**Team Project: Developing and testing a MCT device simulator**

**Course: COMP 3004A (Object-Oriented Software Engin)**

**Prof: Vojislav Radonjic**

**Team Number : 29**

**Team Members:**

* **Alex Mathew (101037324)**
* **Shahroz Haider (101079945)**
* **Matthew Hettich (101014463)**
* **Usman Haider (101167249)**

**USE CASES**

**UC1:  Concrete use case - Performing treatment on a patient**

Actors:

Patient and Operator.

Pre-condition:

The device is fully functional and switched on.

Description:

1. The simulator is initiated/device is powered on.
2. The operator then navigates through the main menu where they are presented with the six main options offered by the device. The list of options which are available are as follows:

* Programs
* Frequency
* Med
* Screening
* Children
* Settings
* History

1. The programs option is selected from the list by the operator.
2. Following this, the appropriate treatment mode is selected for the patient and the time duration is settled by maintaining contact with the patient’s skin.
3. The electrodes on the device then detect the electrical impedance from the patient’s nerve and send out modulated electrical signals for the treatment
4. Visual and audio output is provided to inform the operator about the detected tissues.
5. The operator is also allowed to adjust the power of the modulated signals using the (left and right) buttons on the device.
6. After using the device, operator turns off the device.

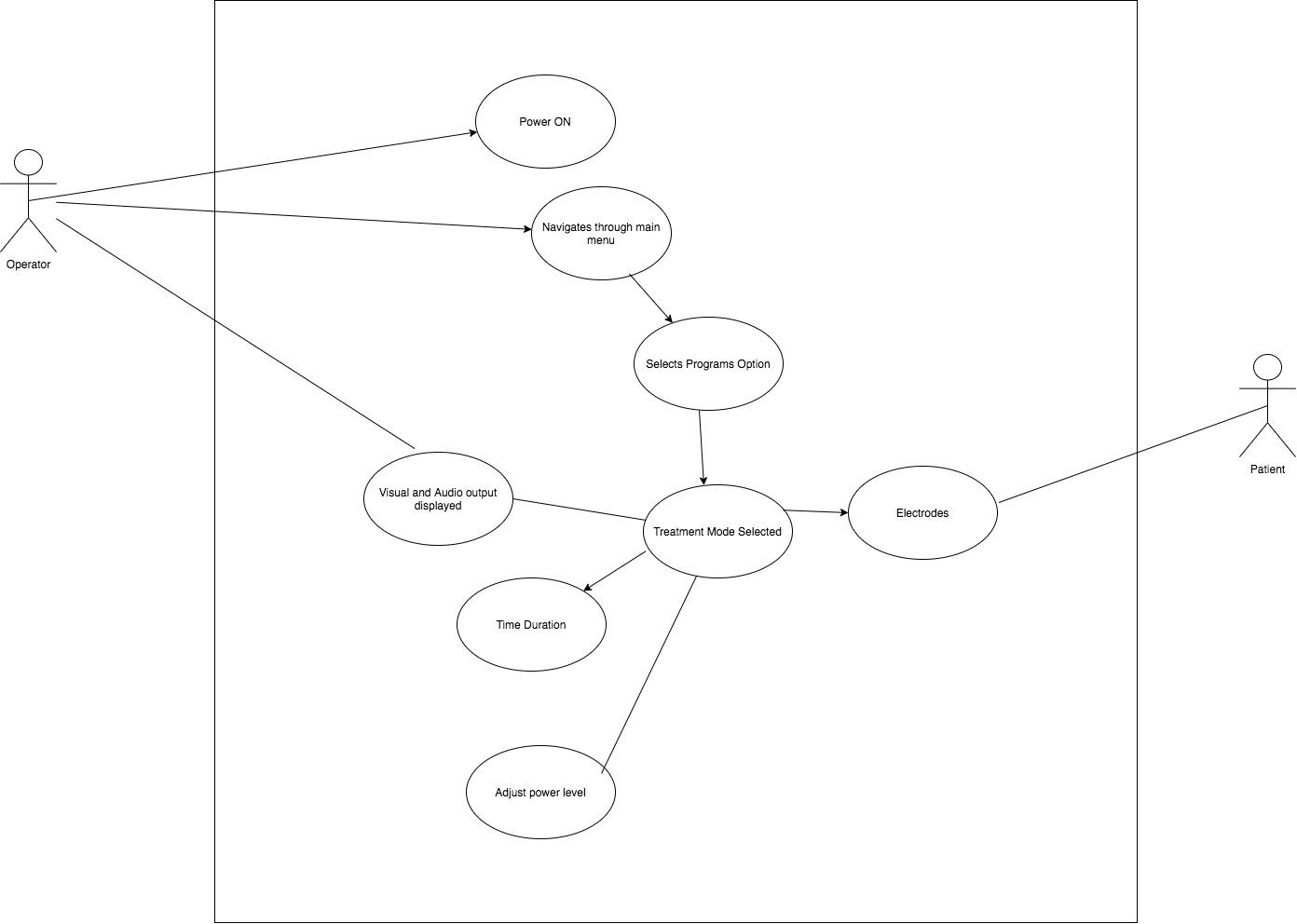
Alternatives:

* If the battery level is low or any sort of malfunction occurs, the device must be charged or replaced.

Post condition:

The patient’s treatment has been successfully completed.

Figure 1: Use Case Diagram 1



**UC2: Selecting the desired treatment mode**

Actors:

The operator

Pre-conditions:

The device is fully functional.

