SMARTINTERNZ EXTERNSHIP PROJECT

PROJECT REPORT

SLEEP TRACKER APP

PROJECT TEAM :

1) ROSHAN VEERVANI 21BAI10057
2) MANCHIKANTI LAXMITEJ 21BAI10419

1. INTRODUCTION :

1.1 Overview

The Application is used to keep users aware about their Sleep and Health status, maintain a Good Lifestyle in Day-to-Day Life.

1.2 Purpose

Health Medicare App is a mobile application designed to help users track their sleep patterns, gain insights into their sleep quality, and improve their overall sleep hygiene. It leverages the Android Sleep API (available on devices running Android 10 or later) to provide a non-intrusive and battery-efficient sleep tracking experience.

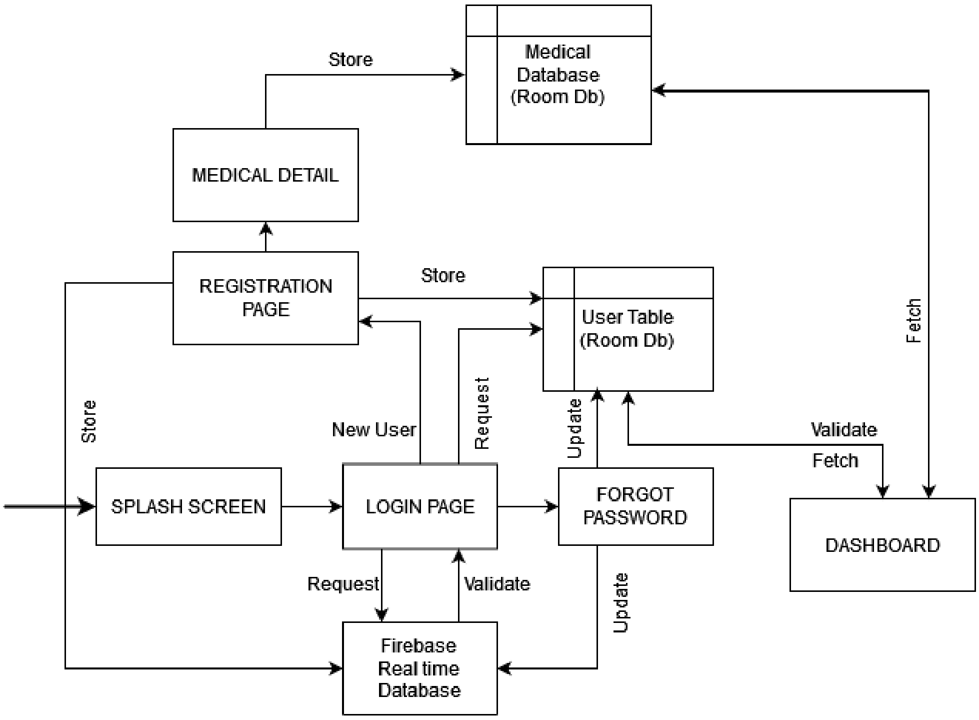
1.3 Existing problem

Many people struggle with inadequate or poor-quality sleep, leading to various health concerns like fatigue, mood swings, decreased productivity, and increased risk of chronic diseases. Existing sleep tracking apps often rely on microphone or motion sensors, which can be inaccurate and drain battery life.

1.4 Proposed solution

Health Medicare App addresses these issues by utilizing the Android Sleep API, which provides sleep data based on device usage patterns, ambient light levels, and motion (if permission is granted). This approach offers a more reliable and battery-friendly tracking method.

1.5 Block diagram (Database Control Flow)



1.6 Hardware / Software designing

Hardware Requirement:

1) A PC / Laptop with more than 8Gb RAM.

Software Requirement:

1) Android Studio Flamingo.
2) Room DB.
3) Google Firebase Realtime Database.
4) Google Firebase Cloud Messaging.
5) A Virtual Android Emulator.

2. EXPERIMENTAL INVESTIGATIONS :

1) User Registration / Forgot Password / Login: We have to analyse which Database to use, so that it can store user's information properly. Also, it has to be stored in the Cloud as well as User's Device. So, we used both **Room Db and Firebase Realtime Database** to store Login Credentials. User Login Details in separate Table and Medical Details in another Table.

2) Dashboard: To display user's information on Successful Login in the Dashboard, we used **Room Db**. We give the User Email Address, to fetch all the user Related Details from the Database and reflect it back.

3) Calorie Manage Page : Collecting the Details from the User. Calculating all the **Calorie Data in the Background**, and displaying back to the user.

4) Health Article : We used **(An Open-source API)** to Load the Health Articles. We did research on finding the Best API to integrate within the App with proper architecture.

5) Sleep Tracker : We did some analysis on how to capture **current Timestamp in a variable**, convert them into Date, find difference, and display back to the User.

6) Map Search : We did some research on how to integrate **Google Maps** into an Android App. Also, how to display the Nearby hospitals at a given Location.

7) Emergency Dial : We did some research on finding all the important HealthCare Toll free numbers in the Country, how to integrate **Messaging and Call option** directly. We used **Intent** to successfully transfer the Mobile number.

