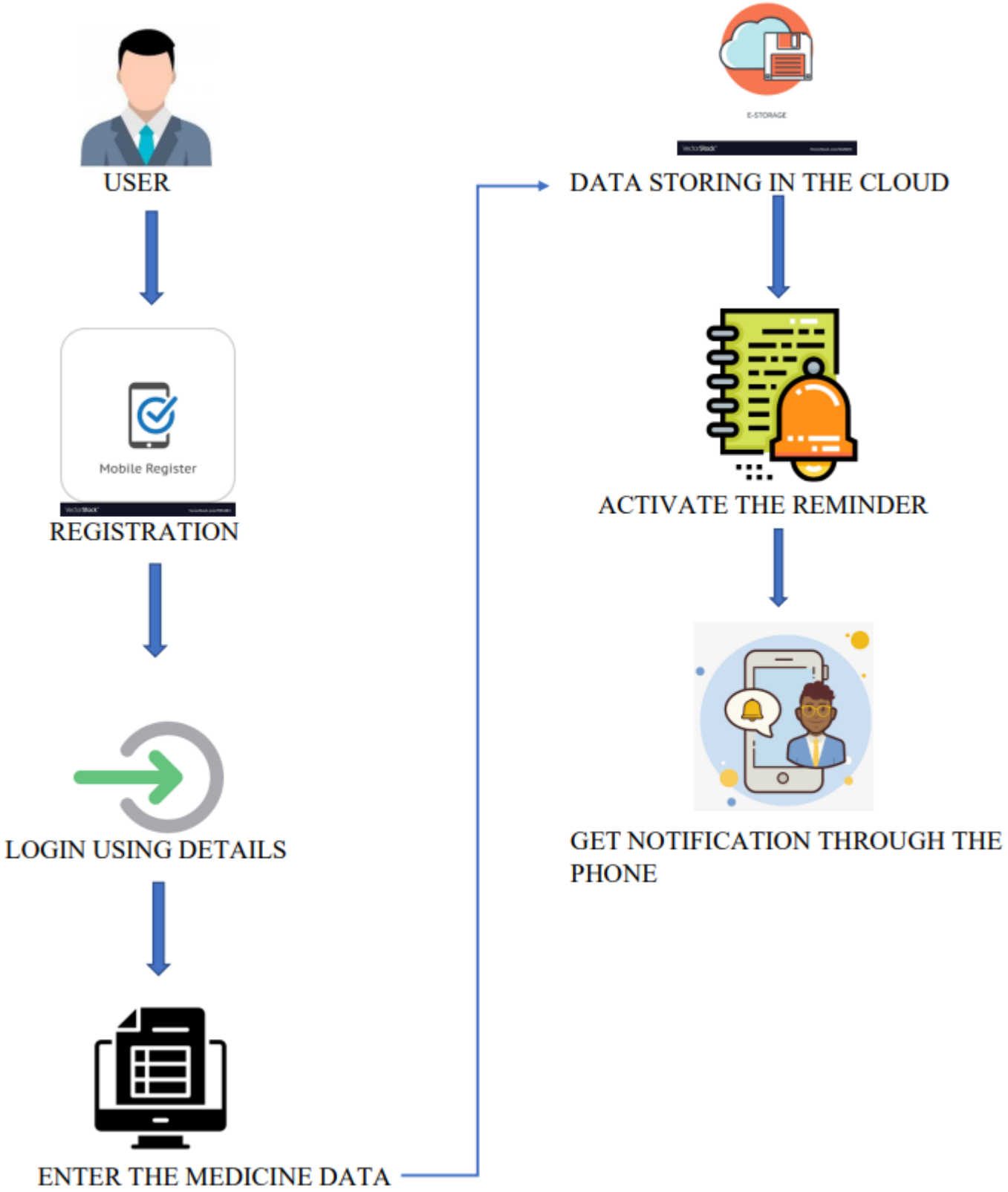
1. Construct and set up IBM Cloud Services
2. Create an IoT device and setup the IBM Watson IoT Platform. Create a Node-RED service.
3. To store information about medications, create a database in CloudantDB. To create a web application, use the Node-RED Service.
4. Create a Python script to sign up for IBM's IoT platform and produceaudio notifications.

4.2 NON-FUNCTIONAL REQUIREMENTS

FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	App can be used by anyone who has knowledgeabout internet and computer.
NFR-2	**Security**	For security, TFA is enabled and biometricsare also added for user safety.
NFR-3	**Reliability**	Highly reliable since, It uses Trusted cloudservices like IBM
NFR-4	**Performance**	Performance is better compared to othermarketproducts.
NFR-5	**Availability**	Available on mobile app.
NFR-6	**Scalability**	Using Cloud services, makes the scalabilityhigher the using traditional database.

