PERSONAL ASSISTANCE FOR SENIORS WHO ARE SELF-RELIANT

PROJECT REPORT

TEAM ID	PNT2022TMID31785
FACULTY MENTOR	SURESH KUMAR C

TEAM MEMBERS:

VISHNU PRIYA C

AKSHAN T

NIZAR R

RONESH L

SANTHIYA S

Title Pg No

PROJECT REPORT

PERSONAL ASSISTANCE FOR SENIORS WHO ARE SELF-RELIANT

1.INTRODUCTION

1.1 Project overview

Whether you are managing your own absent-mindedness or taking care of a relative, medication reminders are one of the most useful aspects of daily medical care. From pill-minder trays to online calendars, a little help keeping track of multiple medicines can go a long way. At best, it is a way to save a little brainpower so you do not have to be super careful with your medication every time you take it. And in worst-case scenarios, effective medication reminders can be the difference between life and death.

Medication compliance is a serious problem – especially among older adults who battle with any number of difficulties from managing complicated medication regimes to impaired vision, arthritis, tremors, or dementia. Elders were facing difficulty with such daily activities as bathing, grooming, cooking, eating, or just getting to the bathroom often end up in hospitals or nursing homes, spending a disproportionately huge number of healthcare dollars, according to a report conducted for the U.S. Department of Health & Human Services. For far less money, perhaps 80 percent of these same elders could be made more self-reliant and healthier. New medication reminders, organizers, dispensers, alerts, planners, and apps are making huge inroads in the safer consumption of prescription drugs.

1.2 Purpose

Non-medical home care often includes the service of “medication reminders” for people who have prescription medicines to take. It might seem odd that someone needs a reminder when there are pill boxes with the days of the week and more, but it is an important part of well-being. Some people are living with cognitive impairment due to some form of dementia or trauma and truly cannot remember to take a pill. Medications often come with crucial instructions such as taking them at a certain time of day, with or without foods, taking them with or without certain liquids, keeping them at specific temperatures, and each of these needs to be adhered to for the medicine to work or minimize possible side effects.

2.LITERATURE SURVEY

2.1 Existing problem

- One downside of the medicine remainder is that developers provide anonymized, aggregated user data to healthcare companies. Although this datais anonymous, and can be helpful to developing better healthcare treatments and systems, some users may not like the fact that their data is being collectedand shared.

- On certain devices, notification reminders will be silenced when the device is on silent mode, so users must leave their sound on to get their reminders.

2.2 References

1. D. a. Clifton, D. Wong, L. Clifton, S. Wilson, R. Way, R. Pullinger, and L. Tarassenko. A large-scale clinical validation of an integrated monitoring system in the Emergency Department. *IEEE J. Biomed. Heal. Informatics* vol. 17, no. 4, pp. 835-842; 2013.

2. M. Parida, H.-C. Yang, S.-W. Jheng, and C.-J. Kuo. Application of RFID Technology for In-House Drug Management System.*15th Int. Conf. Network-Based Inf. Syst.*, pp. 577-581; 2012.

3. A. Sawand, S. Djahel, Z. Zhang, and F. Na. Multidisciplinary Approaches to Achieving Efficient and Trustworthy eHealth Monitoring Systems. Commun. China (ICCC), 2014 IEEE/CIC Int. Conf., pp. 187-192; 2014.

4. A. Kliem, M. Hovestadt, and O. Kao. Security and Communication Architecture for Networked Medical Devices in Mobility-Aware eHealth Environments," 2012 IEEE First Int. Conf. Mob. Serv., pp. 112-114;2012.

5. S. Huang, H. Chang, Y. Jhu, G. Chen the Intelligent Pill Box - Design and Implementation. (2014), pp. 235-236.

6. F.-T. Lin, Y.-C. Kuo, J.-C. Hsieh, H.-Y. Tsai, Y.-T. Liao, and H. C. Lee A Self-Powering Wireless Environment Monitoring System Using Soil Energy. *IEEE Sens. J.*, vol. 15, no. c, pp. 1-1; 2015.

7. D. H. S. D. Privacy and I. A. Committee. Report No. 2006-02 The Use of RFID for Human Identity Verification I. Introduction and Executive Summary II. RFID Technology Overview. Technology, pp. 1-13;20

2.3 Problem Statement Definition

In modern society, busy life has made people forget many things in day-to-day life. The elderly people and the people victims of chronicle diseases who need to take the medicines timely without missing are suffering from dementia, which is forgetting things in their daily routine. Considering this situation study has been done in this. The technologies of home health care which are currently used for improving this situation by reminding the scheduled of medicine, remote monitoring and update new medicine data of patients, which can be done by prescriber through web. Generally, for home-based health care the arrangement includes communications, imaging, sensing and human computer interaction technologies embattled at diagnosis, treatment, and monitoring patients without disturbing the quality of lifestyle.

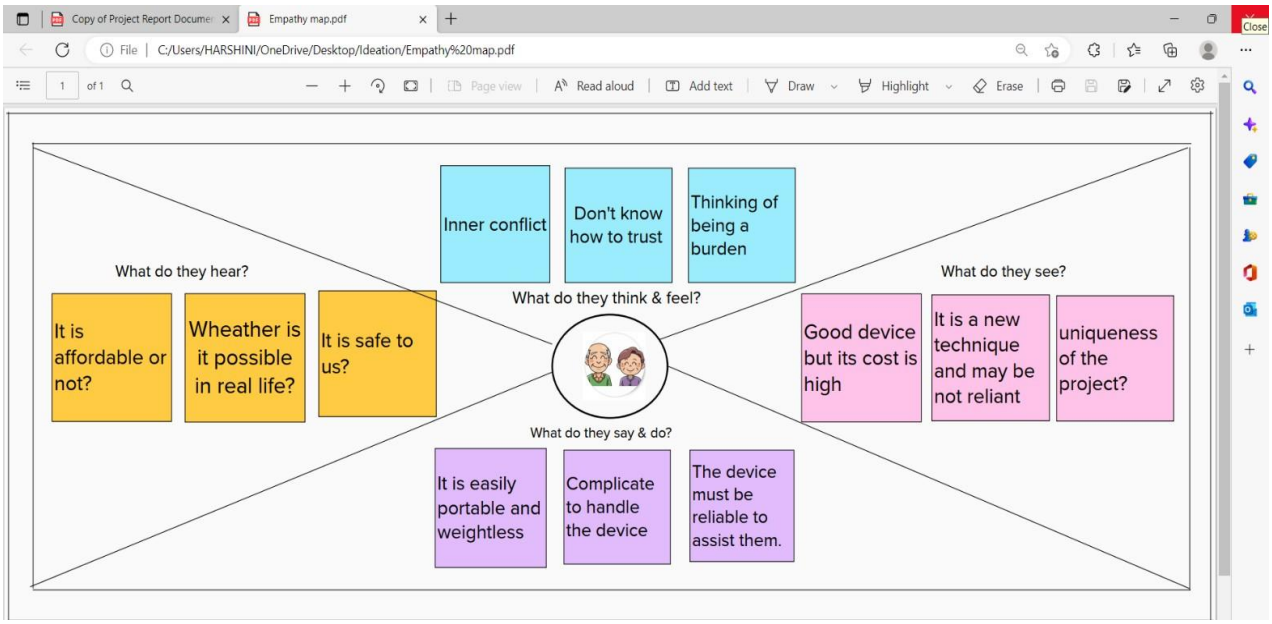
It can be possible the development of a low-cost medical sensing, communication and analytics device that is real-time monitoring internet allowed patients physical conditions. Internet of Things (IoT) network will provide active and real-time

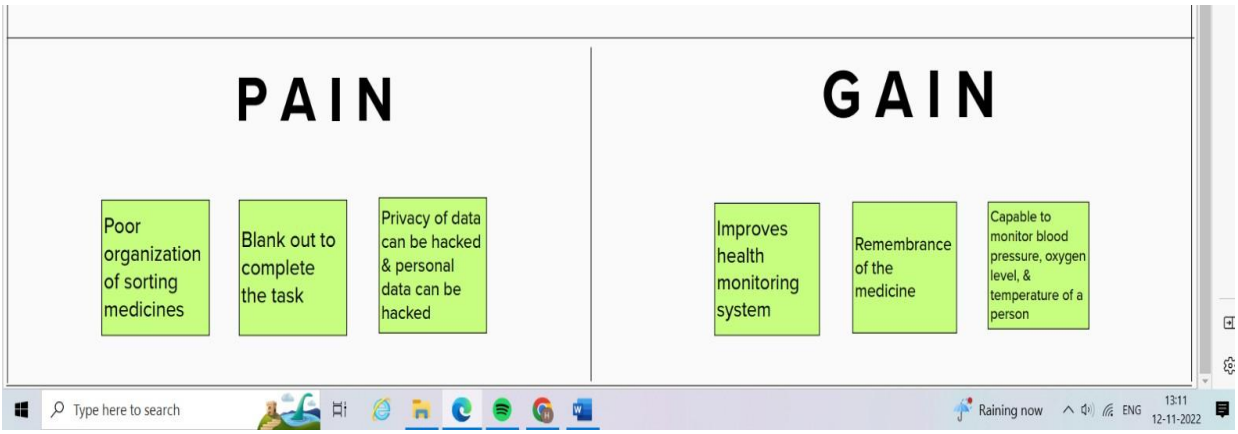
appointment of patient, hospitals, caretaker, and doctors apart from this the secured data transmission from source point to destination for the purpose of remote monitoring there is need of the architecture of a low-cost embedded platform for Web-based monitoring.