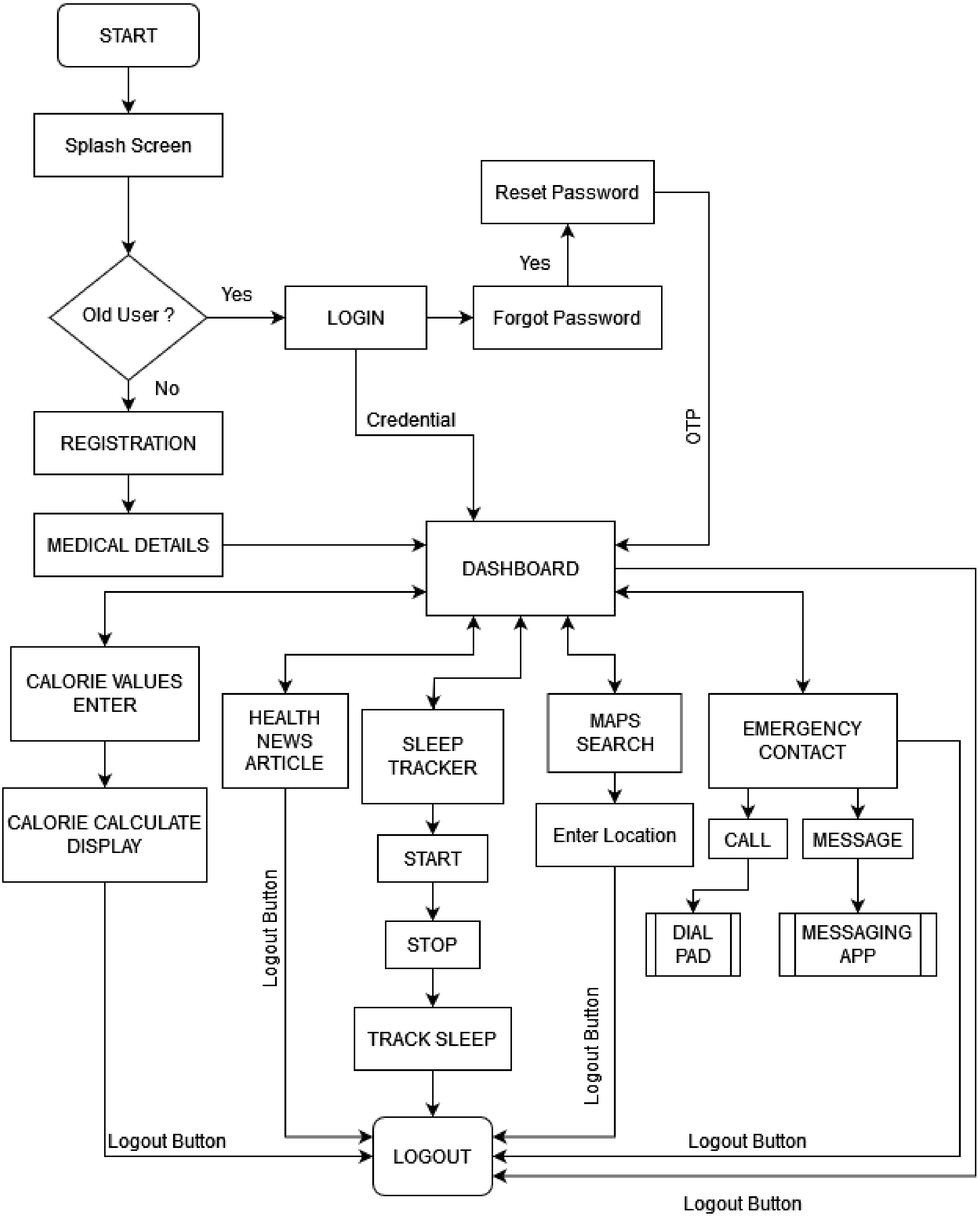
8) OTP (Forgot Password) : We used **SMS-Manager** to directly send OTP to the user's Mobile Number.

9) Daily Reminder Notifications : We are using **Google Firebase Cloud Messaging** to send Automatic Scheduled Notifications to all the users.

10) Permissions : We ask user **SMS, Notification** Allow Permission.

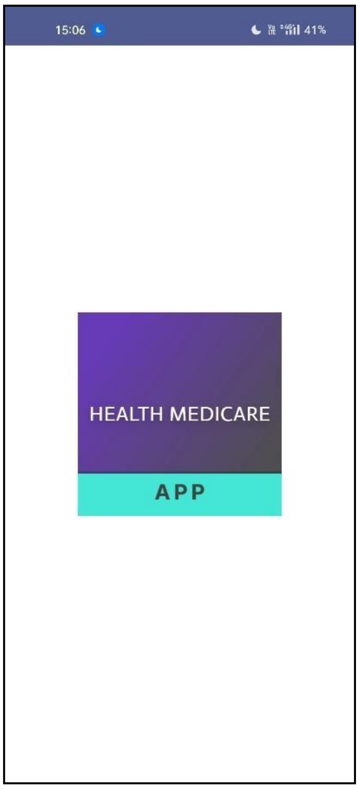
3. FLOWCHART :

