PERSONAL ASSISTANT FOR SENIORS WHO ARE SELF RELIANT

IBM PROJECT REPORT

Team ID - PNT2022TMID14855

SUBMITTED BY

DHARSHINI.A.G (111519106025)

MAHALAKSHIMI.C (111519106080)

EMMIDI MADHURI (111519106032)

EVANGELINE MERCY (111519106034)

Personal Assistance for Seniors Who Are Self-Reliant

Project Overview:

Introduction

- 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 Cloudant DB.
- 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.

Purpose

- 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. **1. Literature survey**

Existing problem

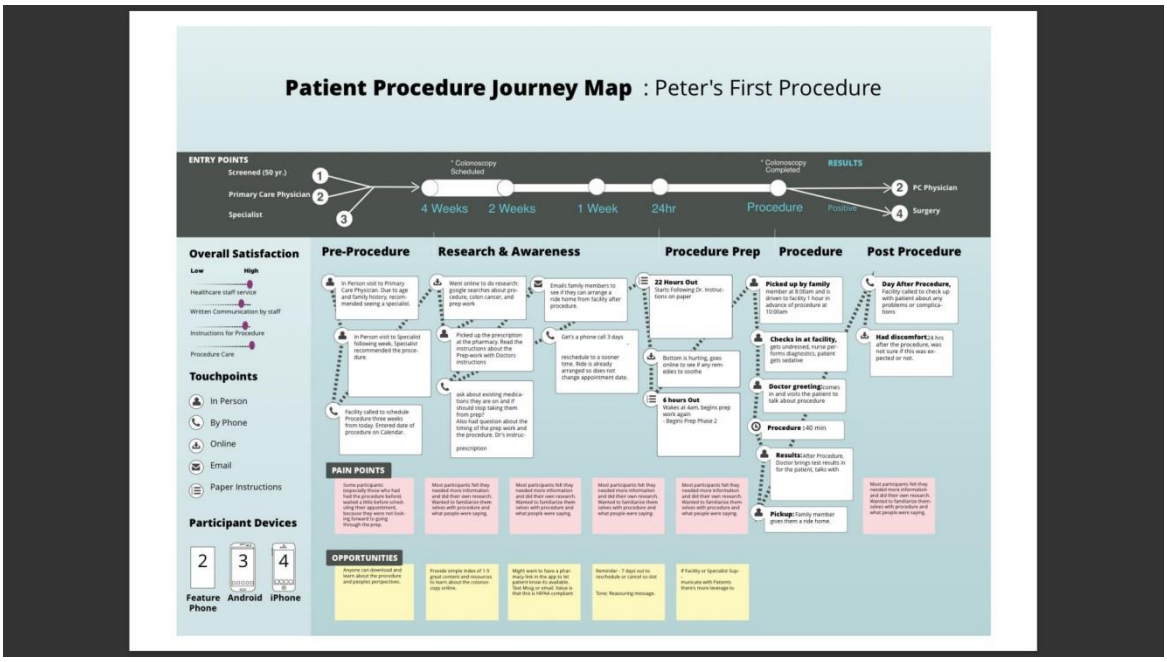
Elderly people let slip the medications at the correct time and the existing solutions for this problem is setting reminders or using pill boxes, calendars, Personal Assistance. Though the solutions give reminders, the voice commands or assistance given by this system is more efficient.

Problem statement definition

Skipping medicines can be serious for some medical health conditions; Sometimes elderly people forget to take their medicine at the correct time.They also forget which medicine one should take at that particular time. And it is difficult for doctors/caretakers to monitor the patients around the clock.

2. Ideation and proposed solution

Empathy Map Canvas



Define CS, fit into CC

1. CUSTOMER SEGMENT(S) CS

Our customers are people who require medical support; Also, our alert system can be used in hospitals and old age homes where people will require medical assistance.

6. CUSTOMER CONSTRAINTS CC

Healthcare costs, lack of financial support, Difficulty with everyday tasks and mobility, Finding the right care provision and seclusion.

5. AVAILABLE SOLUTIONS AS

The existing solutions for this project is setting reminders or using pill boxes, calendar, Personal Assistance. Though, the solutions give reminders, the voice commands or assistance given by this system is more efficient.

