.

Internship Final Report- 2023-2024

Sai Sathwik Kosuru
Young Research Fellow
HAI Healthcare
Mentor: Dr. Maitreya Maity

14 January, 2024, Sunday

Problem Statement	Create an Android App for IoT data acquisition, processing, and analytics
Tech Stack	● Java ● Kotlin ● SQlite Databases: ○ Patient Metadata ○ Visits Database ○ Clinical Database ○ Readings Database
Functionality	● Connect to the IOT Device ● Initiate Bidirectional Communication ● Process data ● Display real-time graph ● Store in CSV file and local database ● Display the data on-demand
Version	1.0
Completed Functionality	● USB Serial Bidirectional Communication ● BLE Bidirectional Communication ● Real-time Data Processing ● Real-time graph display ● Database Integration
Pending Functionality	● Bluetooth Classic Communication ● Integration with the Cloud API

Basic Functioning of the Android App

Algorithm

1. User starts the app
2. Home screen displays which contains 2 buttons
    a. Register: To add new patient
    b. Access Records: To view previous readings/access other data related to already existing patient
3. If Register is selected
    a. Patient Registration screen opens
        i. It asks for
            1. Demography:
                a. Patient id: alphanumeric
                b. Patient name: text
                c. Gender: Radio Button
                d. Age: Numeric
                e. Blood group: drop down menu
                f. Race: drop down menu
                g. Height: Number Picker
                h. Weight: Number Picker
                i. BMI: Colored bar dynamically set according to height and weight keyed in by the user
            2. Commorbidities(All are radio buttons):
                a. Diabetes
                b. HTN
                c. Asthma
                d. Smoking
                e. Alcohol
                f. MI
                g. CVA
                h. Stent
                i. CABG
        ii. If user clicks add button, it will store the data in patient database and opens recording screen if all the data is entered. Else it prompts the user to enter all values
        iii. If user clicks cancel button, it erases all the data and moves to home screen
4. If Access records is selected, dialog box appears asking user to enter patient id
    a. The following data appears:
        i. Patient Id

ii. Patient name
iii. Age
iv. Sex
v. Blood Group
vi. Date of Registration

b. When user clicks new button,
    i. Setup screen opens. It asks for:
        1. Height: Number Picker
        2. Weight: Number Picker
        3. BMI: Color bar dynamically set based on height and weight
        4. Diabetes
        5. HTN
        6. Asthma
        7. Smoking
        8. Alcohol
        9. MI
        10. CVA
        11. Stent
        12. CABG
    ii. If user clicks on submit, dialog box opens asking user to enter chest size
        1. Record screen opens
            a. If user selected Bluetooth, Bluetooth communications happens
            b. If user selects USB Serial, Serial communication happens
            c. User is required to enter following values
                i. Frequency start value
                ii. Frequency end value
                iii. Frequency step
                iv. Count of data
            d. User sets the above values by clicking on set configuration button
            e. Start button initiates connection and bidirectional communication
            f. Stop button disconnects the device
            g. Retake button refreshes the screen
            h. Submit and Generate Button saves the csv file records that visit and it’s readings and then opens an acknowledgement screen showing the following:

i. Graph of current readings
ii. Graph of previous readings
iii. Average bioimpedance
iv. Change from previous day's data
v. Change from previous month's data
vi. Change from previous data all in percentages

i. 2 graphs showing real-time plotting of bioimpedance and phase
j. A table showing
    i. Frequency
    ii. Bioimpedance
    iii. Phase
    iv. Electrode Combination 1
    v. Electrode Combination 2

c. When user clicks view button
    i. Previous visits screen opens showing the following:
        1. Visit number
        2. Date
        3. Time
    ii. When user clicks modify button, record screen opens for updating the readings for that visit date and visit time
    iii. When user clicks view button, archives screen opens showing the following from readings database at that particular visit date and time:
        1. Bioimpedance chart
        2. Phase Chart
        3. A table showing
            a. Frequency
            b. Bioimpedance
            c. Phase
            d. Electrode Combination 1
            e. Electrode Combination 2
    iv. When user clicks report button, it displays a report
d. When user clicks clinical button, it displays the following from clinical database
    i. Hieght
    ii. Weight
    iii. BMI
    iv. Diabtetes
    v. HN

vi. Asthma
vii. Smoking
viii. Alcohol
ix. MI
x. CVA
xi. Stent
xii. CABG

e. When user clicks report button, it shows the report

Data Recording in the app

Data recording happens in the recording screen. The flow of the logics is:

1. User selects the mode of communication from:
    a. USB Serial
    b. Bluetooth
2. User sets the following values
    a. Frequency start value
    b. Frequency End Value(must be less than frequency start value. The app doesnt allow for the opposite and prompts the user to recheck the value.)
    c. Frequency step
    d. Data count(Must be greater than 15. For the numbers less than 15, the app prompts the user to enter a value above 15)

    By clicking the set configuration button
3. User clicks start button when the following happens
    a. App initiates connection in the selected mode by searching for the device. Loading animation is displayed.
    b. If no device is found, it alerts the user in ble mode and USB Serial mode but this currently doesn't happen in Bluetooth classic mode
    c. If found, it connects. User permission is required for USB Serial connection.
    d. The app catches the input stream and starts processing the strings.
    e. For processing of strings, app uses regular expressions to find numbers in the string and if the numbers is greater than 150000, it is taken as bioimpedance values and if it is less than 100, it is taken as phase value. Separate arrays are created for both values respectively. Rest of the values are discarded. All the processing happens in the background thread to prevent app crashing.
    f. The numbers are further processed to display chart of bioimpedance on the left in red and that of phase on the right in red. X axis is time and y axis is the value.
    g. A table appears which is automatically scrolled to the latest value with following attributes:
        i. Frequency
        ii. Bioimpedance
        iii. Phase
        iv. Electrode Combination 1
        v. Electrode Combination 2

h. During this communication, the app continuously count the length of arrays and once they reach the data count value entered by the user, electrode combinations change according to the following schema
    i. 0,0
    ii. 0,1
    iii. 0,3
    iv. 0,4
    v. 0,5
    vi. 0,6
    vii. 1,0
    viii. 1,1
    ix. 1,3
    x. 1,4
    xi. 1,5
    xii. 1,6
    xiii. 3,0
    xiv. 3,1
    xv. 3,3
    xvi. 3,4
    xvii. 3,5
    xviii. 3,6
    xix. 4,0
    xx. 4,1
    xxi. 4,3
    xxii. 4,4
    xxiii. 4,5
    xxiv. 4,6
    xxv. 5,0
    xxvi. 5,1
    xxvii. 5,3
    xxviii. 5,4
    xxix. 5,5
    xxx. 5,6
    xxxi. 6,0
    xxxii. 6,1
    xxxiii. 6,3
    xxxiv. 6,4
    xxxv. 6,5
    xxxvi. 6,6

xxxvii. Frequency increments till it reaches frequency stop value where it disconnects and again 36 combinations are set for each frequency.

i. The app sends the following data to the device everytime:
    i. Current Frequency
    ii. Placeholder(some integer)
    iii. Electrode combination 1
    iv. Electrode Combination 2 in form of “float,int,int,int”.
j. The app expects the following data format from the device: “freq:valueHz,RzMag:valueOhm,RzPhase:value\n”
k. The app expects the device to be
    i. Discoverable
    ii. Serviceable
    iii. Less fault-prone

