AFRL-RH-WP-TR-2020-0063

COG-PACK™ DASHBOARD USERS MANUAL

Mr. Allen W. Dukes
2Lt. Matt T. Rommel
Airman Biosciences Division

Mr. Courtney J. Downs
Mr. Scott J. Duberstein
Mr. Ethan B. Blackford
Ball Aerospace & Technologies

May 2020
Interim Report

AIR FORCE RESEARCH LABORATORY
711TH HUMAN PERFORMANCE WING,
AIRMAN SYSTEMS DIRECTORATE,
WRIGHT-PATTERSON AIR FORCE BASE, OH 45433
AIR FORCE MATERIEL COMMAND
UNITED STATES AIR FORCE

NOTICE AND SIGNATURE PAGE

Using Government drawings, specifications, or other data included in this document for any purpose other than Government procurement does not in any way obligate the U.S. Government. The fact that the Government formulated or supplied the drawings, specifications, or other data does not license the holder or any other person or corporation; or convey any rights or permission to manufacture, use, or sell any patented invention that may relate to them.

This report was cleared for public release by 88th Air Base Wing Public Affairs Office and is available to the general public including foreign nationals. Copies may be obtained from the Defense Technical Information Center (DTIC) (http://www.dtic.mil).

AFRL-RH-WP-TR-2020-0063 HAS BEEN REVIEWED AND IS APPROVED FOR PUBLICATION.

FRANCK.DOUGLAS.L.1230132255 Digitally signed by FRANCK.DOUGLAS.L.1230132255 Date: 2021.01.06 15:50:53 -05'00'

DOUGLAS L. FRANCK, DR-III, DAF
Chief, Performance Optimization Branch
Airman Biosciences Division

MCKINLEY.RICHARD.A.1262239710 Digitally signed by MCKINLEY.RICHARD.A.1262239710 Date: 2020.09.03 10:08:15 -04'00'

R. ANDY MCKINLEY, PhD
Core Research Area Lead
Performance Optimization Branch
Airman Biosciences Division

This report is published in the interest of scientific and technical information exchange, and its publication does not constitute the Government's approval or disapproval of its ideas or findings.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. REPORT DATE *(DD-MM-YY)*	2. REPORT TYPE	3. DATES COVERED *(From - To)*
04 08 20	Interim	January 2019 to August 2020

4. TITLE AND SUBTITLE
COG-PACK™ Dashboard Users Manual

5a. CONTRACT NUMBER
FA8650-16-C-6610

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER
62202F

6. AUTHOR(S)
COURTNEY J. DOWNS+, ALLEN W. DUKES*, ETHAN B. BLACKFORD+, SCOTT J. DUBERSTEIN+, MATT T. ROMMEL*

5d. PROJECT NUMBER
5329

5e. TASK NUMBER
08

5f. WORK UNIT NUMBER
H0KG (53290819)

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
+Ball Aerospace & Technologies
2675 Presidential Drive, Fairborn OH 45324

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

*Air Force Materiel Command
Air Force Research Laboratory
711 Human Performance Wing
Airman Systems Directorate
Airman Biosciences Division
Performance Optimization Branch
Cognitive Neuroscience Section
Wright-Patterson AFB, OH 45433

Air Force Materiel Command**
Air Force Research Laboratory
711 Human Performance Wing
USAF School of Aerospace Medicine
Wright-Patterson AFB, OH 45433

10. SPONSORING/MONITORING AGENCY ACRONYM(S)
711HPW/RHBC

11. SPONSORING/MONITORING AGENCY REPORT NUMBER(S)
AFRL-RH-WP-TR-2020-0063

12. DISTRIBUTION/AVAILABILITY STATEMENT
DISTRIBUTION A. Approved for public release,

13. SUPPLEMENTARY NOTES
88ABW-2020-2718, cleared 28 August 2020

14. ABSTRACT The design of the COG Pack Dashboard allows a user to set up, configure, and manage multiple Sensors by providing a standard look and feel for each Sensor. Generally, when manufacturers develop hardware sensors, they provide a custom application designed to only work with their particular device. From an integration standpoint, this provides a high barrier of entry for researchers and users when an experiment requires multiple disparate sensors. Each independent application often provides unique functionality for actions such as saving files, managing sensor state connectivity, and observing the data-streams in real-time. One of the objectives of the COG Pack Dashboard is to simplify these steps and consolidate them into a single interface. Additionally, by leveraging a database back-end for data storage, COG Pack automatically manages data timestamps and organization. By not requiring a user to execute multiple mouse actions and keyboard inputs for file management, COG Pack streamlines this process and reduces the potential for 'fat-fingered' input errors.

The goal of this document is to provide a user or researcher with the instructions they need to work with the COG Pack Dashboard. This document contains descriptions of the processes and procedures required to interact with the supported Sensors and Signals.

