

# **RaDoTech App Documentation**

## COMP3004

# Use Cases

---

**Use Case 1:** Unbox and check the device

**Use Case 2:** Download the App and create profile

**Use Case 3:** Connect the Device via Bluetooth

**Use Case 4:** Remove Jewelry

**Use Case 5:** Start a Measurement

**Use Case 6:** Scan Your Body

**Use Case 7:** Check results

## Use Case 1: Unbox and Check The Device

**Primary Actor:**

- User

**Scope:**

- RaDoTech

**Level:**

- User goal

**Stakeholders and Interests:**

- User: checks health

**Precondition:**

- User has purchased the RaDoTech gear

**Success Scenario:**

- Package contains all the required equipment and possesses no deficiencies or battery issues

**Main Success scenario:**

1. Package arrives at the door
2. User retrieves the package from the door
3. User opens the package
4. User checks that the package contains the health tracker, spray bottle and charging cable
5. User powers on the health tracker to confirm it is working
6. User checks the battery status of the health tracker

7. User checks that the gear is in pristine shape

**Extensions:**

- 1a. Package does not arrive
  - 1a1. User rechecks their order on the delivery site and calls the RaDoTechCompany for either reimbursement or another delivery free of charge
- 4a. RaDoTech package is missing a piece of gear
  - 4a1. User calls the RaDoTech company for information and a reimbursement or a free replacement
- 5a. Health tracker does not power on
  - 5a1. The user contacts RaDoTech for assistance
- 6a. Battery is low
  - 6a1. User charges the device using the charging cable
- 7a. RaDoTech gear is damaged
  - 7a1. User calls the RaDoTech company for information and a reimbursement or a free replacement

## **Use Case 2: Download The App and Create profile**

**Primary Actor:**

- User

**Scope:**

- RaDoTech app

**Level:**

- User goal

**Stakeholders and Interests:**

- User: wants to begin to use the RaDoTech health tracker and store their results

**Precondition:**

- RaDoTech package has arrived without issue
- All items provided for a successful test are functional

**Success Guarantee:**

- User is able to download the app and access it successfully

**Main Success Scenario:**

1. User downloads the RaDoTech app on their phone
2. User creates a new profile
3. User submits necessary login details

**Extensions:**

- 1a. User is unable to download the RaDoTech app on their phone
  - 1a1. User restarts their phone and tries again
- 2a. Profile creation fails
  - 2a1. User accesses the troubleshooting tips within the app for instructions on creating the profile

## **Use Case 3: Connect the Device via Bluetooth**

**Primary Actor:**

- User

**Scope:**

- RaDoTech Health tracker

**Level:**

- User goal

**Stakeholders and Interests:**

- User: wants to connect the RaDoTech device to the phone

**Precondition:**

- RaDoTech app installed and RaDoTech is functional

**Success Guarantee:**

- Device is successfully connected to the phone via bluetooth

**Main Success scenario:**

1. Open the app
2. Press the device's Bluetooth button
3. Confirm connection on the app

**Extensions:**

- 3a. Bluetooth is turned off on the phone
  - 3a1. App prompts the user to enable Bluetooth on their phone.
  - 3a2. User enables Bluetooth and retries the connection.
- 3b. Bluetooth permissions not granted to the app

- 3b1. App displays a message directing the user to enable permissions in the phone's settings.

## **Use Case 4: Remove Jewelry**

### **Primary Actor:**

- User

### **Scope:**

- User

### **Level:**

- User goal

### **Stakeholders and Interests:**

- User: does not want any metal to alter the test conducted with the RaDoTech

### **Precondition:**

- User has purchased the RaDoTech and has downloaded the app and verified the gear works

### **Success guarantee:**

- All jewelry is successfully removed from the users person

### **Main Success scenario:**

1. User identifies all jewelry on their person
2. User removes all jewelry from their person

### **Extensions:**

- 2a. User is unable to remove certain jewelry
  - 2b1. App provides a warning that results may be inaccurate if jewelry remains on the body.

## **Use Case 5: Start a Measurement**

### **Primary Actor:**

- User

### **Scope:**

- RaDoTech measurement

### **Level:**

- User goal

**Stakeholders and Interests:**

- User: wants to measure health

**Precondition:**

- RaDoTech gear works as intended
- App is connected to the RaDoTech health tracker
- User holds the device in an overhand grip

**Success Guarantee:**

- User is able to begin the measurement process

**Main success scenario:**

1. Click on the button in the RaDoTech that says 'Measure now'
2. Hold the RaDoTech health tracker device in an overhand grip
3. Hit the button 'close' that is shown on the app
4. Click the power button on the bottom of the RaDoTech health check machine to pair it
5. User uses the spray bottle to spray the meridian points that the user intends to scan

**Extensions:**

- 1a. 'Measure now' button does not work
  - 1a1. User refreshes the app and tries again
- 4a. Spray bottle does not contain any water
  - 4a1. User refills the spray bottle

## **Use Case 6: Scan the Body**

**Primary Actor:**

- User

**Scope:**

- RaDoTech measuring health

**Level:**

- User goal

**Stakeholders and interests:**

- User: wants to conduct a health scan of their body

**Precondition:**

- RaDoTech device is functional
- No issues with the app

- No issues with pairing

**Success Guarantee:**

- User follows the instructions shown on app in phone and scans the correct meridian points as indicated until the test is complete

**Main Success Scenario:**

1. User places the node at the top of RaDoTech health scanner at each of the meridian points indicated by the instructions on the app
  - a. User waits for a sound to beep, indicating the scan for that particular meridian point is complete
2. App shows a visual indicator for the next meridian point to scan and confirms the successful completion of each point by sound
3. User uses the pop-out node on the health tracker for easier access to meridian points on ankles or feet
4. User completes the test and puts down the scanner

**Extensions:**

- 1a. Device Node Fails to Detect a Meridian Point:
  - 1a1. App prompts the user to reposition the device node and retry.
  - 1a2. User adjusts the device node as per app instructions.
- 2a. User Misses a Meridian Point:
  - 2a1. App notifies the user and highlights the missed meridian point to retry.
  - 2b2. User returns to the missed point and scans again.
- 3a. Device battery depletes mid-scan:
  - 3a1. App alerts the user to recharge and prompts them to resume the scan after the device is powered on.

