PROJECT REPORT
PERSONAL ASSISTANCE FOR SENIORS WHO ARE SELF-RELIANT

TABLE OF CONTENTS

1.INTRODUCTION

1.1 PROJECT OVERVIEW

Internet of things(IoT) is a sensory-based technology that can connect a few objects to the internet. This connection can provise us with daily tasks.In our project,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.

1.2 PURPOSE

In today's world, people are constantly busy with their own lives. There may be many people out there who need help all the time - like our older people, family members, those with special needs. Some people may forget to take their medication on time and may need to take medication. In order to eliminate this problem pill boxes are already available but many of them are less practical, not suitable for adults.The purpose of this study is to develop a Medicine Reminder System. Once the medicine period has been set, the system will remind patients to take the pills using voice commands. The tablet alert should be taken and notify by the android app kept by the patient. Compared to a regular pill box that requires clients or assistants to plan a daily or routine schedule. This model can help adults to take their medication. IoT can be a powerful and efficient paradigm for storing data collected by cloud devices. In our project, an IoT enabled device will control the entire monitoring system. Also developing an android app that helps patients by reminding them of medication over time and more.

2.LITERATURE SURVEY

2.1 EXISTING PROBLEM

In our daily lives many people are forgetting their duties, in that most of them are forgetting to take their medicines at right time and don't know what is the medicine they have to take. In the existing systems several solutions are provided i.e., setting reminders or using pill boxes, calendars, Personal Assistance such as keeping care takers.

But in our proposed System we have develop a medicine reminder system based on IoT, which reminds them by providing voice commands for taking the medicine.

2.2 REFERENCES

1. Dheeraj Muttin, Avinash Harale, " IOT BASED PERSONAL MEDICAL ASSISTANT SYSTEM" SKN Sinhgad College of Engineering, Pandharpur, India, In IJIRT |October 2021| Volume 8 Issue 5 | ISSN: 2349-6002.

2. Sultan Ahmad, Mahamudul Hasan, Gouse Pasha Mohammed, Mohammad Shahabuddin, Tasnia Tabassum, Mustafa Wasif Allvi "IOT BASED PILL REMINDER AND MONITORING SYSTEM" Department of Computer Science, College of Prince Sattam Bin Abdulaziz University, Al-Kharj, Saudi Arabia in International Journal of Computer Science and Network Security (IJCSNS) VOL.20 No.7, July 2020

3. Abhishek Madankar, Akhilesh Agrawal, Vedant Yede "IOT BASED ADVANCE PILL REMINDER SYSTEM FOR DISTINCT PATIENTS" in 2021 Fifth International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC)Dept. of E and TC Engineering, Y. C. College of Engineering, Nagpur, India.

4. P. Ranjana;Elizabeth Alexander "HEALTH ALERT AND MEDICINE REMAINDER USING INTERNET OF THINGS" in 2018 IEEE International Conference on Computational Intelligence and Computing Research (ICCIC)Department of Computer Science and Engineering, Hindustan Institute of Technology and Science, Chennai, India.

5. Pradnya Mhatre and Rasika Patil"IOT BASED PILL REMINDER SYSTEM" in International Journal of Advanced Research in Science, Communication and Technology (IJARSCT) Volume 2, Issue 1, July 2022 from Bharati Vidyapeeth' Institute of Management and Information Technology, University of Mumbai, India.

6.Balachandra Rao, Preetham M Nayak, Sachin S Nayak "MEDICATION REMINDER AND MONITORING SYSTEM USING IOT" in International Journal Of Current Engineering And Scientific Research (Ijcesr)Issn (Print): 2393-8374, (Online): 2394-0697, Volume-4, Issue-6, 2017 Department of MCA, NMAMIT, Nitte, Karkala, Udupi District.

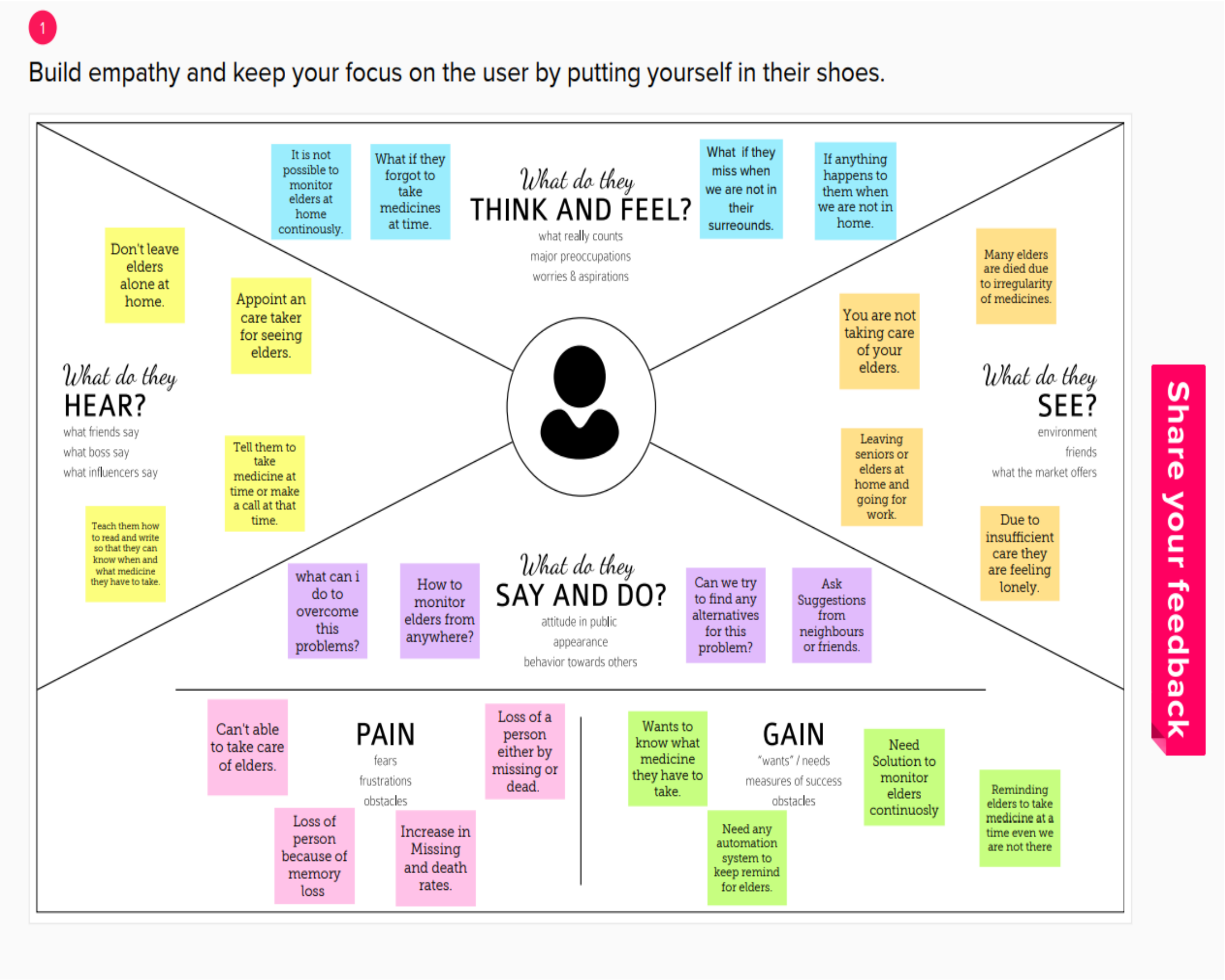
2.3 PROBLEM STATEMENT DEFINITION

I am user(elder people) **I'm trying to** remind the medicine **but** I'm forgetting **because** of my age **which makes me feel** very frustrated. **I am** User(elder people) **I am trying to** ask my family members to keep care taker **but** they are refusing **because** of the money problem **which makes me feel** very sad because of that I'm getting health problems more. **I am** User(care taker) **I'm trying to** give the medicine at right time **but** I can't be there always **because** of my other works **which makes me feel** very disappointed because of me they can't take medicine at right time.

3.IDEATION & PROPOSED SOLUTION

3.1 EMPATHY MAP CANVAS

An empathy map is **a collaborative tool teams can use to gain a deeper insight into their customers**. Much like a user persona, an empathy map can represent a group of users, such as a customer segment. The empathy map was originally created by Dave Gray and has gained much popularity within the agile community.



Build empathy and keep your focus on the user by putting yourself in their shoes.

