PERSONAL ASSISTANCE FOR SENIORS WHO ARE SELF RELIANT

TEAM ID: PNT2022TMID29933

TEAM MEMBERS:

SURYA M
VENKATESH G
PRASANTH S
SABARINATH K

1. INTRODUCTION

➤ 1.1PROJECT OVERVIEW

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 voice commands. 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.

➤ 1.2 PURPOSE

- **The elderly people, memory loss persons and medicine taking persons are affected by this problem.**
- **Suppose, the person couldn't time. It gives decrease in life span and also give life-threatening problems.**
- **The problem is, the people forgetting the medicine at a correct time and also forget which medicine to take.**
- **This problem is overcome by introducing medicine reminder system, it is created by using python IDLE.**
- **When the system is not created, the people who taking medicine, affected by healthy. It causes his/her life duration in this world.**
- **An app is built for the user, the user desired time and medicine. It gives the command at a correct time.**
- **The command includes the medicine name and number of dose to take user.**

2.LITERATURE SURVERY

➤ 2.1 EXISTING PROBLEM

- 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 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.

➤ 2.2 REFERENCES

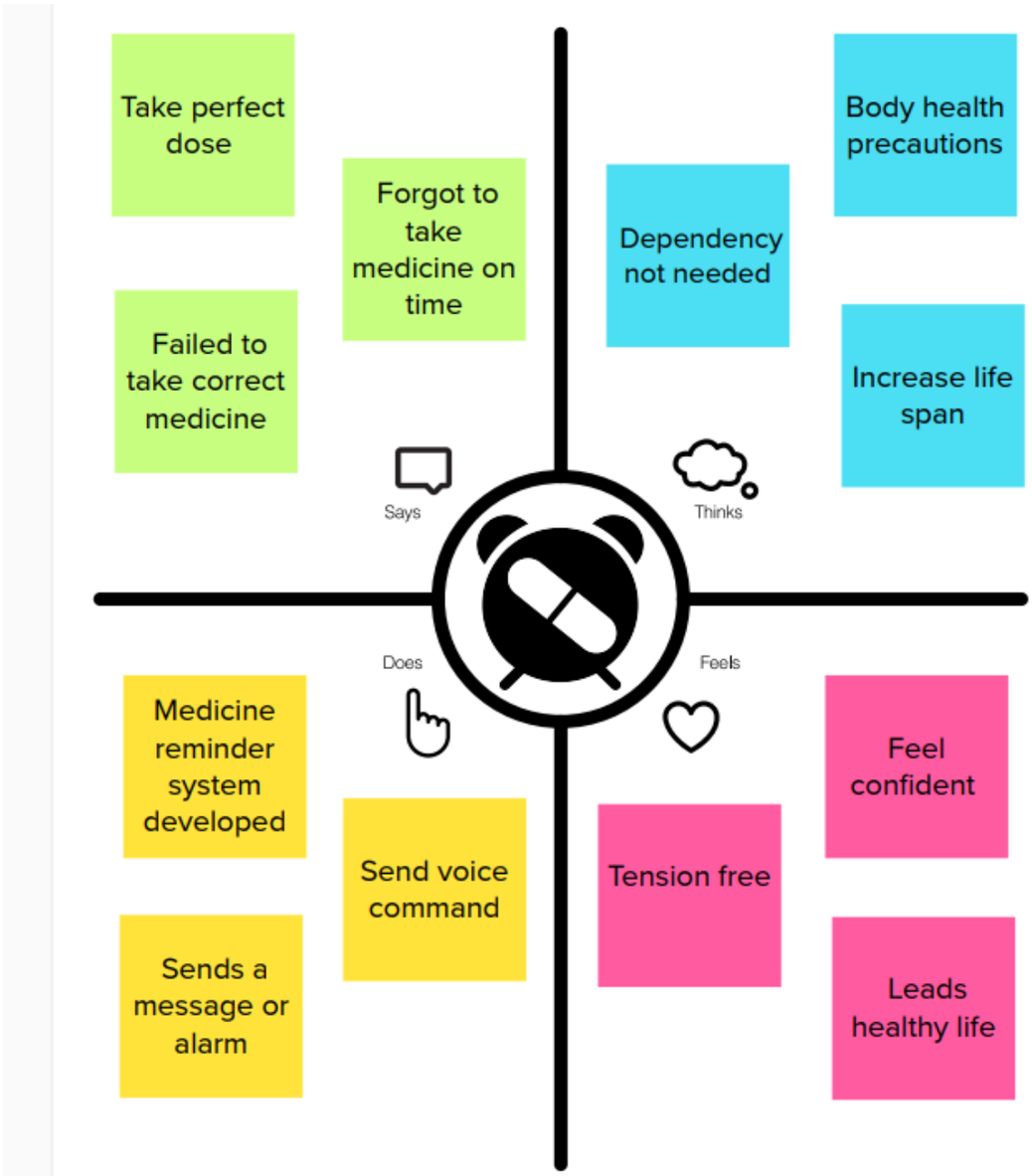
Paper Title	Author Name	Objectives
Voice Based Medicine Reminder Application for Elder People [March-2020]	M. Mehala J. Viji Gripsy	Voice based reminder notification alert can be set for multiple medicines and timings. The front end of the system is created by Xml, JAVA. The back end is created by using SQLite. Android Studio 2.3 is used as a software.
Development of an Android Based Medication Reminder and Adherence System. [Sept-2020]	Jenyo .I.A Adeyemi . T.O Amusan .E.A	This system is used to automatically see the list of patient connect to them and their chat message. It can send health tips and other broadcast messages. The application was implemented in Java while the user interface implemented using Extensible Markup Language (XML) using android studio.
A Case Study of Medication Reminder System [Oct-2019]	Mohammed Alhaj Abdullah Nizar	The case study of medication reminder system that helps to alert patients who forget to take their medicines at prescribed time. The software is design by Unified Modeling Language (UML). The hardware system is designed by using the products of PIC 18F452 microcontroller, DS1307 Serial real time clock, AGSM modem, keypad, Buzzer, LCD.
Medicine Reminder and Monitoring System for Scheme for Secure Health using IoT	Samir V. Zanjal Girish. R. Talmale	The monitoring system consists of an IoT enabled medication reminder system and it gives timely alerts for the patients about their medication time. The RFID based technology is used