Diagram showing the Control flow of the Project

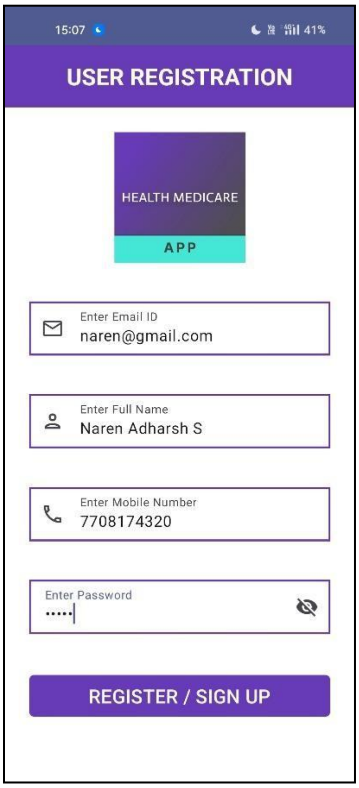


4. RESULT :

Splash screen



Registration



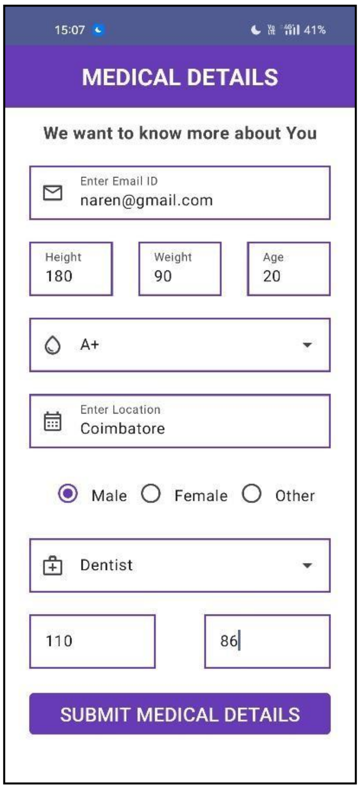
Enter Email ID
naren@gmail.com

Enter Full Name
Naren Adharsh S

Enter Mobile Number
7708174320

Enter Password
•••••

Medical details



We want to know more about You

Enter Email ID
naren@gmail.com

Height
180

Weight
90

Age
20

A+

Enter Location
Coimbatore

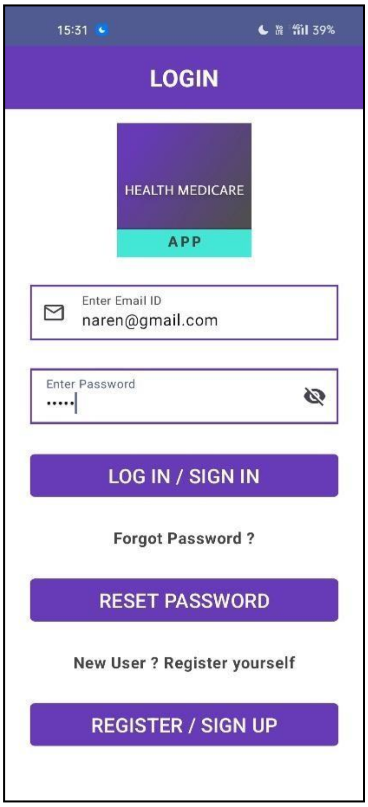
● Male ○ Female ○ Other

Dentist

110

86

Login



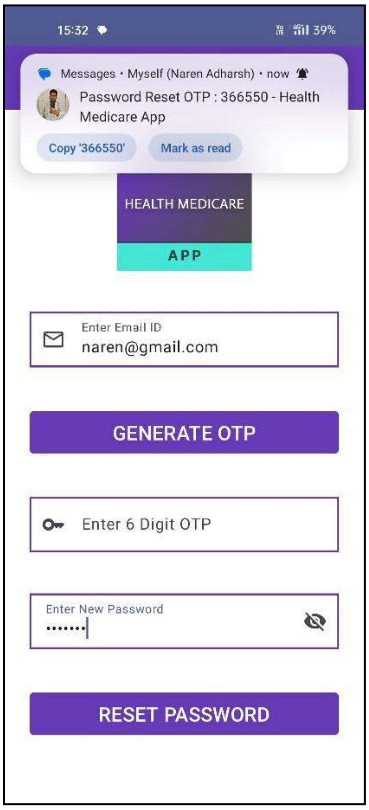
Enter Email ID
naren@gmail.com

Enter Password
•••••

Forgot Password ?

New User ? Register yourself

Forget Password



Messages • Myself (Naren Adharsh) • now

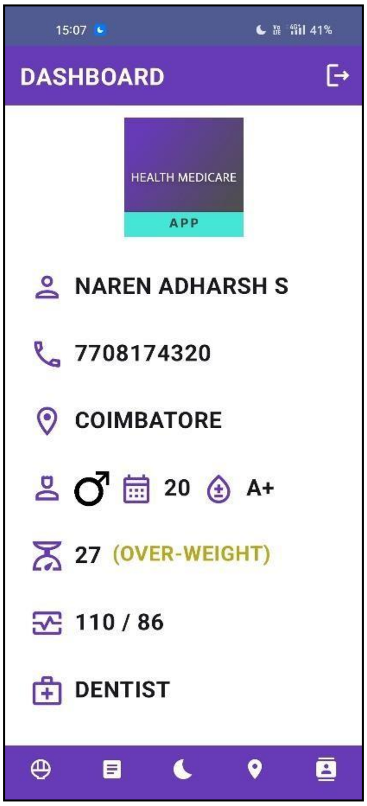
Password Reset OTP : 366550 - Health Medicare App

