

Smart Posture App

30th July, 2022

Introduction:

Maintaining good posture will help you keep your spine healthy and prevent chronic back pain. Since it's so simple to develop the habit of slouching, maintaining proper posture can assist alleviate discomfort and tension in the body. So, let technology to assist you in maintaining a straight posture throughout the day.

Your smartphone's **Smart Posture** app uses information from Arduino Nano BLE 33 attached to the user's back, and when a predetermined threshold is met, the animation in the app changes color and the device vibrates slightly, alerting the user to adjust their posture. Along with real-time angle monitoring, collected data is also sent to a cloud-based mongo DB platform for later analysis.

Smart Posture App:

The application was thoughtfully created to satisfy user's requirements and offer a complete tool for keeping an eye on neck and back posture. The app was created using the Java programming language and Android Studio. Data was also gathered and stored on the Mongo DB realm cloud for later analysis.

FLOW CHART OF APP WORKING:

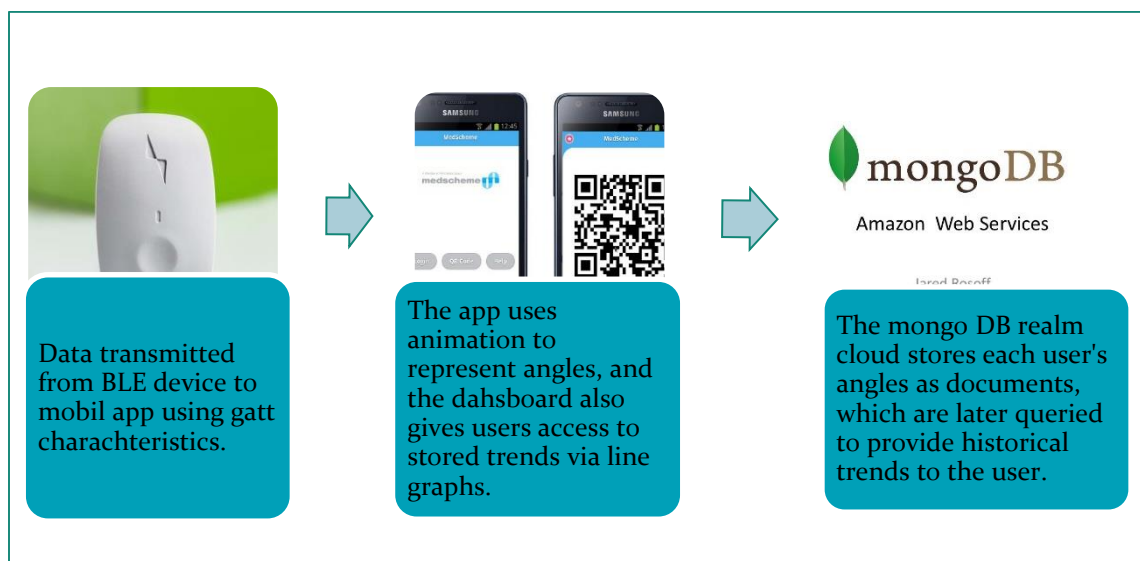


Figure 1: Flow of app working

TECHNOLOGY STACK:

In this section, the specifics of technical tools and libraries are discussed.

Android Studio:

Android app development involves two levels. We first create UI layouts and the screen elements they contain using the XML vocabulary of Android. Later, the on-click functionality is described in the Java files of the respective layout activities.

Mongo DB Realm:

Realm is an embedded, object-oriented database. Its SDK provide a secure backend that can sync data between devices, authenticate and manage users.

Firstly the project was created on cloud and app key and plugins were added in android studio's app level and project level gradle files. Also mongo DB realm sync configuration, app configuration, user and app credentials classes were imported in java files and there instances were created.

Realm Object Schema JAVA SDK:

The cluster created on cloud was partitioned in 3 classes namely **Users**, **Angles** and **Peak**.