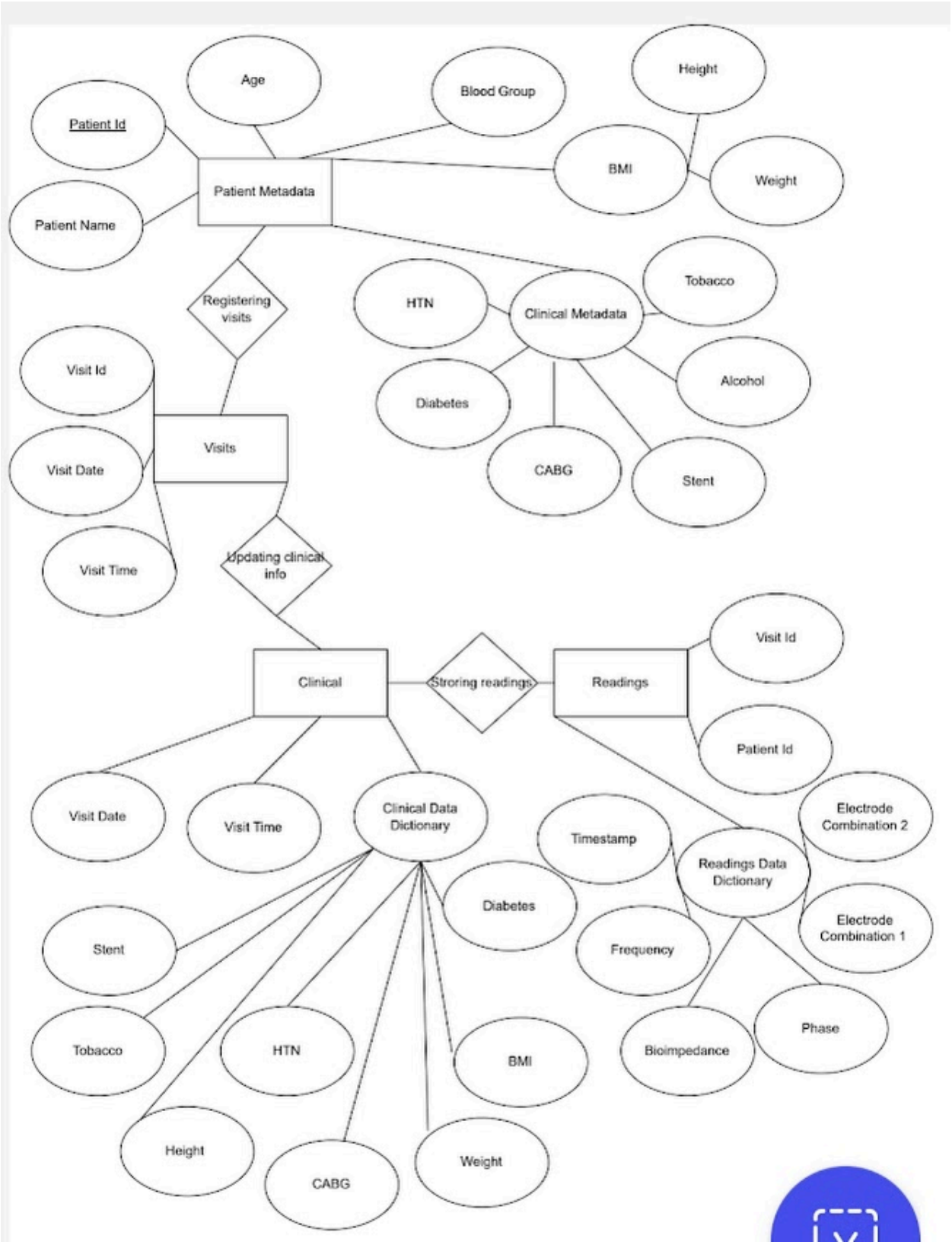
Databases

The app contains the following databases:

1. Patient Database
    a. Schema: Relational
    b. Tech Stack: SQLite
    c. Purpose: To store patient meta data
    d. Characteristic: Master table
    e. Columns/fields:
        i. Patient id: text
        ii. Patient name: text
        iii. Gender: text
        iv. Age: number
        v. Blood group: text
        vi. Race: text
        vii. Date of Registration: text
        viii. Diabetes: number
        ix. HTN : number
        x. Asthma: number
        xi. Smoking: number
        xii. Alcohol: number
        xiii. MI: number
        xiv. CVA: number
        xv. Stent: number
        xvi. CABG: number
    f. Primary Key: Id internally generated
    g. Associated tables:
        i. Status Database
        ii. Clinical Database
        iii. Readings Database
    h. Needed changes: Make patient id unique
2. Status Database:
    a. Schema: Relational
    b. Tech Stack: SQLite
    c. Purpose: To store visit details of the patient
    d. Columns/Fields:
        i. Patient id: text
        ii. Visit: number
        iii. Visit time: text
        iv. Visit date: text
    e. Primary Key: Id internally generated

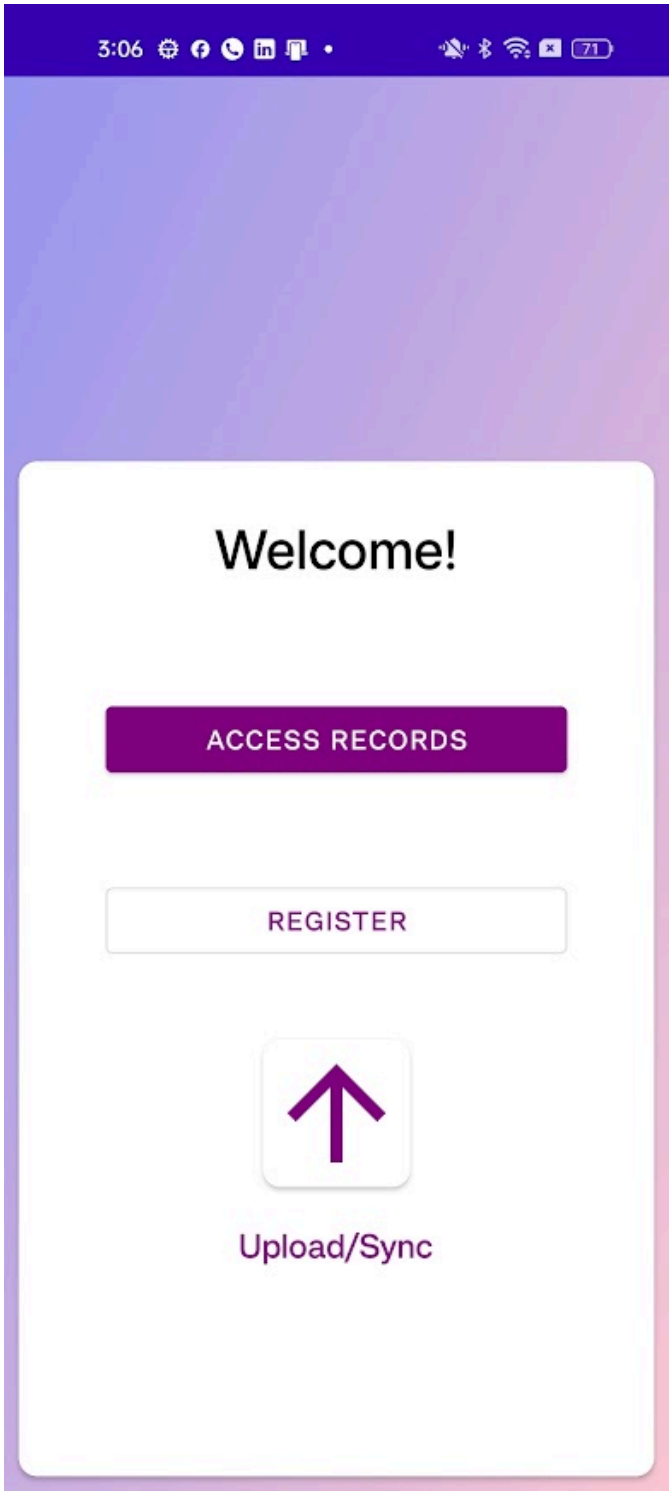
    f. Associated tables: Readings database
    g. Needed changes: NIL
3. Clinical Database
    a. Schema: Relational
    b. Tech Stack: SQLite
    c. Purpose: To store clinical details of the patient periodically.
    d. Columns/Fields:
        i. Patient id: string
        ii. Clinical: JSON containing
            1. Height
            2. Weight
            3. BMI
            4. Diabetes
            5. HTN
            6. Asthma
            7. Smoke
            8. Alcohol
            9. MI
            10. CVA
            11. Stent
            12. CABG
    e. Primary Key: Id internally generated
    f. Associated tables: Patient Database
    g. Needed Changes: Visit field can be added
4. Readings Database
    a. Schema: Relational
    b. Tech Stack: SQLite
    c. Purpose: To store readings from the device
    d. Columns/Fields:
        i. Patient id: text
        ii. Visit date: text
        iii. Visit time: text
        iv. Readings: JSON containing
            1. Timestamp: string
            2. Frequency: number
            3. Bioimpedance: number
            4. Phase: number
            5. Electrode Combination 1: number
            6. Electrode Combination 2: number
    e. Primary Key: Id internal generated