5. PROJECT DESIGN

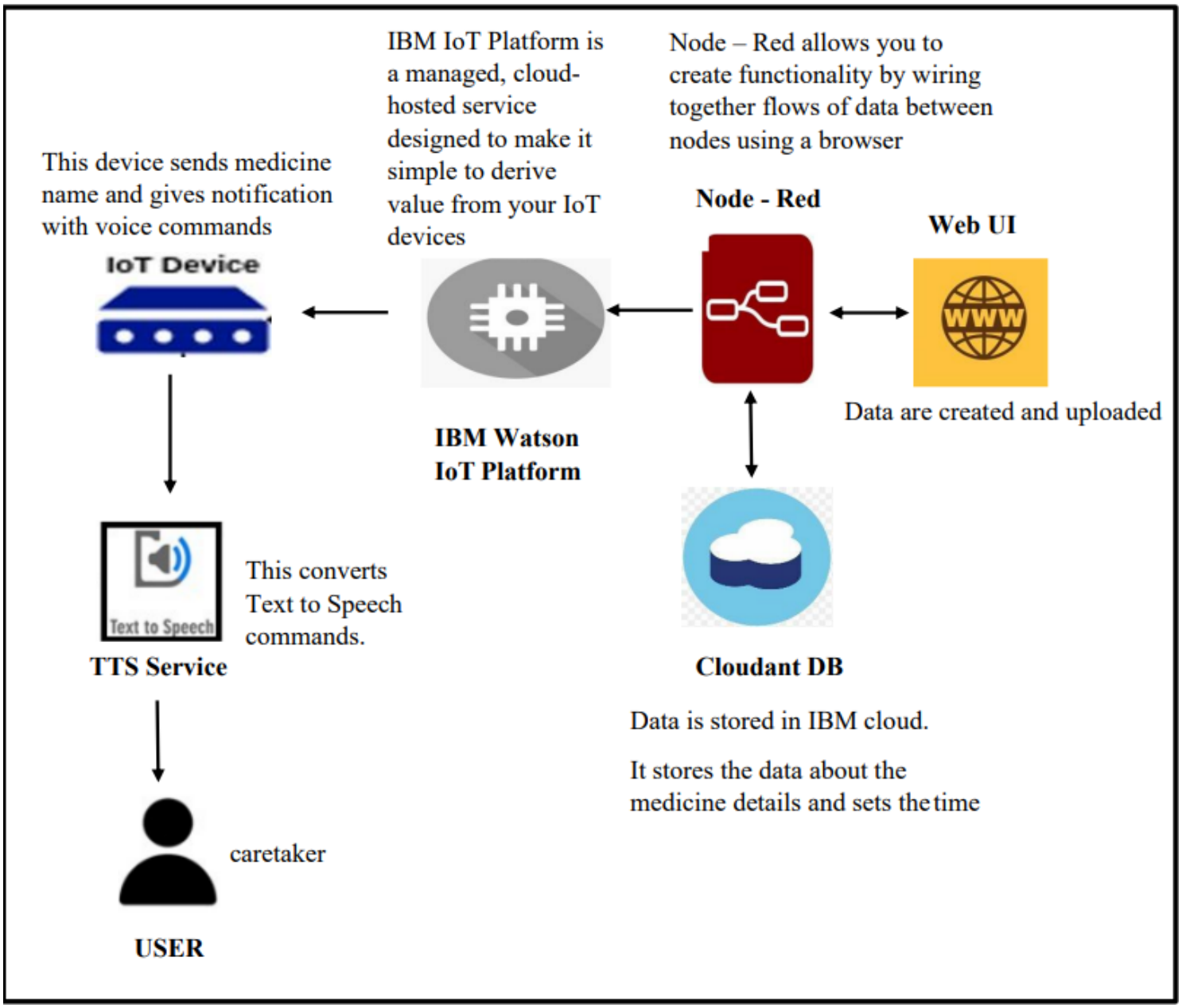
5.1 DATA FLOW DIAGRAMS



User Type	Functional Requirement	User Story Number	User Story/task	Acceptance Criteria	Priority	Release
User	Register User	USN-1	User can register themselves as a user	New user account is created	medium	sprint1
User	Login	USN-2	User login to theapp	Going to the dashboard	medium	sprint2
User	Add Medicine Details	USN-3	User add his/her medicine details	Added medicine details successfully	high	sprint3
User	Activate/ Deactivate Reminder	USN-4	User activate/deactivat emedicine reminder	Activated/ Deactivated successfully	high	sprint4
User	Logout	USN-5	User logout fromthe app	Logout the user	low	sprint4

5.2 SOLUTION & TECHNICAL ARCHITECTURE

SOLUTION ARCHITECTURE



TECHNICAL ARCHITECTURE

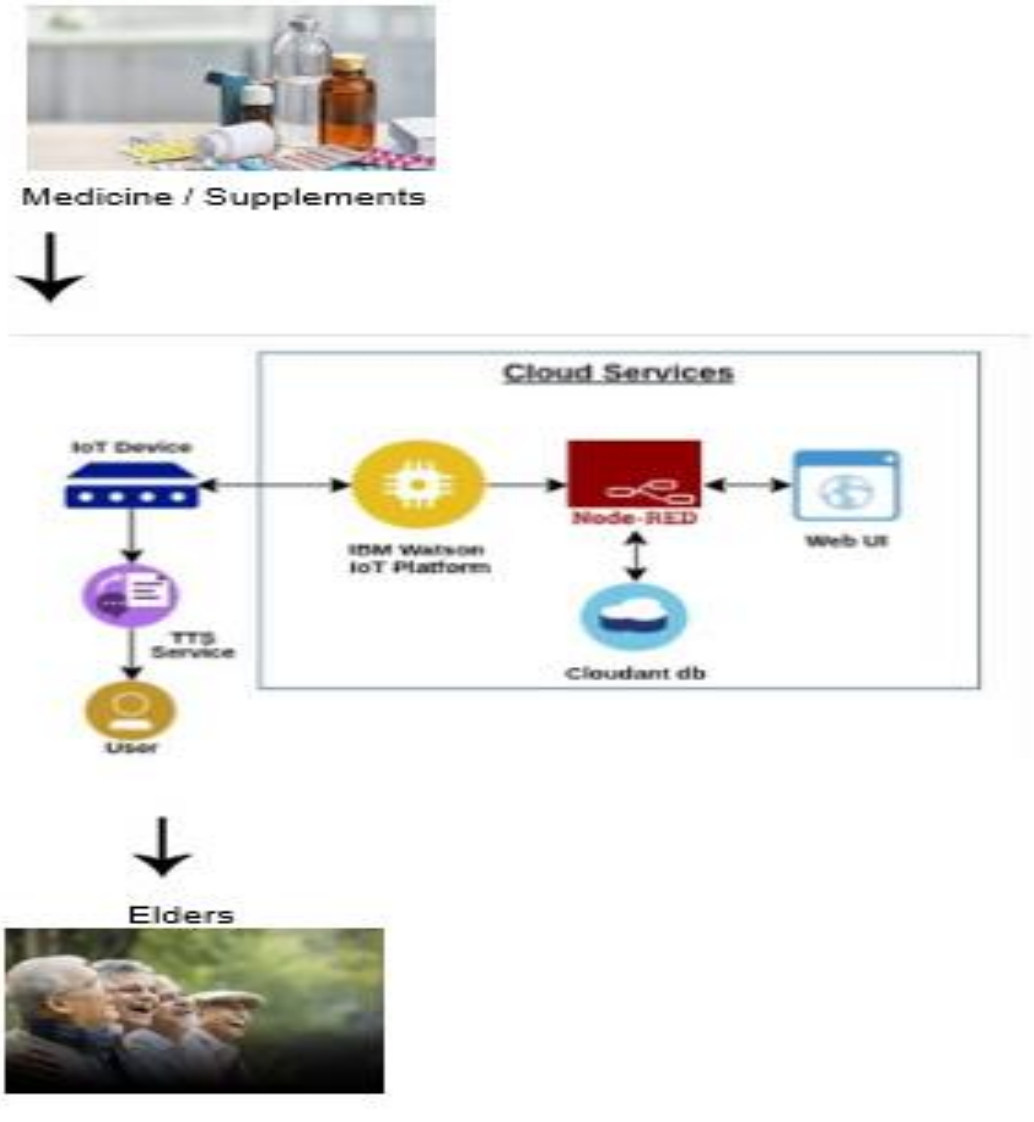


TABLE-1 : COMPONENTS & TECHNOLOGIES:

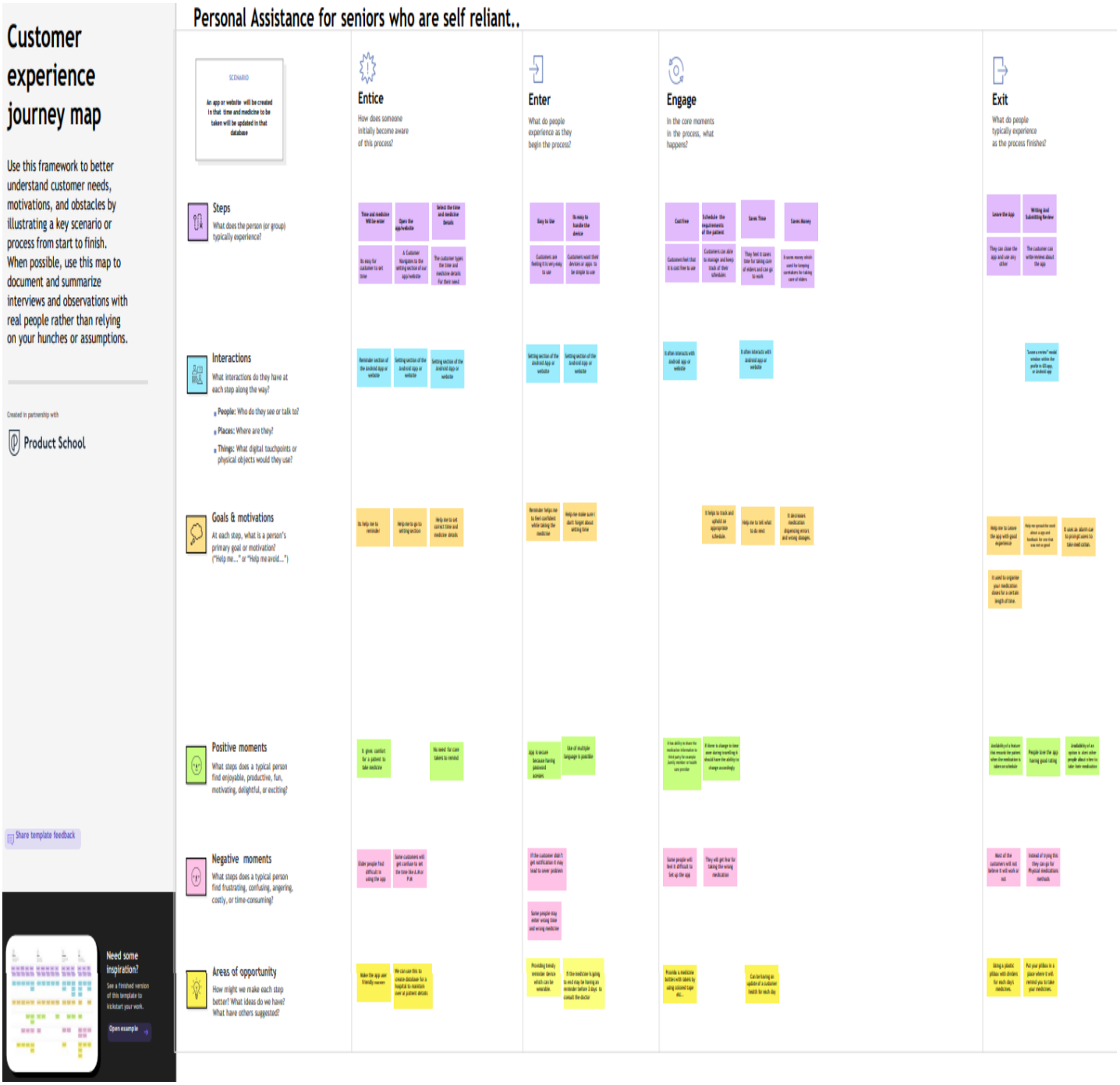
S. No	Component	Description	Technology
1.	User Interface	Mobile App	MIT
2.	Application Logic-1	Gathering information about medications and examining timing	Python
3.	Application Logic-2	Alarms	IBM Watson
4.	Cloud Database	Cloud database service	Cloudant DB by IBM

TABLE-2: APPLICATION CHARACTERISTICS:

S. No	Characteristics	Description	Technology
1.	Open-Source Frameworks	Creating mobile applications, setting up IoT devices, programming IoT devices, using atext-to-speech service, and storingdata in the cloud	BM Watson, Node Red, IBM Cloud, and Mit
2.	Security Implementations	Putting encryption to use forsecurity	SHA-256.
3.	Scalable Architecture	The application can load as manymembers as there are logged in	IBM Watson, Mit

4.	Availability	Application is offered 24/7	IBM Watson, Node Red, IBM Cloud, and Mit
5.	Performance	Reminder with correct timing	Iotf, Ibm, and Watson

5.3 USER STORIES



6. PROJECT PLANNING & SCHEDULING

6.1 SPRINT PLANNING & ESTIMATION

SPRINT	FUNCTIONAL REQUIREMENTS (EPIC)	USER STORY NUMBER	USER STORY / TASK	STORY POINTS	PRIORITY	TEAM MEMBERS
Sprint - 1	APPLICATION DEVELOPMENT	USN - 1	To develop a mobile application for a caretaker to enter medication details in the MIT app inventor.	3	High	Ghowshik, Awinstan
Sprint - 2	HARDWARE IMPLEMENTATION	USN - 2	We will create hardware that acts as a reminder to seniorpeople.	3	Medium	Ghowshik, Suryaa
Sprint - 3	INTEGRATION PHASE	USN - 3	To integrate the application developed and the hardware	3	High	Surya a, Awinstan and Sowbarnika
Sprint - 4	SYSTEM DESIGN	USN - 4	Uses a cloud database to store medication schedules.	3	High	Ghowshik, Sowbarnika

SPRINT	TOTAL STORY POINTS	DURATION	SPRINT START DATE	SPRINT END DATE(PLANNED)	STORY POINTS COMPLETED (As on Planned End Date)	Sprint ReleaseDate (Actual)
Sprint - 1	25	10 Days	24-Oct-2022	3-Nov-2022	25	04-Nov-2022
Sprint - 2	15	4 Days	01-Nov-2022	5-Nov-2022	15	05-Nov-2022
Sprint - 3	25	7 Days	02-Nov-2022	09-Nov-2022	25	09-Nov-2022
Sprint - 4	15	9 Days	08-Nov-2022	17-Nov-2022	15	17-Nov-2022

VELOCITY:

Sprint 1 average velocity:

Average velocity = 25 /10 = 2.5

Sprint 2 average velocity:

Average velocity = 15/ 4 = 3.75

Sprint 3 average velocity:

Average velocity = 25/ 7 = 3.5

Sprint 4 average velocity:

Average velocity = 15/9 = 1.7

6.2 SPRINT DELIVERY SCHEDULE

TITLE	DESCRIPTION	DATE
Literature Survey & Information Gathering	Gather/collect relevant information on the project use case, including references to existing solutions, technical papers,research publications, and so on.	30 AUGUST 2022