## **Use Case 7: Check Results**

**Primary actor:**

- User

**Scope:**

- RaDoTech app

**Level:**

- User goal

**Stakeholders and Interests:**

- User: wants to check the results from their RaDoTech results

**Precondition:**

- User has successfully conducted the tests utilizing the RaDoTech health tracker device

**Success guarantee:**

- User can see their results from the measurements taken by the RaDoTech device

**Main Success Scenario**

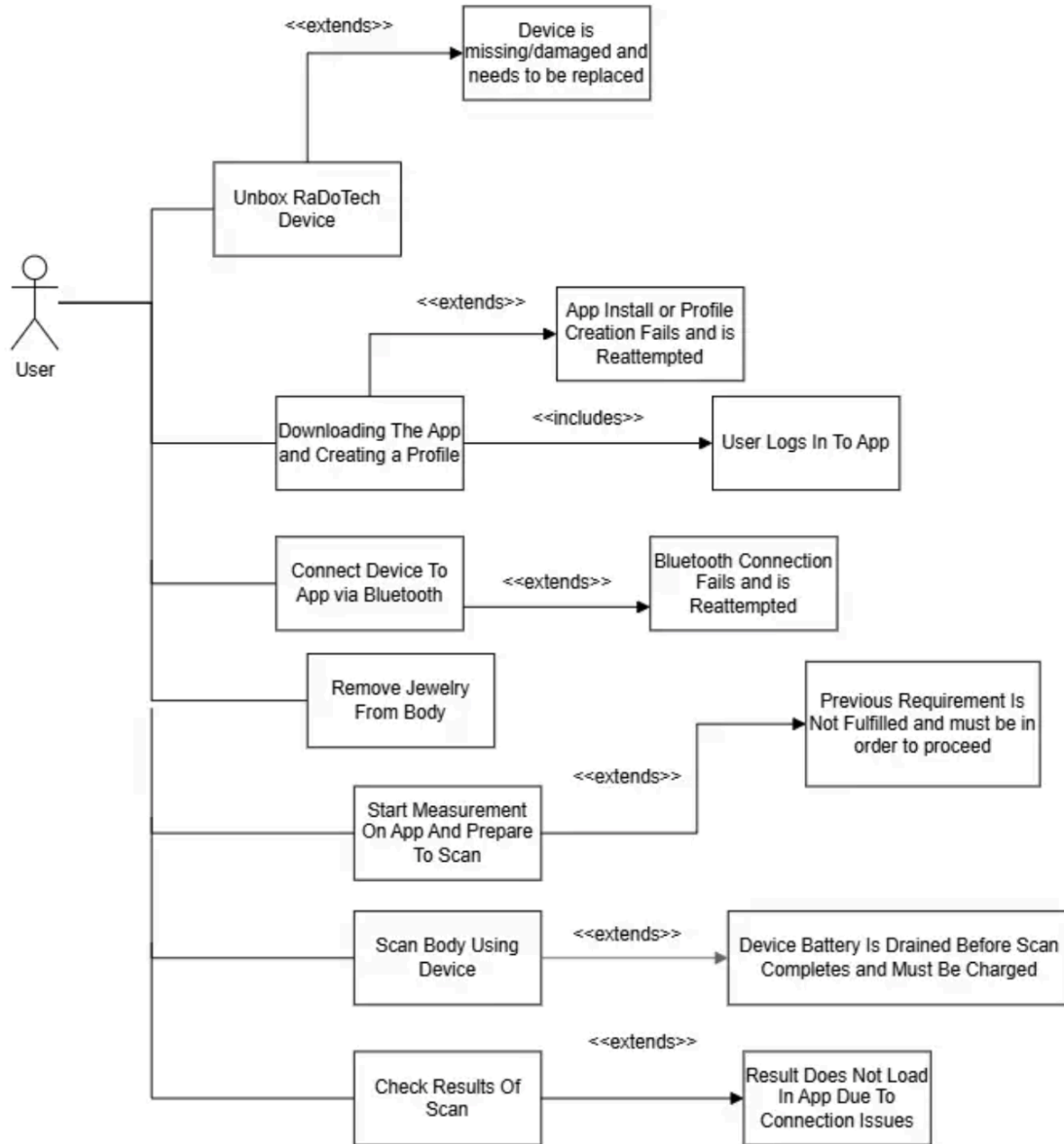
1. User selects the history option in the RaDoTech app.
2. User selects the latest test in the history section
3. User checks the results of the test, including the indicator option and organ level of health
4. User checks the recommendations to their health that is being shown by the app
- 5.