Enter Email ID
naren@gmail.com

Enter 6 Digit OTP

Enter New Password
•••••••

Dashboard



NAREN ADHARSH S

7708174320

COIMBATORE

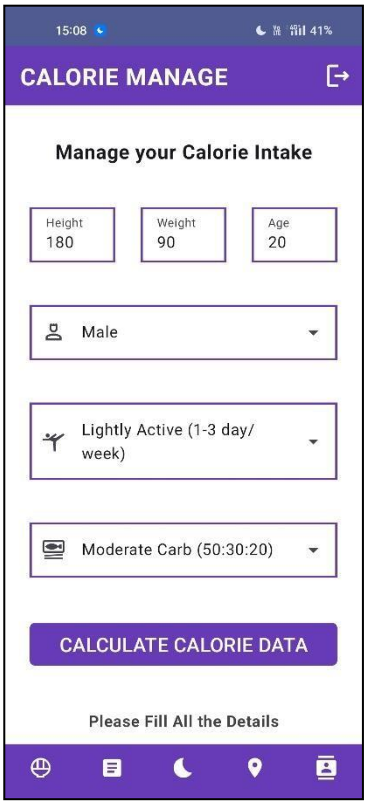
♂ 20 A+

27 (OVER-WEIGHT)

110 / 86

DENTIST

Calorie Manage



15:08 41%

CALORIE MANAGE

Manage your Calorie Intake

Height	Weight	Age
180	90	20

Male

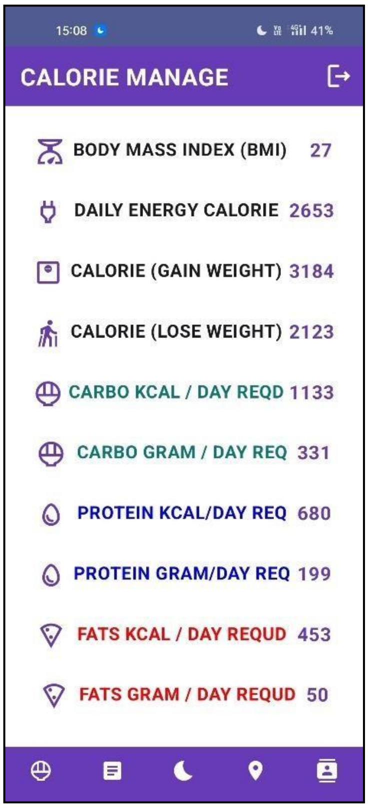
Lightly Active (1-3 day/ week)

Moderate Carb (50:30:20)

CALCULATE CALORIE DATA

Please Fill All the Details

Calorie Manage

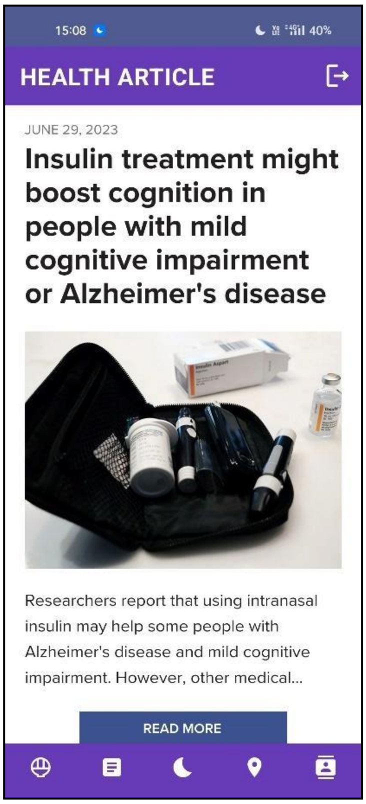


15:08 41%

CALORIE MANAGE

BODY MASS INDEX (BMI)	27
DAILY ENERGY CALORIE	2653
CALORIE (GAIN WEIGHT)	3184
CALORIE (LOSE WEIGHT)	2123
CARBO KCAL / DAY REQD	1133
CARBO GRAM / DAY REQ	331
PROTEIN KCAL/DAY REQ	680
PROTEIN GRAM/DAY REQ	199
FATS KCAL / DAY REQUD	453
FATS GRAM / DAY REQUD	50

Health Article



15:08 40%

HEALTH ARTICLE

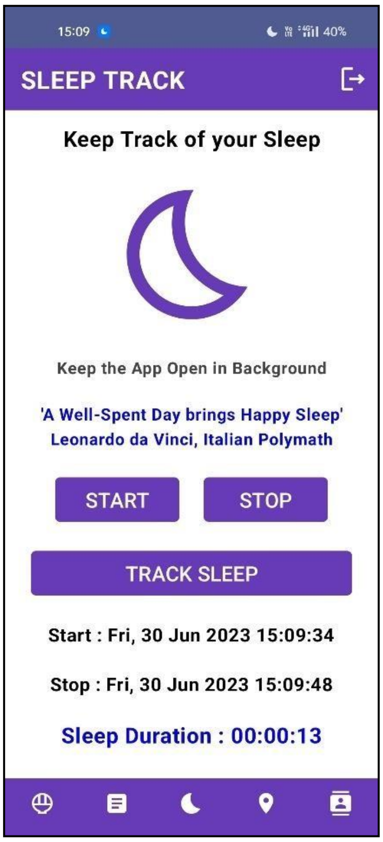
JUNE 29, 2023

Insulin treatment might boost cognition in people with mild cognitive impairment or Alzheimer's disease

Researchers report that using intranasal insulin may help some people with Alzheimer's disease and mild cognitive impairment. However, other medical...