Explore AS, differentiate

Focus on J&P, tap into BE, understand RC

2. JOBS-TO-BE-DONE / PROBLEM J&P

Skipping of medicines can be serious for some medical health conditions; in such cases this system would help the individual to take their medication on time.

9. PROBLEM ROOT CAUSE RC

Sometimes elderly people forget to take their medicine at the correct time.They also forget which medicine one 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.

7. BEHAVIOUR BE

Directly related:To download the web application so that the user can receive voice notifications on the connected IoT device. Through this application, the user can set the details of the medicine name and other details.

Indirectly associated: Customers can be carefree and don't need a person round the clock to check on them.

understand RC

Identify strong TR and EM

3. TRIGGERS TR

There are applications which already exist that give regular reminders to take medicines.

4. EMOTIONS: BEFORE / AFTER EM

With this application built, which gives voice commands and alerting system which is more efficient in helping the elderly to take their medicines on time and can be carefree.

10. YOUR SOLUTION SL

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 Cloudant DB. 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.

8. CHANNELS of BEHAVIOUR CH

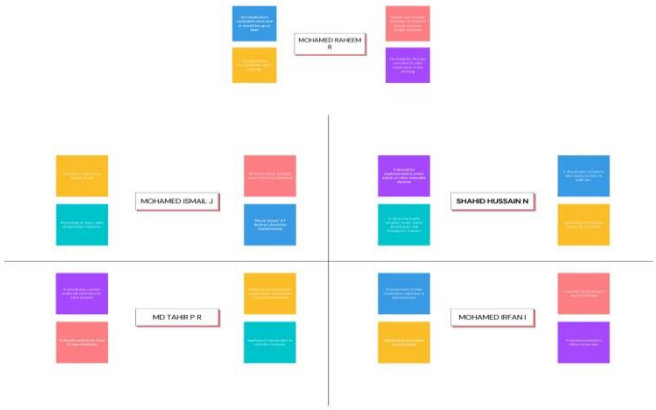
8.1 ONLINE

The customers should have the mobile application on their devices so that they can get regular voice commands.

8.2 OFFLINE

The customer should have the device or mobile near them. Also , the customer should update the schedule.

Ideation and Brainstorming



Problem Solution fit

Proposed solution

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	Sometimes elderly people forget to take their medicine at the correct time. They also forget which medicine should be taken at that particular time. And it is difficult for doctors/caretakers to monitor the patients around the clock.
2.	Idea / Solution description	➢ A 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 Cloudant DB. 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.
3.	Novelty / Uniqueness	➢ Keeping track of the medicines taken by the user at each time interval. ➢ Information is stored in the secured IBM cloud.
4.	Social Impact / Customer Satisfaction	The reminder system enables the user to take tablets at regular intervals prescribed by the physicians.

5.	Business Model (Revenue Model)	**Direct Mode:** We gain revenue from selling the medical reminder system to hospitals, medical health centres and even in old age homes. **Indirect Mode:** We gain profit by having partnership with pharmaceutical companies.
6.	Scalability of the Solution	The medical alert system can be used in hospitals, medical health centres and even in old age homes for dispensing medicines.

3. Requirement analysis Functional Requirements:

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	User Registration	Registration through Form Registration through Gmail
FR-2	User Confirmation	Confirmation via Email Confirmation via OTP
FR-3	Access Cloud services	Accessing the cloud service with correct credentials. Storing the details in the cloud database.
FR-4	IOT configuration	Fine Tuning the IOT device based Cloud DB access via device. Manage the data request and response effectively

