**TRACEABILITY MATRIX**

FUNCTION is the function scope where there is the call or invoking of a signal

SIGNAL is the QT signal emitted to invoke the function being called

CALL is the exact code instruction invoked/executed to fulfil requirement event

| ID | Requirement | Related Use Case | Fulfilled By | Tested By |
| --- | --- | --- | --- | --- |
| UC101 | Patient turns on the Neureset device | **USE CASE 1: Use A Neureset Device** | CALL  MainWindow::on\_powerButton\_clicked() | Clicking the following button under **Device View** |
| UC102 | Device’s red light turns on, indicating electrodes are not connected | **USE CASE 1: Use A Neureset Device** | FUNCTION  MainWindow::on\_powerButton\_clicked()  CALL  **ui->ContactLostIndicator->setStyleSheet("background-color: red;");** | Observing the following display under **Device View** |
| UC103 | Device menu presents three options | **USE CASE 1: Use A Neureset Device** | FUNCTION  MainWindow::on\_powerButton\_clicked()  CALL  **model->setStringList(menuOptions);** | Observing the following display under **Device View** |
| UC104 | Device’s battery indicator shows current charge | **USE CASE 1: Use A Neureset Device** | FUNCTION  MainWindow::on\_powerButton\_clicked()  CALL  **MainWindow::togglePower()** | Observing the following display under **Device View** |
| UC105 | Patient puts on EEG headset | **USE CASE 1: Use A Neureset Device** | CALL  **MainWindow::on\_electrodeReconnect\_clicked()** | Clicking the following button under **Admin View** |
| UC106 | Device checks EEG contact | **USE CASE 1: Use A Neureset Device** | SIGNAL  MainWindow::electrodeContactRegained()  CALL  **Controller::setElectrodeContactSecured()** | **N/A** |
| UC107 | Device’s blue light turns on, indicating good electrode contact | **USE CASE 1: Use A Neureset Device** | FUNCTION  MainWindow::on\_electrodeReconnect\_clicked()  CALL  **ui->ContactSecureIndicator->setStyleSheet("background-color: blue;");** | Observing the following display under **Device View** |
| UC108 | Device’s red light turns off | **USE CASE 1: Use A Neureset Device** | FUNCTION  MainWindow::on\_electrodeReconnect\_clicked()  CALL  **ui->ContactLostIndicator->setStyleSheet("background-color: grey;");** | Observing the following display under **Device View** |
| UC109 | Patient presses select button on ‘New Session’ | **USE CASE 1: Use A Neureset Device** | CALL  **MainWindow::on\_selectButton\_clicked()** | Clicking the following button under **Device View**    In the following display context |
| UC110 | Device confirms electrodes are connected | **USE CASE 1: Use A Neureset Device** | FUNCTION  MainWindow::on\_selectButton\_clicked()  CALL  **if(controller->electrodesConnected())** | **N/A** |
| UC111 | Device initiates new treatment session | **USE CASE 1: Use A Neureset Device** | FUNCTION  MainWindow::on\_selectButton\_clicked()  SIGNAL  MainWindow::signalNewSession()  CALL  **Controller::startNewSession()** | **N/A** |
| UC112a | Device displays timer of approximately 45 seconds, indicating remaining treatment time | **USE CASE 1: Use A Neureset Device** | FUNCTION  Controller::startNewSession()  CALL  **remainingTime = TREATMENT\_TIME\_SECONDS** | Observing the following display under **Device View** *(couldn't capture exact start)* |
| UC112b | Device displays a progress bar of 0%, indicating remaining treatment time |  | FUNCTION  MainWindow::on\_selectButton\_clicked()  CALL  **ui->progressBar->setValue(0)** | Observing the following display under **Device View** |
| UC113  a | Device display timer begins ticking down | **USE CASE 1: Use A Neureset Device** | FUNCTION  Controller::updateSessionTimerAndProgress()  SIGNAL  Controller::updateTimerAndProgressDisplay()  CALL  MainWindow::updateUITimerAndProgress()  **ui->timerLabel->setText(timeString);** | Observing the following display under **Device View** |