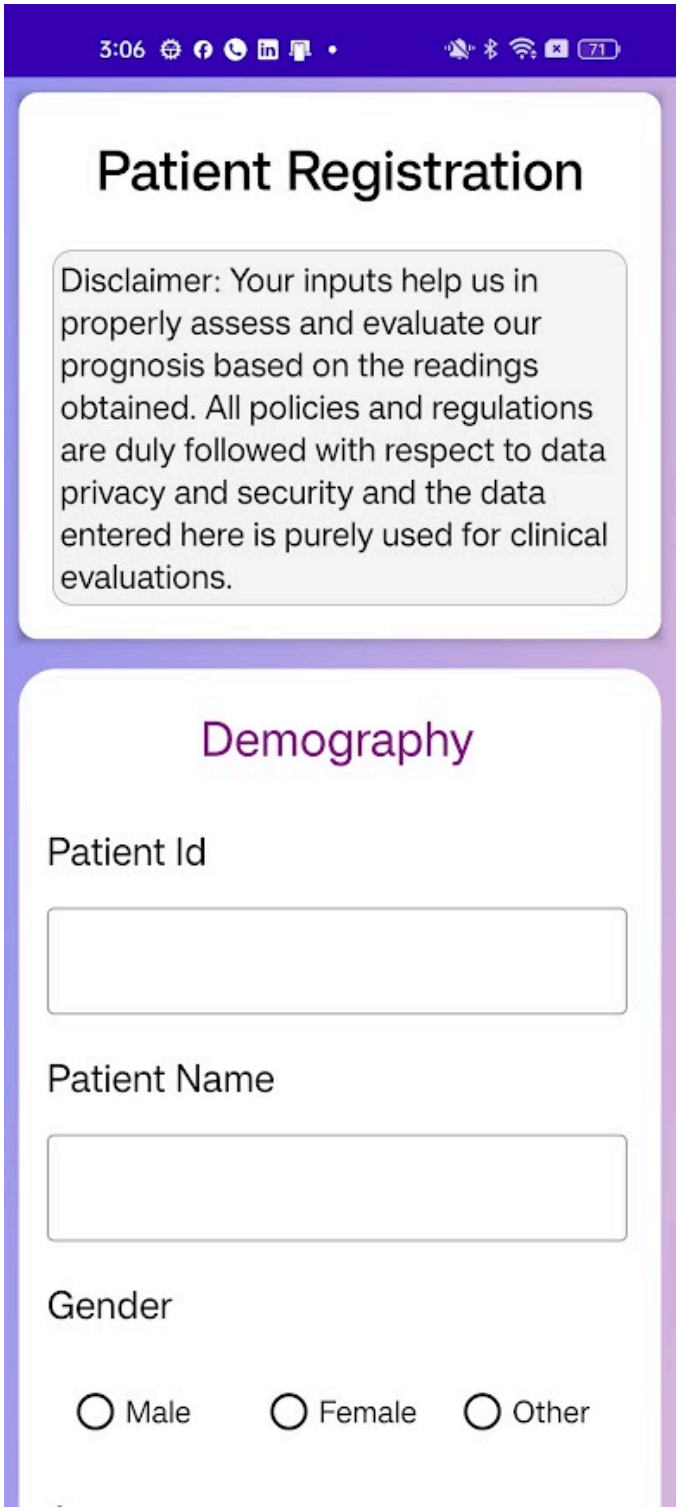
f. Associated tables: status database
g. Needed Changes: NIL

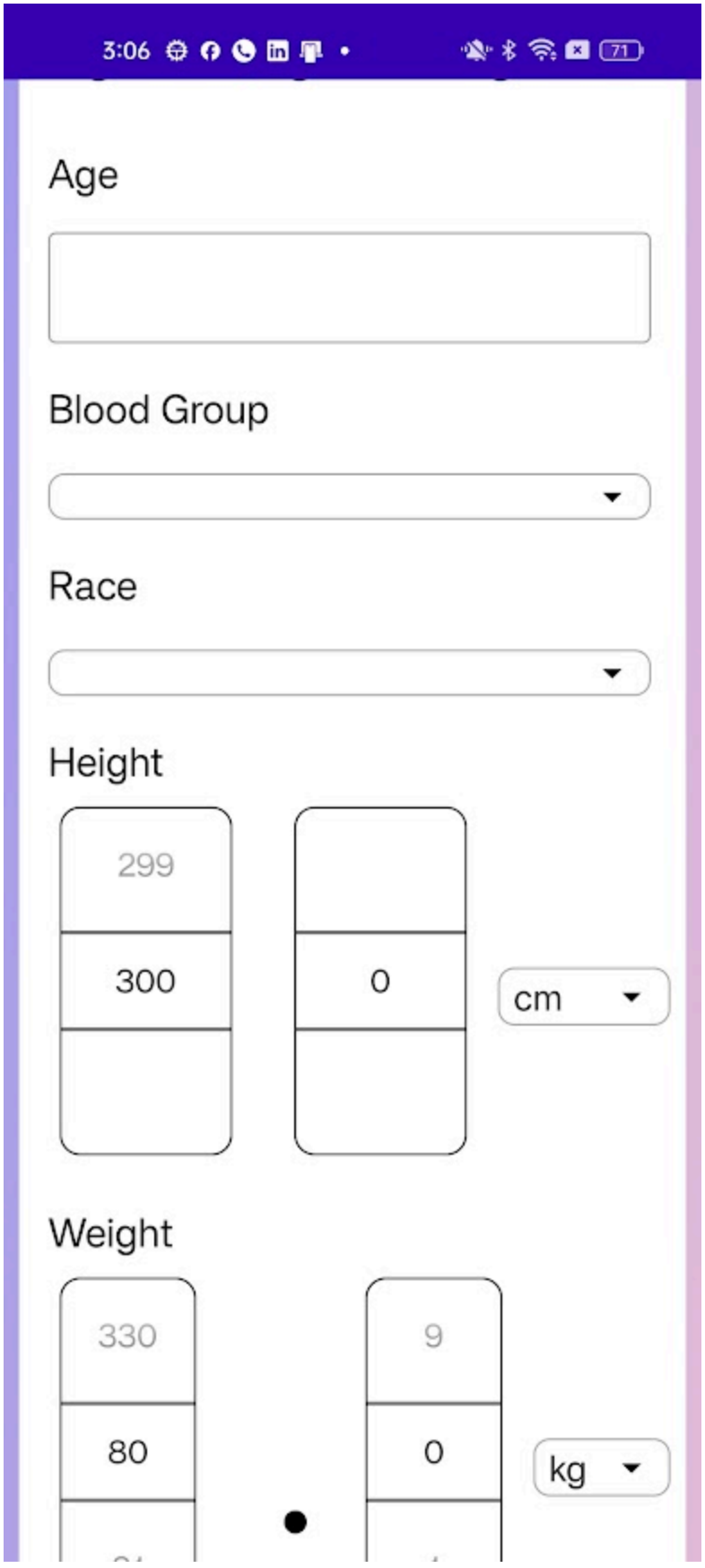

Age
Blood Group
Height
Patient Id
Patient Metadata
BMI
Weight
Patient Name
Registering visits
HTN
Clinical Metadata
Tobacco
Visit Id
Diabetes
Alcohol
Visits
Visit Date
CABG
Stent
Visit Time
Updating clinical info
Clinical
Stroring readings
Readings
Visit Id
Patient Id
Visit Date
Visit Time
Clinical Data Dictionary
Timestamp
Electrode Combination 2
Readings Data Dictionary
Diabetes
Electrode Combination 1
Stent
Frequency
Phase
Tobacco
HTN
BMI
Bioimpedance
Height
CABG
Weight