To overcome these problems, we have designed an application using node-red through cloud. This will have the required data of the patient details and their medicine report description. At the time of their medicine schedule, this system application (app) will notify to the user by displaying with the medicine name with voice command instead of an alarm.

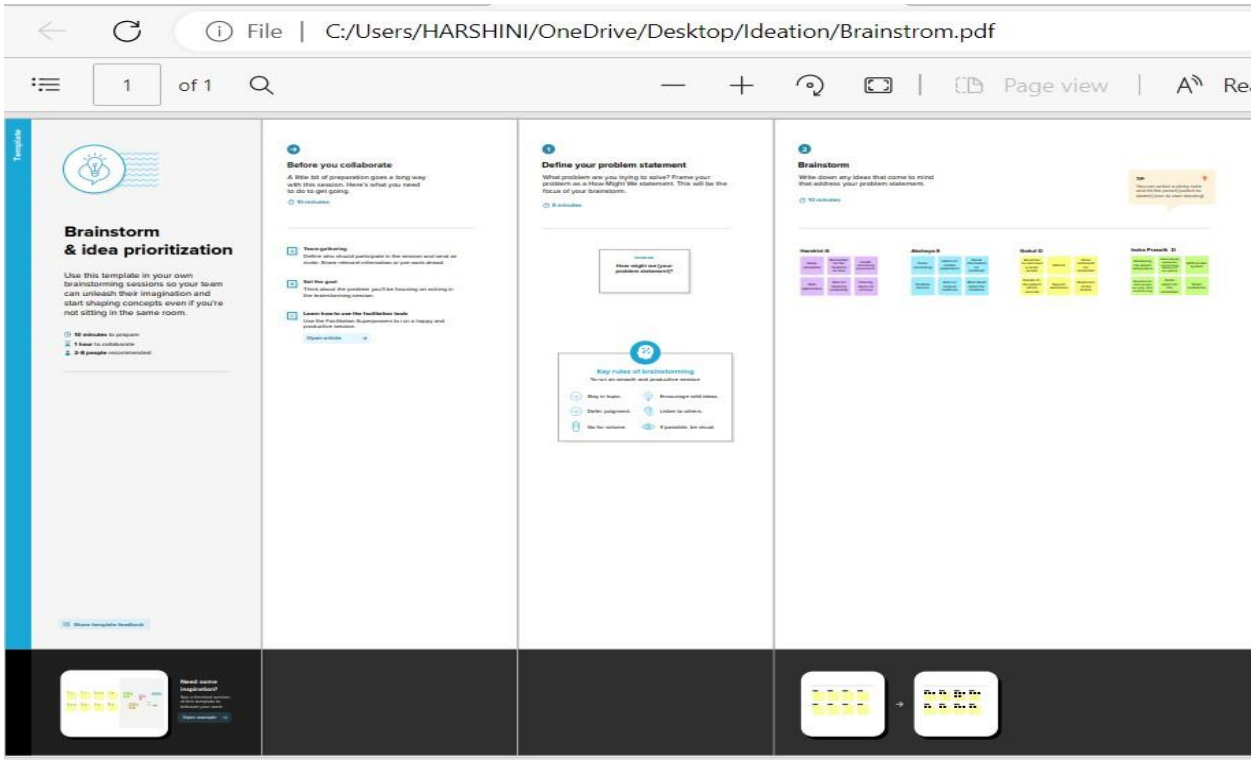
3. IDEATION & PROPOSED SOLUTION

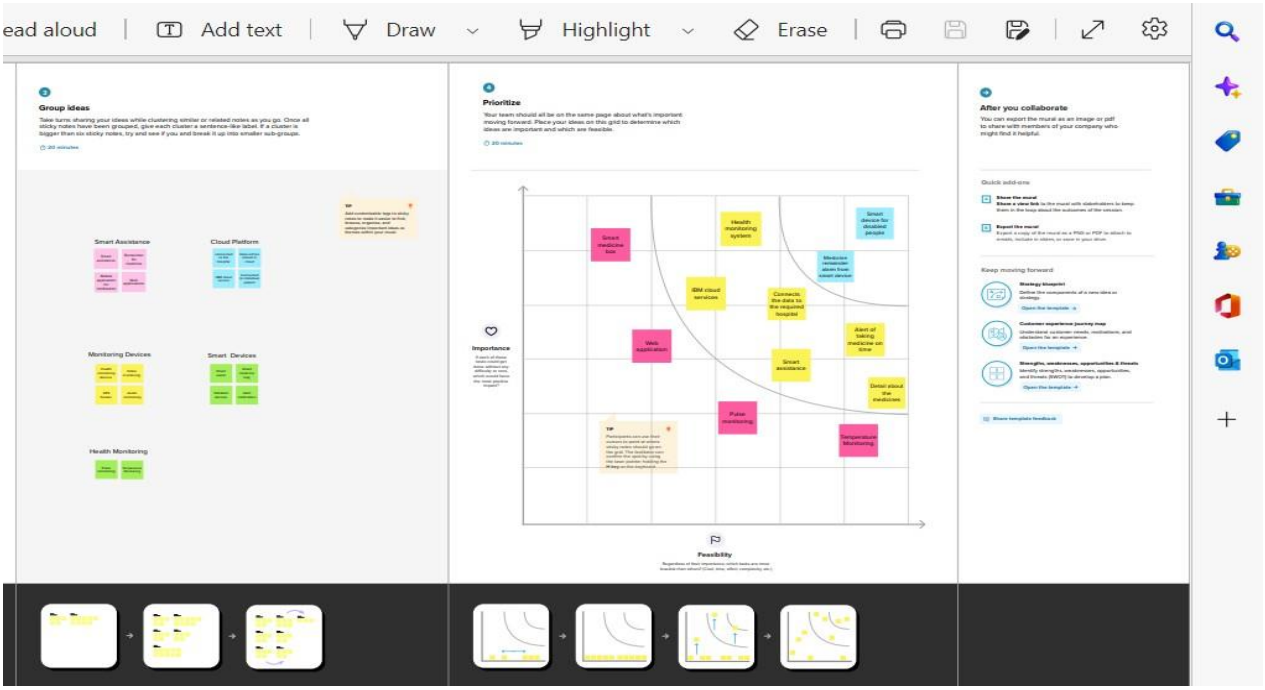
3.1 Empathy Map Canvas





3.2 Ideation & Brainstorming

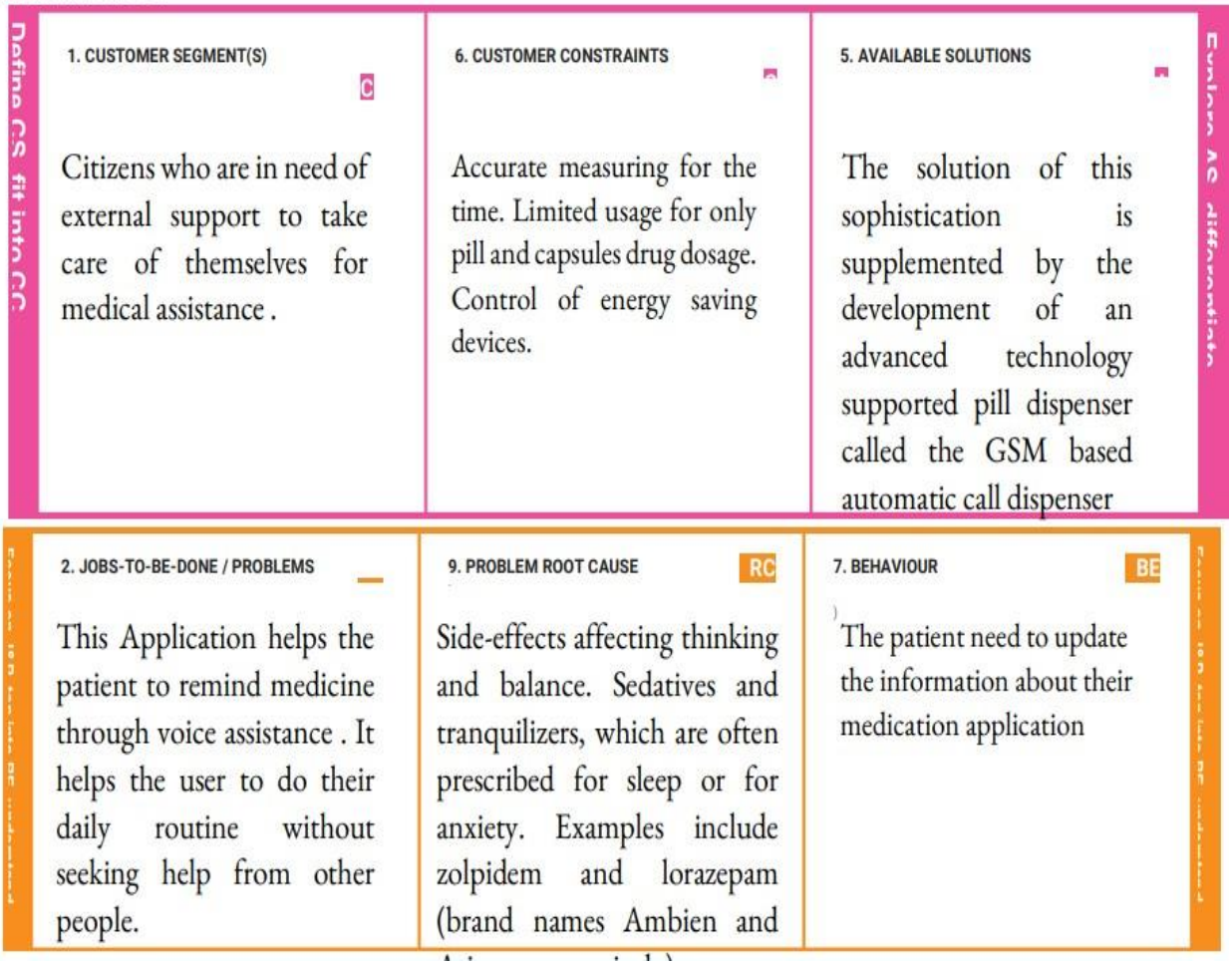


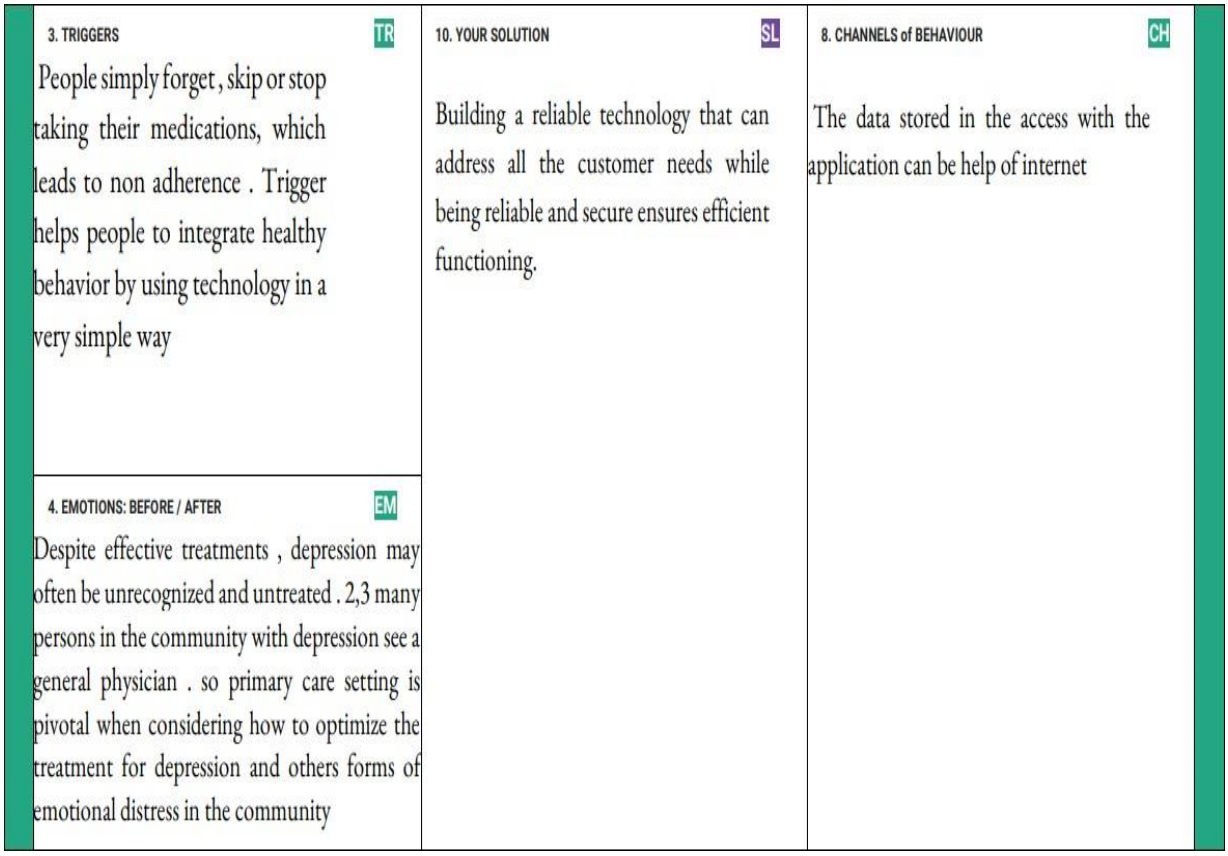


3.3 Proposed Solution

S.no	Parameter	Description
1.	Problem Statement	Now a days elders were facing many health issues, to overcome that they must take their regular scheduled medicine routine. By nature, as they become elder, they were forgetting to take their medicine on time. To overcome this problem medicine remainder app is created for the remembrance of medicine.
2.	Idea / Solution description	To overcome this problem, a web application is created with detailed information about the medicine and when the patient wants to take (time) the medicine. It can be notified by an alarm with vibration and voice command.
3.	Novelty / Uniqueness	Uniqueness in this project is, both the deaf and dump people can also be notified by the means of vibration so that they can also be benefited.
4.	Social Impact / Customer Satisfaction	All design decisions may be assessed and revaluated using these personas, keeping the user and their perspective in mind.
5.	Business Model (Revenue Model)	Through our web application the revenue can be made in the form of pop-up advertisements, overlay ads from third party services.
6.	Scalability of the Solution	Large number of people can be supplied with the wearable devices to ensure their safety and they can easily set their medication time in the web application.

3.4 Problem Solution fit





4. REQUIREMENT ANALYSIS

4.1 Functional requirement

1.Proper Medication - Proper Time for medication Intake of tablets
2.Tablets on Time - Remainder for tables Via Voice message
3.Disabled people – Vibration mode is designed for their requirement

4.2 Non-Functional requirements

Usability
Smart Medicine Box usability is the characteristics of the User Interface that facilitate Use, to make it easier for the users to perceive the information presented by the User Interface, to understand and decide based on that information.

Smart
Smart Medicine Box, like other computer systems, can be vulnerable to security breaches, potentially impacting the safety and effectiveness of the device.

Reliability
The probability of Smart Medicine Box will perform a required function without failure under static conditions for a specified period.