2.3 PROBLEM STATEMENT DEFINITION

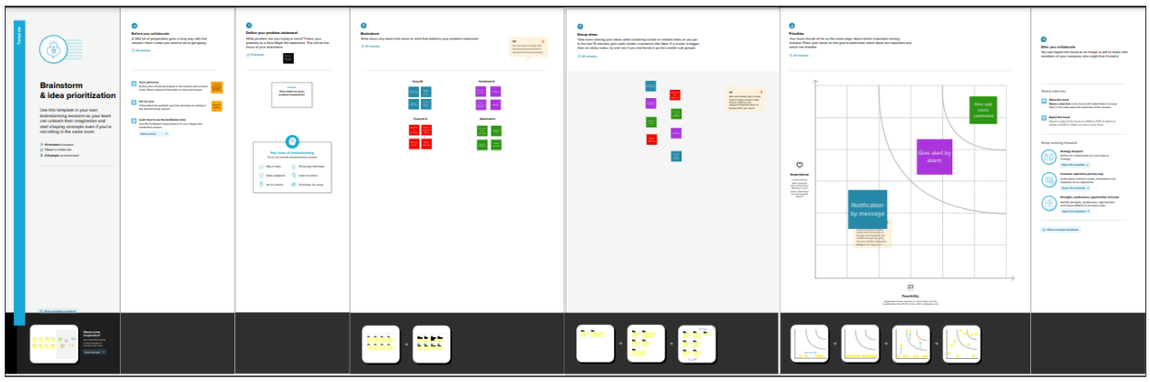
Problem statement	I am (customer)	I am trying to	But	Because	Which makes me feel
Forget to take medicine at correct time	I am patient	take a medicine at a correct time	I couldn't did this	I forget the medicine.	Some reminder is makes me feel better
Forget to take a perfect medicine and dose	I am patient	Take a perfect medicine and perfect dose	I couldn't did this	I forget to take perfect medicine and dose	Reminder should have a medicine name and dose number