READ MORE

Sleep Track



15:09 40%

SLEEP TRACK

Keep Track of your Sleep

Keep the App Open in Background

'A Well-Spent Day brings Happy Sleep'
Leonardo da Vinci, Italian Polymath

START STOP

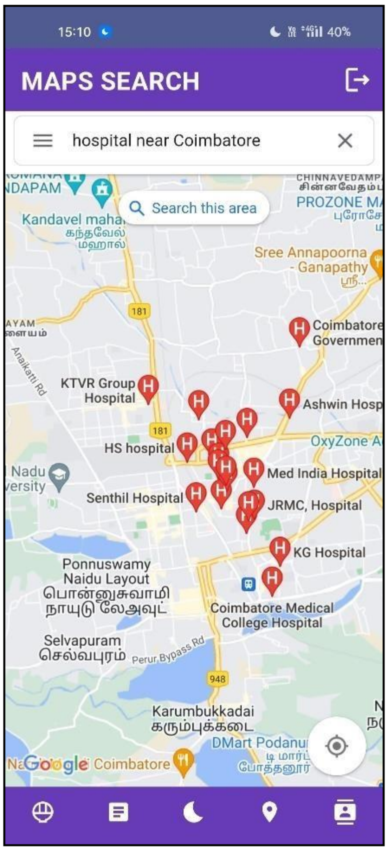
TRACK SLEEP

Start : Fri, 30 Jun 2023 15:09:34

Stop : Fri, 30 Jun 2023 15:09:48

Sleep Duration : 00:00:13

Maps Search

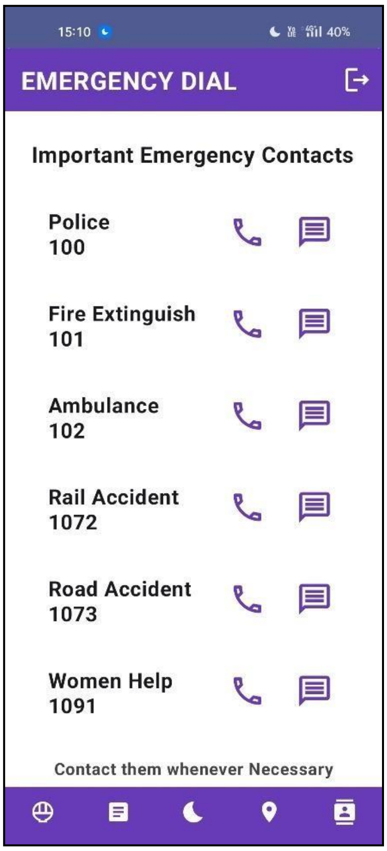


15:10 40%

MAPS SEARCH

hospital near Coimbatore

Emergency Dial



15:10 40%

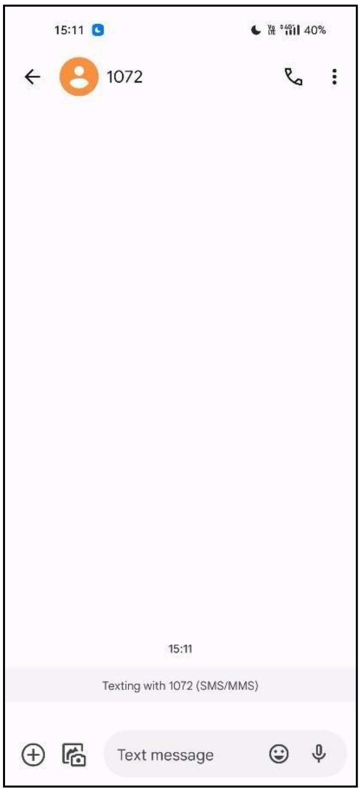
EMERGENCY DIAL

Important Emergency Contacts

Police	100
Fire Extinguish	101
Ambulance	102
Rail Accident	1072
Road Accident	1073
Women Help	1091

Contact them whenever Necessary

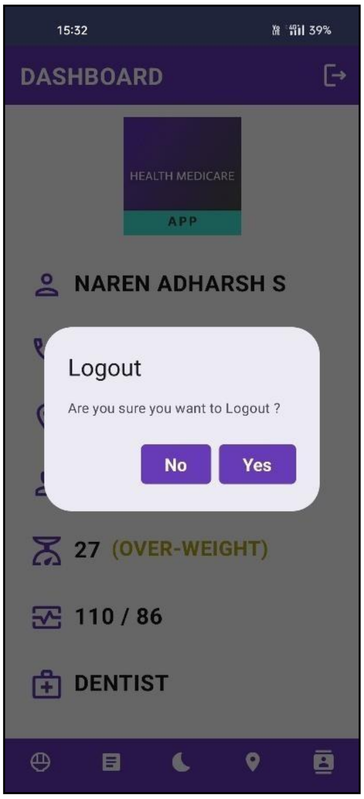
Messaging



15:11

Texting with 1072 (SMS/MMS)