Java classes were created by extending the realm object class to define the attributes and functions of each class. Figure 2a and 2b shows a document saved in mongo DB.

```
_id: ObjectId("623a98121289a33e84d1232b")
_partition: "My project"
BackwardAngle: "0.0"
ForwardAngle: "86.0"
LeftAngle: "1.0"
MonthYear: "03 2022"
RightAngle: "0.0"
date: "23/03/2022"
name: "dawood"
time: "08:45:26"
```

(a) Angle doc of a user saved in cloud

```
_id: ObjectId("623c0fa77cb4bb4a40623c7b")
_partition: "Users"
age: "20"
gender: "Male"
username: "Ashar"
```

(b) User credentials saved in cloud

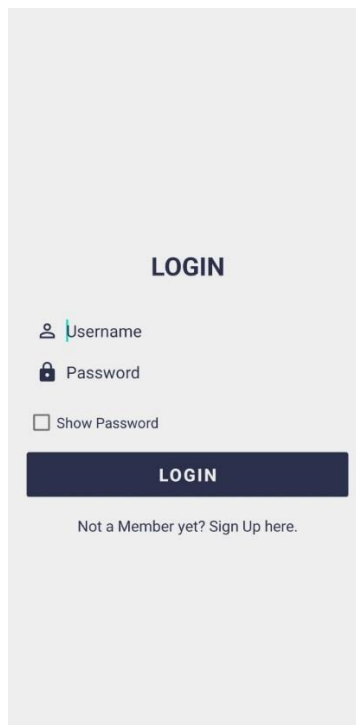
Figure 2: Mongo DB database view

Device Setup Screens:

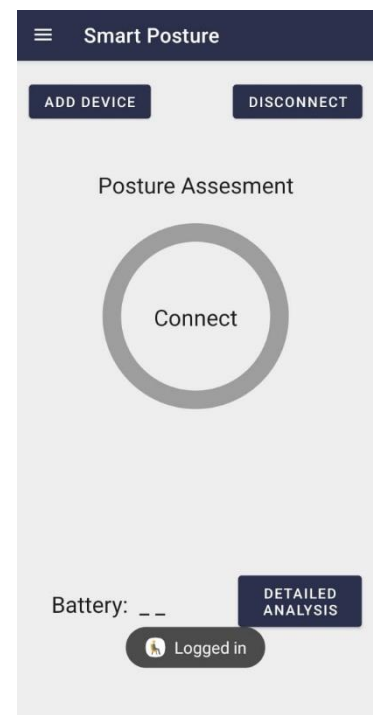
Figure 3 shows the initial screens of the app where Figure (3a) shows the first loading screen of the app when you open it. Then, you will be directed to login screen as shown in Figure (3b) where user can sign up if using the app for first time or they can directly sign in using their credentials. Next, is the main screen see Figure (3c) of the app where different options are available for interaction with the app. You will be prompted with the message of connect device on this screen as shown in Figure (3d) and then through add device button you will be taken on to scan device screen see Figure (3e) where you can scan the BLE device and finally pair the device named as "Neck Posture" as shown in Figure (3f).