3.IDEATION AND PROPOSED SOLUTION

➤ 3.1 EMPATHY MAP CANVAS



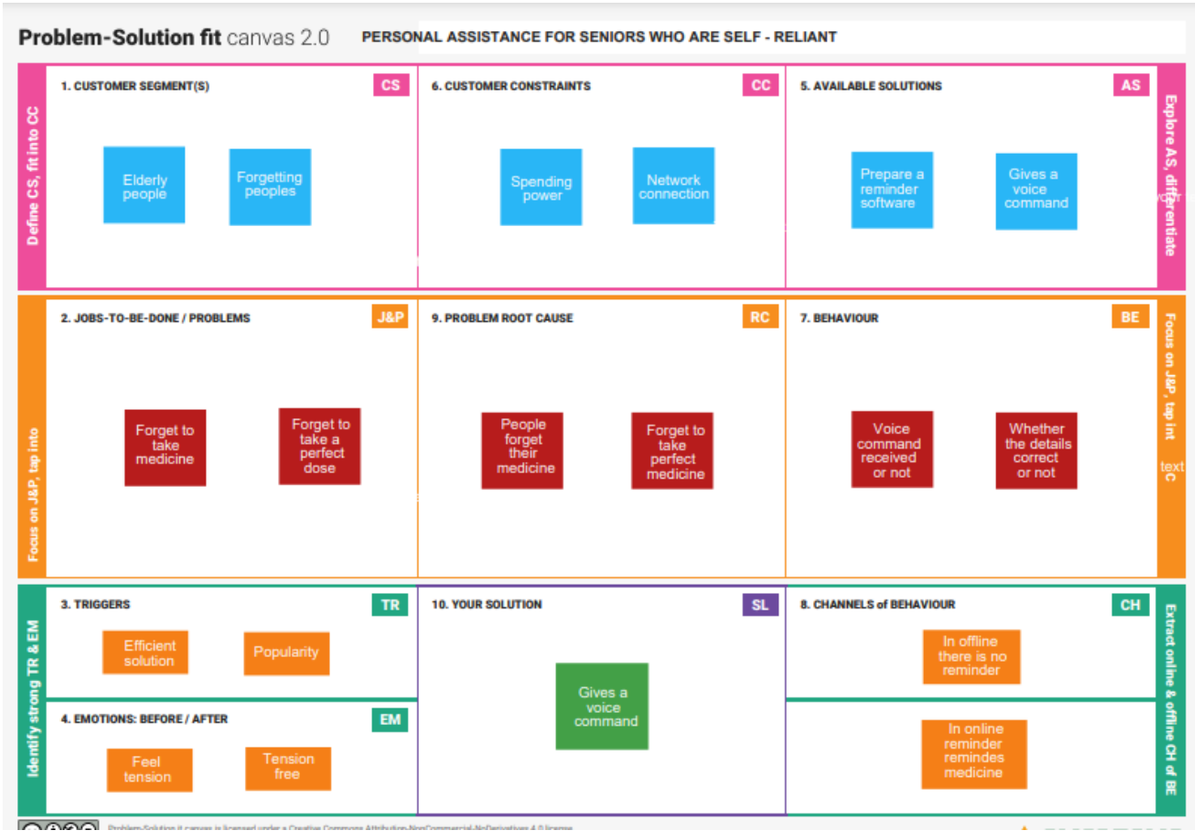
➤ 3.2 IDEATION & BRAINSTOMING



➤ 3.3 PROPOSED SOLUTION

S. No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	Forget to take medicine at a correct time and also forget to take which medicine to take.
2.	Idea / Solution description	This problem is overcome by introducing medicine reminder system, it is created by using python IDLE. When the system is not created, the people who taking medicine, affected by healthy. It causes his/her life duration in this world. An app is built for the user, the user desired time and medicine. It gives the voice command at a correct time. The voice command includes the medicine name and number of dose to take user.
3.	Novelty / Uniqueness	The previous applications give a message only. But we give a voice command.
4.	Social Impact/ Customer Satisfaction	The customers were highly satisfied by using this application. Because it is involved in our life. Its increase the life span.
5.	Business Model (Revenue Model)	• Cost efficiency. • Highly profitable.
6.	Scalability of the Solution	High scalability.

3.4 PROBLEM SOLUTION FIT



4. REQUIREMENT ANALYSIS

➤ 4.1 FUNCTIONAL REQUIREMENT

Functional Requirements:

Following are the functional requirements of the proposed solution.

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	User Registration	Registration through Form Registration through Gmail Registration through LinkedIn
FR-2	User Confirmation	Confirmation via Email Confirmation via OTP
FR-3	Reminds Medicine name	Send alert via application Message in the form of voice command
FR-4	Reminds number of doses and which medicine to take	Medicine and doses are alert at time by voice command Voice command send by application

4.2 NON FUNCTIONAL REQUIREMENT

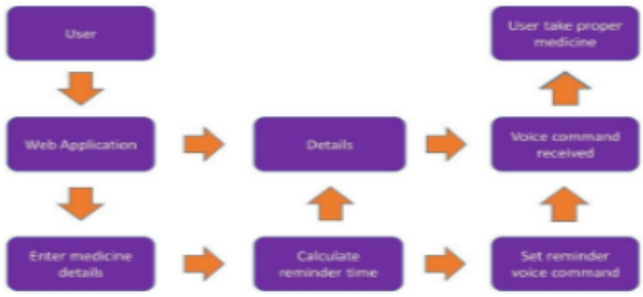
Non-functional Requirements:

Following are the non-functional requirements of the proposed solution.

