Deliverable 4

2. Team Name
   Lifesaver Engineers

3. Project Name
   VitalTracker: Smart Location Tracking Device

4. Names and IDs
   Patrik Prenga n01428752, Michael Carinci n01480052, Jason Macdonald n01246828, Nicholas Rafuse n01440073

5. Table of Contents

6. Project Scope:
Development of the "Vital Tracker" Android application, designed primarily to aid users in monitoring vital statistics, tracking location, and detecting potential falls. The application integrates these core functionalities seamlessly to provide users with peace of mind regarding their safety and well-being.

7 - 9. Member Info and Participation

NAME & GITHUB	ID	SIGNATURE	EFFORT
Patrik Prenga (PatrikPrenga8752)	n01428752	*Patrik Prenga*	100%
Michael Carinci (MichaelCarinci0052)	n01480052	*Michael Carinci*	100%
Jason Macdonald (Jasonmac644)	n01246828	*Jason Macdonald*	100%
Nicholas Rafuse (NicholasRafuse0073)	n01440073	*Nicholas Rafuse*	100%

10 & 11. Github Link

https://github.com/MichaelCarinci0052/LifesaverEngineersVitalTracker.git

12. Test Account
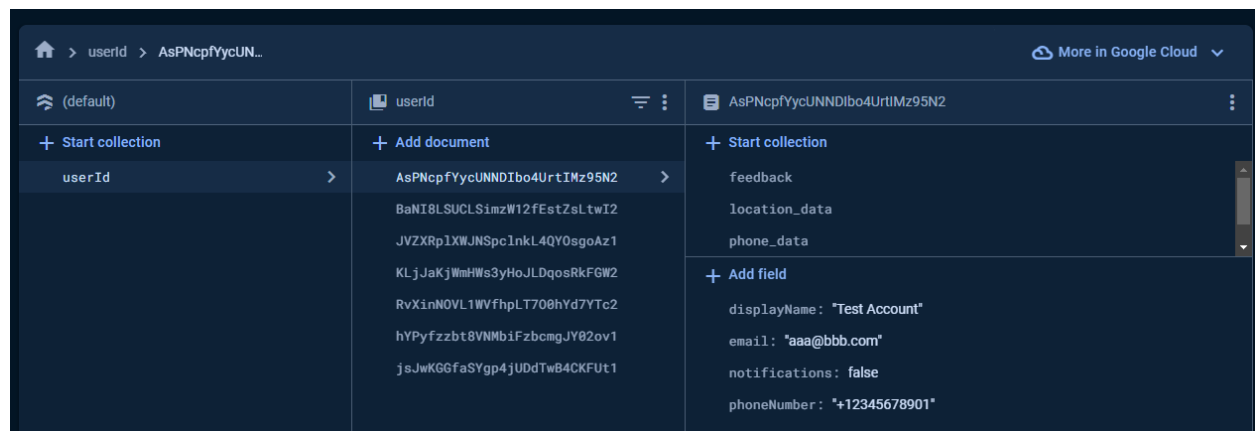
16. Sprint goals
- Complete full app functionality with zero bugs
- Redesign UI for Device, Configuration, and GPS screens
- Be able to completely display data values in the Vitals screen
- Complete test cases to document testing for multiple screens

- Implement offline functionality

17. Container Diagram

Customer
[Person]

The person that is using our product

Views account, vitals, device, and GPS info using

Mobile App
[Container: Android]

The base of the product

Makes API calls to

Database
[Container: Firebase]

Holds all the data containing login and registration, feedback, and data values

Reads from and writes to

API
[Container: Java in Android API]

Provides the mobile app with functionality

Sends e-mail using

Email System
[Software System]

Sends account verification or data exports through the email system

18. Component Diagram

Customer
[Person]

A customer of the product, has an account.

Notifications
[Software System]

Pushes notifications to user's device

Sends notification to

Can view account information, vitals, GPS , device

Mobile App
[Container: Java]

Provides all the needs to use the product. Can view data information all through the customer's mobile device.