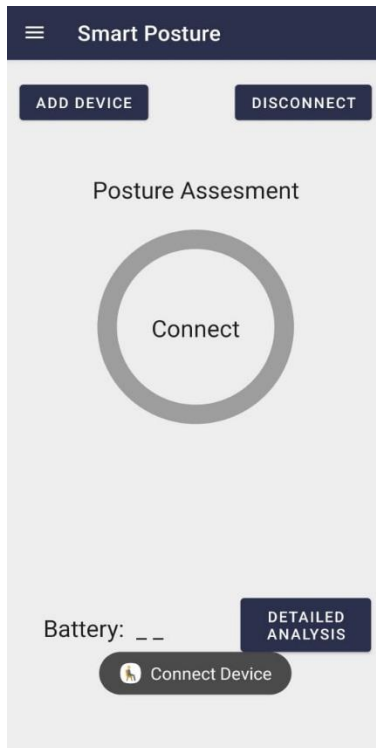
(a) Startup Screen



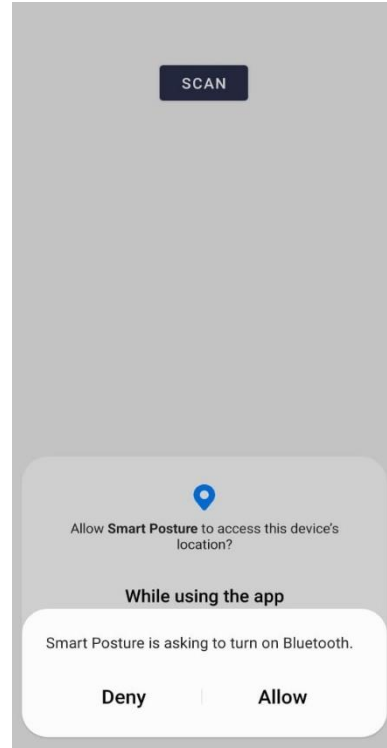
(b) Login Screen



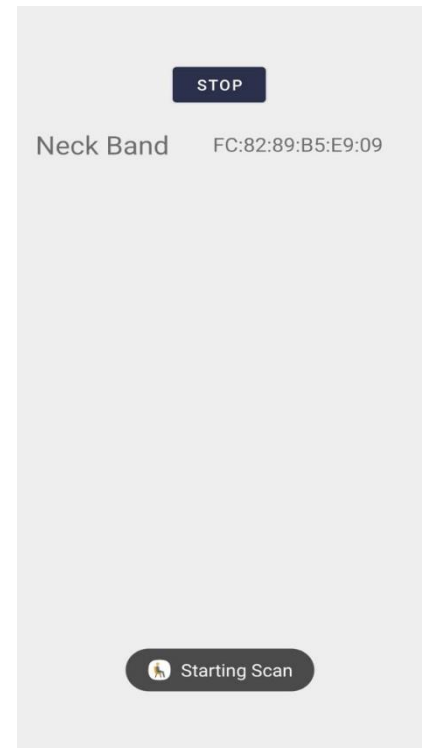
(c) Main Screen



(d) Connect Device Screen



(e) Scan Device Screen



(f) Pair Device Screen

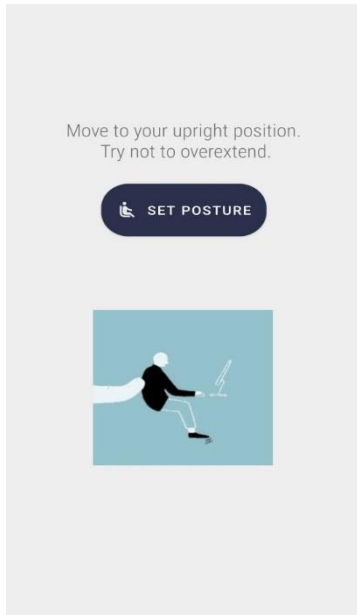
Figure 3: Device setup screens

Posture Assessment Screens:

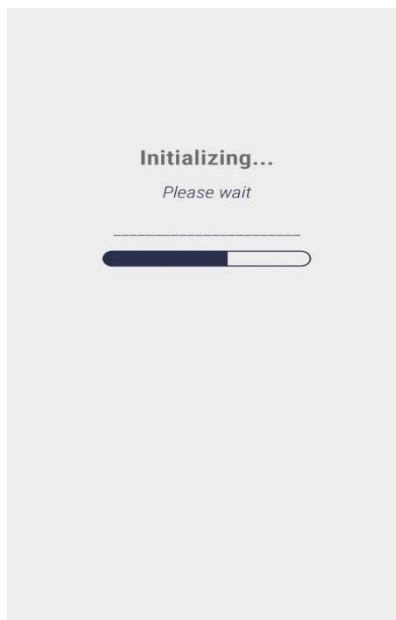
Figure 4 shows the posture assessment screens where all the calculations are done and results are displayed. You will be directed to set posture screen see Figure (4a) for device calibration after the pairing of the device. Then, an initialization screen is shown as in Figure (4b) showing the loading status after which you will be directed to main screen where posture assessment status is displayed in the circular progress bar see Figure (4c). The green color in progress bar indicates the correct posture where the user don't need to change their posture. If the angles exceed the set thresholds for correct posture they will be prompted with vibration feedback on device and "Critical" posture alert will be displayed on app with color yellow as shown in Figure (4d). Furthermore, if the posture angles continue to grow beyond the set threshold limits of critical posture then warning alert with color red is shown as in Figure (4e).