What do they THINK AND FEEL?
what really counts
major preoccupations
worries & aspirations

It is not possible to monitor elders at home continously.

What if they forgot to take medicines at time.

What if they miss when we are not in their surreounds.

If anything happens to them when we are not in home.

What do they HEAR?
what friends say
what boss say
what influencers say

Don't leave elders alone at home.

Appoint an care taker for seeing elders.

Tell them to take medicine at time or make a call at that time.

Teach them how to read and write so that they can know when and what medicine they have to take.

What do they SEE?
environment
friends
what the market offers

Many elders are died due to irregularity of medicines.

You are not taking care of your elders.

Leaving seniors or elders at home and going for work.

Due to insufficient care they are feeling lonely.

What do they SAY AND DO?
attitude in public
appearance
behavior towards others

what can i do to overcome this problems?

How to monitor elders from anywhere?

Can we try to find any alternatives for this problem?

Ask Suggestions from neighbours or friends.

PAIN
fears
frustrations
obstacles

Can't able to take care of elders.

Loss of a person either by missing or dead.

Loss of person because of memory loss

Increase in Missing and death rates.

GAIN
"wants" / needs
measures of success
obstacles

Wants to know what medicine they have to take.

Need Solution to monitor elders continuosly

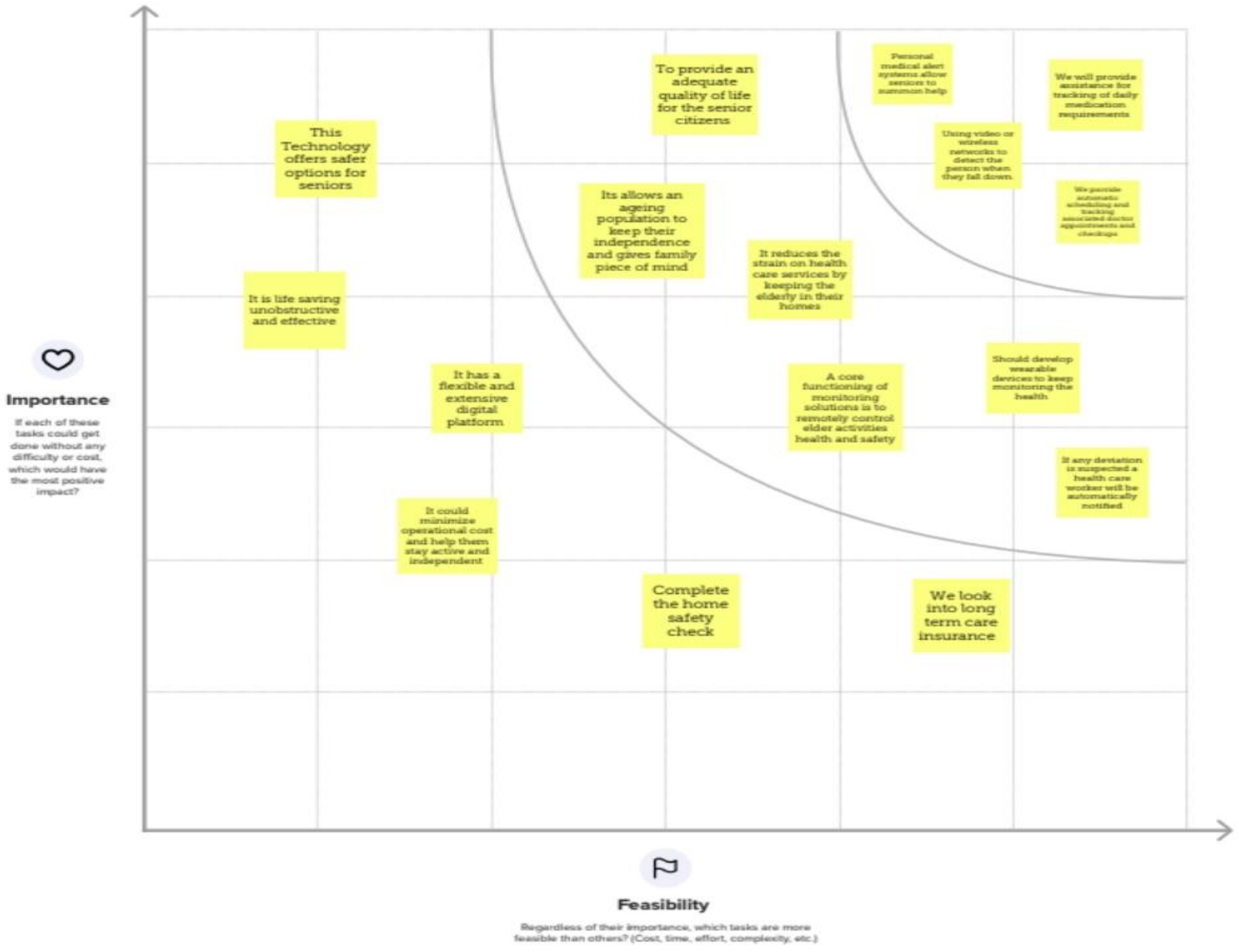
Need any automation system to keep remind for elders.

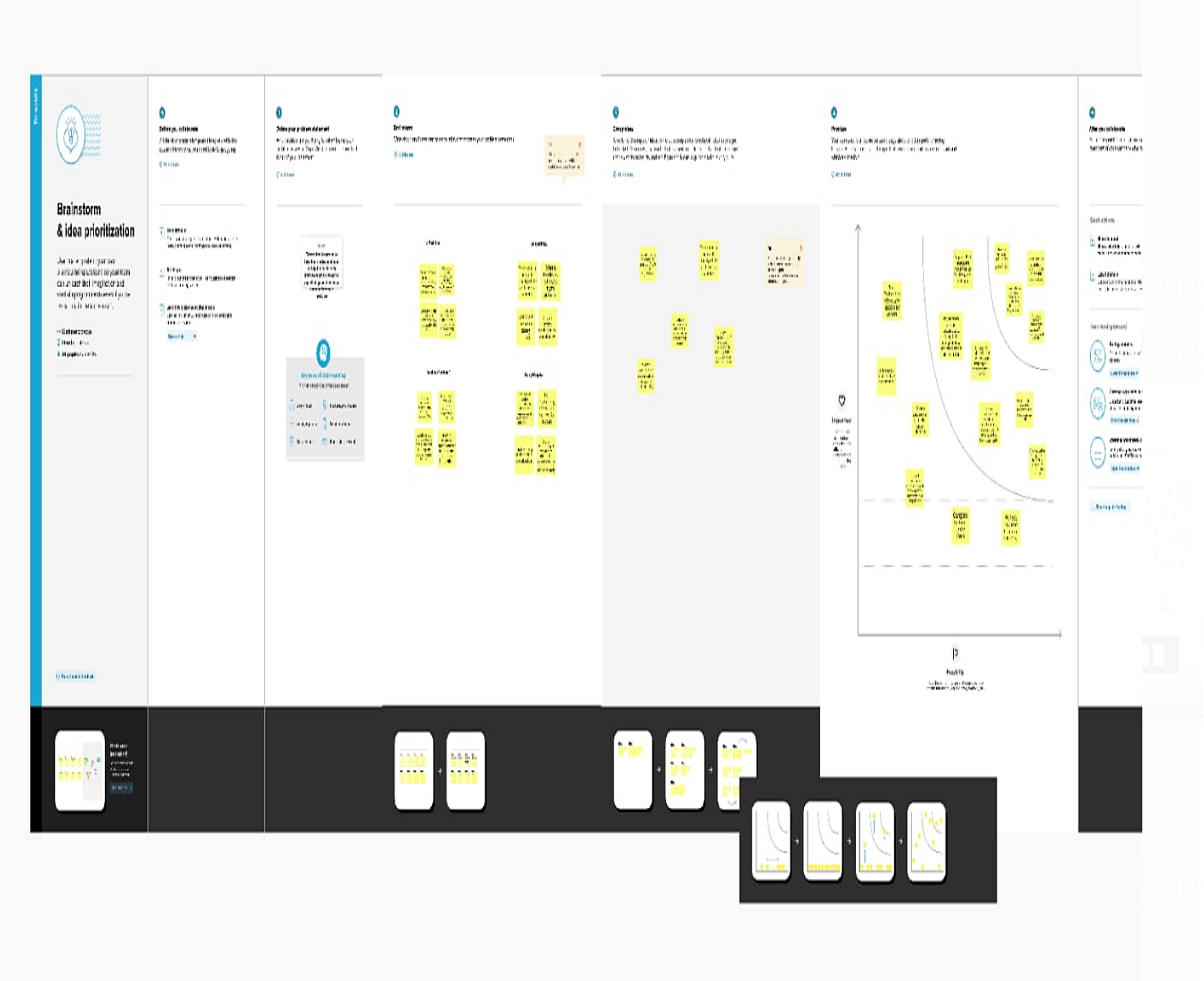
Reminding elders to take medicine at a time even we are not there

3.2 IDEATION AND BRAINSTROMING

Ideation is **the process of forming ideas from conception to implementation, most often in a business setting**. Ideation is expressed via graphical, written, or verbal methods, and arises from past or present knowledge, influences, opinions, experiences, and personal convictions.