15. SUBJECT TERMS COG Pack™, Dashboard, User Interface, UJI, Manual

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT:	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON (Monitor)
a. REPORT	b. ABSTRACT	c. THIS PAGE			Armando Soto
U	U		SAR	18	19b. TELEPHONE NUMBER *(Include Area Code)* N/A

COG PACK™ DASHBOARD

COURTNEY J. DOWNS, ALLEN W. DUKES, ETHAN B. BLACKFORD, SCOTT J. DUBERSTEIN, MATT T. ROMMEL

User's Manual

TABLE OF CONTENTS

LIST OF FIGURES

1.0 OVERVIEW

The design of the COG Pack Dashboard allows a user to set up, configure, and manage multiple Sensors by providing a standard look and feel for each Sensor. Generally, when manufacturers develop hardware sensors, they provide a custom application designed to only work with their particular device. From an integration standpoint, this provides a high barrier of entry for researchers and users when an experiment requires multiple disparate sensors. Each independent application often provides unique functionality for actions such as saving files, managing sensor state connectivity, and observing the data-streams in real-time. One of the objectives of the COG Pack Dashboard is to simplify these steps and consolidate them into a single interface. Additionally, by leveraging a database back-end for data storage, COG Pack automatically manages data timestamps and organization. By not requiring a user to execute multiple mouse actions and keyboard inputs for file management, COG Pack streamlines this process and reduces the potential for 'fat-fingered' input errors.

The goal of this document is to provide a user or researcher with the instructions they need to work with the COG Pack Dashboard. This document contains descriptions of the processes and procedures required to interact with the supported Sensors and Signals.

2.0 STARTING THE COG PACK DASHBOARD

The COG Pack system requires two main software components for execution: the back-end web service and the front-end Dashboard UI. These are the necessary components required, regardless of the configuration. In the most commonly deployed configuration, a single icon on the desktop loads both components. (Figure 1.)

Figure 1 - COG Pack Desktop Shortcut

3.0 ASSIGNING SENSORS TO PARTICIPANTS

After loading the COG Pack processes, the initialization screen is presented. (Figure 2.) This screen presents the first step in the process required for set up and configuration, allowing a user to associate particular Sensors and Signals with a participant.

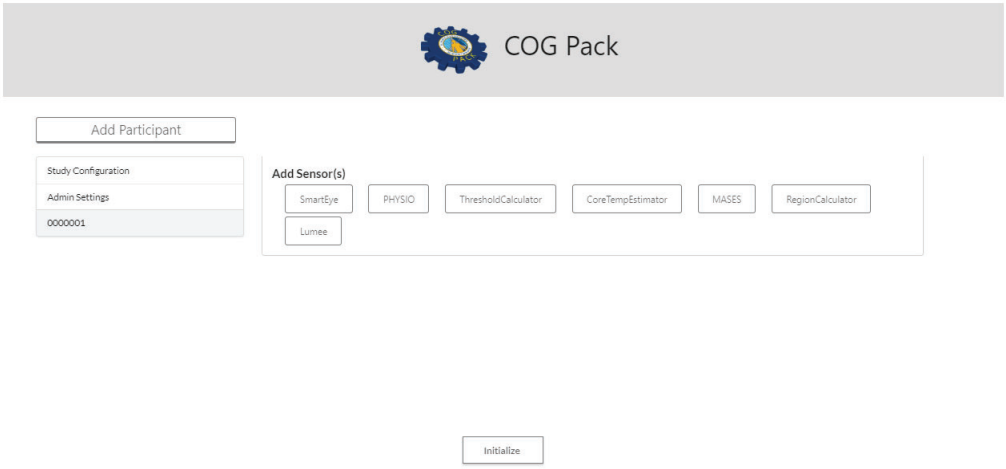

Figure 2 - COG Pack Initialization Screen

To begin the association, complete the following steps:

1. Add a participant by clicking the “Add Participant” button on the left.
    a. By default, the system generates a Participant ID
2. Click the generated Participant ID in the list.

3. Select the desired type of Sensor to assign to this participant by selecting the name of the Sensor from the array of available Sensors found in the “Add Sensor(s)” section.
4. Enter the values and settings required for each Sensor.

Proceed to the following sections for examples showing how to configure the individual Sensors.

3.1 HSI PHYSIO Configuration

Figure 3 displays the card for the configuration of an HSI PHYSIO Sensor. To assign an HSI PHYSIO Sensor to a participant, first label it with a Sensor ID by entering a value into the Sensor ID text field. Then select all of the Signals to capture. These selected Signals are then available to the Signal processing pipeline for viewing, logging, and further analysis. For this Sensor, COG Pack has provided the ability to toggle between live and simulated data. To configure this option, toggle the button labeled “Emulated” to tell the system if this device is live or simulated. Finally, the user has the option to attach additional Signal processing algorithm(s). (See also Attaching Additional Signal Processing Steps)