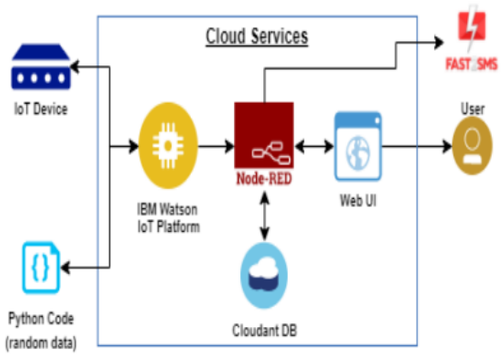
FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	Easy to use and efficient. Simply understand.
NFR-2	**Security**	Software is protected by unauthorized access. Data privacy and high security.
NFR-3	**Reliability**	Application runs accurately. High reliability to work in application.
NFR-4	**Performance**	No delay to give voice command. High responsiveness.
NFR-5	**Availability**	24*7 application is available. Available widely.
NFR-6	**Scalability**	Application is used unlimited users. High scalability.

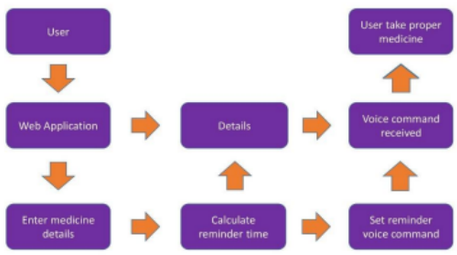
5.PROJECT ANALYSIS

5.1 DATA FLOW DIAGRAMS



5.2 SOLUTION & TECHNICAL ARCHITECTURER





➤ 5.3 USER STORIES

User Stories

Use the below template to list all the user stories for the product.

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer (Mobile user)	Registration	USN-1	As a user, I can register for the application by entering my email, password, and confirming my password.	I can access my account / dashboard	High	Sprint-1
		USN-2	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
		USN-3	As a user, I can register for the application through Facebook	I can register & access the dashboard with Facebook Login	Low	Sprint-2
		USN-4	As a user, I can register for the application through Gmail		Medium	Sprint-1
	Login	USN-5	As a user, I can log into the application by entering email & password		High	Sprint1
Customer (Web user)	Registration	USN-1	As a user, I can register for the application through google	I can access confirmation mail	High	Sprint1
		USN-2	As a user, I can register for the application through firebox	I can access confirmation Login	Low	Sprint-2
		USN-3	As a user, I can register for the application through Gmail		Medium	Sprint-1
Administrator	Registration	USN-1	As a user, I can register for the application through web application	I can access my account	High	Sprint-1
		USN-2	As a user, I can register for the application through Mobile app	I can access confirmation mail	Low	Sprint-2

6. PROJECT PLANNING & SCHEDULE

➤ 6.1 SPRINT PLANNING & ESTIMATION

Sprint	FunctionalRequirement(Epic)	User StoryNumber	UserStory/Task	StoryPoints	Priority	Team Member
Sprint-1	Registration via **App**	USN-1	As a user, I can register for the application by entering my email, password, and confirming my password.	3	High	Venkatesh G Surya M
Sprint-1	Registration via Facebook/ Gmail	USN-2	As a user, I can register for the application through Facebook	3	High	Prasanth S Sabarinath K
Sprint-1	Reminder System in the application	USN-3	As a user, Once after the registration, I can log in and set the reminder for Pills	3	High	Venkatesh G Surya M
Sprint-1	Login	USN-4	As a user, I can login to the application by entering email &password	1	Low	Prasanth S Sabarinath K
Sprint-2	CreateWebUI for **administrative purpose**	USN-5	To create WebUI for Admin Page to store Prescriptions along hospital side.	3	High	Venkatesh G Surya M