Prepare EmpathyMap	Prepare the user's pains and gains for the empathymap canvas. Make a list of problem statements.	17 SEPTEMBER 2022
Ideation	List the ideas generated during the brainstormingsession and rank the top three according to relevance and viability.	17 SEPTEMBER 2022
Proposed Solution	Create a proposal for a solution that details its innovation, viability as a business model, social effect, scalability, and other factors.	1 OCTOBER 2022
Problem Solution Fit	Prepare a problem - solution fit document.	1 OCTOBER 2022
Solution Architecture	Prepare solution architecture document.	1 OCTOBER 2022
Customer Journey	Create customer journey maps to comprehend how users engage with and use the application from entry to exit.	16 OCTOBER 2022
Functional Requirement	Prepare the functional requirement document.	16 OCTOBER 2022
Data Flow Diagrams	Prepare the data flow diagrams and submit for review.	16 OCTOBER 2022
Technology Architecture	Draw the technology architecture diagram.	16 OCTOBER 2022

Prepare Milestone & Activity List	Prepare the milestones & activity list of the project.	22 OCTOBER 2022
Project Development – Delivery of Sprint-1, 2, 3 & 4	Prepare the code for submission and test it.	19 NOVEMBER 2022

7. CODING & SOLUTIONING (EXPLAIN THE FEATURES ADDED IN THE PROJECT ALONG WITH CODE)

Ibm cloud creation code

```
#include <NTPClient.h>
#include <ESP8266WiFi.h>
#include <WiFiUdp.h>
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
#include <TimeLib.h>
LiquidCrystal_I2C lcd(0x27, 16, 2);
const char *ssid = "Awinstan";
const char *password = "Awinstan1";
long utcOffsetInSeconds = 5*60*60+60*30;
int h,m,s;
char daysOfTheWeek[7][4] = {"SUN", "MON", "TUE", "WED", "THU", "FRI", "SAT"};
WiFiUDP ntpUDP;
//NTPClient timeClient(ntpUDP, "pool.ntp.org", utcOffsetInSeconds);
char Time[] = "00:00:00";
byte last_second, second_, minute_, hour_, day_, month_;
void showIndianTime(NTPClient timeClient)
{
    timeClient.update();
    unsigned long unix_epoch = timeClient.getEpochTime();
    second_ = second(unix_epoch);
```

```
    if (last_second != second_) {
    minute_ = minute(unix_epoch);
    hour_   = hour(unix_epoch);
    Time[7] = second_ % 10 + 48;
    Time[6] = second_ / 10 + 48;
    Time[4]  = minute_ % 10 + 48;
    Time[3]  = minute_ / 10 + 48;
    Time[1]  = hour_   % 10 + 48;
    Time[0]  = hour_   / 10 + 48;
    lcd.setCursor(2, 0);
    lcd.print("Mumbai-India");
    lcd.setCursor(2, 1);
    lcd.print(daysOfTheWeek[timeClient.getDay()]);
    lcd.setCursor(6, 1);
    lcd.print(Time);
    last_second = second_;
    }
    delay(500);
}
void showLondonTime(NTPClient timeClient)
{
    timeClient.update();
    unsigned long unix_epoch = timeClient.getEpochTime();
    second_ = second(unix_epoch);
    if (last_second != second_) {
    minute_ = minute(unix_epoch);
    hour_   = hour(unix_epoch);
    Time[7] = second_ % 10 + 48;
    Time[6] = second_ / 10 + 48;
    Time[4]  = minute_ % 10 + 48;
    Time[3]  = minute_ / 10 + 48;
    Time[1]  = hour_   % 10 + 48;
    Time[0]  = hour_   / 10 + 48;
    lcd.clear();
```

```
    lcd.setCursor(5, 0);
    lcd.print("London");
    lcd.setCursor(2, 1);
    lcd.print(daysOfTheWeek[timeClient.getDay()]);
    lcd.setCursor(6, 1);
    lcd.print(Time);
    last_second = second_;
    }
    delay(500);
}

void showNewYorkTime(NTPClient timeClient)
{
    timeClient.update();
    unsigned long unix_epoch = timeClient.getEpochTime();
    second_ = second(unix_epoch);
    if (last_second != second_) {
    minute_ = minute(unix_epoch);
    hour_   = hour(unix_epoch);
    Time[7] = second_ % 10 + 48;
    Time[6] = second_ / 10 + 48;
    Time[4]  = minute_ % 10 + 48;
    Time[3]  = minute_ / 10 + 48;
    Time[1]  = hour_   % 10 + 48;
    Time[0]  = hour_   / 10 + 48;
    lcd.clear();
    lcd.setCursor(5, 0);
    lcd.print("New-York");
    lcd.setCursor(2, 1);
    lcd.print(daysOfTheWeek[timeClient.getDay()]);
    lcd.setCursor(6, 1);
    lcd.print(Time);
    last_second = second_;
    }
```

```
    delay(500);
}

void showParisTime(NTPClient timeClient)
{
    timeClient.update();
    unsigned long unix_epoch = timeClient.getEpochTime();
    second_ = second(unix_epoch);
    if (last_second != second_) {
    minute_ = minute(unix_epoch);
    hour_   = hour(unix_epoch);
    Time[7] = second_ % 10 + 48;
    Time[6] = second_ / 10 + 48;
    Time[4]  = minute_ % 10 + 48;
    Time[3]  = minute_ / 10 + 48;
    Time[1]  = hour_   % 10 + 48;
    Time[0]  = hour_   / 10 + 48;
    lcd.clear();
    lcd.setCursor(5, 0);
    lcd.print("Paris");
    lcd.setCursor(2, 1);
    lcd.print(daysOfTheWeek[timeClient.getDay()]);
    lcd.setCursor(6, 1);
    lcd.print(Time);
    last_second = second_;
    }
    delay(500);
}

void showTokyoTime(NTPClient timeClient)
{
    timeClient.update();
    unsigned long unix_epoch = timeClient.getEpochTime();
    second_ = second(unix_epoch);
```

```
    if (last_second != second_) {
    minute_ = minute(unix_epoch);
    hour_   = hour(unix_epoch);
    Time[7] = second_ % 10 + 48;
    Time[6] = second_ / 10 + 48;
    Time[4]  = minute_ % 10 + 48;
    Time[3]  = minute_ / 10 + 48;
    Time[1]  = hour_   % 10 + 48;
    Time[0]  = hour_   / 10 + 48;
    lcd.clear();
    lcd.setCursor(5, 0);
    lcd.print("Tokyo");
    lcd.setCursor(2, 1);
    lcd.print(daysOfTheWeek[timeClient.getDay()]);
    lcd.setCursor(6, 1);
    lcd.print(Time);
    last_second = second_;
    }
    delay(500);
}

void setup(){
   Serial.begin(115200);
   lcd.begin();
   WiFi.begin(ssid, password);
   lcd.clear();
   lcd.setCursor(0, 0);
   lcd.print("Connecting Wifi");
   lcd.setCursor(0, 1);
   while ( WiFi.status() != WL_CONNECTED )
   {
     delay ( 500 );
     lcd.print(".");
     Serial.print ( "." );
```

```
    }
    //timeClient.begin();
    lcd.clear();
}

void loop() {
    NTPClient timeClient1(ntpUDP, "pool.ntp.org", 5*60*60+60*30);
    for(int i=0;i<8;i++)
    {
      showIndianTime(timeClient1);
    }
    NTPClient timeClient2(ntpUDP, "pool.ntp.org", 1*60*60);
    for(int i=0;i<8;i++)
    {
      showLondonTime(timeClient2);
    }
    NTPClient timeClient3(ntpUDP, "pool.ntp.org", -4*60*60);
    for(int i=0;i<8;i++)
    {
      showNewYorkTime(timeClient3);
    }
    NTPClient timeClient4(ntpUDP, "pool.ntp.org", 2*60*60);
    for(int i=0;i<8;i++)
    {
      showParisTime(timeClient4);
    }
    NTPClient timeClient5(ntpUDP, "pool.ntp.org", 9*60*60);
    for(int i=0;i<8;i++)
    {
      showTokyoTime(timeClient5);
    }

}
```