Description:

1. The simulator is initiated/device is powered on.
2. The operator then has to navigate through the main menu and selects the programs option.
3. Following this the operator is provided the various options of treatments to select from. The options presented are as follows:

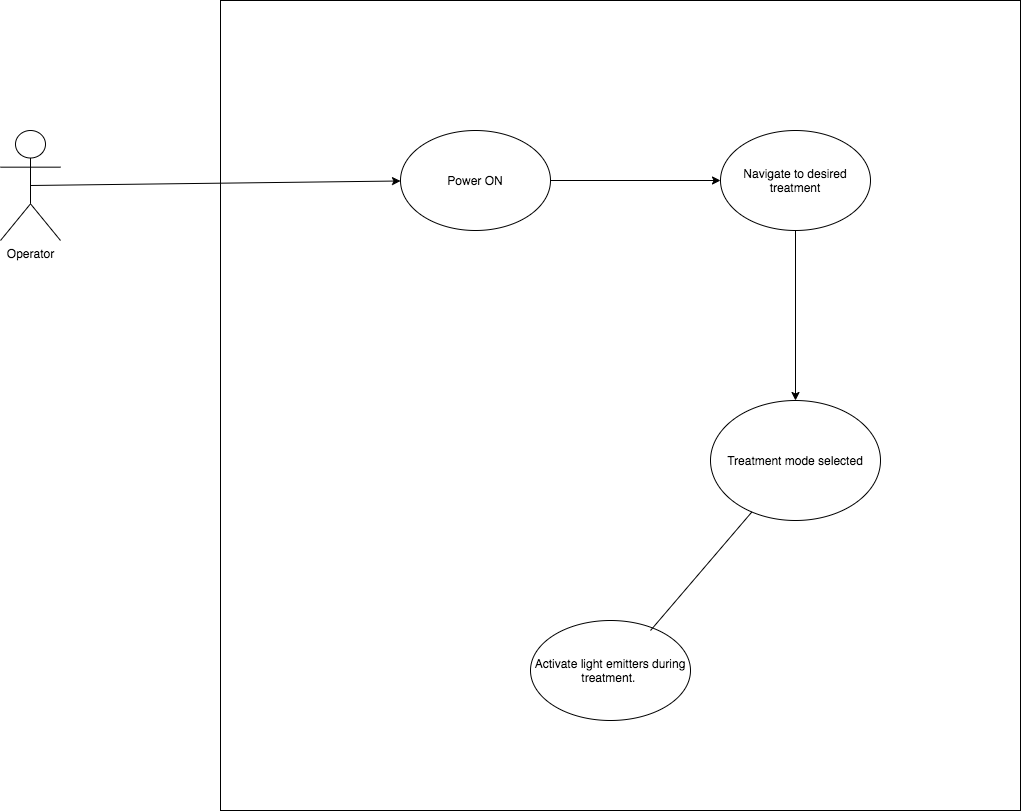
* Coughing
* Pain
* Allergic Reaction
* Bloating

1. The appropriate treatment mode is selected and is ready to perform the therapeutic intervention.
2. During the treatment, the light emitter of the device gets activated and light up.

Post conditions:

Desired treatment mode is selected.

Figure 2: Use Case Diagram 2



**UC3: Changing the display settings of the device**

Actors:

The Operator.

Pre-conditions:

The device is fully functional

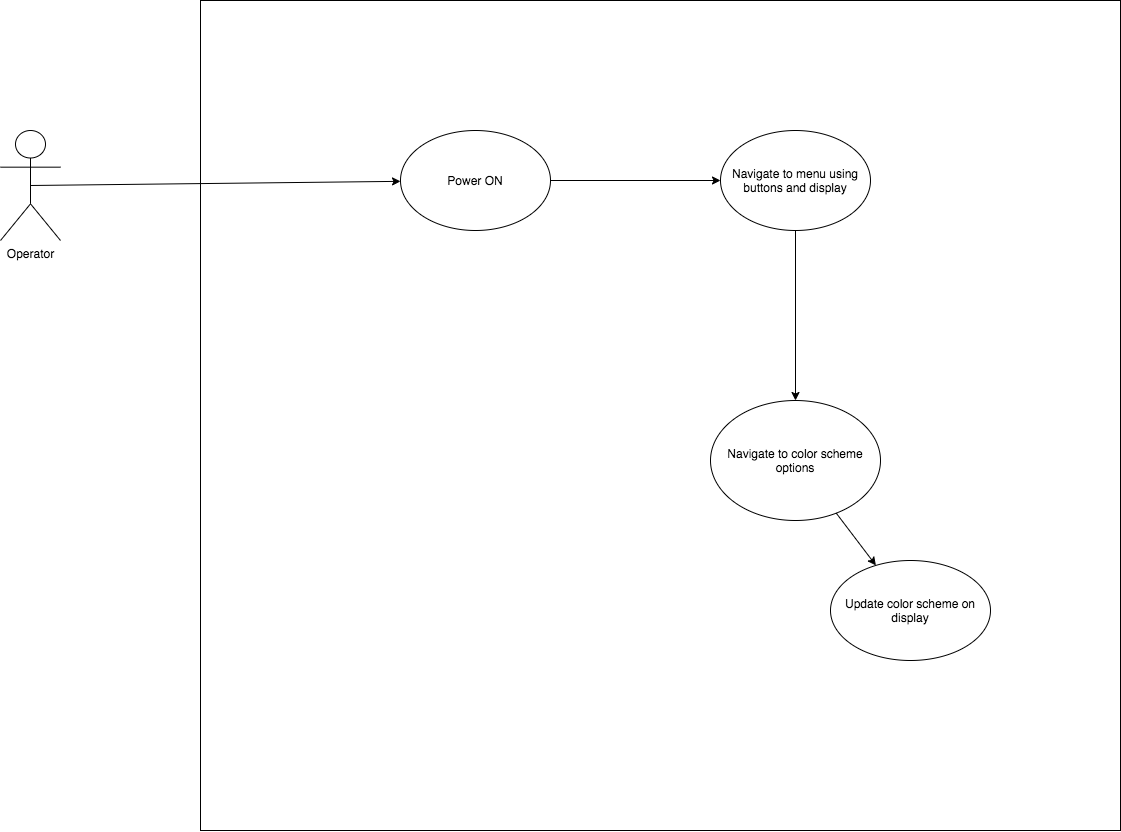
Description:

1. The simulator is initiated/device is powered on.
2. The operator then has to navigate through the main menu and selects the settings option.
3. Over here the operator is provided with a wide range of options that allow them to customize the user-interface of the application including being able to change the language.
4. The user then selects the screen layout option where he would be able to customize the layout of his device’s UI .

Post-conditions:

The desired display changes are made by the user.

Figure 3: Use Case Diagram 3

**

**UC4: Selecting the frequency range of the device**

Actors:

The operator.