A detailed analysis button is provided in bottom right corner to view angles of forward, back ward and lateral (right and left) bending. The green color indication reflects the

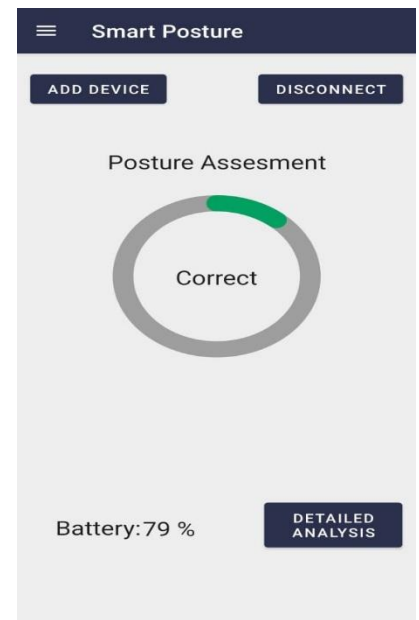
angle of correct posture whereas red color shows that angles are exceeded and that user needs to change their posture. Figure (4f), Figure (4g), Figure (4h), and Figure (4i) shows the angles when forward, backward, right and left bending postures were evaluated by the user.



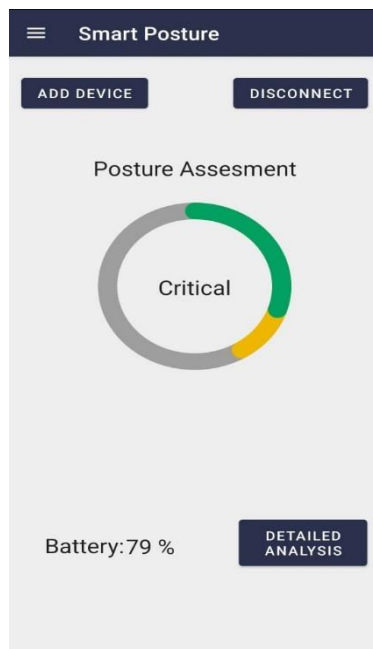
(a) Set Posture Screen