Home
[Software System]

Users can view mini versions of

Displays

Customer can give a review using

Allows user to create

Feedback
[Software System]

Customer can send feedback to the company

Account
[Software System]

Allows user to view account information

Connected to

Displays

Clickable container

Displays

Clickable container

Clickable Container

Clickable container

Displays

Sends data to

Read and write to

Device
[Software System]

Views device information

GPS Location
[Software System]

Displays location

Vitals
[Software System]

Displays customer's vitals information

Database
[Container: Firebase Database]

Stores user's account information, GPS history, vitals, and feedback history.
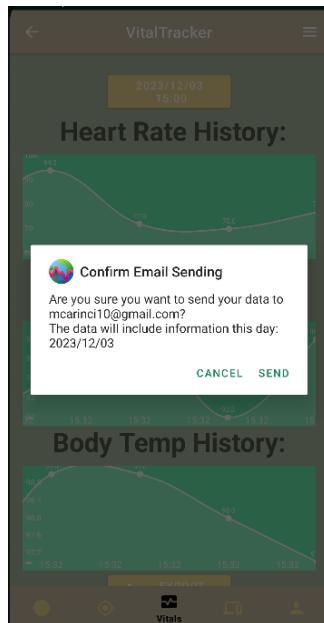
Sends data to

20. Demonstrating the use of features (AlertDialog, Snackbar, Toast, Notification)
AlertDialog:
This is one instance of an AlertDialog being used, when the user clicks on the "Export Data" button on the Vitals screen. It asks the user for confirmation to send data to their email.


Snackbar:
This is one instance of how we use a snackbar in our app. It appears on the GPS screen, and it activates based on if the user allowed GPS permissions. On first open, it will say "Location permission denied". After allowing permission, a new snackbar will appear saying "Permission Granted"
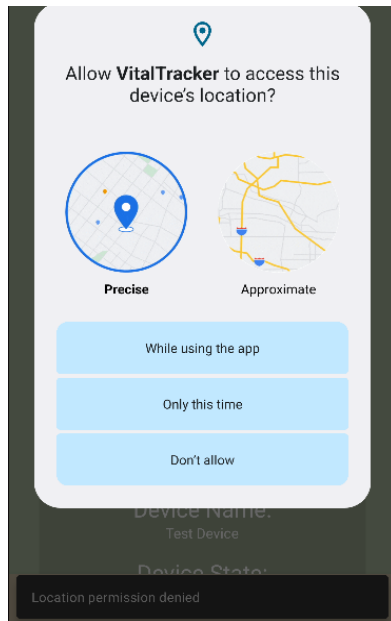
Toast:

This is one instance of the toast being used in our app. This appears when the user clicks the "Turn On / Off" button on the Device Screen. It tells the user whether or not the device was turned on or off.
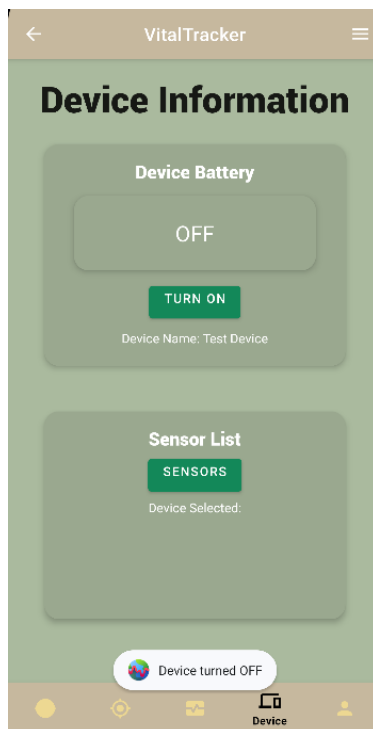
Notification:
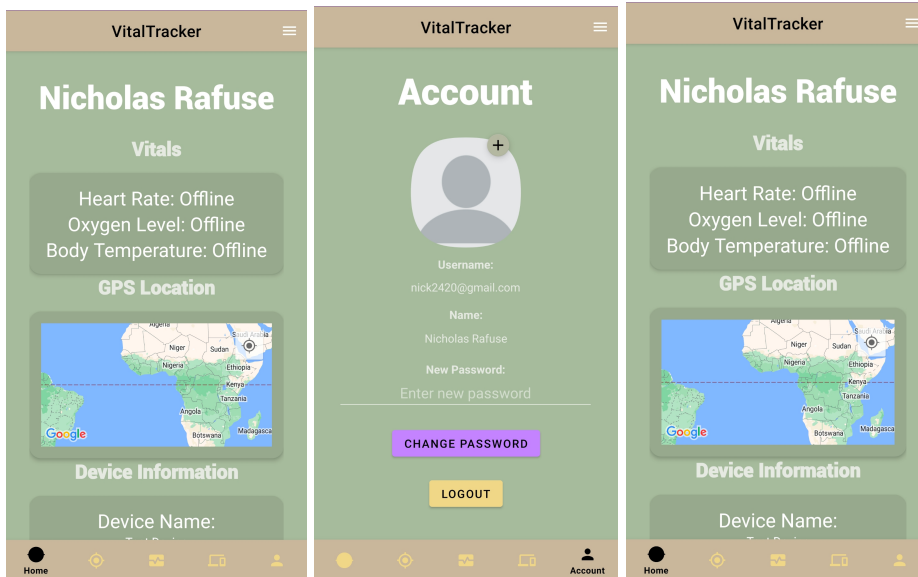
This is how our app uses notifications. It uses the vitals data, and if it detects an abnormal body temperature, it will send a notification to tell the user. This is important for if the user is currently not using the app. There is also a setting provided that allows the user to disable notifications.
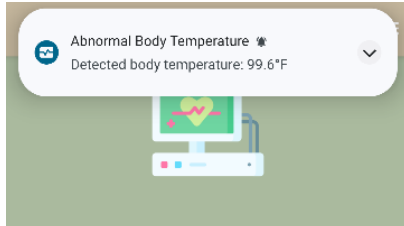
21. Offline mode functionality