**Extensions:**

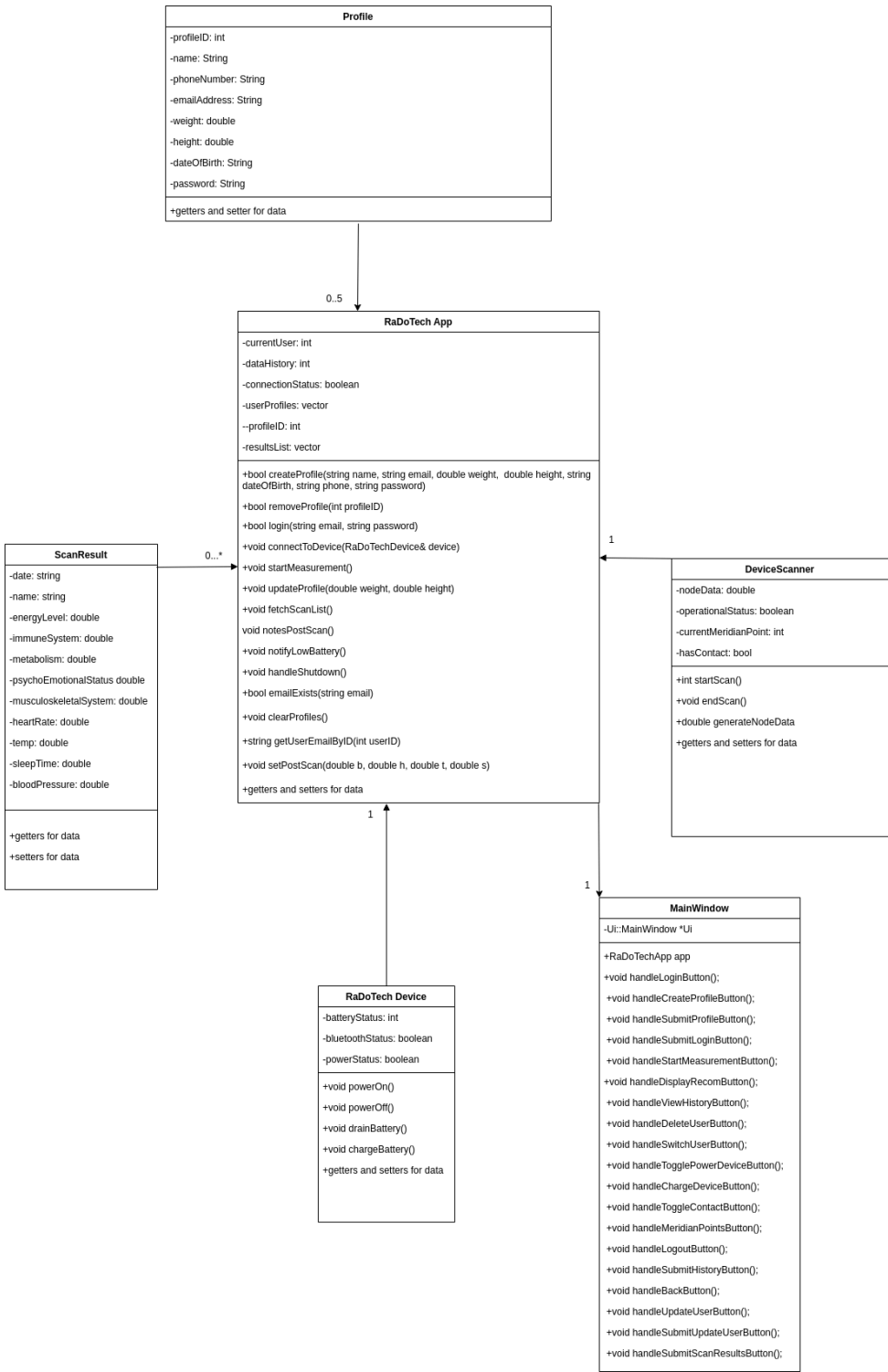
- 1a. User forgets to check the results immediately:
  - 1a1. App reminds the user to review the results through a notification.
  - 1a2. User accesses the history section at a later time.
- 2a. Latest test result do not show up in the history section
  - 2a1. User restarts the app
  - 2a2. User checks the history section of the RaDoTech app for their latest results
- 3a. Results do not load correctly:
  - 3a1. App displays an error message and offers a "Reload" button.
  - 3a2. User restarts the app to resolve the issue.