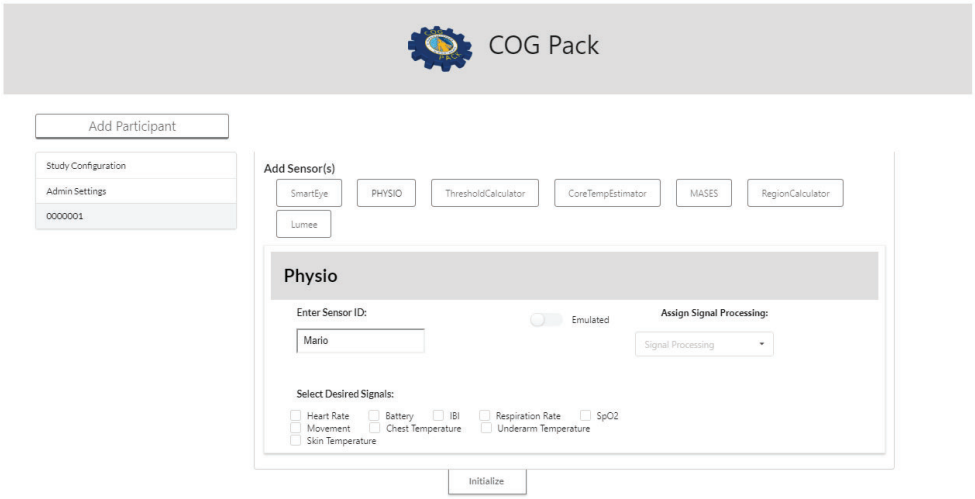

Figure 3 - Configuring an HSI PHYSIO Sensor

3.2 Profusa Lumee Configuration

To add a Profusa Lumee Sensor to a participant, locate the 4 character alpha-numeric ID on the Sensor, and enter that value into the Sensor ID text field. Then select all of the signals to capture. Finally, attach any additional Signal processing algorithms. (See also Attaching

Additional Signal Processing Steps) Figure 4 demonstrates the configuration of a Profusa Lumee Sensor.

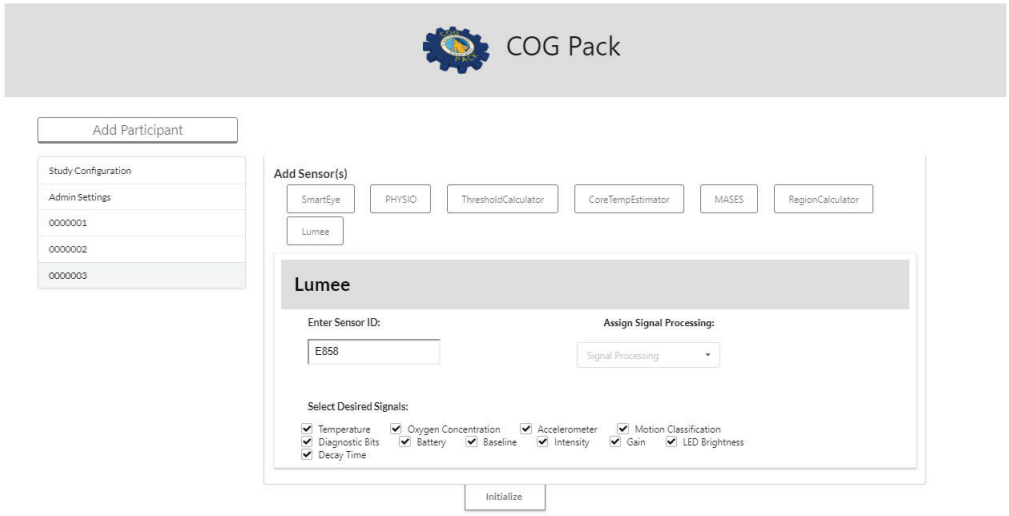

Figure 4 - Configuring a Profusa Lumee Sensor

3.3 Intelligent Optical Systems (IOS) MASES Configuration

Adding an IOS MASES Sensor follows the same process as described for other Sensors. The IOS MASES Sensor requires only a few configuration options. First, a Sensor ID to uniquely describe this instance, then the IP Address of the interface connected to the IOS MASES Sensor. Additionally, specify the port number that the IOS MASES device uses to send data. COG Pack also supports emulation for this Sensor and can be toggled with the “Emulated” button. Finally, select all of the Signals to capture visually by clicking their respective check boxes. Figure 5 displays the configuration of an IOS MASES Sensor.

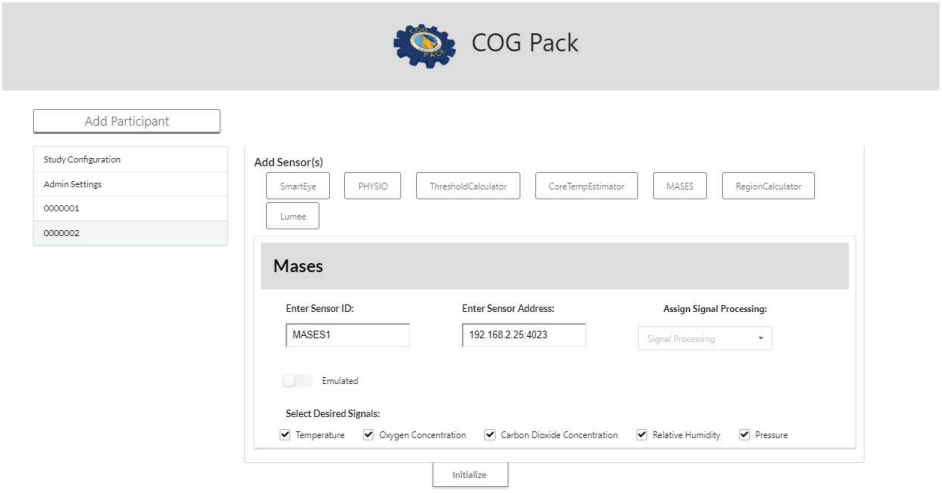

Figure 5 - Configuring an IOS MASES Sensor

3.4 Smart Eye Pro Configuration

Currently, Smart Eye Pro is the most complicated Sensor to configure in COG Pack. To start, fill out all of the listed fields below with the settings formatted as described in Table 1.

