

**NAALAIYA THIRAN PROJECT - 2022
PROFESSIONAL READINESS FOR
INNOVATION, EMPLOYABILITY AND
ENTREPRENEURSHIP**

**PERSONAL ASSISTANCE FOR SENIORS WHO ARE
SELF-RELIANT**

A PROJECT REPORT

Submitted by

DHARANI T	923019106005
GOKUL T	923019106008
KAVIRATHNA S	923019106009
NAVEEN S	923019106012
SNEGA S	923019106013

**DEPARTMENT OF ELECTRONICS AND COMMUNICATION
ENGINEERING**

*In partial fulfillment for the award of the degree
of
BACHELOR OF ENGINEERING
In*

ARULMURUGAN COLLAGE OF ENGINEERING - THENNILAI, KARUR - 639 206

ANNA UNIVERSITY: CHENNAI 600025
NOVEMBER 2022

**ANNA UNIVERSITY : CHENNAI 600 025
BONAFIDE CERTIFICATE**

Certified that this project report “ ***PERSONAL ASSISTANCE FOR SENIORS
WHO ARE SELF-RELIANT*** ” bonafide work of

DHARANI T	923019106005
GOKUL T	923019106008
KAVIRATHNA S	923019106009
NAVEEN S	923019106012
SNEGA S	923019106013

Who carried out the project work under my supervision.

SIGNATURE

Mr.S.NAVIN M.E.,
FACULTY EVALUATOR
Assistant Professor
Department of Electronics
and Communication
Engineering

SIGNATURE

Mr.M.SENTHILKUMAR M.E.,
FACULTY MENTOR
Assistant profesor
Department of Electronics
and Communication
Engineering

SIGNATURE

Mrs.S.YUVARANI M.E.,
FACULTY SPOC
Assistant Profesora
Department of Electronics
and Communication
Engineering

ABSTRACT

Most old people have multiple medicines to take to overcome their illnesses. However, they often forget to take their prescribed medicine on time, making it difficult for the caretakers to keep tabs on the patients and diagnose them in the right manner. Such situations may sometimes escalate to life-threatening ones. Medication reminders serve as a good way to stay on track and uphold an appropriate schedule. Ensuring that you or your loved one is properly taking their medications can help to avoid unnecessary risk and serious illness.

To avoid this, we have built an application (both web and mobile application) which enables the user to set reminders with medicine time and

dosage levels and the same features have been incorporated into the mobile app for the convenience of the user. The mobile app alerts the user with the voice command and displays the name of the medicine to be in taken on that particular time. The web and mobile application displays all their medicine data (medicine name, date and time) and it also allows the user to add or delete the data they have entered. Both the web and mobile applications were tested for various users and test cases and it worked entirely fine for the test cases.

Project Report Format

1. INTRODUCTION

1.1 Project Overview

1.2 Purpose

2. LITERATURE SURVEY

2.1 Existing problem

2.2 References

2.3 Problem Statement Definition

3. IDEATION & PROPOSED SOLUTION

3.1 Empathy Map Canvas

3.2 Ideation & Brainstorming

3.3 Proposed Solution

3.4 Problem Solution fit

4. REQUIREMENT ANALYSIS

4.1 Functional requirement

4.2 Non-Functional requirements

5. PROJECT DESIGN

5.1 Data Flow Diagrams

5.2 Solution & Technical Architecture

5.3 User Stories

6. PROJECT PLANNING & SCHEDULING

6.1 Sprint Planning & Estimation

- 6.2 Sprint Delivery Schedule
- 6.3 Reports from JIRA
- 7. CODING & SOLUTIONING (Explain the features added in the project along with code)
 - 7.1 Feature 1
 - 7.2 Feature 2
 - 7.3 Database Schema (if Applicable)
- 8. TESTING
 - 8.1 Test Cases
 - 8.2 User Acceptance Testing
- 9. RESULTS
 - 9.1 Performance Metrics
- 10. ADVANTAGES & DISADVANTAGES
- 11. CONCLUSION
- 12. FUTURE SCOPE
- 13. APPENDIX
 - Source Code
 - GitHub & Project Demo Link

CHAPTER 1

1.INTRODUCTION

This chapter gives an overview of the project and presents the need and objective of the project.

1.1PROJECT OVERVIEW

In our day to day life, due to busy schedules and workload, people often forget to take their medicines on time. Especially old aged people who have illnesses and who are illiterate have problems while taking the medicine, and sometimes it's not possible for the family members to give them medicine at prescribed time. There might be chances of them taking wrong medicines because

of poor eyesight. It is also possible that they might take an extra dosage of the same medicine, so this may lead to another medical condition which is not desirable. In order to stabilize their health condition they need to take the right medicines at the right time.

In recent years IOT plays an important role in making devices which are very helpful in our day to day life. So to solve the above issue by using IOT, we propose a medicine reminder system with the help of which the person is supposed to take the pill at the respective time. Hence the objective of this project is to

design and develop a medicine reminder system with medicine details, dosage levels and specified timings.

1.2 PURPOSE

Most old people have multiple medicines to take to overcome their illnesses. However, they often forget to take their prescribed medicine on time, making it difficult for the caretakers to keep tabs on the patients and diagnose them in the right manner. Such situations may sometimes escalate to life-threatening ones. Medication reminders serve as a good way to stay on track and uphold an appropriate schedule. Ensuring that you or your loved one is properly taking their medications can help avoid unnecessary risk and serious illness. So, 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 database.

CHAPTER 2

2.LITERATURE SURVEY

2.1.Existing program

- **PAPER 01 : IRCCS INRCA-National Institute of Health and Science on Ageing**
- **PUBLICATION YEAR : 2022**
- **AUTHOR : MARIA GABRIELLA MELCHOIRE**

Caring help is essential for carrying out everyday activities when older persons age alone and become weak with functional limitations. The current study set out to examine the role and features of privately employed Personal Care Assistants (PCAs) who provide care for elderly people in Italy in light of the family's decreasing capacity to provide care and the under-resourcing of governmental services. In the "Inclusive ageing in place" (IN-AGE) project, 120 qualitative interviews with elderly persons in their homes in the Italian regions of Lombardy, Marche, and Calabria were conducted in 2019. Along with some basic quantifications of assertions, a content analysis was done. Results revealed that PCAs were helpful in 27 situations, mostly when older citizens' health difficulties were raised.

- **PAPER 02 : JMIR M health U health**
- **PUBLICATION YEAR : 2021**
- **AUTHOR : GUNTHER EYSENBACH**

With the benefits of hands-free and eyes-free engagement modalities to manage requests, voice assistants based on smart speakers promise to support the elderly population. The advantages of this kind of gadget are seen differently by older persons, although little is known about this. The ease of a speech-based engagement contributed to the favourable first reception to voice assistants. Particularly, it was common to finish an engagement with a voice assistant by expressing gratitude or providing criticism on the quality of the responses. Asking queries about health care and streaming music were the two main themes of orders given during the first conversation. However, the majority of the subsequent responses were negative due to the challenges in creating a structured language for a command.

- **PAPER 03 : IEEE Pune Section International Conference (PuneCon)”**
- **PUBLICATION YEAR : 2021**
- **AUTHOR : Mithra Venkatesan**

The robot for the elderly discussed in this essay is made up of numerous electrical components that can be changed in the future and utilised to create new robotic appliances that may be used in a domestic setting. A personal assistant robot called "Robo care for Elderly" is a prototype that will one day be utilized to care for and accompany the elderly. The Raspberry Pi microcomputer, an ultrasonic sensor, a PIR sensor, a temperature sensor, LEDs, an integrated Bluetooth module, a Dc motor, a servo motor, speakers, etc. are all part of this system. The major goal of the created work is to create a personal assistant robot prototype that is affordable and usable in every home, improving the usage of technology.

- **PAPER 04 : International Workshop on Intelligent Virtual Agents**
- **PUBLICATION YEAR : 2017**
- **AUTHOR : Ramin Yaghoubzadeh**

Torky Cognitively impaired individuals struggle to independently plan their everyday activities. A virtual agent could be a helpful daily calendar aide, but this requires that these particular user groups accept the system and can communicate effectively with it. In this study, studies that address these issues for older users and users with cognitive impairment are presented. Results from focus groups and interviews indicate that using a participatory design approach can boost acceptance. The viability of spoken-language interaction is shown through actual interaction studies with a prototype, which also disclose mitigation techniques for comprehending issues.

- **PAPER 05 : International Conference on Universal Access in Human-Computer Interaction**
- **PUBLICATION YEAR : 2021**
- **AUTHOR : Arsénio Reis**

One of the key contributors to a person's life quality degrading as their ageing process progresses is social isolation and loneliness. These factors, which are brought on by the person's decreased social engagement with their friends, family, and former coworkers groups, can have a significant impact on their general health. On the other hand, software and hardware technologies have advanced to the point where electronic assistants can now both speak with users using natural voice language and gather information from them via camera photos. In this regard, a paradigm for the elderly's acceptance of electronic intelligent assistants has been put forth in prior research. In the current study, it is evaluated whether employing.

- **PAPER 06 : Universidad de Granada, E.T.S. of Computer and Telecommunication Engineering, Granada, Spain**
- **PUBLICATION YEAR : 2022**
- **AUTHOR : Manuel Bolanos**

Because of the trend toward higher population growth worldwide, some authors agree that older people experience social and technological isolation, if not outright exclusion, as a result of their ageing condition. Therefore, studies are required to identify the expectations of this population in terms of the usage and adoption of technology. As a result, new technological developments implement specific requirements that aid older people in adjusting to their use. This essay discusses a study conducted to assess how well-liked smart virtual assistants are

among the elderly. Considering certain experiences in the development and implementation of technology for this kind of study, the design and execution of a recreational strategy to remember taking drugs.

- **PAPER 07 : Information Systems and eBusiness Management**

- **PUBLICATION YEAR : 2018**

- **AUTHOR : Heetae Yang and Hwansoo Lee**

The market for virtual personal assistant (VPA) gadgets is emerging as a new field of conflict for international information technology businesses with the development of artificial intelligence technologies. Based on perceived value theory, this study creates a thorough research model to explain why potential users could choose to embrace and employ VPA devices. It examines the connection between qualities associated to a product's perceived utility, delight, and enjoyment (i.e., portability, automation, and visual attractiveness). Using data from 313 survey samples, partial least squares analysis is used to assess the research model and hypotheses. The findings demonstrate that usage intention is significantly influenced by perceived utility and enjoyment. The software and hardware-based utilitarian value, has the biggest effect on perceived usefulness.