Pre-conditions:

The device is fully functional

Description:

1. The simulator is initiated/device is powered on.
2. The operator then has to navigate through the main menu and selects the frequency option.
3. With this option, the user of the device is presented with a range of frequencies for treatment. The frequencies displayed are:

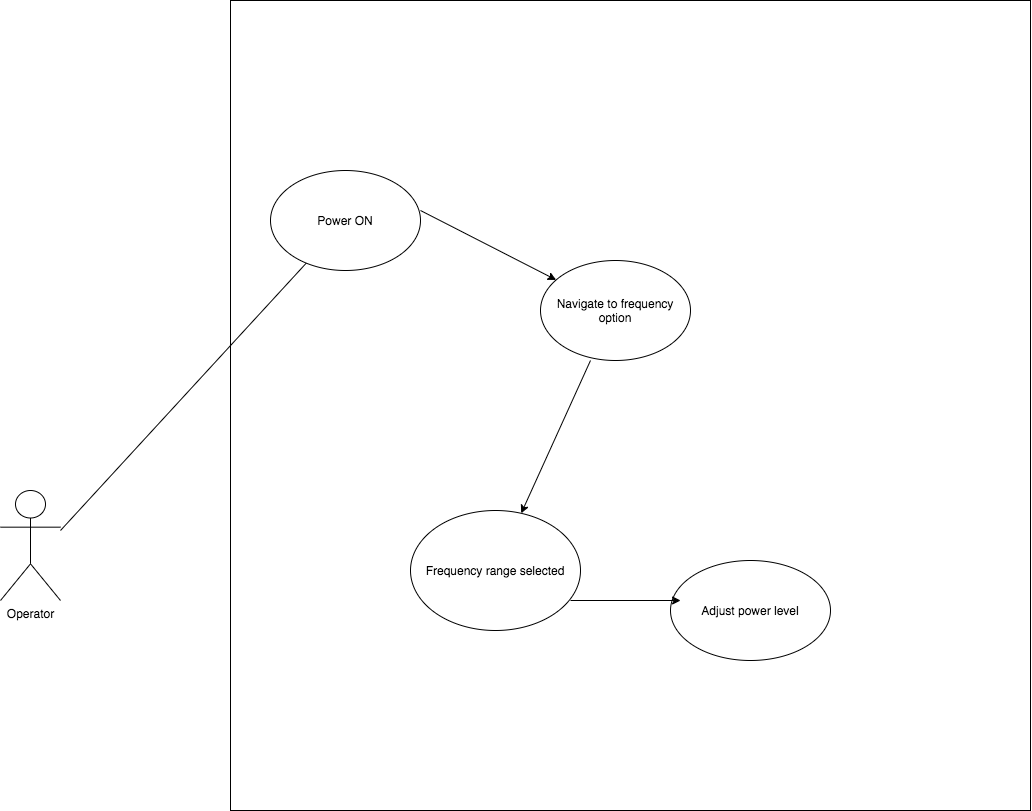
* 1.0 – 9.9 Hz
* 10 Hz
* 20 Hz
* 60 Hz
* 77 Hz
* 125 Hz
* 140 Hz
* 200 Hz
* 7710 Hz
* 7720 Hz

1. The user then selects the required frequency for treatment.
2. The user is instructed to proceed by clicking the OK button
3. Following this, the timer starts and the device starts emitting the electric signals. The user is also allowed to adjust the power level of the frequency using the buttons(left and right) in the device.

Post-conditions:

The frequency range is selected by the operator.

Figure 4: Use Case Diagram 4



**UC5: Selecting the children’s treatment option**

Actors:

The Operator

Pre-conditions:

The device is fully functional

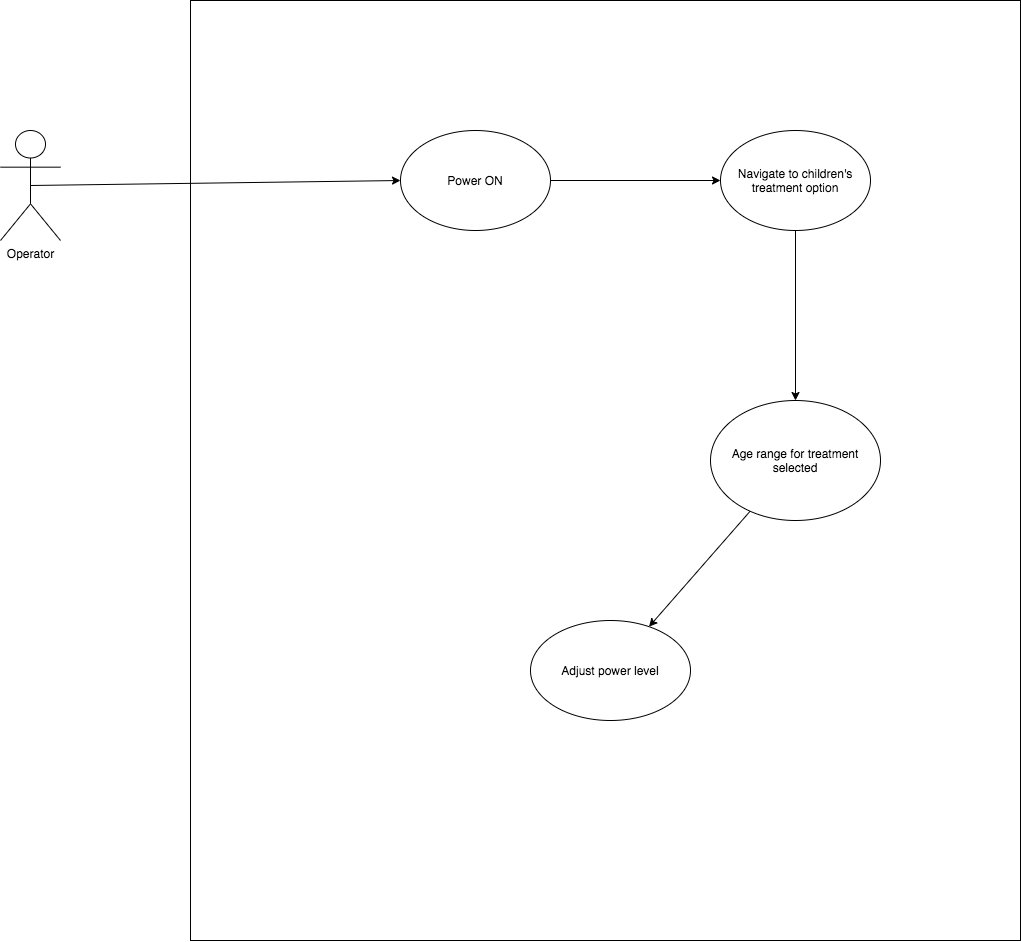
Description:

1. The simulator is initiated/device is powered on.
2. The operator navigates through the main menu and selects the children option. This section is for the treatment options available for children and displays the age-ranges for the treatments.
3. The necessary age range for the treatment is then selected.
4. The operator then proceeds to continue with the treatment.
5. Following this, the timer starts and the device starts emitting the electric signals. The user is also allowed to adjust the power level of the frequency using the buttons(left and right) in the device.