23. Scrum Dashboard (showing only the new tasks)
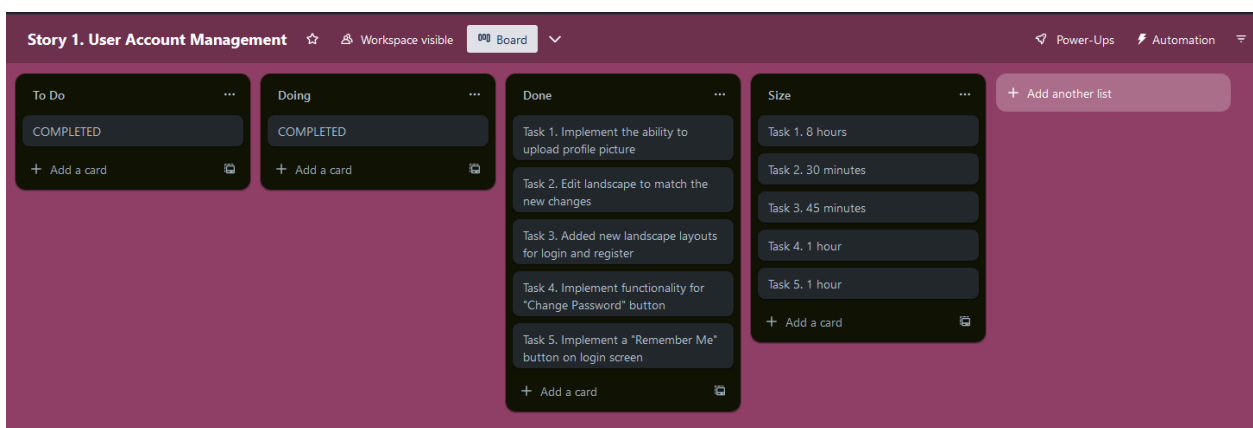
Story 2. Vitals Monitoring
☆ · 👤 Workspace visible · 🔲 Board ⌄   ◁ Power-Ups ⚡ Automation ☰

To Do
Implement and connect sensor readings (Future)
+ Add a card

Doing
COMPLETED
+ Add a card

Done
Task 1. Add new landscape mode
Task 2. Create graphs to display values and time
Task 3. Implement date picker
Task 4. Add a button to allow exportation of data - to user's email
+ Add a card

Size
Task 1. 30 minutes
Task 2. 1 day
Task 3. 2 hours
Task 4. 4 hours
+ Add a card