Table 1. Smart Eye Settings, Descriptions, and Data Types

Setting Name	Description	Data Type
Sensor ID	Unique Identifier for this Smart Eye Instance	String
IP Address	IP Address of the COG Pack system connected to Smart Eye	String
Machine Name	Name of the COG Pack system	String
Port Number	Port Number used by Smart Eye to connect to COG Pack	Integer
World Model File	Full path to *.sew file used by Smart Eye	String
Minimum Fixation Dwell Time	For AdvancedFixations calculations, define the minimum number of milliseconds required for a Fixation	Double
Vergence Angle	For AdvancedFixations calculations, define the Visual Angle, in degrees, describing the maximum dispersion of points possible for a Fixation	Double
Monitor Size in Pixels	A comma-separated list of 4 Doubles that define the pixel space of a single monitor. DEPRECATED – replaced by values in the World Model File	DEPRECATED
Monitor Size in Meters	A comma-separated list of 4 Doubles that define the physical dimensions of a single monitor. DEPRECATED – replaced by values in the World Model File	DEPRECATED
Sample Rate	The frequency at which Smart Eye is configured. Most commonly 60 or 120 Hz.	Integer
Playback File	Full path to a *.sme file recorded previously by COG Pack software to replay previously recorded binary data. (Leave empty for live data capture)	String

These are the settings required for the configuration of the interface between COG Pack and Smart Eye. For additional settings of the visual elements of COG Pack, see the report titled "COG PACK Eye/Head/Gaze/Heatmap 3D Display Documentation". Figure 6 shows the configuration of a Smart Eye Sensor.

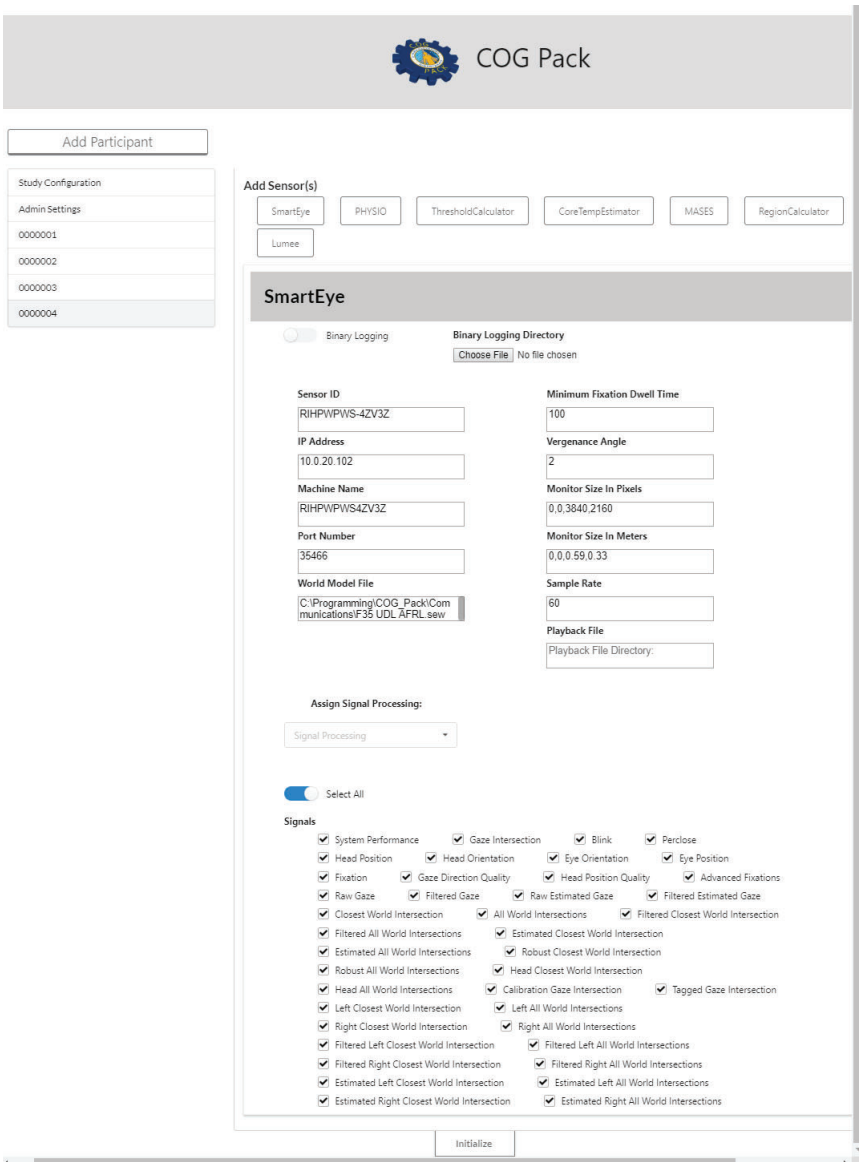

Figure 6 - Configuring a Smart Eye Sensor

4.0 ATTACHING ADDITIONAL SIGNAL PROCESSING STEPS

Each of the above cards for a Sensor's configuration has an option to add additional signal processing algorithm(s). To view all of the available algorithms for the selected Sensor, select the drop-down directly under the title **Assign Signal Processing (Figure 7)**.