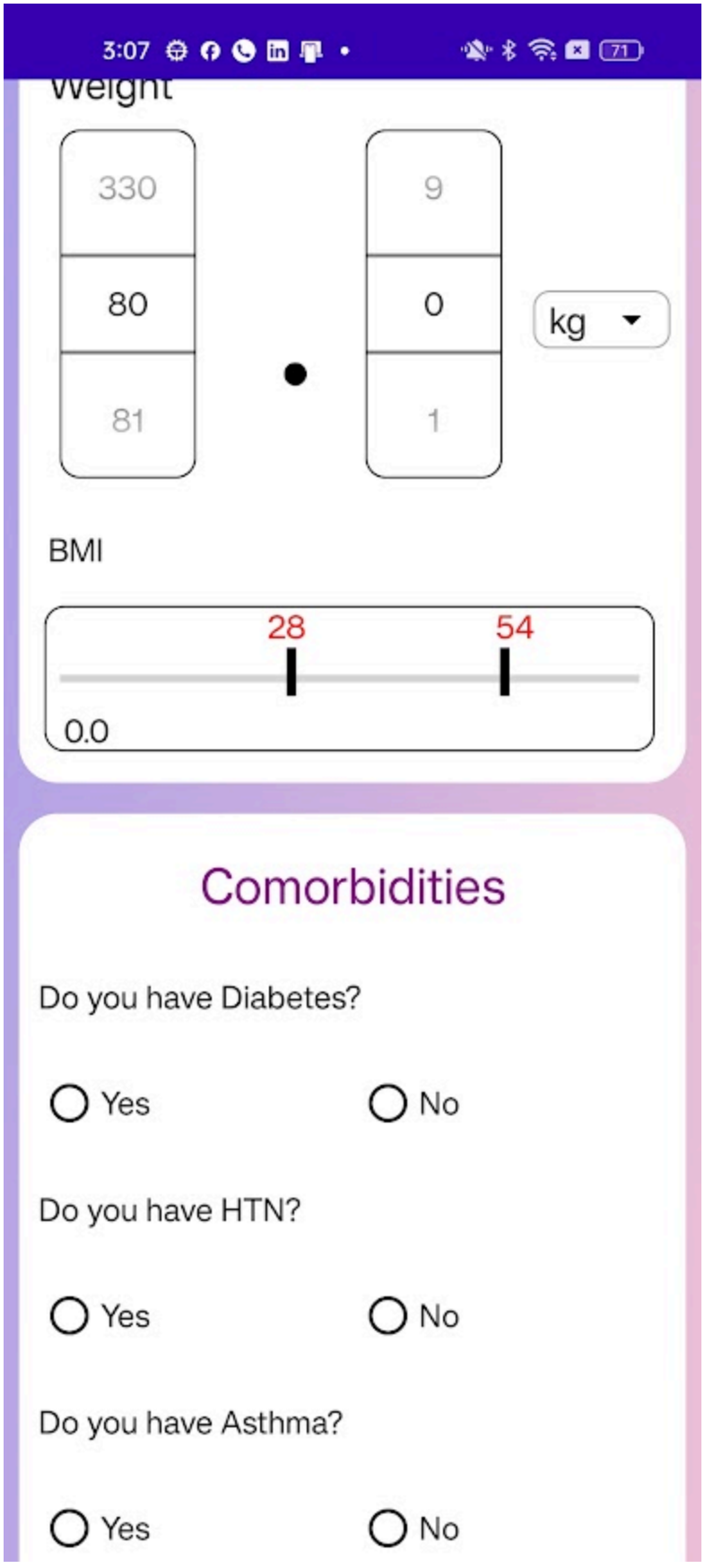
User Interface

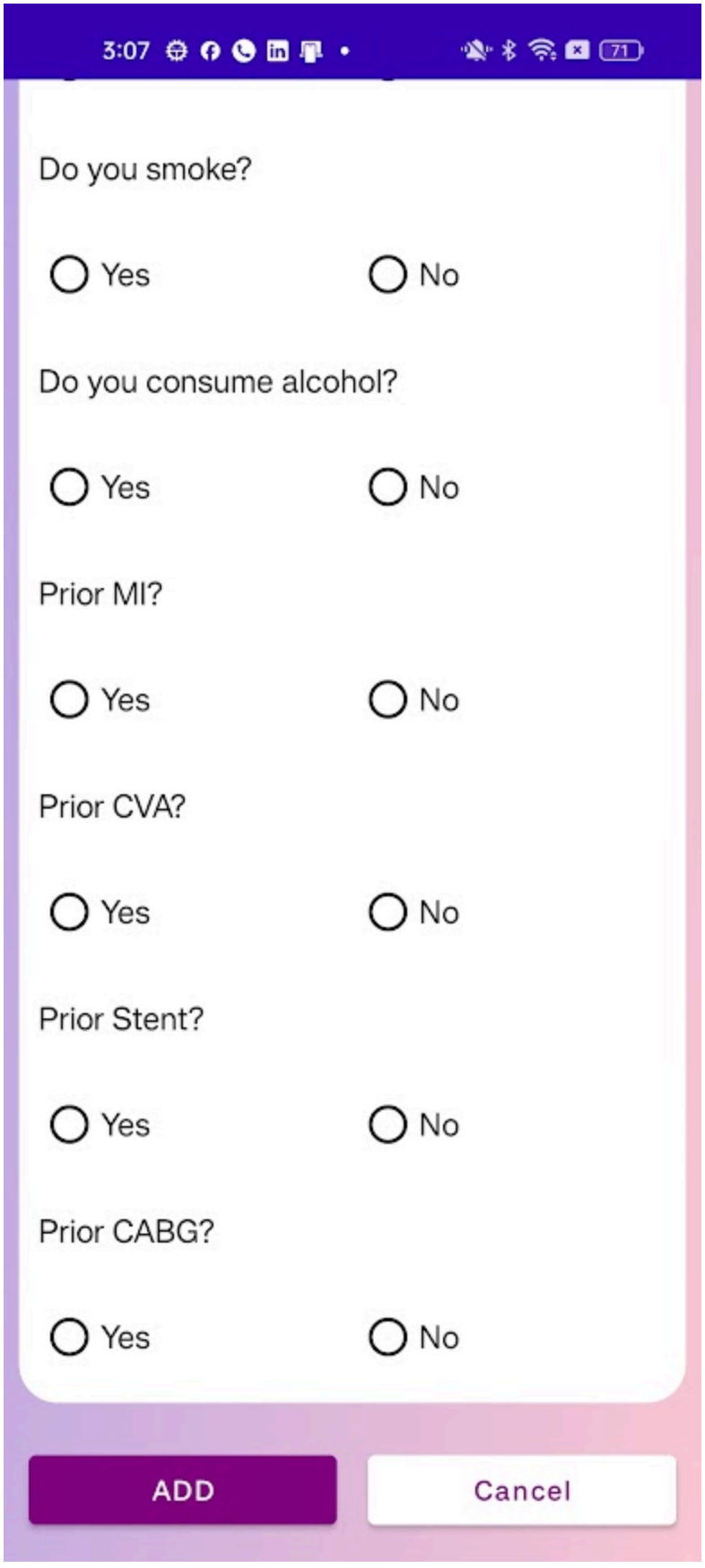
- Home Screen



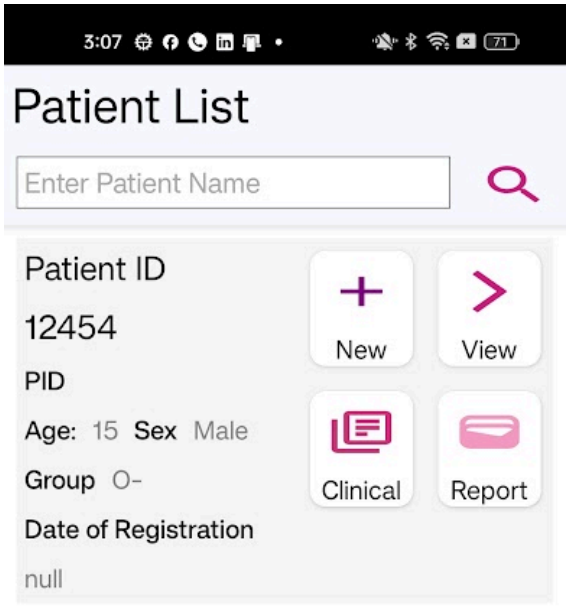
- Patient Registration
- 


3:06
Age
Blood Group
Race
Height
299
300
0
cm
Weight
330
80
9
0
kg

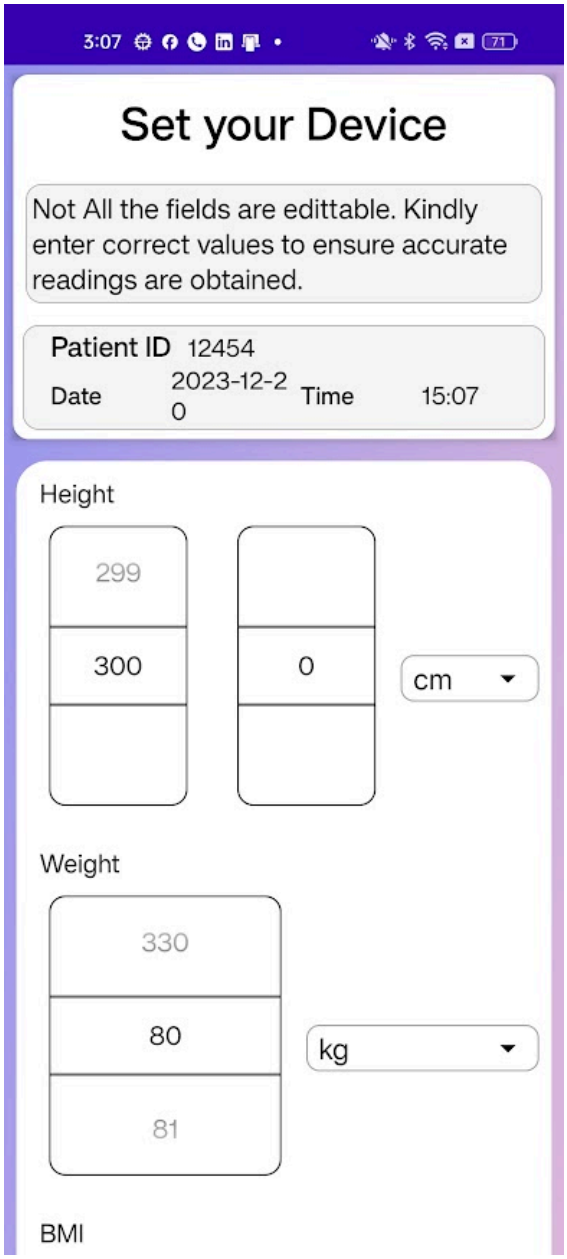

3:07
Weight
330
80
81
9
0
1
kg
BMI
28
54
0.0
Comorbidities
Do you have Diabetes?
Yes
No
Do you have HTN?
Yes
No
Do you have Asthma?
Yes
No

