TRACEABILITY MATRIX

ID	Requirement	Related Use Case	Implemented By	Tested By	Description
1	The application interface contains buttons, display, and electrodes	N/A	MainWindow.ui	Run the program in Qt to observe the UI.	Using QT's built in user interface framework, the AED Plus was replicated. Also, all buttons are clickable with the mouse.
2	The AED Plus can be turned on by pressing the 'ON' button.	Use Case: Operate the AED Plus (UC1)	MainWindow	Press the 'ON' button.	The 'MainWindow' class contains the 'togglePowerButton' function which allows the user to start using the AED Plus.
3	The AED Plus can initiate a self-test to ensure the device is working properly and provide a visual and audible indication that it's operational.	Use Case: Operate the AED Plus (UC1), Use Case: AED Plus performs self-test (UC4)	MainWindow	Once the 'ON' button has been pressed, observe the console to see the outcome of the self-test. Click the 'Electrode connected' checkbox and observe the console to see a new outcome of the self-test.	The 'DataProcessor' and 'AED' classes both have a function called 'selfCheck()' which helps to conduct the self-test alongside the 'MainWindow' class. These classes check whether the battery level of the AED Plus is over 15% and the electrodes are connected. If both are true, the device passes the self-test.
4	The AED Plus can monitor a patient's heart rhythm through the electrodes to determine if there's a shockable rhythm present.	Use Case: Operate the AED Plus (UC1)	MainWindow, QCustomPlot, DataProcessor	Choose which scenario to simulate. Then, click the 'Pad connected' checkbox and observe the display to view the patient's heart rhythm.	When scenario is chosen to simulate by the user, the 'DataProcessor' class will set the heart amplitude to match the state the heart is in. The 'MainWindow' class then graphs the heartbeat onto the ECG using the 'generateHeartData' function.
5	The AED Plus will prompt the user to deliver a shock if there is a shockable rhythm present.	Use Case: Operate the AED Plus (UC1)	MainWindow, DataProcessor, AED	The console will provide a prompt to deliver a shock and the shock button will light up.	The 'DataProcessor' class is able to detect whether a shock is needed with the 'detectShockable' function. Whatever is returned from that function is sent to the 'startAnaylzing' function in the 'MainWinodw' class. This is where the user will be notified whether they need to provide the patient with a shock or not.

6	The user can deliver a shock by pressing the shock button (only if there is a shockable rhythm detected).	Use Case: Operate the AED Plus (UC1)	MainWindow, DataProcessor, AED	Press the shock button when its illuminated.	If the 'startAnaylzing' function in the 'MainWindow' class determines that the patient needs a shock, the 'givingShock' function is called and the shock button illuminates.
7	The AED Plus will provide voice and visual prompts to help the user perform actions (e.g., CPR, when to stand clear during defibrillation)	Use Case: Operate the AED Plus (UC1), Use Case: Provides Real-Time Feedback During CPR (UC3)	MainWindow	View the display or the console after the shock button has been pressed.	The 'doCpr' function in the 'MainWindow' class tells the user when to provide the patient with CPR.
8	The AED Plus will provide real-time feedback on the quality and rate of chest compressions during CPR	Use Case: Operate the AED Plus (UC1), Use Case: Provides Real-Time Feedback During CPR (UC3)	MainWindow, DataProcessor	Give CPR to the patient by clicking '0.5 inches', '1 inch' or '2 inches' buttons and view the display to see feedback.	The 'cprPrompt' function in the 'MainWindow' class provides prompts to the user to help with CPR. These prompts are based on the buttons the user presses during CPR and whether the patient is an adult or a child. The 'detectPad' function in the 'DataProcessor' class is what detects whether the patient is an adult or a child.
9	The AED Plus disables its shock feature if there's no shockable rhythm present in the patient.	Use Case: Prevent Accidental Shocks (UC2)	MainWindow, DataProcessor, AED	View the shock button. When it's disabled, it will no longer be illuminated.	If the 'startAnaylzing' function in the 'MainWindow' class determines that there isn't a shockable rhythm in the patient, the 'givingShock' function is able to disable the shock button.
10	The AED Plus will play an audio message informing the user that shocks are not required.	Use Case: Prevent Accidental Shocks (UC2)	MainWindow	View the shock button. When it's disabled, it will no longer be illuminated.	If the 'startAnaylzing' function in the 'MainWindow' class determines that there isn't a shockable rhythm in the patient, it will print a message to the console saying that and the 'givingShock' function will

				no longer illuminate the shock button.
11	The AED Plus detects CPR compressions carried out by the user.	Use Case: Provides Real-Time Feedback During CPR (UC3)	MainWindow	Give CPR to the patient by clicking '0.5 inches', '1 inch' or '2 inches' buttons and view the display to see feedback.	The 'cprPush' function keeps track of how many compressions the user does. It also calls the 'cprBarDrop' function to show how strong the compressions are.
12	The AED Plus provides real-time feedback on CPR compressions and displays the compression depth on the LCD screen.	Use Case: Provides Real-Time Feedback During CPR (UC3)	MainWindow	Give CPR to the patient by clicking '0.5 inches', '1 inch' or '2 inches' buttons and view the display to see feedback.	The 'cprPrompt' function in the 'MainWindow' class provides feedback to the user on their CPR compressions. This feedback is based on the buttons the user presses during CPR and whether the patient is an adult or a child. The 'detectPad' function in the 'DataProcessor' class is what detects whether the patient is an adult or a child.
13	The AED Plus will tell the user when they can stop providing CPR to the patient.	Use Case: Provides Real-Time Feedback During CPR (UC3)	MainWindow	After providing CPR, view the console.	The 'cprPrompt' function in the 'MainWindow' class provides feedback to the user on their CPR compressions.
14	The AED Plus continuously monitors the battery level while its operating.	Use Case: Operate the AED Plus (UC1)	MainWindow, Data Processor, AED	View the top-right of the GUI where the remaining battery level is displayed.	The 'DataProcessor' class keeps track of the device's battery level. The 'consumingBattery' function in the 'MainWindow' class also keeps track of the battery level and how much is being consumed while it's in operation.
15	When trying to turn on the AED Plus, if the battery level is 0%, it will not turn on.	Use Case: Operate the AED Plus (UC1)	MainWindow, Data Processor, AED	Change the remaining battery level to '0.00' and view the message displayed on the console or try to simulate a scenario with the UI.	The 'DataProcessor' class keeps track of the device's battery level. The 'changeBatteryLeft' function in the 'MainWindow' class checks whether the battery level of the device is over 0%. If it's not, it will call the 'togglePowerButton' function and stop all

					processes from starting/continuing.
16	If the battery level falls below 20%, while in use, the AED Plus detects the low battery condition and provides the user with a visual prompt alerting them of the low battery condition.	Use Case: Operate the AED Plus (UC1)	MainWindow, Data Processor, AED	Change the remaining battery level to '20.00' and view the message displayed on the console.	The 'DataProcessor' class keeps track of the device's battery level. The 'changeBatteryLeft' function in the 'MainWindow' class checks whether the battery level of the device is over 20%. If it's not, it will call the 'togglePowerButton' function and stop all processes from starting/continuing.