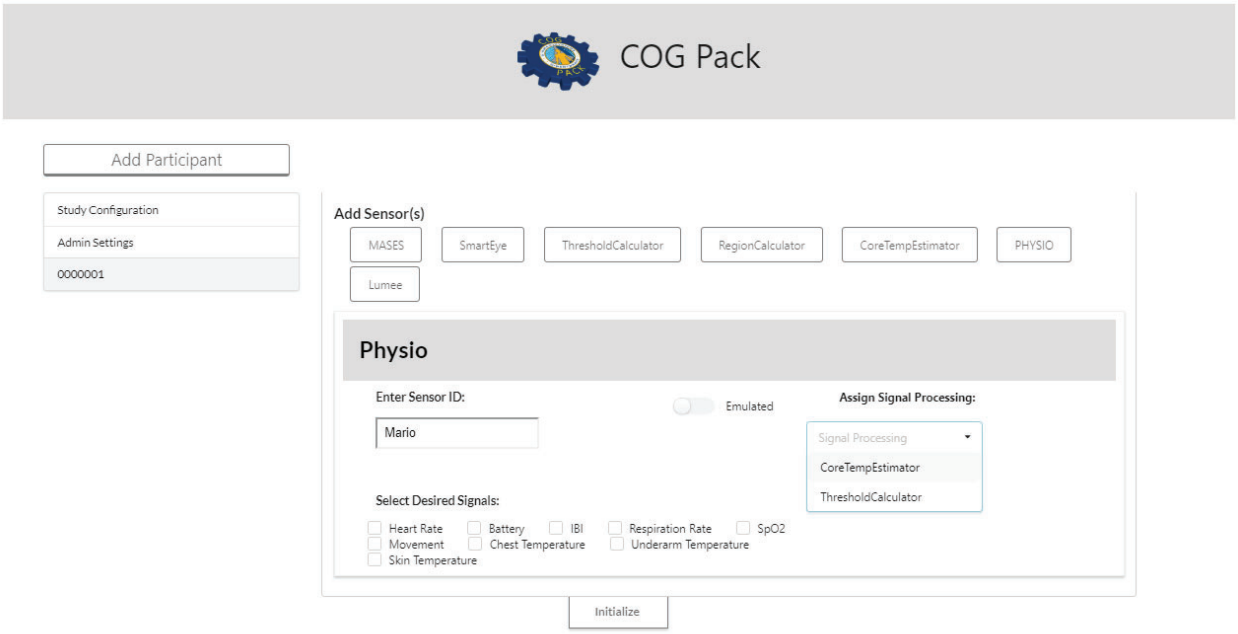

Figure 7 – Assigning Signal Processing

4.1 Attaching a Threshold Calculator

To attach "Threshold Calculator" to a Sensor you must perform the following actions:

1. Select the Signal to Threshold for the current Sensor.
2. Select the "Threshold Calculator" option for Signal processing as shown in **Figure 8**.
3. Fill in the fields with the desired range labels and values.
4. Click the button labeled "Attach".