Post-conditions:

The desired age range is selected by the operator for the treatment.

Figure 5 : Use Case Diagram 5



**UC6: Selecting the recording option**

Actors:

The Operator

Pre-conditions:

The device is fully functional and treatments have been performed prior to this.

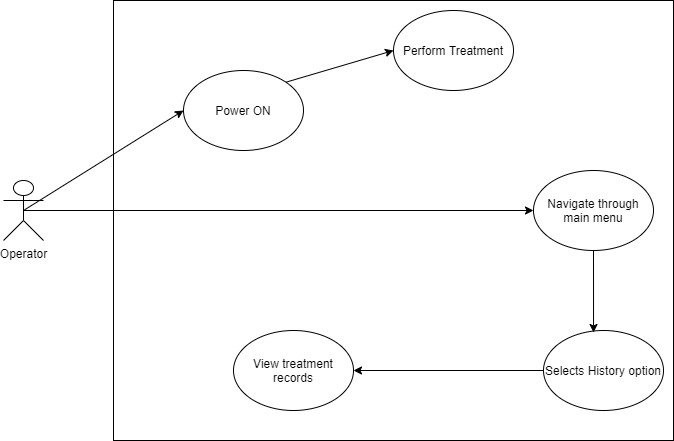
Description:

1. The operator navigates through the main menu and selects the “HISTORY” option after conducting treatments (via the programs section).
2. This section basically records and keeps track of the history of treatments performed.
3. Over here the operator is presented with 2 options to either view the history of the treatments used or to clear the history of the treatments.
4. Having selected the view option, the operator then can scroll through the treatments which were performed earlier.
5. When the clear option is selected, it erases the recorded treatments from the treatments history.

Post-conditions:

The user can now view the history of treatments performed along with their details.

Figure 6: Use Case Diagram 6



**Overall Use Case Diagram**



**Object Oriented Analysis**

The OO design diagram below shows High-level perspective. The figure shows the analysis phase as follows:

• The driver class is mainly used to initiate the start of the application which begins first with initializing the MainWindow class through the use of a main() function.

• The MainWindow class is the primary agent that makes changes to the application; it is responsible for initializing the graphical user interface and displaying the user interaction menu to select the functions provided by the device. In addition, it can also enable users to understand task progress and device hardware status (such as battery power), and send notification sounds as needed.

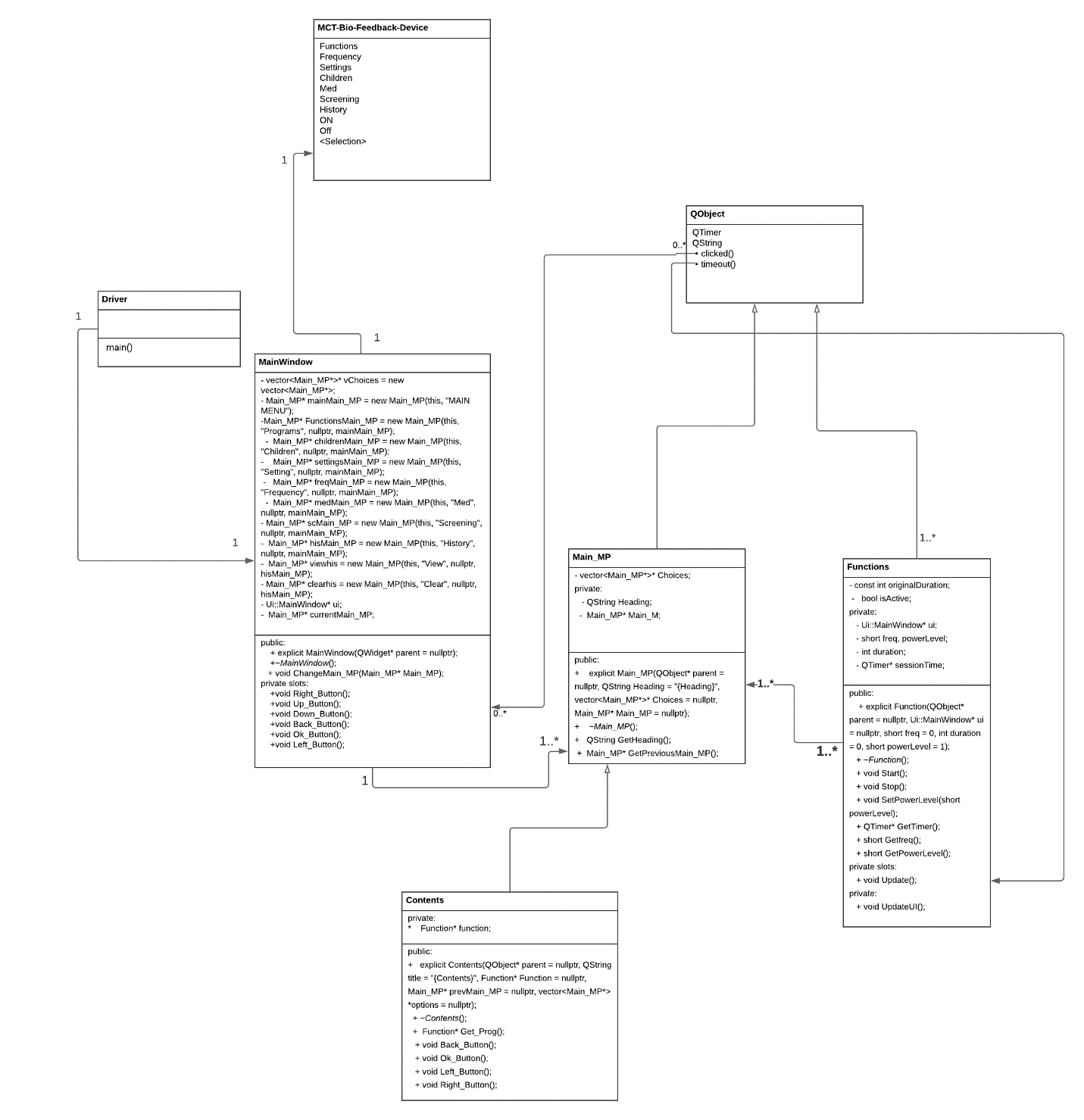
• The main\_MP class acts primarily as a constructor for the options on the MainWindow, it also records what the users selection is and returns that Qstring back to the MainWindow class so that it can update the UI accordingly.