2.2.Reference :

[1] Park, KeeHyun & Lim, SeungHyeon, (2012) "Construction of a Medication Reminder Synchronization System based on Data Synchronization", International Journal of Bio-Science and

Bio-Technology, Vol.4, No. 4, pp1-10.

[2] "Smartphone medication adherence apps: Potential benefits to patients and providers", available at:

<http://www.ncbi.nlm.nih.gov/pmc/articles/PMC3919626/>

[3] Slagle, J.M., Gordon, J.S., Harris, C.E., Davison, C.L., Culpepper, D.K., Scott P. and Johnson, K.B.,

(2011) "MyMediHealth – Designing a next generation system for child-centered medication

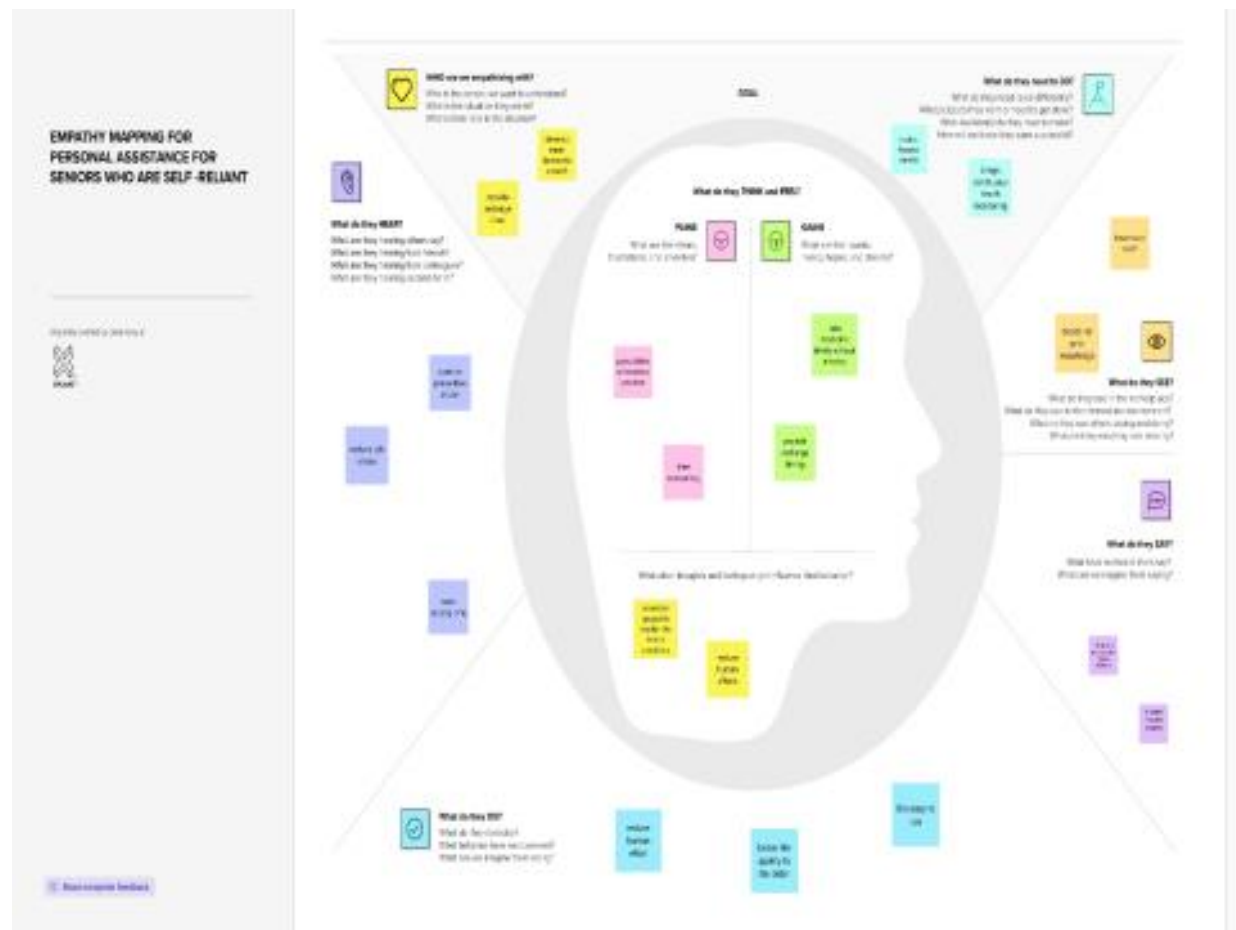
management", Journal of Biomedical Informatics, Vol. 43, No. 5, pp. 27-31.

2.3.Problem Statment Definition :

Some people find it difficult to learn new apps in this ever-expanding digital environment, and people nowadays tend to forget things more easily, such as taking their prescriptions. People need a way to remember to take their prescriptions without having to learn how to use sophisticated program