Log out



NAREN ADHARSH S

Logout

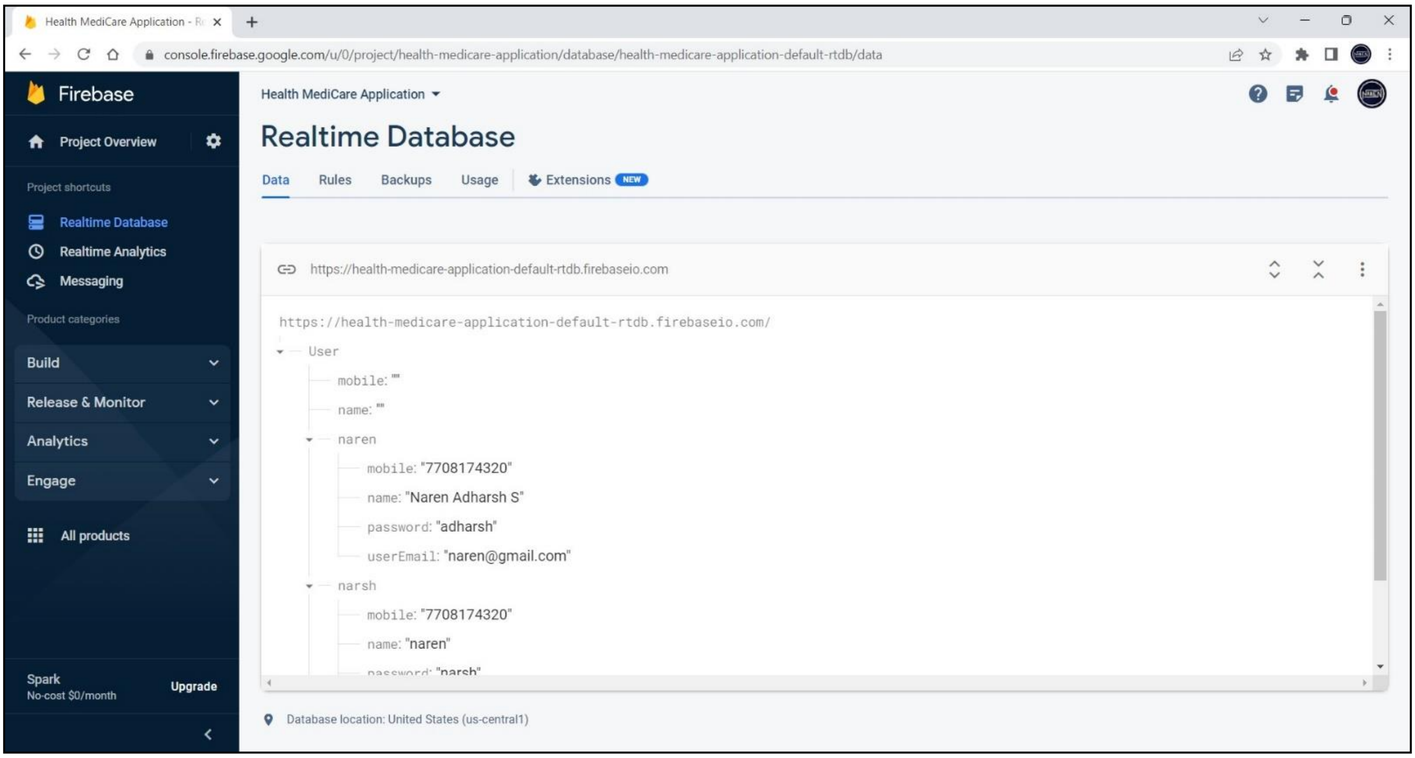
Are you sure you want to Logout ?

27 (OVER-WEIGHT)

110 / 86

DENTIST

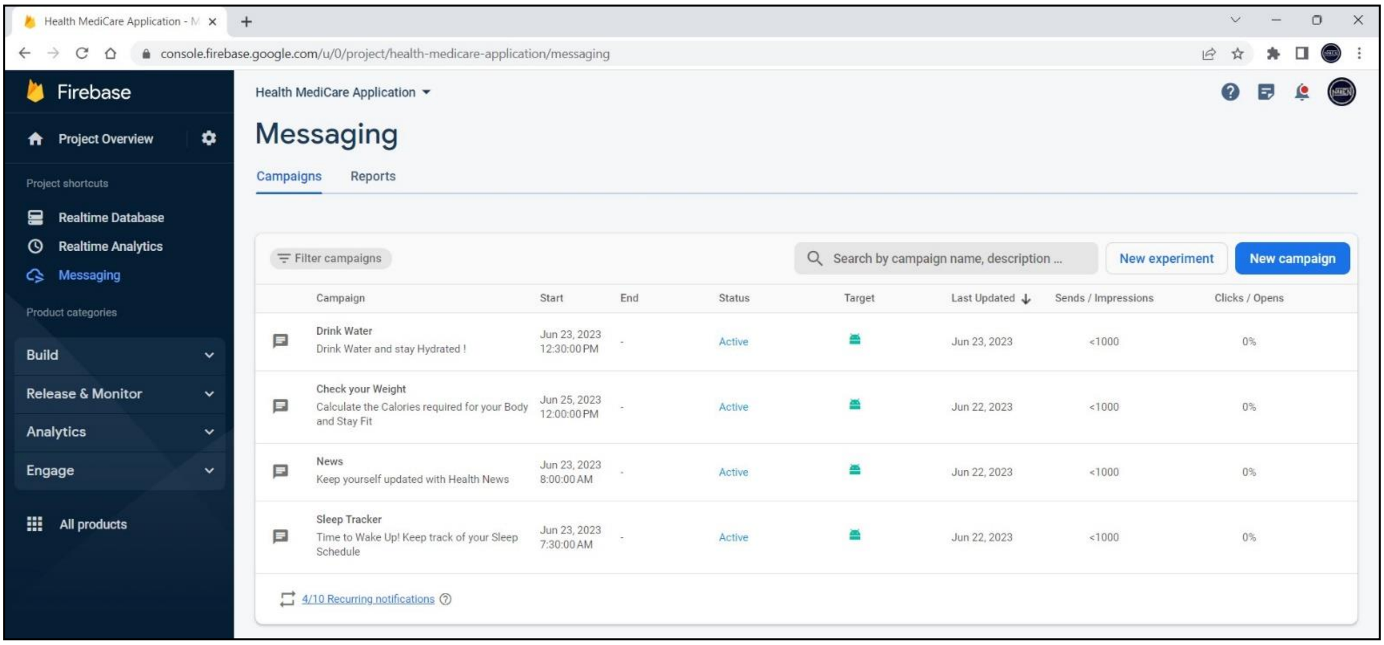
Firebase Realtime Database:


Health MediCare Application - R
console.firebase.google.com/u/0/project/health-medicare-application/database/health-medicare-application-default-rtdb/data
Firebase
Project Overview
Project shortcuts
Realtime Database
Realtime Analytics
Messaging
Product categories
Build
Release & Monitor
Analytics
Engage
All products
Spark
No-cost $0/month
Upgrade
Health MediCare Application
Realtime Database
Data
Rules
Backups
Usage
Extensions
NEW
https://health-medicare-application-default-rtdb.firebaseio.com

```
https://health-medicare-application-default-rtdb.firebaseio.com/
▾ — User
    — mobile: ""
    — name: ""
  ▾ — naren
      — mobile: "7708174320"
      — name: "Naren Adharsh S"
      — password: "adharsh"
      — userEmail: "naren@gmail.com"
  ▾ — narsh
      — mobile: "7708174320"
      — name: "naren"
      — password: "narsh"
```

Database location: United States (us-central1)

Firebase Cloud Messaging :



Messaging

Campaign	Start	End	Status	Target	Last Updated ↓	Sends / Impressions	Clicks / Opens
Drink Water Drink Water and stay Hydrated !	Jun 23, 2023 12:30:00 PM	-	Active		Jun 23, 2023	<1000	0%
Check your Weight Calculate the Calories required for your Body and Stay Fit	Jun 25, 2023 12:00:00 PM	-	Active		Jun 22, 2023	<1000	0%
News Keep yourself updated with Health News	Jun 23, 2023 8:00:00 AM	-	Active		Jun 22, 2023	<1000	0%
Sleep Tracker Time to Wake Up! Keep track of your Sleep Schedule	Jun 23, 2023 7:30:00 AM	-	Active		Jun 22, 2023	<1000	0%

4/10 Recurring notifications

5. ADVANTAGES & DISADVANTAGES :

- **ADVANTAGES :**

Sleep Track : Users can track their Nap / Sleep easily and maintain a healthy Sleep Schedule.

- Non-intrusive sleep tracking using the Sleep API.
- Accurate and battery-efficient sleep data.
- User-friendly interface with sleep data visualization.
- Insights and guidance to improve sleep quality

- **DISADVANTAGES :**

Unable to Consult a doctor: Users cannot Book an Appointment / Consult a doctor.

Unable to Order Medicine: Users cannot Order Medicine that are prescribed by the Doctor.

Unable to make Payments: Users cannot make Payments via the App. A user cannot pay the Doctor Consultation Fee / Medicine Purchase Payment.

Login Every time: The user has to login Every time with his credentials.

6) APPLICATIONS :

- Individuals who want to track their sleep patterns and improve sleep quality.
- People with sleep disorders who can use the app to monitor their sleep and share data with healthcare professionals.
- General wellness enthusiasts seeking to optimize their health through better sleep.

7) CONCLUSION:

Summarizing Entire Work, we have developed a Medical Care Application, which can act as **one Stop Application** for All health-related issues. For Now, we have done only **7 modules (Login, Dashboard, Calorie Manage, Health News Article, Sleep Tracker, Maps Search, Emergency Dial)**. Along with the Reminder Notifications, the App can be useful in Day-to-Day Life.

Health Medicare App offers a valuable tool for users to understand their sleep patterns and make informed choices to improve their overall sleep health. By leveraging the Android Sleep API and focusing on user-friendly data visualization, the app empowers individuals to take control of their sleep and achieve a more rested and productive life.

8) FUTURE SCOPE :

In the Future, we are planning to overcome current Disadvantages by including Doctor Search according to Specific Location, Direct Consultation with Doctor (Call/Video), Book an Appointment with Doctor, Order Medicine According to the Doctor Prescription, Cart, Track Medicine Order, Payment Portal, Health Insurance. With this the App becomes a Complete Package, thus can be launched in Play store.

11 BIBILOGRAPHY :

We took our inspiration from,

1) **Practo** – An Online Doctor Consulting Website / App.
2) **Pharmeasy** – An Online Medicine Order Website / App.

We referred multiple websites,

1) Macronutrient Calculation Websites.
2) MedicalNewsToday.com API / Website.
3) Google Maps.

APPENDIX

A. Source Code

The Entire Source Code can be found in the Below GitHub Link.