+ Add another list

Story 3. Device Mangement
☆ · 👤 Workspace visible · 🔲 Board ⌄   ◁ Power-Ups ⚡ Automation

To Do
Accurately display device information from sensors (Future)
+ Add a card

Doing
COMPLETED
+ Add a card

Done
Task 1. Redesign UI of Device screen
Task 2. Add text to display current device name an display whether it is on or not
Task 3. Allow user to click on a sensor and display the chosen option
Task 4. Sync up the device state text with the home screen to show the same response
Task 5. Create new landscape layout
+ Add a card

Size
Task 1. 3 hours
Task 2. 30 minutes
Task 3. 30 minutes
Task 4. 2 hours
Task 5. 30 minutes
+ Add a card

+ Add another list

Story 4. Location Tracking
☆ · 👤 Workspace visible · 🔲 Board ⌄   ◁ Power-Ups ⚡ Automation ☰ Filters

To Do
Accurately show location of physical device (Future)
+ Add a card

Doing
COMPLETED
+ Add a card

Done
Task 1. Redesign GPS screen
Task 2. Store GPS data in the updated history screen
+ Add a card

Size
Task 1. 1 hour 30 minutes
Task 2. 1 hour
+ Add a card

+ Add another list

Story 5. App Settings Management
☆ · 👤 Workspace visible · 🔲 Board ⌄   ◁ Power-Ups ⚡ Automation ☰ Filters

To Do
COMPLETED
+ Add a card

Doing
COMPLETED
+ Add a card

Done
Task 1. Implement functionality for the "Restore Default" button
Task 2. Add lanscape layout for Settings
Task 3. Redesign Configurations screen to have a container background
Task 4. Add new landscape layout for Configurations
+ Add a card

Size
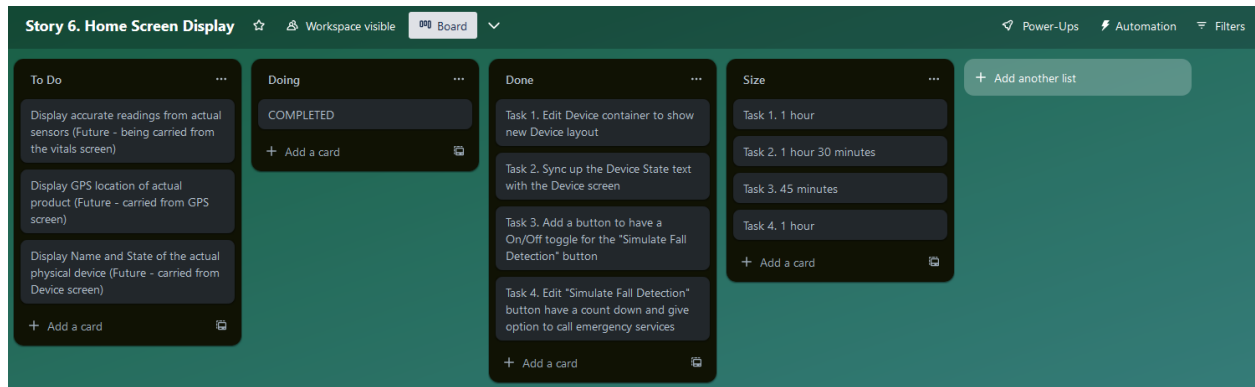Task 1. 1 hour
Task 2. 30 minutes
Task 3. 45 minutes
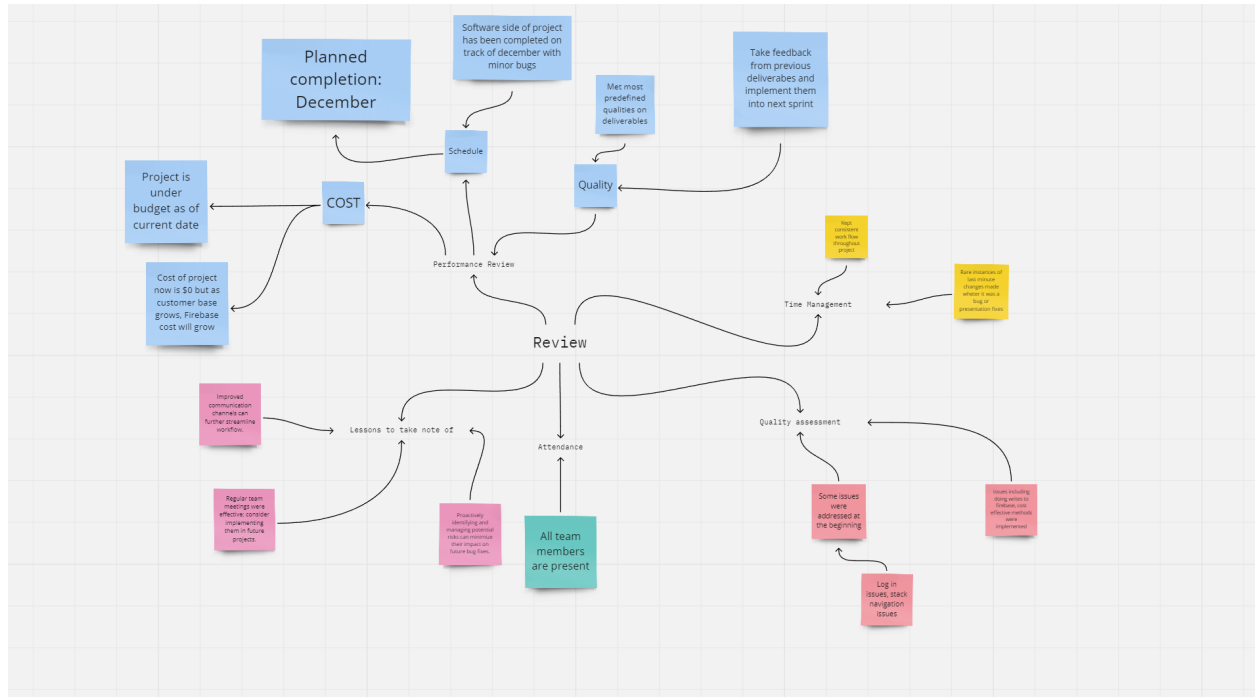Task 4. 30 minutes
+ Add a card