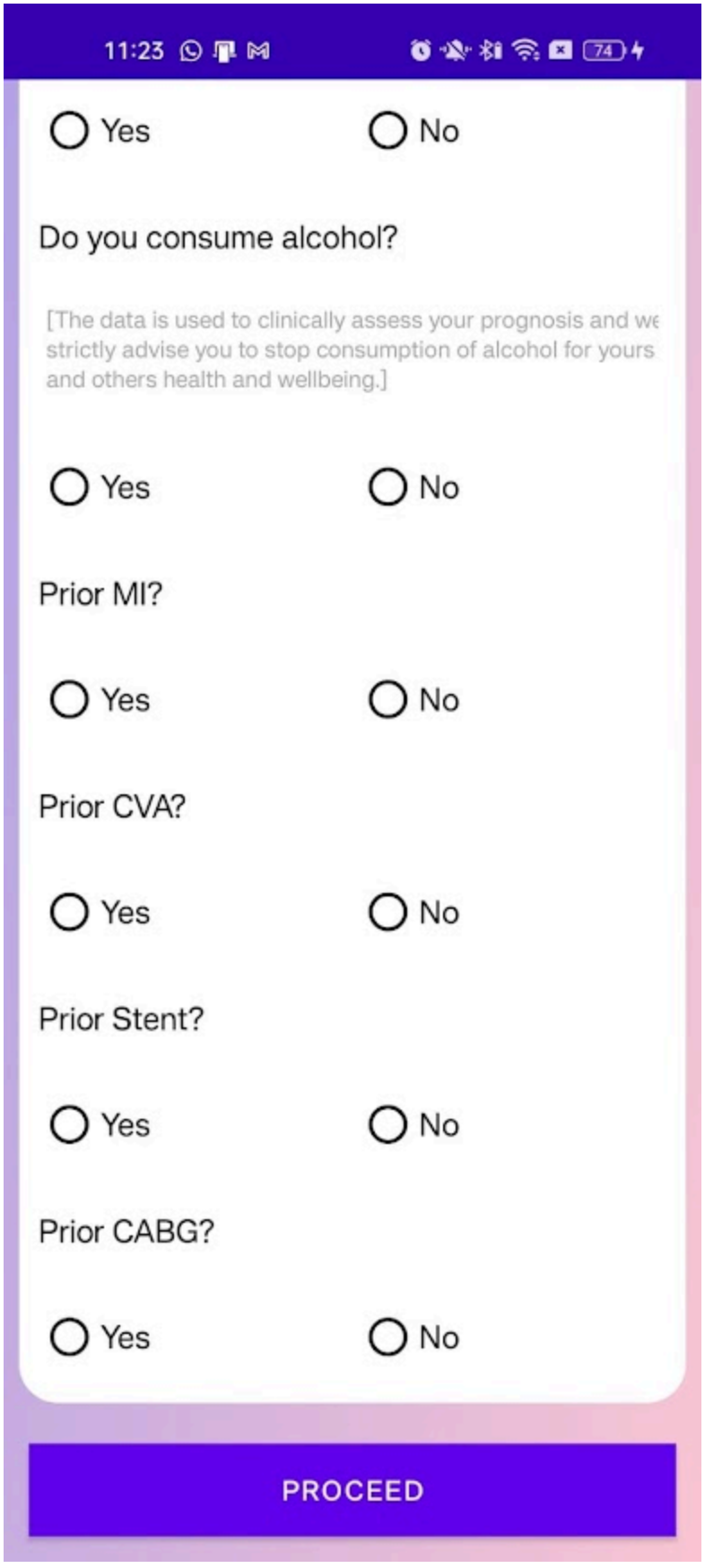

3:07
Do you smoke?
Yes
No
Do you consume alcohol?
Yes
No
Prior MI?
Yes
No
Prior CVA?
Yes
No
Prior Stent?
Yes
No
Prior CABG?
Yes
No
ADD
Cancel

- Patient List

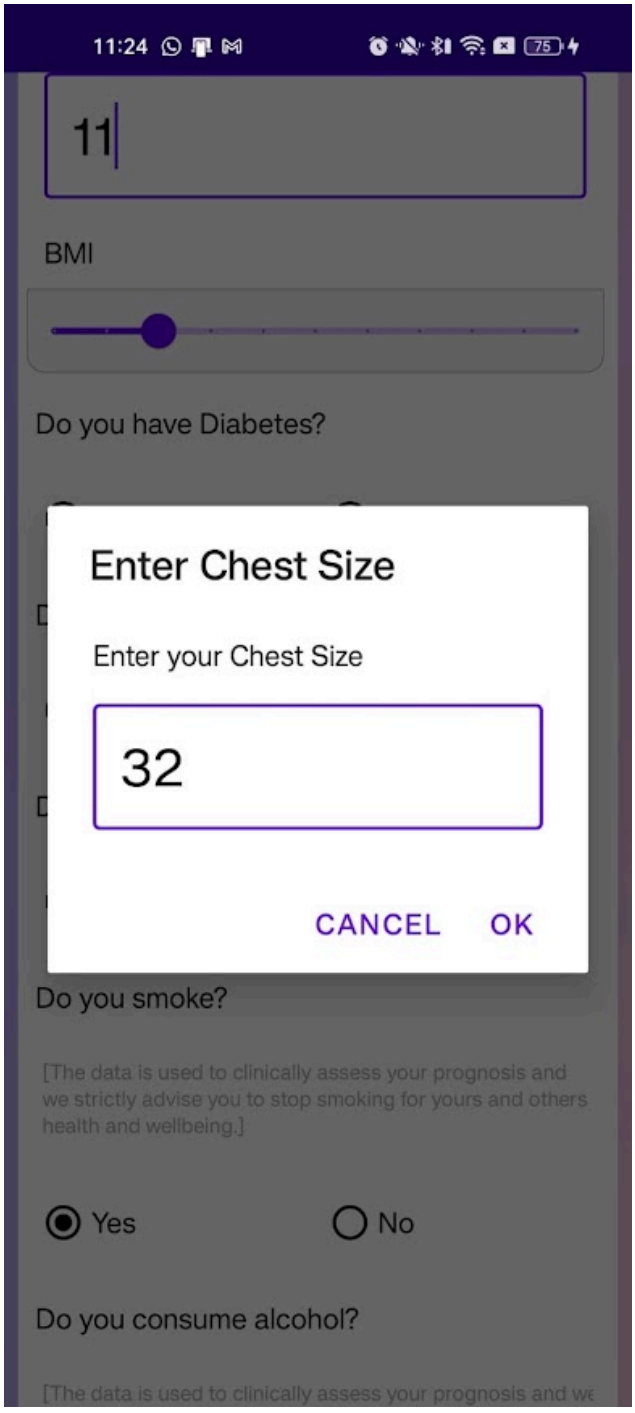


- Setup screen

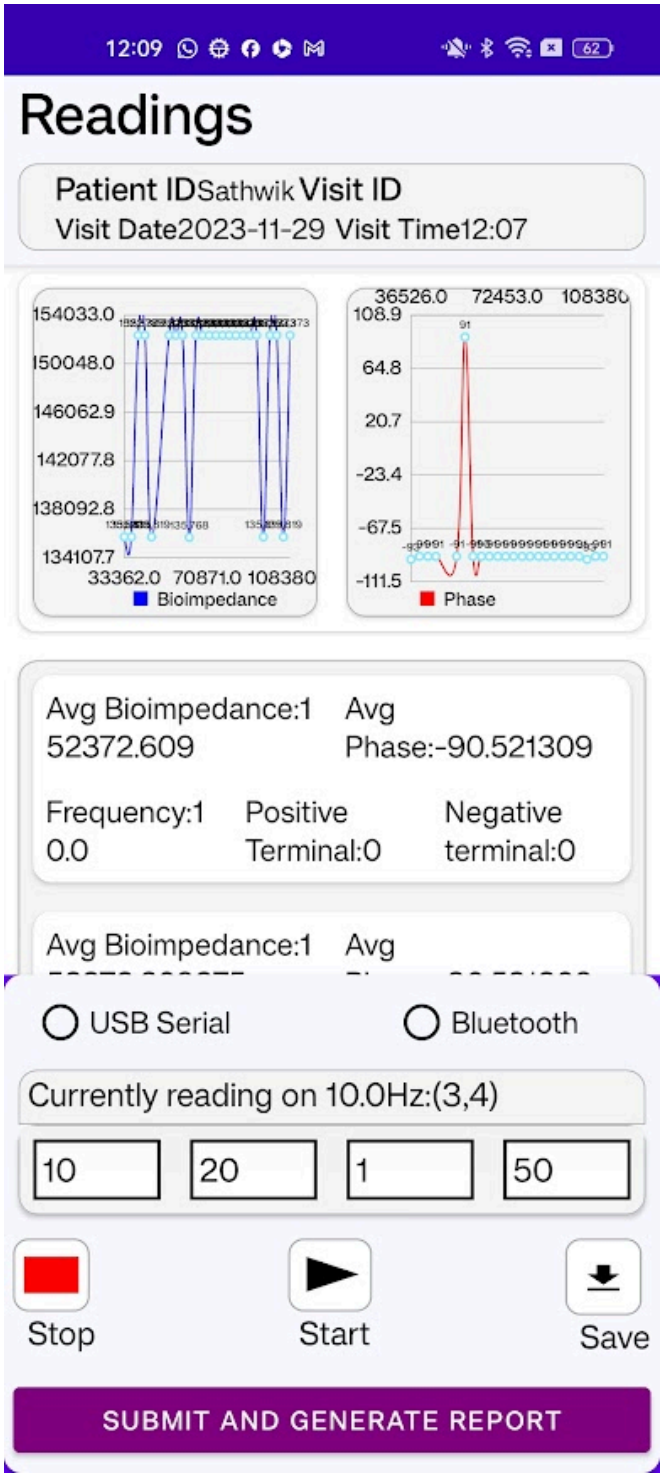



11:23
Yes
No
Do you consume alcohol?
[The data is used to clinically assess your prognosis and we
strictly advise you to stop consumption of alcohol for yours
and others health and wellbeing.]
Yes
No
Prior MI?
Yes
No
Prior CVA?
Yes
No
Prior Stent?
Yes
No
Prior CABG?
Yes
No
PROCEED