Link : https://github.com/Naren-7701/Health_MediCare_App

The **MainActivity.kt** File Code :

```
package com.example.health_medicare_application

class MainActivity : ComponentActivity() {
    private lateinit var databaseHelper1: UserDatabaseHelper
    private lateinit var databaseHelper2: MedicalDatabaseHelper

    @RequiresApi(Build.VERSION_CODES.TIRAMISU)
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        databaseHelper1 = UserDatabaseHelper(this)
        databaseHelper2 = MedicalDatabaseHelper(this)
        setContent {
            Health_MediCare_ApplicationTheme {
                Surface(
                    modifier = Modifier.fillMaxSize(),
                    color = MaterialTheme.colorScheme.background
                ) {
                    val firebasedb = FirebaseDatabase.getInstance();
                    val reference = firebasedb.getReference("User");
                    val get_permission = rememberLauncherForActivityResult(
                        ActivityResultContracts.RequestPermission()
                    ) { isGranted ->
                        if (isGranted) {
                            // Permission accepted Do Something
                        } else {
                            // Permission not accepted show message
                        }
                    }
                    SideEffect {
                        get_permission.launch(POST_NOTIFICATIONS)
                    }

App(applicationContext,reference,databaseHelper1,databaseHelper2)
                }
            }
        }
    }
}
@Composable
fun App(context:Context,databaseReference: DatabaseReference,databaseHelper1:
UserDatabaseHelper,databaseHelper2: MedicalDatabaseHelper) {
    val navController = rememberNavController()
    NavHost(
        navController = navController,
        startDestination = "splashscreen"
    ) {
        composable("splashscreen") {
            SplashScreen(navController)
        }
        composable("reg") {
            RegistrationPage(context, navController, databaseReference,
databaseHelper1)
        }
        composable("medusreg") {
            MedicalRegPage(context, navController, databaseHelper2)
        }
        composable("login") {
            LoginPage(context, navController, databaseHelper1)
        }
        composable("forgotpw") {
            ForgotPasswordPage(context, navController, databaseReference,
databaseHelper1)
        }
        composable("dashboard/{email}") { backStackEntry ->
            val email = backStackEntry.arguments?.getString("email")
            DashboardPage(navController, email.toString(), databaseHelper1,
databaseHelper2)
        }
        composable("caloriemgt") {
            CaloriePage(navController)
        }
        composable("article") {
            HealthArticlePage(navController)
        }
        composable("maps"){
            MapPage(navController)
        }
        composable("sleep"){
            SleepTrackerPage(navController)
        }
        composable("emergency"){
            EmergencyContactPage(navController)
        }
    }
}
@OptIn(ExperimentalMaterial3Api::class)
@Composable
```

```
fun TopBar(abc:String) {
    CenterAlignedTopAppBar(
        colors = TopAppBarDefaults.smallTopAppBarColors(containerColor =
purple673),
        title = {
            Text(
                abc, color = purewhite,
                fontSize = fnt24, fontWeight = txtbold
            )
        })
}
@Composable
fun TopApplicationBar(abc:String,navController: NavController) {
    Row(
        horizontalArrangement = Arrangement.SpaceBetween,
        verticalAlignment = Alignment.CenterVertically,
        modifier = Modifier
            .fillMaxWidth()
            .height(60.dp)
            .background(purple673)
    ) {
        val openDialog = remember { mutableStateOf(false) }
        val context = LocalContext.current
        Text(
            abc, color = purewhite,
            fontSize = fnt24, fontWeight = txtbold,
            modifier = Modifier.padding(start = 15.dp)
        )
        IconButton(
            onClick = { openDialog.value = true; }) {
            Icon(
                imageVector = Icons.Outlined.Logout,
                contentDescription = "Logout",
                tint = purewhite,
                modifier = icon,
            )
        }
        if (openDialog.value) {
            AlertDialog(
                onDismissRequest = { openDialog.value = false },
                title = { Text(text = "Logout") },
                text = { Text("Are you sure you want to Logout ?") },
                confirmButton = {
                    Button(
                        onClick = {
                            openDialog.value = false;
                            Toast.makeText(
                                context, "Logout Successful ",
                                Toast.LENGTH_SHORT
                            ).show()
                            navController.navigate("login")
                        }, colors = ButtonDefaults.buttonColors(purple673),
                        shape = rcshape
                    )
                    {
                        Text(
                            "Yes", color = purewhite,
                            fontSize = fnt18, fontWeight = txtbold,
                        )
                    }
                },
                dismissButton = {
                    Button(
                        onClick = {
                            openDialog.value = false
                        }, colors = ButtonDefaults.buttonColors(purple673),
                        shape = rcshape
                    )
                    {
                        Text(
                            "No", color = purewhite,
                            fontSize = fnt18, fontWeight = txtbold,
                        )
                    }
                },
            )
        }
    }
}
@Composable
fun BottomBar(navController: NavController) {
    Row(
        modifier = Modifier
            .fillMaxWidth()
            .height(55.dp)
            .background(purple673),
        horizontalArrangement = horzspacear
    )
    {
        val butcolor = ButtonDefaults.buttonColors(purple673)
        val size24 = Modifier.size(24.dp)
        Button(
            onClick = { navController.navigate("caloriemgt") },
            colors = butcolor,
        ) {
            Icon(
                imageVector = Icons.Outlined.RiceBowl,
                contentDescription = "Calorie",
                modifier = size24
            )
        }
        Button(
            onClick = { navController.navigate("article") },
            colors = butcolor,
        ) {
            Icon(
                imageVector = Icons.Filled.Article,
                contentDescription = "Article",
                modifier = size24
            )
        }
        Button(
            onClick = { navController.navigate("sleep") },
            colors = butcolor,
        ) {
            Icon(
                imageVector = Icons.Filled.Bedtime,
                contentDescription = "Doctor",
                modifier = size24
            )
        }
        Button(
            onClick = { navController.navigate("maps") },
            colors = butcolor,
        ) {
            Icon(
                imageVector = Icons.Filled.LocationOn,
                contentDescription = "Map",
                modifier = size24
            )
        }
        Button(
            onClick = { navController.navigate("emergency") },
            colors = butcolor,
        ) {
            Icon(
                imageVector = Icons.Filled.Contacts,
                contentDescription = "Emergency",
                modifier = size24
            )
        }
    }
}
```