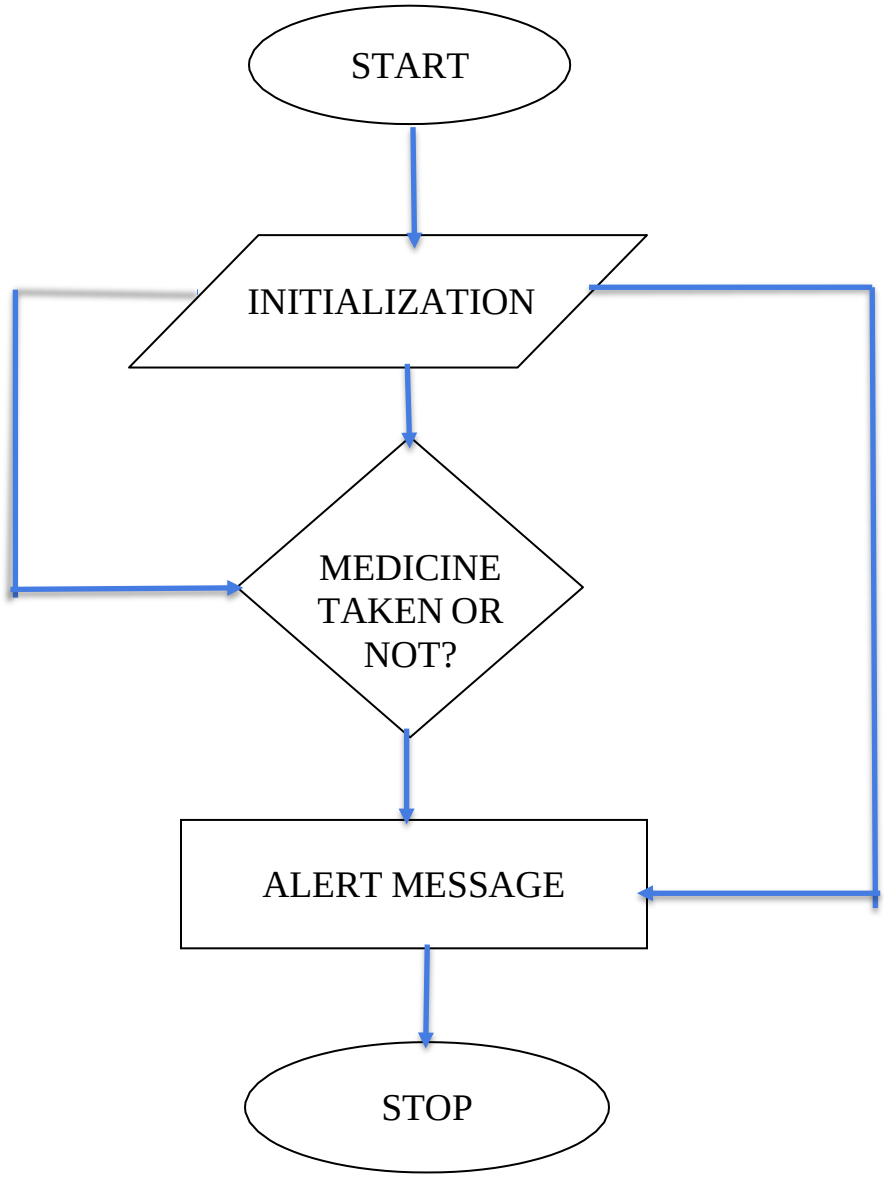
Performance
Medical device testing is the process of demonstrating that the device will reliably and safely perform in use.

Availability Our smart pill box is programmable that enables medical caretakers or clients to determine the pill amount and timing to take pills, and the service times for every day.

Scalability
In Feature, we can upgrade the smart medicine box to the health care assistant to monitor our healthcare and book appointments with doctors.

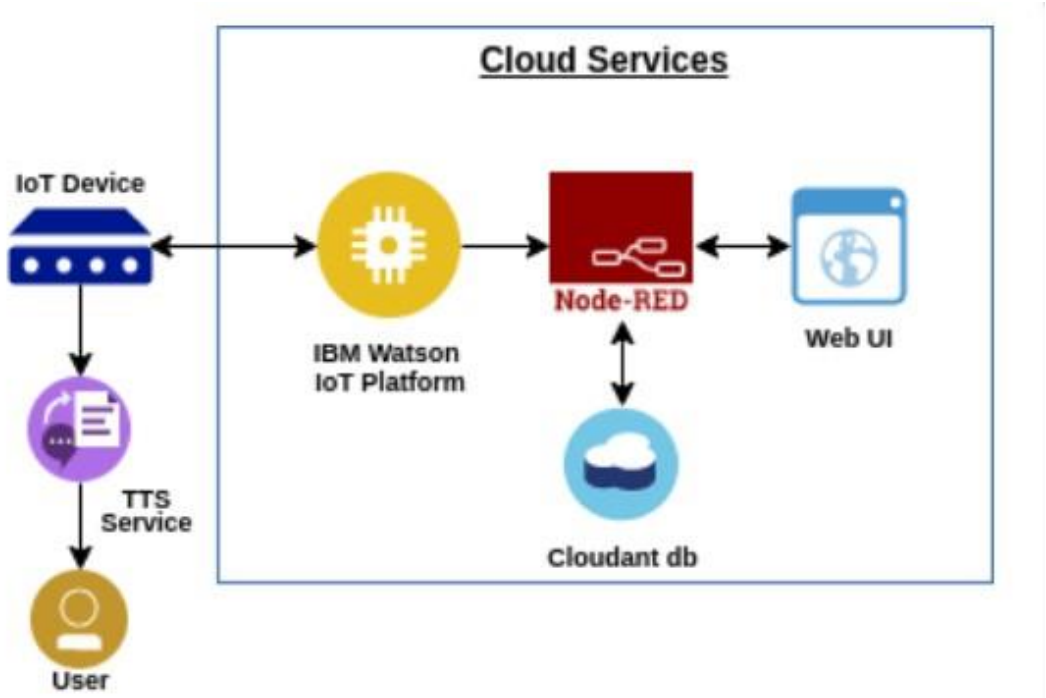
5. PROJECT DESIGN

5.1 Data flow Diagram



5.2 Solution & Technical Architecture

- ➤ As elders were not remembering of their tablet routine, they were facing many health issues.
- ➤ By use of this project, they need not to worry about their medicine remainder because this device automatically notifies the people to take their medicine on time.
- ➤ By creating portable web application, they were able to use this application without facing any diffabilities.
- ➤ As it remains the elder to take their medicine, their health will also maintain in good condition. ϖ 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 and vibrate.



6.PROJECT PLANNING & SCHEDULING

6.1 Sprint Planning & Estimation

Sprint 1

- ➢ Sprint one is about web Development, by which the user can access easily.
- ➢ As the app get started, it requires the user's detail so that they can store their data in cloud platform.
- ➢ Therefore, the user needs to enter their medicine details and their time to take off.

Sprint 2

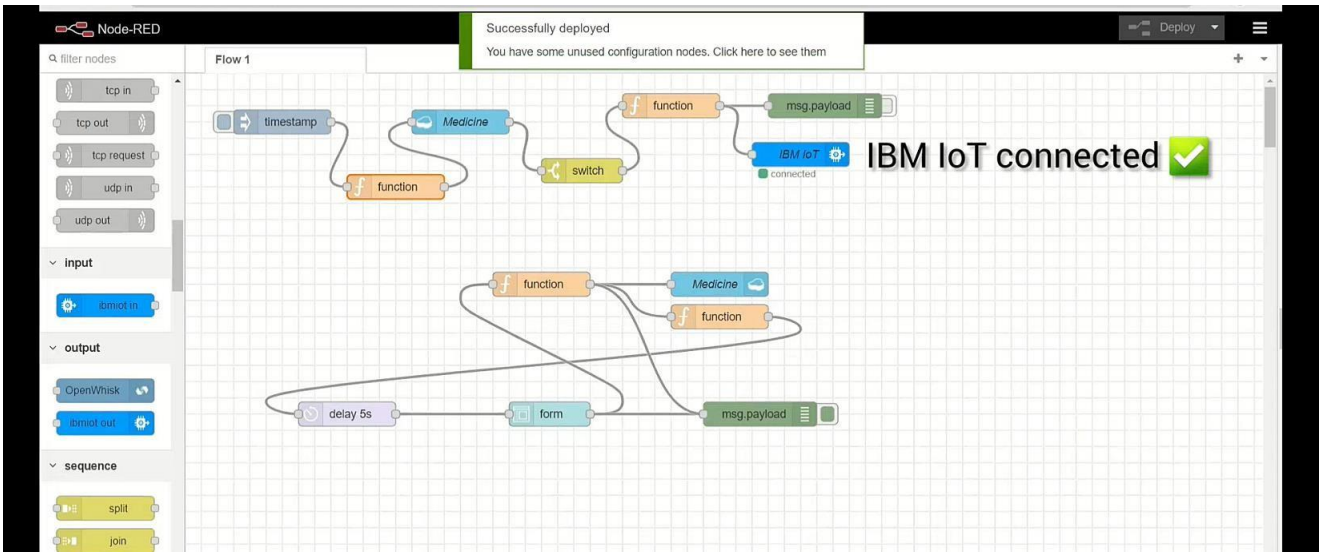
- ➤ To collect the details of the user, an admin page is created to store their details.
- ➤ And to get the details from the user and to transfer it the cloud platform an admin page is created using HTML code.
- ➤

Sprint 3

- ➤ In this sprint, already created app and admin page backend program is created.
- ➤ By using this node red, the remainder will automatically reach the user by voice command with the medicine name.
- ➤ The cloud database connections were made in this sprint.