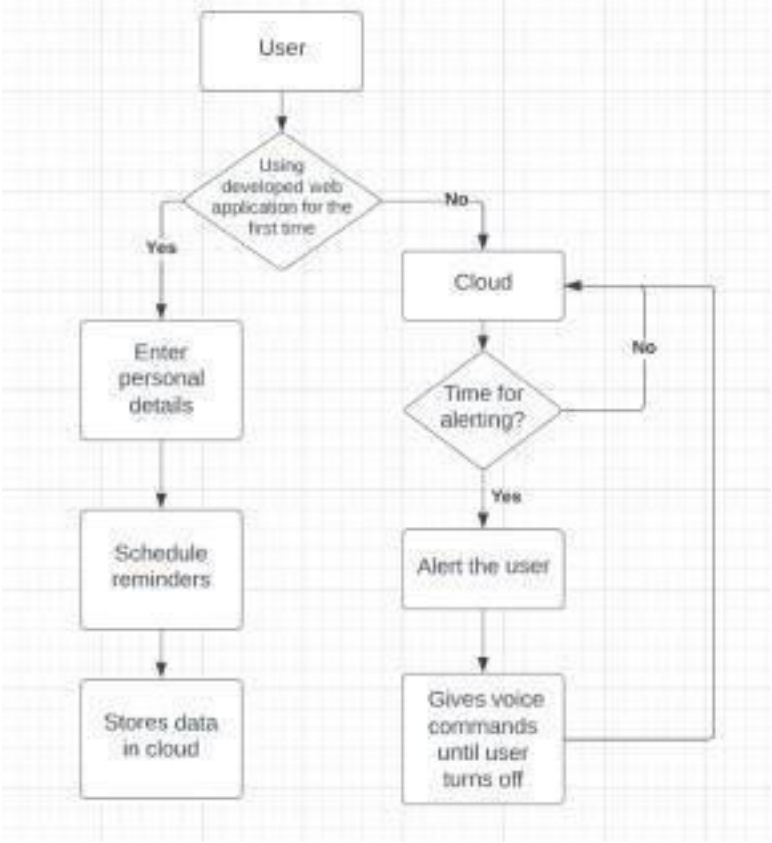
Non-functional Requirements:

FR No.	Non-Functional Requirement	Description

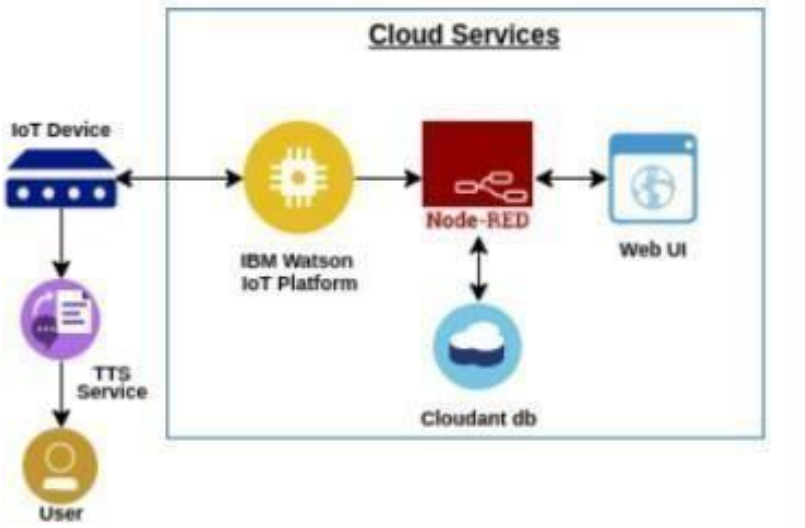
NFR-1	**Usability**	App can be used by anyone who has knowledge about applications and computers.
NFR-2	**Security**	For security, TFA is enabled and biometrics are also added for user safety.
NFR-3	**Reliability**	Highly reliable since, It uses trusted and authentic cloud services like IBM
NFR-4	**Performance**	Performance is better compared to other marketproducts.
NFR-5	**Availability**	Available on mobile app.
NFR-6	**Scalability**	Using Cloud services, makes the scalability higher the using traditional locally stored database.