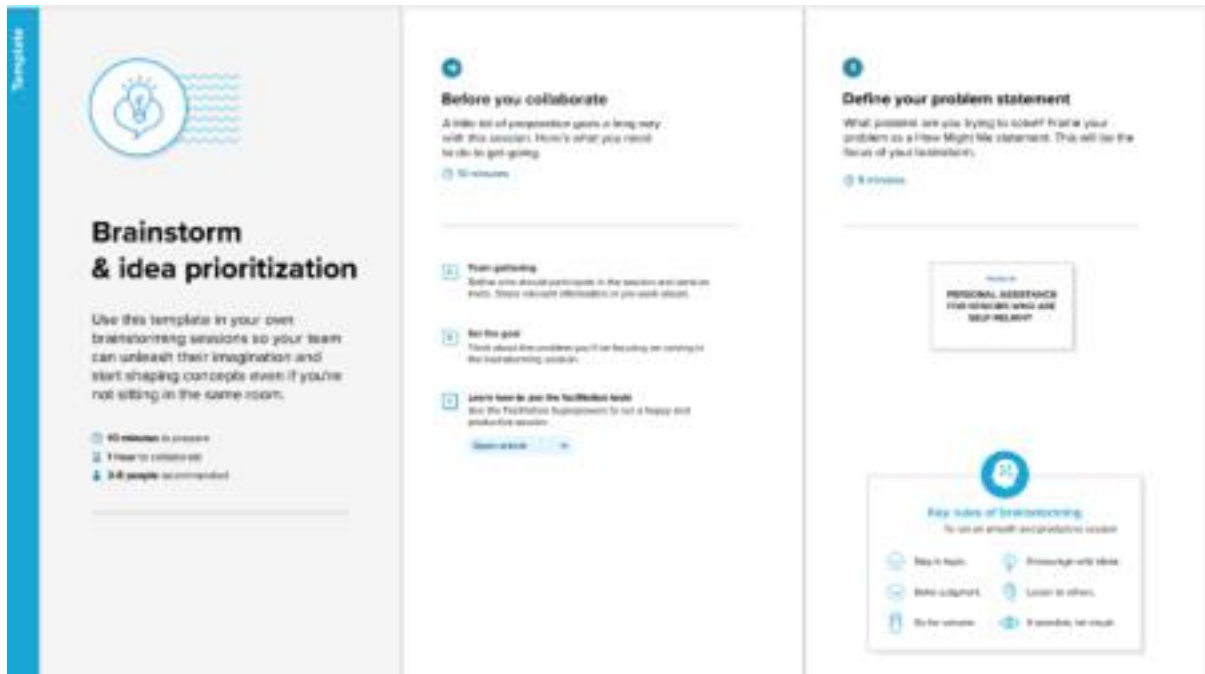
CHAPTER-3

3.1 Empathy Map Canvas

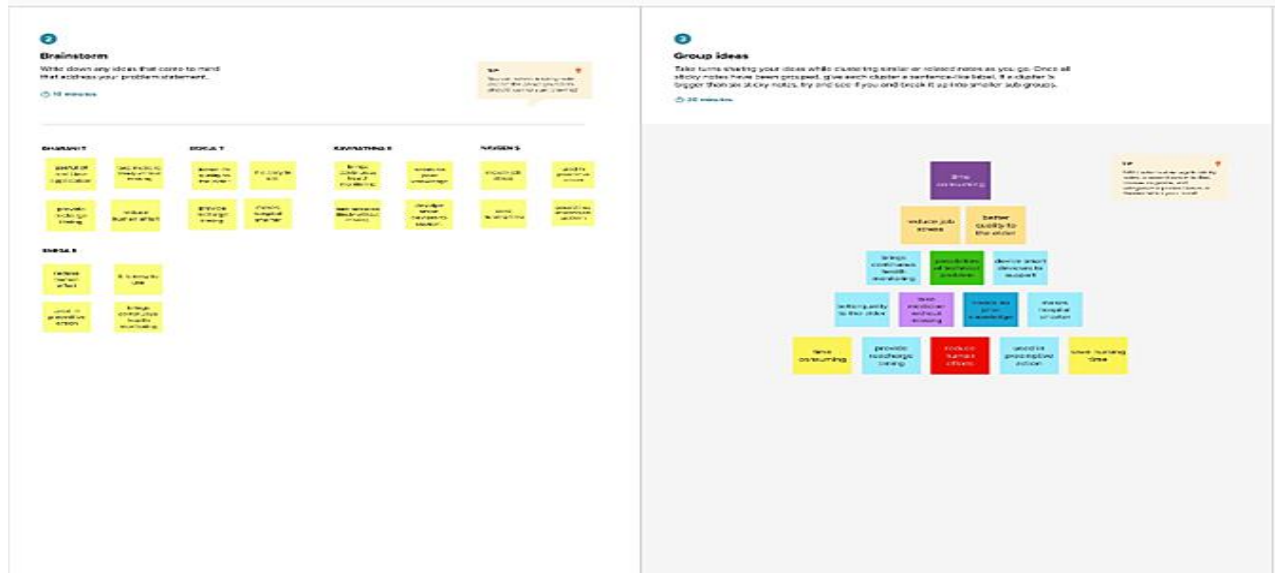


3.2 Ideation & Brainstorming :

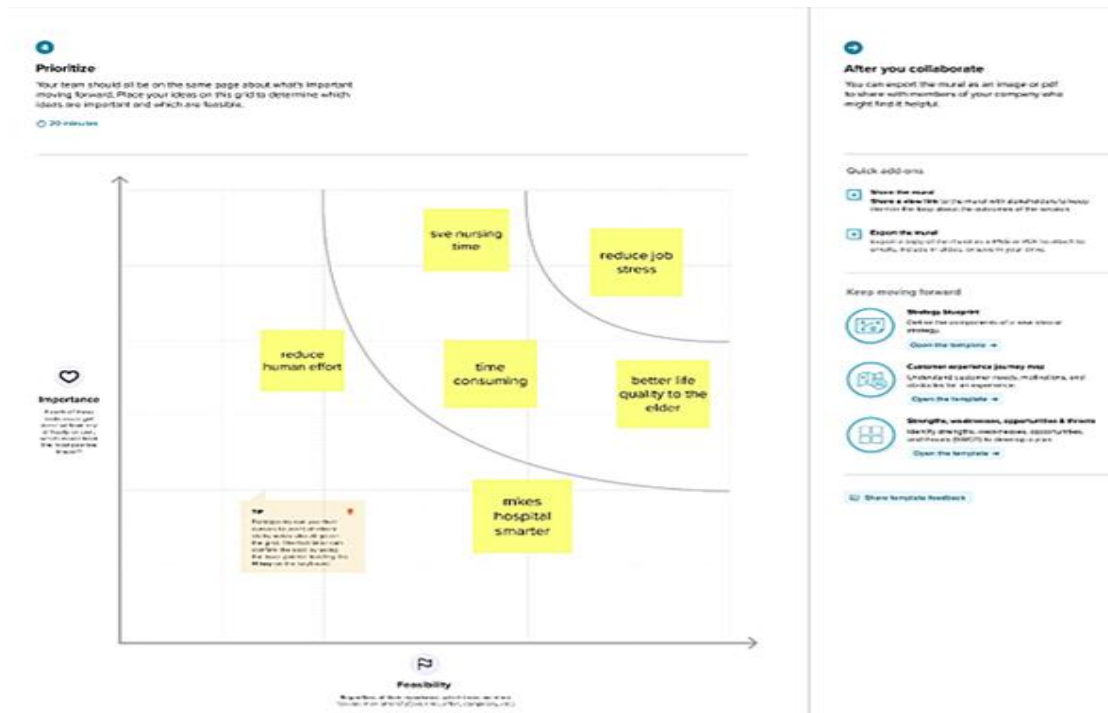
STEP 01: Team gathering, Collaboration and select the problem statement



02: Brainstorm, Idea Listening and Grouping :



03: Idea prioritization :



3.3. Proposed Solution :

Proposed Solution Template:

Project team shall fill the following information in proposed solution template.

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	Who needs to manage my medicines intuitively because she has been sick since she was a child.
2.	Idea / Solution description	Any device that reminds user to take medications.
3.	Novelty / Uniqueness	Get reminders via mobile app.
4.	Social Impact / Customer Satisfaction	Serve as a good way to stay on track and upload an appropriate schedule.
5.	Business Model (Revenue Model)	Ability of an generate medication reminders on different medications for more than one user, for example family members.

3.4. Problem Solution Fit :

<p>Define CS, fit into CC</p> <p>1. CUSTOMER SEGMENT(S) CS</p> <ul style="list-style-type: none"> Majority of patient forget to take medicine as it is prescribed Its difficult for task for the physician is to attempt to open the eyes of patients to take 	<p>6. CUSTOMER CONSTRAINTS CC</p> <ul style="list-style-type: none"> Pdf reports of your health and progress Doctors visit planner to keep track of appointments 	<p>5. AVAILABLE SOLUTIONS AS</p> <ul style="list-style-type: none"> The ability to track information such as your weight and glucose levels A journal to record your thoughts and symptoms <p>Explore AS, differentiate</p>
<p>Focus on J&P, tap into BE, understand RC</p> <p>2. JOBS-TO-BE-DONE / PROBLEMS BE</p> <ul style="list-style-type: none"> Daily health dairy to log your health history Points and rewards when remembering to 	<p>9. PROBLEM ROOT CAUSE RC</p> <ul style="list-style-type: none"> Used to analyze serious adverse event goal of RC is to identify any underlying problems in process 	<p>7. BEHAVIOUR BE</p> <ul style="list-style-type: none"> Personal medication records Drug information and possible interactions Deletes medications or keeps them archived for future reference <p>Focus on J&P, tap into BE, understand RC</p>

<p>3. TRIGGERS TR</p> <ul style="list-style-type: none"> Trigger helps to people to integrate healthy behaviour by using technology in a very simple way <p>4. EMOTIONS: BEFORE / AFTER EM</p> <ul style="list-style-type: none"> Physician can gauge the extent to which the patient is willing to discuss emotional distress Such relationships do occur in primary care Some physicians and patients to deal with mental disorders 	<p>10. YOUR SOLUTION</p> <ul style="list-style-type: none"> Build a morning or bedtime routine Set up timed reminder Wear a trendy reminder device Take meds with your meals 	<p>8. CHANNELS of BEHAVIOUR CH</p> <p>8.1 ONLINE</p> <ul style="list-style-type: none"> All the reminders notification were very soft & can be swiped easily Sometimes reminders are just unnecessarily complicated and tiresome <p>8.2 OFFLINE</p> <ul style="list-style-type: none"> make voice assistants remind you more making reminder more responsive to device
---	---	--

CHAPTER - 4

4.REQUIREMENT ANALYSIS :

4.1. Functional Requirement

Following are the functional requirements of the proposed solution.

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	Medicine reminder	<ol style="list-style-type: none">1. Generate the medicine reminder application2. Input to the system is the information entered by the patient which includes date, time, medicine name, doctor's name, etc...
FR-2	Take medicine on time	<ol style="list-style-type: none">1. The patients take their medicines as prescribed (Eg, twice daily)2. The output of the system focuses on "medication adherence "
FR-3	Patient login module	<ul style="list-style-type: none">• After login the patient will be able to the list of all registered doctors with their names, contacting information, phone numbers, hospital/clinic address.
FR-4	Set alarm module	<ol style="list-style-type: none">1. It helps in remindind about the medicines2. User can add details of his dosage schedules
FR-5	Get notification module	<ol style="list-style-type: none">1. Once the alarm is set then the user gets the notification2. The users can activate or deactivate this accordingly
FR-6	Health care module	<ul style="list-style-type: none">• In it patients can read different posts , article , new technology in medical sciences ,tips and other information

4.2.Non-Functional Requirements :

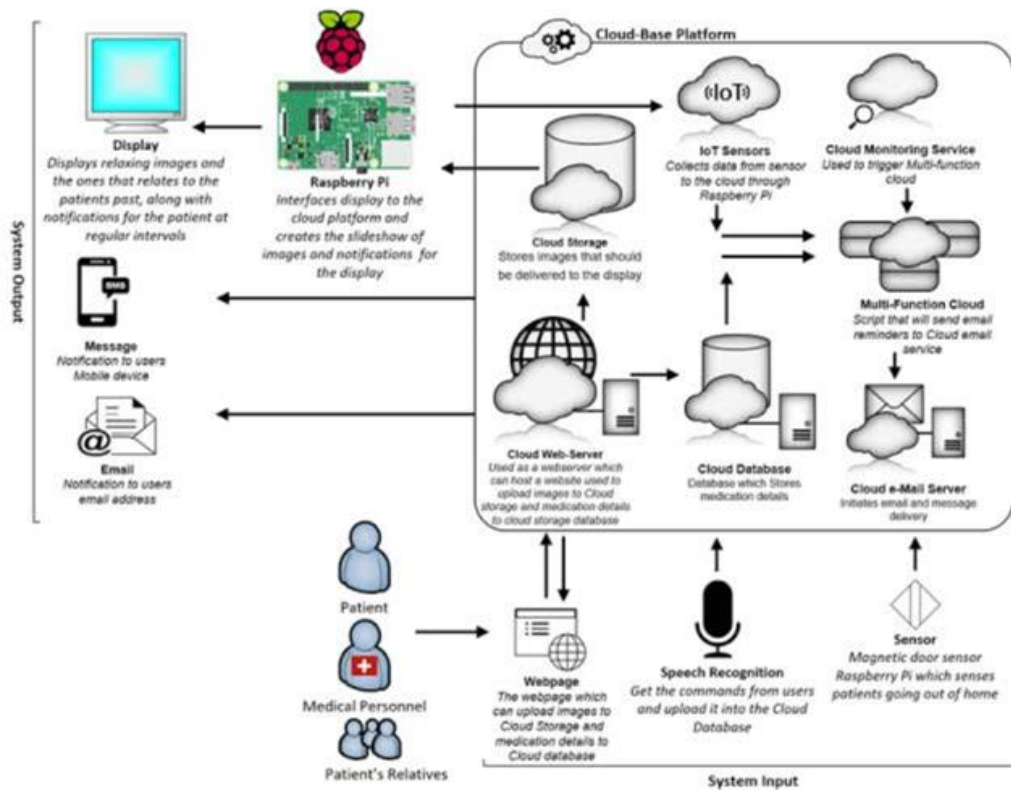
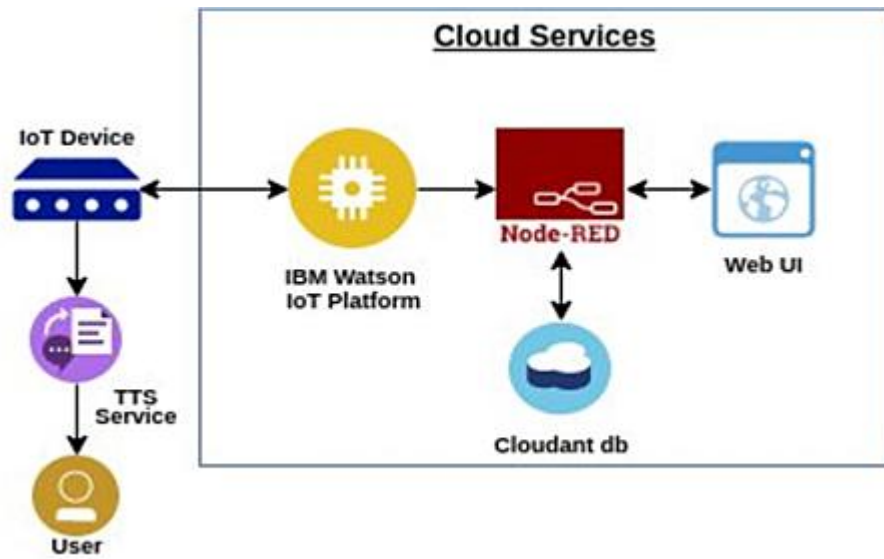
Following are the non-functional requirements of the proposed solution.