| UC113b | Device display progress bar begins ticking up | **USE CASE 1: Use A Neureset Device** | FUNCTION  Controller::updateSessionTimerAndProgress()  SIGNAL  Controller::updateTimerAndProgressDisplay()  CALL  MainWindow::updateUITimerAndProgress()  **ui->progressBar->setValue(progressPercentage)** | Observing the following display under **Device View** |
| UC114 | Device samples EEG data from all electrode sites concurrently for 5 seconds | **USE CASE 1: Use A Neureset Device** | FUNCTION  Controller::startNewSession()  SIGNAL  Controller::startElectrodeInitialBaseline()  CALL  **Electrode::startSession()** | Clicking the following button under **PC View** tosee sampled data  (<- separate call) |
| UC115 | Device calculates initial baseline average dominant frequency for each site at the end of the sampling | **USE CASE 1: Use A Neureset Device** | FUNCTION  Electrode::startSession()  CALL  Electrode::getInitialBaselineFrequency() | OBSERVING **CONSOLE LOG** |
| UC116 | Device begins treatment application at the first EEG site | **USE CASE 1: Use A Neureset Device** | FUNCTION  Electrode::getInitialBaselineFrequency()  CALL  EEGData = source.getSourceData() | OBSERVING **CONSOLE LOG** |
| UC117 | Device’s green light turns on to indicate treatment is being delivered | **USE CASE 1: Use A Neureset Device** | FUNCTION  Electrode::getInitialBaselineFrequency()  CALL  Electrode::generateWaveData()  SIGNAL  Electrode::initialBaselineFinished(electrodeNum)  CALL  Controller::setElectrodeFinishedTreatment()  Controller::startIndividualElectrodeTreatment()  SIGNAL  Controller::startElectrodeTreatment(electrodeNum)  CALL  MainWindow::slotTreatmentApplicationStarted()  **ui->TreatmentIndicator->setStyleSheet("background-color: green")** | Observing the following display under **Device View** |
| UC118 | Device adds an offset frequency of 5hz to the baseline average dominant frequency corresponding to that electrode site | **USE CASE 1: Use A Neureset Device** | FUNCTION  Electrode::startTreatment()  CALL  **offset += 5;** | N/A |
| UC119 | Device Applies offset feedback wave | **USE CASE 1: Use A Neureset Device** | FUNCTION  Electrode::startTreatment()  CALL  **qInfo() << "Delivering signal at " << offset << "Hz Offset To Dominant Frequency of " << freqData.getBefore() << " Hz";** | OBSERVING **CONSOLE LOG**  “Delivering signal at 5 Hz Offset to Dominant Frequency of \_\_ Hz” |
| UC120 | Device adds an offset frequency of 10hz to the baseline average dominant frequency corresponding to that electrode site | **USE CASE 1: Use A Neureset Device** | FUNCTION  Electrode::startTreatment()  CALL  **offset += 5;** | **N/A** |
| UC121 | Device Applies Offset Feedback Wave | **USE CASE 1: Use A Neureset Device** | FUNCTION  Electrode::startTreatment()  CALL  **qInfo() << "Delivering signal at " << offset << "Hz Offset To Dominant Frequency of " << freqData.getBefore() << " Hz";** | OBSERVING **CONSOLE LOG**  “Delivering signal at 10 Hz Offset to Dominant Frequency of \_\_ Hz” |
| UC122 | Device repeats process of for 15hz and 20hz offsets added to the initial dominant frequency | **USE CASE 1: Use A Neureset Device** | FUNCTION  Electrode::startTreatment()  CALL  **offset += 5;**  **qInfo() << "Delivering signal at " << offset << "Hz Offset To Dominant Frequency of " << freqData.getBefore() << " Hz";** | OBSERVING **CONSOLE LOG**  “Delivering signal at \_\_ Hz Offset to Dominant Frequency of \_\_ Hz” |
| UC123 | Device selects next EEG site | **USE CASE 1: Use A Neureset Device** | FUNCTION  Controller::setElectrodeFinishedTreatment()  CALL  **Controller::startIndividualElectrodeTreatment()** | **N/A** |