Ideation is often closely related to the practice of brainstorming, a specific technique that is utilized to generate new ideas. A principal difference between ideation and brainstorming is that **ideation is commonly more thought of as being an individual pursuit, while brainstorming is almost always a group activity**.


This Technology offers safer options for seniors
It is life saving unobstructive and effective
It has a flexible and extensive digital platform
It could minimize operational cost and help them stay active and independent
To provide an adequate quality of life for the senior citizens
Its allows an ageing population to keep their independence and gives family piece of mind
It reduces the strain on health care services by keeping the elderly in their homes
A core functioning of monitoring solutions is to remotely control elder activities health and safety
Complete the home safety check
Personal medical alert systems allow seniors to summon help
Using video or wireless networks to detect the person when they fall down.
We will provide assistance for tracking of daily medication requirements
We provide automatic scheduling and tracking associated doctor appointments and checkups
Should develop wearable devices to keep monitoring the health
If any deviation is suspected a health care worker will be automatically notified
We look into long term care insurance
Importance
If each of these tasks could get done without any difficulty or cost, which would have the most positive impact?
Feasibility
Regardless of their importance, which tasks are more feasible than others? (Cost, time, effort, complexity, etc.)


Brainstorm
& idea prioritization
Before you collaborate
Define your problem statement
Brainstorm
Group ideas
Prioritize
After you collaborate
Importance
Feasibility

3.3 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 He / She should take at that particular time. ● And it is difficult for doctors/caretakers to monitor the patients around the clock.
2.	Idea / Solution description	● To avoid this problem, we will develop a medicine remainder system. ● 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. ● This system provides a real time monitoring system that allows relatives to monitor patient activity from a distance.
3.	Novelty / Uniqueness	● This system provides alerts when it's time to take medicine and the details saved in the cloud database can be update automatically using IoT Technology.
4.	Social Impact / Customer Satisfaction	● This proposed system provides many facilities which help the patients to take medicine at right time and a real time monitoring system that allows relatives to monitor patient activity from a distance.
5.	Business Model (Revenue Model)	● This protype can be developed as communication between devices and the cloud, as well as between the devices themselves such as IoT device and web application app and also IBM IoT platform.
6.	Scalability of the Solution	● This can be developed as a medicine remainder app using web application and IoT devices such as Aurdino devices with cloud services and finally gives voice commands using IBM Text to speech Service.

3.4 PROBLEM SOLUTION FIT

i) CUSTOMER SEGMENTS(S)

- ➤ Elders or seniors i.e. Parents of working children(nearly 55-70 + years)

ii) JOBS -TO-BE-DONE / PRO

- ➤ Elders are not caring by their children.
- ➤ Elders often forgot to take medicine at proper time.
- ➤ Sometimes elders also forgot what medicine they have to take.

iii) TRIGGERS

- ➤ By seeing others keeping medicine reminder systems in home they too approach for medicine reminder systems.
- ➤ By hearing about medical alert systems in news and trying to keep in their homes.

iv) EMOTIONS: BEFORE/AFTER

- ➤ Health got affected more due to insufficient medication.
- ➤ Health recovers fastly with proper medication.

v) AVAILABLE SOLUTIONS

- ➤ Medicine reminder system to give notification when it times to take the medicine.
- ➤ Automatic Reminder system. Mobile Applications to monitor the elders when we are far away from home.

vi) CUSTOMER CONSTRAINTS

- ➤ Limited financial constraints.
- ➤ Too much of work pressure.
- ➤ Loneliness.

vii) BEHAVIOUR

- ➤ Searching for an alternative solution for an existing solution.
- ➤ Ask Suggestions from skilled persons for monitoring them regularly.
- ➤ Try to provide sufficient care for elders.

viii) CHANNELS OF BEHAVIOUR

- ➤ **ONLINE:**

  Using Online apps or any social media apps to promote medicine reminder systems and tells the importance of medicine reminder system.
- ➤ **OFFLINE:**

  Create awareness about medicine remainder system around the people in your surroundings.

  Do the publicity regarding medicine reminder systems.

ix) PROBLEM ROOT/CAUSE

- Due to the age factor of the elders, theyoften forgot to take medicine.
- Because of their loneliness, the elders feel uncomfortable and uneasy to take medicines.
- Because of their working children the elders are not caring properly.

x) YOUR SOLUTION

- For avoiding this problem, we should develop a medicine reminder system.
- It requires an app which can be built using MIT App Inventor used for setting the desired time and medicine.
- Here we are using loT Device Cloudant DB, web applications and text to speech. The details which we used for setting time and medicine that data should be stored in Cloudant DB.
- If the medicine time arrives the web application will send the medicine to the lot device through the IBM IOT platform.

4.REQUIREMENT ANALYSIS

4.1 FUNCTIONAL REQUIREMENTS

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	Authenticate user	User verification using username and password
FR-2	Time and medicine setting	Setting the details of medicine and time
FR-3	Storing the data in Cloudant DB	Stores the details of the medicine such as doses of medicine and time details in Cloudant DB
FR-4	Sending information or data	Web application send the medicine name to IoT device through IoT Platform
FR-5	Giving Notification	The IoT device receives the medicine name and notify the user using voice commands