FR No.	Non-Functional Requirement	Description
NFR-1	Usability	The message medication reminder application is easy to use and accepted by patients and their designated med friend.
NFR-2	Security	Alarmsystem,UI,system,wireless devices,sensors,monitoring system.
NFR-3	Reliability	Medication tracking history Flexible scheduling Reminders with no connectivity
NFR-4	Performance	Digital health technology ,medication event monitoring system.
NFR-5	Availability	The app has these features the option of scheduling multiple reminders.
NFR-6	Scalability	It allows to create customizable reminder of when to take your medications.

CHAPTER - 5

5.PROJECT DESIGN :

5.1.Data Flow Diagrams:

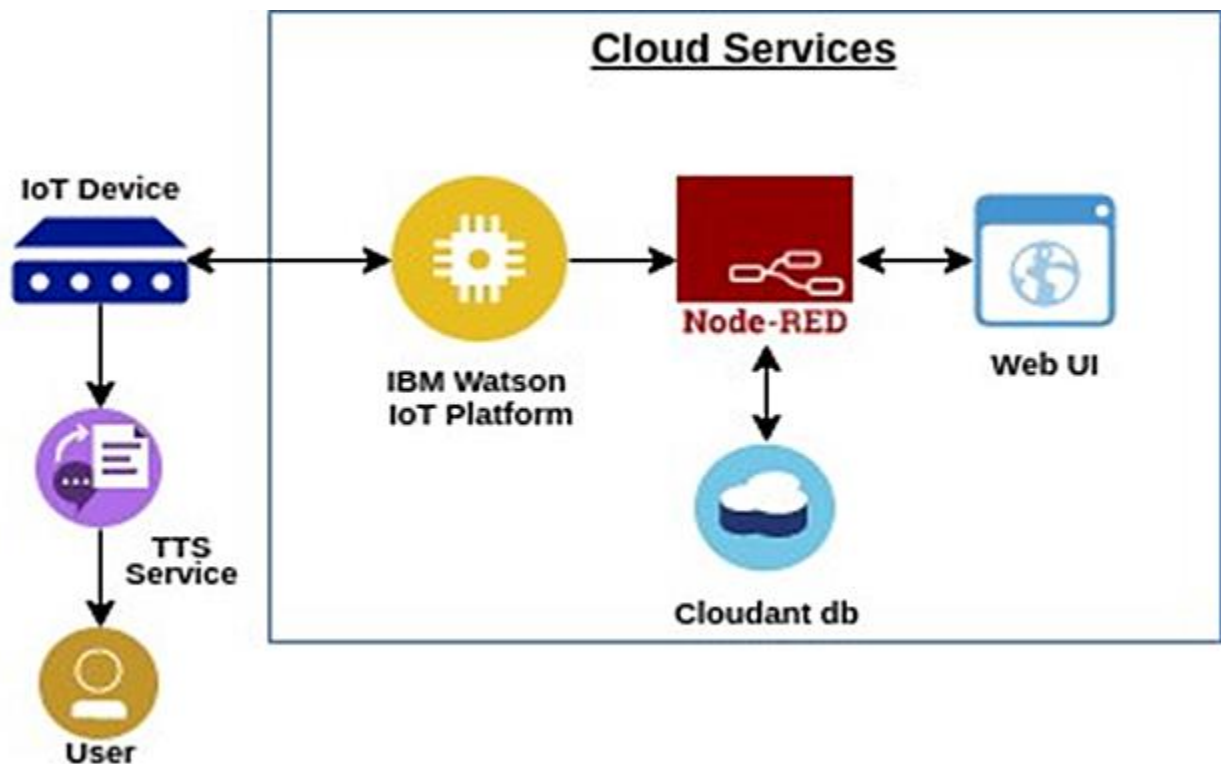


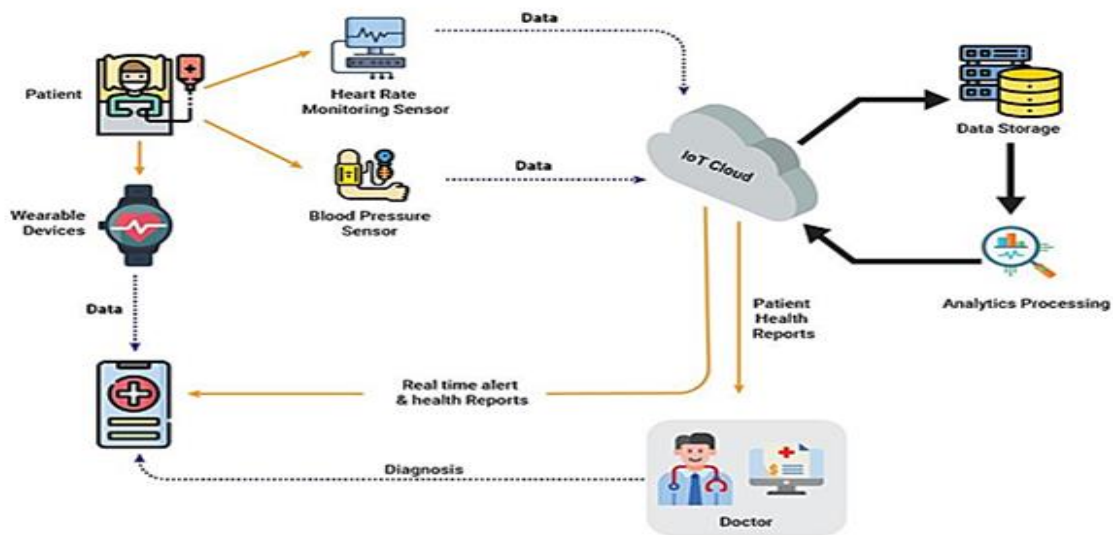
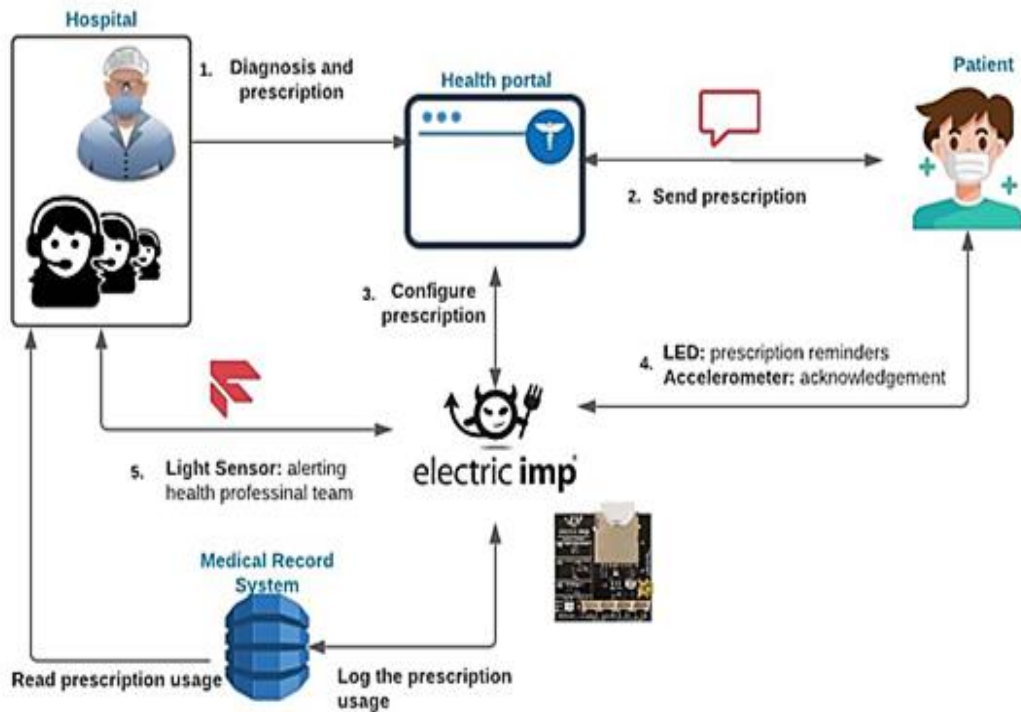
User Stories

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 Medicine Reminder	I can register & access the dashboard with medicine reminder 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	Sprint-1
	Dashboard	USN-6	To use the application and know the user name(patient name), Health details, medicine name,dose, Number of time per day and some etc.	I can access and know the information related to taking medicine	High	Sprint-1
Customer (Web user)	Registration	USN-7	As user can browse the website by entering my email, password, and confirming my password	I can access my account / dashboard	High	Sprint-1
		USN-8	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-9	As a user, I can register for the application through medicine reminder	I can register & access the dashboard with medicine reminder Login	High	Sprint-2
Customer Care Executive	To help the user's		To help the farmer to 24/7. To explain the question from the patient(user)	To help the user to critical situation	High	Sprint-1
Administrator	To maintain the server		The administrator have to maintain the server from the error and hackers	To maintain the server without busy	High	Sprint-1