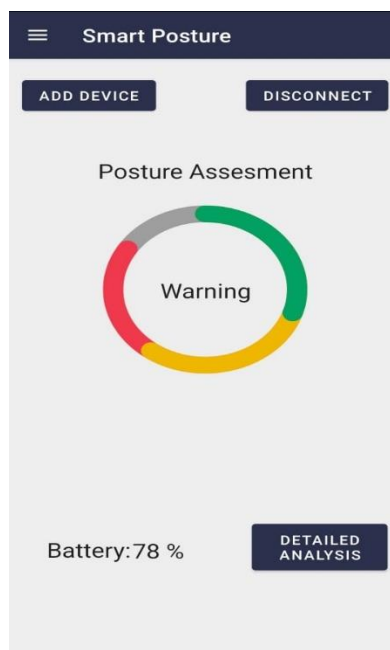
(b) Initialization Screen



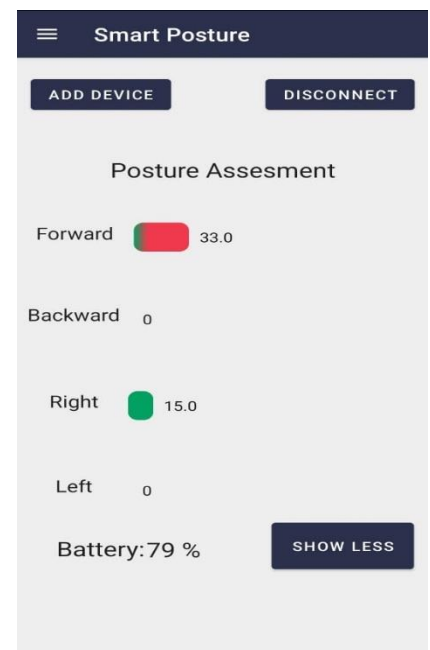
(c) Correct Posture



(d) Critical Posture



(e) Warning Posture



(f) Incorrect Posture

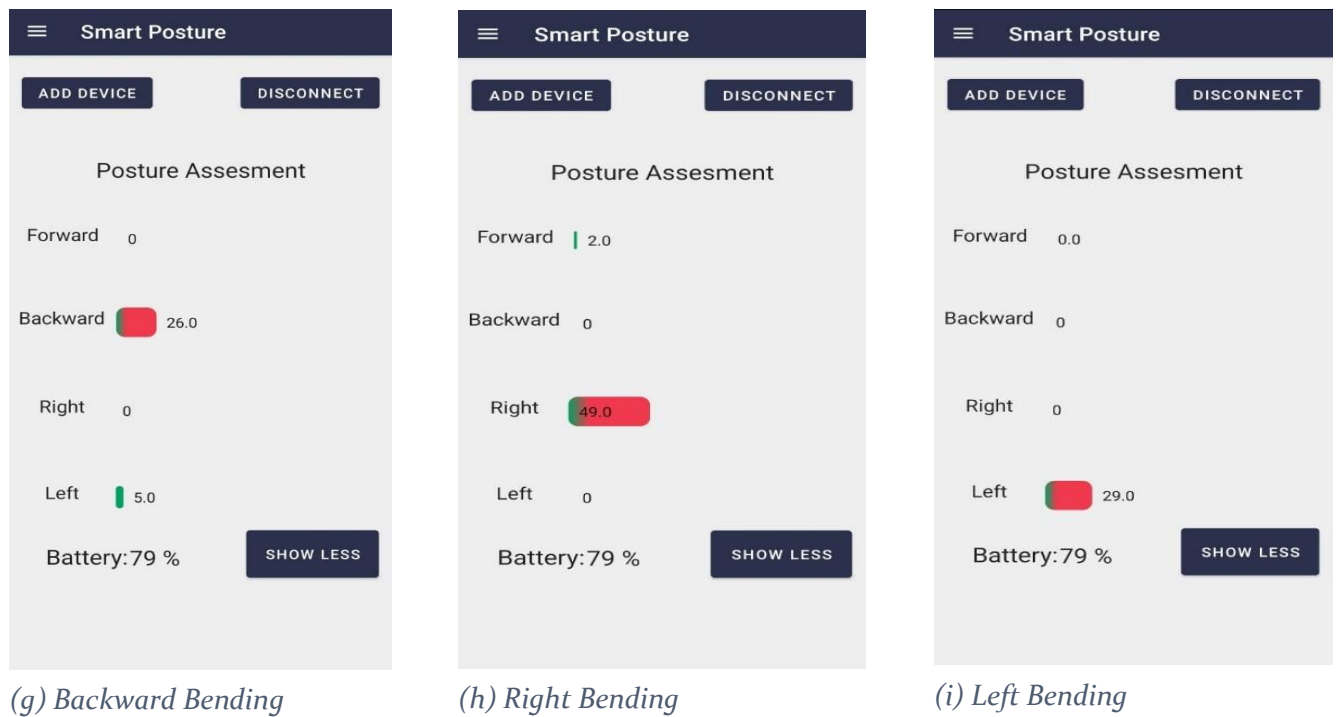


Figure 4: Posture Assessment Screens

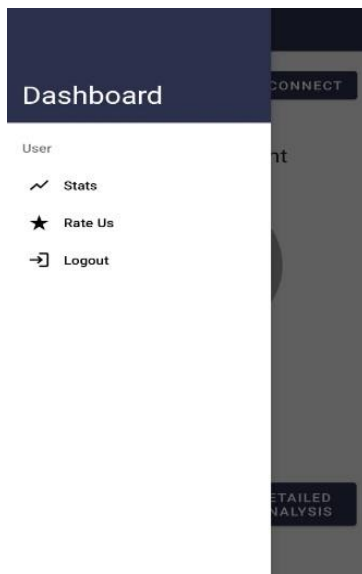
Table 1 summarizes the threshold limits and color indication for each movement respectively. The threshold limits are set in accordance with study that presented ranges of joint angle for healthy people and individual with low back pain.

