APPLICATION PROGRAM INTERFACE (API) FOR AIRASSURE WEB PM2.5 SOFTWARE

Introduction

TSI has released an API for the AirAssure Web Software that will enable customers to extract data from AirAssure Web to use on any other interface of their choice. AirAssure Web is TSI software that enables customers to securely and easily configure multiple meters remotely and also collect aggregate, access, analyze and publish the data read by the AirAssure PM2.5 monitors. This note describes the elements of the API for customers to be able to extract specific data from AirAssure Web.

- Sections 3.2 - Data Controller table, all the entries that start with **/data**

- Section 4 - JSON Data Query and Display Format

- Section 4 - Date format

- Section 5.3 - All

- Other - You also need a note to say that all the URLs that start /data require an authentication session for a user with ADMIN privileges.

The Java based APIs published in this document give the user the ability to accomplish the following:

- Download all data readings as a data dump (helps in moving AirAssure Web data to another database like SQL or Access)

- Downloads all data readings for single instrument (helps in moving AirAssure Web data to another database like SQL or Access)

- Downloads the last data reading of all instruments (for display on a dashboard)

- Downloads the last data reading of single instrument (for display on a dashboard)

- Downloads data readings based on query data (for display on a dashboard or analytical tools)

- Clears all collected data on AirAssure Web for every instrument (useful to periodically clear AirAssure Web memory)

Please note that all the URLs published below require an authentication session for a user with Admin privileges. All values (except for download of all data) returned will be in JSON.

The APIs to be used are listed in the table below:

URL	Request Type	Action	Data	Returns
/data/getJson	GET	Downloads all data readings as a data dump (ADMIN Role only)	None	JSON
/data/getCsv	ANY (GET or POST)	Downloads all data readings (ADMIN Role only)	None	CSV
/data/all/getJson	GET	Downloads all data readings (ADMIN Role only)	None	JSON
/data/{serial}/getJson	GET	Downloads all data readings for single instrument (ADMIN Role only)	None	JSON
/data/current/getJson	GET	Downloads the last data reading of all instruments (ADMIN Role only)	None	JSON
/data/{serial}/current/getJson	GET	Downloads the last data reading of single instrument (ADMIN Role only)	Serial Number	JSON
/data/getJson	POST	Downloads data readings based on query data (ADMIN Role only)	JSON	JSON
/data/clear	Any (GET or POST)	Clears all collected data for every instrument (ADMIN Role only)	None	Redirect to /configure
/data/delete	Any (GET or POST)	Deletes all instruments and data (ADMIN Role only)	None	Redirect to /configure

The relevant Data Query and Display formats are described in more detail to follow.

JSON Data Query and Display Format

JSON formats have been developed to query for and return simplified reading data. It was designed to support multiple instruments/readings in a single message with the return data framed with the list designators [ ]. The query data takes an array of instrument serial numbers and two scalar values for the from and to date. The return structure is flat and each parameter is a name-value pair in the JSON message. This may be evolved over time to allow hierarchical data expression but it is not needed at this time.

Query Data Structure

```
{
        "serialNumbers": [
                "<Serial Number>",
                "<Serial Number>",

                …
                <Serial Number>
        ],
        "fromReadingDate": "<Date format[6]>",
        "toReadingDate": "<Date format[6]>"
}
```

Parameter	Description	Required	Format
serialNumbers	An array of instrument serial numbers. A serial number of all can be specified.	YES	Array of Strings
fromReadingDate	The lower bound date limit	YES	Date format[6]
toReadingDate	The upper bound date limit	YES	Date format[6]

Display Data Structure

```
[
        {
                "serial": "<Serial Number>",
                "reading": "<Reading>",
                "displayStatus": "<AQI color>",
                "readingDate": "<Date format[6]>",
        },
        {
                "serial": "<Serial Number>",
                "reading": "<Reading>",
                "displayStatus": "<AQI color>",
                "readingDate": "<Date format[6]>",
        },
        {

        }
]
```

Parameter	Description	Format
serial	The instrument's serial number	String
reading	The instrument's reading	String
displayStatus	The AQI color the instrument would display for the reading	green|yellow|orange|red|purple|maroon
readingDate	The date the reading was taken	Date format[6]

CSV File Format

A CSV format has been developed to download instrument and reading data. It was designed to have all the instrument data as a standalone record per data point and includes the following tab separated fields:
Serial, Instrument Location, Reading Date, Reading Time, Reading, AQI Color, Correction Factor, Calibration Coefficients, IP Address

Example:
1229077   1/11/15   15:03:22   142   red   1.0   1.0/1.0/1.0/1.0
10.1.0.78/255.255.255.0/10.1.0.1

Date format

The date format used for querying and in the display output is defined by
MM/DD/YYYY hh:mm:ss AM/PM
Example:
06/01/2015 4:30:05 PM

Administrator Access

The user needs to log in to get a session and then use the session ID when making the calls to get data.

Functionally the steps are:

1. Do a form post with username and password to log in—view source on the login page will show the form parameters and URL to post to. This is to authenticate.

2. The application will respond with a sessions cookie called JSESSIONID. Save the value in a variable. This establishes the authentication session.

3. When making the data request set the JSESSIONID cookie for the request or include a URL parameter called JSESSIONID with the session identifier in it. This is how a call is made in an authenticated session.

Languages

The AirAssure Web application supports the following languages:

*   US English
*   Chinese

To switch languages a user selects the appropriate language icon on the menu en, zh. Individual requests can be language selected by passing a query parameter of ?locale=<lang> i.e. ?locale=en, ?locale=zh. To view all the instruments with Chinese language selection the URL would be /instrument/list?locale=zh

Once a language has been selected it will be remembered for the duration of the web session.

UNDERSTANDING, ACCELERATED

TSI Incorporated – Visit our website **www.tsi.com** for more information.

USA	**Tel:** +1 800 874 2811	**India**	**Tel:** +91 80 67877200
UK	**Tel:** +44 149 4 459200	**China**	**Tel:** +86 10 8219 7688
France	**Tel:** +33 1 41 19 21 99	**Singapore**	**Tel:** +65 6595 6388
Germany	**Tel:** +49 241 523030		