# Use Case Diagram



# UML Class Diagrams

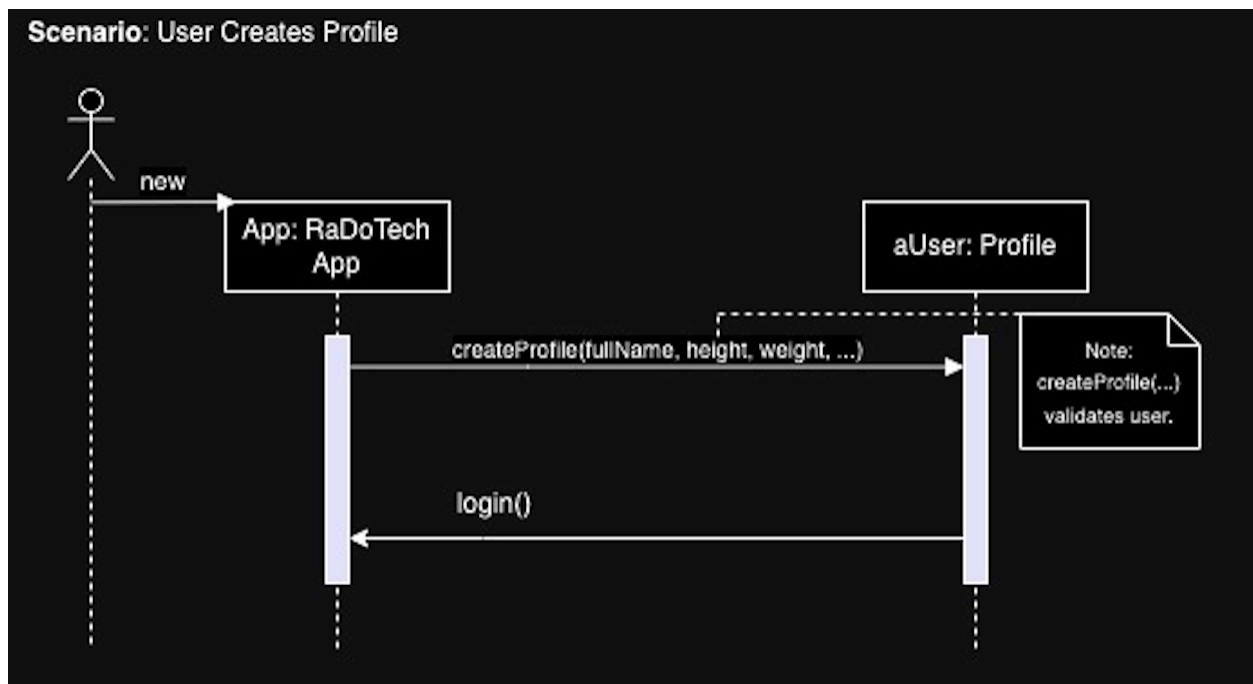


# UML Sequence Diagrams

## Normal Operations:

### Diagram 1: User Creates Profile:

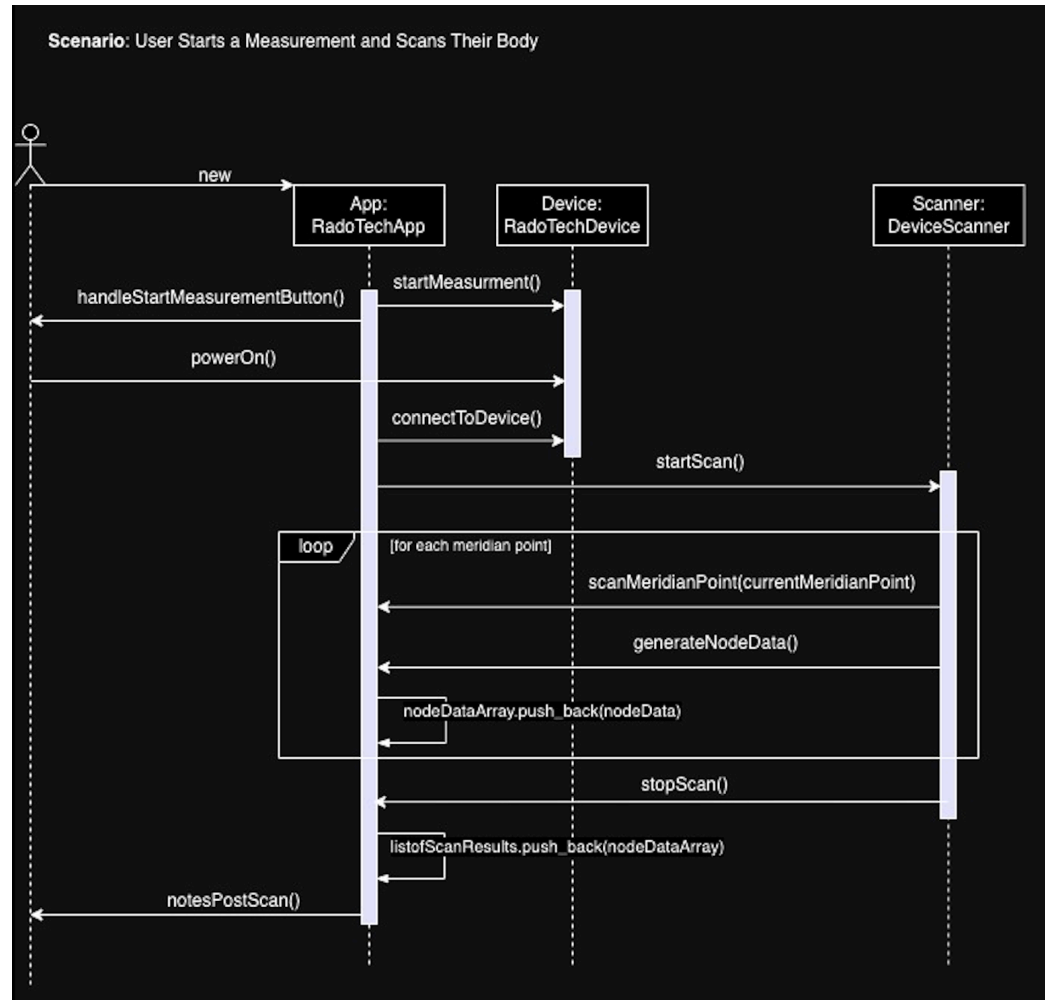
- User inputs login information (Full name, weight, height, DOB, phone, email, password)
- createProfile() in RaDoTech App validates the input
- App validates the profile creation request
- App saves the profile
- User successfully creates the profile