4. Project Design

Data Flow Diagrams



Technical architecture



User Stories

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release

Customer (Mobile user)	Registration	USN1	As a user, I can register for the application by entering my email or mobile number, password, and confirming my password.	I can access my account / dashboard	High	Sprint-1
Confirmation mail	Gmail	USN2	As a user, I will receive confirmation email once I have registered for the application	I can receive confirmation email & click confirm	High	Sprint-1
Accessing	Link	USN3	As a user, I can register for the application through Gmail		Medium	Sprint-1
Medicine Name	Login	USN4	As a user, I can log into the application by entering email or mobile	I can access my account /	High	Sprint-1

Entering Credentials	Medicine Name	USN -5	number & password	dashboard	High	Sprint-1

User Type	Functional Requirem e nt (Epic)	User Story Num ber	User Story / Task	Acceptanc e criteria	Priorit y	Release
	Dashboard	USN5	As a user, I can update my reminders and medicines wherever required		High	Sprint-2
		USN-6	As a user, I can check the application whether the medicine dosage is completed.		Mediu m	Sprint-2
Customer Care Executiv e		USN-7	For any troubleshooting, the user can send a mail to the technical team.		Low	
Administ rator		USN-8	Ensures smooth functioning and data warehousing strategies		Mediu m	Sprint-3

5. Project Planning and Scheduling Sprint Planning and Estimation

Sprint	Functional Requiremen t (Epic)	User Story Number	User Story / Task	Stor y Poin ts	Priori ty	Team Members
Sprint1	Registratio n	USN-1	As a user, I can register for the application by entering my email or mobile number, password, and confirming my password.	2	High	Dharshini
Sprint1		USN-2	As a user, I will receive confirmation email once I have registered for the	1	High	Emmidi madhuri

			application			
Sprint-		USN-3	As a user, I can register	2	Mediu	Mahalakshimi

1			for the application through Gmail		m	
Sprint-1		USN-4	As a user, I can log into the application by entering email or mobile number & password	2	High	Evangeline Mercy
Sprint-2	Login	USN-5	As a user, I can update my reminders and medicines wherever required	1	High	Dharshini
Sprint-2	Dashboard	USN-6	As a user, I can check the application whether the medicine dosage is completed	1	Medium	Mahalakshimi
		USN-7	For any troubleshooting, the user can send a mail to the technical team	1	Low	Emmidi Madhuri
Sprint-3		USN-8	Ensures smooth functioning and data warehousing strategies	1	Medium	Evangeline Mercy

Sprint Delivery Schedule

Sprint	**Total Story Points**	**Duration Sprint Start Date Sprint End Date (Planned)**	**Story Points Completed (as on Planned End Date)**	**Sprint Release Date (Actual)**
Sprint-1	20	10 Days 25 Oct 2022 04 Nov 2022	20	04 Nov 2022
Sprint-2	20	5 Days 5 Nov 2022 10 Nov 2022	20	10 Nov 2022

Sprint-3	20	5 Days 11 Nov 2022 15 Nov 2022	20	16 Nov 2022
Sprint-4	20	2 Days 17 Nov 2022 18 Nov 2022	20	18 Nov 2022