Interfacing hardware component using wokwi

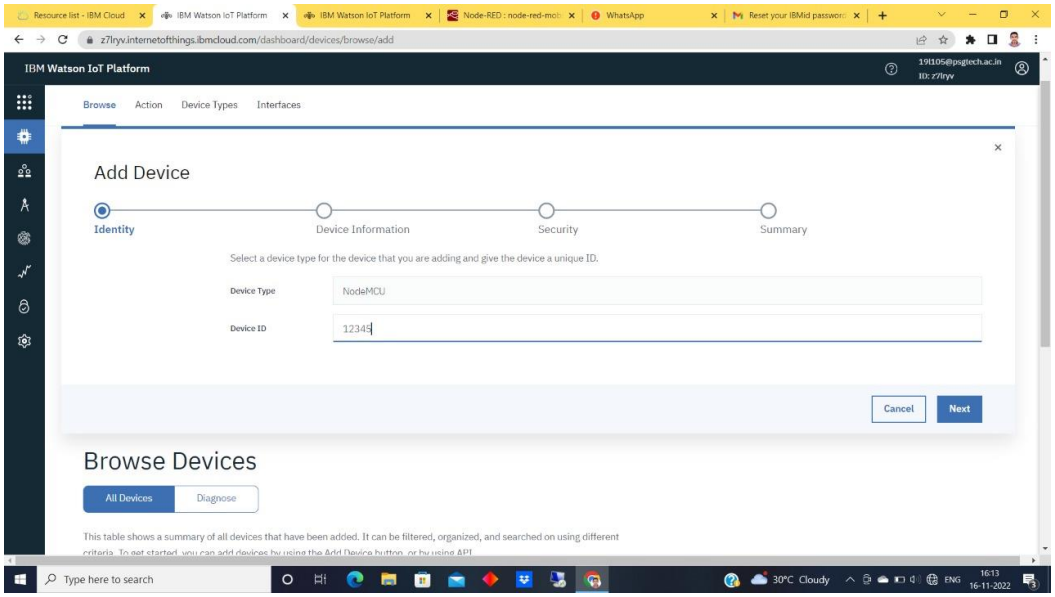
```
#include <LiquidCrystal.h>
#include <Wire.h>
#include <RTClib.h>

RTC_DS3231 rtc;
const int rs = 12, en = 11, d4 = 5, d5 = 4, d6 = 3, d7 = 2; // lcd
pins
LiquidCrystal lcd(rs, en, d4, d5, d6, d7);
#define getWellsoon 0
#define HELP_SCREEN 1
#define TIME_SCREEN 2

int Signal = 0;
int buzz = 13;
long previousMillis = 0;
long interval = 500; // buzzing interval
unsigned long currentMillis;
long previousMillisLCD = 0; // for LCD screen update
long intervalLCD = 2000; // Screen cycling interval
unsigned long currentMillisLCD;
// Set Reminder Change Time
int buzz8amHH = 8; // HH - hours ##Set these for reminder time in
24hr Format
int buzz9pmHH = 20; // HH - hours
int buzz9pmMM = 00; // MM - Minute
int buzz9pmSS = 00; // SS - Seconds
int nowHr, nowMin, nowSec; // to show current mm,hh,ss


void timeScreen() { // function to display Date and time in LCD screen
DateTime now = rtc.now(); // take rtc time and print in display
```

```
lcd.clear();
lcd.setCursor(0, 0);
lcd.print("Time:");
lcd.setCursor(6, 0);
lcd.print(nowHr = now.hour(), DEC);
lcd.print(":");
lcd.print(nowMin = now.minute(), DEC);
lcd.print(":");
lcd.print(nowSec = now.second(), DEC);
lcd.setCursor(0, 1);
lcd.print("Date: ");
lcd.print(now.day(), DEC);
lcd.print("/");
lcd.print(now.month(), DEC);
lcd.print("/");
lcd.print(now.year(), DEC);
}

void at9pm() { // function to start buzzing at 9pm
DateTime now = rtc.now();
if (int(now.hour()) >= buzz8pmHH) {
if (int(now.minute()) >= buzz8pmMM) {
if (int(now.second()) > buzz8pmSS) {

startBuzz();
break;

}

LiquidCrystal_I2C lcd(0x27, 16, 2);
const float BETA = 3950;
//LiquidCrystal_I2C lcd(0x27, 16, 4);
```

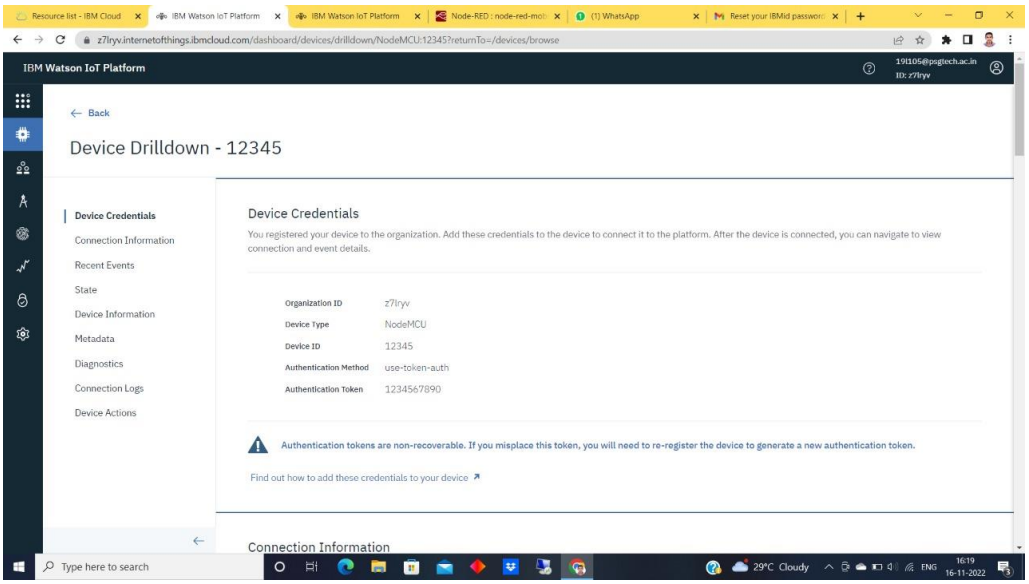
```
int buzzerPin = 8;
void setup() {
  Serial.begin(9600);
  pinMode(buzzerPin, OUTPUT);

}

void loop() {
  int analogValue = analogRead(A0);
  float c = 1 / (log(1 / (1023. / analogValue - 1)) / BETA + 1.0 / 298.15) - 273.15;
  lcd.begin(16,2);
  lcd.backlight();
  lcd.setCursor(0, 0);
  lcd.print(" Take medicine");
  lcd.setCursor(6, 0);
  lcd.setCursor(8, 0);
  tone(buzzerPin, 100, 1000);
  delay(1000);
}
```