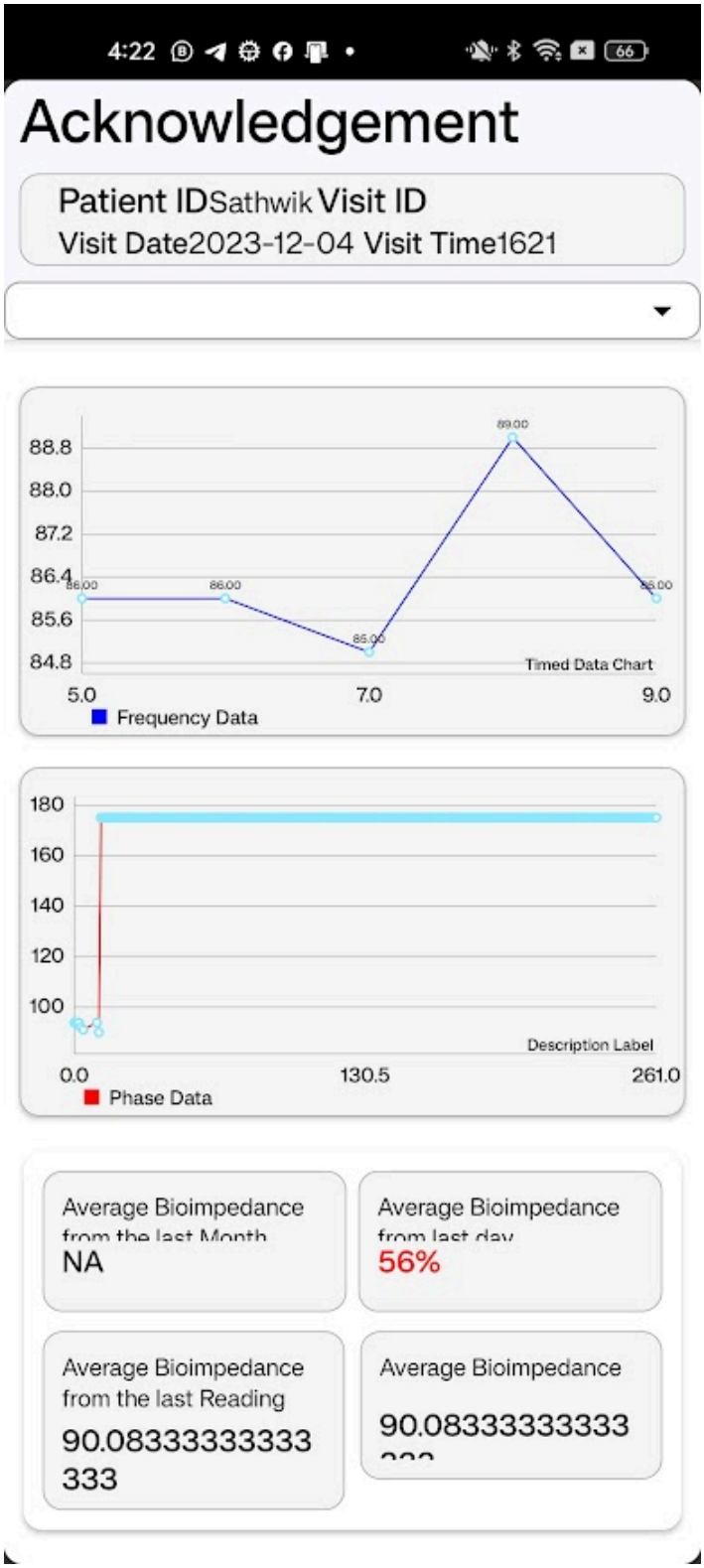
- Chest Size Dialog



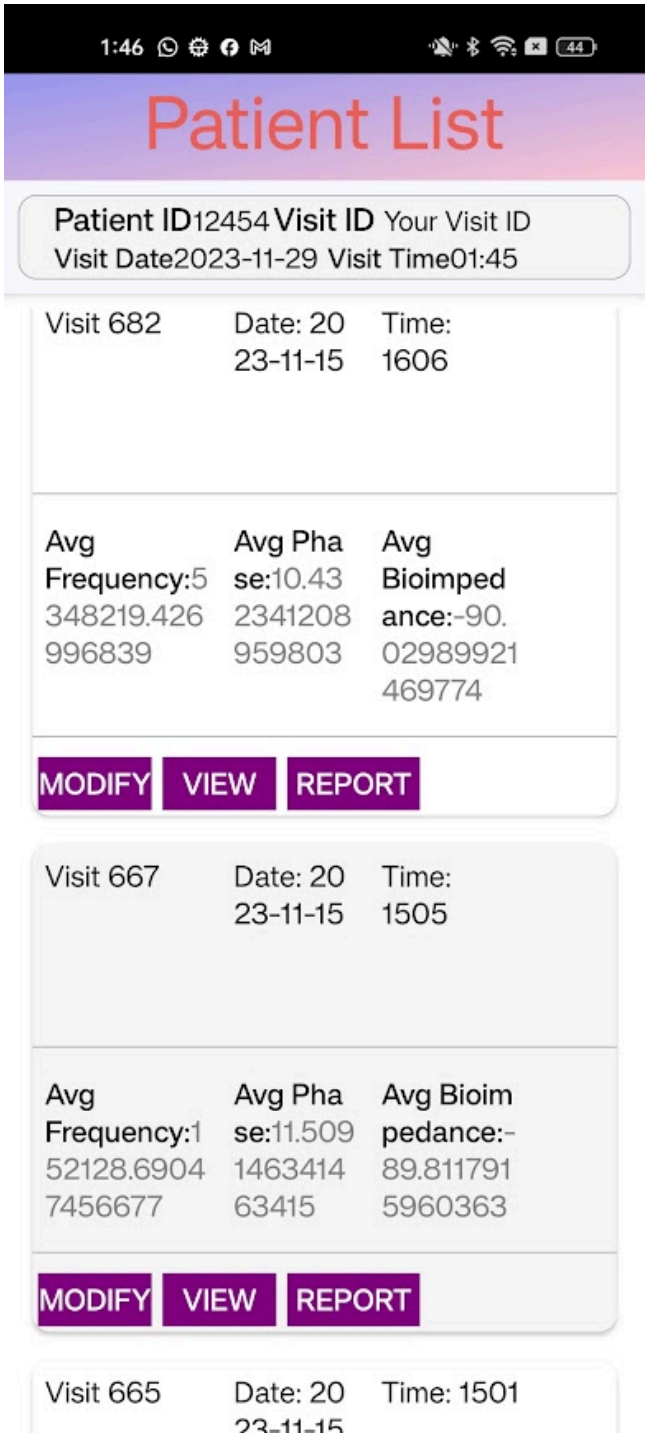
- Record Screen



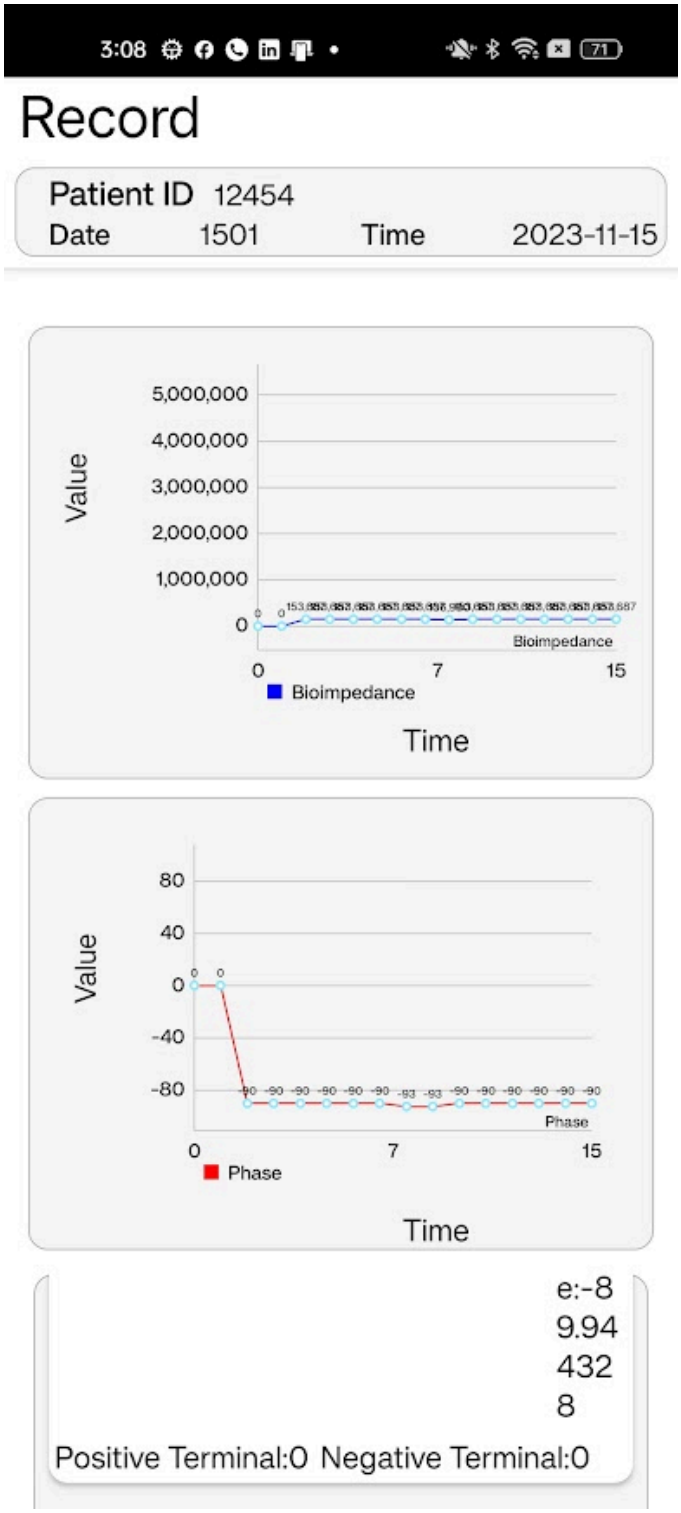
- Acknowledgement screen



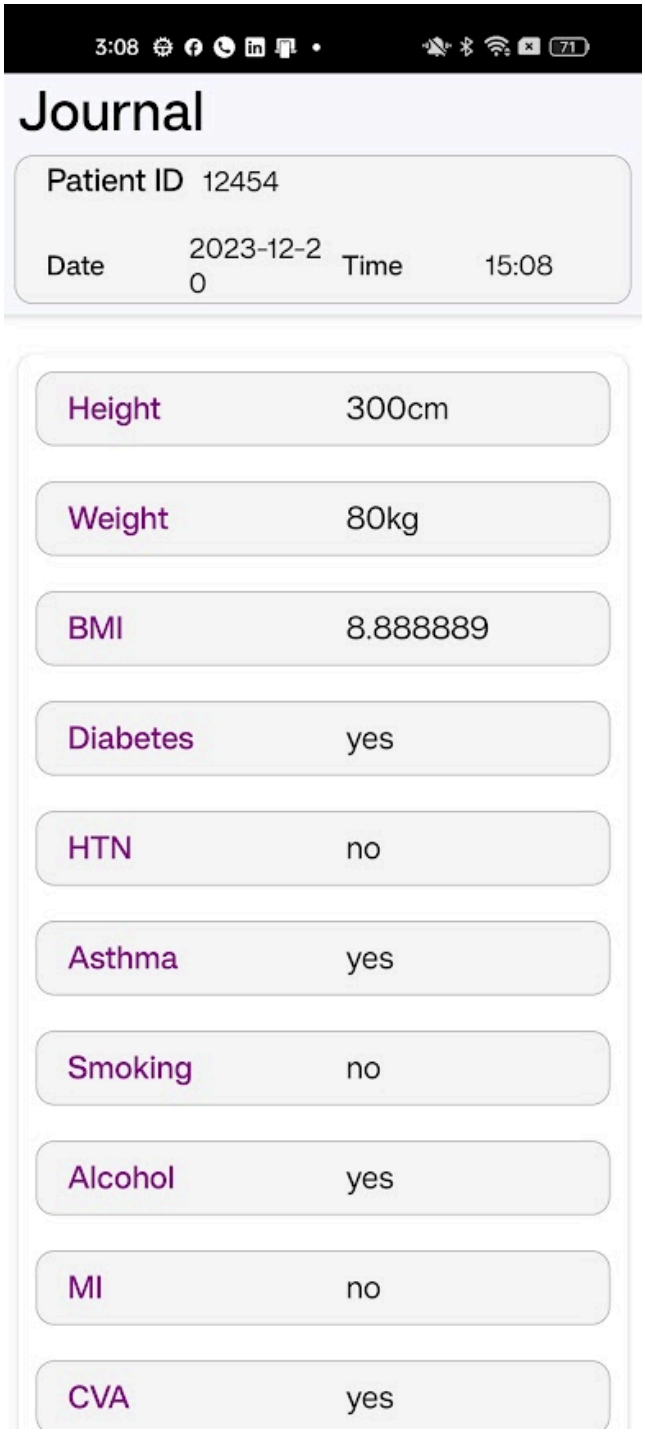
- Previous visits screen