6. Coding and Solutioning

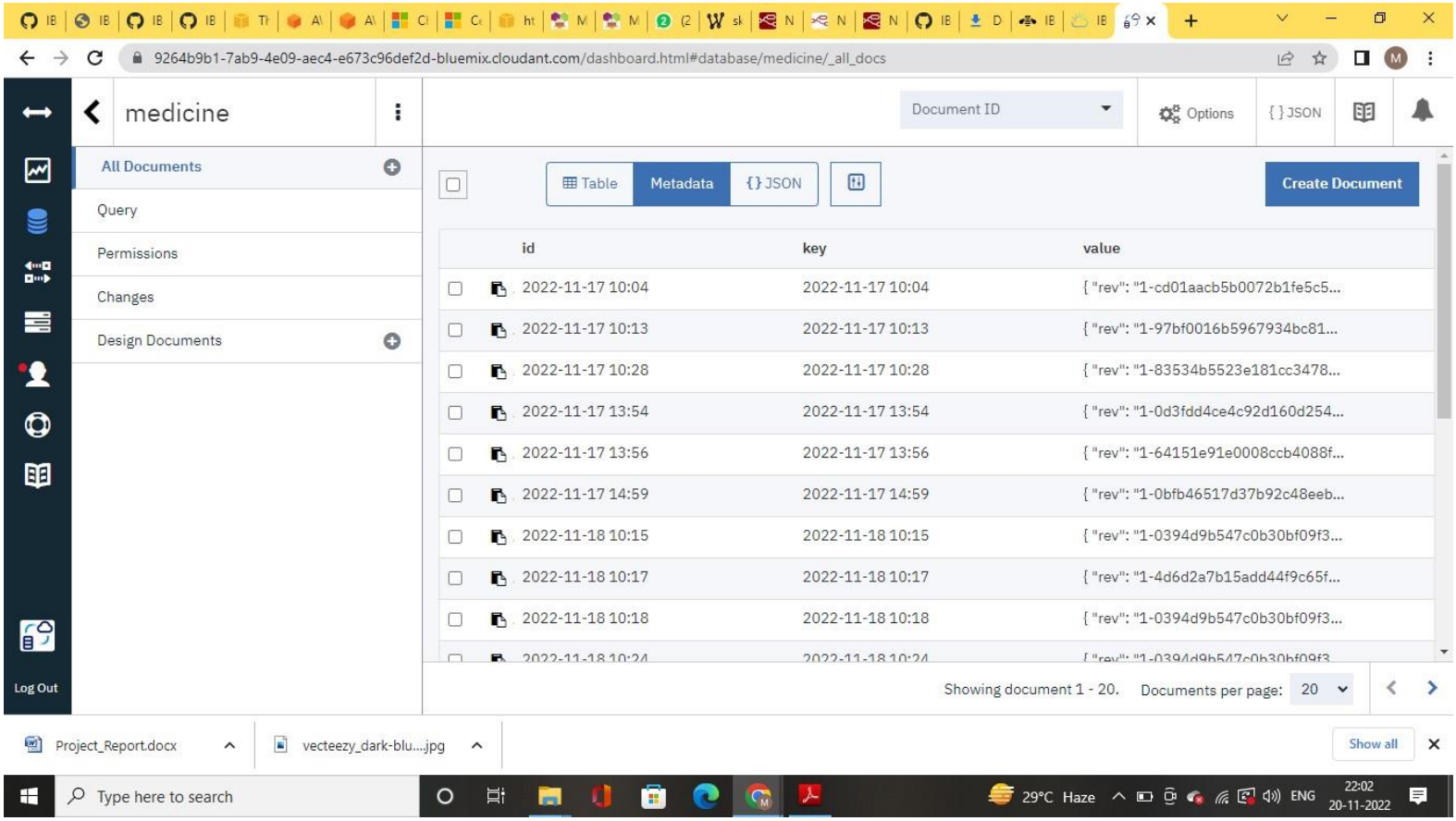
Feature 1

 The mobile application developed has a feature of individual login by different users.

Feature 2

 The mobile application also has the feature of uploading medicine names in the cloud.

Feature 3



The project includes a cloud database system.

7. Testing

Test cases

Test case	Precondition	Test steps	Expected result
Verify login with valid credentials	User should have a network Connection	1. Launch URL 2. Enter valid username. 3. Enter valid password. 4. Click on the “Login” button.	Users should be able to login successfully.
Verify login with	User should	1. Launch URL	Users should

invalid credentials	have a network Connection	2. Enter valid username. 3. Enter invalid password. 4. Click on the “Login” button.	not be able to login.
Update the medicine name with the time.	User should have a network Connection	1. Enter valid medicine name. 2. Enter the time when the medicine has to be consumed. 3. Click on the “Submit” button.	Users should be able to update it successfully.

User acceptance testing

 **Login page testing** **Incorrect login attempt Medicine page testing**

8. Results Performance Metrics

S. NO	Parameter	Performance
1.	Response Time	0.2s (Average of 10 trials)
2.	Workload	500 users (Calculated based on Cloud Space)
3.	Revenue	Individual users and pharmaceutical industries.
4.	Efficiency	Simple and straightforward workflow, which makes the process efficient.
5.	Down Time	Almost no down time due to IBM Cloud enabled solution.

9. Advantages and Disadvantages

Advantages

- Help the elderly people to take their medicine at the correct time.
- Avoid personal assistants or caretakers needed for medically sick people.
- Cost efficient.
- Can store multiple data and many notifications can be generated.
- Since it includes voice assistance, even blind people can use our device.

Disadvantages

- Makes people lethargic and makes them dependent always on others.
- Requires a stable internet connection.