| UC125 | Device completes treatment for all 7 electrode sites | **USE CASE 1: Use A Neureset Device** | FUNCTION  Controller::setElectrodeFinishedTreatment()  CALL  **numElectrodesFinished == static\_cast<int>(electrodesFinishedTreatment.size())** | **N/A** |
| UC126 | Device’s green treatment indicator light turns off | **USE CASE 1: Use A Neureset Device** | FUNCTION  Controller::setElectrodeFinishedTreatment(int electrodeNum)  SIGNAL  Controller::startElectrodeFinalBaseline()  CALL  MainWindow::slotTreatmentApplicationFinished()  **ui->TreatmentIndicator->setStyleSheet("background-color: grey")** | Observing the following display under **Device View** |
| UC127 | Device calculates final baseline average dominant frequency for all 7 EEG sites concurrently over 5 seconds | **USE CASE 1: Use A Neureset Device** | FUNCTION  Controller::setElectrodeFinishedTreatment(int electrodeNum)  SIGNAL  Controller::startElectrodeFinalBaseline()  CALL  **Electrode::getFinalBaselineFrequency()** | OBSERVING **CONSOLE LOG** |
| UC128 | Device finishes calculating final baseline | **USE CASE 1: Use A Neureset Device** | FUNCTION  Electrode::getFinalBaselineFrequency()  SIGNAL  Electrode::finalBaselineFinished(electrodeNum)  CALL  **Controller::setElectrodeFinishedFinalBaseline()** | **N/A** |
| UC129 | Device shows “treatment session complete” on display | **USE CASE 1: Use A Neureset Device** | FUNCTION  Controller::setElectrodeFinishedFinalBaseline()  CALL  Controller::recordSession()  SIGNAL  Controller::signalTreatmentSessionComplete()  CALL  MainWindow::slotTreatmentSessionComplete()  **MainWindow::displayMessage("Treatment Session Complete", MENU\_SCREEN)** | Observing the following display under **Device View** |
| UC130 | Device creates and a session log and saves it to storage | **USE CASE 1: Use A Neureset Device** | FUNCTION  Controller::recordSession()  CALL  **SessionLog\* session = new SessionLog(sessionDateTime, electrodeData)**  **fileManager.addSessionLog(*session*)** | Clicking the following button under **PC View**    And observing the bottom most element in following display under **PC View**    **Optional:** double click last element to display contents in following display under **PC View** |
| UC131 | Device returns to menu options | **USE CASE 1: Use A Neureset Device** | FUNCTION  Controller::recordSession()  SIGNAL  Controller::signalTreatmentSessionComplete()  CALL  MainWindow::slotTreatmentSessionComplete()  **MainWindow::displayMessage("Treatment Session Complete", MENU\_SCREEN);** | Observing the following display under **Device View** |
|  |  |  |  |  |
| UC201 | Device detects contact loss of one or more of the electrodes | **USE CASE 2: Electrode Loses Contact During Treatment Session** | CALL  Controller::setElectrodeContactLost() | Clicking the following button under **Admin View** |
| UC202 | Device pauses treatment session | **USE CASE 2: Electrode Loses Contact During Treatment Session** | FUNCTION  Controller::setElectrodeContactLost()  CALL  Controller::pauseSession()  SIGNAL  Controller::pauseElectrodes()  CALL  **Electrode::handlePauseRequested()** | **N/A** |
| UC203 | Device’s blue light turns off | **USE CASE 2: Electrode Loses Contact During Treatment Session** | FUNCTION  MainWindow::on\_electrodeDisconnect\_clicked()  CALL  **ui->ContactSecureIndicator->setStyleSheet("background-color: grey;");** | Observing the following display under **Device View** |
| UC204 | Device’s red light turns on | **USE CASE 2: Electrode Loses Contact During Treatment Session** | FUNCTION  MainWindow::on\_electrodeDisconnect\_clicked()  CALL  **ui->ContactLostIndicator->setStyleSheet("background-color: red;");** | Observing the following display under **Device View** |