Sprint 4

- ➤ Once all the connection were made, the testing process is deployed
- ➤ And as a result, the app notifies the user by voice command with the required medicine name.

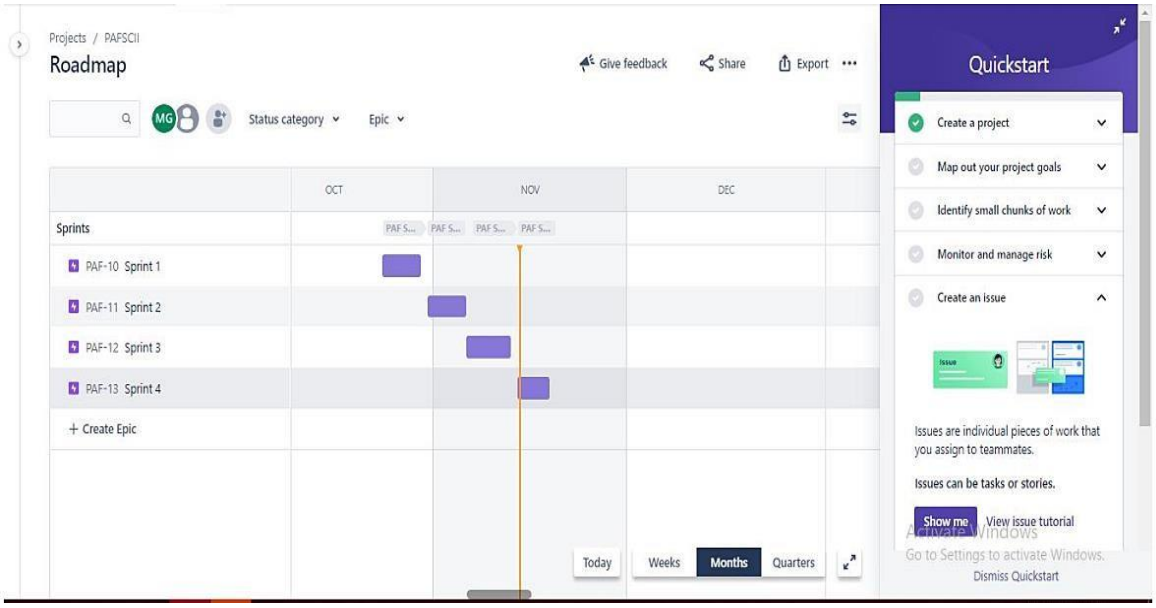


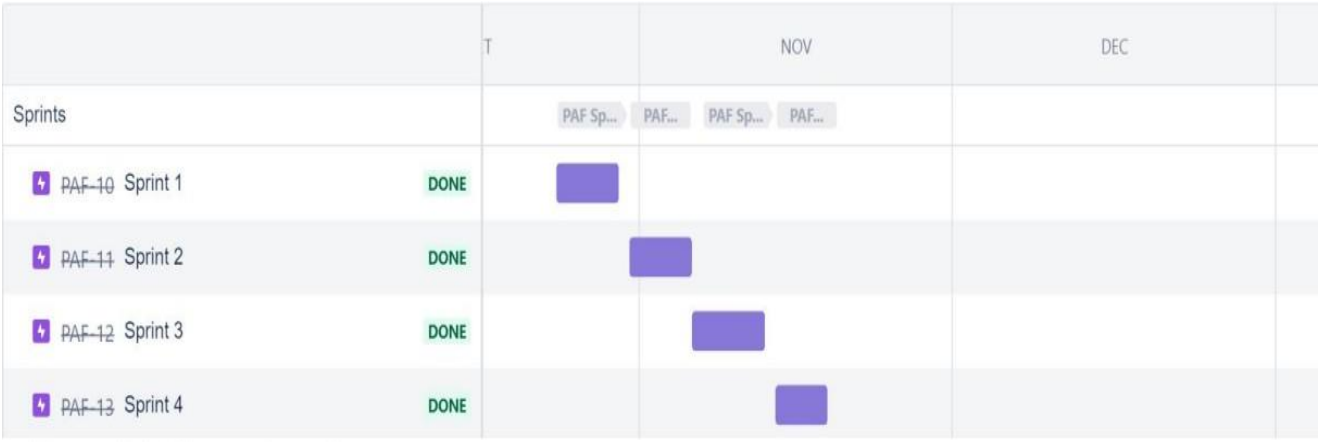
6.2 Sprint Delivery Schedule

Milestone Template:

Sprint	Sprint Topic	Start Date	Expected Delivery
Sprint 1	Web development	29-10-2022	5-11-2022
Sprint 2	Admin page	7-11-2022	14-11-2022
Sprint 3	Node red	16-11-2022	23-11-2022
Sprint 4	Deployment	23-11-2022	30-11-2022