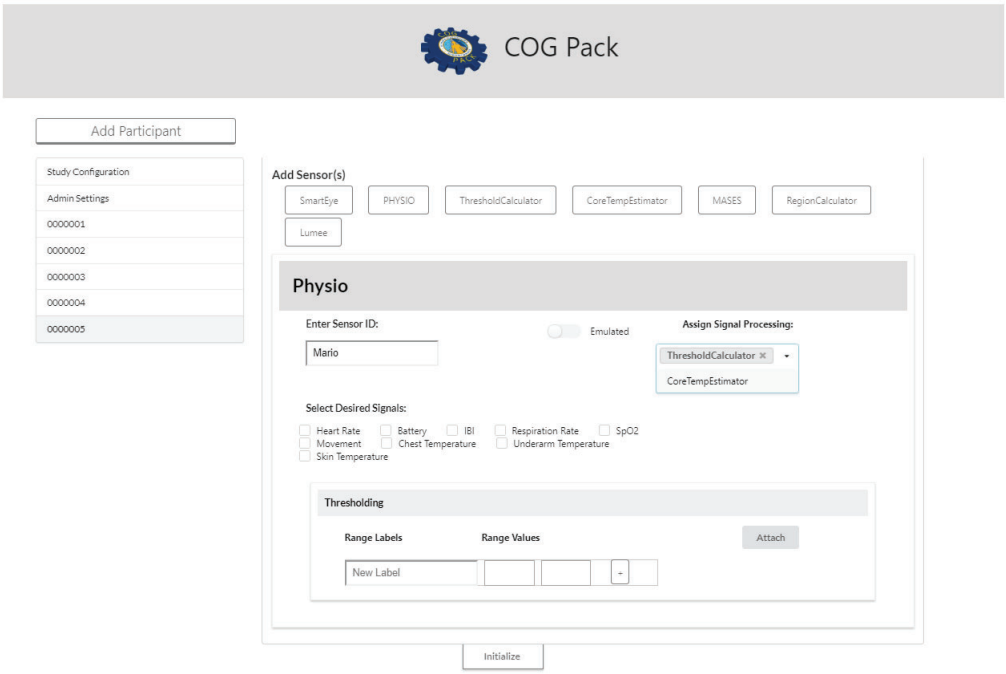

Figure 8 – Assigning a Threshold Calculator

For example, to set the following values for the Signal "Respiration Rate," enter the values in the same order as shown in **Figure 9.**

Low: < 60 **Medium**: >= 60 and < 120 **High**: >= 120

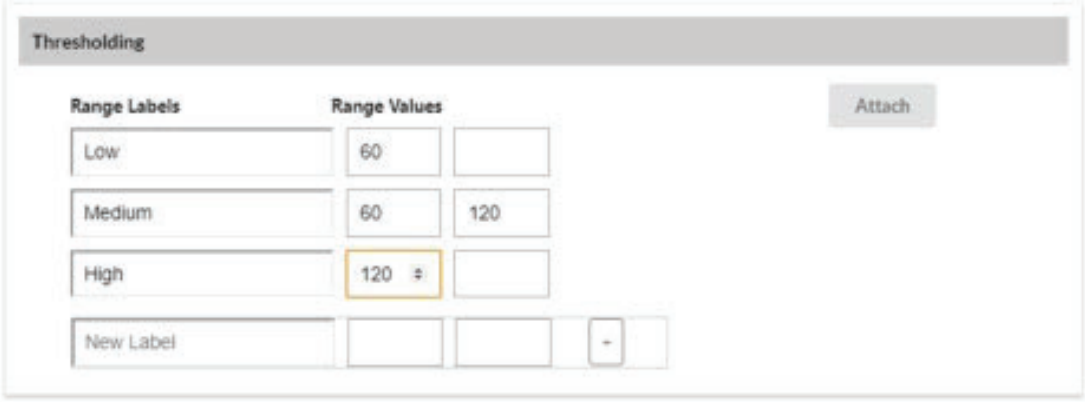

Figure 9 – Assigning Threshold Ranges to a Signal

4.2 Attaching a “Core Temperature Estimator”

The process for attaching a “Core Temperature Estimator” follows a similar method:

1. Select the Signal (Heart Rate) used to derive the Core Temperature Estimate (Figure 10.)
2. Select the “Core Temperature Estimator” option for Signal processing.
3. Fill in the value for “Starting Core Temperature” as seen in Figure 11.
4. Click the button labeled “Attach”.