| UC205 | Device begins beeping | **USE CASE 2: Electrode Loses Contact During Treatment Session** | FUNCTION  MainWindow::on\_electrodeDisconnect\_clicked()  SIGNAL MainWindow::electrodeContactLost()  CALL  Controller::setElectrodeContactLost()  **cout << “BEEP BEEP BEEP….” <<endl** | **N/A** |
| UC206 | Device display updates to show ‘Session Paused: Please Reconnect Electrodes’ | **USE CASE 2: Electrode Loses Contact During Treatment Session** | FUNCTION  MainWindow::on\_electrodeDisconnect\_clicked()  CALL  **MainWindow::displayMessage("Electrode Contact Lost, Please Reconnect", TREATMENT\_SCREEN);** | Observing the following display under **Device View** |
| UC207 | User corrects electrode contact within 5 minutes | **USE CASE 2: Electrode Loses Contact During Treatment Session** | CALL  MainWindow::on\_electrodeReconnect\_clicked() | Clicking the following button under **Admin View** |
| UC208 | Device’s red light turns off | **USE CASE 2: Electrode Loses Contact During Treatment Session** | FUNCTION  MainWindow::on\_electrodeReconnect\_clicked()  CALL  **ui->ContactLostIndicator->setStyleSheet("background-color: grey;")** | Observing the following display under **Device View** |
| UC209 | Device’s blue light turns on | **USE CASE 2: Electrode Loses Contact During Treatment Session** | FUNCTION  MainWindow::on\_electrodeReconnect\_clicked()  CALL  **ui->ContactSecureIndicator->setStyleSheet("background-color: blue;");** | Observing the following display under **Device View** |
| UC210 | Device stops beeping | **USE CASE 2: Electrode Loses Contact During Treatment Session** | FUNCTION  MainWindow::on\_electrodeReconnect\_clicked()  SIGNAL  MainWindow::electrodeContactRegained()  CALL  Controller::setElectrodeContactSecured()  **cout << “....BEEP BEEP BEEP” <<endl** | **N/A** |
| UC211 | Device resumes treatment | **USE CASE 2: Electrode Loses Contact During Treatment Session** | FUNCTION  Controller::setElectrodeContactSecured()  CALL  Controller::resumeTreatmentSession()  SIGNAL  Controller::resumeSession();  CALL  **Electrode::resume()** | **N/A** |
| UC212 | Device display updates to previous treatment time and progress remaining | **USE CASE 2: Electrode Loses Contact During Treatment Session** | FUNCTION  MainWindow::on\_electrodeReconnect\_clicked()  CALL  **ui->menuView->setFocus()** | Observing the following display under **Device View** |
|  |  |  |  |  |
| UC301 | PATIENT presses pause button during a treatment session | **USE CASE 3: Patient Presses Pause During Treatment Session** | CALL  MainWindow::on\_pauseButton\_clicked() | Clicking the following button under **Device View** |
| UC302 | Device detects pause button pressed | **USE CASE 3: Patient Presses Pause During Treatment Session** | FUNCTION  MainWindow::on\_pauseButton\_clicked()  SIGNAL  MainWindow::pauseButtonPressed() | **N/A** |
| UC303 | Device pauses treatment session | **USE CASE 3: Patient Presses Pause During Treatment Session** | SIGNAL  MainWindow::pauseButtonPressed()  CALL  **Controller::pauseSession()** | **N/A** |
| UC304 | Device displays ‘Treatment Paused’ | **USE CASE 3: Patient Presses Pause During Treatment Session** | FUNCTION  MainWindow::on\_pauseButton\_clicked()  CALL  **MainWindow**::**displayMessage("Treatment Session Paused", TREATMENT\_SCREEN);** | Observing the following display under **Device View** |
| UC305 | User presses play button within 5 minutes of pressing pause | **USE CASE 3: Patient Presses Pause During Treatment Session** | *FUNCTION*  *MainWindow::on\_pauseButton\_clicked()*  *CALL*  *powerOffTimer->start( );*  CALL  MainWindow::on\_playButton\_clicked() | Clicking the following button under **Device View** |
| UC306 | Device resumes treatment session | **USE CASE 3: Patient Presses Pause During Treatment Session** | FUNCTION  MainWindow::on\_playButton\_clicked()  SIGNAL  MainWindow::playButtonPressed()  CALL  **Controller::resumeTreatmentSession()** | **N/A** |