•The functions class takes care of the primary operations that need to occur for a function to run(ie. start(), stop()).  It also keeps track of specific task progress info for decreasing values in Qtimer and power levels.  When one of these values hits their threshold, they update the UI by returning Qstrings with the new changes.

•The contents class deals with the buttons that are seen in the MainWindows interface.  This class serves as a storage for button functions and returns whichever is selected by the user and is requested by the Main\_MP class.  The Main\_MP class then uses this data to notify the MainWindow class so that the UI can be updated.

Below is the class diagram for points described above:

Figure 7 : UML Class diagram



**Sequence Diagrams**

Figure 8: Sequence Diagram 1 (for Use Case 1)

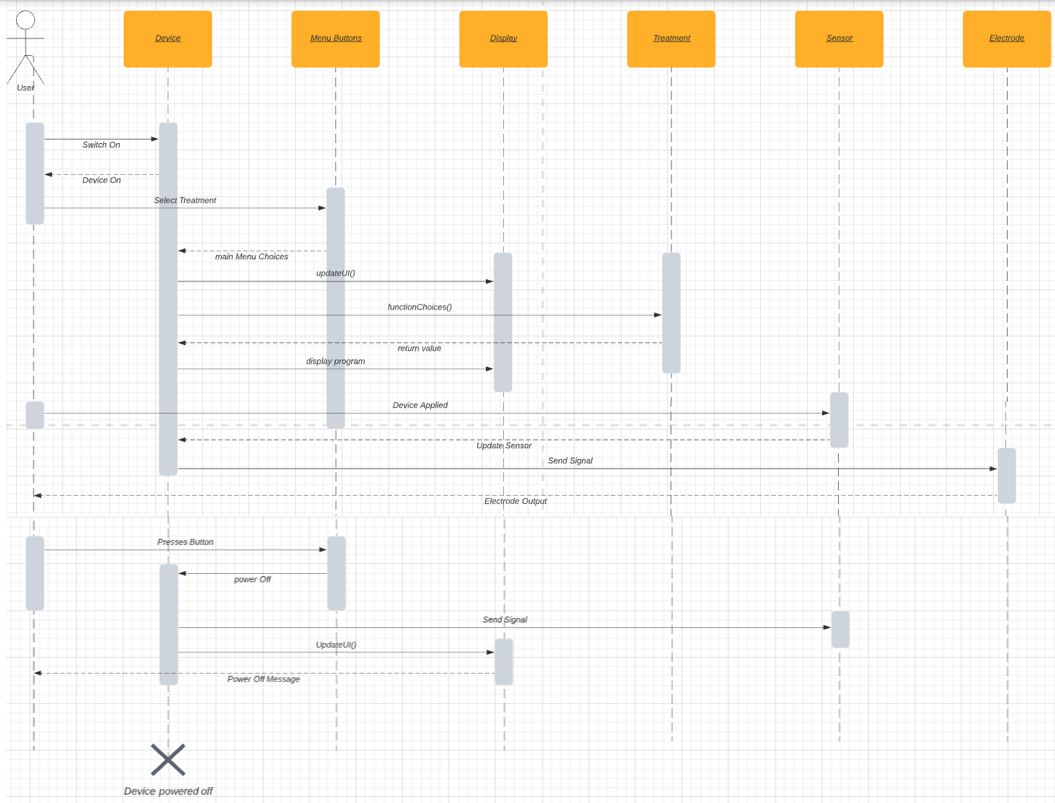


Figure 9 : Sequence Diagram 2 (for Use Case 2)

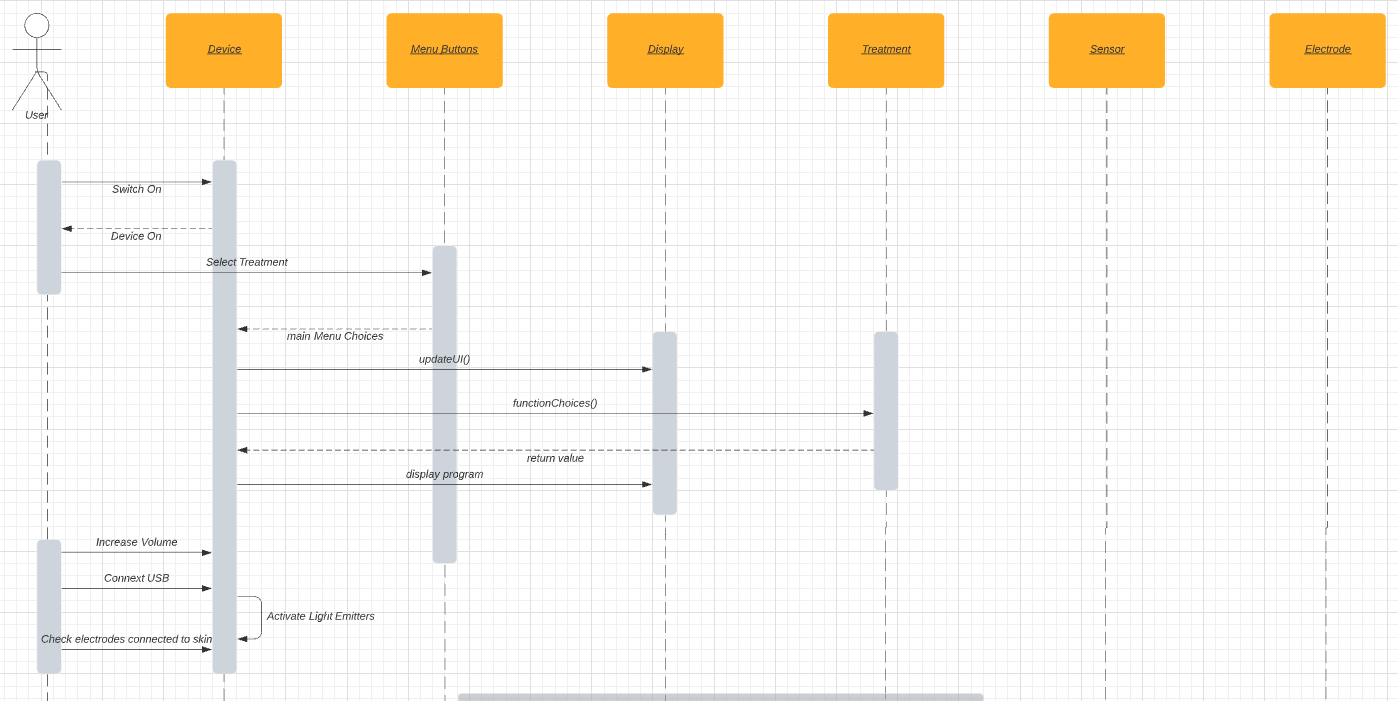


Figure 10: Sequence Diagram 3 (for Use Case 3)