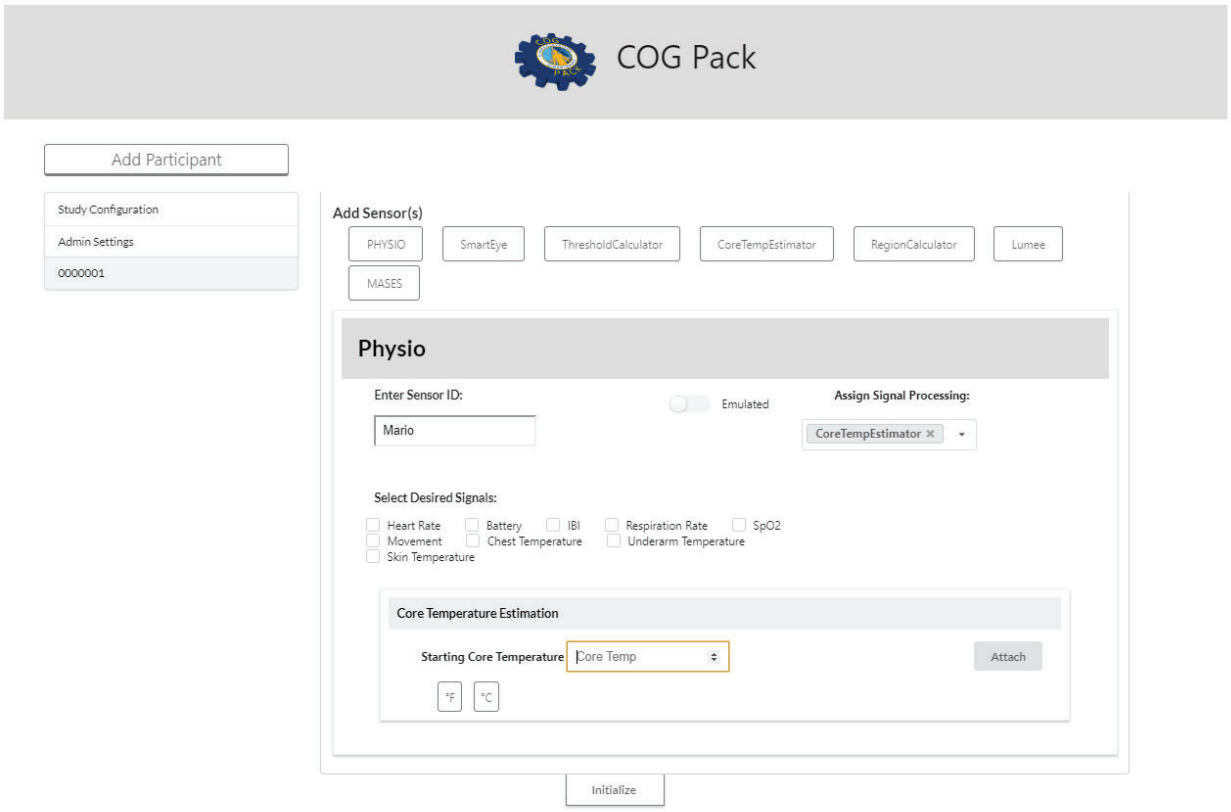

Figure 10 – Assigning a Core Temperature Estimator

Figure 11 – Configuring the Starting Temperature for Core Temperature Estimation

5.0 GRAPHS AND VISUALIZATIONS

When Sensor and Signal configuration is complete, click the button at the bottom of the screen labeled **Initialize**. This action will start the communcation process between the front-end UI and back-end **COG_Pack.Communcations.exe** processes. All of the settings and configurations will be sent to the backend, and the Sensors will begin connecting. The UI will transition to a screen that renders all of the graphs configured in the previous screen. **Figure 12** is an example of a single participant and the output of the configured Sensors and Signals. This example contains the output of various Threshold Signals derived from the Signals produced by a single HSI PHYSIO Sensor and 2 separate Profusa Lumee Sensors.

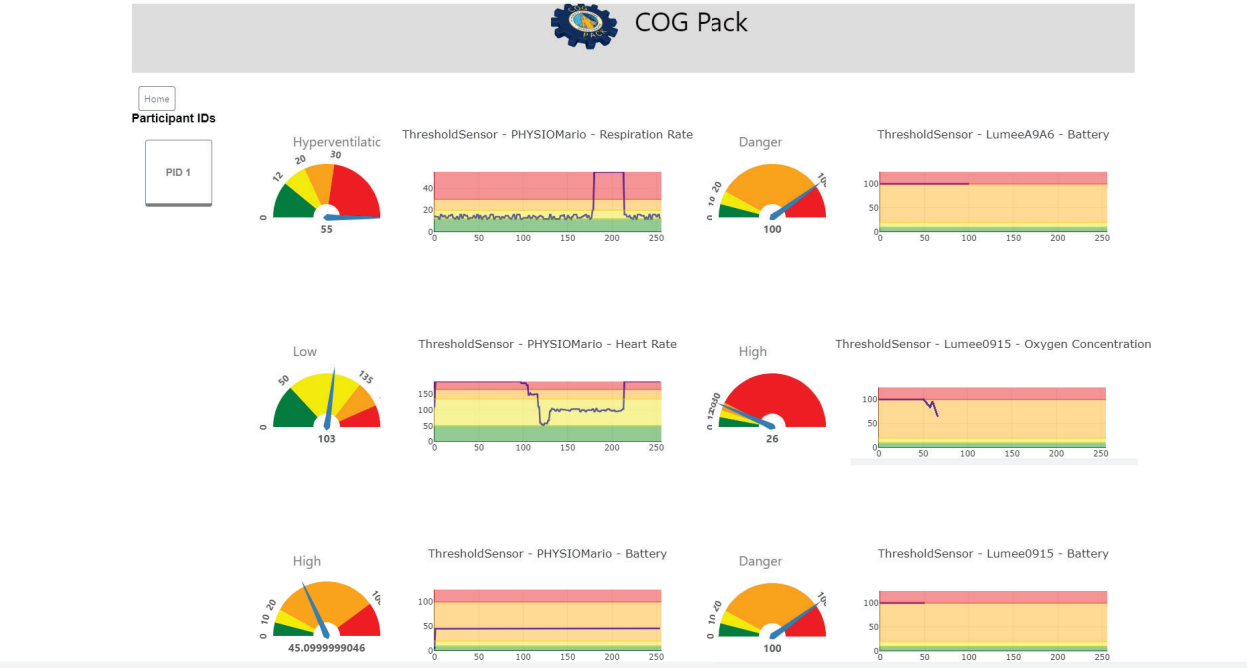

Figure 12 – Graphical Display of Signals

The different colored sections on these graphs and gauges represent the configured values for each Signal's respective "Threshold Calculator" range values. In the COG Pack Threshold Signal definition, these values can change over time, and the graphs and gauges react accordingly.