+ Add another list

24. Post-Mortem, Project Review Meeting

Project Review Meeting (use a tool to record)
https://miro.com/welcomeonboard/d1BrNTBuSUNMN0RaYW9ybFRDbWM2QWRpM3RYcXhGa
Glydnh6cDZwajhaZ21TbEdxQ1FZdldnMlpaRTdlaDJxb3wzNDU4NzY0NTYzNzg3NjU1NzAxfDI
=?share_link_id=521590522966

26. How did we address technical debt

We addressed technical debt in our app through multiple different ways:
- Removing repeated code and creating singular functions that can be called whenever we use it (using design principles to produce better quality)
- Creating test cases to ensure effectiveness and to debug where necessary
- Catching exceptions (Example, when the user logs in or registers their account, providing feedback and returning an error if unsuccessful)
- Code refactoring (expanded in next section)

27. Two areas of refactoring and why we did it

Area 1. One of the areas we refactored was the Device screen. This was done to provide a much better visual experience along with keeping the original code. We accomplished this by separating the two initial buttons into two containers, that way each can have its own separate functionality. We also added additional code onto it to enhance implementation, such as syncing the text between multiple screens and displaying current options with the Sensor List and Device Name.

Code for creating the two containers in the xml file:

```xml
<?xml version="1.0" encoding="utf-8"?>
<FrameLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".DeviceFragment"
    android:background="@color/ashGreen">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:layout_gravity="center_horizontal|top"
        android:gravity="center_horizontal"
        android:orientation="vertical"
        android:paddingTop="16dp">

        <TextView
            android:id="@+id/userAccountName"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginBottom="24dp"
            android:fontFamily="@font/roboto_black"
            android:text="@string/deviceinfo"
            android:textAppearance="@style/TextAppearance.AppCompat.Large"
            android:textSize="40sp" />
```

```xml
    <FrameLayout
        android:id="@+id/batterycontainer"
        android:layout_width="331dp"
        android:layout_height="243dp"
        android:layout_marginTop="0dp"
        android:layout_marginBottom="-50dp"
        android:background="@drawable/container_background"
        android:elevation="4dp"
        android:padding="16dp">

        <TextView
            android:id="@+id/devices"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_gravity="center_horizontal|top"
            android:layout_marginBottom="24dp"
            android:fontFamily="@font/roboto_black"
            android:gravity="center_horizontal"
            android:text="Device Battery"
            android:textAppearance="@style/TextAppearance.AppCompat.Body1"
            android:textColor="@color/white"
            android:textSize="20sp" />

        <FrameLayout
            android:id="@+id/batterycontainer2"
            android:layout_width="266dp"
            android:layout_height="wrap_content"
            android:layout_gravity="center_horizontal|top"
            android:layout_marginTop="40dp"
            android:background="@drawable/container_background"
            android:elevation="4dp"
            android:gravity="center_horizontal"
            android:padding="25dp">

            <LinearLayout
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:gravity="center_horizontal"
                android:orientation="vertical">

                <TextView
                    android:id="@+id/batteryLifeTextView"
                    android:layout_width="wrap_content"
                    android:layout_height="wrap_content"
                    android:text="OFF"

android:textAppearance="@style/TextAppearance.AppCompat.Body1"
                    android:textColor="@color/white"
                    android:textSize="24sp" />
```

```xml
                </LinearLayout>

        </FrameLayout>

        <Button
            android:id="@+id/toggleDeviceButton"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_gravity="center_horizontal|bottom"
            android:layout_marginTop="16dp"
            android:layout_marginBottom="30dp"
            android:gravity="center_horizontal"
            android:text="Turn On" />

        <TextView
            android:id="@+id/devicename"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_gravity="center_horizontal|bottom"
            android:layout_marginBottom="5dp"
            android:gravity="center_horizontal"
            android:text="Device Name: Test Device"
            android:textAppearance="@style/TextAppearance.AppCompat.Body1"
            android:textColor="@color/white" />

    </FrameLayout>

    <FrameLayout
        android:id="@+id/sensorcontainer"
        android:layout_width="331dp"
        android:layout_height="228dp"
        android:layout_marginTop="100dp"
        android:background="@drawable/container_background"
        android:elevation="4dp"
        android:padding="16dp">

        <TextView
            android:id="@+id/sensorlist"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_gravity="center_horizontal|top"
            android:fontFamily="@font/roboto_black"
            android:gravity="center_horizontal"
            android:text="Sensor List"
            android:textAppearance="@style/TextAppearance.AppCompat.Body1"
            android:textColor="@color/white"
            android:textSize="20sp" />
```

```xml
        <Button
            android:id="@+id/buttonSensors"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_gravity="center_horizontal|top"
            android:layout_marginTop="30dp"
            android:gravity="center_horizontal"
            android:text="Sensors" />

        <TextView
            android:id="@+id/deviceselected"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_gravity="center_horizontal|top"
            android:layout_marginTop="80dp"
            android:gravity="center_horizontal"
            android:textColor="@color/white"
            android:text="Device Selected:" />


    </FrameLayout>

  </LinearLayout>

</FrameLayout>
```

Area 2. Another area that we refactored was the Configuration screen, as similarly to the Device screen, we implemented a new container with a background for the configuration screen to sit in. This was done for enhanced visibility as it was a little hard to see and read on the default green background.