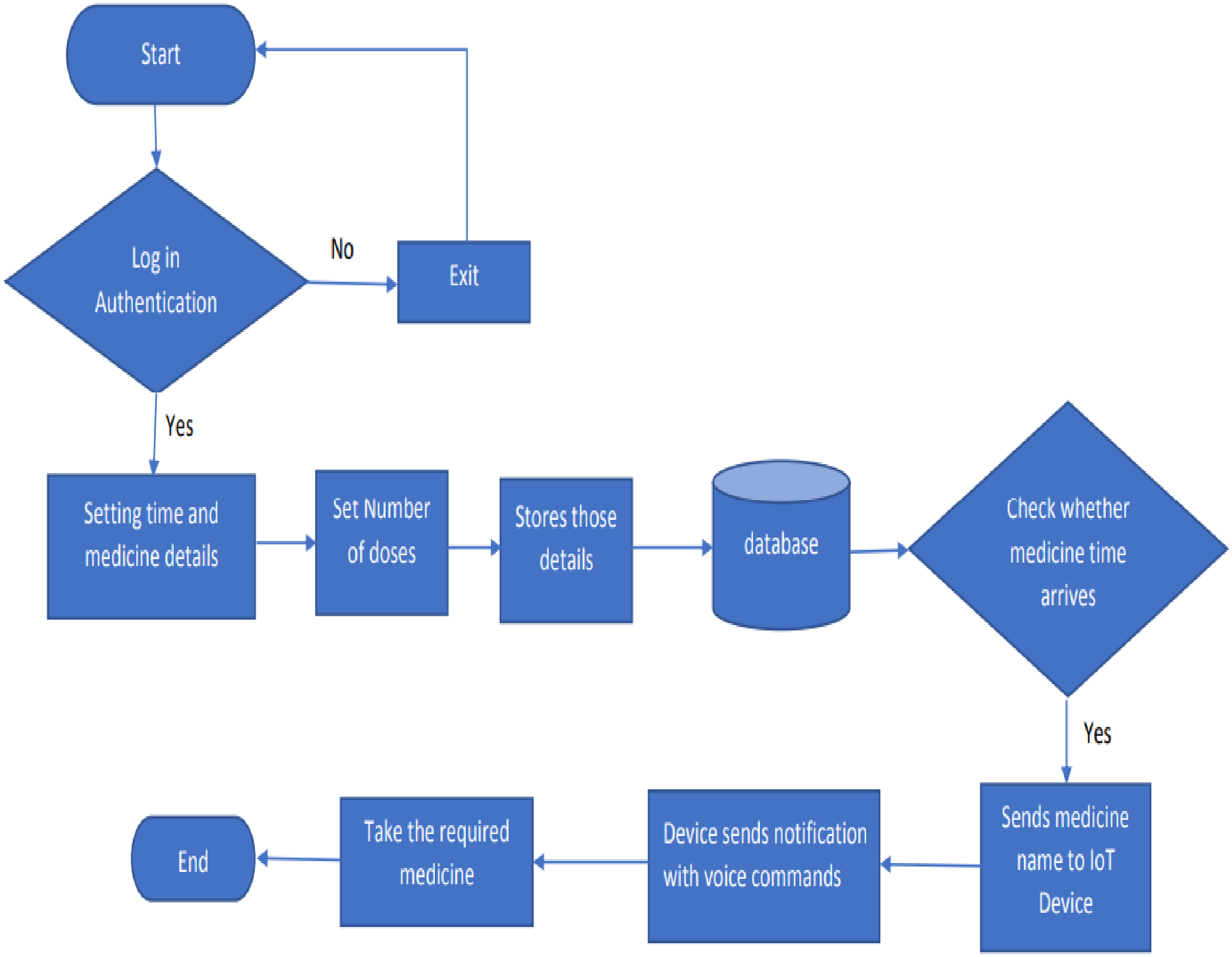
4.2 NON FUNCTIONAL REQUIREMENTS

FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	The System should provide a systematic, simple and user-friendly interfaces.
NFR-2	**Security**	The system must ensure all information of the registered end users are secured and not accessible by other party.
NFR-3	**Reliability**	The system should have lower risk of errors and process failures that can cause patients harm.
NFR-4	**Performance**	To perform good speed, reliability and capacity of the system is needed.
NFR-5	**Availability**	The system should be available if there are any changes to functionalities of the system in future.
NFR-6	**Scalability**	The system is scalable to increase or decrease in performance according to requirements

5.PROJECT DESIGN

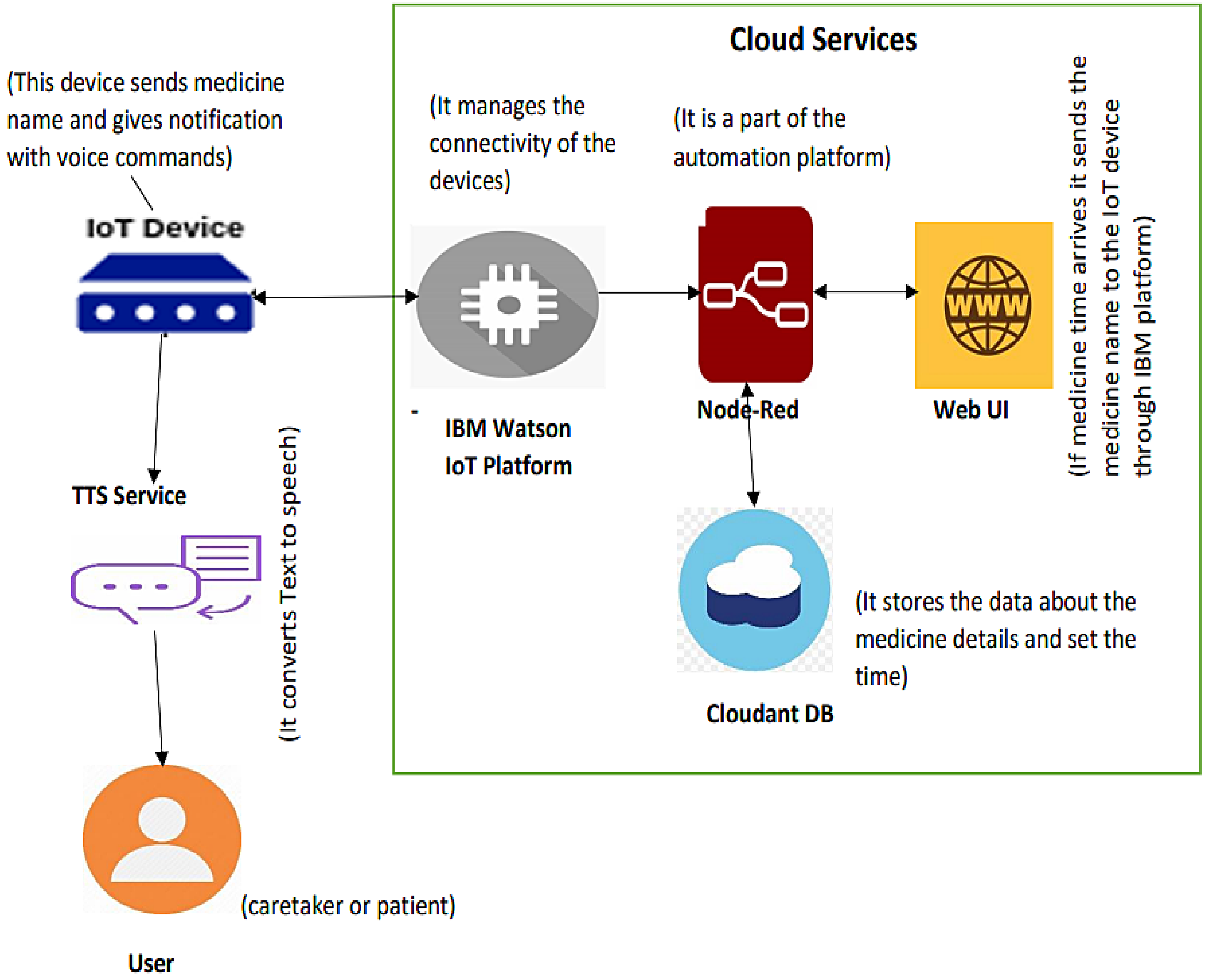
5.1 DATAFLOW DIAGRAMS

A data-flow diagram is a way of representing a flow of data through a process or a system. The DFD also provides information about the outputs and inputs of each entity and the process itself. A data-flow diagram has no control flow — there are no decision rules and no loops.



5.2 SOLUTION AND TECHNICAL ARCHITECTURE

Based on the complexity of the deployment, a solution architecture diagram may actually be **a set of diagrams documenting various levels of the architecture**. The diagram relates the information that you gather on the environment to both physical and logical choices for your architecture in an easily understood manner.


Cloud Services
(This device sends medicine name and gives notification with voice commands)
IoT Device
(It manages the connectivity of the devices)
(It is a part of the automation platform)
(If medicine time arrives it sends the medicine name to the IoT device through IBM platform)
IBM Watson IoT Platform
Node-Red
Web UI
WWW
TTS Service
(It converts Text to speech)
Cloudant DB
(It stores the data about the medicine details and set the time)
(caretaker or patient)
User

5.3 USER STORIES

- **CUSTOMER(MOBILE USER):**

As a user, I can register for the application by entering my username, password, and confirming my password And I can also access my dashboard. As a user, I can log into the application by entering username & password

- **CUSTOMER(MOBILE USER):**

As a user, I can collect the data and set the time and medicine details. As a user, I can set the number of doses of the medicine.As a user, I can receive the medicine name as voice commands from the device.

- **ADMINISTRATOR:**

As a administrator, I can update the system with new requirements.