5.2 Solution & Technical Architecture Diagram :



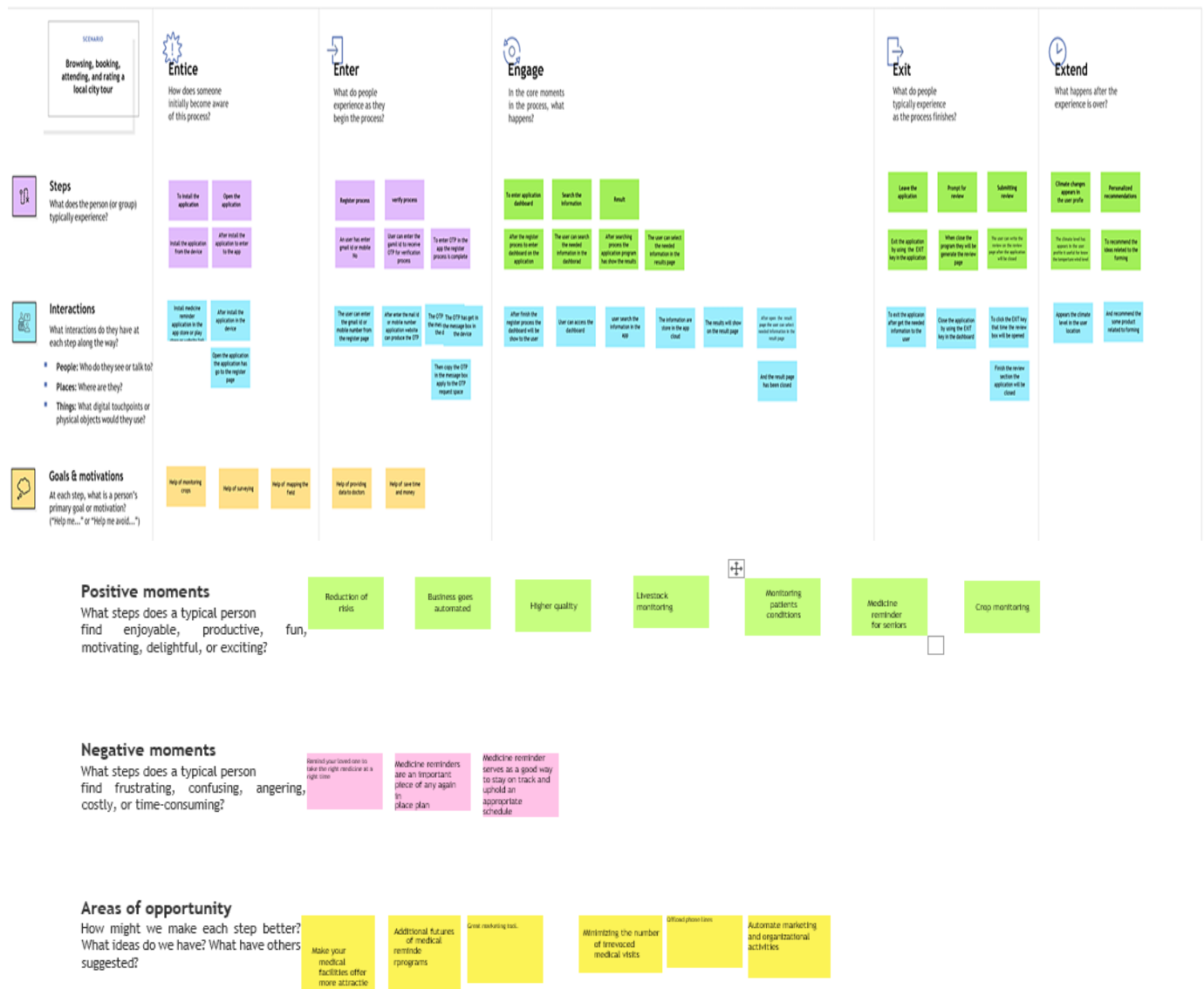


Document an existing experience

Narrow your focus to a specific scenario or process within an existing product or service. In the **Steps** row, document the step-by-step process someone typically experiences, then add detail to each of the other rows.

Document an existing experience

Narrow your focus to a specific scenario or process within an existing product or service. In the **Steps** row, document the step-by-step process someone typically experiences, then add detail to each of the other rows.



6. Project Planning & Scheduling :

6.1. Sprint Planning & Estimation :

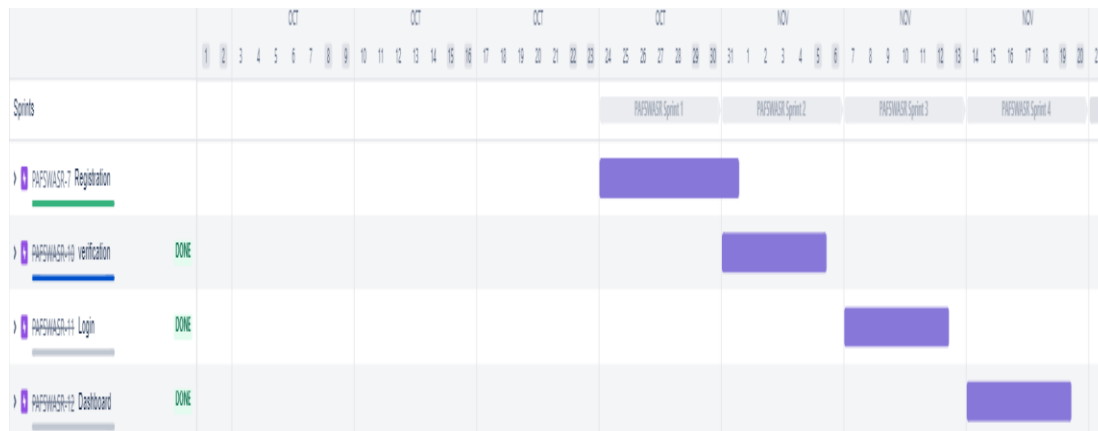
Use the below template to create product backlog and sprint schedule

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Registration	USN-1	As a user, I can register for the application by entering my email, password, and confirming my password.	2	High	Dharani
Sprint-1		USN-2	As a user, I will receive confirmation email once I have registered for the application	1	High	Gokul
Sprint-2	Verification	USN-3	As a user, I can register for the application through Medicine reminder	2	Low	Dharani
Sprint-1		USN-4	As a user, I can register for the application through Gmail	2	Medium	Naveen
Sprint-3	Login	USN-5	As a user, I can log into the application by entering email & password	1	High	Kavirathna
Sprint-4	Dashboard	USN-6	To use the application and know the patient name (user name, health details, medicine name, dose level, number of dose per day)	2	High	Snega

6.2. Sprint Delivery Schedule :

Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date (Planned)	Story Points Completed (as on Planned End Date)	Sprint Release Date (Actual)
Sprint-1	05	6 Days	24 Oct 2022	29 Oct 2022	05	29 Oct 2022
Sprint-2	02	6 Days	31 Oct 2022	05 Nov 2022	02	05 Nov 2022
Sprint-3	01	6 Days	07 Nov 2022	12 Nov 2022	01	12 Nov 2022
Sprint-4	02	6 Days	14 Nov 2022	19 Nov 2022	02	14 Nov 2022

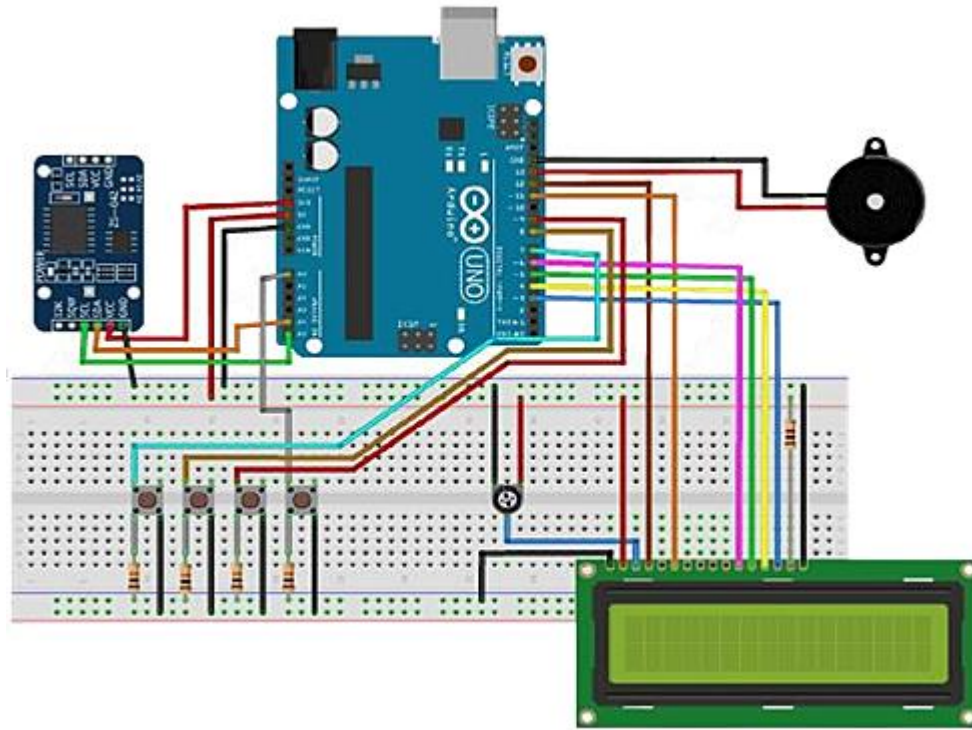
6.3. Reports From Jira :



CHAPTER - 7

7. CODING & SOLUTIONING (Explain the Features Added in The Project Along With Code)

7.1. Feature 1 :



CODE :

```
//Medicine Reminder using Arduino Uno
```

```
// Reminds to take medicine at 8am, 2pm, 8pm
```

```
/* The circuit:
```

LCD RS pin to digital pin 12

LCD Enable pin to digital pin 11

LCD D4 pin to digital pin 5

LCD D5 pin to digital pin 4

LCD D6 pin to digital pin 3

LCD D7 pin to digital pin 2

LCD R/W pin to ground

LCD VSS pin to ground

LCD VCC pin to 5V

10K resistor:

ends to +5V and ground

wiper to LCD VO pin (pin 3)*/

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

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

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

```
#include <EEPROM.h>
```

```
int pushVal = 0;
```

```
int val;
```

```
int val2; int addr = 0;
```

```
RTC_DS3231 rtc;
```

```
const int rs = 12, en = 11, d4 = 5, d5 = 4, d6 = 3, d7 =  
2;          // lcd pins
```

```
LiquidCrystal lcd(rs, en, d4, d5, d6, d7);
```

```
#define getWellsoon  
0
```

```
#define HELP_SCREEN 1
```

```
#define TIME_SCREEN 2
```

```
//bool pushPressed;          //flag to keep track of push button state
```

```
int pushpressed = 0;
```

```

const int ledPin
= LED_BUILTIN;          // buzzer and led pin

int ledState = LOW;
int Signal = 0;

int buzz = 13;

int push1state, push2state, push3state, stopinState =
0;  //

int push1Flag, push2Flag, Push3Flag = false;      // push button flags

int push1pin = 9;

int push2pin = 8;

int push3pin = 7;

int stopPin = A0;

int screens = 0;        // screen to show

int maxScreen = 2;      // screen count
bool isScreenChanged = true;

long previousMillis = 0;

long interval = 500;      // buzzing interval  unsigned long currentMillis;

long previousMillisLCD = 0;  // for LCD screen update

long intervalLCD = 2000;    // Screen cycling interval  unsigned long currentMillisLCD;

// Set Reminder Change Time

int buzz8amHH = 8;        // HH - hours    ##Set these for reminder time in 24hr