Code for the new refactored Configuration screen:

```xml
<?xml version="1.0" encoding="utf-8"?>
<FrameLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="@color/ashGreen"
    tools:context=".FeedbackFragment">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="300dp"
        android:layout_margin="8dp"
```

```xml
        android:layout_marginTop="4dp"
        android:background="@drawable/rounded_background"
        android:padding="16dp"
        app:layout_constraintTop_toTopOf="parent">

    <TableLayout
        android:layout_width="match_parent"
        android:layout_height="290dp">

        <!-- Title Row -->
        <TableRow>

            <TextView
                android:id="@+id/textView6"
                android:layout_width="wrap_content"
                android:layout_height="25dp"
                android:text="@string/configurations"
                android:textSize="18sp"
                android:textStyle="bold" />
        </TableRow>

        <!-- Header Row -->
        <TableRow>

            <TextView
                android:layout_width="0dp"
                android:layout_weight="2"
                android:text="@string/feature"
                android:textStyle="bold" />

            <TextView
                android:layout_width="0dp"
                android:layout_weight="3"
                android:text="@string/description"
                android:textStyle="bold" />

            <TextView
                android:layout_width="0dp"
                android:layout_weight="2"
                android:text="@string/value"
                android:textStyle="bold" />
        </TableRow>

        <!-- Additional Spacing Row -->
        <TableRow>

            <Space
                android:layout_width="match_parent"
                android:layout_height="8dp" />
```

```xml
        </TableRow>

        <!-- Row 1: Home Location -->
        <TableRow>

            <TextView
                android:layout_width="0dp"
                android:layout_weight="2"
                android:text="@string/location2" />

            <TextView
                android:layout_width="0dp"
                android:layout_weight="3"
                android:text="@string/set_home_location" />

            <TextView
                android:layout_width="0dp"
                android:layout_weight="2"
                android:text="@string/user_input" />
        </TableRow>

        <!-- Additional Spacing Row -->
        <TableRow>

            <Space
                android:layout_width="match_parent"
                android:layout_height="8dp" />
        </TableRow>

        <!-- Row 2: Lock Orientation -->
        <TableRow>

            <TextView
                android:layout_width="0dp"
                android:layout_weight="2"
                android:text="@string/lock_orientation" />

            <TextView
                android:layout_width="0dp"
                android:layout_weight="3"
                android:text="@string/prevent_screen_rotation" />

            <TextView
                android:layout_width="0dp"
                android:layout_weight="2"
                android:text="@string/on_off" />
        </TableRow>

        <!-- Additional Spacing Row -->
```

```xml
        <TableRow>

            <Space
                android:layout_width="match_parent"
                android:layout_height="8dp" />
        </TableRow>

        <!-- Row 3: Notification Toggle -->
        <TableRow>

            <TextView
                android:layout_width="0dp"
                android:layout_weight="2"
                android:text="@string/notification_toggle" />

            <TextView
                android:layout_width="0dp"
                android:layout_weight="3"
                android:text="@string/handle_notifications" />

            <TextView
                android:layout_width="0dp"
                android:layout_weight="2"
                android:text="@string/on_off" />
        </TableRow>

        <!-- Additional Spacing Row -->
        <TableRow>

            <Space
                android:layout_width="match_parent"
                android:layout_height="8dp" />
        </TableRow>

        <!-- Row 4: Restore Default -->
        <TableRow>

            <TextView
                android:layout_width="0dp"
                android:layout_weight="2"
                android:text="@string/restore" />

            <TextView
                android:layout_width="0dp"
                android:layout_weight="3"
                android:text="@string/reset" />

            <TextView
                android:layout_width="0dp"
```

```xml
                    android:layout_weight="2"
                    android:text="@string/button_input" />
        </TableRow>

        <!-- Additional Spacing Row -->
        <TableRow>

            <Space
                android:layout_width="match_parent"
                android:layout_height="8dp" />
        </TableRow>

        <!-- Additional Spacing Row -->
        <TableRow>

            <Space
                android:layout_width="match_parent"
                android:layout_height="8dp" />
        </TableRow>

        <!-- Email Verification Status Row -->
        <TableRow>

            <TextView
                android:layout_width="0dp"
                android:layout_weight="2"
                android:text="Email Verification Status" />

            <TextView
                android:id="@+id/emailStatusTextView"
                android:layout_width="0dp"
                android:layout_weight="3"
                android:text="Verification success or unsuccessful" />

            <TextView
                android:layout_width="0dp"
                android:layout_weight="2"
                android:text="Toast" />
        </TableRow>

    </TableLayout>
  </LinearLayout>

</FrameLayout>
```

28. Suggestions to the instructor for future projects, things we liked, and things we would suggest to be done differently and how

Suggestions:
- Add in a requirement that a specific amount of daily stand ups have to be concluded
- Checkpoints for commits

Things we liked:
- Being able to design the app entirely ourselves
- Linking the app with the data values (Sensors in the future)
- Learning about the app life cycle

Things we would suggest to be done differently:
- Less focus on the diagrams, maybe only require 1 or 2 as opposed to all of them

29 - 30. Subfolder for PDF and powerpoint