| UC307 | Device display updates to show previous treatment progress | **USE CASE 3: Patient Presses Pause During Treatment Session** | FUNCTION  MainWindow::on\_playButton\_clicked()  CALL  MainWindow::displayMessage()  **ui->menuView->setFocus()** | Observing the following display under **Device View** |
|  |  |  |  |  |
| UC401 | Patient clicks menu button | **USE CASE 4: View Neureset Session Logs** | CALL  MainWindow::on\_menuButton\_clicked()  **ui->screenStack->setCurrentIndex(MENU\_SCREEN);**  **ui->menuView->setFocus();** | Clicking the following button under **Device View** |
| UC402 | Patient clicks arrow button until ‘Session Log’ option is reached | **USE CASE 4: View Neureset Session Logs** | *ui->screenStack->currentIndex() = MENU\_SCREEN*  CALL  MainWindow::on\_downButton\_clicked()  MainWindow::on\_upButton\_clicked()  *currentIndex.row() = SELECT\_SESSION\_LOGS* | Clicking the following buttons under **Device View** |
| UC403 | Patient clicks select button | **USE CASE 4: View Neureset Session Logs** | *currentIndex.row() == SELECT\_SESSION\_LOGS*  CALL  MainWindow::on\_selectButton\_clicked() | Clicking the following button under **Device View**    In the following display context |
| UC404 | Device display updates to show a list of the date and times of previous treatments | **USE CASE 4: View Neureset Session Logs** | FUNCTION  MainWindow::on\_selectButton\_clicked()  CALL  ui->screenStack->setCurrentIndex(SESSION\_LOG\_SCREEN)  SIGNAL  MainWindow::getPreviousSessionDatesForDevice()  CALL  Controller::getPreviousSessionDatesForDevice()  fileManager.getSessionDates()  SIGNAL  Controller::sessionDatesRetrievedForDevice()  CALL  MainWindow::slotDisplaySessionDatesOnDevice()  **ui->deviceSessionList->*setModel*(new QStringListModel(sessionDates, this))**  **ui->deviceSessionList->setFocus();** | Observing the following display (example) under **Device View** |
| UC405 | Patient pushes arrow buttons to scroll through the list | **USE CASE 4: View Neureset Session Logs** | *ui->screenStack->currentIndex() == SESSION\_LOG\_SCREEN*  CALL  MainWindow::on\_downButton\_clicked()  MainWindow::on\_upButton\_clicked() | Clicking the following buttons under **Device View** |
| UC406 | Patient clicks menu button to return to main menu | **USE CASE 4: View Neureset Session Logs** | CALL  MainWindow::on\_menuButton\_clicked()  **ui->screenStack->setCurrentIndex(MENU\_SCREEN);**  **ui->menuView->setFocus();** | Clicking the following button under **Device View** |
|  |  |  |  |  |
| UC501 | Patient clicks menu button | **USE CASE 5: Change Device Date and Time** | CALL  MainWindow::on\_menuButton\_clicked()  **ui->screenStack->setCurrentIndex(MENU\_SCREEN);**  **ui->menuView->setFocus();** | Clicking the following button under **Device View** |
| UC502 | Patient clicks arrows buttons until ‘Time and Date’ option is highlighted in main menu | **USE CASE 5: Change Device Date and Time** | *ui->screenStack->currentIndex() == MENU\_SCREEN*  CALL  MainWindow::on\_downButton\_clicked()  MainWindow::on\_upButton\_clicked()  *currentIndex.row() = SELECT\_TIME\_AND\_DATE* | Clicking the following buttons under **Device View** |
| UC503 | Patient presses select button | **USE CASE 5: Change Device Date and Time** | *currentIndex.row() == SELECT\_TIME\_AND\_DATE*  CALL  MainWindow::on\_selectButton\_clicked() | Clicking the following button under **Device View**    In the following display context |
| UC504 | Device display updates to show ‘Set Date & Time’ and the year/month/day it’s currently set to | **USE CASE 5: Change Device Date and Time** | FUNCTION  MainWindow::on\_selectButton\_clicked()  CALL  ui->screenStack->setCurrentIndex(SELECT\_TIME\_AND\_DATE)  **QDateTimeEdit::Section currentSection =**  **ui->dateTimeEdit->currentSection()** | Observing the following display (example) under **Device View** |