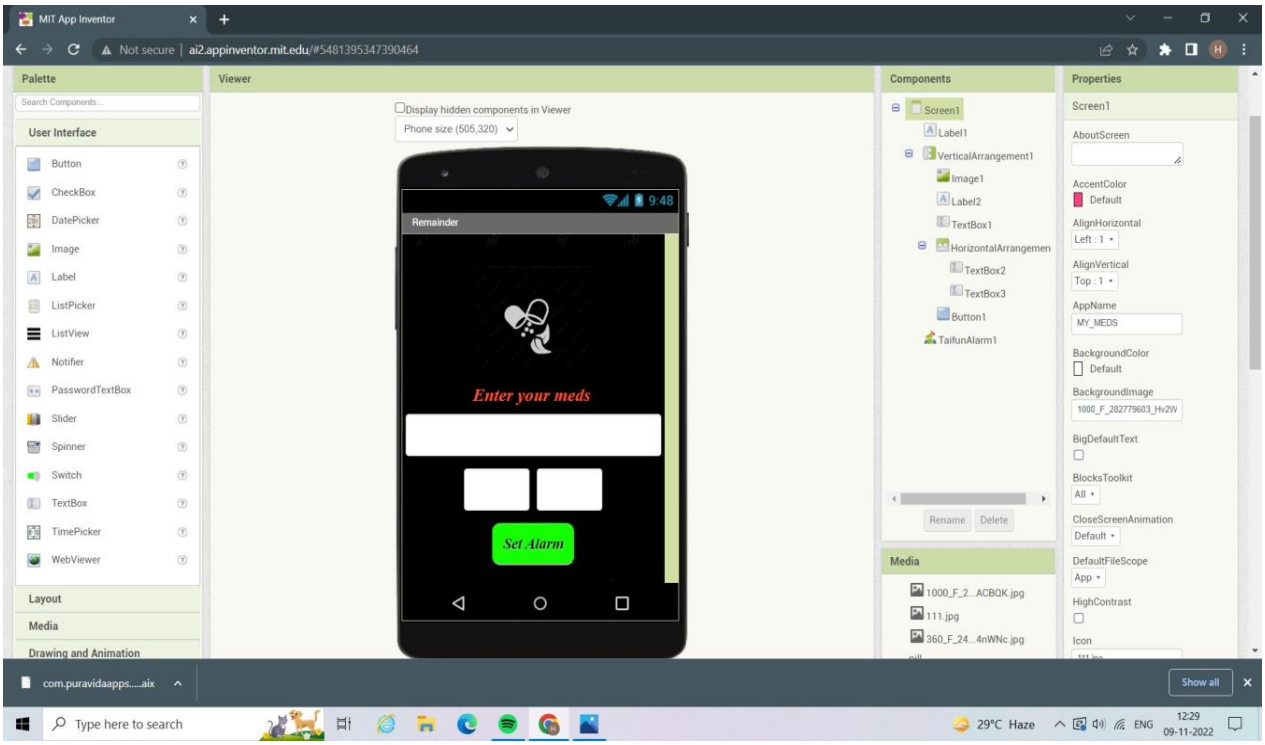
6.3 Reports from JIRA





7. CODING & SOLUTION

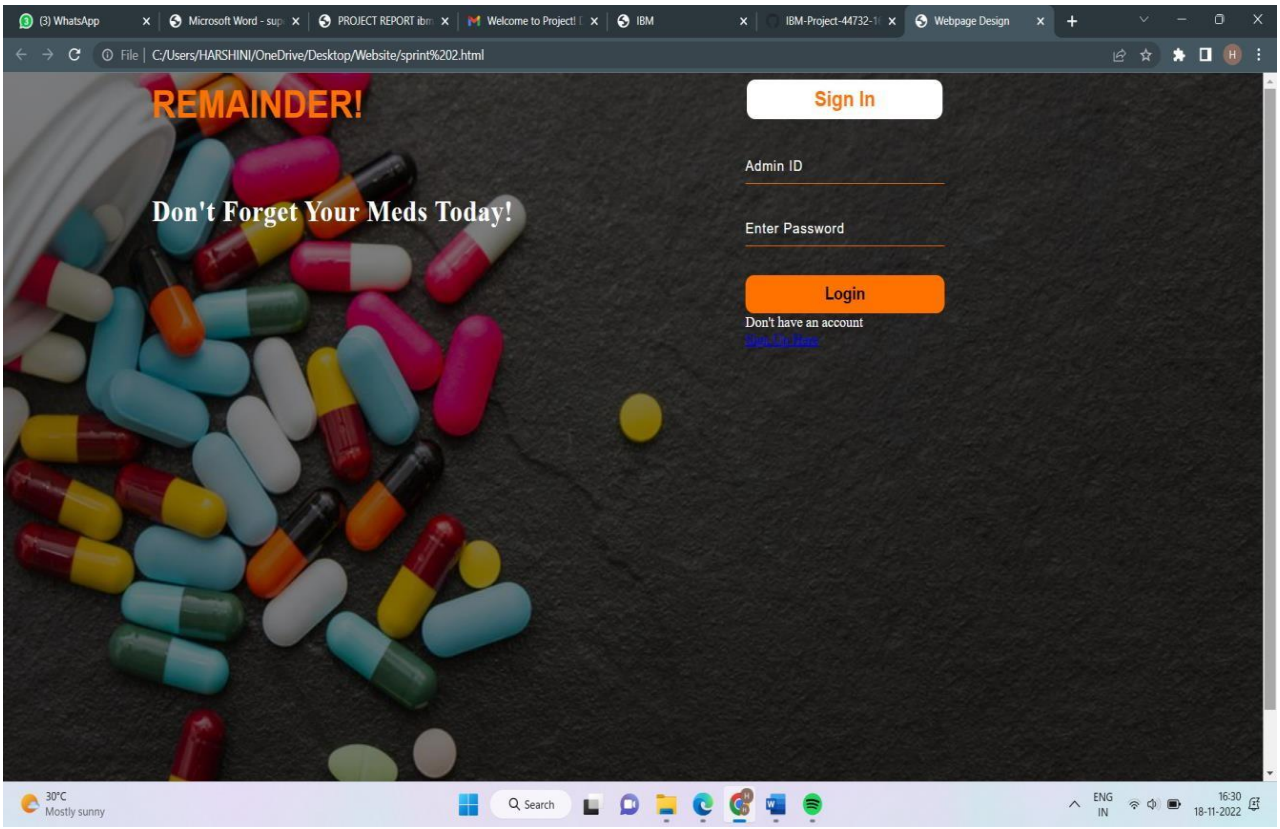
7.1 Feature 1



CODE

```
import './App.css'
import React, {useState, useEffect } from "react"
import axios from "axios"
import DateTimePicker from "react-datetime-picker"
```

7.2 Feature 2



CODE

```
*{
   margin: 0;
   padding: 0;

}

.main{
    width: 100%;
    background: linear-gradient(to top, rgba(0,0,0,0.5)50%,
rgba(0,0,0,0.5)50%),url(med.jpg);
    background-position: center;
    background-size: cover;
    height: 109vh;
}

.navbar{
    width: 1200px;
    height: 75px;
    margin: auto;
}

.icon{
    width: 200px;
    float:center;
    height: 70px;
}

.logo{
    color: #ff7200;
    font-size: 40px;
    font-family: Arial;
    padding-left: 20px;
    float: left;
    padding-top: 10px;
}
```

```css
.content{
    width: 1200px;
    height: auto;
    margin: auto;
    color: #fff;
    position: relative;

}

.content h1{
    font-family: 'Times New Roman';
    font-size: 30px;
    padding-left: 20px;
    margin-top: 2%;
    letter-spacing: 1px;

}

.form{
    width: 250px;
    height: 380px;
    background: linear-gradient(to top, #000000(3, 3, 3, 0.8)50%,#000000(3, 3, 3, 0.8)50%);
    color:#fff;
    position: absolute;
    top: -20px;
    left: 870px;
    border-radius: 10px;
    padding: 25px;

}

.form h2{
    width: 220px;
    font-family: sans-serif;
    text-align: center;
    color: #ff7200;
    font-size: 22px;
    background-color: #fff;
```

```
    border-radius: 10px;
    margin: 2px;
    padding: 8px;
}

.form input{
    width: 240px;
    height: 35px;
    background: transparent;
    border-bottom: 1px solid#ff7200;
    border-top: none;
    border-right: none;
    border-left: none;
    color: #fff;
    font-size: 15px;
    letter-spacing: 1px;
    margin-top: 30px;
    font-family: sans-serif;
}

.from input:focus{
    outline: none;
}
::placeholder{
    color: #fff;
    font-family: Arial;
}

.btrn{
    width: 240px;
    height: 40px;
    background: #ff7200;
    border: none;
    margin-top: 30px;
    font-size: 18px;
    border-radius: 10px;
    cursor: pointer;
    color: #fff;
    transition: 0.4s ease;
}
```

```css
.btrn:hover{
    background: #fff;
    color: #ff7200;
}

.btrn a{
    text-decoration: none;
    color: #000;
    font-weight: bold;
}

.form.link{
   font-family: Arial, Helvetica, sans-serif;
   font-size: 17px;
   padding-top: 20px;
   text-align:center;
}
.form.link a{
    text-decoration: none;
    color: #fff;
}


.liw{
    padding-top: 15px;
    padding-bottom: 10px;
    text-align:left;
    color:#fff;
}
```

8. TESTING

8.1 Test Cases

A test case might be created as an automated script to verify the functionality per the original acceptance criteria. After doing manual exploratory testing, QA testers might suggest other functionality be added to the application as well as updated test cases be incorporated in the automated test suite.