Sprint	FunctionalRequirement(Epic)	User StoryNumber	UserStory/Task	StoryPoints	Priority	TeamMember
Sprint-2	Registration viaFacebook/Gmail	USN-6	Asauser, I can Register through the facebook.	3	High	Prasanth S Sabarinath K
Sprint-2	Registration viaGmail	USN-7	As a user, I can Register through the Gmail.	3	High	Venkatesh G Surya M
Sprint-3	CreateIBMWatsonand devicesettings	USN-8	To create the IBM Watson IOT platform andIntegrate the microcontroller with it to sendsenseddatatocloud	3	High	Prasanth S Sabarinath K
Sprint-3	Create node redservice	USN-9	Store the data of the Prescription in the Web UI	3	High	Venkatesh G Surya M
Sprint-3	Text to speech	USN - 10	Integrating Text to Speech in the application or web application in the user end.	3	High	Prasanth S Sabarinath K
Sprint-4	A remainder to the TTS services	USN-11	The admin or hospital management send the alert message to the user and their guardian (family member) when the user did not consume pills on concern time.	3	High	Prasanth S Sabarinath K
Sprint-4	Alert Message	USN-12	The admin or hospital management send the alert message to the user and their guardian (family member) when the user did not consume pills on concern time.	3	High	Venkatesh G Surya M
Sprint-4	Testing	USN-13	Testing of project and final deliverables	3	High	Venkatesh G Surya M

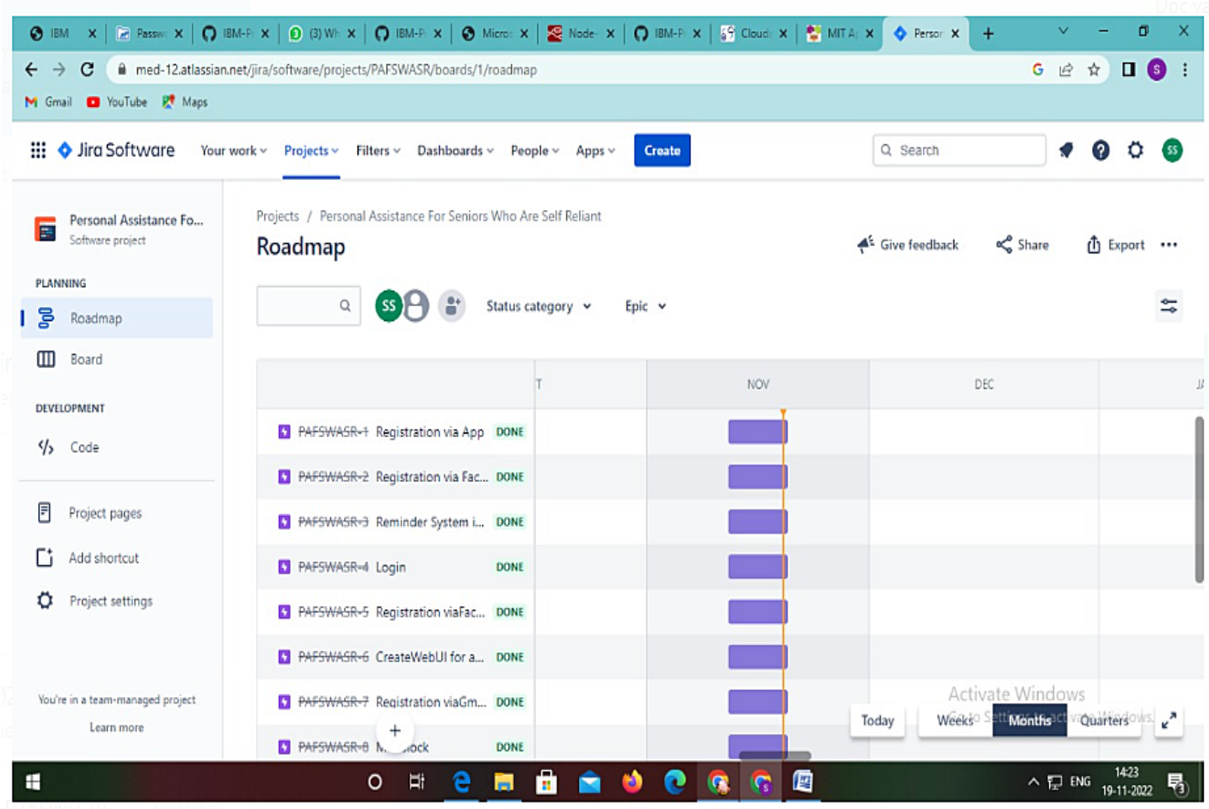
6.2 SPRINT DELIVERY SCHEDULE

Sprint	TotalStory Points	Duration	SprintStartDate	Sprint End Date(Planned)	Story PointsCompleted (as onPlannedEndDate)	Sprint Release Date(Actual)
Sprint-1	10	6Days	24Oct2022	29Oct2022	10	29 Oct 2022
Sprint-2	10	6Days	31Oct2022	05Nov2022	10	05Nov2022
Sprint-3	10	8Days	07 Nov2022	12Nov2022	10	12Nov2022
Sprint-4	10	6Days	14Nov2022	19Nov2022	10	19Nov2022