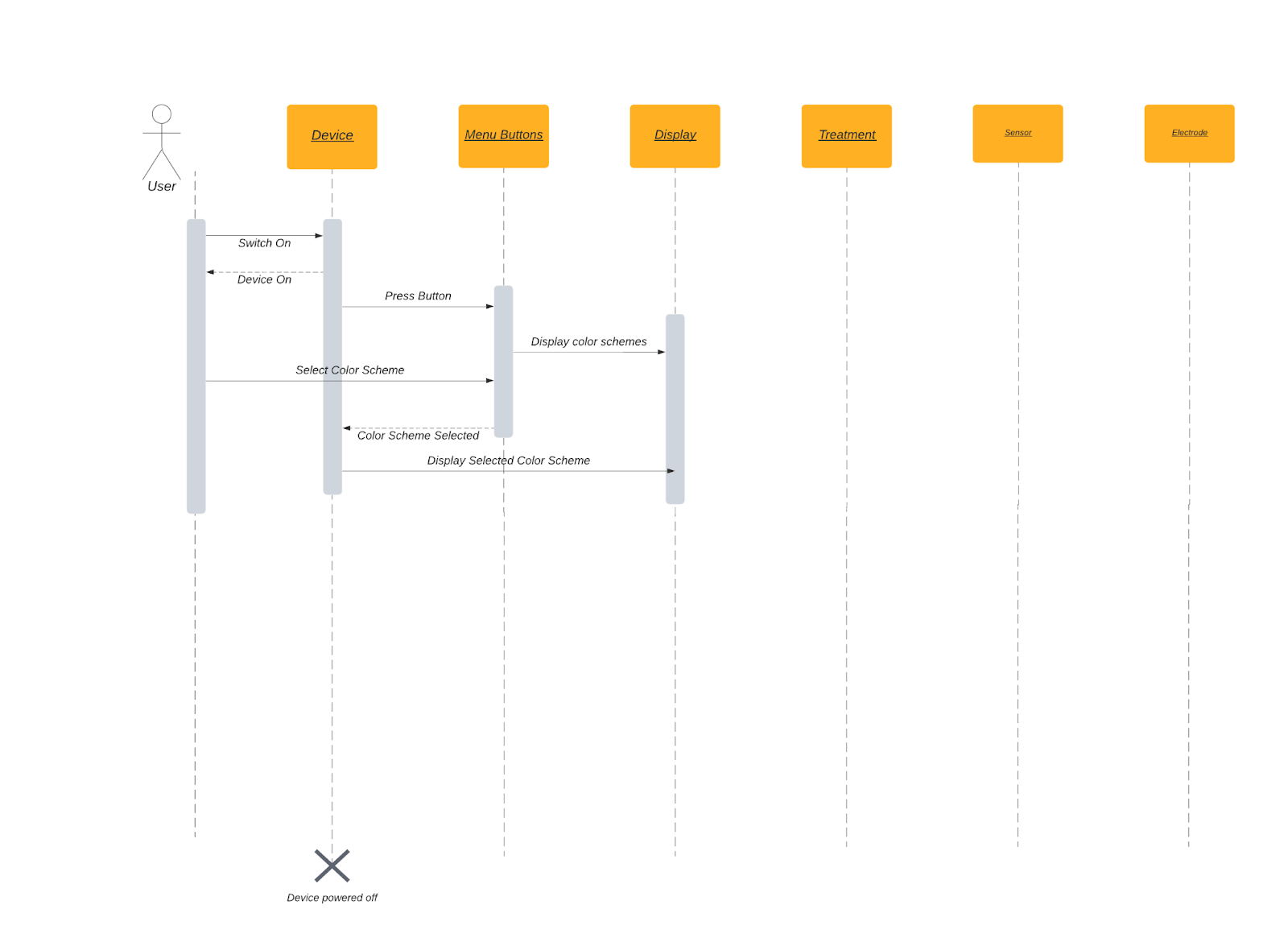
****

Figure 11: Sequence Diagram 4 (from Use Case 4)