6. PROJECT PLANNING & SCHEDULING

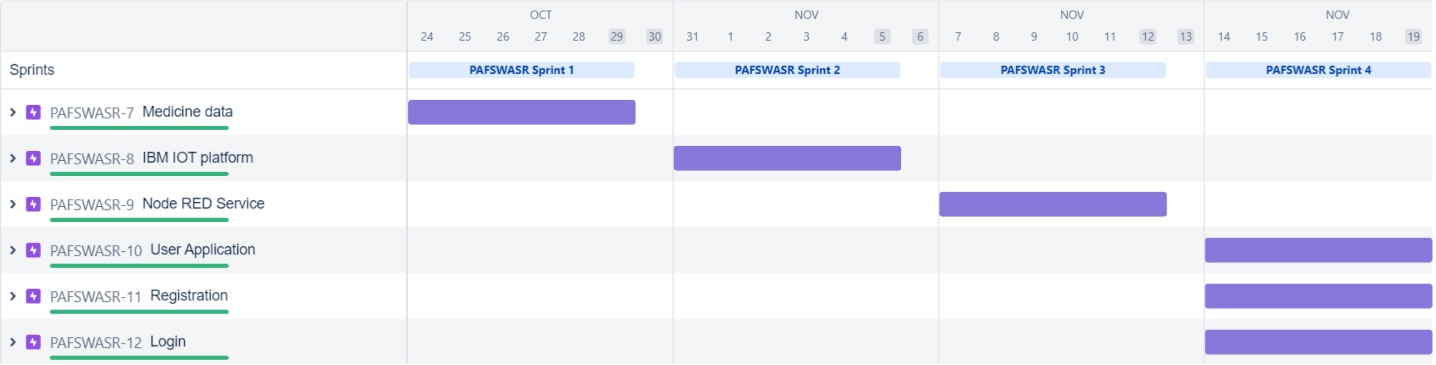
6.1 SPRINT PLANNING AND ESTIMATION

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Medicine data (Python script)	USN-1	Here we are using python script to generate the random data.	8	High	U Ruchitha Daisy Deepika N
Sprint-2	IBM IOT platform	USN-2	To send the data to IOT Device, IBM IOT platform is created.	5	High	U Ruchitha Daisy Deepika N Venkata Sunitha D
Sprint-3	Node RED Service	USN-3	To access the IBM IOT platform from external application or from external UI Node red service is established.	5	High	U Ruchitha Sangeetha E Daisy Deepika N
Sprint-4	User Application	USN-4	To Set time and medicine details the User is provided with an user application created by MIT App Inventor	5	High	Venkata Sunitha D Sangeetha E Daisy Deepika N
Sprint-4	Registration	USN-5	As a user, I can register for the application by entering my username, password, and confirming my password.	3	Low	U Ruchitha Venkata Sunitha D Sangeetha E
Sprint-4	Login	USN-6	As a user, I can login into the application by entering email & password.	2	Low	Venkata Sunitha D Sangeetha E

6.2 SPRINT DELIVERY AND 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	8	6 Days	24 Oct 2022	29 Oct 2022	8	29 Oct 2022
Sprint-2	5	6 Days	31 Oct 2022	05 Nov 2022	5	06 Nov 2022
Sprint-3	5	6 Days	07 Nov 2022	12 Nov 2022	5	12 Nov 2022
Sprint-4	10	6 Days	14 Nov 2022	19 Nov 2022	10	19 Nov 2022

6.3 REPORTS FROM JIRA

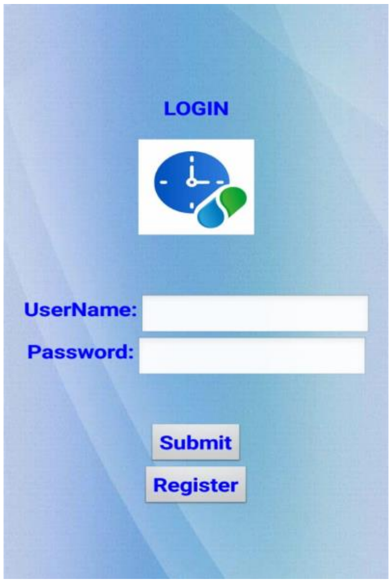


7.CODING AND SOLUTIONING

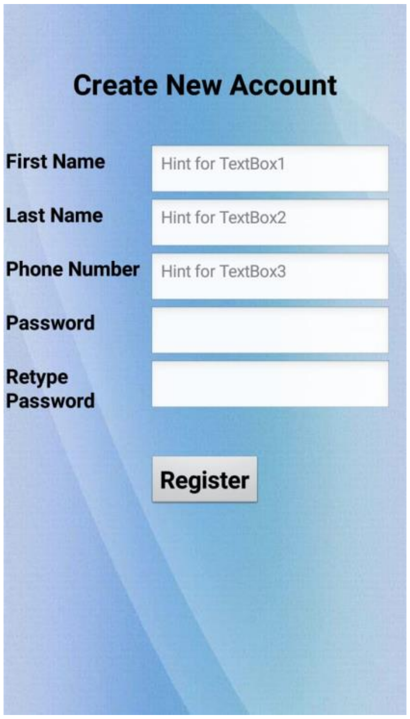
7.1 FEATURE 1

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

Login Page

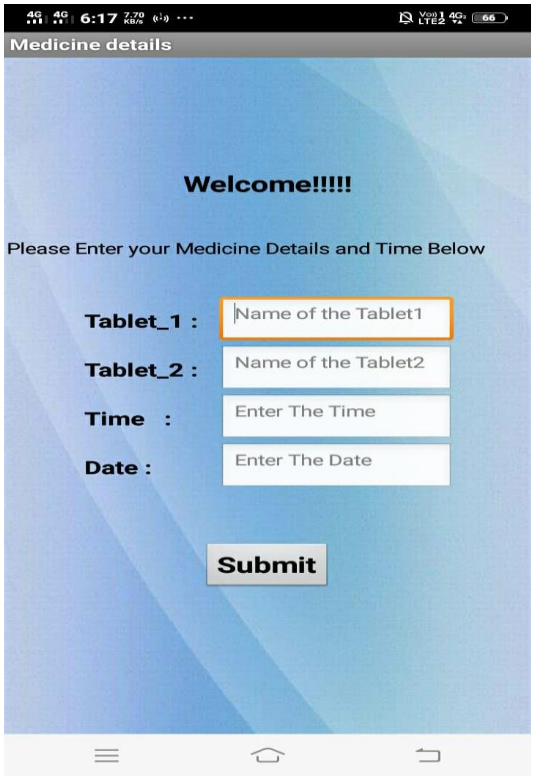


Register Page



7.2 FEATURE 2

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



Medicine details

Welcome!!!!!

Please Enter your Medicine Details and Time Below

Tablet_1 : Name of the Tablet1

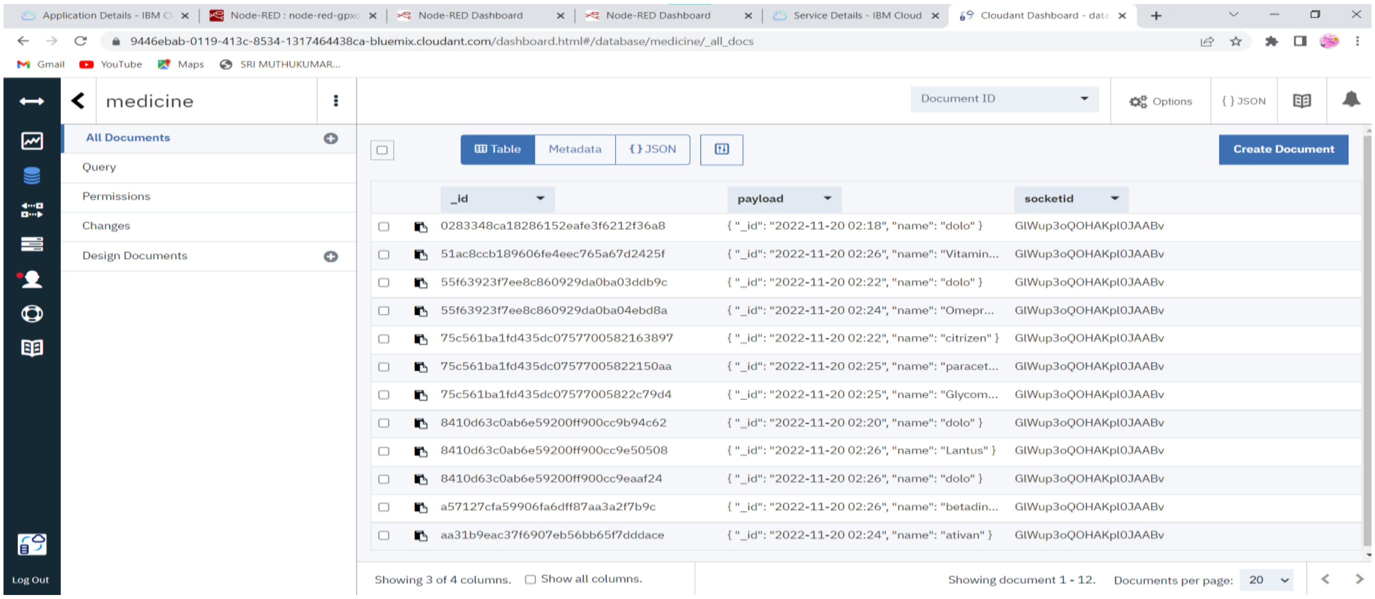
Tablet_2 : Name of the Tablet2

Time : Enter The Time

Date : Enter The Date

7.3 FEATURE 3

The project includes a cloud database system



9446ebab-0119-413c-8534-1317464438ca-bluemix.cloudant.com/dashboard.html#/database/medicine/_all_docs