9. RESULTS

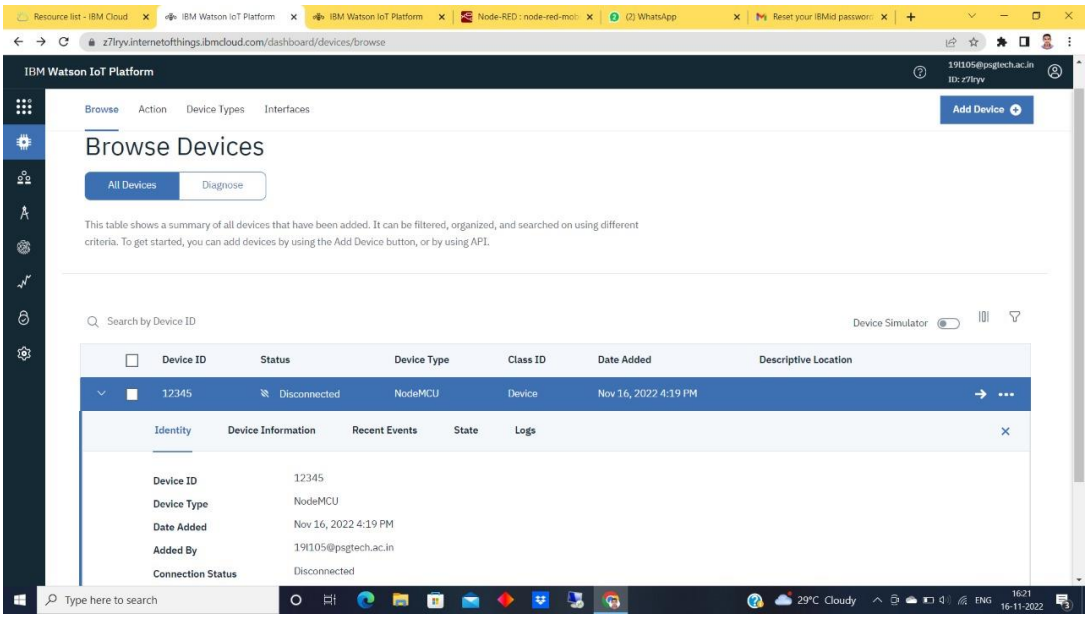
IBM CLOUD CREATION



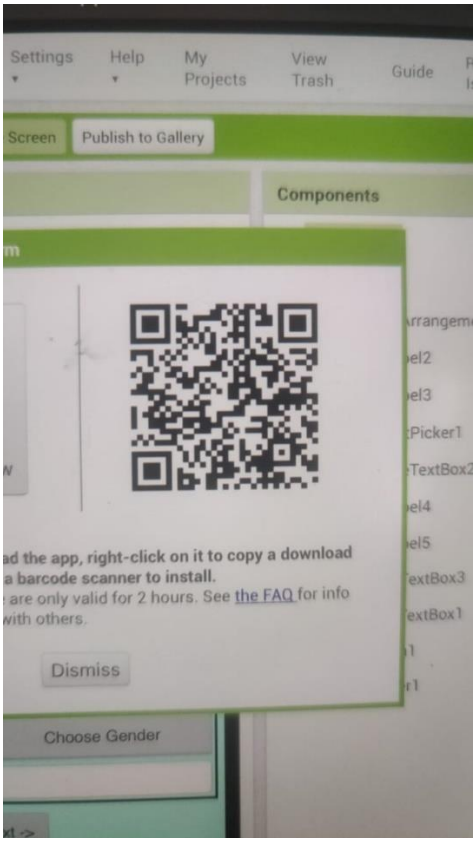
DEVICE INFORMATION



BROWSE DEVICE DETAILS TO CONNECT

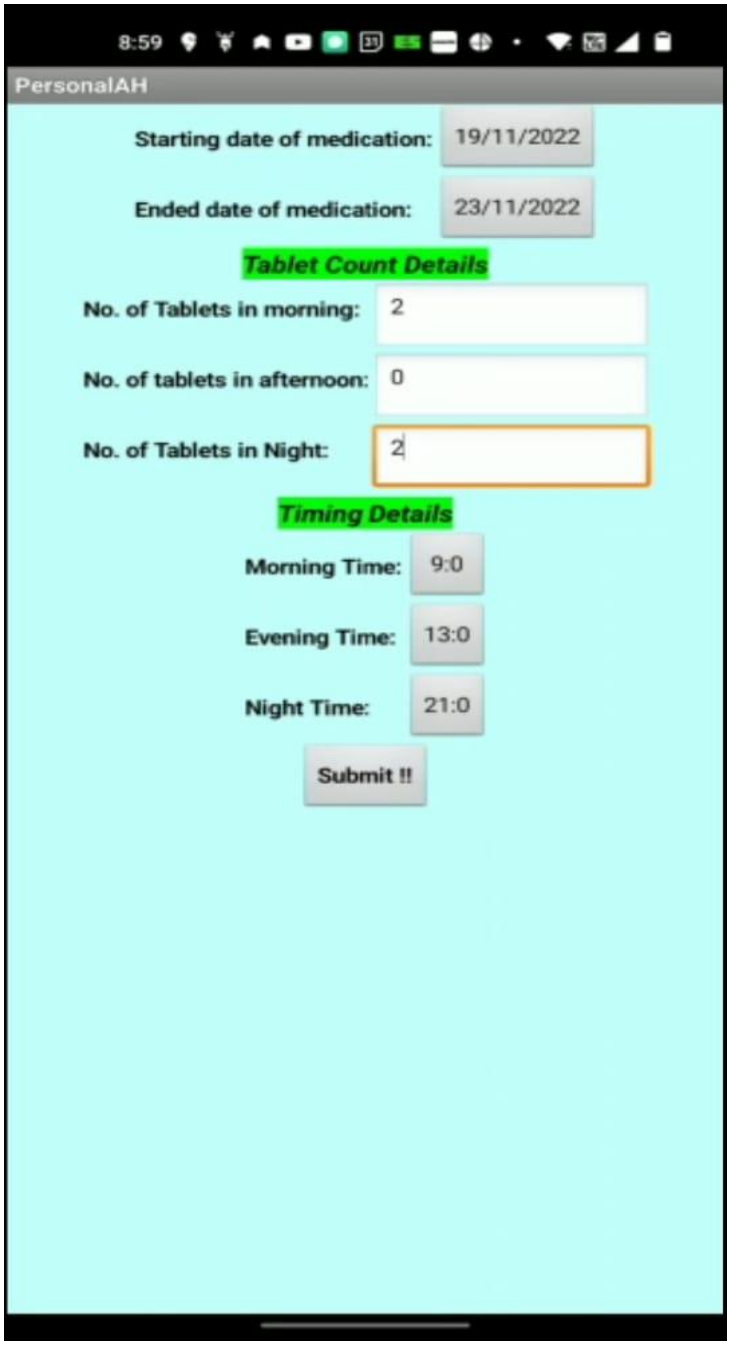


ANDROID APP CREATION

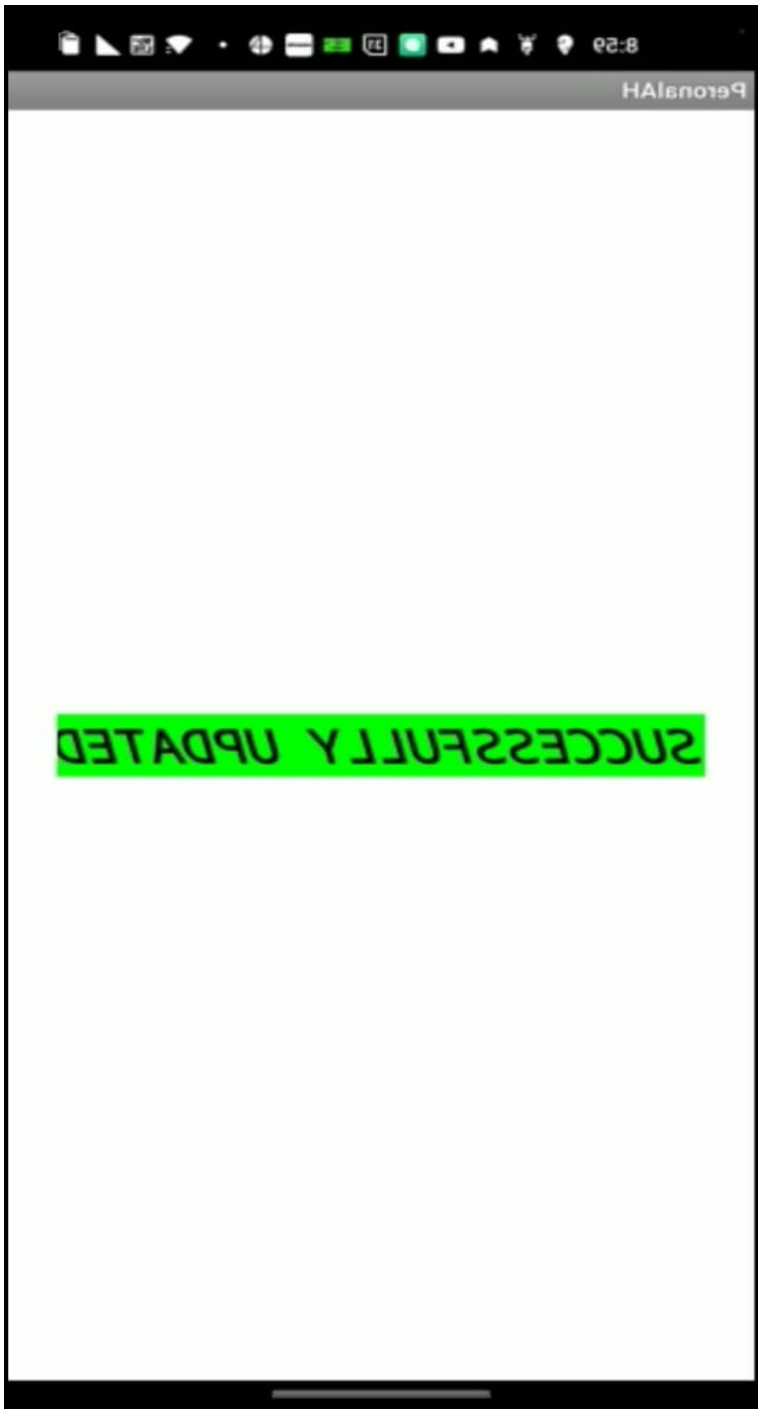


APP DEVELOPMENT AND GETTING REAL TIME DATA IN FIRE BASE

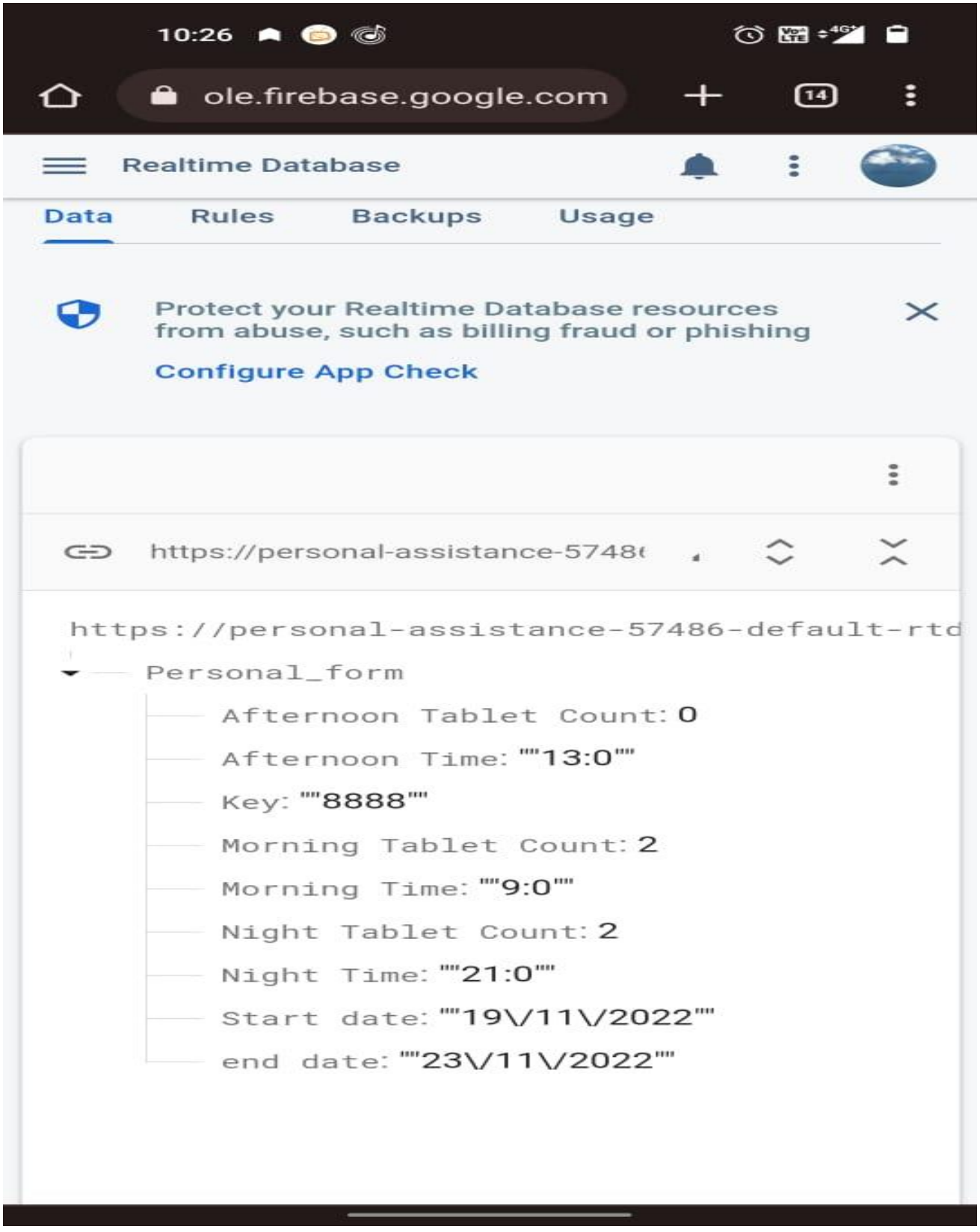
UPDATE THE MEDICINE DETAILS



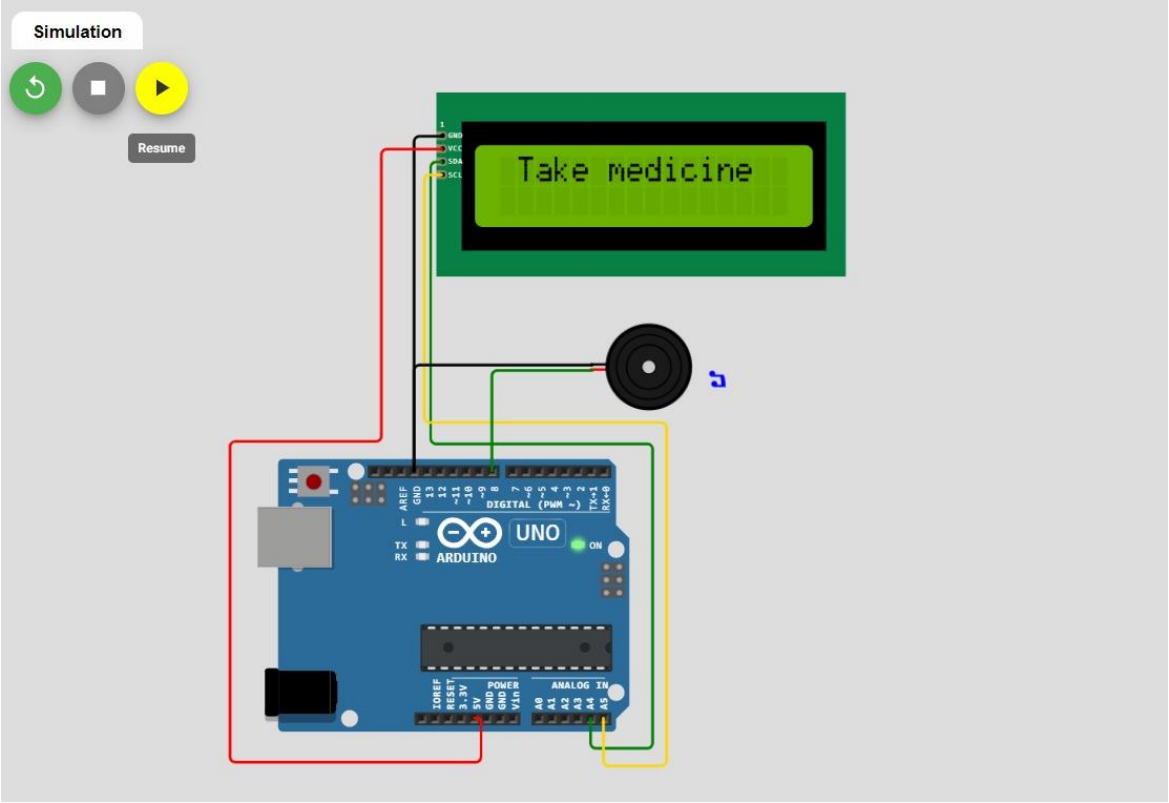
SUCCESSFULLY DETAILS ARE UPDATED



UPDATING THE DATA IN REALTIME



GETTING OUTPUT FROM WOKWI



Getting the real time data by RTC and the correct time the buzzer will on and it will display message to take medicine after that buzzer will off and it will wait for next reminder to update

10. ADVANTAGES & DISADVANTAGES

ADVANTAGES

1. Receive reminders to take your medications you can get timely reminders for taking your medicine regularly.

2. Set personal goals for improving your health With this system , you can track and compare your medicine history and health status over a period of time

3. Provides Security for your health information your personal health information is protected and will not be redistributed or resold.