Velocity:
Imaginewehavea10-daysprintduration,andthevelocityoftheteamis20(pointspersprint).Let'scalculatetheteam'saveragevelocity(AV)periterationunit(storypointsperday)

$$AV = \frac{sprint\ duration}{velocity} = \frac{20}{10} = 2$$

➤ 6.3 REPORTS FORM JIRA



LINK FOR JIRA PLATFORM:

https://med-12.atlassian.net/jira/software/projects/PAFSWASR/boards/1/roadmap

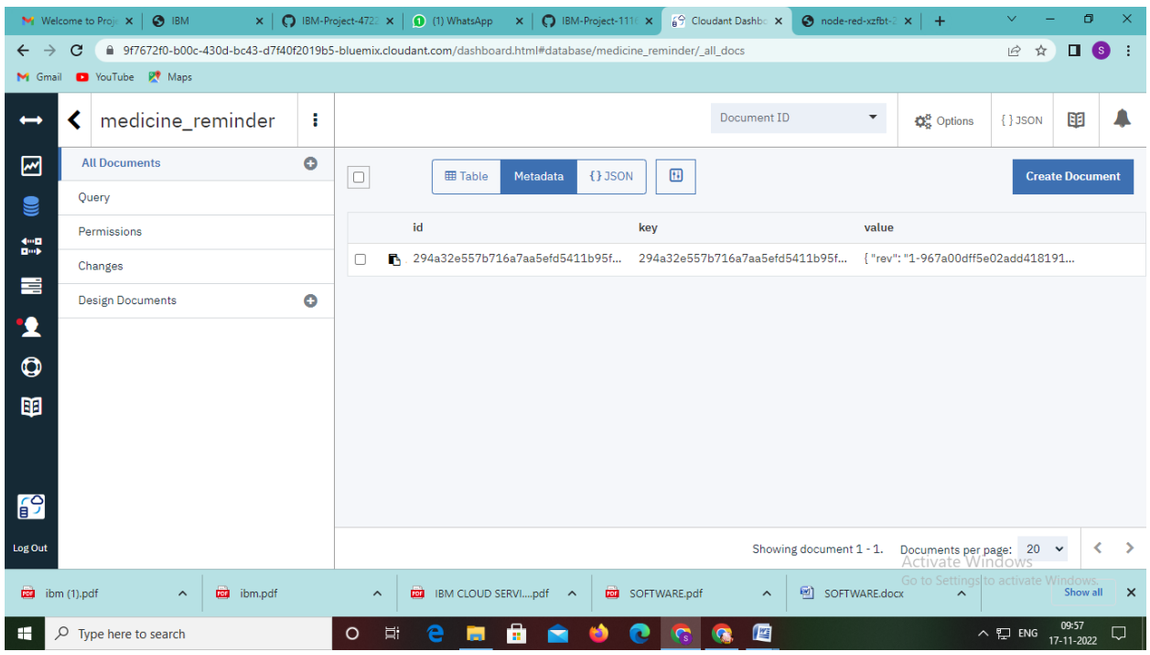
7.CODING & SOLUTIONING

7.1 FEATURE

```
import time
import sys
import ibmiotf.application
import ibmiotf.device
import random


#Provide your IBM Watson Device Credentials
organization = "ts2p3l"
deviceType = "medicine1-device_type."
deviceId = "PNT2022TMID29933-Medicine"
authMethod = "token"
authToken = "lq!RGKJdXNRjtvm0x2"

# Initialize GPIO


def myCommandCallback(cmd):
    print("Command received: %s" % cmd.data['command'])
    status=cmd.data['command']
    if status=="lighton":
        print ("led is on")
    else :
        print ("led is off")

    #print(cmd)




try:
        deviceOptions = {"org": organization, "type": deviceType, "id": deviceId, "auth-
method": authMethod, "auth-token": authToken}
        deviceCli = ibmiotf.device.Client(deviceOptions)
        #.....................................

except Exception as e:
        print("Caught exception connecting device: %s" % str(e))
        sys.exit()

# Connect and send a datapoint "hello" with value "world" into the cloud as an event of
```

```
type "greeting" 10 times
deviceCli.connect()



for i in range(0,1000):


    tablet=["Paracetamol","Aspirine","Azithral","Asthalin","Sinarest"]
    medicinetime=[12.00,1.00,2.00,3.00,5.00,18.00,20.00,7.00]
    name = "mani"
    medicine=random.choice(tablet)
    medicinetime=random.choice(medicinetime)
    mydata = {'Patient Name': name, 'Medicine Name': medicine, 'Time': medicinetime}
    def myOnPublishCallback():
            mydata = {'Patient Name': name, 'Medicine Name': medicine, 'Time':
medicinetime}
            print("Data published to IBM IOT platform :" ,mydata)
    success = deviceCli.publishEvent("IoTSensor", "json", mydata, qos=0,
on_publish=myOnPublishCallback)

    if not success:
        print("Not connected to IoTF")
    time.sleep(1)

    deviceCli.commandCallback = myCommandCallback

# Disconnect the device and application from the cloud
deviceCli.disconnect()
```