medicine

	_id	payload	socketid
□	0283348ca18286152eafe3f6212f36a8	{ "_id": "2022-11-20 02:18", "name": "dolo" }	GIWup3oQOHAKpl0JAABv
□	51ac8ccb189606fe4eec765a67d2425f	{ "_id": "2022-11-20 02:26", "name": "Vitamin...	GIWup3oQOHAKpl0JAABv
□	55f63923f7ee8c860929da0ba03ddb9c	{ "_id": "2022-11-20 02:22", "name": "dolo" }	GIWup3oQOHAKpl0JAABv
□	55f63923f7ee8c860929da0ba04ebd8a	{ "_id": "2022-11-20 02:24", "name": "Omepr...	GIWup3oQOHAKpl0JAABv
□	75c561ba1fd435dc0757700582163897	{ "_id": "2022-11-20 02:22", "name": "citrizen" }	GIWup3oQOHAKpl0JAABv
□	75c561ba1fd435dc07577005822150aa	{ "_id": "2022-11-20 02:25", "name": "paracet...	GIWup3oQOHAKpl0JAABv
□	75c561ba1fd435dc07577005822c79d4	{ "_id": "2022-11-20 02:25", "name": "Glycom...	GIWup3oQOHAKpl0JAABv
□	8410d63c0ab6e59200ff900cc9b94c62	{ "_id": "2022-11-20 02:20", "name": "dolo" }	GIWup3oQOHAKpl0JAABv
□	8410d63c0ab6e59200ff900cc9e50508	{ "_id": "2022-11-20 02:26", "name": "Lantus" }	GIWup3oQOHAKpl0JAABv
□	8410d63c0ab6e59200ff900cc9eaaf24	{ "_id": "2022-11-20 02:26", "name": "dolo" }	GIWup3oQOHAKpl0JAABv
□	a57127cfa59906fa6dff87aa3a2f7b9c	{ "_id": "2022-11-20 02:26", "name": "betadin...	GIWup3oQOHAKpl0JAABv
□	aa31b9eac37f6907eb56bb65f7dddace	{ "_id": "2022-11-20 02:24", "name": "ativan" }	GIWup3oQOHAKpl0JAABv

Showing 3 of 4 columns. □ Show all columns.

Showing document 1 - 12. Documents per page: 20

8.TESTING

8.1 TESTCASES

A test case is a set of actions performed on a system to determine if it satisfies software requirements and functions correctly. A test case is a document, which has a set of test data, preconditions, expected results and postconditions, developed for a particular test scenario.

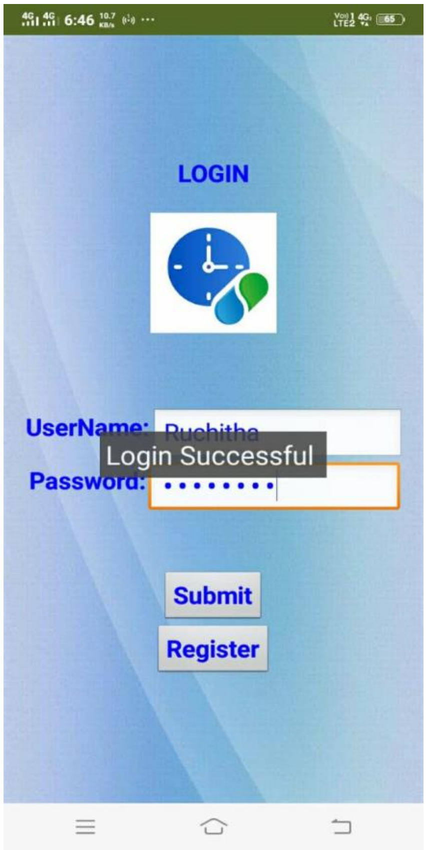
Test case	Precondition	Test steps	Test data	Expected Result
Verify login with valid credentials	User should have a network condition	1. Launch URL 2. Enter valid username. 3. Enter valid password. 4. Click on the “Login” button.	Username: Ruchitha Password: ruchitha	User sholud be able to login successfully
Verify login with invalid credentials	User should have a network condition	1. Launch URL 2. Enter valid username. 3. Enter valid password. 4. Click on the “Login” button.	Username: Ruchitha Password: 12345678	User should not be able to login
Register with valid credentials	User should have a network condition	1. Enter First Name. 2. Enter Last Name. 3. Enter Phone Number 4. Enter Password 5.Retype the Password	First Name: Ruchitha Last Name: Unnam Phone Number: 8878952647 Password: ruchitha Retype Password: ruchitha	User can able to register successfully
Update the medicine name with the time.	User should have a network condition	1. Enter valid medicine name. 2. Enter the time when the medicine has to be consumed. 3. Click on the “Submit” button.	Medicine Name: dolo Medicine Time: 21:00	Users should be able to update it successfully

8.2 USER ACCEPTANCE TESTING

User acceptance testing (UAT), also called application testing or end-user testing, is a phase of software development in which the software is tested in the real world by its intended audience. UAT is often the last phase of the software testing process and is performed before the tested software is released to its intended market. The goal of UAT is to ensure software can handle real-world tasks and perform up to development specifications.

In UAT, users are given the opportunity to interact with the software before its official release to see if any features have been overlooked or if it contains any bugs. UAT can be done in-house with volunteers, by paid test subjects using the software or by making the test version available for download as a free trial.

Login page Testing:

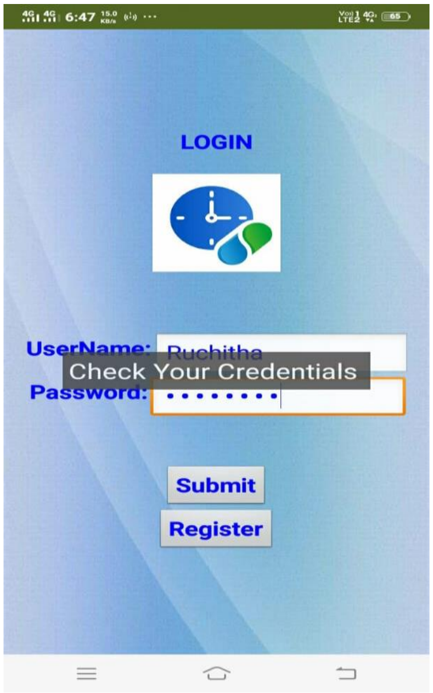

LOGIN

UserName: Ruchitha

Password: ••••••••

Login Successful

Incorrect Login attempt:

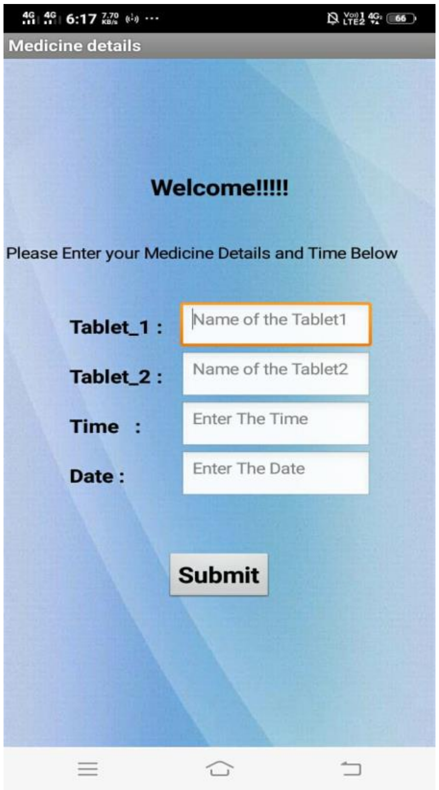


UserName: Ruchitha

Password: ••••••••

Check Your Credentials

Medicine Page Testing :



Welcome!!!!!

Please Enter your Medicine Details and Time Below

Tablet_1 : Name of the Tablet1

Tablet_2 : Name of the Tablet2

Time : Enter The Time

Date : Enter The Date

9.RESULTS

9.1 PERFORMANCE METRICS

S.NO	Parameter	Performance
1.	Response Time	0.25s(Average of 10 trials)
2.	Workload	500 users (Based on cloud space)
3.	Revenue	Individual users and pharmaceutical industries.
4.	Efficiency	SImple and straight forward workflow, which makes the process efficient
5.	Down Time	Almost no down time due to IBM Cloud enabled solution.

