|  |  |
| --- | --- |
| Project | **Specification of Sensor Interface for Cyber and Physical World**<<https://sagroups.ieee.org/2888.1/> **>** |
| Title | **Application programming interfaces for environment-related sensors** |
| DCN | **2888-22-0012-00-0001** |
| Date Submitted | **Jan 31, 2022** |
| Source(s) | Tai-Gil Kwon tgkwon@keti.re.kr (Korea Electronics Technology Institute),Changseok Yoon csyoon@keti.re.kr (Korea Electronics Technology Institute),Tae-Beom Lim tblim@keti.re.kr (Korea Electronics Technology Institute),Kwanghyun Ro khrho@hansung.ac.kr (Hansung University) |
| Re: |  |
| Abstract | This contribution illustrates the application programming interfaces for environment-related sensors. |
| Purpose | To start a discussion on the 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 environment-related sensors.

## Rain sensor

Table 1 – Rain sensor API

|  |
| --- |
| Nested Classes |
| Modifier and Type | Method and Description |
|  |  |
| Constructor |
| Constructor and Description |
| Rain() |
| *Default constructor.* |
|  |
| Rain(String id) |
|  |
| Rain(String id, String serverIPAddress, integer serverPort) |
|  |
| Fields |
| Modifier and Type | Field and Description |
|  |  |
| Methods |
| Modifier and Type | Method and Description |
| JSONObject | getRainSensorData() |
|  | *This function returns sensor data from the rain sensor in JSON format.* |
|  |  |

## Insolation sensor

Table 2 – Insolation sensor API

|  |
| --- |
| Nested Classes |
| Modifier and Type | Method and Description |
|  |  |
| Constructor |
| Constructor and Description |
| Insolation() |
| *Default constructor.* |
|  |
| Insolation(String id) |
|  |
| Insolation(String id, String serverIPAddress, integer serverPort) |
|  |
| Fields |
| Modifier and Type | Field and Description |
|  |  |
| Methods |
| Modifier and Type | Method and Description |
| JSONObject | getInsolationSensorData() |
|  | *This function returns sensor data from the insolation sensor in JSON format.* |
|  |  |

## Soil moisture sensor

Table 3 –Soil moisture sensor API

|  |
| --- |
| Nested Classes |
| Modifier and Type | Method and Description |
|  |  |
| Constructor |
| Constructor and Description |
| SoilMoisture() |
| *Default constructor.* |
|  |
| SoilMoisture(String id) |
|  |
| SoilMoisture(String id, String serverIPAddress, integer serverPort) |
|  |
| Fields |
| Modifier and Type | Field and Description |
|  |  |
| Methods |
| Modifier and Type | Method and Description |
| JSONObject | getSoilMoistureSensorData() |
|  | *This function returns sensor data from the soil moisture sensor in JSON format.* |
|  |  |

## Tensiometer sensor

Table 4 – Tensiometer sensor API

|  |
| --- |
| Nested Classes |
| Modifier and Type | Method and Description |
|  |  |
| Constructor |
| Constructor and Description |
| Tensiometer() |
| *Default constructor.* |
|  |
| Tensiometer(String id) |
|  |
| Tensiometer(String id, String serverIPAddress, integer serverPort) |
|  |
| Fields |
| Modifier and Type | Field and Description |
|  |  |
| Methods |
| Modifier and Type | Method and Description |
| JSONObject | getTensiometerSensorData() |
|  | *This function returns sensor data from the tensiometer sensor in JSON format.* |
|  |  |

## Electrical conductivity sensor

Table 5 –Electrical conductivity sensor API

|  |
| --- |
| Nested Classes |
| Modifier and Type | Method and Description |
|  |  |
| Constructor |
| Constructor and Description |
| ElectricalConductivity() |
| *Default constructor.* |
|  |
| ElectricalConductivity(String id) |
|  |
| ElectricalConductivity(String id, String serverIPAddress, integer serverPort) |
|  |
| Fields |
| Modifier and Type | Field and Description |
|  |  |
| Methods |
| Modifier and Type | Method and Description |
| JSONObject | getElectricalConductivitySensorData() |
|  | *This function returns sensor data from the electrical conductivity sensor in JSON format.* |
|  |  |

## Acidity sensor

Table 6 –Acidity sensor API

|  |
| --- |
| Nested Classes |
| Modifier and Type | Method and Description |
|  |  |
| Constructor |
| Constructor and Description |
| Acidity() |
| *Default constructor.* |
|  |
| Acidity(String id) |
|  |
| Acidity(String id, String serverIPAddress, integer serverPort`) |
|  |
| Fields |
| Modifier and Type | Field and Description |
|  |  |
| Methods |
| Modifier and Type | Method and Description |
| JSONObject | getAciditySensorData() |
|  | *This function returns sensor data from the acidity sensor in JSON format.* |
|  |  |