6.0 LOADING AND SAVING CONFIGURATIONS

Another very important feature of this application is its ability to save the current setup and suit of sensors that you have configured for a group of participants into a single .JSON file. This file captures the settings and setup that you have entered into the application and can represent configuration for a specific trial.
For example, if a researcher wishes to use COG Pack for a study and it always requires the same 5 participants with the same set of sensors then you can just set it up in the application one time and hit save. Then, you will be able to load that saved file into the application later and it will restore the setup without you having to go through and reenter all of that data every time.

Therefore, on the main screen of the COG Pack Dashboard is an option on the left labeled **Save/Load (Figure 13).**

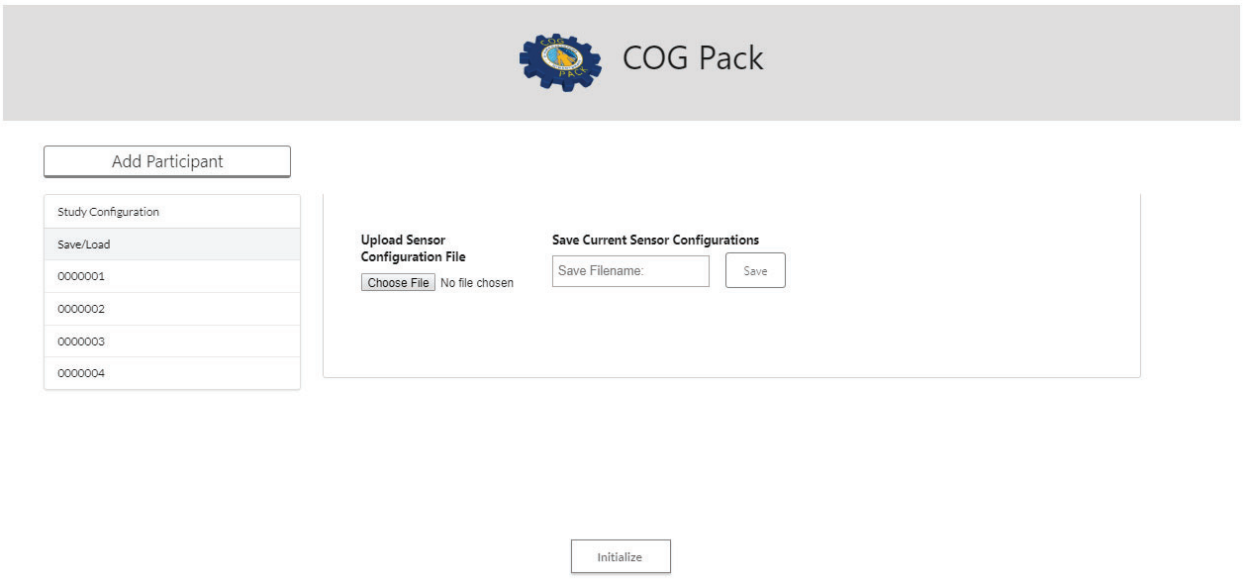

Figure 13 – Dashboard Configuration Save and Recall

Loading a Configuration

The process for loading an existing configuration for the Dashboard is the following:

1. Navigate to the Save/Load tab on the left
2. Click the button labeled "Choose File"

3. Navigate to the .JSON file that you've previously saved
4. Click the open button as seen in **Figure 14** below.
5. Click the Initialize button.

After following the above steps and clicking the initialize button at the bottom of the screen you will see the screen displaying all of the visualizations and graphs associated with the sensors that were saved to the save file.

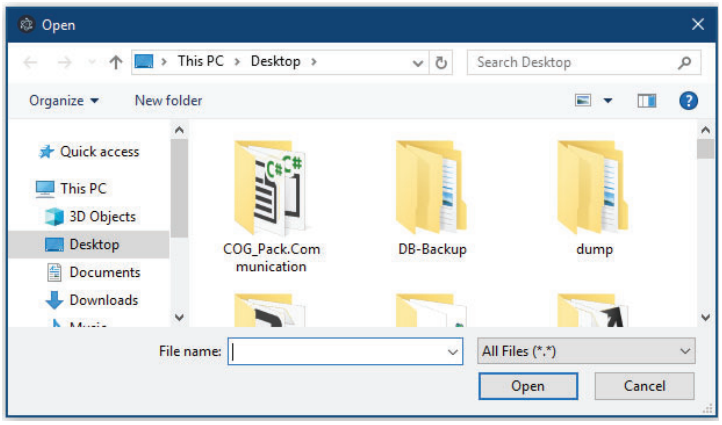

Figure 14 – Loading a Dashboard Configuration

Saving a new Configuration

To save your existing dashboard configuration, do the following:

1. Configure the sensors for your participants
2. Navigate to the Save/Load page in the application
3. Enter a name for the configuration file that will be generated
4. Click the save button

7.0 CONCLUSION

The COG Pack dashboard is a centralized place for multiple sensor configurations and visualizations. Our unique approach of using "Reflection Based" user interface generation allows this application to be very versatile when it comes to adding even more disparate sensors to its existing library of integrated sensors. The choice that we made to use Electron along with the React.js web framework allows this application to also be able to run on many different operating systems. We are constantly working on improving the design and user experience of the Dashboard to meet the needs and suggestions of our user base.