7.2 Database


9f7672f0-b00c-430d-bc43-d7f40f2019b5-bluemix.cloudant.com/dashboard.html#database/medicine_reminder/_all_docs
medicine_reminder
All Documents
Query
Permissions
Changes
Design Documents
Log Out
Document ID
Options
{ } JSON
Table
Metadata
{ } JSON
Create Document
id
key
value
294a32e557b716a7aa5efd5411b95f...
294a32e557b716a7aa5efd5411b95f...
{ "rev": "1-967a00dff5e02add418191...
Showing document 1 - 1.
Documents per page: 20
Activate Windows
Go to Settings to activate Windows.
ibm (1).pdf
ibm.pdf
IBM CLOUD SERVI....pdf
SOFTWARE.pdf
SOFTWARE.docx
Show all
Type here to search
ENG
09:57
17-11-2022

8. TESTING:

8.1 TEST CASES:

TESTCASE 1:

TO DO: Give alert message with medicine name.

OUTPUT: When a time is arrived, application gives the alert message.

TESTCASE 2:

TO DO: Calculate a correct time and give alert message.

OUTPUT: When a time is arrived, application gives the alert message.

8.2 USER ACCEPTANCE TESTING:

1. Purpose of Document

The purpose of this document is to briefly explain the test coverage and open issues of the [Product Name] project at the time of the release to User Acceptance Testing (UAT).

2. Defect Analysis

This report shows the number of resolved or closed bugs at each severity level, and how they were resolved

Resolution	Severity 1	Severity 2	Severity 3	Severity 4	Subtotal
By Design	20	4	2	3	29
Duplicate	3	0	3	0	7
External	2	3	0	1	6
Fixed	11	2	4	20	37
Not Reproduced	0	0	1	0	1
Skipped	0	0	1	1	2
Won't Fix	0	5	2	1	8
Totals	36	14	13	26	86

3. Test Case Analysis

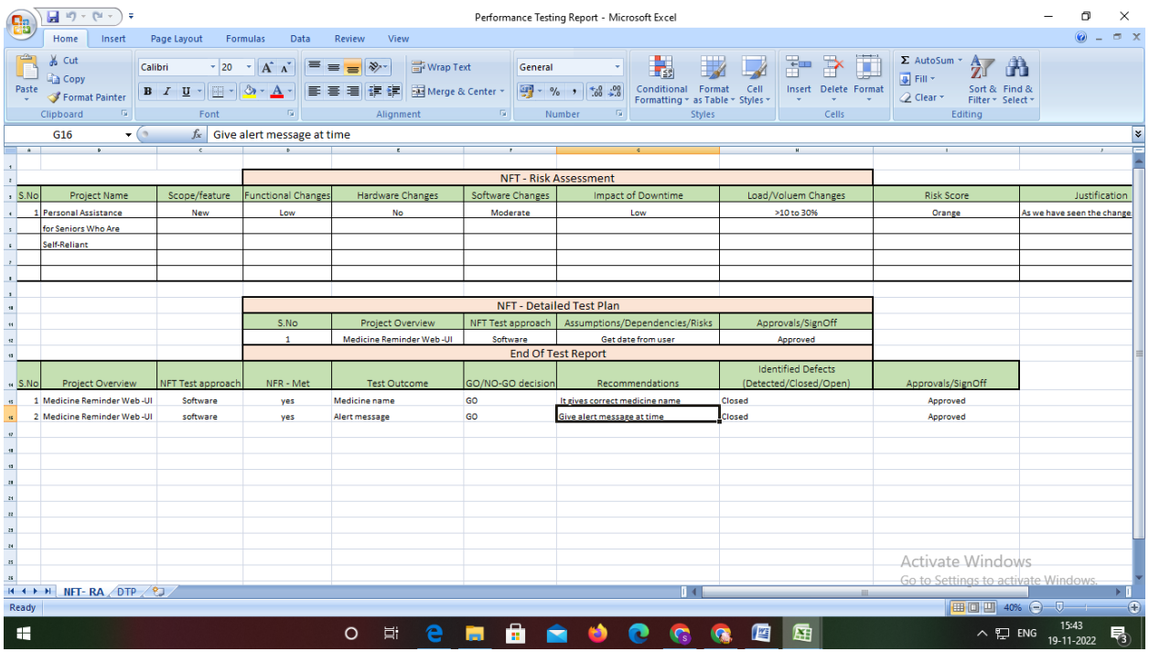
This report shows the number of test cases that have passed, failed, and untested