10.ADVANTAGES & DISADVANTAGES

ADVANTAGES:

- In this system voice command reminder enables user to take medicines in regular times.
- Needs no prior techinical knowledge .
- It is easy to use.
- Simple alerts and notification with voice commands .
- Medication calender reminders and other options .
- Manual entry of medication names , dosages, time records,and notes .
- Medication reminders serve as a good way to stay on track and upload an appropriate schedule.
- Cost Efficient.
- Can store multiple data and many notifications can be generated.
- Avoid personal assistants or caretakers needed for medically sick people.
- Since it includes voice assistance, even blind people can use our device.

DISADVANTAGES:

- Requires a stable internet connection.
- Have to be updated daily.

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 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. This leads to conclusion that all people whether elder or younger can able to take medicine in correct time by using our app through voice commands.

12. FUTURE SCOPE

Future enhancement can be made by preserving the data prior in need. Several other methods can be implemented for reminding the medicines.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:

```
# Importing the libraries
import ibmiotf.application
import ibmiotf.device
from time import sleep
import sys
import schedule
import time
import datetime

#IBM Watson Device Credentials.
organization = "1dh5ok"
deviceType = "ruchitha"
deviceId = "ruchitha18"
authMethod = "token"
authToken = "12345678"


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()
```

```
#Connecting to IBM watson.
deviceCli.connect()
print("")

# Getting details about tablet placed in pill box:

while True:


    tab_typ_count = input("Enter how many types of tablet placed in pill Box,
maximum 2 types: ")


    if (tab_typ_count == "1"):
     tab_1 = input("Enter the name of Tablet 1: ")
     success = deviceCli.publishEvent("tablet 1", "json", { "Tablet_1" : " %s " % tab_1
}, qos=0)
     if success:
         print("published tablet name :%s"% tab_1)
     success = deviceCli.publishEvent("tablet 2", "json", { "Tablet_2" : " No Tablet" },
qos=0)
     if success:
         print("published tablet name : NO tablets")
     print("")

    elif (tab_typ_count == "2"):
     tab_1 = input("Enter the name of Tablet 1: ")
     success = deviceCli.publishEvent("tablet 1", "json", { "Tablet_1" : " %s " % tab_1
}, qos=0)
     if success:
         print("published tablet name :%s"% tab_1)
     print("")


     tab_2 = input("Enter the name of Tablet 2: ")
```

```
        success = deviceCli.publishEvent("tablet 2", "json", { "Tablet_2" : " %s " % tab_2
}, qos=0)
        if success:
              print("published tablet name :%s"% tab_2)
        print("")

       if (tab_typ_count != "1" and tab_typ_count != "2"):
        print("Enter the count betwen 1 to 2")

       else:
         break

# Reminder functions:

def tab_1_mng_reminder():
    success = deviceCli.publishEvent("Medicine reminder", "json", {
"Medicine_reminderM1" : "Now it is the time to consume your Tablet %s " % tab_1 },
qos=0)
    if success:
          print("Publish ok")
          print("Published reminder : Now the time is %s:%s consume your Tablet %s "
% (mng_rem_tab1_time_HH, mng_rem_tab1_time_MM, tab_1))
          print("")


    tab_1_ack = input("Have you consumed your tablet %s:  " %tab_1)
    if (tab_1_ack == "yes"):
  success = deviceCli.publishEvent("Acknowledgement", "json", {
"Aknowledgement" : "The patient has consumed the Tablet %s " % tab_1 }, qos=0)
          if success:
                print("publish ok")
                print("published aknowledgement: The patient has consumed the
tablet")

    elif (tab_1_ack == "no"):
        success = deviceCli.publishEvent("Alert", "json", { "Alert" : "The patient not
consumed the Tablet %s " % tab_1 }, qos=0)
        if success:
            print("publish ok")
            print("published aknowledgement: The patient not consumed the
tablet")


def tab_1_evg_reminder():
    success = deviceCli.publishEvent("Medicine reminder", "json", {
"Medicine_reminderE1" : "Now it is the time to consume your Tablet %s " % tab_1 },
qos=0)
    if success:
        print("Publish ok")
        print("Published reminder : Now the time is %s:%s consume your Tablet %s "
% (evg_rem_tab1_time_HH, evg_rem_tab1_time_MM, tab_1))
        print("")

    tab_1_ack = input("Have you consumed your tablet %s:  " %tab_1)
    if (tab_1_ack == "yes"):
        success = deviceCli.publishEvent("Acknowledgement", "json", {
"Aknowledgement" : "The patient has consumed the Tablet %s " % tab_1 }, qos=0)
        if success:
            print("publish ok")
            print("published aknowledgement: The patient has consumed the
tablet")
    elif (tab_1_ack == "no"):
        success = deviceCli.publishEvent("Alert", "json", { "Alert" : "The patient not
consumed the Tablet %s " % tab_1 }, qos=0)
        if success:
            print("publish ok")
            print("published aknowledgement: The patient not consumed the
tablet")

def tab_2_mng_reminder():
    success =deviceCli.publishEvent("Medicine reminder", "json", {
"Medicine_reminderM2" : "Now it is the time to consume your Tablet %s " % tab_2 },
qos=0)
    if success:
        print("Publish ok")
        print("Published reminder : Now the time is %s:%s consume your Tablet %s "
% (mng_rem_tab2_time_HH, mng_rem_tab2_time_MM,tab_2))
        print("")

    tab_2_ack = input("Have you consumed your tablet %s:  " %tab_2)
    if (tab_2_ack == "yes"):
        success = deviceCli.publishEvent("Acknowledgement", "json", {
"Aknowledgement" : "The patient has consumed the Tablet %s " % tab_2 }, qos=0)
        if success:
            print("publish ok")
            print("published aknowledgement: The patient has consumed the
tablet")

    elif (tab_2_ack == "no"):
        success = deviceCli.publishEvent("Alert", "json", { "Alert" : "The patient not
consumed the Tablet %s " % tab_1 }, qos=0)
        if success:
            print("publish ok")
            print("published aknowledgement: The patient not consumed the
tablet")


def tab_2_evg_reminder():
    success = deviceCli.publishEvent("Medicine reminder", "json", {
"Medicine_reminderE2" : "Now it is the time to consume your Tablet %s " % tab_2 },
qos=0)
    if success:
        print("Publish ok")
        print("Published reminder : Now the time is %s:%s consume your Tablet %s "
% (evg_rem_tab2_time_HH, evg_rem_tab2_time_MM,tab_2))
```

```
        print("")

    tab_2_ack = input("Have you consumed your tablet %s:  " %tab_2)
    if (tab_2_ack == "yes"):
        success = deviceCli.publishEvent("Acknowledgement", "json", {
"Aknowledgement" : "The patient has consumed the Tablet %s " % tab_2 }, qos=0)
        if success:
            print("publish ok")
            print("published aknowledgement: The patient has consumed the
tablet")
    elif (tab_2_ack == "no"):
        success = deviceCli.publishEvent("Alert", "json", { "Alert" : "The patient not
consumed the Tablet %s " % tab_2 }, qos=0)
        if success:
            print("publish ok")
            print("published aknowledgement: The patient not consumed the
tablet")


# If only one type of tablet is placed in the pill box, user will be provided with this
options:

if (tab_typ_count == "1"):

   # For morning reminder
   while True:

      mng_option = input("Do you want reminder for Morning(YES/NO): ")
      print("")

      if (mng_option == "YES" or mng_option == "yes"):

          while True:
            mng_rem_tab_1 = input("Enter the name of tablet  to be reminded at
Morning: ")
```

```
            mng_rem_tab1_time_HH = input("Enter the Hour(HH) to be reminded : ")
            mng_rem_tab1_time_MM = input("Enter the Minute(MM) to be reminded
: ")
            print("")

            if (mng_rem_tab_1 == tab_1):
                 schedule.every().day.at("%s:%s"
%(mng_rem_tab1_time_HH,mng_rem_tab1_time_MM)).do(tab_1_mng_reminder)

            if (mng_rem_tab_1 != tab_1):
                 print("The tablet name you entered is not in pill box, again enter the
correct name")
                 print("")

            else:
                 break


     if (mng_option != "YES" and mng_option != "yes" and mng_option != "NO" and
mng_option != "no"):
        print("Only type YES or NO")
        print("")

     else:
        break


  #For evening reminder
   while True:

     evg_option = input("Do you want reminder for Evening(YES/NO): ")
     print("")


     if (evg_option == "YES" or evg_option == "yes" ):
```

```
