Textual Explanation

Design Patterns Used

- State Pattern: We implement a state machine, and have different states to represent different behavior needed by our device. The following states are:
    - On: The device turns on (the display is set to visible)
    - Off: The device turns off (the display is no longer visible and all lights stop flashing), battery stops draining, and clean up from when the device was running
    - InMainMenu: Displays the three options of the main menu and allows selecting of each option. Battery starts draining. If there is contact between the eeg sites and the scalp the blue light is on indicating the connection.
    - SessionActive: Starts the session timer. Starts receiving signals and calculating the dominant frequency. Runs for 1 minute (for the sake of testing 5 seconds). Once the minute has passed and dominant calculated the device state is updated to Deliver Treatment
    - DeliverTreatment: Green light on. This state occurs only once we have a dominant frequency, the treatment is delivered (dominant frequency + 5hz) for a period of 1 second. The roundCount is incremented indicating the completion of a round of treatment. The device state is then updated to SessionActive
    - SessionCompletion: Once all rounds of treatment are completed, or the session prematurely enters this state is entered to ensure proper clean up of a session and to store all completed sessions
    - SessionPaused: The pause button has been pressed. Session timer is paused. If paused for more than 5 minutes (10 seconds for testing) the session automatically ends. The session will only be resumed if the resume button is pressed.
    - ContactLost: If contact is lost between an EegSite and the scalp and the device was in a session, the red light flashes, and the session remains in a paused state until contact is re-established. If contact is not re-established within 5 minutes the session automatically ends.
    - InTimeDateMenu: Allows for the setting of the date and time
    - InSessionLogMenu: Allows for the viewing of the sessions on either the device or the PC
- Observer Pattern: Although not implemented as traditional observers, the various UI elements and timers respond to these state changes, similar to an Observer pattern, where the state changes could notify different components to update accordingly.

EEG Site

The EEG site class has three main purposes:

1. The generation of test waveforms: Test wave forms are created by adding sin waves from each band of alpha, beta, delta and theta, a value picked from their relative ranges randomly, and each wave has a randomly generated weight applied to it.
2. The calculation of dominant frequency: The dominant is calculated by first doing a Fast Fourier Transform (using the FFTW library: https://www.fftw.org/), then completing a power spectrum analysis (by calculating the magnitude squared of each component), and finally calculating the power of each band to determine the dominant band, followed by calculating the power of each frequency within that band to determine the dominant frequency.
3. To deliver treatment to the EEG site (if connected to actual hardware it would deliver the actual treatment)- for the purposes of testing- it just prints the treatment frequency to the console

The mainwindow stores each EEG site and equivalent most recent waveforms for each site, which for testing allows us to select which site we want to view the waveform for, and then using the class qcustomplot (https://www.qcustomplot.com/) these waveforms can be displayed in the UI in our 'admin' window.

Sessions and Database Manager

The Session class stored the data from a LENS treatment session on the Neureset device. The data includes the treatment's start time, and the before and after treatments baseline average frequencies. Each session can be stored in the neurest database that is created in the DBManager class. The DBManager class manages the neureset database and creates the sessions table within it. It allows sessions to be inserted, and deleted as well as retrieval of the pc and device sessions histories. The mainwindow is where a DBManager instance and a current Session pointer is created and where that pointer is eventually added into the database once populated. This is also where the pc and device sessions histories from the database are connected to the ui to be displayed on the device.