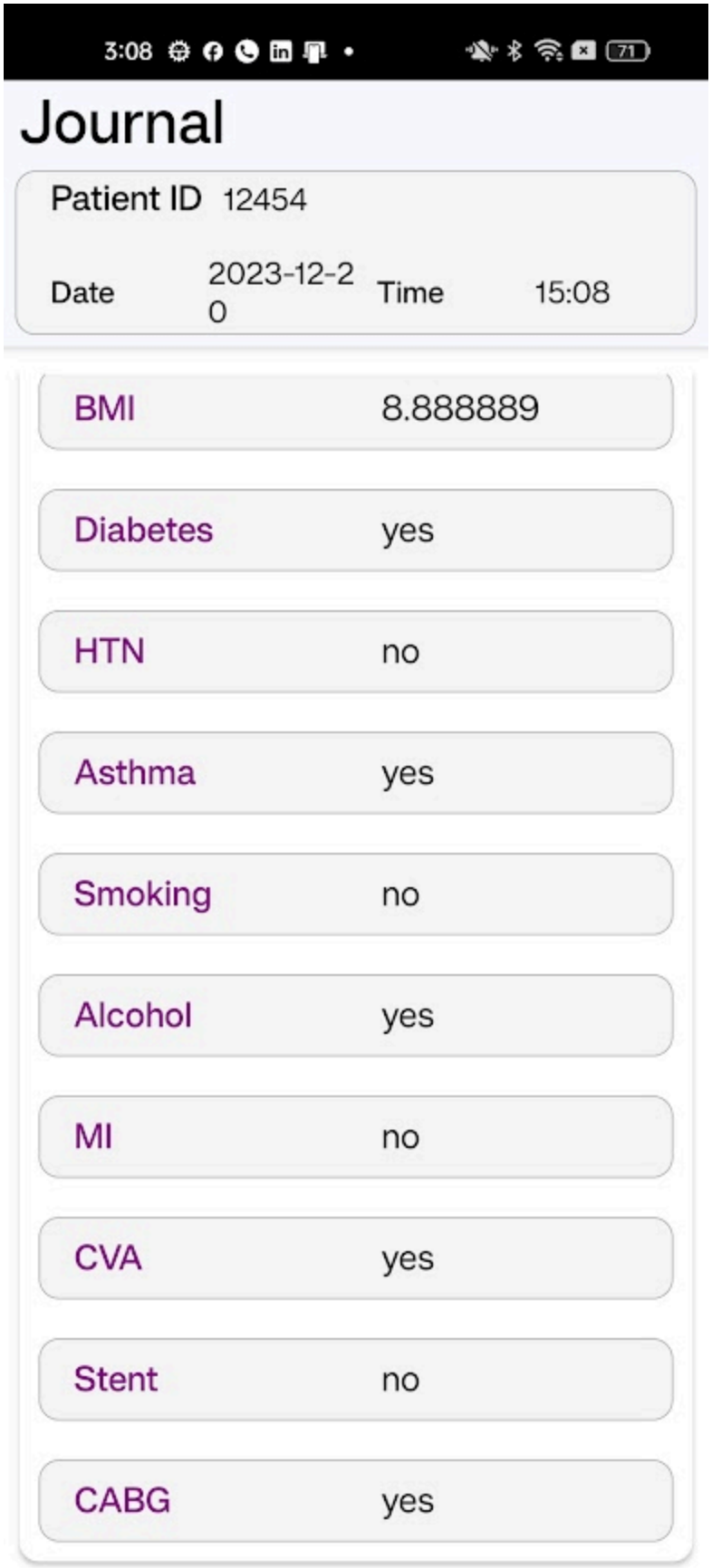


- Archives screen



- Journal Screen




3:08
Journal
Patient ID 12454
Date 2023-12-20
Time 15:08
BMI 8.888889
Diabetes yes
HTN no
Asthma yes
Smoking no
Alcohol yes
MI no
CVA yes
Stent no
CABG yes