Test case ID	Feature Type	Component	Test Scenario
SMS Notification TC	Twilio SMS Notification	Node.JS (Server)	Verify user is able to Receive SMS when Users Medication Time arrives
Backend TC	App Configuration	Node.JS (Server)	It should get the data from the frontend and process data for Twilio SMS notification and DB store
Frontend TC	Dashboard UI	Home page (Client)	It should get the data from the frontend and process data for Twilio SMS notification and DB store
Datebase_TC_OO4	MongoDB	MongoDB (Server)	Verify that it can store the medicine reminder notes in the cloud DB

Test case ID	Feature Type	Component	Test Scenario	Steps To Execute	Test Data	Expected Result	Actual Result	Status	Commnets	TC for Automation	BUG ID	Executed By
LoginPage_TC_OO1	UI	Home Page	Verify whether user is able to access the URL	https://node-red-psifx-2022-11-13.au-syd.mybluemix.net/ui/#!/0?sock	URL	Able to access the URL	failed to access in mobile	Fail	Wrong Browser selected	NO	101	RUFUS AR
LoginPage_TC_OO2	UI	Home Page	Verify whether user is able to access the URL	https://node-red-psifx-2022-11-13.au-syd.mybluemix.net/ui/#!/0?sock	URL	Now User able to access the URL	Able to access in mobile	Pass	Able to access in Chrome and Edge	YES		MARK GERALD
LoginPage_TC_OO3	Functional	Home page	User can enter the data in specified	To have browsers to have enhanced capabilities	URL	enter a data in specified format	specified input is not	Fail	Specify the User formats	NO	110	KINGSTON LEONARD V
LoginPage_TC_OO4	Functional	Home page	User can enter the data in any format	User can enter the data in required format	Time(HH:MM): DATE(YYYY-MM-DD):	User can enter the data in	Input received	Pass	Format specified	YES		MELODINA CARNELIAN D
CLOUD_STORAGE_TC_005	Functional	Cloud	Verify if User input is stored in the cloud	User is able to access the URL with the given link. User has to enter the data(name,time and date) and click the SUBMIT button.Data to be stored in IBM cloud	MEDICINE NAME: Time(HH:MM) : DATE(YYYY-MM-DD):	User inputs has to be stored in cloud	Failed to storing the inputs	Fail	Cloud not connected properly	YES	111	RUFUS AR
CLOUD_STORAGE_TC_OO	Functional	Cloud	Verify if User input is stored in the cloud	User is able to access the URL with the given link. User has to enter the data(name,time and date) and click the SUBMIT button.Data to be stored in IBM cloud	MEDICINE NAME: Time(HH:MM) : DATE(YYYY-MM-DD):	User inputs has to be stored in cloud	Inputs are stored in the cloud	Pass	Cloud connected properly	YES		KINGSTON LEONARD V
OUPUT_TC_007	Functional	Iot device	Verify if it reminds the medicine intake to the user	Comparing the UTC time and medicine intake time	Real time and medicine intake time	Gives True when both times match	Null	Fail	Check the input	YES	113	RUFUS AR

Test case ID	Feature Type	Component	Test Scenario	Steps To Execute	Test Data	Expected Result	Actual Result	Status	Commnets	TC for Automation	BUG ID	Executed By
OUPUT_TC_007	Functional	Iot device	Verify if it reminds the medicine intake to the user	Comparing the UTC time and medicine intake time	Real time and medicine intake time	Gives True when both times match	Null	Fail	Check the input	YES	113	RUFUS AR
OUPUT_TC_007	Functional	Iot device	Verify if it reminds the medicine intake to the user	Comparing the UTC time and medicine intake time	Real time and medicine intake time	Gives True when both times match	TRUE	Fail	verified	Yes		MARK GERALD
TTS_TC_008	Functional	Iot device	Verify if it gives voice notifications	When True it gives a voice notifications	Voice notifications	Voice notifications	Voice notifications service didn't work	Fail	In program, commands are as object instead of string	NO	121	KINGSTON LEONARD V
TTS_TC_009	Functional	Iot device	Verify if it gives voice notifications	When True it gives a voice format notifications	Voice notifications	Voice notifications	Voice notifications	Pass	New string functions were added	YES		RUFUS AR
ACK_TC_010	Functional	URL	Verify whether the patient has taken the medicine or not	The TAKEN button has been included	The status of the medicine intake	The User clicks the TAKEN button to show that medicine has	Button is unfunctional	Fail	Error occurs due to failure of call and connect function of the "taken " button`	NO	132	MELODINA CARNELIAN D
ACK_TC_011	Functional	URL	Verify whether the patient has taken the medicine or not	The TAKEN button has been included	The status of the medicine intake	The User clicks the TAKEN button to show that	The Taken status is updated in	Pass	The status of the medicine intake is updated in the cloud	Yes		KINGSTON LEONARD V

8.2 User Acceptance Testing

Purpose of Document: The purpose of this document is to briefly explain the test coverage and open issues of the personal assistance for seniors who are Self-Reliant project at the time of the release to User Acceptance Testing (UAT). 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	5	3	1	2	11
Duplicate	2	1	0	0	3
External	2	2	0	0	4
Fixed	10	2	3	15	30
Not Reproduced	0	1	0	0	1
Skipped	0	0	2	0	2
Won't Fix	0	2	4	5	11
Totals	19	10	8	22	62

9. RESULTS

9.1 Performance Metrics

Detailed Test Plan

S.No	Project Overview	NFT Test approach	Assumptions/Dependencies/Risks	Approvals/Sign Off
1	Medicine Reminder Web -UI	Stress	App Crash/ Developer team/ Site Down	Approved
2	Medicine Reminder Web -UI	Load	Server Crash/ Developer team/ Server Down	Approved

End Of Test Report

Project Overview	NFT Test approach	NFR - Met	GO/NO-GO decision	Identified Defects	Approvals/ Sign Off
Medicine Reminder Web -UI	Stress	Performance	GO	Closed	Approved
Medicine Reminder Web -UI	Load	Scalability	NO-GO	Closed	Approved

10. ADVANTAGES & DISADVANTAGES

ADVANTAGES

- The software can help people set free from remembering the medication time and names.
- It helps the caretaker to determine the medication time, which can be variable sometimes, depending upon the patient's severity.
- The software is very user-friendly; the need not install any external app by the patient, economic for the caretaker too.
- The single software can be used by the caretaker for managing multiple patients at the same place.
- The details of the time scheduled, and patients' intake is stored in the database for future reference easily.
- The overall stress of patients and caretakers is reduced and maintained under control by the software.

DISADVANTAGES

- The software currently can only alert the patient to take medicine, we cannot ensure whether they have taken it or not.

- The software currently can only alert people with SMS, it cannot make phone calls to help the illiterate.

Elderly people should be aware of how to use the application.
There is no way to determine what happened as it only gives the remainder to take the medicine Internet connection is required.

11. CONCLUSION

This leads to the conclusion that, unless patients indicate otherwise, all patients should receive a reminder or 'reminder plus is that actively encourages patients who are unable to attend to cancel their appointment and to reschedule if further appointments are required. The reminder should be sent around 3 days in advance of the appointment as timing of a reminder, between 1 and 7 days prior to the scheduled appointment, has no effect on patient attendance behaviour. This will allow sufficient time for patient cancellation and health service reallocation of the appointment to another patient or allow the clinician to undertake care-related administrative tasks. To optimise attendance, cancellation and rescheduling there needs to be robust procedures to ensure that patient contact details are up to date and that there are easy to use, probably multiple, systems for cancelling appointments that suit the needs of the patients, e.g., automated SMS cancellation, answerphone, e-mail, etc. Robust 24-hours-per-day rescheduling procedures should allow easy rescheduling of appointments for patients. Finally, an effective reminder system will increase the workload on clinical staff and alternative time will need to be scheduled for staff to undertake health-care-related administration. Further research is required to investigate the differential effectiveness and cost-effectiveness of an 'optimised' reminder system over and above usual reminder systems.

There were few studies investigating the differential effectiveness of alternative types of reminders for different segments of the population and this we believe is a key area for further research. Nevertheless, we have used the findings of our review to suggest possible reminder alternatives for key groups of patients who appear to be at higher risk of not attending appointments, namely deprived, ethnic, substance abusers, and those with comorbidities and illness. Simple reminders and automated reminders to attend may be ignored or overlooked and may put these patient groups at a disadvantage compared with general outpatient populations. Reminders with direct personal contact might be appropriate in these groups. To facilitate attendance in the most at risk, vulnerable groups we have suggested that reminder systems of increasing intensity and interactivity could be introduced to ensure that disparities in health-care opportunities are not compounded. We have introduced the concept of a sequential reminder intervention in order to reach the greatest number of patients and maximise attendance, although their effectiveness in this context needs to be established. The re-engagement of these patient groups with treatment after they have missed their appointment may be important if they have health problems that need ongoing treatment. Intensive approaches, such as 'stepped reminders and patient navigators have been effective at increasing attendance at screening and

immunisation programmes in disadvantaged and vulnerable populations, although their effectiveness in this context needs to be investigated.

Reminder systems are a complex intervention, because of the potential number of interacting components within the interventions, the requirement for tailoring of the intervention to the health service and the number of difficulties and behavioural changes from those receiving and delivering the reminder. Therefore, in addition to following the general recommendations provided above, health service managers will need to tailor their reminder systems to meet the needs of the service and the patient population that it serves. This review provides a range of findings that will inform health service managers' decision-making processes. To this end, we are producing a practice guide to help health service managers consider specific issues that may be relevant to the design of reminder systems for their health service.

