|  |  |
| --- | --- |
| Project | **Specification of Sensor Interface for Cyber and Physical World**  <<https://sagroups.ieee.org/2888.1/> **>** |
| Title | **Application programming interfaces for smart biosensors** |
| DCN | **2888-21-0069-00-0001** |
| Date Submitted | **Oct. 13th, 2021** |
| Source(s) | Sang-Kyun Kim, [goldmunt@gmail.com](mailto:goldmunt@gmail.com) (Myongji University)  Min Hyuk Jeong, [jmh8900@gmail.com](mailto:jmh8900@gmail.com) (Myongji University) |
| Re: |  |
| Abstract | This contribution illustrates the application programming interfaces for biosensors. |
| Purpose | To start discussion on purpose of the standard |
| Notice | This document has been prepared to assist the IEEE 2888 Working Group. It is offered as a basis for discussion and is not binding on the contributing individual(s) or organization(s). The material in this document is subject to change in form and content after further study. The contributor(s) reserve(s) the right to add, amend or withdraw material contained herein. |
| Release | The contributor grants a free, irrevocable license to the IEEE to incorporate material contained in this contribution, and any modifications thereof, in the creation of an IEEE Standards publication; to copyright in the IEEE’s name any IEEE Standards publication even though it may include portions of this contribution; and at the IEEE’s sole discretion to permit others to reproduce in whole or in part the resulting IEEE Standards publication. The contributor also acknowledges and accepts that IEEE 2888 may make this contribution public. |
| Patent Policy | The contributor is familiar with IEEE patent policy, as stated in [Section 6 of the IEEE-SA Standards Board bylaws](http://standards.ieee.org/guides/opman/sect6.html#6.3) <[http://standards.ieee.org/guides/bylaws/sect6-7.html#6](http://127.0.0.1:4664/cache?event_id=757737&schema_id=1&s=5X0vID10lu_E6yrIkWkNd4Wz2H8&q=hancock)> and in *Understanding Patent Issues During IEEE Standards Development* <http://standards.ieee.org/board/pat/faq.pdf> |

# Introduction

This contribution illustrates the application programming interfaces for biosensors.

## Blood pressure sensor

Table 1 – Blood pressure sensor API

|  |  |
| --- | --- |
| Nested Classes | |
| Modifier and Type | Method and Description |
|  |  |
| Constructor | |
| Constructor and Description | |
| BloodPressure() | |
| *Default constructor.* | |
|  | |
| BloodPressure(String id) | |
|  | |
| BloodPressure(String id, String serverIPAddress, integer serverPort) | |
|  | |
| Fields | |
| Modifier and Type | Field and Description |
|  |  |
| Methods | |
| Modifier and Type | Method and Description |
| JSONObject | getBloodPressureSensorData() |
|  | *This function returns sensor data from a blood pressure sensor in JSON format.* |
|  |  |

## Heart rate sensor

Table 2 – Heart rate sensor API

|  |  |
| --- | --- |
| Nested Classes | |
| Modifier and Type | Method and Description |
|  |  |
| Constructor | |
| Constructor and Description | |
| HeartRate() | |
| *Default constructor.* | |
|  | |
| HeartRate(String id) | |
|  | |
| HeartRate(String id, String serverIPAddress, integer serverPort) | |
|  | |
| Fields | |
| Modifier and Type | Field and Description |
|  |  |
| Methods | |
| Modifier and Type | Method and Description |
| JSONObject | getHeartRateSensorData() |
|  | *This function returns sensor data from a heart rate sensor in JSON format.* |
|  |  |