```

Format

```
int buzz8amMM = 00;    // MM - Minute
int buzz8amSS = 00;    // SS - Seconds int buzz2pmHH = 14;    // HH - hours
```

```
int buzz2pmMM = 00;    // MM - Minute
int buzz2pmSS = 00;    // SS - Seconds
```

```
int buzz8pmHH = 20;    // HH - hours
```

```
int buzz8pmMM = 00;    // MM - Minute
int buzz8pmSS = 00;    // SS - Seconds
```

```
int nowHr, nowMin, nowSec;    // to show current mm,hh,ss
```

```
// All messages
```

```
void gwsMessege(){    // print get well soon messege
```

```
    lcd.clear();
```

```
    lcd.setCursor(0, 0);
```

```
    lcd.print("Stay Healthy :)"); // Give some cheers
```

```
    lcd.setCursor(0, 1);
```

```
        lcd.print("Get Well Soon :)"); // wish
```

```
}
```

```
void helpScreen() {    // function to display
1st screen in LCD
```

```
    lcd.clear();
```

```
    lcd.setCursor(0, 0);
```

```
lcd.print("Press Buttons");
```

```
lcd.setCursor(0, 1);
```

```
lcd.print("for Reminder...!");
```

```
}
```

```
void timeScreen() {           // function to display  
Date and time in LCD screen
```

```
DateTime now = rtc.now();      // take rtc time and print in display
```

```
lcd.clear();
```

```
lcd.setCursor(0, 0);
```

```
lcd.print("Time:");
```

```
lcd.setCursor(6, 0);
```

```
    lcd.print(nowHr = now.hour(), DEC);
```

```
lcd.print(":");
```

```
lcd.print(nowMin = now.minute(), DEC);
```

```
lcd.print(":");
```

```
lcd.print(nowSec = now.second(), DEC);
```

```
lcd.setCursor(0, 1);
```

```
lcd.print("Date: ");
```

```
lcd.print(now.day(), DEC);
```



```

    lcd.print("/");

    lcd.print(now.month(), DEC);

    lcd.print("/");

    lcd.print(now.year(), DEC);
}

void setup() {

    Serial.begin(9600);           // start serial debugging

    if (! rtc.begin()) {         // check if rtc is connected
        Serial.println("Couldn't find RTC");

        while (1);
    }

    if (rtc.lostPower()) {
        Serial.println("RTC lost power, lets set the time!");
    }

    //  rtc.adjust(DateTime(F(__DATE__),
    F(__TIME__))); // uncomment this to set the current time and then comment in next upload when u
    set the time

    rtc.adjust(DateTime(2019, 1, 10, 7, 59, 30)); // manual time set

    lcd.begin(16, 2);

    lcd.clear();

    lcd.setCursor(0, 0);

```

```
lcd.print("Welcome  
To");           // print a messege at startup
```

```
lcd.setCursor(0, 1);
```

```
lcd.print("Circuit Digest");
```

```
delay(1000);
```

```
pinMode(push1pin,  
INPUT);           // define push button pins type
```

```
pinMode(push2pin, INPUT);
```

```
pinMode(push3pin, INPUT);
```

```
pinMode(stopPin, INPUT);
```

```
pinMode(ledPin, OUTPUT);
```

```
delay(200);
```

```
Serial.println(EEPROM.read(addr));
```

```
val2 = EEPROM.read(addr);           // read previosuly saved value of push button to start from where  
it was left previously
```

```
switch (val2) {
```

```
case 1:
```

```
Serial.println("Set for 1/day");
```

```
push1state = 1;
```

```
push2state = 0;
```

```
push3state = 0;
```

```
pushVal = 1;
```

```
break;
```

```
case 2:
```

```
Serial.println("Set for 2/day");
```

```
push1state = 0;
```

```
push2state = 1;
```

```
push3state = 0;  
pushVal = 2;
```

```
break;
```

```
case 3:
```

```
Serial.println("Set for 3/day");
```

```
push1state = 0;
```

```
push2state = 0;
```

```
push3state = 1;  
pushVal = 3;
```

```
break;
```

```
}
```

```
}
```

```
void loop() {
```

```
push1();  
/call to set once/day
```

```
push2();  
/call to set twice/day
```

```

    push3();
    /call to set thrice/day

    if (pushVal == 1)
    {
        // if push button 1 pressed then remind at 8am

        at8am();
        /function to start uzzing at 8am

    }

    else if (pushVal == 2)
    {
        // if push button 2 pressed then remind at 8am and 8pm

        at8am();

        at8pm();
        /function to start uzzing at 8mm

    }

    else if (pushVal == 3)
    {
        // if push button 3 pressed then remind at 8am and 8pm

        at8am();

        at2pm();
        //function to start uzzing at 8mm

        at8pm();

    }

    currentMillisLCD =
    millis();
    // start millis for
    LCD screen switching at defined interval of time

    push1state =
    digitalRead(push1pin);
    // start reading all push button pins

    push2state = digitalRead(push2pin);

```

```
push3state = digitalRead(push3pin);
```

```
stopinState = digitalRead(stopPin);
```

```
stopPins();  
// call to stop buzzing
```

```
changeScreen();           // screen cycle function
```

```
}
```

```
// push buttons
```

```
void push1() {           // function to set reminder once/day
```

```
if (push1 state == 1) {
```

```
    push1 state = 0;
```

```
    push2state = 0;
```

```
    push3state = 0;
```

```
//    pushPressed = true;
```

```
    EEPROM.write(addr, 1);
```

```
    Serial.print("Push1 Written : ");  
    Serial.println(EEPROM.read(addr)); // for debugging
```

```
    pushVal =  
1;           //save the state of push button-1
```

```
    lcd.clear();
```

```
lcd.setCursor(0, 0);

lcd.print("Reminder set ");

lcd.setCursor(0, 1);

lcd.print("for Once/day !");

delay(1200);

lcd.clear();
}

}

void push2() {           //function to set reminder twice/day

if (push2state == 1) {

push2state = 0;

push1state = 0;

push3state = 0;

//  pushPressed = true;

EEPROM.write(addr, 2);

Serial.print("Push2 Written : ");
Serial.println(EEPROM.read(addr));

pushVal = 2;

lcd.clear();

lcd.setCursor(0, 0);
```

```
lcd.print("Reminder set ");

lcd.setCursor(0, 1);

lcd.print("for Twice/day !");

delay(1200);

lcd.clear();
}

}

void push3() {           //function to set reminder thrice/day

if (push3state == 1) {

push3state = 0;

push1state = 0;

push2state = 0;

//  pushPressed = true;

EEPROM.write(addr, 3);

Serial.print("Push3 Written : ");
Serial.println(EEPROM.read(addr));

pushVal = 3;

lcd.clear();

lcd.setCursor(0, 0);

lcd.print("Reminder set ");

lcd.setCursor(0, 1);
```

```

    lcd.print("for Thrice/day !");

    delay(1200);

    lcd.clear();

}

}

void stopPins() {           //function to stop buzzing when user pushes stop push button

    if (stopinState == 1) {

//    stopinState = 0;

//    pushPressed = true;

        pushpressed = 1;

        lcd.clear();

        lcd.setCursor(0, 0);

        lcd.print("Take Medicine ");

        lcd.setCursor(0, 1);

        lcd.print("with Warm Water");

        delay(1200);

        lcd.clear();

    }

}

void startBuzz() {         // function to start buzzing when time reaches to defined interval

```



```

// if (pushPressed == false) {

if (pushpressed == 0) {

    Serial.println("pushpressed is false in blink");

    unsigned long currentMillis = millis();

    if (currentMillis - previousMillis >= interval) {

        previousMillis = currentMillis;    // save the last time you blinked the LED

        Serial.println("Start Buzzing");

        if (ledState == LOW) {            // if the LED is off turn it on and vice-versa:

            ledState = HIGH;

        } else {

            ledState = LOW;

        }

        digitalWrite(ledPin, ledState);

    }

}

else if (pushpressed == 1) {

    Serial.println("pushpressed is true");

    ledState = LOW;

    digitalWrite(ledPin, ledState);

}

}

```

```
void at8am() { // function to start buzzing at 8am

    DateTime now = rtc.now();

    if (int(now.hour()) >= buzz8amHH) {

        if (int(now.minute()) >= buzz8amMM) {

            if (int(now.second()) > buzz8amSS) {

                //////////////////////////////////////
                ////

                startBuzz();

                //////////////////////////////////////
                ////

            }

        }

    }

}

}

}

}

void at2pm() { // function to start buzzing at 2pm

    DateTime now = rtc.now();

    if (int(now.hour()) >= buzz2pmHH) {

        if (int(now.minute()) >= buzz2pmMM) {

            if (int(now.second()) > buzz2pmSS) {

                //////////////////////////////////////
                ///

                startBuzz();

                //////////////////////////////////////
                ///

            }

        }

    }

}

}
```

```

    }

    }

    }

}
void at8pm() {           // function to start buzzing at 8pm

    DateTime now = rtc.now();

    if (int(now.hour()) >= buzz8pmHH) {

        if (int(now.minute()) >= buzz8pmMM) {

            if (int(now.second()) > buzz8pmSS) {

                //////////////////////////////////////
                ////

                startBuzz();

                //////////////////////////////////////
                ////

            }

        }

    }

}

//Screen Cycling

void changeScreen() {    //function for
Screen Cycling

    // Start switching screen every defined intervalLCD

    if (currentMillisLCD - previousMillisLCD > intervalLCD)    // save the last time you changed the
display

```

```

{

previousMillisLCD = currentMillisLCD;

screens++;

if (screens > maxScreen) {

    screens = 0; // all screens over -> start from
1st

}

isScreenChanged = true;

}


// Start displaying current screen


if (isScreenChanged) // only update the screen if the screen is changed.

{

isScreenChanged = false; // reset for next iteration


switch (screens)

{

    case getWellsoon:

        gwsMessege();          // get well soon message

        break;


    case HELP_SCREEN:

        helpScreen();          // instruction screen

        break;


    case TIME_SCREEN:

```

```

timeScreen();          // to print date and time

break;

default:

//NOT SET.

break;

}

}

}