### Diagram 2: User Starts a Measurement and Scans Their Body

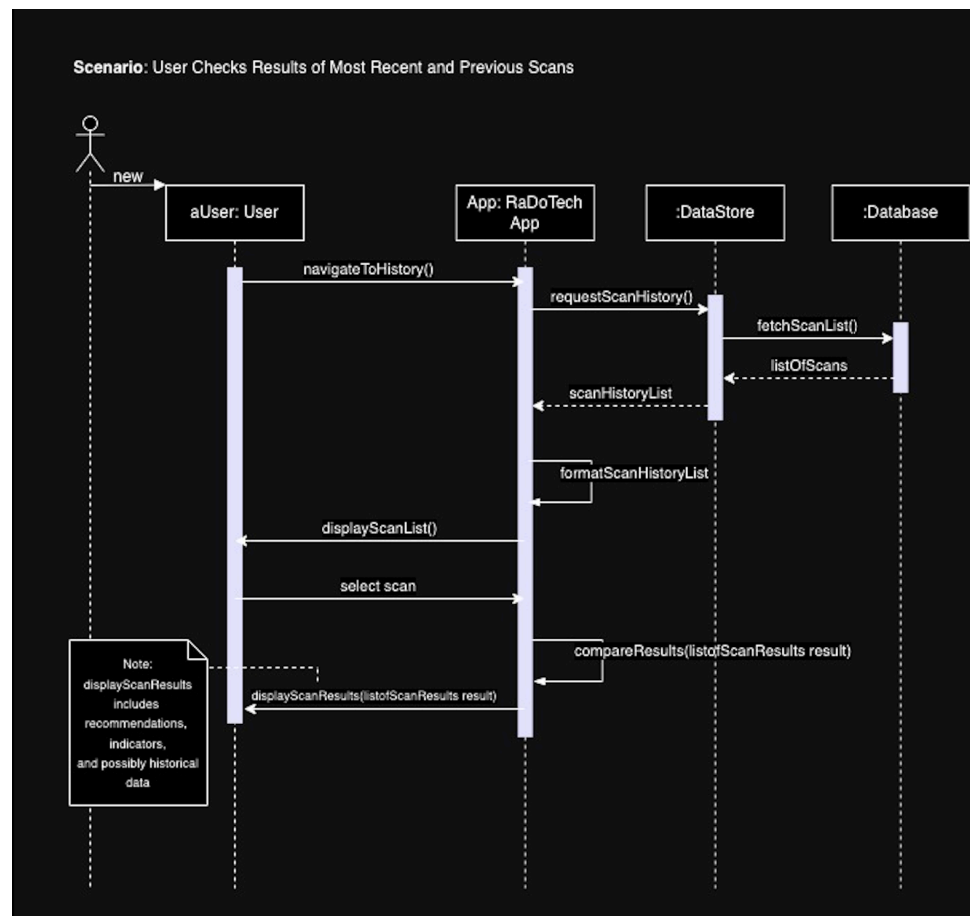
- User clicks the “Start Measurement” button to initiate on the app
- App sends a Start command to the (<MeasurementManager>)
- App sends instructions to user on how to hold the device
- App instructs user to utilize the spray bottle to spray all meridian points
- User clicks the pair button at the bottom of the RaDoTech device to pair it with the app on the phone
- RaDoTech device enters scanning mode and waits for meridian point scanning
- App prompts the user with instructions for scanning (e.g “Place the node on the indicated point”)
- User scans the first meridian point with the device

- App receives and logs the data for the scanned point
- App directs user to the next meridian point
- (\*for loop\*) User continues scanning all required points
- App notifies the user when the scan is complete
- App processes the results



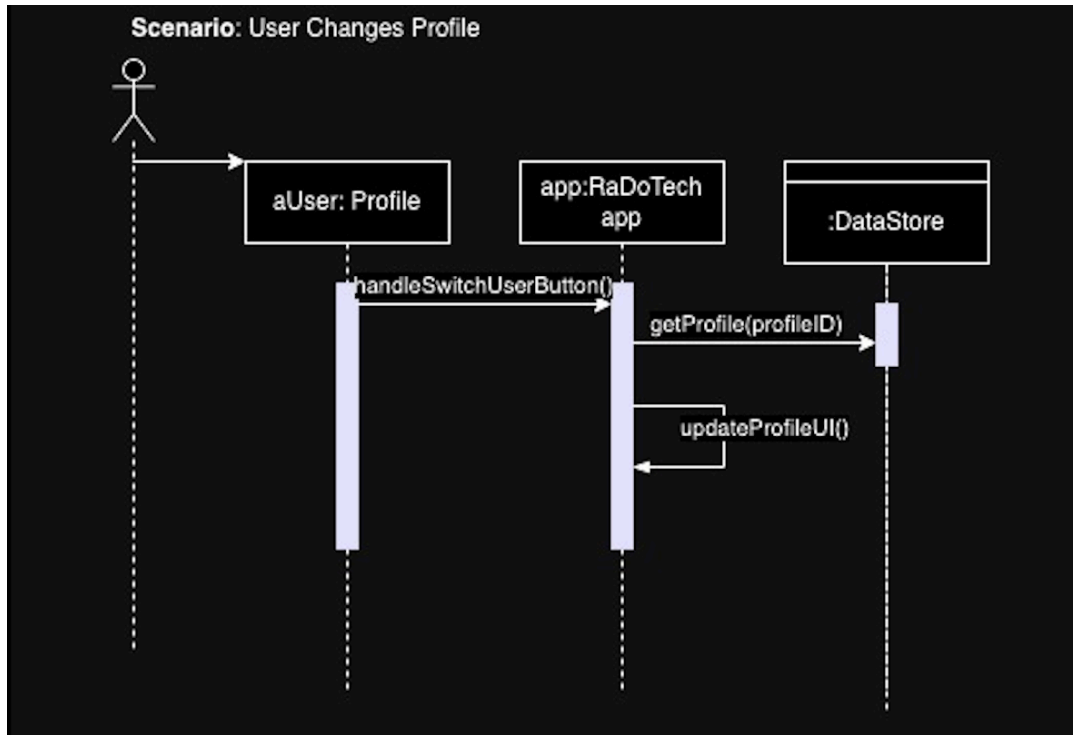
**Diagram 3: User Checks Results of Most Recent and Previous Scans**