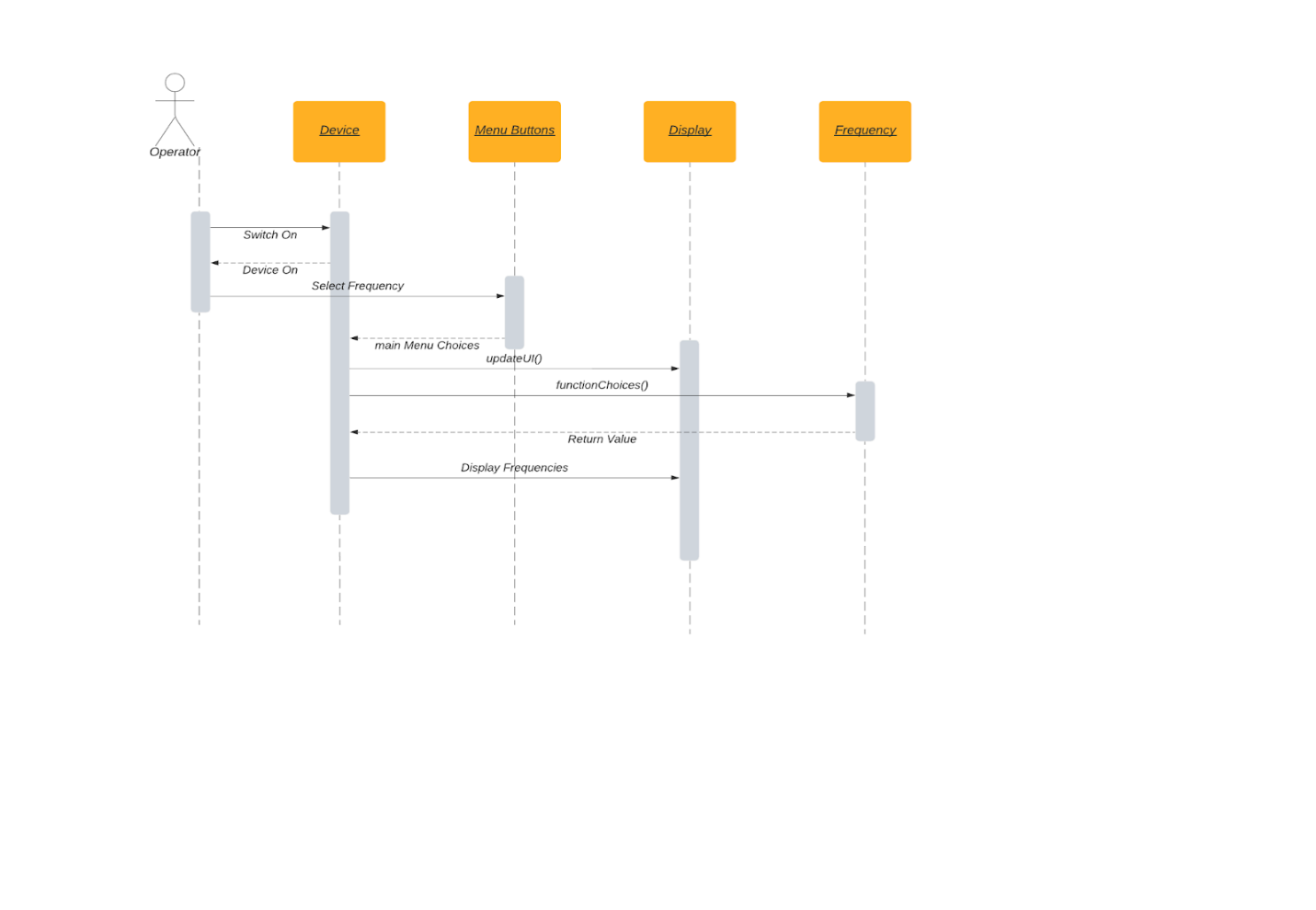


Figure 12 : Sequence Diagram 5 (for Use Case 5)

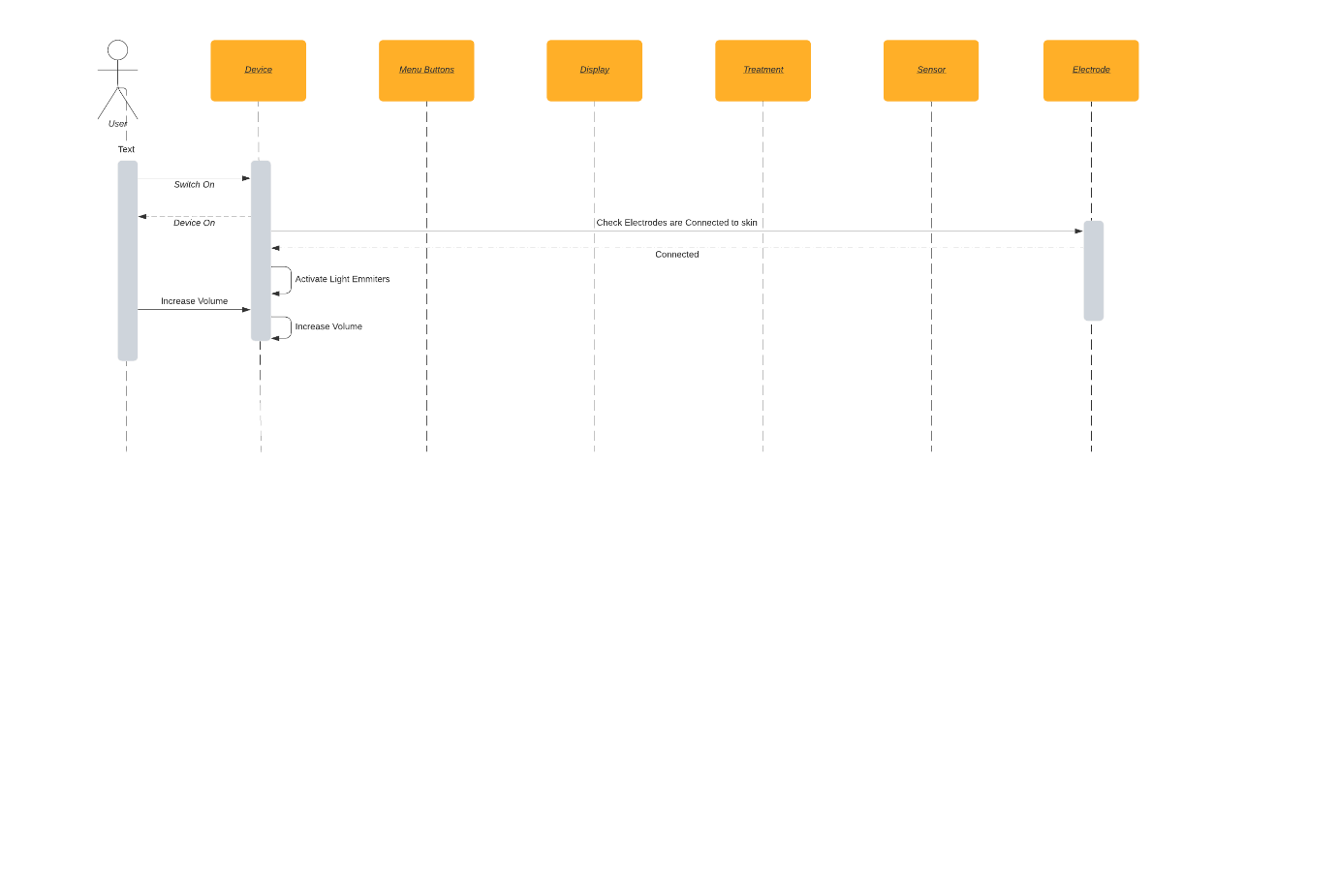


Figure 13 : Sequence Diagram 6 (for Use Case 6)