```

7.3. Database Schema :

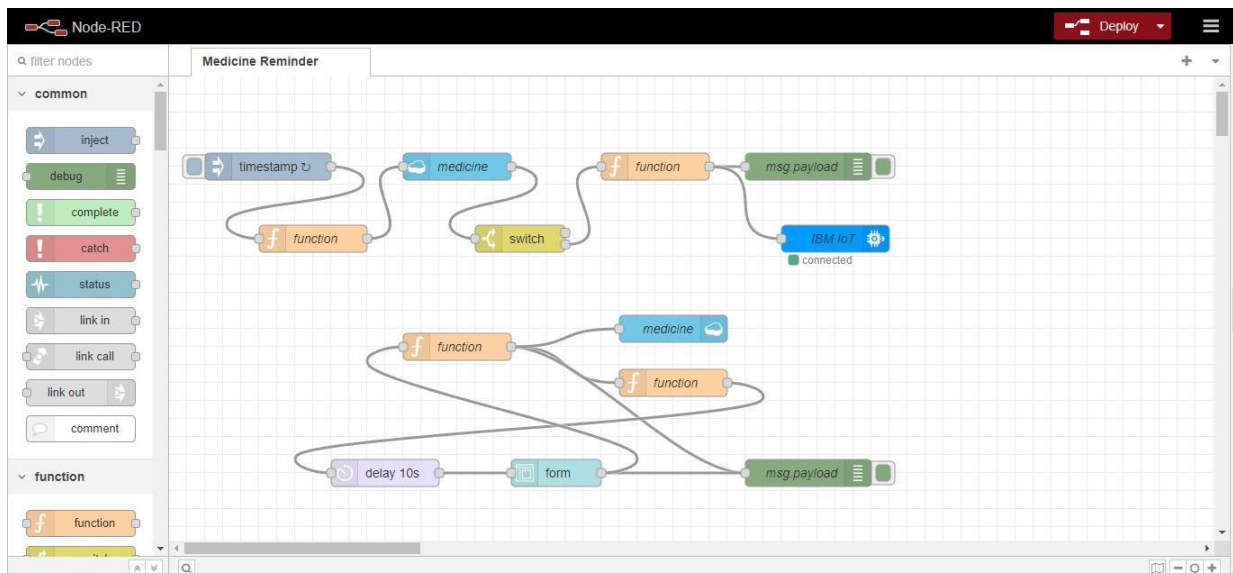
The screenshot shows a database management interface for a table named 'medicine_details'. The interface includes a sidebar with navigation options like 'All Documents', 'Query', 'Permissions', 'Changes', and 'Design Documents'. The main area displays a table with 4 columns: '_id', 'date', 'name', and 'time'. There are 6 rows of data. At the bottom, it shows 'Showing 4 of 5 columns' and 'Showing document 1 - 6'.

_id	date	name	time
2022-11-13 17:05	2022-11-13	PARACETAMOL	17:05
2022-11-13 17:24	2022-11-13	Amoxylin	17:24
2022-11-13 17:30	2022-11-13	Amoxylin	17:30
2022-11-13 17:32	2022-11-13	Paracetamol	17:32
2022-11-13 17:40	2022-11-13	Dolo 360	17:40
2022-11-13 18:55	2022-11-13	Amoxylin	18:55

CHAPTER - 8

8. TESTING

8.1. Test Cases



IBM Watson IoT Platform

dhara923019106005@gmail.com
ID: 125rx6

Browse Action Device Types Interfaces

Add Device

Search by Device ID

Device Simulator

Device ID	Status	Device Type	Class ID	Date Added
12345	Disconnected	new	Device	Nov 17, 2022 4:08 PM

Identity Device Information Recent Events State Logs

Device ID	12345
Device Type	new
Date Added	Nov 17, 2022 4:08 PM
Added By	dhara923019106005@gmail.com
Connection Status	Disconnected

Items per page 50 | 1-1 of 1 item

1 Simulation running

Browse

Action

Device Types

Interfaces

Add Device

Event	Value	Format	Last Received
event	{"data":21}	json	a few seconds ago
event	{"data":51}	json	a minute ago
event	{"data":82}	json	2 minutes ago
event	{"data":4}	json	3 minutes ago
event	{"data":4}	json	4 minutes ago

Items per page 50 | 1-1 of 1 item

1 of 1 page

1 Simulation running

iot_device

LoginPageAdd ScreenRemove ScreenPublish to Gallery

DesignerBlocks

Search Components...

User Interface

Layout

Media

Camcorder

Camera

ImagePicker

Player

Sound

SoundRecorder

SpeechRecognizer

TextToSpeech

Translator

VideoPlayer

Drawing and Animation

Maps

Charts

Sensors

Viewer

Display hidden components in Viewer

WELCOME TO THE MEDICINE REMINDER APP!!

Username:

Password:

LoginCancel

Components

LoginPage

VerticalArrangement1

HorizontalArrangement

Label3

HorizontalArrangement

HorizontalArrangement

Label1

TextBox1

HorizontalArrangement

Label2

PasswordTextBox1

HorizontalArrangement

VerticalArrangement

Button1

Button2

Web1

Media

Upload File...

Properties

LoginPage

AboutScreen

AlignHorizontal

Left: 1

AlignVertical

Top: 1

BackgroundColor

Black

BackgroundImage

None

BigDefaultText

CloseScreenAnimation

Default

HighContrast

OpenScreenAnimation

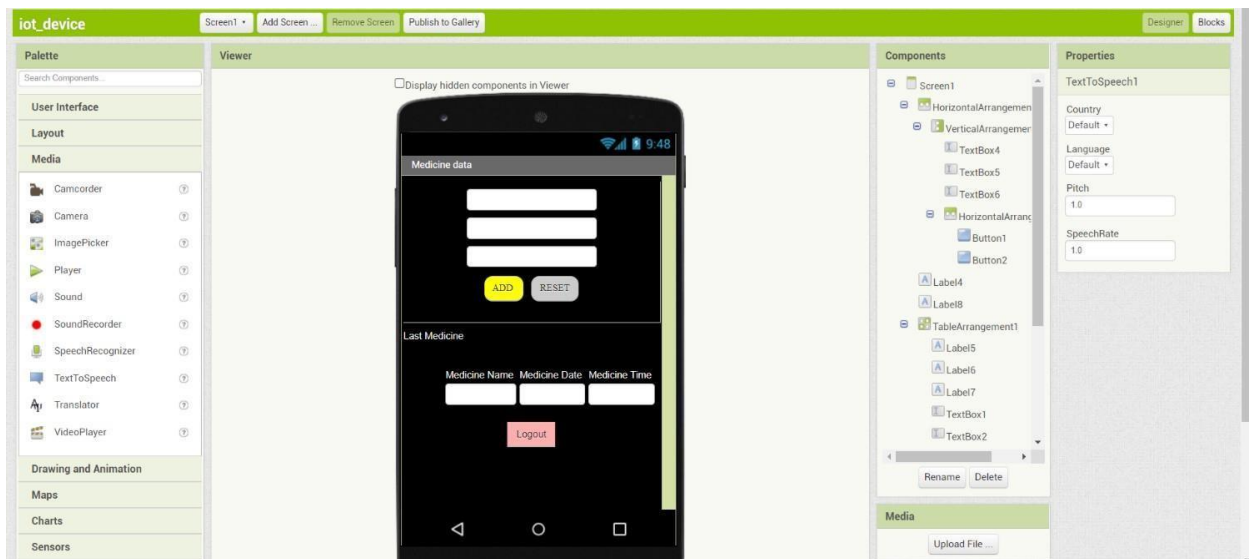
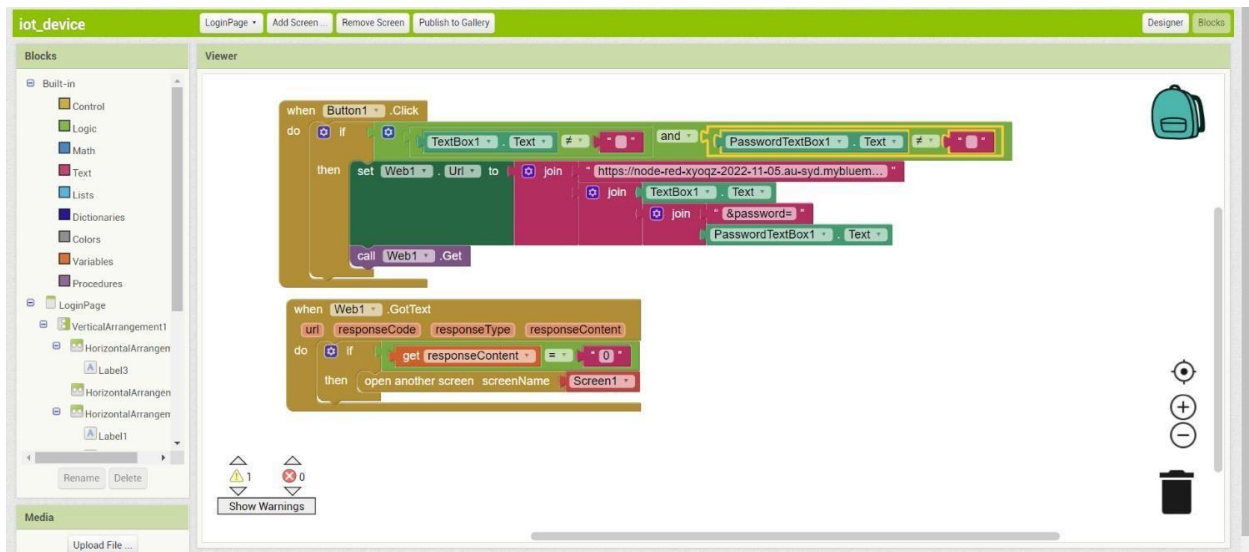
SlideHorizontal

ScreenOrientation

Portrait

Scrollable

ShowStatusBar





CHAPTER - 9

9. RESULTS :

9.1. Performance

WELCOME TO THE MEDICINE REMINDER APP!