Section	Total Cases	Not Tested	Fail	Pass
Print Engine	7	0	0	7
Client Application	11	0	0	11
Security	0	0	0	0
Outsource Shipping	3	0	0	3
Exception Reporting	9	0	0	9
Final Report Output	4	0	0	4
Version Control	2	0	0	2

9. RESULTS

MIT APP INVERTER RESULT

9.1 PERFORMANCE METRICES:



Performance Testing Report - Microsoft Excel

G16 | Give alert message at time

NFT - Risk Assessment

S.No	Project Name	Scope/feature	Functional Changes	Hardware Changes	Software Changes	Impact of Downtime	Load/Voluem Changes	Risk Score	Justification
1	Personal Assistance for Seniors Who Are Self-Reliant	New	Low	No	Moderate	Low	>10 to 30%	Orange	As we have seen the change

NFT - Detailed Test Plan

S.No	Project Overview	NFT Test approach	Assumptions/Dependencies/Risks	Approvals/SignOff
1	Medicine Reminder Web -UI	Software	Get date from user	Approved

End Of Test Report

S.No	Project Overview	NFT Test approach	NFR - Met	Test Outcome	GO/NO-GO decision	Recommendations	Identified Defects (Detected/Closed/Open)	Approvals/SignOff
1	Medicine Reminder Web -UI	Software	yes	Medicine name	GO	It gives correct medicine name	Closed	Approved
2	Medicine Reminder Web -UI	software	yes	Alert message	GO	Give alert message at time	Closed	Approved

Activate Windows
Go to Settings to activate Windows.

NFT- RA | DTP | Ready | 40% | ENG | 15:43 | 19-11-2022

10 .Advantages and Disadvantages

10.1 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.

10.2 Disadvantages

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

11. 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 maylead 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 customizable by the user and easy to use.

12. FutureScope

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.

13 APPENDIX:

SOURCE CODE:

PYTHON CODE:

```
import time
import sys
import ibmiotf.application
import ibmiotf.device
import random
#Provide your IBM Watson Device Credentials
organization = "ts2p3l"
deviceType = "medicine1-device_type."
deviceId = "PNT2022TMID29933-Medicine"
authMethod = "token"
authToken = "lq!RGKJdXNRjtvm0x2"
# Initialize GPIO
def myCommandCallback(cmd):
    print("Command received: %s" % cmd.data['command'])
    status=cmd.data['command']
    if status=="lighton":
        print ("led is on")
    else :
        print ("led is of
 #print(cmd)
try:
        deviceOptions = {"org": organization, "type": deviceType, "id": deviceId, "auth-method": authMethod, "auth-token":
```

```
authToken}
        deviceCli = ibmiotf.device.Client(deviceOptions)
        #.....................................
        except Exception as e:
        print("Caught exception connecting device: %s" % str(e))
        sys.exit()
# Connect and send a datapoint "hello" with value "world" into the cloud as an event of type "greeting" 10 times
deviceCli.connect()
for i in range(0,1000):
tablet=["Paracetamol","Aspirine","Azithral","Asthalin","Sinarest "]
    medicinetime=[12.00,1.00,2.00,3.00,5.00,18.00,20.00,7.00]
    name = "mani"
    medicine=random.choice(tablet)
    medicinetime=random.choice(medicinetime)
      mydata = {'Patient Name': name, 'Medicine Name': medicine, 'Time': medicinetime}
    def myOnPublishCallback():
                    mydata = {'Patient Name': name, 'Medicine Name': medicine, 'Time': medicinetime}
          print("Data published to IBM IOT platform :" ,mydata)
     success = deviceCli.publishEvent("IoTSensor", "json", mydata, qos=0, on_publish=myOnPublishCallback)

    if not success:
        print("Not connected to IoTF")
                            time.sleep(1)deviceCli.commandCallback = myCommandCallback
```

 # Disconnect the device and application from the cloud

deviceCli.disconnect()import time

GITHUB LINK

https://github.com/IBM-EPBL/IBM-Project-35582-1660286348

DEMONSTRATION LINK