- User navigates to the “History” section
- App retrieves stored scan data (UI sends a request to DataStore & database returns a list of scans (if any))
- App displays the list of scans (from most recent to oldest)
- User selects the desired scan
- App retrieves results for the selected scan (UI requests data from DataStore)
- App processes and formats the data
- App displays the scan results to the user
- User views the results



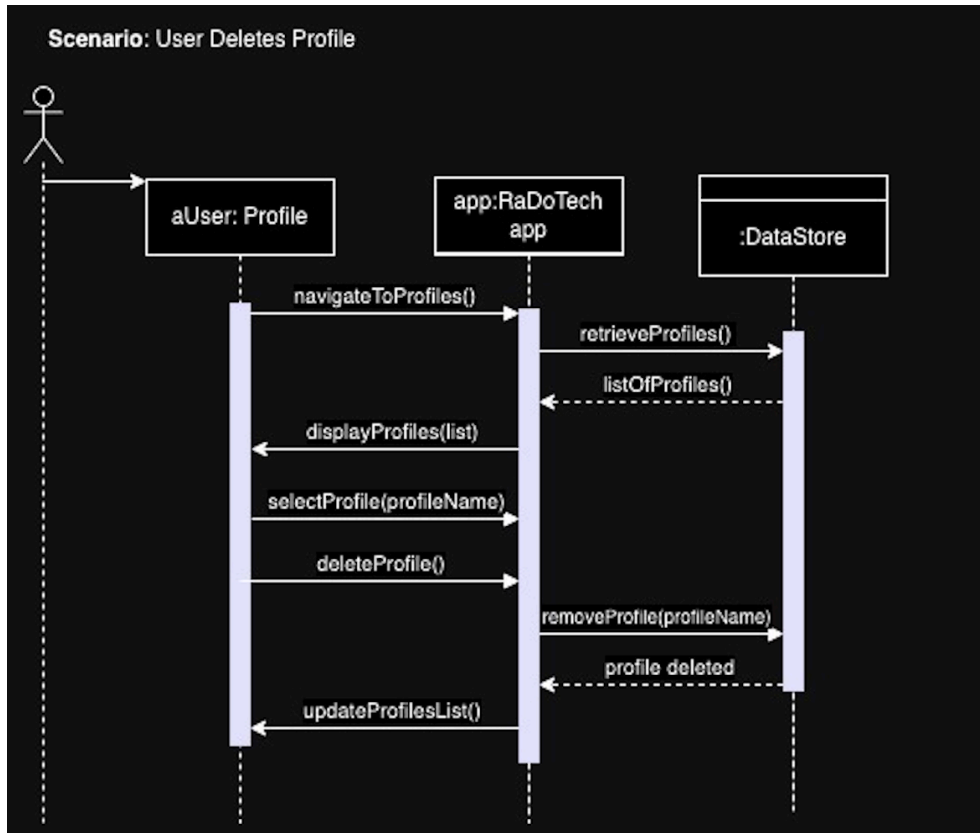
#### Diagram 4: User Change Profiles

- User navigates to the “Profiles” section
- The App retrieves the profiles data from DataStore
- User selects the profile to switch to
- The App updates the current active profile
- User confirms profile has been switched