                while True:
                    evg_rem_tab_1 = input("Enter the name of tablet to be reminded at Evening: ")
                    evg_rem_tab1_time_HH = input("Enter the Hour(HH) to be reminded : ")
                    evg_rem_tab1_time_MM = input("Enter the Minute(MM) to be reminded : ")
                    print("")

                    if (evg_rem_tab_1 == tab_1):
                        schedule.every().day.at("%s:%s" %(evg_rem_tab1_time_HH,evg_rem_tab1_time_MM)).do(tab_1_evg_reminder)

                    if (evg_rem_tab_1 != tab_1):
                        print("The tablet name you entered is not in pill box, again enter the correct name")
                        print("")

                    else:
                        break

        if (evg_option != "YES" and evg_option != "yes" and evg_option != "NO" and evg_option != "no"):
            print("Only type YES or NO")
            print("")

        else:
            break


# If the two types of tablet place in the pill box user will be provided with this option:

if (tab_typ_count == "2"):

    # For morning reminder
```

```
  while True:
    mng_option = input("Do you want reminder for Morning(YES/NO): ")

    if (mng_option == "YES" or mng_option == "yes"):

        while True:
            mng_reminder = input("Enter the number of Tablets to be reminded for
Morning: ")
            print("")

            if (mng_reminder == "1"):

                while True:

                  mng_rem_tab_1 = input("Enter the name of tablet to be
reminded at Morning: ")
                  mng_rem_tab1_time_HH = input("Enter the Hour(HH) to be
reminded : ")
                  mng_rem_tab1_time_MM = input("Enter the Minute(MM) to be
reminded : ")
                  print("")

                  if (mng_rem_tab_1 == tab_1):
                      schedule.every().day.at("%s:%s"
%(mng_rem_tab1_time_HH,mng_rem_tab1_time_MM)).do(tab_1_mng_reminder)

                  if (mng_rem_tab_1 != tab_1):
                      print("The tablet name you entered is not in pill box, again
enter the correct name")
                      print("")

                  else:
                      break
            elif (mng_reminder == "2"):


                while True:

                  mng_rem_tab_1 = input("Enter the name of first tablet  to be reminded at Morning: ")
                  mng_rem_tab1_time_HH = input("Enter the Hour(HH) to be reminded : ")
                  mng_rem_tab1_time_MM = input("Enter the Minute(MM) to be reminded : ")
                  print("")

                  if (mng_rem_tab_1 == tab_1):
                      schedule.every().day.at("%s:%s" %(mng_rem_tab1_time_HH,mng_rem_tab1_time_MM)).do(tab_1_mng_reminder)
                  elif(mng_rem_tab_1 == tab_2):
                      schedule.every().day.at("%s:%s" %(mng_rem_tab1_time_HH,mng_rem_tab1_time_MM)).do(tab_2_mng_reminder)

                  if (mng_rem_tab_1 != tab_1 and mng_rem_tab_1 != tab_2 ):
                      print("The tablet name you entered is not in pill box, again enter the correct name")
                      print("")

                  else:
                      break



                while True:

                  mng_rem_tab_2 = input("Enter the name of second tablet  to be reminded at Morning: ")
                  mng_rem_tab2_time_HH = input("Enter the Hour(HH) to be reminded : ")
```

```
                  mng_rem_tab2_time_MM = input("Enter the Minute(MM) to be
reminded : ")
                  print("")

                  if (mng_rem_tab_2 == tab_1):
                      schedule.every().day.at("%s:%s"
%(mng_rem_tab2_time_HH,mng_rem_tab2_time_MM)).do(tab_1_mng_reminder)
                  elif (mng_rem_tab_2 == tab_2):
                      schedule.every().day.at("%s:%s"
%(mng_rem_tab2_time_HH,mng_rem_tab2_time_MM)).do(tab_2_mng_reminder)

                  if (mng_rem_tab_2 != tab_1 and mng_rem_tab_2 != tab_2 ):
                      print("The tablet name you entered is not in pill box, again
enter the correct name")
                      print("")

                  else:
                      break


            if (mng_reminder != "1" and mng_reminder != "2"):
                print("Enter the value  as 1 or 2")
                print("")

            else:
                break


    if (mng_option != "YES" and mng_option != "yes" and mng_option != "no" and
mng_option != "NO"):
        print("Only type YES or NO")
        print("")

    else:
        break
    #For evening reminder
    while True:
        evg_option = input("Do you want reminder for Evening(YES/NO): ")

        if (evg_option == "YES" or evg_option == "yes"):

                while True:
                        evg_reminder = input("Enter the number of Tablets to be reminded for Evening: ")
                        print("")

                        if (evg_reminder == "1"):

                                while True:

                                     evg_rem_tab_1 = input("Enter the name of tablet to be reminded at Evening: ")
                                     evg_rem_tab1_time_HH = input("Enter the Hour(HH) to be reminded : ")
                                     evg_rem_tab1_time_MM = input("Enter the Minute(MM) to be reminded : ")
                                     print("")

                                     if (evg_rem_tab_1 == tab_1):
                                             schedule.every().day.at("%s:%s" %(evg_rem_tab1_time_HH,evg_rem_tab1_time_MM)).do(tab_1_evg_reminder)

                                     if (evg_rem_tab_1 != tab_1):
                                            print("The tablet name you entered is not in pill box, again enter the correct name")
                                            print("")

                                     else:
```

```
                    break

        elif (evg_reminder == "2"):


            while True:

                evg_rem_tab_1 = input("Enter the name of first tablet  to be
reminded at Evening: ")
                evg_rem_tab1_time_HH = input("Enter the Hour(HH) to be
reminded : ")
                evg_rem_tab1_time_MM = input("Enter the Minute(MM) to be
reminded : ")
                print("")

                if (evg_rem_tab_1 == tab_1):
                    schedule.every().day.at("%s:%s"
%(evg_rem_tab1_time_HH,evg_rem_tab1_time_MM)).do(tab_1_evg_reminder)
                elif (evg_rem_tab_1 == tab_2):
                    schedule.every().day.at("%s:%s"
%(evg_rem_tab1_time_HH,evg_rem_tab1_time_MM)).do(tab_2_evg_reminder)

                if (evg_rem_tab_1 != tab_1 and evg_rem_tab_1 != tab_2):
                    print("The tablet name you entered is not in pill box, again
enter the correct name")
                    print("")

                else:
                    break



            while True:

                evg_rem_tab_2 = input("Enter the name of second tablet  to be
reminded at Evening: ")
```

```
                   evg_rem_tab2_time_HH = input("Enter the Hour(HH) to be
reminded : ")
                   evg_rem_tab2_time_MM = input("Enter the Minute(MM) to be
reminded : ")
                   print("")

                   if (evg_rem_tab_2 == tab_1):
                       schedule.every().day.at("%s:%s"
%(evg_rem_tab2_time_HH,evg_rem_tab2_time_MM)).do(tab_1_evg_reminder)
                   elif (evg_rem_tab_2 == tab_2):
                       schedule.every().day.at("%s:%s"
%(evg_rem_tab2_time_HH,evg_rem_tab2_time_MM)).do(tab_2_evg_reminder)

                   if (evg_rem_tab_2 != tab_1 and evg_rem_tab_2 != tab_2):
                       print("The tablet name you entered is not in pill box, again
enter the correct name")
                       print("")


                     else:
                       break



            if (evg_reminder != "1" and evg_reminder != "2"):
                print("Enter the value  as 1 or 2")
                print("")

            else:
                break

    if(evg_option != "YES" and evg_option != "yes" and evg_option != "no" and
evg_option != "NO"):
        print("Only type YES or NO")
        print("")
    else:
        break



while True:
    schedule.run_pending()
    sleep(1)
    Current_time = time.strftime("%H:%M:%S" , time.localtime())
    success = deviceCli.publishEvent("time", "json", { "Time" : " %s " % Current_time
}, qos=0)
    a = datetime.datetime.now()
    date = a.date()
    success = deviceCli.publishEvent("date", "json", { "Date" : " %s " % date }, qos=0)
```

GITHUB LINK :

https://github.com/IBM-EPBL/IBM-Project-40350-1660628509.git

PROJECT DEMO LINK :

https://youtu.be/6BEJnteoZ3U