

Textual Explanation of the Design

A design implementation that we have used in the development of the neureset device is the use of object oriented programming. Object oriented programming is present in our code in the forms of Classes and Objects, Encapsulation, and Abstraction. The code is split up into 4 main classes. EEG: to handle operations related to the 21 electrodes that make up the EEG headset, Menu: to represent a menu object that is used to display information and interact with the user, Session: to store information about the treatment sessions, and MainWindow: serves as the control class of the application, fulfilling most of the neureset device requirements. These classes work together to fulfill the Classes and Objects principle in object oriented programming. They also use Abstraction. Internal implementation details of the device are hidden, providing only simple functions that increase efficiency. For example, the Session class hides the complex details and provides 3 simple functions `startSession()`, `pauseSession()` and `stopSession()` to be used in the MainWindow class. Encapsulation is also present throughout the code by preventing the access of object details from other classes. This is done by creating private variables and functions, and using getters and setters whenever necessary. Another design implementation that is present in the code is the use of the observer pattern. The MainWindow class contains multiple slots that serve as observers. These functions are notified when changes occur on the device. Some of the changes that the slots observe are buttons in the menu being clicked, signals sent by the session class to control the lights during treatment, and user input to change the date and time. Using the two designs: object oriented programming and the observer pattern helped provide a simple and effective implementation for the neureset device.