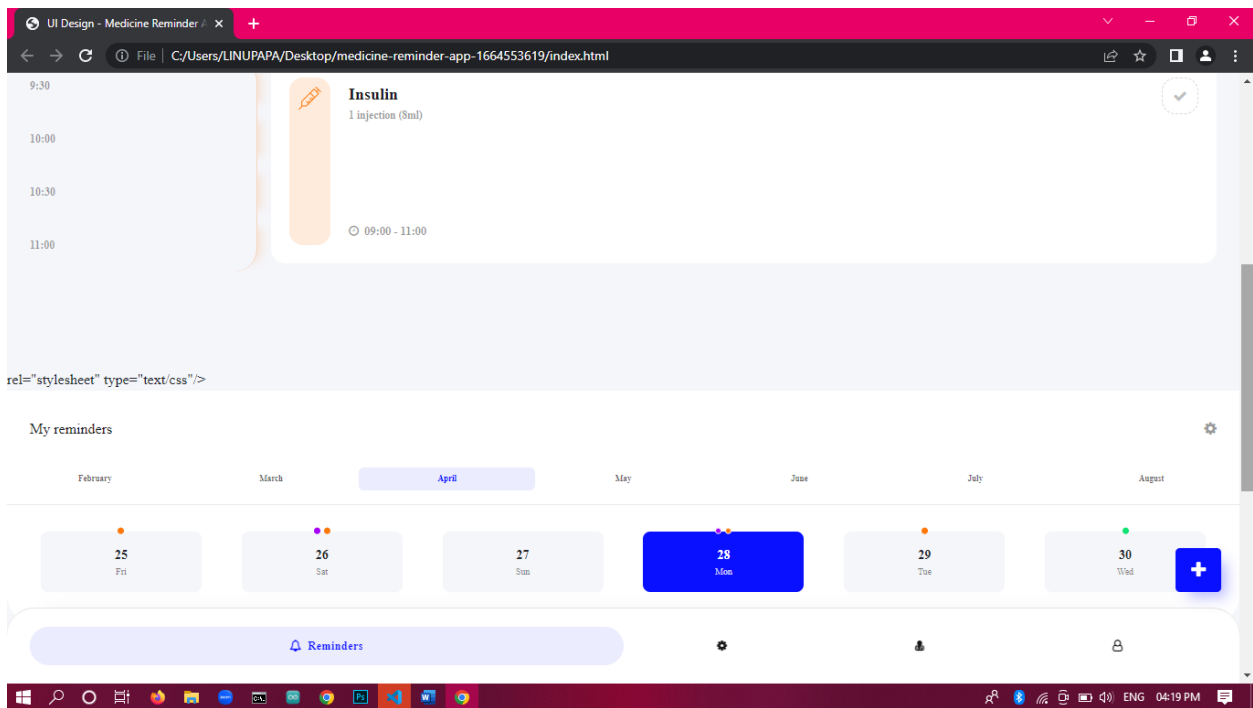
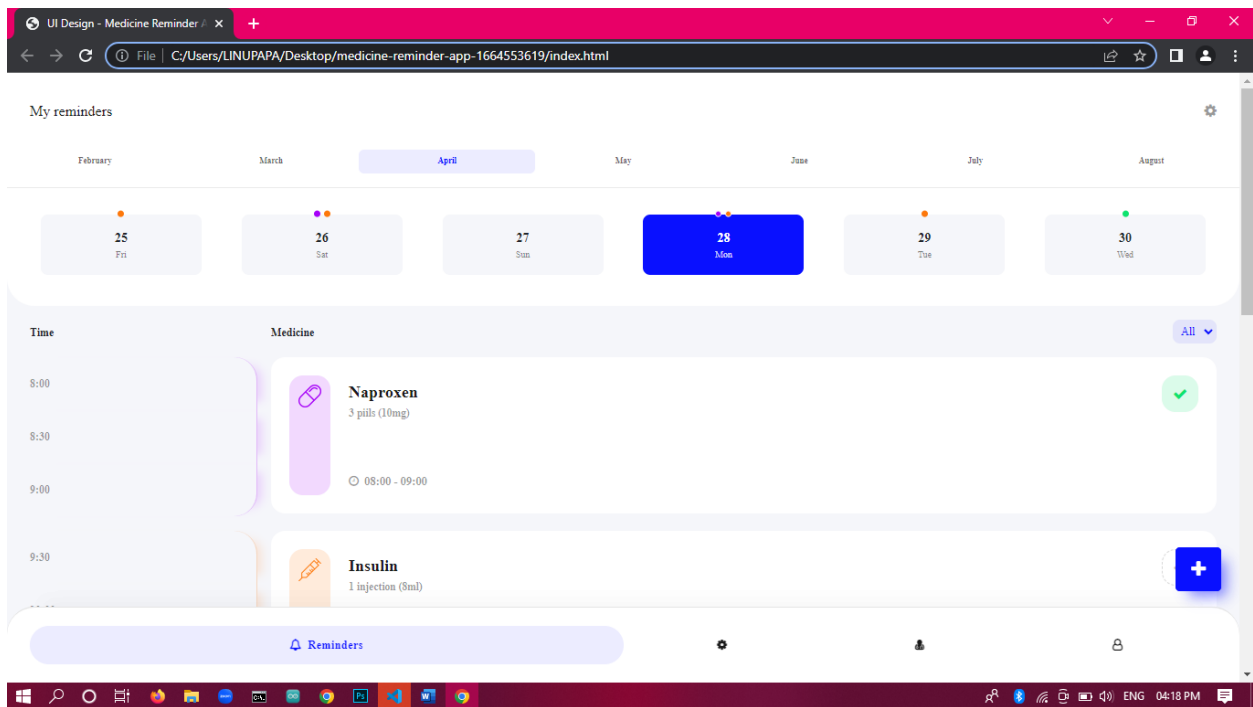
Username:

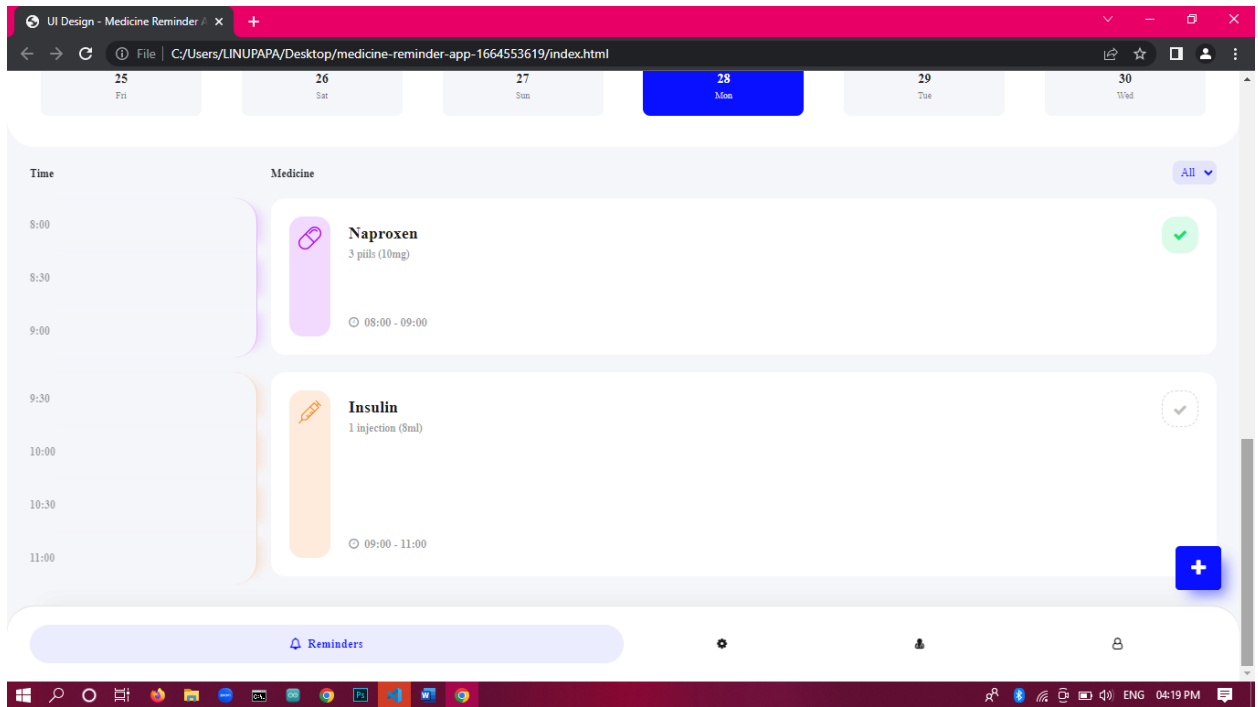
Password:

Medicine data

Last Medicine

Medicine Name	Medicine Date	Medicine Time
<input type="text" value="citrizen"/>	<input type="text" value="2022-11-17"/>	<input type="text" value="23:45"/>





CHAPTER - 10

ADVANTAGES :

1. Medication mix-ups are extremely dangerous. A medication reminder and organizer can help to prevent these life-threatening mistakes. They remind your loved one to take the right medication at the right time.
2. The medication reminder and organizer locks until it is time for your loved one to take a dose. When it is time, the reminder will beep and the compartment will open so your senior can access their pills.
3. By giving your loved one a medication reminder, you are also giving them greater health and independence. Help them age in place with the right tools to help them stay safe and healthy.
4. There are a variety of reminder alarms on the market today. These can range from a watch that the patient or caregiver wears with alarms that sound at various times throughout the day to computer software that can be programmed to let people know when it is time to take the medication.
5. Reduce No-shows. Results vary, but studies show that a simple appointment reminder can reduce no-shows drastically.
6. Keep Care on Track.
7. Increase Customer Satisfaction.
8. Patient Training.
9. Office Efficiency.
10. Staff Workflow.

DISADVANTAGES :

1. Does not encourage cancellation or rescheduling in patients who cannot attend or who no longer wish to attend. Some patients reported that they never received a SMS reminder.
2. It is possible that mobile phone numbers were entered incorrectly on patient records, or that those patients changed their phone numbers during the study.
3. Patients may not receive the SMS reminders due to incorrect data entry. Older patients were considerably less likely to own a mobile phone, making them harder to access using reminder technology.
4. costs associated with running an IT system.
5. the need for training in the operation of the new programme.
6. possibility of technical problems.

CHAPTER - 11

10 : CONCLUSION :

Nowadays, using smart phones and mobile applications are increased dramatically, so developing mobile applications in health services (especially self-care) can create the desired effect in the community. Although there are various medication reminder mobile applications, a native mobile application is essential that is developed on the basis of the specialists' ideas in

this field. In addition to remind the medication administration time and dose, “Seeb” reports the analysis of the patient medication administration, as well as displaying suitable picture of the medication and its administration method when reminded of medication use. Existence of these functions in the medication reminder mobile application prevents medication errors by patients and increases medication adherence.

Undoubtedly, “Seeb” can play an important role in patient health improvement with the suitable reminder of the medication administration by user friendly interfaces, data processing, correct calculation of formulas and appropriate responds, the display of the medication pictures and descriptions. Therefore, we suggest that health care providers increase patients’ awareness and introduce them medication reminder mobile applications to promote these applications utilization and to improve medication adherence as well as decreasing medication errors.

CHAPTER - 12

FUTURE SCOPE :



Therefore, we suggest that health care providers increase patients’ awareness and introduce them medication reminder mobile applications to promote these applications utilization and to improve medication adherence as well as decreasing medication errors.

CHAPTER - 13

APPENDIX :

GitHub Link :

<https://github.com/IBM-EPBL/IBM-Project-44682-1660726203>

Project Demo Link :

https://www.youtube.com/watch?v=_ETdYrAQkqk

work space link :

<https://workdrive.zohoexternal.com/writer/open/rl45i5f7bd434f3cf419eb7e453e032e047ef?authId=%7B%22linkId%22%3A%225k2wApabNUa-LYmIU%22%7D>