### Diagram 5: User Delete Profile

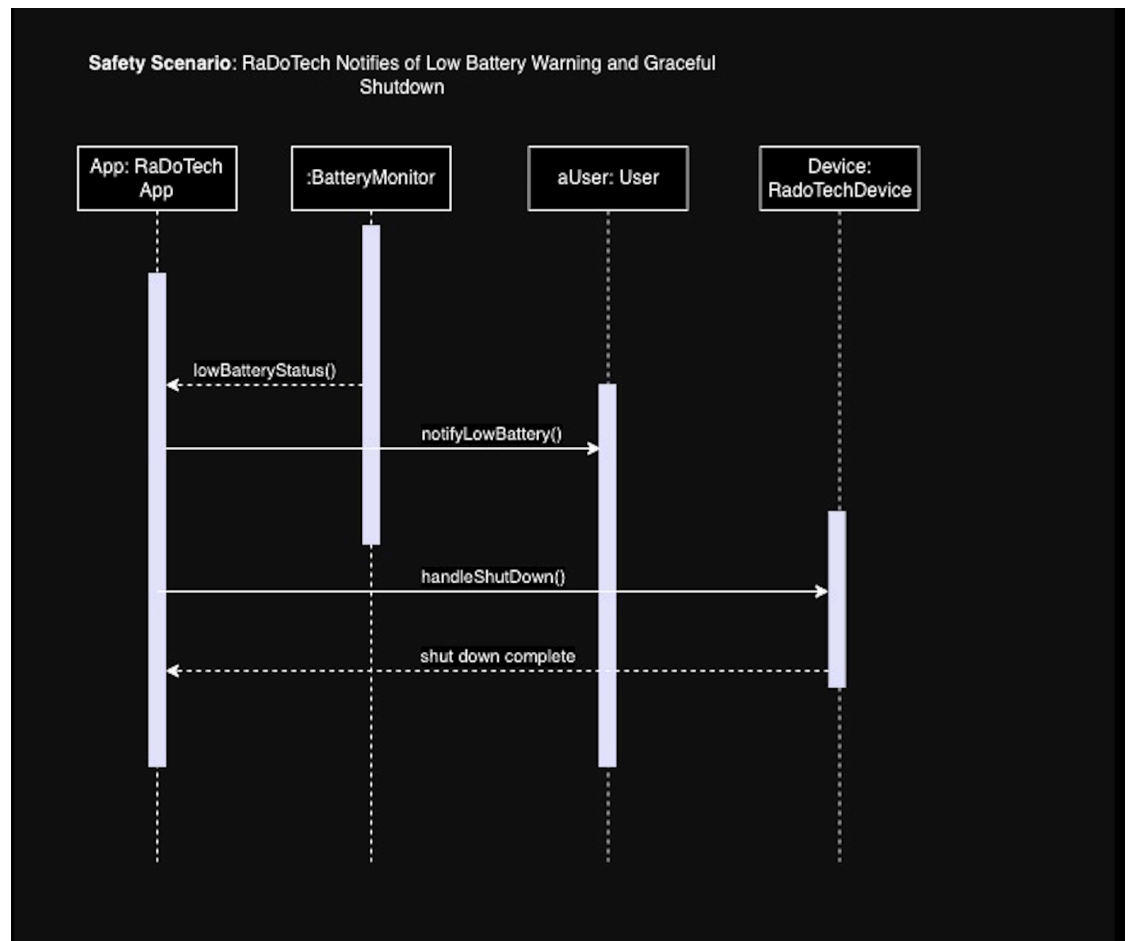
- User navigates to the “Profiles” section
- The App retrieves the profile data from DataStore
- The App displays the profiles
- User clicks and selects the profiles and selects the delete option
- The App processes the delete request with DataStore and successfully deletes the profile



### Safety Operation:

**Diagram 4: RaDoTech Notifies of Low Battery Warning and Gracefully Shutdown.**

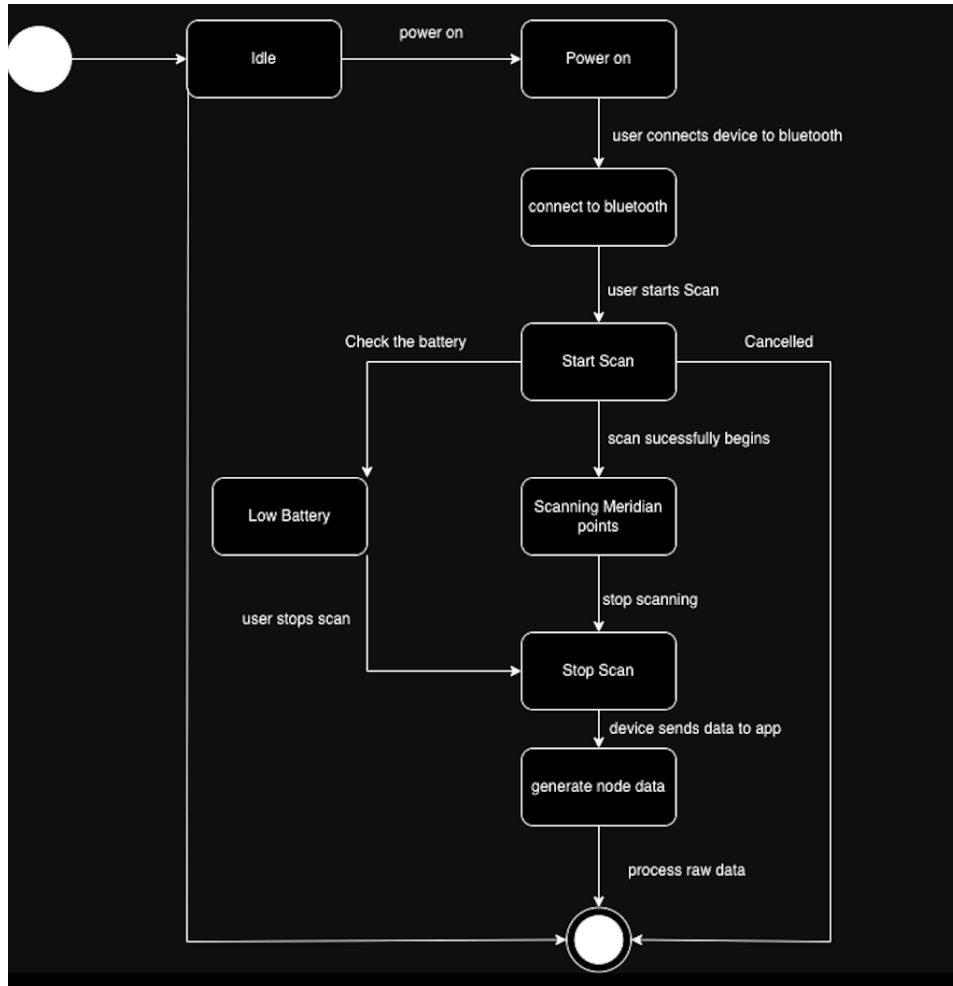
- (\*loop\*) App detects low battery status
- App receives low battery signal from battery monitor
- App informs the user of low battery warning
- App saves current data
- App receives signal for graceful shutdown from battery monitor
- RaDoTech device performs graceful shutdown
- App informs the user of RaDoTech shutdown