4. No need of caretakers to take care
5. Easy to analyse the current health condition of the patient

DISADVANTAGES

1. Elderly people may be not familiar
2. This system will not contain any information about the patients health parameters
3. Have to update over a period of time if the medicine finished
4. If the patients have sudden issues on their health condition it will not remind medicine based on that

11. CONCLUSION

Many Medication Reminder Systems have been developed on different platforms. Many of these systems require special hardware devices to remind the patients about the medicine in-take timings. Purchasing new hardware devices becomes costly and more time and money consuming. So in the given work an attempt has been made to implement a system which is economical, easily accessible and improves medication adherence. Medication non-adherence reduces the effectiveness of a treatment and imposes a financial burden on health care systems .The patients will get the schedule of medicine in-take time with medicine description, starting and ending date of medicine, notification through message , automatic voice notification system The scheduled reminder will not suggest any kind of medicine which is not prescribed by the doctor that will assure the safety of the patient and also will avoid wrong dosages. This medicine reminder system is enables him to set the desired time and medicine. These details will be stored in the IBM Cloud . If the medicine time arrives the web application will send the medicine name to the IoT Device through the IBM IoT platform. The device will receive the medicine name and notify the user with voice commands.

12. FUTURE SCOPE

Plan to focus on improving the overall performance of the system. Also, interaction between patients and doctors through video calling and secure prescription will be focused upon. Some more ways to achieve medication adherence will be focused. Feature of adding patients health parameters like bp, sugar level ,heart rate ect. Feature to add notification send to their relatives when the patients health condition in risk

13. APPENDIX SOURCE CODE GITHUB & PROJECT DEMO LINK

GITUP LINK

https://github.com/IBM-EPBL/IBM-Project-36752-1660297506

DEMO VEDIO LINK M- YOUTUBE LINK

https://youtu.be/w1XIoJwbbTE