Classes:

Battery
DBManager (or FileManager if we decide to go that route)
Menu
Therapy:
Connection_Test

General Structure:

Menu
-> Battery
-> DBManager
-> Therapy (parent)

(intensity and duration) -> not a class, but include the functionalities

Functions:

Battery:

    Vars:
    Int batteryLevel

    Funcs:
    Int getBatteryLevel() - return if battery is high(?), low, critically low, dead
    Void decreaseLevelBy(int) - decrease level by int given

DBManager:

    Vars:
    string fileName

    Funcs:
    string readLog - read the txt/csv file
    writeLog - write (append_ to the txt/csv file

Therapy:

    Vars:
    eNum session (delta, theta, alpha, 100hz)
    int intensity - (1 ~ 8)
    int Duration - 20, 45, custom (1 ~ 180) minutes

    Funcs:
    string getTherapy() - build the therapy info in the right format for DBManger to log

bool startSession() - starts the actual session after 5 seconds delay, start timer (s = m), end the therapy when timer hits duration. Call connectionTest(), softOn, softOff.

softOn - gradually increase intensity until desired intensity is met

softOff - gradually decrease intensity until 0

ConnectionTest:

Vars:

eNum LE - state of earlobe -> not connected, connected but apply gel (okay), excellent

eNum RE - ``

Funcs:

(Done) checkConnection(): Check the status of connection, inform user on status of each ear

(?) blink_modelight(): SLOT - The mode light should blink every second to indicate that test mode has been entered.

(?) QTimer->start(1000): SIGNAL - call blink_modelight every second.

(Done) displayConnectionStatus(): Displays the status of connection after connection test.

(?) softOn(): slowly increase intensity

(?) softOff(): slowly decreases intensity:

(?) clearDisplay(): After the connection test, the display will go blank (if no connection)

(Done) wetEarLobes() - via buttons

Menu (MainWindow):

Vars:

bool power;

Funcs:

powerOn

powerOff

checkAction() - //If the user is inactive for 2 minutes call the powerOff function.

readLog() - read - done via DBManager

writeLog() - write - done via DBManager

displayLog() - display log retrieved, calls on readLog(). Use textedit display

displayBatteryLevel: Display the battery graph (adjust GUI based on the battery level)

batteryDisplay_off: Turn off the battery display after a few seconds.

periodicBatteryDisplay()

drainBattery() - higher the intensity / frequency, faster drain

switchGroups(): SLOT - Pressing/releasing the power button will switch between groups

highlightGroup(): We need to show which group the cursor is on.

navigateSessionDown(): SLOT - when the INT down arrow is pressed, it should navigate down.
navigateSessionUp(): SLOT - when the INT up arrow is pressed, it should navigate up
displayIntensity()
displayGroup() //Isn't this the same thing as the 'highlightGroup()' function? If someone agrees, please delete this function.
displaySession()
startSession():
flashSession(): Once the startSession() is executed, the session numbers should flash for 5 seconds before starting.

powerHeld(): SIGNAL - when power button is held to power off
powerClicked(): SIGNAL - when the power button is clicked it should send a signal to activate

Probably use mediator design. Observer for Ui, singleton?

Q1. Is the menu class too big? Should we break it up
Q2. When would the session symbol light up? The manual mentions selecting the session via the numbers, when would the session symbol icons lit up? After the session starts?

Notes

CheckHistoryButton????
MainWindow = Menu???
ConnectionTest issue???

Mainwindow - god object?

- How to adjust intensity after session starts
- How to quit mid-session when power button is pressed

Still To Do:

Erica:

- Connecting GUI session/timing buttons

Richard:

- Incorporate Therapy class to actually build the log

To do list from the TA

Finish up diagrams

Get ready for demo - explain to each other