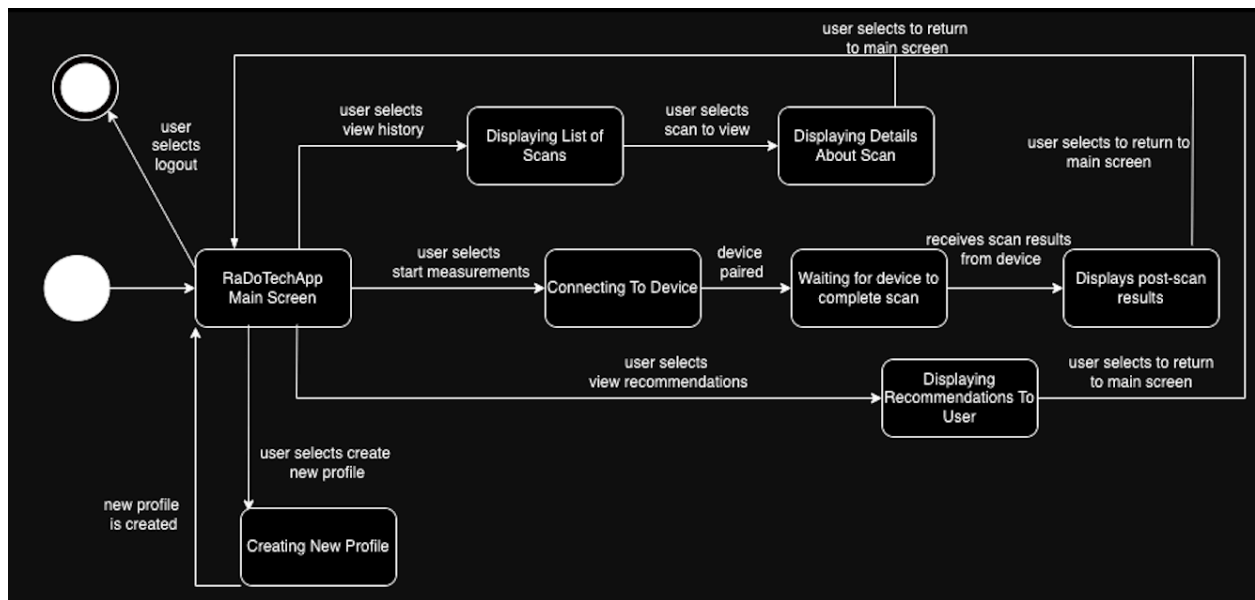


# UML State Machine Diagrams

Diagram 1: RaDoTechDevice State Machine Diagram:



**Diagram 2: RaDoTechApp State Machine Diagram:**



## Traceability Matrix

---

ID	Requirement	Related Use Case	Fulfilled by
0	Program is able to support up to 5 profiles created by users	Use Case 2: Download the app and create profile	MainWindow, RaDoTechApp, Profile
1	RaDoTechDevice connects to RaDoTechApp	Use Case 3: Connect the device via bluetooth	RaDoTechApp, RaDoTechDevice, MainWindow
2	User commences the scanning process via the app	Use Case 5: Start a measurement	MainWindow, RaDoTechApp
3	User is able to view the history of previous scans	Use Case 7: Check results	MainWindow, ScanResult, RaDoTechApp
4	User is able to view details of a specific scan	Use Case 7: Check results	MainWindow, ScanResult, RaDoTechApp
5	User is able to delete and update profiles	N/A	MainWindow, RaDoTechApp, Profile
6	User switches logged in profile	N/A	MainWindow, RaDoTechApp, Profile
7	User charges device after battery is drained	N/A	MainWindow, RaDoTechApp, RaDoTechDevice
8	User's skin is in contact with device while scanning	Use Case 5: Start a measurement	MainWindow, RaDoTechApp, RaDoTechDevice
9	User adds data to the scan result after the scan is complete	Use Case 6: Scan the body	MainWindow, RaDoTechApp
10	User views recommendations	Use Case 7: Check Results	MainWindow
11	User charges the device if the battery is too low	Use Case 1: Unbox and Check The Device	MainWindow, RaDoTechApp, RaDoTechDevice

# Explanation Of Design

---

In the program, we have the mainwindow which contains several slots and signals corresponding to various actions the user wishes to perform when using the program. The mainwindow contains an instance of the RaDoTechApp class, this class is how the mainwindow can interact with the rest of the classes in the the program, the RaDoTechApp contains a single instance of the DeviceScanner, and a vector of Profile type and of ScanResult type.

When the user creates a profile, the RaDoTechApp creates a new Profile object using the information taken in from the user, and stores it in the vector of Profiles, unless there are already 5 Profiles as that is the max. After that the user must turn on the device and ensure it is in contact with their skin before they can proceed to the scan. Once the scan starts and is completed the results are added to the ScanResult vector. The user can then view a list of all the scans that have been performed, and then get more details about a specific scan. After scanning multiple times, the user will need to charge the battery of the device to keep scanning.

In this project we made use of the observer pattern with the mainwindow class as the observer in order to have the program wait for the user to click a button, and then call a function to accomplish what they are trying to do.

The overall structure of the program also follows the model-view-controller design pattern, as we have the mainwindow class representing the view, the RaDoTechApp representing the controller, and the Profile, ScanResult, and RaDoTechDevice representing the model.