12. FUTURE SCOPE

- ➤ The project can be enhanced with many other features that can serve senior citizens even better. The product currently is a simple basic version which can only send SMS alerts on time. Some other additional features that are planned to be incorporated with this existing product are listed below:
- ➤ The dashboard can be made more versatile for the caretakers to manage patients medicine intake time and to monitor how it changes every day, by this a new or speculated time can be scheduled individually.
- ➤ The system can be enhanced with a smartwatch or health devices so that the health conditions can be continuously connected with the hospitals, and doctors to supervise and help them during emergencies.
- ➤ The system can relate to hardware product that stores and automatically opens the container and alerts with a voice message.

The system can further relate to the medical shop so that the hardware system automatically senses the tablet counts and alerts the medical shop to deliver the medicine.

SOURCE CODE

```
import  './App.css'
import React, { useState,
useEffect } from "react"import
axios from "axios"
import DateTimePicker from "react-
datetime-picker"function App () {
const [ reminderMsg, setReminderMsg
] = useState("")const [ remindAt,
setRemindAt ] = useState()
const [ reminderList, setReminderList
] = useState([])useEffect(() => {
axios.get("http://localhost:9000/getAllReminder").then(
\=>setReminderList(res.data))
}, [])
const addReminder = () => {
axios.post("http://localhost:9000/addReminder", {
reminderMsg, remindAt})
.then( res
=>setReminderList(res.
data))
setReminderMsg("")
setRemindAt()
```

```
        }
        const deleteReminder = (id) => {
        axios.post("http://localhost:9000/deleteR
        eminder", { id })
        . then(res =>setReminderList(res.data))
         }
         return (
        <div className="App">

<div className="homepage">
        <div className="homepage_header">
        <h1>Medicine Reminder </h1>
        <input type="text" placeholder="Reminder notes here..."
        value={reminderMsg}onChange={e
        =>setReminderMsg(e.target.value)} />
        <DateTimePicker
        value={rem
        indAt}
        onChange=
        {setRemind
        At}
        minDate={n
        ew Date()}
        minutePlac
        eholder="
        mm"
        hourPlaceh
```

```
older="hh"
dayPlaceho
lder="DD"
monthPlac
eholder="
MM"
yearPlaceh
older="YYY
Y"

<div className="button" onClick={addReminder}>Add Reminder</div>
</div>
<div
className="homepa
ge_body">
reminderList.map(
reminder=> (
<div className="reminder_card" key={reminder._id}>
<h2>{reminder.reminderMsg}</h2>
<h3>Remind Me at:</h3>
<p>{String(new Date(reminder.remindAt.toLocaleString(undefined,
{timezone:"Asia/Kolkata"})))}</p>
<div className="button" onClick={()
=>deleteReminder(reminder._id)}>Delete</div>
</div>
</div>
```

```
</div>
</div>

require('dotenv').config()
const express =
require("express")
const mongoose =
require("mongoose")
const cors =
require("cors")

//APP config
const app =
express()
app.use(expre
ss.json())
app.use(expre
ss.urlencoded(
))
app.use(cors()
)

//DB config
mongoose.connect('mongodb://127.0.0.1:27017/IBM-
Prototype_DB', {useNewUrlParser: true,

useUnifiedTopology: true
}, () =>console.log ("DB connected"))
```

```
const reminderSchema = new
mongoose.Schema({reminderMsg:
String,
remindAt:
String,
isReminded
: Boolean
})
const Reminder = new mongoose.model("reminder", reminderSchema)
//Whatsapp
reminding
functionality
setInterval(() => {
Reminder.find({},  (err,
    reminderList) => {
    if(err) {
        console.log(err)
    }
if(reminderList){
reminderList.forEach(remin
der => {
            if(!reminder
                .isRemind
                ed){const
```

```
                now =
                new Date
                ()
if((new Date(reminder.remindAt) - now) < 0) {
Reminder.findByIdAndUpdate(reminder._id,
{isReminded: true}, (err,remindObj)=>{
                    if(err){
                        console.log(err)
                    }
                    const client =
require('twilio')('ACed0ea1d4fae9d7375672d0742331e96b','dcc8fb9
228ae68d156727d7ed5f656b2');
client.messages
.create({
                                body:
                                reminder.remi
                                nderMsg,to:
                            })
.then(message => console.log(message.sid))
.done();
                    })
                }
            }
        })
    }
  })
```

```
},1000)
//API routes
app.get("/getAllReminder", (req, res)
=> {Reminder.find({}, (err,
reminderList) => {
    if(err){
    console.log(err)
    }
    if(reminderList){ res.send(reminderList)
    }
    })
    })
    app.post("/addReminder", (req, res) => {
    const { reminderMsg, remindAt } = req.body const reminder = new
    Reminder ({
    reminderMsg, remindAt, isReminded: false
    })
    reminder.save(err => { if(err){
    console.log(err)
    }
    Reminder.find({}, (err, reminderList) => { if(err){
    console.log(err)
    }
    if(reminderList){ res.send(reminderList)
    }
    })
```

```
})
})
app.post("/deleteReminder", (req, res) => { Reminder.deleteOne({_id:
req.body.id}, () => { Reminder.find({}, (err, reminderList) => {if(err){
console.log(err)
}
if(reminderList){ res.send(reminderList)
}
})
})
})
app.listen(9000, () => console.log("Be started"))
```

HTML CODE

```
*{
   margin: 0;
   padding: 0;

}

.main{
    width: 100%;
    background: linear-gradient(to top, rgba(0,0,0,0.5)50%,
rgba(0,0,0,0.5)50%),url(med.jpg);
    background-position: center;
    background-size: cover;
    height: 109vh;
}

.navbar{
    width: 1200px;
    height: 75px;
    margin: auto;
```

```
}

.icon{
    width: 200px;
    float:center;
    height: 70px;
}

.logo{
    color: #ff7200;
    font-size: 40px;
    font-family: Arial;
    padding-left: 20px;
    float: left;
    padding-top: 10px;
}

.content{
    width: 1200px;
    height: auto;
    margin: auto;
    color: #fff;
    position: relative;

}

.content h1{
    font-family: 'Times New Roman';
    font-size: 30px;
    padding-left: 20px;
    margin-top: 2%;
    letter-spacing: 1px;

}

.form{
    width: 250px;
    height: 380px;
    background: linear-gradient(to top, #000000(3, 3, 3, 0.8)50%,#000000(3, 3, 3,
0.8)50%);
```

```
    color:#fff;
    position: absolute;
    top: -20px;
    left: 870px;
    border-radius: 10px;
    padding: 25px;

}

.form h2{
    width: 220px;
    font-family: sans-serif;
    text-align: center;
    color: #ff7200;
    font-size: 22px;
    background-color: #fff;
    border-radius: 10px;
    margin: 2px;
    padding: 8px;
}

.form input{
    width: 240px;
    height: 35px;
    background: transparent;
    border-bottom: 1px solid#ff7200;
    border-top: none;
    border-right: none;
    border-left: none;
    color: #fff;
    font-size: 15px;
    letter-spacing: 1px;
    margin-top: 30px;
    font-family: sans-serif;
}

.from input:focus{
    outline: none;
}
::placeholder{
```

```
    color: #fff;
    font-family: Arial;
}

.btrn{
    width: 240px;
    height: 40px;
    background: #ff7200;
    border: none;
    margin-top: 30px;
    font-size: 18px;
    border-radius: 10px;
    cursor: pointer;
    color: #fff;
    transition: 0.4s ease;
}

.btrn:hover{
    background: #fff;
    color: #ff7200;
}

.btrn a{
    text-decoration: none;
    color: #000;
    font-weight: bold;
}

.form.link{
   font-family: Arial, Helvetica, sans-serif;
   font-size: 17px;
   padding-top: 20px;
   text-align:center;
}
.form.link a{
    text-decoration: none;
    color: #fff;
}
```

```
.liw{
    padding-top: 15px;
    padding-bottom: 10px;
    text-align:left;
    color:#fff;
}
```

GITHUB LINK: https://github.com/IBM-EPBL/IBM-Project-44732-1660726502

DEMO VIDEO:https://drive.google.com/file/d/1wrIAE9TvPAMCcI128dc74P3Y7xXbCsQg/view?usp=drivesdk