Posture Alert	Movements	Ranges	Color codes
Correct Posture	Forward bending Backward bending Lateral bending	0° – 30°	Green
Incorrect Posture	Forward bending Backward bending Lateral bending	30° – 60°	Yellow
Warning Posture	Forward bending Backward bending Lateral bending	60° – 90°	Red

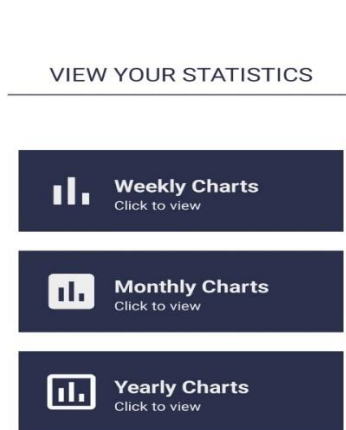
Table 1: Threshold limit of movement and color indication

Statistics Dashboard:

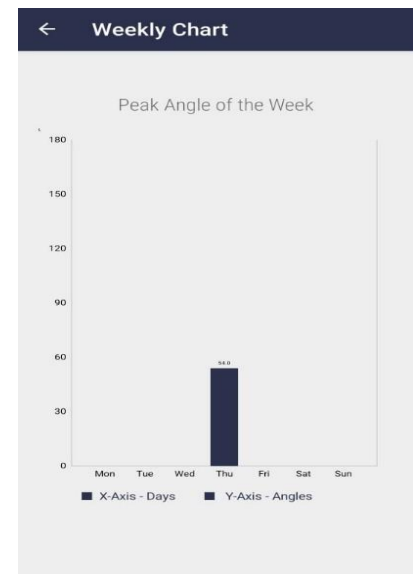
Figure 5 shows the dashboard screens which can be viewed through the top left button on the screen. It monitors the peak values and trends of the user behavior. The collected data is sent to the cloud for maintaining a record of all activities and provide summary of trends at the end of each activity. Figure (5a) shows the dashboard options, Figure (5b) shows the statistics screen for weekly, monthly and yearly charts. Figure (5c) shows the weekly chart screen with peak value of each day. Similarly, Figure (5d) and Figure (5e) shows the monthly and yearly charts respectively. Future work can be done for predictive analytics on the collected data to predict risk factors associated with the postures w.r.t maximum holding time.



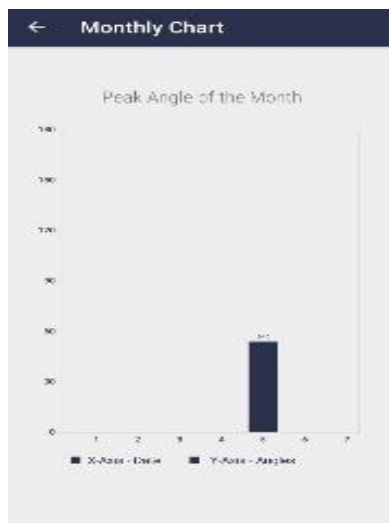
(a) Dashboard Screen



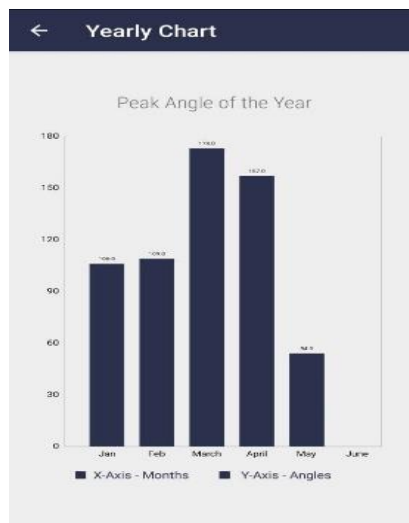
(b) Statistics Screen



(c) Weekly Chart Screen



(d) Monthly Chart Screen



(e) Yearly Chart Screen