Considerations

S.No	Title	Description	Justification/Current Implementation on the Android App
1	Feet-cm conversions in height	UI for inputting patient's height	• Exact conversion between feet and cm in height involves the exact calculation of decimals which is very complex for a common user. • Hence, we may observe a slight inconsistency in feet-cm conversions for height
2	Decimals for weight	UI for exactly inputting patient's weight	• Same case, mentioned above, is applicable here. • BMI calculation is highly subjective and may not need exact values.
3	No setup screen after registration	UI for taking updated clinical parameters of the patient every time before recording	• The details required in the setup screen are already provided by the user in the registration screen and hence, to prevent redundancy in app's operations, this is facilitated. • I will add the chest size entry dialog box for the immediate entry into the recording screen immediately after patient registration screen.
4	Search bar for access records	UI for accessing only a single patient's	• Added for privacy • The app is currently accepting duplicates for patient details. It

		records at a time	will be removed in further sessions.
5	Report button not functional	UI/UX for generating the report covering all the readings taken on the app	• You informed me that you will discuss this feature in future sessions which is still pending
6	Clinical Summary in visits page	Few important pointers shown as a section for each box in visits screen	• You informed me that you will discuss this feature in future sessions which is still pending
7	No acknowledgement from device on powering off but some random data is sent for some time	Needed for UX of the app	• Currently I added an observer in the recordings screen which will inspect the communication process for every 15 seconds and alerts the user in case no more new data is sent be the device • Acknowledgement from the device is still an efficient and productive option
8	Bus fault/cache overflow error on device	Occurring at times on the device side which requires the device to be powered off, put idle for	• Currently the app efficiently handles this and this is verified practically • There is a risk of this issue occurring frequently on the device for long readings.

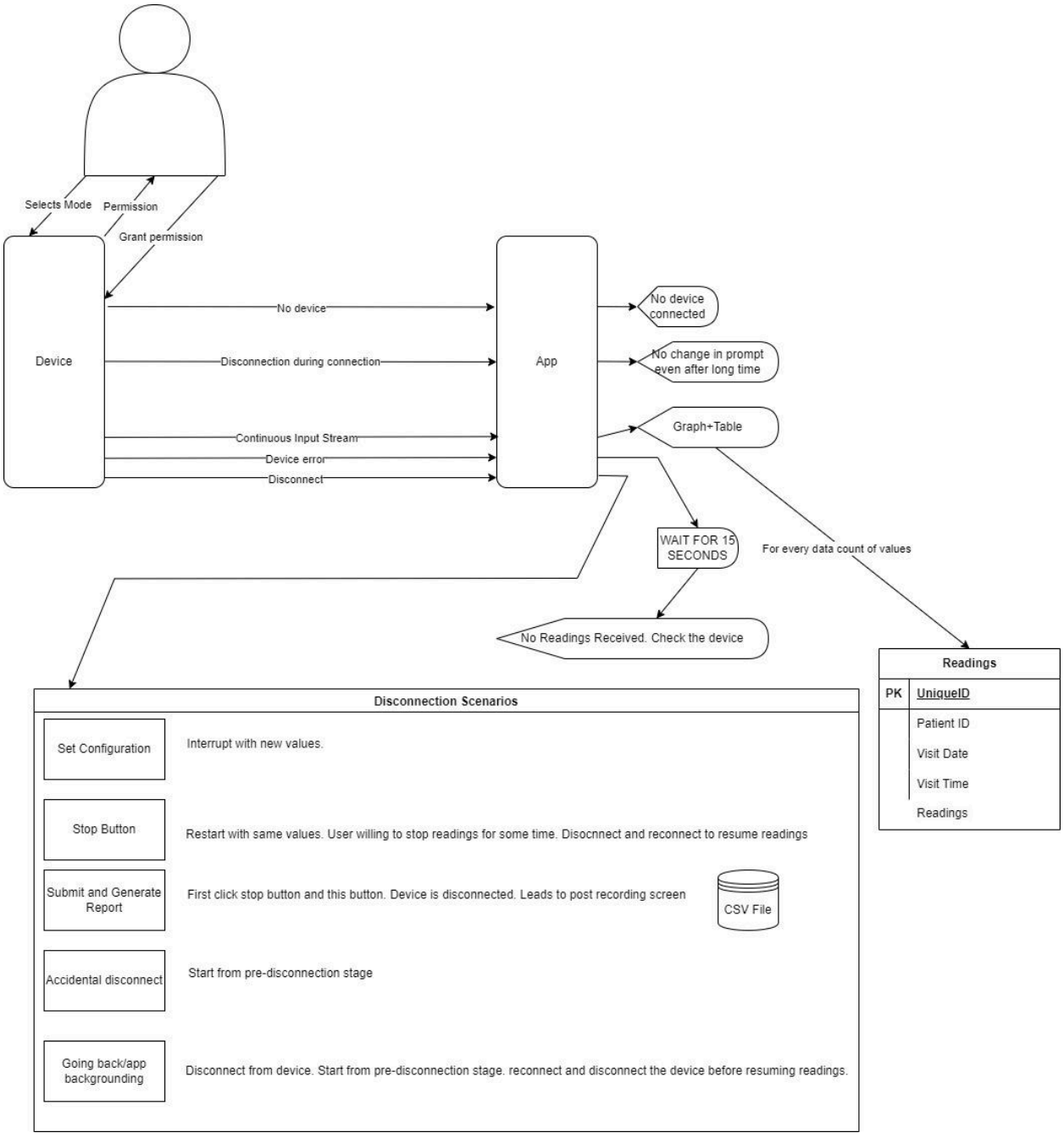
		some time, and then reconnect it for use again	
9	Disconnection and reconnection of already running device from mobile for taking new readings on the app	Need to investigate	• This process takes time and the results can be inducted in the future versions of the app. This issue does not affect the current functionality of the app and the app can be used seamlessly. • Need for error isolation
10	Crash recovery mechanism in readinsgs process	• Systematic addition of dialog boxes on entry/ exit points • Prompt on app backgrounding • Periodic saving into database	• User cannot accidentally exit the readings screen during the recording process since he/she needs to confirm in the dialog box that appears. Similar functionality for acknowledgement screen. • Readings are saved for every datacount number of values as specified by the user. The csv file is also generated on disconnection of the device. This prevents loss of data. • Data processing takes place in background to prevent crashing/ poor performance of the app's UI&UX. Hence, app's backgrounding process automatically effects the performance of the app. Hence, the app is currently designed to disconnect from the device and pause all the activities in the background.

			For resuming, the user needs to disconnect, reconnect and click on start button.

Deployment Characteristics

USB Serial Communication

Corner Case/Test Case	Percentage of Success	Number of Tests Conducted
Device Disconnection	100%	52
Device Powering Off	62%	22
Device connection prior to initiating taking readings	45%	10
Device Reconnection without physical disconnection after the user clicks stop readings	81%	15
Error from the device	95%	30


Selects Mode
Permission
Grant permission
Device
No device
Disconnection during connection
Continuous Input Stream
Device error
Disconnect
App
No device connected
No change in prompt even after long time
Graph+Table
WAIT FOR 15 SECONDS
For every data count of values
No Readings Received. Check the device
Readings
PK
UniqueID
Patient ID
Visit Date
Visit Time
Readings
Disconnection Scenarios
Set Configuration
Interrupt with new values.
Stop Button
Restart with same values. User willing to stop readings for some time. Disocnnect and reconnect to resume readings
Submit and Generate Report
First click stop button and this button. Device is disconnected. Leads to post recording screen
CSV File
Accidental disconnect
Start from pre-disconnection stage
Going back/app backgrounding
Disconnect from device. Start from pre-disconnection stage. reconnect and disconnect the device before resuming readings.

Concluding Remarks

- App complete logical flow implemented according to requirements
- UI updates made on the app according to requirements
- The app is working fine on Deployment with no issues
- USB Serial Communication working flawlessly
- USB Serial is happening in background thread and so, it stops USB Serial Communication on backgrounding the app
- BLE communication happening flawlessly but breaches/corner cases should still be implemented
- Bluetooth classic bidirectional communication to be made robust and hence need major integrations. Only basic functionality is added but it is not perfectly integrated.
- Databases working perfectly but the timing and date formats to be carefully set where there is a high probability of confusion/error.
- CSV File format and saving perfectly implemented and readings database perfectly integrated and everything is happening flawlessly in real-time.