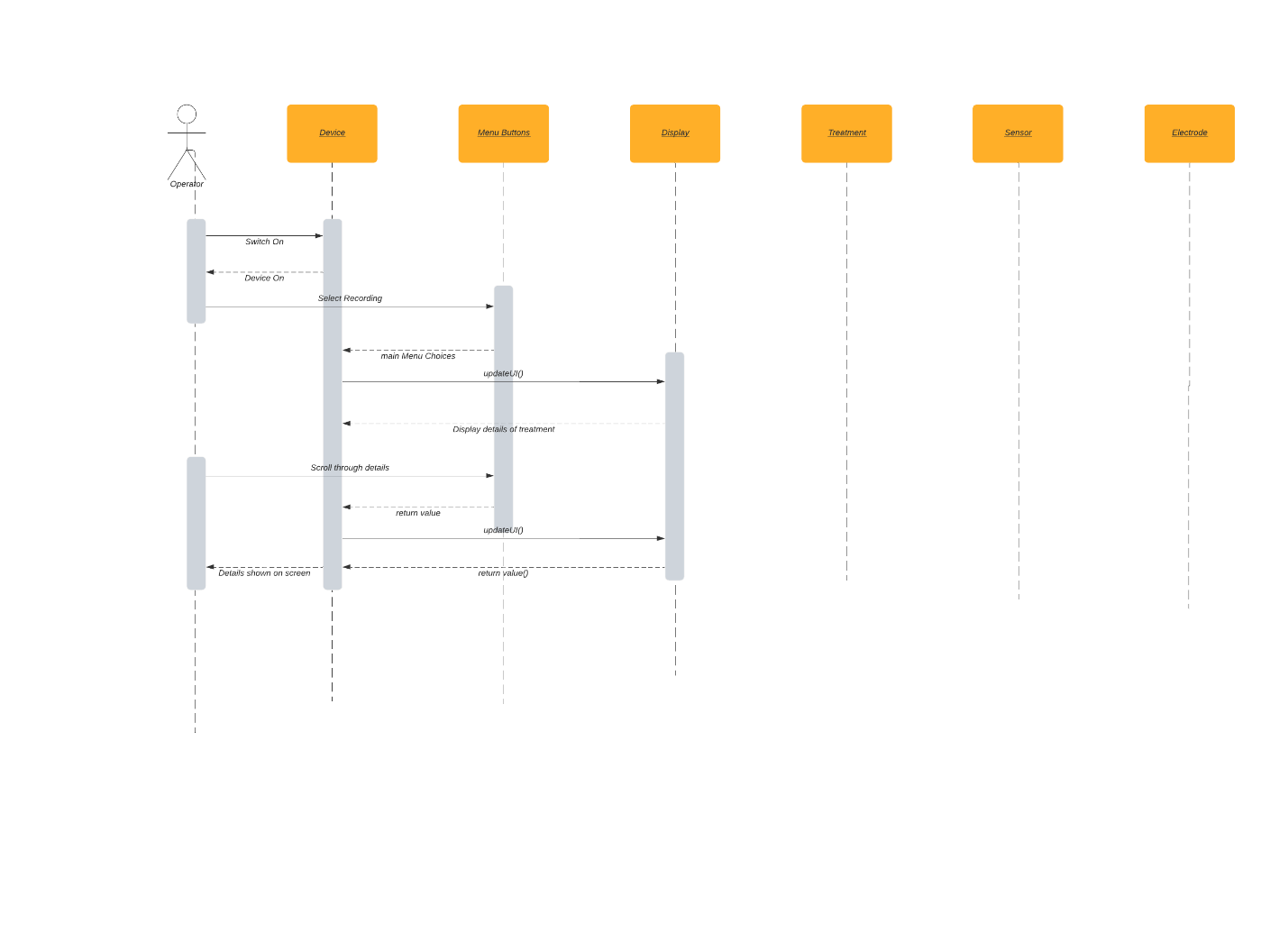
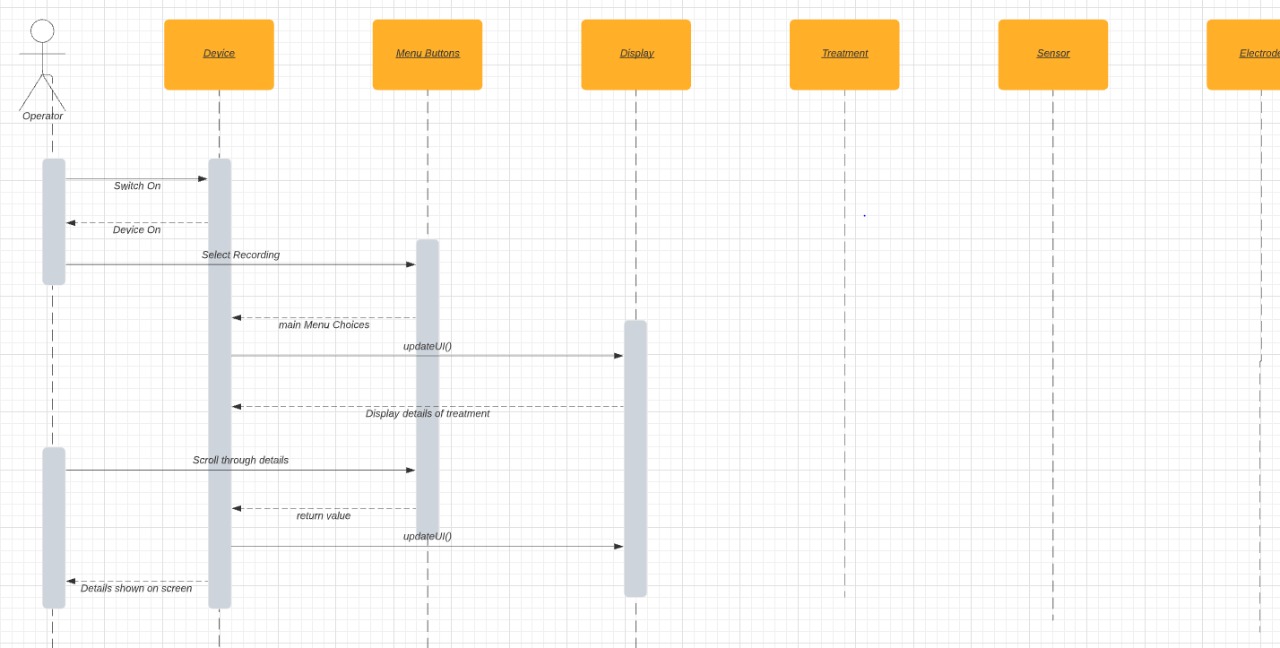


Figure 14: Sequence Diagram 6 (for Use Case 6)



**Activity Diagram**