| UC505 | Patient presses the up/down buttons to scroll through months until desired month is found | **USE CASE 5: Change Device Date and Time** | FUNCTION  MainWindow::on\_selectButton\_clicked()    *QDateTimeEdit Widget*  *QDateTimeEdit::MonthSection = ui->dateTimeEdit->currentSection()* | Clicking the following buttons under **Device View**    In the following context (left most number highlighted) |
| UC506 | Patient presses select to proceed to day | **USE CASE 5: Change Device Date and Time** | CALL  MainWindow::on\_selectButton\_clicked() | Clicking the following buttons under **Device View**    In the following context (left most number highlighted) |
| UC507 | Patient presses up/down buttons to scroll through days until desired day is found | **USE CASE 5: Change Device Date and Time** | FUNCTION  MainWindow::on\_selectButton\_clicked()    *QDateTimeEdit Widget*  *QDateTimeEdit::DaySection = ui->dateTimeEdit->currentSection()* | Clicking the following buttons under **Device View**    In the following context (1+ from left most number highlighted) |
| UC508 | Patient presses select to proceed to year | **USE CASE 5: Change Device Date and Time** | CALL  MainWindow::on\_selectButton\_clicked() | Clicking the following buttons under **Device View**    In the following context (1+ from left most number highlighted) |
| UC509 | Patient presess up/down buttons to scroll through years until desired year is found | **USE CASE 5: Change Device Date and Time** | FUNCTION  MainWindow::on\_selectButton\_clicked()    *QDateTimeEdit Widget*  *QDateTimeEdit::HourSection = ui->dateTimeEdit->currentSection()* | Clicking the following buttons under **Device View**    In the following context (2+ from left most number highlighted) |
| UC510 | Patient presses select to proceed to hour | **USE CASE 5: Change Device Date and Time** | CALL  MainWindow::on\_selectButton\_clicked() | Clicking the following buttons under **Device View**    In the following context (2+ from left most number highlighted) |
| UC511 | Patient presses the up/down buttons to scroll through hours until desired hour is found | **USE CASE 5: Change Device Date and Time** | FUNCTION  MainWindow::on\_selectButton\_clicked()    *QDateTimeEdit Widget*  *QDateTimeEdit::HourSection = ui->dateTimeEdit->currentSection()* | Clicking the following buttons under **Device View**    In the following context (3+ from left most number highlighted) |
| UC512 | Patient presses select to proceed to minute | **USE CASE 5: Change Device Date and Time** | CALL  MainWindow::on\_selectButton\_clicked() | Clicking the following buttons under **Device View**    In the following context (3+ from left most number highlighted) |
| UC513 | Patient presses up/down buttons to scroll through minutes until desired minute is found | **USE CASE 5: Change Device Date and Time** | FUNCTION  MainWindow::on\_selectButton\_clicked()    *QDateTimeEdit Widget*  *QDateTimeEdit::MinuteSection = ui->dateTimeEdit->currentSection()* | Clicking the following buttons under **Device View**    In the following context (right most number highlighted) |
| UC514 | Patient presses menu button to return to main menu | **USE CASE 5: Change Device Date and Time** | CALL  MainWindow::on\_menuButton\_clicked() | Clicking the following button under **Device View** |
| UC515 | Device display updates to show menu options | **USE CASE 5: Change Device Date and Time** | FUNCTION  MainWindow::on\_menuButton\_clicked()  CALL  **ui->screenStack->setCurrentIndex(MENU\_SCREEN);**  **ui->menuView->setFocus();** | Observing the following display under **Device View** |