10. Conclusion

The project offers the elderly or medically sick people a personal assistant which reminds them of the medicines to be consumed at the particular time. Skipping tablets may lead to serious problems if the person has a severe illness and this can be avoided. Since the cloud is integrated with the mobile application, numerous data can be fed into the database and

notifications can be generated. The mobile application developed is highly customisable by the user and easy to use.

11. Future Scope

The project can be further developed by bringing into the feature of informing the medicine name during the notification. The voice assistance which is given can be customized by adding the user's voice or the caretaker's voice. Further the mobile application can update medicines by taking voice commands as an input from the user.

12. Appen

dix Source Code:

```
#include <WiFi.h>//library for wifi
#include <PubSubClient.h>//library for MQtt
#include "SoundData.h"
#include "XT_DAC_Audio.h"
XT_Wav_Class
Sound("voice_command.wav");
XT_DAC_Audio_Class DacAudio(2,0);
uint32_t DemoCounter=0;


void callback(char* subscribetopic, byte* payload, unsigned int payloadLength);

//-------credentials of IBM Accounts------

#define ORG "9a7os9"//IBM ORGANITION ID
#define DEVICE_TYPE " ESP "//Device type mentioned in ibm watson IOT Platform
#define DEVICE_ID " ESP32"//Device ID mentioned in ibm    IOT Platform #define
TOKEN " LC!x?+V9etumdVMaSR "       //Token
String data3;
float h, t;


//-------- Customise the above values --------char server[] = ORG
".messaging.internetofthings.ibmcloud.com";// Server Name char publishTopic[] = "iot-
2/evt/Data/fmt/json";// topic name and type of event perform and format in which data to
be send
```

```
char subscribetopic[] = "iot-2/cmd/test/fmt/String";// cmd        REPRESENT command type
AND COMMAND IS TEST OF FORMAT STRING
char authMethod[] = "use-token-auth";// authentication method
char token[] = TOKEN;
char clientId[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID;//client id


//_____________________________________
WiFiClient wifiClient; // creating the instance for wificlient
PubSubClient client(server, 1883, callback ,wifiClient); //calling the predefined client id by
passing parameter like server id,portand wificredential void setup()// configureing the
ESP32
{
  Serial.begin(115200);

  delay(10);
  Serial.println();
  wificonnect();
  mqttconnect();
}

void loop()// Recursive Function
{




  delay(1000); if
  (!client.loop())
    { mqttconnect();
  }
}



/*.....................................retrieving to Cloud                          */


void mqttconnect() {
  if (!client.connected())
```

```
    { Serial.print("Reconnecting client to ");
    Serial.println(server); while (!!!client.connect(clientId,
    authMethod, token)) { Serial.print("."); delay(500);
    }

     initManagedDevice()
     ; Serial.println();
  }
}
void wificonnect() //function defination for wificonnect
{
  Serial.println();
  Serial.print("Connecting to ");

     WiFi.begin("Wokwi-GUEST", "", 6);//passing the wifi credentials to establish the
connection
  while (WiFi.status() != WL_CONNECTED)
    { delay(500);
    Serial.print(".");
  }
  Serial.println("");
  Serial.println("WiFi
  connected"); Serial.println("IP
  address: ");
  Serial.println(WiFi.localIP());
}

void initManagedDevice() {
  if (client.subscribe(subscribetopic))
    { Serial.println((subscribetopic));
    Serial.println("subscribe to cmd OK");
  } else {
    Serial.println("subscribe to cmd FAILED");
  }
}

void callback(char* subscribetopic, byte* payload, unsigned int payloadLength)
{

  Serial.print("callback invoked for topic: ");
```

```
  Serial.println(subscribetopic);
  for (int i = 0; i < payloadLength; i++)
    { //Serial.print((char)payload[i]);
    data3 += (char)payload[i];
  }
```

Is

```
  Serial.println("data: "+ data3);
  if(data3=="announce")
  {
Serial.println(data3);
for(int i=0;i<5;i++){
DacAudio.F illBuffer();
if(Sound.Playing==false)
    DacAudio.Play(&Sound);
  Serial.println(DemoCounter++);
}
  }

  else
  {
    pass;

  }
data3="";


}
```

Git Hub Link :

Project Demo link :